VEGA2
AT90SC19264RC
Security Target - Lite

Revision 1.0

October, 25th, 2002

Contents

Chapter 1

ST Introduction

1.1 ST identification

1 Title: VEGA2 AT90SC19264RC Security Target (ST-Lite)

2 A glossary of terms used is given in Annex A.

3 This Security Target has been constructed with Common Criteria (CC) Version 2.1.

1.2 ST overview

4 This Security Target is for a microcontroller (MCU) device with security features. The device is a member of a family of single chip MCU devices which are intended for use within Smartcard products. The family codename is AVR ASL4 and the 'parent' device of the family, from which other family members will be derived, is the VEGA2 AT90SC19264RC.

5 The VEGA2 AT90SC19264RC MCU device (AT568D5, rev.F) is being evaluated against the CC Smartcard Integrated Circuit Protection Profile PP/9806 to Evaluation Assurance Level 4 (EAL4) augmented of AVA_VLA.4. The other AVR ASL4 family members will be evaluated in the future under the Common Criteria maintenance scheme. Atmel Smart Card ICs, a division of ATMEL Corporation, is the developer and the sponsor for the AVR ASL4 evaluations.

6 The devices in the AVR ASL4 family are centred around AVR RISC family of single-chip microcontroller devices. The AVR RISC family, with designed-in security features, is based on the industry-standard AVR low-power HCMOS core and gives access to the powerful instruction set of this widely used device. AVR ASL4 devices are equipped with Flash, RAM, ROM and EEPROM, cryptographic coprocessors, and a host of security features to protect device assets, making it suitable for a wide range of smartcard applications.

1.3 CC conformance claim

7 This Security Target is conformant to parts 2 and 3 of the Common Criteria, v2.1, as follows:

> Part 2 conformant: the security functional requirements are based on those identified in part 2 of the Common Criteria.

Part 3 augmented conformant: the security assurance requirements, including those used in the augmentation, are based on those in part 3 of the Common Criteria.

1.4 Document Objective

8 The purpose of this document is to satisfy the CC requirements for a Security Target; in particular, to specify the security requirements and functions, and the assurance requirements and measures, in accordance with Protection Profile PP/ 9806, Smartcard Integrated Circuit v2.0, against which the AVR ASL4 devices will be evaluated.

1.5 Document Structure

9 Chapter 1 introduces the Security Target, and includes sections on terminology and references.

10 Chapter 2 provides a description of the TOE as an aid to the understanding of its security requirements and address the product type, the intended usage and the general features of the TOE.

11 Chapter 3 describes the TOE security environment.

12 Chapter 4 describes the required security objectives

13 Chapter 5 describes the TOE security functional requirements

14 Chapter 6 describes the TOE security assurance requirements

15 Chapter 7 describes the TOE security functions and assurance measures

16 Chapter 8 describes the PP claims

1.6 Scope and Terminology

17 This document is based on the VEGA2 Document [TD], latest issue.

18 The term *Target of Evaluation* (TOE) is standard CC terminology and refers to the product being evaluated, the VEGA2 AT90SC19264RC MCU device in this case. The TOE is subject to hardware evaluation only. Downloaded test software will be used for evaluation purposes but is outside the scope of the TOE. Description of how to use the security features can be found in [TD].

19 Security objectives are defined herein with labels in the form O.xx_xx. These labels are used elsewhere for reference. Similarly, threats, assumptions and organisational security policy are defined with labels of the form T.xx_xx, A.xx_xx, and P.xx_xx respectively.

20 Hexadecimal numbers are prefixed by $, e.g. $FF is decimal 255. Binary numbers are prefixed by %, e.g. %0001 1011 is decimal 27. An integer value may be expressed as a hexadecimal, binary or decimal number, whichever form is the most convenient.

1.7 References

[TD] Technical Data AT90SC19264RC, Latest Issue.

[HSTS] VEGA2 Hardware and Software Test Specifications, Latest Issue.

[SOF] VEGA2 Strength of Security Functions Analysis, Latest Issue.

1.8 Revision History

Rev	Date	Description
1.0	October, 25th, 2002	Initial release, based on AT90SC19264RC VEGA2_ST v1.5

Chapter 2

TOE Description

This part of the ST describes the TOE as an aid to the understanding of its security requirements and address the product type, the intended usage and the general features of the TOE.

2.1 Product type

21 The Target of Evaluation (TOE) is the single chip microcontroller unit to be used in a smartcard product, independent of the physical interface and the way it is packaged. Specifically, the TOE is the VEGA2 AT90SC19264RC device from the AVR ASL4 family of smartcard devices. Generally, a smartcard product may include other optional elements (such as specific hardware components, batteries, capacitors, antennae,...) but these are not in the scope of this Security Target.

22 The devices in the AVR ASL4 family are centered around ATMEL's AVR RISC family of single-chip microcontroller devices. The AVR RISC family, with designed-in security features, is based on the industry-standard AVR RISC low-power HCMOS core and gives access to the powerful instruction set of this widely used device. Different AVR ASL4 family members offer various options. The AVR ASL4 family of devices are designed in accordance with the ISO standard for integrated circuit cards (ISO 7816), where appropriate.

23 Although the TOE evaluation is hardware only, it requires embedded software to test the device and demonstrate certain security characteristics during the development phase. In the end-usage phase there will be no embedded test software in the TOE. Test software will be downloaded into the device EEPROM and be fully erased before devices leave the test environment.

24 The EEPROM contains both Atmel and customer specific data.

25 The TOE includes security logic comprising detectors which monitor voltage, frequency and temperature.

26 The TOE is manufactured in a low voltage (3.0V +/- 0.3V) CMOS process. The device will operate at a supply voltage of 3.0V +/- 10% or 5.0V +/- 10%, with the internal supply regulated to the required operating voltage.

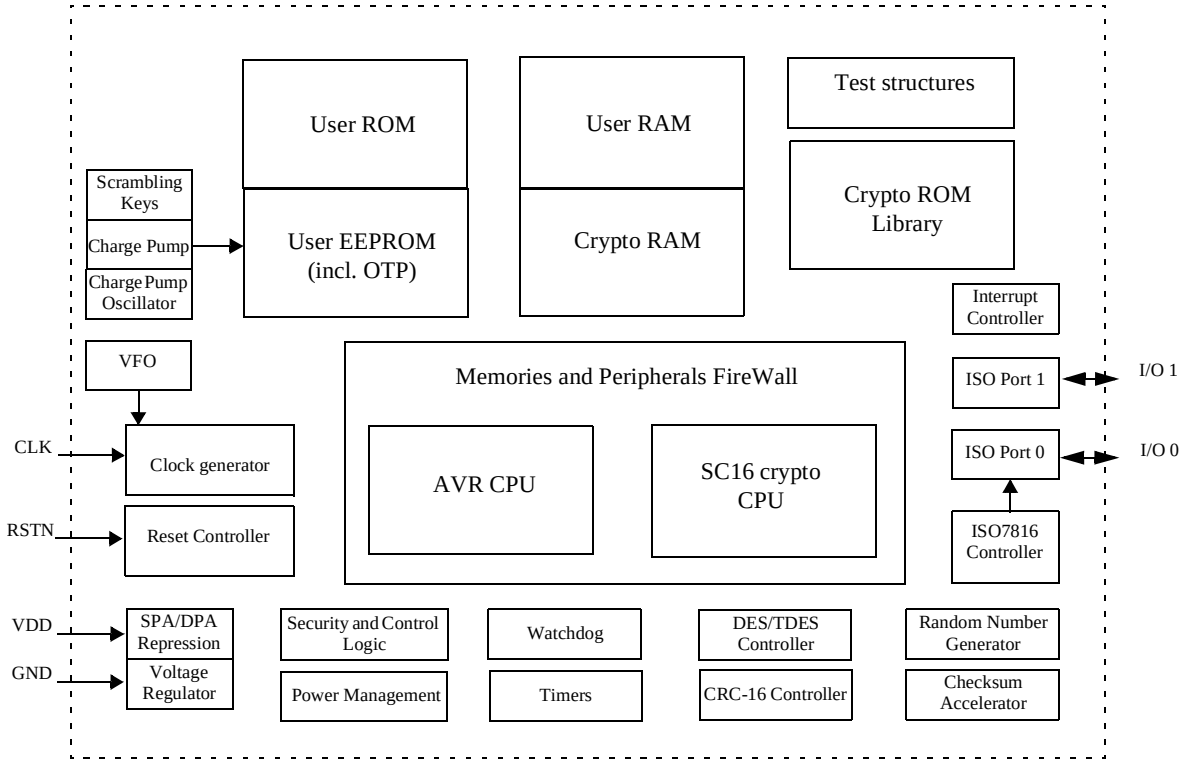

Fig. 2.1 -VEGA2 AT90SC19264RC Block Diagram

27 The embedded software for the VEGA2 device comprises AVR ROM and EEPROM data, and SC16 crypto ROM data.

2.2 Smartcard Product Life-cycle

28 The smartcard product life-cycle comprises 7 phases where the following authorities are involved:

Phase 1	Smartcard software development	**The smartcard software developer** is in charge of the smartcard embedded software development and the specification of IC pre-personalization requirements,
Phase 2	IC Development	**The IC designer** designs the IC, develops IC dedicated software, provides information, software or tools to the smartcard software developer, and receives the software from the developer, through trusted delivery and verification procedures. From the IC design, IC dedicated software and smartcard embedded software, the IC designer constructs the smartcard IC database, necessary for the IC photomask fabrication.
Phase 3	IC manufacturing and testing	**The IC manufacturer** is responsible for producing the IC through three main steps : IC manufacturing, IC testing, and IC pre-personalization.
Phase 4	IC packaging and testing	**The IC packaging manufacturer** is responsible for the IC packaging and testing,
Phase 5	Smartcard product finishing process	**The smartcard product manufacturer** is responsible for the smartcard product finishing process and testing,
Phase 6	Smartcard personalization	**The personalizer** is responsible for the smartcard personalization and final tests. Other application software may be loaded onto the chip at the personalization process.
Phase 7	Smartcard end-usage	**The smartcard issuer** is responsible for the smartcard product delivery to **the smartcard end-user**, and the end of life process.

29 The limits of the evaluation correspond to phases 2 and 3, including the phase 1 delivery and verification procedures and the TOE delivery to the IC packaging manufacturer ; procedures corresponding to phases 4, 5, 6 and 7 are outside the scope of the Security Target.

30 Nevertheless, in certain cases, it would be of great interest to include the phase 4 (IC packaging and testing), within the limits of the TOE. However, for the time being, this option remains outside the scope of this Security Target

31 The figure Fig. 2.2 describes the Smartcard product life-cycle.

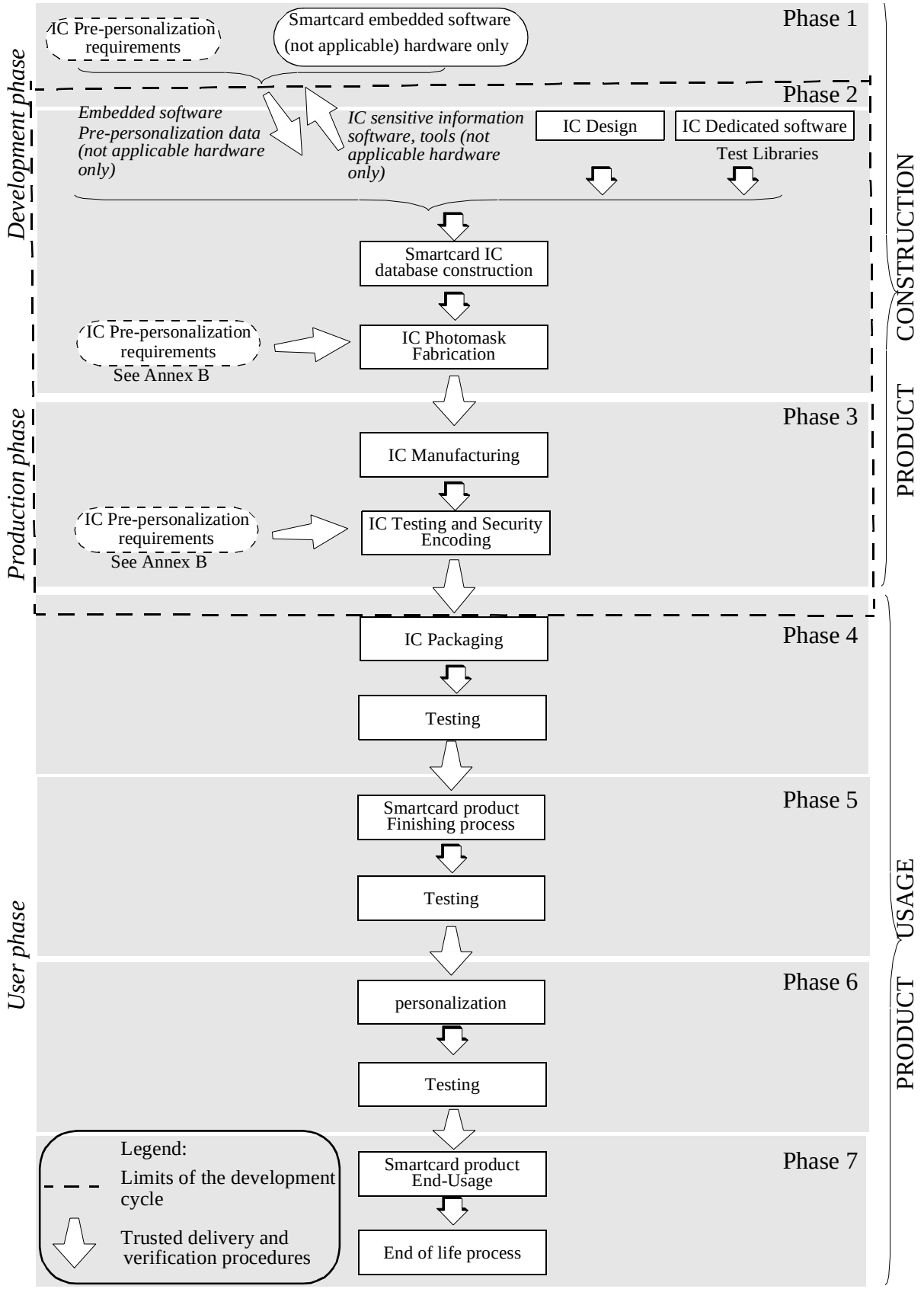

Fig. 2.2 -The Smartcard Product Life Cycle

32 These different phases may be performed at different sites; procedures on the delivery process of the TOE shall exist and be applied for every delivery within a phase or between phases. This includes any kind of delivery performed from phase 1 to phase 7, including:

- intermediate delivery of the TOE or the TOE under construction within a phase,
- delivery of the TOE or the TOE under construction from one phase to the next.

33 These procedures shall be compliant with the assumptions [A_DLV] developed in section 3.2.2.

2.3 TOE environment

34 Considering the TOE, three types of environments are defined :

- Development environment corresponding to phase 2,
- Production environment corresponding to phase 3,
- User environment, from phase 4 to phase 7.

2.3.1 TOE Development Environment

35 The TOE design environment is located in ATMEL RFO (Rousset, France).

36 To assure security, the environment in which the development takes place is made secure with controllable accesses having traceability. Access to the development building is strictly monitored by a Security staff. Visitors must sign a log book and record the time of arrival and time of departure to the building. All visitors are escorted by authorized personnel at all times. All authorized personnel involved fully understand the importance and the rigid implementation of the defined security procedures.

37 The development begins with the TOE's specification. All parties in contact with sensitive information are required to abide by Non-Disclosure Agreements.

38 Design and development of the IC then follows. The design engineer uses appropriate software tools running on a UNIX operating system with the necessary password controls to make his schematic entry/RTL descriptions, design simulations, verifications and generation of the TOE's IC photomask databases. All the components of the TOE were designed in ATMEL RFO (Rousset, France). Sensitive documents, databases on tapes, diskettes, and circuit layout information are stored in appropriate locked cupboards and safes. Disposal of unwanted confidential data is carried out by shredding (paper documents) or complete electronic erasures (electronic documents, databases).

39 Reticles and photomasks are generated from the verified IC database. The reticles and photomasks are then handcarried to the wafer fab processing facilities.

2.3.2 TOE Production Environment

40 Production starts within the ATMEL RFO Wafer Fab; here the silicon wafers undergo diffusion processing in 25-wafer lots. Computer tracking at wafer level throughout the process is achieved by the WORKSTREAM batch tracking system.

41 The WORKSTREAM system is an on-line manufacturing tracking system that monitors the progress of the wafers through the fabrication cycle. After fabrication the wafers are sent to Back End production site. There, they are thinned to a pre-specified thickness. Then, the TOE is tested to assure conformance with the device specification. During the IC testing, security encoding is performed where some of the bytes of the EEPROM are programmed with unique traceability information, and the customer software is loaded in the EEPROM.

42 The wafers are inked to separate the functional from the non-functional ICs. Finally, the wafers are sawn and sent to the customer.

2.3.3 TOE User Environment

43 The TOE user environment is the environment of phases 4 to 7.

44 At phases 4, 5, and 6, the TOE user environment is a controlled environment.

45 Following the sawing step, the wafer are splited into individual dies. The good ICs are assembled into modules in a module assembly plant.

46 Further testing is carried out followed by the shipment of the modules to the smartcard product manufacturer (embedder) by means of a secure carrier.

47 Additional testing occurs followed by smartcard personalization, retesting and then delivery to the smartcard issuer.

End-user environment (phase 7)

48 Smartcards are used in a wide range of applications to assure authorized conditional access. Examples of such are Pay-TV, Banking Cards, Portable communication SIM cards, Health cards, Transportation cards.

49 Therefore, the user environment covers a wide spectrum of very different functions, thus making it difficult to avoid or monitor any abuse of the TOE.

2.4 TOE logical phases

50 During its construction usage, the TOE may be under several life logical phases. These phases are sorted under a logical controlled sequence. The change from one phase to the next shall be under the TOE control.

2.5 TOE Intended usage

51 The TOE can be incorporated in several applications such as:

- banking and finance market for credit/debit cards, electronic purse (stored value cards) and electronic commerce.
- network based transaction processing such as mobile phones (GSM SIM cards), pay-TV (subscriber and pay-per-view cards), communication highways (Internet access and transaction processing).
- transport and ticketing market (access control cards).
- governmental cards (ID-cards, healthcards, driver license etc....).
- multimedia commerce and Intellectual Property Rights protection.

52 During the phases 1, 2, 3, the product is being developed and produced. The **administrators** are the following:

- the IC designer,

  authorized staff who work for the developer, and who design the MCU (such development staff are trusted and privileged user).
- the IC manufacturer,

  authorized staff who work for the developer and who manufacture and test the MCU (such manufacturing staff are trusted and priviliged users).
- the smartcard dedicated software developer.

  authorized staff who work for the developer and who develop the dedicated test software and the crypto libraries (such development staff are trusted and priviliged users).

53 During phases 4 to 7, the users of the product are the following:

Phase 4	- the packaging manufacturer (**administrator**), - the smartcard embedded software developer, - the system integrator such as the terminal software developer.
Phase 5	- the smartcard product manufacturer **(administrator**), - the smartcard embedded software developer, - the system integrator such as the terminal software developer.

Phase 6	- the personalizer (**administrator**), - customers who, before manufacture, determine the MCU's mask options and the initial memory contents (i.e. the application program), and who, after manufacture, incorporate the MCU into devices. Customers are trusted and privileged users. - the smartcard issuer (**administrator**), - the smartcard embedded software developer, - the system integrator such as the terminal software developer.
Phase 7	- the smartcard issuer **(administrator**), - the smartcard end-user, who uses devices incorporating the MCU. End-users are not trusted and may attempt to attack the MCU. - the smartcard software developer, - the system integrator such as the terminal software developer. The IC manufacturer and the smartcard product manufacturer may also receive ICs for analysis, should problems occur during the smartcard usage.

54 The MCU may be used in the following modes

a) Test mode, in which the MCU runs under the control of dedicated test software written to EEPROM via a test interface, and in conjunction with stimulus provided by an external test system. This mode is intended to be used solely by authorized development staff.

b) User mode, in which the MCU runs under control of the smartcard embedded software. It is intended that customers and end-users will always use the MCU in user mode.

55 During the initial part of the manufacturing process, the MCU is set to test mode. Authorized development staff then test the MCU. After testing, test mode is permanently disabled and the MCU is set to user mode.

56 If a faulty MCU is returned from the field then analysis can be done in user mode only because test mode is inhibited prior to devices going to the field.

57 There is no intermediate mode for fault analysis. The only modes of operation are those stated in paragraph 54.

58 Once manufactured, the MCU operates by executing the smartcard embedded software stored in AVR ROM. The contents of the AVR ROM cannot be modified,

whereas the contents of the EEPROM can, in general, be written to or erased, under the control of the smartcard embedded software.

59 The EEPROM includes OTP bytes, which can be used to store security-related information such as cryptographic keys. The OTP bytes cannot be erased in user mode.

60 The FireWall (Memories and Peripherals Protection Unit) allows the smartcard embedded software to prevent read/write/execute access to (parts of) AVR ROM, EEPROM, RAM, Crypto ROM and peripherals from EEPROM.

61 The ISO7816 compliant I/O ports can be used to pass data to or from the MCU. The application program determines how to interpret the data.

2.6 General IT features of the TOE

62 The TOE IT (Information Technology) functionalities consist of data storage and processing such as:

- arithmetical functions (e.g. incrementing counters in electronic purses, calculating currency conversion in electronic purses...);
- data communication;
- cryptographic operations (e.g. data encryption, digital signature verification).

Chapter 3

TOE Security Environment

63 This section describes the security aspects of the environment in which the TOE is intended to be used, and addresses the description of the assumptions, the assets to be protected, the threats, and the organisational security policies.

3.1 Assets

64 Assets are security relevant elements of the TOE that include:

- the application data of the TOE comprising the IC pre-personalization requirements, such as the AVR ROM, EEPROM, the Crypto ROM and OTP contents.
- the smartcard embedded software.
- the IC specification, design, development tools and technology.

65 Therefore, the TOE itself is an asset.

66 Assets must be protected in terms of confidentiality, integrity and availability.

3.2 Assumptions

67 It is assumed that this section concerns the following items:

- due to the definition of the TOE limits, any assumption for the smartcard software development (phase 1 is outside the scope of the TOE),
- any assumption from phases 4 to 7 for the secure usage of the TOE, including the TOE trusted delivery procedures.

68 Security is always dependent on the whole system: the weakest element of the chain determines the total system security. Assumptions described hereafter must be considered for a secure system using smartcard products:

- assumptions on phase 1,
- assumptions on the TOE delivery process (phases 4 to 7),
- assumptions on phases 4-5-6,
- assumptions on phase 7.

3.2.1 Assumptions on phase 1

A.SOFT_ARCHI	The smartcard embedded software shall be designed in a secure manner, i.e. focusing on integrity of program and data.
A.DEV_ORG	Procedures dealing with physical, personnel, organizational, technical measures for the confidentiality and integrity of smartcard embedded software (e.g. source code and any associated documents) and IC designer proprietary information (tools, software, documentation.) shall exist and be applied in software development.

3.2.2 Assumptions on the TOE delivery process (phases 4 to 7)

69 Procedures shall guarantee the control of the TOE delivery and storage process and conformance to its objectives as described in the following assumptions.

A.DLV_PROTECT	Procedures shall ensure protection of TOE material and information under delivery and storage.
A.DLV_AUDIT	Procedures shall ensure that corrective actions are taken in case of improper operation in the delivery process and storage.
A.DLV_RESP	Procedures shall ensure that people dealing with the procedure for delivery have got the required skill.

3.2.3 Assumptions on phases 4 to 6

A.USE_TEST	it is assumed that appropriate functionality testing of the IC is used in phases 4, 5 and 6.
A.USE_PROD	it is assumed that security procedures are used during all manufacturing and test operations through phases 4, 5, 6 to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorized use).

3.2.4 Assumptions on phase 7

A.USE_DIAG	it is assumed that secure communication protocols and procedures are used between smartcard and terminal.
A.USE_SYS	it is assumed that the integrity and confidentiality of sensitive data stored/handled by the system (terminals, communications...) is maintained.

3.3 Threats

70 The TOE as defined in chapter 2 is required to counter the threats described hereafter; a threat agent wishes to abuse the assets either by functional attacks, environmental manipulations, specific hardware manipulations or by any other types of attacks...

71 Threats have to be split in:

- threats against which specific protection within the TOE is required (class I),
- threats against which specific protection within the environment is required (class II).

3.3.1 Unauthorized full or partial cloning of the TOE

T.CLON — Functional cloning of the TOE (full or partial) appears to be relevant to any phases of the TOE life-cycle, from phase 1 to phase 7.

Generally, this threat is derived from specific threats combining unauthorized disclosure, modification or theft of assets at different phases.

3.3.2 Threats on phase 1 (delivery and verification procedures)

72 During phase 1, three types of threats have to be considered:

a) threats on the smartcard's embedded software and its environment of development, such as:

- unauthorized disclosure, modification or theft of the smartcard embedded software and any additional data at phase 1.

  Considering the limits of the TOE, these previous threats are outside the scope of this security target.

b) threats on the assets transmitted from the IC designer to the smartcard software developer during the smartcard development;

c) threats on the smartcard embedded software and any additional application data transmitted during the delivery process from the smartcard embedded software developer to the IC designer.

73 The previous types b and c threats are described hereafter:

T.DIS_INFO — Unauthorized disclosure of the assets delivered by the IC designer to the smartcard software developer such as sensitive information on IC specification, design and technology, software and tools if applicable;

T.DIS_DEL — Unauthorized disclosure of the smartcard embedded software and any additional application data (such as IC pre-personalization requirements) during the delivery process to the IC designer;

T.MOD_DEL	Unauthorized modification of the smartcard embedded software and any additional application data (such as IC pre-personalization requirements) during the delivery process to the IC designer;
T.T_DEL	Theft of the smartcard embedded software and any additional application data (such as IC pre-personalization requirements) during the delivery process to the IC designer;

3.3.3 Threats on phases 2 to 7

74 During these phases, the assumed threats could be described in three types:

- unauthorized disclosure of assets,
- theft or unauthorized use of assets,
- unauthorized modification of assets.

Unauthorized disclosure of assets

75 This type of threats covers unauthorized disclosure of assets by attackers who may possess a wide range of technical skills, resources and motivation. Such attackers may also have technical awareness of the product.

T.DIS_DESIGN	Unauthorized disclosure of IC design. This threat covers the unauthorized disclosure of proprietary elements such as IC specification, IC design, IC technology detailed information, IC hardware security mechanisms specifications.
T.DIS_SOFT	Unauthorized disclosure of smartcard embedded software and data such as access control, authentication system, data protection system, memory partitioning, cryptographic programs.
T.DIS_DSOFT	Unauthorized disclosure of IC dedicated software. This threat covers the unauthorized disclosure of IC dedicated software including security mechanisms specifications and implementation.
T.DIS_TEST	Unauthorized disclosure of test information such as full results of IC testing including interpretations.

T.DIS_TOOLS	Unauthorized disclosure of development tools. This threat covers potential disclosure of IC development tools and testing tools (analysis tools, microprobing tools).
T.DIS_PHOTOMASK	Unauthorized disclosure of photomask information, used for photoengraving during the silicon fabrication process.

Theft or unauthorized use of assets

76 Potential attackers may gain access to the TOE and perform operations for which they are not authorized. For example, such attackers may personalize the product in an unauthorized manner, or try to gain fraudulous access to the smartcard system.

T.T_SAMPLE	Theft or unauthorized use of TOE silicon samples (e.g. bond out chips...).
T.T_PHOTOMASK	Theft or unauthorized use of TOE photomasks.
T.T_PRODUCT	Theft or unauthorized use of smartcard products.

Unauthorized modification of assets

77 The TOE may be subjected to different types of logical or physical attacks which may compromise security. Due to the intended usage of the TOE (the TOE environment may be hostile), the TOE security parts may be bypassed or compromised reducing the integrity of the TOE security mechanisms and disabling their ability to manage the TOE security. This type of threats includes the implementation of malicious trojan horses.

T.MOD_DESIGN	Unauthorized modification of IC design. This threat covers the unauthorized modification of IC specification, IC design including IC hardware security mechanisms specifications and realization.
T.MOD_PHOTOMASK	Unauthorized modification of TOE photomasks.
T.MOD_DSOFT	Unauthorized modification of IC dedicated software including modification of security mechanisms.
T.MOD_SOFT	Unauthorized modification of smartcard embedded software and data.

78 The table Tab. 3.1 - indicates the relationships between the smartcard phases and the threats.

Threats	Phase 1	Phase 2	Phase 3	Phase 4	Phase 5	Phase 6	Phase 7
Functional cloning							
T.CLON	Class II	Class II	Class I/II	Class I	Class I	Class I	Class I
Unauthorized disclosure of assets							
T.DIS_INFO	Class II						
T.DIS_DEL	Class II						
T.DIS_SOFT		Class II	Class I/II	Class I	Class I	Class I	Class I
T.DIS_DSOFT		Class II	Class I/II	Class I	Class I	Class I	Class I
T.DIS_DESIGN		Class II	Class I/II	Class I	Class I	Class I	Class I
T.DIS_TOOLS		Class II	Class II				
T.DIS_PHOTOMASK		Class II	Class II				
T.DIS_TEST			Class I/II	Class I	Class I	Class I	
Theft or unauthorized use of assets							
T.T_DEL	Class II						
T.T_SAMPLE		Class II	Class I/II	Class I	Class I		
T.T_PHOTOMASK		Class II	Class II				
T.T_PRODUCT			Class I/II	Class I	Class I	Class I	Class I
Unauthorized modification threats							
T.MOD_DEL	Class II						
T.MOD_SOFT		Class II	Class I/II	Class I	Class I	Class I	Class I
T.MOD_DSOFT		Class II	Class I/II	Class I	Class I	Class I	Class I
T.MOD_DESIGN		Class II	Class I/II	Class I	Class I	Class I	Class I
T.MOD_PHOTOMASK		Class II	Class II				

Tab. 3.1 - Threats and Phases

3.4 Organizational Security Policies

79 An organizational security policy is mandatory for the smartcard product usage. The specifications of organizational security policies essentially depend on the applications in which the TOE is incorporated.

80 However, it was found relevant to address the following organizational security policy with the TOE because most of the actual Smart Card secure applications make use of cryptographic standards.

P.CRYPTO **Cryptographic entities, data authentication, and approval functions must be in accordance with ISO, associated industry, or organizational standards or requirements.**

Various cryptographic algorithms and mechanisms, such as triple DES, AES, RSA, MACs, and Digital Signatures, are accepted international standards. These, or others in accordance with industry or organizational standards of similar maturity and definition, should be used for all cryptographic operations in the TOE.
These cryptographic operations are used for instance to support establishment and control of a trusted channel between the TOE and the outside environment.
To support these cryptographic functions, the TOE should supply Random Number Generation (RNG) with sufficient unpredictability and entropy. The TOE shall 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.

Chapter 4

Security Objectives

81 The security objectives of the TOE cover principally the following aspects:

- integrity and confidentiality of assets,
- protection of the TOE and associated documentation during development and production phases.

4.1 Security objectives for the TOE

82 The TOE shall use state of art technology to achieve the following IT security objectives:

O.TAMPER **The TOE must prevent physical tampering with its security critical parts.**

The TOE must provide protection against disclosure of User data, against disclosure/reconstruction of the Smartcard Embedded Software or against disclosure of other critical operational information.
This includes protection against direct micro-probing of signals not connected to bonding pads, but also other contact or contactless probing techniques such as laser probing or electromagnetic sensing. Most of these techniques require a prior reverse engineering of parts of the device to understand its architecture and its security functions.
This also includes protection against inherent information leakage (for example shape of signals, power consumption) on the device external interfaces (for example clock, supply, I/O lines) that could be used to disclose confidential data, as well as forced information leakage caused by induced malfunction or physical manipulation.

O.CLON **The TOE functionality needs to be protected from cloning.**

The TOE must include means to prevent an attacker from reproducing the smartcard functionality. Most of these techniques require a prior reverse engineering of parts of the device to understand its architecture and its security functions.

O.OPERATE	**The TOE must ensure the continued correct operation of its security functions.** The TOE must include protection against the use of stolen silicon samples or products that would ease an attacker gaining fraudulous access to the smartcard system. The TOE must also provide mechanisms to avoid the unauthorized modification of the security functions or software and data, by using the device test commands for instance, or by using uncontrolled/unauthentified software access to memories. The TOE must prevent its operation outside the normal operating conditions where reliability and secure operation has not been proven or tested. This is to prevent errors. The environmental conditions may include voltage, clock frequency, temperature, or external energy fields.
O.FLAW	**The TOE must not contain flaws in design, implementation or operation.** The TOE design must include protection against modification of its security mechanisms (for example detectors or memory protections) that would lead to bypass or reduce their integrity, and therefore open security holes that could be used to access embedded software and data. The TOE design must also provide protection against modification of its embedded software that would lead to bypass or reduce the integrity of some software controlled security mechanisms (for example memory areas definition), and therefore open security holes that could be used to access embedded software and data.
O.DIS_MECHANISM	**The TOE shall ensure that the hardware security mechanisms are protected against unauthorized disclosure.** The TOE must be designed and fabricated so that it requires a high combination of complex equipment, knowledge, skill and time to derive detailed designed information or other information which could be used to compromise security through physical attacks.

O.DIS_MEMORY	**The TOE shall ensure that sensitive information stored in memories is protected against unauthorized access.** The TOE must provide protection against unauthorized access to embedded software and data stored in memories, either using test commands, or by some embedded software (for instance a non-supervisor user application) that would try to dump the memories protected by the Firewall programmation (for instance the supervisor program and/or data), or even by some physical attacks.
O.MOD_MEMORY	**The TOE shall ensure that sensitive information stored in memories is protected against any corruption or unauthorized modification.** The TOE must provide protection against unauthorized access to embedded software and data stored in memories, either using test commands, or by some embedded software (for instance a non-supervisor user application) that would try to modify the memories protected by the Firewall programmation (for instance the supervisor program and/or data), or even by some physical attacks.
O.CRYPTO	**Cryptographic capability shall be available for users to maintain integrity and confidentiality of sensitive data.** The TOE must provide hardware implementation of some cryptographic algorithms that can be used by the embedded software in conjunction with appropriate counter-measure to achieve cryptographic operations (for instance encryption, decryption, integrity checking, signature, key generation, for algorithms such as DES, TDES, RSA, SHA-1, DSA, Elliptic Curves, ...). These cryptographic operations are used for instance to support establishment and control of a trusted channel between the TOE and the outside environment, or protect confidential data stored in the TOE memories. The TOE must also provide random number generation and ensure the cryptographic quality of random number generation. For example, random numbers shall not be predictable and shall have a sufficient entropy. The TOE must 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.

4.2 Security objectives for the environment

4.2.1 Objectives on Phase 1

O.DEV_DIS	The smartcard IC designer must have procedures to control the sales, distribution, storage and usage of the software and hardware development tools and classified documentations, suitable to maintain the integrity and the confidentiality of the assets of the TOE. It must be ensured that tools are only delivered to the parties authorized personnel. It must be ensured that confidential information such as data sheets and general information on defined assets are only delivered to the parties authorized personnel on the need-to-know basis.
O.SOFT_DLV	The smartcard embedded software must be delivered from the smartcard embedded software developer (Phase 1) to the IC designer through a trusted delivery and verification procedure that shall be able to maintain the integrity of the software and its confidentiality, if applicable.
O.SOFT_MECH	To achieve the level of security required by this security target, the smartcard embedded software shall use IC security features and security mechanisms as specified in the smartcard IC documentation (e.g. sensors,...) [TD].
O.DEV_TOOLS	The smartcard embedded software shall be designed in a secure manner, by using exclusively software development tools (compilers, assemblers, linkers, simulators etc.) and software-hardware integration testing tools (emulators) that will grant the integrity of program and data.

4.2.2 Objectives on phase 2 (Development Phase)

O.SOFT_ACS	Embedded software shall be accessible only by authorized personnel within the IC designer on the need to know basis.
O.DESIGN_ACS	IC specifications, detailed design, IC databases, schematics/ layout and any other design information shall be accessible only by authorized personnel within the IC designer on the basis of the need-to-know (physical, personnel, organizational, technical procedures).

O.DSOFT_ACS	Any IC dedicated software specification, detailed design, source code or any further information shall be accessible only by authorized personnel within the IC designer on the basis of need-to-know.
O.MASK_FAB	Physical, personnel, organizational, technical procedures during photomask fabrication (including deliveries between photomasks manufacturer and IC manufacturer) shall ensure the integrity and confidentiality of the TOE.
O.MECH_ACS	Details of hardware security mechanisms shall be accessible only to authorized personnel within the IC designer on the basis of need-to-know.
O.TI_ACS	Security relevant technology information shall be accessible only by authorized personnel within the IC designer on the basis of need-to-know.

4.2.3 Objectives on Phase 3 (Manufacturing Phase)

O.TOE_PRT	The manufacturing process shall ensure that protection of the TOE from any kind of unauthorized use such as tampering or theft. During the IC manufacturing and test operations, security procedures shall ensure the confidentiality and integrity of: - TOE manufacturing data (to prevent any possible copying, modification, retention, theft or unauthorized use) - TOE security relevant test programs, test data, databases and specific analysis methods and tools. These procedures shall define a security system applicable during the manufacturing and test operations to maintain confidentiality and integrity of the TOE by control of: - packaging and storage, - traceability - storage and protection of manufacturing process specific assets (such as manufacturing process documentation, further data, or samples, - access control and audit to tests, analysis tools, laboratories, and databases, - change/modification in the manufacturing equipment, management of rejects.
O.IC_DLV	The delivery procedures from the IC manufacturer shall maintain the integrity and confidentiality of the TOE and its assets.

4.2.4 Objectives on the TOE delivery process (Phases 4 to 7)

O.DLV_PROTECT Procedures shall ensure protection of TOE material and information under delivery, including the following objectives:

- non-disclosure of any security relevant information,
- identification of the elements under delivery,
- meet confidentiality rules (confidentiality level, transmittal form, reception acknowledgement),
- physical protection to prevent external damage,
- secure storage and handling procedures are applicable for all TOEs (including rejected TOEs),
- traceability of TOE during delivery including the following parameters:
  - origin and shipment details,
  - reception, reception acknowledgement
  - location of material and information,

O.DLV_AUDIT Procedures shall ensure that corrective actions are taken in the event of improper operation in the delivery process (including, if applicable, any non-conformance to the confidentiality convention) and highlight all non conformance to this process.

O.DLV_RESP Procedures shall ensure that people (shipping department, carrier, reception department) dealing with the procedure for delivery get the required skill, training and knowledge to meet the procedure requirements, and to act in full accordance with the above expectations.

4.2.5 Objectives on Phase 4 to 6

O.TEST_OPERATE Appropriate functionality testing of the IC shall be used in phases 4 to 6.

During all manufacturing and test operations, security procedures shall be used through phases 4, 5, 6, to maintain confidentiality and integrity of the TOE and of its manufacturing and test data.

4.2.6 Objectives on Phase 7

O.USE_DIAG	Secure communication protocols and procedures shall be used between smartcard and terminal.
O.USE_SYS	The integrity and confidentiality of sensitive data stored and handled by the system (terminals, communications....) shall be maintained.

Chapter 5

TOE security functional requirements

83 The TOE security functional requirements define the functional requirements for the TOE using only functional requirements components drawn from the Common Criteria part 2.

84 The minimum strength of function level for the TOE security requirements is SOF-high.

5.1 Functional requirements applicable to phase 3 only (testing phase)

5.1.1 User authentication before any action (FIA_UAU.2)

85 The TOE security functions shall require each user to be successfully authenticated before allowing any other TOE security functions-mediated actions on behalf of that user.

5.1.2 User Identification before any action (FIA_UID.2)

86 The TOE security functions shall require each user to identify itself before allowing any other TOE security functions mediated actions on behalf of that user.

5.1.3 User Attribute Definition (FIA_ATD.1)

87 The TOE security functions shall maintain the following list of security attributes belonging to individual users:

- Test mode access right,

- Test mode functions access rights,

- Read AVR ROM access right,

- Write AVR ROM access right,

- Execute AVR ROM access right,

- Read EEPROM access right,

- Write EEPROM access right,

- Execute EEPROM access right,

- Read Crypto ROM access right,

- Write Crypto ROM access right,

- Execute Crypto ROM access right,

- Read RAM access right,

- Write RAM access right,

- Execute RAM access right,

- Read access right to peripherals and IO registers,

- Write access right to peripherals and IO registers,

- Execute access right to peripherals and IO registers.

5.1.4 TOE Security Functions Testing (FPT_TST.1)

88 The TOE security functions shall run a suite of self tests **at the request of the authorized user** to demonstrate the correct operation of the TOE security functions.

89 The TOE security functions shall provide authorized users with the capability to verify the integrity of TOE security functions data.

90 The TOE security functions shall provide authorized users with the capability to verify the integrity of stored TOE security functions executable code.

5.1.5 Stored Data Integrity Monitoring (FDP_SDI.1)

91 The TOE security functions shall monitor user data stored within the TOE scope of control for **integrity errors** on all objects, based on the following attributes:

- test signatures from AVR ROM, RAM, Crypto ROM and EEPROM.

5.2 Functional requirements applicable to phases 3 to 7

5.2.1 Management of Security functions behaviour (FMT_MOF.1)

92 The TOE security functions shall restrict the ability to **enable** the functions **available in Test Mode** to **the Test Mode Entry Administrator**.

The TOE security functions shall restrict the ability to **disable** the functions **available in Test Mode** to **the Test Mode Entry Administrator**.

5.2.2 Management of security attributes (FMT_MSA.1)

93 The TOE security functions shall enforce the **ACSF_Policy** (Access Control Security Functions Policy) and **IFCSF_Policy** (Information Flow Control Security Functions Policy) to restrict the ability to **access** the security attributes to **Test Mode Entry Administrator and Firewall supervisor/non-supervisor modes.**

5.2.3 Security roles (FMT_SMR.1)

94 The TOE security functions shall maintain the role **of Test Mode Entry Administrator.**

95 The TOE security functions shall maintain the roles **of the Firewall supervisor/ non-supervisor modes.**

96 The TOE security functions shall be able to associate users with roles.

5.2.4 Static Attribute Initialisation (FMT_MSA.3)

97 The TOE security functions shall enforce the **ACSF_Policy and IFCSF_Policy** to provide **restrictive** default values for security attributes that are used to enforce the security functions policy.

98 The TOE security functions shall allow the **Test Mode Entry Administrator** to specify alternate initial values to override the default values when an object or information is created.

5.2.5 Complete Access Control (FDP_ACC.2)

99 The TOE security functions shall enforce the **ACSF_Policy** (this policy is not disclosed in this ST-lite document) on **:**

- Test Mode Entry Administrator, Firewall supervisor / non-supervisor modes,

and

- AVR ROM, EEPROM, RAM, Crypto ROM, peripheral and IO registers,

and all operations among subjects and objects covered by the security functions policy.

100 The TOE security functions shall ensure that all operations between any subject in the TOE scope of control and any object within the TOE scope of control are covered by an access control security functions policy.

5.2.6 Security Attribute Based Access Control (FDP_ACF.1)

101 The TOE security functions shall enforce the **ACSF_Policy** to objects based on **:**

- Read AVR ROM access right,

- Write AVR ROM access right,

- Execute AVR ROM access right,

- Read EEPROM access right,

- Write EEPROM access right,

- Execute EEPROM access right,

- Read Crypto ROM access right,

- Write Crypto ROM access right,

- Execute Crypto ROM access right,

- Read RAM access right,

- Write RAM access right,

- Execute RAM access right,

- Read peripheral and IO registers access right,

- Write peripheral and IO registers access right,

- Execute peripheral and IO registers access right.

102 The TOE security functions shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed :

- Firewall rules, that are not disclosed in this ST-lite document.

103 The TOE security functions shall explicitly authorize access of subjects to objects based on the following additional rules: **no additional rules**.

5.2.7 Subset Information Flow Control (FDP_IFC.1)

104 The TOE security functions shall enforce the **IFCSF_Policy** on **Test Mode Entry Administrator, test commands and test operations that cause controlled information to flow between the AVR ROM memory and the Test Mode Entry Administrator**.

105 The TOE security functions shall enforce the **IFCSF_Policy** on **Test Mode Entry Administrator, test commands and test operations that cause controlled information to flow between the EEPROM and the Test Mode Entry Administrator**.

106 The TOE security functions shall enforce the **IFCSF_Policy** on **Test Mode Entry Administrator, test commands and test operations that cause controlled information to flow between the Crypto ROM and the Test Mode Entry Administrator**.

107 The TOE security functions shall enforce the **IFCSF_Policy** on **Test Mode Entry Administrator, test commands and test operations that cause controlled information to flow between the RAM and the Test Mode Entry Administrator**.

108 The TOE security functions shall enforce the **IFCSF_Policy** on **Test Mode Entry Administrator, test commands and test operations that cause controlled information to flow between the peripheral and IO registers and the Test Mode Entry Administrator**.

5.2.8 Simple Security Attributes (FDP_IFF.1)

109 The TOE security functions shall enforce the **IFCSF_Policy** based on the following types of subject and information security attributes: **test command syntax.**

110 The TOE security functions shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: **test command syntax rules.**

111 The TOE security functions shall provide **no additional information flow control security functions policy rules**.

112 The TOE security functions shall enforce **no additional security functions policy capabilities**.

113 The TOE security functions shall explicitly authorize an information flow based on the following rules :

- Test command syntax rules, based on test command syntax, that explicitly authorize information flows between Test Mode Entry Administrator and AVR ROM.

- Test command syntax rules, based on test command syntax, that explicitly authorize information flows between Test Mode Entry Administrator and EEPROM.

- Test command syntax rules, based on test command syntax, that explicitly authorize information flows between Test Mode Entry Administrator and Crypto ROM.

- Test command syntax rules, based on test command syntax, that explicitly authorize information flows between Test Mode Entry Administrator and RAM.

- Test command syntax rules, based on test command syntax, that explicitly authorize information flows between Test Mode Entry Administrator and peripheral and IO registers.

114 The TOE security functions shall explicitly deny an information flow based on the following rules :

- Test command syntax rules, based on test command syntax, that explicitly deny information flows between Test Mode Entry Administrator and AVR ROM.

- Test command syntax rules, based on test command syntax, that explicitly deny information flows between Test Mode Entry Administrator and EEPROM.

- Test command syntax rules, based on test command syntax, that explicitly deny information flows between Test Mode Entry Administrator and Crypto ROM.

- Test command syntax rules, based on test command syntax, that explicitly deny information flows between Test Mode Entry Administrator and RAM.

- Test command syntax rules, based on test command syntax, that explicitly deny information flows between Test Mode Entry Administrator and peripheral and IO registers.

115 **IFCSF_Policy**

Tab. 5.1 -Information Flow Control Security Functions Policy

Rules	Attribute	TME Administrator
Test command syntax rules	Test command syntax	Data flow (1)
Notes: (1) : All information about possible data flow and Test command syntax can be found in [HSTS].		

5.2.9 Potential Violation Analysis (FAU_SAA.1)

116 The TOE Security Functions shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the TOE Security Policy.

117 The TOE security functions shall enforce the following rules for monitoring audited events:

a) Accumulation or combination of abnormal environmental conditions (Supply voltage, clock input frequency, temperature, UV light) known to indicate a potential security violation,

b) Accumulation or combination of physical tampering (Micro-probing, critical FIB modification) known to indicate a potential security violation,

c) Accumulation or combination of Firewall violations (user trying to illegally access controlled memories or objects, user trying to execute illegal opcodes) known to indicate a potential security violation,

d) Accumulation of watchdog violations known to indicate a potential security violation,

e) No other rules.

5.2.10 Unobservability (FPR_UNO.1)

118 The TOE security functions shall ensure that **any users** are unable to observe the operation **of TOE internal activity** on **TOE objects** by **authorized users or subjects**.

5.2.11 Notification of Physical Attack (FPT_PHP.2)

119 The TOE security functions shall provide unambiguous detection of physical tampering that might compromize the TOE security functions.

120 The TOE security functions shall provide the capability to determine whether physical tampering with the TOE security functions's devices or TOE security functions's elements has occurred.

121 For **values of voltage, clock input frequency, temperature and UV light which go outside acceptable bounds, for micro-probing and critical FIB modification, for Firewall rules violations (including illegal opcodes), and for watchdog violations**, the TOE security functions shall monitor the devices and elements and notify **the Firewall supervisor** when physical tampering with the TOE security functions's devices or TOE security functions' elements has occurred.

5.2.12 Resistance to Physical Attack (FPT_PHP.3)

122 The TOE security functions shall resist **tampering of voltage, clock input frequency, temperature, UV light, micro-probing, critical FIB modification, Firewall rules violations (including illegal opcodes) and watchdog violations** to the **TOE and its security functions** by responding automatically such that the TOE security policy is not violated.

5.2.13 Cryptographic operation (FCS_COP.1)

123 The TSF shall perform **hardware data encryption and decryption** in accordance with **the DES cryptographic algorithm** using **56-bit cryptographic key sizes** that meets **the Data Encryption Standard (DES), FIPS PUB 46-3, 25th October, 1999.**

124 The TSF shall perform **hardware data encryption and decryption** in accordance with **the Triple Data Encryption Standard (TDES) cryptographic algorithm** using **112-bit cryptographic key sizes** that meets **the E-D-E two-key triple-encryption implementation of the Data Encryption Standard, FIPS PUB 46-3, 25th October, 1999.**

125 The TSF shall perform **hardware data hash and signature** in accordance with **the SHA-1 cryptographic algorithm** using **no cryptographic key** that meets **the Secure Hash Standard, FIPS PUB 180-1, 17th April, 1995.**

126 The TSF shall perform **hardware data encryption and decryption** in accordance with **the RSA without CRT cryptographic algorithm** using **512-bit, 1024-bit, 2048-bit cryptographic key sizes** that meets **no standard.**

127 The TSF shall perform **hardware data encryption and decryption** in accordance with **the RSA with CRT cryptographic algorithm** using **512-bit, 1024-bit, 2048-bit cryptographic key sizes** that meets **no standard.**

5.2.14 Cryptographic key generation (FCS_CKM.1)

128 The TSF shall generate cryptographic keys in accordance with cryptographic key generation algorithm **Miller-Rabin algorithm with confidence criteria (t) between 0 and 255** and specified cryptographic key sizes **512-bit, 1024-bit, 2048-bit (respectively 2 primes of 256 bits, 512 bits and 1024 bits)** that meet the **NIST special publication 800-2, April 1991.**

Chapter 6

TOE security assurance requirements

129 The assurance requirement is EAL 4 augmented of additional assurance components listed in the following sections.

130 Some of these components are hierarchical ones to the components specified in EAL4.

131 All the components are drawn from Common Criteria Part 3, v2.1.

6.1 ADV_IMP.2 Implementation of the TSF

132 Developer actions elements:

133 The developer shall provide the implementation representation for the entire TOE security functions.

134 Content and presentation of evidence elements:

135 The implementation representation shall unambiguously define the TOE security functions to a level of detail such that the TOE security functions can be generated without further design decisions.

136 The implementation representation shall be internally consistent.

137 The implementation representation shall describe the relationships between all portions of the implementation.

138 Evaluator actions elements:

139 The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

140 The evaluator shall determine that the implementation representation is an accurate and complete instantiation of the TOE security functional requirements.

6.2 ALC_DVS.2 Sufficiency of security measures

141 Developer actions elements:

142 The developer shall produce development security documentation.

143 Content and presentation of evidence elements:

144 The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.

145 The development security documentation shall provide evidence that these security measures are followed during the development and maintenance of the TOE.

146 The evidence shall justify that the security measures provide the necessary level of protection to maintain the confidentiality and integrity of the TOE.

147 Evaluator actions elements:

148 The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

149 The evaluator shall confirm that the security measures are being applied.

6.3 AVA_VLA.4 Highly resistant

150 Developer actions elements:

151 The developer shall perform and document an analysis of the TOE deliverables searching for ways in which a user can violate the TOE security policy.

152 The developer shall document the disposition of identified vulnerabilities.

153 Content and presentation of evidence elements:

154 The documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.

155 The documentation shall justify that the TOE, with the identified vulnerabilities, is resistant to obvious penetration attacks.

156 The evidence shall show that the search for vulnerabilities is systematic.

157 The analysis documentation shall provide a justification that the analysis completely addresses the TOE deliverables.

158 Evaluator actions elements:

159 The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

160 The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure the identified vulnerabilities have been addressed.

161 The evaluator shall perform independent vulnerability analysis.

162 The evaluator shall perform independent penetration testing, based on the independent vulnerability analysis, to determine the exploitability of additional identified vulnerabilities in the intended environment.

163 The evaluator shall determine that the TOE is resistant to penetration attacks performed by an attacker possessing a high attack potential.

Chapter 7

TOE summary specification Part 1: TOE security functions

7.1 TOE security functions

164 This chapter defines the TOE security functions, and Table Tab. 7.1 - specifies how they satisfy the TOE security functional requirements.

7.1.1 Test Mode Entry (SF1)

165 SF1 shall ensure that only authorized users will be permitted to enter Test Mode. This is provided by Test Mode Entry conditions which are required to enable the TOE to enter Test Mode.

All test entry requirements occur while the TOE is held in reset and failure in any one will prevent Test Mode Entry. It is required that the TOE satisfies the test entry conditions during any internal reset condition.

166 It is not possible to move from User Mode to Test Mode. Any attempt to do this, for example, by forcing internal nodes will be detected and the security functions will disable the ability to enter Test Mode.

167 The Strength of Function claimed for the Test Mode Entry security function is high.

7.1.2 Protected Test Memory Access (SF2)

SF2 shall ensure that, although authenticated users can have access to memories using commands in test mode, they cannot access directly their contents.

168 Only authorized design and production engineers running test on the TOE will have access to the TME conditions.

The Strength of Function claimed for the Protected Test Memory Access security function is high.

7.1.3 Test mode disable (SF3)

SF3 shall make provision for Test Mode Disable which, once activated, shall ensure that none of the test features are available, not even to authenticated users in test mode.

7.1.4 TOE Testing (SF4)

169 SF4 shall provide embedded hardware test circuitry with high fault coverage to prevent faulty devices being released in the field. Devices with manufacturing problems (short circuits, open nets, ...) could lead to a poor level of security by disabling some Security Functions.

170 Testing of Security Functions is dependent on a fault free and fully functional TOE. The RAM, ROM and standard cell logic (including the MCU) is tested by functional tests under the control of the test interface circuit. EEPROM is tested through the test interface circuit by external stimulus. CPU to EEPROM data flow will also be tested by functional tests, run from the EEPROM, using test software loaded under the control of the test interface circuit.

171 To conform with ISO 7816 standards the TOE embedded software will always return an Anwer-To-Reset command via the serial I/O port. This contains messages with information on the integrity and identification of the device. An ATR also verifies significant portions of device hardware (CPU, ROM, EEPROM and logic).

7.1.5 Data error detection (SF5)

172 SF5 shall provide means for performing data error detection.

173 Means of performing checksum error detection and parity error detection is provided. The 32-bit Checksum Accelerator or the CRC-16 hardware peripheral can be used by the embedded software to compute fast data error detection on the program and/or data memories before starting any operation.

7.1.6 FireWall (SF6)

174 SF6 shall enforce access control based on the FireWall rules as defined in the ACSF_Policy :

175 Memory protection

- The FireWall defines different modes to execute embedded software (supervisor / non-supervisor).
- The different modes provide restricted access privilege to the memories, and to the MCU peripheral registers. In case of illegal accesses performed by the embedded software, a security action is invoked.

176 Illegal Address

- If an illegal address is accessed, then a security action is invoked.

177 Illegal Opcode

- If an attempt is made to execute any opcode that is not implemented in the instruction set, a security action is invoked.

7.1.7 Event audit (SF7)

178 The TOE shall provide an Event Audit security function (SF7) to enforce the following rules for monitoring audited events :

179 - Accumulation or combination of the following auditable events would indicate a potential security violation:

1) The external voltage supply goes outside acceptable bounds,
2) The external clock signal goes outside acceptable bounds,
3) The ambient temperature goes outside acceptable bounds,
4) Application program abnormal runaway,
5) Attempts to physically probe the device,
6) Attempts to gain illegal access to reserved RAM memory locations,
7) Attempts to gain illegal access to reserved EEPROM memory locations,
8) Attempts to gain illegal access to reserved peripheral or IO register locations,
9) Attempts to execute instruction to read the program memory from the non-supervisor program location,
10) Attempts to move the RAM stack to an illegal RAM memory location,
11) Attempts to execute an AVR opcode that is not implemented,
12) Attempts to illegally write access the device’s EEPROM,
13) Attempts to gain illegal access to supervisor modes,
14) The exposition to UV light goes outside acceptable bounds.

These events are audited by IO register bits or other hardware mechanisms accessible to the embedded software.

The Strength of Function claimed for the Event audit security function is high.

7.1.8 Event action (SF8)

180 SF8 shall provide an Event Action security function to register occurrences of audited events and take appropriate action. Detection of such occurrences will cause an information flag to be set, and may cause :

- memory wiping actions,

- or different levels of immediate resets,

- or different levels of security interrupts,

to occur if the violation warrants such action.

181 Event Action depends on the type of Event (see [TD] for more information).

7.1.9 Unobservability (SF9)

182 SF9 shall ensure that users/third parties will have difficulty observing the following operations on the TOE by the described means.

1) Extracting information, relating to any specific resource or service being used, by monitoring power consumption

2) Extracting information, relating to any specific resource or service being used, carrying out timing analyses on cryptographic functions

3) Extracting information, relating to any specific resource or service being used, by using mechanical, electrical or optical means, in order to examine the topology of the TOE, including address and data buses and regular structures.

The Strength of Function claimed for the Unobservability security function is high.

7.1.10 Cryptography (SF10)

183 The TSF shall provide a cryptographic algorithm to be able to transmit and receive objects in a manner protected from data retrieval or modification.

184 The TSF shall provide hardware DES, TDES data encryption/decryption capability, and SHA-1 data signing capability. The TSF shall also provide hardware RSA without CRT (i.e. modular exponentiation) data encryption/decryption capability, as well as RSA with CRT data encryption/decryption capability.

185 Those may be used by the smartcard embedded software to support data encryption and decryption for maintaining data integrity, and protect against sensitive data unauthorized disclosure.

186 The TSF shall provide a Random Number Generator (RNG) to support security operations performed by cryptographic applications. This RNG shall not be predictable, have sufficient entropy, and not leaking information related to the value of the generated random numbers as this leakage could be used to retrieve cryptographic keys for instance.

187 The TSF shall provide RSA cryptographic key generation capability using Miller Rabin algorithm with confidence criteria (t parameter) between 0 and 255.

188 An assessment of the strength of the DES algorithm does not form part of the evaluation.

189 An assessment of the strength of the TDES algorithm does not form part of the evaluation.

190 An assessment of the strength of the SHA-1 algorithm does not form part of the evaluation.

191 An assessment of the strength of the RSA without CRT algorithm does not form part of the evaluation.

192 An assessment of the strength of the RSA with CRT algorithm does not form part of the evaluation.

193 An assessment of the strength of the Miller Rabin algorithm does not form part of the evaluation.

194 The TSF shall also provide cryptographic primitives to ease the customer proprietary software implementation of these algorithms (multiply, square, ...) as well as DSA and EC-DSA data signature in the AVR embedded software. As the TOE does not include any embedded software, these cryptographic primitives are out of its scope.

7.1.11 Security Functions based on permutations/combinations

195 The description of the security functions using permutations and/or combination properties is not disclosed in this ST-lite document.

Further details on these mechanisms and on the Strength of Function Analysis performed by ATMEL can be found in [SOF].

Tab. 7.1 -Relationship Between Security Requirements and Security Functions

		SECURITY FUNCTIONS									
		Test Mode Entry	Protected Test Memory Access	Test Mode Disable	TOE Testing	Data Error Detection	FireWall	Event Audit	Event Action	Unobservability	Cryptography
SECURITY REQUIREMENT		**SF1**	**SF2**	**SF3**	**SF4**	**SF5**	**SF6**	**SF7**	**SF8**	**SF9**	**SF10**
FIA_UAU.2	**O1**	X									
FIA_UID.2	**O2**	X									
FIA_ATD.1	**O3**	X	X	X			X				
FPT_TST.1	**O4**	X	X	X	X	X					
FDP_SDI.1	**O5**				X	X					
FMT_MOF.1	**O6**	X		X							
FMT_MSA.1	**O7**	X	X				X				
FMT_SMR.1	**O8**	X		X			X				
FMT_MSA.3	**O9**	X	X	X			X				
FDP_ACC.2	**O10**		X				X				
FDP_ACF.1	**O11**		X				X				
FDP_IFC.1	**O12**		X		X						
FDP_IFF.1	**O13**		X		X						
FAU_SAA.1	**O14**							X			
FPR_UNO.1	**O15**									X	
FPT_PHP.2	**O16**							X	X		
FPT_PHP.3	**O17**							X	X		
FCS_COP.1	**O18**										X
FCS_CKM.1	**O19**										X

TOE summary specification Part 2: TOE assurance measures

7.2 TOE security assurance measures

196 This chapter defines the TOE assurance measures and Table Tab. 7.2 - specifies how they satisfy the TOE security assurance requirements.

7.2.1 Security target (SA1)

197 SA1 shall provide the “VEGA2 Security Target” document plus its references.

7.2.2 Configuration management (SA2)

198 SA2 shall provide the “VEGA2 CC Configuration Management (ACM)” interface document plus its references.

7.2.3 Delivery and operation (SA3)

199 SA3 shall provide the “VEGA2 CC Delivery and Operation (ADO)” interface document plus its references.

7.2.4 Development Activity (SA4)

200 SA4 shall provide the “VEGA2 CC Development Activity (ADV)” interface document plus its references.

7.2.5 Guidance (SA5)

201 SA5 shall provide the “VEGA2 CC Guidance (AGD)” interface document plus its references.

7.2.6 Life cycle support (SA6)

202 SA6 shall provide the “VEGA2 CC Life Cycle Support (ALC)” interface document plus its references.

7.2.7 Test Activity (SA7)

203 SA7 shall provide the “VEGA2 CC Test Activity (ATE)” interface document plus its references, and undertaking of testing described therein.

7.2.8 Vulnerability Assessment (SA8)

204 SA8 shall provide the “VEGA2 CC Vulnerability Assessment (AVA)” interface document plus its references, and undertaking of vulnerability assessment described therein.

7.2.9 Smartcard devices (SA9)

205 SA9 shall provide functional VEGA2 smartcard devices.

7.2.10 Development site (SA10)

206 SA10 shall provide access to development site.

7.2.11 Test site (SA11)

207 SA11 shall provide access to test site.

7.2.12 Manufacturing site (SA12)

208 SA12 shall provide access to manufacturing site.

7.2.13 Sub-contractor sites (SA13)

209 SA13 shall provide access to sub-contractor sites.

Tab. 7.2 -Relationship Between Assurance Requirements and Measures

		Security Target	Configuration Management	Delivery and Operation	Development Activity	Guidance	Life Cycle Support	Test Activity	Vulnerability assessment	Smartcard Devices	Development Site	Test Site	Manufacturing Site	Sub-contractor Site
ASSURANCE REQUIREMENT		**SA1**	**SA2**	**SA3**	**SA4**	**SA5**	**SA6**	**SA7**	**SA8**	**SA9**	**SA10**	**SA11**	**SA12**	**SA13**
ASE_xxx		X												
ACM_AUT.1			X								X	X	X	X
ACM_CAP.4			X								X	X	X	X
ACM_SCP.2			X								X	X	X	X
ADO_DEL.2				X							X	X	X	X
ADO_IGS.1				X							X	X	X	X
ADV_FSP.2					X									
ADV_HLD.2					X									
ADV_IMP.2					X									
ADV_LLD.1					X									
ADV_RCR.1					X									
ADV_SPM.1					X									
AGD_ADM.1						X								
AGD_USR.1						X								
ALC_DVS.2							X				X	X	X	X
ALC_LCD.1							X				X	X	X	X
ALC_TAT.1							X				X	X	X	X
ATE_COV.2								X		X		X		
ATE_DPT.1								X		X		X		
ATE_FUN.1								X		X		X		
ATE_IND.2								X		X		X		
AVA_MSU.2									X	X				
AVA_SOF.1									X	X				
AVA_VLA.4									X	X				

Chapter 8

PP claims

8.1 PP reference

210 This Security Target is compliant with CC Smartcard Integrated Circuit Protection Profile PP/9806, Version 2.0, Issue September 1998, and has been registered at the French Certification Body.

8.2 PP refinements

211 None.

8.3 PP additions

8.3.1 Cryptographic capability

212 In addition to conforming to PP/9806, this Security Target specifies an additional Organisational Security Policy P.CRYPTO in section 3.4.

213 The CC security functional requirements to meet this Organisational Security Policy are Cryptographic Operation (FCS_COP.1) and Cryptographic key generation (FCS_CKM.1), which are specified in Chapter 5 of this Security Target.

214 The security function to satisfy the FCS_COP.1 and FCS_CKM.1 requirements is SF16 and is specified in Chapter 7 of this Security Target.

Annex A

Glossary

Control Bytes

Reserved bytes of EEPROM which can be programmed with traceability information.

IC Dedicated Software

IC Proprietary software which is required for testing purposes and to implement special functions. For VEGA2 this includes the embedded test software and additional test programs which are run from outside of the IC.

The Crypto libraries also form part of the IC dedicated software.

IC Designer

Institution (or its agent) responsible for the IC Development. Atmel is the institution in respect of the TOE.

IC Manufacturer

Institution (or its agent) responsible for the IC manufacturing, testing and pre-personalization. Atmel is the institution in respect of the TOE.

IC Packaging Manufacturer

Institution (or its agent) responsible for the IC packaging and testing.

IC Pre-personalization Data

Required information to enable the smartcard IC to be configured by means of ROM options and to enable programming of the EEPROM with customer specified data.

Personalizer

Institution (or its agent) responsible for the smartcard personalization and final testing.

Smartcard

A credit sized plastic card which has a non volatile memory and a processing unit embedded within it.

Smartcard Embedded Software

Software embedded in the smartcard application (smartcard application software). This software is provided by smartcard embedded software developer (customer). Embedded software may be in any part of User ROM or EEPROM.

Smartcard Embedded software is not applicable in the case of the TOE since it is a hardware evaluation only.

Smartcard Embedded Software Developer

Institution (or its agent) responsible for the smartcard embedded software development and the specification of pre-personalization requirements.

Smartcard Issuer

Institution (or its agent) responsible for the smartcard product delivery to the smartcard end-user.

Smartcard Product Manufacturer

Institution (or its agent) responsible for the smartcard product finishing process and testing.

UNIX

Interactive Time Sharing Operating System.

Abbreviations

CPU	Central Processor Unit
EEPROM	Electrically Erasable Programmable ROM
HCMOS	High Speed Complementary Metal Oxide Semiconductor
IC	Integrated Circuit
I/O	Input/Output
MCU	Microcontroller
NVM	Non Volatile Memory
OTP	One Time Programmable
RAM	Random-Access Memory
RNG	Random Number Generator
ROM	Read-Only Memory
SEF	Security Enforcing Function
TOE	Target of Evaluation
TME	Test Mode Entry