Contents

Customers

Products

Miscellaneous

dSPACE
NEWS

facts
projects
events

**TargetLink
now a
Global Player –
Production code
straight from Simulink**

Spring 2001

EDITORIAL

Dr. Herbert Hanselmann

We are now looking back on 16 months of TargetLink product history, and I feel this product once more deserves some attention in this column. There are customers who have gone far beyond the 'proof-of-concept' stage. TargetLink is already in production ECUs and is used in projects without the backup option of "if it doesn't work, we still can do it by hand." It has to work, and it does.

Just a few days ago I received a letter from one of our more aggressive customers (aggressive in the sense of making full use of TargetLink and squeezing the maximum out of it). About a year ago they were faced with a project with such tight timing that they would not have taken it on at all without TargetLink. On top of this, the processor and compiler were not only new to them, but also to dSPACE. The supplier offered very little RAM and ROM, and the driveable system was to be presented to their vice president of passenger car development within 4 months, as a real ECU, and not a prototyping system. The system was presented, approved, and the software released after just 6 months. The customer particu-

larly emphasized the partnership he experienced with dSPACE. Changes and customizations were quick and to the point. It is no surprise that this customer is among those with a large number of TargetLink licenses.

Speaking of which, TargetLink sales have been increasing fast over the last few months. That's even before version 1.2 and the big enhancements coming with the next major release, some of which are critically important for certain customers. Clearly TargetLink has left behind the "let's see if it works in a real project" stage.

CUSTOMERS

- ◢ **TargetLink cuts development time by approx. 40%**

- ◢ **Seamless design chain based on dSPACE tool suite**

- ◢ **TargetLink generated code will go into series ECU**

Coming Soon: Steer-by-Wire at ZF Lenksysteme

X-by-wire: A hot topic for all car manufactures and suppliers when exploiting potential areas for improving vehicles. In this context, ZF Lenksysteme (ZFLS) is developing the first generation of steer-by-wire systems, planned to be available on the market in the near future. To develop the corresponding electronic control unit (ECU), ZFLS opted exclusively for dSPACE tools. Function prototyping is performed with dSPACE Prototyper, and to generate the code automatically for the target hard-

ware, ZFLS uses TargetLink. By choosing TargetLink as the production coder, ZFLS estimates a cut in development time in the range of 40%.

Steer-by-wire Basics
To establish steer-by-wire systems, the conventional architecture is replaced by electrically and electro-hydraulically driven actuators. The steering forces are generated by an induction motor, which is controlled by the corresponding motor angle sensor. To give the driver the familiar feeling, a brushless DC-motor provides torque feedback at the steering wheel. For safety reasons a hydraulic backup system provides steerability under all circumstances.

Benefits
By using dedicated controllers, the steer-by-wire system revolutionizes steering functionality, in the form of vehicle state dependent steering ratio, automatic

parking assistance and lane keeping control, just to list the most obvious examples. Furthermore this system allows vehicle dynamics control by automatic

Feedback actuator with steering wheel.
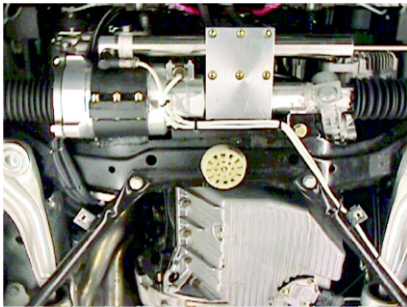
Underside of the engine compartment with steering actuator.

steering interventions. Due to the missing intermediate shaft, there is also potential to improve crash performance and steering system integration: left- and right-hand-drive vehicles follow the same mechanical set-up. Steer-by-wire provides excellent flexibility regarding modifications to the steering characteristics. If a specific steering design is to be installed on a different car type, only the software code needs to be altered in order to modify the desired performance.

The Procedure

To develop the ECU software, we designed the controllers graphically in the MATLAB/Simulink environment. To evaluate the control functions, the Simulink model, consisting of 355 blocks, was transferred to dSPACE Prototyper. Onboard verification and parameter tuning was performed on

dSPACE AutoBox, which features a sample time of 1 ms. The next step is often considered the most critical and error-prone. The Simulink model must be transferred accurately into executable C code in order to run on the target processor, the TMS470 from Texas Instruments. The code must represent the graphical specification exactly. With TargetLink, we found a tool which precisely meets our requirements, such as

- flexible variable naming according to our programming directives
- manual and automatic variable scaling
- floating- and fixed-point simulation

Equally important are the code properties with respect to readability and documentation. Since the report file is generated automatically, we value the good project overview provided by TargetLink. dSPACE's commitment to our success has not gone unnoticed: In cases where we needed some additional assistance, the technical support was friendly, quick and competent. With respect to the training ef-
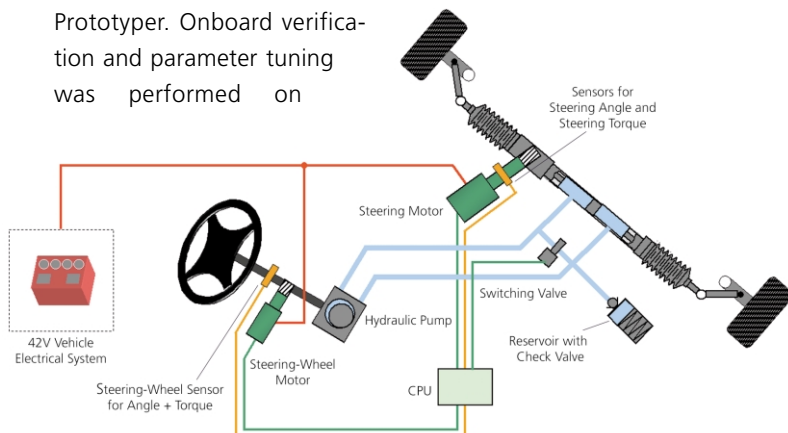
fort, we regard TargetLink as an easy-to-use tool that fits ideally into our development environment. Since we were already familiar with Simulink, it took us only two weeks to achieve useful results. This is significantly less than competitive products.

Speeding It Up

TargetLink is the stepping stone between the prototyping stage and the target hardware. It provides a seamless design approach, which is a frequent re-
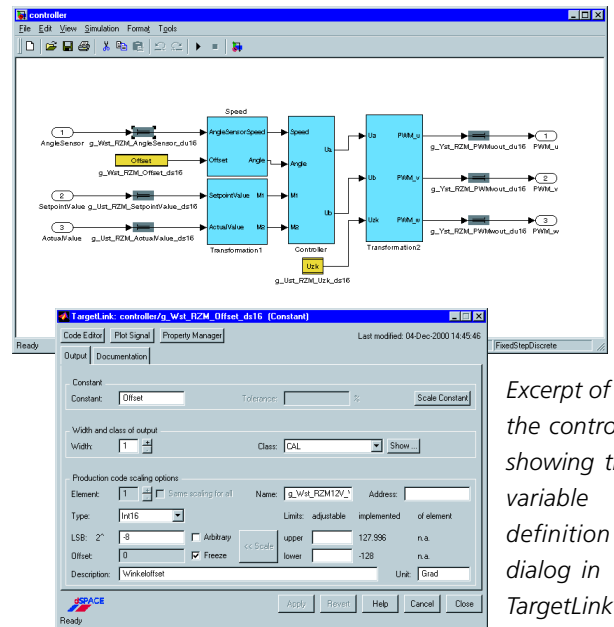
Excerpt of the controller showing the variable definition dialog in TargetLink.

quirement of our customers. Modifications made within the Simulink model are transferred accurately into C code. This speeds up the development process significantly since the main design task is performed solely within MATLAB/Simulink. We are looking forward to using TargetLink as the primary tool for automatic code generation in our development department.

*Matthias Haußmann
ZF Lenksysteme GmbH
Germany*

Principle layout of the steer-by-wire system.

# Mass Measurement for Outer Space Experiments

- **Solution to perform mass measurement in outer space**

- **Two measurement systems available**

- **Experiment hard- and software by dSPACE**

- **Development system based on dSPACE ACE Kit 1102**

How to perform mass measurement under weightlessness or low-gravity conditions has become a vital issue for experiments in outer space. The Department of Mechanical Engineering of Saitama University has developed a laboratory environment for conducting such experiments. The core element is the dSPACE DS1102 DSP Controller Board (part of the dSPACE ACE Kit) for measurement and controlling.

As we all know, there is no natural acceleration present in space, which is required for mass determination. Hence, a mass measurement experiment must gen-
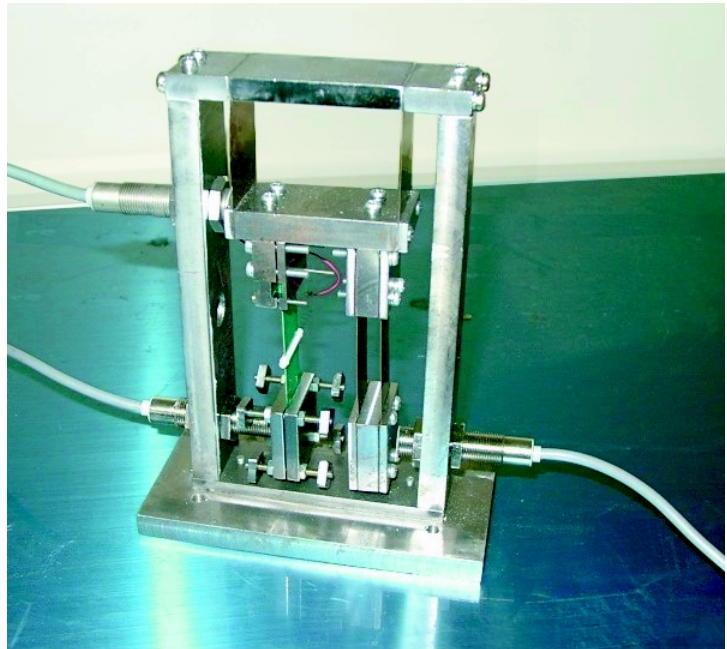
Vibration type system with bimorphic piezoelectric actuator.
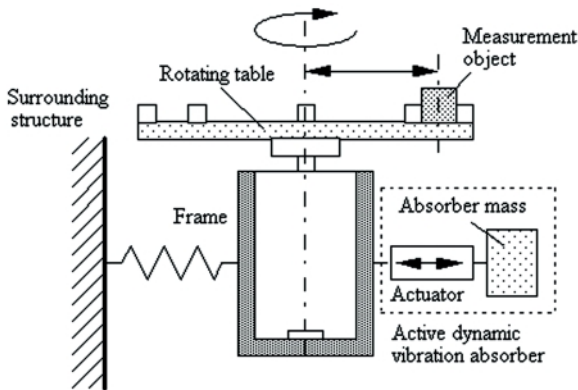
Centrifugal-type mass measurement system.

erate this acceleration to provide measurable forces, for example, by a device emitting vibrations. However, when it comes to applications in outer space, the surrounding structures are usually quite flexible, which is why any transmission must be suppressed. To cope with this challenge, the system must perform mass measurement as well as vibration control.

Two concepts
The developed system is capable of carrying out mass measure-

ment by applying two different concepts: the centrifugal and vibration type. For example, with the centrifugal type, a sample object is attached to a rotating table at a fixed distance from the longitudinal axis. Since the object unbalances the rotor, a centrifugal force with an amplitude proportional to the mass and distance is generated during rotation. This force makes the supporting frame vibrate. To restrain these vibrations, an active absorber is attached to the frame. The active absorber features a controller, which insulates the excitation force and prevents the frame from moving. Or to put it more clearly: the absorber vibrates in such a way that the product of the absorber mass and the amplitude of its vibration is equal to the amount of unbalance. This correlates with the product of the mass to be measured and its distance from the axis. Therefore, the mass of the sample object can be derived from

the vibration amplitude of the absorber mass.

However, there is a disadvantage when the sample object has a complex shape: the distance to the axis must be known exactly to achieve accurate results. One so-
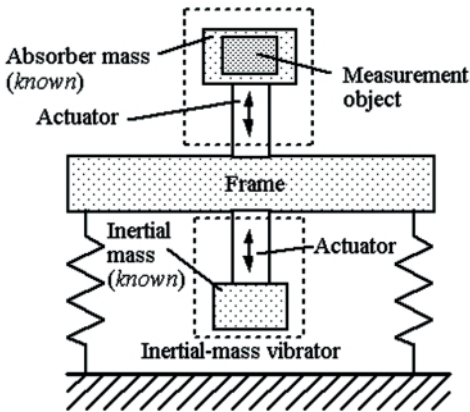
Vibration-type mass measurement system.

stressing the frame, the product of the absorber mass and the amplitude of its vibration equals the inertial mass and the related vibration. Therefore, the mass of the sample object can be determined via the vibration ratio of the absorber and inertial mass.

lution to this problem is to attach the sample object to the absorber mass. The harmonic excitation is generated by an inertial mass vibrator instead of a rotating unbalance. This vibrator consists of an inertial mass and an actuator, hence very similar to the active absorber. Since the absorber features the suppression of the force

Experiment setup with dSPACE

We implemented the corresponding control algorithms very easily on the dSPACE DS1102 DSP Controller Board. The entire programming was performed graphically within MATLAB/Simulink. The configuration of the A/D and D/A chan-

nels was equally convenient, since dSPACE Real-Time Interface (RTI) provides the corresponding Simulink blocks. Subsequently the Simulink blocks were downloaded to the hardware. For experiment control, such as data acquisition or variable display, dSPACE's ControlDesk was used, which allows continuous monitoring of control signals or excitation signals.

The accuracy of the system which can be achieved is around 0.5 percent, and without vibrations occurring on the surrounding structures.

Prof. Mizuno
Saitama University
Japan

New Catalog and Demo CD

Out now: The brand new 2001 dSPACE catalog *Solutions for Control,* providing an excellent overview of dSPACE products. It's far more than just a list of hardware and software components. The catalog offers a selection of

application examples, shows how dSPACE products are tailored to the V-cycle and introduces you to our philosophy of seamless implementation. And there's much more to discover; the list is endless. A real bonus is the attached dSPACE Demo CD. It's not simply a compilation of the catalog, but serves as a step-by-step introduction to the dSPACE software. Software demos are performed in movie sequences, and each step is accompanied by audio text.

So don't hesitate to check the box on your response card to request your copy.

Software Versions on dSPACE Product CD 3.0

All dSPACE software runs under Windows 95 / 98 / 2000 and NT 4.0.

- RTI 3.6 / 4.0
- ControlDesk 2.0
- MLIB/MTRACE 4.2
- RealMotion 1.1 (only Windows NT)

For more detailed information, please refer to *www.dspace.de*.

INFO 3 **01**

# Advanced Frontlighting System at Volkswagen

CUSTOMERS

- Direct download of Simulink models to MicroAutoBox

- Bypassing with built-in ECU interface

- ControlDesk for experiment control

About 90 percent of the information that a driver needs for safe driving is provided visually. Therefore it is obvious that perfect visibility is a crucial factor in reducing the risk of an accident. A major stepping stone towards significant improvements is the Advanced Frontlighting System (AFS): an adaptive headlamp system, which is currently under investigation by automotive OEMs and suppliers. Volkswagen has already implemented this system into prototype vehicles. This state-of-the-art technology was developed and tested using dSPACE Prototyper. VW uses MicroAutoBox for laboratory research as well as for in-vehicle experiments: a unique feature of MicroAutoBox.

Intelligence to Headlamps

Adaptive headlamp control is a key factor for enhancing visibility at night. We are therefore developing the Advanced Frontlighting System (AFS), which contains adaptive control strategies. Sensors continuously capture the parameters of the vehicle's environment such as road geometry, road type and weather conditions [1], [2], [3], [6]. This data,
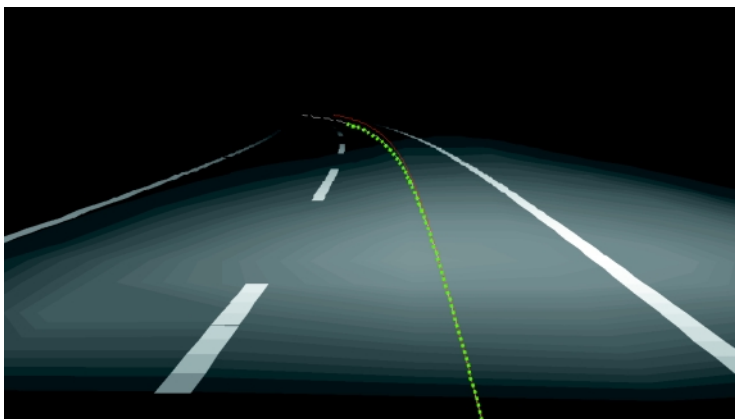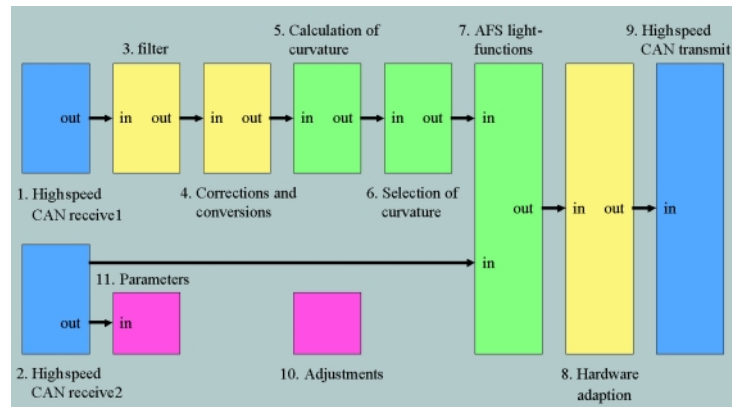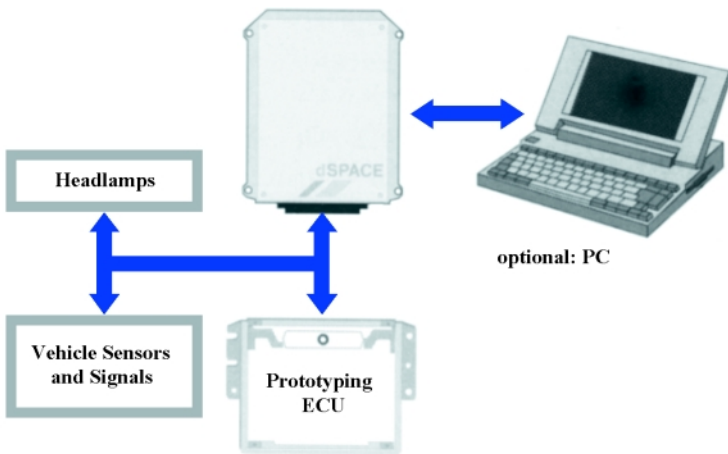

Simulink structure containing several subsystems.

mainly derived from vehicle dynamics sensors [2], [3], is available on the CAN bus. However, there is more involved than simply relating the position of the headlamp to the steering angle. For example, driving through bends requires complex control strategies to achieve the best illumination results [4]. And intelligent headlamps should emit their light distribution in a different way when driving through cities or driving on highways. Accordingly, during city driving the headlamps should illuminate a wide front area, while outside the city long-range illumination is desired. Therefore, the headlamp light distribution is constantly altered according to the implemented control algorithms (predictive, iterative, model based, etc.), which means the driver receives visual information earlier and more focused.

Complex Design, Easy Implementation

The development process of the control algorithms begins in the MATLAB/Simulink environment. The MATLAB/

Simulink design tool is widely used by Volkswagen and its associated subsidiaries and suppliers. This allows us the shared use of existing Simulink blocks and significantly speeds up the system engineering process. At this stage, the control design is entirely PC-based. Lane geometry as well as light distribution are visualized and analyzed during simulation runs with GEOSim [3] and AFSim [5] (Simulink-based tools developed by Volkswagen). The featured Simulink model is hierarchically organized, with each main block representing up to 31 subsystems. To achieve valid results, the offline simulation has to be as realistic as possible. We therefore use the same CAN data for the stimuli of the offline experiment as occurs on a real CAN bus.

To see how the algorithms work in real life, powerful real-time hardware is essential. As a result, the Simulink model is downloaded to MicroAutoBox to conduct further development steps. The code running on Micro-AutoBox is generated automatically and is a 1:1 representation of the controller model. At this stage, we are still validating the

Light distribution and road geometry [3], [5].

Overview of MicroAutoBox system integration.

tuning the parameters and validating the entire functionality. Modifications can be implemented on-the-spot, even during obligatory test drives at night. Since MicroAutoBox boots up automatically, expert know-how is not necessary to get the system up and running. Additional time-consuming and error-prone coding on the supplier prototype ECU is therefore avoided.

Time is Money

MicroAutoBox represents a real-time hardware concept which ideally suits our development environment. Since we consider it absolutely essential to have a seamless design chain, our decision to use MATLAB/Simulink along with dSPACE solutions was a matter of course. The most important benefit is a substantial reduction in design time: Engineers can evaluate their ideas at any stage of the development process.

Gabriel Schwab
Volkswagen AG
Germany

control design within the laboratory. MicroAutoBox is fed with plausible sensor signals, and with ControlDesk running on the connected PC, we control, monitor and record the experiment parameters.

Integrating MicroAutoBox into a prototype vehicle requires some additional effort: the CAN connection and the link to the headlamp hardware have to be set up. The dedicated RTI CAN Blockset to access the CAN bus within MATLAB/Simulink was an enormous help here. The AFS put into the prototype vehicle is based on a supplier system, which comes with a prototype electronic control unit (ECU) and prototype headlamp hardware. Since these elements contain only very basic features, our appreciation of MicroAutoBox's built-in bypass interface via CAN bus is understandable. The control algorithms run on MicroAutoBox and are seamlessly integrated into the hardware setup. At this point in the development stage, we focus on fine

References

[1] Bast: "Richtlinie für die Anlage von Straßen (RAS-L)" Germany

[2] Deutler, P.: "Analyse fahrdynamischer Größen als Eingangsparameter für das Advanced Frontlighting System (AFS)" Wolfsburg, 1999

[3] Fauck, J.: "Programmierung eines Simulationstools zur Darstellung der modellierten Fahrspurgeometrie unter MATLAB/Simulink" Wolfsburg, 1999

[4] Schmidt-Clausen, H.-J.; Damasky, J.: "Further improvement for motorcar headlighting systems" Institute for Lighting Technology, Darmstadt, 1996

[5] von Hoffmann, A.: "Evaluation of Advanced Frontlighting Systems with AFSim" SAE World Congress, Detroit, 2001

[6] Vukotich, A.: "Evaluation of the applicability of digital maps for the reconstruction of the real street shape" Progress in Automotive Lighting (PAL), Darmstadt, 1999
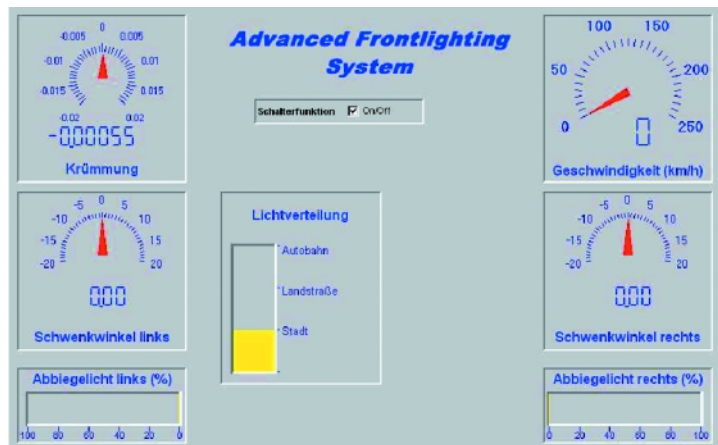
Key parameters featured by ControlDesk.

TargetLink: Positive Feedback by Mannesmann VDO

CUSTOMERS

Mannesmann VDO is a well recognized systems partner to motor-vehicle manufacturers all over the world. And Mannesmann VDO is one of the first companies who opted for TargetLink as their tool for automatic code generation. As dSPACE constantly strives for enhanced products, we highly appreciate feedback from the automotive industry. Therefore, it was a pleasure to interview Stefan Hohrein at Mannesmann VDO on his experience with TargetLink.

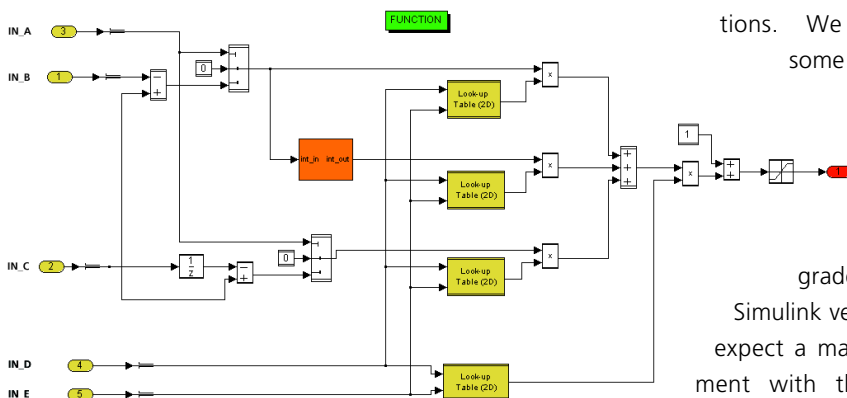*Stefan Hohrein is General Manager of the electronic design department for engine ECUs at Mannesmann VDO.*

Has TargetLink-generated code already been used in series production ECUs?
Yes, although it is a rather new tool, we're already using TargetLink code in our series products. But we still carefully check the code since it takes some time to have the same level of confidence as in our current C compiler.

Did you find any errors in the code?
We have built up several hundred bytes of code and have found no errors. The code looks handwritten, sometimes better, sometimes worse.

What do you think of the code quality in terms of documentation?
The code is well documented, but requires some effort to understand. We prefer to have only one instruction per line. This requirement is not fully supported by TargetLink.

And the tool handling?
Positive all the way. It takes minimal training to get the first usable results. TargetLink is ideally integrated into the MATLAB/Simulink environment. Even our automated tool chain is supported, comprising of C compiler, linker, and version, variant and configuration management, for example.

Did you encounter any problems caused by tool instability?
Not at all. TargetLink is very stable under our featured Windows NT system. Error messages were easy to understand and helpful. We encountered no crashes; however, we did not investigate TargetLink's performance on other platforms.

What would you like to see improved?
The cooperation with Stateflow must be enhanced since it's hard to understand when TargetLink uses FOR or WHILE instructions. We encountered some incompatibility problems as our systems department has upgraded to the latest Simulink version. Here we expect a massive improvement with the release of TargetLink1.2.

So, Mannesmann VDO continues to rely on TargetLink?
Most definitely. Thanks to TargetLink, software engineers can focus on system modeling and designing, rather than doing manual coding. We will certainly be using TargetLink more and more.

Sample controller consisting of TargetLink blocks.

---

dSPACE Training

For more details, please visit *www.dspace.de* or check the corresponding field on your response card. Further dates available on request.

dSPACE Systems

Paderborn, Germany:
- February 20/21, 2001
- May 8/9, 2001

Detroit, USA:
- March 15/16, 2001

ControlDesk Basics

Paderborn, Germany:
- March 13, 2001
- May 15, 2001

ControlDesk Advanced

Paderborn, Germany:
- March 13/14, 2001
- May 15/16, 2001

Test Automation

Paderborn, Germany:
- March 14/15, 2001
- May 16/17, 2001

TargetLink

Paderborn, Germany:
- January 25, 2001
- March 8, 2001
- April 26, 2001
- May 31, 2001

Detroit, USA:
- March 13, 2001
- March 14, 2001

HIL Simulation

Paderborn, Germany:
- March 27/28, 2001
- June 26/27, 2001

INFO 3 02

Don't Miss the Boat: TargetLink 1.2 Just Released

Automatic production code generation has been warmly received by the automotive industry, and TargetLink has become a fixture in many development departments all around the world. To make sure you get maximum benefit from its new features, simply upgrade to TargetLink version 1.2.

What's New?
TargetLink now comes with an enhanced block library comprising discrete time integrator and flip-flop blocks, for instance.

And TargetLink fully supports all the new functions available in Stateflow 3.0/4.0, such as subcharts/supertransitions, temporal logic and graphical functions – once more proving how quickly dSPACE responds to new versions from The Mathworks. TargetLink can be used almost universally with virtually all processors, since it generates ANSI-C compliant code. But that's not all: To harness additional, processor-specific functions, Target-Link's developers have created the TargetLink Optimization Modules. These optional software

modules can boost the computing power of the processors used. This is achieved by using processor optimized code patterns. Using an optimizer like this can significantly reduce the execution time and/or code size needed for a program. For TargetLink 1.2, the available optimization modules were extended to accommodate the Motorola HC12 and the Mitsubishi M32R processors. Another optional module provides superior simulation capabilities. The TargetLink Simulation Module lets you test the code directly on an evaluation board. Thus the execution time and memory consumption can already be determined accurately during the development phase.

Choose Your Processor
As regards to the TargetLink Simulation Module, the list of supported processors is constantly being added to. For release 1.2, the module can additionally be used with Mitsubishi M32R and Texas Instruments TMS470.

What's around the corner?
Further releases of TargetLink will

certainly follow the requirements of the automotive market. So in the long–term, the dSPACE development team has again aimed high, with objectives like these:

- Code generation for OSEK-compliant real-time operating systems
- Support of multirate systems
- Structured datatypes and bit addressing
- Worst-case autoscaling
- Interface to configuration management tools

On Target
It couldn´t be simpler to get your personal TargetLink copy: just contact your local office (see back page) to place your order. From countries not listed, please contact our headquarters in Paderborn, Germany.
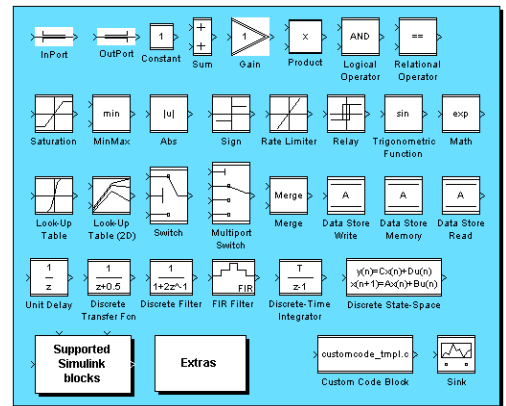
Excerpt of the enhanced TargetLink 1.2 block library.

Hardware Spin-off

On a regular basis we update our office hardware, and for this occasion we hold a dSPACE internal auction. Obsolete equipment such as PCs, monitors or disk drives are sold to dSPACE employees, and in turn this money is invested in a good cause – where it's urgently needed. This time round, all money raised was donated to a children's care project

to set up a crèche in Cochin / India. This project is dedicated to the poor and needy, helping to erect new facilities. The picture shows the almost finished new home, for example. Making the most of it: a saying brought to life at dSPACE.

House „Paderborn" - named after the dSPACE location in Germany.

Bypassing with NBD Bus at DENSO

- ◢ **Access to internal RAM without altering the ECU code**

- ◢ **Graphical programming in MATLAB/Simulink**

- ◢ **Software support with RTI blockset**

Manji Suzuki is Senior Manager in the Corporate Research and Development Center at DENSO CORP.

The growing need for high-performance electronic control units (ECU) increases the demand for powerful 32-bit microcontrollers containing RISC architectures. This refers especially to ECUs for engine management where control functions are becoming more and more complex. Since an integral element of modern 32-bit microcontrollers is built-in debug hardware, such as the Non Break Debug bus (NBD bus), DENSO successfully incorporates these interfaces into their dSPACE development environment.

Historical Background

In 1994, Yokogawa Digital Computer, YDC, launched a built-in debug interface for microcontrollers. This technology has evolved over the years and is likely to become standard in the future. There are already a number of manufacturers that equip their controllers for automotive applications with NBD bus. Several tool suppliers jumped on board making debugging with NBD bus an established approach.

NBD - An Integral Component

Engine management control software implemented on modern ECUs is constantly evolving. This is due to, for example, the demand for more extensive control and diagnosis functions. This trend clearly points out the need for powerful 32-bit microcontrollers based on RISC architectures. For debug purposes, these controllers often accommodate corresponding debug units. NBD basically allows the communication with an external tool, without any dependency on the current target microcontroller. Since NBD is onboard the controller, it is possible to monitor and change RAM data without stopping the CPU. For this reason, we evaluated the dSPACE customer solution: the NBD bus interface. This interface provides access to dSPACE Prototyper and links it to the controller. The question is: Is this approach applicable for performing function bypassing?

Test Arrangement

To realize bypassing, the computation of extracted functions, including the transfer time for I/O data, must be lower than the idle time of the ECU. Therefore, we set up a test arrangement to determine if the dSPACE system can meet this requirement. We used dSPACE Prototyper along with the NBD bus interface. The experiment software we used for testing was dSPACE's ControlDesk. The implementation of Simulink blocks was performed with Real-Time Interface (RTI). We incorporated several control functions on

dSPACE Prototyper and encountered no problems whatsoever due to the lack of communication speed. The ability to access internal RAM without changing the ECU code is a great advantage that we highly appreciate. Thanks to the NBD approach, we can now use the same setup for any type of microcontroller with built-in NBD interface.

> **A customer solution becomes a standard product**
> The dSPACE NBD bus interface has now become a standard solution, available off the shelf. Software support via Real-Time Interface (RTI) can be delivered for various types of microcontrollers.

Conclusion

To answer the question on applicability: Yes, we will certainly adopt the dSPACE approach, since it keeps us independent from the target platform. We do not need to change our development environment to perform bypassing since the NBD bus is integrated into new releases of 32-bit controllers. We intend to complete our design chain with TargetLink, which generates the production type C code for the ECU. We are confident that a development environment built around dSPACE tools is the solution to radically reduce our development time.

Manji Suzuki,
DENSO CORPORATION
Japan

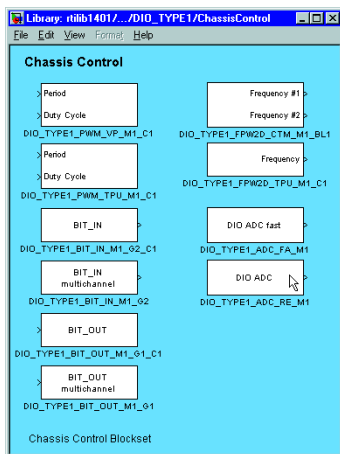*Benchmarking communication speed at DENSO: Takafumi Inaba and Yasuhiro Inaba.*

RTI for MathWorks Release 11.1 and 12

Our Real-Time Interface (RTI) software has been the "right address" for convenient controller prototyping and hardware-in-the-loop simulation for many years. Thanks to RTI's unrivaled scope of functionality, it is possible to realize MATLAB/Simulink/Stateflow models on dSPACE systems and connect them to the real world via I/O. New RTI versions are now available for users of MathWorks Release 11.1 as well as Release 12.

If you are already well acquainted with dSPACE, you know you can rely on dSPACE to deliver an up-

dated version of RTI shortly after a new MATLAB release. Thanks to the close cooperation between The MathWorks and dSPACE, the most up-to-date development tools are always available for use in your projects.

The RTI block libraries for Simulink let you easily connect your models to the I/O of dSPACE systems graphically and via drag & drop. Having connected your model to the real world, you can carry out meaningful tests immediately. The model itself is simulated in real time on the powerful dSPACE hardware. Two new versions have now been released to support users of both MathWorks Release 11.1 and Release 12.

RTI 3.6 for Release 11.1...

RTI 3.6 now offers new serial interfaces and updated ECU interfaces. For MicroAutoBox there is a new Simulink blockset for vehicle dynamics applications. Improved task handling guarantees convenient setting of priorities or overrun behavior for each task individually. RTI now provides even better handling to users of multiprocessor systems: the definition of the multiprocessor topology and the model partition-

ing have been further improved. It is now possible to represent variables that are calculated on different processors in one ControlDesk plot.

...and RTI 4.0 for Release 12

RTI 4.0 offers the same improvements and changes as in RTI 3.6, however, at the same time it is fully compatible with Release 12 from The MathWorks. This new release lets you use all of the new features of MATLAB/Simulink/Stateflow in your dSPACE system: improved model handling, search functions, new signal properties as well as improved code generation.
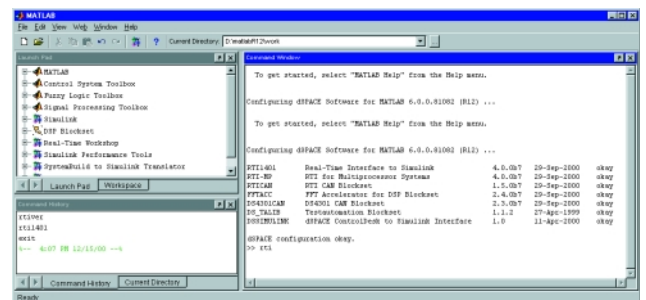
Release 12: New desktop front end showing loaded dSPACE software.

Both RTI versions are available in January on the new dSPACE Product CD 3.0.

MicroAutoBox: Special Simulink blocks for chassis control now available.

New dSPACE Software on Product CD

With the release of the dSPACE Product CD 3.0, we now ship the brand new dSPACE software versions. Once again proof of a strict and accurate development process. We precisely met our goal even with ambitious and tight schedules. The Product CD 3.0 has just been launched and is now available.

Besides the new Real-Time Interface 3.6 / 4.0, the new dSPACE Product CD contains the very latest version of ControlDesk. ControlDesk 2.0 now comes with an integrated Simulink interface, allowing to control, monitor and automate your experiment even during Simulink simulations. And there's much more to discover:

automotive interfaces, operator version and massively improved processing speed – our software engineers have done a good job. For more detailed information, please refer to www.dspace.de
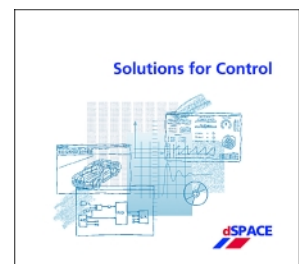
Automated Software Verification at TEMIC

- ◢ **Test automation with ControlDesk 2.0**

- ◢ **Central database for single tests or test sequences**

- ◢ **Automatic genera- tion of standardized test documentation**

The complexity of function- alities offered by automotive systems is growing at an enor- mous speed. This makes high demands on software com- plexity, and coupled with shorter and shorter innovation cycles, it also means that new development approaches are needed. Temic faced this chal- lenge very early on and has now, in cooperation with dSPACE, developed an auto- mated test environment for software verification. This had its debut in the electronic con- trol unit (ECU) designed to get active suspension on board the Mercedes CL Coupé.
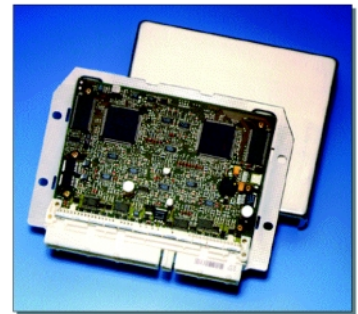
Going Automatic
The use of rapid prototyping tools in functional development, along with procedural method models for software creation, have become today's new stan- dard. Automated software devel- opment processes – from model- based descriptions to the use of automatic code generators and right through to automated software verification – are firmly on the agenda for software devel- opment in the future.
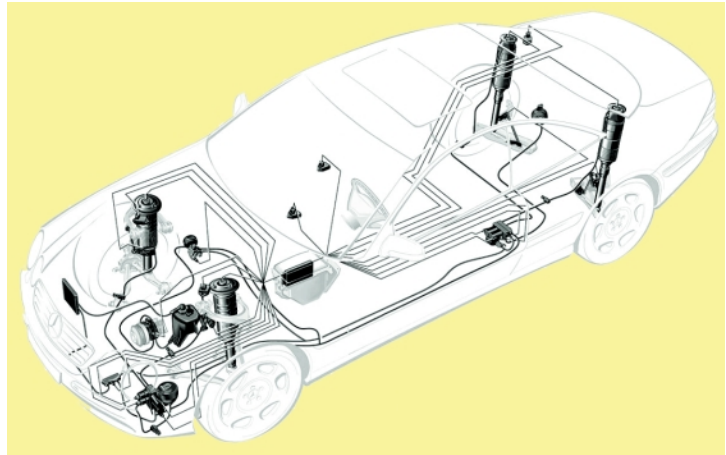
System arrangement of the ABC components.

Actual Project
TEMIC developed and produced the ECU for a new, actively con- trolled suspension, which was first used in production in the new Mercedes CL Coupé . This system, known as Active Body Control or ABC, enhances both driving safety and ride comfort. Driven by the software, electronically controlled hydraulic cylinders in the suspen- sion system adjust the spring and damping properties of the vehicle according to road surface, driving style and the driver's preferences. The ABC system has enormous re- quirements in terms of processing power and real-time control be- havior. This in turn creates the need for an appropriate test environ- ment as described in this article.

From Rapid Prototyping Model to Fully Functional ECU
The functional requirements im- posed on the ECU were defined using rapid prototyping, by cre- ating a MATLAB/ Simulink model. The task then facing production de- velopment was to implement the functionality on the target sys- tem – the ECU itself. When

ECU for ABC on board the Mercedes CL Coupé.

model-based control algorithms are implemented, particular at- tention needs to be paid to veri- fication: it has to be demon- strated that the software imple- mentation completely matches the model in terms of functions.

Classic test procedures are domi- nated by manual execution and have considerable drawbacks when it comes to dynamic functionalities:

- ◢ Large amount of time needed due to the high number of in- dividual test cases
- ◢ Unsatisfactory repro- ducibility of dynamic test patterns
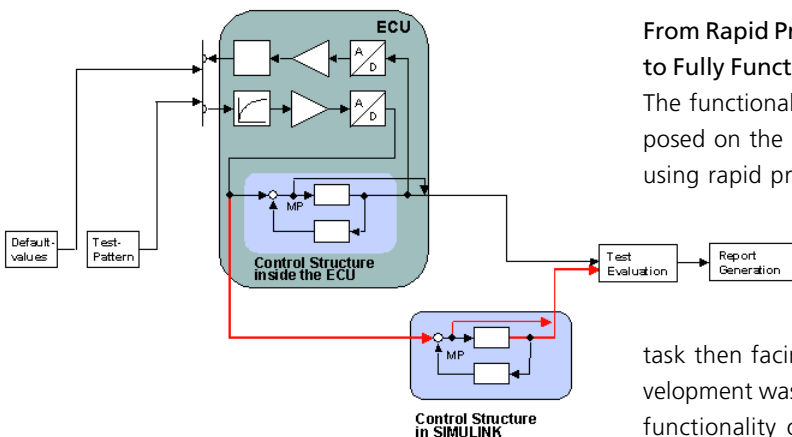- ◢ The documentation has to be produced by hand

Diagram of a test run.

In drawing up the requirements for the next generation of tools for software tests, we were essentially guided by the need to eliminate these disadvantages. The individual objectives were as follows:

- Verification of correct implementation of control and function specifications in the software for the target system (ECU)
- Short-term repeatability of test cases with full integration into the concurrent change management
- Complete, standardized test documentation
- Reduction in time and costs by means of completely automated test runs

To achieve these objectives, we opted for close cooperation with dSPACE, using ControlDesk 2.0. The completely new kinds of options offered by this software, particularly its transparent connection to MATLAB/Simulink, and the professional engineering support that dSPACE staff gave us in adjusting ControlDesk to our needs, were major factors in the success of the pilot project.

Papers

By dSPACE
K. Lamberg, P. Wältermann: "Using HIL Simulation to Test Mechatronic Components in Automotive Engineering"

English 3 **03**
German 3 **04**

T. Pöhlmann: "The Internal Combustion Vehicle with the Lowest Emissions"

English 3 **05**
German 3 **06**

Test Environment
The heart of the automated test environment is a fully automated comparison of the functionality of the MATLAB/Simulink model with its implementation in the ECU. The basic idea is to apply the same stimulus signals to the Simulink model as to the implemented ECU software (see diagram on p. 12). The evaluation compares the behavior patterns of the implementation in the ECU to those of the reference model according to mathematical, computational criteria, and rates the results as pass or fail.

To generate suitable test patterns, we use ControlDesk's Stimulus Editor. This allows stimulus patterns for the planned tests to be defined freely. Parameterizable signal patterns, graphically editable signal patterns and signal behaviors gathered from test drives in the vehicle are all options that can be used for test case generation.

The ECU's electronic connection to the test environment is controlled centrally by ControlDesk. Control and communication are via parameterizable inputs and outputs, CAN bus connection and data bus based on the dSPACE ISO9141 Python module.

At the start of each test, the ECU is put into a defined initial state by means of stimuli, before the actual test stimuli are fed to it. Both the internal computing results from the initialization phase in the ECU and the test case that immediately follows are transferred and recorded to the test environment via the data bus. The next step is to start the Simulink model directly from the

test environment and feed in the signals transferred by the ECU, all under the control of ControlDesk.

The documentation including the graphical results is produced in a standardized Microsoft Word file after each test step. A database administered under ControlDesk is used as the central data basis for all system files, initialization and test stimuli, and test results.

Test Database
For the intended reductions in time and costs, entire test sequences and even complete release tests have to be run completely automatically. dSPACE specially designed a sequence editor for this task. This allows any desired individual tests to be grouped together to produce complex test sequences.

All of the individual tests and the test sequences are placed in the central database. They can subsequently be performed and reproduced as often as required.

Success Story
The resulting enhancement of test quality and reduction in time and cost sets new standards in software verification. This is largely made possible by ControlDesk and its new test automation features. So the project definitely marks the start of the next step in software development.

Udo Teubert
Reinhard Helldörfer
TEMIC
Germany
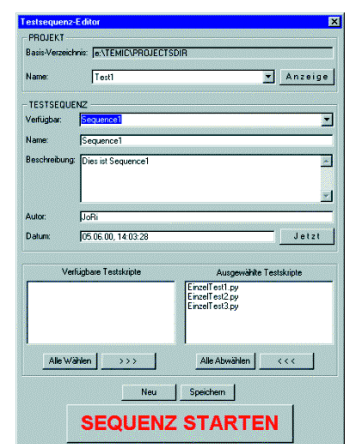
*The sequence editor's graphical interface.*

Total Decoupling with an Adaptive Interface

- ◤ **Adaptive interface for neutralization of structural vibrations**

- ◤ **Controller running on dSPACE Prototyper**

- ◤ **RTI and ControlDesk for implementation and experiment**

In the Department of Adaptronics at the Otto-von-Guericke University in Magdeburg, Germany, an adaptive interface is being developed to decouple two mechanical systems. This interface effectively cancels out vibrations propagating from one mechanical system to another. The model of the mechanical structure as well as an adaptive control algorithm is implemented within the MATLAB/Simulink environment. For real-time simulation, the control algorithm, defined as Simulink blocks, runs on dSPACE Prototyper, comprising a multiprocessor system. The developed adaptive interface has been verified and proved to work fine. As a result, series production levels for various applications have been reached.

The adaptive interface is designed to eliminate vibrations moving from one mechanical system to another via a mechanical joint. The scope provided by the adaptive interface is fairly extensive. For example, if aircraft floors
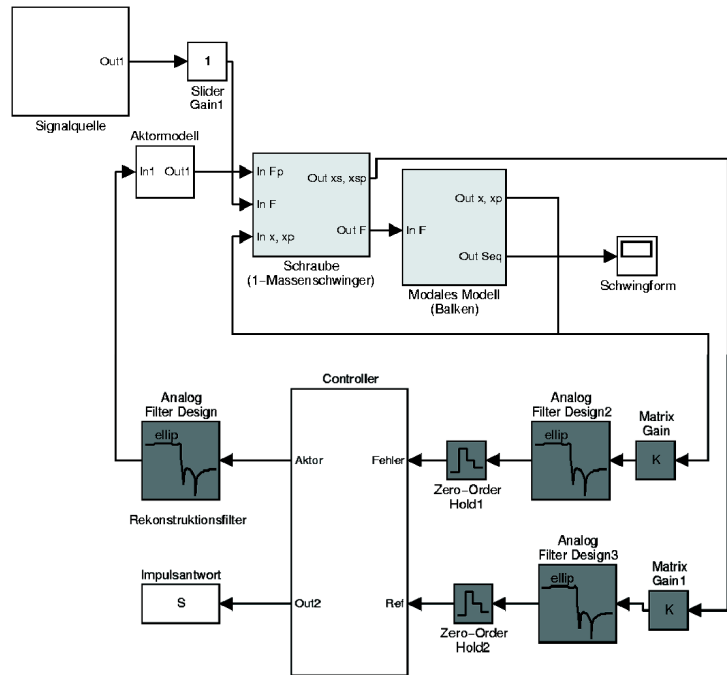

Simulation of the adaptive system with Simulink.

need to be mechanically isolated or sensitive electronic equipment in spacecraft needs to be protected, the adaptive interface cancels out the occurring vibrations and therefore massively reduces the impact via the mechanical connection.

Mechanical Setup
The basic principle is to apply forces at the linkage of the coupled structures. An adaptive interface based on piezoceramic stack actuators is applied, which can transmit static forces while isolating dynamic disturbances. For the experiment setup, the interface is mounted on a bi-clamped beam and driven by a digital controller in such a way that no dynamically disturbing forces will affect the beam. Therefore, the actuators counteract
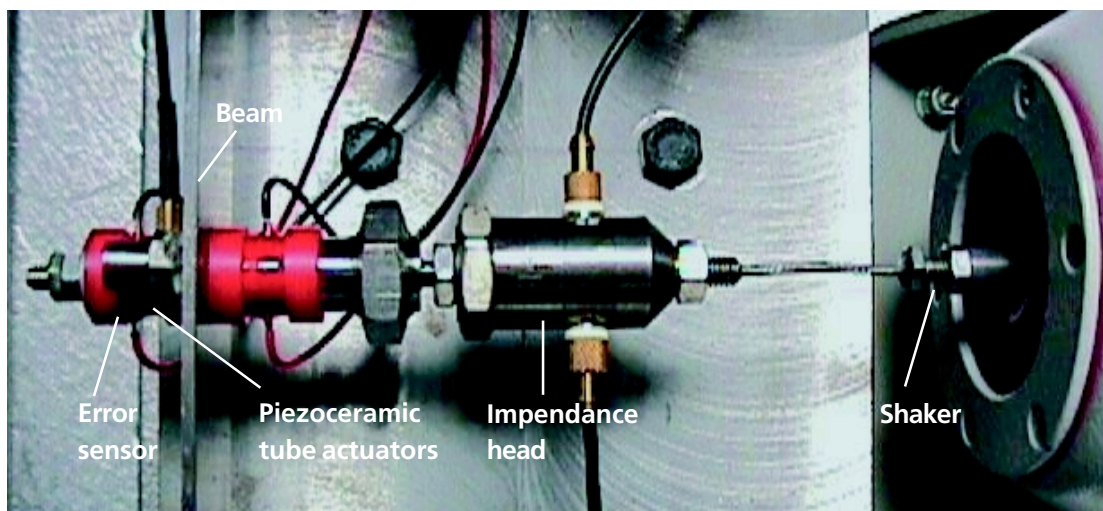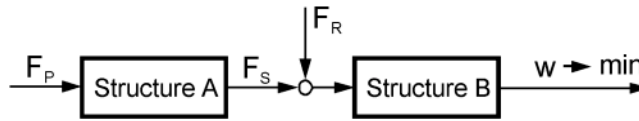

Experimental setup of the adaptive interface.

each other: one actuator is with-drawn while the other is ex-tended.

Principle of the decoupling of structural vibrations.

Development Process

Initially, the entire development takes place within the MATLAB/Simulink environment. This means, a model of the interface's mechanical structure is pro-grammed graphically. Actuators and sensors are then added to the model, along with an adap-tive feedforward-controller (con-sisting of C-coded S-function blocks performing special tasks), to create an entirely software-based experiment. The required reference signal is taken from structure A, since this part vi-brates even in perfectly decoupled structures. The error sensor is attached to structure B. In a next step, the adaptive inter-face within MATLAB/Simulink is replaced by real components with a mechanical setup as intro-duced above. This is where dSPACE Protoytper comes into play. The adaptive control algo-

by a gradient filtered-X-LMS algo-rithm), is implemented on the dSPACE real-time hardware fea-turing a sample time of 0.25 ms. The link to the dSPACE hardware is performed with dSPACE's Real-Time Interface (RTI), which con-tains all required blocks for ad-dressing the hardware graphi-cally. All Simulink blocks are coded automatically and loaded onto dSPACE Prototyper. To con-trol and monitor the real-time ex-periment, dSPACE ControlDesk is used. During the experiment processing, the ability to cancel out vibrations in the frequency range from 30 Hz to 800 Hz is analyzed.

Conclusion

The adaptive interface is a prom-ising approach to eliminate vi-bration propagation from one

bandwidth and provides good reduction of the beam vibration. dSPACE Protoytper proved to be the perfect solution for perform-ing real-time experiments with-
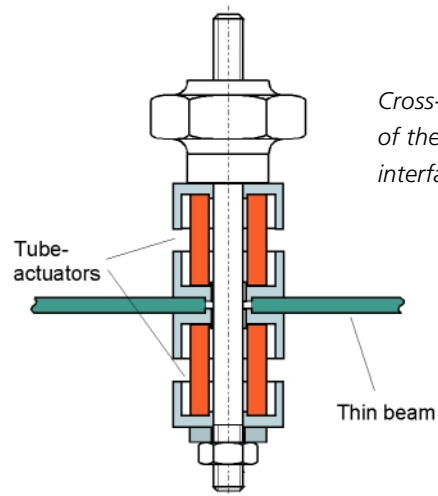
Cross-section of the adaptive interface.

out any implementation over-head. The adaptive control algo-rithm could be analyzed on the spot, and evaluation of a new design takes just a few seconds.

Dirk Mayer
Sven Herold
Otto-von-Guericke-Universität
Magdeburg
Germany
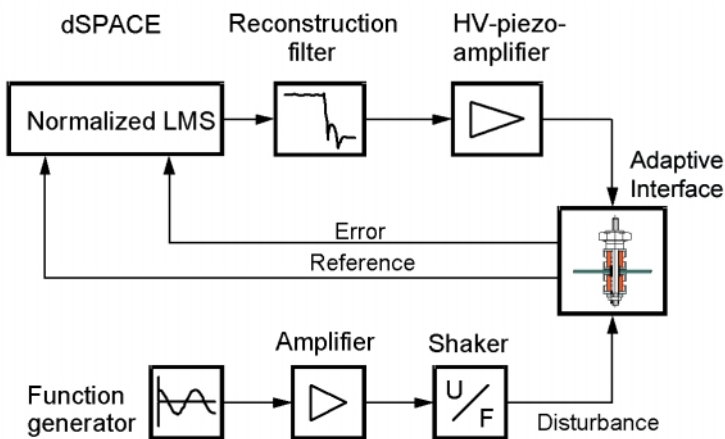
Layout of the adaptive system.

rithm, along with the digital pro-cessing (e.g., FIR filter with order 400 and coefficient adjustment

mechanical system to another. The implemented control algo-rithm covers the entire frequency

Events

Europe

**EUROFORUM-Konferenz
Elektronik-Systeme im Automobil**
February 13–14
Munich, Germany

Embedded Systems
February 14–16
Nuremberg, Germany
Messezentrum Nürnberg, Hall 12
Booth #M10

**FKFS Stuttgarter Symposium
Kraftfahrwesen und
Verbrennungsmotoren**
February 20–22
Stuttgart, Germany

VDI Getriebe in Fahrzeugen
June 19-20
Friedrichshafen, Germany

PCIM
June 19–21
Nuremberg, Germany
Messezentrum Nürnberg, Hall 12
Booth #310a

USA

SAE
March 05–08
Detroit, MI
Cobo Hall
Booth #2801

Embedded Systems Conference
April 09–13
San Francisco, CA
Moscone Center
Booth #2237

Real-Time Computer Show
May 03
Greenbelt, MD

ACC
June 25–27
Arlington, VA

Representatives

Australia
CEANET Pty Ltd.
Level 4, 2 Bligh Street
Sydney NSW 2000
P.O. Box R202
Royal Exchange NSW 1225
Tel.: ++61 2 9232 3699
Fax: ++61 2 9232 3332
info@ceanet.com.au
www.ceanet.com.au

France
The MathWorks France
20 rue Troyon
92316 Sevres Cedex
Tel.: ++33 1 41 14 67 14
Fax: ++33 1 41 14 67 15
info@mathworks.fr
www.mathworks.fr

Israel
Omikron Delta (1927) Ltd.
10 Carlebach St.
Tel-Aviv 67132
Tel.: ++972 03 561 5151
Fax: ++972 03 561 2962
info@omikron.co.il
www.omikron.co.il

Korea
Darim System Co. Ltd.
Expo VentureTown
DoRyong-Dong 3-1
YuSung-Gu
Taejon
Tel.: ++82 431 260 4747
Fax: ++82 431 260 4748
ksh@mail.darimct.co.kr

Poland
ONT
ul. Obozna 11
30-011 Kraków
Tel.: ++48 12 636 32 60
Fax: ++48 12 632 17 80
info@ont.com.pl
www.ont.com.pl

Taiwan
Scientific Formosa Corporation
11th Fl. 354-1 Fu-Shin N. Road
Taipei, Taiwan, R.O.C.
Tel.: ++886 2 2505 05 25
Fax: ++886 2 2503 16 80
info@sciformosa.com.tw
www.sciformosa.com.tw

Czech and Slovak Republic
HUMUSOFT s.r.o.
Novákových 6
180 000 Praha 8
Tel./Fax: ++42 2 684 41 74
info@humusoft.cz
www.humusoft.cz

India
Cranes Software Intern. Ltd.
29, 7th Cross, 14th Main
Vasanthanagar
Bangalore - 560 052
Tel.: ++91 80 2250 260
Fax: ++91 80 2250 413
dspace@cranessoftware.com
www.cranessoftware.com

Japan
LinX Corporation
N-E-S Building, N-7th floor
22-14 Sakuragaoka-cho
Shibuya-ku, Tokyo 150-0031
Tel.: ++81 03 5489 38 71
Fax: ++81 03 5489 38 72
info@linxworld.co.jp
www.linxworld.co.jp

Netherlands
TSS Consultancy
Rietkraag 37
NL-3121 TC Schiedam
Tel.: ++31 10 2 47 00 31
Fax: ++31 10 2 47 00 32
info@tsscon.nl
www.tsscon.nl

Sweden
FENGCO
Hallonbergsplan 10, Kompassen
SE-174 07 Sundbyberg
Tel.: ++46 8 6 28 03 15
Fax: ++46 8 96 73 95
sales@fengco.se
www.fengco.se

United Kingdom
The MathWorks Ltd.
Matrix House
Cowley Park
Cambridge CB4 0HH
Tel.: ++44 1 223 423 200
Fax: ++44 1 223 423 255
info@mathworks.co.uk
www.mathworks.co.uk

dSPACE