

	Grundsätze für Rechner in Systemen mit Sicherheitsaufgaben	Vornorm DIN V VDE 0801
	Principles for computers in safety-related systems Eine Vornorm ist das Ergebnis einer Normungsarbeit, das wegen bestimmter Vorbehalte zum Inhalt oder wegen des gegenüber einer Norm abweichenden Aufstellungsverfahrens vom DIN noch nicht als Norm herausgegeben wird. Zur vorliegenden Vornorm ist kein Norm-Entwurf veröffentlicht worden. Eine Vornorm ist nicht Teil des Deutschen Normenwerks und nicht Teil des VDE-Vorschriftenwerks. Erfahrungen mit dieser Vornorm sind erbeten an die Deutsche Elektrotechnische Kommission im DIN und VDE (DKE), Geschäftsstelle Frankfurt, Stresemannallee 15, 6000 Frankfurt am Main 70. Für den Anwendungsbereich dieser Vornorm bestehen keine europäischen oder internationalen Normen. Es ist beabsichtigt, aufgrund dieser Vornorm Vorschläge für einschlägige internationale und europäische Normungsarbeiten bei der Internationalen Elektrotechnischen Kommission (IEC) und bei der Gemeinsamen Europäischen Normeninstitution (CEN/CENELEC) einzubringen (siehe Erläuterungen). Sobald eine entsprechende Europäische Norm oder ein entsprechendes Harmonisierungsdokument vorliegt und in das Deutsche Normenwerk und das VDE-Vorschriftenwerk übernommen ist, wird diese Vornorm zurückgezogen. Vorbemerkung Zum besseren Verständnis des Inhalts werden mit Bezug auf Anwendungsbereich und Zweck dieser Vornorm (siehe Abschnitt 1) folgende Hinweise vorangestellt: Diese Vornorm wendet sich in erster Linie an anwendungsbezogen arbeitende Normungsgremien (siehe Abschnitt 1.1). Als eine Offerte soll sie diesen Gremien bei der Auswahl und der Festlegung ihrer auf bestimmte Anwendungsfälle bezogenen Anforderungen für den Einsatz von Rechnern mit Sicherheitsaufgaben dienen, wobei es erforderlich sein kann, die vorliegenden Festlegungen auch zu modifizieren. Diese Vornorm kann aber ebenso unmittelbar dem Planer oder dem Entwickler beim beabsichtigten Einsatz von Rechnern mit Sicherheitsaufgaben helfen, die im jeweiligen Anwendungsfall richtigen Entscheidungen zu treffen, solange es dafür keine speziellen Normen oder andere Regeln der Technik gibt. Die Beispiele im Anhang X sollen solche Anwendungen verdeutlichen helfen. Die Grundsätze dieser Vornorm sind zu überarbeiten, sobald sich in der praktischen Umsetzung oder Anwendung abweichende Prinzipien als zweckmäßiger erweisen sollten (siehe Erläuterungen). Fortsetzung Seite 2 bis 172 Deutsche Elektrotechnische Kommission im DIN und VDE (DKE) Normenausschuß Informationsverarbeitungssysteme (NI) Normenausschuß Maschinenbau (NAM) im DIN Deutsches Institut für Normung e.V.	

Inhalt

	Seite
1 Anwendungsbereich und Zweck	5
2 Begriffe	6
3 Grundlegende Sicherheitsbetrachtungen	10
3.1 Zusammenhang mit dem Grundschemata nach DIN V 19250	10
3.2 Besonderheiten der Sicherheitstechnik mit Rechnern	12
3.3 Rechnerspezifische Merkmale, für die Anforderungen gestellt werden	16
3.4 Zusammenhang zwischen Anforderungsklassen, Fehlerannahmen und Fehlerausschlüssen	17
4 Fehlerursachen in Rechnern	18
4.1 Unterscheidung der Fehler nach ihrem zeitlichen Auftreten	18
4.1.1 Fehler, entstanden bis einschließlich Inbetriebnahme	18
4.1.2 Fehler, entstanden nach der Inbetriebnahme	20
4.2 Fehlerarten	20
5 Maßnahmen zur Vermeidung von Fehlern	23
5.1 Vermeidung von Fehlern in der Konzeptphase	24
5.2 Vermeidung von Fehlern bei der Entwicklung	27
5.3 Vermeidung von Fehlern bei der Fertigungsvorbereitung	31
5.4 Vermeidung von Fehlern bei der Fertigung	32
5.5 Vermeidung von Fehlern bei der Inbetriebnahme	32
5.6 Vermeidung von Fehlern bei der Installation	33
5.7 Vermeidung von Fehlern beim Betrieb	34
5.8 Vermeidung von Fehlern bei Änderungen	35
6 Maßnahmen zur Beherrschung von Fehlern	36
6.1 Strukturelle Maßnahmen zur Beherrschung von Fehlern auf Systemebene	37
6.2 Maßnahmen zur Beherrschung von Fehlern unterhalb der Systemebene	39
6.3 Nichttechnische Maßnahmen zur Beherrschung von Fehlern	40
7 Zusammenhang zwischen Anforderungsklassen und Maßnahmen	40
7.1 Aufteilung der Maßnahmen in Gruppen zur Erleichterung der Kombination von Maßnahmen	40
7.2 Merkmale für die Anforderungsklassen	43
7.3 Anforderungen für die einzelnen Merkmale in Abhängigkeit von der Anforderungsklasse	43
7.4 Erfüllung der Anforderungen durch Maßnahmenkombinationen	46
Anhang A Maßnahmen zur Vermeidung von Fehlern	48
A.1 Verwenden einer verbindlichen Systemspezifikation	48
A.2 Verwendung von rechnergestützten Entwicklungswerkzeugen	49
A.3 Strukturierter Entwurf	51
A.4 Verwendung von Checklisten	52
A.5 Inspektionen der Spezifikation	52
A.6 Benutzerfreundlichkeit	53
A.7 Instandhaltungsfreundlichkeit	54
A.8 Maßnahmen zur Vermeidung von Fehlern bei der Fertigungsvorbereitung	55
A.9 Maßnahmen zur Vermeidung von Fehlern bei der Fertigung	56
A.10 Maßnahmen zur Vermeidung von Fehlern bei der Inbetriebnahme	58
A.11 Maßnahmen zur Vermeidung von Fehlern bei der Installation	59
A.12 Maßnahmen zur Vermeidung von Fehlern beim Betrieb	61
A.13 Maßnahme zur Vermeidung von Fehlern bei Änderungen	63

	Seite
Anhang B Maßnahmen zur Beherrschung von Fehlern	64
B.1 Strukturelle Maßnahmen zur Beherrschung von Fehlern auf Systemebene	66
B.2 Maßnahmen zur Beherrschung von Fehlern unterhalb der Systemebene	77
B.3 Nichttechnische Maßnahmen zur Beherrschung von Fehlern	108
Anhang C Checklisten zu den Abschnitten 5.1.1 bis 5.1.3 und A.4	109
C.1 Checkliste für die funktionale Spezifikation	109
C.2 Checkliste für die sicherheitstechnischen Anforderungen	110
C.3 Checkliste für die zu berücksichtigenden Umgebungsbedingungen	111
C.4 Checkliste für die Gestaltungsvorgaben	112
C.5 Checkliste für die Spezifikation der Rahmenbedingungen für Betrieb und Instandhaltung	112
Anhang F Funktionsprüfungen	113
F.1 Prüfung der Funktion des Vormusters bzw. Prototyps	113
F.2 Prüfung der Störfestigkeit	113
F.3 Prüfung der Zerstörfestigkeit	114
F.4 Prüfung der Verträglichkeit gegenüber Umgebungsbedingungen	114
F.5 Einfache Funktionsprüfung	115
F.6 Erweiterte Funktionsprüfung	116
F.7 Funktionsprüfung unter Grenzbedingungen	117
Anhang H Maßnahmen zur Vermeidung von Fehlern bei der Hardware-Entwicklung	118
H.1 Konstruktive Maßnahmen bei der Hardware-Entwicklung	118
H.2 Analytische Maßnahmen bei der Hardware-Entwicklung	122
Anhang O Organisatorische Maßnahmen zur Vermeidung von Fehlern	128
O.1 Projektorganisation	128
O.2 Dokumentation	129
Anhang S Maßnahmen zur Vermeidung von Fehlern bei der Software-Entwicklung	130
S.1 Konstruktive Maßnahmen bei der Software-Entwicklung	130
S.2 Analytische Maßnahmen bei der Software-Entwicklung	139
Anhang X Beispiele für Maßnahmenkombinationen	149
X.1 Grundsätzliche Vorgehensweise	149
X.2 Beispiel 1, Anforderungsklasse 3	150
X.3 Beispiel 2, Anforderungsklasse 5	153
X.4 Beispiel 3, Anforderungsklasse 5	156
X.5 Beispiel 4, Anforderungsklasse 6	159

Bild 1 Zusammenwirken zwischen Rechner (Betrachtungseinheit) und Prozeß	5
Bild 2 Risikoreduzierung durch Nicht-MSR- und MSR-Schutzmaßnahmen	11
Bild 3 Vom Teilrisiko zu den Anforderungen und Maßnahmen	13
Bild 4 Verschiedene Lösungen zur Trennung von Sicherheitsfunktionen und üblichen Prozeßsteuerfunktionen	15
Bild 5 Beispiel für einen Ablaufplan (Phasenmodell) zum Abschnitt 5	19
Bild B.1 (zu Abschnitt B.2.1.1.1)	78
Bild B.2 (zu Abschnitt B.2.1.1.2)	78
Bild B.3 (zu Abschnitt B.2.1.1.4)	79
Bild B.4 (zu Abschnitt B.2.1.1.5)	80
Bild B.5 (zu Abschnitt B.2.1.1.6)	81
Tabelle 1 Wirksamkeit der zu treffenden Maßnahmen gegenüber Fehlern in Abhängigkeit von den Anforderungsklassen	45
Tabelle A.1 Gliederungsschema einer verbindlichen Systemspezifikation	48
Tabelle B.1 Klassifizierungsschema für die in Anhang B beschriebenen Einzelmaßnahmen	65
Tabelle X.1 Geeignete Maßnahmen (außer Basismaßnahmen) zum Beispiel 1, Anforderungsklasse 3	151
Tabelle X.2 Geeignete Maßnahmen (außer Basismaßnahmen) zum Beispiel 2, Anforderungsklasse 5	154
Tabelle X.3 Geeignete Maßnahmen (außer Basismaßnahmen) zum Beispiel 3, Anforderungsklasse 5	157
Tabelle X.4 Geeignete Maßnahmen (außer Basismaßnahmen) zum Beispiel 4, Anforderungsklasse 6	161
Zitierte Normen und andere Unterlagen	165
Weitere Normen	166
Zitierte Literatur	167
Weitere Literatur	171
Erläuterungen	172

1 Anwendungsbereich und Zweck

1.1 Anwendungsbereich

Diese Vornorm gilt für die **funktionale Sicherheit** von Rechnern, die in technischen Systemen Einfluß auf die Abwendung von Gefahren für Leben oder Gesundheit (bei Menschen und Tieren) und für Sachen haben.

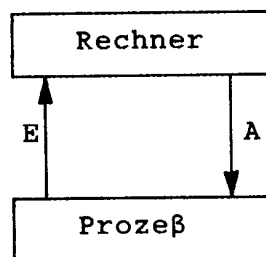
Für die Sicherheit in bezug auf Gefahren, die nicht durch funktionales Versagen verursacht werden, gelten andere Normen.

Diese Vornorm richtet sich in erster Linie an anwendungsbezogen arbeitende Normungsgremien (siehe Vorbemerkung und Erläuterungen).

1.2 Zweck

Rechnern werden in technischen Systemen häufig Sicherheitsaufgaben übertragen. Diese Sicherheitsaufgaben sind im Regelfall sowohl in ihrer Art als auch in ihrer Bedeutung sehr unterschiedlich. Ebenso unterschiedlich sind die Maßnahmen, mit denen die dem Rechner übertragene Sicherheit erreicht werden kann. Diese Vornorm soll in Frage kommende Maßnahmen auflisten und Beispiele aufzeigen, welche Kombinationen von Maßnahmen sich für konkrete Sicherheitsanwendungen eignen.

Bild 1 zeigt schematisch das Zusammenwirken zwischen einem Rechner als Betrachtungseinheit und einem Prozeß. Der Rechner erhält Informationen, z.B. aus dem Prozeß (E), und gibt seinerseits Informationen an den Prozeß (A). Beispielsweise kann der Rechner den Prozeß steuern, regeln oder überwachen. Ebenso kann der Informationsaustausch Rechner – Prozeß unmittelbar durch direkte Verbindung erfolgen oder mittelbar, unter Einbeziehung des Menschen (z.B. Informationseingabe). Generelles Ziel ist es, mit geeigneten Kombinationen von Maßnahmen zu verhindern, daß durch Fehler des Rechners der Prozeß gefährliche Zustände annimmt.



A Ausgaben
E Einnahmen

Bild 1. Zusammenwirken zwischen Rechner (Betrachtungseinheit) und Prozeß

2 Begriffe

Die in dieser Vornorm verwendeten, nachfolgend wiedergegebenen Begriffe wurden, soweit möglich, nicht neu definiert, sondern aus bestehenden Regelwerken übernommen.

2.1 Ausfall (en: failure)

Aussetzen der Ausführung einer festgelegten Aufgabe einer Betrachtungseinheit aufgrund einer in ihr selbst liegenden Ursache und im Rahmen der zulässigen Beanspruchung.
(Aus VDI/VDE 3542 Blatt 1/12.88)

2.1.1 Zufallsausfall (en: random failure)

Ausfall, dessen Eintreten von der Betriebszeit oder von Ausfällen anderer Betrachtungseinheiten statistisch unabhängig ist.
(Aus VDI/VDE 3542 Blatt 1/12.88.)

2.1.2 Weitere Ausfall-Begriffe siehe z.B. VDI/VDE 3542 Blatt 1 und DIN 40041 (z.Z. Entwurf); in vorliegender Vornorm aber nicht verwendet. Siehe jedoch Fehler-Begriffe nach den Abschnitten 2.4 bis 2.4.3.

2.2 Sicherheit (en: safety)

Sicherheit ist eine Sachlage, bei der das Risiko nicht größer als das Grenzkrisiko ist. (Aus DIN VDE 31000 Teil 2/12.87)

2.3 Sicherheitsgerichtet (en: fail-safe)

Fähigkeit eines Systems, beim Auftreten eines Ausfalls im sicheren Zustand zu bleiben oder unmittelbar in einen anderen sicheren Zustand überzugehen.
(Aus VDI/VDE 3542 Blatt 1/12.88, dort jedoch mit der Benennung "fail-safe")

2.4 Fehler (en: fault, defect; auch en: nonconformity)

Nichterfüllung mindestens einer Anforderung an ein erforderliches Merkmal einer Betrachtungseinheit.
(Aus VDI/VDE 3542 Blatt 1/12.88; abweichend: DIN 40041/Entwurf 11.88.)

2.4.1 Systematische Fehler

Fehler in der realisierten Struktur (Hardware oder Software), die auf Denkfehlern ("logische" Fehler) oder auf Ausführungsfehlern (einschließlich Flüchtigkeitsfehlern) beruhen.

2.4.2 Common-mode-Fehler (en: common mode fault)

Systematischer Fehler, bei dem die einzelnen Merkmalabweichungen aufgrund einer gemeinsamen Ursache entstanden sind.

2.4.3 Softwarefehler (en: software error)

Softwarefehler sind Abweichungen zwischen den im Programm realisierten und den beabsichtigten Programmfunktionen.

(Aus DIN VDE 0116 A1/Entwurf 03.87, Abschnitt 2.19)

2.5 Firmware (en: firmware)

Firmware sind meist vom Rechnerhersteller mit Verfahren der Softwaretechnik implementierte Grundfunktionen zur Realisierung der Maschinsprache oder andere Maschinenfunktionen. Firmware ist vom Anwender durch Software-Befehl nicht veränderbar. Für den Anwender erscheint die Firmware wie Hardware.

2.6 Hardware (en: hardware)

Gesamtheit oder Teil der apparativen Ausstattung von Rechensystemen.

Anmerkung 1: Zugehörige Beschreibungen sind nicht Bestandteil der Hardware.

(Aus: DIN 44300 Teil 1/11.88.)

Anmerkung 2: Nach ISO ist Hardware eine materielle Einrichtung in der Datenverarbeitung, im Gegensatz zu Programmen, Prozeduren, Regeln und angegliederter Dokumentation.

(Aus: ITG 6.2/04, Empfehlung 1987.)

2.7 Redundanz (en: redundancy)

Vorhandensein von mehr als für die Ausführung der vorgesehenen Aufgaben an sich notwendigen Mitteln.

(Aus VDI/VDE 3542 Blatt 1/12.88)

2.7.1 Homogene Redundanz (en: homogeneous redundancy)

Redundanz mit gleichartigen Mitteln.

(Aus VDI/VDE 3542 Blatt 1/12.88)

2.7.2 Diversitäre Redundanz (en: diverse redundancy)

Redundanz mit ungleichartigen Mitteln.

Anmerkung 1: Die ungleichartigen Mittel können z.B. sein:

- andere physikalische Prinzipien,
- andere Lösungswege der gleichen Aufgabe,
- andere Aufbauweisen u.s.w.

(Aus VDI/VDE 3542 Blatt 1/ 12.88; abweichend DIN 40041/ Entwurf 11.88)

Anmerkung 2: Weitere Begriffe zur Redundanz siehe z.B. VDI/VDE 3542 Blatt 1/12.88 und DIN 40041/Entwurf 11.88.

2.8 Risiko (en: risk)

Das Risiko, das mit einem bestimmten technischen Vorgang oder Zustand verbunden ist, wird zusammenfassend durch eine Wahrscheinlichkeitsaussage beschrieben, die

- die zu erwartende Häufigkeit des Eintritts eines zum Schaden führenden Ereignisses und
- das beim Ereigniseintritt zu erwartende Schadensausmaß berücksichtigt.

Anmerkung: Das Risiko (R) wird im allgemeinen nicht quantitativ erfaßt; nur selten läßt es sich als Kombination ($\times$) der beiden Größen Häufigkeit (H) des Eintritts und Ausmaß des Schadens (S) quantifizieren: $R = H \times S$.

(Aus DIN VDE 31000 Teil 2, 12.87)

2.9 Robustheit (en: robustness)

Fähigkeit einer Betrachtungseinheit, auch bei Verletzung der spezifizierten Randbedingungen vereinbarte Funktionen zu erfüllen bzw. ihre Funktionsfähigkeit zu erhalten.

Anmerkung 1: Robustheit erstreckt sich somit auf Eigenschaften, für die keine spezifizierten Anforderungen bestehen.

Anmerkung 2: Je umfassender eine Spezifikation auf mögliche Randbedingungen eingeht, desto mehr verliert der Gesichtspunkt Robustheit an Bedeutung.

(Aus: ITG 6.2/04, Empfehlung 1987.)

2.10 Software (en: software)

Gesamtheit oder Teil der Programme für Rechensysteme, wobei die Programme zusammen mit den Eigenschaften der Rechensysteme

- den Betrieb der Rechensysteme,
- die Nutzung der Rechensysteme zur Lösung gestellter Aufgaben oder

– zusätzliche Betriebs- und Anwendungsarten der Rechensysteme ermöglichen.

Anmerkung: Zugehörige Dokumentation kann als Bestandteil der Software angesehen werden.

(Aus: DIN 44300 Teil 1/11.88.)

2.11 Spezifikation (en: specification)

Detaillierte Beschreibung der Teile einer Betrachtungseinheit und ihrer Eigenschaften bezüglich Funktion, Leistung, Qualität und Sicherheitsanforderungen sowie das Zusammenwirken dieser Teile und die getroffenen Annahmen, unter denen die Eigenschaften realisiert werden.

2.11.1 Anforderungsspezifikation (en: requirement specification)

Bei der Planung eines Systems ist die Betrachtungseinheit zunächst das System selbst. Man spricht dann auch von Anforderungsspezifikation; in anderen technischen Regeln zum Teil auch von Pflichten- oder Lastenheft.

2.11.2 Entwurfsspezifikation (en: design specification)

Im Verlauf des Systementwurfs ist die Betrachtungseinheit ein bestimmter Teil des Systems (z.B. die Software), wobei bereits einzelne Entwurfsentscheidungen getroffen wurden. Man nennt dies dann Entwurfsspezifikation.

2.12 Spezifizierte Funktion (en: specified functions)

Teilaufgabe einer Betrachtungseinheit, für die eine detaillierte Beschreibung vorliegt, und die gegenüber den anderen Teilaufgaben der Betrachtungseinheit abgrenzbar ist. Ist diese Abgrenzbarkeit nicht gegeben, so erfüllt die Betrachtungseinheit nur eine einzige spezifizierte Funktion.

2.13 On-line-Test

Ein On-line-Test ist ein Test, der im prozeßgekoppelten Betrieb erfolgt.

2.14 Off-line-Test

Ein Off-line-Test ist ein Test, der außerhalb des prozeßgekoppelten Betriebes erfolgt, z.B. in Instandsetzungs- oder Wartungspausen.

2.15 Betriebsbewährt

Als betriebsbewährt gilt eine Betrachtungseinheit dann, wenn sie im wesentlichen unverändert über einen ausreichenden Zeitraum in zahlreichen, verschiedenen Anwendungen betrieben wurde und dabei keine oder nur unwesentliche Fehler festgestellt wurden.

Anmerkung: Als Synonym zu "betriebsbewährt" wird auch "erprobt" verwendet.

(In Anlehnung an DIN VDE 0116 A1, Entwurf 03.87, Abschnitt 2.26.)

2.16 Korrektheit (en: correctness)

Korrektheit einer Betrachtungseinheit ist die Übereinstimmung der realisierten Betrachtungseinheit mit ihrer Spezifikation.

2.17 Versagen (en: failure)

Siehe Ausfall (Abschnitt 2.1).

2.18 Maßnahmenkombination (en: combination of measures)

Zusammenstellung von Maßnahmen – technischer und nichttechnischer Art -, die in ihrer Gesamtheit geeignet ist, die Fehler entsprechend der jeweiligen Anforderungsklasse abzudecken.

Anmerkung: Begriffe "Maßnahme" und "Anforderungsklasse" siehe DIN V 19250.

3 Grundlegende Sicherheitsbetrachtungen

3.1 Zusammenhang mit dem Grundschemata nach DIN V 19250

In DIN V 19250 wird ein systematisches Verfahren vorgeschlagen, das es erlaubt, anhand qualitativer Risikoabschätzungen die zu stellenden Sicherheitsanforderungen für MSR-Schutzeinrichtungen*) zu ermitteln. Das vorgeschlagene Verfahren ist anwendungs- und technologieunabhängig und kann somit auch auf die MSR-Technologie mit Rechnern übertragen werden, die, soweit sie Sicherheitsrelevanz hat, Gegenstand der vorliegenden Vornorm ist. Da die in DIN V 19250 eingeführten Anforderungsklassen auch als Beurteilungsmaßstab in der vorliegenden Vornorm zugrunde gelegt werden, werden zum Verständnis einige grundsätzliche Zusammenhänge erklärt. Weitergehende Erläuterungen und Beispiele zur Anwendung von DIN V 19250 siehe <Mef89>.

Im Regelfall gilt, daß die sicherheitstechnischen Anforderungen und die Qualität der notwendigen Maßnahmen um so höher sind, je höher das von der MSR-Schutzeinrichtung (im vorliegenden Fall durch den Rechner) abzudeckende Teilrisiko ist. **Bild 2** gibt in etwas vereinfachter Form darüber Auskunft, was unter dem MSR-Teilrisiko verstan-

*) MSR: Messen, Steuern, Regeln

den wird. Ohne MSR-Schutzmaßnahme – weder im eigentlichen Prozeß (Nicht-MSR-Einrichtung) noch in der MSR-Einrichtung – geht von einem sicherheitsbezogenen System ein bestimmtes Risiko aus. Dieses „Risiko ohne Schutzmaßnahme“ ist im Regelfall größer als das vertretbare Risiko, das sogenannte **Grenzrisiko** (siehe DIN VDE 31000 Teil 2). Mit besonderen Maßnahmen muß nun das Risiko mindestens auf das Grenzrisiko reduziert werden. Diese Risikoreduzierung kann durch Nicht-MSR-Schutzmaßnahmen geschehen (z.B. durch Maßnahmen im Prozeß oder im Verfahren) oder aber durch spezielle MSR-Schutzmaßnahmen. Bei den in DIN V 19250 behandelten grundlegenden Sicherheitsbetrachtungen geht es **alleine** um das MSR-Teilrisiko, abgedeckt von MSR-Schutzmaßnahmen. Der Betrachtung liegt also immer ein System (z.B. Maschine, Anlage oder Prozeß) zugrunde, bei dem bereits bestimmte Nicht-MSR-Schutzmaßnahmen getroffen wurden. Die gleichen, hier geschilderten Zusammenhänge ergeben sich für die spezielle Steuerungstechnologie „Rechner“. Es geht also bei den in der vorliegenden Vornorm behandelten Teilrisiken nur um diejenigen **Risiken, die alleine vom Rechner abgedeckt werden müssen**.

Es sei noch darauf hingewiesen, daß die Maßnahmen in der MSR-Schutteinrichtung (hier im Rechner) technischer und nichttechnischer (z.B. organisatorischer) Art sein können und sich gegenseitig ergänzen oder ersetzen können. Das bedeutet, daß die Sicherheit durch verschiedene gleichwertige Wege erreicht werden kann. So kann

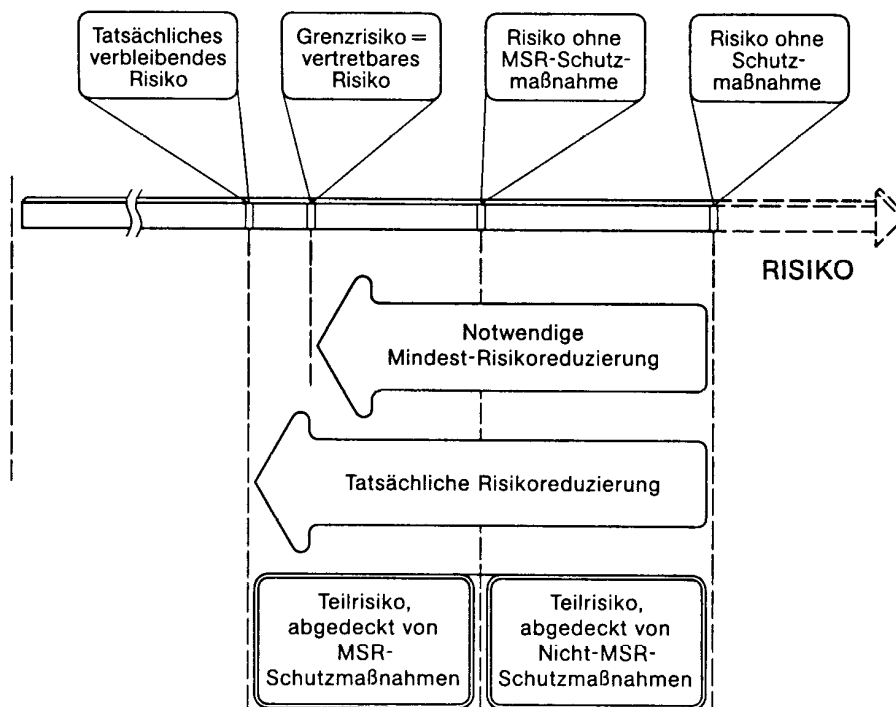


Bild 2. Risikoreduzierung durch Nicht-MSR- und MSR-Schutzmaßnahmen (aus: DIN V 19250/01.89, Bild 1)

beispielsweise eine Lösung mit einem hohen Anteil technischer Maßnahmen gleichwertig sein einer Lösung mit geringem Anteil technischer Maßnahmen, dafür aber einem entsprechend höheren Anteil an nichttechnischen Maßnahmen.

3.2 Besonderheiten der Sicherheitstechnik mit Rechnern

3.2.1 Der grundsätzliche Zusammenhang zwischen dem MSR-Teilrisiko, den Anforderungen an die MSR-Schutzeinrichtung und den Maßnahmen zur Realisierung der Anforderungen ist in **Bild 3a** dargestellt. Das durch die MSR-Schutzeinrichtung abzudeckende MSR-Teilrisiko wird abgeschätzt und beschrieben durch sogenannte Risikoparameter. Aus der Kombination der Risikoparameter ergeben sich direkt die Anforderungen, wobei die Anforderungen in Anforderungsklassen, 1 bis 8, abgestuft sind. Wie dies im einzelnen geschieht, ist in DIN V 19250 festgelegt. **Bild 3b** zeigt den prinzipiell gleichen Zusammenhang, wenn als MSR-Schutzeinrichtung ein Rechner eingesetzt wird.

Weiterhin sind in DIN V 19250 beispielhaft Merkmale genannt, für die Anforderungen an die MSR-Schutzeinrichtung gestellt werden können. Diese Merkmale sind – da sie technologieunabhängig sind – prinzipiell auch auf die Rechnertechnologie übertragbar. Die Merkmale müssen jedoch, damit sie für den Konstrukteur leichter zu berücksichtigen sind, an die Besonderheiten der Rechnertechnologie angepaßt werden.

In der vorliegenden Vornorm wird angenommen, daß die für den konkreten Anwendungsfall erforderliche Anwendungsklasse bereits ermittelt ist. Unter der Voraussetzung einer bekannten Anforderungsklasse werden in der vorliegenden Vornorm die Aspekte nach Aufzählung a) bis d) behandelt:

- a) Welche sicherheitstechnischen Anforderungen sind an den Rechner für jede Anforderungsklasse zu stellen?
- b) Welche Maßnahmen (technisch und nichttechnisch) sind geeignet, die zu jeder Anforderungsklasse gehörenden Anforderungen zu erfüllen?
- c) Welche Maßnahmen gibt es, Fehler, die die Sicherheit beeinträchtigen können, zu **vermeiden**, oder, falls das nicht geht, in ihrer Auswirkung im Betrieb zu begrenzen, d.h. zu **beherrschen**?
- d) Wie können diese Maßnahmen miteinander kombiniert werden, damit sie als gesamtes Paket die in den Anforderungsklassen gestellten Anforderungen erfüllen?

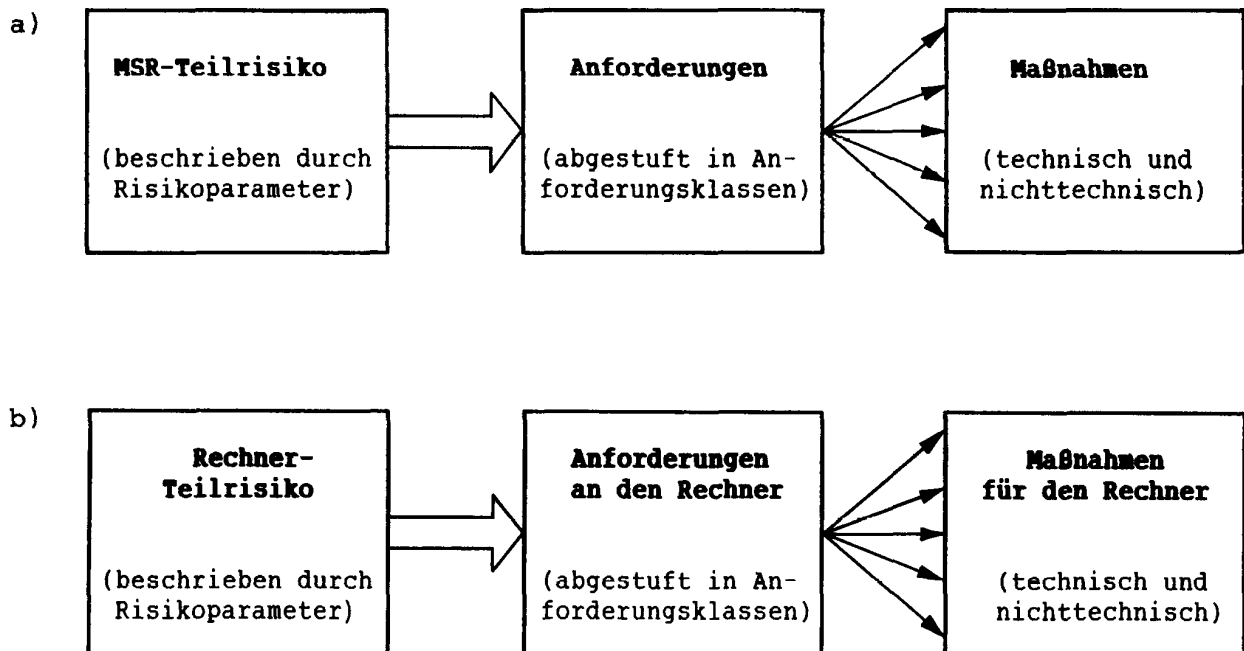


Bild 3. Vom Teilrisiko zu den Anforderungen und Maßnahmen
 a) Für ein MSR-System (aus: DIN V 19250/01.89, Bild 3)
 b) Für den Rechner (als MSR-Einrichtung in konkreter Technologie)

3.2.2 Trennung des Rechnersystems in einen sicherheits- und nicht-sicherheitsbezogenen Teil

In vielen Fällen übernimmt der Rechner in einer Betrachtungseinheit (System) gleichzeitig Sicherheitsfunktionen und reine Prozeßsteuerfunktionen, die mit der Sicherheit nichts zu tun haben. Während es in klassischen Steuerungstechnologien (z.B. Elektromechanik, diskrete Elektronik) möglich war, durch geeigneten Aufbau den sicherheitsbezogenen Teil der Steuerung vom nichtsicherheitsbezogenen Teil der Steuerung zu trennen und dann die Sicherheitsbetrachtungen nur auf den sicherheitsbezogenen Teil anzuwenden, ist dieses Vorgehen bei rechnergesteuerten Systemen schwieriger, teilweise auch gar nicht möglich. Dies muß bei der Projektierung von Rechnersystemen frühzeitig berücksichtigt werden. So kann es beispielsweise dazu kommen, daß ein sehr umfangreiches Programm, das im wesentlichen nichtsicherheitsbezogene Aufgaben erfüllt, einen kleinen Teil enthält, der Sicherheitsfunktionen übernimmt. In diesem Fall muß das

gesamte Programm in vollem Umfange in die Sicherheitsbetrachtung einbezogen werden, wenn nicht der Nachweis gelingt, daß diese Programmteile sich nicht unzulässig beeinflussen. In einigen Fällen kann es deshalb ökonomischer sein – je nach Art und Anteil der sicherheitsbezogenen Aufgabenstellung – den sicherheitsbezogenen Teil der Rechnersteuerung vom nichtsicherheitsbezogenen Teil zu trennen. Der sicherheitsbezogene Teil der Steuerung kann dann entweder in einem getrennten kleineren Rechnersystem realisiert sein oder aber mit anderen Steuerungstechnologien ausgeführt werden.

Bild 4 zeigt verschiedene mögliche Strukturen bei der Kombination von sicherheits- und nichtsicherheitsbezogenen Rechnerteilen.

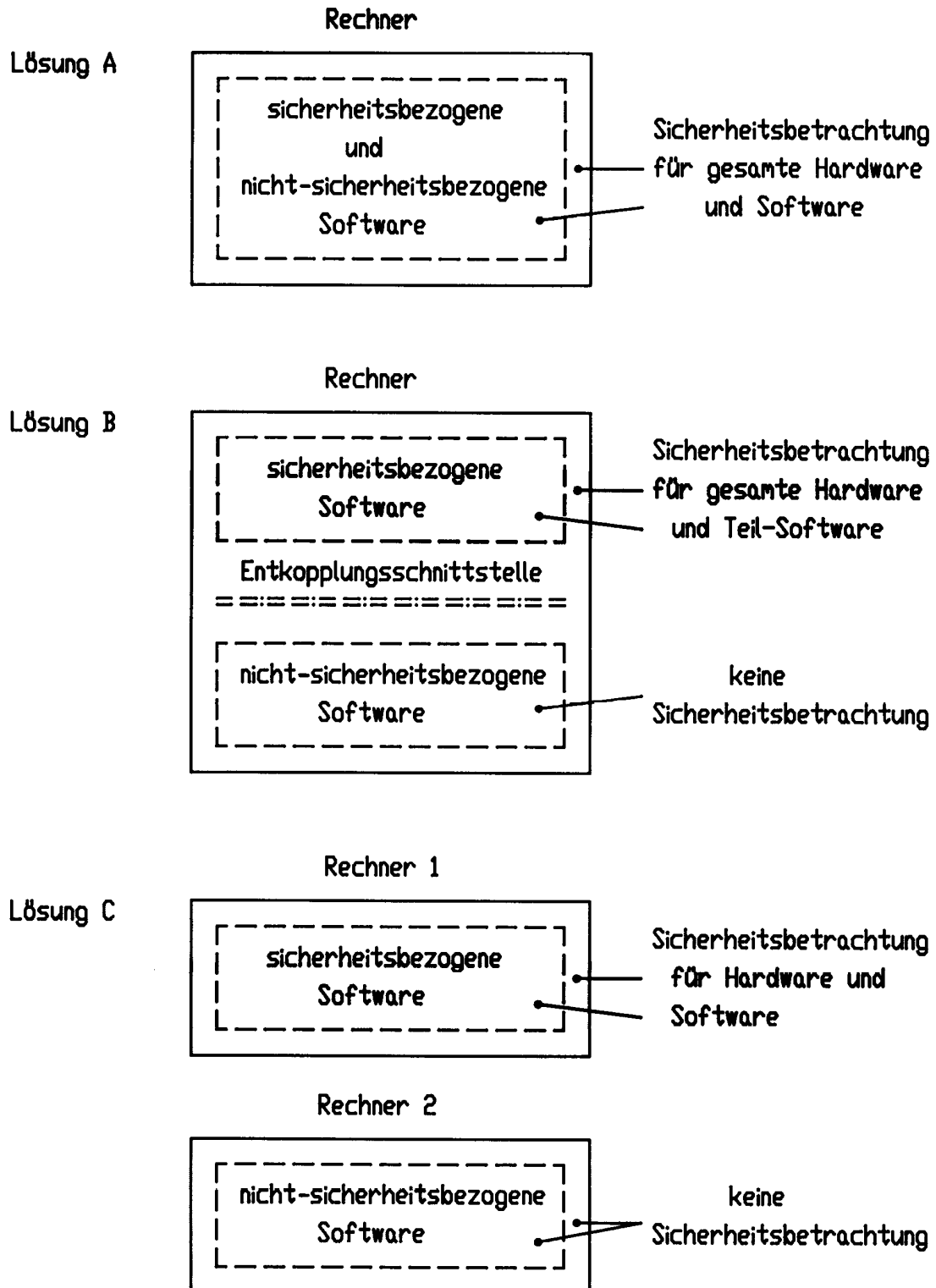


Bild 4. Verschiedene Lösungen zur Trennung von Sicherheitsfunktionen und üblichen Prozeßsteuerfunktionen

3.3 Rechnerspezifische Merkmale, für die Anforderungen gestellt werden

3.3.1 In DIN V 19250 sind beispielhaft Merkmale genannt, für die Anforderungen an die MSR-Schutzeinrichtung gestellt werden, z.B.:

- Fehlerarten
 - o zufällige Fehler
 - o systematische Fehler
- Fehlererkennung
- Fehlerhäufung
- Instandhaltung
- Programmänderung.

3.3.2 Im Abschnitt 4 wird im einzelnen auf Fehlerursachen in Rechnern eingegangen. Dabei fällt auf, daß es im Gegensatz zu den in konventioneller Technik üblichen Fehlerartenannahmen und Fehlerartenausschlüssen bei der Rechnertechnologie nicht mehr sinnvoll ist, alle einzelnen physikalischen Fehlerursachen zu nennen. Der Grund hierfür liegt im wesentlichen in zwei Punkten:

- a) zum einen ist es im Gegensatz zur herkömmlichen Technik nicht ohne weiteres möglich, von der physikalischen Fehlerursache auf die Fehlerauswirkungen zu schließen;
- b) zum anderen verschiebt sich der Schwerpunkt der Fehler durch das Hinzutreten der Software von den Zufallsausfällen zu den systematischen Fehlern.

3.3.3 Aus diesem Grunde werden im Abschnitt 4 die möglichen Fehler in der Soft- und Hardware von Rechnern auch nicht im Detail behandelt, vielmehr wird im wesentlichen auf ihre Art, ihre grundsätzlichen Ursachen und ihren Entstehungszeitpunkt eingegangen. Dabei werden die Fehler in Rechnersystemen im wesentlichen unterschieden nach dem Zeitpunkt ihres Entstehens:

- **Fehler, entstanden bis einschließlich Inbetriebnahme** (z.B. in der Software Spezifikations- und Programmierfehler, oder in der Hardware Fertigungsfehler und falsche Bauteilauswahl)
- **Fehler, entstanden nach der Inbetriebnahme** (z.B. zufällige Fehler in der Hardware oder Fehler durch falsche Nutzung).

Um solche Fehler zu vermeiden oder beim Auftreten solcher Fehler die Auswirkungen zu begrenzen, genügt im Regelfall nicht eine einzige Maßnahme, vielmehr sind hierzu eine ganze Reihe von Maßnahmen notwendig. Die besondere Struktur dieser Vornorm ergibt sich aus der Aufteilung der Maßnahmen in solche zur **Vermeidung** von Fehlern bei Konzeption und Entwicklung, bei den einzelnen Herstellungsstufen, bei Inbetriebnahme und Installation sowie beim Betrieb und bei nachträglichen Änderungen an Hardware und Software (siehe Abschnitt 5) und in solche zur **Beherrschung** von Fehlern während des Betriebs (siehe Abschnitt 6). Diese Maßnahmen zur Fehlervermeidung und Fehlerbeherrschung werden zu **Maßnahmenkombinationen** sinnvollerweise so zusammengestellt, daß sie sich in ihrer Wirkung ergänzen und letztendlich, bezogen auf das Rechner-Teilrisiko des jeweiligen Anwendungsfalls, das verbleibende Risiko auf ein vertretbares Maß reduzieren (Grenzrisiko nach Abschnitt 3.1). Siehe Abschnitt 7.4.

3.4 Zusammenhang zwischen Anforderungsklassen, Fehlerannahmen und Fehlerausschlüssen

Mit den Maßnahmen in den Abschnitten 5 und 6 soll erreicht werden, daß Fehler im Rechner nicht zu Fehlfunktionen führen, die die Sicherheit beeinträchtigen. Das heißt, mit den beschriebenen Maßnahmen soll erreicht werden, daß **bestimmte zu unterstellende Fehler im Rechner mit einer für die betrachtete Anwendung hinreichenden Wahrscheinlichkeit** nicht zu einer gefährlichen Fehlfunktion führen. Dabei gilt qualitativ folgender allgemeine Zusammenhang: Je höher das abzudeckende Rechner-Teilrisiko beim Versagen des Rechners ist,

- desto höher sind die zu stellenden Anforderungen,
- desto höher ist die erforderliche Anforderungsklasse (siehe DIN V 19250),
- desto mehr Fehlerarten werden unterstellt (Fehlerannahmen) und
- desto weniger Fehlerartenausschlüsse (Fehlerausschlüsse) werden gemacht.

4 Fehlerursachen in Rechnern

Fehler dürfen in Rechnersystemen mit Sicherheitsaufgaben nicht zu einer Gefahr führen. Deshalb müssen geeignete Maßnahmen getroffen werden, solche Fehler zu vermeiden, d.h. Fehler treten erst gar nicht auf (siehe Abschnitt 5). Wo dies nicht in ausreichendem Maße möglich ist, müssen diese Fehler im Betrieb beherrscht werden, d. h. bei ihrem Auftreten muß das rechnergesteuerte System ein bestimmtes sicherheitstechnisches Verhalten aufweisen (siehe Abschnitt 6).

In welchem Umfang Maßnahmen zu Fehlervermeidung und Fehlerbeherrschung ergriffen werden müssen, hängt in großem Maße vom jeweils vorliegenden Risiko des Anwendungsfalls und den technischen Gegebenheiten ab, siehe Abschnitt 3 und Erläuterungen.

4.1 Unterscheidung der Fehler nach ihrem zeitlichen Auftreten

4.1.1 Fehler, entstanden bis einschließlich Inbetriebnahme

Im Ablaufplan nach **Bild 5** sind die zeitlich aufeinander folgenden Schritte für die Planung und Realisierung eines rechnergesteuerten Systems beispielhaft dargestellt. In allen Einzelphasen, von der Aufgabenstellung über die Entwicklung bis einschließlich Inbetriebnahme, können die Fehler nach Abschnitt 4.2 entstehen.

Der in Bild 5 der Einfachheit halber linear dargestellte Ablaufplan ist in der Praxis ein iterativer Prozeß mit Schleifen, d.h. festgestellte Abweichungen haben im Regelfall Änderungen in vorhergehenden Phasen oder Schritten zur Folge. Es ist zu beachten, daß bei Änderungen der Aufgabenstellung ein Neueintritt in den Ablaufplan ab Abschnitt 5.1 erforderlich ist.

Der Ablaufplan enthält auch Schritte, die im Einzelfall entbehrlich sein können, z.B. bei Einzelprodukten.

Dem Hersteller eines Rechnersystems stehen eine ganze Reihe von Möglichkeiten an qualitätssichernden Maßnahmen zur Verfügung, um vorhandene und im Entwicklungs- und Fertigungsprozeß entstehende Fehler rechtzeitig vor der Übergabe an den Benutzer des Systems zu entdecken und zu beheben. Konstruktive Maßnahmen zur Reduzierung der Wahrscheinlichkeit zufälliger Fehler sind möglich und müssen ggf. von der Konzeptphase bis zur Auslieferung berücksichtigt werden. Maßnahmen zur Fehlervermeidung werden im Abschnitt 5 behandelt. Wegen der Komplexität der Hard- und Software wird es nicht immer möglich sein, alle im Rechner vorhandenen Fehler aufzudecken und zu beheben, so daß auch nach der Inbetriebnahme noch Fehler im Rechnersystem vorhanden sein können. Diese unentdeckten Fehler sind überwiegend deterministisch und können später im Betrieb zu systematischen Ausfällen und Mehrfachausfällen führen.

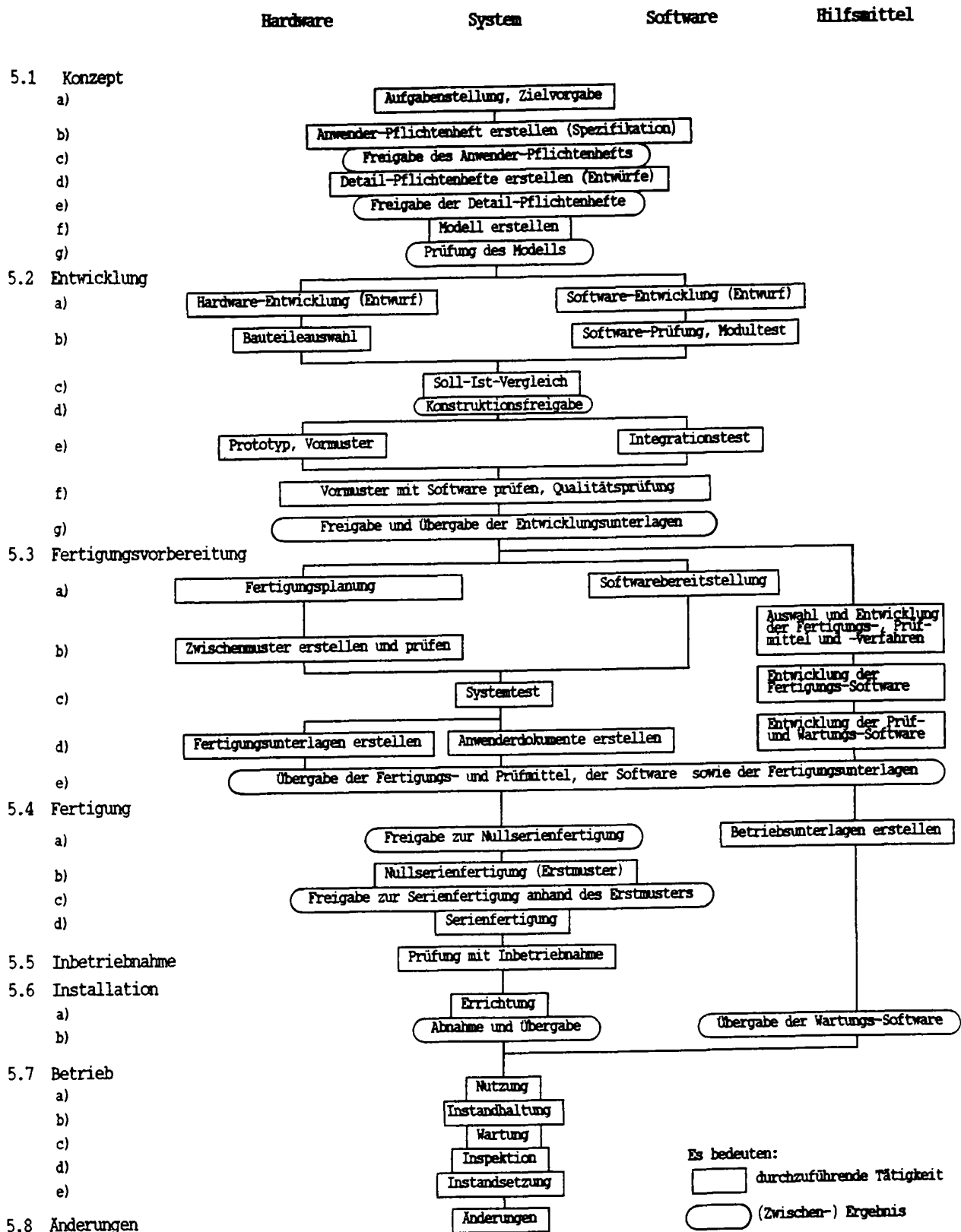


Bild 5. Beispiel für einen Ablaufplan (Phasenmodell) zum Abschnitt 5 (Benummerung bezieht sich auf Abschnitt 5)

Beispiele unentdeckter Fehler sind:

- a) Spezifikationsfehler,
- b) bitmusterabhängiges Fehlverhalten von hochintegrierten Schaltkreisen,
- c) Programmierfehler,
- d) falsche Bauteilauswahl und -dimensionierung,
- e) Fertigungsfehler.

Dort, wo die ergriffenen Maßnahmen zur Vermeidung solcher Fehler nicht ausreichen, müssen diese Fehler bei ihrem Auftreten im Betrieb beherrscht werden. Die entsprechenden Maßnahmen sind in Abschnitt 6 beschrieben.

4.1.2 Fehler, entstanden **nach** der Inbetriebnahme

Nach der Inbetriebnahme, also nach der Übergabe des rechnergesteuerten Systems an den Benutzer, können zusätzlich folgende Fehler entstehen:

- zufällige Fehler in der Hardware,
- Fehler, die durch eine falsche Nutzung, Instandhaltung oder durch nachträgliche Änderung entstehen.

Die zufälligen Fehler in der Hardware können ihrer Natur nach nicht verhindert, sondern nur, was die Wahrscheinlichkeit ihres Auftretens betrifft, reduziert werden. Die bei der Nutzung auftretenden zufälligen Fehler müssen, je nach Höhe des vorliegenden Risikos beherrscht werden, d.h. sie dürfen nicht zu einer Gefahr führen.

4.2 Fehlerarten

Ausgehend von dem Fehlermodell (siehe Abschnitt 3.3.3) werden die Fehler in Fehlerarten eingeteilt. Diese Fehlerarten werden im wesentlichen aus der Art und Weise ihrer Entstehung (Ursachen) und der möglichen Maßnahmenart (Vermeidung oder/und Beherrschung), mit denen diesen Fehlerarten begegnet werden kann, abgegrenzt.

Anmerkung: Für verschiedene Anwendungen sind Fehlerlisten erarbeitet worden, siehe z.B. <BIA87>, <DB> und <ORE>.

4.2.1 Systematische Fehler

Systematische Fehler sind von Anfang an in einer Komponente eines Rechners vorhanden. Sie bleiben jedoch im Regelfall im Betrieb des Rechners längere Zeit unerkannt und wirken sich erst bei bestimmten Zuständen oder in bestimmten Betriebssituationen aus und können damit zu einem Versagen führen. Diese Fehlerart ist in allen Exemplaren oder einer großen Anzahl von Exemplaren einer Serie vorhan-

den. Die Entstehungsursache des Fehlers ist "systematischer" Natur, d.h. der Fehler geht auf eine grundsätzlich bestimmbare und reproduzierbare Ursache zurück (deterministische Ursache).

Systematische Fehler sind prinzipiell vermeidbar, wenn auch in der Praxis für bestimmte Fehler eine vollständige Vermeidung nur schwer oder nur mit sehr hohem Aufwand zu erreichen ist.

4.2.1.1 Systematische Fehler in der Hardware

Diese Fehler sind allgemeine Konstruktionsfehler, Entwurfsfehler, Fertigungsfehler und Fehler, die bei Prüfungs- und Inbetriebnahmearbeiten entstehen können.

Typische Ursachen sind z.B.

- Logische Maskenfehler von integrierten Schaltkreisen (IC),
- Auslegungsfehler bei Taktversorgung, Bus-timing usw.,
- nichtbestimmungsgemäßer Einsatz von Bauelementen,
- fehlerhafte Signalpegel aufgrund fehlerhaft ausgelegter Stromversorgung,
- Entflechtungsfehler auf Platinen oder
- Kurzschlüsse, Unterbrechungen und andere Defekte, verursacht bei der Fertigung oder bei der Prüfung.

4.2.1.2 Systematische Fehler in der Software

Diese Fehler entstehen bei Spezifikation und Entwurf von Software, beim Programmwurf und bei der Implementierung. Werden diese Fehler bei Review und Test nicht aufgedeckt, sind sie als systematische Fehler in der Software des Rechners von Anfang an existent.

Da Software weder einem physikalischen Verschleiß noch zufällig auftretenden Defekten unterliegt, sind Softwarefehler aus sicherheitstechnischer Sicht ausschließlich systematische Fehler. In der Praxis werden zwar aus der Erfahrung gewisse Häufigkeitsbetrachtungen für die Entstehung von Softwarefehlern angestellt (z.B. Fehler pro 1 K Statements einer bestimmten Programmiersprache), die Auswirkungen von Softwarefehlern im Betrieb des Rechners sind jedoch keine statistisch unabhängigen Ereignisse.

4.2.2 Zufällige Fehler in der Hardware (Ausfälle)

Zufällige Fehler ("Zufallsfehler") sind alle Defekte bzw. Ausfälle der Hardware, die im Betrieb des Rechners entstehen und die statistisch mit einer bestimmten Lebensdauerverteilung der Bauelemente

und Komponenten auftreten. Für die Bauelemente eines Rechners gilt dabei im wesentlichen die Exponentialverteilung des Ausfallverhaltens elektronischer Bauelemente mit konstanter Ausfallrate.

Aber auch Fehler mit anderer statistischer Verteilung, z.B. die Gauß'sche Normalverteilung bei Verschleißausfällen (Floppy- bzw. Hard-Disk-Laufwerke), gehören zu den Zufallsfehlern.

Wesentliches Merkmal der Zufallsfehler ist, daß einzelne Fehlerursachen, d.h. "Einzelfehler", statistisch unabhängig von anderen Einzelfehlern sind. Umgekehrt heißt dies, daß eine Fehlerursache und alle daraus resultierenden Fehlerwirkungen oder Folgefehler als Einzelfehler anzusehen sind (z.B. defektes Flag-Register --> falsche Operation --> falsches Datum im Speicher --> usw.).

4.2.3 Handhabungsfehler

Unter Handhabungsfehlern werden alle Fehler verstanden, die beim Umgang mit dem Rechner nach Auslieferung im Anwendungsfall ("vor Ort") verursacht werden. D.h. Fehler, die durch die Handhabung des Rechners beim Anlauf, beim Software-Laden, bei Bedienung und Beobachtung (Dialog), bei allen Tätigkeiten der Instandhaltung, insbesondere der Instandsetzung (Reparatur), beim Testen vor Ort und bei Systemänderungen in Hardware oder Software entstehen können.

4.2.3.1 Unbeabsichtigte Handhabungsfehler

Hierunter fallen alle Fehler, die auf Flüchtigkeit oder Irrtum beruhen. Dies betrifft besonders Bedienungen im Betrieb, die routinemäßig erfolgen, oder Bedienungen, die in Ausnahmesituationen unter Streß durchgeführt werden müssen.

4.2.3.2 Fehler durch Manipulation

Fehler durch Manipulation werden bewußt begangen, jedoch ohne eine besondere kriminelle Absicht. Es sind Handhabungen, die zwar gegen entsprechende Richtlinien oder den Sachverstand verstoßen, mit denen in der Praxis aber aufgrund der betrieblichen Erfahrung oder der Kenntnis von Gewohnheiten gerechnet werden muß.

Eine Sabotage, d.h. eine Manipulation mit krimineller Absicht, wird aus der Betrachtung ausgeklammert (siehe auch DIN V 19250).

4.2.3.3 Instandhaltungsfehler

Instandhaltungsfehler sind Fehler, die bei Inspektion, Wartung, im wesentlichen jedoch bei der Instandsetzung (Reparatur) versehentlich neu in den Rechner eingebracht werden können, wenn die Arbeiten der Instandhaltung nicht mit größter Sorgfalt ausgeführt werden. Typisch sind z.B. die Vertauschung von Signalleitungen beim Wiederanklemmen, Verursachung von Defekten auf den Baugruppen u.a., falsche Brücken-Einstellungen auf erneuerten Baugruppen usw.

4.2.4 Fehler durch Betriebs- und Umgebungseinflüsse

Zu dieser Fehlerart gehören alle Fehler, die im Betrieb des Rechners auftreten, deren Ursachen jedoch in externen Ereignissen liegen, die von außen die Hardware des Rechners beeinflussen.

Hierzu gehören alle Umgebungseinflüsse, die sowohl von der Umwelt als auch von benachbarten anderen Systemen ausgehen können. Typische Beispiele sind Temperatur, elektromagnetische und elektrostatische Einflüsse, mechanischer Stoß und Schwingung.

4.2.4.1 Einflüsse mit vorübergehender Fehlerwirkung (Störung)

Dies sind Störungen der Rechnerfunktion, die zeitweise zu Fehlern bzw. Fehlfunktionen führen. Der Fehler ist nicht von bleibender Dauer; z.B. Störung durch elektromagnetische Einstreuung.

4.2.4.2 Einflüsse mit bleibender Fehlerwirkung (Zerstörung)

Dies sind Einflüsse, die zu einem bleibenden Fehler (Ausfall) irgendeines Bauteiles im Rechner führen; z.B. Kurzschluß durch Übertemperaturstoß, Leitungsbruch durch mechanische Schwingung usw.

5 Maßnahmen zur Vermeidung von Fehlern

Entsprechend dem im **Bild 5** beispielhaft dargestellten Ablaufplan werden die Maßnahmen zur Fehlervermeidung den jeweiligen Phasen zugeordnet (siehe Abschnitte 5.1 bis 5.9). Dabei können einzelne Maßnahmen in mehreren Phasen wirksam sein.

Die Vermeidung von Fehlern setzt voraus, daß sowohl beim Hersteller von Rechnern mit Sicherheitsverantwortung, als auch beim Benutzer Qualitätssicherungssysteme bestehen. Unterlieferanten des Herstellers und Wartungspersonal des Benutzers sind einzubeziehen.

Die Qualitätssicherung wird von allen beteiligten Stellen wahrgenommen. Hinweise zur Organisation eines Qualitätssicherungssystems sind z.B. in DIN ISO 9004 enthalten. Begriffe der Qualitätssicherung siehe z.B. Normen der Reihe DIN 55350 und DIN V 66285. Die Organisation ist im Einzelfall der Struktur des jeweiligen Unternehmens angepaßt.

Alle im folgenden ausgeführten Bezüge auf Aufgabenzuordnungen, insbesondere hinsichtlich Mustererstellung, Fertigung und Qualitätssicherung, sind nur beispielhaft zu verstehen. Die tatsächliche Organisation muß sich im Einzelfall nach den jeweiligen betrieblichen Erfordernissen richten.

5.1 Vermeidung von Fehlern in der Konzeptphase

5.1.1 Bedeutung der Konzeptphase für die Entwicklung eines sicheren Systems

In der Konzeptphase werden die Arbeitsgrundlagen für die nachfolgende Entwicklung erstellt. Fehler in diesen Arbeitsgrundlagen sind danach frühestens beim Integrationstest erkennbar. Die nachträgliche Beseitigung ist also kostspielig. Bleiben sie unerkannt, so führt dies zu systematischen Ausfällen, die nur mit sehr aufwendigen Methoden beherrschbar sind. In der Konzeptphase ist zu klären, ob die geforderte Sicherheit stärker durch Beseitigen aller Fehlerquellen (perfektionistischer Ansatz) oder durch Beherrschen der Fehler im Betrieb (fehlertoleranter Ansatz) erreicht werden soll. Es muß also das sicherheitstechnische Konzept festgelegt werden, da die nachfolgende Entwicklung davon abhängt. So müssen grundlegende Maßnahmen zur Fehlertoleranz bereits in der Konzeptphase festgelegt werden.

5.1.1.1 Bedeutung der Spezifikation für Systeme mit Sicherheitsverantwortung

Die Umschreibung des Begriffes "Fehler" als unerwünschtes Verhalten eines Systems unterstellt, daß das erwünschte Verhalten eindeutig bekannt ist und somit eine Abweichung des tatsächlichen Verhaltens vom erwünschten Verhalten eindeutig feststellbar ist. Das gewünschte Verhalten eines Systems ist in der **Anforderungsspezifikation** dokumentiert. Da es insbesondere für die Software-Komponente eines Systems keine Methode gibt, die Korrektheit in anderer Weise als bezüglich der Spezifikation festzustellen, muß diese in Systemen mit Sicherheitsverantwortung als gesichert und verlässlich gelten. Diese wird daher im folgenden als **"Verbindliche Systemspezifikation"** (en: Authoritative System Reference) bezeichnet (siehe z.B. DIN VDE 0116 A1, z.Z. Entwurf, <ROB82>, <KOP82>). Ihre Erstellung muß mit besonderer Sorgfalt entsprechend den Grundsätzen für Systeme mit Sicherheitsverantwortung vorgenommen werden, die im nachfolgenden dargelegt sind.

Alle Abweichungen gegenüber der Verbindlichen Systemspezifikation gelten als Fehler. Sollte es sich als notwendig erweisen, diese zu ändern, so gilt das ursprüngliche System als fehlerhaft. In diesem Fall ist die Verbindliche Systemspezifikation von Anfang an zu korrigieren und zu überprüfen. Ist für ein System der Sicherheitsnachweis einer Prüfstelle erforderlich, so ist die Verbindliche Systemspezifikation von der zuständigen Prüfstelle zu genehmigen. Normen für die besonderen Anwendungsbereiche können Vorgaben über den Inhalt, z.B. die Festlegung von Parametern, Verfahren und ähnlichem, machen.

Damit eine Verbindliche Systemspezifikation als gesichert und verlässlich gelten kann, müssen für den Inhalt und die Methoden zur Erstellung gewisse Regeln eingehalten werden. Eine mangelhafte

Spezifikation führt im Regelfall zu Mängeln am endgültigen Produkt, ohne daß dies in den nachfolgenden Entwurfs- und Herstellungsphasen erkannt wird. Die Erstellung muß daher entsprechend dem Risiko mit höchster Akribie erfolgen.

In der Verbindlichen Systemspezifikation sind die Anforderungen an das System vollständig niedergelegt. Vorgaben für die Implementierung sind nur so weit zu machen, wie sie aufgrund von technischen oder organisatorischen Gegebenheiten die Freiheit der Implementierung einengen. Die sicherheitstechnischen Anforderungen können beim derzeitigen Stand der Technik nicht in Form eines zulässigen Restrisikos nachprüfbar spezifiziert werden. Es ist vielmehr erforderlich, ein sicherheitstechnisches Konzept darzulegen, das nachprüfbar ist und beim derzeitigen Stand der Technik als angemessen angesehen wird, auch wenn das verbleibende Restrisiko nicht exakt berechenbar ist. Die sicherheitstechnischen Anforderungen werden daher im allgemeinen Vorgaben für die Implementierung enthalten.

5.1.1.2 Entwurf und Prüfung eines Realisierungskonzepts

Nach Erstellen der Spezifikation ist ein Realisierungskonzept zu erarbeiten, das grundlegende Implementierungsentscheidungen enthält.

Der Grad der Detaillierung soll dabei so weit gehen, daß beurteilt werden kann, ob die sicherheitstechnischen und funktionalen Anforderungen auch tatsächlich erreichbar sind.

Da der Vorgang ähnlich dem bei der Erstellung der Spezifikation ist, sind zur Fehlervermeidung prinzipiell die gleichen Methoden anzuwenden. Der Grad der Detaillierung ist jedoch höher. Eine Zusammenfassung der Methoden enthält Abschnitt 5.1.3.

Die Prüfung des Realisierungskonzeptes kann an Hand der erarbeiteten Unterlagen erfolgen. In manchen Fällen erweist es sich jedoch als zweckmäßig, zur Beurteilung ein Modell zu erstellen und daran grundlegende Entwurfsentscheidungen zu überprüfen ("Prototyping"). Dies sollte insbesondere dann erfolgen, wenn die leichte Handhabbarkeit oder allgemein die Mensch-Maschine-Schnittstelle von sicherheitstechnischer Bedeutung ist.

5.1.2 Einzelheiten zu den Ergebnissen der Konzeptphase

5.1.2.1 Inhalt einer Verbindlichen Systemspezifikation

Die Verbindliche Systemspezifikation soll Angaben über folgende Punkte enthalten

- funktionale Spezifikation,
- sicherheitstechnische Anforderungen,
- zu berücksichtigende Umgebungsbedingungen,

- Gestaltungsvorgaben und
- Rahmenbedingungen für Betrieb und Instandhaltung.

Die funktionale Spezifikation enthält alle Angaben über die auszuführenden Funktionen und die Schnittstellen nach außen. Die sicherheitsbezogenen Funktionen müssen untereinander und gegenüber den nicht sicherheitsbezogenen Funktionen entkoppelt sein, d.h. es müssen klare, einfache Schnittstellen existieren. Neben der Beschreibung der originären Sicherheitsaufgabe müssen auch die gewünschten Reaktionen auf Störungen oder Fehler angegeben sein. Zur Herleitung von Plausibilitätsprüfungen sollen Angaben über zulässige Wertebereiche, Änderungsgeschwindigkeiten von Werten, zulässige Kombinationen von Bedingungen u.ä. gemacht werden.

In den sicherheitstechnischen Anforderungen sind ggf. auch Fehler des Betriebspersonals zu betrachten. Solche Fehler können unbeabsichtigt oder beabsichtigt begangen werden. Es ist darzulegen, inwieweit beabsichtigte Fehler, z.B. durch übergeordnete Maßnahmen, wie Personalauswahl und Personalüberwachung auszuschließen sind. Ggf. sind auch Störmöglichkeiten durch betriebsfremdes Personal zu betrachten. Organisatorische Maßnahmen, die in diesem Zusammenhang erforderlich sind, um bestimmte Fehlermöglichkeiten auszuschließen, sind zu dokumentieren.

Eine Verbindliche Spezifikation kann nur gültig sein, wenn sie die folgenden vier **Zielsetzungen** erfüllt

- Vollständigkeit,
- Widerspruchsfreiheit,
- Fehlerfreiheit und
- Prüfbarkeit

und wenn zwischen allen Beteiligten (einschließlich der zuständigen Prüfstelle) eine Übereinkunft über Umfang und Inhalt besteht.

5.1.2.2 Beurteilen eines Realisierungskonzeptes anhand eines Modells

Wird ein Modell zur Beurteilung des Realisierungskonzeptes herangezogen, so ist zu beachten, daß die Modellerstellung erheblich weniger Kosten verursachen darf als der Bau eines Prototyps. Trotz der daraus resultierenden erheblichen Vereinfachung sollen die zu beurteilenden Merkmale des Realisierungskonzeptes im Modell korrekt abgebildet werden. Soll in erster Linie die Mensch-Maschine-Schnittstelle beurteilt werden, so sind die Handhabungselemente und Anzeigen hinsichtlich Art und Anordnung korrekt nachzubilden. Der Benutzerdialog ist zu simulieren, wobei die aufgeführten einzelnen Dialogschritte für eine spätere Auswertung aufgezeichnet werden. Anhand typischer Benutzungsabläufe kann dann beurteilt werden, ob Benutzung und Betrieb des Systems gefahrlos möglich ist, und ob ergonomische Gesichtspunkte ausreichend berücksichtigt sind.

In anderen Fällen wird ein Modell erstellt, um festzustellen, ob der Betrieb des Systems eine Gefährdung für die Umgebung, z.B. durch hervorstehende oder sich bewegende Teile, darstellt, bzw. ob Gefahrenbereiche ausreichend gesichert sind. Hier sollen durch das Modell im wesentlichen die räumlichen Verhältnisse und Bewegungsabläufe nachgebildet werden. Für die Modellerstellung ist es also wesentlich, welcher Aspekt des Realisierungskonzeptes beurteilt werden soll.

5.1.3 Methoden zur Verwendung und Erstellung einer Verbindlichen Systemspezifikation

Aus Abschnitt 5.1.1.1 folgt, daß eine grundlegende Methode zur Vermeidung von Fehlern in der Konzeptphase die Verwendung einer Verbindlichen Systemspezifikation darstellt.

Die vier Zielsetzungen nach Abschnitt 5.1.2.1 lassen sich im Regelfall nur erfüllen, wenn formale Darstellungsmittel benutzt werden. Die Verwendung von rechnergestützten Werkzeugen zwingt zu einer formalen Darstellung, so daß deren Verwendung bei hohen Sicherheitsanforderungen dringend empfohlen wird. Die Methoden betonen verschiedene Aspekte bei der Erstellung einer Spezifikation; sie lassen sich teilweise miteinander kombinieren.

Die nachfolgend aufgeführten Einzelmaßnahmen zur Vermeidung von Fehlern werden in **Anhang A** detailliert behandelt:

- a) Verwenden einer Verbindlichen Systemspezifikation, siehe Abschnitt A.1,
- b) Verwendung von rechnergestützten Entwicklungswerkzeugen, siehe Abschnitt A.2,
- c) Strukturierter Entwurf, siehe Abschnitt A.3,
- d) Verwendung von Checklisten, siehe Abschnitt A.4,
- e) Inspektionen der Spezifikation, siehe Abschnitt A.5,
- f) Benutzerfreundlichkeit, siehe Abschnitt A.6,
- g) Instandhaltungsfreundlichkeit, siehe Abschnitt A.7.

5.2 Vermeidung von Fehlern bei der Entwicklung

5.2.1 Einführung

5.2.1.1 Zur Vermeidung von Fehlern bei der Entwicklung von Systemen mit Sicherheitsaufgaben müssen zahlreiche einzelne **Maßnahmen** durchgeführt werden, die für eine bessere Übersicht folgendermaßen in Gruppen eingeteilt werden:

- a) Die **organisatorischen Maßnahmen** betreffen das Projektmanagement, die Organisation der Qualitätssicherung und die Dokumentation. Mit ihnen werden die Rahmenbedingungen für die Entwicklung sicherheitsrelevanter Systeme geschaffen.
- b) Die **technischen Maßnahmen** betreffen die Verfahren der Entwicklung, die Bereitstellung und den Einsatz von geeigneten Einrichtungen, Hilfsmitteln und Materialien, die Qualitätssicherung und Prüfung der Bauelemente, Baugruppen, Funktionseinheiten und der sicherheitsrelevanten Software. Sie bilden den Kern der Maßnahmen zur Fehlervermeidung.

5.2.1.2 Während die organisatorischen Maßnahmen bei der Hardware- bzw. Software-Entwicklung prinzipiell gleich sind, sind die technischen Maßnahmen zur Vermeidung von Hardware- bzw. Software-Fehlern sehr unterschiedlich. Sie werden – für Hardware und Software getrennt – jeweils den zwei folgenden Gruppen zugeordnet, die sich in der Vorgehensweise grundsätzlich unterscheiden <Hei79>, <TrV83>:

- a) Man ergreift während der Entwicklung **konstruktive Maßnahmen**, um von vornherein Fehler zu vermeiden.
- b) Man ergreift **analytische Maßnahmen**, um Fehler aufzudecken.

5.2.1.3 Es ist zu unterscheiden zwischen der Verifikation und der Validation. Ziel der Verifikation ist zu zeigen, daß eine Funktionseinheit ihre Spezifikation erfüllt, bzw. Widersprüche zur Spezifikation aufzudecken. Bei der Validation wird das gesamte System unter Berücksichtigung der realen Einsatzbedingungen gegenüber den Erfordernissen geprüft (siehe ITG 6.2/04).

5.2.1.4 Die Maßnahmen nach Abschnitt 5.2.3 zur Vermeidung sicherheitsbezogener Fehler haben jeweils für sich gesehen nur eine eingeschränkte Mächtigkeit. Keine Maßnahme ist so mächtig, daß man sich auf sie allein verlassen kann. Jede Maßnahme verlangt ihren gezielten, durchdachten und kontrollierten Einsatz.

Es dürfen auch nicht Maßnahmen der einen Gruppe, z.B. analytische Maßnahmen, gegenüber Maßnahmen der anderen Gruppe, z.B. konstruktiven Maßnahmen, vernachlässigt werden. Die Wirksamkeit der Maßnahmen wird insbesondere durch die gezielte Kombination von Maßnahmen erhöht. Beispielsweise ist der Einsatz bestimmter konstruktiver Maßnahmen für die Effektivität der anschließenden analytischen Maßnahmen besonders wichtig, da konstruktive Maßnahmen neben dem primären Ziel, Fehler von vornherein zu vermeiden, die Prüfbarkeit der Systeme sicherstellen bzw. verbessern sollen.

5.2.2 Fehlerursachen

Als Fehlerursachen müssen systematische Fehler bei der Hardware- und Software-Entwicklung und (Hardware-) Ausfälle berücksichtigt werden.

Ausfälle können im Gegensatz zu systematischen Fehlern grundsätzlich nicht vermieden werden; allerdings kann durch bestimmte Maßnahmen die Häufigkeit von Ausfällen reduziert werden.

Auf eine feinere Unterteilung der Fehlerursachen wird für diese Zusammenstellung verzichtet, da auch einzelne der im folgenden beschriebenen Maßnahmen häufig gegen eine Vielzahl von Fehlerursachen wirken, und da die im vorhergehenden Abschnitt begründete Gliederung der Maßnahmen schlüssiger ist als eine Gliederung nach einzelnen Fehlerursachen.

5.2.3 Maßnahmen

Die nachfolgend aufgeführten Einzelmaßnahmen zur Vermeidung von Fehlern werden in den **Anhängen F, H, O und S** detailliert behandelt.

5.2.3.1 Konstruktive Maßnahmen bei der Hardware-Entwicklung

- a) Beachtung der zutreffenden technischen Regeln, siehe Abschnitt H.1.1,
- b) Verwendung rechnergestützter Konstruktions- und Entwicklungssysteme, siehe Abschnitt H.1.2,
- c) strukturierter Entwurf, siehe Abschnitt H.1.3,
- d) Spezifikation von hoher Qualität, siehe Abschnitt H.1.4,
- e) Baueinheitenauswahl, siehe Abschnitt H.1.5.

5.2.3.2 Analytische Maßnahmen bei der Hardware-Entwicklung

- a) Entwurfsprüfung durch Inspektion, siehe Abschnitt H.2.1,
- b) Entwurfsprüfung durch Walk-through, siehe Abschnitt H.2.2,
- c) statische Analyse, siehe Abschnitt H.2.3,
- d) dynamische Analyse, siehe Abschnitt H.2.4,
- e) Funktionsanalyse, siehe Abschnitt H.2.5,
- f) Simulation, siehe Abschnitt H.2.6,
- g) Berechnung von Ausfallgrößen, siehe Abschnitt H.2.7,
- h) Ausfalleffektanalyse, siehe Abschnitt H.2.8,
- i) Funktionstest, siehe Abschnitt H.2.9.

5.2.3.3 Konstruktive Maßnahmen bei der Software-Entwicklung

- a) Einhaltung von Richtlinien zur Erstellung von Software, siehe Abschnitt S.1.1,
- b) Verwendung rechnergestützter Entwurfs- und Software-Produktionssysteme, siehe Abschnitt S.1.2,

- c) Spezifikation von hoher Qualität, siehe Abschnitt S.1.3,
- d) strukturierter Entwurf, siehe Abschnitt S.1.4,
- e) Modularisierung, siehe Abschnitt S.1.5,
- f) Verwendung höherer Programmiersprachen, siehe Abschnitt S.1.6,
- g) strukturierte Programmierung, siehe Abschnitt S.1.7,
- h) Verwendung besonderer Systemsoftware, siehe Abschnitt S.1.8,
- i) Verzicht auf Optimierungen, siehe Abschnitt S.1.9,
- j) Minimierung von Realzeiteinflüssen, siehe Abschnitt S.1.10,
- k) defensive Programmierung, siehe Abschnitt S.1.11,
- l) Überwachungsmaßnahmen, siehe Abschnitt S.1.12.

5.2.3.4 Analytische Maßnahmen bei der Software-Entwicklung

- a) Inspektion, siehe Abschnitt S.2.1,
- b) Walk-through, siehe Abschnitt S.2.2,
- c) statische Analyse, siehe Abschnitt S.2.3,
- d) Korrektheitsbeweis (formale Verifikation), siehe Abschnitt S.2.4,
- e) symbolische Ausführung, siehe Abschnitt S.2.5,
- f) Simulation, siehe Abschnitt S.2.6,
- g) systematischer Test, siehe Abschnitt S.2.7,
 - Black-box-Test
 - White-box-Test
- h) statistischer Test, siehe Abschnitt S.2.8.

5.2.3.5 Funktionsprüfung des Vormusters bzw. Prototyps

- a) Prüfung der Funktion, siehe Abschnitt F.1,
- b) Prüfung der Störfestigkeit, siehe Abschnitt F.2,
- c) Prüfung der Zerstörfestigkeit, siehe Abschnitt F.3,
- d) Prüfung der Verträglichkeit gegenüber Umgebungsbedingungen, siehe Abschnitt F.4.

5.2.3.6 Organisatorische Maßnahmen

- a) Projektorganisation, siehe Abschnitt 0.1,
- b) Dokumentation, siehe Abschnitt 0.2.

5.3 Fehlervermeidung bei der Fertigungsvorbereitung

5.3.1 Einführung

In der Fertigungsvorbereitung werden mit Hilfe von Entwicklungsunterlagen, wie z.B. Konstruktionszeichnungen, Schaltpläne, Stücklisten, Aufbauanweisungen, technische Daten und Prüfvorschriften, Fertigungsabläufe geplant, Werkzeuge, Montagevorrichtungen und Prüfmittel entwickelt, bestellt, erstellt und für die Fertigung bereitgestellt. Dabei muß sichergestellt werden, daß die im Vormuster enthaltenen Eigenschaften auf das Zwischenmuster und die Fertigungsunterlagen übertragen werden.

5.3.2 Fehlerursachen

Fehlerursachen bei der Fertigungsvorbereitung können z.B. sein:

- Nichtbeachtung oder Fehlinterpretation der Entwicklungsunterlagen,
- Falsche Auswahl von Fertigungs-Hilfsmitteln,
- Mängel in der Fertigungsablaufplanung oder
- Auswahl eines ungeeigneten Fertigungsortes, bezogen auf Umgebungsbedingungen.

5.3.3 Maßnahmen

Folgende Einzelmaßnahmen sind zur Vermeidung von Fehlern bei der Fertigungsvorbereitung erforderlich:

5.3.3.1 Erstellen der Arbeits- und Prüfplanung

- a) Fertigungsplanung der Hardware, siehe Abschnitt A.8.1,
- b) Zwischenmuster der Hardware erstellen und prüfen, siehe Abschnitt A.8.2,
- c) Softwarebereitstellung, siehe Abschnitt A.8.3,
- d) Fertigungsplanung bei der Software, siehe Abschnitt A.8.4.

5.3.3.2 Systemtest, siehe Abschnitt A.8.5.

5.3.3.3 Fertigungs- und Anwenderdokumente erstellen, siehe Abschnitt A.8.6.

5.3.3.4 Fertigungsunterlagen, Fertigungsmittel und Softwareunterlagen übergeben, siehe Abschnitt A.8.7.

5.4 Fehlervermeidung bei der Fertigung

5.4.1 Einführung

In der Fertigung werden diejenigen Produkte produziert, die in der Entwicklung entwickelt und von der Fertigungsvorbereitung als Vormuster zur Verfügung gestellt wurden. Die Fertigung und die Arbeitsschritte in der Fertigung müssen den Vorgaben der Fertigungsvorbereitung entsprechen. Eigene Veränderungen der Fertigungsstätten am Produkt sind nicht zulässig, da sie die vorgeschalteten qualitätssichernden Schritte umgehen würden. Die Fertigung ist dafür verantwortlich, daß über einen längeren Zeitraum hinweg eine konstante Qualität erreicht wird.

Ab der Fertigung der Nullserie geht die Verantwortung für das Produkt von der Fertigungsvorbereitung an die Fertigung über.

5.4.2 Fehlerursachen

Die Fehlerursachen während der Fertigung beziehen sich im wesentlichen auf Nichtbeachtung oder falsche Interpretation der Unterlagen der Fertigungsvorbereitung und falsche Anwendung oder mangelhafte Wartung der Fertigungs- und Prüfmittel.

5.4.3 Maßnahmen

Die nachfolgend aufgeführten Einzelmaßnahmen zur Vermeidung von Fehlern werden im Abschnitt A.9 detailliert behandelt.

- a) Freigabe zur Nullserienfertigung, siehe Abschnitt A.9.1,
- b) Nullserienfertigung, siehe Abschnitt A.9.2,
- c) Freigabe zur Serienfertigung

Die Freigabe zur Serienfertigung wird aufgrund des Nullserien-Berichts erteilt.

- d) Serienfertigung, siehe Abschnitt A.9.3.

5.5 Fehlervermeidung bei der Inbetriebnahme

5.5.1 Bedeutung der Inbetriebnahme

Bei der Inbetriebnahme eines sicherheitsrelevanten Rechnersystems wird nicht nur die Integration des Rechnersystems als solches, sondern auch das Zusammenwirken des Rechnersystems mit der Anlage, deren Sicherheit es beeinflußt, erprobt.

Anmerkung: Im Ablaufplan nach Bild 5 wird vorausgesetzt, daß die Inbetriebnahme werkseitig erfolgt; anwendungsabhängig kann sie auch nach der Installation vor Ort durchgeführt werden.

Es gilt also, bei der Inbetriebnahme möglichst alle noch vorhandenen "Restfehler" aufzudecken, damit sie vor der endgültigen Freigabe des Systems behoben werden können.

Dabei muß natürlich vermieden werden, durch die Inbetriebsetzungsaktivitäten neue Fehler zu erzeugen. Demnach sind grundsätzlich zwei Arten von Maßnahmen zu unterscheiden: Maßnahmen, um Fehler während der Inbetriebnahme zu vermeiden, und Maßnahmen, um eventuell noch vorhandene "Restfehler" aufzudecken.

5.5.2 Fehler während der Inbetriebnahme

Es können bei der Inbetriebnahme eines sicherheitsrelevanten Rechnersystems selbst Fehler gemacht werden. Zum Beispiel könnten eine veraltete Version der Software benutzt werden, die Hardware-Konfiguration des Rechnersystems fehlerhaft sein, die Schnittstelle zwischen dem Rechnersystem und der zu sichernden Anlage falsch eingerichtet werden usw. Auch unwahrscheinlich erscheinende Fehler, wie etwa die Inbetriebnahme eines völlig falschen Rechnersystems, müssen vermieden werden.

5.5.3 Maßnahmen

Die nachfolgend aufgeführten Einzelmaßnahmen zur Vermeidung von Fehlern werden in **Anhang A** und **Anhang F** detailliert behandelt.

- a) Vorschriften für die Inbetriebnahme beachten, siehe Abschnitt A.10.1,
- b) einfache Funktionsprüfung, siehe Abschnitt F.5,
- c) erweiterte Funktionsprüfung, siehe Abschnitt F.6,
- d) Funktionsprüfung unter Grenzbedingungen, siehe Abschnitt F.7.

5.6 Fehlervermeidung bei der Installation

5.6.1 Einführung

Zur Installation gehören der Transport und die Aufstellung des Systems am vorgesehenen Ort.

5.6.2 Fehlerquellen

Fehlerquellen können liegen z.B. in

- falscher Planung oder Ausführung der Installation oder
- unzureichender Berücksichtigung der Umgebungsbedingungen (z.B. Einstrahlungen).

5.6.3 Maßnahmen

Die nachfolgend aufgeführten Einzelmaßnahmen zur Vermeidung von Fehlern werden im Abschnitt A.11 detailliert behandelt.

- a) Planung der Installation, siehe Abschnitt A.11.1,
- b) Einsatzbedingungen berücksichtigen, siehe Abschnitt A.11.2,
- c) Ausführung der Installation, siehe Abschnitt A.11.3.

5.7 Vermeidung von Fehlern beim Betrieb

5.7.1 Einführung und Abgrenzung

Unter dem Stichwort "Betrieb" sind die Phasen "Nutzung" und "Instandhaltung" zusammengefaßt. Der – sich an den Abschnitt "Installation" anschließende – Betrieb umfaßt somit alle mit der tatsächlichen Nutzung eines Systems zusammenhängenden Funktionen und Tätigkeiten **einschließlich** aller anfallenden Instandsetzungsarbeiten sowie eventuell durchzuführender vorbeugender Wartungen. Eine Ausnahme bilden nur die u.U. unvermeidlichen, im Laufe des Betriebes durchzuführenden **Änderungen** von Hard- oder Software, siehe Abschnitt 5.8.

5.7.2 Beschreibung von Einzelheiten, die zu berücksichtigen sind; Fehlerquellen

Es kann davon ausgegangen werden, daß das zu betrachtende System den sicherheitstechnischen Anforderungen entsprechend geplant, entwickelt, gefertigt, geprüft und installiert wurde, so daß während des Betriebes auch bei Auftreten von Ausfällen weder von der Hard- noch von der Software her eine Gefährdung zu erwarten ist.

Sicherungstechnische Systeme kommen jedoch im allgemeinen (noch) nicht ohne die Mithilfe des Menschen aus, der entweder bestimmte Betätigungen vorzunehmen hat oder aber im Rahmen von Instandhaltungsmaßnahmen tätig werden muß. Da es bekannt ist, daß von Zeit zu Zeit auch dem geschulten Bediener einer Anlage Fehlhandlungen unterlaufen, ergibt sich die Frage nach den sicherheitsrelevanten Folgen solcher Fehlhandlungen bzw. ihrer Vermeidung.

5.7.3 Maßnahmen, Methoden

Am zuverlässigsten sind gefährliche Folgen von Fehlhandlungen des Menschen zu vermeiden, wenn sie bereits vom Entwickler des Systems bedacht und eventuelle Fehlhandlungen durch geeignete Schutzmaßnahmen unwirksam gemacht wurden. Dies ist aber nur in begrenztem Maße möglich, sei es, daß es überhaupt keine hierfür geeigneten technischen Möglichkeiten gibt oder aber daß ihre Realisierung zu aufwendig würde, d.h. in keinem angemessenen Verhältnis zu der damit erzielten Schutzwirkung stünde. Dies gilt in besonderem Maße für alle mit der Wartung und Instandsetzung zusammenhängenden Aktivitäten.

Man ist hinsichtlich der Betriebsphase folglich darauf angewiesen, die sicherheitsrelevanten Einzelheiten der dem Menschen überlassenen Aktivitäten in entsprechenden **Regeln für Benutzung und Instandhaltung** festzuhalten. Diese Regeln sind Bestandteil der Betriebsunterlagen (siehe Bild 5, und Abschnitt 5.4). Ist diese Voraussetzung erfüllt, dann darf man davon ausgehen, daß entsprechend geschultes und verantwortungsbewußtes Personal diese Regeln auch befolgt. Dies kann zusätzlich und in gewissem Umfang durch geeignete Aufschreibungen und andere Kontrollen, die anwendungsspezifisch festzulegen sind, überwacht werden. Voraussetzung ist allerdings, daß die genannten Regeln ausführlich, vollständig und in für den Benutzer hinreichend verständlicher Form niedergelegt wurden.

Die nachfolgend aufgeführten Einzelmaßnahmen zur Vermeidung von Fehlern werden im Abschnitt A.12 detailliert behandelt.

- a) Benutzungsregeln beachten, siehe Abschnitt A.12.1,
- b) Instandhaltungsregeln beachten, siehe Abschnitt A.12.2.

5.8 Vermeidung von Fehlern bei Änderungen

5.8.1 Änderungen an sicherheitsbezogenen Rechnersystemen

Jede Änderung in der Hard- oder Software eines sicherheitsbezogenen Rechnersystems stellt einen Eingriff dar, der auch die Sicherheit beeinträchtigen kann. Änderungen können dennoch erforderlich werden, nämlich

- a) bis einschließlich der Inbetriebnahme. Beispielsweise wenn erkannt wird, daß eine vorgesehene Lösung undurchführbar ist, oder zur Anpassung an neue Vorschriften, oder um Schnittstellenprobleme zu vermeiden usw.
- b) nach der Inbetriebnahme, nicht nur zur Fehlerbeseitigung, auch zur Steigerung der Sicherheit des Gesamtsystems, Anpassung an neue Vorschriften usw.

Änderungen können betreffen:

- Hardware, z.B. die Benutzung farbiger statt monochromer Bildschirme,
- Software, z.B. die Berücksichtigung obiger Hardwareänderung oder Verbesserungen an experimentell ermittelten Daten, die als Parameter benutzt werden,
- Organisatorisches, z.B. Qualifikation des Betätigungspersonals.

Änderungen können notwendig sein:

- weil das Rechnersystem sich anders verhält als spezifiziert worden ist (Fehlerbeseitigung),

- weil das Verhalten des Rechnersystems in einer bestimmten Situation gar nicht spezifiziert worden ist (Mangelbeseitigung),
- weil andere Änderungen berücksichtigt werden müssen (Anpassung),
- weil neue Anforderungen gestellt werden (Ergänzung),
- weil das Rechnersystem dadurch wirtschaftlicher wird.

5.8.2 Fehlerquellen bei Änderungen

Bei allen Änderungen können Fehler auftreten:

- weil die Änderung fehlerhaft durchgeführt wurde (beispielsweise kann beim Anschluß eines neuen Gerätes eine Verpolung oder bei einer Parameteränderung ein Vorzeichenfehler vorkommen),
- weil die Auswirkungen einer an sich "richtigen" Änderung nicht ausreichend berücksichtigt wurden ("Folgefehler"; zum Beispiel kann eine Änderung der Versorgungsspannung für ein (neues) Gerät sich auf ein anderes (nicht geändertes) auswirken, oder es kann eine Änderung der Genauigkeit einer arithmetischen Funktion zu Schnittstellenproblemen in anderen Unterprogrammen führen) oder
- weil eine "falsche" Änderung durchgeführt wurde (dies kann besonders dann auftreten, wenn eine vorangegangene Änderung, die mit der neuen nicht verträglich ist, nicht in sämtlichen relevanten Dokumenten nachgetragen wurde, so daß sie unberücksichtigt blieb).

Gerade dieses letzte Beispiel macht deutlich, wie unerläßlich es ist, daß jede noch so "kleine" Änderung in der gesamten Dokumentation – von der "Verbindlichen Spezifikation" an – durchgängig berücksichtigt wird.

5.8.3 Formalisierung des Änderungsgeschehens

Siehe Abschnitt A.13.

6 Maßnahmen zur Beherrschung von Fehlern

Im folgenden werden die möglichen Maßnahmen zur Beherrschung von Fehlern im Rechnersystem behandelt, die sich erst im Betrieb bemerkbar machen.

Wenn ein Rechnersystem betrieben wird, muß man damit rechnen, daß Fehler im Rechnersystem enthalten sind, siehe Abschnitt 4.

Im Betrieb kann man nur mit Maßnahmen zur Fehlerbeherrschung erreichen, daß beim Auftreten dieser Fehler keine gefährliche Situation eintritt. Diese Maßnahmen müssen bei der Planung des Rechnersystems berücksichtigt und in das System eingefügt werden.

Voraussetzung für die Beherrschung von Fehlern ist deren Erkennung. Sie können im wesentlichen mit Hilfe von zwei Methoden erkannt werden. Die erste Methode benutzt den Vergleich von Ergebnissen für die Fehlererkennung und fordert daher einen gewissen Redundanzaufwand, siehe Abschnitt 6.1. Die zweite Methode besteht darin, ein oder mehrere Testverfahren gezielt zur Fehlererkennung in das Rechnersystem einzubinden, siehe Abschnitt 6.2.

Der Konstrukteur eines sicheren Rechnersystems muß nun diese Maßnahmen in einem Rechnersystem vereinen. Er muß deshalb in einem ersten Schritt eine passende Struktur finden, die im wesentlichen vom Verhalten des Prozesses im Fehlerfall abhängig ist (Prozeß mit sicherem oder nicht sicherem Zustand, zulässige Fehlertoleranzzeit, erforderlicher Fehleraufdeckungsgrad). In einem zweiten Schritt müssen zusätzliche Rechnertests eingebunden werden, wenn die gefährlichen Fehler nicht schon in ausreichendem Maße über die Struktur erkannt werden können. In einem dritten Schritt müssen organisatorische Anforderungen festgelegt werden, um die Fähigkeit, Fehler im Rechnersystem aufdecken zu können, auch über die Lebensdauer hin sicherzustellen.

Die folgende Auflistung von Maßnahmen ist deshalb nach drei Kriterien unterteilt worden. Unter dem ersten Kriterium (strukturelle Maßnahmen auf Systemebene) sind diejenigen Maßnahmen aufgelistet, deren Wirkung in der Struktur begründet sind. Unter dem zweiten Kriterium (Maßnahmen unterhalb der Systemebene) sind diejenigen Maßnahmen aufgelistet, die nur innerhalb einer Rechnerkomponente wirksam werden. Unter dem dritten Kriterium sind diejenigen Maßnahmen aufgelistet, die nichttechnischer oder organisatorischer Natur sind.

Die im folgenden aufgeführten Maßnahmen sind im **Anhang B** nach folgendem Schema beschrieben:

- | | | |
|--------------------------|---|-----------------------------|
| - Ziel der Maßnahme |) | |
| |) | |
| - zugrundegelegte Fehler |) | pro Rechnerkomponente sowie |
| |) | |
| - Bewertungsschema |) | |

Beschreibung der Einzelmaßnahmen und

Bewertung der Einzelmaßnahmen.

6.1 Strukturelle Maßnahmen zur Beherrschung von Fehlern auf Systemebene

Durch die Auswahl der Systemstruktur kann die gefährliche Auswirkung von Fehlern durch zwei Prinzipien vermieden werden. Im ersten Fall wird nach dem Auftreten und Erkennen des Fehlers der Prozeß in einen sicheren Zustand überführt. Dies kann bei der Mehrzahl der Prozesse durch Abschalten der Energiezufuhr bewirkt werden. Bei niedrigen Anforderungsklassen kann dies auch durch nichtredundante, d.h. einkanalige Strukturen erreicht werden. Im zweiten Fall wird nach dem Auftreten und Erkennen des Fehlers auf einen redundanten Kanal

umgeschaltet und durch diesen der Prozeß weitergeführt. Bei Prozessen, die beim Ausfall der Energiezufuhr oder der Informationseingabe nicht selbsttätig in einen sicheren Zustand übergehen, muß Redundanz, d.h. eine mehrkanalige Struktur, in jedem Fall vorgesehen werden, um eine gefährliche Situation zu vermeiden. Eine Überführung in den sicheren Zustand, kann bei niedrigen Anforderungsklassen auch durch nichtredundante, d.h. einkanalige Strukturen erreicht werden.

Fehlererkennung auf der Systemebene erfolgt über den Vergleich von Ausgangswerten redundanter Kanäle und geht von der Voraussetzung aus, daß ein Rechner korrekt arbeitet, wenn er richtige Ausgangswerte liefert. Dies ist allerdings nur dann gegeben, wenn der innere Rechnerzustand sich in den Ausgängen oder im Zeitverhalten der Ausgänge offenbart. Diese Methode kann deshalb nur dann sinnvoll eingesetzt werden, wenn Wechsel in Aus- oder Eingängen vorkommen. Ein Vorteil dieser Methode besteht darin, daß der Vergleich unmittelbar an den sicherheitsrelevanten Ausgängen des Rechnersystems stattfinden kann, was auch dazu benutzt werden kann, fehlerhafte Ausgangssignale zu unterdrücken. Weiterhin kann diese Methode sowohl auf Rechnerebene wie auch auf Bauteilebene durchgeführt werden. Der wesentliche Nachteil besteht in dem erhöhten Redundanzaufwand.

Die Maßnahmen auf Systemebene werden für die nachfolgend aufgeführten Strukturen und Systeme detailliert in **Abschnitt B.1** behandelt.

6.1.1 Einkanalige Strukturen, Maßnahmen siehe Abschnitt B.1.1.

- a) Einkanalige Strukturen mit Funktionsprüfung (Funktionstest), Maßnahmen siehe Abschnitt B.1.1.1,
- b) Einkanalige Strukturen mit Selbsttest, Maßnahmen siehe Abschnitt B.1.1.2,
- c) Einkanalige Strukturen mit Selbsttest und Überwachungseinrichtung, Maßnahmen siehe Abschnitt B.1.1.3,

6.1.2 Zweikanalige Strukturen, Maßnahmen siehe Abschnitt B.1.2.

- a) Zweikanalige sicherheitsgerichtete Strukturen mit Umschalter, Maßnahmen siehe Abschnitt B.1.2.1,
- b) Zweikanalige sicherheitsgerichtete Strukturen, Maßnahmen siehe Abschnitt B.1.2.2,
- c) Zweikanalige sicherheitsgerichtete Strukturen mit Vergleich, Maßnahmen siehe Abschnitt B.1.2.3.

6.1.3 Drei- und mehrkanalige Strukturen, Maßnahmen siehe Abschnitt B.1.3.

- a) Eins-von-drei-Struktur (1 von 3), Maßnahmen siehe Abschnitt B.1.3.1,

- b) Zwei-von-drei-Struktur mit Mehrheitsbewerter (2 von 3), Maßnahmen siehe Abschnitt B.1.3.2,
- c) Drei-von-drei-Struktur mit Vergleich (3 von 3), Maßnahmen siehe Abschnitt B.1.3.3,
- d) Weitere mehrkanalige Strukturen, Maßnahmen siehe Abschnitt B.1.3.4.

6.1.4 Diversitäre Hardware, Maßnahmen siehe Abschnitt B.1.4.

6.2 Maßnahmen zur Beherrschung von Fehlern unterhalb der Systemebene

Maßnahmen unterhalb der Systemebene sind spezielle Verfahren zur Fehlererkennung bzw. Beherrschung für einzelne Rechnerkomponenten und erfordern zusätzlichen Hardware- und Softwareaufwand. Software-gestützte Testverfahren haben den Nachteil, daß für die Testverfahren zusätzliche Rechenzeit zur Verfügung gestellt werden muß, die dann nicht mehr für die Prozeßbearbeitung zur Verfügung steht. Sie haben den Vorteil, daß das Rechnersystem für die Testverfahren meist nur gering abgeändert werden muß. Zusätzliche Hardware bedingt oft eine wesentliche Änderung des Rechnersystems, erfordert allerdings kaum zusätzliche Rechenzeit.

Die Maßnahmen unterhalb der Systemebene werden wie nachfolgend aufgeführt im Abschnitt B.2 detailliert behandelt.

6.2.1 Rechnerkomponenten (Betrachtungseinheit), Maßnahmen siehe Abschnitt B.2.1.

- a) Verarbeitungseinheiten (Hardware, z.B. CPU, Controller), Maßnahmen siehe Abschnitt B.2.1.1,
- b) Invariante Speicherbereiche, Maßnahmen siehe Abschnitt B.2.1.2,
- c) Variante Speicherbereiche, Maßnahmen siehe Abschnitt B.2.1.3,
- d) Ein- und Ausgabe-Einheiten, Schnittstellen, Maßnahmen siehe Abschnitt B.2.1.4,
- e) Datenwege (Bus), Maßnahmen siehe Abschnitt B.2.1.5,
- f) Versorgungseinheiten (z.B. Spannung, Takt, Lüfter), Maßnahmen siehe Abschnitt B.2.1.6,
- g) Programmablauf, Maßnahmen siehe Abschnitt B.2.1.7,
- h) Software, Maßnahmen siehe Abschnitt B.2.1.8.

6.2.2 Maßnahmen für sonstige Komponenten, Maßnahmen siehe Abschnitt B.2.2.

- a) Vergleicher und Mehrheitsentscheider, Zeitüberwachungseinrichtungen, Codegeneratoren und Prüfer, Maßnahmen siehe Abschnitt B.2.2.1,

b) Rechnerexterne Datenwege, Maßnahmen siehe Abschnitt B.2.2.2.

6.3 Nichttechnische Maßnahmen zur Beherrschung von Fehlern

Siehe Abschnitt B.3.

7 Zusammenhang zwischen Anforderungsklassen und Maßnahmen

7.1 Aufteilung der Maßnahmen in Gruppen zur Erleichterung der Kombination von Maßnahmen

Die in den Abschnitten 5 und 6 behandelten Maßnahmen zur Fehlervermeidung und Fehlerbeherrschung werden in **drei Gruppen** aufgeteilt, und zwar in Basismaßnahmen (Gruppe I), nichtaustauschbare, jedoch abstufbare Maßnahmen (Gruppe II) und austauschbare Maßnahmen (Gruppe III). Die Aufteilung sagt vom Grundsatz her nichts über die Wertigkeit und Wirksamkeit der Maßnahmen aus.

7.1.1 Gruppe I Basismaßnahmen

Hierunter fallen alle die Maßnahmen, die unabhängig von der Anforderungsklasse angewandt werden müssen und nicht abstufbar sind. Als Basismaßnahmen ist ihre Anwendung in allen Anforderungsklassen unverzichtbar.

Maßnahmen der Gruppe I sind

- a) Beachtung der zutreffenden technischen Regeln, siehe Abschnitt H.1.1,
- b) Verwendung einer Verbindlichen Spezifikation, siehe Abschnitt A.1,
- c) Freigabe zur Nullserienfertigung, siehe Abschnitt A.9.1,
- d) Nullserienfertigung, siehe Abschnitt A.9.2,
- e) Freigabe zur Serienfertigung (wird aufgrund des Nullserien-Berichts erteilt), siehe Abschnitt A.9.2,
- f) Serienfertigung, siehe Abschnitt A.9.3,
- g) Vorschriften für die Inbetriebnahme beachten, siehe Abschnitt A.10.1,
- h) einfache Funktionsprüfung, siehe Abschnitt F.5,
- i) Planung der Installation, siehe Abschnitt A.11.1,
- j) Einsatzbedingungen berücksichtigen, siehe Abschnitt A.11.2,
- k) Ausführung der Installation, siehe Abschnitt A.11.3,

- l) Benutzungsregeln beachten, siehe Abschnitt A.12.1,
- m) Instandhaltungsregeln beachten, siehe Abschnitt A.12.2,
- n) Formalisierung des Änderungsgeschehens, siehe Abschnitt A.13.

7.1.2 Gruppe II Nichtaustauschbare, jedoch abstufbare Maßnahmen

Hierunter fallen die Maßnahmen, die ebenfalls unabhängig von der Anforderungsklasse angewandt werden müssen, die aber abhängig von der Anforderungsklasse abgestuft angewandt werden können. Die Abstufbarkeit dieser Maßnahmen kann in bestimmten Fällen bis hinunter zu den Basismaßnahmen reichen. Der Aufwand zur Realisierung dieser Maßnahmen ist vom Grundsatz her so zu wählen, daß die in **Tabelle 1** (siehe Abschnitt 7.3) geforderte Wirksamkeit (einfach/mittel/hoch) erreicht wird.

Maßnahmen der Gruppe II sind

- a) strukturierter Entwurf (Hardware), siehe Abschnitt H.1.3,
- b) Spezifikation von hoher Qualität (Hardware), siehe Abschnitt H.1.4,
- c) Baueinheitenauswahl, siehe Abschnitt H.1.5,
- d) Einhaltung von Richtlinien zur Erstellung von Software, siehe Abschnitt S.1.1,
- e) Spezifikation von hoher Qualität (Software), siehe Abschnitt S.1.3,
- f) strukturierter Entwurf (Software), siehe Abschnitt S.1.4,
- g) Modularisierung, siehe Abschnitt S.1.5,
- h) strukturierte Programmierung, siehe Abschnitt S.1.7,
- i) defensive Programmierung, siehe Abschnitt S.1.11,
- j) Prüfung der Funktion, siehe Abschnitt F.1,
- k) Prüfung der Störfestigkeit, siehe Abschnitt F.2,
- l) Prüfung der Verträglichkeit gegenüber Umgebungsbedingungen, siehe Abschnitt F.4,
- m) Projektorganisation, siehe Abschnitt 0.1,
- n) Dokumentation, siehe Abschnitt 0.2.

7.1.3 Gruppe III Austauschbare Maßnahmen

Hierunter fallen Maßnahmen, die prinzipiell austauschbar sind, und zwar einzeln oder in Verbindung mit mehreren Maßnahmen. Im Regelfall sind diese Maßnahmen abgestuft realisierbar.

Alle Maßnahmen zur Fehlerbeherrschung nach Abschnitt 6 fallen in die Gruppe III der austauschbaren Maßnahmen. Weiterhin gehören dazu z.B.

- a) Verwendung rechnergestützter Konstruktions- und Entwicklungssysteme, siehe Abschnitt H.1.2,
- b) Entwurfsprüfung durch Inspektion, siehe Abschnitt H.2.1,
- c) Entwurfsprüfung durch Walk-through, siehe Abschnitt H.2.2,
- d) statische Analyse, siehe Abschnitt H.2.3,
- e) dynamische Analyse, siehe Abschnitt H.2.4,
- f) Funktionsanalyse, siehe Abschnitt H.2.5,
- g) Simulation, siehe Abschnitt H.2.6,
- h) Berechnung von Ausfallgrößen, siehe Abschnitt H.2.7,
- i) Ausfalleffektanalyse, siehe Abschnitt H.2.8,
- j) Funktionstest, siehe Abschnitt H.2.9,
- k) Verwendung rechnergestützter Entwurfs- und Software-Produktionssysteme, siehe Abschnitt S.1.2,
- l) Verwendung höherer Programmiersprachen, siehe Abschnitt S.1.6,
- m) Verwendung besonderer Systemsoftware, siehe Abschnitt S.1.8,
- n) Verzicht auf Optimierungen, siehe Abschnitt S.1.9,
- o) Minimierung von Realzeiteinflüssen, siehe Abschnitt S.1.10,
- p) Überwachungsmaßnahmen, siehe Abschnitt S.1.12,
- q) Inspektion, siehe Abschnitt S.2.1,
- r) Walk-through, siehe Abschnitt S.2.2,
- s) statische Analyse, siehe Abschnitt S.2.3,
- t) Korrektheitsbeweis (formale Verifikation), siehe Abschnitt S.2.4,
- u) symbolische Ausführung, siehe Abschnitt S.2.5,

- v) Simulation, siehe Abschnitt S.2.6,
- w) systematischer Test, siehe Abschnitt S.2.7,
- x) statistischer Test, siehe Abschnitt S.2.8,
- y) Prüfung der Zerstörfestigkeit, siehe Abschnitt F.3,
- z) erweiterte Funktionsprüfung, siehe Abschnitt F.6,
- za) Funktionsprüfung unter Grenzbedingungen, siehe Abschnitt F.7.

7.2 Merkmale für die Anforderungsklassen

Grundsätzliches Merkmal für die Maßnahmen, die in Abhängigkeit von den Anforderungsklassen anzuwenden sind, sind bestimmte, im jeweiligen Anwendungsfall anzunehmende Fehlerarten, gegen die Maßnahmen der Vermeidung oder Beherrschung erforderlich sind. Die Eintrittswahrscheinlichkeit der verschiedenen Fehlerarten (nach Abschnitt 4.3) ist unterschiedlich. Die Unterschiede ergeben sich aus den Fehlerarten selbst, aus dem Einsatzfall und seinem Umfeld, aus der Komplexität des Rechners, aus der benutzten Bauelemente-Technik, aus den Zuverlässigkeits- und Qualitätsmaßnahmen der Standardtechnik usw.

Eine quantitative Angabe von Eintrittswahrscheinlichkeiten ist allgemein nur schwer oder nicht im ausreichenden Maße möglich. Qualitativ sind jedoch relative Unterschiede zwischen den Fehlerarten abschätzbar (siehe Abschnitt 7.3).

Die Vollständigkeit, mit der alle die möglichen Fehlerarten durch geeignete Maßnahmen (Maßnahmenkombinationen) der Fehlervermeidung und Fehlerbeherrschung abgedeckt werden, ist ein Charakteristikum der Anforderungsklassen. In niedrigen Anforderungsklassen werden Maßnahmen im wesentlichen gegen Fehlerarten mit relativ hoher Eintrittswahrscheinlichkeit erforderlich sein. Mit zunehmenden Anforderungen (d.h. höheren Anforderungsklassen) werden auch Maßnahmen gegen Fehlerarten mit geringerer Eintrittswahrscheinlichkeit erforderlich.

Maßnahmen bzw. Maßnahmenkombinationen werden demnach nach zwei abgestuften Kriterien für den jeweiligen Anwendungsfall ausgewählt:

- a) Im Anwendungsfall anzunehmende Fehlerarten;
- b) Maßnahmen gegen die anzunehmenden Fehlerarten mit einfacher, mittlerer, hoher Wirksamkeit.

7.3 Anforderungen für die einzelnen Merkmale in Abhängigkeit von der Anforderungsklasse

In der **Tabelle 1** sind in der linken Spalte die wichtigsten Ausfallursachen aufgelistet. Maßnahmen, die entweder aus der Gruppe der Fehlervermeidung oder (und) aus der Gruppe der Fehlerbeherrschung

wählbar sind, werden, entsprechend ihrer Wirksamkeit (einfach, mittel, hoch) auf der rechten Seite der Tabelle den Anforderungsklassen als Mindestanforderung zugeordnet. Sie erleichtert es dem Anwender, entsprechend seiner Anforderungsklasse Maßnahmen gegen die aufgeführten wichtigsten Ausfälle zu wählen. Tabelle 1 macht eine Aussage, welche **Mindestwirksamkeit** die zu wählenden Maßnahmen im Regelfall haben sollten. Sie macht weiter darüber eine Aussage, ob fehlervermeidende oder (und) fehlerbeherrschende Maßnahmen zu treffen sind. Die Bewertung der Wirksamkeit bezieht sich auf Maßnahmenkombinationen, die eine Auswahl oder Zusammenstellung der Einzelmaßnahmen aus den Abschnitten 5 und 6 sein können.

Tabelle 1 ist sozusagen ein **statisches Modell** für die zu fordernde Wirksamkeit der **im Regelfall** zu ergreifenden Maßnahmen zur Fehlervermeidung und Fehlerbeherrschung. Die in Tabelle 1 geforderte Wirksamkeit (einfach/mittel/hoch) der zu wählenden Maßnahmen ist ebenfalls für den Regelfall festgelegt und darf – abhängig vom Einzelfall, begründet – in bestimmten Grenzen variiert werden. Nicht dargestellt ist die grundsätzliche Möglichkeit – in bestimmten Grenzen – Maßnahmen zur Fehlerbeherrschung durch Maßnahmen zur Fehlervermeidung zu ersetzen und umgekehrt. Voraussetzung hierbei ist, daß die Wirksamkeit aller Einzelmaßnahmen zu Maßnahmenkombinationen vergleichbar ist.

Eine ganze Reihe von Maßnahmen, die in den Abschnitten 5 und 6 sowie in den Anhängen aufgeführt sind, ist wirksam gegenüber Fehlern in unterschiedlichen Teilen des Rechners (z.B. Fehler in CPU, Speicher, Ein-/Ausgängen) oder wirkt gegenüber Fehlern verschiedener Herkunft (z.B. zufällige Hardwarefehler, Programmierfehler). Im Regelfall gibt es aber für jede Maßnahme eine bestimmte "Hauptwirkungsrichtung". Bei der Entscheidung über die Einordnung der Maßnahmen wurde diese "Hauptwirkungsrichtung" zugrunde gelegt. Bei der Beurteilung der Wirksamkeit von Maßnahmenkombinationen können selbstverständlich auch die positiven "Nebenwirkungen" der Einzelmaßnahmen berücksichtigt werden.

Tabelle 1. Wirksamkeit der zu treffenden Maßnahmen gegenüber Fehlern in Abhängigkeit von den Anforderungsklassen

Versagen bedingt durch		Besondere sicherheitstechnische Maßnahmen entsprechend den Anforderungsklassen								
		1	2	3	4	5	6	7	8	
Zufallsfehler in der Hardware	Einfachfehler	Maßnahmen der Fehlerbeherrschung								
		einfach		mittel		hoch				
	Mehrfachfehler durch Fehlerhäufung (statistisch)					Maßnahmen der Fehlerbeherrschung				
				einfach		mittel		hoch		
Systematische Fehler einsch. Common-Mode-Fehler	in der Hardware	Basismaßnahmen			Maßnahmen der Fehlervermeidung:					
					einfach		mittel		hoch	
					Maßnahmen der Fehlerbeherrschung					
					mittel		hoch			
	in der Software	Basismaßnahmen			Maßnahmen der Fehlervermeidung:					
					einfach		mittel		hoch	
			Maßnahmen der Fehlerbeherrschung							
				einfach		mittel		hoch		
Handhabungsfehler, Bedienfehler, Manipulation		Basismaßn.	Maßnahmen der Fehlervermeidung:							
			einfach		mittel		hoch			
			Maßnahmen der Fehlerbeherrschung							
		einfach		mittel		hoch				
Fehler durch Betriebs- und Umgebungseinflüsse		Basismaßn.	Maßnahmen der Fehlervermeidung:							
			einfach		mittel		hoch			
			Maßnahmen der Fehlerbeherrschung							
				einfach		mittel		hoch		

Legende : einfach / mittel / hoch = Bezeichnung zur Wirksamkeit der Maßnahmen.

7.4 Erfüllung der Anforderungen durch Maßnahmenkombinationen

7.4.1 Zusammenstellung von geeigneten Maßnahmenkombinationen

Die Zusammenstellung der Kombinationen erfolgt derart, daß – ausgehend von dem einer Anforderungsklasse zugeordneten Rechner-Teilrisiko – Einzelmaßnahmen zur Fehlervermeidung und Fehlerbeherrschung so kombiniert werden, daß sie in der **Summe ihrer Wirksamkeit** geeignet sind, die Wahrscheinlichkeit einer gefährlichen Funktionsstörung hinreichend klein zu machen. In **Anhang X** sind nach dem Stand der Technik beispielhaft solche Kombinationen zusammengestellt.

Man darf auch andere Maßnahmen zu neuen, nicht in Anhang X genannten Kombinationen zusammenstellen oder dort aufgeführte Einzelmaßnahmen ersetzen. Dabei sollte sorgfältig geprüft werden, ob eine neu entstandene Kombination in ihrer Gesamtwirkung mit den bereits beispielhaft genannten Kombinationen vergleichbar ist.

7.4.2 Einfluß von Prozeßvorgaben auf die Auswahl von Maßnahmen

Maßnahmenkombinationen innerhalb einer Anforderungsklasse sind einander gleichwertig. Es wird eine Reihe von Maßnahmenkombinationen vorgegeben, weil mit einer Maßnahmenkombination allein nicht sämtliche Randbedingungen, die durch Prozeß, Herstellung und Prüfung bestimmt werden, berücksichtigt werden können. Solche Randbedingungen sind z.B.:

- a) Die Eigenschaft eines Prozesses, einen sicheren Zustand einnehmen zu können

Die Mehrzahl aller technischen Prozesse hat die Eigenschaft, bei Abschaltung der Energiezufuhr oder Stillstand der Informationsverarbeitung selbsttätig in einen sicheren Zustand überzugehen. Diese Eigenschaft kann man mit sogenannten "sicherheitsgerichteten Schaltungen" ausnutzen, um beim Auftreten eines Fehlers den Prozeß in einen sicheren Zustand übergehen zu lassen. Prozesse ohne einen derartigen sicheren Zustand erfordern zusätzliche Redundanzen; im allgemeinen einen zusätzlichen Hardwarekanal.

- b) Die Eigenschaft eines Prozesses, ein falsches Signal tolerieren zu können

Die Mehrzahl aller technischen Prozesse hat die Eigenschaft, kurzfristig ein fehlerhaftes Eingangssignal tolerieren zu können. Da einkanäle Strukturen zwar Fehler durch Selbsttests erkennen können, dies aber erst nach Auftreten des Fehlers geschehen kann, sind einkanäle Strukturen nur für Prozesse zulässig, die kurzfristig ein fehlerhaftes Eingangssignal tolerieren.

- c) Die Fehlertoleranzzeit des Prozesses

Maßnahmen zur Fehlererkennung benötigen einen gewissen Zeitaufwand, der bei Maßnahmen mit Selbsttests wesentlich größer ist als

bei Maßnahmen durch Redundanz und Vergleich. Der Prozeß setzt diesem Zeitaufwand Grenzen, denn der Zeitaufwand, die Steuer- und Regelfunktion auszuführen (Nutzzeit), Fehler aufzudecken (Fehleroffenbarungszeit) und auf den Fehler zu reagieren (Fehlerreaktionszeit) muß kleiner sein als die Zeitspanne, nach der der Prozeß in einen unsicheren Zustand übergeht (Fehlertoleranzzeit). Aus diesem Grunde ist der Entwickler manchmal gezwungen, auf redundante Strukturen mit Vergleich oder schnellere Selbsttestverfahren (meistens mit Hardware ausgeführt) auszuweichen.

d) Die Dynamik des Prozesses

Fehler in Rechnersystemen können durch Selbsttests, ergänzende Tests oder bei redundanten Strukturen über einen Vergleich von Ergebnissen oder Zwischenergebnissen festgestellt werden. Selbsttests erfassen Fehler zwar datenflußunabhängig, aber dafür meist unspezifisch mit einem Aufdeckungsgrad, der z.Z. noch nicht beliebig steigerbar ist. Fehlererkennung über Vergleich kann zwar direkt an den sicherheitsrelevanten Ausgängen oder Zwischenergebnissen stattfinden, jedoch erfolgt die Fehlererkennung datenflußabhängig. Ist durch die Prozeßdynamik keine automatische Fehlererkennung über einen Vergleich gegeben, sind ergänzende Tests in Form von programmgesteuerten Dynamisierungen vorzusehen.

e) Die Nachweisbarkeit bzw. Nachweiskosten einer Maßnahme

Die Anzahl sowie die Prüfbarkeit von Maßnahmen beeinflussen auch den Aufwand für den Nachweis der Wirksamkeit. Es kann im Einzelfall sogar von Vorteil sein, mehr Aufwand zu treiben als durch die Anforderungsklasse gefordert wird, um den Nachweisaufwand zu senken.

Anhang A Maßnahmen zur Vermeidung von Fehlern (Technische Konkretisierung von Abschnitt 5)

A.1 Verwenden einer Verbindlichen Systemspezifikation (Einzelmaßnahme zu Abschnitt 5.1.3)

A.1.1 Beschreibung der Maßnahme

Die Verbindliche Systemspezifikation beginnt mit ihrem Erstellen, indem die Funktionen und Eigenschaften des geplanten Systems in implementierungsunabhängiger Weise dargestellt werden. Die Verbindliche Systemspezifikation muß den Anforderungen nach Abschnitt 5.1.2.1 genügen. Sie kann sowohl mit formalen wie rechnergestützten Hilfsmitteln erstellt werden.

Für diese Verbindliche Systemspezifikation wird das Gliederungsschema nach Tabelle A.1 vorgeschlagen.

Tabelle A.1. Gliederungsschema einer verbindlichen Systemspezifikation

- | | |
|-----|---|
| 1 | Funktionale Spezifikation |
| 1.1 | Externe Schnittstellen |
| 1.2 | Mensch-Maschine-Schnittstellen |
| 1.3 | Betriebsmodi |
| 1.4 | Systemfunktionen |
| 2 | Sicherheitstechnische Anforderungen |
| 2.1 | Sicherheitsrichtlinien und Regeln für den Sicherheitsnachweis |
| 2.2 | Zu betrachtende Fehler und Ausfälle |
| 2.3 | Verhalten bei Fehlern und Ausfällen (einschließlich Zeitverhalten) |
| 2.4 | Wiederanlaufverfahren |
| 2.5 | Grenzwerte für Sicherheit und Zuverlässigkeit |
| 2.6 | Besondere Maßnahmen zur Sicherstellung der geforderten Fehlertoleranz |
| 2.7 | Organisatorische Maßnahmen zum Schutz vor äußeren Einwirkungen |
| 3 | Zu berücksichtigende Umgebungsbedingungen |
| 3.1 | Art der Umgebungsbedingungen |
| 3.2 | Zulässige Grenzwerte |
| 3.3 | Verhalten des Systems bei bestimmten Umgebungsbedingungen |
| 4 | Gestaltungsvorgaben |
| 4.1 | Besondere Vorgaben für Entwurf und Implementierung |
| 4.2 | Verfügbare Komponenten |
| 4.3 | Verfügbares Personal |
| 4.4 | Verfügbare Betriebsmittel, Ver- und Entsorgung |
| 4.5 | Verfügbare Kommunikationsmittel |
| 5 | Rahmenbedingungen für Betrieb und Instandhaltung |
| 5.1 | Notwendige Einrichtungen und Schnittstellen für die Prüfung und Instandhaltung |
| 5.2 | Technische Rahmenbedingungen für die Installation |
| 5.3 | Organisatorische Rahmenbedingungen für Betrieb und Instandhaltung |
| 5.4 | Abnahmevorschriften. |

Wird das System niedrigen Anforderungsklassen nach DIN V 19250 zugeordnet, so kann ein abgekürztes Gliederungsschema benutzt werden, in dem jedoch Angaben zu den Hauptpunkten

- funktionale Spezifikation,
- sicherheitstechnische Anforderungen und
- zu berücksichtigende Umgebungsbedingungen

vorzusehen sind.

A.1.2 Leistungsmerkmale der Maßnahme

Die Verbindliche Systemspezifikation schafft die notwendigen Grundlagen für den weiteren Systementwurf und ist eine Voraussetzung für alle weiteren Maßnahmen zur Fehlervermeidung in der Konzept- und Entwicklungsphase.

A.1.3 Besonderheiten der Maßnahme

Auf die Verbindliche Systemspezifikation kann verzichtet werden. Der erforderliche Umfang hängt jedoch von der Anforderungsklasse ab.

A.1.4 Voraussetzungen für die Maßnahme

Keine.

A.1.5 Nachweismöglichkeit für die Maßnahme

Vorhandensein der Verbindlichen Systemspezifikation.

A.2 Verwendung von rechnergestützten Entwicklungswerkzeugen (Einzelmaßnahme zu Abschnitt 5.1.3)

A.2.1 Beschreibung der Maßnahme

Die Maßnahme besteht in der Verwendung eines im Regelfall umfangreichen Programmsystems zum Erstellen der Spezifikation. Alle Angaben zur Spezifikation werden in einer Datenbank gesammelt und können nach Kriterien, wie Widerspruchsfreiheit und (zumindest teilweise) Vollständigkeit automatisch überprüft werden. Das Entwicklungswerkzeug kann dem Anwender das spezifizierte System unter verschiedenen Aspekten darstellen. Im Regelfall werden nicht nur das Erstellen der Spezifikation, sondern auch der Entwurf und teilweise weitere Phasen des Projektablaufes unterstützt. Die verschiedenen Entwurfswerkzeuge kann man nach den Abschnitten A.2.1.1 bis A.2.1.4 klassifizieren.

A.2.1.1 Werkzeuge ohne Bevorzugung einer bestimmten Entwurfsmethode

Das Entwicklungswerkzeug nimmt dem Anwender bestimmte Routinearbeiten ab und unterstützt das Projektmanagement; es erzwingt jedoch kein bestimmtes Vorgehen bei der Spezifikation <LAU 85>. Die

relative Ungebundenheit an eine Methode läßt dem Anwender große Freiheit, bringt jedoch auch wenig Unterstützung, wie beim Erstellen der Spezifikation vorzugehen ist. Dies erschwert die Einarbeitung.

A.2.1.2 Modellorientierte Verfahren mit hierarchischer Zerlegung

Diese geben auf verschiedenen Abstraktionsebenen (Feinheitsgrade) ein funktionelles Abbild des gewünschten Systems wieder (Structured Analysis <ROSS77>). Die Zerlegung auf den verschiedenen Ebenen betreffen sowohl die Aktionen als auch die Daten. Überprüfungen auf Widerspruchsfreiheit und Vollständigkeit sind sowohl beim Übergang von einer Hierarchiestufe zur anderen als auch zwischen zwei Funktionseinheiten (Modulen) derselben Ebene möglich.

A.2.1.3 Verfahren zur Beschreibung von Objekten und deren Beziehungen

Das gewünschte System wird als eine Menge von Objekten und deren Beziehungen beschrieben. Durch das verwendete Werkzeug wird festgelegt, welche Beziehungen vom System interpretierbar sind. Im allgemeinen erlauben diese eine Beschreibung der hierarchischen Struktur der Objekte, des Datenflusses, Beziehungen zwischen den Daten und welche Daten bestimmten Verarbeitungsprozessen unterliegen. Das klassische Verfahren <TeHe77, TeHe78> hat eine Reihe von Erweiterungen speziell für Prozeßrechner-Anwendungen erfahren <LUDE81>. Überprüfungsmöglichkeiten und Unterstützung des Anwenders hängen von der Art der darstellbaren Beziehungen ab. Andererseits macht eine große Vielfalt von Darstellungsmöglichkeiten die Anwendung unübersichtlich.

A.2.1.4 Verfahren zur Beschreibung nach dem Schema "Anreiz und Antwort"

Die Beziehungen zwischen den Objekten eines Systems werden unter dem Ordnungsschema "Anreiz und Antwort" dargestellt <ALF77>. Zur Darstellung wird eine einfache, leicht erweiterbare Sprache verwendet, die Sprachelemente zur Darstellung von Objekten, Beziehungen, Eigenschaften und Strukturen enthält. Eine Erweiterung wurde bereits in einem sicherheitsbezogenen Projekt angewandt <DAH83>.

A.2.2 Leistungsmerkmale der Maßnahme

Rechnergestützte Spezifikationswerkzeuge zwingen zur Verwendung formaler Darstellungsmittel und erlauben die automatische Überprüfung der Widerspruchsfreiheit und in gewissem Umfang die Überprüfung der Vollständigkeit. Sie sind hilfreich beim Vermeiden von Trivialfehlern und zwingen durch die formale Darstellung zu einer präzisen, eindeutigen Spezifikation. Durch die Erledigung von Routinearbeiten kann sich der Ersteller einer Spezifikation auf die essentiellen Teile konzentrieren. Die Leistungsmerkmale hängen im einzelnen vom verwendeten Werkzeug ab.

Rechnergestützte Werkzeuge können nur eine relative Vollständigkeit der verbindlichen Spezifikation prüfen, d.h. Funktionen oder Fallunterscheidungen, auf die an keiner Stelle Bezug genommen wird,