

Extend the frontiers of boundary-scan test

Boundary scan has traditionally been difficult to promote as a product-design requirement. But boundary-scan success stories have percolated into the electronic-design community, and the availability of compliant components, design and test tools, and novel ways of applying boundary-scan technology has lowered the barriers to entry.

In the decade following the approval of the IEEE 1149.1 boundary-scan standard, engineers have leveraged it to address test problems throughout the design hierarchy. Once the domain of the PCB test engineer, boundary scan has enjoyed a rise in popularity among other engineering disciplines since the emergence of devices that support in-system programming (ISP).

The ability to reprogram a device without having to physically remove it from a PCB or even to connect probes directly to its I/O pins has simplified life for circuit designers and manufacturing engineers as well as test engineers. Moreover, the increased presence of boundary-scan-compliant devices in circuit boards and modules has helped reduce the cost of test in these units. Nonetheless, to be of value, boundary scan must be integral to a product's design strategy; it can't be applied as a test afterthought. You must take boundary scan into account during chip design, during PCB component selection, and during PCB component placement.

VIT vs. VCCT

Board and system designs that strictly adhere to the IEEE 1149.1 standard can be tested using either or both of two boundary-scan test strategies (**Figure 1**): virtual interconnect test (VIT) and virtual component/cluster test (VCCT). A VIT strategy tests nets that one or more boundary-scan output pins can drive and whose logic state

one or more boundary-scan input pins can capture. Such a configuration lends itself to automatic test-pattern generation (ATPG), high levels of fault coverage, and precise fault isolation. Many nonscan components (such as pull-up/down resistors; series resistors; and noninverting buffers between boundary-scan device pins, nonscan input pins, and nonscan output pins that can be disabled during test) can reside on a VIT-compatible net—ATPG tools can accommodate these as structures, preventing them from interfering with the ATPG process.

You can turn to a VCCT strategy when a combination of scan and nonscan parts on a board share nets. You will need an external set of test vectors to fully test each net and the nonscan component/cluster associated with it. In a VCCT configuration, boundary-scan output pins connected to nonscan component input pins function as virtual-primary-input (VPI) pins. With this arrangement, scan-device pins can drive the nonscan component or cluster, making test

The IEEE 1149.1 standard supports virtual-interconnect and virtual component/cluster test strategies.

David Rolince
Teradyne, North Reading, MA

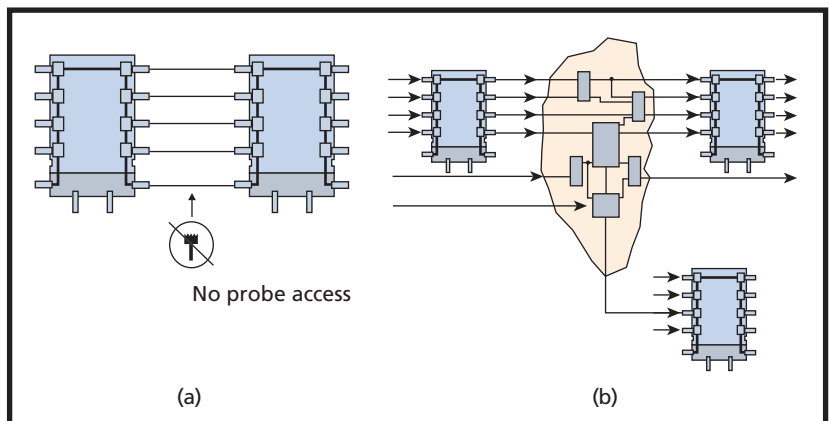


FIGURE 1 (a) VIT is a boundary-scan test strategy employed when scan drivers and receivers reside on the same nets. (b) The VCCT strategy is effective when scan devices drive nonscan devices and when the responses can be captured by other scan devices in the circuit

points unnecessary within the nonscan circuitry. Similarly, nonscan component outputs that connect directly to scan-device inputs function as virtual-primary-output (VPO) pins.

Of the two, the VIT strategy provides the bigger bang for the buck. You can employ

Another advantage of VIT is that it permits test reuse for production and system test. The Cray engineers were able to apply the same VIT patterns for each board to a system consisting of eight boards with 16 CPUs. They could apply tests and analyze the results in less than two minutes. Faulty

boards could be replaced in the factory or the field. Using the same test, the faulty board would be retested to verify the failure and then sent to the repair center.

Several commercially available tools help automate the development of VIT-style tests and can reduce test development to a matter of a few hours. The alternative is generating product-specific self-test code that can take days or weeks for an engineer to write and debug.

strategy, all but the boundary-scan devices that directly participate in the virtual-component or virtual-cluster test can be placed in bypass mode, thereby minimizing the length of the boundary-scan chain and the number of TCK cycles required to apply the test.

- **Use custom, dedicated circuits.** If you design your own boundary-scan test-control circuitry, you can customize the test implementation. This can be advantageous if the boards and system under test include many ASICs and you can easily replicate the custom circuitry in each device. Such custom-controller circuitry is generally has a unique address so it can be distinguished from controller circuitry on the other boards connected to a backplane.

- **Use commercially available scan-chain-controller devices.** Scan-controller devices such as the SCANPSC110F from National Semiconductor and ACT8997 ScanPath Linker from Texas Instruments have been used on many board designs to simplify testing of multiple scan chains and to provide a high degree of boundary-scan-test controllability at the system level. These devices provide a single set of TAP pins at the card edge and can distribute scan vectors to any combination of three (for the PSC110F) or four (for the ACT8997) secondary scan paths on a

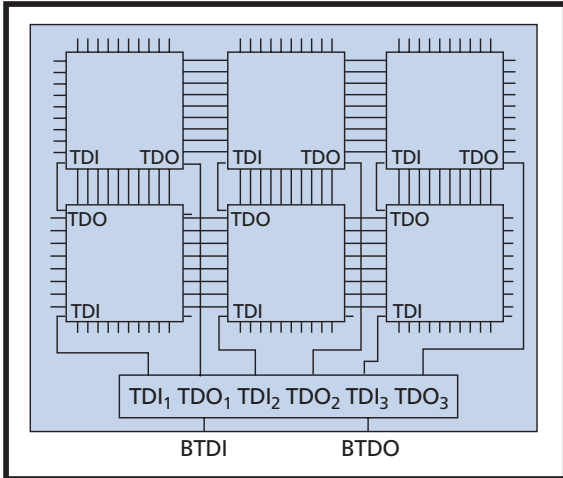


FIGURE 2 Scan-chain control devices provide the flexibility to link individual scan chains or to address each one independently.

VIT only if your circuit designers have followed the highly disciplined design approach required to support it. The reduced test-generation, execution, and debug time make the extra design time worth the effort.

A classic example of the value of a boundary-scan design strategy is the development of the J90 computer by Cray Research. Confronted with developing tests for boards that contained 10 new 350k- to 800k-gate ASIC types, Cray embraced a total boundary-scan design strategy. The rationale was that by designing every device with boundary scan, the engineers could use a VIT strategy to detect 100% of the pin short and open faults that might occur on the board, and designers could test prototype boards and have them repaired quickly to eliminate any structural faults.

This approach detected and isolated all structural faults with one test pass. Before using boundary-scan test, faults were generally detected one at a time. When a fault was detected, the board was sent off for repair and then tested again, which might reveal another fault. This cycle would continue until all the faults were detected, isolated, and repaired.

can extend the advantages of boundary scan to system-level tests of multiple boards connected to a common backplane, whether you're using a VIT or VCCT approach. Boundary-scan test-implementation schemes can take many forms. Here are three examples of boundary-scan design and test implementations that have resulted in high-fault-coverage, easy-to-generate tests:

- **Flatten the design.** Sometimes considered the brute-force approach to system-level boundary-scan test, flattening—treating a multiboard system as if it were a single board—can be the simplest and most effective way to generate and apply tests when the system consists of a small number of boards or modules. For a VIT strategy, the pattern counts will be relatively low. For a VCCT

System-level tests

Although 1149.1 specifies the rules for boundary scan at the board level only, you

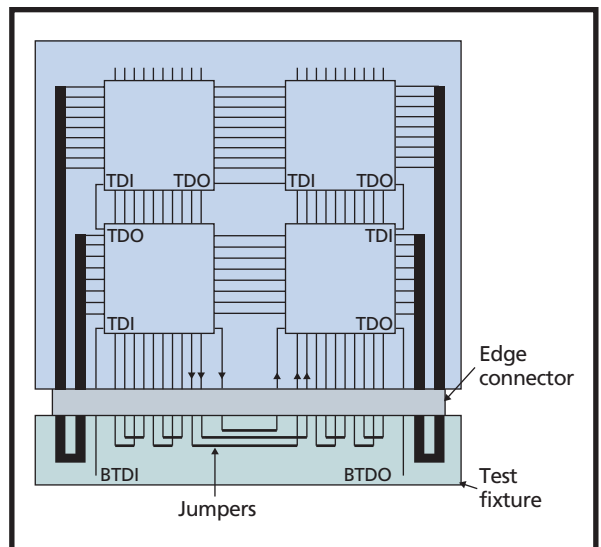


FIGURE 3 If you loop back otherwise unconnected scan outputs to scan inputs in a test fixture, ATPG tools can significantly increase fault coverage.

board or module (Figure 2). For in-system programming, you can target a specific area of a module by programming the scan-path controller to channel serial vectors to a particular secondary scan chain. For test, you can program the scan controller to link all individual chains to form a single chain.

The use of a single boundary-scan chain vs. multiple chains has sparked debate in the design and test communities. Although long, a single chain simplifies test development because it requires only a single set of TAP controls for a board. Multiple scan chains result in shorter individual chains and may help to localize the application of patterns to in-system-programmable devices (a plus for circuit designers), but have the drawback of requiring more I/O pins (each scan chain requires at least four).

To support system-level test, you can individually address each scan-controller device from an external source to a particular board or module on a backplane. In this way, you can reuse at system-level test the test patterns generated for each board or module on the backplane.

Loopback fixturing

The advantage of using VIT for boundary-scan test is it eliminates the need for parallel access to a boundary-scan device. In some board designs, however, boundary-scan device pins are often routed directly to the board edge connector, not to another boundary-scan device pin. With only board I/O access, it would appear that a complete test of the boundary-scan device pins could only be implemented by stimulating the input pins and monitoring output pins with parallel test channels.

You can easily solve this test problem with a little creative fixturing. If you can route boundary-scan device output pins to boundary-scan inputs, then you create a pure boundary-scan net, and automatic test-pattern tools can generate the test and diagnostic data (Figure 3). This technique of looping boundary-scan outputs to boundary-scan inputs can be used to test individual boards or multiple boards that contain a large number of boundary-scan pins terminating at the board edge.

This idea can be extended to system-level testing to test not only the devices on

each board, but also the interconnections between boards. Conducting such tests across a backplane can quickly reveal opens and shorts in the backplane, hung buses, or boards that have come unseated in a card cage.

Many a well-intentioned circuit designer has made boundary-scan test implementation tedious or impossible because a boundary-scan design or board-layout strategy was not carefully thought out. Here are a few tips that can make heroes out of the product design team and result in a significant reduction in product test development and execution time:

- **Provide adequate virtual access.** Typically, boards include a mixture of scan and nonscan devices. If designed and routed carefully, the nonscan devices can be tested as small functional clusters. In such a configuration, boundary-scan output pins connected to nonscan-component input pins can function as virtual primary input (VPI) pins, because the nonscan component or cluster can be driven from the scan-device pins without requiring test points or tester nails. Similarly, nonscan component outputs that connect directly to scan-device inputs function as virtual primary output (VPO) pins. This is the principle behind what has become a very popular way of programming flash memory through the boundary-scan chain.

To ease the implementation of a virtual cluster test, you need to ensure that the cluster is completely controllable by the boundary-scan chain or through direct physical access. For example, to test a RAM cluster, you must be able to set the control pins (read, write, chip enable) to the memory to

the proper states either directly or indirectly through a boundary-scan device or from an external tester pin.

- **Keep control.** Boundary-scan control cells are design structures in a scan device that control boundary-scan output pins.

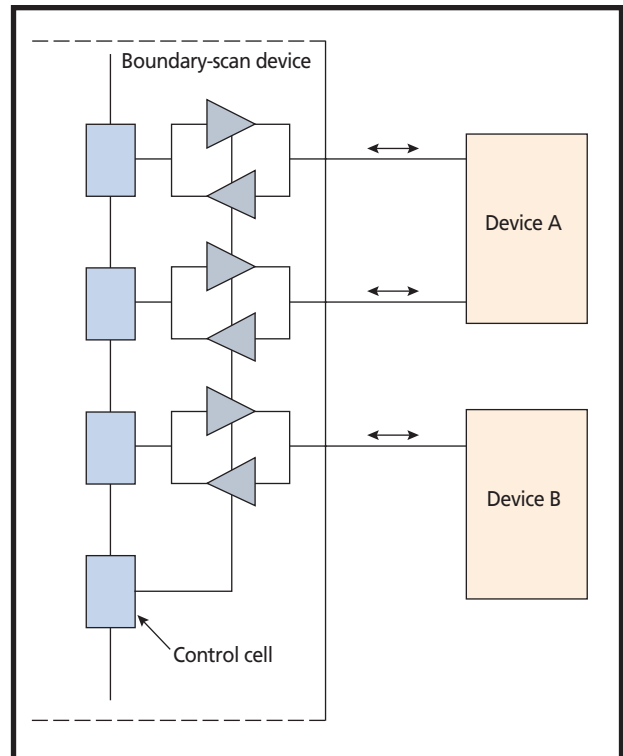


FIGURE 4 If Device B cannot be disabled, no fault coverage can be achieved on any of the three supposedly bidirectional boundary-scan pins, because the scan-control cell, which controls the direction of all three scan pins, must always treat the pins as inputs receiving outputs from devices A and B. This is true even if Device A can be disabled.

They determine, for example, whether a bidirectional device pin functions as an input, functions as an output, or is disabled altogether. In some scan devices, one control cell may control two or more boundary-scan outputs. Although this minimizes scan-design overhead, it also can lead to some debilitating test problems.

Assume, for example, that one control cell controls a bank of three output cells (Figure 4). If one of the output cells is connected to the output of another device that cannot be disabled during test, not only will that pin be untestable, but so will the remaining two pins that are controlled by the same scan-control cell.

The boundary-scan test can never enable the bank of three pins to drive because of the potential for a state conflict at the one net shared by another driving output. Although you might easily overcome this problem during board test by externally disabling the offending output pin, there may be no way to do this at system test, where the only access to the board is through the TAP pins.

Fortunately, few commercially available boundary-scan devices are designed with a single control cell controlling large numbers of output cells. Most FPGAs have one control cell for each output cell. The most common type of device to exhibit the problem of multiple pins controlled by a single control cell is an ASIC for which physical real estate is at a premium.

- **Provide leverage for functional test.**

The inherent reprogrammability of FPGAs through their JTAG port can help provide simplified access to nonscan parts for functional test. An FPGA can be programmed to behave like a “pass-through” device, in which device input pins directly connect internally to device output pins. This approach eliminates the need to functionally “push” test patterns through the FPGA to exercise the pins of nonscan devices connected to the FPGA’s outputs. When functional test of the board in this test configuration is completed, the FPGAs can be reprogrammed into their normal functional state through the boundary-scan control pins. **T&MW**

David Rolince is a product specialist at Teradyne, North Reading, MA. E-mail: drolince@mediaone.net.

BSDL model describes register and control cells

The BSDL model for a scan device defines characteristics of the boundary register cells and identifies the control cell, if applicable. In some devices, a single control cell may control several boundary register cells. In the example below, control cell 164 controls the output of boundary register cells 161, 162, 163, 165, and 166:

```
"161 (BC_6, A(3),  bidir,  0, 164, 0, Z), "&
"162 (BC_6, A(1),  bidir,  0, 164, 0, Z), "&
"163 (BC_6, GBL_L, bidir,  0, 164, 0, Z), "&
"164 (BC_2, *,      control, 0), "&
"165 (BC_2, CI_L,  output3, 0, 164, 0, Z), "&
"166 (BC_2, WT_L,  output3, 0, 164, 0, Z), "&
```

When a single control cell controls a bank of boundary register cells, fault coverage can be limited on those cells if any one of them is connected to another potential driver on their net.

Most FPGAs that support boundary scan have a control cell for each I/O pin, because every pin’s direction is programmable and the flexibility for individual control of every pin is required. The example below shows a portion of the BSDL model for an FPGA; note that a unique control cell controls each boundary register cell:

```
" 9 (BC_1, IO(4),  input,  X), "&
"10 (BC_1, IO(4),  output3, X, 11, 0, Z), "&
"11 (BC_1, *,      control, 0), "&

"12 (BC_1, *,      control, 0), "&
"13 (BC_1, IO(5),  output3, X, 12, 0, Z), "&
"14 (BC_1, IO(5),  input,  X), "&

"15 (BC_1, IO(6),  input,  X), "&
"16 (BC_1, IO(6),  output3, X, 17, 0, Z), "&
"17 (BC_1, *,      control, 0), "&
```

David Rolince

Reproduced with permission from **TEST & MEASUREMENT WORLD** February, 2001
© 2001 by CAHNERS BUSINESS INFORMATION



Teradyne Inc.
Assembly Test Division
2625 Shadelands Dr.
Walnut Creek, CA 94598
TEL 925 932-6900
FAX 925 934-0540

Teradyne Inc.
Assembly Test Division
600 Riverpark Dr.
North Reading, MA 01864
TEL 978 370-2700
FAX 978 370-1440