Journal of Electronic Testing (2023) 39:267–274
https://doi.org/10.1007/s10836-023-06063-w

Modular Test Kit – A Modular Approach for Efficient and Function-Oriented Testing

Benedikt Jooß[1] · Dieter Schramm[2]

Received: 2 May 2022 / Accepted: 3 April 2023 / Published online: 3 June 2023

Abstract
The software architecture in modern vehicles enables highly networked and innovative systems. At the same time, more and more regulations for type approval of a vehicle have to be taken into account in software development. To ensure consistent quality and compliance with regulations, it is therefore important to test effectively and efficiently. In this article, a modular testing approach is presented that assigns test activities based on special features of a software item. These activities are called modules in the approach. Each module is based on a legal regulation or standard. To ensure efficient and effective testing, the test modules with the highest required identification rate per test objective must always be performed. Carrying out several test modules with the same test objective is not expedient.

Keywords Software testing · Test strategy · Modularity · Test planning · Fundamental test process

1 Introduction

Vehicles are increasing complex due to new, mostly software based, functionalities. This change is also altering the vehicle development process. The verification and validation phase transforms from hardware testing to software testing.

However, software testing faces a conflict of goals between efficiency on one hand and effectiveness on the other. That means that the testing needs to be as short as possible and at the same time confirm the quality as good as possible [1]. The conflict of goals is intensified by the number of software releases. Usually during a development project, the number of software releases is higher than that of vehicle hardware versions. In this context, hardware is understood to be car components such as body parts or chassis components but not Electronic Control Units (ECUs). The development cycles are designed to develop these kind of hardware components . But for SW-development, testing, like all other development steps, must therefore take place more frequently than for hardware development. These steps include development of hardware, such as ECUs, that is directly linked to the Software. While testing it is important to consider any impact on the software by such kind of hardware, e.g. by the use of HiL test racks.

However, there are two factors that increase this conflict of goals. The first factor is technological progress, the second is the change in legislation. The development of new technologies in software and ECU design, architecture and performance increases the amount of necessary testing activities. In addition, new regulations and standards all over the world add effort to these activities as well. But the time for development and especially for verification and validation activities is limited due to customer wishes and competition for market shares [1, 2].

To secure the quality and reduce the risk of vehicle recalls it is necessary to conduct software tests, that effectively detect failures. This means that different test objectives can be achieved by different testing activities but a holistic picture of the quality must be obtained with the help of a test strategy. Depending on the overall test objectives and functionalities of the software item there can be many different strategies found in literature. For instance, Ring [3]

Responsible Editor: Y. Zorian

Benedikt Jooß
benedikt.jooss@cariad.technology

Dieter Schramm
dieter.schramm@uni-due.de

[1] Universität Duisburg-Essen, CARIAD SE, Am Dieb 2, Mönsheim 71297, Germany

[2] Universität Duisburg-Essen, Mechatronics, Lotharstraße 1, Duisburg 47057, Germany

proposes a test strategy for cyber security testing, Sattler [4] and Armstrong [5] for functional safety testing.

To keep testing both efficient and effective, it is neither possible nor targeted to follow for all kinds of functions the same strategy and perform the same activities. That means, that for all software items, an overall test strategy is necessary, which addresses all characteristics and requirements on the software on each architectural test level.

To handle all the different technological and regulative requirements and changing conditions, developing parties need a test strategy to plan all test activities according to the time schedule of a project. The primary objective of such a strategy is efficiency and manageability of the enormous test efforts. It must be taken into account that exhaustive testing of software is impossible, but a strategy should be designed in such a manner that as many error states as possible are detected [6].

To manage the increasing technological, regulatory and normative complexity, it is advisable to use a generic test strategy for all software functions. In a previous paper we discussed that this can be reached by the means of a modular approach [1]. The thrust of this article is to present such a modular approach to test the growing amount of software functions placed on few ECUs. The challenge here lies not only in the increasing number but also in the divergent technical characteristics and the associated legal and normative requirements. To gain this modularity the following questions will be answered:

1. Which special features are relevant for a software project?
2. Which standards and laws are relevant for test and certification of the software?
3. Which test activities are important?
4. How to determine test modules relevant per special features?

Consequently, the paper is divided into three parts. The first part will answer the questions listed above in general. It is presented at which point in time the method can support the fundamental test process (further information can be found in [6]) and how the questions might be clarified for the application of the method for individual software projects. The focus here is on the identification of special features and, associated with this, the hermeneutic method for deriving test activities and modules. The generic development of the method, described in the first section, forms the basis for the further chapters. A technical example expands the generic development in the following part. It shows how the method can be applied for a specific software project. In addition, the example confirms the effectiveness of the method. Finally, in the last part the added value of this method is presented.

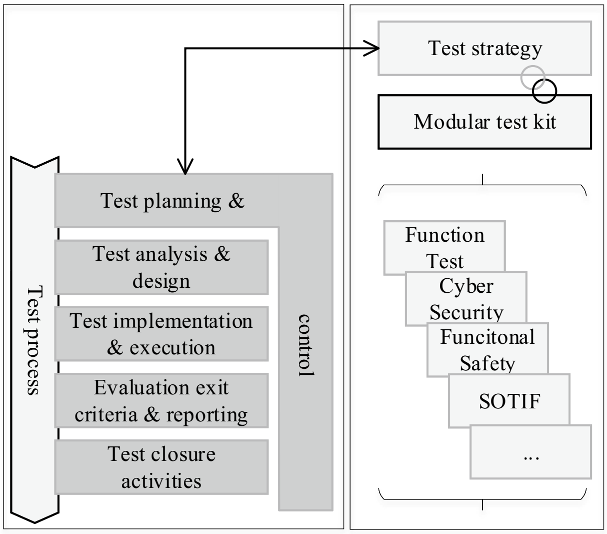


Fig. 1 Use of the MTB during the test planning phase in the fundamental test process [1]

2 Modular Approach for Optimized Test Planning

Software testing is a very complex task and not every approach for testing leads to the defined test objective. The fundamental test process provides a framework for designing an efficient test process. Self-explanatory, the process steps must be individually adapted for each development department or software development project. Different methods or guidelines help to increase the efficiency of testing. Many of these methods aim at tool usage and automation in the process steps test design, implementation, or execution.

The method presented here focuses on the efficient planning during the first process step. This step sets the course for efficient testing and is therefore of great importance for its success. As shown in Fig. 1 test planning depends on a test strategy. The test strategy provides guidelines for the test manager while planning the test activities. For complex development projects, a test strategy must reflect and cover this complexity. Aforementioned, the development of modern vehicles is extremely complex, as they have to fulfil a multitude of requirements, standards and laws. In addition, a huge number of different software technologies for single functions is utilized making the processes even more complex.

To address this, a modular approach was developed as a testing strategy. The basis is built up from special technical features within the software functions. The basic idea is modularization, as it is well known from Volkswagen's vehicle platforms (e.g., the MQB[1] or MSB[2]) [7]. The advantage of platforms is the use of standardized modules that reduce both costs and development time [8]. This advantage is carried over to the test strategy. To further illustrate the idea of vehicle platforms, the modular test strategy will be referred to as the "Modularer Testbaukasten"[3] (MTB) in the following.

Also, the focus during development of the method are software projects for vehicle development the method is generic, adaptable, and suitable for each complex software project. In the centre of the method are the four questions which need individual adaption for each project.

2.1 Which Special Features are Relevant for a Software Project?

The Base of the method are special features. Therefore, it is the first step in adapting the method. The "Verband der Automobilindustrie e.V.[4]" (VDA) defines special features as "[...] features that require increased care and are not already regulated by other processes" [9]. A distinction is made between three different special features[5]:

- BM S = Safety & Security
- BM Z = Certification
- BM F = Function

Contrary to the VDA's definition of special features, the approach presented here deliberately takes features into account that are regulated by other processes and standards (for example functional safety regulated in ISO 26262). Due to the different requirements that are combined within the MTB, the definitions of the VDA must be further refined and specified. Only with a sufficient granularity the desired increase in efficiency can be achieved. For better understanding an example is given. The feature "BM S" includes all test modules that are relevant for safety and security. Therefore, all software functions which depend on safety or security need testing with all test methods required by the standards ISO 26262 [10] and ISO/SAE 21434 [11]. However, it may be that individual functions are affected by either safety or security. For efficient testing it is therefore necessary to further granulate the special features defined by the VDA. The refinement is strongly dependent on the software's purpose and software technologies within the project.

Choosing the right granularity builds success of the method in efficient testing. To identify the right granularity the following three major topics need to be identified:

Hardware and software technologies used by the functions to be tested, e.g., connection to a back-end.

Technical characteristics of the product that can be influenced by the software or system, e.g. tailpipe emissions.

Valid regulations for the type approval of the product, e.g. for mechatronic systems, vehicles.

It is important not only take features into account well known in the area, but also those from other areas. For instance, chassis developers had no need to take tailpipe emissions into account, but with the higher degree of networking between ECUs and SW components the chassis component might influence the emissions. Therefore, this might be an important feature for the MTB of chassis development. As mentioned before hardware might have an impact on the software as well and therefore plays a major role in testing.

Once all features are identified, it is important to group them for efficient testing. This means that if two of the identified features have the same test activities they should be grouped to one special feature. The right granularity is reached as soon as all features must be tested with different activities.

In chapter 3.1 we will give an application example and explain the special features for it (see Fig. 4).

2.2 Which Standards and Laws are Relevant for Test and Certification of the Software?

Next, is the investigation and analysis of standards and acts. The aim is to identify requirements by regulations such as standards or acts regarding software testing. For this purpose, a systematic search algorithm based on the special features is developed. Therefore, keywords are used for each feature and combined to get results. A useful database might be the online portal on European law "EUR-Lex[6]" or the CARB's database on Laws and Regulations[7].

2.3 Which Test Activities are Important?

To gain modularity the MTB uses several test modules for testing. A test module describes one test activity. In their sum, the test modules result in a holistic test run across all levels to be tested. Depending on the special features, the modules are combined in such a way that the test run is most efficient and effective. Especially the measurement

[1] Modularer Querbaukasten

[2] Modularer Standardbaukasten

[3] English: modular test kit

[4] German Association of the Automotive Industry

[5] Special feature in German: Besonderes Merkmal (BM)

[6] https://eur-lex.europa.eu/content/welcome/about.html

[7] https://ww2.arb.ca.gov/resources/documents/laws-and-regulations

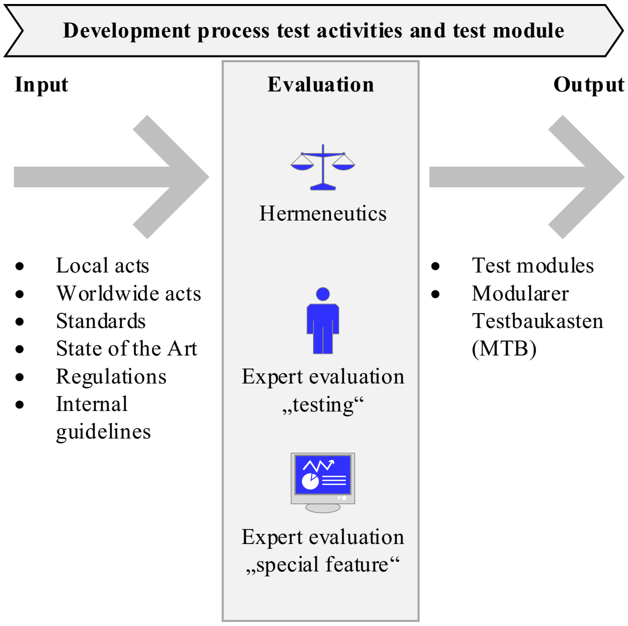

Development process test activities and test module
Input
Evaluation
Output
Hermeneutics
Expert evaluation „testing“
Expert evaluation „special feature“
Local acts
Worldwide acts
Standards
State of the Art
Regulations
Internal guidelines
Test modules
Modularer Testbaukasten (MTB)

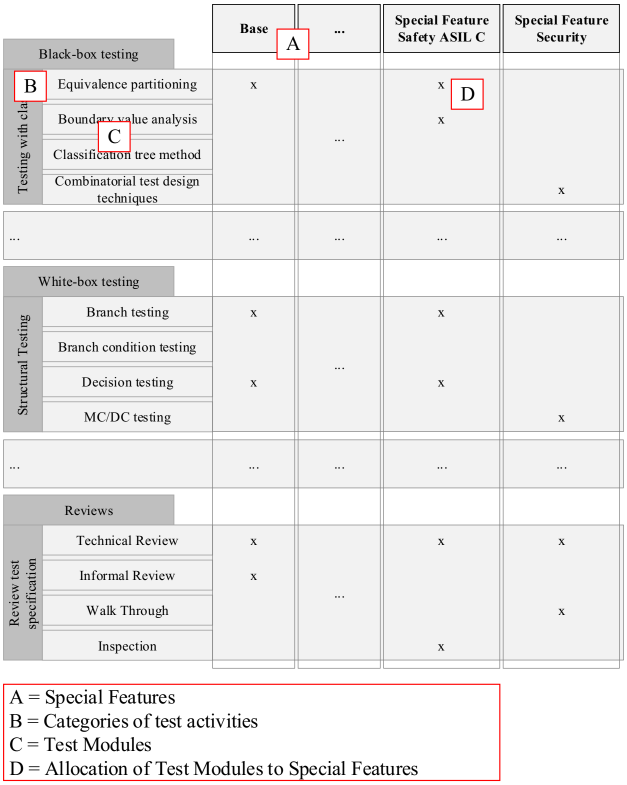
Fig. 2 Development of test modules

of efficiency in testing is a well-known issue for software testers in the automotive industry. In this industry the development cycle is very stringent and uses several milestones planned beforehand the project start. As a developer and tester one has to keep these in mind to plan their work. Therefor the testing needs to be efficient. A metric could be the amount of tests or tested systems within the given timeframe. In addition these metrics also need to address legal requirements. This means only the amount of tests that are relevant from a legal perspective count in this metrics. A tester will keep in mind, that only features which a new or reworked need to be tested for each milestone.

A test module describes the test objective as well as the test design and test method. As shown in Fig. 2 the development process for a test module is based on three major steps. Hermeneutics, and two expert evaluations. One expert is from the field of testing, the other one is an expert for one special feature.

In the first step the standards, acts and the latest technologies serve as input. These are analysed in detail with a hermeneutic approach. The main part of the test modules is based on the content of ISO/IEC/IEEE 29119-4 [12], but other content, as instance, out of ISO 26262 [10], is also considered.

After this step the experts examine the modules for correctness, feasibility, and completeness as well as the correct interpretation of the standards by legal hermeneutics. The output are the single test modules which build up the modular test kit.



		Base	...	Special Feature Safety ASIL C	Special Feature Security
Black-box testing					
Testing with cla	Equivalence partitioning	x	...	x	
	Boundary value analysis			x	
	Classification tree method				
	Combinatorial test design techniques				x
...		...	...	...	...
White-box testing					
Structural Testing	Branch testing	x	...	x	
	Branch condition testing				
	Decision testing	x		x	
	MC/DC testing				x
...		...	...	...	...
Reviews					
Review test specification	Technical Review	x	...	x	x
	Informal Review	x			
	Walk Through				x
	Inspection			x	

A = Special Features
B = Categories of test activities
C = Test Modules
D = Allocation of Test Modules to Special Features

Fig. 3 Structure of the MTB (exemplary cutout)

2.4 How to Determine Relevant Test Modules Per Special Features?

Figure 3 shows the structure of the MTB. The special features [A in Fig. 3] indicate which test modules [C] are relevant to test a software item. To gain legal compliance the test modules are linked to a special feature, e.g., the ISO/SAE 21434 provides requirements for testing of the feature security.

However, different modules can have the same overall test objective, but are associated with a higher effort to gain. Therefore, these modules are categorised Fig. 3B. For the most efficient testing it is necessary to only use the module with the least effort to reach a test objective. Due to legal requirements, it might be, that a test module with more effort must be used to gain the objective. The reason is that the activities with more effort usually have a higher identification rate for failures.

As an example, the test modules “informal review” or “inspection have the target to find errors in the code by a manual review. However, the informal review is less time consuming than the inspection. On the one hand for functions with no special features the informal review is sufficient, on the other hand the ISO 26262 demands an inspection for functions classified with the feature ASIL D.

The efficient testing of modules is the greatest advantage of the MTB during test planning compared to conventional test strategies, in which one strategy is developed for each feature. In the past this was mainly done for safety testing. To be more effective and efficient the MTB needs implementation in existing structures and tooling. Another important advantage is that the traceability between test case, test strategy and standards and acts can be increased by integrating the test kit into the existing test tool chain.

3 Modular Approach in Practice

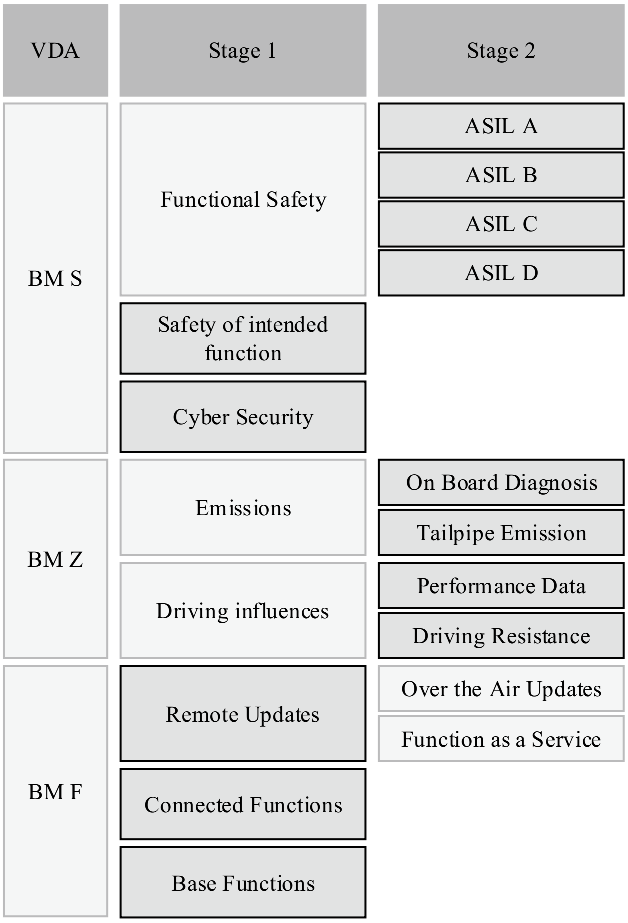
The modular approach to efficient test planning has so far been presented in theory. To get a better picture of the MTB, an application example and the confirmation of the method will be presented in this section.

3.1 The Modular Test Kit for Chassis Development

The planning of software tests in the fundamental test process must meet many requirements. The transforming conditions, as instance, new software technologies, architecture or new standards and laws must be considered while testing. Additionally, software testing must still be efficient and target oriented. The MTB was developed to resolve this conflict. As test planning tool, this modular approach takes the changing conditions into account and makes testing more efficient. It can be adopted for the usage in different areas. Typical areas of application are either individual development projects, departments, or an entire company. In the following, we demonstrate the application using the example of vehicle chassis development.

In the past, one ECU controlled one functionality, e.g. the ABS ECU. At this point the biggest distinction was made between software functions with and without Safety relevance. However, as discussed in [1] the number of ECUs has been decreasing in recent years, whereas the number of software functions has been increasing. Due to this changed architecture, more and more functions are located on one ECU, for instance chassis functions and powertrain functions. Among other things, this leads to the fact that the features assigned to the functions are no longer only FuSi or non-FuSi, but that further requirements, for example OBD, are placed on the functions. For this reason, the MTB for planning the scope of the test based on special features. Fig. 4 shows an exemplary extract of the MTB used for chassis development.

The first step is the identification of the special features. In Fig. 3[A] shows three examples. Base means the test strategy for functions with no additional special features, the other two are Safety ASIL D and Security. These would both be BM S according to VDA [9]. In Fig. 4 a full



VDA	Stage 1	Stage 2
BM S	Functional Safety	ASIL A ASIL B ASIL C ASIL D
	Safety of intended function	
	Cyber Security	
BM Z	Emissions	On Board Diagnosis Tailpipe Emission
	Driving influences	Performance Data Driving Resistance
BM F	Remote Updates	Over the Air Updates Function as a Service
	Connected Functions	
	Base Functions	

Fig. 4 Special features in chassis development

overview of the special features for chassis development is given. Starting point are the special features defined by VDA. For the application for chassis development the three features are detailed by up to two stages. Stage 1 are the main special features; Stage 2 are subordinate features. For testing functional safety testing, the granularity is not sufficient because an ASIL A function needs different activities than ASIL D. Therefore stage 2 is used for the safety features of the MTB. However, “function as a service” and “over the air updates” use the same technologies and therefore can be group to the special feature “remote updates”, tested with the same test modules.

Each of the special features shown in Fig. 4 needs to be tested with different test activities. These are based on legal requirements from different standards and laws. These laws need to be identified and interpreted to extract the test activities.

Therefore the next step is the analysis of standards and acts on the baseline of the special features. For chassis development 32 regulations were identified. These are listed in Table 1. The number is variable as new regulations come into effect or will be replaced. For this reason, the analysis is an ongoing process.

Table 1 Extract of relevant standards and acts

Regulation	Scope
ISO/IEC/IEEE 29119	Software Testing
IEEE 1028-2008	Software Reviews
ISO 20246	Work product reviews
Automotive SPICE	Development Processes
AUTOSAR	ECU and SW Development
ISO 26262	Functional Safety
ISO/PAS 21448	Safety of intended function
ISO/SAE 21434	Cyber Security
UNECE R155	Cyber Security
ISO IEC 15408	Cyber Security
NHTSA Cybersecurity Best Practices for Modern Vehicles	Cyber Security
GB 201-5	Software Updates
UNECE R156	Software Updates
ISO 24089	Software Updates
CARB CCR1986	On Board Diagnosis
ISO 15031	On Board Diagnosis
REGULATION (EU) 2017/1151	Emissions
REGULATION (EU) 2017/1154	Emissions
REGULATION (EU) 2017/1347	Emissions
REGULATION (EU) 2018/1832	Emissions
REGULATION (EC) No 715/2007	Emissions
UNECE R49	Emissions
UNECE R83	Emissions
UNECE R154	Emission
VO EU 2018/858	Type approval
GSR (EU) 2019/2144	Type approval
UNECE R0	Type approval
UNECE R13	Braking systems
UNECE R13H	Braking systems
UNECE R79	Steering systems
UNECE R140	Stability systems
UNECE R141	Tyre pressure systems

The finalization of information gathering represents the starting point for the development of the test modules Fig. 3C. A module contains information on the test objective, entry criteria, test design, test methods, exit criteria and a reference to norm and standards covered by the module. In this extract the categories Fig. 3B used for prioritization are black-box testing with the subcategory “Testing with classes”, white-box testing with the subcategory “Structural testing” and reviews with the subcategory “Review test specification”. The more requirements, features and thus modules exist, the more categories and subcategories are needed. Within each category are test modules that have comparable test objectives. The modules in the category “Testing with classes” aim to confirm that the input and output values of a value range lead to the correct functionality by sorting them into classes.

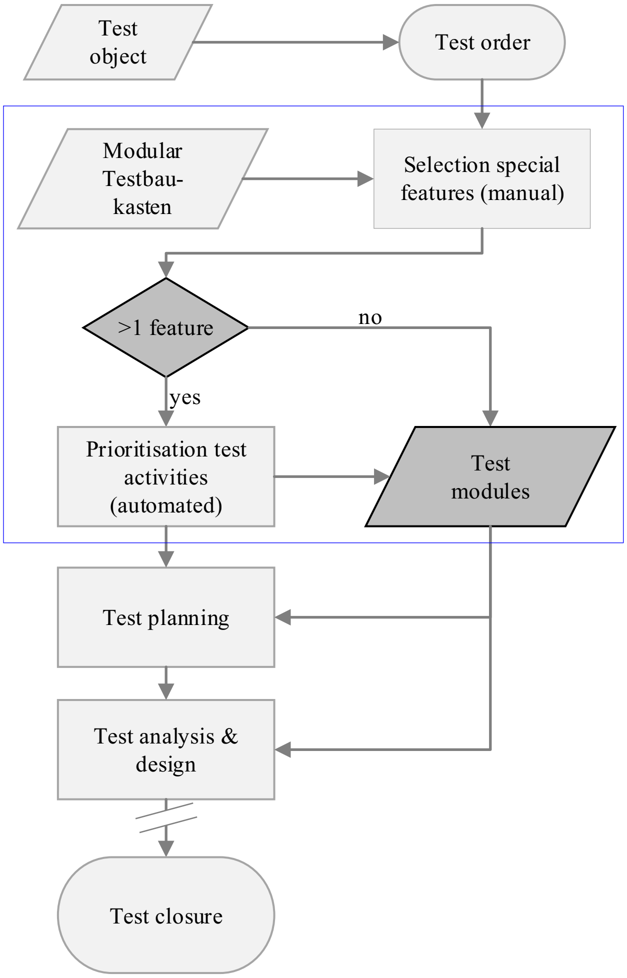


Fig. 5 Application of the MTB during the test process

Besides this the MTB overview indicates which test modules are relevant for which special feature Fig. 3D. In the example the test design method for the special feature “Base” is equivalence partitioning whereas the method for the feature “Safety ASIL C” is equivalence partitioning and boundary value analysis. Due to different requirements the boundary values need more attention but need more effort for testing.

The quality of the tests carried out is difficult to assess. One can argue, that the amount of tests carried out is sufficient but in addition it is necessary to perform reviews and specially designed tools to evaluate the software. With this the quality will be increased.

3.2 Application of the Modular Testbaukasten

As mentioned before the MTB is part of the test strategy therefore applied during the planning phase of the fundamental test process. During test planning, the MTB serves as a support for identifying the relevant test activities. In

Fig. 5 the application process of the MTB is shown. As soon as the test order is placed the MTB is used as input to identify test objectives. Therefore, the special features of a software function are identified based on the SW requirements. Depending on the number of special features the prioritization is either needed or a direct overview over the test modules serves as output. This output is contemporaneous the input for the process steps test planning and test analysis and design of the fundamental test process.

Prioritization is one essential step to make testing efficient. This is because only the test modules with the highest identification rate are used for test design, thus covering those with lower identification rate and less effort. The easiest way to illustrate this is with an example.

In the example in Fig. 3 Special Feature “Safety ASIL C” requires an inspection. Special Feature “Security”, on the other hand, requires a walk through. Yet, for the coverage Safety requires the more time-consuming Modified condition/decision coverage (MC/DC) compared to branch coverage. For legal compliance it is necessary to use MC/DC and inspection as methods for testing. For efficiency reasons only the activities with the higher identification rate are used.

The use of the MTB adds different values into testing. The most value is added by resolving the conflict of goals between efficiency and effectiveness. As it takes different standards and acts into account it is ensured the legal compliance. Secondly the method also takes a variety of technical aspects into account, such as the changing ECU and SW technologies. In addition to legal compliance effectiveness in testing is achieved.

To accomplish not only the effectiveness but also the efficiency the MTB is based on a modular approach. In this way only the tests which are advantageous to detect faults or gathering information about the test object are performed. Overall, the application leads to reduced test effort by increasing the informative value at the same time.

This modular approach is shown using an example on the software level. But it is also usable on different levels such as the system. As each level has its own requirements these need to be extracted separately. However the performance of testing can increase when only the most promising tests will be carried out on each level, so in the end the whole vehicle is tested the most efficient and effective way.

4 Conclusion

In this paper a modular approach to testing SW in vehicle development was introduced. This approach addresses the challenge of the changing, more complex development environment. On the one hand, the technologies in software development are shifting rapidly. In recent years the number of software items has increased steadily as well as the networking and complexity of ECUs. On the other hand, more and more legal requirements must be considered while developing vehicles. To maintain development cycles and improve quality, it is important that software is tested in a target-oriented manner. To resolve the conflict between efficiency and effectiveness, a modular test kit, called “modularer Testbaukasten”, is suggested. This serves as a test strategy in the test project and is used as input during test planning. Based on special characteristics of a SW item, test modules are specified at different test levels. The test modules are the foundation of test cases. The development method of the kit answers the following questions:

1. Which special features are relevant for a software project?
2. Which laws are placed on the software of the project?
3. How do these laws influence the special features and test modules?
4. How are the test modules allocated to the special features?

One distinguishing feature of the method is the development of the special features. The basis is the VDA process description “special features”, which specifies three special characteristics. However, as this is not sufficiently granular, these characteristics are subdivided and restructured.

Furthermore, the importance of researching for regulations, standards and acts was highlighted. These regulations are, one the one hand, the foundation of test modules and, on the other hand, for the link between special features and test activities. By categorising and assigning identification rates to the test modules, they can be prioritised during test case development. Intelligent links between the test modules ensure that tests are carried out efficiently and in compliance with the law.

The presented MTB thus resolves the tension between efficient and effective testing while the development environment is changing rapidly. Particular attention is paid to compliance with standards and laws that go far beyond ISO 26262. By directly linking standards to test modules, traceability is ensured at the same time, which serves as proof that a standard is fulfilled.

Finally, an application example was presented. This example includes 13 special characteristics and 31 associated standards. It was again emphasised how prioritisation can be used to maintain efficiency and effectiveness even with several special characteristics.

Acknowledgements We acknowledge support by the Open Access Publication Fund of the University of Duisburg-Essen.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data Availability Data not available due to legal restrictions.

Declarations

Conflicts of Interest The authors declare no conflict of interest. The authors have no relevant financial or non-financial interests to disclose.

References

1. Jooß B, Schramm D (2022) Komplexität von Softwaretests in der Fahrwerksentwicklung. Automobil Technische Zeitschrift, Nr. 04:44–47
2. Schäuffele J, Zurawka T (2013) Automotive Software Engineering - Grundlagen, Prozesse, Methoden und Werkzeuge effizient einsetzen, 5. Auflage, Wiesbaden: Springer Vieweg
3. Ring M (2019) Systematische Security-Tests von Kraftfahrzeugen, Dissertation. Universität Ulm, Ulm
4. Sattler K (2015) Methodik für den Systemtest in der integralen Fahrzeugsicherheit, Dissertation. Otto-von-Guericke-Universität Magdeburg, Magdeburg
5. Armstrong K (2007) Validation, verification and immunity testing techniques. In: IEEE International Symposium on Electromagnetic Compatibility Honolulu HI USA IEEE 2007. pp 308–313
6. International Software Testing Qualifications Board (2019) Certified Tester Foundation Level Syllabus, Version 2018 V3.1. Int Software Test Qualifications Board
7. Pander J (2012) Gleich ist geil - Neues Konstruktionssystem bei VW. Der Spiegel February 2012
8. Cameron BG, Crawley EF (2014) Crafting platform strategy based on anticipated benefits and cost. Advances in product family and product platform design. Springer, New York, pp 49–70
9. Verband der Automobilindustrie e.V. (VDA) (2020) Qualitäts Management Center (QMC), Porzessbeschreibung Besondere Merkmale (BM), 2. Hrsg., Berlin, Frankfurt am Main: Heinrich Druck + Medien GmbH
10. ISO 26262 (2018) Road vehicles — Functional safety Parts 1 - 12
11. ISO SAE DIS 21434 (2020) Road vehicles — Cybersecurity engineering. Under development
12. ISO/IEC/IEEE 29119 (2016) Software and systems engineering — Software testing Part 1 - 5

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Benedikt Jooß received the Master of Science degree from the University of Duisburg–Essen, Germany, in 2017, where he studied automotive engineering and management, specializing in mechanical engineering. Following the master's thesis, which was conducted with the Dr. Ing. h.c. Porsche AG, Stuttgart, Germany, he worked as a test manager in the automotive software development. In 2018 he rejoined the Dr. Ing. h.c. Porsche AG, Stuttgart, for his Ph.D. in cooperation with Chair of Mechatronics, University of Duisburg–Essen. His thesis contributes to the optimization of test strategies in the field of chassis development. Since 2020 he works full-time at the CARIAD SE as PMT Achitect Testing and Project Management Specialist.

Dieter Schramm received the Ph.D. degree in engineering from the University of Stuttgart in 1986, where he studied mathematics. From 1986 to 2004, he worked in management positions in research, development and technical marketing for leading international automotive suppliers. In 2004, he was appointed as a Full Professor and the Head of the Chair of Mechatronics, University of Duisburg–Essen. His research interests are electrified and alternative fuel driven automobiles, driver assistance systems, vehicle dynamics, and cable driven manipulators. Along with his research activities, he is the director and a partner in several companies in the field of research and post graduated education. In 2015, he was awarded Dr. h. c. by the University of Miskolc, Hungary.