DIN EN 61508-6
(VDE 0803-6)

DIN

Diese Norm ist zugleich eine **VDE-Bestimmung** im Sinne von VDE 0022. Sie ist nach Durchführung des vom VDE-Präsidium beschlossenen Genehmigungsverfahrens unter der oben angeführten Nummer in das VDE-Vorschriftenwerk aufgenommen und in der „etz Elektrotechnik + Automation“ bekannt gegeben worden.

VDE

ICS 35.240.50

Ersatz für
DIN EN 61508-6
(VDE 0803-6):2003-06
Siehe Anwendungsbeginn

Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme – Teil 6: Anwendungsrichtlinie für IEC 61508-2 und IEC 61508-3 (IEC 61508-6:2010); Deutsche Fassung EN 61508-6:2010

Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 6: Guidelines on the application of IEC 61508-2 and IEC 61508-3 (IEC 61508-6:2010); German version EN 61508-6:2010

Sécurité fonctionnelle des systèmes électriques/électroniques/électroniques programmables relatifs à la sécurité – Partie 6: Lignes directrices pour l`application de la CEI 61508-2 et de la CEI 61508-3 (CEI 61508-6:2010); Version allemande EN 61508-6:2010

Gesamtumfang 121 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE

Preisgr. 56 K
VDE-Vertr.-Nr. 0803017

Einzelverkauf und Abonnements durch VDE VERLAG GMBH, 10625 Berlin
Einzelverkauf auch durch Beuth Verlag GmbH, 10772 Berlin

Anwendungsbeginn

Anwendungsbeginn für die von CENELEC am 2010-05-01 angenommene Europäische Norm als DIN-Norm ist 2011-02-01.

Daneben darf DIN EN 61508-6 (VDE 0803-6):2003-06 noch bis 2013-05-01 angewendet werden.

Nationales Vorwort

Vorausgegangener Norm-Entwurf: E DIN EN 61508-6 (VDE 0803-6):2009-06.

Für diese Norm ist das nationale Arbeitsgremium GK 914 „Funktionale Sicherheit elektrischer, elektronischer und programmierbarer elektronischer Systeme (E, E, PES) zum Schutz von Personen und Umwelt“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE (www.dke.de) zuständig.

Die enthaltene IEC-Publikation wurde vom SC 65A „System aspects“ erarbeitet.

Das IEC-Komitee hat entschieden, dass der Inhalt dieser Publikation bis zu dem Datum (maintenance result date) unverändert bleiben soll, das auf der IEC-Website unter „http://webstore.iec.ch“ zu dieser Publikation angegeben ist. Zu diesem Zeitpunkt wird entsprechend der Entscheidung des Komitees die Publikation
– bestätigt,
– zurückgezogen,
– durch eine Folgeausgabe ersetzt oder
– geändert.

Änderungen

Gegenüber DIN EN 61508-6 (VDE 0803-6):2003-06 wurden folgende Änderungen vorgenommen:

a) vollständige Überarbeitung des Anhangs B;

b) vollständige Überarbeitung des Anhangs D.

Frühere Ausgaben

DIN EN 61508-6 (VDE 0803-6): 2003-06

Nationaler Anhang NA
(informativ)

Zusammenhang mit Europäischen und Internationalen Normen

Für den Fall einer undatierten Verweisung im normativen Text (Verweisung auf eine Norm ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnittsnummer, eine Tabelle, ein Bild usw.) bezieht sich die Verweisung auf die jeweils neueste gültige Ausgabe der in Bezug genommenen Norm.

Für den Fall einer datierten Verweisung im normativen Text bezieht sich die Verweisung immer auf die in Bezug genommene Ausgabe der Norm.

Eine Information über den Zusammenhang der zitierten Normen mit den entsprechenden Deutschen Normen ist in Tabelle NA.1 wiedergegeben.

Tabelle NA.1

Europäische Norm	Internationale Norm	Deutsche Norm	Klassifikation im VDE-Vorschriftenwerk
EN 60601 (alle Teile)	IEC 60601 (alle Teile)	DIN EN 60601 (VDE 0750-1) (alle Teile)	VDE 0750-1
EN 61025:2001	IEC 61025:2006	DIN EN 61025:2007-08	–
EN 61078:2006	IEC 61078:2006	DIN EN 61078:2006-10	–
EN 61165:2006	IEC 61165:2006	DIN EN 61165:2007-02	–
EN 61508-1:2010	IEC 61508-1:2010	DIN EN 61508-1 (VDE 0803-1):2011-02	VDE 0803-1
EN 61508-2:2010	IEC 61508-2:2010	DIN EN 61508-2 (VDE 0803-2):2011-02	VDE 0803-2
EN 61508-3:2010	IEC 61508-3:2010	DIN EN 61508-3 (VDE 0803-3):2011-02	VDE 0803-3
EN 61508-4:2010	IEC 61508-4:2010	DIN EN 61508-4 (VDE 0803-4):2011-02	VDE 0803-4
EN 61508-5:2010	IEC 61508-5:2010	DIN EN 61508-5 (VDE 0803-5):2011-02	VDE 0803-5
EN 61508-7:2010	IEC 61508-7:2010	DIN EN 61508-7 (VDE 0803-7):2011-02	VDE 0803-7
EN 61511 (alle Teile)	IEC 61511 (alle Teile)	DIN EN 61511 (VDE 0810-1) (alle Teile)	VDE 0810-1
EN 61800-5-2	IEC 61800-5-2	DIN EN 61800-5-2 (VDE 0160-105-2)	VDE 0160-105-2
EN 62061	IEC 62061	DIN EN 62061 (VDE 0113-50)	VDE 0113-50
–	IEC 65B/725/CD:2009	E DIN IEC 61131-3:2009-12	–
–	IEC 56/1266/CD:2008	E DIN IEC 62551 (VDE 0050-4):2008-10	VDE 0050-4
–	ISA-TR84.00.02-2002	–	–

Nationaler Anhang NB
(informativ)

Literaturhinweise

DIN EN 60601 (VDE 0750-1) (alle Teile), *Klassifikation und Kennzeichnung von Dokumenten für Anlagen, Systeme und Ausrüstungen*

DIN EN 61025:2007-08, *Fehlzustandsbaumanalyse (IEC 61025:2006); Deutsche Fassung EN 61025:2007*

DIN EN 61078:2006-10, *Techniken für die Analyse der Zuverlässigkeit – Zuverlässigkeitsblockdiagramm und Boole'sche Verfahren*

DIN EN 61165:2007-02, *Anwendung des Markoff-Verfahrens (IEC 61165:2006); Deutsche Fassung EN 61165:2006*

DIN EN 61508-6 (VDE 0803-6):2011-02

DIN EN 61508-1 (VDE 0803-1):2011-02, *Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme – Teil 1: Allgemeine Anforderungen (IEC 61508-1:2010); Deutsche Fassung EN 61508-1:2010*

DIN EN 61508-2 (VDE 0803-2):2011-02, *Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme – Teil 2: Anforderungen an sicherheitsbezogene elektrische/elektronische/programmierbare elektronische Systeme (IEC 61508-2:2010); Deutsche Fassung EN 61508-2:2010*

DIN EN 61508-3 (VDE 0803-3):2011-02, *Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme – Teil 3: Anforderungen an Software (IEC 61508-3:2010); Deutsche Fassung EN 61508-3:2010*

DIN EN 61508-4 (VDE 0803-4):2011-02, *Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme – Teil 4: Begriffe und Abkürzungen (IEC 61508-4:2010); Deutsche Fassung EN 61508-4:2010*

DIN EN 61508-5 (VDE 0803-5):2011-02, *Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme – Teil 5: Beispiele von Methoden für die Bestimmung von Sicherheits-Integritätsleveln (IEC 61508-5:2010); Deutsche Fassung EN 61508-5:2010*

DIN EN 61508-7 (VDE 0803-7):2011-02, *Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme – Teil 7: Überblick über Verfahren und Maßnahmen (IEC 61508-7:2010); Deutsche Fassung EN 61508-7:2010*

DIN EN 61511 (VDE 0810-1) (alle Teile), *Funktionale Sicherheit – Sicherheitstechnische Systeme für die Prozessindustrie*

DIN EN 61800-5-2 (VDE 0160-105-2), *Elektrische Leistungsantriebssysteme mit einstellbarer Drehzahl – Teil 5-2: Anforderungen an die Sicherheit – Funktionale Sicherheit*

DIN EN 62061 (VDE 0113-50), *Sicherheit von Maschinen – Funktionale Sicherheit sicherheitsbezogener elektrischer, elektronischer und programmierbarer elektronischer Steuerungssysteme*

E DIN IEC 61131-3:2009-12, *Speicherprogrammierbare Steuerungen – Teil 3: Programmiersprachen; Englische Fassung (IEC 65B/725/CD:2009)*

E DIN IEC 62551 (VDE 0050-4):2008-10, *Analysemethoden für Zuverlässigkeit – Petrinetz-Modellierung (IEC 56/1266/CD:2008)*

EUROPÄISCHE NORM

EUROPEAN STANDARD

NORME EUROPÉENNE

EN 61508-6

Mai 2010

ICS 35.240.50

Ersatz für EN 61508-6:2001

Deutsche Fassung

Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme – Teil 6: Anwendungsrichtlinie für IEC 61508-2 und IEC 61508-3

(IEC 61508-6:2010)

Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 6: Guidelines on the application of IEC 61508-2 and IEC 61508-3 (IEC 61508-6:2010)

Sécurité fonctionnelle des systèmes électriques/électroniques/électroniques programmables relatifs à la sécurité – Partie 6: Lignes directrices pour l`application de la CEI 61508-2 et de la CEI 61508-3 (CEI 61508-6:2010)

Diese Europäische Norm wurde von CENELEC am 2010-05-01 angenommen. Die CENELEC-Mitglieder sind gehalten, die CEN/CENELEC-Geschäftsordnung zu erfüllen, in der die Bedingungen festgelegt sind, unter denen dieser Europäischen Norm ohne jede Änderung der Status einer nationalen Norm zu geben ist.

Auf dem letzten Stand befindliche Listen dieser nationalen Normen mit ihren bibliographischen Angaben sind beim Zentralsekretariat oder bei jedem CENELEC-Mitglied auf Anfrage erhältlich.

Diese Europäische Norm besteht in drei offiziellen Fassungen (Deutsch, Englisch, Französisch). Eine Fassung in einer anderen Sprache, die von einem CENELEC-Mitglied in eigener Verantwortung durch Übersetzung in seine Landessprache gemacht und dem Zentralsekretariat mitgeteilt worden ist, hat den gleichen Status wie die offiziellen Fassungen.

CENELEC-Mitglieder sind die nationalen elektrotechnischen Komitees von Belgien, Bulgarien, Dänemark, Deutschland, Estland, Finnland, Frankreich, Griechenland, Irland, Island, Italien, Kroatien, Lettland, Litauen, Luxemburg, Malta, den Niederlanden, Norwegen, Österreich, Polen, Portugal, Rumänien, Schweden, der Schweiz, der Slowakei, Slowenien, Spanien, der Tschechischen Republik, Ungarn, dem Vereinigten Königreich und Zypern.

CENELEC

Europäisches Komitee für Elektrotechnische Normung
European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique

Zentralsekretariat: Avenue Marnix 17, B-1000 Brüssel

Ref. Nr. EN 61508-6:2010 D

Vorwort

Der Text des Schriftstücks 65A/553/FDIS, zukünftige 2. Ausgabe von IEC 61508-6, ausgearbeitet von dem SC 65A „System aspects“ des IEC/TC 65 „Industrial-process measurement, control and automation“, wurde der IEC-CENELEC Parallelen Abstimmung unterworfen und von CENELEC am 2010-05-01 als EN 61508-6 angenommen.

Diese Europäische Norm ersetzt EN 61508-6:2001.

Es wird auf die Möglichkeit hingewiesen, dass einige Elemente dieses Dokuments Patentrechte berühren können. CEN und CENELEC sind nicht dafür verantwortlich, einige oder alle diesbezüglichen Patentrechte zu identifizieren.

Nachstehende Daten wurden festgelegt:

– spätestes Datum, zu dem die EN auf nationaler Ebene durch Veröffentlichung einer identischen nationalen Norm oder durch Anerkennung übernommen werden muss (dop): 2011-02-01

– spätestes Datum, zu dem nationale Normen, die der EN entgegenstehen, zurückgezogen werden müssen (dow): 2013-05-01

Der Anhang ZA wurde von CENELEC hinzugefügt.

_____

Anerkennungsnotiz

Der Text der Internationalen Norm IEC 61508-6:2010 wurde von CENELEC ohne irgendeine Abänderung als Europäische Norm angenommen.

In der offiziellen Fassung sind unter „Literaturhinweise“ zu den aufgelisteten Normen die nachstehenden Anmerkungen einzutragen:

[1] IEC 61511 Reihe	ANMERKUNG	Harmonisiert in der Reihe EN 61511 (nicht modifiziert).
[2] IEC 62061	ANMERKUNG	Harmonisiert als EN 62061.
[3] IEC 61800-5-2	ANMERKUNG	Harmonisiert als EN 61800-5-2.
[4] IEC 61078:2006	ANMERKUNG	Harmonisiert als EN 61078:2006 (nicht modifiziert).
[5] IEC 61165:2006	ANMERKUNG	Harmonisiert als EN 61165:2006 (nicht modifiziert).
[16] IEC 61131-3:2003	ANMERKUNG	Harmonisiert als EN 61131-3:2003 (nicht modifiziert).
[18] IEC 61025:2006	ANMERKUNG	Harmonisiert als EN 61025:2007 (nicht modifiziert).
[26] IEC 60601 Reihe	ANMERKUNG	Harmonisiert in der Reihe EN 60601 (teilweise modifiziert).
[27] IEC 61508-1:2010	ANMERKUNG	Harmonisiert als EN 61508-1:2010 (nicht modifiziert).
[28] IEC 61508-5:2010	ANMERKUNG	Harmonisiert als EN 61508-5:2010 (nicht modifiziert).
[29] IEC 61508-7:2010	ANMERKUNG	Harmonisiert als EN 61508-7:2010 (nicht modifiziert).

_____

Inhalt

Seite

Seite

Bilder

Seite

Tabellen

Seite

Seite

Einleitung

Systeme, die aus elektrischen und/oder elektronischen Elementen bestehen, werden seit vielen Jahren verwendet, um Sicherheitsfunktionen in den meisten Anwendungsbereichen auszuführen. Auf Rechnern basierende Systeme (allgemein ausgedrückt programmierbare elektronische Systeme) werden in allen Anwendungsbereichen benutzt, um Nichtsicherheitsfunktionen und zunehmend auch um Sicherheitsfunktionen auszuführen. Falls Rechnersystemtechnologie wirksam und sicherheitsgerichtet eingesetzt wird, ist es wichtig, dass die für die Entscheidungsfindung Verantwortlichen ausreichende Hilfestellung bezüglich der Sicherheitsaspekte erhalten, nach denen diese Entscheidungen getroffen werden.

Diese Internationale Norm beschreibt einen allgemeinen Lösungsweg für alle Tätigkeiten während des Sicherheitslebenszyklus für Systeme, die aus elektrischen und/oder elektronischen und/oder programmierbaren elektronischen (E/E/PE) Elementen bestehen und die eingesetzt werden, um Sicherheitsfunktionen auszuführen. Dieser allgemeine Lösungsweg wurde gewählt, um ein sinnvolles und konsistentes technisches Verfahren für alle elektrischen Sicherheitssysteme zu entwickeln. Ein Hauptziel ist es, die Entwicklung von produkt- und anwendungsspezifischen internationalen Normen, die auf der Normenreihe IEC 61508 basieren, zu erleichtern.

ANMERKUNG 1 In den Literaturhinweisen sind Beispiele von produkt- und anwendungsspezifischen internationalen Normen, die auf der Normenreihe IEC 61508 basieren, enthalten (siehe Hinweise [1], [2] und [3]).

In den meisten Situationen wird Sicherheit durch eine Anzahl von Systemen erreicht, die auf vielerlei Technologien (zum Beispiel Mechanik, Hydraulik, Pneumatik, Elektrik, Elektronik, programmierbare Elektronik) basieren. Jede Sicherheitsstrategie muss deshalb nicht nur alle Elemente innerhalb eines Einzelsystems (zum Beispiel Sensoren, Steuereinheiten und Aktoren) betrachten, sondern auch all die sicherheitsbezogenen Systeme, welche die Gesamtheit von sicherheitsbezogenen Systemen bilden. Daher kann diese Internationale Norm, obwohl sie sich mit sicherheitsbezogenen E/E/PE-Systemen beschäftigt, auch einen Rahmen bereitstellen, innerhalb dessen sicherheitsbezogene Systeme basierend auf anderen Technologien betrachtet werden können.

Es ist beachtet worden, dass eine große Vielfalt von Anwendungen in vielfältigen Anwendungsbereichen vorliegt, die sicherheitsbezogene E/E/PE-Systeme verwenden, und diese einen weiten Bereich in Bezug auf Komplexität, Gefährdungs- und Risikopotentiale abdecken. In jeder speziellen Anwendung sind die erforderlichen Sicherheitsmaßnahmen von vielen anwendungsspezifischen Faktoren abhängig. Dadurch, dass diese Internationale Norm allgemein gehalten ist, wird die Formulierung solcher Maßnahmen in zukünftigen produkt- und anwendungsspezifischen internationalen Normen und in Überarbeitungen der bereits bestehenden Normen ermöglicht.

Diese Internationale Norm:

– betrachtet alle relevanten Phasen des Gesamt-Sicherheitslebenszyklus, des Sicherheitslebenszyklus des E/E/PE-Systems und des Software-Sicherheitslebenszyklus (zum Beispiel vom anfänglichen Konzept über Entwurf, Implementierung, Betrieb und Instandhaltung bis zur Außerbetriebnahme), wenn E/E/PE-Systeme benutzt werden, um Sicherheitsfunktionen auszuführen;

– wurde unter Berücksichtigung einer sich schnell entwickelnden Technologie entworfen. Der Betrachtungsrahmen ist ausreichend robust und ausführlich genug, um auch für zukünftige Entwicklungen verwendbar zu sein;

– ermöglicht die Erstellung produkt- und anwendungsspezifischer Internationaler Normen, die sich mit sicherheitsbezogenen E/E/PE-Systemen befassen. Die Entwicklung produkt- und anwendungsspezifischer Internationaler Normen sollte innerhalb des Rahmens dieser Norm zu einem hohen Grad an Übereinstimmung (zum Beispiel von zugrunde liegenden Prinzipien, Terminologie usw.) führen sowohl innerhalb der Anwendungsbereiche als auch über die Anwendungsbereiche hinweg. Dies hat sowohl sicherheitstechnische als auch wirtschaftliche Vorteile;

– liefert eine Methode für die Entwicklung der Spezifikation der Anforderungen an die Sicherheit, die notwendig ist, um die erforderliche funktionale Sicherheit für sicherheitsbezogene E/E/PE-Systeme zu erreichen;

– verwendet einen auf dem Risiko basierenden Lösungsansatz, durch den die Anforderungen an die Sicherheitsintegrität bestimmt werden können;

– führt Sicherheits-Integritätslevel für die Festlegung der Zielvorgabe der Sicherheitsintegrität der Sicherheitsfunktionen ein, die von den sicherheitsbezogenen E/E/PE-Systemen zu implementieren sind;

ANMERKUNG 2 Diese Norm legt weder die Anforderungen an den Sicherheits-Integritätslevel für irgendeine Sicherheitsfunktion fest, noch bestimmt sie, wie der Sicherheits-Integritätslevel festgelegt wird. Stattdessen stellt sie einen risikobasierenden konzeptionellen Rahmen und Beispielverfahren bereit.

– legt Ausfallgrenzwerte für die von den sicherheitsbezogenen E/E/PE-Systemen auszuführenden Sicherheitsfunktionen fest, die mit den Sicherheits-Integritätsleveln verbunden sind;
– legt eine untere Grenze für die Ausfallgrenzwerte für eine Sicherheitsfunktion fest, die von einem einzelnen sicherheitsbezogenen E/E/PE-System ausgeführt wird. Für sicherheitsbezogene E/E/PE-Systeme, die:
  – in der Betriebsart mit einer niedrigen Anforderungsrate betrieben werden, ist die untere Grenze bei einer mittleren Wahrscheinlichkeit eines gefahrbringenden Ausfalls bei Anforderung von 10^{-5} festgelegt;
  – in der Betriebsart mit einer hohen Anforderungsrate oder mit kontinuierlicher Anforderung betrieben werden, ist die untere Grenze bei einer mittleren Häufigkeit eines gefahrbringenden Ausfalls von 10^{-9} [h^{-1}] festgelegt;

ANMERKUNG 3 Ein einzelnes sicherheitsbezogenes E/E/PE-System bedeutet nicht notwendigerweise eine einkanalige Architektur.

ANMERKUNG 4 Es kann für einfache Systeme möglich sein, Entwürfe von sicherheitsbezogenen Systemen mit niedrigeren Zielwerten für die Sicherheitsintegrität zu erreichen, aber diese Grenzen werden als das betrachtet, was für relativ komplexe Systeme (zum Beispiel sicherheitsbezogene programmierbare elektronische Systeme) gegenwärtig erreicht werden kann.

– legt Anforderungen für die Vermeidung und Beherrschung von systematischen Fehlern fest, die auf Erfahrungen und Urteilsvermögen beruhen, die durch praktische Erfahrung in der Industrie gewonnen wurden. Wenn auch die Wahrscheinlichkeit des Auftretens systematischer Ausfälle im Allgemeinen nicht quantifiziert werden kann, erlaubt die Norm jedoch für eine festgelegte Sicherheitsfunktion den Anspruch zu erheben, dass der mit der Sicherheitsfunktion verbundene Ausfallgrenzwert als erreicht betrachtet werden kann, wenn alle Anforderungen dieser Norm erfüllt worden sind;
– führt die systematische Eignung ein, die für ein Element im Hinblick auf das Vertrauen gilt, dass die systematische Sicherheitsintegrität die Anforderungen des festgelegten Sicherheits-Integritätslevels erfüllt;
– lässt einen weiten Bereich von Prinzipien, Verfahren und Maßnahmen zu, um funktionale Sicherheit für sicherheitsbezogene E/E/PE-Systeme zu erreichen, verwendet aber nicht ausdrücklich das Fail-Safe-Konzept. „Fail-Safe“-Konzepte und „inhärent sichere“ Prinzipien können jedoch anwendbar sein, und der Einsatz solcher Konzepte ist zulässig, vorausgesetzt, dass die Anforderungen der zutreffenden Abschnitte in der Norm erfüllt werden.

1 Anwendungsbereich

1.1 Dieser Teil der IEC 61508 enthält Informationen und Hinweise zu IEC 61508-2 und IEC 61508-3.

– Anhang A liefert einen kurzen Überblick über die Anforderungen der IEC 61508-2 und der IEC 61508-3 und zeigt die funktionalen Schritte zu ihrer Anwendung.

– Anhang B zeigt ein Beispiel für die Anwendung der Verfahren zur Berechnung der Wahrscheinlichkeiten von Hardwareausfällen und sollte in Verbindung mit IEC 61508-2, 7.4.3 und Anhang C, und Anhang D gelesen werden.

– Anhang C zeigt ein ausgearbeitetes Beispiel für die Berechnung des Diagnosedeckungsgrads und sollte in Verbindung mit IEC 61508-2, Anhang C, gelesen werden.

– Anhang D zeigt eine Methode zur Quantifizierung der Auswirkungen von hardwarebezogenen Ausfällen infolge gemeinsamer Ursache auf die Ausfallwahrscheinlichkeit.

– Anhang E zeigt für die Sicherheits-Integritätslevel 2 und 3 ausgearbeitete Beispiele für die Anwendung der Tabellen zur Sicherheitsintegrität der Software nach IEC 61508-3, Anhang A.

1.2 IEC 61508-1, IEC 61508-2, IEC 61508-3 und IEC 61508-4 sind Sicherheitsgrundnormen, dieser Status ist aber im Zusammenhang mit einfachen sicherheitsbezogenen E/E/PE-Systemen nicht anwendbar (siehe IEC 61508-4, 3.4.3). Als Sicherheitsgrundnormen sind sie zur Verwendung durch technische Komitees bei der Erstellung von Normen entsprechend den in IEC Guide 104 und ISO/IEC Guide 51 enthaltenen Grundsätzen vorgesehen. IEC 61508-1, IEC 61508-2, IEC 61508-3 und IEC 61508-4 sind ebenfalls zur Verwendung als eigenständige Veröffentlichungen vorgesehen. Die horizontale Sicherheitsfunktion dieser Internationalen Norm trifft nicht auf Medizingeräte in Übereinstimmung mit der Normenreihe IEC 60601 zu.

1.3 Es steht in der Verantwortung eines technischen Komitees, zur Vorbereitung und Erstellung eigener Veröffentlichungen soweit möglich die Sicherheitsgrundnormen anzuwenden. In diesem Zusammenhang gilt, dass die Anforderungen, Prüfverfahren oder Prüfbedingungen dieser Sicherheitsgrundnorm nicht anzuwenden sind, außer in den Veröffentlichungen der technischen Komitees wird speziell darauf verwiesen oder sie werden eingebunden.

1.4 Bild 1 zeigt den Gesamtrahmen für die Normenreihe IEC 61508 und zeigt die Rolle, welche IEC 61508-6 beim Erreichen der funktionalen Sicherheit für sicherheitsbezogene E/E/PE-Systeme spielt.

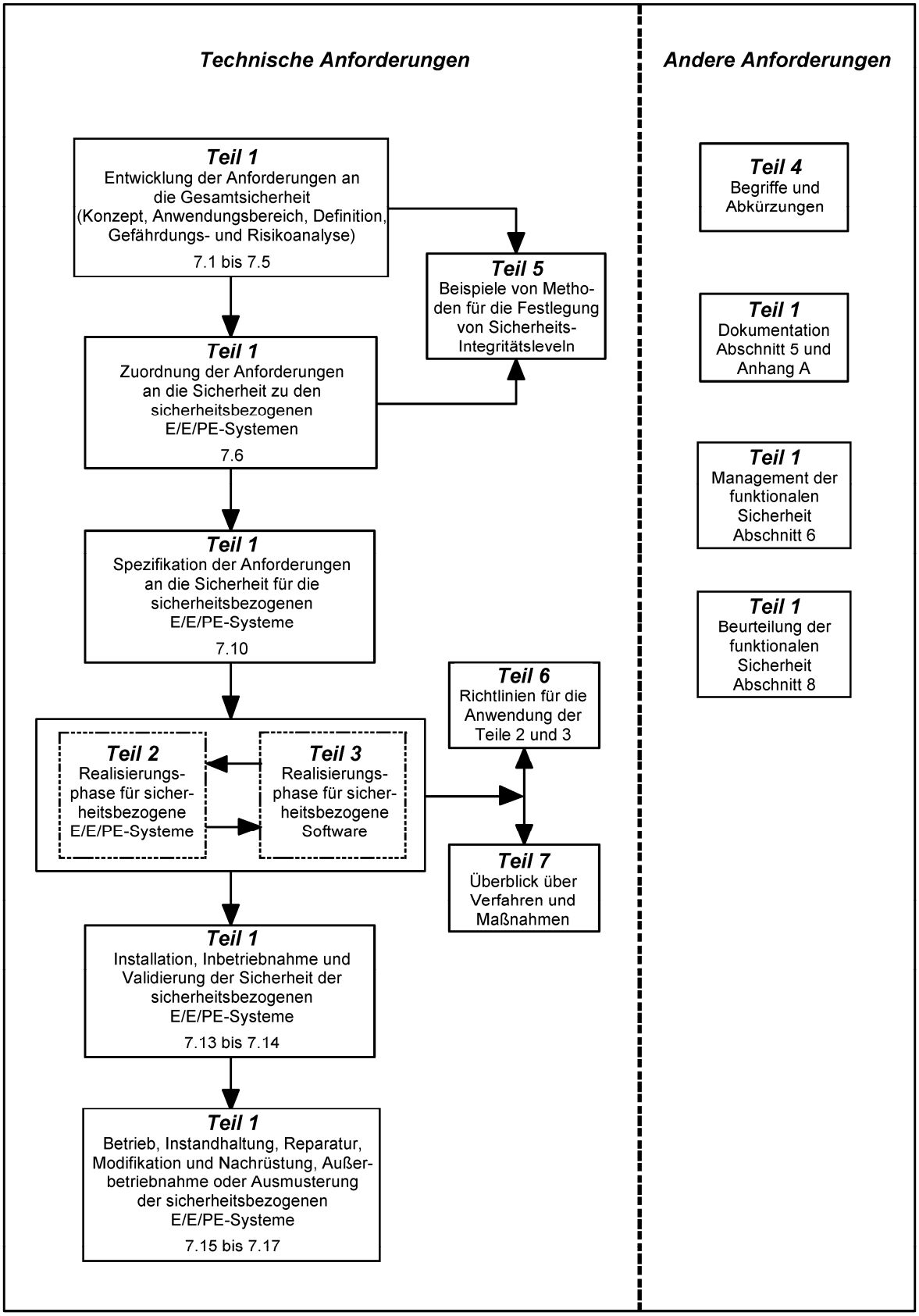


Bild 1 – Gesamtrahmen der Normenreihe IEC 61508

2 Normative Verweisungen

Die folgenden zitierten Dokumente sind für die Anwendung dieses Dokuments erforderlich. Bei datierten Verweisungen gilt nur die in Bezug genommene Ausgabe. Bei undatierten Verweisungen gilt die letzte Ausgabe des in Bezug genommenen Dokuments (einschließlich aller Änderungen).

IEC 61508-2:2010, *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 2: Requirements for electrical/electronic/programmable electronic safety-related systems*

IEC 61508-3:2010, *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 3: Software requirements*

IEC 61508-4:2010, *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 4: Definitions and abbreviations*

3 Begriffe und Abkürzungen

Für die Anwendung dieses Dokuments gelten die in IEC 61508-4 aufgeführten Begriffe und Abkürzungen.

Anhang A
(informativ)

Anwendung der IEC 61508-2 und der IEC 61508-3

A.1 Allgemeines

Maschinen, Prozessanlagen und andere Betriebsmittel können im Fall einer Fehlfunktion (z. B. aufgrund von Ausfällen elektrischer, elektronischer und/oder programmierbarer elektronischer Geräte) durch gefährliche Vorfälle wie Feuer, Explosion, Überdosis von Strahlung oder in eine Maschine eingeschlossen werden Risiken für Personen und die Umwelt darstellen. Ausfälle können entweder durch physikalische Fehler im Gerät (die z. B. zufällige Hardwareausfälle verursachen), durch systematische Fehler (z. B. kann menschliches Versagen bei der Spezifikation oder beim Entwurf systematische Ausfälle bei einer besonderen Kombination von Eingaben verursachen) oder durch Umgebungsbedingungen entstehen.

IEC 61508-1 liefert mit einem risikobasierten Ansatz den Gesamtrahmen zur Vermeidung und/oder Beherrschung von Ausfällen in elektromechanischen, elektronischen oder programmierbaren elektronischen Geräten.

Das Hauptziel ist es, sicherzustellen, dass die Anlage und die Betriebsmittel sicher automatisiert werden können. Ein zentrales Ziel dieser Norm ist die Verhinderung von Folgendem:

– Ausfälle von Steuerungssystemen, die andere Vorfälle auslösen, die wiederum zu Gefahren (z. B. Brand, Freisetzen giftiger Stoffe, Wiederanlauf einer Maschine usw.) führen können; und
– unerkannte Ausfälle in Schutzsystemen (z. B. in einem Notabschaltsystem), die die Systeme für den Fall eines notwendigen Sicherheitseingriffs unverfügbar machen.

IEC 61508-1 fordert, dass eine Gefährdungs- und Risikoanalyse auf Prozess-/Maschinenebene durchgeführt wird, um den für die Risikokriterien der Anwendung notwendigen Betrag der Risikominderung zu bestimmen. Das Risiko basiert auf der Beurteilung sowohl der Auswirkung (oder Schwere) als auch der Häufigkeit (oder Wahrscheinlichkeit) des gefährlichen Vorfalls.

IEC 61508-1 fordert weiterhin, dass der durch die Risikoanalyse festgestellte Betrag der Risikominderung zur Feststellung verwendet wird, ob ein oder mehrere sicherheitsbezogene Systeme [1)] notwendig sind und für welche Sicherheitsfunktionen (jede mit einer spezifizierten Sicherheitsintegrität) [2)] diese gebraucht werden.

IEC 61508-2 und IEC 61508-3 nehmen die Sicherheitsfunktionen und Anforderungen an die Sicherheitsintegrität auf, die jedem System durch die Anwendung der IEC 61508-1 zugeordnet worden sind und das als sicherheitsbezogenes E/E/PE-System bezeichnet wird, und legen die Anforderungen an die Tätigkeiten im Sicherheitslebenszyklus fest, die:

– während der Spezifikation, des Entwurfs und der Modifikation der Hardware und Software angewendet werden müssen; und

---

1) Systeme, die für die funktionale Sicherheit notwendig sind und ein oder mehrere elektrische (elektromechanische), elektronische oder programmierbare elektronische (E/E/PE) Geräte beinhalten, werden als sicherheitsbezogenes E/E/PE-Systeme bezeichnet und schließen alle Betriebsmittel ein, die zur Ausführung der erforderlichen Sicherheitsfunktion (siehe IEC 61508-4, 3.5.1) notwendig sind.

2) Die Sicherheitsintegrität ist als eine von vier diskreten Stufen festgelegt. Der Sicherheits-Integritätslevel 4 ist der höchste und Sicherheits-Integritätslevel 1 der niedrigste (siehe IEC 61508-4, 3.5.4 und 3.5.8).

– sich auf Mittel zur Verhinderung und/oder Beherrschung zufälliger Hardware- und systematischer Ausfälle (dem Sicherheitslebenszyklus des E/E/PE-Systems und dem Software-Sicherheitslebenszyklus)[3] konzentrieren.

IEC 61508-2 und IEC 61508-3 geben keine Anleitung, welcher Sicherheits-Integritätslevel für ein gegebenes erforderliches tolerierbares Risiko angemessen ist. Diese Entscheidung hängt von vielen Faktoren ab, einschließlich der Art der Anwendung, dem Maß, in dem andere Systeme Sicherheitsfunktionen ausführen, und von gesellschaftlichen und wirtschaftlichen Gegebenheiten (siehe IEC 61508-1 und IEC 61508-5).

Die Anforderungen der IEC 61508-2 und der IEC 61508-3 beinhalten:

– die Anwendung von Maßnahmen und Verfahren [4], abgestuft gegen den Sicherheits-Integritätslevel, zur Vermeidung systematischer Ausfälle [5] durch vorbeugende Methoden, und

– die Beherrschung systematischer Ausfälle (einschließlich Softwarefehler) und zufälliger Hardwareausfälle durch Entwurfsmerkmale wie Fehleraufdeckung, Redundanz und Architekturmerkmale (z. B. Diversität).

In IEC 61508-2 beruht die Zusicherung, dass das Ziel der Sicherheitsintegrität für gefahrbringende zufällige Hardwareausfälle erreicht worden ist, auf:

– den Anforderungen zur Hardwarefehlertoleranz (siehe IEC 61508-2, Tabellen 2 und 3), und

– dem Diagnosedeckungsgrad und der Häufigkeit von Wiederholungsprüfungen von Teilsystemen und Bauteilen, indem eine Zuverlässigkeitsanalyse unter Verwendung angemessener Daten ausgeführt wird.

Sowohl in IEC 61508-2 als auch in IEC 61508-3 wird die Gewissheit, dass das Ziel der Sicherheitsintegrität für systematische Ausfälle erreicht worden ist, erzielt durch:

– die korrekte Anwendung von Sicherheitsmanagementverfahren,

– den Einsatz fachkundigen Personals,

– die Durchführung der im Sicherheitslebenszyklus festgelegten Tätigkeiten einschließlich der festgelegten Verfahren und Maßnahmen [6], und

– eine unabhängige Beurteilung der funktionalen Sicherheit [7].

Das Gesamtziel ist es, sicherzustellen, dass entsprechend dem Sicherheits-Integritätslevel verbliebene systematische Fehler keinen Ausfall des sicherheitsbezogenen E/E/PE-Systems verursachen.

---

[3] Um den Anforderungen dieser Norm eine eindeutige Struktur zu geben, wurde die Entscheidung getroffen, die Anforderungen unter Verwendung eines Modells des Entwicklungsprozesses zu ordnen, in dem die Stufen in einer definierten Reihenfolge mit geringer Iteration (manchmal auch Wasserfallmodell genannt) aufeinanderfolgen. Es sei jedoch hervorgehoben, dass jeder Ansatz für einen Lebenszyklus verwendet werden kann, vorausgesetzt, dass eine Aussage zur Gleichwertigkeit im Sicherheitsplan für das Projekt gemacht worden ist (siehe IEC 61508-1, Abschnitt 7).

[4] Die für jeden Sicherheits-Integritätslevel geforderten Verfahren und Maßnahmen sind in den Tabellen der Anhänge A und B von IEC 61508-2 und IEC 61508-3 aufgelistet.

[5] Systematische Ausfälle können gewöhnlich nicht quantifiziert werden. Ursachen hierfür sind u. a.: Spezifikations- und Entwurfsfehler in Hardware und Software, Ausfälle aufgrund der Umgebungsbedingungen (z. B. Temperatur) und betriebsbedingte Fehler (z. B. eine schlechte Schnittstelle).

[6] Andere als in dieser Norm festgelegte Maßnahmen sind ebenfalls annehmbar, wenn sie während der Planung der Sicherheit schriftlich begründet werden (siehe IEC 61508-1, Abschnitt 6).

[7] Eine unabhängige Beurteilung bedeutet nicht unbedingt eine Beurteilung durch Dritte (siehe IEC 61508-1, Abschnitt 8).

IEC 61508-2 ist entwickelt worden, um Anforderungen zum Erreichen der Sicherheitsintegrität der Hardware [8)] des sicherheitsbezogenen E/E/PE-Systems einschließlich Sensoren und Stellgliedern zur Verfügung zu stellen. Verfahren und Maßnahmen werden sowohl gegen zufällige als auch gegen systematische Hardwareausfälle gefordert. Diese beinhalten, wie oben dargelegt, eine angemessene Kombination von Maßnahmen zu Vermeidung von Fehlern und zur Beherrschung von Ausfällen. Wird ein Bedienereingriff für die funktionale Sicherheit benötigt, werden Anforderungen an die Bedienerschnittstelle gestellt. In IEC 61508-2 werden ebenfalls auf Software und Hardware basierende Diagnosetestverfahren und Maßnahmen (zum Beispiel Diversität) zur Aufdeckung zufälliger Hardwareausfälle festgelegt.

IEC 61508-3 ist entwickelt worden, um Anforderungen zum Erreichen der Sicherheitsintegrität der Software, und zwar sowohl der Embedded-Software (einschließlich Dienste für Fehlererkennung durch Diagnose) als auch der Anwendungssoftware, zur Verfügung zu stellen. IEC 61508-3 fordert eine Kombination von Ansätzen zur Fehlervermeidung (Qualitätssicherung) und Fehlertoleranz (Softwarearchitektur), da es keinen bekannten Weg zum Nachweis der Fehlerfreiheit in einigermaßen komplexer sicherheitsbezogener Software, besonders der Freiheit von Spezifikations- und Entwurfsfehlern, gibt. IEC 61508-3 fordert die Anwendung von Software-Entwurfsprinzipien wie Top-Down-Entwurf, Modularität, Verifikation jeder Phase des Entwicklungslebenszyklus, verifizierte Softwaremodule und Software-Modulbibliotheken und verständliche Dokumentation, um die Verifikation und Validierung zu erleichtern. Die verschiedenen Stufen der Software erfordern verschiedene Stufen an Zusicherung, dass diese und ähnliche Prinzipien richtig angewendet worden sind.

Der Softwareentwickler darf, muss aber nicht, unabhängig von der Organisation sein, die die gesamte Entwicklung des E/E/PE-Systems durchführt. In jedem Fall ist eine enge Zusammenarbeit notwendig, insbesondere bei der Entwicklung der Architektur der programmierbaren Elektronik, wo die Abwägungen zwischen Hardware- und Softwarearchitektur im Hinblick auf die Auswirkungen auf die Sicherheit (siehe IEC 61508-2, Bild 4) durchgeführt werden müssen.

A.2 Funktionale Schritte bei der Anwendung der IEC 61508-2

Die funktionalen Schritte zur Anwendung der IEC 61508-2 werden in den Bildern A.1 und A.2 gezeigt. Die funktionalen Schritte zur Anwendung der IEC 61508-3 werden in Bild A.3 dargestellt.

Die funktionalen Schritte der IEC 61508-2 (siehe Bilder A.1 und A.2) sind wie folgt:

a) Erhalt der Zuordnung der Anforderungen an die Sicherheit (siehe IEC 61508-1). Angemessene Aktualisierung der Planung der Sicherheit während der Entwicklung des sicherheitsbezogenen E/E/PE-Systems.

b) Bestimmung der Anforderungen an sicherheitsbezogene E/E/PE-Systeme, einschließlich der Anforderungen an die Sicherheitsintegrität für jede Sicherheitsfunktion (siehe IEC 61508-2, 7.2). Zuweisung der Anforderungen an die Software und Übergabe an die Softwarelieferanten und/oder Entwickler zum Zwecke der Anwendung der IEC 61508-3.

ANMERKUNG 1 An dieser Stelle ist es notwendig, die Möglichkeit gleichzeitiger Ausfälle im EUC-Leit- und Steuersystem und sicherheitsbezogenen E/E/PE-System(en) zu berücksichtigen (siehe IEC 61508-5, A.5.4). Diese können aus Bauteilausfällen infolge gemeinsamer Ursache, zum Beispiel gleichartigen Umgebungseinflüssen, herrühren. Bei ungenügender Berücksichtigung kann das Vorhandensein solcher Ausfälle zu einem höheren als dem erwarteten Restrisiko führen.

c) Start der Phase für die Planung der Validierung der Sicherheit des sicherheitsbezogenen E/E/PE-Systems (siehe IEC 61508-2, 7.3).

d) Spezifikation der Architektur (Konfiguration) für das sicherheitsbezogene E/E/PE-Logik-Teilsystem, die Sensoren und Stellglieder. Überprüfung der Hardware- und Softwarearchitektur und der sicherheitstechnischen Auswirkungen der Abwägungen zwischen Hardware und Software (siehe IEC 61508-2, Bild 4) mit dem Softwarelieferanten/Entwickler. Falls notwendig, Wiederholung dieses Schrittes.

e) Entwicklung eines Modells für die Hardwarearchitektur des sicherheitsbezogenen E/E/PE-Systems. Entwicklung dieses Modell durch getrennten Test jeder Sicherheitsfunktion und Bestimmung jedes Teilsystems (Bauteils), das zur Ausführung dieser Funktion verwendet wird.

---

8) Einschließlich fest eingebauter Software oder softwareähnlicher Objekte (auch Firmware genannt), wie z. B. anwendungsspezifische integrierte Schaltkreise.

f) Festlegung der Systemparameter für jedes der Teilsysteme (Bauteile), die in dem sicherheitsbezogenen E/E/PE-System verwendet werden. Für jedes Teilsystem (Elemente) wird Folgendes bestimmt:
   – das Intervall für die Wiederholungsprüfung für Ausfälle, die nicht automatisch aufgedeckt werden;
   – die mittlere Dauer bis zur Wiederherstellung;
   – der Diagnosedeckungsgrad (siehe IEC 61508-2, Anhang C);
   – die Ausfallwahrscheinlichkeit;
   – die Einschränkungen hinsichtlich der Architektur für Route 1_H, siehe IEC 61508-2, 7.4.4.2 und Anhang C, und für Route 2_H, siehe IEC 61508-2, 7.4.4.3.

g) Bildung eines Zuverlässigkeitsmodells für jede Sicherheitsfunktion, die das sicherheitsbezogene E/E/PE-System ausführen muss.

ANMERKUNG 2 Ein Zuverlässigkeitsmodell ist eine mathematische Formel, die die Beziehung zwischen der Zuverlässigkeit und den betreffenden Parametern der Einrichtung und der Einsatzbedingungen zeigt.

h) Berechnung einer Zuverlässigkeitsvoraussage für jede Sicherheitsfunktion unter Verwendung eines angemessenen Verfahrens. Vergleich des Ergebnisses mit dem oben in b) festgelegten Ausfallgrenzwert und den Anforderungen der Route 1_H (siehe IEC 61508-2, 7.4.4.2) oder Route 2_H (siehe IEC 61508-2, 7.4.4.3). Falls die vorausgesagte Zuverlässigkeit den Ausfallgrenzwert nicht erfüllt und/oder nicht den Anforderungen von Route 1_H oder Route 2_H entspricht, dann wird Folgendes geändert:
   – wenn möglich, einen oder mehrere der Teilsystem-Parameter (zurückgehen zu f) oben); und/oder
   – die Hardwarearchitektur (zurückgehen zu d) oben).

ANMERKUNG 3 Eine Anzahl von Methoden zur Modellierung steht zur Verfügung, und der Analytiker sollte wählen, welche die geeignetste ist (siehe Anhang B für Hinweise zu einigen Methoden, die verwendet werden können).

i) Implementierung des Entwurfs des sicherheitsbezogenen E/E/PE-Systems. Auswahl von Maßnahmen und Verfahren, um systematische Hardwareausfälle, Ausfälle durch Umwelteinflüsse und Bedienfehler (siehe IEC 61508-2, Anhang A) zu beherrschen.

j) Integration der verifizierten Software (siehe IEC 61508-3) in die Zielhardware (siehe IEC 61508-2, 7.5 und IEC 61508-2, Anhang B) und parallel dazu die Entwicklung der Verfahren, denen Anwender und Instandhaltungspersonal beim Betrieb des Systems zu folgen haben (siehe IEC 61508-2, 7.6 und Anhang B). Berücksichtigung der Softwareaspekte (siehe A.3 f)).

k) Validierung des E/E/PE-Systems (siehe IEC 61508-2, 7.7 und IEC 61508-2, Anhang B) zusammen mit dem Softwareentwickler (siehe IEC 61508-3, 7.7).

l) Übergabe der Hardware und der Ergebnisse der Validierung der Sicherheit des sicherheitsbezogenen E/E/PE-Systems an die Systemingenieure für die weitere Integration in das Gesamtsystem.

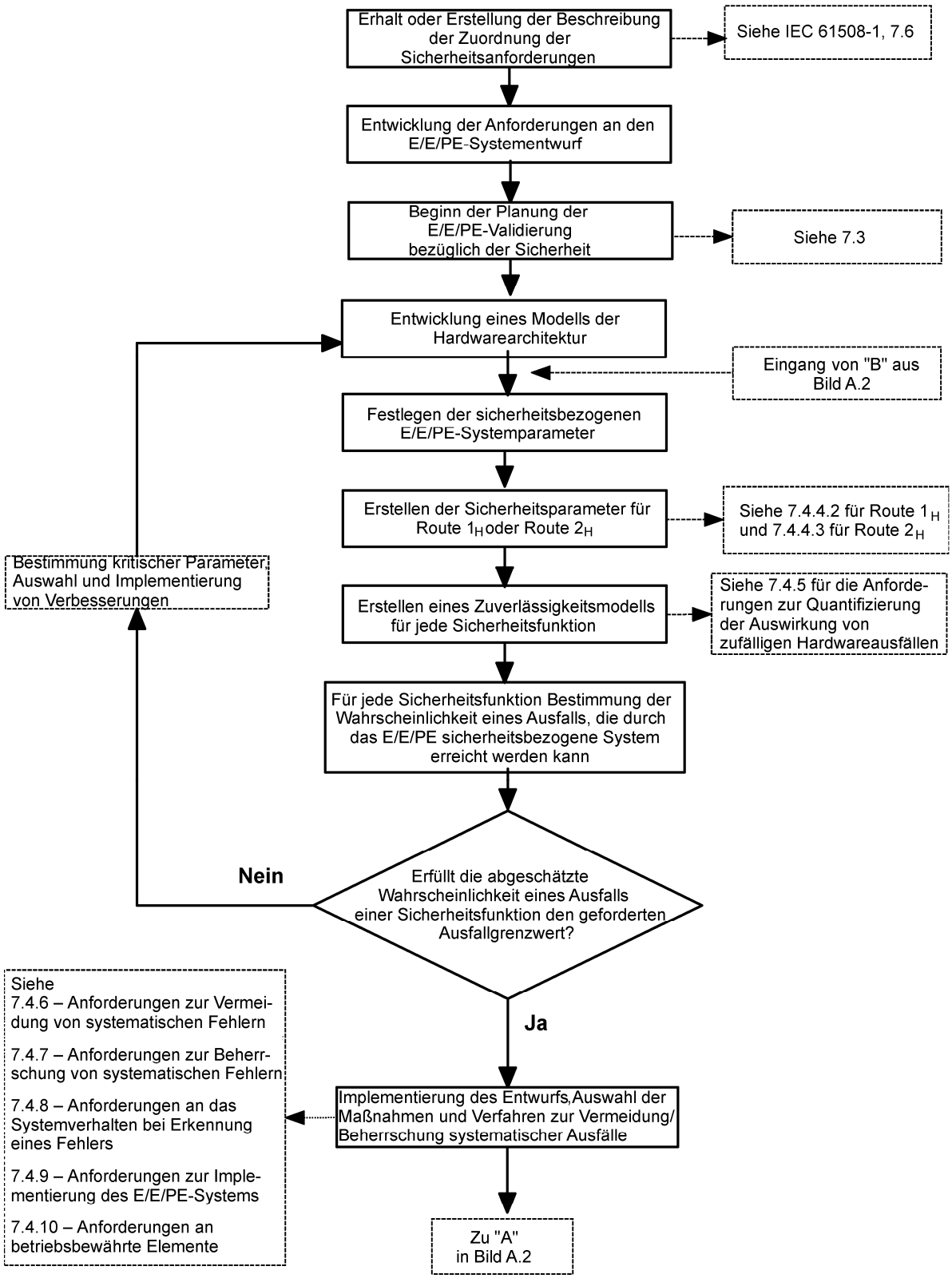
m) Falls eine Instandhaltung/Modifikation des sicherheitsbezogenen E/E/PE-Systems während der Betriebszeit notwendig wird, erneute Anwendung der IEC 61508-2, sofern angebracht (siehe IEC 61508-2, 7.8).

Einige Tätigkeiten finden im gesamten Sicherheitslebenszyklus des sicherheitsbezogenen E/E/PE-Systems statt. Darunter sind die Verifikation (siehe IEC 61508-2, 7.9) und die Beurteilung der funktionalen Sicherheit (siehe IEC 61508-1, Abschnitt 8).

Durch Anwendung der obigen Schritte werden sicherheitsrelevante Verfahren und Maßnahmen für das sicherheitsbezogene E/E/PE-System im Hinblick auf den erforderlichen Sicherheits-Integritätslevel ausgewählt. Zur Hilfestellung bei dieser Auswahl sind Tabellen mit der Einordnung der verschiedenen Verfahren/Maßnahmen unter die vier Sicherheits-Integritätslevel (siehe IEC 61508-2, Anhang B) formuliert worden. In den Tabellen wird für jedes Verfahren und jede Maßnahme auf eine Kurzbeschreibung mit Literaturhinweisen verwiesen (siehe IEC 61508-7, Anhänge A und B).

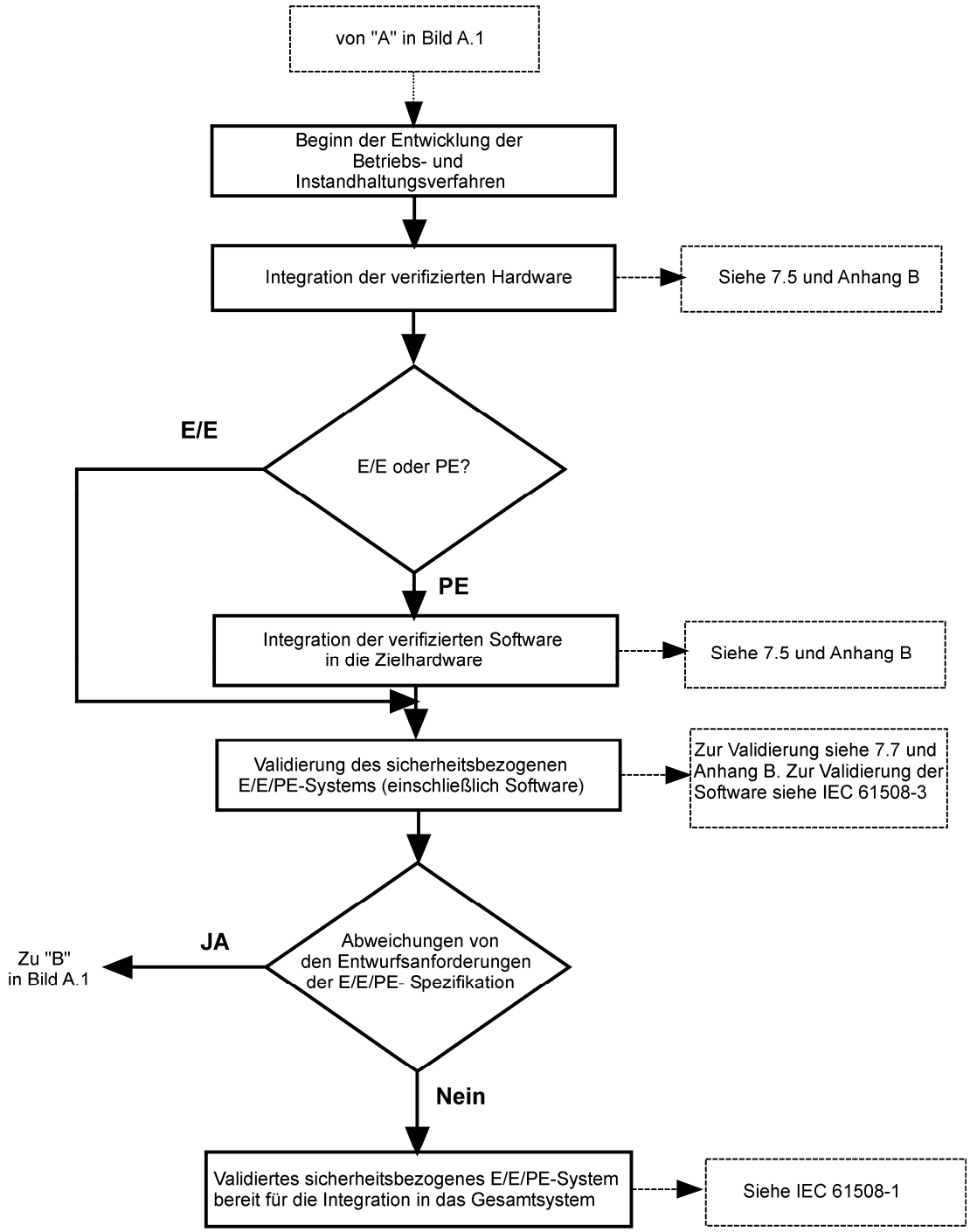
Anhang B stellt ein mögliches Verfahren zur Berechnung der Wahrscheinlichkeiten von Hardwareausfällen für sicherheitsbezogene E/E/PE-Systeme vor.

ANMERKUNG 4 Andere als in der Norm festgelegte Maßnahmen sind bei der Anwendung dieser Schritte ebenfalls annehmbar, wenn sie während der Planung der Sicherheit schriftlich begründet werden (siehe IEC 61508-1, Abschnitt 6).



ANMERKUNG Für sicherheitsbezogene PE-Systeme werden die Tätigkeiten für die Software gleichzeitig durchgeführt. (siehe Bild A.3).

Bild A.1 – Anwendung der IEC 61508-2



ANMERKUNG Für sicherheitsbezogene PE-Systeme werden die Tätigkeiten für die Software gleichzeitig durchgeführt. (siehe Bild A.3).

Bild A.2 – Anwendung der IEC 61508-2 (Fortsetzung von Bild A.1)

A.3 Funktionale Schritte bei der Anwendung der IEC 61508-3

Die funktionalen Schritte für die IEC 61508-3 (siehe Bild A.3) sind wie folgt:

a) Erhalt der Anforderungen an die sicherheitsbezogenen E/E/PE-Systeme und der zutreffenden Teile der Planung der Sicherheit (siehe IEC 61508-2, 7.3). Angemessene Aktualisierung der Planung der Sicherheit während der Softwareentwicklung.

ANMERKUNG 1 Frühere Lebenszyklusphasen haben bereits:

– die erforderlichen Sicherheitsfunktionen und ihre Sicherheits-Integritätslevel festgelegt (siehe IEC 61508-1, 7.4 und 7.5),

– die Sicherheitsfunktionen sicherheitsbezogenen E/E/PE-Systemen zugewiesen (siehe IEC 61508-1, 7.6) und

– Funktionen der Software innerhalb jedes sicherheitsbezogenen E/E/PE-Systems zugewiesen (siehe IEC 61508-2, 7.2).

b) Bestimmung der Softwarearchitektur für alle Sicherheitsfunktionen, die der Software zugewiesen worden sind (siehe IEC 61508-3, 7.4 und IEC 61508-3, Anhang A).

c) Überprüfung der Software und Hardwarearchitektur und der sicherheitstechnischen Auswirkungen der Abwägungen zwischen Software und Hardware (siehe IEC 61508-2, Bild 4) mit dem Lieferanten/Entwickler des sicherheitsbezogenen E/E/PE-Systems. Falls notwendig, Wiederholung dieses Schrittes.

d) Beginn der Planung für die Verifikation und Validierung der Sicherheit der Software (siehe IEC 61508-3, 7.3 und 7.9).

e) Entwurf, Entwicklung und Verifizieren/Testen der Software gemäß:

   – der Planung der Sicherheit der Software;

   – dem Software-Sicherheits-Integritätslevel; und

   – dem Software-Sicherheitslebenszyklus.

f) Vervollständigen der abschließenden Tätigkeiten der Softwareverifikation und Integration der verifizierten Software in die Zielhardware (siehe IEC 61508-3, 7.5) und parallel dazu Entwicklung der Softwareaspekte der Verfahren, denen Anwender und Instandhaltungspersonal beim Betrieb des Systems zu folgen haben (siehe IEC 61508-3, 7.6 und A.2 k)).

g) Validierung der Software in den integrierten sicherheitsbezogenen E/E/PE-Systemen (siehe IEC 61508-3, 7.7) zusammen mit dem Hardwareentwickler (siehe IEC 61508-2, 7.7).

h) Übergabe der Ergebnisse der Validierung der Sicherheit der Software an die Systemingenieure für die weitere Integration in das Gesamtsystem.

i) Falls eine Modifikation der Software des sicherheitsbezogenen E/E/PE-Systems während der Betriebszeit notwendig wird, erneute Anwendung dieser Phase der IEC 61508-3 in angemessener Weise (siehe IEC 61508-3, 7.8).

Einige Tätigkeiten finden im gesamten Software-Sicherheitslebenszyklus statt. Darunter sind die Verifikation (siehe IEC 61508-3, 7.9) und die Beurteilung der funktionalen Sicherheit (siehe IEC 61508-3, Abschnitt 8).

Durch Anwendung der obigen Schritte werden sicherheitsrelevante Verfahren und Maßnahmen für die Software im Hinblick auf den erforderlichen Sicherheits-Integritätslevel ausgewählt. Zur Hilfestellung bei dieser Auswahl sind Tabellen mit der Einordnung der verschiedenen Verfahren/Maßnahmen unter die vier Sicherheits-Integritätslevel (siehe IEC 61508-3, Anhang A) formuliert worden. In den Tabellen wird für jedes Verfahren und jede Maßnahme auf eine Kurzbeschreibung mit Literaturhinweisen verwiesen (siehe IEC 61508-7, Anhang C).

Ausgearbeitete Beispiele für die Anwendung der Tabellen zur Sicherheitsintegrität werden in Anhang E gegeben, und IEC 61508-7 beinhaltet einen probabilistischen Ansatz für die Bestimmung der Sicherheitsintegrität von bereits entwickelter Software (siehe IEC 61508-7, Anhang D).

ANMERKUNG 2 Andere als in dieser Norm festgelegte Maßnahmen sind bei der Anwendung dieser Schritte ebenfalls annehmbar, wenn sie während der Planung der Sicherheit schriftlich begründet werden (siehe IEC 61508-1, Abschnitt 6).

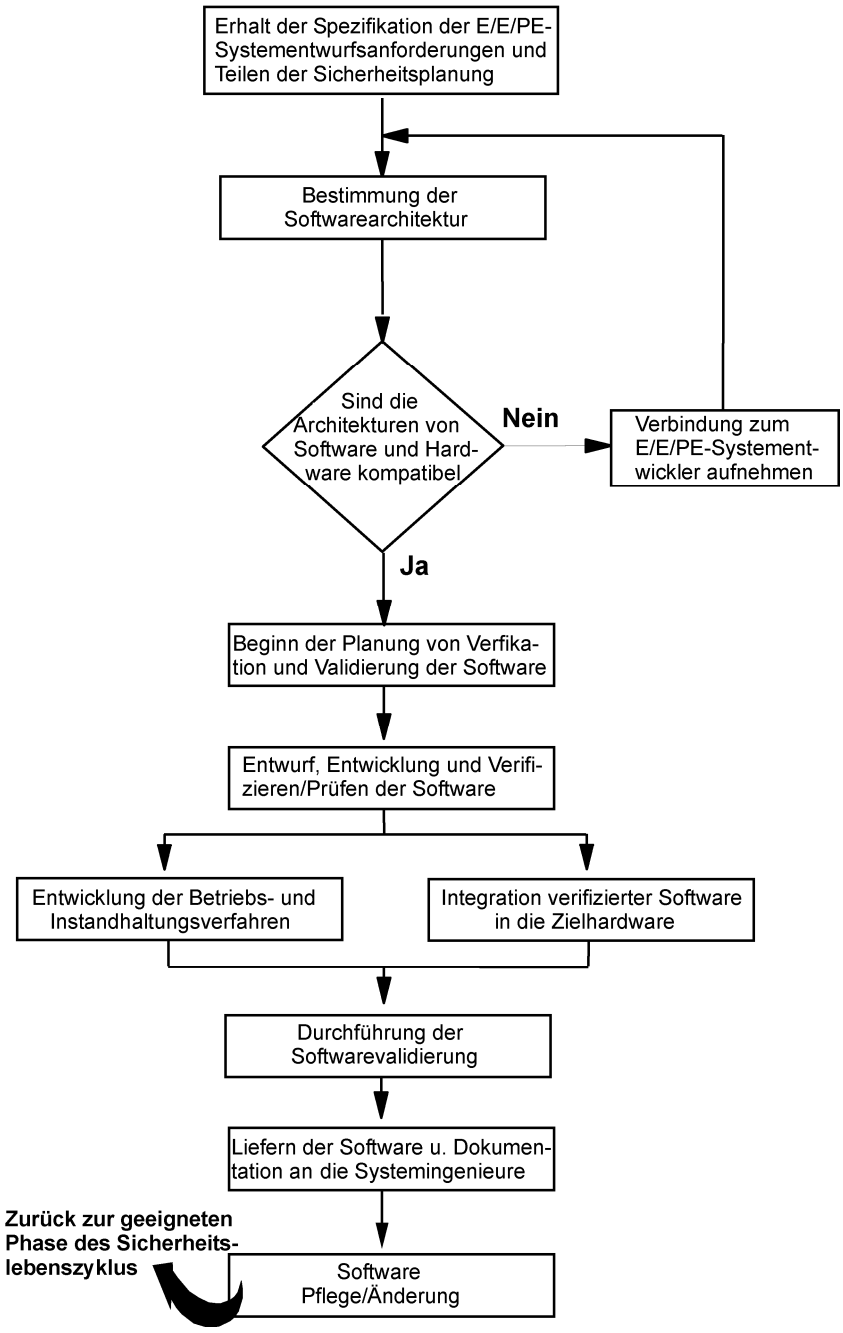


Bild A.3 – Anwendung der IEC 61508-3

Anhang B
(informativ)

Beispielverfahren für die Bewertung von Ausfallwahrscheinlichkeiten der Hardware

B.1 Allgemeines

Dieser Anhang stellt mögliche Verfahren für die Berechnung der Ausfallwahrscheinlichkeiten der Hardware von nach IEC 61508-1, IEC 61508-2 und IEC 61508-3 errichteten sicherheitsbezogenen E/E/PE-Systemen dar. Er ist informativ und sollte nicht als Darstellung der einzigen möglichen Bewertungsverfahren interpretiert werden. Er stellt jedoch sowohl einen verhältnismäßig einfachen Ansatz für die Beurteilung der Fähigkeiten von sicherheitsbezogenen E/E/PE-Systemen dar, als auch eine Anleitung für die Verwendung alternativer Verfahren, die von den herkömmlichen Methoden zur Zuverlässigkeitsberechnung abgeleitet sind.

ANMERKUNG 1 Die in diesem Teil gezeigten Systemarchitekturen stellen Beispiele dar und ihre Darstellung erhebt keinen Anspruch auf Vollständigkeit, da es viele andere Architekturen gibt, die verwendet werden können.

ANMERKUNG 2 Siehe ISA-TR84.00.02-2002 [17] in den Literaturhinweisen.

Einige Zuverlässigkeitsverfahren sind mehr oder weniger direkt für die Analyse der Sicherheitsintegrität der Hardware sicherheitsbezogener E/E/PE-Systeme anwendbar. Üblicherweise werden sie nach den folgenden zwei Gesichtspunkten unterteilt:

– Modelle: statische (boolesch) im Vergleich zu dynamischen (Zustände/Übergänge);
– Berechnungen: analytische im Vergleich zu Monte-Carlo-Simulationen.

Die Booleschen Modelle umfassen alle Modelle, welche die statischen logischen Verknüpfungen zwischen einfachen Ausfällen und Ausfällen, die das gesamte System betreffen, beschreiben. Zu den Booleschen Modellen gehören Zuverlässigkeitsblockdiagramme (RBD, en: reliability block diagram) (siehe IEC 61508-7, C.6.4 und IEC 61078 [4]) und Fehlerbäume (FTA) (siehe IEC 61508-7, B.6.6.5 und B.6.6.9 und IEC 61025 [18]).

Die Zustands-Übergangs-Modelle umfassen alle Modelle, die beschreiben, wie sich das System entsprechend der auftretenden Ereignisse (Ausfälle, Reparaturen, Tests usw.) verhält (Übergänge von einem in den anderen Zustand). Das Markov-Modell (siehe IEC 61508-7, B.6.6.6 und IEC 61165 [5]), Petri-Netze (siehe IEC 61508-7, B.2.3.3 und B.6.6.10 und IEC 62551 [19]) und formale Sprachmodelle gehören zu den Zustands-Übergangs-Modellen. Es werden zwei Ansätze des Markov-Modells untersucht: ein vereinfachter Ansatz basierend auf spezifischen Formeln (B.3) und ein allgemeiner Ansatz, der Berechnungen unmittelbar an Graphen des Markov-Modells (B.5.2) ermöglicht. Für Sicherheitssysteme, die nicht nach Markov modelliert werden können, können stattdessen Monte-Carlo-Simulationen verwendet werden. Mit heutigen PCs ist dies selbst für SIL 4-Berechnungen erreichbar. Die Unterabschnitte B.5.3 und B.5.4 dieses Anhangs enthalten Anleitungen zur Anwendung der Monte-Carlo-Simulation (siehe IEC 61508-7, B.6.6.8) auf Verhaltensmodelle basierend auf Petri-Netzen und formalen Sprachmodellen.

Der vereinfachte Ansatz, der zuerst vorgestellt wird, basiert auf grafischen RBD-Darstellungen und spezifischen Formeln des Markov-Modells, erhalten aus Taylor-Reihen und den zugrunde liegenden, leicht konservativen in B.3.1 beschriebenen Hypothesen.

All diese Verfahren können für die Mehrzahl von sicherheitsbezogenen Systeme verwendet werden und wenn für jede einzelne Anwendung entschieden wird, welches Verfahren angewendet wird, ist es sehr wichtig, dass der Anwender mit der Anwendung des Verfahrens vertraut ist. Dies kann wichtiger sein als die Wahl des Verfahrens selbst. Es liegt in der Verantwortung des Analytikers nachzuweisen, dass die zugrunde liegenden Hypothesen jeder einzelnen Methode erfüllt sind oder ob Anpassungen erforderlich sind, um ein ausreichend realistisches, konservatives Ergebnis zu erhalten. Im Fall schlechter Zuverlässigkeitsdaten oder eines dominanten Ausfalls infolge gemeinsamer Ursache kann die Anwendung der einfachsten Modelle/

Verfahren ausreichend sein. Ob der Genauigkeitsverlust maßgeblich ist, kann nur unter den jeweiligen Umständen ermittelt werden.

Wenn für die Durchführung der Berechnung Softwareprogramme verwendet werden, muss der in der Praxis Tätige die von dem Softwarepaket verwendeten Formeln/Verfahren verstehen, um sicherzustellen, dass deren Verwendung für die bestimmte Anwendung ausreichend ist. Er sollte außerdem durch einige manuell berechnete Testfälle das Ergebnis des Softwareprogramms überprüfen.

Stellt ein Ausfall des EUC-Leit- oder Steuerungssystems eine Anforderung an das sicherheitsbezogene E/E/PE-System, hängt die Wahrscheinlichkeit eines gefährlichen Vorfalls auch von der Wahrscheinlichkeit eines Ausfalls des EUC-Leit- oder Steuerungssystems ab. In dieser Situation ist es notwendig, die Möglichkeit gleichzeitiger Ausfälle von Komponenten des EUC-Leit- oder Steuerungssystems und des sicherheitsbezogenen E/E/PE-Systems aufgrund von Ausfallmechanismen mit gemeinsamer Ursache zu betrachten. Bei unzureichender Berücksichtigung können solche Ausfälle zu einem höheren als dem erwarteten Restrisiko führen.

B.2 Betrachtungen zu grundlegenden Wahrscheinlichkeitsberechnungen

B.2.1 Einleitung

Das Zuverlässigkeitsblockdiagramm (RBD) in Bild B.1 zeigt eine PLT-Schutzeinrichtung mit drei Sensoren (A, B, C), einem Logiksystem (D), zwei Stellgliedern (E, F) und Ausfällen infolge gemeinsamer Ursachen (CCF).

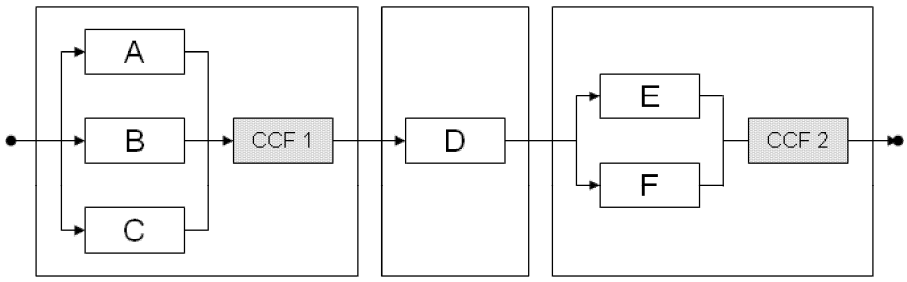


Bild B.1 – Zuverlässigkeitsblockdiagramm einer vollständigen PLT-Schutzeinrichtung

Dies ermöglicht die Identifikation von fünf Fehlerkombinationen, die zum Ausfall des sicherheitsbezogenen E/E/PE-Systems führen. Jedes ist ein sogenannter Minimalschnitt (en: minimal cut set):

– (A, B, C) ist ein Dreifachausfall,
– (E, F) ist ein Doppelausfall,
– (D), (CCF1), (CCF2) sind Einzelausfälle.

B.2.2 Sicherheitsbezogenes E/E/PE-System mit niedriger Anforderungsrate

Wenn ein sicherheitsbezogenes E/E/PE-System in der Betriebsart mit niedriger Anforderungsrate verwendet wird, verlangt die Norm die Beurteilung dessen PFD_{avg} (d. h. dessen mittlerer Nichtverfügbarkeit). Diese ist schlicht das Verhältnis $MDT(T)/T$, wobei $MDT(T)$ die mittlere Ausfalldauer des sicherheitsbezogenen E/E/PE-Systems über den Zeitraum [0, T] ist.

Die Wahrscheinlichkeit des Ausfalls ist für ein Sicherheitssystem in der Regel sehr gering und die Wahrscheinlichkeit des gleichzeitigen Auftretens zweier Minimalschnitte ist vernachlässigbar. Daher ergibt die Summe der mittleren Ausfalldauern infolge der einzelnen Minimalschnitte eine konservative Schätzung der mittleren Ausfalldauer des Gesamtsystems. Aus Bild B.1 kann entnommen werden:

$$MDT \approx MDT^{ABC} + MDT^{D} + MDT^{EF}$$

Dividiert durch T ergibt dies:

$$PFD_{avg} \approx PFD_{avg}^{ABC} + PFD_{avg}^{D} + PFD_{avg}^{EF}$$

Daher sind die Berechnungen von PFD_{avg} für in Reihe geschaltete Teile sehr ähnlich denen, die mit gewöhnlichen Wahrscheinlichkeiten durchgeführt werden, wenn diese sehr klein im Vergleich zu 1 sind.

Für parallel angeordnete Teile, wo erst bei Mehrfachausfällen etwa von (E, F) ein Funktionsverlust eintritt, ist jedoch klar, dass MDT^{EF} [NA1)] nicht direkt aus MDT^{E} und MDT^{F} berechnet werden kann. Die *MDT* des (E, F)-Systems wird wie folgt berechnet:

$$MDT^{EF} = \int_{0}^{T} PFD^{E}(t) PFD^{F}(t) dt$$

Daher sind die gewöhnlichen Wahrscheinlichkeitsberechnungen (Additionen und Multiplikationen) für die PFD_{avg}-Berechnungen (Integrale) von parallel geschalteten Teilen nicht gültig. PFD_{avg} hat nicht die gleichen Eigenschaften einer echten Wahrscheinlichkeit und die Anpassung daran würde voraussichtlich nicht zu einem konservativen Ergebnis führen. Insbesondere ist es nicht möglich, nur durch das Zusammenfassen der $PFD_{avg,i}$ seiner Komponenten auf konventionelle Weise die PFD_{avg} eines sicherheitsbezogenen E/E/PE-Systems zu erhalten. Da dies zuweilen von den käuflichen booleschen Softwareprogrammen so unterstützt wird, sollten die Analytiker aufmerksam sein, um solche nicht konservativen Berechnungen, welche im Umgang mit der Sicherheit unerwünscht sind, zu verhindern.

BEISPIEL Für einen redundanten (1oo2) Kanal mit einer Rate gefahrbringender unerkannter Ausfälle λ_{DU} und einem Intervall τ für die Wiederholungsprüfung könnte eine ungenaue Wahrscheinlichkeitsberechnung $(\lambda_{DU}.\tau)^2/4$ ergeben, bei einem tatsächlichen Ergebnis von $(\lambda_{DU}.\tau)^2/3$.

Die Berechnungen können analytisch oder unter Verwendung von Monte-Carlo-Simulation durchgeführt werden. Dieser Anhang beschreibt die Ausführung unter Verwendung konventioneller Zuverlässigkeitsmodelle basierend auf booleschen Modellen (RBD oder Fehlerbaum) oder Zustands-Übergangs-Modellen (Markov, Petri-Netze, usw.)

B.2.3 Sicherheitsbezogenes E/E/PE-System mit hoher Anforderungsrate oder kontinuierlicher Anforderung

B.2.3.1 Allgemeine *PFH*-Formel

Wenn ein sicherheitsbezogenes E/E/PE-System in der Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung verwendet wird, verlangt die Norm die Berechnung seiner *PFH* (d. h. der mittleren Häufigkeit eines gefahrbringenden Ausfalls). Dies ist der Mittelwert der sogenannten unbedingten Ausfallintensität $w(t)$ (auch Ausfallhäufigkeit genannt) über den betreffenden Zeitraum:

$$PFH(T) = \frac{1}{T} \int_{0}^{T} w(t) dt$$

Arbeitet das sicherheitsbezogene E/E/PE-System in der Betriebsart mit kontinuierlicher Anforderung und bildet es die allerletzte Barriere für die Sicherheit, dann führt der Gesamtausfall des sicherheitsbezogenen E/E/PE-Systems unmittelbar zu einer möglicherweise gefährlichen Situation. Deshalb kann für Ausfälle, die den Verlust der Gesamtsicherheitsfunktion verursachen, eine Reparatur des sicherheitsbezogenen Gesamtsystems in der Berechnung nicht berücksichtigt werden. Wenn der Ausfall des sicherheitsbezogenen Gesamtsystems jedoch aufgrund einer anderen Sicherheitsbarriere oder des Ausfalls einer Einrichtung nicht unmittelbar zu einer möglichen Gefahr führt, dann ist es möglich, die Erkennung und die Reparatur des sicherheitsbezogenen Systems in seiner Berechnung der Risikominderung zu berücksichtigen.

[NA1)] Nationale Fußnote: Für die Bedeutung der Indizes siehe DIN EN 61508-4 (VDE 0803-4):2011, 3.6.15.

B.2.3.2 Unzuverlässigkeit (z. B. eine einzelne Barriere arbeit in der Betriebsart mit kontinuierlicher Anforderung)

Dieser Fall ist von Bedeutung, wenn das sicherheitsbezogene E/E/PE-System, welches in der Betriebsart mit kontinuierlicher Anforderung arbeitet, die allerletzte Barriere für die Sicherheit bildet. Sobald es ausfällt, kann eine mögliche Gefährdung eintreten. Es sind keine Ausfälle des Gesamtsystems über den betreffenden Zeitraum zulässig.

In diesem Fall kann die *PFH* mit Hilfe der Unzuverlässigkeit über den betreffenden Zeitraum berechnet werden.

$$F(T):\ PFH(T) = \frac{1 - \exp\left[-\int_0^T \Lambda(t)dt\right]}{T} = \frac{F(T)}{T}$$

Die Ausfallrate $\Lambda(t)$ des Gesamtsystems kann zeitabhängig oder konstant sein.

Bei Zeitabhängigkeit gilt: $PFH(T) = \frac{1 - \exp(-\Lambda_{avg}.T)}{T} \approx \Lambda_{avg}$

Wenn das System aus vollständig und schnell reparierbaren Bauteilen mit konstanten Ausfall- und Reparaturraten besteht (z. B. gefahrbringende erkannte Ausfälle), erreicht $\Lambda(t)$ schnell einen konstanten asymptotischen Wert Λ_{as} und, falls $PFH(T) << 1$, gilt:

$$PFH(T) = \frac{1 - \exp(-\Lambda_{as}.T)}{T} \approx \Lambda_{as} = \frac{1}{MTTF}$$

Λ_{as} ist nur vorhanden, wenn das sicherheitsbezogene E/E/PE-System in der Betriebsart mit kontinuierlicher Anforderung arbeitet und nur sichere und DD-Ausfälle (d. h. schnell erkannt und repariert) auftreten. Reparaturen von Ausfällen, die unmittelbare Ursache des Gesamtausfalls der Sicherheitsfunktion sein können, können nicht berücksichtigt werden. Bei einer redundanten Konfiguration, bei der es darauf ankommt, die Wiederholungsprüfungen zu berücksichtigen, ist die asymptotische Ausfallrate nicht von Bedeutung und es sind die vorhergehenden Gleichungen anzuwenden. Es ist die Aufgabe des Analysten, zu bestimmen, welcher Fall zutreffend ist.

B.2.3.3 Nichtverfügbarkeit (z. B. mehrere Sicherheitsbarrieren)

Arbeitet das sicherheitsbezogene E/E/PE-System in der Betriebsart mit kontinuierlicher Anforderung und bildet es nicht die allerletzte Barriere, erhöhen dessen Ausfälle lediglich die Anforderungshäufigkeit auf die anderen Sicherheitsbarrieren, oder arbeitet es in der Betriebsart mit hoher Anforderungsrate, so ist es möglich, einen Fehler, der unmittelbar die Ursache für den Verlust der Sicherheitsfunktion sein könnte, innerhalb des erwarteten Anforderungszeitraum zu erkennen (automatisch oder manuell) und zu reparieren. Unter diesen Umständen kann dessen Gesamtausfall behoben werden, und die *PFH* kann aus der Verfügbarkeit $A(t)$ und aus der bedingten Ausfallintensität $\Lambda_v(t)$ des Systems berechnet werden.

Besteht das System wiederum aus Komponenten, die vollständig und schnell zu reparieren sind (d. h. wenn es in allen sich verschlechternden Situationen eine hohe Wahrscheinlichkeit gibt, schnell wieder in einen funktionierenden Zustand zu gelangen), erreicht $\Lambda_v(t)$ schnell den asymptotischen Wert Λ_{vas}, welcher darüber hinaus auch ein guter Schätzwert der in B.2.3.2 dargestellten wahren asymptotischen Ausfallrate Λ_{as} des Gesamtsystems ist.

Dies führt zu den folgenden Näherungen:

$$PFH = \frac{1}{MUT + MDT} = \frac{1}{MTBF} \approx \frac{1}{MUT} \approx \frac{1}{MTTF}$$

Dabei ist

MUT die Abkürzung für die mittlere ausfallfreie Betriebsdauer;

MDT die Abkürzung für die mittlere Ausfalldauer;

MTBF die Abkürzung für die mittlere Betriebsdauer zwischen Ausfällen; und

MTTF die Abkürzung für die mittlere Dauer bis zum Ausfall.

B.2.3.4 Betrachtungen zur Ausfallrate

Einige der obigen Formeln verwenden die Ausfallrate $\Lambda(t)$ des Gesamtsystems. Deren Berechnung ist nicht so einfach und erfordert einige Überlegungen.

In Reihe geschaltete Strukturen sind sehr leicht zu handhaben, da die Ausfallraten addiert werden können. Aus dem Bild B.1 können wir entnehmen: $\Lambda(t) = \Lambda^{abc}(t) + \lambda^{CCF1}(t) + \lambda^{d}(t) + \Lambda^{ef}(t) + \lambda^{CCF2}(t)$, wobei $\Lambda(t)$ die Gesamtausfallrate des sicherheitsbezogenen E/E/PE-Systems ist und $\Lambda^{abc}(t)$, $\lambda^{CCF1}(t)$, $\lambda^{d}(t)$, $\Lambda^{ef}(t)$ und $\lambda^{CCF2}(t)$ die Ausfallraten der fünf Minimalschnitte sind.

Für parallele Strukturen ist dies nicht so leicht, da keine einfachen Beziehungen zu den Ausfallraten der einzelnen Komponenten existieren. Als Beispiel dient die Betrachtung des Minimalschnittes (E, F):

1) Wenn E und F nicht unmittelbar ersetzt werden können (z. B. DU-Ausfälle), wechselt $\Lambda^{ef}(t)$ kontinuierlich von 0 zu λ (Ausfallrate von E oder F). Ein asymptotischer Wert wird erreicht, wenn eine der beiden Komponenten voraussichtlich ausgefallen sein wird. Dies ist ein sehr langsamer Prozess, da dies eintritt, wenn t größer als $1/\lambda$ wird. Dieser asymptotische Wert wird tatsächlich niemals erreicht, wenn E und F periodisch einer Wiederholungsprüfung mit einem Testintervall von $\tau << 1/\lambda$ unterzogen werden.

2) Wenn E und F in einer relativ kurzen Zeitspanne ersetzt werden (z. B. DD-Ausfälle), geht $\Lambda^{ef}(t)$ sehr schnell in einen asymptotischen Wert $\Lambda^{ef}{}_{AS} = 2\lambda^2/\mu$ über, der als eine entsprechende kontinuierliche Ausfallrate verwendet werden kann. Dieser wird erreicht, wenn t zwei- oder dreimal größer wird, als die größte *MTTR* der Komponenten. Dies ist ein spezieller Fall der vollständig und schnell reparierbaren Systeme, die oben behandelt wurden.

Das Abschätzen der Ausfallrate des Gesamtsystems setzt daher im Allgemeinen komplexere Berechnungen voraus, als für die einfacheren in Reihe geschaltete Strukturen.

B.3 Zuverlässigkeitsblockdiagramm-Ansatz mit angenommener konstanter Ausfallrate

B.3.1 Grundlegende Hypothese

Die Berechnungen basieren auf den folgenden Annahmen:

– die sich ergebende mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für das System ist kleiner als 10^{-1} oder die sich ergebende mittlere Häufigkeit eines gefahrbringenden Ausfalls des Systems ist kleiner als 10^{-5} h^{-1},

ANMERKUNG 1 Diese Annahme bedeutet, dass das sicherheitsbezogene E/E/PE-System im Anwendungsbereich der IEC 61508 liegt und sich innerhalb des SIL 1-Bandes befindet (siehe IEC 61508-1, Tabellen 2 und 3).

– die Bauteilausfallraten sind während der Lebensdauer des Systems konstant;

– das Sensor-(Eingangs-)Teilsystem besteht aus dem/den eigentlichen Sensor(en) und allen anderen Bauteilen und Verdrahtungen, bis hin zu denjenigen, in denen die Signale erstmalig durch Ausgangsvergleich oder eine andere Verarbeitung zusammengeführt werden, letztere aber nicht mit eingeschlossen (z. B. eine zweikanalige Sensorkonfiguration nach Bild B.2);

– das Logik-Teilsystem besteht aus dem/den Bauteil(en), in denen die Signale erstmalig zusammengeführt werden, sowie aus allen anderen Bauteilen bis hin zu denjenigen, die die Ausgangssignale dem Ausgangs-Teilsystem übergeben, letztere mit eingeschlossen;

– das Stellglied-(Ausgangs-)Teilsystem besteht aus allen Bauteilen und Verdrahtungen, die die Ausgangssignale des Logik-Teilsystems verarbeiten, die Stellglieder eingeschlossen;

– die Hardwareausfallraten, die als Eingangswerte für die Berechnungen und Tabellen herangezogen werden, gelten für einen Kanal eines Teilsystems (wenn z. B. 2oo3-Sensoren verwendet werden, wird die Ausfallrate eines einkanaligen Sensors genommen und die Auswirkung von 2oo3 gesondert berechnet);

– die Kanäle in einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider haben alle die gleichen Ausfallraten und Diagnosedeckungsgrade;

– die Gesamtausfallrate der Hardware eines Kanals eines Teilsystems ist die Summe der Rate der gefahrbringenden Ausfälle und der Rate der sicheren Ausfälle, die als gleich angenommen werden;

ANMERKUNG 2 Diese Annahme beeinflusst den Anteil sicherer Ausfälle (siehe IEC 61508-2, Anhang C), dieser beeinflusst aber nicht die berechneten Werte der Ausfallwahrscheinlichkeiten in diesem Anhang.

– für jede Sicherheitsfunktion gibt es eine vollständige Wiederholungsprüfung und Reparatur (d. h. alle Ausfälle, die unerkannt geblieben sind, werden bei der Wiederholungsprüfung erkannt), für den Fall einer unvollständigen Wiederholungsprüfung siehe jedoch B.3.2.5;

– das Intervall für die Wiederholungsprüfung ist mindestens um eine Zehnerpotenz größer als die MTR;

– für jedes Teilsystem gibt es genau ein Intervall für die Wiederholungsprüfung und die MTR;

– das erwartete Intervall zwischen den Anforderungen ist mindestens um eine Zehnerpotenz größer als das Intervall für die Wiederholungsprüfung;

– für alle Teilsysteme, die in der Betriebsart mit niedriger Anforderungsrate betrieben werden, und für 1oo2-, 1oo2D-, 1oo3- und 2oo3-Systeme mit Ausgangsvergleich oder Mehrheitsentscheider, die in der Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung betrieben werden, wird der durch den Diagnosedeckungsgrad festgelegte Anteil von Ausfällen innerhalb der MTTR sowohl erkannt als auch repariert. Diese Zeit wurde bei der Festlegung der Anforderungen an die Sicherheitsintegrität der Hardware benutzt;

BEISPIEL Falls eine MTTR von 8 h angenommen wird, schließt dies das Diagnosetestintervall ein, das typischerweise kleiner als 1 h ist, der Rest ist die MRT.

ANMERKUNG 3 Für 1oo2-, 1oo2D-, 1oo3- und 2oo3-Gruppen mit Ausgangsvergleich oder Mehrheitsentscheider wird angenommen, dass jede Reparatur bei laufendem Betrieb durchgeführt wird. Wenn ein sicherheitsbezogenes E/E/PE-System so konfiguriert wird, dass jeder erkannte Fehler die EUC-Einrichtung in einen sicheren Zustand versetzt, dann wird die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung verbessert. Der Grad der Verbesserung hängt vom Diagnosedeckungsgrad ab.

– für 1oo1- und 2oo2-Gruppen mit Ausgangsvergleich, die in der Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung betrieben werden, geht das sicherheitsbezogene E/E/PE-System nach Erkennen eines gefahrbringenden Fehlers immer in einen sicheren Zustand über. Um dies zu erreichen, ist das erwartete Intervall zwischen den Anforderungen mindestens um eine Zehnerpotenz größer als das Diagnosetestintervall oder die Summe der Diagnosetestintervalle, und die Zeit, um den sicheren Zustand zu erreichen, ist kleiner als die Prozess-Sicherheitszeit;

ANMERKUNG 4 Zur Prozess-Sicherheitszeit siehe IEC 61508-4, 3.6.20.

– wenn ein Ausfall der Energieversorgung ein sicherheitsbezogenes E/E/PE-System betrifft und das System beim Abschalten der Energie einen sicheren Zustand einnimmt, beeinflusst die Energieversorgung nicht die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung des sicherheitsbezogenen E/E/PE-Systems. Falls jedoch zum Einnehmen des sicheren Zustandes des Systems das Aufrechterhalten der Versorgung notwendig ist oder die Energieversorgung ein Ausfallverhalten hat, das einen unsicheren Betrieb des sicherheitsbezogenen E/E/PE-Systems verursachen kann, dann sollte die Energieversorgung in die Bewertung mit eingeschlossen werden;

– wenn der Ausdruck Kanal verwendet wird, beschränkt sich dies auf den betrachteten Teil des Systems, also üblicherweise auf das Sensor-, Logik- oder Stellglied-Teilsystem;

– die Abkürzungen der verwendeten Benennungen sind in Tabelle B.1 enthalten.

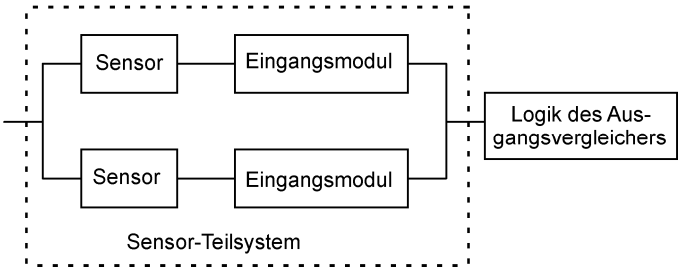


Bild B.2 – Beispielkonfiguration für zwei Kanäle mit Sensoren

Tabelle B.1 – In diesem Anhang verwendete Benennungen und ihre zugehörigen Parameterbereiche
(zutreffend für 1oo1-, 1oo2-, 2oo2-, 1oo2D-, 1oo3- und 2oo3-Architekturen)

Abkürzung	**Beschreibung (Einheit)**	**Parameterbereiche in den Tabellen B.2 bis B.5 und B.10 bis B.13**
T_1	Intervall der Wiederholungsprüfung (in Stunden)	Ein Monat (730 h) [1)] Drei Monate (2 190 h) [1)] Sechs Monate (4 380 h) Ein Jahr (8 760 h) Zwei Jahre (17 520 h) [2)] Zehn Jahre (87 600 h) [2)]
MTTR	Mittlere Dauer bis zur Wiederherstellung (en: Mean time to restoration) (in Stunden)	8 h ANMERKUNG: MTTR = MTR = 8 Stunden basiert auf der Annahme, dass die Dauer bis zur Erkennung eines gefahrbringenden Ausfalls bei einer automatischen Erkennung << MRT ist.
MRT	Mittlere Reparaturdauer (en: Mean repair time) (in Stunden)	8 h ANMERKUNG: MTTR = MTR = 8 Stunden basiert auf der Annahme, dass die Dauer bis zur Erkennung eines gefahrbringenden Ausfalls bei einer automatischen Erkennung << MRT ist.
DC	Diagnosedeckungsgrad (in Gleichungen als Bruch dargestellt, ansonst als Prozentwert)	0 % 60 % 90 % 99 %
β	Anteil unerkannter Ausfälle infolge gemeinsamer Ursache (in Gleichungen als Bruch dargestellt, ansonst als Prozentwert). (In den Tabellen B.2 bis B.5 und B.10 bis B.13 wird $\beta = 2 \times \beta_D$ angenommen.)	2 % 10 % 20 %
β_D	Der Anteil der Ausfälle infolge gemeinsamer Ursache, die durch den Diagnosetest erkannt werden (in Gleichungen als Bruch dargestellt, ansonst als Prozentwert). (In den Tabellen B.2 bis B.5 und B.10 bis B.13 wird $\beta = 2 \times \beta_D$ angenommen.)	1 % 5 % 10 %
λ_{DU} [NA2)]	Rate gefahrbringender unerkannter Ausfälle (je Stunde) eines Kanals in einem Teilsystem	$0{,}05 \times 10^{-6}$ $0{,}25 \times 10^{-6}$ $0{,}5 \times 10^{-6}$ $2{,}5 \times 10^{-6}$ 5×10^{-6} 25×10^{-6}

[NA2)] Nationale Fußnote: Die dargestellte Schreibweise λ_{DU} mit dem zweiten Indexbuchstaben als Großbuchstabe wird in diesem Teil und anderen Teilen der Norm nicht durchgängig verwendet. Teilweise wird Kleinschreibung verwendet (λ_{Du}), dies gilt gleichermaßen auch für λ_{DD} und λ_{SD}.

Tabelle B.1 *(fortgesetzt)*

Abkürzung	Beschreibung (Einheit)	Parameterbereich der Tabellen B.2 bis B.5 und B.10 bis B.13
PFD_G	Mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung einer Gruppe von Kanälen mit Ausgangsvergleicher oder Mehrheitsentscheider (en: *PFD* probability of failure on demand). (Wenn das Sensor-, Logik- oder Stellglied-Teilsystem aus nur einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider besteht, dann ist PFD_G gleich PFD_S, PFD_L oder PFD_{FE}.)	
PFD_S	Mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für das Sensor-Teilsystem	
PFD_L	Mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für das Logik-Teilsystem	
PFD_{FE}	Mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für das Stellglied-Teilsystem (en: final element subsystem)	
PFD_{SYS}	Mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für die Sicherheitsfunktion eines sicherheitsbezogenen E/E/PE-Systems	
PFH_G	Mittlere Häufigkeit eines gefahrbringenden Ausfalls für eine Gruppe von Kanälen mit Ausgangsvergleicher oder Mehrheitsentscheider. (Wenn das Sensor-, Logik- oder Stellglied-Teilsystem aus nur einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider besteht, dann ist PFH_G gleich PFH_S, PFH_L oder PFH_{FE}.)	
PFH_S	Mittlere Häufigkeit eines gefahrbringenden Ausfalls für das Sensor-Teilsystem	
PFH_L	Mittlere Häufigkeit eines gefahrbringenden Ausfalls für das Logik-Teilsystem	
PFH_{FE}	Mittlere Häufigkeit eines gefahrbringenden Ausfalls für das Stellglied-Teilsystem	
PFH_{SYS}	Mittlere Häufigkeit eines gefahrbringenden Ausfalls für die Sicherheitsfunktion eines sicherheitsbezogenen E/E/PE-Systems	
λ	Gesamtausfallrate (je Stunde) eines Kanals in einem Teilsystem	
λ_D	Rate gefahrbringender Ausfälle (je Stunde) eines Kanals eines Teilsystems, entspricht 0,5 λ (mit angenommenen 50 % gefahrbringender Ausfälle und 50 % sicherer Ausfälle, D = en: dangerous = gefahrbringend)	
λ_{DD}	Rate erkannter gefahrbringender Ausfälle (je Stunde) eines Kanals eines Teilsystems. (Dies ist die Summe aller Raten erkannter gefahrbringender Ausfälle innerhalb eines Kanals eines Teilsystems, DD = en: dangerous detected = gefahrbringend erkannt.)	
λ_{DU}	Rate unerkannter gefahrbringender Ausfälle (je Stunde) eines Kanals eines Teilsystems. (Dies ist die Summe aller Raten unerkannter gefahrbringender Ausfälle innerhalb eines Kanals eines Teilsystems, DU = en: dangerous undetected = gefahrbringend unerkannt.)	
λ_{SD}	Rate erkannter sicherer Ausfälle (je Stunde) eines Kanals eines Teilsystems. (Dies ist die Summe aller Raten erkannter sicherer Ausfälle innerhalb eines Kanals eines Teilsystems, SD = en: safe detected = sicher erkannt.)	
t_{CE}	Mittlere Ausfalldauer eines Kanals (in Stunden) für 1001, 1oo2-, 2oo2- und 2oo3-Architekturen. (Dies ist die gemeinsame Ausfalldauer für alle Komponenten in einem Kanal des Teilsystems.)	
t_{GE}	Mittlere Ausfalldauer einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider (in Stunden) für 1oo2- und 2oo3-Architekturen. (Dies ist die gemeinsame Ausfalldauer für alle Kanäle in einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider.)	
t_{CE}'	Mittlere Ausfalldauer (in Stunden) eines Kanals für eine 1oo2D-Architektur. (Dies ist die gemeinsame Ausfalldauer für alle Komponenten in einem Kanal des Teilsystems.)	
t_{GE}'	Mittlere Ausfalldauer einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider (in Stunden) für eine 1oo2D-Architektur. (Dies ist die gemeinsame Ausfalldauer für alle Kanäle in einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider.)	
T_2	Intervall zwischen den Anforderungen (h)	
K	Anteil des Erfolges der Eigentestschaltung im 1oo2D-System	
PTC	Deckungsgrad der Wiederholungsprüfung	

[1] Nur Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung.
[2] Nur Betriebsart mit niedriger Anforderungsrate.

B.3.2 Mittlere Ausfallwahrscheinlichkeit bei Anforderung (für die Betriebsart mit niedriger Anforderungsrate)

B.3.2.1 Verfahren für die Berechnung

Die mittlere Ausfallwahrscheinlichkeit bei Anforderung einer Sicherheitsfunktion eines sicherheitsbezogenen E/E/PE-Systems wird durch die Berechnung und Zusammenführung der mittleren Wahrscheinlichkeiten eines Ausfalls bei Anforderung aller Teilsysteme, die gemeinsam die Sicherheitsfunktion ausführen, bestimmt. Da die Wahrscheinlichkeiten in diesem Anhang klein sind, kann dies wie folgt ausgedrückt werden (siehe Bild B.3):

$$PFD_{SYS} = PFD_S + PFD_L + PFD_{FE}$$

Dabei ist

- PFD_{SYS} die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung einer Sicherheitsfunktion des sicherheitsbezogenen E/E/PE-Systems;
- PFD_S die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für das Sensor-Teilsystem;
- PFD_L die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für das Logik-Teilsystem; und
- PFD_{FE} die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für das Stellglied-Teilsystem.

Bild B.3 – Struktur mit Teilsystemen

Um für jedes der Teilsysteme die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung zu bestimmen, sollte das folgende Verfahren der Reihe nach für jedes Teilsystem angewendet werden:

a) Zeichnen des Blockdiagramms, das die Bauteile des Sensor-Teilsystems (Eingang), des Logik-Teilsystems und des Stellglied-Teilsystems (Ausgang) aufzeigt. Bauteile des Sensor-Teilsystems können z. B. Sensoren, Schutzschaltungen oder Schaltungen zur Signalkonditionierung sein, Bauteile des Logik-Teilsystems z. B. Prozessoren und Abtasteinrichtungen, Bauteile des Stellglied-Teilsystem z. B. Signalumformer, Schutzschaltungen und Aktoren. Darstellung jedes Teilsystems als eine oder mehrere 1oo1-, 1oo2-, 2oo2-, 1oo2D-, 1oo3- oder 2oo3-Gruppen mit Ausgangsvergleicher oder Mehrheitsentscheider.

b) Von den Tabellen B.2 bis B.5 wird die zutreffende Tabelle aufgesucht, je nachdem, ob ein Intervall für die Wiederholungsprüfung von sechs Monaten, einem Jahr, zwei Jahren oder 10 Jahren vorgesehen ist. Diese Tabellen nehmen außerdem eine mittlere Dauer bis zur Wiederherstellung von 8 h für jeden Ausfall an, wenn er aufgedeckt worden ist.

c) In der zutreffenden Tabelle für jede Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider im Teilsystem werden folgende Parameter aufgesucht:
   - Architektur (zum Beispiel 2oo3);
   - Diagnosedeckungsgrad jedes Kanals (zum Beispiel 60 %);
   - die Rate gefahrbringender Ausfälle (je Stunde) λ_D jedes Kanals (zum Beispiel $2{,}5 \times 10^{-06}$);
   - die β-Faktoren β und β_D für den Ausfall infolge gemeinsamer Ursachen, im Hinblick auf das Zusammenwirken der Kanäle in einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider (zum Beispiel 2 % bzw. 1 %).

ANMERKUNG 1 Es wird angenommen, dass jeder Kanal einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider den gleichen Diagnosedeckungsgrad und die gleiche Ausfallrate hat (siehe Tabelle B.1).

ANMERKUNG 2 In den Tabellen B.2 bis B.5 (und in den Tabellen B.10 bis B.13) wird angenommen, dass β, der β-Faktor ohne Diagnosetests (auch verwendet für bei Diagnosetests unerkannt bleibende gefahrbringende Ausfälle), das Zweifache von β_D, dem β-Faktor für durch Diagnosetests erkannte Ausfälle, beträgt.

d) Aus der zugehörigen Tabelle (B.2 bis B.5) wird die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für eine Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider entnommen.

e) Wenn die Sicherheitsfunktion von mehr als einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider, die Sensoren enthält, oder von mehr als einer derartigen Gruppe mit Ausgangsvergleicher, die Stellglieder enthält, abhängt, sind die mittleren Ausfallwahrscheinlichkeiten bei Anforderung des Sensor-Teilsystems PFD_S und des Stellglied-Teilsystems PFD_{FE} durch nachfolgende Gleichungen bestimmt. Hierbei sind PFD_{Gi} und PFD_{Gj} die mittleren Ausfallwahrscheinlichkeiten bei Anforderung für jede Gruppe von Sensoren bzw. Stellgliedern mit Ausgangsvergleicher oder Mehrheitsentscheider:

$$PFD_S = \sum_i PFD_{Gi}$$

$$PFD_{FE} = \sum_j PFD_{Gj}$$

Die in allen Gleichungen verwendeten Formeln, sowohl für *PFD* als auch die Systemausfallrate, sind alle eine Funktion von Bauteilausfallraten und mittlerer Ausfalldauer (*MDT*). Dort, wo mehrere Elemente in dem System vorhanden sind und es erforderlich ist, die Gesamt-*PFD* kombinierter Elemente oder die Ausfallrate des Systems zu berechnen, ist es oft notwendig, einen einzelnen Wert für *MDT* in den Gleichungen zu verwenden. Jedes Element kann jedoch über eine andere Ausfallerkennung mit unterschiedlichen *MDT* verfügen und verschiedene Elemente können unterschiedliche *MDT*-Werte für ein und denselben Ausfallmechanismus aufweisen, wobei es in diesem Fall notwendig wird, einen einzelnen Wert für die *MDT* zu ermitteln, der für alle Elemente im Pfad gültig ist. Dies kann erreicht werden, indem die Gesamtausfallrate der gesamten Pfade berücksichtigt wird und dann die einzelnen *MDTs* entsprechend ihres Ausfallratenanteils an der betrachteten Gesamtausfallrate aufgeteilt werden.

Wenn zum Beispiel zwei Elemente in Reihe geschaltet sind und eines einer Wiederholungsprüfung mit T_1 und das andere einer Wiederholungsprüfung mit T_2 unterworfen ist, dann beträgt der äquivalente Einzelwert für *MDT*:

$$\lambda_T = \lambda_1 + \lambda_2$$

und

$$MDT_E = \frac{\lambda_1}{\lambda_T}\left(\frac{T_1}{2}\right) + \frac{\lambda_2}{\lambda_T}\left(\frac{T_2}{2}\right)$$

B.3.2.2 Architekturen für die Betriebsart mit niedriger Anforderungsrate

ANMERKUNG 1 Dieser Unterabschnitt sollte der Reihe nach gelesen werden, da Gleichungen, die für mehrere Architekturen gültig sind, nur bei ihrer ersten Verwendung angegeben werden.

ANMERKUNG 2 Die Gleichungen basieren auf den in B.3.1 aufgeführten Annahmen.

ANMERKUNG 3 Die folgenden Beispiele stellen typische Konfigurationen dar, die Darstellung erhebt keinen Anspruch auf Vollständigkeit.

B.3.2.2.1 1oo1

Diese Architektur besteht aus einem einzelnen Kanal, bei ihr führt jeder gefahrbringende Ausfall zu einem Ausfall der Sicherheitsfunktion, wenn eine Anforderung auftritt.

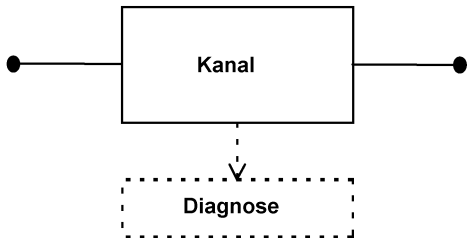


Bild B.4 – Blockschaltbild für 1oo1

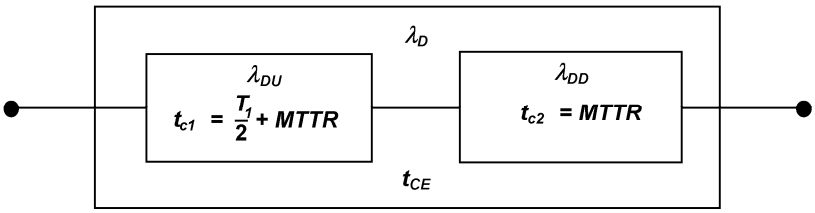


Bild B.5 – Zuverlässigkeitsblockdiagramm für 1oo1

Bild B.4 enthält das zugehörige Blockschaltbild und Bild B.5 das Zuverlässigkeitsblockdiagramm. Die Rate eines gefahrbringenden Ausfalls ist für einen Kanal gegeben durch:

$$\lambda_D = \lambda_{DU} + \lambda_{DD}$$

Bild B.5 zeigt, dass man sich einen Kanal als aus zwei Komponenten zusammengesetzt vorstellen kann, eine mit einer Rate für gefahrbringende Ausfälle λ_{DU} aufgrund unerkannter Ausfälle und eine mit einer Rate für gefahrbringende Ausfälle λ_{DD} aufgrund erkannter Ausfälle. Es ist möglich, die mittlere Ausfalldauer t_{CE} eines Kanals zu berechnen, indem man die einzelnen Ausfalldauern der beiden Komponenten t_{C1} und t_{C2} addiert, jeweils bewertet mit dem Verhältnis, mit dem sie in die Ausfallwahrscheinlichkeit des Kanals eingehen:

$$t_{CE} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{2} + MRT\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR$$

Für jede Architektur ist die Rate erkannter gefahrbringender Ausfälle und die Rate unerkannter gefahrbringender Ausfälle bestimmt durch:

$$\lambda_{DU} = \lambda_D(1 - DC)\,; \quad \lambda_{DD} = \lambda_D DC$$

Für einen Kanal mit einer durch einen gefahrbringenden Ausfall bedingten Ausfalldauer t_{CE} gilt:

$$PFD = 1 - e^{-\lambda_D t_{CE}}$$
$$\approx \lambda_D t_{CE} \qquad \text{da } \lambda_D t_{CE} << 1$$

Daher ist für eine 1oo1-Architektur die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung:

$$PFD_G = (\lambda_{DU} + \lambda_{DD}) t_{CE}$$

B.3.2.2.2 1oo2

Diese Architektur besteht aus zwei parallelen Kanälen, so dass jeder der Kanäle die Sicherheitsfunktion ausführen kann. Daher müsste ein gefahrbringender Ausfall in beiden Kanälen vorliegen, bevor die Sicherheitsfunktion bei Anforderung ausfallen würde. Es wird angenommen, dass jeder Diagnosetest erkannte Fehler nur meldet und keine Ausgangszustände oder den Ausgangsvergleich ändert.

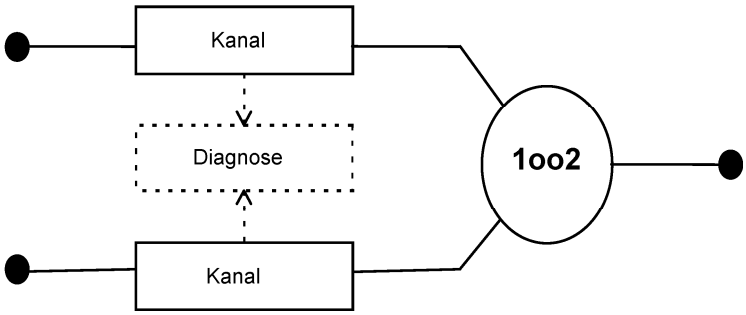


Bild B.6 – Blockschaltbild für 1oo2

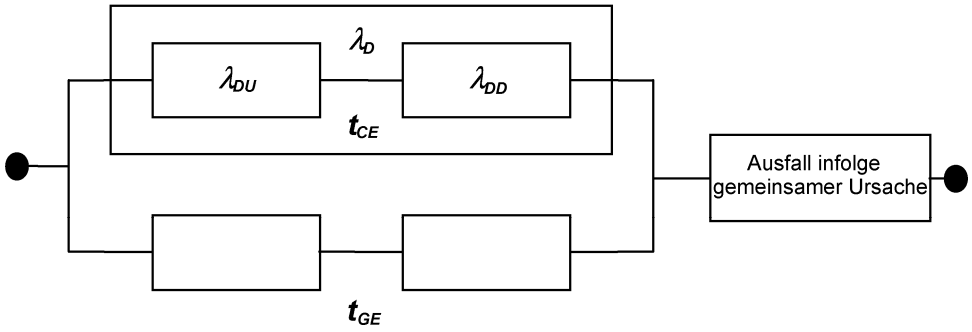


Bild B.7 – Zuverlässigkeitsblockdiagramm für 1oo2

Bild B.6 enthält das zugehörige Blockschaltbild und Bild B.7 das Zuverlässigkeitsblockdiagramm. Der Wert für t_{CE} ist wie in B.3.2.2.1 angegeben, aber jetzt ist es notwendig, die Ausfalldauer t_{GE} des Systems auch zu berechnen, die gegeben ist durch:

$$t_{GE} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{3} + MRT\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR$$

Die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für diese Architektur ist:

$$PFD_G = 2\left((1-\beta_D)\lambda_{DD} + (1-\beta)\lambda_{DU}\right)^2 t_{CE} t_{GE} + \beta_D \lambda_{DD} MTTR + \beta\lambda_{DU}\left(\frac{T_1}{2} + MRT\right)$$

B.3.2.2.3 2oo2

Diese Architektur besteht aus zwei parallelen Kanälen, so dass jeder der Kanäle die Sicherheitsfunktion anfordern muss, bevor sie ausgeführt werden kann. Es wird angenommen, dass jeder Diagnosetest erkannte Fehler nur meldet und keine Ausgangszustände oder den Ausgangsvergleich ändert.

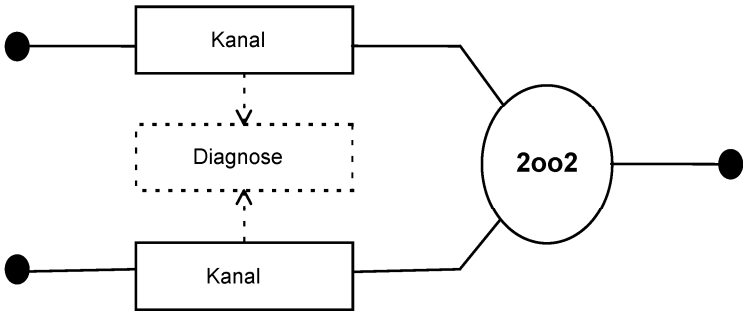


Bild B.8 – Blockschaltbild für 2oo2

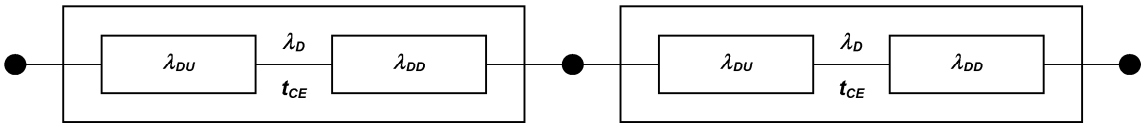


Bild B.9 – Zuverlässigkeitsblockdiagramm für 2oo2

Bild B.8 enthält das zugehörige Blockschaltbild und Bild B.9 das Zuverlässigkeitsblockdiagramm. Der Wert für t_{CE} ist wie in B.3.2.2.1 angegeben, und die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für diese Architektur ist:

$$PFD_G = 2\lambda_D t_{CE}$$

B.3.2.2.4 1oo2D

Diese Architektur besteht aus zwei parallelen Kanälen. Während des normalen Betriebs müssen beide Kanäle die Sicherheitsfunktion anfordern, bevor sie ausgeführt werden kann. Sollte darüber hinaus der Diagnosetest in einem Kanal einen Fehler erkennen, wird der Ausgangsvergleich angepasst, so dass der Gesamtausgangszustand dem anderen Kanal folgt. Falls der Diagnosetest in beiden Kanälen Fehler erkennt oder eine Abweichung, die keinem der beiden Kanäle zugeordnet werden kann, begibt sich der Ausgang in den sicheren Zustand. Um eine Abweichung zwischen den beiden Kanälen zu erkennen, kann jeder Kanal den Zustand des anderen durch von ihm unabhängige Mittel bestimmen. Der Kanalvergleicher/Umschalter ist möglicherweise nicht 100 % wirksam, deshalb wird K für die Darstellung der Wirksamkeit dieses kanalübergreifenden Vergleichs-/Umschaltmechanismus verwendet, d. h. der Ausgang kann, auch wenn ein Kanal als fehlerhaft erkannt wird, weiter dem 2oo2-Ausgangsvergleicher folgen.

ANMERKUNG Der Parameter K ist im Rahmen einer FMEA festzulegen.

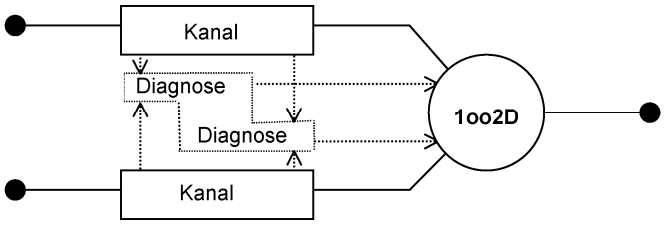


Bild B.10 – Blockschaltbild für 1oo2D

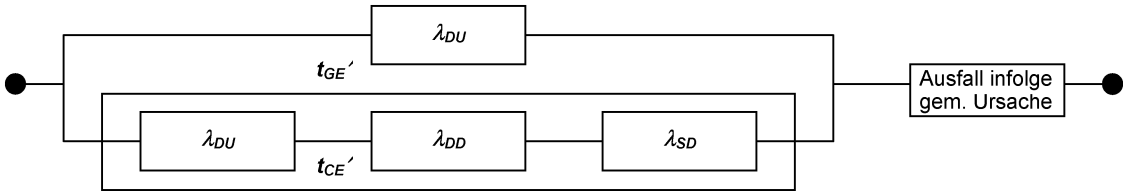


Bild B.11 – Zuverlässigkeitsblockdiagramm für 1oo2D

Die Rate erkannter sicherer Ausfälle für jeden Kanal ist gegeben durch:

$$\lambda_{SD} = \lambda_S DC$$

Bild B.10 enthält das zugehörige Blockschaltbild und Bild B.11 das Zuverlässigkeitsblockdiagramm. Die Werte der mittleren Ausfalldauern unterscheiden sich von denen der anderen Architekturen in B.3.2.2 und werden deshalb als $t_{CE}´$ und $t_{GE}´$ bezeichnet. Ihre Werte sind gegeben durch:

$$t_{CE}' = \frac{\lambda_{DU}\left(\frac{T_1}{2} + MRT\right) + (\lambda_{DD} + \lambda_{SD})MTTR}{\lambda_{DU} + (\lambda_{DD} + \lambda_{SD})}$$

$$t_{GE}' = \frac{T_1}{3} + MRT$$

Die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für diese Architektur ist:

$$PFD_G = 2(1-\beta)\lambda_{DU}((1-\beta)\lambda_{DU} + (1-\beta_D)\lambda_{DD} + \lambda_{SD})t_{CE}'\, t_{GE}' + 2(1-K)\lambda_{DD}t_{CE}' + \beta\lambda_{DU}\left(\frac{T_1}{2} + MRT\right)$$

B.3.2.2.5 2oo3

Diese Architektur besteht aus drei parallelen Kanälen, verbunden mit einem Mehrheitsentscheider, so dass sich der Ausgangszustand nicht ändert, wenn nur ein Kanal ein von den beiden anderen Kanälen unterschiedliches Ergebnis liefert.

Es wird vorausgesetzt, dass jeder Diagnosetest erkannte Fehler nur meldet und keine Ausgangszustände oder den Mehrheitsentscheid ändert.

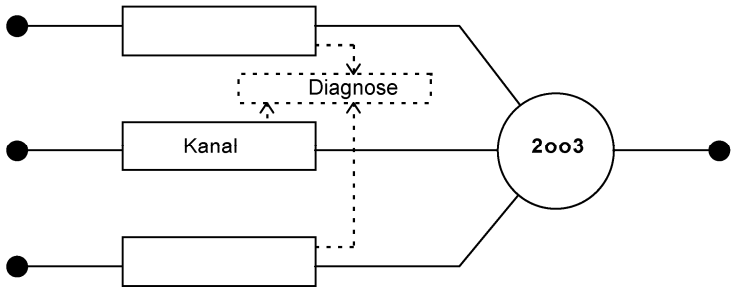


Bild B.12 – Blockschaltbild für 2oo3

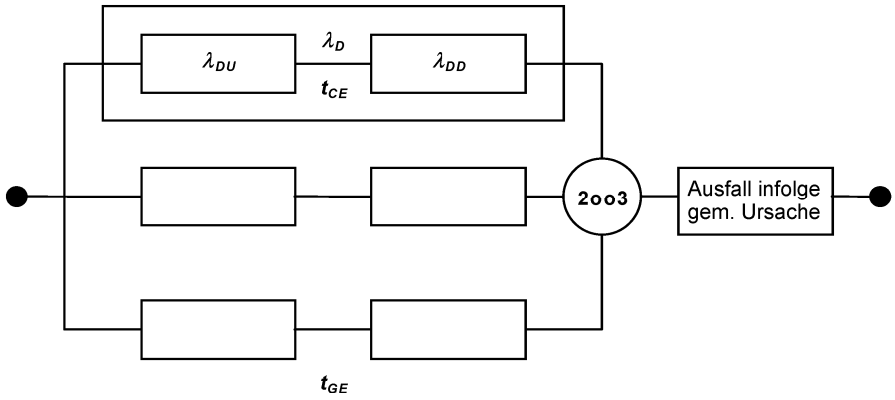


Bild B.13 – Zuverlässigkeitsblockdiagramm für 2oo3

Bild B.12 enthält das zugehörige Blockschaltbild und Bild B.13 das Zuverlässigkeitsblockdiagramm. Der Wert für t_{CE} ist wie in B.3.2.2.1 und der Wert für t_{GE} wie im B.3.2.2.2 angegeben. Die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für diese Architektur ist:

$$PFD_G = 6((1-\beta_D)\lambda_{DD} + (1-\beta)\lambda_{DU})^2 t_{CE}t_{GE} + \beta_D\lambda_{DD}MTTR + \beta\lambda_{DU}\left(\frac{T_1}{2} + MRT\right)$$

B.3.2.2.6 1oo3

Diese Architektur besteht aus drei parallelen Kanälen, verbunden mit einem Ausgangsvergleicher, so dass der Ausgangszustand sich nach der 1oo3-Mehrheitsentscheidung ausrichtet.

Es wird angenommen, dass jeder Diagnosetest erkannte Fehler nur meldet und keine Ausgangszustände oder den Mehrheitsentscheid ändert.

Das Zuverlässigkeitsblockdiagramm entspricht dem für 2oo3, allerdings mit 1oo3-Mehrheitsentscheidung. Der Wert für t_{CE} ist wie in B.3.2.2.1 und der Wert für t_{GE} wie in B.3.2.2.2 angegeben. Die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für diese Architektur ist:

$$PFD_G = 6((1-\beta_D)\lambda_{DD} + (1-\beta)\lambda_{DU})^3 t_{CE} t_{GE} t_{G2E} + \beta_D \lambda_{DD} MTTR + \beta\lambda_{DU}\left(\frac{T_1}{2} + MRT\right)$$

Wobei

$$t_{G2E} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{4} + MRT\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR$$

B.3.2.3 Tabellen für die Betriebsart mit niedriger Anforderungsrate

Tabelle B.2 – Mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für ein Wiederholungsprüfungsintervall von sechs Monaten und eine mittlere Dauer bis zur Wiederherstellung von 8 h

Architektur	*DC*	λ_D = 0,5E-07			λ_D = 2,5E-07			λ_D = 0,5E-06		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	1,1E-04			5,5E-04			1,1E-03		
	60 %	4,4E-05			2,2E-04			4,4E-04		
	90 %	1,1E-05			5,7E-05			1,1E-04		
	99 %	1,5E-06			7,5E-06			1,5E-05		
1oo2	**0 %**	2,2E-06	1,1E-05	2,2E-05	1,1E-05	5,5E-05	1,1E-04	2,4E-05	1,1E-04	2,2E-04
	60 %	8,8E-07	4,4E-06	8,8E-06	4,5E-06	2,2E-05	4,4E-05	9,1E-06	4,4E-05	8,8E-05
	90 %	2,2E-07	1,1E-06	2,2E-06	1,1E-06	5,6E-06	1,1E-05	2,3E-06	1,1E-05	2,2E-05
	99 %	2,6E-08	1,3E-07	2,6E-07	1,3E-07	6,5E-07	1,3E-06	2,6E-07	1,3E-06	2,6E-06
2oo2 (s. Anm. 2)	**0 %**	2,2E-04			1,1E-03			2,2E-03		
	60 %	8,8E-05			4,4E-04			8,8E-04		
	90 %	2,3E-05			1,1E-04			2,3E-04		
	99 %	3,0E-06			1,5E-05			3,0E-05		
1oo2D (s. Anm. 3)	**0 %**	2,2E-06	1,1E-05	2,2E-05	1,1E-05	5,5E-05	1,1E-04	2,4E-05	1,1E-04	2,2E-04
	60 %	1,4E-06	4,9E-06	9,3E-06	7,1E-06	2,5E-05	4,7E-05	1,4E-05	5,0E-05	9,3E-05
	90 %	4,3E-07	1,3E-06	2,4E-06	2,2E-06	6,6E-06	1,2E-05	4,3E-06	1,3E-05	2,4E-05
	99 %	6,0E-08	1,5E-07	2,6E-07	3,0E-07	7,4E-07	1,3E-06	6,0E-07	1,5E-06	2,6E-06
2oo3	**0 %**	2,2E-06	1,1E-05	2,2E-05	1,2E-05	5,6E-05	1,1E-04	2,7E-05	1,1E-04	2,2E-04
	60 %	8,9E-07	4,4E-06	8,8E-06	4,6E-06	2,2E-05	4,4E-05	9,6E-06	4,5E-05	8,9E-05
	90 %	2,2E-07	1,1E-06	2,2E-06	1,1E-06	5,6E-06	1,1E-05	2,3E-06	1,1E-05	2,2E-05
	99 %	2,6E-08	1,3E-07	2,6E-07	1,3E-07	6,5E-07	1,3E-06	2,6E-07	1,3E-06	2,6E-06
1oo3	**0 %**	2,2E-06	1,1E-05	2,2E-05	1,1E-05	5,5E-05	1,1E-04	2,2E-05	1,1E-04	2,2E-04
	60 %	8,8E-07	4,4E-06	8,8E-06	4,4E-06	2,2E-05	4,4E-05	8,8E-06	4,4E-05	8,8E-05
	90 %	2,2E-07	1,1E-06	2,2E-06	1,1E-06	5,6E-06	1,1E-05	2,2E-06	1,1E-05	2,2E-05
	99 %	2,6E-08	1,3E-07	2,6E-07	1,3E-07	6,5E-07	1,3E-06	2,6E-07	1,3E-06	2,6E-06

Architektur	*DC*	λ_D = 2,5E-06			λ_D = 0,5E-05			λ_D = 2,5E-05		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	5,5E-03			1,1E-02			5,5E-02		
	60 %	2,2E-03			4,4E-03			2,2E-02		
	90 %	5,7E-04			1,1E-03			5,7E-03		
	99 %	7,5E-05			1,5E-04			7,5E-04		
1oo2	**0 %**	1,5E-04	5,8E-04	1,1E-03	3,7E-04	1,2E-03	2,3E-03	5,0E-03	8,8E-03	1,4E-02
	60 %	5,0E-05	2,3E-04	4,5E-04	1,1E-04	4,6E-04	9,0E-04	1,1E-03	2,8E-03	4,9E-03
	90 %	1,2E-05	5,6E-05	1,1E-04	2,4E-05	1,1E-04	2,2E-04	1,5E-04	6,0E-04	1,2E-03
	99 %	1,3E-06	6,5E-06	1,3E-05	2,6E-06	1,3E-05	2,6E-05	1,4E-05	6,6E-05	1,3E-04
2oo2 (s. Anm. 2)	**0 %**	1,1E-02			2,2E-02			>1E-01		
	60 %	4,4E-03			8,8E-03			4,4E-02		
	90 %	1,1E-03			2,3E-03			1,1E-02		
	99 %	1,5E-04			3,0E-04			1,5E-03		
1oo2D (s. Anm. 3)	**0 %**	1,5E-04	5,8E-04	1,1E-03	3,8E-04	1,2E-03	2,3E-03	5,0E-03	9,0E-03	1,4E-02
	60 %	7,7E-05	2,5E-04	4,7E-04	1,7E-04	5,2E-04	9,5E-04	1,3E-03	3,0E-03	5,1E-03
	90 %	2,2E-05	6,6E-05	1,2E-04	4,5E-05	1,3E-04	2,4E-04	2,6E-04	6,9E-04	1,2E-03
	99 %	3,0E-06	7,4E-06	1,3E-05	6,0E-06	1,5E-05	2,6E-05	3,0E-05	7,4E-05	1,3E-04
2oo3	**0 %**	2,3E-04	6,5E-04	1,2E-03	6,8E-04	1,5E-03	2,5E-03	1,3E-02	1,5E-02	1,9E-02
	60 %	6,3E-05	2,4E-04	4,6E-04	1,6E-04	5,1E-04	9,4E-04	2,3E-03	3,9E-03	5,9E-03
	90 %	1,2E-05	5,7E-05	1,1E-04	2,7E-05	1,2E-04	2,3E-04	2,4E-04	6,8E-04	1,2E-03
	99 %	1,3E-06	6,5E-06	1,3E-05	2,7E-06	1,3E-05	2,6E-05	1,5E-05	6,7E-05	1,3E-04
1oo3	**0 %**	1,1E-04	5,5E-04	1,1E-03	2,2E-04	1,1E-03	2,2E-03	1,4E-03	5,7E-03	1,1E-02
	60 %	4,4E-05	2,2E-04	4,4E-04	8,8E-05	4,4E-04	8,8E-04	4,6E-04	2,2E-03	4,4E-03
	90 %	1,1E-05	5,6E-05	1,1E-04	2,2E-05	1,1E-04	2,2E-04	1,1E-04	5,6E-04	1,1E-03
	99 %	1,3E-06	6,5E-06	1,3E-05	2,6E-06	1,3E-05	2,6E-05	1,3E-05	6,5E-05	1,3E-04

ANMERKUNG 1 Diese Tabelle liefert Beispielwerte für PFD_G, berechnet unter Verwendung der Gleichungen aus B.3.2 und beruhend auf den Annahmen in B.3.1. Wenn das Sensor-, Logik- oder Stellglied-Teilsystem nur aus einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider besteht, ist PFD_G gleich PFD_S, PFD_L bzw. PFD_{FE} (siehe B.3.2.1).

ANMERKUNG 2 Diese Tabelle nimmt $\beta = 2 \times \beta_D$ an. Die Werte für β und β_D beeinflussen die mittlere Wahrscheinlichkeit eines Ausfalls bei 1oo1- und 2oo2-Architekturen nicht.

ANMERKUNG 3 Es werden gleiche Raten sicherer wie gefahrbringender Ausfälle angenommen sowie K = 0,98.

Tabelle B.3 – Mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für ein Wiederholungsprüfungsintervall von einem Jahr und eine mittlere Dauer bis zur Wiederherstellung von 8 h

Architektur	DC	λ_D = 0,5E-07			λ_D = 2,5E-07			λ_D = 0,5E-06		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	2,2E-04			1,1E-03			2,2E-03		
	60 %	8,8E-05			4,4E-04			8,8E-04		
	90 %	2,2E-05			1,1E-04			2,2E-04		
	99 %	2,6E-06			1,3E-05			2,6E-05		
1oo2	**0 %**	4,4E-06	2,2E-05	4,4E-05	2,3E-05	1,1E-04	2,2E-04	5,0E-05	2,2E-04	4,4E-04
	60 %	1,8E-06	8,8E-06	1,8E-05	9,0E-06	4,4E-05	8,8E-05	1,9E-05	8,9E-05	1,8E-04
	90 %	4,4E-07	2,2E-06	4,4E-06	2,2E-06	1,1E-05	2,2E-05	4,5E-06	2,2E-05	4,4E-05
	99 %	4,8E-08	2,4E-07	4,8E-07	2,4E-07	1,2E-06	2,4E-06	4,8E-07	2,4E-06	4,8E-06
2oo2 (s. Anm. 2)	**0 %**	4,4E-04			2,2E-03			4,4E-03		
	60 %	1,8E-04			8,8E-04			1,8E-03		
	90 %	4,5E-05			2,2E-04			4,5E-04		
	99 %	5,2E-06			2,6E-05			5,2E-05		
1oo2D (s. Anm. 3)	**0 %**	4,5E-06	2,2E-05	4,4E-05	2,4E-05	1,1E-04	2,2E-04	5,0E-05	2,2E-04	4,4E-04
	60 %	2,8E-06	9,8E-06	1,9E-05	1,4E-05	4,9E-05	9,3E-05	2,9E-05	9,9E-05	1,9E-04
	90 %	8,5E-07	2,6E-06	4,8E-06	4,3E-06	1,3E-05	2,4E-05	8,5E-06	2,6E-05	4,8E-05
	99 %	1,0E-07	2,8E-07	5,0E-07	5,2E-07	1,4E-06	2,5E-06	1,0E-06	2,8E-06	5,0E-06
2oo3	**0 %**	4,6E-06	2,2E-05	4,4E-05	2,7E-05	1,1E-04	2,2E-04	6,2E-05	2,4E-04	4,5E-04
	60 %	1,8E-06	8,8E-06	1,8E-05	9,5E-06	4,5E-05	8,8E-05	2,1E-05	9,1E-05	1,8E-04
	90 %	4,4E-07	2,2E-06	4,4E-06	2,3E-06	1,1E-05	2,2E-05	4,6E-06	2,2E-05	4,4E-05
	99 %	4,8E-08	2,4E-07	4,8E-07	2,4E-07	1,2E-06	2,4E-06	4,8E-07	2,4E-06	4,8E-06
1oo3	**0 %**	4,4E-06	2,2E-05	4,4E-05	2,2E-05	1,1E-04	2,2E-04	4,4E-05	2,2E-04	4,4E-04
	60 %	1,8E-06	8,8E-06	1,8E-05	8,8E-06	4,4E-05	8,8E-05	1,8E-05	8,8E-05	1,8E-04
	90 %	4,4E-07	2,2E-06	4,4E-06	2,2E-06	1,1E-05	2,2E-05	4,4E-06	2,2E-05	4,4E-05
	99 %	4,8E-08	2,4E-07	4,8E-07	2,4E-07	1,2E-06	2,4E-06	4,8E-07	2,4E-06	4,8E-06

Architektur	DC	λ_D = 2,5E-06			λ_D = 0,5E-05			λ_D = 2,5E-05		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	1,1E-02			2,2E-02			>1E-01		
	60 %	4,4E-03			8,8E-03			4,4E-02		
	90 %	1,1E-03			2,2E-03			1,1E-02		
	99 %	1,3E-04			2,6E-04			1,3E-03		
1oo2	**0 %**	3,7E-04	1,2E-03	2,3E-03	1,1E-03	2,7E-03	4,8E-03	1,8E-02	2,4E-02	3,2E-02
	60 %	1,1E-04	4,6E-04	9,0E-04	2,8E-04	9,7E-04	1,8E-03	3,4E-03	6,6E-03	1,1E-02
	90 %	2,4E-05	1,1E-04	2,2E-04	5,1E-05	2,3E-04	4,5E-04	3,8E-04	1,3E-03	2,3E-03
	99 %	2,4E-06	1,2E-05	2,4E-05	4,9E-06	2,4E-05	4,8E-05	2,6E-05	1,2E-04	2,4E-04
2oo2 (s. Anm. 2)	**0 %**	2,2E-02			4,4E-02			>1E-01		
	60 %	8,8E-03			1,8E-02			8,8E-02		
	90 %	2,2E-03			4,5E-03			2,2E-02		
	99 %	2,6E-04			5,2E-04			2,6E-03		
1oo2D (s. Anm. 3)	**0 %**	3,8E-04	1,2E-03	2,3E-03	1,1E-03	2,7E-03	4,9E-03	1,8E-02	2,5E-02	3,4E-02
	60 %	1,7E-04	5,1E-04	9,5E-04	3,8E-04	1,1E-03	1,9E-03	3,9E-03	7,1E-03	1,1E-02
	90 %	4,4E-05	1,3E-04	2,4E-04	9,1E-05	2,7E-04	4,8E-04	5,8E-04	1,4E-03	2,5E-03
	99 %	5,2E-06	1,4E-05	2,5E-05	1,0E-05	2,8E-05	5,0E-05	5,4E-05	1,4E-04	2,5E-04
2oo3	**0 %**	6,8E-04	1,5E-03	2,5E-03	2,3E-03	3,8E-03	5,6E-03	4,8E-02	5,0E-02	5,3E-02
	60 %	1,6E-04	5,1E-04	9,4E-04	4,8E-04	1,1E-03	2,0E-03	8,4E-03	1,1E-02	1,5E-02
	90 %	2,7E-05	1,2E-04	2,3E-04	6,4E-05	2,4E-04	4,6E-04	7,1E-04	1,6E-03	2,6E-03
	99 %	2,5E-06	1,2E-05	2,4E-05	5,1E-06	2,4E-05	4,8E-05	3,1E-05	1,3E-04	2,5E-04
1oo3	**0 %**	2,2E-04	1,1E-03	2,2E-03	4,6E-04	2,2E-03	4,4E-03	4,7E-03	1,3E-02	2,3E-02
	60 %	8,8E-05	4,4E-04	8,8E-04	1,8E-04	8,8E-04	1,8E-03	1,0E-03	4,5E-03	8,9E-03
	90 %	2,2E-05	1,1E-04	2,2E-04	4,4E-05	2,2E-04	4,4E-04	2,2E-04	1,1E-03	2,2E-03
	99 %	2,4E-06	1,2E-05	2,4E-05	4,8E-06	2,4E-05	4,8E-05	2,4E-05	1,2E-04	2,4E-04

ANMERKUNG 1 Diese Tabelle liefert Beispielwerte für PFD_G, berechnet unter Verwendung der Gleichungen aus B.3.2 und beruhend auf den Annahmen in B.3.1. Wenn das Sensor-, Logik- oder Stellglied-Teilsystem nur aus einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider besteht, ist PFD_G gleich PFD_S, PFD_L bzw. PFD_{FE} (siehe B.3.2.1).

ANMERKUNG 2 Diese Tabelle nimmt $\beta = 2 \times \beta_D$ an. Die Werte für β und β_D beeinflussen die mittlere Wahrscheinlichkeit eines Ausfalls bei 1oo1- und 2oo2-Architekturen nicht.

ANMERKUNG 3 Es werden gleiche Raten sicherer wie gefahrbringender Ausfälle angenommen sowie K = 0,98.

Tabelle B.4 – Mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für ein Wiederholungsprüfungsintervall von zwei Jahren und eine mittlere Dauer bis zur Wiederherstellung von 8 h

Architektur	*DC*	λ_D = 0,5E-07			λ_D = 2,5E-07			λ_D = 0,5E-06		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	4,4E-04			2,2E-03			4,4E-03		
	60 %	1,8E-04			8,8E-04			1,8E-03		
	90 %	4,4E-05			2,2E-04			4,4E-04		
	99 %	4,8E-06			2,4E-05			4,8E-05		
1oo2	**0 %**	9,0E-06	4,4E-05	8,8E-05	5,0E-05	2,2E-04	4,4E-04	1,1E-04	4,6E-04	8,9E-04
	60 %	3,5E-06	1,8E-05	3,5E-05	1,9E-05	8,9E-05	1,8E-04	3,9E-05	1,8E-04	3,5E-04
	90 %	8,8E-07	4,4E-06	8,8E-06	4,5E-06	2,2E-05	4,4E-05	9,1E-06	4,4E-05	8,8E-05
	99 %	9,2E-08	4,6E-07	9,2E-07	4,6E-07	2,3E-06	4,6E-06	9,2E-07	4,6E-06	9,2E-06
2oo2 (s. Anm. 2)	**0 %**	8,8E-04			4,4E-03			8,8E-03		
	60 %	3,5E-04			1,8E-03			3,5E-03		
	90 %	8,8E-05			4,4E-04			8,8E-04		
	99 %	9,6E-06			4,8E-05			9,6E-05		
1oo2D (s. Anm. 3)	**0 %**	9,0E-06	4,4E-05	8,8E-05	5,0E-05	2,2E-04	4,4E-04	1,1E-04	4,6E-04	9,0E-04
	60 %	5,7E-06	2,0E-05	3,7E-05	2,9E-05	9,9E-05	1,9E-04	6,0E-05	2,0E-04	3,7E-04
	90 %	1,7E-06	5,2E-06	9,6E-06	8,5E-06	2,6E-05	4,8E-05	1,7E-05	5,2E-05	9,6E-05
	99 %	1,9E-07	5,4E-07	9,8E-07	9,5E-07	2,7E-06	4,9E-06	1,9E-06	5,4E-06	9,8E-06
2oo3	**0 %**	9,5E-06	4,4E-05	8,8E-05	6,2E-05	2,3E-04	4,5E-04	1,6E-04	5,0E-04	9,3E-04
	60 %	3,6E-06	1,8E-05	3,5E-05	2,1E-05	9,0E-05	1,8E-04	4,7E-05	1,9E-04	3,6E-04
	90 %	8,9E-07	4,4E-06	8,8E-06	4,6E-06	2,2E-05	4,4E-05	9,6E-06	4,5E-05	8,9E-05
	99 %	9,2E-08	4,6E-07	9,2E-07	4,6E-07	2,3E-06	4,6E-06	9,3E-07	4,6E-06	9,2E-06
1oo3	**0 %**	8,8E-06	4,4E-05	8,8E-05	4,4E-05	2,2E-04	4,4E-04	8,8E-05	4,4E-04	8,8E-04
	60 %	3,5E-06	1,8E-05	3,5E-05	1,8E-05	8,8E-05	1,8E-04	3,5E-05	1,8E-04	3,5E-04
	90 %	8,8E-07	4,4E-06	8,8E-06	4,4E-06	2,2E-05	4,4E-05	8,8E-06	4,4E-05	8,8E-05
	99 %	9,2E-08	4,6E-07	9,2E-07	4,6E-07	2,3E-06	4,6E-06	9,2E-07	4,6E-06	9,2E-06

Architektur	*DC*	λ_D =2,5E-06			λ_D = 0,5E-05			λ_D = 2,5E-05		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	2,2E-02			4,4E-02			>1E-01		
	60 %	8,8E-03			1,8E-02			8,8E-02		
	90 %	2,2E-03			4,4E-03			2,2E-02		
	99 %	2,4E-04			4,8E-04			2,4E-03		
1oo2	**0 %**	1,1E-03	2,7E-03	4,8E-03	3,3E-03	6,5E-03	1,0E-02	6,6E-02	7,4E-02	8,5E-02
	60 %	2,8E-04	9,7E-04	1,8E-03	7,5E-04	2,1E-03	3,8E-03	1,2E-02	1,8E-02	2,5E-02
	90 %	5,0E-05	2,3E-04	4,5E-04	1,1E-04	4,6E-04	9,0E-04	1,1E-03	2,8E-03	4,9E-03
	99 %	4,7E-06	2,3E-05	4,6E-05	9,5E-06	4,6E-05	9,2E-05	5,4E-05	2,4E-04	4,6E-04
2oo2 (s. Anm. 2)	**0 %**	4,4E-02			8,8E-02			>1E-01		
	60 %	1,8E-02			3,5E-02			>1E-01		
	90 %	4,4E-03			8,8E-03			4,4E-02		
	99 %	4,8E-04			9,6E-04			4,8E-03		
1oo2D (s. Anm. 3)	**0 %**	1,1E-03	2,7E-03	4,8E-03	3,4E-03	6,6E-03	1,1E-02	6,7E-02	7,7E-02	9,0E-02
	60 %	3,8E-04	1,1E-03	1,9E-03	9,6E-04	2,3E-03	4,0E-03	1,3E-02	1,9E-02	2,6E-02
	90 %	9,0E-05	2,6E-04	4,8E-04	1,9E-04	5,4E-04	9,8E-04	1,5E-03	3,2E-03	5,3E-03
	99 %	9,6E-06	2,7E-05	4,9E-05	1,9E-05	5,4E-05	9,8E-05	1,0E-04	2,8E-04	5,0E-04
2oo3	**0 %**	2,3E-03	3,7E-03	5,6E-03	8,3E-03	1,1E-02	1,4E-02	1,9E-01	1,8E-01	1,7E-01
	60 %	4,8E-04	1,1E-03	2,0E-03	1,6E-03	2,8E-03	4,4E-03	3,2E-02	3,5E-02	4,0E-02
	90 %	6,3E-05	2,4E-04	4,6E-04	1,6E-04	5,1E-04	9,4E-04	2,4E-03	4,0E-03	6,0E-03
	99 %	4,8E-06	2,3E-05	4,6E-05	1,0E-05	4,7E-05	9,2E-05	6,9E-05	2,5E-04	4,8E-04
1oo3	**0 %**	4,6E-04	2,2E-03	4,4E-03	1,0E-03	4,5E-03	8,9E-03	2,4E-02	3,7E-02	5,5E-02
	60 %	1,8E-04	8,8E-04	1,8E-03	3,6E-04	1,8E-03	3,5E-03	3,1E-03	9,9E-03	1,8E-02
	90 %	4,4E-05	2,2E-04	4,4E-04	8,8E-05	4,4E-04	8,8E-04	4,6E-04	2,2E-03	4,4E-03
	99 %	4,6E-06	2,3E-05	4,6E-05	9,2E-06	4,6E-05	9,2E-05	4,6E-05	2,3E-04	4,6E-04

ANMERKUNG 1 Diese Tabelle liefert Beispielwerte für PFD_G, berechnet unter Verwendung der Gleichungen aus B.3.2 und beruhend auf den Annahmen in B.3.1. Wenn das Sensor-, Logik- oder Stellglied-Teilsystem nur aus einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider besteht, ist PFD_G gleich PFD_S, PFD_L bzw. PFD_{FE} (siehe B.3.2.1).

ANMERKUNG 2 Diese Tabelle nimmt $\beta = 2 \times \beta_D$ an. Die Werte für β und β_D beeinflussen die mittlere Wahrscheinlichkeit eines Ausfalls bei 1oo1- und 2oo2-Architekturen nicht.

ANMERKUNG 3 Es werden gleiche Raten sicherer wie gefahrbringender Ausfälle angenommen sowie K = 0,98.

Tabelle B.5 – Mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für ein Wiederholungsprüfungsintervall von zehn Jahren und eine mittlere Dauer bis zur Wiederherstellung von 8 h

Architektur	*DC*	λ_D = 0,5E-07			λ_D = 2,5E-07			λ_D = 0,5E-06		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	2,2E-03			1,1E-02			2,2E-02		
	60 %	8,8E-04			4,4E-03			8,8E-03		
	90 %	2,2E-04			1,1E-03			2,2E-03		
	99 %	2,2E-05			1,1E-04			2,2E-04		
1oo2	**0 %**	5,0E-05	2,2E-04	4,4E-04	3,7E-04	1,2E-03	2,3E-03	1,1E-03	2,7E-03	4,8E-03
	60 %	1,9E-05	8,9E-05	1,8E-04	1,1E-04	4,6E-04	9,0E-04	2,7E-04	9,6E-04	1,8E-03
	90 %	4,4E-06	2,2E-05	4,4E-05	2,3E-05	1,1E-04	2,2E-04	5,0E-05	2,2E-04	4,4E-04
	99 %	4,4E-07	2,2E-06	4,4E-06	2,2E-06	1,1E-05	2,2E-05	4,5E-06	2,2E-05	4,4E-05
2oo2 (s. Anm. 2)	**0 %**	4,4E-03			2,2E-02			4,4E-02		
	60 %	1,8E-03			8,8E-03			1,8E-02		
	90 %	4,4E-04			2,2E-03			4,4E-03		
	99 %	4,5E-05			2,2E-04			4,5E-04		
1oo2D (s. Anm. 3)	**0 %**	5,0E-05	2,2E-04	4,4E-04	3,7E-04	1,2E-03	2,3E-03	1,1E-03	2,7E-03	4,8E-03
	60 %	2,9E-05	9,9E-05	1,9E-04	1,7E-04	5,1E-04	9,5E-04	3,8E-04	1,1E-03	1,9E-03
	90 %	8,4E-06	2,6E-05	4,8E-05	4,3E-05	1,3E-04	2,4E-04	9,0E-05	2,6E-04	4,8E-04
	99 %	8,9E-07	2,6E-06	4,8E-06	4,5E-06	1,3E-05	2,4E-05	8,9E-06	2,6E-05	4,8E-05
2oo3	**0 %**	6,2E-05	2,3E-04	4,5E-04	6,8E-04	1,5E-03	2,5E-03	2,3E-03	3,7E-03	5,6E-03
	60 %	2,1E-05	9,0E-05	1,8E-04	1,6E-04	5,0E-04	9,3E-04	4,7E-04	1,1E-03	2,0E-03
	90 %	4,6E-06	2,2E-05	4,4E-05	2,7E-05	1,1E-04	2,2E-04	6,3E-05	2,4E-04	4,5E-04
	99 %	4,4E-07	2,2E-06	4,4E-06	2,3E-06	1,1E-05	2,2E-05	4,6E-06	2,2E-05	4,4E-05
1oo3	**0 %**	4,4E-05	2,2E-04	4,4E-04	2,2E-04	1,1E-03	2,2E-03	4,6E-04	2,2E-03	4,4E-03
	60 %	1,8E-05	8,8E-05	1,8E-04	8,8E-05	4,4E-04	8,8E-04	1,8E-04	8,8E-04	1,8E-03
	90 %	4,4E-06	2,2E-05	4,4E-05	2,2E-05	1,1E-04	2,2E-04	4,4E-05	2,2E-04	4,4E-04
	99 %	4,4E-07	2,2E-06	4,4E-06	2,2E-06	1,1E-05	2,2E-05	4,4E-06	2,2E-05	4,4E-05

Architektur	*DC*	λ_D =2,5E-06			λ_D = 0,5E-05			λ_D = 2,5E-05		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	>1E-01			>1E-01			>1E-01		
	60 %	4,4E-02			8,8E-02			>1E-01		
	90 %	1,1E-02			2,2E-02			>1E-01		
	99 %	1,1E-03			2,2E-03			1,1E-02		
1oo2	**0 %**	1,8E-02	2,4E-02	3,2E-02	6,6E-02	7,4E-02	8,5E-02	>1E-01	>1E-01	>1E-01
	60 %	3,4E-03	6,6E-03	1,1E-02	1,2E-02	1,8E-02	2,5E-02	>1E-01	>1E-01	>1E-01
	90 %	3,8E-04	1,2E-03	2,3E-03	1,1E-03	2,8E-03	4,9E-03	1,8E-02	2,5E-02	3,5E-02
	99 %	2,4E-05	1,1E-04	2,2E-04	5,1E-05	2,3E-04	4,5E-04	3,8E-04	1,3E-03	2,3E-03
2oo2 (s. Anm. 2)	**0 %**	>1E-01			>1E-01			>1E-01		
	60 %	8,8E-02			>1E-01			>1E-01		
	90 %	2,2E-02			4,4E-02			>1E-01		
	99 %	2,2E-03			4,5E-03			2,2E-02		
1oo2D (s. Anm. 3)	**0 %**	1,8E-02	2,5E-02	3,3E-02	6,6E-02	7,7E-02	9,0E-02	1,6E+00	1,5E+00	1,4E+00
	60 %	3,9E-03	7,1E-03	1,1E-02	1,3E-02	1,9E-02	2,6E-02	2,6E-01	2,7E-01	2,8E-01
	90 %	5,7E-04	1,4E-03	2,5E-03	1,5E-03	3,1E-03	5,2E-03	2,0E-02	2,7E-02	3,5E-02
	99 %	4,6E-05	1,3E-04	2,4E-04	9,5E-05	2,7E-04	4,9E-04	6,0E-04	1,5E-03	2,5E-03
2oo3	**0 %**	4,8E-02	5,0E-02	5,3E-02	1,9E-01	1,8E-01	1,7E-01	4,6E+00	4,0E+00	3,3E+00
	60 %	8,3E-03	1,1E-02	1,4E-02	3,2E-02	3,5E-02	4,0E-02	7,6E-01	7,1E-01	6,6E-01
	90 %	6,9E-04	1,5E-03	2,6E-03	2,3E-03	3,9E-03	5,9E-03	4,9E-02	5,4E-02	6,0E-02
	99 %	2,7E-05	1,2E-04	2,3E-04	6,4E-05	2,4E-04	4,6E-04	7,1E-04	1,6E-03	2,6E-03
1oo3	**0 %**	4,7E-03	1,3E-02	2,3E-02	2,4E-02	3,7E-02	5,5E-02	2,5E+00	2,0E+00	1,6E+00
	60 %	1,0E-03	4,5E-03	8,9E-03	3,0E-03	9,8E-03	1,8E-02	1,7E-01	1,8E-01	1,9E-01
	90 %	2,2E-04	1,1E-03	2,2E-03	4,6E-04	2,2E-03	4,4E-03	4,8E-03	1,3E-02	2,4E-02
	99 %	2,2E-05	1,1E-04	2,2E-04	4,4E-05	2,2E-04	4,4E-04	2,2E-04	1,1E-03	2,2E-03

ANMERKUNG 1 Diese Tabelle liefert Beispielwerte für PFD_G, berechnet unter Verwendung der Gleichungen aus B.3.2 und beruhend auf den Annahmen in B.3.1. Wenn das Sensor-, Logik- oder Stellglied-Teilsystem nur aus einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider besteht, ist PFD_G gleich PFD_S, PFD_L bzw. PFD_{FE} (siehe B.3.2.1).

ANMERKUNG 2 Diese Tabelle nimmt $\beta = 2 \times \beta_D$ an. Die Werte für β und β_D beeinflussen die mittlere Wahrscheinlichkeit eines Ausfalls bei 1oo1- und 2oo2-Architekturen nicht.

ANMERKUNG 3 Es werden gleiche Raten sicherer wie gefahrbringender Ausfälle angenommen sowie K = 0,98.

B.3.2.4 Beispiel für die Betriebsart mit niedriger Anforderungsrate

Es sei eine Sicherheitsfunktion betrachtet, die ein System mit SIL 2 erfordert. Es sei weiter angenommen, dass die erste Beurteilung auf der Basis früherer Erfahrungen im Hinblick auf die Systemarchitektur zu einer Gruppe von drei analogen Drucksensoren mit 2oo3-Mehrheitsentscheider geführt hat. Das Logik-Teilsystem ist ein PE-System, das als redundantes 1oo2D-System aufgebaut ist und ein einzelnes Absperrventil und ein einzelnes Entlüftungsventil ansteuert. Zum Erreichen der Sicherheitsfunktion müssen sowohl das Absperr-, als auch das Entlüftungsventil arbeiten. Die Architektur ist in Bild B.14 dargestellt. Bei der ersten Beurteilung wurde ein Wiederholungsprüfungsintervall von einem Jahr angenommen.

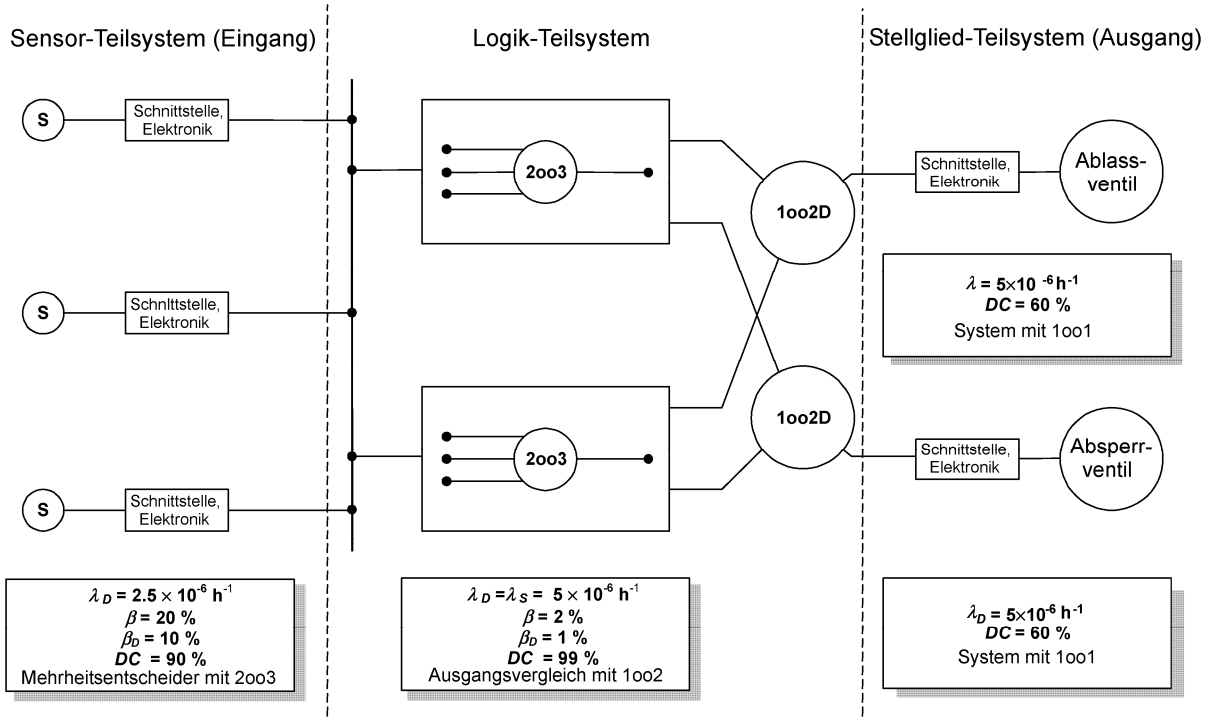


Bild B.14 – Architektur eines Beispiels für die Betriebsart mit niedriger Anforderungsrate

Tabelle B.6 – Mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für das Sensor-Teilsystem in dem Beispiel für die Betriebsart mit niedriger Anforderungsrate (ein Jahr Wiederholungsprüfungsintervall und 8 h *MTTR*)

Architektur	*DC*	λ_D = 2,5E-06		
		β = 2 % β_D = 1 %	β = 10 % β_D = 5 %	β = 20 % β_D = 10 %
2oo3	0 %	6,8E-04	1,5E-03	2,5E-03
	60 %	1,6E-04	5,1E-04	9,4E-04
	90 %	2,7E-05	1,2E-04	2,3E-04
	99 %	2,5E-06	1,2E-05	2,4E-05

ANMERKUNG Diese Tabelle ist ein Auszug aus Tabelle B.3.

Tabelle B.7 – Mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für das Logik-Teilsystem in dem Beispiel für die Betriebsart mit niedriger Anforderungsrate (ein Jahr Wiederholungsprüfungsintervall und 8 h *MTTR*)

Architektur	*DC*	λ_D = 0,5E-05		
		β = 2 % β_D = 1 %	β = 10 % β_D = 5 %	β = 20 % β_D = 10 %
1oo2D	0 %	1,1E-03	2,7E-03	4,8E-03
	60 %	2,0E-04	9,0E-04	1,8E-03
	90 %	4,5E-05	2,2E-04	4,4E-04
	99 %	4,8E-06	2,4E-05	4,8E-05
ANMERKUNG Diese Tabelle ist ein Auszug aus Tabelle B.3.				

Tabelle B.8 – Mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung für das Stellglied-Teilsystem in dem Beispiel für die Betriebsart mit niedriger Anforderungsrate (ein Jahr Wiederholungsprüfungsintervall und 8 h *MTTR*)

Architektur	*DC*	λ_D= 2,5E-06	λ_D= 0,5E-05
1oo1	0 %	1,1E-02	2,2E-02
	60 %	4,4E-03	8,8E-03
	90 %	1,1E-03	2,2E-03
	99 %	1,3E-04	2,6E-04
ANMERKUNG Diese Tabelle ist ein Auszug aus Tabelle B.3.			

Aus den Tabellen B.6 bis B.8 werden folgende Werte entnommen.

Für das Sensor-Teilsystem:

$$PFD_S = 2{,}3 \times 10^{-4}$$

Für das Logik-Teilsystem:

$$PFD_L = 4{,}8 \times 10^{-6}$$

Für das Stellglied-Teilsystem:

$$PFD_{FE} = 4{,}4 \times 10^{-3} + 8{,}8 \times 10^{-3}$$

$$= 1{,}3 \times 10^{-2}$$

Daraus ergibt sich für die Sicherheitsfunktion:

$$PFD_{SYS} = 2{,}3 \times 10^{-4} + 4{,}8 \times 10^{-6} + 1{,}3 \times 10^{-2}$$

$$= 1{,}3 \times 10^{-2}$$

$$\equiv \textbf{Sicherheits-Integritätslevel 1}$$

Um das System für den Sicherheits-Integritätslevel 2 zu ertüchtigen, kann eine der folgenden Maßnahmen durchgeführt werden:

a) Änderung des Wiederholungsprüfungsintervalls auf 6 Monate

$$PFD_S = 1{,}1 \times 10^{-4}$$

$$PFD_{L} = 2{,}6 \times 10^{-6}$$

$$PFD_{FE} = 2{,}2 \times 10^{-3} + 4{,}4 \times 10^{-3}$$

$$= 6{,}6 \times 10^{-3}$$

$$PFD_{SYS} = 6{,}7 \times 10^{-3}$$

≡ **Sicherheits-Integritätslevel 2**

b) Änderung des 1oo1-Absperrventils (das ist das Ausgangsgerät mit der niedrigeren Zuverlässigkeit) auf 1oo2 (Annahme $\beta = 10\ \%$ und $\beta_D = 5\ \%$)

$$PFD_{S} = 2{,}3 \times 10^{-4}$$

$$PFD = 4{,}8 \times 10^{-6}$$

$$PFD_{FE} = 4{,}4 \times 10^{-3} + 9{,}7 \times 10^{-4}$$

$$= 5{,}4 \times 10^{-3}$$

$$PFD_{SYS} = 5{,}6 \times 10^{-3}$$

≡ **Sicherheits-Integritätslevel 2**

B.3.2.5 Auswirkungen einer unvollständigen Wiederholungsprüfung

Fehler im Sicherheitssystem, die weder durch Diagnosetests noch durch die Wiederholungsprüfungen erkannt werden, können durch andere Methoden entdeckt werden, die sich aus Ereignissen ergeben, wie zum Beispiel aus einem gefährlichen Vorfall, der die Ausführung der Sicherheitsfunktion erfordert, oder während einer Überholung der Betriebsmittel. Wenn die Fehler nicht durch solche Methoden erkannt werden, sollte angenommen werden, dass sie über die Laufzeit der Betriebsmittel erhalten bleiben. Es wird eine übliches Intervall für die Wiederholungsprüfung von T_1 angenommen, wobei der Anteil der Fehler, der bei der Ausführung der Wiederholungsprüfung erkannt wird, als PTC (Deckungsgrad der Wiederholungsprüfung) bezeichnet wird und der Anteil der Fehler, der bei der Ausführung der Wiederholungsprüfung nicht erkannt wird, als (1-PTC) bezeichnet wird. Diese letzteren Fehler werden nur erkannt, wenn eine Anforderung an das sicherheitsbezogene System erfolgt, hierfür wird das Intervall der Anforderungen T_2 angenommen. Demzufolge sind das Intervall für die Wiederholungsprüfung (T_1) und das Intervall der Anforderung (T_2) maßgeblich für die wirksame Ausfalldauer.

Ein Beispiel hierfür wird unten stehend für eine 1oo2-Architektur gegeben. T_2 ist dabei die Zeit zwischen den Anforderungen an das System:

$$t_{CE} = \frac{\lambda_{DU}(PTC)}{\lambda_D}\left(\frac{T_1}{2} + MRT\right) + \frac{\lambda_{DU}(1 - PTC)}{\lambda_D}\left(\frac{T_2}{2} + MRT\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR$$

$$t_{GE} = \frac{\lambda_{DU}(PTC)}{\lambda_D}\left(\frac{T_1}{3} + MRT\right) + \frac{\lambda_{DU}(1 - PTC)}{\lambda_D}\left(\frac{T_2}{3} + MRT\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR$$

$$PFD_G = 2\left((1 - \beta_D)\lambda_{DD} + (1 - \beta)\lambda_{DU}\right)^2 t_{CE} t_{GE} + \beta_D \lambda_{DD} MTTR + \beta\lambda_{DU}(PTC)\left(\frac{T_1}{2} + MRT\right) +$$

$$\beta\lambda_{DU}(1 - PTC)\left(\frac{T_2}{2} + MRT\right)$$

Tabelle B.9 zeigt die zahlenmäßigen Ergebnisse eines 1oo2-Systems mit einer 100 %igen jährlichen Wiederholungsprüfung (T_1 = 1 Jahr) im Vergleich zu einer 90 %igen Wiederholungsprüfung, wobei die Anforderungsrate T_2 mit 10 Jahren angenommen wurde. Dieses Beispiel wurde unter der Annahme einer Ausfallrate von $0{,}5 \times 10^{-5}$ je Stunde, einem β-Wert von 10 % und einem β_D-Wert von 5 % berechnet.

Tabelle B.9 – Beispiel für eine unvollständige Wiederholungsprüfung

Architektur	*DC*	λ_D = 0,5E-05	
		100 %-Wieder-holungsprüfung	90 %-Wieder-holungsprüfung
		β = 10 % β_D = 5 %	β = 10 % β_D = 5 %
1oo2	0 %	2,7E-03	6,0E-03
	60 %	9,7E-04	2,0E-03
	90 %	2,3E-04	4,4E-04
	99 %	2,4E-05	4,4E-05

B.3.3 Mittlere Häufigkeit eines gefahrbringenden Ausfalls (für die Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung)

B.3.3.1 Verfahren für die Berechnung

Die Methode für die Berechnung der Wahrscheinlichkeit eines Ausfalls einer Sicherheitsfunktion eines sicherheitsbezogenen E/E/PE-Systems in der Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung ist gleich derjenigen für ein System in der Betriebsart mit niedriger Anforderungsrate (siehe B.2.1), außer dass die mittlere Wahrscheinlichkeit eines Ausfalls bei Anforderung (PFD_{SYS}) durch die mittlere Häufigkeit eines gefahrbringenden Ausfalls (PFH_{SYS}) ersetzt wird.

Die Gesamtwahrscheinlichkeit eines gefahrbringenden Ausfalls einer Sicherheitsfunktion eines sicherheitsbezogenen E/E/PE-Systems (PFH_{SYS}) wird bestimmt durch die Berechnung der Raten der gefahrbringenden Ausfälle aller Teilsysteme, die gemeinsam die Sicherheitsfunktion erbringen, und deren Addition. Da in diesem Anhang die Wahrscheinlichkeiten klein sind, kann dies wie folgt ausgedrückt werden:

$$PFH_{SYS} = PFH_S + PFH_L + PFH_{FE}$$

Dabei ist

– PFH_{SYS} die mittlere Häufigkeit eines gefahrbringenden Ausfalls der Sicherheitsfunktion des sicherheitsbezogenen E/E/PE-Systems;
– PFH_S die mittlere Häufigkeit eines gefahrbringenden Ausfalls des Sensor-Teilsystems;
– PFH_L die mittlere Häufigkeit eines gefahrbringenden Ausfalls des Logik-Teilsystems; und
– PFH_{FE} die mittlere Häufigkeit eines gefahrbringenden Ausfalls des Stellglied-Teilsystems.

B.3.3.2 Architekturen für die Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung

ANMERKUNG 1 Dieser Unterabschnitt sollte der Reihe nach gelesen werden, da Gleichungen, die für mehrere Architekturen gültig sind, nur bei ihrer ersten Verwendung angegeben werden. Siehe auch B.3.2.2.

ANMERKUNG 2 Die Gleichungen basieren auf den in B.3.1 aufgeführten Annahmen.

B.3.3.2.1 1oo1

Die Bilder B.4 und B.5 zeigen die zugehörigen Blockdiagramme.

$$\lambda_D = \lambda_{DU} + \lambda_{DD}$$

$$t_{CE} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{2} + MRT\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR$$

$$\lambda_{DU} = \lambda_D(1 - DC); \qquad \lambda_{DD} = \lambda_D DC$$

Wenn angenommen wird, dass das Sicherheitssystem bei jedem erkannten Ausfall die EUC in den sicheren Zustand bringt, wird für eine 1oo1-Architektur Folgendes erreicht:

$$PFH_G = \lambda_{DU}$$

B.3.3.2.2 1oo2

Die Bilder B.6 und B.7 zeigen die zugehörigen Blockdiagramme. Der Wert von t_{CE} ist wie in B.3.3.2.1 angegeben. Wenn angenommen wird, dass das Sicherheitssystem bei Erkennung eines Ausfalls in beiden Kanälen die EUC in den sicheren Zustand setzt, wird bei einem konservativen Ansatz Folgendes erreicht:

$$PFH_G = 2((1 - \beta_D)\lambda_{DD} + (1 - \beta)\lambda_{DU})(1 - \beta)\lambda_{DU} t_{CE} + \beta\lambda_{DU}$$

B.3.3.2.3 2oo2

Die Bilder B.8 und B.9 zeigen die zugehörigen Blockdiagramme. Wenn angenommen wird, dass jeder Kanal bei Erkennung irgendeines Fehlers in den sicheren Zustand gesetzt wird, wird für eine 2oo2-Architektur Folgendes erreicht:

$$PFH_G = 2\lambda_{DU}$$

B.3.3.2.4 1oo2D

Die Bilder B.10 und B.11 zeigen die zugehörigen Blockdiagramme.

$$\lambda_{SD} = \frac{\lambda}{2} DC$$

$$t_{CE}' = \frac{\lambda_{DU}\left(\frac{T_1}{2} + MRT\right) + (\lambda_{DD} + \lambda_{SD})MTTR}{\lambda_{DU} + \lambda_{DD} + \lambda_{SD}}$$

$$PFH_G = 2(1 - \beta)\lambda_{DU}((1 - \beta)\lambda_{DU} + (1 - \beta_D)\lambda_{DD} + \lambda_{SD})t_{CE}' + 2(1 - K)\lambda_{DD} + \beta\lambda_{DU}$$

B.3.3.2.5 2oo3

Die Bilder B.12 und B.13 zeigen die zugehörigen Blockdiagramme. Der Wert von t_{CE} ist wie in B.3.3.2.1 angegeben. Wenn angenommen wird, dass das Sicherheitssystem bei Erkennung eines Ausfalls in zwei beliebigen Kanälen die EUC in den sicheren Zustand setzt, wird bei einem konservativen Ansatz Folgendes erreicht:

$$PFH_G = 6((1 - \beta_D)\lambda_{DD} + (1 - \beta)\lambda_{DU})(1 - \beta)\lambda_{DU} t_{CE} + \beta\lambda_{DU}$$

B.3.3.2.6 1oo3

Die Bilder B.12 und B.13 zeigen die zugehörigen Blockdiagramme. Die Werte von t_{CE} und t_{GE} sind wie in B.3.3.2.1 und B.3.2.2.2 angegeben. Wenn angenommen wird, dass das Sicherheitssystem bei Erkennung eines Ausfalls in den drei Kanälen die EUC in den sicheren Zustand setzt, wird bei einem konservativen Ansatz Folgendes erreicht:

$$PFH_G = 6((1-\beta_D)\lambda_{DD} + (1-\beta)\lambda_{DU})^2(1-\beta)\lambda_{DU}t_{CE}t_{GE} + \beta\lambda_{DU}$$

B.3.3.3 Tabellen für die Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung

Tabelle B.10 – Mittlere Häufigkeit eines gefahrbringenden Ausfalls (in der Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung) für ein Wiederholungsprüfungsintervall von einem Monat und eine mittlere Dauer bis zur Wiederherstellung von 8 h

Architektur	*DC*	λ_D = 0,5E-07			λ_D = 2,5E-07			λ_D = 0,5E-06		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	5,0E-08			2,5E-07			5,0E-07		
	60 %	2,0E-08			1,0E-07			2,0E-07		
	90 %	5,0E-09			2,5E-08			5,0E-08		
	99 %	5,0E-10			2,5E-09			5,0E-09		
1oo2	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,0E-09	2,5E-08	5,0E-08	1,0E-08	5,0E-08	1,0E-07
	60 %	4,0E-10	2,0E-09	4,0E-09	2,0E-09	1,0E-08	2,0E-08	4,0E-09	2,0E-08	4,0E-08
	90 %	1,0E-10	5,0E-10	1,0E-09	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08
	99 %	1,0E-11	5,0E-11	1,0E-10	5,0E-11	2,5E-10	5,0E-10	1,0E-10	5,0E-10	1,0E-09
2oo2 (s. Anm. 2)	**0 %**	1,0E-07			5,0E-07			1,0E-06		
	60 %	4,0E-08			2,0E-07			4,0E-07		
	90 %	1,0E-08			5,0E-08			1,0E-07		
	99 %	1,0E-09			5,0E-09			1,0E-08		
1oo2D (s. Anm. 3)	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,0E-09	2,5E-08	5,0E-08	1,0E-08	5,0E-08	1,0E-07
	60 %	1,6E-09	3,2E-09	5,2E-09	8,0E-09	1,6E-08	2,6E-08	1,6E-08	3,2E-08	5,2E-08
	90 %	1,9E-09	2,3E-09	2,8E-09	9,5E-09	1,2E-08	1,4E-08	1,9E-08	2,3E-08	2,8E-08
	99 %	2,0E-09	2,0E-09	2,1E-09	1,0E-08	1,0E-08	1,0E-08	2,0E-08	2,0E-08	2,1E-08
2oo3	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,1E-09	2,5E-08	5,0E-08	1,1E-08	5,0E-08	1,0E-07
	60 %	4,0E-10	2,0E-09	4,0E-09	2,0E-09	1,0E-08	2,0E-08	4,1E-09	2,0E-08	4,0E-08
	90 %	1,0E-10	5,0E-10	1,0E-09	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08
	99 %	1,0E-11	5,0E-11	1,0E-10	5,0E-11	2,5E-10	5,0E-10	1,0E-10	5,0E-10	1,0E-09
1oo3	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,0E-09	2,5E-08	5,0E-08	1,0E-08	5,0E-08	1,0E-07
	60 %	4,0E-10	2,0E-09	4,0E-09	2,0E-09	1,0E-08	2,0E-08	4,0E-09	2,0E-08	4,0E-08
	90 %	1,0E-10	5,0E-10	1,0E-09	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08
	99 %	1,0E-11	5,0E-11	1,0E-10	5,0E-11	2,5E-10	5,0E-10	1,0E-10	5,0E-10	1,0E-09

Architektur	*DC*	λ_D =2,5E-06			λ_D = 0,5E-05			λ_D = 2,5E-05		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	2,5E-06			5,0E-06			2,5E-05		
	60 %	1,0E-06			2,0E-06			1,0E-05		
	90 %	2,5E-07			5,0E-07			2,5E-06		
	99 %	2,5E-08			5,0E-08			2,5E-07		
1oo2	**0 %**	5,4E-08	2,5E-07	5,0E-07	1,2E-07	5,2E-07	1,0E-06	9,5E-07	2,9E-06	5,3E-06
	60 %	2,1E-08	1,0E-07	2,0E-07	4,3E-08	2,0E-07	4,0E-07	2,7E-07	1,1E-06	2,1E-06
	90 %	5,1E-09	2,5E-08	5,0E-08	1,0E-08	5,0E-08	1,0E-07	5,5E-08	2,5E-07	5,0E-07
	99 %	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08	5,1E-09	2,5E-08	5,0E-08
2oo2 (s. Anm. 2)	**0 %**	5,0E-06			1,0E-05			5,0E-05		
	60 %	2,0E-06			4,0E-06			2,0E-05		
	90 %	5,0E-07			1,0E-06			5,0E-06		
	99 %	5,0E-08			1,0E-07			5,0E-07		
1oo2D (s. Anm. 3)	**0 %**	5,4E-08	2,5E-07	5,0E-07	1,2E-07	5,2E-07	1,0E-06	9,5E-07	2,9E-06	5,3E-06
	60 %	8,1E-08	1,6E-07	2,6E-07	1,6E-07	3,2E-07	5,2E-07	8,7E-07	1,7E-06	2,7E-06
	90 %	9,5E-08	1,2E-07	1,4E-07	1,9E-07	2,3E-07	2,8E-07	9,6E-07	1,2E-06	1,4E-06
	99 %	1,0E-07	1,0E-07	1,0E-07	2,0E-07	2,0E-07	2,1E-07	1,0E-06	1,0E-06	1,0E-06
2oo3	**0 %**	6,3E-08	2,6E-07	5,1E-07	1,5E-07	5,5E-07	1,0E-06	1,8E-06	3,6E-06	5,9E-06
	60 %	2,2E-08	1,0E-07	2,0E-07	4,9E-08	2,1E-07	4,1E-07	4,2E-07	1,2E-06	2,2E-06
	90 %	5,2E-09	2,5E-08	5,0E-08	1,1E-08	5,1E-08	1,0E-07	6,6E-08	2,6E-07	5,1E-07
	99 %	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08	5,4E-09	2,5E-08	5,0E-08
1oo3	**0 %**	5,0E-08	2,5E-07	5,0E-07	1,0E-07	5,0E-07	1,0E-06	5,1E-07	2,5E-06	5,0E-06
	60 %	2,0E-08	1,0E-07	2,0E-07	4,0E-08	2,0E-07	4,0E-07	2,0E-07	1,0E-06	2,0E-06
	90 %	5,0E-09	2,5E-08	5,0E-08	1,0E-08	5,0E-08	1,0E-07	5,0E-08	2,5E-07	5,0E-07
	99 %	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08	5,0E-09	2,5E-08	5,0E-08

ANMERKUNG 1 Diese Tabelle liefert Beispielwerte für PFH_G, berechnet unter Verwendung der Gleichungen aus B.3.3 und beruhend auf den Annahmen in B.3.1. Wenn das Sensor-, Logik- oder Stellglied-Teilsystem nur aus einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider besteht, ist PFH_G gleich PFH_S, PFH_L bzw. PFH_{FE} (siehe B.3.3.1).

ANMERKUNG 2 Diese Tabelle nimmt $\beta = 2 \times \beta_D$ an. Die Werte für β und β_D für 1oo1- und 2oo2-Architekturen beeinflussen die mittlere Häufigkeit eines gefahrbringenden Ausfalls nicht.

ANMERKUNG 3 Es werden gleiche Raten sicherer wie gefahrbringender Ausfälle angenommen sowie K = 0,98.

Tabelle B.11 – Mittlere Häufigkeit eines gefahrbringenden Ausfalls (in der Betriebsart mit hoher oder Anforderungsrate kontinuierlicher Anforderung) für ein Wiederholungsprüfungsintervall von drei Monaten und eine mittlere Dauer bis zur Wiederherstellung von 8 h

Architektur	DC	λ_D = 0,5E-07			λ_D = 2,5E-07			λ_D = 0,5E-06		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	5,0E-08			2,5E-07			5,0E-07		
	60 %	2,0E-08			1,0E-07			2,0E-07		
	90 %	5,0E-09			2,5E-08			5,0E-08		
	99 %	5,0E-10			2,5E-09			5,0E-09		
1oo2	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,1E-09	2,5E-08	5,0E-08	1,1E-08	5,0E-08	1,0E-07
	60 %	4,0E-10	2,0E-09	4,0E-09	2,0E-09	1,0E-08	2,0E-08	4,1E-09	2,0E-08	4,0E-08
	90 %	1,0E-10	5,0E-10	1,0E-09	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08
	99 %	1,0E-11	5,0E-11	1,0E-10	5,0E-11	2,5E-10	5,0E-10	1,0E-10	5,0E-10	1,0E-09
2oo2 (s. Anm. 2)	**0 %**	1,0E-07			5,0E-07			1,0E-06		
	60 %	4,0E-08			2,0E-07			4,0E-07		
	90 %	1,0E-08			5,0E-08			1,0E-07		
	99 %	1,0E-09			5,0E-09			1,0E-08		
1oo2D (s. Anm. 3)	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,1E-09	2,5E-08	5,0E-08	1,1E-08	5,0E-08	1,0E-07
	60 %	1,6E-09	3,2E-09	5,2E-09	8,0E-09	1,6E-08	2,6E-08	1,6E-08	3,2E-08	5,2E-08
	90 %	1,9E-09	2,3E-09	2,8E-09	9,5E-09	1,2E-08	1,4E-08	1,9E-08	2,3E-08	2,8E-08
	99 %	2,0E-09	2,0E-09	2,1E-09	1,0E-08	1,0E-08	1,0E-08	2,0E-08	2,0E-08	2,1E-08
2oo3	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,4E-09	2,5E-08	5,0E-08	1,2E-08	5,1E-08	1,0E-07
	60 %	4,0E-10	2,0E-09	4,0E-09	2,1E-09	1,0E-08	2,0E-08	4,3E-09	2,0E-08	4,0E-08
	90 %	1,0E-10	5,0E-10	1,0E-09	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08
	99 %	1,0E-11	5,0E-11	1,0E-10	5,0E-11	2,5E-10	5,0E-10	1,0E-10	5,0E-10	1,0E-09
1oo3	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,0E-09	2,5E-08	5,0E-08	1,0E-08	5,0E-08	1,0E-07
	60 %	4,0E-10	2,0E-09	4,0E-09	2,0E-09	1,0E-08	2,0E-08	4,0E-09	2,0E-08	4,0E-08
	90 %	1,0E-10	5,0E-10	1,0E-09	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08
	99 %	1,0E-11	5,0E-11	1,0E-10	5,0E-11	2,5E-10	5,0E-10	1,0E-10	5,0E-10	1,0E-09

Architektur	DC	λ_D = 2,5E-06			λ_D = 0,5E-05			λ_D = 2,5E-05		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	2,5E-06			5,0E-06			2,5E-05		
	60 %	1,0E-06			2,0E-06			1,0E-05		
	90 %	2,5E-07			5,0E-07			2,5E-06		
	99 %	2,5E-08			5,0E-08			2,5E-07		
1oo2	**0 %**	6,3E-08	2,6E-07	5,1E-07	1,5E-07	5,4E-07	1,0E-06	1,8E-06	3,6E-06	5,9E-06
	60 %	2,2E-08	1,0E-07	2,0E-07	4,9E-08	2,1E-07	4,1E-07	4,2E-07	1,2E-06	2,2E-06
	90 %	5,1E-09	2,5E-08	5,0E-08	1,1E-08	5,0E-08	1,0E-07	6,4E-08	2,6E-07	5,1E-07
	99 %	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08	5,2E-09	2,5E-08	5,0E-08
2oo2 (s. Anm. 2)	**0 %**	5,0E-06			1,0E-05			5,0E-05		
	60 %	2,0E-06			4,0E-06			2,0E-05		
	90 %	5,0E-07			1,0E-06			5,0E-06		
	99 %	5,0E-08			1,0E-07			5,0E-07		
1oo2D (s. Anm. 3)	**0 %**	6,3E-08	2,6E-07	5,1E-07	1,5E-07	5,4E-07	1,0E-06	1,8E-06	3,6E-06	5,9E-06
	60 %	8,2E-08	1,6E-07	2,6E-07	1,7E-07	3,3E-07	5,3E-07	1,0E-06	1,8E-06	2,8E-06
	90 %	9,5E-08	1,2E-07	1,4E-07	1,9E-07	2,3E-07	2,8E-07	9,6E-07	1,2E-06	1,4E-06
	99 %	1,0E-07	1,0E-07	1,0E-07	2,0E-07	2,0E-07	2,1E-07	1,0E-06	1,0E-06	1,0E-06
2oo3	**0 %**	9,0E-08	2,8E-07	5,3E-07	2,6E-07	6,3E-07	1,1E-06	4,5E-06	5,9E-06	7,6E-06
	60 %	2,6E-08	1,1E-07	2,0E-07	6,6E-08	2,2E-07	4,2E-07	8,5E-07	1,6E-06	2,5E-06
	90 %	5,4E-09	2,5E-08	5,0E-08	1,2E-08	5,1E-08	1,0E-07	9,3E-08	2,9E-07	5,3E-07
	99 %	5,1E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08	5,7E-09	2,6E-08	5,1E-08
1oo3	**0 %**	5,0E-08	2,5E-07	5,0E-07	1,0E-07	5,0E-07	1,0E-06	5,5E-07	2,5E-06	5,0E-06
	60 %	2,0E-08	1,0E-07	2,0E-07	4,0E-08	2,0E-07	4,0E-07	2,0E-07	1,0E-06	2,0E-06
	90 %	5,0E-09	2,5E-08	5,0E-08	1,0E-08	5,0E-08	1,0E-07	5,0E-08	2,5E-07	5,0E-07
	99 %	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08	5,0E-09	2,5E-08	5,0E-08

ANMERKUNG 1 Diese Tabelle liefert Beispielwerte für PFH_G, berechnet unter Verwendung der Gleichungen aus B.3.3 und beruhend auf den Annahmen in B.3.1. Wenn das Sensor-, Logik- oder Stellglied-Teilsystem nur aus einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider besteht, ist PFH_G gleich PFH_S, PFH_L bzw. PFH_{FE} (siehe B.3.3.1).

ANMERKUNG 2 Diese Tabelle nimmt $\beta = 2 \times \beta_D$ an. Die Werte für β und β_D beeinflussen die mittlere Häufigkeit eines gefahrbringenden Ausfalls bei 1oo1- und 2oo2-Architekturen nicht.

ANMERKUNG 3 Es werden gleiche Raten sicherer wie gefahrbringender Ausfälle angenommen sowie K = 0,98.

Tabelle B.12 – Mittlere Häufigkeit eines gefahrbringenden Ausfalls (in der Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung) für ein Wiederholungsprüfungsintervall von sechs Monaten und eine mittlere Dauer bis zur Wiederherstellung von 8 h

Architektur	DC	λ_D = 0,5E-07			λ_D = 2,5E-07			λ_D = 0,5E-06		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	5,0E-08			2,5E-07			5,0E-07		
	60 %	2,0E-08			1,0E-07			2,0E-07		
	90 %	5,0E-09			2,5E-08			5,0E-08		
	99 %	5,0E-10			2,5E-09			5,0E-09		
1oo2	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,3E-09	2,5E-08	5,0E-08	1,1E-08	5,1E-08	1,0E-07
	60 %	4,0E-10	2,0E-09	4,0E-09	2,0E-09	1,0E-08	2,0E-08	4,2E-09	2,0E-08	4,0E-08
	90 %	1,0E-10	5,0E-10	1,0E-09	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08
	99 %	1,0E-11	5,0E-11	1,0E-10	5,0E-11	2,5E-10	5,0E-10	1,0E-10	5,0E-10	1,0E-09
2oo2 (s. Anm. 2)	**0 %**	1,0E-07			5,0E-07			1,0E-06		
	60 %	4,0E-08			2,0E-07			4,0E-07		
	90 %	1,0E-08			5,0E-08			1,0E-07		
	99 %	1,0E-09			5,0E-09			1,0E-08		
1oo2D (s. Anm. 3)	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,3E-09	2,5E-08	5,0E-08	1,1E-08	5,1E-08	1,0E-07
	60 %	1,6E-09	3,2E-09	5,2E-09	8,0E-09	1,6E-08	2,6E-08	1,6E-08	3,2E-08	5,2E-08
	90 %	1,9E-09	2,3E-09	2,8E-09	9,5E-09	1,2E-08	1,4E-08	1,9E-08	2,3E-08	2,8E-08
	99 %	2,0E-09	2,0E-09	2,1E-09	1,0E-08	1,0E-08	1,0E-08	2,0E-08	2,0E-08	2,1E-08
2oo3	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,8E-09	2,6E-08	5,1E-08	1,3E-08	5,3E-08	1,0E-07
	60 %	4,1E-10	2,0E-09	4,0E-09	2,1E-09	1,0E-08	2,0E-08	4,5E-09	2,0E-08	4,0E-08
	90 %	1,0E-10	5,0E-10	1,0E-09	5,1E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08
	99 %	1,0E-11	5,0E-11	1,0E-10	5,0E-11	2,5E-10	5,0E-10	1,0E-10	5,0E-10	1,0E-09
1oo3	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,0E-09	2,5E-08	5,0E-08	1,0E-08	5,0E-08	1,0E-07
	60 %	4,0E-10	2,0E-09	4,0E-09	2,0E-09	1,0E-08	2,0E-08	4,0E-09	2,0E-08	4,0E-08
	90 %	1,0E-10	5,0E-10	1,0E-09	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08
	99 %	1,0E-11	5,0E-11	1,0E-10	5,0E-11	2,5E-10	5,0E-10	1,0E-10	5,0E-10	1,0E-09

Architektur	DC	λ_D = 2,5E-06			λ_D = 0,5E-05			λ_D = 2,5E-05		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	2,5E-06			5,0E-06			2,5E-05		
	60 %	1,0E-06			2,0E-06			1,0E-05		
	90 %	2,5E-07			5,0E-07			2,5E-06		
	99 %	2,5E-08			5,0E-08			2,5E-07		
1oo2	**0 %**	7,6E-08	2,7E-07	5,2E-07	2,1E-07	5,9E-07	1,1E-06	3,1E-06	4,7E-06	6,8E-06
	60 %	2,4E-08	1,0E-07	2,0E-07	5,7E-08	2,1E-07	4,1E-07	6,3E-07	1,4E-06	2,3E-06
	90 %	5,3E-09	2,5E-08	5,0E-08	1,1E-08	5,1E-08	1,0E-07	7,8E-08	2,7E-07	5,2E-07
	99 %	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08	5,4E-09	2,5E-08	5,0E-08
2oo2 (s. Anm. 2)	**0 %**	5,0E-06			1,0E-05			5,0E-05		
	60 %	2,0E-06			4,0E-06			2,0E-05		
	90 %	5,0E-07			1,0E-06			5,0E-06		
	99 %	5,0E-08			1,0E-07			5,0E-07		
1oo2D (s. Anm. 3)	**0 %**	7,6E-08	2,7E-07	5,2E-07	2,1E-07	5,9E-07	1,1E-06	3,1E-06	4,7E-06	6,8E-06
	60 %	8,4E-08	1,6E-07	2,6E-07	1,8E-07	3,3E-07	5,3E-07	1,2E-06	2,0E-06	2,9E-06
	90 %	9,5E-08	1,2E-07	1,4E-07	1,9E-07	2,3E-07	2,8E-07	9,8E-07	1,2E-06	1,4E-06
	99 %	1,0E-07	1,0E-07	1,0E-07	2,0E-07	2,0E-07	2,1E-07	1,0E-06	1,0E-06	1,0E-06
2oo3	**0 %**	1,3E-07	3,2E-07	5,5E-07	4,2E-07	7,7E-07	1,2E-06	8,4E-06	9,2E-06	1,0E-05
	60 %	3,3E-08	1,1E-07	2,1E-07	9,1E-08	2,4E-07	4,4E-07	1,5E-06	2,1E-06	2,9E-06
	90 %	5,8E-09	2,6E-08	5,1E-08	1,3E-08	5,3E-08	1,0E-07	1,3E-07	3,2E-07	5,6E-07
	99 %	5,1E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08	6,1E-09	2,6E-08	5,1E-08
1oo3	**0 %**	5,0E-08	2,5E-07	5,0E-07	1,0E-07	5,0E-07	1,0E-06	7,1E-07	2,7E-06	5,1E-06
	60 %	2,0E-08	1,0E-07	2,0E-07	4,0E-08	2,0E-07	4,0E-07	2,1E-07	1,0E-06	2,0E-06
	90 %	5,0E-09	2,5E-08	5,0E-08	1,0E-08	5,0E-08	1,0E-07	5,0E-08	2,5E-07	5,0E-07
	99 %	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08	5,0E-09	2,5E-08	5,0E-08

ANMERKUNG 1 Diese Tabelle liefert Beispielwerte für PFH_G, berechnet unter Verwendung der Gleichungen aus B.3.3 und beruhend auf den Annahmen in B.3.1. Wenn das Sensor-, Logik- oder Stellglied-Teilsystem nur aus einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider besteht, ist PFH_G gleich PFH_S, PFH_L bzw. PFH_{FE} (siehe B.3.3.1).

ANMERKUNG 2 Diese Tabelle nimmt $\beta = 2 \times \beta_D$ an. Die Werte für β und β_D beeinflussen die mittlere Häufigkeit eines gefahrbringenden Ausfalls bei 1oo1- und 2oo2-Architekturen nicht.

ANMERKUNG 3 Es werden gleiche Raten sicherer wie gefahrbringender Ausfälle angenommen sowie K = 0,98.

Tabelle B.13 – Mittlere Häufigkeit eines gefahrbringenden Ausfalls (in der Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung) für ein Wiederholungsprüfungsintervall von einem Jahr und eine mittlere Dauer bis zur Wiederherstellung von 8 h

Architektur	DC	λ_D = 0,5E-07			λ_D = 2,5E-07			λ_D = 0,5E-06		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	5,0E-08			2,5E-07			5,0E-07		
	60 %	2,0E-08			1,0E-07			2,0E-07		
	90 %	5,0E-09			2,5E-08			5,0E-08		
	99 %	5,0E-10			2,5E-09			5,0E-09		
1oo2	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,5E-09	2,5E-08	5,0E-08	1,2E-08	5,2E-08	1,0E-07
	60 %	4,0E-10	2,0E-09	4,0E-09	2,1E-09	1,0E-08	2,0E-08	4,3E-09	2,0E-08	4,0E-08
	90 %	1,0E-10	5,0E-10	1,0E-09	5,1E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08
	99 %	1,0E-11	5,0E-11	1,0E-10	5,0E-11	2,5E-10	5,0E-10	1,0E-10	5,0E-10	1,0E-09
2oo2 (s. Anm. 2)	**0 %**	1,0E-07			5,0E-07			1,0E-06		
	60 %	4,0E-08			2,0E-07			4,0E-07		
	90 %	1,0E-08			5,0E-08			1,0E-07		
	99 %	1,0E-09			5,0E-09			1,0E-08		
1oo2D (s. Anm. 3)	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,5E-09	2,5E-08	5,0E-08	1,2E-08	5,2E-08	1,0E-07
	60 %	1,6E-09	3,2E-09	5,2E-09	8,1E-09	1,6E-08	2,6E-08	1,6E-08	3,2E-08	5,2E-08
	90 %	1,9E-09	2,3E-09	2,8E-09	9,5E-09	1,2E-08	1,4E-08	1,9E-08	2,3E-08	2,8E-08
	99 %	2,0E-09	2,0E-09	2,1E-09	1,0E-08	1,0E-08	1,0E-08	2,0E-08	2,0E-08	2,1E-08
2oo3	**0 %**	1,1E-09	5,1E-09	1,0E-08	6,6E-09	2,6E-08	5,1E-08	1,6E-08	5,5E-08	1,0E-07
	60 %	4,1E-10	2,0E-09	4,0E-09	2,3E-09	1,0E-08	2,0E-08	5,0E-09	2,1E-08	4,1E-08
	90 %	1,0E-10	5,0E-10	1,0E-09	5,2E-10	2,5E-09	5,0E-09	1,1E-09	5,1E-09	1,0E-08
	99 %	1,0E-11	5,0E-11	1,0E-10	5,0E-11	2,5E-10	5,0E-10	1,0E-10	5,0E-10	1,0E-09
1oo3	**0 %**	1,0E-09	5,0E-09	1,0E-08	5,0E-09	2,5E-08	5,0E-08	1,0E-08	5,0E-08	1,0E-07
	60 %	4,0E-10	2,0E-09	4,0E-09	2,0E-09	1,0E-08	2,0E-08	4,0E-09	2,0E-08	4,0E-08
	90 %	1,0E-10	5,0E-10	1,0E-09	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08
	99 %	1,0E-11	5,0E-11	1,0E-10	5,0E-11	2,5E-10	5,0E-10	1,0E-10	5,0E-10	1,0E-09

Architektur	DC	λ_D = 2,5E-06			λ_D = 0,5E-05			λ_D = 2,5E-05		
		β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %	β = 2 %, β_D = 1 %	β = 10 %, β_D = 5 %	β = 20 %, β_D = 10 %
1oo1 (s. Anm. 2)	**0 %**	2,5E-06			5,0E-06			2,5E-05		
	60 %	1,0E-06			2,0E-06			1,0E-05		
	90 %	2,5E-07			5,0E-07			2,5E-06		
	99 %	2,5E-08			5,0E-08			2,5E-07		
1oo2	**0 %**	1,0E-07	2,9E-07	5,4E-07	3,1E-07	6,8E-07	1,1E-06	5,8E-06	6,9E-06	8,5E-06
	60 %	2,9E-08	1,1E-07	2,1E-07	7,4E-08	2,3E-07	4,2E-07	1,1E-06	1,7E-06	2,6E-06
	90 %	5,5E-09	2,5E-08	5,0E-08	1,2E-08	5,2E-08	1,0E-07	1,0E-07	3,0E-07	5,4E-07
	99 %	5,1E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08	5,6E-09	2,6E-08	5,0E-08
2oo2 (s. Anm. 2)	**0 %**	5,0E-06			1,0E-05			5,0E-05		
	60 %	2,0E-06			4,0E-06			2,0E-05		
	90 %	5,0E-07			1,0E-06			5,0E-06		
	99 %	5,0E-08			1,0E-07			5,0E-07		
1oo2D (s. Anm. 3)	**0 %**	1,0E-07	2,9E-07	5,4E-07	3,1E-07	6,8E-07	1,1E-06	5,8E-06	6,9E-06	8,5E-06
	60 %	8,9E-08	1,7E-07	2,7E-07	1,9E-07	3,5E-07	5,4E-07	1,7E-06	2,3E-06	3,2E-06
	90 %	9,6E-08	1,2E-07	1,4E-07	1,9E-07	2,3E-07	2,8E-07	1,0E-06	1,2E-06	1,4E-06
	99 %	1,0E-07	1,0E-07	1,0E-07	2,0E-07	2,0E-07	2,1E-07	1,0E-06	1,0E-06	1,0E-06
2oo3	**0 %**	2,1E-07	3,8E-07	6,1E-07	7,3E-07	1,0E-06	1,4E-06	1,6E-05	1,6E-05	1,6E-05
	60 %	4,6E-08	1,2E-07	2,2E-07	1,4E-07	2,9E-07	4,7E-07	2,8E-06	3,2E-06	3,8E-06
	90 %	6,6E-09	2,6E-08	5,1E-08	1,6E-08	5,6E-08	1,0E-07	2,1E-07	3,9E-07	6,2E-07
	99 %	5,2E-10	2,5E-09	5,0E-09	1,1E-09	5,1E-09	1,0E-08	6,9E-09	2,7E-08	5,1E-08
1oo3	**0 %**	5,1E-08	2,5E-07	5,0E-07	1,1E-07	5,1E-07	1,0E-06	1,4E-06	3,2E-06	5,5E-06
	60 %	2,0E-08	1,0E-07	2,0E-07	4,0E-08	2,0E-07	4,0E-07	2,6E-07	1,0E-06	2,0E-06
	90 %	5,0E-09	2,5E-08	5,0E-08	1,0E-08	5,0E-08	1,0E-07	5,1E-08	2,5E-07	5,0E-07
	99 %	5,0E-10	2,5E-09	5,0E-09	1,0E-09	5,0E-09	1,0E-08	5,0E-09	2,5E-08	5,0E-08

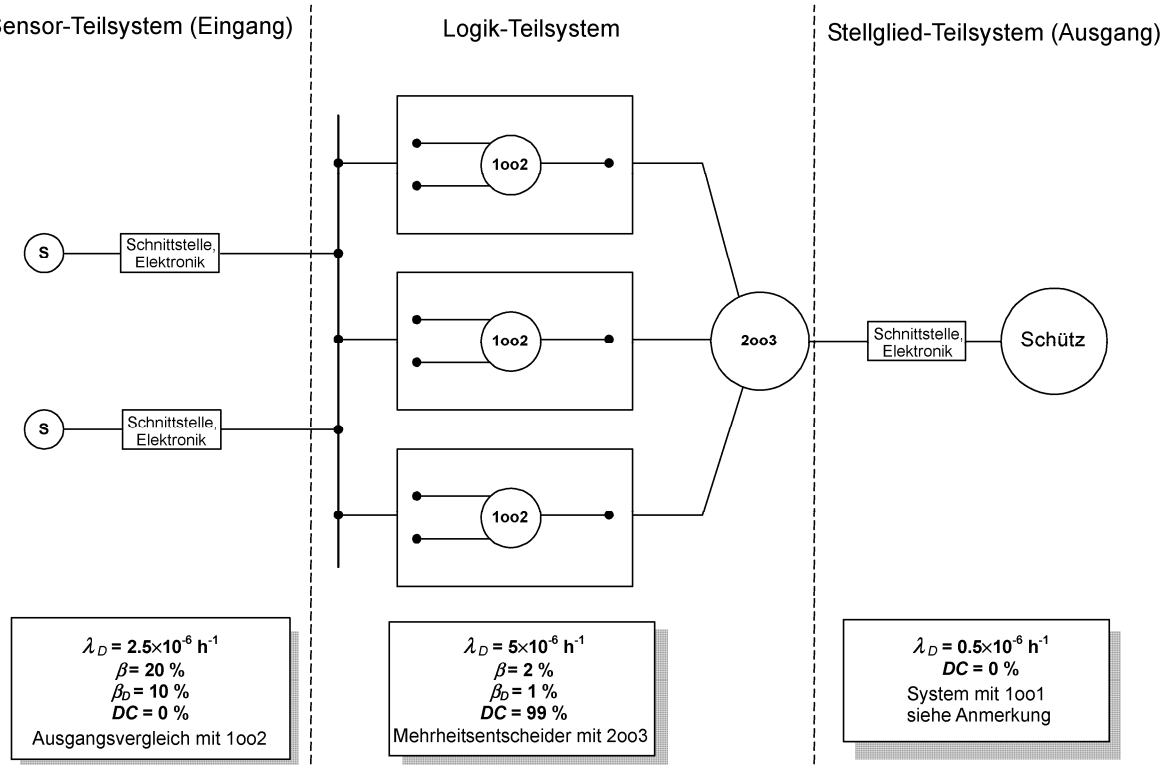
ANMERKUNG 1 Diese Tabelle liefert Beispielwerte für PFH_G, berechnet unter Verwendung der Gleichungen aus B.3.3 und beruhend auf den Annahmen in B.3.1. Wenn das Sensor-, Logik- oder Stellglied-Teilsystem nur aus einer Gruppe mit Ausgangsvergleicher oder Mehrheitsentscheider besteht, ist PFH_G gleich PFH_S, PFH_L bzw. PFH_{FE} (siehe B.3.3.1 und B.2.1).

ANMERKUNG 2 Diese Tabelle nimmt $\beta = 2 \times \beta_D$ an. Die Werte für β und β_D beeinflussen die mittlere Häufigkeit eines gefahrbringenden Ausfalls bei 1oo1- und 2oo2-Architekturen nicht.

ANMERKUNG 3 Es werden gleiche Raten sicherer wie gefahrbringender Ausfälle angenommen sowie K = 0,98.

B.3.3.4 Beispiel für die Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung

Es sei eine Sicherheitsfunktion betrachtet, die ein System mit SIL 2 erfordert. Es sei weiter angenommen, dass die erste Beurteilung auf der Basis früherer Erfahrungen im Hinblick auf die Systemarchitektur zu einer Gruppe von zwei Sensoren mit 1oo2-Ausgangsvergleich geführt hat. Das Logik-Teilsystem ist ein PE-System, das als redundantes 2oo3-System aufgebaut ist und ein einzelnes Schütz mit Abschaltfunktion ansteuert. Dies ist in Bild B.15 dargestellt. Bei der ersten Beurteilung wurde ein Wiederholungsprüfungsintervall von sechs Monaten angenommen.



ANMERKUNG Das Stellglied-Teilsystem hat einen Gesamtanteil sicherer Ausfälle von mehr als 60 %.

Bild B.15 – Architektur des Beispiels für die Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung

Tabelle B.14 – Mittlere Häufigkeit eines gefahrbringenden Ausfalls für das Sensor-Teilsystem in dem Beispiel für die Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung (sechs Monate Wiederholungsprüfungsintervall und 8 h *MTTR)*

Architektur	*DC*	λ_D = 2,5E-06		
		β = 2 % β_D = 1 %	β = 10 % β_D = 5 %	β = 20 % β_D = 10 %
1oo2	0 %	7,6E-08	2,7E-07	5,2E-07
	60 %	2,4E-08	1,0E-07	2,0E-07
	90 %	5,3E-09	2,5E-08	5,0E-08
	99 %	5,0E-10	2,5E-09	5,0E-09
ANMERKUNG Diese Tabelle ist ein Auszug aus Tabelle B.12.				

Tabelle B.15 – Mittlere Häufigkeit eines gefahrbringenden Ausfalls für das Logik-Teilsystem in dem Beispiel für die Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung (sechs Monate Wiederholungsprüfungsintervall und 8 h *MTTR*)

Architektur	*DC*	λ_D = 0,5E-05		
		β = 2 % β_D = 1 %	β = 10 % β_D = 5 %	β = 20 % β_D = 10 %
2oo3	0 %	4,2E-07	7,7E-07	1,2E-06
	60 %	9,1E-08	2,4E-07	4,4E-07
	90 %	1,3E-08	5,3E-08	1,0E-07
	99 %	1,0E-09	5,0E-09	1,0E-08
ANMERKUNG Diese Tabelle ist ein Auszug aus Tabelle B.12.				

Tabelle B.16 – Mittlere Häufigkeit eines gefahrbringenden Ausfalls für das Stellglied-Teilsystem in dem Beispiel für die Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung (sechs Monate Wiederholungsprüfungsintervall und 8 h *MTTR*)

Architektur	*DC*	λ_D = 0,5E-06
1oo1	0 %	5,0E-07
	60 %	2,0E-07
	90 %	5,0E-08
	99 %	5,0E-09
ANMERKUNG Diese Tabelle ist ein Auszug aus Tabelle B.12.		

Aus den Tabellen B.14 bis B.16 werden die folgenden Werte entnommen.

Für das Sensor-Teilsystem:

$$PFH_S = 5{,}2 \times 10^{-7}/\text{h}$$

Für das Logik-Teilsystem:

$$PFH_L = 1{,}0 \times 10^{-9}/\text{h}$$

Für das Stellglied-Teilsystem:

$$PFH_{FE} = 5{,}0 \times 10^{-7}/\text{h}$$

Daraus ergibt sicht für die Sicherheitsfunktion:

$$PFH_{SYS} = 5{,}2 \times 10^{-7} + 1{,}0 \times 10^{-9} + 5{,}0 \times 10^{-7}$$

$$= 1{,}02 \times 10^{-6}/\text{h}$$

$$\equiv \textbf{Sicherheits-Integritätslevel 1}$$

Um das System für den Sicherheits-Integritätslevel 2 zu ertüchtigen, kann eine der folgenden Maßnahmen ausgeführt werden:

a) Änderung der Bauart und Einbau des Eingangssensors, um Ausfälle infolge gemeinsamer Ursachen einzuschränken, dadurch wird β von 20 % auf 10 % und β_D von 10 % auf 5 % gebracht.

$PFH_S = 2{,}7 \times 10^{-7}/h$

$PFH_L = 1{,}0 \times 10^{-9}/h$

$PFH_{FE} = 5{,}0 \times 10^{-7}/h$

$PFH_{SYS} = 7{,}7 \times 10^{-7}/h$

≡ **Sicherheits-Integritätslevel 2**

b) Änderung des einzelnen Ausgangsgerätes in zwei Geräte in 1oo2 (β = 10 % und β_D = 5 %).

$PFH_S = 5{,}2 \times 10^{-7}/h$

$PFH_L = 1{,}0 \times 10^{-9}/h$

$PFH_{FE} = 5{,}1 \times 10^{-8}/h$

$PFH_{SYS} = 5{,}7 \times 10^{-7}/h$

≡ **Sicherheits-Integritätslevel 2**

B.4 Boolescher Ansatz

B.4.1 Allgemeines

Der Boolesche Ansatz umfasst die Verfahren zur Darstellung der Verknüpfung der logischen Funktionen einzelner Komponentenausfälle mit dem Ausfall des Gesamtsystems. Die auf dem Gebiet der Zuverlässigkeit angewendeten wichtigsten Booleschen Modelle sind Zuverlässigkeitsblockdiagramme (RBD), Fehlerbäume (FT), Ereignisablaufanalysen und Ursache-Folgen-Diagramme. Nur die ersten beiden Methoden werden hier betrachtet. Das Ziel all dieser Methoden ist die Darstellung der logischen Strukturen des Systems. Das Zeitverhalten ist jedoch in diesen Modellierungsverfahren nicht eingeschlossen. Daher ist bei der Berücksichtigung von Verhaltensmerkmalen (z. B. zeitabhängige Merkmale wie periodische Wiederholungsprüfungen) in den Berechnungen Vorsicht angebracht. Der erste Ansatz für die Anwendung des Booleschen Modells ist die Trennung der grafischen Darstellung von den Berechnungen. Dies wurde in den vorherigen Abschnitten beschrieben, wo RBD für die Modellierung der Struktur und Markov-Berechnungen für die Ermittlung von *PFD* und *PFH* verwendet werden. Weitere Überlegungen werden nun für Wahrscheinlichkeitsberechnungen mit RBD und FT erörtert.

Dieser Ansatz ist auf Bauteile begrenzt, die einigermaßen unabhängig voneinander agieren.

B.4.2 Modell des Zuverlässigkeitsblockdiagramms

Viele RBD-Beispiele wurden im Vorfeld schon angegeben und in Bild B.1 wird ein Beispiel für eine vollständige Sicherheitsschleife mit drei Sensoren (A, B, C) als 1oo3-System, einem Logiksystem (D) und zwei Stellgliedern (E, F) als 1oo2-System, dargestellt.

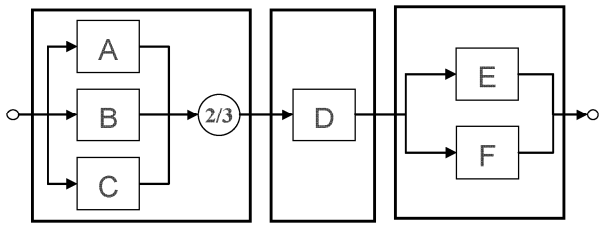


Bild B.16 – Zuverlässigkeitsblockdiagramm einer einfachen vollständigen Schleife mit Sensoren als 2oo3-Logik

Bild 16 zeigt eine ähnliche Schleife mit Sensoren als 2oo3-System. Es gibt drei wichtige Gründe für eine solche grafischen Darstellungsweise: sie ist immer noch sehr nah an der physikalischen Struktur des zu untersuchenden Systems, sie wird häufig von den Ingenieuren angewendet und ist eine gute Grundlage bei Diskussionen.

Der größte Nachteil ist, dass RBD eher eine Methode zur Darstellung als eine Methode zur Analyse selbst ist.

Für weitere Einzelheiten zu RBD siehe IEC 61508-7, C.6.4, und IEC 61078.

B.4.3 Fehlerbaummodell

Fehlerbäume (FT) haben exakt die gleichen Merkmale wie RBDs, bieten aber zusätzlich eine effektive Herleitungsmethode (top-down – hierarchisch) für die Analyse, was Ingenieuren, die die Zuverlässigkeit betrachten, hilft, schrittweise ein Modell zu entwickeln, vom unerwünschten Ereignis (top event) bis zu den Ausfällen einzelner Bauteile.

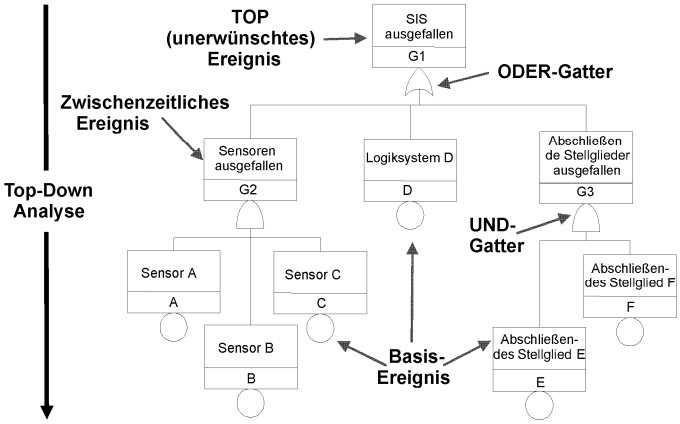


Bild B.17 – Einfacher Fehlerbaum entsprechend dem in Bild B.1 dargestellten Zuverlässigkeitsblockdiagramm

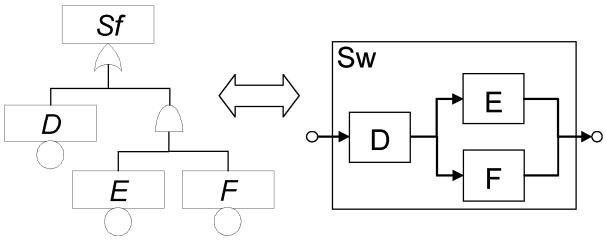
Bild B.17 zeigt einen Fehlerbaum, der genau dem in Bild B.1 dargestellten RBD entspricht, allerdings mit den ermittelten Ablaufschritten der hierarchischen (top-down) Analyse (zum Beispiel: Ausfall des sicherheitsbezogenen E/E/PE-Systems => Sensorausfall => Ausfall von Sensor A). In FT sind die seriellen Elemente durch „ODER-Gatter“ und die parallelen (d. h. redundanten) Elemente durch „UND-Gatter“ verknüpft.

Für weitere Einzelheiten zu FT siehe IEC 61508-7, B.6.6.5 und C.6.6.9, und IEC 61025.

B.4.4 *PFD*-Berechnungen

B.4.4.1 Allgemeiner Überblick

RBD und FT stellen dieselbe Sache dar, die Berechnungen können auf genau die gleiche Weise erfolgen. Bild B.18 zeigt FT und RBD, die gleichbedeutend sind und verwendet werden, um die Grundzüge der Berechnungen darzustellen.



ANMERKUNG Die kursiven Buchstaben in diesem Bild stehen für ausgefallene Komponenten, die nichtkursiven für betriebsfähige Komponenten.

Bild B.18 – Gleichartigkeit von Fehlerbaum und Zuverlässigkeitsblockdiagramm

Der kleine Fehlerbaum (FT) stellt die logische Funktion $Sf = D \cup (E \cap F)$ dar, wobei *Sf* der Systemausfall und *D*, *E*, *F* die Ausfälle der einzelnen Komponenten sind. Das kleine Zuverlässigkeitsblockdiagramm (RBD) zeigt die logische Funktion Sw = D ∩ (E∪F), wobei Sw das Funktionieren des Systems und D, E, F das Funktionieren der einzelnen Komponenten im Betrieb darstellen. Dann ist *Sf* = NOT Sw und *Sf* und Sw stehen für die gleiche Information (d. h. die logische Funktion und ihr Doppel).

Die primäre Verwendung von FT und RBD ist die Ermittlung der Kombinationen verschiedener Komponentenausfälle, die zum Gesamtsystemausfall führen. Dies sind die Minimalschnitte, so genannt, weil sie anzeigen, wo das RBD zu unterbrechen ist, damit ein vom Eingang gesendetes Signal nicht den Ausgang erreichen kann. Im vorliegenden Fall gibt es zwei Minimalschnitte: der Einzelausfall von (*D*) und der Doppelausfall (*E*, *F*).

Werden die mathematischen Grundlagen der Wahrscheinlichkeit auf die logischen Funktionen angewendet, führt dies direkt zu der Ausfallwahrscheinlichkeit des Systems P_{Sf} und man erhält:

$$P_{Sf} = P(D) + P(E \cap F) - P(D \cap E \cap F)$$

Bei unabhängigen Komponenten wird aus der Formel:

$$P_{Sf} = P_D + P_E P_F - P_D P_E P_F$$

Dabei ist

P_i die Wahrscheinlichkeit des Ausfalls der Komponente i.

Diese Formel ist zeitunabhängig und spiegelt nur die logische Struktur des Systems wider.

Demzufolge sind RDB und FT beide grundsätzlich statische, das heißt zeitunabhängige Modelle.

Wenn jedenfalls die Ausfallwahrscheinlichkeit jeder einzelnen Komponente zum Zeitpunkt *t* unabhängig davon ist, was mit den anderen Komponenten während [*0, t*] passiert ist, bleibt die obige Formel zu jeder Zeit gültig und es gilt:

$$P_{Sf}(t) = P_D(t) + P_E(t)P_F(t) - P_D(t)P_E(t)P_F(t)$$

Der Analytiker sollte überprüfen, ob die erforderlichen Näherungen zulässig sind oder nicht, und schließlich erhält man die momentane Nichtverfügbarkeit $U_{Sf}(t)$ des Systems:

$$U_{Sf}(t) = U_D(t) + U_E(t)U_F(t) - U_D(t)U_E(t)U_F(t)$$

Das Ergebnis ist, dass Fehlerbäume und Zuverlässigkeitsblockdiagramme die unmittelbare Berechnung der momentanen Nichtverfügbarkeit $U_{Sf}(t)$ des sicherheitsbezogenen E/E/PE-Systems erlauben und dass zusätzliche Berechnungen notwendig sind, und gemäß B.2.2 ist:

$$PFD_{avg}(T) = \frac{1}{T} MDT(T) = \frac{1}{T} \int_0^T U_{Sf}(t)dt$$

Diese grundsätzliche Vorgehensweise kann auf Minimalschnitte angewendet werden:

– Einzelausfall (*D*): $PFD^{D}(\tau) = \frac{1}{\tau} \int_0^\tau \lambda_D t.dt = \lambda_D \tau / 2$

– Doppelausfall (E, F): $PFD^{EF}(\tau) = \frac{1}{\tau} \int_0^\tau \lambda_E.\lambda_F.t^2 dt = \lambda_E \lambda_F \tau^2 / 3$

B.4.4.2 Berechnungen mit Werkzeugen für Fehlerbäume oder Zuverlässigkeitsblockdiagramme

Die oben behandelte Formel $U_{Sf}(t) = U_D(t) + U_E(t)U_F(t) – U_D(t)U_E(t)U_F(t)$ ist nur ein Spezialfall der sogenannten Poincaré-Formel. Wenn $Sf = \bigcup_i C_i$, wobei (C_i) die Minimalschnitte des Systems darstellt, erhält man allgemeiner:

$$P(\bigcup_{i-1}^{n} C_i) = \sum_{j=1}^{n} P(C_j) - \sum_{j=1}^{n} \sum_{i=1}^{j-1} P(C_j \bigcap C_i) + \sum_{j=3}^{n} \sum_{i=2}^{j-1} \sum_{k=1}^{j-1} P(C_j \bigcap C_i \bigcap C_k) - ...$$

Die Anzahl der Minimalschnitte steigt exponential mit der Anzahl der einzelnen Komponenten an. Damit ist die Poincaré-Formel wiederum durch die enorme Vervielfachung der Kombinationen umständlich von Hand zu lösen. Glücklicherweise ist dieses Problem in den letzten vierzig Jahren untersucht worden und es sind zahlreiche Algorithmen entwickelt worden, um diese Berechnungen beherrschen zu können. Die zurzeit neuesten Entwicklungen und leistungsfähigsten Methoden basieren auf den sogenannten binären Entscheidungsdiagrammen (en: Binary Decision Diagram – BDD), welche von einer ausgeklügelten Shannon-Zerlegung der logischen Funktion abgeleitet werden.

Eine Vielzahl von auf Fehlerbaummodellen basierenden kommerziellen Softwareanwendungen wird täglich von den für die Zuverlässigkeit zuständigen Ingenieuren in unterschiedlichsten industriellen Umfeldern (Kerntechnik, Mineralölwirtschaft, Luftfahrt, Fahrzeugbau usw.) angewendet. Diese können für die Berechnung von PFD_{avg} verwendet werden, Analytiker müssen aber aufgrund teilweise verwendeter falscher Berechnungen der PFD_{avg} sehr vorsichtig damit umgehen. Hauptfehlerursache ist die herkömmliche Zusammenführung der $PFD_{avg,i}$ der einzelnen Komponenten (im Allgemeinen einfach durch $\lambda_i \tau/2$ ermittelt), um ein Ergebnis zu erzielen, von dem angenommen wird, dass es der PFD_{avg} für das Gesamtsystem entspricht. Wie oben dargestellt ist dies falsch und nicht konservativ.

Die auf Fehlerbaummodellen basierenden Softwareanwendungen können jedenfalls angewendet werden, um aus der augenblicklichen Nichtverfügbarkeit der Komponenten $U_i(t)$ die augenblickliche Nichtverfügbarkeit des Systems $U_{Sf}(t)$ zu berechnen. Daraus kann dann für den betrachteten Zeitraum PFD_{avg} ermittelt werden. Dies geschieht abhängig von der verwendeten Software von dieser selbst oder anhand von Nebenrechnungen.

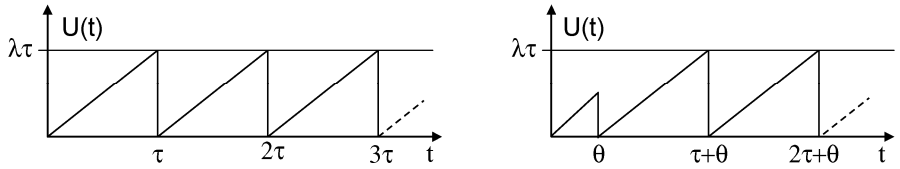


Bild B.19 – Augenblickliche Nichtverfügbarkeit *U*(*t*) einzelner, periodisch getesteter Komponenten

Der schon beschriebene Idealfall ist im linken Teil von Bild B.19 dargestellt:

$$U_i(t) = \lambda\zeta \text{ und } \zeta = t \bmod \tau$$

Dies ist eine sogenannte Sägezahnkurve, linear ansteigend von 0 nach $\lambda\tau$ und nach einem Test oder einer Reparatur (die als augenblicklich betrachtet werden, weil die EUC währenddessen angehalten wird) wieder neu ansteigt von 0.

Werden mehrere Komponenten in redundanten Strukturen verwendet, können die Tests versetzt auftreten, wie im rechten Teil von Bild B.19 dargestellt, wo das erste Intervall sich von den anderen unterscheidet. Dies hat keine Auswirkung auf die PFD_{avg} oder auf den Höchstwert, welcher in beiden Fällen $\lambda\tau/2$ und $\lambda\tau$ entspricht.

Bei Abweichungen von diesem Idealfall werden die Kurvenverläufe natürlich komplizierter. Im B.5.2 gibt es Anleitungen, um genauere Sägezahnkurven zu entwickeln, die in Bild B.19 dargestellten Kurvenverläufe sind aber für den Zweck dieses Unterabschnitts ausreichend.

Dies kann auf den kleinen in Bild B.18 dargestellten Fehlerbaum, wie in Bild B.20 dargestellt, angewendet werden (dabei bedeutet DU gefahrbringend unerkannt und CCF bedeutet Ausfall infolge gemeinsamer Ursache). Es wurde berücksichtigt, dass das System aus zwei redundanten Komponenten (E und F) besteht und (D) ein Ausfall infolge gemeinsamer Ursache dieser Komponente ist. Die Berechnungen wurden mit den folgenden Werten erzielt:

$$\lambda_{DU} = 3{,}5 \times 10^{-6}/\text{h},\ \tau = 4\,380 \text{ h und } \beta = 1\ \%$$

Es wurde ein geringer β-Faktor gewählt, um sicherzustellen, dass das Ergebnis für das unerwünschte Ereignis nicht durch CCF dominiert wird, und um ein besseres Verständnis für die praktische Anwendung zu erhalten.

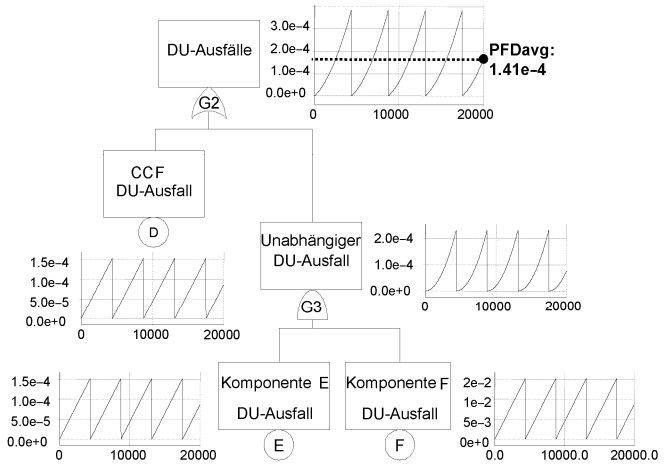


Bild B.20 – Prinzip der Berechnung von PFD_{avg} bei Verwendung von Fehlerbäumen

Die im linken Teil von Bild B.19 dargestellten Sägezahnkurven sind leicht als Eingaben für *D*, *E* und *F* zu erkennen. CCF (*D*) wird gleichzeitig mit *E* oder *F* getestet. Indem *E* und *F* zur selben Zeit alle 6 Monate getestet werden, wird also auch CCF (*D*) alle 6 Monate getestet.

Mit Hilfe eines der für Fehlerbaumberechnungen entwickelten Algorithmen ist es einfach, die Sägezahnkurven aller logischen Ausgänge darzustellen. Die Berechnung von PFD_{avg} erfolgt durch Mittelung der für das unerwünschte Ereignis erzielten Ergebnisse. Dies kann durch die Software für Fehlerbäume selbst oder anhand manueller Berechnungen durchgeführt werden. Man erhält $PFD_{avg} = 1{,}4 \times 10^{-4}$ und dies erfüllt nach dieser Norm den Ausfallgrenzwert für SIL 3 in der Betriebsart mit niedriger Anforderungsrate.

Wie in Bild B.20 dargestellt, sind die Graphen zwischen den Tests glatt. Daher gestaltet sich die Ermittlung des Mittelwertes als nicht so schwierig, vorausgesetzt, dass die Zeitpunkte der Tests identifiziert und berücksichtigt werden.

Es ist interessant festzustellen, dass die Sägezahnkurve für das unerwünschte Ereignis, sobald Redundanz vorhanden ist, zwischen den Tests nicht mehr linear verläuft (d. h. die Ausfallrate des Gesamtsystems ist nicht länger konstant).

Weiterhin ist interessant, den Einfluss von versetzten Tests auf PFD_{avg} anstatt gleichzeitiger an den redundanten Bauteilen zu messen. Dies wird in Bild B.21 veranschaulicht, wo die Tests des Bauteils F um 3 Monate zu denen des Bauteils E versetzt worden sind.

Dies hat mehrere wichtige Auswirkungen:

– Alle 3 Monate wird nun auf CCF getestet (d. h. jedes Mal, wenn E, und jedes Mal, wenn F getestet wird). Diese Wiederholungsprüfung wird doppelt so häufig durchgeführt wie in dem vorherigen Fall.

– Die Sägezahnkurve des unerwünschten Ereignisses zeigt ebenfalls eine doppelte Häufigkeit der Wiederholungsprüfung im Vergleich zu vorher.

– Die Sägezahnkurve hat um ihren Mittelwert weniger Varianz als im vorherigen Fall.

– Die PFD_{avg} hat sich auf $8{,}3 \times 10^{-5}$ reduziert: mit dieser neuen Testmethode entspricht das System SIL 4.

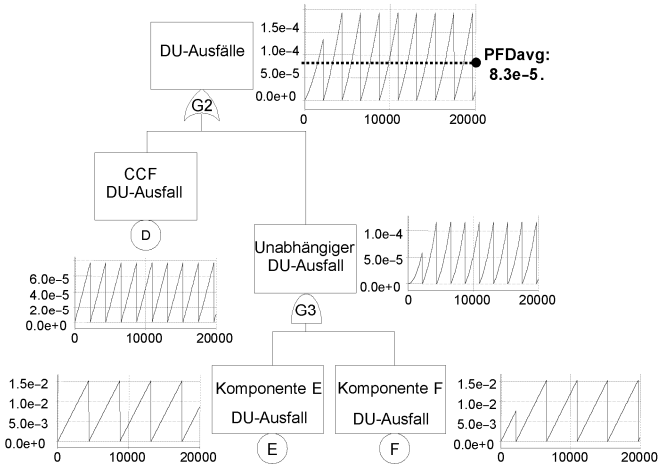


Bild B.21 – Auswirkung von versetzten Tests

Wenn die Tests versetzt und angemessene Verfahren realisiert sind, erhöht dies die Wahrscheinlichkeit der Erkennung von CCF und stellt eine wirksame Methode zu ihrer Verringerung in Systemen dar, die in der Betriebsart mit niedriger Anforderungsrate betrieben werden. In diesem Fall hat es sich von SIL 3 auf SIL 4 verbessert (aus Sicht der Hardwareausfälle und wenn weitere Anforderungen der Normenreihe IEC 61508 erfüllt werden).

Bild B.22 stellt die Sägezahnkurve dar, die erreicht wird, wenn eine Komponente G ($\lambda_{DU} = 7 \times 10^{-9}$/h und niemals getestet) und eine Komponente H ($\lambda_{DU} = 4 \times 10^{-8}$/h alle 2 Jahre getestet) in Reihe geschaltet und dem in Bild B.20 modellierten System hinzugefügt werden.

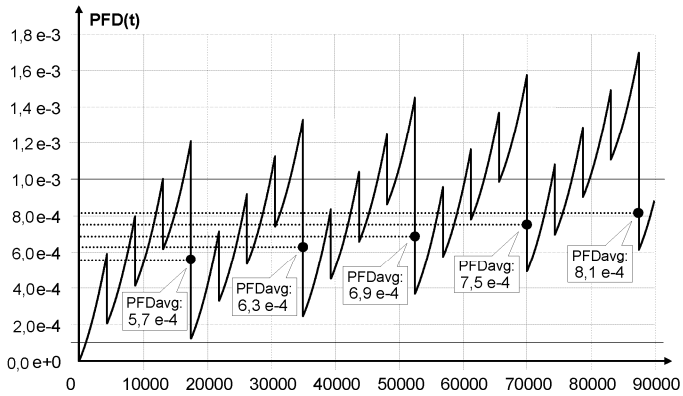


Bild B.22 – Beispiel eines komplexen Testmusters

Das nie getestete Bauteil G hat zweierlei Auswirkung: nach dem alle zwei Jahre durchgeführten Test geht $PFD(t)$ nie mehr zurück auf null und PFD_{avg} steigt kontinuierlich an (schwarze Kreise entsprechen der PFD_{avg} über den durch die zugehörige gepunktete Linie angezeigten Zeitraum).

Selbst, wenn die grundlegende Sägezahnkurve (wie in Bild B.19 dargestellt) sehr einfach ist, können sich die Ergebnisse bei dem unerwünschten Ereignis eher kompliziert gestalten, dies führt aber nicht zu besonderen Schwierigkeiten.

Dieser Abschnitt beschränkt sich darauf, das Prinzip der Berechnung anhand der Booleschen Modelle darzustellen. Unterabschnitt B.5.2 mit den Anwendungen des Markov-Modells gibt einige Anleitungen zum Entwurf anspruchsvollerer Eingangsägezahnkurven für Grundkomponenten.

Zusammenfassend kann festgestellt werden, dass, wenn die einzelnen Bauteile einigermaßen unabhängig sind, es kein Problem darstellt, die Berechnungen von PFD_{avg} für sicherheitsbezogenen E/E/PE-Systeme mit Hilfe der klassischen Booleschen Verfahren durchzuführen. Von der Theorie her ist dies nicht so einfach und der Analytiker, der die Untersuchung durchführt, sollte fundierte Kenntnisse über Wahrscheinlichkeitsberechnungen besitzen, um die mitunter falschen Anwendungen für die Berechnung von PFD_{avg} zu erkennen und auszuschließen. Unter der Voraussetzung, dass diese Vorsichtsmaßnahmen eingehalten werden, kann jede auf Fehlerbäumen basierende Softwareanwendung für die obigen Berechnungen verwendet werden.

Die Booleschen Verfahren können ebenfalls für *PFH*-Berechnungen angewendet werden, die theoretische Abhandlung ist aber nicht Zweck dieses informativen Anhangs.

B.5 Zustands-/Übergangs-Ansätze

B.5.1 Allgemeines

Die Booleschen Modelle sind grundsätzlich zeitunabhängig und die Einbeziehung der Zeit ist nur in speziellen Einzelfällen möglich. Dies ist ein wenig übertrieben und erfordert gute Kenntnisse der Wahrscheinlichkeitsberechnungen, um Fehler auszuschließen. Deshalb können stattdessen andere, dynamische Wahrscheinlichkeitsmodelle verwendet werden. Auf dem Gebiet der Zuverlässigkeitsberechnungen basieren diese grundsätzlich auf dem Ansatz der folgenden zwei Schritte:

– Feststellen aller Zustände des zu untersuchenden Systems;
– Analyse der Sprünge (Übergänge) von einem Zustand des Systems in einen anderen, entsprechend der während der gesamten Lebensdauer auftretenden Ereignisse.

Aus diesem Grund gehören sie zu der Gruppe der Zustands-/Übergangs-Modelle.

Der gegenwärtige allgemeine Ansatz besteht darin, Automaten zu konstruieren, die sich beim Auftreten von Ereignissen (Ausfälle, Reparaturen, Tests usw.) wie das zu untersuchende System verhalten. Sicherheitsbezogene E/E/PE-Systeme besitzen gemäß dieser Norm nur diskrete Zustände, dies entspricht der Konstruktion eines sogenannten endlichen Automaten (en: finite state automaton). Diese Modelle sind von der Art her

dynamisch und können auf verschiedene Weise umgesetzt werden: als grafische Darstellung, spezielle formale Sprachen oder allgemeine Programmiersprachen. In diesem Anhang werden zwei von ihnen vorgestellt, welche sehr unterschiedlich sind, sich aber ergänzen:

– das Markov-Modell, das zu Beginn des letzten Jahrhunderts entwickelt wurde. Es ist allgemein bekannt und wird für analytische Anwendungen eingesetzt;

– das Petri-Netz-Modell, welches in den sechziger Jahren entwickelt wurde. Es ist weniger gut bekannt (wird aber immer häufiger wegen seiner Flexibilität angewendet) und wird mit Hilfe von Monte-Carlo-Simulation gehandhabt.

Beide beruhen auf für den Anwender sehr hilfreichen grafischen Darstellungen. Weitere auf der Modellierung mit formalen Sprachen basierende Verfahren werden am Ende dieses Abschnitts kurz analysiert.

B.5.2 Markov-Ansatz

B.5.2.1 Grundsätze der Modellierung

Der Markov-Ansatz ist der älteste aller dynamischen Ansätze auf dem Gebiet der Zuverlässigkeitsberechnungen. Die Markov-Prozesse sind aufgeteilt auf solche, die „gedächtnislos“ sind (homogene Markov-Prozesse, bei denen alle Übergangsraten konstant sind) und die anderen (Semi-Markov-Prozesse). Da die Zukunft eines homogenen Markov-Prozesses nicht von seiner Vergangenheit abhängt, sind die analytischen Berechnungen ziemlich überschaubar. Mehr Schwierigkeiten treten bei den Semi-Markov-Prozessen auf, für die Monte-Carlo-Simulation angewendet werden kann. In diesem Teil der Normenreihe IEC 61508 werden nur homogene Markov-Prozesse betrachtet und zur Vereinfachung wird dafür der Begriff „Markov-Prozesse“ verwendet (siehe IEC 61508-7, C.6.4, und IEC 61165).

Die grundlegende Formel für Markov-Prozesse lautet wie folgt:

$$P_i(t+dt) = \sum_{k \neq i} P_k(t)\lambda_{ki}dt + P_i(t)(1 - \sum_{k \neq i} \lambda_{ik}dt)$$

In dieser Formel ist λ_{ki} die Rate für die Übergänge (z. B. Ausfall- oder Reparaturrate) vom Zustand *i* zum Zustand *k*. Selbsterklärend ist: die Wahrscheinlichkeit, sich zum Zeitpunkt *t* + *dt* im Zustand *i* zu befinden, entspricht der Wahrscheinlichkeit nach *i* zu springen (falls im anderen Zustand *k*) oder zwischen *t* und *t* + *dt* im Zustand *i* zu verbleiben (wenn schon in diesem Zustand).

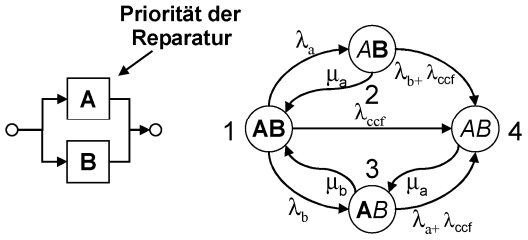


Bild B.23 – Markov-Graph, der das Verhalten eines Systems mit zwei Komponenten modelliert

Es gibt einen direkten Zusammenhang zwischen der obigen Gleichung und einer grafischen Darstellung wie in Bild B.23, welche ein System mit zwei Komponenten und einem einzelnen Reparaturteam (Priorität der Reparatur auf Komponente A) und einen Ausfall infolge gemeinsamer Ursache modelliert. In diesem Bild stellt **A** dar, das A funktionsfähig ist, und *A*, dass das Bauteil ausgefallen ist. Da die Erkennungszeiten berücksichtigt werden müssen, entsprechen μ_a und μ_b in Bild B.23 den Wiederherstellungsraten der Bauteile (d. h. $\mu_a = 1/MTTR_a$ und $\mu_b = 1/MTTR_b$).

Die Wahrscheinlichkeit, sich im Zustand 4 zu befinden, wird zum Beispiel einfach wie folgt berechnet:

$$P_4(t+dt) = [P_1(t)\lambda_{ccf} + P_2(t)(\lambda_b + \lambda_{ccf}) + P_3(t)(\lambda_a + \lambda_{ccf})]dt + P_4(t)(1 - \mu_a dt)$$

Dies führt zu einer vektoriellen Differentialgleichung,

$d\vec{P}(t)/dt = [M]\vec{P}(t)$, welche herkömmlich gelöst wird durch:

$$\vec{P}(t) = e^{t[M]}\vec{P}(0)$$

wobei

[*M*] die Markov-Matrix mit den Raten für die Übergänge und $\vec{P}(0)$ den Vektor der Anfangsbedingungen (grundsätzlich ein Spaltenvektor mit 1 für den fehlerlosen Zustand und 0 für die anderen) darstellt.

Selbst wenn ein Matrixexponential nicht genau die Eigenschaften eines gewöhnlichen Exponentials hat, ist es doch möglich zu schreiben:

$$\vec{P}(t) = e^{(t-t1)[M]}e^{t1[M]}\vec{P}(0) = e^{(t-t1)[M]}\vec{P}(t1)$$

Dies verdeutlicht die Grundeigenschaft des Markov-Prozesses: die Kenntnis der Wahrscheinlichkeiten der Zustände zu einem gegebenen Zeitpunkt *t1 fasst die gesamte Vergangenheit zusammen und reicht für die Berechnung aus, wie sich das System von t1 an entwickeln wird*. Dies ist sehr hilfreich für *PFD*-Berechnungen.

Schon vor längerer Zeit wurden leistungsfähige Algorithmen entwickelt und in die Softwareanwendungen übernommen, um die obigen Gleichungen zu lösen. Der Analytiker kann, wenn er mit diesen Anwendungen arbeitet, sich ganz auf die Konstruktion der Modelle konzentrieren und muss sich nicht mit den mathematischen Grundlagen beschäftigen, auch wenn er letztendlich trotzdem den Inhalt dieses Abschnitts verstehen muss.

Bild B.24 zeigt das Prinzip der *PFD*-Berechnungen:

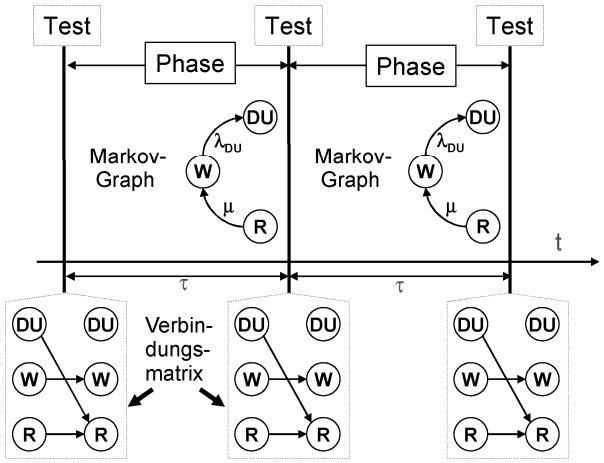


Bild B.24 – Prinzip der Modellierung eines mehrphasigen Markov-Prozesses

PFD-Berechnungen sind auf sicherheitsbezogene E/E/PE-Systeme bezogen, die in der Betriebsart mit niedriger Anforderungsrate arbeiten und periodischen Tests (durch Wiederholungsprüfungen) unterzogen werden. Für solche Systeme werden nur Reparaturen eingeleitet, wenn die Tests durchgeführt werden. Diese Tests sind singuläre Punkte entlang der Zeit, was aber kein Problem darstellt, da zur Behandlung ein mehrphasiger Markov-Ansatz verwendet werden kann.

Ein einfaches System mit einer periodisch getesteten einzelnen Komponente hat zum Beispiel, wie in Bild B.24 dargestellt, drei Zustände: in Betrieb (W), unerkannter gefahrbringender Ausfall (DU) und in Reparatur (R).

Zwischen den Tests ist sein Verhalten durch den Markov-Prozess im oberen Teil des Bildes B.24 modelliert: es kann zum Ausfall kommen (W → DU) oder in Reparatur sein (R → W). Da innerhalb eines Testintervalls keine Reparatur begonnen werden wird, gibt es keinen Übergang von DU zu R. Da die Diagnose des Ausfalls ausgeführt worden ist, bevor der Zustand R eingenommen wird, entspricht μ in Bild B.24 der Reparaturrate der Komponente (d. h. $\mu = 1/MRT$).

Wird ein Test durchgeführt (siehe Verbindungsmatrix in Bild B.24 [NA3)]), startet eine Reparatur, wenn ein Ausfall aufgetreten ist (DU → R), das Bauteil verbleibt in Betrieb, wenn es in einem betriebsfähigen Zustand war (W → W), und im sehr hypothetischen Fall, dass eine Reparatur gestartet wird, ohne dass der vorhergehende Test beendet war, verbleibt die Komponente in Reparatur (R → R). Um die Anfangsbedingungen zu Beginn des Zustands $i + 1$ aus den Wahrscheinlichkeiten der Zustände am Ende des Tests i zu berechnen, kann eine Verbindungsmatrix [L] verwendet werden. Dies ergibt die folgende Gleichung:

$$\begin{bmatrix} P_{DU}(0) \\ P_w(0) \\ P_R(0) \end{bmatrix}_{i+1} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P_{DU}(\tau) \\ P_w(\tau) \\ P_R(\tau) \end{bmatrix}_i \equiv \vec{P}_{i+1}(0) = [L]\vec{P}_i(\tau)$$

Setzt man für $\vec{P}_1(\tau)$ Werte ein, erhält man eine Rekursionsgleichung, die eine Berechnung der Anfangsbedingungen am Beginn aller Diagnosetestintervalle ermöglicht:

$$\vec{P}_{i+1}(0) = [L]\ e^{t[M]}\ \vec{P}_i(0)$$

Damit kann die Wahrscheinlichkeit zu jeder Zeit $t = i\tau + \zeta$ berechnet werden. Innerhalb des Testintervalls i zum Beispiel erhält man Folgendes:

$$\vec{P}(t) = \vec{P}_i(\zeta) = e^{\zeta[M]}\ \vec{P}_i(0), \quad (i-1)\ \tau \le t < i\tau, \quad \zeta = t \bmod \tau$$

Die augenblickliche Nichtverfügbarkeit erhält man direkt durch Aufsummierung der Wahrscheinlichkeiten der Zustände, in denen das System nicht verfügbar ist. Ein linearer Vektor (q_k) ist hilfreich, dies auszudrücken:

$$U(t) = \sum_{k=1}^{n} q_k P_k(t)$$

wobei $q_k = 1$ ist, wenn sich das System im nicht verfügbaren Zustand k befindet, andernfalls ist $q_k = 0$.

Für das einfache Modell erhält man: $PFD(t) = U(t) = P_{DU}(t) + P_R(t)$ und die Form der sich mit diesem Modell ergebenden Sägezahnkurve ist in Bild B.25 abgebildet.

PFD_{avg} wird auf die vorhergehend beschriebene Weise über MDT berechnet, die wiederum einfach aus den mittleren kumulierten Zeiten (en: Mean Cumulated Times), die in den Zuständen

$$\vec{MCT}(T) = \int_0^T \vec{P}(t)dt$$

verbracht wurden, zu berechnen ist.

Ähnlich wie für $\vec{P}(t)$ sind funktionierende Algorithmen für die Durchführung dieser Berechnung über [0, T] bekannt und abschließend erhält man: $PFD_{avg}(T) = \frac{1}{T}\sum_{k=1}^{n} q_k MCT_k(T)$

NA3) Nationale Fußnote: In IEC 61508-6:2010 wird auf Bild B.14 verwiesen.

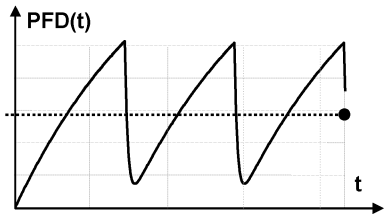


Bild B.25 – Sägezahnkurve bei mehrphasigem Markov-Ansatz

Wird diese Formel auf das in Bild B.24 dargestellte Modell angewendet, führt das zu:

$$PFD_{avg}(T) = \frac{1}{T}\left[MCT_{DU}(T) + MCT_{R}(T)\right].$$

Ist die EUC während der Reparatur abgeschaltet, kann die Formel auf den ersten Term reduziert werden.

Der schwarze Punkt in Bild B.25 stellt die PFD_{avg} der Sägezahnkurve über den gesamten Zeitraum der Berechnung dar.

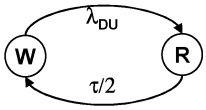


Bild B.26 – Angenähertes Markov-Modell

Zu beachten ist, dass die obige Berechnung oft mit Hilfe des in Bild B.26 dargestellten angenäherten Modells durchgeführt wird, wobei die Zustände DU und R zusammengefasst werden und $\tau/2$ (d. h. die mittlere Dauer zur Erkennung des Ausfalls) als entsprechende Dauer bis zur Wiederherstellung verwendet wird. Dies ist nur zulässig, wenn die Gleichungen des Markov-Modells vorher auf andere Weise aufgelöst wurden, um diese Gleichwertigkeit festzustellen. Diese Näherung ist nur anwendbar, wenn die Reparaturdauer vernachlässigbar ist. Weiterhin kann die Anwendung dieser Methode für große komplexe Systeme sehr schwierig sein.

Das einfache Modell in Bild B.24 kann leicht für realistischere Komponenten verbessert werden. Die Verbindungsmatrix in Bild B.27 modelliert eine Komponente, die sowohl eine Wahrscheinlichkeit γ besitzt, infolge des Tests auszufallen (d. h. ein echter Ausfall bei Anforderung), als auch eine Wahrscheinlichkeit σ, dass der Ausfall durch den Test nicht entdeckt wurde (durch menschliches Versagen).

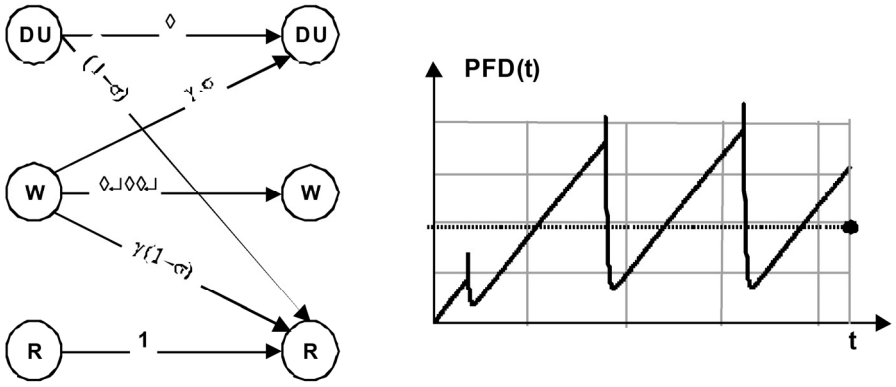


Bild B.27 – Auswirkungen von Ausfälle durch die Anforderung selbst

Die veränderte Sägezahnkurve und die für alle Tests festgestellten Sprünge stimmen mit der Wahrscheinlichkeit γ eines Ausfalls bei Anforderung überein. Der schwarze Punkt stellt wiederum die PFD_{avg} dar.

Wenn eine (redundante) Komponente abgeschaltet wird, um getestet zu werden, ist sie während der gesamten Ausführung der Tests nicht verfügbar und dies wirkt sich auf ihre PFD_{avg} aus. Aus diesem Grund

muss die Testdauer π berücksichtigt und eine zusätzliche Phase zwischen den Testintervallen eingefügt werden. Dies wird in Bild B.28 dargestellt, wobei die Zustände R und W in dieser Phase nur der Vollständigkeit halber modelliert sind.

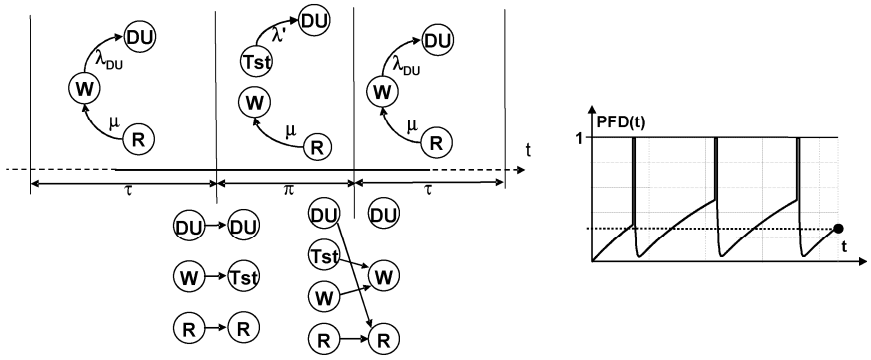


Bild B.28 – Modellierung des Einflusses der Testdauer

In diesem Markov-Modell ist das System in den Zuständen R, DU und Tst nicht verfügbar. Dies ist komplizierter als vorher, das Prinzip der Berechnung ist aber genau das gleiche. Das Verhalten der Sägezahnkurve ist im rechten Teil dargestellt. Das System ist während der Testdauer nicht verfügbar und dies kann den größten Beitrag zur PFD_{avg} darstellen.

Bei den vorherigen Markov-Graphen wurden nur die unerkannten gefahrbringenden Ausfälle berücksichtigt, der erkannte gefahrbringende Ausfall kann aber ebenso dargestellt werden. Der Unterschied besteht darin, dass die Reparatur sofort beginnt, wie in Bild B.29 dargestellt. Somit stellt μ_{DD} die Wiederherstellungsrate des Bauteils dar ($\mu_{DD} = 1/MTTR$), wenn μ_{DU} seine Reparaturrate ist ($\mu_{DU} = 1/MRT$).

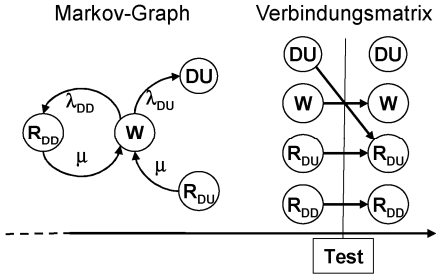


Bild B.29 – Mehrphasiges Markov-Modell mit DD- und DU-Ausfällen

Wenn erforderlich, sollten sichere Ausfälle dargestellt werden, hier wurden aber für die Darstellung Markov-Graphen ausgewählt, die so einfach wie möglich sind.

Das Hauptproblem bei Markov-Graphen ist, dass, wenn die Anzahl der Komponenten des zu untersuchenden Systems zunimmt, die Anzahl der Zustände exponential ansteigt. Deshalb ist das Aufstellen der Markov-Graphen und die Durchführung der oben aufgeführten Berechnungen ohne tiefgreifende Vereinfachungen von Hand sehr schnell nicht mehr zu bewältigen.

Mit Hilfe einer leistungsstarken auf Markov-Modellen basierenden Softwareanwendung können die Schwierigkeiten bei der Berechnung beherrscht werden. Es stehen viele solcher Anwendungen zur Verfügung und selbst wenn sie nicht unbedingt direkt für die Berechnung der PFD_{avg} verwendet werden können: die meisten führen Berechnungen für die augenblickliche Nichtverfügbarkeit durch, einige berechnen die in den Zuständen verbrachten mittleren kumulierten Zeiten und nur wenige ermöglichen eine mehrphasige Modellierung. Es ist jedoch kein Problem, alle für die Berechnung der PFD_{avg} anzupassen.

Die Modellierung betreffend können, wenn die Abhängigkeiten zwischen den Komponenten gering sind, der Markov- und der Boolesche Ansatz gemischt werden:

– die Markov-Modelle werden zur Ermittlung der augenblicklichen Nichtverfügbarkeit jeder Komponente eingesetzt,
– Fehlerbäume oder Zuverlässigkeitsblockdiagramme werden zum Zusammenfassen einzelner Nichtverfügbarkeiten verwendet, um die augenblickliche Nichtverfügbarkeit *PFD*(*t*) des Gesamtsystems zu berechnen,
– die Mittelung von *PFD*(*t*) ergibt dann PFD_{avg}.

Der gemischte Ansatz wurde im Unterabschnitt B.4 beschrieben und Sägezahnkurven wie die in den Bildern B.25, B.27 und B.28 können als Eingaben für die Fehlerbäume verwendet werden.

Sind die Abhängigkeiten zwischen den Komponenten nicht vernachlässigbar, gibt es einige Werkzeuge, die in der Lage sind, automatisch die Markov-Graphen zu erstellen. Diese basieren auf Modellen mit einer höheren Stufe als Markov-Modelle (z. B. Petri-Netze, formale Sprache). Aufgrund der sich explosionsartig vermehrenden Anzahl von Kombinationen der Zustände kann dieser Ansatz weiterhin zu Schwierigkeiten führen.

Dieser kombinierte Ansatz ist für die Modellierung komplexer Systeme sehr geeignet.

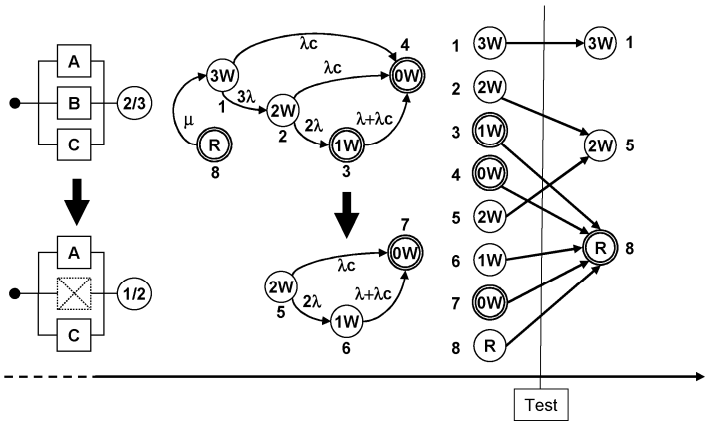


Bild B.30 – Logikänderung (2oo3 nach 1oo2) anstelle einer Reparatur des ersten Ausfalls

Das in Bild B.30 modellierte 2oo3-System besteht aus drei gleichzeitig getesteten Komponenten. Wird ein Ausfall erkannt, wechselt die Logik von 2oo3 zu 1oo2, da aus Gründen der Sicherheit eine 1oo2-Logik besser als eine 2oo3-Logik ist (aus Sicht von Ausfällen durch Fehlauslösung allerdings schlechter). Nur wenn ein zweiter Ausfall erkannt wurde, erfolgt die Reparatur durch Austausch aller drei Elemente durch drei neue. Dies führt Beschränkungen ein, die es unmöglich machen, das Verhalten des Gesamtsystems nur durch Kombination des unabhängigen Verhaltens seiner Bauteile aufzustellen.

B.5.2.2 Grundsätze von *PFH*-Berechnungen

Für Berechnungen der *PFH* kann der gleiche Typ der Modellierung eines mehrphasigen Markov-Prozesses für DU-Ausfälle, die durch Wiederholungsprüfungen erkannt werden, angewendet werden. Zur Vereinfachung werden nur die Grundsätze von *PFH*-Berechnungen für DD-Ausfälle aufgezeigt, da dies nur konventionelle (einphasige) Markov-Modelle erfordert. Selbstverständlich sollte für sicherheitsbezogene E/E/PE-Systeme in der Betriebsart mit kontinuierlicher Anforderung mit während der Wiederholungsprüfung erkannten DU-Ausfällen der mehrphasige Markov-Ansatz angewendet werden. Dies ändert aber nichts an der nachfolgend betrachteten Prinzipdarstellung.

Bild B.31 zeigt zwei Markov-Graphen des gleichen Systems, aus zwei redundanten Komponenten zusammengesetzt und mit einem Ausfall infolge gemeinsamer Ursache. Auf der linken Seite sind die Bauteile (A und B) reparierbar, auf der rechten nicht.

In beiden Graphen ist der Zustand 4 (*AB*) absorbierend. Nach einem Gesamtausfall verbleibt das System im fehlerhaften Zustand und $P(t) = P1(t) + P2(t) + P3(t)$ ist die Wahrscheinlichkeit, dass kein Ausfall über [*0, t*] aufgetreten ist. Dann ist $R(t) = P(t)$ die Zuverlässigkeit des Systems und $F(t) = 1 - R(t) = P4(t)$ seine Unzuverlässigkeit.

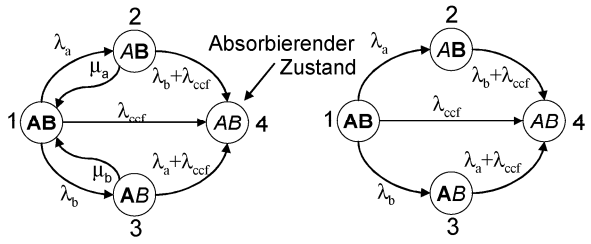


Bild B.31 – Markov-Graphen für die „Zuverlässigkeit“ mit einem absorbierenden Zustand

Dieses Zuverlässigkeitsmodell ist, wie unter B.2.3 erörtert, geeignet, diejenigen Fälle zu behandeln, in denen der Ausfall des sicherheitsbezogenen E/E/PE-Systems unmittelbar zu einer gefahrbringenden Situation führt. Auch in diesem Fall sind μ_a und μ_b die Wiederherstellungsraten der Bauteile (d. h. $\mu_a = 1/MTTR_a$ und $\mu_b = 1/MTTR_b$).

Mit solch einem Markov-Graph für die Zuverlässigkeit kann die *PFH* direkt mit $PFH = F(T)/T$ bestimmt werden. Zum Beispiel erhält man aus Bild B.31 direkt $PFH(T) = P4(T)/T$ (vorausgesetzt $P4(T) << 1$).

Er ermöglicht ebenfalls mit Hilfe der folgenden Formel die Berechnung der *MTTF* des Systems:

$$MTTF = \lim_{t\to\infty} \sum_{k=1}^{n} a_k MCT_k(t)$$

In dieser Formel ist $MCT_k(t)$ die mittlere kumulierte Zeit, die im Zustand k verbracht wurde, und $a_k = 1$, falls k ein funktionstüchtiger Zustand ist, andernfalls ist $a_k = 0$.

Eine obere Grenze wird erreicht mit:

$$PFH \approx 1/MTTF$$

Es sind leistungsfähige Algorithmen entwickelt worden und fast alle der auf Markov-Modellen basierenden Softwareanwendungen können für die Berechnungen von $F(T)$ und *MTTF* verwendet werden.

Die obigen Schätzungen der *PFH* sind in jedem Fall gültig, auch wenn keine konstante Ausfallrate des Gesamtsystems vorhanden ist (so wie für den Graphen rechts in Bild B.31). Die einzige Einschränkung ist es, einen Markov-Graph für die Zuverlässigkeit mit einem (oder mehreren) absorbierenden Zustand (Zuständen) zu verwenden. Dies gilt natürlich auch für die Anwendung eines mehrphasigen Modells.

Sind alle Zustände vollständig und schnell reparierbar, nähert sich die Ausfallrate des Gesamtsystems $\Lambda(t)$ schnell einem asymptotischen Wert $\Lambda_{as} = 1/MTTF$. In solchen Graphen sind alle Zustände, außer den perfekten und den absorbierenden, gewissermaßen Momentanzustände (da die *MTTRs* der Bauteile gegenüber deren *MTTF* gering sind). Dies ermöglicht eine direkte Bewertung der konstanten Ausfallraten des Gesamtsystems für alle Szenarien, beginnend vom perfekten Zustand bis zum absorbierenden Zustand. Der linke Markov-Graph in Bild B.31 modelliert solch ein vollständig und schnell reparierbares System. Es ergibt sich:

– $1\to4$: $\Lambda_{14} = \lambda_{ccf}$

– $1\to2\to4$: $\Lambda_{124} = \lambda_a \cdot (\lambda_b + \lambda_{ccf}) / [(\lambda_b + \lambda_{ccf}) + \mu_a] \approx \lambda_a \cdot (\lambda_b + \lambda_{ccf}) / \mu_a$

– $1\to3\to4$: $\Lambda_{134} = \lambda_b \cdot (\lambda_a + \lambda_{ccf}) / [(\lambda_a + \lambda_{ccf}) + \mu_b] \approx \lambda_b \cdot (\lambda_a + \lambda_{ccf}) / \mu_b$

Für das Szenario $1\to3\to4$ in der Formel oben ist λ_b die Übergangsrate für den Übergang aus dem perfekten Zustand heraus und $(\lambda_a + \lambda_{ccf}) / \mu_b$ die Wahrscheinlichkeit des Übergangs zu 4 und nicht zu 1 zurückzukehren, wenn sich das System im Zustand 3 befindet.

Letztendlich erhält man: $\Lambda_{as} = \Lambda_{12} + \Lambda_{124} + \Lambda_{134} = 1/MTTF$.

Dies kann leicht auf komplexe Markov-Graphen verallgemeinert werden, gilt aber nur für vollständig und schnell reparierbare Systeme, d. h. DD-Ausfälle.

Der rechte Markov-Graph in Bild B.31 ist nicht vollständig und schnell reparierbar. Die Anwendung der obigen Berechnungen würde deshalb zu fehlerhaften Ergebnissen führen.

Wird das sicherheitsbezogene E/E/PE-System in der Betriebsart mit kontinuierlicher Anforderung zusammen mit anderen Sicherheitsbarrieren angewendet, muss deren Verfügbarkeit beachtet werden. Dies wird in den beiden Graphen in Bild B.32 unten dargestellt: Es gibt keinen absorbierenden Zustand und das System wird nach einem Gesamtausfall repariert. $P(t) = P1(t) + P2(t) + P3(t)$ ist die Wahrscheinlichkeit der Betriebsfähigkeit des Systems zum Zeitpunkt t. Dann ist $A(t) = P(t)$ seine Verfügbarkeit und $U(t) = 1 - A(t) = P4(t)$ seine Nichtverfügbarkeit.

Dieser Fall unterscheidet sich sehr von dem in Bild B.31 dargestellten Fall und $R(t)$ und $A(t)$ sollten genau so wie $U(t)$ und $F(t)$ korrekt verwendet werden, wenn einwandfreie Ergebnisse erzielt werden sollen.

Für DD-Ausfälle ist der einfachste Weg die Berechnung der oberen Grenze von *PFH* durch *MDT* und *MUT*, wie in B.2.3 erläutert

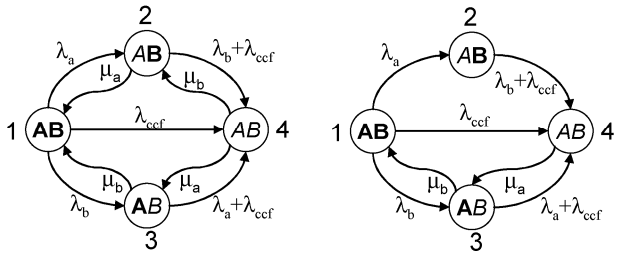


Bild B.32 – Markov-Graphen für die „Verfügbarkeit“ ohne absorbierende Zustände

Eine interessante Eigenschaft der Markov-Graphen für die Verfügbarkeit ist es, dass ein asymptotischer Beharrungszustand erreicht wird, wenn die Wahrscheinlichkeit, in einen gegebenen Zustand einzutreten, genauso groß ist wie die Wahrscheinlichkeit, diesen zu verlassen. Dies wird dargestellt durch:

– $P_{i,as} = \lim_{t\to\infty} P_i(t)$, den asymptotischen Wert von $P_i(t)$;

– $\lambda_i = \sum_{j\neq i} \lambda_{ij}$, die Übergangsrate von i in jeden anderen Zustand.

Jedes Mal, wenn das System den Zustand i erreicht, beträgt die mittlere Dauer in diesem Zustand $Mst_i = 1/\lambda_i$.

Dies ermöglicht die Berechnung von $MUT = \sum_i (1-q_i) P_{i,as} Mst_i$ und $MDT = \sum_i q_i P_{i,as} Mst_i$

wobei

$q_i = 0$ ist, wenn i ein funktionsfähiger Zustand ist, andernfalls ist $q_i = 1$.

Letztendlich erhält man Folgendes: $PFH = 1/(MUT + MDT) = 1/\sum_i P_{i,as} Mst_i = 1/\sum_i \frac{P_{i,as}}{\lambda_i}$

Es sollte beachtet werden, dass die Anzahl der über $[0, T]$ festgestellten Ausfälle gegeben ist durch: $n = T/\sum_i \frac{P_{i,as}}{\lambda_i}$

Da die Mehrzahl der auf Markov-Modellen basierenden Softwareanwendungen in der Lage ist, die asymptotischen Wahrscheinlichkeiten aufzufinden, sollte es keine Schwierigkeiten geben, die oben aufgeführten Berechnungen auszuführen.

Sollte der zu betrachtende Zeitraum in dem Prozess zu kurz sein für eine Markov-Näherung, kann die *PFH* berechnet werden mit: $w(t) = \sum_{i \neq f} \lambda_{if} P_i(t)$

Dies ergibt: $PFH(T) = \sum_{i \neq f} \lambda_{if} \frac{\int_0^T P_i(t)dt}{T} = \frac{\sum_{i \neq f} \lambda_{if} MCT_i(T)}{T}$

Hier dürfte es ebenfalls keine Schwierigkeiten geben, die Berechnung mit einer auf Markov-Modellen basierenden Softwareanwendung durchzuführen, die die kumulierte Dauer in jedem Zustand bestimmt.

Im Fall des vollständig und schnell reparierbaren Systems (DD-Ausfälle) nähert sich die Vesely-Rate $\Lambda_V(t)$ sehr schnell dem asymptotischen Wert Λ_{as}, welcher eine gute Näherung der konstanten Ausfallrate für das Gesamtsystem ist. Deshalb kann in diesem Fall die *PFH* auf die gleiche Weise berechnet werden wie für die Zuverlässigkeit.

Für DU-Ausfälle ist dies wegen der mehrphasigen Modellierung schwieriger. Die oben aufgeführte Formel kann verallgemeinert werden zu: $PFH(T) = \frac{\sum_{\varphi=1}^{n} \sum_{i \neq f} \lambda_{if} MCT_i(T_\varphi)}{\sum_{\varphi=1}^{n} T_\varphi}$

In dieser Formel ist T_φ die Dauer der Phase φ.

Mehrphasige Markov-Modelle erreichen im Allgemeinen den Beharrungszustand, wenn die Wahrscheinlichkeit, den gegebenen Zustand zu verlassen, genauso groß ist wie die Wahrscheinlichkeit, in diesen Zustand einzutreten. Die asymptotischen Werte haben, wie oben beschrieben, damit nichts zu tun, können aber für die obige Formel angewendet werden.

Zusammenfassend kann gesagt werden, dass der Markov-Ansatz viele Möglichkeiten bietet, die *PFH* eines sicherheitsbezogenen E/E/PE-Systems in der Betriebsart mit kontinuierlicher Anforderung zu berechnen. Gute Kenntnisse der mathematischen Grundlagen sind jedoch für dessen richtige Anwendung erforderlich.

B.5.3 Ansätze basierend auf Petri-Netzen und Monte-Carlo-Simulation

B.5.3.1 Grundsätze der Modellierung

Die Modellierung eines endlichen Automaten, dessen Verhalten dem zu untersuchenden sicherheitsbezogenen E/E/PE-System weitestgehend entspricht, stellt eine wirkungsvolle Art dar, dynamische Systeme zu modellieren. Petri-Netze (siehe IEC 61508-7, B.2.3.3 und B.6.6.10) haben sich aus folgenden Gründen als sehr effizient für diesen Zweck erwiesen:

– sie sind graphisch einfach zu handhaben;
– die Größe der Modelle wächst linear entsprechend der Anzahl der zu modellierenden Komponenten;
– sie sind sehr flexibel und ermöglichen die Modellierung nahezu aller Arten von Beschränkungen;
– sie sind ein ausgezeichnetes Hilfsmittel für die Monte-Carlo-Simulation (siehe IEC 61508-7, B.6.6.8).

Ursprünglich in den 1960er-Jahren für die formale Prüfung von Automaten entwickelt, wurden sie schnell in den 1970er-Jahren von den für die Zuverlässigkeit zuständigen Ingenieuren für die automatische Konstruktion von umfangreichen Markov-Graphen und in den 1980er-Jahren zum Zweck der Monte-Carlo-Simulation übernommen.

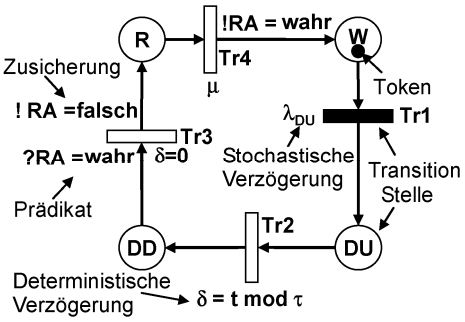


Bild B.33 – Petri-Netz für die Modellierung einer einzelnen periodisch getesteten Komponente

Das typische Sub-Petri-Netz für ein einfache periodisch getestete Komponente besteht aus drei Teilen:

1) dem statischen Teil (d. h. einer Zeichnung):
   a) Stellen (Kreisen) entsprechend möglichen Zuständen;
   b) Übergängen (Rechtecken) entsprechend möglichen Ereignissen;
   c) Pfeilen stromaufwärts (von den Stellen zu den Übergängen) für die Validierung der Übergänge;
   d) Pfeilen stromabwärts (von den Übergängen zu den Stellen), die anzeigen was passiert, wenn Übergänge aktiviert werden.
2) dem Teil der Zeitsteuerung:
   a) stochastischen Zeitverzögerungen, welche die ablaufenden zufälligen Verzögerungen darstellen, bevor die Ereignisse stattfinden;
   b) deterministischen Zeitverzögerungen, welche die bekannten ablaufenden Verzögerungen darstellen, bevor die Ereignisse stattfinden.
3) dem dynamischen Teil:
   a) Token (schwarze Punkte), die sich bei auftretenden Ereignissen verschieben, um anzuzeigen, welche von den möglichen Zuständen gegenwärtig eingetreten sind;
   b) Prädikaten (jede Formel, die wahr oder falsch sein kann) für die Validierung der Übergänge;
   c) Zusicherungen (jede Gleichung) für die Aktualisierung einiger Variablen, wenn ein Übergang aktiviert wird.

Zusätzlich erlauben einige Regeln die Validierung und die Aktivierung eines Übergangs:

4) Validierung eines Übergangs (d. h. die Bedingungen, unter denen das entsprechende Ereignis auftreten kann)
   a) alle Stellen stromaufwärts besitzen mindestens ein Token;
   b) alle Prädikate müssen „wahr“ sein.
5) Aktivierung eines Übergangs (d. h. was passiert, wenn das entsprechende Ereignis auftritt)
   a) aus den Stellen stromaufwärts wird ein Token entnommen;
   b) den Stellen stromabwärts wird ein Token hinzugefügt;
   c) die Zusicherungen (en: assertions) werden aktualisiert.

Die meisten der im Zusammenhang mit Petri-Netzen stehenden Begriffe sind oben genannt worden, die noch ausstehenden werden erwähnt, wenn sie benötigt werden.

B.5.3.2 Grundsätze der Monte-Carlo-Simulation

Die Monte-Carlo-Simulation besteht aus der Animation von Verhaltensmodellen, bei der mit Hilfe von Zufallszahlen bestimmt wird, wie oft das System in bestimmten Zuständen verbleibt, die entweder durch zufällige oder deterministische Verzögerungen bestimmt werden (siehe auch IEC 61508-7, B.6.6.8).

Dies lässt sich gut anhand des in Bild B.33 dargestellten Petri-Netzes erläutern:

– Zu Beginn befindet sich das Token an der Stelle *W* und die Komponente ist in einem funktionsfähigen Zustand.
– Es kann nur ein Ereignis aus diesem Zustand hervorgehen – ein unentdeckter gefahrbringender Ausfall (Übergang *Tr1* ist gültig und schwarz markiert).
– Die Verweildauer in diesem Zustand ist stochastisch und wird bestimmt von einer exponentiellen Verteilung des Parameters λ_{DU}. Die Monte-Carlo-Simulation besteht aus der Aktivierung einer Zufallszahl (siehe unten), damit die Verzögerungszeit *d1*, bis der Ausfall eintritt, ermittelt werden kann (d. h. *Tr1* wird aktiviert).
– Wenn *d1* abgelaufen ist, wird *Tr1* aktiviert und das Token wird auf die Stelle *DU* verschoben (genauer gesagt wird ein Token an der Stelle *W* entnommen und ein Token an der Stelle *DU* hinzugefügt).
– Die Komponente erreicht den gefahrbringenden unentdeckten Zustand und *Tr2* wird gültig.
– Nach einer deterministischen Verzögerung *d2* (*d2* = *t* modulo τ, wenn *t* die augenblickliche Zeit und τ das Diagnosetestintervall sind) erfolgt die Erkennung des gefahrbringenden Ausfalls. Damit wird das Testintervall simuliert.
– Wenn *t2* abgelaufen ist, d. h. der gefahrbringende Ausfall erkannt ist, tritt das Token in die Stelle *DD* ein. Die Komponente wartet nun darauf, repariert zu werden und *Tr3* wird gültig.
– Die Verzögerung *d3* für die Aktivierung von *Tr3* (Beginn der Reparatur) hängt nicht von der Komponente selbst, aber von der Verfügbarkeit der durch die Nachricht *RA* dargestellten Reparaturressourcen ab. Dies wird durch Vorgänge in einem anderen Teil des vollständigen Petri-Netzes bestimmt, der nicht in Bild B.33 dargestellt ist.
– Die Reparatur beginnt, sobald das Reparaturteam verfügbar ist (d. h. das Prädikat *?RA = wahr* wird wahr) und der Token tritt in die Stelle *R* ein. Die Reparaturressourcen werden unverzüglich für einen anderen Vorgang unverfügbar und die Zusicherung *!RA = falsch* wird für eine Aktualisierung des Wertes von *RA* verwendet. Damit wird jede weitere Reparatur zum gleichen Zeitpunkt verhindert.
– Der stochastische Übergang *Tr4* (d. h. die Beendigung der Reparatur) wird gültig und die Verzögerung *d4* kann durch Aktivierung einer Zufallszahl entsprechend der Reparaturrate μ bestimmt werden.
– Wenn *d4* abgelaufen ist, wird *Tr4* aktiviert und die Komponente kehrt in einen funktionsfähigen Zustand zurück (das Token tritt in die Stelle *W* ein). Die Reparaturressourcen werden wieder verfügbar und *RA* wird durch die Zusicherung *!RA = wahr* aktualisiert.
– Das geht immer so weiter, solange wie die Aktivierung des nächsten gültigen Übergangs innerhalb des zu betrachtenden Zeitraums [*0, T*] liegt.

Liegt die nächste Aktivierung nicht mehr innerhalb von [*0, T*], wird die Simulation angehalten und eine Historie der Komponente ist erreicht. Entlang des gesamten Ablaufs der Historie können die relevanten Parameter aufgezeichnet werden, wie die mittlere Dauer für die Markierung der Stellen (d. h. das Verhältnis der Verweildauer eines Token in der Stelle über den Zeitraum T), die Aktivierungsrate des Übergangs, die Zeit bis zum ersten Auftreten eines gegebenen Ereignisses usw.

Das Prinzip der Monte-Carlo-Simulation ist es, eine große Anzahl solcher Historien zu erstellen und über die Ergebnisse eine klassische Statistik durchzuführen, um die relevanten Parameter zu beurteilen.

Im Gegensatz zu den analytischen Berechnungen ist es mit der Monte-Carlo-Simulation ohne Probleme möglich, deterministische und zufällige Verzögerungen, die aus der kumulativen Wahrscheinlichkeitsverteilung $F(d)$ und den gleichmäßig über [*0, 1*] verteilten Zufallszahlen z_i simuliert werden können, zu mischen. Solche Zufallszahlen sind in fast jeder Programmiersprache vorhanden und zur Ausführung sind leistungsstarke Algorithmen verfügbar.

Dann wird durch die Operation $d_i = F^{-1}(z_i)$ eine Stichprobe (d_i), verteilt nach $F(d)$, aus einer Stichprobe (z_i) gezogen. Dies ist sehr einfach, wenn die analytische Form von $F^{-1}(z)$ zum Beispiel als exponentiell verteilte Verzögerungen $d_i = -\frac{1}{\lambda_{DU}} Log(z_i)$ verfügbar ist.

Hinsichtlich der Genauigkeit der Berechnungen eines gegebenen simulierten Parameters *X* erlauben grundlegende Statistikverfahren die Berechnung des Mittelwerts, der Varianz, der Standardabweichung und der Konfidenz der Stichprobe (X_i), welche simuliert wurde:

– Mittelwert: $\overline{X} = \frac{\sum_i x_i}{N}$

– Varianz: $\sigma^2 = \frac{\sum_i (x_i - \overline{X})^2}{N}$ und Standardabweichung: σ

– 90 % Konfidenzintervall um $\overline{X}$: $Conf = 1{,}64 \frac{\sigma}{\sqrt{N}}$

Demnach kann die Genauigkeit der Ergebnisse immer abgeschätzt werden, wenn die Monte-Carlo-Simulation angewendet wird. Zum Beispiel bei der Betrachtung einer Wahrscheinlichkeit von 90 %, dass das wahre Ergebnis $\hat{X}$ zum Bereich $\left[\overline{X} - 1{,}64\sigma / \sqrt{N},\quad \overline{X} + 1{,}64\sigma / \sqrt{N}\right]$ gehört.

Dieser Bereich verringert sich, wenn die Anzahl der Historien ansteigt und die Häufigkeit des Auftretens von *X* ansteigt.

Mit heutigen PCs sind Berechnungen ohne Probleme selbst für sicherheitsbezogene E/E/EP-Systeme in SIL 4 durchführbar.

B.5.3.3 Grundsätze der *PFD*-Berechnungen

Das Sub-Petri-Netz in Bild B.33 kann unmittelbar für die Bestimmung der PFD_{avg} der Komponente verwendet werden, da die mittlere Dauer für die Markierung der Stelle *W*, welche dem Verhältnis der Verweildauer in *W* (d. h. *W* ist durch ein Token markiert) zum Zeitraum T entspricht, in der Tat die mittlere Verfügbarkeit *A* des Bauteils ist. Es gilt $PFD_{avg} = 1 - A$.

Die Genauigkeit der Berechnung lässt sich wie oben aufgeführt abschätzen.

Die Anwendung von speziellen Sub-Petri-Netzen kann komplexere Verhaltensweisen darstellen und Bild B.34 vermittelt einen Eindruck, wie periodisch getestete Komponenten, Ausfälle infolge gemeinsamer Ursache (CCF) und Reparaturressourcen modelliert werden können.

Auf der linken Seite ist ein periodisch getestetes Bauteil mit Übergängen zu den Zuständen Betrieb (*W*), gefahrbringend unentdeckt ausgefallen (*DU*), Diagnosetest (*DUT*), gefahrbringend entdeckt ausgefallen (*DD*), bereit zur Reparatur (*RR*) und Reparatur (*R*) modelliert.

Ein Ausfall (*DU*) führt zu der Nachricht *!-Ci* (äquivalent zu *!Ci = falsch*), um mitzuteilen, dass das Bauteil ausgefallen ist. Dann wird gewartet, bis ein periodischer Test (*DUT*) beginnt. Hierbei sind das periodische Testintervall τ und der Versatz θ. Wonach der Test für die Dauer π durchgeführt und der Zustand (*DD*) erreicht wird. Ist ein Ersatzteil verfügbar (mindestens ein Token in *SP* vorhanden), wird die Komponente bereit zur Reparatur (*RR*) und die Variable *NbR* erhöht sich um 1, damit die Reparaturressourcen über die Anzahl der zu reparierenden Komponenten informiert werden. Wenn die Reparaturressourcen vor Ort sind (ein Token in *OL*), beginnt die Reparatur (*R*) und das Token wird aus *OL* herausgenommen. Ist sie ausgeführt und die Komponente wieder in einem funktionsfähigen Zustand, führt das zu der Nachricht !Ci (d. h. *!Ci = wahr*), *NbR* wird um 1 herabgesetzt und das Token wird OL wieder hinzugefügt, um weitere Reparaturen zuzulassen. Und immer so weiter.

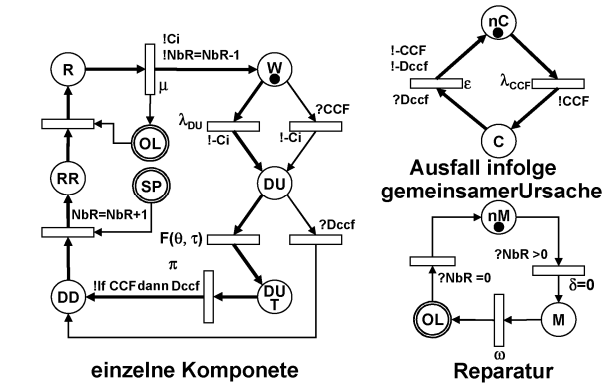


Bild B.34 – Petri-Netze zur Modellierung von Ausfällen infolge gemeinsamer Ursache und Reparaturressourcen

Die Variable *NbR* wird durch das Sub-Petri-Netz für die Reparaturen verwendet. Wird sie positiv, beginnt die Aktivierung der Ressourcen (*M*) und nach einer bestimmten Verzögerung sind diese vor Ort (*OL*) funktionsbereit. Das Token in *OL* wird für die Validierung des Beginns der Reparatur einer ausgefallenen Komponente verwendet. Somit ist nur eine Reparatur zum selben Zeitpunkt möglich. Wenn alle Reparaturen durchgeführt sind (d. h. *NbR* = 0), werden die Reparaturressourcen wieder abgezogen.

In Bild B.34 ist auch ein Ausfall infolge gemeinsamer Ursache (CCF) modelliert worden. Wenn dieser (λ_{DCC}) auftritt, wird die Nachricht *!CFF* wahr und dafür verwendet, alle betroffenen Komponenten in deren Zustände *DU* zu versetzen. Die relevante Nachricht *Ci* wird falsch und die Komponenten werden unabhängig voneinander repariert. Ist der Test einer Komponente beendet, sorgt die Zusicherung *!IF CCF dann Dccf* für die Information aller anderen Bauteile, dass ein CCF erkannt worden ist. Diese Nachricht wird verwendet, um sie unverzüglich in ihre DD-Zustände zu versetzen. Diese Nachricht wird auch verwendet, um das Petri-Netz mit dem CCF zurückzusetzen, was allerdings erst nach einer Zeitspanne (ε) geschieht, um sicherzustellen, dass alle Bauteile vor dem Rücksetzen in die *DD*-Zustände gesetzt worden sind.

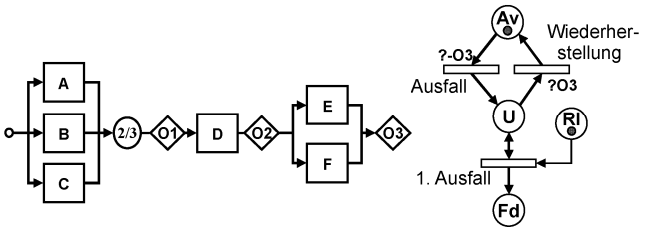


Bild B.35 – Verwendung von Zuverlässigkeitsblockdiagrammen zur Aufstellung von Petri-Netzen und Auxiliary-Petri-Netzen für die *PFD*- und *PFH*-Berechnungen

Die Sub-Petri-Netze in Bild B.34 sind zur Verwendung als Teile komplexerer Modelle vorgesehen. In Bild B.35 ist eine Möglichkeit, diese zu verwenden, dargestellt, wobei hier einfach das Zuverlässigkeitsblockdiagramm des Bildes B.16 verwendet wird, leicht verändert, indem die zwischengeschalteten Ausgänge *Oi* hinzugefügt wurden.

Die Bauteile A, B, C, D, E, F können durch eine Menge von Sub-Petri-Netzen, wie diejenigen in Bild B.34, mit zum Beispiel einem CCF für (A, B, C), einem weiteren für (E, F) und den gleichen Reparaturressourcen für alle Bauteile modelliert werden. Es bleibt nur das Problem der Verbindung der Bauteile untereinander entsprechend der Logik des Zuverlässigkeitsblockdiagramms und die Berechnung der gewünschten PFD_{avg}.

Die Verknüpfung der Bauteile ist, mit Hilfe der Nachrichten *Ci* und folgender Zusicherungen relativ einfach:

– $O_1 = C_a.C_b + C_a.C_c + C_b.C_c$

– $O_2 = O_1.C_d$

– $O_3 = O_2.(C_e + C_f)$

Wenn O_3 wahr ist, funktioniert demzufolge das gesamte sicherheitsbezogene E/E/EP-System einwandfrei und andernfalls ist es nicht verfügbar. Diese Nachricht wird in dem rechten Sub-Petri-Netz angewendet, um die verschiedenen Zustände des sicherheitsbezogenen E/E/EP-Systems zu modellieren: verfügbar (*Av*), nicht verfügbar (*U*), zuverlässig (*RI*) und nicht zuverlässig (*Fd*).

Für die Berechnung der *PFD* wird nur *Av* und *U* betrachtet: Wenn O_3 falsch wird, hat das System einen Ausfall und wird nicht verfügbar, wird O_3 wahr, ist das System wiederhergestellt und wird wieder verfügbar. Dies ist sehr einfach und die mittlere Dauer für die Markierung von *Av* ist die mittlere Verfügbarkeit des Systems und die mittlere Dauer für die Markierung von *U* ist dessen Nichtverfügbarkeit, d. h. dessen PFD_{avg}.

Deswegen führt die Monte-Carlo-Simulation das Integral der augenblicklichen Nichtverfügbarkeit automatisch aus und es ist nicht notwendig, dieses zu berechnen, es sei denn, man will die Sägezahnkurve erhalten. Dafür ist es einfacher, die mittlere Markierung von *U* an einem bestimmten Zeitpunkt als über die gesamte Zeitspanne [*0, T*] auszurechnen.

Die oben gemachten Ausführungen sind nur dazu gedacht, die breite Anwendung der Petri-Netze für die Berechnung des Sicherheits-Integritätslevels aufzuzeigen, die Modellierungsmöglichkeiten sind gewissermaßen unendlich.

B.5.3.4 Grundsätze der *PFH*-Berechnungen

Die Prinzipien der *PFH*-Berechnung sind genau dieselben wie oben ausgeführt und für DU-Ausfälle können die gleichen Submodelle verwendet werden. Bild B.36 stellt ein Sub-Petri-Netz dar, das einen DD-Ausfall modelliert, welcher repariert wird, sobald er erkannt worden ist.

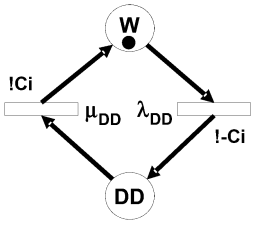


Bild B.36 – Einfaches Petri-Netz für eine einzelne Komponente mit entdeckten Ausfällen und Reparaturen

Solche Komponentenmodelle können, wie schon vorher dargestellt, in Verbindung mit einem das Gesamtsystem abbildenden Zuverlässigkeitsblockdiagramm, wie in Bild B.35 gezeigt, verwendet werden.

Bei einem sicherheitsbezogenen E/E/PE-System in der Betriebsart mit kontinuierlicher Anforderung, welches die allerletzte Barriere für die Sicherheit bildet, tritt ein Störfall ein, sobald es ausfällt, und die *PFH* muss über die Zuverlässigkeit des Systems ermittelt werden. Dies geschieht durch den unteren Teil des rechts in Bild B.35 dargestellten Sub-Petri-Netzes: die mittlere Häufigkeit des ersten Ausfalls des Systems über [*0, T*] ist seine Unzuverlässigkeit $F(T)$. Dann, wenn $F(T)$ gegenüber 1 klein ist und gemäß der *PFH*-Definition, ergibt das $PFH = F(T)/T$.

Wegen des Tokens in *RI* ist der erste Ausfall ein Übergang mit einer Aktivierung. Sofern alle Historien zu einem Ausfall führen (d. h. *T* ist lang genug), entspricht die mittlere Verweildauer eines Token in *RI* der *MTTF* des Systems. Dann ergibt $PFH \approx 1/MTTF$ eine obere Grenze der *PFH*.

Wenn das sicherheitsbezogene E/E/PE-System in der Betriebsart mit kontinuierlicher Anforderung nicht die allerletzte Barriere für die Sicherheit bildet, löst dessen Ausfall nicht unmittelbar einen Störfall aus. Nach einem Gesamtausfall wird es repariert und seine *PFH* muss über die Nichtverfügbarkeit des Systems berechnet werden. Diese erhält man unmittelbar aus der Häufigkeit *Nbf* des Übergangs Ausfall. Dies liefert die Anzahl der Systemausfälle über einen festgelegten Zeitraum und damit ergibt sich $PFH(T) = Nbf/T$.

Interessant ist die Feststellung, dass, wenn *T* lang genug ist, *MUT* über die mittlere kumulierte Dauer MCT_{Av} im Zustand *Av* und *MDT* über die mittlere kumulierte Dauer MCT_U im Zustand *U* berechnet werden können.

Die mittlere kumulierte Dauer MCT_A und die mittlere kumulierte MCT_U sind sehr einfach mit der Monte-Carlo-Simulation zu berechnen, indem die Zeiten, in denen ein Token in *Av* oder *U* vorhanden ist, kumuliert werden. Es ergibt sich $MUT = MCT_A / Nbf$ und $MUT = MCT_u / Nbf$. Dies kann für die Berechnung $PFH = 1/(MUT + MDT) = 1/MTBF = Nbf/T$ verwendet werden.

All diese Ergebnisse erhält man unmittelbar, da durch die Monte-Carlo-Simulation natürlicherweise die Mittelwerte bereitgestellt werden. Die oben gemachten Ausführungen sind nur dazu gedacht, die breite Anwendung der Petri-Netze für die Berechnung des Sicherheits-Integritätslevels aufzuzeigen, die Modellierungsmöglichkeiten sind gewissermaßen unendlich.

B.5.4 Andere Ansätze

Das Verhältnis der Größe des Modells zu der Anzahl der Komponenten des zu betrachtenden Systems ist entsprechend des verwendeten Ansatzes sehr unterschiedlich. Bei Fehlerbäumen und Petri-Netzen verhält es sich linear, bei Markov-Prozessen dagegen exponentiell. Deshalb machen Ansätze mit Fehlerbaum und Petri-Netz die Bearbeitung von größeren Systemen möglicherweise einfacher, als es mit dem Markov-Ansatz der Fall ist. Darum werden Petri-Netze mitunter dazu verwendet, große Markov-Graphen anzufertigen.

Die den oben beschriebenen graphischen Darstellungen zugrunde liegenden formalen Sprachen produzieren flache Modelle: jede Komponente wird einzeln auf derselben Ebene beschrieben. Dies hat umfangreiche Modelle zur Folge, die oft schwer zu beherrschen und zu pflegen sind. Eine Möglichkeit, dieses Problem zu bewältigen, ist, strukturierte Sprachen zu verwenden, die kompakte hierarchische Modelle liefern. Mehrere dieser formalen Sprachen sind in letzter Zeit entwickelt worden und einige Softwareanwendungen stehen zur Verfügung. Als ein Beispiel kann die Sprache „AltaRica Data Flow“ angesehen werden, die im Jahr 2 000 veröffentlichte wurde, um für die an Zuverlässigkeit Interessierten frei verfügbar zu sein, und entwickelt wurde, um die funktionalen und dysfunktionalen Eigenschaften industrieller Systeme zu modellieren (siehe Verweise in B.7).

Bild B.37 entspricht dem in Bild B.1 dargestellten Zuverlässigkeitsblockdiagramm. Dieses Modell ist hierarchisch, weil einzelne Module nur einmal modelliert wurden und dann so oft wie auf der Systemebene (d. h. im Knoten „main“) benötigt wieder verwendet werden. Damit werden sehr kompakte Modelle erreicht.

Um die Darstellung zu vereinfachen, sind nur zwei Übergänge für die Komponenten abgebildet worden: Ausfall und Reparatur (d. h. DD-Ausfälle erkannt und repariert, sobald sie auftreten).

Die logischen Operatoren (*oder*, *und*) dienen zur Beschreibung der Logik des Systems. Dies geschieht in direktem Zusammenhang mit dem Zuverlässigkeitsblockdiagramm und mit dem Ablauf „Out“ wird der Zustand des Systems modelliert: es ist betriebsbereit, wenn *Out wahr* ist, und ausgefallen, wenn *Out falsch* ist.

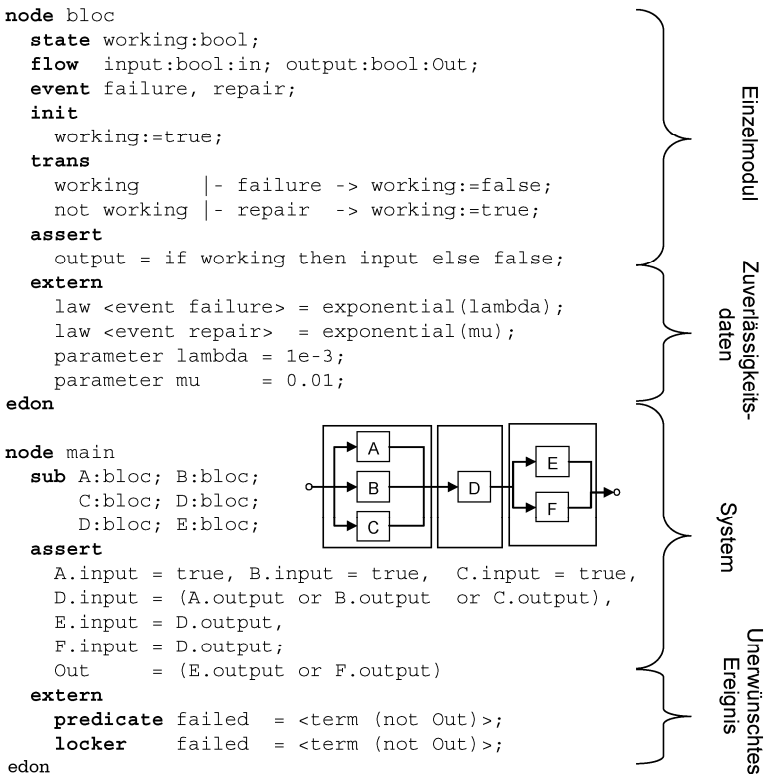


Bild B.37 – Beispiel einer funktionalen und dysfunktionalen Modellierung mit einer formalen Sprache

Damit werden gute Verhaltensmodelle für eine wirksame Monte-Carlo-Simulation erstellt und alles, was oben über das Petri-Netz geschrieben wurde, behält hier für die Berechnungen von PFD_{avg} oder *PFH* seine Gültigkeit. Deshalb wird dieser Teil hier nicht weiter fortgeführt.

Diese formale Sprache weist ähnliche mathematische Eigenschaften auf wie die Petri-Netze und demzufolge ist es möglich, ein Modell den anderen ohne Schwierigkeiten hinzuzufügen. Die Eigenschaften der den Fehlerbäumen oder Markov-Prozessen zugrundeliegenden Sprachen sind aber ebenso verallgemeinert. Deshalb ist es möglich, die Modelle in entsprechende Markov-Graphen oder Fehlerbäume zu übertragen, vorausgesetzt, dass die Beschreibung auf die Eigenschaften von Markov-Prozessen oder Fehlerbäumen eingegrenzt wurde. Zweck der Schlüsselwörter „predicate“ und „locker“ am Ende dieses Modells ist es, Anweisungen für die Generierung von Fehlerbäumen oder Markov-Modellen oder für eine direkte Monte-Carlo-Simulation zu geben.

Die Anwendung einer formalen Sprache, welche für die genaue Modellierung des Systemverhaltens entworfen ist, ermöglicht aus sowohl der funktionalen als auch der dysfunktionalen Sicht:

– Monte-Carlo-Simulationen, die unmittelbar an den Modellen durchgeführt werden können,
– Markov-Graphen zu erzeugen und analytische Berechnungen durchzuführen wie zuvor aufgezeigt (wenn die Sprache auf die Eigenschaften der Markov-Prozesse eingegrenzt worden ist),
– äquivalente Fehlerbäume zu erzeugen und analytische Berechnungen durchzuführen wie zuvor aufgezeigt (wenn die Sprache auf die Booleschen Eigenschaften eingegrenzt worden ist).

Solche funktional und dysfunktional formalen Sprachen sind Universalsprachen. Es gibt keine Schwierigkeiten, diese jeweils auf sicherheitsbezogene E/E/PE-Systeme anzuwenden. Sie bieten eine wirkungsvolle Methode der Berechnung der PFD_{avg} und *PFH* des sicherheitsbezogenen E/E/PE-Systems, wenn mehrere Schutzebenen, verschiedene Arten von Ausfallmodi, komplexe Muster für die Wiederholungsprüfung, Bauteilabhängigkeiten, Instandhaltungsressourcen usw. vorhanden sind, d. h. wenn die anderen Methoden an ihre Grenzen gestoßen sind.

B.6 Behandlung von Unsicherheiten

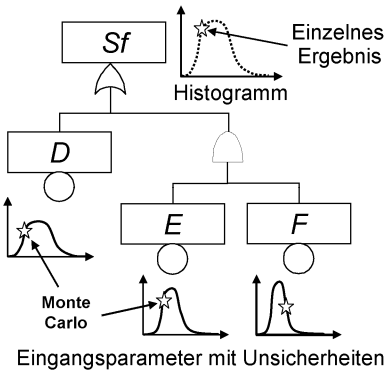


Bild B.38 – Prinzip der Fortpflanzung von Unsicherheiten

Eines der größten Probleme bei der Durchführung der Wahrscheinlichkeitsberechnungen ist mit den Unsicherheiten bei den Zuverlässigkeitsparametern verbunden. Darum ist es hilfreich, bei der Berechnung der *PFD* oder der *PFH* zu bewerten, welcher entsprechende Einfluss auf die Unsicherheit des Ergebnisses ausgeübt wird.

Bei der Behandlung dieses Themas ist äußerste Sorgfalt notwendig und die Anwendung der Monte-Carlo-Simulation bietet für diesen Zweck eine wirkungsvolle Methode, so wie in Bild B.38 veranschaulicht.

In diesem Bild sind die Eingangszuverlässigkeitsparameter (z. B. die Rate gefahrbringender unentdeckter Ausfälle) nicht mehr sicher und werden durch zufällige Variablen ersetzt. Die Wahrscheinlichkeitsdichte solcher zufälligen Variablen ist entsprechend des Grades der Unsicherheit mehr oder weniger hoch oder niedrig: die Wahrscheinlichkeitsdichte von F ist höher als die von E oder D. Dies bedeutet, dass zum Beispiel die Unsicherheit für F geringer ist als für E oder D.

Das Prinzip der Berechnungen ist wie folgt:

1) Erzeugung einer Gruppe von Eingangsparametern mit Hilfe von Zufallszahlen entsprechend der Wahrscheinlichkeitsverteilung dieser Parameter (ähnlich zu dem, was in B.3.2 erläutert worden ist);
2) Durchführung einer Berechnung unter Verwendung der oben erzeugten Gruppe von Eingangsparametern;
3) Aufzeichnung des Ausgangsergebnisses (dies stellt einen in Schritt 4 verwendeten Wert dar);
4) Wiederholung der Schritte 1) bis 3), bis eine ausreichende Anzahl von Werten (zum Beispiel 100 oder 1 000) erreicht wurde, um ein Histogramm erstellen zu können (gepunktete Linie in Bild B.38);
5) statistische Analyse des Histogramms, um den Mittelwert und die Standardabweichung des Ausgangsergebnisses zu ermitteln.

Der Mittelwert dieses Histogramms ist entsprechend der Art der durchgeführten Berechnung die PFD_{avg} oder die *PFH* und die Standardabweichung erfasst die Unsicherheit dieses Ergebnisses. Je kleiner die Standardabweichung ist, desto genauer ist die Berechnung von PFD_{avg} oder *PFH*.

Das hier an einem Fehlerbaum dargestellte Prinzip ist sehr allgemein und kann für jede der in diesem Anhang beschriebenen Berechnungsmethoden angewendet werden: vereinfachte Formeln, Markov-Prozesse und auch Petri-Netze oder Ansätze mit formalen Sprachen. Wenn die Berechnung bereits mit der Monte-Carlo-Simulation durchgeführt worden ist, muss eine Monte-Carlo-Simulation in zwei Stufen verwendet werden.

Die Wahrscheinlichkeitsverteilung für einen gegebenen Eingangszuverlässigkeitsparameter muss entsprechend der gesammelten Erkenntnisse über diesen ausgewählt werden. Diese kann sein:

– eine einheitliche Verteilung zwischen den unteren und oberen Grenzen;
– eine Dreiecksverteilung mit einem am meisten wahrscheinlichen Wert;

– eine Verteilung nach dem Gesetz logarithmischer Normalverteilung mit einem gegebenen Abweichungsfaktor;
– eine Verteilung nach dem Chi-Quadrat-Gesetz usw.

Wenn nicht sehr viele Erfahrungswerte aus dem Feld zur Verfügung stehen, können die Erstgenannten durch eine ingenieurtechnische Beurteilung beurteilt werden. Stehen mehr Rückmeldungen aus dem Feld zur Verfügung, können die Letztgenannten angewendet werden, weil die Erfahrungswerte aus dem Feld durchschnittliche Parameterwerte liefern sowie Konfidenzbereiche für diese Mittelwerte.

Wenn zum Beispiel *n* Ausfälle über eine kumulierte Beobachtungszeit *T* festgestellt worden sind, ergibt das:

– $\hat{\lambda} = n/T$ für die Maximum-Likelihood-Schätzung der Ausfallrate

– $\lambda_{Inf,\alpha} = \frac{1}{2T}\chi^2_{(1-\alpha),2n}$ als untere Grenze, die mit einer Wahrscheinlichkeit von α % niedriger ist als $\lambda_{\text{Inf},\alpha}$

– $\lambda_{Sup,\alpha} = \frac{1}{2T}\chi^2_{\alpha,2(n+1)}$ als obere Grenze, die mit einer Wahrscheinlichkeit von α % höher ist als $\lambda_{\text{Sup},\alpha}$

Wenn α = 5 % ist, liegt die Wahrscheinlichkeit von 90 zu 100 vor, dass der wahre Wert von λ zum Intervall $[\lambda_{\text{Inf},\alpha}, \lambda_{\text{Sup},\alpha}]$ gehört. Je kleiner das Intervall ist, desto genauer ist der Wert von λ. Üblicherweise werden diese Informationen von guten Zuverlässigkeitsdatenbanken zur Verfügung gestellt. Analytiker sollten Zuverlässigkeitsdaten, die ohne Konfidenzbereiche (oder Informationen diese zu berechnen) bereitgestellt werden, sehr vorsichtig behandeln.

Für die Modellierung der Ausfallrate λ eines festgelegten Ausfallmodus und ihrer Unsicherheiten können $\hat{\lambda}$, $\lambda_{Inf,\alpha}$ und $\lambda_{Sup,\alpha}$ zur Aufstellung einer entsprechenden Verteilung verwendet werden. Dies ist für die χ^2-Verteilung eindeutig, die logarithmische Normalverteilung ist aber für diesen Zweck ebenfalls sehr gut geeignet. Eine derartige logarithmische Normalverteilung ist durch ihren Mittelwert $\bar{\lambda}$ definiert oder ihren Halbwert $\lambda_{50\,\%}$ und ihren sogenannten Abweichungsfaktor.

Dieses Gesetz hat eine sehr interessante Eigenschaft: $\lambda_{\text{Inf},\alpha} = \lambda_{50\,\%}/ef_\alpha$ und $\lambda_{\text{Sup},\alpha} = \lambda_{50\,\%} \cdot ef_\alpha$.

Dann wird es mit nur zwei Parametern definiert: $\hat{\lambda}$ und $ef_\alpha \approx \sqrt{\frac{\lambda_{Sup,\alpha}}{\lambda_{Inf,\alpha}}}$

Wenn ef_α = 1 ist, gibt es keine Unsicherheiten, bei ef_α = 3,3 ist ein Faktor von ungefähr 10 zwischen den unteren und den oberen Konfidenzgrenzen vorhanden, usw.

Diese Gesetze können wiederum in der Monte-Carlo-Simulation angewendet werden, um sowohl die Einflüsse der Mittelwerte als auch der Unsicherheiten von PFD_{avg} oder *PFH* zu berücksichtigen. Deshalb ist es immer möglich, die Unsicherheiten innerhalb von Wahrscheinlichkeitsberechnungen zu bewältigen. Solche Berechnungen werden von einigen Softwareanwendungen direkt durchgeführt.

Bei der Analyse von redundanten Systemen sollte nicht nur die Unsicherheit der Ausfallrate der einfachen Komponenten betrachtet werden, sondern auch die Genauigkeit der CCF-Ausfallrate. Selbst wenn es nützliche Rückmeldungen für jedes dieser Elemente aus dem Feld gibt, gute CCF-Erfahrungswerte aus dem Feld sind selten und dies ist folglich die größte Unsicherheit.

B.7 Verweise

[4] bis [9] und [22] bis [24] in den Literaturhinweisen nennen weitere Einzelheiten zur Bewertung der Wahrscheinlichkeit von Ausfällen.

Anhang C
(informativ)

Ausgearbeitetes Beispiel für die Berechnung des Diagnosedeckungsgrads und des Anteils sicherer Ausfälle

Eine Methode für die Berechnung des Diagnosedeckungsgrads und des Anteils sicherer Ausfälle ist in IEC 61508-2, Anhang C, beschrieben. Der vorliegende Anhang beschreibt die Verwendung dieser Methode, um den Diagnosedeckungsgrad zu berechnen. Es wird vorausgesetzt, dass alle in der IEC 61508-2 festgelegten Informationen zur Verfügung stehen und erforderlichenfalls zur Gewinnung der Werte in Tabelle C.1 verwendet worden sind. Die Tabelle C.2 zeigt Grenzen des Diagnosedeckungsgrads, wie er für bestimmte sicherheitsbezogene E/E/PE-Elemente beansprucht werden kann. Die Werte in Tabelle C. 2 basieren auf einer ingenieurmäßigen Abschätzung.

Zum Verständnis aller Werte in Tabelle C.1 wäre ein genaues Hardwareschaltbild erforderlich, aus dem die Auswirkungen aller Ausfallarten bestimmt werden können. Diese Werte sind nur Beispiele, so kann für einige Bauteile in Tabelle C.1 kein Diagnosedeckungsgrad angenommen werden, weil es praktisch unmöglich ist, alle Ausfallarten für alle Bauteile zu erkennen.

Die Tabelle C.1 ist wie folgt hergeleitet worden:

a) Eine FMEA (Ausfallarten- und Auswirkungsanalyse) ist durchgeführt worden, um die Auswirkung jeder Ausfallart für jedes Bauteil auf das Verhalten des Systems ohne Diagnosetest zu bestimmen. Für jede Ausfallart und jedes Bauteil werden die Anteile der sicheren (S) und gefahrbringenden (D) Ausfälle an der Gesamtausfallrate gezeigt. Die Aufteilung in sichere und gefahrbringende Ausfälle kann für einfache Bauteile deterministisch und andernfalls auf der Basis ingenieurmäßiger Beurteilung erfolgen. Für komplexe Bauteile, bei denen eine genaue Analyse jeder Ausfallart nicht möglich ist, wird allgemein eine Aufteilung der Ausfälle in 50 % sicher und 50 % gefahrbringend angenommen. Für diese Tabelle sind die Ausfallverhalten nach Verweis a) verwendet worden, obwohl andere Aufteilungen zwischen den Ausfallverhalten möglich und zu bevorzugen sein können.

b) Der Diagnosedeckungsgrad für jeden spezifischen Diagnosetest jedes Bauteils ist in der mit „DC_{COMP}“ bezeichneten Spalte angegeben. Spezifische Diagnosedeckungsgrade werden für die Aufdeckung von sowohl sicheren als auch gefahrbringenden Ausfällen angegeben. Die Anwendung der Tabelle C.2 hat den Diagnosedeckungsgrad für U 16, ein komplexes Typ B-Bauteil, auf 90 % begrenzt, während bei Fehlern aufgrund einer Stromkreisunterbrechung oder aufgrund eines Kurzschlusses bei einfachen Bauteilen wie Widerständen, Kondensatoren oder Transistoren von einem Diagnosedeckungsgrad von 100 % ausgegangen werden kann.

c) Die Spalten (1) und (2) geben für jedes Bauteil die Raten für sichere und gefahrbringende Ausfälle (λ_S bzw. $\lambda_{DD} + \lambda_{DU}$) ohne Diagnosetest an.

d) Man kann einen erkannten gefahrbringenden Ausfall als einen tatsächlich sicheren Ausfall betrachten und daher zwischen tatsächlich sicheren Ausfällen (d. h. entweder erkannten sicheren, unerkannten sicheren oder erkannten gefahrbringenden Ausfällen) und unerkannten gefahrbringenden Ausfällen unterscheiden. Die Rate der tatsächlich sicheren Ausfälle wird durch die Multiplikation der Rate der gefahrbringenden Ausfälle mit dem spezifischen Diagnosedeckungsgrad für gefahrbringende Ausfälle und der Addition des Ergebnisses zur Rate der sicheren Ausfälle gebildet (siehe Spalte (3)). Ebenso wird die Rate der unerkannten gefahrbringenden Ausfälle durch die Subtraktion des spezifischen Diagnosedeckungsgrads für gefahrbringende Ausfälle von eins und Multiplikation des Ergebnisses mit der Rate der gefahrbringenden Ausfälle (siehe Spalte (4)) gebildet.

e) Spalte (5) gibt die Rate der erkannten sicheren und Spalte (6) die der erkannten gefahrbringenden Ausfälle an. Diese werden gebildet durch die Multiplikation des spezifischen Diagnosedeckungsgrades mit der Rate der sicheren bzw. gefahrbringenden Ausfälle.

f) Die Tabelle liefert die folgenden Ergebnisse:

- Gesamtrate sicherer Ausfälle $\sum \lambda_S + \sum \lambda_{Dd} = 9{,}9 \times 10^{-7}$
  (einschließlich erkannter gefahrbringender Ausfälle)
- Gesamtrate der unerkannten gefahrbringenden Ausfälle $\sum \lambda_{Du} = 5{,}1 \times 10^{-8}$
- Gesamtausfallrate $\sum \lambda_S + \sum \lambda_{Dd} + \sum \lambda_{Du} = 1{,}0 \times 10^{-6}$
- Gesamtrate unerkannter sicherer Ausfälle $\sum \lambda_{Su} = 2{,}7 \times 10^{-8}$
- Diagnosedeckungsgrad für sichere Ausfälle $\frac{\sum \lambda_{Sd}}{\sum \lambda_S} = \frac{3{,}38}{3{,}65} = 93\ \%$
- Diagnosedeckungsgrad für gefahrbringende Ausfälle

  $\frac{\sum \lambda_{Dd}}{\sum \lambda_{Dd} + \sum \lambda_{Du}} = \frac{6{,}21}{6{,}72} = 92\ \%$ (üblicherweise einfach „Diagnosedeckungsgrad“ genannt)
- Anteil sicherer Ausfälle $\frac{\sum \lambda_S + \sum \lambda_{Dd}}{\sum \lambda_S + \sum \lambda_{Dd} + \sum \lambda_{Du}} = \frac{986}{365 + 672} = 95\ \%$

g) Die Aufteilung der Ausfallrate ohne Diagnosetests beträgt 35 % sichere Ausfälle und 65 % gefahrbringende Ausfälle.

Tabelle C.1 – Beispielberechnung für den Diagnosedeckungsgrad und den Anteil sicherer Ausfälle

Teil	Nr.	Typ	Aufteilung in sichere und gefahrbringende Ausfälle für jede Ausfallart								Aufteilung in sichere und gefahrbringende Ausfälle für den Diagnosedeckungsgrad und berechnete Ausfallraten ($\times 10^{-9}$)							
			OC		SC		Drift		Funktion		DC_{comp}		(1)	(2)	(3)	(4)	(5)	(6)
			S	D	S	D	S	D	S	D	S	D	λ_S	$\lambda_{DD}+\lambda_{Du}$	$\lambda_S+\lambda_{DD}$	λ_{DU}	λ_{SD}	λ_{DD}
Leiter-platte	1	Leiterplatte	0,5	0,5	0,5	0,5	0	0	0	0	0,99	0,99	11,0	11,0	21,9	0,1	10,9	10,9
CN1	1	Con96pin	0,5	0,5	0,5	0,5					0,99	0,99	11,5	11,5	22,9	0,1	11,4	11,4
C1	1	100nF	1	0	1	0	0	0	0	0	1	0	3,2	0,0	3,2	0,0	3,2	0,0
C2	1	10µF	0	0	1	0	0	0	0	0	1	0	0,8	0,0	0,8	0,0	0,8	0,0
R4	1	1M	0,5	0,5	0,5	0,5					1	1	1,7	1,7	3,3	0,0	1,7	1,7
R6	1	100k									0	0	0,0	0,0	0,0	0,0	0,0	0,0
OSC1	1	OSC24Mhz	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	1	1	16,0	16,0	32,0	0,0	16,0	16,0
U8	1	74HCT85	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,99	0,99	22,8	22,8	45,4	0,2	22,6	22,6
U16	1	MC68000-12	0	1	0	1	0,5	0,5	0,5	0,5	0,90	0,90	260,4	483,6	695,6	48,4	234,4	435,2
U26	1	74HCT74	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,99	0,99	22,8	22,8	45,4	0,2	22,6	22,6
U27	1	74F74	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,99	0,99	14,4	14,4	28,7	0,1	14,3	14,3
U28	1	PAL16L8A	0	1	0	1	0	1	0	1	0,98	0,98	0,0	88,0	86,2	1,8	0,0	86,2
T1	1	BC817	0	0	0	0,67	0	0,5	0	0	1	1	0,0	0,2	0,4	0,0	0,0	0,2
Gesamt													365	672	986	50,9	338	621

ANMERKUNG Keiner der Ausfallmodi von R6 wird erkannt, aber ein Ausfall beeinflusst weder die Sicherheit noch die Verfügbarkeit.

Schlüssel

S	Sicherer Ausfall
D	Gefahrbringender Ausfall
OC	Unterbrechung (en: open circuit)
SC	Kurzschluss (en: short circuit)
Drift	Änderung von Werten
Funktion	Funktionsausfälle
DC_{comp}	Spezifischer Diagnosedeckungsgrad für das Bauteil

Siehe auch Tabelle B.1, obwohl die Ausfallraten in Tabelle C.1 eher für die betreffenden einzelnen Bauteile gelten als für jedes Bauteil in einem Kanal.

Tabelle C.2 – Diagnosedeckungsgrad und Wirksamkeit für verschiedene Teilsysteme

Bauteil	Diagnose-deckungsgrad niedrig	Diagnose-deckungsgrad mittel	Diagnose-deckungsgrad hoch
CPU (siehe Anmerkung 3)	insgesamt kleiner 70 %	insgesamt kleiner 90 %	
Register, internes RAM	50 % bis 70 % [NA4)]	85 % bis 90 %	99 % bis 99,99 %
Codierung und Ausführung einschließlich Flag-Register (siehe Anmerkung 3)	50 % bis 60 %	75 % bis 95 %	–
Adressberechnung (siehe Anmerkung 3)	50 % bis 70 %	85 % bis 98 %	–
Programm-Counter, Stack-Pointer	40 % bis 60 %	60 % bis 90 %	85 % bis 98 %
Bus			
Speichermanagementeinheit (MMU)	50 %	70 %	90 % bis 99 %
Bus-Arbitration	50 %	70 %	90 % bis 99 %
Interrupt-Behandlung	40 % bis 60 %	60 % bis 90 %	85 % bis 98 %
Takt (Quarz) (siehe Anmerkung 4)	50 %	–	95 % bis 99 %
Programmlaufüberwachung			
zeitlich (siehe Anmerkung 3)	40 % bis 60 %	60 % bis 80 %	–
logisch (siehe Anmerkung 3)	40 % bis 60 %	60 % bis 90 %	–
zeitlich und logisch (siehe Anmerkung 5)	–	65 % bis 90 %	90 % bis 98 %
Unveränderlicher Speicher	50 % bis 70 %	99 %	99,99 %
Veränderlicher Speicher	50 % bis 70 %	85 % bis 90 %	99 % bis 99,99 %
Diskrete Hardware			
digitale E/A	70 %	90 %	99 %
analoge E/A	50 % bis 60 %	70 % bis 85 %	99 %
Stromversorgung	50 % bis 60 %	70 % bis 85 %	99 %
Kommunikation und Massenspeicher	90 %	99,9 %	99,99 %
Elektromechanische Bauteile	90 %	99 %	99,9 %
Sensoren	50 % bis 70 %	70 % bis 85 %	99 %
Stellglieder	50 % bis 70 %	70 % bis 85 %	99 %

ANMERKUNG 1 Diese Tabelle sollte in Zusammenhang mit IEC 61508-2, Tabelle A.1 gelesen werden, diese liefert die zu berücksichtigenden Ausfallarten.

ANMERKUNG 2 Wenn ein Bereich für den Diagnosedeckungsgrad angegeben ist, können die oberen Grenzen nur bei eng toleranten Überwachungseinrichtungen oder bei Prüfmaßnahmen, die die zu prüfende Funktion hochdynamisch belasten, angesetzt werden.

ANMERKUNG 3 Für Verfahren, für die keine Zahl zum hohen Diagnosedeckungsgrad angegeben ist, sind gegenwärtig keine Maßnahmen und Verfahren hoher Wirksamkeit bekannt.

ANMERKUNG 4 Gegenwärtig sind keine Maßnahmen und Verfahren mittlerer Wirksamkeit für Quarzoszillatoren bekannt.

ANMERKUNG 5 Der kleinste Diagnosedeckungsgrad für die Kombination von zeitlicher und logischer Programmlaufüberwachung ist mittel.

Siehe Verweise [10] bis [12] in den Literaturhinweisen.

[NA4)] Nationale Fußnote: In der IEC 61508-6:2010 sind in diesem Tabellenelement einige Eintragungen doppelt gedruckt.

Anhang D
(informativ)

Eine Methode zur Quantifizierung der Auswirkungen von hardwarebedingten Ausfällen infolge gemeinsamer Ursache in E/E/PE-Systemen

D.1 Allgemeines

D.1.1 Einleitung

Diese Norm beinhaltet eine Anzahl von Maßnahmen zur Behandlung systematischer Ausfälle. Doch so gut wie diese Maßnahmen auch angewendet werden, es wird immer eine Restwahrscheinlichkeit für systematische Ausfälle geben. Obwohl dies keinen signifikanten Einfluss auf die Zuverlässigkeitsberechnungen für einkanalige Systeme hat, so ist doch die Möglichkeit für Ausfälle, die mehr als einen Kanal in einem mehrkanaligen System (oder mehrere Bauteile in einem redundanten Sicherheitssystem) beeinträchtigen, d. h. Ausfälle infolge gemeinsamer Ursache, gegeben, und sie wird bei den Zuverlässigkeitsberechnungen für mehrkanalige oder redundante Systeme zu wesentlichen Abweichungen führen.

Dieser informative Anhang beschreibt zwei Methoden, die es erlauben, Ausfälle infolge gemeinsamer Ursache in der Sicherheitsbeurteilung eines mehrkanaligen programmierbaren elektronischen Systems oder eines redundanten E/E/PE-Systems zu berücksichtigen. Die Anwendung dieser Methoden wird eine genauere Abschätzung der Integrität eines solchen Systems liefern, als wenn die Möglichkeit von Ausfällen infolge gemeinsamer Ursache ignoriert wird.

Die erste Methode wird verwendet, um einen Wert für den β-Faktor zu berechnen, der oft zur Modellierung von Ausfällen infolge gemeinsamer Ursache verwendet wird. Er kann verwendet werden, um die Rate von Ausfällen von zwei oder mehreren parallel arbeitenden Systemen infolge gemeinsamer Ursache mit Hilfe der Rate zufälliger Hardwareausfälle eines dieser Systeme zu bestimmen (siehe D.5). Es ist allgemein anerkannt, dass die gesammelten Zahlen zu zufälligen Hardwareausfällen eine Anzahl von Ausfällen einschließen, die durch systematische Ausfälle bedingt sind.

Falls, z. B. aufgrund der Verfügbarkeit von Daten über Ausfälle infolge gemeinsamer Ursache, ein genauerer β-Faktor dargestellt werden kann oder wenn die Anzahl der betroffenen Elemente größer als vier ist, können andere Methoden bevorzugt werden. Die zweite Methode, d. h. das Binomial-Ausfallraten-Modell (auch Schockmodell genannt), kann angewendet werden.

D.1.2 Kurzübersicht

Für die Ausfälle eines Systems werden zwei Ursachen angenommen:

– zufällige Hardwareausfälle; und

– systematische Ausfälle.

Es wird angenommen, dass die erstgenannten Ausfälle zeitlich zufällig für jedes Bauteil auftreten und zu einem Ausfall eines Kanals innerhalb des Systems führen, zu dem das Bauteil gehört, während die zuletzt genannten Ausfälle sofort und deterministisch auftreten, wenn für das System die Situation eintritt, für die der zugrundeliegende systematische Fehler vorhanden ist.

Es gibt eine endliche Wahrscheinlichkeit dafür, dass unabhängige zufällige Hardwareausfälle in allen Kanälen eines mehrkanaligen Systems so auftreten, dass alle Kanäle gleichzeitig ausfallen. Da angenommen wird, dass zufällige Hardwareausfälle zufällig über die Zeit auftreten, ist die Wahrscheinlichkeit solcher Ausfälle, die parallele Kanäle gleichzeitig beeinträchtigen, niedrig im Vergleich zur Wahrscheinlichkeit des Ausfalls eines einzelnen Kanals. Diese Wahrscheinlichkeit kann unter Verwendung von bewährten Verfahren be-

rechnet werden, das Ergebnis kann aber sehr optimistisch ausfallen, wenn die Ausfälle nicht vollständig unabhängig voneinander sind.

Abhängige Ausfälle werden für gewöhnlich aufgeteilt in (siehe Verweis [18] in den Literaturhinweisen):

– Ausfall infolge gemeinsamer Ursache (CCF), der Mehrfachausfälle aufgrund einer einzelnen Ursache hervorruft. Die Mehrfachausfälle können gleichzeitig oder über einen Zeitraum verteilt auftreten;
– gleichartige Ausfälle (CMF), die einen Sonderfall von CCF darstellen, bei dem mehrere Bestandteile auf dieselbe Art ausfallen;
– Ausfall-Kaskaden, bei denen sich die Ausfälle fortpflanzen.

Der Begriff CCF wird oft verwendet, um damit alle Arten von abhängigen Ausfällen zu beschreiben, wie in diesem Anhang. Hier wird aber auch unterschieden zwischen:

– abhängigen Ausfällen aus eindeutig deterministischen Ursachen;
– einem verbleibenden Anteil von auftretenden möglichen Mehrfachausfällen, die nicht ausdrücklich bei den Analysen berücksichtigt wurden, wegen zu großer Ungenauigkeit, keinen eindeutigen deterministischen Ursachen oder es unmöglich ist, Zuverlässigkeitswerte zu erfassen.

Erstere sollten auf konventionelle Weise analysiert, modelliert und quantifiziert werden und nur letztere sollten so behandelt werden, wie es in diesem informativen Anhang D beschrieben wird. Dennoch werden systematische Ausfälle – welche eindeutig abhängige Ausfälle sind und während der Sicherheitsanalyse nicht erkannt wurden (sonst wären sie entfernt worden) – in dieser Norm speziell behandelt und dieser Anhang gilt vor allem für zufällige hardwarebedingte abhängige Ausfälle.

Demnach können Ausfälle infolge einer einzelnen gemeinsamen Ursache mehr als einen Kanal oder mehr als ein Bauteil beeinträchtigen. Sie können aus einem systematischen Fehler (z. B. einem Irrtum im Entwurf oder in der Spezifikation) oder einer externen Belastung, die zu einem frühzeitigen, zufälligen Hardwareausfall führt (z. B. einer Übertemperatur, die aus einem zufälligen Hardwareausfall eines gemeinsamen Lüfters herrührt, dies beschleunigt die Alterung der Bauteile oder bringt sie außerhalb ihrer spezifizierten Betriebsumgebung), oder aus einer Kombination von beiden herrühren. Da Ausfälle infolge gemeinsamer Ursache wahrscheinlich mehr als einen Kanal in einem mehrkanaligen System beeinträchtigen, ist anzunehmen, dass deren Wahrscheinlichkeit der wesentliche Faktor in der Bestimmung der Gesamtausfallwahrscheinlichkeit eines mehrkanaligen Systems ist, und wenn dies nicht berücksichtigt wird, ist eine realistische Einschätzung des Sicherheits-Integritätslevel des zusammengesetzten Systems kaum zu erreichen.

D.1.3 Schutz gegen Ausfälle infolge gemeinsamer Ursache

Obwohl Ausfälle infolge gemeinsamer Ursache auf eine einzige Ursache zurückzuführen sind, zeigen sie sich nicht unbedingt in allen Kanälen gleichzeitig. Fällt z. B. ein Lüfter aus, können alle Kanäle eines mehrkanaligen E/E/PE-Systems ausfallen, was als ein Ausfall infolge gemeinsamer Ursache aufzufassen ist. Es ist jedoch unwahrscheinlich, dass sich die Temperatur aller Kanäle gleichmäßig erhöht oder diese die gleiche kritische Temperatur aufweisen. Deshalb werden die Ausfälle zu unterschiedlichen Zeiten und in den unterschiedlichen Kanälen auftreten.

Die Architektur programmierbarer Systeme erlaubt ihnen das Ausführen interner Diagnosetestfunktionen während des Betriebes. Diese können auf verschiedene Weise eingesetzt werden:

– Ein einkanaliges programmierbares elektronisches System kann seine internen Operationen zusammen mit der Arbeitsweise der Ein- und Ausgabeeinrichtungen ständig testen. Eine Testaufdeckung im Bereich von 99 % ist erreichbar (siehe [13] in den Literaturhinweisen), wenn dies von Anfang an beim Entwurf berücksichtigt wird. Wenn 99 % der internen Fehler aufgedeckt werden, bevor sie zu einem Ausfall führen, wird die Wahrscheinlichkeit von Fehlern eines einzelnen Kanals, die schließlich zu Ausfällen infolge gemeinsamer Ursache beitragen können, deutlich vermindert.
– Zusätzlich zu internen Tests kann jeder Kanal in einem programmierbaren elektronischen System die Ausgaben anderer Kanäle in einem mehrkanaligen programmierbaren elektronischen System überwachen (oder jedes programmierbare elektronische Gerät kann ein anderes in einem solchen System überwachen). Falls ein Ausfall in einem Kanal auftritt, kann dieser deshalb erkannt und eine sichere Abschaltung durch einen oder mehrere der übrigen Kanäle, die nicht ausgefallen sind und die kreuzweise Überwachungstests durchführen, eingeleitet werden (es sollte zur Kenntnis genommen werden, dass

eine kreuzweise Überwachung nur dann wirkungsvoll ist, wenn sich der Zustand des Steuerungssystems ständig ändert, wie es z. B. bei der Verriegelung einer häufig verwendeten Schutzeinrichtung einer Maschine mit sich wiederholenden Arbeitsvorgängen der Fall ist, oder wenn kurzzeitige Änderungen eingebracht werden können, ohne die zu überwachende Funktion zu beeinträchtigen). Diese kreuzweise Überwachung kann mit einer hohen Wiederholrate ausgeführt werden, so dass sie wahrscheinlich den Ausfall des ersten Kanals erkennt und damit in der Lage ist, das System in einen sicheren Zustand zu versetzen, bevor ein zweiter Kanal betroffen wird und bevor es zu einem nicht gleichzeitigen Ausfall infolge gemeinsamer Ursache des Systems kommt.

Im Beispiel des Lüfters wird die Rate für den Temperaturanstieg und die Anfälligkeit jedes Kanals geringfügig anders sein, was möglicherweise zu einem Ausfall des zweiten Kanals einige zehn Minuten nach dem ersten führt. Dies ermöglicht dem Diagnosetest die Einleitung einer sicheren Abschaltung, bevor der zweite Kanal ebenfalls dem Fehler infolge gemeinsamer Ursache unterliegt.

Daraus kann geschlossen werden:

– Programmierbare elektronische Systeme haben die Möglichkeit, Schutz gegen Ausfälle infolge gemeinsamer Ursache zu bieten und sind deshalb im Vergleich zu anderen Technologien hiergegen weniger anfällig.

– Für programmierbare elektronische Systeme darf ein anderer β-Faktor als bei anderen Technologien angewendet werden. Daher treffen Schätzungen des β-Faktors aufgrund von früheren Daten wahrscheinlich nicht zu (keines der bestehenden untersuchten Modelle zur Schätzung der Wahrscheinlichkeit eines Ausfalls infolge gemeinsamer Ursache berücksichtigt die Auswirkung der automatischen kreuzweisen Überwachung).

– Da Ausfälle infolge gemeinsamer Ursache, die über die Zeit verteilt sind, vom Diagnosetest aufgedeckt werden können, bevor sie alle Kanäle beeinträchtigen, werden solche Ausfälle möglicherweise nicht als Ausfälle infolge gemeinsamer Ursache erkannt oder gemeldet.

Es gibt drei Wege, die beschritten werden können, um die Wahrscheinlichkeit von möglicherweise gefährlichen Ausfällen infolge gemeinsamer Ursache zu vermindern.

a) Verringerung der Anzahl zufälliger Hardware- und systematischer Ausfälle insgesamt. (Dies reduziert die Flächen der Ellipsen in Bild D.1 und führt zu einer Verringerung des Überlappungsbereichs.)

b) Vergrößerung der Unabhängigkeit (Trennung und Diversität) der Kanäle (dies verringert die Überlappung zwischen den Ellipsen in Bild D.1 bei Aufrechterhaltung ihrer Größe).

c) Aufdeckung nicht gleichzeitiger Ausfälle infolge gemeinsamer Ursache, solange nur einer und bevor ein zweiter Kanal betroffen ist, d. h. Anwendung von Diagnosetests oder versetzten Wiederholungsprüfungen.

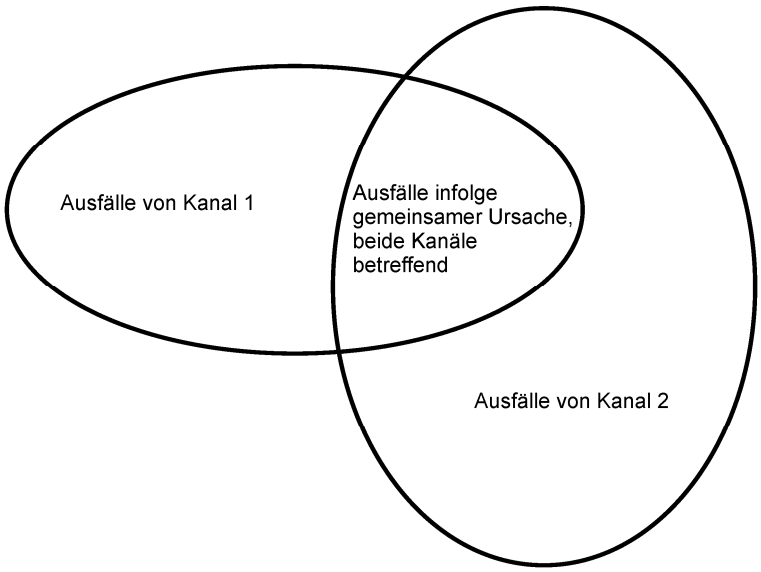


Bild D.1 – Beziehung zwischen Ausfällen infolge gemeinsamer Ursache und Ausfällen einzelner Kanäle

Bei Systemen mit mehr als zwei Kanälen kann der Ausfall infolge gemeinsamer Ursache abhängig von deren Ursprung alle oder nur mehrere Kanäle betreffen. So ist der erste Ansatz in diesem Anhang, den β-Wert bezogen auf das Doppelsystem 1oo2, zu berechnen und dann abhängig von der Gesamtanzahl der Kanäle und den Anforderungen an den Vergleich einen Multiplikationsfaktor für den erhaltenen β-Wert zu verwenden (siehe Tabelle 5).

D.1.4 In der Normenreihe IEC 61508 gewählter Ansatz

Die Normenreihe IEC 61508 beruht auf diesen drei Wegen und erfordert eine dreifache Vorgehensweise.

a) Anwendung der in der IEC 61508-2/3 festgelegten Verfahren, um die Gesamtwahrscheinlichkeit systematischer Ausfälle auf ein Niveau entsprechend der Wahrscheinlichkeit zufälliger Hardwareausfälle zu verringern.

b) Quantifizierung der quantifizierbaren Faktoren, d. h. Berücksichtigung der Wahrscheinlichkeit von zufälligen Hardwareausfällen nach IEC 61508-2.

c) Herleitung eines Faktors, der die Wahrscheinlichkeit eines Ausfalls der Hardware infolge gemeinsamer Ursache zur Wahrscheinlichkeit eines zufälligen Ausfalls der Hardware in Beziehung setzt. Dieser Faktor wird mit Mitteln, die zum gegenwärtigen Zeitpunkt als die geeignetsten angesehen werden, hergeleitet. Die in diesem Anhang beschriebene Methode bezieht sich auf die Herleitung dieses Faktors.

Die meisten Methoden zur Abschätzung der Wahrscheinlichkeiten von Ausfällen infolge gemeinsamer Ursache versuchen, Voraussagen von zufälligen Hardwareausfallwahrscheinlichkeiten abzuleiten. Eine Rechtfertigung für einen Zusammenhang zwischen diesen beiden Wahrscheinlichkeiten scheint kaum gegeben zu sein, trotzdem ist er in der Praxis gefunden worden und ergibt sich wahrscheinlich aus nachgeordneten Auswirkungen. Je höher z. B. die Wahrscheinlichkeit eines zufälligen Hardwareausfalls eines Systems ist:

– desto höher ist der vom System benötigte Instandhaltungsaufwand. Die Wahrscheinlichkeit eines während der Instandhaltung eingebrachten systematischen Fehlers hängt von der Anzahl der ausgeführten Instandhaltungstätigkeiten ab, und diese wird ebenfalls die Anzahl der Fälle menschlichen Versagens beeinflussen, die zum Ausfall infolge gemeinsamer Ursache führen. Dies führt zu einer Beziehung zwischen der Wahrscheinlichkeit zufälliger Hardwareausfälle und der Wahrscheinlichkeit von Ausfällen infolge gemeinsamer Ursache. Beispielsweise
  – ist nach einem zufälligen Hardwareausfall jedes Mal eine Reparatur, gefolgt von Tests und möglicherweise neuen Justierungen, erforderlich;
  – sind für einen gegebenen Sicherheits-Integritätslevel und ein System mit einer höheren Wahrscheinlichkeit zufälliger Hardwareausfälle häufigere Wiederholungsprüfungen mit größerer Tiefe/Komplexität erforderlich, die zu zusätzlichen menschlichen Eingriffen führen;

– desto komplexer ist das System. Die Wahrscheinlichkeit eines zufälligen Hardwareausfalls hängt von der Anzahl der Bauteile und daher der Komplexität eines Systems ab. Ein komplexes System ist weniger leicht zu verstehen und daher anfälliger gegen die Einbringung systematischer Fehler. Zusätzlich macht es die Komplexität schwierig, die Fehler durch Analyse oder Test zu erkennen und kann dazu führen, dass Teil der Logik außer in seltenen Umständen nicht bearbeitet werden. Dies führt ebenfalls zu einer Beziehung zwischen der Wahrscheinlichkeit zufälliger Hardwareausfälle und der Wahrscheinlichkeit von Ausfällen infolge gemeinsamer Ursache.

Mehrere Ansätze werden gegenwärtig verwendet, um den CCF (β-Faktor, verschiedene griechische Buchstaben, α-Faktor, Binomial-Ausfallraten-Modell …) zu handhaben [20]. Zwei der gegenwärtigen Modelle werden in diesem informativen Anhang für den dritten Teil des oben beschriebenen Dreifachansatzes vorgeschlagen. Trotz Einschränkungen scheinen diese zum gegenwärtigen Zeitpunkt den besten Weg darzustellen, die Wahrscheinlichkeit eines Ausfalls infolge gemeinsamer Ursache zu handhaben:

– Das bekannte β-Faktor-Modell, das weit verbreitet und praxisnah ist, um mit Mehrkanalsystemen mit üblicherweise bis zu vier abhängigen Elementen umzugehen.

– Das Binomial-Ausfallraten-Modell [21] (auch als das Schockmodell bekannt), das für mehr als vier abhängige Elmente angewendet werden kann.

Bei Anwendung der β-Faktor- oder Schockmodelle auf ein E/E/PE-System steht man vor folgenden beiden Schwierigkeiten:

– Welcher Wert sollte für die Parameter gewählt werden? Viele Quellen (für ein Beispiel siehe Verweis [13]) schlagen Bereiche vor, innerhalb derer der Wert von β wahrscheinlich liegen wird, geben aber keinen Wert an, sondern überlassen es dem Anwender, eine Auswahl nach eigenem Belieben zu treffen. Um dieses Problem zu überwinden, basiert die β-Faktor-Methode dieses Anhangs auf dem System, das ursprünglich in Verweis [14] beschrieben wurde und das vor kurzem in Verweis [15] neu definiert wurde.

– Weder das β-Faktor-Modell noch das Schockmodell berücksichtigen die fortgeschrittenen Fähigkeiten des Diagnosetests moderner programmierbarer elektronischer Systeme, einen nicht gleichzeitigen Ausfall infolge gemeinsamer Ursache zu erkennen, bevor er ausreichend Zeit hatte, sich vollständig durchzusetzen. Um diesen Mangel zu überwinden, ist der in den Verweisen [14] und [15] beschriebene Ansatz um die Berücksichtigung der Wirkung von Diagnosetests bei der Schätzung des wahrscheinlichen Wertes von β verbessert worden.

Die Diagnosetest-Funktionen, die innerhalb eines programmierbaren elektronischen Systems ablaufen, vergleichen fortwährend seine Operationen mit vorher festgelegten Zuständen. Diese Zustände können in der Software oder Hardware hinterlegt sein (z. B. durch einen Watchdog-Timer). Unter diesem Blickwinkel kann man sich die Funktion des Diagnosetests wie einen zusätzlichen und teilweise diversitären Kanal parallel zum programmierbaren elektronischen System vorstellen.

Eine gegenseitige Überwachung zwischen Kanälen kann ebenfalls durchgeführt werden. Dieses Verfahren ist seit Jahren in zweikanaligen Verriegelungssystemen, die allein auf Relais basierten, verwendet worden. Mit Relaistechnik ist der kreuzweise Vergleich jedoch nur dann durchführbar, wenn die Kanäle den Zustand ändern. Dies macht solche Prüfungen ungeeignet für die Aufdeckung von nicht gleichzeitigen Ausfällen infolge gemeinsamer Ursache, wo Systeme für lange Zeit im gleichen Zustand (z. B. EIN) verbleiben. Mit programmierbarer elektronischer Technik jedoch kann die gegenseitige Überwachung mit einer hohen Wiederholungsfrequenz durchgeführt werden.

D.2 Anwendungsbereich der Methode

Der Anwendungsbereich der Methode ist auf Ausfälle infolge gemeinsamer Ursache innerhalb der Hardware begrenzt. Die Gründe sind u. a.:

– Das β-Faktor-Modell und das Schockmodell setzen die Wahrscheinlichkeit eines Ausfalls infolge gemeinsamer Ursache in Beziehung zur Wahrscheinlichkeit eines zufälligen Hardwareausfalls. Die Wahrscheinlichkeit von Ausfällen infolge gemeinsamer Ursache, die das System als Ganzes betreffen, ist von der Komplexität des Systems (möglicherweise beherrscht durch die Anwendersoftware) und nicht der Hardware allein abhängig. Offensichtlich können alle Berechnungen auf der Basis der Wahrscheinlichkeit eines zufälligen Hardwareausfalls die Komplexität der Software nicht berücksichtigen.

– Meldungen von Ausfällen infolge gemeinsamer Ursache sind allgemein auf Hardwareausfälle, also das Interessengebiet der Hardwarehersteller, begrenzt.
– Modelle für systematische Ausfälle (z. B. Softwareversagen) werden allgemein als nicht realisierbar angesehen.
– Die Maßnahmen nach IEC 61508-3 sind dazu vorgesehen, die Wahrscheinlichkeit eines softwarebezogenen Versagens infolge gemeinsamer Ursache auf ein für den angestrebten Sicherheits-Integritätslevel annehmbares Maß zu vermindern.

Deshalb bezieht sich die Schätzung der Wahrscheinlichkeit eines Ausfalls infolge gemeinsamer Ursache nach dieser Methode nur auf Ausfälle, die der Hardware zuzurechnen sind. Es sollte NICHT angenommen werden, dass diese Methode verwendet werden kann, um eine Gesamtausfallrate zu erhalten, die die Wahrscheinlichkeit eines softwarebezogenen Ausfalls einbezieht.

D.3 Durch die Methode berücksichtigte Punkte

Da die Sensoren, das Logik-Teilsystem und die Stellglieder beispielsweise unterschiedlichen Umgebungsbedingungen oder Diagnosetests mit unterschiedlicher Leistungsfähigkeit ausgesetzt sind, sollte die Methode für jedes dieser Teilsysteme getrennt angewendet werden. So befindet sich das Logik-Teilsystem eher in einer kontrollierten Umgebung, während die Sensoren möglicherweise im Außenbereich auf Rohrleitungen montiert und somit Wind und Wetter ausgesetzt sind.

Kanäle in programmierbarer elektronischer Technik haben die Möglichkeit, hoch entwickelte Diagnosetestfunktionen auszuführen. Diese können:

– einen hohen Diagnosedeckungsgrad innerhalb der Kanäle haben;
– zusätzliche redundante Kanäle überwachen;
– eine hohe Wiederholrate haben; und
– in einer größer werdenden Zahl von Fällen auch Sensoren und Stellglieder überwachen.

Ein großer Anteil von Ausfällen infolge gemeinsamer Ursache ereignet sich nicht in allen betroffenen Kanälen gleichzeitig. Deshalb kann, wenn die Frequenz der Diagnosetests ausreichend hoch ist, ein großer Anteil von Ausfällen infolge gemeinsamer Ursache aufgedeckt und daher vermieden werden, bevor sie alle verfügbaren Kanäle beeinträchtigen.

Nicht alle Eigenschaften eines mehrkanaligen Systems, die einen Einfluss auf die Widerstandsfähigkeit gegen Ausfälle infolge gemeinsamer Ursache haben, können durch Diagnosetests bewertet werden, aber diejenigen, die in Zusammenhang mit Diversität oder Unabhängigkeit stehen, werden dadurch wirksamer. Jede Eigenschaft, die bei einem nicht gleichzeitigen Ausfall infolge gemeinsamer Ursache die Zeit zwischen den Ausfällen der Kanäle wahrscheinlich vergrößert (oder den Anteil von gleichzeitigen Ausfällen infolge gemeinsamer Ursache verringert), erhöht die Wahrscheinlichkeit, dass ein Diagnosetest den Ausfall erkennt und die Anlage in einen sicheren Zustand bringt. Deshalb werden die Eigenschaften, die zur Widerstandsfähigkeit gegen Ausfälle infolge gemeinsamer Ursache gehören, eingeteilt in diejenigen, deren Wirksamkeit durch Diagnosetests verbessert werden soll, und diejenigen, bei denen dies nicht der Fall ist. Dies führt zu den beiden Spalten X und Y in Tabelle D.1.

Obwohl die Wahrscheinlichkeit von Ausfällen aller Kanäle infolge gemeinsamer Ursache für ein dreikanaliges System wahrscheinlich geringfügig kleiner ist als für ein zweikanaliges, wird zur Vereinfachung der β-Faktor-Methode vorausgesetzt, dass diese unabhängig von der Anzahl der betroffenen Kanäle ist, d. h. es wird vorausgesetzt, dass, falls ein Ausfall infolge gemeinsamer Ursache auftritt, er alle Kanäle betrifft. Das Schockmodell stellt eine alternative Methode dar.

Zur Kalibrierung der Methode sind keine bekannten Daten für hardwarebezogene Ausfälle infolge gemeinsamer Ursache verfügbar. Deshalb basieren die Tabellen in diesem Anhang auf einer ingenieurmäßigen Abschätzung.

Diagnosetest-Routinen werden manchmal nicht so betrachtet, als ob sie eine direkte Rolle für die Sicherheit spielen würden und erfahren daher nicht das gleiche Niveau der Qualitätssicherung wie die Routinen der Hauptsteuerungsfunktionen. Die Methode wurde auf der Basis entwickelt, dass der Diagnosetest eine Integ-

rität entsprechend des angestrebten Sicherheits-Integritätslevels hat. Deshalb sollten alle softwarebasierten Diagnosetest-Routinen mit Hilfe von Verfahren, die für den angestrebten Sicherheits-Integritätslevel geeignet sind, entwickelt werden.

D.4 Verwendung des β-Faktors, um die Wahrscheinlichkeit eines Ausfalls eines sicherheitsbezogenen E/E/PE-Systems durch Ausfälle infolge gemeinsamer Ursache zu berechnen

Betrachtet sei die Auswirkung von Ausfällen infolge gemeinsamer Ursache in einem mehrkanaligen System mit Diagnosetest innerhalb jedes Kanals.

Bei Verwendung des β-Faktor-Modells ist die Rate gefahrbringender Ausfälle infolge gemeinsamer Ursache:

$$\lambda_D \beta$$

wobei

λ_D die Rate gefahrbringender zufälliger Hardwareausfälle für jeden individuellen Kanal und β der β-Faktor ohne Diagnosetest ist, d. h. der Anteil von einkanaligen Ausfällen, die alle Kanäle beeinträchtigen.

Es sei weiter angenommen, dass Ausfälle infolge gemeinsamer Ursache alle Kanäle beeinträchtigen und die Zeit zwischen dem Ausfall des ersten Kanals und dem weiterer Kanäle klein ist im Vergleich zur Zeit zwischen aufeinander folgenden Ausfällen infolge gemeinsamer Ursache.

Wird weiter angenommen, dass in jedem Kanal Diagnosetests laufen und einen Teil der Ausfälle erkennen und aufdecken, so können wir alle Ausfälle in zwei Kategorien einteilen: in diejenigen außerhalb der Aufdeckung durch den Diagnosetest (und die daher nie erkannt werden können) und in diejenigen innerhalb dieser Aufdeckung (und die daher schließlich durch ihn erkannt werden).

Die Gesamtausfallrate aufgrund gefahrbringender Ausfälle infolge gemeinsamer Ursache ist dann gegeben durch:

$$\lambda_{DU}\beta + \lambda_{DD}\beta_D$$

Dabei ist

- λ_{DU} die Rate unerkannter gefahrbringender Ausfälle eines einzelnen Kanals, d. h. die Rate der Ausfälle, die außerhalb der Aufdeckung durch die Diagnosetests liegen. Eine Verringerung des β-Faktors, die aus der Wiederholrate des Diagnosetests herrührt, kann diesen Anteil der Ausfälle offensichtlich nicht beeinflussen;
- β der Faktor der Ausfälle infolge gemeinsamer Ursache für nicht erkennbare gefahrbringende Fehler, der gleich dem gesamten β-Faktor ist, der bei Abwesenheit von Diagnosetests zutreffend wäre;
- λ_{DD} die Rate erkannter gefahrbringender Ausfälle eines einzelnen Kanals, d. h. die Rate der Ausfälle eines einzelnen Kanals, die innerhalb der Aufdeckung des Diagnosetests liegt. Wenn die Wiederholrate des Diagnosetests hoch ist, wird ein Anteil der Ausfälle aufgedeckt werden und zu einer Verringerung des Werts von β führen, d. h. β_D;
- β_D der Faktor von Ausfällen infolge gemeinsamer Ursache für erkennbare gefahrbringende Fehler. Wenn die Wiederholrate des Diagnosetests zunimmt, fällt der Wert von β_D in zunehmendem Maße unterhalb von β;
- β erhält man aus Tabelle D.5, die die Ergebnisse von Tabelle D.4 verwendet, unter Verwendung des Ergebnisses $S = X + Y$ (siehe D.5);
- β_D erhält man aus Tabelle D.5, die die Ergebnisse von Tabelle D.4 verwendet, unter Verwendung des Ergebnisses $S_D = X(Z + 1) + Y$.

D.5 Schätzung von β unter Verwendung der Tabellen

Der β-Faktor sollte für das Sensor-, Logik- und Stellglied-Teilsystem getrennt berechnet werden.

Um die Wahrscheinlichkeit des Auftretens von Ausfällen infolge gemeinsamer Ursache zu verringern, sollte zuerst festgestellt werden, welche Maßnahmen ihr Auftreten einschränken können. Die Verwirklichung geeigneter Maßnahmen im System wird zu einer Verringerung des Wertes für den β-Faktor führen, der zur Schätzung der Wahrscheinlichkeit von Ausfällen aufgrund von Ausfällen infolge gemeinsamer Ursache verwendet wird.

Die Tabelle D.1 führt die Maßnahmen auf und enthält zugehörige Werte auf der Basis ingenieurmäßiger Schätzungen für den Beitrag, den jede Maßnahme zur Verringerung von Ausfällen infolge gemeinsamer Ursache liefert. Für die Bewertung der programmierbaren Elektronik einerseits sowie der Sensoren und Stellglieder andererseits werden in Tabelle D.1 eigene Spalten verwendet, da diese unterschiedlich behandelt werden.

Umfangreiche Diagnosetests, die die Aufdeckung von nicht gleichzeitigen Ausfällen infolge gemeinsamer Ursache ermöglichen, können in programmierbare elektronische Systeme integriert werden. Um diese Diagnosetests in der Abschätzung des β-Faktors zu berücksichtigen, ist der Gesamtbeitrag jeder Maßnahme in Tabelle D.1 auf der Basis ingenieurmäßiger Schätzungen in zwei Werte, X und Y, geteilt. Für jede Maßnahme liefert das Verhältnis X : Y das Ausmaß, bis zu dem der Beitrag dieser Maßnahme gegen Ausfälle infolge gemeinsamer Ursache durch Diagnosetests verbessert werden kann.

Der Anwender der Tabelle D.1 sollte feststellen, welche Maßnahmen für das betrachtete System zutreffen, und jeweils die Summe der zugehörigen Werte in den Spalten X_{LS} und Y_{LS} für das Logiksystem bilden, oder in den Spalten X_{SA} und Y_{SA} für die Sensoren und Stellglieder, wobei die Summen als X bzw. Y bezeichnet werden.

Die Tabellen D.2 und D.3 können verwendet werden, um aus der Frequenz und dem Diagnosedeckungsgrad einen Faktor Z zu bestimmen unter Berücksichtigung der wichtigen Anmerkung 4, die die Anwendung eines von null verschiedenen Wertes für Z eingrenzt. Das Ergebnis S wird dann wie oben beschrieben und unter Verwendung der folgenden Gleichungen berechnet (siehe vorhergehenden Abschnitt):

– $S = X + Y$, um den Wert für β_{int} zu erhalten (der β-Faktor für unerkannte Ausfälle); und

– $S_D = X(Z + 1) + Y$, um den Wert für $\beta_{D\ int}$ zu erhalten (β-Faktor für erkannte Ausfälle).

Dabei ist S oder S_D ein Ergebnis, das in Tabelle D.4 verwendet wird, um den geeigneten β_{int}-Faktor zu bestimmen.

β_{int} und $\beta_{D\ int}$ sind die Werte für den Ausfall infolge gemeinsamer, Ursache bevor der Einfluss verschiedener Grade von Redundanz berücksichtigt wird.

Tabelle D.1 – Bewertung programmierbarer Elektronik oder Sensoren/Stellglieder

Merkmal	Logik-Teilsystem		Sensoren und Stellglieder	
	X_{LS}	Y_{LS}	X_{SF}	Y_{SF}
Trennung				
Werden die Signalkabel für alle Kanäle an allen Stellen getrennt geführt?	1,5	1,5	1,0	2,0
Befinden sich die Kanäle des Logik-Teilsystems auf getrennten Leiterplatten?	3,0	1,0		
Sind die die Logik-Teilsysteme physikalisch in wirksamer Weise getrennt? Zum Beispiel in getrennten Schaltschränken.	2,5	0,5		
Wenn die Sensoren/Stellglieder zugeordnete Steuerelektroniken haben, befinden sich diese für jeden Kanal auf getrennten Leiterplatten?			2,5	1,5
Wenn die Sensoren/Stellglieder zugeordnete Steuerelektroniken haben, befinden sich diese für jeden Kanal im Innenbereich und in getrennten Schaltschränken?			2,5	0,5
Diversität/Redundanz				
Verwenden die Kanäle verschiedene Technologien, z. B. einer Elektronik oder programmierbare Elektronik und der andere Relais?	8,0			
Verwenden die Kanäle verschiedene elektronische Technologien, z. B. einer Elektronik, der andere programmierbare Elektronik?	6,0			
Verwenden die Sensoren verschiedene physikalische Prinzipien, z. B. Druck und Temperatur, Messung der Strömungsgeschwindigkeit mit einem Flügelradaufnehmer oder mit dem Doppler-Effekt?			9,0	
Verwenden die Geräte andere elektrische Prinzipien oder Konstruktionen, z. B. digital und analog, verschiedene Hersteller (nicht nur baugleiches Gerät mit verschiedenen Marken) oder verschiedene Technologien?			6,5	
Wird Diversität niedriger Wirksamkeit verwendet, z. B. Hardware-Diagnosetests unter Verwendung gleicher Technologie?	2,0	1,0		
Wird Diversität mittlerer Wirksamkeit verwendet, z. B. Hardware-Diagnosetests unter Verwendung unterschiedlicher Technologien?	3,0	2,0		
Wurden die Kanäle von verschiedenen Entwicklern entworfen, die während der Entwurfstätigkeit keine Kommunikation hatten?	1,5	1,5		
Werden während der Inbetriebnahme für jeden Kanal unterschiedliche Testmethoden und Personen eingesetzt?	1,0	0,5	1,0	2,0
Wird die Instandhaltung für jeden Kanal von anderen Personen zu anderen Zeiten durchgeführt?	3,0		3,0	
Komplexität/Entwurf/Anwendung/Reife/Erfahrung				
Schließt die Querverbindung zwischen den Kanälen den Austausch jeglicher Information, außer der für den Diagnosetest oder den Ausgangsvergleicher, aus?	0,5	0,5	0,5	0,5
Basiert der Entwurf auf Verfahren, die seit mehr als 5 Jahren in Betriebsmitteln im Feldeinsatz erfolgreich verwendet werden?	0,5	1,0	1,0	1,0
Gibt es mehr als 5 Jahre Erfahrung mit der gleichen Hardware unter ähnlichen Umgebungsbedingungen?	1,0	1,5	1,5	1,5
Ist das System einfach, z. B. nicht mehr als 10 Eingänge oder Ausgänge je Kanal?		1,0		
Werden die Eingänge und Ausgänge vor möglichen Überspannungen und Überströmen geschützt?	1,5	0,5	1,5	0,5
Sind alle Geräte und Bauteile konservativ dimensioniert? (z. B. Überdimensionierung mit einen Faktor 2 oder mehr)	2,0		2,0	
Beurteilung/Analyse und Rückmeldung von Daten				
Sind die Ergebnisse der Ausfallarten- und Auswirkungsanalyse (FMEA) oder der Fehlerbaumanalyse (FTA) ausgewertet worden, um Quellen von Ausfällen infolge gemeinsamer Ursache festzustellen, und sind zuvor bestimmte derartige Quellen durch den Entwurf beseitigt worden?		3,0		3,0
Wurden Ausfälle infolge gemeinsamer Ursache, die in den Entwurfsüberprüfungen besprochen wurden, mit dem Ergebnis in den Entwurfsprozess zurückgemeldet? (Dokumentierter Nachweis der Tätigkeiten der Entwurfsüberprüfung ist notwendig.)		3,0		3,0
Werden alle Ausfälle im Feld vollständig analysiert und in den Entwurfsprozess zurückgemeldet? (Dokumentierter Nachweis des Verfahrens ist notwendig.)	0,5	3,5	0,5	3,5

Tabelle D.1 *(fortgesetzt)*

Merkmal	Logik-Teilsystem		Sensoren und Stellglieder	
	X_{LS}	Y_{LS}	X_{SF}	Y_{SF}
Prozeduren/Bedienerschnittstelle				
Gibt es eine schriftliche Arbeitsanweisung, die sicherstellt, dass alle Bauteilausfälle (oder Verschlechterungen) erkannt, deren Ursachen festgestellt und ähnliche Objekte im Hinblick auf mögliche ähnliche Ausfallursachen überprüft werden?		1,5	0,5	1,5
Sind Verfahren eingerichtet, die sicherstellen, dass die Instandhaltung (einschließlich Anpassung oder Kalibrierung) aller Teile der unabhängigen Kanäle zeitlich versetzt erfolgt und zusätzlich zu den manuellen Prüfungen, die der Instandhaltung folgen, zwischen der Beendigung der Instandhaltung eines Kanals und dem Start der Instandhaltung des nächsten Kanals ein einwandfreier Durchlauf des Diagnosetests erfolgt?	1,5	0,5	2,0	1,0
Schreiben die Instandhaltungsanweisungen vor, dass alle Teile der redundanten Systeme (z. B. Kabel usw.), die voneinander unabhängig sein müssen, nicht geändert werden dürfen?	0,5	0,5	0,5	0,5
Wird die Instandhaltung aller Leiterplatten usw. extern in einem qualifizierten Reparaturzentrum ausgeführt und gehen alle reparierten Einheiten durch einen vollständigen Test vor der Installation?	0,5	1,0	0,5	1,5
Hat das System einen niedrigen Diagnosedeckungsgrad (60 % bis 90 %) und zeigt Ausfälle auf dem Niveau eines vor Ort austauschbaren Moduls?	0,5			
Hat das System einen mittleren Diagnosedeckungsgrad (90 % bis 99 %) und zeigt Ausfälle auf dem Niveau eines vor Ort austauschbaren Moduls?	1,5	1,0		
Hat das System einen hohen Diagnosedeckungsgrad (> 99 %) und zeigt Ausfälle auf dem Niveau eines vor Ort austauschbaren Moduls?	2,5	1,5		
Meldet der Diagnosetest des Systems Ausfälle auf dem Niveau eines vor Ort austauschbaren Moduls?			1,0	1,0
Kompetenz/Ausbildung/Sicherheitskultur				
Sind die Entwickler dazu ausgebildet worden (mit Ausbildungsdokumentation), die Ursachen und Folgen von Ausfällen infolge gemeinsamer Ursache zu verstehen?	2,0	3,0	2,0	3,0
Sind die Instandhalter dazu ausgebildet worden (mit Ausbildungsdokumentation), die Ursachen und Folgen von Ausfällen infolge gemeinsamer Ursache zu verstehen?	0,5	4,5	0,5	4,5
Überwachung der Umgebungsbedingungen				
Ist der Zutritt für Personal eingeschränkt (z. B. verschlossene Schaltschränke, unzugängliche Position)?	0,5	2,5	0,5	2,5
Arbeitet das System wahrscheinlich auch ohne äußere Maßnahmen innerhalb des Temperatur-, Feuchte-, Korrosions-, Staub- und Vibrationsbereiches usw., für das es getestet worden ist?	3,0	1,0	3,0	1,0
Sind alle Signal- und Energieleitungen an allen Stellen getrennt?	2,0	1,0	2,0	1,0
Test unter Umgebungsbedingungen				
Ist das System auf Beständigkeit gegen alle wichtigen Umgebungseinflüsse (z. B. EMV, Temperatur, Vibration, Schock, Feuchte) entsprechend eines angemessenen, in anerkannten Normen festgelegten Niveaus getestet worden?	10,0	10,0	10,0	10,0

ANMERKUNG 1 Mehrere Merkmale beziehen sich auf den Betrieb des Systems, der während des Entwurfs möglicherweise schwierig vorherzubestimmen ist. In diesen Fällen sollten die Entwickler vernünftige Abschätzungen machen und weiterhin sicherstellen, dass dem Endanwender des Systems z. B. die Verfahren bewusst gemacht werden, die zum Erreichen des vorgesehenen Sicherheits-Integritätslevels notwendig sind. Dieses könnte auch durch Aufnahme der notwendigen Information in die Begleitdokumentation erfolgen.

ANMERKUNG 2 Die Werte in den Spalten *X* und *Y* basieren auf ingenieurmäßigen Abschätzungen und berücksichtigen sowohl die direkten als auch die indirekten Wirkungen der Merkmale in der ersten Spalte. Zum Beispiel führt die Verwendung eines vor Ort austauschbaren Moduls zu:

– Reparaturen, die unter kontrollierten Bedingungen durch den Hersteller ausgeführt werden anstelle von (möglicherweise schlechten) Reparaturen unter weniger geeigneten Bedingungen vor Ort. Dies führt zu einem Beitrag in der Spalte Y, weil die Möglichkeit von systematischen Ausfällen (und daher Ausfälle infolge gemeinsamer Ursache) verringert wird.

– einer Verringerung der Notwendigkeit manueller Eingriffe vor Ort und der Fähigkeit des schnellen Austauschs fehlerhafter Module möglicherweise während des laufenden Betriebes. Dadurch wird die Wirksamkeit der Diagnose von Ausfällen erhöht, bevor diese zu Ausfällen infolge gemeinsamer Ursache werden. Dies führt zu einem hochwertigen Eintrag in der X-Spalte.

Tabelle D.2 – Werte für *Z* – programmierbare Elektronik

Diagnose-deckungsgrad	Diagnose-Testintervall		
	Geringer als 1 min	Zwischen 1 min und 5 min	Größer als 5 min
≥ 99 %	2,0	1,0	0
≥ 90 %	1,5	0,5	0
≥ 60 %	1,0	0	0

Tabelle D.3 – Werte für *Z* – Sensoren oder Stellglieder

Diagnose-deckungsgrad	Diagnose-Testintervall			
	Geringer als 2 h	Zwischen 2 h und zwei Tagen	Zwischen zwei Tagen und einer Woche	Größer als eine Woche
≥ 99 %	2,0	1,5	1,0	0
≥ 90 %	1,5	1,0	0,5	0
≥ 60 %	1,0	0,5	0	0

ANMERKUNG 1 Die Methode ist am wirksamsten, wenn die Liste der Kategorien aus Tabelle D.1 gleichmäßig berücksichtigt wird. Deshalb wird dringend empfohlen, dass das Ergebnis aus den Spalten *X* und *Y* für jede Kategorie nicht kleiner sein sollte als das Gesamtergebnis aus den Spalten *X* und *Y*, dividiert durch 20. Wenn zum Beispiel das Ergebnis ($X + Y$) die Zahl 80 ergibt, sollte keine der Kategorien (z. B. die Prozeduren/Bedienerschnittstelle) für ($X + Y$) ein Ergebnis kleiner als 4 haben.

ANMERKUNG 2 Wenn die Tabelle D.1 verwendet wird, sind alle Werte aller auftretenden Merkmale zu berücksichtigen. Die Bewertung wurde für Merkmale entworfen, die sich nicht gegenseitig ausschließen. Zum Beispiel kann ein System, bei dem sich die Kanäle des Logik-Teilsystems in getrennten Baugruppenträgern befinden, sowohl für „Sind die Kanäle des Logik-Teilsystems in getrennten Schaltschränken?“ und „Sind die Kanäle des Logik-Teilsystems auf getrennten Leiterplatten?“ Punkte erzielen.

ANMERKUNG 3 Wenn Sensoren oder Stellglieder auf programmierbarer Elektronik basieren und sie sich innerhalb des gleichen Gebäudes oder Fahrzeuges in einem Gehäuse mit dem Gerät befinden, das den Hauptteil des Logik-Teilsystems ausmacht, sollten sie wie ein Teil des Logik-Teilsystems behandelt werden. Wenn sie sich dagegen nicht im gleichen Gebäude (oder Fahrzeug) befinden, sollten sie als Sensoren oder Stellglieder behandelt werden.

ANMERKUNG 4 Wird ein Wert für *Z* ungleich null verwendet, sollte sichergestellt werden, dass die EUC-Einrichtung in den sicheren Zustand versetzt wird, bevor ein nicht gleichzeitiger Ausfall infolge gemeinsamer Ursache beide Kanäle beeinträchtigen kann. Die Zeit, um diesen sicheren Zustand zu erreichen, sollte kleiner sein als das beanspruchte Diagnosetestintervall. Ein Wert für *Z* ungleich null kann nur eingesetzt werden, wenn

- das System bei erkanntem Fehler eine automatische Abschaltung einleitet; oder
- die sichere Abschaltung nicht nach dem ersten Fehler [9] erfolgt, die Diagnosetests jedoch:
  - den Ort bestimmen und die Fähigkeit der Lokalisierung des Fehlers haben; und
  - weiterhin imstande sind, die EUC-Einrichtung nach Erkennen jedes Folgefehlers in den sicheren Zustand zu versetzen; oder
- eine formale Arbeitsanweisung in Kraft ist, die sicherstellt, dass die Ursache jedes aufgedeckten Fehlers vollständig innerhalb des beanspruchten Diagnosetestintervall ermittelt wird; und

[9] Zur Identifikation eines Fehlers sollte der Betrieb des Systems mit berücksichtigt werden. Ein einfaches 2oo3-System sollte nach einem identifizierten Fehler innerhalb der Zeiten der Tabellen D.2 und D.3 abgeschaltet (oder repariert) werden. Wenn das nicht erfolgt, kann ein Fehler eines weiteren Kanals dazu führen, dass zwei jetzt fehlerhafte Kanäle den dritten (intakten) überstimmen. Ein System, das sich automatisch zu einem 1oo2-System rekonfiguriert, wenn ein Kanal fehlerhaft wird, und das sich automatisch abschaltet, sollte ein zweiter Fehler auftreten, hat eine höhere Wahrscheinlichkeit der Aufdeckung eines Fehlers im zweiten Kanal, so dass ein Wert für *Z* ungleich null beansprucht werden kann.

– wenn der Fehler zu einem Ausfall infolge gemeinsamer Ursache führen kann, die Anlage unmittelbar abgeschaltet wird; oder

– der fehlerhafte Kanal innerhalb des beanspruchten Diagnose-Testintervalls repariert wird.

ANMERKUNG 5 In der Prozessindustrie ist es wahrscheinlich nicht machbar, die EUC-Einrichtung innerhalb des in Tabelle D.2 angegebenen Diagnosetestintervalls abzuschalten, wenn ein Fehler erkannt wird. Diese Methode sollte nicht als Anforderung an Prozessanlagen angesehen werden, diese abzuschalten, wenn solche Fehler erkannt werden. Sollte jedoch keine Abschaltung vorgesehen sein, kann der β-Faktor nicht aufgrund von Diagnosetests der programmierbaren Elektronik vermindert werden. In einigen Industriezweigen kann eine Abschaltung innerhalb der angegebenen Zeit möglich sein. In diesen Fällen kann ein Wert für Z ungleich null verwendet werden.

ANMERKUNG 6 Wenn Diagnosetests modular ausgeführt werden, ist die Zeit in den Tabellen D.2 und D.3 die Zeit zwischen den erfolgreichen Beendigungen aller Diagnosetestmodule. Der Diagnosedeckungsgrad wird durch alle Module der Diagnose erreicht.

Tabelle D.4 – Berechnung von β_{int} oder $\beta_{D\ int}$

Bewertung (S oder S_D)	**Entsprechender Wert von β_{int} oder $\beta_{D\ int}$ für:**	
	Logik-Teilsystem	**Sensoren oder Stellglieder**
120 oder darüber	0,5 %	1 %
70 bis 120	1 %	2 %
45 bis 70	2 %	5 %
kleiner als 45	5 %	10 %
ANMERKUNG 1 Die höchsten Werte von $\beta_{D\ int}$ in dieser Tabelle sind niedriger, als sie üblicherweise verwendet werden. Dies spiegelt die Verwendung von an anderer Stelle in dieser Norm festgelegten Verfahren zur Verringerung der Wahrscheinlichkeit systematischer Ausfälle insgesamt und den daraus hervorgehenden Ausfällen infolge gemeinsamer Ursache wider. ANMERKUNG 2 Werte von $\beta_{D\ int}$ kleiner als 0,5 % für das Logik-Teilsystem und kleiner als 1 % für die Sensoren sind schwierig zu rechtfertigen.		

Das aus Tabelle D.4 abgeleitete β_{int} steht für den Ausfall infolge gemeinsamer Ursache eines 1oo2-Systems. Für andere Redundanzstufen (MooN) ändert sich dieser Wert von β_{int} entsprechend den Festlegungen in Tabelle D.5, um den endgültigen Wert von β zu erhalten.

Tabelle D.5 kann auch für die Bestimmung des endgültigen Wertes von β_D verwendet werden, wo jedoch β_{int} vorhanden ist, kann dies für $\beta_{D\ int}$ ersetzt werden.

ANMERKUNG 7 Siehe [25] in den Literaturhinweisen für zugehörige themenbezogene Informationen (zur PDS-Methode).

Tabelle D.5 – Berechnung von β für Redundanzsysteme größer als 1oo2

MooN		**N**			
		2	3	4	5
M	1	β_{int}	0,5 β_{int}	0,3 β_{int}	0,2 β_{int}
	2	-	1,5 β_{int}	0,6 β_{int}	0,4 β_{int}
	3	-	-	1,75 β_{int}	0,8 β_{int}
	4	-	-	-	2 β_{int}

D.6 Beispiele für die Anwendung der β-Faktor-Methode

Um diese Methode zu erläutern, wurden in der Tabelle D.6 einige einfache Beispiele für programmierbare Elektronik durchgerechnet.

Für Kategorien ohne Diversität oder Redundanz wurden typische Werte für *X* und *Y* verwendet. Diese wurden durch Halbierung der höchsten Bewertung für diese Kategorie erhalten.

In den Beispielen für diversitäre Systeme wurden die Werte für die Kategorie Diversität/Redundanz aus folgenden Merkmalen in der Tabelle D.1 betrachteten Merkmalen abgeleitet:

– Ein System ist elektronisch, das andere nutzt Relaistechnik;
– die Hardware-Diagnosetests verwenden unterschiedliche Technologien;
– die Entwickler hatten während der Entwicklung keinen Kontakt zueinander;
– die Inbetriebnahme des Systems erfolgte durch unterschiedliche Testmethoden und unterschiedliches Testpersonal; und
– die Instandhaltung erfolgt durch verschiedene Mitarbeiter zu verschiedenen Zeiten.

In den Beispielen für redundante Systeme wurden die Werte für die Kategorie Diversität/Redundanz aus der Eigenschaft abgeleitet, dass die Hardwarediagnose durch ein unabhängiges System erfolgt unter Verwendung der gleichen Technologie, wie sie auch für das redundante System selbst eingesetzt wird.

Sowohl für die Systeme mit Redundanz als auch für das mit Diversität wurde ein Maximal- und ein Minimalwert für *Z* verwendet, was zu insgesamt vier Beispielsystemen führt.

Tabelle D.6 – Beispielwerte für programmierbare Elektroniken

Kategorie		Diversitäres System mit gutem Diagnosetest	Diversitäres System mit schlechtem Diagnosetest	Redundantes System mit gutem Diagnosetest	Redundantes System mit schlechtem Diagnosetest
Trennung	*X*	3,50	3,50	3,50	3,50
	Y	1,50	1,50	1,50	1,50
Diversität/Redundanz	*X*	14,50	14,50	2,00	2,00
	Y	3,00	3,00	1,00	1,00
Komplexität/Entwurf/...	*X*	2,75	2,75	2,75	2,75
	Y	2,25	2,25	2,25	2,25
Beurteilung/Analyse ...	*X*	0,25	0,25	0,25	0,25
	Y	4,75	4,75	4,75	4,75
Prozeduren/Bediener-schnittstelle	*X*	3,50	3,50	3,50	3,50
	Y	3,00	3,00	3,00	3,00
Kompetenz/Ausbildung/...	*X*	1,25	1,25	1,25	1,25
	Y	3,75	3,75	3,75	3,75
Überwachung der Umgebungsbedingungen	*X*	2,75	2,75	2,75	2,75
	Y	2,25	2,25	2,25	2,25
Test unter Umgebungsbedingungen	*X*	5,00	5,00	5,00	5,00
	Y	5,00	5,00	5,00	5,00
Diagnosedeckungsgrad	*Z*	2,00	0,00	2,00	0,00
Gesamt *X*		33,5	33,5	21	21
Gesamt *Y*		25,5	25,5	23,5	23,5
Ergebnis *S*		59	59	44,5	44,5
β		2 %	2 %	5 %	5 %
Ergebnis S_D		126	59	86,5	44,5
β_D		0,5 %	2 %	1 %	5 %
Diversitäres System 1oo2 (Tabelle D.5)		0,5 %	2 %		
Nichtdiversitäres Dreifachsystem mit 2oo3-Mehrheitsentscheid (Tabelle D.5)				1,5 %	7,5 %

D.7 Binomialverteilte Ausfallrate (Schockmodell) – CCF-Ansatz

Die Erfahrungswerte aus dem Feld in Bezug auf den Ausfall infolge gemeinsamer Ursache (CCF) zeigen, dass im Gegensatz zu zahlreichen Doppelausfällen, einigen Dreifachausfällen und eventuell einem vierfachen Ausfall keine Mehrfachausfälle über Vierfachausfälle hinaus jemals beobachtet wurden, die eine eindeutige einzelne Ursache hatten und während der Sicherheitsanalyse nicht erkannt werden konnten. Demzufolge verringert sich die Wahrscheinlichkeit von mehrfach abhängigen Ausfällen, wenn die Größenordnung der CCF zunimmt. Wenn daher die Auslegung des β-Faktor-Modells für Doppelausfälle realistisch und für Dreifachausfälle etwas pessimistisch ist, wird sie für Vierfach- und weitere Mehrfachausfälle all zu konservativ. In einem Beispiel wird ein typisches technisches Sicherheitssystem betrachtet, das auf einem Erdölfeld n (z. B. n = 150) fördernde Bohrlöcher absperrt, wenn eine Verstopfung einer Austrittsöffnung eintritt. Natürlich können 2, 3 oder gar 4 Bohrlöcher infolge eines nicht eindeutigen CCF ausfallen, aber nicht die durch den β-Faktor modellierten n Bohrlöcher (andernfalls wäre der CCF eindeutig und sollte als ein einzelner Ausfall analysiert werden). Ein weiteres typisches Beispiel tritt ein bei verschiedenen gleichzeitigen Sicherheitsstufen. Betrachtet man zum Beispiel den möglichen CCF unter Sensoren zweier verschiedener Schutzebenen, kann dies bedeuteten, die CCFs unter sechs Sensoren (d. h. drei Sensoren auf jeder Ebene) zu betrachten.

Es wurden mehrere Modelle vorgeschlagen [18], um diese Schwierigkeiten zu umgehen, die meisten von ihnen benötigen aber so viele Zuverlässigkeitsparameter (z. B. verschiedene griechische Buchstaben oder α-Modelle), dass sie unrealistisch werden. Unter ihnen bietet jedoch die durch Vesely 1977 eingeführte und von Atwood 1986 verbesserte binomialverteilte Ausfallrate (Schockmodell) eine pragmatische Lösung [18, 19] an. Das Prinzip besteht darin, dass, wenn ein CCF auftritt, dies einem Schockereignis für die ent-

sprechenden Bauteile gleichkommt. Diese Störung kann letal (d. h. gleiche Auswirkung wie beim β-Faktor-Modell) oder nicht letal sein und in diesem Fall gibt es nur eine gewisse Wahrscheinlichkeit, dass ein gegebenes Bauteil infolge des Schocks ausfällt. Dann ist die Wahrscheinlichkeit von k Ausfällen infolge eines nicht letalen Schocks binomial verteilt.

Dieses Model benötigt nur drei zu implementierende Parameter:

– ω letale Schockrate;

– ρ nicht letale Schockrate;

– γ bedingte Wahrscheinlichkeit des Ausfalls eines Bauteils in Anbetracht eines nicht letalen Schocks.

Bild D.2 zeigt ein Beispiel zur Implementierung dieser Methode mit Hilfe eines Fehlerbaums.

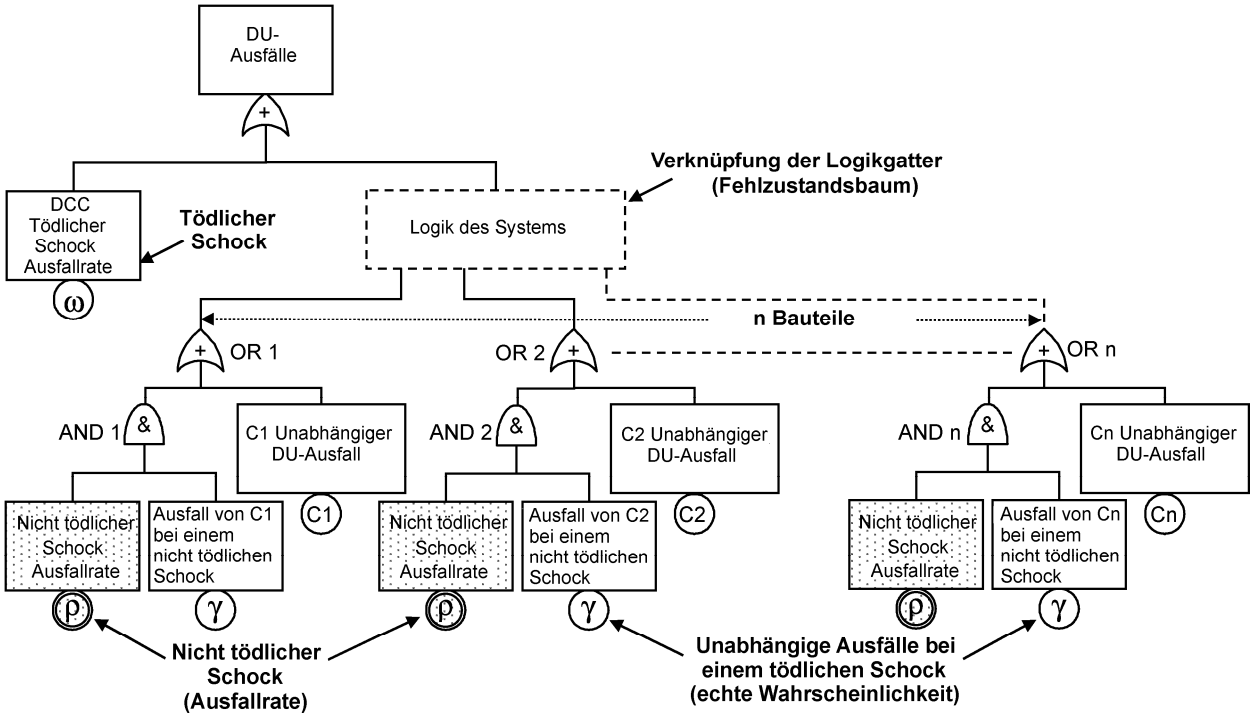


Bild D.2 – Umsetzung des Schockmodells anhand von Fehlerbäumen

Identische Bauteile können mit dem β-Faktor-Modell verbunden werden, indem β in zwei Teile aufgeteilt wird, β_L und β_{NL}:(L = letal = letal, NL= nicht letal = nicht letal)

– $\beta = \beta_L + \beta_{NL}$

– Ausfallrate infolge eines letalen Schocks: $\lambda_{DU} \times \beta_L$

– Ausfallrate infolge eines nicht letalen Schocks: $\lambda_{DU} \times \beta_{NL}$

– unabhängige Ausfallrate: $\lambda_{DU}[1 - (\beta_L + \beta_{NL})]$

In dem in Bild D.2 gezeigten Fehlerbaum wird dies zu:

– letale Schockrate: $\omega = \lambda_{DU} \times \beta_L$

– nicht letale Schockrate: $\rho = \lambda_{DU} \times \beta_{NL}/\gamma$

Das Hauptproblem ist wie immer, die Werte der drei Parameter (ω, ρ, γ) oder (β_L, β_{NL}, $\tilde{\gamma}$) einzuschätzen. Die Literatur [19] liefert einige Hinweise und stellt weitere Quellenangaben über statistische Verfahren zur Verfügung, die es ermöglichen, mit Hilfe der Daten aus den Rückmeldungen aus dem Feld (ω, ρ, γ) einzuschätzen.

Wenn keine Daten verfügbar sind, können die ingenieurmäßigen Abschätzungen über pragmatische Ansätze erfolgen. Im folgenden Verfahren kann zum Beispiel die Fehlerbaummodellierung angewendet werden, wenn mehr als drei gleichartige Einheiten vorhanden sind:

1) Schätzen von β wie bei der β-Faktor-Methode,

2) Berücksichtigung, dass β_L vernachlässigbar klein ist ($\beta_{NL} = \beta$),

3) Schätzen von γ, um sicher zu sein, konservative Ergebnisse zu erhalten. Wird berücksichtigt, dass Doppelausfälle einen mindestens 10-mal größeren Einfluss nehmen als die Vierfachausfälle (eine zweifellos konservative Hypothese), kann die folgende Formel zur Anwendung kommen:

$$\gamma = \sqrt{\frac{C_N^2}{10.C_N^4}}$$

Dabei ist

N die Anzahl der gleichartigen Einheiten;

C_N^2 die Anzahl möglicher Doppelausfälle; und

C_N^4 die Anzahl möglicher Vierfachausfälle.

4) Berechnen von ρ abhängig von der Anzahl N gleichartiger Einheiten:

$$p = \frac{\beta\lambda_{DU}}{C_N^2\gamma^2 + C_N^3\gamma^3}$$

Bei dieser Vorgehensweise sind die Hauptbeitragenden die Doppel- und Dreifachausfälle und die Ergebnisse sind im Vergleich zu denjenigen der β-Faktor-Methode mit nur drei Bauteilen konservativ. Die CCF Doppel- und Dreifachausfälle sind zufriedenstellend berücksichtigt und unrealistische Mehrfachausfälle werden nicht vollständig vernachlässigt.

Dieses Modell kann sehr einfach in Fehlerbaumberechnungsmodelle wie in Anhang B, z. B. Fehlerbäume in B.4.3, implementiert werden. Dies ermöglicht, Sicherheitssysteme mit sehr vielen gleichartigen Bauteilen auf einfache Art und Weise zu handhaben.

D.8 Literaturhinweise

[13] bis [15], [20] und [21] in den Literaturhinweisen geben hilfreiche Informationen zu Ausfällen infolge gemeinsamer Ursache.

Anhang E
(informativ)

Beispiele für die Anwendung der Tabellen zur Sicherheitsintegrität der Software aus IEC 61508-3

E.1 Allgemeines

Dieser Anhang enthält zwei ausgearbeitete Beispiele für die Anwendung der Tabellen zur Sicherheitsintegrität der Software in IEC 61508-3, Anhang A:

a) Sicherheits-Integritätslevel 2: Ein sicherheitsbezogenes programmierbares elektronisches System, wie es für einen Prozess innerhalb einer chemischen Anlage benötigt wird;

b) Sicherheits-Integritätslevel 3: Eine Anwendung mit Abschaltung, die auf einer Hochsprache basiert.

Diese Beispiele zeigen, wie Softwareentwicklungsverfahren aus den Tabellen der Anhänge A und B von IEC 61508-3 unter bestimmten Umständen ausgewählt werden könnten.

Es ist zu betonen, dass diese Veranschaulichungen keine vollständigen Anwendungen der Norm auf diese Beispiele darstellen. IEC 61508-3 macht an mehreren Stellen eindeutig klar, dass es angesichts der großen Anzahl der Einflussfaktoren, die die systematische Eignung der Software beeinflussen, unmöglich ist, einen Algorithmus für die Kombination der Verfahren und Methoden anzugeben, der für jede gegebene Anwendung zutreffend ist.

Für ein reales System sollten alle Einträge in den Tabellen durch eine dokumentierte Begründung gestützt werden, um zu zeigen, dass alle gemachten Anmerkungen richtig und für das betrachtete System und die Anwendung angemessen sind. Diese Begründung wird naheliegenderweise durch Bezug auf die Anleitung in IEC 61508-3, Anhang C, unterstützt, die die erstrebenswerten Eigenschaften erörtert, die, wenn in der entsprechenden Lebenszyklusphase erreicht, glaubhaft ein Vertrauen begründen können, dass die endgültige Software eine ausreichende systematische Sicherheitsintegrität besitzt.

E.2 Beispiel für den Sicherheits-Integritätslevel 2

Dieses Beispiel ist ein sicherheitsbezogenes programmierbares elektronisches System im Sicherheits-Integritätslevel 2, das für einen Prozess innerhalb einer chemischen Anlage benötigt wird. Das sicherheitsbezogene programmierbare elektronische System benutzt die Programmiersprache Kontaktplan für das Anwendungsprogramm und ist eine Veranschaulichung für Anwendungsprogrammierung in einer Programmiersprache mit eingeschränktem Sprachumfang.

Die Anwendung besteht aus mehreren Reaktionskesseln, die durch Zwischenspeicher verbunden sind. An bestimmten Punkten im Reaktionsablauf werden sie mit Inertgas gefüllt, um eine Zündung und Explosionen zu vermeiden. Die Funktionen des programmierbaren elektronischen sicherheitsbezogenen Systems beinhalten: Empfangen der Eingaben der Sensoren, Ansteuern und Verriegeln der Ventile, Pumpen und Aktoren, Erkennen gefahrbringender Situationen und Herbeiführen des Alarms, Bereitstellen der Schnittstelle zu einem Leitsystem entsprechend der Spezifikation der Anforderungen an die Sicherheit.

Annahmen:

– das programmierbare elektronische sicherheitsbezogene System ist eine SPS;
– die Gefährdungs- und Risikoanalyse hat ergeben, dass ein programmierbares elektronisches sicherheitsbezogenes System erforderlich ist, und dass für diese Anwendung der Sicherheits-Integritätslevel 2 (durch Anwendung der IEC 61508-1 und IEC 61508-2) erforderlich ist;
– obwohl die Steuerung in Echtzeit arbeitet, wird nur eine verhältnismäßig langsame Reaktionszeit benötigt;
– es gibt Schnittstellen zum Anlagenfahrer und zum Leitsystem;

– der Source-Code der Systemsoftware und die Konstruktionsunterlagen der SPS stehen für eine Untersuchung nicht zur Verfügung, sie ist aber für den Sicherheits-Integritätslevel 2 nach IEC 61508 qualifiziert worden;
– das Anwendungsprogramm wurde in der Programmiersprache Kontaktplan und unter Verwendung des Entwicklungssystems des SPS-Lieferanten erstellt;
– der Programmcode braucht nur auf einem einzigen Typ von SPS zu laufen;
– die gesamte Softwareentwicklung wurde von einer vom Softwareteam unabhängigen Person überprüft;
– eine vom Softwareteam unabhängige Person war beim Validierungstest anwesend und genehmigte ihn;
– Modifikationen (falls erforderlich) benötigen die Autorisierung einer vom Softwareteam unabhängigen Person.

ANMERKUNG 1 Zur Definition einer unabhängigen Person siehe IEC 61508-4.

ANMERKUNG 2 Für Informationen zur Aufteilung der Verantwortlichkeit zwischen dem SPS-Lieferanten und dem Anwender bei der Verwendung einer Programmiersprache mit eingeschränktem Sprachumfang siehe Anmerkungen zu IEC 61508-3, 7.4.2, 7.4.3, 7.4.4 und 7.4.5.

Die folgenden Tabellen zeigen, wie IEC 61508-3, Anhang A, für diese Anwendung interpretiert werden kann.

Tabelle E.1 – Spezifikation der Anforderungen an die Sicherheit der Software

(siehe IEC 61508-3, 7.2)

Verfahren/Maßnahme		siehe	SIL 2	Interpretation in dieser Anwendung
1a	Semiformale Methoden	Tabelle B.7	+	Ursache-Folgen-Diagramme, Ablaufdiagramme, Funktionsbausteine. Typischerweise verwendet für die Spezifikation der Anforderungen an die Software einer SPS-Anwendung
1b	Formale Methoden	B.2.2 C.2.4	+	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet
2	Vorwärtsverfolgbarkeit von den Anforderungen an die Sicherheit des Systems zu den Anforderungen an die Sicherheit der Software	C.2.11	+	Überprüfen auf Vollständigkeit: Überprüfung, um sicherzustellen, dass alle Anforderungen an die Sicherheit des Systems durch die Anforderungen an die Sicherheit der Software angesprochen werden
3	Rückwärtsverfolgbarkeit von den Anforderungen an die Sicherheit zu den wahrgenommenen Sicherheitsbedürfnissen	C.2.11	+	Verringern der Komplexität und Funktionalität: Überprüfung, um sicherzustellen, dass alle Anforderungen an die Sicherheit der Software tatsächlich zur Erfüllung der Anforderungen an die Sicherheit des Systems benötigt werden
4	Rechnergestützte Spezifikationswerkzeuge, die die obigen Verfahrenen/Maßnahmen zu angemessen unterstützen	B.2.4	+	Entwicklungswerkzeuge durch den SPS-Hersteller geliefert

ANMERKUNG 1 In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.

ANMERKUNG 2 Die Anforderungen an die Sicherheit der Software wurden in natürlicher Sprache festgelegt.

Tabelle E.2 – Softwareentwurf und Softwareentwicklung – Entwurf der Softwarearchitektur

(siehe IEC 61508-3, 7.4.3)

Verfahren/Maßnahme		siehe	SIL 2	Interpretation in dieser Anwendung
1	Fehlererkennung	C.3.1	+	Kontrolle des Datenbereichs, des Watchdog-Zeitglieds, der E/As, der Kommunikation. Auslösen eines Alarmes, falls Abweichungen auftreten (siehe 3a)
2	Fehlererkennende Codes	C.3.2	+	Eingebettet in Anwenderoptionen – sorgfältige Auswahl erforderlich
3a	Assertion-Programmierung (en: Failure assertion programming)	C.3.3	+	Einiger Teile der SPS-Kontaktplan-Software dazu verwenden, bestimmte wichtige Sicherheitsbedingungen zu testen (siehe 1)
3b	Diversitäre Überwachungseinrichtungen (mit Unabhängigkeit zwischen der überwachenden und der überwachten Funktion innerhalb des selben Rechners)	C.3.4	+	Nicht bevorzugt: Erhöhte Softwarekomplexität zur Gewährleistung von Unabhängigkeit
3c	Diversitäre Überwachungseinrichtungen (mit Trennung zwischen dem überwachenden und dem überwachten Rechner)	C.3.4	+	Kontrolle der zulässigen E/A-Kombinationen mit einer Sicherheitsüberwachung in unabhängiger Hardware
3d	Diversitäre Redundanz, Implementierung der gleichen Spezifikation der Anforderungen an die Sicherheit der Software	C.3.5	o	Nicht bevorzugt: Ungenügend erhöhter sicherheitstechnischer Nutzen gegenüber 3c
3e	Funktionale diversitäre Redundanz, Implementierung verschiedener Spezifikation der Anforderungen an die Sicherheit der Software	C.3.5	o	Nicht bevorzugt: Im Wesentlichen durch 3c erreicht
3f	Rückwärtsregeneration	C.3.6	+	Eingebettet mit Anwenderoptionen – sorgfältige Auswahl erforderlich
3g	Zustandsloser Software-Entwurf (oder Entwurf eingeschränkter Zustände)	C.2.12	o	Nicht verwendet. Prozesssteuerung benötigt Zustände, um Anlagenzustände zu speichern
4a	Regeneration durch Wiederholung	C.3.7	+	Verwendet, wie durch die Anwendung erforderlich (siehe 2 und 3c)
4b	Abgestufte Funktionseinschränkungen	C.3.8	+	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet
5	Künstliche Intelligenz – Fehlerkorrektur	C.3.9	--	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet
6	Dynamische Rekonfiguration	C.3.10	--	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet
7	Modularer Ansatz	Tabelle B.9	++	
8	Verwendung gesicherter/verifizierter Softwareelemente (wenn verfügbar)	C.2.10	++	Bereits existierender Code aus früheren Projekten
9	Vorwärtsverfolgbarkeit von der Spezifikation der Anforderungen an die Sicherheit der Software zur Softwarearchitektur	C.2.11	+	Überprüfen auf Vollständigkeit: Überprüfung, um sicherzustellen, dass alle Anforderungen an die Sicherheit der Software durch die Softwarearchitektur angesprochen werden

Tabelle E.2 *(fortgesetzt)*

Verfahren/Maßnahme		siehe	SIL 2	Interpretation in dieser Anwendung
10	Rückwärtsverfolgbarkeit von der Spezifikation der Anforderungen an die Sicherheit der Software zur Softwarearchitektur	C.2.11	+	Verringern der Komplexität und Funktionalität: Überprüfung, um sicherzustellen, dass alle Anforderungen an die Sicherheit der Software tatsächlich zur Erfüllung der Anforderungen an die Sicherheit des Systems benötigt werden
11a	Strukturierte, schematische Methoden	C.2.1	++	Datenflussmethoden und Datenlogiktabellen können verwendet werden, um zumindest die Architektur des Entwurfs aufzuzeigen
11b	Semiformale Methoden	Tabelle B.7	+	Können für die Schnittstelle zum Leitsystem verwendet werden
11c	Formale Entwurfs- und Verfeinerungsmethoden	B.2.2 C.2.4	+	Selten bei Programmiersprachen mit eingeschränktem Sprachumfang verwendet
11d	Automatische Softwaregenerierung	C.4.6	+	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet
12	Rechnergestützte Spezifikations- und Entwurfswerkzeuge	B.2.4	+	Entwicklungswerkzeuge durch den SPS-Hersteller geliefert
13a	Zyklisches Verhalten, mit garantierter maximaler Zykluszeit	C.3.11	++	Nicht verwendet. Überwachung der SPS-Zykluszeit durch Hardware
13b	Zeitgesteuerte Architektur (en: time triggered architecture TTA)	C.3.11	++	Nicht verwendet. Überwachung der SPS-Zykluszeit durch Hardware
13c	Ereignisgesteuert, mit garantierter maximaler Reaktionszeit	C.3.11	++	Nicht verwendet. Überwachung der SPS-Zykluszeit durch Hardware
14	Statische Ressourcenzuteilung	C.2.6.3	+	Nicht verwendet. Dynamische Ressourcenprobleme fallen bei Programmierung mit eingeschränktem Sprachumfang nicht an
15	Statische Synchronisation des Zugriffs auf gemeinsam genutzte Ressourcen	C.2.6.3	o	Nicht verwendet. Dynamische Ressourcenprobleme fallen bei Programmierung mit eingeschränktem Sprachumfang nicht an

ANMERKUNG 1 In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.

ANMERKUNG 2 Bei einigen der oben genannten Verfahren ist es nicht sinnvoll, sie in einer Programmiersprache mit eingeschränktem Sprachumfang durchzuführen.

Tabelle E.3 – Softwareentwurf und Softwareentwicklung – Werkzeuge und Programmiersprache

(siehe IEC 61508-3, 7.4.4)

Verfahren/Maßnahme		siehe	SIL 2	Interpretation in dieser Anwendung
1	Geeignete Programmiersprache	C.4.5	++	Gewöhnlich Kontaktplan und oft herstellerspezifische Ausprägungen
2	Streng typisierte Programmiersprache	C.4.1	++	Nicht verwendet. Verwendung von SPS-orientiertem strukturierter Text (siehe [16] in den Literaturhinweisen)
3	Sprachenteilmenge	C.4.2	o	Vorsicht vor komplexen „Makro“-Anweisungen und Interrupts, die den SPS-Abtastzyklus ändern usw.

Tabelle E.3 *(fortgesetzt)*

Verfahren/Maßnahme		siehe	SIL 2	Interpretation in dieser Anwendung
4a	Zertifizierte Werkzeuge und zertifizierte Übersetzter	C.4.3	++	Verfügbar von einigen SPS-Herstellern
4b	Werkzeuge und Übersetzter: erhöhtes Vertrauen durch die Anwendung	C.4.4	++	Entwicklungswerkzeug des SPS-Lieferanten, betriebsinterne Werkzeuge entwickelt über mehrere Projekte
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.4 – Softwareentwurf und Softwareentwicklung – Detaillierter Entwurf

(siehe IEC 61508-3, 7.4.5 und 7.4.6)
(einschließlich Software-Systementwurf, Entwurf der Softwaremodule und Kodierung)

Verfahren/Maßnahme		siehe	SIL 2	Interpretation in dieser Anwendung
1a	Strukturierte Methoden	C.2.1	++	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet
1b	Semiformale Methoden	Tabelle B.7	++	Ursache-Folgen-Diagramme, Ablaufdiagramme, Funktionsblöcke. Typisch bei Programmiersprachen mit eingeschränktem Sprachumfang
1c	Formale Entwurfs- und Verfeinerungsmethoden	B.2.2 C.2.4	+	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet
2	Rechnergestützte Entwurfswerkzeuge	B.3.5	+	Entwicklungswerkzeuge durch den SPS-Hersteller geliefert
3	Defensive Programmierung	C.2.5	+	In der Systemsoftware enthalten
4	Modularer Ansatz	Tabelle B.9	++	Das SPS-Programm des Kontaktplans ordnen und gruppieren, um die Modularität mit Bezug zu den erforderlichen Funktionen zu maximieren
5	Entwurfs- und Programmierrichtlinien	C.2.6 Tabelle B.1	++	Betriebsinterne Vereinbarungen zur Dokumentation und Softwarepflege
6	Strukturierte Programmierung	C.2.7	++	In diesem Zusammenhang ähnlich wie bei Modularität
7	Verwendung bewährter/verifizierter Softwareelemente (wenn verfügbar)	C.2.10	++	Funktionsblöcke, Teilprogramme
8	Vorwärtsverfolgbarkeit von der Spezifikation der Anforderungen an die Sicherheit der Software zum Softwareentwurf	C.2.11	+	Überprüfen auf Vollständigkeit: Überprüfung, um sicherzu-stellen, dass alle Anforderungen an die Sicherheit der Software durch den Softwareentwurf angesprochen werden
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.5 – Softwareentwurf und Softwareentwicklung – Test der Softwaremodule und Integration

(siehe IEC 61508-3, 7.4.7 und 7.4.8)

Verfahren/Maßnahme		siehe	SIL 2	Interpretation in dieser Anwendung
1	Statistische Tests	C.5.1	+	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet
2	Dynamische Analyse und Test	B.6.5 Tabelle B.2	++	Verwendet
3	Datenaufzeichnung und Analyse	C.5.2	++	Aufzeichnungen der Testfälle und Ergebnisse
4	Funktionstest und Black-Box-Test	B.5.1 B.5.2 Tabelle B.3	++	Eingangsdaten werden ausgewählt, um alle spezifizierten funktionalen Fälle auszuführen, einschließlich Fehlerbehandlung. Testfälle aus Ursache-Folgen-Diagrammen, Grenzwertanalyse und Partitionierung des Eingangsbereichs
5	Leistungstest	Tabelle B.6	+	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet
6	Modellbasiertes Testen	C.5.27	+	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet
7	Schnittstellentest	C.5.3	+	Im Funktions- und Black-Box-Test enthalten
8	Testmanagement und Automatisierungswerkzeuge	C.4.7	++	Entwicklungswerkzeuge durch den SPS-Hersteller geliefert
9	Vorwärtsverfolgbarkeit von der Spezifikation des Softwareentwurfs zu den Spezifikationen des Modul- und Integrationstests.	C.2.11	+	Überprüfen auf Vollständigkeit: Überprüfung, um sicherzustellen, dass ein angemessener Test geplant ist, um die Funktionalität aller Module und ihre Integration mit geeignet in Beziehung stehenden Modulen zu untersuchen
10	Formale Verifikation	C.5.12	o	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.6 – Integration der programmierbaren Elektronik (Hardware und Software)

(siehe IEC 61508-3, 7.5)

Verfahren/Maßnahme		siehe	SIL 2	Interpretation in dieser Anwendung
1	Funktionstest und Black-Box-Test	B.5.1 B.5.2 Tabelle B.3	++	Eingangsdaten werden ausgewählt, um alle spezifizierten funktionalen Fälle auszuführen, einschließlich Fehlerbehandlung. Testfälle aus Ursache-Folgen-Diagrammen, Grenzwertanalyse und Partitionierung des Eingangsbereichs
2	Leistungstest	Tabelle B.6	+	Wenn das SPS-System für die Werksendprüfung zusammengebaut ist
3	Vorwärtsverfolgbarkeit von den Anforderungen an den System- und Softwareentwurf für die Hardware-/Softwareintegration zu den Spezifikationen der Hardware-/Softwareintegrationstests	C.2.11	+	Überprüfung, um sicherzustellen, dass die Hardware-/Softwareintegrationstests angemessen sind
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.7 – Softwareaspekte bezüglich der Validierung der Sicherheit des Systems

(siehe IEC 61508-3, 7.7)

Verfahren/Maßnahme		siehe	SIL 2	Interpretation in dieser Anwendung
1	Statistische Tests	C.5.1	+	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet
2	Simulation des Prozesses	C.5.18	+	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet, aber immer häufiger bei der Entwicklung von SPS-Systemen
3	Modellbildung	Tabelle B.5	+	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet, aber immer häufiger bei der Entwicklung von SPS-Systemen
4	Funktionstest und Black-Box-Test	B.5.1 B.5.2 Tabelle B.3	++	Eingangsdaten werden ausgewählt, um alle spezifizierten funktionalen Fälle auszuführen, einschließlich Fehlerbehandlung. Testfälle aus Ursache-Folgen-Diagrammen, Grenzwertanalyse und Partitionierung des Eingangsbereichs
5	Vorwärtsverfolgbarkeit von der Spezifikation der Anforderungen an die Sicherheit der Software zum Validierungsplan der Sicherheit der Software	C.2.11	+	Überprüfen auf Vollständigkeit: Überprüfung, um sicherzustellen, dass angemessene Softwarevalidierungstests geplant sind, um die Anforderungen an die Sicherheit der Software anzusprechen
6	Rückwärtsverfolgbarkeit vom Validierungsplan der Sicherheit der Software zur Spezifikation der Anforderungen an die Sicherheit der Software	C.2.11	+	Verringern der Komplexität: Überprüfung, um sicherzustellen, dass alle Validierungstests von Bedeutung sind

ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.

Tabelle E.8 – Softwaremodifikation

(siehe IEC 61508-3, 7.8)

Verfahren/Maßnahme		siehe	SIL 2	Interpretation in dieser Anwendung
1	Einflussanalyse	C.5.23	++	Es wird eine Einflussanalyse durchgeführt, um zu erwägen, inwieweit die Auswirkung der beabsichtigten Änderungen durch die Modularität des Gesamtsystems begrenzt wird
2	Neuverifikation geänderter Softwaremodule	C.5.23	++	Wiederholung früherer Tests
3	Neuverifikation betroffener Softwaremodule	C.5.23	++	Wiederholung früherer Tests
4a	Neuvalidierung des gesamten Systems	Tabelle A.7	+	Die Einflussanalyse hat die Notwendigkeit der Modifikation gezeigt, eine Neuvalidierung wird daher wie gefordert durchgeführt
4b	Validierung durch Regressionstest	C.5.25	++	
5	Software-Konfigurationsmanagement	C.5.24	++	Ausgangspunkte, Aufzeichnungen von Änderungen, Einfluss auf andere Systemanforderungen
6	Datenaufzeichnung und Analyse	C.5.2	++	Aufzeichnungen der Testfälle und Ergebnisse
7	Vorwärtsverfolgbarkeit von der Spezifikation der Anforderungen an die Sicherheit der Software zum Plan der Softwaremodifikation (einschließlich Neuverifikation und Neuvalidierung)	C.2.11	+	Angemessene Modifikationsverfahren, um die Anforderungen an die Sicherheit der Software zu erfüllen
8	Rückwärtsverfolgbarkeit vom Plan der Softwaremodifikation (einschließlich Neuverifikation und Neuvalidierung) zur Spezifikation der Anforderungen an die Sicherheit der Software	C.2.11	+	Angemessene Modifikationsverfahren, um die Anforderungen an die Sicherheit der Software zu erfüllen
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.9 – Softwareverifikation

(siehe IEC 61508-3, 7.9)

Verfahren/Maßnahme		siehe	SIL 2	Interpretation in dieser Anwendung
1	Formaler Beweis	C.5.12	+	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet
2	Animation der Spezifikation und des Entwurfs	C.5.26	+	
3	Statische Analyse	B.6.4 Tabelle B.8	++	Schriftliche Querverweise der Verwendung von Variablen, Bedingungen usw.
4	Dynamische Analyse und Test	B.6.5 Tabelle B.2	++	Automatische Prüfeinrichtung, um wiederholte Tests zu erleichtern
5	Vorwärtsverfolgbarkeit von der Spezifikation des Softwareentwurfs zum Plan der Softwareverifikation (einschließlich Verifikation der Daten)	C.2.11	+	Überprüfen auf Vollständigkeit: Überprüfung, um einen angemessenen Test der Funktionalität sicherzustellen
6	Rückwärtsverfolgbarkeit vom Plan der Softwareverifikation (einschließlich Verifikation der Daten) zur Spezifikation des Softwareentwurfs	C.2.11	+	Verringern der Komplexität: Überprüfung, um sicherzustellen, dass alle Verifikationstests von Bedeutung sind
7	Numerische Analyse offline	C.2.13	+	Nicht verwendet. Die numerische Beständigkeit von Berechnungen spielt in diesem Fall keine besondere Rolle
Test der Softwaremodule und der Integration		Siehe Tabelle E.5 dieser Norm		
Test der Integration der programmierbaren Elektronik		Siehe Tabelle E.6 dieser Norm		
Test des Softwaresystems (Validierung)		Siehe Tabelle E.7 dieser Norm		
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.10 – Beurteilung der funktionalen Sicherheit

(siehe IEC 61508-3, Abschnitt 8)

Beurteilung/Verfahren		siehe	SIL 2	Interpretation in dieser Anwendung
1	Checklisten	B.2.5	+	Verwendet
2	Entscheidungs-/Wahrheitstabellen	C.6.1	+	Verwendet bis zu einem begrenzten Grad
3	Ausfallanalyse	Tabelle B.2	+	Ursache- Folgen-Diagramme auf Systemebene, aber die Ausfall-/Versagensanalyse wird nicht verwendet bei eingeschränkter Programmierung
4	Analyse von Versagensfällen mit gemeinsamer Ursache in diversitärer Software (falls diversitäre Software verwendet wird)	C.6.3	+	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet

Tabelle E.10 *(fortgesetzt)*

Beurteilung/Verfahren		siehe	SIL 2	Interpretation in dieser Anwendung
5	Zuverlässigkeitsblockdiagramm	C.6.4	+	Wird bei Programmiersprachen mit eingeschränktem Sprachumfang nicht verwendet
6	Vorwärtsverfolgbarkeit von den Anforderungen des Abschnitts 8 zum Plan für die Beurteilung der funktionalen Softwaresicherheit	C.2.11	+	Überprüfen der Vollständigkeit der Beurteilung der funktionalen Sicherheit auf Vollständigkeit
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

E.3 Beispiel für den Sicherheits-Integritätslevel 3

Dieses zweite Beispiel ist eine Anwendung mit Abschaltung, die auf einer Hochsprache basiert, und hat den Sicherheits-Integritätslevel 3.

Das Softwaresystem ist im Vergleich zu anderen Sicherheitssystemen groß, mehr als 30 000 Zeilen Source-Code wurden eigens für dieses System entwickelt. Weiter werden die üblichen internen Funktionen einschließlich mindestens zweier diversitärer Betriebssysteme und bereits bestehenden Codes früherer Projekte (Betriebsbewährung) verwendet. Insgesamt würde das entworfene System aus mehr als 100 000 Zeilen Source-Code bestehen, wenn es als solches zur Verfügung stünde.

Die gesamte Hardware (einschließlich Sensoren und Aktoren) ist ein zweikanaliges System, dessen Ausgänge zu den Stellgliedern als logisches UND verbunden sind.

Annahmen:

– obwohl ein schnelle Reaktion nicht erforderlich ist, wird eine höchste Reaktionszeit garantiert;
– es gibt Schnittstellen zu Sensoren, Aktoren und Meldungen an das Bedienungspersonal;
– der Source-Code der Betriebssysteme, der graphischen Routinen und der handelsüblichen mathematischen Routinen ist nicht verfügbar;
– das System wird sehr wahrscheinlich künftig geändert werden;
– die speziell entwickelte Software verwendet eine der gebräuchlichen prozeduralen Sprachen;
– sie ist teilweise objektorientiert;
– alle Teile, deren Source-Code nicht verfügbar ist, wurden diversitär implementiert, die Softwaremodule von verschiedenen Lieferanten bezogen, der Objektcode von diversitären Übersetzern erzeugt;
– die Software läuft auf verschiedenen handelsüblichen Prozessoren, die die Anforderungen der IEC 61508-2 erfüllen;
– alle Anforderungen zur Beherrschung und Vermeidung von Hardwarefehlern nach IEC 61508-2 werden von dem Embedded-System erfüllt und
– die Softwareentwicklung wurde von einer unabhängigen Organisation beurteilt.

ANMERKUNG Zur Definition einer unabhängigen Person siehe IEC 61508-4.

Die folgenden Tabellen zeigen, wie die Tabellen der Anhänge in IEC 61508-3 für diese Anwendung interpretiert werden können.

Tabelle E.11 – Spezifikation der Anforderungen an die Sicherheit der Software

(siehe IEC 61508-3, 7.2)

Verfahren/Maßnahme		siehe	SIL 3	Interpretation in dieser Anwendung
1a	Semiformale Methoden	Tabelle B.7	++	Blockdiagramme, Ablaufdiagramme, endliche Zustandsübergangsdiagramme
1b	Formale Methoden	B.2.2 C.2.4	+	Nur ausnahmsweise
2	Vorwärtsverfolgbarkeit von den Anforderungen an die Sicherheit des Systems zu den Anforderungen an die Sicherheit der Software	C.2.11	++	Überprüfen auf Vollständigkeit: Überprüfung, um sicherzustellen, dass alle Anforderungen an die Sicherheit des Systems durch die Anforderungen an die Sicherheit der Software angesprochen werden
3	Rückwärtsverfolgbarkeit von den Anforderungen an die Sicherheit zu den wahrgenommenen Sicherheitsbedürfnissen	C.2.11	++	Verringern der Komplexität und Funktionalität: Überprüfung, um sicherzustellen, dass alle Anforderungen an die Sicherheit der Software tatsächlich benötigt werden, um die Anforderungen an die Sicherheit des Systems anzusprechen
4	Rechnergestützte Spezifikationswerkzeuge, um die obigen angemessenen Verfahrenen/Maßnahmen zu unterstützen	B.2.4	++	Werkzeuge, die die gewählten Methoden unterstützen
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.12 – Softwareentwurf und Softwareentwicklung – Entwurf der Softwarearchitektur

(siehe IEC 61508-3, 7.4.3)

Verfahren/Maßnahme		siehe	SIL 3	Interpretation in dieser Anwendung
1	Fehlererkennung	C.3.1	++	Verwendet, soweit Sensor-, Aktor- und Datenübertragungsausfälle betroffen und nicht durch die Maßnahmen innerhalb des Embedded-Systems gemäß den Anforderungen der IEC 61508-2 abgedeckt sind
2	Fehlererkennende Codes	C.3.2	+	Nur für externe Datenübertragung
3a	Assertion-Programmierung (en: Failure assertion programming)	C.3.3	+	Die Ergebnisse der Anwendungsfunktionen werden auf Gültigkeit kontrolliert
3b	Diversitäre Überwachungseinrichtungen (mit Unabhängigkeit zwischen der überwachenden und der überwachten Funktion innerhalb des selben Rechners)	C.3.4	+	Nicht bevorzugt: Erhöhte Softwarekomplexität zur Gewährleistung von Unabhängigkeit
3c	Diversitäre Überwachungseinrichtungen (mit Trennung zwischen dem überwachenden und dem überwachten Rechner)	C.3.4	+	Verwendet für einige sicherheitsbezogene Funktionen, wo 3a nicht verwendet wird

Tabelle E.12 *(fortgesetzt)*

Verfahren/Maßnahme		siehe	SIL 3	Interpretation in dieser Anwendung
3d	Diversitäre Redundanz, Implementierung der gleichen Spezifikation der Anforderungen an die Sicherheit der Software	C.3.5	o	Verwendet für einige Funktionen, bei denen der Source-Code nicht verfügbar ist
3e	Funktionale diversitäre Redundanz, Implementierung verschiedener Spezifikation der Anforderungen an die Sicherheit der Software	C.3.5	+	Nicht bevorzugt: Im Wesentlichen durch 3c erreicht
3f	Rückwärtsregeneration	C.3.6	o	Nicht verwendet
3g	Zustandsloser Software-Entwurf (oder Entwurf eingeschränkter Zustände)	C.2.12	+	Nicht verwendet. Eine kontrollierte Abschaltung erfordert Zustände, um Anlagenzustände zu speichern
4a	Regeneration durch Wiederholung	C.3.7	o	Nicht verwendet
4b	Abgestufte Funktionseinschränkungen	C.3.8	++	Ja, aufgrund der Art des technischen Prozesses
5	Künstliche Intelligenz – Fehlerkorrektur	C.3.9	--	Nicht verwendet
6	Dynamische Rekonfiguration	C.3.10	--	Nicht verwendet
7	Modularer Ansatz	Tabelle B.9	++	Wegen der Größe des Systems erforderlich
8	Verwendung gesicherter/verifizierter Softwareelemente (wenn verfügbar)	C.2.10	++	Bereits existierender Code aus früheren Projekten
9	Vorwärtsverfolgbarkeit von der Spezifikation der Anforderungen an die Sicherheit der Software zur Softwarearchitektur	C.2.11	++	Überprüfen auf Vollständigkeit: Überprüfung, um sicherzustellen, dass alle Anforderungen an die Sicherheit der Software durch die Softwarearchitektur angesprochen werden
10	Rückwärtsverfolgbarkeit von der Spezifikation der Anforderungen an die Sicherheit der Software zur Softwarearchitektur	C.2.11	++	Verringern der Komplexität und Funktionalität: Gewährleisten, dass alle Architektur-Sicherheitsanforderungen tatsächlich benötigt werden, um die Anforderungen an die Sicherheit der Software anzusprechen
11a	Strukturierte, schematische Methoden	C.2.1	++	Wegen der Größe des Systems erforderlich
11b	Semiformale Methoden	Tabelle B.7	++	Blockdiagramme, Ablaufdiagramme, Zustands-Übergangsdiagramme
11c	Formale Entwurfs- und Verfeinerungsmethoden	B.2.2 C.2.4	+	Nicht verwendet
11d	Automatische Softwaregenerierung	C.4.6	+	Nicht verwendet. Vermeidung von Unsicherheiten durch Übersetzer/Codegenerator
12	Rechnergestützte Spezifikations- und Entwurfswerkzeuge	B.2.4	++	Werkzeuge, die die gewählte Methode unterstützen
13a	Zyklisches Verhalten, mit garantierter maximaler Zykluszeit	C.3.11	++	Nicht verwendet
13b	Zeitgesteuerte Architektur (en: time triggered architecture TTA)	C.3.11	++	Nicht verwendet
13c	Ereignisgesteuert, mit garantierter maximaler Reaktionszeit	C.3.11	++	Nicht verwendet

Tabelle E.12 *(fortgesetzt)*

Verfahren/Maßnahme		siehe	SIL 3	Interpretation in dieser Anwendung
14	Statische Ressourcenzuteilung	C.2.6.3	++	Nicht verwendet. Auswahl einer Programmiersprache zur Vermeidung von dynamischen Ressourcenproblemen
15	Statische Synchronisation des Zugriffs auf gemeinsam genutzte Ressourcen	C.2.6.3	+	Nicht verwendet. Auswahl einer Programmiersprache zur Vermeidung von dynamischen Ressourcenproblemen
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.13 – Softwareentwurf und Softwareentwicklung – Werkzeuge und Programmiersprache

(siehe IEC 61508-3, 7.4.4)

Verfahren/Maßnahme		siehe	SIL 3	Interpretation in dieser Anwendung
1	Geeignete Programmiersprache	C.4.5	++	Programmiersprache mit vollem Sprachumfang ausgewählt
2	Streng typisierte Programmiersprache	C.4.1	++	Verwendet
3	Sprachenteilmenge	C.4.2	++	Definierte Teilmenge für die ausgewählte Sprache
4a	Zertifizierte Werkzeuge und zertifizierte Übersetzter	C.4.3	++	Nicht verfügbar
4b	Werkzeuge und Übersetzter: erhöhtes Vertrauen durch die Anwendung	C.4.4	++	Verfügbar und verwendet
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.14 – Softwareentwurf und Softwareentwicklung – Detaillierter Entwurf

(siehe IEC 61508-3, 7.4.5 und 7.4.6)
(einschließlich Software-Systementwurf, Entwurf der Softwaremodule und Kodierung)

Verfahren/Maßnahme		siehe	SIL 3	Interpretation in dieser Anwendung
1a	Strukturierte Methoden	C.2.1	++	Weitverbreitete Verwendung, insbesondere SADT und JSD
1b	Semiformale Methoden	Tabelle B.7	++	Endliche Automaten/Zustands-Übergangsdiagramme, Blockdiagramme, Ablaufdiagramme
1c	Formale Entwurfs- und Verfeinerungsmethoden	B.2.2 C.2.4	+	Nur ausnahmsweise, nur für einige Basiskomponenten
2	Rechnergestützte Entwurfswerkzeuge	B.3.5	++	Verwendet für ausgewählte Methoden
3	Defensive Programmierung	C.2.5	++	Alle Maßnahmen werden, wo sie wirksam sind, ausdrücklich in der Anwendersoftware verwendet, abgesehen von denjenigen, die vom Compiler automatisch eingefügt werden
4	Modularer Ansatz	Tabelle B.9	++	Softwaremodule mit beschränkter Größe, Geheimnisprinzip/ Kapselung, ein Eingang und ein Ausgang in Unterprogrammen und Funktionen, vollständig definierte Schnittstellen, ...

Tabelle E.14 *(fortgesetzt)*

Verfahren/Maßnahme		siehe	SIL 3	Interpretation in dieser Anwendung
5	Entwurfs- und Programmierrichtlinien	C.2.6 Tabelle B.1	++	Verwendung einer Programmierrichtlinie, keine dynamischen Objekte, keine dynamischen Variablen, eingeschränkte Verwendung von Interrupts, eingeschränkte Verwendung von Pointern, eingeschränkte Verwendung von Rekursionen, keine unbedingten Sprünge, ...
6	Strukturierte Programmierung	C.2.7	++	Verwendet
7	Verwendung bewährter/verifizierter Softwareelemente (wenn verfügbar)	C.2.10	++	Verfügbar und verwendet
8	Vorwärtsverfolgbarkeit zwischen der Spezifikation der Anforderungen an die Sicherheit der Software und dem Softwareentwurf	C.2.11	++	Überprüfen auf Vollständigkeit: Überprüfung, um sicherzustellen, dass alle Anforderungen an die Sicherheit der Software durch den Softwareentwurf adressiert werden
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.15 – Softwareentwurf und Softwareentwicklung – Test der Softwaremodule und Integration

(siehe IEC 61508-3, 7.4.7 und 7.4.8)

Verfahren/Maßnahme		siehe	SIL 3	Interpretation in dieser Anwendung
1	Statistische Tests	C.5.1	+	Verwendet für Softwaremodule, für die kein Source-Code verfügbar ist und die Festlegung von Grenzwerten und Äquivalenzklassen für Testdaten schwierig ist
2	Dynamische Analyse und Test	B.6.5 Tabelle B.2	++	Verwendet für Softwaremodule, für die der Source-Code verfügbar ist. Testfälle aus Grenzwertanalyse, Modellierung der Leistungsfähigkeit, Äquivalenzklassen und Partitionen des Eingangsbereichs, strukturbasierte Tests
3	Datenaufzeichnung und Analyse	C.5.2	++	Aufzeichnungen der Testfälle und Ergebnisse
4	Funktionstest und Black-Box-Test	B.5.1 B.5.2 Tabelle B.3	++	Verwendet für den Test der Softwaremodule, für die kein Source-Code verfügbar ist, und für den Integrationstest. Eingangsdaten werden ausgewählt, um alle spezifizierten funktionalen Fälle auszuführen, einschließlich Fehlerbehandlung. Testfälle aus Ursache- Folgen-Diagrammen, Prototypenerstellung, Grenzwertanalyse, Äquivalenzklassen und Partitionen des Eingangsbereichs
5	Leistungstest	Tabelle B.6	++	Verwendet während des Integrationstests auf der Zielhardware
6	Modellbasiertes Testen	C.5.27	++	Nicht verwendet
7	Schnittstellentest	C.5.3	++	Im Funktions- und Black-Box-Test enthalten
8	Testmanagement und Automatisierungswerkzeuge	C.4.7	++	Verwendet wo verfügbar

Tabelle E.15 *(fortgesetzt)*

Verfahren/Maßnahme		siehe	SIL 3	Interpretation in dieser Anwendung
9	Vorwärtsverfolgbarkeit von der Spezifikation des Softwareentwurfs zu den Spezifikationen des Modul- und Integrationstests.	C.2.11	++	Überprüfung, um sicherzustellen, dass die Hardware-/Softwareintegrationstests angemessen sind
10	Formale Verifikation	C.5.12	+	Nicht verwendet
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.16 – Integration der programmierbaren Elektronik (Hardware und Software)

(siehe IEC 61508-3, 7.5)

Verfahren/Maßnahme		siehe	SIL 3	Interpretation in dieser Anwendung
1	Funktionstest und Black-Box-Test	B.5.1 B.5.2 Tabelle B.3	++	Verwendet als zusätzlicher Test zum Softwareintegrationstest (siehe Tabelle E.15 oberhalb). Eingangsdaten werden ausgewählt, um alle spezifizierten funktionalen Fälle auszuführen, einschließlich Fehlerbehandlung. Testfälle aus Ursache- Folgen-Diagrammen, Prototypenerstellung, Grenzwertanalyse, Äquivalenzklassen und Partitionen des Eingangsbereichs
2	Leistungstest	Tabelle B.6	++	Umfangreich verwendet
3	Vorwärtsverfolgbarkeit von den Anforderungen an den System- und Softwareentwurf für die Hardware-/Softwareintegration zu den Spezifikationen der Hardware-/Softwareintegrationstests	C.2.11	++	Überprüfung, um sicherzustellen, dass die Hardware-/Softwareintegrationstests angemessen sind
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.17 – Softwareaspekte bezüglich der Validierung der Sicherheit des Systems

(siehe IEC 61508-3, 7.7)

Verfahren/Maßnahme		siehe	SIL 3	Interpretation in dieser Anwendung
1	Statistische Tests	C.5.1	+	Für die Validierung nicht verwendet
2	Simulation des Prozesses	C.5.18	++	Endliche Automaten, Modellierung der Leistungsfähigkeit, Prototypenerstellung und Animation
3	Modellbildung	Tabelle B.5	++	Für die Validierung nicht verwendet
4	Funktionstest und Black-Box-Test	B.5.1 B.5.2 Tabelle B.3	++	Eingangsdaten werden ausgewählt, um alle spezifizierten funktionalen Fälle auszuführen, einschließlich Fehlerbehandlung. Testfälle aus Ursache- Folgen-Diagrammen, Grenzwertanalyse und Partitionierung des Eingangsbereichs

Tabelle E.17 *(fortgesetzt)*

Verfahren/Maßnahme		siehe	SIL 3	Interpretation in dieser Anwendung
5	Vorwärtsverfolgbarkeit von der Spezifikation der Anforderungen an die Sicherheit der Software zum Validierungsplan der Sicherheit der Software	C.2.11	++	Überprüfen auf Vollständigkeit: Überprüfung, um sicherzustellen, dass alle Anforderungen an die Sicherheit der Software durch den Validierungsplan angesprochen werden
6	Rückwärtsverfolgbarkeit vom Validierungsplan der Sicherheit der Software zur Spezifikation der Anforderungen an die Sicherheit der Software	C.2.11	++	Verringern der Komplexität: Überprüfung, um sicherzustellen, dass alle Validierungstests von Bedeutung sind
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.18 – Modifikation

(siehe IEC 61508-3, 7.8)

Verfahren/Maßnahme		siehe	SIL 3	Interpretation in dieser Anwendung
1	Einflussanalyse	C.5.23	++	Verwendet
2	Neuverifikation geänderter Softwaremodule	C.5.23	++	Verwendet
3	Neuverifikation betroffener Softwaremodule	C.5.23	++	Verwendet
4a	Neuvalidierung des gesamten Systems	Tabelle A.7	++	Abhängig vom Ergebnis der Einflussanalyse
4b	Validierung durch Regressionstest	C.5.25	++	Verwendet
5	Software-Konfigurationsmanagement	C.5.24	++	Verwendet
6	Datenaufzeichnung und Analyse	C.5.2	++	Verwendet
7	Vorwärtsverfolgbarkeit von der Spezifikation der Anforderungen an die Sicherheit der Software zum Plan der Softwaremodifikation (einschließlich Neuverifikation und Neuvalidierung)	C.2.11	++	Überprüfen auf Vollständigkeit: Überprüfung, um sicherzustellen, dass die Modifikationsverfahren angemessen sind um die Anforderungen an die Sicherheit der Software zu erreichen
8	Rückwärtsverfolgbarkeit vom Plan der Softwaremodifikation (einschließlich Neuverifikation und Neuvalidierung) zur Spezifikation der Anforderungen an die Sicherheit der Software	C.2.11	++	Verringern der Komplexität: Überprüfung, um sicherzustellen, dass alle Modifikationsverfahren notwendig sind
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.19 – Softwareverifikation

(siehe IEC 61508-3, 7.9)

Verfahren/Maßnahme		siehe	SIL 3	Interpretation in dieser Anwendung
1	Formaler Beweis	C.5.12	+	Nur ausnahmsweise, nur für einige Basisklassen
2	Animation der Spezifikation und des Entwurfs	C.5.26	+	Nicht verwendet
3	Statische Analyse	B.6.4 Tabelle B.8 C.5.14	++	Für jeden neuentwickelten Code. Grenzwertanalyse, Checklisten, Kontrollflussanalyse, Datenflussanalyse, Fagan-Inspektionen, Entwurfsüberprüfungen
4	Dynamische Analyse und Test	B.6.5 Tabelle B.2	++	Für jeden neuentwickelten Code
5	Vorwärtsverfolgbarkeit von der Spezifikation des Softwareentwurfs zum Plan der Softwareverifikation (einschließlich Verifikation der Daten)	C.2.11	++	Überprüfen auf Vollständigkeit: Überprüfung, um sicherzustellen, dass die Modifikationsverfahren für die Anforderungen an die Sicherheit der Software angemessen sind
6	Rückwärtsverfolgbarkeit vom Plan der Softwareverifikation (einschließlich Verifikation der Daten) zur Spezifikation des Softwareentwurfs	C.2.11	++	Verringern der Komplexität: Überprüfung, um sicherzustellen, dass alle Modifikationsverfahren notwendig sind
7	Numerische Analyse offline	C.2.13	++	Nicht verwendet. Auf der numerischen Stabilität der Berechnungen liegt hier kein Hauptaugenmerk
Test der Softwaremodule und der Integration		Siehe Tabelle E.15 dieser Norm		
Test der Integration der programmierbaren Elektronik		Siehe Tabelle E.16 dieser Norm		
Test des Softwaresystems (Validierung)		Siehe Tabelle E.17 dieser Norm		
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Tabelle E.20 – Beurteilung der funktionalen Sicherheit

(siehe IEC 61508-3, Abschnitt 8)

Beurteilung/Verfahren		siehe	SIL 3	Interpretation in dieser Anwendung
1	Checklisten	B.2.5	+	Verwendet
2	Entscheidungs-/Wahrheitstabellen	C.6.1	+	In beschränktem Umfang verwendet
3	Ausfallanalyse	Tabelle B.4	++	Fehlerbaumanalyse (FTA) wird sehr umfassend verwendet und Ursache- Folgen-Diagramme werden in beschränktem Umfang verwendet
4	Analyse von Versagensfällen mit gemeinsamer Ursache in diversitärer Software (falls diversitäre Software verwendet wird)	C.6.3	++	Verwendet
5	Zuverlässigkeitsblockdiagramm	C.6.4	+	Verwendet

Tabelle E.20 *(fortgesetzt)*

Beurteilung/Verfahren		siehe	SIL 3	Interpretation in dieser Anwendung
6	Vorwärtsverfolgbarkeit von den Anforderungen des Abschnitts 8 zum Plan für die Beurteilung der funktionalen Softwaresicherheit	C.2.11	++	Überprüfen der Abdeckung der Beurteilung der funktionalen Sicherheit auf Vollständigkeit
ANMERKUNG In der Spalte „siehe“ beziehen sich die informativen Verweise „B.x.x.x“ und „C.x.x.x“ auf Beschreibungen von Verfahren in IEC 61508-7, Anhänge B und C, wogegen „Tabelle A.x“ und „Tabelle B.x“ sich auf Tabellen mit Verfahren in IEC 61508-3, Anhänge A und B, bezieht.				

Literaturhinweise

[1] IEC 61511 (alle Teile), *Functional safety – Safety instrumented systems for the process industry sector*

ANMERKUNG Harmonisiert in der Reihe EN 61511 (nicht modifiziert).

[2] IEC 62061, *Safety of machinery – Functional safety of safety-related electrical, electronic and programmable electronic control systems*

ANMERKUNG Harmonisiert als EN 62061.

[3] IEC 61800-5-2, *Adjustable speed electrical power drive systems – Part 5-2: Safety requirements – Functional*

ANMERKUNG Harmonisiert als EN 61800-5-2.

Die folgenden Empfehlungen enthalten weitere Einzelheiten zur Ermittlung von Ausfallwahrscheinlichkeiten (siehe Anhang B).

[4] IEC 61078:2006, *Analysis techniques for dependability – Reliability block diagram and boolean methods*

ANMERKUNG Harmonisiert als EN 61078:2006 (nicht modifiziert).

[5] IEC 61165:2006, *Application of Markov techniques*

ANMERKUNG Harmonisiert als EN 61165:2006 (nicht modifiziert).

[6] BS 5760, *Reliability of system equipment and components – Part 2: Guide to assessment of reliability*

[7] D. J. SMITH, *Reliability, maintainability and risk – Practical methods for engineers*, Butterworth-Heinemann, 5th edition, 1997, ISBN 0-7506-3752-8

[8] R. BILLINGTON and R. N. ALLAN, *Reliability evaluation of engineering systems*, Plenum, 1992, ISBN 0-306-44063-6

[9] W. M. GOBLE, *Evaluating control system reliability – Techniques and applications*, Instrument Society of America, 1992, ISBN 1-55617-128-5

Nützliche Empfehlungen für die Berechnung des Diagnosedeckungsgrades (siehe Anhang C) sind zum Beispiel die folgenden.

[10] Reliability Analysis Center (RAC), *Failure Mode/Mechanism Distributions*, 1991, Department of Defense, United States of America, PO Box 4700, 201 Mill Street, Rome, NY 13440-8200, Organization report number: FMD-91, NSN 7540-01-280-5500

[11] ALLESSANDRO BIROLINI, *Qualität und Zuverlässigkeit technischer Systeme, Theorie, Praxis, Management*, Dritte Auflage,1991, Springer-Verlag, Berlin Heidelberg New York, ISBN 3-540-54067-9, 3 Aufl., ISBN 0-387-54067-9 3 ed. (nur in Deutsch erhältlich)

[12] MIL-HDBK-217F, *Military Handbook Reliability prediction of electronic equipment*, 2 December 1991, Department of Defense, United States of America

Die folgenden Empfehlungen bieten nützliche Informationen zu Ausfällen infolge gemeinsamer Ursache (siehe Anhang D).

[13] *Health and Safety Executive Books*, email hsebooks@prolog.uk.com

[14] R. HUMPHREYS, A., PROC., *Assigning a numerical value to the beta factor common-cause evaluation,* Reliability 1987

[15] UPM3.1, *A pragmatic approach to dependent failures assessment for standard systems*, AEA Technology, Report SRDA-R-13, ISBN 085 356 4337, 1996

Die folgende Norm wird in Tabelle E.3 in Bezug genommen.

[16] IEC 61131-3:2003, *Programmable controllers – Part 3: Programming languages*

ANMERKUNG Harmonisiert als EN 61131-3:2003 (nicht modifiziert).

[17] ISA-TR84.00.02-2002 – Parts 1-5, Safety Instrumented Functions (SIF) Safety Integrity Level (SIL) Evaluation Techniques Package.

[18] IEC 61025:2006, *Fault tree analysis (FTA)*

ANMERKUNG Harmonisiert als EN 61025:2007 (nicht modifiziert).

[19] IEC 62551, *Analysis techniques for dependability – Petri Net technique* [1)]

[20] ANIELLO AMENDOLA, kluwer academic publisher, ISPRA 16-19 November 1987, *Advanced seminar on Common Cause Failure Analysis in Probabilistic Safety Assessment,* ISBN 0-7923-0268-0

[21] CORWIN L. ATWOOD, *The Binomial Failure Rate Common Cause Model,* Technometrics May 1986 Vol 28 n°2

[22] A. ARNOLD, A. GRIFFAULT, G. POINT, AND A. RAUZY. *The altarica language and its semantics*. *Fundamenta Informaticae*, 34,pp.109–124, 2000

[23] M. BOITEAU, Y. DUTUIT, A. RAUZY AND J.-P. SIGNORET, The AltaRica *Data-Flow Language in Use: Assessment of Production Availability of a MultiStates System*, *Reliability Engineering and System Safety*, Elsevier, Vol. 91, pp 747-755

[24] A. RAUZY. *Mode automata and their compilation into fault trees. Reliability Engineering and System Safety*, Elsevier 2002, Volume 78, Issue 1, pp 1-12

[25] For PDS method; see <www.sintef.no/pds>; and further background material in: Hokstad, Per; Corneliussen, Kjell Source: Reliability Engineering and System Safety, v 83, n 1, p 111-120, January 2004

[26] IEC 60601 (alle Teile), *Medical electrical equipment*

ANMERKUNG Harmonisiert in der Reihe EN 60601 (teilweise modifiziert).

[27] IEC 61508-1:2010 *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 1: General requirements*

ANMERKUNG Harmonisiert als EN 61508-1:2010 (nicht modifiziert).

[28] IEC 61508-5:2010 *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 5: Examples of methods for the determination of safety integrity levels*

ANMERKUNG Harmonisiert als EN 61508-5:2010 (nicht modifiziert).

---

1) In Beratung.

[29] IEC 61508-7:2010 *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 7: Overview of techniques and measures*

ANMERKUNG Harmonisiert als EN 61508-7:2010 (nicht modifiziert).

Anhang ZA
(normativ)

Normative Verweisungen auf internationale Publikationen mit ihren entsprechenden europäischen Publikationen

Die folgenden zitierten Dokumente sind für die Anwendung dieses Dokuments erforderlich. Bei datierten Verweisungen gilt nur die in Bezug genommene Ausgabe. Bei undatierten Verweisungen gilt die letzte Ausgabe des in Bezug genommenen Dokuments (einschließlich aller Änderungen).

ANMERKUNG Wenn internationale Publikationen durch gemeinsame Abänderungen geändert wurden, durch (mod) angegeben, gelten die entsprechenden EN/HD.

Publikation	Jahr	Titel	EN/HD	Jahr
IEC 61508-2	2010	Functional safety of electrical/electronic/ programmable electronic safety-related systems – Part 2: Requirements for electrical/electronic/ programmable electronic safety-related systems	EN 61508-2	2010
IEC 61508-3	2010	Functional safety of electrical/electronic/ programmable electronic safety-related systems – Part 3: Software requirements	EN 61508-3	2010
IEC 61508-4	2010	Functional safety of electrical/electronic/ programmable electronic safety-related systems – Part 4: Definitions and abbreviations	EN 61508-4	2010