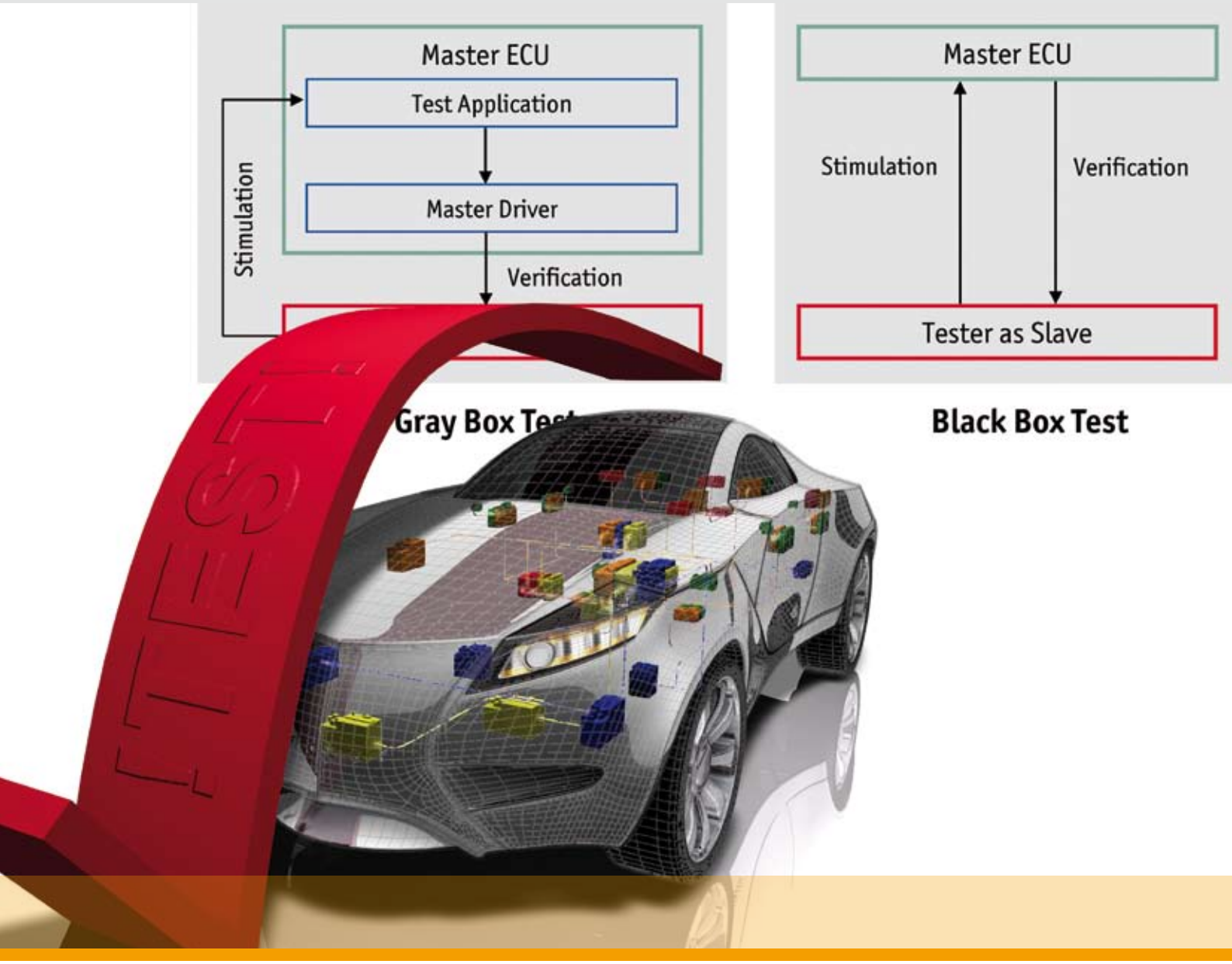

Assuring the Quality of LIN ECUs

Current and Future Strategies for LIN Master Conformance Tests

High quality and reliability are essential preconditions for the successful application of bus systems to modern automobiles. Due to the significant increase in the number of LIN (Local Interconnect Network) components used in automotive developments, efficient test strategies for this cost-efficient bus system are gaining in importance. Vector Informatik describes the current possibilities for testing LIN nodes according to the latest LIN conformance test specification. For Master nodes a new and efficient black box test strategy is presented based on the well-known development and simulation tool CANoe.LIN.

The Author

Gavin C. Rogers, B.Eng M.Sc., is team leader and product manager for LIN tools at Vector Informatik in Stuttgart (Germany).

1 Introduction

The LIN consortium provides in addition to the specifications for each the protocol version corresponding conformance test specifications. The LIN conformance tests are used to verify whether a LIN device is conform to a specific protocol version and also serve as a basis for the LIN accreditation. Since LIN networks operate according to the Master-Slave principle, the protocol conformity of a Master node is of utmost importance. The LIN conformance tests are specified separately for each OSI layer: physical layer, data link layer, network management and node configuration. Only application layer tests need to be specified by the OEM or supplier.

Black box tests are best suited for the methodical implementation of conformance tests, since they exclusively use a device's external interfaces (e.g. LIN interface) to stimulate and verify each test case. White box tests on the other hand always require access to a device's internal interfaces (e.g. the LIN driver's standardized interface). A LIN Master ECU offers a very limited number of stimulation options via its external interfaces such as CAN. Gray box tests that combine the two test methods are therefore the most commonly used method use to realize a LIN Master conformance test, Figure 1.

2 Test Environment for LIN Conformance Tests

A typical ECU test case requires configuration and initialization of the test system and ECU under test, as well as subsequent stimulation and verification. The Slave conformance tests provided with CANoe.LIN from Vector Informatik are implemented almost entirely as black box tests. The tester in its role as LIN Master can usually perform the stimulation and verification directly via the LIN bus. Only a few tests require manual stimulation or verification e.g. in order to stimulate a wakeup signal. The Master conformance test, on the other hand, is implemented by the Stuttgart-based tool provider as a gray

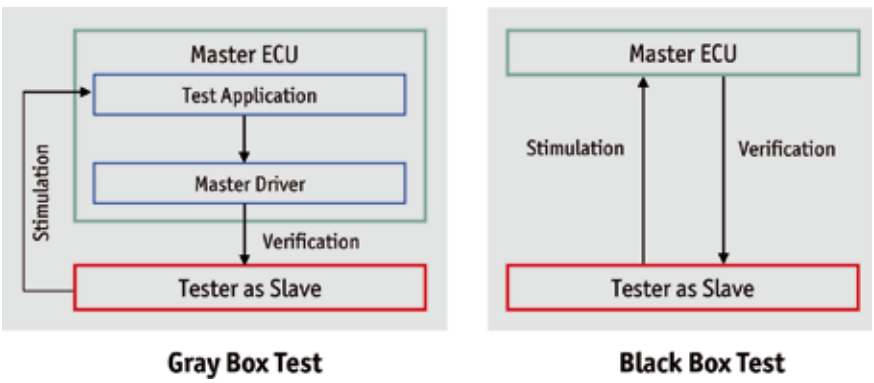

Figure 1: The LIN Master conformance test is usually implemented as a gray box test. A black box test would, however, be preferable

box. A special test LIN Description File (LDF) is provided to ensure the correct configuration of the Master driver. Together the generated driver code a special test application must be downloaded to the Master under test. In this way, the LIN bus can be used for both stimulation and verification purposes, despite the tester's role as LIN Slave.

The Delphi Technical Center in Krakau, Poland, has gained considerable experience with CANoe.LIN in the field of testing for both LIN2.0 and J2602, the U.S. version of LIN. Since the LIN conformance tests do not cover the OSI application layer, Delphi TCK has extended their test activities to cover various application tests. The main focus of these tests is to test: signals, schedule tables, gateway routings, and diagnostics. The CAPL test functions provided with CANoe.LIN have proved to be indispensable in implementing and automating such tests. According to Delphi TCK even very complex test cases were easy to implement and extend using the C-like CAPL syntax.

3 Disadvantages of Gray Box Master Tests

With respect to usability, a gray box implementation of the Master conformance test has several disadvantages. For example it is not possible to perform this test during all phases of development. Furthermore, some preparation effort is involved, e.g. configuration and generation of the Master driver code according to the test LDF, before the Master-ECU can be tested.

The gray box Master conformance test allows the Master driver interface to be indirectly accessed by the LIN-bus via a test application. Although full test coverage can be achieved in this way, this approach means that the usual V-model development process cannot be strictly followed. For example, it is only possible to verify the Master's conformity at the beginning of the development process. As soon as the productive database (LDF) and application are running on the Master ECU, further verifications can no longer be easily performed. Furthermore, the OEM can not repeat the Master conformance test without assistance from supplier.

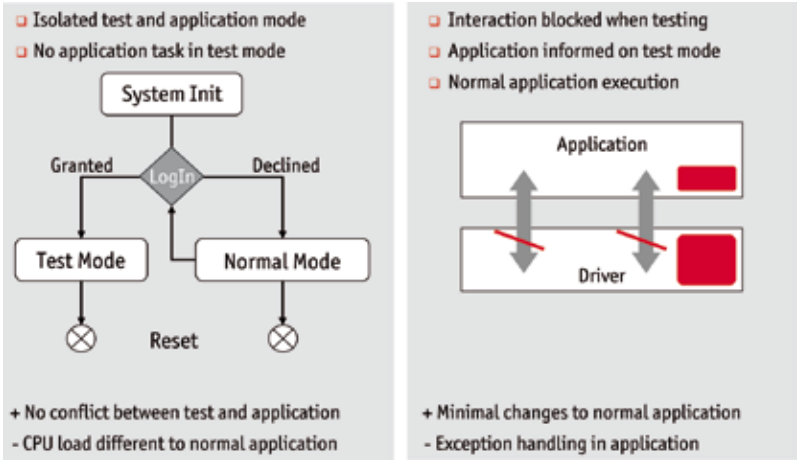

Figure 2: The test communication required to realize a Master conformance test as a black box test

4 The Way to a Black Box Master Test

Consequently, many automotive OEMs and suppliers often ask whether the Master conformance test can be implemented as a black box test. This would have the big advantage that OEMs and suppliers could independently perform tests during the entire development process with minimal configuration effort.

However, in order for a black box Master conformance test to have genuine advantages over a gray box test, certain requirements must be first met. Probably the most important one is that the same driver software used for development must also be used for tests. One way of achieving this is to extend the LIN Master driver with a special test interface.

This driver extension must permit the direct stimulation and verification of the Master under test via the LIN bus by providing the tester with special test services, e.g. to change the schedule table or read the driver's status word. The test communication over the LIN bus be-

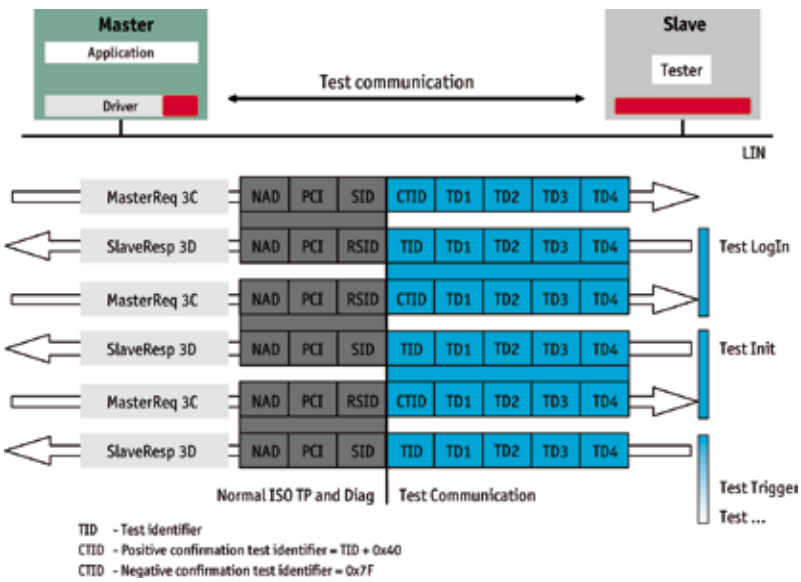

Figure 3: Two possible strategies for the harmonic interaction of master application and test execution

Test Commands	Parameters	Test ID
Test LogIn	none	0x00
Test Presence / Idle	none	0x01
Test Setup / Trigger	T_{base} Cycles, Reset/Init indication	0x02
Load Parameters	Offset, D1, D2, D3	0x03
Read Result	Offset, D1, D2, D3, D4	0x04
Set schedule table	Table index, Slot index, T_{base} Cycles	0x05
Read Status Word	PID / none	0x06
Send Config Command	RCSID, RCNAD, RCPCI	0x07
Request Sleep Mode	Sleep cyc., Schedule tab, wakeup Del.	0x08
Set signal	Signal handle, Signal value	0x09
Read signal	Signal handle	0x0A

Figure 4: An overview of the 11 test commands, which allow a complete coverage of the current LIN2.0 Master conformance test

tween Master and tester can use a special diagnostic service, comparable to those used for reconfiguration services. A further requirement on such a driver extension is of course the minimal increase in ECU resources. The test interface should also be removable from the productive code, e.g. via a preprocessor define.

5 Prototype of a Black Box Test for LIN 2.x

The LIN specialists at Vector were requested by an OEM to design and specify a black box Master conformance test. During specification of the prototype, it became quickly clear that the test communication must not negatively affect the normal operation of the Master-ECU. This can be only achieved by applying the standard LIN diagnostics consisting of Master Requests and Slave Responses using a new special diagnostic service. Only in this case, it is the tester as Slave and not the Master who initiates communication. The tester sends a test command to the Master-ECU, which can respond either positively or negatively, **Figure 2**.

Before sending each test command it is necessary to execute a test log-in and a test initialization. This raises the question to which extent a conformance test can be implemented independently of the Master's normal application. A completely independent and parallel operation of both test and application proved to be impossible. The application must therefore be involved in the realization.

There are basically two possible strategies for handling the interaction between the application and the driver's test mode, **Figure 3**. One way is clearly inform the application that a test is being executed. An alternative method is to hide the test execution from the application as far as possible. Both strategies have their advantages and disadvantages. The final choice of strategy may depend on the feedback and wishes of Master-ECU suppliers. In both cases existing drivers need to be extended to support the required test services. The additional code required for the test-server functionality has been estimated to be 20-30 % of current LIN2.x drivers and can be considered unproblematic for most Master ECU projects.

The current prototype implementation provides eleven services, which is more than sufficient to satisfy the LIN 2.0 Master conformance test, **Figure 4**. The proposed concept from Vector is not only extendable, but can also be easily standardized, since it is based on the existing LIN development process and protocol specifications.

6 Perspective

Apart from these activities, Vector plans to further develop CANoe.LIN. For example, support of LIN 2.1 Slave tests is planned for SP4/5 of Release 7.0 in the third quarter of 2008, assuming that the LIN 2.1 conformance test specifications are published in the second quarter of 2008. Support of the LIN 2.1 Master tests can be expected with Release 7.1 of CANoe.LIN in the fourth quarter of 2008. In addition to its development, analysis and simulations tools, the Stuttgart-based company also provides training and workshops related to all aspects of networking with CAN, LIN, MOST and FlexRay. Through numerous customer projects, both OEMs and suppliers also profit directly from 20 years of networking experience and expertise at Vector Informatik. ■