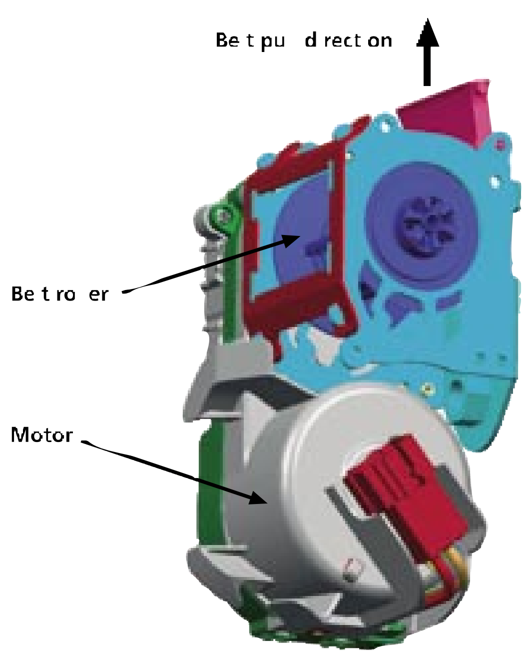
Active Control Retractor with TargetLink at TRW

- TargetLink production code for the vehicle dynamics functionality
- TargetLink, the easy-to-use development environment
- Integration of the software module after only four weeks

TRW has developed a reversible belt pretensioner that gives the driver and other vehicle occupants enhanced protection compared with conventional seat belt systems. When vehicle dynamics become critical, the seat belt is pretensioned by an electric motor. The project took only 18 months to develop and successfully used TargetLink to generate production code for the entire vehicle dynamics part.

New Belt Pretensioner – Proactive and Reversible

Present-day belt retractors are usually triggered pyrotechnically by the airbag ECU (electronic control unit) during a collision. Such systems are irreversible and only leap into action *after* a crash. In future, vehicle occupants will be protected *before* a crash occurs, in the pre-crash phase, whenever a critical situation is assessed on the basis of vehicle dynamics. The system uses an electric motor to minimize belt slack. This gives occupants firmer restraint. If the crash is avoided, and vehicle dynamics have restabilized, the belts are automatically loosened and the reversible belt pretensioner is immediately ready for action again. If a collision does occur, the vehicle's occupants are already held firmly in position, which considerably improves the effectiveness of the other safety systems.



TRW's reversible belt retractor.

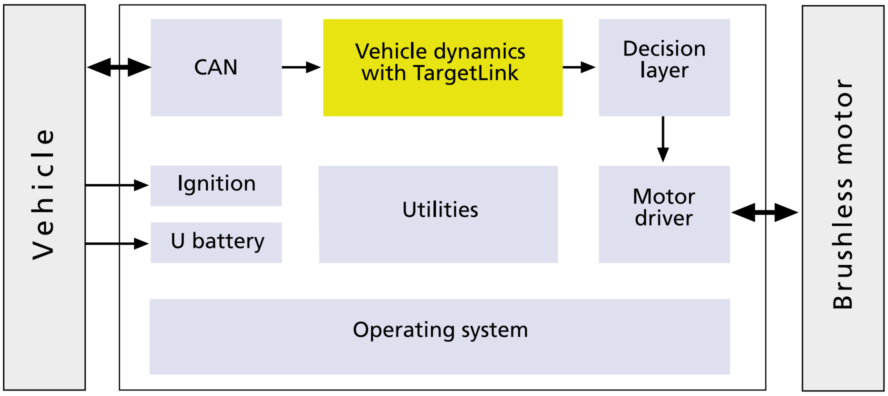
How It Works

The electric motor works on the belt roller to tighten the belt. The motor is activated by an ECU. This receives data on the current vehicle dynamics via the CAN bus, evaluates the data and if necessary tells the electric motor to tighten or loosen the belt. The vehicle dynamics functionality that forms part of the ECU processes a total of 14 input signals to produce 11 computed output values.

TargetLink generated the production code for this part of the ECU, which amounts to around 20% of the overall functionality including all utilities and drivers.

Why TargetLink?

The customer already had the functions for evaluating vehicle dynamics in the form of a MATLAB®/Simulink® model. TargetLink was ideal in this situation, because it could create and integrate the vehicle dynamics software model straight from MATLAB/Simulink. Developers became familiar with TargetLink's easy-to-use development environment quickly, resulting in faster implementation and verification of the specification. Module testing was carried out completely on an evaluation board. This meant that we could assure the quality of the production code very early on. TargetLink's documentation features meant that no further documentation work on the production code was needed. Thus, in addition to the time saved, the quality of the documentation was also improved as regards being up-to-date and code-consistent.



The ECU software module.

Integrating TargetLink into Our Development Environment

To begin, we acquired TargetLink Base Suite, Target Simulation Module and an evaluation board. After installing these tools, we adapted the Target Simulation Module to the compiler we were using, and created interfaces to the software project. When the Simulink models were converted into TargetLink models, special library functions were also created. Further steps included specifying the maximum stack size and the maximum computing time, and defining the variable types and their scaling. The code generation, testing and model optimization processes were then run iteratively until all the requirements had been met. After only four weeks of putting TargetLink into action, we were able to integrate the finished vehicle dynamics software module into the project.

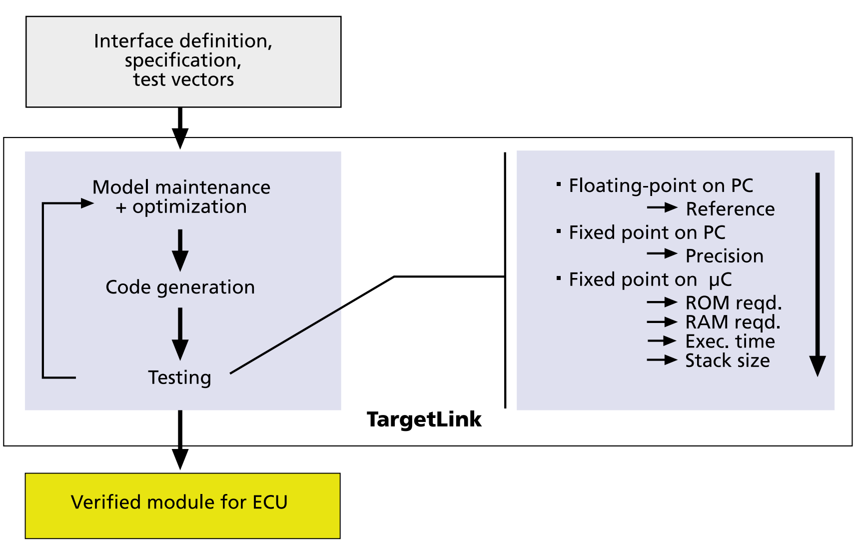
Particular Challenges

Increasing complexity and stiffer requirements regarding computing precision made it necessary to develop the vehicle dynamics model further. We also had to adapt it to the microprocessor resources, to meet constraints such as stack size and computing time. One example of how we did this was by optimizing the data flow to eliminate 64-bit operations. More resources were freed by changing the representation of the parameters and variables from arbitrary to a power of two.

Outlook

TargetLink was the right choice. The software module we produced will be integrated into future product applications and existing modules will be implemented using TargetLink. We also plan to use TargetLink to generate production code from Stateflow.

Ega Tschiskale
TRW Automotive
Electronic Control Systems
Electrical Drives Department
Head of Hardware\System
Germany



Module test with TargetLink.