

Inhalt

	Seite
Vorwort.....	2
Einleitung.....	12
1 Anwendungsbereich	14
2 Normative Verweisungen.....	16
3 Begriffe und Abkürzungen	16
Anhang A (informativ) Überblick über Verfahren und Maßnahmen für sicherheitsbezogene E/E/PE- Systeme: Beherrschung von zufälligen Hardwareausfällen (siehe IEC 61508-2).....	17
A.1 Elektrik	17
A.1.1 Erkennung von Ausfällen durch Überwachung während des Betriebs	17
A.1.2 Überwachung von Relaiskontakten	17
A.1.3 Vergleicher.....	17
A.1.4 Mehrheitsentscheider.....	18
A.1.5 Ruhestromprinzip.....	18
A.2 Elektronik	18
A.2.1 Tests durch redundante Hardware	18
A.2.2 Dynamische Prinzipien	18
A.2.3 Standardtestschnittstelle (Access Port) und Boundary-Scan-Architektur	19
A.2.4 (Nicht verwendet).....	19
A.2.5 Überwachte Redundanz	19
A.2.6 Hardware mit automatischen Tests	19
A.2.7 Analogsignal-Überwachung.....	20
A.2.8 Unterlastung.....	20
A.3 Verarbeitungseinheiten (CPUs).....	20
A.3.1 Selbsttest durch Software: begrenzte Anzahl von Mustern (ein Kanal)	20
A.3.2 Selbsttest durch Software: Walking Bit (ein Kanal)	20
A.3.3 Selbsttest unterstützt durch Hardware (ein Kanal)	21
A.3.4 Codierte Verarbeitung (ein Kanal)	21
A.3.5 Gegenseitiger Vergleich durch Software	21
A.4 Unveränderliche Speicherbereiche.....	21
A.4.1 Wortsicherungsverfahren mit Mehr-Bit-Redundanz (zum Beispiel ROM-Überwachung mit einem modifizierten Hammingcode)	21
A.4.2 Modifizierte Prüfsumme	22
A.4.3 Signatur mit einfacher Wortbreite (8 Bit).....	22
A.4.4 Signatur mit doppelter Wortbreite (16 Bit)	22
A.4.5 Blockwiederholung (zum Beispiel doppeltes ROM mit Hardware- oder Softwarevergleich).....	23
A.5 Veränderliche Speicherbereiche.....	23
A.5.1 RAM-Test „Checkerboard“ oder „March“	24

	Seite
A.5.2 RAM-Test „Walkpath“	24
A.5.3 RAM-Test „Galpat“ oder „transparenter Galpat“	24
A.5.4 RAM-Test „Abraham“	25
A.5.5 Ein-Bit-Redundanz (zum Beispiel RAM-Überwachung mit einem Parity-Bit)	25
A.5.6 RAM-Überwachung mit einem modifizierten Hammingcode oder Erkennung von Datenfehlern mit fehlererkennenden und -korrigierenden Codes (en: error-detection-correction codes, EDC)	25
A.5.7 Doppeltes RAM mit Hardware- oder Softwarevergleich und Schreib-/Lesetest	26
A.6 E/A-Einheiten und Schnittstellen (externe Kommunikation)	26
A.6.1 Testmuster	26
A.6.2 Codesicherung	26
A.6.3 Mehrkanalige parallele Ausgabe	27
A.6.4 Überwachte Ausgaben	27
A.6.5 Eingabevergleich/-entscheidung	27
A.7 Datenwege (interne Kommunikation)	27
A.7.1 Ein-Bit-Hardware-Redundanz	27
A.7.2 Mehr-Bit-Hardware-Redundanz	27
A.7.3 Vollständige Hardware-Redundanz	28
A.7.4 Inspektion durch Verwendung von Testmustern	28
A.7.5 Übertragungs-Redundanz	28
A.7.6 Informations-Redundanz	28
A.8 Spannungsversorgung	28
A.8.1 Überspannungsschutz mit Sicherheitsabschaltung	28
A.8.2 Spannungsüberwachung (sekundärseitig)	29
A.8.3 Energieabschaltung mit Sicherheitsabschaltung	29
A.9 Zeitliche und logische Programmablaufüberwachung	29
A.9.1 Watchdog mit separater Zeitbasis ohne Zeitfenster	29
A.9.2 Watchdog mit separater Zeitbasis und Zeitfenster	29
A.9.3 Logische Überwachung des Programmablaufs	30
A.9.4 Kombination von zeitlicher und logischer Überwachung des Programmablaufs	30
A.9.5 Zeitliche Überwachung mit Test während des Betriebs	30
A.10 Lüftung und Beheizung	30
A.10.1 Temperatursensor	30
A.10.2 Lüfterkontrolle	30
A.10.3 Auslösung der Sicherheitsabschaltung über eine thermische Sicherung	31
A.10.4 Gestaffelte Meldung von Thermosensoren und bedingter Alarm	31
A.10.5 Zuschaltung von Umluftkühlung und Statusanzeige	31
A.11 Kommunikation und Massenspeicher	31
A.11.1 Trennung elektrischer Energieleitungen von Informationsleitungen	31
A.11.2 Räumliche Trennung mehrfacher Leitungen	31

	Seite
A.11.3 Erhöhung der Störfestigkeit	31
A.11.4 Antivalente Signalübertragung.....	32
A.12 Sensoren.....	32
A.12.1 Referenzsensor.....	32
A.12.2 Zwangsöffnender Schalter.....	33
A.13 Stellglieder (Aktoren)	33
A.13.1 Überwachung.....	33
A.13.2 Kreuzweise Überwachung mehrfacher Aktoren	33
A.14 Maßnahmen gegen die Einwirkung der physikalischen Umgebung.....	33
Anhang B (informativ) Überblick über Verfahren und Maßnahmen für sicherheitsbezogene E/E/PE-Systeme: Vermeidung von systematischen Ausfällen (siehe IEC 61508-2 und IEC 61508-3)	35
B.1 Allgemeine Maßnahmen und Verfahren.....	35
B.1.1 Projektmanagement.....	35
B.1.2 Dokumentation.....	36
B.1.3 Trennung der Sicherheitsfunktionen des E/E/PE-Systems von Nichtsicherheitsfunktionen	37
B.1.4 Diversitäre Hardware	37
B.2 Spezifikation der Anforderungen an den Entwurf des E/E/PE-Systems	37
B.2.1 Strukturierte Spezifikation.....	37
B.2.2 Formale Methoden.....	38
B.2.3 Semi-formale Methoden	39
B.2.4 Rechnerunterstützte Spezifikationswerkzeuge.....	41
B.2.5 Checklisten	42
B.2.6 Inspektion der Spezifikation.....	43
B.3 Entwurf und Entwicklung des E/E/PE-Systems.....	43
B.3.1 Beachtung von Richtlinien und Normen	44
B.3.2 Strukturierter Entwurf.....	44
B.3.3 Verwendung von bewährten Bauteilen.....	44
B.3.4 Modularisierung	45
B.3.5 Rechnerunterstützte Entwurfswerkzeuge.....	45
B.3.6 Simulation	46
B.3.7 Inspektion (Überprüfungen und Analysen)	46
B.3.8 Walkthrough.....	47
B.4 Betriebs- und Instandhaltungsverfahren für das E/E/PE-System.....	47
B.4.1 Betriebs- und Instandhaltungsanweisungen.....	47
B.4.2 Benutzerfreundlichkeit	47
B.4.3 Instandhaltungsfreundlichkeit	48
B.4.4 Eingeschränkte Betriebsmöglichkeiten.....	48
B.4.5 Betrieb nur durch erfahrene Bediener	48
B.4.6 Schutz gegen Irrtümer des Bediener	49

	Seite
B.4.7 (Nicht verwendet)	49
B.4.8 Schutz vor Modifikation	49
B.4.9 Eingabebestätigung	49
B.5 Integration des E/E/PE-Systems	50
B.5.1 Funktionstest	50
B.5.2 Black-Box-Test	50
B.5.3 Statistisches Testen	51
B.5.4 Felderfahrung	51
B.6 Validierung der Sicherheit des E/E/PE-Systems	52
B.6.1 Funktionstest unter Umgebungsbedingungen	52
B.6.2 Test der Störfestigkeit gegen Stoßspannungen	53
B.6.3 (Nicht verwendet)	53
B.6.4 Statische Analyse	53
B.6.5 Dynamische Analyse und Test	54
B.6.6 Ausfallanalyse	54
B.6.7 Worst-Case-Analyse	60
B.6.8 Erweiterte Funktionstests	61
B.6.9 Test unter Grenzbedingungen	61
B.6.10 Test durch Fehlereinbau	61
Anhang C (informativ) Überblick über Verfahren und Maßnahmen zum Erreichen der Sicherheitsintegrität der Software (siehe IEC 61508-3)	63
C.1 Allgemeines	63
C.2 Anforderungen und detaillierter Entwurf	63
C.2.1 Strukturierte schematische Methoden	63
C.2.2 Datenflussdiagramme	65
C.2.3 Strukturdiagramme	66
C.2.4 Formale Methoden	67
C.2.5 Defensives Programmieren	71
C.2.6 Entwurfs- und Programmierrichtlinien	72
C.2.7 Strukturiertes Programmieren	77
C.2.8 Geheimnisprinzip/Kapselung	77
C.2.9 Modularer Ansatz	78
C.2.10 Verwendung bewährter/verifizierter Softwareelemente	79
C.2.11 Nachvollziehbarkeit	81
C.2.12 Zustandsloser Softwareentwurf (oder Entwurf eingeschränkter Zustände)	82
C.2.13 Numerische Offline-Analyse	83
C.2.14 Nachrichtenverlaufstabellen	83
C.3 Entwurf der Architektur	83
C.3.1 Fehlererkennung und Diagnose	83

	Seite
C.3.2 Fehlererkennende und korrigierende Codes	84
C.3.3 Assertion-Programmierung (en: failure assertion programming)	85
C.3.4 Diversitäre Überwachungseinrichtung	85
C.3.5 Diversitäre Programmierung	86
C.3.6 Rückwärtsregeneration	86
C.3.7 Regeneration durch Wiederholung	87
C.3.8 Abgestufte Funktionseinschränkungen	87
C.3.9 Künstliche Intelligenz – Fehlerkorrektur	88
C.3.10 Dynamische Rekonfiguration	88
C.3.11 Sicherheit und Leistungsfähigkeit in Echtzeit: zeitgesteuerte Architektur	89
C.3.12 UML	90
C.4 Entwicklungswerkzeuge und Programmiersprachen	91
C.4.1 Streng typisierte Programmiersprache	91
C.4.2 Sprachenteilmenge	92
C.4.3 Zertifizierte Werkzeuge und Compiler	92
C.4.4 Betriebsbewährte Werkzeuge und Compiler	92
C.4.5 Geeignete Programmiersprache	93
C.4.7 Testmanagement und Automatisierungswerkzeuge	97
C.5 Verifikation und Modifikation	97
C.5.1 Statistisches Testen	97
C.5.2 Datenaufzeichnung und Analyse	98
C.5.3 Schnittstellenprüfung	98
C.5.4 Durchführung von Testfällen nach einer Grenzwertanalyse	99
C.5.5 Durchführung von Testfällen aus der Fehlererwartung („Fehler erraten“)	99
C.5.6 Durchführung von Testfällen nach Fehlereinpflanzung	100
C.5.7 Äquivalenzklassen und Test von Partitionen des Eingangsbereichs	100
C.5.8 Strukturabhängige Tests	101
C.5.9 Kontrollflussanalyse	102
C.5.10 Datenflussanalyse	102
C.5.11 Symbolische Ausführung	103
C.5.12 Formaler Beweis (Verifikation)	103
C.5.13 Softwarekomplexitätskontrolle	105
C.5.14 Formale Inspektion	105
C.5.15 Walk-through (Software)	106
C.5.16 Entwurfsüberprüfung	106
C.5.17 Prototypenerstellung/Animation	107
C.5.18 Simulation des Prozesses	107
C.5.19 Anforderungen an die Leistungsfähigkeit	108
C.5.20 Modellierung der Leistungsfähigkeit	108

	Seite
C.5.21 Belastungstest (en: avalanche/stress testing).....	109
C.5.22 Reaktionszeiten und Speicherbeschränkungen.....	110
C.5.23 Einflussanalyse.....	110
C.5.24 Software-Konfigurationsmanagement.....	110
C.5.25 Validierung durch Regressionstest.....	111
C.5.26 Animation der Spezifikation und des Entwurfs.....	111
C.5.27 Modellbasiertes Testen (Testfallgenerierung).....	112
C.6 Beurteilung der funktionalen Sicherheit.....	113
C.6.1 Entscheidungs-/Wahrheitstabellen.....	114
C.6.2 Software-Gefährdungs- und Betriebbarkeitsuntersuchung (en: software hazard and operability study, CHAZOP, FMEA).....	114
C.6.3 Analyse von Ausfällen infolge gemeinsamer Ursache.....	114
C.6.4 Zuverlässigkeitsblockdiagramm.....	115
Anhang D (informativ) Ein probabilistischer Ansatz zur Bestimmung der Sicherheitsintegrität von vorentwickelter Software.....	117
D.1 Allgemeines.....	117
D.2 Gleichungen für statistische Tests und Beispiele für ihre Anwendung.....	118
D.2.1 Einfacher statistischer Test für die Betriebsart mit niedriger Anforderungsrate.....	118
D.2.2 Test des Wertebereichs der Eingänge für die Betriebsart mit niedriger Anforderungsrate.....	118
D.2.3 Einfacher statistischer Test für die Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung.....	119
D.2.4 Vollständiger Test.....	120
D.3 Literaturhinweise.....	121
Anhang E (informativ) Überblick über Verfahren und Maßnahmen für den Entwurf von ASICs.....	122
E.1 Entwurfsbeschreibung in (V)HDL.....	122
E.2 Schaltplaneingabe.....	122
E.3 Strukturierte Beschreibungsmethodik.....	122
E.4 Betriebsbewährte Werkzeuge.....	123
E.5 (V)HDL-Simulation.....	123
E.6 Funktionstest auf Modulebene.....	123
E.7 Funktionstest auf oberster Ebene.....	124
E.8 Funktionstest eingebettet in Systemumgebung.....	124
E.9 Eingeschränkte Verwendung asynchroner Konstrukte.....	124
E.10 Synchronisation von primären Eingängen und Kontrolle von Metastabilitäten.....	124
E.11 Entwurf für Testbarkeit.....	125
E.12 Modularisierung.....	125
E.13 Testabdeckung der Verifikationsszenarien (Testbenches).....	125
E.14 Beachtung von Programmierrichtlinien.....	126
E.15 Verwendung eines Codecheckers.....	127
E.16 Defensive Programmierung.....	127

	Seite
E.17	Dokumentation von Simulationsergebnissen 127
E.18	Codeinspektion 128
E.19	Walkthrough..... 128
E.20	Anwendung validierter Soft-Cores 128
E.21	Validierung von Soft-Cores 129
E.22	Simulation der Gatter-Netzliste zur Überprüfung der Zeitvorgaben 129
E.23	Statische Laufzeitanalyse (STA)..... 129
E.24	Vergleich der Gatter-Netzliste gegen ein Referenzmodell durch Simulation 130
E.25	Vergleich der Gatter-Netzliste mit dem Referenzmodell (formale Äquivalenzprüfung)..... 130
E.26	Überprüfung der Anforderungen und Beschränkungen des Herstellers 130
E.27	Dokumentation der Synthesevorgaben, Ergebnisse und Werkzeuge 131
E.28	Verwendung von betriebsbewährten Synthesewerkzeugen 131
E.29	Verwendung von betriebsbewährten Zielbibliotheken 131
E.30	Skriptbasierende Verfahren 131
E.31	Implementierung von Teststrukturen 132
E.32	Abschätzung der Testabdeckung durch Simulation 132
E.33	Abschätzung der Testabdeckung durch Anwendung eines ATPG-Werkzeugs 133
E.34	Nachweis der Betriebsbewährung für verwendete Hard-Cores 133
E.35	Verwendung validierter Hard-Cores 133
E.36	Online-Tests der Hard-Cores..... 133
E.37	Design-Rule-Check (DRC) 133
E.38	Überprüfung des Layouts Versus Schematic (LVS) 134
E.39	Zusätzliche Reserven (> 20 %) für Prozesstechnologien mit weniger als 3 Jahren Anwendung 134
E.40	Burn-In-Test 134
E.41	Verwendung von betriebsbewährten Schaltkreisfamilien 134
E.42	Betriebsbewährter Fertigungsprozess 134
E.43	Qualitätskontrolle des Fertigungsprozesses..... 135
E.44	Produktions-Qualitätsprüfung des Schaltkreises 135
E.45	Funktionale Qualitätsprüfung des Schaltkreises 135
E.46	Qualitätsnormen..... 135
Anhang F (informativ) Definitionen der Eigenschaften der Software-Lebenszyklusphasen..... 136	
Anhang G (informativ) Anleitung zur Entwicklung von sicherheitsbezogener objektorientierter Software..... 143	
Stichwortverzeichnis 145	
Literaturhinweise 153	
Anhang ZA (normativ) Normative Verweisungen auf internationale Publikationen mit ihren entsprechenden europäischen Publikationen..... 156	
Bilder	
Bild 1 – Gesamtrahmen der IEC 61508 15	

	Seite
Tabellen	
Tabelle C.1 – Empfehlungen für bestimmte Programmiersprachen	95
Tabelle D.1 – Notwendige Vorgeschichte zur Zuordnung von Sicherheits-Integritätsleveln bei gegebenem Vertrauensniveau	117
Tabelle D.2 – Wahrscheinlichkeiten eines Versagens für die Betriebsart mit niedriger Anforderungsrate	118
Tabelle D.3 – Mittlerer Abstand von zwei Testpunkten	119
Tabelle D.4 – Wahrscheinlichkeit eines Versagens für die Betriebsart mit hoher Anforderungsrate oder kontinuierlicher Anforderung	120
Tabelle D.5 – Wahrscheinlichkeit des Tests aller Programmeigenschaften	121
Tabelle F.1 – Spezifikation der Anforderungen an die Sicherheit der Software.....	136
Tabelle F.2 – Softwareentwurf und Softwareentwicklung: Entwurf der Softwarearchitektur.....	137
Tabelle F.3 – Softwareentwurf und Softwareentwicklung: unterstützende Werkzeuge und Programmiersprache	138
Tabelle F.4 – Softwareentwurf und Softwareentwicklung: detaillierter Entwurf.....	138
Tabelle F.5 – Softwareentwurf und Softwareentwicklung: Test der Softwaremodule und Integration.....	139
Tabelle F.6 – Integration der programmierbaren Elektronik (Hardware und Software)	139
Tabelle F.7 – Softwareaspekte bezüglich der Validierung der Systemsicherheit.....	140
Tabelle F.8 – Softwaremodifikation	140
Tabelle F.9 – Softwareverifikation	141
Tabelle F.10 – Beurteilung der funktionalen Sicherheit.....	141
Tabelle G.1 – Objektorientierte Softwarearchitektur.....	143
Tabelle G.2 – Detaillierter objektorientierter Entwurf.....	144
Tabelle G.3 – Einige detaillierte objektorientierte Begriffe.....	144