Common Criteria Security Target Lite

IFX_CCI_000007h, IFX_CCI_000009h, IFX_CCI_00000Ah, IFX_CCI_00000Bh, IFX_CCI_000016h, IFX_CCI_000017h, IFX_CCI_000018h, IFX_CCI_000023h, IFX_CCI_000024h

Design step G13

Including optional Software Libraries
Flash Loader –ACL – 4x HSL–CCL

Multi-assurance with global assurance level EAL5 augmented with ADV_IMP.2, ADV_INT.3, ADV_TDS.5, ALC_CMC.5, ALC_DVS.2, ALC_FLR.3, ALC_TAT.3, ATE_FUN.2, ATE_COV.3, AVA_VAN.5 and sub-TSF assurance level EAL6 augmented with ALC_FLR.3

Table of Contents

1 Security Target Introduction (ASE_INT) 5
1.1 Security Target and Target of Evaluation Reference 5
1.2 Target of Evaluation overview 14
2 Target of Evaluation Description 19
2.1 TOE Definition 19
2.2 Scope of the TOE 23
2.2.1 Hardware of the TOE 24
2.2.2 Firmware and software of the TOE 25
2.2.2.1 Firmware 28
2.2.2.2 Optional Software 28
2.2.3 Interfaces of the TOE 28
2.2.4 Guidance documentation 30
2.2.5 Forms of delivery 30
2.2.6 Production sites 32
3 Conformance Claims (ASE_CCL) 33
3.1 PP Claim 33
3.2 Package Claim 33
3.3 Conformance Rationale 33
3.4 Adding Objectives 33
3.5 AES and TDES 34
3.6 Loader 34
3.7 CIPURSE Cryptographic Library (CCL) 35
3.8 Summary 35
3.9 Application Notes 35
4 Security Problem Definition (ASE_SPD) 37
4.1 Threats 37
4.1.1 Additional Threat due to TOE specific Functionality 37
4.1.2 Assets regarding the Threats 38
4.2 Organizational Security Policies 39
4.2.1 Augmented Organizational Security Policy 40
4.3 Assumptions 41
4.3.1 Augmented Assumptions 41
4.3.2 Note regarding CIPURSE™ CL 42
5 Security objectives (ASE_OBJ) 43
The user has the following standard high-level security goals related to the assets: 43
5.1 Security objectives for the TOE 43
5.2 Security Objectives from PP for development and environment 46
5.3 Security Objectives for the Environment 47
5.3.1 Clarification of “Treatment of User Data (OE.Resp-Appl)” 48
5.3.2 Clarification of “Protection during Composite product Manufacturing (OE.Process-Sec-IC)” 48
5.4 Security Objectives Rationale 49
5.5 P.Add-Functions 50
5.6 A.Key-Function 50
5.7 T.Mem-Access 51
5.8 P.Ctrl_Loader, P.Lim_Block_Loader, T.Masquerade_TOE and T.Open_Samples_Diffusion 51
5.9 P.Crypto-Service 53
6 Extended Component Definition (ASE_ECD) 54
6.1 Component Subset TOE security testing (FPT_TST.2) 54
6.1.1 Definition of FPT_TST.2 54
6.1.2 TSF self-test (FPT_TST) 55

7 Security Requirements (ASE_REQ) 56
7.1 TOE Security Functional Requirements 56
7.1.1 Extended Component FAU_SAS.1 56
7.1.1.1 FAU_SAS 56
7.1.2 Memory access control 56
7.1.3 Support of Cipher Schemes 60
7.1.3.1 Preface regarding Security Level related to Cryptography 61
7.1.3.2 Random Number Generation 61
7.1.3.2.1 True Random Number Generation, meeting AIS31 PTG.2 61
7.1.3.2.2 Hybrid Random Number Generation, meeting AIS31 PTG.3 62
7.1.3.2.3 Deterministic Random Number Generation (DRNG) AIS31 DRG.3 63
7.1.3.2.4 Deterministic Random Number Generation (DRNG) AIS31 DRG.2 64
7.1.3.3 Cryptography by the Symmetric Cryptographic Coprocessor SCP 65
7.1.3.3.1 Triple-DES Operation 65
7.1.3.3.2 AES Operation 67
7.1.3.4 Cryptography by CIPURSE™ Cryptographic Library 69
7.1.3.4.1 CIPURSE™ CL Cryptographic Functions 69
7.1.3.5 The Asymmetric Cryptographic Library v2.09.002 71
7.1.3.5.1 Rivest-Shamir-Adleman (RSA) operation 71
7.1.3.5.2 Rivest-Shamir-Adleman (RSA) key generation 72
7.1.3.5.3 General Preface regarding Elliptic Curve Cryptography 73
7.1.3.5.4 Elliptic Curve (EC) key generation 76
7.1.3.5.5 Elliptic Curve Diffie-Hellman (ECDH) key agreement 77
7.1.4 Data Integrity FDP_SDI.2 and FDP_SDC.1 78
7.1.5 Limited Capabilities and Limited Availability 78
7.1.6 Application Note to FPT_FLS.1 Failure with preservation of secure state 80
7.2 Support by the Flash Loader 80
7.3 TOE Security Assurance Requirements 85
7.3.1 Refinements 86
7.3.1.1 Development (ADV) 86
7.3.1.2 Life-cycle Support (ALC) 88
7.3.1.3 Tests (ATE) 89
7.3.1.4 AVA_VAN Vulnerability Analysis 89
7.4 Security Requirements Rationale 90
7.4.1 Rationale for the Security Functional Requirements 90
7.4.1.1 Cryptographic Aspects 92
7.4.1.2 Hardware related Aspects 93
7.4.1.3 Flash Loader related Aspects 94
7.4.1.4 Dependencies of Security Functional Requirements 95
7.4.2 Rationale of the Assurance Requirements 98
8 TOE Summary Specification (ASE_TSS) 99
8.1 SF_DPM: Device Phase Management 99
8.1.1 Listing of SFRs implemented by SF_DPM Device Phase Management 100
8.2 SF_PS: Protection against Snooping 101
8.2.1 Listing of SFRs implemented by SF_PS Protection against Snooping 103
8.3 SF_PMA: Protection against Modifying Attacks 103
8.3.1 Listing of SFRs implemented by SF_PMA Protection against Modifying Attacks 105
8.4 SF_PLA: Protection against Logical Attacks 105
8.4.1 Listing of SFRs implemented by SF_PLA Protection against Logical Attacks 106
8.5 SF_CS: Cryptographic Support 106
8.5.1 Implementation of AES and TDES by the Symmetric Cryptographic Coprocessor SCP 107
8.5.1.1 Triple DES 107

8.5.1.2 AES ... 107
8.5.2 RSA Cryptographic Library ... 109
8.5.2.1 Encryption, Decryption, Signature Generation and Verification ... 109
8.5.2.2 Asymmetric Key Generation ... 110
8.5.3 Elliptic Curves Cryptographic Library ... 110
8.5.3.1 Asymmetric Key Generation ... 110
8.5.3.1 ECC ADD point addition ... 110
8.5.4 Toolbox Library ... 111
8.5.5 CIPURSE™ Cryptographic Library ... 111
8.5.6 Hybrid PTRNG ... 111
8.5.7 Listing of SFRs implemented by SF_CS "Cryptographic Support" ... 112
8.6 Assignment of Security Functional Requirements to TOE's Security Functionality ... 112
8.7 Security Requirements are internally Consistent ... 114
9 Literature and References ... 116
10 Hash Signatures of Cryptographic Libraries ... 120
10.1 ACL–0: RSA, EC, Toolbox Version v2.09.002 ... 120
10.2 HW- Support Libraries HSL in four versions ... 120
10.3 CIPURSE™ Cryptographic Library (CCL) in version v02.00.0004 ... 121
11 List of Abbreviations ... 122
12 Glossary ... 124
13 Revision History ... 126

1 Security Target Introduction (ASE_INT)

1.1 Security Target and Target of Evaluation Reference

The title of this document is Security Target Lite as given on the front page and it covers and describes one hardware platform respectively one IC representing the Target of Evaluation (TOE) with following Common Criteria Identifiers:

- IFX_CCI_000007h
- IFX_CCI_000009h
- IFX_CCI_00000Ah
- IFX_CCI_00000Bh
- IFX_CCI_000016h
- IFX_CCI_000017h
- IFX_CCI_000018h
- IFX_CCI_000023h
- IFX_CCI_000024h

in the design step G13 and including optional software libraries and dedicated firmware in several versions as stated and assigned in the table below.

In order to ease the readability of this document the bunch of Common Criteria Identifiers as listed above is shortened and simply expressed with TOE (Target of Evaluation) or IFX_CCI_000007h with options.

This document is formed according to Common Criteria CC:2022 EAL5 and EAL6 and comprises the Infineon Technologies AG Security Controller (Integrated Circuit IC) with the above listed Common Criteria Identifiers and with specific IC dedicated firmware in alternatives and optional software.

The target of evaluation (TOE) is described in the following.
This Security Target Lite has the revision 3.9 and is dated 2026-09-02.

The Target of Evaluation (TOE) is the Infineon Security Controller with following optional available software packages:

- The asymmetric cryptographic library (ACL) v2.09.002 provides
  - RSA[1] 2048/4096 cryptography.
    The library supports also smaller key lengths, but the certification follows the national regulation by the BSI.
  - elliptic curve cryptography EC[2]
- the Toolbox library in the version v2.09.002 provides basic mathematical functions for a simplified user interface to the Crypto@2304T. The Toolbox library is not part of the TOE Security Functionality (TSF).
- the hardware support library (HSL) in the versions v01.22.4346 and v02.01.6634 provides a simplified interface and utilizes the full services of the SOLID FLASH™ NVM to the user

[1] Rivest-Shamir-Adleman asymmetric cryptographic algorithm

[2] The Elliptic Curve Cryptography is abbreviated with EC only in the further, in order to avoid conflicts with the abbreviation for the Error Correction Code ECC.

- the hardware support library (HSL) in the versions v03.11.8339 and v03.12.8812 provide equal to the above a simplified interface, utilizes the full services of the SOLID FLASH™ NVM to the user, and enables the use of an additional advanced reading respectively writing method
- the CIPURSE™ Cryptographic Library (CIPURSE™ CL and CCL) in the version v02.00.0004 provides the user OSPT[1] alliance CIPURSE™ V2 conformant communication functionality between a PICC and a PCD.

and with further specific IC dedicated firmware in several alternative versions.

Please note that always the highest version number points to the most recent status of a library coming usually with improvements and/or additional functionality.

Note that each of the versions of the ACL, HSL and the CCL are optional and alternatively. This means that the TOE comes either with one of the alternative library types– depending on the user choice - or with none. The library types can freely be combined but not two versions of the equal type. For example, the ACL with HSL is welcome but a combination of HSL in version v03.11.8339 and at the same time with HSL in version v01.22.4346 would not work. The library types are entirely independent from each other.
The security relevant differences in the versions - if any - are detailed in chapter 2.2.2. The claims of this Security Target take the corresponding versions into account and thus the wording in the following refers to the library type which is cited or else the claim includes always all named versions.

The design step of this TOE is G13.

The Security Target is based on the Protection Profile PP-0084 Security IC Platform Protection Profile with Augmentation Packages [9] as publicly available for download at https://www.bsi.bund.de and certified under BSI-CC-PP-0084-2014.

The Security Target takes into account all relevant current final interpretations.

This TOE concept is based on the architecture, family concept and principles of the Integrity Guard implemented in the controllers by Infineon Technologies AG deemed for high security requiring applications.

The certification body of this process is the German BSI, whereas the abbreviation stands for Federal Office for Information Security, in German language Bundesamt für Sicherheit in der Informationstechnik.

[1] Open Standard for Public Transport (OSPT™) Alliance

Table 1 Identification

	Version	Date	Registration
Security Target	3.9	2026-09-02	This document
Target of Evaluation			The hardware controller with following identifiers: • IFX_CCI_000007h • IFX_CCI_000009h • IFX_CCI_00000Ah • IFX_CCI_00000Bh • IFX_CCI_000016h • IFX_CCI_000017h • IFX_CCI_000018h • IFX_CCI_000023h • IFX_CCI_000024h In the Design Step G13 With FW-Identifiers v80.101.07.2 v80.101.07.1 v80.101.07.0 And following optional SW - libraries: RSA2048 v2.09.002 RSA4096 v2.09.002 EC v2.09.002 Toolbox v2.09.002 HSL-1 v03.12.8812 HSL-2 v03.11.8339 HSL-3 v02.01.6634 HSL-4 v01.22.4346 CIPURSE™ CL v02.00.0004 with belonging user guidance documentation

This TOE is represented by a number of various products which are all based on the equal design sources. The TOE hardware remains entirely equal throughout all derivatives, but the usage for example in form of available memory sizes, availability of the various interfaces, or other functions varies by means of blocking and chip configuration. The firmware identifier on board depends on the order.
All TOE derivatives are derived from the equal hardware design results.

The TOE can be identified with the Generic Chip Identification Mode (GCIM). The IFX_CCI_000007h with options hardware platform is identified by defined bytes of the GCIM as detailed in the HRM [1].
The unique hexadecimal values as stated in the title are:

- IFX_CCI_000007h
- IFX_CCI_000009h
- IFX_CCI_00000Ah
- IFX_CCI_00000Bh
- IFX_CCI_000016h
- IFX_CCI_000017h
- IFX_CCI_000018h
- IFX_CCI_000023h
- IFX_CCI_000024h

These bytes clearly identify the hardware platform, or, in other words, the therein possible values for the IFX_CCI_000007h with options (without prefix IFX_CCI_) represent the equal hardware platform of this TOE. This means that the hardware entirely equals throughout all derivatives and that the differences are achieved by configuration and blocking means only. These values are unique for this hardware platform. This means that these values will not be used in any other platform or product.

The interpretation of the output GCIM data, including also the HW configuration and setting affecting the perturbation robustness, is clearly explained in the user guidance, Hardware Reference Manual HRM [1]. Additionally a new HW configuration is provided, which can be identified via specific GCIM bits, as described in the Errata Sheet [7], section 2.4.
All HW configuration settings achieve the targeted EAL.

Although the TOE is represented by a number of various products, all are based on the equal design sources. Even the TOE hardware and firmware – depending on the order option - remains entirely equal throughout all derivatives, but the usage for example in form of available memory sizes, availability of the various interfaces, or other functions varies by means of blocking and chip configuration. This blocking is applied by Infineon settings during the production only. Again, all TOE derivatives are derived from the equal hardware design results, the IFX_CCI_000007h with options.
The differences between the derivatives are achieved by blocking only and have no impact on the TOEs security policies and related functions. Details are explained in the user guidance hardware reference manual HRM [1].
The products of IFX_CCI_000007h with options enable for a maximum of configuration possibilities defined by the customer order following the market needs. For example, a TOE product can come in one project with the fully available SOLID FLASH™ NVM[1] or in another project with any other SOLID FLASH™ NVM -size below the physical implementation size, or with a different RAM size. And more, the user has the free choice, whether he needs the symmetric co-processor SCP, or the asymmetric co-processor Crypto2304T, or both, or none of them. In addition, the user decides, whether the TOE comes with a free combination of software libraries or without any. And, to be even more flexible, various interface options can be chosen as well.
To sum up the major selections, the user defines by his order:

- the firmware identifier in several alternatives
- the available memory sizes of the SOLID FLASH™ NVM and RAM
- the availability of the cryptographic coprocessors SCP and Crypto@2304T
- the availability of the cryptographic Library
  ACL-0
- the availability of the CIPURSE™ Cryptographic Library

[1] SOLID FLASH™ is an Infineon Trade Mark and stands for the Infineon Flash NVM. The abbreviation NVM is short for Non Volatile Memory. The information remains stored even the power has been removed.

- the availability of the Flash Loader
- the availability of the alternative libraries HSL-1 or HSL-2 or HSL-3 or HSL-4
- the availability of various interface options
- the possibility to tailor the product by blocking on his own premises (BPU)
- the possibility to apply the PIN letter in combination with the Flash Loader

The degree of freedom of the chip configuration is predefined by Infineon Technologies AG and made available via the order tool.

Beside fix TOE configurations, which can be ordered as usual, this TOE implements optionally the so called Billing-Per-Use (BPU) ability. This solution enables our customer to tailor the product on his own to the required configuration – project by project. By that BPU allows for significant reduction of logistic cost at all participating parties and serves for acceleration of delivery of tailored product to the end-user.
BPU enables our customers to block the chip on demand into the final configuration at his own premises, without further delivery or involving support by Infineon Technology.
The realization of it requires the presence of the Flash Loader software, enhanced with the BPU blocking software part. The presence of the BPU ability defines the customer with his order.
If the user decides to use the Flash Loader, regardless whether it is ordered with or without BPU, an additional process option can be ordered which results in an additional status of the Flash Loader. This process is called PIN-Letter and enables for simplified logistics and thereby for faster delivery of the ordered TOE products to the user. The PIN-Letter feature enabling for the PIN-Letter process is an implemented part of the Flash Loader. The resulting logistical acceleration is possible since the PIN-Letter enables for delivery of not user-specific configured, not flashed and not personalized TOE products to the user warehouse.
By delivery the user warehouse gets filled and depending on market demands the user can immediate apply the authentication means of the PIN-letter. If passing, the TOE products become user specific configured and the Flash Loader can be used for this specific user in a second step.
The following table outlines the different ways how the user can input his software on this TOE – a TOE without user available ROM. User software comprises usually the operating system and applications, which are for Infineon Technologies simply a user data package which is handled as a fixed data package during production. This provides high process flexibility for the user of which an overview is given in the following table:

Table 2 Options to implement user software on the chip

Case	Option	Flash Loader Status
1.	The user or/and a subcontractor downloads the software into the SOLID FLASH™ NVM on his own. Infineon Technologies has not received user software and there are no user data of the Composite TOE in the ROM.	The Flash Loader can be activated or reactivated by the user or subcontractor to download his software in the SOLID FLASH™ NVM.
2	The user provides his complete software for the download into the SOLID FLASH™ NVM to Infineon Technologies AG. The software is downloaded to the SOLID FLASH™ NVM during chip production.	The Flash Loader is permanently disabled prior delivery.

3	The user provides software for the download into the SOLID FLASH™ NVM to Infineon Technologies AG. The software is downloaded to the SOLID FLASH™ NVM during chip production.	When leaving the Infineon Technologies production facility, the Flash Loader is blocked, but can be activated or reactivated by the user or subcontractor to complete the previously stored software parts in the SOLID FLASH™ NVM. Precondition is that the user has provided an own reactivation procedure in software prior chip production to Infineon Technologies AG.

For the cases with active Flash Loader on board and whenever the user has finalized his SW-download, respectively the TOE is in the final state and about to be delivered to the end-user, the user is obligated to lock the Flash Loader. This locking is the final step and results in a permanent deactivation of the Flash Loader. This means that once being in the locked status, the Flash Loader cannot be reactivated anymore. Note that whenever a TOE comes without active Flash Loader, BPU and PIN-Letter process are not possible. All in all various delivery combinations are given and for example, a product can come with a fix configuration and with Flash Loader, to enable the user to download software, but without BPU option and with PIN-Letter. The following cases can occur:

Table 3 Options with Flash Loader, BPU and PIN-Letter

Case	Order	Option
1	Fix configuration, Flash Loader is locked (permanent)	• Infineon Technologies configures and flashes all software as ordered. • The entire user software must be delivered to Infineon Technologies prior production.
2	Flash Loader functional, BPU feature blocked	• Infineon configures the chip as ordered and • the user flashes his software at his own premises. • If requested, Infineon Technologies can optionally download also shares of the user software during production. These user software shares must be delivered to Infineon Technologies prior production. The user can finalize his software package at his premises.
3	Flash Loader functional and active BPU feature	The user: • Activates the Flash Loader, • configures the chip applying the BPU feature and • flashes his software at his own premises. • If requested, Infineon Technologies can optionally download also shares of the user software during production. These user software shares must be delivered to Infineon Technologies prior production. The user can finalize his software package at his premises.

Case	Order	Option
4	Flash Loader functional and PIN-Letter	Infineon configures the chip as ordered. The user receives his PIN-letter and fills his warehouse. As required the user: • applies the PIN-Letter on the chips taken from his warehouse, gets the chips user specific configured, • activates the Flash Loader and • the user flashes his software at his own premises. If requested, Infineon Technologies can optionally download also shares of the user software during production. These user software shares must be delivered to Infineon Technologies prior production. The user can finalize his software package at his premises.
5	Flash Loader functional, active BPU and PIN-Letter	Infineon configures the chip as ordered. The user receives his PIN-letter and fills his warehouse. As required the user: • applies the PIN-Letter on the chips taken from his warehouse, gets the chips user specific configured, • activates the Flash Loader, • applies his user specific chip configuration with the BPU feature and • flashes his software at his own premises. If requested, Infineon Technologies can optionally download also shares of the user software during production. These user software shares must be delivered to Infineon Technologies prior production. The user can finalize his software package at his premises.

The listing as given in the confidential security target of this TOE contains the memory size ranges and other blocking options, focusing on the maximum respectively minimum user available limitations. Within those limitations the TOE configurations can vary under only one identical IC-hardware, regardless whether the configurations are set by Infineon or within further limitations by the user. All configurations and all thereof resulting derivatives have no impact on security and are covered by the certificate.

Note that this TOE has no user available ROM. The user software and data are entirely located in a dedicated and protected part of the SOLID FLASH™ NVM. The long life storage endurance together with the means for error detection and correction serves for excellent reliability and endurance.
In addition to the above listed flexible ranges, the user guidance contains a number of predefined configurations for those customers not making use of the BPU option. All of these configurations belong to the TOE as well and are of course made of the equal hardware and are inside the above declared ranges. Today's predefined configurations of the TOE are listed in the hardware reference manual HRM [1] and is completed with the list of identification data of the derivatives. These predefined products come with the most requested configurations and enables to produce volumes on stock in order to simplify logistic processes.
According to the BPU option, a non-limited number of configurations of the TOE may occur in the field. The number of various configurations depends on the user and order contract only.
This TOE provides dedicated identification means and outputs the platform identifier, the design step and further configuration information. The hardware reference manual HRM [1] is part of the user guidance and enables for the clear interpretation of the read out identification data. More information is given in the confidential Security Target [8].
In addition, a dedicated RMS function allows reading out the present configuration in detail. The output RMS data together with the hardware reference manual HRM [1] enables for clear identification of a product and its configuration. All these steps for gathering identification and detailed configuration information can be done by the user himself, without involving Infineon Technologies AG.

The TOE consists of the hardware part, the firmware parts in several alternative versions and the optional software parts. The Smartcard Embedded Software, i.e. the operating system and applications are not part of the TOE.

The firmware comes with several alternative versions and consists of:

- the Boot Software (BOS) firmware conducting configuration and testing task (see chapter 2.2.2) at start-up of the TOE
- the Resource Management System (RMS) library providing essential basis functions for the management of the RAM, the branch table, the Memory Management Unit (MMU) and other resources
- the optional Flash Loader enabling for the download of user software to the SOLID FLASH™ NVM and required for the optional Bill per Use (BPU) feature and the PIN-letter feature

The BOS functions
The firmware part BOS is implemented in a separated Test-ROM also being part of the TOE but not available for the user. Completing firmware components are located in the SOLID FLASH™ NVM.

The optional software parts are differentiated into following libraries:

- The asymmetric cryptographic librariy (ACL) in version v2.09.002 provides
  - RSA 2048/4096 cryptography.
    The library supports also smaller key lengths, but the certification follows the national regulation by the BSI.
  - elliptic curve cryptography EC
- the Toolbox library in the version v2.09.002provides basic mathematical functions for a simplified user interface to the Crypto@2304T. None of the Toolbox libraries is part of the TOE Security Functionality (TSF).
- the hardware support library (HSL) in the versions v01.22.4346 and v02.01.6634 provides a simplified interface and utilizes the full services of the SOLID FLASH™ NVM to the user

- the hardware support library (HSL) in the version v03.11.8339 and v03.12.8812 provides equal to the above a simplified interface, utilizes the full services of the SOLID FLASH™ NVM to the user, and enables the use of an additional advanced reading respectively writing method
- the CIPURSE™ Cryptographic Library (CIPURSE™ CL and CCL) in the version v02.00.0004 provides the user OSPT alliance CIPURSE™ V2 conformant communication functionality between a PICC and a PCD.

The RSA, EC and Toolbox libraries provide certain functionality via an API to the Smartcard Embedded Software. The private parts of the cryptographic libraries are only used internally and have no user interface. If neither the RSA- nor the EC library is delivered, also the belonging private parts are not on board. The Toolbox library does not have private library parts.
Each of the libraries ACL and HSL is independent from the other libraries and also independent of the alternative library version of the equal type. This means for example that the HSL-1 runs alone and does not need parts of HSL-2.
A combination respectively mix up of the two alternative libraries of equal type has not been considered in the design and is not allowed: The user can select either one or the other of the same library type or none of it.
If the user considers the optional software libraries, the TOE can be delivered including - in free combinations - or not including any of the optional libraries.

If the user decides not to use the offered asymmetric cryptographic library, none of the cryptographic libraries is consequently delivered and the accompanying Additional Specific Security Functionality (O.Add-Functions) Rivest-Shamir-Adleman (RSA) and/ or EC is/are not provided by the TOE. Else it depends of the chosen library whether the Additional Specific Security Functionality (O.Add-Functions) Rivest-Shamir-Adleman (RSA) and/ or EC is/are supported.

The Toolbox libraryprovides the user optionally basic arithmetic and modular arithmetic operations, in order to support user software development using long integer operations. These basic arithmetic operations do not provide security functionality, implement no security mechanism, and do not proved additional specific security functionality - as defined for the cryptographic libraries.
The user developed software using the Toolbox basic operations is not part of the TOE.
The Toolbox library is not part of the TOE Security Functionality (TSF).

Beside the inclusion and support of cryptographic libraries this TOE comes with the optional Hardware Support Library (HSL) in different versions significantly simplifying the management of the SOLID FLASH™ NVM functionality. The HSL constitutes an application interface (API) accessing the HSL state machine and abstracting low level properties like special function registers and settings of specific hardware features. In short the HSL provides a user friendly also use case oriented interface considering endurance, reliability and performance requirements.
In certain configurations each of the HSL versions provide also functions implementing tearing safe behaviour of the SOLID FLASH™ NVM. If applied the user has no need to care about cases where the TOE is unintentionally removed from the power supply even during managing the SOLID FLASH™ NVM.
Anyhow, the HSL – regardless of the version - remains as an optional library, as even sudden power off situations do not lead to exploitable conditions of the TOE. In the worse, the TOE ends operation in case of a faulty programmed SOLID FLASH™ NVM location.
In addition to the above, the HSL in version v03.11.8339 and v03.12.8812 provide an advanced additional method called Incremental Write (IWR). This method provides enhanced endurance of the SOLID FLASH NVM even beyond the erase endurance limits for often written objects. Thus this method serves applications with a demand for high endurance and fast writing times. On the other hand, the read times using this method are slower than the other methods provided.

The order option CIPURSE™ Cryptographic Library (CIPURSE™ CL) provides cryptographic functionality to implement a CIPURSE™ V2 conformant protocol.

This protocol provides a secure mutual authentication of two entities, namely the terminal (denoted as PCD = Proximity Coupling Device (CIPURSE™-compliant terminal)) and a smart card or a token in other form factors which is called PICC. PICC stands for Proximity Integrated Circuit Card (CIPURSE™-compliant card). Beside the mutual authentication, the protocol implements measures to maintain the integrity of the transferred data and preserves in parallel the confidentiality of the transferred data.
By that the CIPURSE™ CL supports the user to implement systems conformant to the CIPURSE™ open standard implementing a secured, interoperable and flexible transit fare collection solution, including ISO 7816 communication and AES-128 bit cryptography for multiple payment types.

Deselecting one of the optional libraries does not include the code implementing functionality, which the user decided not to use. Not including the code of the deselected functionality has no impact of any other security policy of the TOE; it is exactly equivalent to the situation where the user decides just not to use the functionality.

All optional software libraries are stored respectively can be loaded into the SOLID FLASH™ NVM.
All other Smartcard Embedded Software loaded into the SOLID FLASH™ NVM does not belong to the TOE and is not subject of the evaluation.

1.2 Target of Evaluation overview

The TOE comprises the Infineon Technologies Security Controller IFX_CCI_000007h with options with specific IC dedicated software, optional cryptographic- and further services libraries.
The TOE is a member of the Infineon Technologies AG high security controller-family meeting the highest requirements in terms of performance and security. A summary product description is given in this Security Target.
This TOE is intended to be used in any application and device requiring the highest level of security, for example as secure element in various devices with various form factors.
This member of the high security controller family features a security philosophy focusing on data integrity instead of numerous sensors. By that two main principles combined in close synergy are utilized in the security concept called the Integrity Guard. These main principles are the comprehensive error detection, including the double CPU, and the full encrypted data path, leaving no plain data on the chip. These principles proved that they provide excellent protection against invasive and non-invasive attacks known today.
The intelligent shielding algorithm finishes the upper layers, finally providing the so called intelligent implicit active shielding I²-shield. This provides physical protection against probing and forcing.
This TOE provides various contact based interface options for various applications and markets. Due to the interface flexibility the product can be used in almost any application, within any device and almost any form factor. Due to these contact based communication possibilities, the TOE can be seen as a stand-alone security device being capable to maintain the various communication interfaces simultaneously. Therefore this TOE is able to run multiple applications, using multiple interfaces independently at the same time.
Again these various contact based communication and application independency capabilities enable the usage to almost everywhere, where highly secure applications are in use and of course in any other application as well. This TOE is deemed for governmental, corporate, transport and payment markets, or wherever a secure root of trust is required. Various types of applications can use this TOE, for example in closed loop logical access controls, physical access controls, secure internet access control and internet authentication, or as multi-application token or simply as encrypted storage.

This controller is able to communicate contact based using following communication protocols respectively methods:

- ISO/IEC 7816-3 card
  This is the ISO/IEC defined standard contact based communication protocol, using the UART and the belonging pads.

- Inter Integrated Circuit Interface (I2C)
  The Inter-Integrated Circuit (IIC) module is able to be connected to an external multi-master-serial-bus-system. The IIC protocol software is not part of the TOE. More information is given in the confidential hardware reference manual [1].
- General Purpose Input/Output (GPIO)
  The GPIO module supports a number of general purpose I/O signals in parallel and independent of each other. Each of the I/O signals can be configured as push-pull or open-drain output with a fast or slow slew rate. The GPIO interface can be directly accessed by the user software.

The TOE provides a real 16-bit CPU-architecture and is compatible to the MCS®251 instruction set with an execution time faster than a standard MCS®251 microcontroller at the same clock frequency. The major components of the core system are the two CPUs (Central Processing Units), acting as one, the MMU (Memory Management Unit) and the MED (Memory Encryption/Decryption Unit). The Core implements also the Post Failure Detection (PFD) covering CPU, Cache and MED. The two CPUs control each other in order to detect faults and serve by this for data integrity. The TOE implements a full 16 MByte linear addressable memory space for each privilege level, a simple scalable Memory Management concept and a scalable stack size. The flexible memory concept consists of a ROM, RAM and the non-volatile memory (NVM), which we call SOLID FLASH™ NVM. The ROM is not available for the user and contains the main parts of firmware components only.

The firmware comes in several alternative versions and is composed out of the:

- Boot-up software (BOS), the Resource Management System (RMS), the Flash Loader (FL). The BOS applies the essential configuration, internal testing and the start-up.
- The RMS implements a low level application interface (API) to the Smartcard Embedded Software and provides handling and managing routines for RAM, MMU, Branch table, configuration and further functions.
- The Flash Loader allows downloading user software to the SOLID FLASH™ NVM during the manufacturing process and also at user premises - if ordered.

This TOE implements a Hybrid Random Number Generator (HRNG). This HRNG equals to the expression Hybrid Physical True Random Number Generator (hybrid PTRNG) as defined by the BSI. In the following, the BSI expression hybrid PTRNG is used. The hybrid PTRNG implements a true physical random source and has evidenced its conformance to the classes of AIS31 [13] as declared in chapter
The produced genuine random numbers are available as a security service for the user and are also used for internal purposes. Together with the guidelines in [6] the hybrid PTRNG operates in the following modes of operation and is conformant to the named classes:

- True Random Number Generation, meeting AIS31 PTG.2
- Hybrid Random Number Generation, meeting AIS31 PTG.3
- Deterministic Random Number Generation (DRNG) AIS31 DRG.3
- Key Stream Generation (KSG), stream cipher generation AIS31 DRG.2

The hybrid PTRNG is deemed for any application requiring excellent physical random data entropy.

The two cryptographic coprocessors serve the need of modern cryptography:

- The symmetric co-processor (SCP) combines both AES and DES with one, two or triple-key hardware acceleration. Please note that the single DES algorithm is not in the scope of evaluation due to national regulation by BSI.
- And, the Asymmetric Crypto Co-processor, called Crypto@2304T, provides optimized high performance calculations for the user software executing cryptographic operations and is also used by the optional cryptographic libraries for Rivest-Shamir-Adleman (RSA) and Elliptic Curve (EC) cryptography.

The optional software parts are differentiated into following libraries:

- The asymmetric cryptographic library(ACL) v2.09.002, provides
  - RSA 2048/4096 cryptography.
    The library supports also smaller key lengths, but the certification follows the national regulation by the BSI.
  - elliptic curve cryptography EC
- the Toolbox library in the version v2.09.002, provides basic mathematical functions for a simplified user interface to the Crypto@2304T. The Toolbox library is not part of the TOE Security Functionality (TSF).
- the hardware support library (HSL) in the versions v01.22.4346 and v02.01.6634 provides a simplified interface and utilizes the full services of the SOLID FLASH™ NVM to the user
- the hardware support library (HSL) in the version v03.11.8339 and v03.12.8812 provides equal to the above a simplified interface, utilizes the full services of the SOLID FLASH™ NVM to the user, and enables the use of an additional advanced reading respectively writing method
- the CIPURSE™ Cryptographic Library (CIPURSE™ CL and CCL) in the version v02.00.0004 provides the user OSPT alliance CIPURSE™ V2 conformant communication functionality between a PICC and a PCD.

The RSA cryptographic library

- provides a high level interface to the hardware component Crypto2304T and includes countermeasures against fault injection and side channel attacks.
- implements RSA signature verification (RsaVerify), the RSA signature generation (RsaSign) and the RSA modulus recalculation (RsaModulus).
- Implements generation of RSA Key Pairs (RsaKeyGen)
- implements the high level interface to hardware cryptographic coprocessor Crypto2304T which runs the basic long number calculations (add, subtract, multiply, square) with high performance.
- can perform RSA operations from 1024 to 4096 bits.

The RSA library is delivered as object code and in this way integrated in the user software.

The EC library:

- provides a high level interface to Elliptic Curve Cryptography computed on the hardware component Crypto2304T and includes countermeasures against fault injection and side channel attacks.
- implements routines for ECDSA signature generation, for ECDSA signature verification, ECDSA key generation and for the Elliptic Curve Diffie-Hellman key agreement.
- In addition, the EC library provides an additional function for calculating primitive elliptic curve operations like ECC Add and ECC Double.
- EC curves over prime field Fp, as well as over GF(2^n) finite field are supported too.

The EC library is delivered as object code and in this way integrated in the user software.
The security functional requirement covers the standard Brainpool [19] and NIST [26] Elliptic Curves with key lengths of 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 bits.
Numerous other curve types, being also secure in terms of side channel attacks on this TOE, exist, which the user optionally can add in the composition certification process.

If the user decides not to use any of the offered cryptographic library(s), regardless of the version, none of the cryptographic libraries is consequently delivered and the accompanying Additional Specific Security Functionality (O.Add-Functions) Rivest-Shamir-Adleman (RSA) and/ or EC is/are not provided by the TOE.

Else it depends of the chosen library whether the Additional Specific Security Functionality (O.Add-Functions) Rivest-Shamir-Adleman (RSA) and/ or EC is/are supported.

The Toolbox library does not provide cryptographic support or additional security functionality as it provides only the following basic long integer arithmetic and modular functions in software, supported by the cryptographic coprocessor: Addition, subtraction, division, multiplication, comparison, reduction, modular addition, modular subtraction, modular multiplication, modular inversion and modular exponentiation. No security relevant policy, mechanism or function is supported. The Toolbox library is deemed for software developers as support for simplified implementation of long integer and modular arithmetic operations. The Toolbox library is not part of the TOE Security Functionality (TSF).

Note that this TOE can come with both cryptographic co-processors accessible, or with a blocked SCP or with a blocked Crypto2304T, or with both cryptographic co-processors blocked. The blocking depends on the user's choice. No accessibility of the deselected cryptographic co-processors is without impact on any other security policy of the TOE; it is exactly equivalent to the situation where the user decides just not to use the cryptographic co-processors.

Beside the inclusion and support of cryptographic libraries this TOE comes with the optional Hardware Support Library (HSL) in different alternative versions, significantly simplifying the management of the SOLID FLASH™ NVM functionality. The HSL constitutes an application interface (API) accessing the HSL state machine and abstracting low level properties like special function registers and settings of specific hardware features. In short the HSL provides a user friendly also use case oriented interface considering endurance, reliability and performance. In certain configurations the HSL provides also functions implementing tearing safe behaviour of the SOLID FLASH™ NVM. If used the user has no need to care about cases where the TOE is suddenly cut off the power supply even during managing the SOLID FLASH™ NVM.
Anyhow, the HSL remains as an optional library as even sudden power off situations does never lead to exploitable conditions of the TOE. In the worse the TOE ends operation in case of a faulty programmed SOLID FLASH™ NVM location due to the Integrity Guard.
In addition to the above, the HSL in version v03.11.8339 and v03.12.8812 provide an advanced additional method called Incremental Write (IWR). This method provides enhanced endurance of the SOLID FLASH NVM even beyond the erase endurance limits for often written objects. Thus this method serves applications with a demand for high endurance and fast writing times. On the other hand, the read times using this method are slower than the other methods provided.

The order option CIPURSE™ Cryptographic Library (CIPURSE™ CL) provides cryptographic functionality to implement a CIPURSE™ V2 conformant protocol.
This protocol provides a secure mutual authentication of two entities, namely the terminal (denoted as PCD = Proximity Coupling Device (CIPURSE™-compliant terminal)) and a smart card or a token in other form factors which is called PICC. PICC stands for Proximity Integrated Circuit Card (CIPURSE™-compliant card). Beside the mutual authentication, the protocol implements measures to maintain the integrity of the transferred data and preserves in parallel the confidentiality of the transferred data.
By that the CIPURSE™ CL supports the user to implement systems conformant to the CIPURSE™ open standard implementing a secured, interoperable and flexible transit fare collection solution, including ISO 7816, and AES-128 bit cryptography for multiple payment types.
The order option in two alternative versions of the CIPURSE™ CL is conformant to the CIPURSE™ open standard [18] for both, the PICC and then PCD software parts. It implements the by the OSPT alliance standardized application interface for the card and the terminal side.
The scope of this certification of this TOE covers all parts of the CIPURSE™ CL versions which are later implemented by the user on the user card respectively token based on this TOE and the functionality of the PCD software part which is implemented in the terminal side. The PCD software operates also on the hardware of this TOE which is implemented in the terminal.

Security Target Introduction (ASE_INT)

The CIPURSE™ CL implements the by the OSPT alliance standardized application interface for the card and the terminal side.

To fulfill the highest security requirements for smartcards today and also in the future, this TOE implements a progressive digital security concept, which already has been certified in various forerunner processes. This TOE utilizes digital security features to include customer friendly security, combined with a robust design overcoming the disadvantages on analogue protection technologies. The TOE provides full on-chip encryption covering the complete core, busses, memories and cryptographic co-processors leaving no plain data on the chip. A further security feature has been implemented for this TOE protecting also the involved addresses transferred over the memory bus. Therefore the attractiveness for attackers is a step further extremely reduced as encrypted signals are of no use for the attacker – neither for manipulation nor for eavesdropping.
In addition, the TOE is equipped with a comprehensive error detection capability for the complete data path. The dual CPU approach allows error detection even while processing. A comparator detects whether a calculation was performed without errors. This approach does not leave any parts of the core circuitry unprotected. The concept allows that the relevant attack scenarios are detected, whereas other conditions that would not lead to an error would mainly be ignored. That renders the TOE robust against environmental influences.
Furthermore, the TOE implements what we call intelligent implicit shielding (I^2). These measures constitute a shield on sensitive and security critical signals which is not recognizable or obvious as a shield. This provides excellent protection against invasive physical attacks, such as probing, forcing or similar.
In this Security Target the TOE is briefly described and a summary specification is given. The security environment of the TOE during its different phases of the lifecycle is defined. The assets are identified which have to be protected through the security policy. The threats against these assets are described. The security objectives and the security policy are defined, as well as the security requirements. These security requirements are built out of the security functional requirements as part of the security policy and the security assurance requirements. These are the formal steps applied during the evaluation and certification showing that the TOE meets the targeted requirements. In addition, simplified functionality of the TOE matching the requirements is described.

The assets, threats, security objectives and the security functional requirements are defined in this Security Target and in the Security IC Platform Protection Profile [9] and are referenced here. These requirements build up a minimal standard common for all Security ICs.
The security functions are defined here in the security target as property of this specific TOE. Here it is shown how this specific TOE fulfills the requirements for the common standard defined in the Common Criteria documents [CCBook2], [CCBook3] and in the Security IC Platform Protection Profile [9].
The TSF of the TOE is divided into two sub TSF. One sub TSF includes all SFR's and is called "ALL_SFR". The other sub TSF are all SFR, which are covered by ADV_SPM and is called "MODELLED_SFR". The assurance for ALL_SFR is EAL5 augmented and for "MODELLED_SFR" EAL6 augmented. For further details see section 7.3.1.1.

2 Target of Evaluation Description

The TOE description helps to understand the specific security environment and the security policy. In this context the assets, threats, security objectives and security functional requirements can be employed. The following is a more detailed description of the TOE than in the Security IC Platform Protection Profile [9] as it belongs to the specific TOE. The Security IC Platform Protection Profile is in general often abbreviated with 'PP' and its version number.

2.1 TOE Definition

This TOE consists of Security Controllers as an integrated circuit (IC), meeting the highest requirements in terms of performance and security. The TOE products are manufactured by Infineon Technologies AG in 65 nm CMOS-technology. This TOE is intended to be used in smart cards and any other form factor for particularly applications requiring highest levels of security and for its previous use as developing platform for smart card operating systems according to the lifecycle model from [9].
The term Smartcard Embedded Software is used in the following for all operating systems and applications stored and executed on the TOE. The TOE is the platform for the Smartcard Embedded Software. The Smartcard Embedded Software itself is not part of the TOE.
The TOE consists of a core system, memories, coprocessors, system peripherals, a control block and the peripherals. The following picture provides a simplified overview upon the hardware components of this TOE which are subsequently briefly described:

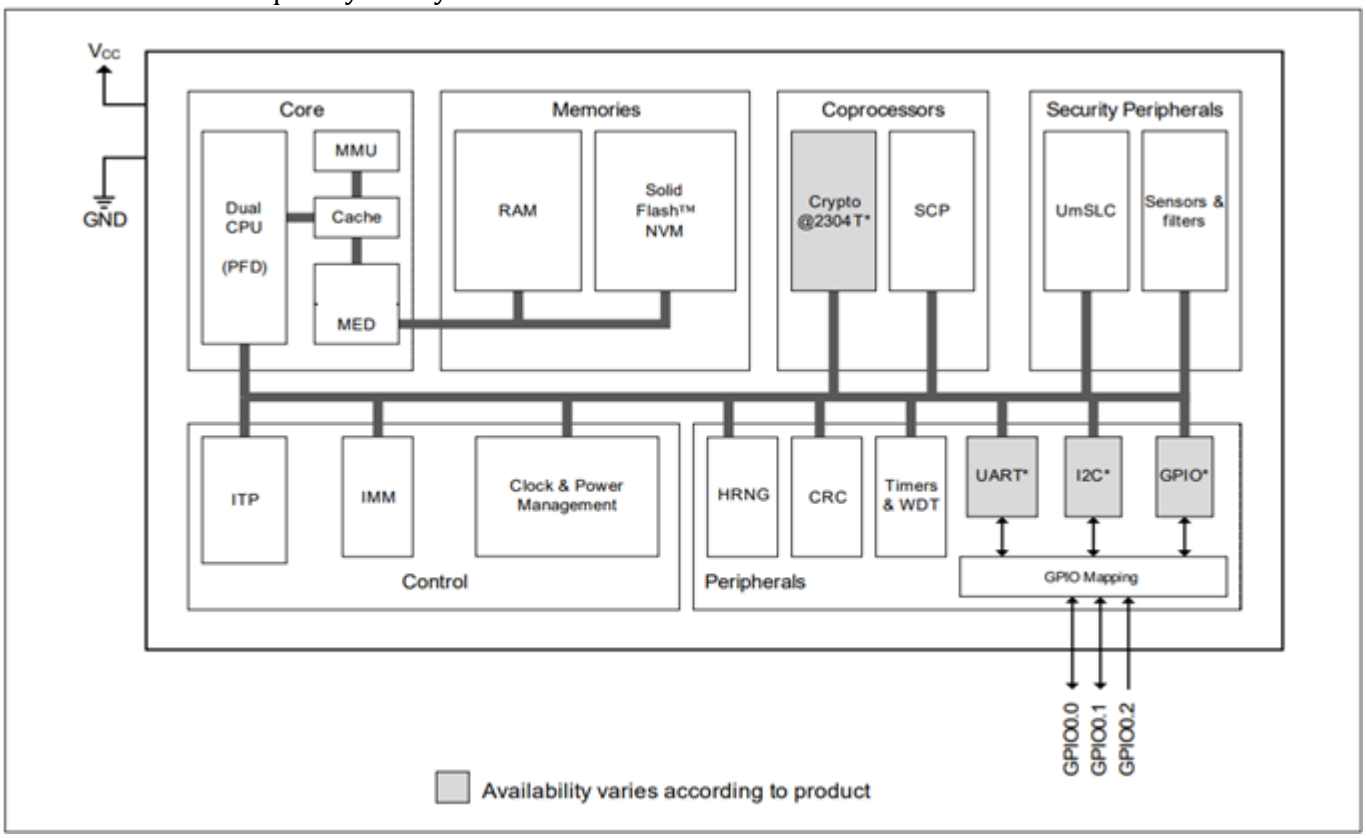


Figure 1 Simplified block diagram of the TOE

Note that the chip also contains a small ROM part which is executed after reset and contains the startup initialization code. The ROM part is not shown in the diagram because it is opaque to the user. The embedded software sees only the hand-over address of the startup code at reset.

Introduction
The main components of the core system are the dual CPU (Central Processing Units), the MMU (Memory Management Unit) and the MED (Memory Encryption/Decryption Unit). The coprocessor block contains the cryptographic processors for RSA/EC and DES/AES processing, while the peripheral block contains the random number generation, the module for the Cyclic Redundancy Check (CRC), the timers and watchdogs and last but not least the external interfaces service.
All data of the memory block is encrypted and all memory types are equipped with an error detection code (EDC), the SOLID FLASH™ NVM in addition with an error correction code (ECC). All data and addresses transferred over the two bus systems are encrypted respectively masked.

The Core
The dual CPU, constituted out of two CPUs and acting as one from users view, is based on a 16-bit architecture based on the MCS® 251 instruction set with an execution time faster than a standard MCS® 251 at the same clock frequency. The instruction set for the architecture is also largely compatible with the well-known 80251 microcontroller family. Anyhow, the CPU has a special internal architecture and timing that differs from the standard 80251 and it provides additional powerful instructions, meeting the requirements for new operating system generations. Despite its compatibility the CPU implementation is entirely proprietary and not standard.
The CPU accesses the memory via the integrated Memory Encryption and Decryption unit (MED). The Post Failure Detection (PFD) covers the modules CPU, Cache and MED, automatically manages the error detection of the individual memories and detects incorrect transfer of data between the memories by means of error code comparison. The access rights of the application to the memories can be controlled with the memory management unit (MMU). Errors in the memories are automatically detected by an error detection code (EDC) and in terms of the SOLID FLASH™ NVM 1-Bit-errors are also corrected (ECC). The two processors of the CPU control each other in order to detect faults and maintain by this the data integrity. A comparator detects whether a calculation was performed without errors and allows error detection even while processing. Therefore the TOE is equipped with a full error detection capability for the complete data path, which does not leave any parts of the circuitry unprotected.
The Cache memory – or simply, the Cache – is a high-speed memory-buffer located between the CPU and the (external) main memories holding a copy of some of the memory contents to enable access to the copy, which is considerably faster than retrieving the information from the main memory. In addition to its fast access speed, the Cache also consumes less power than the main memories. All Cache systems own their usefulness to the principle of locality, meaning that programs are inclined to utilize a particular section of the address space for their processing over a short period of time. By including most or all of such a specific area in the Cache, system performance can be dramatically enhanced. The implemented post failure detection identifies and manages errors if appeared during storage.

The Busses
The bus system comprises two separate bus entities: a memory bus and peripheral bus for high-speed communication internally between the modules and to the outer world with the peripherals. All transfer of data and addresses via the memory and the peripheral bus systems is protected by means of encryption respectively masking leaving no plain contents anywhere on the chip.

The cryptographic Coprocessors
The TOE implements two cryptographic co-processors: The symmetric cryptographic co-processor (SCP) combines both AES and DES with one, two or triple-key hardware acceleration. The Asymmetric Crypto Co-processor, called Crypto2304T, provides optimized high performance calculations for the user software executing cryptographic operations and is also used by the optional cryptographic libraries for RSA and Elliptic Curve (EC) cryptography. These co-processors are especially designed for smart card applications with respect to the security and power consumption. The SCP module computes the complete DES algorithm within a few clock cycles and is especially designed to counter attacks like DPA, EMA and DFA.

Note that this TOE can come with both crypto co-processors accessible, or with a blocked SCP or with a blocked Crypto2304T, or with both crypto co-processors blocked. The blocking depends on the customer demands prior to the production of the hardware. No accessibility of the deselected cryptographic co-processors is without impact on any other security policy of the TOE; it is exactly equivalent to the situation where the user decides just not to use the cryptographic co-processors.

The Memories
The ROM contains the BOS (boot-up software). The RMS (Resource Management System), Flash Loader code and BOS patches are stored in the SOLID FLASH™ NVM. Those parts together compose the TOE firmware in several alternative versions.
All mandatory functions for internal testing, production usage and start-up behavior (BOS), and also the RMS functions are grouped together in a common privilege level. These privilege levels are protected by a hardwired Memory Management Unit (MMU) setting.
The controllers of this TOE store both code and data in a linear 16-MByte memory space, allowing direct access without the need to swap memory segments in and out of memory using a memory management unit.
The user software can be implemented in various options depending on the user's choice and described in chapter 1.1. Thereby the user software, or parts of it, can be downloaded into the SOLID FLASH™ NVM, either during production of the TOE or at customer side. In the latter case, the user downloads his software or the final parts of it at his own premises, using the Flash Loader software.
All content regardless whether stored or transferred remains encrypted and EDC protected. Also the addresses are protected by cryptographic protection means.
The TOE uses also Special Function Registers SFR. These SFR registers are used for general purposes and chip configuration. The start-up register values are stored in the SOLID FLASH™ NVM, in the configuration page area.

The System Peripherals
The system peripheral block serves for operation within the specified ranges and manages the alarms and self-testing. Note that there is a small set of sensors left in order to detect excessive deviations from the specified operational range, while not being over-sensitive. These digital features do not need adjustment or calibration and are deemed to increase the robustness of the chip against environmental influences even more. Conditions that would not be harmful for the operation would therefore in most cases not disturb the proper function.
By implementing the integrity guard concept, the sensors are no more required for the TOE security. The sensors are therefore assigned to be security supporting but do not implement a security mechanism on their own. The only sensor contributing to a security mechanism is the frequency sensor.
After the BOS has finished, the operating system or application can call the User Mode Security Life Control (UMSLC) test. The UMSLC tests several modules, various functions and sensors for correct operation. Some of them have a user configurable interface.

The Control
The Interface Management Module (IMM) handles all interfaces in a unified manner and simplifies by this the variety of interfaces for the user. It provides simultaneous maintenance of a multitude of various interfaces in a non-conflicting manner simultaneously if so configured. The Interrupt and Peripheral Event Channel Controller (ITP) manages individual interrupt requests signaled as events by peripherals.

The Peripherals
This block implements the various interface options, communication protocols and operation modes as outlined in chapter 1.2.
In addition to the interfaces it implements the Hybrid Random Number Generator (HRNG). This HRNG equals to the expression Hybrid Physical True Random Number Generator (hybrid PTRNG) as defined by the BSI. In the following, the BSI expression hybrid PTRNG is used. The hybrid PTRNG implements a true physical random source and has evidenced its conformance to the classes of AIS31 [13] as declared in chapter7.1.3.1.

The produced genuine random numbers are available as a security service for the user and are also used for internal purposes. Together with the guidelines in [6] the hybrid PTRNG operates in the following modes of operation and is conformant to the named classes:

- True Random Number Generation, meeting AIS31 PTG.2
- Hybrid Random Number Generation, meeting AIS31 PTG.3
- Deterministic Random Number Generation (DRNG) AIS31 DRG.3
- Key Stream Generation (KSG), stream cipher generation AIS31 DRG.2

The hybrid PTRNG is deemed for any application requiring excellent physical random data entropy.
Several timer modules are implemented used for example to control the communication via the UART, other interfaces behavior, for asynchronous wake-up and similar timed events.
The timer permits easy implementation of communication protocols such as T=1 and all other time-critical operations. The UART-controlled I/O interface allows the security controller and the terminal interface to be operated independently. The watchdog timers implement a configurable time out for various purposes.
More information can be found in the hardware reference manual HRM [1].
The cyclic redundancy check (CRC) module is used to compute a checksum over any input data and allows by that explicit checking integrity of a piece of data.

Feature Summary

The following is a list of central features provided by this TOE:

- 24-bit linear addressing
- Up to 16 MByte of addressable memory
- Data and addresses protected against eavesdropping
- Register-based architecture
  (registers can be accessed as bytes, words (2 bytes), and double words (4 bytes))
- 2-stage instruction pipeline
- Based MCS® 251 instruction set and largely compatible with the well-known 80251 microcontroller family
- Extensive additional set of powerful instructions, including 16- and 32-bit arithmetic and logic instructions
- Cache with single-cycle access searching
- 16-bit self-checking dual CPU
- Hybrid Physical True Random Number Generation for random numbers at highest entropy quality.
- Extended Temperature Range

The TOE sets a new, improved standard of integrated security features, thereby meeting the requirements of all smart card and other related applications or form factors, such as information integrity, access control, mobile telephone and identification, as well as use cases in electronic funds transfer and healthcare systems.
To sum up, the TOE is a powerful security controller with a large amount of memory and special peripheral devices with improved performance, optimized power consumption, free to choose contact based operation, at minimal chip size while implementing high security. It therefore constitutes the basis for future smart card and other related applications in unlimited form factors.

2.2 Scope of the TOE

The TOE comprises:

- The silicon die, respectively the Integrated Circuit (IC) respectively the hardware of this TOE.
- The TOE is also delivered in various configurations, achieved by means of blocking by the customer and/or depending on the customer order.
- All according firmware in several alternative versions and
- All optional software in various combinations which can be freely combined as desired and ordered

All hardware configurations of any individual TOE product
All product derivatives of this TOE, including all configuration possibilities differentiated by the GCIM data and the configuration information output, are manufactured by Infineon Technologies AG. In the following descriptions the term “manufacturer” stands short for Infineon Technologies AG, the manufacturer of the TOE.
New configurations can occur at any time depending on the user blocking or by different configurations applied by the manufacturer. In any case the user is able to clearly identify the TOE hardware, its configuration and proof the validity of the certificate independently, meaning without involving the manufacturer.
The various blocking options, as well as the means used for the blocking, are done during the manufacturing process and/or at user premises. This depends on the user order. Entirely all means of blocking and the, for the blocking involved firmware respectively software parts, used at Infineon and/or the user premises, are subject of the evaluation. All resulting configurations of a TOE derivative and all possible combinations with optional software libraries are subject of the certificate. Additionally a new HW configuration is provided, which can be identified via specific GCIM bits, as described in the Errata Sheet [7], section 2.4 ID#SYS115b. All resulting hardware configurations are either at the predefined limits or within the predefined configuration ranges.

The firmware in several alternative versions used for the TOE internal testing and TOE operation, the firmware and software parts exclusively used for the blocking, and the parts of the firmware and software required for cryptographic support are part of the TOE and therefore part of the certification. The documents as described in section 2.2.4 and listed in Table 1, are supplied as user guidance.

Not part of the TOE and not part of the certification are:

- the Smartcard Embedded Software respectively user software, and
- the piece of software running at user premises and collecting the BPU receipts coming from the TOE. This BPU software part is the commercially deemed part of the BPU software, not running on the TOE, but allowing refunding the customer, based on the collected user blocking information. The receipt from each blocked TOE is collected by this software – chip by chip.

2.2.1 Hardware of the TOE

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

Core System

Proprietary dual CPU implementation being comparable to the 80251 microcontroller architecture from functional perspective and with enhanced MCS® 251 instruction set
Cache with Post Failure Detection
Memory Encryption/Decryption Unit (MED)
Memory Management Unit (MMU)

Memories

Read-Only Memory (ROM), not available for the user

Random Access Memory (RAM)
SOLID FLASH™ NVM, the flash cell based nonvolatile memory

Buses

Memory Bus

Peripheral Bus

Coprocessors

Crypto2304T for asymmetric algorithms like RSA and EC (optionally blocked)
Symmetric Crypto Co-processor for DES and AES Standards (optionally blocked)

Control

Interface Management Module (IMM)
Interrupt and Peripheral Event Channel Controller (ITP)
Clock & Power Management
Control

System Peripherals

Sensors & Filters
User mode Security Life Control (UmSLC)

Peripherals

Hybrid Physical True Random Number Generator (HPTRNG) implementing also a Deterministic Random Number Generator (DRNG)
Timers and Watchdogs
Cyclic Redundancy Check module (CRC)
Universal Asynchronous Receiver/Transmitter (UART)
Inter-Integrated Circuit module (I2C)
General Purpose Input Output (GPIO)

2.2.2 Firmware and software of the TOE

The following description holds true for all alternative versions with add on for the new version v80.101.07.2. The former FW identifiers implement no difference on the delivered and board of the IC located firmware parts, with the exception of the firmware identifier number itself. The new version v80.101.07.2, based on the predecessor, has been completed with an additional functionality regarding the GCIM identification data output of the TOE, which is an order option. For the TOE security, policy and related claims the firmware identifier is regardless. The user may consider the new version if the GCIM identification data output matters in the application.

The entire firmware in several alternative versions of the TOE consists of the following parts.

- Resource Management System (RMS) with routines for managing the Cache, RAM, MMU, the branch table, configuration and the testing functions.
- Boot Software (BOS), consisting of initialization and various testing routines and providing the different operation modes of the TOE. The BOS is not accessible for the user software.
- Flash Loader: This piece of software enables the download of the user software or parts of it to the SOLID FLASH™ NVM. Depending on the order the Flash Loader comes with the BPU-software enabling for TOE configuration at user premises. After completion of the download and/or final configuration of the TOE, and prior delivery to the end user, the user is obligated to lock the Flash Loader. Locking is the permanent deactivation of the Flash Loader meaning that if once locked it can no more be reactivated and used. Note that the Flash Loader routines are always present, but are deactivated in case of the derivatives ordered without the software download option. Thus the user interface is identically in both cases – with and without Flash Loader on board - and consequently the related interface routines can be called in each of the derivatives. Already the MMU blocks calls of the Flash Loader software at derivatives coming without Flash Loader. In derivatives with Flash Loader the related function is performed.

The firmware version can be identified by the firmware identifier.

The optional software parts are differentiated into following libraries:

- The asymmetric cryptographic library v2.09.002 provides
  - RSA 2048/4096 cryptography.
    The library supports also smaller key lengths, but the certification follows the national regulation by the BSI.
  - elliptic curve cryptography EC
- the Toolbox library in the version v2.09.002 provides basic mathematical functions for a simplified user interface to the Crypto@2304T
- the hardware support library (HSL) in the versions v01.22.4346 and v02.01.6634 provides a simplified interface and utilizes the full services of the SOLID FLASH™ NVM to the user
- the hardware support library (HSL) in the version v03.11.8339 and v03.12.8812 provides equal to the above a simplified interface, utilizes the full services of the SOLID FLASH™ NVM to the user, and enables the use of an additional advanced reading respectively writing method
- the CIPURSE™ Cryptographic Library (CIPURSE™ CL and CCL) in the version v02.00.0004 provides the user OSPT alliance CIPURSE™ V2 conformant communication functionality between a PICC and a PCD.

The RSA cryptographic library:

- provides a high level interface to the hardware component Crypto2304T and includes countermeasures against fault injection and side channel attacks.
- implements RSA signature verification (RsaVerify), the RSA signature generation (RsaSign) and the RSA modulus recalculation (RsaModulus).

- Implements generation of RSA Key Pairs (RsaKeyGen).
- implements the high level interface to hardware cryptographic coprocessor Crypto2304T which runs the basic long number calculations (add, subtract, multiply, square) with high performance.
- can perform RSA operations from 1024 to 4096 bits.

The RSA library is delivered as object code and in this way integrated in the user software.
Depending on the customer's choice, the TOE can be delivered with the 4096 code portion or with the 2048 code portion only. The 2048 code portion is included in both. Part of the evaluation are the RSA straight operations with key length from 1024 bits to 2048 bits, and the RSA CRT[1] operations with key lengths of 1024 Bits to 4096 Bits.

The EC library used to:

- provide a high level interface to Elliptic Curve Cryptography computed on the hardware component Crypto2304T and includes countermeasures against fault injection and side channel attacks.
- implement routines for ECDSA signature generation, for ECDSA signature verification, ECDSA key generation and for the Elliptic Curve Diffie-Hellman key agreement.
- implement additional functions for calculating primitive elliptic curve operations like ECC Add and ECC Double interfaces, the EC library provides an .
- implement a high level interface to Elliptic Curve Cryptography computed on the hardware component Crypto2304T and includes countermeasures against fault injection and side channel attacks.
- implements routines for ECDSA signature generation, for ECDSA signature verification, ECDSA key generation and for the Elliptic Curve Diffie-Hellman key agreement.
- In addition, the EC library provides an additional function for calculating primitive elliptic curve operations like ECC Add and ECC Double.
- over prime field Fp, as well as over GF(2^n) finite field are supported too.

The EC library is delivered as object code and in this way integrated in the user software.
The security functional requirement covers the standard Brainpool [19] and NIST [26] Elliptic Curves with key lengths of 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 bits. Numerous other curve types, being also secure in terms of side channel attacks on this TOE, exist, which the user optionally can add in the composition certification process.

The Toolbox library does not provide cryptographic support or additional security functionality as it provides only the following basic long integer arithmetic and modular functions in software, supported by the cryptographic coprocessor: Addition, subtraction, division, multiplication, comparison, reduction, modular addition, modular subtraction, modular multiplication, modular inversion and modular exponentiation. No security relevant policy, mechanism or function is supported. The Toolbox library is deemed for software developers as support for simplified implementation of long integer and modular arithmetic operations. The Toolbox library is not part of the TOE Security Functionality (TSF).

Beside the inclusion and support of cryptographic libraries this TOE comes with the optional Hardware Support Library (HSL) in different alternative versions significantly simplifying the management of the SOLID FLASH™ NVM functionality. The HSL (all versions) constitutes an application interface (API) accessing the HSL state machine and abstracting low level properties like special function registers and settings of specific hardware features. In short the HSL provides a user friendly also use case oriented interface considering endurance, reliability and performance.
All HSL library versions are delivered as object code.
The HSL is delivered as object code and in this way integrated in the user software.

[1] CRT: Chinese Remainder Theorem

All HSL versions implement, beyond the low level driver, the basic method In-place Update leveraging the dedicated advantages of the new SOLID FLASH™ NVM technology.
This method provides enhanced endurance of the SOLID FLASH NVM even beyond the erase endurance limits for often written objects. Thus this method serves applications with a demand for high endurance and fast writing times. On the other hand, the read times using this method are slower than the other methods provided.
The HSL versions provide tearing save behavior, whereas we define tearing as an untimed power cut off which could occur anytime and in the worse therewith could also occur during writing to or erasing of pages in the SOLID FLASH™ NVM.
If the HSL – all versions are meant - comes with the TOE and the user implements the offered configuration and dedicated functions tearing save behaviour of the SOLID FLASH™ NVM is provided. In these cases the user does not need to care about tearing events since either the old data or the new data are correctly in place.
In addition, HSL versions v03.11.8339 und v03.12.8812 provide an advanced additional method called Incremental Write (IWR).
But, even in the cases where the user decides not to use the HSL and did also not implement own routines preserving the consistency of the SOLID FLASH™ NVM, the hardware protection means prevent from operation of inconsistent data. Therefore, in no cases a tearing event leads to an exploitable situation respectively vulnerability.
Anyhow, the user should be aware and is recommended to use either the HSL or implement own routines managing tearing events since if there would occur a faulty programmed SOLID FLASH™ NVM location the TOE ends operation at that point.

The order option CIPURSE™ Cryptographic Library (CIPURSE™ CL) provides cryptographic functionality to implement a CIPURSE™ V2 conformant protocol.
This protocol provides a secure mutual authentication of two entities, namely the terminal (denoted as PCD = Proximity Coupling Device (CIPURSE™-compliant terminal)) and a smart card or a token in other form factors which is called PICC. PICC stands for Proximity Integrated Circuit Card (CIPURSE™-compliant card). Beside the mutual authentication, the protocol implements measures to maintain the integrity of the after passing successfully the authentication transferred data. It depends on the chosen operation mode whether the user requires integrity protection only, for example if the exchange is used in a secure environment only, or whether complete protection including the encrypted transfer of user data between the two authentication entities is an issue. Both operation modes are part of the CIPURSE™ open standard and offered as integrity protection mode and as confidential communication mode.
The CIPURSE™ CL supports the user to implement systems conformant to the CIPURSE™ open standard implementing a secured, interoperable and flexible transit fare collection solution, including ISO 7816 communication and AES-128 bit cryptography for multiple payment types.
The CIPURSE™ CL is conformant to the CIPURSE™ open standard [18] for both, the PICC and then PCD software parts. . It implements the by the OSPT alliance standardized application interface for the card and the terminal side.
The scope of this certification of this TOE covers all parts of the CIPURSE™ CL which are later implemented by the user on the user card respectively token based on this TOE and the functionality of the PCD software part which is implemented in the terminal side. The PCD software operates also on the hardware of this TOE which is implemented in the terminal.
The certification comprises the entire functionality of the optional CIPURSE™ CL implemented and operated on the TOE hardware. On one hand the TOE can operate the PICC side software part as a token and on the other hand, a second TOE product operates the PCD side software part if used inside a terminal or similar system.
The environment on the terminal, the terminal systems, their security and their interfaces to the background systems are not in the scope of this certification. The user operating system and further applications implemented on the TOE are also out of scope of this certification.
To summarize, if used with the PCD software the certification view equals to the case where the TOE is running the PICC software: The TOE operates one of the optional software parts of the CIPURSE™ CL –

regardless whether PICC or PCD part - and is enabled to communicate via the selected interfaces. The surrounding environment is in both cases out of scope. The CIPURSE™ CL is delivered as object code and in this way integrated in the user software.

Note 1:
If the TOE is delivered without the asymmetric cryptographic library, it does not provide the Additional Specific Security Functionality Rivest-Shamir-Adleman Cryptography (RSA) and/or Elliptic Curve Cryptography (EC).

The cryptographic library CIPURSE™ CL is a delivery option.
Therefore the TOE may come with free combinations with the other libraries of or without these libraries. In the case of coming without the CIPURSE™ CL, the TOE does not provide the specific security functionality implemented by this software.
End of note.

The firmware and software parts of the TOE comprise:

2.2.2.1 Firmware

- in several alternative versions,
  whereas the most recent version provides additional functionality regarding the TOE identification as order option
- Boot Software (BOS)
- Resource Management System (RMS)
- Flash Loader (FL)

2.2.2.2 Optional Software

- RSA cryptographic library
- EC cryptographic library
- Toolbox library. The Toolbox library is not part of the TOE Security Functionality (TSF).
- HSL library in four alternative versions
- CIPURSE™ Cryptographic Library

2.2.3 Interfaces of the TOE

- The physical interface of the TOE to the external environment is the entire surface of the IC.
- The electrical interface of the TOE to the external environment is constituted by the pads of the chip, particularly ISO pads, the CLK and power supply pads, as well as the pads used for contact based interfacing.
- The data-oriented I/O interface to the TOE is formed by the pads used for contact based communication.
- The interface to the firmware – regardless of the version in use - is constituted by special registers used for hardware configuration and control (Special Function Registers, SFR).
- The interface of the TOE to the operating system is constituted by the RMS and by the instruction set of the TOE.
- The interface of the TOE to the test routines is formed by the BOS test routine call, i.e. entry to the test modes.

- The interface to the RSA calculations is defined from the RSA library interface. This is regardless of the version chosen.
- The interface to the EC calculations is defined from the EC library interface. This is regardless of the version chosen.
- The interface to the Toolbox is defined by the Toolbox library interface. This is regardless of the version chosen.
- The interface to the HSL is defined by the functions of the Hardware Support Library interface. This is regardless of the version chosen.
- The interface to the CIPURSE™ CL is defined by the functions of the CIPURSE™ cryptographic library.

Note that the interfaces to the optional software parts are only present, if the TOE comes with the belonging software part depending on the customer order.

2.2.4 Guidance documentation

The following provides a brief overview of the document set constituting the user guidance for this TOE. The exact document titles and versions are given in chapter 9.

- The document Hardware Reference Manual HRM [1] is the user data book of the TOE and contains the relevant module, function and feature description
- The document Production and Personalization User Manual [2] contains detailed information about the usage of the Flash Loader
- The document Programmers Reference Manual [3] describes the usage and interfaces of the Resource Management System RMS.
- The document [6] Security Guidelines User Manual provides the guidance and recommendations to develop secure software for and secure usage of this TOE.
- The document [7] Errata Sheet contains latest updates and corrections of the TOE relevant for the user and it is a kind addendum to the hardware reference manual HRM [1]. The Errata Sheet can be changed during the life cycle of the TOE. New Errata Sheet releases are reported in a monthly updated list provided from Infineon Technologies AG to the user. This list is not part of the certification process. Part of the TOE certification is the released version valid at the point in time the certificate was issued.
- The document asymmetric Cryptographic Library ACL [4-0] for Crypto@2304T user interface for the version v2.09.002 contains all interfaces of the RSA, EC and Toolbox library and are only delivered to the user in case the RSA library and/or the EC library is/are part of the delivered TOE.
- The document [5] Crypto@2304T User Manual describes the architecture of cryptographic coprocessor on register level. It also provides a functional description of the register architecture, instruction set and gives programming guidance. This document is not required if one of Infineon's asymmetric cryptographic libraries ACL is used. The ACL takes properly care about the low level hardware interfaces.
- The user guidance for the CIPURSE™ Cryptographic Library [17-1] provides detailed information and the complete application interface for the user for implementation of an OSPT™ compliant PCD / PICC communication solution.
- The document reference [15-1] for the Hardware Support Library (HSL) in version v03.12.8812 provides an application interface (API) accessing the HSM state machine and abstracting low level properties like special function registers and settings of specific hardware features.
- The document reference [15-2] for the Hardware Support Library (HSL) in version v03.11.8339 provides an application interface (API) accessing the HSM state machine and abstracting low level properties like special function registers and settings of specific hardware features.
- The document reference [15-3] for the Hardware Support Library (HSL) in version v02.01.6634 provides an application interface (API) accessing the HSM state machine and abstracting low level properties like special function registers and settings of specific hardware features.
- The document reference [15-4] for the Hardware Support Library (HSL) in version v01.22.4346 provides an application interface (API) accessing the HSM state machine and abstracting low level properties like special function registers and settings of specific hardware features.
- Finally the certification report by BSI may contain an overview of the recommendations to the software developer regarding the secure use of the TOE. These recommendations are also included in the ordinary user documentation, the Security Guidelines User Manual [6].

2.2.5 Forms of delivery

The TOE can be delivered:

- in form of complete modules
- with or without inlay mounting

- in form of plain wafers
- in any IC case (for example TSSOP28, VQFN32, VQFN40, CCS-modules, etc.)
- in no IC case or package, simply as bare dies
- or in whatever type of package

The form of delivery does not affect the TOE security and it can be delivered in any type, as long as the processes applied and sites involved have been subject of the appropriate audit.
The delivery can therefore be at the end of phase 3 or at the end of phase 4 which can also include pre-personalization steps according to PP [9]. Nevertheless in both cases the TOE is finished and the extended test features are removed. In this document are always both cases mentioned to avoid incorrectness but from the security policy point of view the two cases are identical.
The delivery to the software developer (phase 2 ➔ phase 1) contains the development package and is delivered in form of documentation as described above, data carriers and/or download from a secure server containing the tools and emulators as development and debugging tool.
Part of the software delivery can also be the Flash Loader software running on the TOE and receiving the transmitted information of the user software to be loaded into the SOLID FLASH™ NVM. The download is only possible after successful mutual authentication and the user software can also be downloaded in an encrypted way. This download can also be done at a third party acting as download service provider for the user. This organization is subject of the user.
In any case, the user is after finalization of the download and prior delivery to the field obligated to permanently lock further use of the Flash Loader. How the user can do this is detailed in the user guidance [2].
Note that it depends on the procurement order, whether the Flash Loader program is present or not.

The belonging user guidance documents are delivered in electronic form: Either by user downloads from a secure server or alternatively on request as encrypted email attachment.

Table 4 TOE delivery types

TOE Component	Delivered Format	Delivery Method	Comment
Hardware			
All derivatives as stated in chapter 1	Plain wafers, bare dies, complete modules, IC cases of any type (for example TSSOP28, VQFN32, VQFN40, CCS-modules, etc.), with or without inlay mounting, or in whatever package	Customer chooses delivery method	All materials are delivered to distribution centres in locked cages or metal boxes.
Firmware			
All	–	–	Stored on the delivered hardware.
Software			
All software libraries	Precompiled object format code – L251 file type with ending ".lib"	Secured download as zipped-file (1)	–
Guidance Documentation			
All User Guidance documents	Personalized pdf or compiled help files (chm)	Secured download (1) or on demand by encrypted email.	–

(1) Users having an account on the secure server can login and download the according document- respectively software files

2.2.6 Production sites

The TOE may be handled in different production sites but the silicon of this TOE is produced in Tainan, Taiwan only, as listed below. To distinguish the different production sites of various products in the field, the site is coded into the Generic Chip Ident Mode (GCIM) data. The exact coding of the generic chip identification data is described in the hardware reference manual HRM [1].
The delivery measures are described in the ALC_DEL aspect.
The production site of this TOE is Taiwan, Fab14, coded with 0Ah.

3 Conformance Claims (ASE_CCL)

This ST and TOE claim conformance to CC:2022. The ST claims conformance to [CCBook3]. It is [CCBook2] extended. [CCErrata] and [CCtrans] is taken into consideration.

3.1 PP Claim

This ST is strictly conformant to [PP0084].

The Security IC Platform Protection Profile with Augmentation Packages is registered and certified by the Bundesamt für Sicherheit in der Informationstechnik[1] (BSI) under the reference [PP0084].

The security assurance requirements of the TOE are according to [PP0084] and [CCBook3].

3.2 Package Claim

This ST claims conformance to the following additional packages taken from [PP0084]:

- Package Authentication of the Security IC, section 7.2, conformant.
- Package Loader, Package 1: Loader dedicated for usage in secured environment only, section 7.3.1, conformant. This package is optional and fulfilled only by TOE products with Flash Loader.
- Package Loader, Package 2: Loader dedicated for usage by authorized users only, section 7.3.2, conformant.
- Package TDES ; section 7.4.1, augmented
- Package AES ; section 7.4.2, augmented

The assurance level is multi-assurance with global assurance level EAL5 augmented with ADV_IMP.2, ADV_INT.3, ADV_TDS.5, ALC_CMC.5, ALC_DVS.2, ALC_FLR.3, ALC_TAT.3, ATE_FUN.2, ATE_COV.3, AVA_VAN.5 and sub-TSF assurance level EAL6 augmented with ALC_FLR.3

The assurance for the EAL5 augmented sub TSF ALL_SFR is equivalent compared to EAL6 augmented sub TSF MODELLED_SFR with the exception of ADV_SPM.1, which is not claimed for the EAL5 augmented sub TSF ALL_SFR.

3.3 Conformance Rationale

The TOE is a typical security IC as defined in [PP0084] chapter 1.2.2 comprising:

- the circuitry of the IC (hardware including the physical memories),
- configuration data, initialization data related to the IC Dedicated Software and the behaviour of the security functionality
- the IC Dedicated Software with the parts
    - the IC Dedicated Test Software,
    - the IC Dedicated Support Software.

The TOE is designed, produced and/or generated by the TOE Manufacturer.

3.4 Adding Objectives

Due to additional security functionality coming from

- the cryptographic libraries

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

  - O.Add-Functions,
- the memory access control
  - O.Mem-Access
- and objectives related to the Flash Loader
  - O.Authentication,
  - O.Cap_Avail_Loader,
  - O.Ctrl_Auth_Loader and

additional security objectives have been introduced.
These add-ons have no impact on the conformance statements regarding [CCBook1] and PP [9] with following rational:

- The security target remains conformant to [CCBook1],E.3 as the possibility to introduce additional restrictions is given.
- The security target fulfils the strict conformance of the PP [9] due to the application note 9 applying here. This note allows the definition of high-level security goals due to further functions or services provided to the Security IC Embedded Software.

3.5 AES and TDES

The PP [9] implements the optional policy cryptographic services P.Crypto_Service with its packages TDES and AES. This TOE provides these optional packages requiring secure hardware based cryptographic services for the IC Embedded Software as outlined in chapter 7.1.3.1.
Due to these optional additional security functionalities the security objectives O.TDES and O.AES have been introduced. These add-ons have no impact on the conformance statements regarding [CCBook1] and PP [9], with following rational:

- The security target fulfils the strict conformance claim of the PP [9] due to the application notes applying here. By these notes the addition of further security functions and security services are covered, even without deriving particular security functionality from a threat or a policy.

3.6 Loader

The PP [9] implements the optional policy for applying a Loader. The Loader is used to load data into the SOLID FLASH™ NVM.
The Flash Loader provides the service for authentication and implements the Package for Authentication of the Security IC containing FIA_API.1 Authentication Proof of Identity of the TOE against a user.
This means that the user clearly can identify the TOE on his external request. This fulfils the objective O.Authentication, authentication to external entities, and obligates an objective to the environment OE.TOE_Auth, external entities authenticating of the TOE as outlined in the PP [9].
The Loader policy defines the Package 1 with its policy P.LIM_Block_Loader where the Loader is dedicated for usage in secured environment only and the Package 2 with its policy P.Ctlr_Loader where the Loader is dedicated for usage by authorized users only.

This TOE provides a Flash Loader complying with the optional packages:

- Package 1:
  Loader dedicated for usage in secured environment only
- Package 2:
  Loader dedicated for usage by authorized users only as outlined in sections 7.2 and 7.3 of the PP [9]

Due to these optional additional security functionalities the security objectives

- O.Cap_Avail_Loader, capability and availability of the Loader,
- O.Ctrl_Auth_Loader, access control and authenticity for the Loader and

- OE.Loader_Usage. secure communication and usage of the Loader
- O.Prot_TSF_Confidentiality, protection of the confidentiality of the TSF
- OE.Lim_Block_Loader, limitation of capability and blocking the Loader

have been introduced.

3.7 CIPURSE Cryptographic Library (CCL)

The TOE provides "Additional Specific Security Functionality (O.Add-Functions)" including the CIPURSE™ Cryptography. This cryptographic functionality implements additional cryptographic services by implementation of a protocol providing a secure mutual authentication of two entities, namely the terminal (denoted as PCD = Proximity Coupling Device (CIPURSE™-compliant terminal)) and a smart card or a token in other form factors which is called PICC. PICC stands for Proximity Integrated Circuit Card (CIPURSE™-compliant card).
Beside the mutual authentication, the protocol implements measures to maintain the integrity of the transferred data and preserves in parallel the confidentiality of the transferred data.
Due to these additional security functionalities the security objectives

- O.Ctrl_Auth_CCL
- O.Prot_Integrity
- O.Prot_Confidentiality

have been introduced.

3.8 Summary

All of the above add-ons have no impact on the conformance statements regarding [CCBook1] and PP [9], with following rational:

The security target fulfils the strict conformance claim of the PP [9] due to the application notes 9 applying here. By this note the addition of further security functions and security services are covered, even without deriving particular security functionality from a threat or a policy.

Due to the above rational, the security objectives of this security target are consistent with the statement of the security objectives in the PP [9], as the security target claims package augmentation to the PP [9].

All security functional requirements defined in the PP [9] are included and completely defined in this ST.

A listing of all security functional requirements and where those are taken from is given in chapter 7.1.

The following security functional requirement is included and completely defined in this ST, section 6.

FPT_TST.2	Subset TOE security testing[1]

All assignments and selections of the security functional requirements are done in the PP [9] and in this Security Target.

3.9 Application Notes

The functional requirements

With CC:2022 several SFR changes are introduced. Due to this ST claiming conformance to CC:2022 and [PP0084], rationales are provided that these changes do not affect the conformance claim to [PP0084]:

- FCS_COP.1: for this SFR dependencies are changed in CC:2022. FCS_CKM.4 is removed and instead FCS_CKM.6 added. Further FCS_CKM.5 is added for key derivation as an alternative.

[1] Requirement from the PP [9]

- FCS_CKM.1: for this SFR dependencies are changed in CC:2022. Additionally to FCS_CKM.2 and FCS_COP.1, one further SFR is introduced as alternative: FCS_CKM.5. This SFR targets key derivation, subsequent to FCS_CKM.1. In CC:2022 key derivation would have been part of FCS_CKM.1 and thus conformancy to [PP0084] can still be claimed. FCS_CKM.4 is removed and instead FCS_CKM.6 added. All other dependencies (i.e. FCS_RNG.1 or FCS_RBG.1) are in addition to the already existing ones, i.e. add stricter requirements.
- FCS_CKM.6 replaces FCS_CKM.4 and adds further requirements on the timing of key destruction. As an alternative dependency to FCS_CKM.1, FCS_CKM.5 (key derivation) can be used. As FCS_CKM.5 is neither used within [PP0084] nor within this ST, it has no relevance in this context.
- FCS_RNG.1: this SFR is taken from [CCBook2] rather than [PP0084]. The SFR is identical in [CCBook2]
- FMT_LIM.1 and FMT_LIM.2 are taken from [CCBook2] rather than [PP0084]. There is a slightly different phrasing (i.e. removing redundancy from FMT_LIM.1) and availability and capability policy mentioned in both SFR's. The meaning though is the same and therefore conformancy can still be claimed.
- FIA_API.1: this SFR is taken from [CCBook2] rather than [PP0084]. The SFR requires additional information.
- FDP_SDC.1: this SFR is taken from [CCBook2] rather than [PP0084]; An assignment from [PP0084] is changed to a selection [CCbook2], which means the requirement is more stringent and thus can be considered a subset of the requirement from [PP0084].

Further with CC:2022 some SAR changes were introduced. Rationales are provided that these changes do not affect the conformance claim to [PP0084]:

- ASE_CCL.1: for CC:2022 several extensions were introduced (e.g. exact conformance to PP), which add to the already existing assurance requirements. No relaxation was introduced.
- ASE_INT.1: introduction of multi-assurance in combination with PP-configuration: not relevant for [PP0084]
- ASE_REQ.2: extended for multi assurance: not relevant for [PP0084]
- AVA_VAN.5: extension about third party components introduced. No relaxation was introduced.
- ALC_TAT.1: extension with guidance on the minimum content for an implementation standards description and rules with ADV_COMP.1. No relaxation was introduced.
- ADV_SPM: the major difference is the requirement of the formal model to cover the complete set of SFRs that define the TSF. This requirement is the reason for the multi assurance claim, because the formal model does not cover the entire SFRs. Conformance to [PP0084] however is not affected, as all assurance claims are beyond the ones requested by [PP0084].
  - FCS_RNG.1/HRNG
  - FCS_RNG.1/TRNG,
  - FCS_RNG.1/DRNG and
  - FCS_RNG.1/KSG

are iterations of the FCS_RNG.1 as defined in the Protection Profile [9] according to "Anwendungshinweise und Interpretationen zum Schema (AIS)" respectively "Functionality classes and evaluation methodology for physical random number generators", AIS31 [13].

4 Security Problem Definition (ASE_SPD)

The content of the PP [9] applies to this chapter completely.

4.1 Threats

The threats are directed against the assets and/or the security functions of the TOE. For example, certain attacks are only one step towards a disclosure of assets while others may directly lead to a compromise of the application security. The more detailed description of specific attacks is given later on in the process of evaluation and certification.
The threats to security are defined and described in PP [9] section 3.2, respectively for T.Masquerade_TOE in chapter 7.2.1.

Table 5 Threats according PP [9]

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

4.1.1 Additional Threat due to TOE specific Functionality

Threat Memory Access Violation
The additional functionality of introducing sophisticated privilege levels and access control allows the secure separation between the operation system(s) and applications, the secure downloading of applications after personalization and enables multitasking by separating memory areas and performing access controls between different applications. Due to this additional functionality “area based memory access control” a new threat is introduced.
The Smartcard Embedded Software is responsible for its User data of the Composite TOE according to the assumption “Treatment of User data of the Composite TOE (A.Resp-Appl)”. However, the Smartcard Embedded Software may comprise different parts, for instance an operating system and one or more applications. In this case, such parts may accidentally or deliberately access data (including code) of other parts, which may result in a security violation.

The TOE shall avert the threat “Memory Access Violation (T.Mem-Access)” as specified below:

T.Mem-Access	**Memory Access Violation** Parts of the Smartcard Embedded Software may cause security violations by accidentally or deliberately accessing restricted data (which may include code) or privilege levels. Any restrictions are defined by the security policy of the specific application context and must be implemented by the Smartcard Embedded Software.

Threat Diffusion of Open Samples
The additional functionality of a Loader as defined in the PP [9], section 7.3 requires to address the following threat, as defined in the document “PP0084: Interpretation” [9.1].
The TOE shall avert the threat “Diffusion of open Samples (T.Open_Samples_Diffusion)” as specified below:

T.Open_Samples_Diffusion	**Diffusion of Open Samples** An attacker may get access to open samples of the TOE and use them to gain information about the TSF (loader, memory management unit, ROM code …). He may also use the open samples to characterize the behavior of the IC and its security functionalities (for example: characterization of side channel profiles, perturbation cartography …). The execution of a dedicated security features (for example: execution of a DES computation without countermeasures or by de-activating countermeasures) through the loading of an adequate code would allow this kind of characterization and the execution of enhanced attacks on the IC.

Note:
This threat is only relevant if the Flash Loader is still available and the attacker would in addition have the authentication key. If both would be given, then this would enable downloading additional software.
End of note.

Table 6 Additional threats due to TOE specific functions and augmentations

T.Mem-Access	Memory Access Violation	Hardware
T.Open_Samples_Diffusion	Diffusion of open samples	Flash Loader

4.1.2 Assets regarding the Threats

The primary assets concern the User data which includes the user data of the Composite TOE as well as program code (Security IC Embedded Software) stored and in operation and the provided security services. These assets have to be protected while being executed and or processed and on the other hand, when the TOE is not in operation.
This leads to four primary assets with its related security concerns:

- SC1 integrity of user data of the Composite TOE
- SC2 confidentiality of user data of the Composite TOE being stored in the TOE’s protected memory areas
- SC3 correct operation of the security services provided by the TOE for the Security IC Embedded Software
- SC4 continuous availability of random numbers

SC4 is an additional security service provided by this TOE which is the availability of random numbers. These random numbers are generated either by a true random number or a deterministic random number generator or by both, when a true random number is used as seed for the deterministic random number generator. Note that the generation of random numbers is a requirement of the PP [9].

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

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

The information and material produced and/or processed by the TOE Manufacturer in the TOE development and production environment (Phases 2 up to TOE Delivery) can be grouped as follows:

- logical design data,
- physical design data,
- IC Dedicated Software, Security IC Embedded Software, Initialization Data and Pre-personalization Data,
- specific development aids,
- test and characterization related data,
- material for software development support, and
- reticles and products in any form

as long as they are generated, stored, or processed by the TOE Manufacturer.
For details see PP [9] section 3.1.

4.2 Organizational Security Policies

The TOE has to be protected during the first phases of their lifecycle (phases 2 up to TOE delivery which can be after phase 3 or phase 4). Later on each variant of the TOE has to protect itself. The organizational security policy according to PP [9] covers this aspect.

P.Process-TOE	**Identification during TOE Development and Production** An accurate identification must be established for the TOE. This requires that each instantiation of the TOE carries this unique identification.

Due to the augmentations of PP [9] and the chosen packages additional policies are introduced and described in the next chapter.

4.2.1 Augmented Organizational Security Policy

Due to the augmentations of the PP [9] and the chosen packages additional policies are introduced.
The TOE provides specific security functionality, which can be used by the Smartcard Embedded Software. In the following specific security functionality is listed which is not derived from threats identified for the TOE's environment because it can only be decided in the context of the smartcard application, against which threats the Smartcard Embedded Software will use the specific security functionality.
The IC Developer / Manufacturer must apply the policy "Additional Specific Security Functionality (P.Add-Functions)" as specified below.

P.Add-Functions	**Additional Specific Security Functionality** The TOE shall provide the following specific security functionality to the Smartcard Embedded Software: • Rivest-Shamir-Adleman Cryptography (RSA) • Elliptic Curve Cryptography (EC) • CIPURSE™ Cryptographic Library (CCL)

Note 2:
The cryptographic libraries RSA, EC, CCL and the Toolbox library as stated in chapter 1.1 are delivery options. Therefore the TOE may come with free combinations of or even without these libraries. In the case of coming without one or any combination of the cryptographic libraries RSA, EC, CCL and the TOE does not provide the Additional Specific Security Functionality Rivest-Shamir-Adleman Cryptography (RSA) and/or Elliptic Curve Cryptography (EC) and/or CIPURSE™ Cryptographic Library.
The Toolbox Library is no cryptographic library, provides no additional specific security functionality and thus the Toolbox library is not part of the TOE Security Functionality (TSF).
End of note.

The IC Developer / Manufacturer must apply the organizational security policy "Cryptographic services of the TOE (P.Crypto-Service)" as specified below:

P.Crypto-Service	**Cryptographic services of the TOE** The TOE provides secure hardware based cryptographic services for the IC Embedded Software. • Triple Data Encryption Standard (TDES) • Advanced Encryption Standard (AES)

Note 3:
This TOE can come with both cryptographic coprocessors accessible, or with a blocked SCP or with a blocked Crypto2304T, or with both crypto co-processors blocked. The blocking depends on the customer demands prior to the production of the hardware. In case the SCP is blocked, no AES and DES computation supported by hardware is possible. In case the Crypto2304T is blocked, no RSA and EC computation – of any version as stated in chapter 1.1 - supported by hardware is possible. No accessibility of the deselected cryptographic co-processors is without impact on any other security policy of the TOE; it is exactly equivalent to the situation where the user decides just not to use the cryptographic co-processors.
End of note.

The IC Developer / Manufacturer must apply the organizational security policy “Limiting and Blocking the Loader Functionality (P.Lim_Block_Loader)” as specified below:

P.Lim_Block_Loader	**Limiting and Blocking the Loader Functionality** The composite manufacturer uses the Loader for loading of Security IC Embedded Software, user data of the Composite Product or IC Dedicated Support Software in charge of the IC Manufacturer. He limits the capability and blocks the availability of the Loader in order to protect stored data from disclosure and manipulation.
P.Ctrl_Loader	**Controlled usage to Loader Functionality** Authorized user controls the usage of the Loader functionality in order to protect stored and loaded user data from disclosure and manipulation.

4.3 Assumptions

The TOE assumptions on the operational environment are defined and described in PP [9] section 3.4.
The assumptions concern the phases where the TOE has left the chip manufacturer.
The support of cipher schemas requires an additional assumption.

A.Process-Sec-IC	**Protection during Packaging, Finishing and Personalization** It is assumed that security procedures are used after delivery of the TOE by the TOE Manufacturer up to delivery to the end-consumer to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorized use).
A.Resp-Appl	**Treatment of User data of the Composite TOE** All User data of the Composite TOE are owned by Security IC Embedded Software. Therefore, it must be assumed that security relevant User data of the Composite TOE (especially cryptographic keys) are treated by the Security IC Embedded Software as defined for its specific application context.

4.3.1 Augmented Assumptions

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

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

Note that here the routines which may compromise keys when being executed are part of the Smartcard Embedded Software. In contrast to this the threats T.Leak-Inherent and T.Leak-Forced address (i) the cryptographic routines which are part of the TOE. For details please refer to PP [9] section 3.4.

4.3.2 Note regarding CIPURSE™ CL

The CIPURSE™ CL as cryptographic functionality establishes a cryptographic secured communication channel between two identified entities. Depending on the implementation and usage, the CIPURSE™ CL can act either in the PICC or in the PCD role. In any case the secrets applied for establishing and usage of the secured channel must be treated in an appropriate way by both entities PICC and PCD.

This means that it is essential on user side that the critical data for establishing this cryptographic secured communication channel is generated and stored in an appropriate way and that integrity and confidentiality is maintained.
These preconditions are treated in the PP [9] section 3.1 claims 67 and 68, and are the reason for not placing an assumption here.

5 Security objectives (ASE_OBJ)

This section shows the subjects and objects where are relevant to the TOE.
A short overview is given in the following.
The user has the following standard high-level security goals related to the assets:

- SG1 maintain the integrity of user data (when being executed/processed and when being stored in the TOE´s memories)
- SG2 maintain the confidentiality of user data (when being executed/processed and when being stored in the TOE´s memories)
- SG3 maintain the correct operation of the security services provided by the TOE for the Security IC Embedded Software
- SG4 provision of random numbers.

5.1 Security objectives for the TOE

The security objectives of the TOE are defined and described in PP [9] section 4.1, 7.2.1, 7.3.1, 7.3.2, 7.4.1 and 7.4.2.

Table 7 Objectives for the TOE according to PP [9]

O.Phys-Manipulation	Protection against Physical Manipulation
O.Phys-Probing	Protection against Physical Probing
O.Malfunction	Protection against Malfunction
O.Leak-Inherent	Protection against Inherent Information Leakage
O.Leak-Forced	Protection against Forced Information Leakage
O.Abuse-Func	Protection against Abuse of Functionality
O.Identification	TOE Identification
O.RND	Random Numbers
O.Cap_Avail_Loader	Capability and availability of the Loader Valid only for the TOE derivatives delivered with activated Flash Loader.
O.Authentication	Authentication to external entities Valid only for the TOE derivatives delivered with activated Flash Loader
O.Ctrl_Auth_Loader	Access control and authenticity for the Loader - valid only for the TOE derivatives delivered with activated Flash Loader
O.TDES	Cryptographic service Triple-DES
O.AES	Cryptographic service AES

Note 4:
The objectives O.Cap_Avail_Loader, O. Authentication, O.Ctrl_Auth_Loader and O.Prot_TSF_Confidentiality apply only at TOE products coming with activated Flash Loader enabled for software or data download by the user. In other cases the Flash Loader is not available anymore and the user software or data download is completed. Depending on the capabilities of the user software these objectives may then reoccur as subject of the composite TOE.
End of note.

The TOE provides “Additional Specific Security Functionality (O.Add-Functions)” as specified below.

O.Add-Functions	**Additional Specific Security Functionality** The TOE must provide the following specific security functionality to the Smartcard Embedded Software: • Rivest-Shamir-Adleman Cryptography (RSA) • Elliptic Curve Cryptography (EC) • CIPURSE™ Cryptography

Note 5:
The cryptographic libraries RSA, EC, CCL and the Toolbox library are delivery options. Therefore the TOE may come with free combinations of or even without these libraries. In the case of coming without one or any combination of the cryptographic libraries RSA, EC, and CCL the TOE does not provide the Additional Specific Security Functionality Rivest-Shamir-Adleman Cryptography (RSA) and/or Elliptic Curve Cryptography (EC) and/or CIPURSE™ Cryptographic Library.
The Toolbox Library is no cryptographic library, provides no additional specific security functionality and thus the Toolbox library is not part of the TOE Security Functionality (TSF).
End of note.

Note 6:
This TOE can come with both crypto co-processors accessible, or with a blocked SCP or with a blocked Crypto2304T, or with both crypto co-processors blocked. The blocking depends on the customer demands prior to the production of the hardware. In case the SCP is blocked, no AES and DES computation supported by hardware is possible. In case the Crypto2304T is blocked, no RSA and EC computation – in any of the version stated in chapter 1.1 - supported by hardware is possible. No accessibility of the deselected cryptographic co-processors is without impact on any other security policy of the TOE; it is exactly equivalent to the situation where the user decides just not to use the cryptographic co-processors.
End of note.

Note 7:
In case the SCP is blocked, no AES, no DES computation supported by hardware is possible and thus the CIPURSE™ CL usage is not possible. This is regardless of the library version chosen.
End of note.

The TOE shall provide “Area based Memory Access Control (O.Mem-Access)” as specified below.

O.Mem Access	**Area based Memory Access Control** The TOE must provide the Smartcard Embedded Software with the capability to define restricted access memory areas. The TOE must then enforce the partitioning of such memory areas so that access of software to memory areas and privilege levels is controlled as required, for example, in a multi-application environment.

The additional functionality of a Loader as defined in the PP [9], section 7.3 requires to address the following objective, as defined in the document “PP0084: Interpretation” [9.1].

The TOE shall provide “Protection of the confidentiality of the TSF (O.Prot_TSF_Confidentiality)” as specified below:

O.Prot_TSF_Confidentiality	**Protection of the confidentiality of the TSF** The TOE must provide protection against disclosure of confidential operations of the Security IC (loader, memory management unit …) through the use of a dedicated code loaded on open samples.

If the CIPURSE™ CL is ordered the additional functionality address the following objectives as defined in this document:

O.Ctrl_Auth_CCL	**Authentication of entities** The CIPURSE™ CL must implement mutual authentication to establish a ready to use secure communication channel between two authenticated entities before any other communication between the two entities is applied.

O.Prot_Integrity	**Integrity protection** The CIPURSE™ CL must implement integrity protection functionality for the user data to be exchanged via the secure communication channel.

O.Prot_Confidentiality	**Confidentiality protection** The CIPURSE™ CL must protect the confidentiality of the user data to be exchanged via the secure communication channel if the user configures accordingly.

Table 8 Additional objectives due to TOE specific functions and augmentations

O.Add-Functions	Additional specific security functionality
O.Mem-Access	Area based Memory Access Control
O.Prot_TSF_Confidentiality	Protection of the confidentiality of the TSF
O.Ctrl_Auth_CCL	Authentication of entities
O.Prot_Integrity	Integrity protection
O.Prot_Confidentiality	Confidentiality protection

5.2 Security Objectives from PP for development and environment

The security objectives for the security IC embedded software development environment and the operational environment are defined in PP [9] section 4.2, 4.3, 7.2.1and 7.3.

For secure use of the CIPURSE™ CL it is essential that on user side the common secret is generated and stored in an appropriate way and that integrity and confidentiality of this user secret is maintained. These preconditions are treated in the PP [9] section 3.1 claims 67 and 68.

The operational environment of the TOE shall provide “Limitation of capability and blocking the Loader “OE.Lim_Block_Loader”, “External entities authentication of the TOE “OE.TOE_Auth" and “Secure communication and usage of the Loader “OE.Loader¬Usage” as specified below:

OE.Lim_Block_Loader	**Limitation of capability and blocking the Loader** The Composite Product Manufacturer will protect the Loader functionality against misuse, limit the capability of the Loader and terminate irreversibly the Loader after intended usage of the Loader.
OE.TOE_Auth	**Authentication to external entities** The operational environment shall support the authentication verification mechanism and know authentication reference data of the TOE.
OE.Loader_Usage	**Secure communication and usage of the Loader** The authorized user must support the trusted communication with the TOE by confidentiality protection and authenticity proof of the data to be loaded and fulfilling the access conditions required by the Loader.
OE.Process-Sec-IC	**Protection during composite product manufacturing** The protection during packaging, finishing and personalization includes also the personalization process (Flash Loader) and the personalization data (TOE software components) during Phase 4, Phase 5 and Phase 6.

Note 8:
The objectives OE.Lim_Block_Loader, OE.TOE_Auth, OE.Loader_Usage and OE.Process-Sec-IC for the development and operation environment apply only at TOE products coming with activated Flash Loader enabled for user data download. In other cases the Flash Loader is not available anymore and the user data download is completed. Depending on the capabilities of the user software this objective may then reoccur as subject of the composite TOE.
End of note.

5.3 Security Objectives for the Environment

Since the CIPURSE™ CL requires the presence of a common secret on both communication entities an additional objective for the environment is generated:
Following objectives are defined here.

OE.Resp-Appl	**Treatment of User data of the Composite TOE** Please refer to chapter 5.3.1 for clarification

Note 9:
The CIPURSE™ CL is a delivery option. In case the user has ordered the CIPURSE™ CL, the user is responsible to implement the CIPURSE™ CL into his user software. In addition, the user has to generate and treat the common secret in an appropriate way. The objective common secret is therefore similar to the OE.Resp-Appl. Anyhow, processes and treatment is exclusively subject of the user and his logistic processes.
Note 10:
If the user decides to use the integrity protection mode of the CIPURSE™ CL then the confidentiality of the user data to be transferred is solely subject of the user.
End of note.

Note 11:
The CIPURSE™ CL as cryptographic functionality establishes a cryptographic secured communication channel between two identified entities. Depending on the implementation and usage, the CIPURSE™ CL can act either in the PICC or in the PCD role. In any case the secrets applied for establishing and usage of the secured channel must be treated in an appropriate way by both entities PICC and PCD.
This means that it is essential on user side that the critical data for establishing this cryptographic secured communication channel is generated and stored in an appropriate way and that integrity and confidentiality is maintained.
These preconditions are treated in the PP [9] section 3.1 claims 67 and 68, and are the reason for not placing an objective for the environment here.
End of note.

OE.Secure_Delivery	**Secure Delivery of the TOE in case of deactivated flash loader** When the TOE is ordered with a disabled Flash Loader, it does not provide full transport protection. Therefore, technical and / or organisational security procedures (e.g. a custom mutual authentication mechanism or a security transport) should be put in place by the customer to secure the personalized TOE during delivery as required by the security needs of the loaded IC Embedded Software

Security objectives (ASE_OBJ)

The table below lists the security objectives.

Table 9 Security Objectives for the Environment according to the PP [9]

Phase 1	OE.Resp-Appl	Treatment of User data of the Composite TOE
Phase 5 – 6 optional Phase 4	OE.Process-Sec-IC	Protection during composite product manufacturing
Phase 5 – 6 optional Phase 4	OE.Lim_Block_Loader (1)	Limitation of capability and blocking the loader.
	OE.TOE_Auth (1)	Authentication to external entities
	OE.Loader_Usage (1)	Secure communication and usage of the Loader

(1) These objectives are only valid if the TOE is delivered with active Flash Loader.

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

Regarding the cryptographic services this objective of the environment has to be clarified.
By definition cipher or plain text data and cryptographic keys are user data of the Composite TOE. The Smartcard Embedded Software shall treat these data appropriately, use only proper secret keys (chosen from a large key space) as input for the cryptographic function of the TOE and use keys and functions appropriately in order to ensure the strength of cryptographic operation.
This means that keys are treated as confidential as soon as they are generated. The keys must be unique with a very high probability, as well as cryptographically strong. For example, it must be ensured that it is beyond practicality to derive the private key from a public key if asymmetric algorithms are used. If keys are imported into the TOE and/or derived from other keys, quality and confidentiality must be maintained. This implies that appropriate key management has to be realized in the environment.
Regarding the memory, software and firmware protection and the SFR and peripheral access rights handling these objectives of the environment has to be clarified. The treatment of user data of the Composite TOE is also required when a multi-application operating system is implemented as part of the Smartcard Embedded Software on the TOE. In this case the multi-application operating system should not disclose security relevant user data of one application to another application when it is processed or stored on the TOE.

5.3.2 Clarification of "Protection during Composite product Manufacturing (OE.Process-Sec-IC)"

The protection during packaging, finishing and personalization includes also the personalization process (Flash Loader) and the personalization data (TOE software components) during Phase 4, Phase 5 and Phase 6.

5.4 Security Objectives Rationale

The security objectives rationale of the TOE are defined and described in PP [9] section 4.4. For organizational security policy P.Add-Functions, OE.Plat-Appl and OE.Resp-Appl the rationale is given in the following description.

Table 10 Security Objectives Rationale

Assumption, Threat or Organizational Security Policy	Security Objective
A.Key-Function	OE.Resp-Appl
P.Add-Functions	O.Add-Functions O.Ctrl_Auth_CCL O.Prot_Integrity O.Prot_Confidentiality
P.Crypto-Service	O.TDES O.AES
P.Ctrl_Loader	O.Ctrl_Auth_Loader O.Authentication OE.Loader_Usage OE.TOE_Auth OE.Process-Sec-IC
P.Lim_Block_Loader	O.Cap_Avail_Loader OE.Lim_Block_Loader OE.Process-Sec-IC
T.Masquerade_TOE	O.Authentication OE.TOE_Auth OE.Process-Sec-IC OE.Secure_Delivery
T.Mem-Access	O.Mem-Access
T.Open_Samples__Diffusion	O.Prot_TSF_Confidentiality O.Leak-Inherent O.Leak-Forced OE.Process-Sec-IC OE.Secure_Delivery

5.5 P.Add-Functions

The justification related to the security objective “Additional Specific Security Functionality (O.Add-Functions)” is as follows: Since O.Add-Functions requires the TOE to implement exactly the same specific security functionality as required by P.Add-Functions; the organizational security policy is covered by the objective.

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

The policy P.Add-Functions includes the function CIPURSE™ Cryptographic Library (CCL). This functionality implements the objectives O.Ctrl_Auth_CCL, O.Prot_Integrity and O.Prot_Confidentiality. The CIPURSE™ Cryptographic Library (CCL) first prepares and establishes a secure communication channel between two authenticated entities. The mutual authentication means covers the objective O.Ctrl_Auth_CCL. The implemented means of session key generation and integrity protection prepares for the confidential and integrity protected exchange of user data between the two authenticated entities. Thus the objectives O.Prot_Integrity and O.Prot_Confidentiality are covered too.

5.6 A.Key-Function

Compared to the PP [9] a further clarification has been made for the security objective “Treatment of user data of the Composite TOE (OE.Resp-Appl)”: By definition cipher or plain text data and cryptographic keys are user data of the Composite TOE. So, the Smartcard Embedded Software will protect such data if required and use keys and functions appropriately in order to ensure the strength of cryptographic operation. Quality and confidentiality must be maintained for keys that are imported and/or derived from other keys. This implies that appropriate key management has to be realized in the environment. That is expressed by the assumption A.Key—Function which is covered from OE.Resp–Appl. These measures make sure that the assumption A.Resp-Appl is still covered by the security objective OE.Resp-Appl although additional functions are being supported according to P.Add-Functions.

5.7 T.Mem-Access

Compared to the PP [9] an enhancement regarding memory area protection has been established. The clear definition of privilege levels for operated software establishes the clear separation of different restricted memory areas for running the firmware – regardless of the version in use -, downloading and/or running the operating system and to establish a clear separation between different applications. Nevertheless, it is also possible to define a shared memory section where separated applications may exchange defined data. The privilege levels clearly define by using a hierarchical model the access right from one level to the other.
These measures ensure that the threat T.Mem-Access is clearly covered by the security objective O.Mem-Access.

5.8 P.Ctrl_Loader, P.Lim_Block_Loader, T.Masquerade_TOE and T.Open_Samples_Diffusion

The PP [9] section 7.3 considers the life cycle phases of the TOE also with the organizational policy P.Lim_Block_Loader and P.Ctlr_Loader as the TOE must be protected against unauthorized usage and control and against download of malicious software before, during and after the user downloads his software. This is formalized with the objectives O.Cap_Avail_Loader, O.Authentication and O.Ctrl_Auth_Loader requiring authentication of the TOE to external entities and a trusted communication channel. The O.Authentication implements a mutual authentication and involves the environment therefore.
And the policies cover also the environmental objective OE.Process-Sec-IC at user premises, as it must be regardless in which environment the Flash Loader is operated.
The implemented mutual authentication requires first to authenticate the TOE to the external user. This requires also knowledge by the user about the sequence of the protocol, interpretation of the transferred data and how to start the authentication of the TOE. This authentication mean counters the threat T.Masquerade_TOE as only the genuine TOE is able to identify itself correctly to the enabled user, which covers O.Authentication.
The second step of the mutual authentications mean implements the authentication of the user to the TOE. Only the user enabled to present the correct authentication data and knowing about the sequence, data interpretation and signaling of the authentication to the TOE is able to proceed further on.
This enforces OE.TOE_Auth, requiring the support of the verification mechanism and known authentication reference data by the operational environment.

It is normal business that products enabled for software download by the user are either on their way to the user or are already stored at user premises. At both situations it cannot be excluded that an attacker could manage to steal such products enabled for software downloads. The ability to download software on a chip without operating system and/or application is defined as open sample.
This situation generates the threat T.Open_Samples_Diffusion which is defined as follows:
The download of analysis software could enable an attacker to characterize the product and to construct an attack path out of the gained information.
This threat is countered by the Flash Loader by following rational:
As long as the Flash Loader is active, controlled usage to the Flash Loader functionality (P.Ctrl_Loader) is enforced which protects the TOE from achieving the status of being an open sample. And more, even the attacker could observe, meaning in the sense of measurements during, or induce faults during an authorized download, the Flash Loader protects the user data of the download by confidentiality and integrity protection means. The Flash Loader functionalities of mutual authentication, establishing a dedicated trusted communication channel, the encryption and integrity protection means cover the objectives O.Prot_TSF_Confidentiality, O.Leak-Inherent and O.Leak-Forced.
The policy P.Ctrl_Loader covers also the environmental objective OE.Process-Sec-IC at user premises, as it must be regardless in which environment the Flash Loader is operated.
Note regarding the Flash Loader:

The algorithm strength of the Flash Loader has not been cryptographic-analytically evaluated by the BSI, as Infineon has implemented a proprietary extension.
End of note.

The objective O.Ctrl_Auth_Loader Access control and authenticity for the Loader is covered by following rational:
The identification of the communication entities of the Flash Loader requires the presence of dedicated identification data for passing successfully the mutual authentication. This enforces the policy P.Ctrl_Loader comprising the aspect of the mutual authentication. After successful authentication the user is enabled to change the keys used for authentication and downloading the user data. This first user is defined as the administrator. The TOE can then further be operated for example by a service partner who is defined as being the Download Operator. The equal protecting means as for the Administrator apply here again but due to the key change different roles are established. The Download Operator downloads then the encrypted user data with the Flash Loader into the defined and accessible SOLID FLASH™ NVM area. This area is access protected by the MMU.
The policy O.Ctrl_Auth_Loader covers also the environmental objective OE.Process-Sec-IC at user premises, as it must be regardless in which environment the Flash Loader is operated.

This objective O.Leak-Inherent is covered with following rational:
This false download threat is countered with the mutual authentication mean as only the correct identified user is able to download the user intended software and data. Since it is not practical for an attacker to authenticate correctly a threatening download of attack software is countered. By that possible confidential user data already stored on the TOE remain protected from disclosure by this method.
If the user is the attacker, or does not follow the user guidance, or bad designed user software implements weaknesses, the user data remain protected anyway, since even after passing the mutual authentication of the loader the download is conducted encrypted only. And even more, a different encryption is applied to store the data in the SOLID FLASH™ NVM. Since also the number of Flash Loader trials is limited even comprehensive side channel analysis would not leave sufficient information to conduct a successful attack.
The implemented authentication means cover also the environmental objective OE.Process-Sec-IC at user premises, as it must be regardless in which environment the Flash Loader is operated.

This objective O.Leak-Forced is covered with following rational:
Another method to gain information is to force information leakage of confidential data processed in the TOE. Such forcing requires malfunction or physical manipulation. Inducing errors of any kind during data processing will be discovered by the Integrity Guard with high probability which leads to a security reset. Failures induced during the mutual authentication or encrypted download process of the Flash Loader will also be discovered as the perturbation of the sequences leads to fail of the process with trial counter decrement or a fail of the integrity check of the downloaded data will occur. It would anyway not be practical to induce targeted errors as any process managing data is protected due to the permanent and differently data encryption and integrity protection on the TOE.
Physical static manipulation requires the presence of a worthwhile target. This is on one hand hard to identify and would require intensive reverse engineering due to the topological means such as synthesis of the TOE and other means. But, on the other hand, if we assume than an attacker could identify such spot, and the physical preparation was successful too and thus it was possible to probe the targeted signal, then, even assumed the attacker could analyze the traffic on the signal, the results would be worthless, since the signal data is encrypted or masked. Thus, the data remains confidentiality protected even outside the TOE, since the analyst neither has neither the encryption algorithm nor the key.
The Integrity Guard concept and encryption of memory contents are always present and cover also the environmental objective OE.Process-Sec-IC at user premises, as it must be regardless in which environment the Flash Loader is operated in.

All requirements are fulfilled by the Flash Loader due to the strong mutual authentication means enabling only the authorized user for the download, due to the download of encrypted data only and due to the final

locking command to be applied by the user before delivery. Therefore, the operational environment objectives OE. Lim_Block_Loader, OE.TOE_Auth and OE.Loader_Usage obligate the composite manufacturer to protect the authentication data (e.g. keys) against misuse and limit the capability of the Loader.
In addition, the user guidance implements the obligation to permanently disable the Flash Loader prior delivery to the end-user.

Package 1 defines the final locking of the Flash Loader prior delivery to the end-user and the usage in secure environment. Package 2 defines that the Flash Loader can be used also in insecure environment. By claiming both packages, the user has the choice to apply the active Flash Loader either in insecure or in secure environment and achieves by that maximum flexibility. Anyhow, the user is obligated to lock the Flash Loader prior delivery to the end-user in both cases. This is an obligation implemented by user guidance.
The Flash Loader provides the required functionality to be applied by the composite manufacturer for covering these objectives.

The objectives O.Cap_Avail_Loader, O.Authentication, O.Loader_Usage and O.Prot_TSF_Confidentiality and the organizational policies P.Lim_Block_Loader and P.Ctlr_Loader as discussed in the PP [9] section 7.2 and 7.3 apply only at TOE products at the life cycle phase delivery, if these products come with activated Flash Loader enabled for software or data download by the user. In other cases, the Flash Loader is not available anymore and the user software or data download is completed. Depending on the capabilities of the user software these objectives may then reoccur as subject of the composite TOE.
In case the flash loader is disabled the user is responsible for secure delivery of the TOE expressed by OE.Secure_Delivery.

5.9 P.Crypto-Service

The PP [9] includes the organizational security policy P.Crypto-Service Cryptographic services of the TOE in a different extend as it formalizes the objectives O.TDES and O.AES.
For the objective O.TDES a concrete standard reference (NIST) with operational modes is given the implementation must follow and also the cryptographic key destruction is regulated. The implementation complies to the given security functional requirements and the objective O.TDES is met.
For the objective O.AES a concrete standard reference (NIST) with a selection of key lengths is given the implementation must follow and also the cryptographic key destruction is regulated. The implementation complies to the given security functional requirements and the objective O.AES is met.
The justification of the additional policy and the additional assumption show that they do not contradict to the rationale already given in the Protection Profile for the assumptions, policy and threats defined there.

6 Extended Component Definition (ASE_ECD)

There are following extended components defined and described for the TOE:

- the family FAU_SAS at the class FAU Security Audit
- the component FPT_TST.2 at the class FPT Protection of the TSF

The extended component FAU_SAS, FDP_SDC is defined and described in PP [9] section 5.

The component FPT_TST.2 is defined in the following.

6.1 Component Subset TOE security testing (FPT_TST.2)

The security is strongly dependent on the correct operation of the security functions. Therefore, the TOE shall support that particular security functions or mechanisms are tested in the operational phase (Phase 7). The tests can be initiated by the Smartcard Embedded Software and/or by the TOE or is done automatically and continuously.
Part 2 of the Common Criteria provides the security functional component “TSF testing (FPT_TST.1)”. The component FPT_TST.1 provides the ability to test the TSF’s correct operation.
For the user it is important to know which security functions or mechanisms can be tested. The functional component FPT_TST.1 does not mandate to explicitly specify the security functions being tested. In addition, FPT_TST.1 requires verification of the integrity of TSF data and of the stored TSF executable code which might violate the security policy. Therefore, the functional component **”Subset TOE security testing (FPT_TST.2)”** of the family TSF self-test has been newly created. This component allows that particular parts of the security mechanisms and functions provided by the TOE are tested.

6.1.1 Definition of FPT_TST.2

The functional component “Subset TOE security testing (FPT_TST.2)” has been newly created (Common Criteria Part 2 extended). This component allows that particular parts of the security mechanisms and functions provided by the TOE can be tested after TOE Delivery or are tested automatically and continuously during normal operation transparent for the user.
This security functional component is used instead of the functional component FPT_TST.1 from Common Criteria Part 2. For the user it is important to know which security functions or mechanisms can be tested. The functional component FPT_TST.1 does not mandate to explicitly specify the security functions being tested. In addition, FPT_TST.1 requires verifying the integrity of TSF data and stored TSF executable code which might violate the security policy.
The functional component “Subset TOE testing (FPT_TST.2)” is specified as follows (Common Criteria Part 2 extended).

6.1.2 TSF self-test (FPT_TST)

Family Behavior The Family Behavior is defined in [CCBook2] section 15.17.1.

Component levelling

FPT_TST.1: The component FPT_TST.1 is defined in [CCBook2] section 15.17.5.

FPT_TST.2: Subset TOE security testing, provides the ability to test the correct operation of particular security functions or mechanisms. These tests may be performed at start-up, periodically, at the request of the authorized user, or when other conditions are met. It also provides the ability to verify the integrity of TSF data and executable code.

Management: FPT_TST.2

The following actions could be considered for the management functions in FMT:

- management of the conditions under which subset TSF self testing occurs, such as during initial start-up, regular interval or under specified conditions
- management of the time of the interval appropriate.

Audit: FPT_TST.2

There are no auditable events foreseen.

FPT_TST.2	**Subset TOE testing**
Hierarchical to:	No other components.
Dependencies:	No dependencies
FPT_TST.2.1:	The TSF shall run a suite of selftests [selection: during initial start-up, periodically during normal operation, at the request of the authorized user, and/or at the conditions [assignment: conditions under which selftest should occur]] to demonstrate the correct operation of [assignment: functions and/or mechanisms].

7 Security Requirements (ASE_REQ)

For this section the PP [9] section 6 can be applied completely.
For the CC operations the following convention is used:

- CC operations which have been already completed in [PP0084] or [AIS 31] are typeset without underline.
- CC (nested) iteration operations are started by a slash "/" symbol, followed by an iteration identifier text. Iterations may be recursively nested.
- CC operations which are completed in this ST are typed in italic font.

7.1 TOE Security Functional Requirements

The security functional requirements (SFR) for the TOE are defined and described in the PP [9] section 6.1 and in the following description.

7.1.1 Extended Component FAU_SAS.1

7.1.1.1 FAU_SAS

The [PP0084] defines additional security functional requirements with the family FAU_SAS of the Class FAU (Security Audit). This family describes the functional requirements for the storage of audit data. It has a more general approach than FAU_GEN, because it does not necessarily require the data to be generated by the TOE itself and because it does not give specific details of the content of the audit records.
The TOE shall meet the requirement "Audit storage (FAU_SAS.1)" as specified below (Common Criteria Part 2 extended).

FAU_SAS.1	Audit Storage
Hierarchical to:	No dependencies
Dependencies:	No dependencies.
FAU_SAS.1.1	The TSF shall provide the test process *before TOE Delivery* with the capability to store *the Initialization Data (GCIM) and/or Pre-personalization Data and/or supplements of the Security IC Embedded Software* in the *not changeable configuration page area and non-volatile memory.*

7.1.2 Memory access control

Usage of multiple applications in one Smartcard often requires code and data separation in order to prevent that one application can access code and/or data of another application. For this reason the TOE provides Area based Memory Access Control. The underlying memory management unit (MMU) is documented in section 4 in the hardware reference manual HRM [1].
The security service being provided is described in the Security Function Policy (SFP) **Memory Access Control Policy**. The security functional requirement **"Subset access control (FDP_ACC.1)"** requires that this policy is in place and defines the scope were it applies. The security functional requirement **"Security attribute based access control (FDP_ACF.1)"** defines security attribute usage and characteristics of policies. It describes the rules for the function that implements the Security Function Policy (SFP) as identified in FDP_ACC.1. The decision whether an access is permitted or not is taken based upon attributes allocated to the software. The Smartcard Embedded Software defines the attributes and memory areas. The corresponding permission control information is evaluated "on-the-fly" by the hardware so that access is granted/effective or denied/inoperable.

The security functional requirement **"Static attribute initialization (FMT_MSA.3)"** ensures that the default values of security attributes are appropriately either permissive or restrictive in nature. Alternative values can be specified by any subject provided that the **Memory Access Control Policy** allows that. This is described by the security functional requirement **"Management of security attributes (FMT_MSA.1)"**. The attributes are determined during TOE manufacturing (FMT_MSA.3) or set at run-time (FMT_MSA.1).
From TOE's point of view the different roles in the Smartcard Embedded Software can be distinguished according to the memory based access control. However the definition of the roles belongs to the user software.
The following Security Function Policy (SFP) **Memory Access Control Policy** is defined for the requirement "Security attribute based access control (FDP_ACF.1)":

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

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

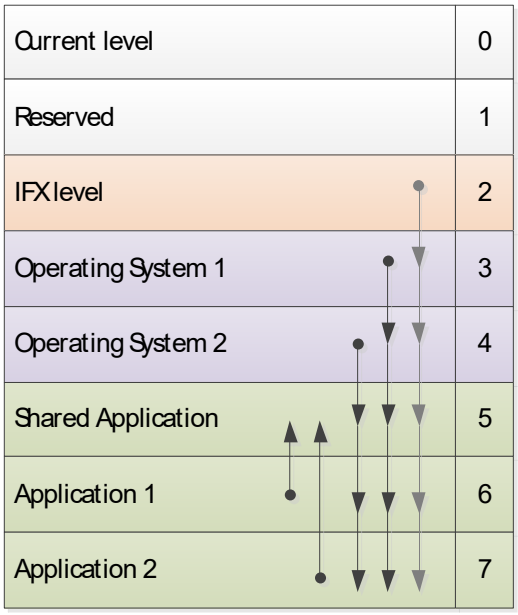


Figure 2 Privilege levels of the TOE

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

FDP_ACC.1	**Subset access control**
Hierarchical to:	No other components.
Dependencies:	FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1	The TSF shall enforce the *Memory Access Control Policy* on *all subjects (software running at the defined and assigned privilege levels), all objects (data including code stored in memories) and all the operations defined in the Memory Access Control Policy, i.e. privilege levels*.

Security Requirements (ASE_REQ)

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

FDP_ACF.1	**Security attribute based access control**
Hierarchical to:	No other components.
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
FDP_ACF.1.1	The TSF shall enforce the *Memory Access Control Policy* to objects based on the following: *Subject*: *- software running at the IFX, OS1 and OS2 privilege levels required to securely operate the chip. This includes also privilege levels running interrupt routines.* *- software running at the privilege levels containing the application software* *Object*: *- data including code stored in memories* *Attributes*: *- the memory area where the access is performed to and/or* *- the operation to be performed.*
FDP_ACF.1.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *evaluate the corresponding permission control information of the relevant memory range before and during the access so that accesses to be denied cannot be utilized by the subject attempting to perform the operation.*
FDP_ACF.1.3	The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: *none.*
FDP_ACF.1.4	The TSF shall explicitly deny access of subjects to objects based on the *following additional rules: none.*

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

FMT_MSA.3	**Static attribute initialisation**
Hierarchical to:	No other components.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
FMT_MSA.3.1	The TSF shall enforce the *Memory Access Control Policy* to provide *well defined*[1] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow *any subject, provided that the Memory Access Control Policy is enforced and the necessary access is therefore allowed*[2], to specify alternative initial values to override the default values when an object or information is created.

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

FMT_MSA.1	**Management of security attributes**
Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] FMT_SMF.1 Specification of management functions FMT_SMR.1 Security roles
FMT_MSA.1.1	The TSF shall enforce the *Memory Access Control Policy* to restrict the ability to *change default, modify or delete* the security attributes *permission control information to the software running on the privilege levels.*

The TOE shall meet the requirement "Specification of management functions (FMT_SMF.1)" as specified below:

FMT_SMF.1	**Specification of management functions**
Hierarchical to:	No other components
Dependencies:	No dependencies
FMT_SMF.1.1	The TSF shall be capable of performing the following security management functions*:* *access the configuration registers of the MMU.*

7.1.3 Support of Cipher Schemes

The following additional specific security functionality is implemented in the TOE:
FCS_COP.1 Cryptographic operation requires a cryptographic operation to be performed in accordance with a specified algorithm and with a cryptographic key of specified sizes. The specified algorithm and cryptographic key sizes can be based on an assigned standard; dependencies are discussed in chapter 7.4.1.4.

The following additional specific security functionality is implemented in the TOE:

- Rivest-Shamir-Adleman (RSA)[3]
- Elliptic Curve Cryptography (EC)[3]
- Advanced Encryption Standard (AES)

[1] The static definition of the access rules is documented in the hardware reference manual as listed in chapter 1.1

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

[3] For the case the TOE comes without RSA and/or EC library, the TOE provides basic HW-related routines for RSA and/or EC calculations. For a secure library implementation the user has to implement additional countermeasures himself.

- Triple Data Encryption Standard (TDES)
- CIPURSE™ Cryptographic Library (CCL)

The RSA cryptographic library is offered in two parts: The 2k part of the RSA library can be used for key lengths of up to 2048 bits and the 4k part of the RSA library can be used for key lengths of up to 4096 bits.

Note 12:
This TOE can come with both crypto co-processors accessible, or with a blocked SCP or with a blocked Crypto2304T, or with both crypto co-processors blocked. The blocking depends on the customer demands prior to the production of the hardware. In case the SCP is blocked, no AES and DES computation, as well as CCL operation supported by hardware is possible. In case the Crypto2304T is blocked, no RSA and EC computation supported by hardware is possible. No accessibility of the deselected cryptographic co-processors is without impact on any other security policy of the TOE; it is exactly equivalent to the situation where the user decides just not to use the cryptographic co-processors.
End of note.

7.1.3.1 Preface regarding Security Level related to Cryptography

Implemented cryptography, regardless whether proprietary or following public standards is subject of ageing due to the evolution and progress of technology, mathematics and analysis methods. For that reason, guarantees for the absence of exploitable properties cannot be given - by anybody. But, the risk for the absence of such properties is severely reduced if the implementation is based on a long time published and publicly discussed standard algorithm.
If the vendor implements a proprietary algorithm, then the BSI analyses this algorithm in the course and time frame of the certification process. It is obvious that this local, one party and short timed analysis cannot replace a long time publicly discussed standard algorithm. Possible risk mitigation, if not an industrial secret, is the publication of the proprietary standard and starting the public discussion. This transparency could provide on one hand some mitigation of risk; on the other hand, the algorithm is then directly a valuable analyst target. In that case, also no one can say, when such publication is sufficiently discussed; and again guarantees for the life time being in the field can no one give.
As of today, using public standards is often not possible due to compromises with regard to acceptable computing time and available energy. For that reason it is up to the user to decide. The SFR claims assign whether an algorithm or proprietary or following a standard or implementing a standard with deviations.

The BSI states that the strength of the cryptographic algorithms is not rated in the course of the product certification (see [25] Section 9, Para.4, Clause 2). But cryptographic functionalities with a security level of lower than 120 bits can no longer be regarded as secure without considering the application context. Therefore, for these functions it shall be checked whether the related cryptographic operations are appropriate for the intended system. The assignment to the 120 Bit level by BSI can be read in the certification report issued by BSI.
Some further hints and guidelines can be derived from the Technische Richtlinie BSI TR-02102, www.bsi.bund.de.

7.1.3.2 Random Number Generation

7.1.3.2.1 True Random Number Generation, meeting AIS31 PTG.2

FCS_RNG.1/TRNG	Random Number Generation
Hierarchical to:	No other components
Dependencies:	No dependencies

FCS_RNG.1/TRNG	Random numbers generation Class PTG.2 according to [13]
FCS_RNG.1.1/TRNG	The TSF shall provide a *physical* random number generator that implements:
PTG.2.1	*A total failure test detects a total failure of entropy source immediately when the RNG has started. When a total failure is detected, no random numbers will be output.*
PTG.2.2	*If a total failure of the entropy source occurs while the RNG is being operated, the RNG prevents the output of any internal random number that depends on some raw random numbers that have been generated after the total failure of the entropy source.*
PTG.2.3	*The online test shall detect non-tolerable statistical defects of the raw random number sequence (i) immediately when the RNG has started, and (ii) while the RNG is being operated. The TSF must not output any random numbers before the power-up online test has finished successfully or when a defect has been detected.*
PTG.2.4	*The online test procedure shall be effective to detect non-tolerable weaknesses of the random numbers soon.*
PTG.2.5	*The online test procedure checks the quality of the raw random number sequence. It is triggered continuously. The online test is suitable for detecting non-tolerable statistical defects of the statistical properties of the raw random numbers within an acceptable period of time.*
FCS_RNG.1.2/TRNG	The TSF shall provide *numbers in the format 8- or 16-bit* that meet
PTG.2.6	*Test procedure A, as defined in [13] does not distinguish the internal random numbers from output sequences of an ideal RNG.*
PTG.2.7	*The average Shannon entropy per internal random bit exceeds 0.997.*

7.1.3.2.2 Hybrid Random Number Generation, meeting AIS31 PTG.3

FCS_RNG.1/HRNG	**Random Number Generation**
Hierarchical to:	No other components
Dependencies:	No dependencies
FCS_RNG.1/HRNG	Random numbers generation **Class PTG.3** according to [13]
FCS_RNG.1.1/HRNG	The TSF shall provide a *hybrid physical* random number generator that implements:
PTG.3.1	*A total failure test detects a total failure of entropy source immediately when the RNG has started. When a total failure has been detected no random numbers will be output.*
PTG.3.2	*If a total failure of the entropy source occurs while the RNG is being operated, the RNG prevents the output of any internal random number that depends on some raw random numbers that have been generated after the total failure of the entropy source.*
PTG.3.3	*The online test shall detect non-tolerable statistical defects of the raw random number sequence (i) immediately when the RNG has started, and (ii) while the RNG is being operated. The TSF must not output any random numbers before the power-up online test and the seeding of the DRG.3 post-processing algorithm have been finished successfully or when a defect has been detected.*

PTG.3.4	*The online test procedure shall be effective to detect non-tolerable weaknesses of the random numbers soon.*
PTG.3.5	*The online test procedure checks the raw random number sequence. It is triggered continuously. The online test is suitable for detecting non-tolerable statistical defects of the statistical properties of the raw random numbers within an acceptable period of time.* *Note:* *Continuously means that the raw random bits are scanned continuously.* *The algorithmic post-processing belongs to Class DRG.3 with cryptographic state transition function and cryptographic output function. The output data rate of the post-processing algorithm shall not exceed its input data rate.* *End of note.*
PTG.3.6	*The algorithmic post-processing algorithm belongs to Class DRG.3 with cryptographic state transition function and cryptographic output function, and the output data rate of the post-processing algorithm shall not exceed its input data rate.*
FCS_RNG.1.2/HRNG	The TSF shall provide *numbers in the format 8- or 16-bit* that meet
PTG.3.7	*Statistical test suites cannot practically distinguish the internal random numbers from output sequences of an ideal RNG. The internal random numbers must pass test procedure A.*
PTG.3.8	*The internal random numbers shall use the PTRNG of class PTG.2 as random source for the post processing.* *Note:* *The internal random numbers produced by the employed PTG.2-conform PTRNG are adaptively compressed raw bits, where the compression rate is controlled by a so-called entropy estimator. The concept ensures that the random numbers provided by the PTRNG have high entropy, i.e., each delivered random byte will have more the 7.976 bit of entropy. In addition, the PTRNG produced random numbers have been tested against test procedures A and B under varying environment conditions.* *End of note.*

7.1.3.2.3 Deterministic Random Number Generation (DRNG) AIS31 DRG.3

FCS_RNG.1/DRNG	**Random Number Generation**
Hierarchical to:	No other components
Dependencies:	No dependencies
FCS_RNG.1/DRNG	Random numbers generation **Class DRG.3** according to [13]
FCS_RNG.1.1/DRNG	The TSF shall provide a *deterministic* random number generator that implements:
DRG.3.1	*If initialized with a random seed using a PTRNG of class PTG.2 as random source the internal state of the RNG shall have at least 100 bit of entropy.* *Note:* *Furthermore, the length of the internal state shall have at least 200 bit. (For the DRG.3 under consideration, the internal state has 351 bit.). The seed is provided by a certified PTG.2 physical TRNG with guaranteed 7,976 bit of entropy per byte.* *End of note.*

DRG.3.2	*The RNG provides forward secrecy.*
DRG.3.3	*The RNG provides backward secrecy even if the current internal state is known.*
FCS_RNG.1.2/DRNG	The TSF shall provide *numbers in the format 8- or 16-bit* that meet
DRG.3.4	*The RNG, initialized with a random seed, where the seed has at least 100 bit of entropy and is derived by a PTG.2 certified PTRNG. The RNG generates output for which any consecutive 2^{34} strings of bit length 128 are mutually different with a probability that is greater than $1 – 2^{(-16)}$.*
DRG.3.5	*Statistical test suites cannot practically distinguish the random numbers from the output sequences of an ideal RNG. The random numbers must pass test procedure A and the U.S. National Institute of Standards and Technology (NIST) test suite for RNGs used for cryptographic purposes [41] containing following 16 tests:* *Frequency (Monobit) Test, Frequency Test within a Block, Runs Tests, Test for the Longest-Run-of-Ones in a Block, Binary Matrix Rank Test, Discrete Fourier Transform (Spectral) Test, Non-overlapping (Aperiodic) Template Matching Test, Overlapping (Periodic) Template Matching Test, Maurer´s "Universal Statistical" Test, Liner Complexity Test, Serial Test, Approximate Entropy Test, Cumulative Sums (Cusums) Test, Random Excursions Test and Random Excursions Variant Test.*

7.1.3.2.4 Deterministic Random Number Generation (DRNG) AIS31 DRG.2

This additional operation mode is named Key Stream Generation (KSG), which is a stream cipher generation. It is conformant to DRG.2 and implements therefore forward and additional backward secrecy.

FCS_RNG.1/KSG	**Random Number Generation**
Hierarchical to:	No other components
Dependencies:	No dependencies
FCS_RNG.1/KSG	Random numbers generation **Class DRG.2** according to [13]
FCS_RNG.1.1/KSG	The TSF shall provide a *deterministic* random number generator that implements:
DRG.2.1	*If initialized with a random seed using a PTRNG of class PTG.2 as random source, the applied seed shall have at least 100 bits of entropy, the internal state of the RNG shall have at least the size of 200 bit – in this case the size of the internal state amounts to 351 bit, has the work factor for breaking the algorithm of 2^{127} due to the restriction on the maximum amount of keystream computed from a given seed, required guess work amounts to 2^{127} as well.*
DRG.2.2	*The RNG provides forward secrecy.*
DRG.2.3	*The RNG provides backward secrecy.*
FCS_RNG.1.2/KSG	The TSF shall provide numbers in the format *8- or 16-bit* that meet
DRG.2.4	*The RNG, initialized with a random seed of length at least 100 bit delivered by an PTRNG of the class PTG.2, generates output for which any consecutive 2^{34} strings of the length 128 bits are mutually different with probability greater than $1\text{-}2^{(-16)}$.*

DRG.2.5	*Statistical test suites cannot practically distinguish the random numbers from the output sequences of an ideal RNG. The random numbers must pass test procedure A.* *Note:* *The random numbers have been shown to fulfill all statistical tests of the AIS 20/31 statistical tests of procedure A. The random numbers are in the format 8- or 16 Bit.* *End of Note.*

7.1.3.3 Cryptography by the Symmetric Cryptographic Coprocessor SCP

7.1.3.3.1 Triple-DES Operation

FCS_COP.1/TDES	Cryptographic operation -TDES
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation, or FCS_CKM.5 Cryptographic key derivation] FCS_CKM.6 Timing and event of cryptographic key destruction
FCS_COP.1/TDES	The TSF shall perform *encryption and decryption* in accordance with a specified cryptographic algorithm *TDES in* • *the Electronic Codebook Mode (ECB)* • *the Cipher Block Chaining Mode (CBC)* • *the Cipher Block Chaining Message Authentication Code (CBC-MAC)* • *the Cipher Block Chaining Message Authentication Code Encrypt Last Block (CBC-MAC-ELB)* • *the Blinding Mode (BLD)* • *the Recrypt Mode* and cryptographic key sizes *112 and 168 bit* that meet the following standards:

	• *ECB, CBC:*
	  ○ *National Institute of Standards and Technology (NIST) 800-67 [20]*
	  ○ *National Institute of Standards and Technology (NIST) SP 800-38A [21]*
	  ○ *ISO/IEC 18033-3 [30]*
	• *CBC_MAC*
	  ○ *National Institute of Standards and Technology (NIST) 800-67 [20]*
	  ○ *National Institute of Standards and Technology (NIST) SP 800-38A [21]*
	  ○ *ISO/IEC 18033-3 [30]*
	  ○ *ISO/IEC 9797-1 Mac Algorithm 1 [32]*
	• *CBC-MAC-ELB:*
	  ○ *National Institute of Standards and Technology (NIST) 800-67 [20]*
	  ○ *National Institute of Standards and Technology (NIST) SP 800-38A [21]*
	  ○ *ISO/IEC 18033-3 [30]*
	  ○ *ISO/IEC 9797-1 Mac Algorithm 2 [32]*
	• *BLD, Recrypt Mode*
	Proprietary, description given in the hardware reference manual HRM [1]

Notes for FCS_COP.1/TDES:

1. The proprietary BLD and Recrypt operation modes are described in the hardware reference manual HRM [1] while the implementations of the other modes follow the referenced standards. Also the BLD is compliant to the referenced standards but is operated in a masked way. The proprietary modes offer increased protection against failure and side channel analysis.
2. Using the TDES algorithm with three keys of which two keys equal is a so called two key triple DES operation. This operation can be configured and managed by the user.

   End of note.

FCS_CKM.6/TDES	**Timing and event of cryptographic key destruction – TDES**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation, or FCS_CKM.5 Cryptographic key derivation]
FCS_CKM.6.1/TDES	The TSF shall destroy *cryptographic keys* when *the key register interfaces are overwritten or a software reset of the SCP is triggered*
FCS_CKM.6.2/TDES	The TSF shall destroy cryptographic keys and keying material specified by FCS_CKM.6.1 in accordance with a specified cryptographic key destruction method *overwriting or zeroing* that meets the following: *None*

Note for FCS_CKM.6/TDES:
The key destruction can be done by overwriting the key register interfaces or by software reset of the SCP

which provides immediate zeroing of all SCP key registers.
End of note.

7.1.3.3.2 AES Operation

FCS_COP.1/AES	Cryptographic operation -AES
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation, or FCS_CKM.5 Cryptographic key derivation] FCS_CKM.6 Timing and event of cryptographic key destruction
FCS_COP.1.1/AES	The TSF shall perform *decryption and encryption* in accordance with a specified cryptographic algorithm *AES in* • *the Electronic Codebook Mode (ECB)* • *the Cipher Block Chaining Mode (CBC)* • *the Cipher Block Chaining Message Authentication Code (CBC-MAC)* • *the Cipher Block Chaining Message Authentication Code Encrypt Last Block (CBC-MAC-ELB)* • *the Blinding Mode (BLD)* • *the Recrypt Mode* and cryptographic key sizes *128 bit or 192 bit or 256 bit* that meet the following standards: • *ECB, CBC:* ○ *Federal Information Processing Standards Publication 197 [31]* ○ *National Institute of Standards and Technology (NIST) SP 800-38A [21]* ○ *ISO/IEC 18033-3 [30]* • *CBC_MAC* ○ *Federal Information Processing Standards Publication 197 [31]* ○ *National Institute of Standards and Technology (NIST) SP 800-38A [21]* ○ *ISO/IEC 18033-3 [30]* ○ *ISO/IEC 9797-1 Mac Algorithm 1[32]* • *CBC-MAC-ELB:* ○ *Federal Information Processing Standards Publication 197 [31]* ○ *National Institute of Standards and Technology (NIST) SP 800-38A [21]* ○ *ISO/IEC 18033-3 [30]* ○ *ISO/IEC 9797-1 Mac Algorithm 2 respectively [32]* • *BLD, Recrypt Mode* *Proprietary, description given in the hardware reference manual HRM [1]*

Note for FCS_COP.1/AES:
The proprietary BLD and Recrypt operation modes are proprietary and described in the hardware reference manual HRM [1] while the implementations of the other modes follow the referenced standards. Also the BLD is compliant to the referenced standards but is operated in a masked way. The proprietary modes offer increased protection against failure and side channel analysis.
End of note.

FCS_CKM.6/AES	**Timing and event of cryptographic key destruction – AES**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation, or FCS_CKM.5 Cryptographic key derivation]
FCS_CKM.6.1/AES	The TSF shall destroy *cryptographic keys* when *the key register interfaces are overwritten or a software reset of the SCP is triggered*
FCS_CKM.6.2/AES	The TSF shall destroy cryptographic keys and keying material specified by FCS_CKM.6.1 in accordance with a specified cryptographic key destruction method *overwriting or zeroing* that meets the following: *None*

Note for FCS_CKM.6/AES:
The key destruction can be done by overwriting the key register interfaces or by software reset of the SCP which provides immediate zeroing of all SCP key registers.
End of Note.

Requirement “Subset TOE testing (FPT_TST.2)” as specified below:

The security is strongly dependent on the correct operation of the security functions. Therefore, the TOE shall support that particular security functions or mechanisms are tested in the operational phase (Phase 7). The tests can be initiated by the Smartcard Embedded Software and/or by the TOE.
The TOE shall meet the requirement “Subset TOE testing (FPT_TST.2)” as specified below (Common Criteria Part 2 extended).

FPT_TST.2	**Subset TOE testing**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_TST.2.1	The TSF shall run a suite of self-tests *at the request of the authorized user* to demonstrate the correct operation *of the alarm lines and/or following environmental sensor mechanisms:* • I2 Shield • Light – Light detection alarm • Temperature alarm • FSE – Internal Frequency Sensor alarm • WDT - Watch Dog Timer related alarms • SW – Software triggered alarm

7.1.3.4 Cryptography by CIPURSE™ Cryptographic Library

The CIPURSE™ CL is a delivery option. If the CCL is on board the belonging security functional requirements are fulfilled else the functionality is not on board and the security functional requirements are not covered.

7.1.3.4.1 CIPURSE™ CL Cryptographic Functions

The CCL shall meet the security functional requirement FCS_COP.1/CCL as specified below.

FCS_COP.1/CCL	**Cryptographic Operation CCL Trusted Channel**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation, or FCS_CKM.5 Cryptographic key derivation] FCS_CKM.6 Timing and event of cryptographic key destruction
FCS_COP.1.1/CCL	The TSF shall perform • *an authentication and cryptographic protected protocol* in accordance with a specified cryptographic algorithm • *CIPURSE™ V2 Cryptographic Protocol* and cryptographic key sizes *of 128 Bit* that meet the following: • *Federal Information Processing Standards Publication 197 [31]* • *NIST Special Publication SP 800-38A, chapter 6.1 AES in ECB mode [21]* • *CIPURSE™ V2 Cryptographic Protocol [35-1] chapter 5.2 Session key Derivation* • *CIPURSE™ V2 Cryptographic Protocol [35-1] chapter 6.2 Key Derivation for the first frame* • *CIPURSE™ V2 Cryptographic Protocol [35-1] chapter 6.3 Integrity Protection* • *CIPURSE™ V2 Cryptographic Protocol [35-1] chapter 6.4 Confidential Communication* • *CIPURSE™ V2 Cryptographic Protocol Errata and Precision list [35-2] chapters P.2*

Security Requirements (ASE_REQ)

The CCL shall meet the security functional requirement FCS_CKM.1/CCL as specified below.

FCS_CKM.1/CCL	**Cryptographic key generation**
Hierarchical to:	No other components.
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_CKM.5 Cryptographic key derivation, or FCS_COP.1 Cryptographic operation] [FCS_RBG.1 Random bit generation, or FCS_RNG.1 Generation of random numbers] FCS_CKM.6 Timing and event of cryptographic key destruction
FCS_CKM.1.1/CCL	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm: • *In the mode confidential communication:* *session key derivation and key derivation for the exchange protocol* • *In the mode integrity protection:* *session key derivation* and specified cryptographic key sizes of *128 bits* that meet the following*:* • *CIPURSE™ V2 Cryptographic Protocol [35-1] chapters 5.2 and 6.2*

The CCL shall meet the security functional requirement FCS_CKM.1/CCL as specified below.

FCS_CKM.6/CCL	**Cryptographic key destruction**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation, or FCS_CKM.5 Cryptographic key derivation]
FCS_CKM.6.1/CCL	The TSF shall destroy cryptographic keys when the CCL is entered or exited.
FCS_CKM.6.2/CCL	The TSF shall destroy cryptographic keys and keying material specified by FCS_CKM.6.1 in accordance with a specified cryptographic key destruction method *wiping with random values* that meets the following: *None*

Note for FCS_CKM.6/CCL:
The key destruction is triggered by the SCP at every entry and exit. The key storage in the SCP is wiped with random values.
End of note.

7.1.3.5 The Asymmetric Cryptographic Library v2.09.002

7.1.3.5.1 Rivest-Shamir-Adleman (RSA) operation

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

FCS_COP.1/RSA-0	**Cryptographic operation**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation, or FCS_CKM.5 Cryptographic key derivation] FCS_CKM.6 Timing and event of cryptographic key destruction
FCS_COP.1.1/RSA-0	The TSF shall perform *encryption, decryption, signature generation and signature verification in accordance with* in accordance with a specified cryptographic algorithm *Rivest-Shamir-Adleman (RSA)* and cryptographic key sizes from *1024 bits up to 4096+128 bits in 1 Bit stepping* that meet the following: ***Encryption:*** *1. According to section 5.1.1 RSAEP in PKCS [22]:* • *Supported for $n < 2^{2048+64}$* • *5.1.1(1) not supported* *2. According to section 8.2.2 IFEP-RSA in IEEE [29]:* *Supported for $n < 2^{2048+64}$* ***Decryption (with or without CRT):*** *1. According to section 5.1.2 RSADP in PKCS [22] for $u = 2$, i.e., without any (r_i, d_i, t_i), $i > 2$* • *5.1.2(1) not supported* • *5.1.2(2.a) supported for $n < 2^{2048 + 64}$* • *5.1.2(2.b) supported for $p \times q < 2^{4096 + 128}$* • *5.1.2(2.b) (ii)&(v) not applicable due to $u = 2$* *2. According to section 8.2.3 IEEE [29]:* • *8.2.1(I) supported for $n < 2^{2048 + 64}$* • *8.2.1(II) supported for $p \times q < 2^{4096 + 128}$* • *8.2.1(III) not supported*

Signature Generation (with or without CRT): *1. According to section 5.2.1 RSASP1 in PKCS [22] for u = 2, i.e., without any (r_i, d_i, t_i), i >2* • *5.2.1(1) not supported* • *5.2.1(2.a) supported for $n < 2^{2048 + 64}$* • *5.2.1(2b) supported for $p \times q < 2^{4096 + 128}$* • *5.2.1(2b) (ii)&(v) not applicable due to u = 2* *2. According to section 8.2.4 IFSP-RSA1 in IEEE [29]:* • *8.2.1(I) supported for $n < 2^{2048 + 64}$* • *8.2.1(II) supported for $p \times q < 2^{4096 + 128}$* • *8.2.1(III) not supported* ***Signature Verification:*** *1. According to section 5.2.2 RSAVP1 in PKCS [22]: supported for $n < 2^{4096 +128}$* • *5.2.2(1) not supported* *2. According to section 8.2.5 IEEE [29]:* • *Supported for $n < 2^{4096 +128}$* • *8.2.5(1) not supported*

Please consider also the statement of chapter 7.1.3.1.

7.1.3.5.2 Rivest-Shamir-Adleman (RSA) key generation

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

The RSA cryptographic library is offered in two parts: The 2k part of the RSA library can be used for key lengths of up to 2048 bits and the 4k part of the RSA library can be used for key lengths of up to 4096 bits.

FCS_CKM.1/RSA-0	Cryptographic key generation
Hierarchical to:	No other components.
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_CKM.5 Cryptographic key derivation, or FCS_COP.1 Cryptographic operation] [FCS_RBG.1 Random bit generation, or FCS_RNG.1 Generation of random numbers] FCS_CKM.6 Timing and event of cryptographic key destruction

FCS_CKM.1.1/RSA-0	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *implemented by following functions:* • *CryptoRSAKeyGen* • *CryptoRSAKeyGenMask_CRT plus CryptoGeneratePrime* • *CryptoRSAKeyGenMask_D plus CryptoGeneratePrime* • *CryptoRSAKeyGenMask_N plus CryptoGeneratePrime* and specified cryptographic key sizes of *1024 up to 4096+128 bits* that meet the following*:* *RSA key generation according to Infineon key generation methods. The generated keys are in conformance with:* *1. According to sections 3.1 and 3.2 in PKCS [22], for u = 2, i.e. without any (r_i, d_i, t_i), i > 2:* *3.1 supported for $n < 2^{4096 + 128}$* *3.2.(1) supported for $n < 2^{2048 + 64}$* *3.2.(2) supported for $p \times q < 2^{4096 + 128}$* *2. According to section 8.1.3.1 in IEEE [29]:* *8.1.3.1(1) supported for $n < 2^{2048 + 64}$* *8.1.3.1(2) supported for $p \times q < 2^{4096 + 128}$* *8.1.3.1(3) supported for $p \times q < 2^{2048 + 64}$*

Notes for FCS_CKM.1.1/RSA-0:
1. The function CryptoGeneratePrime has been assigned to the appropriate security level by BSI.
The proprietary CryptoGeneratePrimeMask function is not part of the scope of this certification.
End of note.

2. For easy integration of RSA functions into the user's operating system and/or application, the library contains single cryptographic functions respectively primitives which are compliant to the standard. The primitives are referenced above. Therefore, the library supports the user to develop an application representing the standard if required.
Please consider also the statement of chapter 7.1.3.1.
End of note.

3. The TOE can be delivered with or without the RSA library. In the case of coming without the RSA library, the TOE does not provide the Additional Specific Security Functionality Rivest-Shamir-Adleman Cryptography (RSA) realized with the security functional requirements FCS_COP.1/RSA-0 and FCS_CKM.1/RSA-0. In case of a blocked Crypto2304T the optionally delivered cryptographic and the supporting Toolbox cannot be used in that TOE product.
End of notes.

7.1.3.5.3 General Preface regarding Elliptic Curve Cryptography

The EC library is delivered as object code and in this way integrated in the user software. The security functional requirement covers the standard Brainpool [19] and NIST [26] Elliptic Curves with key lengths of 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 bits. Numerous other curve types, being also secure in terms of side channel attacks on this TOE, exist, which the user optionally can add in the composition certification process.
All curves are based on finite field GF(p) with value $p \in [2^{41-1}; 2^{521}[$ as well as curves based on a finite field $GF(2^n)$ with size $n \in [41-1; 521]$ are supported.

Elliptic Curve DSA (ECDSA) operation

FCS_COP.1/ECDSA-0	Cryptographic operation
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation, or FCS_CKM.5 Cryptographic key derivation] FCS_CKM.6 Timing and event of cryptographic key destruction
FCS_COP.1.1/ECDSA-0	The TSF shall perform *signature generation and signature verification* in accordance with a specified cryptographic algorithm *ECDSA* and cryptographic key sizes *160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 bits* that meet the following: ***ECDSA Signature Generation:*** *1. According to section 7.3 Signing Process in ANSI [23]* • *Step d) and e) are not supported* • *The output of step e) has to be provided as input to our function by the caller.* • *Deviation of step c) and f):* o *The jumps to step a) were substituted by a return of the function with an error code, the jumps are emulated by another call to our function.* *2. According to sections 6.4.3 Signature Process in ISO/IEC [27]* • *Chapter 6.4.3.3 is not supported* • *Chapter 6.4.3.5 is not supported* o *the hash-code of H of the message has to be provided by the caller as input for our function.* • *Chapter 6.4.3.7 is not supported* • *Chapter 6.4.3.8 is not supported* *3. According to section 7.2.7 ECSP-DSA in IEEE [29]* • *Deviation of step (3) and (4):* o *The jumps to step 1 were substituted by a return of the function with an error code, the jumps are emulated by another call to our function* ***ECDSA Signature Verification:*** *1. According to section 7.4.1 in ANSI [23]* • *Step b) and c) are not supported.* • *The output of step c) has to be provided as input to our function by the caller.* • *Deviation of step d):* o *Beside noted calculation, our algorithm adds a random multiple of BasepointerOrder n to the calculated values u1 and u2.* *2. According to sections 6.4.4 Signature Verification Process in ISO/IEC [27]* • *Chapter 6.4.4.2 is not supported* • *Chapter 6.4.4.3 is not supported:* o *The hash-code H of the message has to be provided by the caller as input to our function* *3. According to section 7.2.8 ECVP-DSA in IEEE [29].*

Note for FCS_COP.1/ECDSA-0
For easy integration of EC functions into the user's operating system and/or application, the library contains single cryptographic functions respectively primitives which are compliant to the standard. The primitives are referenced above. Therefore, the library supports the user to develop an application representing the standard if required.
End of note.

7.1.3.5.4 Elliptic Curve (EC) key generation

The key generation for the EC shall meet the requirement “Cryptographic key generation (FCS_CKM.1)”.

FCS_CKM.1/EC-0	Cryptographic key generation
Hierarchical to:	No other components.
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_CKM.5 Cryptographic key derivation, or FCS_COP.1 Cryptographic operation] [FCS_RBG.1 Random bit generation, or FCS_RNG.1 Generation of random numbers] FCS_CKM.6 Timing and event of cryptographic key destruction
FCS_CKM.1.1/EC-0	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *implemented by following functions which can be used independently of each other:* • *ECC_ECDSAKeyGen* • *ECC_ECDSAKeyGenMask* *specified in [23], [27] and [29]* and specified cryptographic key sizes *160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 bits* that meet the following: *ECDSA Key Generation:* *1. According to the appendix “A4.3 Elliptic Curve Key Pair Generation” in ANSI [23]:* *The optional cofactor h is not supported.* *2. According to section “6.4.2 Generation of signature key and verification key” in ISO/IEC [27].* *3. According to appendix “A.16.9 An algorithm for generating EC keys” in IEEE [29]*

Note for FCS_CKM.1/EC-0:
For easy integration of EC functions into the user’s operating system and/or application, the library contains single cryptographic functions respectively primitives which are compliant to the standard. The primitives are referenced above. Therefore, the library supports the user to develop an application representing the standard if required.
End of note.

7.1.3.5.5 Elliptic Curve Diffie-Hellman (ECDH) key agreement

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

FCS_COP.1/ECDH-0	Cryptographic operation
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation, or FCS_CKM.5 Cryptographic key derivation] FCS_CKM.6 Timing and event of cryptographic key destruction
FCS_COP.1.1/ECDH-0	The TSF shall perform *elliptic curve Diffie-Hellman key agreement* in accordance with a specified cryptographic algorithm *ECDH* and cryptographic key sizes 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 bits that meet the following: *1. According to section "5.4.1 Standard Diffie-Hellman Primitive" in ANSI [24]* *Unlike section 5.4.1(3) our implementation not only returns the x-coordinate of the shared secret, but rather the x-coordinate and the y-coordinate.* *2. According to "Appendix D.6 Key agreement of Diffie-Hellman" type in ISO/IEC [28]* *The function enables the operations described in appendix D.6* *3. According to section 7.2.1 ECSVHDP-DH" in IEEE [29]* *Unlike section 7.2.1 our implementation not only returns the x-coordinate of the shared secret, but rather the x-coordinate and the y-coordinate.*

Notes for: FCS_COP.1/ECDH-0

1. The security functional requirement covers the standard Brainpool [19] and NIST [26] Elliptic Curves with key lengths of 256, 283, 320, 384, 409, 512 or 521 Bits, due to national regulations by the BSI. Furthermore, 160, 163, 224, 233 and 192 bits are supported. Numerous other curve types, being also secure in terms of side channel attacks on this TOE, exist, which the user optionally can add in the composition certification process.
End of note

2. For easy integration of EC functions into the user's operating system and/or application, the library contains single cryptographic functions respectively primitives which are compliant to the standard. The primitives are referenced above. Therefore, the library supports the user to develop an application representing the standard if required.
End of note.

3. The TOE can be delivered with or without the EC library. In the case the TOE comes without, it does not provide the Additional Specific Security Functionality Elliptic Curve Cryptography realized with the security functional requirements FCS_COP.1/ECDSA-0. In case of a blocked Crypto2304T, the RSA and EC cryptographic library cannot be used. In case of a blocked Crypto2304T the optionally delivered cryptographic RSA and EC, as well as the supporting Toolbox cannot be used in that TOE product.
End of note.

4. The EC primitives allow the selection of various curves. The selection of the curves depends to the user.

End of notes.

7.1.4 Data Integrity FDP_SDI.2 and FDP_SDC.1

The TOE shall meet the requirement Stored data integrity monitoring and action (FDP_SDI.2) as specified below:

FDP_SDI.2	**Stored data integrity monitoring and action**
Hierarchical to:	FDP_SDI.1 stored data integrity monitoring
Dependencies:	No dependencies
FDP_SDI.2.1	The TSF shall monitor user data *of the Composite TOE* stored in containers controlled by the TSF for *data integrity and one- and/or more-bit-errors* on all objects, based on the following attributes: *corresponding EDC value for RAM, ROM and SOLID FLASH™ NVM and error correction ECC for the SOLID FLASH™ NVM.*
FDP_SDI.2.2	Upon detection of a data integrity error, the TSF shall *correct 1 bit errors in the SOLID FLASH™ NVM automatically and inform the user about more bit errors.*

Note:
All HSL versions implement tearing save behavior for the SOLID Flash™ NVM. Its features are close to physical SOLID Flash™ NVM behavior which is not part of the SPM. Therefore, the HSL functionality contributing to FPT_FLS:1, FPT_PHP.3 and FDP_SDI.2 is excluded from the SPM even though it contributes to the named SFRs which are listed and modelled in the ADV_SPM.1.1D definition.
End of note.

The TSF shall ensure the confidentiality of [selection: all user data, the following user data [assignment: list of user data]] while it is stored in the [selection: temporary memory, persistent memory, any memory].

The TOE shall meet the requirement “Stored data confidentiality (FDP_SDC.1)” as specified below:

FDP_SDC.1	**Stored data confidentiality**
Hierarchical to:	No other components
Dependencies:	No dependencies
FDP_SDC.1.1	The TSF shall ensure the confidentiality of *the information of the user data of the Composite TOE* while it is stored in the *any memory.*

7.1.5 Limited Capabilities and Limited Availability

The SFR’s FMT_LIM.1 and FMT_LIM.2 from [PP0084] are adapted due to CC:2022.

FMT_LIM.1	**Limited Capabilities**
Hierarchical to:	No other components.
Dependencies:	FMT_LIM.2: Limited availability
FMT_LIM.1.1	The TSF shall limit its capabilities so that in conjunction with “Limited availability (FMT_LIM.2)” the following policy is enforced: *Deploying Test Features after TOE Delivery does not allow user data of the Composite TOE to be disclosed or manipulated, TSF data to be disclosed or manipulated, software to be reconstructed and no substantial information about construction of TSF to be gathered which may enable other attacks.*

FMT_LIM.2	Limited availability
Hierarchical to:	No other components.
Dependencies:	FMT_LIM.1 Limited capabilities.
FMT_LIM.2.1	The TSF shall be designed in a manner that limits its availability so that in conjunction with "Limited capabilities (FMT_LIM.1)" the following policy is enforced: *Deploying Test Features after TOE Delivery does not allow user data of the Composite TOE to be disclosed or manipulated, TSF data to be disclosed or manipulated, software to be reconstructed and no substantial information about construction of TSF to be gathered which may enable other attacks.*

7.1.6 Application Note to FPT_FLS.1 Failure with preservation of secure state

The SFR FPT_FLS.1 is defined in the PP [9] but with application note 14 it is required to describe the secure state:
Where ever FPT_FLS.1 is claimed it depends on the source or affected piece of data the failure was detected. Induced failures on other than memory contents could in the worse lead to exploitable situations if not discovered and treated properly. This includes i.e. internal transported keys and other TSF data. For that reasons the integrity of the transport ways and also the modules the TSF and critical data are used is protected against failure.
Depending on the criticality where the failure occurred and in most cases a security reset is triggered. Only in case of a peripheral bus error or an address translation mismatch, a user software action is required as described in the user guidance.
In case of a security reset, the user cannot take any action. The chip immediately stops operation and reboots.

7.2 Support by the Flash Loader

The TOE provides the Flash Loader to download user data into the SOLID FLASH™ NVM, either during production of the TOE or at customer site. The Flash Loader is dedicated for usage by authorized users only in secured and insecure environment during the production up to Phase 6 Security IC Personalisation.
The Flash Loader has to be permanently deactivated before entering Phase 7 Security IC end-usage.
For this reason the TOE shall meet the requirements as defined and described in the PP [9] section 7.3 Packages for Loader and 7.2 Package Authentication of the Security IC:

- Limited capabilities (FMT_LIM.1/Loader),
- Limited availability – Loader (FMT_LIM.2/Loader),
- Authentication Proof of Identity (FIA_API.1),
- Inter-TSF trusted channel (FTP_ITC.1),
- Basic data exchange confidentiality (FDP_UCT.1),
- Data exchange integrity (FDP_UIT.1),
- Subset access control – Loader (FDP_ACC.1/Loader) and
- Security attribute based access control – Loader (FDP_ACF.1/Loader)

as defined in the PP [9], section 7.2 and 7.3.

The Flash Loader supports the following security function policy (SFP):

- Loader SFP:
  provides the mutual authentication between the TOE and the administrator user or download operator user and the download of the user data into the memory of the TOE.

The Flash Loader supports the following two subjects:

- Administrator user:
  is enabled performing mutual authentication with the keys Kc and Kd, to manage (set, exchange, delete) the keys Kc, Kd and Kfdi and to process the download of the user data into the memory of the TOE.
- Download operator user:
  is enabled performing mutual authentication with Kd, to exchange the key Kd and to perform the download of the user data into the memory of the TOE. He can also delete Kfdi.

The Flash Loader supports the following object:

- User data:
  Data loaded into the memory of the TOE.

The TOE shall meet the requirement “Limited capabilities (FMT_LIM.1/Loader)” as specified below:

FMT_LIM.1/Loader	**Limited capabilities**
Hierarchical to:	No other components.
Dependencies:	FMT_LIM.2 Limited availability.
FMT_LIM.1.1/Loader	The TSF shall limit its capabilities so that in conjunction with “Limited availability (FMT_LIM.2)” the following policy is enforced *Deploying Loader functionality after permanent deactivation does not allow stored user data of the Composite TOE to be disclosed or manipulated by unauthorized user.*

The TOE shall meet the requirement “Limited availability – Loader (FMT_LIM.2/Loader)” as specified below:

FMT_LIM.2/Loader	**Limited availability - Loader**
Hierarchical to:	No other components.
Dependencies:	FMT_LIM.1 Limited capabilities.
FMT_LIM.2.1/Loader	The TSF shall be designed in a manner that limits its availability so that in conjunction with “Limited capabilities (FMT_LIM.1)” the following policy is enforced: *The TSF prevents deploying the Loader functionality after permanent deactivation.*

The TOE shall meet the requirement “Limited availability – Loader (FIA_API.1)” as specified below:

FIA_API.1	**Authentication Proof of Identity**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FIA_API.1.1	The TSF shall provide an *authentication mechanism according to [34] ISO/IEC 9798-2 section 6.2.2 Mechanism 4: Three-path authentication based on the security attributes (keys) Kc or Kd* to prove the identity of *the TOE* by including the following properties *Flash Loader administrator or download operator role* to an external entity.

Security Requirements (ASE_REQ)

The TOE shall meet the requirement "Limited availability – Loader (FTP_ITC.1)" as specified below:

FTP_ITC.1	**Inter-TSF trusted channel**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FTP_ITC.1.1	The TSF shall provide a communication channel between itself and the *administrator user and the Download Operator user as described in the Loader SFP* that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2	The TSF shall permit *another trusted IT product* to initiate communication via the trusted channel.
FTP_ITC.1.3	The TSF shall initiate communication via the trusted channel for *deploying Loader for downloading user data.*

The TOE shall meet the requirement "Limited availability – Loader (FDP_UCT.1)" as specified below:

FDP_UCT.1	**Basic data exchange confidentiality**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FDP_UCT.1.1	The TSF shall enforce the *Loader SFP* to *receive* user data in a manner protected from unauthorised disclosure.

The TOE shall meet the requirement "Limited availability – Loader (FDP_UIT.1)" as specified below:

FDP_UIT.1	**Data exchange integrity**
Hierarchical to:	No other components.
Dependencies:	[FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control].
FDP_UIT.1.1	The TSF shall enforce the *Loader SFP* to *receive* user data in a manner protected from *modification, deletion or insertion* errors.
FPP_UIT.1.2	The TSF shall be able to determine on receipt of user data, whether *modification, deletion or insertion* have occurred.

Note regarding the Flash Loader:
The algorithm strength of the Flash Loader has not been cryptographic-analytically evaluated by the BSI, as Infineon has implemented a proprietary extension.
End of note.

End of note.

The TOE shall meet the requirement "Limited availability – Loader (FDP_ACC.1/Loader)" as specified below:

FDP_ACC.1/Loader	**Subset access control - Loader**
Hierarchical to:	No other components.
Dependencies:	FDP_ACF.1 Security attribute based access control.
FDP_ACC.1.1/Loader	The TSF *shall enforce the Loader SFP on* (1) the subjects *Administrator user and the Download operator user,* (2) the objects User data in *the SOLID FLASH™ NVM memory of the TOE*, (3) the operation deployment of the Loader.

The TOE shall meet the requirement "Limited availability – Loader (FTP_ACF.1/Loader)" as specified below:

FDP_ACF.1/Loader	**Security attribute based access control - Loader**
Hierarchical to:	No other components.
Dependencies:	FMT_MSA.3 Static attribute initialisation
FDP_ACF.1.1/Loader	The TSF shall enforce the *Loader SFP* to objects based on the following: *(1) the subjects Administrator user and the Download operator user with security attributes Kc, Kd and Kfdi* (2) *the objects user data in data loaded into the SOLID FLASH™ NVM memory of the TOE with security attributes Kfdi.*
FDP_ACF.1.2/Loader	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *(1) evaluate the corresponding access control information of the relevant subject, administrator user and download operator user, before the access, so that accesses to be denied cannot be utilized by the subject attempting to perform the operation. The subsequent download is then protected by the key Kfdi.*
FDP_ACF.1.3/Loader	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: None
FDP_ACF.1.4/Loader	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: None

Note 13:
Regarding FDP_ACF.1.4/Loader it is added in the User Guidance that the Flash Loader has to be permanently deactivated prior delivery to the end-user.
End of note.

Note 14:
The security functional requirements FMT_LIM.1/Loader, FMT_LIM.2/Loader, FIA_API.1, FTP_ITC.1, FDP_UCT.1, FDP_UIT.1, FDP_ACC.1/Loader and FDP_ACF.1/Loader apply only at TOE products coming with activated Flash Loader enabled for user data download. In other cases the Flash Loader is not available anymore and the user software or data download is completed. Depending on the capabilities of the user software these security functional requirements may then reoccur as subject of the composite TOE.
The permanent locking of the Flash Loader after finalizing the download and prior delivery to the end-user is added to package 2 with LIM1/Loader and FMT_LIM.2/Loader.
End of note.

7.3 TOE Security Assurance Requirements

The evaluation assurance level is EAL6 augmented with ALC_FLR.3 and EAL5 augmented with ALC_FLR.3, ADV_IMP.2, ADV_INT.3, ADV_TDS.5, ALC_CMC.5, ALC_DVS.2, ATE_COV.3, ATE_FUN.2, AVA_VAN.5 and ALC_TAT.3 (multi assurance).
In the following table, the security assurance requirements are given. The augmentation of the assurance components compared to the Protection Profile [9] is expressed with bold letters. The global SARs are all from the table as follows bar ADV_SPM.1:

Table 11 Assurance Components

Aspect	Acronym	Description	Refinement
Development	ADV_ARC.1	Security Architecture Description	In PP [9]
	ADV_FSP.5	**Complete semi-formal functional specification with additional error information**	in ST
	ADV_IMP.2	**Complete mapping of the implementation representation of the TSF**	in ST
	ADV_INT.3	**Minimally complex internals**	
	ADV_TDS.5	**Complete semi-formal modular design**	
	ADV_SPM.1	**Formal TOE security policy model (only claimed for the EAL6 augmented covered sub TSF MODELLED_SFR)**	
Guidance Documents	AGD_OPE.1	Operational user guidance	in PP [9]
	AGD_PRE.1	Preparative procedures	in PP [9]
Life-Cycle Support	**ALC_CMC.5**	**Advanced support**	in ST
	ALC_CMS.5	**Development tools CM coverage**	in ST
	ALC_DEL.1	Delivery procedures	in PP [9]
	ALC_DVS.2	Sufficiency of security measures	in PP [9]
	ALC_LCD.1	Developer defined life-cycle model	
	ALC_TAT.3	**Compliance with implementation standards – all parts**	
	ALC_FLR.3	**Systematic Flaw Remediation**	
Security Target Evaluation	ASE_CCL.1	Conformance claims	
	ASE_ECD.1	Extended components definition	
	ASE_INT.1	ST introduction	
	ASE_OBJ.2	Security objectives	
	ASE_REQ.2	Derived security requirements	
	ASE_SPD.1	Security problem definition	
	ASE_TSS.1	TOE summary specification	
Tests	**ATE_COV.3**	**Rigorous analysis of coverage**	In ST
	ATE_DPT.3	**Testing: modular design**	
	ATE_FUN.2	**Ordered functional testing**	
	ATE_IND.2	Independent testing – sample	
Vulnerability Assessment	AVA_VAN.5	Advanced methodical vulnerability testing	in PP [9]

7.3.1 Refinements

Some refinements are taken unchanged from the PP [9]. In some cases a clarification is necessary. In Table 17 an overview is given where the refinement is done.
The refinements from the PP [9] have to be discussed here in the Security Target, as the assurance level is increased. The refinements from the PP [9] are included in the chosen assurance level EAL 6 augmented with ALC_FLR.3.

7.3.1.1 *Development (ADV)*

ADV_IMP Implementation Representation:

The refined assurance package ADV_IMP.1 implementation representation of the TSF requires the availability of the entire implementation representation, a mapping of the design description to the implementation representation with a level of detail that the TSF can be generated without further design decisions. In addition, the correspondence of design description and implementation representation shall be demonstrated.
The covered higher assurance package ADV_IMP.2 requires a complete and not curtailed mapping of the implementation representation of the TSF, and the mapping of the design description to the entire implementation representation. In addition, the correspondence of design description and the implementation representation shall be demonstrated. The ADV_IMP.1 aspect and refinement remains therefore valid. The enhancement underlines the refinement in the PP [9] and by that the entirely complete design i.e. not curtailed representation with according mapping was provided, demonstrated and reviewed.

ADV_FSP Functional Specification:

The ADV_FSP.4 package requires a functional description of the TSFIs and there assignment to SFR-enforcing, SFR-supporting, SFR-non-interfering, including related error messages, the assurance package. The enhancement of ADV_FSP.5 requires additionally a complete semi-formal functional specification with additional error information. In addition the package includes a tracing from the functional specification to the SFRs, as well as the TSFIs descriptions including error messages not resulting from an invocation of a TSFI.
These aspects from ADV_FSP.5 are independent from the ADV_FSP.4 refinements from the PP [9] but constitute an enhancement of it. By that the aspects of ADV_FSP.4 and its refinement in the PP [9] apply also here. The assurance and evidence was provided accordingly.

ADV_SPM Formal Security Policy Model

It is the objective of this family to provide additional assurance from the development of a formal security policy model of the TSF, and establishing a correspondence between the functional specification and this security policy model. Preserving internal consistency the security policy model is expected to formally establish the security principles from its characteristics by means of a mathematical proof.

Security Requirements (ASE_REQ)

The SFR's covered by ADV_SPM is represented by the sub TSF MODELLED_SFR:

- FDP_ACC.1 Subset Access Control
- FDP_ACF.1 Security attribute based access control
- FMT_MSA.1 Management of Security Attributes
- FMT_MSA.3 Static Attribute Initialization.
- FDP_SDI.2 Stored data integrity monitoring and action
- FDP_SDC.1 Stored data confidentiality
- FDP_ITT.1 Basic Internal Transfer Protection
- FDP_IFC.1 Information Flow Control
- FPT_ITT.1 Basic internal TSF data transfer protection
- FPT_PHP.3 Resistance to physical attack
- FPT_FLS.1 Failure with preservation of secure state
- FRU_FLT.2 Limited fault tolerance
- FMT_LIM.1 Limited capabilities
- FMT_LIM.2 Limited availability
- FAU_SAS.1 Audit storage
- FMT_SMF.1 Specification of Management Functions
- FIA_API.1
- FMT_LIM.1/Loader
- FMT_LIM.2/Loader
- FTP_ITC.1
- FDP_UCT.1
- FDP_UIT.1
- FDP_ACC.1/Loader
- FDP_ACF.1/Loader

The assurance level for these SFR's is EAL6 augmented. For sub TSF ALL_SFR it is EAL5 augmented.

Security Objectives covered by sub TSF MODELLED_SFR:

- O.Phys-Manipulation
- O.Phys-Probing
- O.Malfunction
- O.Leak-Inherent
- O.Leak-Forced
- O.Abuse-Func
- O.Identification
- O.Cap_Avail_Loader
- O.Authentication
- O.Ctrl_Auth_Loader

- O.Mem-Access
- O.Prot_TSF_Confidentiality

other objectives are not modelled.

7.3.1.2 *Life-cycle Support (ALC)*

ALC_CMS Configuration Management Scope:

The refinement of ALC_CMS.4 in the PP [9] defines with some conditions the possibility that the Security IC embedded firmware can be part of the delivered TOE. This is the case as the Security IC embedded firmware – regardless of the version in use - and the optional software are part of TOE and delivered together with the TOE as the firmware and optional software are stored in the ROM and/or SOLID FLASH™ NVM. The presence of the optional parts belongs to the user order. Both, the firmware and software delivered with the TOE are controlled entirely by Infineon Technologies.

In addition, the TOE offers the possibility that the user can download his software at his own premises. These parts of the software are user controlled only and are not part of this TOE. The download of this solely user controlled software into the SOLID FLASH™ NVM is protected by strong authentication means. In addition, the download itself could also be encrypted.

The requirements by ALC_CMS.4 are fulfilled including the refinements from the PP [9].
The ALC_CMS.5 implements the additional requirement to include development tools and related information into the configuration list. This add on has no influence on the refinement of ALC_CMS.4 in the PP [9] as the delivery of Security IC embedded firmware is independent from the TOE development tools and related information. The package ALC_CMS.5 is therefore an enhancement to ACL_CMS.4 and the package with its refinement in the PP [9] remains valid. The assurance and evidence was provided accordingly.

ALC_CMC Configuration Management Capabilities:

The PP refinement from the assurance package ALC_CMC.4 Production support, acceptance procedures and automation points out that the configuration items comprise all items defined under ALC_CMS to be tracked under configuration management. In addition a production control system is required guaranteeing the traceability and completeness of different charges and lots. Also the number of wafers, dies and chips must be tracked by this system as well as procedures applied for managing wafers, dies or complete chips being removed from the production process in order to verify and to control predefined quality standards and production parameters. It has to be controlled that these wafers, dies or assembled devices are returned to the same production stage from which they are taken or they have to be securely stored or destroyed otherwise.
The additionally covered extended package of ALC_CMC.5 Advance Support requires advanced support considering the automatisms configuration management systems, acceptance and documentation procedures of changes, role separation with regard to functional roles of personnel, automatisms for tracking and version controlling in those systems, and includes also production control systems. The additional aspects of ADV_CMC.5 constitute an enhancement of ACL_CMC.4 and therefore the aspects and ACL_CMC.4 refinements in the PP [9] remain valid. The assurance and evidence was provided.

ALC_DVS Development Security:

The requirement ALC_DVS.2 is refined in the PP [9] requiring the development security documentation comprising all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment. The documentation and evidence was given, reviewed, audited and thus the assurance and evidence was provided accordingly.

7.3.1.3 *Tests (ATE)*

ATE_COV Test Coverage:

The PP refined assurance package ATE_COV.2 Analysis of coverage addresses the extent to which the TSF is tested, and whether or not the testing is sufficiently extensive to demonstrate that the TSF operates as specified. It includes the test documentation of the TSFIs in the functional specification. In particular the refinement requires that The TOE must be tested under different operating conditions within the specified ranges. In addition, the existence and effectiveness of mechanisms against physical attacks should be covered by evidence that the TOE has the particular physical characteristics. This is furthermore detailed in the PP [9].
This assurance package ATE_COV.2 has been enhanced to ATE_COV.3 to cover the rigorous analysis of coverage. This requires the presence of evidence that exhaustive testing on rigorous entirely all interfaces as documented in the functional specification was conducted. By that ATE_COV.2 and refinements as given in the PP [9] are enhanced by ATE_COV.3 and remain as well. The TSFIs were completely tested according to ATE_COV.3 and the assurance and evidence was provided.

7.3.1.4 *AVA_VAN Vulnerability Analysis*

The assurance package AVA_VAN remains unchanged compared to the forerunner processes and requires advanced methodical vulnerability analysis.

7.4 Security Requirements Rationale

7.4.1 Rationale for the Security Functional Requirements

The objectives O.Authentication and OE.TOE_Auth are discussed in the PP [9] chapter 7.2.1.
The objectives O.Cap_Avail_Loader and OE.Lim_Block_Loader and the covering security functional requirements FMT_LIM.1/Loader and FMT_LIM.2/Loader are discussed in the PP [9] chapter 7.3.1.
The policy P.Ctrl_Loader and the objectives O.Ctrl_Auth_Loader and OE.Loader_usage are discussed in the PP [9] chapter 7.3.2.

Additionally, the objective O.Add-Function includes the objectives implemented by each of the CIPURSE™ Cryptographic Library versions: O.Ctrl_Auth_CCL, O.Prot_Integrity and O.Prot_Confidentiality. These extended objectives are discussed in this ST in chapter 5.5 P.Add-Functions.

The additional objectives O.Prot_TSF_Confidentiality is defined in chapter 5.1 and 5.8 in this document.
PP [9] chapter 6.1 includes also the definition of FDP_SDI.2 „Stored data integrity monitoring and action“.
While the above mentioned security functional requirements rationale of the TOE are defined and described in PP [9] section 6.3.1, the additional introduced SFRs are listed and discussed below:

Table 12 Rational for additional SFR in the ST

Objective	TOE Security Functional Requirements
O.Add-Functions	FCS_COP.1/RSA-0Cryptographic operation for ACL-0 FCS_COP.1/ECDSA-0 Cryptographic operation for ACL-0 FCS_COP.1/ECDH-0 Cryptographic operation for ACL-0 FCS_CKM.1/RSA-0 Cryptographic key generation for ACL-0 FCS_CKM.1/EC-0 Cryptographic key generation for ACL-0 FCS_CKM.1/CCL Cryptographic key generation for CCL FCS_COP.1/CCL Cryptographic Operation CCL Trusted Channel for CCL FCS_CKM.6/CCL Cryptographic key destruction for CCL
O.Phys-Manipulation	FPT_TST.2 Subset TOE security testing
O.Mem-Access	FDP_ACC.1 Subset access control FDP_ACF.1 Security attribute based access control FMT_MSA.3 Static attribute initialisation FMT_MSA.1 Management of security attributes FMT_SMF.1 Specification of Management Functions
O.Malfunction	FDP_SDI.2 Stored data integrity monitoring and action
O.RND	FCS_RNG.1/TRNG Generation of Random Numbers -TRNG FCS_RNG.1/HRNG Generation of Random Numbers - HRNG FCS_RNG.1/DRNG Generation of Random Numbers -DRNG FCS_RNG.1/KSG Generation of Random Numbers - KSG
O.Prot_TSF_ Confidentiality	FTP_ITC.1 Inter-TSF-trusted channel FDP_ACC.1/Loader Subset access control –Loader FDP_ACF.1/Loader Security attribute based access control – Loader
O.TDES	FCS_COP.1/TDES Cryptographic operation – TDES by SCP FCS_CKM.6/TDES Cryptographic key destruction – TDES by SCP
O.AES	FCS_COP.1/AES Cryptographic operation – AES by SCP

Security Requirements (ASE_REQ)

Objective	TOE Security Functional Requirements
	FCS_CKM.6/AES Cryptographic key destruction – AES by SCP
O.Ctrl_Auth_CCL	FCS_COP.1/CCL Cryptographic operation –by CCL FCS_CKM.1/CCL Cryptographic key generation –by CCL FCS_CKM.6/CCL Cryptographic key destruction –by CCL
O.Prot_Integrity	FCS_COP.1/CCL Cryptographic operation –by CCL FCS_CKM.1/CCL Cryptographic key generation –by CCL FCS_CKM.6/CCL Cryptographic key destruction –by CCL
O.Prot_Confidentiality	FCS_COP.1/CCL Cryptographic operation –by CCL FCS_CKM.1/CCL Cryptographic key generation –by CCL FCS_CKM.6/CCL Cryptographic key destruction –by CCL

The table above gives an overview, how the security functional requirements are combined to meet the security objectives.

For clarity please find following assignment to versions:

- ACL-0 covers version v2.09.002
- CCL covers version v02.00.0004

7.4.1.1 Cryptographic Aspects

The justification related to the security objective “Additional Specific Security Functionality (O.Add-Functions)” is as follows:
The security functional requirement(s) “Cryptographic operation (FCS_COP.1)” exactly require(s) those functions to be implemented which are demanded by O.Add-Functions:

- The ACL library implements the SFR FCS_CKM.1/EC, supporting the generation of EC keys, needed for these cryptographic operations.

- The ACL library v2.09.002 implements the SFRs FCS_CKM.1/RSA-0, supporting the generation of RSA keys.

- The CIPURSE™ Cryptographic Library implements the SFRs FCS_COP.1/CCL Cryptographic Operation CCL Trusted Channel, establishing the trusted channel between two authenticated entities and FCS_CKM.1/CCL Cryptographic key generation, supporting the generation of the keys used for authentication and confidentiality. The CIPURSE™ CL operation requires the presence of the SCP.

The implementation covers the functional requirements and meets the objective O.Add-Functions.
The use of the supporting library Toolbox has no impact on any security functional requirement nor does the use generate additional requirements. The Toolbox library is not part of the TOE Security Functionality (TSF).

All these requirements have to be fulfilled to support OE.Resp-Appl for the SFRs implemented by the:

- SCP:
  - FCS_COP.1/TDES
  - FCS_COP.1/AES
- CCL for version v02.00.0004:
  - FCS_COP.1/CCL Cryptographic Operation CCL Trusted Channel,
  - FCS_CKM.1/CCL Cryptographic key generation
  - FCS_CKM.6/CCL Cryptographic key destruction

Nevertheless, the developer of the Smartcard Embedded Software must ensure that the additional functions are used as specified and that the User data of the Composite TOE processed by these functions are protected as defined for the application context. These issues are addressed by the specific security functional requirements:

- [FDP_ITC.1 Import of user data of the Composite TOE without security attributes or FDP_ITC.2 Import of user data of the Composite TOE with security attributes or FCS_CKM.1 Cryptographic key generation],
- FCS_CKM.6 Cryptographic key destruction

The security functional requirements required to meet the security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced define how to implement the specific security functionality. However, key-dependent functions could be implemented in the Smartcard Embedded Software.

The usage of cryptographic algorithms requires the use of appropriate keys. Otherwise these cryptographic functions do not provide security. The keys have to be unique with a very high probability, and must have a certain cryptographic strength etc. In case of a key import into the TOE (which is usually after TOE delivery) it has to be ensured that quality and confidentiality are maintained. Keys for DES and AES are provided by the environment. Keys for RSA and EC algorithms can be provided either by the TOE or the environment.

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

7.4.1.2 Hardware related Aspects

The security functional component Subset TOE security testing (FPT_TST.2) has been newly created (Common Criteria Part 2 extended). This component allows that particular parts of the security mechanisms and functions provided by the TOE can be tested after TOE Delivery. This security functional component is used instead of the functional component FPT_TST.1 from Common Criteria Part 2. For the user it is important to know which security functions or mechanisms can be tested. The functional component FPT_TST.1 does not mandate to explicitly specify the security functions being tested. In addition, FPT_TST.1 requires verification of the integrity of TSF data and stored TSF executable code which might violate the security policy.
The tested security enforcing functions are SF_DPM Device Phase Management, SF_CS Cryptographic Support and SF_PMA Protection against modifying attacks.

The justification related to the security objective “Protection against Physical Manipulation (O.Phys-Manipulation)” is as follows:
The security functional requirement FPT_TST.2 will detect attempts to conduce a physical manipulation on the monitoring functions of the TOE. The objective of FPT_TST.2 is O.Phys-Manipulation. The physical manipulation will be tried to overcome security enforcing functions.

The security functional requirement “Subset access control (FDP_ACC.1)” with the related Security Function Policy (SFP) “Memory Access Control Policy” exactly require the implementation of an area based memory access control as required by O.Mem-Access. The related TOE security functional requirements FDP_ACC.1, FDP_ACF.1, FMT_MSA.3, FMT_MSA.1 and FMT_SMF.1 cover this security objective. The implementation of these functional requirements is represented by the dedicated privilege level concept.

The justification of the security objective and the additional requirements show that they do not contradict to the rationale already given in the Protection Profile for the assumptions, policy and threats defined there. Moreover, these additional security functional requirements cover the requirements by the PP [9] user data of the Composite TOE protection of chapter 1.2.5 claim 35 and 36 which are not refined by the PP [9]. Nevertheless, the developer of the Smartcard Embedded Software must ensure that the additional functions are used as specified and that the User data of the Composite TOE processed by these functions are protected as defined for the application context. The TOE only provides the tool to implement the policy defined in the context of the application.

The justification related to the security objective Protection against Malfunction due to Environmental Stress (O.Malfunction) is as follows:
The security functional requirement Stored data integrity monitoring and action (FDP_SDI.2) requires the implementation of an integrity observation and correction which is implemented by the Error Detection (EDC) and Error Correction (ECC) measures. The EDC is present throughout all memories of the TOE while the ECC is realized in the SOLID FLASH™ NVM. These measures detect and inform about one and more bit errors. In case of the SOLID FLASH™ NVM 1 bit errors of the data are corrected automatically. The ECC

mechanism protects the TOE from the use of corrupt data. Therefore FDP_SDI.2 is suitable to meet the security objective O.Malfunction.

The presence of true random numbers is the security goal 4 (SG4) which is formalized in the objective O.RND Random Numbers. This objective must be covered by fulfillment of the security functional requirement FCS_RNG. This is defined in the PP [9] chapter 5.1. The requirement implements a quality metric which is defined by national regulations. The implemented random number generation fulfils the definitions of AIS 31 [13] in the quality classes as outlined in chapter 7.1.3.2. Therefore the SFR FCS_RNG and the objective O.RND are covered.

7.4.1.3 Flash Loader related Aspects

The justification related to the Flash Loader security objectives are as follows. Note that the following objectives and related rationales apply only at TOE products coming with activated Flash Loader enabled for software or data download by the user. In other cases the Flash Loader is not available anymore and the user data download is completed. Depending on the capabilities of the user software these security functional requirements may then reoccur as subject of the composite TOE.
The Flash Loader related objectives are:

The objective O.Authentication requires the presence of an authentication mechanism proving the identity of a given security IC to an external entity. This objective is covered by the functional requirement FIA_API Authentication Proof of Identity. The Flash Loader implements this functionality and outputs identification data to external requesting entity. The user guidance describes in more detail how this authentication request is applied and conducted by the external entity. As the functional requirements are met by the Flash Loader the objective is covered.

The objective O.Cap_Avail_Loader requires limited capabilities of the Loader functionality and irreversible termination of the Loader. First, this is covered by the functional requirement FMT_LIM.1/Loader which implements protection against data manipulation and disclosure by unauthorized users after permanent deactivation of the Flash Loader. Second, the functional requirement FMT_LIM.2/Loader limits the Flash Loader availability after the download has been finished by the user. The Flash Loader provides a final locking command which irreversibly terminates the Flash Loader availability. This command execution must be applied after user has finalized his download. As the functional requirements are met by the Flash Loader the objective is covered.

The objectives O.Ctrl_Auth_Loader and O.Prot_TSF_Confidentiality require that a trusted communication channel with an authorized user, a confidentiality protection during the download and authentication of the user data and access control for the usage of the Loader functionality are provided by the Loader.
Without successfully passing the authentication functionality of the Flash Loader no usage of the Flash Loader is possible. Passing the authentication successfully assigns also a user role to the current user.
The Flash Loader implements mutual authentication functionality and if successfully passing this mutual authentication, the TOE and the external user are established as trusted entities. Furthermore, the Flash Loader enforces the exchange of the download key by the user which provides clear separation from the hardware vendor and preserves confidentiality of the user data download. In addition, the Flash Loader preserves the integrity of the downloaded data against for example induced errors by hashing functionality. As the functional requirements are met by the Flash Loader the objective is covered.

Note regarding the Flash Loader:
The algorithm strength of the Flash Loader has not been cryptographic-analytically evaluated by the BSI, as Infineon has implemented a proprietary extension.
End of note.

7.4.1.4 Dependencies of Security Functional Requirements

The dependencies of the security functional requirements are defined and described in PP [9] section 6.3.2, with FDP_SDI.2, and with regard to the Flash Loader related security functional requirements, the description is given at the individual package chapters 7.2.3, 7.3.1 and 7.3.2.

FDP_ITT.1	FDP_IFC.1	FPT_ITT.1	FPT_PHP.3	FPT_FLS.1
FRU_FLT.2	FMT_LIM.1	FMT_LIM.2	FCS_RNG.1	FAU_SAS.1
FDP_SDI.2	FDP_SDC.1	FMT_LIM.1/Loader	FMT_LIM.2/Loader	FDP_ACC.1/Loader
FDP_ACF.1/Loader				

The security functional requirements FIA_API.1 and FTP_ITC.1 have no dependencies.
The security functional requirements FIA_API.1, FMT_LIM.1/Loader, FMT_LIM.2/Loader, FTP_ITC.1, FDP_UCT.1, FDP_UIT.1, FDP_ACC.1/Loader and FDP_ACF.1/Loader apply only at TOE products which are delivered with activated Flash Loader.

Further dependencies of security functional requirements are given in following table:

Table 13 Dependency for cryptographic operation requirement

Security Functional Requirement	Dependencies	Fulfilled by Security Requirements
ACL-0		
FCS_COP.1/RSA-0	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5	FCS_CKM.1/RSA-0
	FCS_CKM.6	see comment 2
FCS_CKM.1/RSA-0	FCS_CKM.2 or FCS_CKM.5 or FCS_COP.1	FCS_COP.1/RSA-0
	[FCS_RBG.1 or FCS_RNG.1	FCS_RNG.1/*
	FCS_CKM.6	see comment 2
	FCS_CKM.6	see comment 2
FCS_COP.1/ECDSA-0	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5	FCS_CKM.1/EC-0
	FCS_CKM.6	see comment 2
FCS_CKM.1/EC-0	FCS_CKM.2 or FCS_CKM.5 or FCS_COP.1	FCS_COP.1/ECDSA-0 FCS_COP.1/ECDH-0
	FCS_RBG.1 or FCS_RNG.1	FCS_RNG.1/*
	FCS_CKM.6	see comment 2

Security Requirements (ASE_REQ)

Security Functional Requirement	Dependencies	Fulfilled by Security Requirements
FCS_COP.1/ECDH-0	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5	FCS_CKM.1/EC-0
	FCS_CKM.6	see comment 2
CCL		
FCS_COP.1/CCL	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5	see comment 4
	FCS_CKM.6	FCS_CKM.6/CCL
FCS_CKM.1/CCL	None	see comment 4
FCS_CKM.6/CCL	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1or FCS_CKM.5	see comment 4
Hardware and Firmware (all alternative versions)		
FCS_COP.1/TDES	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5	see comment 1
	FCS_CKM.6	FCS_CKM.6/TDES
FCS_CKM.6/TDES	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1or FCS_CKM.5	see comment 1
FCS_COP.1/AES	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5	see comment 1
	FCS_CKM.6	FCS_CKM.6/AES
FCS_CKM.6/AES	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1or FCS_CKM.5	see comment 1
FPT_TST.2	No dependencies	N/A
FDP_ACC.1	FDP_ACF.1	FDP_ACF.1
FDP_ACF.1	FMT_MSA.3	FMT_MSA.3
	FDP_ACC.1	FDP_ACC.1
FMT_MSA.3	FMT_MSA.1	FMT_MSA.1

Security Requirements (ASE_REQ)

Security Functional Requirement	Dependencies	Fulfilled by Security Requirements
	FMT_SMR.1	see comment 5
FMT_MSA.1	FDP_ACC.1 or FDP_IFC.1	FDP_ACC.1
	FMT_SMR.1	see comment 5
	FMT_SMF.1	FMT_SMF.1
FMT_SMF.1	None	N/A
FMT_LIM.1/Loader	FMT_LIM.2/Loader	FMT_LIM.2/Loader
FMT_LIM.2/Loader	FMT_LIM.1/Loader	FMT_LIM.1/Loader
FTP_ITC.1	None	see comment 3
FDP_UCT.1	[FTP_ITC.1 or FTP_TRP.1]	FTP_ITC.1
	[FDP_ACC.1 or FDP_IFC.1]	FDP_ACC.1/Loader
FDP_UIT.1	[FTP_ITC.1 or FTP_TRP.1]	FTP_ITC.1
	[FDP_ACC.1 or FDP_IFC.1]	FDP_ACC.1/Loader
FDP_ACC.1/Loader	FDP_ACF.1	FDP_ACC.1/Loader
FDP_ACF.1/Loader	FMT_MSA.3	see comment 3
	FDP_ACC.1/Loader	see comment 3

Comment 1:
The TOE does not provide functions to generate, import or derive keys for AES and TDES.
The respective dependencies FCS_CKM.1 or FCS_CKM.5 or FDP_ITC.1 or FDP_ITC.2 have to be fulfilled by the environment, because the TOE does not provide the accompanying functionality; i.e. generate and import keys.
That means that the environment shall meet the requirements FCS_CKM.1 or FCS_CKM.5 or FDP_ITC.1 or FDP_ITC.2 as defined in Common Criteria Part 2 [CCBook2], section 10.2 and 11.9.
End of comment.

Comment 2:

The cryptographic key destruction can be done by overwriting the key register interfaces or by software reset of the SCP which provides immediate zeroing of all SCP key registers. Please refer also to the application notes 41 and 42 in the PP [9].

The respective dependency FCS_CKM.6 has to be fulfilled by the environment because the TOE does not provide the functionality to delete keys. That mean, that the environment shall meet the requirement FCS_CKM.6 as defined in Common Criteria Part 2 [CCBook2]
End of comment.

Comment 3:
The inter-TSF trusted channel SFR FTP_ITC.1 has no dependency and is provided as main purpose by the Flash Loader. The Flash Loader provides a distinct and independent communication channel with authenticated end points and protection from modification or disclosure.

The dependency FMT_MSA.3 introduced by the component FDP_ACF.1/Loader is considered to be not required, because the security attributes enforcing the Loader SFP are fixed by the IC manufacturer and no new objects under the control of the Loader SFP are created. Claim 371 of PP [9] applies.
End of comment.

Comment 4:
The security of the cryptographic functions FCS_CKM.1/CCL, FCS_CKM.6/CCL and FCS_COP.1/CCL relies on the secure use of the CIPURSE™ CL: This means that it is essential on user side that the common secret is generated and stored in an appropriate way and that integrity and confidentiality of this user secret is maintained.
These preconditions are treated in the PP [9] section 3.1 claims 67 and 68.
The key destruction FCS_CKM.6/CCL applies only for the keys generated during and for a session and not for the common secret.
End of comment.

Comment 5:
The dependency FMT_SMR.1 introduced by the two components FMT_MSA.1 and FMT_MSA.3 is considered to be satisfied because the access control specified for the intended TOE is not role-based but enforced for each subject. Therefore, there is no need to identify roles in form of a security functional requirement FMT_SMR.1.
End of comment.

7.4.2 Rationale of the Assurance Requirements

The chosen assurance levels are EAL5 and EAL6 augmented. In section 7.3.1.1 the assurance components are shown as well as the augmentations. The augmentations are in compliance with the Protection Profile [9].
These assurance levels are required for this type of TOE since it is intended to defend against **highly sophisticated attacks** without protective environment over a targeted long life time. Thereby, the TOE must withstand attackers with high attack potential, which is achieved by fulfilling the assurance class AVA_VAN.5. In order to provide a meaningful level of assurance and that the TOE provides an adequate level of defense against such high potential attacks, the evaluators have access to all information regarding the TOE including the TSF internals, the low level design and source code including the testing of the modular design. Additionally the mandatory technical document “Application of Attack Potential to Smartcards” [14] shall be taken as a basis for the vulnerability analysis of the TOE.
Due to the targeted long life time of the Infineon Technologies products, a comprehensive flaw remediation process and database is in place to maintain the TOE also in future. State of the art systematic flaw remediation requires ALC_FLR.3
This evaluation assurance package was selected to permit a developer gaining maximum assurance from positive security engineering based on good commercial practices as well as the assurance that the TOE is maintained during its targeted life time. The evaluation assurance packages follow the EAL5 and 6 assurance classes as given in Common Criteria Part 5 [CCBook5].

8 TOE Summary Specification (ASE_TSS)

The product overview is given in section 2.1. In the following the Security Features are described and the relation to the security functional requirements is shown.
The TOE is equipped with following Security Features to meet the security functional requirements:

- SF_DPM Device Phase Management
- SF_PS Protection against Snooping
- SF_PMA Protection against Modification Attacks
- SF_PLAProtection against Logical Attacks
- SF_CS Cryptographic Support

The following description of the Security Features is a complete representation of the TSF.

8.1 SF_DPM: Device Phase Management

The life cycle of the TOE is split-up in several phases: Chip development and production (covering phases 2, 3 and 4) and the final use (phase 4 to 7). This is a rough split-up from TOE point of view. These phases are implemented in the TOE as test mode (phase 3) and user mode (phase 4-7).

In addition, chip identification modes are implemented being active in all TOE life cycle phases. The chip identification data (O.Identification) is stored in a in the not changeable and access protected configuration page area of the SOLID FLASH™ NVM. In the same area further TOE configuration data is stored. In addition, user initialization data can be stored in the SOLID FLASH™ NVM during the production phase as well. During this first data programming, the TOE is still in the secure environment and in Test Mode.
The covered security functional requirement is FAU_SAS.1 “Audit storage”.

During start-up of the TOE the decision for one of the various operation modes is taken in dependency of the corresponding phase identifiers. The decision of accessing a certain mode is defined as phase entry protection as defined conditions have to be met. The phases follow also a defined and flow protected sequence. The sequence of the phases is protected by means of authentication.
The covered security functional requirements are FMT_LIM.1”Limited Capabilities” and FMT_LIM.2 “Limited availability”.

During the production phase (phase 3 and 4) or after the delivery to the user (phase 5 or phase 6), the TOE provides the possibility to download and finalize the user data. Using the download functionality of the Flash Loader requires passing a successful authentication process and a key exchange. The key exchange ensures that only the user defines the encryption key which is used during the download. If the conditions are met, user code and data can be flashed into the SOLID FLASH™ NVM area as specified by the associated control information of the Flash Loader software. The download into the chip is done in an encrypted way only.
The usage of the Flash Loader is only possible after a successful mutual authentication process of the external entity and the TOE itself.
In case the user has ordered TOE derivatives without Flash Loader, the software download by the user (phase 5 or phase 6) is disabled and all user data of the Composite TOE has been flashed on the TOE at Infineon premises. In both cases the integrity of the loaded data is checked with a hashing process. The data to be loaded is transferred always in encrypted form.
After finalizing the load operation and prior delivery to the end-user, the Flash Loader shall be permanently deactivated. The permanent deactivation is named locking and is a user obligation documented in the user guidance. This locking removes any possibility to use or reactivate the Flash Loader and provides a clear separation between the firmware-domain – regardless of the version in use - and the user software domain regarding downloads: Software updates after delivery to the end user are exclusively in the responsibility of the user software.
The covered security functional requirement are FMT_LIM.1/Loader “Limited capabilities”, FMT_LIM.2/Loader “Limited availability-Loader”, FIA_API.1 “Authentication Proof of Identity”, FTP_ITC.1

"Inter-TSF trusted channel", FDP_UCT.1 "Basic data exchange confidentiality", FDP_UIT.1 "Data exchange integrity, FDP_ACC.1/Loader "Subset Access Control – Loader" and FDP_ACF.1/Loader "Security Attribute based Access Control – Loader".
These Flash Loader related security functional requirements apply only at TOE products coming with activated Flash Loader enabled for user data download by the user. In other cases the Flash Loader is not available anymore and the user data download is completed.

During operation within a selected life cycle phase the accesses to memories are granted by the MMU controlled access rights and related privilege levels. The TOE operates always in a dedicated life cycle phase. The covered security functional requirements are FDP_ACC.1 "Subset Access Control, FDP_ACF.1 "Security Attribute based Access Control and FMT_MSA.1 "Management of Security Attributes.

In addition, during each start-up of the TOE the address ranges, belonging memory keys and access rights are initialized by the BOS with predefined values. The covered security functional requirement is FMT_MSA.3 "Static Attribute Initialization".

The TOE clearly defines access rights and privilege levels in conjunction with the appropriate key management in dependency of the firmware – regardless of the version in use - or software to be executed. By this clearly defined management functions are implemented, enforced by the MMU, and the covered security functional requirement is FMT_SMF.1 "Specification of Management Functions".

During the testing phase in production within the secure environment the entire SOLID FLASH™ NVM is deleted. The covered security functional requirement is FPT_PHP.3 "Resistance to physical attack".
Each operation phase is protected by means of authentication and encryption. The covered security functional requirements are FDP_ITT.1 "Basic internal Transfer Protection" and FPT_ITT.1 "Basic internal TSF Data Transfer Protection".

8.1.1 Listing of SFRs implemented by SF_DPM Device Phase Management

Table 14 Security Functional Requirements covered by SF_DPM Device Phase Management

1. FAU_SAS.1 Audit storage
2. FMT_LIM.1 Limited Capabilities
3. FMT_LIM.2 Limited availability
4. FDP_ACC.1 Subset access control
5. FDP_ACF.1 Security attribute based access control
6. FMT_MSA.1 Management of security attributes
7. FMT_MSA.3 Static attribute initialization
8. FMT_SMF.1 Specification of Management Functions
9. FPT_PHP.3 Resistance to physical attack
10. FDP_ITT.1 Basic internal transfer protection
11. FPT_ITT.1 Basic internal TSF data transfer protection
12. FMT_LIM.1/Loader Limited capabilities
13. FMT_LIM.2/Loader Limited availability-Loader
14. FIA_API.1 Authentication Proof of Identity
15. FTP_ITC.1 Inter-TSF trusted channel
16. FDP_UCT.1 Basic data exchange confidentiality
17. FDP_UIT.1 Data exchange integrity
18. FDP_ACC.1 Subset access control
19. FDP_ACC.1/Loader Subset access control – Loader

20. FDP_ACF.1/Loader Security attribute based access control – Loader

8.2 SF_PS: Protection against Snooping

Memories
All contents of all memories of the TOE are encrypted on chip to protect against data analysis on stored data as well as on internally transmitted data. There is no plain data on the chip and the encryption of the memories cover also the stored error detection values, and with regard to the SOLID FLASH™ NVM, also the error correction values. Induced errors will lead with very high probability to an encryption and/or decryption fail in the MED and to the appropriate action.
In contrast to the linear virtual address range, the physical SOLID FLASH™ NVM pages are transparently mapped to different physical address ranges and controlled by the MMU. Thus the data is continuously protected during transfer and storage by encryption and the mapping means of the address ranges. On top this, the address scrambling provides a completely user transparent chip-individual physical memory layout of the SOLID FLASH™ NVM. By that even in the unlike event of two equal TOE derivatives coming with equal software, equal MMU settings and equal MMU mapping, finding an equal piece of data – for example a previously identified target - at the equal physical location in the SOLID FLASH™ NVM on the second chip is extremely unlikely and from attackers perspective not practical. This address scrambling is entirely independent from the user software and the MMU.
MED
The encryption of the memories is performed by the MED with a proprietary cryptographic algorithm and with a complex and dynamic key management providing protection against cryptographic analysis attacks. This includes also the possibility of user chosen keys for SOLID FLASH™ NVM areas. The only key remaining static over the product life cycle is the specific ROM key, changing in case of a mask change only. The few keys which have to be stored on the chip, for example the user chosen key and the chip specific ROM key, are protected against read out. Note that the ROM contains the firmware only and no user data of the Composite TOE.
The covered security functional requirements are FPT_PHP.3 “Resistance to physical attack”, FDP_IFC.1 “Subset information flow control”, FPT_ITT.1 “Basic internal TSF data transfer protection”, FDP_ITT.1 “Basic internal transfer protection”FPT_FLS.1, “Failure with preservation of secure state” and FDP_SDC.1 “Stored data confidentiality”.
Peripheral Bus
In addition the data transferred over the memory bus to and from (bi-directional encryption) the CPU, Co-processors, the special SFRs and selected peripheral devices connected to the peripheral bus are encrypted automatically with a dynamic key change. More information is given in the confidential Security Target [8].
The covered security functional requirements are FPT_PHP.3 “Resistance to physical attack”, FDP_IFC.1 “Subset information flow control”, FPT_ITT.1 “Basic internal TSF data transfer protection”, FDP_ITT.1 “Basic internal transfer protection and FPT_FLS.1 “Failure with preservation of secure state”.
CPU
The TOE computes and handles data even in the core only encrypted respectively masked. At no time plain data is processed – except when communicating to the outer world. By this plain data is only available at the interface modules. The dual CPU computes entirely maskedMore information is given in the confidential Security Target [8].
The covered security functional requirements are FPT_PHP.3 “Resistance to physical attack”, FDP_IFC.1 “Subset information flow control”, FPT_ITT.1 “Basic internal TSF data transfer protection”, FDP_ITT.1 “Basic internal transfer protection and FPT_FLS.1 “Failure with preservation of secure state”.
SCP/Cache
The symmetric cryptographic co-processor (SCP) is also entirely masked. The Cache being in ongoing use during core operation is also entirely encrypted. More information is given in the confidential Security Target [8]. The encryption covers the data processing policy and FDP_IFC.1 “Subset information flow control“. The covered security functional requirements are FPT_PHP.3, FDP_IFC.1, FPT_ITT.1 and FDP_ITT.1.

MMU
In addition to their protection during processing of code and data, their storage in the SOLID FLASH™ NVM is protected with a further mean too: Even if users operate with direct and static addressing for storing their secrets, the addresses are always translated to virtual addresses. More information is given in the confidential Security Target [8].
The covered security functional requirements are FPT_PHP.3 “Resistance to physical attacks”, FPT_ITT.1 “Basic internal TSF data transfer protection”, FDP_ITT.1 “Basic internal transfer protection” and FPT_FLS.1 “Failure with preservation of secure state”.
Proprietary CPU
A proprietary dual CPU implementation with a non-public bus protocol renders analysis very complicated and time consuming. Besides the proprietary structures also the internal timing behavior is proprietary and by this aggravating significantly the analysis in addition. Important parts of the chip are especially designed to counter leakage or side channel attacks like DPA/SPA or EMA/DEMA. Therefore, even the physical data gaining is difficult to perform, since timing and current consumption is almost independent of the dynamically encrypted, respectively masked and/or randomized data.
Important parts of the chip are especially designed to counter leakage or side channel attacks like DPA/SPA or EMA/DEMA. Therefore, even identifying and collecting physical data is difficult to perform, since timing and current consumption are almost independent of the processed data, as those are protected by a bunch of other internal protecting means.
Synthesis
In the design a number of components are automatically synthesized and mixed up to disguise and complicate analysis. The covered security functional requirement is FPT_PHP.3 “Resistance to physical attack”.
Secure Wiring/I^2 shield
A further protective design method used is secure wiring. All security critical wires have been identified and protected by special routing measures against probing. Additionally, artificial shield lines are implemented and mixed up with normal signal lines required for chip operation, rendering probing attacks with high feasibility to not practical. This provides the so called intelligent implicit active shielding “I^2-shield”.
The covered security functional requirements are FPT_PHP.3 “Resistance to physical attack”, FPT_ITT.1 “Basic internal TSF data transfer protection” and FDP_ITT.1”Basic internal transfer protection”.
FSE
A low system frequency sensor FSE is implemented to prevent the TOE from single stepping.
The covered security functional requirements are FPT_PHP.3 “Resistance to physical attack” and FPT_FLS.1 “Failure with preservation of secure state”.

8.2.1 Listing of SFRs implemented by SF_PS Protection against Snooping

Table 15 Security Functional Requirements covered by SF_PLA Protection against Snooping

1. FPT_PHP.3 Resistance to physical attack
2. FDP_SDC.1 Stored data confidentiality
3. FDP_IFC.1 Subset information flow control
4. FPT_ITT.1 Basic internal TSF data transfer protection
5. FDP_ITT.1 Resistance to physical attack
6. FPT_FLS.1 Failure with preservation of secure state

8.3 SF_PMA: Protection against Modifying Attacks

First of all we can say that all security mechanisms effective against snooping **SF_PS** apply also here since a reasonable modification of data is almost impossible on dynamically encrypted, masked, scrambled, transparently relocated, randomized and topologically protected hardware. Due to this the covered security functional requirements are FPT_PHP.3, FDP_SDC.1, FDP_IFC.1, FPT_ITT.1, FDP_ITT.1 and FPT_FLS.1.

The TOE is equipped with an error detection code (EDC) which covers the memory system of RAM, ROM and SOLID FLASH™ NVM and includes also the MED and MMU. Thus introduced failures could be detected and the appropriate action is taken. In terms of single bit errors in the SOLID FLASH™ NVM, the errors are also automatically corrected. This contributes to FDP_SDI.2 Stored data integrity monitoring and action and FRU_FLT.2 Limited fault tolerance. In order to prevent accidental bit faults during production in the ROM an EDC value is calculated and stored as well.
The error detection and partly correction means protect against physical and provide the appropriate reaction in terms of induced errors and faults. The covered security functional requirements are FRU_FLT.2 Limited fault tolerance, FPT_PHP.3 Resistance to physical attack, and FDP_SDI.2 Stored data integrity monitoring and action.

The Cache integrity is protected by a different method as the other memories and provides also error detection and appropriate reaction in case of induced errors. This contributes to FDP_SDI.2 Stored data integrity monitoring and action.

The TOE is protected against fault and modifying attacks. The core provides the functionality of double-computing and result comparison of all tasks to detect incorrect calculations. The detection of an incorrect calculation is stored and the TOE enters a defined secure state which causes the chip internal reset process. The implementation of the dual CPU computing on the same data is by this one of the most important security features of this platform. As also the results of both CPU parts are compared at the end, a fault induction of modifying attacks would have to be done on both CPU parts at the correct place with the correct timing – despite all other countermeasures like dynamic masking, encryption and others. The comparison and the register files are also protected by various measures. The covered security functional requirements are FPT_FLS.1 Failure with preservation of secure state, FPT_PHP.3 Resistance to physical attack, FPT_ITT.1 Basic internal TSF data transfer protection and FDP_ITT.1 Basic internal transfer protection.

During start up, the BOS performs various configurations and subsystem tests. After the BOS has finished, the operating system or application can call the RMS function User Mode Security Life Control (UMSLC) test. This function provides the testing of the security features enabled to generate an alarm and can be released actively by the user software during normal chip operation any time. Attempts to modify the security features will be detected from this test and lead to the appropriate reaction. The covered security functional requirement is FPT_TST.2 Subset TOE security testing.

In the case that a physical manipulation or a physical probing attack is detected, for example by the intelligent intrinsic shield (I^2), the processing of the TOE is immediately stopped and the TOE enters a secure state called security reset. The I^2 shield is also part of the UmSLC. The covered security functional requirements are Failure with preservation of secure state, FPT_FLS.1 Failure with preservation of secure state, FPT_PHP.3 Resistance to physical attack and FPT_TST.2 Subset TOE security testing.

As physical effects or manipulative attacks may also address the program flow of the user software, a watchdog timer and a check point register are implemented. These features allow the user to check the correct processing time and the integrity of the program flow of the user software.
Another measure against modifying and perturbation respectively differential fault attacks (DFA) is the implementation of backward calculation in the SCP. By this induced errors are discovered.
The covered security functional requirements are FPT_FLS.1 Failure with preservation of secure state, FDP_IFC.1 Subset information flow control, FPT_ITT.1 Basic internal TSF data transfer protection, FDP_ITT.1 Basic internal transfer protection and FPT_PHP.3 Resistance to physical attack.

All communication via the busses is in addition protected by a monitored hardware handshake. If the handshake was not successful an alarm is generated.
The covered security functional requirements are FPT_FLS.1 Failure with preservation of secure stateand FPT_PHP.3 Resistance to physical attack.

The virtual memory system and privilege level model are enforced by the MMU. This controls the access rights throughout the TOE. There is a clear differentiation within the defined privilege levels. Addresses and privilege level must match and induced errors and/or manipulation lead to appropriate error message and action. The covered security functional requirements are FDP_ACC.1 Subset access control, FDP_ACF.1 Security attribute based access control, FMT_MSA.1 Management of security attributes, FMT_MSA.3 Static attribute initialization and FMT_SMF.1 Specification of Management Functions.

All the measures of controlling the access rights, checking the integrity of data and code, the coverage of the integrity protecting values by means of encryption, the continuously masked calculation and operation stands for the Integrity Guard. The implemented measures interact like a gearing mechanism and by that an induced error will be discovered with very high feasibility followed by the appropriate reaction. While single bit faults may be corrected automatically, other faults which cannot be corrected lead to an alarm, and in case of security critical detections a security alarm and reset is generated. The covered security functional requirement is FPT_FLS.1 Failure with preservation of secure state.

If the hardware support library HSL – in all versions - comes with the TOE and the Low Level Driver (LLD) and/or the In-Place-Update (IPU) functionality are used in certain configurations as outlined in the version specific user guidance document HSL [15], the TOE behavior is protected against sudden power off events and its behavior is tearing safe. Induced faults will be both detected and repaired, or the system enters a secure state. This protection is additionally claimed for the IWR method as offered in the version v03.11.8339 and v03.12.8812.
This means in detail that the SFRs Failure with preservation of secure state FPT_FLS.1 and FPT_PHP.3 Resistance to physical attack as well as FDP_SDI.2 Stored data integrity and action are covered for the methods LLD and IPU for the HSL versions v01.22.4346 and v02.01.6634 if the user follows the recommendation and configuration of the user guidance.
In addition, the equal SFRs Failure with preservation of secure state, Resistance to physical attacks and Stored data integrity and action, FPT_FLS.1, FPT_PHP.3 and FDP_SDI.2 are claimed for versions v03.11.8339 and v03.12.8812. These versions claim the named SFR for the methods LLD, IPU and also IWR if the user follows the recommendation and configuration of the user guidance.
In all versions, tearing safe implements an atomicity in the concerned operations, resulting that if the process of writing to the SOLID FLASH™ NVM is interrupted by an accidental or intentional power loss or reset, the SOLID FLASH™ NVM data will be either the original data or will be in the new data. The interruption possibly involves some recovery steps that have to be taken before the data is accessed. After

successful completion of the concerned operations the relevant data are always in a defined status. If errors are detected during the processing a secure state is entered.
The covered security functional requirements are FPT_FLS.1 Failure with preservation of secure state, FPT_PHP.3 Resistance to physical attack and FDP_SDI.2 Stored data integrity and action.

8.3.1 Listing of SFRs implemented by SF_PMA Protection against Modifying Attacks

Table 16 Security Functional Requirements covered by SF_PMA Protection against Modifying Attacks

1. FPT_PHP.3 Resistance to physical attack
2. FDP_IFC.1 Subset information flow control
3. FPT_ITT.1 Basic internal TSF data transfer protection
4. FDP_ITT.1 Basic internal transfer protection
5. FMT_MSA.1 Management of security attributes
6. FMT_MSA.3 Static attribute initialization
7. FMT_SMF.1 Specification of Management Functions
8. FDP_ACC.1 Subset access control
9. FDP_ACF.1 Security attribute based access control
10. FRU_FLT.2 Limited fault tolerance
11. FPT_TST.2 Subset TOE security testing
12. FDP_SDI.2 Stored data integrity monitoring and action
13. FPT_FLS.1 Failure with preservation of secure state

8.4 SF_PLA: Protection against Logical Attacks

The memory access control of the TOE uses a memory management unit (MMU) to control the access to the available physical memory by using virtual memory addresses and to segregate the code and data to a privilege level model. The MMU controls the address permissions of up to seven privileged levels and gives the software the possibility to define different access rights for the user available privileged levels. The address permissions of the privilege levels are controlled by the MMU. In case of an access violation the MMU will trigger a reset and then a trap service routine can react on the access violation. The policy of setting up the MMU and specifying the memory ranges for the privilege levels – with the exception of the IFX level - is defined from the user software (OS). The privilege levels 0, 1 and 2 are reserved for TOE internal operations. The privilege levels 3 and 4 are reserved for operation systems and the privilege levels 5, 6 and 7 are reserved for applications.
Therefore, the TOE provides support for secure separation of memory areas covering the security functional requirements FPT_PHP.3 “Resistance to physical attack”, FDP_ACC.1 “Subset access control”, FDP_ACF.1 “Security attribute based access control”, FMT_MSA.1 “Management of security attributes”, FMT_MSA.3 “Static attribute initialisation” and FMT_SMF.1 “Specification of Management functions”.

The TOE provides the possibility to protect the property rights of user code and data by the encryption of the SOLID FLASH™ NVM areas with a specific key defined by the user. Due to this key management FDP_ACF.1 is fulfilled. In addition, each memory present on the TOE is encrypted using either mask specific or chip individual or even session keys, assigned by a complex key management. Induced errors are recognized by the Integrity Guard concept and lead to an alarm with high feasibility. In case of security critical errors a security alarm is generated and the TOE ends up in a secure state. The covered security functional requirements are FPT_PHP.3 “Resistance to physical attack”, FPT_ITT.1 “Basic internal transfer protection”, FDP_IFC.1 “Subset information flow control” and FPT_FLS.1 “Failure with preservation of secure state”.

Beside the access protection and key management, also the use of illegal operation code is detected and will release a security reset. The covered security functional requirements FDP_ITT.1 “Basic internal transfer protection” and FPT_FLS.1 “Failure with preservation of secure state”.

8.4.1 Listing of SFRs implemented by SF_PLA Protection against Logical Attacks

Table 17 Security Functional Requirements covered by SF_PLA Protection against Logical Attacks

1. FDP_ACC.1 Subset access control
2. FDP_ACF.1 Security attribute based access control
3. FMT_MSA.1 Management of security attributes
4. FMT_MSA.3 Static attribute initialisation
5. FPT_PHP.3 Resistance to physical attack
6. FPT_ITT.1 Basic internal transfer protection
7. FDP_IFC.1 Subset information flow control
8. FPT_FLS.1 Failure with preservation of secure state
9. FMT_SMF.1 Specification of Management functions

8.5 SF_CS: Cryptographic Support

The TOE is equipped with several hardware accelerators and software modules to support the standard symmetric and asymmetric cryptographic operations. This security function is introduced to include the cryptographic operation in the scope of the evaluation as the cryptographic function respectively mathematic algorithm itself is not used from the TOE security policy. On the other hand these functions are of special interest for the use of the hardware as platform for the software. The components are a cryptographic co-processor supporting the DES and AES algorithms and a combination of a co-processor and software library modules to support RSA cryptography, RSA key generation, ECDSA signature generation and verification, ECDH key agreement and EC public key calculation and public key testing.
Note that the additional function of the EC library, ECC_ADD, providing the primitive elliptic curve operations, does not add specific security functionality and that the according user guidance abbreviates the Elliptic Curve cryptographic functions with ECC.

Note 15:
The cryptographic libraries RSA, EC and the Toolbox library are delivery options. Therefore the TOE may come with free combinations of or even without these libraries. In the case of coming without one or any combination of the cryptographic libraries RSA and EC, the TOE does not provide the Additional Specific Security Functionality Rivest-Shamir-Adleman Cryptography (RSA) and/or Elliptic Curve Cryptography (EC). The Toolbox Library is no cryptographic library, provides no additional specific security functionality and thus the Toolbox library is not part of the TOE Security Functionality (TSF).
End of note.

Note 16:
This TOE can come with both crypto co-processors accessible, or with a blocked SCP or with a blocked Crypto2304T, or with both crypto co-processors blocked. The blocking depends on the user’s choice. No accessibility of the deselected cryptographic co-processors is without impact on any other security policy of the TOE; it is exactly equivalent to the situation where the user decides just not to use the cryptographic co-processors. In case the SCP is blocked, no AES and DES computation supported by hardware is possible. In case the Crypto2304T is blocked, no RSA and EC computation supported by hardware is possible. No accessibility of the deselected cryptographic co-processors is without impact on any other security policy of the TOE; it is exactly equivalent to the situation where the user decides just not to use the cryptographic co-

processors.
End of note.

Note 17:
The presence of the alternative cryptographic library CIPURSE™ CL is a delivery option.
Therefore the TOE may come with free combinations with the other libraries of or without these libraries.
In the case of coming without the CIPURSE™ CL the TOE does not provide the specific security functionality implemented by this software.
End of note.

8.5.1 Implementation of AES and TDES by the Symmetric Cryptographic Coprocessor SCP

8.5.1.1 Triple DES

The TOE supports the encryption and decryption in accordance with the specified cryptographic algorithm Triple Data Encryption Standard (TDES) with cryptographic key sizes of 112 and 168 bits meeting the standard:

National Institute of Standards and Technology (NIST), SP 800-67 [20]
ISO/IEC 18033-3 [30]

The TOE implements the following alternative block cipher modes for the user:

- the Electronic Codebook Mode (ECB),
- the Cipher Block Chaining Mode (CBC),
- the Blinding Feedback Mode (BLD),
- the Cipher Block Chaining Mode Message Authentication Code (CBC-MAC),
- the CBC-MAC- encrypt-last-block (CBC-MAC-ELB) and
- the Recrypt Mode.

The CBC-MAC and CBC-MAC-ELB complies with the standard: *ISO/IEC 9797-1 Mac Algorithm 1 and 2 [32]*

The Recrypt Mode and the BLD are described in the hardware reference manual HRM [1], while the implementation of ECB, CBC and CFB follow the standard:

National Institute of Standards and Technology (NIST), SP 800-38A [21]

Note that the BLD follows also the standard, but in a masked way.

The key destruction as required by FCS_CKM.6 can be done by overwriting the key register interfaces or by software reset of the SCP which provides immediate zeroing of all SCP key registers.

Please consider also the statement of chapter 7.1.3.1.

The covered security functional requirements are FCS_COP.1/TDES and FCS_CKM.6/TDES.

8.5.1.2 AES

The TOE supports the encryption and decryption in accordance with the specified cryptographic algorithm Advanced Encryption Standard (AES) and cryptographic key sizes of 128 bit or 192 bit or 256 bit that meet the standard:

ISO/IEC 18033-3 [30]
FIPS 197 [31]

The TOE implements the following alternative block cipher modes for the user:

- The Electronic Codebook Mode (ECB),
- the Cipher Block Chaining Mode (CBC),
- the Blinding Feedback Mode (BLD),
- the Cipher Block Chaining Mode Message Authentication Code (CBC-MAC),
- the CBC-MAC- encrypt-last-block (CBC-MAC-ELB) and
- the Recrypt Mode.

The CBC-MAC and CBC-MAC-ELB complies with the standard:

ISO/IEC 9797-1 Mac Algorithm 1 and 2 respectively [32]

The Recrypt Mode and the BLD are described in the hardware reference manual HRM [1], while the implementation of ECB and CBC follow the standard:

National Institute of Standards and Technology (NIST) SP 800-38A [21]

The key destruction as required by FCS_CKM.6 can be done by overwriting the key register interfaces or by software reset of the SCP which provides immediate zeroing of all SCP key registers.
Please consider also the statement of chapter 7.1.3.1.
The covered security functional requirement is FCS_COP.1/AES and FCS_CKM.6/AES.

8.5.2 RSA Cryptographic Library

8.5.2.1 Encryption, Decryption, Signature Generation and Verification

The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm Rivest-Shamir-Adleman (RSA) and cryptographic key sizes *1024- 4096* bits that meet the following standards:

Encryption: 1. According to section 5.1.1 RSAEP in PKCS [22]: • Supported for $n < 2^{2048+64}$ • 5.1.1(1) not supported 2. According to section 8.2.2 IFEP-RSA in IEEE [29]: Supported for $n < 2^{2048+64}$
Decryption (with or without CRT): 1. According to section 5.1.2 RSADP in PKCS [22] for $u = 2$, i.e., without any (r_i, d_i, t_i), $i > 2$ • 5.1.2(1) not supported • 5.1.2(2.a) supported for $n < 2^{2048 + 64}$ • 5.1.2(2.b) supported for $p \times q < 2^{4096 + 128}$ • 5.1.2(2.b) (ii)&(v) not applicable due to $u = 2$ 2. According to section 8.2.3 IEEE [29]: • 8.2.1(I) supported for $n < 2^{2048 + 64}$ • 8.2.1(II) supported for $p \times q < 2^{4096 + 128}$ • 8.2.1(III) not supported
Signature Generation (with or without CRT): *1. According to section 5.2.1 RSASP1 in PKCS [22] for $u = 2$, i.e., without any (r_i, d_i, t_i), $i > 2$* • *5.2.1(1) not supported* • *5.2.1(2.a) supported for $n < 2^{2048 + 64}$* • *5.2.1(2b) supported for $p \times q < 2^{4096 + 128}$* • *5.2.1(2b) (ii)&(v) not applicable due to $u = 2$* *2. According to section 8.2.4 IFSP-RSA1 in IEEE [29]:* • *8.2.1(I) supported for $n < 2^{2048 + 64}$* • *8.2.1(II) supported for $p \times q < 2^{4096 + 128}$* • *8.2.1(III) not supported*
Signature Verification: *1. According to section 5.2.2 RSAVP1 in PKCS [22]:* *supported for $n < 2^{4096 + 128}$* • *5.2.2(1) not supported* *2. According to section 8.2.5 IEEE [29]:* • *Supported for $n < 2^{4096 + 128}$* • *8.2.5(1) not supported*

Please consider also the statement of chapter 7.1.3.1.

The covered security functional requirement is FCS_COP.1/RSA-0

8.5.2.2 Asymmetric Key Generation

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *RSA specified in PKCS#1 [22]* and specified cryptographic key sizes of *1024 – 4096 bits* that meet the following standard:

RSA key generation:
1. According to sections 3.1 and 3.2 in PKCS [22], for u = 2, i.e. without any (r_i, d_i, t_i), i > 2:
3.1 supported for $n < 2^{4096 + 128}$
3.2.(1) supported for $n < 2^{2048 + 64}$
3.2.(2) supported for $p \times q < 2^{4096 + 128}$
According to section 8.1.3.1 in IEEE [29]:
8.1.3.1(1) supported for $n < 2^{2048 + 64}$
8.1.3.1(2) supported for $p \times q < 2^{4096 + 128}$
8.1.3.1(3) supported for $p \times q < 2^{2048 + 128}$

The covered security functional requirements are FCS_CKM.1/RSA-0

8.5.3 Elliptic Curves Cryptographic Library

The security functional requirement covers the standard Brainpool [19] and NIST [26] Elliptic Curves. Note that numerous other side channel attack resistant curve types exist, which the user optionally can add in the composition certification process.
All curves are based on finite field GF(p) with value $p \in [2^{41-1}; 2^{521}[$ as well as curves based on a finite field $GF(2^n)$ with size $n \in [41-1; 521]$ are supported.

Signature Generation and Verification
The TSF shall perform signature generation and signature verification in accordance with a specified cryptographic algorithm ECDSA and cryptographic key sizes of 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 bits that meet the standards described in the SFR FCS_COP.1/ECDSA-0.

8.5.3.1 Asymmetric Key Generation

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm Elliptic Curve EC specified in [23], [27] and [29] and specified cryptographic key sizes 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 bits that meet the standards described in the SFR FCS_CKM.1/EC-0.
Asymmetric Key Agreement
The TSF shall perform elliptic curve Diffie-Hellman key agreement in accordance with a specified cryptographic algorithm ECDH and cryptographic key sizes 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 bits that meet the described in the SFR FCS_COP.1/ECDH-0.

8.5.3.1 ECC ADD point addition

The TOE further provides functions to perform a full ECC point addition ECC_ADD according to section 2.3.1 in BSI TR [38]. These functionalities are additionally assessed by the ITSEF performing evaluation.

8.5.4 Toolbox Library

The Toolbox provides the following basic long integer arithmetic and modular functions in software, supported by the cryptographic coprocessor: Addition, subtraction, division, multiplication, comparison, reduction, modular addition, modular subtraction, modular multiplication, modular inversion and modular exponentiation. No security relevant policy, mechanism or function is supported. The Toolbox library is deemed for software developers as support for simplified implementation of long integer and modular arithmetic operations.
The Toolbox does not cover security functional requirements and thus the Toolbox library is not part of the TOE Security Functionality (TSF).

8.5.5 CIPURSE™ Cryptographic Library

The order option CIPURSE™ Cryptographic Library (CCL) provides cryptographic functionality to implement a CIPURSE™ V2 conformant protocol [35-1].
This protocol provides a secure mutual authentication of two entities, namely the terminal (denoted as PCD = Proximity Coupling Device (CIPURSE™-compliant terminal)) and a smart card or a token in other form factors which is called PICC. PICC stands for Proximity Integrated Circuit Card (CIPURSE™-compliant card). Beside the mutual authentication, the protocol implements cryptographic measures to maintain the integrity of the transferred data and preserves in parallel the confidentiality of the transferred data.

The CIPURSE™ CL covers the following cryptographic security functional requirements as follows:
CCL in version v02.00.0004

- FCS_CKM.1/CCL Cryptographic key generation,
- FCS_CKM.6/CCL Cryptographic key destruction
- FCS_COP.1/CCL Cryptographic Operation - CCL Trusted Channel

The implemented cryptographic operation applies following standards:

- Federal Information Processing Standards Publication 197 [31]
- NIST Special Publication SP 800-38A [21]
- CIPURSE™ V2 Cryptographic Protocol [35-1] chapter 5.2 Session key Derivation
- CIPURSE™ V2 Cryptographic Protocol [35-1] chapter 6.2 Key Derivation for the first frame
- CIPURSE™ V2 Cryptographic Protocol [35-1] chapter 6.3 Integrity Protection
- CIPURSE™ V2 Cryptographic Protocol [35-1] chapter 6.4 Confidential Communication

8.5.6 Hybrid PTRNG

Random data is essential for cryptography as well as for security mechanisms. The TOE is equipped with a Hybrid Physical True Random Number Generator (HPTRNG, FCS_RNG.1). The random data can be used from the Smartcard Embedded Software and is also used from the security features of the TOE, i.e. masking. The HPTRNG implements various topological means, masked bus interface and is self-checking.
The produced genuine random numbers are available as a security service for the user and are also used for internal purposes. Together with the security guidelines in [6] the hybrid PTRNG operates in the following modes of operation and is conformant to the named classes:

- True Random Number Generation, meeting AIS31 PTG.2
- Hybrid Random Number Generation, meeting AIS31 PTG.3
- Deterministic Random Number Generation (DRNG) AIS31 DRG.3
- Key Stream Generation (KSG), stream cipher generation AIS31 DRG.2

The details of AIS31 are given in [13].

The Hybrid PTRNG implements protected a protected peripheral bus interface is a synthesized module and covers therefore FPT_PHP.3 “Resistance to physical attack”. The transferred random data are masked as well as other configuration data transferred over the peripheral bus. This covers FDP_ITT.1 “Basic internal transfer protection” and FPT_ITT.1 “Basic internal TSF data transfer protection”. The correct function of the Hybrid PTRNG is subject of internal self-testing and in case of errors a secure state is achieved to protect the user from random data with bad entropy. Therefore, the output of the Hybrid PTRNG is conformant to the above claimed classes or there is no random data output. This covers FPT_FLS.1 “Failure with preservation of secure state”.

The hybrid PTRNG covers the security functional requirements FCS_RNG.1/*, Random Number Generation”, FPT_PHP.3 “Resistance to physical attack”, FDP_ITT.1 “Basic internal transfer protection”, FPT_ITT.1 “Basic internal TSF data transfer protection” and FPT_FLS.1 “Failure with preservation of secure state”.

8.5.7 Listing of SFRs implemented by SF_CS “Cryptographic Support”

Table 18 Security Functional Requirements covered by SF_CS “Cryptographic Support”

1. FCS_COP.1/TDES Cryptographic Operation
2. FCS_CKM.6/TDES Cryptographic key destruction TDES
3. FCS_COP.1/AES Cryptographic Operation
4. FCS_CKM.6/AES Cryptographic key destruction AES
5. FCS_COP.1/RSA-0 Cryptographic Operation RSA the ACL version v2.09.002
6. FCS_CKM.1/RSA-0 Cryptographic key generation RSA for ACL version v2.09.002
7. FCS_COP.1/ECDSA-0 Cryptographic Operation ECDSA for ACL version v2.09.002
8. FCS_CKM.1/EC-0 Cryptographic key generation Elliptic Curve for ACL version v2.09.002
9. FCS_COP.1/ECDH-0 Cryptographic Operation ECDH for ACL version v2.09.002
10. FPT_PHP.3 Resistance to physical attack
11. FDP_ITT.1 Basic internal transfer protection
12. FPT_ITT.1 Basic internal TSF data transfer protection
13. FPT_FLS.1 Failure with preservation of secure state
14. FCS_RNG.1/TRNG True Random Number Generation
15. FCS_RNG.1/HRNG Hybrid Random Number Generation
16. FCS_RNG.1/DRNG Deterministic Random Number Generation
17. FCS_RNG.1/KSG Key Stream Generation
18. FCS_CKM.1/CCL Cryptographic key generation for version v02.00.0004
19. FCS_CKM.6/CCL Cryptographic key destruction – AES for version v02.00.0004
20. FCS_COP.1/CCL Cryptographic operation CCL Trusted Channel for version v02.00.0004

8.6 Assignment of Security Functional Requirements to TOE’s Security Functionality

The justification and overview of the mapping between security functional requirements (SFR) and the TOE’s security functionality (SF) is given in sections the sections above. The results are shown in Table 20. The security functional requirements are addressed by at least one relating security feature.
The various functional requirements are often covered manifold. As described above the requirements ensure that the TOE is checked for correct operating conditions and if a not correctable failure occurs that a stored secure state is achieved, accompanied by data integrity monitoring and actions to maintain the integrity although failures occurred. An overview is given in the table below.

Table 19 Mapping of SFR and SF

Security Functional Requirement	SF_DPM	SF_PS	SF_PMA	SF_PLA	SF_CS
By the ACL					
FCS_CKM.1/EC-0					X
FCS_CKM.1/RSA-0					X
FCS_COP.1/ECDH-0					X
FCS_COP.1/ECDSA-0					X
FCS_COP.1/RSA-0					X
By the CIPURSE™ CL					
FCS_CKM.1/CCL					X
FCS_COP.1/CCL					X
FCS_CKM.6/CCL					X
By the Hardware, Firmware (all alternative versions) and the HSLs (all alternative versions)					
FCS_CKM.6/AES (by SCP)					X
FCS_CKM.6/TDES (by SCP)					X
FCS_COP.1/AES (by SCP)					X
FCS_COP.1/TDES (by SCP)					X
FAU_SAS.1	X				
FCS_RNG.1/TRNG					X
FCS_RNG.1/HRNG					X
FCS_RNG.1/DRNG					X
FCS_RNG.1/KSG					X
FDP_ACC.1	X		X	X	
FDP_ACC.1/Loader	X				
FDP_ACF.1	X		X	X	
FDP_ACF.1/Loader	X				
FDP_IFC.1		X	X	X	
FDP_ITT.1	X	X	X	X	X
FDP_SDC.1		X	X		
FDP_SDI.2 (1)			X		
FDP_UCT.1	X				
FDP_UIT.1	X				
FIA_API.1	X				

Security Functional Requirement	SF_DPM	SF_PS	SF_PMA	SF_PLA	SF_CS
FMT_LIM.1	X				
FMT_LIM.1/Loader	X				
FMT_LIM.2	X				
FMT_LIM.2/Loader	X				
FMT_MSA.1	X		X	X	
FMT_MSA.3	X		X	X	
FMT_SMF.1	X		X	X	
FPT_FLS.1 (1)		X	X	X	X
FPT_ITT.1	X	X	X		X
FPT_PHP.3 (1)	X	X	X	X	X
FPT_TST.2			X		X
FRU_FLT.2			X		X
FTP_ITC.1	X				X

(1) All HSL versions implement tearing save behavior for the SOLID Flash™ NVM. Its features are close to physical SOLID Flash™ NVM behavior which is not part of the SPM. Therefore, the HSL functionality contributing to FPT_FLS:1, FPT_PHP.3 and FDP_SDI.2 is excluded from the SPM even though it contributes to the named SFRs which are listed and modelled in the ADV_SPM.1.1D definition.

8.7 Security Requirements are internally Consistent

For this chapter the PP [9] section 6.3.4 can be applied completely.
In addition to the discussion in section 6.3 of PP [9] the security functional requirement FCS_COP.1 is introduced. The security functional requirements required to meet the security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced also protect the cryptographic algorithms implemented according to the security functional requirement FCS_COP.1. Therefore, these security functional requirements support the secure implementation and operation of FCS_COP.1.
As disturbing, manipulating during or forcing the results of the test checking the security functions after TOE delivery, this security functional requirement FPT_TST.2 has to be protected. An attacker could aim to switch off or disturb certain sensors or filters and preserve the detection of his manipulation by blocking the correct operation of FPT_TST.2. The security functional requirements required to meet the security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced also protect the security functional requirement FPT_TST.2. Therefore, the related security functional requirements support the secure implementation and operation of FPT_TST.2.
The requirement FPT_TST.2 allows testing of some security mechanisms by the Smartcard Embedded Software after delivery. In addition, the TOE provides an automated continuous user transparent testing of certain functions.
The implemented privilege level concept represents the area based memory access protection enforced by the MMU. As an attacker could attempt to manipulate the privilege level definition as defined and present in the TOE, the functional requirement FDP_ACC.1 and the related other requirements have to be protected themselves. 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 area based memory access control function implemented according to the security functional requirement described in the security functional requirement FDP_ACC.1 with reference to the Memory Access Control Policy and details

given in FDP_ACF.1. Therefore, those security functional requirements support the secure implementation and operation of FDP_ACF.1 with its dependent security functional requirements.
The requirement FDP_SDI.2 enables for detection of integrity errors of data stored in memory and for the correction of one bit errors and/or taking further action. This meets the security objective O.Malfunction. The requirements FRU_FLT.2, FPT_FLS.1, and FDP_ACC.1 which also meet this objective are independent from FDP_SDI.2 since they deal with the observation of the correct operation of the TOE and not with the memory content directly.

9 Literature and References

Note that the final versions of these documents are defined at the end of the evaluation process and that the documents are listed in the certification report as well.

No.	Vers.	As of	Document Title
1	5.0	2019-06-13	16-bit Security Controller Family - V02, Hardware Reference Manual (HRM)
2	Rev. 3.6	2019-06-24	Production and Personalization, 16-bit Security Controller
3	Rev. 9.14	2019-12-03	16-bit Security Controller, 65 nm Technology, Programmer's Reference Manual (PRM)
4-0	v2.09.002	2026-09-02	ACL52-Crypto2304T-C65 Asymmetric Crypto Library RSA/ECC/Toolbox, 16-bit Security Controller, User Interface manual
5	2.0	2024-06-21	16-bit Security Controller, Crypto@2304T V3, User Manual
6	1.01-2479	2020-03-09	16-bit Security Controller - V02, Security Guidelines (SG)
7	Rev. 7.1	2019-11-05	16-bit Security Controller - V02, Errata Sheet
8	3.9	2026-09-02	The confidential Security Target for this TOE.
9	1.0	2014-01-13	Security IC Protection Profile PP-0084 "Security IC Platform Protection Profile with Augmentation Packages", BSI-CC-PP-0084-2014, available at https://www.bund.bsi.de
9.1	PP0084.02	2016-04-21	Secrétariat general de la defense et de la sécurité nationale, Agence nationale de la sécurité des systèmes d'information, PP0084: Interpretations (ANSSI-CC-CER/F/06.002)
CCBook1	CC:2022	2022-11	Common Criteria for Information Technology Security Evaluation, CC:2022, revision 1, November 2022 Part 1: Introduction and general model
CCBook2	CC:2022	2022-11	Common Criteria for Information Technology Security Evaluation, CC:2022, revision 1, November 2022 Part 2: Security functional components
CCBook3	CC:2022	2022-11	Common Criteria for Information Technology Security Evaluation, CC:2022, revision 1, November 2022 Part 3: Security assurance components
CCBook5	CC:2022	2022-11	Common Criteria for Information Technology Security Evaluation, Part 5: Pre-defined packages of security requirements, November 2022, CC:2022, Revision 1
CCTrans	CC:2022	2023-04	CCMC-2023-04-001, Transition Policy to CC:2022 and CEM:2022, 2023-04-20
CCErrata	CC:2022	2025-10	Errata and Interpretation for CC:2022 (Release 1) and CEM:2022 (Release 1), V1.2, 2025-10-15
CC_SPM	V1.0	2024-05	ADV_SPM.1 interpretation for [CC:2022] transition, V1.0, 2024-05
13	3.0	2013-05-15	Functionality classes and evaluation methodology for physical random number generators AIS31, Version 3.0, 2013-05-15, Bundesamt für Sicherheit in der Informationstechnik and belonging "A proposal for: Functionality classes for random number generators", Version 2.0, 2011-09-18, Wolfgang

Literature and References

No.	Vers.	As of	Document Title
			Killmann, T-Systems GEI GmbH, Werner Schindler, Bundesamt für Sicherheit in der Informationstechnik
14	2	2025-02	EUCC SCHEME STATE-OF-THE-ART DOCUMENT Application of attack potential to smartcards and similar devices
15-1	v03.12.8812	2018-11-23	HSL library for SLCx2, User Guidance
15-2	v03.11.8339	2018-07-12	HSL library for SLCx2, User Guidance
15-3	v02.01.6634	2017	Hardware Support Library for SLCx2 (HSL), User Guidance
15-4	v01.22.4346	2016	Hardware Support Library for SLCx2 (HSL), User Guidance
17	Rev. 1.6	2018-02-02	CIPURSE™ Crypto Library, CCLX2xCIP v02.00.0004, CIPURSE™ V2, Compliant to OSPT™ Alliance CIPURSE™ V2 Cryptographic Protocol, User Interface
18	2.0	n.a.	The CIPURSE™ V2 Revision 2.0 standard issued by the OSPT™ alliance. The standard consists of a set of documents as given in the CIPURSE™V2 Revision 2.0 Documentation Overview. Http://www.osptalliance.org/resources/documentation
19	RFC 5639	2010-03	RFC 5639, Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation, IETF Trust and the persons identified as the document authors, March 2010, http://www.ietf.org/rfc/rfc5639.txt
20	800-67 Rev. 2	2017-11	National Institute of Standards and Technology (NIST), Special Publication 800-67, Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, Technology Administration, U.S. Department of Commerce
21	SP800-38A	2001-12	National Institute of Standards and Technology(NIST), Technology Administration, US Department of Commerce, NIST Special Publication SP 800-38A (for AES and DES)
22	PKCS #1 v2.2	2012-10-27	PKCS #1 v2.2: RSA Cryptography Standard, RSA Laboratories
23	ANSI X.9.62	2005-11-16	American National Standard for Financial Services ANSI X9.62-2005, Public Key Cryptography for the Financial Services Industry, The Elliptic Curve Digital Signature Algorithm (ECDSA), American National Standards Institute
24	ANSI X.9.63	2011-12-21	American National Standard for Financial Services X9.63-2011, Public Key Cryptography for the Financial Services Industry: Key Agreement and Key Transport Using Elliptic Curve Cryptography, American National Standards Institute
25	German law	2009-08-14	Act on the Federal Office for Information Security (BSI-Gesetz - BSIG), Bundesgesetzblatt I p. 2821; BSIG Section 9, Para.4, Clause 2
26	FIPS Pub 186-4	2013-07	Federal Information Processing Standards Publication, FIPS PUB 186-4, Digital Signature Standard (DSS), U.S. Department of Commerce, National Institute of Standards and Technology (NIST)

Literature and References

No.	Vers.	As of	Document Title
27	ISO/IEC 14888-3	2006, published 2009-02-15	INTERNATIONAL STANDARD ISO/IEC 14888-3:2006, TECHNICAL CORRIGENDUM 2, Published 2009-02-15, Information technology — Security techniques — Digital signatures with appendix — Part 3: Discrete logarithm based mechanisms
28	ISO/IEC 11770-3	2008, published 2009-09-15	INTERNATIONAL STANDARD ISO/IEC 11770-3:2008, TECHNICAL CORRIGENDUM 1, Published 2009-09-15, Information technology — Security techniques —Key management — Part 3: Mechanisms using asymmetric techniques
29	IEEE 1363	2000-01-30 (approved)	IEEE Standard Specification for Public key Crytography, IEEE Standards Board. The standard covers specification for public key cryptography including mathematical primitives for secret value deviation, public key encryption and digital signatures and cryptographic schemes based on those primitives.
30	ISO/IEC 18033	2010	ISO/IEC 18033-3: 2010, Information technology – Security techniques - Encryption algorithms - Part 3: Block ciphers [18033]
31	FIPS 197	2001-11-26	Federal Information Processing Standards Publication 197, ADVANCED ENCRYPTION STANDARD (AES), U.S. DEPARTMENT OF COMMERCE / National Institute of Standards and Technology, November 26, 2001
32	ISO/IEC 9797-1	2011-03-01	ISO/IEC 9797-1: 2011, Information technology – Security techniques - Message Authentication Codes (MACs) Part 1 Mechanisms using a block cipher
33	1.0	2015-02-13	Nachweis der Einhaltung der Sicherheitsanforderungen für Chipkarten im Zulassungsverfahren der Deutschen Kreditwirtschaft (DK) (German only)
34	ISO/IEC 9798-2	2008-12-15	Information Technologies - Security Techniques - Entity Authentication, part 2: Mechanisms using symmetric encipherment algorithms, 3rd edition 2008-12-15
35-1	1.0	2012-09-28	CIPURSE™ V2 Cryptographic Protocol issued by the OSPT™ Alliance
35-2	1.0	2014-09-18	CIPURSE™ V2 Cryptographic Protocol issued by the OSPT™ Alliance with Errata and Precision List
36	PCBC mode	1996	Bruce SCHNEIER, Applied Cryptography, Second Edition, John Wiley & Sons,1996
38	TR03111 v2.10	2018-06-01	Technical Guideline BSI TR-03111, Elliptic Curve Cryptography, Federal Office for Information Security – Bundesamt für Sicherheit in der Informationstechnik
39	FIPS Pub 180-4	August 2015	Information Technology Laboratory, National Institute of Standards and Technology(NIST), Federal Information Processing Standards Publication Secure Hash Standard (SHS)
40	FIPS SP 800-22 Revision 1a	2010 April	National Institute of Standards and Technology(NIST), Technology Administration, US Department of Commerce, Special Publication, A Statistical Test Suite for Random and

Literature and References

No.	Vers.	As of	Document Title
			Pseudorandom Number Generators for Cryptographic Applications

10 Hash Signatures of Cryptographic Libraries

Following listings document the hash signatures of the respective optional cryptographic library software version. For convenience purpose several hash algorithms were used.

10.1 ACL–0: RSA, EC, Toolbox Version v2.09.002

ACL52-Crypto2304T-C65-base.lib:
MD5 cf5d71a1810bd1399f675054e2cb118f
SHA1 80a00ff2f21a5a9ac7591bacf3d9aa436e26180b
SHA256 ad562f9fb45afa42b76f82fc9aee66c07d21ada5241afb5fa074176cad8b24db

ACL52-Crypto2304T-C65-ecc.lib:
MD5 cf403b6bfbac9a8e0c3000b50f4bafde
SHA1 22aeda2019b0a7650bd00e12f4b367ea109db0bc
SHA256 1324761a47047478523144c7675c9b26aeef96e5530c669a07a248ee301dab88

ACL52-Crypto2304T-C65-rsa2k.lib:
MD5 3c4840b63bd6fea8d7713d64a9e06c82
SHA1 9b78ce2a8afb2a4e758747f2a7282803a6cd6205
SHA256 869558c6c961724e53418cbac0814756a8a90bce55946177705ae31231e8f12b

ACL52-Crypto2304T-C65-rsa4k.lib:
MD5 a08a0059b28b419311cf780cb5b633b1
SHA1 3e662d9efeba26eea6a32bf915af69470f0e3b77
SHA256 167ec2b91f68c1d78ec18e6eb14b95c6c6b12c3a99be441a66223b00b3e6a2ea

ACL52-Crypto2304T-C65-toolbox.lib:
MD5 bcbf568ca1720f1002db470fed08a81a
SHA1 088b5d323016259074d602e8ae87c33dc48e5045
SHA256 74a1b1811006a53c694a28035100a281abdf9f609cde749de61311245b03af8c

10.2 HW- Support Libraries HSL in four versions

HSL - Hardware Support Library v03.12.8812:

MD5	2acc21f62c007f11b111eec3c2ddf9e2
SHA-1	b94276696231596d7c9a13d2f025d1f295781d3c
SHA-256	d7a474169bd457b29ac74b2676af3658c2b1339381c2f9cf13aab25f3f9cab93

HSL - Hardware Support Library v03.11.8339:

MD5	39183d4e35a7a87006a90ab1d962c65e
SHA-1	623b62fc003acdb0765a3f524bbc055e72e0760a
SHA-256	d54e43f6c977d17a1b4af1bf6fada7325891a31e5a73f124dc0116a90e84ec40

HSL - Hardware Support Library v02.01.6634:

MD5	fce6789dea3d49c0df32c05930ed3ff7
SHA-1	c6d2d3be94fdd539835cf2fb229a4e478a144b8e
SHA-256	12bf22478b88acf16344a970e4a42f5ebc2fe71d85c36a2be4895a1318862114

HSL – Hardware Support Library v01.22.4346:

MD5	57b0eb1f5e54922be46218431da53e4d
SHA-1	3cb599186885e3e167295bf7db583652d9e143b9
SHA-256	f898ac7ad4ba43732ec05e9ab7de752811c2a9553c078b5187ccba8b2a67843b

10.3 CIPURSE™ Cryptographic Library (CCL) in version v02.00.0004

Dual mode library files hash details:

MD5	30CDE00190AF243ACFE19E6D7BDA0907
SHA1	A74A592FA5EC162C3CA01B038AB77ABD316123AF
SHA256	921DA4663A0A231152704A46FAB5F567309791585E1C06D6517162F7E45D7742

11 List of Abbreviations

ACL	Asymmetric Cryptographic Library
AES	Advanced Encryption Standard
AIS31	Anwendungshinweise und Interpretationen zu ITSEC und CC: Funktionalitätsklassen und Evaluationsmethodologie für physikalische Zufallszahlengeneratoren
API	Application Programming Interface
APDU	Application Protocol Data Unit
BOS	Boot-up Software
BSI	German: Bundesamt für Sicherheit in der Informationstechnik English: Federal Office for Information Security
CC	Common Criteria
CI	Chip Identification Mode (BOS-CI)
CIM	Chip Identification Mode (BOS-CI), same as CI
CPU	Central Processing Unit
CMAC	Cipher-bases Message Authentication Code
CRC	Cyclic Redundancy Check
Crypto2304T	Asymmetric Cryptographic Processor
CRT	Chinese Remainder Theorem
DES	Data Encryption Standard
DPA	Differential Power Analysis
DFA	Differential Failure Analysis
DTRNG	Deterministic Random Number Generator
EC	Elliptic Curve Cryptography
ECC	Error Correction Code and Elliptic Curve Cryptography depending on the context
EDC	Error Detection Code
SOLID FLASH™ NVM	Nonvolatile memory based on flash cell technology
EMA	Electromagnetic analysis
FL	Flash Loader
Flash	SOLID FLASH™ Memory
FW	Firmware
HW	Hardware
HCL	Hash Cryptographic Library
HSL	Hardware Support Library
IC	Integrated Circuit
ICO	Internal Clock Oscillator
ID	Identification
IMM	Interface Management Module
ITP	Interrupt and Peripheral Event Channel Controller
I/O	Input/Output
IRAM	Internal Random Access Memory
ITSEC	Information Technology Security Evaluation Criteria
M	Mechanism

List of Abbreviations

MED	Memory Encryption and Decryption
MMU	Memory Management Unit
NVM	Non Volatile Memory
O	Object
OS	Operating system
PEC	Peripheral Event Channel
PFD	Post Failure Detection Unit
PICC	Proximity Integrated Circuit Card (can be of any form factor)
PCD	Proximity Coupling Device
PRNG	Pseudo Random Number Generator
PROM	Programmable Read Only Memory
PTRNG	Physical Random Number Generator
RAM	Random Access Memory
RFI	Radio Frequency Interface
RMS	Resource Management System
RNG	Random Number Generator
ROM	Read Only Memory
RSA	Rives-Shamir-Adleman Algorithm
SCP	Symmetric Cryptographic Processor
SF	Security Feature
SFR	Special Function Register, as well as Security Functional Requirement. The specific meaning is given in the context
SHA	Secure Hash Algorithm
SPA	Simple power analysis
SW	Software
SO	Security objective
T	Threat
TM	Test Mode (STS)
TOE	Target of Evaluation
TRNG	True Random Number Generator
TSC	TOE Security Functions Control
TSF	TOE Security Functionality
UART	Universal Asynchronous Receiver/Transmitter
UM	User Mode (STS)
UMSLC	User mode Security Life Control
WDT	Watch Dog Timer
XRAM	eXtended Random Access Memory
TDES	Triple DES Encryption Standard also known as 3DES

12 Glossary

Application Program/Data	Software which implements the actual TOE functionality provided for the user or the data required for that purpose
Bill-Per-Use	Bill-Per-Use concept allowing the user to configure the chips
Central Processing Unit	Logic circuitry for digital information processing
Chip	Integrated Circuit]
Chip Identification Data	Data stored in the SOLID FLASH™ NVM containing the chip type, lot number (including the production site), die position on wafer and production week and data stored in the ROM containing the STS version number
Chip Identification Mode	Operational status phase of the TOE, in which actions for identifying the individual chip by transmitting the Chip Identification Data take place
Controller	IC with integrated memory, CPU and peripheral devices
Crypto2304T	Cryptographic coprocessor for asymmetric cryptographic operations (RSA, Elliptic Curves)
Cyclic Redundancy Check	Process for calculating checksums for error detection
End User	Person in contact with a TOE who makes use of its operational capability
Firmware	Is software essential to put the chip into operation. The firmware is located in the ROM and parts of it in the SOLID FLASH™ NVM
Flash Loader	Software enabling to download software after delivery
Hardware	Physically present part of a functional system (item)
Integrated Circuit	Component comprising several electronic circuits implemented in a highly miniaturized device using semiconductor technology
Internal Random Access Memory	RAM integrated in the CPU
Mechanism	Logic or algorithm which implements a specific security function in hardware or software
Memory Encryption and Decryption	Method of encoding/decoding data transfer between CPU and memory
Memory	Hardware part containing digital information (binary data)
Microprocessor	CPU with peripherals
Object	Physical or non-physical part of a system which contains information and is acted upon by subjects
Operating System	Software which implements the basic TOE actions necessary to run the user application
Programmable Read Only Memory	Non-volatile memory which can be written once and then only permits read operations
Random Access Memory	Volatile memory which permits write and read operations
Random Number Generator	Hardware part for generating random numbers
Read Only Memory	Non-volatile memory which permits read operations only
Resource Management System	Part of the firmware containing SOLID FLASH™ NVM programming routines, AIS31 test bench etc.
SCP	Is the symmetric cryptographic coprocessor for symmetric cryptographic operations (TDES, AES).
Security Function	Part(s) of the TOE used to implement part(s) of the security objectives

Glossary

Security Target	Description of the intended state for countering threats
Smart Card	Is a plastic card in credit card format with built-in chip. Other form factors are also possible, i.e. if integrated into mobile devices
Software	Information (non-physical part of the system) which is required to implement functionality in conjunction with the hardware (program code)
Subject	Entity, generally in the form of a person, who performs actions
Target of Evaluation	Product or system which is being subjected to an evaluation
Test Mode	Operational status phase of the TOE in which actions to test the TOE hardware take place
Threat	Action or event that might prejudice security
User Mode	Operational status phase of the TOE in which actions intended for the user takes place

13 Revision History

Version	Description of change
2.8	Previous version
3.9	Final version

All referenced product or service names and trademarks are the property of their respective owners.

Edition 2026-09-02

Published by
Infineon Technologies AG
81726 Munich, Germany

© 2026 Infineon Technologies AG.
All Rights Reserved.

Do you have a question about this document?
Email: erratum@infineon.com

Document reference

IMPORTANT NOTICE

The information contained in this Security Target is given as a hint for the implementation of the product only and shall in no event be regarded as a description or warranty of a certain functionality, condition or quality of the product. Before implementation of the product, the recipient of this application note must verify any function and other technical information given herein in the real application. Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of non-infringement of intellectual property rights of any third party) with respect to any and all information given in this Security Target.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office (www.infineon.com).

WARNINGS

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.