Entwicklung eines Ansatzes zur Analyse der Netzwerktechnologien in sicherheitsrelevanten Leittechniksystemen hinsichtlich Verbreitung und Auswirkung postulierter Fehler

GRS - 377

Gesellschaft für Anlagen-
und Reaktorsicherheit
(GRS) gGmbH

Entwicklung eines Ansatzes zur Analyse der Netzwerktechnologien in sicherheitsrelevanten Leittechniksystemen hinsichtlich Verbreitung und Auswirkung postulierter Fehler

Joachim Herb
Manuela Jopen
Falk Lindner
Ewgenij Piljugin
Pascal Vogt

Juni 2015

Anmerkung:

Das diesem Bericht zugrunde liegende F&E-Vorhaben 3612R01351 wurde im Auftrag des Bundesministeriums für Umwelt, Naturschutz, Bau und Reaktorsicherheit (BMUB) durchgeführt.

Die Arbeiten wurden von der Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) gGmbH ausgeführt. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Auftragnehmer.

Der Bericht gibt die Auffassung und Meinung des Auftragnehmers wieder und muss nicht mit der Meinung des Auftraggebers übereinstimmen.

GRS - 377
ISBN 978-3-944161-58-7

Deskriptoren:

Datenkommunikation, Netzwerk, Protokoll, OSI-Modell, Topologie, digitale Leittechnik, Hardware, Software, Analyse, Ausfallarten, PSA Werkzeuge, Fehlerbaummodellierung, domainspezifische Sprache

Kurzfassung

Sicherheitsrelevante Leittechnik-Funktionen, wie etwa die Steuerung der Sicherheitssysteme, wurden bisher in den Kernkraftwerken weitgehend durch konventionelle leittechnische Einrichtungen (Analogleittechnik) ausgeführt. Seit einigen Jahren werden weltweit, aber auch in Deutschland, leittechnische Systeme und Einrichtungen in den Kernkraftwerken auf der Basis rechnerbasierter Leittechnik modernisiert.

Die Signalverarbeitung rechnerbasierter Leittechnik nutzt sowohl für die interne als auch für die externe Kommunikation moderne Netzwerktechnologien, wobei die Zuverlässigkeit und die Sicherheit der Informationsübertragung und -verarbeitung eine wichtige Rolle spielt. Die nationale und internationale Betriebserfahrung zeigt einen deutlichen Einfluss der Kommunikation eines vernetzten Leittechniksystems auf dessen Zuverlässigkeit.

Die GRS hat im Rahmen des Vorhabens 3612R01351 „Entwicklung eines Ansatzes zur Analyse der Netzwerktechnologien in sicherheitsrelevanten Leittechniksystemen hinsichtlich Verbreitung und Auswirkung postulierter Fehler“ das Ziel verfolgt, die Kompetenzen der GRS auf diesem Gebiet der Netzwerkkommunikation zu verbessern, phänomenologische Untersuchungen zu potentiellen Fehlerquellen und Fehlerfortpflanzungspfaden (Netzwerkfehler) in einem generischen Leittechniksystem durchzuführen sowie methodische Ansätze zur Analyse der Verbreitung und der Auswirkungen postulierter Fehler in typischen Netzwerken zu entwickeln.

Die GRS hat im Rahmen des Vorhabens umfangreiche Recherchen zum Thema „Datenkommunikation in der digitalen Leittechnik“ durchgeführt. Im vorliegenden Bericht werden die Grundlagen der Datenkommunikation rechnerbasierter Leittechnik erfasst. Dazu zählen u.a. Netzwerktopologien, Kommunikationsprotokolle und -standards sowie generische Fehlerausfallarten. Des Weiteren werden im Bericht die Eigenschaften der verschiedenen Analysemethoden und deren Anwendbarkeit zur Zuverlässigkeitsanalyse der Netzwerkkommunikation für die rechnerbasierten vernetzten Leittechniksysteme diskutiert.

Auf der Basis dieser Recherchen zum Stand von Wissenschaft und Technik wurde ein Analyseansatz entwickelt, in dem die spezifischen Eigenschaften der Netzwerkkommunikation und die Aspekte sicherheitstechnischer Bewertung sicherheitsrelevanter di-

gitaler Leittechnik berücksichtigt werden. Hierzu wurden als erster Schritt Kriterien abgeleitet, welche die Modellierung des generischen Leittechniksystems und die Auswahl der auf dieses System angewandten Methodik maßgeblich bestimmen sollen. Die Kriterien orientieren sich an den für Netzwerktechnologien charakteristischen Merkmalen, welche für die Zuverlässigkeitsbewertung bestimmend sind. Dies führte zu dem Ergebnis, dass zunächst ein Modell der Netzwerkkommunikation eines digitalen Leittechniksystems, wie es typischerweise in einem Kernkraftwerk eingesetzt wird, entwickelt werden muss. Anhand dessen können zunächst eine vereinfachte Funktionsausfallanalyse und anschließend eine detaillierte FMEA-Analyse der Hardware und Software erfolgen. Abschließend wird eine Fehlerbaumanalyse durchgeführt.

Im Rahmen dieses Vorhabens wurde das Modell der Netzwerkkommunikation entwickelt und eine erste Analyse durchgeführt. Ergebnis dieser Analyse ist, dass Ausfälle und latente Fehler der Hardware-Komponenten im Kommunikationsnetzwerk dazu führen können, dass entweder keine Datenübertragung stattfindet oder Daten fehlerhaft übertragen werden. Diese Ausfallarten wurden im Fehlerbaum als selbstmeldende oder nicht-selbstmeldende Ausfälle modelliert. Sie können entweder den Ausfall oder die Fehlanregung eines Leittechnik-Signals verursachen.

Für die Quantifizierung der Ausfälle der Netzwerkkommunikation wurden in der Fehlerbaummodellierung zunächst nur die Ausfallraten der Hardware des Netzwerkes eingesetzt. Die Modellierung der Auswirkungen potentieller Softwarefehler wurde zunächst vernachlässigt, weil hierzu ein geeignetes Modell erforderlich ist. Dieser Aspekt soll in einem separaten Projekt zur Entwicklung eines Modellierungsansatzes der durch die Softwarefehler verursachten Ausfälle analysiert werden.

Im Rahmen der Fehlerbaummodellierung hat die GRS zur Unterstützung der Erstellung von strukturgleichen Fehlerbäumen und für die Verbesserung der Modellierung komplexer Netzwerktopologien das Werkzeug „RiskLang“ entwickelt und erprobt.

Bei der Erprobung der Fehlerbaummodellierung der Netzwerkkommunikation wurde festgestellt, dass weiterhin ein Entwicklungsbedarf bei der Methode der Zuverlässigkeitsanalyse hinsichtlich Berücksichtigung potentieller Fehler in der Software und dynamischer Eigenschaften der systemeigenen Fehlertoleranz rechnerbasierter Leittechnik besteht. Hierzu sollen u.a. kohärente Modelle der Fehlerauswirkung (z.B. GVA durch latente Software-Fehler), Sensitivitäts- und Unsicherheitsanalysen durchgeführt werden. Diese Aspekte sollen in zukünftigen Vorhaben der GRS berücksichtigt werden.

Abstract

So far, safety related instrumentation and control (I&C) functions in nuclear power plants, such as controlling of safety systems, were mostly performed by conventional (analog) I&C equipment. For some years now, I&C systems and equipment in nuclear power plants worldwide, but also in Germany, are modernized by computer-based I&C systems.

In signal processing of the computer-based I&C systems, modern network technologies are used both for internal and external communication, whereas the reliability and safety for information transfer and processing plays an important role. National and international operational experience shows a significant influence of communication in a networked I&C system on its reliability.

The aim of the GRS within the project 361R01351 „Development of an approach for an analysis of network technologies in safety related I&C systems in view of distribution and effect of postulated failures" was to improve the expertise in the field of network communication, to investigate phenomenologically potential sources of failures and fault propagation paths (Network failures) in a generic I&C system as well as to develop methodic approaches for analyses of propagation and effect of postulated failures in typical networks.

The GRS conducted extensive research in the field of „Data communication in digital I&C systems". In this report, the basic principles of data communication of computer-based I&C systems are presented. This includes, among other things, network topologies, communication protocols and standards as well as generic failures. Additionally, the properties of different analysis methods and its applicability for reliability analyses of network communication in computer-based I&C systems are discussed.

Based on state of the art evaluation, an analysis approach was developed, which takes into account the specific properties of network communication and assessment aspects for safety related digital I&C systems. For this purpose, criteria were derived in a first step, which determine significantly the modelling of the generic I&C system and the selection of the methodology to be applied to this system. This criteria are oriented to the characteristic network technology properties, which are relevant for a reliability evaluation. This lead to the result, that first a model of typical network communication in a dig-

ital I&C system should be developed. Then, first a simplified failure analysis and afterwards a detailed FMEA-Analysis of hardware and software can be done. Finally, a fault tree analysis can be performed.

Within this project, the network communication model was developed and a first analysis was done. First results of this analysis are that failures and latent faults of hardware components in communication networks can lead to either missing or incorrect data transfer. These failures were modelled in the fault tree either as self-signaling or not self-signaling failures. They can cause a failure or spurious actuation of a I&C signal.

In the fault tree analysis, only failure rates for hardware were initially inserted for the quantification of failures in network communication. The modelling of failure rates of potential software faults were initially neglected, since an appropriate model is necessary. This aspect shall be considered in a separate project.

The GRS developed and tested the tool „RiskLang" to support the generation of structurally similar fault trees and to improve the modelling of complex network topologies.

Within the testing of fault tree modelling of network communication it was found out, that further development is needed for the method of reliability analysis for an appropriate consideration of potential faults in software and of dynamic properties of fault-tolerance measures of computer-based I&C systems. Therefore, i.a. sensitivity and uncertainty analyses (e.g. CCF by latent software faults within coherent failure effect models) shall be performed. These aspects shall be considered in future GRS projects.

Inhaltsverzeichnis

1 Einleitung 1

2 Datenkommunikation digitaler Leittechnik 5

2.1 Grundlagen zu Netzwerktechnologien und Datenkommunikation 5

2.1.1 Definitionen 5

2.1.2 Generische Merkmale der Netzwerkkommunikation 10

2.1.3 Merkmale verschiedener Netzwerktechnologien 12

2.2 Netzwerktechnologien in der digitalen Leittechnik 25

2.2.1 Peripherie 26

2.2.2 Anforderungen 28

2.2.3 Generische Fehlerausfallarten 31

2.2.4 Sicherheitsbussysteme 38

3 Entwicklung methodischer Ansätze zur Analyse potentieller Netzwerkfehler 43

3.1 Vorgehensweise 43

3.2 Kriterien zur Entwicklung einer Methodik für die Bewertung potentieller Netzwerkfehler 44

3.3 Modellierung eines Kommunikationsnetzwerks in einem generischen Leittechniksystem 47

3.3.1 Allgemeine Aspekte 47

3.3.2 Beschreibung des generischen Modells eines digitalen Sicherheitsleittechniksystems 50

3.3.3 Beschreibung der Netzwerktechnologien im Sicherheitsnetzwerk und im sicherheitsrelevanten Netzwerk 53

3.3.4 Beschreibung der Fehlermodellierung im DSLS 62

3.4 Analyse 68

4 Zusammenfassung 71

Literaturverzeichnis 75

Abbildungsverzeichnis **83**

Tabellenverzeichnis **87**

Abkürzungsverzeichnis **89**

A Methoden der Zuverlässigkeits- und Sicherheitsanalyse **93**

A.1 Einleitung 93

A.2 Einführung und Definitionen 93

A.3 Allgemeine Vorgehensweise bei der Zuverlässigkeitsanalyse 96

A.3.1 Zuverlässigkeitsaspekte 99

A.4 Statische Methoden 105

A.4.1 Fehlerbaumanalyse 105

A.4.2 Fehlerart- und Auswirkungsanalyse 106

A.4.3 Risikograph-Methode 107

A.5 Dynamische Methoden 108

A.5.1 Zuverlässigkeitsblockdiagramm 108

A.5.2 Markov-Prozesse und Petri-Netze 109

A.5.3 Dynamic Flowgraph Methodology 110

A.5.4 Monte-Carlo-Simulation 112

A.6 Vergleich der Methoden 115

B Weiterentwicklung der Fehlerbaumanalyse-Methodik zur Modellierung redundanter bzw. vernetzter Systeme **123**

B.1 Grundlagen 123

B.1.1 Fehlerbaummodellierung 123

B.1.2 Software für Fehlerbaummodellierung und –analysen 124

B.2 Referenzsystem 126

B.2.1 Erfassungsrechner 128

B.2.2 Verarbeitungsrechner 129

B.2.3 Voter-Rechner 131

B.3 Fehlerbaumbeschreibungssprache RiskLang 133

B.3.1 Spezifikation von RiskLang durch die GRS 134

B.3.2 Implementierung von RiskLang 137
B.3.3 Automatische Generierung von RiskLang Code 139
B.3.4 Einschränkungen bei der Implementierung von RiskLang 140
B.4 Anwendung von RiskLang für das Referenzsystem 140
B.5 Quantitative Erprobung der Fehlerbaummodellierung mit RiskLang 147
B.6 Schlussfolgerung 150

1 Einleitung

Sicherheitsrelevante Leittechnik-Funktionen, wie etwa die Steuerung der Sicherheitssysteme, wurden bisher in den Kernkraftwerken weitgehend durch konventionelle leittechnische Einrichtungen (Analogleittechnik) ausgeführt. Dennoch werden seit einigen Jahren weltweit, auch in Deutschland, leittechnische Systeme und Einrichtungen in den Kernkraftwerken auf der Basis rechnerbasierter Leittechnik modernisiert. Ursachen für die Durchführung solcher Modernisierungsmaßnahmen sind eine erschwerte Ersatzteilbeschaffung bei den bisher eingesetzten konventionellen leittechnischen Einrichtungen, Kompatibilitätsprobleme der alten Leittechnik mit neuen maschinenbautechnischen Komponenten und die Möglichkeit, durch die Einführung softwarebasierter leittechnischer Einrichtungen Prozessoptimierungen vornehmen zu können. Des Weiteren bietet die digitale Leittechnik Vorteile hinsichtlich Systemerweiterung, Diagnose, Verbesserung der Informationsübertragung, -speicherung und -darstellung.

Im Bereich der betrieblichen Leittechnik kommen in den meisten deutschen Kernkraftwerken bereits softwarebasierte leittechnische Einrichtungen zum Einsatz. Im Bereich der Begrenzungen, die Bestandteil der Sicherheitsleittechnik sind, setzen ebenfalls einige Anlagen softwarebasierte leittechnische Einrichtungen ein. Weitere sicherheitsrelevante Einsatzgebiete digitaler Leittechnik sind Hebezeug- und Brennelementhandhabungseinrichtungen, Brandschutzeinrichtungen, Strahlenschutzüberwachung, elektrische Schutz- und Steuerungseinrichtungen. Diese Einrichtungen erfordern in hohem Maße Signal- und Datenaustausch zwischen einzelnen Komponenten (u. a. Sensoren, Verarbeitungseinrichtungen, Antrieben), wobei die Zuverlässigkeit und die Sicherheit der Informationsübertragung und -verarbeitung eine wichtige Rolle spielt.

Die Kommunikationswege für diesen Signal- und Datenaustausch leittechnischer Einrichtungen und Systeme werden durch Netzwerktopologien und Datenübertragungstechnologien bestimmt, wobei die übergeordneten Leittechnik-Architekturen auf der Basis sicherheitstechnischer Anforderungen aufgebaut werden.

Die Signalverarbeitung der sicherheitsrelevanten softwarebasierten Leittechnik ist vorwiegend redundant aufgebaut und nutzt sowohl für die redundanzinterne als auch für die redundanzübergreifende Kommunikation moderne Netzwerktechnologien (z. B. Industrial Ethernet, Feldbus-Kommunikation mittels Profibus- und HART-Protokollen). Die nationale und internationale Betriebserfahrung zeigt einen deutlichen Einfluss in-

terner Kommunikation eines vernetzen Leittechniksystems auf dessen Zuverlässigkeit (siehe u. a. WLN 2006/05 „Temporäre Störung von Symphony-Baugruppen“ /GRS 06/).

Die Auswirkungen potentieller Fehlereffekte innerhalb und außerhalb von Kommunikationsnetzwerken in der Leittechnik wurden durch die GRS bisher noch nicht systematisch für sicherheitsrelevante Funktionen in Kernkraftwerken untersucht.

Die erforderlichen Sicherheits- und Zuverlässigkeitsnachweise für komplexe softwarebasierte Systeme können auf der Basis der Betriebserfahrungen und der Prüfungen nicht erbracht werden, weil dadurch nicht alle potentiellen Fehler und deren Auswirkungen erfasst werden. Deshalb ist es erforderlich, geeignete Analysemethoden und Modelle zu entwickeln, um die entsprechenden zuverlässigkeits- und sicherheitsrelevanten Systemeigenschaften zu analysieren und zu bewerten.

In den Fachgebieten der Zuverlässigkeitsbewertung komplexer technischer Systeme und der Sicherheitsbewertung sicherheitsrelevanter Systeme und Einrichtungen existiert bereits ein breites Spektrum von analytischen Methoden und Analysewerkzeugen, u. a. Fehlerart- und Ausfalleffektanalyse, Fehler- und Ereignisbaumanalyse, Markov-Prozess und Petri-Netz sowie Monte-Carlo-Simulation.

Alle oben genannten Methoden erfordern, dass ein reales System bzw. eine Einrichtung als ein Modell abgebildet wird. Die Ergebnisse der Analyse hängen in hohem Maße sowohl von der eingesetzten Methode als auch von der Modellierung der Eigenschaften des zu bewertenden Systems (Einrichtung) ab. Des Weiteren ist die Auswahl der Methode und die entsprechende Modellierung durch die Zielstellung der Analyse (z. B. quantitative Bestimmung der Zuverlässigkeit einer Funktion oder qualitative Identifizierung der Fehlerauswirkungen in einem System) bestimmt. Die Eigenschaften der verschiedenen Analysemethoden werden in Anhang A dieses Berichts diskutiert. Auf die Modellierung der vernetzten digitalen Leittechnik wird in Kapitel 3 und in Anhang B eingegangen.

Ziel des Vorhabens ist es, eine phänomenologische Untersuchung zu potentiellen Fehlerquellen und Fehlerfortpflanzungspfaden (Netzwerkfehler) in einem generischen Leittechniksystem durchzuführen sowie methodische Ansätze zur Analyse der Verbreitung und der Auswirkungen postulierter Fehler in typischen Netzwerken zu entwickeln.

Die GRS hat bisher noch keine umfassenden Analysen zu den Auswirkungen potentieller Fehler in sicherheitsrelevanten Netzwerken digitaler Leittechnik vorgenommen. Deshalb sollen in diesem Vorhaben auch die Kompetenzen auf diesem Gebiet entwickelt werden. Hierzu sollen zunächst auf der Basis generischer Informationen die Bewertungsansätze zur Auswirkung von zu postulierenden Fehlern in typischen Netzwerken sicherheitsrelevanter leittechnischer Einrichtungen entwickelt werden. Diese Ansätze sollen künftig systematische Analysen von Fehlerauswirkungen in sicherheitsrelevanten Netzwerken unterstützen.

Die einzelnen Zielsetzungen des Vorhabens umfassen:

- Ermittlung des Stands von Wissenschaft und Technik zu eingesetzten Netzwerktechnologien und zu Methoden für die Analyse von Netzwerkfehlern,

- Entwicklung methodischer Ansätze zur Analyse potentieller Netzwerkfehler, z. B. Schäden in den Übertragungssystemen (Kabel- und Lichtwellenleiter-Verbindungen), Instandhaltungsfehler an der Soft- und Hardware (z. B. Update/Upgrade der Hard- und Software oder Anwendung ungeeigneter Servicetechnik) oder Fehler durch Fremdeinwirkungen (Intrusion, Virus),

- Entwicklung von Ansätzen zur Analyse der Verbreitung von Auswirkungen postulierter Fehler in typischen Netzwerken.

Ein möglicher Ansatz zur Analyse von Netzwerkfehlern ist die Fehlerbaumanalyse. Bei der Modellierung von (hoch-) redundanten bzw. stark vernetzten Systemen, wie z. B. im Modell der Hardware-Ausfälle digitaler Leittechnik, ist jedoch eine sehr große Zahl von komplexen, aber strukturell sich wiederholenden Fehlerbäumen notwendig. Bisher wurden die Fehlerbäume manuell erstellt, was sehr zeitaufwendig und potentiell fehleranfällig ist. Im Rahmen dieses Vorhabens wurde daher die Möglichkeit geschaffen, strukturell identische Fehlerbäume automatisch zu erstellen (s. Anhang B).

2 Datenkommunikation digitaler Leittechnik

2.1 Grundlagen zu Netzwerktechnologien und Datenkommunikation

2.1.1 Definitionen

Verschiedene Begriffe, die im Rahmen dieses Berichts im Zusammenhang mit Netzwerken und Datenkommunikation relevant sind, werden im Folgenden kurz erläutert.

- Bussystem
  Paralleles oder serielles Kommunikationsnetzwerk zur Datenkommunikation zwischen mehreren Teilnehmern (Computer, Messsysteme, Sensoren, Aktoren etc.), bei denen der Datenzugriff mittels eines Zugriffsverfahrens über gesonderte Steuerleitungen erfolgt. Parallele Busse verfügen neben den Datenleitungen meist noch über Steuer- und Handshake-Leitungen sowie über mindestens einen Master, der die Bus-Steuerung übernimmt. Die Teilnehmer werden als Clients bezeichnet. Bei seriellen Bussen ist nicht zwingend ein Master erforderlich.

- Black Channel
  Das Black-Channel-Prinzip erlaubt die Übertragung sicherer und nicht sicherer Prozessdaten über dieselbe Netzwerk- oder Busleitung /ABB 08/. Die Hard- und Software dieses Kommunikationskanals ist im Gegensatz zum „White Channel“ nicht entsprechend des IEC 61508, Teil 2 /IEC 10b/ und 3 /IEC 10c/, spezifiziert. Zur Datenübertragung wird ein Standardprotokoll verwendet. Diesem überlagert existiert ein zusätzliches Sicherheitsprotokoll, welches die Anforderungen der IEC 61508 erfüllt und mit dem sicherheitsrelevante Daten über diese Netzwerk- oder Busleitung übertragen werden können (siehe auch Kapitel 2.2.4 „Sicherheitsbussysteme“).

- Broadcast-Übertragung
  Die von einem Sender verschickten Datenpakete werden von allen Netzwerkteilnehmern empfangen, jedoch nur von dem Empfänger ausgewertet, an den das Paket adressiert ist.

- Datenkommunikationskanal
  Logische Verbindung zwischen zwei Endpunkten eines Datenkommunikationssystems /DKE 11/.

- Datenkommunikationsknoten
  Verbindungspunkt in einem Datenkommunikationsnetzwerk, zu oder von dem Daten über Datenkommunikationskanäle von oder zu anderen Punkten im Netzwerk übertragen werden /DKE 11/.

- Datenkommunikationssystem
  Anordnung von Hardware, Software und Übertragungsmedien zur Datenübermittlung von einer zur anderen Anwendung /DKE 11/.

- Datenkommunikationseinrichtung
  Sammelbegriff für die Medien sowie die modulations- und codeabhängigen Teile einer mit dem Bus verbundenen Einrichtung, die den niedrigeren Bereich der physikalischen Ebene innerhalb der Einrichtung bilden /DKE 11/.

- Domäne
  Abgegrenztes Netzwerk oder Subnetz

- Fehlerdetektion
  Prozess, mit dem das Auftreten von Fehlern erkannt werden kann.

- Fehlereingrenzung
  Prozess zur Isolierung eines Fehlers, um die Ausbreitung seiner Auswirkungen zu verhindern.

- Fehlerlokalisierung
  Bestimmung des Fehlerorts zur Wiederherstellung des betroffenen Systemteils.

- Fehlermaskierung
  Ein vorhandener Fehlerzustand wird durch einen oder mehrere andere Fehlerzustände kompensiert, so dass dieser Fehlerzustand keine Fehlerwirkung hervorruft.

- Fehlertoleranz
  Fähigkeit eines Systems oder einer Komponente, auch bei Auftreten eines Hardware- oder Softwarefehlers den Normalbetrieb aufrecht zu erhalten /ABB 08/.

- Fehlervermeidung
  Technik, die verwendet wird, um das Auftreten von Fehlern zu vermeiden/ABB 08/.

- Frequenzumtastung
  Frequency Shift Keying (FSK): Frequenzmodulation mit zwei Frequenzen. Die eine Frequenz repräsentiert die digitale „Eins“, die andere die digitale „Null“. Beide Fre-

quenzen sind dabei im gleichen Frequenzabstand um eine Trägerfrequenz angeordnet /ITW 15/.

- Grey-Channel
  Kommunikationskanal, bei dem ein Teil der Hard- und Software-Komponenten entsprechend IEC 61508 spezifiziert ist, das geforderte Sicherheitsniveau des angeschlossenen sicherheitsrelevanten Systems jedoch nicht ausreichend erfüllt wird /MIC 05/ (siehe auch Kapitel 2.2.4 „Sicherheitsbussysteme“).

- Halbduplex-Verbindung
  Daten können abwechselnd, aber nicht gleichzeitig, in beide Richtungen fließen /BAL 12/.

- Hamming-Distanz
  Bei zwei, Stelle für Stelle verglichenen Binärwörtern gleicher Länge die Anzahl der Stellen unterschiedlichen Inhalts. Von der Hamming-Distanz hängt ab, ob eine Fehlererkennung oder -korrektur erfolgen kann. Bei einer Hamming-Distanz von 2 ist beispielsweise eine Fehlererkennung aller 1-Bit-Fehler möglich, jedoch keine Fehlerkorrektur. Bei einer Hamming-Distanz von 3 können 1-Bit-Fehler erkannt und behoben werden.

- Handshake
  Verfahren zur Aushandlung der Verbindungsparameter zwischen Sender und Empfänger beim Verbindungsaufbau.

- Kollision
  Eine Kollision tritt bei gleichzeitigem Senden (Zugriff auf ein Übertragungsmedium) von Netzteilnehmern auf, wodurch die Datenpakete unbrauchbar werden.

- Netzwerk
  Zusammenschluss von eigenständigen Rechnersystemen, bzw. allgemein eindeutig identifizierbaren Geräten (engl. *device* oder *node*), mit dem Ziel, unabhängig vom Standort, verfügbare Ressourcen und insbesondere Informationen gemeinsam zu nutzen und eine Kommunikation zu ermöglichen. Ressourcen können z. B. Server, Drucker, Datenbanken, Programme, Prozessrechner und auch Einrichtungen zur Messwerterfassung sein.

- Protokoll
  Vereinbarung über Datenformate, Zeitfolgen und Fehlerkorrektur beim Datenaustausch von Datenkommunikationssystemen.

- Prozessoreinheit

  Ein oder mehrere Prozessorkerne, deren Instruktionen zur Handhabung von netz- oder kommunikationsbezogenen Funktionen in einem spezifischen Datenkommunikationsstandard dienen.

- Robustheit

  Fähigkeit, die geforderten Funktionen während der gesamten Lebensdauer des Systems zuverlässig auszuführen.

- Schnittstelle

  Parallele oder serielle Verbindungsstelle (Interface) zwischen zwei Systemen (z. B. Messwerterfassung und -auswertung) zum direkten uni- oder bidirektionalen Datenaustausch.

- Sicherheits-Integritäts-Level (SIL)

  SIL ist ein quantitatives Maß, welches die maximal zulässige Ausfallrate des Sicherheitssystems festlegt und durch die Norm IEC 61508 /IEC 10a/ definiert wird. Es gibt die SIL-Level 1 bis 4, wobei bei SIL 1 die höchsten und SIL 4 die geringsten Ausfallraten zulässig sind. Das SIL-Level einer Sicherheitsfunktion berechnet sich aus dem Schadensausmaß, der Möglichkeit der Gefahrenabwendung und der Eintrittswahrscheinlichkeit bei Fehlfunktion des Systems /IEC 10a/.

- Simplex-Verbindung

  Daten können nur in eine Richtung übertragen werden. Diese Technik ermöglicht keine Antwort /BAL 12/.

- Summenrahmenverfahren

  Beim Summenrahmenverfahren sind die Steuerungssysteme und alle Teilnehmer in einer logischen Ringtopologie miteinander verbunden /ITW 15/. Es wird ein Rahmen mit Datenslots für jeden Busteilnehmer erstellt. Wie bei einem Schieberegister werden die Daten nacheinander von Teilnehmer zu Teilnehmer durch den Ring geschoben. Die Teilnehmer lesen dabei die Eingangsdaten in „ihrem“ Slot ein und speichern dafür ihre Ausgangsdaten. Durch eine Markierung am Ende des Rahmens („Loopback-Wort“) erkennt der Master die Ankunft am anderen Ende des Rings und somit das Ende eines Zyklus.

- Telegramm
  Bezeichnung für ein Datenpaket (auch Datagramm genannt), d. h. einer geschlossenen Dateneinheit, welche zur Übertragung von Nutzdaten zwischen einem Sender und einem Empfänger dient.

- Token-Passing
  Token (Engl. Marke, Zeichen): Kurznachricht (kleines Datenpaket), welche fortwährend von Station zu Station weitergereicht wird und über die die Sendeberechtigung einer Station festgelegt wird. Beim Token-Passing handelt es sich demnach um ein Medienzugriffsverfahren in Rechnernetzen, bei dem das Senderecht mit einem im Netz kreisenden Token von Station zu Station weitergegeben wird.

- Topologie
  Logischer und physikalischer Aufbau eines Netzwerks.

- Unicast-Übertragung
  Datenübermittlung im Netzwerk, direkt vom Sender zum Empfänger.

- Vollduplex-Verbindung /ITW 15/
  Daten können gleichzeitig in beide Richtungen übertragen werden /ITW 15/.

- Watchdog
  Kombination aus Diagnose- und Ausgabegerät (typischerweise ein Switch) zur Überwachung des korrekten Arbeitsablaufs eines programmierbaren elektronischen Bauteils und zur Ergreifung von Maßnahmen, wenn ein falscher Arbeitsablauf detektiert wird /ABB 08/.

- White Channel
  Kommunikationskanal, der an sicherheitsrelevante Systeme angeschlossen ist und dessen Hard- und Software-Komponenten den Anforderungen in IEC 61508 entsprechen /MIC 05/ (siehe auch Kapitel 2.2.4 „Sicherheitsbussysteme“).

- Zugriffsverfahren
  Regelt, welches Datenendgerät zu welchem Zeitpunkt welche Datenmenge an wen übertragen darf.

- Zyklische Redundanzprüfung
  Jedem Datenblock wird vor Versendung ein CRC-Wert („Cyclic redundancy check“) angefügt. Hierbei handelt es sich um einen nach einem bestimmten Ver-

fahren berechneten Prüfwert, mit dessen Hilfe man eventuelle während der Speicherung bzw. Übertragung aufgetretene Fehler erkennen kann /WIK 15/.

- Zuverlässigkeit
  Bedingte Wahrscheinlichkeit, dass ein System seine Funktionen in einem bestimmten Zeitintervall korrekt ausführt /KUW 09/.

2.1.2 Generische Merkmale der Netzwerkkommunikation

Ein Netzwerk ist ein Zusammenschluss von datentechnischen Systemen zum Zweck der Kommunikation der Systeme untereinander sowie der Bereitstellung von Schnittstellen nach außen. Die Kommunikation der Netzwerkkomponenten erfolgt dabei mithilfe von vorher vereinbarten, standardisierten Protokollen. Die Protokolle definieren die Art und Weise, wie die Nutzdaten („payload“) in Datenpakete[1] verpackt und über das Kommunikationsmedium übertragen werden.

Die Bestandteile eines Netzwerks, d. h. die Netzwerkkomponenten, können sehr verschiedenartige Systeme sein. Es ist daher sinnvoll, diese Komponenten entsprechend ihrer Funktion im Netzwerk in aktive und passive Systeme zu unterscheiden /ZIL 15/. Passive Komponenten sind beispielsweise Kabel, Antennen, Stecker oder Dosen. Mit aktiven Komponenten sind alle beteiligten Systeme gemeint, die tatsächlich Signale und Daten beziehen, verarbeiten oder steuern. Hierzu zählen u. a. Netzwerkkarten, Hubs, Switches, Firewalls sowie Sende- und Empfangsmodule.

Die Klassifizierung von Netzwerken kann nach der räumlichen Ausdehnung eines Netzwerks sowie der verwendeten Netzwerktechnologie erfolgen. Hinsichtlich der räumlichen Ausdehnung werden Netzwerke in folgende Klassen aufgeteilt:

- GAN („Global Area Network“), weltweite Netzwerke, z. B. das Internet,
- WAN („Wide Area Network“), u. a. Landesnetze, z. B. ISDN, Highspeed Standleitungen,
- MAN („Metropolitan Area Network“), u. a. Stadtnetze, z. B. die Datenübertragung über das Kabelfernsehnetz oder das städtische Stromnetz,

[1] Der Begriff Datenpaket und (Daten-)Telegramm wird in diesem Dokument synonym verwendet.

- LAN („Local Area Network“) und Netzwerke im abgegrenzten Umfeld, z. B. Unternehmensnetzwerke,
- PAN („Personal Area Network“). Netzwerk im nahen Umfeld einer Person, z. B. Bluetooth-Netze.

Um beschreiben zu können, welche Voraussetzungen gegeben sein müssen, damit verschiedene Netzwerkkomponenten miteinander kommunizieren können, wurde das OSI-Schichtenmodell (OSI: „Open System Interconnection“) entwickelt. Demnach läuft die Kommunikation folgendermaßen ab: Ein Sender sendet eine Information über eine Anwendung, z. B. ein Email-Programm. Diese Information gelangt dann von dieser Anwendung zur Netzwerkkarte, verlässt den Rechner über ein Übertragungsmedium und erreicht dann über die Netzwerkkarte des Zielrechners die Anwendung des Empfängers. Alle Schritte, die vom Sender bis zum Empfänger gemacht werden müssen, werden während der Übertragung in einem Protokoll festgehalten. Damit die Datenvermittlung funktioniert, muss der Weg, den das Datenpaket gehen soll, eindeutig festgelegt werden und alle Geräte und jede Software, die in diesem Prozess involviert sind, müssen den Ablauf kennen. Diese Normen legt das OSI-Schichtenmodell fest. Es wurde 1983 von der Internationalen Organisation für Normung (ISO) standardisiert /ISO 94/.

Da das Thema der Datenkommunikation sehr komplex ist, wurde das OSI-Schichtenmodell in sieben Schichten unterteilt. Sie sind in Tabelle 2.1 dargestellt. Oftmals wird bei der netzwerkgestützten Kommunikation nicht jede Schicht des OSI-Modells implementiert, indem bestimmte Protokolle die Funktionalität mehrerer Schichten umsetzen.

Um die prinzipiellen Kommunikationsabläufe vereinfacht bzw. realitätsnäher darzustellen, eignet sich daher das DoD-Schichtenmodell („Department of Defense“, ursprünglich genutzt zur Entwicklung des Internet-Vorläufers ARPANET), das auch die Grundlage für das OSI-Modell war. Hier werden nur vier Schichten unterschieden. Beispielsweise implementiert das HTTP-Protokoll („Hypertext Transfer Protokoll“) als anwendungsorientiertes Protokoll die Schichten 5-7 des OSI-Modells, welche im DoD-Schichtenmodell als eine Schicht, die Anwendungsschicht, dargestellt sind /BAL 12/.

Tab. 2.1 Schichtenmodelle nach OSI und DoD /BAL 12/

Schicht	OSI-Schichtenmodell	Schicht	DoD-Schichtenmodell
7 6 5	**Verarbeitungsschicht** Zugriff von Anwendungen auf das Netz **Datendarstellungsschicht** Ermöglichung eines systemunabhängigen Datenaustausch **Sitzungsschicht** Steuerung der logischen Verbindungen zwischen den Teilnehmern	4	**Anwendungsschicht** Definition der eigentlichen anwendungsorientierten Nutzdaten
4	**Transportschicht** Regelung des Transports und Datenflusses	3	**Transportschicht** Regelung des Transports und Datenflusses
3	**Vermittlungsschicht** Schalten von Verbindungen	2	**Vermittlungsschicht** Schalten von Verbindungen
2 1	**Sicherungsschicht** Sicherstellung einer zuverlässigen und fehlerfreien Kommunikation **Bitübertragungsschicht** Physikalische Datenübertragung	1	**Netzzugriff** Sicherstellung einer zuverlässigen und fehlerfreien Kommunikation und physikalische Übertragung

Eine detaillierte Unterscheidung von Netzwerktechnologien, die auf den verschiedenen Schichten des OSI-Modells zum Einsatz kommen, kann man anhand der nachfolgenden Merkmale treffen /ZIL 15/:

- Netzwerktopologie,
- Netzwerkstandard,
- Zugriffs- und Übertragungsverfahren,
- Übertragungsmedium.

Im Folgenden werden diese Merkmale näher erläutert.

2.1.3 Merkmale verschiedener Netzwerktechnologien

2.1.3.1 Netzwerktopologie

Die Topologie beschreibt die Architektur des betrachteten Netzwerks. Dabei kann nach der Art (z. B. nach Zugangsberechtigungen) und Anzahl der Teilnehmer sowie der Organisation bzw. Anordnung ihrer Verbindungen zum Datenaustausch unterschieden werden. Außerdem differenziert man zwischen physikalischer Topologie, welche die

Hardwareverbindungen der Teilnehmer durch Kabel, Leitungen oder Funkstrecken darstellt, und logischer Topologie, welche die Administration bzw. Organisation des Datenflusses zwischen den Teilnehmern regelt.

Sowohl physikalische als auch logische Topologien können in einer Reihe von Grundformen aufgebaut sein, wie z. B.:

- Punkt-zu-Punkt-Topologie,
- Ring-Topologie,
- Stern-Topologie,
- Bus-Topologie.

Dabei kann bei einem Netzwerk die physikalische Topologie von der logischen Topologie abweichen. Die Wahl der Topologie ist maßgeblich für die Ausfallsicherheit eines Teils bzw. des gesamten Netzwerks verantwortlich /BAL 12/. Eine Reihe einfacher Topologien ist in Abbildung 2.1 gezeigt.

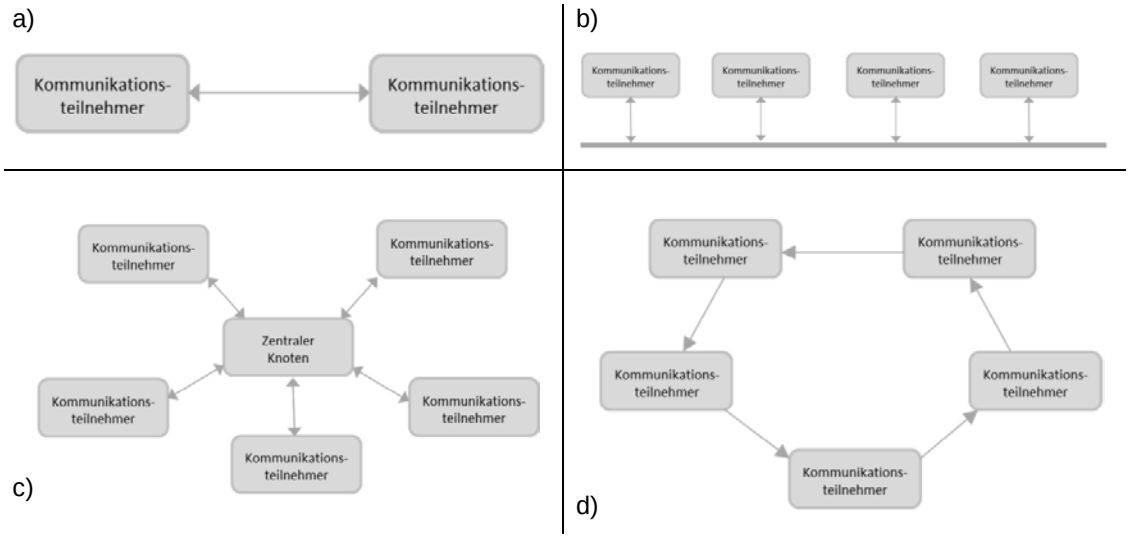


Abb. 2.1 Einfache Topologien für Netzwerke

a) Punkt-zu-Punkt b) Bus c) Stern d) Ring /BAL 12/.

Eine Punkt-zu-Punkt-Verbindung stellt die einfachste Konfiguration einer Kommunikationsverbindung dar (Abbildung 2.1a). Es sind zwei Kommunikationspartner miteinander verbunden. Die Kommunikationsschnittstellen der Partner werden nur für diese Verbindung genutzt. Die Verbindung kann als Simplex, Halb- oder Vollduplex ausgelegt sein /BAL 12/.

Bei der Bus-Topologie sind mehrere Kommunikationsteilnehmer über eine (serieller Bus) oder mehrere (paralleler Bus) gemeinsame Leitungen, den Bus, miteinander verbunden (Abbildung 2.1b). Jeder Teilnehmer kann im Bus-System mit jedem anderen Teilnehmer direkt kommunizieren. Bei Ausfall eines Teilnehmers wird die Kommunikation der anderen nicht beeinflusst. Ein Zugriffsverfahren regelt die Abläufe auf dem Bus. Busse können mit deterministischer Zugriffszuteilung, z B. Master-Slave oder Token-Passing, oder mit stochastischer Zuteilung betrieben werden.

Die Sternverbindung (Abbildung 2.1c) ist dadurch gekennzeichnet, dass der Ausfall des zentralen Knotens die gesamte Kommunikation unterbricht. Bei Ausfall eines Teilnehmers wird die Kommunikation der anderen Teilnehmer (über den zentralen Knoten) nicht beeinträchtigt /BAL 12/. Verbindet man mehrere solcher Sternverbindungen miteinander, erhält man eine Baum-Topologie. Hier sind mehrere Knoten miteinander verbunden. Mit jedem Knoten sind wiederum Teilnehmer verbunden.

Bei der Ring-Topologie (close-loop) erfolgt die Verbindung eines Kommunikationsteilnehmers mit je zwei (benachbarten) Teilnehmern über Punkt-zu-Punkt-Verbindungen (Abbildung 2.1d). Der Datenaustausch der Teilnehmer erfolgt in einer Richtung (z. B. Ring mit festgelegter Sendeumlaufrichtung) oder in beiden Richtungen (Linie/Ring mit Richtungsumschaltung). Die übertragenen Daten werden von Teilnehmer zu Teilnehmer bis zum Empfänger weitergeleitet. Die beteiligten Teilnehmer fungieren dabei als Sender, Empfänger bzw. Repeater (Weiterleiter/Verstärker). Die Wirkung der Teilnehmer als Repeater ermöglicht große Übertragungsentfernungen. Die Steuerung der Kommunikation, z. B. die Richtung der Datenleitung im Ring und die Adressierung, erfolgt über spezielle Protokolle. Um bei Unterbrechung der Ringstruktur durch Ausfall eines Teilnehmers den Ausfall der Kommunikation zu verhindern, kann durch spezielle Maßnahmen die Richtung der Datenleitung im Ring umgekehrt werden. Eine spezielle Form der Ring-Topologie ist der „offene“ Ring. Dabei wird der gebildete Kommunikationsring zwischen zwei benachbarten Kommunikationsteilnehmern nicht verbunden, so dass eine Kommunikationslinie (Open-loop) entsteht. Dies kann physikalisch eine Linien-Topologie, logisch eine Ring-Topologie darstellen bzw. durch die Verwendung eines Ringleitungsverteilers physikalisch eine Stern-Topologie, logisch eine Ring-Topologie bilden /BAL 12/.

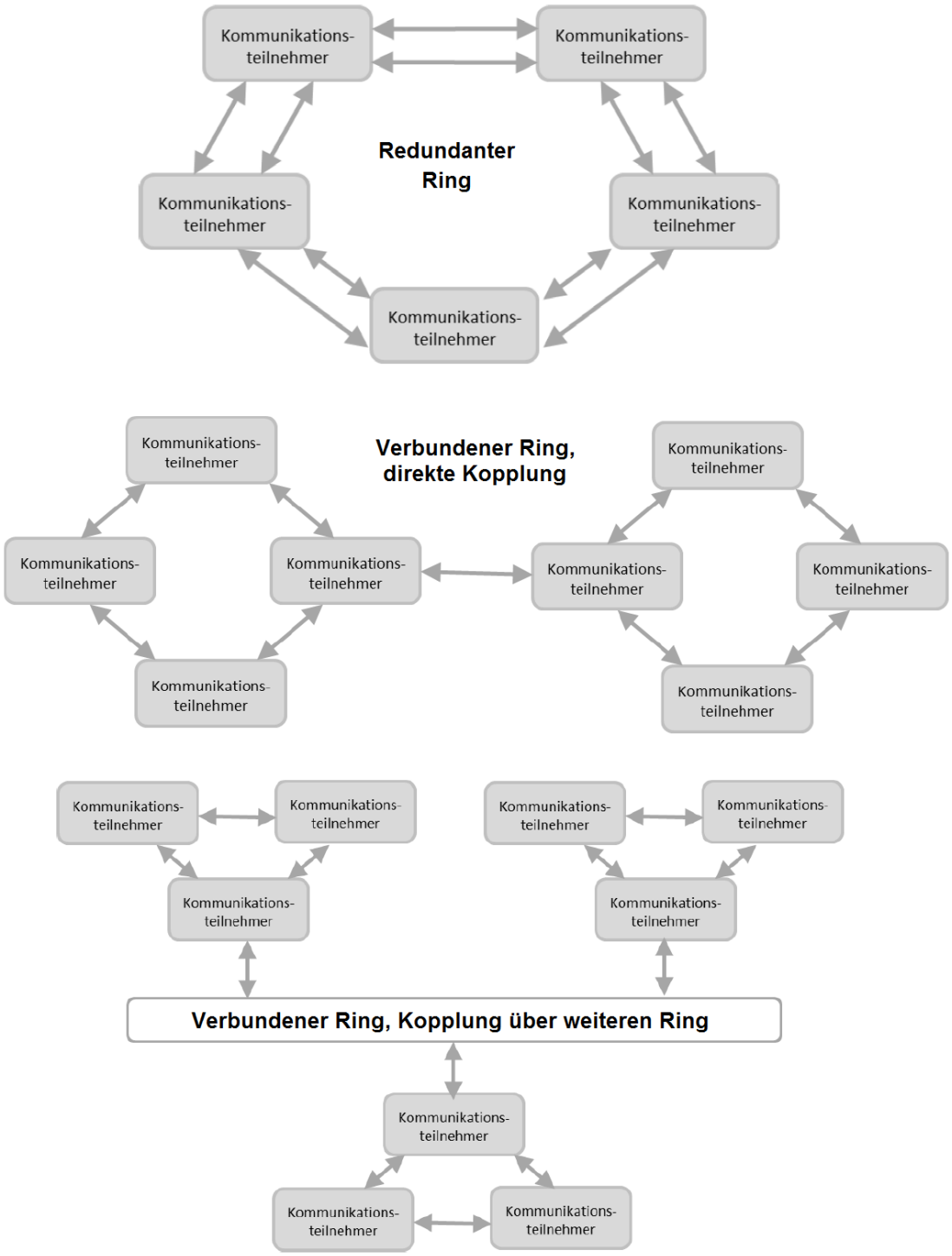


Abb. 2.2 Zusammengesetzte Topologien /BAL 12/

Komplexere Topologien, wie sie in Automatisierungsnetzwerken vorkommen, kann man meist durch Kombination der einfachen Topologien beschreiben. Eine Auswahl solcher Topologien ist in Abbildung 2.2 gezeigt. Bei verbundenen Ring-Topologien ist die Kommunikation innerhalb jedes Rings autark.

Durch die Segmentierung (vorzugshalber nach prozessualen Kriterien) sinkt die Teilnehmeranzahl pro Ring, was eine höhere Datenrate in jedem Ring ermöglicht /BAL 12/.

2.1.3.2 Kommunikationsstandards für lokale Netze und Feldbusse

Kommunikationsstandards bestimmen die Bedingungen und Vorgaben für den Datenaustausch auf den jeweiligen Verbindungsschichten des OSI-Modells. Dabei definieren die Standards nicht zwingend sämtliche Schichten, die für die Kommunikation zwischen den Teilnehmern eines Netzwerkes benötigt werden. So enthält die Reihe des IEEE 802 Standards lediglich Vorgaben für lokale Netze auf Ethernet-Basis zur Umsetzung der Schichten 1 und 2 des OSI-Modells bzw. der Schicht 1 des DoD-Modells. Andere Standards, wie beispielsweise der Profibus-DP Standard /IEC 15/, definieren lediglich die Schichten 1, 2 und 7 des OSI-Modells /BAL 12/.

Im kerntechnischen Bereich sind als betrachtungsrelevante Kommunikationsstandards solche für lokale Netze und Feldbusse relevant. Die grundlegenden Eigenschaften dieser Standards werden im Folgenden dargestellt.

Ethernet-Standard

Ethernet ist eine für LAN konzipierte Netzwerktechnologie und über IEEE 802.3 /IEE 12/ genormt. Ethernet definiert verschiedene Festlegungen für Hardware (verwendete Kabel, Anschlüsse, Signale; Schicht 1 des OSI-Modells) und Software (Ethernet-Datenrahmen mit Datenflusskontrolle; Schicht 2 des OSI-Modells). Funktionalitäten bei der Kommunikation, die auf den Schichten 3-7 des OSI-Modells aufbauen, werden mittels anderer standardisierter oder proprietärer Protokolle bewerkstelligt (z. B. HTTP über TCP/IP) /BAL 12/.

Die Kommunikation über den Ethernet-Standard ist ohne zusätzliche Maßnahmen nicht deterministisch. Trotzdem sind Ethernet-basierte Feldbusse wie PROFINET in Automatisierungsnetzwerken von Bedeutung. Mit Ethernet können in Abhängigkeit von den verwendeten Verbindungselementen die in Tabelle 2.2 genannten physikalischen Topologien realisiert werden /BAL 12/.

Tab. 2.2 Übersicht über Ethernet-Topologien in Abhängigkeit der möglichen passiven Verbindungselemente /BAL 12/

	Punkt-zu-Punkt	**Ring**	**Stern**	**Bus**
Koaxial-Kabel (10BASE5, 10BASE2)	möglich	-	-	Standard
Twisted Pair/LWL – Kabel (10BASE-T, 10BASE-F, 100BASE-TX, 100BASE-FX, ...)	möglich	bedingt möglich	Standard mittels Hub und Switch	-

Feldbus-Standards

Feldbusse werden vor allem zur Verringerung des Verkabelungsaufwands beim Anschluss von Sensoren an leittechnische Geräte eingesetzt. Mit Feldbussen können im Vergleich zur konventionellen Verkabelung der Leittechnik mit den 24-V-Signalen oder Signalen im Bereich von 4-20 mA durchschnittlich 40 % der Kosten eingespart werden /KLA 01/.

Es gibt ca. 50 verschiedene Arten der Feldbus-Kommunikation, die sich hinsichtlich ihrer technischen Funktionen, Einsatzgebiete und Anwendungshäufigkeit voneinander unterscheiden. In der Regel steht hinter jedem Feldbusprotokoll eine Nutzerorganisation. Neben den originären Feldbussen gewinnen echtzeitfähige Ethernet-basierte Feldbusse aufgrund der großen Verbreitung von Ethernet immer mehr an Einfluss für die Feldbus-Kommunikation /BAL 12/.

Bei Feldbussen kann man die Entwicklung beobachten, dass auch hier verstärkt auf die weit verbreitete Ethernet-Technologie zurückgegriffen wird. Dabei stellt sich jedoch im Bereich der Automatisierungstechnik meist die Anforderung der Echtzeitfähigkeit der verwendeten Kommunikationstechnologie. Die IEEE 802.3 Ethernet-Standards definieren einen nicht echtzeitfähigen Ansatz, was durch das verwendete Zugriffsverfahren begründet ist. Deshalb handelt es sich bei Ethernet-basierten Feldbussen um echtzeitfähige Weiterentwicklungen der Ethernet-Technologie.

Feldbusse für industrielle Anwendungen werden international in der IEC 61158-1 /IEC 14a/ definiert, wobei die einzelnen originären Feldbusse in IEC 61784-1 /IEC 15/

und die echtzeitfähigen Ethernet-basierten Feldbusse in IEC 61784-2 /IEC 14b/ geführt werden /BAL 12/.

So heterogen die Protokolllandschaft ist, so heterogen sind auch die mit Feldbussen realisierbaren Netzwerktopologien. In der Kerntechnik sind Feldbusse wie PROFIBUS und Ethernet-basierte Anwendungen wie PROFINET von Bedeutung. Die in Tabelle 2.3 genannten, für kerntechnische Anwendungen relevanten Feldbusse werden im Folgenden betrachtet /BAL 12/.

Tab. 2.3 Relevante Feldbusse in kerntechnischen Anwendungen /BAL 12/

Feldbus	Ethernet-basiert?	Nutzerorganisation
PROFIBUS DP/PA	Nein	PROFIBUS & PROFINET International
PROFINET	Ja	PROFIBUS & PROFINET International
PROFIsafe	Ja/Nein	PROFIBUS & PROFINET International
HART	Nein	HART Communication Foundation

PROFIBUS

PROFIBUS („Process Field Bus“) entstand aus der Absicht der Normierung eines universalen (bitseriellen[2]) Feldbusses. Von PROFIBUS existieren mehrere Varianten. PROFIBUS DP wird für den schnellen, zyklischen Datenaustausch mit Feldgeräten zur Ansteuerung dezentraler Peripherie (siehe Kap. 2.2.1) wie Sensoren und Aktoren eingesetzt. PROFIBUS PA wird für die Prozess-Automation verwendet. Zur Kommunikationssteuerung dient das Buszugriffsprotokoll FDL („Fieldbus Data Link“, Schicht 2 des OSI-Modells). Dabei wird durch die Buszugriffssteuerung MAC („Medium Access Control“, Schicht 2 des OSI-Modells) festgelegt, welcher Kommunikationsteilnehmer zu welchem Zeitpunkt Daten übertragen kann /BAL 12/.

Die Teilnehmer werden hierarchisch in Master und Slaves eingeteilt. Den Master-Teilnehmern wird eine Zugriffsberechtigung innerhalb eines festgelegten Zeitrahmens zugewiesen, in dem sie mit anderen Mastern kommunizieren oder Daten zu Slaves senden bzw. von diesen abholen können. Durch das MAC wird ebenfalls eine Funktionsüberwachung vorgenommen /BAL 12/.

[2] Bei der bitseriellen Übertragung werden alle Bits eines Zeichens nacheinander über eine Datenleitung von Sender zum Empfänger versandt.

Zur physikalischen Datenübertragung (Schicht 1 des OSI-Modells) können elektrische oder optische Verfahren verwendet werden /PRO 10a/.

PROFINET

PROFINET ist ein Industrial-Ethernet Standard und definiert die beiden Ausprägungen PROFINET CBA („Component Based Automation“) und PROFINET IO. Dabei ist PROFINET CBA ein Ansatz zur Maschine-Maschine-Kommunikation und zur Real-Time-Kommunikation. PROFINET IO stellt die Ein-/Ausgabe-Schicht auf Komponenten der dezentralen Peripherie dar. Es wird zur Real-Time- und isochronen Real-Time-Kommunikation von zyklischen Prozessdaten verwendet. PROFINET CBA und PROFINET IO können sowohl separat als auch parallel in einem Netz betrieben werden /PRO 11/.

PROFINET basiert in der Netzzugriffsschicht auf Ethernet und in der Vermittlungs- und Transportschicht sowie in Teilen der Anwenderschicht auf Standardprotokollen aus der Internetprotokollfamilie (TCP(UDP)/IP, RPC, ...). Zusätzliche proprietäre Funktionalitäten setzen lediglich auf der Anwenderschicht auf. Als mögliche Topologien können die Linien-, Bus-, Stern- und Ringtopologie eingesetzt werden /BAL 12/.

PROFISafe

PROFISafe ist eine Kommunikationserweiterung für das PROFIBUS- und PROFINET-Protokoll /PRO 10b/. Dabei handelt es sich um einen zusätzlichen PROFIsafe-Layer über der Standard-PROFIBUS/PROFINET-Schicht 7 des OSI-Modells. Über diesen Layer werden folgende Sicherheitsfunktionen implementiert /BAL 12/:

- Fortlaufende Nummerierung von F[3]-Nachrichten („Sign-of-life“)
- Zeiterwartung mit Quittierung („Watchdog“)
- Kennung zwischen Sender und Empfänger („F-Adresse“)
- Datenintegritätsprüfung (CRC = Cyclic Redundancy Check)

PROFIsafe arbeitet mit einem Master/Slave-Verfahren nach dem Polling-Prinzip. Die Zentrale (Master) fragt die Busteilnehmer (Slaves) in festgelegter Reihenfolge zyklisch

[3] „F“ steht in diesem Zusammenhang für „Fail-Safe“, „Funktionale Sicherheit“ oder „sicherheitsgerichtet“ /PRO 10b/.

nacheinander ab. Eine direkte Sensor-Aktor-Kommunikation ist aufgrund der zentralen Kommunikationsstruktur nicht möglich. Die Verwaltung des Bussystems obliegt der Zentrale. Die Identifizierung der Busteilnehmer erfolgt anhand der konfigurierbaren Busadressen der Busteilnehmer. Da der Ablauf der Busteilnehmerabfrage bekannt ist, lässt sich die Reaktionszeit im fehlerfreien Fall exakt berechnen (Echtzeitfähigkeit). Für den Fehlerfall können Worst-Case-Abschätzungen vorgenommen werden. Im Extremfall entspricht die Fehlererkennungszeit einem kompletten Abfrageumlauf, da der Ausfall eines Busteilnehmers bei Polling-Betrieb spätestens bei der nächsten Abfrage sicher erkannt wird. Der Ausfall der Zentrale bedeutet den vollständigen Ausfall der Buskommunikation. In diesem Fall erkennen die Busteilnehmer den Ausfall des Busses und wechseln in den als solchen definierten Fail-Safe-Zustand /MIC 05/.

Als fertige Automatisierungssysteme, die PROFIsafe für die Buskommunikation zwischen Zentraleinheit und Ein-/Ausgabebaugruppen einsetzt, werden von der Firma Siemens AG die beiden Systeme SIMATIC S7 Distributed Safety und SIMATIC S7 F/FH Systems angeboten. Während das System SIMATIC S7 Distributed Safety den Mischbetrieb betrieblicher und sicherheitsgerichteter Anwendungen auf einem gemeinsamen Bus zulässt, wird SIMATIC S7 F/FH Systems für reine Sicherheitsfunktionen eingesetzt. In der Ausführung FH wird das System zur Erhöhung der Verfügbarkeit redundant aufgebaut (siehe Abbildung 2.3) /MIC 05/.

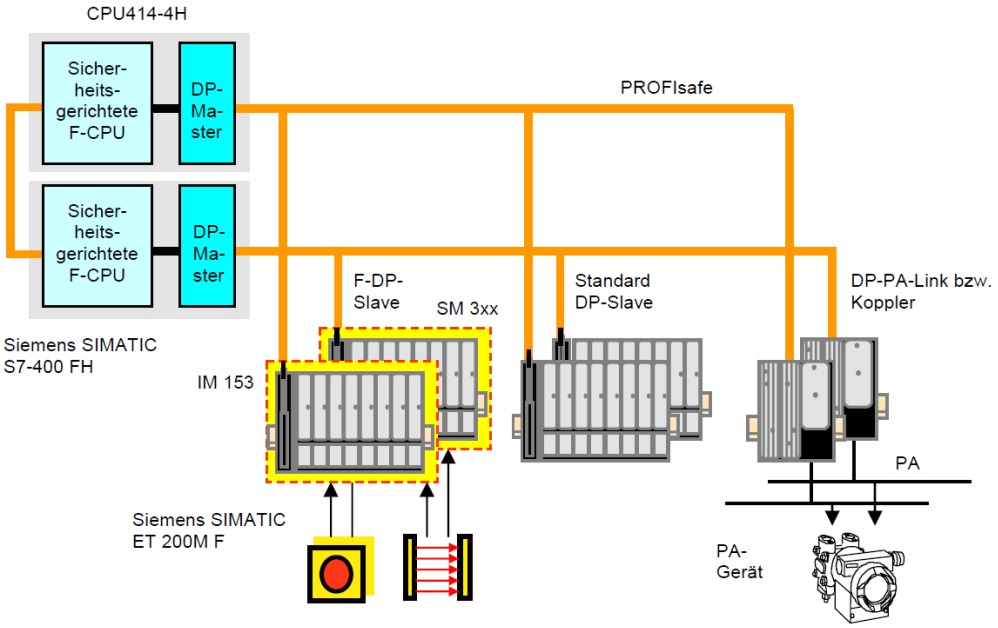


Abb. 2.3 Systemaufbau SIMATIC S7 FH /MIC 05/

Die Ein-/Ausgabebaugruppen werden über einen Rückwandbus zur sog. „dezentralen Peripherie“ zusammengefasst (ET 200M). Die vom Hersteller als „fehlersicher“ bezeichneten Ein-/Ausgabebaugruppen sind mit zwei redundanten Mikrocontrollern ausgestattet, die sich gegenseitig überwachen. Mit der Buskoppelbaugruppe IM 153 erfolgt die Anbindung an die Zentrale über PROFIsafe. Die Zentrale besteht aus einer vom Hersteller als „sicherheitsgerichtet“ bezeichneten CPU, auf der das Automatisierungsprogramm abläuft, und einem DP-Master als Buskoppler. Sowohl Master, dezentrale Peripherie als auch die Busverbindung können redundant aufgebaut werden /MIC 05/.

HART

HART (Highway Addressable Remote Transducer) ist ein Kommunikationsstandard zum Aufbau industrieller Feldbusse. Er ermöglicht die Kommunikation mehrerer Teilnehmer (Feldgeräte, also Sensoren oder Aktuatoren) über einen gemeinsamen Datenbus /BAL 12/.

Dabei wird auf den bereits vorhandenen analogen Signalleitungen für niederfrequente analoge Signale mittels Frequency Shift Keying (FSK) gemäß dem Standard Bell 202 [4] ein hochfrequentes digitales Signal übertragen. Mittels HART können Prozess- und Diagnoseinformationen sowie Steuersignale zwischen den Feldgeräten und Automatisierungssystemen ausgetauscht werden. Dabei implementiert HART verschiedene Protokolle auf mehreren Ebenen im OSI-Modell. Die Kommunikation basiert auf dem Master-Slave-Prinzip /HAR 15/.

2.1.3.3 Zugriffs- und Übertragungsverfahren

Im industriellen Bereich wird eine große Vielfalt von Buszugriffsverfahren eingesetzt. Abbildung 2.4 zeigt eine Übersicht der wichtigsten Verfahren.

Grundsätzlich kann zwischen den Buszugriffsverfahren mit deterministischem und stochastischem Zugriff unterschieden werden. Beim deterministischen Zugriffssystem ist der Ablauf der Buskommunikation fest vorgegeben. Ein Master hat die Oberhand

[4] Beim Bell 202 Standard erfolgt die Datenübertragung über FSK. Dabei wird einem niederfrequenten analogen Signal eine hochfrequente Schwingung überlagert. Eine digitale „1“ wird hierbei mit der Frequenz 1200 Hz und eine „0“ mit der Frequenz 2200 Hz dargestellt.

über die Buskommunikation. Die Zuteilung des Busses erfolgt in immer gleichen Abständen in einer vorgegebenen Reihenfolge an die Busteilnehmer, d. h. der Buszugriff ist zyklisch und deterministisch. Die Zuteilung kann hierbei zentral oder dezentral organisiert sein /MIC 05/.

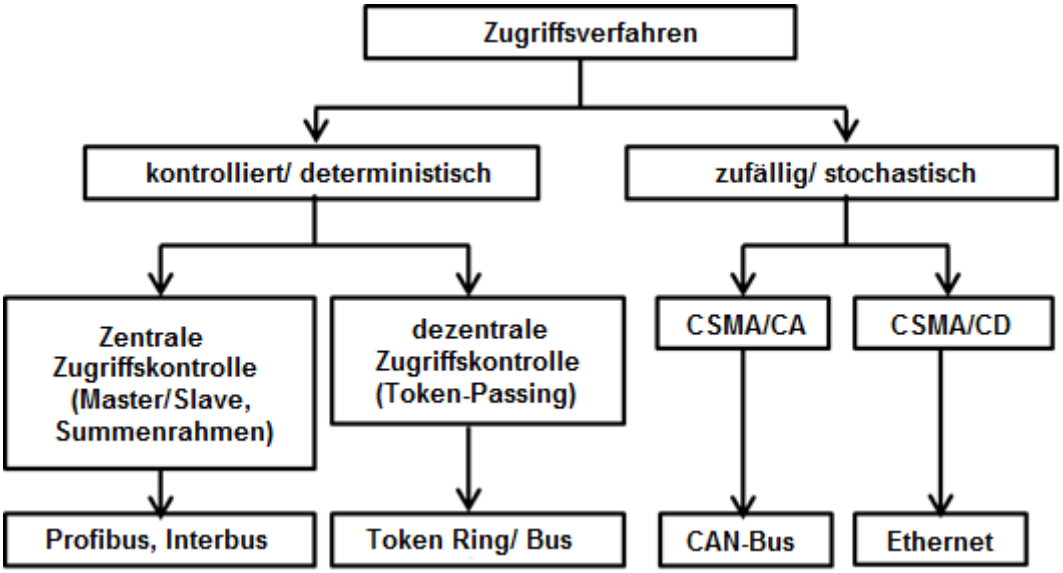


Abb. 2.4 Übersicht der wichtigsten Buszugriffsverfahren /MIC 05/

Bei zentral verwalteten Bussen mit Master/Slave-Verfahren (z. B. PROFIsafe) oder Summenrahmen-Verfahren (z. B. Interbus) gibt es einen Master, der den Busteilnehmern das Senderecht explizit erteilt. Dieses Verfahren ermöglicht einfache Businterfaces bei den Busteilnehmern, da die Intelligenz des Bussystems im Master konzentriert ist. Er teilt den Busteilnehmern nacheinander den Bus für eine definierte Zeit zu. Dazu sendet er ein Telegramm an einen der Busteilnehmer und fordert eine Nachricht bzw. eine Empfangsbestätigung von ihm.

Die Busteilnehmer werden in fester Reihenfolge zyklisch nacheinander abgefragt. Fällt ein Busteilnehmer aus, so erkennt dies der Master und überspringt ihn beim nächsten Mal und der Bus kann ohne Einschränkung weiter betrieben werden. Das Buszugriffsverfahren ist prinzipiell echtzeitfähig, da jeder Busteilnehmer in festgelegter Weise Zugriff auf den Bus bekommt. Eine direkte Kommunikation zwischen den Busteilnehmern ist nicht möglich. Fällt der Master aus, kommt die gesamte Buskommunikation zum Erliegen /MIC 05/.

Dezentrale Zugriffsverfahren arbeiten nach dem Token-Prinzip (z. B. Token-Ring). Dieses Vorgehen gleicht einem Staffellauf, bei dem das Token von einem Busteilnehmer

zum nächsten weitergereicht wird. Nur der Busteilnehmer, der im Besitz des Tokens ist, hat das Senderecht. Ein Master bringt das Token in Umlauf und überwacht den Bus z. B. auf ein verloren gegangenes Token. Fällt ein Busteilnehmer aus, so schließt der Master ihn vom Token-Umlauf aus und der Bus kann weiter betrieben werden. Die maximale Token-Umlaufzeit, und damit die maximale Reaktionszeit des Feldbussystems, ist bekannt. Die einzelnen Busteilnehmer des Bussystems können direkt miteinander kommunizieren, sobald einer von beiden im Besitz des Tokens ist. Ändert sich die Teilnehmeranzahl, muss der Bus jedes Mal neu konfiguriert werden. Mit wachsender Teilnehmerzahl erhöht sich die maximale Token-Umlaufzeit /MIC 05/.

Bei stochastischen Buszugriffsverfahren sind die Busteilnehmer zu jeder Zeit gleichberechtigt. Die Busteilnehmer legen ihre Telegramme ereignisgesteuert auf den Bus. Naturgemäß kann es dabei zu Kollisionen mit Telegrammen anderer Busteilnehmer kommen. In diesem Fall muss der Sendeversuch zumindest eines Busteilnehmers zu einem späteren Zeitpunkt wiederholt werden. Mit einer Erhöhung der Auslastung des Busses steigt die Wahrscheinlichkeit von Kollisionen. Das Buszugriffsverfahren ist damit vom zugrunde liegenden Prinzip her nicht deterministisch. Bei den stochastischen Buszugriffsverfahren wird zwischen dem CSMA/CD- und dem CSMA/CA-Verfahren unterschieden /MIC 05/.

Das CSMA/CD-Verfahren („Carrier Sense Multiple Access/Collision Detection“, z. B. Ethernet) ist in der Lage, aufgetretene Kollisionen auf dem Bus zu erkennen. Beide Busteilnehmer brechen in diesem Fall ihre Übertragung ab. Nach einer zufällig gewählten Zeit versuchen es beide Busteilnehmer erneut. Bei diesem Verfahren können Busteilnehmer jederzeit hinzugefügt oder entfernt werden, ohne dass der Bus neu konfiguriert werden muss. Greifen allerdings zu viele Busteilnehmer auf den Bus zu, sinkt die Effizienz des Buszugriffsverfahrens erheblich. Da die Kollisionen nicht vorhersagbar sind und die Zeitspanne zwischen einer Telegrammkollision und der erneuten Sendung der Telegramme zufällig ist, können die maximalen Signalzeiten nur geschätzt werden /MIC 05/.

Im Unterschied zum CSMA/CD-Verfahren arbeitet das CSMA/CA-Verfahren („Carrier Sense Multiple Access/Collision Avoidance“, z. B. CAN-Bus) auf Basis der Kollisionsvermeidung. Wollen mehrere Busteilnehmer zufällig zum gleichen Zeitpunkt senden, erfolgt zu Beginn mit dem parallelen, synchronen Senden der Telegrammheader ein Vergleich der Prioritäten der Telegramme. Die Priorität ist von dem absendenden Busteilnehmer abhängig und erlaubt eine eindeutige Rangfolge der Telegramme. Bei die-

sem als Arbitrierung bezeichneten Vorgang setzt sich letztendlich das Telegramm mit der höchsten Priorität durch und seine Übermittlung bleibt ungestört. Die unterlegenen Busteilnehmer beginnen anschließend einen erneuten Sendeversuch. Die Kommunikation läuft demnach zwar ereignisgesteuert ab, aber die einzelnen Busteilnehmer können mittels Prioritäten hinsichtlich ihrer Bedeutung gewichtet werden /MIC 05/.

2.1.3.4 Übertragungsmedium

Daten können sowohl leitungsgebunden als auch nicht-leitungsgebunden (leitungslos, wireless) mittels elektromagnetischer oder optischer Signale wie folgt übertragen werden /MIC 05/:

- Elektromagnetische leitungsgebundene Übertragung:
  - analoge oder digitale Signale (Signalpegel im Kleinspannungs- (<50 V AC oder 120 V DC) und/oder mA-Bereich), z. B. Twisted-Pair-Kabel, Flachbandkabel, Backplane/Busplatinen, eingesetzt in diversen leittechnischen Anwendungen,
  - als analoge oder digitale modulierte Hochfrequenz-Signale z. B. auf Koaxialkabel, Hohlleiter, eingesetzt in diversen leittechnischen Anwendungen,
  - als analoge oder digitale modulierte Signale z. B. auf Mittel- oder Hochspannungsleitungen des Stromnetzes (Powerline Communication, PLC), eingesetzt, z. B. zum Datenaustausch mit dezentralen Einrichtungen von Energieversorgungsunternehmen.
- Elektromagnetische nicht-leitungsgebundene Übertragung:
  - als analoge oder digitale Funksignale (kHz – GHz-Bereich), eingesetzt z. B. zur Steuerung von Kranen/Lademaschinen in Kraftwerken.
- Optische leitungsgebundene Übertragung:
  - als digitale optische Signale (erzeugt durch LED oder Laserdioden) auf Lichtwellenleiter (LWL) bzw. Lichtleitkabel (LLK), eingesetzt in diversen leittechnischen Anwendungen
- Optische nicht-leitungsgebundene Übertragung:
  - als digitale optische Signale per Laserlink (optische Freiraumdatenübertragung/Free-Space Optics, FSO), eingesetzt in diversen datentechnischen bzw. nachrichtentechnischen Anwendungen.

2.2 Netzwerktechnologien in der digitalen Leittechnik

Ein digitales Leittechniksystem im Kernkraftwerk erfasst und generiert kontinuierlich eine große Menge von unterschiedlichen Daten, die nach deren Erfassung durch unterschiedliche Kommunikationspfade über das Netzwerk den einzelnen Prozessoren zur Verfügung gestellt und dort zu unterschiedlichen Zwecken verarbeitet werden /IAE 11/, /KOR 09/. Nach der Verarbeitung werden die so erzeugten Ausgabedaten über das Netzwerk weitergegeben. Somit werden vom Leittechniksystem verschiedene Aufgaben unter Nutzung von Netzwerkfunktionen wahrgenommen. Hierzu gehören u. a.:

- die Ausführung von Leittechnikfunktionen, wie z. B. Ansteuerung von Antrieben und Relais, Anzeige und Speicherung von Parametern,
- die Durchführung von Selbstüberwachungsroutinen,
- Validierung und Verifizierung von Daten redundanter leittechnischer Einrichtungen.

Ein Leittechniksystem kann unterschiedliche Netzwerktopologien und Netzwerkprotokolle nutzen, um verschiedene Leittechnikfunktionen zu implementieren. Beispielsweise kann die Netzwerkkommunikation im Bereich der digitalen Sicherheitsleittechnik folgende Kommunikationswege umfassen /KIS 07/:

- Kommunikation innerhalb einer Redundanz des Sicherheitssystems,
- Kommunikation zwischen den Redundanzen der Sicherheitssysteme,
- Kommunikation zwischen sicherheitsrelevanten und betrieblichen Leittechniksystemen,
- Kommunikation mit temporären Datenquellen, wie Wartungs- bzw. Servicegeräten.

Üblicherweise ist ein rechnerbasiertes Automatisierungssystem in drei Hierarchieebenen unterteilt, die Leitebene, die Logikebene und den Feldbereich. Die Kommunikation der in den Prozess eingebundenen Sensoren und Aktoren untereinander und mit der Logikebene erfolgt auf der untersten Hierarchieebene, dem Feldbereich. Dabei gibt es verschiedene Wege, die Netzwerkkommunikation zwischen den Teilnehmern aufzubauen. Auf diese wird im Folgenden eingegangen.

2.2.1 Peripherie

Traditionelle, fest verdrahtete Automatisierungssysteme stellen die Kommunikation der räumlich getrennten Logikebene mit dem Feldbereich über elektrische Einzelverbindungen her (siehe Abbildung 2.5). Die Automatisierungsstruktur ist durch einen zentralisierten Aufbau gekennzeichnet. Bei der Auslegung der Kommunikationsverbindung werden Sensoren und Aktoren über getrennte Ein- bzw. Ausgabemodule an die Logikebene angekoppelt /MIC 05/.

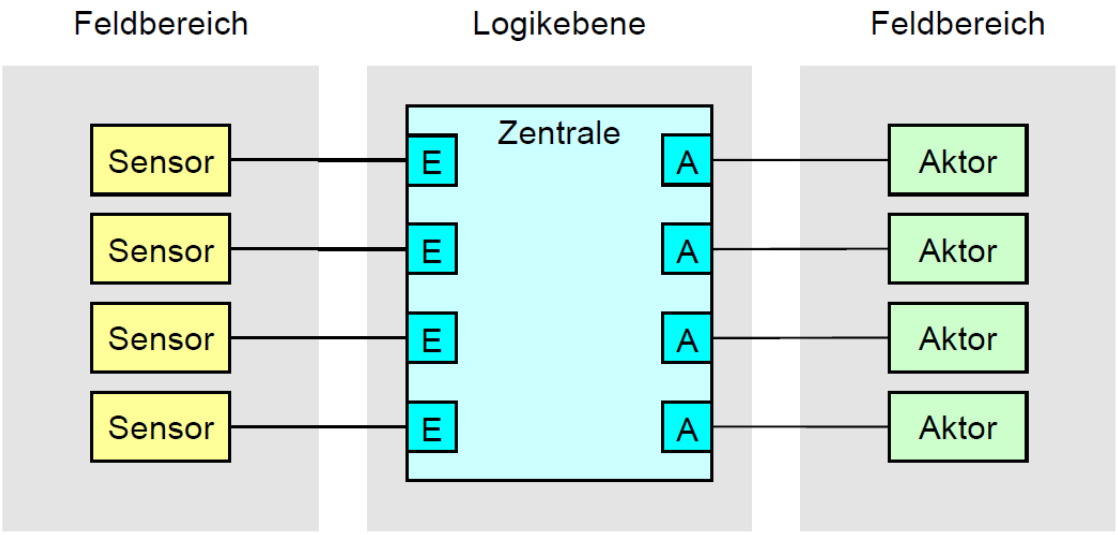


Abb. 2.5 Bisherige Sensor-Aktor-Anbindung

Moderne industrielle Automatisierungsstrukturen verfolgen den Trend zur Dezentralisierung. Über Bussysteme werden die Ein-/Ausgabebaugruppen quasi in den Feldbereich, in die sog. dezentrale Peripherie, verlagert (siehe Abbildung 2.6). Die Anbindung der Peripherie an die Zentrale über Bussysteme erfolgt vornehmlich über sogenannte Busklemmen. Diese sind individuell konfigurierbar und können sowohl Ein- als auch Ausgabebaugruppen aufnehmen. Eine Strukturierung gemäß den Redundanzen ist für den kerntechnischen Einsatz hierbei erforderlich /MIC 05/.

Eine weitergehende Ausführungsform, die bereits in der Industrie eingesetzt wird, ist eine Systemarchitektur, die keine Zentrale mehr aufweist. Die Intelligenz ist dezentral im Feld verteilt und die Sensoren und Aktoren kommunizieren direkt untereinander über einen gemeinsamen Bus (siehe Abbildung 2.7).

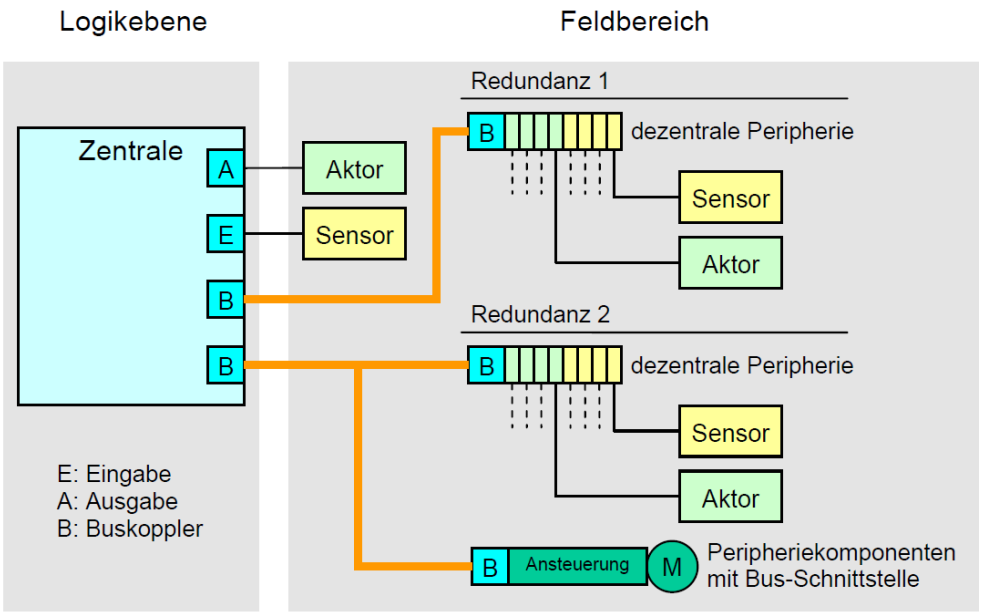


Abb. 2.6 Sensor-Aktor-Anbindung über Bussysteme (Orange) an die Zentrale

Diese Technik ist besonders sinnvoll in Bereichen, die geringe Logikkomplexität aufweisen. Die Reaktionszeiten können durch einen dezentralen Aufbau stark reduziert werden. Für den Einsatz bei sicherheitskritischen Aufgaben kann ein sogenannter Sicherheitsmonitor als zentrale Überwachungsinstanz dem Netz hinzugefügt werden. Sobald er einen Komponentenausfall oder einen Kommunikationsfehler auf dem Bus erkennt, initiiert er definierte Fehlerreaktionen über fest verdrahtete Ausgänge /MIC 05/.

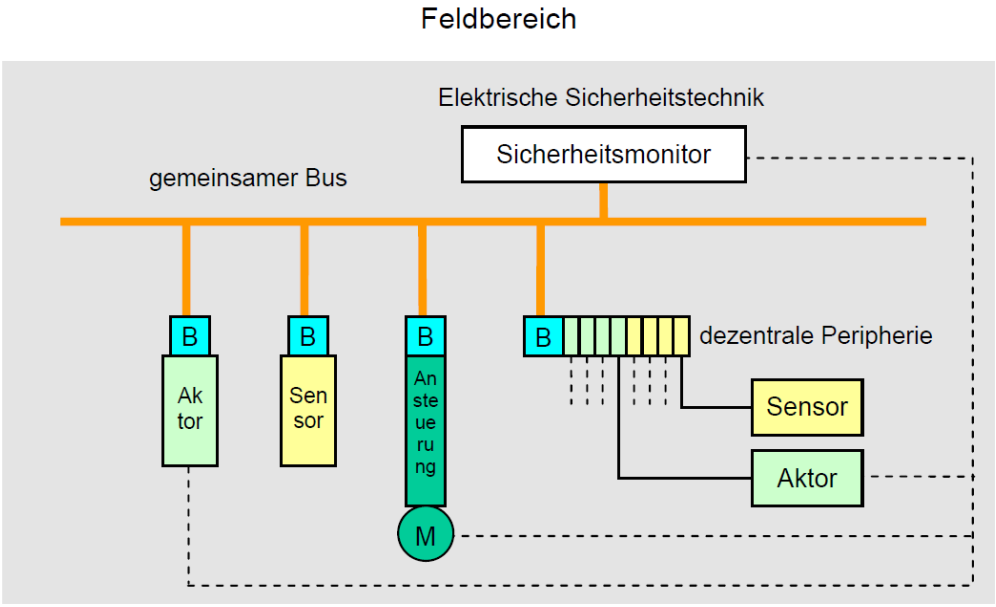


Abb. 2.7 Logikebene dezentral im Feldbereich verteilt

2.2.2 Anforderungen

Die Anfälligkeit vernetzter Datenverarbeitung gegenüber systeminternen und externen Fehlern wurde bereits zu Beginn der digitalen Signalverarbeitung erkannt. Deshalb werden bei der Auslegung, der Prüfung und dem Betrieb vernetzter digitaler softwarebasierter Systeme unterschiedliche Methoden eingesetzt, die das Ziel haben, eine robuste Netzwerkkommunikation zu erreichen. Einige grundsätzliche Prinzipien zur Auslegung der sicherheitsrelevanten Kommunikation wurden in technischen Regeln festgelegt, z. B. in

- Norm DIN EN 61500, „Kernkraftwerke, Datenkommunikation in Systemen, die Kategorie-A-Funktionen ausführen“ 2011 /DKE 11/,
- Normenreihe IEC 61850 „Communication Network and Systems”,
- IEEE 603 „Standard Criteria for Safety Systems for Nuclear Power Generating Units” /IEE 09/.

Die grundsätzliche Anforderung an die Funktion des sicherheitsrelevanten Netzwerkes ist die Sicherstellung der Zuverlässigkeit der erforderlichen Leittechnikfunktionen der Sicherheitsleittechnik auch bei einzelnen Fehlern im Netzwerk (siehe auch /DKE 11/, /IEE 09/). Dementsprechend sollen in der Auslegung sicherheitsrelevanter Leittechnik-Funktionen und deren Einrichtungen folgende Fehler- bzw. Ausfallszenarien berücksichtigt werden:

- Verlust von Daten im Netzwerk,
- Teil- und Gesamtausfall der Kommunikation im Netzwerk,
- Übertragung fehlerhafter Information im Netzwerk.

Die Auslegung der Leittechnik für Kernkraftwerke erfolgt u. a. nach folgenden Anforderungen:

- Aufrechterhaltung des Defence-in-Depth-Konzeptes der Anlage
  - Eine geforderte Maßnahme zur Umsetzung dieser Anforderung für digitale Leittechniksysteme ist hierbei, dass der Signal- bzw. Datenfluss in einer Richtung von den Systemen der höheren zu der niedrigeren Kategorie verlaufen soll. Weiterhin werden Vorkehrungen gegen die Fortpflanzung unterschiedlicher Fehler/Fehlerausfallarten (u. a. elektrische Fehler, Signal- und Datenfeh-

ler) über die Verbindungen, wie z. B. durch elektrische Trennung/Isolierung oder Signalvalidierung, gefordert.

- Aufrechterhaltung der verfahrenstechnischen Redundanz und Diversität
  - Für Kategorie A wird der Verzicht auf Verbindungen zwischen diversitären Leittechniksystemen gefordert. Weiterhin müssen für Leittechnikfunktionen der Kategorie A die Vorrang- und Einzelantriebssteuerungen für die verfahrenstechnischen Redundanzen strangweise aufgebaut werden.
- Leittechniksystemarchitektur
  - Zur Erreichung der geforderten Zuverlässigkeit der einzelnen Leittechniksysteme ist auch für die Interfaces innerhalb eines Systems ein redundanter Aufbau vorzusehen, um die zugrunde zu legenden Fehlerarten und -kombinationen sowie die versagensauslösenden Ereignisse zu beherrschen. Ebenso darf die in der Systemarchitektur vorhandene Redundanz der Teilsysteme aufgrund von Vermaschungen durch die Interfaces nicht beeinträchtigt werden.
- Unabhängigkeit von kommunizierenden Teilsystemen
  - Für Verbindungen zwischen redundanten Teilsystemen wird die gegenseitige Unabhängigkeit gefordert, so dass versagensauslösende Ereignisse nicht den Ausfall mehrerer Redundanzen zur Folge haben. So ist u. a. durch Barrieren zur Fehlerausbreitung sicherzustellen, dass sich Ausfälle und Fehlfunktionen in einer Redundanz nicht auf andere Redundanzen auswirken. Hierbei sind auch indirekte Verknüpfungen von Systemen über Service- oder Prüfgeräte zu berücksichtigen.
  - Bestehen auch Verbindungen zwischen Systemen mit unterschiedlichen leittechnischen Funktionen, sind die Anforderungen an die Rückwirkungsfreiheit der Schnittstellen untereinander ebenfalls gefordert. Andererseits werden hierzu auch datentechnische Barrieren gefordert, die die Auswirkungen durch das Fehlverhalten eines Teilsystems auf damit verbundene Teilsysteme verhindern. Hierzu sind geeignete Kommunikationsarchitekturen und Kommunikationsprotokolle einzusetzen.
- Unabhängigkeit und Performance
  - Zur Sicherstellung des vorherbestimmbaren (gerichteten) Verhaltens der Leittechniksysteme wird insbesondere auch die Unabhängigkeit der Funktion der

Interfaces von den zu übertragenden Daten sowie eine strikt zyklische Versendung der Daten gefordert.

  - Unter Berücksichtigung der Auswirkungen von Zeitverzögerungen infolge der Datenkommunikation muss die Einhaltung der spezifizierten Antwort- und Reaktionszeiten des Gesamtsystems unter allen zu unterstellenden Bedingungen garantiert werden können. Hierzu ist auch die vollständige Spezifikation und Dokumentation der unterschiedlichen Verhaltensweisen der Interfaces hinsichtlich Funktionalität und gerätetechnischer Eigenschaften gefordert.

- Fehlererkennung und -beherrschung

  - Es werden im Normenwerk sowohl Maßnahmen zur Fehlererkennung als auch zur Fehlerbeherrschung gefordert.

  - Fehlerhafte Signale sind zu erkennen und in ihrer Ausbreitung einzugrenzen.

  - Es ist eine regelmäßige Selbstüberwachung der Verbindungen vorzusehen.

  - Das Ausfallverhalten muss vollständig offen gelegt sein und in den jeweiligen Anwendungen eine definierte Vorzugsrichtung im Fehlerfall einnehmen.

- Sicherung

  - In den meisten Fällen erhöht der Einsatz von datentechnischen Verbindungen das Potenzial für unbefugte Manipulationen, da die datentechnischen Verbindungen zumeist den Fernzugriff auf die Systemkomponenten ermöglichen und aufgrund ihrer starken Verteilung in der Anlage nur schwer vollständig durch physikalische Maßnahmen geschützt werden können.

  - Es wird daher die Analyse der Bedrohungen, die sich durch die Interfaces ergeben, sowie die Festlegung entsprechender wirksamer informationstechnischer Schutzmaßnahmen gefordert. Ziel ist es hierbei, den Zugriff auf die Software und die Daten der Leittechniksysteme zu beschränken und unbefugte Änderungen zu verhindern.

- Kabel und Kabelverlegung

  - Die Anforderungen an die Kabel und die Kabelverlegung, die auch für elektrische Einzelsignalleitungen gelten, gewinnen bei der Verwendung von Interfaces eine höhere Bedeutung, da aufgrund der Konzentration vieler Signale in einem Signalkabel ein Ausfall eine weit höhere Fehlerauswirkung aufweist.

- Weiterhin können bei Netzwerkverbindungen teilweise aktive Netzwerkkomponenten in den Kabelweg eingeschleift werden, für welche die Robustheit gegen die Belastungen durch die Einsatzbedingungen ebenfalls gegeben sein muss.
- Die Kabel müssen für die unterstellten Umgebungsbedingungen ausgelegt sein, wobei auch eventuelle spezielle Belastungen, z. B. durch die Auslösung von Feuerbekämpfungssystemen, berücksichtigt werden sollen.
- Weiterhin sind physikalische Schutzmaßnahmen für die zu beherrschenden versagensauslösenden Ereignisse zu treffen. Es ist die physikalische Trennung zwischen redundanten Buskabeln und anderen Kabeltypen, die zu Beeinflussungen führen könnten, vorzusehen.
- Zusätzlich sind geeignete Maßnahmen zur Sicherstellung der Elektromagnetischen Verträglichkeit (EMV) zu treffen.

2.2.3 Generische Fehlerausfallarten

Digitale Leittechniksysteme bestehen aus elektronischen Komponenten, deren Eigenschaften und insbesondere Ausfallarten aufgrund ihrer Komplexität nicht von außen erkennbar sind und sie auf diese Art schwer zugänglich für Sicherheitsanalysen macht. Es ist nicht möglich, die Gesamtheit potenzieller Fehler für diese Systeme zu erfassen. In sicherheitstechnisch wichtigen Anwendungen jedoch ist dies von essenzieller Relevanz, um das System derart auszulegen, dass es bei eintretenden Fehlfunktionen die sicherheitstechnische Aufgabe dennoch erfüllt.

Als Grundlage der Untersuchungen zur Entwicklung eines methodischen Ansatzes zur Analyse des Einsatzes von Netzwerktechnik in Sicherheitsanwendungen ist es unerlässlich, Fehlermodi zu definieren. Auf diesen stützt sich die Analyse dieser Systeme. Es ist daher notwendig, möglichst abdeckende Fehlermodi zu finden und diese klar von den Fehlerursachen zu trennen. Denn die Ursache ist im Endeffekt von untergeordneter Relevanz, wenn durch sie derselbe Fehlermodus induziert wird.

Im Vergleich zur analogen Signalübertragung treten bei der digitalen Datenkommunikation neuartige Fehlertypen auf, die sich aus der paketorientierten Datenübertragung in einem Netzwerk ergeben /MIC 05/.

Jedes Datentelegramm besteht aus einem Telegrammkopf (Header) und einem Nutzdatenanteil. Der Header enthält für den Kommunikationsablauf relevante Informationen, während der Nutzdatenanteil die zu übermittelnden Nutzdaten enthält. Eine Nachricht kann hierbei über mehrere Telegramme verteilt sein.

Um ein hohes Abstraktionsniveau und den generischen Charakter der Untersuchung zu wahren, wird die Netzwerkkommunikation in ihrer einfachsten Form betrachtet:

- Ein Sender übermittelt Informationen in Form von Datentelegrammen auf dem Übertragungsmedium an einen Empfänger.
- Der Erfolg der Kommunikation ist davon abhängig, dass auf dem Weg vom Sender zum Empfänger die Fehler oder Übertragungsstörungen - gleich welcher Art - nicht auftreten oder falls doch, deren Auswirkungen auf die Kommunikation beherrscht werden.

Die folgenden Komponenten können mögliche Fehlerquellen darstellen:

1. Fehler auf Senderseite
2. Fehler auf dem Übertragungsmedium
3. Fehler auf Empfängerseite
4. Systemübergreifende Fehler aufgrund von Interaktionen der Netzwerkkomponenten

Generell betrachtet man zwei Klassen von Fehlern, die in der Kommunikation von Sicherheitssystemen auftreten können: Entweder ist (1) die Information fehlerhaft oder (2) die Übertragung der Information schlägt fehl. Fehler der Klasse (1) betreffen meist die Struktur des Datentelegramms selbst oder führen zu einer Verzögerung der Nachricht, so dass sie als ungültig angesehen wird. Fehler der Klasse (2) beziehen sich auf den vollständigen Verlust des Datentelegramms.

Dabei ist zu berücksichtigen, dass die Grenzen zwischen den Klassen fließend sind und eine eindeutige Zuordnung in manchen Fällen sehr schwierig ist.

Nachfolgend werden die möglichen kommunikationsbezogenen Fehlerarten beschrieben.

- Verfälschtes Telegramm, Nachrichtenverfälschung /ORN 07/, /MIC 05/

  - Ein Datentelegramm kann aufgrund verschiedenster Ursachen in seiner Struktur verändert sein. Elektromagnetische Einkopplungen auf dem Übertragungsmedium, Fehler im Prozessor oder Pufferspeicher oder Interferenzen jeglicher Natur können der Auslöser dafür sein, dass ein Bit kippt.

  - Da die Verfälschung von Daten ein allgemein bekanntes Phänomen in der Digitaltechnik ist, gibt es wirkungsvolle Verfahren wie Paritäts- oder Cyclic-Redundancy-Checks, die in der Lage sind, Fehler zu erkennen und bis zu einem gewissen Grad auch zu beheben.

  - Je nach Protokoll muss ein verfälschtes Telegramm nicht zum vollständigen Verlust des Telegramms führen. Es gibt neben der Fehlerkorrektur beispielsweise die Möglichkeit, das Telegramm nach einer bestimmten Zeit erneut zu senden. Für den Einsatz fehlerkorrigierender Techniken ist zumindest eine Halb-Duplex-Kommunikation notwendig, andernfalls hat der Empfänger keine Möglichkeit, eine Rückmeldung über die Richtigkeit der empfangenen Daten an den Sender zurückzugeben.

- Ungewollte Telegrammwiederholung /ORN 07/, /MIC 05/

  - Aufgrund von Störungen kann es zu einer Wiederholung eines Telegramms zu einem unerwünschten bzw. unerwarteten Zeitpunkt kommen.

  - Im Allgemeinen stellt die Wiederholung eines Telegramms einen gängigen Vorgang in der Netzwerkkommunikation dar. Beispielsweise kann ein Empfänger feststellen, dass ein erwartetes Datentelegramm fehlt und dieses erneut anfordern. Um die Zuverlässigkeit der korrekten Übertragung von Telegrammen zu erhöhen, ist es darüber hinaus üblich, dass ein Sender ein Telegramm redundant sendet.

  - Abhängig von der Kommunikationstopologie, dem Kommunikationsmedium und dem Netzwerkprotokoll werden verschiedene Verfahren zum mehrfachen Senden von Nachrichten genutzt: Der Sender kann dasselbe Telegramm mehrfach zeitlich versetzt senden oder er schickt das Telegramm auf verschiedenen Routen durch das Netzwerk.

- Falsche Reihenfolge, Falscher Ablauf /ORN 07/, /MIC 05/
  - Dieser Fehlertyp liegt vor, wenn die zeitliche Reihenfolge von Telegrammen durch eine Störung so beeinflusst wird, dass eine ältere Nachricht nach einer zeitlich aktuelleren Nachricht beim Empfänger eintrifft.
  - Generell kann das Netzwerkprotokoll Verfahren enthalten, welche die Reihenfolge von Telegrammen verändert. Beispielsweise können Nachrichten Prioritäten zugeordnet werden und die Nachricht mit der höheren Priorität wird zeitlich bevorzugt gesendet. Es gibt darüber hinaus verschiedenartige Speichersysteme zum Zwischenspeichern von Telegrammen. Diese Speichersysteme befinden sich in aktiven Netzwerkkomponenten wie Switches, Routern und Bridges. Da dies ein bekannter Fehlermodus in der Netzwerkkommunikation ist, gibt es Verfahren zur Erkennung von Telegrammen, die in der falschen Reihenfolge gesendet wurden.
- Telegrammverlust /ORN 07/, /MIC 05/
  - Ein Verlust durch einen Fehler im Sender, Empfänger oder auf der Busverbindung bedeutet die komplette Löschung des Telegramms. Das Telegramm gilt als verloren, wenn es nicht beim Empfänger ankommt oder der Empfänger den Erhalt des Telegramms nicht bestätigt.
- Inakzeptable Verzögerung /ORN 07/, /MIC 05/
  - Eine inakzeptable Verzögerung liegt vor, wenn Nachrichten außerhalb des Zeitintervalls, welches für die Nachricht als akzeptabel definiert wurde, beim Empfänger eintreffen.
  - Ein Vorteil von digitalen Verarbeitungssystemen ist deren Fähigkeit zur Fehlererkennung und -behebung (z. B. CRC). Diese Techniken benötigen jedoch Zeit zur Ausführung und generieren einen natürlichen Verzug. Das Auslesen von Nachrichten durch aktive Netzwerkkomponenten wie Switches zur Identifikation der Telegrammdestination kosten ebenfalls Zeit. Auch die Belegung des Übertragungsmediums durch andere Telegramme führt dazu, dass Nachrichten verzögert werden.
  - Die auftretenden Zeitverzögerungen durch Fehlerbehebungsmechanismen und der Aufbau des Netzes müssen in den Abfragezyklen des Netzwerks berück-

sichtigt werden und maximale Zeitintervalle definiert werden, außerhalb derer die Nachricht als verloren gilt.

- Insertion, Einfügung /ORN 07/, /MIC 05/

  - Es kann durch Fehler, Störungen oder Interferenzen passieren, dass Daten von unbekannten Quellen über das Übertragungsmedium „eingefügt“ werden. Es gibt also zusätzliche Daten zu dem bereits bestehenden Datenstrom. Diese eingefügten Daten können beim Empfänger weder als richtig interpretiert werden noch stellen sie eine unbeabsichtigte Wiederholung oder eine falsche Reihenfolge dar, da der Empfänger die Quelle nicht kennt.

- Maskerade /ORN 07/

  - Als Maskerade wird bezeichnet, wenn ein Datenstrom bzw. -telegramm die richtige Struktur hat, um beim Empfänger als valide anerkannt zu werden, es sich jedoch tatsächlich um ein „falsches“ Telegramm handelt. Dabei kann die Quelle des Telegramms durchaus ebenfalls „echt“ sein.

  - Dieser Fehlermodus sollte insbesondere in Bezug auf die IT-Sicherheit betrachtet werden, da auf diese Weise unerwünschte oder schadhafte Software aufgespielt werden könnte. Zusätzliche Erkennungsmechanismen zur Erkennung von Maskeraden sollten in sicherheitsrelevanten Netzwerken implementiert werden.

- Falsche Adressierung /ORN 07/

  - Durch Störeinflüsse kann es dazu kommen, dass das Adressierungsfeld derart verändert wird, dass das Datentelegramm an einen anderen für dieses Telegramm nicht vorgesehenen Teilnehmer geschickt wird. Da lediglich die Adressierung fehlerhaft ist und der Rest des Telegramms der durch das Protokoll vorgegebenen Struktur entspricht, würde dieser Teilnehmer die Nachricht als wahr behandeln.

- Broadcast-Sturm /ORN 07/

  - Bei einem Broadcast-Sturm wird das Netzwerk mit Broadcast- oder Multicast-Nachrichten geflutet. Ein Broadcast-Telegramm bewirkt, dass alle Stationen gleichzeitig Antworten generieren, welche wiederum von allen Stationen beantwortet werden. Es entsteht ein lawinenartig steigender Anfall an Broadcast-Nachrichten, was einen signifikanten Teil der zur Verfügung stehenden Netz-

werkkapazitäten aufbrauchen kann. Dies kann soweit führen, dass das Netzwerk durch Überlastung lahmgelegt wird.

  - Das Auftreten von Broadcast-Stürmen ist vorwiegend von der Netzwerktopologie und der Konfiguration des Netzwerks abhängig. Durch korrekte Implementierung von Protokollen und Netzwerkkomponenten können destruktive Broadcast-Nachrichten geblockt werden. Broadcast-Stürme treten meist in Netzwerken auf, in denen Nachrichten auf vielen redundanten Wegen ihr Ziel erreichen können.

- Inkonsistenz (byzantinischer Fehler) /ORN 07/

  - Eine Inkonsistenz entsteht dann, wenn ein Netzwerkteilnehmer z. B. einen Wert von zwei redundanten Quellen erhält und diese sich unterscheiden. Der Teilnehmer weiß nicht, welche Information wahr ist und es entsteht Konfusion, welche sich weiter ausbreiten kann.

  - Der byzantinische Fehler kommt ausschließlich in redundant aufgebauten Systemen vor. Dabei kann er, durch einen Einzelfehler getriggert, die Kommunikation über das gesamte redundante System hinweg zum Erliegen bringen. Eine Möglichkeit, diesen Fehler zu unterbinden, wäre zwei Prozessoren den gleichen Wert berechnen zu lassen und ein Voting für diesen Wert durchzuführen.

- Babbling Idiot /ORN 07/

  - Bei einem Babbling Idiot handelt es sich um einen Netzwerkteilnehmer, der zu beliebigen Zeitpunkten Daten sendet.

  - Dieses Problem ist in Broadcast-Bussystemen ein besonders schwerwiegender Fehler. Die Station, welche von dem Babbling-Idiot-Fehler betroffen ist, ignoriert das dem Busnetzwerk zugrunde gelegte Buszugriffsverfahren und sendet zu jedem beliebigen Zeitpunkt Daten, sodass Nachrichten von intakten Stationen korrumpiert werden. Des Weiteren beschneidet der Babbling Idiot die restlichen Busteilnehmer ihrer Netzwerkressourcen, da er zusätzliche Nachrichten produziert. Vor allem ereignisgesteuerte Bussysteme unterliegen hier der Kritik, den Babbling-Idiot-Fehler nicht beherrschen zu können, da es im Prinzip vorstellbar ist, dass er bei Dauersenden die gesamte Netzwerkkommunikation zum Erliegen bringen kann.

- Der Babbling Idiot ist ein Fehler, der sowohl in deterministischen als auch in nicht deterministischen Netzwerken vorkommen kann.

- Exzessives Jittern /ORN 07/

  - Jittern ist die zeitlich variable Verzögerung von Ausgangswerten in verschiedenen Abfragezyklen, d. h. der berechnete Wert wird in verschiedenen Abfragezyklen zu verschiedenen Zeiten ausgegeben. Die Abweichung ist dabei so klein, dass es keine inakzeptable Verzögerung darstellt.

  - Die Auswirkungen des Jitterns sind in Echtzeitanwendungen naturgemäß kritischer als in Anwendungen, deren Reaktionszeit von untergeordneter Bedeutung ist.

  - Die Probleme, welche durch Jittern induziert werden, liegen meist in der Akkumulation von Verzögerungen. Einerseits können diese Verzögerungen auf dem Übertragungsmedium auftreten. Beispielsweise entstehen in nichtdeterministischen Netzwerken Verzögerungen durch die Auflösung von Kollisionen. Andererseits tritt Jittern als direkte Folge von verzögerten Antworten der Netzwerkteilnehmer auf die Anfrage eines Masters im System auf.

  - Um Problemen durch Jittern vorzubeugen, ist es möglich Puffer einzusetzen, welche ihrerseits eine zusätzliche, fixe Verzögerung in das Netzwerk einbringen.

- Kollision /ORN 07/

  - Kollisionen entstehen, wenn zwei oder mehr Teilnehmer gleichzeitig Daten über das Übertragungsmedium senden wollen. Dabei überlagern sich die gesendeten Daten, diese Interferenz führt zu Verfälschungen der Originaldaten, sodass diese nicht mehr genutzt werden können. In nichtdeterministischen Netzwerken sind Kollisionen intrinsisch. Dort werden Verfahren (z. B. CSMA) genutzt, welche Kollisionen detektieren und auflösen.

  - Da Kollisionen zu aperiodischen Ansprechzeiten führen können und nicht alle auf Kollisionen zurückzuführenden Fehler vom Netzwerk erkannt werden können, wird die Nutzung nichtdeterministischer Netzwerke für den Einsatz in der Sicherheitsleittechnik als kritisch betrachtet.

2.2.4 Sicherheitsbussysteme

Gegen die verschiedenen Fehlertypen der Datenkommunikation sind zur Erreichung der geforderten Zuverlässigkeit Maßnahmen zur Fehlerbeherrschung zu ergreifen. In Standardfeldbussen sind derartige Vorkehrungen nur teilweise implementiert, so dass Sicherheitsbusvarianten auf der Basis von Standardfeldbussystemen entwickelt wurden.

Tabelle 2.4 zeigt eine Zusammenfassung der wichtigsten zurzeit am Markt vertretenen Sicherheitsbussysteme. Sie zeichnen sich dadurch aus, dass sie auftretende Kommunikationsfehler sicher erkennen und eine entsprechende Fehlerkorrektur vornehmen oder entsprechende Fehlerreaktionen veranlassen können. Die Konzepte, um diese Zuverlässigkeit zu erreichen, sind bei den Bussystemen jedoch unterschiedlich.

Die Entwicklung der Sicherheitsbussysteme orientiert sich meist an der Norm IEC 61508 /IEC 10a/ für allgemeine industrielle Sicherheitsanwendungen. Hier werden Sicherheitsfunktionen entsprechend ihrer sicherheitstechnischen Bedeutung in SIL 1-4 eingeteilt. Zur Sicherstellung der Zuverlässigkeit der Sicherheitsfunktion fordert die IEC 61508 in Abhängigkeit vom zu erreichenden SIL-Level fehlerbeherrschende sowie fehlervermeidende Maßnahmen. Zur Sicherstellung der Zuverlässigkeit der Sicherheitsfunktion wird bei Erkennung eines Fehlers ein Fail-Safe-Verhalten durch den Übergang in einen definierten Fehlerzustand genutzt. Um die Wirksamkeit der fehlerbeherrschenden Maßnahmen zu überprüfen, muss die Sicherheitsfunktion einer Wahrscheinlichkeitsbetrachtung unterzogen werden. Dabei wird die sog. gefährliche Versagenswahrscheinlichkeit für nicht entdeckte Fehler, die zum Versagen der Sicherheitsfunktion führen, ermittelt. Diese muss unterhalb eines vom SIL-Level festgelegten Grenzwerts bleiben.

Eine grundsätzliche Voraussetzung beim Einsatz dieser Systeme stellt das Vorhandensein eines Fail-Safe-Zustandes dar, der als Reaktion auf erkannte Fehler eingenommen werden kann. Für die Sicherheitsleittechnik in Kernkraftwerken muss hierzu im Einzelnen geprüft werden, ob und wie sich leittechnische Komponenten mit diesem Verhalten unter Berücksichtigung der zu stellenden Zuverlässigkeitsanforderungen in das Leittechniksystem integrieren lassen.

Tab. 2.4 Sicherheitsbussysteme

Gelb: Pollingverfahren (Master/Slave), weiß: Master/Slave-Verfahren mit Sicherheitsmonitor, rot: Summenrahmen, grün: CSMA/CA (CAN), blau: CSMA/CD (Ethernet) /MIC 05/.

BUS	Topologie	Bemerkungen	Zertifizierung	Einführung	Medium	Hersteller
PROFIsafe	Linienförmig (zentral: Ringförmig)	Monomaster Pollingverfahren (zentral: Multimaster Token)	IEC 61508 SIL 3 (TÜV Süd), DIN 19250 AK 6, EN954-1, Kategorie 4	2002	CU, LWL	PNO offen
AS-Interface Safety	Linien-, Stern- und Baumförmig	Monomaster Pollingverfahren Sicherheitsmonitor (keine Sicherheits-SPS erforderlich)	IEC 61508 SIL 3 (TÜV Rheinland) EN954-1 Kategorie 4 (TÜV-Nord)	2000	CU	Siemens
Interbus Safety	Ringförmig (von außen gesehen Linienstruktur)	Monomaster Summenrahmen Überwachungseinheit „Safe-Control“ nur für einfache E/A-Geräte	EN954-1 Kategorie 4 (BIA)	1987	CU, LWL	Phoenix DIN E 19258
SafetyBUS p	Linienförmig (Stern-/Baumstrukturen mit Netzstrukturelementen)	basiert auf CAN, Master/Slave Polling-Verfahren, Multimasterfähig	IEC 61508 SIL 3 (TÜV Süd) u. U. SIL 4, DIN 19250 AK 6 (TÜV Süd) EN954-1, Kategorie 4	1999	CU, LWL	Pilz offen
DeviceNET Safety	Linienförmig, Stichleitungen	basiert auf CAN CIP Multimasterfähig, direkte Kommunikation möglich (ohne Master)	IEC 61508 SIL 3 (TÜV Rheinland)	2005	CU	OVDA offen
CANopen Safety Chip	Linienförmig	CSMA/CA Busarbitrierung, White Channel-Prinzip, Multimasterfähig, direkte Kommunikation möglich (ohne Master)	IEC 61508 SIL 3 (TÜV Rheinland)	2004	CU, LWL	CAN in Automation
Esalan	Linienförmig	basiert auf CAN Multimasterfähig direkte Kommunikation möglich (ohne Master)	DIN 19250 AK 6 EN 954-1 Kategorie 4 (BIA)		CU, LWL	Elan proprietär
EPL Safety	Linien-/Baumförmig	basiert auf Ethernet-Hardware Safety-Frame CAN entlehnt direkte Kommunikation möglich (ohne Master)	IEC 61508 SIL 3 (TÜV Rheinland) u. U. SIL 4 (Zertifizierung durch TÜV Rheinland noch nicht abgeschlossen)	2005	CU, LWL	Innotec, B&R,….
SaveEthernet	Sternförmig	basiert auf Ethernet	IEC 61508 SIL 3 (TÜV Rheinland) EN 954-1 Kategorie 4 (TÜV Rhld)	1997	CU, LWL	HIMA proprietär

Bei Sicherheitsbussystemen lassen sich zwei verschiedene Ansätze unterscheiden: Das Grey-Channel-Prinzip und der Ausbau bestehender Sicherungsmechanismen, der in Analogie zum Grey-Channel nachfolgend als „White Channel“ bezeichnet wird /PEL 10/. Beide Ansätze werden im Folgenden beschrieben.

2.2.4.1 White-Channel-Prinzip

Der Ausbau bestehender Sicherungsmechanismen, in Analogie zum Grey-Channel künftig White-Channel genannt, ist im Gegensatz zum Grey-Channel ein hardwarenaher Ansatz (z. B. CANopen, Safety Chip, Esalan). Die Basis bildet ein Standardfeldbus, der bereits mit fehlerkorrigierenden Maßnahmen ausgestattet ist (z. B. Hamming-Distanz, Telegrammwiederholung). Die bereits geringe Wahrscheinlichkeit unerkannter Fehler des zugrunde liegenden Standardfeldbusses wird durch weitere Maßnahmen in der Treiber-Hardware, wie Wiederholung der Telegramme mit invertierten Daten auf redundanten Busanschaltungen oder durch die Einführung zusätzlicher unabhängiger Überwachungskomponenten, weiter reduziert. Die resultierende Fehlerrate gelangt so in eine Größenordnung, die für die sicherheitsgerichtete Kommunikation gemäß IEC 61508 anforderungsgerecht ist.

Beim Ausbau bestehender Sicherungsmechanismen müssen keine Sicherungsinformationen in das Telegramm eingebettet werden und der gesamte Nutzdatenanteil bleibt erhalten. Da die sichere Datenübertragung durch die Erweiterung bestehender Sicherungsmechanismen auf Hardwareebene erfolgt, können Übertragungsfehler teilweise korrigiert und die Verfügbarkeit der Netzwerkkommunikation verbessert werden. Auch in diesem Fall ist ein Mischbetrieb mit nicht sicherheitsgerichteten Nachrichten möglich, da das Busprotokoll selbst nicht verändert wird. Im Gegensatz zum Grey-Channel-Ansatz können jedoch keine Standardkomponenten für die Buskommunikation verwendet werden, da das Übertragungsmedium nicht, wie beim Grey-Channel-Ansatz, unverändert beibehalten wird. Die spezielle Entwicklung von Hardwarekomponenten ist daher erforderlich.

2.2.4.2 Grey-Channel-Prinzip

Für sicherheitsrelevante Kommunikationsnetzwerke gilt es in vielen Branchen, Anforderungen an die Zuverlässigkeit der Kommunikation zu erfüllen. In der Verkehrstechnik

ist hierbei die DIN EN 50159-2 /DIN 11/ maßgeblich. Sie definiert Grundsätze zum Design von Kommunikationsarchitekturen in sicherheitsrelevanten Bereichen. Die Architekturen werden in zwei verschiedene Teilbereiche aufgeteilt, einer Sicherheitsebene, die verschiedenste Sicherheitsanforderungen gemäß ihres SIL-Levels erfüllen muss, und einem „Grey Channel“ ohne detaillierte Anforderungen an die funktionale Sicherheit, in dem Spezifikationen nur teilweise erfüllt werden /PEL 10/. Eine schematische Darstellung ist in Abbildung 2.8 zu sehen.

Die eingebauten Funktionen auf der Sicherheitsebene müssen das Fail-Safe-Prinzip berücksichtigen, was typischerweise die Unterbrechung der Kommunikation bis hin zum Neustart des fehlerhaften Systems oder einem manuellen Zurücksetzen bedeutet.

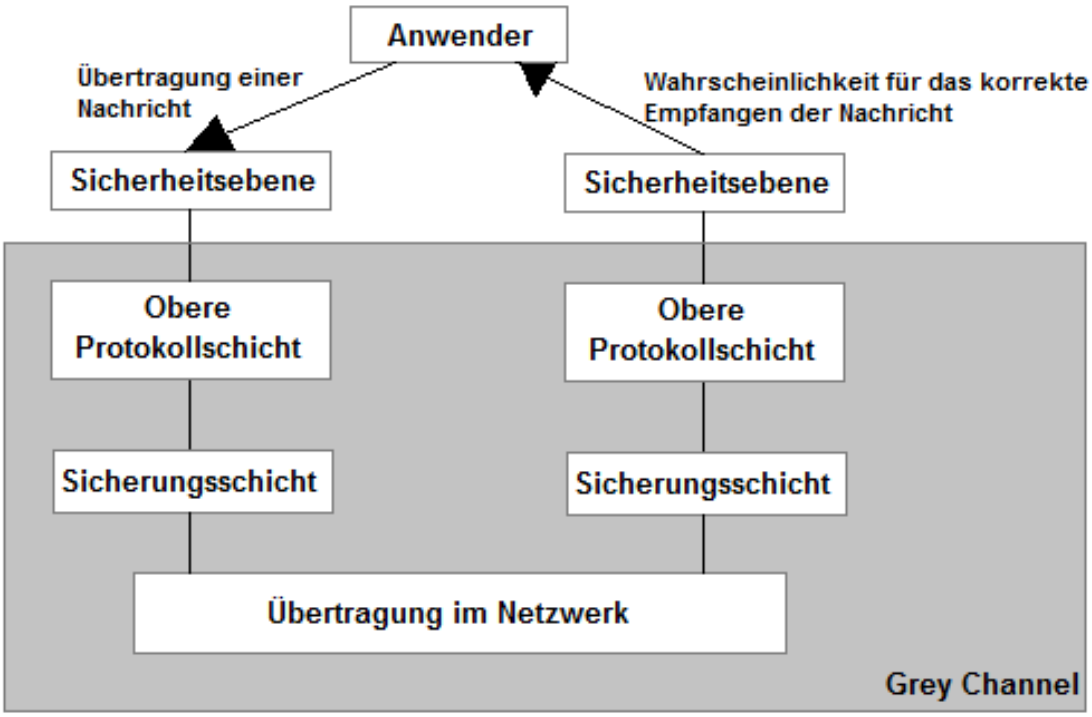


Abb. 2.8 Schematische Darstellung des Grey Channels

Die einfache Kombination von Maßnahmen zur Erhöhung der Fehlertoleranz im Grey Channel mit denen in der Sicherheitsebene im Kommunikationsprotokoll führt nicht zwangsläufig zu einer allgemeinen Verbesserung der Zuverlässigkeit des Kommunikationsnetzwerks. Werden beispielsweise verlorene Datentelegramme im Grey Channel nach einer gewissen Zeit wiederholt abgesendet, kann es passieren, dass eine Zeitüberprüfung, die in der Sicherheitsebene etabliert ist, das erneut gesendete Telegramm als veraltet erkennt und verwirft. Daher müssen Maßnahmen, die die Zuverläs-

sigkeit des Grey Channels erhöhen sollen, im Detail auf die Frage hin analysiert werden, ob ihre Wirksamkeit im Zusammenspiel mit der Sicherheitsebene gewährleistet ist. Nur in diesem Fall wird die Zuverlässigkeit des Kommunikationsnetzwerkes, bestehend aus Sicherheitsebene und Grey Channel, tatsächlich erhöht /ORN 07/.

Sicherheitsbuskomponenten nach dem Grey-Channel-Prinzip setzen auf einem Standardfeldbus-System auf, wobei dessen Übertragungskanal unverändert genutzt wird. Da dieser Übertragungskanal als unsicher betrachtet wird, dient der Nutzdatenteil des Telegramms lediglich als Transportmedium für das eingebettete sicherheitsgerichtete Telegramm, welches zusätzliche, eigenständige Sicherungsmechanismen aufweist. Die Sicherheitstelegramme werden damit durch den unsicheren Übertragungskanal getunnelt. Der Empfänger überprüft anhand der mitgelieferten Sicherungsinformationen, ob die Übertragung korrekt war und veranlasst gegebenenfalls den Übergang in den Fail-Safe-Zustand. Da die Sicherungsmechanismen im Protokoll verwirklicht und in das Telegramm eingebettet werden, ist dies ein softwarebasierter Ansatz (z. B. PROFIsafe, DeviceNET Safety) /MIC 05/.

Vorteil dieses Ansatzes ist, dass der in dezentralen Ein-/Ausgabebaugruppen verwendete Rückwandbus als Teil des unsicheren Kanals aufgefasst werden kann und in der Zertifizierung bzw. Begutachtung nicht zusätzlich betrachtet werden muss. Mit diesem Ansatz ist weiterhin eine Übertragung sowohl sicherheitsrelevanter und als auch konventioneller Telegramme möglich. Die Standardbusteilnehmer ignorieren in diesem Fall die zusätzlichen Sicherungsinformationen, die für sie Nutzdaten darstellen. Ebenso ist die Verwendung von Standardindustriekomponenten für den Aufbau der Businfrastruktur möglich. Da die Sicherungsmechanismen im Protokoll verwirklicht werden, ist es leicht, diesen softwarebasierten Ansatz auf Fremdprodukte zu portieren. Beim Einsatz entsteht hinsichtlich der Busanschaltung kein zusätzlicher Hardwareaufwand im Vergleich zu Standardbusanschaltungen. Die zusätzlichen Sicherungsinformationen bedingen im Gegenzug ein komplexeres Kommunikationsprotokoll und damit eine höhere Komplexität der Software. Der je Telegramm übertragbare Nutzdatenanteil sinkt im Vergleich zum Standardbussystem durch das Hinzufügen der Sicherungsinformation. Der geringere Nutzdatenanteil und der softwarebasierte Ansatz führen weiterhin zu Geschwindigkeitseinbußen im Vergleich zum Standardfeldbus /MIC 05/.

3 Entwicklung methodischer Ansätze zur Analyse potentieller Netzwerkfehler

3.1 Vorgehensweise

Ziel des Vorhabens ist die Durchführung einer phänomenologischen Untersuchung zu potentiellen Fehlerquellen und Fehlerfortpflanzungspfaden im Netzwerk eines generischen Leittechniksystems sowie die Entwicklung methodischer Ansätze zur Analyse der Verbreitung und der Auswirkungen postulierter Fehler in typischen Netzwerken.

Um dieses Ziel zu erreichen, wurde in diesem Vorhaben gemäß Abbildung 3.1 vorgegangen:

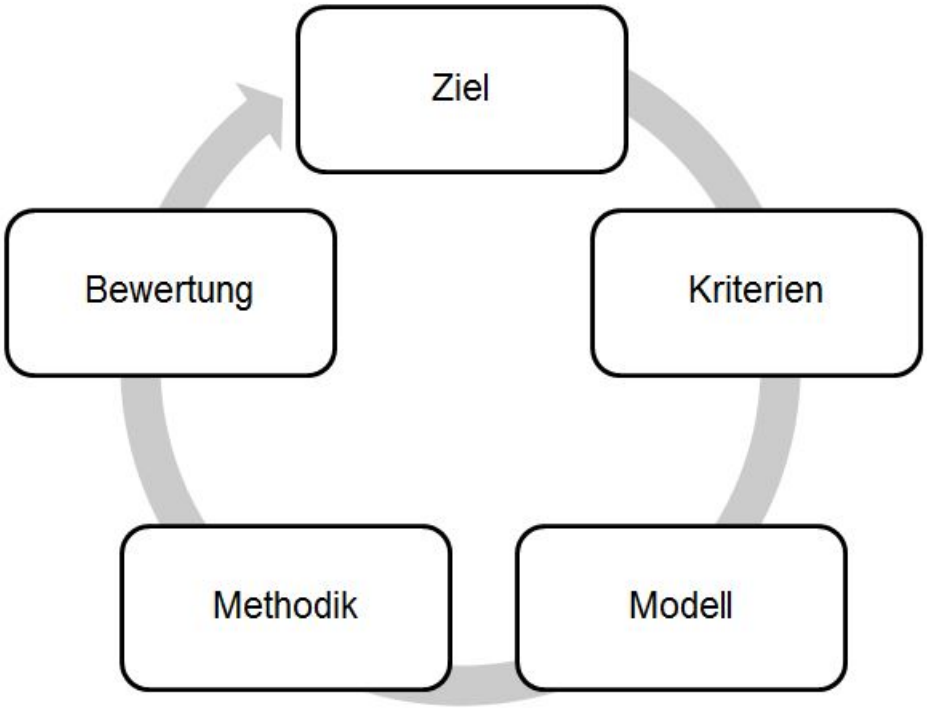


Abb. 3.1 Ablaufdiagramm zur Entwicklung eines Ansatzes zur Analyse der Verbreitung und Auswirkung postulierter Fehler in typischen Netzwerken

Der Ablauf wird in die folgenden Schritte unterteilt:

- Definition der **Ziel**stellung des methodischen Ansatzes
- Anhand der Zielstellung werden **Kriterien** abgeleitet, welche die Modellierung des generischen Leittechniksystems und die Auswahl der auf dieses System angewandten Methodik maßgeblich bestimmen sollen. Die Kriterien orientieren sich an

den für Netzwerktechnologien charakteristischen Merkmalen, welche für die Zuverlässigkeitsbewertung bestimmend sind. Eine Erläuterung dieser Kriterien befindet sich in Kapitel 3.2.

- Im nächsten Schritt wird ein generisches **Modell** eines digitalen Leittechniksystems erarbeitet. Dazu wurden der GRS bekannte und in Kernkraftwerken eingesetzte digitale Leittechniksysteme zugrunde gelegt. Solche Systeme werden z. T. in deutschen Anlagen zur Realisierung von Begrenzungsfunktionen, im internationalen Umfeld auch für Reaktorschutzfunktionen, eingesetzt. Das Modell wird in Kap. 3.3 vorgestellt.

- Anschließend wird eine geeignete **Methodik** für die Zuverlässigkeitsbewertung ausgewählt. Dazu werden die in Anhang A.4 und A.5 vorgestellten Methoden hinsichtlich der in Kapitel 3.2 beschriebenen Kriterien bewertet und der Detaillierungsgrad des Leittechniksystems sowie des Kommunikationsnetzwerks festgelegt. In Kapitel 3.3.4 wird der gewählte Ansatz zur Bewertung des generischen Modells des digitalen Leittechniksystems vorgestellt.

- Der letzte Schritt umfasst die Festlegung des gewählten Bewertungsansatzes für die **Bewertung** des generischen Modells des digitalen Leittechniksystems. Dieser wird in Kapitel 3.3.4 vorgestellt. Insbesondere wird der Bezug der Auswahl von Methodik und Modell zur Zielstellung hergestellt.

3.2 Kriterien zur Entwicklung einer Methodik für die Bewertung potentieller Netzwerkfehler

Kriterium 1: Generischer Anwendungsbereich

Zur Entwicklung einer Methodik zur Analyse und Bewertung der Netzwerkkommunikation in einem Sicherheitsleittechniksystem soll in diesem Projekt ein Modell eines generisches Leittechniksystems verwendet werden, um die Fehlerbeherrschung bzw. das Fehlerverhalten in unterschiedlichen Netzwerktopologien beurteilen zu können. Damit werden Details der logischen Signalverarbeitung eines Leittechniksystems vernachlässigt und der Fokus auf die Kommunikationsabläufe gelegt. Somit soll das Modell eine Vielzahl von Leittechniksystemen und -architekturen beschreiben können und deren Bewertung auf einer generischen Basis ermöglichen.

Eine Bewertungsmethode muss daher auch dann noch imstande sein, Modelle eines Leittechniksystems bewerten zu können, wenn es sich um ein generisches Modell handelt, in dem die Wechselwirkungen zwischen Elementen des Leittechnikmodells nicht genau bekannt sind.

Kriterium 2: Schwachstellenidentifikation

Eine wichtige Aufgabe zur Bewertung der Netzwerkkommunikation in Leittechniksystemen ist die Identifizierung von Schwachstellen. Eine Schwachstelle eines modellierten Systems ist durch Elemente oder Funktionen des Systems charakterisiert, deren Ausfall in Häufigkeit und Auswirkung besonders hervorstechen.

Die zur Bewertung verwendeten Methoden sollen die Möglichkeit bieten, Schwachstellen der Netzwerkkommunikation des betrachteten Systems zu erkennen und zu lokalisieren (z. B. Hardware, Software, Betriebsart, Prüfung). Die Methode soll nicht nur qualitative oder quantitative Aussagen zum Einfluss der Netzwerkkommunikation auf das Gesamtsystem produzieren, sondern auch den Einfluss von Teilausfällen der Netzkommunikation sichtbar machen. Methoden mit hierarchischer Struktur, welche die Bewertung von vernetzten Teilsystemen bzw. -funktionen transparent in die Gesamtbewertung aufnehmen, sind hierfür gut geeignet.

Kriterium 3: Möglichkeiten zur Quantifizierung der Zuverlässigkeit

Die Methode soll die Möglichkeit bieten, das Modell auch quantitativ zu bewerten. Die Erstellung von Zuverlässigkeitskenngrößen soll dabei nachvollziehbar sein und bzgl. der wesentlichen Faktoren wie Topologie, Kommunikationsprotokoll, Schnittstellen etc. sensibel sein. Solche Indikatoren sind zum Vergleich verschiedener Kommunikationstechnologien zur Umsetzung der gleichen Sicherheitsfunktionen nützlich.

Kriterium 4: Berücksichtigung von redundanten und diversitären Topologien der Netzwerkkommunikation

Eine wichtige Eigenschaft moderner Kommunikationstechnologien ist die Möglichkeit, viele Kommunikationsteilnehmer u. a. auch gleichzeitig über dasselbe Medium kommunizieren zu lassen. Dies kann zu einer effektiven Auslastung des Mediums sowie zu schnelleren Prozesszeiten führen. Des Weiteren führt die redundante und z. T. diversi-

täre Signalverarbeitung in der Sicherheitsleittechnik zu einer Vervielfachung der Signale im Auslösepfad.

Die Methode soll daher in der Lage sein, parallele und redundante Kommunikation sowie logische Verschachtelungen der Signalwege bewerten zu können. Moderne Leittechniksysteme profitieren von der Vervielfachung der Signale, z. B. durch den Austausch von Daten zwischen den Redundanzen zur Fehlererkennung. Solche Aspekte der Kommunikation sollen von der Methode beachtet werden können und in die Zuverlässigkeitsbewertung einfließen.

Kriterium 5: Berücksichtigung von automatischen Fehlerbehandlungsmaßnahmen in Netzwerken

Moderne digitale Leittechniksysteme besitzen zahlreiche Vorkehrungen zur Entdeckung und Korrektur von Netzwerkfehlern sowie zur Vermeidung der Fehlerausbreitung. Dabei kann es sich um netzwerkimmanente Maßnahmen (z. B. im Netzwerkprotokoll spezifiziert) oder auch externe, im System implementierte Maßnahmen (z. B. redundante Kommunikation, Checksummenkontrolle durch Hardware oder Software) handeln.

Den Maßnahmen gemeinsam ist die Tatsache, dass sie oft komplexe, systemübergreifende Funktionen darstellen. Die Methode soll in der Lage sein, fehlererkennende und fehlervermeidende Funktionalitäten des Systems in die Bewertung zu integrieren.

Schlussfolgerung

Um eine aussagekräftige Methode zur Analyse von potentiellen Netzwerkfehlern zu erhalten, müssen die Kriterien 1, 2 und 4 in jedem Fall erfüllt werden. Die Quantifizierung der Zuverlässigkeit ist wünschenswert, kann jedoch nicht von allen Methoden geleistet werden. Kriterium 5 ist notwendig, um eine möglichst realistische Darstellung der Fehlerauswirkungen in einem Netzwerk zu erhalten. Da jedoch oftmals Worst-Case-Annahmen gemacht werden, um möglichst konservative Abschätzungen zu erhalten, müssen solche automatischen Fehlerbehandlungsmaßnahmen nicht zwangsläufig implementiert werden. Sie dienen aber in jedem Fall einer Verbesserung der Zuverlässigkeit.

3.3 Modellierung eines Kommunikationsnetzwerks in einem generischen Leittechniksystem

3.3.1 Allgemeine Aspekte

Für die generische Analyse des Kommunikationsnetzwerkes eines Leittechniksystems ist es erforderlich, dass die Funktionen und Schnittstellen des Leittechniksystems zum Kommunikationsnetzwerk sowie die zu erfüllenden leittechnischen Funktionen des Gesamtsystems umrissen werden. In diesem Kapitel wird nun zunächst das von der GRS in diesem Projekt entwickelte, generische, digitale Leittechniksystem vorgestellt. In einem weiteren Schritt wird dann die Modellierung der Netzwerkfehler vorgenommen.

Das gewählte generische, digitale Leittechniksystem soll die Umsetzung einer aktiven leittechnischen Funktion der Kategorie A (Reaktorschutzfunktion) im Kernkraftwerk implementieren. Die genaue Funktion spielt für die Zuverlässigkeitsanalyse des Kommunikationsnetzwerks keine Rolle, ein Beispiel könnte aber das Starten einer Sicherheitseinspeisepumpe (in einer verfahrenstechnischen Redundanz) bei Ansprechen der Notkühlkriterien sein.

Eines dieser Leittechniksysteme besteht selbst aus redundanten (und z. T. auch zueinander diversitären) Strängen bzw. leittechnischen Redundanzen. Das gewählte System soll drei solcher Stränge aufweisen, welche identisch aufgebaut sind, d. h. keine Diversität aufweisen.

In einem Strang des Leittechniksystems kann man grundsätzlich zwei Informationspfade unterscheiden, den Automatisierungs- oder Auslösepfad und den Meldepfad.

Im Automatisierungs- oder Auslösepfad wird die o. g. leittechnische Funktion ausgeführt. Dies geschieht wie folgt (siehe auch KTA 3501 /KTA 14/ und Abbildung 3.2):

- Prozessvariablen werden gemessen und in für das Leittechniksystem verwertbare Signale gewandelt. Beispielsweise wird ein Spannungssignal der Differenzdruckmessung am Primärkreis in einem Messumformer in ein Signal einer Stromschnittstelle (4-20 mA) umgewandelt.

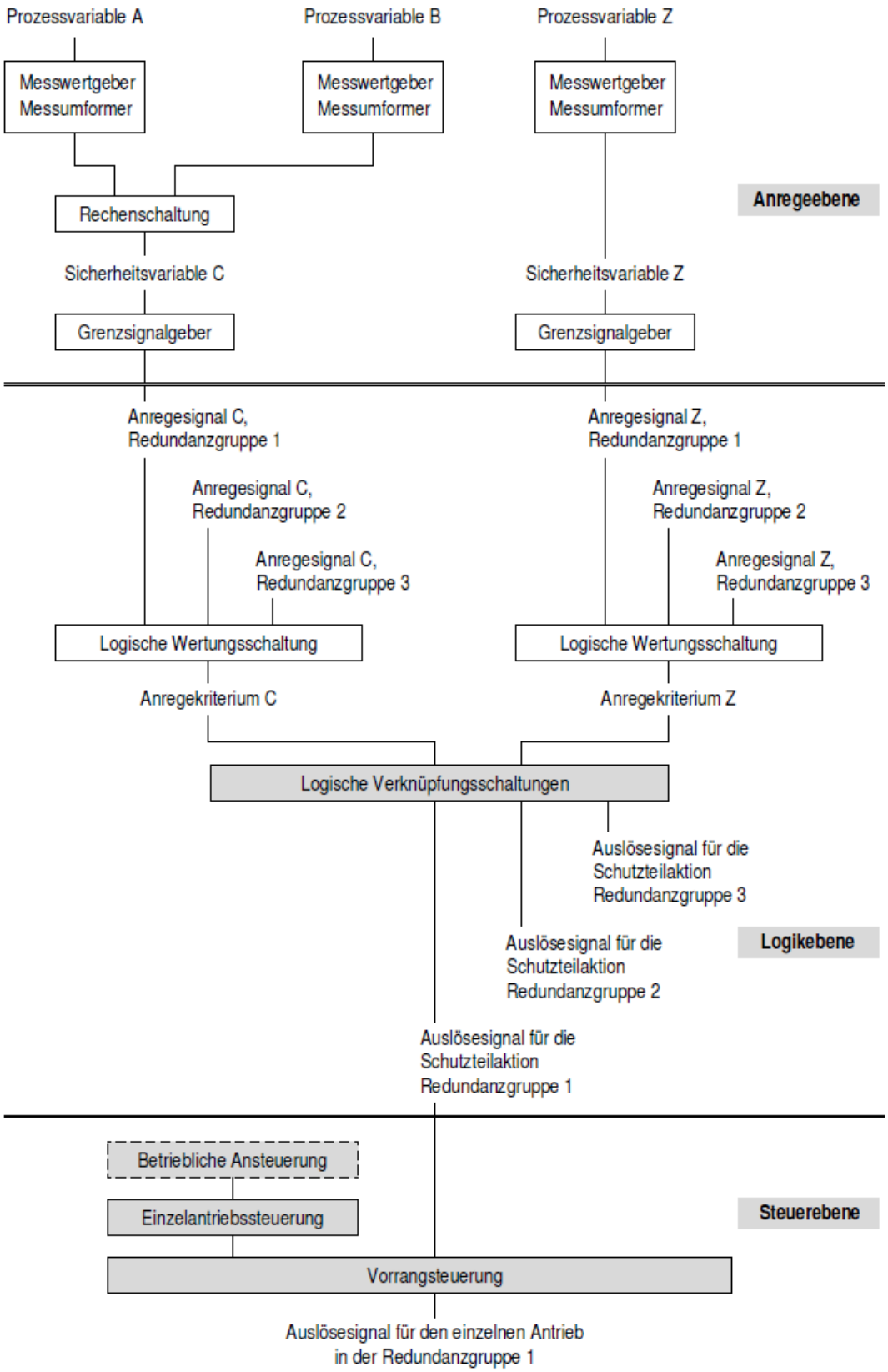


Abb. 3.2 Zuordnung von Begriffen zum funktionellen Aufbau von A-Funktions-Einrichtungen /KTA 14/

- Die Prozessvariable wird in eine Sicherheitsvariable umgewandelt und diese mit den definierten Grenzwerten verglichen. Bei Verletzung der Grenzwerte wird ein Anregesignal ausgegeben. Beispielsweise wird ein Stromsignal von einer A/D-Baugruppe in ein Datentelegramm, welches den entsprechenden numerischen Druckwert (Sicherheitsvariable) enthält, umgewandelt; ein Programm vergleicht dann diesen Wert mit dem Grenzwert und sendet ggf. ein Anregesignal: „Primärkreisdruck tief“. Dieser Prozess wird hier mit Datenverarbeitung und -auswertung (acquisition and processing) bezeichnet.

- Es wird zuerst eine logische Wertung des Anregesignals aus dem Strang mit den Anregesignalen aus den anderen Strängen durchgeführt. Danach erfolgt eine logische Wertungsschaltung mit den Anregesignalen aus anderen Prozess- und Sicherheitsvariablen (z. B. eine 2-von-3-Schaltung mit den Anregesignalen „Druck im Containment hoch“ und „DH-Füllstand tief“ für den Fall der Notkühlkriterien). Beide Schaltungen werden im Weiteren auch als Voting bezeichnet.

- Das Auslösesignal wird an die (Vorrangbaugruppen der) aktiven Sicherheitseinrichtungen der entsprechenden verfahrenstechnischen Redundanz gesendet (z. B. Senden des Startsignals für die HD-Einspeisepumpe beim Anstehen von 2 der 3 Notkühlkriterien).

Um die leittechnischen Funktionen in einem rechnerbasierten Leittechniksystem zu gewährleisten, ist es notwendig, dass die Kommunikation zwischen den Rechnereinheiten gemäß der Spezifikation arbeitet. Dies beinhaltet im Automatisierungspfad die folgenden Aufgaben des Kommunikationsnetzwerks:

- Übermittlung der Telegramme in der spezifizierten Zeit,

- Überprüfung der Integrität der übermittelten Telegramme.

Im Meldepfad werden verschiedenste Zwischenergebnisse und Meldungen, die im Automatisierungspfad erzeugt werden, wie z. B. Ein- und Ausgabewerte an den Votern, die analogen und digitalen Zwischenwerte bei der Signalverarbeitung, aber auch Statusmeldungen der verschiedenen Baugruppen verarbeitet und weitergeleitet.

Die typischen Zykluszeiten eines Sicherheitsleittechniksystems liegen im Bereich von 50 ms. Die Anzahl solcher Statusmeldungen in einem Leittechniksystem ist demnach sehr groß.

Um diese Menge an Meldungen verarbeiten zu können sowie eine sinnvolle Zusammenstellung für die Weiterverarbeitung (z. B. zur Anzeige auf einem Melderechner oder einem qualifizierten Display in der Warte) zu ermöglichen, werden sogenannte Melde- und Serviceinterfaces (MSI) benutzt. Diese stellen die Schnittstelle zu den weiteren Systemen der Anlage, wie der betrieblichen Leittechnik, der Warte und den örtlichen Leitständen sowie Servicegeräten zur Wartung und Modifizierung der Parameter der Automatisierungsebene dar.

Die Aufgaben des Kommunikationsnetzwerks hinsichtlich des Meldepfades sind in Tabelle 3.1 aufgeführt.

3.3.2 Beschreibung des generischen Modells eines digitalen Sicherheitsleittechniksystems

Das für dieses Projekt entwickelte generische Modell eines digitalen Sicherheitsleittechniksystems (DSLS) soll im Weiteren vorgestellt werden. Dazu werden zunächst der Automatisierungspfad und danach der Meldepfad erläutert. Die von dem DSLS ausgeführte leittechnische Funktion soll, wie oben beschrieben, eine der Kategorie-A-Funktionen (Reaktorschutzfunktion) im Kernkraftwerk sein, z. B. das Starten des HD-Einspeisesystems beim Anstehen von 2 von 3 Notkühlkriterien.

Die Modellierung des DSLS wird auf unterschiedlichen Abstraktionsebenen durchgeführt. Komponenten, welche Kommunikationsfunktionen (Funktionen, die unmittelbar zur Erfüllung der Aufgaben des Kommunikationsnetzwerks dienen) realisieren, werden auf der Ebene von Baugruppen modelliert. Alle übrigen Komponenten des Leittechniksystems werden auf der Ebene von Rechenschränken dargestellt.

In Abbildung 3.3 wird eine leittechnische Redundanz dargestellt, wobei insbesondere auf die netzwerktechnische Verschaltung eingegangen wird. Links in der Abbildung ist der Automatisierungspfad, rechts der Meldepfad eingezeichnet. Die Instrumentierung und Messwertumformung im Automatisierungspfad sowie die Vorrangschaltung und alle nachstehenden Systeme werden als vollständig analoge Systeme modelliert. Die Signalübermittlung findet über eine Drahtverbindung (als Messstromkreis) statt. Insbesondere findet hier keine Netzwerkkommunikation statt.

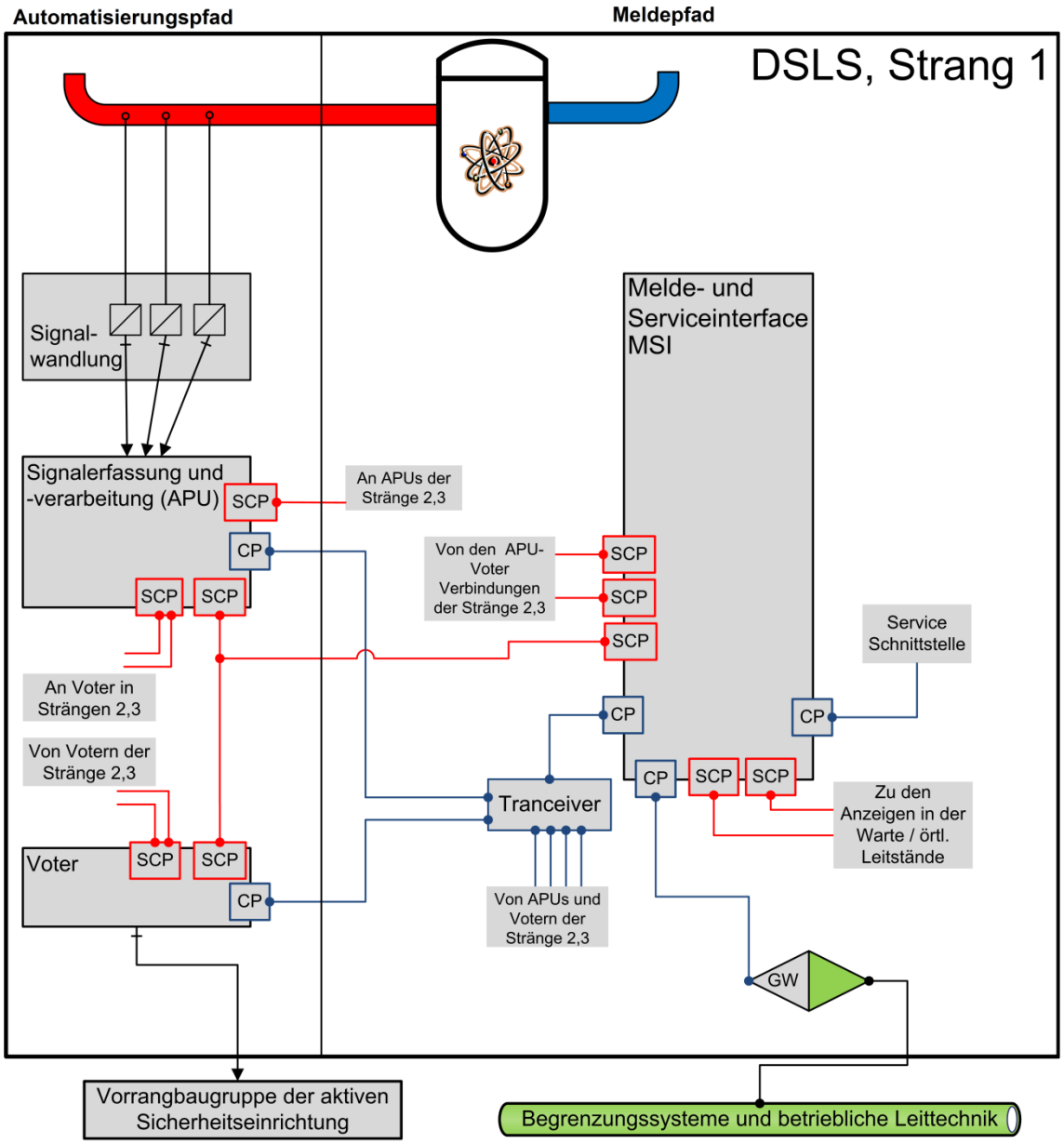


Abb. 3.3 Netzwerkaufbau einer leittechnischen Redundanz

Rot: Sicherheitsnetzwerk (SCP), blau: sicherheitsrelevantes Netzwerk (CP).

Die Signalerfassungs- und Signalverarbeitungseinheit (APU) beinhaltet Module, welche die analogen Signale in für rechnerbasierte Systeme lesbare Signale umwandeln (z. B. A/D-Wandler). Das rechnerbasierte System übernimmt dann die weitere Signalverarbeitung (Berechnung der Sicherheitsvariablen, Grenzwertvergleich, Ausgabe eines Anregesignals). Die Kommunikation innerhalb der APU findet über einen Rückwandbus statt. Der Versand des Anregesignals an die Logik wird mithilfe eines Sicherheitsnetzwerks (SCP) realisiert, wobei nur Punkt-zu-Punkt-Verbindungen existieren.

Die Logik (logische Wertungs- und Verknüpfungsschaltung) wird in einer separaten Einheit (Voter) realisiert. Intern kommuniziert der Voter, wie die APU, über einen Rückwandbus. Das dabei gebildete Auslösesignal für die Schutzaktion der verfahrenstechnischen Redundanz wird nach einer Entkopplung wieder als analoges Signal an die Vorrangbaugruppe gesendet.

Die Umsetzung der Kommunikationsfunktionen im Meldepfad wird zum größten Teil über das MSI realisiert. Das MSI ist ein vom Automatisierungspfad unabhängiges Rechnersystem, welches aus einer großen Anzahl an Schnittstellen (Kommunikationskarten) zum Sicherheitsnetzwerk (SCP) und dem sicherheitsrelevanten Netzwerk (CP) sowie Verarbeitungskarten besteht (siehe Abbildung 3.3 rechts).

Die in Tabelle 3.1 aufgeführten Kommunikationsaufgaben werden im DSLS durch die sicherheitsrelevanten Netzwerke bzw. die Sicherheitsnetzwerke realisiert. Dort wird jede Kommunikationsaufgabe Baugruppen zugeordnet, die diese Funktionen ausführen. Auf die datentechnische Vernetzung dieser Netzwerke wird im Folgenden eingegangen.

Tab. 3.1 Kommunikationsaufgaben der im MSI vorhandenen Baugruppen

Kommunikationsaufgabe	Verantwortliche Baugruppe
M1. Empfangen und Zwischenspeichern von Telegrammen aus dem Automatisierungspfad.	Sicherheitsnetzwerk, SCP-Baugruppen
M2. Auslesen und Verarbeiten der für die im Weiteren angeschlossenen Systeme relevanten Informationen	Interne Schrankkommunikation
M3. Empfangen und Zwischenspeichern von Telegrammen, welche die Statusmeldungen (Selbstüberwachungssignale) der Module aus dem Automatisierungspfad enthalten	Sicherheitsrelevantes Netzwerk, CP-Baugruppen
M4. Verarbeiten der Informationen dieser Statusmeldungen	Interne Schrankkommunikation
M5. Senden der verarbeiteten Daten an die im Weiteren angeschlossenen Systeme, wie z. B. analoge und digitale Anzeigeeinrichtungen in der Warte und der Notsteuerstelle, den örtlichen Leitständen, weiteren leittechnischen Systemen (Begrenzungen und betriebliche Leittechnik)	Sicherheitsnetzwerk, SCP-Baugruppen für Kommunikation zur Warte/Notsteuerstelle/örtliche Leitstände Sicherheitsrelevantes Netzwerk, CP-Baugruppen für Kommunikation zu den Begrenzungssystemen und der betrieblichen Leittechnik
M6. Bereitstellen einer Schnittstelle zum Konfigurieren und Parametrieren der Module im Automatisierungspfad	Sicherheitsrelevantes Netzwerk, CP-Baugruppen

3.3.3 Beschreibung der Netzwerktechnologien im Sicherheitsnetzwerk und im sicherheitsrelevanten Netzwerk

In diesem Kapitel wird die datentechnische Vernetzung des generischen Modells des DSLS erläutert. Wie bereits im letzten Kapitel angedeutet, werden drei verschiedene Verbindungstypen verwendet. Es gibt analoge, sogenannte hartverdrahtete Verbindungen (z. B. zwischen Messinstrumentierung und Messumformer), welche als Messstromkreis arbeiten und keine echte datentechnische Verbindung darstellen. Sie sind nur der Vollständigkeit halber im Modell aufgeführt und haben für die weitere Betrachtung keine Relevanz. Insbesondere werden keine Fehler oder Ausfälle in diesen Verbindungen postuliert.

Des Weiteren gibt es Verbindungen, welche als Sicherheitsnetzwerk (SCP) oder sicherheitsrelevantes Netzwerk (CP) ausgeführt sind. Auf die technologischen Aspekte und die Konfiguration dieser Netzwerktypen soll im Folgenden eingegangen werden. Der prinzipielle Unterschied zwischen beiden Netzwerken ist in den Anforderungen hinsichtlich der Zuverlässigkeit des Netzwerkbetriebs zu sehen. Während die Netzwerkkommunikation des Sicherheitsnetzwerks auch die umfangreichsten Anforderungen erfüllen muss, ist das sicherheitsrelevante Netzwerk als unterstützend für den sicheren Betrieb zu erachten. Dementsprechend können hier abgestufte Anforderungen an die Zuverlässigkeit zugrunde gelegt werden.

3.3.3.1 Eigenschaften des Sicherheitsnetzwerks (SCP)

Das Sicherheitsnetzwerk im generischen Modell des DSLS ist mithilfe der industriellen Feldbustechnologie „PROFIBUS“ realisiert. Für einführende Informationen zu Feldbussen und PROFIBUS, siehe Kapitel 2.1.3.

Die Begriffe „Feldbus“, „PROFIBUS“, „Buszugriff“ usw. sind branchenspezifische Begriffe, die sich etabliert haben und nicht unbedingt die Topologie des Netzwerks berücksichtigen. Es ist durchaus möglich, dass ein Netzwerk mit Profibustechnologie rein aus Punkt-zu-Punkt-Verbindungen besteht und keine Bustopologie zum Einsatz kommt. In diesem Fall spezifiziert der Name PROFIBUS nur das Netzwerkprotokoll. Dies ist beispielsweise im Sicherheitsnetzwerk der Fall.

Im Übersichtsbild des DSLS, Abbildung 3.3, sind die durch das Sicherheitsnetzwerk realisierten Verbindungen durch rote Linien dargestellt und die entsprechenden Kommunikationsbaugruppen mit dem Baugruppennamen „SCP“ versehen.

Zur Realisierung von PROFIBUS-Netzwerken in realen Systemen werden oft mehrere Baugruppen verwendet. Im Leittechniksystem „TELEPERM XS“ sind z. B. ein Verarbeitungsmodul („SVE“) und ein darauf aufgestecktes Kommunikationsmodul („SL“) notwendig, um eine Schnittstelle zu einem elektrisch ausgeführten PROFIBUS-Netzwerk bereitzustellen. Für eine Schnittstelle zu einem optisch ausgeführten PROFIBUS-Netzwerk ist sogar ein weiteres Linkmodul (Optokoppler „SLM“) notwendig. In dem für dieses Projekt entwickelten Modell soll ein solcher Detaillierungsgrad nicht angestrebt werden. Alle Funktionalitäten, die die Module im o. g. Beispiel leisten, sollen sich in diesem Modell auf der SCP-Baugruppe befinden.

Die Konfiguration des Sicherheitsnetzwerks implementiert eine zyklische und deterministische Kommunikation der Teilnehmer des Netzwerks. Darüber hinaus ist der Informationsfluss strikt unidirektional von der Quelle zur Zielkomponente. In Tabelle 3.2 sind alle in einem Strang modellierten Verbindungen aufgelistet.

Tab. 3.2 Verbindungen, die durch das Sicherheitsnetzwerk hergestellt werden

Quelle	Ziel	Inhalt der Telegramme
APU	APU (andere Stränge)	Sicherheitsvariable, Anregesignal
APU	Voter	Anregesignal
APU	Voter (andere Stränge)	Anregesignal
APU	MSI	Sicherheitsvariable, Anregesignal
APU (andere Stränge)	MSI	Sicherheitsvariable, Anregesignal
APU (andere Stränge)	Voter	Anregesignal
Voter	MSI	Anregesignal, Auslösesignal
MSI	Qualified Displays (QDS) in der Warte, der Notsteuerstelle und örtl. Leitständen	Sicherheitsvariable, Anregesignal, Auslösesignal, Statusmeldungen APU/Voter

Alle Verbindungen aus Tabelle 3.2 sind Punkt-zu-Punkt-Verbindungen zwischen Quelle und Ziel. Eine zufällige Blockade der Kommunikationsverbindung durch andere Teilnehmer kann somit für diese Verbindungen ausgeschlossen werden. Die physikalische Übertragung soll über Lichtwellenleiter erfolgen, um elektromagnetische Einflüsse bei

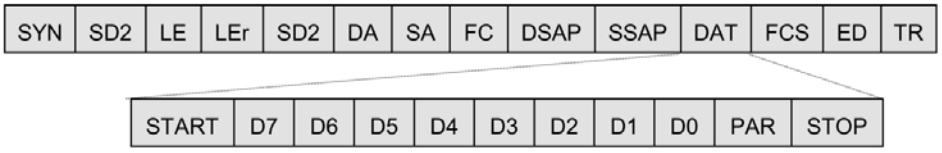
der Übertragung ausschließen zu können sowie zur Herstellung einer galvanischen Trennung zwischen den Komponenten.

Der Ablauf einer Kommunikation ist durch die Topologie mit dedizierten Punkt-zu-Punkt-Verbindungen sehr einfach: Im Auslöse- und Meldepfad wird ein unidirektionaler Kommunikationsvorgang von der jeweiligen Quelle zum Ziel (siehe Tabelle 3.2) in jedem Zyklus (typischerweise alle 50 ms) durchgeführt. Die Kommunikation erfolgt dabei derart, dass der Sender Telegramme über das Netzwerkmedium sendet, ohne sich zu vergewissern, ob der Empfänger empfangsbereit ist. Dieses Verhalten ist konform zur geforderten Rückwirkungsfreiheit der Kommunikation im Sicherheitsnetzwerk und erhöht zusätzlich den Datendurchsatz im System.

Trotz der hier einfach gehaltenen Strukturen wäre es möglich, ein PROFIBUS DP Netzwerk auch in anderen Topologien zu betreiben. Dabei stehen bei einer elektrischen Signalübertragung Linien- und Baum-Topologien, bei der Übertragung per Lichtwellenleiter Linien-, Ring- und Stern-Topolgien zur Auswahl. Die Konfiguration des Netzwerks, d. h. die Telegrammflusssteuerung sowie die Sicherungsmaßnahmen, sind im Protokoll „Field Data Link“ (FDL) festgelegt, welches auf Schicht 2 des OSI-Modells angelegt ist.

Mechanismen zur Fehlerdetektion und Fehlerbehebung sind im Sicherheitsnetzwerk auf vielen Ebenen implementiert. Diese lassen sich entsprechend ihrer Implementierung in den verschiedenen OSI-Schichten unterscheiden.

Ein Datentelegramm im PROFIBUS-Netzwerk hat grundsätzlich den in Abbildung 3.4 dargestellten Aufbau.



SYN	Busruhe	FC	Funktionscode	START	Startbit
SD2	Start-Delimiter 2	DAT	Datenbits	D7…D0	Datenbits
LE	Längenbyte	FCS	Quersumme	PAR	Parität
LEr	Längenbyte-Wiederholung	ED	Ende-Delimiter	STOP	Stopbit
DA	Zieladresse	TR	Mindestverzögerung	DSAP	Destination Service Access Point
SA	Quelladresse		(8 bit-Zeiten)	SSAP	Source Service Access Point

Abb. 3.4 Aufbau eines Datentelegramms mit prinzipiell variabler Länge im PROFIBUS-Netzwerk

Das gezeigte Beispieltelegramm überträgt 8 Datenbits /PRO 13/.

Eine Sicherungsmaßnahme, welche die gesamten Telegrammdaten prüft, ist die „frame check sequence“ (FCS, auf Schicht 2 des OSI-Modells). Dabei wird vom Sender die Quersumme aller Datenbits des Telegramms gebildet. Der Empfänger berechnet ebenfalls die Quersumme und vergleicht sie mit dem vom Sender in das Telegramm eingetragenen Wert. Eine darüber hinaus im FDL verankerte Sicherungsmaßnahme ist die Angabe des Längenbytes sowie dessen Wiederholung.

Die höchste Protokollschicht (Schicht 7 im OSI-Modell, zwischen START- und STOP-Bit) enthält ein Paritätsbit, welche die Parität der übermittelten Daten (D0 bis D7 in Abbildung 3.4) zählt. Ist die Anzahl der mit „1“ belegten Bits eine gerade Zahl, so ist das Paritätsbit „1“ und im ungeraden Fall „0“. Mit dieser Maßnahme ist es möglich, die fehlerhafte Übertragung von Datenbits zu erkennen. Diese Methode deckt jedoch nicht alle Bitfehler ab, da nur eine ungerade Anzahl an fehlerhaften Datenbits erkannt werden kann.

Darüber hinaus wird den Telegrammen eine laufende Nummer zugewiesen. Mit dieser Maßnahme kann die Empfangsbaugruppe feststellen, ob ein Telegramm irrtümlich in den Datenverkehr gekommen ist, bzw. ob Telegramme nicht bei der Baugruppe angekommen sind.

Zählt man diese Maßnahmen zusammen, so erreicht dieses Verfahren eine minimale Hamming-Distanz (siehe Kapitel 2.1.1) /ITS 96/ von 4, d. h. dass bis zu 3 Bitfehler pro Kommunikationsvorgang vom Empfänger detektiert werden können.

In dem Fall, dass ein ungültiges Telegramm detektiert wird, ist typischerweise vorgesehen, dass dieses Telegramm verworfen wird. Aufgrund der hohen Telegrammrate wird die Leittechnik-Funktion des Empfängers durch das Ignorieren eines fehlerhaften Telegramms nicht beeinflusst. Bei der Datenkommunikation eines auf dem TELEPERM XS basierten Leittechniksystems kann z.B. die Fehlertoleranz so implementiert werden, dass 3-5 fehlerhafte Telegramme in Folge vom Empfänger ignoriert werden können, ohne dass die betroffene Leittechnikfunktion dadurch beeinflusst wird. Dies soll auch für das generische Modell des DSLS gelten.

Um die ungestörte Funktion des Gesamtsystems (Leittechnik-Modell DSLS) im Fall einzelner fehlerhafter Telegramme (weniger als 3) zu gewährleisten, werden als fehlerhaft übertragen erkannte Telegramme von der Kommunikationsbaugruppe verworfen und durch vordefinierte Standardtelegramme ersetzt. Damit wird das Ansprechen der

im weiteren Verlauf eingebauten Überwachungsmaßnahmen vermieden und ein Meldeschwall verhindert.

3.3.3.2 Eigenschaften des sicherheitsrelevanten Netzwerks (CP)

Das sicherheitsrelevante Netzwerk im generischen Modell des DSLS ist mithilfe des industriellen Feldbusses „PROFInet“ realisiert. Für einführende Informationen sei auf Kapitel 2.1.3.2 verwiesen. In Abbildung 3.3 sind die durch das sicherheitsrelevante Netzwerk realisierten Verbindungen durch blaue Linien dargestellt und die entsprechenden Kommunikationsbaugruppen mit dem Baugruppennamen „CP“ versehen.

PROFInet ermöglicht die Anwendung mehrerer Modi für den Betrieb des Netzwerks. Im generischen Modell des DSLS sollen zwei dieser Modi Anwendung finden:

- Der Realtime-Modus (RT) ermöglicht die zyklische Kommunikation von Komponenten, wobei 10 ms Zykluszeiten erreicht werden können. Tatsächlich wird hier aber eine Zykluszeit von 50 ms, also genau wie im Sicherheitsnetzwerk, verwandt. Dieser Modus soll für die Kommunikation der Statusdaten zwischen den Komponenten des Automatisierungspfades (APU und Voter aus allen Redundanzen) und dem MSI (vgl. Aufgabe M3 aus Tabelle 3.1) sowie für die Weiterleitung der relevanten Informationen an weitere Automatisierungsnetze (vgl. Aufgabe M5 aus Tabelle 3.1) benutzt werden.
- Der Standard-Modus ermöglicht die azyklische bzw. spontane Kommunikation einer Vielzahl von Komponenten. Dieser Modus wird für das Parametrieren von Feldgeräten (Sensoren, Aktoren,…) eingesetzt und soll hier zur Erfüllung der Aufgabe M6 aus Tabelle 3.1 („Bereitstellen einer Schnittstelle zum Konfigurieren und Parametrieren der Module im Automatisierungspfad“) genutzt werden. Des Weiteren werden über azyklische Kommunikation Alarmdaten, z. B. bei Alarmierung eines Watchdogs, mit Hilfe dieses Modus übermittelt.

Beide Modi bedienen sich der Ethernet-Protokollfamilie, d. h. die physikalische Schicht (Schicht 1 im OSI-Modell) wird gemäß dem Ethernet-Protokoll 100BASE-TX (für eine elektrische Datenübermittlung im Vollduplexbetrieb mit 100 Mbit/s, siehe Kapitel 2.1 zur Erläuterung) betrieben. Die Sicherungsschicht (Schicht 2 im OSI-Modell) ist ebenfalls uniform durch die Standards Logical Link Control (LLC, IEEE 802.2) und Media Access Control (MAC, IEEE 802.3) definiert. In Abbildung 3.5 ist eine Einordung der beiden

Modi in das OSI-Schichtenmodell dargestellt. Dort werden folgende Abkürzungen verwendet:

- ASIC: Application Specific Integrated Circuits. Ethernet-Controller für Echtzeitkommunikation,
- IP: Internet Protocol,
- IRT: „Isochronous Real Time“-Datenaustausch mit Taktsynchronität,
- RT: „Real Time“-Datenaustausch ohne Taktsynchronität,
- TCP/UDP: Transmission Control Protocol/User Datagram Protocol. Während TCP Bestätigungen beim Datenempfang sendet, verzichtet UDP darauf.

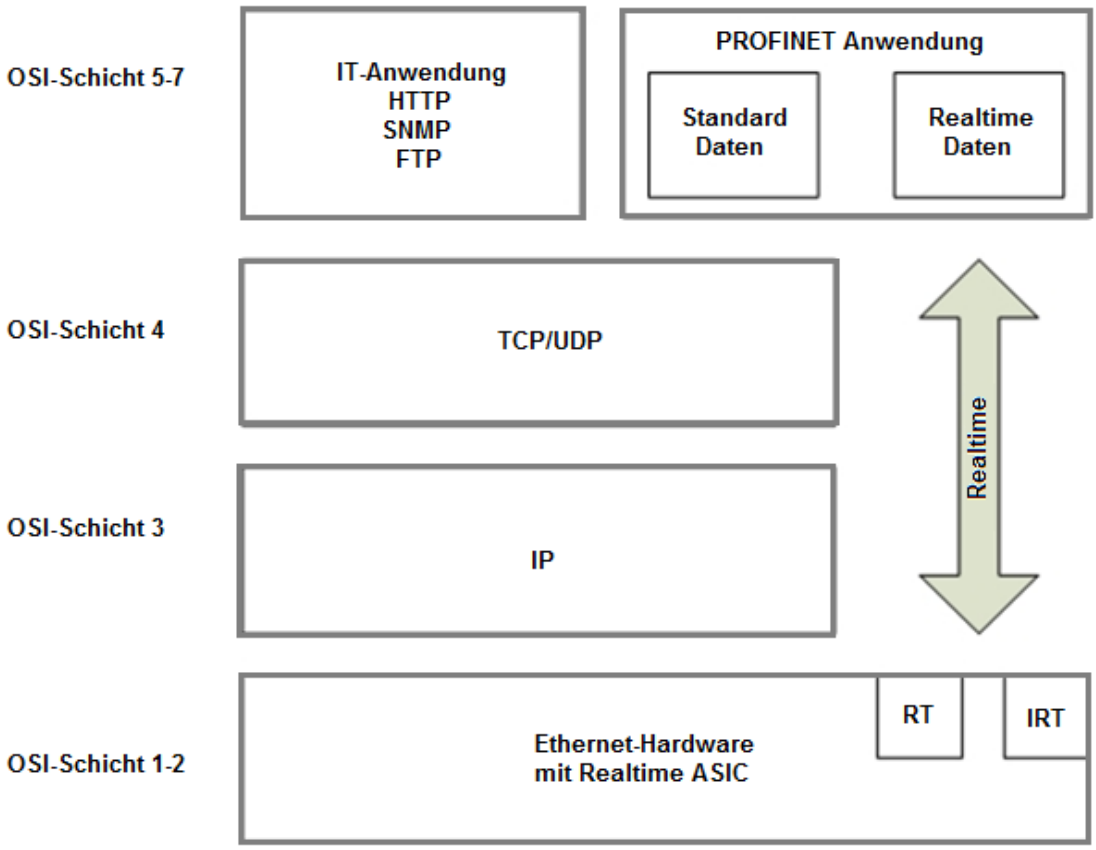


Abb. 3.5 Einordnung der Profinet Modi „Standard“ und „Realtime“ in das OSI-Schichtenmodell /BOH 06/

Obwohl mit PROFInet eine Vielzahl von Topologien (z. B. Stern-, Baum- oder Linien-Topologie) realisiert werden können, wird im generischen Modell des DSLS angenom-

men, dass alle Verbindungen zwischen den Komponenten des sicherheitsrelevanten Netzwerks Punkt-zu-Punkt-Verbindungen sind.

Die beiden Modi unterscheiden sich ab der Transportschicht (Schicht 4 im OSI-Modell): der Standard-Modus basiert auf der standardisierten TCP/IP-Familie und ist somit kompatibel mit sehr vielen Funktionen und Applikationen aus der Bürokommunikationswelt, wie z. B. Webservern, Browsern etc. Die Anwendung des TCP-Protokolls impliziert eine verbindungsorientierte Kommunikation, d. h. ein Senden und Empfangen von Daten ist nur bei vorherigem „handshake“ (siehe Erläuterung in Kapitel 2.1.1) möglich. Im Gegensatz dazu ist die Kommunikation im Realtime-Modus verbindungslos. Hier erfolgt der Datenaustausch ohne TCP/IP-Informationen direkt auf Schicht 2 des OSI-Modells.

Zusätzlich zur Kollisionsüberwachung CSMA/CD wird beim Realtime-Modus zur Priorisierung einzelner Telegramme ein sogenannter VLAN-Tag (nach IEEE 802.1q standardisiert) verwendet /BHM 12/, wobei hier bis zu 8 Priorisierungsstufen definiert werden können.

Eine Übersicht über den Aufbau eines solchen Telegramms ist in Abbildung 3.6 gezeigt. Die wichtigen Bestandteile sind Quell- und Zielbyte (DestAddr, SrcAddr), Nutzdaten (Data), Telegrammnummer (Cycle) und frame check sequence (FCS).

Dest Addr	Src Addr	Eth. Type + VLAN	Ether Type	Frame ID	Data	IOPS	…	IOCS	…	Cycle	Data Sts	X Sts	FCS
6	6	4	2	2	1..	1		1		2	1	1	4

Abb. 3.6 Aufbau eines PROFInet-Telegramms in Realtime-Modus /FRA 09/

Die mit dem sicherheitsrelevanten Netzwerk hergestellten Verbindungen sowie der Inhalt der in den Verbindungen übertragenen Telegramme sind in Tabelle 3.3 zusammengefasst. Die angegebenen Verbindungen sind im Allgemeinen bidirektional.

Wie schon oben erwähnt, handelt es sich bei allen Verbindungen um Vollduplexverbindungen, d. h. eine Kommunikation zwischen den Komponenten in beide Richtungen ist zu jeder Zeit möglich. Der typische Fall ist jedoch, dass die zyklische Kommunikation wie bei dem Sicherheitsnetzwerk unidirektional versendet wird.

Tab. 3.3 Durch das sicherheitsrelevante Netzwerk hergestellte Verbindungen

Quelle	Ziel	Inhalt der Telegramme
APU	MSI (via Tranceiver)	Sicherheitsvariable, Anregesignal, Statusdaten, Alarm
APU (andere Stränge)	MSI (via Tranceiver)	Sicherheitsvariable, Anregesignal, Statusdaten, Alarm
Voter	MSI (via Tranceiver)	Anregesignal, Auslösesignal, Statusdaten, Alarm
Voter (andere Stränge)	MSI (via Tranceiver)	Anregesignal, Auslösesignal, Statusdaten, Alarm
MSI	Gateway	Anregesignal, Auslösesignal, Statusdaten
Service Interface	MSI	Parametrierdaten
MSI	APU, Voter	Parametrierdaten

Darüber hinaus wird die azyklische Kommunikation der Komponenten im Standardmodus (z. B. das Absetzen einer Alarmmeldung oder das Aufspielen einer neuen Parametrierung von einem Servicerechner) parallel auf das Medium zugreifen.

Auch im sicherheitsrelevanten Netzwerk gibt es an unterschiedlichen Stellen Mechanismen zur Fehlerdetektion. Auf der höchsten Schicht des OSI-Modells (Schicht 7) werden, genau wie im Sicherheitsnetzwerk, je 8 Datenbits mit einem Paritätsbit abgesichert. Darüber hinaus besitzen sowohl zyklische, als auch azyklische Telegramme eine FCS, welche in diesem Fall die von Ethernet standardisierte 32-bit Checksummen-Bildung (CRC) integriert. CRC umfasst dabei die gesamten Telegrammdaten. Diese Maßnahme ist der Schicht 2 des OSI-Modells zuzuordnen. Wie beim Sicherheitsnetzwerk gibt es für die Telegramme im Realtime-Modus eine Nummerierung der Telegramme.

3.3.3.3 Zusammenfassung der Eigenschaften des Sicherheitsnetzwerks und des sicherheitsrelevanten Netzwerks

Die wichtigsten Eigenschaften des Sicherheitsnetzwerks und des sicherheitsrelevanten Netzwerks sind in Tabelle 3.4 zusammengefasst. Diese werden auch im Netzwerkmodell umgesetzt.

Tab. 3.4 Vergleich des Sicherheitsnetzwerks und dem sicherheitsrelevantem Netzwerk im generischen Modell des DSLS

	Sicherheitsnetzwerk	**Sicherheitsrelevantes Netzwerk**	**Sicherheitsrelevantes Netzwerk**
Modulname im DSLS	SCP	CP	CP
Protokoll	Profibus DP (DIN 19245)	Profinet real time	Profinet Standard
Bruttodatendurchsatz	12 Mbit/s	100 Mbit/s	100 Mbit/s
Physikalische Schicht (OSI Schicht 1)	Lichtwellenleiter-Kabel	Ethernet 100BASE-TX (Twisted Pair Kupferkabel)	Ethernet 100BASE-TX (Twisted Pair Kupferkabel)
Sicherungsschicht (OSI Schicht 2)	Field Data Link (FDL)	Logical Link Control (LLC, IEEE 802.2) Media Access Control (MAC, IEEE 802.3)	Logical Link Control (LLC, IEEE 802.2) Media Access Control (MAC, IEEE 802.3)
Buszugriffsverfahren	Strikt zyklische Master-Master Komm. (logisches Token-Pass.)	CSMA/CD, VLAN-Tag	CSMA/CD
Netzwerkschicht (OSI-S. 3)	Nicht ausgeprägt	Nicht ausgeprägt	Internet Protocol (IP)
Transportschicht (OSI- S. 4)	Nicht ausgeprägt	Nicht ausgeprägt	Transmission Control Protocol (TCP)
Sitzungsschicht (OSI-S. 5)	Nicht ausgeprägt	Nicht ausgeprägt	Nicht ausgeprägt
Präsentationschicht (OSI-S. 5)	Nicht ausgeprägt	Nicht ausgeprägt	Nicht ausgeprägt
Anwendungsschicht (OSI-S. 7)	DP Anwendungsfunktionen	Profinet Application Layer	Profinet Application Layer
Zykluszeit	50 ms	50 ms	azyklische Kommunikation
Eingesetzte Netzwerkhardware	Kommunikationsmodule, Opto-koppler	Kommunikationsmodule, Transceiver, Switches, Gateways	Kommunikationsmodule, Transceiver, Switches, Gateways
Fehlerdetektionssysteme oder -mechanismen	Paritätsbit für 8 Datenbits, Frame check sequence (FCS, Quersummenbildung), Nummerierung der Telegramme	Paritätsbit für 8 Datenbits 32 bit CRC /IEE 12/ Nummerierung d. Telegramme	Paritätsbit für 8 Datenbits 32 bit CRC /IEE 12/
Minimale Hamming Distanz	4 /PRO 13/	3 (bzgl. des CRC-Checks)	3 (bzgl. des CRC-Checks)

Während es für das SCP ein Protokoll gibt, welches ausschließlich eine zyklische Master-Master-Kommunikation erlaubt, kann man beim CP zwischen zwei Protokollen mit zyklischer oder azyklischer Kommunikation wählen. Für die Systeme mit zyklischer Kommunikation beträgt in beiden Fällen die Zykluszeit 50 ms. Im SCP werden Lichtwellenleiterkabel eingesetzt, um eine elektrische Entkopplung der Signale zu gewährleisten. Im CP werden Kupferkabel eingesetzt. Der Datendurchsatz ist beim CP mit 100 Mbit/s deutlich höher als im SCP mit 12 Mbit/s.

In allen drei Varianten kommen zur Fehlerdetektion Paritätsbits für 8 Datenbits sowie Prüfsummen zum Einsatz. Während beim SCP als Prüfsumme eine Quersummenbildung erfolgt, wird bei CP die Prüfsumme aus einer Polynomfunktion ermittelt. Darüber hinaus werden die Telegramme im SCP und im CP mit Realtime-Modus nummeriert, um nicht ankommende oder in der Reihenfolge vertauschte Telegramme zu erkennen.

3.3.4 Beschreibung der Fehlermodellierung im DSLS

3.3.4.1 Einordnung der Fehleranalyse des Netzwerkssystems in den Zusammenhang eines Fehlermodells für das Leittechniksystem

Ziel dieses Vorhabens ist die Entwicklung eines Ansatzes zur Zuverlässigkeitsbewertung digitaler Netzwerktechnologien, insbesondere auch von Netzwerktopologien. Für eine Einordnung der Vorgehensweise in den Gesamtkontext soll dazu im Folgenden ein Modell zur Analyse von Fehlern und Ausfällen in Leittechniksystemen vorgestellt werden, in welches sich die Analyse der Netzwerkfehler eingliedert. Dabei orientiert sich das Fehlermodell zu großen Teilen an dem Modell, welches in /NEA 15/ beschrieben wird.

Zunächst sollen (Sicherheits-) Leittechnikfunktionen in den Kontext des gestaffelten Sicherheitskonzepts gestellt werden. Dabei ist zu beachten, dass einzelne Leittechnikfunktionen immer nur einen Teil des gestaffelten Sicherheitskonzepts darstellen und nur ein Mehrfachversagen verschiedener Leittechnikfunktionen (z. B. Regelungs-, Begrenzungs- und Reaktorschutzfunktionen) zu unzulässigen Anlagenzuständen führen kann.

In dem hier verwendeten Ansatz wird davon ausgegangen, dass Leittechnikfunktionen immer in einem gewissen Kontext zu betrachten sind. Dieser Kontext besteht aus den verfahrenstechnischen Randbedingungen, d. h. dem Anlagenzustand sowie gewissen einleitenden Ereignissen (siehe Abbildung 3.7).

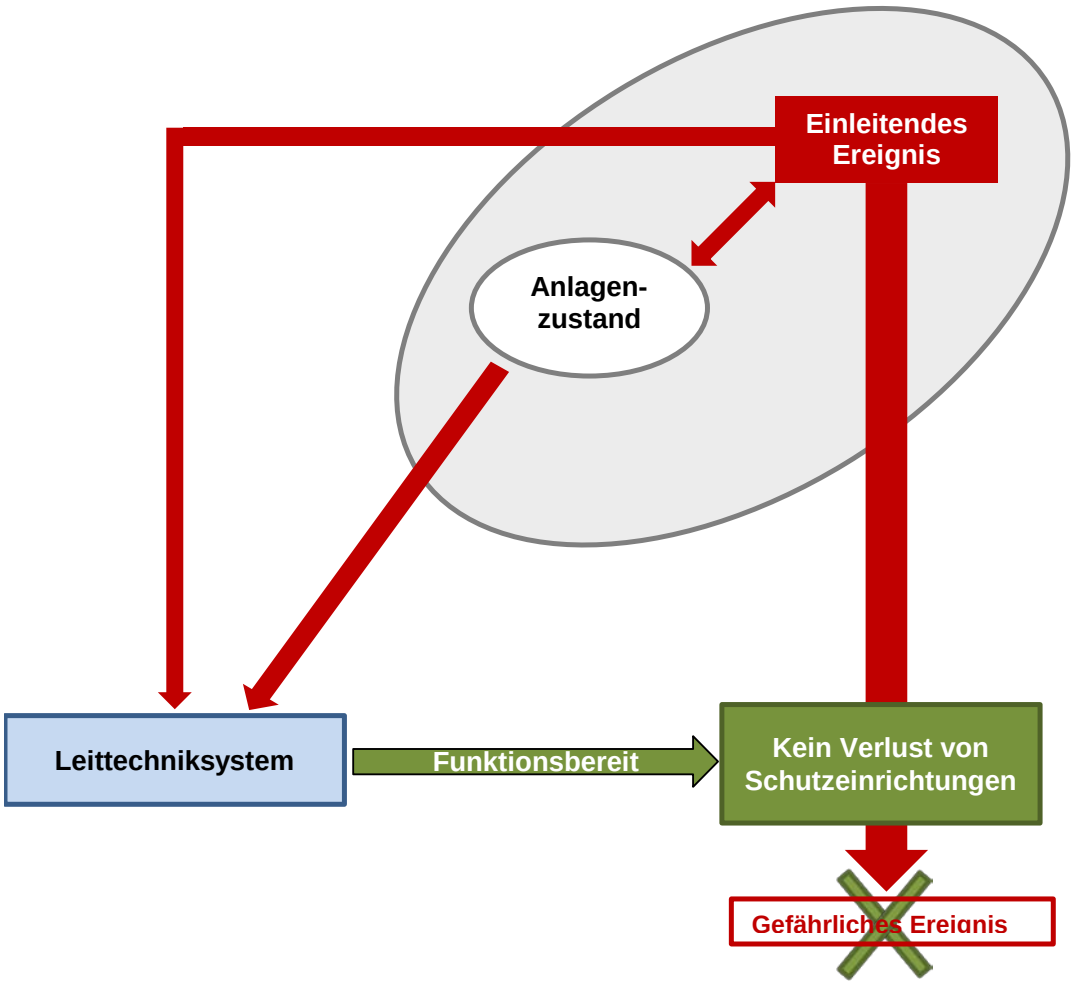


Abb. 3.7 Darstellung des Leittechniksystems und der Leittechnikfunktion

Darstellung des Kontext der Anlage in Bezug auf das gestaffelte Sicherheitskonzept /NEA 15/.

Bei auslegungsgemäßer Funktion des Leittechniksystems kommt es im Fall eines Abweichens des Anlagenzustands vom Sollzustand zu einer Ansteuerung von Komponenten, welche den Istzustand in den Sollzustand überführen sollen. Insbesondere soll der Übergang eines einleitenden Ereignisses zu einem auslegungsüberschreitenden Ereignis verhindert werden.

Im Fall einer Fehlfunktion in der Hard- oder Software des Leittechniksystems kann es dazu kommen, dass die Leittechnikfunktion ausfällt oder aktiviert wird, obwohl sie nicht angefordert wurde. Die begleitenden Umstände, welche hinreichend dafür sind, dass aus einer Fehlfunktion der Endeffekt (der Ausfall oder die fehlerhafte Aktivierung) folgt, nennt man Aktivierungsbedingungen. Eine Darstellung dieses Zusammenhangs wird in Abbildung 3.8 gezeigt.

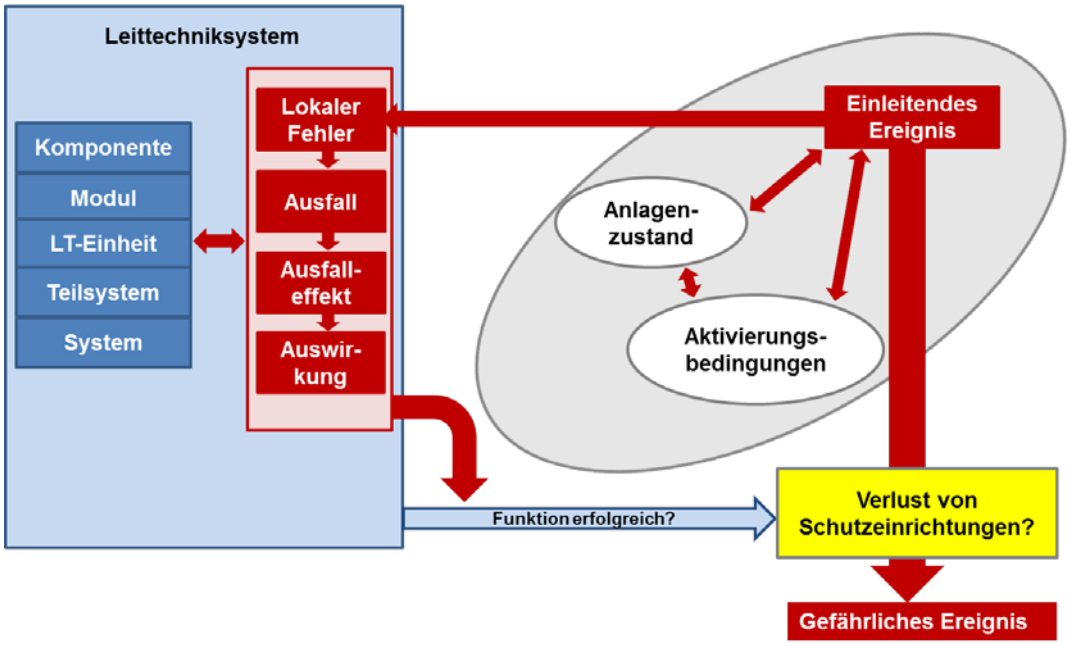


Abb. 3.8 Zusammenhänge zwischen einleitendem Ereignis, Aktivierungsbedingungen, Fehlfunktion und Ausfall der Leittechnikfunktion /NEA 15/

Der im Leittechniksystem in Abbildung 3.8 gezeigte Pfad von einem (lokalen) Fehler bis zum Endeffekt entspricht dem in Abbildung A.1 gezeigten Fehlerfortpflanzungsmodell. Mehr Informationen zu diesem Modell sind in Anhang A.1 und in /NEA 15/ zu finden.

Ein wichtiges Merkmal, nach dem man das Versagen von Leittechnikfunktionen klassifizieren kann, ist der Umstand, wie das Versagen erkannt wird. In /NEA 15/ wird die Erkennung eines Fehlers in die folgenden Kategorien unterschieden:

- Der Ausfall wird ohne eine Anforderung an eine Leittechnikfunktion erkannt
  - Der Ausfall wird durch eine Erkennungsmaßnahme im Leittechniksystem erkannt

- Der Ausfall wird durch eine fehlerhafte Aktivierung einer Leittechnikfunktion (bzw. einer Komponente im Leittechniksystem) erkannt

- Der Ausfall wird bei Anforderung erkannt.

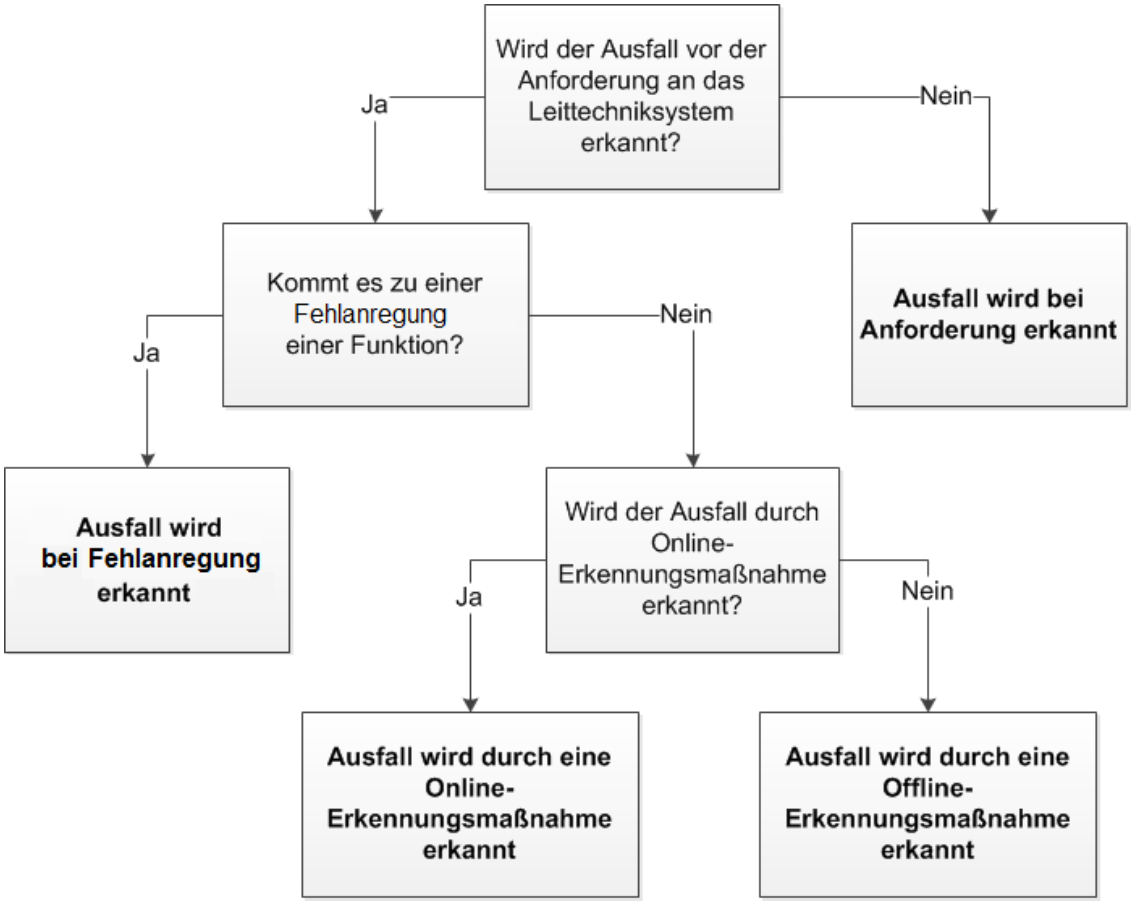


Abb. 3.9 Unterscheidung der verschiedenen Kategorien bei der Erkennung von Ausfällen bei Leittechnikfunktionen /NEA 15/

Wird ein Ausfall vor der Anforderung erkannt (siehe Abbildung 3.9), so unterscheidet man weiter in Online- und Offline-Erkennung. Die Online-Erkennung geschieht durch automatische Fehlererkennung im System. Im Gegensatz dazu wird bei der Offline-Erkennung der Ausfall durch wiederkehrende Prüfungen oder Instandhaltungsmaßnahmen erkannt /NEA 15/.

Ein weiterer Aspekt der Fehlerbetrachtung ist der Ursprungsort des Fehlers, welcher einen Ausfall verursacht. Da die Netzwerkkommunikation im DSLS auf Modulebene modelliert wird, sind folgende Elemente bzgl. eines Hardwareausfalls relevant:

- Hardwaremodule auf der SCP- bzw. der CP-Baugruppe (Prozessmodule, Speichermodule)

- Netzwerkkabel (optische und elektrische Datenkabel, Stromversorgungskabel)

- Schnittstellen der SCP/CP-Baugruppe zum APU, Voter und zum MSI.

Über reine Hardwarefehler hinaus gibt es bei digitaler Leittechnik immer auch die Möglichkeit eines Versagens von Komponenten aufgrund eines Softwarefehlers. Diese treten im Gegensatz zu Hardwarefehlern nie stochastisch auf, d. h. es handelt sich immer um systematische Fehler. Dadurch können sie schnell ein Versagen über die Modulgrenzen hinaus bis auf die Systemebene verursachen, sofern keine diversitären Stränge im DSLS vorhanden sind. In diesem generischen Modell eines DSLS werden jedoch keine Softwarefehler postuliert.

3.3.4.2 Netzwerkfehler, Fehlererkennung und Fehlerpropagation im generischen Modell des DSLS

In diesem Kapitel soll nun die Brücke vom oben beschriebenen Fehlermodell zum generischen Modell des DSLS und der Netzwerkkommunikation geschlagen werden.

Die Postulierung und Analyse von Ausfällen und Ausfalleffekten im generischen Modell des DSLS findet auf der Ebene von Modulen statt. Die Ausfallarten in der Netzwerkkommunikation werden hinsichtlich der Aufgabe, Telegramme mit dem korrekten Inhalt zur korrekten Zeit an den richtigen Empfänger zu übermitteln, gespiegelt. Damit können folgende Ausfallarten auftreten:

- Verfälschung im Datenbereich des Telegramms

- Verfälschung im übrigen Teil des Telegramms

- Telegramm kommt nicht im vereinbarten Zeitrahmen an

- Telegramm kommt nicht beim vereinbarten Empfänger an

- Telegrammverlust

- Langfristiger Ausfall der Kommunikation (mehr als 3 aufeinander folgende Telegramme kommen nicht an)

In Tabelle 3.5 sind den verschiedenen Ausfallarten in der Netzwerkkommunikation die Arten der Fehlererkennung und die Ausfalleffekte zugeordnet.

Tab. 3.5 Ausfallarten, -effekte und deren Erkennung in einer Verbindung eines deterministischen Netzwerks

(Sicherheitsnetzwerk und sicherheitsrelevantes Netzwerk im Realtime-Modus).

Ausfallart	**Art der Erkennung**	**Ausfalleffekt**
Verfälschung im Datenbereich des Telegramms	Online-Erkennung	Verwerfen des Telegramms, keine Propagation möglich
	Ungewollte Aktivierung	Ungewolltes Auslösen eines Stranges der Leittechnikfunktion
	Ausfall bei Anforderung	Ausfall des Leittechnikstrangs für die Leittechnikfunktion
Verfälschung im übrigen Teil des Telegramm	Online-Erkennung	Verwerfen des Telegramms
	Ungewollte Aktivierung	Ungewolltes Auslösen eines Stranges der Leittechnikfunktion
	Ausfall bei Anforderung	Ausfall des Leittechnikstrangs für die Leittechnikfunktion
Telegramm kommt nicht im vereinbarten Zeitrahmen an;	Online-Erkennung	Empfänger generiert Standardtelegramm, keine Propagation
Telegramm kommt nicht am vereinbarten Empfänger an; Telegrammverlust	Ausfall bei Anforderung (bei Versagen der Online-Erkennung)	Ausfall des Leittechnikstrangs für die Leittechnikfunktion
Langfristiger Ausfall der Kommunikation	Online-Erkennung	Selbstabschaltung des Strangs, Ausfall des Leittechnikstrangs für die Leittechnikfunktion
	Offline-Erkennung	Manuelle Abschaltung des Strangs
	Ausfall bei Anforderung	Ausfall des Leittechnikstrangs für die Leittechnikfunktion

Unter Online-Erkennung sind im Fall des generischen Modells des DSLS jene Maßnahmen zu verstehen, die in Kapitel 3.3.3 für PROFIBUS und PROFINET zur Sicherung der Kommunikation beschrieben worden sind, wie z. B. „frame check sequence“, Paritätsbit und Telegrammnummerierung. Weitere Sicherungsmaßnahmen auf der Applikationsschicht, die unter Umständen einen zusätzlichen Schutz gegen Netzwerkfehler etablieren sollen, werden in diesem Vorhaben nicht betrachtet, da die Konfiguration solcher Maßnahmen individuell gestaltet wird und damit diese Maßnahmen nicht in den generischen Rahmen des Modells passen.

Ein großer Teil der bekannten Ausfälle der Netzwerkkommunikation digitaler Leittechniksysteme in nuklearen und nicht-nuklearen Anwendungsgebieten basiert jedoch nicht auf den klassischen, in Tabelle 3.5 erwähnten Ausfallarten, sondern auf Fehlern in der Spezifikation, im Design und in der Implementierung der Systeme bzw. der Komponenten. Diese Fehler sind, wie Softwarefehler, den systematischen Fehlern zuzuordnen.

Die Analyse systematischer Fehler ist mit Hilfe der hier verwendeten, generischen Modellierung aber nicht zielführend. Zur tieferen Behandlung systematischer Fehler, wie auch gemeinsam verursachten Ausfällen (GVA), sind umfangreiche GVA-Modelle notwendig, die dann aber nicht generisch sind, sondern ein konkretes System mit den entsprechenden Spezifikationen umfassen. Daher wird dieser Aspekt der Fehleranalyse in diesem Vorhaben nicht weiter verfolgt. Für eine umfassende Zuverlässigkeitsanalyse eines Kommunikationsnetzwerks ist die Betrachtung systematischer Fehler jedoch zwingend erforderlich.

3.4 Analyse

Nach Auswertung der Analysemethoden, die im Anhang A.1 genauer beschrieben werden, und der in Kapitel 3.2 festgelegten Kriterien sowie der Berücksichtigung der Zielstellung des Vorhabens wurde für die Analyse von Netzwerkfehlern folgende Vorgehensweise gewählt:

- Entwicklung eines Modells der Netzwerkkommunikation eines digitalen Leittechniksystems, wie es typischerweise in einem Kernkraftwerk für sicherheitsrelevante Funktionen eingesetzt wird. Hierzu werden Informationen aus anderen bereits abgeschlossenen Vorhaben der GRS verwendet (u. a. /PIL 14/),
- FMEA-Analyse

- Vereinfachte Funktionsausfallanalyse,
- Detaillierte FMEA der Hardware und der Software,
- Modellierung auf Baugruppenebene, weil dies für die Fehlereffekte wichtig ist und das Modell diese Gradierung aufweist

- Fehlerbaummodellierung.

Diese Vorgehensweise ist in Abbildung 3.10 dargestellt. Das Modell der Netzwerkkommunikation wurde im Rahmen dieses Vorhabens entwickelt und in Kapitel 4.3 beschrieben.

Die Fehlerbaummodellierung von vernetzten Redundanzen eines digitalen Leittechniksystems ist mit einem hohen Aufwand verbunden, da für eine solche Modellierung eine große Anzahl an Fehlerbäumen notwendig ist. Dieser Ansatz wurde für eine solche Analyse im Arbeitspunkt 2 dieses Vorhabens angepasst. Die durchgeführten Änderungen werden in Anhang B dargestellt.

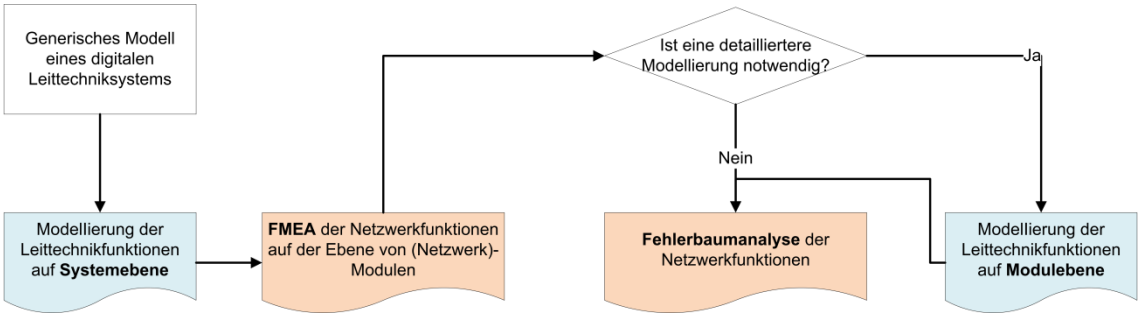


Abb. 3.10 Vorgehensweise zur Festlegung der gewählten Detailtiefe bei der Modellierung

Die Fehlerbaummodellierung im Anhang B stellt einen konservativen Ansatz dar, weil die dynamischen Eigenschaften eines Netzwerks hinsichtlich Fehlerbehandlung durch konservative Annahmen rein deterministisch im Fehlerbaum-Modell umgesetzt wurden.

Demnach führen Ausfälle und latente Fehler der Hardware-Komponenten im Kommunikationsnetzwerk zu folgenden Ausfallarten:

- Keine Datenübertragung,
- Übertragung von fehlerhaften Daten (siehe Tabelle 3.5).

Die Ausfallarten der Hardware-Komponenten des Kommunikationsnetzwerks wurden im Fehlerbaum als selbstmeldende oder nicht-selbstmeldende Ausfälle auf der Basis der Annahmen aus Tabelle 3.5 modelliert.

Die Ausfallarten in der Kommunikation verursachen im Fehlerbaummodell eines Leittechniksystems:

- Ausfall eines Leittechnik-Signals ggf. einer Leittechnik-Funktion bei Anforderung,
- Fehlanregung eines Leittechnik-Signals ggf. einer Leittechnik-Funktion.

Für die Quantifizierung der Ausfälle der Netzwerkkommunikation wurden in der Fehlerbaummodellierung zunächst die Ausfallraten der Hardware des Netzwerkes eingesetzt. Die Modellierung der Auswirkungen potentieller Softwarefehler wurde zunächst vernachlässigt, weil hierzu ein geeignetes Modell erforderlich ist. Dieser Aspekt soll in einem separaten Projekt zur Entwicklung eines Modellierungsansatzes der durch die Softwarefehler verursachten Ausfälle analysiert werden.

Weitere Informationen zur Fehlerbaummodellierung der Ausfälle der Netzwerkkommunikation sind im Anhang B enthalten.

4 Zusammenfassung

Die GRS hat im Rahmen des Vorhabens umfangreiche Recherchen zum Thema „Datenkommunikation in der digitalen Leittechnik“ durchgeführt. Hierzu wurden grundsätzliche Aspekte der Datenkommunikation und der Netzwerktechnologie erfasst und in Kapitel 2 des Berichts dokumentiert. Die gewonnenen Erkenntnisse sollen zur Erweiterung der Kompetenz der GRS auf dem Gebiet „Einsatz digitaler Technologien in der Prozess- und Sicherheitsleittechnik“ beitragen und somit die Grundlage für zukünftige sachgerechte Bewertungen von softwarebasierten Leittechniksystemen und -einrichtungen, z. B. im Rahmen von Sicherheitsüberprüfungen und im Rahmen der Bewertung von meldepflichtigen Ereignissen in deutschen und ausländischen Kernkraftwerken bilden.

Des Weiteren wurden verschiedene Methoden der Zuverlässigkeits- und Sicherheitsanalysen hinsichtlich der Anwendbarkeit für die rechnerbasierte vernetzte Leittechniksysteme ausgewertet (s. Anhang A).

Auf der Basis dieser Recherchen wurde ein Analyseansatz entwickelt, in dem die spezifischen Eigenschaften der Netzwerkkommunikation und die Aspekte sicherheitstechnischer Bewertung sicherheitsrelevanter digitaler Leittechnik berücksichtigt werden.

Dabei standen Aspekte wie die Verwendung von konservativen Annahmen und die Nachvollziehbarkeit der Ergebnisse und der generischen Analysen der für sicherheitsrelevante digitale Leittechnik typischen Netzwerktopologien im Vordergrund.

Dementsprechend wurden zunächst Kriterien zur Entwicklung einer Methodik für die Bewertung potentieller Netzwerkfehler formuliert, welche die Modellierung von Ausfällen der Netzwerkkommunikation und die Auswahl der auf dieses System angewandten Methodik maßgeblich bestimmen sollen. Diese Kriterien orientieren sich an den für Netzwerktechnologien charakteristischen Merkmalen (u.a. Fehlererkennung und Fehlerbeseitigung, bidirektionale Kommunikation, gemeinsame Nutzung der Medien durch Kommunikationsteilnehmer), welche sich von der analogen Leittechnik stark unterscheiden und für die Zuverlässigkeitsbewertung relevant sind.

Für eine Bewertung von Netzwerkkommunikation in digitaler Leittechnik muss demnach zunächst ein Modell der Netzwerkkommunikation eines digitalen Leittechniksystems, wie es typischerweise in einem Kernkraftwerk eingesetzt wird, entwickelt werden.

Anhand dessen können zunächst eine vereinfachte Funktionsausfallanalyse und anschließend eine detaillierte FMEA-Analyse der Hardware und Software erfolgen. Abschließend wird eine Fehlerbaumanalyse durchgeführt.

Die für die Methodenentwicklung maßgeblichen Kriterien, die Annahmen und das Modell eines Kommunikationsnetzwerkes in einem generischen Leittechniksystems (DSLS - Digitales Sicherheitsleittechniksystem) sind in Kapitel 3 beschrieben. Des Weiteren sind in Kapitel 3 die im Modell berücksichtigten Fehlerquellen und -fortpflanzungspfade erläutert. Hierzu wurden generische Fehlerausfallarten in der Netzwerkkommunikation der Prozessleittechnik ermittelt und phänomenologische Untersuchungen hinsichtlich Auswirkungen, Identifikation und Beherrschung postulierter Netzwerkfehler durchgeführt.

Im Rahmen dieses Vorhabens wurde das Modell der Netzwerkkommunikation entwickelt und eine erste Analyse durchgeführt. Ergebnis dieser Analyse ist, dass Ausfälle und latente Fehler der Hardware-Komponenten im Kommunikationsnetzwerk dazu führen können, dass entweder keine Datenübertragung stattfindet oder Daten fehlerhaft übertragen werden. Diese Ausfallarten wurden im Fehlerbaum als selbstmeldende oder nicht-selbstmeldende Ausfälle modelliert. Sie können entweder den Ausfall oder die Fehlanregung eines Leittechnik-Signals verursachen.

Für die Quantifizierung der Ausfälle der Netzwerkkommunikation wurden in der Fehlerbaummodellierung zunächst nur die Ausfallraten der Hardware des Netzwerkes eingesetzt. Auf die Modellierung systematischer Fehler der Hard- und Software eines generischen Leittechniksystems wurde zunächst verzichtet, weil hierzu geeignete GVA-Modelle eines spezifischen Leittechniksystems (u.a. Kenntnisse zu Technologien, Spezifikationen) notwendig sind. Diese Modelle müssen noch entwickelt werden, weil für eine umfassende Zuverlässigkeitsanalyse eines konkreten Kommunikationsnetzwerks die Analyse der Auswirkungen systematischer Fehler zwingend erforderlich ist.

Die Analyse der Hardwareausfälle des o. g. Sicherheitsleittechniksystems erfolgte mittels Fehlerbaummodellierung (s. Anhang B). Die Festlegung der zu unterstellenden Fehlerausfallarten (Basisereignisse im Fehlerbaum) erfolgte auf der Grundlage einer vereinfachten Fehlerart- und Ausfalleffektanalyse, wobei die generischen Fehlerausfallarten der Hardware digitaler Leittechnik und der Netzwerkkommunikation berücksichtigt wurden. Die Postulierung und Analyse von Ausfällen und Ausfalleffekten im

generischen DSLS-Modell fand auf der Ebene von Modulen (z.B. Kommunikation- und Linkmodule, Kommunikationsprozessor) statt.

Bei der Fehlerbaummodellierung redundanter Strukturen digitaler Sicherheitsleittechnik und komplexer Topologien der Netzwerkkommunikation wurde festgestellt, dass dies die Erstellung einer Vielzahl von Teil-Fehlerbäumen erfordern würde. Um einen immensen Modellierungsaufwand der Ausfälle der vielfach redundanten Einrichtungen zu reduzieren, wurde beschlossen, die Modellierungstechnik (Fehlerbaumanalyse-Software: RiskSpectrum®) zu verbessern. Die GRS hat hierzu im Rahmen des Vorhabens das Werkzeug „RiskLang“ entwickelt, um die Erstellung der strukturgleichen Fehlerbäume zu unterstützen. Die Erprobung des RiskLang-Werkzeugs bei der Fehlerbaummodellierung und -analyse ist in Anhang B dargestellt.

Bei der Erprobung der Fehlerbaummodellierung der Netzwerkkommunikation wurde festgestellt, dass weiterhin ein Entwicklungsbedarf bei den Methoden der Zuverlässigkeitsanalyse hinsichtlich Berücksichtigung potentieller Fehler in der Software und dynamischer Eigenschaften der systemeigenen Fehlertoleranz rechnerbasierter Leittechnik (u.a. Fehlerbehandlungsprozeduren, Master-Slave-Betrieb der Komponenten) besteht. Hierzu sollen u.a. kohärente Modelle der Fehlerauswirkung (z.B. GVA durch latente Software-Fehler), Sensivitäts- und Unsicherheitsanalysen durchgeführt werden. Diese Aspekte sollen in zukünftigen Vorhaben der GRS berücksichtigt werden.

Literaturverzeichnis

/ABB 08/ ABB's Safety Jargon Buster, Stuart Nunns, 2008

/AUT 10/ Authen, S., et al.: Guidelines for reliability analysis of digital systems in PSA context – Phase 1. Status Report, NKS-230, Nordic nuclear safety research, 2010.

/BAL 12/ Baleanu M. et al.: Repräsentative generische Netzwerke von Leittechnik in KKW. ISTec – A – 2327, Institut für Sicherheitstechnologie (ISTec) GmbH, Dezember 2012.

/BFS 05/ BfS: Methoden zur probabilistischen Sicherheitsanalyse für Kernkraftwerke BfS-SCHR-37/05, ISBN 3-86509-414-7, Wirtschaftsverlag NW/Verlag für neue Wissenschaft GmbH, Salzgitter, Oktober 2005.

/BHM 12/ Heizer, B., Mottok, J.: Real-time behavior of Ethernet on the example of PROFINET. 2012.

/BOH 06/ Bormann, A., Hilgenkamp, I.: Industrielle Netze, Hüthig Verlag Heidelberg, 2006.

/CHU 10/ Chu, T., Martinez-Guridi, G., Lehner J.: Review of quantitative software reliability methods, BNL Letter report, September 2010.

/DIN 06/ DIN EN 60812:2006: Analysetechniken für die Funktionsfähigkeit von Systemen –Verfahren für die Fehlzustandsart- und -auswirkungsanalyse (FMEA). 2006.

/DIN 06a/ DIN EN 61078:2006-10: Techniken für die Analyse der Zuverlässigkeit - Zuverlässigkeitsblockdiagramm und Boole'sche Verfahren. 2006.

/DIN 11/ DIN EN 50159:2011-04, Railway applications – Communication, signalling and processing systems – Safety-related communication in transmission systems, April 2011.

/DKE 05/ DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik Zuverlässigkeitsmanagement – Teil 3-1: Anwendungsleitfaden - Verfahren zur Analyse der Zuverlässigkeit – Leitfaden zur Methodik (IEC 60300-3-1:2003); Deutsche Fassung EN 60300-3-1: 2004, Mai 2005.

/DKE 11/ DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik DIN EN 61500; Kernkraftwerke - Leittechnik mit sicherheitstechnischer Bedeutung - Datenkommunikation in Systemen, die Kategorie-A-Funktionen ausführen (IEC 61500:2009); Deutsche Fassung EN 61500: 2011.

/FLA 08/ Flanagan, D., Matsumoto Y.: The Ruby Programming Language, O'Reilly Media, Sebastopol, CA, USA, 2008.

/FOW 03/ Fowler, M.: Patterns of Enterprise Application Architecture. Addison-Wesley, Boston, USA, 2003.

/FRA 09/ Heinz. F.: Industrielle Kommunikation mit Profinet, Hochschule Heilbronn, Dezember 2009.

/FRE 06a/ Frey, W., et al.: Erprobung und Bewertung der Methoden einer PSA für SWR-Anlagen der Baulinie 69 nach Stand von Wissenschaft und Technik (PSA SWR 69)., Fachband 1, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) gGmbH, GRS-A-3292, Garching, 2006.

/FRE 06b/ Frey, W., et al.: Erprobung und Bewertung der Methoden einer PSA für SWR-Anlagen der Baulinie 69 nach Stand von Wissenschaft und Technik (PSA SWR 69), Fachband 2, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) gGmbH, GRS-A-3293, Garching, 2006.

/GHO 11/ Ghosh, D.: DSLs in Action, Manning Publications Co., Stamford, CT, USA, 2011.

/GRA 06/ Graf, A.: Experience gained from various Modernization Projects. 1st Joint IAEA–EPRI Workshop on Modernization of Instrumentation and Control Systems in Nuclear Power Plants, Vienna, Austria, 2006.

/GRS 06/ Gesellschaft für Anlagen und Reaktorsicherheit (GRS) gGmbH: „Temporäre Störung von Symphony-Baugruppen“ im Kernkraftwerk Isar 1 am 26.01.2005. Weiterleitungsnachricht zu Ereignissen in Kernkraftwerken der Bundesrepublik Deutschland (WLN 2006/05), Köln, 09.10.2006.

/GUA 96/ Guarro S., Yau M., Motamed, M.: Development of Tools for Safety Analysis of Control Software in Advanced Reactors. U. S. Nuclear Regulatory Commission Report NUREG/CR-6465, 1996.

/HAA 02/ Haapanen Pentti, Helminen Atte: Failure mode and effects analysis of software-based automation systems. ISBN951-712-585-2, STUK-YTO-TR 190, STUK , Finland AUGUST 2002.

/HAR 15/ Wireless Hart Überblick, HART Communication Foundation, http://de.hartcomm.org/hcp/tech/aboutprotocol/aboutprotocol_what.html, abgerufen am 20.05.2015.

/HER 12a/ Herb, J.: Fault Tree Auto-Generator: How to Cope with Highly Redundant Systems. Proceedings of the Tenth International Conference on Probabilistic Safety Assessment and Management (PSAM 11), Helsinki, Finland, 2012.

/HER 12b/ Herb, J.: Tools and Procedures at GRS to Automatically Compare and Modify Fault and Event Trees, Next Generation PSA Software, Methods, and Model Representation Standards. Paris, France, 2012.

/IAE 98/ International Atomic Energy Agency (IAEA): Modernization of instrumentation and control in nuclear power plants. International Atomic Energy Agency, Vienna, Austria, 1998.

/IAE 11/ International Atomic Energy Agency (IAEA): Core knowledge on instrumentation and control systems in nuclear power plants. IAEA Nuclear Energy Series No. NP-T-3.12, Vienna, 2011.

/IEC 10a/ IEC 61508-1 (2010-04): Functional safety of electrical/electronic/programmable electronic safety-related systems. April 2010.

/IEC 10b/ IEC 61508-2 (2010-04): Functional safety of electrical/electronic/programmable electronic safety-related systems, Part 2: Requirements for electrical/electronic/programmable electronic safety-related systems. April 2010.

/IEC 10c/ IEC 61508-3 (2010-04): Functional safety of electrical/electronic/programmable electronic safety-related systems, Part 3: Software requirements. April 2010.

/IEC 10d/ IEC 61508-7 (2010-04): Functional safety of electrical/electronic/programmable electronic safety-related systems, Part 7: Overview of techniques and measures. April 2010.

/IEC 14a/ IEC 61158-1 (2014-05): Industrial communication networks – Fieldbus specifications – Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series. 23.05.2014.

/IEC 14b/ IEC 61784-2 (2014-07): Industrial communication networks – Profiles – Part 2: Additional fieldbus profiles for real-time networks based on ISO/IEC 8802-3. 17.07.2014.

/IEC 15/ IEC 61784-1 (2014-08): Industrial communication networks – Profiles – Part 1: Fieldbus profiles. August 2014.

/IEE 09/ Institute of Electrical and Electronics Engineers IEEE: Standard Criteria for Safety Systems for Nuclear Power Generating Stations. ISBN: 978-0-7381-6037-5, USA, 5.11.2009.

/IEE 12/ Institute of Electrical and Electronics Engineers IEEE: IEEE Standard for Ethernet. 802.3-2012, IEEE Computer Society, 28.12.2012.

/ISO 94/ ISO/IEC 7498-1:1994: Information technology – Open Systems Interconnection – Basic Reference Model, Part 1: The Basic Model. 17.11.1994.

/ISO 12/ ISO/IEC 30170:2012: Information technology – Programming languages – Ruby. 15.04.2012.

/ITS 96/ Federal Standard 1037C: Telecommunications: Glossary of Telecommunication Terms. Institute for Telecommunication Sciences, National Telecommunications and Information Administration, August 1996.

/ITW 15/ Webseite IT-Wissen – Das große Online-Lexikon für Informationstechnologie, abgerufen am 15.06.2015. http://www.itwissen.info/definition/lexikon/Summenrahmenverfahren-summing-frame-method.html

/KAM 13/ Kamyab, S., et al.: Sensitivity analysis on the effect of software-induced common cause failure probability in the computer-based reactor trip system unavailability. Annals of Nuclear Energy 57, pp. 294–303, 2013.

/KAN 02/ Kang, H. G., Sung, T.: An analysis of safety-critical digital systems for risk-informed design, Reliability Engineering and System Safety 78, pp 307–314, 2002.

/KIS 07/ Kisner R., et al.: Safety and Nonsafety Communications and Interactions in International Nuclear Power Plants. ORNL/NRC/LTR-07/05, prepared by the Oak Ridge National Laboratory, Oak Ridge, Tennessee, August 2007.

/KLA 01/ Klasen, F., Oestreich, V., Volz, M.: Industrielle Kommunikation mit Feldbus und Ethernet. VDE-Verlag, ISBN 978-3-8007-3297-5, 2010.

/KÖB 01/ Köberlein, K., et al.: Bewertung des Unfallrisikos fortschrittlicher Druckwasserreaktoren in Deutschland. Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) gGmbH, GRS-175, 2001.

/KOR 09/ Korsah K., et al.: Instrumentation and Controls in Nuclear Power Plants: An Emerging Technologies Update. NUREG/CR-6992, Office of Nuclear Regulatory Research, U.S. NRC, October 2009.

/KTA 14/ Kerntechnischer Ausschuss: KTA 3501, Reaktorschutzsystem und Überwachungseinrichtungen des Sicherheitssystems. 11. 2014, Gründruckfassung.

/KUW 09/ Al-Kuwaiti, M., Kyriakopoulos N., Hussein S.: A Comparative Analysis of Network Dependability, Fault-tolerance, Reliability, Security and Survivability, IEEE Communications Surveys & Tutorials, Vol. 11, No. 2, Second Quarter 2009.

/LIN 06/ Linden, von J. et al.: Erprobung und Bewertung der Methoden einer PSA für SWR-Anlagen der Baulinie 69 nach Stand von Wissenschaft und Technik (PSA SWR 69), Fachband 1 - Ereignisablauf- und Fehlerbaumanalysen für Ereignisse aus dem Leistungsbetrieb bis zum Kernschmelzen (ohne Brand). Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) gGmbH, GRS-A-3292, Garching, April 2006.

/MAR 07/ Marshall, K, et al.: Pro Active Record Databases with Ruby and Rails, Apress. Berkeley, CA, USA, 2007.

/MIC 05/ Michas C., Bühler C.: Zusammenstellung sicherheitstechnischer Anforderungen an Interfaces der Mess- und Stelltechnik in software-basierten Leittechniksystemen mit sicherheitstechnischer Bedeutung in Kernkraftwerken Abschlussbericht zum Forschungsvorhaben SR 2499 (BMU)., TÜV Industrie Service GmbH TÜV SÜD Gruppe Energie und Technologie, November 2005.

/NEA 15/ NEA/CSNI: Failure Modes Taxonomy for Reliability Assessment of Digital Instrumentation and Control Systems for Probabilistic Risk Analysis, Nuclear Safety. NEA/CSNI/R(2014)16, February 2015.

/ORN 07/ Oak Ridge National Laboratory: Safety and Nonsafety Communications and Interactions in International Nuclear Power Plants. August 2007

/PEL 10/ Peleska J., Schulz O.: Reliability Analysis of Safety-Related Communication Architectures. Safecomp 2010, Universität Bremen, 2010.

/PIL 04/ Piljugin, E.: Fachliche Unterstützung des BMU bei der Entwicklung probabilistischer Bewertungsmethoden – Anpassung und Erprobung von Methoden zur probabilistischen Bewertung digitaler Leittechnik. Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) gGmbH, GRS-A 3258, 01.12.2004.

/PIL 10/ Piljugin, E., Herb, J.: Entwicklung eines aktualisierten Ansatzes zur Berücksichtigung softwarebasierter Sicherheitsleittechnik in der PSA. Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) gGmbH, GRS-A-3550, Garching, August 2010.

/PIL 14/ Piljugin E., et al.: Entwicklung einer Bewertungsmethode für das GVA-Potenzial in der digitalen Leittechnik mit dissimilarer/diversitärer Architektur. Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) gGmbH, GRS-A-3746, Technischer Bericht, Garching, 2014.

/PRO 10a/ PROFIBUS Nutzerorganisation e.V. (PNO): PROFIBUS Systembeschreibung – Technologie und Anwendung. November 2010.

/PRO 10b/ PROFIBUS Nutzerorganisation e.V. (PNO): PROFIsafe Systembeschreibung – Technologie und Anwendung. November 2010.

/PRO 11/ PROFIBUS Nutzerorganisation e.V. (PNO): PROFINET Systembeschreibung – Technologie und Anwendung. Juni 2011.

/PRO 13/ Felser M.: PROFIBUS Handbuch: Eine Sammlung von Erläuterungen. 05.04.2010.

/SÖR 10/ Sörman, J. et al.: Exchange of PSA data and models. Proceedings of the Tenth International Conference on Probabilistic Safety Assessment and Management (PSAM 10). Seattle, WA, USA, 2010.

/TEO 11/ Teolis, D. et al.: Application of Fault Tree Methodology to Modeling of the AP1000® Plant Digital Reactor Protection System. ANS PSA 2011 International Topical Meeting on Probabilistic Safety Assessment and Analysis Wilmington. NC, USA, 2011.

/TÜR 13/ Türschmann M., Babst S., Röwekamp M.: Application of Fire PSA in case of modifications for post operational shutdown states. EUROSAFE Forum 2013, Köln.

/ZIL 15/ Zillikens, R.: Untersuchung des Standes von W&T von Netzwerktechnologien in Bezug auf Netzwerkfehler für sicherheitsrelevante Anwendungen in Leittechniksystemen von Kernkraftwerken. Entwurf. Masterarbeit im Fachbereich Security Management, FH Brandenburg, Februar 2015

/ZIO 06/ Zio, E., Baraldi, P., Patelli, E.: Assessment of the availability of an offshore installation by Monte Carlo Simulation. April 2006.

/ZIO 13/ Zio, E.: The Monte Carlo Simulation Method for System Reliability and Risk Analysis. Springer, ISBN: 978-1-4471-4587-5, 2013

Abbildungsverzeichnis

Abb. 2.1 Einfache Topologien für Netzwerke. 13

Abb. 2.2 Zusammengesetzte Topologien /BAL 12/. 15

Abb. 2.3 Systemaufbau SIMATIC S7 FH /MIC 05/. 20

Abb. 2.4 Übersicht der wichtigsten Buszugriffsverfahren /MIC 05/. 22

Abb. 2.5 Bisherige Sensor-Aktor-Anbindung. 26

Abb. 2.6 Sensor-Aktor-Anbindung über Bussysteme (Orange) an die Zentrale. 27

Abb. 2.7 Logikebene dezentral im Feldbereich verteilt. 27

Abb. 2.8 Schematische Darstellung des Grey Channels. 41

Abb. 3.1 Ablaufdiagramm zur Entwicklung eines Ansatzes zur Analyse der Verbreitung und Auswirkung postulierter Fehler in typischen Netzwerken. 43

Abb. 3.2 Zuordnung von Begriffen zum funktionellen Aufbau von A-Funktions-Einrichtungen /KTA 14/. 48

Abb. 3.3 Netzwerkaufbau einer leittechnischen Redundanz. 51

Abb. 3.4 Aufbau eines Datentelegramms mit prinzipiell variabler Länge im PROFIBUS-Netzwerk. 55

Abb. 3.5 Einordnung der Profinet Modi „Standard“ und „Realtime“ in das OSI-Schichtenmodell /BOH 06/. 58

Abb. 3.6 Aufbau eines PROFInet-Telegramms in Realtime-Modus /FRA 09/. 59

Abb. 3.7 Darstellung des Leittechniksystems und der Leittechnikfunktion. 63

Abb. 3.8 Zusammenhänge zwischen einleitendem Ereignis, Aktivierungsbedingungen, Fehlfunktion und Ausfall der Leittechnikfunktion /NEA 15/. 64

Abb. 3.9 Unterscheidung der verschiedenen Kategorien bei der Erkennung von Ausfällen bei Leittechnikfunktionen. 65

Abb. 3.10 Vorgehensweise zur Festlegung der gewählten Detailtiefe bei der Modellierung. 69

Abb. A.1 Logischer Zusammenhang von Fehler, Fehlerzustand und Versagen..... 96

Abb. A.2 Allgemeines Verfahren zur Analyse der Zuverlässigkeit eines technischen Systems /DKE 05/. 97

Abb. B.1 Modell generischer Sicherheitsleittechnik des Referenzsystems, Struktur der Hardware..... 126

Abb. B.2 Architektur der Erfassungsrechner..... 128

Abb. B.3 Architektur der Verarbeitungsrechner. 130

Abb. B.4 Architektur der Voter-Rechner. 132

Abb. B.5 Beispielfehlerbaum. 136

Abb. B.6 RiskLang Programm, um den Fehlerbaum in Abb. B.5 zu erzeugen..... 137

Abb. B.7 Implementierung von RiskLang..... 138

Abb. B.8 Fehlerbaum für das Top-Ereignis..... 141

Abb. B.9 RiskLang Programm, um den Fehlerbaum in Abb. B.8 zu erzeugen..... 141

Abb. B.10 Fehlerbaum für das Startversagen der Pumpe ECCP1..... 142

Abb. B.11 RiskLang Programm, um den Fehlerbaum in und strukturell gleiche für die restlichen drei Redundanzen zu erzeugen. 142

Abb. B.12 Fehlerbaum für das Startversagen der Pumpe ECCP1 (aufgrund fehlender Anregung durch das Frischdampfdurchsatzsignal oder Ausfall des Voter in der ersten Redundanz). 144

Abb. B.13 Fehlerbaum den nicht selbstmeldenden Ausfall des Voter der ersten Redundanz aufgrund einer fehlenden Anregung durch das Frischdampfdurchsatzsignal. 145

Abb. B.14 RiskLang Programm, um den Fehlerbaum in Abb. B.13 und strukturell gleiche Fehlerbäume für die restlichen drei Redundanzen zu erzeugen. 147

Tabellenverzeichnis

Tab. 2.1 Schichtenmodelle nach OSI und DoD /BAL 12/ 12

Tab. 2.3 Übersicht über Ethernet-Topologien in Abhängigkeit der möglichen passiven Verbindungselemente /BAL 12/ 17

Tab. 2.4 Relevante Feldbusse in kerntechnischen Anwendungen /BAL 12/..... 18

Tab. 2.5 Sicherheitsbussysteme..... 39

Tab. 3.1 Kommunikationsaufgaben der im MSI vorhandenen Baugruppen..... 52

Tab. 3.2 Verbindungen, die durch das Sicherheitsnetzwerk hergestellt werden..... 54

Tab. 3.3 Durch das sicherheitsrelevante Netzwerk hergestellte Verbindungen..... 60

Tab. 3.4 Vergleich des Sicherheitsnetzwerks und dem sicherheitsrelevantem Netzwerk im generischen Modell des DSLS 61

Tab. 3.5 Ausfallarten, -effekte und deren Erkennung in einer Verbindung eines deterministischen Netzwerks..... 67

Tab. A.1 Messbarkeit von Aspekten der Zuverlässigkeitsbewertung /KUW 09/. ... 101

Tab. A.2 Gegenüberstellung unterschiedlicher Bewertungskonzepte und ihre Eigenschaften /KUW 09/..... 103

Tab. A.3 Gegenüberstellung unterschiedlicher Bewertungskonzepte und ihre Eigenschaften /KUW 09/ (Fortsetzung)..... 104

Tab. A.4 Zulässige Fehlerhäufigkeiten bei SIL-Level 1 bis 4. 108

Tab. A.5 Einteilung der Verfahren zur Schätzung der Maßgrößen für Grundereignisse. 115

Tab. A.6 Merkmale der beschriebenen Zuverlässigkeitsanalyseverfahren /DKE 05/. 118

Tab. A.7 Merkmale weiterer ausgewählter Zuverlässigkeitsanalyseverfahren (Fortsetzung). 119

Tab. A.8 Anwendbarkeit allgemeiner Zuverlässigkeitsanalyseverfahren auf die Leittechniksysteme /DKE 05/. 120

Tab. A.9 Anwendbarkeit allgemeiner Zuverlässigkeitsanalyseverfahren auf die Leittechniksysteme /DKE 05/ (Fortsetzung). 121

Tab. B.1 Verhalten der zweiten Maxima-Berechnung in der Anwendersoftware der Verarbeitungsrechner bei Ausfallkombinationen der Eingänge. 131

Tab. B.2 Verhalten der 2 aus 4 Auswahl in der Anwendersoftware der Voter-Rechner bei Ausfallkombinationen der Eingänge. 133

Tab. B.3 Spezifikation der RiskLang-Befehle. 135

Tab. B.4 Mögliche Typen von Basisereignissen bzw. Verknüpfungen in RiskLang 135

Tab. B.5 Alle Fehlerbäume zur Modellierung des Referenzsystems. 146

Tab. B.6 Minimalschnitte mit den höchsten Beiträgen. 148

Abkürzungsverzeichnis

ADSL	Asymmetric Digital Subscriber Line
AES	Advanced Encryption Standard
ASIC	Application Specific Inte¬grated Cirquit
ATM	Asynchronous Transfer Mode
CAN	Controller Area Network
CCF	Common Cause Failure
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
DHCP	Dynamic Host Configuration Protocol
DIN	Deutsches Institut für Normung
DNA	Digital Network Architecture
DNS	Domain Name System
DP	Decentralized Peripherals
EMI	Electromagnetic Interference
FMEA	Failure Mode and Effect Analysis
FPGA	Field Programmable Gate Array
FTAM	File Transfer and Access Management
FTP	File Transfer Protocol
GAN	Global Area Network
HART	High Addressable Remote Transducer
HMI	Human Machine Interface
HTTP	Hyper Text Transfer Protokoll

HTTPS	Hyper Text Transfer Protokoll Secure
IAEA	International Atomic Energy Agency
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronic Engineers
IP	Internet Protocol
ISO	International Standards Organization
IT	Informationstechnik
LAN	Local Area Network
MAC	Media Access Control
MTU	Maximum Transmission Unit
NTP	Network Time Protocol
OSI	Open Systems Interconnection
PAN	Personal Area Network
PFD	Probability of Failure on Demand
PLC	Programmable Logic Controller
PLD	Programmable Logic Device
POP	Post Office Protokoll
PPP	Point to Point Protocol
PROFIBUS	Process Field Bus
RFC	Requests for Comments
RFI	Radio Frequency Interference
RFID	Radio Frequency Identification
RTP	Real Time Transport Protocol

SCADA	Supervisory Control and Data Acquisition
SIL	Safety Integrity Level
SMTP	Simple Mail Transfer Protokoll
SSH	Secure Shell
SSL	Secure Socket Layer
TELNET	Telecommunication Network Protocol
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
USB	Universal Serial Bus

A Methoden der Zuverlässigkeits- und Sicherheitsanalyse

A.1 Einleitung

Im Folgenden werden verschiedene Methoden der Zuverlässigkeits- und Sicherheitsanalyse dahingehend untersucht, inwiefern sie für eine Bewertung der Netzwerkkommunikation sicherheitsrelevanter softwarebasierter Leittechnik hinsichtlich der Auswirkungen potentieller Ausfälle geeignet sind.

A.2 Einführung und Definitionen

Die softwarebasierte Leittechnik nutzt sowohl für redundanzinterne wie auch redundanzübergreifende Kommunikation moderne Netzwerktechnologien. Die nationale und internationale Betriebserfahrung zeigt, dass die interne und externe Kommunikation eines vernetzten Leittechniksystems auf dessen Zuverlässigkeit einen erheblichen Einfluss haben kann (siehe u. a. GRS WLN 2006/05 „Temporäre Störung von Symphony-Baugruppen“ /GRS 06/). Die Auswirkungen potentieller Fehlereffekte innerhalb und außerhalb der Kommunikationsnetzwerke sind bisher noch nicht systematisch für sicherheitsrelevante Funktionen in Kernkraftwerken im Rahmen einer probabilistischen Sicherheitsanalyse untersucht worden. In diesem Anhang A werden daher verschiedene Methoden betrachtet, die für eine Bewertung der Netzwerkkommunikation sicherheitsrelevanter softwarebasierter Leittechnik hinsichtlich der Auswirkungen potentieller Ausfälle zum Einsatz kommen können.

Zuverlässigkeits- und Sicherheitsanalysen komplexer technischer Systeme können unterschiedliche Zielstellungen haben, z. B.

- Qualitative Abschätzung der Auswirkungen postulierter Fehler,
- Identifizierung von Fehlerquellen bei der Herstellung und/oder beim Betrieb eines Systems oder einer Einrichtung,
- Quantitative Bestimmung von Zuverlässigkeitskenngrößen (u. a. Verfügbarkeit eines Systems oder einer Einrichtung).

Solche Analysen werden auf der Basis von Modellen durchgeführt, wobei typischerweise folgende Modelle eingesetzt werden:

- Boolesche Modelle für analytische Analyseverfahren, z. B. Fehlerbaumanalyse,
- Zustandsänderungsmodelle für analytische Analyseverfahren, z. B. mittels Markov-Prozessen oder -Ketten,
- Monte-Carlo-Simulationen.

Die Zielsetzung der Analyse und die Analysemethode machen es erforderlich, frühzeitig die Betrachtungstiefe („Abstractionlevel“) eines Systems bzw. einer Einrichtung festzulegen und die Elemente des Modells zu definieren. Dazu zählen z. B. /NEA 15/:

- die Systemfunktion, z. B. die Gliederung eines Reaktorschutzsystems in Funktionen der Reaktorschnellabschaltung oder der Sicherheitseinspeisung,
- die Steuerung von verfahrenstechnischen Komponenten durch die Auswahllogik mit redundanten Signalen,
- die Signale aus den Leittechnikschränken,
- die Signale der Leittechnik-Baugruppen,
- die Bauelemente einer Leittechnikbaugruppe.

Bei einem rechnerbasierten Leittechniksystem werden diese Elemente durch die Implementierung der spezifizierten Funktionen durch Hard- und Softwarekomponenten und dem (logischen und physikalischen) Zustand von deren Bestandteilen bestimmt /NEA 15/. In diesem Zusammenhang werden in /NEA 15/ und /DKE 05/ die folgenden Definitionen verwendet.

- Ausfall, Versagen
  Beendigung der Fähigkeit einer Einheit, eine geforderte Funktion zu erfüllen. Nach einem Ausfall befindet sich die Einheit in einem Fehlerzustand. Der Ausfall ist ein Ereignis, im Unterschied zum Fehlerzustand. In dem Fall, dass einige Funktionen des Systems ein Versagen aufweisen, aber andere Funktionen weiterhin spezifikationsgemäß arbeiten, spricht man von einer eingeschränkten Funktionalität des Systems.

- Ausfallart, Fehlerzustandsart
  Die Art und Weise, wie sich das Versagen einer Funktion äußert. Ein Fehlerzustand ist oft das Ergebnis eines Ausfalls der Einheit selbst, er kann aber auch ohne vorherigen Ausfall vorhanden sein.

- Ausfallauswirkung
  Auswirkungen des Versagens eines Systems auf externe Systeme.

- Funktionseinheit
  Ein aus mehreren Komponenten bestehendes Teilsystem oder System, das für sich allein betrachtet werden kann. Eine Funktionseinheit kann aus Hardware, Software oder beidem bestehen und in besonderen Fällen Personen einschließen.

- Fehlerzustand
  Zustand einer Einheit, in dem sie unfähig ist, eine geforderte Funktion zu erfüllen.

- Fehler
  Hypothetische oder nachgewiesene Ursache von Fehlerzuständen. Es gibt zahlreiche Klassifikationsmerkmale von Fehlern, wie etwa Zeitpunkt (im Herstellungsprozess oder während des Betriebs) oder Persistenz (dauerhafte oder transiente Fehler). Ein weiteres wichtiges Kriterium für einen Fehler ist seine Aktivität: produziert ein Fehler einen Fehlerzustand im System, so ist er aktiv, erzeugt er keinen Fehlerzustand, so heißt er inaktiv oder passiv.

- Gemeinsam verursachter Ausfall (GVA)
  Ausfall mehrerer Komponenten/Systeme aufgrund eines einzelnen Fehlers /Ereignisses

- Komponente
  Einzelteil, Bauelement oder Gerät einer Funktionseinheit. In der niedrigsten Ebene der Analyse betrachtete Einheit.

Die logische Beziehung dieser Begriffe kann in einem Diagramm aufgezeigt werden (siehe Abbildung A.1). Die Pfeile in Abbildung A.1 stellen kausale Verbindungen der Ereignisse dar. Hängen die Funktionen des Systems B von denen des Systems A ab, können die Fehlerzustände dorthin propagieren.

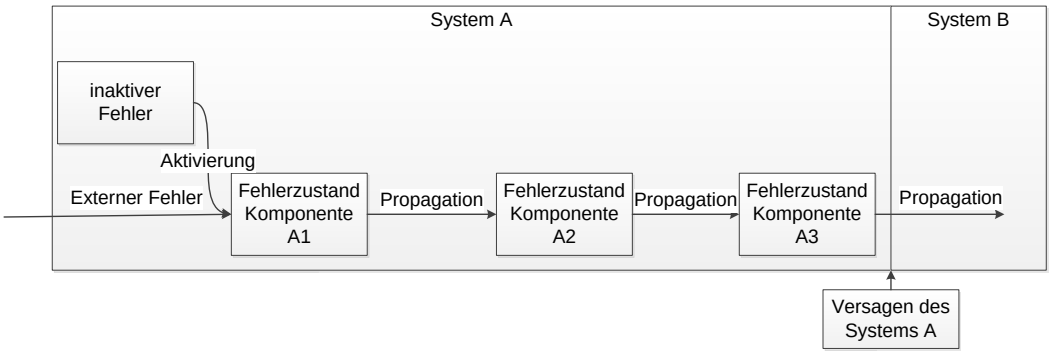


Abb. A.1 Logischer Zusammenhang von Fehler, Fehlerzustand und Versagen

Für die Analyse der Fehlerfortpflanzung in den vernetzten Leittechniksystemen sind folgende Aspekte sehr wichtig:

- Fehlerort,
- Fehlerausfallart,
- Fehlererkennung (z. B. unentdeckt, selbstmeldend, entdeckt bei Anforderung),
- potentielle Auswirkungen (Fehlereffekt auf die nächste Komponente, Endeffekt).

Um die Zuverlässigkeit und Verfügbarkeit eines Systems bewerten zu können, müssen verschiedene dieser Aspekte gleichzeitig in der Analyse betrachtet werden, da diese sich gegenseitig beeinflussen können. Beispielsweise muss die Fehlertoleranz des Systems für eine Zuverlässigkeitsbewertung betrachtet werden. Diese hat jedoch auch einen Einfluss auf die Sicherheit und die Robustheit des Systems. Es ist demnach nicht ausreichend, nur einen Aspekt bei der Zuverlässigkeitsanalyse zu betrachten. Die verschiedenen, derzeit verfügbaren Methoden zur Zuverlässigkeitsbewertung behandeln diese Aspekte auf unterschiedliche Weise und sind daher nur teilweise geeignet, die Zuverlässigkeit der Kommunikationssysteme in digitalen Leittechniksystemen zu bewerten. Im Folgenden wird zunächst die allgemeine Vorgehensweise bei einer Zuverlässigkeitsbewertung erläutert. Anschließend werden die verschiedenen Methoden vorgestellt und zum Schluss dahingehend verglichen, welche der Methoden für eine Bewertung von bestimmten Aspekten der digitalen Leittechnik geeignet sind.

A.3 Allgemeine Vorgehensweise bei der Zuverlässigkeitsanalyse

Ein allgemeines Verfahren zur Analyse der Zuverlässigkeit technischer Systeme ist in Abbildung A.3 dargestellt und umfasst die nachfolgenden Aufgaben /DKE 05/:

- Definition des Systems

  Definition des zu analysierenden Systems, seiner Betriebszustände, seiner funktionalen Beziehungen mit seiner Umgebung einschließlich der Schnittstellen und Prozesse. Im Allgemeinen ist die Systemdefinition eine aus der Systementwicklung folgende Angabe.

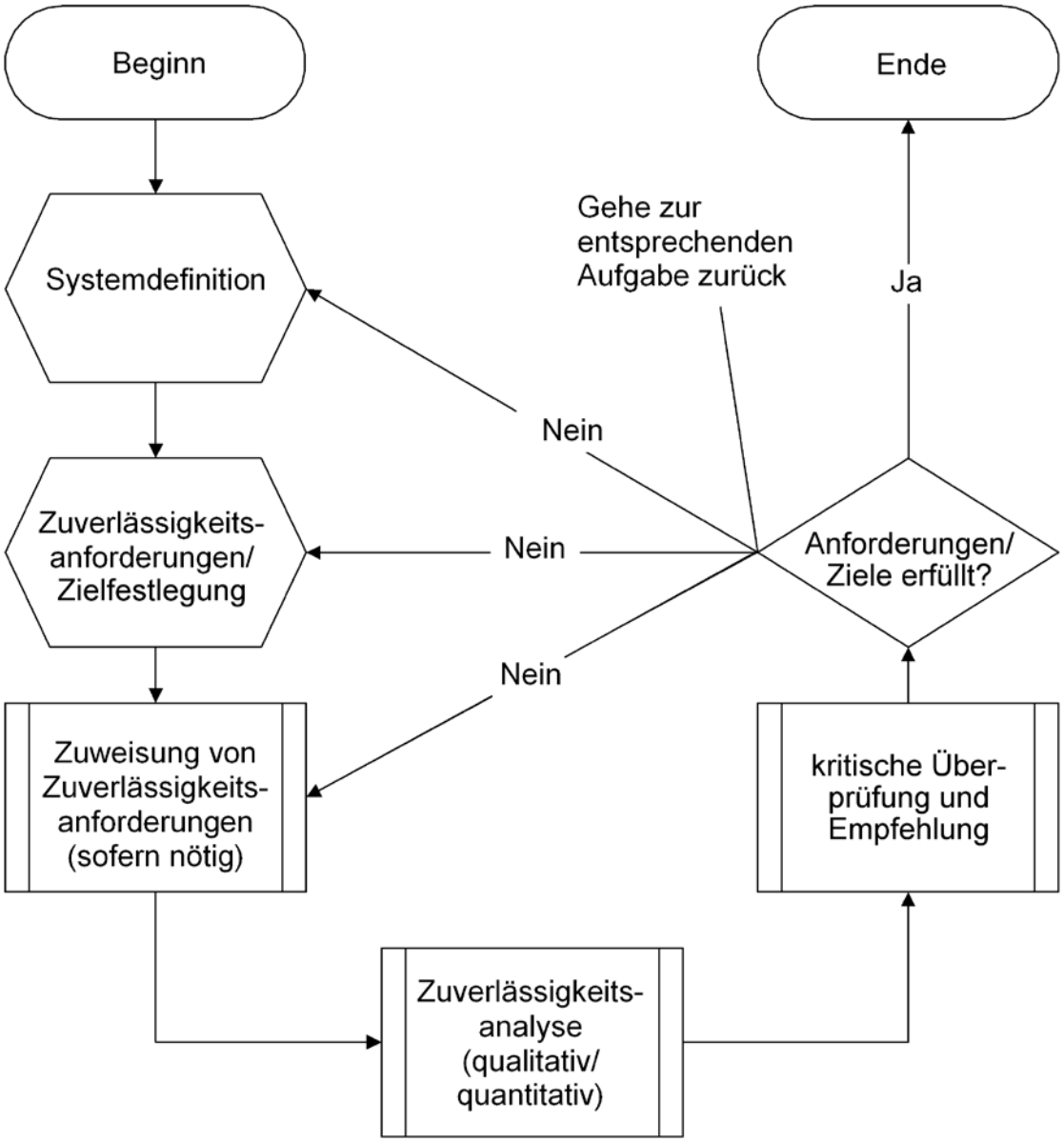


Abb. A.2 Allgemeines Verfahren zur Analyse der Zuverlässigkeit eines technischen Systems /DKE 05/

- Festlegung der Zuverlässigkeitsanforderungen und Ziele

  Auflistung aller Anforderungen an die Funktionsfähigkeit und Verfügbarkeit oder an Ziele, Eigenschaften und Merkmale des Systems, zusammen mit Umgebungs- und Betriebsbedingungen sowie Anforderungen an die Instandhaltung. Definition von

Systemausfall, Ausfallkriterien und Ausfallbedingungen auf der Grundlage funktionaler Systemspezifikationen, erwarteter Betriebsdauer und Betriebsumgebung (Nutzungsprofil und Nutzungsdauer). Entsprechend der /DKE05/ sollte hierfür als Anleitung die IEC 60300-3-4 verwendet werden.

- Zuweisung von Zuverlässigkeitsanforderungen
  Zuweisung von Zuverlässigkeitsanforderungen oder Zuverlässigkeitszielen des Systems an die verschiedenen Teilsysteme, wenn dies notwendig ist, in der frühen Entwicklungsphase.

- Zuverlässigkeitsanalyse
  Analyse des Systems mittels Methoden zur Zuverlässigkeitsbewertung und zugehöriger Leistungsdaten.

Bei der Vorgehensweise unterscheidet man zwischen qualitativer und quantitativer Analyse:

- Qualitative Analyse:
  - Analyse der funktionalen Systemstruktur,
  - Ermittlung der Fehlerzustandsarten des Systems und seiner Komponenten, Ausfallmechanismen, Ursachen, Auswirkungen und Folgen von Ausfällen,
  - Ermittlung von Abnutzungsmechanismen, die zu Ausfällen führen können,
  - Analyse der Ausfall-/Fehlerzustandspfade,
  - Analyse der Instandhaltbarkeit hinsichtlich Zeit, Problemeingrenzungs- und Reparaturverfahren,
  - Ermittlung, wie zutreffend die vorgesehenen Diagnosemittel zum Erkennen von Fehlerzuständen sind,
  - Analyse der Möglichkeiten zur Vermeidung von Fehlerzuständen,
  - Bestimmung möglicher Instandhaltungs- und Reparaturstrategien usw.
- Quantitative Analyse:
  - Entwicklung von Zuverlässigkeitsmodellen,
  - Festlegung der zu verwendenden numerischen Referenzdaten,

- Durchführung der Zuverlässigkeitsberechnungen.
- Bei Bedarf können Analysen der kritischen Eigenschaften von Komponenten und Sensitivität der Systemfunktionen gegenüber deren Fehler durchgeführt werden.

A.3.1 Zuverlässigkeitsaspekte

Für die Bestimmung von Zuverlässigkeitsanforderungen und zur Durchführung einer Zuverlässigkeitsanalyse müssen aussagekräftige Kenngrößen definiert werden. Aus der Betriebserfahrung ergeben sich hierfür zunächst verschiedene Aspekte, die für die Zuverlässigkeit von Systemen wichtig sind. Die verschiedenen Begriffe sind gemäß /KUW 09/ wie folgt definiert:

- Die Ausführbarkeit gibt den Grad an, in dem ein System oder eine Komponente die Funktion erfüllt, für die es/sie ausgelegt ist.
- Die Authentizität gibt an, inwiefern ein System oder eine Komponente in der Lage ist, die Identität eines Benutzers, Prozesses oder einer Funktionseinheit zu verifizieren. Sie dient oft als Voraussetzung für Zugriffsrechte in einem Informationssystem.
- Die Erreichbarkeit definiert die Möglichkeit, den Zugriff auf ein System und die Informationen, die ein System an eine Person oder ein weiteres Teilsystem ausgibt, zu bestimmen, zu kontrollieren oder einzuschränken.
- Die Fehlertoleranz gibt an, inwieweit das System auch bei Auftreten eines Fehlers dazu in der Lage ist, einen Systemausfall zu verhindern.
- Die Instandhaltbarkeit macht Aussagen über die Möglichkeit von Komponenten oder Systemen, inwieweit Änderungen, Wartungsarbeiten oder Instandsetzungen durchgeführt werden können.
- Die Integrität gibt an, inwieweit es an einer Komponente oder in einem System zu unerwünschten Zuständen kommen kann. In Netzwerken zählt hierzu auch die Sicherstellung der unversehrten und korrekten Datenübertragung.
- Die Nachweisbarkeit liefert Aussagen über die Gewissheit eines Systems oder einer Komponente, dass für den Sender einer Information der Nachweis er-

bracht wurde, dass er die Information verschickt hat, und dass für den Empfänger die Identität des Senders nachgewiesen ist.

- Die Prüfbarkeit gibt an, inwieweit es möglich ist, das System oder die Komponente zu prüfen.
- Die Sicherheit gibt an, inwieweit ein System in der Lage ist, ein Fehlverhalten, welches zu einem katastrophalen Schaden innerhalb eines bestimmten Zeitfensters führen kann, zu vermeiden.
- Die Sicherung beinhaltet im Wesentlichen administrative Maßnahmen zum Schutz von Komponenten und Systemen gegen Fremdeinwirkungen.
- Die (Un-) Verfügbarkeit gibt an, ob ein System oder eine Komponente für ihre auszuführende Funktion verfügbar ist und bei Anforderung entsprechend ihre Aufgabe erfüllt.
- Bei der Vertraulichkeit handelt sich um einen Teilaspekt der Sicherung. Sie gibt an, inwieweit der Schutz vor unbefugter Preisgabe von Informationen gewährleistet ist.
- Die Zurechenbarkeit beschreibt die Fähigkeit, die Aktionen von Systemen, Komponenten oder Personen an anderen Systemen zu verfolgen oder zu prüfen.
- (Un-) Zuverlässigkeit gibt die Fähigkeit eines Systems oder einer Komponente an, seine Funktionen unter bestimmten Bedingungen für eine bestimmte Zeit auszuführen.

Nicht jeder dieser Aspekte ist tatsächlich messbar (quantifizierbar) und damit auch als Kenngröße für Zuverlässigkeitsanalysen geeignet. In /KUW 09/ wurden daher diese verschiedenen Aspekte hinsichtlich ihrer Messbarkeit untersucht. Das Ergebnis ist in Tabelle A.1 dargestellt.

Aspekte wie Fehlertoleranz, Instandhaltbarkeit, Zuverlässigkeit und Verfügbarkeit sind demnach messbare Größen, während Aspekte wie Authentizität, Sicherung und Erreichbarkeit nicht messbar sind.

Tab. A.1 Messbarkeit von Aspekten der Zuverlässigkeitsbewertung /KUW 09/

Aspekte	messbar	nicht messbar	Aspekte	messbar	nicht messbar
Ausführbarkeit	x		Prüfbarkeit		x
Authentizität		x	Sicherheit		x
Erreichbarkeit		x	Sicherung		x
Fehlertoleranz	x		Verfügbarkeit	x	
Instandhaltbarkeit	x		Vertraulichkeit		x
Integrität	x		Zurechenbarkeit		x
Nachweisbarkeit		x	Zuverlässigkeit	x	

Der Zusammenhang der Aspekte aus Tabelle A.1 mit verschiedenen System- bzw. Komponenteneigenschaften ist in Tabelle A.2 und Tabelle A.3 verdeutlicht. In der Zeile „Definition und Ziele“ (Zeile 2) werden zunächst die jeweiligen Eigenschaften definiert, bevor in Zeile „Aspekte“ (Zeile 3) die verschiedenen Zuverlässigkeitsaspekte den jeweiligen System- bzw. Komponenteneigenschaften zugeordnet werden. Zu den Eigenschaften zählen die Robustheit, die Fehlertoleranz, die Zuverlässigkeit, die Sicherheit und die Überlebensfähigkeit eines Systems. Man erkennt, dass einige Aspekte bei unterschiedlichen Systemeigenschaften aufgeführt sind. Die Eigenschaft der Sicherung von Systemen und Komponenten beinhaltet beispielsweise die Aspekte Vertraulichkeit, Integrität und Verfügbarkeit, welche auch zur Robustheit von Systemen und Komponenten beitragen. Demnach hat die Änderung eines Aspekts immer Auswirkungen auf verschiedenen Systemeigenschaften.

In der ersten Zeile „Aufgaben“ von Tabelle A.3 werden die Maßnahmen aufgeführt, die zur Erfüllung der Anforderungen der verschiedenen Aspekte und damit der jeweiligen Eigenschaft notwendig sind. In der letzten Zeile „Gefährdungsursachen und mögliche Modellbetrachtung“ wird schließlich das Gefahrenpotential aufgeführt, welches besteht, wenn diese Eigenschaft nicht ausreichend betrachtet wurde.

Bei der Zuverlässigkeit handelt es sich einerseits um einen Aspekt, welcher zu verschiedenen System- oder Komponenteneigenschaften beiträgt. Andererseits ist die Zuverlässigkeit aber auch selber eine System- oder Komponenteneigenschaft, die sich wiederum durch verschiedene Aspekte beschreiben lässt. Ähnliches gilt für die Sicherung. Um die verschiedenen Aspekte und Eigenschaften von Systemen und Komponenten zu betrachten und auszuwerten, gibt es verschiedene, nachfolgend beschrie-

bene Analysemethoden. Wie in Tabelle A.1 aufgezeigt, gibt es verschiedene Aspekte, die in solchen Analysen quantifiziert und analysiert werden können, wodurch Aussagen über verschiedene Systemeigenschaften getroffen werden. Dabei muss jedoch beachtet werden, dass, wie in Tabelle A.2 und Tabelle A.3 veranschaulicht, verschiedene Aspekte der betrachteten Eigenschaft auch Auswirkungen auf andere Eigenschaften und damit auf die Gesamtreaktion des betrachteten Systems hat. Es ist demnach nicht ausreichend für eine Sicherheitsanalyse eines Systems, nur eine Systemeigenschaft für sich genommen zu betrachten.

Tab. A.2 Gegenüberstellung unterschiedlicher Bewertungskonzepte und ihre Eigenschaften /KUW 09/.

Bewertungskonzepte	Verlässlichkeit/Robustheit	Fehlertoleranz	Zuverlässigkeit	Sicherung	Überlebensfähigkeit
Definition und Ziele	Fähigkeit, die geforderten Funktionen während der gesamten Lebensdauer des Systems vertrauenswürdig und zuverlässig auszuführen.	Fähigkeit, die Funktion auch bei Auftreten von Fehlern fortzuführen.	Bedingte Wahrscheinlichkeit, dass ein System seine Funktionen in einem bestimmten Zeitintervall korrekt ausführt.	Fähigkeit, ein System vor unerwünschten Aktionen zu schützen und die Zuverlässigkeit, Intaktheit und Verfügbarkeit des Systems zu erhalten.	Fähigkeit, seine Funktionen auch bei Angriffen, Fehlern oder Unfällen im vorgegebenen zeitlichen Rahmen zu erfüllen.
Aspekte	– Fehlertoleranz – Instandhaltbarkeit – Integrität – Sicherheit – Sicherung – Verfügbarkeit – Vertraulichkeit – Zuverlässigkeit	– Ausführbarkeit – Instandhaltbarkeit – Prüfbarkeit – Verfügbarkeit	– Fehlertoleranz – Instandhaltbarkeit – Prüfbarkeit – Verfügbarkeit	– Erreichbarkeit – Authentizität – Integrität – Nachweisbarkeit – Sicherheit – Verfügbarkeit – Vertraulichkeit – Zurechenbarkeit	– Ausführbarkeit – Fehlertoleranz – Sicherheit – Sicherung – Zuverlässigkeit

Tab. A.3 Gegenüberstellung unterschiedlicher Bewertungskonzepte und ihre Eigenschaften /KUW 09/ (Fortsetzung).

Bewertungskonzepte	Verlässlichkeit/Robustheit	Fehlertoleranz	Zuverlässigkeit	Sicherung	Überlebensfähigkeit
Aufgaben	– Fehlervorbeugung – Fehlerbeseitigung – Fehlervorhersage	– Fehlerdetektion – Systemwiederherstellung – Fehlermaskierung – Umgestaltung – Redundanz	– Fehlervermeidung – Fehlererkennung und -isolation – Wiederherstellung nach Fehler	– Überwachungssystem: Kryptographie, Analyse, Firewalls, Authentifizierung – Raumüberwachung: Zugangskontrolle, Autorisierung, Protokollierung. – Richtlinien – Bewusstsein und Training.	– Definition wesentlicher und nicht lebenswichtiger Funktionen – Definition von Überlebensfunktionen zur Stärkung der Widerstandsfähigkeit gegen Angriffe
Ursachen und Bewertungskriterien	– Fehler und Ausfall – Quantifizierung und Modellierung von zufälligen, fehlerhaften und unbeabsichtigten Hardwarefehlern oder seltenen Softwarefehlern	– Fehler und Ausfall – Quantifizierung und Modellierung von zufälligen, fehlerhaften und unbeabsichtigten Hardwarefehlern oder seltenen Softwarefehlern	– Fehler und Ausfall – Quantifizierung und Modellierung von zufälligen, fehlerhaften und unbeabsichtigten Hardwarefehlern oder seltenen Softwarefehlern	– Absichtliche und feindliche Angriffe – Analyse von schwer zu modellierenden Ausfällen, die absichtlich vom Menschen verursacht werden und zu Sicherheitsdefiziten führen	– Absichtliche Angriffe, Ausfälle und Unfälle, die zu potentiell zerstörenden Ereignissen führen können – In Modellen kann die Zufälligkeit für versehentliche Fehler angenommen werden, nicht jedoch für bewusste Angriffe

A.4 Statische Methoden

A.4.1 Fehlerbaumanalyse

Die Fehlerbaumanalyse ist eine bewährte Art der Systemanalyse, die sich einer deduktiven logischen Vorgehensweise bedient. Sie dient allgemein der Analyse der Fehlerfortpflanzung innerhalb von redundanten Strukturen. Die Methodik entspricht sowohl der DIN 25424 als auch dem IEC 61025.

Die Fehlerbaumanalyse eignet sich prinzipiell zur Modellierung von logischen und quantitativen Wechselwirkungen bzw. Zusammenhängen in einem komplexen Netzwerk eines technischen Systems.

Die Fehlerbaumanalyse ist eine valide und bewährte Methode zur Fehleranalyse. Diese Methode setzt voraus, dass detaillierte Kenntnisse der Funktionen und der Abhängigkeiten der Komponenten untereinander vorhanden sind. Des Weiteren sind die Voraussetzungen zur Ermittlung der Zuverlässigkeitskenndaten (z.B. Daten zur Ermittlung der Ausfallraten) erforderlich. Unter diesen Voraussetzungen ist die Fehlerbaumanalyse in der Lage, besonders kritische Komponenten zu identifizieren und die Eintrittswahrscheinlichkeiten der Top-Ereignisse zu quantifizieren.

Ein möglicher Ansatz für die Bewertung von Fehlerauswirkungen in sicherheitsrelevanten Netzwerken ist die Fehlerbaumanalyse, da diese allgemein der Analyse der Fehlerfortpflanzung innerhalb von redundanten Strukturen dient. Die Fehlerbaumanalyse eignet sich prinzipiell zur Modellierung von logischen und quantitativen Wechselwirkungen bzw. Zusammenhängen in einem komplexen Netzwerk eines technischen Systems.

Im Bereich probabilistischer Sicherheitsanalysen (PSA) von Kernkraftwerken ist die Fehlerbaummodellierung von verfahrenstechnischen Systemen seit mehreren Jahrzehnten eine bewährte Methode. Beispielsweise werden Ausfälle der (analogen) Leittechnik von Kernkraftwerken mit Hilfe von Fehlerbäumen in der PSA modelliert. Die Vorgehensweise ist im PSA-Leitfaden /BfS 05/ beschrieben.

Die Fehlerbaumanalyse kann aufgrund des Modellierungsansatzes hauptsächlich für statische Analysen der Netzwerke digitaler Leittechnik verwendet werden. Dabei wer-

den üblicherweise sogenannte „worst-case“-Annahmen hinsichtlich Fehlerauswirkungen gemacht bzw. die Ergebnisse anderer Analysemethoden (u. a. FMEA, Markov-Chain-Verfahren) als Basisereignisse im Fehlerbaum verwendet. Bei einer statischen Analyse werden konservative Annahmen (z. B. ein worst-case-Ansatz) bezüglich der Fehlerfortpflanzung getroffen, so dass deren zeitliche Entwicklung nicht explizit modelliert werden muss. Eine Modellierung der zeitlichen Ausfallabfolgen wäre prinzipiell auch mit Fehlerbäumen möglich. Dafür wären aber sehr komplexe Fehlerbaummodelle notwendig, um die transienten Übergange zwischen den einzelnen Systemzuständen durch die statische Fehlerbaumlogik abzubilden. Bei entsprechenden Annahmen ist die Fehlerbaummodellierung der Ausfälle ein konservatives Vorgehen, da eine automatische Fehlerbehandlung (z. B. eine fehlertolerante Signalverarbeitung, die durch Watch-Dog-Timer eines Rechners initialisiert werden kann) nicht berücksichtigt wird.

Bei der Analyse digitaler, vernetzter Leittechnik mittels der Fehlerbaumanalyse ist bei der Modellierung von Ausfällen in (hoch-)redundanten Systemen, wie z. B. von Hardwareausfällen digitaler Sicherheitsleittechnik, eine sehr große Zahl von komplexen, aber strukturell sich wiederholenden Fehlerbäumen notwendig /PIL 10/. Bisher wurden bei der in der GRS angewandten Fehlerbaumanalyse alle Fehlerbäume manuell erstellt, was bei einem komplexen, redundanten System sehr zeitaufwendig und fehleranfällig ist. Durch eine Automatisierung der Fehlerbaumerstellung könnten strukturell ähnliche Fehlerbäume durch ein Programm automatisch generiert werden. Die automatisierte Fehlerbaumerstellung hat neben der Zeitersparnis den Vorteil, dass mögliche Fehler bei der manuellen Eingabe verhindert werden und der Validierungsaufwand für die Modellierung reduziert wird. Aus diesem Grund wurde im Rahmen dieses Vorhabens die bisher in der GRS angewandte Fehlerbaumanalyse entsprechend erweitert. Eine Beschreibung der durchgeführten Änderungen befindet sich in Anhang B.

A.4.2 Fehlerart- und Auswirkungsanalyse

Die Fehlerart- und –auswirkungsanalyse (FMEA) ist eine weit verbreitete Methode zur qualitativen Zuverlässigkeitsanalyse in sicherheitskritischen Systemen. Die Arbeitsweise der FMEA ist induktiv, d. h. die Analyse beginnt bei einer Basiskomponente (oder einem primären Ereignis) und beschäftigt sich dann mit deren Fehlerarten sowie den Einflüssen der Fehlerarten auf nachfolgende Systeme (siehe hierzu auch /DIN 06/ und /HAA 02/.

Wie bei der Fehlerbaumanalyse sollen die Experten zum Anfang einer möglichst aussagekräftigen FMEA die umfassende Kenntnisse zu dem zu betrachtenden Systems (oder technischer Einrichtung) aneignen, um insbesondere der Funktion der Bestandteile (Komponenten) und deren wechselseitige Abhängigkeit zu verstehen. Danach werden die Fehlerarten der individuellen Komponenten erfasst und deren Effekte auf das System untersucht. Dies geschieht auf der Basis einer FMEA-Tabelle. Die Spalten dieser Tabelle sind dann durch Komponente, Fehlerart und Fehlereffekt usw. definiert.

Anschließend folgt die Bewertung der einzelnen Fehlerarten, dem quantitativen Teil der FMEA. Die Bewertungskriterien einer jeden Fehlerart sind üblicherweise:

- Bedeutung der Fehlerart: Schwere des am System verursachten Schadens, bzw. des durch das fehlhafte System verursachten Schadens,
- Auftrittswahrscheinlichkeit,
- Entdeckungswahrscheinlichkeit.

Die Grenzen der Anwendbarkeit einer FMEA ergeben sich aus der geradeaus „bottom-up“-Vorgehensweise der Methode, wobei nur die Auswirkungen von Einzelfehler qualitativ analysiert werden. Eine detaillierte Kenntnis der Systemkomponenten und ihrer Fehlerarten ist Voraussetzungen bei der Durchführung einer FMEA. Die Möglichkeiten zur Entdeckung von Abhängigkeiten in einem System oder von Auswirkungen der Mehrfachversagen von Systemkomponenten sind eingeschränkt.

A.4.3 Risikograph-Methode

Die Risikograph- oder Risikomatrixmethode wird zur Formulierung von Anforderungen an Systeme mit Sicherheitsfunktionen benutzt. Die IEC 61508 /IEC 10/ beschreibt sowohl die Art der Risikobewertung (Risikograph) als auch die Maßnahmen zur Auslegung entsprechender Sicherheitsfunktionen von Sensoren, Logikverarbeitung bis hin zum Aktor bezüglich „Fehlervermeidung“ und „Fehlerbeherrschung“. Die Ermittlung der potentiellen Auswirkungen von Fehlfunktionen oder Ausfällen von sicherheitsrelevanten Funktionen des Systems mithilfe der Risikograph-Methode erfolgt idealerweise in der Konzeptionsphase des Systems. Ziel der Methode ist die Vergabe eines SIL-Levels an das System. In Tabelle A.4 sind die in den verschiedenen SIL-Leveln zulässigen Ausfallraten aufgeführt.

Tab. A.4 Zulässige Fehlerhäufigkeiten bei SIL-Level 1 bis 4

Bei diskreter Betriebsart: Fehlerhäufigkeit pro Zugriff (PFD); bei kontinuierlicher Betriebsart: Fehlerhäufigkeit pro Stunde (PFH).

SIL-Level	Diskrete Betriebsart, PFD	Kontinuierliche Betriebsart, PFH
4	10^{-5}-10^{-4}	10^{-9}-10^{-8}
3	10^{-4}-10^{-3}	10^{-8}-10^{-7}
2	10^{-3}-10^{-2}	10^{-7}-10^{-6}
1	10^{-2}-10^{-1}	10^{-6}-10^{-5}

Diese Methode liefert nur bei hinreichend einfachen Systemen mit unabhängigen Sicherheitsfunktionen zuverlässige und repräsentative Ergebnisse. Des Weiteren ist das Ergebnis einer Risikograph-Methode eine Anforderung an die Ausfallrate des Systems. Es ist nicht dokumentiert, inwiefern die Fehlerrate, die mit der Risikograph-Methode für ein System bestimmt wurde, verifizierbar ist.

A.5 Dynamische Methoden

A.5.1 Zuverlässigkeitsblockdiagramm

Zuverlässigkeitsblockdiagramme (RBD, Reliability Block Diagram) bestehen aus für ein Sicherheitssystem notwendigen Komponenten und deren logischen Verbindungen /DIN 06a/. Jede Komponente des Systems wird dafür durch einen Block repräsentiert und hat zwei definierte Zustände: funktionierend oder ausgefallen. Die Methode erlaubt zudem eine dynamische Beschreibung des Systems, indem jeder Komponente eine Zuverlässigkeitsfunktion $R(t)$ statt dem statischen Zustand zugeordnet wird.

Die Zuverlässigkeitsblockmethode veranschaulicht die Folge von Komponentenausfällen und kann sogar einige statistische Kenngrößen aus der Kenntnis der Ausfallwahrscheinlichkeiten der Einzelkomponenten ableiten. Daher eignet sie sich gut zur Redundanzanalyse und zu quantitativen Analysen einfacher Systeme.

Limitierend wirken bei dieser Methode die einfachen Annahmen für die Komponenten: Entweder sie funktionieren oder sie sind ausgefallen. Eine genauere Bezugnahme auf die konkrete Funktion einer Komponente sowie den Einfluss bzw. die Signifikanz für das System ist nicht möglich.

A.5.2 Markov-Prozesse und Petri-Netze

Eine Markov-Kette oder ein Markov-Prozess ist ein mathematisches Modell, welches stochastische Veränderungen eines zu beschreibenden Systems innerhalb eines definierten Zustandsraumes darstellt. Die besondere Eigenschaft des Modells, die sogenannte Markov-Eigenschaft, ist hierbei, dass das Modell keine (bzw. nur begrenzte Speichermöglichkeiten) besitzt. Dies impliziert, dass die Zustandsänderungen des Modells in der Zukunft nicht (bzw. nur sehr begrenzt) von den vergangenen Ereignissen im System abhängen. Man nennt Markov-Ketten, deren Zukunft nur von den Werten des Systems in der Gegenwart abhängt, Markov-Ketten erster Ordnung. Sie stellen die mathematisch einfachsten Modelle von Markov-Prozessen dar.

Ziel der Modellierung eines Systems mithilfe von Markov-Ketten ist es, verlässliche Aussagen statistischer Natur über das zu beschreibende System zu treffen, z. B. Größen wie Zuverlässigkeit des Systems $R(t)$ (siehe auch Kapitel A.5.1) bzw. die mittlere Dauer zwischen zwei Ausfällen. Dabei werden vor allem jene Systeme gut modelliert, die für eine Vielzahl stochastischer Fehler anfällig sind, deren Verhalten aber nur von Fehlern in der unmittelbaren Vergangenheit essentiell beeinflusst wird. Weitere Informationen zum Einsatz der Markov-Ketten-Analysemethode sind in IEC 61508-7 /IEC 10d/ zu finden.

Die wesentlichen Elemente einer Markov-Kette sind die Zustände des Systems. Ein Zustand des Systems kann hierbei eine Vielzahl an Informationen enthalten, z. B. den Status von Temperatur, Druck und Umlaufgeschwindigkeit des Primärkreislaufes, etc.

Vorteile der Methode sind die hohe Relevanz der Modellparameter für die anschließende statistische Analyse des Systems und die Möglichkeit, dynamische Vorgänge wie Reparaturvorgänge oder transiente Fehler darzustellen. Die Methode kann jedoch – in zuverlässiger Weise – nur in kleinen Systemen mit wenigen Freiheitsgraden angewendet werden, da sich die Lösung der Markov-Kette bei komplexen Systemen als sehr zeitaufwendig darstellt (Zustandsraumexplosion).

Petri-Netze sind ein Werkzeug zum Veranschaulichen und Modellieren von dynamischen Systemen mit wechselseitigen Abhängigkeiten. Sie stellen eine wichtige und moderne Analysemethode für Zuverlässigkeits-Untersuchungen insbesondere in Systemen dar. Die wesentliche Modellierungsstärke von Petri-Netzen basiert auf der Beschreibung des globalen kausalen Verhaltens eines Systems durch Modellieren der

Beziehungen zwischen lokalen Zuständen und lokalen Ereignissen. Im Vergleich zu anderen Zuverlässigkeitsmethoden (z. B. Markov-Methode) erlaubt das eindeutige Modellieren von lokalen Zuständen und auch von lokalen unabhängigen Ereignissen ein angemessenes und intuitives Modellieren von Systemen und deren Dynamik (Störungsbäume oder Zuverlässigkeits-Blockdiagramme). Darüber hinaus wird die Modellierung von stochastischem Verhalten nicht auf exponentialverteilte Eigenschaften beschränkt (Markov-Prozesse), sondern kann frei gewählt werden (s. Die Norm DIN EN 62551 (VDE 0050-4) „Analysemethoden für Zuverlässigkeit – Petri-Netze“).

A.5.3 Dynamic Flowgraph Methodology

Die wesentlichen Eigenschaften der Dynamic Flowgraph Methodology (FDM) wurden bereits ausführlich in /PIL 04/ beschrieben und werden hier nur kurz zusammengefasst. Mit der DFM /GUA 96/ können (Sicherheits-) Systeme modelliert werden. Das Ziel der Modellierung ist es, Ereigniskombinationen zu ermitteln, die zu ausgewählten Top-Ereignissen führen, insbesondere zu Fehlersituationen des Systems. Das DFM stellt die Logik des Systems durch kausale Abhängigkeiten zwischen physikalischen Variablen und Zuständen des Kontrollsystems dar. Die Zustände des Systems werden dabei durch eine Folge von zeitlich diskreten Übergängen modelliert. Das Ergebnis der Modellierung sind sogenannte Primterme („Prime Implicants“), die mit den Minimalschnitten einer Fehlerbaumanalyse verglichen werden können. Jeder dieser Primterme stellt eine Kombination von Systemzuständen dar, die zu einem Top-Ereignis (z. B. zu einem Versagen eines Sicherheitssystems) führen. Die Primterme enthalten dabei die Information, welcher Systemteil sich zu einem bestimmten Zeitpunkt in einem bestimmten Zustand befinden muss.

Ein DFM-Modell kann auf zwei Arten analysiert werden: deduktiv oder induktiv. Bei der induktiven Analyse wird das Verhalten des Systems „vorwärts“ in der Zeit analysiert, d. h. ausgehend von einem Zustand des Systems wird der Zustand für die folgenden Zeitschritte ermittelt. Damit kann überprüft werden, ob es möglich ist, aus einem zulässigen Systemzustand unzulässige Zustände zu erreichen. Bei der deduktiven Analyse wird umgekehrt vorgegangen. Für einen gegebenen Zeitpunkt und Systemzustand (entspricht dem Top-Ereignis eines Fehlerbaums) werden rückwärts die Kombinationen an Systemzuständen ermittelt, die zu dem vorgegebenen Zustand führen können. Es wird gleichsam ein (zeitabhängiger) Fehlerbaum konstruiert, der zu diesem Systemzustand führt.

Wie in /CHU 10/ betont wird, dient die DFM nicht dazu, die Zuverlässigkeits-/Nichtverfügbarkeitsparameter von Systemen zu bestimmen. Sie kann lediglich dazu verwendet werden, wichtige Fehlerzustände des Systems zu ermitteln. Die Auftretenshäufigkeit/Wahrscheinlichkeit von Komponentenausfällen muss anderweitig bestimmt werden. Dies entspricht dem Vorgehen bei der Fehlerbaummodellierung. Auch dort sind die Parameter der Basisereignisse durch geeignete Methoden, wie z. B. der Auswertung der Betriebserfahrung zu ermitteln.

Wie bei der Fehlerbaumanalyse setzt die Anwendung der DFM auf komplexe Systeme die Verfügbarkeit von entsprechenden Tools für die Modellerstellung und Auswertung voraus. Ob die beiden bekannten Tools DYMONDA bzw. YADRAT diese Anforderung erfüllen, konnte mangels Zugangs zu diesen Tools allerdings nicht geklärt werden. Händisch können zwar die Abhängigkeiten dargestellt werden, allerdings ist eine quantitative Analyse ohne Tools nicht möglich.

Die Methode ist sowohl für bestehende Systeme wie auch in der Entwicklungsphase beim Design von Systemen einsetzbar. Sie liefert quantitative Ergebnisse über die Häufigkeit von Top-Ereignissen. Das gleichzeitige Auftreten verschiedener Fehler kann mit der DFM behandelt werden. Durch die Berücksichtigung von Systemzuständen zu verschiedenen Zeiten können auch Sequenzabhängigkeiten und Abhängigkeiten von Ereignissen untereinander berücksichtigt werden.

Die Ergebnisse der DFM, die Primterme, stellen die Kombinationen von Ausfällen bzw. der zeitliche Abfolge dar, die zu einem Top-Ereignis führen. Sie haben damit eine anschauliche Interpretation und können entsprechend vom Analysten auf Plausibilität überprüft werden.

Die Modellentwicklung erfolgt wie bei der Fehlerbaumanalyse Top-Down, d. h. die zu berücksichtigenden Systeme werden ausgehend von einem groben Schema immer weiter verfeinert, bis der nötige Detaillierungsgrad erreicht ist. Dies ermöglicht auch die Zuweisung von Zuverlässigkeitsanforderungen an die einzelnen Subsysteme bzw. Systemkomponenten.

Die Anwendung der DFM setzt voraus, dass der Analyst das System im notwendigen Detaillierungsgrad und in den Abhängigkeiten zwischen den Subsystemen bzw. Systemkomponenten verstanden hat. Die Anforderung an die Anwendung der DFM ist damit vergleichbar zur Anwendung der Fehlerbaumanalyse.

Die DFM ist immer noch Gegenstand von Forschungsarbeiten. Es wurden bisher keine Informationen über den Einsatz der DFM in Genehmigungsverfahren gefunden. Entsprechend ist die Akzeptanz der Methode momentan als gering anzusehen.

A.5.4 Monte-Carlo-Simulation

Monte-Carlo-Simulationen (MCS) können dazu eingesetzt werden, die Wahrscheinlichkeiten verschiedener Systemzuständen komplexer Systeme und Einrichtungen, z. B. Sicherheitssysteme und Einrichtungen in Kernkraftwerken oder Netzwerke, zu berechnen. So ist es möglich, mit Hilfe der MCS die Nichtverfügbarkeit von Sicherheitseinrichtungen und -systemen oder die Auftrittswahrscheinlichkeit einzelner Fehlerarten zu berechnen und zwar auch für den Fall, dass eine analytische Ermittlung nicht möglich ist.

Die Analyse der Zuverlässigkeit eines Systems, z. B. eines Netzwerkes, mit Hilfe der MCS ist vergleichbar der Durchführung eines Experiments /ZIO 13/ zur Ermittlung der Zuverlässigkeit von Werkstücken: Bei einem solchen Experiment wird eine Anzahl zufällig ausgewählter Werkstücke einer Produktionscharge einem Belastungstest ausgesetzt. Aus der Anzahl der den Test nicht bestehenden Werkstücke wird dann eine Statistik über die Zuverlässigkeit aller produzierten Werkstücke abgeleitet.

Bei der MCS wird eine Anzahl von Simulationen des Verhaltens des zu untersuchenden Systems durchgeführt. Die einzelnen Simulationsläufe werden auch als „Spiele“ bezeichnet. Für jedes Spiel wird mit Hilfe von (pseudo-) Zufallszahlen und Verteilungsfunktionen der Verfügbarkeiten bzw. der Ausfallarten der Zustand der einzelnen Komponenten bestimmt. Aus diesem Komponentenzustand (z. B. ausgefallen, eingeschränkt funktionstüchtig, voll funktionstüchtig) wird dann der Zustand des Systems für dieses Spiel ermittelt und protokolliert. Aus der statistischen Auswertung des Systemzustands für alle Spiele können dann die Wahrscheinlichkeiten für verschiedene Systemzustände ermittelt werden.

In jedem Spiel wird die zeitliche Entwicklung der Komponentenzustände simuliert. Eine Simulation durchläuft unterschiedliche Systemzustände, die sich durch den Zustand der einzelnen Systemkomponenten unterscheiden. Die statistische Auswertung der durchgeführten Simulationsläufe ergibt die Nichtverfügbarkeit des Systems.

Die MCS ist auch auf Modelle von Systemen anwendbar, deren Komponenten Übergänge zwischen mehr als zwei Zuständen ausführen, z. B. für die Zustände verfügbar, selbstmeldend ausgefallen und nicht-selbstmeldend ausgefallen. Es ist auch möglich, den zeitlichen Verlauf von (vorbeugenden) Wartungs- und Reparaturvorgängen zu berücksichtigen /ZIO 06/. Dabei kann z. B. auch die Kapazität der Wartungsteams bezüglich der gleichzeitigen Abarbeitung von Reparaturaufträgen und ihrer Priorisierung untereinander berücksichtigt werden. Die Modellierung von verschiedenen Phasen mit unterschiedlichen Verfügbarkeiten von Komponenten ist ebenfalls möglich.

Auch Netzwerke digitaler Leittechniksysteme, die einzelne Komponenten bei einem selbstmeldenden Ausfall automatisch neu starten oder bei der Bestimmung von Grenzwerten oder Anregungen nicht berücksichtigen, können mit Hilfe von MCS analysiert werden. Prinzipiell lassen sich Komponenten mit mehr als zwei Zuständen auch in einer Fehlerbaumanalyse berücksichtigen (siehe das Beispielmodell für die automatische Erstellung von Fehlerbäumen in Anhang A). Eine solche Fehlerbaumanalyse erfordert aber eine sehr große Anzahl an Fehlerbäumen, Fehlerbaumgattern und Basisereignissen. Entsprechend hoch ist der Aufwand der Modellierung, falls keine Unterstützung zur automatisierten Fehlerbaumerstellung vorhanden ist. Des Weiteren wird auch die Anzahl der Minimalschnitte drastisch zunehmen.

Die Anzahl an Spielen, die notwendig sind, um die Verfügbarkeit eines Systems mit ausreichender Genauigkeit zu bestimmen, wächst sehr schnell mit der Anzahl der Komponenten und der Anzahl ihrer Fehlerarten. Treten diese Fehler selten auf, so muss eine größere Anzahl von Spielen durchgeführt werden, damit auch seltene Fehler statistisch ausreichend berücksichtigt werden. Daraus ergeben sich hohe Anforderungen an die notwendigen Rechnerkapazitäten, besonders wenn für jedes Spiel und dessen zeitliche Entwicklung der Systemzustand bestimmt werden soll. Steht eine ausreichende Rechnerkapazität zur Verfügung, so können auch komplexe Systeme simuliert werden[5].

Die Ergebnisse einer Zuverlässigkeitsanalyse mit Hilfe einer MCS können auch zur Auslegung neuer Systeme eingesetzt werden, indem z. B. Ausfälle von Komponenten

[5] Hier unterscheidet sich die MCS wieder wesentlich von einer Markov-Prozessanalyse, da bei der Markov-Prozessanalyse Übergänge zwischen den Systemzuständen berücksichtigt werden. Damit werden Systemausfälle, die aufgrund von seltenen Komponentenausfällen (oder Kombinationen davon) auftreten, besser erfasst.

identifiziert werden, die besonders häufig zu einem Ausfall des Systems führen. Aus dem Ergebnis einer MCS-Analyse können dann auch Anforderungen an die Zuverlässigkeit einzelner Komponenten abgeleitet werden.

Wie beschrieben, liefert eine MCS quantitative Angaben über die Wahrscheinlichkeiten für die Zustände des untersuchten Systems. Es ist auch möglich, quantitative Angaben für die Wahrscheinlichkeitsverteilung von abgeleiteten Größen des Systems zu ermitteln, z. B. der Output einer Produktionsanalage.

Gemeinsam verursachte Ausfälle (GVA) können ebenfalls in einer MCS-Analyse berücksichtigt werden, indem zusätzlich zu den Einzelfehlern der Komponenten auch GVAs bei der Ermittlung der Komponentenzustände berücksichtigt werden.

Da der zeitliche Verlauf der Komponentenzustände innerhalb der Spiele einer MCS berücksichtigt werden, ist es möglich innerhalb eines Spiels den Einfluss von Komponentenausfällen auf das nachfolgende Verhalten anderer Komponenten zu berücksichtigen. So kann z. B. der Ausfall einer Komponente zur höheren Belastung einer anderen Komponente führen, was deren Ausfallverhalten (negativ) beeinflussen und bei ihrer Ausfallwahrscheinlichkeit berücksichtigt werden kann.

Im kerntechnischen Bereich wird die MCS für „Best Estimate Plus Uncertainty"-Analysen im Rahmen von deterministischen Sicherheitsnachweisen und zur Bestimmung von Unsicherheiten der Ergebnisse von probabilistischen Sicherheitsanalysen eingesetzt. Als eigenständige Methode zur Bestimmung von Systemnichtverfügbarkeiten ist die MCS-Methode momentan hauptsächlich Forschungsgegenstand und wird nicht bei Sicherheitsbewertungen der Reaktoranlagen eingesetzt.

Aufgrund des großen Rechenaufwands und der während den MC-Simulationen notwendigen Bestimmung von komplizierten Zusammenhängen zwischen Komponentenzuständen und Systemzustand ist eine Unterstützung durch Softwarewerkzeuge (Tools) zwingend erforderlich. Momentan gibt es hierzu kein allgemein akzeptiertes Werkzeug, wie es z. B. RiskSpectrum® auf dem Gebiet der Fehlerbaumanalyse ist.

Da für jedes Spiel der Zusammenhang zwischen Komponentenzuständen und Systemzustand ermittelt und protokolliert wird, kann die Plausibilität der Ergebnisse standardmäßig überprüft werden.

Einen (internationalen) Standard, wie MCS-Analyse durchgeführt werden soll, gibt es momentan nicht. Als Vorlage bzw. Anleitung können jedoch eine große Anzahl an wissenschaftlichen Veröffentlichungen und Fachbücher (z. B. /ZIO 13/) dienen.

A.6 Vergleich der Methoden

Zuverlässigkeitsanalyseverfahren werden in /DKE 05/ in folgende Kategorien eingeteilt:

- Verfahren zur Vermeidung von Fehlerzuständen, z. B. Beanspruchungsanalyse.
- Verfahren zur Analyse der Systemarchitektur und Zuverlässigkeitsbewertung, z. B.
  - Induktive Verfahren behandeln hauptsächlich die Auswirkungen einzelner Fehlerzustände, u. a. Ereignisbaumanalyse (ETA), Fehlerzustandsart- und -auswirkungsanalyse (FMEA);
  - Deduktive Verfahren können Auswirkungen von Kombinationen von Fehlerzuständen erklären, z. B. Fehlerzustandsbaumanalyse, Markov-Analyse;
- Verfahren zur Schätzung der Maßgrößen für Grundereignisse, z. B. Vorhersage der Ausfallrate, Analyse der menschlichen Zuverlässigkeit, statistische Zuverlässigkeitsverfahren.

Ob die o. g. Verfahren mit Abfolgen von Ereignissen oder zeitabhängigen Eigenschaften arbeiten, stellt ein weiteres Unterscheidungsmerkmal dar. Hiermit ergibt sich die in Tabelle A.5 zusammengefasste Einteilung /DKE 05/.

Tab. A.5 Einteilung der Verfahren zur Schätzung der Maßgrößen für Grundereignisse

	induktiv (nur ein Fehlerzustand)	**deduktiv (mehrfache Fehlerzustände)**
Ablaufabhängig	Ereignisbaumanalyse	Markov, Petri-Netze, Wahrheitstabelle
Ablaufunabhängig	FMEA, Hazard and operability study (HAZOP)	Fehlerbaumanalyse, RBD-Zuverlässigkeits-blockdiagramm

In /DKE 05/ wird erwähnt, dass die Wahl des geeigneten Analyseverfahrens ein sehr individueller Prozess ist, so dass eine allgemeine Empfehlung für die Wahl eines oder

mehrerer spezieller Verfahren nicht gegeben werden kann. Dennoch kann die Wahl anhand folgender Fragen erleichtert werden:

- Systemkomplexität: Handelt es sich um ein komplexeres System, z. B. mit Redundanz oder diversitären Merkmalen, das eine tiefergehende Analyse erfordert als ein einfacheres System?
- Neuartigkeit des Systems: Handelt es sich um einen vollständig neuen Systementwurf, der eine intensivere Analyse erfordert als ein bewährtes System?
- Qualitative oder quantitative Analyse: Ist eine quantitative Analyse notwendig?
- Einzel- oder Mehrfach-Fehlerzustände: Wirken sich Kombinationen von Fehlerzuständen aus oder können diese vernachlässigt werden?
- Von der Zeit oder von Abfolgen abhängiges Verhalten: Spielt das Aufeinanderfolgen von Ereignissen eine Rolle in der Analyse (z. B. fällt das System nur dann aus, wenn dem Ereignis A das Ereignis B vorangegangen ist, aber nicht umgekehrt) oder zeigt das System zeitabhängiges Verhalten (z. B. verschlechterte Betriebsarten nach Ausfall, in Phasen gestaffelte Einsätze)?
- Kann für abhängige Ereignisse verwendet werden: Sind die Ausfall- oder Reparaturmerkmale einer einzelnen Einheit vom Zustand des Systems abhängig?
- Induktive oder deduktive Analyse: Sollte ein induktives Verfahren (Bottom-up: z.B. FMEA-Methode) angewandt werden, das in einer geradlinigen Art durchgeführt werden kann, oder ist ein deduktives Verfahren (Top-down: z.B. Fehlerbaumanalyse) besser geeignet, obwohl es mehr Nachdenken und Kreativität erfordert und daher eher für Fehler anfällig ist?
- Zuweisung der Anforderungen an die Funktionsfähigkeit: Sollte das Verfahren fähig sein, Anforderungen an die Funktionsfähigkeit quantitativ aufzuteilen?
- Erforderlicher Ausbildungsstand: Welcher Ausbildungsgrad oder welche Erfahrung ist erforderlich, um das Verfahren sinnvoll und richtig anzuwenden?
- Akzeptanz und Allgemeingültigkeit: Ist das Verfahren allgemein anerkannt, z. B. von Behörden oder einem Kunden?
- Werkzeugunterstützung: Benötigt das Verfahren (rechnergestützte) Werkzeugunterstützung oder kann es auch von Hand durchgeführt werden?

- Plausibilitätsprüfungen: Kann die Plausibilität der Ergebnisse leicht von Hand nachgeprüft werden? Falls nicht, sind die verfügbaren Werkzeuge validiert?
- Verfügbarkeit von Werkzeugen: Sind Werkzeuge entweder im Hause oder im Handel verfügbar? Haben diese Werkzeuge eine gemeinsame Schnittstelle mit anderen Analysewerkzeugen, damit man Ergebnisse wiederverwenden oder exportieren kann?
- Normung: Gibt es eine Norm, die das Merkmal des Verfahrens und die Darstellung der Ergebnisse (z. B. Symbole) beschreibt?

Entsprechend dieser Kriterien ist in Tabelle A.6 und Tabelle A.7 ein Überblick über die verschiedenen Zuverlässigkeitsanalyseverfahren mit ihren Eigenschaften und Merkmalen aufgeführt. NE (Nicht Empfohlen) symbolisiert dort, dass dieses Verfahren zwar für einfache Systeme verwendet werden kann, als alleiniges Verfahren jedoch nicht empfohlen wird. Dieses Verfahren wird besser in Kombination mit anderen Verfahren eingesetzt. NA (Nicht Anwendbar) bedeutet, dass das betrachtete Kriterium für dieses Verfahren nicht anwendbar ist. Um eine vollständige Analyse eines Systems durchzuführen, kann jedoch mehr als ein Verfahren erforderlich sein /DKE 05/.

In Tabelle A.8 und Tabelle A.9 sind die verschiedenen Methoden im Hinblick auf ihre Anwendbarkeit auf digitale Leittechniksysteme aufgeführt. „Anwendbar“ bedeutet hier, dass das Verfahren allgemein anwendbar ist und für diese Aufgabe empfohlen wird (möglicherweise mit den erwähnten Einschränkungen). „Möglich“ bedeutet, dass das Verfahren für diese Aufgabe verwendet werden kann, es aber gegenüber anderen Verfahren gewisse Nachteile hat. Mit „Unterstützend“ ist gemeint, dass es für einen gewissen Teil der Aufgabe allgemein anwendbar ist, aber nicht als einziges Verfahren für die gesamte Aufgabe. „Nicht anwendbar“ bedeutet, dass es für diese Aufgabe nicht verwendet werden kann. Monte-Carlo-Simulation und Dynamic Flowgraph Methodology sind in Tabelle A.8 und Tabelle A.9 nicht enthalten, da sie in /DKE 05/ nicht betrachtet wurden. Eine entsprechende Bewertung dieser beiden Methoden findet sich in Kapitel A.5.3 und A.5.4.

Tab. A.6 Merkmale der beschriebenen Zuverlässigkeitsanalyseverfahren /DKE 05/

NE Nicht Empfohlen, NA: Nicht Anwendbar

Verfahren	Geeignet für komplexe Systeme	Geeignet für neuartige Systemauslegungen	Quantitative Analyse	Geeignet für Kombinationen von Fehlerzuständen	Geeignet zur Behandlung von Abfolgeabhängigkeit	Kann für abhängige Ereignisse verwendet werden	Deduktiv (D) oder induktiv (I)	Geeignet für Zuverlässigkeitszuweisung[6]	Erforderlicher Ausbildungsstand (von niedrig bis hoch)	Akzeptanz und Allgemeingültigkeit	Braucht Werkzeugunterstützung	Plausibilitätsprüfungen	Verfügbarkeit von Werkzeugen	IEC-Norm
Fehlerbaumanalyse	Ja	Ja	Ja	Ja	Nein	Nein	D	Ja	Mittel	Hoch	Mittel	Ja	Hoch	IEC 61025
Ereignisbaumanalyse	NE	NE	Ja	NE	Ja	Ja	I	NE	Hoch	Mittel	Mittel	Ja	Mittel	
FMEA	NE	NE	Ja	Nein	Nein	Nein	I	NE	Niedrig	Hoch	Niedrig	Ja	Hoch	IEC 60812
Zuverlässigkeitsblockdiagramm	NE	NE	Ja	Ja	Nein	Nein	D	Ja	Niedrig	Mittel	Mittel	Ja	Mittel	IEC 61078
Markov-Analyse	Ja	Ja	Ja	Ja	Ja	Ja	D	Ja	Hoch	Mittel	Hoch	Nein	Mittel	IEC 61165
Petri-Netz-Analyse	Ja	Ja	Ja	Ja	Ja	Ja	D	Ja	Hoch	Niedrig	Hoch	Nein	Niedrig	
DFM	Ja	Ja	Ja	Ja	Ja	Ja		Ja	Mittel	Niedrig	Mittel	Ja	Niedrig	
MCS	Ja	Ja	Ja	Ja	Ja	Ja		Eingeschr.	Hoch	Modellabh.	Hoch	Ja	Niedrig	

[6] Aufteilung der Systemzuverlässigkeit auf die Teilsysteme und Baugruppen /DKE 05/

Tab. A.7 Merkmale weiterer ausgewählter Zuverlässigkeitsanalyseverfahren (Fortsetzung)

NE: Nicht Empfohlen, NA: Nicht Anwendbar

Verfahren	Geeignet für komplexe Systeme	Geeignet für neuartige Systemauslegungen	Quantitative Analyse	Geeignet für Kombinationen von Fehlerzuständen	Geeignet zur Behandlung von Abfolgeabhängigkeit	Kann für abhängige Ereignisse verwendet werden	Deduktiv (D) oder induktiv (I)	Geeignet für Zuverlässigkeitszuweisung	Erforderlicher Ausbildungsstand (von niedrig bis hoch)	Akzeptanz und Allgemeingültigkeit	Braucht Werkzeugunterstützung	Plausibilitätsprüfungen	Verfügbarkeit von Werkzeugen	IEC-Norm
Vorhersage der Ausfallrate	Nein	Ja	Ja	Nein	Nein	Nein	I	Ja	Niedrig	Hoch	Mittel	Ja	Hoch	IEC 61709
PAAG/HAZOP-Untersuchung	Ja	Ja	Nein	Nein	Nein	Nein	I	Nein	Niedrig	Mittel	Niedrig	Ja	Mittel	IEC 61882
Analyse der menschlichen Zuverlässigkeit	Ja	Ja	Ja	Ja	Ja	Ja	I	Nein	Hoch	Hoch	Mittel	Ja	Mittel	
Beanspruchungsanalyse	NA	NA	Ja	NA	NA	Nein	NA	Nein	Hoch	Mittel	Hoch	Ja	Mittel	
Wahrheitstabelle	Nein	Ja	Ja	Ja	Nein	Nein	NA	Ja	Hoch	Mittel	Hoch	Nein	Niedrig	

Tab. A.8 Anwendbarkeit allgemeiner Zuverlässigkeitsanalyseverfahren auf die Leittechniksysteme /DKE 05/

Analyseverfahren	Zuweisung von Zuverlässigkeitsanforderungen/Ziele	Qualitative Analyse	Quantitative Analyse	Bewertung und Empfehlung
Vorhersage der Ausfallrate	Anwendbar auf serielle Systeme ohne Redundanz	Möglich für Instandhaltungsstrategieanalyse	Berechnung der Ausfallraten und der mittleren Betriebsdauer bis zum Ausfall von elektronischen Bauelementen und Geräten	Unterstützend
Fehlerzustandsbaumanalyse bzw. Fehlerbaumanalyse	Anwendbar, falls das System-verhalten nicht stark von der Zeit- oder der Abfolge abhängig ist	Fehlerzustandskombinationen	Berechnung der Systemzuverlässigkeit und zugehöriger Beiträge von Teilsystemen zur Systemnichtverfügbarkeit	Anwendbar
Ereignisbaumanalyse	Möglich	Ausfallfolgen	Berechnung der Systemausfallraten	Anwendbar
Zuverlässigkeitsblockdiagramm	Anwendbar auf Systeme, bei denen voneinander unabhängige Blöcke angenommen werden	Erfolgspfade	Berechnung der Systemzuverlässigkeit	Anwendbar
Markov-Analyse	Anwendbar	Ausfallreihenfolgen	Berechnung der Systemzuverlässigkeit	Anwendbar
Petri-Netz-Analyse	Anwendbar	Ausfallreihenfolgen	liefert die Systembeschreibung für die Markov-Analyse	Anwendbar

Tab. A.9 Anwendbarkeit allgemeiner Zuverlässigkeitsanalyseverfahren auf die Leittechniksysteme /DKE 05/ (Fortsetzung)

Analyseverfahren	Zuweisung von Zuverlässigkeitsanforderungen/Ziele	Qualitative Analyse	Quantitative Analyse	Bewertung und Empfehlung
Fehlerzustandsart- und Auswirkungsanalyse (FMEA)	Anwendbar auf Systeme, in denen ein unabhängiger Einzelausfall vorherrschend ist	Auswirkungen von Ausfällen	Berechnung der Systemausfallraten (und Kritizität)	Anwendbar
PAAG-Verfahren **Gefährdungs- und Betreibbarkeitsuntersuchung (HAZOP)**	Unterstützend	Ursachen und Auswirkungen von Abweichungen	Nicht anwendbar	Unterstützend
Analyse der menschlichen Zuverlässigkeit	Unterstützend	Auswirkung menschlicher Leistung auf den Systembetrieb	Berechnung der Wahrscheinlichkeiten menschlichen Fehlverhaltens	Unterstützend
Beanspruchungsanalyse	Nicht anwendbar	Brauchbar als Mittel zur Vermeidung von Fehlerzuständen	Berechnung der Funktionsfähigkeit von (elektro-) mechanischen Komponenten	Unterstützend
Wahrheitstabelle (Strukturfunktionsanalyse)	Nicht anwendbar	Möglich	Berechnung der Systemzuverlässigkeit	Unterstützend
Statistische Zuverlässigkeitsverfahren	Möglich	Auswirkungen von Fehlerzuständen	Quantitative Schätzung der Funktionsfähigkeit mit Unsicherheiten	Unterstützend

B Weiterentwicklung der Fehlerbaumanalyse-Methodik zur Modellierung redundanter bzw. vernetzter Systeme

B.1 Grundlagen

B.1.1 Fehlerbaummodellierung

Die softwarebasierte Leittechnik nutzt sowohl für redundanzinterne wie auch redundanzübergreifende Kommunikation moderne Netzwerktechnologien. Die nationale und internationale Betriebserfahrung zeigt, dass die interne und externe Kommunikation eines vernetzen Leittechniksystems auf dessen Zuverlässigkeit einen erheblichen Einfluss haben kann (siehe u. a. /GRS 06/). Die Auswirkungen potentieller Fehlereffekte innerhalb und außerhalb der Kommunikationsnetzwerke sind bisher noch nicht systematisch für sicherheitsrelevante Funktionen in Kernkraftwerken im Rahmen einer probabilistischen Sicherheitsanalyse untersucht worden.

Die GRS modellierte und analysierte im Rahmen von PSAs u. a. für die Kernkraftwerke Neckarwestheim 2 /KÖB 01/ und Philippsburg 1 /FRE 06a/ die Ausfälle der analogen Sicherheitsleittechnik. In der PSA Stufe 1 der Referenzanlage Philippsburg 1 /FRE 06a/ wurden erstmalig durch die GRS die potentiellen Ausfälle von einzelnen Hardwarebaugruppen der nachgerüsteten softwarebasierten Leittechnik eines sicherheitsrelevanten Systems (TELEPERM XS) in der Fehlerbaumanalyse modelliert. Auch international wird die Fehlerbaumanalyse für die Modellierung der Ausfälle von digitalen bzw. softwarebasierten[7] Leittechniksystemen eingesetzt (z. B. in Skandinavien /AUT 10/ und Korea /KAN 02/).

In dem oben erwähnten Fehlerbaummodell der Referenzanlage Philippsburg 1 wurde auch die Vernetzung der Redundanzen des digitalen Leittechniksystems miteinander berücksichtigt. Dabei zeigte sich, dass selbst bei vereinfachten Annahmen eine sehr große Anzahl von Fehlerbäumen für eine solche Modellierung notwendig ist (ca. 220 von 1900 Fehlerbäumen der gesamten PSA) /PIL 10/. Dies führte zu einem hohen Aufwand sowohl bei der Modellierung des Systems mit Hilfe der RiskSpectrum®-Software als auch bei der Validierung der erstellten Fehlerbäume. Gleichzeitig waren

[7] Die Begriffe „digitale“ bzw. „softwarebasierte“ Leittechnik werden im Folgenden synonym verwendet.

viele der Fehlerbäume in ihrer Struktur sehr ähnlich bzw. identisch. Während sich die Basisereignisse bzw. Transferelemente je nach untersuchter Redundanz unterschieden, waren jedoch die logischen Verknüpfungen innerhalb der Fehlerbäume identisch.

Dies legte es nahe, eine Methode zur automatisierten Erstellung von Fehlerbäumen zu entwickeln, um zunächst den Aufwand der manuellen Fehlerbaummodellierung zu reduzieren. Dazu kam eine spezielle Programmiersprache, eine sogenannte domänenspezifische Sprache („RiskLang“), zum Einsatz, mit der die Fehlerbäume generiert bzw. verändert werden können. Diese automatisierte Fehlerbaumerstellung wurde anschließend im Rahmen dieses Projekts an einem Referenzsystem getestet. Dazu wurde die Ausfallwahrscheinlichkeit einer Leittechnikfunktion des Referenzsystems mit Hilfe von RiskSpectrum berechnet. Für die Ausfälle der Hardwarekomponenten wurden dabei, soweit möglich, die Daten aus /PIL 10/ verwendet. Die Praktikabilität der Methode der automatisierten Fehlerbaumerstellung konnte im Rahmen dieses Projektes nachgewiesen werden. Die vorläufigen Arbeitsergebnisse zur Weiterentwicklung der Methode der Fehlerbaummodellierung wurden bereits auf den Konferenzen PSAM11 /HER 12a/ und OpenPSA 2012 /HER 12b/ präsentiert.

Die erzielten Ergebnisse könnten prinzipiell auch dazu verwendet werden, um eine untere Grenze für die Sensitivität der Systemnichtverfügbarkeit bezüglich der Auftrittswahrscheinlichkeit von Software-GVAs zu ermitteln. Dies erfordert jedoch eine Modellanpassung.

B.1.2 Software für Fehlerbaummodellierung und –analysen

Bei der Modellierung von Ausfällen in (hoch-)redundanten digitalen, vernetzten Leittechnik-Systemen, wie z. B. von Hardwareausfällen digitaler Sicherheitsleittechnik, mittels der Fehlerbaumanalyse ist eine sehr große Zahl von komplexen, aber strukturell sich wiederholenden Fehlerbäumen notwendig /PIL 10/. Die bisher angewandte manuelle Erstellung aller Fehlerbäume eines redundanten, komplexen Systems ist sehr zeitaufwendig und erfahrungsgemäß potentiell fehleranfällig. Durch eine Automatisierung der Fehlerbaumerstellung können dagegen die verschiedenen Varianten von strukturell ähnlichen Fehlerbäumen direkt durch ein Programm erzeugt werden. Beispielsweise könnte zunächst ein Fehlerbaummodell eines redundanten Teils eines zu analysierenden Systems erstellt und danach dieses Modell automatisch für weitere redundante Teile repliziert werden.

Die im Bereich der Kerntechnik sehr verbreitete und auch von der GRS eingesetzte PSA-Software RiskSpectrum® bietet in bisherigen Versionen nur sehr eingeschränkte Möglichkeiten, die Erstellung von PSA-Modellen zu automatisieren. So ist es lediglich möglich, die Parameter von Basisereignissen (Wahrscheinlichkeiten, Frequenzen, Reparatur- und Testzeiten, usw.) durch einen Im- und Export von/nach Microsoft Excel durch Arbeitsblattfunktionen bzw. Visual Basic Makros zu erstellen oder zu modifizieren. Die Struktur der Fehlerbäume, das heißt die logischen Verknüpfungen und die Basisereignisse selbst, können bisher nicht automatisiert erstellt oder verändert werden, da das bisher vorhandene statische RiskSpectrum® ASCII Format /SÖR 10/ keine automatische Generierung von Fehlerbäumen erlaubt. Es dient dazu, alle Fehlerbäume in einer Datei im Text-Format zu speichern.

In der GRS existieren bisher unterschiedliche Vorgehensweisen zur Generierung von Fehlerbäumen komplexer Systeme. So wurden z. B. für die Modellierung der gemeinsam verursachten Ausfälle (GVA) von redundanten (analogen) leittechnischen Einrichtungen die verschiedenen kombinatorischen Varianten von GVA mit Hilfe von MS Excel ermittelt und zu sogenannten GVA-Modulen zusammengefasst /FRE 06b/. Die diese GVA-Module enthaltenden Fehlerbäume wurden dann manuell erstellt. Die manuelle Umsetzung der Excel-Tabellenwerte in die Fehlerbaummodellierung im Analyseprogramm (z. B. RiskSpectrum®) bedeutet erheblichen Zeitaufwand und birgt zusätzliches Fehlerpotential.

Eine automatisierte Erstellung der Fehlerbäume hat neben der Zeitersparnis den weiteren Vorteil, dass mögliche Fehler bei der manuellen Eingabe verhindert werden und der Validierungsaufwand für die Modellierung reduziert wird. Außerdem erhöht sich die Nachvollziehbarkeit der Modellierung und der Ergebnisse, da die Gemeinsamkeiten und Unterschiede sowie die Verknüpfung zwischen den Redundanzen explizit in dem Programm dargestellt werden, das zur automatischen Erzeugung des Fehlerbaummodells dient.

Ein weiterer Vorteil ist, dass alle Änderungen in der Beschreibung eines Fehlerbaummodells bei der Programmentwicklung mittels Versionskontrollsoftware verfolgt werden können. Die RiskSpectrum®-Software selbst bietet keine Möglichkeiten, den Änderungsverlauf aufzuzeichnen, sondern zeigt den aktuellen Stand eines Fehlerbaummodells, das letzte Änderungsdatum sowie Review- und Freigabeinformationen an.

B.2 Referenzsystem

Um die Implementierung mit der domainenspezifischen Programmiersprache DSL zur Erzeugung von Fehlerbäumen zu testen, wurden mit dem entwickelten Programm die Fehlerbäume für die Modellierung der Leittechnikausfälle eines Referenzsystems erstellt. Das Referenzsystem basiert auf dem in /PIL 10/ beschriebenen generischen Leittechniksystem, das sich an einem TELEPERM XS-Leittechniksystem orientiert. Öffentlich zugängliche Quellen für softwarebasierte Leittechniksysteme von verschiedenen Herstellern können in /IAE 98/, /GRA 06/ und /TEO 11/ gefunden werden.

In Abbildung B.1 ist die Struktur der Leittechnik des Referenzsystems und das Netzwerk zwischen den einzelnen Komponenten dargestellt.

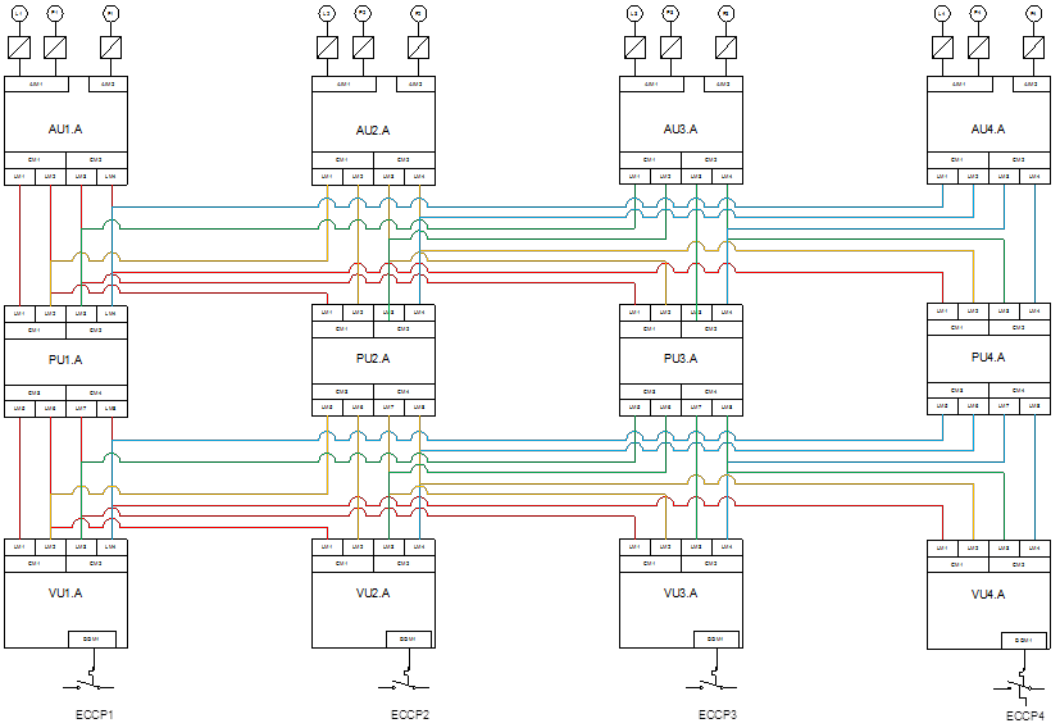


Abb. B.1 Modell generischer Sicherheitsleittechnik des Referenzsystems - Struktur der Hardware

Das Referenzsystem besteht aus vier Redundanzen mit je drei Ebenen. In der ersten Ebene, der Erfassungsebene, werden pro Redundanz jeweils drei Messwerte (Druck, Frischdampfdurchsatz und Füllstand) durch einen Erfassungsrechner (englisch „Acquisition Unit“ (AU)) erfasst und aufbereitet. Die Ergebnisse werden dann jeweils an die zweite Ebene, die Verarbeitungsebene, in allen Redundanzen verteilt. Dort erfolgt eine

Verarbeitung der aufbereiteten Messwerte in einem Verarbeitungsrechner (englisch „Processing Unit“ (PU)).

Aus den vier Messwerten einer physikalischen Größe wird dabei jeweils das zweite Maximum gebildet und mit einem Grenzwert verglichen. Wird der Messwert überschritten, so wird ein Anregesignal für eine Sicherheitsfunktion (hier das Starten einer Notkühlpumpe (englisch „Emergency Core Cooling Pump“ (ECCP)) ausgelöst. Die Anregesignale werden dann an die dritte Ebene, die Voterebene) jeder Redundanz weitergeleitet. Dort werden sie durch einen Voter-Rechner (englisch „Voting Unit“ (VU)) ausgewertet. Die einzelnen Ebenen des Leittechniksystems werden in Kapitel B.2.1 bis B.2.3 detailliert vorgestellt.

Im Folgenden werden zwei Ausfallarten unterschieden: selbstmeldende (englisch „self signaling failure“ (SSF)) und nicht selbstmeldende (englisch „non self signaling failure“ (NSSF)) Ausfälle. Selbstmeldende Ausfälle werden von der Auswertelogik in den Verarbeitungs- und Voter-Rechnern erkannt und berücksichtigt. Nur wenn alle Eingangssignale selbstmeldend ausgefallen sind (und keine weiteren nicht selbstmeldenden Ausfälle vorliegen) kommt es zu einem (selbstmeldenden) Ausfall des entsprechenden Rechners. Nicht selbstmeldende Ausfälle werden hingegen von diesen Rechnern nicht erkannt und können deshalb zu einem unerkannten Ausfall in der Verarbeitungs- bzw. Voter-Ebene führen.

Es wird angenommen, dass alle Hardwareausfälle im Kommunikationsnetzwerk der Sicherheitsleittechnik immer erkannt werden. Fallen Kommunikationskomponenten in der Erfassungsrechnerebene aus, so wird dies spätestens durch die Verarbeitungsrechner erkannt. Ein Hardwareausfall in der Kommunikation der Verarbeitungsebene wird in der Voter-Ebene erkannt. Es wird deshalb im Folgenden davon ausgegangen, dass Hardwareausfälle von Kommunikationsbaugruppen zu selbstmeldenden Ausfällen führen.

Die Fehlfunktion der Hardware der Messeinrichtungen und der Eingangsbaugruppen in der Verarbeitungsebene kann auch zu nicht selbstmeldenden Ausfällen führen, z. B. durch das „Einfrieren“ der Messwerte einer Füllstandsmessung oder durch Fehler in der Signalverarbeitung in den Eingangsbaugruppen.

Die Software der verschiedenen Ebenen (Erfassung, Verarbeitung, Voter) kann sowohl selbstmeldende wie auch nicht selbstmeldende Ausfälle verursachen.

B.2.1 Erfassungsrechner

Im Erfassungsrechner werden die analogen Messwerte erfasst, digitalisiert und nach der Digitalisierung ohne weitere Verarbeitung durch das Netzwerk auf die Verarbeitungsrechner verteilt. Die detaillierte Architektur des Erfassungsrechners des Referenzsystems ist in Abbildung B.2 dargestellt.

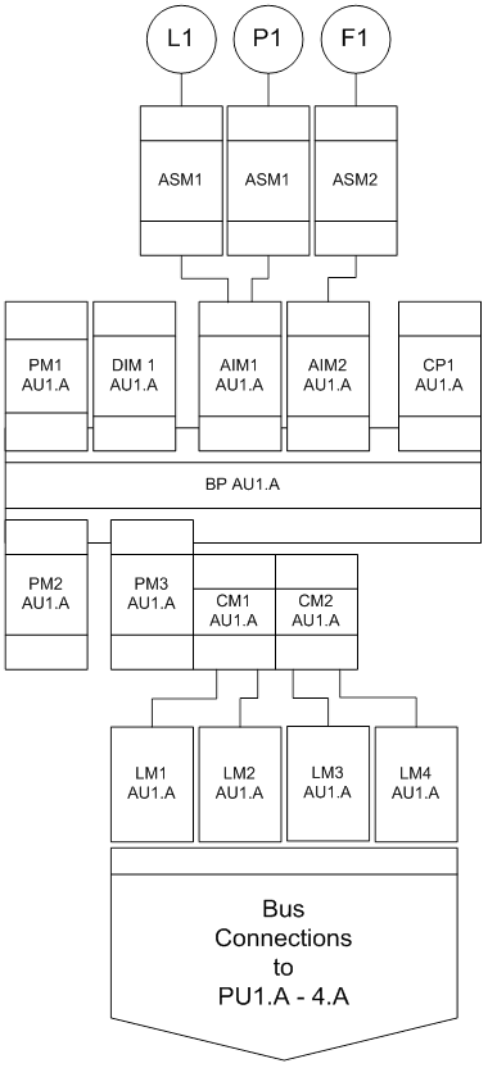


Abb. B.2 Architektur der Erfassungsrechner

Im Erfassungsrechner werden die Stromsignale der Messwerte (L1, P1, F1) mit Hilfe von Stromspannungswandlern (englisch „Analogous Signal Module“ (ASM)) in Spannungswerte umgewandelt. Diese Spannungen werden dann mit Hilfe von Analog-Digital-Wandlern (englisch „Analogous Input Module“ (AIM)) digitalisiert.

Alle Baugruppen des Erfassungsrechners sind in einen Rückwandbus (englisch „Backplane“ (BP)) eines Rechnereinschubs gesteckt.

Die digitalisierten Messwerte werden in einer Prozessorbaugruppe (englisch „Processing Module“ (PM)) verarbeitet. Dann werden sie über eine weitere Prozessorbaugruppe, ein Kommunikationsmodul (englisch „Communication Module“ (CM)) und ein Verbindungsmodul (englisch „Link Module“ (LM)) an die Verarbeitungsebene aller Redundanzen verteilt. Über welche Baugruppen die Kommunikation zwischen einer bestimmten Redundanz in der Erfassungsebene und einer Redundanz in der Verarbeitungsebene stattfindet, ergibt sich aus dem Plan in Abbildung B.1. Für die Fehlerbaumgenerierung wird zusätzlich angenommen, dass sich im Rückwandbus der Erfassungsrechner eine digitale Eingangsbaugruppe (englisch „Digital Input Module“ (DIM)), eine weitere Prozessorbaugruppe sowie ein Kommunikationsprozessor (englisch „Communication Processor“ (CP)), der Störungsmeldungen an die Anzeige- und Meldeeinrichtungen sendet, befinden.

B.2.2 Verarbeitungsrechner

In der Verarbeitungsebene wird im Referenzsystem zunächst aus den Eingangssignalen aus den vier Redundanzen das zweite Maximum für jeden zu überwachenden Parameter ermittelt. Dazu werden die vier Eingangssignale in aufsteigender Reihenfolge sortiert und dann der zweithöchste Wert weiterverwendet. Dies dient dazu, unerkannte Ausreißer der Eingangsdaten in einer Redundanz herauszufiltern. Anschließend wird auf Basis des zweiten Maximums ermittelt, ob ein Grenzwert verletzt wurde und somit eine entsprechende Leittechnikfunktion (LEFU) aktiviert werden soll.

Die Signale werden dabei über Verbindungsmodule, Kommunikationsmodule und ein Prozessmodul von den verschiedenen Redundanzen der Erfassungsrechner empfangen. Die Berechnung des zweiten Maximums und die mögliche Generierung eines Anregesignals erfolgt dann in einem (separaten) Prozessormodul. Die Anregesignale werden dann wieder über ein Prozessormodul, Kommunikationsmodule und Verbindungsmodule an die Voter-Rechner in allen Redundanzen weitergeleitet. Für die Fehlerbaumgenerierung wird zusätzlich angenommen, dass sich im Rückwandbus der Verarbeitungsrechner ein Kommunikationsprozessor als Schnittstelle zu Melde- und Anzeigeeinrichtungen befindet (siehe Abbildung B.3).

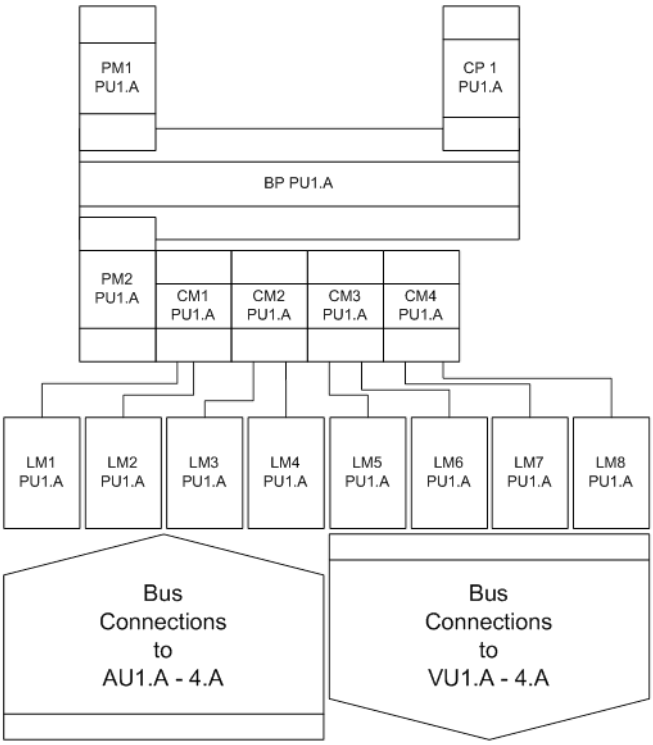


Abb. B.3 Architektur der Verarbeitungsrechner

Auf der Verarbeitungsebene werden mögliche selbstmeldende Ausfälle ermittelt und in der Signalverarbeitungslogik berücksichtigt. Ist z. B. eines der Eingangssignale des Verarbeitungsrechners selbstmeldend ausgefallen, so wird das zweite Maximum nur aus den drei restlichen Signalen des zu überwachenden Parameters gebildet. Sind zwei Eingangssignale selbstmeldend ausgefallen, so wird das zweite Maximum nur aus den beiden restlichen Eingangssignalen des zu überwachenden Parameters gebildet. Sind drei Eingangssignale ausgefallen, so wird das verbleibende Eingangssignal direkt für die mögliche Bildung eines Anregesignals verwendet. Sind alle vier Eingangssignale selbstmeldend ausgefallen, so wird an die Voter-Rechner signalisiert, dass der Erfassungsrechner selbstmeldend ausgefallen ist.

In Tabelle B.1 sind alle möglichen Kombinationen aus intakten, selbstmeldend und nicht selbstmeldend ausgefallenen Eingangssignalen sowie dem sich daraus ergebenden Zustand des Ausgangssignals des Verarbeitungsrechners dargestellt. Es wird dabei immer konservativ angenommen, dass ein nicht selbstmeldend ausgefallenes Eingangssignal so ausgefallen ist, dass es, wenn möglich, zu einem nicht selbstmeldenden Ausfall des Ausgangssignals kommt.

Tab. B.1 Verhalten der zweiten Maxima-Berechnung in der Anwendersoftware der Verarbeitungsrechner bei Ausfallkombinationen der Eingänge

Anzahl der Eingänge in Zustand			**Zweites Maximum**
Nicht selbstmeldend ausgefallen	**intakt**	**Selbstmeldend ausgefallen**	**Zustand**
-	4	-	intakt
-	3	1	intakt
-	2	2	intakt
-	1	3	Intakt
-	-	4	selbstmeldend ausgefallen
1	3	-	intakt
1	2	1	intakt
1	1	2	nicht selbstmeldend ausgefallen
1	-	3	nicht selbstmeldend ausgefallen
2	2	-	intakt
2	1	1	nicht selbstmeldend ausgefallen
2	-	2	nicht selbstmeldend ausgefallen
3	1	-	nicht selbstmeldend ausgefallen
3	-	1	nicht selbstmeldend ausgefallen
4	-	-	nicht selbstmeldend ausgefallen

B.2.3 Voter-Rechner

Im Voter-Rechner werden die Anregesignale mit Hilfe einer 2 aus 4 Auswahlmethode verarbeitet. Liegt an mindestens 2 der 4 Eingangssignale ein Anregesignal vor, so wird die Leittechnikfunktion, z. B. „Start eine Notkühlpumpe“, aktiviert. Die detaillierte Architektur ist in Abbildung B.4 dargestellt.

Die Anregesignale der Verarbeitungsrechner werden über Verbindungsmodule, Kommunikationsmodule und ein Prozessormodul empfangen. Die Voting-Verarbeitung findet in einem anderen Prozessormodul statt. Eine mögliche Anregung erfolgt über ein Digitalausgabemodul (englisch „Digital Output Module“ (DOM)). Für die Fehlerbaumgenerierung wird zusätzlich angenommen, dass sich im Rückwandbus der Verarbeitungsrechner ein Kommunikationsprozessor als Schnittstelle zu Melde- und Anzeigeeinrichtungen befindet.

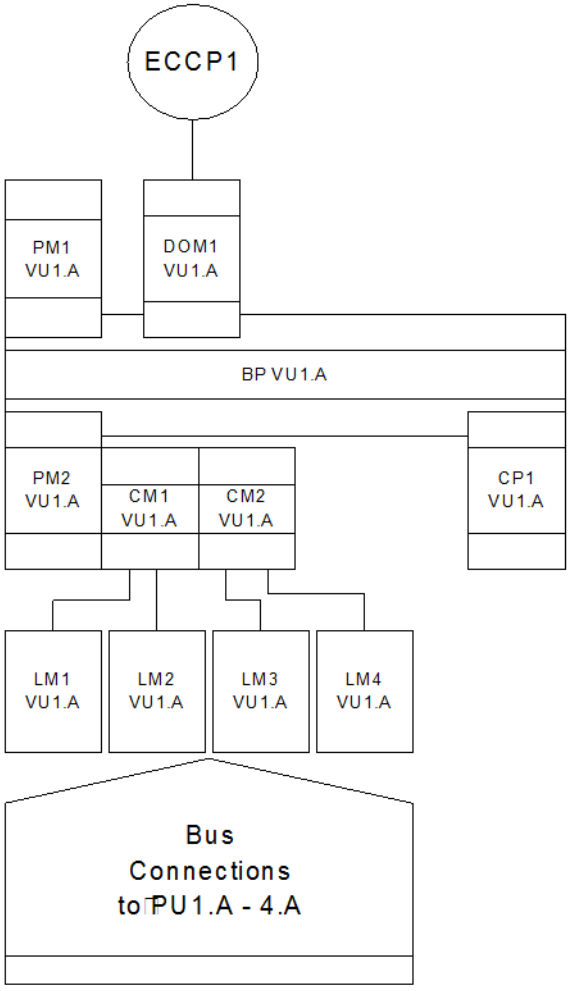


Abb. B.4 Architektur der Voter-Rechner

Fällt eines der Eingangssignale eines Voter-Rechners selbstmeldend aus, so werden die restlichen drei Anregesignale mit Hilfe einer 2 aus 3 Schaltung verarbeitet. Fallen zwei Eingangssignale aus, so erfolgt eine mögliche Anregung mit einer 1 aus 2 Schaltung. Fallen drei Eingangssignale selbstmeldend aus, so wird das verbleibende Signal direkt für eine mögliche Anregung verwendet.

In Abbildung B.2 sind die Zustände des Ausgangssignals der Voter-Rechner in Abhängigkeit der Anzahl der intakten, selbstmeldend und nicht selbstmeldend ausgefallenen Eingangssignale aufgeführt. Dabei wurde wieder konservativ angenommen, dass ein nicht selbstmeldender Ausfall des Eingangssignals, wenn möglich, zu einem nicht selbst-meldenden Ausfall des Ausgangssignals führt.

Tab. B.2 Verhalten der 2 aus 4 Auswahl in der Anwendersoftware der Voter-Rechner bei Ausfallkombinationen der Eingänge

Anzahl der Eingänge in Zustand			2 aus 4 Auswahl
Nicht selbstmeldend ausgefallen	**intakt**	**Selbstmeldend ausgefallen**	**Zustand**
-	4	-	intakt
-	3	1	intakt
-	2	2	intakt
-	1	3	Intakt
-	-	4	selbstmeldend ausgefallen
1	3	-	intakt
1	2	1	intakt
1	1	2	intakt
1	-	3	nicht selbstmeldend ausgefallen
2	2	-	intakt
2	1	1	nicht selbstmeldend ausgefallen
2	-	2	nicht selbstmeldend ausgefallen
3	1	-	nicht selbstmeldend ausgefallen
3	-	1	nicht selbstmeldend ausgefallen
4	-	-	nicht selbstmeldend ausgefallen

B.3 Fehlerbaumbeschreibungssprache RiskLang

Bei RiskLang handelt es sich um eine domänenspezifische Programmiersprache (DSL). Sie dient dazu, Aufgaben in einem bestimmten Problemfeld (einer sogenannten „Domäne“) zu lösen. Dazu enthält die Sprache Eigenschaften, die speziell auf ihre Domäne ausgerichtet sind /GHO 11/.

Ziel von „RiskLang“ ist die Erstellung bzw. Modifizierung von Fehlerbäumen. RiskLang wurde in der GRS mit Hilfe der Programmiersprache Ruby /FLA 08/ entwickelt, wobei der Ansatz einer sogenannten internen DSL gewählt wurde. Dadurch stehen die üblichen Programmstrukturen wie Schleifen oder Kontrollstrukturen von Ruby der

RiskLang-Anwendung zu Verfügung. Fehlerbäume können dann als Programme in der neu entwickelten DSL dargestellt und in RiskSpectrum®-Modelle überführt werden.

Die Programmiersprache Ruby ist eine interpretierte und objektorientierte Programmiersprache, die seit 1995 (weiter-)entwickelt wird. Sie ist eine General Purpose Language („Allzweckprogrammiersprache“), die unter anderem für Datenbankprogrammierung gut geeignet ist. Es gibt verschiedene Interpreter für Ruby, die als Open Source veröffentlicht wurden. Für die Programmiersprache selbst läuft ein Standardisierungsprozess /ISO 12/.

B.3.1 Spezifikation von RiskLang durch die GRS

RiskLang fügt Ruby drei Befehle hinzu, um Basisereignisse bzw. logische Verknüpfungen, Fehlerbaumknoten und Fehlerbäume zu erzeugen bzw. zu modifizieren. In Tabelle B.3 sind die Befehle mit ihren jeweiligen Argumenten aufgeführt. Ein Basisereignis bzw. eine logische Verknüpfung wird mit dem Befehl *Event* erzeugt, oder, falls es bereits ein solches mit derselben ID (Kennung) und demselben Typ gibt, modifiziert. Die möglichen Typen für Basisereignisse bzw. Verknüpfungen sind in Tabelle B.4 aufgeführt. Optional können ein Beschreibungstext eingefügt sowie Angaben zur RiskSpectrum® internen Modellierung und Berechnung gemacht werden.

Um Basisereignisse bzw. Verknüpfungen in einen Fehlerbaum einzufügen, müssen diese mit Hilfe des Kommandos *FTNode* einem Fehlerbaumknoten zugewiesen werden. Der Befehl *FTNode* muss eines der beiden Argumente: *Event* oder: *Transfer* besitzen, um anzuzeigen, ob es sich um ein Basisereignis/eine Verknüpfung handelt oder um einen Verweis auf das TOP-Element eines anderen Fehlerbaums. (Existiert dieses, durch ein Transferelement referenziertes TOP-Element noch nicht, so wird es neu erzeugt). Zusätzlich können optional Argumente angegeben werden, die die Positionierung des Fehlerbaumknotens in der grafischen Darstellung beeinflussen. Die Eingangsknoten für logische Verknüpfungen werden mit Hilfe des Arguments: *Children* spezifiziert. Dabei handelt es sich um ein Datenfeld (in Ruby) aus einem oder mehreren *FTNode*.

Tab. B.3 Spezifikation der RiskLang-Befehle

Kommando	Argumente	Datatypen	Notwendig
FT	*:ID*	Zeichenkette (maximale Länge: 20)	Ja
	:Top	*FTNode*	Ja
	:Text	Zeichenkette (maximale Länge: 100)	Optional
FTNode	*:Event*	*Event*	Ja (genau eines von beiden)
	:Transfer	*Event*	
	:Pos	ganze Zahl (Vorgabe: 0)	Optional
	:InLevel	ganze Zahl (Vorgabe: 1)	Optional
	:Children	Datenfeld aus *FTNode*	Optional
Event	*:ID*	Zeichenkette (maximale Länge: 20)	Ja
	:Type	*Event Type*	Ja
	:Text	Zeichenkette (maximale Länge: 100)	Optional
	:Model	ganze Zahl (Vorgabe: 0)	Optional
	:CalcType	ganze Zahl (Vorgabe: 1)	Optional

Tab. B.4 Mögliche Typen von Basisereignissen bzw. Verknüpfungen in RiskLang

Event Types	RiskSpectrum® Equivalent
:circle	Basisereignis
:diamond	Basisereignis (mit Diamand als Symbol)
:house	House Event
:orgate	ODER Verknüpfung
:andgate	UND Verknüpfung
:kofn	K aus N Verknüpfung
:norgate	NICHT-ODER Verknüpfung
:nandgate	NICHT-UND Verknüpfung
:xorgate	Exklusives ODER Verknüpfung
:comment	Kommentarverknüpfung
:continue	Fortsetzungsverknüpfung
:notfound	Platzhalter, falls RiskSpektrum-Datenbank inkonsistent

Fehlerbäume werden mit Hilfe des Kommandos *FT* erzeugt bzw. modifiziert, falls schon ein Fehlerbaum mit der gleichen ID (Kennung) existiert. Zusätzlich zur ID muss

der Fehlerbaumknoten angegeben werden, der den TOP-Knoten des Fehlerbaums darstellt. Optional kann ein Beschreibungstext für den Fehlerbaum hinzugefügt werden.

Jeder der drei Befehle liefert ein Ruby-Objekt des entsprechenden Typs zurück, der entweder einer Variablen zugewiesen oder direkt als entsprechendes Argument für einen der anderen Befehle verwendet werden kann.

Zusätzlich bietet RiskLang die Möglichkeit, RiskSpectrum® Attribute zu erzeugen und Fehlerbäumen bzw. Basisereignissen/Verknüpfungen zuzuweisen. Damit können Ergänzungen bzw. Änderungen, die durch RiskLang vorgenommen wurden, in der RiskSpectrum® Datenbank markiert werden.

In Abbildung B.5 ist ein einfacher Beispielfehlerbaum dargestellt. Dieser kann mit Hilfe des RiskLang Programms in Abbildung B.6 erzeugt werden.

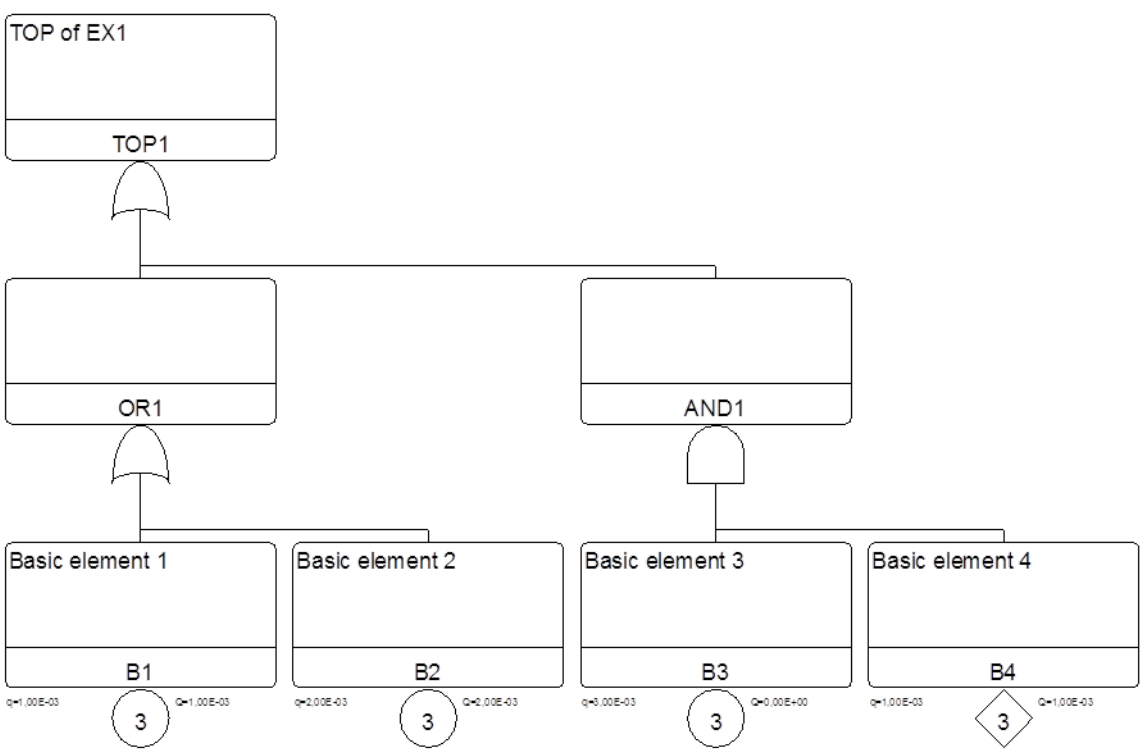


Abb. B.5 Beispielfehlerbaum

```
require "RiskRobot/ParseFT"
RiskSpectrumConnection.init('C:\temp\k2.rpp')

FT(:ID=>"EX1", ::Text=>"Example fault tree", ::top=>
  FTNode(:Event=>Event(:ID=>"TOP1", ::Type=>:orgate, ::CalcType=>1,
    ::Text=>"TOP of EX1"), ::Pos=>1, ::InLevel=>0, ::Children=>[
      FTNode(:Event=>Event(:ID=>"OR1", ::Type=>:orgate, ::CalcType=>1),
        ::Pos=>1, ::InLevel=>1, ::Children=>[
          FTNode(:Event=>Event(:ID=>"B1", ::Type=>:circle, ::Model=>3,
          :CalcType=>1,
            ::Text=>"Basic element 1"), ::Pos=>1, ::InLevel=>1),
          FTNode(:Event=>Event(:ID=>"B2", ::Type=>:circle, ::Model=>3,
          :CalcType=>1,
            ::Text=>"Basic element 2"), ::Pos=>2, ::InLevel=>1)
      ]),
      FTNode(:Event=>Event(:ID=>"AND1", ::Type=>:andgate, ::CalcType=>1),
        ::Pos=>3, ::InLevel=>1, ::Children=>[
          FTNode(:Event=>Event(:ID=>"B3", ::Type=>:circle, ::Model=>3,
          :CalcType=>1,
            ::Text=>"Basic element 3"), ::Pos=>3, ::InLevel=>1),
          FTNode(:Event=>Event(:ID=>"B4", ::Type=>:diamond, ::Model=>3,
          :CalcType=>1,
            ::Text=>"Basic element 4"), ::Pos=>4, ::InLevel=>1)
      ])
  ])
)
```

Abb. B.6 RiskLang Programm, um den Fehlerbaum in Abb. B.5 zu erzeugen

Das RiskLang Programm in Abbildung B.6 verwendet dabei direkt die von den einzelnen Befehlen zurückgegebenen Objekte als Argumente für die umschließenden Befehle. Die Reihenfolge, in der die Befehle ausgeführt werden, ergibt sich aus der Art und Weise, wie Ruby das Programm verarbeitet. Die Ausführung erfolgt von den „innersten“ Befehlen zu den „äußeren“. Zuerst werden in RiskSpectrum® die Basisereignisse erzeugt, dann die sie verwendenden Fehlerbaumknoten, dann die logischen Verknüpfungen, die die Basisereignisse als Eingabesignale verwenden, dann das Top-Ereignis und zum Schluss der Datenbankeintrag für den Fehlerbaum.

Die ersten beiden Zeilen des RiskLang Programms in Abbildung B.6 dienen dazu, den Parser, der die Befehle von RiskLang verarbeitet und daraus Ruby Code bzw. Objekte generiert, zu starten sowie eine Verbindung zu einer bestimmten RiskSpectrum®-Datenbank herzustellen.

B.3.2 Implementierung von RiskLang

Die Implementierung von RiskLang durch die GRS verwendet mehrere Schichten, um die Zugriffe auf die Datenbank der PSA-Anwendung, z. B. RiskSpectrum®, zu abstrahieren. Dadurch ist es möglich, dass RiskLang Programme mit verschiedenen Versionen von RiskSpektrum® und zukünftig mit relativ wenigen Anpassungen auch mit an-

deren PSA-Anwendungen zusammenarbeiten können. So war bis vor kurzem bei der GRS noch die Version RiskSpectrum® PSA Professional im Einsatz, die mittlerweile durch die Version RiskSpectrum® PSA 1.2.1.1 ersetzt wurde. Die beiden Versionen basieren auf unterschiedlichen Datenbanken und werden beide von RiskLang unterstützt. Zukünftig könnte es notwendig werden, auch mit anderen PSA-Anwendungen (z. B. FinPSA/STUK Finnland, Saphire/NRC USA) zusammenzuarbeiten.

In Abbildung B.7 ist die Architektur der aktuellen RiskLang Implementierung dargestellt. Auf dem höchsten Abstraktionsniveau befinden sich die eigentlichen RiskLang Programme. Sie repräsentieren die Fehlerbäume und ihre Bestandteile.

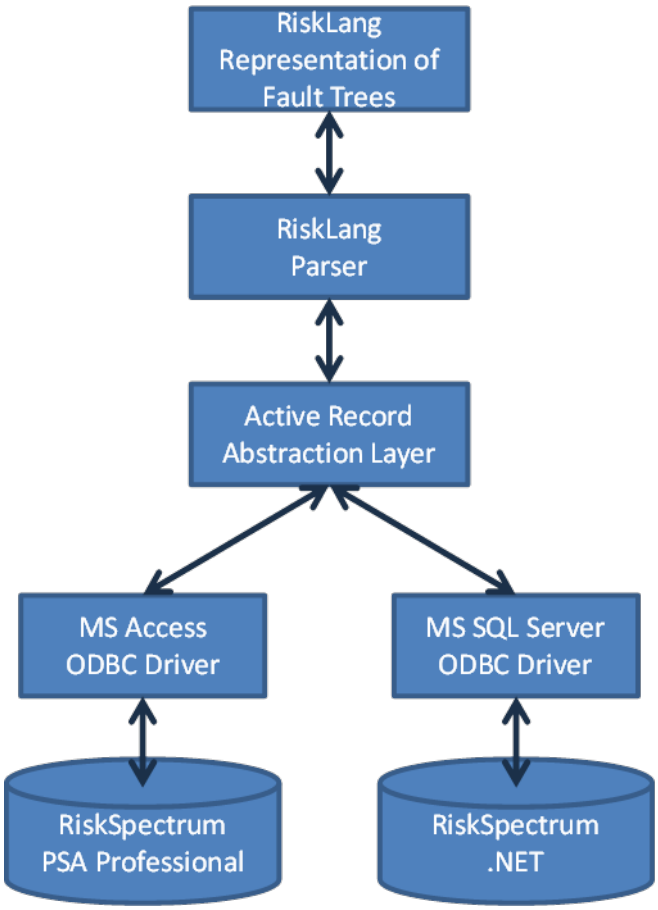


Abb. B.7 Implementierung von RiskLang

In der Schicht darunter befindet sich der Parser für RiskLang, der die Befehle von RiskLang verarbeitet und daraus Ruby-Code bzw. Objekte generiert. Diese Schicht ermöglicht es dem Anwender, Fehlerbaummodelle direkt in Ruby einzubetten und RiskLang Programme direkt im Interpreter von Ruby auszuführen. Es sind keine (expliziten) Übersetzungsschritte notwendig. Die Parserschicht verwendet eine weitere Abs-

traktionsebene, die die Ruby Bibliothek ActiveRecord /MAR 07/ verwendet. Diese setzt das Entwurfsmuster „ActiveRecord“ /FOW 03/ um, bei dem Objekte der Programmiersprache (hier Ruby) auf Datenbankeinträge abgebildet werden. Werden solche Objekte in Ruby erzeugt, gelöscht oder verändert, werden die entsprechenden Vorgänge in der Datenbank übernommen. Die oberen drei Schichten sind dabei unabhängig von der eingesetzten PSA-Software sowie deren Datenbankformat.

Für die Erprobung mit RiskSpectrum® wurden Anbindungen an eine ältere und die aktuelle Version der Software implementiert. Die ältere RiskSpectrum Professional Version verwendet als Datenbank Backend Microsoft Access, die aktuelle RiskSpectrum Version Microsoft SQL Server Express. Diese werden über entsprechende ODBC[8]-Treiber angesprochen.

Um die Datenbank von RiskSpectrum® in einem konsistenten Zustand zu halten und um Änderungen nachvollziehbar zu machen, sind die folgenden Automatismen implementiert:

- Neu hinzugefügte Objekte bzw. Änderungen in der Datenbank werden mit einer eindeutigen Benutzerkennung („RiskRobot“) und dem Änderungsdatum gekennzeichnet.
- Alle Änderungen werden markiert („tagged“).
- In der neuen Version von RiskSpectrum® werden, falls vorhanden, die Informationen über Review und Freigabe („approved“) zurückgesetzt.

B.3.3 Automatische Generierung von RiskLang Code

Um den Anwender bei der Erstellung von RiskLang Programmen zu unterstützen, gibt es die Möglichkeit, Fehlerbäume aus bestehenden RiskSpectrum®-Modellen als RiskLang Code zu exportieren. Dadurch ist es z. B. möglich, Fehlerbäume für hochredundante Systeme erst für eine Redundanz in RiskSpectrum grafisch zu modellieren, und dann als RiskLang Code zu exportieren. Anschließend können die exportierten

8 OBDC: Open Database Connectivity, standardisierte Datenbankschnittstelle, die SQL als Datenbanksprache verwendet.

Fehler-bäume als Vorlage dienen, um in RiskLang die Fehlerbäume in allen Redundanzen, sowie die Vernetzung zwischen den Redundanzen zu modellieren.

Zur Validierung von RiskLang wurden auch die 1902 Fehlerbäume der PSA einer Referenzanlage /LIN 06/ nach RiskLang exportiert, die Fehlerbäume in RiskSpectrum gelöscht und anschließend durch das Ausführen des RiskLang Programms wieder neu erstellt. Es wurde anschließend festgestellt, dass die Struktur der Fehlerbäume wie auch die Ergebnisse der quantitativen Berechnungen identisch waren mit den unveränderten Fehlerbäumen der RiskSpectrum®-Datenbank.

B.3.4 Einschränkungen bei der Implementierung von RiskLang

RiskSpectrum® sieht von Hause aus die Möglichkeit vor, Parameter, House Events , Exchange Events und Einstellungen für die quantitativen Analysen durch einen Im- und Export von/nach Microsoft Excel vorzugeben. Deshalb wurde beim Entwurf von RiskLang darauf verzichtet, diese Daten ebenfalls in RiskLang abzubilden.

B.4 Anwendung von RiskLang für das Referenzsystem

Die GRS hat die Anwendbarkeit von RiskLang an Hand der Modellierung der Ausfälle der digitalen Leittechnik eines generischen Referenzsystems erprobt. Hierzu wurden die Fehlerbäume für die Nichtverfügbarkeit des Referenzsystems aus Kapitel B.2 mit Hilfe von RiskLang modelliert und anschließend quantitative Testrechnungen in RiskSpectrum® durchgeführt.

Für die Nichtverfügbarkeiten der Hardwarekomponenten wurden die Daten aus /PIL 10/ verwendet. Dazu wurden aus /PIL 10/ jeweils Komponenten mit gleicher oder (möglichst) ähnlicher Funktion identifiziert wie die Komponenten im Referenzsystem. Falls keine entsprechende Komponente identifiziert werden konnte, so wurde das Basisereignis auf logisch „falsch“ gesetzt, so dass es bei den quantitativen Berechnungen ignoriert wird.

Im Folgenden werden exemplarisch Fehlerbäume und der entsprechende Code in RiskLang vorgestellt.

Als Ausfall des Gesamtsystems wird angenommen, dass drei oder vier der vier im Modell vorhandenen Notkühlpumpen nicht starten. Ein Pumpenstart wird ausgelöst, falls für mindestens eine der drei vorhandenen physikalischen Parameter (Druck, Füllstand, Frischdampfdurchsatz) ein Anregesignal erzeugt wird (durch Verarbeitungsrechner und Voter).

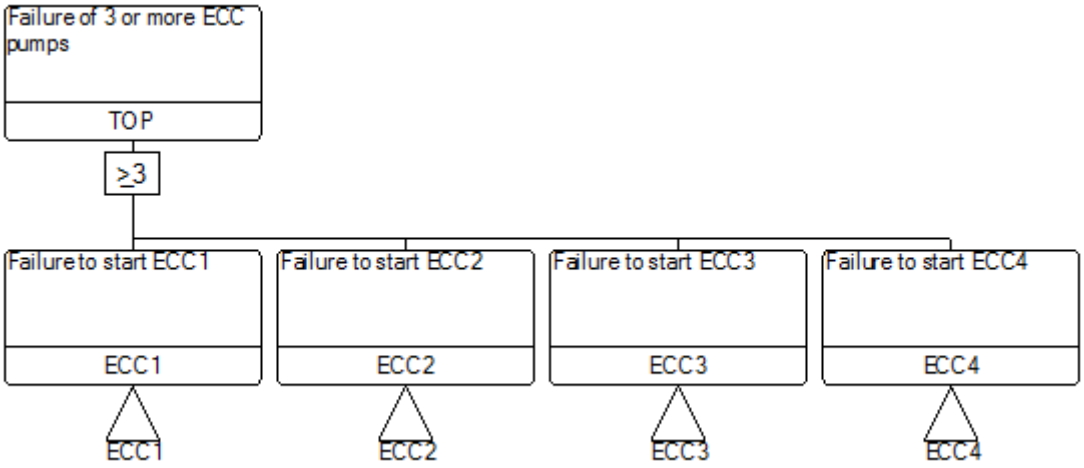


Abb. B.8 Fehlerbaum für das Top-Ereignis

Das RiskLang Programmfragment in Abbildung B.9 erzeugt den Fehlerbaum in Abbildung B.8. Das Top-Ereignis tritt ein, falls drei oder vier der vier vorhandenen Notkühlpumpen nicht starten.

```
FT(:ID=>"TOP",·:top=>¶
··FTNode(:Event=>Event(:ID=>"TOP",·:Type=>:kofn,·:Model=>3,·:CalcType=>1,·¶
·········:Text=>"Failure·of·3·or·more·ECC·pumps"),·:Pos=>1,·:InLevel=>0,·¶
·········:Children=>[¶
··········FTNode(:Transfer=>Event(:ID=>"ECC1",·:Type=>:andgate,·:CalcType=>1,·¶
·····················:Text=>"Failure·to·start·ECC1"),·:Pos=>1,·:InLevel=>1),¶
··········FTNode(:Transfer=>Event(:ID=>"ECC2",·:Type=>:andgate,·:CalcType=>1,·¶
·····················:Text=>"Failure·to·start·ECC2"),·:Pos=>2,·:InLevel=>1),¶
··········FTNode(:Transfer=>Event(:ID=>"ECC3",·:Type=>:andgate,·:CalcType=>1,·¶
·····················:Text=>"Failure·to·start·ECC3"),·:Pos=>3,·:InLevel=>1),¶
··········FTNode(:Transfer=>Event(:ID=>"ECC4",·:Type=>:andgate,·:CalcType=>1,·¶
·····················:Text=>"Failure·to·start·ECC4"),·:Pos=>4,·:InLevel=>1)¶
·········]¶
··)¶
)¶
```

Abb. B.9 RiskLang Programm, um den Fehlerbaum in Abb. B.8 zu erzeugen

In Abbildung B.10 ist der Fehlerbaum für das Startversagen der Notkühlpumpe in der ersten Redundanz dargestellt. Es gibt strukturell gleiche Fehlerbäume für das Pumpenstart-versagen in den anderen drei Redundanzen. Alle vier Fehlerbäume können mit dem RiskLang Programmfragment in Abbildung B.11 generiert werden.

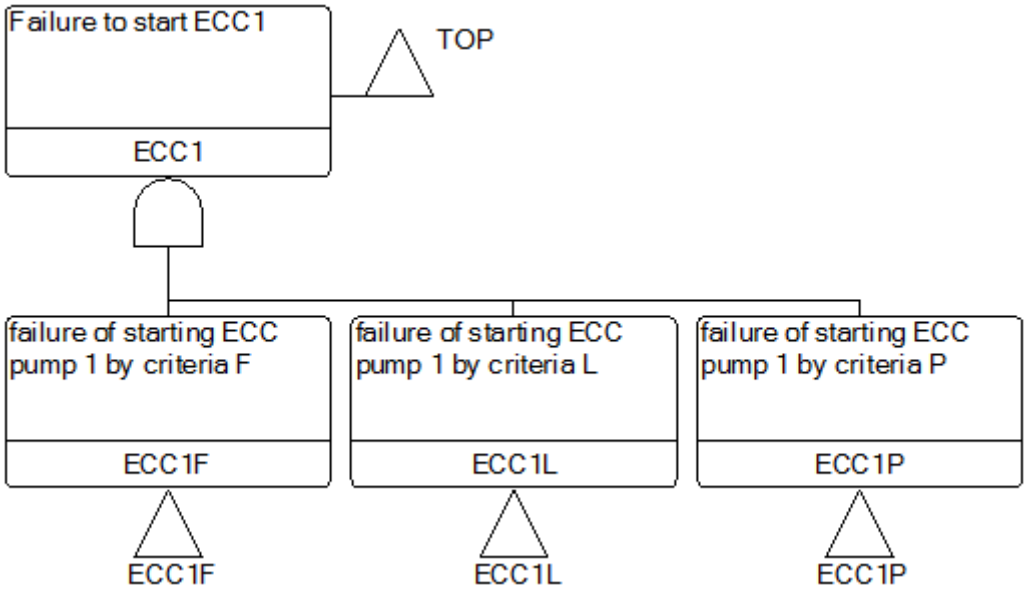


Abb. B.10 Fehlerbaum für das Startversagen der Pumpe ECCP1

```
(1..4).each do |r|
  FT(:ID=>"ECC#{r}", :top=>
    FTNode(:Event=>Event(:ID=>"ECC#{r}", :Type=>:andgate, :CalcType=>1,
      :Text=>"Failure to start ECC#{r}"), :Pos=>1, :InLevel=>0,
      :Children=>[
        FTNode(:Transfer=>Event(:ID=>"ECC#{r}F", :Type=>:orgate, :CalcType=>1,
          :Text=>"failure of starting ECC pump #{r} by criteria F"),
          :Pos=>1,
          :InLevel=>1),
        FTNode(:Transfer=>Event(:ID=>"ECC#{r}L", :Type=>:orgate, :CalcType=>1,
          :Text=>"failure of starting ECC pump #{r} by criteria L"),
          :Pos=>2,
          :InLevel=>1),
        FTNode(:Transfer=>Event(:ID=>"ECC#{r}P", :Type=>:orgate, :CalcType=>1,
          :Text=>"failure of starting ECC pump #{r} by criteria P"),
          :Pos=>3,
          :InLevel=>1)
      ])
  )
end
```

Abb. B.11 RiskLang Programm, um den Fehlerbaum in und strukturell gleiche für die restlichen drei Redundanzen zu erzeugen

Dies ist ein erstes Beispiel, wie eine Kombination aus RiskLang und Eigenschaften von Ruby dazu genutzt werden können, die Erstellung von Fehlerbäumen zu automatisieren. Eine Schleife im Programmcode iteriert über alle vier Redundanzen und erzeugt den jeweiligen Fehlerbaum. Dabei werden sowohl die Kennungen der Fehlerbäume und Basisereignisse/logischen Verknüpfungen, wie auch die Beschreibungstexte automatisch für die jeweilige Redundanz angepasst. In den Zeichenketten für die Kennung der Basisereignisse, logischen Verknüpfungen und Fehlerbäume sowie in deren

Beschreibungstexten wird dabei automatisch der Platzhalter `#{r}` durch die Nummer der jeweiligen Redundanz ersetzt.

In Abbildung B.12 ist der Fehlerbaum dargestellt, der einen Ausfall des Starts der ersten Notkühlpumpe entweder durch eine fehlende Anregung durch das Frischdampfdurchsatzsignal oder aufgrund von Hardwareausfällen des Voter der ersten Redundanz generiert.

Als Beispiele für Hardwareausfälle im Voter werden ein Auslösen der Sicherung in der Stromversorgung der Leittechnikschränke oder Ausfälle der Ausgabebaugruppe berücksichtigt. Im Fehlerbaum in Abbildung B.12 ist der selbstmeldende Ausfall aller Eingangssignale (entweder durch einen Ausfall der Kommunikation oder einen selbstmeldenden Ausfall des jeweiligen Verarbeitungsrechners) modelliert. Außerdem wird der nicht selbstmeldende Ausfall des Voter berücksichtigt.

Mögliche Ausfälle der Anregung durch die 2 aus 4 Auswahl des Voter können durch verschiedene Kombinationen von Ausfällen an den Eingängen des Voter bedingt sein (siehe Tabelle B.2). Diese Kombinationen sind die Eingangssignale der ODER-Verknüpfung in Abbildung B.13. Kodiert sind die Transferverknüpfungen mit den Nummern und den Ausfallarten der Verarbeitungsrechner. „N“ steht für einen nicht selbstmelden-den, „S“ für einen selbstmeldenden Ausfall. Beispielsweise bedeutet die Zeichenkette 2O4_VU1.A_2S3N4N, dass das Signal vom Verarbeitungsrechner 2 selbstmeldend ausgefallen ist (entweder weil der Verarbeitungsrechner ein solches Signal generiert, oder weil keine Kommunikation mit ihm möglich ist) und dass die Signale von den Verarbeitungsrechnern 3 und 4 nicht selbstmeldend ausgefallen sind.

Der Fehlerbaum in Abbildung B.13 (und die strukturell gleichen für die restlichen 3 Redundanzen, jeweils für die drei zu berücksichtigenden analogen Messsignale) kann durch den RiskLang Code in Abbildung B.14 erzeugt werden. Zunächst wird für alle 3^4 = 81 Kombinationen der möglichen Eingangssignale (3 verschiedene Zustände: intakt, selbst-meldend und nicht selbstmeldend ausgefallen) überprüft, ob diese zu einem nicht selbstmeldenden Ausfall des Voter führen. Falls ja, wird die entsprechende Transfer-verknüpfung neu generiert und in einem Feld zwischengespeichert. Nachdem alle Kombinationen durchgetestet wurden, wird der Fehlerbaum zusammengesetzt.

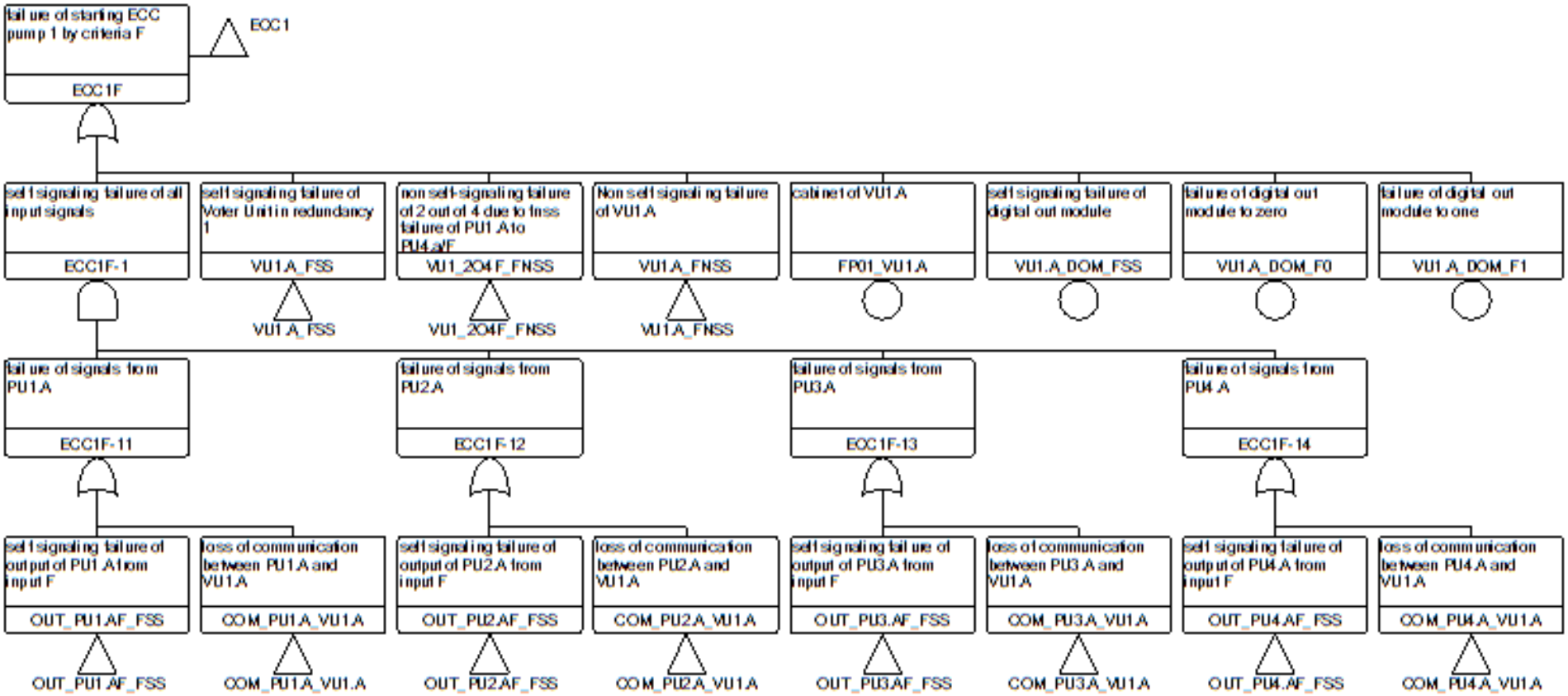


Abb. B.12 Fehlerbaum für das Startversagen der Pumpe ECCP1 (aufgrund fehlender Anregung durch das Frischdampfdurchsatzsignal oder Ausfall des Voter in der ersten Redundanz).

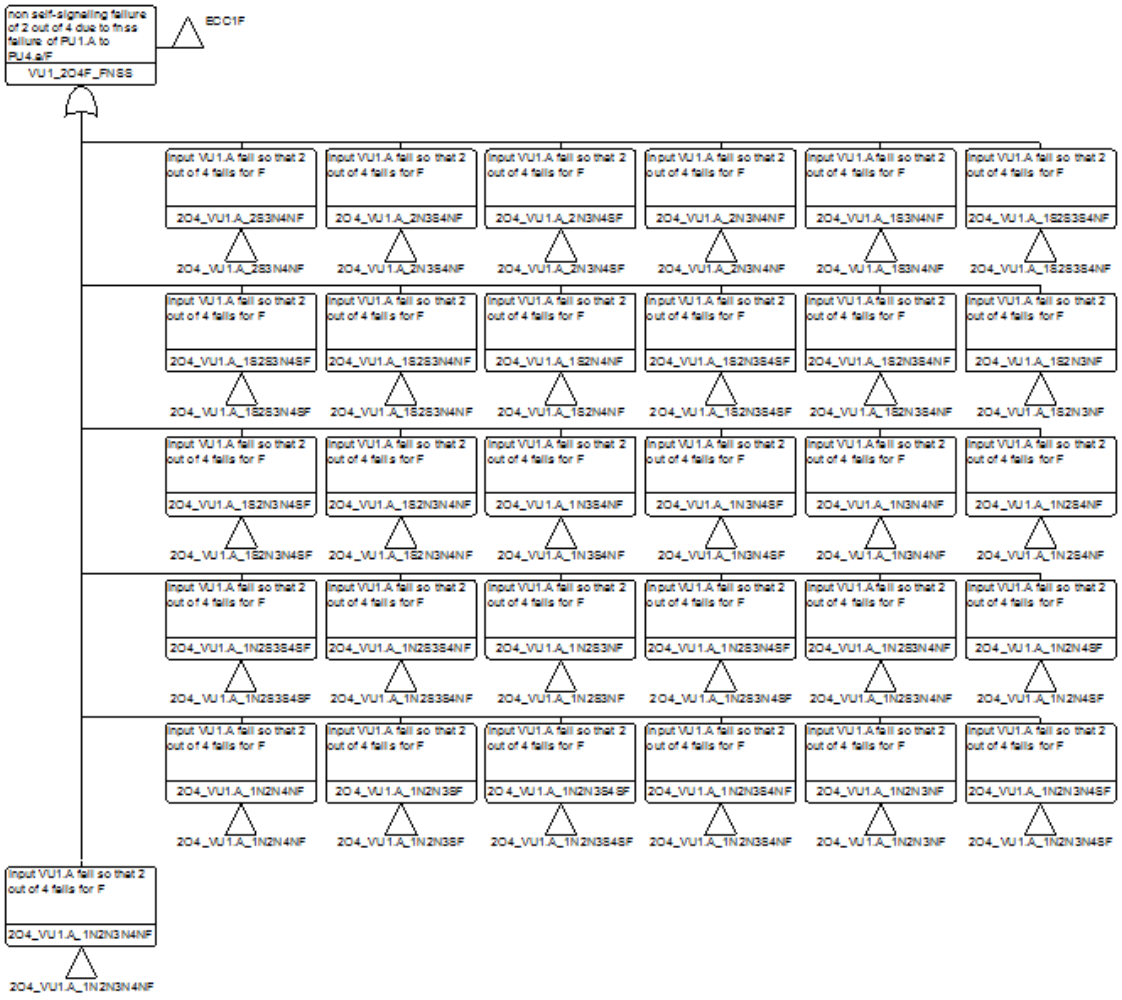


Abb. B.13 Fehlerbaum des nicht selbstmeldenden Ausfalls des Voters der ersten Redundanz aufgrund einer fehlenden Anregung durch das Frischdampfdurchsatzsignal

In Tabelle B.5 sind die 20 Fehlerbaumtypen aufgeführt, die zur Modellierung der Nichtverfügbarkeit des Referenzsystems notwendig sind. In den Kennungen der Fehlerbäume werden dabei Platzhalter für die Redundanz (#{r}, und falls eine Kommunikation zwischen Redundanzen stattfindet, #{r2}), für die analoge Messung (#{m}), sowie für die Kombinationen der Eingangssignale des zweiten Maximums bzw. der 2 aus 4 Auswertung (#{comb}) verwendet. Insgesamt werden für die Modellierung 1053 Fehlerbäume verwendet.

Für alle Fehlerbäume zusammen sind ca. 600 Zeilen Code erforderlich, um sie in RiskLang unter Verwendung der vorgestellten Ruby Spracheigenschaften darzustellen.

Tab. B.5 Alle Fehlerbäume zur Modellierung des Referenzsystems

Fehlerbaum Kennung (Platzhalter: #{r}, #{r2}, #{m}, #{comb})	Beschreibung	Anzahl Fehlerbäume
2O4_VU#{r}.A_#{comb}_#{m}	ODER aus allen Kombinationen, die zu einem Ausfall der 2 aus 4 Auswahl führen	372
AU#{r}.A_#{m}_FNSS	Nicht selbstmeldender Ausfall eines Erfassungsrechners	12
AU#{r}.A_FSS	Selbstmeldender Ausfall eines Erfassungsrechners	4
COM_AU#{r}.A_PU#{r2}.A	Ausfall der Kommunikation zwischen einem Erfassungs- und einem Verarbeitungsrechners	16
COM_PU#{r}.A_VU#{r2}.A	Ausfall der Kommunikation zwischen einem Verarbeitungs- und einem Voter-Rechners	16
ECC#{r}	Ausfall des Starts einer Notkühlpumpe	4
ECCP#{r}_#{m}	Ausfall des Starts einer Notkühlpumpe durch fehlerhafte Auswertung einer Messung	12
F2M_PU#{r}.A_#{comb}_#{m}	ODER aus allen Kombinationen, die zu einem Ausfall der Auswertung des zweiten Maximums führen	516
OUT_AU#{r}.A_#{m}_FNSS	Nicht selbstmeldender Ausfall des Ausgangssignals eines Erfassungsrechners	12
OUT_AU#{r}.A_#{m}_FSS	Selbstmeldender Ausfall des Ausgangssignals eines Erfassungsrechners	12
OUT_PU#{r}.A_#{m}_FNSS	Nicht selbstmeldender Ausfall des Ausgangssignals eines Verarbeitungsrechners	12
OUT_PU#{r}.A_#{m}_FSS	Selbstmeldender Ausfall des Ausgangssignals eines Verarbeitungsrechners	12
PU#{r}.A_FNSS	Nicht selbstmeldender Ausfall Verarbeitungsrechner	4
PU#{r}.A_FSS	Selbstmeldender Ausfall eines Verarbeitungsrechners	4
PU#{r}_F2M_#{m}_FNSS	Nicht selbstmeldender Ausfall der Auswertung des zweiten Maximums durch Eingangsfehler	12
SIG_RED#{r}_FSS_#{m}	Selbstmeldender Ausfalls v. Analogeingangsbaugruppe	12
TOP	3 oder 4 Notkühlpumpen starten nicht	1
VU#{r}.A_FNSS	Nicht selbstmeldender Ausfall eines Voter-Rechners	4
VU#{r}.A_FSS	Selbstmeldender Ausfall eines Voter-Rechners	4
VU#{r}_2O4_#{m}_FNSS	Nicht selbstmeldender Ausfall der Auswertung der 2 aus 4 Auswahl durch Eingangsfehler	12

```
(1..4).each·do·|r|¶
··["L",·"P",·"F"].each·do·|m|¶
····states·=·[:ok,·:fss,·:fnss]¶
····subtrees·=·[]¶
····combine_arrays(states,·states,·states,·states)·do·|a,b,c,d|¶
·····if·TwoOutOfFour.fnss([a,b,c,d])·then·#·2·out·of·4·fails·non·self·signaling¶
·····comb·=·comb_to_str([a,b,c,d])¶
·····subtrees·<<·Event(:ID=>"204_VU#{r}.A_#{comb}#{m}")¶
·····end¶
····end·#·|a,b,c,d|¶
····FT(:ID=>"VU#{r}_204#{m}_FNSS",·:top=>¶
·····FTNode(:Event=>Event(:ID=>"VU#{r}_204#{m}_FNSS",·:Type=>:orgate,¶
·····:Text=>"..."),·#·removed·text·¶
·····:Pos=>1,·:InLevel=>0,·¶
·····:Children=>create_nodes(subtrees,·6,·"VU#{r}_204#{m}_FNSS",¶
·····[:Transfer]*subtrees.length)¶
·····)¶
····)¶
··end¶
end¶
```

Abb. B.14 RiskLang Programm, um den Fehlerbaum in Abb. B.13 und strukturell gleiche Fehlerbäume für die restlichen drei Redundanzen zu erzeugen

Es zeigt sich, dass 20 Fehlerbaumtypen ausreichen, um das Referenzsystem zu beschreiben. Allerdings ist die Anzahl der Fehlerbäume, die sich aus den verschiedenen Kombinationen von Redundanzen, Messgrößen und Ausfallkombinationen ergibt, ca. 50-mal so groß. Eine manuelle Erstellung einer solchen Zahl von Fehlerbäumen ist schon für dieses relativ einfache Beispiel nur mit sehr großem Aufwand möglich.

Bei einer automatischen Erstellung der Fehlerbäume ist es hingegen möglich, die 20 generischen Fehlerbaumtypen mit Hilfe von RiskSpectrum® grafisch zu erstellen. Diese 20 generischen Fehlerbaumtypen sollen dann vollständig validiert werden. Anschließend können die generischen Fehlerbaumtypen in RiskLang exportiert werden, dort die Verallgemeinerungen für alle möglichen Redundanzen, Messgrößen und Kombinationen von Ausfällen vorgenommen und anschließend daraus ein vollständiges RiskSpectrum® Modell erstellt werden. Änderungen, wie z. B. das Hinzufügen einer weiteren Redundanz oder einer weiteren Messgröße, sind bei dieser Vorgehensweise (relativ) einfach möglich.

B.5 Quantitative Erprobung der Fehlerbaummodellierung mit RiskLang

Die GRS hat den RiskLang Ansatz nicht nur qualitativ, sondern auch quantitativ erprobt. Um zu testen, ob das mit RiskLang erstellte Fehlerbaummodell von RiskSpectrum® richtig verarbeitet wird, werden für verschiedene Fehlerbaumanalysefälle die

Ausfallwahrscheinlichkeiten für die modellierten Leittechnikfunktionen (Nichtverfügbarkeit bei Anforderung) berechnet. Die Eingabe der probabilistischen Kennwerte der einzelnen Komponenten erfolgte dabei über die RiskSpectrum® Importschnittstelle für Microsoft Excel Dateien.

Zunächst wurde die Nichtverfügbarkeit der Notkühlpumpe der vierten Redundanz aufgrund der fehlerhaften Verarbeitung der Anregung durch einen Parameter (Frischdampfdurchsatz) berechnet. Dabei wurden alle Basisereignisse auf FALSCH gesetzt, für die keine Parameter für die Nichtverfügbarkeiten gefunden werden konnten. Dies gilt besonders für alle potentiellen Einzelfehler der Software innerhalb einer Redundanz und für gemeinsam in allen Redundanzen gleichzeitig auftretende Softwarefehler (GVA).

In Tabelle B.6 sind die 15 Minimalschnitte aufgeführt, die in diesem Fall den größten Bei-trag zur Nichtverfügbarkeit liefern.

Tab. B.6 Minimalschnitte mit den höchsten Beiträgen

Nichtverfügbarkeit	**%**	**Basisereignisse**		
3,82E-04	54,12	VU4.A_PM1_FNSS	-	-
8,02E-05	11,35	VU4.A_BP_FSS	-	-
6,29E-05	8,9	VU4.A_PM1_FSS	-	-
6,29E-05	8,9	VU4.A_PM2_FSS	-	-
4,66E-05	6,59	VU4.A_DOM_F0	-	-
2,24E-05	3,18	VU4.A_CM12_FSS	-	-
2,02E-05	2,86	VU4.A_DOM_FSS	-	-
1,19E-05	1,69	VU4.A_PM1_BP	-	-
5,62E-06	0,8	VU4.A_PM2_BP	-	-
5,62E-06	0,8	VU4.A_CP1_BP	-	-
4,08E-06	0,58	FP01_VU4.A	-	-
1,13E-06	0,16	VU4.A_DOM1_BP	-	-
2,54E-07	0,04	VU4.A_DOM_F1	-	-
1,55E-08	0	F-SENSOR-2_FNSS	F-SENSOR-3_FNSS	F-SENSOR-4_FNSS
1,55E-08	0	F-SENSOR-1_FNSS	F-SENSOR-2_FNSS	F-SENSOR-4_FNSS

Für den Fall, dass keine gemeinsam verursachten Ausfälle berücksichtigt werden, tragen Einzelausfälle der Hardwarekomponenten des Voter-Rechners in der Redundanz 4 mit über 99,7 % zur gesamten Nichtverfügbarkeit bei. Die Beiträge der Komponenten der Erfassungs- und Verarbeitungsrechner hingegen sind vernachlässigbar.

Wird die Nichtverfügbarkeit des Fehlerbaums aus Abbildung B.8 berechnet, so erhält man einen sehr niedrigen Wert von 4,5E-9. Zu diesem Wert tragen im Wesentlichen ca. 100 Kombinationen von Ausfällen verschiedener Hardwarekomponenten gleichwertig bei. Insgesamt ist aber die Wahrscheinlichkeit eines Ausfalls des gesamten Referenzleittechniksystems sehr niedrig. Dies liegt sowohl an der vierfachen Redundanz von Komponenten der Leittechnikhardware und den drei diversitären Messsignalen, die bei der Anregung des Starts der Notkühlpumpe berücksichtigt werden als auch an den Annahmen hinsichtlich der Erkennung der Hardwareausfälle (u.a. selbstmeldende Ausfälle der Hardware der Kommunikationseinrichtungen).

Die Ermittlung der Minimalschnitte zur Berechnung der Nichtverfügbarkeit des Referenzsystems mit Hilfe von RiskSpectrum® erfordert auf einem typischen PC ca. 2 bis 3 Stunden. Dieser Zeitaufwand ist sehr hoch verglichen mit einer Analyse für ein typisches verfahrenstechnisches System, die im Wesentlichen die Ausfälle von systemtechnischen Komponenten (Ventile, Pumpen, Motoren, Stromversorgung) berücksichtigt. Der Grund dafür liegt in der starken Vernetzung des Referenzsystems. Für ein typisches verfahrenstechnisches System kann der Algorithmus von RiskSpectrum® zur Ermittlung der Minimalschnitte Fehlerbaummodule identifizieren. Diese sind Teilfehlerbäume, die mehrfach im Gesamtfehlerbaum vorkommen. Die in den Teilfehlerbäumen enthaltenen Basisereignisse dürfen dabei im Rest des Fehlerbaums nicht mehr auftauchen. Die Fehlerbaummodule können dann durch ein Basisereignis mit der Unverfügbarkeit des Teilfehlerbaums ersetzt werden. Dadurch entsteht (sukzessive) ein vereinfachtes (internes) Fehlerbaummodell. Damit sinkt der Aufwand bei der Bestimmung der Minimalschnitte durch RSAT[9] gegebenenfalls erheblich, entsprechend verringert sich die Rechenzeit. Durch die starke Vernetzung des Referenzsystems sind solche Vereinfachungen nicht möglich. Entsprechend verlängert sich der Zeitaufwand für die Analyse.

9 RSAT ist der Bestandteil von RiskSpectrum®, mit dem die Minimalschnitte ermittelt werden.

Die starke Vernetzung des Leittechniksystems hat aber auch zur Folge, dass die Bedeutung möglicher, redundanzübergreifender Ausfälle, die durch potentielle Softwarefehler verursacht werden, steigt. Sollte ein solcher softwarebedingter Ausfall in allen vier Redundanzen gleichzeitig eine Wahrscheinlichkeit in der Größenordnung oder größer als 10^{-8} haben, so würde er die Häufigkeit des Top-Ereignisses dominieren.

In /KAM 13/ wurde ebenfalls eine Sensitivitätsanalyse zum Einfluss von GVA auf die Nichtverfügbarkeit eines softwarebasierten Reaktorschnellabschaltsystems untersucht. Unter den dort getroffenen Annahmen beeinflusst ein Software-GVA mit einer Auftrittswahrscheinlichkeit in der Größenordnung von 10^{-7} bis 10^{-5} die Nichtverfügbarkeit des Gesamtsystems signifikant. Allerdings wird in dem dort untersuchten System angenommen, dass ein Software-GVA durch eine diversitäre Architektur des Systems nicht zum Ausfall des gesamten Systems führen kann.

Ob und wie redundanzübergreifende softwarebedingte Ausfälle modelliert, quantifiziert oder ausgeschlossen werden könnten, ist weiterhin Gegenstand der Forschung. Abschließende Ergebnisse hierzu liegen bisher nicht vor. Dennoch bietet Fehlerbaumtechnik prinzipiell die Möglichkeit die Auswirkungen softwarebedingter Ausfälle zumindest auf der Systemebene (z. B. Basisereignis „Ausfall einer Leittechnik-Funktion auf Grund Software-GVA in einer Redundanz“) zu berücksichtigen.

B.6 Schlussfolgerung

Die GRS hat ein Softwarewerkzeug entwickelt und erprobt, um die Fehlerbaummodellierung redundanter und vernetzter digitaler Leittechniksysteme in der PSA-Software RiskSpectrum® zu vereinfachen. Mit Hilfe der DSL „RiskLang“ ist es nun möglich, ein Fehlerbaummodell für ein komplexes, vernetztes System effizient und nachvollziehbar zu erstellen. Dies gilt besonders, falls die Fehlerbäume aus Variationen von wenigen generischen Fehlerbäumen erzeugt werden können.

Der hohe Grad an Redundanz im Referenzsystem sorgt dafür, dass die Nichtverfügbarkeit dieses Systems ohne Berücksichtigung von redundanzübergreifenden Ausfällen (GVA), z. B. der Software, so niedrig ist, dass einzelne Ausfallkombinationen der Hardware in Bezug auf das Gesamtergebnis der PSA vernachlässigbar wären. Erst die Modellierung der GVA in der Hard- und Software digitaler Sicherheitsleittechnik kann zu relevanten Beiträgen führen. Die Fehlerbaummodellierung liefert einen guten An-

satz, die Einzelausfälle, die GVA und deren Kombinationen nachvollziehbar in der Analyse zu berücksichtigen. Hierzu kann die Anwendung von RiskLang wichtige Unterstützung sowohl für Modellierung, als auch für Validierung und Dokumentation der bereits vorhandenen bzw. zu modifizierenden Fehlerbaumanalysen leisten.

RiskLang bietet zusammen mit der zugrundeliegenden Programmiersprache Ruby aber noch weitere Möglichkeiten, die teilweise sogar schon getestet werden konnten. So wurde RiskLang bereits erfolgreich eingesetzt, um eine PSA der Stufe 1 um brandspezifische Basisereignisse zu erweitern /TÜR 13/. Es ist angedacht, in einem Forschungsprojekt zu untersuchen, ob damit auch die Brandausbreitung direkt in der PSA-Software durch entsprechende Fehlerbäume bestehend aus Brandereignissen in Räumen und den Übergangswahrscheinlichkeiten zwischen diesen modelliert werden.

Durch die Implementierung von RiskLang in Schichten mit unterschiedlichem Abstraktionsgrad der zugrunde liegenden Datenbank des PSA-Tools ist es auch möglich, RiskLang für andere PSA-Tools (als RiskSpectrum®) zu implementieren.

Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) gGmbH

Schwertnergasse 1
50667 Köln
Telefon +49 221 2068-0
Telefax +49 221 2068-888

Forschungszentrum
85748 Garching b. München
Telefon +49 89 32004-0
Telefax +49 89 32004-300

Kurfürstendamm 200
10719 Berlin
Telefon +49 30 88589-0
Telefax +49 30 88589-111

Theodor-Heuss-Straße 4
38122 Braunschweig
Telefon +49 531 8012-0
Telefax +49 531 8012-200

www.grs.de

ISBN 978-3-944161-58-7