

dSPACE

MAGAZINE

1/2008

Daimler Introduces AUTOSAR

IABG/Continental:
Test System for Brake-by-Wire

Simulating Heavy Machinery
at Caterpillar



We CAN

So CAN you.

dSPACE in practice:

Read how easy it is to achieve perfect results with the RTI CAN Multi-Message Blockset. Our current issue of dSPACE Magazine takes a detailed look at the capabilities you have when using dSPACE systems in CAN communication networks, to be continued in the next issue.

You will see a typical tool-independent workflow, from CAN configuration to automated real-time testing, and learn about the benefits of tool coupling.

Handling complex CAN setups – particularly in hardware-in-the-loop applications – is easy with RTI CAN MultiMessage Blocksets. You can control, configure, and manipulate a very high number of CAN messages from one single Simulink block. Once the CAN system has been configured, it is available for extensive experiments, which you can run from a convenient graphical user interface. In the next step, you can perform reactive real-time tests with the greatest possible time precision. For example, you can capture ECU interactions and react to them within milliseconds.

Our report on working with the RTI CAN MultiMessage Blockset and the tools used downstream for rapid control prototyping and hardware-in-the-loop simulation starts on page 42. ■



*Dr. Herbert Hanselmann
President*

dSPACE is well-known for continuity. But there comes a time when you just have to try something new. That's what we're doing now: We've got a brand-new corporate design. New logo, new slogan, new magazine. Our logo and slogan were almost 20 years old, and dSPACE NEWS, our newsletter, had been running since 1992.

What's changed? For one thing, the stripes have gone from our logo. They originated because we wanted to add a touch of color to the slipcases for our first manuals, with long stripes running across. We then just cut them for the logo. Some people might have looked for a deeper meaning in them later, but there really wasn't one. We didn't intend to imply connections with either a well-known French airline or

Dutch cheese. And we felt it was time for a change, more "rectangular", which is what engineers like.

The situation is different with our new slogan, "Embedded Success". The old one, "Solutions for Control", was chosen intentionally at the time. We were control engineers working in control engineering. Closed-loop control systems were our specialty. Today, the field is much broader, and so is our tool chain. We believe that "Embedded Success" reflects everything that makes dSPACE stand out from the crowd. With our tool chain, success is embedded in the process.

This is why we often report on our customers' success stories. Until now, dSPACE NEWS was the platform for this. Now we've completely

redesigned it. The look and feel is more like a magazine, so that's what we're calling it. The articles will go into a lot more depth, whenever more information on customer projects is available.

Our 20th anniversary was a good time to change our corporate design in other respects as well. Web site, trade fair booths, flyers, job ads, PowerPoint presentations: They've all changed or will do so soon. To broadcast what we are, what we want to be, and how people see us, as we know from our brand image survey: solid, high-tech, reliable, innovative. These are the qualities our customers really find important. Compared with that, corporate design is mere decoration. But it's something that should look nice too, don't you think?

Dr. Herbert Hanselmann
President



AUTOSAR | PAGE

6



BRAKE-BY-WIRE | PAGE

14



CATERPILLAR | PAGE

24

IMPRINT

dSPACE MAGAZINE is published periodically by:

dSPACE GmbH · Technologiepark 25
33100 Paderborn · Germany
Tel.: +49 5251 16 38-0
Fax: +49 5251 6 65 29
dspace-magazine@dspace.com
www.dspace.com

Project Manager and Author: André Klein
Technical Authors: Bettina Henking-Stuwe,
Ralf Lieberwirth, Sonja Lillwitz,
Julia Reinbach, Dr. Gerhard Reiß

Co-Workers on This Issue:
Dr. Herbert Hanselmann
Editors and Translators:
Robert Bevington, Stefanie Bock,
Dr. Michelle Kloppenburg, Christine Smith

Design: Krall & Partner, Düsseldorf
Project Manager: Ines Schönenberg
AD/Layout: Tamara Stoiljkovic

Picture credits, pages 4, 48–53:
© Martin Langhorst
“Entrepreneur” by Ernst & Young

© Copyright 2008
All rights reserved. Written permission is
required for reproduction of all or parts of
this publication. The source must be stated
in any such reproduction.

This publication and the contents hereof are
subject to change without notice.
Brand names or product names are
trademarks or registered trademarks of their
respective companies and organizations.

Contents



2 We CAN.
So CAN you.

3 Editorial by
Dr. Herbert Hanselmann,
President

Customers

6 AUTOSAR in the development
process – procedure for
introducing model-based
AUTOSAR function develop-
ment into production projects
(Daimler)

14 Brake-by-wire gathers
momentum – HIL test system
for developing a 12-V
brake-by-wire system
(IABG/Continental)

20 Optimal air-fuel ratio – neural
networks for air-fuel ratio
control performed
by MicroAutoBox
(University of Salerno)

24 Indestructible from the
start – HIL and SIL tests for
heavy-duty equipment with
ASM (Caterpillar)

28 Major winnings at the DARPA
Urban Challenge
(TU Braunschweig,
University of Karlsruhe,
University of Parma)

34 Preparing students for
the real world (University
of Buenos Aires)

Products

36 Traffic jam on virtual
roads – ASM Traffic: new
model for simulating
road traffic

42 All you CAN test – from CAN
set-up to real-time test

46 Safe modeling – MISRA:
modeling guidelines for
TargetLink

Business

48 Anniversary of an idea –
twenty years of continuous
success

54 Tomorrow's engineers –
hybrid bicycle: dSPACE
supports young researchers

58 Compact News

59 Info

AUTOSAR

in the Development

Procedure for Introducing Model-Based AUTOSAR Function Development





Process

into Production Projects



The AUTOSAR standard defines a software architecture, software interfaces and interchange formats for creating function libraries that are independent of ECU suppliers. Daimler AG is introducing AUTOSAR in the fields of comfort and vehicle interior functions as part of its activities for standardizing function library software modules. This process includes using the autocoder TargetLink for modeling AUTOSAR software components and generating code from them.

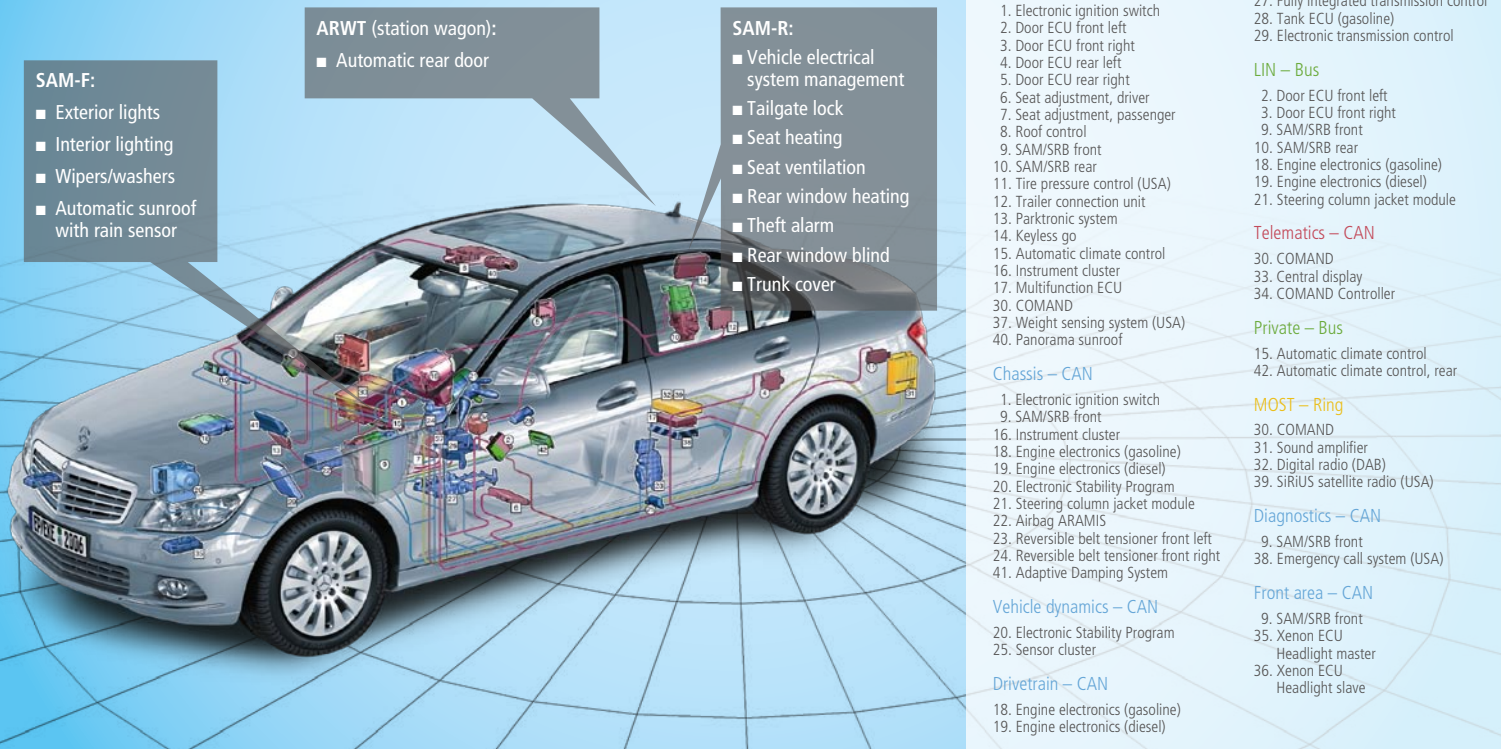


Fig. 1: Comfort functions for the C class (204 series) designed via modeling and automatically generated production code (SAM: central body computers ARWT: automatic hatch door).

Model-based design and automatic production code generation are used for passenger car development at Daimler AG. This article describes the step-by-step method of converting a model-based development process to the AUTOSAR architecture and reports on experience gathered so far.

Model-Based Development of Vehicle Interior Functions

For some years now, Daimler AG has used modeling to develop vehicle functions, especially for comfort and interior functions. Model-based development for the vehicle interior focuses on the main functions of the central body computers, each of which has integrated multiple functions. Model-based development was introduced on a large scale for the current C class (204 series) (fig. 1). The model-based approach is systematically applied to the series

currently in production development, and the number of functions to be integrated is constantly increasing. Some of the functions we develop comprise the entire ECU application software. Others are what are called added value functions, i.e., extra functions added to existing ECUs to differentiate them from the competition. In each development cycle, the supplier is given the function model that we developed, plus a test specification, and is responsible for producing software for the model and implementing it on the ECU (fig. 2). Essentially, model-based development gives OEMs the following advantages:

- Early maturity validation by simulation
- Functions represented and validated by rapid prototyping in the vehicle, even before the ECUs are available

- Direct implementation by autocoding
- Supplier-independent function development and further development
- Reusable functions
- Protection of intellectual property

Integration into Software Architecture

At present, when functions are created via modeling, suppliers still have a high manual workload when integrating them into the ECU. The amount of work involved greatly depends on the software architecture used by each supplier, even when the OEM has specified the communication part of the basic software. In some cases, the software architecture has to be adapted or specially extended. There is no completely standardized software architecture, so sometimes extensive coordination meetings have to be

held with specific suppliers. The need for coordination goes beyond just the software architecture, however. The OEM and the ECU supplier also have to jointly define the descriptions of metadata for integrating functions, such as the interface lists for the functions and the mapping of application signals to bus signals. Thus, the prerequisites for broader and process-safe use of model-based development are a uniform, supplier-independent software architecture and a standardized description of the metadata.

The AUTOSAR Software Architecture

The AUTOSAR standard defines a software architecture for ECUs, an integration method, and the interchange formats that these require. In other words, AUTOSAR largely addresses the requirements for the

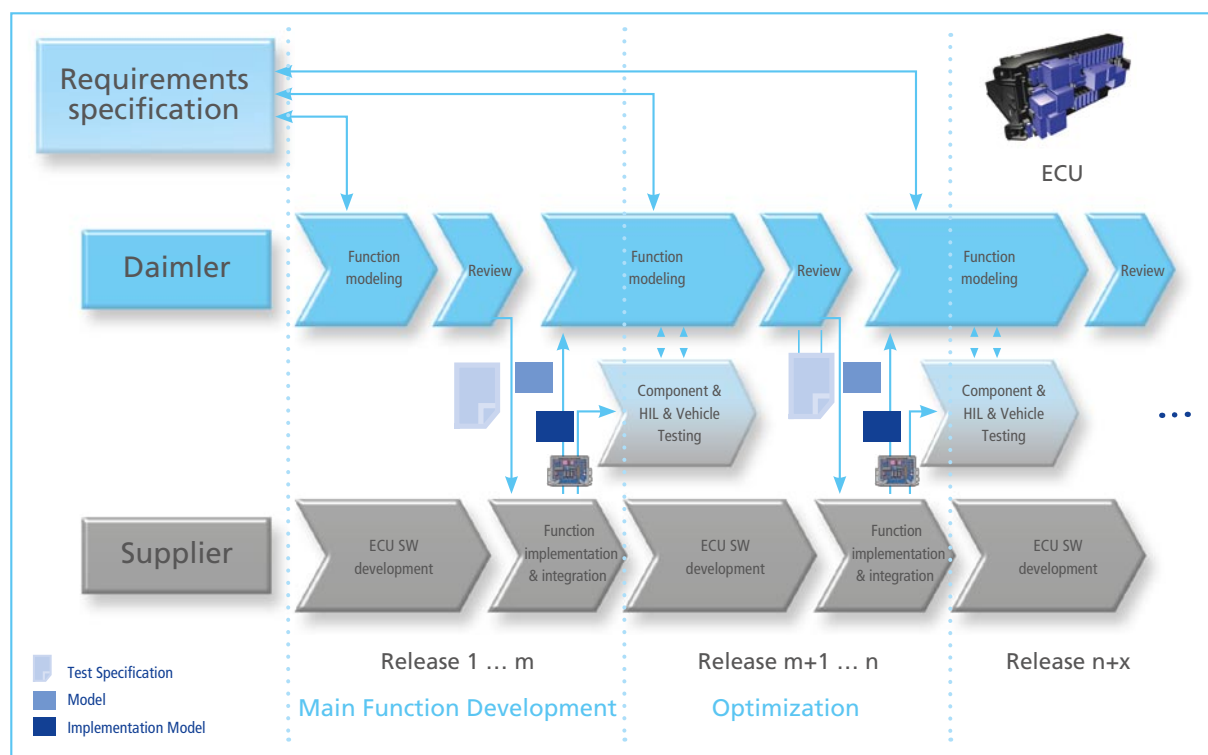
process-safe integration of model-based functions described above. AUTOSAR divides the application software of an ECU into several software components (SWCs), which communicate with one another via middleware (the run-time environment, or RTE). SWCs encapsulate the software and give it type definitions, allowing data exchange only via well-defined interfaces. Two mapping steps are needed for integration on an ECU: First the SWC instances are mapped to ECUs, and then the data elements are mapped to network signals for communication across ECUs.

Strategy for Introducing AUTOSAR

For the next generation of vehicle interior ECUs, Daimler AG is taking the first step towards an AUTOSAR architecture. The introduction

process starts at the top of the layer model for AUTOSAR software architectures and works its way down, i.e., it begins with the application software components (SWCs) and the RTE (fig. 3). This will involve systematically dividing the software architecture into application parts and basic software parts, which will communicate via a defined interface. The basis for defining this interface is the AUTOSAR standard, though this is partially restricted here and has also been more precisely formulated at some points. In this first step, the standard software will still be based on the established Daimler Standard Core, to which selected AUTOSAR software services (such as memory management for NVRAM data) will be added. In this early phase, the ECUs developed in this way will still be network-compatible with ECUs

Fig. 2: Typical development cycle with multiple iterations.





developed by the classic method. This means that AUTOSAR technology can be introduced step by step.

Modeling at Behavior and Architecture Level

With AUTOSAR, modeling is performed at two levels: Beside the behavior level, where the behavior of the functions is modeled, there is an architecture level where the interfaces of the SWCs and their connections have to be formally described.

In a top-down strategy, when new vehicle functions are developed it is useful to first subdivide their functionality into several SWCs and then to define their interfaces at architecture level (fig. 4). Then the behavior of the resulting SWCs is modeled.

For previously existing function models, a bottom-up procedure can be used to generate the SWC descriptions from the model interfaces. The resulting SWCs are then connected with one another at architecture level.

Stepwise introduction means that it is not possible to produce a complete top-down design of a whole vehicle. It also means that at the level of single ECUs, not all functions are initially available as models. A meet-in-the-middle strategy has therefore proved useful, with the following steps being performed for each ECU:

- The SWC descriptions are derived from existing behavior models.
- For new vehicle functions, the interfaces are first defined at architecture level via SWCs.
- Sensor/actuator SWCs are generated automatically according to fixed rules.

The resulting SWCs are then collected together in a composition at architecture level and networked with one another. The remaining unconnected ports are led through to the outside, which turns them into ports in the composition. The ports now represent the communication interface of the ECU. Data elements referenced via the ports can be mapped to the signals specified for

the ECU by the communication matrix. This makes it possible to create the SWC structure of an ECU at a reasonable cost. The next section explains these activities in greater detail.

Defining the AUTOSAR Interfaces

The basis for developing software components is a shared database or common object pool (COP) of interface and data type definitions that the SWC developers can use when defining SWC ports. The pool was derived from the signal definitions in the communication matrix and must be constantly reconciled with this (fig. 8).

The derivation rules state that one interface with a data element must be generated for each communication matrix signal. This facilitates the transition from the current communication matrix signal world to the AUTOSAR world, since the same names are used and the corresponding interfaces are therefore easy to find. AUTOSAR's structural options for collecting several data elements together in one interface are used

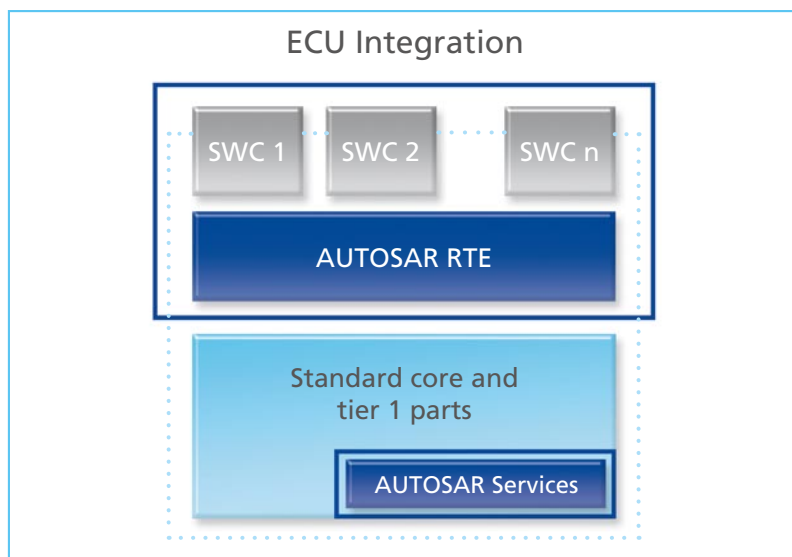


Fig. 3: In stepwise AUTOSAR introduction, first the application software (SWCs) and the middleware (RTE) will be developed in compliance with AUTOSAR. The basic software will still consist of the Daimler Standard Core, extended by selected AUTOSAR software services.

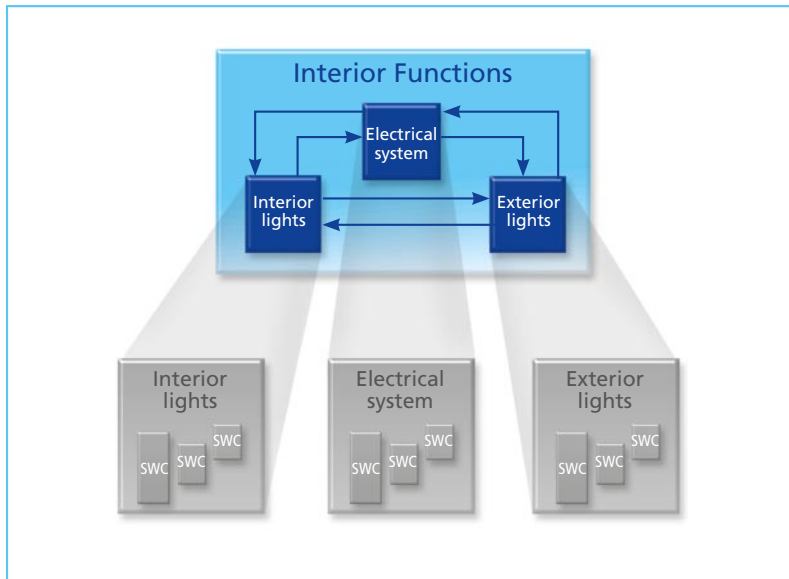


Fig. 4: In the top-down approach, first the functionality is divided into several SWCs, and then their interfaces are defined at architecture level.

for new interfaces. The derived interfaces will not profit from this for the time being.

To ensure that the SWCs are compatible with one another, modeling rules are used to ensure that the only interfaces used are the ones in the pool. If necessary, a coordinator must add further interfaces to the pool.

Creating New Vehicle Functions with the Top-Down Strategy

The modeling of new vehicle functions begins at architecture level, with the descriptions of the SWC interfaces in the COP database described above. This involves creating ports with types defined by interfaces. Runnables and their subsets of the SWC interface are also defined. RTE events are used to define the runnables' cycle times and, if necessary, event triggers such as the receipt of a new signal. Finally, an AUTOSAR SWC description is generated from this.

This AUTOSAR description is the basis for behavior modeling. The SWC descriptions are imported into the Data Dictionary and a modeling frame is generated as the starting point for modeling a behavior function. The modeling frame contains the relevant AUTOSAR interfaces and runnables, using the TargetLink AUTOSAR blocks (fig. 5). The frame is then filled with standard blocks for modeling the actual algorithms. Any changes to the interface are always performed in the database.

Migrating Existing Models in the Bottom-up Strategy

Existing behavior models were migrated, using the interface definition for deriving and typing the ports. We were able to automate virtually 90% of model migration, since we already had modeling conventions for using the names of the associated communication matrix signals for input and output variables.

Glossary

Composition – In a composition, several SWCs are grouped and connected to form one superordinate component.

DBC file – Database container file. Communication matrix description file for CAN.

Communication matrix – Describes the signals/data exchanged between individual ECUs.

NVRAM – Nonvolatile random access memory, used for storing diagnostic information, etc.

Runnable – An executable element in an AUTOSAR SWC, comparable to a function.

System architecture tool – Used for planning and integrating complex ECU architectures. It can import, export, and modify AUTOSAR files.

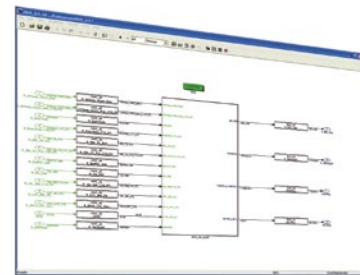


Fig. 5: Frame model created with TargetLink.

First the SWC was created in the Data Dictionary, then the appropriate AUTOSAR blocks were inserted in the model (fig. 6). Finally, the XML file (SWC description file) created during code generation was loaded to the SWC database. The interface can now be maintained in the database.

The modeled functionality (behavior model) was migrated in its entirety and with no modifications; only the model interface had to be adapted to the standard (fig. 7).

AUTOSAR Tool Chain

An existing tool chain was extended for developing AUTOSAR-compliant vehicle functions (fig. 8). The common object pool for storing AUTOSAR interfaces and SWC descriptions is a major component of the development environment. It was implemented as an extension to an existing database application used for creating communication matrices. The pool not only contains AUTOSAR modeling elements; in it, parameters and NVRAM variables can also be described and given values. Because of its similarity to the communication matrix, AUTOSAR interfaces can be generated elegantly from the data in the communication matrix and sub-

sequently updated. The tried and tested release concept used for communication matrix development can also continue to be used.

The SWC descriptions can be imported and exported via the AUTOSAR interchange formats. This supports the connection to the Data Dictionary and to system architecture tools. The model-based development of AUTOSAR-compliant function models is supported by model-in-the-loop (MIL) and software-in-the-loop (SIL) tests, both integrated in the development environment. Finally, AUTOSAR-compliant production code together with a consistent AUTOSAR SWC description file is generated.

The system architecture tools are used to network the SWCs. This is usually done in graphical editors, which provide better visualization than purely form-based database applications. The system architecture tool is now being used to map the non-local data elements to network signals. The resulting AUTOSAR system template is used for ECU integration at the supplier.

Conclusions and Outlook

AUTOSAR is answering a long-standing need to standardize the description formats and interfaces

for model-based function development. The type-safe AUTOSAR descriptions make it possible to ensure consistency between separately developed function models at the very early stage in the development process when we as the OEM hand over the function models to the ECU supplier. The expectation is that this will make the function integration by the supplier much more efficient. Coordination meetings held between the OEM and the supplier to discuss software architecture are considerably more productive because both sides can use terms that are standardized by AUTOSAR. The current necessity of using two development tools (for modeling behavior and for describing interfaces) in developing AUTOSAR-compliant function networks poses new challenges, as each system has to be divided into manageable and logically useful software components. In the future, the transitions between different modeling tools, often from different vendors, will have to be made more efficient to ensure a 'round trip'. Moreover, the current division of AUTOSAR development environments, into tools for architecture modeling and system integration and tools for behavior modeling with their own

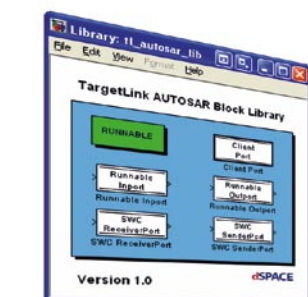


Fig. 6: The functions are modeled with TargetLink's AUTOSAR blocks.

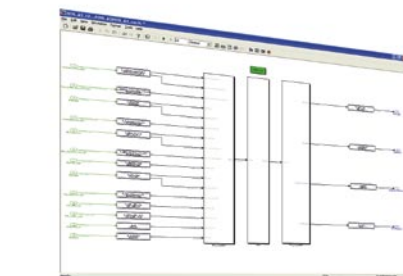


Fig. 7: Modeled runnable with appropriate interface blocks after AUTOSAR migration.

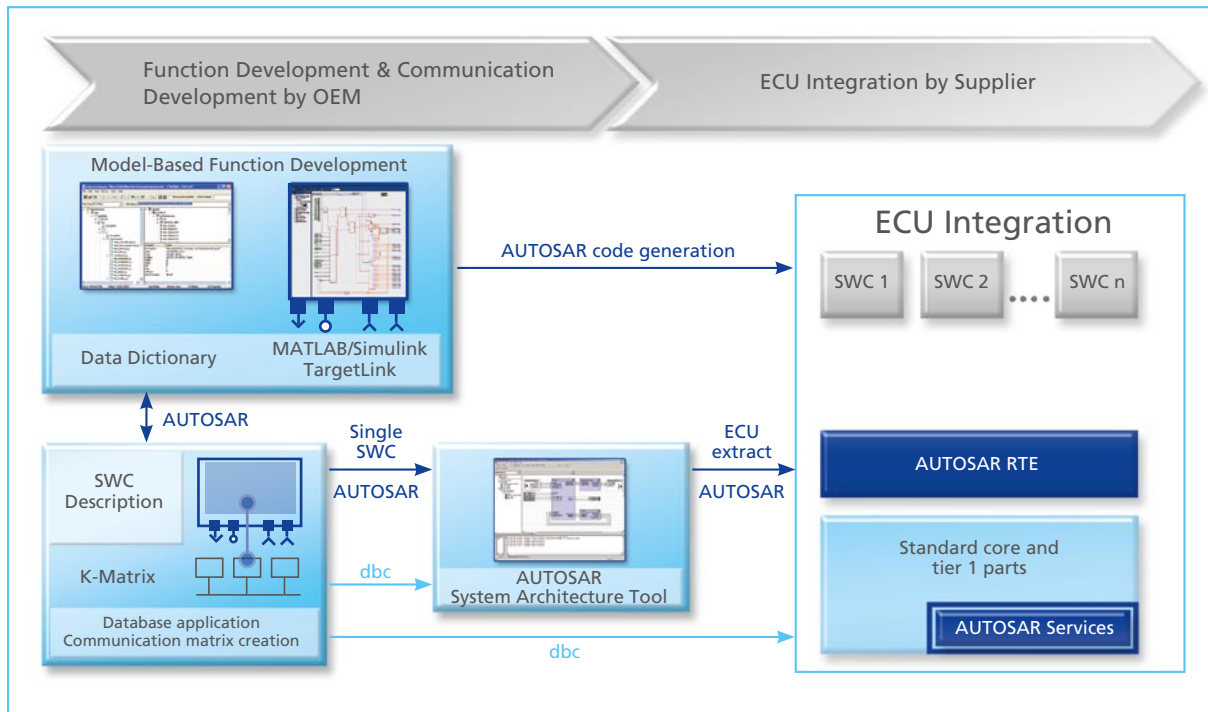


Fig. 8: AUTOSAR tool environment for generating AUTOSAR software components (SWCs). Data exchange between tools is based on AUTOSAR XML and DBC files.

autocoders, is to a large part due to the tool domains. This limits the potential for tool-independent system modeling and resource optimization. Architecture and design decisions therefore need to be thought through carefully at the outset. Because the AUTOSAR description is so extensive, and because individual tools do not yet provide complete equivalents, developers have to begin by deciding on a subset of the standard via suitable application profiles.

As we see it, AUTOSAR is capable of evolutionary introduction into the development process as it is known today. With a stepwise approach to introducing AUTOSAR, we can continue using existing function models as well as proven processes and tool chains, and optimize them step by step.

Due to its success, using function modelling to develop comfort functions and interior functions has become a standard method at Daimler AG. Much of the experience gathered in these projects can be transferred to AUTOSAR-compliant development and will make it possible to replace manual steps by automated and standardized process steps in the future.

Since the definition of the AUTOSAR standard is not yet complete, close cooperation with the standardization groups and tool manufacturers continues to be necessary to ensure that investments made in converting to AUTOSAR are future-proof. ■

*Christian Dziobek, Dr. Florian Wohlgenuth
Mercedes-Benz Cars Development
Dr. Thomas Ringler,
Group Research & Advanced Engineering
Daimler AG, Germany*

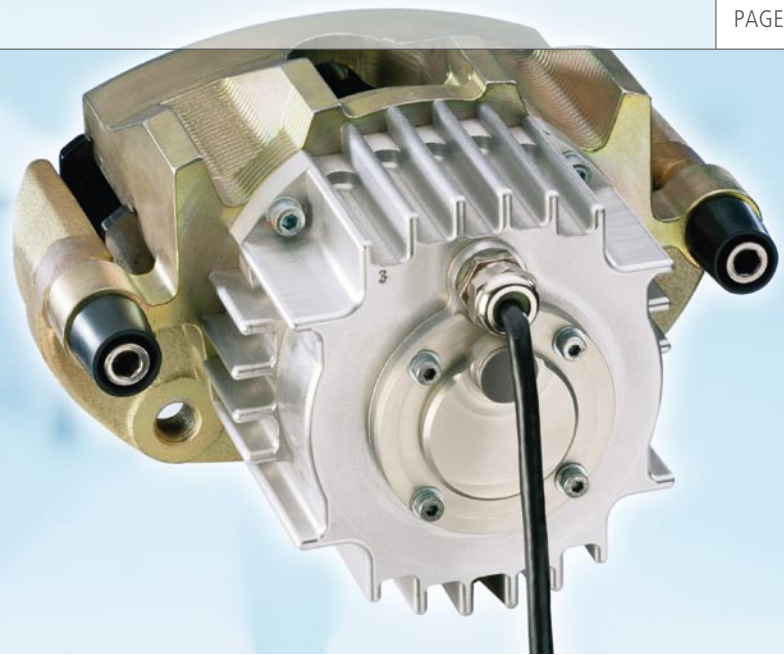
Summary

- Daimler AG is introducing AUTOSAR in a production project for vehicle interior ECUs.
- Existing tools and processes were extended for AUTOSAR.
- Existing models can be reused with the AUTOSAR-compliant code generator.

Brake-by-Wire Gathers Momentum

HIL Test System for Developing
a 12-V Brake-by-Wire System





The future of the brake is electric (brake-by-wire system). An electric motor (in the foreground) replaces the hydraulic cylinders which press against the brake disc.



Continental is currently developing brake-by-wire systems that can use a new brake technology based on a 12-V electrical system. For Continental to perform function optimization and concept validation for this innovative brake-by-wire system, IABG developed a test system with an interface to a real-time vehicle model based on dSPACE technology.



This hardware-in-the-loop (HIL) test system has proven to be an efficient development and testing tool within the development process. Its main tasks are putting the system network into operation, function verification, fail-safe trials and investigating the energy management functions of the braking system. HIL test procedures have numerous advantages over test drives in laboratory vehicles – reproducibility, efficiency, cost efficiency, etc., – and all these are fully utilized.

Development Approach to Complete Brake-by-Wire Systems

The completely new braking technology in the electronic braking system utilizes a principle that is as simple as it is effective. Based on conventional friction braking, it involves no hydraulics whatsoever: the actuators are operated entirely by electronically controlled electric drives. Each wheel has its own control unit. When the car driver presses the braking pedal, an electronic signal is sent to the wheel brake. Sophisticated sensors and software optimally adjust the braking power on the wheel to a wide variety of road conditions. During braking, the friction arising between the brake pad and the brake disk generates torque on the pad. This is fed back into clamping force generation. Thus, much of the brake caliper's clamping energy is obtained from the vehicle's kinetic energy, so it does not have to be mainly fed in externally as electrical power.

The main technical point is that this braking principle consumes less energy than other systems. For the first time ever, very high clamping energies and excellent control dynamics can be achieved with the 12-V vehicle electrical systems currently in widespread use. The system also has an integrated parking brake function.

HIL Test System: Integral Part of the Development Process for the Electrical Braking System

The HIL test system was designed to verify the networked system functions and system communication and to perform further investigations during the development process. It acts as a system test bench for putting the system network into operation and for testing system functionality. Other purposes include investigating and optimizing basic functions and energy management in the back-up level of the braking system, and testing how the braking

system interacts with the simulated vehicle electrical system.

To create a complete HIL test system, it is supplemented by an application-specific, adapted, extended simulation environment based on the dSPACE Automotive Simulation Models (ASM). This simulates the vehicle dynamics' behavior in a virtual vehicle in real time. The physical vehicle variables and mean value variables calculated for the braking system are used to control the dynamic restbus simulation and the test system actuators.

The applications focus on putting the system into operation, testing communication between system components, and studying error behavior when individual components fail. The integrated vehicle dynamics simulation makes it possible to systematically test different driving maneuvers including specific vehicle dynamics situations. The results are used to validate the system network and its functional properties, taking into account the effects of failures on driving behavior.

Set-up and System Architecture

Since the generation of clamping forces depends on (load) torques being transmitted via the brake disks, the task at actuator level is to implement a suitable load for the test bench under cost-efficiency constraints, i.e., to avoid using actuators operated via a real gyrating mass like

Set-up of the HIL test bench.



with flywheel mass test benches. This is done by means of a device that emulates loads.

On the test bench, the motors of the actuators form the interface to the simulated braking actuators. Highly dynamic load equipment consisting of servo drives is applied to the motors to ensure dynamic simulation of the real loads that would occur in the vehicle. The electrical and mechanical components are designed so that prototypes can be installed and assembled flexibly in various test configurations.

“The seamlessly integrated development environment from dSPACE lets us carry out projects efficiently. The help given by dSPACE Support was exemplary.”

Franz Hangl, IABG

The complete communication network of the electrical braking system, with the pedal unit and the additional function of the electronic parking brake, is represented in hardware form in the test system. In addition to CAN communication with the braking system's central ECU, a flexible FlexRay communication architecture was also set up. This enables all four wheel units of the brake-by-wire system to be simulated as required. Switching between the wheel units is performed by software, right through to the model. The mechanical, electrical, and communication variables of the system network are captured at various measurement points and archived in the automation system for subsequent evaluation.

Load representation at the actuator.

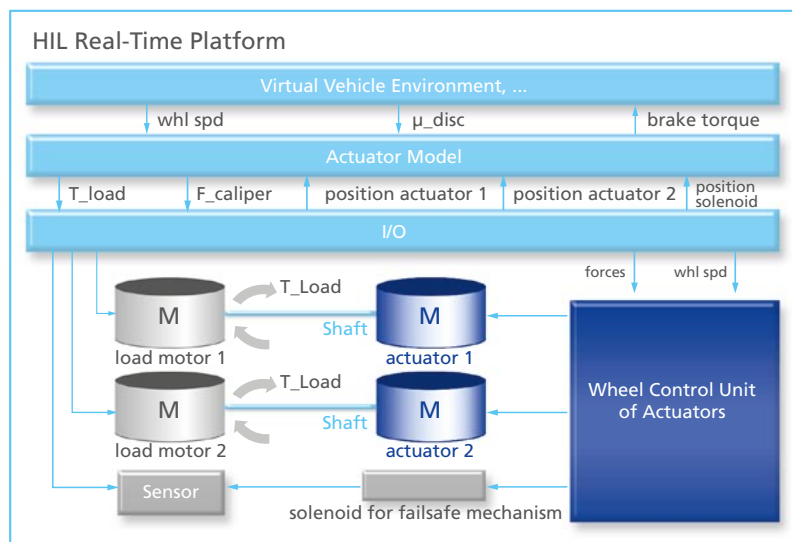
The core of the test bench automation system (AT) is a DS1005 PPC Board networked with several dSPACE I/O boards. The test bench therefore has a complex FlexRay network with a total of two independent networks in addition to CAN communication with the ECUs. To connect the test system to the FlexRay bus, the FlexRay Configuration Tool from dSPACE is used with a network description in a FIBEX (field bus exchange format) file. The vehicle model is calculated on an additional DS1005 that is

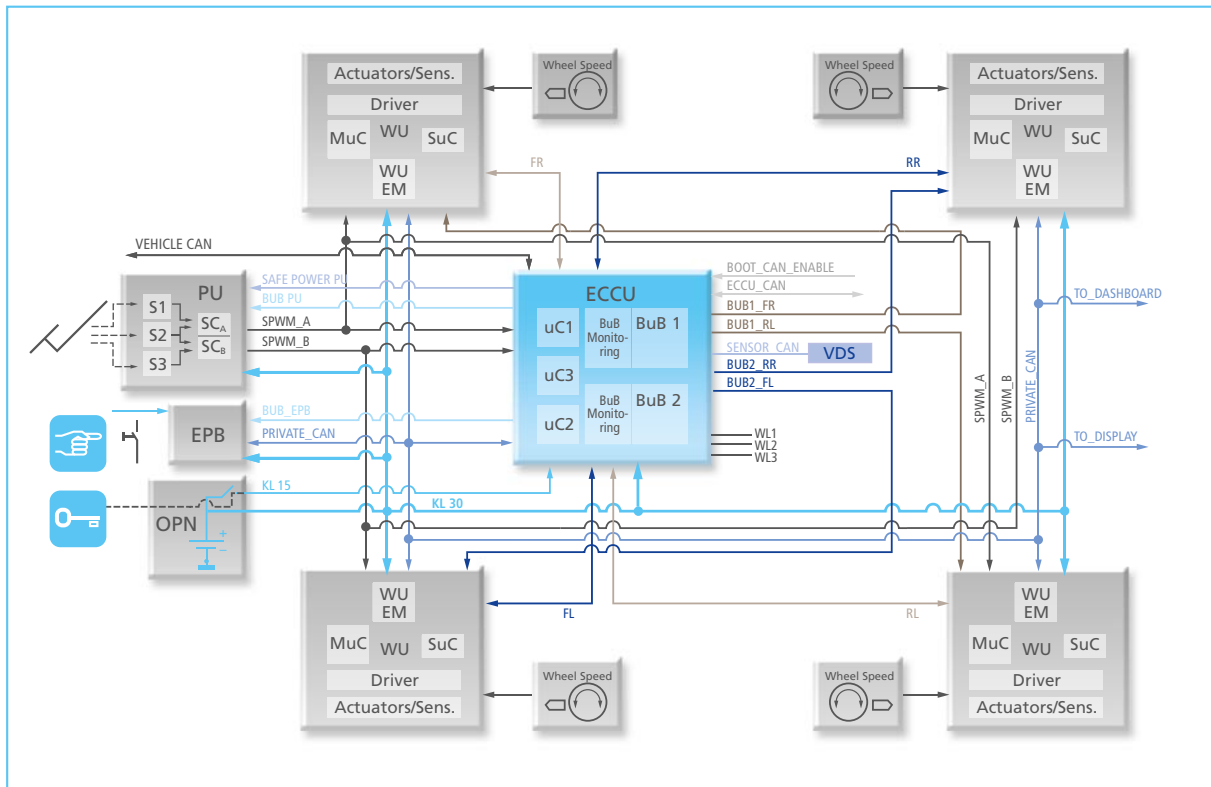
side, the HIL platform provides integrated vehicle dynamics behavior based on the dSPACE Automotive Simulation Models (ASM). The major components of this simulation environment are models for the vehicle, driver, and environment, the associated model control, and model parameterization functions. One point particularly worth noting is that integrated simulation is performed for any system components that are not installed as real parts. Because the model structure is open down to the Simulink block level, it did not take long to make optimum modifications to the model components of the ASM Vehicle Dynamics Simulation Package.

The automation software builds on the MATLAB®/Simulink®/Stateflow® development environment and the ControlDesk experiment software from dSPACE. Some of the functions that have to be executed on the AT real-time system are:

- Event-discrete sequence control for the test system control and the implemented single tests
- Communication with the ECUs via FlexRay and CAN protocols
- Communication with the HIL real-time board via Gigalink interface

connected to the test bench automation system in real time via dSPACE Gigalink (a high-speed optical connection). The processes on the automation platform and the model platform communicate at intervals of 1 ms. On the software





System architecture and communication structure of the electrical brake based on a 12-V vehicle electrical system.

In addition to basic automation, the user can also use functionalities such as:

- Implementation of the graphical user interface to handle and configure the test system
- Event-controlled layout and experiment management
- Test automation and test management with IABG software modules in conjunction with AutomationDesk to configure and generate test sequences
- Automatic evaluation and generation of test reports
- Access to ECU operating states, variables and parameters via a diagnostic interface
- Evaluation software to display measurement data graphically

Test Automation and Failure Simulation

The task of test automation is to define single tests individually, execute them sequentially in a coordinated manner, and abort test execution in the event of a failure. A test report and test run documentation are automatically generated for each test. The user can configure the form and contents of the test report.

Test automation based on AutomationDesk was developed on the HIL test system. AutomationDesk lets the user freely define test sequences that are executed by the test bench automation system. This gives the user the greatest possible flexibility when specifying and implementing tests. The test database that was set

up now contains approx. 800 single tests of varying complexity and is constantly being added to.

The automated environment provides access to the requirements, test specifications, and test results via a direct connection to the test configuration and management tool that is used in the development and test process.

One of the applications for the test automation system is inserting fault scenarios to analyze system behavior. The following fault types can be implemented on the test bench:

- Cable harness faults
- Signal faults
- Communication faults in CAN and FlexRay

“The test system that IABG set up for us, based on dSPACE components, enables us to reliably develop innovative brake-by-wire products up to production level.”

Stephan Lehl, Continental AG

Cable harness faults are implemented by a failure simulation unit with currents of up to 50 A. The automation system controls the unit in real time via a CAN interface. Signal faults (sensitivity changes, offset drift) and communication faults (message interruptions, checksums) are switched to physical signals by the real-time system.

Extending the Test Features

The test system described here is completely integrated into Continental's development and testing process. Individual test sequences can be programmed graphically, which supports extensive testing of the system network and of single components.

The connection to the test configuration and management tool ensures a seamlessly integrated process from the requirements sheet for the overall system to single tests on signal software functions. Thanks to constant further development of the implemented test cases, the system network can largely be validated at an early stage on the HIL test system. ■

Franz Hangl, Automotive Test Systems, IABG Ottobrunn, Germany
Stephan Lehl, Electronic Brake Systems, Chassis & Safety, Continental AG, Regensburg, Germany

Summary

- IABG relies on dSPACE technology for complex test system solutions
- Function optimization and verification for networked vehicle control systems focusing on brake-by-wire
- Extensive test automation solution implemented with ControlDesk and AutomationDesk, including a connection to a test management tool



ControlDesk layout for controlling the test system.

Engine test bench equipped with lambda sensors in the exhaust pipe.



Neural networks for

Optimal Air-fuel

Optimal air-fuel ratio is a key challenge and keeps coming up as an open issue for the engine control community. Since the 1980s, the transition from carburetors to electronically controlled injection systems has been motivating researchers to concentrate on this topic. Proper control of the air-fuel ratio is greatly beneficial to the performance of three-way catalysts in both steady and transient operations. This control task therefore plays a fundamental role in limiting exhaust emissions in SI, GDI and lean-burn engines.

air-fuel ratio control performed by MicroAutoBox

Ratio

Requirements For Air-fuel Ratio Control

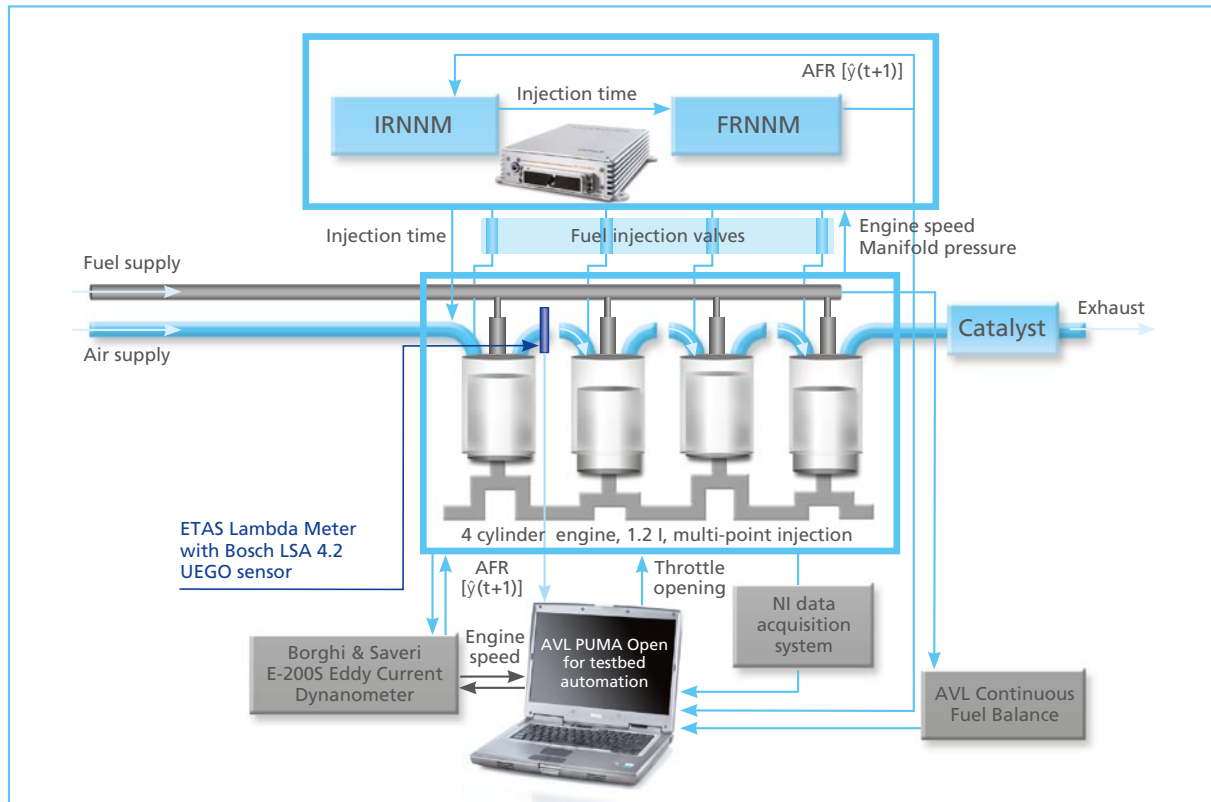
Despite the considerable efforts made in limiting exhaust emissions, the increasingly stringent environmental regulations imposed throughout the world still make achieving a satisfactory air-fuel ratio an ambitious goal. Furthermore, engine control system designers have to deal with the onboard diagnostics requirements, introduced in 1996 in the US and later in Europe. This represents a more critical goal in the field of automotive control, since it requires the continuous monitoring of all powertrain components to prevent critical

faults in exhaust systems. Air-fuel ratio (AFR) control presently relies on a mean value engine model representation. But these mean value models have some significant limitations, such as the high level of experiments needed for parameter identification and the intrinsic non-adaptive features. On the other hand, the AFR signal delay is a very critical issue to be mastered to improve the performance of closed-loop control strategies. Neural networks are a promising solution for these challenges. They have high mapping capabilities and guarantee a good generalization even with a reduced set of identification

data. Moreover, by implementing adaptive training procedures we can consider the influence of exogenous effects on the control performance.

Developed Control Strategy

The AFR control strategy is based on a recurrent neural network (RNN). The neural network is used as a controller and its output directly determines the control actions. A forward RNN model (FRNNM) of AFR dynamics was developed. This took into account the fact that the dynamic processes affecting the AFR response depend on both air and fuel dynamics. Therefore the output,



Experimental setup with the controller:

The control strategy is made up of two RNNs, describing inverse (IRNNM) and forward (FRNNM) dynamics.

control and external input variables are: AFR, injection time, engine speed, and manifold pressure. The output feedbacks are simulated by the network itself, so the FRNNM does not require any AFR measurement to perform the online estimation. This makes the controller a suitable solution for AFR virtual sensing when the lambda sensor does not guarantee an accurate measurement, which happens during cold start phases. It also allows the delay due to engine cycle, transport phenomena and sensor response to be removed.

Neural controller

The control actions are computed by an inverse RNN model (IRNNM) as a function of sensor measurements of engine states and external inputs. The output values predicted by the FRNNM are fed as feedbacks to the IRNNM which evaluates the control

actions as a function of the desired output at the next time step. The more accurate the FRNNM prediction, the less the difference between the FRNNM and plant outputs.

Experimental Setup

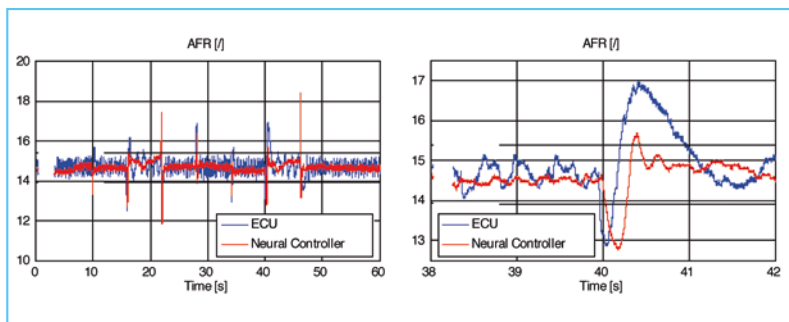
The developed control strategy was trained and tested vs. transient data sets measured on an engine test bench. The lambda sensor was placed right after the exhaust valve of the first cylinder to investigate the air-fuel mixing process in one cylinder only. This choice allows the dynamic effects induced by gas transport and mixing phenomena occurring in the exhaust pipes to be removed. Non-predictable effects generated by cylinder-to-cylinder unbalance due to uneven processes such as air breathing, thermal state and fuel injection can also be neglected. Therefore, the time shift between the injection tim-

ing and the lambda sensor measurement mostly accounts for the intake and exhaust valve phasing. As mentioned before, the time delay could be a significant problem for control applications.

For the real-time application the controller was modeled with MATLAB®/ Simulink® and then uploaded to a dSPACE MicroAutoBox. This compact prototyping system lets all engine tasks be controlled directly and customized variants of the controller be performed immediately. The direct controller is intended to provide the actual injection time by processing actual and earlier measurements of the engine speed and manifold pressure, and the earlier prediction of AFR performed by the FRNNM. Furthermore, the target AFR was imposed and was set to the stoichiometric value (i.e. 14,67) for the current application. Due to the lambda

“We replaced the ECU with a dSPACE Micro-AutoBox. It allowed us to control all engine tasks and easily customize the control laws.”

Cesare Pianese, University of Salerno



The developed control strategy is very accurate because it follows the target AFR faster and more precisely than the native ECU does.

sensor location, the controller was tested on the first cylinder only, while a classical map-based injection strategy was used for the three remaining cylinders.

Results for the Direct Controller

The trained IRNNM simulates the inverse AFR dynamics as accurately as the FRNNM does for the forward dynamics. Online tests on the developed RNNs were performed by integrating the FRNNM and IRNNM in the framework of a MicroAutoBox, resulting in the neural controller structure.

Conclusion

The virtual sensor (the FRNNM) adequately predicted the AFR dynamics with an estimation error vs. the measured trajectory lower than 2% for most of the test transients, even when wide AFR spikes were present. This proves that the RNN dynamic behavior is satisfactorily close to the real system dynamics.

The controller, which also uses the virtual sensor prediction, was implemented on the ECU and tested over

an experimental transient. The comparison with the AFR trajectory resulting from the action of the reference ECU shows that the controller performs well. In particular, the integration with the virtual sensor prediction induces a higher-order response that results in a faster AFR compensation and, particularly, in the removal of the overshoot observed by the ECU. In this context the MicroAutoBox was a great help, since its high computing power always ensured sufficient headroom for the real-time execution of the complex algorithms. The results demonstrate that neural controllers have a great potential for improving engine control strategies, especially since they significantly reduce the amount of experimentation and calibration needed by other current methods. ■

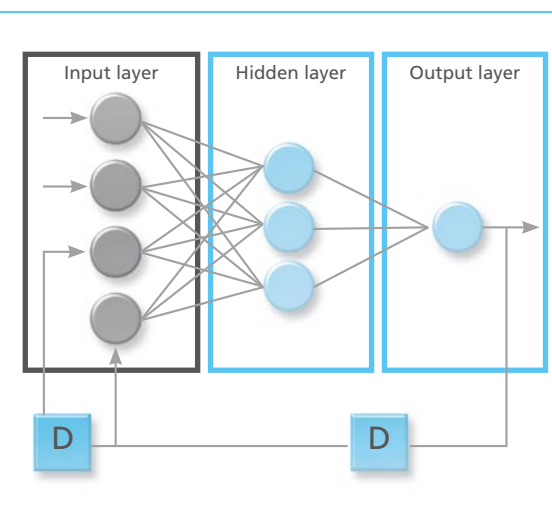
Ivan Arsie, Silvana Di Iorio, Giuseppe Noschese,
Cesare Pianese, Marco Sorrentino
pianese@unisa.it
Department of Mechanical Engineering
University of Salerno, Italy

Glossary

Air-Fuel Ratio (AFR) – Mass ratio of air to fuel present during combustion. It is an important measurement for anti-pollution and performance tuning. Lambda (λ) is the alternative expression for the AFR. For pure octane the optimal ratio is $\lambda = 1.00$ (stoichiometric mixture of air to fuel 14,67:1).

Lambda Sensor – Also known as oxygen sensor. It monitors the amount of oxygen in the exhaust, so the ECU can determine how rich or lean the fuel mixture is and make adjustments if necessary.

Neural Network – Made of individual units named neurons. Each neuron has a weight associated with each input. A function is then generated as output. Typically the neurons are connected together with an input layer, an output layer and one or more hidden layers. Recurrent neural networks (RNN) are derived from the static networks by introducing feedback connections among the neurons. A dynamic effect is introduced into the computational system by a local memory process. Advantages of RNNs are that they can be sensitive and adapted to past inputs.




Recurrent neural network (RNN).

Indestructible right from the start

HIL and SIL tests for heavy-duty equipments with ASM





Construction, mining, demolition and forestry in the modern age are dynamic, increasingly complex tasks, and the heavy-duty equipment involved has a lot more intelligence inside than one would guess from the outside. The release tests on the electronic systems for the brand-new machinery illustrate how development processes are conducted at Caterpillar. From function design with custom-built high-fidelity models to the validation of control software, Caterpillar has efficient processes in place.

Caterpillar (Cat, for short) is an international company that manufactures heavy-duty equipment for the worldwide markets of construction, mining, agriculture, and power generation. The products are equipped with electronic control units (ECUs) ranging from engine and transmission management to dedicated control devices for specific system features. At Caterpillar, software-in-the-loop (SIL) and hardware-in-the-loop (HIL) testing are the standard methods for productive development and to efficiently ensure the quality and reliability of these ECUs.

To test the engine ECU of its heavy-duty equipment, Caterpillar chose the diesel engine model and physical turbocharger model from dSPACE's Automotive Simulation Models (ASM). A customized drive train model from Caterpillar was used to meet the requirements of the equipment drive train. Due to the openness of the ASM, it was quite easy to integrate the custom drive train models with ASM.

High-Fidelity Models for Combustion Control

The development of new control functions begins with using high-fidelity engine models created by Caterpillar that simulate the combustion process in great detail – but not in real time of course. The result of many years of development, these “Enterprise” models are ideal for function design, where precise control of the fuel and air systems and of combustion is essential. The models use a one-dimensional approach. The combustion models also take cylinder pressure into account. During simulation with Enterprise, development focuses on optimized torque output, minimized fuel consumption, etc., but not on diagnostics and inter-ECU communication yet.

Early Controller Software Validation

In the next step, more ECU functions are integrated with the combustion control functions. SIL testing is a common procedure at Caterpillar to



Caterpillar's machinery produces unmistakable results.

test and validate the entire controller software under development at an early stage. In an ideal process, the same model is used for SIL and HIL testing, since the parameterization can be reused and signals can be compared directly. After Caterpillar chose the ASM Diesel Simulation Package from dSPACE to perform HIL tests, the goal was to run real-time simulations with the package during function design. The company's in-house simulation environment, Dynasty, was extended to execute not only Caterpillar's high-fidelity models but also the mean value models from dSPACE. Caterpillar and dSPACE worked together closely to integrate the simulation models. To transfer the parameters from Enterprise to the ASM, a solution based on a table with one-to-one variable referencing was implemented. Measurement data either from the real engine or from the simulation is required to complete the parameterization.

Co-simulation

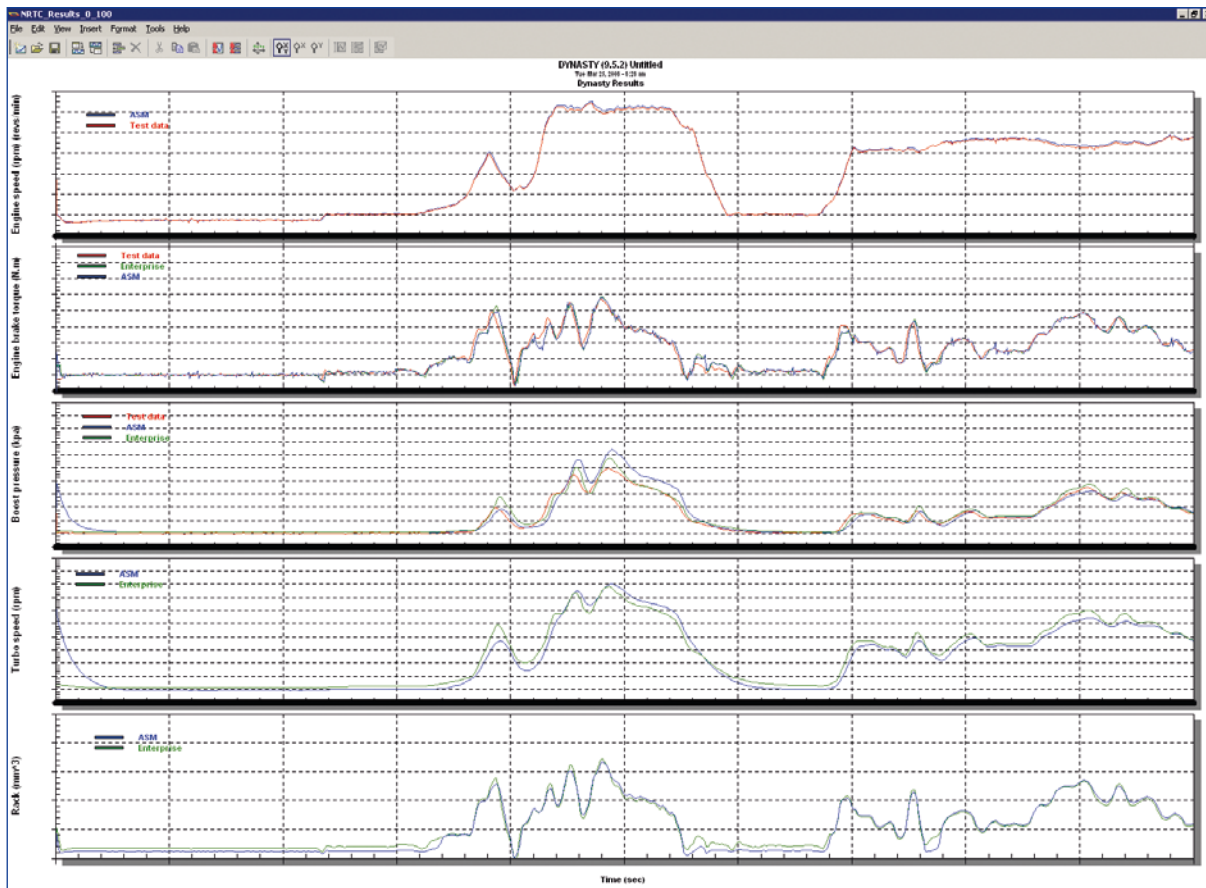
To evaluate the ASM diesel and turbocharger models and check that their quality and performance is sufficient, they were co-simulated with the Enterprise model in Dynasty and the results were compared. During co-simulation the ECU code was connected to the inputs and outputs of the ASM diesel engine model. The model was parameterized with ASMPara. Steady-state simulation results were checked against measurement data. Dynamic simulation results were compared with transient measurement data and with the Enterprise simulation results based on test cycles like the EPA Non-Road-Transient-Cycle. The simulation results of the ASM models matched the reference signals with excellent agreement in every case. As a result the ASM Diesel Engine Simulation Package is now used for controller validation during offline simulation and for ECU validation during HIL simulation.

ECU Testing at Caterpillar

Caterpillar already equipped its testing laboratory with numerous HIL test benches from dSPACE, ranging from Mid-Size to Full-Size systems. These are used to test ECUs for the engine, transmission, and other parts of the equipment. Using ASM has made parameterization much easier, since parameters are available early in the development process when SIL is performed. Another advantage is that the open model design lets you easily couple custom transmission and drive train models with the ASM diesel models. Thus the HIL benches can be configured to match any project.

The Benefits of ASMs

Practical experience has shown that performing real-time HIL and SIL tests with the same model makes development much more efficient. Moreover, we have used ASM diesel engine and turbocharger blocks for a wide range of engines



The results of the Enterprise/ASM comparison.

(4–20 cylinder, VTG or wastegate turbo, CGI, ...) and to streamline our development processes in many different projects. This is something that pays off quickly. We especially like the openness of the ASM. They are easy to understand and modify, and we can also easily integrate custom models.

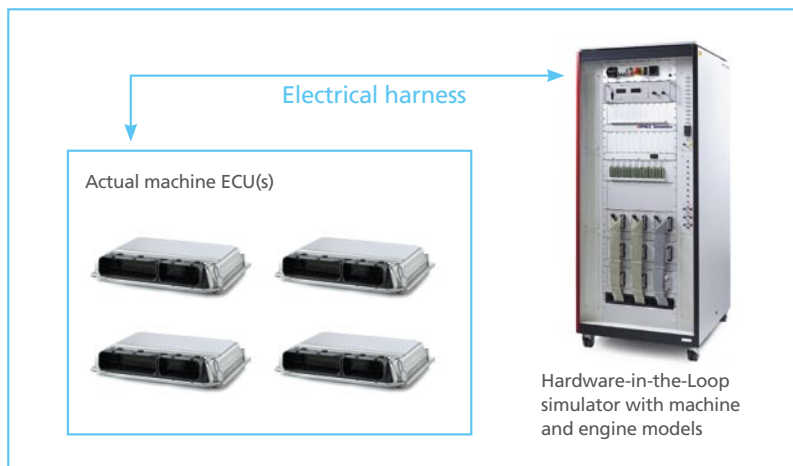
Summary and Outlook

The ASM are installed as add-ons to Caterpillar’s Enterprise model to support the development of new controller functions, where they are used for controller validation during function design and ECU testing. Caterpillar intends to use the ASM for most of its real-time engine

models on HIL systems. Due to the advantages that the ASM bring, Caterpillar will expand its test lab with more systems from dSPACE.

Mark Yu, Caterpillar Inc., USA

Setup of the HIL system.



Glossary

- Dynasty** – Caterpillar’s multi-domain simulation environment
- Enterprise** – Caterpillar’s high-fidelity engine simulation models
- CGI** – Clean Gas Induction, Caterpillar’s solution for meeting emission regulation
- EPA** – Environment Protection Agency



Major Winnings

at the DARPA
Urban Challenge



Eleven robot car teams out of the original hundred made it through to the finals of the DARPA Urban Challenge 2007. The winning team and three other finalists developed the control systems for their vehicles with dSPACE tools.

The race on November 3, 2007 was the third competition held by the USA's Defense Advanced Research Projects Agency (DARPA) as part of its program supporting the development of autonomous vehicles.

Unlike the previous two competitions, which had the robot vehicles driving through the desert, this time the finals took place in an urban environment. The teams had less than 6 hours to negotiate a 100 km route along the suburban roads of the former George Air Force Base in Victorville, California, about 160 km north of Los Angeles. The vehicles were required to negotiate live traffic while obeying traffic laws and avoiding obstacles, and to compute a new route to their destination if roads were blocked. To mimic road

conditions as realistically as possible and to simulate normal urban traffic, the driverless vehicles had to face 50 additional cars driven by stunt drivers.

All the teams taking part in the DARPA Urban Challenge proved that autonomous vehicles can work. The teams developed their driverless vehicles in an amazingly short time, with the main focus on the further development of driver assistance systems.

Successful Teams with dSPACE Involvement

The winning team, Tartan Racing from Carnegie Mellon University, used dSPACE tools, and so did three other teams: TEAM CarOLO,

Technische Universität Braunschweig, Germany; Team AnnieWay, Universität Karlsruhe; Team Oshkosh Truck, Wisconsin, USA.

Team CarOLO, Technische Universität Braunschweig, Germany

The interdisciplinary Team CarOLO from the Technische Universität Braunschweig consists of five institutes from computer science, electrical engineering, and mechanical engineering, and includes 6 professors, 10 other research associates, and 20 students. Together with students from the relevant university courses, the research associates conceived, designed, implemented, tested and optimized the concept and the software for an autonomous



vehicle called Caroline. The vehicle body is based on a VW Passat and equipped with special sensors such as multiple laser scanners, cameras, radar and LIDAR sensors to detect its environment. A high-performance GPS receiver is also used to ensure that the vehicle's current position is always known precisely. The team used the dSPACE MicroAutoBox as a development tool for the control system (longitudinal and latitudinal dynamics). This allowed various control algorithms to be evaluated quickly by means of rapid prototyping. The team got the project up and running within one year.

The challenge was organized in several stages. The first critical test was the site visit, a kind of quarter-final. The team went to San Antonio, Texas for just four weeks in June 2007 to perform intensive testing at the South West Research Institute. On the day of the site visit, the vehicle had to pass various tests on a circular track.

As project leader, Jörn Marten Wille from Team CarOLO at the Technische Universität Braunschweig was responsible for the development of vehicle control (longitudinal and latitudinal control and trajectory generation).

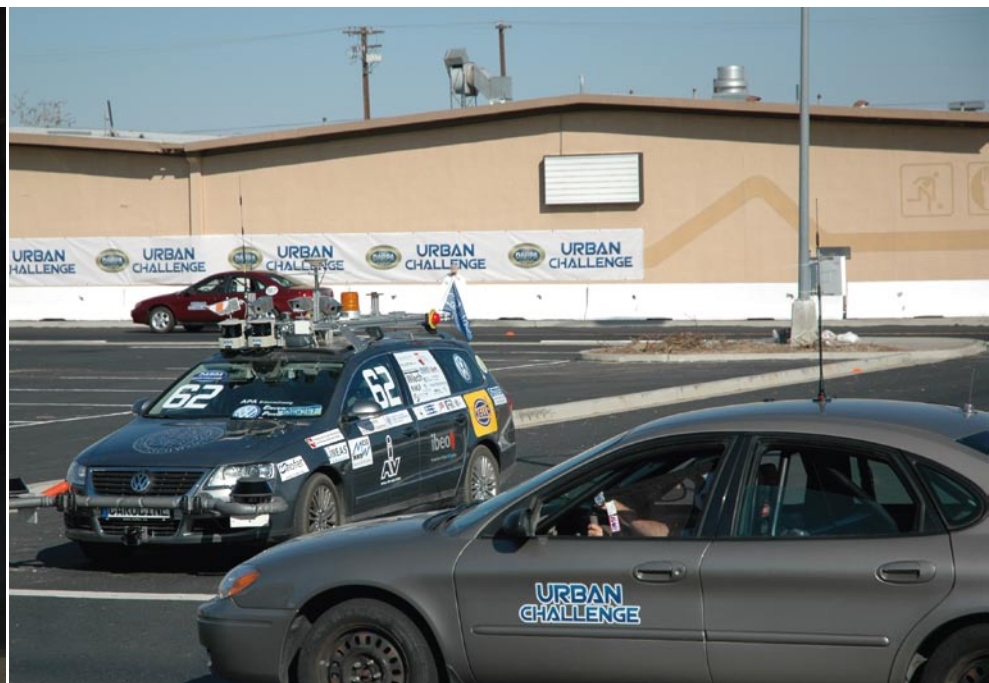
“Within a year and a half, we succeeded in converting a normal, off-the-peg Passat into an automotive robot.”

Jörn Marten Wille, Team CarOLO, Technische Universität Braunschweig

Then the team returned to Germany to do further development work on the vehicle. About five weeks before the actual challenge at the end of October, the team flew to San Antonio again to perform final tests. A few days before the start of the National Qualification Event (a kind of semifinal) at the end of October, the team set off for Victorville, California. For days, the teams had to pass tests set up by the DARPA jury members on unknown terrain. Of the 100 teams that originally entered for the event, only 36 were still around in the semifinal. 11 of these were invited to the finals, which took place on November 3, 2007.

During the race: the autonomous vehicle Caroline encounters one of 50 cars driven by stunt drivers that DARPA put on the scene to create a realistic traffic scenario.

The know-how that the team gathered while developing Caroline will go into a follow-up project that will be carried out by the Institute of Control Engineering in cooperation with the Institute of Aviation and the Institute of Operating Systems and Computer Architectures. Caroline will still be used as a technology carrier for the new autonomous vehicle and for further research on autonomous driving in urban traffic. It is likely to be quite a few years, though, before this scenario is truly mastered. The Urban Challenge was a big step in the right direction, but it was only a very simplified scenario compared to the traffic conditions in a real city.





Moritz Werling from Team AnnieWAY from the Universität Karlsruhe, Germany. Werling was responsible for vehicle control.



AnnieWAY in training with its fellow robot Stanley at Stanford University in Palo Alto, near San Francisco.

Team AnnieWAY, Universität Karlsruhe, Germany

AnnieWAY is a joint team consisting of members of the TR28 special research department "Cognitive Automobiles" coordinated by Universität Karlsruhe and researchers at the Technische Universität München, the Universität der Bundeswehr München, and the Fraunhofer Institute of Information and Data Processing in Karlsruhe. The driverless vehicle is a VW Passat. Its electronic control

system was developed with the aid of the dSPACE AutoBox. The control system is given the trajectory that the vehicle has to follow and receives indicator commands from the host computer via UDP (User Datagram Protocol). The controller is implemented on the AutoBox in C and stabilizes the trajectory by comparing the current position with the desired one. The position is provided by a coupled navigation device connected to the AutoBox via

CAN (DS4302 CAN Interface Board). Finally, the controller's actuating variables for the steering wheel, accelerator, and brake are sent to the converted Passat via CAN. Digital I/Os (DS4002 Timing and Digital I/O Board) are used to engage the gears, to switch the laser scanner, warning lights and status LEDs on and off, and to read DARPA's emergency stop status.

"My adrenalin still shoots up when I hear our car's siren in the video. During the challenge I had to get a grip on myself every time I switched the car 'live', then left it and waited for it to drive off on its own. During trials, we always had a driver at the wheel as a safety backup. It was even worse when the vehicle disappeared from sight. You can't imagine how relieved our team was when AnnieWAY came round that corner after completing the route successfully."

Moritz Werling from Team AnnieWAY from the Universität Karlsruhe, Germany

This system has the following advantages:

- Meets tough real-time requirements, even with a non-real-time-capable PC.
- Safety functions can be implemented directly on the reliable AutoBox (emergency stop).
- Trajectory planning and trajectory control are not coupled.

While there can be no doubt that the challenge has given an enormous boost to the development of self-driving vehicles, it is actually only a sideshow. TR28's real goals are a lot higher (<http://www.kognimobil.org/>), even though achieving them is a long way off. In the future, the vehicles will have to cope without receiving prior information on their environment. A detailed map of the road network like that used by the vehicles in the Urban Challenge might no longer be necessary. A vehicle approaching an intersection would recognize it as such with the aid of its sensors. Data interchange between vehicles is also conceivable. ■

Team Oshkosh Truck, Wisconsin, USA

"We were mainly 3 groups working together – Oshkosh Truck, Wisconsin (headquarters of the Oshkosh company); Teledyne, California; and Parma University in Italy with the Artificial Vision and Intelligent Systems Laboratory (VisLab). We organized weekly teleconferences and many trips to the US to develop and test the system. The teamwork was very successful and well coordinated. We had several critical, very exciting situations. For example, we were still testing the system the night before the race. We – just like everyone else, I believe – had a very tight schedule that did not allow us to include everything we designed and realized. For example, we unfortunately did not use the whole potential of vision (we developed 4 systems and only used 2 of them!) due to lack of testing time. Oshkosh is indeed interested in having these technologies in its vehicles. For Oshkosh and VisLab the Urban Challenge was not just a race and nothing more.

We see a wide range of possibilities to implement these technologies on real vehicles as a retrofit. And we already did so: We demonstrated a palletized load system (PLS) vehicle able to reach a destination, unload, and get back autonomously. We did that just after the Grand Challenge, in January 2006."



“The whole event was amazing! A wonderful experience, even if our vehicle didn’t make it to the finish line as it did two years ago during the DARPA Grand Challenge. We learned a lot from this and previous experiences: We already have 15 years’ experience in the field of autonomous vehicles, but nevertheless this experience let us improve our systems in terms of robustness and maintenance.”

Prof. Alberto Broggi, Team Oshkosh Truck, Parma University, Italy



Prof. Alberto Broggi was responsible for Team Oshkosh in the VisLab team. In particular, his group was involved with sensors for the surrounding environment. They used video cameras and a link between vision and laser for this.

The Oshkosh truck navigating the city streets.



Preparing Students for the Real World

To equip the new generation of engineers for life in industry, universities need to give them more than just theory: Hands-on experience of state-of-the-art development technology is vital. As part of their studies, students at the University of Buenos Aires constructed their own test bench for 3-phase induction motors. Most of the components were funded by partners in industry, with additional support from the university's own development program and endowments. The test bench will also play a key role in the transfer of rapid prototyping technology between the university and industry.

Prof. Hernán Tacca's students use the test bench extensively for their studies and for orders from industry. From left to right: Prof. Hernán Tacca, Pablo Witis, Yaki Nachajón and Gustavo Bongiovanni.



Two of our students in the faculty of engineering required an induction motor test bench for developing control functions as part of their work on their degree theses. They quickly set up the first version of the test bench, which other students developed further in subsequent semesters. The work of Marcelo Bruno, one of these students, was selected to start the joint graduation program agreed between the University of Buenos Aires and the University of Rome "La Sapienza". The test bench is now fitted with modern control technology equipment and has become an indispensable teaching aid. Moreover, it is an important link between the university and industry, and has proved invaluable in industry-related development projects and commissioned work. A joint project between the Argentine company Motortech S.A. and the faculty of engineering at the University of Buenos Aires is currently underway.

Uses of the Test Bench

The test bench for 3-phase induction motors has a wide range of uses: For project-related research, in work on degree & doctoral theses and in cooperation with industry. Our main aim in using the test bench is to ensure fast transfer from theory to practice, i.e., when students have learned something, they can put it into practice at once. The students

develop new control functions and can modify them continuously. They see the results of their work immediately without having to make time-consuming or expensive hardware modifications. This way, students acquire comprehensive knowledge and can familiarize themselves with the latest technology and developments in the field of embedded systems. The AADECA (Argentine Association of Automatic Control) awarded this work the first prize in the development contest for graduation projects carried out by undergraduate students in 2006.

A Worthwhile Investment

From the beginning, we opted to use components from dSPACE. We began with the DS1102 Controller Board. After positive experience with

"We chose dSPACE because we want to prepare our students optimally for the standards used in industry."

Prof. Hernán E. Tacca, University of Buenos Aires

this, we continued with its successor, the DS1103 PPC Controller Board. Real-Time Interface lets us program the DS1103 straight from the block diagram environment in Simulink. This means that new control functions can be quickly implemented and evaluated on the board. We use

Acknowledgements

The success of this work has been possible thanks to the research grants obtained for projects I022 and I003 from the UBACYT Scientific Program of Buenos Aires University and the YPF Foundation donations.

the ControlDesk experiment software to create human-machine interfaces and control real-time simulations.

Why We Chose dSPACE

Advised by Prof. Carlos Godfrid, Professor of Control Engineering at

the University of Buenos Aires with many years of experience in using dSPACE tools, we chose these tools because we want to prepare our students optimally for the standards used in industry. The user-friendly handling of dSPACE tools was another factor, since we can easily connect any hardware to the test bench in plug & play mode to develop and test new control functions. The post-sales support we are given is also first-class. Funding from industry helps us purchase these reliable, high-performance tools. ■

*Yaki Nachajón, Pablo Witis, Pablo Rossi, and Marcelo Bruno (Students)
Prof. Hernán E. Tacca and
Prof. Carlos Godfrid (Advisors)
Faculty of Engineering
University of Buenos Aires, Argentina*

The torque gauge helps the students check the efficiency of the algorithms developed.





ASM Traffic: New Model for Simulating

Traffic Jam on Virtual





Road Traffic

Roads

Imagine you are developing a driver assistance system that has to detect traffic situations and react to them correctly. This is a unique challenge – as is testing function algorithms that have just been developed. Who wants to be driving along in a vehicle, waiting to see how a brand new, untested automated braking maneuver will perform? A simulated test drive is so much more relaxed.

To have a lot happening on the roads during the virtual test drive, a whole bunch of other vehicles need to be driving around in such a manner that the driver assistance system has plenty to do. This is exactly where dSPACE's new simulation model comes in: ASM Traffic puts virtual road traffic into the simulation. In addition to includ-

ing fellow vehicles, it also provides a further indispensable component for this type of assistance system: models for object detection sensors with a 3-D scanning beam, such as radar. Several of these can be "fitted" to the test vehicle. With equipment like this, nothing stands in the way of the virtual testing of new algorithms.

Real-World Experience

DENSO CORPORATION in Japan has been using ASM Traffic in a pilot project since November 2007. The new model is used for function development and testing for ACC systems. Mr. Nishimura from DENSO sums up his experience with the model as follows:

“In DENSO’s previous method, HIL users first had to create Simulink models for fellow vehicles’ behaviors and the radar themselves. Then they had to make timing adjustments to synchronize the traffic with the HIL test case. Now ASM Traffic provides all the models that are required, and also a convenient editor for creating traffic behavior. Events can be used to synchronize the traffic with the HIL test case. The result is a significant improvement in HIL efficiency.”

Takao Nishimura, DENSO CORPORATION

Setting up Road Traffic

At this point, developers will be asking how they can set up the traffic situations their assistance systems will have to face. There has to be a simple, flexible way of creating at least the virtual traffic situations defined in the specifications.

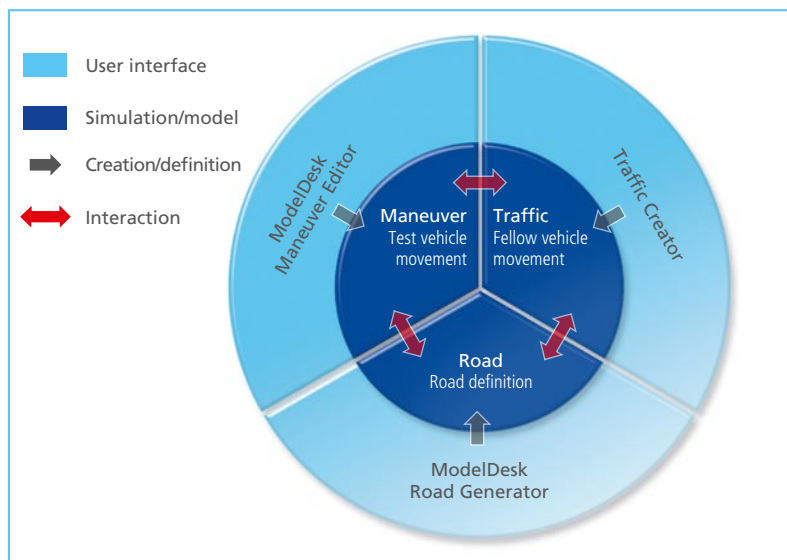
This is why ASM Traffic has a special editor called the Traffic Creator, which can be used to create almost any traffic scenario. In fact, the

enormous flexibility in scenario creation that Traffic Creator offers is ASM Traffic’s particular strength. Each of the 15 fellow vehicles is given its own individual trajectory, which can even change in response to events.

Workflow

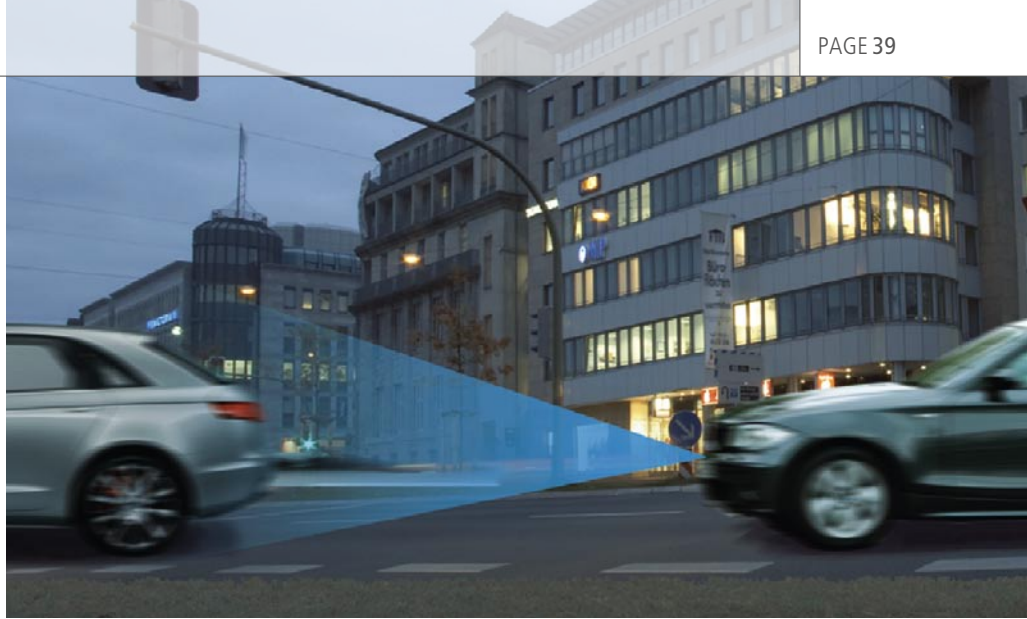
To understand the way the Traffic Creator works, you need to know the overall workflow. First a road

consisting of segments is defined, then the driving maneuver for the test vehicle that will run on the road. The simplest case is a straight stretch of road with a test vehicle traveling along it at a predefined speed. The road and the maneuver are both created in ModelDesk. ModelDesk is the parameterization software for the ASM Vehicle Dynamics Simulation Package, which is the basis for the driving simulation and therefore also needed for running ASM Traffic. Traffic definitions in the Traffic Creator are also based on segments and are therefore fully consistent with road and maneuver definitions. With the Traffic Creator’s options you can define the movements of each fellow vehicle in relation to the road or to other vehicles.



Test scenarios are a combination of road, test vehicle movements, and fellow vehicle movements.

Driver assistance systems such as ACC and precrash are equipped with radar sensors for object detection.



Every Which Way

The trajectories of the fellow vehicles are calculated from distances or speeds, which can be defined either as absolute values or in relation to other vehicles or the road. Basic movement profiles such as constants and ramps are available for segment definition. The transition from one movement segment to the next can be defined freely. The transition criteria can be any event such as approaching a certain speed limit, reaching a predefined distance to other vehicles, or any external trigger signal such as CAN signals or user inputs.

Sensor Model

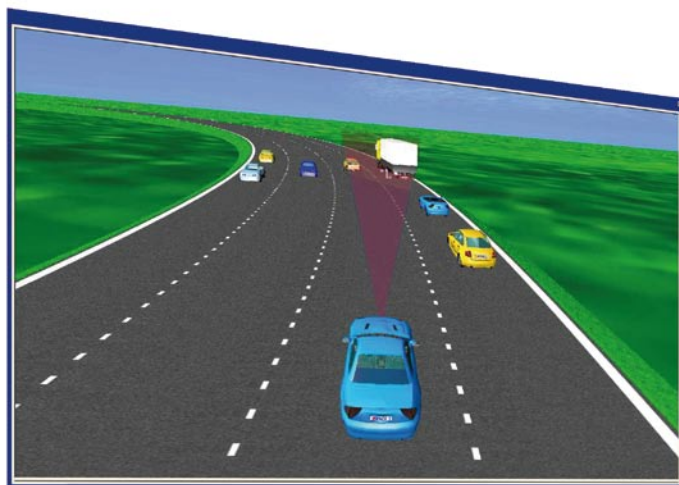
The sensor model in ASM Traffic simulates a 3-D detection range consisting of three layered triangles. The fellow vehicles are treated as cuboids. The sensor model uses a purely geometric approach, so it can also be used for sensor types other than just radar sensors, such as lidar and camera. The sensor model calculates the nearest point of each of the fellow vehicles detected in the sensor's range and returns the distance, the relative speed, the relative acceleration, and the relative horizontal angle for each one.

Visualization and Examples

Visualization of the simulated traffic scenarios is performed by Motion-Desk, the 3-D animation software from dSPACE. ASM Traffic provides seven traffic scenario examples, including special ones such as for oncoming traffic and pedestrians. Thus equipped, the new model can be put into operation quickly.

Applications

ASM Traffic is ideal for hardware-in-the-loop (HIL) testing of adaptive cruise control (ACC) and precrash systems. Function development is supported by Simulink® simulation on a host PC. ASM Traffic can mimic traffic scenarios of all kinds, from simple lane changes to complex scenarios with 15 fellow vehicles. The tests can cover how the ECUs function, and how they behave at the limits of the sensor range. ■



3-D animation of a traffic scenario simulated with ASM Traffic.

Example Application

The following example test case shows how the function of an ACC ECU is tested on a HIL simulator. First the scenario is defined, and then automated testing is performed. Finally, the example explains the results that are obtained. The objective of this example is to show how certain properties of an ACC system can be evaluated.

Defining the Traffic Scenario

The scenario comprises the test vehicle and three fellow vehicles, one of which cuts into the test vehicle's lane. This lane change is triggered when the distance between the test vehicle and the fellow vehicle drops below a certain value. In addition to showing the basic definition options, this example scenario also explains the method of segment-based definition and event-based segment transition.

Segment 1:

The starting point is a maneuver in which the test vehicle T travels along the left lane of a straight road at a speed of 120 km/h. The first fellow vehicle is initialized at a distance of 200 meters ahead of the test vehicle. The longitudinal positions of the two other fellow vehicles are each defined in terms of the distance to the vehicle behind. All the fellow vehicles have a speed of 100 km/h.

The test vehicle T approaches the fellow vehicles in the passing lane at the ACC-regulated speed of 120 km/h. When the distance between the test vehicle and fellow

vehicle F2 drops below 30 meters, a segment end condition is reached. This event triggers the transition to the next segment, the cut-in maneuver.

Segment 2:

Fellow vehicle F2 now performs a lane change into the passing lane (cut in). This brings it into the detection range of the test vehicle's distance sensor. The ACC system now has to adjust the distance between the test vehicle and the fellow vehicle to the desired distance setting.

Test Sequence

Using the defined test scenario the following steps are performed at the HIL testing station to test the ACC ECU:

1. The ACC is switched on and its settings are specified.
2. The test scenario is run.
3. Measurement begins with the fellow vehicle's first steering action.
4. Measurement data is captured until a constant driving state has been reached.
5. The measured data is evaluated.

To vary the simulation, steps 1 to 5 can be repeated with the same scenarios but with different parameters.

Variations

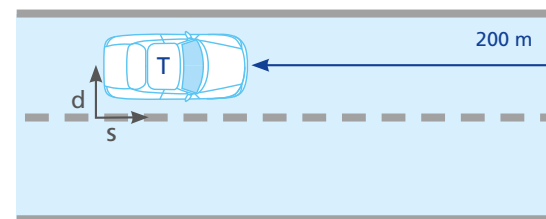
Variations of the scenario can be run automatically in batch mode by using Python scripts in ModelDesk. The variations let the user apply different distance settings in the ACC and different fellow vehicle F2 speeds.

Measurement and Evaluation

For evaluating this use case the following variables will be captured:

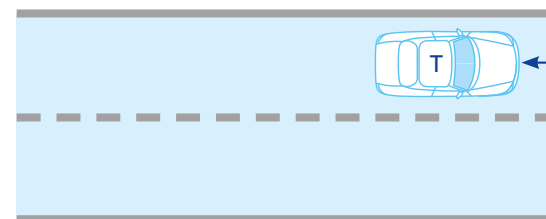
- Speed of the test vehicle
- Deceleration of the test vehicle
- Speed of the fellow vehicle
- Distance between the test vehicle and fellow vehicle F2

Segment 1 (Initial)

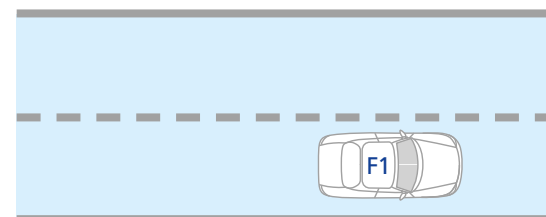


$$\begin{aligned} d_r &= +2 \text{ m} \\ v_r &= 120 \text{ km/h} \\ s_r &= f \text{ (maneuver)} \end{aligned}$$

Segment 1 (End)



Segment 2

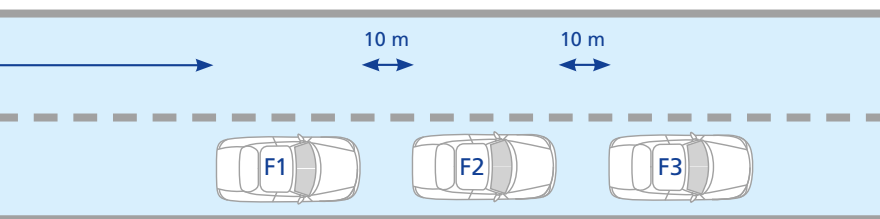


$$\begin{aligned} d_{f1} &= -2 \text{ m} \\ v_{f1} &= 100 \text{ km/h} \end{aligned}$$

The measured signals are evaluated according to the following criteria:

- Response time and control time of the ACC system
- Distance to target object (as specified; as defined by law)
- Controller behavior with regard to comfort (e. g., evaluation of test vehicle deceleration)

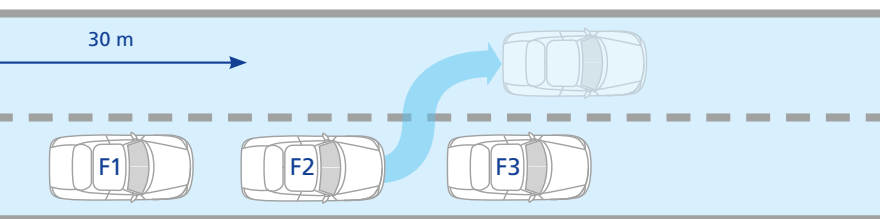
d = lateral position on road
 s = longitudinal position on road
 v = vehicle speed



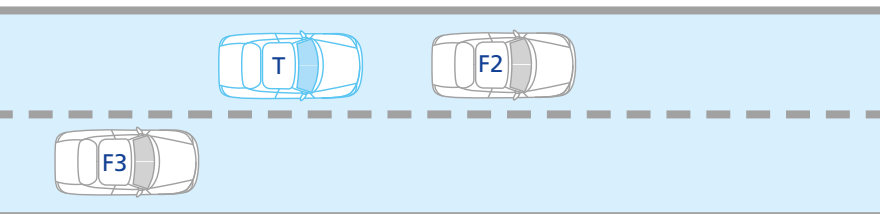
$$\begin{aligned} d_{F1} &= -2 \text{ m} \\ v_{F1} &= 100 \text{ km/h} \\ s_{F1} &= s_T + 200 \text{ m} \end{aligned}$$

$$\begin{aligned} d_{F2} &= -2 \text{ m} \\ v_{F2} &= 100 \text{ km/h} \\ s_{F2} &= s_{F1} + 10 \text{ m} \end{aligned}$$

$$\begin{aligned} d_{F3} &= -2 \text{ m} \\ v_{F3} &= 100 \text{ km/h} \\ s_{F3} &= s_{F2} + 10 \text{ m} \end{aligned}$$



Segment end condition $\text{dist}(F2, T) < 30 \text{ m}$ triggers a lane change that will be performed in the next segment.



$$\begin{aligned} d_{F3} &= -2 \text{ m} \\ v_{F3} &= 100 \text{ km/h} \end{aligned}$$

$$\begin{aligned} d_T &= +2 \text{ m} \\ v_T &= f(\text{ACC}) \end{aligned}$$

$$\begin{aligned} d_{F2} &= -2 \dots +2 \text{ m} \\ v_{F2} &= 100 \text{ km/h} \end{aligned}$$

ASM Traffic's Profile

Product Class:

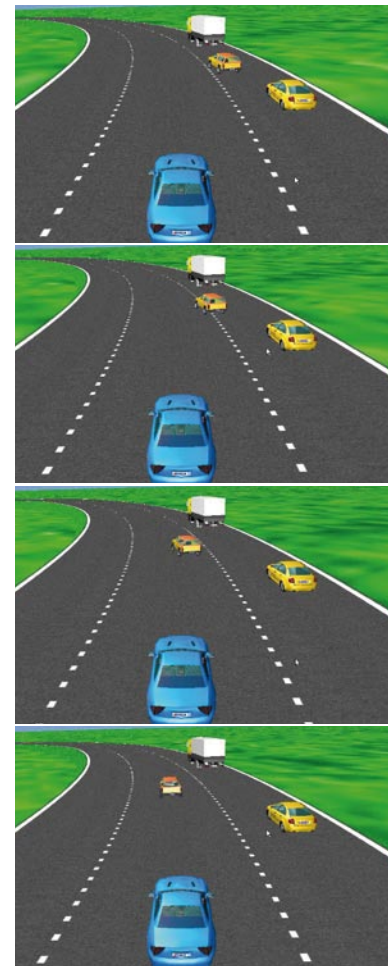
- Simulation of road traffic surrounding a test vehicle
- Extension for the ASM Vehicle Dynamics Simulation Package

Key functions:

- Simulation of 15 independent fellow vehicles
- Radar sensor model with a 3-D beam
- Multiple sensors
- Flexible definition of traffic scenarios
- Online and offline simulation in real time

Applications:

- HIL testing of ACC and precrash ECUs
- Function development for ACC and precrash systems





All you CAN test

From CAN Set-up to Real-Time Test

The RTI CAN MultiMessage Blockset from dSPACE is a proven tool for representing large CAN set-ups flexibly and conveniently. It can now be integrated into ControlDesk via the CAN Navigator and also supports real-time testing, allowing users to switch seamlessly between developing and testing. This article describes a typical multi-tool workflow.

Automotive applications in the fields of rapid control prototyping (RCP) and hardware-in-the-loop (HIL) simulation typically use a CAN bus. Depending on the application area, either the controller or the controlled system is represented by a real-time model which also includes the CAN communication. This bus is configured with the RTI CAN MultiMessage Blockset using a data basis (such as a DBC file) that defines CAN bus signals and messages. The blockset provides a graphical

user interface for Simulink® (fig. 1), from which users can select the necessary Rx (receive) and Tx (transmit) messages. This procedure provides a basic configuration of the CAN bus modeled in the real-time model. Variant handling, transmission control and various signal manipulations can also be configured. This configuration is particularly relevant to HIL applications, as it generates (within the real-time application) the dynamic intervention points to be used later for



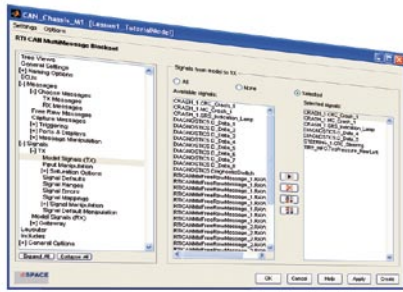


Fig. 1: Selecting model signals for transmission via the RTI CAN MultiMessage Blockset.

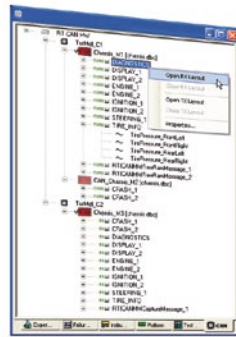


Fig. 2: The CAN Navigator is the central access point to handle CAN in ControlDesk.

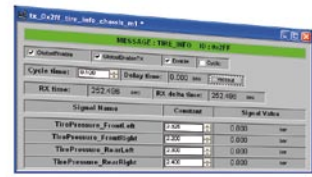


Fig. 3: Tx layouts can be generated directly from the CAN Navigator.

testing and automating the HIL simulator's CAN communication. CAN monitoring in ControlDesk and CAN support for real-time testing are also prepared within the blockset.

CAN Navigator Is the Control Center

In ControlDesk, the CAN Navigator's tree is the central access point for handling CAN. The process of generating code for the real-time model supplies all the data needed for producing the tree: the user just has to insert a reference to the application in ControlDesk and the tree fills automatically (fig. 2). The tree then displays a consistent configuration of the CAN communication as defined in the model and configured with the RTI CAN MultiMessage Blockset.

The tree visualizes all the CAN controllers contained in the model with their assigned DBC files, and the CAN messages with their signals. During run time, it can be used to switch between variants of the CAN controller's DBC configuration. In addition to generating these layouts from the CAN MultiMessage Blockset in advance with Python, as was previously the case, users can now generate the layouts from the tree whenever they are needed – no Simulink installation is necessary

(fig. 3). The layouts directly mirror the send and receive configurations of the messages and signals in the real-time model. Typical examples of such generated layout elements are input fields for setting the cycle and time buttons for transmitting messages sporadically.

Analyzing Communication

Global layouts can be generated to handle the transmission control of multiple messages simultaneously. In the CAN Navigator, it is also possible to create layouts for the online configuration of CAN gateways – this is not possible in the blockset. There is a CAN monitoring window with various views (fig. 4, fig. 5) and sort options for comprehensive analysis of communication behavior. Monitoring can either be based on raw data or be performed symbolically with the current DBC reference.

The messages selected for monitoring can be restricted via freely definable and savable filter rules, but it is also possible to visualize the entire CAN traffic. Pass and stop filters can be applied selectively to IDs, ID ranges and ECUs and can also be combined. The messages visualized in the CAN monitoring window can also be saved to a file (*.csv or *.asc). This file can be used as the starting point for precisely-timed CAN replays of the CAN

communication. For example, communication data recorded during real test drives can easily be reproduced in a restbus simulation on the HIL simulator as often as required.

Testing under Real-Time Conditions

For more complex test scenarios, there is Real-Time Testing (RTT) for the Python-based development of real-time test scenarios. In this, the tests run time-synchronously to the real-time model, and read and write access can be performed to all model variables in each simulation clock cycle.

If the real-time tests have to access the CAN bus, they are developed via a special library (canmmlib) for reading and writing raw-data-based messages. The necessary messages are added to the model by means of the RTI CAN MultiMessage Blockset and are then available in transparent form for real-time tests or for CAN replays of the CAN Navigator. The advantage is that changes to the CAN database or the model structure do not affect the executability of the real-time tests.

An example test scenario is the monitoring of one of the HIL simulator's analog input signals. When a defined trigger threshold (for example, 14.7 V) is exceeded, a

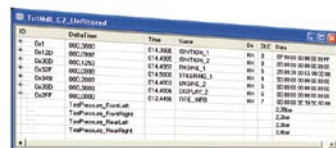


Fig. 4: Static monitor view with the decoded contents of CAN message TIRE_INFO.

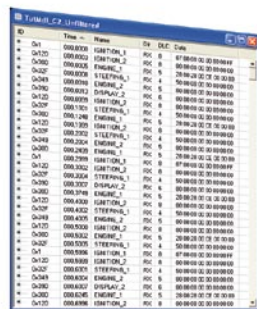


Fig. 5: Continuous monitor view (here sorted according to the time received).

predefined CAN message has to be sent cyclically every 50 ms until the value goes below the threshold again (fig. 6). The associated real-time test implementation is explained in the text inset.

Multi-Tool CAN Handling

The versatile configuration options provided by the RTI CAN Multi-Message Blockset ensure that CAN communication is configured consistently across multiple tools. It covers all the application scenarios that are typical of ECU testing, such as interactive experimentation, CAN monitoring and replay, and

automated restbus simulation. Thus, dSPACE provides a complete, flexible, convenient solution for editing CAN communication. Such solutions will be available for other bus systems in the near future. ■

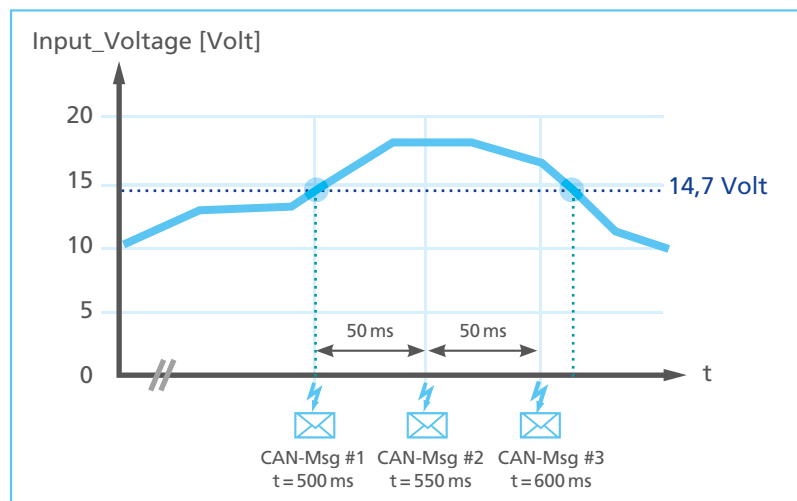


Fig. 6: Example scenario: Monitoring a model signal and its reaction with triggered, cyclic CAN communication.

Real-Time Test Implementation of the Example Test Scenario

In the real-time test, the send data of the CAN message can be defined freely via Python programming. This means that real-time tests can include test-case-specific CAN behavior that is not intended to be a permanent part of the real-time model. Real-time testing is therefore an ideal supplement to the static (model-defined) CAN configurations of the RTI CAN MultiMessage Blockset. The ability to load several real-time tests independently of one another and execute them simultaneously ensures maximum scalability in implementing restbus simulation. The bandwidth reaches from simple, reactive real-time observers (as in the example) to extensive test sequences that are executed completely on the real-time hardware.

```

1.  # -*- coding: ascii -*-
    from rttlib.canlib.controllers import canmmlib
    from rttlib import utilities, variable
    CurrentTime = utilities.currentTime

2.  # Voltage to be observed
    Input_Voltage = variable.Variable(r'Model
    Root/Input_Voltage/Value')

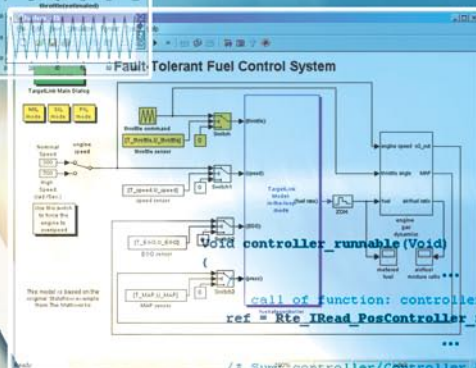
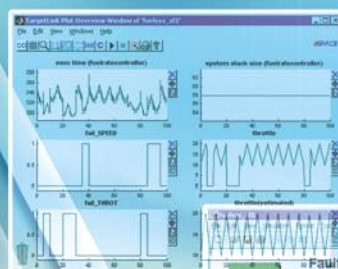
3.  # Initialize CANMM controller
    Controller = None
    ControllerTRCPathName = r"BusSystems/CAN/Chassis_M1"
    Controller = canmmlib.GetController(
        ControllerTRCPathName)
    Channel = Controller.GetChannel()

4.  # Initialize the response CAN message
    Message = Channel.GetRawMessage()
    Message.Format = canmmlib.canmmbase.lib.ftSTD
    Message.ID = 0x123
    Message.DLC = 8
    Message.TX.Data = 0x1020304050607080

5.  # Wait helper function
    def WaitGen(Duration):
        EndTime = CurrentTime.Value + Duration
        while (EndTime > CurrentTime.Value):
            yield None

6.  def MainGenerator():
        # Observe Input Voltage continuously
        while (1):
            if (Input_Voltage.Value > 14.7):
                Message.Transmit()
                yield WaitGen(0.05)
            yield None
    
```

1. Import real-time testing libraries (canmmlib, etc.)
2. Generate the variable object of the voltage to be monitored
3. Select the CAN send controller
4. Define the Send message with the real-time test
5. Help function for precise time measurement (for later transmission of CAN messages at exact time intervals)
6. Real-time test sequence: Check the voltage once per simulation step and react by sending a cyclic CAN message



```

call of function: controller/Controller_Runnable/Recap
ref = Rtc_IRead_PosController_rport1_ref();
...
/* Sum of controller/Controller_Runnable/e */
S12_e = (Int16) (((UInt16) ref) - ((UInt16) Rtc_IrRead_
...
/* TargetLink output: controller/Controller_Runnable/Out
call of function: controller/Controller_Runnable/Recap
# combined # update of inport for controller/Controller
Rtc_IWrite_PosController_pport1_up1(((Int16) (S12_sPI << 2)

```

MISRA

Safe Modeling

MISRA: Modeling Guidelines for TargetLink



MISRA, the Motor Industry Software Reliability Association, has published MISRA modeling guidelines for TargetLink. The new guidelines support function developers and software developers in implementing vehicle functions with regard to functional safety. TargetLink is so far the only production code generator for which such MISRA guidelines have been published.

Transparent and Safe Modeling

The new MISRA guidelines for TargetLink contain requirements for modeling with regard to functional safety and are particularly relevant to safety-critical projects. They reflect the greatly increased importance of model-based design and automatic production code generation. The main issues covered are:

- Avoiding specific modeling elements and modeling styles in MATLAB®/Simulink®/Stateflow® for reasons of functional safety
- Modeling elements that support a transparent model structure to simplify model reviews

- Instructions on specifying implementation information to ensure that the generated C code complies with a maximum number of MISRA C:2004 guidelines

The new MISRA guidelines for TargetLink are available at the official MISRA Web site, www.misra.org.uk ("MISRA AC TL – Modelling style guidelines for the application of TargetLink in the context of automatic code generation").

New Version of the dSPACE Modeling Guidelines for TargetLink

In addition to the MISRA TargetLink guidelines, TargetLink users can also

refer to dSPACE's own modeling guidelines, the latest version of which is 2.1. These guidelines are an ideal supplement to the MISRA TargetLink guidelines and the MAAB guidelines (MathWorks Automotive Advisory Board). The dSPACE TargetLink Guidelines are available on request from dSPACE Sales. They are also available online as part of the e-Guidelines: <http://www.e-guidelines.de>. ■

Please note:

MISRA does not endorse or make recommendations for any specific manufacturers, their products, or services.

Interview

with Michael Beine,
TargetLink Product Manager
(dSPACE)



Mr. Beine, could you briefly explain the practical significance of the new MISRA guidelines for TargetLink users?

Imagine you want to model a vehicle function in Simulink®/TargetLink and then implement it as production code. How do you make sure, for example, that the generated code is sufficiently compliant with the MISRA C guidelines and suitable for code reviews, yet still fits the limited resources available on the ECU? Do you always know precisely whether the modeling constructs you use are transparent in Simulink®/Stateflow® and have no hidden pitfalls? The MISRA modeling guidelines for TargetLink give you useful tips and instructions that will benefit your production projects.

What do the guidelines mean for TargetLink as a product, which after all has been on the market for many years?

We're very pleased that the MISRA C guidelines, which are widely used at code level, are now supplemented

by the MISRA TargetLink guidelines at model level. The MISRA TargetLink guidelines show that TargetLink is firmly established as the de-facto standard for autocoding in the automotive sector. Production applications using TargetLink can be found in all automotive domains such as powertrain, body electronics, chassis, driver assistance systems, vehicle safety, and even infotainment. The TargetLink modeling guidelines from MISRA give TargetLink users additional support in these applications, especially with regard to functional safety. Incidentally, dSPACE also provides its own TargetLink modeling guidelines, which cover further aspects such as AUTOSAR and achieving maximum code efficiency.

Thank you for talking to us, Mr. Beine.

Anniversary of an Idea

Twenty Years of Continuous Success





A conversation with Dr. Herbert Hanselmann: To mark dSPACE's 20th anniversary, the company's founder and president describes how an idea became an internationally successful company.

Dr. Hanselmann, before we talk about the early days of dSPACE, could you tell us how you became interested in control engineering?

When I was 16, I read an article about cybernetics. I didn't understand everything, but the article fascinated me and sparked my interest in this subject. When I decided which subject to study at the university, I chose electrical engineering, and from there it was almost inevitable that I would specialize in control engineering. I also had a side job developing controls for the machine tools of a small neighborhood company.

What events encouraged you to found the dSPACE company?

While I was still working on finalizing my doctoral thesis at Karlsruhe University, I was approached by a distinguished researcher working at

Mercedes-Benz, Joachim Lückel. He had been granted a professorship at the University of Paderborn, with the task of setting up a completely new and novel institute.

I had really planned to go into industry, but this project was too interesting to say no. The main objective was to implement something that was known as "modern control engineering", using relatively fast mechanical systems. In other words, mechatronics. Just one small thing was missing: We didn't have any computers that were fast enough to calculate the higher-order control functions in real time.

In 1981, partly working at home like a hobbyist, I constructed an Intel kit for a fairly exotic analog-digital signal processor. You couldn't expect mechanical engineers to do the required optimal microcode program-



“The gist of Professor Leonhard's evaluation was: If they don't try to do too many things at once, it might work.”

Dr. Herbert Hanselmann

ming. Therefore an automatic code generator and a couple of associated tools were quickly developed. This processor passed its first real-life test in a Mercedes bus with electronic lane keeping. After numerous further developments and successful applications, we had the idea of producing something similar commercially for worldwide use. Three colleagues, involved in this university project in different functions, and I founded dSPACE GmbH in 1987, and on January 2, 1988 we started work on developing an almost completely new toolset.

What obstacles did you face when starting the company?

The know-how was there, but the products we wanted to market still had to be developed. One year was scheduled for developing the first toolset – a year in which there

would be no sales. Fortunately, loans and subsidies were available for such cases. To obtain these, we needed more than just brilliant ideas.

We had to convince people that our plans were viable. So one of the things we did was to acquire statements on our chances of success from people who were specialists in the field and seemed unbiased. The appraisal given by Professor Leonhard from Braunschweig, a world-renowned electrical drives specialist, was critical in a positive way. The gist of his conclusion was, “If they don't try to do too many things at once, it might work.”

What products did it all begin with, and for what applications?

The hardware consisted of boards with a signal processor from Texas Instruments – which in those days

still used fixed-point arithmetics – and several I/O boards dedicated to control engineering. The software was a toolset for preparing and simulating state controllers and generating code in DSPL, a language that we defined ourselves and which we also wrote the compiler for. We placed great importance on optimal machine code: “C” was not an option.

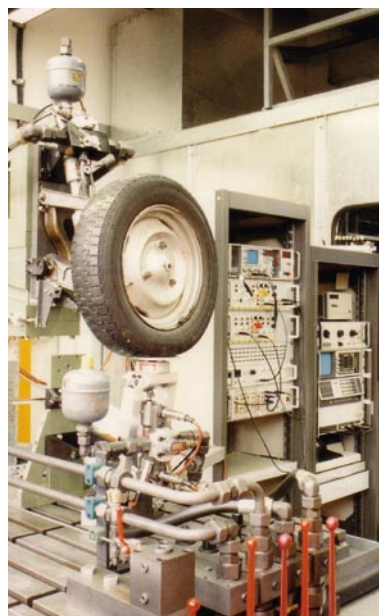
Who were the first customers?

Philips is one I remember. Their Manufacturing Technology Center had a load of unsolved control tasks, including one for wafer steppers with a positioning precision of nanometers. One particularly unexpected customer was Hilti from Liechtenstein, known all over the world as a manufacturer of high-quality rock drills. They also did an astonishing amount of mechatronics. Basically, our first customers were engineers who lacked the means to implement fast controllers quickly, just like us a few years previously.

What was the most interesting project with dSPACE tools at that time?

That's a difficult question, there were so many, including really exotic ones like the control system for hydraulically shifted masses used as earthquake dampers at the top of skyscrapers in Japan. Walt Disney needed our equipment to control one of their theme park attractions.

The starting point for further developments: A test bench used by the University of Paderborn in the 1980s for research on mechatronic control systems. Among other things, it was the basis for the active wheel suspensions used in today's vehicles.

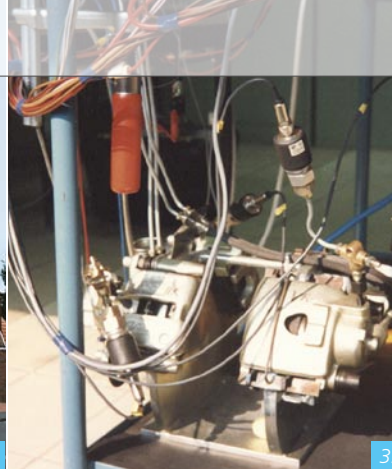




1.



2.



3.



4.

And a Japanese company was experimenting with active noise canceling in windows for new housing construction. This is understandable, considering that in Tokyo, the highways run just a few meters away from people's living-room windows.

What was your greatest challenge?

In the starting phase, it was having enough staying power in the second year, when things progressed more slowly than the business plan said they would. The costs went according to schedule, the sales didn't. That's not unusual, as costs are simply easier to plan for. But selling the necessary numbers of a completely new kind of toolset for very special applications just takes time. This aspect is frequently underestimated. The greatest challenge of the past few years was when we had to get a whole Japanese subsidiary up and running in almost no time. With no prior warning and for reasons that were beyond our control.

What was your biggest success?

Convincing The MathWorks to cooperate with us to give Simulink, which was just being developed then, a real-time extension. There were only 6 people at dSPACE then. The MathWorks was already ten times as big and had never entered into a cooperation like that before. In the end, the project was a really important key to success for both companies.

What events and developments are you particularly pleased with?

The best thing is that we have been growing organically for so long now and have always been able to stick to our core competencies. The context has changed repeatedly. Competitors appeared on the horizon and, luckily, some of them disappeared again. Focuses have shifted. New but related fields were added. I've also heard this from customers: it's our continuity that is admired.

1. dSPACE's four founding members. From left to right: Dr. Herbert Hanselmann, Albert Schwarte, Dr. Hermann Henrichfreise and Andreas Hostmann.

2. dSPACE started out on the second floor of this building on January 2, 1988. Step by step the company expanded to fill the basement and the attic, until it moved into its own building on January 2, 1995.

3. Components of dSPACE's first HIL simulator (fixed-point processor). The photograph was taken in the customer's laboratory (Fatec) in 1989.

4. In the early 1990s, the basement was the hardware laboratory for performing tests.

"An engineer without the ambition to deliver the best work possible that he or she can be proud of is not really an engineer."

Dr. Herbert Hanselmann



The first turn-key HIL simulator in 1995. A cable from the simulator (multiprocessor system based on the TMS320C40 DSP, Tesis model) goes from an upstairs window down to the car to test its ABS and electronic differential. The workflow: Sit in the car, start the engine, drive off (in stationary mode) and brake to activate the ABS. The Audi 8 was on loan, so it had to be chained up.



1. First presentation at Convergence 1990 in Detroit. Rent the computer, monitor and oscilloscope, spread out the boards and the brochures, hang up two self-made posters — ready to exhibit.



2. Product presentation at the American Control Conference in Boston in 1991. Professor Skelton from University of California, San Diego (UCSD) scratches his head while considering if his budget can cover a dSPACE System.



3. In 1992, we had to do the impossible. Our mission: to transport the demo equipment on the overcrowded Japanese subway.



4. One of the first product demonstrations of the Real-Time Interface (RTI) in 1992. Japan is still one of the first countries where new products are presented today.

dSPACE has a strong position in the automotive industry. Where would this industry be without products like those from dSPACE?

No one knows. Electronics would definitely have arrived on the scene without us, but probably more slowly. I sometimes say that our tools enabled our customers to develop so many new functions for their vehicles in the 1990s that afterwards they needed hardware-in-the-loop test systems to keep up with burgeoning new developments in terms of quality. It looks like dSPACE had a clever strategy, but honestly, it wasn't strategy, things just turned out that way.

What will the automotive industry's major innovations be over the next 10 years, and how will dSPACE be involved in them?

Let me focus just on development processes. When it comes to innovations in vehicles, there are people better qualified than I am. A recurrent situation with development processes and tools is that progress made on one point generates new challenges somewhere else. Rapid control prototyping ultimately necessitated automatic production code generation. Now that this is "in the can", there are innumerable software modules and the complexity is overwhelming. New tools are emerging, and indeed they have to. For example, we have produced System-Desk as our contribution to master-

ing complexity. More coordination and validation has to be front-loaded. At the moment, it all seems like a bottomless pit. And as for the time constants at which technologies like rapid control prototyping, auto-coding and HIL have spread, we're looking at an average of more than 10 years till even one essential process technology becomes established and really widespread. We will definitely not be running out of things to do any time soon.

As the company founder, you have a very special relationship with dSPACE. What role does dSPACE play in your life?

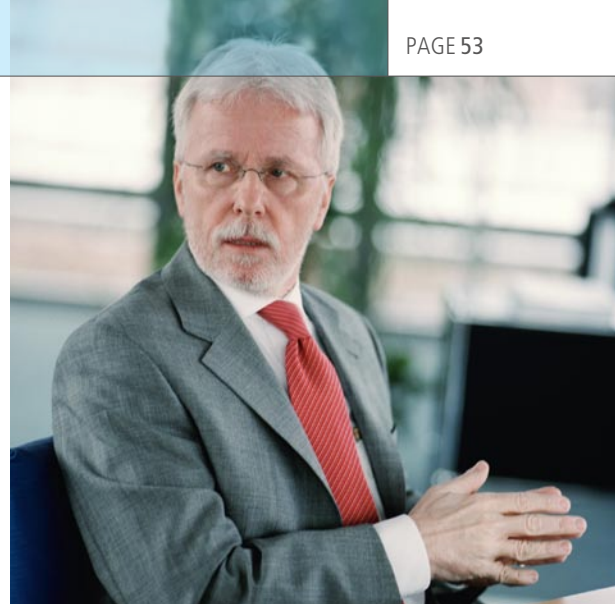
The main role of course, next to my family. A company like dSPACE, which is constantly evolving, demands my complete attention. When it leads to success, which fortunately is the case with us, that is very satisfying.

On the subject of human resources: What do you look for when hiring employees?

By far the most important thing is spirit. People only work well if they enjoy it, and they enjoy it if the tasks are challenging but solvable. An engineer without the ambition to deliver the best work possible that he or she can be proud of is not really an engineer. The important thing is intrinsic motivation, and I can safely say that dSPACE people have plenty of that.

“We haven’t wasted our time on acquisitions or peripheral activities, and we have extended our range bit by bit without abandoning the core.”

Dr. Herbert Hanselmann



Are there any developments or circumstances you are dissatisfied with?

This is not a political magazine, so some topics won’t be touched. However, I’m definitely dissatisfied with the lack of suitable applicants for all the jobs that we always advertise. So far, we have invested a great deal of energy in recruitment and will continue to do so. But if there is nobody to recruit, it’s not enough. Like many other companies, we have to do our part to ensure that more young people keep to the straight and narrow and study the right subjects. This is why we set up our ProMINT initiative, with student grants, information activities, and school partnerships.

To come back to the expert opinion you mentioned before: Has dSPACE really not done too many things in the past 20 years, or how would you describe the company’s success?

In a manner of speaking we did follow Professor Leonhard’s advice.

Basically, we’re still doing the same thing. We’re connecting control engineering, software, electronics. Instead of producing custom devices, we make products that are useful to many people so that the high cost of development pays off. We haven’t wasted our time on acquisitions or peripheral activities, and we have extended our range bit by bit without abandoning the core.

How do you see the company’s future?

As long as there is enough to do, it will be a simple extrapolation, fairly linear. The tasks are growing, so we are growing too. New issues emerge, we provide the solutions. We can carry on like this for a very long time.

Are you still going to be on board in 20 years’ time?

Yes, on board something floating on the Mediterranean.

We wish you fair trade winds, now and in the future. Thank you very much for talking to us,

Dr. Hanselmann. ■

1. The AutoBox is known for its high reliability under extreme mechanical conditions. The shake test performed in 1993 shows why. Andreas Hostmann is definitely impressed.
2. Professor Joachim Lückel greeting the Hanselmanns at the housewarming party for our first very own office building in 1995.
3. Compared with the very first HIL, the systems have grown to an impressive size – since it is no longer just the ABS that is simulated, but whole vehicles with all their comfort functions and driver assistance systems. The picture shows a complete, networked HIL system – the basis for a virtual vehicle.





Tomorrow's Engineers

Hybrid Bicycle:
dSPACE Supports Young Researchers

A hybrid bicycle that stores braking energy and then releases it to support acceleration: This exciting new idea took three young researchers from a Paderborn school, and their physics teacher, to first place at the regional Young Scientists Competition. dSPACE provided the funds for acquiring important components and tools for this project and also helped out with advice to the young engineers. This sponsorship was part of dSPACE's ProMINT initiative (mathematics, informatics, natural science, technology), which aims to motivate school students to study scientific and technical subjects at university level.



The young scientists Wolfgang Fritz, Georg Enzian, and Stefan Beller (left to right) from the Theodorianum School in Paderborn together with the hybrid bicycle.



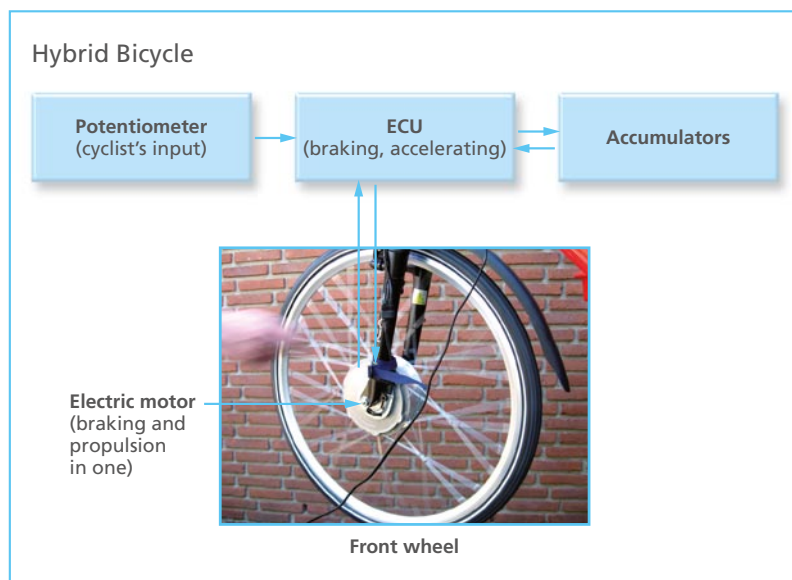


Hybrid Bicycle: Power to Start

Cycling is healthy and environmentally friendly. But repeated stop-and-go – braking and then starting or accelerating again – can put quite a strain on the muscles. Long journeys can be very tiring, and often enough, people resort to their cars for this very reason, even though they would really like to do more cycling. But what if the energy that gets lost in braking could be harnessed for starting off again? It can be done – as three students from the graduating class at the Theodorinum School in Paderborn notably demonstrated, with the help of their physics teacher and advice from dSPACE engineers. The result is impressive: a hybrid bicycle with an energy recovery system that captures most of the energy used in braking and releases it via an electric motor when the cyclist pushes off again or accelerates. The electric motor, which is installed in the front wheel hub, is actually a brake and a propulsion unit in one. Batteries are used to store the energy. A simple electronic control unit regulates the

flow of energy, and the cyclist uses a potentiometer to brake and accelerate. Programming the micro-controller was a particular challenge, and the students mastered it brilliantly. No wonder the innovative hybrid bicycle took the Paderborn students straight up to a well-earned first place in the Young Scientists Competition at regional level,

winning second prize at state level and also a special award. Congratulations! ■



A hybrid bicycle that stores braking energy and releases it for accelerating – this innovative prototype won the first prize in the Young Scientists Competition for three Paderborn students. dSPACE supported the young researchers financially so they could buy components and tools, and gave them technical tips on implementing their idea.

Interview

Dr. Herbert Schütte,
Director of Applications &
Engineering (dSPACE)



Dr. Schütte, what was it like providing consultation services for a student project, instead of for a major automotive customer?

I was really impressed by the dedication and technical creativity of the students. These are precisely the qualities that tomorrow's engineers need. Never give up and be prepared to try unusual solutions – even in industry, that's the only way we can develop the technology of the future. My engineering colleagues and I had great fun working with the students, particularly since they had a surprisingly high level of knowledge of software and hardware. I hope I'll have enough time outside my core activity as head of customer applications and engineering at dSPACE to continue advising on ProMINT projects.

In what way do you think dSPACE can stimulate young people's interest in technology and science?

Well, to develop an interest for any subject – and this doesn't only apply to technology, of course – you first have to come into contact with it somewhere, somehow. Most young people handle electronic systems on a daily basis: cell phones, cars, cameras, etc. But sadly, the majority have no idea how they function or how they're developed. How can someone know that they might enjoy developing electronic controls, when the subject is not even touched during their school career? Obviously, in view of the

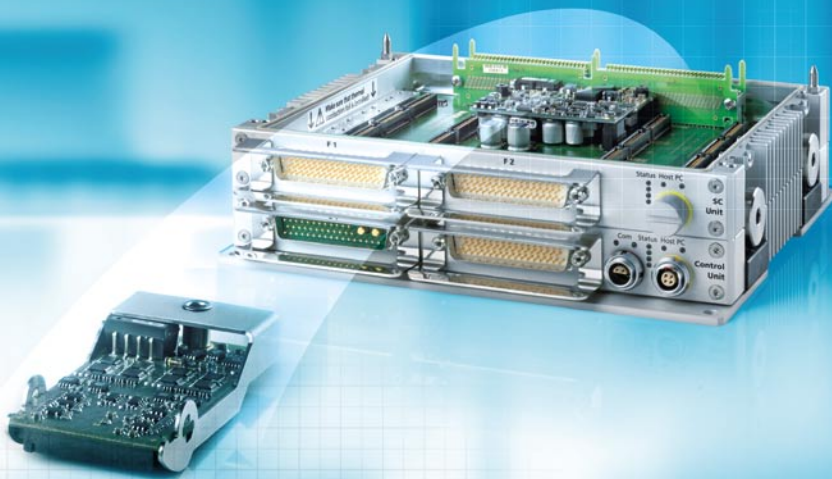
rapid expansion in knowledge, only a fraction of topical issues can be tackled in school, that much is clear. So control engineering is just one example among many, though it is one of particular interest to dSPACE. What dSPACE above all can and will do is to provide hands-on experience in technology in the broadest sense of the word and stimulate technical interest. And we can do it well: by giving exciting presentations in physics classes, for example, hosting school visits to our facilities, or supporting projects like the hybrid bicycle. With the experience we have had of ProMINT projects so far, I'm very confident that we already strengthened several young people in their decision to study a technical or scientific subject. And if they later turn into young dSPACE engineers, I'll be even more pleased.

Thank you for talking to us, Mr. Schütte.

ProMINT

With the ProMINT initiative (mathematics, informatics, natural science, technology), dSPACE is making an active contribution to nurturing the next generation of engineers and scientists. As part of ProMINT, dSPACE is involved in organizing physics and computer science lessons in schools, and also provides university grants.

www.promint.de (German version only)



RapidPro: Integrating DENSO Lambda Probes

dSPACE's RapidPro hardware will soon make it possible to integrate DENSO lambda probes (PLUS2.1, PLUS3.x) into a function prototyping set-up (the Bosch LSU 4.2 and 4.9 probes are already supported). dSPACE offers an extensive selection of signal conditioning and power stage modules, for example, for integrating crankshaft, camshaft, and knock sensors and for controlling high-current injectors. The DENSO

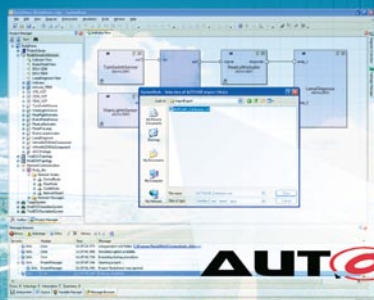
broadband probes supported by the new RapidPro SC-UHEGO 2/1 module can capture air-fuel mixes in the range AFR 10 ... ∞ bzw. 12 ... 25. dSPACE provides an example model for integrating the lambda probes into the MATLAB®/Simulink® modeling environment. Release date for SC-UHEGO 2/1: www.dspace.com/releases

SystemDesk Supports AUTOSAR 2.1

dSPACE is continuing to extend the AUTOSAR support given by the SystemDesk architecture software. Version 1.1 of SystemDesk (planned for July 2008) will support AUTOSAR

2.1. This version of the AUTOSAR standard includes more precise specifications for developing a standardized electrics/electronics architecture, so users now have improved compatibility and greater stability in interactions and data exchange between tools for electronic control unit development.

For information on the release dates for SystemDesk 1.1, please visit www.dspace.com/releases



A Winning Team

Once again, dSPACE took part in the Easter Run in Paderborn, home of our headquarters in Germany. This is the oldest road race in Germany and attracts top-level international competitors. This year, 8,000 runners gathered at the starting line, undaunted by adverse weather conditions. The dSPACE team took the first-place trophy in the 5-kilometer Company Cup, a competition held for the first time this year. A total of 37 dSPACE runners took part in the run. To get ready, they all trained together once a week after work, inspiring even inexperienced runners to join in. In the official handbike races, the athletes were delighted that dSPACE sponsored their events, like last year. With spirits high, the handbikers received their prizes in the dSPACE hospitality tent.



dSPACE direct



Would you like up-to-the-minute information on releases, events and activities? Then why not subscribe to our monthly e-mail newsletter, dSPACE direct.

www.dspace.com/dspace_direct

Request Information



Please check the corresponding field on your response card and return it

- By mail
- By fax to +49 52 51 – 6 65 29
- or request information via our Website:
www.dspace.com/magazin

Your opinion is important. Please send your criticism, praise, or comments to dspace-magazine@dspace.com – thank you!

Jobs



Are you an engineer who is just graduating?
Are you looking for new professional challenges?
Then come and join our team in Paderborn, Munich or Stuttgart, Germany; Paris, France; Cambridge, United Kingdom; Tokyo, Nagoya, Japan or Detroit, MI, USA!
Due to our continuous growth, dSPACE is looking for engineers in

- Software development
- Hardware development
- Applications
- Technical sales
- Product management
- User documentation

Current offers at
www.dspace.com/jobs



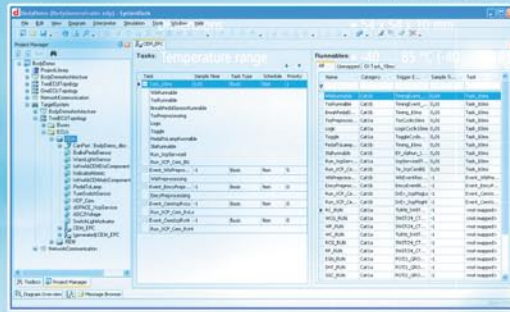


www.dspace.com

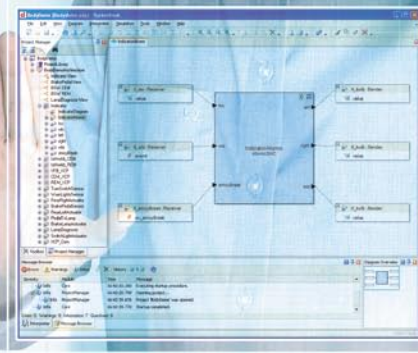
Chip select configuration

- Power dissipation < 350 W in typical operating conditions
- Stand-by current < 5 mA @ 12 V

- Option to select chip select line numbers 0 ... 3
- Two options for chip select assignment:
 - a) One chip select line for both RAM and DPMEM
 - b) Two separate chip select lines



ambient temperature



System Architecture

Rapid Prototyping

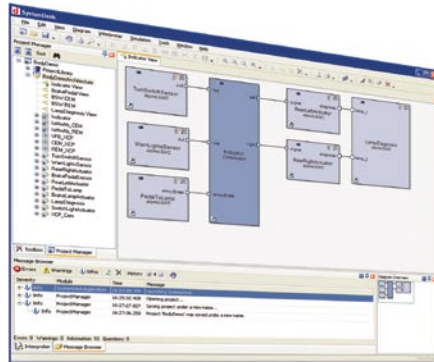
ECU Autocoding

HIL Testing

ECU Calibration

SystemDesk

Now it's easy to see what's coming tomorrow.



Automotive control architectures are becoming more and more complex. This presents developers with challenge after challenge. What's the answer? SystemDesk from dSPACE. With graphical system modeling, SystemDesk facilitates the development of automotive electronic systems and subsystems by giving you a more complete picture. It makes models easier to understand and helps you communicate better. Helps you avoid inconsistencies and redundancies right from the beginning. Makes it easier to exchange and integrate software between OEMs and suppliers. Detect errors earlier. And reuse software components that were already tested. Plus, it's backed by AUTOSAR support. SystemDesk. The tool you need to keep up with the ever-changing challenges of developing automotive electronic systems and subsystems.