Security Target
BSI-DSZ-CC-0203

Version 1.1

January 24th, 2003

Evaluation of the Philips P16WX064V0C Secure 16-bit Smart Card Controller

Developed and provided by

Philips Semiconductors, Business Unit Identification

According to the Common Criteria for Information Technology Evaluation (CC) at Level EAL5 augmented

by

Philips Semiconductors GmbH
Stresemannallee 101
22505 Hamburg

Document Information

Document History

Version	Date	Changes	Remarks
1.0	January 06th, 2003		Final Version
1.1	January 24th, 2003	update F.LOG in section 6.1 and 8.3.1	

Latest version is: Version 1.1 (January 24th, 2003)

Document Invariants

Name	Value (to be edited)	Test Output (to copy)
File name and length	Automatically	st_smartxa2_v1_1.doc
Latest version	Version 1.1	Version 1.1
Date of this version	January 24th, 2003	January 24th, 2003
Classification	Security Document – Strictly Confidential	Security Document – Strictly Confidential
TOE name (long)	Philips P16WX064V0C Secure 16-bit Smart Card Controller	Philips P16WX064V0C Secure 16-bit Smart Card Controller
TOE name (short)	P16WX064V0C	P16WX064V0C
Developer (long)	Philips Semiconductors, Business Unit Identification	Philips Semiconductors, Business Unit Identification
Developer (short)	Philips	Philips
Sponsor (long)	Philips Semiconductors, Business Unit Identification	Philips Semiconductors, Business Unit Identification
Sponsor (short)	Philips	Philips
Certification ID	BSI-DSZ-CC-0203	BSI-DSZ-CC-0203
list of authors	Hans-Gerd Albertsen	Hans-Gerd Albertsen
certific. body (long)	Bundesamt für Sicherheit in der Informationstechnik	Bundesamt für Sicherheit in der Informationstechnik
certific. body (short)	BSI	BSI

Table of Contents

1 ST Introduction

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

1.1 ST Identification

This Security Target (st_smartxa2_v1_1.doc, Version 1.1, January 24th, 2003) refers to the "Philips P16WX064V0C Secure 16-bit Smart Card Controller" (TOE) provided by Philips Semiconductors, Business Unit Identification for a Common Criteria evaluation.

1.2 ST Overview

1.2.1 Introduction

The TOE is the hardware of the microcontroller chip P16WX064V0C of the Smart Card Controller IC family produced by Philips. The TOE includes also IC Dedicated Test Software for test purposes stored in the Test-ROM of the microcontroller. The Smart Card Controller hardware comprises a 16-bit processing unit, volatile and non-volatile memories accessible via a memory management unit, cryptographic co-processors, security components and I/O ports.

The TOE includes a Data Sheet and the Guidance Document, both for the P16WX064V0C. This documentation contains a description of the architecture, the secure configuration and usage of the chip by the Smartcard Embedded Software and the instruction set.

The security measures of the P16WX064V0C are designed to act as an integral part of the complete security system in order to strengthen the design as a whole. Several security measures are completely implemented and controlled in the hardware. Other security measures are controlled by the hardware and allow a configuration by software or software guided exceptions. With the three modes of operation and the memory management unit the TOE is intended to support multi-application projects.

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

Hence the TOE shall

- maintain the integrity and the confidentiality of data stored in the memories of the TOE and
- maintain the different modes of operation with the related capabilities for configuration and memory access and

- maintain the integrity, the correct operation and the confidentiality of security functions (security mechanisms and associated functions) provided by the TOE.

These features are ensured by the construction of the TOE and the security functions it provides. The "Philips P16WX064V0C Secure 16-bit Smart Card Controller" (TOE) mainly provides a hardware platform for a smart card with

- functions to calculate the Data Encryption Standard (Triple-DES) with two keys,
- support for large integer arithmetic (multiplication, addition and logical) operations,
- a random number generator,
- memory management control features,
- cyclic redundancy check calculation (CRC) and
- I/O ports with different behaviours (UART and USB port)
- control of the TOE mode regarding a Test Mode and an Application Mode.

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

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

1.2.2 Life-Cycle

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

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

Regarding the Application Note 1 of [7] the TOE supports the authentic delivery using the Fabkey feature (refer to the Data Sheet, P16WX064 SmartXA-Family, Secure 16-bit Smart Card Controller and the Guidance, Delivery and Operation Manual for the P16WX064).

Security during Development and Production

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

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

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

The test process of every die is performed by a test centre of Philips. Delivery processes between the involved Philips sites provide accountability and traceability of the produced wafers. Non-functional ICs are marked on the wafer but will be delivered on the wafer if wafers are ordered by the customer.

1.2.3 Specific Issues of Smartcard Hardware and the Common Criteria

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

1.3 CC Conformance and Evaluation Assurance Level

The evaluation is based upon

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; Version 2.1, August 1999, [1]
- Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements; Version 2.1, August 1999, [2]
- Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; Version 2.1, August 1999, [3]

For the evaluation the following methodology will be used:

- Common Methodology for Information Technology Security Evaluation CEM-99/045 Part 2: Evaluation Methodology, Version 1.0, August 1999, [4]

The chosen level of assurance is **EAL 5 augmented**. The minimum strength level for the TOE security functions is **SOF-high (Strength of functions high)**.

This Security Target claims the following CC conformances:

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

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

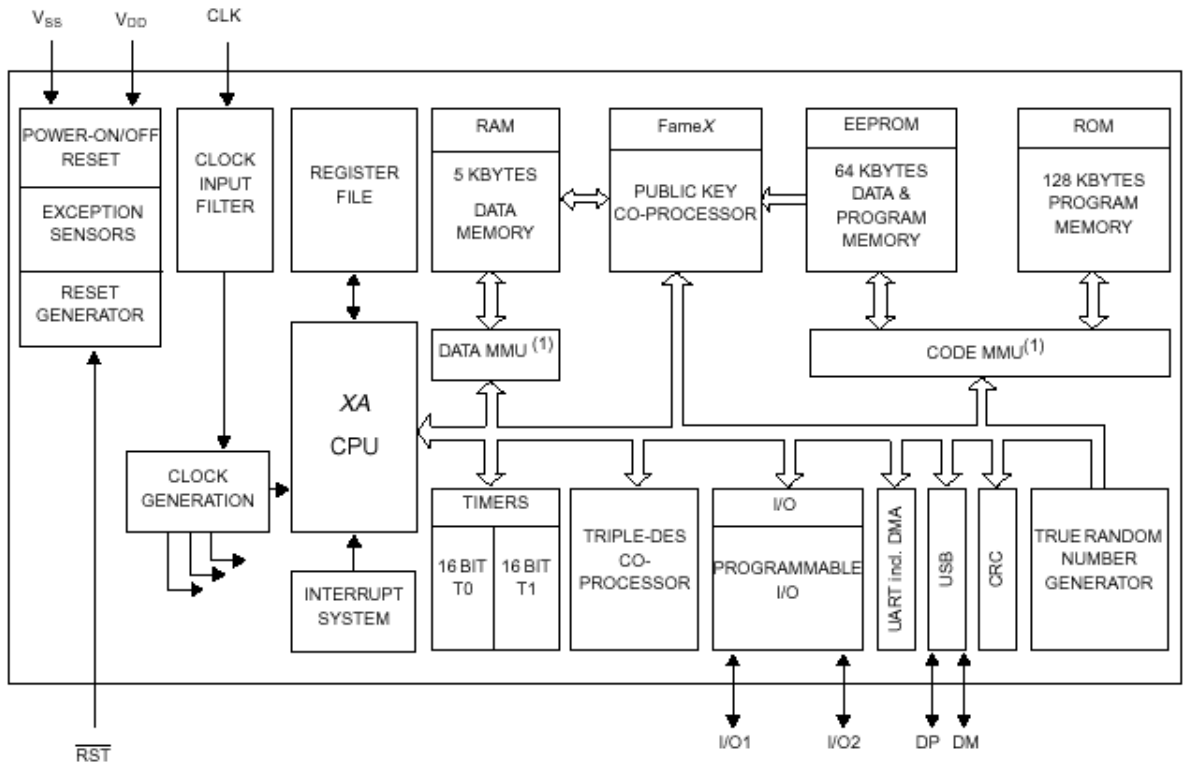
Note: The "Smartcard IC Platform Protection Profile", [7] requires the assurance level EAL4 augmented. Regarding the Application Note 3 of [7] the changes which are needed for EAL5 are described in the different relevant sections of this Security Target.

2 TOE Description

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

2.1 TOE Definition

The Target of Evaluation (TOE) is the smartcard integrated circuit depicted in figure 1 as block diagram. The TOE named P16WX064V0C is manufactured in an advanced CMOS process. The TOE includes IC Designer/Manufacturer proprietary IC Dedicated Test Software. All other software is called Smartcard Embedded Software and is not part of the TOE.



(1) MMU = Memory Management Unit

Figure 1: Block Diagram of the P16WX064V0C

The following table lists the TOE components.

Type	Name	Release	Date	Form of delivery
Hardware	Philips P16WX064V0C Secure 16-bit Smart Card Controller	V0C	C013C.gds2_20020926(GDS2 File)	wafer (dice include reference C013C)
Software	Test ROM Software (the *IC Dedicated Test Software*)	2.1	03.09.2001	Test ROM on the chip *(Testrom_os_xsa_080801.lst)*
Document	Data Sheet, P16WX064 SmartXA-Family, Secure 16-bit Smart Card Controller	3.1	November 29th, 2002	electronic document
Document	Instruction Set P16WX064 SmartXA-Family, Secure 16-bit Smart Card Controller	3.0	July 5th, 2002	electronic document
Document	Guidance, Delivery and Operation Manual for the P16WX064			electronic document

Table 1: Components of the TOE

2.1.1 Hardware Description

The CPU of the P16WX064V0C has a 16-bit architecture with an instruction set that is extended from the 80C51 family instruction set. The first and in some cases the second byte of an instruction are used for operation encoding. The P16WX064V0C distinguishes between three modes of operations with different privileges: System Mode, Meta Mode and User Mode. The System Mode provides unlimited access to the hardware components. For the Meta Mode all hardware components are accessible except the Code Memory Management Unit. In the User Mode the access is restricted to the CPU and specific Special Function Register. The on-chip hardware components are controlled by the Smartcard Embedded Software via Special Function Registers. These registers are correlated to the activities of the CPU, the memory management unit, interrupt control, I/O configuration, EEPROM, timers, UART, USB and the two co-processors. The communication with the P16WX064V0C can be performed through an UART, the direct usage of a I/O port or the USB interface. The P16WX064V0C provides four different types of interrupts: (i) exceptions, (ii) software traps, (iii) hardware events and (iv) software interrupts. These interrupts force the jump to specific fixed vector addresses in the ROM. Every different interrupts can therefore be controlled and guided by a specific part of Smartcard Embedded Software. In conjunction with the jump to a specific fixed vector address the hardware always enables the System Mode. Therefore the handling of the interrupts is supported by the separation between the modes of operation.

The device includes ROM (128 kByte User-ROM + 12 kByte Test-ROM), RAM (5152 Byte) and EEPROM (64 kByte) memory. The access control by the memory management unit for code is related to the ROM and the EEPROM. The access control by the memory management unit for data is related to the RAM. Nevertheless the EEPROM can be accessed as data memory as well as program memory. Smartcard Embedded Software running in the System Mode has unlimited access to the memories. In the Meta Mode the Smartcard Embedded Software is not allowed to make modifications of the configuration of the code memory management unit. Smartcard Embedded Software running in the User Mode can not make configuration changes to the memory management.

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

The P16WX064V0C operates with a single 3V or 5V nominal power supply except the power supply for the USB operation that must be nominal 5V. The nominal maximum external clock frequency is 6 MHz. The micro controller can be operated with the internal clock especially to decrease the calculation time for security algorithms. The controller provides power saving modes with reduced activity: the IDLE Mode and the SLEEP Mode, which includes the CLOCK STOP Mode.

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

2.1.2 Software Description

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

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

The so called micro code is part of the CPU and used to decode the CPU instructions. This micro code is a special software related to the internal CPU functionality and realised as logic. This logic is implemented using the same properties as used for the CPU.

2.1.3 Documentation

The Data Sheet [9] of the P16WX064V0C is also part of the TOE. It contains a functional description needed to develop software and guidelines for the use of security features. The instruction set of the TOE is described in a separate document [10]. Additional Guidance describe aspects of the program interface and the use of programming techniques to improve the security [11]. The provided documentation can be used by the software developer to develop the Smartcard Embedded Software.

2.1.4 Interface of the TOE

In the Application Mode the electrical interface of the TOE are the pads to connect the lines power supply, reset input, clock input, ground, UART, USB and I/O2.

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

- In the Test Mode (used before delivery of the TOE after production) 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 Application Mode (used after TOE Delivery) the software interface is the set of instructions, the bits in the special function registers that are related to the Application Mode 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 mode of operation (system, meta or user) configured by the Smartcard Embedded Software.

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

2.1.5 Life Cycle and Delivery of the TOE

For the usage phase the P16WX064V0C chip will be implemented in a credit card sized plastic card (micro-module embedded into the plastic card) or another sealed package. The chip provides a hardware computing platform to applications and multiple applications executed by a smart

card operating system. Smart card applications will be used to store secret data and calculate cryptographic functions.

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

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

Regarding the Application Note 5 of [7] Philips will deliver the TOE without IC Dedicated Support Software. The IC Dedicated Test Software stored in the Test-ROM is disabled before TOE Delivery and cannot be used in the following phases.

The TOE is able to control two different logical phases. After production the chip is in the Test Mode that means under the control of the IC Dedicated Test Software. At the end of the production test the chip will be switched into the Application Mode so that the chip is under the control of the Smartcard Embedded Software.

2.1.6 TOE Intended Usage

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

The device is developed for most high-end safeguarded applications, and is designed for embedding into chip cards according to ISO 7816 [14]. Usually the smart card is assigned to a single individual only although the smartcard may be expected to be used for multiple applications in a multi-provider environment. Therefore the TOE may store and process secrets of several systems that must be protected from each other. So the TOE must meet security requirements to be applied to security modules. The secret data shall be used as input for the calculation of authentication data, the calculation of signatures and the encryption of data and keys.

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

2.1.7 TOE User Environment

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

In the end-user environment (phase 7) Smartcard ICs are used in a wide range of applications to assure authorised conditional access. Examples of such are Pay-TV, Banking Cards, Portable communication SIM cards, Health cards, Transportation cards. The end-user environment

therefore covers a wide spectrum of very different functions, thus making it difficult to avoid and monitor any abuse of the TOE.

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

2.1.8 General IT features of the TOE

The TOE IT functionality consists of:

- tamper resistant data storage
- control of operation conditions to provide correct operation in the specified range
- basic cryptographic functions (Triple-DES co-processor)
- basic arithmetic functions for large integer numbers (FameX co-processor for the calculation of asymmetric crypto algorithms)
- physical random number generator
- memory management to separate different applications
- data communication

2.2 Further Definitions and Explanations

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

3 TOE Security Environment

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

3.1 Description of Assets

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

The assets regarding the threats are:

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

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

3.2 Assumptions

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

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

Table 2: Assumptions defined in the Protection Profile

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

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

The following assumption considers the Application Notes 8 and 9 of [7] related to the specialised encryption hardware of the P16WX064V0C (refer to the augmentation paper [8]).

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

A.Key-Function Usage of Key-dependent Functions

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

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

3.3 Threats

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

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

Table 3: Threats defined by the Protection Profile

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

3.4 Organisational Security Policies

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

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

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

P.Add-Components Additional Specific Security Components

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

- Triple DES encryption and decryption
- Area based Memory Access Control
- Special Function Register Access Control.

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

4 Security Objectives

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

4.1 Security Objectives for the TOE

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

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

Table 4: Security objectives defined in the PP

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

O.HW_DES3 Triple DES Functionality

The TOE shall provide the cryptographic functionality to calculate a Triple DES encryption and decryption to the Smartcard Embedded Software. The TOE supports directly the calculation of Triple DES with two 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.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 Mode of Operation (System Mode, Meta Mode or User Mode) and the configuration of the Memory

Management Units (MMU) if the requested type of access to the memory area addressed by the operands in the instruction is allowed.

O.SFR_ACCESS Special Function Register Access Control

The TOE shall provide access control to the Special Function Registers based on the Mode of Operation (System Mode, Meta Mode or User Mode). The access control is used to restrict access to the Memory Management Units and all Specialised Components of the TOE.

The administration of the access conditions for ROM and EEPROM shall be restricted to code running in System Mode. The administration of the access conditions for RAM shall be restricted to code executed in System Mode or Meta Mode.

The access to specialised hardware components of the TOE shall be restricted to code running in System Mode or Meta Mode.

O.RANGE_CHK Value Range Check

The TOE shall provide a range check for Special Pointer Registers. The range check comprises checking a lower and an upper bound for the value stored in the register. A violation of the allowed range shall interrupt the running code and allow an exception handling.

4.2 Security Objectives for the Environment

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

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

Table 5: Security objectives for the environment, taken from the PP

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

The TOE supports cipher schemes as additional specific security functionality. If required the Smartcard Embedded Software shall use these cryptographic services of the TOE and their interface as specified. When key-dependent functions implemented in the Smartcard Embedded

Software are just being executed, the Smartcard Embedded Software must provide protection against disclosure of confidential data (User Data) stored and/or processed in the TOE by using the methods described under "Inherent Information Leakage (T.Leak-Inherent)" and "Forced Information Leakage (T.Leak-Forced)".

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

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

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

By definition cipher or plain text data and cryptographic keys are User Data. The Smartcard Embedded Software shall treat these data appropriately, use only proper secret keys (chosen from a large key space) as input for the cryptographic function of the TOE and use keys and functions appropriately in order to ensure the strength of cryptographic operation.

This means that keys are treated as confidential as soon as they are generated. The keys must be unique with a very high probability, as well as cryptographically strong. For example, if asymmetric algorithms are used, it must be ensured that it is not possible to derive the private key from a related public key using the attacks defined in this Security Target. If keys are imported into the TOE and/or derived from other keys, quality and confidentiality must be maintained. This implies that appropriate key management has to be realised in the environment.

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

Check of initialisation data

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

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

5 IT Security Requirements

5.1 TOE Security Requirements

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

5.1.1 TOE Security Functional Requirements

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

5.1.1.1 SFRs of the Protection Profile

Table 6 below shows all SFRs which are specified in the Protection Profile Smartcard IC Platform Protection Profile [7] (in the order of definition in the PP). Some of the SFRs are CC Part 2 extended and defined in the Protection Profile. This is shown in the third column of the table.

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

Table 6: SFRs taken from the PP

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

FCS_RND.1 Quality metric for random numbers

FCS_RND.1.1 The TSF shall provide a mechanism to generate random numbers that meet *the requirement to provide an entropy of at least 7 bit in each byte*[1].

Dependencies: No dependencies.

Note: The entropy of the random number is measured by the Shannon-Entropy as follows:

$E = -\sum_{i=0}^{255} p_i \cdot \log_2 p_i$, 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.

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

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

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

5.1.1.2 Additional SFRs

Considering the Application Note 15 of [7] in the following paragraphs the additional functions for cryptographic support and access control are defined.

The following table lists the additional SFRs. These SFRs are not required in the Protection Profile.

SFR	Title	Defined in ...
FCS_COP.1	Cryptographic operation	CC, Part 2
FDP_ACC.1[MEM]	Subset access control	CC, Part 2
FDP_ACC.1[SFR]	Subset access control	CC, Part 2
FDP_ACF.1[MEM]	Security Attribute based access control	CC, Part 2
FDP_ACF.1[SFR]	Security Attribute based access control	CC, Part 2
FMT_MSA.3[MEM]	Static attribute initialisation	CC, Part 2

[1] [assignment: a defined quality metric]

SFR	Title	Defined in ...
FMT_MSA.3[SFR]	Static attribute initialisation	CC, Part 2
FMT_MSA.1[MEM]	Management of security attributes	CC, Part 2
FMT_MSA.1[SFR]	Management of security attributes	CC, Part 2
FMT_SMF.1	Specification of Management Functions	CC, Part 2[2]
FRU_VRC.1	Simple value range check	Section 9.1

Table 7: Additional SFRs

Additional SFR regarding cryptographic functionality

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

FCS_COP.1 Cryptographic operation

Hierarchical to: No other components.

FCS_COP.1.1 The TSF shall perform *encryption and decryption* [3] in accordance with a specified cryptographic algorithm *Triple Data Encryption Algorithm (TDEA)* [4] and cryptographic key sizes *of 112 bit* [5] that meet the following *list of standards* [6]:

FIPS PUB 46-3 FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION DATA ENCRYPTION STANDARD (DES) Reaffirmed 1999 October 25, keying option 2.

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

[2] Refer to Final Interpretation 065

[3] [assignment: list of cryptographic operations]

[4] [assignment: cryptographic algorithm]

[5] [assignment: cryptographic key sizes]

[6] [assignment: list of standards]

Additional SFRs regarding access control

Access Control Policy

The hardware shall provide different modes of operation to an Smartcard Embedded Software. The management of access to code and data as well as the configuration of the hardware shall be performed in a dedicated mode. The hardware shall provide a mode that ensures the separation between different applications running on the TOE. An application shall not be able to access Specialised Components directly to support the separation of applications. The functions used by the IC Dedicated Test Software to test the chip shall not be available to the Smartcard Embedded Software.

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

The subjects are

- **Smartcard Embedded Software** i.e. data in the memories of the TOE executed as instructions by the CPU

The objects are

- the **memories** (ROM, EEPROM and RAM) consisting of
  - the **data in the Code Memory Areas** defined by the Code Memory Management Unit (Code MMU) in ROM and EEPROM
  - the **data in the Data Memory Areas** defined by the Data Memory Management Unit (Data MMU) in RAM
- the **Special Function Registers** consisting of
  - Special Function Registers to configure the Code MMU
  - Special Function Registers to configure the Data MMU
  - Special Function Registers related to Specialised Components
  - Special Function Registers related to General CPU Functions

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 read and/or write access to a SFR may be not allowed depending on the function of the register, on the mode of operation or on the TOE mode to enforce the access control police or ensure a secure operation.)

The security attributes are:

- Mode of Operation: There are three different mode of operation based on the configuration of the Special Function Register "Program Status Word" defining whether the instruction is executed in the System Mode, Meta Mode and User Mode.
- TOE mode: The TOE mode depends on the life cycle phase of the TOE. For the production test the Test Mode is used. After the production test within the usage phase the Application Mode is used. It is not possible to switch back from the Application Mode into the Test Mode. Both modes provide the three different modes of operation System Mode, Meta Mode and User Mode in the same way.
- Special Function Registers to configure the Code MMU: Configuration of the Code MMU comprising access rights (read, write, execute and enabled/disabled), the virtual code memory base address of the first and last valid block, and the relocation offset to the physical memory location for each of 12 possible Code Memory Areas.
- Special Function Registers to configure the Data MMU: Configuration of the Data MMU comprising access rights (read, write and enabled/disabled), the virtual data memory base address of the first and last valid block, and the relocation offset to the physical memory location for each of 4 possible Data Memory Areas.

The operation "enabled/disabled" of the Code MMU and Data MMU does not define a new operation beyond read, write and execute, but is an implementation detail that allows faster context switching. In "enabled" Code/Data Memory Areas, the MMU uses the values of all other attributes to allow or to deny access. In "disabled" Code/Data Memory Areas, the MMU only denies any access regardless of the values of all other attributes.

Note: A Code/Data Memory Area will be disabled for use if the virtual code/data memory base address of the last valid block is lower than the address of first valid block.

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

FDP_ACC.1[MEM] Subset access control

Hierarchical to: No other components.

FDP_ACC.1.1 The TSF shall enforce the *Access Control Policy*[7] on *all code running on the TOE, all memories and all memory operations*[8].

[7] [assignment: access control SFP]

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

Dependencies: FDP_ACF.1 Security attribute based access control

Application Note: The Access Control Policy shall be enforced by implementing a Code MMU and a Data MMU, which map virtual addresses to physical addresses. The CPU always uses virtual addresses, which are mapped to physical addresses by the MMU. Prior to accessing the respective memory at the physical address, the respective MMU checks if the access is allowed.

FDP_ACC.1[SFR] Subset access control

Hierarchical to: No other components.

FDP_ACC.1.1 The TSF shall enforce the *Access Control Policy*[9] on *all code running on the TOE, all Special Function Registers, and all SFR operations*[10].

Dependencies: FDP_ACF.1 Security attribute based access control

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

FDP_ACF.1[MEM] Security attribute based access control

Hierarchical to: No other components.

FDP_ACF.1.1 The TSF shall enforce the *Access Control Policy*[11] to objects based on *the Mode of Operation, the Special Function Registers to configure the Code MMU and the Special Function Registers to configure the Data MMU*[12].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

Code executed in the System Mode

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

Code executed in the Meta Mode

---

9 [assignment: access control SFP]

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

11 [assignment: access control SFP]

12 [assignment: security attributes, named groups of security attributes]

- *has read and/or execute access to code/data in the User-ROM controlled by the Code MMU,*
- *has read, write and/or execute access to code/data in the EEPROM controlled by the Code MMU,*
- *has read and write access to all data in RAM controlled by the Data MMU,*

Code executed in the User Mode

- *has read and/or execute access to code/data in the User-ROM controlled by the Code MMU,*
- *has read and/or write and/or execute access to code/data in the EEPROM controlled by the Code MMU,*
- *has read and/or write access to data in RAM controlled by the Data MMU*[13]

Implications of the Access Control Policy

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

- Code executed in the System Mode can administrate the configuration of Code MMU and Data MMU, because it has access to all Special Function Registers.
- Code executed in the Meta Mode can administrate the configuration of the Data MMU, but not the Code MMU, because it has access to all Special Function Registers with exception of the Special Function Registers to configure the Code MMU.
- Code executed in the User Mode cannot administrate the configuration of both Code MMU and Data MMU, because it has only access to the Special Function Registers related to General CPU Functions.

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *Code running in System Mode has unrestricted access to all memories*[14].

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the rules: *disabled Code/Data area*[15].

---

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

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

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

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialisation

Application Note: The explicitly authorised access according to FDP_ACF.1.3 shall be implemented in System Mode by switching the Code and the Data MMU to a transparent behaviour, which means that the virtual address is mapped one-to-one to the physical address without controlling access to the address.

FDP_ACF.1[SFR] Security attribute based access control

Hierarchical to: No other components.

FDP_ACF.1.1 The TSF shall enforce the *Access Control Policy*[16] to objects based on *the Mode of Operation and the TOE mode*[17].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

(i) *The code executed in System Mode is allowed to access all Special Function Registers.*

(ii) The code executed in the Meta Mode is allowed to access all Special Function Registers except the Special Function Registers to configure the Code MMU where only read access is allowed.

(iii) The code executed in the User Mode is only allowed access to the Special Function Registers related to General CPU Functions[18].

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *Within the Test Mode the IC Dedicated Test Software running in the System Mode or in the Meta Mode is allowed to read and write additional SFR for test purposes*[19].

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the rules: *Smartcard Embedded Software executed in the Meta Mode is not allowed to write the SCR Register. Smartcard Embedded Software executed in the System Mode or Meta Mode is not allowed to directly modify the bits 6 and 7 of the PSWH Register. Within the Application Mode the Smartcard*

---

16 [assignment: access control SFP]

17 [assignment: security attributes, named groups of security attributes]

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

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

Embedded Software executed in any mode of operation is not allowed to read and write additional SFR for test purposes[20].

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialisation

Application Note: The access control is enforced regardless of the access as 16-bit or 8-bit Special Function Register. Modifying bits 6 and 7 of the PSWH Register with an 8-bit access is equivalent to modifying bits 14 and 15 of the PSW Registers with an 16-bit access.

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

FMT_MSA.3[MEM] Static attribute initialisation

Hierarchical to: No other components.

FMT_MSA.3.1 The TSF shall enforce the *Access Control Policy*[21] to provide *permissive*[22] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow *the Smartcard Embedded Software*[23] to specify alternative initial values to override the default values when an object or information is created.

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

Application Note: Permissive means here that the reset values of the Special Function Register do not provide any restrictions. The SFR of the Code/Data memory management units must be configured after reset by the Smartcard Embedded Software. In addition the Smartcard Embedded Software must define and maintain security attributes for all objects generated by it.

FMT_MSA.3[SFR] Static attribute initialisation

Hierarchical to: No other components.

---

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

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

[22] [selection: restrictive, permissive, other property]

[23] [assignment: the authorised identified roles]

FMT_MSA.3.1 The TSF shall enforce the *Access Control Policy*[24] to provide *restictive*[25] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow *no subject*[26] to specify alternative initial values to override the default values when an object or information is created.

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

Application Note: Here restrictive means that all exceptions including the reset are set up by the hardware in System Mode with disabled MMU for Code and data. Thereby the selection of the dedicated entry in the vector table and the complete control of the TOE is ensured. Nevertheless the developer of the Smartcard Embedded Software is able to run the assigned exception routine in any Mode of Operation as configured in the vector table.

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

FMT_MSA.1[MEM] Management of security attributes

Hierarchical to: No other components.

FMT_MSA.1.1 The TSF shall enforce the *Access Control Policy*[27] to restrict the ability to *modify*[28] the security attributes *Special Function Registers to configure the Code MMU and Special Function Registers to configure the Data MMU*[29] to *code executed in the System Mode or Meta Mode*[30].

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

Application Note: Code executed in the System Mode is able to modify the Special Function Registers to configure the Code MMU and the Special Function Registers to configure the Data MMU, code executed in the Meta Mode is only able to modify the Special Function Registers to configure the Data MMU.

---

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

[25] [selection: restrictive, permissive, other property]

[26] [assignment: the authorised identified roles]

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

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

[29] [assignment: list of security attributes]

[30] [assignment: the authorised identified roles]

FMT_MSA.1[SFR]	Management of security attributes
Hierarchical to:	No other components.
FMT_MSA.1.1	The TSF shall enforce the *Access Control Policy*[31] to restrict the ability to *modify*[32] the security attributes *Mode of Operation*[33] to *the hardware executed on behalf of an exception or interrupt*[34].
Dependencies:	[FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
Application Note:	The Mode of Operation is coded in the register Program Status Word. The relevant bits 6 and 7 of the PSWH can only be changed directly by the hardware based on an exception or interrupt.

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	The TSF shall be capable of performing the following security management functions: *Change of the Mode of Operation by invoking an exception or interrupt* *Change of the Mode of Operation by finishing an interrupt (with a RETI instruction executed in System Mode or Meta Mode)* *Modification of the Code Memory Management Unit Attributes* *Modification of the Data Memory Management Unit Attributes* *Modification of the clock settings and power configuration*[35]
Dependencies:	No dependencies

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

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

[33] [assignment: list of security attributes]

[34] [assignment: the authorised identified roles]

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

Application Note: A separation of the Specification of Management Functions based on the Iterations used for FMT_MSA.1 that requires this security functional requirement is not needed because all management functions rely on the same features implemented in the hardware.

The Mode of Operation is changed at the time the interrupt is invoked by loading a new value for the PSW Register from the interrupt vector table. Similarly, at the time the interrupt is finished by executing the RETI instruction, a previously saved value for the PSW Register is loaded from the stack.

The TOE shall meet the requirement "Simple value range check (FRU_VRC.1)" as specified below.

FRU_VRC.1	Simple value range check
Hierarchical to:	No other components
FRU_VRC.1.1	The TSF shall enforce a range checking for the value of the following resources: *R15/AP1 CPU register and CSFVAL Special Function Register*[36].
FRU_VRC.1.2	The TSF shall notify *the related exception*[37] that the value is out of the defined range.
Dependencies:	No dependencies.

5.1.1.3 SOF claim for TOE security functional requirements

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

5.1.2 TOE Security Assurance Requirements

Table 8 below lists all security assurance components that are valid for this Security Target. These security assurance components are required by EAL5 (see section 1.3) or by the Protection Profile.

Considering the Application Note 18 of [7] the column "Required by" shows the differences in the requirements of security assurance components between the PP and the Security Target. The entry "EAL5 / PP" denotes that a SAR is required by both EAL5 and the requirement of the PP, "EAL5" means that this requirement is due to EAL5 and beyond the requirement of the PP, and

[36] [assignment: controlled resources]

[37] [assignment: the authorised identified roles]

"PP" identifies this component as a requirement of the PP which is beyond EAL5. The Security Target does not include additional augmentations. The refinements of the PP "Smartcard IC Platform Protection Profile" that must be adapted for EAL5 are described in section 5.1.3.

SAR	Title	Required by
ACM_AUT.1	Partial CM automation	EAL5 / PP
ACM_CAP.4	Generation support and acceptance procedures	EAL5 / PP
ACM_SCP.3	Development tools CM coverage	EAL5
ADO_DEL.2	Detection of modification	EAL5 / PP
ADO_IGS.1	Installation, generation, and start-up procedures	EAL5 / PP
ADV_FSP.3	Semiformal functional specification	EAL5
ADV_HLD.3	Semiformal high-level design	EAL5
ADV_IMP.2	Implementation of the TSF	EAL5 / PP
ADV_INT.1	Modularity	EAL5
ADV_LLD.1	Descriptive low-level design	EAL5 / PP
ADV_RCR.2	Semiformal correspondence demonstration	EAL5
ADV_SPM.3	Formal TOE security policy model	EAL5
AGD_ADM.1	Administrator guidance	EAL5 / PP
AGD_USR.1	User guidance	EAL5 / PP
ALC_DVS.2	Sufficiency of security measures	PP
ALC_LCD.2	Standardised life-cycle model	EAL5
ALC_TAT.2	Compliance with implementation standards	EAL5
ATE_COV.2	Analysis of coverage	EAL5 / PP
ATE_DPT.2	Testing: low-level design	EAL5
ATE_FUN.1	Functional testing	EAL5 / PP
ATE_IND.2	Independent testing – sample	EAL5 / PP
AVA_CCA.1	Covert channel analysis	EAL5
AVA_MSU.3	Analysis and testing for insecure states	PP
AVA_SOF.1	Strength of TOE security function evaluation	EAL5

SAR	Title	Required by
AVA_VLA.4	Highly resistant	PP

Table 8: Security Assurance Requirements EAL5 and PP augmentations

5.1.3 Refinements of the TOE Security Assurance Requirements

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

Table 9 lists the influences of the refinements of the PP on the ST. Most of the refined security assurance components have the same level in both documents (Protection Profile and Security Target). The following two subsections apply the refinements to ACM_SCP.3 and ADV_FSP.3 which are different between the PP and the ST.

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

Table 9: Security Assurance Requirements, overview of differences of refinements

5.1.3.1 Refinements regarding CM scope (ACM_SCP)

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

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

5.1.3.2 Refinements regarding functional specification (ADV_FSP)

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

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

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

5.2 Security Requirements for the Environment

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

5.2.1 Security Requirements for the IT-Environment

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

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

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

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

Table 10: Security Requirements for the IT Environment

5.2.2 Security Requirements for the Non-IT-Environment

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

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

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

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

RE.Cipher Cipher Schemata

The developers of Smartcard Embedded Software must not implement routines in a way which may compromise keys when the routines are executed as part of the Smartcard Embedded Software. Performing functions which access cryptographic keys could allow an attacker to misuse these functions to gather information about the key which is used in the computation of the function.

Keys must be kept confidential as soon as they are generated. The keys must be unique with a very high probability, as well as cryptographically strong. For example, it must be ensured that it is not possible to derive the private key from a public key if asymmetric algorithms are used. If keys are imported into the TOE and/or derived from other keys, quality and confidentiality must be maintained. This implies that an appropriate key management has to be realised in the environment.

RE.RNG Test of Random Numbers

The developers of Smartcard Embedded Software must implement test routines dependent on the usage of the random number generator. The requirements for testing the random numbers provided by the random number generator are given by the AIS31 and described in the Guidance, Delivery and Operation Manual for the P16WX064.

RE.Check-Init Check of initialisation data

The Card Manufacturer shall use appropriate measures to protect and check a sufficient part of the pre-personalisation data. This shall include at least the Fabkey data that is part of the pre-personalisation data (to prevent the use of Smartcard ICs that are not correctly tested and pre-personalised by the TOE Manufacturer.

6 TOE Summary Specification

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

6.1 TOE Security Functions

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

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

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

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

F.RNG: Random Number Generator

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

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

F.HW_DEA: Triple-DES Co-processor

The TOE provides the Triple Data Encryption Algorithm (TDEA) according to the Data Encryption Standard (DES). F.HW_DEA is a modular basic cryptographic function which provides the TDEA algorithm as defined by FIPS PUB 46 by means of a hardware co-processor and supports the 2-key Triple DEA algorithm according to keying option 2 in FIPS PUB 46-3 [12]. The two 56 bit keys (112 bit) for the 2-key Triple DES algorithm shall be provided by the Smartcard Embedded Software. For encryption the Smartcard Embedded Software provides 8 bytes of the plain text and F.HW_DEA calculates 8 bytes cipher text. The calculation output is read by the Smartcard Embedded Software. For decryption the Smartcard Embedded Software also provides 8 bytes of cipher text and F.HW_DEA calculates 8 bytes plain text. The calculation output is read by the Smartcard Embedded Software.

F.OPC: Control of Operating Conditions

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

The TOE ensures its correct operation and prevents any malfunction using the following sub-functions: filtering of power supply and clock input as well as monitoring of power supply, the frequency of the clock and the temperature of the chip by means of sensors. The thresholds allowed for these parameters are defined within the range where the TOE ensures its correct operation.

If one of the monitored parameters is out of the specified range a reset is forced and the actual running program is aborted. All components of the TOE are initialised with their reset values.

Before TOE delivery the TOE mode is set to Application Mode. In Application Mode the TOE enables the sensors automatically when operated. Furthermore it prevents that the Smartcard Embedded Software disables the sensors.

In addition, the TOE controls the specified range of the System Stack Pointer and the User Stack Pointer. In case the specified limits are reached an exception is generated.

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

F.PHY: Protection against Physical Manipulation

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

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

F.LOG: Logical Protection

The function F.LOG implements measures to limit or eliminate the information that might be contained in the shape and amplitude of signals or in the time between events found by measuring such signals. This comprises the power consumption and signals on the other pads that

are not intended by the terminal or the Smartcard Embedded Software. Thereby this security function prevents the disclosure of User Data or TSF data stored and/or processed in the Smartcard IC through the measurement of the power consumption and subsequent complex signal processing. The protection of the TOE comprises different features within the design that support the other security functions.

The Triple-DES co-processor includes special features to prevent SPA/DPA analysis of shape and amplitude of the power consumption and ensures the same calculation time for all operations.

The FameX co-processor provides measures to prevent timing attacks on basic modular function. The calculation time of one modular operation depends on the lengths of the operands but not on the value of the operands. In addition special features are included to provide limitations of the capability for the analysis of shape and amplitude of the power consumption. Of course the FameX does not realise an algorithm on its own and algorithm-specific leakage countermeasures have to be added for the FameX.

Additional features that can be configured by the Smartcard Embedded Software comprise (i) the secure DCDC-converter that can be used to smooth the power consumption and (ii) several clock configurations that can be used to prevent the possibility to synchronise the internal operation with the external clock or to synchronise with the characteristics of the power consumption that can be used as trigger signal to support leakage attacks (DPA or timing attacks)

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

F.COMP: Protection of Mode Control

The function F.COMP provides a control of the TOE mode for (i) Test Mode and (ii) Application Mode. This includes the protection of electronic fuses stored in a protected memory area. In addition F.COMP provides a write once memory area. All bits in this area can only be set once.

The control of the TOE mode prevents the use of features implemented in the TOE that are used during production/test and that are disabled before the delivery of the TOE. The initial TOE mode is the Test Mode. F.COMP limits the capabilities of the test functions and provides test personnel during Phase 3 with the capability to store the identification and/or pre-personalisation data and/or supplements of the Smartcard Embedded Software in the EEPROM.

The implemented control of the TOE mode ensures that in the Test Mode (i) allows to execute the IC Dedicated Test Software and (ii) prevents to execute the Smartcard Embedded Software. In the Application Mode the TOE (i) allows to execute the Smartcard Embedded Software and (ii) prevents to execute the IC Dedicated Test Software.

The protection of electronic fuses ensures the secure storage of configuration- and calibration data stored in the Test Mode. The TOE allows to change the TOE mode only one time from the Test Mode into the Application Mode. The TOE prevents to change the TOE mode from the Application Mode into the Test Mode.

The write once memory area is erased during the Test Mode to ensure a well defined content. After the switch to the Application Mode the Smartcard Embedded Software is able to read this memory area and to set every bit in this memory area once. The access to the OTP user memory area is only possible in the system mode or in the meta mode. The OTP user memory area is designed to store the identification of a dedicated smartcard or a sequence of events over the life cycle that can be coded by an increasing number of bits set to "one".

The security function F.COMP maintains the security domain for its own execution that protects it from interference and tampering by untrusted subjects both in the Test Mode and in the Application Mode. It also enforces the separation between the security domains of subjects within each mode of operation. The OTP memory is maintained in the same way to ensure the settings stored in that memory area.

F.MEM_ACC: Memory Access Control

F.MEM_ACC controls access of any subject (program code comprising processor instructions) to the memory of the TOE through the Code Memory Management Unit and the Data Memory Management Unit. Thereby the code is also stored in a dedicated memory area. Based on the value of the Special Function Register “Program Status Word” the processor is assigned to (i) System Mode, (ii) Meta Mode or (iii) User Mode. In the System Mode both the Code MMU and Data MMU are transparent. In the Meta Mode both MMU’s are enabled.

Memory access is based on virtual addresses that are mapped to physical addresses. The CPU uses virtual addresses, physical addresses are used to access the memories. The Memory Management Units perform the translation from virtual to physical addresses. The access control is performed by the definition of memory areas with related access rights. It is possible to define several code memory areas at the same time with the access rights read, write, execute and enable/disable. It is possible to define several data memory areas at the same time with the access rights read, write and enable/disable.

A disabled MMU is transparent and performs no address translation and no access control. Instead, the virtual addresses are mapped one-to-one to physical addresses. An enabled MMU performs both tasks.

In addition the memory management units permanently check whether the selected addresses are within the boundary of the physical implemented memory range. Access violations and accesses outside the boundary of the physical implemented memory range are notified by raising an exception.

F.SFR_ACC: Special Function Register Access Control

The function F.SFR_ACC controls access to the Special Function Registers and the switch between the System Mode, the Meta Mode and the User Mode. Based on the value of the Special Function Register “Program Status Word” the processor is assigned to (i) System Mode, (ii) Meta Mode or (iii) User Mode. The access control is as follows:

- In System Mode, all Special Function Registers are accessible.

- In Meta Mode, all Special Function Registers are accessible except the Special Function Registers to configure the Code MMU and the SCR Register that are only readable.
- In User Mode, only the Special Function Registers related to General CPU Functions are accessible.

This implies that the security functions Random Number Generator and Triple-DES Co-processor can only be used and the security function Logical Protection can only be configured in System Mode or Meta Mode. In addition also the FameX co-processor and all Special Function Register of Specialised Components are only accessible in the System Mode or the Meta Mode. Only the Special Function Register related to General CPU Functions including most registers of the Value Range Check are directly accessible in the User Mode. For the Memory Access Control and the Value Range Check refer to the definition of the related security function.

Based on the function of the register, on the mode of operation or on the TOE mode, the read and/or write access for a specific SFR is not allowed (e.g. read access to DES key register or write access to the output register of the random number generator). There are two different implementations to prevent an access. The first method provides an exception and the second method will ignore any operation on the SFR. Ignored means that the write access has no influence and/or that the read access always provides a fixed return value independent of the content of the SFR. One of these two methods is implemented for the affected register.

Regardless of the Mode of Operation the respective bits in the “Program Status Word” which store the Mode of Operation can not be changed by direct access. Instead, the bits can only be changed by invoking an exception or an interrupt or returning from the respective routine. When an interrupt is invoked, the actual state of the PSW register is stored on the stack and a new value is set from the interrupt vector table. When an interrupt is finished, the previously saved value of the PSW is restored.

F.RANGE_CHK: Value Range Check

The function F.RANGE_CHK provides range checking for dedicated registers of the TOE. Range checking comprises checking against lower and upper bounds. Any violation of the allowed range is notified via an interrupt.

Range checking is performed on the following registers:

- R15/AP1 register accessible in System, Meta and User Mode
- CSFVAL Special Function Register write only in System, Meta and User Mode

Note: Although this security function (as a hard-wired functionality) can not be disabled, the range checking can be stopped by setting the lower and upper bound to the minimum and maximum values that the register is able to store. The reset values are 0000h for the lower limit and FFFFh for the upper limit. Therefore the security function must be enabled by loading more restrict values.

SOF claim

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

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

1. The output of the Random Number Generator F.RNG can be analysed with probabilistic methods.

2. The quality of the mechanism contributing to the leakage attacks of F.LOG especially for F.HW_DEA can be analysed using probabilistic methods on power consumption of the TOE.

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

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

6.2 Assurance Measures

Appropriate assurance measures will be employed to satisfy the security assurance requirements defined in section 5.1.1.3. The developer will provide documents containing the measures and further information needed to examine conformance of the measures to the assurance requirements. The following table gives a mapping between the assurance requirements and the documents containing the information needed for the respective requirement either directly or referring to further documents containing this information.

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

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

Table 11: List of documents describing the measures regarding the assurance requirements

7 PP Claims

This Security Target claims conformance to the following Protection Profile:

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

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

8 Rationale

This chapter contains the following sections: "Security Objectives Rationale", "Security Requirements Rationale", "TOE Summary Specification Rationale" and "PP Claims Rationale".

8.1 Security Objectives Rationale

Section 7.1 of the Protection Profile provides a rationale how the assumptions, threats, and organisational security policies are addressed by the objectives that are subject of the PP "Smartcard IC Platform Protection Profile". The following table 12 reproduces the table in section 7.1 of [7].

Assumption, Threat or OSP	Security Objective	Note
A.Plat-Appl	OE.Plat-Appl	(Phase 1)
A.Resp-Appl	OE.Resp-Appl	(Phase 1)
P.Process-TOE	OE.Process-TOE O.Identification	(Phase 2 – 3)
A.Process-Card	OE.Process-Card	(Phase 4 – 6)
T.Leak-Inherent	O.Leak-Inherent	
T.Phys-Probing	O.Phys-Probing	
T.Malfunction	O.Malfunction	
T.Phys-Manipulation	O.Phys-Manipulation	
T.Leak-Forced	O.Leak-Forced	
T.Abuse-Func	O.Abuse-Func	
T.RND	O.RND	

Table 12: Security Objectives versus Assumptions, Threats or Policies

The following Table 13 provides the justification for the additional security objectives. They are in line with the security objectives of the Protection Profile and supplement these according to the additional assumption and organisational security policy.

Assumption/Policy	Security Objective	Note
P.Add-Components	O.HW_DES3 O.MEM_ACCESS O.SFR_ACCESS O.RANGE_CHK	

Assumption/Policy	Security Objective	Note
A.Key-Function	OE.Plat-Appl OE.Resp-Appl	(Phase 1)
A.Check-Init	OE.Check-Init	(Phase 1) and (Phase 4 – 6)

Table 13: Additional Security Objectives versus Assumptions or Policies

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

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

Nevertheless the security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced define how to implement the specific security functionality required by P.Add-Components. These security objectives are also valid for the additional specific security functionality since they must avert the related threats also for the components added related to the policy.

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

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

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

has to be realised 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 initialisation data by the Smartcard Embedded Software (A.Check-Init)" is as follows:

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

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

8.2 Security Requirements Rationale

8.2.1 Rationale for the security functional requirements

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

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

Objective	TOE Security Functional Requirements	Security Requirements for the environment
O.Phys-Manipulation	- FPT_PHP.3 "Resistance to physical attack"	RE.Phase-1 "Design and Implementation of the Smartcard Embedded Software" (e.g. by implementing FDP_SDI.1 Stored data integrity monitoring)
O.Leak-Forced	All requirements listed for O.Leak-Inherent - FDP_ITT.1, FPT_ITT.1, FDP_IFC.1 plus those listed for O.Malfunction and O.Phys-Manipulation - FRU_FLT.2, FPT_FLS.1, FPT_SEP.1, FPT_PHP.3	RE.Phase-1 "Design and Implementation of the Smartcard Embedded Software"
O.Abuse-Func	- FMT_LIM.1 "Limited capabilities" - FMT_LIM.2 "Limited availability" plus those for O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation, O.Leak-Forced - FDP_ITT.1, FPT_ITT.1, FDP_IFC.1, FPT_PHP.3, FRU_FLT.2, FPT_FLS.1, FPT_SEP.1	
O.Identification	- FAU_SAS.1 "Audit storage"	

Objective	TOE Security Functional Requirements	Security Requirements for the environment
O.RND	- FCS_RND.1 "Quality metric for random numbers" plus those for O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation, O.Leak-Forced - FDP_ITT.1, FPT_ITT.1, FDP_IFC.1, FPT_PHP.3, FRU_FLT.2, FPT_FLS.1, FPT_SEP.1	RE.Phase-1 "Design and Implementation of the Smartcard Embedded Software" (e.g. by implementing FPT_AMT.1 "Abstract machine testing")
OE.Plat-Appl		RE.Phase-1 "Design and Implementation of the Smartcard Embedded Software"
OE.Resp-Appl		RE.Phase-1 "Design and Implementation of the Smartcard Embedded Software"
OE.Process-TOE	- FAU_SAS.1 "Audit storage"	Assurance Components: refer to below ✳
OE.Process-Card		RE.Process-Card possibly supported by RE.Phase-1

Table 14: Security Requirements versus Security Objectives

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

The Security Target additionally defines the SFRs for the TOE that are listed in Table 15. In addition one Security Requirement for the Environment is defined. The following table gives an overview, how the requirements are combined to meet the security objectives.

Objective	TOE Security Functional Requirement	Security Requirements for the environment
O.HW_DES3	FCS_COP.1	RE.Phase-1 with RE.Cipher

Objective	TOE Security Functional Requirement	Security Requirements for the environment
O.MEM_ACCESS	FDP_ACC.1[MEM] FDP_ACF.1[MEM] FMT_MSA.3[MEM] FMT_MSA.1[MEM] FMT_MSA.1[SFR] FMT_SMF.1	RE.Phase-1 "Design and Implementation of the Smartcard Embedded Software" (e.g. definition of separated memory windows and sufficiently graded exception handling)
O.SFR_ACCESS	FDP_ACC.1[SFR] FDP_ACF.1[SFR] FMT_MSA.3[SFR] FMT_MSA.1[SFR] FMT_SMF.1	
O.RANGE_CHK	FRU_VRC.1	RE.Phase-1 "Design and Implementation of the Smartcard Embedded Software" (e.g. configuration of sufficiently limits)
OE.Plat-Appl (clarification)		RE.Phase-1 with RE.Cipher and RE.RNG
OE.Resp-Appl (clarification)		RE.Phase-1 with RE.Cipher
OE.Check-Init		RE.Check-Init

Table 15: Mapping of security objectives and requirements

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. Therefore FCS_COP.1 is suitable to meet O.HW_DES3.

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 initialisation (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 realised 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 Units can be changed by the Smartcard Embedded Software whereas the security attributes for the Special Function Register are implemented in the hardware and cannot be changed. Since the security attributes of the Memory Management Units can only be changed depending on the Mode of Operation, both iterations are needed for O.MEM_ACCESS.

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 System Mode, Meta Mode or User Mode. The Special Function Register used to configure the Code MMU can only be accessed in the System Mode. The Special Function Register required to use e.g. the Triple-DES co-processor, the FameX co-processor, the Random Number Generator and to configure the Data MMU can be accessed in the System Mode and in the Meta 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. In addition specific Special Function Register (USP, R15/AP1 and CSFVAL) used for the range check are accessible in the User Mode.

The security functional requirement "Security attribute based access control (FDP_ACF.1[SFR])" with the related Security Function Policy (SFP) "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 initialisation (FMT_MSA.3[SFR])" requires that the TOE provides default values for the Special Function Register and the Program Status Word that defines the Mode of Operation for the TOE. 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 realised in a way that 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.MEM_ACCESS and O.SFR_ACCESS. Therefore, FMT_SMF.1 is suitable to meet the security objective.

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

The justification related to the security objective "Value Range Check (O.RANGE_CHK)" is as follows:

O.RANGE_CHK requires the TOE to check a predefined lower and upper limit for the value stored in a register. This range check must be provided for dedicated registers and the related notification shall ensure an exception handling related to the specific register is possible. Exactly this is the requirement of FRU_VRC.1. Therefore FRU_VRC.1 is suitable to meet O.RANGE_CHK.

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

The usage of cryptographic algorithms requires to use appropriate keys. Otherwise they do not provide security. RE.Cipher requires that keys must be unique with a very high probability, cryptographically strong etc. If keys are imported into the TOE (usually after TOE Delivery), it must be ensured that quality and confidentiality is maintained.

RE.Cipher addresses the usage of keys generated inside the Smartcard IC as well as keys downloaded into the Smartcard IC. If keys are generated by the Smartcard Embedded Software using the security function F.RNG these random numbers must be tested since F.RNG does not include tests implemented in the hardware. The required test effort depends on the intended usage of the random numbers. The requirements RE.Cipher and RE.RNG for the usage of appropriate cryptographic keys for the cryptographic functions and strong random numbers are suitable to meet OE.Plat-Appl and OE.Resp-Appl.

Nevertheless, the developer of the Smartcard Embedded Software must ensure that the additional functions are used as specified and that the User Data processed by these functions are protected as defined for the application context. Using a multi-application operating system may add additional requirements for the separation of different applications by a memory management scheme based upon security mechanisms of the TOE. These issues are addressed by the requirement RE.Phase-1. The Smartcard Embedded Software must implement additional measures regarding RE.Phase-1 defined in [7] (refer to the third point of the enumeration under

RE.Phase-1 "findings of the TOE evaluation reports relevant for the Smartcard Embedded Software"). These measures are addressed in the Guidance, Delivery and Operation Manual for the P16WX064.

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

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

RE.Check-Init requires at least to check the Fabkey data that is part of the pre-personalisation data to prevent the use of Smartcard ICs that are not correctly tested and pre-personalised by the TOE Manufacturer. The Fabkey comprises secret information that is exchanged between the Card Manufacturer and the TOE Manufacturer. F.COMP supports the storage of the Fabkey data at the end of the test phase in the Test Mode. Only the Smartcard Embedded Software is able to check this data in the Application Mode. Therefore RE.Check-Init is suitable to meet OE.Check-Init.

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

8.2.2 Dependencies of security functional requirements

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

The following discussion demonstrates how the dependencies defined by Part 2 of the Common Criteria for the requirement FCS_COP.1 and (both iterations) are satisfied.

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

Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
FCS_COP.1	FDP_ITC.1 or FCS_CKM.1 FCS_CKM.4 FMT_MSA.2	Yes (by the environment)
FDP_ACC.1[MEM]	FDP_ACF.1	Yes, by FDP_ACF.1[MEM]
FDP_ACC.1[SFR]	FDP_ACF.1	Yes, by FDP_ACF.1[SFR]
FDP_ACF.1[MEM]	FDP_ACC.1 FMT_MSA.3	Yes, by FDP_ACC.1[MEM] Yes
FDP_ACF.1[SFR]	FDP_ACC.1 FMT_MSA.3	Yes, by FDP_ACC.1[SFR] Yes
FMT_MSA.3[MEM]	FMT_MSA.1 FMT_SMR.1	Yes, by FMT_MSA.1[MEM] See discussion below
FMT_MSA.3[SFR]	FMT_MSA.1 FMT_SMR.1	Yes, by FMT_MSA.1[SFR] See discussion below
FMT_MSA.1[MEM]	FDP_ACC.1 or FDP_IFC.1 FMT_SMR.1 FMT_SMF.1	Yes, by FDP_ACC.1[MEM] See discussion below Yes
FMT_MSA.1[SFR]	FDP_ACC.1 or FDP_IFC.1 FMT_SMR.1 FMT_SMF.1	Yes, by FDP_ACC.1[SFR] See discussion below Yes
FRU_VRC.1	-	-

Table 16: Dependencies of security functional requirements

The developer of the Smartcard Embedded Software must ensure that the additional security functional requirement (FCS_COP.1) is used as specified and that the User Data processed by the related security function is protected as defined for the application context. These issues are addressed by the requirement RE.Phase-1.

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

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

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

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

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

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

As stated in the Protection Profile, section 7.2.3, it has to be assumed that attackers with high attack potential try to attack smart cards used for digital signature applications or payment systems. Therefore specifically AVA_VLA.4 was chosen by the PP in order to assure that even these attackers cannot successfully attack the TOE. For the same reason the Strength of Function level “high” is required.

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

8.2.4 Security Requirements are Mutually Supportive and Internally Consistent

The discussion of security functional requirements and assurance components in the preceding sections has shown that mutual support and consistency are given for both groups of requirements. The arguments given for the fact that the assurance components are adequate for the functionality of the TOE also show that the security functional requirements 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 area based memory access control function as well as the access control to Special Function Register and the value range check implemented according to the security functional requirement FCS_COP.1 and FDP_ACC.1[MEM], FDP_ACC.1[SFR] with reference to the Access Control Policies defined in FDP_ACF.1[MEM], FDP_ACF.1[SFR] as well as FRU_VRC.1. Therefore, these security functional requirements support the secure implementation and operation of FCS_COP.1 and of FDP_ACC.1 with FDP_ACF.1 as well as the dependent security functional requirements.

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

8.3 TOE Summary Specification Rationale

8.3.1 Rationale for TOE security functions

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

F.RNG

The security function F.RNG provides an 8 bit random number. The random numbers provided by the random number generator are strong since the environmental conditions of the random number generator are controlled by the sensors. The behaviour of the random number generator is independent of the Smartcard Embedded Software. The entropy of the random numbers as claimed by the security functional requirement are ensured by the requirements of the AIS31. Therefore F.RNG obviously meets FCS_RND.1. Note that tests are required by the Smartcard Embedded Software (refer to [11]).

F.HW_DEA

F.HW_DEA is realised by a co-processor approach. F.HW_DEA applies the encryption/decryption function to 8 bytes data. F.HW_DEA provides a 16 byte key register for a fast 2-key Triple-DEA calcualtion. The Triple-DES is performed with a minimum control by the Smartcard Embedded Software. The control of the Triple-DEA within the encryption/decryption function is provided by an own sequencer of the co-processor. Therefore F.HW_DEA is suitable to meet FCS_COP.1.

F.OPC

The function F.OPC ensures the correct operation of the TOE within the limits specified in [9]. Therefore the TOE comprises filters for power supply and clock input. In addition F.OPC controls the allowed range of temperature, clock frequency and voltage.

The filters support the correct function of the TOE within the limits of the operating conditions. This robustness implements FRU_FLT.2 and ensures that the processing is performed without failure that may be caused by interference of the ISO-interface or other external influences. Therefore the proper operation of the Random Number Generator, the Triple-DES co-processor and the arithmetic co-processor (FameX) that may be used for cryptographic operations can be ensured within the specified limits. This also holds for the CPU and all other specialised components.

FPT_FLS.1 is implemented by sensors for the upper and lower threshold of the operating conditions temperature, clock frequency as well as voltage. The sensors detect whether one parameter is outside the specified range. The secure state required by FPT_FLS.1 is realised by an internal reset of the Smartcard IC. This secure state is applied as long as one sensor identifies a corrupt condition.

The high voltage sensor that controls the voltage for the write process to the EEPROM can be checked by the Smartcard Embedded Software as part of the verification procedure for a write sequence to the EEPROM.

Considering Application Note 20 of [7], an internal reset of the Smartcard IC is sufficient because all internal operations are stopped and the relevant special function registers are set to defined reset values.

The Smartcard Embedded Software cannot disable the sensors. In addition the filters and sensors together with the reset block are implemented mostly independent of the other hardware components. This means that the TSF maintain a security domain for its own execution that protects it from interference and tampering especially by (potentially) untrusted parts of the Smartcard Embedded Software. This meets the SFR FPT_SEP.1.

Note: The TOE fulfils the security functional component FPT_SEP.1 for F.OPC only in the Application Mode because FPT_SEP.1 is supported by the security function F.COMP.

F.PHY

F.PHY prevents successful physical tampering of the TOE. This is realised using various special features in the design and layout of the circuitry. This features mainly shield and hide the relevant design and include (i) security routing that adds unused lines between active ones to fill the topmost layers, (ii) route thick supply lines over interface areas and (iii) cover memory blocks and sensitive analogue parts with meshes and tiles.

There is no common bus, only local dedicated data and address lines interconnect the different memories and the CPU. All interfaces, including the data and address scrambling logic for the

memories are also part of the 'Glue Logic'. Therefore it is never possible to observe any clear data by tapping local memory busses.

Beside the measures mentioned above the General CPU Functions, the Specialised Components, the memory management untis with all memory interfaces including the scrambling functions are realised in a so called glue logic. The 'Glue Logic' is a sea of basic low level gates. These gates are placed and interconnected by using automated tools which provide random, heuristic and deterministic placement and routing procedures. The CPU bus is never leaving the 'Glue Logic' area. The five metal layer technology allows routing on top of the cells. By this on one hand no routing channels can be found (and therefore also no bus structures can be found) and on the other hand even the cell structure itself is no longer visible.

This security function meets FPT_PHP.3.

Note: This security function also supports all other SFRs because prevention of successful manipulation of security functionality is a pre-condition for the reliable work of all other functions.

F.LOG

F.LOG prevents the reconstruction of TOE internal information that can be found by analysis of external measured signals like power, clock, or I/O lines. Within the different components of the TOE dedicated functions are implemented to sufficiently limit or eliminate the information that might be contained in the shape and amplitude of signals or in the time between events.

The countermeasures implemented in the Triple-DES co-processor are independent of the keys and plain- or ciphertext calculated by coprocessor. In addition the countermeasures cannot be influenced by the Smartcard Embedded Software. The same calculation time for the encryption and decryption function with all operands is also ensured by the design of the co-processor.

The measures to prevent timing attacks on basic modular function are based on the design of the FameX co-processor. The calculation time of one modular operation depends on the lengths of the operands but not on the value of the operands. Additional design measures limit the dependency between the internal operated data of the FameX co-processor and the shape and amplitude of the power consumption of the chip. Since the FameX does not realise an algorithm on its own specific countermeasures against leakage must be implemented related to the realised cryptographic algorithm.

The secure DCDC-converter is a block within the internal power supply circuitry of the Smartcard IC that can be switched to a mode that enables a constant current consumption at the power supply pads as interface of the Smartcard IC. This can be configured by the Smartcard Embedded Software. Several clock configurations allow the usage of internally generated clock signals for different components (e.g. the co-processors) on the Smartcard IC to operate independent of the external clock. In addition the execution of instructions can be randomised to some extent to prevent the possibility to synchronise the internal behaviour based on external signals (clock and power consumption) for leakage attacks.

Other features like filtering and scrambling that are implemented to increase the robustness and confidentiality also contribute to counter leakage attacks. The behaviour of F.LOG is supported by different features realised in the functions F.OPC and F.PHY.

The features implement the SFRs FDP_ITT.1, FPT_ITT.1 and FDP_IFC.1. They are mostly independent of the Smartcard Embedded Software and cannot be influenced from outside the TOE.

F.COMP

F.COMP realises the control between the Test Mode and the Application Mode of the TOE. Within these two TOE modes the System Mode, the Meta Mode and the User Mode provide the same behaviour.

F.COMP provides access to the IC Dedicated Test Software in the Test Mode or to the Smartcard Embedded Software in the Application Mode by evaluating the related fuses during the boot sequence. It assures that it is not possible to switch back to the Test Mode once the Application Mode is activated. Moreover it prevents users in Application Mode from using test functions by denying access to the Special Function Register for test or deactivating these Special Function Register thereby read or write operations have no effect. In addition F.COMP restricts the access for configuration of security features to the Test Mode. This is supported by F.SFR_ACC since Special Function Registers for test purpose are not accessible in the Application Mode.

There is a test concept with specific hardware operations and a set of test functions. The results of the specific hardware operations do not provide information on stored data or internal functionality. The test functions are designed in a way that they can not be used to read out directly any data stored in one of the memories of the TOE. Therefore the capabilities to abuse the test functions is very limited as required by FMT_LIM.1.

FMT_LIM.2 is realised by the separation of TOE modes via the mode switch. Once the TOE mode is set to Application Mode F.COMP ensures that it is not possible to switch back to Test Mode to reuse the test functions. In addition the functions of the IC Dedicated Test Software require a special sequence to execute a dedicated test routine. Therefore F.COMP limits the availability of the test functions as stated by FMT_LIM.2.

The OTP area is erased during the Test Mode and thereby set to a defined status. In the Application Mode the erase protection of the OTP area is controlled in the same way as the write and erase protection of the security area for the configuration and fuses.

FAU_SAS.1 is implemented by a test function that allows to store identification and/or pre-personalisation data for the TOE in the EEPROM at the end of the tests in phase 3. Also this function is only available after a special authentication sequence.

Since the IC Dedicated Test Software (in Test Mode) can perform test functions for configuration and disable the sensors temporarily for test reasons, while the Smartcard Embedded Software (in Application Mode) is not able to do this, the TOE supports the separation between

the security domains in the Application Mode and therefore enables FPT_SEP.1 of the security function F.OPC.

F.MEM_ACC

The function F.MEM_ACC is designed in a way that dedicated software routines of a smartcard operating system (Smartcard Embedded Software) can be used to configure the memory management units for code and data. All Code Memory Area Attributes and Data Memory Area Attributes are stored in Special Function Register. The software routines must at least run in System Mode to configure the Code MMU and in System Mode or Meta Mode to configure the Data MMU.

The access control for the Code Memory Areas can be configured with the granularity of (i) read, (ii) write, (iii) execute and (iv) enable/disable. The access control for the Data Memory Areas can be configured with the granularity of (i) read, (ii) write and (iii) enable/disable. The memory management provides the possibility to define different independent windows that provide access to a memory area. Every window has its own set of Code Memory Area Attributes or Data Memory Area Attributes that define the memory location and limitation as well as the access rights. This can be used in a way that effectively protects the data of one application from the access by another application.

Access violations are notified using a specific exception. Exceptions must be handled by a sufficient software routing in System Mode.

The function F.MEM_ACC realises the SFRs FDP_ACC.1[MEM] and FDP_ACF.1[MEM].

The static attribute initialisation can be seen as the reset values that are provided by the hardware. These reset values cannot be changed. Therefore FMT_MSA.3[MEM] is realised by the definition of the reset values for the dedicated registers. Nevertheless the static attribute initialisation for the memory management units can be changed by the Smartcard Embedded Software that may be called based on a reset exception. A respective routine is able to set the initial attributes after the reset.

The management of security attributes is only possible for dedicated software routines (Smartcard Embedded Software) that are running in System Mode or Meta Mode as described in FMT_MSA.1[MEM]. The management functions required to change the security attributes are implemented as specified by FMT_SMF.1. This covers the access control configuration for the Code MMU by Smartcard Embedded Software running in the System Mode and for the Data MMU by Smartcard Embedded Software running in the System Mode or in the Meta Mode.

Note that the Smartcard IC Platform provides access control based on the definition of and operation on memory areas. The access control based on file types and related access flags must be implemented and enforced by the Smartcard Embedded Software.

F.SFR_ACC

The function F.SFR_ACC realises the access control to the Special Function Registers and the switch between the System Mode, the Meta Mode and the User Mode. The actual mode of operation is based on the bits 6 and 7 of the Program Status Word. Access to the Special Function Register is granted or not depending on the Mode of Operation (System Mode, Meta Mode, User Mode).

The System Mode and the Meta Mode are mainly the same. They provide access to all Special Function Registers including the Data MMU, the cryptographic co-processors, I/O interfaces and the configuration of the hardware. Exceptions are the Special Function Registers of the Code MMU and the System Configuration Register that are only writable in the System Mode. The configuration of the Code MMU is readable in the Meta Mode but cannot be changed in this mode of operation.

In the User Mode only the Special Function Registers related to General CPU Functions are accessible and specific Special Function Registers that can be used for the value range check.

The only effective possibility to switch between different Modes of Operation is provided by exceptions or interrupts that allow different applications to use common parts of the software and vice versa allows the (operating system) software to control access to the resources of the TOE. Every specific exception leads to a dedicated address within the vector table (refer to the memory address map). The vector table provides a value for the Program Status Word to set the Mode of Operation for the routine and a pointer to the dedicated routine of the Smartcard Embedded Software that is developed to handle this exception. The developer of the Smartcard Embedded Software must design these exception and interrupt routines carefully. The implementation of F.SFR_ACC realises the SFRs FDP_ACC.1[SFR] and FDP_ACF.1[SFR] and thereby provides the separation of resources as required for a multi application platform.

The static attribute initialisation FMT_MSA.3[SFR] is given by the implementation of the hardware part used to set up an exception. When a triggered exception is fetched the hardware sets the System Mode to get control on the hardware and disables the Code and Data MMU to ensure the correct access to the vector table. Based on the exception the associated value for the Program Status Register and the associated address for the exception handling routine are loaded from the vektor table. Since the static attribute initialisation is realised in hardware it cannot be changed. The change of the Mode of Operation is performed by invoking exceptions or interrupts. The management of the security attributes is completely implemented in the hardware because the relevant bits of the Program Status word (bit 6 adn bit 7) can only be loaded from the vector table based on the associated exception as required in FMT_MSA.1[SFR]. The management functions related to F.SFR_ACC as specified by FMT_SMF.1 are implemented by requiring an exceptions or interrupts – which is the only way to change the Mode of Operation – and therefore the execution of dedicated code.

F.RANGE_CHK

The security function F.RANGE_CHK provides range checking for certain registers (CPU internal and Special Function Register). The range checking comprises checking against a lower and an upper bound. In summary F.RANGE_CHK obviously meets FRU_VRC.1.

8.3.2 Rationale for assurance measures

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

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

8.4 PP Claims Rationale

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

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

The evaluation assurance level claimed for this target (EAL5+) is shown in section 5.1.1.3 to include resp. exceed the requirements claimed by the PP (EAL4+).

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

9 Annexes

9.1 Definition of the family FRU_VRC

The CC class FRU provides families that support the availability of required resources such as processing capability and/or storage capacity. The existent families cover cases in which resources are used directly, e.g. the family Resource Allocation provides limits on the use of available resources that the TOE provides.

At a more general level than direct resource access, limiting the availability may be realised by counters or limits for certain values, e.g. maintaining a counter and forcing an automatic event when a certain limit is reached. Furthermore, automatic checking of bounds of e.g. a stack pointer, improves fault-tolerance.

The family FRU_VRC is added to the class FRU to provide security functional requirements for this sort of fault-tolerance.

Family behaviour

The requirements of this family ensure that the TOE will check that values of certain parameters are within specified bounds. An automatic event (notification) is generated in a case of violation.

Component levelling

FRU_VRC Value range checking	1

FRU_VRC.1 Simple value range checking requires the TOE to check the allowed range for certain resources and to notify an authorised role in case of a violation.

Management: FRU_VRC.1

There are no management activities foreseen.

Audit: FRU_VRC.1

There are no actions identified that should be auditable if FAU_GEN Security audit data generation is included in the PP/ST.

FRU_VRC.1 Simple value range checking

Hierarchical to: No other components

FRU_VRC.1.1 The TSF shall enforce a range checking for the value of the following resources: [assignment: *controlled resources*].

FRU_VRC.1.2 The TSF shall notify [assignment: *the authorised identified roles*] that the value is out of the defined range.

Dependencies: No dependencies.

9.2 Further Information contained in the PP

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

9.3 Glossary and Vocabulary

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

Administrator	(in the sense of the Common Criteria) The TOE may provide security functions which can or need to be administrated (i) by the Smartcard Embedded Software or (ii) using services of the TOE after delivery to Phases 4-6. Then a privileged user (in the sense of the Common Criteria, refer to definition below) becomes an administrator.
Application Mode	Configuration of the TOE for operational use after disabling the Test Mode. The executed instructions are fetched from User-ROM or EEPROM only.
Card Manufacturer	The customer of the TOE Manufacturer who receives the TOE during TOE Delivery. The Card Manufacturer includes all roles after TOE Delivery up to Phase 7 (refer to [7], Figure 4 on page 17 and Section 8.1.1). The Card Manufacturer has the following roles (i) the Smartcard Product Manufacturer (Phase 5) and (ii) the Personaliser (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.
Code Memory Areas	Address spaces provided by the Code Memory Management Unit based on the configuration of the Code Memory Area Attributes. The Code Memory Areas can be

	used for persistent stored data which may be used as data and executable code. They are located in ROM and EEPROM.
Code Memory Area Attributes	Configuration of the Special Function Registers of the Code Memory Management Unit CW*x*BOT (the virtual code memory start address Code Memory Area *x*), CW*x*TOP (the virtual code memory base address of the last valid block of the Code Memory Area *x*) and CWACRL*yz* (access rights read, write execute and enabled/disabled in User Mode), where *x*=0, 1, 2,...,11 as indexes of the Code Memory Areas, *y*=0,1,2 and *z*=L,H mapped to the indexes of the Code Memory Areas, (see [9] for details). In addition the CW*x*OFS defines the offset in the physical address space.
Code Memory Management Unit	The Code MMU maps the virtual addresses used by the CPU into the physical addresses of the ROM and EEPROM. The mapping is provided by the Code Memory Area Attributes that allow to define 12 Code Memory Areas (windows). Each of them is individually (i) positioned and sized (ii) enabled or disabled in User Mode and (iii) controlled by access permissions for read, write and execute. The Code Memory Area Attributes are writable in System Mode. The unit is active in Meta Mode and User Mode.
Data Memory Areas	Windows with an address space provided by the Data Memory Management Unit based on the configuration of the Data Memory Area Attributes. The Data Memory Areas can be used for storing volatile data but can not be executed as instruction. They are located in the RAM.
Data Memory Area Attributes	Configuration of the windows of the Code Memory Management Unit DW*x*BOT (the virtual data memory start address Data Memory Area *x*), DW*x*TOP (the virtual data memory base address of the last valid block of the Data Memory Area *x*) and DWACRL*z* (access rights write, read and enabled/disabled in User Mode), where x stands for A, B, C and D as index of the Data Memory Area, *z*=H,L (see [9] for details). In addition the DW*x*OFS defines the offset in the physical address space.
Data Memory Management Unit	The Data MMU maps the virtual addresses used by the CPU into the physical addresses of the RAM. The mapping is provided by the Data Memory Area Attributes

	that allow to define 4 Data Memory Areas (windows) each of them is individually (i) positioned and sized, (ii) visible or invisible in User Mode, and (iii) controlled by permissions for write access. The Data Memory Area Attributes are accessible in System Mode and in Meta Mode. The unit is active in Meta Mode and User Mode.
Exceptions interrupts	Non-maskable interrupt of program execution starting from fixed (depending on exception source) addressees and enabling the System Mode. The source of exceptions are: reset, stack overflow, divide-by-zero, User mode execution of RETI instruction and hardware breakpoints.
General CPU Functions	All registers that can be used by a software running in the User Mode are summarised as general CPU Functions. This includes the extended register file with banked and global registers as well as the APMIN and APMAX register. In addition the segment select register (SSEL), the code segment register (CS1, CS2, CS3), the CSFVAL register with the associated CSFVALMIN and CSFVALMAX register, the low byte of the program status word and the User Stack Register are included.
Integrated Circuit (IC)	Electronic component(s) designed to perform processing and/or memory functions.
IC Dedicated Software	IC proprietary software embedded in a smartcard IC (also known as IC firmware) and developed by the IC Developer. Such software is required for testing purpose (IC Dedicated Test Software) but may provide additional services to facilitate usage of the hardware and/or to provide additional services (IC Dedicated Support Software).
IC Dedicated Support Software	Part of the IC Dedicated Software (refer to above) which provides functions after TOE Delivery. The usage of parts of the IC Dedicated Software might be restricted to certain phases.
IC Dedicated Test Software	Part of the IC Dedicated Software (refer to above) which is used to test the TOE before TOE Delivery but which does not provide any functionality thereafter.
Initialisation Data	Any data defined by the TOE Manufacturer and injected into the non-volatile memory by the Integrated Circuits manufacturer (Phase 3). These data are for instance used

	for traceability and for TOE identification (identification data).
Mode of Operation	Configuration of the Special Function Register "Program Status Word" defining the behaviour of the CPU registers, the Special Function Registers and the Memory Management Units to control the use of the User-ROM, the EEPROM, the RAM and the TOE resources in the Application Mode. The Modes of Operation are the System Mode, the Meta Mode and the User Mode.
Memory	The memory comprises of the User-ROM, the EEPROM and the RAM of the TOE.
Meta Mode	The Meta Mode of operation has unlimited access to the hardware resources except the Special Function Registers of the Code Memory Management Unit and the System Configuration Register.
Pre-personalisation Data	Any data supplied by the Card Manufacturer that is injected into the non-volatile memory by the Integrated Circuits manufacturer (Phase 3). These data are for instance used for traceability and/or to secure shipment between phases.
Smartcard	(as used in the Protection Profile [7]) Composition of the TOE, the Smartcard Embedded Software, User Data and the package (the smartcard carrier).
Smartcard Embedded Software	Software embedded in a smartcard IC and not being developed by the IC Designer. The Smartcard Embedded Software is designed in Phase 1 and embedded into the Smartcard IC in Phase 3 or in later phases of the smartcard product life-cycle. Some part of that software may actually implement a smartcard application others may provide standard services. Nevertheless, this distinction doesn't matter here so that the Smartcard Embedded Software can be considered as being application dependent whereas the IC Dedicated Software is definitely not.
Special Function Registers	Registers used to access and configure the functions for the communication with an external interface device, the cryptographic co-processor for Triple-DES, the FameX co-processor for basic arithmetic functions to perform

	asymmetric cryptographic algorithms, the random numbers generator and chip configuration.
Specialised Components	Specialised Components are blocks of the hardware that provide Special Function Registers as an interface. The access to the Special Function Register is mandatory for the usage of the specific functions. This comprises the Triple-DES co-processor, the FameX co-processor, the CRC calculation unit, the Random Number Generator, the I/O lines as well as the USB interface, the Timers and specific configurations (e.g. clock settings, power configuration).
Special Pointer Register	The Special Pointer Register (R15 and CSFVAL) are specific Special Function Register that comprise a permanent overflow and underflow supervision. In addition also the System Stack and the User Stack include this kind of protection. If an overflow or an underflow is detected an exception is triggered.
System Mode	The System Mode has unlimited access to the hardware resources. The Memory Management Units of code and of data are switched off, which means that the virtual address of the CPU and the physical memory address are mapped one to one.
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	Configuration of the TOE enabling the IC Dedicated Test Software. The Test Mode is permanently and irreversible disabled after production testing. In the Test Mode specific Special Function Registers are accessible in System Mode or Meta Mode for test purpose.
TOE Delivery	The period when the TOE is delivered which is (refer to [7], Figure 4 on page 17) either (i) after Phase 3 (or before Phase 4) if the TOE is delivered in form of wafers or sawn wafers (dice) or (ii) after Phase 4 (or before Phase 5) if the TOE is delivered in form of modules.
TOE Manufacturer	The TOE Manufacturer must ensure that all requirements for the TOE and its development and production environment are fulfilled (refer to [7], Figure 4 on page 17).

	The TOE Manufacturer has the following roles: (i) IC Developer (Phase 2) and (ii) IC Manufacturer (Phase 3). If the TOE is delivered after Phase 4 in form of modules, he has the role of the (iii) IC Packaging Manufacturer (Phase 4) in addition.
TOE Mode	The Test Mode and the Application Mode can be seen as main modes of the TOE that are assigned to the life cycle. In both TOE modes the System Mode, the Meta Mode and the User Mode are available with the mode specific behaviour.
Trap interrupt	Non-maskable interrupt of program execution processed as part of the execution of the TRAP instruction. The TOE supports 16 different trap interrupt vectors as start address of a trap handling routine.
TSF data	Data created by and for the TOE, that might affect the operation of the TOE (for example configuration data). Note that the TOE is the Smartcard IC. Initialisation Data defined by the Integrated Circuits manufacturer to identify the TOE and to keep track of the product's production and further life-cycle phases are also considered as belonging to the TSF data.
User	(in the sense of the Common Criteria) The TOE serves as a platform for the Smartcard Embedded Software. Therefore, the "user" of the TOE (as used in the Common Criteria assurance class AGD: guidance) is the Smartcard Embedded Software. Guidance is given for the Smartcard Embedded Software Developer. On the other hand the Smartcard (with the TOE as a major element) is used in a terminal where communication is performed through the ISO interface provided by the TOE. Therefore, another "user" of the TOE is the terminal (with its software).
User Data	All data managed by the Smartcard Embedded Software in the application context. User data comprise all data in the final Smartcard IC except the TSF data.
User Mode	The User Mode of operation has access to the ROM and EEPROM under control of the Memory Management Unit for code and RAM under control of the Memory Management Unit for data. It uses the User stack. The

	access to the Special Function Registers is very limited. The TOE leaves the User Mode by TRAP instruction to execute instructions selected by the trap interrupt vector.
User-ROM	Read-only memory (ROM) enabled after disabling the Test Mode.

9.4 List of Abbreviations

CC	Common Criteria Version 2.0 or Version 2.1. Note that the Version 2.1 (ISO 15408) is technically identical with Version 2.0 of the Common Criteria.
DEA	Data Encryption Algorithm.
DES	Data Encryption Standard.
DRNG	Deterministic Random Number Generator
EAL	Evaluation Assurance Level.
IC	Integrated circuit.
IT	Information Technology.
MMU	Memory Management Unit
NDA	Non Disclosure Agreement.
PP	Protection Profile.
SAR	Security Assurance Requirement.
SFR	as abbreviation of the CC term: Security Functional Requirement, as abbreviation of the technical term of the SmartXA-family: Special Function Register[38]
SF	Security function.
SIM	Subscriber Identity Module.
SOF	Strength of function.
ST	Security Target.
TOE	Target of Evaluation.
TSC	TSF Scope of control.
TSF	TOE Security functions.
TSFI	TSF Interface.
TSP	TOE Security Policy.
UART	Universal Asynchronous Receiver and Transmitter.

[38] This security target does not use SFR as abbreviation of Special Function Register to avoid confusion.

USB Universal Serial Bus

9.5 Bibliography

9.5.1 Evaluation Documents

[1] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; Version 2.1, August 1999

[2] Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements; Version 2.1, August 1999

[3] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; Version 2.1, August 1999

[4] Common Methodology for Information Technology Security Evaluation CEM-99/045 Part 2: Evaluation Methodology, Version 1.0, August 1999

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

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

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

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

9.5.2 Developer Documents

[9] Data Sheet, P16WX064 SmartXA-Family, Secure 16-bit Smart Card Controller, Product Specification, Philips Semiconductors, Revision 3.1, Document Number: 053531, November 29th , 2002

[10] Instruction Set P16WX064 SmartXA-Family, Secure 16-bit Smart Card Controller, Product Specification, Philips Semiconductors, Revision 3.0, Document Number: 074630, July 5th, 2002

[11] Guidance, Delivery and Operation Manual for the P16WX064

9.5.3 Other Documents

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

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

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

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