

A novel approach to integrating Boundary Scan with other board-level ATE

Heiko Ehrenberg (h.ehrenberg@goepel.com)
GOEPEL Electronics LLC, Austin, Texas, USA

This article introduces a novel approach for the integration of JTAG/Boundary Scan tools with other Automated Test Equipment. Previous technologies typically stopped short of being really integrated by just executing the same stand-alone Boundary Scan test vectors on a third party ATE, instead of combining resources of both the JTAG tool set and the third party ATE. HySCAN, the method introduced in this article, takes advantage of abstract test resource definitions, allowing the same set of Boundary Scan test vectors to be executed on various ATE equipment, utilizing ATE test resources to extend the test coverage compared to stand-alone Boundary Scan, without the need of adapting the Boundary Scan test pattern.

Introduction

Boundary Scan¹ – as we know it – is well established and widely used for various board- and system-level test and in-system programming applications. These applications typically include connectivity tests between Boundary Scan capable devices and so called cluster tests to verify connectivity to memory and logic devices that don't provide Boundary Scan capabilities (also referred to as non-Boundary Scan devices). The testability and the achievable test coverage are Unit Under Test (UUT) specific and depend on the level of Boundary Scan implementation [1][2][3]. Especially in production test, measures are taken to extend the Boundary Scan test coverage by utilizing tester resources connected to peripheral I/O on the UUT. This article discusses such extended Boundary Scan applications in the following paragraphs. In the recent past, integrations of third-party Boundary Scan tools in board level production test equipment, such as Flying Probe Testers or In-Circuit Testers, won in popularity. Often times such integrations provide little benefit over stand-alone Boundary Scan systems. While some of those solutions provide improved test coverage by combining Boundary Scan and ATE² resources, they introduce new problems like skewed test

coverage statistics and very limited portability. This paper introduces the HySCAN™ concept, which has been developed to overcome many of these limitations.

What is “extended Boundary Scan”?

Stand-alone JTAG/Boundary Scan systems

A typical stand-alone Boundary Scan system consists of a processing unit (e.g. a PC³) running Boundary Scan software, a Boundary Scan controller, a power supply (to power the UUT), and cabling to connect the UUT to the Boundary Scan controller and the power supply (Figure 1).

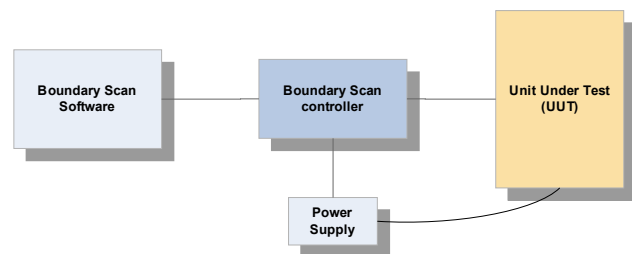


Figure 1 Stand-alone Boundary Scan system

Such a stand-alone Boundary Scan system supports various test and in-system programming applications utilizing the test resources available on the UUT (i.e. the Boundary Scan cells and other test features) – including, but not limited to, Interconnect Test, Memory and Logic Cluster Test, FLASH programming and PLD programming (Figure 2).

Some Boundary Scan systems provide tools to extend the test coverage by including non-Boundary Scan circuitry typically not testable with automatically generated test pattern. Such applications include, among other things, the verification of an oscillating clock signal, test of switch or jumper settings and illumination of optical indicators with operator feedback (Figure 3).

¹ A.k.a.: JTAG, BScan, IEEE Std.1149.1

² ATE = Automated Test Equipment; here: reference to board level production test equipment, such as In-Circuit Tester, Flying Probe Tester

³ PC = Personal Computer

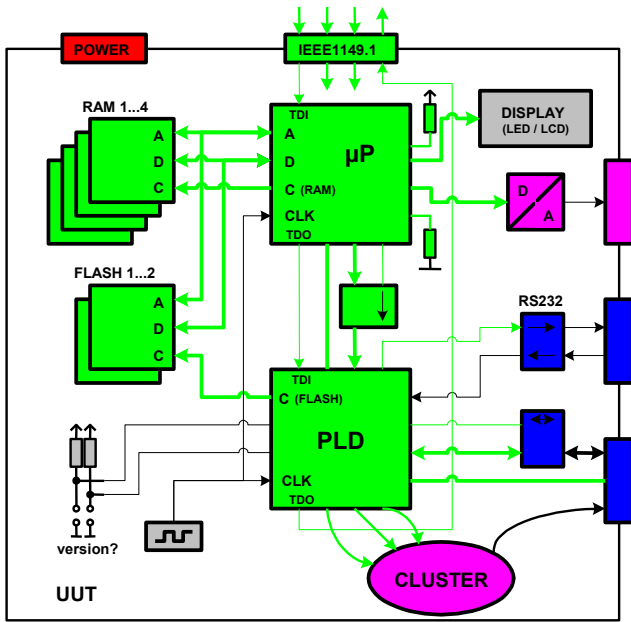


Figure 2 Common Boundary Scan applications (green)

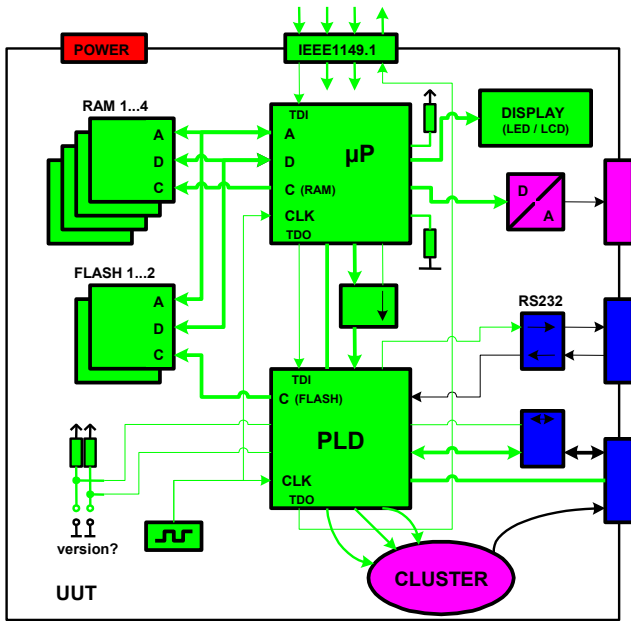


Figure 3 Extended Boundary Scan applications (display, oscillator, jumper tests)

By complementing the UUT's on-board test resources with external test channels (Figure 4), the test coverage can be further extended to the UUT's interface circuitry connected to peripheral connectors (Figure 5). Especially when utilizing external access to test points on otherwise via Boundary Scan inaccessible nets, the diagnostic output can be improved (e.g. for cluster testing).

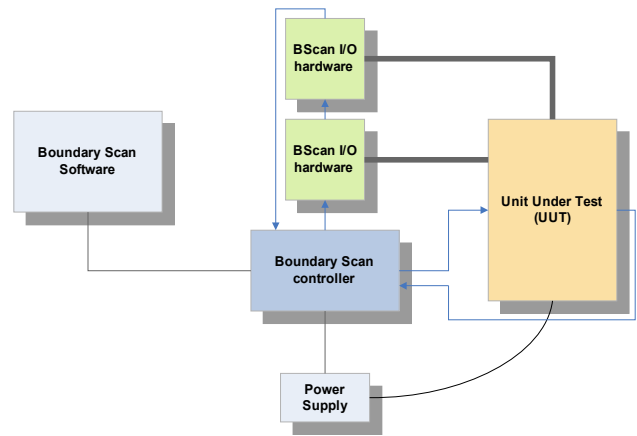


Figure 4 Stand-alone Boundary Scan system with BScan I/O resources

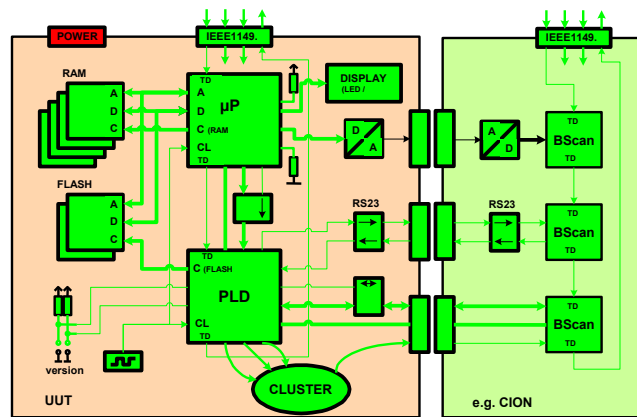


Figure 5 Extended Boundary Scan applications, including interface tests

Even though both test coverage and diagnostics are improved when BScan I/O tester hardware is utilized, new problems are introduced with this method. When Boundary Scan tests are developed considering just the UUT's test resources, then all test coverage statistics are based on the number of nets and the number of pins of the UUT only. By adding tester resources to the Boundary Scan setup, however, the tester resources themselves are included by automated test coverage analyzers, skewing the statistics. For test generation, these Boundary Scan I/O tester resources are merged with the UUT net list. Then test programs are generated based on this merged net list (the tester I/O's become part of the UUT). This causes the number of nets and pins evaluated during the test coverage analysis to increase. The result of the test coverage analysis does not represent the UUT anymore, but rather the combined UUT and tester I/O configuration. Table 1 provides an example for the impact of this combined net count on test coverage statistics for a Boundary Scan Interconnect Test.

	UUT	BScan test adapter	UUT + BScan test adapter ¹⁾
Total net number	163 (100%)	389	532 (100%)
BScan net number	116 (71.2%)	384	480 (90.2%)
Completely testable ²⁾	11 (11.7%)		31 (6.5 %)
Partially testable ²⁾	75 (79.7%)		441 (91.9%)
Not testable ²⁾	8 (8.5%)		8 (1.6%)

Table 1 Skewed test coverage statistics

1) Assumption: 20 BScan nets on UUT connected to BScan test adapter
 2) nets connected through serial resistors are counted as one net for testability statistics

Such a skewed test coverage statistic is misleading when one wants to analyze the suitability of Boundary Scan for a specific UUT and the impact of added I/O resources.

Furthermore, since the Boundary Scan I/O resources are merged with the UUT net list for the purpose of test program generation, the developed test pattern can only be executed on this specific tester configuration. As soon as the number or the order or the type of Boundary Scan I/O resources changes, the test pattern need to be modified accordingly. This drastically reduces the portability of the Boundary Scan test programs.

A third problem – although less problematic for most applications – of adding Boundary Scan I/O resources to the tester setup is the increased length of the scan chain. Additional TCK clocks required to shift test pattern through the Boundary Scan I/O module result in longer test execution time and reduced effective test throughput. However, such I/O modules can be connected to a separate Boundary Scan chain, reducing or eliminating the impact on test execution time (depending on the number of Boundary Scan cells in the I/O module as compared to the number of Boundary Scan cells on the UUT).

Integrating JTAG/Boundary Scan with other test equipment

Traditionally, Boundary Scan test coverage has been extended using Boundary Scan I/O resources on stand-alone test configurations only. For many years, however, there also have been available Boundary Scan tools on board level ATE equipment, such as In-Circuit Testers and Flying Probe Testers. Often times, though, these Boundary Scan tools were limited in their capabilities, which triggered interest in integrating third-party Boundary Scan tools, which typically offer more advanced features and better debugging capabilities, in addition to their ability to run as a stand-alone setup. Such integrations allow the execution of previously for stand-alone systems developed Boundary Scan applications on ATE equipment. However, this was only possible without

modifications to the test programs if they did not utilize any tester I/O resources. One example is the integration of Boundary Scan tools on Flying Probe Testers. A stand-alone Boundary Scan system may utilize Boundary Scan I/O resources to extend the test coverage on the UUT's interface circuitry. An integration of the Boundary Scan tools on a Flying Probe Tester, utilizing the probes to contact nets in order to improve testability and diagnostics (Figure 6), requires the development of additional test programs. It would be desirable to run the same test programs on the stand-alone system and on the integrated solution in order to reduce test development time and maintenance.

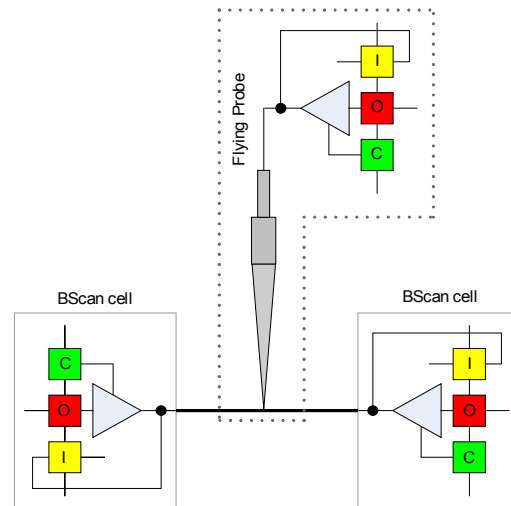


Figure 6 Boundary Scan net contacted with Flying Probe to improve diagnostics

HySCAN – a novel approach for Boundary Scan integrations

The portability of Boundary Scan test programs and in-system programming routines would be greatly improved if the test pattern would be independent from the test system hardware, especially when considering the use of I/O resources used to extend the Boundary Scan test coverage.

Meet HySCAN⁴. This new concept of independently controlling tester I/O resources has been developed to allow the execution of the same Boundary Scan test pattern on various ATE systems, without any modification. The principle behind HySCAN is the separation of shift vectors and parallel I/O vectors. Test programs include both the test pattern for the UUT's Boundary Scan chain and for the tester's I/O resources. However, the test pattern for the I/O resources is independent from the actual tester hardware. The same

⁴ HySCAN: short for Hybrid Scan

I/O test pattern is used for Boundary Scan I/O modules, for Flying Probes, for In-Circuit Test Nails, or for other ATE I/O resources. When the test programs are ported to the various ATE, the actual tester I/O resources are mapped to the I/O test pattern by means of a tester configuration and a wiring list.

Figure 7 shows a tester configuration with a Boundary Scan controller and additional Tester I/O hardware. The Boundary Scan controller provides access to the UUT's JTAG test access port as well as a trigger interface to the Tester I/O hardware. Latter is controlled directly by the software – test pattern is transferred to and from the I/O resources over the host bus interface, rather than through a Boundary Scan chain.

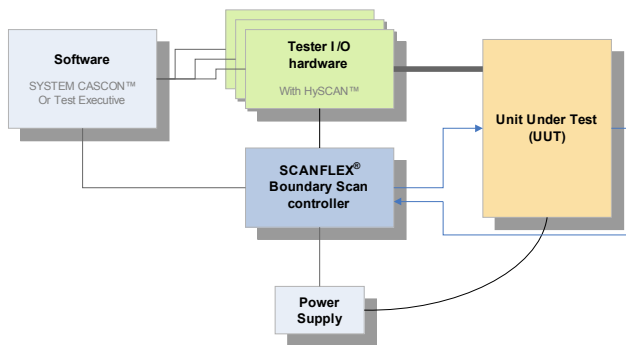


Figure 7 Boundary Scan system with HySCAN capability

Providing access to the UUT peripheral interface connectors with HySCAN I/O resources provides the same improvements in test coverage that can be achieved with Boundary Scan I/O modules that utilize scan chain access. However, since HySCAN I/O pattern are not described as being part of a Boundary Scan chain, but rather as parallel I/O pattern, the test programs for such a configuration can be ported to utilize other ATE I/O resources (e.g. Flying Probes) without any modifications in the test program. Furthermore, since the HySCAN I/O pattern is independent from the UUT Boundary Scan description, the test coverage statistics are not distorted. Table 2 shows the difference in test coverage statistics when utilizing Boundary Scan I/O modules that are merged with the UUT vs. the implementation of the HySCAN concept. Considering UUT and tester I/O resources separately ensures accurate and UUT oriented test coverage statistics.

	UUT	UUT + BScan test adapter ¹⁾	UUT + HySCAN ¹⁾
Total net number	163 (100%)	532 (100%)	163 (100%)
BScan net number	116 (71.2%)	480 (90.2%)	116 (71.2%)
Completely testable ²⁾	11 (11.7%)	31 (6.5 %)	31 (33 %)
Partially testable ²⁾	75 (79.7%)	441 (91.9%)	55 (58.5%)
Not testable ²⁾	8 (8.5%)	8 (1.6%)	8 (8.5%)

Table 2 Correct test coverage statistics with HySCAN

1) Assumption: 20 BScan nets on UUT connected to test adapter
 2) nets connected through serial resistors are counted as one net for testability statistics

The HySCAN concept is based on hybrid test pattern, combining serial access through the UUT's Boundary Scan chain and parallel access through tester I/O resources. These tester I/O resources should be individually programmable as input, output or 3-State. Voltage level programmability and considerable driver strength, combined with over current protection or other safety measures, are desirable. The Boundary Scan test programs utilizing these HySCAN I/O resources do not need to take into consideration the physical realization of these I/O's. For the test pattern it does not matter whether the I/O's are provided via parallel I/O modules or via tester channels on an In-Circuit Tester or a Flying Probe Tester. The test program is generated based on the UUT net list and a description of which nets are accessible to tester I/O resources. Then, a tester configuration file and wiring list provides information about the actual assignment of tester I/O's to nets on the UUT. Therefore, only the tester configuration and wiring list needs to be adapted to the actual test setup, not the test program itself.

Figure 8 shows an example Boundary Scan project that utilizes tester I/O resources. The project includes three tester configurations for a stand-alone Boundary Scan system, an In-Circuit Test system, and a Flying Probe Test system. Only one set of test programs is required, the test programs include pattern for the tester I/O resources. The same tests can be executed on all three test systems; the mapping of tester I/O pattern in the test program to the physical tester I/O resources is realized with the respective tester configurations.

In this example, the stand-alone Boundary Scan system is realized with a PXI Boundary Scan controller, a PXI Digital I/O module with 192 channels (only 20 channels are used), a PXI Power Supply module, and a PXI Mixed-Signal I/O module. The 20 test channels from the I/O module are connected to a peripheral interface connector on the UUT. The test program controls both the UUT's scan chain and the I/O module's test channels.

On the In-Circuit Test system, the UUT's peripheral interface connector is accessed through a test fixture. The

test channels are realized by the In-Circuit Tester's I/O resources. The same test program that had been used on the stand-alone Boundary Scan test system now controls the UUT's scan chain and the In-Circuit Tester's test channels.

The Flying Probe Tester provides access to the UUT's peripheral interface connector through moving probes, connected to the tester's I/O resources. In this configuration, the same test program used on the stand-alone Boundary Scan system and the In-Circuit Tester is applied.

Control of the various test systems' I/O resources is realized through HySCAN drivers, which interface the Boundary Scan software and the tester configuration with the ATE software/hardware.

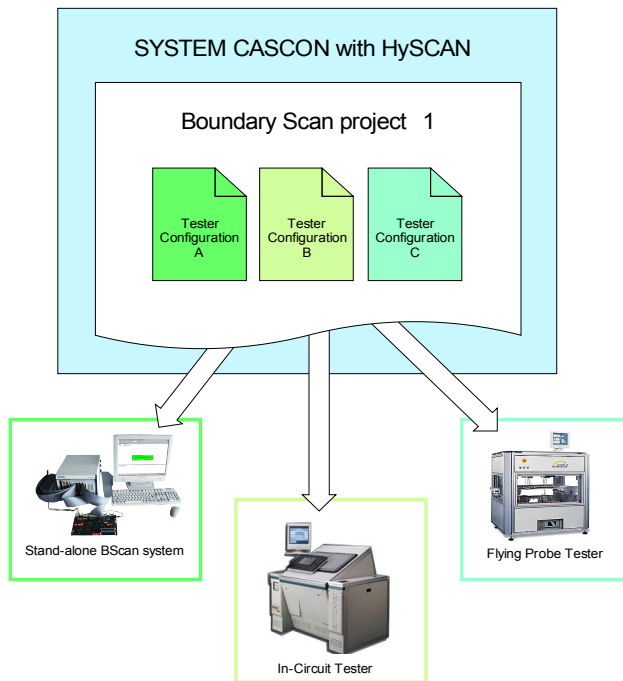


Figure 8 Example project with three HySCAN tester configurations

Conclusions

This article introduced the reader to a new approach of integrating Boundary Scan tools in various board level ATE. The HySCAN concept provides a number of benefits over the use of traditional Boundary Scan I/O modules and previous integrations:

- Portability between various test channel (adapter) types;
- Accurate, non-misleading test results and test coverage statistics;
- UUT and tester I/O resources do not have to be merged for the purpose of test program generation;
- Developed Boundary Scan tests are independent from the test adapter type and the test system they are executed on;

References

- [1] Kenneth P. Parker, *The Boundary Scan Handbook*, 3rd Edition, 2003, Kluwer Academic Publishers, Norwell, MA, 02061, ISBN: 1-4020-7496-4
- [2] GOEPEL Electronics, *White Paper: Design for Testability Guidelines for Boundary Scan Test*, 2005
- [3] H. Ehrenberg, N. Muench, "Leveraging Boundary Scan resources for comprehensive cluster testing", *Proceedings of Board Test Workshop 2004*,
- [4] Jan Heiber, "Extended Boundary Scan Test using hybrid test vectors", *Proceedings of European Board Test Workshop 2005*