

dSPACE

NEWS

FACTS · PROJECTS · EVENTS

Products

AutomationDesk:
The Next Generation
of Testing

Customers

TargetLink Supports
Magneti Marelli's
Design Methods

BYPASSING



Audi Applies Service-Based Bypassing

Editorial

- 3** By Dr. Herbert Hanselmann
President and CEO

Customers

- 4** Audi Applies Service-Based
Bypassing
- 6** Utilization of TargetLink for Delphi
Control Applications
- 8** Magneti Marelli's Design of a
GDI Engine Controller
- 10** Honda: Testing Automatic
Transmission ECUs

Products

- 12** Enhanced RTI CAN Blockset
- 12** Update and LIN Node Simulation
- 13** AutomationDesk: The Next
Generation of Testing

Business

- 14** Japanese User Conference
- 15** Info and Dates

dSPACE NEWS

dSPACE NEWS is published periodically by:

dSPACE GmbH · Technologiepark 25 · D-33100 Paderborn
Tel.: +49 52 51 1638-0 · Fax: +49 52 51 66529
info@dspace.de · support@dspace.de
www.dspace.de

Editor: Bettina Henking

Layout: Beate Eckert, Ute Bergmann

dSPACE Contributors:

Robert Bevington, Günther Gruhn, Louise Hackett,
Herbert Hanselmann, Ralf Lieberwirth, Thomas Pöhlmann,
Gerhard Reiß, Christine Smith

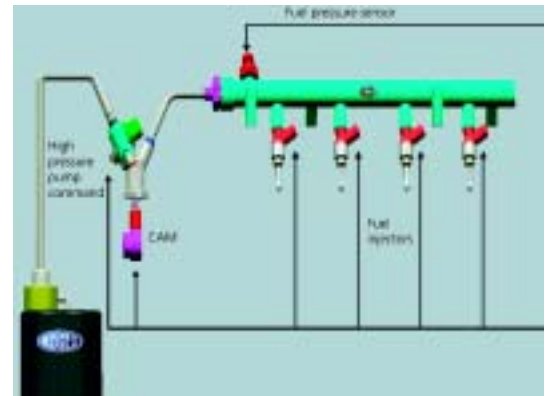
© Copyright 2002

All rights reserved. Reproduction of all or parts of this
publication is permitted provided the source is stated.

Products and brand names are trademarks or registered
trademarks of their respective holders.



4 Audi applies service-based bypassing for the development of new fuel pressure control functions using dSPACE Prototyper and special dSPACE bypass routines.



8 TargetLink supports the model-based design methodology at Magneti Marelli, which drastically decreased the time-to-market of a new powertrain controller of the gasoline direct injection project.



I just returned from our German User Conference 2002 in Stuttgart, feeling highly delighted with the response. There'll be a detailed report on it in the next issue, but I would nevertheless like to give you my first impressions.

More than 150 conference participants came to discuss their views and experience with other users, product experts, and engineers from a variety of companies. 15 customers presented papers giving interesting insights into their development work. I'm sure that many of you would also be interested in what they had to say, so we are including as much as possible of it in this and the next few issues of dSPACE NEWS.

One of the topics of the conference was bypassing. This is an approach that allows new electronic control unit (ECU) functions to be executed on the prototyping system, while existing code runs on the existing ECU. We have supported this method with rapid prototyping systems for several years now. Suppliers use it for quick development, and OEMs

have a growing need to develop new functions themselves. Bypassing is ideal for that. Our tools enable them to concentrate on new functions hassle-free. In this issue you can read how Audi is using the bypassing approach to develop a new fuel pressure control.

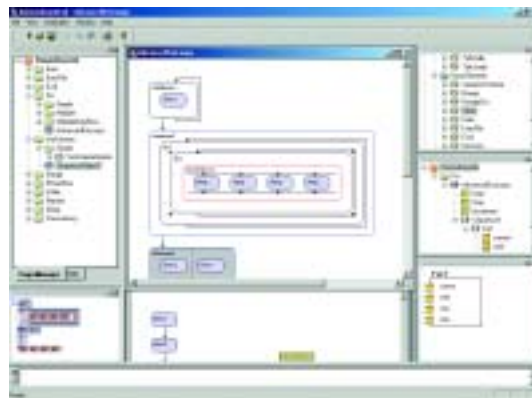
We also took the opportunity to show the conference participants our innovations and further developments. Version 2.0 of TargetLink, our production code generator, supports OSEK/VDX-compliant operating systems and provides the new dSPACE Data Dictionary for central model data management. AutomationDesk is the solution for automated testing of ECUs and test project management. This is also described in detail in this issue. The dSPACE Calibration System for calibration and measurement will complete the dSPACE tool chain in the near future.

We ourselves always learn a lot from our User Conferences. This time our customers again confirmed how important seamlessly integrated tools and optimized development processes are to them, as they endeavor to avoid expensive mistakes and cut development times.

*Dr. Herbert Hanselmann
President and CEO*



10 *Honda has developed a new direct control 5-speed automatic transmission system. Using dSPACE Simulator dramatically reduced the overall development time.*



13 *AutomationDesk, another important enhancement to the dSPACE range of tools, will reduce development time by facilitating the management of large test projects.*

Audi Applies Service-Based Bypassing

➤ dSPACE Prototyper for Audi's fuel pressure control design

➤ New, flexible dSPACE bypass concept

➤ Minimized "footprint" in the target code

For the development of new fuel pressure control functions, Audi decided to apply the bypassing method using dSPACE Prototyper and special dSPACE routines for bypassing.

Today's engine electronic control units (ECUs) are the product of the know-how amassed by original equipment manufacturers (OEMs) and suppliers over many years. A typical ECU only gives way to a new generation after 5 to 8 years and is supplied to several OEMs. This is reflected in the software by the relatively small customer-specific part when compared to the platform content. At Audi, however, the proliferation of niche models and the introduction of new engine technology is increasing the demand for custom content in engine ECUs. Unfortunately, unwieldy software architectures along with cost considerations are forcing suppliers to neglect developments that cannot easily be integrated into their software platform. Consequently, Audi must be in a position to design and verify its own custom content without having to rely on the ECU supplier.

First Pilot Project in 2001

A pilot project using dSPACE Prototyper together with a Bosch ME7 ECU was completed successfully



Audi uses dSPACE Prototyper and bypass services for fuel pressure control development.

in 2001. For future rapid prototyping projects, it was agreed that the tailored solutions applied to the pilot project needed to mature into generic and reusable solutions. The first suitable project arose in close cooperation with Siemens VDO. For this, the rapid prototyping activities concentrated on the control of an electronic fuel pump unit via a Simos6 engine ECU. This was required for new direct injection gasoline engines in order to continuously vary the fuel flow to a mechanically-driven step-up pump. The rapid prototyping system used consisted of a plug-on device (POD) in the ECU, communicating via dual-ported RAM with a dSPACE MicroAutoBox. The POD was developed by dSPACE to be mounted in the FLASH socket for the Simos6 B sample ECU.

Minimized "Footprint" in the Target Code

In order to bypass the existing function in the engine ECU, it was necessary to insert special routines at the call and return points. These RCP bypass services were responsible for passing the calling parameters to the POD and overwriting the results from the original function with the returned values. The RCP bypass service at the return point had to delay the return until new values became available. For this reason, data transfer had to be performed with minimum delay, and MicroAutoBox had to execute function models at very high speeds. The RCP bypass services were provided by dSPACE as C-code to be linked into the target code at build time. These services appealed to Audi because of the efforts made by dSPACE to minimize the "footprint" they made in the target code. The services also offered a high degree of flexibility.

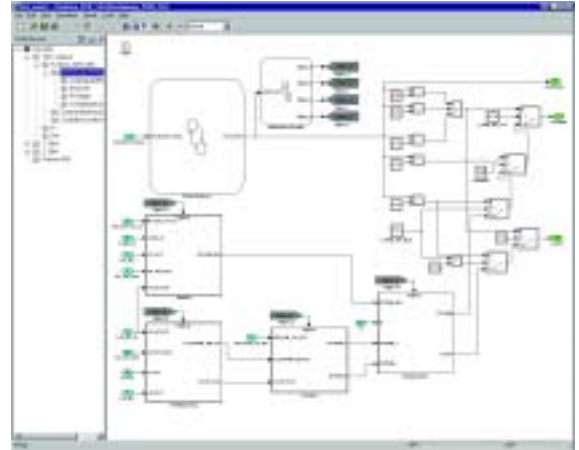
Configuring and Activating the Services

An existing function in the Simos6 formed the basis for the new algorithm and was transferred as a MATLAB®/Simulink®/Stateflow® model to MicroAutoBox. In order to bypass this function in the ECU, it was necessary to insert calls to RCP bypass services at specific points in the scheduling routines. The function consisted of two modules. The first module was always executed in the 10 ms task, the second was executed in different tasks depending upon engine operation (10 ms task or crankshaft-synchronous task). Configuring and activating the services in accordance with the particulars of each module proved to be simple. The control of the fuel pump unit relied on functionality in the unit itself, that is, the overall function could be considered as being distributed between the fuel pump unit and the engine ECU. At the initial stage of the project, no appropriate fuel pump units were available, and it was necessary to implement the missing functionality in MicroAutoBox. Although both parts of the distributed function resided in MicroAutoBox, there was no direct connection between them. The communication took place via ECU hardware outputs.

Successfully Completed in Only Six Weeks

Various aspects of the control algorithm needed to be clarified, in particular the necessity for the fuel pump to be switched on prior to the ignition in order to guarantee enough pressure at engine start. Within a very short time it was possible to improve the governor and to gain valuable insight into the behavior of the fuel system before and during engine starting. The availability of a rapid prototyping system at such an early stage of the development supported the decision-making process by providing hard facts. As a result it was possible to avoid design errors in the ECU software. The integration of dSPACE bypassing services into the target posed no notable problems. The development of the POD was the greatest challenge. This was completed within six weeks thanks to an open working relationship between the ECU hardware developer and the dSPACE developer. The only significant improvement that will be made for future rapid prototyping applications concerns the behavior of the RCP system during initialization and power-down phases.

In future, the RCP system must be entirely unaffected by ignition cycling. Despite boasting a Motorola PPC603e processor running at 300 MHz and the high-speed communication interface to the ECU, the turnaround time for functions executed in MicroAutoBox was only marginally less than the execution time for a manually coded function in a target running at 60 MHz. This is partly due to the use of Audi's own Simulink library, which has been especially set up to provide modeling style guides rather than taking into account code efficiency during rapid control prototyping. It is also testament to the overheads resulting from automatically generated floating-point code, compared to optimized production code on the target ECU. For this project, this posed no practical problem, but must be borne in mind for future projects.



Simulink function blocks for fuel control.

More Effective Design of New Functions

By applying a model-based approach, it is possible for Audi to apply its expertise in engines and systems more effectively to the design of new functions. RCP hardware is considered most suitable for the experimental stage; the footprint in the target software is modest and the load placed on the target's processor is negligible. Once the algorithm has been optimized, translating this into source code for embedded code – preferably using an automatic code generator such as TargetLink – and incorporating this into the target is highly desirable in order to confirm that the target's resources are adequate. It will then be possible to equip a larger number of test vehicles with the new function for validation purposes.

*BSc. (Eng.) Adrian James, Team Leader
Dipl.-Ing. Manfred Nerb, Function Designer
Powertrain Electronics, Audi AG
Germany*

Utilization of TargetLink for Delphi Control Applications

As a part of the company's effort to increase the quality of the products and minimize its development cycle, Delphi is rapidly adopting algorithm modeling and code generation techniques. The production code generator TargetLink from dSPACE is a major component of the tool chain that Delphi has been using in a new development environment.



Dr. Lev Vitkin, Delphi Delco Electronics Systems, U.S.A.

We had the chance to interview Dr. Lev Vitkin, who is the Autocode Technology Leader of Delphi Delco Electronics Systems.

dSPACE NEWS: Lev, thank you for giving us the opportunity to ask you more about the usage of TargetLink for automotive control application development within Delphi's projects. Why is Delphi using TargetLink as a production code generator?

Lev Vitkin: Using TargetLink is demonstrably advantageous for Delphi's software developers. There are really no syntax errors, and logical errors are more quickly identified and traced to their root cause. We have found out that the speed of coding a model and testing of the generated software is greatly increased; therefore the engineers have more time to find the "optimum solution" rather than the quickest or easiest one. Model changes midstream – of which there are usually many – do not force the developer to go back to step one. Another advantage is that TargetLink's code consistency is unmatched, allowing better handoffs between several engineers working on the same project.

dSPACE NEWS: What were the criteria TargetLink had to meet in order to fulfill its role as a production code generator ?

Lev Vitkin: One of the main criteria for an autocode tool for automotive applications is the

efficiency of generated code. TargetLink did not cause any significant overhead on the project budget with regard to ROM, RAM, and throughput. Generated code featured the same quality as hand-written code for similar applications. In terms of consistency, I can state that the same type of model-to-C-code implementation is repeated, which makes work a lot easier. Also, the model can be set to generate code that is itself modular, we call that a self-contained function call, and it allows global or local access to the model variables. So TargetLink supports and delivers modularity. TargetLink is very flexible. You can easily "tune" automatically generated code to comply with Delphi software coding standards. And finally, the quality of the technical support of a new complex tool, like a code generator, is another main criterion in choosing a tool's supplier. We are very pleased with the level of technical expertise and responsiveness of the TargetLink support team.

dSPACE NEWS: We have heard that you are optimizing the usage of TargetLink with the help of a special Delphi User Guide. Could you tell us more about that?

Lev Vitkin: Yes, we have developed a set of procedures and guidelines for algorithm modeling in a MATLAB®/Simulink®/Stateflow® environment and within TargetLink code generation so that changes to the model are more readily noticed and logical or implementation errors are less likely to occur.

For our engineers we recommend: The model and its components should follow a specified Delphi structure (Fig. 2). This ensures that minimum modifications are involved, if any, in a stage of generation of highly structured and efficient code. Moreover, we advise our engineers to use Delphi's Simulink and Stateflow library blocks (Fig. 1). These blocks were created firstly, to implement in Simulink/Stateflow some reusable routines that are common in Delphi software applications, and secondly, to ensure the efficient TargetLink code generation representing these blocks. To minimize the project development cycle and promote the consistency in algorithm development, Delphi's TargetLink User Guide specifies how to apply TargetLink for Delphi applications. This guide covers topics like variable initialization, accommodation of legacy code in autocode, integration of autocode into handwritten applications, code verification procedures, and tips on setting TargetLink properties.

dSPACE NEWS: What did you personally experience as the greatest challenge of Delphi's application projects with TargetLink so far ?

Lev Vitkin: That was to incorporate legacy code and structures into autocode as well as assign specific memory allocation for variables and constants. But our engineers overcame it without any handwritten modifications of generated code by using the full power of TargetLink's flexibility in code generation.

Finally, the usage of a code generator requires strong discipline to follow the process of building the software application. So, we have to be sure that our Common Systems and Software Development Process properly accommodates new autocode technology, and our engineers thoroughly follow that process.

dSPACE NEWS: Thanks a lot for your time, Lev!

dSPACE NEWS: Which application did you apply TargetLink to?

Lev Vitkin: We used TargetLink for several Motorola 16-bit and 32-bit processor based applications, like EGR Control, Rollover Detection, and Adaptive Cruise Control. These algorithms were modeled with a mixture of Simulink blocks and Stateflow charts.

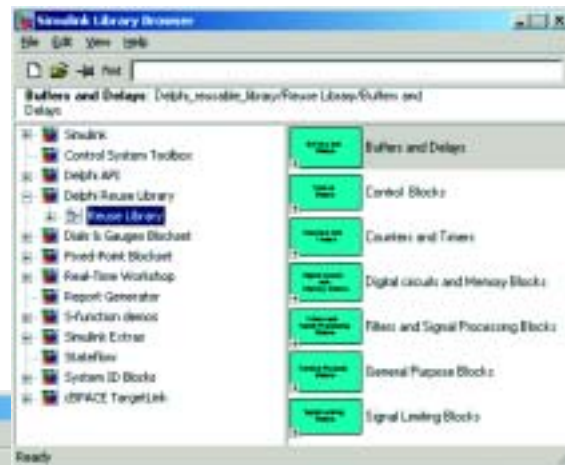


Fig. 1: Delphi's reuse library – created, among other things, for efficient use of TargetLink code generation.

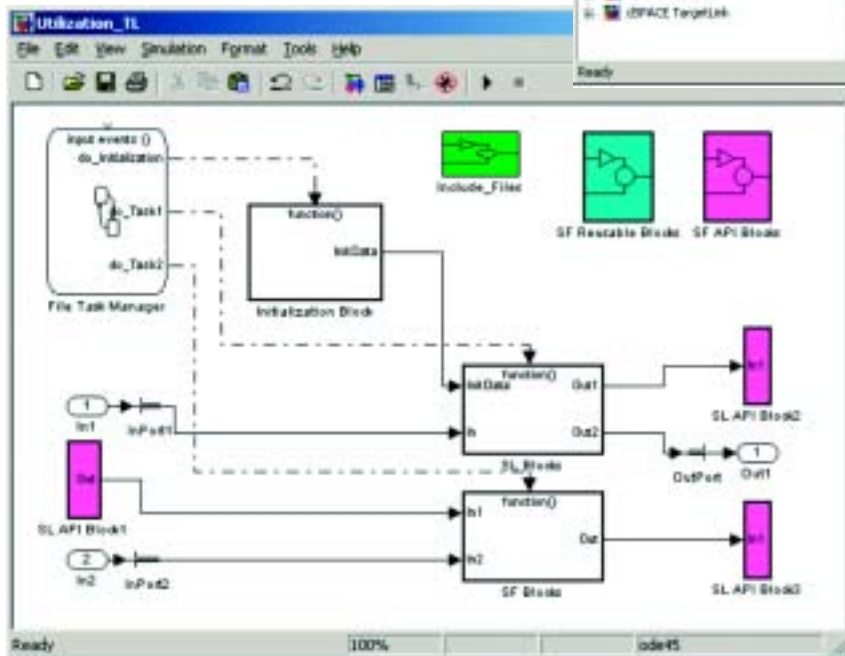


Fig. 2: A specified Delphi structure for models and model components.

Magneti Marelli's Design of a GDI Engine Controller

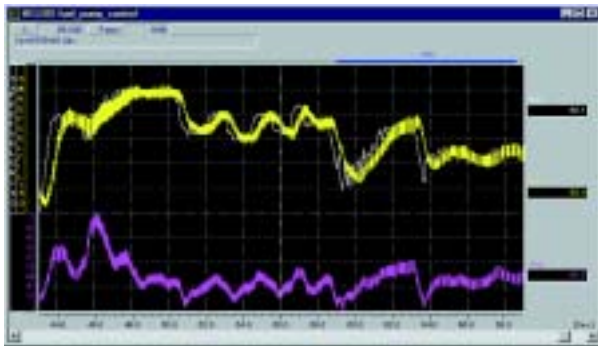
TargetLink supports Magneti Marelli's model-based design methodology

Improved time-to-market with gasoline direct injection (GDI) project

The design of powertrain controllers is among the most challenging problems in automotive electronics because of the complexity of the functions that the system has to support, like safety aspects, and tight cost limits. Time-to-market requirements and continuously changing specifications have contributed to the migration of most functions to software implementations. Magneti Marelli Powertrain is developing a model-based design methodology clearly defined by a set of successive refinement steps from a level of abstraction as high as possible down to the details needed for the final implementation. In this methodology framework, TargetLink has been successfully used.

In our approach, we identify five main levels of abstraction: system level, function level, operation level, architecture level, and component level. At the system level, the car manufacturer specifications are analyzed and expressed in an analytical formalism. At the function level, the system is decomposed into a set of functions. At the operation level, those functions are built by a network of operations. An operation is an

algorithm described in an analytical (at least executable) form, typically Simulink®/Stateflow®. At the architecture level, when an operation is committed to a particular architecture, it produces a software or hardware component.



Fuel pressure control displayed with Magneti Marelli's calibration tool, Helios: Fuel pressure target (white), fuel pressure measure (yellow), control valve angle command (magenta).

Convincing Features

In September 2000, Magneti Marelli Powertrain started to evaluate dSPACE's TargetLink to automatically transpose a Simulink/Stateflow operation specification into a software component. The first application chosen for this evaluation was a part of

a gasoline direct injection (GDI) control system: An electronic control unit (ECU) controls a high-pressure fuel pump in a closed loop. After 3-months of evaluation, we conclude that the tool is:

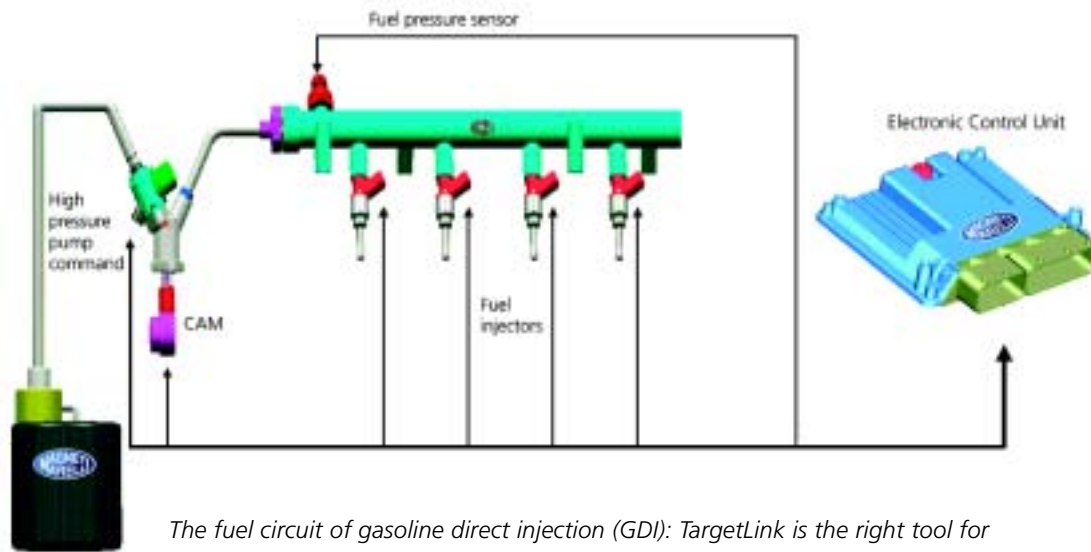
- Applicable to our model-based methodology
- User-friendly

And the generated code is:

- Suitable for our legacy software architecture
- Efficient enough in terms of size and speed

TargetLink As a Model Compiler

Today, we are using TargetLink as our main Model Compiler for the entire GDI engine control application. Any operation designed and verified by our control engineers using Simulink/Stateflow and assigned to be a software component is automatically translated to C code using TargetLink. The functional and implementation verification are still fundamental steps, for which the plant (i.e. engine) and driver inputs are modeled in Simulink/Stateflow using traces and/or complex models. The translation of a Simulink/Stateflow operation to C code might require different design steps. Moreover, the translation is part of a qualified software design process in which each step must be fully traceable. If the target architecture does not support native floating-point operations, the Simulink/Stateflow specification must be correctly and efficiently translated to the fixed-point



The fuel circuit of gasoline direct injection (GDI): TargetLink is the right tool for implementing the model-based design methodology in the new application.

representation of the target microcontroller. This floating-to-fixed point transformation must be verified by simulation (using a host machine), since no formal equivalence between the two representations can be proved. The last design step requires the definition of all the target dependencies, such as target software architecture hooks, RAM and ROM mapping. The outputs of this phase are the C code modules and the ASAP2 file, generated by TargetLink for fine-tuning activities in the application department.

Shorter Time-to-Market

By using the model-based design methodology, we aim to drastically decrease the time-to-market of a new powertrain controller. TargetLink has been a fundamental factor in achieving this goal.

It is important to emphasize that the correctness of the final implementation is closely linked to the correctness of the model, i.e., to the high-level description of the implementation, and hence to the ability to validate the specification. This validation requires the use of complex models of the environment (engine and car driver), and the more correctly addressed it is, the less usage of prototypes or targets is required, resulting in an enormous reduction in design time and cost.

Nowadays, this goal is still far from being completely achieved because of the difficulties of modeling the physical powertrain processes and the other electromechanical parts of the environment interacting with the controller in real time. To overcome these difficulties, prototyping and several

design cycles are still required. An important characteristic of the model compiler is that it hides the details of the target hardware/software architecture. For this reason we are also expecting to reduce the design time of other engine control systems due to extensive reuse of operation specifications.

The Right Tool for Model-Based Design

In conclusion, TargetLink has been the right tool for implementing our model-based design methodology, which is essential if we are to improve the time-to-market and cope with the complexity of modern powertrain controllers. In the future, we plan to extend the use of the model-based design methodology to other powertrain applications, and to exploit the new features of the coming releases of TargetLink.

The GDI project team has made extensive use of the model-based design methodology, mainly because of fast evolving system requirements, which create the need for a quick design cycle. Moreover, given the fairly new application, the methodology was applied from the earliest phases onward, allowing the GDI project team to manage complexity at a higher level of abstraction.

Magneti Marelli B.U. Systems:

Giacomo Gentile, Giovanni Stara, Luigi Romagnoli

Parades EEIG:

Alberto Ferrari

Italy

Honda: Testing Automatic Transmission ECUs

- **5-speed automatic transmission system**
- **dSPACE Simulator to develop the transmission's electronic control unit**
- **Fewer durability tests and dramatically reduced development time**

Honda has developed a direct control 5-speed automatic transmission system for front engine, front-wheel drive vehicle applications, which can be regarded as a new generation of automatic transmission for passenger cars. In addition to conventional simulation technology, Honda utilized a dSPACE Simulator to develop the transmission's electronic control unit (ECU). Using hardware-in-the-loop (HIL) technology dramatically reduced the overall development time.

Five-Speed Automatic Transmission

The new five-speed automatic transmission (AT) provides smooth gearshifting and a good response, along with great fuel economies in various driving conditions. Moreover, it has a sequential mode called S-MATIC, which drivers find fun to use because it gives them a real "gearshift" feeling. The new AT system is more compact in size, yet provides higher performance and higher quality compared to existing transmission systems.

This is all made possible by employing advanced technology such as ultra-flat torque converters, which make the transmission more compact, and the world's first ever clutch pressure control system using linear solenoids.

The AT is used in several models, for example, Honda's 2001 Acura RSX model.

The Need for HIL

Previously, too much time and manpower would have been required to determine the ECU specifications for the AT. We had to do validation tests with actual vehicles, and some of the tests really caused trouble.

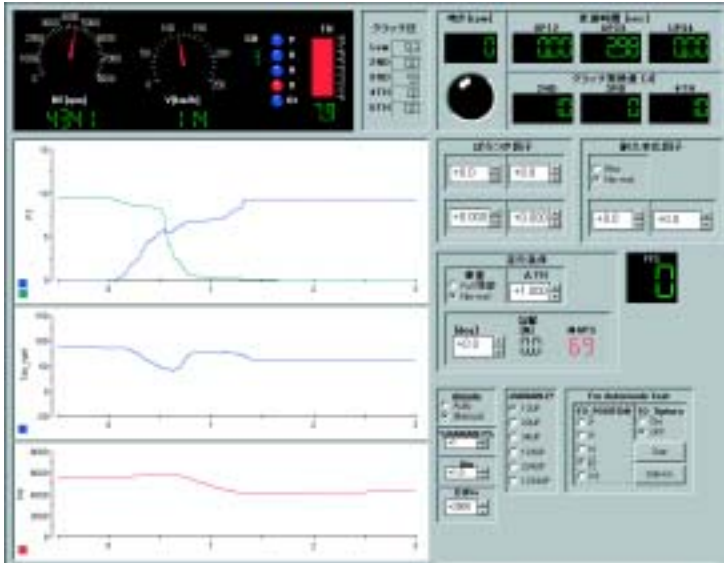
We felt that if the tests were implemented via simulation, we could expect a huge decrease in the amount of manpower required. We therefore incorporated the existing high-accuracy AT gearshift feeling simulation model into the system.

To find the best solution for our HIL system, we carefully compared and evaluated the various products available. The choice fell on dSPACE Simulator because of its mature technology both in overall functionality and in the built-in support structure for customization.

Our HIL environment consists of the actual ECU, a diagnostic monitor, dSPACE Simulator to run the simulation, and a host computer. There is an interface box for handling the communication processes by adjusting the signal levels between dSPACE Simulator and the ECU. Finally, there is a relay box that generates wire breaks and short circuits with arbitrary signals to simulate system failures.



Honda Acura RSX, equipped with the new 5-speed automatic transmission system.



ControlDesk is essential software for Honda's production AT development.

Regarding the electrical interface, we were able to build a highly accurate HIL system in a short time using dSPACE's high-performance I/O boards, for example, the DS5101 Digital Waveform Output Board. These modular dSPACE boards are capable of responding to the many input and output signals required for connection to the ECU, for example, the complex pulse patterns required by an engine or for data communication. We also constructed a new relay box in-house. This recreates electrical failure modes and is controlled from ControlDesk.

Simulating the Shift Feeling

In manual operation, parameters are specified by using the cockpit panel on the interface box and the ControlDesk software. Simulations are executed under the desired conditions by setting parameters for throttle angle signals, AT shift-lever signals, control signals for the relay box and simulation conditions.

In automatic mode, the operation signals are created by a driver model. The simulations are automatically executed by referring to a driving mode table and a failure mode table, generated in advance with ControlDesk's Stimulus Editor. The ECU connection information table was created in advance with Microsoft Excel.

However, some problems needed to be overcome. Because the offline simulation model for the AT transmission shift feeling required enormous

computing power, it was actually impossible to run the model in real time. By analyzing the necessary computation sampling rate for each component of the simulation model, we found that the model part for the AT torque transmission required the lowest execution time. By using a multirate system, we finally achieved satisfactory accuracy and real-time operation.

ECU Development for Mass Production

We also used our dSPACE Simulator to test our newly designed 5-speed automatic transmission systems, which are now going

into production, for example, in the Honda Acura RSX. Through simulation, we were able to reduce the number of durability tests carried out with actual vehicles, especially because dSPACE Simulator was able to reproduce the vehicle conditions at the end of the durability tests.

HIL systems face constantly growing demands for computation power caused by increasingly complex real-time models. The number of automatic test patterns and models an HIL system must handle is also growing.

We could not have developed the system without the integrated development environment, including dSPACE Simulator, the functionality and flexibility of its integrated I/O boards, and the powerful software, above all ControlDesk. We plan to widen the application range of dSPACE Simulator in our production AT development, and are anticipating a further reduction in the time required to determine the ECU specifications.

*Satoshi Terayama,
Ko Yoda,
Kenji Hagiwara
Honda R&D
Japan*

Enhanced RTI CAN Blockset

dSPACE Simulator has proven its value for many years, not only in standard hardware-in-the-loop (HIL) simulations, but also in the simultaneous testing of all the electronic control units (ECUs) in a car. Such ECU tests with large HIL systems require a high degree of performance and flexibility in all the tools involved, especially the CAN bus interface.

Typical tasks involved in the testing of networked ECUs are restbus simulation and the sending and monitoring of test-specific messages – in other words, the simultaneous testing of several interacting ECUs. For a successful test run, it is vital to provide efficient handling of the real-time models and

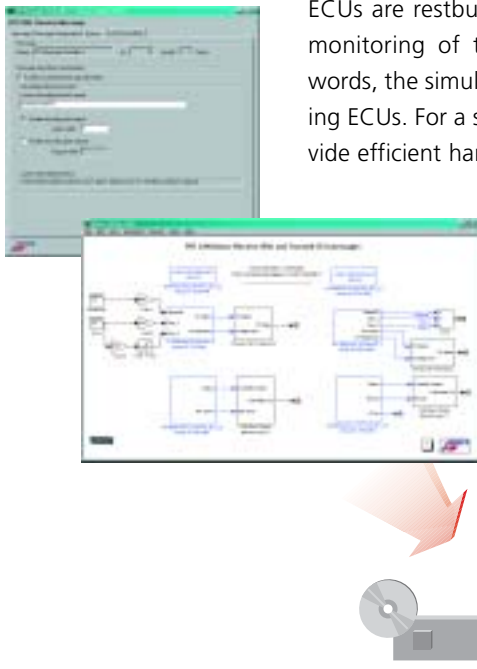
flexible processing of CAN bus messages, because the interrelationships between communicating ECUs are constantly changing. dSPACE has already released significant improvements in the form of the blockset extension RTI (Real-Time Interface) CAN Blockset 2.0, and there are further improvements with RTI CAN Blockset 2.1. The communication

behavior between control units is usually described by DBC files. These can be loaded by the RTI CAN Blockset and used for automatic configuration of the model's CAN I/O parts. The RTI CAN Blockset 2.1 offers fundamental improvements for efficient file import and for fast and automatic model reconfiguration.

What's New in RTI CAN Blockset 2.1?

- Multiple message access makes it possible to send or receive the same CAN messages from within different model subsystems. This makes the model much more flexible and gives it a clear structure
- Multiple DBC file support enables simultaneous use of several DBC files for one model, so the DBC file variants are easier to handle and can be switched during run time
- The new DBC file parser is much faster than the former version, which is particularly helpful for working with large DBC files in body electronics applications

The new version of the RTI CAN Blockset was released on CD-ROM in October. For more information, please contact your sales representative or www.dspace.de.



Update and LIN Node Simulation

- **Compatibility with MATLAB® Release 13**
- **New board for LIN node simulation**

MATLAB® Compatibility Update

Since mid-October, dSPACE provides an extension to dSPACE Release 3.4. This ensures the compatibility of dSPACE products with MATLAB Release 13, and is downloadable free of charge from our Web site: www.dspace.de



LIN Node Simulation with dSPACE Simulator

With the new DS4330 LIN Interface Board and the corresponding RTI (Real-Time Interface) LIN Blockset, dSPACE Simulator can be connected to a LIN (Local Interconnect Network) bus to simulate LIN slave nodes such as intelligent sensors and actuators. The board offers 16 independent LIN channels, with simulation of 16 LIN slave nodes per channel.

For more details, see www.dspace.de

If you wish to receive dSPACE release information via e-mail, please send your e-mail address to release.news@dspace.de

The Next Generation of Testing

Automotive software is steadily becoming more complex. More and more ECUs are being packed into modern vehicles, even networked, and often, several different bus systems are used. And as software complexity grows, so does the complexity of software development. With the time-to-market shrinking, and with diverse original equipment manufacturers and suppliers becoming involved, the development process is becoming harder to handle.

The situation calls for simultaneous engineering. New vehicle functions need efficient testing in the very early stages of the development process. So one of the major issues is automated, model-based testing in every development phase.

Today's Situation

Currently, testing procedures are often not seamlessly integrated into the overall development process. The usual method of writing tests by hand is error-prone and takes a lot of time and experience. Moreover, it is extremely difficult to prevent the handling of thousands of different tests and immense volumes of test data and test results from descending into chaos.

The dSPACE Solution

With many years of experience in test automation projects behind us, we at dSPACE set ourselves the goal of providing a powerful and convenient test automation environment to meet the requirements of automotive development for today and tomorrow. The result is AutomationDesk, which will support the task of developing and managing tests.

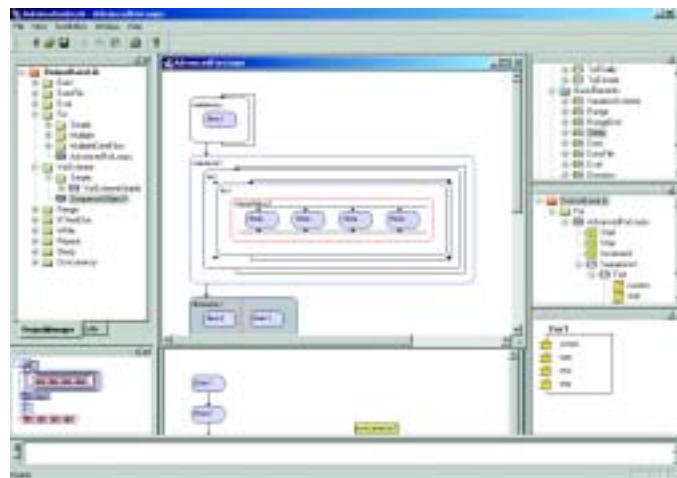
AutomationDesk

AutomationDesk will offer a comfortable graphical front end. This allows intuitive operation and facilitates system familiarization.

Sequence Editor

- Description of tests using a highly efficient graphical sequence editor
- Efficient development of automation sequences by drag & drop from AutomationDesk's library
- Options for test automation: Several automation interfaces can be used in the Sequence Editor by simply dragging them from AutomationDesk's library
- Easy access to Simulink® and real-time simulations. This enables automated testing to be done offline as well as online using hardware-in-the-loop simulators

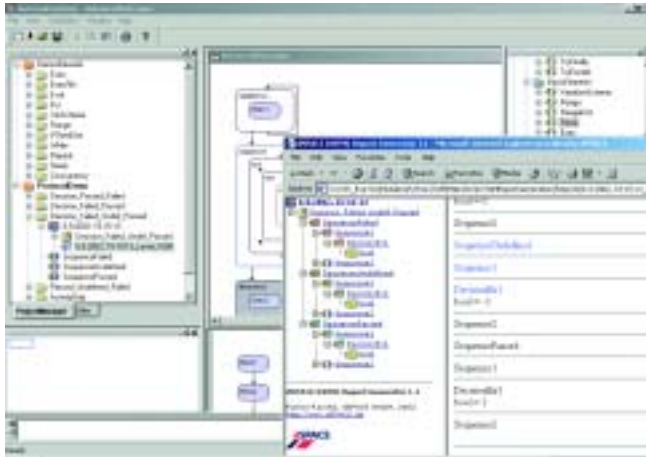
- Easy-to-use graphical sequence editor
- Management of large test projects
- Testing throughout the tool chain
- AutomationDesk release planned for mid 2003



AutomationDesk's Project Manager – a convenient way for organizing your test projects.

Project Manager

- Management of large test projects
- Test administration by easy structuring of test projects
- Interface for archiving and version control
- Automated report generation in different formats (for example, HTML, PDF)



Report generated automatically.

INFO ■ 01

Automation Libraries

AutomationDesk will provide a library containing a wide range of automation functions. Typical examples are the handling of stimulus signals or measurement data, the remote control of hardware-based failure simulation, diagnostic tools or calibration systems, and communication with external devices via different interfaces (for example, a RS232 PC interface).

- Interaction with third-party tools, for example, MATLAB® or office tools (using script modules, sending commands, performing calculations, creating plots, and so on)
- Developers can extend the libraries to fit their requirements

AutomationDesk is yet another powerful addition to dSPACE's range of tools. Because of its seamless integration into the tool chain and all its different features, it will mean great savings in the time and cost spent on future ECU developments.

Japanese User Conference

We are pleased to announce that our third Japanese User Conference will be held on November 26, 2002 at Cross Tower Hall in Tokyo. The conference will focus on new products to be released by dSPACE in the near future, including the TargetLink 2.0 code generator and the dSPACE Calibration System. Another main topic is the application

of model-based design approaches throughout the whole V-cycle. Our Japanese customers who are currently approaching this objective will present their successful results.

The comprehensive program includes speeches

from leading automotive customers presenting their development work from the individual stages of the V-cycle to the complete dSPACE tool chain. Hardware-in-the-loop tests with networked ECUs will be highlighted by a European application – the virtual car concept: This allows simultaneous testing of all



The participants exchange their knowledge and experience with other experts from the automotive and mechatronic sectors.



265 experts came together at the User Conference in 2001, providing us with positive feedback throughout.

interacting ECUs in a car. In addition, the dSPACE tool chain, which covers all stages of the V-cycle, will be introduced with vivid demo systems, such as rapid control prototyping, automatic production code generation and hardware-in-the-loop simulation.

We look forward to providing our Japanese customers with a forum for development experts to share their technical expertise.

Papers



Dr. Klaus Lamberg:
"Systematic Testing – Modern Solutions for
Hardware-in-the-Loop Simulation"

German ■02
English ■03

Please check the corresponding field on your response card. For more papers to download, visit www.dspace.de.

Job Opportunities



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

- Software Development
- Hardware Development
- Applications
- Technical Sales
- Product Management

Training



- dSPACE Systems
- ControlDesk Basics
- ControlDesk Advanced
- Test Automation
- HIL Simulation
- TargetLink

For more details, please visit www.dspace.de or check the corresponding field on your response card.

Events



EUROPE

Total Vehicle Technology

November 11-12, Brighton, United Kingdom
University of Sussex

Fortschritt in der Automobil-Elektronik

November 26-27, Stuttgart, Germany
Liederhalle
Booth #16

Autosport International 2003

January 9-10, Birmingham, United Kingdom
NEC
Booth #E235-E237

Embedded World 2003

February 18-20, Nuremberg, Germany
Messe Nuremberg, Hall 12
Booth #438

USA

Embedded Systems (ESC)

November 19-21, Boston, MA
Hynes Convention Center
Booth #1131

Conference on Decision and Control (CDC)

December 10-13, Las Vegas, NV
The Venetian Hotel

JAPAN

3rd dSPACE User Conference

November 26, Tokyo, Japan
Shibuya Cross Tower Hall 2F

For more details, please visit www.dspace.de.

Australia

CEANET Pty Ltd.
Level 1, 265 Coronation Drive
Milton
Queensland 4064
Tel.: +61 7 3369 4499
Fax: +61 7 3369 4469
info@ceanet.com.au
www.ceanet.com.au

India

Cranes Software Intern. Ltd.
29, 7th Cross, 14th Main
Vasanthanagar
Bangalore - 560 052
Tel.: +91 80 2381 740
Fax: +91 80 2384 317
info@cranessoftware.com
www.cranessoftware.com

Korea

Darim Systems Co., Ltd.
Expo VentureTown 3rd Fl.
DoRyong-Dong 3-1
YuSung-Gu
Taejon
Tel.: +82 42 601 1366
Fax: +82 42 864 4749
sales@darimsystems.co.kr
www.darimsystems.co.kr

Sweden

FENGCO Real Time Control AB
Hallonbergsplan 10
Box 7068
SE-174 07 Sundbyberg
Tel.: +46 8 6 28 03 15
Fax: +46 8 96 73 95
sales@fengco.se
www.fengco.se

China and Hong Kong

Beijing JiuZhou HiRain Tech. Co. Ltd.
Shangfang Plaza No. 27
Room 620
Bei San Huan Zhong Lu
100029 Beijing, P.R. China
Tel.: +86 10 820 219 50
Fax: +86 10 620 736 00
ycji@hirain.com
www.hirain.com

Israel

Omikron Delta (1927) Ltd.
10 Carlebach St.
Tel-Aviv 67132
Tel.: +972 3 561 5151
Fax: +972 3 561 2962
info@omikron.co.il
www.omikron.co.il

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

Taiwan

Scientific Formosa Incorporation
11th Fl. 354 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 00 Praha 8
Tel.: +420 2 84 01 17 30
Fax: +420 2 84 01 17 40
info@humusoft.cz
www.humusoft.cz

Japan

LinX Corporation
1-13-11 Eda-nishi
Aoba-ku, Yokohama-shi
Kanagawa, 225-0014 Japan
Tel.: +81 45 979 0731
Fax: +81 45 979 0732
info@linx.jp
www.linx.jp

Poland

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

Headquarters in Germany

dSPACE GmbH
Technologiepark 25
33100 Paderborn
Tel.: +49 52 51 1638-0
Fax: +49 52 51 66529
info@dspace.de
www.dspace.de

France

dSPACE Sarl
Parc Burospace
Bâtiment 17
Route de la Plaine de Gisy
91573 Bièvres Cedex
Tel.: +33 1 6935 5060
Fax: +33 1 6935 5061
info@dspace.fr
www.dspace.fr

USA and Canada

dSPACE Inc.
28700 Cabot Drive · Suite 1100
Novi · MI 48377
Tel.: +1 248 567 1300
Fax: +1 248 567 0130
info@dspaceinc.com
www.dspaceinc.com

United Kingdom

dSPACE Ltd.
2nd Floor Westminster House
Spitfire Close · Ermine Business Park
Huntingdon
Cambridgeshire PE29 6XY
Tel.: +44 1480 410700
Fax: +44 1480 410701
info@dspace.ltd.uk
www.dspace.ltd.uk

