

# Significant Reduction of Validation Efforts for Dynamic Light Functions with FMI for Multi-Domain Integration and Test Platforms

Dr. Stefan-Alexander Schneider      Johannes Frimberger  
BMW AG, 80788 Munich, Germany  
stefan-alexander.schneider@bmw.de, Johannes.JF.Frimberger@bmw.de

Michael Folie  
IPG Automotive GmbH  
Agnes-Pockels-Bogen 1  
D - 80992 Munich, Germany  
michael.folie@ipg.de

## Abstract

Modern advanced driver assistance systems (ADAS) provide a significant increase in comfort and safety. In many cases, a single vehicle, today, contains more than one assistance system, while the trend to use ADAS continues to grow. At the same time, the number of systems that perform control interventions with safety-relevant functions increases as well. From an overall perspective, it must be assured that the driver assistance systems – individually as well as in the way they interact – function flawlessly in any driving situation and with any driver at the wheel anywhere in the world. This results in an increased development and testing effort for modern ADAS in general. In-development testing based on virtual test driving offers an approach to a solution that allows the validation effort to be significantly reduced while meeting the requirements for safety-relevant functions in the vehicle associated with ISO 26262. This is particularly evident when developing and testing new light functions, which in real-world road tests can often be performed only in conditions of darkness. Dynamic Light Functions are defined as the situation-dependent headlight adjustment consisting of cornering light, headlight leveling and (glare-free) headlight assistance. This paper presents this new methodology.

*Keywords: FMI; AUTOSAR; CarMaker; Sil; Xil; Dynamic Light Function*

## 1 Motivation and Current State of Technology

Today, new light functions are developed in the early stage based on models. These models are subsequently tested for functional performance in the Model-in-the-Loop (MiL) process using simple test cases. Based on requirement specifications, the Simulink models are subsequently migrated to C Code and put into an electronic control unit as an AUTOSAR software component (AUTOSAR-SWC). The approval occurs strictly in the Hardware-in-the-Loop (HiL) test based on real-world measurement data or stimuli. After approval of the software on the HiL-/(SiL-) test rig, testing in the vehicle commences. When using this procedure, though, feedback in the whole vehicle takes place only at a very late point in time. Today, in the Dynamic Light Functions project, dynamic SiL tests are performed based on recorded test drives and manually generated scenarios. Improvement potential still exists here with respect to certain aspects:

1. The open-loop tests, in the case of modified controller software, are reusable only to a limited extent.
2. The transfer to different vehicle configurations or new vehicle variants is possible only to a limited extent. The low availability of prototype vehicles in the various development stages does not permit any tests to be run on the physical prototype.
3. The additional evaluation scenarios which may emerge in the analysis of the interaction of the controller and controller distance have to be recorded in the vehicle test again. The

reproducibility of the scene and the large number of real-world variants is often unfeasible.

By using a vehicle model and corresponding test automation real-world effects can be examined early in the SiL approach. This SiL approach can be integrated into the seamless X-in-the-Loop (XiL) development method [1]. The limited availability of the HiL test rig resource (and the associated late availability of the whole vehicle) could thus at least be partially alleviated.

## 2 FMI Brings Light into Darkness

The Functional Mock-up Interface (FMI) [2] offers the possibility to make use of C Code and Simulink models in a standardized form as so-called Functional Mock-Up Units (FMU) in integration tools. These FMUs can be used for interaction chain tests and design decisions at a very early stage of development. In addition, they are available in the validation chain for dynamic residual bus simulation models. Furthermore, the fast and easy connection of an electronic control unit or an array of ECUs is possible as well.

An FMU can be generated quickly and at low cost using simple means. FMUs consist of a ModelDescription, i.e. a description of the interface (signals and parameters) and a container with a dynamic library for the relevant operating system. Therefore, the generation of such an FMU requires some type of interface specification (e.g. the arxml format [3]) and the associated source code and/or a previously compiled library for the relevant target systems (Figure 1). Even though some tools already offer the possibility to generate FMUs, the objective here is to point out a license-free alternative for quick implementation that can also be used to carry out new requirements with ease.

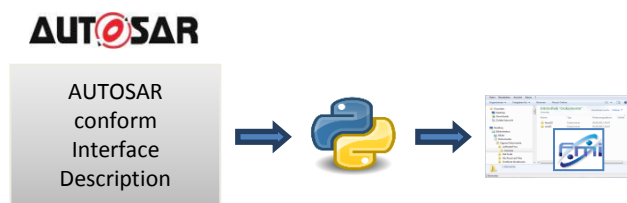


Figure 1: FMU generation.

The ModelDescription is an XML format specified in the FMI standard. All the information required to generate it is available in the AUTOSAR

specification in the form of an arxml description and is therefore retrievable in an automated form. In addition to the interface description, the AUTOSAR XML specification contains the function calls for initialization and the individual calculation steps as well as their step widths. Furthermore, the interface description provides information about the names of the buffers (global variables) for inputs and outputs as well as coding parameters. The FMU interface merely provides an array with all model variables. Consequently, they have to be additionally mapped to the global variables in order to enable the controller to process the current data.

To simplify the integration of the controller into different development environments, a floating-point to fixed-point conversion was directly implemented in the wrapper, again automatically generated from the existing signal description.

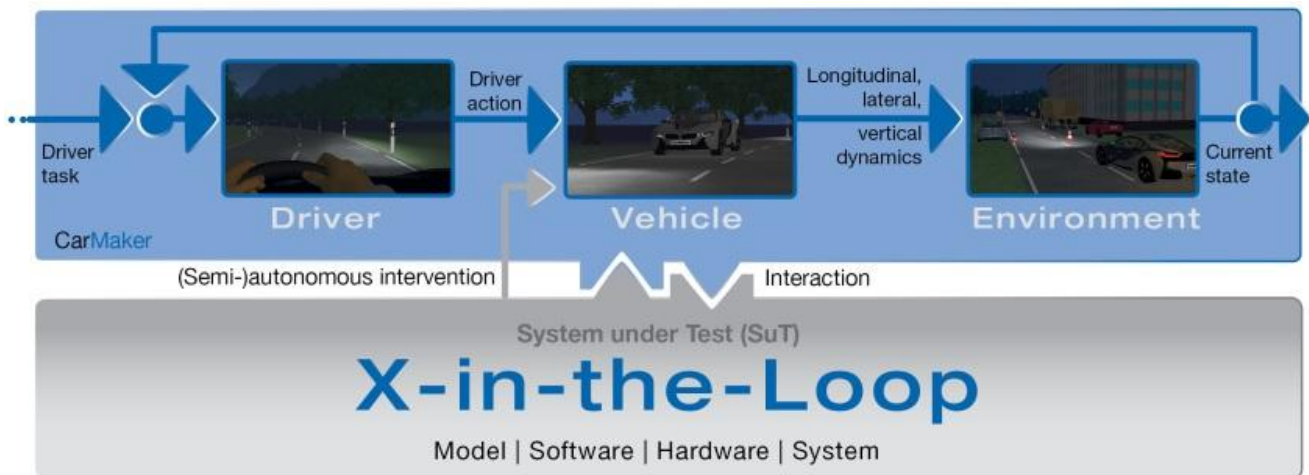
By using this information it is now possible to generate code which initially sets the parameters of the FMU and in the subsequent calculation steps manages the inputs and outputs and calls the actual controller. Together with the controller source code it is possible to compile dynamic libraries which together with the ModelDescription yield a functional FMU.

The FMUs generated in this way can be imported, linked and simulated in various integration tools.

## 3 Consistent Tests through Efficient Integration of Standardized Components

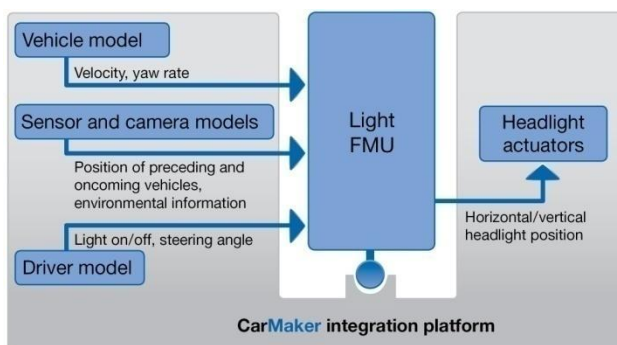
To create the link between the FMUs and the whole vehicle, additional plausible models of vehicle physics and the vehicle environment are necessary. For this purpose, the seamless X-in-the-Loop (XiL) approach was rigorously implemented in CarMaker (Figure 2).

The XiL integration makes it possible to integrate and comprehensively validate all relevant system components, either as models, ECU software or hardware, into the whole virtual and real-world vehicle. As an open integration platform, CarMaker offers interface architecture that is adapted to the vehicle development and in which the FMU approach can be efficiently used. Models, software components and real-world vehicle components are integrated by mouse click into so-called digital prototypes – from the single component through to interlinked (networked) systems.



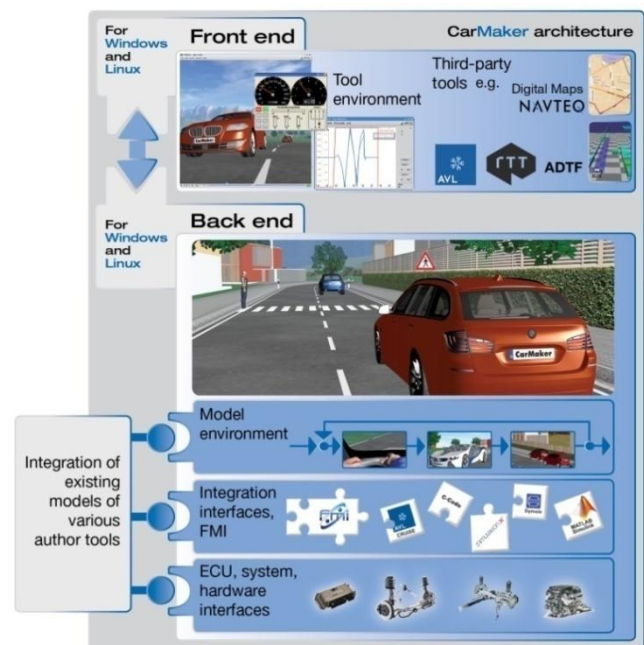
**Figure 2:** X-in-the-Loop enables the early verification and validation of systems. CarMaker offers the appropriate interfaces to integrate all relevant components and systems into a whole virtual vehicle.

Various components such as the powertrain, assistance and control systems etc. as well as instrument and operating concepts may be integrated as needed [4]. The virtual integration creates the prerequisites to check what effects the tested components have on the overall performance of the whole vehicle. Unfavorable design decisions and even functional faults can thus be detected earlier and allow corresponding design decisions to be optimized [5]. The digital prototypes are verified as a total system in virtual test driving and serve as input for our light assistance FMU (Figure 3).



**Figure 3:** Interfacevariables to implement the light FMU into the CarMaker integration platform.

In practical terms, we receive signals from vehicle dynamics, additional environment sensors and camera systems as well as driver inputs such as steering angle or manual headlight requests. In addition to the powerful vehicle and driver model, CarMaker encompasses a complete environment simulation, consisting of roads with corresponding static traffic objects and moving traffic as well as environment sensors and digital maps (such as NAVTEQ or Google Earth). As a result, the test environment is modeled at a high level of realism, see Figure 4.



**Figure 4:** Function models and software components can be integrated into the integration environment either as Functional Mock-Up Units or hardware components. In addition, CarMaker allows navigation systems and software development tools to be connected as well.

Furthermore, as a test platform, CarMaker offers a maneuver description that is based on the principles of real-world road tests. Complex open- and closed-loop tests are carried out as maneuver instructions. The virtual test drive is reproducible and can be easily modified as needed. This makes it possible to evaluate new developments in virtual test driving and to tangibly experience them in realistic driving situations. The test cases for the light functions are migrated from test driving into the CarMaker language and carried out in the TestManager (Figure 5).

The screenshot shows the 'Test Manager' window in CarMaker. It displays a list of test cases under 'Global Settings'. The test cases are categorized into 'Driving\_Ser', 'Steer\_Ser', and 'Backwards\_Ser'. Each category has several variations. The table below represents the data shown in the screenshot:

Item Description	Par1	Par2	Par3	Par4	Criteria	Res.Date	Result
<b>Driving_Ser</b>							
Variation 1	30					15:25:32	●
Variation 2	50					15:25:41	●
Variation 3	90					15:25:44	●
Variation 4	120					15:25:44	●
<b>Steer_Ser</b>							
Variation 1	30	15				15:25:49	●
Variation 2	50	10				15:25:54	●
Variation 3	90	3				15:25:56	●
<b>Backwards_Ser</b>							
Variation 1	5	25				15:25:59	●
Variation 2	10	20				15:26:01	●
Variation 3	20	10				15:26:03	●

**Figure 5:** Test cases from real test driving.

By combining in-development testing, the standardized FMI and the multi-domain integration and test platform the development and test effort can be significantly reduced. A maneuver catalog that has been defined once is thus available for design, verification and validation purposes in all stages of the development process. The maneuver catalog is specifically extended according to the development stage in order to comprehensively verify and validate the design and implementation of the ECU software. This allows fast, convenient exchanges and testing of vehicle components as FMUs, both within the BMW organization and in collaboration with suppliers. In other words, the OEM and the suppliers are able to use the same models and functions for architectural decisions and virtual test driving.

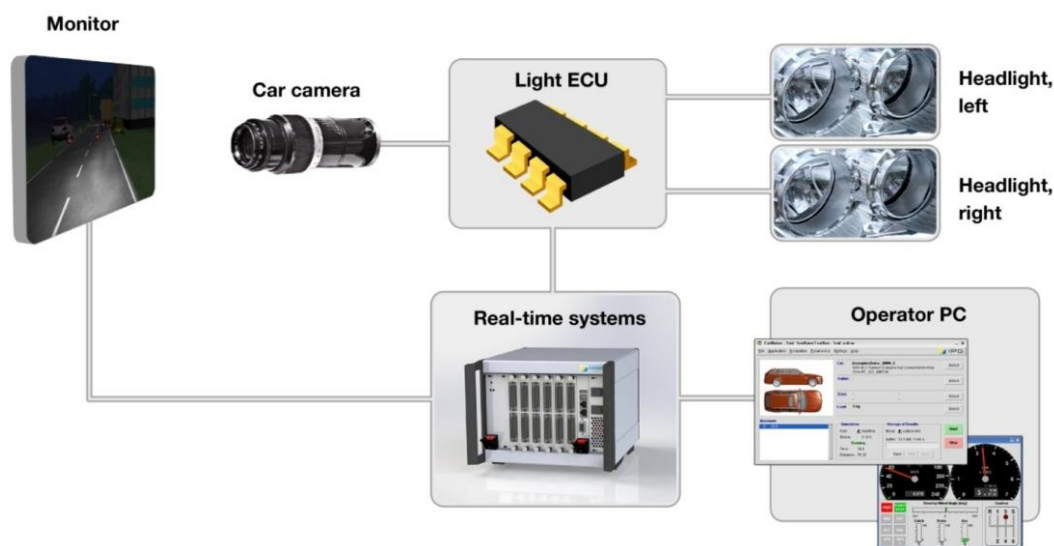
## 4 Use in the Development Process

Last but not least, this is where the advantages of virtual test driving prove their viability (Figure 5): providing test scenarios (through to complex customer use cases) which can be modified as needed and are reproducible on the one hand and reproducible test results on the other.

In the development process, FMUs can be used in an SiL environment at an early development stage. Software modifications and their effects on existing test cases can be easily realized and tested due to the standardized components.

Aside from this range of applications, FMUs can also be used as substitutes for electronic control units not currently in existence in an array or composite of ECUs together with existing, real-world ECUs. This way, they help to make tests on the whole system possible at an early development stage.

In the case of the dynamic light function, in addition to numerous vehicle dynamics parameters provided by CarMaker, camera-based information is processed as well. This information is either provided by virtual sensors or – ideally – by the real ECU of the camera. To enable the ECU to actually deliver realistic values, virtual driving scenes including other traffic from CarMaker are captured and processed by the camera in a darkened room. This means that the virtual test drive is filmed by the real-world vehicle camera in order to run a so-called interaction chain test. The camera is used to detect the presence of other vehicles within the blinding range of the light. This information is forwarded to the light function in order to activate the glare-free range of the high beam (Figure 6).



**Figure 6:** Set-up of an HiL test rig for the event chain test: Operator PC and HiL system, real-world or simulated controls for the light, light ECU, headlights with headlight actuators, camera with integrated ECU and monitor for filming.

This adds significant value compared to the tests that have been run up to now, in which only the ECU could be served with a residual bus simulation but the specific signals and functions of the camera could not be used.

Linking real-world headlights with the environment described above now represents the next step. This allows the entire interaction chain, from the camera through to the actuator, to be modeled and tested. As a result, detailed statements about the functional performance of the whole system can already be made on the test rig, which makes it possible to considerably reduce the validation effort on the real-world vehicle. Figure 7 shows a snapshot from a video [6] for comparison of a real-world and a corresponding virtual test driving situation.



**Figure 7:** Comparison of the light function: video with glare-free high beam and its simulation.

## 5 Conclusions

By using both the standardized interface specification FMI and an integration and test platform (from MiL, to SiL and through to HiL) the current development process can be made significantly more efficient and the existing gap in the area of the SiL tests can be closed. In such a development environment the presented control loop consisting of the “the car on the road” and “light events detected by the camera” can be studied in detail. The control loop of interest can then be extended step by step, e.g. additionally including the position of the headlight actuators, and/or the simulated illumination of the headlights. A reduction of the testing effort associated with the large number

of variants can be achieved by the DoE (Design of Experiment) method in combination with carefully selected real-world road tests [7]. For this purpose, critical corner case parameters can simultaneously be tested in the design space to be validated and, in parallel, simulated in virtual test driving. The models on which these tests are based are validated. Proceeding from this database, additional real-world road tests could be made superfluous by the continuing validation being supported by the results thus obtained. This harbors further remarkable potential for reducing the validation effort and, ultimately, controlling this effort as well.

## References

- [1] Schick, B.: Mission V-Process enhancement by integrated vehicle performance evaluation within an entire X-in-the-Loop process, Keynote SIAT ARAI 2013
- [2] FMI Development Group: FMI – The Functional Mock-up Interface. [\[https://www.fmi-standard.org/\]](https://www.fmi-standard.org/): [January 21, 2014]
- [3] AUTOSAR Development Partnership: AUTOSAR, [\[http://www.autosar.org/\]](http://www.autosar.org/): [January 21, 2014]
- [4] Martinus, M., Deicke, M. and Folie, M.: Virtual Test Driving: Hardware-Independent Integration of Series Software, ATZ Elektronik 10/2013
- [5] Schneider, S.-A., B. Schick and H. Palm: Virtualization, Integration and Simulation in the Context of Vehicle Systems Engineering, embedded world 2012, Nuremberg 2012
- [6] BMW UK: BMW Intelligent Headlight Technology: Long Version: [\[http://www.youtube.com/watch?v=dvPZ3H1Vm4\]](http://www.youtube.com/watch?v=dvPZ3H1Vm4): [January 21, 2014]
- [7] Palm, H., Holzmann, J., Schneider, S.-A., Kögeler, H.M. and Pfister, F.: The Future of Car Design: Optimisation with Systems Engineering, ATZ Elektronik 06/2013