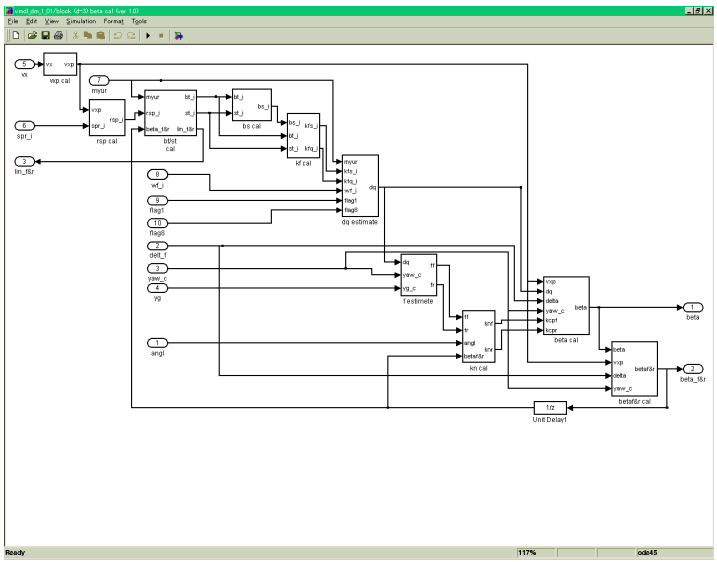
TargetLink for ESP Controller Development

CUSTOMERS

- *Porting time for multiple processors massively reduced*
- *Implementation of functions without any assistance from programmers*
- *TargetLink code for production-type controller*
- *100% ANSI-C-compliant code with TargetLink*

The most recent automotive controllers tend to serve more demanding functions, which increases the complexity of the implemented control logic. For example, the Electronic Stability Program (ESP) system developed at Unisia Jecs Corp. supports oversteer and understeer control. Compared to ABS, the code of the ESP system is up to 3 times longer. The demand for shorter development cycles increases the pressure on design engineers to come up with new products. We therefore evaluated the dSPACE production code generator, TargetLink, and its impact on the development process.



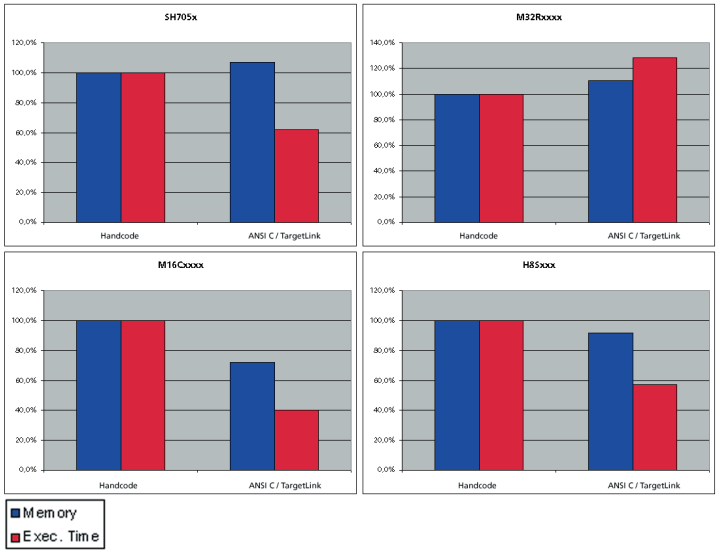
Simulink model for slip-angle estimation with an observer.

A Natural Decision

The control logic is developed on the basis of a MATLAB/Simulink/Stateflow environment. The conversion of the graphical specification into C code is usually done manually for use in a production-type electronic control unit (ECU). Consequently, we investigated if a tool can be applied to generate the production code automatically. Since TargetLink works hand in hand with Simulink, we decided to evaluate TargetLink's suitability.

Starting Small

We chose the vehicle slip-angle estimation module for the evaluation, which is the core element of the ESP controller. The corresponding algorithm performs a slip-angle estimation, which is implemented by an observer methodology. The input data includes steering angle, yaw rate and lateral acceleration. The controller also contains error-sensitive integrators and various other types of control functions, which increases the code size considerably. We compared the handwritten C code with the code generated by TargetLink. This comparison covered code size and execu-



Memory consumption and execution time on each processor.

Papers

L. Köster, T. Thomsen, R. Stracke: "Connecting Simulink to OSEK: Automatic Code Generation for Real-Time Operating Systems with TargetLink" (SAE Paper 2001-01-0024)
English 04
German 05

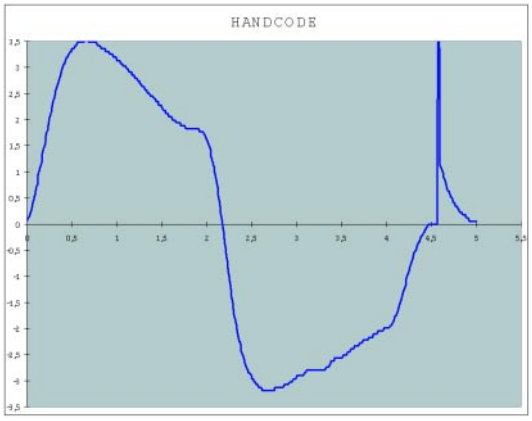
H. Schütte, M. Plöger, K. Diekstall, P. Wältermann, T. Michalsky: "Test Systems in the ECU Development Process"
English 06
German 07

T. Pöhlmann: "Nissan Sentra: Wesentliche Funktionen des Steuergeräte-Codes mit automatisch generiertem Code programmiert"
German 08

tion time, and was carried out according to ANSI specifications. In other words, no processor-specific code was generated since at that time there was no optimization module available for our processor types.

The Comparison

For evaluation purposes, we tried the TargetLink code on various types of processors, with either RISC or CISC cores, from Mitsubishi and Hitachi. We particularly had a close look at portability and compared the hand code with the generated code. What we found was this: The TargetLink-generated code for the ESP controller performed identically on each examined processor. In contrast, the ESP controller we coded manually always required further modifications, even though our programmers thought they complied with the ANSI standard fully. This is especially important when transferring the code to a different processor type. With TargetLink, the time for porting the code is decreased by 75%. The maximum increment in code size is only a few percent, except for the Mitsubishi M32R. The code for the H8S and the M16C is actually smaller than the hand code. Since the TargetLink code shows improved execution times, we achieved a better performance than we initially thought by going for the SH705, H8S and M16C.



Hand code for the H8S needs further amending due to overflow.

The Consequences

Based on our findings, we decided to opt for TargetLink as the tool to code the ESP controller. TargetLink has now become the standard for generating production code in our development department. In our opinion, TargetLink's most important feature is the ability to generate the code directly from the Simulink diagram. Our engineers achieved a highly-efficient development process with TargetLink without any assistance from programmers. This is extremely important in a real vehicle environment, especially when carrying out off-site tests such as winter test drives.

Please note that this article only reflects TargetLink's capability to generate ANSI C compliant code. If you use the optional TargetLink Optimization Modules, you can generate target-specific code, which takes full advantage of processor and compiler-specific features.
TargetLink Optimization Modules with TargetLink 1.2 are now available for various types of processors, such as Mitsubishi M32R, Hitachi H8S or Motorola HC12.

Future Plan

We intend to use TargetLink for coding projects with greater complexity and even more code. This will require improved libraries. Here we expect further progress with the use of TargetLink 1.2.

Hitoshi Kobayashi
Motohiro Higuma
Chassis & Drive Train
Control Systems
Development Dept.
Unisia Jecs Corporation

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:
- August 21/22, 2001

ControlDesk Basics
Paderborn, Germany:
- September 11, 2001

ControlDesk Advanced
Paderborn, Germany:
- September 11/12, 2001

Test Automation
Paderborn, Germany:
- September 12/13, 2001

TargetLink
Paderborn, Germany:
- July 19, 2001
- August 30, 2001
- October 18, 2001

HIL Simulation
Paderborn, Germany:
- October 10/11, 2001

INFO 01