Hard Real-Time Test Tools - Concepts and Implementation

Jan Peleska
TZI Center of Computing Technologies, University of Bremen, jp@tzi.de
Cornelia Zahlten
Verified Systems International GmbH Bremen, cmz@verified.de

In this presentation, the authors describe concepts and techniques which are crucial for testing embedded hard real-time systems:

- Test specification formalisms should be capable of describing timing conditions, sequencing and synchronisation of events and transformation of both discrete and time-continuous ("analogue") data.
- Test generation algorithms have to provide coverage both in the data and the time domain.
- Test executions have to be performed in real-time, preferably with on-the-fly data generation and evaluation of the responses from the system under test (SUT).
- Test tools require both hardware and operating system support in order to guarantee precise timing of generated test data and measured SUT responses.

In the past, the fourth requirement listed above has led to the development of test systems based on VME, VXI platforms or on specialised hardware, driven by proprietary operating systems. In many applications, this resulted in high hardware costs and performance problems due to the limited VME and VXI bus capacities. Moreover, software services typically available on standard operating systems (standard graphics, file systems, graphical user interfaces or communication protocols) were difficult (and expensive) to integrate into the real-time environments.

The novel approach described by the authors relies on a test system cluster architecture based on dual CPU or 4-CPU PCs acting as cluster nodes. The nodes communicate and synchronise over a Myrinet high-speed network. A modification of the Linux operating system allows to run the test generation and evaluation algorithms in hard real-time on reserved CPUs, where scheduling is non-preemptive and controlled by the test system itself. The interrupts caused by interfaces to the system under test may be relayed to CPUs designated explicitly for their handling. This approach offers the opportunity to utilise high-performance standard hardware and the services provided by the widely accepted Linux operating system in combination with all mechanisms required for hard real-time computing. The cluster architecture presents an opportunity to distribute interfaces with high data throughput on different nodes, so that PCI bus overload can be avoided. In addition, the CPU load can be balanced by allocating test data generators, environment simulations and checkers for SUT behaviour ("test oracles") on dedicated CPUs.

The test data algorithms proposed by the authors derive inputs to the SUT and their associated points in time automatically from test specifications. The specifications are designed as networks of cooperating timed state machines. These networks can be shown to operate in bounded response time; that is, they are suitable for on-the-fly test data generation in hard real-time. The state machines are enhanced by specifications for the time-continuous behaviour of physical observables like temperature, thrust, speed which are measured or changed using analog sensors and actuators. Test evaluation algorithms operate in soft real-time; that is, SUT errors in the test execution can be detected on-the-fly, with minimum delay.

The specification of test data generation rules, simulations of components in the SUT environment and checking rules for expected SUT responses can be composed using a variety of formalisms such as tables, diagrams, textual specification language and explicit C or C++ code. This allows to choose description techniques which are most appropriate for different specification requirements.

The presentation is intended for an audience of people responsible for testing embedded real-time systems, decision makers in the tool selection process, as well as research and development specialists in the field of test tools for embedded real-time systems.

The concepts and techniques described have been implemented in Verified Systems' RT-Tester product which has been developed in cooperation with the first author's research group in the Center for Computing Technologies at the University of Bremen. RT-Tester is applied for the test of embedded systems in the field of avionics, railways, space applications, automation and automotive controllers. Support to connect to the SUT via most interfaces used in the embedded systems domain is available, e.g. Ethernet, serial, parallel, discrete and analogue I/O, IEEE 1394 (FireWire), CAN, Profibus, ARINC 429, MIL-STD 1553, AFDX.

RT-Tester has been qualified for the test of several controllers used in the Airbus aircraft family according to the RTCA DO178-B standard applicable for airborne software-based control systems. Verified Systems has the policy to publish the underlying methods and implementation techniques used in the RT-Tester tool, in order to facilitate the tool qualification process. The modifications of the Linux kernel supporting hard real-time PC clusters are available to the open source community under the Gnu Public Licence agreement.

The examples presented have been derived from tests recently performed and from new testing environments currently developed by Verified Systems for controllers in the Airbus A318, A340 and A380 aircrafts.