



# Discover Real-Time Testing

The real-time testing toolbox for a wide range of application fields

Real-Time Testing is the ideal companion to the AutomationDesk test automation tool. For over two years, customers have been impressed by its value. The method allows the clock-synchronous execution of real-time test scripts and the Simulink simulation model.

The real-time test scripts are executed on a Python interpreter that is integrated into the real-time application. The interpreter can load various scripts during model run time and execute them independently of the PC without recompiling. Python is an easy-to-learn standard language for creating an understandable, flexibly extensible description of test scenarios that can

be reproduced with precise timing. The scripts can detect all changes to model variables and react to them in real time in the same simulation step.

#### Cover More Test Scenarios

The first version of Real-Time Testing, for the DS1006 Processor Board, came out at the end of 2006. DS1005 support was added a few

months later. These classic HIL platforms are supported as single- and multiprocessor systems. Then many of our customers asked for portation to MicroAutoBox. We responded to this need by providing AutomationDesk 2.2, which was launched in late 2008. Real-Time Testing can now also be used on the MicroAutoBox, for example, for RCP and for in-vehicle scenarios.



for that to happen. The automatic load mechanism supports the replay of large data volumes, such as test drive log files several hundred MB in size. Several replay processes can run simultaneously, and they can all be controlled independently of one another. In a multiprocessor system, it is even possible to stimulate on several subnodes time-synchronously.

#### **Generate More Traffic on the CAN Bus**

For programming CAN restbus simulation, Real-Time Testing is conveniently integrated with the RTI CAN MultiMessage Blockset. The blockset contains an option for preparing CAN access for Real-Time Testing, so that the real-time scripts can send and receive CAN messages with freely definable CAN

transmission, restbus simulation, etc., are distributed across several processor boards to ensure real-time capability. To implement real-time scripts independently of the structure of the HIL model, Real-Time Testing provides transparent variable access in the MP system. The communication channels needed between the processor boards are set up dynamically by the Python test scripts during model run time, instead of being created statically in advance by the modeler. Local and remote variables can both be accessed during testing: It is sufficient to specify the unique MP variable path.

#### **Do More with Real-Time Testing**

Customers are already using Real-Time Testing in very diverse ways: for real-time stimulus generation,

## **Real-Time Testing: Create precisely reproducible test scenarios with Python.**

#### **Stimulate More Data with Precise Timing**

Numerous customers use Python real-time scripts to perform precisely timed stimulus generation for pre-defined signal behaviors. Recorded measurement data can be replayed via an intelligent load mechanism, in which the Python real-time script references a recorded file (MAT file) on the PC's hard disk. The script links data vectors in the file to the target parameters in the model. If the real-time test requires data to be replayed exactly 50 ms after a specific CAN message is received, for example, a simple replay command in the Python script triggers real-time-capable data transport from the PC to the real-time hardware in time

IDs and contents (see "All you CAN test" in dSPACE Magazine 01/2008). "This mechanism is so powerful that we also use it for replaying recorded bus traffic in ControlDesk," says Product Manager Holger Krisp. With Real-Time Testing, it is very easy to implement dynamic stress tests that continuously push the CAN bus to its transmission limits by feeding in interference messages. This procedure verifies whether the connected ECUs function correctly even in this extreme situation.

#### **More Flexibility in Multiprocessor Systems**

Today, HIL integration tests are executed on multiprocessor (MP) systems where submodels for the engine,

real-time observers, online time measurement in a millisecond raster, dynamic restbus simulation, and even complex integration tests – there's almost no limit to the possible uses. "dSPACE will continue putting new things into the Real-Time Testing toolbox in the future to open up new applications fields, such as direct ECU access from real-time tests via XCP on CAN," says Holger Krisp. ■