# IN THE LOOP

**Vehicle automation and advanced driving assistance systems are being streamlined using ANSYS SCADE capabilities.**

**By Frank Köster**, Head of Department, Institute of Transportation Systems, German Aerospace Center, Braunschweig, Germany

Vehicle automation offers the potential to substantially reduce the hundreds of thousands of deaths — including motorists, motorcyclists, bicyclists and pedestrians — that occur each year around the world in automobile accidents. This automation provides the opportunity to improve traffic flow, increase driver comfort, and reduce fuel consumption and emissions. Many vehicles already have automated lighting, intelligent parking-assist systems, proximity sensors with alarms and other automated systems. However, there are many technical, regulatory and legal obstacles to fully autonomous vehicle operation or self-driving cars. Only a few U.S. states currently allow semi-autonomous vehicle operation (for example, systems that can take over control of the vehicle if the driver makes a mistake), and fully autonomous, driverless vehicles are not allowed anywhere in the United States. For the foreseeable future, vehicle automation and advanced driving-assistance systems (ADAS) will supply assistance and automation ranging from full control by the driver to full control by the automation system. When developing these systems, a major challenge is transitioning between these different levels of automation,

## Vehicle automation offers the potential to substantially reduce hundreds of thousands of deaths.

including transitions initiated by the driver and those initiated by the automation system.

The Institute of Transportation Systems at the German Aerospace Center (DLR) is working in cooperation with leading automobile original equipment manufacturers (OEMs) to develop vehicle automation and ADAS that will overcome this and other challenges. DLR is combining its technological expertise with psychological and ergonomic research to produce vehicle automation systems that can be tailored to the capabilities and expectations of each driver. Systems that are currently under development involve integration of the

▲ Dynamic driving simulator

driver and automation system so that, for example, when the limits of the automation system are reached, control is handed back to the driver. In this situation, the driver needs to be presented with the right information at the right time by the human–machine interface (HMI) so that he or she can safely resume control of the vehicle. DLR uses ANSYS SCADE Suite and ANSYS SCADE Display to develop HMIs in the model environment using prebuilt and specific components. By simulating behavior using the model, it is possible to identify and correct defects as well as gain critical insights early in the design process to rapidly improve system performance.

### ROLE OF THE HUMAN–MACHINE INTERFACE

One example of how the HMI operates occurs when the automation system cannot sense the lane markings due


▲ HMI on heads-up display for highly automated driving

**By simulating behavior using the model, it is possible to identify and correct defects as well as gain critical insights early in the design process to rapidly improve system performance.**

# The time required to develop and validate the HMI was been substantially reduced by moving the development process to SCADE Suite and SCADE Display.

to dirt on the road, in which case it may need to return control to the driver. The HMI generates acoustic, tactile and visual alarms to bring the driver back into the loop; it also performs various checks to confirm that the driver has taken over as intended, for example, by sensing that the driver has gripped the steering wheel. If the driver does not react, the automation system brings the vehicle to a safe stop. In the reverse situation, when the driver is controlling the vehicle and the automation system senses sudden danger, the system may issue a warning to the driver and take over control to avoid an accident.

Management of this handover process is just one of the many functions performed by HMIs that are continually gaining functionality as vehicle automation systems continue to mature. As a result, the HMI development process has become increasingly challenging. In the past, when HMIs were developed using


▲ Automated lane change


▲ Warning presented on HMI

manual coding methods, developers typically did not receive feedback until the code was compiled and run on the expensive and complicated target hardware environment. Making changes to the HMI was difficult because the engineer making the changes had no way to validate them until the code could be run on the target. Many different scenarios had to be evaluated in the target environment for each iteration of the HMI, which was a long process. A considerable manual coding and testing effort was required to make changes to the HMI, such as moving an element from one display to another.

## TRANSITION TO MODEL-BASED DEVELOPMENT

The time required to develop and validate the HMI has been substantially reduced by moving the development process to SCADE Suite and SCADE Display. Functional requirements and test cases are linked to the SCADE model using the SCADE Requirements Management Gateway. DLR engineers now use a model-based design approach built on the creation of an executable model in a block-diagram design environment. Engineers define the functionality of the HMI using blocks that represent algorithms or subsystems. They have created a library of blocks in the SCADE environment that perform and display common vehicle automation HMI functions, so the development process largely consists of selecting and adapting existing blocks and connecting their outputs and inputs.

Engineers simulate the behavior of the model and receive immediate feedback on its performance. Test cases are run in the virtual PC environment rather than in the more-complicated and expensive target environment. For example, for each new iteration of the code, engineers must check hundreds of different scenarios to ensure that certain information is presented on the screen at critical points, such as handoff from

the automation system to the driver. In the past, this involved a lengthy manual process. Now, the engineer developing the model can run an automated routine that quickly evaluates each scenario.
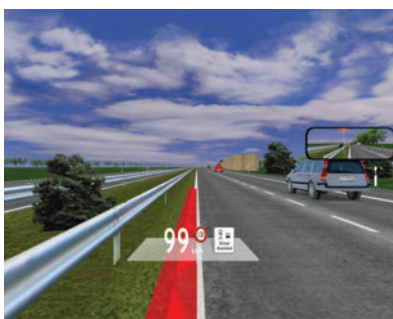
## AUTOMATIC CODE GENERATION

After the model has been validated, the SCADE KCG code generator produces code for the target environment. The SCADE Suite KCG C code generator provides complete traceability from model to generated code by establishing an unambiguous one-to-one relationship between the model and the code. The code is first run in different DLR driving simulators, including, finally, DLR's dynamic driving simulator, which combines a high-fidelity immersive visual system with an integrated cockpit and a hydraulic motion system to form a realistic driving environment to test prototype automation systems. Here engineers can evaluate HMI performance in driving scenarios that are very close to reality, such as bringing the driver back into the control loop when the system reaches its limits because of a construction detour. Once the HMI's operation is verified on the driving simulator, code is then generated for the test vehicle that can be controlled by a virtual copilot for system evaluation.

SCADE Suite and SCADE Display make it easy to modify the HMI to evaluate different alternative designs and to produce different variants of the HMI for different vehicles. DLR engineers can re-arrange the way that elements are positioned on the different displays of a vehicle simply by re-arranging blocks in the model. In the past, this would have required major amounts of manual coding. ANSYS SCADE Suite and SCADE Display have substantially improved the process of developing HMIs by continually testing and validating the HMI, first in the model phase, then in the vehicle simulator and finally in the target environment on the test vehicle, so that problems can be identified and corrected at the earliest possible stage. ∧