

Jörg Schäufler
Thomas Zurawka

Automotive Software Engineering

Aus dem Programm

Kraftfahrzeugtechnik

Handbuch Verbrennungsmotor

herausgegeben von R. van Basshuysen und F. Schäfer

Lexikon Motorentechnik

herausgegeben von R. van Basshuysen und F. Schäfer

Vieweg Handbuch Kraftfahrzeugtechnik

herausgegeben von H.-H. Braess und U. Seiffert

Bremsenhandbuch

herausgegeben von B. Breuer und K. H. Bill

Verbrennungsmotoren

von E. Köhler

Passive Sicherheit von Kraftfahrzeugen

von F. Kramer

Motorradtechnik

von J. Stoffregen

Omnibustechnik

von O.-P. A. Bühler, herausgegeben vom VDA

Kurbeltriebe

von S. Zima

Motorkolben

von S. Zima

Die BOSCH-Fachbuchreihe

- **Ottomotor-Management**
- **Dieselmotor-Management**
- **Autoelektrik/Autoelektronik**
- **Sicherheits- und Komfortsysteme**
- **Fachwörterbuch Kraftfahrzeugtechnik**
- **Kraftfahrtechnisches Taschenbuch**

herausgegeben von ROBERT BOSCH GmbH

vieweg

Jörg Schäuffele
Thomas Zurawka

Automotive Software Engineering

**Grundlagen, Prozesse,
Methoden und Werkzeuge
effizient einsetzen**

3., verbesserte und erweiterte Auflage

Mit 277 Abbildungen

ATZ/MTZ-Fachbuch



Bibliografische Information Der Deutschen Bibliothek
Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie;
detaillierte bibliografische Daten sind im Internet über <http://dnb.ddb.de> abrufbar.

1. Auflage Juli 2003
- 2., verbesserte Auflage Oktober 2004
- 3., verbesserte und erweiterte Auflage März 2006

Alle Rechte vorbehalten

© Friedr. Vieweg & Sohn Verlag | GWV Fachverlage GmbH, Wiesbaden 2006

Lektorat: Thomas Zipsner

Der Vieweg Verlag ist ein Unternehmen von Springer Science+Business Media.
www.vieweg.de



Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Verlags unzulässig und strafbar. Das gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Konzeption und Layout des Umschlags: Ulrike Weigel, www.CorporateDesignGroup.de
Druck- und buchbinderische Verarbeitung: Wilhelm & Adam, Heusenstamm
Gedruckt auf säurefreiem und chlorfrei gebleichtem Papier.
Printed in Germany

ISBN 3-8348-0051-1

Zur Bedeutung von Software im Automobil

Neue Entwicklungsmethoden zur Beherrschung der Komplexität erforderlich

Auch in Zukunft ist Software die Technik, um komplexe Algorithmen im Kraftfahrzeug umzusetzen. Der nahezu exponentiell wachsende Software-Umfang wird getrieben durch die Funktionszunahme in den Feldern sicher, sauber, sparsam und seit einiger Zeit auch im Bereich der Fahrerassistenz. Die Beherrschung der daraus resultierenden Komplexität ist für Fahrzeughersteller wie Zulieferer eine Herausforderung, die nur mit leistungsfähigen Methoden und Werkzeugen möglich ist. Mit ihrer Hilfe gilt es, die sichere Funktion der Software und der Systeme zu gewährleisten. Das vorliegende Buch liefert dazu vielfältige Anregungen zur Gestaltung von Entwicklungsprozessen und zum Einsatz von Methoden und Werkzeugen.

Dr. Siegfried Dais, Stellvertretender Vorsitzender der Geschäftsführung der Robert Bosch GmbH, Stuttgart

Software im Fahrzeug als strategisches Produkt

Software im Fahrzeug wird mit weiter steigendem Anteil zunehmend mehr zum strategischen Produkt eines Automobilherstellers. Der Elektronik- und Software-Anteil in Fahrzeugen ist ein wesentlicher Innovationsmotor – etwa 90 % der Innovationen im Fahrzeug sind von der Elektronik geprägt. Einerseits werden immer mehr klassische Funktionen des Fahrzeugs durch Software implementiert, andererseits entstehen durch die Vernetzung bisher getrennter Funktionen völlig neue Möglichkeiten. Eine konsequente Anwendung von Systems Engineering ist ein wesentlicher Erfolgsfaktor, um das Gesamtsystem „Fahrzeug“ zu beherrschen. Das vorliegende Buch adressiert nun schon in der 3. Auflage diesen Themenkomplex und stellt die Fahrzeugsubsysteme Antrieb, Fahrwerk und Karosserie in den Vordergrund.

Hans-Georg Frischkorn, München

Vom Kostentreiber zum Wettbewerbsvorteil

Technischer Vorsprung in der Automobilindustrie kann nur durch eine Vorreiterrolle in der Software-Technik erreicht werden. Die erfolgreiche Zusammenarbeit von Ingenieuren unterschiedlicher Fachrichtungen in der Systementwicklung erfordert jedoch ein einheitliches Hintergrundwissen, eine gemeinsame Begriffswelt und ein geeignetes Vorgehensmodell. Die in diesem Buch dargestellten Grundlagen und Methoden demonstrieren dies anhand von Beispielen aus der Praxis sehr eindrucksvoll.

Dr.-Ing. Wolfgang Runge, Geschäftsführer, ZF Lenksysteme GmbH, Schwäbisch Gmünd

Ausbildung als Chance und Herausforderung

Gerade für den Raum Stuttgart spielt der Fahrzeugbau eine herausragende Rolle. Die Entwicklungszentren bedeutender Fahrzeughersteller und Zulieferer bieten hier viele Arbeitsplätze. Die Ausbildung in der Software-Technik ist ein fester Bestandteil eines Ingenieurstudienganges an der Universität Stuttgart. Dieses Buch bietet die Chance, praktische Erfahrungen in der Automobilindustrie bereits in der Ausbildung zu berücksichtigen. Viele der vorgestellten Vorgehensweisen können sogar Vorbildcharakter für andere Branchen haben.

Prof. Dr.-Ing. Dr. h.c. Peter Göhner, Institut für Automatisierungs- und Software-Technik, Universität Stuttgart

Vorwort

Auch heute, nach seiner über 100-jährigen Geschichte, ist das Kraftfahrzeug durch eine rasante Weiterentwicklung gekennzeichnet. Seit Anfang der 1970er Jahre ist die Entwicklung geprägt von einem – bis heute anhaltenden – stetigen Anstieg des Einsatzes von elektronischen Systemen und von Software im Fahrzeug. Dies führt zu gravierenden Veränderungen in der Entwicklung, in der Produktion und im Service von Fahrzeugen. So ermöglicht die zunehmende Realisierung von Fahrzeugfunktionen durch Software neue Freiheitsgrade beim Entwurf und die Auflösung bestehender Zielkonflikte.

Zur Beherrschung der dadurch entstehenden Komplexität sind Prozesse, Methoden und Werkzeuge notwendig, die die fahrzeugspezifischen Anforderungen berücksichtigen. Grundlegende charakteristische Anforderungen an Software im Automobil sind der Entwurf für eingebettete und verteilte Echtzeitsysteme bei hohen Sicherheits- und Verfügbarkeitsanforderungen sowie großem Kostendruck. Diese Anforderungen stehen bei diesem Buch im Vordergrund. Wir freuen uns sehr, dass es wegen hoher Nachfrage bereits in der 3. Auflage erscheinen kann.

Zur Entwicklung von Software für elektronische Systeme von Fahrzeugen wurden in den letzten Jahren eine Reihe von Vorgehensweisen und Standards entwickelt, die wohl am besten unter dem Begriff „Automotive Software Engineering“ zusammengefasst werden können.

Damit ist eine komplexe Begriffswelt entstanden, mit der wir ständig konfrontiert werden. Es wird immer schwieriger, genau zu verstehen, was sich hinter den Vokabeln verbirgt. Erschwerend kommt hinzu, dass manche Begriffe mehrfach in unterschiedlichem Zusammenhang verwendet werden. So etwa der Begriff Prozess, der im Zusammenhang mit der Regelungstechnik, aber auch mit Echtzeitsystemen oder generell mit Vorgehensweisen in der Entwicklung benutzt wird. Nach einem Überblick zu Beginn werden deshalb in diesem Buch die wichtigsten Begriffe definiert und durchgängig so verwendet.

In den folgenden Kapiteln stehen Prozesse, Methoden und Werkzeuge für die Entwicklung von Software für die elektronischen Systeme des Fahrzeugs im Mittelpunkt. Eine entscheidende Rolle spielen dabei die Wechselwirkungen zwischen der Software-Entwicklung als Fachdisziplin und der übergreifenden Systementwicklung, die alle Fahrzeugkomponenten berücksichtigen muss. Die dargestellten Prozesse haben Modellcharakter. Das bedeutet, die Prozesse sind ein abstraktes, idealisiertes Bild der täglichen Praxis. Sie können für verschiedene Entwicklungsprojekte als Orientierung dienen, müssen aber vor der Einführung in einem konkreten Projekt bewertet und eventuell angepasst werden. Auf eine klare und verständliche Darstellung haben wir deshalb großen Wert gelegt.

Wegen der Breite des Aufgabengebietes können nicht alle Themen in der Tiefe behandelt werden. Wir beschränken uns aus diesem Grund auf Gebiete mit automobilspezifischem Charakter.

Wir erheben an dieser Stelle nicht den Anspruch, die einzig richtige oder eine vollständige Vorgehensweise zu beschreiben. Als Mitarbeiter der ETAS GmbH und der SYSTECS GmbH sind wir natürlich der Überzeugung, dass die Werkzeuge und Software-Komponenten der ETAS zur Unterstützung der in diesem Buch vorgestellten Vorgehensweisen bestens geeignet sind.

Beispiele aus der Praxis

Ein Prozess ist nur dann ein erfolgreiches Instrument für ein Entwicklungsteam, wenn die Vorteile der nachvollziehbaren Bearbeitung umfangreicher Aufgabenstellungen in der Praxis anerkannt werden. Dieses Buch soll daher kein theoretisches Lehrbuch sein, das sich jenseits der Praxis bewegt. Alle Anregungen basieren auf praktischen Anwendungsfällen, die wir anhand von Beispielen anschaulich darstellen. Die Vielzahl von Erfahrungen, die wir in enger Zusammenarbeit mit Fahrzeugherstellern und Zulieferern in den letzten Jahren sammeln konnten, wurde dabei berücksichtigt. Dazu gehören Serienentwicklungen mit den dazugehörigen Produktions- und Serviceaspekten genauso wie Forschungs- und Vorentwicklungsprojekte.

Leserkreis

Wir wollen sprechen mit dem vorliegenden Buch alle Mitarbeiter bei Fahrzeugherstellern und Zulieferern ansprechen, die in der Entwicklung, in der Produktion und im Service mit Software im Fahrzeug konfrontiert sind. Wir hoffen nützliche Anregungen weitergeben zu können.

Dieses Buch soll andererseits auch eine Basis zur Ausbildung von Studierenden und zur Einarbeitung von neuen Mitarbeitern zur Verfügung stellen. Grundkenntnisse in der Steuerungs- und Regelungstechnik, in der Systemtheorie und in der Software-Technik sind von Vorteil, aber nicht Voraussetzung.

Sicherlich wird an einigen Stellen der Wunsch nach einer detaillierteren Darstellung aufkommen. Wir freuen uns deshalb zu allen behandelten Themen über Hinweise und Verbesserungsvorschläge. Zahlreiche Rückmeldungen der Leser zur 1. und 2. Auflage wurden bei der 3. Auflage des Buches berücksichtigt.

Danksagungen

An dieser Stelle möchten wir uns besonders bei allen unseren Kunden für die jahrelange, vertrauensvolle Zusammenarbeit bedanken. Ohne diesen Erfahrungsaustausch wäre dieses Buch nicht möglich gewesen.

Des Weiteren bedanken wir uns bei der BMW Group für die freundliche Zustimmung, dass wir in diesem Buch Erfahrungen darstellen dürfen, die wir in BMW Projekten – im Falle des erstgenannten Autors auch als BMW Mitarbeiter – gesammelt haben. Dies schließt auch Empfehlungen für Serienprojekte bei BMW ein. Wir bedanken uns besonders bei Herrn Heinz Merkle, Herrn Dr. Helmut Hochschwarzer, Herrn Dr. Maximilian Fuchs, Herrn Prof. Dr. Dieter Nazareth, sowie allen ihren Mitarbeitern.

Viele Vorgehensweisen und Methoden entstanden während der jahrelangen, engen und vertrauensvollen Zusammenarbeit mit der Robert Bosch GmbH. Die dabei entwickelten und inzwischen breit eingesetzten Vorgehensweisen finden sich in diesem Buch an vielen Stellen wieder. Wir sagen dafür den Mitarbeiterinnen und Mitarbeitern der Bereiche Chassis Systems, Diesel Systems, Gasoline Systems, sowie der Forschung und Voraentwicklung der Robert Bosch GmbH vielen Dank.

Wir bedanken uns außerdem herzlich bei Herrn Dr. Siegfried Dais, Herrn Hans-Georg Frischkorn, Herrn Dr. Wolfgang Runge und Herrn Prof. Dr. Peter Göhner für ihre einleitenden Geleitworte.

Abschließend bedanken wir uns bei allen unseren Kolleginnen und Kollegen, die in den letzten Jahren auf vielfältige Weise zu diesem Buch beigetragen haben. Für die sorgfältige und kritische Durchsicht des Manuskripts bedanken wir uns insbesondere bei Roland Jeutter,

Dr. Michael Nicolaou, Dr. Oliver Schlüter, Dr. Kai Werther und Hans-Jörg Wolff. Unser Dank gilt nicht zuletzt dem Vieweg-Verlag und Herrn Thomas Zipsner für die sehr gute Zusammenarbeit.

Stuttgart, Januar 2006

Jörg Schäuffele, Thomas Zurawka

Inhaltsverzeichnis

1 Einführung und Überblick	1
1.1 Das System Fahrer-Fahrzeug-Umwelt	2
1.1.1 Aufbau und Wirkungsweise elektronischer Systeme	2
1.1.2 Elektronische Systeme des Fahrzeugs und der Umwelt.....	5
1.2 Überblick über die elektronischen Systeme des Fahrzeugs.....	6
1.2.1 Elektronische Systeme des Antriebsstrangs	8
1.2.2 Elektronische Systeme des Fahrwerks	9
1.2.3 Elektronische Systeme der Karosserie	11
1.2.4 Multi-Media-Systeme.....	13
1.2.5 Verteilte und vernetzte elektronische Systeme.....	14
1.2.6 Zusammenfassung und Ausblick	15
1.3 Überblick über die logische Systemarchitektur.....	16
1.3.1 Funktions- und Steuergerätenetzwerk des Fahrzeugs	16
1.3.2 Logische Systemarchitektur für Steuerungs-/Regelungs- und Überwachungssysteme	17
1.4 Prozesse in der Fahrzeugentwicklung	18
1.4.1 Überblick über die Fahrzeugentwicklung	18
1.4.2 Überblick über die Entwicklung von elektronischen Systemen.....	19
1.4.3 Kernprozess zur Entwicklung von elektronischen Systemen und Software	22
1.4.4 Unterstützungsprozesse zur Entwicklung von elektronischen Systemen und Software	24
1.4.5 Produktion und Service von elektronischen Systemen und Software	27
1.5 Methoden und Werkzeuge für die Entwicklung von Software für elektronische Systeme	27
1.5.1 Modellbasierte Entwicklung	28
1.5.2 Integrierte Qualitätssicherung	28
1.5.3 Reduzierung des Entwicklungsrisikos	31
1.5.4 Standardisierung und Automatisierung.....	32
1.5.5 Entwicklungsschritte im Fahrzeug	34
2 Grundlagen	37
2.1 Steuerungs- und regelungstechnische Systeme	37
2.1.1 Modellbildung	37
2.1.2 Blockschaltbilder.....	38
2.2 Diskrete Systeme	42
2.2.1 Zeitdiskrete Systeme und Signale	43
2.2.2 Wertdiskrete Systeme und Signale.....	44
2.2.3 Zeit- und wertdiskrete Systeme und Signale.....	45
2.2.4 Zustandsautomaten.....	45
2.3 Eingebettete Systeme.....	47
2.3.1 Aufbau von Mikrocontrollern	48

2.3.2	Speichertechnologien	50
2.3.3	Programmierung von Mikrocontrollern	53
2.4	Echtzeitsysteme	60
2.4.1	Festlegung von Tasks	60
2.4.2	Festlegung von Echtzeitanforderungen	62
2.4.3	Zustände von Tasks	64
2.4.4	Strategien für die Zuteilung des Prozessors	66
2.4.5	Aufbau von Echtzeitbetriebssystemen	71
2.4.6	Interaktion zwischen Tasks	72
2.5	Verteilte und vernetzte Systeme	78
2.5.1	Logische und technische Systemarchitektur.....	81
2.5.2	Festlegung der logischen Kommunikationsbeziehungen	82
2.5.3	Festlegung der technischen Netzwerktopologie	84
2.5.4	Festlegung von Nachrichten	85
2.5.5	Aufbau der Kommunikation und des Netzwerkmanagements	86
2.5.6	Strategien für die Zuteilung des Busses	90
2.6	Zuverlässigkeit, Sicherheit, Überwachung und Diagnose von Systemen.....	92
2.6.1	Grundbegriffe	93
2.6.2	Zuverlässigkeit und Verfügbarkeit von Systemen	94
2.6.3	Sicherheit von Systemen	98
2.6.4	Überwachung und Diagnose von Systemen	101
2.6.5	Aufbau des Überwachungssystems elektronischer Steuergeräte.....	105
2.6.6	Aufbau des Diagnosesystems elektronischer Steuergeräte	108
2.7	Zusammenfassung	113
3	Unterstützungsprozesse zur Entwicklung von elektronischen Systemen und Software	117
3.1	Grundbegriffe der Systemtheorie	117
3.2	Vorgehensmodelle und Standards	120
3.3	Konfigurationsmanagement.....	122
3.3.1	Produkt und Lebenszyklus	122
3.3.2	Varianten und Skalierbarkeit.....	123
3.3.3	Versionen und Konfigurationen	124
3.4	Projektmanagement	127
3.4.1	Projektplanung	127
3.4.2	Projektverfolgung und Risikomanagement	132
3.5	Lieferantenmanagement	133
3.5.1	System- und Komponentenverantwortung.....	133
3.5.2	Schnittstellen für die Spezifikation und Integration.....	134
3.5.3	Festlegung des firmenübergreifenden Entwicklungsprozesses	134
3.6	Anforderungsmanagement.....	136
3.6.1	Erfassen der Benutzeranforderungen	136
3.6.2	Verfolgen von Anforderungen	140
3.7	Qualitätssicherung	141
3.7.1	Integrations- und Testschritte	141
3.7.2	Maßnahmen zur Qualitätssicherung von Software.....	142

4 Kernprozess zur Entwicklung von elektronischen Systemen und Software	145
4.1 Anforderungen und Randbedingungen	146
4.1.1 System- und Komponentenverantwortung	146
4.1.2 Abstimmung zwischen System- und Software-Entwicklung	147
4.1.3 Modellbasierte Software-Entwicklung.....	149
4.2 Grundbegriffe	149
4.2.1 Prozesse.....	149
4.2.2 Methoden und Werkzeuge	150
4.3 Analyse der Benutzeranforderungen und Spezifikation der logischen Systemarchitektur	151
4.4 Analyse der logischen Systemarchitektur und Spezifikation der technischen Systemarchitektur	154
4.4.1 Analyse und Spezifikation steuerungs- und regelungstechnischer Systeme	158
4.4.2 Analyse und Spezifikation von Echtzeitsystemen.....	159
4.4.3 Analyse und Spezifikation verteilter und vernetzter Systeme.....	160
4.4.4 Analyse und Spezifikation zuverlässiger und sicherer Systeme	161
4.5 Analyse der Software-Anforderungen und Spezifikation der Software-Architektur.....	162
4.5.1 Spezifikation der Software-Komponenten und ihrer Schnittstellen	162
4.5.2 Spezifikation der Software-Schichten	165
4.5.3 Spezifikation der Betriebszustände	166
4.6 Spezifikation der Software-Komponenten	167
4.6.1 Spezifikation des Datenmodells	168
4.6.2 Spezifikation des Verhaltensmodells	169
4.6.3 Spezifikation des Echtzeitmodells.....	171
4.7 Design und Implementierung der Software-Komponenten	173
4.7.1 Berücksichtigung der geforderten nichtfunktionalen Produkteigenschaften	174
4.7.2 Design und Implementierung des Datenmodells.....	176
4.7.3 Design und Implementierung des Verhaltensmodells.....	177
4.7.4 Design und Implementierung des Echtzeitmodells	178
4.8 Test der Software-Komponenten.....	178
4.9 Integration der Software-Komponenten	179
4.9.1 Erzeugung des Programm- und Datenstands.....	180
4.9.2 Erzeugung der Beschreibungsdateien	181
4.9.3 Erzeugung der Dokumentation.....	182
4.10 Integrationstest der Software	183
4.11 Integration der Systemkomponenten	184
4.11.1 Integration von Software und Hardware	184
4.11.2 Integration von Steuergeräten, Sollwertgebern, Sensoren und Aktuatoren.....	185
4.12 Integrationstest des Systems	187
4.13 Kalibrierung	190
4.14 System- und Akzeptanztest	190

5 Methoden und Werkzeuge in der Entwicklung	193
5.1 Offboard-Schnittstelle zwischen Steuergerät und Werkzeug	194
5.2 Analyse der logischen Systemarchitektur und Spezifikation der technischen Systemarchitektur	196
5.2.1 Analyse und Spezifikation steuerungs- und regelungstechnischer Systeme	196
5.2.2 Analyse und Spezifikation von Echtzeitsystemen	200
5.2.3 Analyse und Spezifikation verteilter und vernetzter Systeme	206
5.2.4 Analyse und Spezifikation zuverlässiger und sicherer Systeme	211
5.3 Spezifikation von Software-Funktionen und Validation der Spezifikation	218
5.3.1 Spezifikation der Software-Architektur und der Software-Komponenten	220
5.3.2 Spezifikation des Datenmodells	224
5.3.3 Spezifikation des Verhaltensmodells mit Blockdiagrammen	224
5.3.4 Spezifikation des Verhaltensmodells mit Entscheidungstabellen	227
5.3.5 Spezifikation des Verhaltensmodells mit Zustandsautomaten	230
5.3.6 Spezifikation des Verhaltensmodells mit Programmiersprachen	235
5.3.7 Spezifikation des Echtzeitmodells	235
5.3.8 Validation der Spezifikation durch Simulation und Rapid-Prototyping	235
5.4 Design und Implementierung von Software-Funktionen	247
5.4.1 Berücksichtigung der geforderten nichtfunktionalen Produkteigenschaften	247
5.4.2 Design und Implementierung von Algorithmen in Festpunkt- und Gleitpunktarithmetik	255
5.4.3 Design und Implementierung der Software-Architektur	270
5.4.4 Design und Implementierung des Datenmodells	274
5.4.5 Design und Implementierung des Verhaltensmodells	277
5.5 Integration und Test von Software-Funktionen	280
5.5.1 Software-in-the-Loop-Simulationen	281
5.5.2 Laborfahrzeuge und Prüfstände	283
5.5.3 Experimental-, Prototypen- und Serienfahrzeuge	289
5.5.4 Design und Automatisierung von Experimenten	290
5.6 Kalibrierung von Software-Funktionen	291
5.6.1 Arbeitsweisen bei der Offline- und Online-Kalibrierung	292
5.6.2 Software-Update durch Flash-Programmierung	294
5.6.3 Synchrones Messen von Signalen des Mikrocontrollers und der Instrumentierung	295
5.6.4 Auslesen und Auswerten von Onboard-Diagnosedaten	295
5.6.5 Offline-Verstellen von Parametern	296
5.6.6 Online-Verstellen von Parametern	297
5.6.7 Klassifizierung der Offboard-Schnittstellen für das Online-Verstellen	298
5.6.8 Management des CAL-RAM	303
5.6.9 Management der Parameter und Datenstände	306
5.6.10 Design und Automatisierung von Experimenten	307

6 Methoden und Werkzeuge in Produktion und Service	309
6.1 Offboard-Diagnose	310
6.2 Parametrierung von Software-Funktionen.....	311
6.3 Software-Update durch Flash-Programmierung.....	312
6.3.1 Löschen und Programmieren von Flash-Speichern.....	313
6.3.2 Flash-Programmierung über die Offboard-Diagnoseschnittstelle.....	313
6.3.3 Sicherheitsanforderungen.....	314
6.3.4 Verfügbarkeitsanforderungen	316
6.3.5 Auslagerung und Flash-Programmierung des Boot-Blocks	317
6.4 Inbetriebnahme und Prüfung elektronischer Systeme	319
7 Zusammenfassung und Ausblick	321
Literaturverzeichnis	323
Abkürzungsverzeichnis	329
Sachwortverzeichnis	331