INTERNATIONAL STANDARD

ISO 26262-10

Second edition
2018-12

Road vehicles — Functional safety —

Part 10: Guidelines on ISO 26262

Véhicules routiers — Sécurité fonctionnelle —

Partie 10: Lignes directrices relatives à l'ISO 26262

Reference number
ISO 26262-10:2018(E)

COPYRIGHT PROTECTED DOCUMENT

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 22, *Road vehicles* Subcommittee SC 32, *Electrical and electronic components and general system aspects*.

This edition of ISO 26262 series of standards cancels and replaces the edition ISO 26262:2011 series of standards, which has been technically revised and includes the following main changes:

— requirements for trucks, buses, trailers and semi-trailers;

— extension of the vocabulary;

— more detailed objectives;

— objective oriented confirmation measures;

— management of safety anomalies;

— references to cyber security;

— updated target values for hardware architecture metrics;

— guidance on model based development and software safety analysis;

— evaluation of hardware elements;

— additional guidance on dependent failure analysis;

— guidance on fault tolerance, safety related special characteristics and software tools;

— guidance for semiconductors;

— requirements for motorcycles; and

— general restructuring of all parts for improved clarity.

NOTE The first edition of this document was published in 2012, therefore this document cancels and replaces ISO 26262-10:2012.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

A list of all parts in the ISO 26262 series can be found on the ISO website.

Introduction

The ISO 26262 series of standards is the adaptation of IEC 61508 series of standards to address the sector specific needs of electrical and/or electronic (E/E) systems within road vehicles.

This adaptation applies to all activities during the safety lifecycle of safety-related systems comprised of electrical, electronic and software components.

Safety is one of the key issues in the development of road vehicles. Development and integration of automotive functionalities strengthen the need for functional safety and the need to provide evidence that functional safety objectives are satisfied.

With the trend of increasing technological complexity, software content and mechatronic implementation, there are increasing risks from systematic failures and random hardware failures, these being considered within the scope of functional safety. ISO 26262 series of standards includes guidance to mitigate these risks by providing appropriate requirements and processes.

To achieve functional safety, the ISO 26262 series of standards:

a) provides a reference for the automotive safety lifecycle and supports the tailoring of the activities to be performed during the lifecycle phases, i.e., development, production, operation, service and decommissioning;

b) provides an automotive-specific risk-based approach to determine integrity levels [Automotive Safety Integrity Levels (ASILs)];

c) uses ASILs to specify which of the requirements of ISO 26262 are applicable to avoid unreasonable residual risk;

d) provides requirements for functional safety management, design, implementation, verification, validation and confirmation measures; and

e) provides requirements for relations between customers and suppliers.

The ISO 26262 series of standards is concerned with functional safety of E/E systems that is achieved through safety measures including safety mechanisms. It also provides a framework within which safety-related systems based on other technologies (e.g. mechanical, hydraulic and pneumatic) can be considered.

The achievement of functional safety is influenced by the development process (including such activities as requirements specification, design, implementation, integration, verification, validation and configuration), the production and service processes and the management processes.

Safety is intertwined with common function-oriented and quality-oriented activities and work products. The ISO 26262 series of standards addresses the safety-related aspects of these activities and work products.

Figure 1 shows the overall structure of the ISO 26262 series of standards. The ISO 26262 series of standards is based upon a V-model as a reference process model for the different phases of product development. Within the figure:

— the shaded “V”s represent the interconnection among ISO 26262-3, ISO 26262-4, ISO 26262-5, ISO 26262-6 and ISO 26262-7;

— for motorcycles:

  — ISO 26262-12:2018, Clause 8 supports ISO 26262-3;

  — ISO 26262-12:2018, Clauses 9 and 10 support ISO 26262-4;

— the specific clauses are indicated in the following manner: “m-n”, where “m” represents the number of the particular part and “n” indicates the number of the clause within that part.

EXAMPLE “2-6” represents ISO 26262-2:2018, Clause 6.

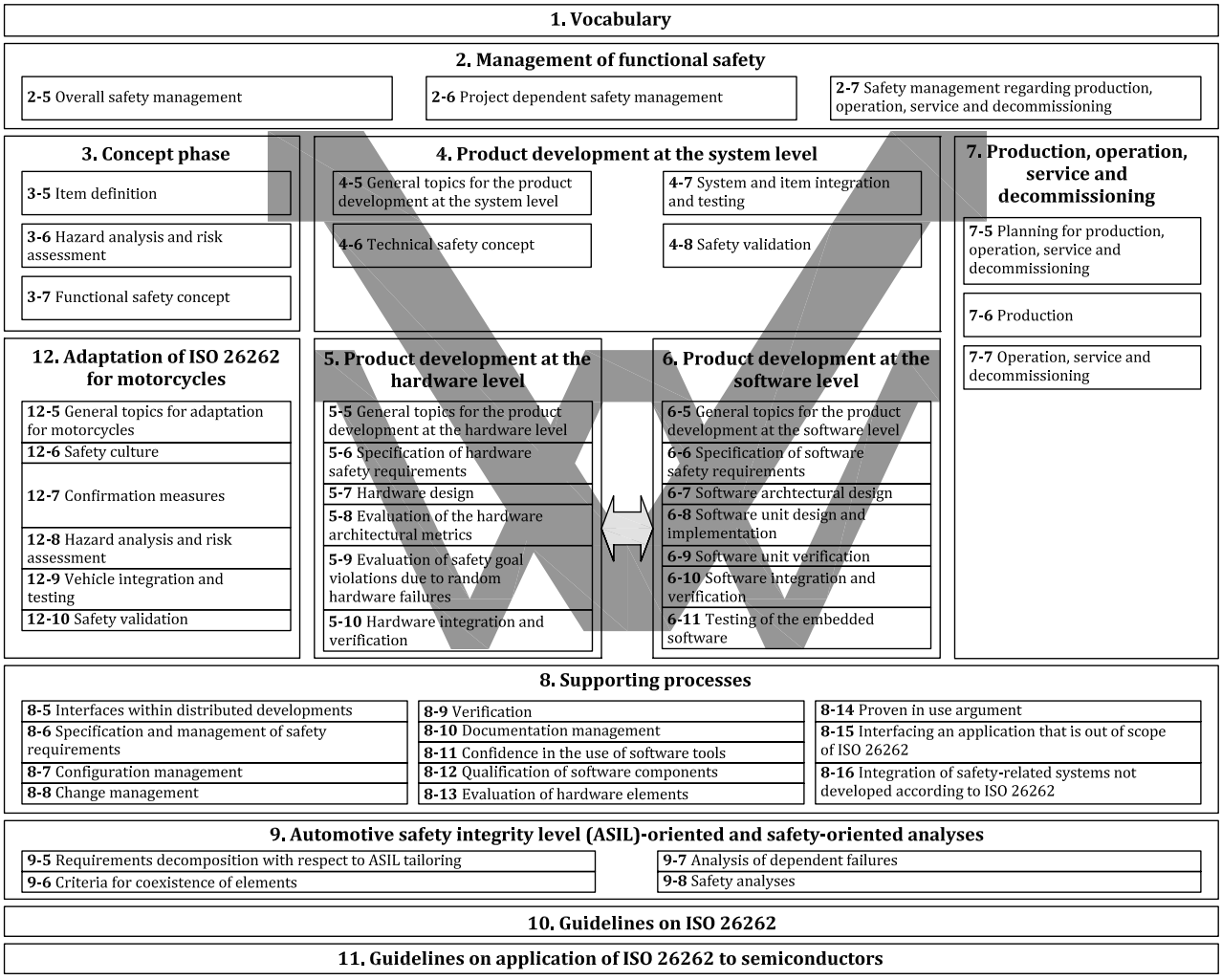


Figure 1 — Overview of the ISO 26262 series of standards

Road vehicles — Functional safety —

Part 10:
Guidelines on ISO 26262

1 Scope

This document is intended to be applied to safety-related systems that include one or more electrical and/or electronic (E/E) systems and that are installed in series production road vehicles, excluding mopeds. This document does not address unique E/E systems in special vehicles such as E/E systems designed for drivers with disabilities.

NOTE Other dedicated application-specific safety standards exist and can complement the ISO 26262 series of standards or vice versa.

Systems and their components released for production, or systems and their components already under development prior to the publication date of this document, are exempted from the scope of this edition. This document addresses alterations to existing systems and their components released for production prior to the publication of this document by tailoring the safety lifecycle depending on the alteration. This document addresses integration of existing systems not developed according to this document and systems developed according to this document by tailoring the safety lifecycle.

This document addresses possible hazards caused by malfunctioning behaviour of safety-related E/E systems, including interaction of these systems. It does not address hazards related to electric shock, fire, smoke, heat, radiation, toxicity, flammability, reactivity, corrosion, release of energy and similar hazards, unless directly caused by malfunctioning behaviour of safety-related E/E systems.

This document describes a framework for functional safety to assist the development of safety-related E/E systems. This framework is intended to be used to integrate functional safety activities into a company-specific development framework. Some requirements have a clear technical focus to implement functional safety into a product; others address the development process and can therefore be seen as process requirements in order to demonstrate the capability of an organization with respect to functional safety.

This document does not address the nominal performance of E/E systems.

This document provides an overview of the ISO 26262 series of standards, as well as giving additional explanations, and is intended to enhance the understanding of the other parts of the ISO 26262 series of standards. It has an informative character only and describes the general concepts of the ISO 26262 series of standards in order to facilitate comprehension. The explanation expands from general concepts to specific contents.

In the case of inconsistencies between this document and another part of the ISO 26262 series of standards, the requirements, recommendations and information specified in the other part of the ISO 26262 series of standards apply.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 26262-1, *Road vehicles — Functional safety — Part 1: Vocabulary*

3 Terms and definitions

For the purposes of this document, the terms, definitions and abbreviated terms given in ISO 26262-1 apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— IEC Electropedia: available at http://www.electropedia.org/

— ISO Online browsing platform: available at https://www.iso.org/obp

4 Key concepts of ISO 26262

4.1 Functional safety for automotive systems (relationship with IEC 61508[1])

IEC 61508, *Functional Safety of electrical/electronic/programmable electronic safety-related systems,* is designated by IEC as a generic standard and a basic safety publication. This means that industry sectors will base their own standards for functional safety on the requirements of IEC 61508.

In the automotive industry, there are a number of issues with applying IEC 61508 directly. Some of these issues and corresponding differences in the ISO 26262 series of standards are described below.

IEC 61508 is based upon the model of "equipment under control", for example an industrial plant that has an associated control system as follows:

a) A hazard analysis identifies the hazards associated with the equipment under control (including the equipment control system), to which risk reduction measures will be applied. This can be achieved through electrical/electronic/programmable electronic (E/E/PE) systems, or other technology safety-related systems (e.g. a safety valve), or external measures (e.g. a physical containment of the plant). The ISO 26262 series of standards contains a normative automotive scheme for hazard classification based on severity, probability of exposure and controllability.

b) Risk reduction allocated to E/E/PE systems is achieved through safety functions, which are designated as such. These safety functions are either part of a separate protection system, or can be incorporated into the plant control. It is not always possible to make this distinction in automotive systems. The safety of a vehicle depends on the behaviour of the control systems themselves.

The ISO 26262 series of standards uses the notion of safety goals and a safety concept as follows:

— a hazard analysis and risk assessment identifies hazards and hazardous events that need to be prevented, mitigated, or controlled;

— at least one safety goal is associated with each hazardous event that has been classified as ASIL A, B, C or D;

— an Automotive Safety Integrity Level (ASIL) is associated with each safety goal;

— the functional safety concept is a statement of the functionality to achieve the safety goal(s);

— the technical safety concept is a statement of how this functionality is implemented on the system level by hardware and software; and

— software safety requirements and hardware safety requirements state the specific safety requirements which will be implemented as part of the software and hardware design.

EXAMPLE The airbag system.

— One of the hazards is unintended deployment.

— An associated safety goal is that the airbag only deploys when a crash occurs that requires the deployment.

— The functional safety concept can specify a redundant function to detect whether the vehicle is in a collision.

— The technical safety concept can specify the implementation of two independent accelerometers with different axial orientations and two independent firing circuits. The squib deploys if both are closed.

IEC 61508 is aimed at singular or low volume systems. The system is built and tested, then installed in the plant, and then safety validation is performed. For mass-market systems such as road vehicles, safety validation is performed before the release for volume (series) production. Therefore, the order of lifecycle activities in the ISO 26262 series of standards is different. Related to this, ISO 26262-7 addresses requirements for production. These are not covered in IEC 61508.

IEC 61508 does not address specific requirements for managing development across multiple organizations and supply chains. Because automotive systems are produced by vehicle manufacturers themselves, by one or more suppliers to the manufacturer or by collaboration between manufacturer and supplier(s), the ISO 26262 series of standards includes requirements to explicitly address this issue, including the Development Interface Agreement (DIA) (see ISO 26262-8:2018, Clause 5).

IEC 61508 does not contain normative requirements for hazard classification. The ISO 26262 series of standards contains an automotive scheme for hazard classification. This scheme recognises that a hazard in an automotive system does not necessarily lead to an accident. The outcome will depend on whether the persons at risk are actually exposed to the hazard in the situation in which it occurs; and whether the involved people are able to take steps to control the outcome of the hazard. An example of this concept, applied to a failure which affects the controllability of a moving vehicle, is given in Figure 2.

NOTE This concept is intended only to demonstrate that there is not necessarily a direct correlation between a failure occurring and the accident. It is not a representation of the hazard analysis and risk assessment process, although the parameters evaluated in this process are related to the probabilities of the state transitions shown in the figure.

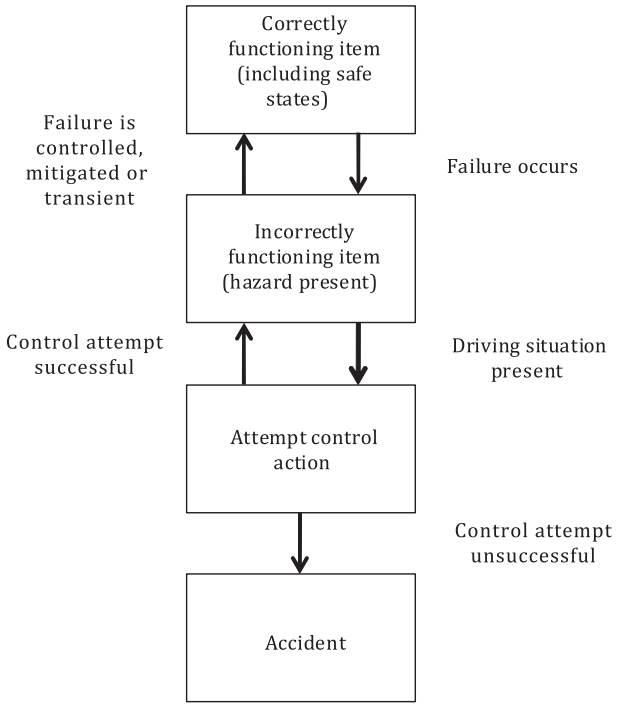


Figure 2 — State machine model of automotive risk

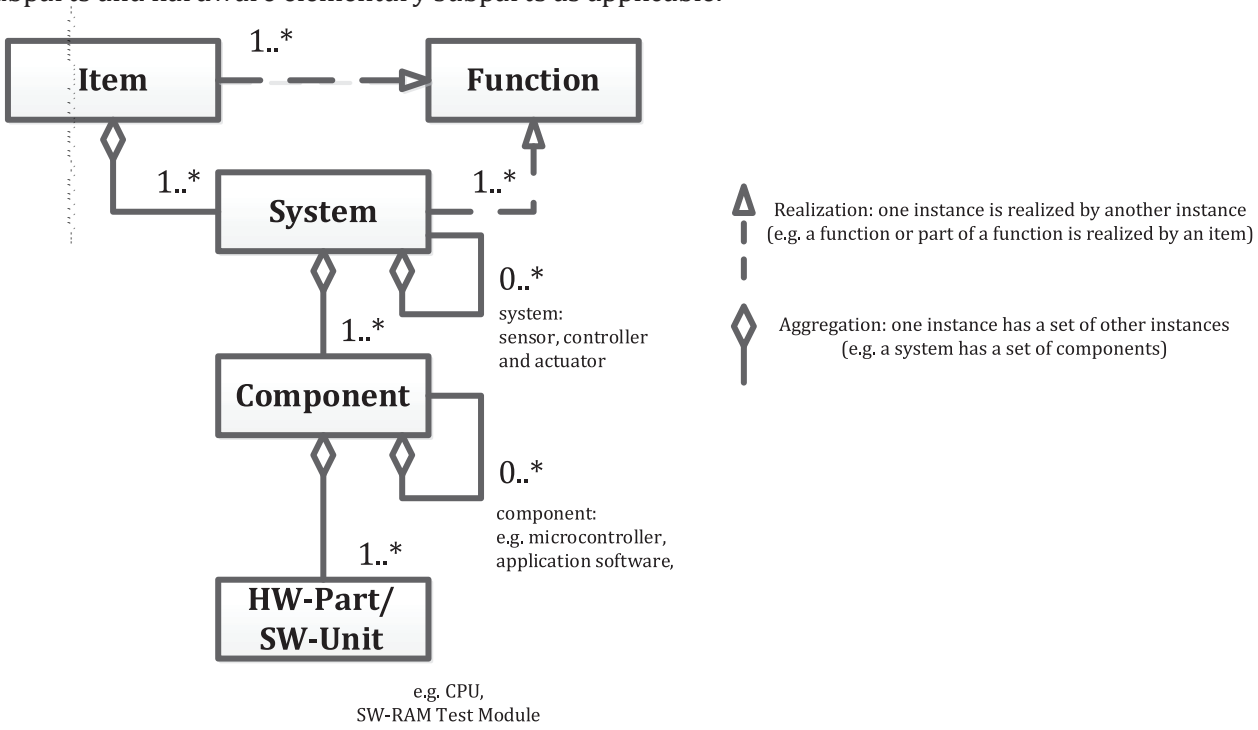
The requirements for hardware development (ISO 26262-5) and software development (ISO 26262-6) are adapted for the state-of-the-art in the automotive industry. For the methods listed in the ISO 26262 series of standards specific goals are provided. To achieve these goals, the provided methods can be applied or a rationale that alternative methods can also achieve the goal is provided.

Safety requirements in the ISO 26262 series of standards are assigned an ASIL (Automotive Safety Integrity Level) rather than a SIL (Safety Integrity Level). The main motivation for this is that the SIL

in IEC 61508 is stated in probabilistic terms (see IEC 61508-1:2010, Table 3). IEC 61508 acknowledges that qualitative judgement is often required in respect of systematic safety integrity while requiring quantitative techniques for hardware safety integrity. An ASIL in ISO 26262 is primarily concerned with requirements for achieving systematic safety in the system, hardware and software; however, there are probabilistic targets associated with compliance to the requirements of an ASIL with respect to random hardware failures.

4.2 Item, system, element, component, hardware part and software unit

The terms item, system, element, component, hardware part and software unit are defined in ISO 26262-1:2018. Figure 3 shows the relationship of item, system, component, hardware part and software unit. Figure 4 shows an example of item dissolution. A divisible element can be labelled as a system or a component. A divisible element that meets the criteria of a system can be labelled as a system. A component is a non-system level, logically and technically separable element. Often the term component is applied to an element that is only comprised of parts and units, but can also be applied to an element comprised of lower-level elements from a specific technology area e.g. electrical / electronic technology (see Figure 4). A hardware part can be further hierarchically composed of hardware subparts and hardware elementary subparts as applicable.



NOTE 1 Depending on the context, the term “element” can apply to the entities “system”, “component”, “hardware part” and “software unit” in this chart, according to ISO 26262-1:2018, 3.41.

NOTE 2 The system, as it is defined in ISO 26262-1:2018, 3.163, relates at least a sensor, a controller, and an actuator with one another. The related sensor or actuator can be included in the system, or can be external to the system.

NOTE 3 *means N are possible.

Figure 3 — Relationship of item, system, component, hardware part and software unit

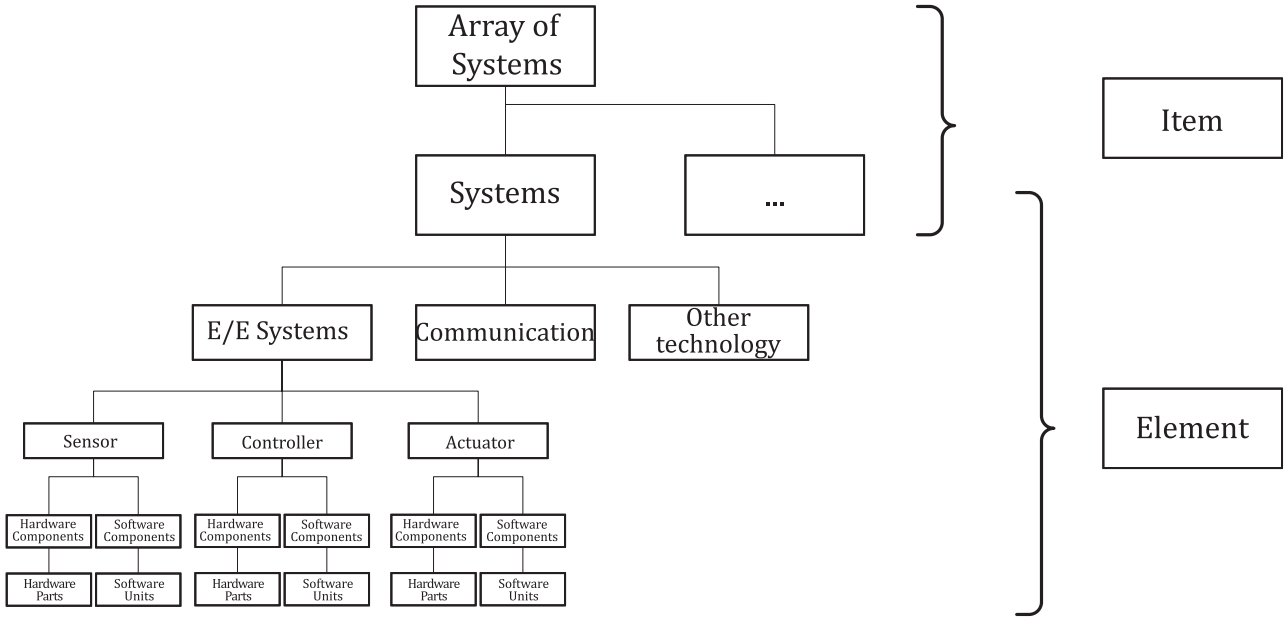


Figure 4 — Example item dissolution

4.3 Relationship between faults, errors and failures

4.3.1 Progression of faults to errors to failures

The terms fault, error and failure are defined in ISO 26262-1:2018. Figure 5 depicts the progression of faults to errors to failures from three different types of causes: systematic software issues, random hardware issues and systematic hardware issues. Systematic faults (see ISO 26262-1:2018, 3.165) are due to design or specification issues; software faults and a subset of hardware faults are systematic. At the component level, each different type of fault can lead to different failures. However, failures at the component level are faults at the item level. Note that in this example, at the vehicle level, faults from different causes can lead to the same failure. A subset of failures at the item level will be hazards (see ISO 26262-1:2018, 3.75) if additional environmental factors permit the failure to contribute to an accident scenario.

EXAMPLE If unexpected behaviour of the vehicle occurs while the vehicle is starting to cross an intersection, a crash can occur, e.g. the risk of the hazardous event “vehicle bucking when starting to cross intersection” is assessed for severity, exposure and controllability ("bucking" refers to making sudden jerky movements).

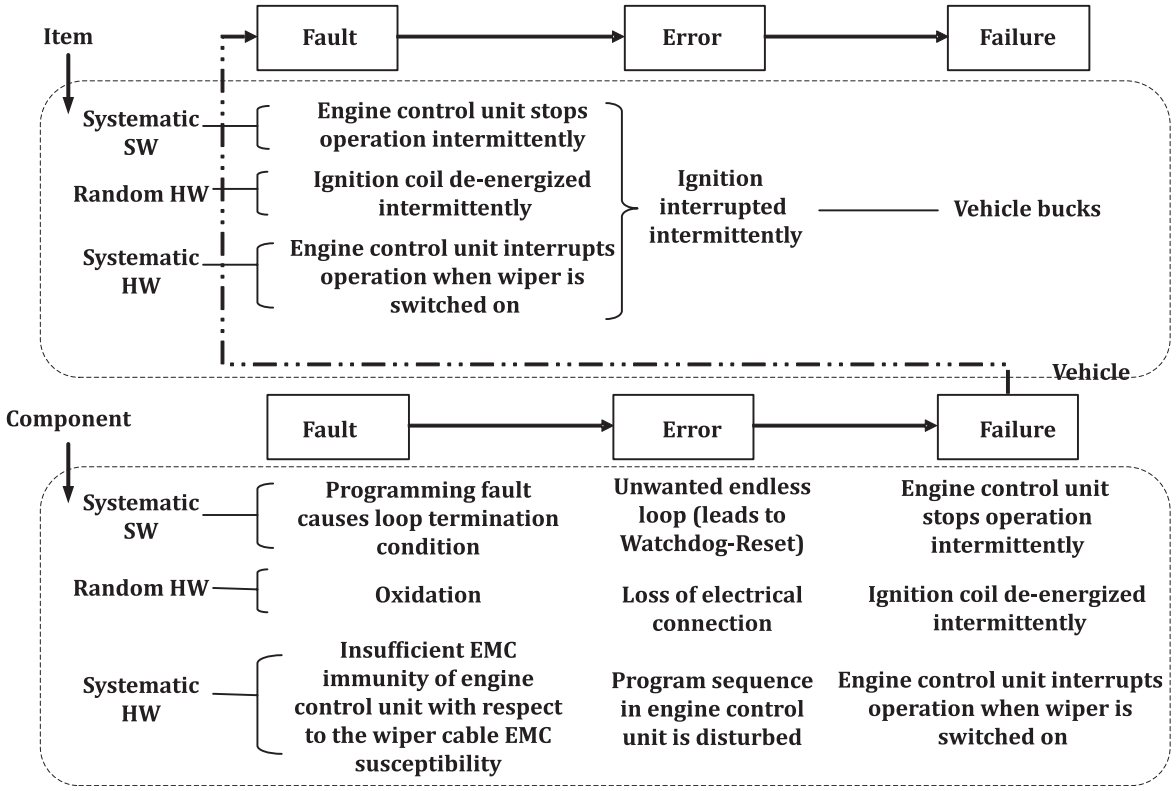


Figure 5 — Example of faults leading to failures

NOTE 1 Possible implemented error detection and control on component or item level is not depicted in Figure 5.

NOTE 2 The failure of the component is the fault on the item level (indicated by the dot dot dashed arrow).

4.4 FTTI and emergency operation tolerant time interval

4.4.1 Introduction

ISO 26262-3:2018, 6.4.4.3 states in the NOTE that the FTTI can be included as part of a safety goal. Furthermore, ISO 26262-4:2018, 6.4.2.2 specifies that the FTTI and emergency operation tolerance time interval are to be taken into account in the definition of fault handling time interval for each safety mechanism.

NOTE Fault handling time interval is a characteristic for a given safety mechanism. Fault tolerant time interval (FTTI) is a characteristic of an item.

As a part of the process of determining safety goals and functional safety requirements at concept phase, the FTTI is specified on vehicle level based on vehicle functionality. This time span can be taken into consideration during product development, leading to the determination of the maximum fault handling time interval (i.e. sum of the fault detection time interval and the fault reaction time interval as described in ISO 26262-1:2018, Figure 5) needed to avoid a hazardous event. FTTI is a necessary value in order to design the response time of a safety mechanism. Within the FTTI, the fault is controlled by a safety mechanism and the occurrence of a hazardous event can be prevented. This is achieved when the sum of the fault detection time interval and the fault reaction time interval is shorter than the FTTI.

An emergency operation (ISO 26262-4:2018, 6.4.2.2) is specified when a safe state cannot be reached within the FTTI. The emergency operation is an operating mode defined as part of the warning and degradation strategy. Emergency operation is initiated prior to the end of the FTTI and is maintained until the safe state is reached prior to the end of the emergency operation tolerance time interval. To

meet the safety goal, a safe state has to be reached before the end of the emergency operation tolerance time interval.

4.4.2 Timing model — Example control system

4.4.2.1 Control system description

This sub-clause applies the concepts of fault detection time interval (FDTI), fault tolerant time interval (FTTI), fault reaction time interval (FRTI), emergency operation tolerance time interval (EOTTI) and diagnostic test time interval (DTTI) to a valve control system example. The system consists of a valve, position sensor, controller and an electrical motor. The function of the system is to control the valve to a desired position using the electric motor.

A hazardous event resulting from an unintended flow can occur if the valve is opened a percentage more than intended. As a fault reaction, the motor is de-energized by a separate circuit in combination with a mechanical spring which pulls the valve to a default fixed opening position. This fixed opening position limits the flow resulting in a safe state for the item.

4.4.2.2 Application of timing model to example control system

The specific failure mode considered in this example is a motor fault which drives the valve to its maximum opening position. This condition can be the result of motor shorted to power or other motor control issues. Four scenarios are considered.

— Scenario 1: System without any safety mechanism preventing the violation of the safety goal.

  A short in the motor occurs resulting in the valve reaching its maximum position. Because no safety mechanism is in place a hazardous event can occur once the FTTI is exceeded.

— Scenario 2: System with implemented safety mechanism without emergency operation and a safe state is achieved within FTTI.

  A short in the motor occurs resulting in the valve reaching its maximum position. The implemented safety mechanism de-energizes the valve motor and the mechanical spring returns the valve to a low flow position within the FTTI, preventing a hazardous event. The safety mechanism (the spring) is designed to operate indefinitely and the safe state can be infinite.

— Scenario 3: System with implemented safety mechanism which prevents a hazardous event within the FTTI, but emergency operation is needed to transit to a safe state. The safe state is achieved within the emergency operation tolerance time interval by restricting the vehicle operating state.

  A short in the motor occurs resulting in the valve reaching its maximum position. The implemented safety mechanism de-energizes the valve motor and the mechanical spring returns the valve to a low flow position within the FTTI. The safety mechanism (the spring) is only designed to operate for a limited amount of time, the EOTTI. Prior to the expiration of the EOTTI, the vehicle operating state is restricted such that the flow from the valve cannot cause a hazardous event.

— Scenario 4: System with implemented safety mechanism which prevents a hazardous event is within the FTTI but emergency operation is needed to transit to a safe state. However, the transition time takes longer than the EOTTI. As a consequence, the cumulated risk becomes unacceptable, exceeding the target specified in the functional safety concept.

  A short in the motor occurs resulting in the valve reaching its maximum position. The implemented safety mechanism de-energizes the valve motor and the mechanical spring returns the valve to a low flow position within the FTTI. The safety mechanism (the spring) is only designed to operate for a limited amount of time, the EOTTI. In this scenario, the vehicle operation is not restricted and the item is in emergency operation longer than the expiration of the EOTTI, resulting in an unreasonable risk of safety goal violation.

Figure 6 shows the timing model associated with the four scenarios. Figure 6 is based on ISO 26262-1:2018, Figure 4 and ISO 26262-1:2018, Figure 5.

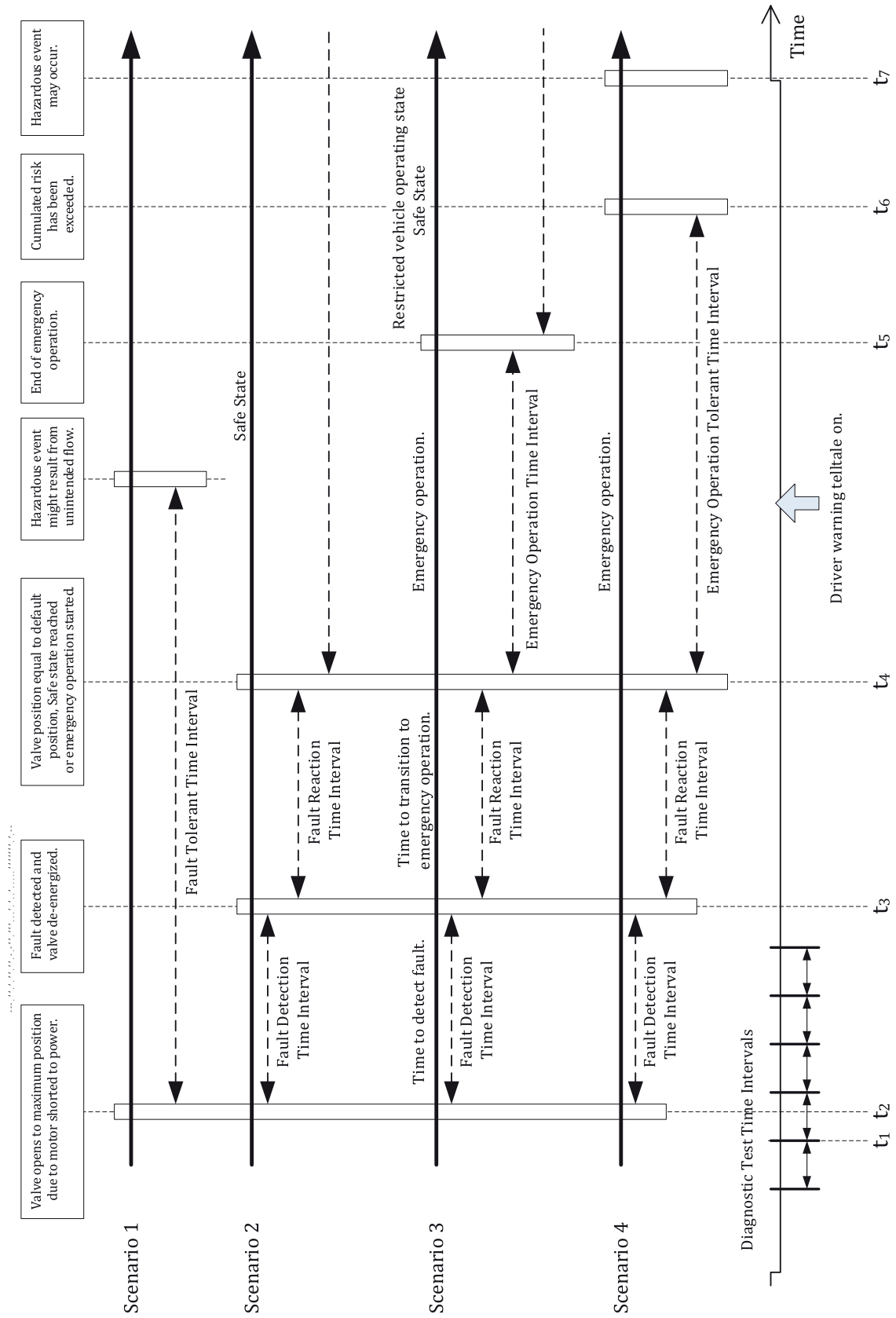


Figure 6 — Timing model example based on ISO 26262-1:2018, Figure 4

Figure 6 includes 7 time stamps described below:

t_1 Time of diagnostic test prior to occurrence of the fault.

t_2 Occurrence of fault, fault not detected.

t_3 Detection of fault (e.g. due to error counter reaching its threshold, see ISO 26262-1:2018, 3.55 FDTI EXAMPLE), start of fault reaction time interval.

t_4 Transition to safe state completed (scenario 2), start of emergency operation (scenarios 3 and 4).

t_5 End of emergency operation (scenario 3).

t_6 Time limit for emergency operation.

t_7 Occurrence of hazardous event.

4.4.2.3 Warning and degradation strategy

When the valve is in the default position, it can have an impact at the vehicle level. The functional safety concept may also include a requirement to warn the driver when in this state. This is part of the warning and degradation strategy and is indicated by the “Driver warning telltale on” arrow on Figure 6 which can occur within a specified time after t_3.

5 Selected topics regarding safety management

5.1 Work product

This sub-clause describes the term "work product".

A work product is the result of meeting the corresponding requirements of the ISO 26262 series of standards (see ISO 26262-1:2018, 3.185). Therefore, a work product can provide evidence of compliance with these safety requirements.

EXAMPLE A requirements specification is a work product that can be documented by means of a requirements database or a text file. An executable model is a work product that can be represented by modelling language files that can be executed (e.g. for simulation purposes by using a software tool).

The documentation of a work product (see ISO 26262-8:2018, Clause 10) serves as a record of the executed safety activities, safety requirements or of related information. Such documentation is not restricted to any form or medium.

EXAMPLE The documentation of a work product can be represented by electronic or paper files, by a single document or a set of documents. It can be combined with the documentation of other work products or with documentation not directly dedicated to functional safety.

To avoid the duplication of information, cross-references within or between documentation can be used.

5.2 Confirmation measures

5.2.1 General

In ISO 26262, specified work products are evaluated during subsequent activities, either as part of the confirmation measures or as part of the verification activities. This sub-clause describes the difference between verification and confirmation measures.

The verification activities are the primary measures in ISO 26262 to provide evidence that a work product is suitable and complies with the corresponding requirements. The verification of work products can include:

— verification of the specification, or implementation, of derived safety requirements against the safety requirements at a higher level, regarding completeness and correctness; or

— the execution of test cases and the examination of test results to provide evidence of the fulfilment of specified safety requirements, by exercising the item or its element(s).

The verification activities are specified in ISO 26262-3, ISO 26262-4, ISO 26262-5 and ISO 26262-6. Furthermore, generic requirements regarding the verification activities in the ISO 26262 series of standards are specified in ISO 26262-8:2018, Clause 9 and further details specific to the verification of safety requirements are specified in ISO 26262-8:2018, Clause 6.

Verification of work products can be performed by using techniques such as:

— reviews;

— simulation;

— analysis; or

— testing.

Confirmation measures are specified in ISO 26262-2:2018, 6.4.9 and are performed to evaluate the item's achievement of functional safety.

EXAMPLE If an ASIL decomposition is applied during the system architectural design phase:

— the verification of the resulting system architectural design is performed against the technical safety concept (see ISO 26262-4:2018, 6.4.9); and

— the confirmation of the correct application of the ASIL decomposition can be performed as part of a functional safety assessment, with regard to ISO 26262-9:2018, Clause 5, including the confirmation that a dependent failure analysis has been performed and justifies the claim of sufficient independence between the elements that implement the corresponding redundant safety requirements.

5.2.2 Functional safety assessment

If the highest ASIL of the item's safety goals is ASIL C or D, a functional safety assessment is performed to evaluate an item's achievement of functional safety. In ISO 26262-2, certain aspects of a functional safety assessment are described as well as further aspects of confirmation measures.

The scope of the functional safety assessment is defined in ISO 26262-2:2018, Clause 6.

In the case a functional safety assessment is performed, the results of the functional safety audit and of the confirmation reviews are an input for the functional safety assessment. The person responsible for the assessment can perform the assessment according to his/her discretion, including how to make use of the results of the functional safety audit and confirmation reviews.

EXAMPLE 1 If the results of the functional safety audit are satisfactory, the person responsible for the functional safety assessment can decide to rely on the results of the audit, without making a further judgement of the implementation of the processes required for functional safety.

EXAMPLE 2 Based on the confirmation review report of a particular work product, the person responsible for the assessment can decide to perform, or to request, a more in-depth review of certain aspects of that work product, or can check whether the confirmation review sufficiently considered the interplay between that work product and related work products.

NOTE 1 It is possible that the person responsible for the functional safety assessment performs a particular confirmation review i.e. a confirmation review is not necessarily performed by a person different from the person responsible for the assessment.

A functional safety assessment can be repeated or updated.

EXAMPLE 3 A functional safety assessment update because of a change of the item, or element(s) of the item, that is identified by the change management as having an impact on the functional safety of the item (see ISO 26262-8:2018, Clause 8).

EXAMPLE 4 A functional safety re-assessment that is triggered by a functional safety assessment report that recommends a conditional acceptance or a rejection of the item's functional safety. In this case, the iteration includes a follow-up of the recommendations resulting from the previous functional safety assessment(s), including an evaluation of the performed corrective actions, if applicable.

If the highest ASIL of the item's safety goals is ASIL A or ASIL B, a functional safety assessment can be omitted or performed less rigorously. However, even if the functional safety assessment is not performed, other confirmation measures are still performed (see ISO 26262-2:2018, Table 1).

In the case of a distributed development, the scope of a functional safety assessment includes the work products generated, and the processes and safety measures implemented, by a vehicle manufacturer and the suppliers in the item's supply chain (see ISO 26262-2 and ISO 26262-8:2018, Clause 5).

The purpose of a functional safety assessment is to evaluate an item's achievement of functional safety, which is only possible at the item level. Therefore, a functional safety assessment of a supplier (that develops elements of the item) refers only to an assessment with a limited scope, which essentially serves as an input for the subsequent functional safety assessment activities (at the customer level). As the final customer in the item development, the vehicle manufacturer appoints person(s) to perform an overall functional safety assessment to judge an item's achievement of functional safety. This judgement includes providing a recommendation for acceptance, conditional acceptance, or rejection of the item's functional safety.

NOTE 2 For the case where a Tier 1 supplier is responsible for the item development including vehicle integration, this supplier takes over the aforementioned role of the vehicle manufacturer.

In a practical manner, a functional safety assessment in the case of a distributed development can thus be broken down into:

— functional safety assessments with a limited scope, concerning the suppliers in the supply chain. The applicable ASIL is the highest inherited ASIL (of the item's safety goals) across the elements, of the item, that are developed by the supplier (see also ISO 26262-8:2018, 5.4.5); and

— a final functional safety assessment that includes a judgement of the functional safety achieved by the integrated item e.g. performed by the vehicle manufacturer. The applicable ASIL is the highest ASIL of the safety requirements (see also ISO 26262-2).

EXAMPLE 5 A vehicle manufacturer develops an item with an ASIL D Safety Goal (SG1) and an ASIL B Safety Goal (SG2), and will perform a functional safety assessment regarding this item. It is possible that, for example, a Tier 2 or Tier 3 supplier only develops ASIL B elements of the item, i.e. only elements that inherit the ASIL of SG2 (however, refer to ISO 26262-9:2018, Clause 6, if criteria for coexistence of elements are applicable). There is a recommendation in ISO 26262-2:2018, Table 1 to perform a functional safety assessment with independence I0 regarding this element development.

The scope, procedure (e.g. work products to be made available by the supplier, work products to be reviewed by the customer) and execution of a functional safety assessment concerning the interface between a customer and a supplier are specified in the corresponding Development Interface Agreement (see ISO 26262-8:2018, Clause 5).

EXAMPLE 6 DIA between a vehicle manufacturer (customer) and a Tier 1 supplier. DIA between a Tier 1 supplier (customer) and a Tier 2 supplier.

A possible manner to perform a functional safety assessment in the case of a distributed development is that the vehicle manufacturer and the suppliers in the supply chain each address those aspects of the assessment activities for which the respective party is responsible for, as follows:

— a supplier reviews the safety measures implemented in the developed elements including their appropriateness and effectiveness to comply with the corresponding safety goals or safety

requirements (provided by the customer or developed by the supplier), and evaluates its implemented processes and the applicable work products. A supplier also evaluates the potential impacts of the developed elements on the item's functional safety, e.g. identifies whether implemented safety measures can lead to new hazards; and

— the vehicle manufacturer evaluates the functional safety of the integrated item. A part of the evaluation can be based on the work products or information provided by one or more suppliers, including reports of the functional safety assessments.

NOTE 3 A customer can evaluate the safety measures implemented by a supplier and the work products made available by a supplier. A customer can also evaluate the processes implemented by a supplier at the supplier's premises (see ISO 26262-8:2018, 5.4.3.1)

5.3 Understanding of safety cases

5.3.1 Interpretation of safety cases

The purpose of a safety case is to provide a clear, comprehensive and defensible argument, supported by evidence, that an item is free from unreasonable risk when operated in an intended context.

The guidance given here focuses on the scope of ISO 26262.

There are three principal elements of a safety case, namely:

— the safety goals and related safety requirements (safety objectives of the item or element);

— the safety argument; and

— the ISO 26262 series of standards work products (i.e. the evidence).

The relationship between these three elements, in the context of the ISO 26262 series of standards, is depicted in Figure 7.

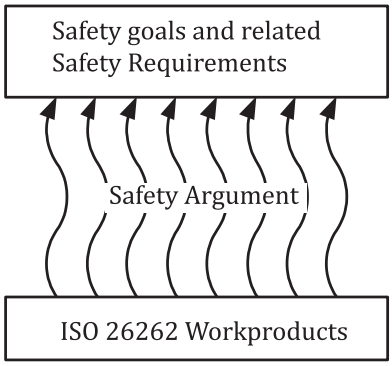


Figure 7 — Key elements of a safety case (see Reference [2])

The safety argument communicates the relationship between the evidence and the objectives. It is possible to present many pages of supporting evidence without clearly explaining how this evidence relates to the safety objectives. Both the argument and the evidence are crucial elements of the safety case. An argument without supporting evidence is unfounded, and therefore unconvincing. Evidence without an argument is unexplained, resulting in a lack of clarity as to how the safety objectives have been satisfied. Safety cases are communicated through the development and presentation of safety case reports. The role of a safety case report is to summarise the safety argument and then reference the reports capturing the supporting safety evidence (e.g. test reports).

Safety arguments used to date in other industries have often been communicated in safety case reports through narrative text. Narrative text can describe how a safety objective has been interpreted, allocated and decomposed, ultimately leading to references to evidence that demonstrate fulfilment

of lower-level safety claims. Alternatively, or as a support, graphical argument notations (such as Claims–Argument–Evidence and the Goal Structuring Notation [2]) could be used to visually and explicitly represent the individual elements of a safety argument (requirements, claims, evidence and context) and the relationships that exist between these elements (i.e. how individual requirements are supported by specific claims, how claims are supported by evidence and the assumed context that is defined for the argument).

A safety argument that argues safety through direct appeal to features of the implemented item (e.g. the behaviour of a timing watchdog) is often termed a product argument. A safety argument that argues safety through appeal to features of the development and assessment process (e.g. the design notation adopted) is often termed a process argument.

Both types of argument can be used to achieve a sound argument for the safety of the item where a process argument can be seen as providing the confidence in the evidences used in the product argument.

5.3.2 Safety case development lifecycle

The development of a safety case can be treated as an incremental activity that is integrated with the rest of the development phases of the safety lifecycle.

NOTE The safety plan can include the planning for incremental steps and the preliminary versions of the safety case.

Such an approach allows intermediate versions of the safety case at given milestones of the product development. For example, a preliminary version of the safety case can be created after the verification of the technical safety requirements; an interim version of the safety case can be created after the verification of the system design; and the final version can be created just prior the final report of the functional safety assessment.

The safety case is subject to a confirmation review as given in ISO 26262-2:2018, 6.4.9.

If the item is modified, the impact on the safety case is evaluated and if necessary the safety case is updated considering the modifications.

6 Concept phase and system development

6.1 General

This clause provides an overview of the principles behind the hazard analysis and risk assessment using simplified examples to the concepts.

6.2 Example of hazard analysis and risk assessment

6.2.1 General

Consider the example of an item controlling an energy storage device embedded in the vehicle. For the purpose of this example, the stored energy is intended to be released only if the vehicle is running greater than or equal to 15 km/h. The release of the stored energy at less than 15 km/h can lead to the overheating and consequent explosion of the device.

6.2.2 HARA example 1

a) Hazard identification

The hazard, “unwanted release of energy of the device that can result in an explosion”, is identified.

b) Hazardous event

The driving situation in which the identified hazard can lead to a hazardous event is considered as driving less than 15 km/h. If an unwanted release of energy due to a failure in the item occurs during this driving condition, the energy storage device could explode, causing severe harm to the occupants of the vehicle.

c) Classification of the identified hazardous event

The explosion leads to life-threatening injuries for the passengers of the vehicle, with survival uncertain: the severity could be estimated as S3.

The vehicle is travelling at a speed of less than 15 km/h. Based on traffic statistic for the target market of the vehicle, this condition occurs between 1 % and 10 % of the driving time: the exposure of this situation could be estimated as E3.

The ability of the driver or the passengers of the vehicles to control the item failure and the explosion of the device is considered as implausible: this controllability could be estimated as C3 (difficult to control or uncontrollable).

The application of ISO 26262-3:2018, Table 4: ASIL determination leads to an ASIL C.

6.2.3 HARA example 2

This clause considers the case where the impact of unwanted release of energy is inherently restricted by design improvement. This would result in an evaluation of the HARA as follows:

a) Hazard identification

As a hazard, “unwanted release of energy of the device that can result in an explosion”, is identified.

b) Hazardous event

For all driving situations an unwanted release of energy does not lead to a hazardous event. Therefore, the item failure cannot result in harm.

c) Classification of the identified hazardous event

Since the item failure does not lead to harm, the severity is classified as S0 and controllability does not need to be determined. Therefore, a safety goal does not need to be defined.

6.3 An observation regarding controllability classification

As explained in ISO 26262-3:2018, Clause 6, the controllability represents an estimation of the probability that the driver or other traffic participant is able to avoid the specific harm.

In the simplest case, only one outcome is considered for a given hazardous event and the controllability represents an estimation of the probability that this outcome is avoided. However, there can be other cases. For example, a severe outcome (e.g. severity class S2) can be possible but relatively easy to avoid (e.g. controllability C1) while a less severe outcome (e.g. S1) is more difficult to avoid (e.g. C3). Assuming that the exposure class is E4, the following set of values can be the result, which illustrates that it is not necessarily the highest severity that leads to the highest ASIL:

— E4, S2, C1 → ASIL A; and

— E4, S1, C3 → ASIL B.

In this example, ASIL B is an appropriate classification of the hazardous event.

6.4 External measures

6.4.1 General

An external measure is a measure separate and distinct from the item that reduces or mitigates the risks resulting from a failure of the item.

NOTE 1 External measures can be considered in the HARA, if they are independent with regards to the function to be implemented by the item.

NOTE 2 External measures as a technical assumption to reduce an ASIL are validated according to ISO 26262-3:2018, 6.4.4.4.

6.4.2 Example of vehicle dependent external measures 1

Vehicle A is equipped with a manually operated transmission gear box which can be left in any gear, including neutral, upon key off. Vehicle B is equipped with an automatic gear box which, at key off, maintains one gear engaged and a normally closed clutch. Both vehicles have an additional item, Electrical Parking Brake (EPB).

A scenario is analysed for both vehicles which includes:

— The vehicle is parked (key off, driver not present);

— Vehicle is kerbside on an incline, located in a populated urban area; and

— A failure involving a sudden release of EPB occurs.

In this scenario, Vehicle A, when unintentionally left in neutral at key off, will potentially roll away if left unattended. This can result in an assessed controllability rating of C3, a severity rating of S2 or higher depending on the presence of nearby vulnerable persons, and an exposure ranking greater than E0. Depending on the exposure rating assigned, the ratings proposed result in an assigned ASIL between ASIL A and ASIL C or QM.

Vehicle B however always engages a gear, so it does not move. Thus there is no resulting hazard. The vehicle-dependent external measures included in this design contribute to the elimination of risk for this scenario, but only if the automatic gear box and the EPB can be shown to be sufficiently independent.

6.4.3 Example of vehicle dependent external measures 2

Vehicle A is equipped with dynamic stability control in addition to a stop-start feature. Vehicle B is only equipped with the stop-start feature.

A scenario is analysed for both vehicles which includes:

— The vehicle is being driven at medium-high speed (50 km/h < v < 90 km/h);

— The road surface is paved and dry, and in a suburban area;

— The vehicle is approaching a medium curvature bend in the road;

— The vehicle speed and road curvature contribute to a medium-high lateral acceleration; and

— A failure in the stop-start feature triggering an undesired engine shutdown results in a sudden loss of traction power during the scenario.

As a result of the sudden loss of traction power, a yaw moment is induced on the vehicle, requiring the driver to adjust steering input to re-establish the control of the vehicle. Performing this manoeuvre in Vehicle B can be shown to have a lower controllability, which can contribute to high risk. The risk classification will be dependent on the exposure rating assigned. By contrast, the dynamic stability control feature in Vehicle A limits the effects of the lateral instability. As a result, the controllability rating will be better for Vehicle A. Therefore, the vehicle-dependent external measures provided by the

dynamic stability control contribute to the reduction of risk for this scenario. However, this is the case only if it can be shown that the failure in the start-stop function being considered cannot propagate to the dynamic stability control function.

NOTE An in-depth analysis of the hazard used in the example can be found in Reference [6].

6.5 Example of combining safety goals

6.5.1 Introduction

Safety goals are top-level safety requirements for the item. They lead to the functional safety requirements needed to avoid an unreasonable risk for a hazardous event. They are determined in the concept phase in accordance with ISO 26262-3:2018, 6.4.4. When safety goals are similar or refer to the same hazard in different situations, they can be combined into a single safety goal with the highest ASIL of the original safety goals. This can simplify the further development, as fewer safety goals will be managed, while still covering all the identified hazards.

6.5.2 General

In the following example, the item, the safety goals and the ASIL classifications shown are only intended to illustrate the safety goal combination process. This example does not reflect the application of the ISO 26262 series of standards on a similar real-life project. In particular, it is not complete in terms of failure modes identification, situation analysis and the assessment of vehicle level effects.

For simplicity, the example is limited to the composition of two safety goals, but the same approach can be extended to a higher number of initial safety goals.

6.5.3 Function definition

Consider a vehicle equipped with an Electrical Parking Brake (EPB) system. The EPB system, when activated by a specific driver's request, applies brake torque to the vehicle's rear wheels to prevent unintended vehicle movement while parked (parking function).

6.5.4 Safety goals applied to the same hazard in different situations

6.5.4.1 Hazard analysis and risk assessment

To simplify the example, consider just the following failure mode of the parking function:

— unintended parking brake activation.

NOTE In this context, the term "unintended activation" refers to function actuation without the driver's request.

This failure mode can lead to different vehicle effects depending on the specific situation present when the fault occurs, as shown in Table 1.

Table 1 — Safety goals resulting from the same hazard in different situations

Failure mode	Hazard	Specific situation	Hazardous event	Possible consequences	ASIL	Safety goal	Safe state
Unintended parking brake activation	Unexpected deceleration	High speed **OR** taking a bend **OR** low friction surface	Unexpected deceleration at high speed **OR** taking a bend **OR** low friction surface	Loss of vehicle stability	*Higher ASIL*	Avoid activating the parking function without the driver's request when the vehicle is moving	EPB disabled
Unintended parking brake activation	Unexpected deceleration	Medium-low speed **AND** high friction surface	Unexpected deceleration at medium-low speed **AND** high friction surface	Rear end collision with the following vehicle	*Lower ASIL*	Avoid activating the parking function without the driver's request when the vehicle is moving	EPB disabled

6.5.4.2 Safety goals elaboration

As shown above, the same safety goals and safe states are applicable to both situations. Therefore, the following safety goal can be defined:

— Safety goal: Avoid unintended activation of the parking function when the vehicle is moving;

— Safe state: EPB disabled; and

— ASIL: Higher ASIL determined in Table 1 is assigned to this safety goal.

7 Safety process requirement structure — Flow and sequence of the safety requirements

The flow and sequence of the safety requirement development in accordance with the ISO 26262 series of standards is illustrated in Figure 8 and Figure 9, and outlined below. The specific clauses are indicated in the following manner: "m-n", where "m" represents the number of the part and "n" indicates the number of the clause or sub-clause within that part.

A hazard analysis and risk assessment is performed to identify the risks and to define the safety goals for these risks (see ISO 26262-3:2018, Clause 6).

A functional safety concept is derived which specifies functional safety requirements to satisfy the safety goals. These requirements define the safety mechanisms and the other safety measures that will be used for the item. In addition, the system architectural elements that support these requirements are identified (see ISO 26262-3:2018, Clause 7).

A technical safety concept is derived which specifies the technical safety requirements and their allocation to system elements for implementation by the system design. These technical safety requirements will indicate the partitioning of the elements between the hardware and the software (see ISO 26262-4:2018, Clause 6).

The system design will be developed in accordance with the technical safety requirements. Their implementation can be specified in the system design specification (see ISO 26262-4:2018, Clause 6).

Finally, the hardware and software safety requirements will be provided to comply with the technical safety requirements and the system design (see ISO 26262-5:2018, Clause 6 and ISO 26262-6:2018, Clause 6).

Figure 8 illustrates the relationship between the hardware requirements and the design phases of the ISO 26262 series of standards.

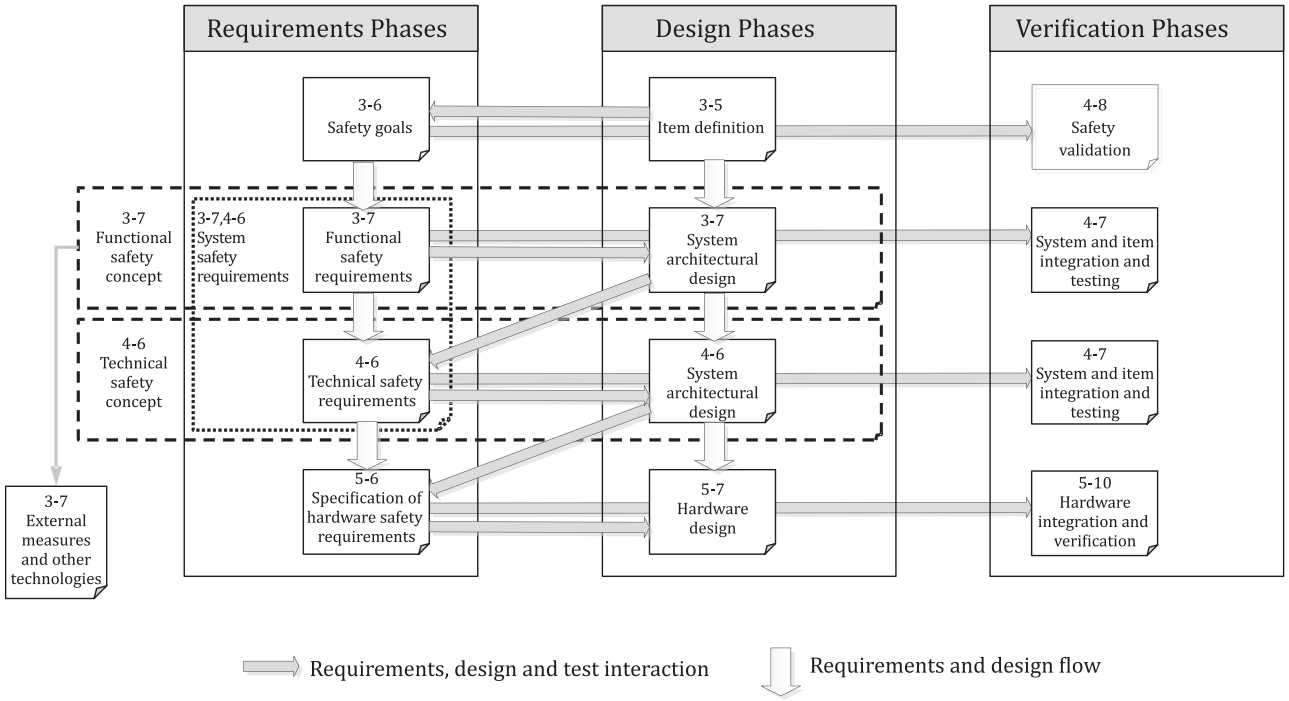


Figure 8 — Safety requirements, design and test flow from concept to hardware

Figure 9 illustrates the relationship between the software requirements, the design, and the test sub-phases of the ISO 26262 series of standards.

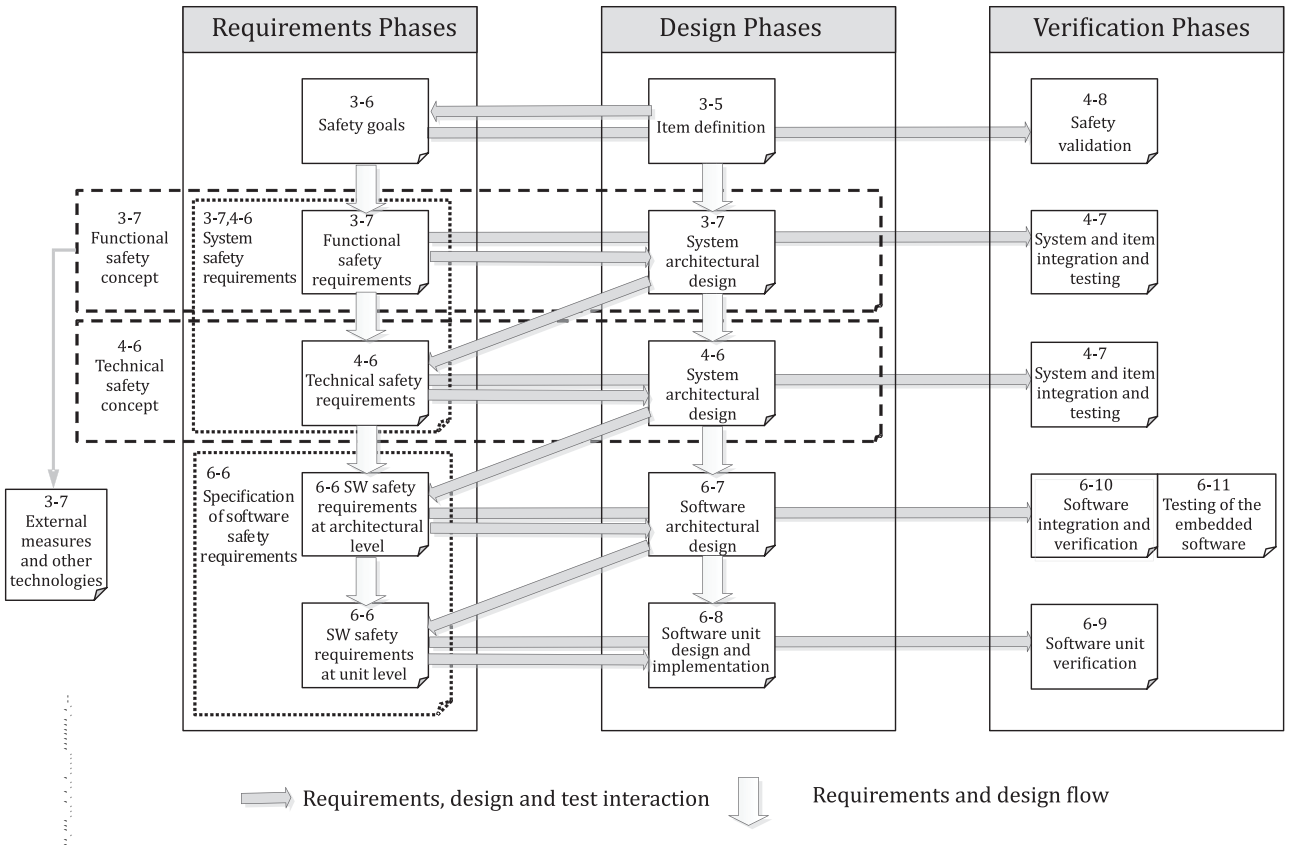


Figure 9 — Safety requirements, design and test flow from concept to software

— System design:

The system design is continuously refined from the item definition (3–6) to the system architectural design (4–6).

— Dependence among test levels:

The test specifications and test cases on each level mainly depend on the corresponding requirements and design. They typically do not depend on the test specifications, test cases and tests results of other test levels. The test specifications typically depend on the test environment.

— Dependence of test levels on requirements and design levels:

Test specifications and test cases are derived from the requirements on the same level, supported by information on the design at the same level.

EXAMPLE For performance testing, information on the design is necessary.

— Software safety requirements verification:

The phase of software safety requirements verification (6–11) requires the integration of software and hardware.

— External measures and other technologies:

External measures and other technologies are validated at the vehicle level.

8 Concerning hardware development

8.1 The classification of random hardware faults

8.1.1 General

In general, the combinations of faults that are considered are limited to combinations of two independent hardware faults, unless analysis based on the functional or technical safety concept has shown that n point faults with n >2 are relevant. Therefore, for a given safety goal and a given HW element a fault can be classified in most cases as either:

a) single-point fault;

b) residual fault;

c) detected dual-point fault;

d) perceived dual-point fault;

e) latent dual-point fault; or

f) safe fault.

Explanations of the various fault classes, as well as examples, are given below.

8.1.2 Single-point fault

This fault:

— can lead directly to the violation of a safety goal; and

— is a fault of a hardware element that does not have at least one safety mechanism.

EXAMPLE An unsupervised resistor for which at least one failure mode (e.g. open circuit) has the potential to violate the safety goal.

NOTE If a hardware element is covered by at least one safety mechanism (e.g. a watchdog for a runaway of a microcontroller), then none of its faults are classified as single-point faults. The faults for which the safety mechanisms do not prevent the violation of the safety goal are classified as residual faults.

8.1.3 Residual fault

This fault or portion of this fault:

— can lead directly to the violation of the safety goal; and

— is a fault of a hardware element that is covered by at least one safety mechanism and these safety mechanisms do not mitigate or control this fault or portion of this fault.

EXAMPLE If a Random Access Memory (RAM) module is only checked by a checkerboard RAM test safety mechanism, certain kinds of bridging faults are not detected. The violation of the safety goals due to these faults are not prevented by the safety mechanism. These faults are examples of residual faults.

NOTE The safety mechanism has less than 100 % diagnostic coverage in this case.

8.1.4 Detected dual-point fault

This fault:

— contributes to the violation of the safety goal, but can only lead to a safety goal violation in combination with one other independent hardware fault; and

— is detected by a safety mechanism which prevents it from being latent.

EXAMPLE 1 Flash memory that is protected by parity: a single bit fault which is detected and triggers a reaction according to the technical safety concept, like switching off the system and informing the driver via a warning lamp.

EXAMPLE 2 Flash memory that is protected by Error Correction Code (ECC): faults in the ECC logic that are detected by a test and a reaction is triggered according to the technical safety concept, like informing the driver via a warning lamp.

In the case where a safety mechanism mitigates a transient fault by restoring the item to a fault free state, such a fault can be considered as a detected dual-point fault even if the driver is never informed about its existence.

EXAMPLE A transient bit flip which is corrected by an ECC before the data is provided to the CPU and is corrected later on by writing back the correct value. Logging can be used to distinguish between intermittent faults and true transient faults.

NOTE Dual-point faults can be classified as primary dual-point faults and as secondary dual-point faults. Primary dual-point faults cannot lead to a safety goal violation by themselves even if there are no safety mechanism present to control their fault. Secondary dual-point faults do have the potential to violate a safety goal, but a safety mechanism is present that mitigates the safety goal violation.

8.1.5 Perceived dual-point fault

This fault:

— contributes to the violation of the safety goal, but will only lead to a safety goal violation in combination with one other independent hardware fault; and

— is perceived by the driver with or without detection by a safety mechanism within a prescribed time.

EXAMPLE A dual-point fault can be perceived by the driver if the functionality is significantly and unambiguously affected by the consequence of the fault.

NOTE If a dual-point fault is perceived by the driver, as well as detected by a safety mechanism, it can be classified as either a detected or a perceived dual-point fault. It cannot be classified as both since the latent-fault metric would be incorrectly calculated due to the fact that one fault would then contribute to the detected dual-point faults as well as to the perceived dual-point faults, counting this fault twice.

8.1.6 Latent dual-point fault

This fault:

— contributes to the violation of the safety goal, but will only lead to the violation of the safety goal in combination with one other independent fault; and

— is neither detected by a safety mechanism nor perceived by the driver. Until the occurrence of the second independent fault, the system is still operable and the driver is not informed about the fault.

EXAMPLE 1 In the case of a flash memory that is protected by ECC: a permanent single bit fault for which the value is corrected by the ECC when read but that is neither corrected in the flash memory nor signalled. In this case, the fault cannot lead to a safety goal violation (since the faulty bit is corrected), but it is neither detected (since the single bit fault is not signalled) nor perceived (since there is no impact on the functionality of the application). If an additional fault occurs in the ECC logic, it can lead to a loss of control of this single bit fault, leading to a potential violation of the safety goal.

EXAMPLE 2 In the case of a flash memory which is protected by ECC: a permanent single fault in the ECC logic leading to an unavailability of the ECC not detected and controlled within the maximum fault handling time interval.

8.1.7 Safe fault

Safe faults can be faults of one of the two categories:

a) all n point faults with n > 2, unless the safety concept shows that they are relevant contributors to a safety goal violation; or

b) faults that will not contribute to the violation of a safety goal.

EXAMPLE 1 In the case of a flash memory that is protected by ECC and a Cyclic Redundancy Check (CRC): a single bit fault which is corrected by ECC but is not signalled. The fault is prevented from violating the safety goal but is not signalled by the ECC. If the ECC logic fails, the fault is detected by the CRC and the system switches off. A violation of a safety goal can occur (n = 3) only if a single bit fault in the flash is present and the ECC logic fails and the CRC checksum supervision fails.

EXAMPLE 2 In case three resistors are connected in series to overcome the problem of a single-point fault in the case of a short circuit, the short circuit of each individual resistor can be considered to be a safe fault as three independent short circuits are needed (n = 3).

8.1.8 Flow diagram for fault classification and fault class contribution calculation

Failure modes of a hardware element can be classified as shown in ISO 26262-5:2018, Figure B.1 and using the flow diagram described in ISO 26262-5:2018, Figure B.2. Figure 10 shows the calculation of the various failure rates considering the basic failure rate and coverage of the different failure modes (residual vs. latent).

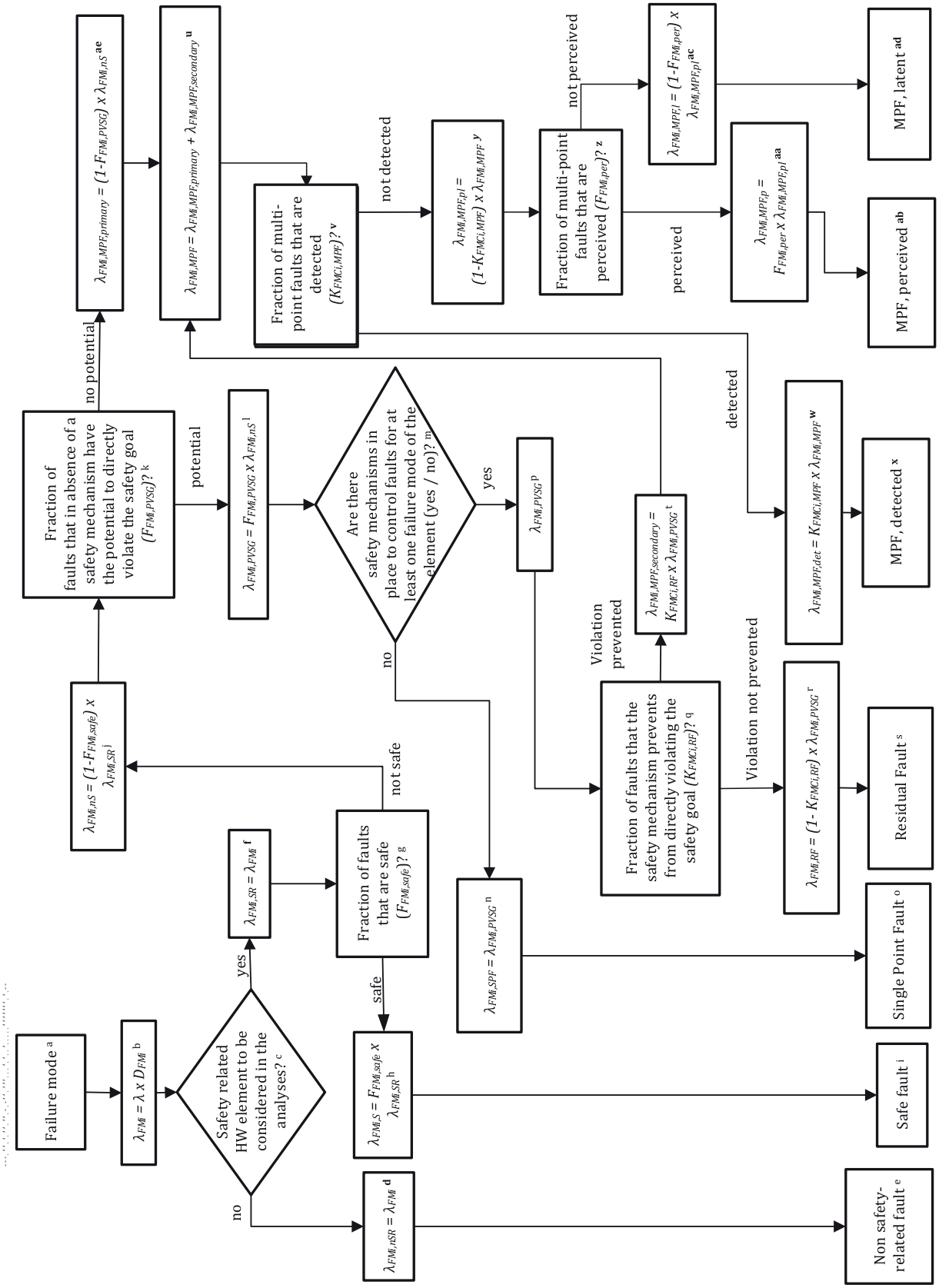


Figure 10 — Classification of failure categories and calculation of corresponding failure rates

a Failure mode to be analysed.

b λ_{FMi} is the failure rate associated with the ith failure mode of the hardware element under consideration where D_{FMi} is the failure mode distribution for the failure mode.

c If any failure mode of the HW element under consideration is safety-related, then the hardware element is safety-related.

d $\lambda_{FMi,nSR}$ is the "**n**on **S**afety-**R**elated" failure rate.

e Faults of non safety-related HW elements are not considered within the single-point fault metric or the latent-fault metric.

f $\lambda_{FMi,SR}$ is the "**S**afety-**R**elated" failure rate and is considered within the single-point fault metric and the latent-fault metric.

g $F_{FMi,safe}$ is the fraction of safe faults of this failure mode. Safe faults do not significantly contribute to the violation of the safety goal. For complex HW elements (e.g. microcontrollers) it can be difficult to give the exact proportion. In this case, a conservative F_{safe} of 0,5 (i.e. 50 %) can be assumed.

h $\lambda_{FMi,S}$ is the failure rate for the "**S**afe" faults.

i The $\lambda_{FMi,S}$ will contribute to the total rate of safe faults.

j $\lambda_{FMi,nS}$: "**n**on **S**afe" failure rate. These include the single-point faults, residual faults and multiple-point faults (with n = 2).

k $F_{FMi,PVSG}$ is the fraction of non-safe faults that have the **P**otential to directly **V**iolate the **S**afety **G**oal without considering any of the safety mechanisms that can exist to prevent this. No additional independent fault is necessary to violate the safety goal.

l $\lambda_{FMi,PVSG}$ is the failure rate of the faults that have the potential to directly violate the safety goal without considering any of the safety mechanisms that can exist to prevent this.

m Decision if the faults leading to the failure mode under consideration are single-point faults. Single-point faults have no safety mechanisms implemented to prevent any fault of the hardware element under consideration from directly violating a safety goal.

n $\lambda_{FMi,SPF}$ is the "**S**ingle-**P**oint **F**aults" failure rate. If there is not at least one safety mechanism present to control failures of the considered hardware element, all faults contributing to $\lambda_{FMi,PVSG}$ are single-point faults.

o $\lambda_{FMi,SPF}$ will contribute to the total rate of single-point faults.

p For the HW element under consideration, if there is at least one safety mechanism which prevents at least one of its failure modes from directly violating a safety goal, the faults leading to the failure under consideration are not single-point faults. In the following procedure the $\lambda_{FMi,PVSG}$ is split up into residual fault and detected, perceived and latent multiple-point faults.

q What fraction of $\lambda_{FMi,PVSG}$ is prevented by safety mechanisms from violating the safety goal? This fraction is equivalent to the failure mode coverage with respect to residual faults (see also ISO 26262-5:2018, Annex E Example calculation of hardware architectural metrics: "single-point fault metric" and "latent-fault metric"). $K_{FMCi,RF}$ is the acronym of the failure mode coverage with respect to residual faults.

r $\lambda_{FMi,RF}$ is the "**R**esidual **F**ault" failure rate.

s $\lambda_{FMi,RF}$ contributes to the total rate of residual faults.

t $\lambda_{FMi,MPF,secondary}$ is the (secondary) "**M**ultiple-**P**oint **F**aults" failure rate resulting from the $\lambda_{FMi,PVSG}$ that are controlled by a safety mechanism.

u $\lambda_{FMi,MPF}$ is the overall "**M**ultiple-**P**oint **F**aults" failure rate resulting from the primary and secondary multiple-point faults.

v Identify detected and not detected faults. $K_{FMCi,MPF}$ is the failure mode coverage with respect to multiple-point faults.

w $\lambda_{FMi,MPF,det}$ is the "**M**ultiple-**P**oint **F**aults, **det**ected" failure rate.

NOTE If the failure mode coverage with respect to multiple-point faults is different for the primary and secondary multiple-point faults then the detected multiple-point failure rate can be calculated the following way:

$$\lambda_{FMi,MPF,det} = K_{FMCi,MPF,primary} \times \lambda_{FMi,MPF,primary} + K_{FMCi,MPF,secondary} \times \lambda_{FMi,MPF,secondary}$$

x $\lambda_{FMi,MPF,det}$ contributes to the total rate of detected multiple-point faults.

y $\lambda_{FMi,MPF,pl}$ is the "**M**ultiple-**P**oint **F**aults, **p**erceived or **l**atent" failure rate.

NOTE If the failure mode coverage with respect to multiple-point faults is different for the primary and secondary multiple-point faults then the perceived or latent multiple-point failure rate can be calculated the following way:

$$\lambda_{FMi,MPF,pl} = (1 - K_{FMCi,MPF,primary}) \times \lambda_{FMi,MPF,primary} + (1 - K_{FMCi,MPF,secondary}) \times \lambda_{FMi,MPF,secondary}$$

z $F_{FMi,per}$ is the fraction of the multi-point faults that are not detected but are perceived by the driver.

aa $\lambda_{FMi,MPF,p}$ is the "**M**ultiple-**P**oint **F**aults, **p**erceived" failure rate.

NOTE If the fraction of perception is different for the primary and secondary multiple-point faults then the perceived multiple-point failure rate can be calculated the following way:

$$\lambda_{FMi,MPF,p} = F_{FMi,per,primary} \times (1 - K_{FMCi,MPF,primary}) \times \lambda_{FMi,MPF,primary} + F_{FMi,per,secondary} \times (1 - K_{FMCi,MPF,secondary}) \times \lambda_{FMi,MPF,secondary}$$

ab $\lambda_{FMi,MPF,p}$ contributes to the total rate of perceived multiple-point faults.

ac $\lambda_{FMi,MPF,l}$ is the "**M**ultiple-**P**oint **F**aults, **l**atent" failure rate.

NOTE If the fraction of perception is different for the primary and secondary multiple-point faults then the latent multiple-point failure rate can be calculated the following way:

$$\lambda_{FMi,MPF,l} = (1 - F_{FMi,per,primary}) \times (1 - K_{FMCi,MPF,primary}) \times \lambda_{FMi,MPF,primary} + (1 - F_{FMi,per,secondary}) \times (1 - K_{FMCi,MPF,secondary}) \times \lambda_{FMi,MPF,secondary}$$

ad $\lambda_{FMi,MPF,l}$ contributes to the total rate of latent multiple-point faults.

ae $\lambda_{FMi,MPF,primary}$ is the (primary) "**M**ultiple-**P**oint **F**aults" failure rate resulting from faults that contribute to the violation of the safety goal but cannot directly violate it by themselves (i.e. at least one other independent fault is needed in order to potentially violate the safety goal).

NOTE The distinction between primary and secondary multiple-point faults of a given failure mode is only useful if the associated diagnostic coverages, failure mode coverages or perceived fractions are different.

8.1.9 How to consider the failure rate of multiple-point faults related to software-based safety mechanisms addressing random hardware failures

While systematic faults of software and hardware are not quantified in the ISO 26262 series of standards, a failure rate can be calculated for random hardware failures of hardware resources that support the execution of the software-based safety mechanisms addressing random hardware failures.

If those hardware resources are shared with functions which have the potential to directly violate a safety goal, then the fault models are chosen to reflect this and potential dependent failures are considered.

8.2 Example of residual failure rate and local single-point fault metric evaluation

8.2.1 General

This example demonstrates a way to evaluate the residual failure rate $\lambda_{\mathrm{RF,Sensor}}$, the single-point failure rate λ_{SPF} and the localized version of the single-point fault metric $M_{\mathrm{SPFM,Sensor}}$ of a sensor. In this example, the sensor is compared to the value of another sensor where both sensors measure the same physical quantity and have known tolerances. The values of a sensor, A_Master, are used by a feature of the application. The values of the other sensor, A_Checker, are solely used to validate values of sensor A_Master.

This monitoring is referenced in ISO 26262-5:2018, Annex D either as “Sensor Rationality Check” or as “Input comparison/voting”.

Only faults of the sensor A_Master are classified and evaluated in this example. Faults of sensor A_Checker are not addressed here.

Since sensor A_Master has a safety mechanism defined, all remaining faults which have the potential to violate the safety goal and which are not controlled (i.e. the violation of the safety goal is not prevented) are classified as residual faults. The single-point failure rate λ_{SPF} is (per definition) equal to zero.

8.2.2 Technical safety requirement for sensor A_Master

The boundary for safe operation of sensor A_Master is shown in Figure 11, and is regarded as given within this example (i.e. the derivation from the safety goal is not discussed here). It can be expressed using the following terms:

With

$$\mu_{\mathrm{SafRel,A,min}} = \mathrm{Max}[C_{\mathrm{PVSG}}; v \times (1 + a)]$$

where

C_{PVSG} is a constant value;

$\mu_{\mathrm{SafRel,A,min}}$ is the safety-related lower boundary of sensor A_Master;

v is the physical value that is to be measured;

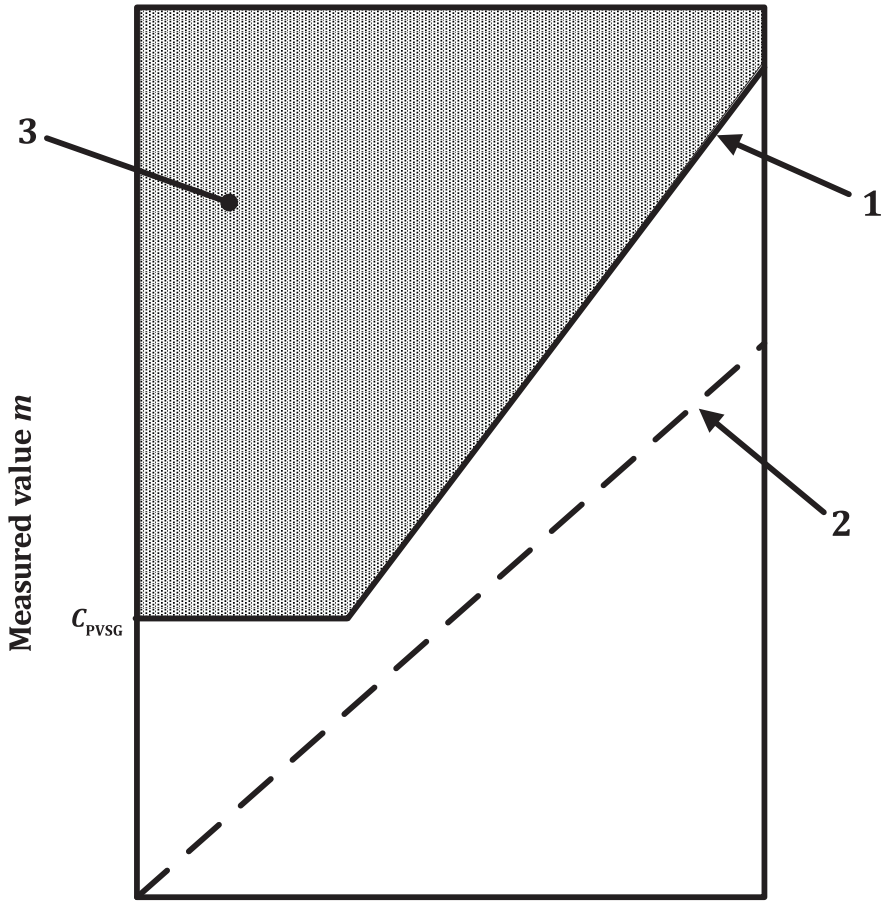
a is a constant value.

A safety-related failure of the sensor occurs when

$$m_{\mathrm{A,Master}} \geq \mu_{\mathrm{SafRel,A,min}}$$

where $m_{\mathrm{A,Master}}$ is the value reported by the sensor A_Master.

The safety requirement is to detect and control a safety-related failure of sensor A_Master within the maximum fault handling time interval of T_{SenA}.



Key

1 safety-related lower boundary $\mu_{\mathrm{SafRel,A,min}}$ of sensor A_Master
2 return value of an ideal sensor with zero tolerance (as reference)
3 faults with the potential to violate the safety goal

Figure 11 — Boundary for safe operation of sensor A_Master

In Figure 11 the x axis is the real physical value v to be measured, the y axis is the value $m_{\mathrm{A,Master}}$ reported by sensor A_Master. The dashed line shows the return value of an ideal sensor (i.e. a sensor with zero tolerances) as a reference. The solid line shows $\mu_{\mathrm{SafRel,A,min}}$. If the sensor A_Master reports a value $m_{\mathrm{A,Master}}$ that is on or above the solid line, a violation of a safety goal can occur.

8.2.3 Description of the safety mechanism

The elements of the safety mechanisms are the sensor A_Checker and the monitor hardware which consists of a microcontroller with embedded software. The software periodically compares the values of the two sensors with each other, with the periodicity being smaller than the maximum fault detection time interval T_{SenA}. The evaluation is done by the following pseudo code:

$\Delta_{\mathrm{A}} = m_{\mathrm{A,Master}} - m_{\mathrm{A,Checker}}$

if $\Delta_{\mathrm{A}} \geq \Delta_{\mathrm{Max}}$ then failure is TRUE

if failure is TRUE then switch into safe state

where

$m_{A,Master}$ is the value reported by the sensor A_Master;

$m_{A,Checker}$ is the value reported by the sensor A_Checker;

Δ_{Max} is a predefined constant maximum threshold used as pass/fail criteria.

It is assumed that the sensors have the following known tolerances:

$$m_{A,Master} = v +/- c_{A,Master}$$

$$m_{A,Checker} = v +/- c_{A,Checker}$$

where

$m_{A,Master}$ is the value reported by the sensor A_Master;

$m_{A,Checker}$ is the value reported by the sensor A_Checker;

$c_{A,Master}$ is a constant value representing the tolerance of sensor A_Master;

$c_{A,Checker}$ is a constant value representing the tolerance of sensor A_Checker;

v is the physical value to be measured.

The value Δ_{Max} is chosen so that a failure of sensor A_Master that can violate the safety goal is detected. To prevent false failure detections, Δ_{Max} is selected considering the tolerances of each sensor and other tolerances summarized in $c_{A,other}$ e.g. effects of sampling at different times:

$$\Delta_{Max} \geq c_{A,Master} + c_{A,Checker} + c_{A,other}$$

With this approach the worst case of an undetected failure is:

$$\mu_{A,Master,wc} = m_{A,Checker} + \Delta_{Max}$$

$$= v + c_{A,Checker} + \Delta_{Max}$$

where

$\mu_{A,Master,wc}$ is the worst case detection threshold, i.e. the maximum value $m_{A,Master}$ of sensor A_Master that is not detected as a failure;

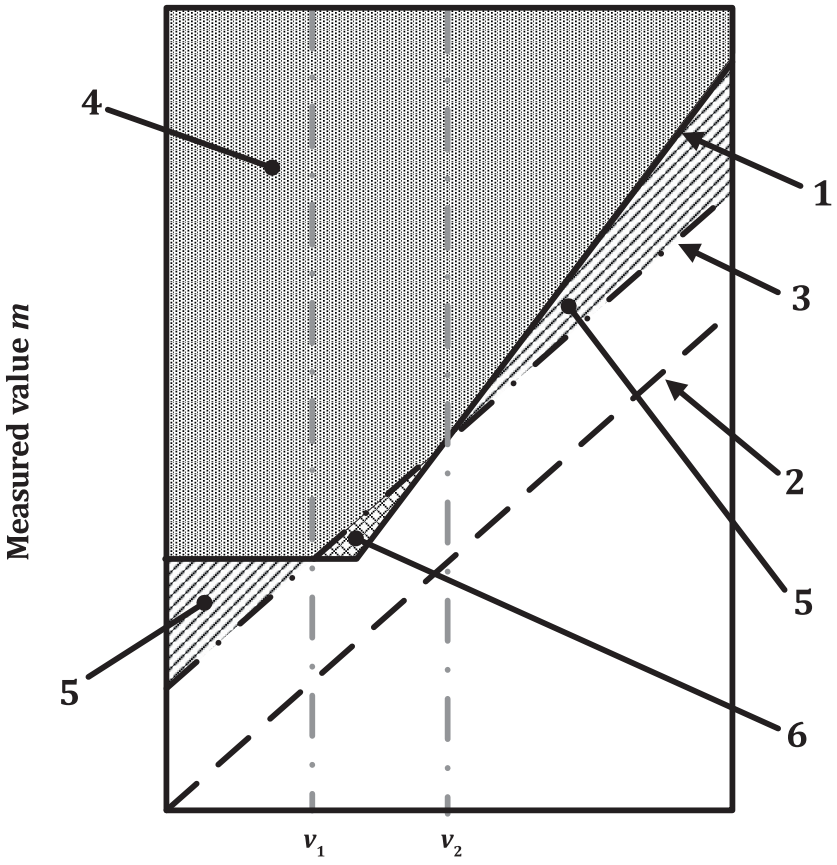
$m_{A,Checker}$ is the value reported by the sensor A_Checker;

Δ_{Max} is a predefined constant maximum threshold used as pass/fail criteria;

v is the physical value to be measured.

Every value $m_{A,Master}$ equal or above $\mu_{A,Master,wc}$ is classified as a sensor failure.

Depending on the tolerance values, different detection scenarios are possible. Two examples are visualized in Figure 12 and Figure 13.



Key

1 safety-related lower boundary $\mu_{SafRel,A,mi}$ of sensor A_Master
2 return value of an ideal sensor with zero tolerance (as reference)
3 worst case detection threshold $\mu_{A,Master,wc}$
4 dual-point faults, detected
5 detected faults with no potential to violate the safety goal
6 residual faults

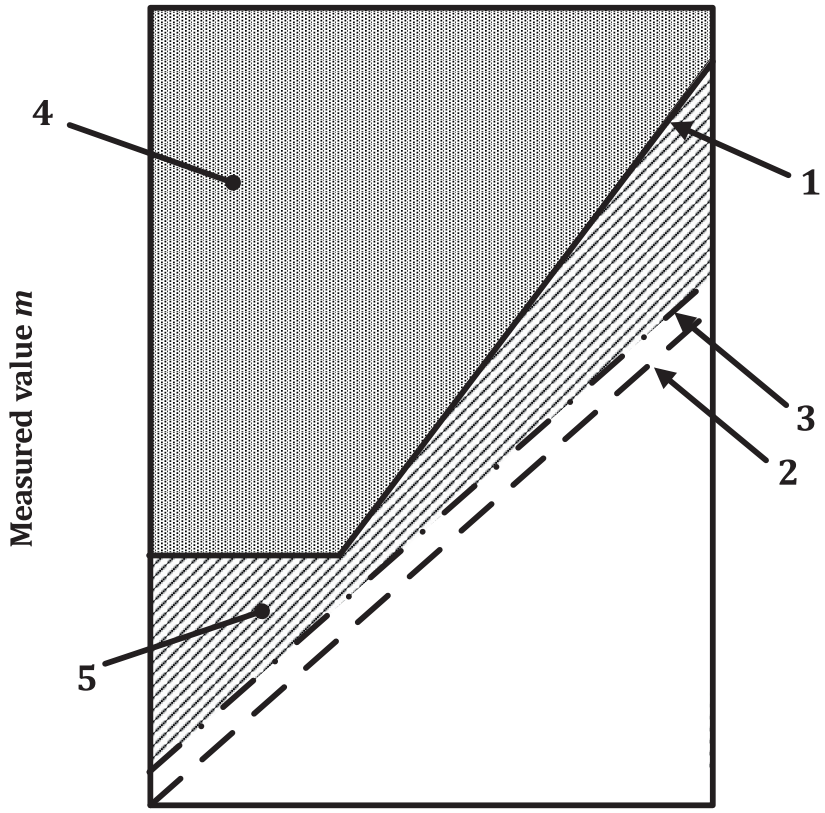
Figure 12 — Example 1 of worst case detection threshold (too high)

Three regions are indicated by arrows in Figure 12.

Region 5 — "detected faults with no potential to violate the safety goal" are faults that are detected by the safety mechanism because they are above the worst case detection threshold $\mu_{A,Master,wc}$ but alone would not cause a violation of the safety goal because they are below the safety-related lower boundary $\mu_{SafRel,A,min}$.

Region 4 — "dual-point faults, detected" are faults that could cause a violation of the safety goal but are detected and mitigated by the safety mechanism. They are above both the worst case detection threshold $\mu_{A,Master,wc}$ and the safety-related lower boundary $\mu_{SafRel,A,min}$. The dual-point nature of these faults means that it would require a failure of the safety mechanism and the sensor to cause a potential violation of the safety goal.

Region 6 — "residual faults" are not detected by the safety mechanism and can directly lead to a violation of the safety goal. The region $\mu_{SafRel,A,min} < \mu_{A,Master,wc}$ for $v \in [v_1, v_2]$ lies below the worst case detection threshold $\mu_{A,Master,wc}$ but above the safety-related lower boundary $\mu_{SafRel,A,min}$.



Key

1 safety-related lower boundary $\mu_{SafRel,A,min}$ of sensor A_Master
2 return value of an ideal sensor with zero tolerance (as reference)
3 worst case detection threshold $\mu_{A,Master,wc}$
4 dual-point faults, detected
5 detected faults with no potential to violate the safety goal

Figure 13 — Example 2 of worst case detection threshold ($M_{SPFM,Sensor}$ = 100 %)

In the case of Figure 13 the worst case detection threshold $\mu_{A,Master,wc}$ is always smaller than the safety-related lower boundary $\mu_{SafRel,A,min}$. In this case the residual failure rate is zero and the local single-point fault metric $M_{SPFM,Sensor}$ of the sensor is equal to 100 %.

8.2.4 Evaluation of example 1 described in Figure 12

8.2.4.1 General

In the case of Figure 12, there are conditions when the worst case detection threshold $\mu_{A,Master,wc}$ is higher than the safety-related lower boundary $\mu_{SafRel,A,min}$ for sensor A_Master:

for $v \in [v_1, v_2]$: $\mu_{SafRel,A,min} \leq \mu_{A,Master,wc}$

To determine the residual failure rate $\lambda_{RF,Sensor}$ and the $M_{SPFM,Sensor}$ under these conditions, further analysis is necessary. The following is an example of this analysis. In ISO 26262-5:2018, Table D.1 considers the following failure modes for sensors including signal switches:

— Out-of-range;

— Offsets;

— Stuck-in-range; and

— Oscillations.

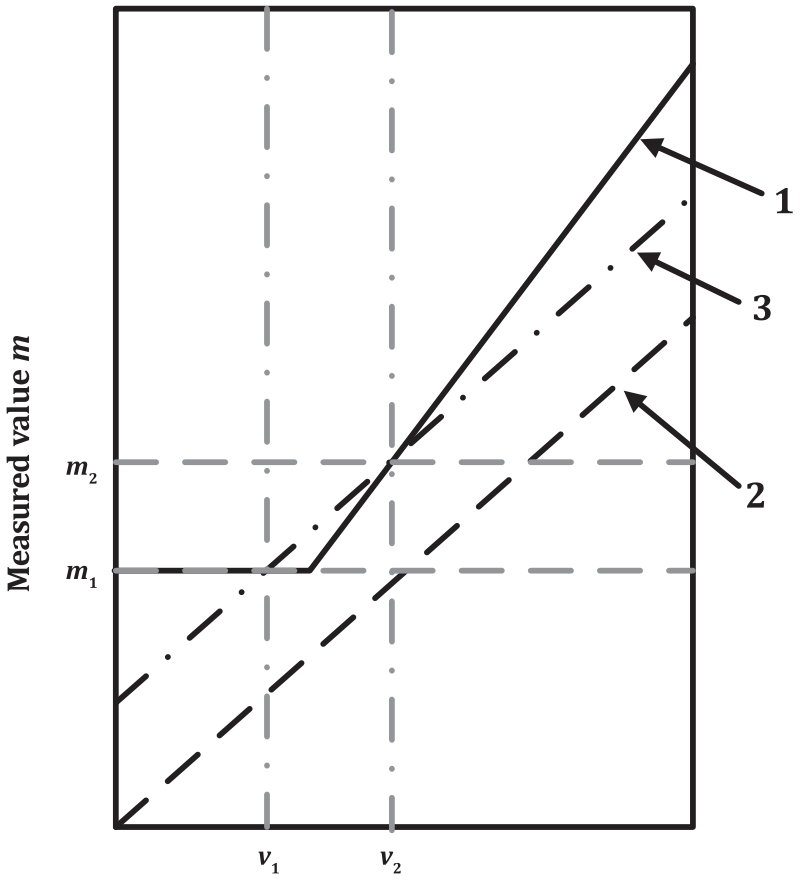
Within this example, only the stuck-at a constant value m (in range) is evaluated. For a complete assessment of the residual failure rate of the sensor and the $M_{\mathrm{SPFM,Sensor}}$ all other failure modes need evaluating.

For the analysis we distinguish three different stuck-at fault scenarios for the sensor (see Figure 14):

1) sensor stuck-at value $m > m_2$;

2) sensor stuck-at value $m < m_1$; and

3) sensor stuck-at value m between m_1 and m_2.



Key

1 safety-related lower boundary $\mu_{\mathrm{SafRel,A,min}}$ of sensor A_Master

2 return value of an ideal sensor with zero tolerance (as reference)

3 worst case detection threshold $\mu_{\mathrm{A,Master,wc}}$

Figure 14 — Stuck-at fault scenarios

The impact of the stuck-at fault of the sensor at the system level depends on the current physical value v, e.g. a stuck-at m_2 fault has the potential to violate a safety goal for the physical values $v \leq v_2$. For values $v > v_2$ this fault does not have the potential to violate a safety goal. In the following analysis the

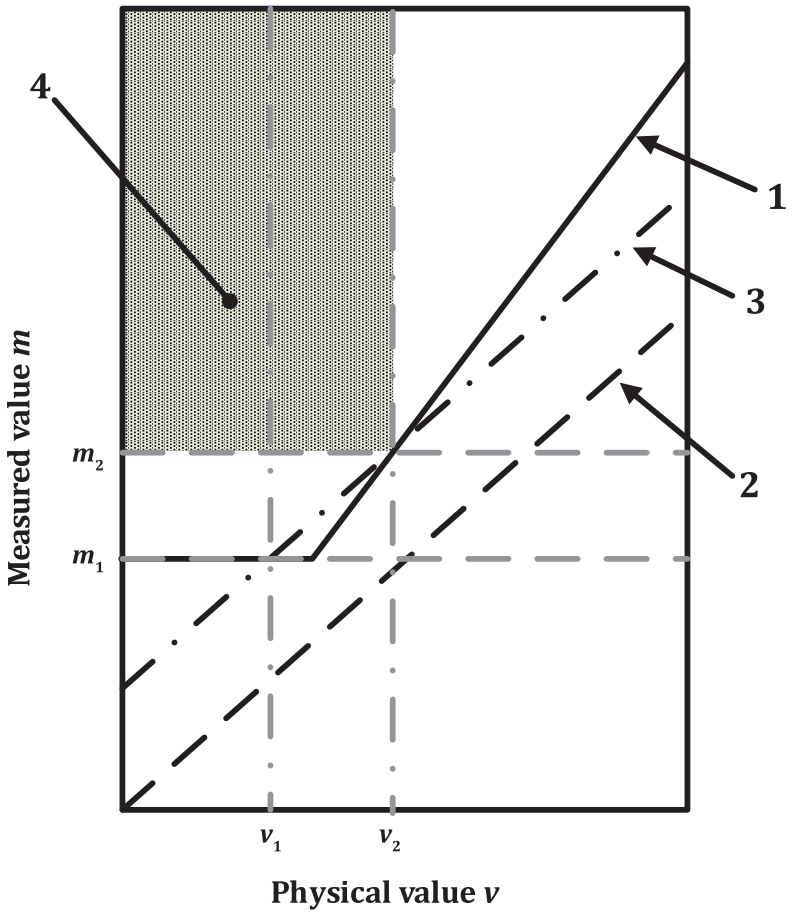
probability p_{RF} of a fault being a residual fault, is evaluated considering the detection thresholds as well as the physical values v and their probability distribution.

8.2.4.2 Case 1: Sensor stuck-at value $m > m_2$ fault

If $v \leq v_2$, the fault has the potential to violate a safety goal (see Figure 15). The sensor deviation, however, is always above the worst case detection threshold $\mu_{A,Master,wc}$, so the safety-related sensor failure is detected and controlled in time. Every fault is a detected dual-point fault. There is zero probability p_{RF} of a residual fault in the case of $v \leq v_2$. If $v > v_2$ the fault does not always have the potential to violate a safety goal (see Figure 16). If the fault has the potential to violate a safety goal (Figure 16, region 6), it will be above the worst case detection threshold and is detected in time. The faults of Figure 16 regions 4 and 5 cannot lead to a violation of a safety goal.

Some of these faults lie above the worst case detection threshold and are detected (Figure 16, region 4). There is zero probability p_{RF} of a residual fault in the case of $v > v_2$.

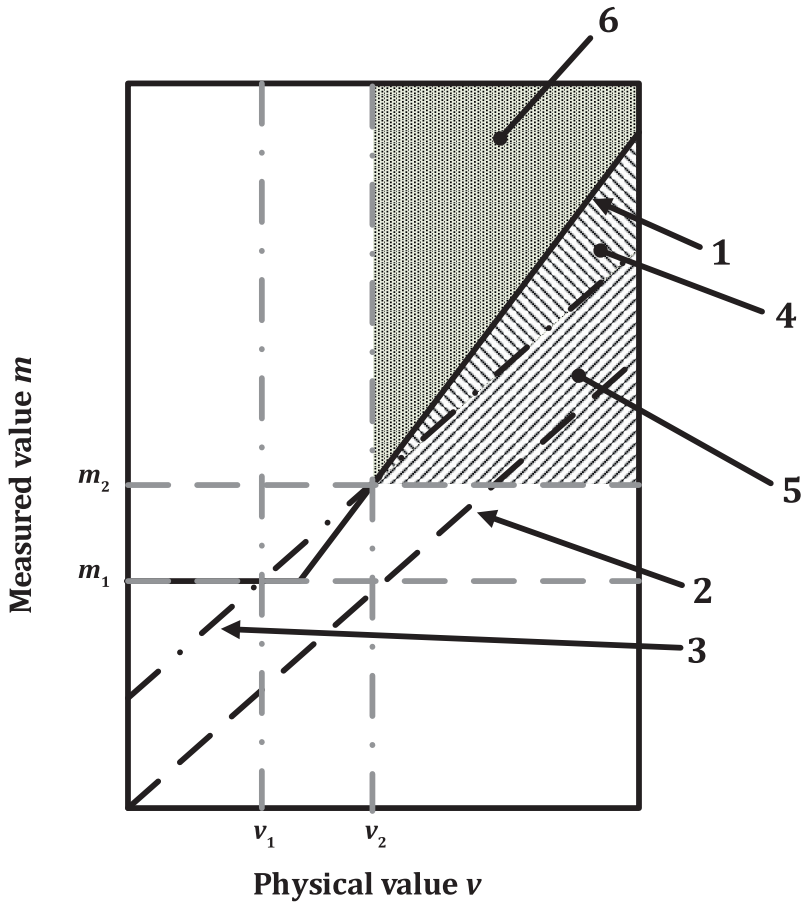
Stuck-at faults with $m > m_2$ (Figure 15 region 4 and Figure 16, regions 4, 5 and 6) have the potential to violate a safety goal if $v \leq v_2$, therefore they cannot be considered safe faults. Since all faults are detected and controlled before they can lead to the violation of a safety goal, they are detected dual-point faults; therefore, the probability $p_{RF\_stuck@m>m2}$ of a residual fault for a stuck-at $m > m_2$ fault is zero.



Key

1 safety-related lower boundary $\mu_{SafRel,A,min}$ of sensor A_Master
2 return value of an ideal sensor with zero tolerance (as reference)
3 worst case detection threshold $\mu_{A,Master,wc}$
4 dual-point faults, detected

Figure 15 — Fault classification for stuck-at $m > m_2$ fault, with $v \leq v_2$



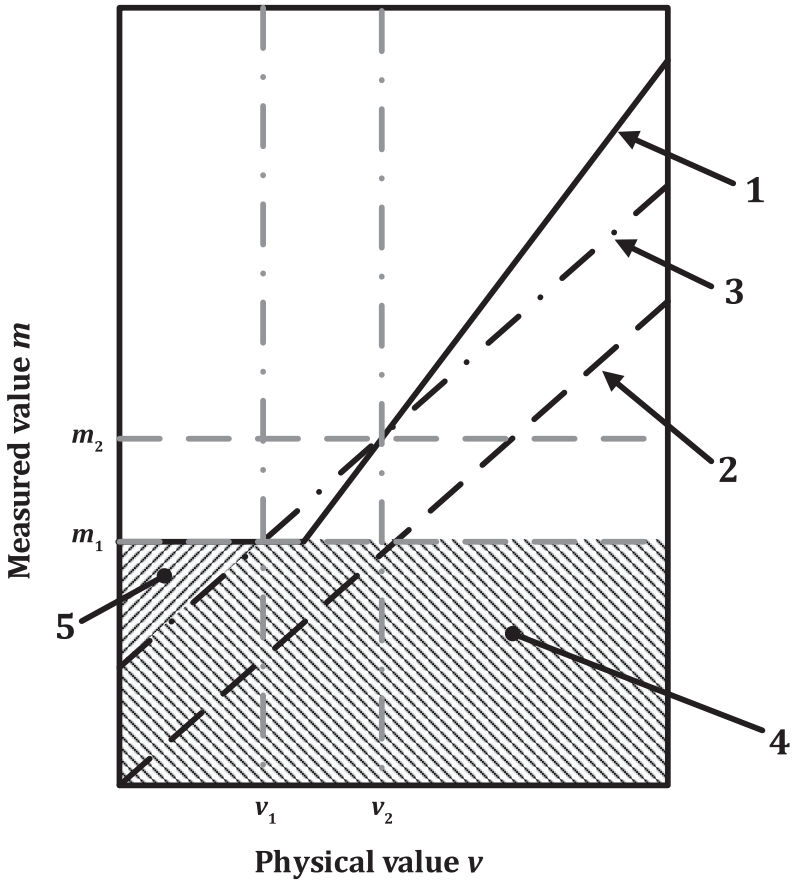
Key

1 safety-related lower boundary $\mu_{SafRel,A,min}$ of sensor A_Master
2 return value of an ideal sensor with zero tolerance (as reference)
3 worst case detection threshold $\mu_{A,Master,wc}$
4 detected faults with no potential to violate the safety goal
5 not detected faults with no potential to violate the safety goal
6 dual-point faults, detected

Figure 16 — Fault classification for stuck-at $m > m_2$ fault, with $v > v_2$

8.2.4.3 Case 2: Sensor stuck-at value $m < m_1$ fault

Stuck-at faults with $m < m_1$ are visualized in Figure 17. These faults are safe faults, as they cannot lead to a safety-related failure, as they are always below the worst case detection threshold for the whole range of physical value v. Therefore, the resulting probability $p_{RF\_stuck@m<m1}$ of a residual fault for the whole range of physical value v is zero.



Key

1 safety-related lower boundary $\mu_{SafRel,A,min}$ of sensor A_Master

2 return value of an ideal sensor with zero tolerance (as reference)

3 worst case detection threshold $\mu_{A,Master,wc}$

4 safe faults, undetected

5 safe faults, detected

Figure 17 — Fault classification for stuck-at $m < m_1$ fault

8.2.4.4 Case 3: Sensor stuck-at value $m \in [m_1, m_2]$ fault

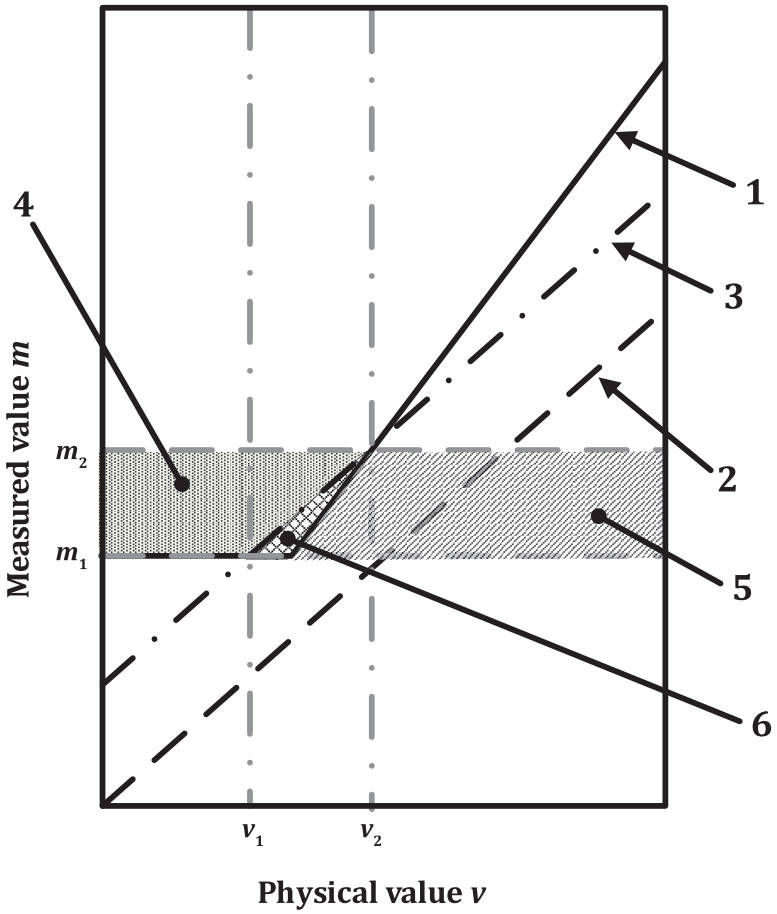
The potential to violate a safety goal and the detection of a stuck-at fault with m ∈ [m1, m2] depends on the current physical value v (see Figure 18), i.e. the risk of a violation of a safety goal depends on the current value of v at the time when the fault occurs. The probability of a stuck-at residual fault, pRF_stuck@m∈[m1,m2], is evaluated for three different intervals of v at the time of fault occurrence:

— $v < v_1$;

— $v_1 \leq v \leq v_2$; and

— $v > v_2$.

For each of these conditions, the probability of a residual fault is evaluated separately. The final probability of a residual fault is the expectation value of these three probabilities:



Key

1 safety-related lower boundary $\mu_{SafRel,A,min}$ of sensor A_Master
2 return value of an ideal sensor with zero tolerance (as reference)
3 worst case detection threshold $\mu_{A,Master,wc}$
4 dual-point faults, detected
5 faults that do not violate the safety goal but remain undetected
6 residual faults

Figure 18 — Fault classification for stuck-at $m \in [m_1, m_2]$ fault

Depending on the current value of *v*, the faults can be detected dual-point faults (region 4), residual faults (region 6) or do not have the potential to violate the safety goal (region 5).

$$p_{\text{RF\_stuck@}m\in[m1,m2]} = p_{\text{RF\_stuck@}m\in[m1,m2],v<v1} \times p_{v<v1}$$

$$p_{\text{RF\_stuck@}m\in[m1,m2],v1\le v\le v2} \times p_{v1\le v\le v2}$$

$$p_{\text{RF\_stuck@}m\in[m1,m2],v>v2} \times p_{v>v2}$$

where

$p_{\text{RF\_stuck@}m\in[m1,m2]}$ is the probability that a stuck-at value m sensor fault, with $m \in [m_1, m_2]$, manifests itself as a residual fault;

$p_{\text{RF\_stuck@}m\in[m1,m2],v<v1}$ is the probability that a stuck-at value m sensor fault, with $m \in [m_1, m_2]$, manifests itself as a residual fault if the $v < v_1$ at the point of time when the fault occurs;

$p_{v<v1}$ is the probability that $v < v_1$ at the point of time when the fault occurs;

$p_{\text{RF\_stuck@}m\in[m1,m2],v1\le v\le v2}$ is the probability that a stuck-at value m sensor fault, with $m \in [m_1, m_2]$, manifests itself as a residual fault if $v_1 \le v \le v_2$ at the point of time when the fault occurs;

$p_{v1\le v\le v2}$ is the probability that $v_1 \le v \le v_2$ at the point of time when the fault occurs;

$p_{\text{RF\_stuck@}m\in[m1,m2],v>v2}$ is the probability that a stuck-at value m sensor fault, with $m \in [m_1, m_2]$, manifests itself as a residual fault if $v > v_2$ at the point of time when the fault occurs;

$p_{v>v2}$ is the probability that $v > v_2$ at the point of time when the fault occurs;

$p_{v<v1} + p_{v1\le v\le v2} + p_{v>v2} = 1$.

If $v < v_1$ the stuck-at faults have the potential to violate a safety goal, but are detected in time. The probability $p_{\text{RF\_stuck@}m\in[m1,m2],v<v1}$ of a residual fault is zero.

If $v > v_2$ the stuck-at fault does not have the potential to violate a safety goal, but it is not detected. Since sooner or later the value v is in between v_1 and v_2, $p_{\text{RF\_stuck@}m\in[m1,m2],v>v2} = p_{\text{RF\_stuck@}m\in[m1,m2],v1\le v\le v2}$.

If $v_1 \le v \le v_2$ the probability $p_{\text{RF\_stuck@}m\in[m1,m2],\ v1\le v\le v2}$ of a residual fault is not zero.

Its exact determination of the probability to remain in the residual fault area long enough to lead to a potential violation of a safety goal is not trivial. It can depend on parameters such as:

— dynamic behaviour of the physical value v and its corresponding probability distributions, e.g. a temperature value is more of a static signal while the angle position of an electric motor in use is more of a dynamic signal;

— probability distribution of value v within $v \in [v_1, v_2]$;

— reaction time of the monitoring software, e.g. due to filtering times. In the example a single event with $\Delta_A \ge \Delta_{Max}$ is enough to detect a sensor failure and switch into the safe state. It is common practice however to implement an error counter, so that more than one event is necessary in order to assess a sensor failure and switch into the safe state. Especially error counter recovery, e.g. resetting the error counter once one not safety-related event (in this example this would correspond to $\Delta_A < \Delta_{Max}$) is detected, can have a significant impact on the detection capability of the monitoring software, drastically reducing it; and

— the number of measured safety-related sensor deviations necessary to lead to a potential violation of a safety goal. Also the number of valid measurements that must lie between two measured safety-related sensor deviations, so that the safety goal is not violated anymore, can be of interest.

If the exact detail of each influencing parameter is not available, it is legitimate to use expert judgement and engineering practises (e.g. using an equal distribution for unknown probability distributions) to derive a conservative estimate.

Having assessed the different probabilities $p_{RF\_stuck@m>m2}$, $p_{RF\_stuck@m<m1}$ and $p_{RF\_stuck@m\in[m1,\ m2]}$ the probability $p_{RF\_stuck@m}$ of a sensor stuck-at residual fault can be calculated:

$$p_{RF,stuck@m} = p_{RF,stuck@m<m1} \times p_{m<m1} + p_{RF,stuck@m\in[m1,m2]} \times p_{m1\le m\le m2} + p_{RF,stuck@m>m2} \times p_{m>m2}$$

where

$p_{m<m1}$ is the probability of a stuck-at $m < m_1$ fault;

$p_{m1\le m\le m2}$ is the probability of a stuck-at $m_1 \le m \le m_2$ fault;

$p_{m>m2}$ is the probability of a stuck-at $m > m_2$ fault;

$p_{m<m1} + p_{m1\le m\le m2} + p_{m>m2} = 1$.

8.2.4.5 Final residual failure rate assessment

If each relevant failure mode FM_i is assessed the same way as above, the overall probability $p_{RF,Sensor}$ of a sensor fault manifesting itself as a residual fault can be calculated:

$$p_{RF,Sensor} = \sum_i p_{FM,i} \times p_{RF,FM,i}$$

where

$p_{FM,i}$ is the probability of failure mode FM_i;

$P_{RF,FM,i}$ is the probability that failure mode FM_i manifests itself as a residual fault;

$\sum_i p_{FM,i} = 1$.

With this probability, the residual failure rate, $\lambda_{RF,\ Sensor}$, can be assessed as

$$\lambda_{RF,Sensor} = p_{RF,Sensor} \times \lambda_{RF,Sensor}$$

leading to a $M_{SPFM,Sensor}$ of

$$M_{SPFM,Sensor} = 1 - {\lambda_{RF,Sensor}}/{\lambda_{Sensor}} = 1 - p_{RF,Sensor}$$

8.2.4.6 Improvement of $SPFM_{Sensor}$

An efficient way to reduce the residual failure rate of the sensor is to reduce the value of Δ_{Max}. The reduction of Δ_{Max} could be done without a significant increase of false detection under the following conditions:

— The probability distribution of the tolerances could show that the estimated worst case scenario is extremely unlikely. Therefore, the probability of a false alarm is sufficiently low and therefore acceptable.

— A redesign of the system can lead to improved tolerance values.

Note that in this example only sensor faults are evaluated, not faults occurring in the remaining sensor path. The malfunction of shared HW resources which could lead to a malfunction of both sensors or which could falsify both sensor values, e.g. the ADC of the microcontroller, are evaluated separately. In addition, a dependent failure analysis as given in ISO 26262-9:2018, Clause 7 is done.

8.3 Further explanation concerning hardware

8.3.1 How to deal with microcontrollers in the context of an ISO 26262 series of standards application

Microcontrollers are an integral component of modern E/E automotive systems. They can be developed as a Safety Element out of Context (SEooC, see Clause 9).

Their complexity is handled by combining qualitative and quantitative safety analyses of the microcontroller's parts and subparts, performed at the appropriate level of abstraction, i.e. from block diagram during concept phase to the netlist and layout level during product development phases.

A guideline, including a non-exhaustive list of examples of how to deal with microcontrollers in the context of the ISO 26262 series of standards, is described in ISO 26262-11.

It describes a method for the calculation of failure rates of a microcontroller, including how to consider permanent and transient faults.

It includes examples of:

— dependent failures analysis;

— avoidance of systematic failures during microcontroller design;

— verification of the safety mechanisms of the microcontroller; and

— consideration of the microcontroller stand-alone analysis at the system-level.

8.3.2 Safety analysis methods

8.3.2.1 General

Annex A discusses techniques for analysing system fault modes, including inductive and deductive analysis.

8.3.2.2 Consideration of exposure duration in the calculation of Probabilistic Metric for random Hardware Failures (PMHF)

As described in ISO 26262-5:2018, 9.4.2.4, quantitative analysis provides evidence that target values of requirement ISO 26262-5:2018, 9.4.2.1 have been achieved. As given in ISO 26262-5:2018, 9.4.2.4, this quantitative analysis considers the exposure duration in the case of dual-point faults. Failure scenarios of higher order than n = 2 in this example are considered as safe and are not included in the calculation.

Based on the NOTE 2 in ISO 26262-5:2018, 9.4.2.4, the exposure duration starts as soon as the fault occurs.

It includes:

— the multiple-point fault detection interval associated with each safety mechanism, or the lifetime of the vehicle if the fault is not indicated to the driver (latent fault);

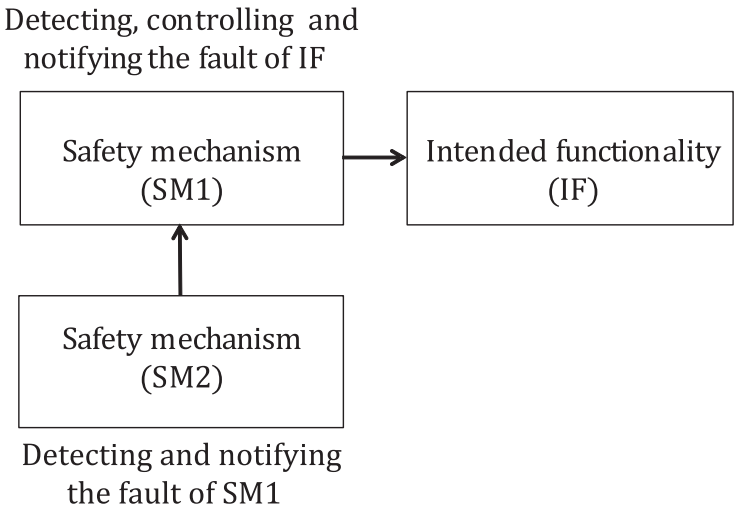
— the maximum duration of a trip (in the case where the driver is requested to stop in a safe manner); and

— the average time interval between a warning and the vehicle repair in a workshop (in the case where the driver is alerted to have the vehicle repaired).

8.3.2.3 Typical pattern of dual point failure (Intended functionality and safety mechanism)

The following example is provided to show a possible way to consider the exposure duration. In this example, it is assumed that an intended functionality (block "IF") is supervised by a safety mechanism "SM".

The architecture assumption is shown in Figure 19. The example assumes that the faults of the intended functionality IF are detected and mitigated by the safety mechanism SM1. SM1 is also responsible for notification of IF fault status to the driver. In addition, faults in safety mechanism SM1 are detected by another safety mechanism SM2 which is responsible for the mitigation of SM1 faults and the notification of SM1 fault status to the driver.



Key

⟶ Safety mechanism at origin of arrow detects fault in component at tip of arrow

Figure 19 — System architectural design of the example

Figure 19 shows a typical dual point failure path of intended functionality (IF) and safety mechanism (SM1), which is intended to detect failures in IF. Assuming that there are no dependent failures between SM1 and IF, the dual point failures resulting from the combination of IF and SM1 are determined considering

— order of occurrence of the faults;

— rate of detecting and controlling the first fault;

— rate of notifying to the driver of a detected fault; and

— time to repair after driver notification.

From the above consideration, four cases of dual point failure can be listed as shown in Table 2.

Table 2 — Patterns of dual point failure in the example architecture

	First fault: SM1 → Second fault: IF	First fault: IF → Second fault: SM1
Cannot notify the driver	Pattern 1 A fault in SM1 is mitigated by SM2 but not notified. The exposure duration of the fault is taken as the vehicle lifetime which is the worst case exposure duration. Or A fault in SM1 is not mitigated by SM2. The exposure duration of the fault is taken as the vehicle lifetime which is the worst case exposure duration.	Pattern 3 A fault in IF is mitigated by SM1 but not notified. The exposure duration of the fault is taken as the vehicle lifetime which is the worst case exposure duration.
Can notify the driver	Pattern 2 A fault in SM1 is mitigated and notified by SM2. The exposure duration of the fault is taken as the expected time required for the driver to take the vehicle in for repair.	Pattern 4 A fault in IF is mitigated and notified by SM1. The exposure duration of the fault is taken as the expected time required for the driver to take the vehicle in for repair.

8.3.2.4 The formula for calculation

The formula in this sub-clause refers to the patterns listed in Table 2 and the content of ISO 26262-5:2018, 9.4.2.4.

$$M_{PMHF} = \lambda_{SPF} + \lambda_{RF}$$

$$+ 0{,}5 \times \lambda_{SM1,DPF,latent} \times \lambda_{IF,DPF} \times T_{lifetime} \quad \text{Pattern 1}$$

$$+ \lambda_{SM1,DPF,detected} \times \lambda_{IF,DPF} \times T_{service} \quad \text{Pattern 2}$$

$$+ 0{,}5 \times \lambda_{IF,DPF,latent} \times \lambda_{SM1,DPF} \times T_{lifetime} \quad \text{Pattern 3}$$

$$+ \lambda_{IF,DPF,detected} \times \lambda_{SM1,DPF} \times T_{service} \quad \text{Pattern 4}$$

where

M_{PMHF} is the PMHF value determined using ISO 26262-5:2018, 9.4.2.2;

λ_{SPF} is the single point failure rate;

λ_{RF} is the residual failure rate;

$\lambda_{IF,DPF}$ is the dual point failure rate for IF;

$\lambda_{IF,DPF.detected}$ is the IF's detected and notified dual point failure rate;

$\lambda_{IF,DPF.latent}$ is the IF's latent dual point failure rate (mitigated but not notified);

$\lambda_{SM1,DPF}$ is the SM1's dual point failure rate;

$\lambda_{SM1,DPF,detected}$ is the SM1's detected and notified dual point failure rate;

$\lambda_{SM1,DPF,latent}$ is the SM1's latent dual point failure rate;

$T_{lifetime}$ is the vehicle lifetime;

$T_{service}$ is the expected time to repair after notification provided to driver.

NOTE 1 In this example, since all hardware elements are monitored by a safety mechanism, the single-point failure rate is equal to zero ($\lambda_{SPF} = 0$).

NOTE 2 In pattern 1 and 3, the order which the individual faults for the dual-point failure occur is important. In pattern 1, the latent dual-point fault of the SM1 occurs before the dual-point fault of the IF. In pattern 3, the latent dual-point fault of the IF occurs before the dual-point fault of the SM1.

Using the terms defined in 8.1.8 the different dual-point failure rates can be calculated as follows:

$$\lambda_{IF,DPF} = \lambda_{IF,DPF,primary} + \lambda_{IF,DPF,secondary}$$

$$\lambda_{IF,DPF,primary} = (1 - F_{IF,safe}) \times (1 - F_{IF,PVSG}) \times \lambda_{IF}$$

$$\lambda_{IF,DPF,secondary} = (1 - F_{IF,safe}) \times F_{IF,PVSG} \times K_{FMC,SM1,RF} \times \lambda_{IF}$$

$$\lambda_{IF,DPF.detected} = \lambda_{IF,DPF,detected,primary} + \lambda_{IF,DPF,detected,secondary}$$

$$\lambda_{IF,DPF,detected,primary} = \lambda_{IF,DPF,primary} \times K_{FMC1,SM1,MPF} = (1 - F_{IF,safe}) \times (1 - F_{IF,PVSG}) \times K_{FMC1,SM1,MPF} \times \lambda_{IF}$$

$$\lambda_{IF,DPF,detected,secondary} = \lambda_{IF,DPF,secondary} \times K_{FMC2,SM1,MPF} = (1 - F_{IF,safe}) \times F_{IF,PVSG} \times K_{FMC,SM1,RF} \times K_{FMC2,SM1,MPF} \times \lambda_{IF}$$

$$\lambda_{IF,DPF.latent} = \lambda_{IF,DPF,latent,primary} + \lambda_{IF,DPF,latent,secondary}$$

$$\lambda_{IF,DPF,latent,primary} = \lambda_{IF,DPF,primary} \times (1 - K_{FMC1,SM1,MPF}) = (1 - F_{IF,safe}) \times (1 - F_{IF,PVSG}) \times (1 - K_{FMC1,SM1,MPF}) \times \lambda_{IF}$$

$$\lambda_{IF,DPF,latent,secondary} = \lambda_{IF,DPF,secondary} \times (1 - K_{FMC2,SM1,MPF}) = (1 - F_{IF,safe}) \times F_{IF,PVSG} \times K_{FMC,SM1,RF} \times (1 - K_{FMC2,SM1,MPF}) \times \lambda_{IF}$$

$$\lambda_{SM1,DPF} = \lambda_{SM1,DPF,primary} + \lambda_{SM1,DPF,secondary}$$

$$\lambda_{SM1,DPF,primary} = (1 - F_{SM1,safe}) \times (1 - F_{SM1,PVSG}) \times \lambda_{SM1}$$

$$\lambda_{SM1,DPF,secondary} = (1 - F_{SM1,safe}) \times F_{SM1,PVSG} \times K_{FMC,SM2,RF} \times \lambda_{SM1}$$

$$\lambda_{SM1,DPF.detected} = \lambda_{SM1,DPF,detected,primary} + \lambda_{SM1,DPF,detected,secondary}$$

$$\lambda_{SM1,DPF,detected,primary} = \lambda_{SM1,DPF,primary} \times K_{FMC1,SM2,MPF} = (1 - F_{SM1,safe}) \times (1 - F_{SM1,PVSG}) \times K_{FMC1,SM2,MPF} \times \lambda_{SM1}$$

$$\lambda_{SM1,DPF,detected,secondary} = \lambda_{SM1,DPF,secondary} \times K_{FMC2,SM2,MPF} = (1 - F_{SM1,safe}) \times F_{SM1,PVSG} \times K_{FMC,SM2,RF} \times K_{FMC2,SM2,MPF} \times \lambda_{SM1}$$

$$\lambda_{SM1,DPF.latent} = \lambda_{SM1,DPF,latent,primary} + \lambda_{SM1,DPF,latent,secondary}$$

$$\lambda_{SM1,DPF,latent,primary} = \lambda_{SM1,DPF,primary} \times (1 - K_{FMC1,SM2,MPF}) = (1 - F_{SM1,safe}) \times (1 - F_{SM1,PVSG}) \times (1 - K_{FMC1,SM2,MPF}) \times \lambda_{SM1}$$

$$\lambda_{SM1,DPF,latent,secondary} = \lambda_{SM1,DPF,secondary} \times (1 - K_{FMC2,SM2,MPF}) = (1 - F_{SM1,safe}) \times F_{SM1,PVSG} \times K_{FMC,SM1,RF} \times (1 - K_{FMC2,SM2,MPF}) \times \lambda_{SM1}$$

where

λ_{IF} is the IF's failure rate;

λ_{SM1} is the SM1's failure rate;

$F_{IF,safe}$ is the ratio of safe faults of the IF;

$F_{SM1,safe}$ is the ratio of safe faults of SM1;

$F_{IF,PVSG}$ is the ratio of faults of the IF that have the potential to directly violate a safety goal in absence of a safety mechanism;

$F_{SM1,PVSG}$ is the ratio of faults of SM1 that have the potential to directly violate a safety goal in absence of a safety mechanism.

NOTE Failure of some safety mechanisms can cause a safety goal violation by themselves, e.g. an ECC can corrupt a correct value by falsely correcting it.

$K_{FMC,SM1,RF}$ is the IF's diagnostic coverage with respect to residual faults (SM1);

$K_{FMC1,SM1,MPF}$ is the IF's detection and notification diagnostic coverage with respect to primary multiple-point faults (SM1);

$K_{FMC2,SM1,MPF}$ is the IF's detection and notification diagnostic coverage with respect to secondary multiple-point faults (SM1);

$K_{FMC,SM2,RF}$ is SM1's diagnostic coverage with respect to residual faults (SM2);

$K_{FMC1,SM2,MPF}$ is SM1's detection and notification diagnostic coverage with respect to primary multiple-point faults (SM2);

$K_{FMC2,SM2,MPF}$ is SM1's detection and notification diagnostic coverage with respect to secondary multiple-point faults (SM2).

Table 3 — Example failure rates for the example architecture in Figure 19

Component Name	Failure rate (e-9/h)	Failure Mode	Failure Mode Distribution	Failure mode has the potential to violate the safety goal in absence of safety mechanisms	Safety mechanism(s) preventing the failure mode from violating the safety goal	Failure mode coverage wrt. violation of safety goal	Residual or Single-Point Fault failure rate (e-9/h)	Dual Point Fault failure rate (e-9/h)	Description	Detection means or Safety mechanism(s) preventing the failure mode from being latent	Failure mode coverage with respect to latent failures	Latent Multiple-Point Fault failure rate (e-9/h)	Description	Detected Dual Point Fault failure rate (e-9/h)	Description
IF	100	Safe	0,5												
		FM A	0,2	X	SM1	0,9	2,0	18,0	$\lambda_{IF,DPF,secondary}$	SM1	0,8	3,6	$\lambda_{IF,DPF,latent,secondary}$	14,4	$\lambda_{IF,DPF,detected,secondary}$
		FM B	0,15	X	none	0	15,0								
		FM C	0,15					15,0	$\lambda_{IF,DPF,primary}$	SM1	0,7	4,5	$\lambda_{IF,DPF,latent,primary}$	10,5	$\lambda_{IF,DPF,detected,primary}$
SM1	50	Safe	0,5												
		FM D	0,15	X	SM2	0,8	1,5	6,0	$\lambda_{SM1,DPF,primary}$	SM2	0,6	2,4	$\lambda_{SM1,DPF,latent,secondary}$	3,6	$\lambda_{SM1,DPF,detected,secondary}$
		FM E	0,2					10,0	$\lambda_{SM1,DPF,primary}$	SM2	0,4	6,0	$\lambda_{SM1,DPF,latent,primary}$	4,0	$\lambda_{SM1,DPF,detected,primary}$
		FM F	0,15					7,5	$\lambda_{SM1,DPF,primary}$	none	—	7,5	$\lambda_{SM1,DPF,latent,primary}$		$\lambda_{SM1,DPF,detected,primary}$

EXAMPLE The formula M_{PMHF} in 8.3.2.4 can be calculated based on the values in Table 3 according to the equations for dual-point failure rates calculation as follows:

$\lambda_{IF,DPF}$ = 33e-9/h

$\lambda_{IF,DPF,detected}$ = 24,9e-9/h

$\lambda_{IF,DPF,latent}$ = 8,1e-9/h

$\lambda_{SM1,DPF}$ = 23,5e-9/h

$\lambda_{SM1,DPF,detected}$ = 7,6e-9/h

$\lambda_{SM1,DPF,latent}$ = 15,9e-9/h

M_{PMHF} = 18,5e-9/h + 0,5 × 15,9e-9/h × 33e-9/h × 10 000 h + 7,6e-9/h × 33e-9/h × 20h + 0,5 × 8,1e-9/h × 23,5e-9/h × 10 000 h + 24,9e-9/h × 23,5e-9/h × 20 h = 18,504e-9/h

If for example

$F_{IF,safe}$ = 0 (the IF has no safe faults),

$F_{SM1,safe}$ = 0 (SM1 has no safe faults),

$F_{IF,PVSG}$ = 1 (the IF has only faults with the potential to violate the safety goal in absence of a safety mechanism), and

$F_{SM1,PVSG}$ = 0 (SM1 has no faults with the potential to violate the safety goal in absence of a safety mechanism),

the dual-point failure rate could be calculated as follows:

$$\lambda_{IF,DPF} = K_{FMC,SM1,RF} \times \lambda_{IF}$$

$$\lambda_{IF,DPF.detected} = K_{FMC,SM1,RF} \times K_{FMC2,SM1,MPF} \times \lambda_{IF}$$

$$\lambda_{IF,DPF.latent} = K_{FMC,SM1,RF} \times (1 - K_{FMC2,SM1,MPF}) \times \lambda_{IF}$$

$$\lambda_{SM1,DPF} = \lambda_{SM1}$$

$$\lambda_{SM1,DPF.detected} = K_{FMC1,SM2,MPF} \times \lambda_{SM1}$$

$$\lambda_{SM1,DPF.latent} = (1 - K_{FMC1,SM2,MPF}) \times \lambda_{SM1}$$

The formulas in this sub-clause assume an exponential failure rate model and first order approximation [e.g. $T_{lifetime} \times \lambda_{SM1}$ and $T_{lifetime} \times \lambda_{IF}$ both small (typically <0,1)].

The contribution of $T_{service}$ is evaluated in the following cases, where the calculation of M_{PMHF} is conducted to verify if the PMHF target value can be achieved with the considered HW design assumptions:

a) If the PMHF target value is higher than or equal to $\lambda_{SPF} + \lambda_{RF} + \lambda_{SM1,DPF} \times \lambda_{IF,DPF} \times T_{lifetime}$, the PMHF target value can be achieved independent of the value of $T_{service}$.

NOTE $M_{PMHF} = \lambda_{SPF} + \lambda_{RF} + \lambda_{SM1,DPF} \times \lambda_{IF,DPF} \times T_{lifetime}$ when all dual point faults are assumed to be latent for calculation.

b) If the PMHF target value is lower than $\lambda_{SPF} + \lambda_{RF} + (\lambda_{SM1,DPF,latent} \times \lambda_{IF,DPF} + \lambda_{IF,DPF,latent} \times \lambda_{SM1,DPF}) \times 0,5 \times T_{lifetime}$, the PMHF target value cannot be achieved independent of the value of $T_{service}$.

NOTE $M_{PMHF} = \lambda_{SPF} + \lambda_{RF} + (\lambda_{SM1,DPF,latent} \times \lambda_{IF,DPF} + \lambda_{IF,DPF,latent} \times \lambda_{SM1,DPF}) \times 0,5 \times T_{lifetime}$ when $T_{service}$ is assumed to be equal to zero for calculation.

c) If the PMHF target value is lower than $\lambda_{SPF} + \lambda_{RF} + \lambda_{SM1,DPF} \times \lambda_{IF,DPF} \times T_{lifetime}$, and is higher than or equal to $\lambda_{SPF} + \lambda_{RF} + (\lambda_{SM1,DPF,latent} \times \lambda_{IF,DPF} + \lambda_{IF,DPF,latent} \times \lambda_{SM1,DPF}) \times 0,5 \times T_{lifetime}$, the PMHF target value can be achieved if the value of $T_{service}$ satisfies the following equation:

$T_{service} \leq (\text{PMHF target value} - \lambda_{SPF} - \lambda_{RF} - (\lambda_{SM1,DPF,latent} \times \lambda_{IF,DPF} + \lambda_{IF,DPF,latent} \times \lambda_{SM1,DPF}) \times 0,5 \times T_{lifetime}) / (\lambda_{SM1,DPF,detected} \times \lambda_{IF,DPF} + \lambda_{IF,DPF,detected} \times \lambda_{SM1,DPF})$.

NOTE This equation is used in 12.3.1.2.

8.4 PMHF units — Average probability per hour

Reliability analysis commonly provides the failure rate for individual components or parts. Functional safety needs to consider the effect of fault detection, control and notification functions provided by safety mechanisms. Therefore, even if the same units (1/h) as reliability analysis is used, the meaning is not the same.

ISO 26262-5:2018, 9.4.2.1 gives the units for the PMHF calculation as average probability per hour over the operational lifetime of the item. NOTE 1 of the sub-clause points out that failure rate and average probability of failure per hour over the operational lifetime of the item are different values even if they share the same units.

The PMHF calculation is used to determine if the risk of safety goal violation due to random hardware failure of the item is sufficiently low with respect to the assigned ASIL. PMHF does not show how often random hardware faults occur. Even if the failure rate of a hardware element is high, the PMHF may be low due to good hardware architectural design including safety mechanisms.

PMHF consists of contributions from single point faults (8.1.2), residual faults (8.1.3), detected or perceived dual point faults (8.1.4 and 8.1.5), and latent faults (8.1.6). The average probability per hour over the operational lifetime of the item of each type of fault contributes to PMHF differently.

The following shows the derivation of the average probability of failure per hour over the operational lifetime of the item regarding single point faults.

$$\text{Prob}(T \leq t) = F(t) = 1 - R(t), t \geq 0$$

where

$\text{Prob}(T \leq t)$ is the probability that failure occurs until t;

$F(t)$ is the failure distribution;

t is the time;

T is the time the failure occurs;

$R(t)$ is the system reliability over time.

$F(t)$ is represented as:

$$F(t) = \int_0^t f(\tau) \times d\tau$$

where $f(t)$ is failure density function.

The instantaneous failure rate $\lambda(t)$ is:

$$\lambda(t) = \lim_{\Delta t \to 0}\left[\frac{R(t)-R(t+\Delta t)}{\Delta t \times R(t)}\right] = \frac{f(t)}{R(t)} = \frac{f(t)}{1-F(t)}$$

In general, failure rate is considered as constant and failure density has exponential distribution.

Therefore, $F(t) = 1 - e^{-\lambda t}$, where λ is the failure rate.

Then, average probability of failure per hour (according to ISO 26262-5), over operational life time (T_{lifetime}) is:

$$\frac{\text{Prob}(T \le T_{\text{lifetime}})}{T_{\text{lifetime}}} = \frac{1-e^{-\lambda \times T_{\text{lifetime}}}}{T_{\text{lifetime}}}$$

If $\lambda \times T_{\text{lifetime}} << 1$, then $1-e^{-\lambda \times T_{\text{lifetime}}} \cong \lambda \times T_{\text{lifetime}}$.

Therefore, average probability of failure per hour over the operational lifetime of the item can be simplified as:

$$\frac{\text{Prob}(T \le T_{\text{lifetime}})}{T_{\text{lifetime}}} = \lambda$$

Consequently, regarding the single point fault, average probability of failure per hour over the operational lifetime of the item is equivalent with failure rate. Average probability of failure per hour over the operational lifetime of the item by residual fault is similarly shown by considering the fraction of fault detected and handled by safety mechanisms.

If multiple-point faults are considered, exposure durations are taken into account as stated in ISO 26262-5:2018, 9.4.2.4. If the multiple-point fault is detected or perceived, then exposure duration is equivalent to the time span to maintain service after the occurrence of the fault. In this case, average probability per hour over the operation time depends on the failure rate and exposure duration and is independent of vehicle lifetime. If the multiple-point fault remains latent, then average probability of failure per hour over the operational lifetime depends on the operational lifetime. These are mathematically represented in the formula shown in 8.3.2.4.

EXAMPLE A system has two independent components, A and B, providing system redundancy. Both components must fail to violate the safety goal. The failure of neither component by itself is detected and reported, i.e. all faults of components A and B are latent dual-point faults. Both have exponential failure distributions:

$$F_A(t) = \int_0^t \lambda_A \times e^{-\lambda_A \times \tau} d\tau = 1 - e^{-\lambda_A \times t}, \text{ and}$$

$$F_B(t) = \int_0^t \lambda_B \times e^{-\lambda_B \times \tau} d\tau = 1 - e^{-\lambda_B \times t}$$

Where λ_A and λ_B are failure rates of component A and B respectively, with $\lambda_A = \lambda_B = 3e{-}6/h$.

Figure 20 shows the failure distribution $F(t)$ of a multiple-point failure of component A and component B for lifetimes up to 10 000 h (black solid line). The time derivative $f(t)$ of this failure distribution, divided by the corresponding reliability $R(t)$, is the instantaneous failure rate of this multiple-point failure. Since $f(t)$ increases with time while $R(t)$ remains very close to 1,0, this instantaneous failure rate increases with time.

The slope of the dotted and dashed grey lines represents the average probability per hour at 5 000 h and 8 000 h respectively. Note that average probability per hour is **not** equivalent to a constant failure rate. The slope of the dot-dash and solid grey lines represents the failure density function $f(t)$ at 5 000 h and 8 000 h respectively. This failure density is very close to the instantaneous failure rate in this example since $R(t)$ is very close to 1,0.

Assuming $\lambda_A \times T_{\text{lifetime}}$ and $\lambda_B \times T_{\text{lifetime}}$ are small, the probability of violating the safety goal within the vehicle life time [$F(t)$] is approximately:

$$F\left(T_{\text{lifetime}}\right) \approx \left(\lambda_A \times T_{\text{lifetime}}\right) \times \left(\lambda_B \times T_{\text{lifetime}}\right)$$

Consequently, average probability per hour over the operational lifetime of the item is:

$$F\left(T_{\text{lifetime}}\right) / T_{\text{lifetime}} \approx \lambda_A \times \lambda_B \times T_{\text{lifetime}}$$

The corresponding instantaneous failure rate is approximately:

$$\lambda\left(T_{\text{lifetime}}\right) = \frac{\frac{dF}{dt}\left(T_{\text{lifetime}}\right)}{1 - F\left(T_{\text{lifetime}}\right)} = \frac{f\left(T_{\text{lifetime}}\right)}{R\left(T_{\text{lifetime}}\right)} \approx 2 \times \lambda_A \times \lambda_B \times T_{\text{lifetime}}$$

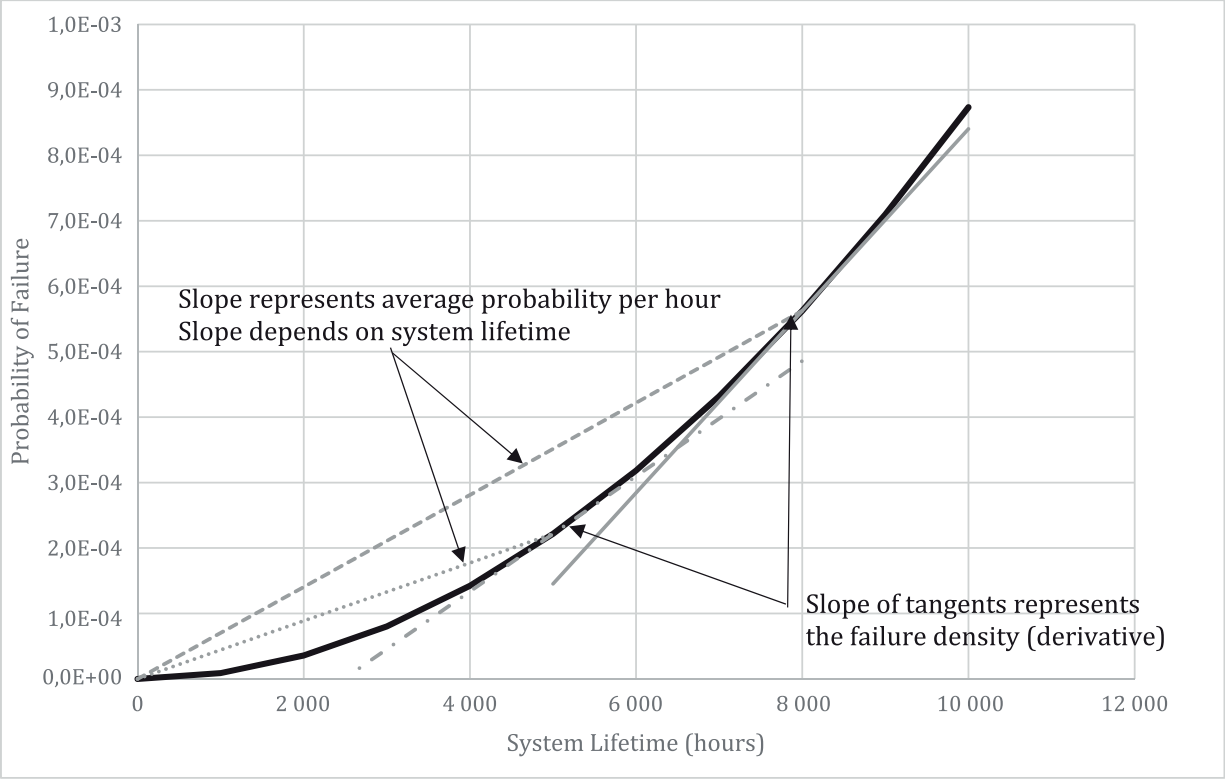


Figure 20 — Example Failure Rate Distribution

In summary, the example illustrates the main intention of the PMHF calculation, as defined in ISO 26262-5. The probability of violation of the safety goal over the operational life time of the item is determined and divided by the operational life time of the item. The result is expressed as average probability per hour, typical for this kind of evaluation (i.e. same unit is used in IEC 61508) and is not a failure rate.

The system failure rate is a time dependent function that may become very complicated to evaluate, especially when redundancy is included in the system design. The example illustrates this complexity on the simplest redundant architecture. In addition, the interpretation of such a function is not obvious for non-specialists, while a probability is an easily understandable quantity.

NOTE ISO 26262-5 gives no guidance on how to determine operational lifetime. It is the responsibility of the item owner to specify the operational lifetime to be used.

9 Safety Element out of Context

9.1 Safety Element out of Context development

The automotive industry develops generic elements for different applications and for different customers. These generic elements can be developed independently by different organizations. In such cases assumptions are made about the requirements and the design; including the safety requirements that are allocated to the element by higher design levels and on the design external to the element.

Such elements can be developed by treating these as Safety Elements out of Context (SEooC). An SEooC is a safety-related element which is not developed for a specific item. This means, it is not developed in the context of a particular vehicle.

An SEooC can be a system, a combination of systems, a subsystem, a software component, a hardware component or a part. Examples of SEooC include system controllers, ECUs, microcontrollers, software implementing a communication protocol or an AUTOSAR software component.

An SEooC cannot be an item as the development of an item always requires the context of a vehicle intended for series production. In the case where the SEooC is a system, this system is not developed in the context of a vehicle and therefore it is not an item.

SEooCs differ from qualified software components described in ISO 26262-8:2018, Clause 12 and evaluated hardware elements described in ISO 26262-8:2018, Clause 13:

— An SEooC is developed, based on assumptions, in accordance with the ISO 26262 series of standards. It is intended to be used in multiple different items when the validity of its assumptions can be established during integration of the SEooC.

— Qualification of software components and evaluation of hardware elements address the use of pre-existing software components or hardware elements for an item developed under the ISO 26262 series of standards. These are not necessarily designed for reusability nor developed under the ISO 26262 series of standards.

For software development, Table 4 describes the intended use of qualification, safety element out of context and the proven in use argument for different software components. For hardware development, an equivalent table could be constructed.

Table 4 — Categorization of the software components

Categorization of Software Component	ISO 26262-6 in context of an item	ISO 26262-8:2018, Clause 12 Qualification of SW Component	ISO 26262-6 as Safety Element out of Context	ISO 26262-8:2018, Clause 14 Proven in use argument
Newly developed	Suitable	Not suitable	Suitable	Not suitable
Re-use with change	Suitable	Not suitable	Suitable	Suitable[a]
Re-use without change	Not suitable	Suitable	Suitable (if originally developed as SEooC)	Suitable
[a] See ISO 26262-8:2018, 14.4.4.				

When developing an SEooC, applicable safety activities are tailored as described in ISO 26262-2:2018, 6.4.5.7. Such tailoring for the SEooC development does not imply that any step of the safety lifecycle can be omitted. In case certain steps are deferred during the SEooC development they are completed during the item development.

The ASIL capability of an SEooC designates the capability of the SEooC to comply with assumed safety requirements assigned with a given ASIL. Consequently, it defines the requirements of the ISO 26262 series of standards that are applied for the development of this SEooC.

An SEooC is thus developed based on assumptions; on an intended functionality and use context which includes external interfaces. These assumptions are set up in a way that addresses a superset of items, so that the SEooC can be used later in multiple different, but similar, items.

The validity of these assumptions is established in the context of the actual item while integrating the SEooC.

An item may contain multiple SEooCs, with SEooCs interfacing directly to each other. In this case the validity of the assumptions of one SEooC is established considering the interfacing SEooC.

In a case where the validity of the assumptions made during the SEooC development cannot be established during its integration into the item, either a change to the SEooC or to the item can be made as described in ISO 26262-8:2018, Clause 8.

9.2 Use cases

9.2.1 General

The development of an SEooC involves making assumptions on the prerequisites of the corresponding phase in the product development, e.g. for a software component, which is a part of the software architectural design, the corresponding phase is ISO 26262-6:2018, Clause 7. It is not necessary to make assumptions on all prerequisites.

Figure 21 shows relationship between assumptions and SEooC development. The development of an SEooC can start at a certain hierarchical-level of requirements and design. Each individual requirement or design prerequisite is pre-determined in the status "assumed".

The correct implementation of the requirements for the SEooC (derived from the assumed high-level requirements and assumptions on the design external to the SEooC) will be verified during the SEooC development.

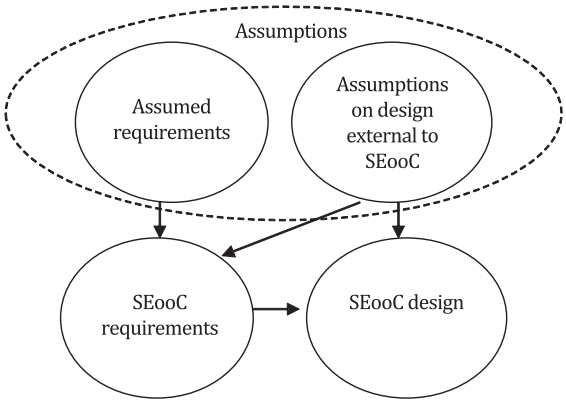


Figure 21 — Relationship between assumptions and SEooC development

The validation of these requirements and assumptions are then established during the development of the item.

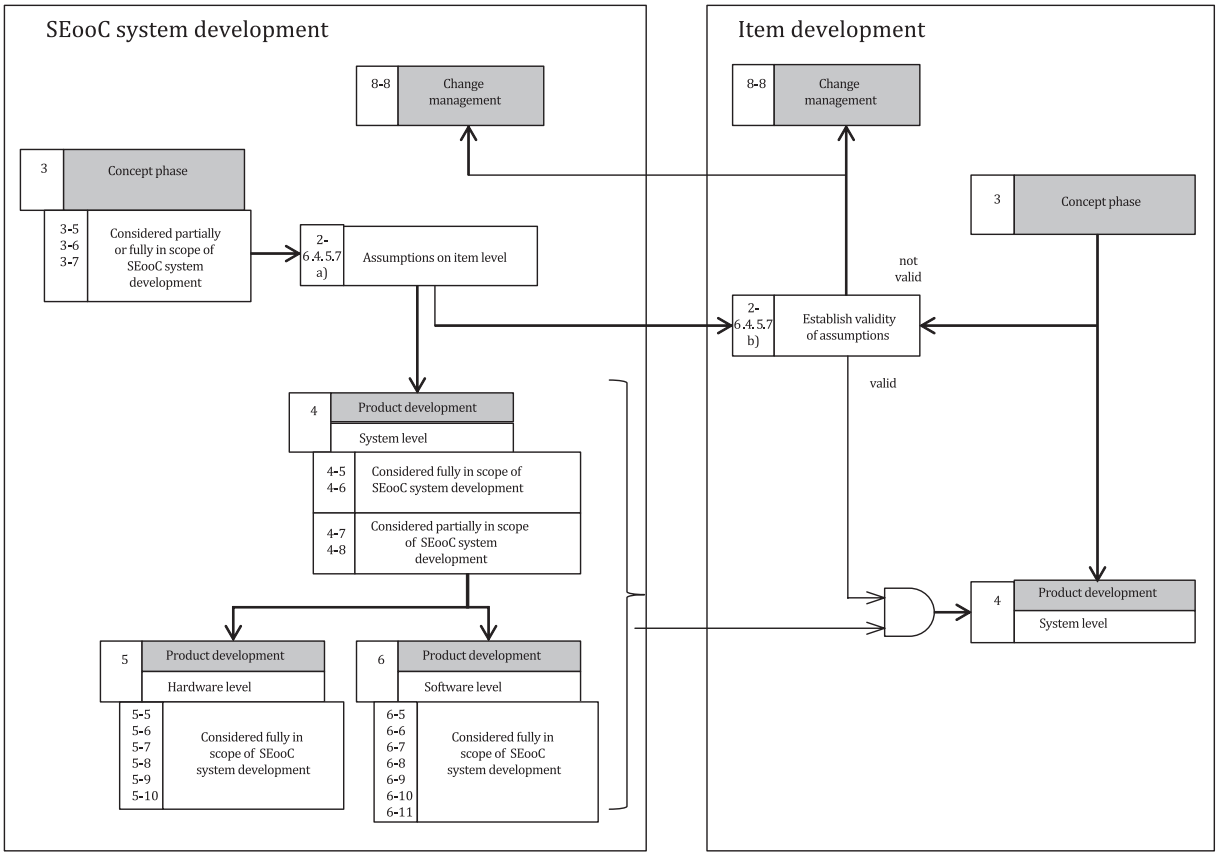
Similarly, verification activities demonstrate that a developed SEooC, at any level, is consistent with the requirements in the context where it is used. For example, when a software component, developed out of context, is used, the verification of the software specification can demonstrate that the requirements in the software architectural design specification are met. This verification report can be produced when development of the SEooC is finished and the item development reaches the phase where requirements on the safety element are formulated.

Some typical examples of SEooC are given below; namely a system, a hardware component and a software component.

9.2.2 Development of a system as a Safety Element out of Context example

This section is intended to show how the tailoring of the SEooC concept is applied to a new E/E system which can be integrated by different vehicle manufacturers.

For the purpose of this example, the system includes functionality to both activate a function under certain vehicle conditions and to allow the deactivation of the function on proper driver requests. The process flow is given in Figure 22.



NOTE 1 Some additional tailoring of the requirements can be necessary depending on the exact nature of the SEooC.

NOTE 2 Depending on the exact nature of the SEooC, some requirements of ISO 26262-3 and ISO 26262-4 cannot be applicable, and therefore only partial consideration is made.

NOTE 3 Although all the clauses of the ISO 26262 series of standards are not shown, this does not imply that they are not applicable.

Figure 22 — SEooC system development

Step 1a — Definition of the scope of SEooC

Based on assumptions, the SEooC developer defines the purpose, functionalities and external interfaces of the SEooC.

Examples of such assumptions on the scope of the SEooC can be:

— The system is designed for vehicles with a gross mass up to 1 800 kg.

— The system is designed for front-wheel driven vehicles.

— The system is designed for maximum road slope of 32 %.

— The system has interfaces with other external systems to get the required vehicle information.

— Functional requirements:

  — The system activates the function when requested by the driver under certain vehicle conditions;

  — The system deactivates the function when requested by the driver.

Step 1b — Assumptions on safety requirements for the SEooC

The development of an SEooC makes assumptions about the item definition, the safety goals of the item, and the corresponding functional safety requirements related to the SEooC functionality in order to identify the technical safety requirements of the SEooC.

Examples of assumptions on the functional safety requirements allocated to the SEooC can be:

— The system does not activate the function at high vehicle speeds (ASIL x).

— The system does not deactivate the functionality when the driver request is not detected (ASIL y).

To achieve the assumed safety goals, specific assumptions on the context are defined.

Examples of assumptions on the context of the SEooC can be:

— An external source will provide information at the requested ASIL enabling the system to detect the proper vehicle condition (ASIL x).

— An external source will provide information about the driver request at the requested ASIL (ASIL y).

Step 2 — Development of the SEooC

When the technical safety requirements have been derived from the assumed functional safety requirements of the item, the SEooC is developed following the requirements of the ISO 26262 series of standards.

Step 3 — Work products

At the end of the SEooC development, the work products that show that the derived technical safety requirements are fulfilled, are made available. All necessary information from the work products is then provided to the item integrator, including SEooC safety requirements and the assumptions made on the context.

Step 4 — Integration of the SEooC into the item

During item development, the safety goals and the functional safety requirements are specified. The functional safety requirements of the item are matched with the functional safety requirements assumed for the SEooC to establish their validity.

In the case of an SEooC assumption mismatch, a change management activity, beginning with an impact analysis, is conducted as described by ISO 26262-8:2018, Clause 8. Potential outcomes include:

— the difference can be deemed to be acceptable with regard to the achievement of the safety goal, and no action is taken.

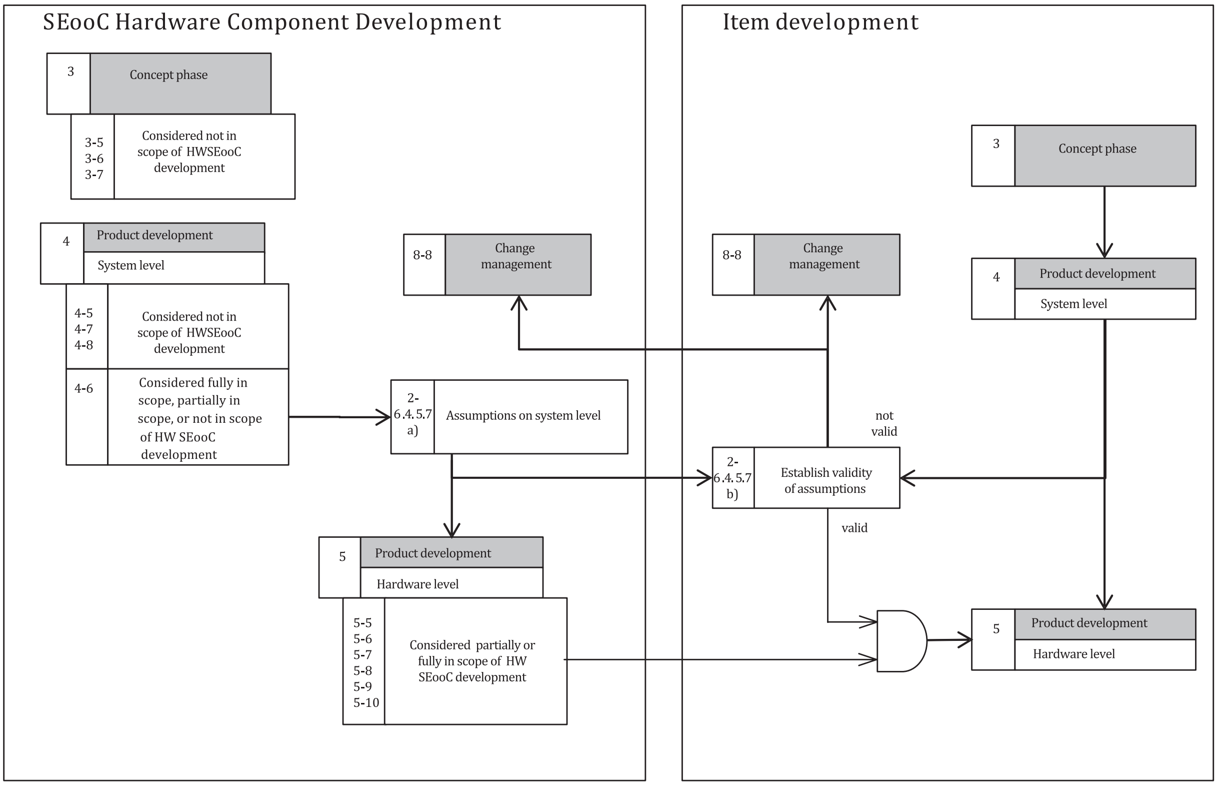
— the difference can be deemed to impact the achievement of the safety goal and a change can be necessary to either the item definition or the functional safety concept.

— the difference can be deemed to impact the safety goal and a change is required to the SEooC component (including possibly a change of component).

9.2.3 Development of a hardware component as a Safety Element out of Context example

9.2.3.1 General

This section uses a microcontroller (MCU) as an example hardware component SEooC. The process flow is given in Figure 23.



NOTE 1 Some additional tailoring of the requirements can be necessary depending on the exact nature of the SEooC, e.g. to adapt target values for the probability to violate a safety goal due to random hardware failure.

NOTE 2 Depending on the exact nature of the SEooC, some requirements of ISO 26262-5 are not applicable, and therefore only partial consideration is made.

NOTE 3 Although all the clauses of the ISO 26262 series of standards are not shown, this does not imply that they are not applicable.

Figure 23 — SEooC hardware component development

9.2.3.2 Step 1 — Assumptions on system level

The development of an MCU (see Figure 23) as an SEooC starts (step 1) with an assumption of the system level attributes and requirements as per ISO 26262-2:2018, 6.4.5.7.

This stage can be broken down into two sub-steps (1a and 1b) based on the analysis of some reference applications. The requirements are assumed with respect to the pre-requisites for HW product development (ISO 26262-5:2018, Table A.1); examples follow.

9.2.3.3 Step 1a — Assumptions on technical safety requirements

Below are some example assumed technical safety requirements created for an MCU.

Assumptions on technical safety requirements (step 1a):

a) Failures of the CPU instruction memory are mitigated by safety mechanism(s) in hardware with at least the target value (e.g. 90 %) assigned for the single-point fault metric at the HW part level (might also be expressed in terms of required DC).

b) The contribution of the MCU to the total probability of violation of a safety goal is no more than 10 % of the indicated probability for the relevant ASIL.

c) To achieve a safe state, the MCU drives all I/O outputs to a low state when reset is asserted.

d) Any safety mechanisms implemented related to the processing function completes in less than 10 milliseconds (assigned portion of the fault handling time interval on the appropriate level within the system architecture).

e) A memory protection unit is present to provide the possibility of separating software tasks with different ASILs.

ASIL capability is established at this step.

9.2.3.4 Step 1b — Assumptions on system level design

Some examples of system level design assumptions, external to the SEooC:

a) The system will implement a safety mechanism on the power supply to the MCU to detect over voltage and under voltage failure modes.

b) The system will implement a windowed watchdog safety mechanism external to the MCU to detect either clocking or program sequence failures of the MCU.

c) A software test will be implemented to detect latent faults in the EDC safety mechanism of the MCU.

d) A SW-based test is executed at key-on to verify the absence of latent faults in the logical monitoring of the program sequence of the CPU.

e) Debug interfaces of the MCU are not used during safety-related operation. Therefore, any faults in the debug logic will be considered safe faults.

9.2.3.5 Step 2 — Execution of hardware development

On the basis of these decisions (assumed technical safety requirements and assumptions related to the design external to the SEooC), the SEooC is developed (step 2) as written in ISO 26262-5 and each applicable work product is prepared. For example, the evaluation of safety goal violations due to random HW failures (see work product written in ISO 26262-5:2018, 9.5.1) is done considering the SEooC assumptions, including any budget for FIT rate found in the assumed technical safety requirements. On the basis of the SEooC assumptions, the safety analyses and the analysis of dependent failures internal to the MCU are performed referring to ISO 26262-9.

9.2.3.6 Step 3 — Work products

At the end of the MCU product development (step 3), the necessary information from the work products is provided to the system integrator. This includes the following documentation: assumed requirements, assumptions related to the design external to the SEooC and applicable work products of the ISO 26262 series of standards (for example, the report on the probability of a violation of a safety goal due to random HW failure).

9.2.3.7 Step 4 — Integration of the SEooC into the system

When the MCU developed as an SEooC is considered in the context of the item HW product development phase, the validity of all SEooC assumptions including SEooC assumed technical safety requirements

and the assumptions related to the design external to the SEooC are established (step 4). It is plausible that mismatches between SEooC assumptions and system requirements will occur. For example, the item developer could decide not to implement an assumed external component. As a consequence, the evaluation of safety goal violations due to random HW failures done by the SEooC developer might no longer be consistent with the item.

In the case of an SEooC assumption mismatch, a change management activity beginning with impact analysis is conducted as written in ISO 26262-8:2018, Clause 8. Potential outcomes include:

— The difference can be deemed to be acceptable with regard to the achievement of the safety goal, and no action is taken.

— The difference can be deemed to impact the achievement of the safety goal and a change can be necessary to the either functional safety concept or the technical safety requirements.

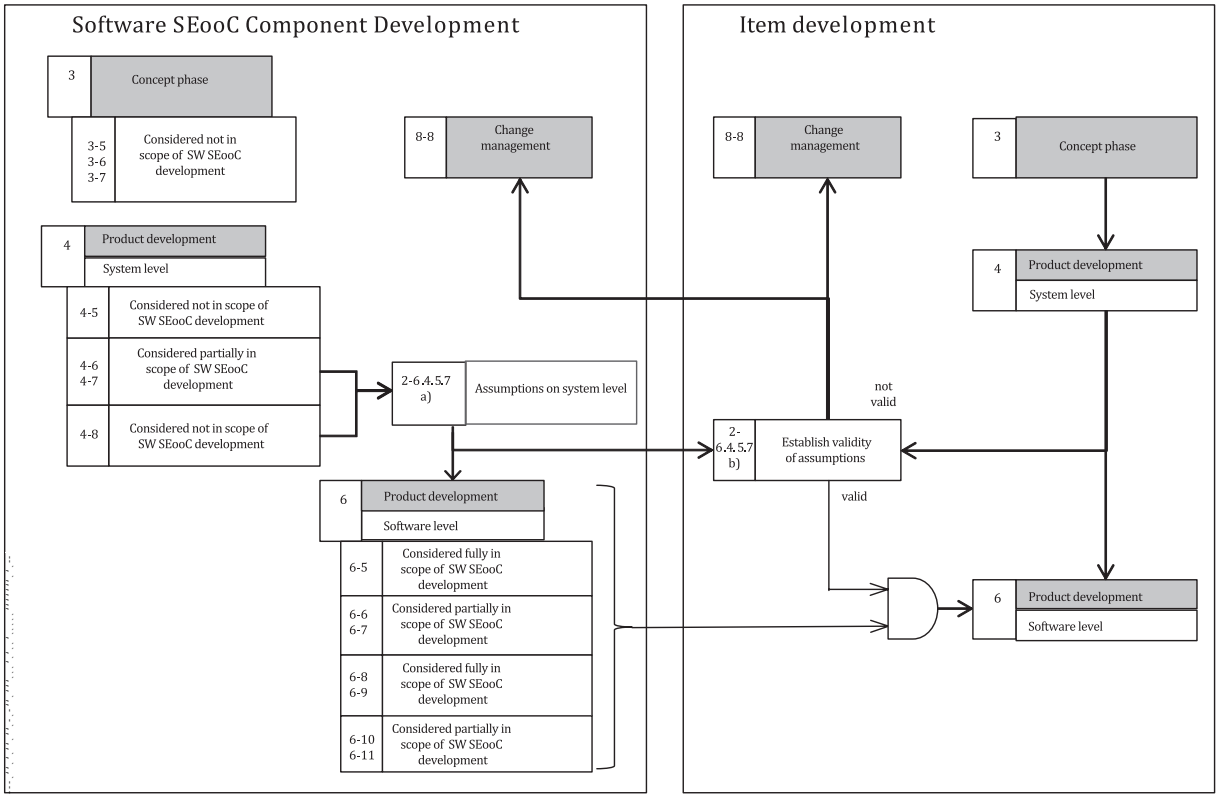
— The difference can be deemed to impact the achievement of the safety goal and a change is required to the SEooC component (including possibly a change of component).

— The difference can be deemed to impact the achievement of the safety goal, and therefore safety metrics are recalculated, but the recalculated metrics show that the design meets the system targets, so no change is necessary.

9.2.4 Development of a software component as a Safety Element out of Context example

9.2.4.1 General

This clause illustrates the different steps of the application of the SEooC concept to a new medium/low level software component. The process flow is given in Figure 24.



NOTE 1 Some additional tailoring of the requirements can be necessary depending on the exact nature of the SEooC.

NOTE 2 Depending on the exact nature of the SEooC, some requirements of ISO 26262-6 are not applicable, and therefore only partial consideration is made.

NOTE 3 Although all the clauses of the ISO 26262 series of standards are not shown, this does not imply that they are not applicable.

Figure 24 — SEooC software component development

9.2.4.2 Step 1a — Assumptions on the scope of the software component as an SEooC

This step is intended to state the relevant assumptions regarding the purpose of the software component, its boundaries, its target environment, its functionalities and its properties.

Examples of such assumptions include:

— The software component is integrated into a given software layered architecture.

— Any potential interference caused by the software component is detected and handled by its environment.

— The software component provides the functions as specified in the assumed software functional requirements.

9.2.4.3 Step 1b — Assumptions on the safety requirements of the software component

Step 1b is intended to make assumptions on higher level safety requirements that potentially impact the software component in order to derive its software safety requirements. For example, if a given

set of data calculated by the software component is assumed to be of high integrity (ASIL x), then the resulting software safety requirements allocated to the SEooC can be:

— The software component detects any corruption on the input data which can violate safety goals (ASIL x);

— The software component signals the error conditions to be notified based on the assumed technical safety requirements (ASIL x);

— A default value is returned with a fault status for any error condition detected (ASIL x); and

— The software component returns the following results coded with CRC and a status (ASIL x).

9.2.4.4 Step 2 — Development of the software component

Once the necessary assumptions on the software component are explicitly stated, the SEooC is developed in accordance with the requirements of ISO 26262-6 corresponding to its ASIL capability (ASIL x in this example). All applicable work products are made available for further integration in different contexts, including the work products related to the verification of the assumed software safety requirements.

9.2.4.5 Step 3 — Integration of the software component in a new particular context

Before the software component is integrated with other software components in a new particular context, the validity of all the assumptions made on this SEooC are checked with regard to this context. This includes the assumed software safety requirements with their ASIL capability and all the assumptions made on the purpose, boundaries, target environment, functionalities and properties of the software component (see 9.2.4.2 and 9.2.4.3).

In the case where some assumptions regarding the software component do not fit with this new context, an impact analysis is initiated in accordance with ISO 26262-8:2018, Clause 8. Potential outcomes of the impact analysis include:

— The discrepancies are acceptable with regard to the achievement of the safety requirements applicable at the software architectural design level, and no further action is taken.

— The discrepancies impact the achievement of the safety requirements applicable at the software architectural design level and a change can be necessary to these requirements in accordance with ISO 26262-8:2018, Clause 8.

— The discrepancies impact the achievement of the safety requirements applicable at the software architectural design level and a change is required to the SEooC component (including possibly a change of component) in accordance with ISO 26262-8:2018, Clause 8.

NOTE In the case where the integration of a software component in a particular software architectural design results in the coexistence of software safety-related elements that have different ASILs assigned, the criteria for coexistence of elements are fulfilled as described in ISO 26262-9:2018, Clause 6, or alternatively the elements with lower ASILs are upgraded to the higher ASIL.

10 An example of proven in use argument

10.1 General

The item and its requirements described in this clause are an example. The safety goal, its ASIL and the following requirements are given to illustrate the proven in use argument defined in ISO 26262-8:2018, Clause 14 (proven in use argument). This example does not reflect what the application of the ISO 26262 series of standards on a similar real-life example would be.

10.2 Item definition and definition of the proven in use candidate

A vehicle manufacturer wants to integrate new functionality into a new vehicle. For the purpose of this example, the item implementing this functionality is composed of sensors, one ECU that includes the complete hardware and software necessary to implement the functionality, and one actuator.

The incorrect activation of the functionality is ranked ASIL C by the vehicle manufacturer. The corresponding safety goal is derived into an ASIL C functional safety requirement allocated to the ECU.

The supplier of the ECU proposes to carry over an existing ECU already in the field.

The differences between the previous use of the ECU and its intended use in the new application are analysed. The analysis shows that the software will be modified to implement the new functionality by changing calibration data, but the ECU hardware can be carried over without modification. The supplier intends to substitute the demonstration of compliance to requirements of ISO 26262-5 by a proven in use argument for the hardware of the ECU. The hardware of the ECU is therefore the proven in use candidate.

10.3 Change analysis

To establish a proven in use credit, the supplier performs a change analysis of the proven in use candidate.

This analysis shows that no change that could have an impact on the safety behaviour of the proven in use candidate has been introduced since the beginning of its production.

Moreover, the analysis shows that the differences between the previous use of the proven in use candidate and its intended use have no safety impact:

— the candidate's boundary is within the specification limits;

— the previous integration environment requires the same technical behaviour; and

— the cause and effects at the boundary of the candidate are the same in the previous and future integration environments.

10.4 Target values for proven in use

To establish the validity of the proven in use argument, the supplier estimates the number of cumulated hours the proven in use candidate has been in the field. The supplier also analyses the field data from the service period for any safety-related event, i.e. any reported event that would potentially cause, or contribute to, the violation of a safety goal or a safety requirement regarding the intended usage of the candidate in the new item.

The estimation of the duration of the service history is performed, based on the number of produced vehicles embedding the proven in use candidate, together with their production date, and data on the typical usage of a vehicle in this segment of the market (number of driving hours per year).

The service history is based on the field return of the different vehicles embedding the proven in use candidate:

— Warranty claims;

— In-the-field defects analyses; or

— Return of defective parts from the vehicle manufacturers.

At the date of the initiation of the hardware development of the item, these analyses show that no safety-related event has occurred in the field. The total cumulated driving hours are estimated to be less than the target for the definite proven in use status for an ASIL C, but meets the interim service period as defined in ISO 26262-8:2018, 14.4.5.2.5.

The conclusion is then as follows:

— The development of the item can carry on taking credit that the hardware of the ECU is provisionally anticipated to be proven in use.

— The field observation continues to obtain a definite proven in use status (see ISO 26262-8:2018, 14.4.5.2.5 and ISO 26262-8:2018, 14.4.5.2.6)

11 Concerning ASIL decomposition

11.1 Objective of ASIL decomposition

The objective of ASIL decomposition is to comply with the safety goal by using multiple sufficiently independent elements with respect to systematic faults.

11.2 Description of ASIL decomposition

ASIL decomposition refers to the allocation of redundant safety requirements to sufficiently independent elements of the item. Redundancy in this context does not necessarily imply classical modular redundancy (see ISO 26262-1:2018, 3.122).

EXAMPLE The main processor of an ECU can be monitored by a redundant monitoring processor, both of which are independently capable of initiating a defined safe state, even if the monitoring processor is not able to fulfil the functional requirements allocated to the ECU.

ASIL decomposition can only be understood in the context of systematic failures, that is, the methods and measures applied to reduce the likelihood of these failures. The requirements on the evaluation of the hardware architectural metrics and the evaluation of safety goal violations due to random hardware failures will remain unchanged by ASIL decomposition (See ISO 26262-9:2018, 5.4.5),

EXAMPLE In the case of an ASIL B(D) decomposition, the ASIL D target for the evaluation of the hardware architectural metrics is not decomposed into separate ASIL B targets for each HW element. As written in ISO 26262-5:2018, 8.2, target values can be assigned to hardware elements, but those targets are assigned case-by-case based on an analysis started at the level of the whole hardware of the item. The target metric according to the safety goal applies at the item level.

In such a decomposed architecture, the safety requirement before decomposition is only violated if both elements simultaneously violate their safety requirements resulting from the decomposition.

The possible decompositions in the ISO 26262 series of standards are described in ISO 26262-9:2018, Clause 5.

11.3 An example of ASIL decomposition

11.3.1 General

The item and its requirements described in this clause are examples. The safety goal, its ASIL, and the following requirements are only designed to illustrate the ASIL decomposition process. This example does not reflect what the application of the ISO 26262 series of standards on a similar real-life example would be.

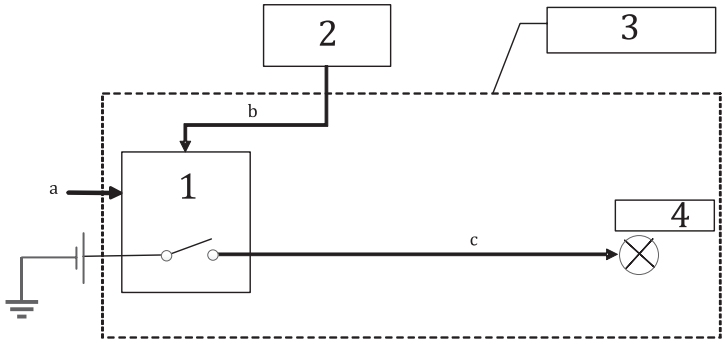
11.3.2 Item definition

Consider the example of a system with an actuator that is triggered on demand by the driver using a dashboard switch (see Figure 25). For the purpose of this example, the actuator provides a comfort function if the vehicle is at zero speed, but can cause hazards if activated above 15 km/h.

For the purpose of this example, the initial architecture of the item is as follows:

— The dashboard switch input is read by a dedicated ECU (referred to as "Actuator Control ECU (AC ECU)" in this example), which powers the actuator through a dedicated power line.

— The vehicle equipped with the item is also fitted with an ECU which is able to provide the vehicle speed. For the purpose of this example, the ability of this ECU to provide the information that the vehicle speed is greater than 15 km/h is assumed to be compliant with ASIL C requirements. This ECU is referred to as "VS ECU" in this section.



Key

1	AC ECU	a	Driver’s request.
2	VS ECU	b	Vehicle speed.
3	item boundary	c	Command to the actuator.
4	actuator		

Figure 25 — Item boundary

11.3.3 Hazard analysis and risk assessment

The hazardous event considered in the analysis is the activation of the actuator while driving at a speed above 15 km/h, with or without a driver request.

For the purpose of the example, the ASIL associated to this hazardous event is classified as ASIL C.

11.3.4 Associated safety goal

Safety Goal 1: Avoid activating the actuator while the vehicle speed is greater than 15 km/h: ASIL C

11.3.5 System architectural design

The following lists the purpose of the initial architectural elements:

— The VS ECU provides the Actuator Control ECU (AC ECU) with the vehicle speed.

— The AC ECU monitors the driver's requests, tests if the vehicle speed is less than or equal to 15 km/h, and if so commands the actuator.

— The actuator is activated when it is powered.

11.3.6 Functional safety concept

11.3.6.1 General

This example of a Functional Safety Concept is only being used as an illustration of the ASIL decomposition. It is not intended to be exhaustive and does not include all the functional safety requirements.

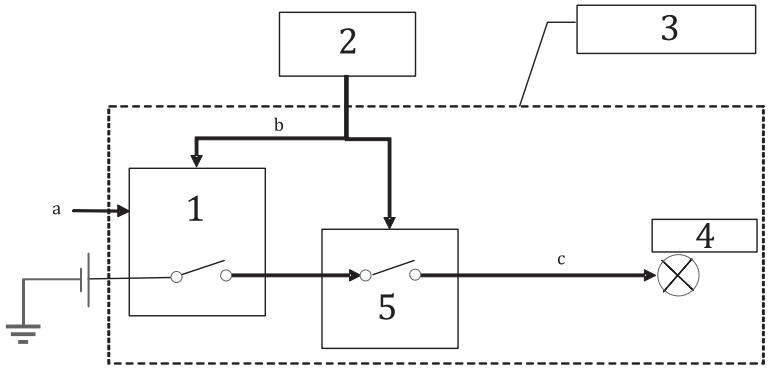
— Requirement A 1: The VS ECU sends the accurate vehicle speed information to the AC ECU. → ASIL C

— Requirement A 2: The AC ECU does not power the actuator if the vehicle speed is greater than 15 km/h. → ASIL C

— Requirement A 3: The actuator is activated only when powered by the AC ECU. → ASIL C

11.3.6.2 Evolved safety concept of the item

The developers can choose to introduce a redundant element, here a Redundant Switch, as illustrated in Figure 26. By introducing this redundant element, the AC ECU is developed with an ASIL that is equal to or lower than ASIL C, in accordance with the results of an ASIL decomposition.



Key

1	AC ECU	5	redundant switch
2	VS ECU	a	Driver's request.
3	item boundary	b	Vehicle speed.
4	actuator	c	Command to the actuator.

Figure 26 — Second iteration on the item design

Purpose of these elements (evolved architecture):

— The VS ECU control unit provides the AC ECU with the vehicle speed.

— The AC ECU monitors the driver's requests, tests if the vehicle speed is less than or equal to 15 km/h, and if so commands the actuator.

— The Redundant Switch is located on the power line between the AC ECU and the actuator. It switches on if the speed is less than or equal to 15 km/h, and off whenever the speed is greater than 15 km/h. It does this regardless of the state of the power line (its power supply is independent).

— The actuator operates only when it is powered.

Functional safety requirements:

— Requirement B 1: the VS ECU sends accurate vehicle speed information to the AC ECU. → ASIL C

— Alternatively: the incorrect transmission that vehicle speed is less than or equal to 15 km/h is prevented. → ASIL C

— Requirement B 2: the AC ECU does not power the actuator if the vehicle speed is greater than 15 km/h. → ASIL X(C) (see Table 5)

— Requirement B 3: the VS ECU sends accurate vehicle speed information to the Redundant Switch. → ASIL C

— Requirement B 4: The Redundant Switch is in an open state if the vehicle speed is greater than 15 km/h. → ASIL Y(C) (see Table 5)

— Requirement B 5: The actuator operates only when powered by the AC ECU and the Redundant Switch is closed. → ASIL C

To permit an ASIL decomposition, the developers add an independency requirement if deemed necessary:

— Requirement B 6: Sufficient independence of the AC ECU and the Redundant Switch is shown. → ASIL C

The original requirement A 2 has been replaced by the redundant requirements B 2 and B 4, both of which comply with the safety goal, and therefore ASIL decomposition can be applied.

Table 5 — Possible decompositions

	Requirement B 2: ASIL X(C)	Requirement B 4: ASIL Y(C)
Possibility 1	ASIL C(C) requirements	QM(C) requirements
Possibility 2	ASIL B(C) requirements	ASIL A(C) requirements
Possibility 3	ASIL A(C) requirements	ASIL B(C) requirements
Possibility 4	QM(C) requirements	ASIL C(C) requirements

12 Guidance for system development with safety-related availability requirements

12.1 Introduction

For many E/E systems, the loss of functionality cannot lead to a hazard. Therefore, the safe state can be achieved by switching off the functionality in case of a malfunction within the system. However, in some cases the HARA can show that the loss of a certain functionality can lead to a hazardous event. This can lead to a safety goal specifying a safety-related availability requirement.

The term "fault tolerance" is used in a restricted sense within this clause. Within this clause the term "fault tolerance" is only used in the context where the specified functionality is the intended functionality or a subset of the intended functionality that is to be provided in the presence of one or more faults ((see ISO 26262-1:2018, 3.60). This clause does not address the context where the specified functionality is used to switch off the system. This clause does not address the context where a safe-state can be directly reached by switching off the functionality.

NOTE 1 There are various measures to ensure sufficient availability, including fault tolerance, fault avoidance and fault forecasting, where fault tolerance means the capability to deliver a specified functionality for at least a limited time after a fault belonging to a specified fault set has occurred. Fault avoidance means measures to reduce the occurrence of a fault and fault forecasting means the capability to detect a fault or degradation before it can lead to a failure.

NOTE 2 In case of fault tolerance not every imaginable fault can be tolerated. For clarification, the tolerable fault sets (e.g. single bit faults in case of an ECC with single error correction, double error detection capability) are specified.

12.2 Notes on concept phase when specifying fault tolerance

12.2.1 General

The following topics are considered in the concept phase when specifying fault tolerance:

a) vehicle operating states in which the availability of a functionality is safety-related;

b) faults to be tolerated;

c) prevention of a hazardous event after a fault has occurred;

d) operation after the item fault reaction;

e) safe state to be achieved after a fault has occurred;

f) ASIL decomposition of fault tolerant items; and

g) safety requirements on other items.

12.2.2 Vehicle operating states in which the availability of a functionality is safety-related

Whether the loss of the availability of an item's functionality can lead to a hazardous event or not depends on the vehicle operating state. For example, the loss of the functionality can lead to a hazardous event in a specific vehicle operating state (e.g. steering at high vehicle speeds) whereas in another vehicle operating state (e.g. steering at zero vehicle speed) it might not. If the vehicle is in a vehicle operating state in which the loss of the functionality does not lead to a hazardous event, then the availability of the functionality is considered not safety-related.

The measures to fulfil the safety related availability requirement are based on possible interactions with other item(s), the system architecture including other technologies (e.g. mechanical backup) and the results of safety analyses. If fault tolerant measures are adopted, 12.2.3 and 12.2.4 are applicable.

NOTE If the vehicle operating state is maintained by other items, possible new or changed hazards for these items are considered and the HARA is re-evaluated, if necessary.

EXAMPLE System X is an E/E system without mechanical back up. When driving at high speeds on country roads, the sudden loss of its functionality is difficult to control and can lead to a hazardous event with an ASIL rating. At very low speeds, the sudden loss of its functionality can be controlled easily by applying a different function from an available and sufficiently independent system Y, resulting in a C0 classification. So, in the vehicle operating state "item in normal operating mode at high vehicle speeds" the availability of its functionality is considered to be safety-related while in the vehicle operating state "item in normal operating mode at very low vehicle speeds" the availability of the functionality is not considered to be safety-related.

12.2.3 Prevention of hazardous events after a fault

12.2.3.1 Allowable time-span from the occurrence of a fault to completion of fault reaction

As part of the safety requirements, the maximum fault handling time interval consistent with the fault tolerant time interval of the item is specified.

12.2.3.2 Functions and performance to be maintained after fault reaction

For the case that the fault can cause the loss of the intended functionality or a subset of the intended functionality, and this loss can result in a hazardous event, the functions and the performance to be maintained after the occurrence of the fault are specified in order to transition to a safe state, transition between safe states or maintain a safe state.

NOTE Such functions and performance are provided during emergency operation or in the safe state.

EXAMPLE A hazardous event can only occur if the item output performance drops below 50 % of its specified maximum output. The item is composed of two systems each of which is capable of providing 50 % of the maximum specified output. If one system fails, the hazardous event can be prevented by switching off the faulty system while the other one maintains up to 50 % of the maximum specified output.

12.2.4 Operation after fault reaction

12.2.4.1 Emergency operation

During emergency operation, the item is still free from unreasonable risk even though the ASIL capability of the item is lower than the ASIL rating of the possible hazard. To address this situation, the operating time in this state is limited, such that it is unlikely that an additional fault occurs which leads to a violation of the safety goal.

NOTE 1 The emergency operation tolerance time interval is defined and verified from the probability of a next fault in accordance with 12.3.1.

NOTE 2 The transition to the emergency operation is defined and verified to be safe.

12.2.4.2 Safe states for fault tolerant item

In the context of fault tolerant behaviour, typically one of the following two safe states is chosen:

a) A vehicle operating state in which the specified functionality is no longer needed for safety reasons. In this case, the specified functionality is permanently switched-off and thus no longer provided until the item is repaired; or

b) The possible vehicle operating states are limited in such a way that the ASIL rating of the hazardous events which can occur in the limited vehicle operating states is equal to or lower than the ASIL capability of the remaining system. In this case the specified functionality of the limited operating states provided by the remaining system can be interpreted as an item in its own right and can be operated without time restrictions. The possible vehicle operating states will return to unlimited once the item is repaired.

   NOTE 1 The restrictions of the possible vehicle operating states can have an impact on the E, C and S parameters of the possible hazardous event.

   EXAMPLE 1 Limiting the vehicle speed can reduce the severity and can improve the controllability of the hazardous event resulting in a lower ASIL rating than without the limitation.

   NOTE 2 The exposure E in this HARA evaluation does not consider the occurrence of the fault.

   NOTE 3 This ASIL rating can be used to:

   — Restrict ASIL decompositions for items (12.2.6).

   — Specify safety requirements for individual redundant components constituting the item. This includes determination of the ASIL capability of the remaining system in case of the loss of one redundant component.

   NOTE 4 If a safety goal for an item is to maintain full or degraded functionality in the presence of a fault then the HARA can be extended to cover the functionality of the restricted vehicle operating states.

   NOTE 5 If after the occurrence of the fault, the vehicle operating states are not changed (e.g. the vehicle operation is unrestricted without warning), then the ASIL is the same as that derived from the vehicle operating states without fault (the original HARA is applicable).

   NOTE 6 The safety mechanism implementing the restrictions of the possible vehicle operating states inherits the original ASIL of the safety goal. If the safe state is reached or maintained with the support of functions of other items, these are identified as safety requirements on those items.

EXAMPLE 2 The ASIL of the safety goal of the item, in absence of faults, is "D", Controllability = C3, Severity = S3 and Exposure = E4 and if in the operating state of the degraded functionality, the controllability is improved, for example, to C2 then the ASIL requirements within this operating state is ASIL C: S3, E4, C2.

EXAMPLE 3 A by-wire system in the presence of a fault restricts vehicle operation to a low speed where most drivers can prevent a collision via another system. This could improve Controllability and can also reduce Severity in the vehicle speed restricted state.

NOTE 7 Once a safe state is reached, and the operator is notified, any repairs are the responsibility of the vehicle owner/operator as per the Vienna Convention on Road Traffic [17].

12.2.4.3 Emergency operation time interval

For a fault tolerant item, once a fault occurs the specified functionality is maintained and the item transitions to a safe state in accordance with 12.2.3.2. During the transition to a safe state, the fault reaction at vehicle level occurs (e.g. limiting vehicle speed to 30 km/h). However, before completing vehicle level fault reaction, a possible hazardous event caused by another fault occurring during the emergency operation time interval is not mitigated.

To minimize the risk, the emergency operation time interval is limited to the emergency operation tolerance time interval defined as a part of the safety concept.

NOTE 1 For determining the emergency operation tolerance time interval, the following are considered:

— physical system limitations;

— additional safety requirements for the hardware and software elements used during emergency operation, if required; and

— possibility of remaining system failing in a common way.

NOTE 2 Regarding random hardware faults, ISO 26262-5:2018, 9.4.2.4 e) can ensure the effectiveness of the hardware architecture in mitigating random hardware faults by taking the emergency operation time interval into account as an exposure duration.

NOTE 3 Ensuring that the emergency operation time interval does not exceed the emergency operation tolerance time interval is not always the responsibility of the vehicle manufacturer. This can also be the responsibility of the driver of the vehicle.

EXAMPLE One of the headlight bulbs is burned out. The failure is detected and the driver is informed about the failure. It is the responsibility of the driver to repair the headlights within a reasonable amount of time.

12.2.5 Fault tolerant item example

12.2.5.1 Introduction

An example is used to describe possible flow of events related to behaviours of fault tolerant systems. This will illustrate the application of the various fault time interval notations.

12.2.5.2 Assumptions

The HARA for the item shows that a significant loss of the specified functionality for longer than a time x can lead to a hazardous event with an ASIL rating which is dependent on vehicle speed (v_{vehicle}) at the time when the loss of the specified functionality occurs.

In this example, it is assumed, with respect to HARA, that the exposure of the operational situation does not change for v_{vehicle}. Therefore, the safety goal is formulated under the assumption of worst case conditions.

The safety goal in this example is formulated as:

— avoid significant loss of the specified functionality for longer than a time x (the FTTI) (ASIL D).

NOTE Significant loss of the specified functionality means the output performance of the function is below the minimally required performance level.

12.2.5.3 Strategies of the example

Two strategies are considered to realize the safety goal assumed in 12.2.5.2.

— Strategy 1: The item maintains the specified functionality after occurrence of a fault. The functionality is kept operating until the item is repaired. The vehicle operating state is not restricted until repair. In this case, the item is repaired within an allowable time interval (i.e. the emergency operation tolerance time interval).

— Strategy 2: The item maintains the specified functionality after occurrence of a fault. The functionality is kept to the limited vehicle operating state without time limitation. In this case, the vehicle reaches the limited vehicle operating state within the allowable time interval (i.e. the emergency operation tolerance time interval).

12.2.5.4 System architecture description of the item

The item description for the example is provided below:

— The item consists of two sufficiently independent channels, channel A and channel B.

— Channel A provides the nominal function.

— Channel B is a backup system. Its performance is greater than minimally required performance level which is the functionality necessary for a safe operation. The sufficiency of this functionality is validated according to ISO 26262-4:2018, Clause 8.

— If channel A fails leading to a significant loss of functioning capability, channel B is activated within time *x* to prevent the violation of the safety goal.

— Each one of channel A and channel B can, by itself fulfil the safety requirements on an ASIL D level as far as systematic faults are concerned.

— The combination of channel A and channel B can fulfil the safety requirements on an ASIL D level as far as random hardware faults (e.g. ≤ 1 % of the random hardware faults can lead to a significant loss of the functionality) are concerned.

NOTE Other safety goals can exist leading to additional safety requirements for the elements of the item. These are not considered within this discussion.

This example considers two strategies which differ on the capability of channel B:

Strategy 1: Repair within emergency operation tolerance time interval

— Channel B by itself does not fulfil the safety requirements for random hardware faults on an ASIL D level.

— When the loss of channel A is detected, the driver is notified and the driver is required to repair the item within the emergency operation tolerance time interval. The probability of fault occurrence during emergency operation time interval is taken into account as part of the PMHF calculation and can also be ensured by methods in 12.3.

Strategy 2: Limited operation without time restrictions

— Channel B by itself does not fulfil the safety requirements for random hardware faults on an ASIL D level. It can only fulfil the safety requirements as far as random hardware faults are concerned on an ASIL A level, i.e. it only has an ASIL A capability regarding this safety goal. It achieves this via fault prevention measures only.

— The HARA for the item in which ASIL rating depending on $v_{vehicle}$ is referred. In this example, ASILs are assumed as:

  — if the maximum vehicle speed is not limited, the rating is ASIL D;

  — if the maximum vehicle speed is limited to v_4, the rating is ASIL C;

  — if the maximum vehicle speed is limited to v_3, the rating is ASIL B;

  — if the maximum vehicle speed is limited to v_2, the rating is ASIL A;

  — if the maximum vehicle speed is limited to v_1, the rating is QM; and

  — if the maximum vehicle speed is limited to v_0, no hazard can occur.

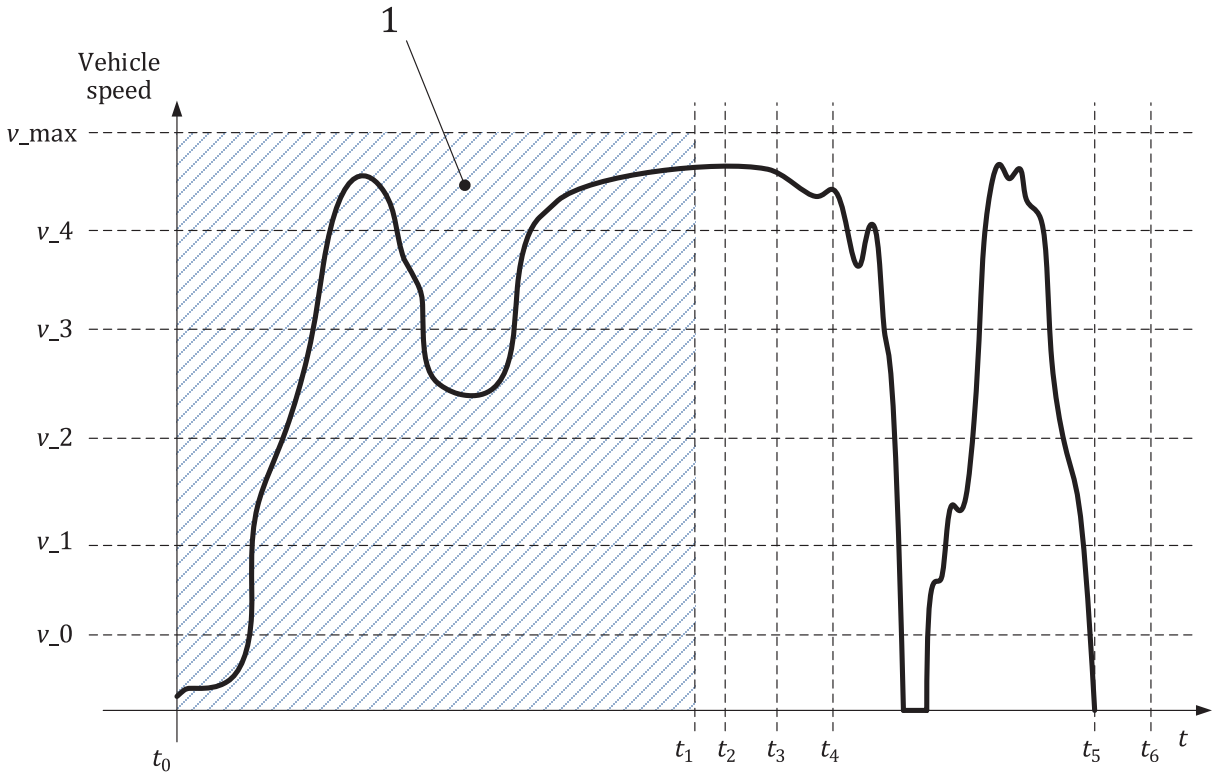
NOTE This safety goal is rated ASIL D as the possible vehicle operating states are not limited (i.e. $v_{vehicle} > v\_4$). However, for specific driving situations (e.g. $v_{vehicle} < v\_2$), either certain hazards cannot occur or the S, C and E ratings are not the same as worst case assumption. This results in an overall lower ASIL requirement for this hazard. In this example, the hazard is only rated ASIL D for higher vehicle speeds. If the vehicle speed is equal or less than v_2, then better controllability and less severity than for higher vehicle speeds are assumed and the resulting hazard from significant loss of the specified functionality is rated as an ASIL A.

— The vehicle speed is reduced to less than v_2 by other items and this function is an additional safety requirement for such item with ASIL D. This is a prerequisite for implementing Channel B in strategy 2.

12.2.5.5 Flow of events for this example

Figures 27 and 28 show examples which describe the concepts of FTTI, fault detection and fault reaction time, emergency operation and safe state in the context of a safety-related availability requirement addressed by a fault tolerant system with strategy 1 and strategy 2 respectively.

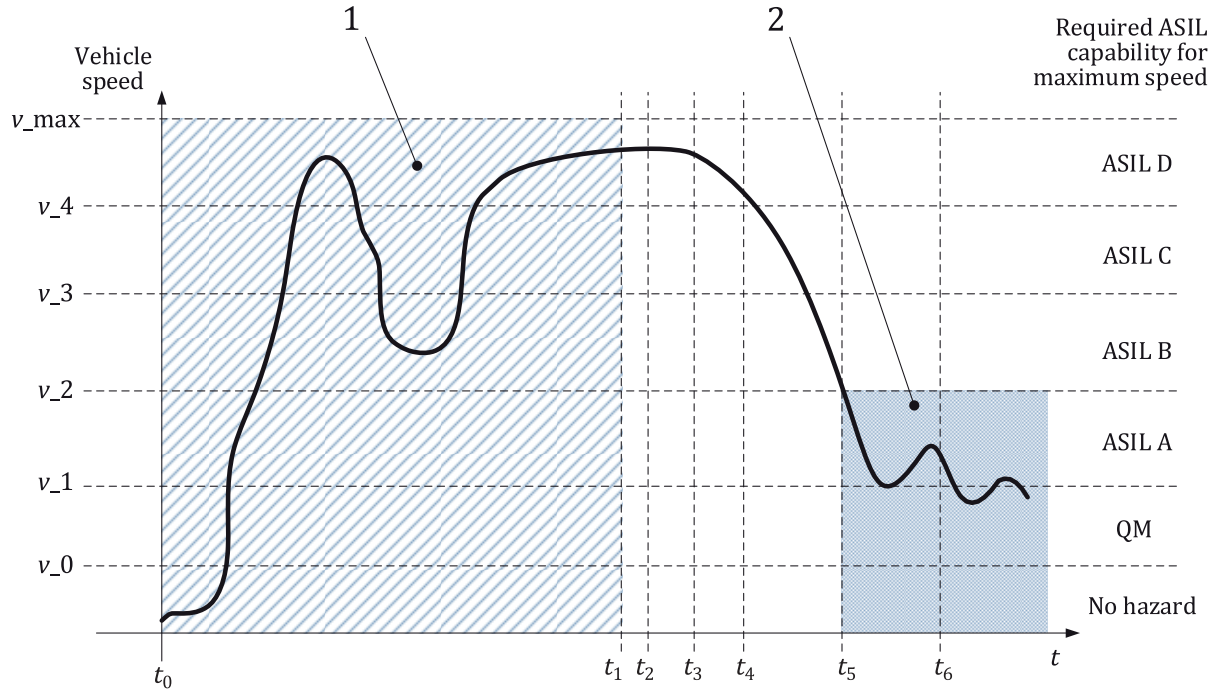
In these figures, the same time scales are used to clarify the difference between the two strategies.



Key

1 item is in normal operation mode with ASIL D capability

Figure 27 — Example vehicle speed history for strategy 1



Key

1 item is in normal operation mode with ASIL D capability

2 item is in fault mode with ASIL A capability

Figure 28 — Example vehicle speed history for strategy 2

Figure 27 and 28 events t_1 through t_6 are:

— t_1: time when a fault manifests itself as a significant loss of the specified functionality;

A fault in channel A manifests itself as a loss of channel A, which results in a significant loss of the specified functionality. The loss of the function occurs at $v_{vehicle} > v\_4$ which could result in an ASIL D rated hazard.

— t_2: time when the fault is detected;

The significant loss of the specified functionality is detected by a safety mechanism. This occurs in both strategies.

NOTE The time span between t_1 and t_2 is also referred to as the fault detection time interval.

— t_3: time when the item completes changing its operation mode, end of the fault reaction time interval;

As an error reaction, channel B is activated providing the required functionality. As the required functionality is provided within a time less than x, i.e. $t_1 < t_3 < t_1 + x$, the hazard is prevented.

In strategy 1, the occurrence of the fault is notified to the driver by means specified as part of the warning and degradation strategy.

In strategy 2, deceleration within v_2 starts at this time.

NOTE The safety requirement to reduce the vehicle speed to v_2 has an ASIL D rating. Therefore, the function to reduce and maintain the vehicle speed to v_2 needs an ASIL D capability.

— t_4: end of fault tolerant time interval;

$t_4 = t_1 + x$. The time x corresponds to the FTTI. If the significant loss of specified functionality lasts for t_4 or longer, an ASIL D rated hazardous event can occur. This occurs in both strategies.

— t_5: time when the item reaches the safe state, t_5–t_3 is defined as the emergency operation time interval;

In strategy 1, the item is repaired at this point. Before the item is repaired, vehicle operating state is not limited. Depending on the warning and degradation strategy, t_5 is reached at the end of one trip or after several drive cycles.

In strategy 2, the vehicle speed reached $v_{vehicle} < v\_2$, at this point of time. In this state, a significant loss of the specified functionality (e.g. due to failure of channel B) can only lead to an ASIL A rated hazard. Since the remaining effective safety measures support the safety goal up to an ASIL A, i.e. channel B has an ASIL A capability regarding this safety goal, the operating state of the item can be considered to be without an unreasonable level of risk.

NOTE The time between t_3 and t_5, the emergency operation time interval, can also be considered as free from unreasonable risk not due to the achieved amount of risk reduction but due to the argument that the item spends a limited time in this vehicle operating state.

— t_6: maximum allowable time to reach the safe state, t_6–t_3 is defined as the emergency operation tolerance time interval;

The allowable time span from t_3 to reaching the safe state is defined as the emergency operation tolerance time interval. The emergency operation tolerance time interval is $t_3 + y$ and is the latest point in time.

In strategy 1, t_6 is the expected time until when the item is repaired.

In strategy 2, t_6 is the target time when the limitation of the vehicle speed to $v_{vehicle} < v\_2$ can be achieved.

NOTE For items where availability is not safety-related, the item reaches the safe state within t_4.

12.2.6 ASIL decomposition of fault tolerant items

The basic idea of ASIL decomposition is that an initial safety requirement with ASIL X is decomposed into a combination of redundant safety requirements with ASIL Y1(X) and ASIL Y2(X). The target risk reduction is achieved by the combination of decomposed redundant safety requirements and is not achieved by one of them alone. This approach is also applicable to fault tolerant items with redundancy implementing decomposed safety requirements. Therefore, there are no additional restrictions on ISO 26262-9:2018, Clause 5.

NOTE Fault tolerant items need to demonstrate sufficient independency (see ISO 26262-9:2018, 5.4.3)

If the ASIL decomposition is applied to the redundant elements of a fault tolerant item, then the design decisions consider both the result of ASIL decomposition and the result of the hazard analysis and risk assessment of the item applied to the state after the loss of redundancy [in case of 12.2.4.2 b)].

The ASILs of redundant safety requirements are assigned considering:

— the required minimum ASIL for maintaining operation of the item after the loss of redundancy; and

— the ASIL resulting from the decomposition of an initial safety requirement.

EXAMPLE An item has an ASIL D safety goal that "The loss of more than 60 % of the output capability for longer than X shall be prevented". The item is implemented as two independent components, each providing 50 % of the desired output. The two outputs are summed together (Figure 29). Each component has sufficient authority to maintain the intended service above 40 % and is sufficient to maintain control and prevent a hazard.

In response to the failure of one of the components, the vehicle operates in a degraded mode because its performance is limited independently of the remaining component. A hazard analysis and risk assessment determines an ASIL B level for operation during degraded mode because the vehicle performance is limited by another item and the malfunction during degraded mode is mitigated from initial operating mode with ASIL D to satisfy the original safety goal.

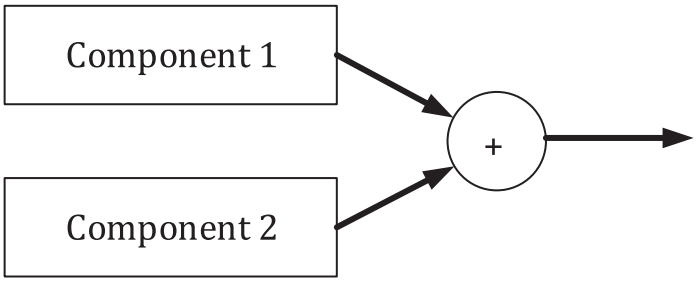


Figure 29 — Fault Tolerant Item consisting of two independent components summed together

ISO 26262-9:2018 5.4.9, allows the following ASIL decomposition pairs for the two requirements implemented into independent components:

a) ASIL C(D) and ASIL A(D);

b) ASIL B(D) and ASIL B(D); and

c) ASIL D(D) and QM(D).

However, for a design decision that restricts the ASIL of the item's safety goals to ASIL B, when the vehicle performance is limited, the minimum ASIL capability for each component would be ASIL B and the most suitable decomposition for this design decision is b) ASIL B(D) and ASIL B(D). Options a) and c) are not suitable unless the ASIL of the remaining component is raised to at least ASIL B [i.e. ASIL C(D) and ASIL B(D) or ASIL D(D) and ASIL B(D)].

12.3 Availability considerations during hardware design phase

12.3.1 Random hardware fault quantitative analysis

12.3.1.1 Emergency Operation Tolerance Time Interval calculation method

For systems where fault tolerance is achieved using redundancy, once the system has completed the fault reaction to the first fault, the system is in an operating mode without redundancy. If the ASIL capability of the system with such an operating mode does not meet the ASIL derived from the vehicle operating state, the amount of time allowable to stay in this vehicle operating state is limited to reduce the risk of a second fault. This is a possible factor to determine emergency operation tolerance time interval and the rationale of the time is confirmed by quantitative analysis required by ISO 26262-5:2018, Clause 9.

If the method of ISO 26262-5:2018, 9.4.2 is used to determine the metric for random hardware failure, the PMHF estimation considering the Emergency Operation Tolerance Time Interval (T_{eotti}) satisfies the PMHF target. Alternatively, the PMHF can be used to calculate a limit for T_{eotti} with respect to the risk of safety goal violation from a subsequent random hardware fault of an element.

NOTE Since the PMHF value itself has no absolute significance (see ISO 26262-5:2018, 9.4.2.2, NOTE 1), using it to calculate T_{eotti} is one option. Other quantitative or qualitative approaches are possible.

The T_{eotti} can also be restricted by considering it as a property of the item state after the occurrence of the fault or loss of redundancy. For this state, appropriateness of the T_{eotti} is decided by comparing the probabilistic metric of violating the safety goal over the expected usage of the vehicle (PMHF × T_{lifetime}) to the probabilistic metric of violating the safety goal while operating without redundancy. A sample formula (making a first order approximation) is given in Equation (1):

$$T_{\text{eotti}} \leq T_{\text{lifetime}} \times \lambda_{\text{target}} / \lambda_{\text{degr}} \quad (1)$$

where

λ_{target} is the target PMHF (derived in accordance with ISO 26262-5:2018, 9.4.2.2) corresponding to the ASIL rating of the item after the occurrence of the fault or loss of redundancy. Without any specification for the degraded mode or emergency operation, the initial ASIL is used;

λ_{degr} for the item state after the occurrence of the fault or loss of redundancy, the average probability per hour over the Emergency Operation Tolerance Time Interval of a failure that results in a violation of the safety goal.

The specific formula depends on the system architecture and detailed design.

12.3.1.2 Example — Dynamic redundant architecture with standby

Figure 30 shows a dynamic redundant architecture with standby which is used to demonstrate the calculation of T_{eotti}.

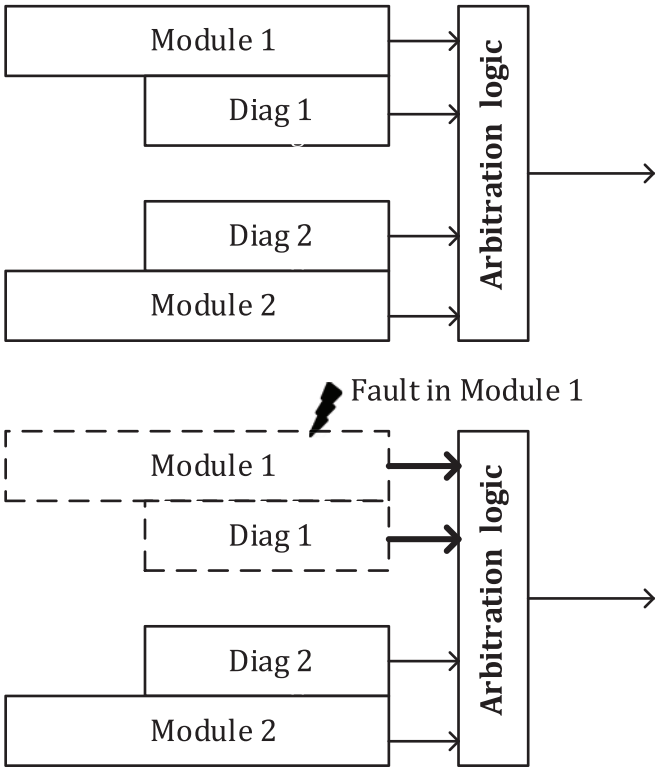


Figure 30 — Example — Dynamic redundant architecture with standby

This case is analogous to the example of 8.3.2.2 with Module 1 as the IF and Module 2 as the SM1. The formula for the PMHF is the same as the one given in 8.3.2.4 with $T_{service}$ replaced with T_{eotti}. For this system, T_{eotti} can now be calculated as Equation (2):

$$T_{eotti} \leq (M_{PMHF} - \lambda_{SPF} - \lambda_{RF} - 0{,}5 \times \lambda_{SM1,DPF,latent} \times \lambda_{IF,DPF} \times T_{lifetime} - 0{,}5 \times \lambda_{IF,DPF,latent} \times \lambda_{SM1,DPF} \times T_{lifetime}) / (\lambda_{SM1,DPF,detected} \times \lambda_{IF,DPF} + \lambda_{IF,DPF,detected} \times \lambda_{SM1,DPF}) \quad (2)$$

Equation (1) can also be determined using $\lambda_{target} = M_{PMHF}$ and $\lambda_{degr} = \lambda_{SM1,DPF}$ as

$$T_{eotti} \leq T_{lifetime} \times M_{PMHF} / \lambda_{SM1,DPF} \quad (3)$$

Table 6 compares the results from the two equations for $T_{lifetime}$ = 10 000 h and two sets of failure rates. For Case 1, Equation (3) is the limiting factor and $T_{eotti} \leq$ 167 h. For Case 2, Equation (2) is the limiting factor and $T_{eotti} \leq$ 31 h.

The emergency operation tolerance time interval is specified as a part of the functional safety requirements. In this example, the appropriateness of the EOTTI is confirmed by considering both methods described in 12.3.1.1, illustrated by Equations (2) and (3). Equation (2) gives the amount of margin for the emergency operation tolerance time interval for a given PMHF target and element failure rates and latent diagnostic test time intervals. On the other hand, Equation (3) calculates an additional limitation for T_{eotti} as a property of the item state after the occurrence of the fault or loss of redundancy.

Table 6 — Example values, Equations (2) and (3)

Lambdas (h^{-1})	Case 1	Case 2
PMHF	1,0E-7	1,0E-7
SF	2,0E-9	2,0E-9
RF	1,2E-8	6,0E-8

Table 6 *(continued)*

Lambdas (h^{-1})	Case 1	Case 2
IF,DPF	6,0E-6	6,0E-6
IF,DPF,DETECTED	5,4E-6	5,4E-6
IF,DPF,LATENT	6,0E-7	6,0E-7
SM1,DPF	6,0E-6	6,0E-6
SM1,DPF,DETECTED	5,4E-6	5,4E-6
SM1,DPF,LATENT	6,0E-7	6,0E-7
Results		
Equation (2)	772 h	31 h
Equation (3)	167 h	167 h

NOTE 1 ISO 26262-5:2018, 9.4.2.4 gives the mean duration of a vehicle trip can be considered as being equal to 1 h.

EXAMPLE To allow operation of a system for 10 trips or key cycles after the occurrence of the fault, the Emergency Operation Tolerance Time Interval needs to be greater or equal to 10 h.

NOTE 2 If the resulting T_{eotti} based on the PMHF calculation is too restricted, then other parameters such as residual failure rates can be addressed to relax the restriction of T_{eotti}.

12.3.1.3 Emergency Operation Time Interval calculation if no PMHF value is available

If the method of ISO 26262-5:2018, 9.4.3 is used, the criteria provided in ISO 26262-5:2018 9.4.3.13 is applicable.

12.3.1.4 Allocation of requirements after transition to safe state

If an item still provides some specified functions after reaching another safe state without timing restrictions, the ASIL capability of the remaining safety measures is evaluated. Relevant clauses of ISO 26262-5:2018 can be applied, including Clauses 8 and 9.

NOTE For systematic faults, fault avoidance measures are comprehended as part of the development. If the effectiveness of these measures is quantified, then these measures can be taken into account for quantitative safety analysis (i.e. hardware architectural metrics of ISO 26262-5:2018, Clause 8 and PMHF or EEC of Clause 9).

12.4 Software development phase

12.4.1 Software fault avoidance and tolerance

Safety-related availability requirements for software can be addressed by two approaches: fault avoidance (12.4.2) and fault tolerance (12.4.3).

12.4.2 Software fault avoidance

The methods for fault avoidance are intended to reduce the overall occurrence of systematic faults. The necessary amount of fault avoidance can be achieved by developing software elements using ISO 26262-6.

12.4.3 Software fault tolerance

Techniques for fault tolerance try to keep the item operational despite the presence of software systematic faults. Some fault tolerant mechanisms are mentioned in ISO 26262-6:2018, 7.4.12, NOTE 2 and NOTE 3.

13 Remark on “Confidence in the use of software tools”

The process for determining confidence in the use of software tools, described in ISO 26262-8:2018, Clause 11, is divided into two steps:

1st step: Evaluation of use cases of the tools

The requirements for tool qualification are based on the determination of “Tool impact” (TI) and the “Tool error Detection” (TD) classes. TI represents the possibility that a malfunction of a particular software tool can introduce or fail to detect errors in a safety-related item or element being developed. TD represents the confidence in measures that prevent the software tool from malfunctioning and producing corresponding erroneous output, or in measures that detect that the software tool has malfunctioned and has produced corresponding erroneous output. TI and TD are used to determine the “Tool Confidence Level” (TCL).

TI and TD are determined based on the specific use-cases of the intended software tool. The evaluation of use-cases can be done independently from the specific tool itself.

2nd step: Qualification of a software tool

If the 1st step results in a Tool Confidence Level TCL2 or TCL3, then qualification measures are means to ensure, that the user can rely on the correct functioning of a software tool and that the software tool is suitable to be used to support the activities or tasks required by the ISO 26262 series of standards. In this case, at least one of four qualification methods is recommended:

— Increased confidence from use.

— Evaluation of the tool development process.

— Validation of the software tool.

— Development in accordance with a safety standard.

The qualification method is applied to a specific software tool, its version and its environment. Therefore, ISO 26262-8:2018, 11.4.6.2, describes the documentation of:

— a unique identification and version number of the tool (a), and

— the configuration and environment for which the software tool is qualified (d).

Tool qualification often leads to high effort, especially in case of frequent changes of the tool or its version (e.g. in case of updates, patches, etc.), because the tool needs to be re-qualified for each new version. Re-qualification also applies to changes of the tool environment (e.g. the operating system or commonly used software libraries) which could have an impact on the tool output.

An alternative to tool qualification is to increase the probability to detect erroneous tool outputs by introducing additional measures into the product development process which uses the software tool. This would reduce the Tool error Detection classification to TD1. In this case the process flow is given in Figure 31:

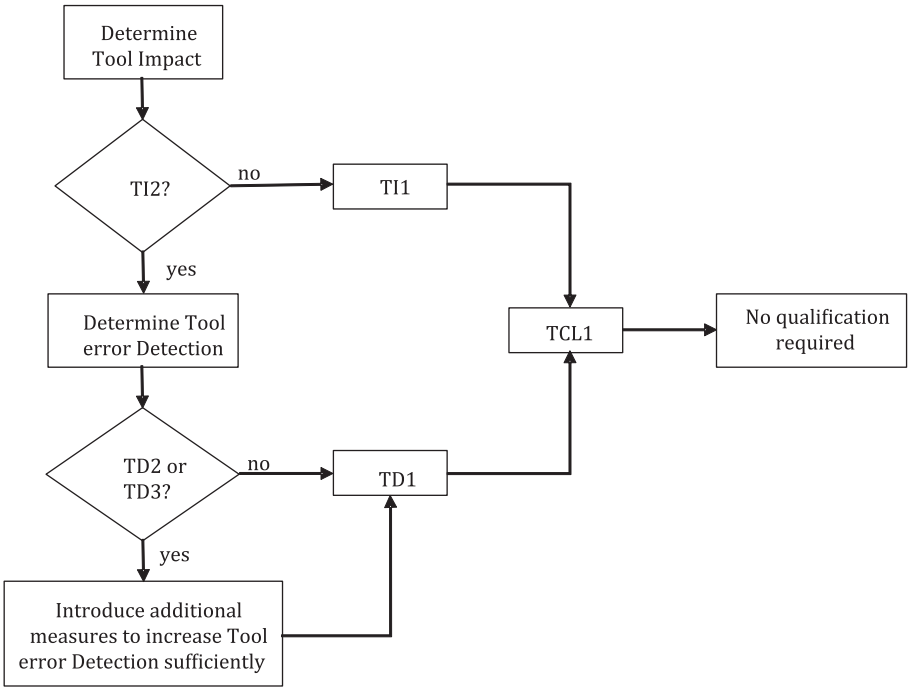


Figure 31 — Tool qualification flow chart to achieve TCL1 classification of the tool

Since this alternative does not require the qualification of the specific software tool (2nd step), it is based only on the use-cases of the tool and can be performed independently of the specific tool, tool version and its environment.

This approach can lead to a higher initial and ongoing development effort because of the necessity to introduce additional measures for increasing the Tool error Detection (e.g. review of the tool output, additional test step, check by a subsequent tool, etc.). However, this typically results in a much lower tool qualification effort since the subsequent qualification steps can be omitted and in an ideal case this procedure is done only once.

The Tool Confidence Level is valid as long as the use-cases remain unchanged. In case of additional use-cases, the classification (1st step) is updated (impact analysis), which could result in the need for further measures to increase the Tool error Detection.

14 Guidance on safety-related special characteristics

14.1 General

This section gives guidance on safety-related special characteristics from their identification during the product development phase to the monitoring during the production phase.

Management of special characteristics is an established procedure to ensure that manufactured products or their elements provide the level of safety and quality required by customers. Therefore, the general approach in the ISO 26262 series of standards is compatible to the approach defined in established automotive quality management systems (see ISO 26262-2:2018, 5.4.5). The ISO 26262 series of standards has a specific focus on safety-related electrical, electronic and software elements of automotive products.

According to ISO 26262-7, the compliance with all safety-related special characteristics of items or elements during their production is necessary to achieve the functional safety of a product.

NOTE Special characteristics can be product characteristics or manufacturing process parameters.

The management of the safety-related special characteristics consists of:

— their identification during development;

— the specification of control measures used to control them during production planning; and

— the monitoring of their fulfilment during production.

The safety-related special characteristics are specified in ISO 26262-4:2018, 6.4.8.1 and ISO 26262-5:2018, 7.4.5.1, and are traceable in all of these three phases. Moreover, it is checked that each identified safety-related special characteristic has been suitably planned and controlled. The functional safety assessment can be used to provide evidence that the proper safety related special characteristics have been identified during the development phase. The production capability assessment is used to provide evidence that the production is capable of meeting the safety related special characteristics.

NOTE Relevant safety-related special characteristics could be exchanged between different organizations, e.g. customers and suppliers, to ensure traceability.

14.2 Identification of safety-related special characteristics

Safety-related special characteristics are identified both during product development and during production planning. To be able to identify the safety impact of the special characteristic on the item or element, information could be retrieved from the safety analyses reports according to ISO 26262-9.

NOTE 1 Production planning is initiated during the development.

NOTE 2 Not all special characteristics are safety-related.

Safety-related special characteristics can be identified at system, hardware, and software levels, and for production.

EXAMPLE 1 Calibration of an e-Motor Resolver offset is identified as a safety requirement for manufacturing during a system FMEA and an action is assigned to specify a safety-related special characteristic to be met during production for end-of-line testing, including storing calibration data and test results. The Process Control Plan specifies that e-Motor calibration is a safety-related special characteristic.

EXAMPLE 2 Minimum distance between two adjacent PINs ensuring electrical insulation is identified as a safety requirement for manufacturing during a component FMEA. An action is assigned to specify a safety-related special characteristic to be met during production of the hardware component, including test of electrical parameters, and an action is assigned to place a special characteristic symbol on the assembly drawing.

EXAMPLE 3 Correct selection of embedded software including calibration data for downloading to an ECU is identified as a safety-related special characteristic in a Process FMEA to be met during end-of-line programming of an ECU by comparing checksums.

EXAMPLE 4 Amount of solder paste deposited during the production process is identified as a safety-related special characteristic during a process FMEA to be met during production of the PCB, including control by vision systems.

14.3 Specification of the control measures of safety-related special characteristics

Once the safety-related special characteristics have been identified, criteria and requirements to control them during the production are specified to ensure functional safety of the item or element.

Typically, the specification of the criteria and measures for controlling safety-related special characteristics includes:

— acceptable parameter range;

EXAMPLE Acceptable current and voltage ranges.

— evaluation or measurement technique including test ID;

EXAMPLE Automatic Optical Inspection, End-Of-Line test, and In-Circuit Test.

— control strategy; and

NOTE Control of samples could be statistically based or applied on all samples with a certain frequency. In this case sample size and frequency of the control are specified. These requirements can be given to external (e.g. suppliers) or internal production teams.

— Acceptance criteria.

EXAMPLE Accepted tolerance for the width of a soldering patch

14.4 Monitoring of the safety-related special characteristics

Evidence of the planning and implementation of the control of safety-related special characteristics, including those from customers and suppliers, can be documented in the work products of ISO 26262-7:2018, 6.5.1 and 6.5.2.

Annex A
(informative)

Fault tree construction and applications

A.1 General

The two most common techniques for analysing faults and failures of items and elements are FTA and FMEA. The FMEA is typically performed as an inductive (bottom up, see Figure A.1) approach focusing on the individual parts of the system, how they can fail and the impact of these failures on the system. The FTA is typically performed as a deductive (top down, see Figure A.2) approach starting with the undesired system behaviour and determining the possible causes of this behaviour. FTA includes coverage for combinations of multiple faults and events or situations which may lead to a hazard, while FMEA considers the effects of individual faults.

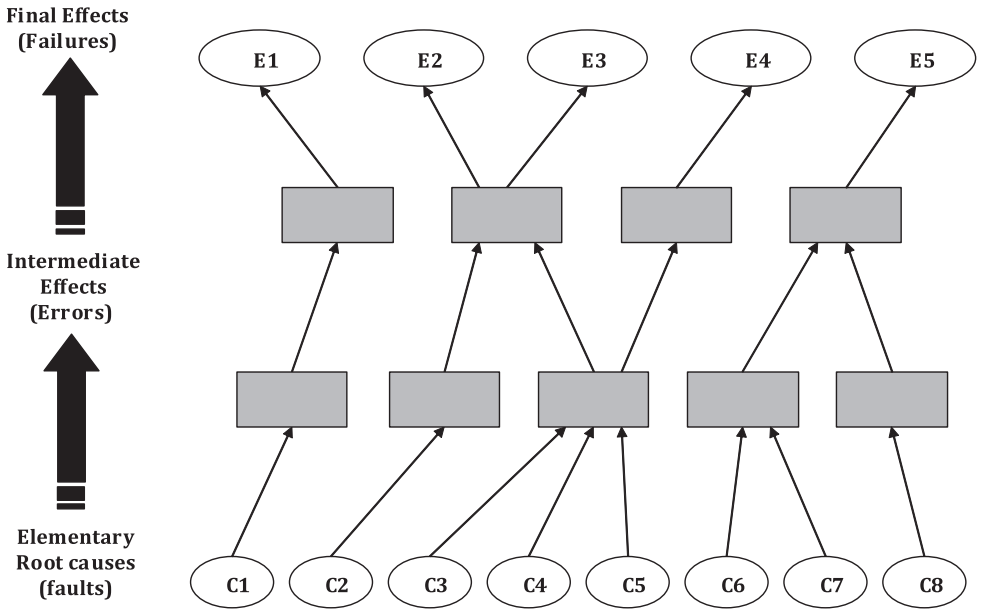


Figure A.1 — Illustration of FMEA, Bottom Up Approach

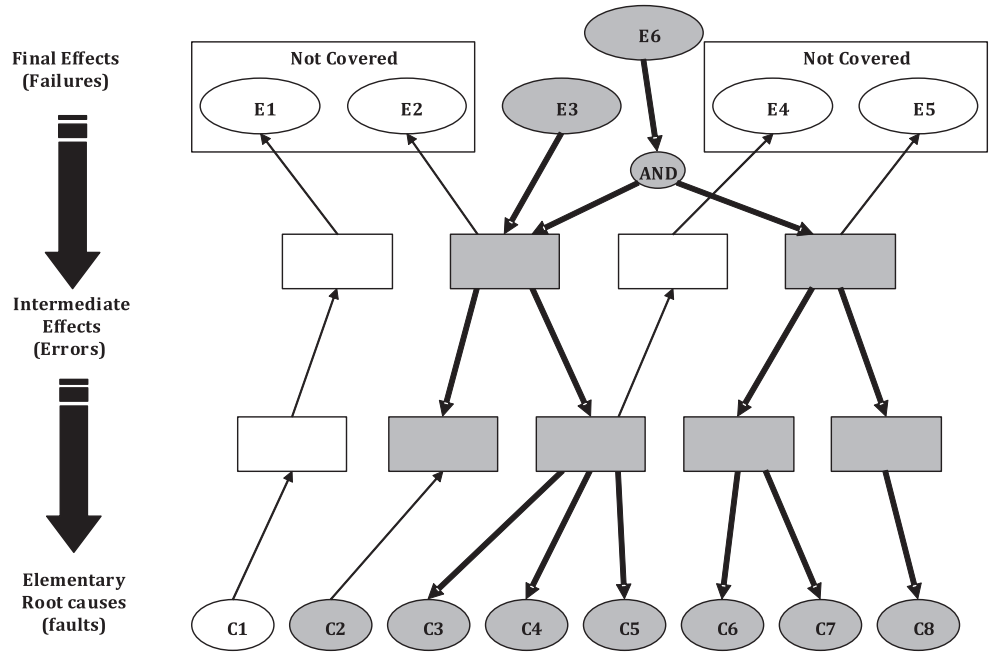


Figure A.2 — Illustration of FTA, Top Down Approach

The approaches are complementary as stated in ISO 26262-5:2018 7.4.3.1, Table 2 NOTE: "The level of detail of the analysis is commensurate with the level of detail of the design. Both methods can, in certain cases, be carried out at different levels of detail." The "Cx" ovals of Figures A.1 and A.2 represent either hardware or software components. A typical approach is to use the FTA to analyse the hazards down to the component level. The failure modes of the components are then analysed from the bottom up using an FMEA to determine their failure modes and safety mechanisms to close out the bottom level of the fault tree. It is desirable to avoid duplicate work which would be caused by overlap between FTA modelling and FMEA. Preferably the results of the FMEA of serial system parts are fed as failure rates of the base events into the fault tree model.

NOTE 1 As stated in ISO 26262-9:2018, 7.4.2, the contribution of dependent failures is estimated on a qualitative basis because no general and sufficiently reliable method exists for quantifying such failures.

NOTE 2 Example standards for FMEA include JEP131A [3] and SAE J1739 [4], and for FTA IEC 61025 [5].

A.2 Combining FTA and FMEA

Systems are composed of many parts and subparts. FTA and FMEA can be combined to provide the safety analysis with the right balance of top-down and bottom-up approach. Figure A.3 shows a possible approach to combining an FTA with an FMEA. In this figure, the basic events are derived from different FMEAs (labelled FMEA A-E within this example) which was performed at a lower level of abstraction (e.g. subpart, part or component level). Within this example, basic events 1 and 2 are fault effects as found in FMEA D, while no fault effects from FMEA B are used in the fault tree.

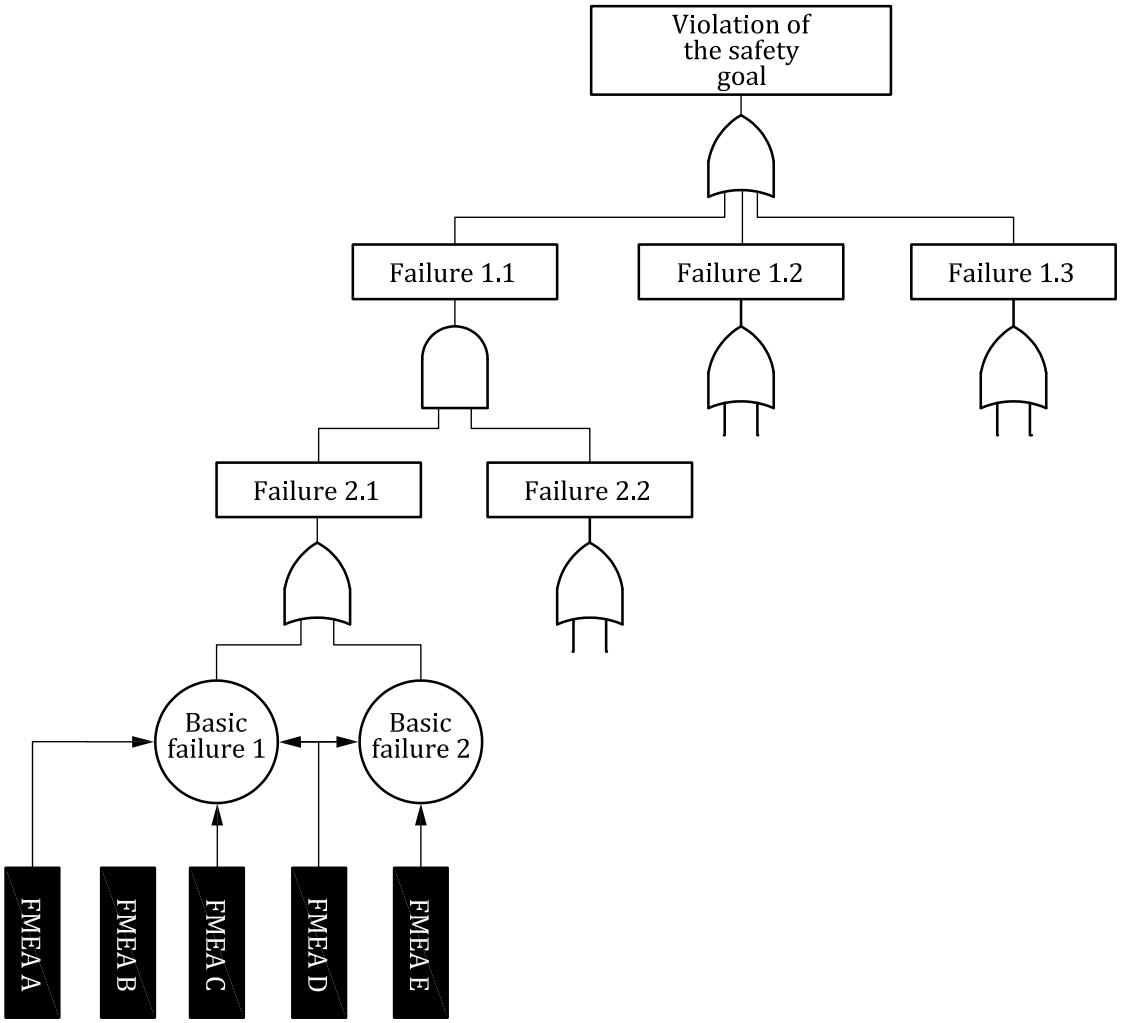


Figure A.3 — Illustration of a combination of FTA and FMEA

Bibliography

[1] IEC 61508 (all parts), *Functional safety of electrical/electronic/programmable electronic safety-related systems*

[2] GSN COMMUNITY STANDARD VERSION 1, November 2011

[3] JEDEC – JEP131A (May 2005), Potential Failure Mode and Effects Analysis (FMEA)

[4] SAE-J1739_200901, *Potential Failure Mode and Effects Analysis in Design (Design FMEA) and Potential Failure Mode and Effects Analysis in Manufacturing and Assembly Processes (Process FMEA) and Effects Analysis for Machinery (Machinery FMEA)*

[5] IEC 61025, *ed. 2.0 — Procedures and Symbols for FTA*

[6] SAE J2980, *Considerations for ISO 26262 ASIL Hazard Classification*

[7] ISO 26262-2:2018, *Road vehicles — Functional safety — Part 2: Management of Functional Safety*

[8] ISO 26262-3:2018, *Road vehicles — Functional safety — Part 3: Concept phase*

[9] ISO 26262-4:2018, *Road vehicles — Functional safety — Part 4: Product development at the system level*

[10] ISO 26262-5:2018, *Road vehicles — Functional safety — Part 5: Product development at the hardware level*

[11] ISO 26262-6:2018, *Road vehicles — Functional safety — Part 6: Product development at the software level*

[12] ISO 26262-7:2018, *Road vehicles — Functional safety — Part 7: Production, operation, service and decommissioning*

[13] ISO 26262-8:2018, *Road vehicles — Functional safety — Part 8: Supporting processes*

[14] ISO 26262-9:2018, *Road vehicles — Functional safety — Part 9: Automotive Safety Integrity Level (ASIL) oriented and safety-oriented analyses*

[15] ISO 26262-11:2018, *Road vehicles — Functional safety — Part 11: Guideline on application of ISO 26262 to semiconductors*

[16] ISO 26262-12:2018, *Road vehicles — Functional safety — Part 12: Adaptation of ISO 26262 for motorcycles*

[17] Convention on Road Traffic. Done at Vienna on 8 November 1968 including amendment 1, Economic Commission for Europe, Inland Transportation Committee, [viewed 2018-09-25] Available at: https://www.unece.org/fileadmin/DAM/trans/conventn/crt1968e.pdf

ICS 43.040.10

Price based on 79 pages