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.
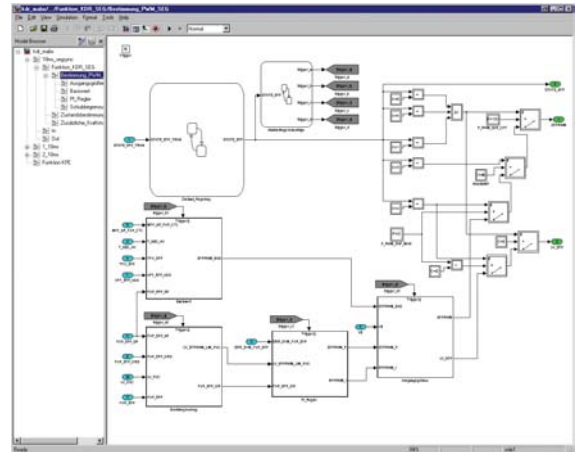
Simulink function blocks for fuel control.

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