# Auswirkungsbereiche von Softwarefehlern in sicherheitstechnisch wichtigen Einrichtungen von Kernkraftwerken

GRS - 456

Gesellschaft für Anlagen-
und Reaktorsicherheit
(GRS) gGmbH

# Auswirkungsbereiche von Softwarefehlern in sicherheitstechnisch wichtigen Einrichtungen von Kernkraftwerken

Manuela Jopen
Hervé Mbonjo
Dagmar Sommer
Birte Ulrich

März 2021
mit Addenda September 2018
und August 2021

**Anmerkung:**

Das diesem Bericht zugrunde liegende Forschungsvorhaben wurde mit Mitteln des Bundesministeriums für Umwelt, Naturschutz, Bau und Reaktorsicherheit (BMUB) unter dem Förderkennzeichen 3614R01304 durchgeführt.

Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei der GRS.

Der Bericht gibt die Auffassung und Meinung der GRS wieder und muss nicht mit der Meinung des BMUB übereinstimmen.

**GRS - 456**
**ISBN 978-3-946607-39-7**

**Addenda**

Gegenüber der Version vom März 2017 wurde auf Seite 103 eine Fußnote eingefügt.
Gegenüber der Version vom September 2018 wurden auf den Seiten 109, 110, 113 Fußnoten und auf den Seiten II, IV Textabsätze (3. Abs.) eingefügt. Das Literaturverzeichnis wurde um die Literaturstelle /GRS 21/ ergänzt.

**Deskriptoren**

Betriebserfahrung, Auswirkungen in die Leittechnik, Auswirkungen von Softwarefehler,
COCS-Klassifizierungsschema, sicherheitstechnische wichtige Einrichtungen der Kernkraftwerke

## Abstract

This report presents results that have been developed within a BMUB-funded research project (Promotion Code 3614R01304). The overall objective of this project was to broaden the knowledge base of GRS regarding software failures and their impact in software-based instrumentation and control (I&C) systems.

To this end, relevant definitions and terms in standards and publications (DIN, IEEE standards, IAEA standards, NUREG publications) as well as in the German safety requirements for nuclear power plants were analyzed first. In particular, it was found that the term "software fault" is defined differently and partly contradictory in the considered literature sources. For this reason, a definition of software fault was developed on the basis of the software life cycle of software-based I&C systems within the framework of this project, which takes into account the various aspects relevant to software faults and their related effects. It turns out that software failures result from latent faults in a software-based control system, which can lead to a non-compliant behavior of a software-based I&C system. Hereby a distinction should be made between programming faults and specification faults.

In a further step, operational experience with software failures in software-based I&C systems in nuclear facilities and in nonnuclear sector was investigated. The identified events were analyzed with regard to their cause and impacts and the analysis results were summarized. Based on the developed definition of software failure and on the COMPSIS-classification scheme for events related to software based I&C systems, the COCS-classification scheme was developed to classify events from operating experience with software failures, in which the events are classified according to the criteria "cause", "affected system", "impact" and "CCF potential". This classification scheme was applied to evaluate the events identified in the framework of this project. As a result, most software failures are due to programming faults and/or specification faults. Software failure can also result from the interaction of the control system with its operating environment, e.g. due to hardware fault. Another type of software failure is related to faults that result from a faulty system requirement specification. Such errors are independent of the realization of the control system (analog or software-based).

Further, potential effects of software failure types identified by evaluation of the operating experience with software-based I&C system were investigated for specific I&C application functions and their associated controlled process equipment.

Various concepts have been developed for this purpose. The developed concepts have in common that, first, individual software faults are postulated in a designated software-based I&C function, and their effects are then analyzed. Essentially the developed concepts differ in their implementation in the achievable level of detail in the modeling of the software faults of the I&C functions or of the systems and their linking with the controlled process equipment. Using the example of a software-based neutron flux measurement system in a pressurized water reactor it was shown that postulated software faults in the computation of neutron flux signals can lead to the blockage of the reactor trip via neutron flux criteria in power operation of the plant. In such cases, the reactor trip would be triggered by other diversified triggering paths, however longer reaction times are to be expected.

For the safety-related assessment of the applicability of these investigation results for German nuclear power plants authorized for power operation, GRS performed further investigations after the publication of this report. The results of these investigations are documented in /GRS 21/. This was the reason for the amendment of this report.

## Kurzfassung

Der vorliegende Bericht umfasst Ergebnisse, die im Rahmen des vom BMUB geförderten Forschungsvorhabens „Auswirkungsbereiche von Softwarefehlern in sicherheitstechnisch wichtigen Einrichtungen“ (Förderkennzeichen 3614R01304), erarbeitetet wurden. Das übergeordnete Ziel dieses Vorhabens war es, die Wissensbasis der GRS über Softwarefehler in softwarebasierter Leittechnik und deren Auswirkungen zu erweitern.

Dazu wurde zunächst im Rahmen der Ermittlung und Aufbereitung des relevanten Standes von Wissenschaft und Technik nach Definitionen von für das Vorhaben relevanten Begriffen in Normen, Standards und Veröffentlichungen (DIN-, IEEE-Normen, IAEA-Standards, NUREG-Veröffentlichungen) sowie in den Sicherheitsanforderungen für Kernkraftwerke recherchiert. Dabei stellte sich insbesondere heraus, dass der Begriff „Softwarefehler“ in den betrachteten Literaturquellen unterschiedlich und teilweise widersprüchlich definiert wird. Aus diesem Grund wurde im Rahmen dieses Vorhabens unter Betrachtung des Software-Lebenszyklus softwarebasierter Leittechniksysteme eine Definition von Softwarefehlern entwickelt, welche die verschiedenen relevanten Aspekte zu Softwarefehlern und deren Auswirkungen berücksichtigt. Es ergibt sich, dass Softwarefehler latente Fehler in einem softwarebasierten Leittechniksystem sind, die zu einem nichtanforderungsgerechten Verhalten eines softwarebasierten Leittechniksystems führen können. Bei den Softwarefehlern ist grundsätzlich zwischen Programmierfehlern und Spezifikationsfehlern zu unterscheiden.

In einem weiteren Arbeitsschritt wurden Recherchen zur Betriebserfahrung mit Softwarefehlern in softwarebasierter Leittechnik im nuklearen und im nichtnuklearen Bereich durchgeführt. Diese Ereignisse wurden hinsichtlich ihrer Ursache und Auswirkungen analysiert und die Ergebnisse zusammenfassend beschrieben. Aufbauend auf der entwickelten Definition von Softwarefehlern und aus dem COMPSIS-Klassifizierungsschema für Ereignisse in softwarebasierter Leittechnik wurde im Rahmen dieses Vorhabens zur Klassifizierung von Ereignissen mit Softwarefehlern aus der Betriebserfahrung das COCS-Klassifizierungsschema entwickelt, bei dem die Ereignisse nach den Kriterien „Ursache“, „betroffenes System“, „Auswirkung“ sowie „GVA-Potenzial“ klassifiziert werden. Dieses Klassifizierungsschema wurde bei der Auswertung der im Rahmen des Vorhabens ermittelten Ereignisse angewandt. Daraus ergibt sich, dass die meisten Softwarefehler auf Programmierfehler und/oder Spezifikationsfehler zurückzuführen sind. Softwarefehler können auch aus dem Zusammenwirken des

Leittechniksystems mit ihrer Betriebsumgebung resultieren, z. B. aufgrund von Hardwarefehlern. Eine weitere Softwarefehlerart betrifft Fehler, die sich aufgrund fehlerhafter System-Anforderungsspezifikation ergeben. Derartige Fehler sind unabhängig von der Realisierung des Leittechniksystems (analog oder softwarebasiert).

In einem weiteren Arbeitsschritt wurden potentielle Auswirkungen von einzelnen aus der Betriebserfahrung identifizierten Softwarefehlerarten auf spezielle Leittechnikfunktionen und auf die damit verknüpfte Verfahrenstechnik untersucht. Dazu wurden verschiedene Konzepte erarbeitet. Den Untersuchungskonzepten ist gemeinsam, dass zunächst jeweils einzelne Softwarefehlerarten in einer betrachteten softwarebasierten Leittechnikfunktion postuliert und anschließend deren Auswirkungen analysiert werden. Die Untersuchungskonzepte unterscheiden sich in deren Umsetzung im Wesentlichen in der erzielbaren Detaillierungstiefe bei der Modellierung der Softwarefehler, der betrachteten Leittechnikfunktionen bzw. Leittechniksysteme und deren Verknüpfung mit der angesteuerten Verfahrenstechnik. Am Beispiel eines softwarebasierten Neutronenflussmesssystems in einem Druckwasserreaktor wurde gezeigt, dass unterstellte Fehler in der Software zur Berechnung der Neutronenflusssignale zur Blockade von einzelnen Auslösepfaden für die Reaktorschnellabschaltung (RESA) im Leistungsbetrieb der Anlage führen können. Die RESA-Auslösung wäre in solchen Fällen durch andere diversitäre RESA-Auslösepfade sichergestellt, bei denen jedoch längere Reaktionszeiten zu erwarten sind.

Zur sicherheitstechnischen Bewertung der Übertragbarkeit dieser Untersuchungsergebnisse für deutsche Anlagen mit Berechtigung zum Leistungsbetrieb hat die GRS nach der Veröffentlichung dieses Berichtes weitere Untersuchungen angestellt. Die Ergebnisse dieser Untersuchungen sind in /GRS 21/ dokumentiert. Dies war der Anlass für die Ergänzung des vorliegenden Berichts.

## Inhaltsverzeichnis

# 1 Einleitung, Aufgabenstellung und Zielsetzung

In den letzten Jahren haben softwarebasierte leittechnische Einrichtungen eine wachsende Bedeutung in deutschen Kernkraftwerken gewonnen. Dies beruht zum einen auf der sich zunehmend verschlechternden Möglichkeiten zur Beschaffung von Ersatzteilen bei den eingesetzten konventionellen leittechnischen Einrichtungen und zum anderen auf der Prozessoptimierung, die durch den Einsatz softwarebasierter leittechnischer Einrichtungen realisiert werden kann.

Der Umfang, in dem softwarebasierte leittechnische Komponenten eingesetzt werden, erstreckt sich je nach Anlage von einzelnen Betriebsmitteln bis zu komplett in softwarebasierter Leittechnik ausgeführten Systemen. Dies betrifft sowohl betriebliche Systeme und Komponenten (z. B. Turbinenregelung, Reaktorregelung, einzelne Baugruppen von konventionellen Leittechniksystemen) als auch Einrichtungen, Komponenten und Systeme mit sicherheitstechnischer Bedeutung sowie Einrichtungen mit sonstiger sicherheitstechnischer Bedeutung (z. B. Begrenzungssysteme, elektrische Schutz- und Steuerungseinrichtungen, Steuerung von Notstromdieseln, Überstromschütze, Hebezeug- und Brennelementhandhabungseinrichtungen, Messumformer, Messwertgeber, Neutronenflussmesseinrichtungen).

Die bisherige Betriebserfahrung mit softwarebasierten leittechnischen Einrichtungen und Geräten in deutschen Kernkraftwerken hat gezeigt, dass softwarebasierte bzw. programmierbare Einrichtungen und Geräte zusätzliches Fehlerpotential im Vergleich zu Geräten und Einrichtungen der konventionellen Leittechnik aufweisen. Dazu zählen Softwarefehler. Softwarefehler umfassen u. a. Fehler in der Firmware und Programmierfehler in Algorithmen und können sich auf die Integrität und Funktion von sicherheitstechnischen wichtigen Einrichtungen eines Kernkraftwerkes sowie von Einrichtungen mit sonstiger sicherheitstechnischer Bedeutung eines Kernkraftwerkes auswirken. Ein weiteres Fehlerpotential ist durch Schadsoftware gegeben. So hat z. B. die Schadsoftware "stuxnet" /SPI 11/ gezeigt, dass softwarebasierte leittechnische Einrichtungen derart manipuliert werden können, dass schädigende Auswirkungen auf die Anlage eintreten können.

Die Ausführungen im vorangegangenen Abschnitt verdeutlichen, dass Softwarefehler in softwarebasierten leittechnischen Einrichtungen und Geräten – aufgrund ihrer Auswirkungen auf die Einrichtungen und Systeme – die Sicherheit eines Kernkraftwerkes beeinträchtigen können. Für eine fundierte sicherheitstechnische Bewertung von softwarebasierten Leittechniksystemen ist deshalb eine umfassende Kenntnis über Auswirkungen von Softwarefehlern auf die Einrichtungen und Systeme eines Kernkraftwerkes erforderlich.

Die GRS hat im Rahmen des Vorhabens "Vertiefte Untersuchungen von nach AtSMV /BAN 92/, /BAN 10/ meldepflichtigen Ereignissen und sonstigen Betriebserfahrungen aus Kernreaktoren des In- und Auslandes" und des Vorhabens "Vertiefte Untersuchungen von Betriebserfahrungen aus Kernreaktoren, Teil A: Technisch-wissenschaftliche Arbeiten und Teil B: Fachberatung" bereits Erkenntnisse über Auswirkungen von Softwarefehlern in softwarebasierten und programmierbaren Geräten auf Einrichtungen von kerntechnischen Anlagen in Deutschland und im Ausland gewinnen können. Die GRS hat als interdisziplinäre Sachverständigenorganisation in Fragen der Reaktorsicherheit ein Eigeninteresse daran, ihr Wissen über Auswirkungen von Softwarefehlern auf Einrichtungen insbesondere auf sicherheitstechnisch wichtige Einrichtungen in Kernkraftwerken zu erweitern, weil dies eine wesentliche Grundlage für eine fundierte und sachgerechte sicherheitstechnische Bewertung von softwarebasierten Leittechniksystemen darstellt. Zudem können gewonnene Erkenntnisse aus Untersuchungen zu den Auswirkungen von Softwarefehlern auf sicherheitstechnisch wichtige Einrichtungen in Kernkraftwerken bei der Bewertung von meldepflichtigen Ereignissen in Kernkraftwerken sowie im Rahmen aktueller und zukünftiger Diskussionen in einschlägigen Arbeitsgruppen und Fachausschüssen verwendet werden.

Das übergeordnete Ziel dieses Vorhabens ist es deshalb, die Wissensbasis der GRS über Auswirkungen von Softwarefehlern in softwarebasierten Leittechniksystemen zu erweitern. Dies kann durch eine umfassende und systematische Untersuchung möglicher Auswirkungen von Softwarefehlern in softwarebasierten und programmierbaren Geräten auf Einrichtungen und Systeme von Kernkraftwerken erzielt werden. Dazu sind möglichst unterschiedliche Systeme und verschiedene Softwarefehler in der Untersuchung zu betrachten. Eine derartige Untersuchung ist der GRS bisher nicht bekannt.

Hierfür ist es einerseits notwendig, die Betriebserfahrung mit softwarebasierten leittechnischen Systemen und Einrichtungen auf nationaler und internationaler Ebene tiefergehend auszuwerten, um weitere als die bisher betrachteten Ereignisse mit Softwarefehlern in softwarebasierten leittechnischen Einrichtungen und Systemen zu identifizieren und hinsichtlich Fehlerart, Fehlerursache und der betroffenen softwarebasierten und programmierbaren Geräte zu analysieren, und dabei insbesondere die sicherheitstechnische Bedeutung der verfahrenstechnischen Auswirkungen auf die Anlage zu betrachten. Andererseits sind Auswirkungen von postulierten Softwarefehlern auf sicherheitstechnisch wichtige Einrichtungen von Kernkraftwerken generisch zu untersuchen.

In diesem Vorhaben wurden dazu

- der relevante Stand von Wissenschaft und Technik ermittelt und aufbereitet (Arbeitspaket 1),
- Ereignisse mit Softwarefehlern in softwarebasierten leittechnischen Einrichtungen identifiziert und analysiert (Arbeitspaket 2),
- potentielle Auswirkungen von postulierten Softwarefehlern in sicherheitstechnisch wichtigen Einrichtungen generisch untersucht (Arbeitspaket 3),
- die gewonnenen Erkenntnisse aus den Untersuchungen in dem vorliegenden Abschlussbericht aufbereitet und dokumentiert (Arbeitspaket 4).

Die vier Arbeitspakete im Rahmen dieses Vorhabens sind nachfolgend beschrieben.

## 1.1 Arbeitspaket 1: Ermittlung und Aufbereitung des relevanten Standes von Wissenschaft und Technik

In diesem Arbeitspaket wird der für die Bearbeitung des Vorhabens relevante Stand von Wissenschaft und Technik ermittelt und aufbereitet. Dabei werden insbesondere folgende Aspekte betrachtet:

- Begriffsbestimmungen: Software, Softwarefehler, sicherheitstechnisch relevantes Softwareversagen, sicherheitstechnisch wichtige Einrichtung
- Bisherige Erkenntnisse zum Einsatz softwarebasierter Leittechnik in Kernkraftwerken: Definition softwarebasierte Leittechnik.

Dazu wurden u. a. folgende Informationsquellen herangezogen:

- Bisherige Arbeiten der GRS zum Einsatz von softwarebasierter Leittechnik in Kernkraftwerken: Vorhaben "Entwicklung und Einsatz von Analysemethoden zur Beurteilung softwarebasierter leittechnischer Einrichtungen in deutschen Kernkraftwerken".
- Bisherige Arbeiten der GRS zu der Betriebserfahrung auf nationaler und internationaler Ebene in kerntechnischen Anlagen: Vorhaben "Vertiefte Untersuchungen von Betriebserfahrungen aus Kernreaktoren" Teil A: Technisch-wissenschaftliche Arbeiten und Teil B: Fachberatung“ und OECD/NEA/COMPSIS-Projekt.
- Einschlägige Fachliteratur und weitere Informationsquellen wie z. B. NUREG (Veröffentlichungen der USNRC), relevante DIN-Normen, Herstellerunterlagen, Berichte von Gutachterorganisationen und Forschungsinstituten.

Der auf diese Weise aufbereitete Stand von Wissenschaft und Technik ist im Kapitel 2 dieses Abschlussberichtes dokumentiert. Dieser Bericht soll es Lesern mit fachlicher Grundkompetenz ermöglichen, sich rasch einen Überblick über den Stand von Wissenschaft und Technik und über aktuelle Entwicklungen zu der Thematik „Auswirkungen von Softwarefehlern in softwarebasierter Leittechnik in Kernkraftwerken“ zu verschaffen.

## 1.2 Arbeitspaket 2: Identifikation von Ereignissen mit sicherheitsrelevanten Softwarefehlern in softwarebasierten Leittechniksystemen

In diesem Teil des Vorhabens werden sicherheitstechnisch relevante Softwarefehler in softwarebasierten Leittechniksystemen und Geräten identifiziert. Dazu werden nationale und internationale Ereignisse mit Softwarefehlern in Leittechniksystemen sowohl im kerntechnischen als außerhalb des kerntechnischen Bereichs betrachtet.

Hierfür werden u. a. meldepflichtige Ereignisse, Veröffentlichungen von Sachverständigenorganisationen und Forschungsinstituten sowie Arbeitsberichte im Hinblick auf sicherheitstechnisch relevante Softwarefehler vertieft ausgewertet. Die Methodik zur vertieften Auswertung beruht auf dem Durchsuchen von Datenbanken (GRS-Datenbank ICDE-GVA-Datenbank, COMPSIS-Datenbank, IAEA/IRS-Datenbank) nach Ereignissen mit Softwarefehlern. Weiterhin werden Ereignisse unterhalb der Meldeschwelle, sofern Informationen verfügbar sind, berücksichtigt.

Hier werden auch, sofern Informationen verfügbar sind, Auswirkungen von Malware in softwarebasierten Einrichtungen und Systemen in Kernkraftwerken ermittelt.

Die identifizierten Ereignisse mit Softwarefehlern werden anschließend hinsichtlich Fehlerart, Fehlerursache, betroffener Einrichtungen und in Bezug auf ihre tatsächliche und ihre potentielle sicherheitstechnische Bedeutung klassifiziert und analysiert. Die Analyse der sicherheitsrelevanten verfahrenstechnischen Auswirkungen der Ereignisse und die Klassifizierung der identifizierten Ereignisse mit Softwarefehlern orientieren sich an den folgenden Leitfragen:

- Welche Softwarefehler wurden bisher beobachtet?
- Was ist die Ursache des beobachteten Softwarefehlers? (z B. Fehlerhafte Spezifikation, fehlerhafte Programmierung, Kombination aus fehlerhafter Spezifikation und fehlerhafter Programmierung).
- Worin liegt die tatsächliche und potentielle sicherheitstechnische Relevanz eines aufgetretenen Ereignisses mit Softwarefehler?
- Welche sicherheitstechnisch wichtigen Einrichtungen des Kernkraftwerkes waren betroffen bzw. wären betroffen gewesen?

Ziel ist es, die Kenntnisse der GRS über Auswirkungen von Softwarefehlern in der Leittechnik in Einrichtungen von Kernkraftwerken insbesondere in sicherheitstechnisch wichtigen Einrichtungen zu erweitern und systematisch aufzubereiten. Dieses Wissen kann als Grundlage für die Analyse und Bewertung von softwarebasierten Leittechniksystemen herangezogen werden.

Die Ergebnisse der Arbeiten im Arbeitspaket 2 sind in Kapitel 3, Kapitel 4 und Kapitel 5 zusammenfassend dargestellt.

## 1.3 Arbeitspaket 3: Untersuchung der Auswirkungen von postulierten Softwarefehlern in sicherheitstechnisch wichtigen Einrichtungen von Kernkraftwerken

Im Rahmen dieses Arbeitspaketes werden potentielle Auswirkungen von Softwarefehlern in softwarebasierter Leittechnik auf die Funktion sicherheitstechnisch wichtiger Einrichtungen untersucht.

Auf der Grundlage der Analyse der ermittelten Ereignisse mit Softwarefehlern aus dem Arbeitspaket 2 werden Softwarefehler in sicherheitstechnisch wichtigen leittechnischen Einrichtungen in einer generischen deutschen Anlage postuliert und deren potentielle verfahrenstechnische Auswirkungen untersucht. Für die Untersuchungen sind die postulierten Softwarefehler sowie die zu betrachtenden sicherheitstechnisch wichtigen Einrichtungen festzulegen.

Die Ergebnisse des Arbeitspaketes 3 sind in Kapitel 6 dargestellt.

### 1.4 Arbeitspaket 4: Dokumentation der Ergebnisse

Die Ergebnisse des Vorhabens wurden fortlaufend dokumentiert, aufbereitet und in dem vorliegenden Abschlussbericht nach den folgenden Punkten, die sich an den erzielten Ergebnissen in den einzelnen Arbeitspaketen orientieren, gegliedert zusammenfassend dargestellt. In diesem Abschlussbericht wurden folgende Aspekte betrachtet:

- Ermittlung und Aufbereitung des relevanten Standes von Wissenschaft und Technik auf nationaler und internationaler Ebene (Kapitel 2).
- Identifizierte Ereignisse mit Softwarefehlern in softwarebasierter Leittechnik außerhalb der Kerntechnik (Kapitel 3).
- Identifizierte Ereignisse mit Softwarefehlern in softwarebasierter Leittechnik in kerntechnischen Anlagen (Kapitel 4).
- Auswertung und Klassifizierung der identifizierten Ereignisse im nichtnuklearen und im nuklearen Bereich (Kapitel 5).
- Untersuchungen zu den Auswirkungen von Softwarefehlern auf sicherheitstechnisch wichtige Einrichtungen in Kernkraftwerken (Kapitel 6).

Zudem wurden die im Rahmen dieses Vorhabens erarbeiteten Ergebnisse bzw. Zwischenergebnisse bei der International Conference on Nuclear Engineering (ICONE) im Juni 2016 in Charlotte, USA präsentiert und dem interessierten Fachpublikum zugänglich gemacht /MBO 16/.

# 2 Ermittlung und Aufarbeitung des relevanten Standes von Wissenschaft und Technik

## 2.1 Begriffsbestimmungen

Im Folgenden werden die für dieses Vorhaben relevanten Begriffe definiert. Dabei werden insbesondere die in verschiedenen Regelwerken, Normen und Standards unterschiedlich verwendeten Formulierungen dieser Begriffe dargestellt. Es ist hierbei anzumerken, dass die Begriffe gemäß dem Wortlaut in den betrachteten Regelwerken, Normen und Standards wiedergegeben werden.

### 2.1.1 Software

Der Begriff Software wird nach DIN 62138 /DIN 09/, DIN 60880 /DIN 60/, DIN 61513 /DIN 02/ und KTA 1401 /KTA 14/ definiert als Programme (d. h. Sätze geordneter Instruktionen), Daten, Regeln einschließlich zugehöriger Dokumentation, die zum Betrieb eines rechnerbasierten leittechnischen Systems gehören.

IEEE 24765 /IEE 10/ definiert Software u. a. als

- Programme oder Programmteile, Verfahrensweisen, Regeln einschließlich der zugehörigen Dokumentationen eines informationsverarbeitenden Systems.
- Programme, Verfahrensweisen und eventuell dazugehörige Dokumentation sowie Daten, die zum Betrieb eines Rechnersystems benötigt werden.

1. Software kann weiter unterteilt werden in Systemsoftware und Anwendungssoftware:

- Systemsoftware wird nach DIN EN 62138 /DIN 09/, DIN EN 61513 /DIN 02/ und DIN EN 60880 /DIN 60/ für ein bestimmtes Rechnersystem oder eine Rechnerfamilie mit dem Zweck erstellt, Entwicklung, Betrieb und Modifikation des Systems und der zugehörigen Programme zu erleichtern.

  Üblicherweise wird Systemsoftware weiter kategorisiert in Betriebssoftware und unterstützende Software (oder Softwarewerkzeuge).

Während Betriebssoftware in sicherheitsklassifizierten leittechnischen Systemen implementiert ist, ist unterstützende Software in nicht sicherheitsklassifizierten Hilfssystemen implementiert oder wird offline genutzt.

DIN 60880 /DIN 60/ definiert Betriebssoftware als Software, die auf dem Zielprozessor während dessen Betrieb läuft. Hierzu zählen beispielsweise Eingangs- und Ausgangstreiber und –dienste, Interrupt-Management, Scheduler, Kommunikationstreiber, anwendungsorientierte Bibliotheken und Online-Diagnostik.

Laut IEEE 24765 /IEE 10/ wird Systemsoftware erstellt, um den Betrieb und die Wartung eines Rechnersystems und der zugehörigen Programme zu erleichtern.

- Anwendungssoftware ist nach DIN 60880 /DIN 60/ und DIN 61513 /DIN 02/ der Teil der Software, durch die Anwendungsfunktionen realisiert werden.

  Nach IEEE 24765 /IEE 10/ wird Anwendungssoftware entwickelt, um spezielle Wünsche des Nutzers zu erfüllen. Daher ist diese Software ein speziell für die Lösung eines Anwendungsproblems geschriebenes Programm. Sie ermöglicht es dem Benutzer, eine spezielle Aufgabe zu erfüllen oder bestimmte Probleme zu handhaben und unterscheidet sich dadurch von Systemsoftware.

- DIN 60880 /DIN 60/ definiert darüber hinaus Multiversionssoftware als Satz unterschiedlicher Programme, die entwickelt wurden, um einer gemeinsamen Anforderung und einer gemeinsamen Abnahmeprüfung zu genügen. Diese Versionen laufen gleichzeitig und voneinander unabhängig ab, üblicherweise in redundanter Hardware. Identische Eingangssignale von Prüfsystemen oder sich entsprechende Eingangssignale in redundanten Systemen werden verwendet. Zur Entscheidung bei unterschiedlichen oder widersprüchlichen Ausgangssignalen der verschiedenen Versionen werden vorbestimmte Strategien wie z. B. Mehrheitsentscheidungen eingesetzt.

- Weiterhin definiert DIN 60880 /DIN 60/ vorgefertigte Software als Software, die bereits vorgefertigt als kommerzielles oder gesetzlich geschütztes Produkt verfügbar und für den Einsatz in einem rechnerbasierten System vorgesehen ist.

Die einzelnen Auslegungseinheiten einer Software werden als Softwarekomponenten bezeichnet und sind entweder der Systemsoftware oder der Anwendungssoftware zugehörig /DIN 09/.

Der Begriff Firmware bezeichnet die in ein Gerät fest eingebaute Software, die nicht frei programmierbar ist und definierte gerätespezifische Funktionen erbringt. Wird die Firmware modifiziert, handelt es sich um ein modifiziertes Gerät /KTA 14b/.

**Sicherheitskritische Software**

Als sicherheitskritische Software (engl. safety-critical software) wird laut /IEE 10/ jede Software bezeichnet, die unter eine oder mehrere der folgenden Kategorien fällt:

1. Software, die durch unangemessene Reaktionen auf Ereignisse, Ausfall bei Anforderung, Reaktionen in falscher Reihenfolge oder Folgereaktionen einen Unfall verursachen kann.

2. Software, die mit dem Ziel entwickelt wurde, die Auswirkungen eines Unfalls so gering wie möglich zu halten (Schadensbegrenzung).

3. Software, die mit dem Ziel entwickelt wurde, den Ausgangszustand nach einem Unfall wiederherzustellen.

Sicherheitskritische Software kann demzufolge über unterschiedliche Aspekte definiert werden: zum einen über die möglichen Auswirkungen, die durch bestimmte Softwarefehler verursacht werden können und die von großen finanziellen Verlusten bis hin zum Verlust von Menschenleben führen können, und zum anderen über die Aufgaben der Systeme, in denen sie eingesetzt werden. Zudem werden mögliche Ausfallarten genauer spezifiziert.

### 2.1.2 Softwarefehler und Softwareversagen

Der weitgefasste Begriff des Softwarefehlers wird in den verschiedenen Standards durch die nicht ausschließlich auf Software bezogenen Begriffe wie Fehler (engl. fault), Abweichung (engl. error), menschliches Fehlverhalten (Irrtum) (engl. mistake), Versagen (engl. failure), schwerwiegender Fehler (engl. fatal error) und Mangel (engl. defect) definiert. Je nach Standard werden jedoch unterschiedliche Terminologien verwendet, die im Folgenden aufgeführt sind.

In DIN 60880 /DIN 60/ und DIN 61513 /DIN 02/ wird der Begriff Fehler allgemein definiert als Mangel an einer Hardware-, Software- oder Systemkomponente.

Laut DIN 61513 /DIN 02/ ist ein Softwarefehler (engl. software fault) ein Auslegungsfehler in einer Softwarekomponente. Häufiger wird jedoch der Oberbegriff Fehler (fault) verwendet, der nach DIN 62138 /DIN 09/ als ein Mangel an einer Hardware-, Software- oder Systemkomponente definiert ist. Der Standard merkt an, dass im Unterschied zu Hardwarefehlern, die auch Zufallsfehler sein können, Softwarefehler systematische Fehler sind, die auf Auslegungsmängel zurückgehen und bei gleichen Bedingungen systematisch zu denselben Ausfällen führen. In diesem Zusammenhang wird in /DIN 09/ nicht erläutert, was unter „gleichen Bedingungen“ und „denselben Ausfällen“ zu verstehen ist. Insbesondere Auslegungsfehler, aber auch andere Fehler, können in einem System unentdeckt bleiben. Erst wenn besondere Bedingungen auf dieses System einwirken, so dass die resultierende Funktion nicht mehr mit der vorgesehenen Funktion übereinstimmt, führen solche Fehler zum Versagen.

Als Folge eines Fehlers kann es zu Versagen (engl. failure) kommen, das als Abweichung der ausgeführten von der vorgesehenen Funktion definiert ist. Als Abweichung wird die Diskrepanz zwischen berechneten, beobachteten oder gemessenen Werten oder Bedingungen und den tatsächlichen, spezifizierten oder theoretischen Werten oder Bedingungen definiert. /DIN 09/

Der Standard IEEE 24765 /IEE 10/ definiert einen Fehler (fault) vor allem hinsichtlich der Art des Ereignisses:

1. das Vorhandensein einer Abweichung (error) in einer Software,
2. ein unzulässiger Schritt, unzulässiger Prozess oder eine unzulässige Definition der Daten in einem Computerprogramm,
3. ein Defekt in einem Hardware-Gerät oder einer Hardware-Komponente. Als Synonym wird hier auch der Begriff „Bug“ verwendet.

Der Begriff der Abweichung (error) beschreibt nach IEEE 24765 /IEE 10/ detaillierter die (möglichen) Ursachen, die zu einem Fehler (fault) führen und schließt auch das menschliche Fehlverhalten als (mögliche) Ursache mit ein.

Demnach ist eine Abweichung:

1. eine menschliche Handlung, die ein unzulässiges Ergebnis zur Folge hat, wie zum Beispiel fehlerhafte Software als Folge eines Fehlers bei der Erstellung der Software.

2. ein unzulässiger Schritt, unzulässiger Prozess oder eine unzulässige Definition der Daten.

3. ein unzulässiges Resultat.

Als Auswirkung eines Fehlers wird das Versagen (engl. failure) definiert als

1. Verlust der Fähigkeit eines Produktes, eine benötigte Funktion auszuführen, oder Unfähigkeit, die Funktion innerhalb der vorgegebenen spezifizierten Randbedingungen auszuführen.

2. ein Ereignis, bei dem ein System oder eine Systemkomponente eine benötigte Funktion nicht innerhalb der spezifizierten Randbedingungen ausführen kann.

Noch umfassender als das Versagen ist der schwerwiegende Fehler (fatal error), der laut IEEE 24765 /IEE 10/ eine Abweichung darstellt, die zu einem vollständigen Ausfall des Systems oder der Komponenten führt. Neben den Begriffen des Fehlers (Ursache) und des Versagens (Auswirkung) wird auch der allgemeinere Begriff des „Mangels“ (engl. defect) verwendet. Laut IEEE 24765 /IEE 10/ ist er definiert als

1. ein Problem, welches ohne Korrektur zum Ausfall einer Anwendung oder zu fehlerhaften Resultaten führen kann.

2. eine Unzulässigkeit oder ein Defizit in einer Komponente, so dass Anforderungen und Spezifikationen nicht erfüllt werden können und die Komponente repariert oder ersetzt werden muss.

Laut technischem Bericht Nr. 384 der IAEA /IAE 99/ ist ein Softwarefehler (engl. fault) das Ergebnis einer Abweichung im Quellcode oder in den Daten der Spezifikationen, die zu einem unvorschriftsmäßigen Verhalten der Software führt, wenn der fehlerhafte Abschnitt der Software oder die fehlerhaften Daten durchlaufen werden.

Zudem wird angemerkt, dass Fehler nicht notwendigerweise zum Systemausfall führen, wenn die Umgebungsbedingungen des Systems den Fehler nicht zur Wirkung bringen.

Gemäß /IAE 07/ ist ein Software-Versagen (engl. failure) die Unfähigkeit einer Struktur, eines Systems oder einer Komponente, innerhalb eines vorgegebenen Akzeptanzkriteriums zu funktionieren. Das Software-Versagen ist das Ergebnis eines Signalverarbeitungsprozesses unter Verwendung eines fehlerhaften Quellcodes oder unzulässiger Daten. Der Ausfall kann z. B. dann auftreten, wenn das Ausgangsdatensignal des Systems unzulässig für den erforderlichen Signalverarbeitungsprozess ist. Dabei kann ein Systemausfall das Ergebnis einer falsch konzipierten Hardware, eines Hardwarefehlers oder eines Softwarefehlers sein, der nur dann zur Wirkung gebracht wird, wenn die entsprechende Hardware oder Software für den Signalverarbeitungsprozess erforderlich ist /IAE 99/.

Laut SiAnf /BMUB 12/ ist ein Softwarefehler ein Fehler in einer Software, der bei bestimmten Kombinationen oder einer bestimmten Abfolge von Eingangsdaten nicht spezifizierte Ausgangsdaten erzeugt.

Die Nuclear Regulatory Commission (NRC) definiert den Begriff des Fehlers (engl. fault) als abweichendes Verhalten eines Rechnersystems von seinen zugewiesenen Spezifikationen /LAW 93/. Weiter wird unterschieden zwischen Hardwarefehler, der aufgrund einer physischen Veränderung in der Hardware das Verhalten des Computersystems auf eine unerwünschte Weise ändert, und Softwarefehler, der durch einen menschlichen Irrtum (mistake) im Quellcode entsteht. Nach /LAW 93/ wird synonym für einen Softwarefehler auch der Begriff „bug“ verwendet, der in anderen Standards ausschließlich für Hardwarefehler verwendet wird /IEE 10/.

Eine Abweichung (engl. error) ist ein unzulässiger Zustand der Hardware, Software oder der Daten, der aus einem Fehler resultiert. Abweichungen, die durch menschlichen Irrtum in die Software eingebracht werden, bleiben zunächst nur verborgene Abweichungen (engl. latent error) und haben erst dann Auswirkungen, wenn sie durch entsprechende Daten oder ähnliches zur Wirkung gebracht werden. Dann können sie zum Versagen führen. Eine effektive Abweichung kann sich von einer Komponente zur anderen fortpflanzen und so weitere Abweichungen verursachen. /LAW 93/

Aus den vorangegangenen Ausführungen geht hervor, dass der Begriff Softwarefehler in der Literatur unterschiedlich und teils widersprüchlich definiert wird. Dies lässt aus Sicht der GRS an bestimmten Stellen einen breiten Interpretationsspielraum bei den Definitionen. Aufgrund dessen können diese unterschiedlichen Definitionen für eine einheitliche Auswertung von Ereignissen mit Softwarefehlern nicht herangezogen werden. Um

Ereignisse mit Softwarefehlern einheitlich auswerten und klassifizieren zu können, wurde deshalb im Rahmen dieses Vorhabens eine Beschreibung von Softwarefehlern erarbeitet, welche die relevanten Aspekte von Softwarefehlern berücksichtigt. Diese Beschreibung ist im Kapitel 5.1 angegeben. Sie wurde als Grundlage für die Auswertung der identifizierten Ereignisse mit Softwarefehlern im Rahmen dieses Vorhabens angewendet.

### 2.1.3 Systemfehler und Systemversagen

Die NRC klassifiziert Fehler nach ihrer übergeordneten Fehlerart, d. h. der Beschaffenheit eines Fehlers und nach der detaillierten Fehlerursache. Eine solche Klassifikation kann vor allem bei der Wahl der richtigen Methode zur Fehleranalyse und der geeigneten Modellierung der Fehler im Rahmen der Fehleranalyse hilfreich sein. Auf diese Weise erleichtert eine Klassifikation der Fehler die Problemlösung. Die Systemfehler (Fehlermodi) werden in Bezug auf ihre Fortdauer im Rechnersystem in folgende Klassen unterteilt /LAW 93/:

- **Designfehler**
  Ein Designfehler oder Spezifikationsfehler wird beschrieben als ein Fehler, der auf eine fehlerhafte Spezifikation zurückzuführen ist. Die meisten Softwarefehler fallen in diese Kategorie. Designfehler können durch Änderung/Modifikation behoben werden. Ein einzelner Designfehler kann zu vielen Abweichungen und Versagensfällen führen, bevor er entdeckt und verbessert wird. /LAW 93/.

- **Betrieblich bedingter Fehler**
  Ein betrieblich bedingter Fehler wird beschrieben als ein Fehler im Betrieb eines Rechnersystems, der zum Teilausfall des Rechnersystems führt und eine Instandsetzung des betroffenen Rechnersystems vor dessen Wiederinbetriebnahme erfordert. Beispiele sind elektronische und mechanische Fehler, Datenbank-Fehler und Bedienungsfehler /LAW 93/.

- **Temporärer Fehler**
  Ein temporärer Fehler (auch als „Soft Error“ bezeichnet) ist ein Fehler, der zwar zum Versagen des Rechnersystems führt, der sich aber durch einen Neustart des Systems beheben lässt. Häufig lässt sich die zugrunde liegende Ursache eines solchen Fehlers nicht mehr genau bestimmen.

Beispiele für vorübergehende Fehler sind Rauschen in der Spannungsversorgung, Störungen eines Speichers durch kosmische Strahlung oder Timing-Fehler im Betriebssystem /LAW 93/.

Die NRC /LAW 93/ unterteilt das Systemversagen in drei Kategorien: Die Auswirkungen des Versagens auf das Computersystem, den Umfang der Auswirkungen und die sicherheitstechnische Bedeutung des Versagens.

**1. Versagensmodus**

- **Plötzliches Versagen**
  Das plötzliche Versagen ist eine Versagensart, die trotz vorangegangenen Kontrollen nicht vorherzusehen war und vollkommen unerwartet auftritt.

- **Allmähliches Versagen**
  Beim allmählichen Versagen geht das System zunächst in einen gestörten Betriebszustand über, bevor das Versagen eintritt. Daher ist das Versagen bei vorangegangenen Kontrollen bereits vorauszusehen.

- **Teilweises Versagen**
  Beim teilweisen Versagen werden Abweichungen vom spezifizierten Verhalten beobachtet, die jedoch nicht den kompletten Verlust benötigter Funktionen zur Folge haben.

- **Vollständiges Versagen**
  Das vollständige Versagen zeichnet sich dadurch aus, dass Abweichungen vom Verhalten beobachtet werden, die außerhalb der spezifizierten Grenzen liegen, so dass sie einen kompletten Verlust benötigter Funktionen zur Folge haben.

- **Katastrophales Versagen**
  Ein katastrophales Versagen ist definiert als plötzliches und vollständiges Versagen.

- **Versagen nach Verschlechterung**
  Ein Versagen nach Verschlechterung ist definiert als allmähliches und teilweises Versagen.

**2. Umfang des Versagens**

- Ein Versagen wird als “intern” (engl. internal) bezeichnet, wenn das fehlerhafte Gerät oder der fehlerhafte Prozess mit dem Fehler in angemessener Weise umgehen kann.
- Ein Versagen ist „begrenzt“ (engl. limited), wenn es nicht mehr als „intern“ bezeichnet werden kann, aber seine Auswirkungen auf das Gerät oder den Prozess beschränkt bleiben.
- Ein Versagen wird als „beherrschend“ (engl. pervasive) bezeichnet, wenn es sich auch auf andere Geräte und Prozesse auswirkt.

**3. Sicherheitstechnische Bedeutung des Versagens**

- Ein System ist intrinsisch sicher (engl. „intrinsically safe“), wenn das System keine gefährlichen Zustände besitzt.
- Ein System wird als sicher gegenüber Folgeschaden (engl. „fail safe“) bezeichnet, wenn das System so ausgelegt ist, dass es in einem gefährlichen Zustand einen Unfall vermeiden kann.
- Ein System wird als System zur Beherrschung von Unfällen bezeichnet, wenn es die Auswirkungen eines ausgelösten Unfalls mindern kann.
- Ein System kann als Warnsystem eingesetzt werden, um einen Operateur zu warnen, wenn ein Versagen, zu einem gefährlichen Zustand führen kann.

### 2.1.4 Fehlerursachen

Fehlerursachen werden in /DIN 09/ durch die Begriffe Abweichung (error) und menschliches Fehlverhalten (Irrtum) (mistake) konkretisiert. Menschliches Fehlverhalten (Irrtum) ist die Handlung oder Vorgehensweise eines Menschen, die eine unbeabsichtigte Folge hat.

Je nachdem, ob es sich um Hardware oder Software handelt, treten unterschiedliche Systemfehler auf. Die Fehlerursachen der oben klassifizierten Systemfehler werden vom NRC /LAW 93/ wie folgt genauer spezifiziert:

- Hardwarefehler können Designfehler, betrieblich bedingte Fehler oder auch temporäre Fehler sein. Häufig treten Hardwarefehler als temporäre Fehler auf.

- Softwarefehler sind theoretisch immer Designfehler, die nach /LAW 93/ in die folgenden acht Kategorien eingeteilt werden können:
  - Logische Fehler
  - Interface-Fehler
  - Datenbank-Fehler
  - Fehler in der Datenbeschreibung/Spezifikation
  - Eingabe/Ausgabe-Fehler
  - Rechenfehler
  - Datenverarbeitungsfehler
  - Sonstige Fehler
- Eingangsdaten-Fehler: Ein Fehler in den Eingangsdaten kann auf einen Designfehler zurückzuführen sein, wenn beispielsweise ein Sensor mit dem falschen Gerät verbunden wird. Es kann sich aber auch um einen betriebsbedingten Fehler handeln, wenn ein Nutzer die falschen Daten bereitstellt.
- Operateur/Benutzer-Fehler: Wenn die Betriebsanweisungen, die einem Operateur/Benutzer zur Verfügung stehen, falsche Anweisungen enthalten, so handelt es sich um einen Designfehler, der auch als Verfahrensfehler (engl. procedure fault) bezeichnet wird. Ein betrieblich bedingter Fehler kann z .B. als Folge eines Bedienfehlers des Operateurs vorliegen, wenn der Operateur den Anweisungen zuwider handelt. Ein temporärer Fehler kann auftreten, wenn der Operateur, den Anweisungen folgend, einen versehentlichen Fehler (beispielsweise Eingabefehler) macht.

Es werden noch weitere Fehlerursachen aufgezählt, wie z. B. fehlerhafte Daten auf permanenten/temporären Speichermedien, Fehler aufgrund von Umwelteinflüssen (z. B. unterbrochene Stromversorgung) sowie Fehler unbekannter Ursache.

Aus den vorangegangenen Ausführungen wird deutlich, dass unterschiedliche Fehlerursachen sowie Kombinationen dieser Fehlerursachen zum Versagen eines Rechnersystems führen können.

### 2.1.5 Malware

Der Begriff Malware steht für „malicious software“ und bezeichnet Software, die entwickelt wurde, um auf Computersystemen Schadfunktionen auszuführen oder das System auf andere Weise zu manipulieren /TEC 15/. Typische Beispiele für Malware sind Viren, Würmer, Trojanische Pferde und Spyware. Synonym werden auch die Begriffe Schadsoftware oder Evilware verwendet. Häufig nutzen Schadprogramme sogenannte Zero-Day-Lücken, d. h. Sicherheitslücken, die dem Hersteller vorher nicht bekannt waren und die erst durch die Computer-Attacke aufgedeckt werden /KUP 10/.

#### 2.1.5.1 Computer-Wurm

Ein Computerwurm ist ein schädliches Softwareprogramm, das sich von einem einzelnen Computersystem ausgehend von Computer zu Computer weiterverbreitet. Typische Verbreitungswege für Würmer sind Dateien, die als E-Mail-Anhang gesendet werden, über Links zu einer Web- oder FTP-Ressource, in einer textbasierten Chatnachricht von sogenannten Instant-Messaging Diensten (z. B. ICQ oder IRC) oder über Filesharing-Netzwerke verteilt werden. Aber auch Wechseldatenträger können mit einem Computerwurm infiziert sein. Damit der Wurm sich weiterverbreitet, muss entweder der Anwender die infizierte Datei oder den Link öffnen oder das Computersystem führt den Wurmcode automatisch aus. Durch die Verbreitung über das Netzwerk kann sich ein Wurm rasant weiterverbreiten. Im Juli 2001 verbreitete sich z. B. der Computerwurm „Code Red“ innerhalb von etwa 9 Stunden 250000-mal. Computerwürmer verbrauchen aufgrund ihrer Weiterverbreitung Netzwerkbandbreite und haben zusätzlich schadhafte Inhalte /NEW 09/, /KAS 15/.

##### 2.1.5.1.1 Der Computerwurm „Flame“

Der Computerwurm „Flame“ wurde im Mai 2012 vom Softwareunternehmen Kaspersky Labs entdeckt und kann vor allem zur Cyberspionage eingesetzt werden. Der Wurm verfährt nach der sogenannten „Man-in-the Middle“ Methode, bei der das Schadprogramm auf den Computer geschleust wird, indem sich der Angreifer unbemerkt zwischen zwei Netzwerkteilnehmern einschaltet und so die Kontrolle über den Datenverkehr erlangt.

Mit „Flame“ infizierte Computer können ferngesteuert und umfangreich ausspioniert werden. So können nicht nur Dateien und E-Mails kopiert werden, sondern auch angeschlossene Mikrofone und Kameras zur Überwachung eingesetzt sowie Tastatureingaben und Netzwerkverkehr aufgezeichnet werden /SCH 12/.

Durch den Computerwurm Flame wurde ein Microsoft Windows Update für Desktop Minianwendungen (Gadgets) so manipuliert, dass Anwender, die sich mit dem Microsoft Windows Update verbanden, auf einen infizierten Computer umgeleitet wurden und ein manipuliertes Update-Paket erhielten /BHA 12/. Den Flame-Entwicklern gelang es offenbar bei der Programmierung des Wurms, ein gültiges Microsoft Zertifikat auszustellen.

Laut Kaspersky Labs wurden ebenfalls Computer hauptsächlich im Mittleren Osten und aus dem Iran infiziert, wobei die Hacker nach ihren Studien offenbar ein besonderes Interesse an AutoCAD-Plänen, pdf- und Office-Dateien hatten. Sofort nach der Entdeckung des Wurms wurde die bereits seit einigen Jahren unbemerkt aufgestellte Netzwerkinfrastruktur zur Verbreiterung des Wurms und Steuerung der mit dem Wurm infizierten Rechner offline geschaltet /GOS 12/.

#### 2.1.5.2 Virus

Ein Computervirus ist ein Schadprogramm, das andere Computer infiziert, indem es sich unbemerkt in ausführbare Dateien kopiert. Viren benötigen immer eine Programmdatei, Makros oder andere ausführbare Inhalte, um ihren Maschinencode auszuführen. Je nach Infektionsart lassen sich Viren kategorisieren in Dateiviren, Bootsektorviren, Makroviren und Skriptviren. Wenn der Anwender die infizierte Programmdatei (Host-Programm) ausführt, wird unbemerkt auch der Virus ausgeführt, der dann die noch nicht betroffenen Dateien infizieren kann. Die Verbreitung von Viren auf andere Computersysteme erfolgt nur dann, wenn der Anwender eine betroffene Datei beispielsweise über Wechselmedien oder Rechnernetzwerke (E-Mail, Web-Server) auf ein anderes System kopiert. Daher sind Computerviren bei ihrer Ausbreitung auf die aktive Interaktion des Anwenders angewiesen und verbreiten sich nicht so schnell und effektiv wie Computerwürmer /NEW 09/, /KAS 15/.

#### 2.1.5.3 Trojanisches Pferd

Als Trojanisches Pferd (engl. Trojan Horse, kurz auch Trojaner) werden Schadprogramme bezeichnet, die sich als „reguläre“ Programme tarnen, aber im Computersystem im Hintergrund unbemerkt Anwendungen ausführen. Trojanische Pferde können sich beispielsweise hinter Computerspielen, Antivirus-Programmen oder unseriöser Software verstecken, die von Anwendern selber auf dem Computer installiert wird /TEC 15/.

Trojaner können beispielsweise Daten löschen, sperren, modifizieren, kopieren oder die Funktionalität von Computern oder Computernetzwerken beeinträchtigen.

Entsprechend ihrer Aktivität können Trojaner weiter kategorisiert werden. Trojaner, die beispielsweise mit dem Ziel programmiert wurden, Daten des Computers auszuspionieren, verwenden sogenannte Keylogger, die die Eingabe des Benutzers an der Computertastatur aufzeichnen und so Informationen über das System, Kennwörter etc. erhalten. Noch weiter geht der sogenannte Backdoor-Trojaner, der es dem Hacker ermöglicht, unentdeckt ferngesteuerten Zugriff auf den Computer zu nehmen und alle Sicherheitsmaßnahmen des Computers, wie beispielsweise Passwörter, Verschlüsselung etc., zu umgehen (engl. remote access trojan, RAT). Der Hacker kann dann jede beliebige Aktion wie senden, empfangen, starten oder löschen von Dateien ausführen. Die bekanntesten Backdoor-Hilfsprogramme sind Netcat und Rootkit, die dazu dienen die Entdeckung der Schadsoftware zu verhindern /NEW 09/.

Im Gegensatz zu Viren und Würmern können sich Trojanische Pferde nicht selbstständig vervielfältigen.

##### 2.1.5.3.1 Das trojanische Pferd „Duqu“

Der unter dem Namen Duqu (nach dem Präfix „~DQ“ der erzeugten Dateien) bekannt gewordene Nachfolger des Stuxnet Wurms gehört zur Kategorie der Trojanischen Pferde.

Duqu hat nicht zum Ziel industrielle Steuerungssysteme zu manipulieren, sondern ist vielmehr ein Spionage-Programm, das Daten von Herstellern industrieller Steuerungssysteme sammelt, die zukünftig gezielt für Cyber-Angriffe eingesetzt werden könnten /SYM 11/.

Die Hacker haben es dabei auf Informationen über Konstruktionsunterlagen (beispielsweise AutoCAD Pläne etc.) abgesehen, deren Kenntnis bei der gezielten Sabotage industrieller Steuerungsanlagen als hilfreich eingeschätzt werden. Laut /SYM 11/ beschränkt sich die Gefährdung auf eine kleinere Anzahl ausgewählter Organisationen, die spezielle Produkte herstellen.

Der Trojaner installierte auf den betroffenen Computern Schadsoftware („Infostealer“), die unter Verwendung Keylogger die Eingabe des Benutzers an der Computertastatur aufzeichnen, um Informationen über das System, Kennwörter etc. zu erhalten. Alle Informationen werden in einer verschlüsselten und komprimierten Datei aufgezeichnet und herausgeschleust.

Das Schadprogramm wurde so konfiguriert, dass es 36 Tage auf dem Computer läuft, bevor es sich selbstständig deinstalliert /SYM 11/.

## 2.2 Bisherige Erkenntnisse zum Einsatz von Software in Leittechniksystemen

### 2.2.1 Leittechnik

Der Begriff Leittechnik bezeichnet die Gesamtheit aller leittechnischen Einrichtungen zum Ausführen von Leittechnik-Funktionen /KTA 14b/. Als leittechnische Einrichtungen werden laut dem Kerntechnischen Ausschuss (KTA) Geräte und Systeme definiert, die zum

- Messen,
- Steuern,
- Regeln,
- Überwachen,
- Aufzeichnen und
- Schützen

eines Prozesses oder einer Einrichtung verwendet werden.

Dabei umfassen die leittechnischen Einrichtungen verschiedene Geräte vom Messwertgeber bis zu den Einzelantrieben zugeordneten Teilen der Steuerung zur Auslösung von Schutzaktionen. Leittechnische Einrichtungen können sowohl automatische Einrichtungen als auch Einrichtungen zur Prozessführung durch einen Operateur sein /KTA 14b/.

### 2.2.2 Generischer Aufbau eines Leittechniksystems

Eine generische Darstellung eines Leittechniksystems lässt sich mit drei Funktionsebenen definieren, die bei der Ausführung der Leittechnik-Funktionen von Bedeutung sind /GRS 15/:

- Eingabeebene
- Verarbeitungsebene
- Ausgabeebene

**Eingabeebene**

Die Eingabeebene des generischen Leittechniksystems bildet alle Komponenten für die Generierung von Eingabedaten aus dem zu steuernden Prozess für das Leittechniksystem ab. Dazu zählen Geräte zur Erfassung von Messgrößen und zur Eingabe von Daten:

- Sensor/Detektor (Messgrößenaufnehmer): Zur Erfassung der Anlagenparameter (physikalisch messbare Größen, z. B. Druck, Temperatur) eingesetztes Gerät, das an seinem Eingang die Messgröße erfasst und an seinem Ausgang ein entsprechendes Messsignal abgibt /DIN 14/.
- Messumformer: Geräte zur Umformung der Messsignale, die mittels Sensoren/Detektoren aus den physikalisch messbaren Größen, z. B. Druck, Temperatur (Anlagenparameter), generiert wurden. Messumformer und Sensoren/Detektoren können in einem einzigen Gerät untergebracht sein bzw. können eine Funktionseinheit bilden.
- Einheitsmessumformer: Messumformer, dessen Ausgangsgröße ein standardisiertes Signal ist /DIN 14/.
- Geber: Stellungsgeber zur Erfassung der Stellung von Komponenten der Anlage.

- Bedienelemente/Eingabeeinheiten: Diese Vorrichtungen ermöglichen eine manuelle Bedienung oder Konfiguration einer Komponente oder eines Systems durch manuelle Eingabe von Signalen durch den Operateur. Hierbei unterscheidet man zwischen analoger und digitaler Eingabe von Signalen ins Leittechniksystem. Typische Bedienelemente sind Taster, Schalter, Potentiometer, Anzeigegeräte (Touchscreen), etc.

**Verarbeitungsebene**

Auf der Verarbeitungsebene werden die Daten aller Komponenten verarbeitet, die vom Leittechniksystem aus der Eingabeebene zur Verfügung gestellt werden. Die Verarbeitung der Daten erfolgt nach einer entsprechend der auszuführenden leittechnischen Funktion festgelegten Verarbeitungsvorschrift. Dabei kann es sich beispielsweise um die Grenzwertbildung, logische Operationen, den Signalvergleich oder die Berechnung der Ausgangssignale für die Ausgabeebene handeln. Auf der Verarbeitungsebene werden die folgenden Typen von Baugruppen und Komponenten eingesetzt /GRS 15/:

- Nicht programmierbare Baugruppen ohne/mit Softwarebestandteilen: Hierzu zählen Baugruppen, die ohne programmierbare Bauelemente aufgebaut sind. Sie können jedoch auch Bauelemente enthalten, bei denen vom Hersteller eine softwarebasierte Konfiguration vorgenommen wurde, die vom Kunden nicht mehr veränderbar ist. Beispiele sind einfache ASICs (application-specific integrated circuit) ohne integrierten Mikroprozessor. /GRS 15/

- Programmierbare Baugruppen: Bei programmierbaren Baugruppen ist mindestens ein programmierbares Bauelement enthalten. Die softwarebasierte Konfiguration der Baugruppen kann auch nach dem Herstellungsprozess vom Kunden noch verändert werden. Die Ausführungen dieser Baugruppen können unterschiedliche Komplexität besitzen, wie z. B. CPLDs (complex programmable logic devices) oder FPGAs (field programmable gate arrays). Der Begriff „Programm“ bezieht sich dabei vor allem auf die Definition der Funktionsstruktur der Baugruppe. Die Vorgabe zeitlicher Abläufe, wie sie bei der Programmierung von Computern erfolgt, ist bei diesen Bausteinen zweitrangig /GRS 15/.

- Rechnerbasierte Baugruppe: Im Unterschied zu programmierbaren Baugruppen besitzen rechnerbasierte Baugruppen einen oder mehrere Prozessoren. So lassen sich sowohl Konfiguration als auch Funktion der Baugruppe durch die Ausführung von Software in einem Betriebssystem realisieren. Auch hier gibt es Baugruppen

sehr unterschiedlicher Komplexität von einfachen Mikrocontrollern über Mikroprozessoren bis hin zu Multi-Core-Prozessoren. ASICs und FPGAs, welche Mikroprozessoren enthalten, werden ebenfalls dieser Baugruppenart zugeordnet /GRS 15/.

- Rechnersystem: Mit dem Leittechniksystem dauerhaft oder zeitweise verbundene, eigenständige Rechneranlagen, die für die Ausführung von Leittechnik-Funktionen relevant sind, wie beispielsweise Steuerstabfahrrechner /GRS 15/.

**Ausgabeebene**

Der Ausgabeebene des generischen Leittechniksystems werden alle Komponenten zugeordnet, die für die Umformung der elektrischen Signale aus der Verarbeitungsebene und für die Ansteuerung der verfahrenstechnischen Komponenten (Aktuatoren), wie beispielsweise Ventile, Schieber, Pumpen, Motoren und Schalter, zur Beeinflussung der Anlagenparameter genutzt werden. Baugruppen für die Prioritätssteuerung der elektrischen Signale aus der Verarbeitungsebene werden hier ebenso betrachtet. Die Komponenten der Ausgabeebene können funktionell in Steuerungseinrichtungen und in Anzeige- und Meldeeinrichtungen unterteilt werden. Sie bilden die Schnittstelle zwischen dem Leittechniksystem und dem zu steuernden Prozess bzw. dem Bedienpersonal.

- Koppelebene ohne/mit Softwarebestandteilen: Geräte/Komponenten, welche die Ausgangssignale aus der Verarbeitungsebene umformen, um die Ansteuerung der Aktuatoren zu ermöglichen. Sie werden der Koppelebene zugeordnet. Auch diese Geräte können mit Softwarebestandteilen realisiert werden, wie beispielsweise Relais mit softwarebasierter Ansteuerung der Relaisspule /GRS 15/.

- Anzeigen (einschließlich Meldeeinrichtungen) ohne/mit Softwarebestandteile(n): Anzeigen werden zur Signalisierung von Informationen (z. B. Messwerten, Stellung der Aktuatoren) verwendet und können ohne (z. B. Zeigerelemente, Siebensegmentanzeigen, Schreiber etc.) oder mit Softwarebestandteilen (z. B. Displays, Monitore, Melderechner etc.) umgesetzt werden /GRS 15/. Diese Anzeigen gehören zur Schnittstelle zwischen dem Prozess und den Operateuren.

Auch weitere technische Aspekte wie Stromversorgung, Lüftung, Schutzeinrichtungen, Kommunikation und Zugriffsmöglichkeiten, die für die Ausführung der leittechnischen Funktionen unverzichtbar sind, zählen zum Leittechniksystem. Sie können mit Softwarebestandteilen realisiert werden /GRS 15/.

### 2.2.3 Softwarebasierte Leittechnik

Der generische Aufbau eines Leittechniksystems verdeutlicht bereits, dass der Begriff „softwarebasierte Leittechnik“ nicht eindeutig ist, da unter softwarebasierter Technik sowohl im herkömmlichen Sinne programmierbare Technologien mit einem Prozessor als auch nutzerprogrammierbare Hardware wie FPGAs verstanden werden können. Da die Programmierung auch nach dem Herstellungsprozess noch verändert werden kann, zählt die GRS auch nutzerprogrammierbare Hardware wie FPGAs zu den softwarebasierten Komponenten /GRS 15/. Dementsprechend wird im Rahmen dieses Vorhabens die Definition eines softwarebasierten Leittechniksystems nach DIN 60880 angewendet: Ein softwarebasiertes Leittechniksystem ist ein leittechnisches System, dessen Funktionen meistens von Mikroprozessoren, programmierten elektronischen Einheiten oder Rechnern abhängen oder durch die Verwendung derartiger Gerätschaften zur Gänze durchgeführt werden.

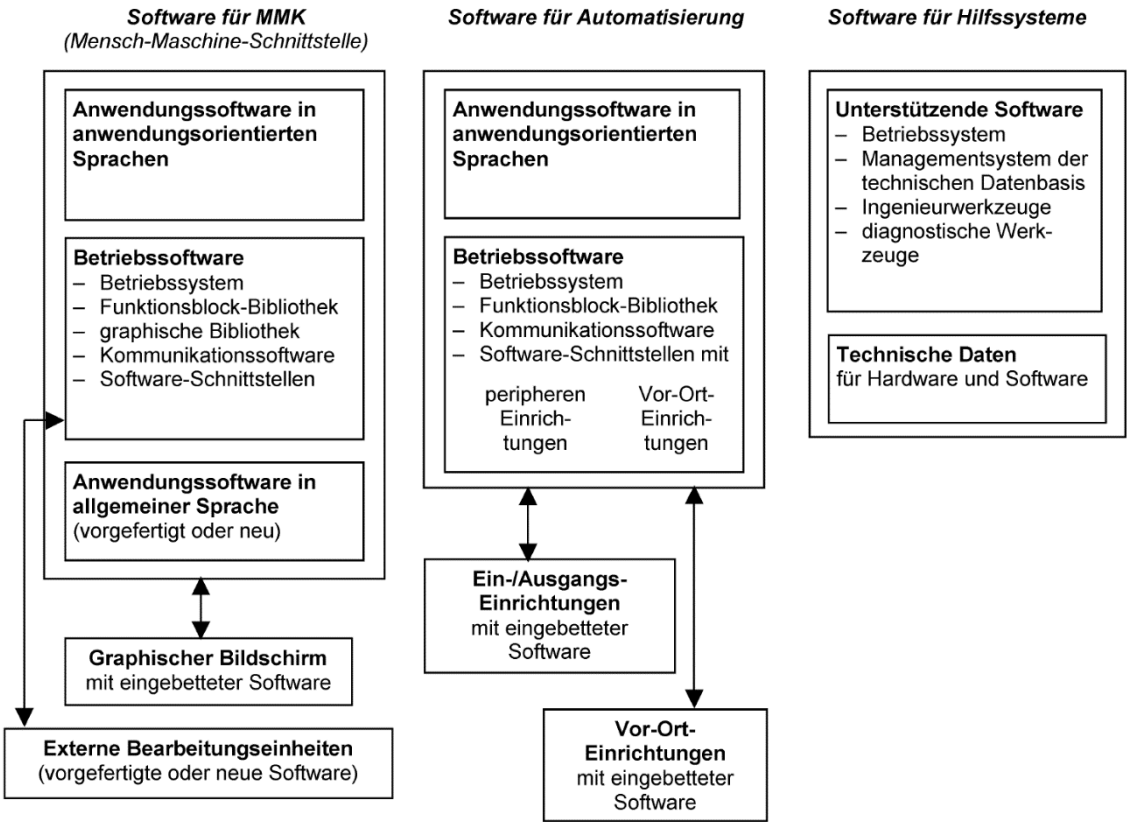



**Abb. 2.1** Typische Softwarebestandteile in rechnerbasierten leittechnischen Systemen /DIN 09/

Typische in einem softwarebasierten leittechnischen System oder einer softwarebasierten leittechnischen Architektur verwendetenn Softwareteile /DIN 09/.

Es ist zu erkennen, dass Betriebssoftware und Anwendungssoftware in softwarebasierten leittechnischen Systemen in der Mensch-Maschine-Schnittstelle eingesetzt werden und für die Automatisierung verwendet werden. Unterstützende Software (Softwarewerkzeuge) wird in Hilfssystemen eingesetzt.

Software kann auch in Ein-/Ausgangs-Einrichtungen (z. B. Betätigungsbausteinen, Anzeige wie z. B. Graphischer Bildschirm), Vor-Ort-Einrichtungen (z. B. Messfühler) sowie in speziellen Einrichtungen wie Kommunikationsbausteinen und Einrichtungen zur unterbrechungsfreien Stromversorgung verwendet werden bzw. dort integriert sein („eingebetteter Software“).

Die Software eines leittechnischen Systems kann auch in vorgefertigte Software und neue Software unterteilt werden.

Systemsoftware, zu der Betriebssoftware und unterstützende Software gehören, ist üblicherweise vorgefertigte Software, Anwendungssoftware ist in der Regel neu. /DIN 09/

Gerätefamilien softwarebasierter leittechnischer Systeme sind mit anwendungsorientierten Entwicklungswerkzeugen ausgestattet, die die Spezifizierung der auszuführenden leittechnischen Funktionen auf der Basis der Anforderungen mit Hilfe graphischer Techniken (Funktionspläne) ermöglichen. Mit den Entwicklungswerkzeugen werden die graphischen Programme automatisch in ablauffähige Anwendungssoftware umgesetzt.

Softwarebasierte Leittechniksysteme weisen i. d. R. typische/gemeinsame Merkmale auf. Einige davon sind nachfolgend aufgelistet:

- Modularer Aufbau softwarebasierter Leittechniksysteme (Eingabemodule, Verarbeitungsmodule, Ausgabemodule…)
- Softwarebasierte Leittechniksysteme bestehen i. d. R. aus verteilten Rechnersystemen/Modulen, die über Kommunikationsnetzwerke gekoppelt/verbunden sind.
- Vorhandensein von Kommunikationsnetzwerken für Datenaustausch zwischen den Modulen/Komponenten/Rechnern mit Kommunikationssoftware (Netzwerkprotokolle) und Kommunikationshardware
- Vorhandensein von Schnittstellen zwischen den einzelnen Modulen des Leittechniksystems und zu anderen Rechnersystemen (z.B. Steuerstabfahrrechner, Leistungsverteilungsrechner) und Kommunikationsnetzwerken

- Vorhandensein von Schnittstellen für Instandhaltung des Leittechniksystems, Konfigurierung und Parametrierung der System- und Anwendungsfunktionen (z.B. Servicerechner, Servicegerät)

In Tab. 2.1 wird eine Übersicht über die in verschiedenen Kernkraftwerken eingesetzten softwarebasierten Leittechniksysteme gegeben. Im Wesentlichen kommen in diesen Anlagen die softwarebasierten Leittechniksysteme Teleperm XS, Ovation und Spinline 3 zum Einsatz. In einigen wenigen Anlagen werden die softwarebasierten Leittechniksysteme Himax, Meltac und Melody/Symphony verwendet.

**Tab. 2.1** Übersicht über eingesetzte softwarebasierte Leittechniksysteme in Kernkraftwerken /GRS 15/

| **Leittechniksystem** | **Leittechnische Funktion** | **Kernkraftwerk** |
|---|---|---|
| HIMAX | Leittechnische Einrichtungen wie z. B. Reaktorregelungen, Mess- und Überwachungseinrichtungen, betrieblicher Schutz, Prozessrechner | Gundremmingen B und C |
| MELTAC | Leittechnische Einrichtungen einschließlich Reaktorschutz /MIT14/ | Tomari 3 (Japan) |
| | | Ikata 1 und 2 (Japan) |
| | | Mihama 3 (Japan) |
| | | Takahama 1 bis 4 (Japan) |
| | | Ohi 1 bis 4 (Japan) |
| OVATION | Leittechnische Einrichtungen einschließlich Reaktorschutz | Temelín 1 und 2 (Tschechische Republik) |
| | Leittechnische Einrichtungen wie z. B. Reaktorregelungen, Mess- und Überwachungseinrichtungen, betrieblicher Schutz, Prozessrechner /WES04/ | South Texas 1 und 2 (USA) |
| | | Point Beach 1 und 2 (USA) |
| | | Surry 1 und 2 (USA) |
| | | Braidwood 1 und 2 (USA) |
| | | Byron 1 und 2 (USA) |
| | | Vandellos 2 (Spanien) |
| | | Almaraz 1 und 2 (Spanien) |
| | | Ascó 1 und 2 (Spanien) |
| | | Leningrad (Russische Föderation) |

| Leittechnik-system | Leittechnische Funktion | Kernkraftwerk |
|---|---|---|
| | | Ringhals 2 (Schweden) |
| | | Kozloduy 5 und 6 (Bulgarien) |
| | | Shin-Kori 1 und 2 (Südkorea) |
| | | Shin Wolsong 1 und 2 (Südkorea) |
| SPINLINE 3 | Leittechnische Einrichtungen einschließlich Reaktorschutz /ROL12/ | Belleville 1 und 2 (Frankreich) |
| | | Cattenom 1 bis 4 (Frankreich) |
| | | Flamanville 1 und 2 (Frankreich) |
| | | Golfech 1 und 2 (Frankreich) |
| | | Nogent 1 und 2 (Frankreich) |
| | | Paluel 1 bis 4 (Frankreich) |
| | | Penly 1 und 2 (Frankreich) |
| | | St. Alban 1 und 2 (Frankreich) |
| | | Fessenheim 1 und 2 (Frankreich) |
| | | Bugey 2 bis 5 (Frankreich) |
| | | Dukovany 3 (Tschechische Republik) |
| | | Kozloduy (Bulgarien) |
| | | Qinshan Phase II 1 bis 4 (Volksrepublik China) |
| | | Tihange (Belgien) |
| | | Ignalina (Litauen) |
| | | Isar 1 |

| Leittechnik-system | Leittechnische Funktion | Kernkraftwerk |
|---|---|---|
| SYMPHONY MELODY | Leittechnische Einrichtungen wie z. B. Reaktorregelungen, Mess- und Überwachungseinrichtungen, betrieblicher Schutz, Prozessrechner | Gundremmingen B und C |
| TELEPERM XS | Leittechnische Einrichtungen einschließlich Reaktorschutz | Beznau 1 und 2 (Schweiz) |
| | | Ringhals 1 (Schweden) |
| | | |
| | | Qinshan 1 (China) |
| | | Tianwan 1 und 2 (China) |
| | | Paks 1 bis 4 (Ungarn) |
| | | Bohunice (Slowakische Republik) |
| | | Kozloduy 5 und 6 (Bulgarien) |
| | Leittechnische Einrichtungen einschließlich Reaktorbegrenzungen | Gösgen (Schweiz) |
| | | Unterweser |
| | | Philippsburg 2 |
| | | Neckarwestheim 1 |
| | | Biblis B |
| | Leittechnische Einrichtungen wie z. B. Reaktorregelungen, Mess- und Überwachungseinrichtungen, betrieblicher Schutz, Prozessrechner | Emsland |
| | | Biblis A |
| | | Grafenrheinfeld |
| | | Neckarwestheim 2 |
| | | Brunsbüttel |
| | | Grohnde |
| | | Brokdorf |
| | | Krümmel |
| | Leittechnische Einrichtungen einschließlich USUS | Philippsburg 1 |

### 2.2.4 Nationale und internationale Regelwerke, Normen und Standards für leittechnische Systeme

Die für dieses Vorhaben relevanten nationalen und internationalen Standards und Normen sind in Tab. 2.2 entsprechend ihrer jeweiligen Thematik aufgeführt. Links sind die Standards der IAEA, in der Mitte die deutschen Normen und rechts die US-amerikanischen Standards dargestellt. Eine ausführliche Behandlung der verschiedenen nationalen und internationalen Regelwerke, Normen und Standards, die sich mit den Anforderungen an Software von Leittechniksystemen in sicherheitskritischen Anwendungen auseinander setzen, erfolgt in /GRS 16/.

**Tab. 2.2** Tabellarischer Überblick über nationale und internationale Standards und Normen für leittechnische Systeme in der Kerntechnik

<table>
<tr><td>Klassifizierung von Systemen mit sicherheitstechnischen Funktionen<br>NS-G-1.3</td><td colspan="2">Kategorisierung sicherheitsrelevanter Systeme<br>DIN IEC 61226</td><td>Kategorisierung sicherheitsrelevanter Systeme<br>DIN IEC 61226</td><td></td></tr>
<tr><td></td><td colspan="2">Allgemeine Systemanforderungen und Klassifizierung leittechnischer Systeme<br>DIN IEC 61513</td><td>Kriterien für digitale Rechner in Sicherheitssystemen für Kernkraftwerke<br>IEEE 7-4.3.2-1993</td><td></td></tr>
<tr><td>Dokumentation der Sicherheit und Zuverlässigkeit von Software und Hardware<br>NS-G.1.1</td><td>Software Qualifizierungsanforderungen Kat. A<br>DIN IEC 60880</td><td>Qualifizierungsanforderungen von elektrischen Geräten<br>DIN IEC 60780</td><td>Qualifizierung digitaler Systeme<br>NUREG 800, Appendix 7.0-A<br>NRC, Regulatory Guide 1.152<br>Softwareentwicklung<br>NUREG 6734</td><td rowspan="2">Qualifizierung von Geräten der Klasse 1E<br>IEEE 323-2003</td></tr>
<tr><td>Entwicklung und Design rechnerbasierter Leittechniksysteme<br>NS-G-1.3</td><td>Software Qualifizierungsanforderungen Kat. B und Kat. C<br>DIN EN 62138</td><td>Qualifizierungsanforderungen von Hardware rechnerbasierter Systeme<br>DIN IEC 60987</td></tr>
<tr><td>Schutz gegen GVA in der Leittechnik<br>NP-T-1.5</td><td colspan="2">Schutz gegen Fehler gemeinsamer Ursache (GVA) in der Leittechnik<br>DIN IEC 62340</td><td>Diversitäre Software und Schutz gegen GVA<br>NUREG 7007 (basierend auf NUREG 6303)</td><td></td></tr>
</table>

### 2.2.5 Überblick über die Klassifizierung/Kategorisierung leittechnischer Funktionen

In der Kerntechnik wird die konventionelle Leittechnik zunehmend durch den Einsatz von softwarebasierter Leittechnik ersetzt. Dabei führen leittechnische Systeme Funktionen unterschiedlicher sicherheitstechnischer Bedeutung aus. Entscheidend für die Bewertung der sicherheitstechnischen Funktionen sind dabei die Folgen ihres Versagens oder ihr spontanes fälschliches Ansprechen. Zur Einstufung leittechnischer Funktionen nach ihren Sicherheitsanforderungen gibt es je nach Regelwerk unterschiedliche Kategorien. Ein Überblick über die Kategorien bzw. Klassen der nationalen und internationalen Standards ist in Tab. 2.3 zu sehen.

Die deutschen Regelwerke und Normen /BMUB 12/, /DIN 10/ und /KTA 14b/ stufen die leittechnischen Funktionen in drei Kategorien A, B und C ein. Die RSK-Leitlinien /RSK 96/ verwenden ebenfalls drei Kategorien, die jedoch mit 1, 2 und 3 bezeichnet werden.

**Tab. 2.3** Überblick über verschiedene Kategorisierungen/Klassifikationen der Leittechnik in sicherheitstechnisch wichtigen Systemen

| **Nationaler oder internationaler Standard** | **Kategorisierungs-/Klassifizierungsschema der sicherheitsrelevanten Systeme** | | |
|---|---|---|---|
| **IAEA SSR-2/1** | Sicherheitstechnisch wichtige Systeme | | |
| | Sicherheitssysteme | Sicherheitsrelevantes System | |
| **DIN 61226** | Sicherheitstechnisch wichtige Systeme | | |
| | Kat. A | Kat. B | Kat. C |
| **KTA** | Kat. A | Kat. B | Kat. C |
| **RSK-Leitlinien** | Kat. 1 | Kat. 2 | Kat. 3 |
| **Sicherheitsanforderungen Modul 5** | Kat. A | Kat. B | Kat. C |
| **USA und IEEE** | Sicherheitstechnisch wichtige Systeme | | |
| | Sicherheitsrelevant oder Klasse 1E | kein eigener Name zugewiesen | |

Gemäß den Sicherheitsanforderungen an Kernkraftwerke /BMUB 12/ gelten folgende abgestuften Kategorien:

- Kategorie A: Die Leittechnikfunktionen der Kategorie A umfassen alle Funktionen, die erforderlich sind, um Ereignisse der Sicherheitsebene 3 (Kategorie des Anlagenzustands, die einem Störfall entspricht) zu beherrschen.
- Kategorie B: Die Leittechnikfunktionen der Kategorie B umfassen alle Funktionen, die erforderlich sind, um Ereignisse der Sicherheitsebene 2 (anomaler Betrieb) zu beherrschen sowie das Eintreten von Ereignissen der Sicherheitsebene 3 zu vermeiden.
- Kategorie C: Die Leittechnikfunktionen der Kategorie C umfassen alle übrigen Funktionen mit sicherheitstechnischer Bedeutung.

Die Erfüllung der jeweiligen Anforderungen wird durch Einrichtungen realisiert, bei denen sowohl Hardware also auch Software Leittechnikfunktionen mit sicherheitstechnischer Bedeutung ausführen.

Der Standard IEEE 323 /IEE 03b/ definiert die höchste Sicherheitsklasse als Klasse 1E für elektrische Geräte und Systeme, wobei synonym auch der Begriff der „sicherheitsrelevanten elektrischen Geräte“ gebraucht wird. In diese Klasse fallen alle elektrischen Geräte und Systeme, die wesentlich bei der Reaktorschnellabschaltung, dem Gebäudeabschluss, der Kühlung des Reaktorkerns, der Reaktorsicherheitsbehälter- und Reaktor-Wärmeabfuhr sind oder die in anderer Weise wichtig sind, um die Freisetzung radioaktiven Materials in die Umgebung zu verhindern.

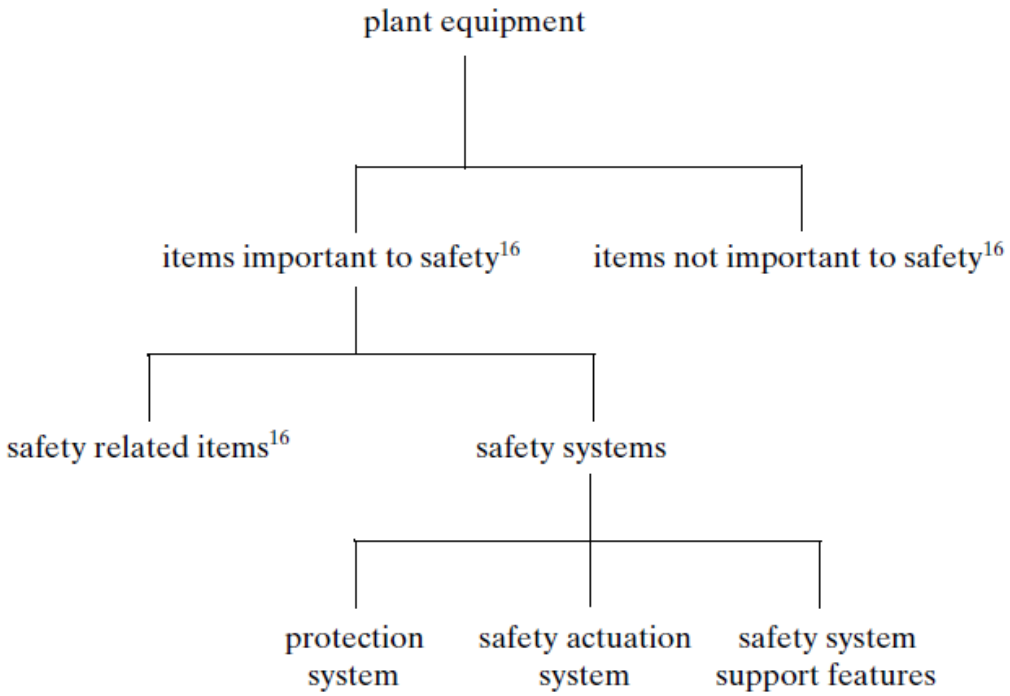
Die Klassifizierung leittechnischer Systeme und Komponenten nach dem Standard der IAEA ist in Abb. 2.2 dargestellt. Ein sicherheitstechnisch wichtiges System wird dabei zunächst als eine Struktur, ein System oder eine Komponente definiert, die bei einer Störung oder einem Versagen zur Strahlenexposition des Personals oder der Umgebung führen kann. Diese Strukturen, Systeme oder Komponenten verhindern, dass anzunehmende betriebliche Vorkommnisse zu Störfallbedingungen führen und begrenzen die Auswirkungen von Störfällen.

Darüber hinaus wird zwischen „Sicherheitssystem“ (safety system) und „sicherheitsrelevantem System“ (safety related system) unterschieden:

- Sicherheitssystem: Sicherheitstechnisch wichtiges System, das dafür vorgesehen ist das sichere Abfahren des Reaktors sicherzustellen, die Nachwärme des Kerns abzuführen oder die Auswirkungen von anzunehmenden betrieblichen Vorkommnissen und Auslegungsstörfällen zu begrenzen. Das Sicherheitssystem besteht aus dem Reaktorschutzsystem (protection system), dem Sicherheitsauslösesystem (safety actuation system) und den Hilfseinrichtungen des Sicherheitssystems (safety system support features).
- Sicherheitsrelevantes System: System, das sicherheitstechnisch wichtig ist, aber nicht zum Sicherheitssystem gehört.

Die entsprechenden Auslegungsanforderungen für das „Sicherheitssystem“ bzw. das „sicherheitsrelevante System“ befinden sich in /IAE 80/ und /IAE 84/.

Die Klassifizierung eines Leittechniksystems beschreibt die sicherheitstechnische Bedeutung, die bei der Ausführung sicherheitstechnischer Funktionen erforderlich ist. Dabei ist zu beachten, dass die von der IAEA und von DIN verwendeten Definitionen und Konzepte nicht identisch sind, da die IAEA eine Systemklassifizierung verwendet, während die DIN eine FSE-Kategorisierung gebraucht, wobei FSE für Funktionen und zugehörige Systeme und Geräte steht /DIN 02/. Laut /DIN 10/ sind Anforderungen an „Sicherheitssysteme“ im Allgemeinen mit den Anforderungen in Kategorie A oder B konsistent, während „sicherheitsrelevante Systeme“ im Allgemeinen den Kategorien B oder C zugeordnet werden.




[16] In this context, an 'item' is a structure, system or component.

**Abb. 2.2** Klassifizierung leittechnischer Systeme und Komponenten nach dem Standard der IAEA /IAE 02/

Im IAEA Safety Standard SSG-30 /IAE 14/ werden die sicherheitstechnischen Funktionen noch detaillierter identifiziert und nach ihrer sicherheitstechnischen Bedeutung kategorisiert. Dabei werden alle Betriebszustände einschließlich des Normalbetriebs berücksichtigt. Die sicherheitstechnische Bedeutung richtet sich nach den folgenden Kriterien:

1. Die Auswirkungen eines Versagens bei Anforderung der Funktion,
2. die Häufigkeit, mit der ein angenommenes, auslösendes Ereignis die betreffende Funktion anfordert,
3. die Bedeutsamkeit, mit der die Funktion dazu beiträgt einen kontrollierten oder sicheren Zustand zu erreichen.

Im Unterschied zu den oben beschriebenen Kategorisierungen berücksichtigen die drei Sicherheitskategorien der IAEA nicht nur den Einsatzbereich einer Funktion, sondern auch, wie schwer die Auswirkungen sind, die ein Ausfall der Funktion bei Anforderung hat.

In die höchste Sicherheitskategorie 1 fallen alle Funktionen, die zum Erreichen ei-nes kontrollierten Zustands nach einer Betriebsstörung oder nach einem Auslegungs-störfall eingesetzt werden und deren Versagen bei Anforderung sehr schwere Auswir-kungen zur Folge hätte. In Sicherheitskategorie 2 fallen zwar auch Funktionen, die zur Herstellung eines kontrollierten Zustands nach einer betrieblichen Störung oder einem Auslegungsstörfall zum Einsatz kommen, die jedoch bei einem Ausfall bei Anforderung nur „mittelschwere“ Auswirkungen haben. Zudem fallen in Kategorie 2 auch Funktio-nen, die zum Erreichen und zur Aufrechterhaltung eines sicheren Zustands benötigt werden und die bei Versagen sehr schwere Auswirkungen haben. Eine Übersicht über die Sicherheitskategorien und ihren Zusammenhang zwischen den unterschiedlichen Sicherheitsfunktionen und ihren Auswirkungen bei Ausfall ist in Tab. 2.4 gegeben.

**Tab. 2.4** Zusammenhang zwischen Funktionen, die in der Analyse versagensauslösender Ereignisse berücksichtigt werden müssen, und unterschiedlichen Sicherheitskategorien /IAE 14/

| **Sicherheitstechnisch wichtige Funktionen** | **Schwere der Auswirkungen, wenn die Funktion nicht ausgeführt wird** | | |
|---|---|---|---|
| | **hoch** | **mittel** | **gering** |
| Funktionen zum Erreichen eines kontrollierten Zustands nach einer Betriebsstörung | Sicherheits-Kat.1 | Sicherheits-Kat.2 | Sicherheits-Kat.3 |
| Funktionen zum Erreichen eines kontrollierten Zustands nach Auslegungsstörfall | Sicherheits-Kat.1 | Sicherheits-Kat.2 | Sicherheits-Kat.3 |
| Funktionen zum Erreichen und zur Aufrechterhaltung eines sicheren Zustands | Sicherheits-Kat.2 | Sicherheits-Kat.3 | Sicherheits-Kat.3 |
| Funktionen zur Schadensvermeidung in Folge von auslegungsüberschreitenden Zuständen | Sicherheits-Kat.2 oder 3 | Nicht Kategorisiert | Nicht Kategorisiert |

#### 2.2.5.1 Softwareaspekte für rechnerbasierte Systeme nach DIN

Noch einen Schritt weiter geht die DIN EN 61513 /DIN 02/, die entsprechend der Abstufung der leittechnischen Funktionen in die Kategorien A, B und C auch eine Klassifizierung von leittechnischen Systemen in die Klassen 1, 2 und 3 durchführt. Die verschiedenen Klassen mit ihren Aufgaben sind in Tab. 2.5 dargestellt.

**Tab. 2.5** Beispiele für die Klassifizierung von Leittechniksystemen nach DIN EN 61513

| | Kasse 1 | Klasse 2 | Klasse 3 | Nicht klassifiziert |
|---|---|---|---|---|
| Schutz- und Betätigungssysteme für Sicherheitsfunktionen | **X** | | | |
| Steuerung des Notstromsystems | **X** | | | |
| Automatisierungs-, Steuerungs- und Regelungssysteme der Anlage | | **X** | **X** | **X** |
| Mensch-Maschine-Kommunikationssysteme (MMK) | | **X** | **X** | **X** |

Laut DIN EN 61513 /DIN 02/ wird diese Klassifizierung von leittechnischen Systemen notwendig, da die Kategorisierung und Zuordnung von rechnerbasierten Systemen nicht mehr so eindeutig ist, wie die der festverdrahteten Technologie, da:

- bei rechnergestützten Systemen mehrere Funktionen mit denselben Hardwarekomponenten realisiert werden können, während bei der festverdrahteten Technologie die einzelnen Funktionen üblicherweise als Ketten separater elektronischer Komponenten oder Relais realisiert werden.
- rechnergestützte Systeme eine Anzahl von Hilfsfunktionen (wie beispielsweise zur Selbstüberwachung, Diagnose) beinhalten, die in der Anlagenauslegung nicht kategorisiert werden und die bei einer funktionalen Trennung mit einem niedrigeren Qualifikationsniveau auskommen können.
- bei den Funktionen der höchsten Sicherheitskategorie die Auswahl der Systemarchitektur beschränkt sein kann, um die Komplexität zu beschränken und die Implementierung zu erleichtern.
- Anforderungen an die Architektur der Systeme gestellt werden können, die nicht mit den Einzelfunktionen zusammenhängen wie beispielsweise funktionale Trennung, internes Systemverhalten, Komplexität, Vorkehrungen gegen GVA, sondern die leittechnischen Systeme und die Eigenschaften der zur Realisierung dieser Systeme verwendeten Gerätefamilie und die Qualifizierung solcher Systeme betreffen.

Laut /DIN 02/ sind leittechnische Systeme der Klasse 1 grundsätzlich dafür gedacht, Funktionen der Kategorie A auszuführen. Es ist jedoch auch möglich, sie für Funktionen einer niedrigeren Kategorie einzusetzen.

Die Zuordnung der Systemanforderung richtet sich immer nach der implementierten Funktion mit der höchsten Kategorie. Funktionen, die in eine geringere Kategorie eingestuft wurden, dürfen nicht mit Funktionen einer höheren Kategorie interferieren. Systemanforderungen der Klasse 1 dienen der Erstellung einer hochzuverlässigen Software /DIN 60/ und umfassen alle Stufen der Software-Entwicklung und Dokumentation, einschließlich Anforderungsspezifikation, Auslegung, Realisierung, Verifizierung, Validierung und Betrieb. Die entsprechenden Anforderungen sind in Tab. 2.6 zusammengefasst.

**Tab. 2.6** Anforderungen an Auslegung und Qualifizierung von leittechnischen Systemen und Geräten nach DIN IEC 61513 /DIN 02/

| Klasse 1 | Klasse 2 | Klasse 3 |
|---|---|---|
| **Typische geeignete Geräte** | | |
| Nach kerntechnischen Normen der IEC entwickelt | Ausgewählte qualifizierte, kommerziell erhältliche Einrichtungen und Geräte | Ausgewählte kommerziell erhältliche Einrichtungen und Geräte |
| **Anforderungen an Voraussetzungen für die Geräte/Gerätefamilien** | | |
| Trennung von Redundanzen zur Erfüllung des Einzelfehlerkriteriums | | |
| Deterministisches Verhalten bei statischer Abfolge von Aufgaben | Deterministisches Verhalten auch durch geeignete Dimensionierung in Bezug auf die Anforderungen | |
| Minimierung der Abhängigkeit des Systembetriebs von Anlagenanforderungen | | |
| Einrichtungen zur Selbstprüfung: IEC 60880 und IEC 60987 | | |
| **Prüfbarkeit** | | |
| Merkmale zur Realisierung der Unabhängigkeit: IEC 60709 | | |
| **Qualifizierungsanforderungen** | | |
| Hardware: IEC 60987<br>Software: IEC 60880 und IEC 60880-2 | Hardware: IEC 60987<br>Software: IEC 61513 | |

## 2.3 Sicherheitstechnisch wichtige Einrichtungen in Kernkraftwerken

Nach /BMUB 12/ ist eine sicherheitstechnisch wichtige Einrichtung eine Einrichtung, die erforderlich ist, den Reaktor jederzeit aus dem bestimmungsgemäßem Betrieb, bei Störfällen, sehr seltenen Ereignissen und bei Einwirkungen von innen und außen sowie bei Notstandsfällen

- sicher abzuschalten und in abgeschaltetem Zustand zu halten
- die Nachwärme abzuführen
- das Auftreten unkontrollierter Kritikalität zu verhindern sowie
- die erforderliche Vorsorge gegen Schäden zu gewährleisten und
- jede Strahlenexposition oder Kontamination von Personen, Sachgütern oder der Umwelt unter Beachtung des Standes von Wissenschaft und Technik auch unterhalb der festgesetzten Grenzwerte so gering wie möglich zu halten

Diese Definition schließt die gesamte Wirkungskette zur Erfüllung der sicherheitstechnisch wichtigen Funktion ein, d. h. Sensoren/Sonden, Messumformer, Baugruppen zur Verarbeitung der elektrischen Signale zur Ansteuerung der Aktuatoren, Baugruppen zur Ansteuerung der Aktuatoren sowie die Aktuatoren (Pumpen, Ventile, Armaturen, Schalter), die zur Erfüllung der jeweiligen sicherheitstechnisch wichtigen Funktion, z. B. Reaktor abschalten und in abgeschaltetem Zustand halten, erforderlich sind, zählen zu den jeweiligen sicherheitstechnisch wichtigen Einrichtungen. Hierbei gehören gemäß den Ausführungen in Kapitel 2.2.2 Messumformer (Eingabeebene), Baugruppen zur Verarbeitung der elektrischen Signale zur Ansteuerung der Aktuatoren (Verarbeitungsebene) und Baugruppen zur Ansteuerung der Aktuatoren (Ausgabeebene) zu den leittechnischen Einrichtungen/zum Leittechniksystem für die Realisierung der sicherheitstechnisch wichtigen Funktion. Systeme, die für die Stromversorgung der sicherheitstechnisch wichtigen Einrichtungen erforderlich sind, sind ebenfalls in der Wirkungskette für die Ausführung der sicherheitstechnischen wichtigen Funktion(en) einzubeziehen.

Um beispielsweise die sicherheitstechnisch wichtige Funktion „Reaktor sicher abschalten und in abgeschaltetem Zustand halten“ in einem Druckwasserreaktor (DWR) zu erfüllen, sind u. a. folgende sicherheitstechnisch wichtige Einrichtungen notwendig:

- Steuerstäbe/Steuerstabsystem (Steuerstabeinwurf-Funktion)
- Borierpumpen mit Einspeiseweg (Boreinspeisung)
- Zusatzborierpumpen mit Einspeiseweg (Boreinspeisung)
- Reaktorschutzsystem (Leittechnisches System zur Auslösung der Reaktorschnellabschaltung)
- Begrenzungssystem (Steuerstabeinwurf-Funktion)
- Einrichtungen zur elektrischen Stromversorgung der benötigten Einrichtungen zur Erfüllung der Funktion „Reaktor sicher abschalten und in abgeschaltetem Zustand halten“ (z. B. Stromversorgung der Borierpumpen und der Zusatzborierpumpen, Stromversorgung des Reaktorschutzsystems, usw.)

Sicherheitstechnisch wichtige leittechnische Einrichtungen können softwarebasiert sein.

# 3 Für das Vorhaben identifizierte Ereignisse aufgrund von Softwarefehlern außerhalb der Kerntechnik

Im Folgenden werden Ereignisse beschrieben, bei denen es aufgrund von Softwarefehlern zu unterschiedlichstem Fehlverhalten von Flugzeugen oder zu Zwischenfällen in der Strahlentherapie gekommen ist. Zudem wird am Beispiel eines Ereignisses die Wirkungsweise der Malware „stuxnet“ beschrieben, die zur gezielten Manipulation von Siemens Simatic WinCC SCADA (Supervisory Control and Data Acquisition) Systeme entwickelt wurde. Die Beschreibung der Wirkungsweise von „stuxnet“ erfolgt im Rahmen dieses Kapitels, da Siemens Simatic WinCC SCADA (Supervisory Control and Data Acquisition) Systeme überwiegend zur Steuerung von technischen Prozessen im nichtnuklearen Bereich eingesetzt werden /SIE 16/. Ereignisse aus dem nichtnuklearen Bereich, zu denen nur wenige Informationen zur Verfügung stehen, die aber trotzdem einen Software-Aspekt aufweisen, werden in Kap. 3.6 kurz zusammengefasst.

## 3.1 Absturz der Ariane 5

Am 4. Juni 1996 endete der Jungfernflug der Ariane 5 Rakete bereits 39 Sekunden nach dem Start, als die Rakete von ihrem Kurs abkam und sich selbst zerstörte. In der Folge wurde von der ESA (European Space Agency) und dem CNES (Centre national d’études spatiales) ein unabhängiger, internationaler Untersuchungsausschuss zusammengestellt, mit dem Ziel, den Grund des Fehlers herauszufinden. Neben der Ursachenklärung wurde der Ausschuss auch damit beauftragt, geeignete Qualifikationstests und Abnahmeprüfungen zu entwickeln, mit denen sich das aufgetretene Problem rechtzeitig entdecken ließe, und Korrekturmaßnahmen zu definieren, mit denen sich Fehler und andere Systemschwächen beheben ließen /LIO 96/.

Mittels der aufgezeichneten Telemetriedaten der Bodenkontrolle bis 42 Sekunden nach dem Start, den Daten der Radarstation und Filmen von Infrarotkameras konnte der Fehler schnell auf das Inertiale Navigationssystem (Inertial Navigation System, INS) der Steuerungseinheit eingegrenzt werden. Dieses System berechnet auf einem internen Computer aus den Daten der Beschleunigungssensoren („strap down“) die Winkel und Geschwindigkeiten. Diese Daten werden über einen Datenbus auf den Bordcomputer übertragen, der für das Flugprogramm und die Kontrolle der Düsen der Feststofftriebwerke und der Vulcain Raketentriebwerke zuständig ist.

Um die Zuverlässigkeit zu erhöhen, ist das INS redundant ausgelegt, so dass ein zweiter Computer mit identischer Hard- und Software automatisch aktiviert wird, falls das aktive INS ausfällt. Die Software dieses Systems wurde von der Vorgängerversion, der Ariane 4, fast ohne Änderungen übernommen /LIO 96/.

Der Untersuchungsausschuss fand heraus, dass die Ursache des Unfalls ein Designfehler in der Software war. Da die Ariane 5 mit einer wesentlich höheren Geschwindigkeit als die Ariane 4 flog, wurde ein unerwartet hoher Wert für die horizontale Geschwindigkeit der Ausrichtungsfunktion (Horizontal Bias) gemessen. Bei der Umwandlung einer 64-Bit-Gleitkomma Variablen in eine vorzeichenbehaftete 16-Bit-Ganzzahl, kam es zu einem Fehlerzustand in der Software (exception). Da die 64-Bit-Gleitkommazahl einen größeren Wert hatte, als mit einer vorzeichenbehafteten 16-Bit-Ganzzahl darstellbar ist, kam es zu dem Fehlerzustand in der Software. Im Gegensatz zu einigen vergleichbaren Variablen im Quellcode (Ada-Programmiersprache) war die Umwandlung dieser Variable nicht gegen diesen Softwarefehler abgesichert.

Der Bordcomputer konnte den zweiten Computer des Navigationssystems nicht aktivieren, da dieser bereits beim vorherigen Datensatz 72 Millisekunden zuvor aufgrund des gleichen Softwarefehlers ausgefallen war. So wurden vom Navigationssystem keine Flugdaten mehr an den Bordcomputer gesendet, sondern Informationen über die Fehlerdiagnose. Der Bordcomputer interpretierte diese Daten jedoch als Flugdaten, stellte eine vermeintliche Kursabweichung fest und sendete Befehle zur Ablenkung der Triebwerke.

Aufgrund der Schwenkung der Feststofftriebwerke und der Vulcain-Raketentriebwerke erreichte die Rakete einen Anströmwinkel von mehr als 20° und konnte den hohen aerodynamischen Kräften nicht mehr standhalten, so dass die Triebwerke abbrachen. Daraufhin wurde das Selbstzerstörungssystem der Rakete ausgelöst.

Das fehlerverursachende Softwaremodul, das für die Ausrichtung der Rakete vor dem Start vorgesehen ist, blieb die ersten 40 Sekunden nach dem Start aktiv. Diese Zeitspanne wurde ursprünglich für die Ariane 4 benötigt, um bei den Startvorbereitungen einen Countdown einfacher unterbrechen zu können. Für die Ariane 5 war diese Funktion aufgrund anderer technischer Voraussetzungen nicht mehr verwendbar und daher unnötig.

**Empfehlungen des Untersuchungsausschusses**

Der Untersuchungsausschuss kam zu dem Ergebnis, dass die Analysen trotz umfangreicher Überprüfungen und Tests während der Softwareentwicklung ungeeignet waren, den Fehler im Navigationssystem zu entdecken.

Für eine Verbesserung der Zuverlässigkeit von Software gaben sie die folgenden Empfehlungen:

1. Abschaltung der Ausrichtungsfunktion des INS sofort nach dem Start. Allgemein sollten niemals Softwarefunktionen während des Fluges ausgeführt werden, die nicht benötigt werden.

2. Prüfeinrichtungen sollten Systemtests, soweit es die Technik erlaubt, unter möglichst realistischen Rahmenbedingungen durchführen, d. h. es sollten realistische Datensätze und echte Geräte verwendet werden und Tests kompletter Abläufe simuliert werden.

3. Durch Bodentests mit den aus den Flugparametern simulierten Beschleunigungssignalen wäre der Fehler wahrscheinlich entdeckt worden.

4. Es wurde als kritisch eingestuft, dass die Ariane 5 auch ohne die vom Bordcomputer falsch interpretierten Daten aus dem INS abgestürzt wäre. Denn die projektierte Behandlung der in diesem Fall vorliegenden Softwarefehlerart (exception) sah ein Herunterfahren des INS-Computers vor. Ein gleichzeitig an beiden INS Computer vorliegender Exception-Fehler hätte dann zum Ausfall der beiden redundanten INS geführt.

5. Für alle rechnerbasierten oder programmierbaren Geräte sollte eine Qualitätsprüfung durchgeführt werden und Berichte über die durchgeführten Systemtests erstellt werden. Auf alle Einschränkungen beim Betrieb der Geräte sollte ausdrücklich verwiesen werden.

In den weiteren Empfehlungen werden genaue Anweisungen zum Validieren und Verifizieren, die Anforderung an die durchzuführenden Tests und die Verwendung von Daten gegeben. Zudem wird eine bessere Zusammenarbeit in interdisziplinären Teams mit Ingenieuren, Softwareentwicklern etc. gefordert.

## 3.2 Unfälle in der Strahlentherapie

Von 1985 bis 1987 kam in den USA und in Kanada der Linearbeschleuniger Therac-25 in der Strahlentherapie zum Einsatz, der aufgrund mehrerer Softwarefehler und des fahrlässigen Umgangs bei der Fehlersuche zu mindestens drei Todesfällen führte.

Der Therac-25 war eine Weiterentwicklung der Therac-Serie, die von der kanadischen Firma Atomic Energy of Canada Limited (AECL) entwickelt und gebaut wurde. Die Vorgängermodelle besaßen zwar bereits Computerunterstützung zur Erleichterung der Bedienung, die Software wurde jedoch nicht für den Betrieb benötigt. Als Schutzvorrichtungen dienten sogenannte Hardware Interlocks und alle Sicherheitssysteme wurden manuell bedient und durch analoge Messgeräte überwacht.

Der Therac-25 war ein Dual-Mode Linearbeschleuniger, der zur Behandlung der Patienten sowohl mit Röntgenstrahlung (25 MeV) für die Bestrahlung von tieferliegendem Gewebe als auch mit Elektronenstrahlung (5-25 MeV) für die Bestrahlung von oberflächennahem Gewebe eingesetzt werden konnte. Diese beiden unterschiedlichen Behandlungsmodi spielten eine wichtige Rolle bei der Entstehung der Unfälle. Für die Behandlung mit Elektronenstrahlen konnte die Energie und der Strahlstrom eingestellt werden. Durch die vom Computer eingestellten Magnete wurde der Strahl aufgeweitet und seine Konzentration verringert. Bei der Behandlung mit Röntgenstrahlen wurde die Energie auf 25 MeV festgelegt. Für die Erzeugung der Röntgenstrahlen wurde eine 100-fach größere Brillianz (stärke des Elektronenstrahls) eingestellt und ein Drehteller mit vier Komponenten in den Strahlengang eingebracht: ein Target, um die Elektronenstrahl in Röntgenstrahlen umzuwandeln, ein Filter zur Aufweitung und Glättung des Strahlenprofils, ein Set von Kollimatoren, zur Formung des Strahls sowie ein Messgerät. Ohne diese vier Komponenten im Strahlengang war die Strahlungsdosis so groß, dass sie bei Patienten zu einer fatalen Überdosis führen würde.

Beim Therac-25 wurde die Positionierung des Drehtellers mit den vier Komponenten über einen Computer gesteuert, Sensoren zeichneten die Drehungen auf und sendeten die Daten zur Überprüfung an die Software /LEV 93/.

Eine Behandlung lief so ab, dass der Mitarbeiter den Patienten zunächst im Behandlungsraum positionierte und die nötigen Parameter wie die Behandlungsmodus (Drehung des Drehtellers) und das Bestrahlungsfeld einstellte. Nach dem Verlassen des Behandlungsraums wurden die Behandlungsparameter (Behandlungsmodus,

Energieniveau, Dosis und Zeit), das Bestrahlungsfeld und die Drehung des Drehtellers noch einmal in den Computer eingegeben. Wenn sich die Daten zwischen den Behandlungen nicht geändert hatten, konnte der Mitarbeiter auch einfach die Eingabetaste betätigen und die Daten von der letzten Behandlung übernehmen. Die Software überprüfte die Daten mit den Gerätesensormessungen im Behandlungsraum. Wenn auf dem Display der Status verifiziert war, konnte mit der Behandlung begonnen werden /LEV 93/.

Während des Betriebs des Therac-25 kam es im Laufe der Jahre in verschiedenen Krankenhäusern in den USA und in Kanada zu sechs schweren Behandlungsfehlern, in deren Folge mindestens drei Menschen an einer Strahlenüberdosis verstarben. Alle Unfälle entstanden dadurch, dass die Maschine im Röntegenmodus arbeitete ohne dass der Drehteller mit Target, Filter und Kollimatoren im Strahlengang befand, so dass es zu einer erheblichen Überdosis von Elektronenstrahlung kam. /LEV 93/

Da nach den ersten Unfällen seitens des Herstellers eine Fehlfunktion ausgeschlossen wurde und sich die Fehlersuche auf die Hardware des Systems konzentrierte, blieb die Fehlerursache zunächst unerkannt. Die Suche nach der Ursache wurde auch dadurch erschwert, dass der Fehler lange nicht reproduzierbar erschien.

Die eigentliche Fehlerursache waren zwei Programmierfehler. In einem Fall kam es zu einer fehlerhaften Synchronisation der beiden computergesteuerten Prozesse der Messwerterfassung und Messwertsteuerung. Nach Eingabe der Behandlungsparameter wurden über die Software die Geräte entsprechend eingestellt, wozu 8 Sekunden benötigt wurden. Während dieser 8 Sekunden wurden Änderungen der Parameter im Eingabefeld von der Software nicht mehr weiter verarbeitet, obwohl sie auf dem Bildschirm des Mitarbeiters angezeigt wurden. In mindestens zwei Fällen kam es aufgrund dieses Softwarefehlers zu schweren Behandlungsfehlern. In beiden Fällen wurde von einer Mitarbeiterin nach der Eingabe aller Betriebsparameter bemerkt, dass der Betriebsmodus noch falsch eingestellt war. Nachdem sie diesen innerhalb der 8 Sekunden von Röntgenstrahlen auf Elektronenstrahlen umgestellt hatte und die Daten auf dem Bildschirm verifiziert wurden, begann sie mit der Behandlung. Da das System jedoch nur überprüfte, ob die Cursor-Position am Ende des Eingabefelds stand und nicht, ob Daten innerhalb des 8-Sekunden-Zeitfensters verändert wurden, wurde ihre Eingabe ignoriert. Die für das System nun inkonsistenten Daten (Eingabewerte für unterschiedliche Modi) führten zu einer Fehlermeldung „malfunction 54“, die weiter als „dose input 2“ Fehler beschrie-

ben wurde. Eine weitere Fehleranalyse konnte aufgrund mangelnder Fehlerbeschreibung nicht erfolgen. Auch im Betriebshandbuch fanden sich keine weiteren Erklärungen. Zudem wurde von der Maschine nur eine „treatment pause“, also eine Behandlungspause angezeigt, die auf ein nicht so schwerwiegendes Problem hindeutete. Die Mitarbeiterin konnte die Behandlung durch Drücken von P („proceed“) ohne weitere Einschränkungen fortsetzen.

Die amerikanische Arzneimittelzulassungsbehörde (Food and Drug Administration, FDA) fand später bei ihren Untersuchungen heraus, dass der fahrlässige Umgang mit Fehlermeldungen darauf zurückzuführen war, dass es ohne erkennbare Gründe bis zu 40 Fehlermeldungen am Tag gab. Diese gaben lediglich die Information „malfunction xx“ aus, welche nirgendwo dokumentiert war. Viele Fehler ließen sich scheinbar durch Drücken von P („proceed“) beheben, andere erforderten einen kompletten Neustart der Maschine. Zudem wurde den Mitarbeitern bei der Einweisung erklärt, dass aufgrund der Sicherheitsvorkehrungen keine Überdosierung der Patienten möglich sei.

Es fand sich noch ein zweiter Softwarefehler im Quellcode des Therac-25, der vermutlich mindestens zwei weitere Opfer forderte. Während der Einstellung der Maschine überprüfte die Software periodisch die Konfiguration des Drehtellers. Die Prüfvariable, die die Notwendigkeit der Positionsprüfung angab, wurde im Quellcode immer um eins inkrementiert, anstatt auf einen festen Wert ungleich Null gesetzt zu werden. Da es sich bei der Variable um einen 8-bit-Parameter handelte, wurde diese bei jeder 256. (mod $2^8$) Erhöhung wegen Überlaufs auf den Wert Null gesetzt. Wenn der Mitarbeiter die Eingabe im Computer gerade zu dem Zeitpunkt beendete, an dem der Parameter auf Null stand, wurden die Einstellungen nicht überprüft und die Maschine arbeitete offenbar ohne dass der Drehteller mit Target, Filter und Kollimatoren sich im Strahlengang befand. /PRE 11/, /LEV 93/.

**Ergebnisse aus den Untersuchungen der FDA**

Nach dem fünften Unfall begann die amerikanische Arzneimittelzulassungsbehörde mit einer genaueren Untersuchung der Unfallserie. Es stellte sich heraus, dass bei der Softwareentwicklung und dem Gesamtdesign des Therac-25 eine Reihe von Fehlern gemacht wurden, die schließlich zu den Unglücken führen konnten. Häufig wird der Therac-25 als Lehrbeispiel für die Anforderungen an Software im sicherheitsrelevanten Bereich behandelt:

- **Softwareentwicklungsprozess**

  Der Softwareentwicklungsprozess war nicht nur für die Fehlerentstehung verantwortlich, sondern trug auch zu einer langwierigen und schwierigen Fehleranalyse bei.

  Das Programm, das von einer einzigen Person über mehrere Jahre geschrieben wurde, war schon aufgrund des Designs für den Einsatz in einem sicherheitsrelevanten Bereich ungeeignet. Die Software beinhaltete keine Selbstkontrollen, Fehlererkennungs- und Fehlerbehandlungsmechanismen. Es gab keinerlei unabhängige Tests, ob die Maschine und ihre Software korrekt arbeiteten.

  Während der Entwicklungsphase wurde keine Qualitätssicherung der Software durchgeführt, es gab keine Softwarespezifikationen und Testpläne. Auch wurde die Software nur sehr unzureichend dokumentiert. Das Softwaredesign war nicht klar strukturiert und im Detail sehr verworren.

  Zwar gab die Herstellerfirma AECL an, dass die Hardware und Software über mehrere Jahre getestet wurde, aber nur ein kleiner Teil der Software wurde an einem Simulator getestet, während der größte Teil Systemintegrationstests waren. Offenbar wurden auch unzureichende Modultests durchgeführt.

- **Fehleranalyse**

  Vor allem der unprofessionelle Umgang bei der Fehleranalyse führte dazu, dass die Fehlfunktion über mehrere Jahre vorlag. Aufgrund der mangelnden Dokumentation der Fehlermeldungen und deren Häufigkeit wurden Fehlermeldungen von den Mitarbeitern konsequent ignoriert.

  Zudem zeigte sich bei der Fehlersuche, dass die Fokussierung auf einen einzelnen Fehler (Software oder Hardware) nicht zielführend war, sondern nur die Analyse des Gesamtsystems geeignet war, die Sicherheit zu erhöhen. Unrealistische Risikoanalysen und zu hohes Vertrauen in deren Ergebnisse führten zudem dazu, dass vor allem die Software anfänglich nicht auf Fehler untersucht wurde. Die nach den ersten Unfällen unternommenen Korrekturen brachten keinerlei Verbesserung, da sie nicht ursächlich für die Fehler waren. Erst als das Gesamtsystem systematisch analysiert wurde, fand man die den Unglücken zugrunde liegenden Fehler.

- **Hardware Sicherheitsschaltungen**

  Der schwerste Fehler lag im Design des Gesamtsystems, das keine unabhängige Absicherung im Falle eines Softwarefehlers besaß. Ein Schutz vor Softwarefehlern sollte sowohl in der Software selber durch Kontrollmechanismen als auch in der Hardware integriert sein. Obwohl im Vorgängermodell unabhängige Schutzvorrichtungen in der Hardware vorhanden waren, wurden diese aufgrund des großen Vertrauens in die Software entfernt. Spätere Untersuchungen fanden heraus, dass der Softwarefehler der Eingabe-Synchronisation, der zum Tod zweier Patienten führte, auch in der Software vom Vorgängermodell Therac-20 vorhanden war. Da bei diesem Gerät die unabhängigen Schutzvorrichtungen jedoch rechtzeitig eingriffen, kam es niemals zu Unfällen.

## 3.3 Zwischenfall beim Flug des Airbus A321 von Bilbao nach München

Am 11. November 2014 kam es auf dem Linienflug LH 1829 der Lufthansa von Bilbao nach München zu einer schweren Störung. Ausgelöst durch ein computergesteuertes Schutzsystem wurde der Airbus A321 kurz nach dem Start in einen steilen Sinkflug gezwungen, der es den Piloten nahezu unmöglich machte, das Flugzeug zu navigieren. Der Sachverhalt wurde in einem Zwischenbericht der Bundesstelle für Flugunfalluntersuchung in Braunschweig veröffentlicht und ist ein weiteres Beispiel für die möglichen Auswirkungen von Softwarefehlern in computergesteuerten Systemen /BFU 14/.

Die Steuerung moderner Flugzeuge basiert auf dem sogenannten „Fly by Wire“ System, bei dem Steuerbewegungen nicht mehr mechanisch oder hydraulisch an die Steuerflächen oder Rotoren übertragen werden, sondern über Sensoren, deren elektrische Signale Aktoren (Elektromotoren, Hydraulik) an den Steuerflächen ansteuern.

Die Steuerung des Airbus A321 basiert auf sieben Flugsteuerungsrechnern, welche die Steuerflächen ansteuern:

- zwei Elevator Aileron Computer (ELAC)
- drei Spoiler Elevator Computer (SEC)
- zwei Flight Augmentation Computer (FAC)

Die von diesen Systemen übertragenen elektrischen Steuersignale bewirken dann das hydraulische Auslenken der Steuerflächen. Die Verarbeitung kann auf Basis unterschiedlicher Steuerungsmodi („Control Laws“) erfolgen. Sie legen fest, wie die Flugsteuerungsrechner die Befehle zur Steuerflächen-Verstellung verarbeiten. Die drei wesentlichen „Control Laws“ sind: „Normal Law“, „Alternate Law“ und „Direct Law“ /BFU 14/.

Um gefährliche Flugzustände zu verhindern, gibt es in den Flugsteuerungsrechnern verschiedene Schutzfunktionen, die beispielsweise einen automatischen Schutz vor zu hohen Anstellwinkeln bieten oder Lastvielfache limitieren. Diese Schutzfunktionen sind im „Normal Law“ vollständig verfügbar, im „Alternate Law“ gibt es eine begrenzte Auswahl an Schutzfunktionen, während im „Direct Law“ keine Schutzfunktionen vorhanden sind /BFU 14/.

Auch der A321 verfügt über die oben genannte Schutzfunktion vor zu hohem Anstellwinkel, die im vorliegenden Fall durch eine unvorhergesehene Verkettung von Ereignissen zu einem schweren Zwischenfall führte.

Zu hohe Anstellwinkel können dazu führen, dass die Strömung an der Tragfläche abreißt und die Flugbewegungen nicht mehr zu kontrollieren sind. Daher werden die Anstellwinkel mittels dreier redundanter Sensoren AOA (engl. angle of attack) gemessen. Sie bestehen jeweils aus einer beheizbaren und beweglichen Fahne, die dem Luftstrom ausgesetzt ist. Die Drehbewegung dieser Fahne wird für jeden Sensor in ein elektrisches Signal umgewandelt und damit im jeweiligen Air Data Reference (ADR) System der Anstellwinkel des Flugzeugs (AOAcor) bestimmt. Der Flugsteuerungsrechner Elevator Aileron Computer (ELAC) erhält die Werte aller drei AOA-Sensoren über die jeweiligen ADR-Systeme. Weicht einer der Werte stark vom Mittelwert ab, wird er für die weitere Berechnung nicht berücksichtigt. Zwei fehlerhafte Werte sind durch das System nur schwer erkennbar. Im ELAC wird aus den gemessenen Werten der Anstellwinkel bestimmt. Liegt dieser Wert über einem bestimmten Grenzwert, wird die Schutzfunktion aktiviert, um einen zu hohen Anstellwinkel zu vermeiden. Wenn zu diesem Zeitpunkt der Autopilot aktiviert ist, schaltet die Funktion den Autopiloten aus und aktiviert sich sofort /BFU 14/.

Die Bundesstelle für Flugunfalluntersuchung in Braunschweig rekonstruierte die Ereignisse des Fluges anhand des „Flight Data Recorders (FDR)“, des „Cockpit Voice Recorders (CVR)“ sowie aus Flugzeug- und Wartungsdokumenten und den Aussagen der Zeugen wie folgt:

Nachdem das Flugzeug 8 Minuten nach dem Start die Wolkenobergrenze durchbrochen hatte, blieb zunächst der Anstellwinkelsensor #1 auf einem konstanten Wert (4,2°) und eine Minute später auch der Wert des Sensors #2 (4,6°). Als Ursache dafür wird das Einfrieren der beiden Sensoren (bei einer Außentemperatur von -35°C) angenommen. Diese Werte sind den Piloten jedoch nicht direkt zugänglich, so dass sie nicht erkennen konnten, was passiert war. Der Autopilot war zu diesem Zeitpunkt im „Climb Mode“.

Die Piloten bemerkten wenige Minuten später, dass der Anstellwinkel ungewöhnlich stark erhöht war und reduzierten daraufhin die Steigrate mit dem Vertical-Speed-Knopf des Autopiloten von 800 ft/min auf 500 ft/min, um das Flugzeug stärker zu beschleunigen. Nachdem der Copilot den Autopiloten ausgeschaltet hatte und die Längsneigung des Flugzeugs korrigiert hatte, bemerkte er, dass sich die Flugzeugnase weiter senkte, obwohl er dieser Bewegung manuell entgegensteuerte.

Aufgrund dieser Beobachtungen übernahm der Pilot die Steuerung des Flugzeugs, das sich zu diesem Zeitpunkt mit einer Rate von 4000 ft/min im Sinkflug befand. Dem Pilot gelang es zwar durch die maximal möglichen Steuerungseingriffe von Hand, die Sinkrate zu reduzieren und das Flugzeug wieder in den Horizontalflug zu überführen, es waren jedoch kontinuierliche Handeingriffe erforderlich und der Autopilot ließ sich nicht mehr einschalten. Zunächst überprüften die beiden Piloten die Richtigkeit der angezeigten Fluggeschwindigkeit anhand des „Quick Reference Handbook (QRH)“. Dann starteten sie nacheinander die beiden Flight Augmentation Computer (FAC) 1 und 2 neu. Das „Aircraft Communications Addressing and Reporting System (ACARS)“ sendete eine automatisch generierte Nachricht an die Techniker der Lufthansa. Diese Nachricht enthielt unter anderem die Werte der drei AOA-Sensoren.

Die beiden Piloten hielten schließlich Rücksprache mit den Technikern der Lufthansa, die durch die Fehlermeldung „PH6AOA3“ des „Centralised Fault Display System (CFDS)“ auf die richtige Spur kamen und den Piloten folgende Mitteilung machten: „…nach Sichtung der AOA Werte, könnte es sein, dass AOA1 und AOA2 eingefroren sind und einen zu hohen Anströmwinkel melden. Falls das Problem weiterhin besteht, ADR 1 und ADR 2 ausschalten, was aber zu Alternate Law führt….“ „…vielleicht reicht es auch nur den ADR 2 auszuschalten…“

Durch das Einfrieren der beiden Sensoren AOA 1 und AOA 2 hatte das System, nach Prüfung der Abweichung vom Mittelwert, den Wert des einzigen noch funktionierenden Sensors verworfen und einen falschen Anstellwinkel berechnet.

Die automatische Schutzfunktion vor zu hohen Anstellwinkeln, die im „Normal Law“ aktiviert ist, leitete dann automatisch den Sinkflug ein und setzte die automatischen Steuerungsfunktionen wie den Autopiloten außer Kraft.

Tatsächlich bewirkte das Ausschalten des ADR 2, dass das Flugsteuerungssystem in „Alternate Law“ (begrenzte Auswahl an Schutzfunktionen) schaltete und das Schutzsystem deaktiviert wurde. Daraufhin wurden die automatischen Steuerungsfunktionen wie der Autopilot wieder funktionsfähig.

Im Nachgang an das Ereignis informierte der Hersteller Airbus alle Betreiber der Flugzeugtypen Airbus A318, A319, A320 und A321 mit den betroffenen AOA-Sensoren, dass falls zwei oder drei AOA-Sensoren bei gleichen Winkeln „blockiert“ sind, es bei einem Ansteigen der Fluggeschwindigkeit zu einer fälschlichen Aktivierung des Schutzsystems vor zu hohen Anstellwinkeln kommen kann. Für diese Situation wurde vom Hersteller Airbus die Empfehlung gegeben, sofort zwei der drei ADRs auszuschalten, um das Flugsteuerungssystem in „Alternate Law“ zu setzen und damit das Schutzsystem vor zu hohen Anstellwinkeln zu deaktivieren.

Die bei diesem Ereignis beobachtete Fehlfunktion des softwarebasierten Flugsteuerungssystems (fehlerhafte Aktivierung der Schutzfunktion vor zu hohen Anstellwinkeln) wurde durch das Einfrieren von 2 der 3 redundanten AOA-Sensoren verursacht, was vermutlich auf einen betriebsbedingten (niedrige Außentemperatur) Hardwareausfall der betreffenden AOA-Sensoren zurückzuführen ist. Das vorliegende Ereignis zeigt daher beispielhaft, dass Softwarefehler aufgrund von betriebsbedingten Hardwareausfällen auftreten können. Des Weiteren deutet das vorliegende Ereignis auf einen Auslegungsfehler in dem softwarebasierten Flugsteuerungssystem hin: Vermutlich wurden nicht alle möglichen Ausfallkombinationen der Messsignale der redundanten AOA-Sensoren in der Software zur Berechnung des AOA-Wertes berücksichtigt. Auf diesen Aspekten wird im Kapitel 5 näher eingegangen.

## 3.4 Absturz des Airbus A320-216 beim Flug von Surabaya/Indonesien nach Singapur

Am 28. Dezember 2014 stürzte der Airbus A320-216 nur 45 Minuten nach dem Start vom Flughafen Juanda in Surabaya/Indonesien ins Meer. Die mit 162 Personen besetzte Maschine war auf dem Weg nach Singapur.

Der Sachverhalt wurde in einem Bericht der KNKT, die indonesische Flugunfalluntersuchungsbehörde, veröffentlicht /KNK 15/ und wird nachfolgend erläutert.

Nachdem das Flugzeug um 5:35 Uhr Ortszeit abgehoben hatte, erreichte das Flugzeug um 5:49 Uhr seine Reiseflughöhe von 32000ft. Um 6:00 Uhr zeigte das Electronic Centralized Aircraft Monitoring-System (ECAM) die Meldung „AUTO FLT RUD TRV LIM 1“ (Auto Flight Rudder Travel Limiter 1) an. Das ECAM, das als „Triebwerksanzeige- und Warnsystem“ bezeichnet werden kann, stellt die wichtigsten Triebwerksparameter auf einer Anzeige dar und informiert die Piloten darüber, wenn bestimmte Parameter der Triebwerksanzeige nicht mehr dem Soll entsprechen. Die angezeigte Meldung war ein Hinweis auf eine Fehlfunktion an einer der beiden Einheiten des Ruderausschlagbegrenzersystems (Rudder Travel Limiter Unit, RTLU).

Laut Flugschreiber (Flight Data Recorder, FDR) kam es jedoch schon eine Minute später um 6:01 Uhr zu einer weiteren Meldung „AUTO FLT RUD TRV LIM SYS“ (Auto Flight Rudder Travel Limiter System), da nun beide Ruderausschlagbegrenzereinheiten ausgefallen waren. Der verantwortliche Pilot führte daraufhin die „ECAM actions“ aus, die ein Reset der Flight Augmentation Computer 1 und 2 (Flugunterstützungscomputer) durch Aus- und Einschalten am Schaltpult vorsahen. Beide Ruderausschlagbegrenzereinheiten waren danach wieder funktionsfähig.

Der verantwortliche Pilot bat um 6:04 Uhr die zuständige Luftraumüberwachung um Erlaubnis 15 Meilen weiter links der vorgesehenen Flugroute zu fliegen, um eine Schlechtwetterzone zu umfliegen. Die neue Route wurde genehmigt und das Flugzeug änderte seinen Kurs. Auf dem Flugdatenschreiber wurde um 6:09 Uhr eine erneute Meldung („AUTO FLT RUD TRV LIM SYS“) des Ruderausschlagbegrenzersystems aufgezeichnet, die von den Piloten durch ein Reset erfolgreich behoben werden konnte. Der Pilot informierte die Luftraumüberwachung über die Kursänderung, welche bestätigt und vom Radar erfasst wurde. Der Pilot fragte zudem um die Erlaubnis auf 38000 ft aufsteigen zu dürfen. Um 6:13 Uhr kam es zum dritten Mal zur Fehlermeldung des Ruderausschlagbegrenzersystems, die mit der gleichen Reset-Maßnahme behoben wurde. Die Meldung wurde jedoch 2 Minuten später um 6:15 erneut ausgegeben. Der Fluglotse wies den Piloten an, auf 34000 ft zu steigen, da sich ein weiteres Flugzeug auf 38000 ft befand. Es gab jedoch keine Rückmeldung auf diese Anweisung. Der Flugschreiber zeichnete um 6:16 Uhr eine weitere Fehlermeldung durch den Flugunterstützungscomputer auf.

Die vollständigen Daten konnten jedoch vom Flugschreiber nicht aufgenommen werden, was darauf hinweist, dass die Piloten ein Reset des Leistungsschalters (circuit breaker) des Flugunterstützungscomputers durchführten, was eine komplette Unterbrechung der Stromversorgung zur Folge hatte, so dass die Daten nicht zum Flugschreiber gesendet werden konnten. Die Stromunterbrechung des Flugunterstützungscomputers verursachte das Abschalten des Autopiloten und wechselte die Flight Control Logic des Flugsteuerungsrechners von „Normal Law“ (alle Schutzfunktionen vollständig verfügbar) zu „Alternate Law“ (nur begrenzte Auswahl an Schutzfunktionen verfügbar, u. a. wird ein Abriss der Luftströmung in diesem Modus nicht durch Schutzfunktionen verhindert).

Erst neun Sekunden nach dem Abschalten des Autopiloten wurde der Steuerknüppel betätigt und das Flugzeug durch die Piloten manuell gesteuert. Das Flugzeug stieg mit 11000 ft/min auf eine Höhe von 38000 ft, was etwa doppelt so schnell ist, wie für Verkehrsflugzeuge bei Windstille zulässig ist. Um 06:17 Uhr wurde die Überziehwarnung aktiviert, die vor einem Strömungsabriss warnte. Der Pilot gab dem Copiloten die Anweisung „pull down“, um die Nase des Flugzeugs nach unten zu steuern. Der Copilot missverstand diese Anweisung jedoch und zog stattdessen den Steuerknuppel rückwärts, wodurch das Flugzeug weiter nach oben gesteuert wurde. Alle weiteren Bemühungen der Piloten das Flugzeug wieder manövrierfähig zu machen, führten zum vollständigen Kontrollverlust.

Um 6:17 Uhr verlor das Flugzeug dann mit einer Geschwindigkeit von 20000 ft/min an Höhe und stürzte schließlich ins Meer.

Es zeigte sich, dass dieser Fehler bereits in mehreren Flügen aufgetreten war und vor allem in den vorherigen zwei Monaten viele Piloten über das Problem berichtet hatten. Auch der verantwortliche Pilot hatte drei Tage vor dem Absturz bereits Probleme mit der Maschine gehabt, als er von Surabaya nach Kuala Lumpur flog. Schon vor dem Start des Fliegers trat der besagte Fehler auf. Der herbeigerufene Flugzeugingenieur führte ein Reset des Leistungsschalters (circuit breaker) des Flight Augmentation Computers mit anschließendem Test durch. Der verantwortliche Pilot und der Ingenieur diskutierten daraufhin die bei diesem Fehler zu treffenden Maßnahmen. Der Ingenieur erklärte, dass der Pilot bei Ansprechen des ECAM ein Reset des Flight Augmentation Computers durchführen kann. Die Fehlermeldung trat jedoch kurze Zeit später erneut auf.

Auch ein Reset des Leistungsschalters des Flight Augmentation Computers konnte das Problem nicht dauerhaft lösen. Der Flugzeugingenieur tauschte daraufhin den Flugunterstützungscomputer 2 aus. Das Problem schien damit behoben zu sein und der Pilot flog daraufhin störungsfrei von Surabaya nach Kuala Lumpur.

Bei der Untersuchung der Absturzursache wurde als vermutlicher Grund für die mehrmalig generierten Meldungen des ECAM-Systems, die auf eine Fehlfunktion des Ruderausschlagbegrenzersystems hinwiesen, Folgendes aufgeführt: Als das Elektronikmodul des RTLU (Rudder Travel Limiter Unit) unter dem Mikroskop vergrößert wurde, wurden Risse an den Lötstellen der beiden Kanäle erkennbar. Diese Risse, die offenbar durch betrieblich bedingte Materialermüdung entstanden sind, waren so groß, dass sie vermutlich zu einer kurzzeitigen Stromunterbrechung der beiden Kanäle der RTLU geführt haben müssen. Diese Stromunterbrechungen hätten zu den diskontinuierlich auftretenden Störmeldungen des Flight Augmentation Computers geführt, die dann über das ECAM-System angezeigt wurden. Die Fehlersuche der Flugzeugingenieure hatte sich jedoch entsprechend dem Fehlerbehebungshandbuch auf den Flight Augmentation Computer beschränkt.

Die Ausführungen im Untersuchungsbericht deuten im vorliegenden Fall darauf hin, dass der Fehler auf dem Elektronikmodul des RTLU (Risse an Lötstellen) anhand seiner Auswirkungen (kurzzeitige Stromunterbrechungen) durch das softwarebasierte Flugzeugsteuerungssystem erkannt und entsprechend interpretiert wurde (Fehlermeldungsanzeigen im ECAM-System). Das vorliegende Ereignis ist daher im engeren Sinne nicht auf einen Softwarefehler zurückzuführen. Die im Fehlerbehebungshandbuch vorgeschriebene Vorgehensweise zur Behandlung der Fehlermeldung (Reset/Neustart der Flight Augmentation Computer 1 und 2) lässt jedoch vermuten, dass bei Vorliegen der Fehlermeldung von einem behebbaren temporären Softwarefehler, d. h. „Soft Error“ (siehe Kapitel 2.1.3), ausgegangen wurde. Ein Hardwareausfall als möglicher Ursache für die Fehlermeldung sowie die integralen Auswirkungen eines derartigen Ausfalls auf das softwarebasierte Flugsteuerungssystem wurden vermutlich nicht vollumfänglich betrachtet. Das vorliegende Ereignis verdeutlicht daher die Problematik der Abgrenzung von Softwarefehlern zu Hardwarefehlern bzw. deren Behandlung in softwarebasierten Leittechniksystemen. Auf diesen Aspekten wird im Kapitel 5.2 näher eingegangen.

## 3.5 Die Malware „Stuxnet“

Im Juli 2010 wurde eine Schwachstelle im Microsoft-Betriebssystem Windows entdeckt, die gezielt dazu genutzt wurde, Malware auf fremde Systeme zu schleusen.

Bereits im Juni desselben Jahres waren im Iran Computer mit einem neuartigen Wurm namens „Stuxnet“ infiziert worden, der gegen Ende des Jahres bereits auf schätzungsweise 100000 Computern installiert war. Anhand der ausführbaren Dateien zeigte sich, dass der Wurm entwickelt worden war, um in Siemens Simatic WinCC SCADA (Supervisory Control and Data Acquisition) Systeme einzugreifen /ZET 10/.

Unter Ausnutzung einer Sicherheitslücke im Microsoft-Betriebssystem konnte „Stuxnet“ über mobile Datenträger wie USB-Sticks in Umlauf gebracht und über Netzwerke weiterverbreitet werden. Der Wurm installiert zwei Programme, zum einen die Schadkomponente, die Daten ausspäht und zum anderen das sogenannte Rootkit, eine Komponente die die Infektion mit Schadsoftware verschleiert /BSI 10/.

Für die Installation des „Stuxnet“ Wurms genügt es bereits, sich den Verzeichnisinhalt eines USB-Sticks mit der Schadsoftware anzeigen zu lassen. Auf diese Weise ist kein aktives Zutun des Nutzers erforderlich und der Wurm kann sich installieren, selbst wenn das automatische Auslesen des Datenträgers deaktiviert wurde. Da das Rootkit bei der Installation signierte Treiberdateien eines bekannten Softwareherstellers nutzt, wird keine Warnung vom Betriebssystem ausgegeben und die Infektion bleibt unbemerkt. Auf welche Weise die hochentwickelte Schadsoftware Zugriff auf den geheimen Zertifikatschlüssel erlangen konnte, ist bisher ungeklärt /HEI 10/.

Nach der Installation der Schadsoftware sucht das Programm nach der Installation der Siemens Software Simatic WinCC oder Simatic PCS7, um Daten auszulesen und die Systeme zu manipulieren.

Laut dem amerikanischen Technologie-Magazin „Wired“ /ZET 10/ fanden Wissenschaftler heraus, dass Stuxnet mit dem Ziel entwickelt wurde, Befehle von SCADA-Systemen abzufangen, um die Kontrolle bestimmter Funktionen von Anlagen zu manipulieren. Demzufolge wurden durch Stuxnet gezielt Frequenzumwandler manipuliert, welche die Drehgeschwindigkeit von Motoren steuern.

Es stellte sich heraus, dass nur Frequenzumwandler einer finnischen und einer iranischen Firma betroffen waren, die mit sehr hohen Frequenzen laufen und die für die Urananreicherung in Zentrifugen verwendet werden können. Aufgrund dieser Erkenntnisse und der Tatsache, dass Stuxnet am stärksten im Iran auftrat (fast 60 % der Infektionen), wird vermutet, dass Stuxnet zur Sabotage der Urananreicherung im Iran entwickelt wurde.

Obwohl der vollständige und originale Quellcode des Stuxnet Wurms nicht veröffentlicht wurde, gibt es umfangreiche Analysen seiner Architektur. Die Hackergruppe Anonymous behauptete 2011 sogar den Quellcode von einer amerikanischen IT-Firma gehackt zu haben /SPI 11/.

## 3.6 Weitere Ereignisse mit Softwarefehlern oder Malware außerhalb der Kerntechnik

In Tab. 3.1 werden weitere Ereignisse aufgeführt, welche im nichtnuklearen Bereich aufgetreten sind.

**Tab. 3.1** Weitere Beispiele für Ereignisse mit Softwarefehlern und Malware im nichtnuklearen Bereich

| Datum | Vorfall | Ursache |
|---|---|---|
| 23.12.2015 | Bei einer Cyberattacke auf drei verschiedene ukrainische Stromversorgungsgesellschaften, kam es zu einem flächendeckenden Stromausfall, der etwa 225000 Verbraucher betraf. | Der Cyber-Angriff wurde in mehreren Schritten durchgeführt. Zunächst wurde das Büronetz der Stromversorger durch Phishing E-Mails mit Malware infiziert, um Benutzeranmeldeinformationen zu stehlen. Mit gestohlenen VPN Zugangsdaten verschafften sie sich die Hacker schließlich Zugang zu industriellen Steuerungssystemen und manipulierten so gezielt die Stromversorgung /ICS 16/ |
| 09.05.2015 | Bei einem Testflug stürzte der Airbus A400M ab. Vier Besatzungsmitglieder sterben. | Bei der Installation neuer Software wurden versehentlich wichtige Kalibrationsparameter gelöscht, die für die Auswertung bestimmter Sensorwerte benötigt wurden. Infolgedessen konnte die elektronische Steuerung die Sensordaten nicht mehr verarbeiten, woraufhin die widersprüchlichen Befehle zur Abschaltung dreier Triebwerke führten /COM 15/. |
| Frühjahr 2015 | Cyberattacke durch Trojaner auf den Bundestag | Nachdem zunächst ein Computer im Bundeskanzleramt infiziert wurde, wurde der Trojaner durch E-Mails weiterverbreitet. |
| 12.02.2014 | Toyota startet Rückrufaktion von 1,9 Millionen Prius-Fahrzeugen | Ein Softwarefehler führte zu erhöhter Hitzebildung in Transistoren und Schaltkreisen der Bordelektronik. Das konnte dazu führen, dass die Steuerung des Hybrid-Antriebs zerstört wurde und der Wagen stehen blieb /WIN 15/. |
| 01.01.2005 | Stromausfall in Brasilien (Rio de Janeiro, Espírito Santo) | Stromausfall wird einer Cyberattacke zugeschrieben, nähere Details sind derzeit nicht bekannt. |
| 1978 | Das Kampfflugzeug F-16 stellte sich beim simulierten Überflug über den Äquator auf den Kopf. | Im Programmcode des computergesteuerten Jagdflugzeuges befand sich in einem Algorithmus ein Vorzeichenfehler bei der Berücksichtigung der geographischen Breite /CER 17/. |

# 4 Für das Vorhaben identifizierte Ereignisse aufgrund von Softwarefehlern in kerntechnischen Anlagen

Im Rahmen dieses Vorhabens wurden verschiedene Ereignis-Datenbanken im nuklearen Bereich nach Ereignissen in softwarebasierter Leittechnik im Hinblick auf Ereignisse, die aufgrund von Softwarefehlern und/oder durch Malware aufgetreten sind, durchsucht. Hierzu zählen Datenbanken der GRS für Ereignisse aus der deutschen Betriebserfahrung, die ICDE-GVA-Datenbank, die COMPSIS-Datenbank und die IRS-Datenbank der IAEA für Ereignisse aus der internationalen Betriebserfahrung. Darüber hinaus wurden Ereignisse unterhalb der Meldeschwelle ausgewertet, welche der GRS im Rahmen des Vorhabens „Entwicklung und Einsatz von Analysemethoden zur Beurteilung softwarebasierter leittechnischer Einrichtungen in deutschen Kernkraftwerken“ /GRS 15b/ zur Verfügung standen. Nachfolgend werden die auf diese Weise ermittelten Ereignisse vorgestellt.

## 4.1 Ereignisse aus der deutschen Betriebserfahrung

### 4.1.1 Fehlerbedingte sekundärseitige Lastabsenkung und nicht erfolgter Stabeinwurf (Ereignis Nr. 1)

Zur Instandsetzung einer gestörten Ionisationskammer der Neutronenflussinstrumentierung sollte der betroffene Gliederzug der Kernaußeninstrumentierung gezogen werden. Hierfür wurden in der softwarebasierten Leittechnik der Begrenzungseinrichtungen Simulierungen vorgenommen. Dadurch entstand eine Störung in einer Betriebsbegrenzung, die u. a. zur schnellen Absenkung der Generatorleistung führte und gleichzeitig den Einwurf und das Fahren der Steuerstäbe durch die Begrenzungseinrichtungen blockierte. Damit waren auch die Stellglieder der sicherheitstechnisch wichtigen Schutzbegrenzungen ausgefallen.

Die Reaktorschutzgrenzwerte des Primärkreises für die Auslösung einer Reaktorschnellabschaltung wurden bei dem Ereignis nicht erreicht. In einem Anforderungsfall wäre die automatische Schnellabschaltung des Reaktors über das autarke, festverdrahtete Reaktorschutzsystem erfolgt. Eine Reaktorschnellabschaltung hätte auch jederzeit von Hand ausgelöst werden können.

Zusätzlich waren durch die Simulierungen die automatische Steuerung der betrieblichen Bor- und Deionateinspeisung und einer HD-Förderpumpe blockiert.

Über Handbefehle hätten diese Einrichtungen aber aktiviert werden können. Weiterhin war die Unterdrückung der Körperschallüberwachung blockiert.

Nach ca. 14 Minuten wurden die Simulierungen von Hand aufgehoben. Die Funktion der Begrenzungseinrichtungen war dadurch wieder hergestellt und die Steuerstäbe konnten wieder verfahren werden.

Die betroffene Reaktorleistungsbegrenzung wurde 1998 von festverdrahteter auf softwarebasierte Leittechnik umgerüstet. Bei der Projektierung der neuen Leittechnik wurde nicht berücksichtigt, dass die vier Ionisationskammern des Leistungsbereichs eines Gliederzuges der Neutronenfluss-Kernaußeninstrumentierung für Instandsetzungsmaßnahmen gemeinsam gezogen werden. Es wurde eine Schaltungslogik realisiert, bei der zwangsläufig durch drei fehlerhafte Signale aus einem Gliederzug die beim Ereignis aufgetretenen Fehlfunktionen der Begrenzungseinrichtungen entstehen.

Die Betreiberin hat nach dem Ereignis die leittechnische Fachanweisung für die Simulation überarbeitet und zusätzliche Bausteine zur Blockade von Signalen mit dem Status „fehlerhaft“ in die softwarebasierte Leittechnik integriert. Durch diese Bausteine wird die Ausbreitung der Signale mit dem Status „fehlerhaft“ begrenzt. Damit werden bei den Simulierungen für das Ziehen eines Gliederzuges eine fehlerhafte Generatorleistung-Sollwertabsenkung und ein Blockieren der Steuerstäbe verhindert. Darüber hinaus wurde in den Funktionsplänen das Blockieren der Signale mit dem Status „fehlerhaft“ gekennzeichnet. In der Dokumentation der Funktionsbausteine wird in einem einleitenden Abschnitt auf die Ausbreitung und Begrenzung von Signalen mit Fehlerstatus hingewiesen.

### 4.1.2 Sporadische Funktionsstörungen in der Leittechnik der Steuerung des Masthubwerks der Brennelementwechselbühne (Ereignis Nr. 2)

Bei Arbeiten mit dem Haupthubwerk der Brennelementwechselbühne mit dem Meisterschalter (Joystick) am Steuerpult kam es sporadisch zu einem Verfahren des Haupthubwerks entgegengesetzt zu der mit dem Meisterschalter angesteuerten Fahrrichtung. Bei

entsprechender Bewegung des Meisterschalters wurde sporadisch die Richtung Senken/Anheben von der Leittechnik fehlinterpretiert und der Vorgang in entgegengesetzter Richtung ausgelöst. Die Fehlbewegungen konnten vom Operateur sofort gestoppt und erneut, dann mit richtiger Reaktion der Steuerung, durchgeführt werden.

Ursache für das Fehlfahren war ein Programmierfehler in der Steuerung des Hubwerks. Über eine Schnittstelle werden Prozessparameter und Startbefehl an die Positionierbaugruppe der Steuerung, in der verschiedene Fahrbewegungen des Masthubwerks in Form von Verfahrprogrammen abgelegt werden, übertragen. Dafür werden mehrere Aufrufzyklen benötigt. Pro Aufrufzyklus wird ein Prozessparameter übertragen. Während dieser Übertragung dürfen sich die vom Anwenderprogramm an die Schnittstelle gelieferten Prozessparameter nicht ändern. Bei den Untersuchungen wurde festgestellt, dass sich der analoge Messwert während der Übertragung ändern kann, der die Auslenkung des Meisterschalters wiedergibt. Die Veränderung dieses Parameters führte zur Wiederholung der Übertragung der Prozessparameter von der Schnittstelle an die Positionierbaugruppe. Dabei kam es sporadisch zu Überschneidungen zwischen einer noch laufenden Übertragung von Prozessparametern und deren Quittierung. Dadurch wurde die Ist-Position des Masthubwerks nicht an die Positionierbaugruppe weitergeleitet. Da das Anwenderprogramm die Quittierung der Übertragung der Prozessparameter nicht abfragte, erteilte es den Startbefehl für das Verfahren des Masthubwerks, obwohl der Parameter nur unvollständig übertragen worden war. Das Verfahrprogramm startete dann mit falsch interpretierter Ist-Position des Masthubwerks am Anfang des Verfahrsatzes und es kam zum Verfahren des Masthubwerks entgegengesetzt der geforderten Fahrtrichtung.

Im Rahmen der Fehlersuche wurden zunächst zusätzliche Überwachungsmodule in die Leittechnik eingebracht. Nach Feststellung der Fehlerursache hat der Betreiber zur Sicherstellung der zuverlässigen Funktion der Steuerung des Masthubwerks verschiedene Programmanpassungen vornehmen lassen. Als redundante Überwachungsmaßnahme wurden einige Überwachungsmodule auch nach den Anpassungen beibehalten, um die korrekte Funktion des Masthubwerks sicherzustellen.

Einige Tage später kam es wiederum zu Störungen im Betrieb der Brennelementwechselbühne. Die Störungen betrafen die Gleichlaufregelung des Fahrwerks, die zu unerwarteten Fahrbewegungen während der Fahrt des Brennelementwechselbühnenfahrwerks führte. Die Störungen traten bei Fahrten ohne Last auf. Das Fahrwerk wurde durch

Betätigung des Not-AUS-Schalters vor Erreichen der Abschaltgrenzwerte gestoppt. Ursache waren ein defekter Weggeber und ein defekter Antrieb. Die Gleichlaufregelung wurde durch fehlerhafte Signale des Positionsgebers (Ist-Wert) des ausgefallenen Antriebes beeinträchtigt.

Etwa vier Monate später kam es bei Handhabungen im Zuge der Entsorgung von Steuerstäben zu einem Fehlverhalten des Bühnenfahrwerks.

In der Nähe der Zielposition zum Absetzen eines Steuerstabes im Lagergestell kam es beim Versuch der Feinjustage des Bühnenfahrwerks bei minimaler Auslenkung des Meisterschalters zu einer starken Beschleunigung des Bühnenfahrwerks. Die automatische Geschwindigkeitsreduzierung erfolgte nicht. Ursache für die unerwartete Beschleunigung war ein Fehler in der Software der betrieblichen Steuerung des Fahrwerks, welcher durch Fehler in der Software-Spezifikation verursacht wurde.

Bei Untersuchungen zur Ursachenermittlung wurde darüber hinaus festgestellt, dass sich das Hauptfuhrwerk schneller als die vom Bühnenfahrer geforderte Geschwindigkeit bewegte. Ursache hierfür waren fehlerhaft durchgeführte Änderungsmaßnahmen, bei denen unter anderem die Vorbelegung für eine fehlerhafte Fahrtfreigabe geändert wurde. Bei fehlerhafter Fahrtfreigabe und nichtbetätigtem Meisterschalter wurde der Geschwindigkeitssollwert für diesen Fall von maximaler auf minimale Geschwindigkeit geändert. Bei der Planung und Umsetzung dieser Änderungen wurde jedoch übersehen, dass dieser Wert nur bei Änderung des Sollwertes an die Positionierbaugruppe übergeben wird. Die Positionierbaugruppe löscht unter bestimmten Bedingungen bei Stillstand ihren internen Sollwert. Dies führte bei Nichtbetätigung des Meisterschalters und Fahrtfreigabe zur Fahrt des Hubwerks mit maximaler Geschwindigkeit, weil vom Meisterschalter kein neuer Wert übergeben wurde.

### 4.1.3 Temporäre Störung von elektronischen Baugruppen (Ereignis Nr. 3)

Es kam zu einer Störung in der schrankinternen Buskommunikation (Rückwandbus) eines Leittechnikschranks. Ein Fehler eines softwarebasierten Automatisierungssystems führte zu einem vollständigen Ausfall der Regelungs- und Überwachungsfunktion im betroffenen Schrank. Der Schrank beinhaltete ausschließlich betriebliche Leittechnikfunktionen. Aufgrund der Auswirkungen auf den Betrieb der Anlage wurde die Reaktorleistung abgesenkt.

In Abstimmung mit dem Hersteller wurde die Störung durch Neustart der Buskommunikation des betroffenen Schrankes behoben.

Die gemeinsam mit dem Hersteller durchgeführte Analyse der Störung ergab, dass Baugruppen einer bestimmten Hardwareversion von der Störung betroffen waren, die nicht mehr mit der ASIC- (Application Specific Integrated Circuit) Technologie bestückt waren, sondern mit neuen Kommunikationsbausteinen in FPGA (Field Programmable Gate Array) Technologie ausgestattet waren.

Diese Umstellung von den ASIC- zu FPGA-Chips war notwendig geworden, weil die Bausteine nicht mehr in ASIC-Technologie hergestellt wurden. Der Programmcode der ASIC Chips wurde nach einigen Anpassungen für die FPGA-Chips übernommen.

Bei der vom Hersteller durchgeführten Ursachenanalyse wurde ein fehlerhaftes Dauersenden eines Kommunikationsbausteins ausgemacht. Dieses Dauersenden führte dazu, dass die Buskommunikation zwischen den Baugruppen des betroffenen Schrankes unterbrochen wurde. Die exakte Ursache für das Dauersenden ließ sich nicht mehr ermitteln. Es wurde jedoch davon ausgegangen, dass ein systematischer Fehler in der Firmware, der nur durch sehr seltene Konstellationen im System zur Wirkung gebracht wurde, das Dauersenden verursacht hatte.

In den betrieblichen leittechnischen Einrichtungen wurden die Kommunikationsbausteine auf Baugruppen der betroffenen Hardwareversion durch modifizierte Kommunikationsbausteine in FPGA-Technologie ausgetauscht. Bei diesen wurden vom Hersteller zur Verhinderung des Dauersendens die folgenden Änderungen in der Firmware vorgenommen:

1. Die Telegrammlängen von Sender und Empfänger wurden begrenzt, so dass ein Dauersenden sofort unterbrochen werden kann und die Busfunktionalität ohne Beeinträchtigung bleibt.

2. Jede Sende-/Empfangseinheit wird nach Abschluss der jeweiligen Kommunikationsfunktion zurückgesetzt, so dass sich das System wieder in einem definierten Zustand befindet.

### 4.1.4 Fehlfunktion eines Lagerkrans (Ereignis Nr. 4)

Beim Anfahren einer Lagerposition im Teilautomatikbetrieb überfuhr ein Portalkran die beiden Softwareendschalter ohne, wie vorgesehen, vollständig abzubremsen. Der Kran fuhr in die mechanische Endlage. Dabei kam es zur mechanischen Beschädigung der Räumschilder und deren Halterungen sowie zu oberflächlichen Betonabtragungen an den Betonsockeln der mechanischen Endlagen durch die Raumschilder.

Im Anschluss an das Ereignis wurden die mechanischen Komponenten, die Antriebe und die leittechnische Ansteuerung (Steuerungssoftware, Automatisierungsgeräte, Lasermess-strecken zur Wegerfassung) des Portalkrans überprüft.

Dabei wurde festgestellt, dass bei Ausfall des Messsignals einer Lasermessstrecke das zuletzt anstehende Messsignal in der der Messtrecke nachgeschalteten Baugruppe gespeichert wird. Hierdurch wird der Steuerung ein falscher Positionswert vorgegeben und bei Weiterfahrt des Krans dieser über die Diskrepanzüberwachung gestoppt. Tritt dieser Fehler bei beiden redundanten Lasermessstrecken gleichzeitig auf, werden die gespeicherten Positionswerte beibehalten und an die Steuerung weitergegeben. Die Diskrepanzüberwachung ist somit bei Weiterfahrt nicht mehr wirksam. Dies kann folglich zum Überfahren der Softwareendschalter führen. Die Vorgabe falscher Positionswerte aus den beiden redundanten Lasermessstrecken wird daher als Ursache für das Ereignis angesehen. Als Ursache für den Ausfall der Messsignale beider redundanten Lasermessstrecken wird Beeinflussung durch Nebel angenommen.

Als Maßnahmen gegen Wiederholung wurde u. a. die Firmware dahingehend modifiziert, dass ungültige Messwerte aus den Messstrecken zu keinen falschen Positionswerten mehr führen können.

Bei dem vorliegenden Ereignis handelt es sich vermutlich um einen Auslegungsfehler in der Steuerungssoftware des Portalkrans. Vermutlich wurden nicht alle möglichen Ausfallkombinationen der Messsignale der redundanten Lasermessstrecken bei der Auslegung berücksichtigt und folglich in der Steuerungssoftware nicht umgesetzt.

### 4.1.5 Ereignisse unterhalb der Meldeschwelle

In /GRS 15b/ wurden Ereignisse unterhalb der Meldeschwelle aus verschiedenen deutschen Anlagen ausgewertet, die im Zusammenhang mit programmierbaren oder rechnerbasierten leittechnischen Komponenten aufgetreten sind. Dabei wurde u. a. untersucht, welche Fehler aufgetreten sind und wie diese behoben wurden. Folgende Softwarefehler wurden dabei identifiziert:

- Baugruppeninterner Fehler im Diagnosepuffer: Baugruppe wurde ausgetauscht
- Kommunikationsstörung: Programm nach Urlöschen der CPU neu übersetzt und übertragen
- Kompletter Programmverlust der Sicherheitssteuerung trotz betriebsbereiter Pufferbatterien: Programm neu übertragen
- Ausfall einer Systemfunktion eines Überwachungssystems: Ertüchtigten Softwarebaustein übertragen
- Automatisierungsprozessor wegen Speicherfehler ausgefallen: Urgelöscht und zurückgesetzt
- Siebbänder schalten sporadisch im Automatikbetrieb nicht zu: Programm der Steuerung angepasst
- Wartenmeldung steht nur ca. 20s an und quittiert sich dann selbst: Programm geändert
- Steuerschrank in Störung: Software geladen und geprüft
- Baugruppe zeigt „Run“ und „Stop“ gleichzeitig an: Telegrammaufträge beim Kommunikationsprozessor neu geladen
- Ausfall der CPU: Busverbindungstelegramme neu geladen und synchronisiert
- Untergruppensteuerung Störung (Rückmeldung „Ein“ fehlt): Neustart der Steuerung
- Programmverlust der Betriebssteuerung: Software wurde wieder eingespielt

Da hierbei ausschließlich betriebliche Systeme betroffen waren, wurde die genaue Ursache für das Auftreten des Fehlers in der Regel nicht weiter untersucht. Der GRS ist daher nur die nach Auftreten ergriffene Maßnahme zur Behebung des Fehlers bekannt.

Wesentliche Beiträge zu Fehlern an Softwarekomponenten haben gemäß /GRS 15b/ Ereignisse geliefert, bei denen Pufferbatterien ausgefallen sind. Pufferbatterien kommen in den Anlagen beispielsweise in CPUs zum Einsatz. Dort werden sie benötigt, um bei Ausfall der Spannungsversorgung den Inhalt des RAM-Speichers zu erhalten. Fällt die Pufferbatterie bei abgeschalteter Spannungsversorgung aus, führt dies dazu, dass der Inhalt des RAM-Speichers verloren geht. Ohne Speicherinhalt läuft die Steuerung bei Wiedereinschalten der Spannungsversorgung nicht an. Folgende Ursachen für den Ausfall der Pufferbatterien wurden in /GRS 15b/ ermittelt:

- Chargenproblem
  Die Pufferbatterien einer Charge wiesen nicht die gewünschten Eigenschaften auf. Nach Entdeckung dieses Problems wurden die Batterien dieser Charge vollständig ausgetauscht.

- Ständig belastete Pufferbatterien
  Auch wenn eine Spannungsversorgung, in der eine Pufferbatterie eingebaut ist, beispielsweise nur während der Revisionszeit benötigt wird, werden die vorhandenen Pufferbatterien zum Erhalt des RAM-Speichers dauerhaft belastet. Dies führt dazu, dass die Batterien häufiger ausfallen und dadurch ein entsprechender Austausch erforderlich ist. Um die Ausfälle zu verhindern, wurde ein jährliches Austauschintervall statt den typischen 2 Jahren für diese Pufferbatterien eingeführt.

Weiterhin hat sich gezeigt, dass Programmierfehler auftreten können. Diese Fehler sind meist schwer zu detektieren, da sie beispielsweise nur sporadisch oder bei bestimmten Betriebszuständen auftreten. In den in /GRS 15b/ betrachteten Fällen wurden vorgefundene Fehler durch ein Firmware-Update im Rahmen einer vorbeugenden Instandhaltung behoben. Dieses wird im Allgemeinen von der Herstellerfirma geliefert. Den Mitarbeitern der Anlage ist dabei meistens nicht bekannt, welche Details sich in der neuen Firmware geändert haben. Da für betriebliche Komponenten keine entsprechende Anforderung aus dem Regelwerk besteht, werden typischerweise hierzu auch keine weiteren Nachforschungen seitens des Betreibers veranlasst. Für sicherheitstechnisch wichtige Komponenten bestehen entsprechende Anforderungen, so dass hier eine Firmware nur mit einer genehmigten Versionsnummer aufgespielt werden darf.

Folgendes Fazit wurde in /GRS 15b/ aus der Auswertung von Ereignissen unterhalb der Meldeschwelle zu Fehlern bei Softwarekomponenten gezogen:

1. Einige der herkömmlichen Ausfallmechanismen und Fehlerursachen entfallen durch die programmierbare oder rechnerbasierte Technik, neue kommen jedoch hinzu.

2. Programmierungsfehler treten selten in Erscheinung, aber sie werden beobachtet.

3. Firmware-Updates werden von den Herstellerfirmen geliefert. Die Anlagen können beim Aufspielen der Updates durch Abgleichen der Versionsnummer die Firmware unterscheiden. Über den Inhalt der Updates, d. h. welche Fehler durch diese behoben werden, liegen den Anlagen üblicherweise keine Informationen vor.

### 4.1.6 Sonstige Ereignisse aus der GRS-Datenbank

In Tab. Tab. 4.1 werden weitere Ereignisse mit Softwarefehlern aus der GRS-Datenbank (GRS-DB) zusammengefasst.

**Tab. 4.1** Kurzbeschreibung weiterer Ereignisse mit Softwarefehlern aus der GRS-Datenbank

| Ereignis-Nr. | Ereignis | Ursache |
|---|---|---|
| **5** | Teilausfall der Brandmeldeunterzentrale im Maschinenhaus | Bei einer Fehlersuche an der Brandmeldeunterzentrale wurde eine Elektronikbaugruppe als Fehlerquelle angenommen und getauscht. Dabei musste der Datenspeicher auf die neu eingesetzte Baugruppe gewechselt werden. Dabei kam es zum Datenverlust, so dass die Baugruppe nicht mehr funktionsfähig war. |
| **6** | Befund bei der Einstellung der Weg-Not-Endabschaltung der Lademaschine bei Wiederkehrenden Prüfungen | Einstellwert für Weg-Not-Endabschaltung in der Steuerung entsprach nicht dem spezifizierten Wert. |
| **7** | Beschädigung eines Brennelementbügels im Lagerbecken mit der Lademaschine | Designfehler der Steuerungssoftware der BE-Wechselbühne (Keine Verriegelung zur Verhinderung horizontaler Bewegungen bei einer zu tiefen Hubhöhe) |
| **8** | Ausfall der betrieblichen Anzeigen der Leittechnik in der Warte während des Aufspielens einer neuen Datei bei gleichzeitiger Nichtverfügbarkeit eines Servers. | Ein Server war aufgrund von häufiger An- und Ab-schaltung der Stromversorgung aufgrund von Diesel-WKP ausgefallen. Da es keine Alarmmeldung auf der Warte gab, fiel der Ausfall dem Wartenpersonal nicht auf. Bei einer automatischen Neuinitialisierung des zweiten Servers fielen beide Server aus, so dass keine Anzeigen auf der Warte mehr möglich waren. |
| **9** | Nicht spezifikationsgerechtes Verhalten des Mittelbereichskanals | Fehlerhafte programmiertechnische Umsetzung des spezifizierten Algorithmus zur Berechnung des Signals RELFAEG führte wiederholt zu Summenstabeinwürfen. |
| **10** | Störung im Begrenzungssystem | Ausfall einer Signalverarbeitungseinheit in einem Leittechniksystem |
| **11** | Ausfall eines Messkanals in der Neutronenflussaußeninstrumentierung | Ansprechen der internen Überwachung einer Baugruppe eines softwarebasierten Leittechniksystems |

## 4.2 Ereignisse aus der internationalen Betriebserfahrung

### 4.2.1 Ereignisse aus der ICDE-GVA-Datenbank

Im Vorhaben "Systematische Aufbereitung der weltweiten Betriebserfahrung mit gemeinsam verursachten Ausfällen (GVA) im Rahmen einer internationalen Expertengruppe - ICDE (Internationales GVA-Datenaustauschprojekt der OECD/NEA)" werden meldepflichtige Ereignisse aus den teilnehmenden Ländern im Hinblick auf GVA-Potential erfasst, aufbereitet und in der ICDE-GVA-Datenbank archiviert. Die Ereignisse in der ICDE-GVA-Datenbank wurden im Rahmen dieses Vorhabens hinsichtlich Softwarefehler als Ursache untersucht. Im derzeitigen Stand der ICDE-GVA Datenbank liegen außer bereits bekannten Ereignissen mit Softwarefehlern aus der deutschen Betriebserfahrung noch keine weiteren Ereignisse mit Softwarefehlern vor.

### 4.2.2 Ereignisse aus der IAEA/IRS-Datenbank

In der IAEA/IRS-Datenbank wurden insgesamt 53 Ereignisse identifiziert, die einen Bezug zu softwarebasierter Leittechnik haben. Daraus wurden insgesamt 16 Ereignisse mit Softwarefehlern und Malware ermittelt. Diese Ereignisse sind in Tab. 4.2 zusammengefasst.

**Tab. 4.2** Software-relevante Ereignisse aus der IRS-Datenbank mit Malware in softwarebasierten Leittechniksystemen

| Ereignis-Nr. | Titel | Beschreibung des Softwarefehlers |
|---|---|---|
| **1** | Reduzierung eines Grenzwertes für die Schnellabschaltung | Durch einen zu hohen Grenzwert eines Anregekriteriums im Computer Code wurde eines von zwei unabhängigen Anregekriterien für die Schnellabschaltsysteme unwirksam.<br>Es kam zu Anlagenzuständen, in denen eine hohe Reaktorleistung das einzige Anregekriterium für das Abschaltsystem war. Der Einstellwert war auf Rat des Herstellers im Nachbarblock bereits reduziert worden. |
| **2** | Gemeinsam verursachte Ausfälle von Zwangsumwälzpumpen mit Schwungradgeneratoren | Hinweise auf mögliche GVA von Zwangsumwälzpumpen durch Umrüstung auf digital einstellbare Drehzahlregler. Als mögliche Verursacher werden Softwarefehler, Netzwerkprobleme und Stromausfall genannt, die zu unerwarteten Drehzahländerungen führen können. |
| **3** | Fehlerhafte Umsetzung eines digitalen Kontrollsystems | Bei der Umstellung vom analogen Managementsystem der Stabsteuerung auf ein softwarebasiertes System wurde entgegen den Anforderungen die Evaluierung von GVA durch Softwarefehler nicht berücksichtigt. |
| **4** | Fehler in der Spezifikationsbeschreibung von Ersatzteilen | Es wurden u. a. falsch gesetzte digitale Grenzwerte des Überstromschutzes, die aufgrund unzulänglicher Werktests und Betriebstest nicht erkannt wurden, festgestellt. |
| **5** | Unverfügbarkeit von HD-Einspeisesystemen durch Fehleinstellung der Durchsatzregler und Nichterkennung der Fehleinstellung durch ungeeignete Prüfungen | In Folge einer fehlerhaften Einstellung in der digitalen Speisewasserregelung fiel die Speisewasserversorgung aus, wodurch der Füllstand im RDB sank und es zur Reaktorschnellabschaltung (RESA) kam. |
| **6** | Schwankungen der elektrischen Generatorleistung | Für das Auftreten von Schwankungen in der Generatorleistung wird die verwendete Software als Ursache angesehen, die für das Anfahren des Generators ungeeignet war. Die Ursache des Ereignisses ließ sich jedoch nicht endgültig klären. |

| Ereignis-Nr. | Titel | Beschreibung des Softwarefehlers |
|---|---|---|
| **7** | Oszillation in der Reaktorleistung | Wegen einer fehlerhaft verzögerten Auswertung der Reaktivitätskoeffizienten, dem Kontrollsystem und des Software basierten Regelsystems kam es zu einer Oszillation der Reaktorleistung.<br>Ursache war u. a. eine in der Software falsch eingestellte Zykluszeit der Ein- und Ausgangssignale. |
| **8** | Auswirkungen Ethernet basierter, nicht sicherheitstechnisch wichtiger Leittechnik auf den sicheren Betrieb | Die auf einem Mikroprozessor basierende Regelung des Frequenzumrichters wurde durch große Datenmengen im Ethernet basierten Netzwerk gestört. Dadurch kam es zum Abschalten von zwei Zwangsumwälzpumpen und zum Auslösen einer manuellen RESA. Es stellte sich heraus, dass das Frequenzumrichter-Kontrollsystem bei Datenverkehr in der Größenordnung von 10-Mbit/s versagt. |
| **9** | Zweifel an der Zuverlässigkeit der zur automatischen Druckentlastung verwendeten Funktion | **Fehler 1:** Bei der DE-Füllstandsmessung enthielt die Logik zur Bildung des Signals „Füllstand tief“ einen Fehler, der dazu führen konnte, dass ein Minimum Level ohne Warnung unterschritten werden konnte.<br>**Fehler 2:** In der Software zur DE-Füllstandsmessung, die einer Leittechnikfunktion der Kategorie C ausführt, wurde die automatische Druckentlastung (Kategorie A) als Unterroutine integriert. Aufgrund der unterschiedlichen Kategorien hätten die beiden Programme funktional voneinander getrennt werden müssen, so dass eine gegenseitige Beeinflussung unmöglich ist. |
| **10** | Programmierfehler bei der Bestimmung von Wirkungsquerschnitten mit Programmcodes zur Kritikalitätsbestimmung | Ein Mitarbeiter entdeckte in Programmcodes einen Programmierfehler, der unter bestimmten Bedingungen zu einer nicht mehr konservativ ausgelegten Bestimmung der Kritikalität führte. |
| **11** | Start der Notspeisepumpen verursacht durch fehlerhafte Anregung des „Speisewasserverlust“-Signals beim Anfahren des Reaktors | Durch eine Modifikation des Algorithmus der zur Auswertung der Füllstands-Sensoren des Dampferzeugers verwendet wird, kam es zu einem fehlerhaften Auslösesignal „Speisewasserverlust“. Die fehlerhafte Änderung wurde auf allen Baugruppen durchgeführt und erste Fehleranzeichen nicht ausreichend beachtet. |

| Ereignis-Nr. | Titel | Beschreibung des Softwarefehlers |
|---|---|---|
| **12** | Sicherheitslücke in der Software von Gamma Monitoren | Aufgrund eines nicht durchgeführten Upgrades der Software auf dem Speicherbausteins (Erasable programmable read-only memory, EPROM) von Gamma Monitoren konnte durch Betätigen der Reset-Taste der Alarm bei erhöhter Strahlung und die damit verknüpften Verriegelungen unterdrückt werden. Statt der neuen befand sich noch eine alte Version der Software auf dem Speicherbaustein. |
| **13** | Fehlerhafter Algorithmus in einem Speichermodul einer Überlastschutzeinrichtung | In einer Überlastschutzeinrichtung wurde eine Modifikation der Software vorgenommen, um das fehlerhafte Auslösen der Überlastschutzeinrichtung zu beheben. Dabei wurde zwar der alte Fehler behoben, der neue Algorithmus war jedoch fehlerhaft, so dass es erneut zu einem vorzeitigen Auslösen der Überlastschutzeinrichtung kam. Bei den durchgeführten Integrationstests konnte der Fehler jedoch nicht entdeckt werden. Die modifizierte Software wurde in allen Überlastschutzeinrichtungen gleichzeitig eingesetzt und vergrößerte das Risiko eines GVA. |
| **14** | Erfahrung aus Audits bei 12 Betreibern hinsichtlich der Jahr 2000-Problematik (Y2K) | Erfahrungen mit der Vorgehensweise zur Y2K-Problematik. Wenn die Ziffernfolge „00“ im Datum als das Jahr 1900 - anstelle des gewünschten Jahres 2000 - interpretiert worden wäre, hätte es zu Datenfehlern in Rechenanlagen, Betriebsmitteln mit Mikroprozessoren (mit „embedded“ Software) und Fehlern in Softwareprogrammen kommen können. Es hätte zu Fehlfunktionen von Rechnern und zu Totalausfällen von Systemen kommen können. |
| **15** | Malware auf speicherprogrammierbaren Steuerungen unter SIMATIC WinCC und SIMATIC PCS7 | Durch Ausnutzung einer Sicherheitslücke im Microsoft-Betriebssystem wurde es möglich Malware gezielt auf SIMATIC WinCC und SIMATIC PCS7 Steuerungssysteme zu schleusen (siehe auch „Stuxnet“ Wurm). |
| **16** | Potentielle Sicherheitslücke im Computernetzwerk eines Kernkraftwerkes führte zu einer Computerwurm-Infektion | Durch eine Sicherheitslücke im Netzwerk Server konnte das Netzwerk mit einem Wurm infiziert werden, der einen Speicherüberlauf im Netzwerk verursachte und somit die Kommunikation der Computer verhinderte (siehe auch „Microsoft SQL Server“ Wurm) |

### 4.2.3 Ereignisse aus der COMPSIS-Datenbank

Das Vorhaben „Computer-Based Systems Important to Safety“ (COMPSIS) wurde von der OECD/NEA zur systematischen Erfassung und Auswertung der internationalen Betriebserfahrung mit softwarebasierten leittechnischen Einrichtungen initiiert. Im Rahmen des COMPSIS-Vorhabens wurden aus den teilnehmenden Ländern im Zeitraum 2005-2011 ca. 100 meldepflichtige Ereignisse im Bereich softwarebasierter Leittechnik erfasst und archiviert. Im Rahmen dieses Vorhabens wurden diese Ereignisse in Hinblick auf Softwarefehler als Ursache analysiert. Es wurden insgesamt 10 Ereignisse aus internationalen Anlagen zu Softwarefehlern identifiziert, die nicht bereits in der IRS-Datenbank enthalten waren. Die Ergebnisse dieser Auswertung sind in Tab. 4.3 zusammenfassend dargestellt.

**Tab. 4.3** Ereignisse mit Softwarefehlern aus der COMPSIS-Datenbank

| Ereignis Nr. | Titel | Beschreibung des Softwarefehlers |
|---|---|---|
| **1** | Neigung eines BE während BE-Wechsel aufgrund von Fehlern im Softwaredesign-Prozess und im Betrieb des Steuerungssystems der BE-Lademaschine- | Fehlerhafte Anwendung eines für Testzwecke vorgesehenen Betriebsmodus der BE-Lademaschine führte zur Neigung eines BE beim Herabsetzen während BE-Wechsel.<br>Spezieller Betriebsmodus wurde ohne Alarmfunktionen und ohne Zugriffskontrolle (Sperre für Normalbetrieb) in der Steuerungssoftware der BE-Lademaschine umgesetzt. Die Nutzung dieses speziellen Betriebsmodus wurde auch nicht administrativ geregelt. |
| **2** | Ausfall der Software der BE-Lademaschine | Die BE-Lademaschine startete bei Wiederinbetriebnahme nicht: Das Betriebssystem und das Anwendungsprogramm der betrieblichen Steuerungssoftware der BE-Lademaschine konnten nicht gestartet werden. |
| **3** | Abweichung der Borkonzentration beim Anfahren der Anlage | Berechnete und gemessene Borkonzentration beim Anfahren der Anlage nach BE-Wechsel wichen um mehr als den spezifizierten Wert voneinander ab.<br>Grund: Fehlerhaftes Isotopenmodell in der Software für Reaktorphysik. |

| Ereignis Nr. | Titel | Beschreibung des Softwarefehlers |
|---|---|---|
| **4** | Softwarefehler in der Steuerung von Ventilen im Kondensatreinigungssystem führt zu Reaktorschnellabschaltung | Fehlerhaftes Schließen von Ventilen im Kondensatreinigungssystem führt zum Verlust der Speisewasserversorgung und in der Folge zur Schnellabschaltung des Reaktors.<br>Beim Starten des softwarebasierten Leittechniksystems zur Ansteuerung von Ventilen im Kondensatreinigungssystem nach Tausch einer Prozessorkarte wurden die Speicherwerte für den Zustand von Ventilen im Kondensatreinigungssystem nach Einschalten der Spannungsversorgung automatisch auf "0" gesetzt. Dies entspricht nach Systemspezifikation einem Schließen der Ventile im Kondensatreinigungssystem. In der Folge wurde die Speisewasserversorgung des Reaktors unterbrochen. Es kam zu einer Schnellabschaltung des Reaktors.<br>Der gleiche Fehler wurde in der Dieselsteuerungssoftware des Blocks 2 und in dem Leittechniksystem des Kondensatreinigungssystems des Blocks 3 entdeckt. |
| **5** | Reaktorschnellabschaltung aufgrund eines Softwarefehlers in der Speisewasserfüllstands-Regelung | Prellen der F12 Taste bei einer Befehlseingabe führte zum ungewollten Absenken des Reaktorfüllstandes während eines Tests. Die Schichtmann-schaft übernahm die Füllstandsregelung von Hand. Es kam jedoch zu einer RESA, weil der Füllstand nicht gehalten werden konnte.Dieser Softwaredesign-fehler wurde während des Tests entdeckt |
| **6** | Reaktorschnellabschaltung (RESA) aufgrund eines Softwarefehlers in der Turbinenregelung | Bei einem Schaltvorgang wurde eine 120-V-Spannungsschiene fehlerhaft spannungslos, so dass ein Stromsignal der elektrohydraulischen Turbinenregelung anstand. Da durch einen unbemerkten Softwarefehler das zweite Signal ebenfalls bereits anstand, kam es daraufhin durch ein plötzliches Schließen der Turbinenregelventile zur RESA.<br>In der Software-Konfiguration der elektrohydraulischen Turbinenregelung war das Signal des zweiten Kanals fehlerhaft dauerhaft überbrückt, ohne Statusmeldung an die Schichtbesetzung. Dieses Signal stand damit dauerhaft an. |

| Ereignis Nr. | Titel | Beschreibung des Softwarefehlers |
|---|---|---|
| **7** | Ausfall der Kommunikation zwischen Wartenterminal und Feldgeräte des Luftüberwachungssystems | Im Leistungsbetrieb kam es plötzlich zum Ausfall der kontinuierlichen Kommunikation zwischen dem Wartenterminal und Feldgeräte des Lüftungsüberwachungssystems.<br>Ursache für den Ausfall waren Softwareprobleme und limitierte Hardwarekapazitäten des Wartenterminals. |
| **8** | Manuelle Auslösung der Reaktorschnellabschaltung (RESA) aufgrund eines Programmierfehlers in der Testsoftware des Speisewassersystems | Die Schichtbesetzung löste manuell RESA aus, kurz bevor beim Test einer Speisewasserpumpe ein Grenzwert zur Schutzabschaltung aufgrund von Überhitzung erreicht wurde.<br>Ursache war ein Programmierfehler in der Testsoftware des Speisewassersystems. Es war vorgesehen, dass bei einer Reduktion des Durchflusses für mehr als 15 Sek. automatisch die Drehzahl der Pumpe reduziert wird. Es war nicht erwartet worden, dass der Temperaturgrenzwert innerhalb dieser Zeit erreicht bzw. überschritten wird. Bei Änderungen u. einer Leistungserhöhung ein Jahr vor dem Ereignis war eine mögliche Temperaturerhöhung bei diesem Test nicht berücksichtigt worden. |
| **9** | Reaktorschnellabschaltung aufgrund eines Softwarefehlers in der Turbinenregelung | Die digitale elektrohydraulische Turbinenregelung wurde durch ein neues digitales System (Ovation von Westinghouse/ Emerson) ersetzt. In diesem neuen System war versehentlich ein Logikfehler in der Software enthalten, der unter spezifischen Umständen selbst-korrigierend war. Im vorliegenden Fall führte der Fehler zu einem raschen Abfall des Füllstandes in einem Dampferzeuger, wodurch die Reaktorschnellabschaltung ausgelöst wurde. Dieser Fehler wurde bei den Tests vor und nach der Installation in der Anlage nicht entdeckt, trat dann jedoch im Betrieb auf. |
| **10** | Ausfall eines Kanals zur Überwachung des DNB und der Leistungsdichte aufgrund eines Softwarefehlers | Die Anlage befand sich im Leistungsbetrieb, als ein Ausfall der Datenverbindung eines Kanals für die Überwachung des DNB und der lokalen Leistungsdichteverteilung auffiel.<br>Ursache war ein ausgefallenes Datenverbindungsmodem sowie eine fehlerhaft in der Software implementierte Prozedur zur Berechnung von Abweichungen zwischen den Kanälen. |

# 5 Auswertung und Klassifizierung der identifizierten Ereignisse

In diesem Kapitel werden die Ergebnisse der Auswertung und die Klassifizierung der im Rahmen dieses Vorhabens ermittelten Ereignisse aufgrund von Softwarefehlern in softwarebasierter Leittechnik (siehe Kapitel 3 und Kapitel 4) vorgestellt. Hierfür wurde zunächst ein Klassifizierungsschema, das sich an der Ursache des aufgetretenen Softwarefehlers orientiert, entwickelt. Dieses ursachenorientierte Klassifizierungsschema basiert auf der im Rahmen dieses Vorhabens erarbeiteten Definition von Softwarefehlern. Zudem wurde das im Rahmen des COMPSIS-Vorhabens angewandte Klassifizierungsschema analysiert. Es wurde insbesondere untersucht, inwieweit sich das COMPSIS-Klassifizierungsschema bzw. Klassifizierungskriterien aus dem COMPSIS-Vorhaben zur Auswertung und Klassifizierung der im Rahmen dieses Vorhabens identifizierten Ereignisse eignen. Aus beiden Klassifizierungsschemata wird dann das in diesem Vorhaben verwendete Klassifizierungsschema entwickelt.
Nachfolgend wird zunächst die im Rahmen dieses Vorhabens entwickelte Definition von Softwarefehlern vorgestellt. Anschließend werden das ursachenorientierte Klassifizierungsschema und das COMPSIS-Klassifizierungsschema beschrieben. Darauf aufbauend wird das im Vorhaben verwendete Klassifizierungsschema beschrieben und die Ergebnisse der Auswertung der ermittelten Ereignisse anhand ausgewählter Klassifizierungskriterien zusammenfassend dargestellt.

## 5.1 Vorgehensweise bei der Definition von Softwarefehlern im Rahmen des Vorhabens

### 5.1.1 Einleitung

Wie bereits in Kapitel 2.1.2 beschrieben, wird der Begriff Softwarefehler in der Literatur unterschiedlich und teils widersprüchlich definiert. In Tab. 5.1 sind die wesentlichen Aspekte der Definitionen noch mal zusammengefasst. Während in DIN EN 61513 /DIN 02/ und DIN EN 62138 /DIN 09/ der Softwarefehler als Auslegungs- oder systematischer Fehler definiert wird, der auch unentdeckt existiert, solange bestimmte Bedingungen nicht erfüllt werden, die den Softwarefehler sichtbar machen, handelt es sich in /IEE 10/, /IAE 99/ und /BMUB 12/ erst bei Auftreten einer Abweichung um einen Softwarefehler.

/LAW 93/ spricht von einem Softwarefehler, wenn das tatsächliche Verhalten vom spezifizierten Verhalten abweicht.

**Tab. 5.1** Definition des Softwarefehlers in verschiedenen Normen, Standards und Regelwerken

| Dokument | Definition „Softwarefehler“ |
|---|---|
| **DIN EN 61513 /DIN 02/** | Auslegungsfehler in einer Softwarekomponente |
| **DIN EN 62138 /DIN 09/** | Systematischer Fehler, der auf Auslegungsmängel zurückgeht und bei gleichen Bedingungen systematisch zu denselben Ausfällen führt.<br>Softwarefehler können in einem System unentdeckt bleiben. Erst wenn besondere Bedingungen auf dieses System einwirken, so dass die resultierende Funktion nicht mehr mit der vorgesehenen Funktion übereinstimmt, führen solche Fehler zum Versagen. |
| **ISO/IEC/IEEE 24765 /IEE 10/** | Auftreten einer Abweichung in der Software.<br>Unzulässiger Schritt, unzulässiger Prozess oder unzulässige Definition der Daten in einem Computerprogramm. |
| **IAEA TR Nr. 384 /IAE 99/** | Ergebnis einer Abweichung im Quellcode oder den Daten der Spezifikationen, die zu einem unvorschriftsmäßigen Verhalten der Software führt, wenn der fehlerhafte Abschnitt der Software oder die fehlerhaften Daten durchlaufen werden.<br>Softwarefehler führen nicht notwendigerweise zum Systemausfall, wenn die Umgebungsbedingungen des Systems den Fehler nicht zur Wirkung bringen. |
| **Sicherheitsanforderungen an Kernkraftwerke /BMUB 12/** | Fehler in einer Software, der bei bestimmten Kombinationen oder einer bestimmten Abfolge von Eingangsdaten nicht spezifizierte Ausgangsdaten erzeugt. |
| **USNRC /LAW 93/** | Abweichendes Verhalten eines Computersystems von seinen zugewiesenen Spezifikationen |

Aus diesem Grund wurde anhand der bestehenden Definitionen von Software, Softwarefehlern, der Erkenntnisse aus der Auswertung der Betriebserfahrung mit softwarebasierten Leittechniksystemen und insbesondere unter Betrachtung des System-Sicherheitslebenszyklus, (nachfolgend als System-Lebenszyklus bezeichnet) eines softwarebasierten Leittechniksystems eine Definition von Softwarefehlern entwickelt, welche die verschiedenen für Softwarefehler relevanten Aspekte berücksichtigt.

Nachfolgend wird zunächst der System-Lebenszyklus eines Leittechniksystems beschrieben. Darauf aufbauend wird die entwickelte Definition von Softwarefehlern im Rahmen dieses Vorhabens im Abschnitt 5.1.3 vorgestellt.

### 5.1.2 System-Lebenszyklus eines Leittechniksystems

Der System-Lebenszyklus eines Leittechniksystems umfasst gemäß /DIN 02/ die erforderlichen Aktivitäten in Zusammenhang mit der Realisierung eines sicherheitstechnisch wichtigen Leittechniksystems in einer Zeitspanne beginnend mit den Systemanforderungen in der Konzeptphase bis zu dem Zeitpunkt, in dem das Leittechniksystem nicht mehr für die Nutzung zur Verfügung steht. Die Phasen eines typischen System-Lebenszyklus umfassen /DIN 02/:

- die System-Anforderungsspezifikation,
- die Systemspezifikation,
- die detaillierte Auslegung und Realisierung des Systems,
- die Integration des Systems,
- die Validierung des Systems,
- den Errichtung des Systems,
- die Modifikationen des Systems (wenn zutreffend).

Das Ziel der Aktivitäten in den jeweiligen Phasen eines System-Lebenszyklus wird anhand des in /DIN 09/ dargestellten System-Sicherheitslebenszyklus eines softwarebasierten Leittechniksystems auf der Basis der Ausführungen in /DIN 02/ übersichtlich in der Tabelle 5.2 dargestellt.

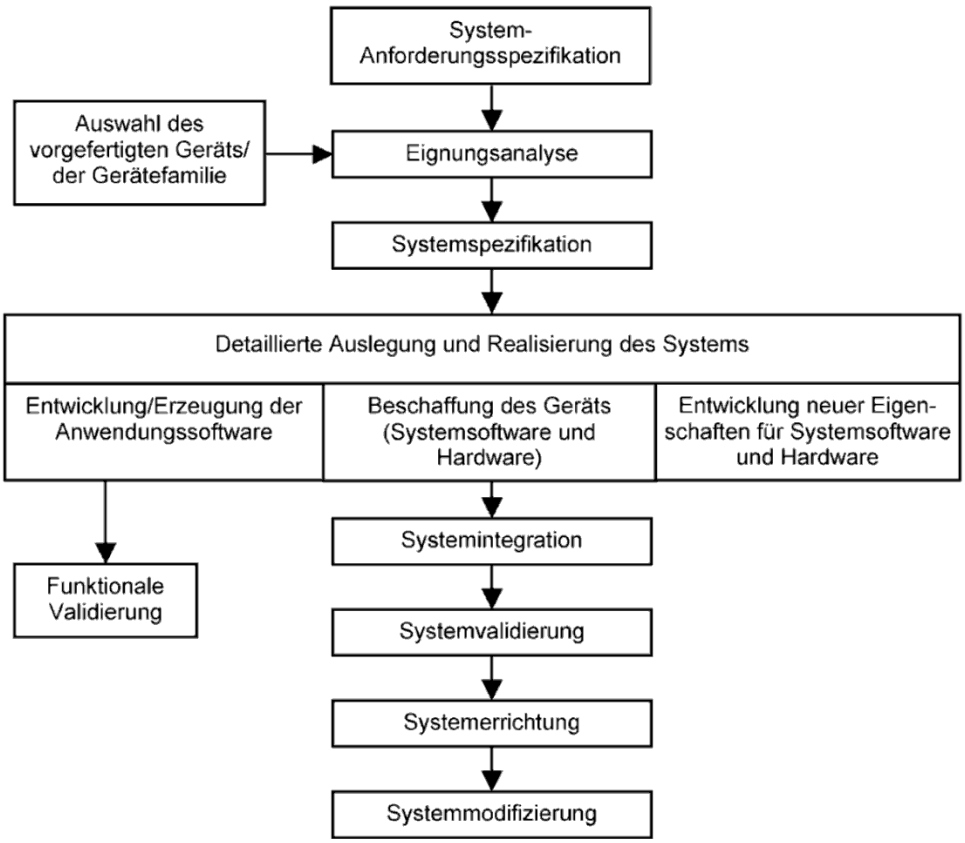



**Abb. 5.1** System-Lebenszyklus eines softwarebasierten Leittechniksystems /DIN 09/

**Tab. 5.2** Übersicht über den System-Lebenszyklus nach /DIN 02/

| **Phase** | **Ziel der Aktivitäten** |
|---|---|
| System-Anforderungsspezifikation | Beschreibung der Anforderungen an das System auf einem hohen Abstraktionsniveau, unabhängig von der Entscheidung für eine spezifische technische Lösung<br>Entwicklung der System-Anforderungsspezifikation für die Funktionen, für die Auslegungs-Randbedingungen, für die Grenzen und Schnittstellen mit anderen Systemen und Werkzeugen, für die Mensch-Maschine-Schnittstellen und für die Umgebungsbedingungen |
| Systemspezifikation | Entwicklung der Auslegung der Systemarchitektur (Hardware und Software) derart, dass die System-Anforderungsspezifikation erfüllt wird.<br>Bewertung und Beurteilung der Eignung von ggf. einzusetzendem bereits vorhandenem Gerät zur Integration in die Systemauslegung (Eignungsanalyse)<br>Zuordnung der Anwendungsfunktionen zu Teilsystemen |
| Detaillierte Systemauslegung und Realisierung | Erweiterung und Verfeinerung der Architekturauslegung<br>Entwicklung der Hardware und (System- oder Anwendungs-) Software<br>Validierung der Anforderungen an die Anwendungsfunktionen |
| Systemintegration | Zusammenbau der einzelnen Hardware- und Softwarekomponenten, die das System bilden |

| Phase | Ziel der Aktivitäten |
|---|---|
| Systemvalidierung | Validierung der Systemspezifikation |
| Systemerrichtung | Errichtung und Prüfung des Systems |
| Systemmodifizierung | Durchführung von Korrekturen, Verbesserungen oder Anpassungen des Systems |

Der System-Lebenszyklus ist ein iterativer Prozess; eine Phase darf beginnen, bevor die Tätigkeiten der vorausgehenden Phase abgeschlossen sind; eine Phase darf jedoch nur abgeschlossen werden, wenn die vorausgehenden Phasen zuvor abgeschlossen worden sind und ihre Ergebnisse konsistent mit ihren zur Verfügung gestellten Eingangsinformationen sind. /DIN 02/

Es ist anzumerken, dass gemäß /DIN 02/ und /DIN 09/ mit Ausnahme der System-Anforderungsspezifikation Software bezogene Tätigkeiten in allen Phasen des System-Lebenszyklus enthalten sind. In diesem Zusammenhang kommen den Phasen „Systemspezifikation“ und „Detaillierte Auslegung und Realisierung des Systems“ eine große und wichtige Bedeutung zu, denn die Software (System- und/oder Anwendungssoftware) wird in diesen beiden Phasen des System-Lebenszyklus spezifiziert und entwickelt bzw. erzeugt.

In der Software-Anforderungsspezifikation, die im Rahmen der Systemspezifikation entwickelt wird, müssen u. a. spezifiziert werden /DIN 09/, /DIN 60/, /DIN 02/:

- die von der Software zur Verfügung zu stellenden Anwendungsfunktionen
- die Schnittstellen und Wechselwirkungen der Software mit ihrer Umgebung (z. B. mit Operateuren, dem übrigen leittechnischen System, den anderen Systemen und Geräten, mit denen die Software in Wechselwirkung steht oder sich Ressourcen teilt), einschließlich der Rolle, Typen, Formate und Randbedingungen der Eingänge und Ausgänge;
- die Softwareparameter, die während des Betriebs von den Operateuren geändert werden können, ihre Rolle, Typen, Formate, Bereiche und Randbedingungen, und die per Software durchzuführenden Überprüfungen, wenn sie geändert werden
- was die Software nicht tun darf oder verhindern muss
- die für Software getroffenen Annahmen hinsichtlich der Umgebung

- die Verhaltensarten der Software, die gefordert werden, wenn Abweichungen oder Ausfälle entdeckt werden.

### 5.1.3 Definition von Softwarefehlern im Rahmen des Vorhabens

Auf der Basis der Ausführungen im Abschnitt 5.1.2 ergibt sich, dass Software als Produkt eines mehrphasigen Entwicklungsprozesses (Systemanforderungsspezifikation, Systemspezifikation, Systementwicklung, Systemintegration, Systemvalidierung, Systemerrichtung, Systemmodifikation) eines softwarebasierten Leittechniksystems zu betrachten ist. Bis auf die System-Anforderungsspezifikation-Phase enthalten die Phasen „Systemspezifikation“, „Systementwicklung“, „Systemintegration“, „Systemvalidierung“, „Systemerrichtung“, „Systemmodifikation“ dieses Entwicklungsprozesses Software bezogene Tätigkeiten, die nachfolgend als Software-Lebenszyklus zusammenfassend bezeichnet werden.

Fehler können in jeder beliebigen Phase des Software-Lebenszyklus eingebracht werden und/bzw. unentdeckt bleiben. Dazu zählen Fehler bei der Entwicklung der Software im Rahmen der Systementwicklung und/oder Systemmodifikation, fehlerhafte Systemspezifikation sowie unerkannte Fehler im Rahmen der Systemintegration, –validierung, –errichtung und –modifikation. Ein Softwarefehler ist daher ein latenter Fehler im Software-Lebenszyklus eines softwarebasierten Leittechniksystems, der zu einem nichtanforderungsgerechten Verhalten eines softwarebasierten Leittechniksystems führen kann. Hierbei ist zu unterscheiden zwischen:

- Programmierfehler: Fehler in der programmiertechnischen Umsetzung der Software-Anforderungsspezifikation bei angenommener anforderungsgerechter Spezifikation des softwarebasierten Leittechniksystems
- Spezifikationsfehler: Fehler in der Systemspezifikation (Hard- und/oder Software) des softwarebasierten Leittechniksystems bei angenommener spezifikationsgemäßer programmiertechnischer Umsetzung der Software-Anforderungsspezifikation des softwarebasierten Leittechniksystems

## 5.2 Ursachenorientierte Klassifizierung von Ereignissen mit Softwarefehlern in softwarebasierter Leittechnik

Dieses Klassifizierungsschema wurde im Rahmen dieses Vorhabens aufbauend auf der Definition von Softwarefehlern gemäß dem Abschnitt 5.1.3 entwickelt. Es orientiert sich an den Ursachen für die beobachteten Softwarefehler.

Bei Programmierfehlern und Spezifikationsfehlern (siehe Abschnitt 5.1.3) handelt es sich um Fehler die im Rahmen des Software-Lebenszyklus des softwarebasierten Leittechniksystems eingebracht wurden. Sie werden in Anlehnung an deren Ursachen als interne Softwarefehler bezeichnet. Sie bilden die erste Kategorie von Softwarefehlern im ursachenorientierten Klassifizierungsschema.

Für interne Softwarefehler wurden im Rahmen der Ereignisauswertung folgende typischen Ursachen identifiziert:

- Fehler in der Softwarespezifikation: z. B. fehlerhafte oder unvollständige Beschreibung der funktionalen Anforderungen an die Software.
- Fehler bei der Umsetzung der Softwarespezifikation: z. B. Programmierfehler, Logikfehler in der Auslegung.
- Fehler bei der Software-Integration und -validierung: z. B. unvollständige oder fehlerhafte Testprozeduren.
- Fehler bei Betrieb, Instandhaltung, Modifikation der Software: z. B. unsachgemäße Software-Upgrades.

Für eine umfassende Beschreibung von Softwarefehlern zwecks Auswertung und Klassifizierung der identifizierten Ereignisse im Rahmen dieses Vorhabens werden externe Softwarefehler als eine weitere Kategorie von Softwarefehlern definiert. Als externe Softwarefehler werden Softwarefehler bezeichnet, deren Ursachen nicht auf Fehler im Software-Lebenszyklus des softwarebasierten Leittechniksystems zurückzuführen sind. Dazu zählen beispielsweise Softwarefehler, die aufgrund von Fehlern oder Ausfällen in der Betriebsumgebung der Software auftreten. Auch Hardwarefehler, beispielweise an Prozessoren, Ein- und Ausgabe-Baugruppen oder A/D-Wandlern des betrachteten Leittechniksystems, die nicht auf eine fehlerhafte Systemspezifikation zurückzuführen sind, können zu derartigen externen Softwarefehlern führen.

Externe Softwarefehler können ebenfalls aus der Wechselwirkung des betrachteten Leittechniksystems mit seiner Betriebsumgebung oder mit anderen Leittechniksystemen resultieren. Diese Wechselwirkung geschieht beispielweise über Anlagenprozess- und/oder Mensch-Maschine-Schnittstellen des Leittechniksystems mittels Kommunikationspfaden oder Kommunikationsnetzwerken (siehe Abb. 5.2).

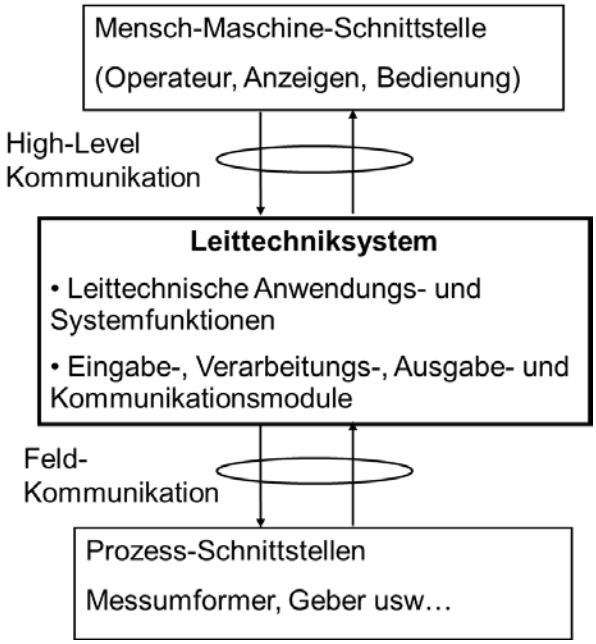



**Abb. 5.2** Modell eines Leittechniksystems mit Mensch-Maschinen- und Prozess-Schnittstellen /MBO 16/

Zu den möglichen Ursachen, die in diesem Zusammenhang zu Softwarefehlern führen können, zählen beispielsweise menschliche Fehlhandlungen an der Mensch-Maschinen-Schnittstelle, defekte Kabel in Kommunikationspfaden oder defekte Messumformer an der Prozess-Schnittstelle.

Kann ein beobachteter Softwarefehler nicht den internen oder externen Softwarefehlern zugeordnet werden, wird er als sonstiger Softwarefehler bezeichnet. Zu den sonstigen softwarerelevanten Fehlern zählen darüber hinaus Fehler aufgrund von Defiziten in der Systemanforderungsspezifikation bzw. in der Systemauslegung oder Fehler, die durch Malware verursacht wurden.

Fehler, die durch Defizite in der Systemanforderungsspezifikation bzw. in der Systemauslegung des Leittechniksystems verursacht werden, sind als unabhängig von der Realisierung der vorgesehenen Leittechnikfunktionen (softwarebasiert oder festverdrahtet) zu betrachten, denn die Beschreibung der Anforderungen an das System im Rahmen der Systemanforderungsspezifikation erfolgt gemäß /DIN 02/ auf einem hohen Abstraktionsniveau, unabhängig von der Entscheidung für eine spezifische technische Lösung.

Aus den vorangegangenen Ausführungen ergibt sich das in Abb. 5.3 dargestellte ursachenorientiere Klassifizierungsschema von Ereignissen mit Softwarefehlern in softwarebasierter Leittechnik.

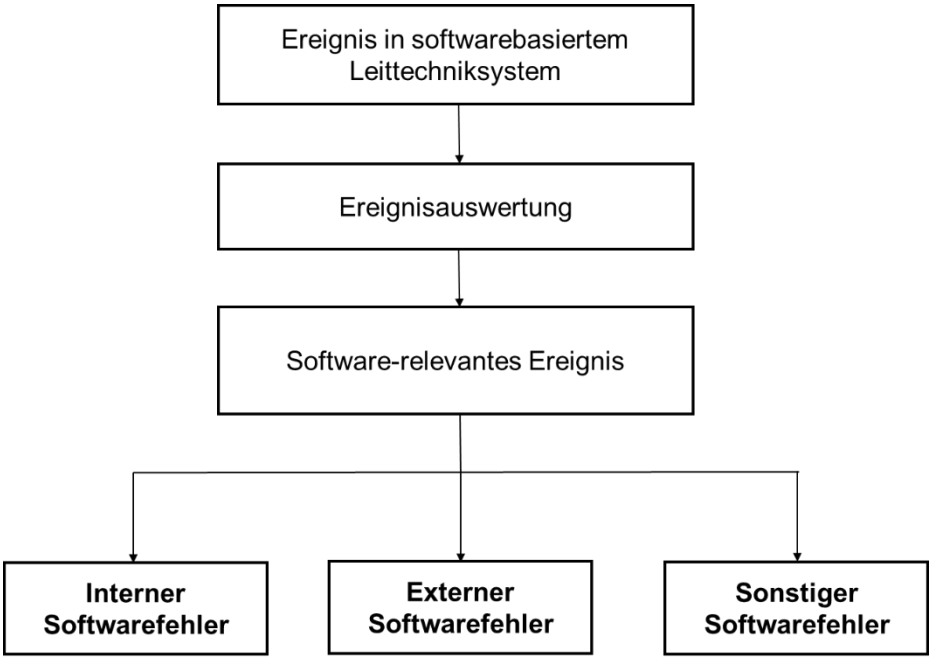



**Abb. 5.3** Ursachenorientierte Klassifizierung von Ereignissen mit Softwarefehlern in softwarebasierter Leittechnik /MBO 16/

## 5.3 Klassifizierung von Ereignissen in der COMPSIS-Datenbank

Die Richtlinien für die Codierung der Ereignisse computerbasierter Leittechniksysteme in der COMPSIS-Datenbank, die auf den Standards der IAEA und der IEC (International Electrotechnical Commission) basieren, beinhalten auch eine Klassifizierung der Softwarefehler. Die Klassifizierung berücksichtigt folgende Aspekte /COM 11/:

- Klassifizierung des computerbasierten Systems und seinen Funktionen
- Klassifizierung des Aufbaus des computerbasierten Systems
- Klassifizierung des Anlagenzustands und der Auswirkungen auf den Anlagenzustand
- Klassifizierung der Fehlereigenschaften
- Klassifizierung der Schwere des Fehlers

Im Folgenden wird die Vorgehensweise bei der Klassifizierung der einzelnen Aspekte ausführlich beschrieben.

### 5.3.1 Klassifizierung des computerbasierten Systems und seiner Funktionen

#### 5.3.1.1 Sicherheitsrelevanz des Systems bzw. der Funktion

Die Klassifizierung des Systems bzw. der Funktion hinsichtlich ihrer Sicherheitsrelevanz erfolgt nach den Standards der IAEA NS-G-1.3 /IAE 02/ (siehe Abb. 2.2), des IEEE 603 /IEE 09/ und der DIN IEC 61226 /DIN 10/ (siehe Tab. 2.1). Dabei sollte das Ereignis für die COMPSIS-Datenbank nach (mindestens) einem der drei Standards klassifiziert werden /COM 11/.

#### 5.3.1.2 Identifikation des Systems bzw. der Funktion

In einem weiteren Schritt wird das betroffene computerbasierte System nach dessen Funktion klassifiziert. Dabei werden die computerbasierten Systeme in folgende Klassen unterteilt:

- Reaktorschutzsystem
  - Reaktorschnellabschaltsystem
  - Auslösesystem für aktive Sicherheitseinrichtung
- Sicherheitstechnisch wichtige Verriegelung
- Rechnergestützte Betriebs- und Überwachungssysteme
  - Überwachungssystem
  - Alarmsystem
  - Diagnosesystem
  - Optimierungssystem
  - Regelungssystem
- Informationssystem
- Begrenzungssystem
- Risikominderung
- Mensch-Maschine-Schnittstelle
  - Leitwarte
  - Nebenleitstände
  - Notfall-Reaktionseinrichtung
  - Steuereinrichtungen
  - Anzeigeeinrichtungen
  - Störfallinstrumentierung
  - Gefahrenmeldeanlage
  - Datenarchivierung

Die Datenbank ermöglicht darüber hinaus noch eine detailliertere Klassifizierung der betroffenen Sicherheitsfunktion. Dabei werden die rechnerbasierten Systeme unter anderem danach unterschieden, ob ihre Sicherheitsfunktionen im direkten Zusammenhang mit der Funktion der Anlage stehen (z. B. Reaktivitätskontrolle, Wärmeabfuhr, Integrität des Sicherheitsbehälters und der Barrieren, etc.) oder andere Sicherheitsfunktionen betroffen sind /COM 11/.

### 5.3.2 Klassifizierung des Aufbaus des computerbasierten Systems

#### 5.3.2.1 Identifikation der betroffenen Funktionsebenen des Systems

Nachdem das computerbasierte System und die Funktion, in der der Fehler aufgetreten ist, klassifiziert wurden, wird der Aufbau des Systems genauer analysiert. Es werden die fehlerhaften Funktionsebenen (siehe Kapitel 2.2.1) des Leittechniksystems identifiziert. Dabei wird unterschieden zwischen Fehlern auf der Eingabe- bzw. Ausgabeebene (process layer /COM 11/), auf der Verarbeitungsebene (application layer /COM 11/) und auf der Kommunikationsebene (communication layer /COM 11/).

#### 5.3.2.2 Identifikation des fehlerhaften Elements

Für das betroffene fehlerhafte Element, wird nach den folgenden Klassen unterschieden, wobei noch weitere Unterkategorien definiert sind, die hier nicht im Einzelnen aufgezählt werden:

- Computer Hardware
- Computer Software
- Firmware
- Daten
- Dokumentation

### 5.3.3 Klassifizierung des Anlagenzustands und der Auswirkungen auf den Anlagenzustand

Für die COMPSIS-Datenbank werden der Anlagenzustand (Kritikalität, Revision, Stillstand, etc.) und der Zustand des rechnerbasierten Systems (in Betrieb, in Standby, Wartung/Änderung, Test, Inbetriebnahme) zum Zeitpunkt des Ereignisses angegeben. Zudem werden die Auswirkungen des Ereignisses auf den Anlagenzustand beschrieben (keine Auswirkungen, Abfahren, Teilabfahren, Reaktorschnellabschaltung, etc.).

### 5.3.4 Klassifizierung der Fehlereigenschaften

In einem nächsten Schritt, werden die Fehler nach ihren Eigenschaften klassifiziert. In einer übergeordneten Einteilung der Fehler werden tatsächliche bzw. potentielle Auswirkungen des Fehlers auf die Anlage u. a. im Hinblick Wirksamkeit von Sicherheitsfunktionen, Sicherstellung der Stromversorgung, Nachwärmeabfuhr und Kontrolle der Reaktivität beschrieben.

In einer weiteren Einteilung wird bezüglich der Ursache des Fehlers nach Hardware- und Softwarefehlern unterschieden. Hardware- und Softwarefehler werden gemäß den nachfolgenden Unterkategorien weiter charakterisiert:

- Hardwarefehler/-versagen
  - Systematischer Fehler
  - Nicht-systematischer Fehler
- Softwarefehler/-versagen
  - Dokumentation (Kommentare, Meldungen)
  - Syntax (Schreibweise, Tippfehler, falsche/fehlende Zeichen, Befehlsformate)
  - Erstellungsprozess, Pakete (Change-Management, Bibliothek, Versionsverwaltung)
  - Zuweisung (Deklaration, doppelte Namensvergabe, Sichtbarkeitsbereich von Variablen (scope),
  - Schnittstelle (Prozeduraufruf- und Referenz, I/O, Benutzereinstellung)
  - Fehlererkennung (Fehlermeldungen, mangelhafte Fehlererkennung)

- Daten (Struktur, Inhalt)
- Funktionen (Logik, Zeiger, Schleife, Rekursion, Rechnung, Mängel in der Funktion)
- System (Konfiguration, Timing, Speicher)
- Umgebung (Design, Kompilieren, Testen, andere Probleme in der Betriebsunterstützung)

In einem weiteren Klassifizierungskriterium wird die beobachtete Fehlerart in der betroffenen Einrichtung beschrieben.

Es wird zwischen den nachfolgenden Fehlerarten der betroffenen Einrichtung unterschieden:

- Transienter Fehler
- Intermittierender Fehler
- Dauerhafter Fehler

Es wird ebenfalls die Kategorie der Abhängigkeit des bzw. der Fehler definiert:

- Einzelfehler/-abweichung/-versagen
- Mehrfachfehler/-abweichung/-versagen
  - Unabhängig
  - Abhängig
- Systematischer Fehler
- Gemeinsam verursachter Ausfall (GVA)

### 5.3.5 Klassifizierung der Schwere des Fehlers

Die Klassifizierung der Schwere des Fehlers ist in der COMPSIS-Datenbank nicht obligatorisch. Die Schwere des Fehlers wird nach drei Kriterien charakterisiert: Auswirkungen auf die Menschen, die Anlage und die Umwelt. Für die drei Kriterien werden fünf Klassen von Fehlerauswirkungen definiert, die für jedes Kriterium genau beschrieben sind. Sie sind in Tab. 5.2 dargestellt.

**Tab. 5.3** Charakterisierung der Fehlerauswirkungen in der COMPSIS-Datenbank

| **Klassifizierung der Auswirkungen auf** | | | |
|---|---|---|---|
| | **Menschen** | **Anlage** | **Umwelt** |
| **katastrophal** | Todesopfer | Verlust des Systems, das nicht mehr reparierbar ist und ersetzt werden muss | Schwerwiegende Umweltschäden |
| **kritisch** | ernsthafte Verletzung/Erkrankung, mit medizinischer Versorgung (lange Genesung, permanente Beeinträchtigung) | Großer Verlust des Systems, Aufgabe kann nicht durchgeführt werden | Große Umweltschäden |
| **gering** | Geringe Verletzung/Erkrankung mit medizinischer Versorgung ohne permanente Beeinträchtigung | Aufgaben von geringerer Wichtigkeit können nicht durchgeführt werden | Geringfügige Umweltschäden |
| **vernachlässigbar** | Kaum Verletzung/Erkrankung, kaum/keine Erstbehandlung | System weniger als einen Tag nicht funktionsfähig | Sehr geringe Umweltschäden |
| **keine** | | | |

## 5.4 Auswertung und Klassifizierung von Ereignissen im Rahmen dieses Vorhabens

In Kapitel 5.3 wurde die Klassifizierung von Ereignissen mit Softwarefehlern am Beispiel der COMPSIS-Datenbank beschrieben. Diese Klassifizierung ist sehr detailliert und bedarf vieler Informationen, die oftmals bei Ereignissen nicht in dem notwendigen Detaillierungsgrad zur Verfügung stehen. Im Rahmen dieses Vorhabens soll in erster Linie untersucht werden, inwieweit sich Ereignisse mit Softwarefehlern im Hinblick auf ihre Ursache entsprechend der im Kapitel 5.1.3 formulierten Definition von Softwarefehlern und dem im Kapitel 5.2 beschriebenem ursachenorientierten Klassifizierungsschema unterscheiden. Daher wird bei dem Klassifizierungsschema für die ermittelten Ereignisse der Schwerpunkt auf die Ursache der Ereignisse gelegt und die Ereignisse entsprechend der Ursache gemäß Kapitel 5.2 nach internem (I), externem (E) oder sonstigem software-relevanten Fehler (S) unterschieden.

Das ursachenorientierte Klassifizierungsschema wird für die Betrachtung um das Klassifizierungskriterium “betroffenes System“ erweitert. Es wird zudem in Anlehnung auf das COMPSIS-Klassifizierungsschema dargestellt, ob der Softwarefehler eine tatsächliche Auswirkung auf das System hatte oder er lediglich potentiell (pot.) zu einer Auswirkung hätte führen können und bis zur Entdeckung ohne Auswirkung blieb. Weiterhin wird zusätzlich angegeben, ob es sich bei dem Ereignis um ein Ereignis mit Potenzial zum gemeinsam verursachten Ausfall (GVA) handelt. Anhand dieser Kriterien ergibt sich das erweiterte ursachenorientierte Klassifizierungsschema „COCS“ der GRS für Software relevante Ereignisse mit den Klassifizierungskriterien „Ursache“, „betroffenes System“, „Auswirkung“ und „GVA-Potential“.

In Tab. 5.3 wird die Klassifizierung der ermittelten Ereignisse außerhalb der Kerntechnik (siehe Kap. 3) nach dem COCS-Klassifizierungsschema dargestellt, wobei das GVA-Potenzial der Ereignisse nicht berücksichtigt wurde. In zwei Fällen handelte es sich um einen internen Softwarefehler, der durch einen Programmierfehler entstanden ist. In den beiden anderen Fällen handelte es sich um einen externen Fehler, der durch die Hardware der Betriebsumgebung verursacht wurde. In allen Fällen hatte der Softwarefehler eine tatsächliche Auswirkung auf die betroffenen Systeme.

**Tab. 5.4** Klassifizierung der ermittelten Ereignisse außerhalb der Kerntechnik nach dem COCS-Klassifizierungsschema

| Ereignis | Klassifizierung der Ursache | betroffenes System | Auswirkung |
|---|---|---|---|
| Absturz der Ariane 5 | I: Programmierfehler | Flugkontrollsystem | ✓ |
| Unfälle in der Strahlentherapie | I: Programmierfehler | Steuerungssystem des Bestrahlungsgerätes | ✓* |
| Flug des Airbus A321 von Bilbao nach München | E: Hardware der Betriebsumgebung | Flugzeugsteuerungssystem | ✓ |
| Absturz des Airbus A320-216 von Indonesien nach Singapur | E: Hardware der Betriebsumgebung | Flugzeugsteuerungssystem | ✓* |

✓*: Auswirkungen auf Menschen (Todesopfer)
I: interner Softwarefehler, E: externer Softwarefehler

Die in Kapitel 3 beschriebenen Malware-Ereignisse werden gemäß Kapitel 5.2 den Ereignissen mit sonstigem software-relevanten Fehler (S) zugeordnet.

In den Tabellen  Tab. 5.3, Tab. 5.4 und Tab. 5.5 ist die Klassifizierung der nationalen und internationalen Ereignisse in kerntechnischen Anlagen (siehe Kapitel 4) nach dem COCS-Klassifizierungsschema dargestellt.

Die Auswertung der Ereignisse ergibt, dass in deutschen kerntechnischen Anlagen folgende Systeme von Softwarefehlern betroffen waren:

- Neutronenflussinstrumentierung
- BE-Wechselbühne/Lademaschine
- Kransteuerung
- Turbinenregelung
- Leistungsverteilungsüberwachung, DNB/Leistungsdichte-Berechnung
- Brandmeldeunterzentrale
- Wartenanzeige

**Tab. 5.5** Klassifizierung der Ereignisse in kerntechnischen Anlagen (GRS-Datenbank) nach dem COCS-Klassifizierungsschema

| Ereignis Nr. | Klassifizierung der Ursache | Betr. System | GVA | Auswirkung |
|---|---|---|---|---|
| **1** | S: Fehler der Systemspezifikation | Neutronenflussinstrumentierung | ✓ | ✓ |
| **2a** | I: Programmierfehler | BE-Lademaschine | ✓ | ✓ |
| **2b** | E: Hardware der Betriebsumgebung<br>*und*<br>I: Fehler in der Softwarespezifikation<br>*und*<br>I: Fehler bei Betrieb, Instandhaltung, Modifikation | BE-Wechselbühne | - | pot. |
| **3** | E: Hardware der Betriebsumgebung | Turbinenregelung | ✓ | ✓ |

| Ereignis Nr. | Klassifizierung der Ursache | Betr. System | GVA | Auswir-kung |
|---|---|---|---|---|
| **4** | S: Fehler der Systemspezifikation<br>*und*<br>E: Hardware der Betriebsumgebung<br>*und*<br>I: Programmierfehler | Kransteuerung | ✓ | ✓ |
| **5** | E: Hardware der Betriebsumgebung | Brandmeldeunterzentrale | - | pot. |
| **6** | I: Programmierfehler | BE-Lademaschine | - | pot. |
| **7** | S: Fehler der Systemspezifikation | BE-Lademaschine | ✓ | ✓ |
| **8** | S: Fehler der Systemspezifikation | Wartenanzeige | ✓ | ✓ |
| **9** | I: Programmierfehler | Neutronenflussinstrumentierung | ✓ | ✓ |
| **10** | E: Hardware der Betriebsumgebung | Begrenzungseinrichtung | ✓ | ✓ |
| **11** | S: unbekannt | Neutronenflussinstrumentierung | ✓ | ✓ |

I: Interner Softwarefehler, E: externer Softwarefehler, S: sonstiger Softwarefehler

**Tab. 5.6** Klassifizierung der IRS-Ereignisse nach dem COCS-Klassifizierungsschema

| Ereignis Nr. | Klassifizierung der Ursache | Betr. System | GVA | Auswir-kung |
|---|---|---|---|---|
| **1** | I: Programmierfehler *oder*<br>S: Fehler der Systemspezifikation | Schnellabschaltsystem | - | pot. |
| **2** | I: Programmierfehler *oder*<br>E: Hardware der Betriebsumgebung | Zwangsumwälzpumpen | - | pot. |
| **3** | S: Fehler der Systemspezifikation | Steuerstabregelung | - | pot. |
| **4** | I: Programmierfehler *oder*<br>S: Fehler der Systemspezifikation | Überstromschutz | - | pot. |

| Ereignis Nr. | Klassifizierung der Ursache | Betr. System | GVA | Auswirkung |
|---|---|---|---|---|
| **5** | I: Spezifikationsfehler der Software<br>*oder*<br>I: Programmierfehler<br>*oder*<br>S: Fehler der Systemspezifikation | Speisewasserversorgung | ✓ | ✓ |
| **6** | S: Unbekannt | Turbinenregelung | ✓ | ✓ |
| **7** | I: Programmierfehler *oder*<br>S: Fehler der Systemspezifikation | Reaktorleistungsregelung | ✓ | ✓ |
| **8** | E: Hardware der Betriebsumgebung | Netzwerk, Zwangsumwälzpumpen | ✓ | ✓ |
| **9** | I: Programmierfehler *und*<br>S: Fehler der Systemspezifikation | FD-System, Speisewassersystem, DE-Füllstandsmessung | - | pot. |
| **10** | I: Programmierfehler | Neutronenflussinstrumentierung | - | pot. |
| **11** | I: Programmierfehler | DE-Füllstand, Notspeisepumpen | ✓ | ✓ |
| **12** | I: Fehler bei Betrieb, Instandhaltung, Modifikation | Aktivitätsmessstelle | - | pot. |
| **13** | I: Fehler bei Betrieb, Instandhaltung, Modifikation<br>*und*<br>I: Unvollständige Testprozeduren | Überstromschutz, Transformator | ✓ | ✓ |
| **14** | S: Fehler der Systemspezifikation | Leittechnik allgem. | - | ✓ |
| **15** | S: Malware | Prozesssteuerung | | pot. |
| **16** | S: Malware | Prozesssteuerung | ✓ | ✓ |

I: Interner Softwarefehler, E: externer Softwarefehler, S: sonstiger Softwarefehler

**Tab. 5.7** Klassifizierung der COMPSIS-Ereignisse nach dem COCS-Klassifizierungsschema

| Ereignis | Klassifizierung der Ursache | betroffenes System | GVA | Auswirkung |
|---|---|---|---|---|
| **1** | I: Programmierfehler | Steuerung der BE-Lademaschine | ✓ | ✓ |
| **2** | S: Unbekannt | Steuerung der BE-Lademaschine | - | pot. |
| **3** | S: Fehler der Systemspezifikation | Softwaretool für Reaktorphysik | ✓ | ✓ |
| **4** | I: Programmierfehler<br>*oder*<br>S: Fehler der Systemspezifikation | Kondensat-reinigungssystem | ✓ | ✓ |
| **5** | I: fehlerhafte Testprozedur | Leittechniksystem des Speisewassersystems | ✓ | ✓ |
| **6** | I: Programmierfehler<br>*oder*<br>I: Fehler der Software-Spezifikation | Turbinenregelung | ✓ | ✓ |
| **7** | I: fehlerhafte Testprozedur<br>*und*<br>S: Fehler der Systemspezifikation | Lüftungsüber-wachung | - | pot. |
| **8** | I: Fehler in Betrieb, Instandhaltung, Modifikation | Leittechniksystem des Speisewassersystems | - | ✓ |
| **9** | I: Programmierfehler | Turbinenregelung | ✓ | ✓ |
| **10** | E: Hardware der Betriebsumgebung | Softwaretool für Reaktorphysik | - | pot. |

I: Interner Softwarefehler, E: externer Softwarefehler, S: sonstiger Softwarefehler

In internationalen Anlagen waren darüber hinaus die folgenden Systeme von Softwarefehlern betroffen:

- Reaktorphysik-Berechnungen
- Kondensatfiltersystem
- Lüftungssystem der Warte
- Speisewasserregelung
- FD-System
- Schnellabschaltsystem, Steuerstabregelung
- Reaktorleistungsregelung
- Überlast-/Überstromschutz

In Abb. 5.4 ist dargestellt, in wie vielen Fällen es national und international zu einer tatsächlichen Auswirkung von Ereignissen mit Softwarefehlern gekommen ist. Von insgesamt 38 Ereignissen traten 12 Ereignisse in Deutschland auf. Bei 23 von 38 Ereignissen hatte der Softwarefehler eine tatsächliche Auswirkung auf das jeweilige System. Neun dieser Ereignisse traten in Deutschland auf.

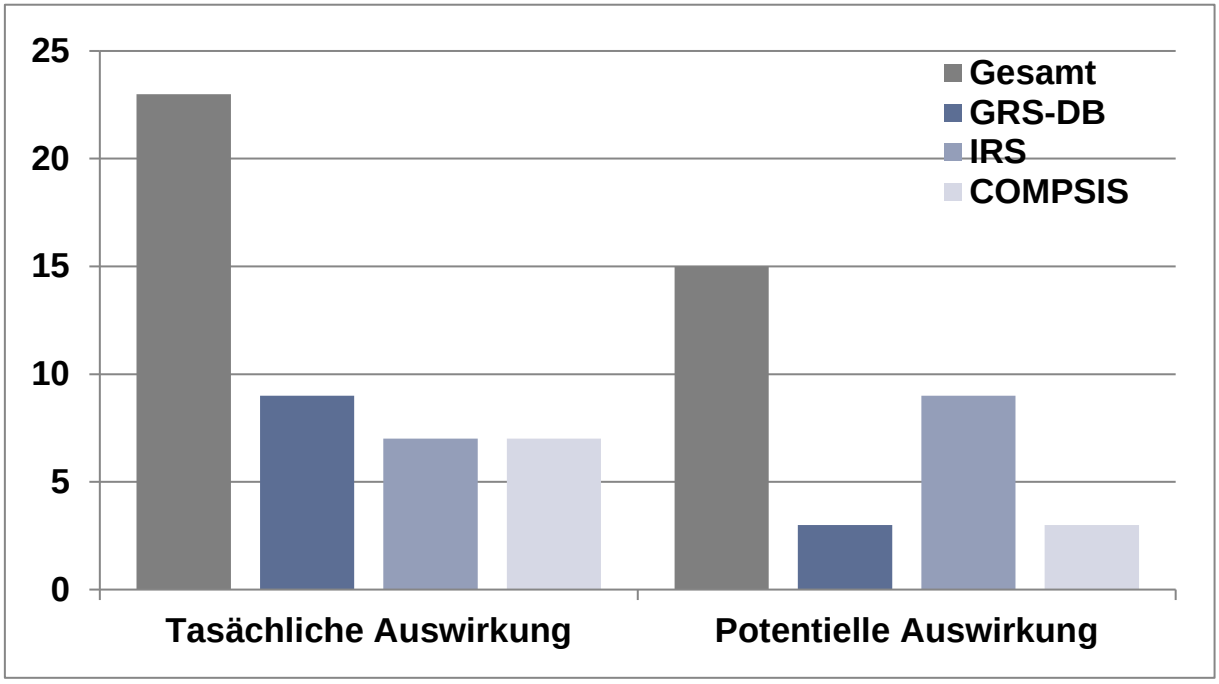



**Abb. 5.4** Auswertung der nationalen und internationalen Ereignisse mit Softwarefehlern in Hinblick auf ihre tatsächliche oder potentielle Auswirkung

In Abb. 5.5 ist die Anzahl an internen, externen und sonstigen Softwarefehlern dargestellt. Da manche Ereignisse mehrere Ursachen haben, wurden den 38 Ereignissen insgesamt 53 Ursachen zugeordnet. Davon handelt es sich um 27 interne Softwarefehler, 8 externe Softwarefehler und 18 sonstige Software-relevante Fehler. Bei den Ursachen handelt es sich zu 50,9 % um interne, 15,1 % um externe und 34,0 % um sonstige Fehler. Bei sonstigen Fehlern handelt es sich um 2 Malware-Ereignisse und 3 Ereignisse, bei denen die genaue Ursache unbekannt ist.

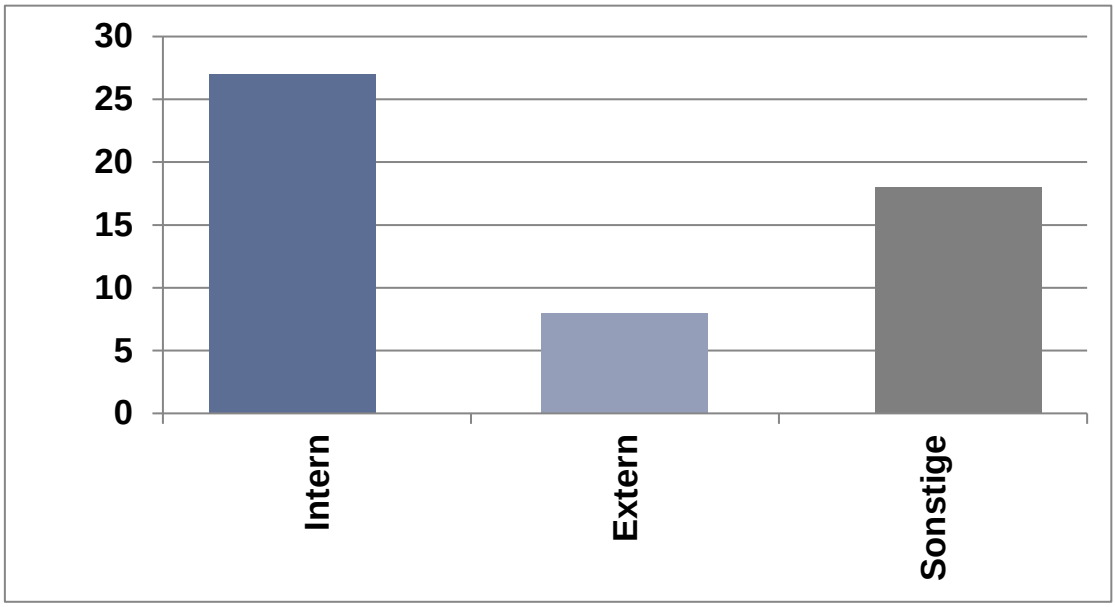



**Abb. 5.5** Auswertung der nationalen und internationalen Ereignisse mit Softwarefehlern in Hinblick auf die Ursachenart (Intern, Extern, Sonstige)

In Abb. 5.6 sind die Ursachen detailliert aufgegliedert. Hauptursachen in den betrachteten Ereignissen mit Softwarefehlern waren demnach Programmierfehler (intern) und Fehler in der System-Spezifikation (extern).

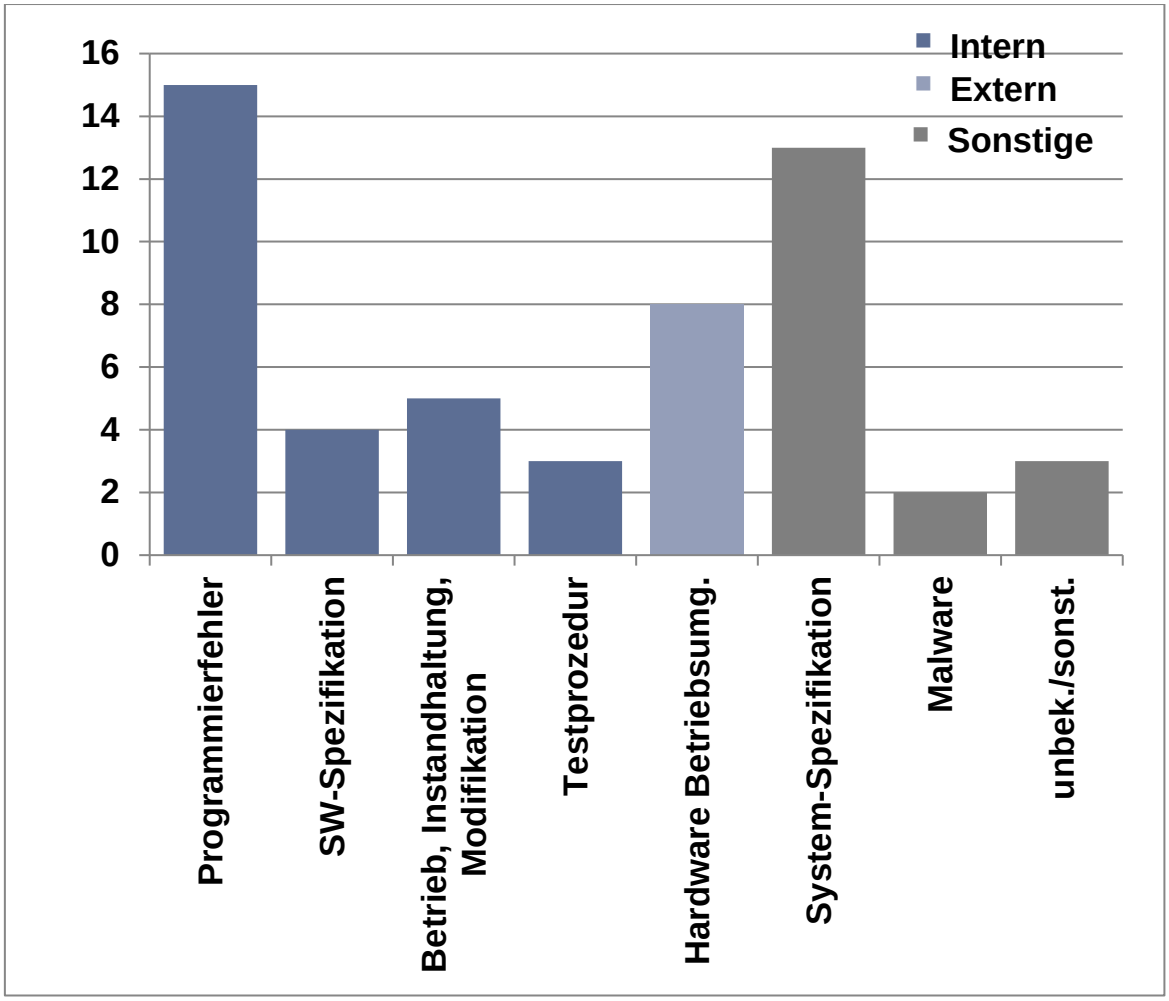



**Abb. 5.6** Auswertung der nationalen und internationalen Ereignisse mit Softwarefehlern in Hinblick auf ihre Ursache

# 6 Untersuchungen zu den Auswirkungen von postulierten Softwarefehlern in sicherheitstechnisch wichtigen Einrichtungen in Kernkraftwerken

## 6.1 Einleitung

Um die Wissensbasis der GRS über Auswirkungen von Softwarefehlern zu erweitern, wurden im Rahmen dieses Arbeitsabschnittes potentielle Auswirkungen von Softwarefehlern in softwarebasierter Leittechnik auf die Funktion sicherheitstechnisch wichtiger Einrichtungen untersucht.

Dazu wurden zunächst zu untersuchende sicherheitstechnisch wichtige leittechnische Einrichtungen ausgewählt. Als Kriterien wurden hierfür u. a. die gewonnenen Erkenntnisse aus der Auswertung der Betriebserfahrung zu Ereignissen mit Softwarefehlern (siehe Kapitel 3, 4 und 5) sowie die sicherheitstechnische Bedeutung der Einrichtungen herangezogen. Anschließend wurden aufbauend auf den bei der Auswertung der Betriebserfahrung identifizierten Softwarefehlerarten (siehe Kapitel 3, 4 und 5) potentielle Auswirkungen von einzelnen Softwarefehlerarten auf Leittechnikfunktionen in den ausgewählten sicherheitstechnisch wichtigen leittechnischen Einrichtungen und auf die damit verknüpfte Verfahrenstechnik untersucht.

## 6.2 Untersuchungskonzepte

Um die Auswirkungen potentieller Softwarefehler in sicherheitstechnisch wichtigen softwarebasierten leittechnischen Einrichtungen zu untersuchen, wurden im Rahmen dieses Arbeitsabschnittes verschiedene Konzepte erarbeitet. Die Untersuchungskonzepte unterscheiden sich im Wesentlichen in der erzielbaren Detaillierungstiefe bei der Modellierung der Softwarefehler, der betrachteten Leittechnikfunktionen bzw. Leittechniksysteme und deren Verknüpfung mit der angesteuerten Verfahrenstechnik. Dies hängt wiederum u. a. von den verfügbaren Informationen über die betrachteten Systeme ab.

Folgende Konzepte wurden für die Untersuchung potentieller Auswirkungen von Softwarefehlern entwickelt:

- Untersuchung der potentiellen Auswirkungen von Softwarefehlern mit einem Analysesimulator: In einem Analysesimulator sind die Verknüpfungen zwischen den Leittechnikfunktionen bzw. Leittechniksystemen und der angesteuerten Verfahrenstechnik einer Anlage nachgebildet. Zur Untersuchung der Auswirkungen von Softwarefehlern werden die zu betrachtenden Leittechnikfunktionen entsprechend den Softwarefehlerarten „modifiziert“ und die Auswirkungen dieser Softwarefehler auf die Verfahrenstechnik analysiert.

- Untersuchung der potentiellen Auswirkungen von Softwarefehlern auf der Basis von Leittechnikfunktionsplänen: Die Auswirkungen von Softwarefehlern auf die Verfahrenstechnik werden basierend auf leittechnischen Funktionsplänen (Darstellung der logischen Verknüpfungen zur Realisierung der Leittechnikfunktion) der zu betrachtenden Leittechnikfunktionen untersucht. Dazu werden in den leittechnischen Funktionsplänen die aus der Betriebserfahrung identifizierten Softwarefehlerarten postuliert, deren Auswirkungen auf die Leittechnikfunktion untersucht und anschließend anhand von Systembeschreibungen die Auswirkungen auf die verknüpfte Verfahrenstechnik untersucht.

- In Fällen bei denen detaillierte Leittechnikfunktionspläne nicht verfügbar sind, wird ein postulierter Softwarefehler durch seine Auswirkungen am Ausgang des betrachteten softwarebasierten Leittechniksystems bzw. an seinen Schnittstellen zu anderen Systemen modelliert, z. B. die Ausgangswerte des Leittechniksystems fallen nach MIN/MAX-Werten aus. Die Auswirkungen dieser softwarebedingten Ausfallarten auf angesteuerte Komponenten oder auf andere Systeme werden dann in einem weiteren Schritt, z. B. anhand von Systembeschreibungen, analysiert.

Die Umsetzung der Untersuchungen der potentiellen Auswirkungen von Softwarefehlern mit einem Analysesimulator bedingt die Entwicklung einer Software-Schnittstelle zur „Modifizierung“ der leittechnischen Funktionen. In dem vorhandenen Analysesimulator der GRS ist eine entsprechende Schnittstelle noch nicht implementiert. Für die Untersuchungen im Rahmen dieses Abschnitts standen daher die beiden zuletzt genannten Konzepte zur Auswahl. Aufgrund seiner sicherheitstechnischen Bedeutung wurde im Rahmen dieses Abschnitts das digitale Neutronenflussaußenmesssystem in einem

Druckwasserreaktor untersucht. Da für dieses System keine leittechnischen Funktionspläne zur Verfügung standen wurde bei den Untersuchungen das dritte Konzept angewendet. Nachfolgend werden die dabei erzielten Ergebnisse vorgestellt.

## 6.3 Erprobung

Aus den durchgeführten Recherchen, u. a. Auswertung der Betriebserfahrung mit softwarebasierten Leittechnikeinrichtungen (siehe Kapitel 3, 4 und 5) zwecks Identifizierung und Auswahl von zu untersuchenden relevanten softwarebasierten leittechnischen Einrichtungen ergibt sich, dass softwarebasierte Einrichtungen, Geräte und Systeme u a. im Neutronenflussmesssystem[1], in der Steuerung von BE-Wechselbühne/ Lademaschine, in der Kransteuerung, in der Turbinenregelung, in Funktionen zur Leistungsverteilungsüberwachung, in der DNB/Leistungsdichte-Berechnung und in Steuerungseinrichtungen der Brandmeldeunterzentrale in deutschen Anlagen eingesetzt sind. Zusätzlich wurden in internationalen Anlagen softwarebasierte leittechnische Einrichtungen und Systeme u a. im Kondensatreinigungssystem, im Lüftungssystem der Warte, in der Speisewasserregelung und im Frischdampfsystem identifiziert.

Aufgrund seiner sicherheitstechnischen Bedeutung wurde für die Untersuchungen im Rahmen dieses Abschnitts das Neutronenflussaußenmesssystem („excore"-Instrumentierung) eines Druckwasserreaktors betrachtet, dessen Aufbau und Funktion nachfolgend kurz auf der Basis der Ausführungen in /WES 11/ und /ZA 13/ beschrieben werden.

### 6.3.1 Beschreibung des Neutronenflussaußenmesssystems in einem DWR

Das Neutronenflussaußenmesssystem dient der Überwachung der im Reaktorkern durch Kernspaltungen erzeugten Leistung. Mit dieser Außeninstrumentierung kann der Reaktor vom abgeschalteten kalten, unterkritischen Zustand bis über Nennleistung hinaus überwacht werden. Das Neutronenflussaußenmesssystem liefert u. a.:

[1] In den deutschen DWR-Anlagen mit Berechtigung zum Leistungsbetrieb werden zur Aufbereitung der Detektorsignale, welche aus dem erfassten Neutronenfluss im Neutronenflussaußenmesssystem resultieren, ausschließlich analoge Systeme eingesetzt. Zur Verarbeitung der aufbereiteten Detektorsignale werden in deutschen DWR-Anlagen mit Berechtigung zum Leistungsbetrieb analoge und digitale Systeme eingesetzt.

- Informationen für ein kontrolliertes Anfahren des Reaktors, z.B. die relative Flussänderungsgeschwindigkeit (RELFAEG)
- Messdaten zur Erfassung/Berechnung der im Reaktorkern erzeugten Leistung (in Verbindung mit der Kühlmittelaufwärmspanne)
- Eingangsdaten für das Reaktorschutzsystem zur Auslösung von Schutzaktionen, z. B. RESA
- Eingangsdaten für das Begrenzungssystem zur Auslösung von Maßnahmen der Begrenzung, z. B. Leistungsbegrenzung
- Eingangsdaten für betriebliche Leittechniksysteme wie z. B. Reaktorleistungsregelungssystem

Da der zu überwachende Bereich des Neutronenflusses ($10^{-2}$-$10^{8}$ $cm^{-2}s^{-1}$) nicht durch ein einziges Detektorsystem mit ausreichender Messgenauigkeit erfasst werden kann, wird dieser Bereich in drei sich überlappende Bereiche unterteilt, wobei jedem Messbereich ein Detektorsystem zugeteilt wird:

- Impulsbereich (IB, 2-fach aufgebaut, $10^{-2}$-$10^{5}$ $cm^{-2}s^{-1}$)
- Mittelbereich (MB, 4-fach aufgebaut, $10^{2}$-$10^{6}$ $cm^{-2}s^{-1}$)
- Leistungsbereich (LB, 4-fach aufgebaut, $10^{6}$-$10^{8}$ $cm^{-2}s^{-1}$)

Von den Detektoren, die in Gliederzügen untergebracht sind, werden Signale (2 IB-, 4 MB-, 8 LB-Signale) generiert und zu den Leittechnikschränken des Neutronenflussmesssystems geführt. Dort werden die erfassten Detektorsignale aufbereitet und verarbeitet. Die Detektorsignale werden u. a. aufgrund deren Temperaturabhängigkeit für Kalibrierungszwecke „temperaturkorrigiert“.

Aus den Zählratensignalen der IB-Detektoren werden in der Signalverarbeitung die Signale der Impulsrate und der reziproken Reaktorperiode bestimmt. Das Signal der Impulsrate wird der Temperaturkorrektur zugeführt und mit einem entsprechenden Korrekturfaktor multipliziert. Aus der Impulsrate wird die relative Flussänderungsgeschwindigkeit RELFAEG im Impulsbereich abgeleitet. Sie ist definiert als die reziproke Reaktorperiode.

Die Signalverarbeitung der MB-Detektorsignale bestimmt aus dem erfassten Neutronenfluss die Neutronenflussdichte und die RELFAEG im Mittelbereich, wobei die Neutronenflussdichte als Stromsignal MB [A] angegeben wird. Dieses Stromsignal wird einer Temperaturkorrektur unterzogen. Anschließend wird das „temperaturkorrigierte“ Signal in zwei Signale aufgeteilt, das Leistungssignal und das Stromsignal, die sich nur durch ihre physikalische Einheit und Kalibrierung unterscheiden. Durch Skalierung in $\%_{PR}$ ($10^{-5}$A ^=100$\%_{PR}$) wird das Leistungssignal gebildet. Dieses Leistungssignal wird anschließend kalibriert. Das Stromsignal dient im Reaktorschutz zur Überwachung der Überlappung von Impuls- und Mittelbereich.

In der Signalverarbeitung im Leistungsbereich wird das den Signalen überlagerte Neutronenflussrauschen durch ein Rauschfilter unterdrückt. Die gefilterten Signale werden der Temperaturkorrektur zugeführt und anschließend kalibriert.

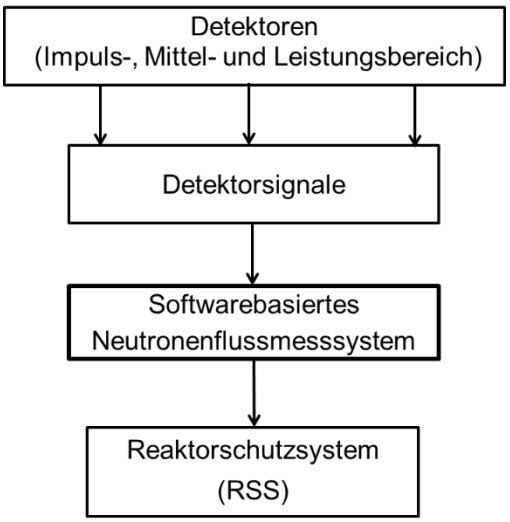



**Abb. 6.1** Struktureller Aufbau eines Neutronenflussmesssystems mit Schnittstellen zum Reaktorschutzsystem /MBO 16/

Als Ergebnis der Verarbeitung der Detektorsignale stehen die berechneten Signale der Impulsrate im Impulsbereich (IB [ips]), der Neutronenflussdichte im Mittelbereich (MB [A]), der Neutronenflussdichte im Leistungsbereich (LB [%]) und der relativen Flussänderungsgeschwindigkeit (RELFAEG [%/s]) für die weitere Verwendung in anderen Leittechniksystemen (Reaktorschutzsystem, Begrenzungssystem, Reaktorregelung, betriebliche Leittechnik) zur Verfügung (siehe Abb. 6.1).

### 6.3.2 Annahmen für die Untersuchungen

Es wird im Rahmen der durchgeführten Untersuchungen in diesem Abschnitt angenommen, dass die Aufbereitung und Verarbeitung der erfassten Detektorsignale gemäß den Ausführungen im Abschnitt 6.3.1 durch ein softwarebasiertes Neutronenflusssystem erfolgen, d.h. die Berechnung der Ausgangssignale, Impulsrate im Impulsbereich (IB [ips]), Neutronenflussdichte im Mittelbereich (MB [A]), Neutronenflussdichte im Leistungsbereich (LB [%]) und relative Flussänderungsgeschwindigkeit (RELFAEG [%/s]), erfolgen softwarebasiert. Die so berechneten Signale werden für die weitere Verwendung an entsprechende Leittechniksysteme weitergegeben.

Ziel der Untersuchungen ist, die Auswirkungen von postulierten Softwarefehlern in leittechnischen Funktionen zur Berechnung der Ausgangssignale des softwarebasierten Neutronenflusssystems zu analysieren. Die Analyse erfolgt am Beispiel der Grenzsignalverarbeitung des Reaktorschutzsystems eines Kernkraftwerks mit einem Druckwasserreaktor. Es ergibt sich unter diesen Annahmen der in Abbildung 6.2 dargestellte vereinfachte logische Funktionsplan der Grenzsignalverarbeitung der Neutronenflusssignale im Reaktorschutzsystem. Dieser vereinfachte Funktionsplan ist die Grundlage für die durchgeführten Untersuchungen zu den Auswirkungen von postulierten Softwarefehlern in dem softwarebasierten Neutronenflussmesssystem.

In der Grenzsignalverarbeitung des Reaktorschutzsystems werden die Ausgangssignale des Neutronenflussmesssystems mittels Grenzsignalgeber auf das Unterschreiten (MIN-Grenzsignalgeber) bzw. Überschreiten (MAX-Grenzsignalgeber) von festgelegten Grenzwerten überwacht. Die sich auf diese Weise ergebenden binären Ausgangssignale der MIN- bzw. MAX-Grenzsignalgeber der Neutronenflusssignale werden zur Auslösung der Reaktorschnellabschaltung miteinander und/oder mit binären Ausgangssignalen der MIN-Grenzsignalgeber für den Kühlmitteldruck und für die Kühlmitteltemperatur logisch verknüpft (siehe Abb. 6.2).

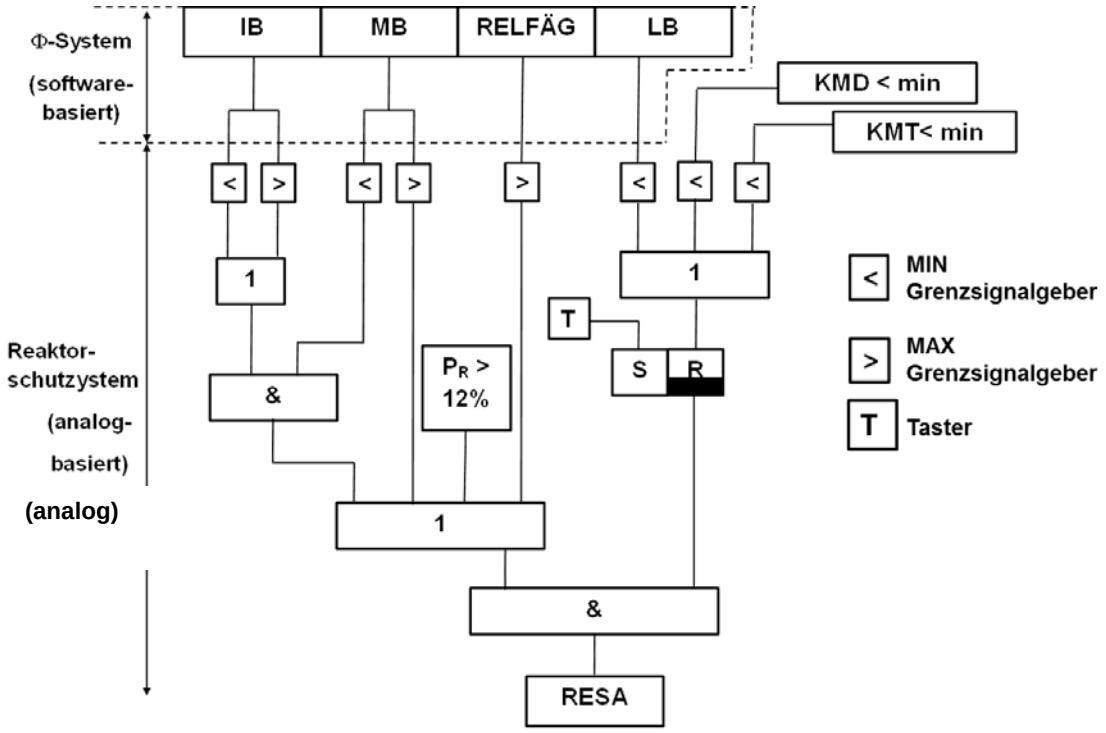



**Abb. 6.2** Logikplan der Grenzsignalverarbeitung der Neutronenflusssignale im Reaktorschutzsystem (vereinfachte Darstellung)

### 6.3.3 Untersuchungsergebnisse

Für die Untersuchungen wird ein Softwarefehler in der Software zur Berechnung der Ausgangssignale des softwarebasierten Neutronenflussmesssystems postuliert. Dieser Softwarefehler wird anhand seiner Auswirkungen an den Ausgangssignalen des softwarebasierten Neutronenflussmesssystems modelliert (siehe Abb. 6.3). Es wird angenommen, dass die Anlage sich im Leistungsbetrieb mit einer Reaktorleistung $P_R > 12$ % befindet. Folgende Fälle, z. B. als Folge einer fehlerhaften Kalibrierung der Detektorsignale in der Software, wurden betrachtet:

- Fall 1: Ausfall aller Ausgangssignale des softwarebasierten Neutronenflusssystems nach MIN
- Fall 2: Ausfall aller Ausgangssignale des softwarebasierten Neutronenflusssystems nach MAX
- Fall 3: Fehlerhaftes „Einfrieren“ aller Ausgangssignale des softwarebasierten Neutronenflussmesssystems im Bereich [MIN, MAX]

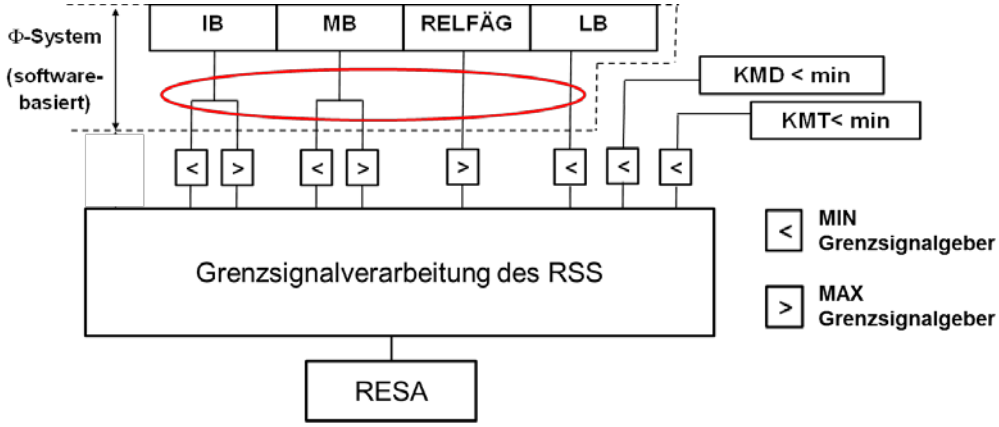



**Abb. 6.3** Softwarefehlermodell für die Untersuchungen: Postulierter Softwarefehler wird durch Auswirkungen an den Ausgangssignalen modelliert

Ein Ausfall aller Ausgangssignale des softwarebasierten Neutronenflussmesssystems nach MIN (Fall 1) aufgrund eines Softwarefehlers führt zur Unwirksamkeit der MAX-Grenzsignalgeber im betroffenen Auslösepfad des Reaktorschutzsystems (siehe Abb. 6.4).

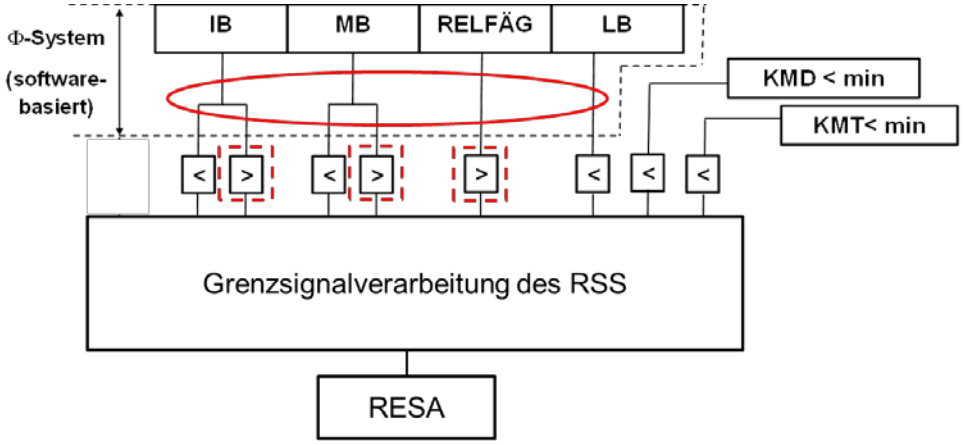



**Abb. 6.4** MAX-Grenzsignalgeber der Grenzsignalverarbeitung des RSS unwirksam bei Ausfall der Ausgangssignale des Neutronenflussmesssystems nach MIN

In diesem Fall kommt es zur Auslösung der Reaktorschnellabschaltung (RESA) über das Kriterium “Neutronenfluss LB < min” (siehe Abb. 6.2). Der Softwarefehler zeigt ein „fail-safe“-Verhalten in diesem Fall.

Ein Ausfall aller Ausgangssignale des softwarebasierten Neutronenflusssystems nach MAX (Fall 2) aufgrund eines Softwarefehlers führt zur Unwirksamkeit der MIN-Grenzsignalgeber im betroffenen Auslösepfad des Reaktorschutzsystems (siehe Abb. 6.5).

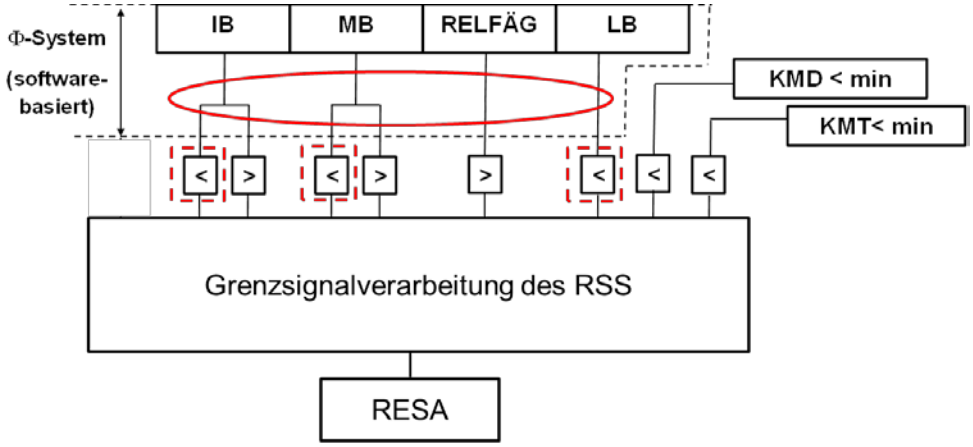



**Abb. 6.5** MIN-Grenzsignalgeber der Grenzsignalverarbeitung des RSS unwirksam bei Ausfall der Ausgangssignale des Neutronenflussmesssystems nach MAX

Es ergibt sich, dass in diesem Fall der unterstellte Softwarefehler zur Blockierung der Reaktorschnellabschaltung (RESA) über das Kriterium “Neutronenfluss LB < min” führen kann (siehe Abb. 6.2).

Die RESA-Auslösung wäre in einem solchen Fall durch andere diversitäre RESA-Auslösepfade (z. B. über KMD < min und/oder KMT < min) sichergestellt. Bei diesen RESA-Kriterien sind jedoch längere Reaktionszeiten als beim Neutronenfluss zu erwarten, so dass die RESA über die Kriterien KMD<min und/oder KMT < min im Vergleich zum Kriterium „Neutronenfluss LB < min“ verzögert angeregt wird[2].

[2] Nach der Veröffentlichung des vorliegenden Berichts hat die GRS Untersuchungen zur sicherheitstechnischen Bedeutung der verzögert angeregten RESA durchgeführt. Diese ergaben, dass die Auslösung der RESA bei einem Ausfall der Ausgangssignale des Neutronenflussaußenmesssystems nach MIN oder MAX (für MAX bei Berücksichtigung der Rechenschaltung für die Reaktorleistung) zu einer sofortigen RESA-Auslösung führt. Die detaillierten Ergebnisse der von der GRS durchgeführten Untersuchungen sind in /GRS 21/ dokumentiert.

Ein fehlerhaftes „Einfrieren“ aller Ausgangssignale des softwarebasierten Neutronenflussmesssystems im Bereich [MIN, MAX] führt zur Unwirksamkeit der MIN- und MAX-Grenzsignalgeber im betroffenen RESA-Auslösepfad des Reaktorschutzsystems. Dies hat zur Folge, dass die RESA-Auslösung über die Neutronenflussgrenzwerte unwirksam wird. Die RESA-Auslösung wäre in einem solchen Fall ähnlich wie im Fall 2 durch andere diversitäre RESA-Auslösepfade (z. B. über KMD < min und/oder KMT < min) sichergestellt[3].

[3] Nach der Veröffentlichung des vorliegenden Berichts hat die GRS Untersuchungen zur sicherheitstechnischen Bedeutung der verzögert angeregten RESA durchgeführt. Diese ergaben, dass auch bei einem „Einfrieren“ der Ausgangssignale des Neutronenflussaußenmesssystems als Folge eines Softwarefehlers alle Anforderungen des zugrunde gelegten Regelwerks erfüllt werden. Die detaillierten Ergebnisse der von der GRS durchgeführten Untersuchungen sind in /GRS 21/ dokumentiert.

# 7 Zusammenfassung und Ausblick

Ziel dieses Vorhabens war es, die Wissensbasis der GRS über Auswirkungen von Softwarefehlern in softwarebasierten Leittechniksystemen zu erweitern, um dadurch eine wesentliche Grundlage für eine fundierte und sachgerechte sicherheitstechnische Bewertung von softwarebasierten Leittechniksystemen zu schaffen.

Dazu wurde der relevante Stand von Wissenschaft und Technik ermittelt und aufbereitet. Unter Einbeziehung der Ergebnisse bzw. Zwischenergebnisse bisheriger Arbeiten der GRS zum Einsatz von softwarebasierter Leittechnik in Kernkraftwerken wurden relevante Aspekte softwarebasierter Leittechniksysteme u. a. hinsichtlich Aufbau, Einsatz und sicherheitstechnische Klassifizierung aufbereitet und zusammenfassend dargestellt. Weiterhin wurden anhand einer Literaturstudie nach Definitionen/Beschreibungen von für das Vorhaben relevanten Begriffen (Software, sicherheitskritische Software, Softwarefehler, Malware, Softwareversagen, Systemfehler, Systemversagen) in relevanten Normen, Standards und Veröffentlichungen (DIN-, IEEE-Normen, IAEA-Standards, NUREG-Veröffentlichungen) sowie in den Sicherheitsanforderungen recherchiert. Dabei stellte sich insbesondere heraus, dass der Begriff Softwarefehler in den betrachteten Literaturquellen unterschiedlich und teilweise widersprüchlich definiert wird.

Aus diesem Grund wurde im Rahmen dieses Vorhabens unter Betrachtung der Lebenszyklusphasen der Software bzw. des betrachteten softwarebasierten Leittechniksystems eine Definition von Softwarefehlern entwickelt, welche die verschiedenen für Softwarefehler relevanten Aspekte berücksichtigt. Basierend auf den hierfür durchgeführten Untersuchungen wurde definiert, dass Softwarefehler als latente Fehler in einem softwarebasierten Leittechniksystem vorliegen und zu einem nichtanforderungsgerechten Verhalten eines softwarebasierten Leittechniksystems führen können.

Bei den Softwarefehlerarten ist im Wesentlichen zwischen:

- Programmierfehler: Fehler in der programmiertechnischen Umsetzung der Software-Anforderungsspezifikation bei angenommener anforderungsgerechter Spezifikation des softwarebasierten Leittechniksystems

und

- Spezifikationsfehler: Fehler in der Systemspezifikation (Hard- und/oder Software) des softwarebasierten Leittechniksystems bei angenommener spezifikationsgemäßer programmiertechnischer Umsetzung der Software-Anforderungsspezifikation des softwarebasierten Leittechniksystems

zu unterscheiden.

In einem weiteren Arbeitsschritt wurden Recherchen zur Betriebserfahrung mit Softwarefehlern in softwarebasierter Leittechnik im nuklearen und im nichtnuklearen Bereich angestellt. Hierzu wurden für die Ereignisse aus dem nuklearen Bereich die GRS-Datenbank (GRS-DB) für Ereignisse aus der deutschen Betriebserfahrung, die ICDE-GVA-Datenbank, die COMPSIS-Datenbank und die IRS-Datenbank der IAEA für Ereignisse aus der internationalen Betriebserfahrung durchsucht. Im nichtnuklearen Bereich wurde nach Ereignissen aufgrund von Softwarefehlern in der Luft- und Raumfahrt sowie in der Medizin recherchiert. Diese Ereignisse wurden hinsichtlich ihrer Ursache und Auswirkungen analysiert und die Ergebnisse zusammenfassend beschrieben. Die Ergebnisse dieser Auswertung der Betriebserfahrung sind in Kapitel 3 und Kapitel 4 dargestellt.

Auf der Basis der im Rahmen dieses Vorhabens entwickelten Definition von Softwarefehlern und des COMPSIS-Klassifizierungsschemas wurde zur Klassifizierung ermittelter Ereignissen aus der Betriebserfahrung mit Softwarefehlern das COCS-Klassifizierungsschema entwickelt, bei dem die Ereignisse nach den Kriterien „Ursache“, „betroffenes System“, „Auswirkung“ sowie „GVA-Potenzial“ klassifiziert werden. Dieses Klassifizierungsschema wurde bei der Auswertung der im Rahmen des Vorhabens ermittelten Ereignisse angewandt. Daraus ergibt sich, dass die meisten Softwarefehler auf Programmierfehler und/oder Spezifikationsfehler zurückzuführen sind. Softwarefehler können auch aus dem Zusammenwirken des Leittechniksystems mit ihrer Betriebsumgebung resultieren, z. B. aufgrund von Hardwarefehlern. Eine weitere Softwarefehlerart betrifft Fehler, die sich aufgrund fehlerhafter System-Anforderungsspezifikation ergeben. Derartige Fehler sind unabhängig von der Realisierung des Leittechniksystems (analog oder softwarebasiert). Detaillierte Ergebnisse dieser Auswertung sind in Kapitel 5 zu finden.

Aufbauend auf den identifizierten Softwarefehlerarten aus der Auswertung der Betriebserfahrung wurden potentielle Auswirkungen von einzelnen Softwarefehlerarten auf spezielle Leittechnikfunktionen und auf die damit verknüpfte Verfahrenstechnik untersucht. Dazu wurden im Rahmen dieses Arbeitsabschnittes verschiedene Konzepte erarbeitet.

Den Untersuchungskonzepten ist gemeinsam, dass zunächst jeweils einzelne Softwarefehlerarten in einer betrachteten softwarebasierten Leittechnikfunktion postuliert und anschließend deren Auswirkungen analysiert werden. Die Untersuchungskonzepte unterscheiden sich in deren Umsetzung im Wesentlichen in der erzielbaren Detaillierungstiefe bei der Modellierung der Softwarefehler, der betrachteten Leittechnikfunktionen bzw. Leittechniksysteme und deren Verknüpfung mit der angesteuerten Verfahrenstechnik. Prinzipiell ist es möglich die potentiellen Auswirkungen von Softwarefehlern mit einem Analysesimulator und/oder auf der Basis von Leittechnikfunktionsplänen und/oder anhand von Systembeschreibungen (falls keine detaillierten Leittechnikfunktionspläne verfügbar) zu untersuchen.

Bei den Untersuchungen auf der Basis von Systembeschreibungen wird ein postulierter Softwarefehler durch seine Auswirkungen am Ausgang des betrachteten softwarebasierten Leittechniksystems bzw. an seinen Schnittstellen zu anderen Systemen modelliert. Diese Vorgehensweise wurde am Beispiel eines softwarebasierten Neutronenflussmesssystems in einem Druckwasserreaktor erprobt. Dazu wurden in der Betriebserfahrung von anderen Leittechniksystemen identifizierten Softwarefehlerarten bei der Software des Neutronenflussmesssystems in der Funktion zur Bildung der Neutronenfluss-Grenzwerte für Auslösesignale des Reaktorschutzsystems unterstellt: Ausgangswerte des softwarebasierten Neutronenflussmesssystems fallen nach MIN- (Fall 1) bzw. MAX-Werten (Fall 2) aus, fehlerhaftes „Einfrieren“ der Ausgangswerte des softwarebasierten Neutronenflussmesssystems im Wertebereich [MIN, MAX] (Fall 3). Es zeigte sich, dass die unterstellten Fehler in Fall 2 und Fall 3 zur Blockade von einzelnen Auslösepfaden für die Reaktorschnellabschaltung (RESA) im Leistungsbetrieb der Anlage führen können. Die RESA-Auslösung wäre in solchen Fällen durch andere diversitäre RESA-Auslösepfade sichergestellt, bei denen jedoch längere Reaktionszeiten zu erwarten sind[4].

[4] Nach der Veröffentlichung des vorliegenden Berichts hat die GRS Untersuchungen zur sicherheitstechnischen Bedeutung der verzögert angeregten RESA durchgeführt. Diese ergaben, dass die Auslösung der RESA bei einem Ausfall der Ausgangssignale des Neutronenflussaußenmesssystems nach MIN oder MAX (für MAX bei Berücksichtigung der Rechenschaltung für die Reaktorleistung) zu einer sofortigen RESA-Auslösung führt, was einem „fail-safe“-Verhalten entspricht, da hierdurch eine eindeutig sicherheitsgerichtete Schutzaktion gemäß der KTA-Regel 3501 /KTA 15/ ausgelöst wird. Weiterhin ergaben diese, dass auch bei einem „Einfrieren“ der Ausgangssignale des Neutronenflussaußenmesssystems als Folge eines Softwarefehlers alle Anforderungen des zugrunde gelegten Regelwerks erfüllt werden. Die detaillierten Ergebnisse der von der GRS durchgeführten Untersuchungen sind in /GRS 21/ dokumentiert.

Bei der im Rahmen dieses Vorhabens entwickelten und erprobten Methode zur Analyse der Auswirkungen von Softwarefehlern in softwarebasierter Leittechnik lag der Schwerpunkt der Untersuchungen in der Analyse der Auswirkungen von Softwarefehlern auf die Systemebene (Auswirkung auf die angesteuerte verfahrenstechnische Komponente). Für eine umfassende Analyse der Auswirkungen von Softwarefehlern ist es ebenfalls von Bedeutung, die Auswirkungen von Softwarefehlern innerhalb von Leittechniksystemen zu untersuchen.

## Literatur


/BAN 92/ Verordnung über den kerntechnischen Sicherheitsbeauftragten und über die Meldung von Störfällen und sonstigen Ereignissen (Atomrechtliche Sicherheitsbeauftragten- und Meldeverordnung - AtSMV), Bundesgesetzblatt, Jahrgang 1992, Teil I, S. 1766-1775

/BAN 10/ Erste Verordnung zur Änderung der Atomrechtlichen Sicherheitsbeauftragten- und Meldeverordnung, Bundesgesetzblatt, Jahrgang 2010, Teil I, S. 755-780

/BFU 14/ Bundesstelle für Flugunfalluntersuchung (BFU): Unfälle und Störungen beim Betrieb ziviler Luftfahrzeuge, erreichbar unter http://www.bfu-web.de/DE/Publikationen/Bulletins/2014/Bulletin2014-11.pdf;jsessionid=52B7650517B1B650B9A70FCA4D9855A5.live11291?__blob=publicationFile.

/BHA 12/ Bharadwaj, P.: Flame Maleware Exploits Microsoft's Digital Certificate, Symantec, Juni 2012, erreichbar unter http://www.symantec.com/connect/blogs/flame-malware-exploits-microsofts-digital-certificate.

/BMUB 12/ Bundesministerium für Umwelt, Naturschutz, Bau und Reaktorsicherheit (BMUB): Sicherheitsanforderungen an Kernkraftwerke, November 2012, erreichbar unter http://www.bfs.de/SharedDocs/Downloads/BfS/DE/rsh/3-bmub/3_0_1.pdf?__blob=publicationFile&v=6.

/BSI 10/ Bundesamt für Sicherheit in der Informationstechnik (BSI): Neue Sicherheitslücke im Windows-Betriebssystem von Microsoft, 21. Juli 2010, erreichbar unter https://www.bsi.bund.de/DE/Presse/Pressemitteilungen/Presse2010/Sicherheitsluecke_Windows_210710.html.

/CER 17/ Certitudo GmbH, http://www.certitudo-gmbh.de/g_geschichte.html, März 2017

/COM 11/ GRS gGmbH: Mitarbeit im OECD/NEA COMPSIS-Projekt, Abschlussbericht 810410, Oktober 2011.

/COM 15/ Vital engine software files accidentally wiped, linked to fatal A400M plane crash. Computerworld, 20.08.2015.

/DIN 02/ Deutsches Institut für Normung (DIN): Kernkraftwerke - Leittechnik für Systeme mit sicherheitstechnischer Bedeutung - Allgemeine Systemanforderungen, DIN EN 61513:2002, März 2002.

/DIN 09/ Deutsches Institut für Normung (DIN): Kernkraftwerke - Leittechnik für Systeme mit sicherheitstechnischer Bedeutung - Softwareaspekte für rechnerbasierte Systeme zur Realisierung von Funktionen der Kategorien B und C, DIN EN 62138:2009, März 2010.

/DIN 10/ Deutsches Institut für Normung (DIN): Leittechnische Systeme mit sicherheitstechnischer Bedeutung - Kategorisierung leittechnischer Funktionen, DIN EN 61226, August 2010.

/DIN 14/ Deutsches Institut für Normung (DIN): Internationales Elektrotechnisches Wörterbuch - Teil 351: Leittechnik, DIN IEC 60050-351:2014-09, September 2014.

/DIN 60/ Deutsches Institut für Normung (DIN): Leittechnik für Systeme mit sicherheitstechnischer Bedeutung - Softwareaspekte für rechnerbasierte Systeme zur Realisierung von Funktionen der Kategorie A,
DIN EN 60880:2009, März 2010.

/GOS 12/ Gostev, A.: Teh Roof is on Fire: Tackling Flame's C&C Servers, Juni 2012, erreichbar unter http://securelist.com/blog/incidents/33033/the-roof-is-on-fire-tackling-flames-cc-servers-6/.

/GRS 15/ GRS gGmbH: Aufstellung von Kriterien und Kenngrößen zur deterministischen Prüfung der Eignung von Redesign-Komponenten für den Einsatz in der Sicherheitsleittechnik von Kernkraftwerken, 3611R01355, März 2015.

/GRS 15b/ GRS gGmbH: Entwicklung und Einsatz von Analysemethoden zur Beurteilung softwarebasierter leittechnischer Einrichtungen in deutschen Kernkraftwerken, März 2015.

/GRS 16/ GRS gGmbH: Zuverlässigkeitsbewertung unter neuen Anforderungen an Sicherheitsleittechnik in Kernkraftwerken: Analysen der Anwendungspraxis, 3613R01322, Oktober 2016.

/GRS 21/ GRS-Stellungnahme zur Bewertung der sicherheitstechnischen Bedeutung der Ergebnisse des GRS-Forschungsvorhabens „Auswirkungsbereiche von Softwarefehlern in sicherheitstechnisch wichtigen Einrichtungen von Kernkraftwerken“ für deutsche Kernkraftwerke im Leistungsbetrieb, März 2021

/HEI 10/ Heise.de: Windows-LNK-Lücke: Lage spitzt sich zu. Pressemitteilung, 20. Juli 2010.

/IAE 02/ IAEA: Instrumentation and Control Systems Important to Safety in Nuclear Power Plants, Safety Standards Series No. NS-G-1.3, 2002.

/IAE 07/ IAEA: Terminology Used in Nuclear Safety and Radiation Protection, Safety Glossary, 2007.

/IAE 14/ IAEA: Safety Classification of Structures, Systems and Components in Nuclear Power Plants, Safety Standards No. SSG-30, 2014.

/IAE 80/ IAEA: Protection System and Related Features in Nuclear Power Plants, Safety Guides No. 50-SG-D3, 1980.

/IAE 84/ IAEA: Safety Related Instrumentation and Control Systems for Nuclear Power Plants, Safety Guides No. 50-SG-D8, 1984.

/IAE 99/ IAEA: Verification and Validation of Software Related to Nuclear Power Plant Instrumentation and Control, Technical Report No. 384, 1999.

/ICS 16/ E-ISAC: Electricity-Information Sharing and Analysis Center: Analysis of the Cyber Attack on the Ukrainian Power Grid, März, 2016

/IEE 03b/ IEEE: Standard for Qualifying Class 1E Equipment for Nuclear Power Generating Stations, IEEE 323, 2003.

/IEE 09/ IEEE: Standard Criteria for Safety Systems for Nuclear Power Generating Stations, IEEE Std. 603, 2009.

/IEE 10/ IEEE; ISO; IEC: Systems and Software Engineering - Vocabulary, IEEE 24765, 2010.

/KAS 15/ Kaspersky Lab: Was ist ein Computervirus oder ein Computerwurm?, 10. März 2015, erreichbar unter http://www.kaspersky.com/de/internet-security-center/bedrohungen/viren-wuermer.

/KNK 15/ Komite Nasional Keselamatan Transportasi republic of indonesia: Aircraft Accident Investigation Report, PT. Indonesia Air Asia Airbus A320-216; PK-AXC10 Final Report, December 2015, erreichbar unter https://www.bea.aero/uploads/tx_elydbrapports/Final_Report_PK-AXC-reduite.pdf

/KTA 14/ Kerntechnischer Ausschuss (KTA): Allgemeine Anforderungen an die Qualitätssicherung, KTA 1401, 11.2014.

/KTA 14b/ Kerntechnischer Ausschuss (KTA): Reaktorschutzsystem und Überwachungseinrichtungen des Sicherheitssystems, KTA 3501, 2014.

/KUP 10/ Kupreev, O., Ulasen, S.: Trojan-Spy.0485 and Malware-Cryptor.Win32.Inject.gen.2 Review, Juli 2010.

/LAW 93/ Lawrence, J.D.: Software Reliability and Safety in Nuclear Reactor Protection Systems, US NRC, Juni 1993.

/LEV 93/ Leverson, N., Turner, C.: An Investigation of the Therac-25 Accidents, 26. Aufl., IEEE: Computer, Juli 1993.

/LIO 96/ Lions, J.L.: Ariane 5, Flight 501 Failure, Report by the Inquiry Board, 1996.

/MBO 16/ H. Mbonjo, M. Jopen, B. Ulrich, D. Sommer: “Approach for the Evaluation of the Impact of Potential Software Failures In Software-Based Instrumentation And Control (I&C) Equipment in Nuclear Power Plants (NPP)”, Proceedings of the 2016 24th International Conference on Nuclear Engineering (ICONE), June 26-30, 2016, Charlotte North Carolina

/NEW 09/ Newman, R.C.: Computer Security: Protecting Digital Resources, Februar 2009.

/PRE 11/ Prechelt, L.: Anwendungssysteme - Sicherheit: Therac-25, Institut für Informatik, Freie Universität Berlin, 2011, erreichbar unter http://www.inf.fu-berlin.de/inst/ag-se/teaching/V-AWS-2011/22_Sicherheit.pdf.

/RSK 96/ RSK: RSK-Leitlinien für Druckwasserreaktoren, 15. November 1996, erreichbar unter http://www.rskonline.de/sites/default/files/German/downloads/8110dwr.pdf.

/SCH 12/ Schmidt, J.: FAQs zum Superspion Flame, Heise.de, Mai 2012, erreichbar unter http://www.heise.de/security/artikel/FAQs-zum-Superspion-Flame-1586382.html.

/SIE 16/ Schmidt, J.: SIMATIC WinCC V7, April 2016, The scalable and open SCADA system for maximum plant transparency and productivity, erreichbar unter https://c4b.gss.siemens.com/resources/images/articles/e20001-a820-p810-v2-7600.pdf

/SPI 11/ Stuxnet-Code in freier Wildbahn. Spiegel Online, 14.02.2011.

/SYM 11/ Symantec: W32.Duqu: The Precursor of the Next Stuxnet, 2011, erreichbar unter http://www.symantec.com/connect/w32_duqu_precursor_next_stuxnet.

/TEC 15/ techterms.com: Computer Dictionary, 9. März 2015, erreichbar unter http://techterms.com/.

/WES 11/ Westinghouse Technology Systems Manual, Section 9.1 Excore Nuclear Instrumentation, erreichbar unter https://www.nrc.gov/docs/ML1122/ML11223A263.pdf

/WIN 15/ WinFuture: Softwarefehler: Toyota ruft fast 2 Mio. Prius zurück, 21. August 2015, erreichbar unter http://winfuture.de/news,80287.html.

/ZA 13/ Ziegler, A., Allelein, H.-J., Reaktortechnik Physikalisch-technische Grundlagen 2., neu bearbeitete Auflage 2013

/ZET 10/ Zetter, K.: Clues Suggest Stuxnet Virus was built for subtle nuclear sabotage, 2010, erreichbar unter http://www.wired.com/2010/11/stuxnet-clue.

## Tabellenverzeichnis

## Abbildungsverzeichnis

**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 978-3-946607-39-7**