



Joint Interpretation Library

The Application of ITSEC to Integrated Circuits

Version 1.0
January 1999

Table of contents

1	Introduction.....	1
1.1	Objective	1
2	Application of ITSEC to Integrated Circuits.....	3
2.1	Introduction.....	3
2.2	Assurance Correctness	4
2.2.1	Security Target	4
2.2.2	Architectural Design.....	6
2.2.3	Detailed Design.....	8
2.2.4	Implementation.....	10
2.2.5	Configuration Control.....	13
2.2.6	Programming Languages and Compilers	14
2.2.7	Developers Security	16
2.2.8	User Documentation	18
2.2.9	Administration Documentation.....	18
2.2.10	Delivery and Configuration	18
2.2.11	Start-up and Operation	19
2.3	Assurance Effectiveness.....	20
2.3.1	Suitability of the Functionality	20
2.3.2	Binding of the Functionality.....	20
2.3.3	Strength of Mechanisms	20
2.3.4	Assessment of Construction Vulnerabilities.....	21
2.3.5	Ease of Use	22
2.3.6	Assessment of Operational Vulnerabilities.....	22
3	Glossary.....	23
4	References	25

1 Introduction

1 Through the increasing use of information technology, not only society has changed but also methods of storage and processing of information. As a result of the packaging density on silicon, in the form of a microchip and improvements in performance, it is possible to process more and more information in parallel and at speed.

2 Complex microchips, which are able to process information, unfortunately introduce risks and dangers as well as huge advantages. Dependence on trouble-free functioning, as well as on the effectiveness of the protection measures which have been carried out at the system and chip levels, has grown a great deal.

3 One must therefore be aware of the increased opportunities to test important information systems, including hardware against accepted criteria, in order to make the assurance of the security measures more transparent to the manufacturer, operator and user.

4 This publication has been developed by the BSI certification body (BSI - Bundesamt für Sicherheit in der Informationstechnik, BSI) and agreed by the JIWG (Joint interpretation working group). According to the BSI law [BSIG], the BSI has responsibility for evaluating the security of IT systems or IT components (in the sense of an IT product) and issuing security certificates (§3 Abs.1 No. 3 BSIG) and is involved in many activities for the promotion of IT security.

5 This publication has been reproduced in the Framework of the basic work for Certification. It should serve as a handbook for the application of ITSEC to hardware components in respect of integrated circuits. This document will be of particular interest to manufacturers, evaluators and certifiers.

1.1 Objective

6 It is important that a certification authority achieve evaluation consistency and comparability of the results: the prerequisite is that ITSEC be applied consistently by evaluators and certifiers.

7 The security properties of both the hardware products and the software products can be certified in accordance with ITSEC (cf. ITSEC 1.2). The wording of ITSEC, however, relates in many ways to software, and the terminology makes the translation to hardware not immediately recognisable. This has led to ITSEC mainly being applied to software. However, it has increasingly become generally accepted that hardware needs to be included within the scope of ITSEC evaluations.

8 The following comments on the individual aspects of ITSEC work packages should support the application and the consistent interpretation of ITSEC to integrated circuits up to evaluation level E4. ITSEC, ITSEM and ITSEC-JIL (see chapter 5, Sources) should be consulted for software aspects, as this document touches on software only for the purposes of clarification and comparison.

2 Application of ITSEC to Integrated Circuits

2.1 Introduction

9 In applying ITSEC to hardware components, two types of Target of Evaluation (TOE) should be considered:

- a TOE produced from a series of discrete parts on a printed circuit board or as a hybrid through different or several dice on one carrier.
- a TOE produced as an individual integrated circuit (IC).

10 The following guidance concerning the ITSEC assurance aspects for a TOE is applicable to the hardware of single ICs.

11 In general, logical functionality in an IC can be implemented in simple PLD structures (Programmable Logic Device), FPGA structures (Field Programmable Gate Array), as an ASIC (Application Specific Integrated Circuit), or as well as a customer IC.

12 In respect of security applications, intelligent memory ICs with a hard-wired security logic (e.g. phonecards) or micro-controller based ICs in an ASIC design (e.g. German EC-electronic purse) are used mainly for the elementary core components of security functionality.

13 Contemporary memory in ICs is based on EPROM or E²PROM cells, which makes the nonvolatile storage of data possible. Security logic can be utilised to implement identification and authentication, access control and internal IC sequence controls.

14 Micro-controller-based ICs offer the possibility of carrying out independently complex processes controlled by an IC operating system. Items which also belong to the aforementioned functionality are accountability functions and services such as encryption and digital signature, functionality which is implemented in hardware as well as software.

15 The mechanisms for the protection of software and operational data in various memory and the internal sequences are realised through the hardware of an IC (e.g. by means of certain technical measures and technological features) in order to support logical functionality.

16 So, for instance, the operating system of a micro-controller IC is contained (placed) in a ROM and/or an E²PROM memory, and is protected from disclosure and modification during the operational phase by the technical or technological properties of the hardware. While technological properties are inherent in a TOE, technical properties do not have to be inherent in a TOE.

2.2 Assurance Correctness

2.2.1 Security Target

17 Irrespective of whether the TOE comprises software and/or hardware, the Security Target should provide specification of the security properties of an IT product at an abstract level, as well as a description of the product and its intended environment. This specification is independent of the method of implementation. Nevertheless, in the following pages some observations concerning individual requirements are made.

2.2.1.1 Product Description / Identification of the TOE (E1 – E4)

18 An IC can be regarded as a product in the ITSEC sense. In keeping with a classical Software-TOE, the IC needs to be clearly identified, and separated in its technical and operational environment.

19 Since the hardware parts on an IC are both physically and functionally difficult to separate from one another, without additional information it is not possible to exclude parts of the hardware from the TOE; it is therefore sensible to define the whole of the IC hardware as part of the TOE. In the case of certain parts of the IC being outside the TOE, a clear, logical and physical interface must be recognisable (cf. also ITSEC-JIL Chapter 4.1.2). The inclusion of strongly hardware-orientated software/firmware into the TOE is appropriate. It would be sensible to look at an IC in its entirety and not only at the hardware or only at the software.

20 The point in time at which the TOE comes into being during its manufacturing process life cycle must be determined. In contrast to purely software TOEs, as in the cases noted here, this determination is only possible with precise knowledge of the manufacturing process. However, this determination also has a direct influence on the threat and attack scenarios in the operation of the TOE which could be adopted in the context of the evaluation of the TOE. In the case of an IC TOE, this could be after the testing of the IC as a die (at the earliest), or upon completion of packaging and associated testing. In the event that the TOE also comprises a personalisation phase, this must be included within the scope of evaluation, too.

2.2.1.2 Product description / intended method of use of the product (E1-E4)

21 For the method of using the TOE operationally, it is of interest to find out who is able to use the IC after delivery and in what operational modes it is possible to use it. The relevant subjects, objects and methods of access should be defined in order to be able to gain a better understanding of methods of use.

22 In this context, the actions of all personnel who come into contact with the TOE after delivery need to be examined.

2.2.1.3 Product description / planned intended environment (E1-E4)

23 With regard to 2.2.1 (2), corresponding requirements should also be formulated. Otherwise there is no specific interpretation of ITSEC.

2.2.1.4 Security policy / Security objectives (E1-E4) / Formal Security Model (E4)

24 This does not differ from the examination of a software TOE, however, security objectives should also address technical and technological properties of the IC.

25 The Formal Security Model (from E4 on) is a formal description of the security policy using appropriate formal languages (cf. ITSEC-JIL, chapter 19). It is utilized on an abstract level independently from the TOE implementation in hardware or software. There is no specific interpretation of ITSEC.

2.2.1.5 Assumed threats (E1-E4)

26 The threats are directed at the security objectives identified in 3.2.1 (4).

27 Beside logical functionalities, technical and technological properties can be attacked during the operational phase of the TOE, too. The corresponding threats can therefore be formulated (e.g. selection of objects also by means of physical attacks, operation of the TOE outside specific parameters, such as voltage, frequency and temperature).

28 With respect to determining specific threats, it should be noted that attacks on ICs during production processes, and in particular during test phases, are possible; they can become apparent during the operational phase in the form of vulnerabilities (cf. Developers Security, Vulnerability Analysis).

2.2.1.6 Definition of Security Enforcing Functions

29 For ICs, it is important that aspects of design and implementation which are not necessarily functional in nature be addressed by evaluation. Thus the concept of Security Enforcing Functions (SEFs) has been extended beyond the traditional logical and functional meaning to embrace technological or technical properties implemented of an IC (cf. ITSEC-JIL, chapter 17.2.1). Technological properties are inherent to the TOE, whereas technical properties need not necessarily be so.

30 The following types of SEFs should be considered:

- SEFs in the form of logical functionality¹ (e.g. authentication logic) which are directed against logic attacks; they counter the threats directly.
- SEFs which, in addition to being logical functionality, comprise technical and technological properties, too. These functions can be implemented by a combination of passive structure with active logic (e.g. monitoring of supply voltage by means of integrated CMOS technology parameters in certain areas of the IC).
- SEFs in the sense of technical properties which make an attack more difficult (e.g. security which has purposely been built in). Memory circuits and function elements of the hardware are protected from disclosure and

1. Meant are: functionalities realized in hardware, but comparable to software functionalities.

modification with the help of such technical properties. These technical properties are the result of measures employed in the development/production process. SEFs in this category comprise: bus obscurity (to guard against spoofing), protective layering (passivation of the metallisation layer).

31 All of the above types of SEF should be taken into consideration during evaluation.

2.2.1.7 For effectiveness of the functionality and the counteracting of threats

32 Effectiveness analyses should consider the combination of logical, technological and technical SEFs.

2.2.2 Architectural Design

33 For each evaluation level, the architectural design must contain, among other things, the general structure and the external interfaces at a high-level of abstraction, with a subdivision into the main components. In this case a block diagram, which originates in the design and conception phase, as well as an informal description, can be an integral part of the architectural design.

2.2.2.1 Basic structure of the TOE (E1-E4)

34 Since the security properties of an IC TOE can consist of logical functionality as well as technical and technological properties, it is necessary to bring out the general structure of the fundamental components and also to explain the technical and technological structure because of their importance to the hardware security properties and evaluation.

35 In many cases, components which represent the general structure of an IC TOE can be definite logical units; they are possibly even implemented as a physical unit on the IC. Examples comprise: memory, data/address bus-memory interface, arithmetic block, contact interface, watchdog timer, sensors with analysis logic, controls for the voltage supply, logic blocks for access controls or authentication for memory ICs with security logic, a micro controller block on micro controller ICs.

36 A protective layer could, for example, be seen as a component of the general structure of the physical composition of the TOE.

2.2.2.2 External interfaces of the TOE (E1-E4)

37 External interfaces can be:

- interfaces to the hardware parts of the IC which are not part of the TOE (provided that the TOE is separated in this way), or
- interfaces to the software/firmware which is stored on the IC and is not part of the TOE, but which runs on the IC hardware under consideration (e.g. the triggering of an interrupt via hardware), or

- logical and physical interfaces of the IC (IC contacts), which guarantee the connection to the outside world.
- Simply observing external interfaces from the point of view of their logical behaviour is unlikely to be sufficient. Externally adjustable operational parameters and their limits should also be investigated because direct attacks or vulnerabilities may derive from it.

2.2.2.3 Hardware and firmware required by the TOE (E1-E4)

38 Since the TOE itself consists of hardware, further items of hardware may be required for its operation. It is noteworthy that an external voltage and timing supply, or a defined data interface can be necessary for the operation of the TOE. If it is the case that not all of the IC hardware is to be included in the TOE (e.g. there may be hardware parts on the IC which are needed for the TOE's subsequent operation) they should not be overlooked by evaluators in case vulnerabilities may occur.

2.2.2.4 Functionality of the supporting protection mechanisms (E1-E4)

39 If supporting external protection mechanisms are available, a clear distinction between TOE internal mechanisms and external mechanisms is necessary. While doing so, dependencies within the hardware or possibly through the firmware which is involved should be taken into consideration. Only the TOE internal mechanisms are subject to the evaluation requirements of ITSEC.

40 A check sum algorithm could be implemented in firmware which may not be within the scope of the TOE but the result of the check used by the TOE. An exact description of the dependencies and the behaviour of the interface are necessary.

2.2.2.5 The separation of the TOE into security enforcing and other components (E2-E4)

41 At E2 and above, the separation of security enforcing components from other components of the TOE or from components of the intended environment should be examined carefully. This is because within the IC itself there are strong dependencies between various physical components at the implementation level, which complicate an effective separation in the ITSEC sense. Consequently, it is mostly necessary to classify as security enforcing all of the components of an IC TOE at the level of the architectural design.

42 In particular, a rationale how the chosen structure provides for largely independent security enforcing components (required at E4 and above) should be based on logical and physical dependency.

43 A maximal independence of components within an IC TOE could be possible if there were no or only minimal physical overlaps and logical dependencies between the individual components and the interfaces are clearly defined.

2.2.2.6 Semi-formal notation (E4)

44 E4 semi-formal description of the architecture can take the form of block circuit diagrams or hardware description language documents (HDL – hardware description language). In many cases, however, a hardware description language would first be used at the Detailed Design level.

45 Meaningful graphical representations of the technical or technological properties as part of the SEFs of the TOE may be considered equivalent to semi-formal representation.

46 The informal documentation of the technical and technological properties, as well as their integration into the structure and the realisation of SEFs, is necessary.

2.2.2.7 Evidence of how the SEFs of the Security Target shall be provided (E1-E4)

47 Assigning SEFs to components is especially difficult, since individual components not only provide the realisation of a single SEF and very strong interactions and dependencies between components exist. For this reason, a description of the functional flow of SEFs to defined components is of particular significance.

2.2.3 Detailed Design

48 A hardware description language (HDL) will usually be used at this level of representation of the IC. The construction plans and functional descriptions, which result from the use of an HDL tool and a CAD tool, can, according to ITSEC, be used or brought into play directly during the construction of the detailed design, but needs only to be presented fully from E3.

49 If necessary, the evaluator needs to be in a position to be able to use the tools provided by the manufacturer in the evaluation.

50 The adjustment of the parameters for the production process will be determined during this phase by means of CAD tools under the consideration of technology specific characteristics.

51 The design elements necessary for the construction of the TOE are:

- logical plans which for example consist of analogue cells, standard cells, gates, transistors and diodes, in order to realize individual functionality as well as SEFs,
- layout plans which describe the organisation of the physical components with regard to the process masks and determine the metallisation masks,
- mask plans which are necessary for the technological process. The mask plans need only be shown from E3 on in certain cases when they are used in vulnerability analyses.

- 52 The technical and technological structure of the TOE is to be refined, to make an effectiveness analysis possible during the examination of physical attacks on the TOE in the context of the assumed threats.
- 2.2.3.1 Basic components (ITSEC, 4.21 for levels E2-E4) and their specifications in the detailed design (E3.8 and E4.8 for E3-E4), interfaces
- 53 A basic component is a component which is identifiable at the lowest hierarchical level of specification in the Detailed Design (ITSEC 6.10). In the case of an IC, the actual logic and layout plans will have been derived from basic components, whereby the separation of the basic components has to fulfil the testing requirements. This means that the interfaces of the basic components (from E3 on) and the basic components itself (from E4 on) need to be testable. Supplementary to this, it should be noted that basic components at E2 level serve simply to refine the structure of the TOE.
- 54 Interfaces between components must be described especially carefully at the level of basic components, since there are strong dependencies between basic components in an IC (E3-E4). Functionality which runs in parallel should be considered with the description of the interfaces.
- 55 The timing of the basic component interfaces should be described if they are accessible from the outside (e.g. pads) for tests.
- 2.2.3.2 Semi-formal notation
- 56 E4 semi-formal descriptions may be made available in the following forms: refined block circuit diagrams, flowcharts, conditional diagrams, truth tables, documents using a hardware description language (HDL). Basic components must be indicated and described.
- 57 Meaningful graphical representations of the technical or technological properties as part of the SEFs of the TOE as well as layouts may be considered equivalent to semi-formal representation.
- 2.2.3.3 Realisation of the SEFs by the security mechanisms (ITSEC, En.9, n>1)
- 58 If security properties are to be used for the realization of SEFs which arise as a result of certain design or layout requirements or because of demands on technology, they must be described with sufficient precision in connection with affected components, interfaces or mechanisms. This is of particular significance to subsequent effectiveness analyses.
- 2.2.3.4 Description of the mechanisms of the security functions
- 59 Mechanisms of logical functionalities can be described similar to mechanisms of a software TOE.

60 If technical and technological properties are part of a specified SEFs of the TOE, the effect of the technical and technological properties in the operation of the TOE need to be provided in the sense of a description of a mechanism.

2.2.4 Implementation

61 Test documents produced according to ITSEC must, for E3 and above, contain logical plans and layouts. Layouts are required to check the correctness of the implementation of technical and technological properties. The mask plans need only be presented in certain cases, if they are necessary for follow-up analyses. The layout indicates the ease with which physical attacks can be mounted and for example the accessibility of the metallisation layer. For the conduct of tests, IC data sheets are of particular importance.

62 ITSEC-JIL Chapter 7.2. should be applied appropriately. This means that hardware bases, as well as the HDL bases, should be used for: testing of correspondence between detailed design and implementation, carrying out additional tests, examination of the absence of functional properties and for examination of test coverage.

63 The evaluator must be in a position to repeat the manufacturer's tests and perform additional tests. For the conduct of tests, the evaluator needs test vectors which determine the course of tests. If required, the evaluator must be in the position to use the tools for evaluation applied by the manufacturer. In many cases, owing to tool availability, this will only be possible in the development laboratory or during production by the manufacturer. In these cases it is sufficient for the evaluator to witness the tests at the manufacturer.

64 Correctness tests can also be implemented at the design level with the help of HDL tools.

65 Apart from the functional tests under standard conditions, tests (if necessary real time tests) under defined stress conditions (temperature, frequency, voltage, EPROM cycle tests etc.) are also to be planned, since such conditions could arise during the operation of the TOE (comparable with extreme situations for software TOEs, which could lead to run-time errors).

66 Tests of individual components of the TOE, or the control of certain technical or technological properties, could only possibly be implemented at a certain time during the manufacturing process or only in test mode, since the respective physical components can be neither logically nor physically accessed after the end of the production of the TOE. This should be considered during test planning and be appropriately documented.

2.2.4.1 The test documentation shall contain plan, purpose, procedures results of the tests (E1-E4)

Test plan:

67 A test plan determines the framework of the test cases. In the test plan, the exact specification and scope of the test cases, as well as the documentation describing all input and environment parameters of the IC, are of great importance. These parameters are partially given in data sheets. Therefore the data sheet must be an integral part of the test documentation.

68 The test cases can vary greatly with analogue and digital circuits.

69 Test parameters must be taken into consideration in the test planning. They could be for example:

- test frequencies with minimum and maximum limits
- voltage supply corresponding to the data sheet
- test temperatures
- test vectors for the selection of the test areas in the IC

70 Nevertheless, equipment which is necessary for a test case must be specified exactly with all adjustments. This also includes the precise identification of the test libraries for simulation as well as the driver program for the test equipment.

Test objective:

71 The objective of the test is to give evidence for the correctness of the logic by means of simulation using the HDL tool and to test the correctness of the implementation. Since a test is a type of quality control, after simulation it must be proven whether the implementation has been successful. Individual tests on the finished IC must show that the implementation of the security functions and mechanisms is correct, and that the timing requirements are fulfilled. During mechanism testing, binding of components is of particular note, especially if there is parallel functionality.

72 If the manufacturer would like to do without simulation by means of the HDL tools, all tests must be carried out in real time in order to give evidence that the implementation be correct.

Test procedure:

73 The test procedure is the instigation of the test plan using the planned test vectors. In the event that a logical part or memory area should be locked, this must be explained or described, together with the circumstances for locking.

74 From E3 on, even the methodology of fusing, which is used for deactivation of test hardware or for the transition from the test- or initialisation mode to the operational mode, needs to be described in detail, so that, within the framework of the effectiveness testing, the resistance against attacks can be judged.

75 If functionality is implemented using technical and technological properties, it may be sensible, to prove the evidence of a characteristic not only by functional testing,

but it may be sufficient to verify the presence of a special technical or technological design obscurity from analysis of the design (e.g. by optical inspection).

Test results:

- 76 The test results, which will be obtained on special test equipment, must be presented in a form that can be analysed (analogue tests, timing tests).
- 77 For test results concerning functionality which runs in parallel, the assignment of the results to specific basic components, SEFs and mechanisms is of significance. The dependencies of the results of the tests resulting from parallel processing functionality should be explained.
- 2.2.4.2 The library of test programs shall contain test programs and tools to enable all tests covered by the test documentation to be repeated (E1-E4)
- 78 Driver software, among other things, with its associated equipment (tester) is required for the testing of the chip. This is also necessary for repeating tests.
- 79 Other tools that have been used, such as the logic analyser, oscilloscope, debugger, operating system etc. also need to be stated.
- 2.2.4.3 The description of correspondence shall describe the correspondence between source code, or hardware drawings and basic components of the detailed design (E3-E4)
- 80 The hardware construction drawings for an IC consist of logical plans and layouts (cf. also the detailed design).
- 81 The basic components need to be traced down on the logical plan and on the layout.
- 2.2.4.4 The test documentation shall contain a justification why the extent of test coverage is sufficient (E4)
- 82 Appropriate test coverage (ITSEM 4.5.72) is achieved when all instructions (E3/E4) and branches (E4) of the whole logical plan (“source code”), which belong to security enforcing and security relevant basic components, have been tested.
- 83 With tests at the design level by means of simulation using HDL, at least every statement of the security-enforcing parts of the HDL code must be tested.
- 84 The correctness of the implementation (integration) and the test coverage must also be proven after production. The test vectors must be chosen appropriately, so that they cover the requirements mentioned above completely (E3 or E4). Analyses should be performed in this way.
- 85 The justification for test coverage, required of the designer, can be done with respect to one of the following items:

- The manufacturer can, if possible, show that he has toggled each junction of a basic component during testing.
- If, according to the logical plan, the basic components or parts of the basic components can only be tested in parallel, the manufacturer must show that all junctions have been toggled at least once via the underlying test vectors.
- If the designer has not taken the testability rules (undo influence permitting) into consideration, or the testing of some basic components is not immediately possible (the testing of a timer over 24 hours), the circuit cannot be considered 100% tested. In this case, test coverage is achieved if the manufacturer can show that all junctions were achieved via the test vectors and that there are no conditions which compromise security.

86 Regarding test coverage, attention must be paid to the inclusion of security functions which result from design or technology.

87 If during operation of the TOE external HW or SW functionality be included dynamically in the functional flow, then the relationship of the external components to the level of the external interface should be tested.

2.2.4.5 Additional tests to search for errors are to be performed (E2-E4)

88 The additional evaluator tests must be performed at the level of the detailed design, logical plans (source code) and layouts (E3-E4).

89 The evaluators must also perform additional tests on a completed IC (final part), because:

- An errors can be introduced by technology and may not be detected by logic tests (cf. the aging process in chapter 3.3.4),
- The scattering of security-enforcing and security-relevant parameters cannot be tested via simulation. Such scattering can only be carried out by means of testing several ICs. In order to do this, the evaluator has to select an appropriate sample or he has to fall back on the results of the manufacturers quality tests. For example, the scattering of a mistake in digitalisation can only be detected if several ICs are tested, as assembly of basic components can lead to a timing deviation.

2.2.5 Configuration Control

90 The configuration control system and the acceptance procedures should be considered during the whole development and production process of the TOE. They must be in a position also to control the construction plans and hardware parts, in addition to all relevant data files for all development steps. If applicable, various development and production sites are also to be included in this.

91 The TOE must be clearly identifiable in accordance with ITSEC En.15, n>1. However, in certain cases it can be necessary, in order to make an attack more

difficult, to mark the ICs (or the chips held within the ICs) with not visible logos or IDs. In such cases, the manufacturer must, however, find a suitably hidden possibility for the label on the TOE, such as for example in a non-deletable access-protected area of memory.

- 92 In accordance with ITSEC-JIL, Chapter 9.1.2, for E2 and above, all of the parts which can lead to a change in the TOE should also be cited in the configuration list. All of the parts which are necessary for the creation and testing of the TOE should also be listed. That includes all test equipment, libraries and the list of test vectors used during testing.
- 93 The identification and listing of basic components in the configuration list seems to be difficult to apply to ICs. Since, within the framework of the design, function blocks can be taken out of a developer's HDL or CAD library and out of the IC manufacturer's technology library (lists of the technology parameters), at the very least the libraries that are used, together with the possible parameters that are used, must be clearly identifiable if the individual components in the library do not have their own identifier.
- 94 If the hardware design is carried out with the aid of a computer, the design documents are presented in the form of data files. Tool-supported configuration control can succeed in any case, at the level of the data files, as it is also implemented in software development environments. All of the relevant data files detailing all of the development steps must be included for the purpose of reproducibility.
- 95 The E4 requirement on the evaluator to re-build selected parts of the TOE has the objective of testing effectiveness of the configuration control system with regard to the various versions and changes to the TOE. This "re-building" of the TOE is possible using the appropriate design tools (HDL, CAD tools) under the control of the configuration control system.
- 96 It is not possible to relate this requirement directly to the technological process of a customer specific IC, because the process cannot be repeated for the benefit of the evaluator. In this case, the evaluator must, however, audit the configuration control in the technological process in order to guarantee that the correct masks, which belong to a particular version of the TOE, are used and organisational measures are effective in the process.
- 97 In the case of programmable standard ICs (PLD, FPGA), in which the hardware configuration is programmed via firmware, the requirement for "re-building" can be done by programming of a new IC. The evaluator then conducts comparison tests of the functions of the newly produced IC with the original TOE (to be equated with a 'file compare' for a re-built software TOE).

2.2.6 Programming Languages and Compilers

- 98 The test aspect of the languages and compilers is, according to ITSEC 4.25, explicitly oriented to software and firmware. Nevertheless, it is both sensible and

possible to apply this aspect, as an analogy, to the hardware TOE and especially here to an IC TOE.

- 99 In accordance with ITSEC En.18/19/20 (n>2), for languages and compilers for software TOEs, it is a question of testing whether the implementation languages which have been used are clearly and unambiguously defined and documented, and whether all options of the language (and from E4 also the options of the compiler) have been documented. The objective here, apart from a higher assurance into the correct implementation of the TOE, is also to ensure repeatability of the construction of the TOE.
- 100 In order to achieve the objective intended by ITSEC for hardware TOEs, it is necessary to document and test the hardware description languages (HDL), representation elements (graphical logic elements) and tools (HDL-compiler, simulation tools, CAD tools) used in the hardware, with regard to clear and precise definition and to options used.
- 101 In the case of software, various compilers can create different object codes even with the same functionality of the TOE (i.e. with the same logical design). Functionality is defined by the processor commands and the compiler options that are used.
- 102 In the case of microchip ICs different masks, and therefore different physical implementations, can arise as a result of different technologies, even with the same functionality (i.e. with the same logical design at the level of the circuit diagram). Functionality is defined finally only by means of the cell structure which has been implemented in the silicon. As a result of this, from ITSEC level E3, the technology that has been used for the implementation of the chip, and from E4 also adjustable process parameters that have been used, needs to be presented clearly.
- 103 For the purpose of clarification, the following table shows the development processes of hardware and software:

Software	Hardware
<p>Program text input via the editor for the creation of the source file. Syntax and semantics of the input language will be determined via the compiler.</p>	<p>Creation of the logical plan through graphic input via a CAD tool or text input via a HDL-Editor. Syntax and semantics of the graphics symbols are determined via the CAD tool and the technology. For the modelling and the simulation of the circuit design a HDL will be used.</p>
<p>In order to construct a functional software TOE, several steps are required:</p> <ul style="list-style-type: none"> - Determining the compiler and linker adjustments, in order, for example, to realise certain optimisation possibilities. - Compiling and linking of the source files to a program which is executable from a processor during its run time (object files as program data files, run time library, program code for a hardware memory). - Testing and de-bugging within the framework of the compilation of individual source files as well as the whole TOE. 	<p>In order to construct a functioning IC from the design, several steps are needed:</p> <ul style="list-style-type: none"> - The construction of a netlist out of the logical plans and the synthesis of the gate structure as well as the test structure. - The simulation of this logic at the gate level and at the level of the layout using timing defaults. - The construction of the layout and the masks. - The manufacture of the microchip from this masks after several process steps, which are dependent on the semiconductor technology used (e.g. 0.8_µm CMOS-, BiCMOS- or Bipolar technology).

104 A programming language used is based on the features of a compiler, interpreter or assembler, while the logic which has been constructed using an HDL is finally only available after the conclusion of the technological process. Therefore the title of this aspect should be understood in the sense of “Languages, Tools and Technology”.

2.2.7 Developers Security

105 The intended protection measures of the integrity of the TOE and the confidentiality of the documentation should be documented in accordance with ITSEC requirement En.21 (n>1).

106 In accordance with ITSEC-JIL, Chapter 8 and Chapter 16, the production of the TOE must be taken into consideration so that the ITSEC requirement is valid for all

development phases until the delivery of the TOE. Therefore this is of particular importance because: an IC goes through various phases as well as various sites between the design and delivery stages. The requirements on the technology are finally only realized during the production of the ICs, and tests within the framework of the production come into use, too (e.g. wafer tests).

107 Furthermore, in accordance with ITSEC 4.23 and ITSEC-JIL, Chapter 8.1.2, the development and production environments should be included in the assessment which is required by ITSEC En.23 (n>1).

108 During the course of the development and production of a microchip there are several such sensitive areas:

- development (construction of the logical plans, layout, masks and drawings)
- construction of the masks
- process control (integration of the circuits onto silicon)
- product engineering (fault analysis with regard to the process)
- production and tests (production and tests of SEFs)
- assembly and tests (if necessary, e.g. bonding on a card)
- quality control for security functionality
- storage / delivery

109 The TOE is available in the various stages of development and production in various different physical shapes. The integrity of the layout masks is of particular significance.

110 Physical, procedural, personnel and other measures necessary for the realisation of the TOE's security properties, as given in the Security Target, will be transposed in the development and production and will have an effect on the security in the operational phase of the TOE. These measures are also to be documented and examined. It is here in particular that measures in the test phase, as well as the assembly phases if applicable, and measures for the management of the manufacturing process, play a role.

111 In comparison, compilation of a software TOE takes place with fixed options uniquely in the development environment (prototype and master copy). The series production of the software is simply a process of copying, in which the integrity aspects of the copy with respect to the master copy play a role.

112 With respect to IC TOEs, drawings and associated data files will be created within the framework of the development. The IC production of the prototype as well as the series is essentially more complex than a copying process in the case of software

and is variable through a multitude of process parameters, which are potentially manipulable by personnel.

113 The test phase during IC production is of particular importance, since in this phase an IC is already completely physically available, but, for example, internal IC structures are, however, adjustable or compromisable via a test mode, which is still activated.

114 Measures, which are taken, in order to mark (ink) the faulty dice on the wafer and to sort out faulty TOEs (final parts), including the criteria how to sort out, can be of importance. Measures for the destruction of defective parts should then be described.

2.2.8 User Documentation

115 The hardware designer as the user of the TOE (who either implements the IC in hardware or uses the IC for an application), needs details of the IC and its security properties with the intention of the Security Target, in order to be able to transfer the security policy correctly. An end-user has no direct contact with the IC hardware, but rather uses the chip within the framework of an application. Therefore no documentation of the security properties of the hardware is necessary for this end-user.

116 The information for an IC user should usually be found in a technical IC data sheet (e.g. description of the functionality, pin out, timing diagrams, as well as programming guidelines). For a secure use of the TOE it is necessary, in accordance with ITSEC requirement En.25, to document details of the security properties, special applications advice in the sense of the security policy and advice for the use of SEFs.

2.2.9 Administration Documentation

117 It should be decided on a case by case basis whether appropriate documentation is necessary.

118 If appropriate to the TOE, personalisation of the IC can be considered as system administration. In this respect, suitable documentation which makes it possible to administer all of the controllable security parameters and events relevant to security will be needed. In most cases, the IC leaves the manufacturer with all of the adjustments which concern security, without a change to this adjustment being necessary or even possible.

2.2.10 Delivery and Configuration

119 The interpretations in ITSEC-JIL, Chapter 10, and the lists of the delivery procedures approved by the national certification body should be followed. At E2 and above requirements for documentation of results for the generation of the TOE (which is, according to ITSEC-JIL, Chapter 16, to be regarded as installation) is not applicable to an IC, since TOEs of this type need not be installed.

- 120 According to ITSEC En.32, if various configurations are possible, the impact of the various configurations on security shall be described. Configurations on the TOE could be certain modes, such as the test and operational modes, or a choice of certain options through programming of the IC.
- 121 Functionality for the deactivation of test hardware or for the transition from test mode or installation mode to operational mode, should be considered and described since this is important in the context of vulnerability analysis and the authenticity of the delivered TOE.
- 122 In the event that within the delivery of the TOE tests of security-relevant functionalities are necessary, the results of these tests must be documented. This must be a part of the delivery procedure. Likewise procedures which cover the delivery from the development/design centre to the end-customer, including the production location, should be described.
- 123 For the examination, as demanded according to ITSEC En.34 (n>1), of the correct application of the delivery results within the framework of the evaluator actions, an examination of procedures and standards for the whole path from the development/design centre to the end-customer is necessary.

2.2.11 Start-up and Operation

- 124 According to ITSEC En.35, all procedures for secure start-up and operation need to be explained.
- 125 For secure start-up, a secure initial condition for the IC should be determined. This must occur from E2 by means of the diagnostic procedures for all of the security-enforcing hardware components (e.g. the power-on-reset procedure). A correct adjustment of all operational parameters is necessary.
- 126 As an evidence of the results of the diagnostic procedures, according to ITSEC En.36 (n>1), it is a question of the description of the signal states at the external interfaces of the TOE, or the description of the status information stored on the chip and supplied to the external interfaces of the TOE.
- 127 The procedures required from E4, in order to restore the TOE to a secure state after failure in operation, will be made available with ICs mostly through defined reset conditions (e.g. after the identification of a voltage failure).
- 128 If, in the context of the assumed threats, there is a possibility that attacks on a TOE could be successful even if it is finally out of operation, then the termination phase of the use of the TOE should be included in the evaluation (e.g. withdrawing and destruction of a chip or the object reuse of memory areas on the chip).

2.3 Assurance Effectiveness

2.3.1 Suitability of the Functionality

129 The analysis of the suitability of the functionality shall, in addition to the logical security functionalities of the TOE, also take into consideration technical and technological properties, in so far as they are defined within the Security Target. It is needed to be demonstrated which threats (transposed by certain attack scenarios) would be suitably countered by means of all types of SEF and by mechanisms.

130 By performing attack scenarios the technical peripheral conditions of the specification, such as temperature, voltage and frequency should be considered. Nevertheless, scenarios of physical attacks on the hardware can be of significance, if applicable.

2.3.2 Binding of the Functionality

131 The architectural features of the TOE have some influence in binding analysis. Here the following are of particular importance:

- physical connections between physical components in the form of signal paths and circuits
- physical connections between physical components because of the layout (i.e. that information on the technical and technological implementation needs to have some influence in the analysis)
- dynamic interweaving in the timing behaviour of individual security functions or mechanisms
- influence on binding through the setting of external signals on the microchip.

132 The effectiveness of the functionality depends on the technology used in the implementation phase. This must be taken into consideration in the binding analysis.

133 Binding of hardware- and firmware-functionalities is to be taken into consideration, depending on the separation of the TOE (cf. chapter 3.2.1, 1.).

2.3.3 Strength of Mechanisms

134 The analyses of the Strength of Mechanisms (SoM) put the attack scenarios in concrete terms and examine how the mechanisms withstand in the face of direct attacks.

135 This can be a question of the following types of direct attack:

- attacks without physical modification of the TOE, they are similar to traditional direct attacks on software but possibly involving physical means

- attacks on the mechanisms implementing technical and technological properties of the TOE without physical modification of the TOE

136 In the assessment of the SoM, the following aspects, among others, could have an influence, depending on the security objective:

- active hardware security (e.g. active sensors, I&A etc.)
- hardware security through technical and technological properties (e.g. fixed masks, bus circuits, casings, passivation, E²PROM cycles etc.)
- supporting mechanisms as firmware/software modules (e.g. check sums)

137 With respect to attacks which physically modify the internal technical structures of the TOE, it is a question of an indirect attack which needs to be examined in the context of a vulnerability analysis, since SEFs may be bypassed and therefore may lose their effectiveness.

2.3.4 Assessment of Construction Vulnerabilities

138 Vulnerabilities can be introduced in both the construction of the mechanism itself, as well as through the production of technical and technological measures intended to counter threats.

139 The process of aging of the IC should be taken into consideration during vulnerability analysis. So, for example, the vulnerabilities of an IC TOE can lie in the semiconductor technology. E²PROM cells only withstand a restricted number of program cycles. The limitation of the number of possible delete and write cycles of a cell is an inherent vulnerability, which an attacker could possibly exploit. This kind of technologically based vulnerabilities is new in contrast to software TOEs, and requires a vulnerability analysis relating to the technology and its implementation.

140 Nevertheless, vulnerabilities can possibly be exploited by means of the physical tampering of the IC or through influencing the timing of signals to the external interfaces.

141 With respect to attacks which physically modify the internal technical structures of the TOE, it is a question of an indirect attack which needs to be examined in the context of a vulnerability analysis, since SEFs may be bypassed and therefore may lose their effectiveness. Additionally, the binding of distinct components realizing mechanisms, has to be taken into consideration.

142 For example, attacks via micro-probes on IC internal signals (white box attack) could possibly be successful after a physical modification of the TOE. Protection structures in the IC, which simply make an attack more difficult, could, however, be fundamentally overcome (e.g. the etching of individual layers, the deactivation of sensors through the masquerade of the logical sensor signal). The exploitability of such indirect attacks must be viewed as a vulnerability. Specialist tools and expertise are necessary for the penetration of a HW-chip.

143 Additional vulnerabilities can be of the following types: non documented functionalities or hardware areas on the chip, covered/inference channels, radiation, power consumption, timing.

2.3.5 Ease of Use

144 If, in the framework of the configuration, an initialisation of the HW is necessary, for example through special external cabling or through certain programming, or if certain operational modes such as the initialisation and test modes need to be made explicitly ineffective by special measures, then this should be analysed with regard to the maintenance of security properties of the TOE and with regard to the ease of use for an end-user.

145 If there are types of operations among peripheral conditions, like temperature, voltage, frequency etc., which the end-user can possibly influence during operation of the TOE, then the effects on the security should be analysed.

2.3.6 Assessment of Operational Vulnerabilities

146 The operational vulnerabilities are also to be considered in the context of the use of the chip by an operating system or an application developer. For instance, the security measures which should be taken in the application development and which influence the operation of the IC could possibly be exploited by an attacker (e.g. demands on external cabling, external technical parameters, or confidentiality measures).

147 If the examination the of hardware sensors signals is not done by the TOE itself, but needs to be done by the overlying operating system, this can cause an operational vulnerability.

148 Also the possibilities of deliberately provoking faulty conditions or technical defects in the operation of the IC in order to make them exploitable for an attack can be counted amongst operational vulnerabilities.

3 Glossary

BiCMOS	Bipolar Complementary Metal Oxide Semiconductor, specific semiconductor technology
CAD	Computer Aided Design
CMOS	Complementary Metal Oxide Semiconductor, specific semiconductor technology
Die(Pl.:Dice)	individual IC on a wafer
EPROM	Erasable Programmable Read Only Memory
E ² PROM	Electrically Erasable Programmable Read Only Memory
Fusing	the calculated melting of a contact on an IC
HDL	Hardware Description Language
HW	Hardware
IC	Integrated Circuit, integrated electronic circuits in a microchip
Passivation	a protection layer on top of the metallization layer in a microchip
Pin	external contact of a packaged microchip
SW	Software
TOE	Target of Evaluation
Wafer	silicon slice for chip production

4 References

- [BSIG] Act setting up the Bundesamt für Sicherheit in der Informationstechnik (BSIG) of 17 December 1990, Bundesgesetzblatt I p. 2834
- [ITSEC-JIL] ITSEC Joint Interpretation Library ITSEC JIL, Version 2.0, November 1998
- [ITSEC] Information Technology Security Evaluation Criteria (ITSEC), Version 1.2, June 1991
- [ITSEM] Information Technology Security Evaluation Manual (ITSEM), Version 1.0, September 1993

