TargetLink Goes Off-Road

The Chrysler group of DaimlerChrysler used TargetLink to generate production code for the 2005 Jeep® Grand Cherokee. 100% of the application code for the Electronic Transfer Case and the Electronic Limited Slipped Differentials (ELSDs) was autocoded by TargetLink. During implementation, the developers found that "TargetLink is a very versatile tool" and they were "continuously impressed with the readability of the generated code". There was no need to modify the code manually.

- ◤ **Autocode in production vehicle**

- ◤ **Chrysler group uses TargetLink for Jeep Grand Cherokee**

- ◤ **Large model fits on a standard controller**

New Active 4x4 Drive

For the 2005 Jeep Grand Cherokee, we designed a new four-wheel drive system. The system can actively shift torque to all four wheels to provide maximum available traction in off-road driving. The electronic limited slip differentials (ELSD) help provide the ultimate in active traction management for every off-road challenge. Expected wheel speeds are compared with the actual speeds to detect wheel slip at the earliest possible moment. When slip is detected, the appropriate differentials are electronically activated to redirect engine torque to the wheels with the most traction. Known as Quadra-Drive II®, this system helped the Jeep Grand Cherokee win the 2005 4x4 of the year award from Petersen's 4-Wheel & Off-Road magazine.

Introducing Automatic Production Code Generation

To develop the application software for the Jeep Grand Cherokee, we implemented a model-based autocode integration process for the first time. After reviewing available options, TargetLink was chosen as the production code generator. Our group had experience using model-based function development for rapid control prototyping using a dSPACE AutoBox and ControlDesk for another program. For this application, a dSPACE bypass POD was used to allow the dSPACE MicroAutoBox to interface directly to our target hardware for rapid prototyping. TargetLink is also used to create the ECU description file in the standardized ASAM-MCD 2MC file format (formerly ASAP2) to make the variables and parameters available for ECU measurement and calibration.

◀ *The Jeep Grand Cherokee is the perfect combination of Jeep's legendary off-road capabilities with the ultimate in technical equipment and interior features.*

Autocode Integration Process

To support autocode generation with TargetLink, we have developed our own internal process as follows:

Step 1
Model conversion

After function design, simulation, and verification with MATLAB®/Simulink®/Stateflow®, we convert the Simulink model to TargetLink. The first time we did this, functional equivalents had to be found for any Simulink blocks which were not supported. The result was a 4-page modeling guideline called the "Model Autocode Readiness Review Checklist".
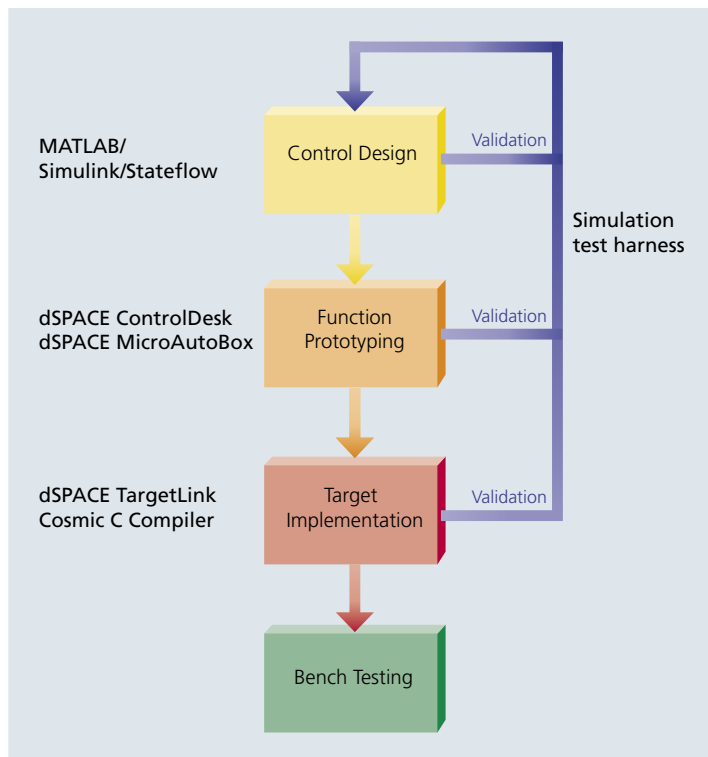
▲ *Workflow from control design to target implementation.*

Step 2
Simulation test harness

We created a simulation test harness which allows the execution of an integration test suite containing over 110 test cases. These tests are executed under simulation while the minimum and maximum values of every node in the model are recorded. We used custom written M-scripts to compare the model outputs with a baseline output file.

Step 3
Simulation-based autoscaling

Using the data collected in step 2, the fixed-point model can be autoscaled with TargetLink.

Step 4
Floating-point versus fixed-point simulation

The integration test suite is run again, only this time in fixed-point mode, recording the data at every node in the model.

Step 5
Error resolution (loss of resolution/word width)

Using the data collected in step 4, the fixed-point model output can be compared against the floating-point model output. Where there are significant errors, we have to trace back from the output to the source of the error, which is usually loss of resolution due to a too narrow integer data type.

Step 6
Target mode simulation on target processor

Initially, we did not use processor-in-the-loop simulation (PIL) on an evaluation board. We thought it was better to test the software after it was integrated with the other software components. But we have found PIL simulation with TargetLink to be a valuable tool in that it allows us to run our integration test suite and log stack usage and execution time for each test case.
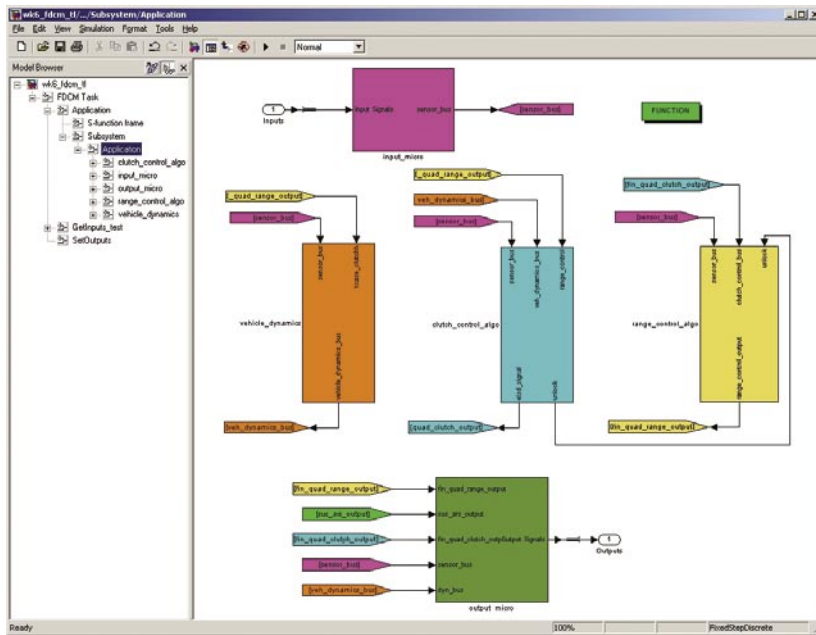
Step 7
Export C code and integrate with other components (integration layer)

The model is then exported as C code, and integrated with the low-level drivers, the OSEK operating system, and CAN drivers. This integration involves writing a small function to call the low-level functions, pass the inputs to the model functions, and update the outputs via the low-level functions with the values returned by the model.

Step 8
Verification

The integration test suite is repeated on the target hardware using an HIL simulator, and the results verified.

▲ *Screenshot of the TargetLink model.*

We produced good results in terms of execution time, about 5 ms out of the 10 ms sample time. Stack size was 1600 bytes and RAM size was 4400 bytes.

TargetLink 2.0 and MTest for the Next Jeep Model

TargetLink is a very versatile tool. We will reuse our Simulink models for autocode generation and testing with TargetLink for the 2006 Jeep Grand Cherokee. We'd like to upgrade to TargetLink Version 2.0, since we expect further benefits in time savings from the improved autoscaling capabilities of TargetLink 2.0. We would also like to investigate using dSPACE MTest for more convenient, automatic, and systematic model-based testing and test case development during control design with TargetLink.

Thomas H. Pruett
DaimlerChrysler Corporation
Active Chassis Controls
USA

Amazing Results

The newly designed ECU controls the Electronic Transfer Case Control and the Electrically Controlled Limited Slip Differentials. Each subsystem has its own control algorithm modeled as a separate Simulink subsystem. The vehicle dynamics model constitutes another Simulink subsystem. An input block consisting of conversions and filters and an output block form two more Simulink subsystems. The TargetLink model file on the host PC is about 13 MB in size and consists of 4709 blocks. The final code size of the application code is only 90 KB (on a Motorola MC9S12DT256 processor). The final executable code size, after being linked with the OSEK operating system, CAN drivers, diagnostics, and low-level drivers is about 160 KB. We are highly pleased with the results achieved using TargetLink for autocode generation. We were impressed with the readability of the generated code, and there was no need to modify it manually.