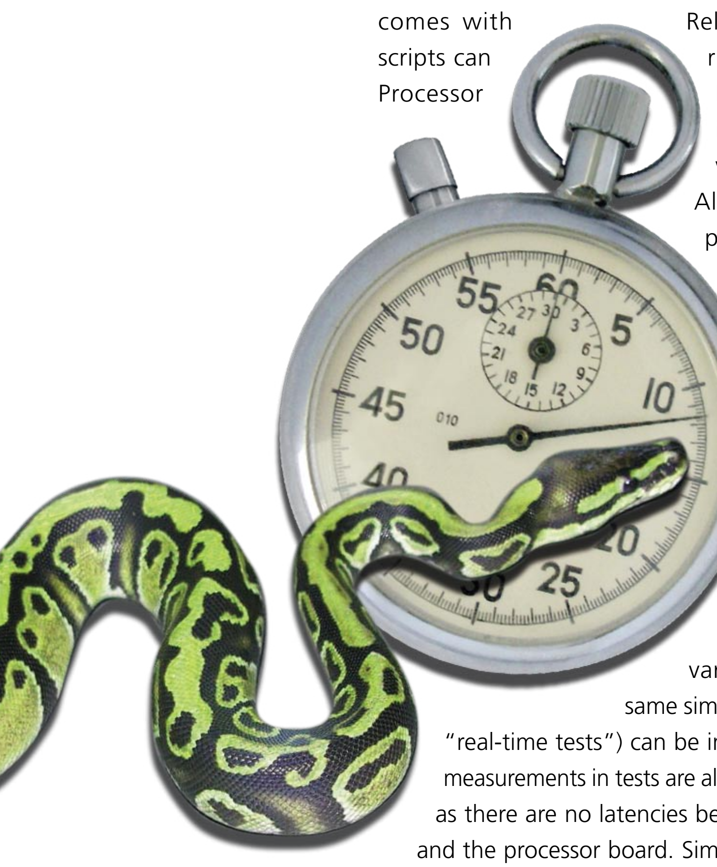
Automated Tests in Real Time

- Tests executed synchronously to model
- Python is the test description tool for real-time tests
- Test execution in real-time on the processor board

Real-time, hardware-in-the-loop (HIL) simulation on special processor boards has become a standard for testing mechatronic electronic control units (ECUs). Harnessing the full potential of the simulator hardware requires powerful tool automation, however. AutomationDesk is a test automation and test management tool that meets this need. You can create tests either graphically or via script programming and run them on a PC. From AutomationDesk 1.4, Python scripts can run on the DS1006 Processor Board in real time, i.e., synchronously with the model, independently of the PC. This means you can implement tests with tough requirements regarding timing accuracy and reactivity.

Describing Real-Time Tests in Python

A test automation package containing Python, the object-oriented scripting language, and AutomationDesk has long been available from dSPACE. From AutomationDesk 1.4, which comes with Release 5.2, Python scripts can run on the DS1006 Processor Board in real-time, i.e., synchronously with the model. All test actions are performed on a real-time basis, so the solution opens up completely new test options on the HIL simulator. Reactive tests that can react to changes in model variables within the same simulation step (called "real-time tests") can be implemented. Time measurements in tests are also far more precise, as there are no latencies between the test PC and the processor board. Simulation step size is now the only thing restricting the maximum time resolution of the measurements.

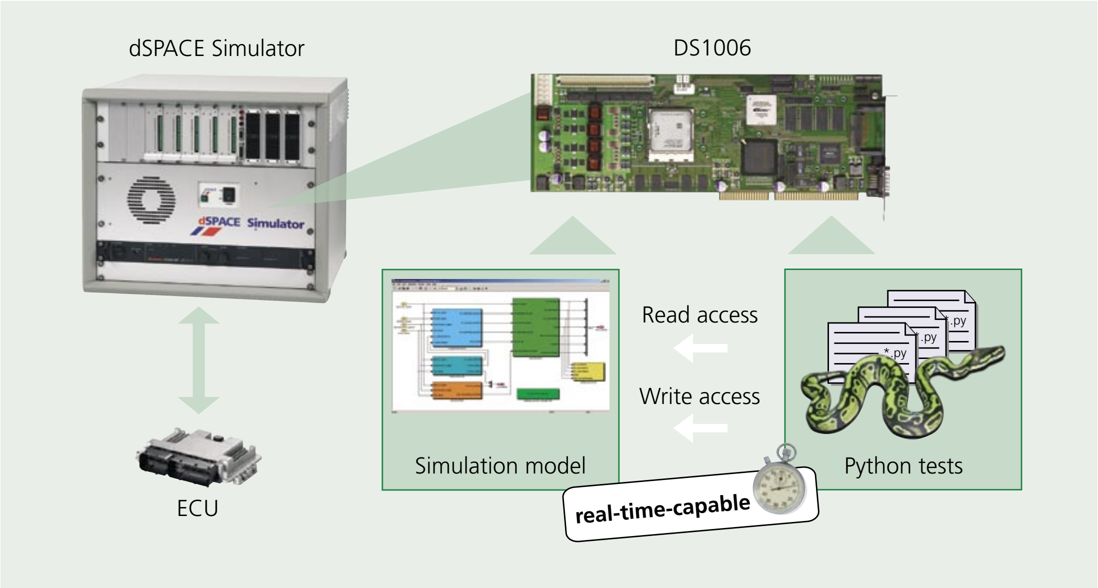


Real-Time-Capable Python Interpreter

A real-time Python interpreter, running on the DS1006 Processor Board along with the model, allows the script to execute synchronously. The interpreter can execute several real-time tests simultaneously and independently of one another. The tests can interact with the simulation model in real-time via the memory on the processor board. You can therefore observe and influence the ECUs connected to the HIL simulator in every individual simulation step.The Python interpreter is added to the application during the translation process via a Real-Time Workshop build option. You can formulate real-time tests using standard Python scripts from the new libraries that dSPACE provides (for example, for accessing model variables and executing several test branches within one real-time test in parallel). You can also create your own libraries and reuse them in several tests. The tests you have created can be loaded from the PC to the simulator's processor board and executed regardless of whether there is a real-time test already running. Even though executing the real-time tests requires additional memory capacity and computing time, it is no problem to implement typical test scenarios, which run in parallel to complex engine and vehicle dynamics models, at simulation step sizes of 1 ms.

PC and Processor Board Work Hand in Hand

Because the real-time tests run on the processor board, the PC is no longer needed once they have



▲ *Real-time-capable Python interpreter for the DS1006 Processor Board.*

been loaded and started. However, in some cases it may make sense to combine test execution on the PC and the processor board. Future versions will support this, making it possible for Python scripts running on the PC and on the processor board to call each other's functions and exchange data. For example, the real-time test can collect result data for a certain period and transfer the data to the PC, which then generates a report. Or a real-time test can access a diagnostics system connected to the PC to query a fault memory entry in the ECU.

Summary and Outlook

AutomationDesk 1.4 provides the real-time-capable Python interpreter for the DS1006 Processor Board. Several real-time tests can be loaded separately and executed synchronously to the simulation model. The tests have read and write access to model variables. The functionality for real-time testing will be extended step by step. Various libraries will be added for convenient access to HIL hardware (such as FIU, CAN, and diagnostic access) and for easier data exchange between the PC and the processor board. There will be library functions for extended real-time data acquisition, allowing data recording to be started and stopped according to model variables. The Python interpreter will be further developed to include the DS1005 PPC Board and multiprocessor systems (based on DS1005 and DS1006).

Advantages of Python Real-Time Tests

- Real-time tests programmed in Python standard language
- Test descriptions can be extended by user (libraries)
- Test action descriptions with precise timing information
- Reproducible execution of test cases
- Simulation models need no modification for real-time tests
- Read and write access to model variables possible in every simulation step
- Dynamic reloading and execution of tests during model simulation