Automotive Software Engineering
Principles, Processes, Methods, and Tools
SECOND EDITION

About the Authors
Jörg Schäuffele, Dipl.-Ing., studied Mechanical Engineering at the University of Stuttgart with a focus on control engineering. His professional career began with the firm of IKG GmbH, a spinoff of the Institute of Control Engineering at the University of Stuttgart. He later joined ETAS GmbH and supported the introduction of ETAS tools in numerous engineering and consulting projects with various customers in the automotive industry worldwide. He worked in the Systems Engineering Methods, Software Process, Tools department of the Electronics Development section at BMW Headquarters in Munich and he coordinated the various product divisions at ETAS. Since 2007 he works as Product Manager at Vector Informatik GmbH.

Thomas Zurawka, Dipl.-Ing., studied electrical engineering at the University of Stuttgart and wrote a detailed doctorate thesis on digital signal processing. He started his professional career with ZF Friedrichshafen AG before joining Advanced Development at Robert Bosch GmbH where he worked on real-time operating systems. Dr. Zurawka initially developed code generators at ETAS and subsequently headed the development of the ASCET engineering tool for software development. He has served as CEO of ETAS and has been responsible for overall development. He worked as Executive Vice President for Robert Bosch GmbH. Since 2006 he is shareholder and managing director at Systecs Informationssysteme GmbH.

About the Book
Since the early seventies, the development of the automobile has been characterized by a steady increase in the deployment of onboard electronics systems and software. This trend continues unabated and is driven by rising end-user demands and increasingly stringent environmental requirements. Today, almost every function onboard the modern vehicle is electronically controlled or monitored.

The software-based implementation of vehicle functions provides for unparalleled freedoms of concept and design. However, automobile development calls for the accommodation of contrasting prerequisites — such as higher demands on safety and reliability vs. lower cost ceilings, longer product life cycles vs. shorter development times — along with growing proliferation of model variants. Automotive Software Engineering has established its position at the center of these seemingly conflicting opposites. This book provides background basics as well as numerous suggestions, rare insights, and cases in point concerning those processes, methods, and tools that contribute to the surefooted mastery of the use of electronic systems and software in the contemporary automobile.
## Contents

*Foreword: The Role of Software in the Automobile* .............. xi

*Preface to the Second English Edition* ........................... xiii

*Acknowledgments* .......................................................... xv

**Chapter 1: Introduction and Overview** .............................. 1

1.1 The Driver-Vehicle-Environment System ......................... 2
  1.1.1 Design and Method of Operation of Vehicle Electronic  
  Systems .................................................. 3
  1.1.2 Electronic Systems of the Vehicle and the Environment .... 5

1.2 Overview of Vehicle Electronic Systems .......................... 6
  1.2.1 Electronic Systems of the Powertrain ........................ 9
  1.2.2 Electronic Systems of the Chassis .......................... 11
  1.2.3 Body Electronics ............................................ 13
  1.2.4 Multimedia Systems ........................................... 15
  1.2.5 Distributed and Networked Electronic Systems .............. 15
  1.2.6 Summary and Outlook ....................................... 16

1.3 Overview of the Logical System Architecture ..................... 17
  1.3.1 ECU and Function Networks of the Vehicle .................. 18
  1.3.2 Logical System Architecture for Open-Loop/Closed-Loop  
  Control and Monitoring Systems .................................. 18

1.4 Processes in Vehicle Development .............................. 19
  1.4.1 Overview of Vehicle Development ......................... 19
  1.4.2 Overview of the Development of Electronic Systems ....... 20
  1.4.3 Core Process for Electronic Systems and Software  
  Development ............................................... 24
  1.4.4 Support Processes for Electronic Systems and Software  
  Development ............................................... 26
  1.4.5 Production and Service of Electronic Systems and  
  Software .................................................. 28

1.5 Methods and Tools for the Development of Software for  
Electronic Systems .................................................. 28
  1.5.1 Model-Based Development .................................... 29
  1.5.2 Integrated Quality Management ............................... 31
  1.5.3 Reducing the Development Risk ............................. 32
  1.5.4 Standardization and Automation ............................. 34
  1.5.5 Development Steps in the Vehicle ......................... 37
Chapter 2: Essential System Basics

2.1 Open-Loop and Closed-Loop Control Systems
   2.1.1 Modeling
   2.1.2 Block Diagrams

2.2 Discrete Systems
   2.2.1 Time-Discrete Systems and Signals
   2.2.2 Value-Discrete Systems and Signals
   2.2.3 Time- and Value-Discrete Systems and Signals
   2.2.4 State Machines

2.3 Embedded Systems
   2.3.1 Microcontroller Construction
   2.3.2 Memory Technologies
   2.3.3 Microcontroller Programming

2.4 Real-Time Systems
   2.4.1 Defining Tasks
   2.4.2 Defining Real-Time Requirements
   2.4.3 Task States
   2.4.4 Strategies for Processor Scheduling
   2.4.5 Organization of Real-Time Operating Systems
   2.4.6 Interaction Among Tasks

2.5 Distributed and Networked Systems
   2.5.1 Logical and Technical System Architecture
   2.5.2 Defining Logical Communication Links
   2.5.3 Defining the Technical Network Topology
   2.5.4 Defining Messages
   2.5.5 Organization of Communication and Network Management
   2.5.6 Strategies for Bus Arbitration

2.6 System Reliability, Safety, Monitoring, and Diagnostics
   2.6.1 Basic Terms
   2.6.2 System Reliability and Availability
   2.6.3 System Safety
   2.6.4 System Monitoring and Diagnostics
   2.6.5 Organization of a Monitoring System for Electronic Control Units
   2.6.6 Organization of a Diagnostic System for Electronic Control Units

2.7 Electrics/Electronics and Software Architecture

Chapter 3: Support Processes for Electronic Systems and Software Development

3.1 Basic Definitions of System Theory
3.2 Process Models and Standards
3.3 Configuration Management ........................................ 129
  3.3.1 Product and Life Cycle .................................. 129
  3.3.2 Variants and Scalability .............................. 130
  3.3.3 Versions and Configurations ....................... 131
3.4 Project Management ........................................... 133
  3.4.1 Project Planning ........................................ 135
  3.4.2 Project Tracking and Risk Management ............. 140
3.5 Subcontractor/Supplier Management ............................ 141
  3.5.1 System and Component Responsibilities ............ 141
  3.5.2 Interfaces for Specification and Integration ........ 142
  3.5.3 Defining the Cross-Corporation Development Process .. 142
3.6 Requirements Management ...................................... 143
  3.6.1 Mining, Recording, and Interpreting User Requirements . 144
  3.6.2 Tracking User Requirements ......................... 149
3.7 Quality Assurance ............................................. 150
  3.7.1 Integration and Testing Procedures ................... 150
  3.7.2 Software Quality Assurance Methods ............... 151

Chapter 4: Core Process for Electronic Systems and
Software Engineering ........................................... 153
  4.1 Requirements and Prerequisites ............................. 155
    4.1.1 Shared System and Component Responsibilities .... 155
    4.1.2 Coordination of Systems Engineering and Software
      Engineering .......................................... 155
    4.1.3 Model-Based Software Development .................. 157
  4.2 Basic Definitions and Notations ........................... 158
    4.2.1 Processes, Process Steps, and Artifacts ............ 158
    4.2.2 Methods and Tools ................................ 159
  4.3 Specification of Logical System Architecture .............. 160
  4.4 Specification of Technical System Architecture .......... 163
    4.4.1 Analysis and Specification of Open-Loop/Closed-Loop
      Control Systems ...................................... 167
    4.4.2 Analysis and Specification of Real-Time Systems .... 168
    4.4.3 Analysis and Specification of Distributed and Networked
      Systems ............................................. 169
    4.4.4 Analysis and Specification of Reliable and Safe Systems .. 169
  4.5 Specification of Software Architecture .................... 170
    4.5.1 Specification of Software Components and Associated
      Interfaces .......................................... 171
    4.5.2 Specification of Software Layers .................... 174
    4.5.3 Specification of Operating States ................... 174
  4.6 Specification of Software Components .................... 177
    4.6.1 Specification of Data Model ........................ 177
4.6.2 Specification of Behavioral Model ......................... 178
4.6.3 Specification of Real-Time Model ......................... 180
4.7 Design and Implementation of Software Components 183
  4.7.1 Consideration of Requested Nonfunctional Product Properties 184
  4.7.2 Design and Implementation of Data Model ................ 186
  4.7.3 Design and Implementation of Behavioral Model ........... 187
  4.7.4 Design and Implementation of Real-Time Model .......... 188
4.8 Software Component Testing ................................... 189
4.9 Integration of Software Components ............................. 189
  4.9.1 Generating Program Version and Data Version .......... 190
  4.9.2 Generating Description Files ............................ 192
  4.9.3 Generating Documentation .............................. 192
4.10 Software Integration Testing .................................. 193
4.11 Integration of System Components .............................. 194
  4.11.1 Integration of Software and Hardware .................... 195
  4.11.2 Integration of ECUs, Setpoint Generators, Sensors, and Actuators .............................................. 196
4.12 System Integration Test ........................................ 198
4.13 Calibration ................................................... 200
4.14 System and Acceptance Test .................................... 201

Chapter 5: Methods and Tools for Development ............ 203
  5.1 Offboard Interface Between Electronic Control Units and Tools ...... 205
  5.2 Analysis of Logical System Architecture and Specification of Technical System Architecture .................................. 206
    5.2.1 Analysis and Specification of Open-Loop and Closed-Loop Control Systems .................................. 207
    5.2.2 Analysis and Specification of Real-Time Systems ........ 211
    5.2.3 Analysis and Specification of Distributed and Networked Systems ............................................. 217
    5.2.4 Analysis and Specification of Reliable and Safe Systems ... 222
  5.3 Specification and Validation of Software Functions ........... 230
    5.3.1 Specification of Software Architecture and Software Components ................................................. 232
    5.3.2 Specification of Data Model ................................ 237
    5.3.3 Specification of Behavioral Model Using Block Diagrams 237
    5.3.4 Specification of Behavioral Model Using Decision Tables 240
    5.3.5 Specification of Behavioral Model Using State Machines 242
    5.3.6 Specification of Behavioral Model Using High-Level Languages .................................................... 247
    5.3.7 Specification of Real-Time Model .......................... 249
5.3.8 Validating the Specification Through Simulation and Rapid Prototyping ............................................ 249
5.4 Design and Implementation of Software Functions ................. 259
  5.4.1 Consideration of Requested Nonfunctional Product Properties ............................................. 259
  5.4.2 Design and Implementation of Algorithms for Fixed-Point and Floating-Point Arithmetic ....................268
  5.4.3 Design and Implementation of Software Architecture ...... 286
  5.4.4 Design and Implementation of Data Model ................. 290
  5.4.5 Design and Implementation of Behavioral Model .......... 294
5.5 Integration and Testing of Software Functions ..................... 297
  5.5.1 Software-in-the-Loop Simulations ........................ 298
  5.5.2 Laboratory Vehicles and Test Benches ..................... 299
  5.5.3 Experimental, Prototype, and Production Vehicles .......... 306
  5.5.4 Design and Automation of Experiments ................... 307
5.6 Calibration of Software Functions ....................................... 308
  5.6.1 Offline and Online Calibration Procedures ................. 310
  5.6.2 Software Update Through Flash Programming ............. 312
  5.6.3 Synchronous Measuring of Microcontroller and Instrumentation Signals ................................. 313
  5.6.4 Downloading and Evaluating Onboard Diagnostic Data .... 314
  5.6.5 Offline Calibration of Parameters ......................... 314
  5.6.6 Online Calibration of Parameters ......................... 315
  5.6.7 Classification of Offboard Interfaces for Online Calibration ............................................. 316
  5.6.8 CAL-RAM Management .................................... 322
  5.6.9 Parameter and Data Version Management ................. 325
  5.6.10 Design and Automation of Experiments ................... 326

Chapter 6: Methods and Tools for Production and Service ................................................................. 327

  6.1 Offboard Diagnostics .......................................... 328
  6.2 Parameterization of Software Functions ......................... 329
  6.3 Software Update Through Flash Programming .................... 331
    6.3.1 Erasing and Programming Flash Memory ................ 332
    6.3.2 Flash Programming Through the Offboard Diagnostic Interface ............................................. 332
  6.3.3 Security Requirements ................................... 334
  6.3.4 Availability Requirements .................................. 336
  6.3.5 Boot Block Shifting and Flash Programming ................ 337
  6.4 Startup and Testing of Electronic Systems ..................... 338
## Contents

**Chapter 7: Summary and Outlook** ........................................... 339  
**References** ........................................................................ 343  
**Illustration Credits** ............................................................. 351  
**List of Acronyms** ................................................................. 353  
**Index** .................................................................................. 355  
**About the Authors** .............................................................. 371