<<RCL on RS46X Version 01 Security Target Public Version>>

PROJECT TITLE	RCL on RS46X
PRODUCT	RCL on RS46X
PRODUCT VARIANT	RCL
DOCUMENT VERSION	1.4
AUTHOR	Tyrone Stodart
DATE	19th April 2011

ABSTRACT

The Security Target of the RCL on RS46X aims to provide potential users of the product with

- A definition of the main properties of the RCL running on the RS46X IC that are evaluated and certified as a composite evaluation
- Confidence in the RCL properties that can be used along with the IC to build an integrated TOE (i.e. IC + RCL + operating system + other application software)

TABLE OF CONTENTS

List of Figures

List of Tables

ABBREVIATIONS

Term	Meaning
3DES	Triple-DES. A block cipher formed from the Data Encryption Standard (DES) cipher by using it three times. May use 2 or 3 keys
AES	Advanced Encryption Standard
CBC	Cipher Block Chaining. A mode of DES & AES encryption.
CC	Common Criteria (ISO 15408)
COT	Chip-on-Tape - an IC packaged in a form suitable for embedding into a plastic card to form a security IC.
CPU	Central Processing Unit
CRAM	RAM for use by MMC coprocessor
CRC	Cyclic Redundancy Check
CTR	Counter Mode. A mode of AES encryption.
DES	Data Encryption Standard
DESX	A simple extension of DES
DFA	Differential Fault Analysis
EAL	Evaluation Assurance Level
ECB	Electronic Codebook. A mod of DES and AES encryption. In this document the term single-block is used.
EEPROM	Electronically Erasable Programmable Read-Only Memory
HW RNG	Hardware random number generator (physical random number generator), also considered a ‘true’ RNG (TRNG).
IC	Integrated Circuit
ISC	Integrated Security Concept
IT	Information Technology
MCU	Micro Computer Unit
MMC	Modular Multiplication Coprocessor
OFB	Output Feedback operation mode. A mode of DES Encryption
PKI	Public Key Infrastructure
PP	Protection Profile
PRNG	Pseudo Random Number Generator
RAM	Random Access Memory
RL	Random Logic (Glue Logic)
RCL	Renesas Cryptographic Library
RNG	Random Number Generator
ROM	Read-Only Memory
RSA	Rivest, Shamir, Adelmann – a public key encryption algorithm, named after its inventors.
SFP	Security Function Policy
SFR	Security Functional Requirement
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Functionalities
WAP	Wireless Application Protocol

GLOSSARY

Term	Meaning
Embedded Software	Software held in the chip, having been developed by users of the TOE. Such software generally includes an operating system and may include all or part of applications. There are two types of Embedded Software: • Hard-coded – held in ROM • Soft-coded – held in EEPROM or RAM. The chip does not depend on such software, and Embedded Software is not part of the TOE. However, hard-coded Embedded Software will be included in the ROM of any issued TOE at manufacturing time. Note that Embedded Software includes all software on a TOE other than the IC Dedicated Software. Embedded Software is also referred to as 'Security IC Software' (especially in [BSI-PP-0035]).
EWE	An interrupt generated by the TOE whenever an attempt is made to write to EEPROM. The interrupt results in a jump to the address held at a fixed location in the memory map allowing embedded software to be executed. The embedded software is not part of the TOE.
IC Dedicated Software	Software developed by Renesas and embedded in the IC. (Adopted from [BSI-PP-0035])
IC Dedicated Test Software	Software developed by Renesas for testing the TOE during manufacture. This software is part of the TOE, but is not available for general use by operating systems, applications or end-users in phase 7 of the lifecycle (see section 1.4.3).
IC Dedicated Support Software	That part of the IC Dedicated Software which provides functions after TOE delivery (Adopted from [BSI-PP-0035])
Manufacturing Identification Data	Some basic data injected into EEPROM, enabling traceability of an IC to the lot and line in which it was manufactured, the Security IC Embedded Software present, and the versions of masks and specifications applicable.
Option List	A form supplied by Renesas and filled in by a TOE customer, specifying various options for the manufacture of TOE ICs for that customer. The aspect of particular interest to this security target is : • Selection of whether pre-personalisation data injection is required The option list also describes the content and structure of the manufacturing identification data that Renesas will inject (see section 1.4.4.2).
Renesas	Refer to Renesas Electronics Corporation (http://www.renesas.com/)
Reset state	A state in which the chip does not execute instructions or engage in input/output. The chip can exit the reset state by receiving an external reset. See also section 7.1.
Smartcard	Composition of the TOE, the Smartcard Embedded Software, User Data and the package (the smartcard carrier).
TOE	Target of Evaluation. TOE consists of the chip (IC) and other materials and data. However, the term is sometimes used to indicate just the chip.
TOE Delivery	The point at which the TOE is delivered, as shown in section 1.4.3. This may be either in the form of wafers (at the end of phase 3) or as packaged modules (at the end of phase 4).
TOE Manufacturer	(As defined in section 8.7 of [BSI-PP-0035]) The IC developer and manufacturer. If the TOE is delivered after phase 4 (i.e. as packaged modules, rather than wafers) then this is also the packager. For the RS46X, the TOE Manufacturer refers to Renesas.
Type-A	ISO14443 Type-A
Type-B	ISO14443 Type-B
Type-C	FeliCa contactless transmission protocol.
TSF Data	Data created by and for the TOE and that might affect the operation of the TOE.
UART	Universal Asynchronous Receiver Transmitter – in accordance with ISO/IEC7816-3.
User Data	All data managed by the Security IC Embedded Software in the application context. User data comprises all data in the final Security IC except for the TSF data.
User Mode	The mode of operation after TOE Delivery. The TOE is set to this mode just before delivery, which renders the IC Dedicated Test Software permanently unavailable.
WDT	Watchdog Timer – a feature of the chip that enables embedded software to be regularly executed during the operation of the IC. This allows checks to be made on the execution environment to help detect potential attacks or insecure conditions.

1 ST INTRODUCTION

The ST aims to provide potential users of the TOE with

- A definition of the main properties of the RCL running on the RS46X IC that are evaluated and certified as a composite evaluation
- Confidence in the RCL properties that can be used along with the IC to build an integrated TOE (i.e. IC + RCL + operating system + other application software).

1.1 ST REFERENCE

Title: RCL on RS46X Version 01 Security Target (Public Version of the document used in the evaluation process)

Version: 1.4

Provided by: Renesas Electronics Europe Ltd.

This Security Target applies to the Renesas RCL v5126 on RS46X Version 01 integrated circuit (as defined in detail in the configuration list for the evaluation). This public version is an abridged form of the evaluated version (rev. 5341) with the approval of the Certification Body.

1.2 TOE REFERENCE

The TOE configuration is summarised in the table below:

Table 1-1: TOE Configuration

Item Type	Item	Version	Form of delivery
Hardware	RS46X integrated circuit	01	Wafer or packaged module (see section 1.4.3)
Software	IC Dedicated Test Software Test ROM software	13347	Included in RS46X test ROM
Software	RNG on-line test software	1.3 (defined by the version of [UGM])	Hardcopy: provided as a part of [UGM]. (This is implemented in the Embedded Software by the user)
Software	DES/AES Library for RS-4*		Electronic data (This is implemented in the Embedded Software by the user)
	RS4_LL.lib	4658	
	RS4_LL.txt	4658	
	RS4_LL.h	4658	
	RCL	5126	
Document	RS46X Hardware Manual [HM]	1.10	Electronic data/Hardcopy
Document	RS-4 User Guidance [UGM]	1.3	Electronic data/Hardcopy
Document	RS46X Option List [OPT]	1.3	Electronic data/Hardcopy
Document	RCL User's Manual [RCLUM]	1.20	Electronic data/Hardcopy

Further description of the TOE is provided in section 1.4.1.

* While the DES/AES Library are listed as part of the HW Platform TOE, they are also provided within the RCL. Within this ST it is assumed that the customer will integrate the DES and AES functions of the RCL as described in the RCL User's Manual, rather than the DES and AES Libraries separately (as described in the RS-4 User Guidance).

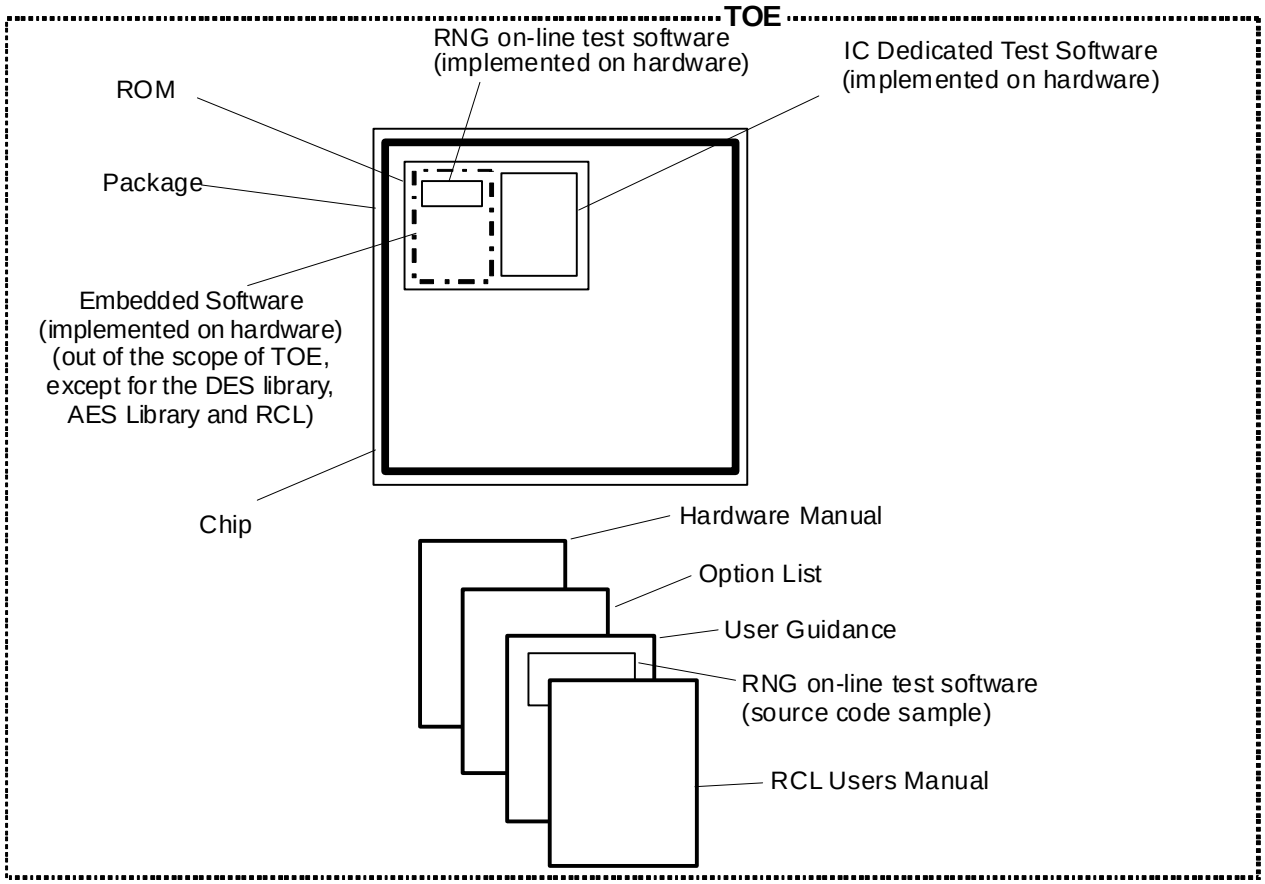

Figure 1-1: Configuration of the TOE

Figure 1-1 shows the configuration of the TOE. The TOE consists of hardware, software, and documents. The software (as part of the TOE) is provided as being implemented on the hardware. Among software, the Embedded Software developed by the TOE user is outside the scope of the TOE. The RNG online test software is provided to the user as a source code sample in [UGM], and it is implemented in the TOE as the embedded software by the Embedded Software Developer. This is the same configuration as described for the RS46X hardware platform (see [HWST]). The RCL is provided to the user as an object code in an electronic form, and it is implemented in the TOE as the embedded software by the user. The RCL consists of three selectable libraries, PKLIB, SKLIB, HLIB and one common code library, CCPLIB. The RCL also provides a function to perform the RNG online tests, as an alternative to the source code sample in [UGM].

1.3 TOE OVERVIEW

This document is the ST for the Renesas RCL on RS46X IC product, intended for use as a Security IC. The TOE is a composite product of the RCL and the RS46X secure IC which complies with the Eurosmart Protection Profile developed by the Secure Semiconductor Vendor Group [BSI-PP-0035]. The RCL is provided as IC Dedicated Support Software, as defined in [BSI-PP-0035]. The RCL provides cryptographic services using the RS46X hardware, with countermeasures against attacks described in this ST. These may be incorporated into the Users security IC embedded software.

Since this is a composite TOE on a certified hardware platform, the RS46X hardware platform and product types are not described in detail. Details can be found in the RS46X Hardware Security Target [HWST] or Public Version of the RS46X Hardware Security Target [HWSTLite].

Where appropriate, the assets, assumptions, threats, objectives and security functional requirements of the hardware platform are included in this ST, but where applicable only to the hardware platform (and not to the composite TOE) are listed separately.

1.4 TOE DESCRIPTION

1.4.1 RCL on RS46X Product Description

The TOE consists of the RCL on the RS46X hardware, along with IC Dedicated Test Software, some embedded software, and reference and guidance documents. IC Dedicated Test Software is used in IC production only, and is not available to users.

The RCL is provided as IC Dedicated Support Software, as defined in [BSI-PP-0035].

As well as the functional interfaces, the IC surface is also considered as a TOE interface for some potential physical attacks, as described in section 3.2 of [BSI-PP-0035].

1.4.1.1 RS46X Hardware

The RS46X is an integrated circuit based around the high-speed RS-4 Series CPU core (derived from Renesas' widely used H8S general purpose core). The MCU comprises the following major blocks in addition to the CPU: ROM, RAM, EEPROM, random number generator (RNG), MMC, DES coprocessor, AES coprocessor, CRC coprocessor, UART, three interval timers, WDT (optional), two I/O lines and contactless interface unit. More details can be found in [HWST]

1.4.1.2 Software

The TOE includes the following software that are part of the RS46X hardware platform (see [HWST]):

IC Dedicated Test Software:

The IC Dedicated Test Software is integrated into the TOE hardware. It is used for mode transition and testing during IC production, and is not available to users.

The RNG On-line Test Software:

The RNG On-line Test Software is provided to perform the on-line test for randomness, as required in [AIS31]. To enable users to deploy the software as necessary, this software is supplied as a listing in [UGM]. Note that the use of this software is part of the intended method of use of the hardware platform under certain conditions, and it is therefore part of the evaluated configuration.

DES Library for RS-4:

In order to comply with the device's security countermeasure requirements as straightforward as possible for the user, a DES Library for RS-4 is provided. The user manual of the DES library for RS-4 has been described within the [UGM].

A**ES Library for RS-4:**

In order to comply with the device's security countermeasure requirements as straightforward as possible for the user, an AES Library for RS-4 is provided. The user's manual of the AES library for RS-4 has been described within the [UGM].

. In addition, the composite TOE comprises the RCL running on the RS46X Hardware:

The RCL (Renesas Cryptographic Library):

The RCL provides both 'secure' and 'fast' version of certain functions;

Secure functions are those for which security claims are made, and that provide secure implementations of functions for the TOE, subject to any limitations described in [RCLUM].

Fast functions are functions optimised for fast operation, and in this ST no security claims are made regarding these functions. However, use of these functions may be appropriate in an application claiming to be secure under certain situations; for example, if the application data within the function is not security related, or if the operating environment at the time can provide security against information leakage or fault injection. An example may be signature generation in a secure personalisation environment protecting against leakage or fault injection attacks.

It is, of course, possible for the composite product developer to make security claims in the composite product ST for an application using these fast functions under appropriate conditions.

The RCL is provided as four different libraries (PKLIB, SKLIB, HLIB, CCPLIB), which provide the following functions:

Public Key Algorithm library (PKLIB):

- Secure implementation of the standard RSA algorithm[1].
- Secure implementation of the RSA CRT algorithm[2].
- Secure implementation of key generation for RSA and RSA CRT
- Secure check sum calculation of standard RSA key set.
- Secure check sum calculation of RSA CRT key set.

[1] The RSA function supports the encryption key lengths from 512-bit to 2048-bit, but only the lengths from 1024-bit to 2048-bit are claimed as sufficiently secure, and at least 1976 bits are required from 2011 for signature applications (see [SA])

[2] The RSA function supports the encryption key lengths from 512-bit to 2048-bit, but only the lengths from 1024-bit to 2048-bit are claimed as sufficiently secure, and at least 1976 bits are required from 2011 for signature applications (see [SA])

- Fast implementation of the standard RSA algorithm
- Fast implementation of key generation for RSA and RSA CRT.

Secret Key Algorithm library (SKLIB):

- Secure implementation of the 3DES algorithm in single block, CBC and OFB modes using the hardware DES coprocessor
- Secure implementation of the AES algorithm in single block, CBC and OFB modes using the hardware AES coprocessor
- Secure implementation of a software random number generator providing high quality pseudo random numbers meeting the so-called K4 class standard (AIS20).
- Fast implementation of the 3DES algorithms in single block, CBC and OFB modes using the hardware DES coprocessor
- Fast implementation of the AES algorithms in single block, CBC and OFB modes using the hardware AES coprocessor
- Fast implementation of a software random number generator providing high quality pseudo random numbers meeting the so-called K4 class standard (AIS20).

HASH library (HLIB):

- Secure implementation of the hash functions SHA-224, SHA-256, SHA-384 and SHA-512.
- Implementation of the hash function SHA-1 is provided for legacy support. It is functionally correct, but no security is claimed.

Common Code Platform library (CCPLIB):

- Secure Modular Inversion. Example of usage: securely calculate the public exponents of a key pair from supplied secret exponents.
- Multiplication of data blocks containing secret information, based on the hardware MMC. This function supports 4 different modes of security.
- Secure implementation of online tests of the underlying platform’s hardware random number generator.
- Secure seed loading and seed updating functions for RCL functions and h/w digital number generator.
- Secure implementation of a random data string generation and secure implementation of the online test of the 16-bit hardware random number generator.
- Scrambled copy / compare / xor: To copy, compare or XOR data blocks containing secret information.

- Scrambled 16 and 32-bit check sum – To compute a 16 or 32-bit check sum over a given data block containing secret information.
- Fast Modular Inversion.
- Fast copy / compare / xor.
- Fast CRC-16 and CRC-32 generation.

Cryptographic functions processing sensitive data inside of a work area in RAM overwrite the respective memory areas after use with variable values, to protect against inherent leakage.

The RCL does not include any key management functionality, other than RSA key generation, since this depends on the application context. The TOE is able to select and combine appropriate co-processor security functions and to provide an interface to perform basic cryptographic functions (as listed above).

All other Security IC Embedded Software (e.g. an operating system) is outside the scope of the TOE. The Security IC Dedicated Support Software is supplied to Renesas by the customer in a secure manner, and is then protected by Renesas' secure production environment.

The relationship between the RCL, the TOE Hardware, and the customer application (out of the scope of the TOE) is illustrated in Figure 1-2:

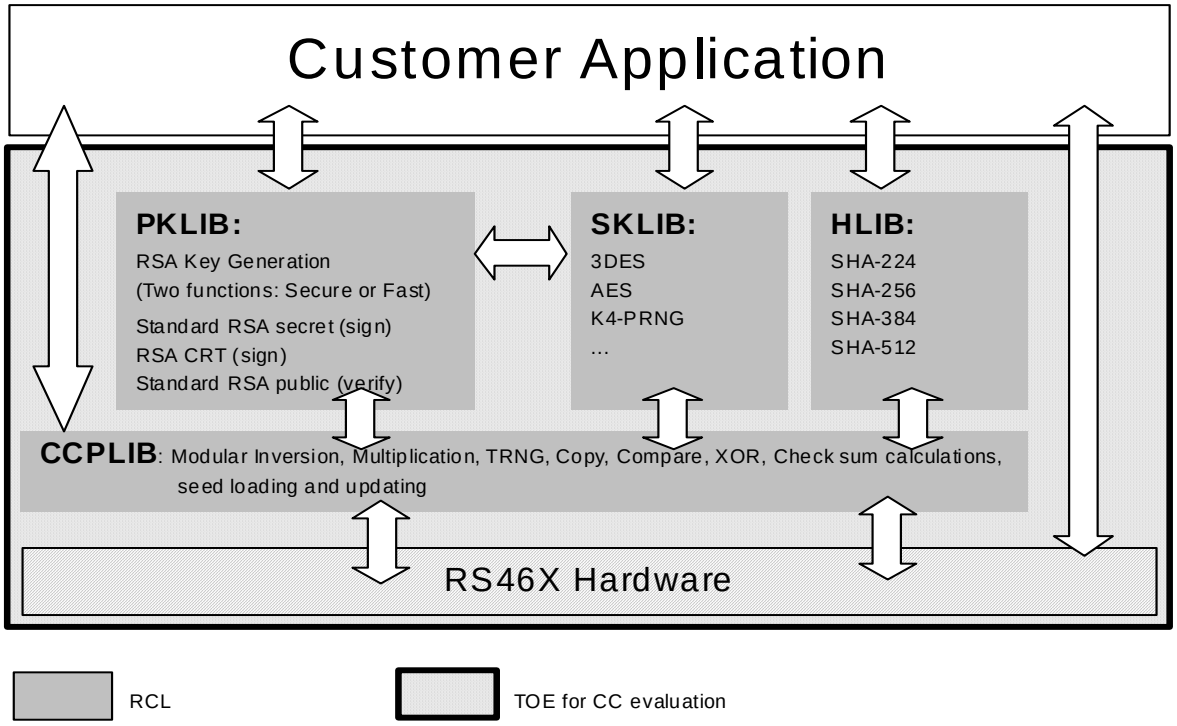

Figure 1-2: RCL (TOE Software), TOE Hardware and Customer Application

1.4.1.3 Documents

The RS46X Hardware Manual [HM] and RCL User's Manual [RCLUM] are supplied as the basic reference for users who are developing Security IC Embedded Software. Guidance for the secure use of the RS-4 Series in applications is given in the User Guidance Manual [UGM], and RCL User's Manual [RCLUM]. Options and contents of the identification data stored in the EEPROM are described in [OPT].

1.4.2 TOE Intended Usage

The TOE is intended for use in a range of high security applications, including high speed secure authentication, data encryption or electronic signature. Examples include: PKI, WAP, m-commerce, digital signature, USIM/UMTS, and banking card.

1.4.3 TOE Lifecycle

The design and manufacturing lifecycle for the RS46X hardware is shown in 1.4.3 of [HWST] (and in section 7.1.1 of [BSI-PP-0035]). The TOE can be delivered either at the end of phase 3, or at the end of phase 4.

The RCL is delivered during Phase 1 as a signed and encrypted PGP file using trusted public keys, either via e-mail or physical media (CD-ROM).

1.4.3.1 Test and User Modes

These are a property of the hardware platform. Details can be found in section 1.4.3.1 of the [HWST].

1.4.3.2 TOE Delivery

The TOE delivery is identical as that for the underlying hardware platform. As noted above, the TOE is delivered at the end of either phase 3 or phase 4, as requested by the customer. In either case, the chip will be delivered in User Mode, and Renesas will apply secure delivery procedures for the transport of the TOE from Renesas premises.

1.4.4 TOE Environments

1.4.4.1 Development Environment

Section [1.4.4.1] of the [HWST] details that the development environment aspects for the underlying hardware platform. Only the additional considerations for the RCL are listed here.

Renesas' development environment for the RCL has implemented security measures specifically to ensure the security of the RCL delivery to customers. The relevant area of the development environment is:

- Design sites

This provides the following main security properties:

- Design sites

  - Confidentiality and integrity of RCL (object code and source code).

Security issues for each of these areas are addressed by processes and procedures put in place by Renesas and within the scope of evaluation. The security measures include IT security to protect information stored on Renesas computer systems, as well as physical security measures for secure storage to ensure that design information and objects are only accessible to authorised staff with a need to know the information. Renesas' integrated security concept (ISC) covers the entire development process, from specification, through design and implementation to manufacturing and shipping. ISC is implemented through the use of standards and procedures that form part of the quality system at the heart of Renesas' business. The rigorous adoption and adherence to procedures, including those relating to security, is an integral part of the quality system at the heart of Renesas' business.

The security of the RCL at design and manufacturing sites is ensured by the same level of security measures as for the hardware design. This ensures that only authorised persons have access to the software and its related information.

1.4.4.2 Injection of Manufacturing Identification and Secret Data

These are a property of the hardware platform. Details can be found in section 1.4.4.2 of the [HWST].

2 CONFORMANCE CLAIMS AND STATEMENT OF COMPATIBILITY

2.1 CC CONFORMANCE CLAIM

This ST is compliant with [CC/1], [CC/2] and [CC/3].

Because the ST conforms to [BSI-PP-0035], it includes extended functionality classes defined in section 5 of [BSI-PP-0035]. The ST is therefore [BSI-PP-0035] conformant, [CC/2] extended and [CC/3] conformant. In addition, this ST includes some additional assumptions, threats, objectives and SFRs defined in [PA]. Therefore, this ST is also [PA] conformant.

The Assurance level is EAL5 augmented (for augmentations see section 6.2).

2.2 PP CLAIMS

2.2.1 PP Reference

This ST conforms to [BSI-PP-0035]

Note that [PA] is used to define additional requirements relating to cryptographic functions. This ST is also [PA] conformant.

2.2.2 PP Tailoring

FCS_RNG.1 [K4-PRNG] is completed with quality metrics – see section. 6.1.1.2.

2.2.3 PP Additions

The inclusions from [BSI-PP-0035] are clearly shown in the relevant section titles. All other threats, assumptions, objectives, extended components, and SFRs, in sections 3.3.2, 3.4.2, 3.4.3, 4.1.2, 4.1.3, 4.2.3, and 6.1.2 are additional to those in the [BSI-PP-0035].

2.3 PACKAGE CLAIM

The assurance level for this Security Target is EAL5 augmented. The augmentations to EAL5 are ALC_DVS.2, and AVA_VAN.5.

2.4 CONFORMANCE RATIONALE

2.4.1 CC Conformance Rationale

This ST implements all of the requirements of [CC/1], [CC/2] and [CC/3] by inclusion (as shown in each of the relevant sections), and hence no further rationale is required.

2.4.2 PP Claim Rationale

This ST, for the TOE type as described in section 1.3 implements all of the requirements, security problem definitions, objectives and security requirements of [BSI-PP-0035] by inclusion (as shown in each of the relevant sections), and hence no further rationale is required.

2.4.3 Package Claims Rationale

This ST implements all of the requirements of EAL5 augmented.

[BSI-PP-0035] requires the assurance level EAL4 augmented. Regarding the Application Note 21 of [BSI-PP-0035] the changes which are needed for EAL5 are described in the different relevant sections of this Security Target.

2.5 STATEMENT OF COMPATIBILITY

There is no conflict between this Security Target and that of the underlying hardware platform:

2.5.1 CC Conformance, Configuration and Lifecycle

The CC Conformance claim and package claim are the same as the underlying hardware platform ST.

The configuration of the TOE is aligned with the configuration of the hardware platform and conforms to the requirements specified by the hardware platform.

The TOE follows the same lifecycle model as the underlying hardware platform.

2.5.2 Assets, Threats and Organisation Security Policies

The assets to be protected by the composite TOE are the same as those for the underlying hardware platform.

The Threats to the composite TOE are that same as those for the hardware platform, except that T.NoSWDetect and T.NoSWResponse are removed since these are both covered entirely by the hardware platform. These threats also align with those of the hardware platform protection profile.

The Organisation Security Policy P.Process-TOE is the same for both the Composite TOE and underlying hardware platform. Specific additional security functionality for the RCL is provided under [PA].

2.5.3 Assumptions, Objectives and Security Functional Requirements.

The assumptions for the composite TOE are that same as those for the hardware platform.

The security objectives of the composite TOE are that same as those for the hardware platform, except that for the additional objective for the RCL functionality covered under [PA] and that O.SWResponse is covered entirely by the hardware platform. These security objectives also align with those of the hardware platform protection profile.

The Security Objectives for the environment of the composite TOE are that same as those for the hardware platform.

The Security Functional Requirements are detailed in section 6.1, and are in three categories

1) Those that are also adopted from the hardware platform for the composite TOE.

   -These are SFRs that are further considered due to the features of the RCL

2) Those that are satisfied entirely by the hardware platform itself

3) Those that are added to consider the Security Functions specifically provided by the RCL.

Since the SFRs are either the same, or an enhancement or addition to those of the underlying hardware platform, then they are compatible with it. The relationships are summarised in the following table:

Table 2-1: Relationship between Platform SFRs and Composite SFRs

Platform SFR Type	Platform SFR	Composite SFR	Relationship
Irrelevant (not being used by the composite ST)	FRU_FLT.2 FPT_FLS.1 FMT_LIM.1 FMT_LIM.2 FAU_SAS.1 FPT_PHP.3	none	Used by the platform but no composite SFRs dependent upon them.
Relevant (being used by the composite ST)	FDP_ITT.1 FPT_ITT.1 FDP_IFC.1	FDP_ITT.1 [RCL] FPT_ITT.1 [RCL] FDP_IFC.1 [RCL]	These are iterated for the composite product but the iterations are not dependent upon the platform SFR's.
	FCS_COP.1 [3DES] FCS_COP.1 [AES] FCS_RNG.1	FCS_COP.1 [3DES] FCS_COP.1 [AES] FCS_RNG.1 [TRNG]	The same SFRs are provided on both the platform and the composite product. In the case of the platform, the software is provided by the DES/AES Wrapper or UGM (for FCS_RNG.1). In the case of the RCL, the software is provided by the RCL, but the same platform hardware is used.
Not applicable	none	FCS_RNG.1 [K4-PRNG] FCS_COP.1[RSA] FCS_COP.1[SHA-224] FCS_COP.1[SHA-256] FCS_COP.1[SHA-384] FCS_COP.1[SHA-512] FCS_CKM.1[RSA]	Composite SFRs not dependent on platform SFRs

3 SECURITY PROBLEM DEFINITION

3.1 DESCRIPTION OF ASSETS

This section defines the assets to be protected by the TOE. Section 3.1 of [BSI-PP-0035] gives the assets relating to the threats, and these are summarised below. The assets are identical to those adopted for the underlying hardware platform RS46X ST [HWST]

The assets to be protected are:

- the User Data
  this includes injection/pre-personalisation data and data generated and managed by the Security IC Embedded Software (subject to adequate protection by the software, see A.Key-Function, A.Plat-Appl and A.Resp-Appl in section 3.4)

- the Security IC Embedded Software stored and in operation, comprising

  - Hard-coded Embedded Software – this is fixed and generally consists of parts or all of the operating system, and parts or all of a number of applications

  - Soft-coded Embedded Software – this may include parts of the operating system or applications.

Both of these types of asset need to have their confidentiality and integrity protected.

A further asset is:

- the security services provided by the TOE for the Security IC Embedded Software

In particular integrity of the Security IC Embedded Software means that the Security IC Embedded Software will be correctly executed, which includes the correct operation of the TOE's functions.

Because random numbers are likely to be used by embedded software for generating cryptographic keys, another asset is:

- the random numbers generated by the TOE[1]

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

- logical design data, physical design data, IC Dedicated Software, and configuration data

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

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

[1] The confidentiality of random numbers is generally protected by embedded software (which is responsible for requesting random numbers). However, it is important that random numbers should not be subject to leakage (cf. T.Leak-Inherent), because of their potential role in cryptographic key generation.

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

3.2 THREATS

3.2.1 Threats Defined in [BSI-PP-0035]

This section adopts the threats to ICs defined in section 3.2 of [BSI-PP-0035]. The threats are identical to those adopted for the underlying hardware platform RS46X ST [HWST]

The TOE has the following high-level security concerns, as in section 3.1 of [BSI-PP-0035]:

SC1 integrity of User Data and of the Security IC Embedded Software (while being executed/processed and while being stored in the TOE's memories).

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

SC3 correct operation of the security services provided by the TOE for the Security IC Embedded Software.

SC4 deficiency of random numbers.

The above high-level security concerns are refined below by defining specific threats. Note that manipulation of the TOE is only a means to threaten User Data or the Security IC Embedded Software and is not a success for the attacker in itself.

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

- Phase 1 and the Phases from TOE Delivery up to the end of Phase 6 are covered by assumptions, and
- The development and production environment starting with Phase 2 up to TOE Delivery are covered by an organisational security policy

3.2.1.1 Standard Threats

See section 3.2 of [BSI-PP-0035], and the example attack scenarios in section 7.3 of [BSI-PP-0035]. For completeness, the threats are summarised below.

T.Leak-Inherent Inherent Information Leakage

An attacker may exploit information which is leaked from the TOE during usage of the Security IC in order to disclose confidential User Data as part of the assets.

No direct contact with the Security IC internals is required here. Leakage may occur through emanations, variations in power consumption, I/O characteristics, clock frequency, or by changes in processing time requirements.

T.Phys-Probing Physical Probing

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

(i) to disclose User Data

(ii) to disclose/reconstruct the Security IC Embedded Software

(iii) to disclose other critical information about the operation of the TOE to enable attacks disclosing or manipulating the User Data or the Security IC Embedded Software.

Physical probing requires direct interaction with the Security IC internals. Techniques commonly employed in IC failure analysis and IC reverse engineering efforts may be used. Before that, hardware security mechanisms and layout characteristics need to be identified. Determination of software design, including treatment of User Data may also be a pre-requisite.

T.Malfunction Malfunction due to Environmental Stress

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

(i) modify security services of the TOE

(ii) modify functions of the Security IC Embedded Software.

(iii) deactivate or affect security mechanisms of the TOE to enable attacks disclosing or manipulating the User Data or the Security IC Embedded Software

This may be achieved by operating the Security IC outside its normal operating conditions.

The modification of security services of the TOE may e.g. affect the quality of random numbers provided by the random number generator up to undetected deactivation when the random number generator does not produce random numbers and the Security IC Embedded Software gets constant values. In another case errors are introduced in executing the Security IC Embedded Software. To exploit this an attacker needs information about the functional operation, e.g. to introduce a temporary failure within a register used by the Security IC Embedded Software with light or a power glitch.

T.Phys-Manipulation Physical Manipulation

An attacker may physically modify the Security IC in order to

(i) modify User Data,

(ii) modify the Security IC Embedded Software

(iii) modify or deactivate security services of the TOE, or

(iv) modify security mechanisms of the TOE to enable attacks disclosing or manipulating the User Data or the Security IC Embedded Software.

The modification may be achieved through techniques commonly employed in IC failure analysis and IC reverse-engineering. The modification may result in the deactivation of a security feature. Determination of software design including treatment of User Data may be a pre-requisite. Changes to circuitry or data can be permanent or temporary.

In contrast to malfunctions (refer to T.Malfunction), the attacker requires to gather significant knowledge about the TOE's internal construction here.

T.Leak-Forced Forced Information Leakage

An attacker may exploit information which is leaked from the TOE during usage of the Security IC in order to disclose confidential User Data as part of the assets even if the information leakage is not inherent but caused by the attacker.

This threat pertains to attacks where methods described in "Malfunction due to Environmental Stress" (refer to T.Malfunction) and/or "Physical Manipulation" (refer to T.Phys-Manipulation) are used to cause leakage from signals which normally do not contain significant information about secrets.

Differential Fault Analysis (DFA) is an example of an attack based on the forced leakage threat.

T.Abuse-Func Abuse of Functionality

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

(i) disclose or manipulate User Data

(ii) manipulate (explore, bypass, deactivate or change) security services of the TOE or of the Security IC Embedded Software or

(iii) manipulate (explore, bypass, deactivate or change) functions of the Security IC Embedded Software or

(iv) enable an attack disclosing or manipulating the User Data or the Security IC Embedded Software.

For the TOE, T.Abuse-Func concerns the threat of unauthorised access to the IC Dedicated Test Software, which is rendered inaccessible by placing the IC into User Mode before TOE Delivery (see section 1.4.3).

3.2.1.2 Threats Related to Security Services

T.RND Deficiency of Random Numbers

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

An attacker may gather information about the random numbers produced by the TOE security service. Because unpredictability is the main property of random numbers this may be a problem in case they are used to generate cryptographic keys. Here the attacker is expected to take advantage of statistical properties of the random numbers generated by the TOE.

Malfunctions or premature ageing are also considered which may assist in getting information about random numbers.

Attacks on random number generation are significant because the random numbers generated may be used as secrets - e.g. to generate cryptographic keys.

Under the threat T.RND, the attacker is assumed to take advantage of statistical properties of the random numbers generated by the TOE without specific knowledge about the RNG itself.

3.3 ORGANISATIONAL SECURITY POLICIES

3.3.1 Policy Requirement from [BSI-PP-0035]

The following policy requirement is taken from section 3.3 of [BSI-PP-0035].

P.Process-TOE Protection 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.

Assets relating specifically to P.Process-TOE are given in section 3.1 of [BSI-PP-0035].

Renesas implement the security measures to satisfy this policy requirement, and these are assessed as part of evaluation and certification against this ST. However, since they are not directly relevant to users of the TOE, the detailed measures and processes that implement the policy are not given here.

3.3.2 Policy Requirement from [PA]

As an additional policy, the TOE provides specific security functionality which can be used by the Security IC Embedded Software for cryptographic algorithm implementation. The policy P.Add-Functions is therefore adopted from [PA]. In the following policy, 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 Composite Product application, against which threats the Security IC Embedded Software will use the specific security functionality.

P.Add-Functions Additional Specific Security Functionality

The TOE shall provide the following specific security functionality to the Security IC Embedded Software:

Public Key Algorithm library (PKLIB):

- Secure implementation of the standard RSA algorithm[1].
- Secure implementation of the standard RSA CRT algorithm[2].
- Secure implementation of key generation for RSA and RSA CRT.
- Secure check sum calculation of standard RSA key set.
- Secure check sum calculation of RSA CRT key set.

Secret Key Algorithm library (SKLIB):

- Secure implementation of the 3DES algorithm in single block, CBC and OFB modes using the hardware DES coprocessor
- Secure implementation of the AES algorithm in single block, CBC and OFB modes using the hardware AES coprocessor

HASH library (HLIB):

- Secure implementation of the hash functions SHA-224, SHA-256, SHA-384 and SHA-512.

Common Code Platform library (CCPLIB):

- Secure Modular Inversion.
- Multiplication of data blocks containing secret information.
- Implementation of online tests of the underlying platform's hardware random number generator.
- Secure seed loading and seed updating functions for RCL functions and h/w digital number generator.
- Secure implementation of a random data string generation and secure implementation of the online test of the 16-bit hardware random number generator.
- Scrambled copy / compare / xor: To copy compare or XOR data blocks containing secret information.

[1] The RSA function supports the encryption key lengths from 512-bit to 2048-bit, but only the lengths from 1024-bit to 2048-bit are claimed as a part sufficiently secure, and at least 1976 bits are required from 2011 for signature applications (see [SA])

[2] The RSA function supports the encryption key lengths from 512-bit to 2048-bit, but only the lengths from 1024-bit to 2048-bit are claimed as a part sufficiently secure, and at least 1976 bits are required from 2011 for signature applications (see [SA])

- Scrambled 16 and 32-bit check sum – To compute a 16 or 32 -bit check sum over a given data block containing secret information.

3.4 ASSUMPTIONS

3.4.1 Assumptions from [BSI-PP-0035]

Appropriate "Protection during Packaging, Finishing and Personalisation (A.Process-Sec-IC)" must be ensured after TOE Delivery up to the end of Phase 6, as well as during the delivery to Phase 7 as specified below. The assumptions are identical to those adopted for the underlying hardware platform RS46X ST [HWST], with TOE guidance documents updated to include the RCL User's Manual.

A.Process-Sec-IC Protection during Packaging, Finishing and Personalisation

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

This means that the Phases after TOE Delivery are assumed to be protected appropriately.

The information and material produced and/or processed by the Security IC Embedded Software Developer in Phase 1 and by the Composite Product Manufacturer can be grouped as follows:

– the Security IC Embedded Software including specifications, implementation and related documentation,

– pre-personalisation and personalisation data including specifications of formats and memory areas, test related data,

– the User Data and related documentation, and

– material for software development support

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

The developer of the Security IC Embedded Software must ensure the appropriate "Usage of Hardware Platform (A.Plat-Appl)" while developing this software in Phase 1 as specified below.

A.Plat-Appl Usage of Hardware Platform

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

(i) TOE guidance documents (refer to the Common Criteria assurance class AGD) such as the hardware data sheet, and the hardware application notes:

i.e. TOE hardware manual [HM], user guidance manual [UGM], RCL User's Manual [RCLUM] and the hardware application notes

(ii) findings of the TOE evaluation reports relevant for the Security IC Embedded Software as documented in the certification report.

The developer of the Security IC Embedded Software must ensure the appropriate "Treatment of User Data (A.Resp-Appl)" while developing this software in Phase 1 as specified below

A.Resp-Appl Treatment of User Data

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

This assumption requires that the Security IC Embedded Software define and positively manage its security relevant User Data, in the manner required by the application context. Without this, the protection provided by the TOE itself may be of no use if the Security IC Embedded Software itself allows data to be compromised.

Examples of embedded software security concerns are given in section 7.2 of [BSI-PP-0035].

3.4.2 Assumptions from [PA]

A.Key-Function Usage of Key-dependent Functions

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

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

Note here that the functions considered in this assumption are part of the Security IC Embedded Software; T.Leak-Inherent and T.Leak-Forced address the cryptographic functions that are part of the hardware.

For example: consider the RSA encryption system that may be implemented in the Security IC Embedded Software. This uses the modular arithmetic functions of the MMC. In this case the leakage characteristic of the implementation will depend partly on the hardware characteristics of the TOE and its MMC, partly on the way in which the embedded software uses the hardware. The properties of the TOE are assessed under this Security Target, but the software implementation is clearly outside the scope of the TOE evaluation.

To assist embedded software developers to implement leak-resistant code, guidance on secure software implementation is given in [UGM].

3.4.3 Other Assumptions

A variety of keys and other security-critical data may be injected for use by Security IC Embedded Software. These may include shared private keys, public/private key pairs, etc. This information could contribute to a cloning attack, or to breaking the security of an instance of the TOE (e.g. by compromising its keys). The integrity of this data is also vital to ensuring the security of the TOE (e.g. preventing unauthorised changes to mask code, IC design or keys). All data supplied for injection/pre-personalisation is assumed to be supported off-card in a secure manner:

A.InjDatSupp Injected Data Support

Data for injection/pre-personalisation will be supplied from the various bodies controlling the operations of the system in which the TOE is functioning, based on [OPT]. It is assumed that the generation, distribution, maintenance, and destruction of this data is adequately secure.

4 SECURITY OBJECTIVES

4.1 SECURITY OBJECTIVES FOR THE TOE

4.1.1 Objectives from [BSI-PP-0035]

The TOE shares the following high-level security goals from section 4.1 of [BSI-PP-0035]:

SG1 maintain the integrity of User Data and of the Security IC Embedded Software (when being executed/processed and when being stored in the TOE's memories).

SG2 maintain the confidentiality of User Data and of the Security IC Embedded Software (when being 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 provide random numbers.

These high-level security goals in the context of the security problem definition build the standing point for the definition of security objectives as required by the Common Criteria. Note that the integrity of the TOE is a means to reach these objectives.

These objectives are identical to those adopted for the underlying hardware platform RS46X ST [HWST], with additional objectives defined under O.Add-Functions.

4.1.1.1 Standard Security Objectives

O.Leak-Inherent Protection against Inherent Information Leakage

The TOE must provide protection against disclosure of confidential data stored and/or processed in the Security IC

- by measurement and analysis of the shape and amplitude of signals (for example on the power, clock, or I/O lines)
- by measurement and analysis of the time between events found by measuring signals (for example on the power, clock, or I/O lines).

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

Note that this objective relates to security services provided by the TOE itself, and Security IC Embedded Software should ensure that the security services are appropriately used in conjunction with any additional leakage countermeasures implemented in software (cf. A.Plat-Appl and A.Resp-Appl in section 3.4.1).

O.Phys-Probing Protection against Physical Probing

The TOE must provide protection against disclosure of User Data, against the disclosure/reconstruction of the Security IC Embedded Software or against the disclosure of other critical information about the operation of the TOE. This includes protection against

- measuring through galvanic contacts, which is direct physical probing on the chip's surface other than on pads being bonded (using standard tools for measuring voltage and current) or
- measuring not using galvanic contacts but other types of physical interaction between charges (using tools used in solid-state physics research and IC failure analysis)

with a prior

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

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

O.Malfunction Protection against Malfunctions

The TOE must ensure its correct operation.

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

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

O.Phys-Manipulation Protection against Physical Manipulation

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

- reverse-engineering (understanding the design and its properties and functions)
- manipulation of the hardware and any data, as well as
- controlled manipulation of memory contents (Application Data).

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

O.Leak-Forced Protection against Forced Information Leakage

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

- by forcing a malfunction (refer to "Protection against Malfunction due to Environmental Stress (O.Malfunction)" and/or
- by a physical manipulation (refer to "Protection against Physical Manipulation (O.Phys-Manipulation)".

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

This objective includes resistance to attacks where T.Phys-Manipulation and T.Leak-Inherent are combined.

O.Abuse-Func Protection against Abuse of Functionality

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

(i) disclose critical User Data

(ii) manipulate critical User Data of the Security IC Embedded Software

(iii) manipulate Soft-coded Security IC Embedded Software

(iv) bypass, deactivate, change or explore security features or security services of the TOE

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

O.Identification TOE Identification

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

The TOE identification data is described in section 1.4.4.2.

4.1.1.2 Security Objectives Related to Specific Functionality (referring to SG4)

O.RND Random Numbers

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

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

O.RND assumes that random number includes both true random number and pseudo random number.

4.1.2 Objectives Based on [PA]

This section includes an objective for the TOE to provide a 3DES and an AES function and objectives for RCL Functionality, which is based on O.Add-Functions in [PA].

O.Add-Functions Additional Specific Security Functionality

The TOE must provide the following specific security functionality to the Security IC Embedded Software:

- Triple Data Encryption Standard (3DES)
- Advanced Encryption Standard (AES)
- Secure implementation of the standard RSA and RSA CRT algorithm (private and public key operation).
- Secure implementation of key generation for standard RSA and RSA CRT.
- Secure check sum calculation of standard RSA and RSA CRT key sets.
- Secure implementation of the 3DES algorithms in single block, CBC and OFB modes using the hardware DES coprocessor.
- Secure implementation of the AES algorithms in single block, CBC and OFB modes using the hardware AES coprocessor.
- Secure implementation of the hash functions SHA-224, SHA-256, SHA-384 and SHA-512.
- Secure Modular Inversion
- Multiplication of data blocks containing secret information.
- Secure seed loading and seed updating functions for RCL functions and h/w digital number generator.

- Secure implementation of a random data string generation and secure implementation of the online test for the 16-bit hardware random number generator.

- Scrambled copy / compare / xor: To copy, compare or XOR data blocks containing secret information.

- Scrambled 16 and 32-bit check sum – To compute a 16 or 32 -bit check sum over a given data block containing secret information.

Note that although single-DES functions are available, only triple-DES functions are stated as objectives. The use of Single-DES may be sufficient for a given application context of the Security IC Embedded Software (e.g. Short-usage sessions keys), which should be evaluated during the full composite product evaluation. The TOE Hardware platform provides both DES and triple-DES functionality – this is discussed in section 15.3.1 of [HM] and in section 6.1.2.1. The triple-DES implementations realised in the RCL use this functionality.

4.2 SECURITY OBJECTIVES FOR THE ENVIRONMENT

4.2.1 Security Objectives for the Security IC Embedded Software development environment from [BSI-PP-0035]

Phase 1

OE.Plat-Appl Usage of Hardware Platform

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

(i) hardware data sheet for the TOE:
i.e. The TOE hardware manual [HM], user guidance manual [UGM], and RCL User's Manual [RCLUM].

(ii) data sheet of the IC Dedicated Software of the TOE,

(iii) TOE application notes

(iv) findings of the TOE evaluation reports relevant for the Security IC Embedded Software as referenced in the certification report.

Because the TOE implements additional specific security functionality (as in O.Add-Functions), OE.Plat-Appl covers the use of these functions by Security IC Embedded Software as follows:

If required, the Security IC Embedded Software shall use these cryptographic services of the TOE and their interface as specified. When key-dependent functions implemented in the Security IC Embedded Software are just being executed, the Security IC Embedded Software must provide protection against disclosure of confidential data (User Data) stored and/or processed in the TOE by using the methods described under

"Inherent Information Leakage (T.Leak-Inherent)" and "Forced Information Leakage (T.Leak-Forced)".

OE.Resp-Appl Treatment of User Data

Security relevant User Data (especially cryptographic keys) are treated by the Security IC Embedded Software as required by the security needs of the specific application context.

Because the TOE implements additional specific security functionality (as in O.Add-Functions), OE.Resp-Appl covers the use of these functions by Security IC Embedded Software as follows:

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

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

4.2.2 Security Objectives for the Operational Environment from [BSI-PP-0035]

TOE Delivery up to the end of Phase 6

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

OE.Process-Sec-IC Protection during composite product manufacturing

Security procedures shall be used after TOE Delivery 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 unauthorised use).

This means that Phases after TOE Delivery up to the end of Phase 6 must be protected appropriately.

Assets relating specifically to the assumption A.Process-Sec-IC (which is the source of this objective) are given in section 3.1 of [BSI-PP-0035].

The precise nature of the protection required will depend on the application context.

4.2.3 Other Environment Security Objectives

For injected/pre-personalisation data, the sources and holders of the data need to support its security requirements.

OE.InjDatSupp Injected Data Support

All data for injections/pre-personalisation shall be generated, distributed, maintained and destroyed in an adequately secure fashion. In general, the data shall be protected for both confidentiality and integrity.

Renesas ensures a secure interface with suppliers of this data by using the injection approach in section 1.4.4.2. Transmission of data to Renesas is secured by a variety of measures dependent on the transmission medium (e.g. ROM data may be sent by encrypted e-mail). The data is securely stored within the Renesas environment according to the medium.

4.3 SECURITY OBJECTIVES RATIONALE

The way in which [BSI-PP-0035] assumptions, organisational security policy and threats are met by objectives is given in section 4.4 of [BSI-PP-0035]. The table below includes the mapping from section 4.4 of [BSI-PP-0035] and adds the rationale for the additional assumptions, policy and threats in this Security Target.

Table 4-1: Coverage of Security Assumptions, Policies and Threats by Objectives

Assumption/Threat/ Organisational Security Policy	Addressed by Objective
A.Plat-Appl	OE.Plat-Appl
A.Resp-Appl	OE.Resp-Appl
P.Process-TOE	O.Identification
A.Process-Sec-IC	OE.Process-Sec-IC
T.Leak-Inherent	O.Leak-Inherent
T.Phys-Probing	O.Phys-Probing
T.Malfunction	O.Malfunction
T.Phys-Manipulation	O.Phys-Manipulation
T.Leak-Forced	O.Leak-Forced
T.Abuse-Func	O.Abuse-Func
T.RND	O.RND
P.Add-Functions	O.Add-Functions
A.Key-Function	OE.Plat-Appl, OE.Resp-Appl
A.InjDatSupp	OE.InjDatSupp

A.Key-Function is enforced by OE.Plat-Appl and OE.Resp-Appl, which directly requires the embedded software to use the features in TOE documentation (including, for example, the countermeasures against side-channel attacks noted in [HM]) to take measures to ensure that keys are not compromised by the way in which the TOE's cryptographic functions are used. Note that this recognises the fact that measures in hardware are only part of the solution for software TOEs, which must also ensure that their algorithms protect keys.

A.Plat-Appl is enforced by a directly corresponding requirement on the environment in OE.Plat-Appl. (Note that as in section 4.4 of [BSI-PP-0035] this applies to Phase 1 of the lifecycle.)

A.Process-Sec-IC is enforced by a directly corresponding requirement on the environment in OE.Process-Sec-IC. (Note that as in section 4.4 of [BSI-PP-0035] this applies to Phases 4-6 of the lifecycle.)

A.Resp-Appl is enforced by a directly corresponding requirement on the environment in OE.Resp-Appl. (Note that as in section 4.4 of [BSI-PP-0035] this applies to Phase 1 of the lifecycle.)

A.InjDatSupp is enforced by a directly corresponding requirement on the environment in OE.InjDatSupp.

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

O.Identification requires that the TOE has to support the possibility of a unique identification. The unique identification can be stored on the TOE. Since the unique identification is generated by the production environment the production environment must support the integrity of the generated unique identification. The technical and organisational security measures that ensure the security of the development environment and production environment are evaluated based on the assurance measures that are part of the evaluation. For a list of material produced and processed by the TOE Manufacturer refer to section 3.1 of [BSI-PP-0035]. All listed items and the associated development and production environments are subject of the evaluation. Therefore, the organisational security policy P.Process-TOE is covered by this objective, as far as organisational measures are concerned.

The basic rationale for T.Leak-Inherent, T.Phys-Probing, T.Phys-Manipulation, T.Malfunction, T.Leak-Forced, and T.Abuse-Func is given in section 4.4 of [BSI-PP-0035]: for all threats the corresponding objectives O.Leak-Inherent, O.Phys-Probing, O.Phys-Manipulation, O.Malfunction, O.Leak-Forced, and O.Abuse-Func are stated in a way, which directly corresponds to the description of the threat. It is clear from the description of each objective, that the corresponding threat is removed if the objective is valid. More specifically, in every case the ability to use the attack method successfully is countered, if the objective holds.

Since O.Add-Functions requires the TOE to implement exactly the same specific security functionality as required by P.Add-Functions, the organisational security policy is covered by the objective. The text below gives further rationale from [PA]

Compared to [BSI-PP-0035] a clarification has been made for the security objective "Usage of Hardware Platform (OE.Plat-Appl)": If required the Security IC Embedded Software shall use these cryptographic services of the TOE and their interface as specified. In addition, the Security IC Embedded Software must implement functions which perform operations on keys (if any) in such a manner that they do not disclose information about confidential data. This addition ensures that the assumption A.Plat-Appl is still covered by the objective OE.Plat-Appl although additional functions are being supported according to O.Add-Functions.

Compared to [BSI-PP-0035] a clarification has been made for the security objective "Treatment of User Data (OE.Resp-Appl)": By definition, encryption data, plain text data, and cryptographic keys are User Data. So, the Security IC Embedded Software will protect such data if required, and the keys and functions are used appropriately in order to ensure the strength of cryptographic operation. Quality and confidentiality must be maintained for keys that are imported and/or derived from other keys. This implies that appropriate key management has to be realised in the environment. These measures make sure that the assumption A.Resp-Appl is still covered by the

security objective OE.Resp-Appl although additional functions are being supported according to P.Add-Functions.

The justification of the additional policy and the additional assumption show that they do not contradict to the rationale already given in [BSI-PP-0035] for the assumptions, policy and threats defined there.

T.RND is addressed by O.RND.

5 EXTENDED COMPONENTS DEFINITION

This ST does not define extended components and only refers to the extended components of [BSI-PP-0035].

5.1 EXTENDED COMPONENTS DEFINITION FROM [BSI-PP-0035]

This ST does not define extended components and only refers to selected extended components of [BSI-PP-0035], and are identical to those adopted for the underlying hardware platform RS46X ST [HWST].

5.1.1 Definition of the Family FCS_RNG

The additional family (FCS_RNG) of the Class FCS (cryptographic support) is defined in [BSI-PP-0035]. This family describes the functional requirements for random number generation used for cryptographic purposes.

This family defines quality requirements for the generation of random numbers which are intended to be used for cryptographic purposes.

5.1.2 Definition of the Family FMT_LIM

The additional family (FMT_LIM) of the Class FMT (Security Management) is defined in [BSI-PP-0035]. This family describes the functional requirements for the Test Features of the underlying hardware platform. For more information see [HWST]

5.1.3 Definition of the Family FAU_SAS

The additional family (FAU_SAS) of the Class FAU (Security Audit) is defined in [BSI-PP-0035]. This family describes the functional requirements for the storage of audit data by the underlying hardware platform. For more information see [HWST]

6 SECURITY REQUIREMENTS

These extended components are identical to those adopted for the underlying hardware platform ST [HWST], with additional Security Functional Requirements under [PA].

6.1 SECURITY FUNCTIONAL REQUIREMENTS

The functional requirements for the TOE come from three sources:

- [BSI-PP-0035] The following SFRs are from the hardware platform Security IC requirements, and also adopted for the composite TOE:

  – FDP_ITT.1
  – FPT_ITT.1
  – FDP_IFC.1
  – FCS_RNG.1

  The following SFRs are also from the hardware platform Security IC requirements, but are entirely satisfied by security functionalities provided by the hardware platform. They are provided here for reference.

  – FRU_FLT.2
  – FPT_FLS.1
  – FMT_LIM.1
  – FMT_LIM.2
  – FAU_SAS.1
  – FPT_PHP.3

- [PA]

  - 3DES – FCS_COP.1. This SFR covers the 3DES cryptographic requirements.
  - AES – FCS_COP.1. This SFR covers the AES cryptographic requirements.
  - RSA – FCS_COP.1 and FCS_CKM.1 are iterated for RSA and RSA CRT cryptographic operation, and key generation.
  - SHA-224 – FCS_COP.1
  - SHA-256 - FCS_COP.1

- SHA-384 - FCS_COP.1
- SHA-512 - FCS_COP.1

- TOE features — Some SFRs introduced from [BSI-PP-0035] are given a wider scope to reflect additional security features of the TOE and functions which support secure features in embedded software:
  - Additional failure detection – the scope of FPT_FLS.1 includes the ability of embedded software to cause a hardware reset (this is covered under FPT_FLS.1 in section 6.1.1.1).

Note that all SFRs are drawn from [CC/2] except for FCS_RNG.1, FMT_LIM.1 & 2, and FAU_SAS.1, which are all defined in section 5 of [BSI-PP-0035].

6.1.1 Security Functional Requirements from [BSI-PP-0035]

In the specifications of SFRs listed below, 'Refinement' sections are taken from [BSI-PP-0035]; 'Application Notes' add information specific to the TOE.

6.1.1.1 Prevention of Malfunction

The underlying hardware platform implements a pair of security functional requirements, FRU_FLT.2 and FPT_FLS.1, that ensure it operates within conditions under which it can maintain a secure state. There is no additional iteration for the composite TOE. For more information, refer to [HWST] and [BSI-PP-0035].

6.1.1.2 Protection against Abuse of Functionality

The underlying hardware platform controls access to test mode. Following [BSI-PP-0035], this is specified using the extended functional family FMT_LIM, defined in section 5.2 of [BSI-PP-0035]. There is no additional iteration for the composite TOE. For more information, refer to [HWST] and [BSI-PP-0035]

The TOE stores identification/pre-personalisation data as described in section 1.4.4.2 of the [HWST]. This is included in [BSI-PP-0035] as the CC Part 2 extended functional component FAU_SAS.1, replacing FAU_GEN.1, and defined in section 6.1 of [BSI-PP-0035]. There is no additional iteration for the composite TOE. For more information, refer to [HWST] and [BSI-PP-0035]

6.1.1.3 Protection against Physical Manipulation and Probing

The underlying hardware platform implements the security functional requirement FRU_PHP.3 to provide resistance to physical attack. There is no additional iteration for the composite TOE. For more information, refer to [HWST] and [BSI-PP-0035].

6.1.1.4 Protection against Leakage

The following SFRs are required as part of protection against inherent leakage. FDP_ITT.1, FPT_ITT.1, and FDP_IFC.1 are iterated here to address protection and control implemented by the RCL of the composite TOE.

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

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

FPT_ITT.1	**Basic internal TSF data transfer protection**
Hierarchical to:	No other components
FPT_ITT.1.1 [RCL]	The TSF shall protect TSF data from *disclosure* when it is transmitted between separate parts of the TOE.
Dependencies:	No dependencies.
Refinement:	The different memories, the CPU and other functional units of the TOE (e.g. a cryptographic co-processor) are seen as separated parts of the TOE.

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

FDP_IFC.1	**Subset information flow control**
Hierarchical to:	No other components
FDP_IFC.1.1 [RCL]	The TSF shall enforce the *Data Processing Policy* on *all confidential data when they are processed or transferred by the TOE or by the Security IC Embedded Software*.
Dependencies:	FDP_IFF.1 Simple security attributes
Data Processing Policy	User Data and TSF data shall not be accessible from the TOE except when the Security IC Embedded Software decides to communicate the User Data via such an external interface. The protection shall be applied to confidential data only, but without the distinction of attributes controlled by the Security IC Embedded Software.

Application note:

1. The above three TSFs are iterated for the Composite TOE RCL (TOE Software), which is required to provide specific measures to protect information using: scrambled data operations to copy / compare / XOR and checksum data; and operand and exponent blinding for functions implemented on the MMC, including secure multiply and secure modular inversion. They are iterated as: FDP_ITT.1 [RCL], FPT_ITT.1 [RCL] and FDP_IFC.1 [RCL].

6.1.1.5 Generation of Random Numbers

The TOE generates random numbers that can be used for cryptographic key generation. To capture the functional requirement, the family FCS_RNG, defined in section 6.1 of [BSI-PP-0035] is used.

Note that the hardware platform provides true random number generation. The Composite TOE RCL provides interface functions to access and support this hardware, along with a software implementation of a pseudo random number generator. Therefore this family is iterated twice, for Hardware Random Number Generator provided by the hardware platform (TRNG - same as in [HWST]) and Pseudo Random Number Generator (K4-PRNG) respectively.

FCS_RNG.1 [TRNG]	**True Random number generation**
Hierarchical to:	No other components
FCS_RNG.1.1	The TSF shall provide a *physical* random number generator that implements *total failure test of the random source and On-line test [assignment: list of additional security capabilities]*
Here, assignment :	*On-line test*
FCS_RNG.1.2	The TSF shall provide random numbers that meet *[selection: independent bits with Shannon entropy of 7.976 bit per octet, Min-entropy of 7.95 bit per octet, [assignment: other comparable quality metric]*.
Here, selection :	*independent bits with Shannon entropy of 7.976 bit per octet*
Dependencies:	No dependencies.
FCS_RNG.1 [K4-PRNG]	**Pseudo Random number generation**
Hierarchical to:	No other components
FCS_RNG.1.1	The TSF shall provide a *deterministic* [selection: *physical, non-physical true, deterministic, hybrid*] random number generator that implements the *functionality class K4 as defined in [AIS20]*
Here, selection :	*deterministic*
Here, assignment :	*functionality class K4 as defined in [AIS20]*

FCS_RNG.1.2	The TSF shall provide random numbers that meet *[*assignment: *the requirements of the formal and statistical tests to meet functionality class K4 as defined in [AIS20]]*. *[*assignment: *a defined quality metric]*.
Here, assignment :	*the requirements of the formal and statistical tests to meet functionality class K4 as designed in [AIS20]*
Dependencies:	No dependencies.

6.1.2 Security Functional Requirements Based on [PA]

6.1.2.1 Cryptographic Support

The TOE provides a DES coprocessor functions, using the hardware platform DES coprocessor. The TOE provides both single and triple DES coprocessor can be used to implement single or triple DES functions.

FCS_COP.1 Cryptographic operation

FCS_COP.1 is iterated here to address the 3DES encryption and decryption (FCS_COP.1 [3DES]) and AES encryption and decryption (FCS_COP.1[AES]).

3DES Encryption and Decryption

FCS_COP.1 [3DES]

Hierarchical to:	No other components
FCS_COP.1.1 [3DES]	The TSF shall perform *encryption and decryption* in accordance with a specified cryptographic algorithm *Triple Data Encryption Standard (3DES) in one of the following modes of operation: ECB, CBC, OFB,* and cryptographic key sizes *of 112 or 168 bits* that meet the following: *U.S. Department of Commerce / National Bureau of Standards* *Data Encryption Standard (DES), FIPS PUB 46-3, 1999 October 25, keying option 1 and 2*
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction

Application notes:

1. Although the TOE provides a single-DES function, DES is only ever used as an encryption function in the context of particular Security IC Embedded Software, and for this reason it is noted that application software may need to use triple DES to achieve a suitable strength. In this case, Security IC Embedded Software shall use the TOE to implement triple DES functions, as described in the guidance for software developers in section 15.3.1 of [HM].

2. Although Smartcard Embedded software may use the DES coprocessor directly it is recommended to use it through the RCL functions.

3. The RCL TOE provides CBC and OFB mode in additional to single block (ECB) mode.

AES Encryption and Decryption

FCS_COP.1 [AES]

Hierarchical to:	No other components
FCS_COP.1.1 [AES]	The TSF shall perform *encryption and decryption* in accordance with a specified cryptographic algorithm *Advanced Encryption Standard (AES) in one of the following modes of operation: ECB, CBC, OFB,* and cryptographic key sizes of *128, 192, and 256 bits* that meet the following: *U.S. Department of Commerce / National Bureau of Standards Advanced Encryption Standard, FIPS PUB 197, 2001 November 26.* *National Institute of Standards and Technology Special Publication 800-38A 2001 Edition, Recommendation for Block Cipher Modes of Operation Methods and Techniques.*
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data without security attributes, or FCS_CKM.1 Cryptographic key generation], FCS_CKM.4 Cryptographic key destruction.

Application notes:

1. Although Smartcard Embedded software may use the AES coprocessor directly it is recommended to use it through the RCL functions.

2. The RCL TOE provides CBC and OFB mode in additional to single block (ECB) mode.

RSA Encryption and Decryption

FCS_COP.1 [RSA]

Hierarchical to:	no other components
FCS_COP.1.1 [RSA]	The TSF shall perform *encryption and decryption* in accordance with a specified cryptographic algorithm *"Rivest-Shamir-Adleman"* (RSA) *and Rivest-Shamir-Adleman "Chinese Remainder Theorem" (RSA CRT)* and cryptographic key sizes *of 1024-2048 bit* that meet the following: *ISO/IEC 9796-2, Annex A*
Dependencies:	[FDP_ITC.1 Import of user data without security attributes

	or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction FMT_MSA.2 Secure security attributes

RSA Key Generation

FCS_CKM.1 [RSA]

Hierarchical to:	No other components
FCS_CKM.1.1 [RSA]	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *Rivest-Shamir-Adleman (RSA) and Rivest-Shamir-Adleman "Chinese Remainder Theorem" (RSA CRT)* and specified cryptographic key sizes *1024-2048 bit* that meet the following: *ISO/IEC 9796-2, Annex A*
Dependencies:	[FCS_CKM.2 Cryptographic key distribution or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction FMT_MSA.2 Secure security attributes

Secure Hash Functions

FCS_COP.1 [SHA-224]

Hierarchical to:	no other component
FCS_COP.1.1 [SHA-224]	The TSF shall perform *hashing* in accordance with a specified cryptographic algorithm *Secure Hash Algorithm (SHA-224)* and cryptographic key sizes *(Not applicable to hashing)* that meet the following: *Secure Hash Standard, Federal Information Processing Standards Publication 180-3, 2008*
Dependencies:	[FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction FMT_MSA.2 Secure security attributes

Application Notes:

1. Hash operations do not use a cryptographic key; the assignment of keysize is not applicable and thus not fulfilled.

FCS_COP.1 [SHA-256]

Hierarchical to:	no other component
FCS_COP.1.1 [SHA-256]	The TSF shall perform *hashing* in accordance with a specified cryptographic algorithm *Secure Hash Algorithm (SHA-256)* and cryptographic key sizes *(Not applicable to hashing)* that meet the following:

Secure Hash Standard, Federal Information Processing Standards Publication 180-3, 2008

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

Application Notes:

1. Hash operations do not use a cryptographic key; the assignment of keysize is not applicable and thus not fulfilled.

FCS_COP.1 [SHA-384]

Hierarchical to: no other component

FCS_COP.1.1 [SHA-384] The TSF shall perform *hashing* in accordance with a specified cryptographic algorithm *Secure Hash Algorithm (SHA-384)* and cryptographic key sizes *(Not applicable to hashing)* that meet the following:

Secure Hash Standard, Federal Information Processing Standards Publication 180-3, 2008

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

Application Notes:

1. Hash operations do not use a cryptographic key; the assignment of keysize is not applicable and thus not fulfilled.

FCS_COP.1 [SHA-512]

Hierarchical to: no other component

FCS_COP.1.1 [SHA-512] The TSF shall perform *hashing* in accordance with a specified cryptographic algorithm *Secure Hash Algorithm (SHA-512)* and cryptographic key sizes *(Not applicable to hashing)* that meet the following:

Secure Hash Standard, Federal Information Processing Standards Publication 180-3, 2008

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

Application Notes:

1. Hash operations do not use a cryptographic key; the assignment of keysize is not applicable and thus not fulfilled.

6.2 SECURITY ASSURANCE REQUIREMENTS

The evaluation assurance level is EAL 5 augmented. An assurance level of EAL5+ is required for smartcard product TOE since it is intended to defend against highly sophisticated attacks without a protected environment. This evaluation assurance level was selected since it provides even formal evidence on the conducted vulnerability assessment. Table 6-1 describes the security assurance requirements. The increase of the assurance components compared [BSI-PP-0035] is expressed with bold letters. And the increase of the assurance components compared to EAL5 are ALC_DVS.2 and AVA_VAN.5.

Table 6-1: Assurance Components

Assurance Class	Assurance components	Required By
ADV: Development	ADV_ARC.1 Security architecture description	EAL5, PP
	ADV_FSP.5 Complete semi-formal functional specification with additional error information	EAL5
	ADV_IMP.1 Implementation representation of the TSF	EAL5, PP
	ADV_INT.2 Well-structured internals	EAL5
	ADV_TDS.4 Semiformal modular design	EAL5
AGD: Guidance documents	AGD_OPE.1 Operational user guidance	EAL5, PP
	AGD_PRE.1 Preparative procedures	EAL5, PP
ALC: Life-cycle support	ALC_CMC.4 Production support, acceptance procedures and automation	EAL5, PP
	ALC_CMS.5 Development tools CM coverage	EAL5
	ALC_DEL.1 Delivery procedures	EAL5, PP
	ALC_DVS.2 Identification of security measures	EAL5, PP
	ALC_LCD.1 Developer defined life-cycle model	EAL5
	ALC_TAT.2 Compliance with implementation standards	EAL5
ATE: Tests	ATE_COV.2 Analysis of coverage	EAL5, PP
	ATE_DPT.3 Testing: modular design	EAL5
	ATE_FUN.1 Functional testing	EAL5
	ATE_IND.2 Independent testing - sample	EAL5
AVA: Vulnerability assessment	AVA_VAN.5 Methodical vulnerability analysis	PP

Regarding, AVA_VAN.5, the following Mandatory Technical Document is expected to be used for the vulnerability analysis:

Supporting Document, Mandatory Technical Document: Application of Attack Potential to Smartcards, April 2008, Version 2.5, Revision 1, CCDB-2008-04-001

6.2.1 Refinements of the TOE Security Assurance Requirements

This ST claims conformance to the [BSI-PP-0035], and therefore it has to conform to the refinements of the TOE security assurance requirements. Because the refinements in the PP are defined for the security assurance components of EAL4, some refinements have to be applied to assurance components of the higher level EAL5 stated in the Security Target.

Table 6-2 lists the influences of the refinements of the PP on the ST. Most of the refined security assurance components have the same level in both documents (PP and ST). The following two subsections apply the refinements to ACM_CMS.5 and ADV_FSP.5 which are different between the PP and the ST.

Table 6-2: Security Assurance Requirements, overview of differences of refinements

Refined in PP	Influence on ST
ALC_DEL.1	Same as in PP, refinement valid without change
ALC_DVS.2	Same as in PP, refinement valid without change
ALC_CMS.4	ALC_CMS.5, refinements have to be adapted
ALC_CMC.4	Same as in PP, refinement valid without change
ADV_ARC.1	Same as in PP, refinement valid without change
ADV_FSP.4	ADV_FSP.5, refinements have to be adapted
ADV_IMP.1	Same as in PP, refinement valid without change
ATE_COV.2	Same as in PP, refinement valid without change
AGD_OPE.1	Same as in PP, refinement valid without change
AGD_PRE.1	Same as in PP, refinement valid without change
AVA_VAN.5	Same as in PP, refinement valid without change

6.2.2 Refinements regarding CM scope (ALC_CMS)

This Security Target requires a higher evaluation level for the CC family ALC_CMS, namely ALC_CMS.5 instead of ALC_CMS.4. The refinement of the PP regarding ALC_CMS.4 is a clarification of the configuration items. Since in ALC_CMS.5, the content and presentation of evidence element ALC_CMS.5.1C only adds a further configuration item (development tool) to the list of items to be tracked by the CM system, the refinement can be applied without changes. The refinement of the configuration item of ALC_CMS.4 can be found in section 6.2.1.3 of the [BSI-PP-0035] and is not cited here.

6.2.3 Functional specification (ADV_FSP)

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

Since the higher level ADV_FSP.5 requires a Functional Specification. The changes only affect the style of description and “error message” should be added in the functional specification. The refinements can be applied without changes and are valid for ADV_FSP.5. The refinement of the original component ADV_FSP.4 can be found in section 6.2.1.6 of the [BSI-PP-0035] and is not cited here.

6.2.4 Rationale for the Assurance Requirements

ALC_DVS.2 and AVA_VAN.5, which have been augmented in 6.3.3 of [BSI-PP-0035], are additionally claimed in ST to comply with BSI-PP-0035.

Therefore, the information will be referred from [BSI-PP-0035].

ALC_DVS.2 Sufficiency of security measures

Development security is concerned with physical, procedural, personnel and other technical measures that may be used in the development environment to protect the TOE.

In the particular case of a Security IC the TOE is developed and produced within a complex and distributed industrial process which must especially be protected.

Details about the implementation, (e.g. from design, test and development tools as well as Initialisation Data) may make such attacks easier. Therefore, in the case of a Security IC, maintaining the confidentiality of the design is very important.

This assurance component is a higher hierarchical component to EAL4 (which only requires ALC_DVS.1). ALC_DVS.2 has no dependencies.

AVA_VAN.5 Advanced methodical vulnerability analysis

Due to the intended use of the TOE, it must be shown to be highly resistant to penetration attacks. This assurance requirement is achieved by the AVA_VAN.5 component.

Independent vulnerability analysis is based on highly detailed technical information.

The main intent of the evaluator analysis is to determine that the TOE is resistant to penetration attacks performed by an attacker possessing high attack potential.

AVA_VAN.5 has dependencies to ADV_ARC.1 “Security architecture description”, ADV_FSP.2 “Security enforcing functional specification”, ADV_TDS.3 “Basic modular design”, ADV_IMP.1 “Implementation representation of the TSF”, AGD_OPE.1

“Operational user guidance”, and AGD_PRE.1 “Preparative procedures”.

All these dependencies are satisfied by EAL5.

It has to be assumed that attackers with high attack potential try to attack Security ICs like smart cards used for digital signature applications or payment systems.

Therefore, specifically AVA_VAN.5 was chosen in order to assure that even these attackers cannot successfully attack the TOE.

6.3 SECURITY REQUIREMENT RATIONALE

6.3.1 Rational for the Security Functional Requirements

The way in which [BSI-PP-0035] objectives are implemented by SFRs is given in section 6.3 of [BSI-PP-0035]. The table below includes the mapping from section 6.3 of [BSI-PP-0035] and adds the rationale for the additional SFRs in this Security Target. This is a complete list for the composite TOE – since there are interactions between the underlying hardware platform and the RCL of the composite product.

The SFRs FDP_ITT.1, FPT_ITT.1 and FDP_IFC.1 are found both on the Hardware Platform and iterated for this composite product. Where the SFR is marked by [HW], this indicates this is from the hardware platform, and details can be found in the [HWST]. Where the SFR is marked by [RCL], this indicates this is the iteration for the composite product and more details can be found in section 6.1.1.4

Table 6-3: Security Requirements versus Security Objectives for the TOE

Objective	TOE Security Functional and Assurance Requirements
O.Leak-Inherent	– FDP_ITT.1 [HW] “Basic internal transfer protection”, – FDP_ITT.1 [RCL] “Basic internal transfer protection”, – FPT_ITT.1 [HW] “Basic internal TSF data transfer protection”, – FPT_ITT.1 [RCL] “Basic internal TSF data transfer protection”, – FDP_IFC.1 [HW] “Subset information flow control”, – FDP_IFC.1 [RCL] “Subset information flow control”
O.Phys-Probing	– FPT_PHP.3 “Resistance to physical attack”
O.Malfunction	– FRU_FLT.2 “Limited fault tolerance”, – FPT_FLS.1 “Failure with preservation of secure state”
O.Phys-Manipulation	FPT_PHP.3 “Resistance to physical attack”
O.Leak-Forced	All requirements listed for O.Leak-Inherent – FDP_ITT.1 [HW], FDP_ITT.1 [RCL], FPT_ITT.1 [HW], FPT_ITT.1 [RCL], FDP_IFC.1 [HW], FDP_IFC.1 [RCL], plus those listed for O.Malfunction and O.Phys-Manipulation – FRU_FLT.2, FPT_FLS.1, FPT_PHP.3
O.Abuse-Func	FMT_LIM.1 “Limited capabilities”, FMT_LIM.2 “Limited availability”, plus those for O.Leak-Inherent, O.Phys-Probing, O.Malfunction, – FDP_ITT.1 [HW], FDP_ITT.1 [RCL], FPT_ITT.1 [HW], FPT_ITT.1 [RCL], FDP_IFC.1 [HW], FDP_IFC.1 [RCL], FPT_PHP.3, FRU_FLT.2, FPT_FLS.1
O.Identification	FAU_SAS.1 “Audit storage”
O.RND	FCS_RNG.1 [TRNG] “Quality metric for random numbers”, FCS_RNG.1 [K4-PRNG] “Quality metric for random numbers”, plus those for O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation, O.Leak-Forced – FDP_ITT.1 [HW], FDP_ITT.1 [RCL], FPT_ITT.1 [HW], FPT_ITT.1 [RCL], FDP_IFC.1 [HW], FDP_IFC.1 [RCL], FPT_PHP.3, FRU_FLT.2, FPT_FLS.1
O.Add-Functions	– FCS_COP.1 [3DES] – FCS_COP.1 [AES] – FDP_ITT.1 [RCL] – FPT_ITT.1 [RCL] – FDP_IFC.1 [RCL], – FCS_COP.1 [RSA], – FCS_COP.1 [SHA-224] – FCS_COP.1 [SHA-256] – FCS_COP.1 [SHA-384], – FCS_COP.1 [SHA-512], – FCS_CKM.1 [RSA]
OE.Plat-Appl	not applicable
OE.Process-Sec-IC	not applicable
OE.Resp-Appl	not applicable
OE.InjDatSupp	not applicable

Reference is made to section 6.3 of [BSI-PP-0035] for the basic rationale. The remainder of this section deals with the additional parts of the rationale introduced for this Security Target.

O.Phys-Manipulation and O.Malfunction can be further addressed by Security IC Embedded Software by using the TOE’s features that allow embedded software to detect and respond to execution states that could represent attacks (included under FPT_FLS.1). However, this depends on the Security IC Embedded Software and is therefore beyond the scope of the TOE.

The requirements of O.Add-Functions to provide secure seed loading, scrambled copy, compare, XOR and checksums; to provide secure multiply; and to provide secure modular inversion, are covered by SFRs: FDP_ITT.1 [RCL], FPT_ITT.1 [RCL] and FDP_IFC.1 [RCL]; as these operations include transfer between separated parts of the TOE (between memory and operation units).

The cryptographic functions requirement of O.Add-Functions is directly implemented by FCS_COP.1.1.

The security functional requirement "Cryptographic operation (FCS_COP.1)" exactly requires those functions to be implemented which are demanded by O.Add-Functions. Therefore, FCS_COP.1 is suitable to meet the security objective.

Nevertheless, the developer of the Security IC Embedded Software must ensure that the additional functions are used as specified and that the User Data processed by these functions are protected as defined for the application context. And more specifically by the security functional requirements specified below.

- FDP_ITC.1 Import of user data without security attributes or FDP_ITC.2 Import of user data without security attributes, or FCS_CKM.1 Cryptographic key generation.

- FCS_CKM.4 Cryptographic key destruction,

which will be fulfilled in the environment (addressed by the security environment objective OE.Resp-Appl), and the details are not known, with the following exceptions:

- The TOE itself fulfils FCS_CKM.1 [RSA] to meet the dependency of the RSA iteration of FCS_COP.1;

- The hash functions SHA-224, SHA-256, SHA-384 and SHA-512 do not use cryptographic keys so the FCS_CKM.1 and FCS_CKM.4 dependencies for the SHA-224, SHA-256, SHA-384 and SHA-512 iterations of FCS_COP.1 can be considered to be automatically discharged.

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 Security IC Embedded Software.

The justification of the security objective O.Add-Functions and the additional requirements (both for the TOE and its environment) show that they do not contradict to the rationale already given in [BSI-PP-0035] for the assumptions, policy and threats defined there.

The assignment/selection operations performed on the SFRs drawn from [BSI-PP-0035] are shown in [BSI-PP-0035] itself. The additional operations performed in this ST are as follows:

Table 6-4: Completion of SFRs

SFR	Operation required	Operation performed
FAU_SAS.1	**[assignment: *list of subjects*]**	*the test process before TOE Delivery*
	[assignment: *list of audit information]*	*the Initialisation Data and/or Pre-personalisation Data and/or supplements of the Security IC Embedded Software*
	[assignment: *type of persistent memory*]	*EEPROM*
FCS_RNG.1 [TRNG]	**[assignment: *list of additional security capabilities*]**	*On-line test*
	[selection: independent bits with Shannon entropy of 7.976 bit per octet, Min-entropy of 7.95 bit per octet, [assignment: other comparable quality metric]	*independent bits with Shannon entropy of 7.976 bit per octet*
FCS_RNG.1 [K4-PRNG]	[selection: *physical, non-physical true, deterministic, hybrid*] [assignment: *list of security capabilities*]	deterministic functionality class K4 as defined in [AIS20].
	[assignment: *a defined quality metric*]	The requirements of the formal and statistical tests to meet functionality class K4 as defined in [AIS20]
FCS_COP.1 [3DES]	[assignment: list of cryptographic operations]	Encryption and decryption
	[assignment: cryptographic algorithm]	3DES in one of the following modes of operation: ECB, CBC, OFB,
	[assignment: cryptographic key sizes]	2-Key 3DES:112 bit 3-Key 3DES: 168 bit
	[assignment: list of standards]	U.S. Department of Commerce / National Bureau of Standards Data Encryption Standard (DES), FIPS PUB 46-3, 1999 October 25, keying option 1 & 2
FCS_COP.1 [AES]	[assignment: list of cryptographic operations]	Encryption and decryption
	[assignment: cryptographic algorithm]	AES in one of the following modes of operation: ECB, CBC, OFB,
	[assignment: cryptographic key sizes]	128, 192, and 256 bits
	[assignment: list of standards]	U.S. Department of Commerce / National Bureau of Standards Advanced Encryption Standard, FIPS PUB 197, 2001 November 26. National Institute of Standards and V. Rijmen, AES Proposal: Rijndael, AES Algorithm Submission, September 3, 1999Technology Special Publication 800-38A 2001 Edition, Recommendation for Block Cipher Modes of Operation Methods and Techniques.
FCS_COP.1 [RSA]	[assignment: list of cryptographic operations]	Encryption and decryption
	[assignment: cryptographic algorithm]	RSA and RSA CRT
	[assignment: cryptographic key sizes]	1024-2048
	[assignment: list of standards]	ISO/IEC 9796-2, Annex A

SFR	Operation required	Operation performed
FCS_COP.1 [SHA-224]	[assignment: list of cryptographic operations]	hashing
	[assignment: cryptographic algorithm]	SHA-224
	[assignment: cryptographic key sizes]	*Not applicable to hashing*
	[assignment: list of standards]	Secure Hash Standard, Federal Information Processing Standards Publication 180-3, 2008
FCS_COP.1 [SHA-256]	[assignment: list of cryptographic operations]	hashing
	[assignment: cryptographic algorithm]	SHA-256
	[assignment: cryptographic key sizes]	*Not applicable to hashing*
	[assignment: list of standards]	Secure Hash Standard, Federal Information Processing Standards Publication 180-3, 2008
FCS_COP.1 [SHA-384]	[assignment: list of cryptographic operations]	hashing
	[assignment: cryptographic algorithm]	SHA-384
	[assignment: cryptographic key sizes]	*Not applicable to hashing*
	[assignment: list of standards]	Secure Hash Standard, Federal Information Processing Standards Publication 180-3, 2008
FCS_COP.1 [SHA-512]	[assignment: list of cryptographic operations]	hashing
	[assignment: cryptographic algorithm]	SHA-512
	[assignment: cryptographic key sizes]	*Not applicable to hashing*
	[assignment: list of standards]	Secure Hash Standard, Federal Information Processing Standards Publication 180-3, 2008
FCS_CKM.1 [RSA]	[assignment: cryptographic key generation algorithm]	RSA
	[assignment: cryptographic key sizes]	1024-2048
	[assignment: list of standards]	ISO/IEC 9796-2, Annex A

6.3.2 Dependencies of Security Functional Requirements

The basic dependencies are shown in section 6.3.2 of [BSI-PP-0035] and are applicable to this ST – these are summarised in the table below:

Table 6-5: Dependencies of Security Functional Requirements

SFR	Dependencies	Fulfilled by Security Requirements in [BSI-PP-0035]?
FRU_FLT.2	FPT_FLS.1	Yes
FPT_FLS.1	None	No dependency
FMT_LIM.1	FMT_LIM.2	Yes
FMT_LIM.2	FMT_LIM.1	Yes
FAU_SAS.1	None	No dependency
FPT_PHP.3	None	No dependency
FDP_ITT.1 [HW]	FDP_ACC.1 or FDP_IFC.1	Yes
FDP_IFC.1 [HW]	FDP_IFF.1	See discussion below
FPT_ITT.1 [HW]	None	No dependency
FDP_ITT.1 [RCL]	FDP_ACC.1 or FDP_IFC.1	Yes
FDP_IFC.1 [RCL]	FDP_IFF.1	See discussion below
FPT_ITT.1 [RCL]	None	No dependency
FCS_RNG.1 [TRNG]	None	No dependency
FCS_RNG.1 [K4-PRNG]	None	No dependency

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

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

Table 6-6: Additional SFR Dependencies

SFR	Dependencies	Fulfilled by Security Requirements in this ST?
FCS_COP.1 [3DES]	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] FCS_CKM.4	Yes (by the environment)
FCS_COP.1 [AES]	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] FCS_CKM.4	Yes (by the environment)
FCS_COP.1 [RSA]	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] FCS_CKM.4	Yes (by FCS_CKM.1 [RSA], and for others by the environment)
FCS_COP.1 [SHA-224]	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] FCS_CKM.4	Yes (by the environment)
FCS_COP.1 [SHA-256]	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] FCS_CKM.4	Yes (by the environment)
FCS_COP.1 [SHA-384]	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] FCS_CKM.4	Yes (by the environment)
FCS_COP.1 [SHA-512]	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] FCS_CKM.4	Yes (by the environment)

SFR	Dependencies	Fulfilled by Security Requirements in this ST?
FCS_CKM.1 [RSA]	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.4	Yes (by FCS_COP.1 [RSA], and for others by the environment)
FDP_ITC.1	[FDP_ACC.1, or FDP_IFC.1] FMT_MSA.3	No additional requirement – see discussion below
FCS_CKM.1	FCS_CKM.4 FMT_MSA.2	No additional requirement – see discussion below
FCS_CKM.4	[FDP_ITC.1 or FCS_CKM.1] FMT_MSA.2	No additional requirement – see discussion below

The dependencies defined for FCS_COP.1 in [CC/2] are discharged by FCS_CKM.1 [RSA] implemented in the TOE, and the requirements on the environment, as described in [PA].

Hence there is no further functional requirement on the TOE arising from the dependencies of FCS_COP.1.

The dependencies defined for FDP_ITC.1, FDP_ITC.2, FCS_CKM.1 and FCS_CKM.4, are not resolved because they will be fulfilled in the environment, where the appropriate decisions will be made.

The discussion in sections 6.3 and 7.3, and the rationale in section 6.3 of [BSI-PP-0035], show how the security functional requirements support each other in meeting the security objectives of this ST. Together with the discussion of dependencies above this shows that the security functional requirements build a mutually supportive whole.

7 TOE SUMMARY SPECIFICATION

7.1 TOE SECURITY FUNCTIONALITIES

The underlying hardware platform provides TOE Security Functionalities SF.HWProtect, SF.Leakprotect, SF.TestModeControl, SF.Inject and SF-ESFunctions. For details refer to [HWST].

1. SF.RCL-LeakProtect

   In addition to the measures provided by the TOE Hardware, the RCL provides additional measures to protect against leakage of information from the IC. The protection features include:

   - Secure RSA operation and keyset generation – the RCL uses its own secure copy/compare and secure multiply in order to realise the RSA (CRT and standard modular exponentiation, and also keyset generation) functions.
   - The secure DES functions of the DES/AES library uses methods to prevent exhaustive key-search attacks. In addition secure copying routines are applied to move data from/to corresponding DES coprocessor data registers in a secure manner.
   - The secure AES functions of the DES/AES library uses methods to prevent exhaustive key-search attacks. In addition secure copying routines are applied to move data from/to corresponding AES coprocessor data registers in a secure manner.
   - The PRNG uses a one way hash algorithm. In addition the methods have been applied in order to prevent exhaustive key-search attacks.
   - Secure SHA-224, SHA-256, SHA-384 and SHA-512 functions – the RCL provides functions to securely calculate SHA-256, SHA-384 and SHA-512 hashes.
   - Secure Modular Inversion – the RCL provides a function to securely calculate the public exponents from the secret exponents of a key pair.
   - Secure multiply – the RCL provides a function to securely multiply two large numbers using the MMC.
   - Scrambled copy / compare / XOR / checksum – the RCL provides functions for securely copying, comparing, XORing or checksum calculations of strings of bytes.
   - Secure seed loading and seed updating – The RCL provides functions for secure seed loading and seed updating to support the RCL functions and h/w digital number generator.

2. SF.RNG

   The hardware platform of the composite TOE includes a physical random number generator (HW RNG) designed to produce random numbers for the generation of cryptographic keys and for other critical uses. For details, see [HWST]

RNG support function – The RCL includes functions to generate seeds using the HW RNG, and also to run the on-line test of the HW RNG to verify that its operation has not been compromised. This function implements a "Poker Test" compatible with the requirements of the [UGM].

Since generating random numbers using the HW RNG is time consuming, the TOE also includes a software implementation of a pseudo random number generator (K4-PRNG). The K4-PRNG is initialised with a seed from the HW RNG. This K4-PRNG meets the functionality class K4 (as specified in [AIS20]).

3. SF.DES

The RCL provides single block (ECB), CBC and OFB mode DES using the hardware platform DES coprocessor that carries out DES and decryption in single block mode according to the following standard:

a) U.S. Department of Commerce / National Bureau of Standards Data Encryption Standard, FIPS PUB 46-3, 1999 October 25.
(As to 3DES, keying option 1 and 2 of FIPS PUB 46-3 specifically)

Application Notes

1. Although the TOE hardware platform provides a DES coprocessor, DES is only actually used as an encryption function in the context of particular Security IC Embedded Software. Only triple-DES is claimed as security functionality for this product. For some application contexts, single-DES may be sufficient[1] – this is a matter for the Security IC Embedded Software security target.

2. The RCL software functions for DES and triple-DES is implemented using the specific features of the DES coprocessor, as described in section 15.3.1 of [HM].

3. Although the TOE provides functions to access and implement DES and Triple-DES respectively, only Triple-DES has sufficient Cryptographic strength to be claimed as a Security Function and for use in secure part of any Smartcard Embedded Software

4. To provide secure embedded software, the software developer is required to ensure that the DES and (where used in a cryptographic algorithm) modular multiplication coprocessors, and related RCL functions, are used in a way that does not compromise the key or plain text (see A.Plat-Appl, A.Resp-Appl and A.Key-Function). Guidance for the implementation of secure Security IC Embedded Software is given in [UGM] and [RCLUM]

[1] Although strength of cryptographic functions is beyond the scope of a Common Criteria evaluation, triple DES would probably be required to be resistant to attacks performed by an attacker possessing High attack potential. Therefore only triple-DES is claimed as a security function.

4. SF.AES

The RCL provides single block (ECB), CBC and OFB mode AES using the hardware platform AES coprocessor that carries out AES encryption and decryption in single block, CBC, and OFB modes according to the following standard:

a) U.S. Department of Commerce / National Bureau of Standards Advanced Encryption Standard, FIPS PUB 197, 2001 November 26.

b) National Institute of Standards and Technology Special Publication 800-38A 2001 Edition, Recommendation for Block Cipher Modes of Operation Methods and Techniques.

Application Notes

1. The RCL software functions for AES are implemented using the specific features of the AES coprocessor, as described in section 18.3.1 of [HM].

2. To provide secure embedded software, the software developer is required to ensure that the AES and (where used in a cryptographic algorithm) modular multiplication coprocessors, are used in a way that does not compromise the key or plain text (see A.Plat-Appl, A.Resp-Appl and A.Key-Function). Guidance for the implementation of secure Security IC Embedded Software is given in [UGM].

5. SF.MMCopro

The hardware platform of the composite TOE provides a coprocessor that carries out modular multiplication according to the specification as in section 15 of [HM].

This forms the basis for software implementation of algorithms such as RSA.

Secure RSA Operation and Keyset Generation – The RCL provides an implementation of the RSA and RSA CRT algorithm using the hardware MMC. It also provides secure generation of keys for the RSA and RSA CRT algorithm using the hardware RNG and software K4-PRNG, and secure checksum calculations of standard RSA and RSA CRT key sets. These are implemented according to standard ISO/IEC 9796-2, Annex A.

The RSA and RSA CRT encryption, decryption and key generation functions of the RCL can handle key sizes ranging from 512 to 2048 bits in length, however only key sizes above and including 1024 bits in length are considered for the purpose of Common Criteria evaluation to provide sufficient resistance to attack. However, it should be noted that for high security applications such as signature generation, the current recommendation from [SA] is a minimum RSA keylength of 1976 from 2011 until 2016.

6. SF.Hash

Hash functions - The RCL provides implementations of the hash functions SHA-224, SHA-256, SHA-384 and SHA-512 according to the standard *Secure Hash Standard, Federal Information Processing Standards Publication 180-2, 2002*.

7.2 CORRESPONDENCE BETWEEN TOE SECURITY FUNCTIONALITIES AND SFR

Table below shows the correspondence between SFRs and each of the Security Functionalities provided in the section 7.1 above.

Table 7-1: TOE Security Functionalities Mapping to SFRs for the composite TOE

TOE Security Functionalities	SFR
SF.RCL-LeakProtect	FDP_ITT.1 [RCL], FPT_ITT.1 [RCL], FDP_IFC.1 [RCL]
SF.RNG	FCS_RNG.1 [K4-PRNG]
SF.DES	FCS_COP.1 [3DES]
SF.AES	FCS_COP.1 [AES]
SF.MMCopro	FCS_COP.1 [RSA], FCS_CKM.1 [RSA]
SF.Hash	FCS_COP.1 [SHA-224], FCS_COP.1 [SHA-256], FCS_COP.1 [SHA-384], FCS_COP.1 [SHA-512]

The table below (for reference) shows the ways in which the hardware platform SFRs are implemented by the hardware platform TOE security functionalities. For details on these, refer to [HWST]

Table 7-2: TOE Security Functionalities Mapping to SFRs for the underlying hardware platform

TOE Security Functionalities	SFR
SF.HWProtect,	FRU_FLT.2, FPT_FLS.1, FPT_PHP.3
SF.LeakProtect	FDP_ITT.1 [HW], FPT_ITT.1 [HW], FDP_IFC.1 [HW]
SF.RNG	FCS_RNG.1 [TRNG]
SF.DES	FCS_COP.1 [3DES]
SF.ESFunctions,	FRU_FLT.2, FPT_FLS.1, FPT_PHP.3,
SF.TestModeControl	FMT_LIM.1, FMT_LIM.2
SF.Inject	FAU_SAS.1

7.3 TOE SUMMARY SPECIFICATION RATIONALE

The table below shows the ways in which the SFRs are implemented by the composite TOE security functionalities.

Table 7-3: SFR Mapping to TOE Security Functionalities for the composite TOE

SFR	TOE Security Functionalities
FDP_ITT.1 [RCL]	SF.RCL-LeakProtect
FPT_ITT.1 [RCL]	SF.RCL-LeakProtect
FDP_IFC.1 [RCL]	SF.RCL-LeakProtect
FCS_RNG.1 [K4-PRNG]	SF.RNG
FCS_COP.1 [3DES]	SF.DES
FCS_COP.1 [AES]	SF.AES
FCS_COP.1 [RSA]	SF.MMCopro
FCS_COP.1 [SHA-224]	SF.Hash
FCS_COP.1 [SHA-256]	SF.Hash
FCS_COP.1 [SHA-384]	SF.Hash
FCS_COP.1 [SHA-512]	SF.Hash
FCS_CKM.1 [RSA]	SF.MMCopro

The table below (for reference) shows the ways in which the hardware platform SFRs are implemented by the hardware platform TOE security functionalities. For details on these, refer to [HWST]

Table 7-4: SFR Mapping to TOE Security Functionalities for the underlying hardware platform

SFR	TOE Security Functionalities
FRU_FLT.2	SF.HWProtect, SF.ESFunctions
FPT_FLS.1	SF.HWProtect, SF.ESFunctions
FMT_LIM.1	SF.TestModeControl
FMT_LIM.2	SF.TestModeControl
FAU_SAS.1	SF.Inject
FPT_PHP.3	SF.HWProtect, SF.ESFunctions
FDP_ITT.1 [HW]	SF.LeakProtect
FPT_ITT.1 [HW]	SF.LeakProtect
FDP_IFC.1 [HW]	SF.LeakProtect
FCS_RNG.1 [TRNG]	SF.RNG

Details of the TOE summary specification rationale are not given in this version of the Security Target.

8 REFERENCE

8.1 REFERENCE MATERIALS

A reference of the form [*REF*, *n*] refers to section *n* of *REF*.

Literature

[BSI-PP-0035] Security IC Platform Protection Profile, BSI-CC-PP-0035-2007, v1.0, Eurosmart, 15 June 2007

[AIS31] Functionality classes and evaluation methodology for physical random number generators, AIS 31, v3.1 (part of Bundesamt für Sicherheit in der Informationstechnik Application Notes and Interpretation of the Scheme), Certification Body of the BSI

[AIS20] Functionality classes and evaluation methodology for deterministic random number generators, AIS 20, v1 (part of Bundesamt für Sicherheit in der Informationstechnik Application Notes and Interpretation of the Scheme), Certification Body of the BSI

[CC/1] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model, July 2009, version3.1, revision3, CCMB 2009-07-001

[CC/2] Common Criteria for Information Technology Security Evaluation, Part 2: Security functional components, July 2009, version3.1, revision3, CCMB 2009-07-002

[CC/3] Common Criteria for Information Technology Security Evaluation, Part 2: Security assurance components, July 2009, version3.1, revision3, CCMB 2009-07-003

Note: the combination of all 3 parts is also referred to in this document as "Common Criteria" or "CC".

[PA] Smartcard Integrated Circuit Platform Augmentations, v1.0, Atmel, Hitachi Europe, Infineon Technologies & Philips Semiconductors, March 2002

[SA] Bekanntmachung zur elektronischen Signatur nach dem Signaturgesetz und der Signaturverordnung Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahnen, 6th January 2010

Document

[HM] RS46X User's Manual: Hardware Renesas Secure Microcomputer RS-4 Series, Rev. 1.10, Renesas Electronics Corporation, 14 September , 2010

[OPT] Option List for Smart Card Microcomputer (for RS46X), v.1.3, Renesas Electronics Corporation., 26 November 2010.

[SM] H8S/2600 Series, H8S/2000 Series Programming Manual, Revision 4.00, Renesas Technology Corp., 24th February, 2006

[UGM] RS-4 dual-way Series User Guidance Manual, Rev. 1.3, Renesas Electronics Corporation, 22 February, 2011

[RCLUM] RS-4 Series Renesas Cryptographic Library User's Manual, Rev. 1.20, Renesas Electronics Corporation, March 22, 2011

[HWST] Renesas RS46X Hardware Security Target rev. 5038, 10th March 2011

[HWSTLite] Renesas RS46X Hardware Security Target Public Version rev 1.3, 10th March 2011

8.2 OTHERS

None.

9 DOCUMENT CHANGE HISTORY

DATE	SECTIONS	CHANGE
14th January 2011	All	Sanitised version based on rev 5204 of the full ST.
16th March 2011	All	Updated to match ST (v5288)
24th March 2011	Section 1.1, Table 1.1, Section 8	Updated reference to [RCLUM] and approved ST (v5341)
8th April 2011	Section 8	Corrected refrerence for [UGM]
19th April 2011	Section 1.1	Corrected version reference and removed word duplication.

** End of Document **