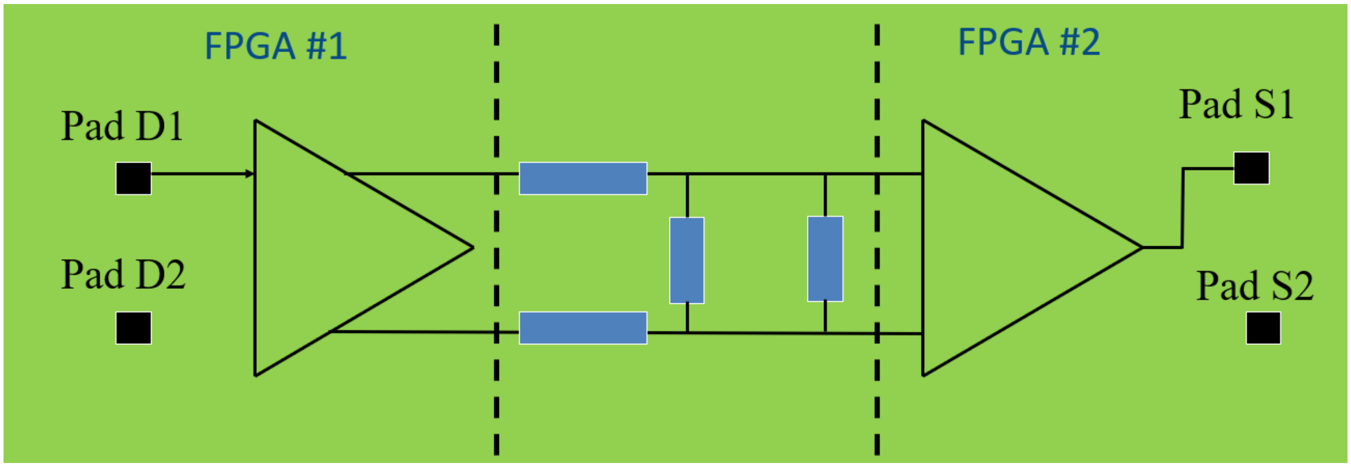
Guidelines for Board Design for Test (DFT) based on Boundary Scan

Volume 2



ASSET™

Michael R. Johnson – Product Manager

Michael R. Johnson serves as Product Manager for ScanWorks Boundary Scan Test (BST) for ASSET InterTech, Inc. He also serves as manager of ASSET's Application Engineering and Professional Services organization. As Product Manager, Michael provides strategic direction for the ScanWorks BST product, ensuring an exceptional customer experience and adherence to current and future IEEE standards requirements. Michael coordinates with ASSET's cross-functional teams such as Marketing, Sales, Support, and Research and Development, to bring ASSET's business goals to fruition.

Before ASSET, Michael's background included roles as a Cellular System Engineer with Nortel Networks and a Hardware Design Engineer with Alcatel USA. While at Nortel Networks, Michael analyzed and provided handoff measurement data for Nortel's mobile and Personal Communications Services (PCS) networks. As a Hardware Design Engineer, Michael designed printed circuit board modules capable of transmitting and receiving optical signals at 622.08 Mbits/s for Alcatel's transport fiber-optic systems.

Michael earned a Bachelor of Science degree with honors in Electrical Engineering from Southern University and A&M College located in Baton Rouge, Louisiana, and a Master of Business Administration degree with an emphasis in Strategic Leadership from Amberton University located in Garland, Texas.

Table of Contents

Table of Figures

Executive Summary

In this eBook, we will examine Design for Test (DFT) guidelines specific to the design of boards to be tested through the Boundary Scan registers of IEEE 1149.1-compliant devices. Implementing Boundary Scan DFT guidelines adds the unique capability of accessing onboard test resources for a non-intrusive test which provides open and short faults coverage. Following DFT guidelines during the board design process makes the board easier to test for defects. DFT guidelines provide an opportunity to increase boar test coverage. As boards move through the manufacturing process, the goal is to reduce the number of defective boards produced. Defects need to be identified quickly so they can be repaired, and processes can be adjusted. Boundary Scan can also be used to test memory devices and for configuration of programmable logic devices (PLD) and flash memory devices. The DFT guidelines contained herein have been assembled over many years of experience by the technical staff of ASSET InterTech and validated across a variety of simple and complex board designs. Volume 2 of this eBook will specifically address DFT guidelines related to interconnect testing, memory interconnect testing, flash programming, and testing with Field Programmable Gate Array (FPGA) devices.

Agenda

- Guidelines for Board DFT based on Boundary Scan Volume 1 Recap
- Guidelines for Board DFT based on Boundary Scan Volume 2
  - Interconnect Testing
  - Memory Interconnect Testing
  - Flash Programming
  - Testing with FPGAs

SourcePoint®
Platform for Software Debug and Trace

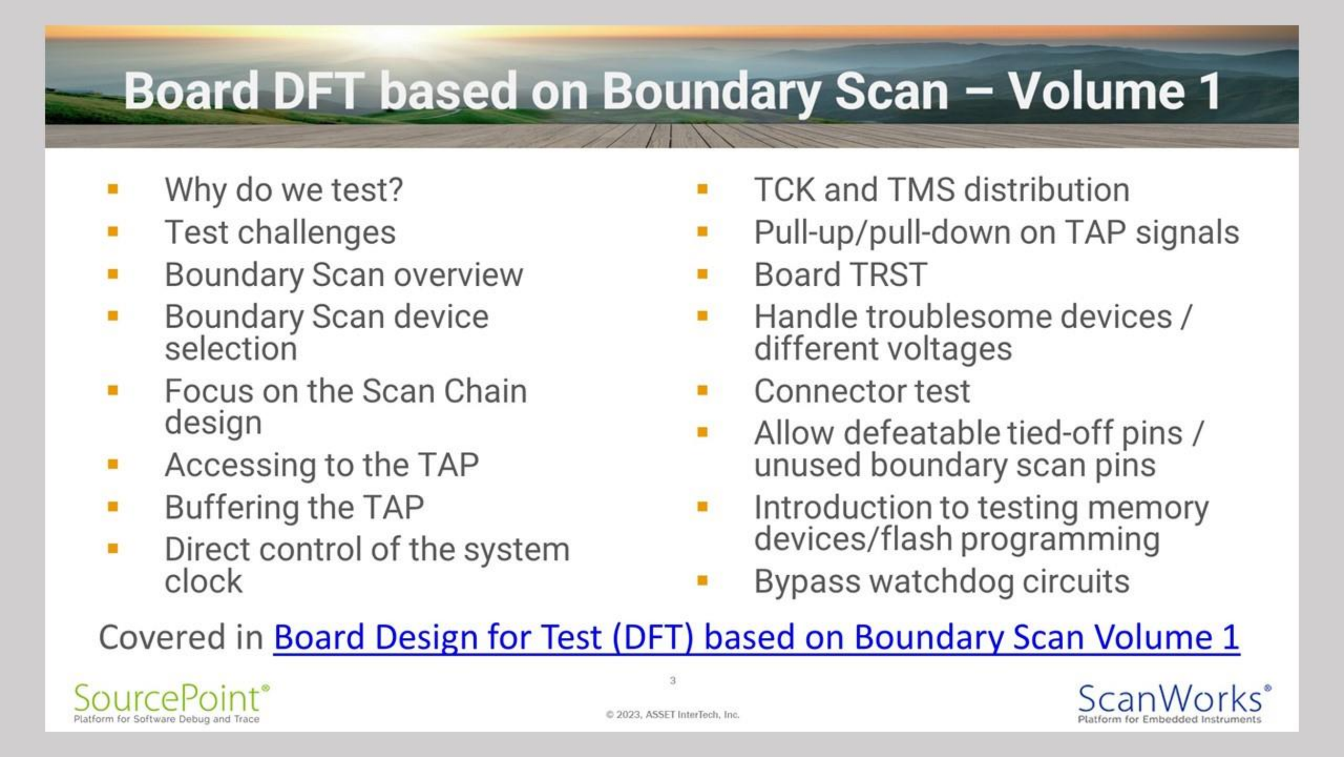
2

© 2023, ASSET InterTech, Inc.

ScanWorks®
Platform for Embedded Instruments

Figure 1: Board DFT based on Boundary Scan – Agenda

Board Level Design for Test (DFT) Guidelines – Volume 1



Board DFT based on Boundary Scan – Volume 1

- Why do we test?
- Test challenges
- Boundary Scan overview
- Boundary Scan device selection
- Focus on the Scan Chain design
- Accessing to the TAP
- Buffering the TAP
- Direct control of the system clock
- TCK and TMS distribution
- Pull-up/pull-down on TAP signals
- Board TRST
- Handle troublesome devices / different voltages
- Connector test
- Allow defeatable tied-off pins / unused boundary scan pins
- Introduction to testing memory devices/flash programming
- Bypass watchdog circuits

Covered in Board Design for Test (DFT) based on Boundary Scan Volume 1

SourcePoint® Platform for Software Debug and Trace

3

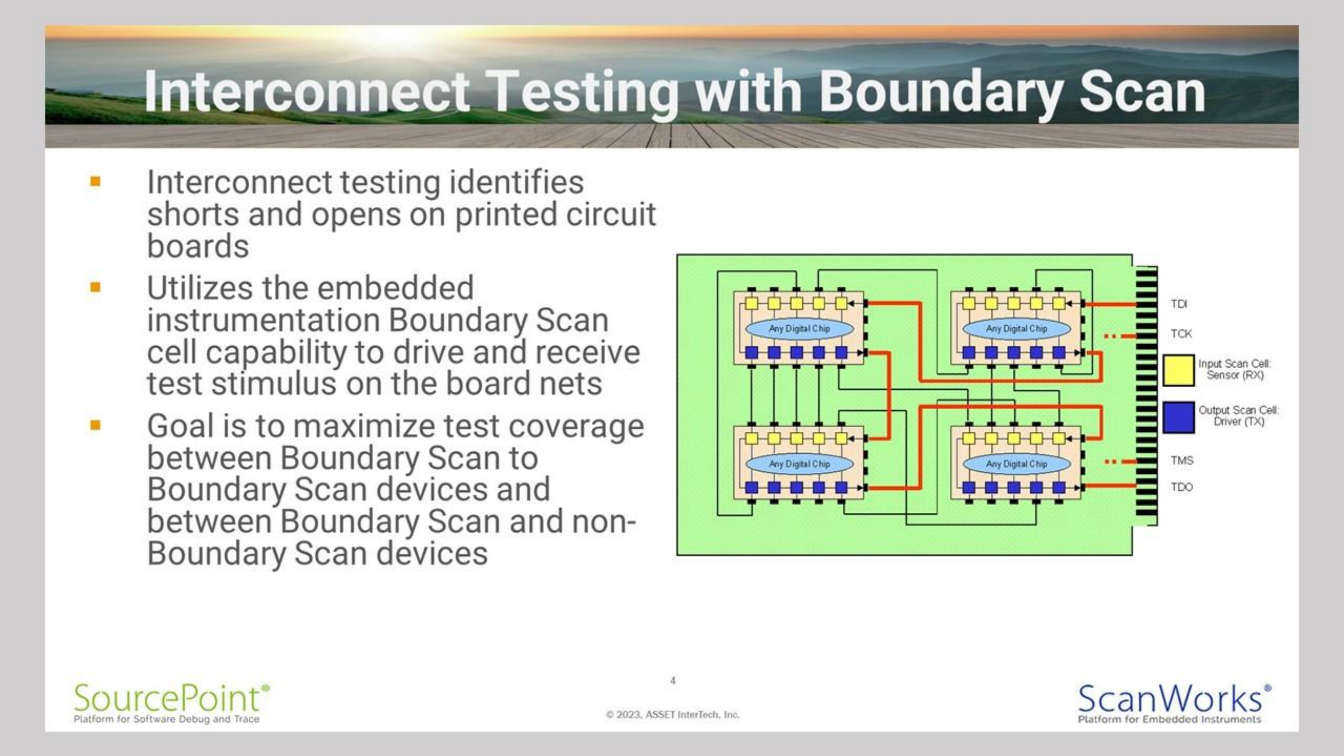
© 2023, ASSET InterTech, Inc.

ScanWorks® Platform for Embedded Instruments

Figure 2: Board DFT based on Boundary Scan – Volume 1

These DFT guidelines were covered in volume 1 of this DFT eBook which can be downloaded from the ASSET website.

Interconnect Testing with Boundary Scan



Interconnect Testing with Boundary Scan

- Interconnect testing identifies shorts and opens on printed circuit boards
- Utilizes the embedded instrumentation Boundary Scan cell capability to drive and receive test stimulus on the board nets
- Goal is to maximize test coverage between Boundary Scan to Boundary Scan devices and between Boundary Scan and non-Boundary Scan devices

SourcePoint® Platform for Software Debug and Trace

4

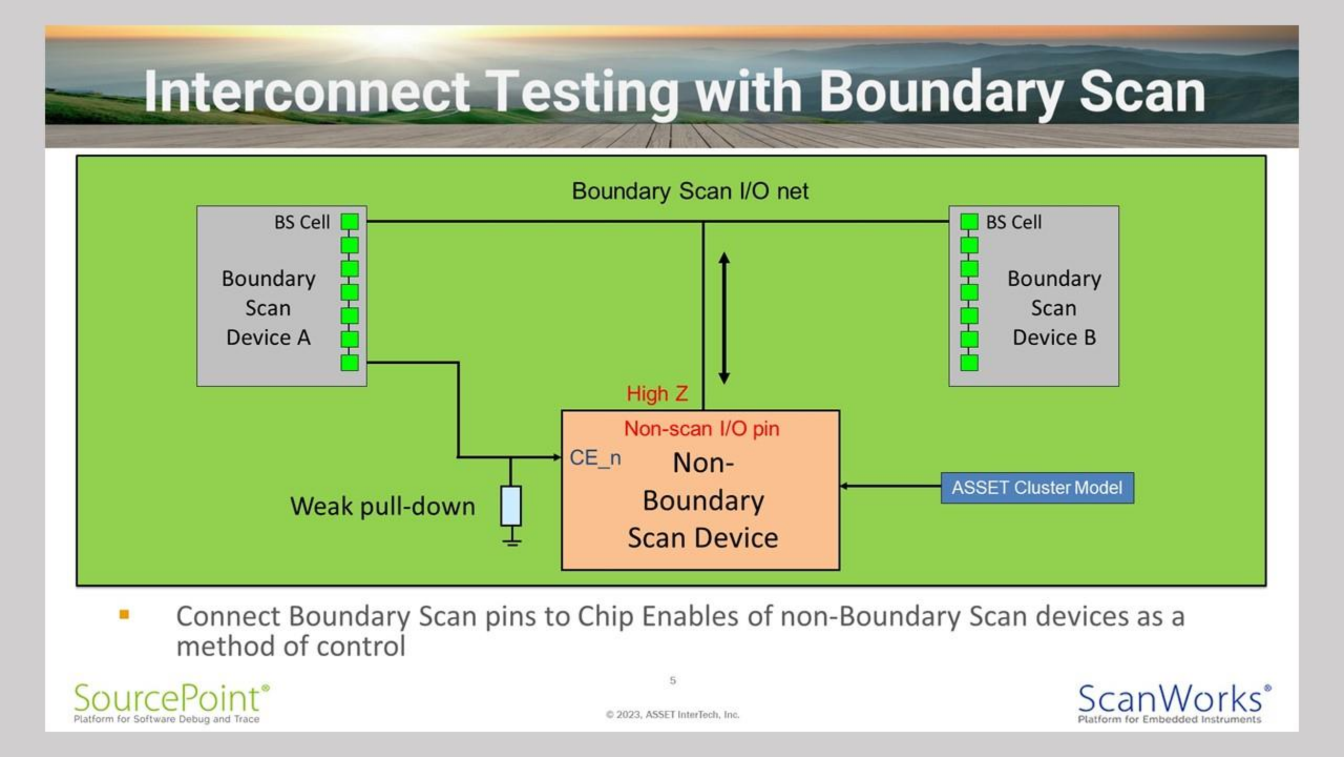
© 2023, ASSET InterTech, Inc.

ScanWorks® Platform for Embedded Instruments

Figure 3: Interconnect Testing with Boundary Scan

Our goal when testing interconnects is to maximize board test coverage. We want to maximize the number of opens and shorts detected with the set of test patterns generated. One obstacle to board test coverage is non-Boundary Scan devices driving signals onto Boundary Scan nets. Increasing board test coverage is the first DFT guideline we will examine.

Interconnect Testing with Boundary Scan – Chip Enables



Interconnect Testing with Boundary Scan

- Connect Boundary Scan pins to Chip Enables of non-Boundary Scan devices as a method of control

SourcePoint® Platform for Software Debug and Trace

5

© 2023, ASSET InterTech, Inc.

ScanWorks® Platform for Embedded Instruments

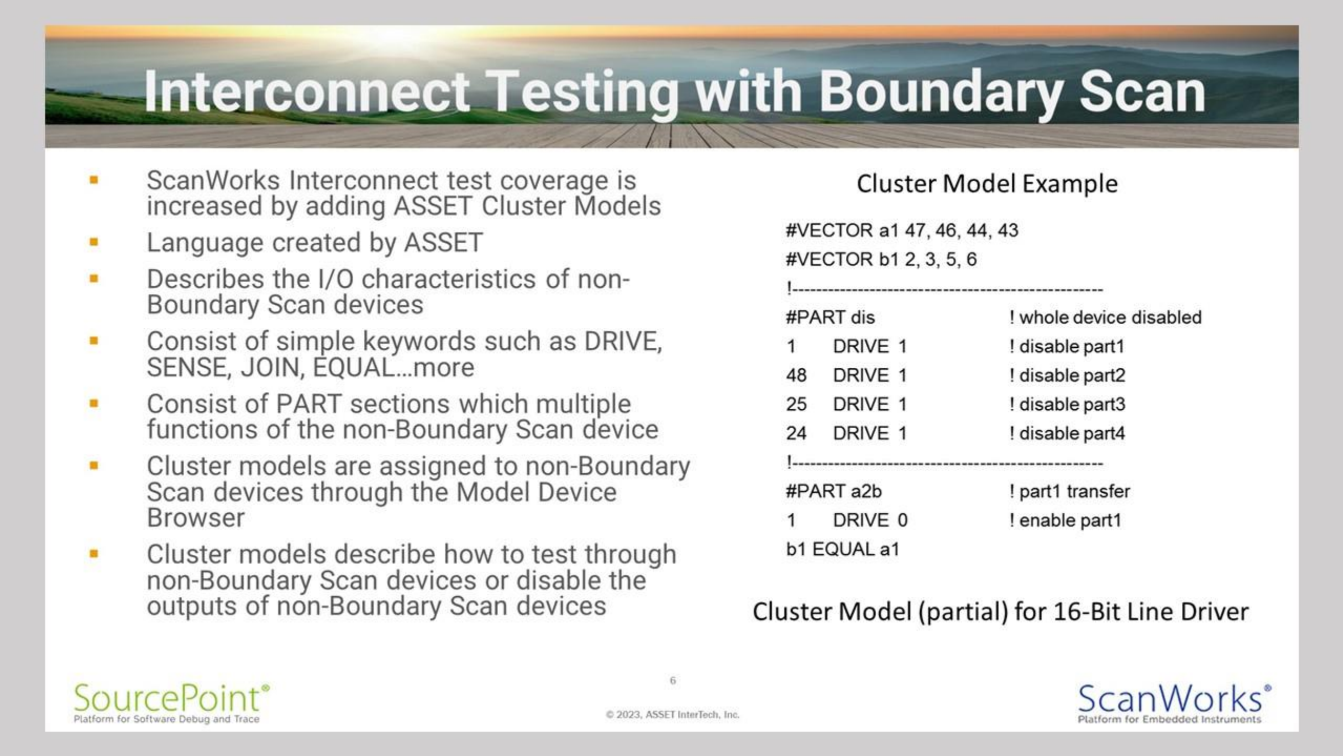
Figure 4: Interconnect Testing with Boundary Scan – Chip Enables

In this scenario, we have a Boundary Scan I/O net connected between device A and device B. Assuming the Boundary Scan pins are bi-directional I/O, a stimulus can be driven from device A to device B then from device B to device A. However, when a non-Boundary Scan device drives signals onto the net, the entire net will not be driven by ScanWorks unless it knows it is safe to do so. To supply this information to ScanWorks, we apply a cluster model to the non-Boundary Scan device. The cluster model describes the I/O characteristics of the device. It also tells ScanWorks how to disable the I/O via the Chip Enable pin so it can be driven safely.

Following the DFT guideline of connecting Boundary Scan pins to Chip Enables of Non-Boundary Scan devices serves as a method of control. We want to disable the outputs of as many of these devices as possible and force the outputs to a high-impedance state.

Not disabling the non-Boundary Scan device could lead to false fails due to outputs from the non-Boundary Scan device corrupting the test signals or it could cause damage as we back-drive into output amplifiers.

Interconnect Testing with Boundary Scan – Cluster Model



Interconnect Testing with Boundary Scan

- ScanWorks Interconnect test coverage is increased by adding ASSET Cluster Models
- Language created by ASSET
- Describes the I/O characteristics of non-Boundary Scan devices
- Consist of simple keywords such as DRIVE, SENSE, JOIN, EQUAL...more
- Consist of PART sections which multiple functions of the non-Boundary Scan device
- Cluster models are assigned to non-Boundary Scan devices through the Model Device Browser
- Cluster models describe how to test through non-Boundary Scan devices or disable the outputs of non-Boundary Scan devices

Cluster Model Example

```
#VECTOR a1 47, 46, 44, 43
#VECTOR b1 2, 3, 5, 6
!--------------------------------------------------
#PART dis                    ! whole device disabled
1     DRIVE  1               ! disable part1
48    DRIVE  1               ! disable part2
25    DRIVE  1               ! disable part3
24    DRIVE  1               ! disable part4
!--------------------------------------------------
#PART a2b                    ! part1 transfer
1     DRIVE  0               ! enable part1
b1 EQUAL a1
```

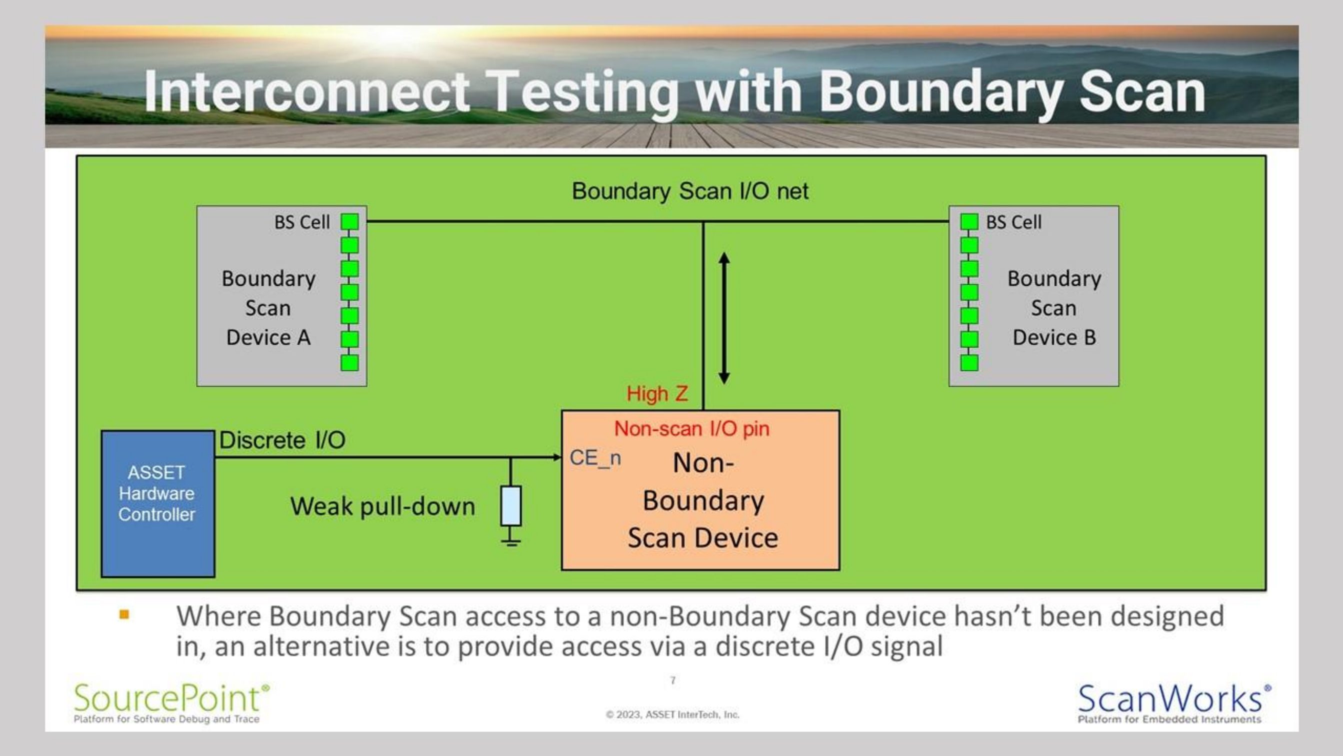
Cluster Model (partial) for 16-Bit Line Driver

SourcePoint®
Platform for Software Debug and Trace
6
© 2023, ASSET InterTech, Inc.
ScanWorks®
Platform for Embedded Instruments

Figure 5: Interconnect Testing with Boundary Scan – Cluster Model

This is an example of an ASSET Cluster model. Cluster models describe the I/O characteristics of non-Boundary Scan devices. Cluster models also can describe how to test through non-Boundary Scan devices. For more information on cluster models, you can refer to the document “bscaninterconnect.pdf” located in the C:\ScanWorks\Doc directory.

Interconnect Testing with Boundary Scan – Discrete I/O



Interconnect Testing with Boundary Scan

Boundary Scan I/O net

BS Cell

Boundary Scan Device A

BS Cell

Boundary Scan Device B

High Z

Non-scan I/O pin

CE_n

Non-Boundary Scan Device

ASSET Hardware Controller

Discrete I/O

Weak pull-down

- Where Boundary Scan access to a non-Boundary Scan device hasn't been designed in, an alternative is to provide access via a discrete I/O signal

SourcePoint® Platform for Software Debug and Trace

7

© 2023, ASSET InterTech, Inc.

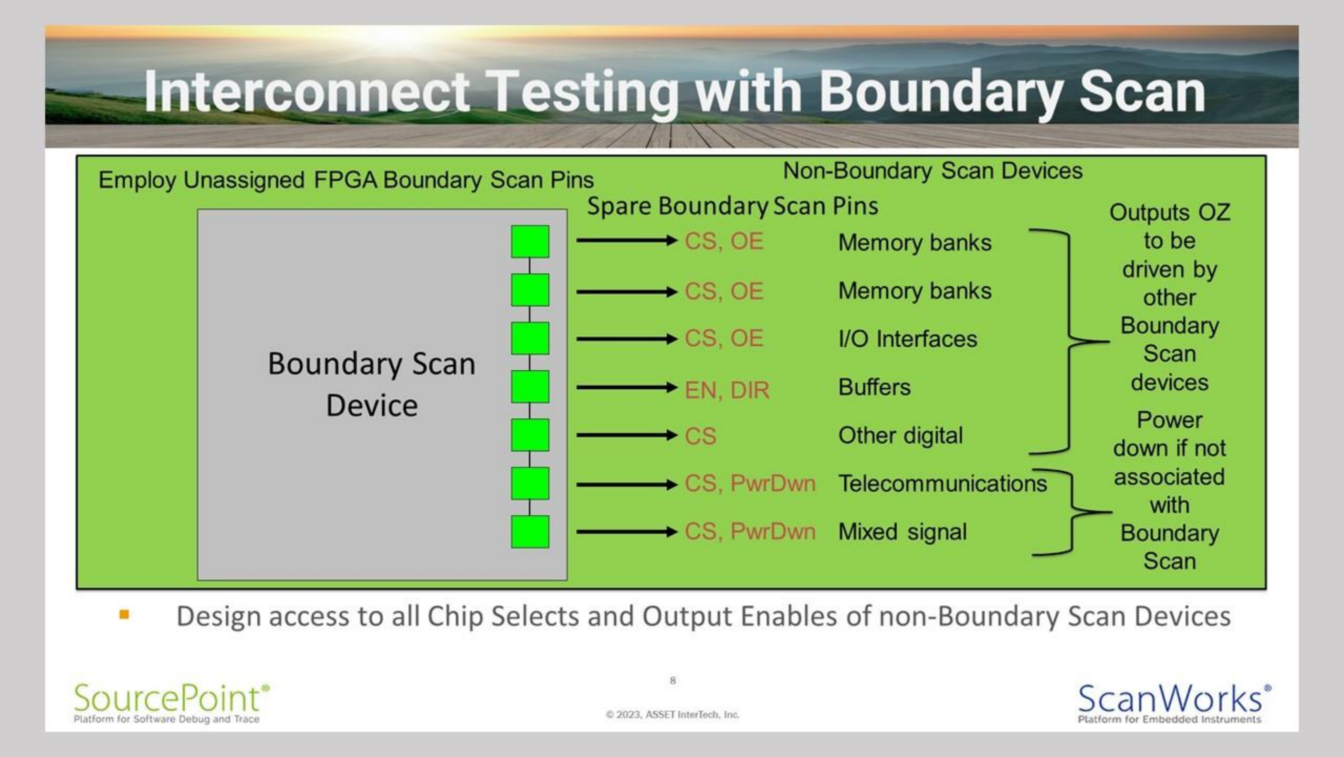
ScanWorks® Platform for Embedded Instruments

Figure 6: Interconnect Testing with Boundary Scan – Discrete I/O

Where Boundary Scan access to a non-Boundary Scan device's Chip Enable has not been designed in via a Boundary Scan pin, an alternative is to provide direct access via a discrete I/O signal. ASSET hardware controllers can control pins independent of the test bus.

Discrete I/O actions can be created to drive pins of non-Boundary Scan devices. In this scenario, a Discrete I/O action can be created to drive the Chip Enable to disable the non-Boundary Scan device outputs.

Interconnect Testing with Boundary Scan – Output Enables



Interconnect Testing with Boundary Scan

Employ Unassigned FPGA Boundary Scan Pins

Non-Boundary Scan Devices

Spare Boundary Scan Pins

Boundary Scan Device

CS, OE → Memory banks

CS, OE → Memory banks

CS, OE → I/O Interfaces

EN, DIR → Buffers

CS → Other digital

CS, PwrDwn → Telecommunications

CS, PwrDwn → Mixed signal

Outputs OZ to be driven by other Boundary Scan devices

Power down if not associated with Boundary Scan

- Design access to all Chip Selects and Output Enables of non-Boundary Scan Devices

8

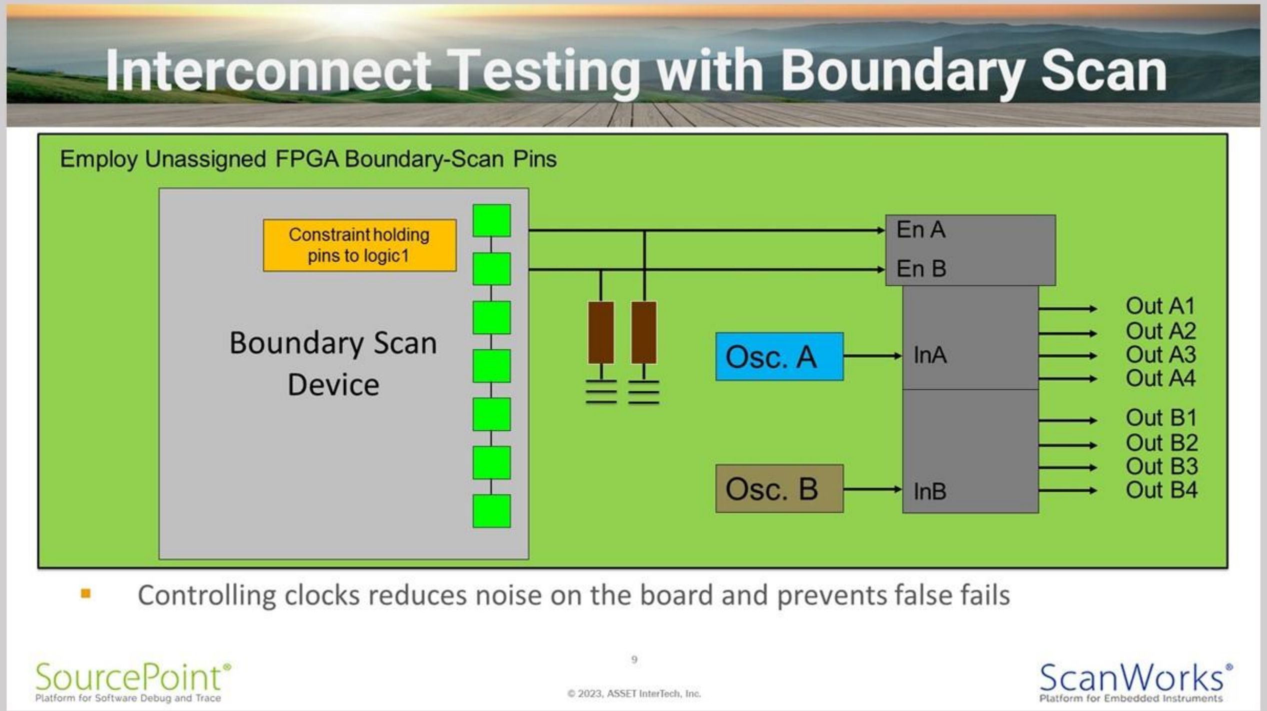
© 2023, ASSET InterTech, Inc.

Figure 7: Interconnect Testing with Boundary Scan – Output Enables

Unassigned FPGA pins can be used for disabling non-scan Boundary Scan devices. Unassigned FPGA pins are normally floating when the system is functioning in the field. However, Boundary Scan can still access and drive these pins during tests.

With bi-directional buffers devices, it can be useful to control the direction. Maybe some nets are tested in one direction and some in the other. Some devices do not have a Chip Select but they do have a power-down Chip Enable pin that tri-states all outputs. The ability to selectively disable devices can be extremely useful during prototype testing.

Interconnect Testing with Boundary Scan – Controlling Clocks



Interconnect Testing with Boundary Scan

Employ Unassigned FPGA Boundary-Scan Pins

- Controlling clocks reduces noise on the board and prevents false fails

SourcePoint® Platform for Software Debug and Trace

9

© 2023, ASSET InterTech, Inc.

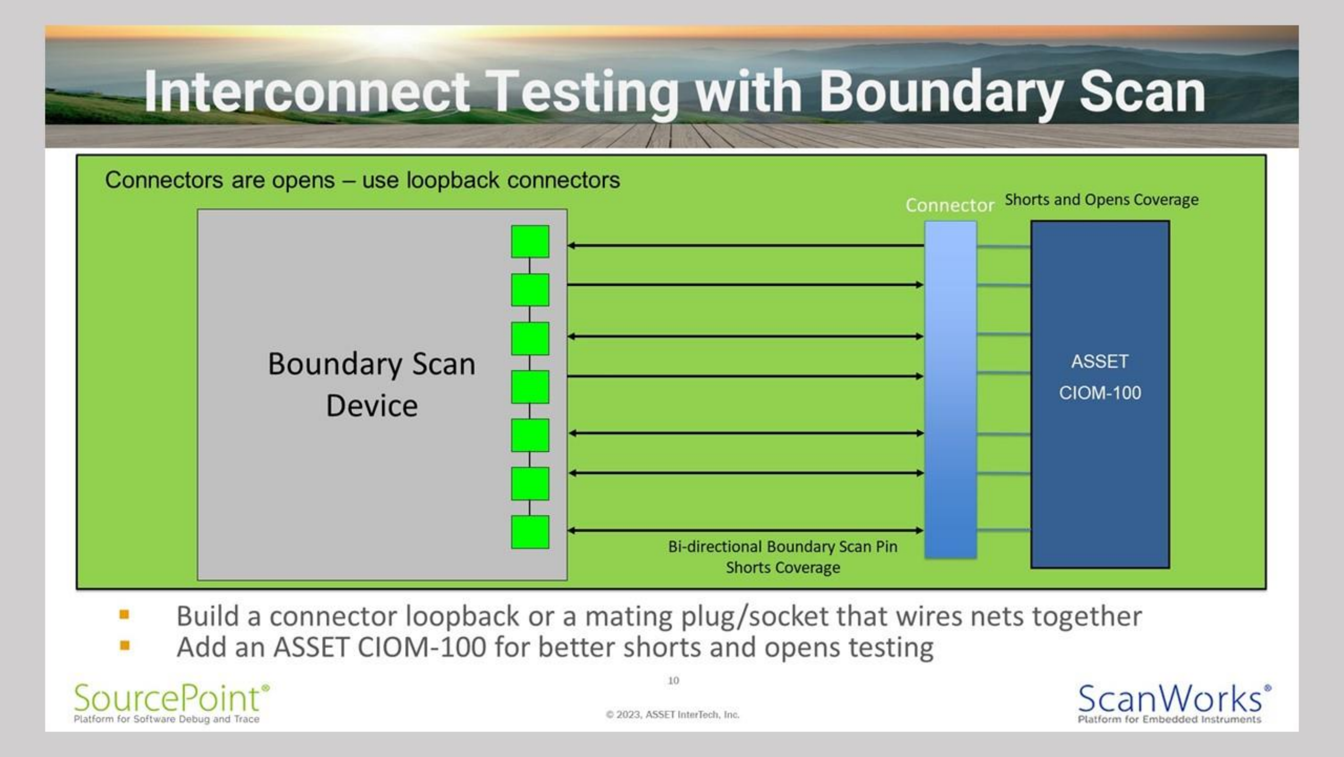
ScanWorks® Platform for Embedded Instruments

Figure 8: Interconnect Testing with Boundary Scan – Controlling Clocks

By controlling clocks, we can reduce noise on the board. Noise can cause false fails on tests. The ability to selectively disable clocks can be extremely useful during prototype testing.

The clock outputs could be disabled by using a ScanWorks constraint on Enable A and/or Enable B. With a constraint, a Boundary Scan pin can be held to a specific level, in this case, a logic 1, throughout the interconnect test.

Interconnect Testing with Boundary Scan – Connectors



Interconnect Testing with Boundary Scan

Connectors are opens – use loopback connectors

Boundary Scan Device
Connector
Shorts and Opens Coverage
ASSET CIOM-100
Bi-directional Boundary Scan Pin Shorts Coverage

- Build a connector loopback or a mating plug/socket that wires nets together
- Add an ASSET CIOM-100 for better shorts and opens testing

SourcePoint®
Platform for Software Debug and Trace
10
ScanWorks®
Platform for Embedded Instruments

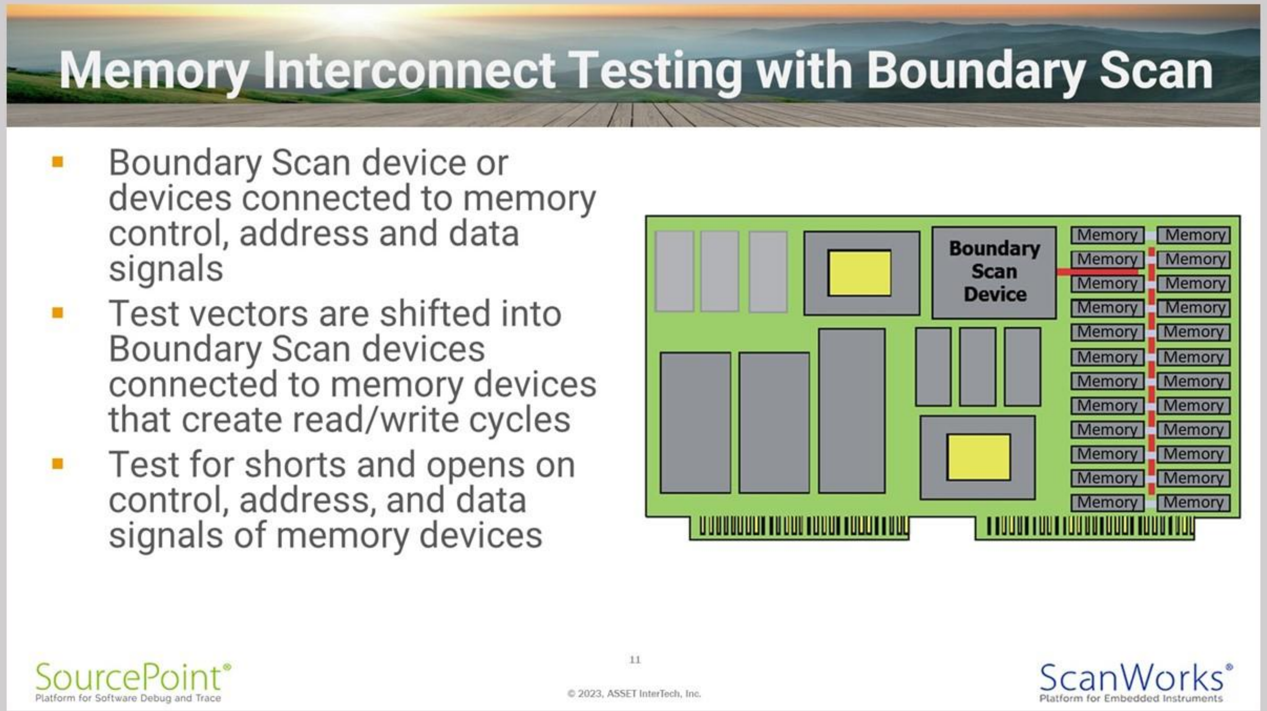
Figure 9: Interconnect Testing with Boundary Scan - Connectors

When nets are routed from a Boundary Scan device to a connector, we cannot test for opens (assuming no pull-up or pull-down resistor on net). In effect, the net is already an open.

A solution is to build a connector loopback, a mating plug/socket that wires nets together during the interconnect tests. The Boundary Scan test drives test signals out on one pin and senses them on one or more other pins. Opens will be detected if the driven pattern is not received.

For the net that does not have a loopback, if the pin is bi-directional, ScanWorks will drive the net and sense the input for shorts. Another way to test through a connector is to utilize a Boundary Scan device. ASSET has a configurable module with 100 I/O that is suited for this purpose.

Memory Interconnect Testing with Boundary Scan



Memory Interconnect Testing with Boundary Scan

- Boundary Scan device or devices connected to memory control, address and data signals
- Test vectors are shifted into Boundary Scan devices connected to memory devices that create read/write cycles
- Test for shorts and opens on control, address, and data signals of memory devices

SourcePoint® Platform for Software Debug and Trace

11

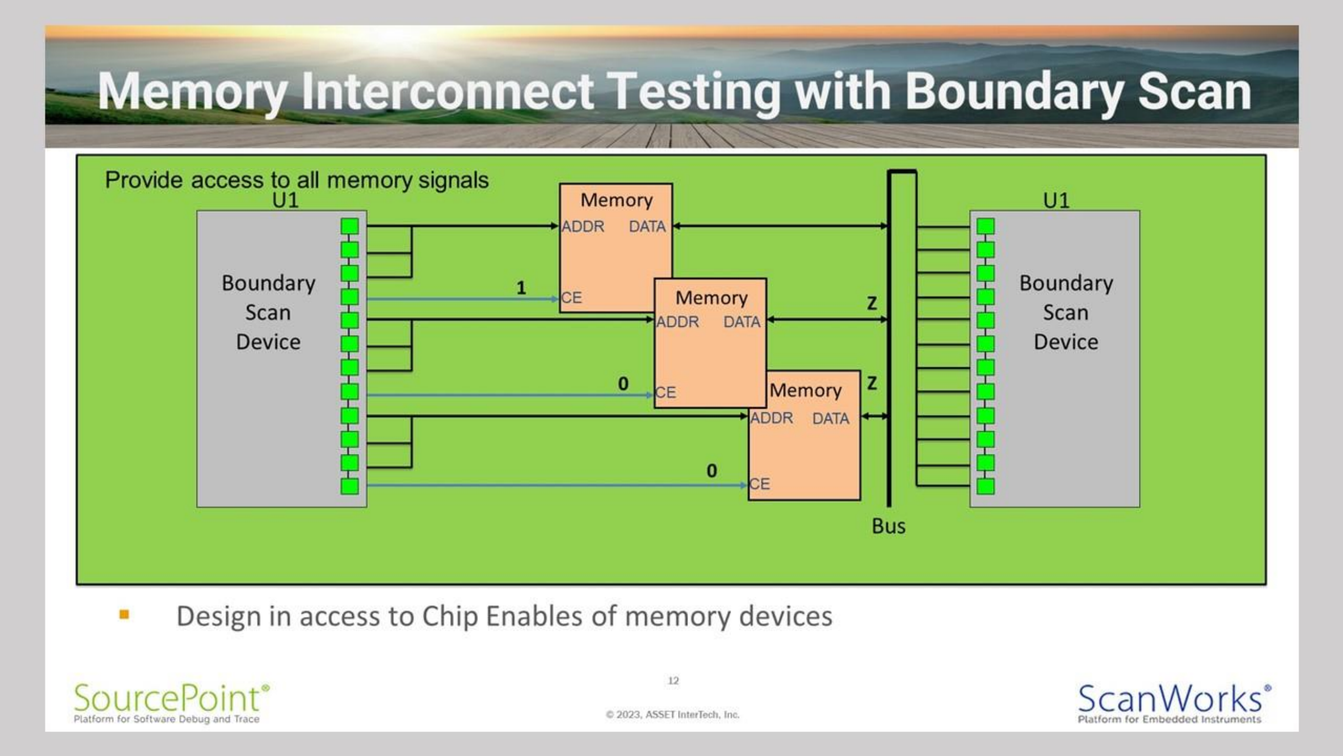
© 2023, ASSET InterTech, Inc.

ScanWorks® Platform for Embedded Instruments

Figure 10: Memory Interconnect Testing with Boundary Scan

Memory devices rarely have Boundary Scan cells. For memory interconnect testing, test patterns are written into memory and then read back from the memory. If the patterns read back match the patterns written, the signals to the memory are structurally intact.

Memory Interconnect Testing with Boundary Scan – Chip Enables



Memory Interconnect Testing with Boundary Scan

Provide access to all memory signals

- Design in access to Chip Enables of memory devices

SourcePoint®
Platform for Software Debug and Trace

12

© 2023, ASSET InterTech, Inc.

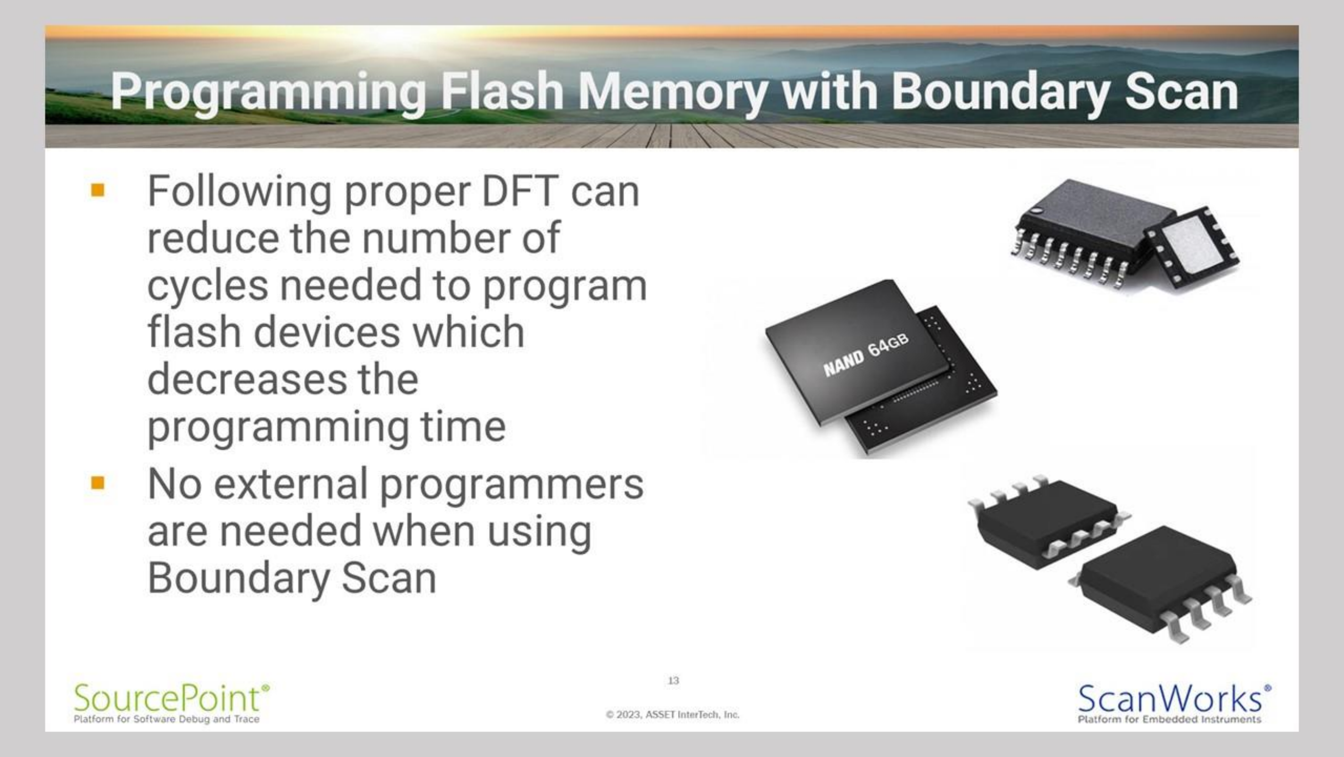
ScanWorks®
Platform for Embedded Instruments

Figure 11: Memory Interconnect Testing with Boundary Scan – Chip Enables

For Boundary Scan memory interconnect testing, Boundary Scan access is needed to all address, data, and control pins. Access should come from the same Boundary Scan device in the same chain.

When testing memory banks, follow the DFT guideline of connecting Boundary Scan pins to Chip Enables of memory devices as a method of control. We want to disable the outputs (the data bus) to prevent false interconnect fails and contention.

Programming Flash Memory with Boundary Scan



Programming Flash Memory with Boundary Scan

- Following proper DFT can reduce the number of cycles needed to program flash devices which decreases the programming time
- No external programmers are needed when using Boundary Scan

SourcePoint®
Platform for Software Debug and Trace

13

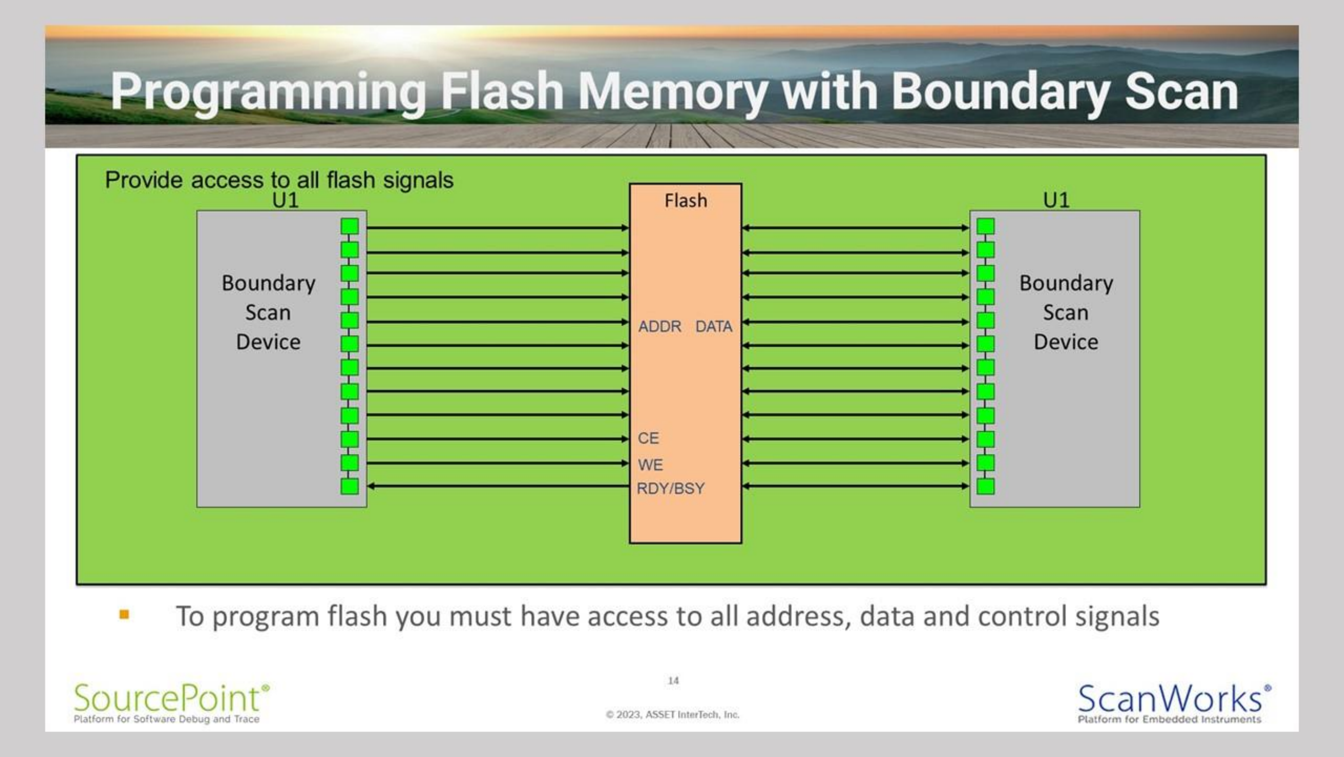
© 2023, ASSET InterTech, Inc.

ScanWorks®
Platform for Embedded Instruments

Figure 12: Programming Flash Memory with Boundary Scan

The time involved in programming a flash device through Boundary Scan can range from minutes to hours. This range in programming times can depend on following appropriate DFT guidelines for flash devices.

Programming Flash Memory with Boundary Scan – All Access



Programming Flash Memory with Boundary Scan

Provide access to all flash signals
U1
Boundary Scan Device
Flash
ADDR
DATA
CE
WE
RDY/BSY
U1
Boundary Scan Device

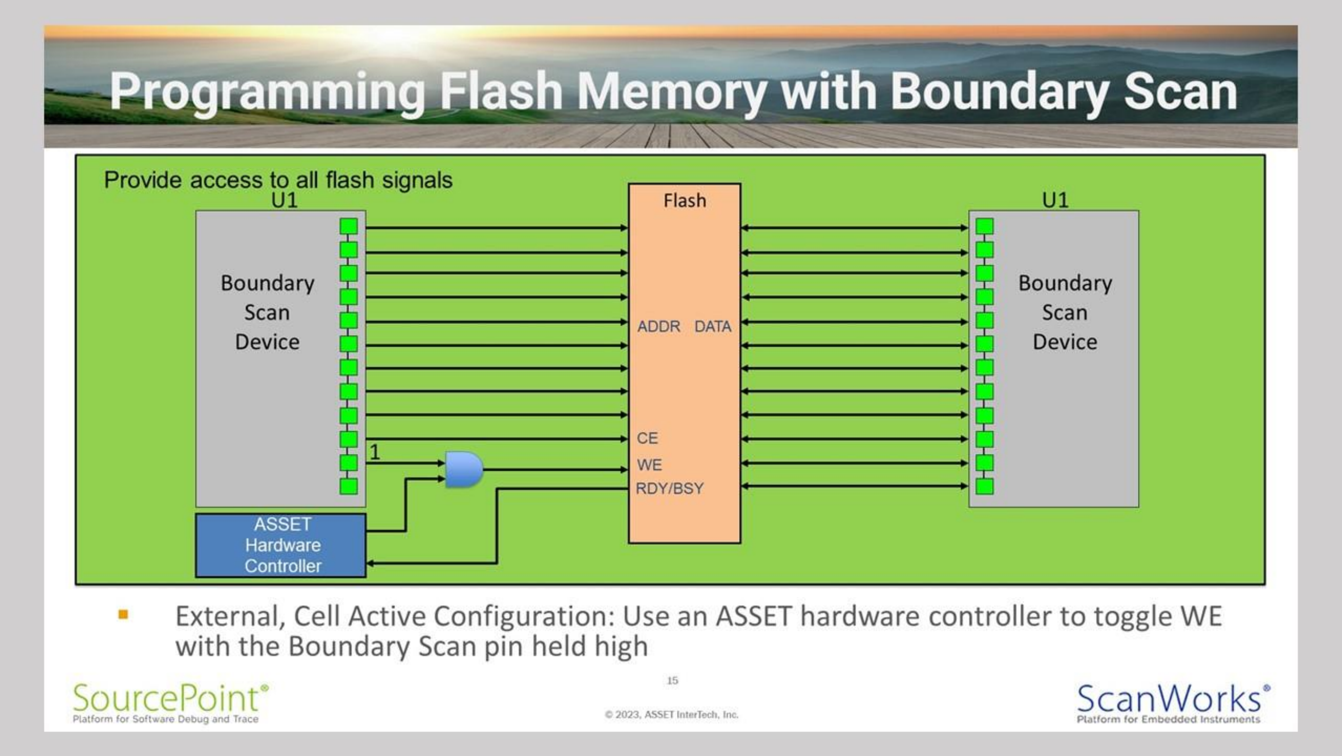
- To program flash you must have access to all address, data and control signals

SourcePoint®
Platform for Software Debug and Trace
14
© 2023, ASSET InterTech, Inc.
ScanWorks®
Platform for Embedded Instruments

Figure 13: Programming Flash Memory with Boundary Scan – All Access

To program flash, you must have access to all address, data, and control signals and it is best, from a DFT perspective, that all access comes from the same Boundary Scan device.

Programming Flash Memory with Boundary Scan – External Cell Active



Programming Flash Memory with Boundary Scan

- External, Cell Active Configuration: Use an ASSET hardware controller to toggle WE with the Boundary Scan pin held high

15

SourcePoint® Platform for Software Debug and Trace

© 2023, ASSET InterTech, Inc.

ScanWorks® Platform for Embedded Instruments

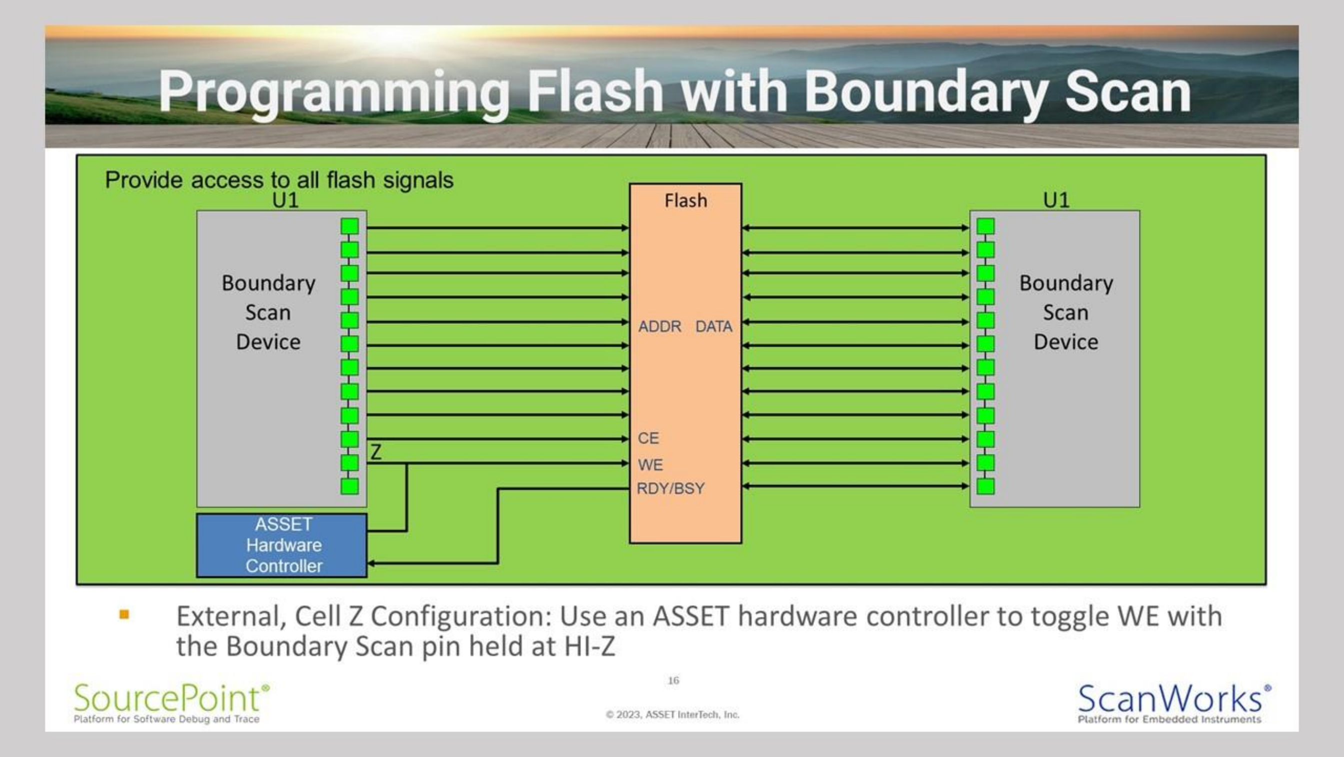
Figure 14: Programming Flash Memory with Boundary Scan – External Cell Active

For Write/Enable it takes multiple scans to toggle WE while holding the address and data steady. If we control Write/Enable with ASSET's hardware controller, the multiple scans are not necessary. We now have one scan for the address and data, then Write/Enable is toggled virtually instantaneously. This can be set up in two ways:

The first is with the External Cell Active configuration. We use the ASSET hardware controller to toggle Write/Enable with the Boundary Scan pin connected to the input of an AND gate held High.

Monitoring Ready/Busy on the ASSET Hardware controller allows ScanWorks to program the next word/byte immediately after the previous one.

Programming Flash with Boundary Scan – External Cell Z



Programming Flash with Boundary Scan

Provide access to all flash signals
U1
Boundary Scan Device
Z
ASSET Hardware Controller
Flash
ADDR
DATA
CE
WE
RDY/BSY
U1
Boundary Scan Device

- External, Cell Z Configuration: Use an ASSET hardware controller to toggle WE with the Boundary Scan pin held at HI-Z

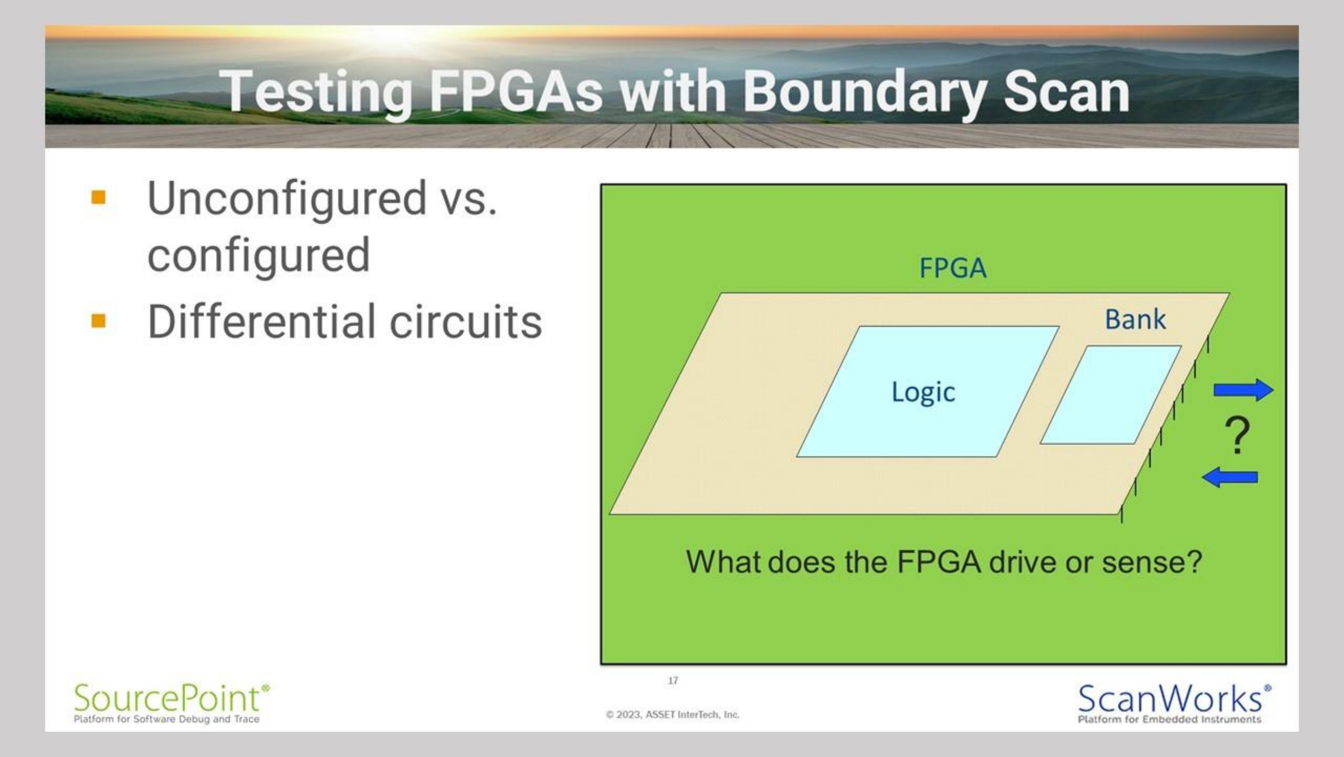
SourcePoint®
Platform for Software Debug and Trace
16
ScanWorks®
Platform for Embedded Instruments

Figure 15: Programming Flash with Boundary Scan – External Cell Z

The second way is with the External Cell Z configuration. We use the ASSET Hardware controller to toggle Write/Enable with the Boundary Scan pin connected to Write/Enable held at HI-Z.

Using discrete I/O signals to control the Write Enable signal and monitor the status of the Ready/Busy can decrease flash programming time by 50% or more.

Testing FPGAs with Boundary Scan



Testing FPGAs with Boundary Scan

- Unconfigured vs. configured
- Differential circuits

17

© 2023, ASSET InterTech, Inc.

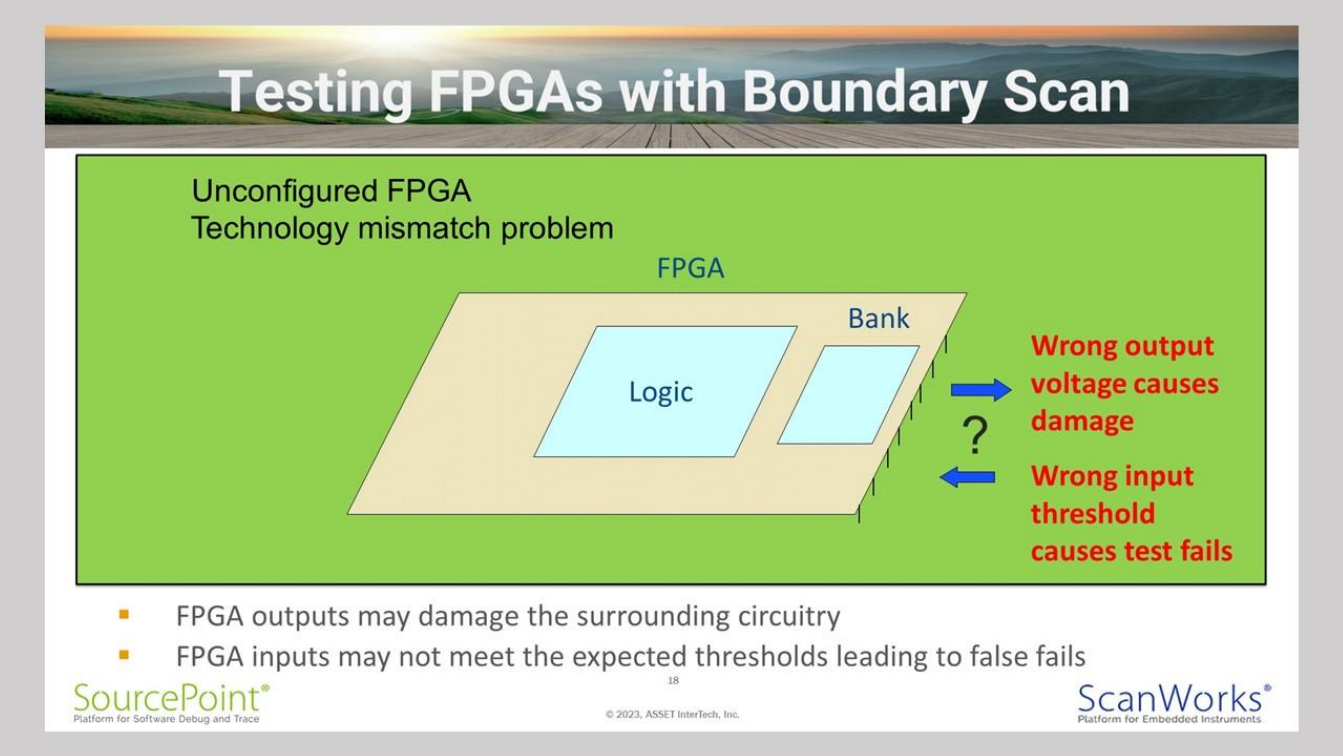
SourcePoint® Platform for Software Debug and Trace

ScanWorks® Platform for Embedded Instruments

Figure 16: Testing FPGAs with Boundary Scan

FPGAs raise several test issues. Some are not strictly DFT, but they are issues that design teams should be aware of. FPGAs can be tested before or after configuration (or both). There are pros and cons to either approach which need to be considered.

Testing FPGAs with Boundary Scan – Technology Mismatch Problem



Testing FPGAs with Boundary Scan

Unconfigured FPGA
Technology mismatch problem

FPGA

Bank

Logic

?

Wrong output voltage causes damage

Wrong input threshold causes test fails

- FPGA outputs may damage the surrounding circuitry
- FPGA inputs may not meet the expected thresholds leading to false fails

18

SourcePoint® Platform for Software Debug and Trace

© 2023, ASSET InterTech, Inc.

ScanWorks® Platform for Embedded Instruments

Figure 17: Testing FPGAs with Boundary Scan – Technology Mismatch Problem

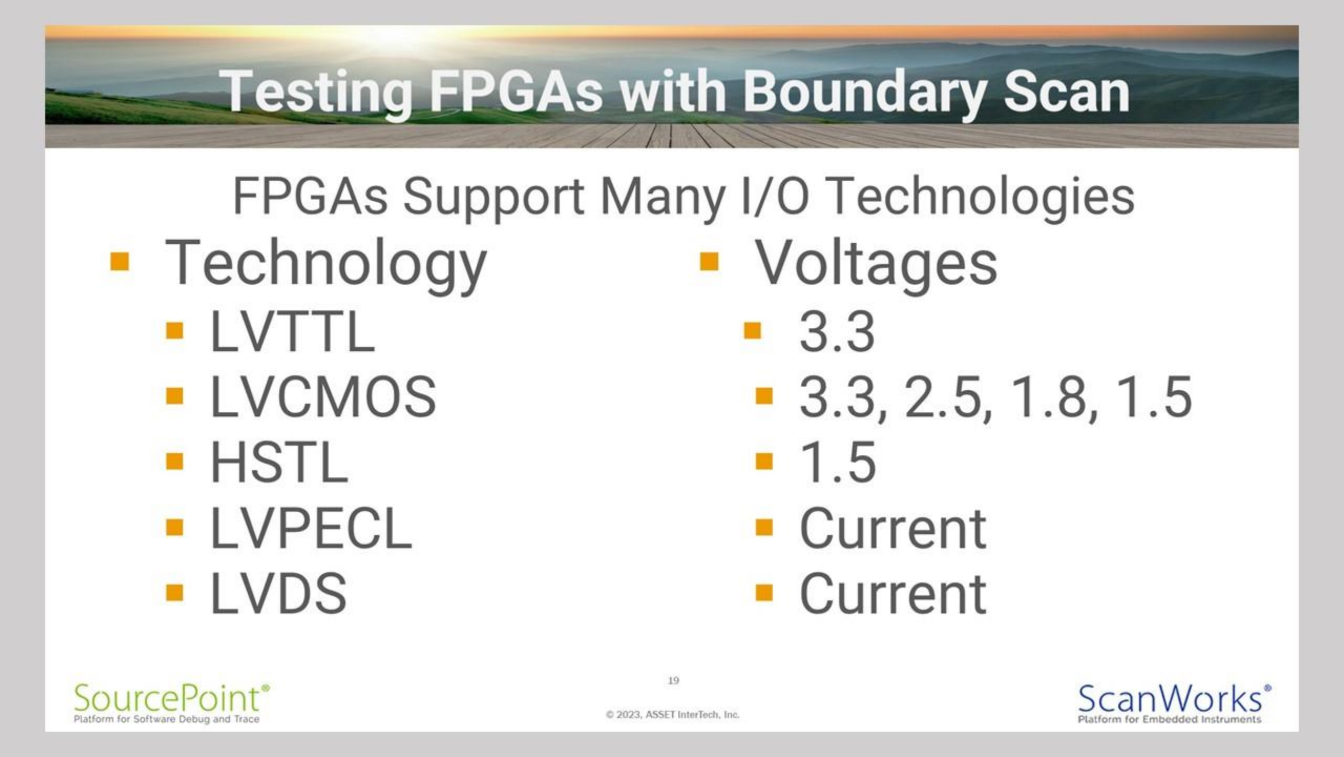
In an unconfigured FPGA, the I/O details have not been programmed in yet. When we run Boundary Scan tests, we drive outputs and sense inputs. But what voltages are driven out and what technologies are we expecting to sense?

The technology mismatch problem raises two issues:

- Outputs may damage the surrounding circuitry (now in practice, we can avoid this by disabling those outputs)
- Inputs will not meet the expected thresholds, leading to false fails in our Boundary Scan tests

So, this is a clear disadvantage of testing with unconfigured FPGAs.

Testing FPGAs with Boundary Scan – Voltages



Testing FPGAs with Boundary Scan

FPGAs Support Many I/O Technologies

- Technology
  - LVTTL
  - LVCMOS
  - HSTL
  - LVPECL
  - LVDS
- Voltages
  - 3.3
  - 3.3, 2.5, 1.8, 1.5
  - 1.5
  - Current
  - Current

19

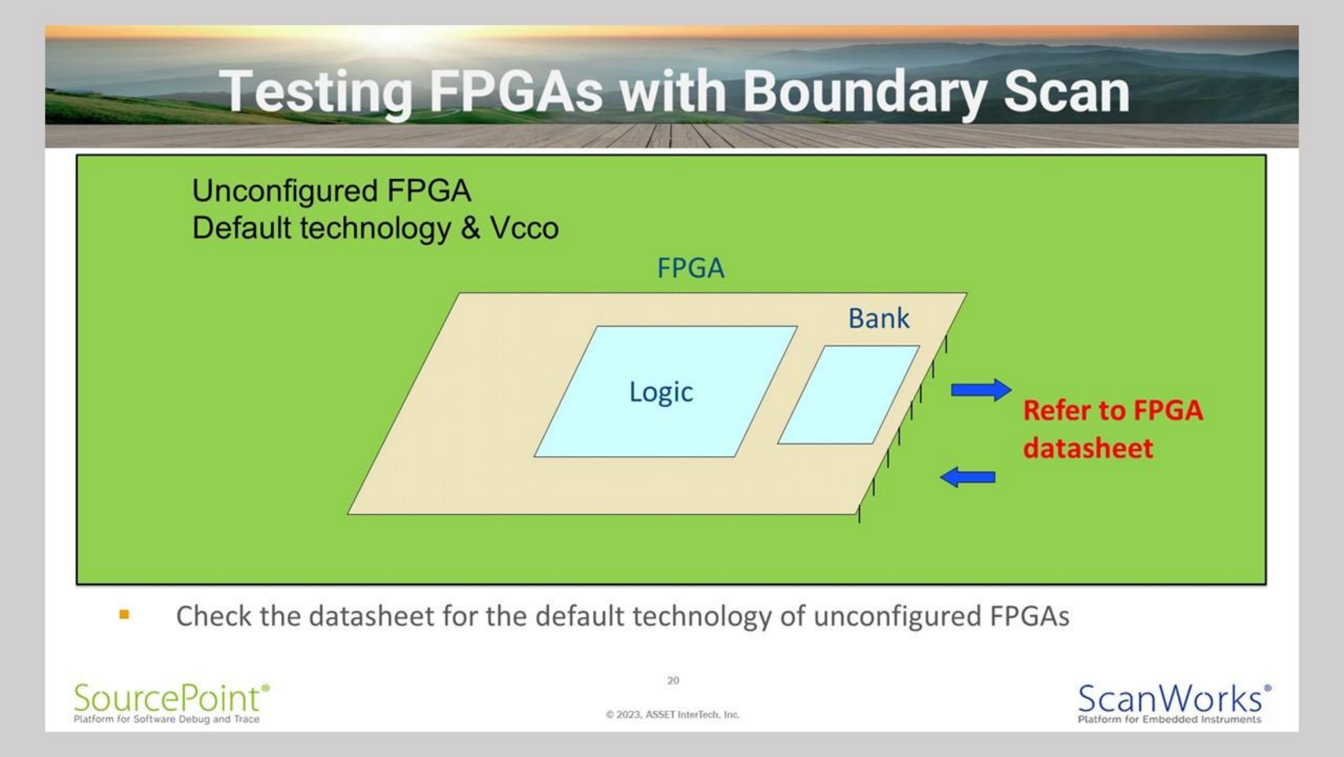
© 2023, ASSET InterTech, Inc.

Figure 18: Testing FPGAs with Boundary Scan - Voltages

The surrounding circuitry will be expecting the technology and voltages of the configured FPGA. There are an entire range of diverse types, many are incompatible with each other.

Which voltage does the unconfigured FPGA use?

Testing FPGAs with Boundary Scan – Default Technology



Testing FPGAs with Boundary Scan

Unconfigured FPGA
Default technology & Vcco
FPGA
Bank
Logic
Refer to FPGA datasheet

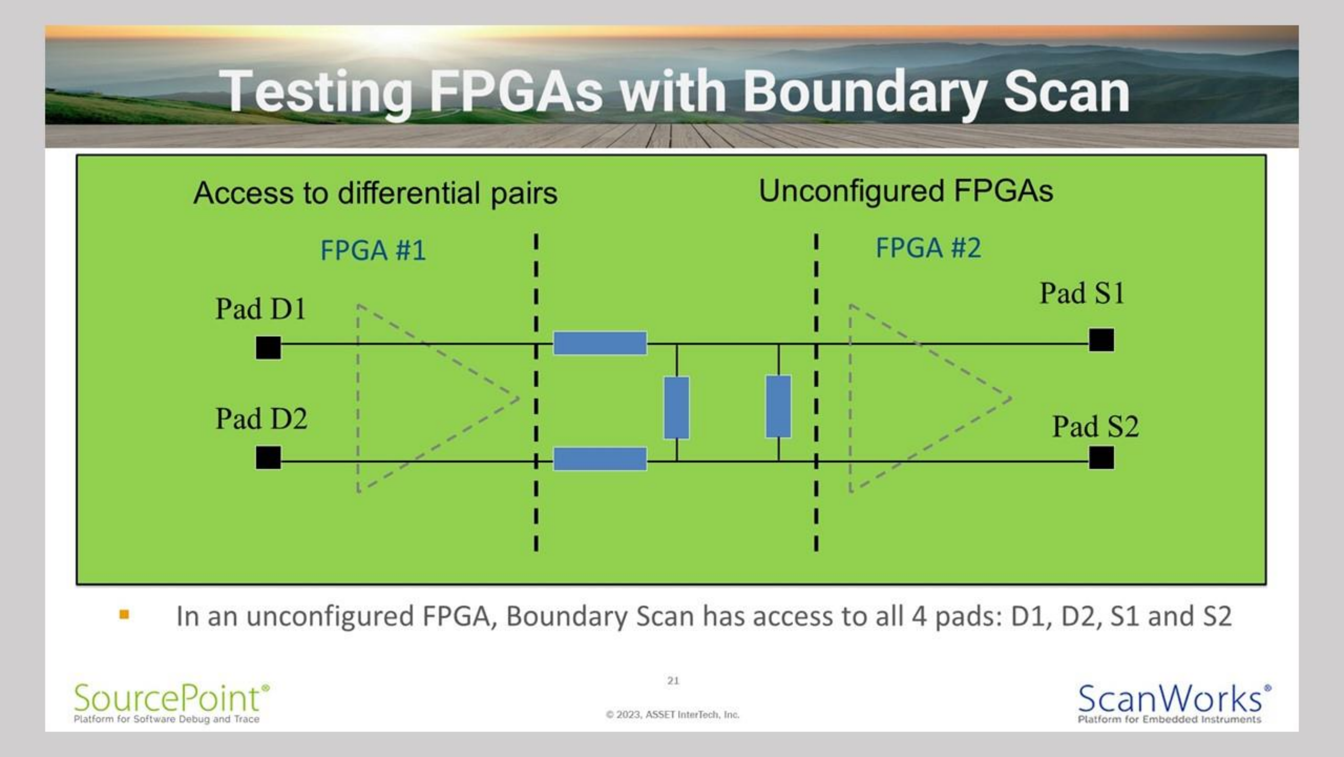
- Check the datasheet for the default technology of unconfigured FPGAs

SourcePoint®
Platform for Software Debug and Trace
20
© 2023, ASSET InterTech, Inc.
ScanWorks®
Platform for Embedded Instruments

Figure 19: Testing FPGAs with Boundary Scan – Default Technology

Due to the considerable number of FPGAs on the market, check the datasheet for the default technology of your unconfigured FPGA.

Testing FPGAs with Boundary Scan – Unconfigured


Testing FPGAs with Boundary Scan

Access to differential pairs
Unconfigured FPGAs
FPGA #1
FPGA #2
Pad D1
Pad D2
Pad S1
Pad S2

- In an unconfigured FPGA, Boundary Scan has access to all 4 pads: D1, D2, S1 and S2

SourcePoint®
Platform for Software Debug and Trace
21
ScanWorks®
Platform for Embedded Instruments

© 2023, ASSET InterTech, Inc.

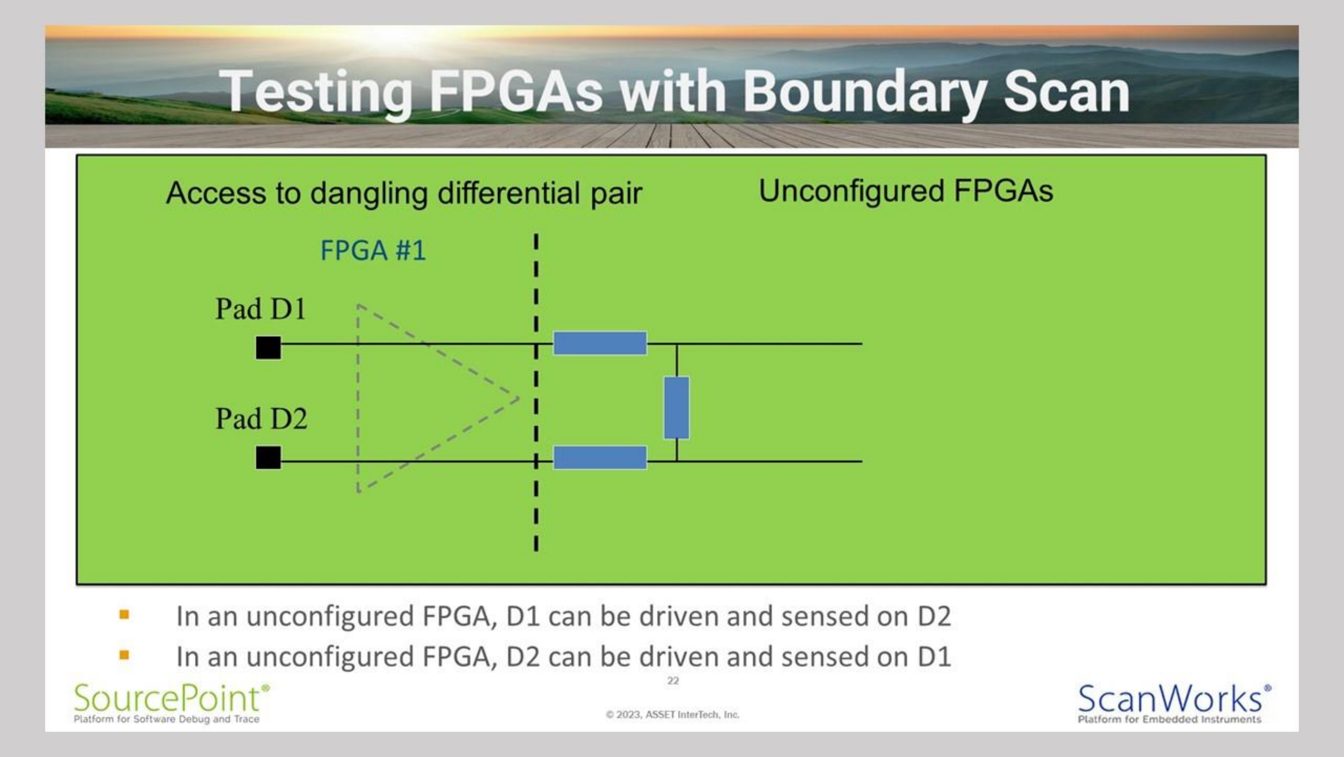
Figure 20: Testing FPGAs with Boundary Scan - Unconfigured

This diagram shows an FPGA on the left with a differential pair output. On the right is another FPGA with the input differential pair. In between are the LVDS resistor network terminating the drive and receive ends. Notice that there are two pins associated with the pair and each has its own pad on the silicon. The differential drive and receive amps are shown in outline because, in an unconfigured FPGA, they are not connected to the pair yet.

Boundary Scan has access to all 4 pads: D1, D2, S1 and S2.

The resistors are digitally transparent. We can drive patterns from D1 and sense them at D2, S1 and S2. In fact, we can drive from each pad in turn, sensing at the other 3. Thus, we can test these nets and detect and diagnose interconnect faults that could occur.

Testing FPGAs with Boundary Scan – Dangling Pair Transmitter



Testing FPGAs with Boundary Scan

Access to dangling differential pair

Unconfigured FPGAs

- In an unconfigured FPGA, D1 can be driven and sensed on D2
- In an unconfigured FPGA, D2 can be driven and sensed on D1

22

SourcePoint® Platform for Software Debug and Trace

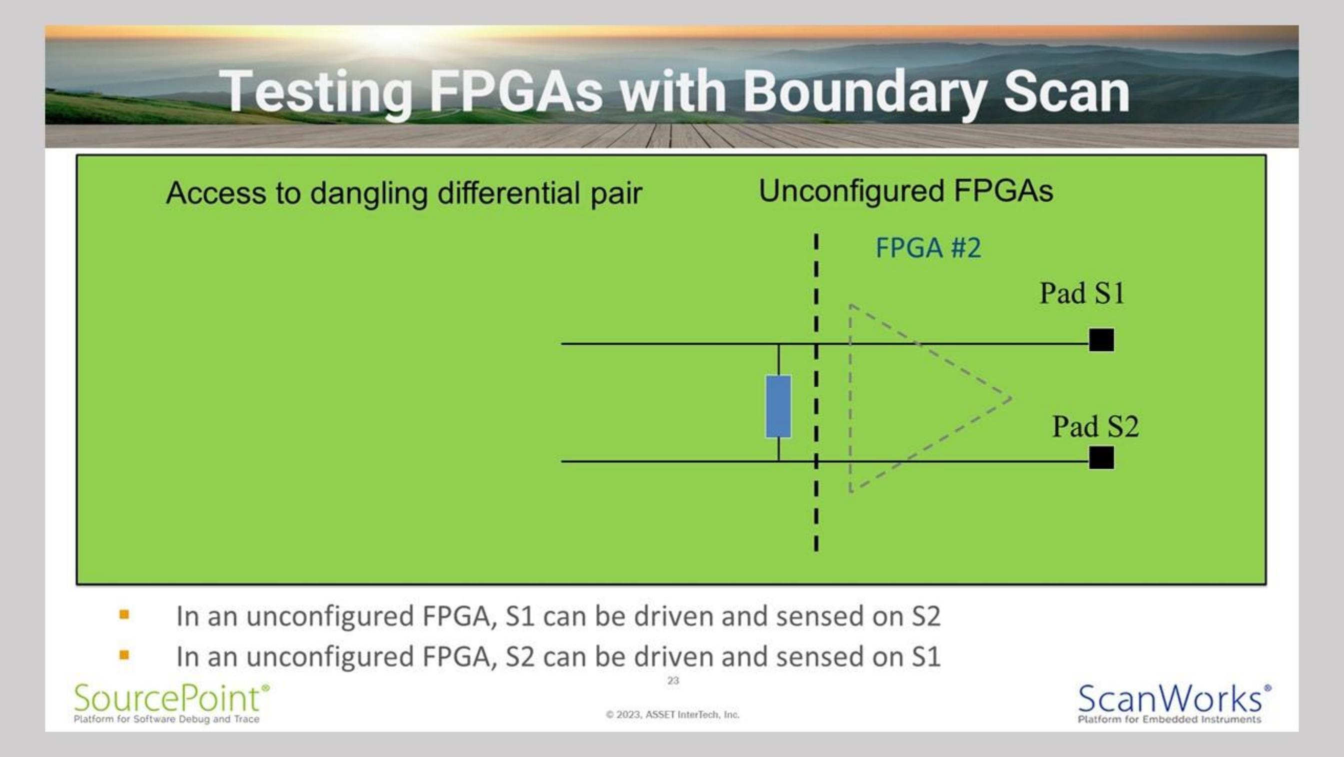
© 2023, ASSET InterTech, Inc.

ScanWorks® Platform for Embedded Instruments

Figure 21: Testing FPGAs with Boundary Scan – Dangling Pair Transmitter

Even if we only have access to the drive end of the differential circuit, we can drive D1 and sense on D2 and vice versa. Before configuration, these are not "drive" or "sense" ends but bi-directional.

Testing FPGAs with Boundary Scan – Dangling Pair Receiver



Testing FPGAs with Boundary Scan

Access to dangling differential pair

Unconfigured FPGAs

- In an unconfigured FPGA, S1 can be driven and sensed on S2
- In an unconfigured FPGA, S2 can be driven and sensed on S1

23

SourcePoint® Platform for Software Debug and Trace

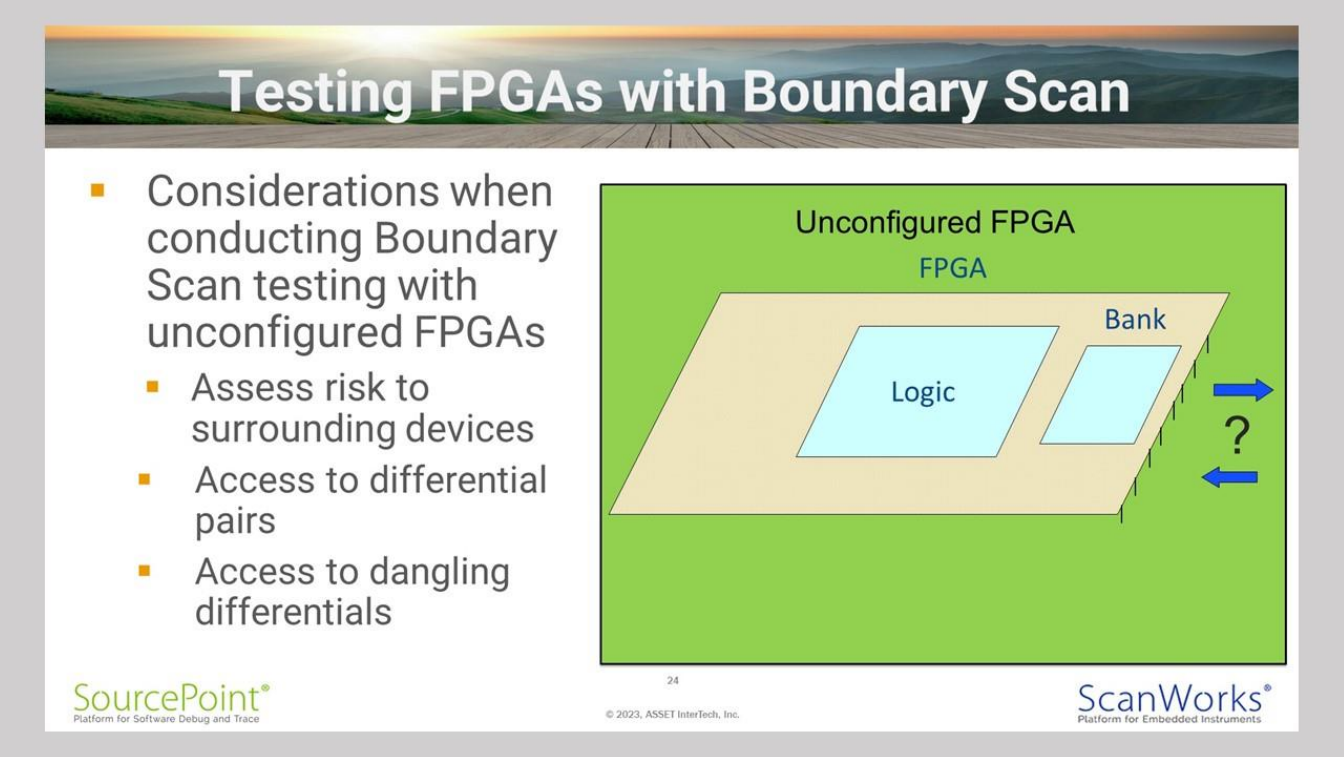
© 2023, ASSET InterTech, Inc.

ScanWorks® Platform for Embedded Instruments

Figure 22: Testing FPGAs with Boundary Scan – Dangling Pair Receiver

The same goes for the other end of the differential pair.

Testing FPGAs with Boundary Scan – Unconfigured Considerations



Testing FPGAs with Boundary Scan

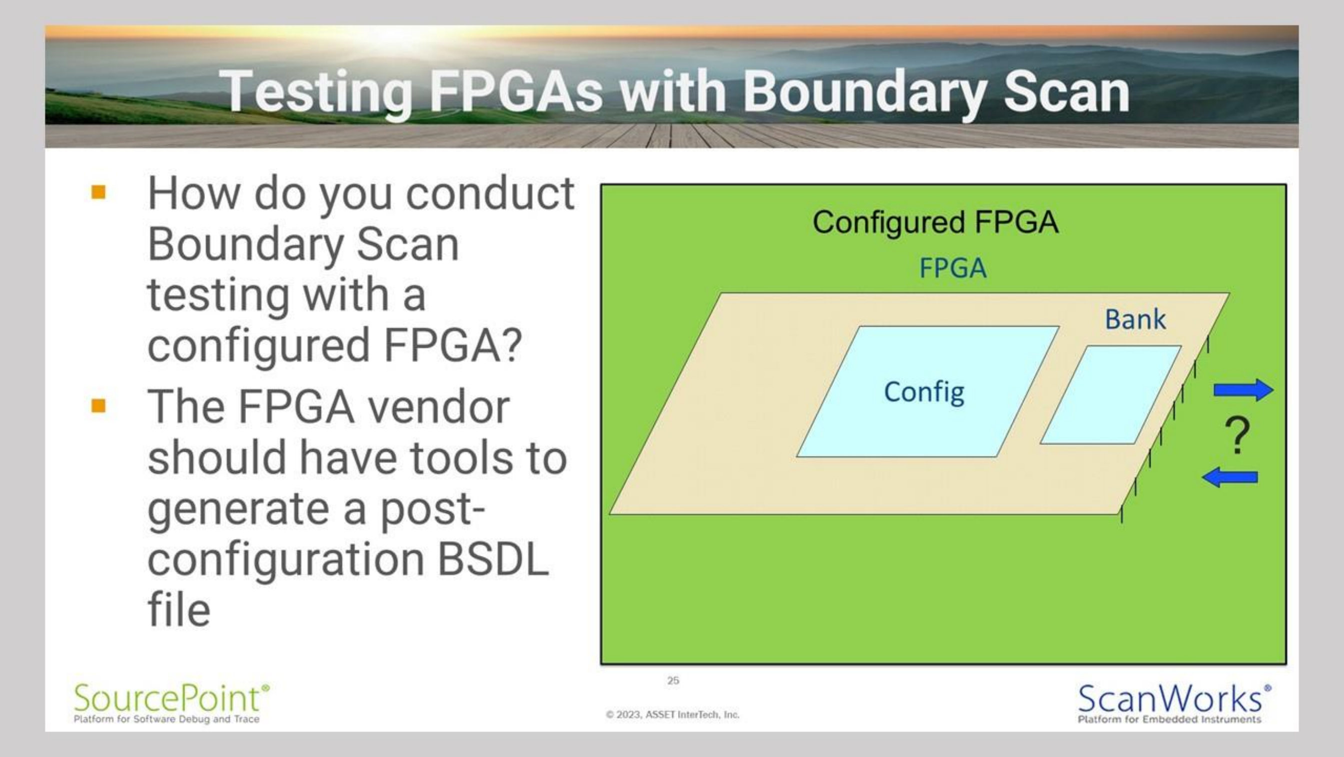
- Considerations when conducting Boundary Scan testing with unconfigured FPGAs
  - Assess risk to surrounding devices
  - Access to differential pairs
  - Access to dangling differentials

24

Figure 23: Testing FPGAs with Boundary Scan – Unconfigured Considerations

To summarize, the considerations you should keep in mind when conducting Boundary Scan testing with unconfigured FPGAs are that you should assess risk to surrounding devices and consider the amount of test coverage that will be available on differential signal nets.

Testing FPGAs with Boundary Scan – Configured



Testing FPGAs with Boundary Scan

- How do you conduct Boundary Scan testing with a configured FPGA?
- The FPGA vendor should have tools to generate a post-configuration BSDL file

SourcePoint® Platform for Software Debug and Trace

25

© 2023, ASSET InterTech, Inc.

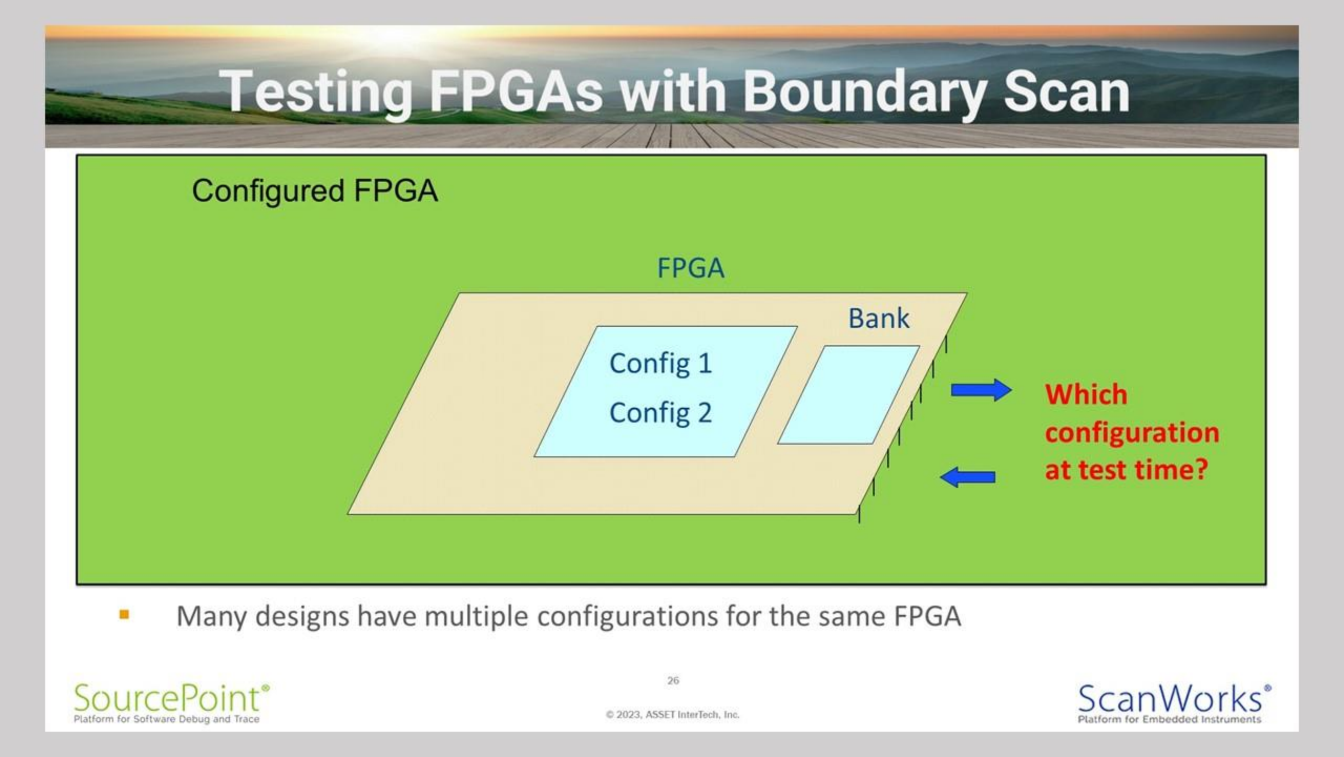
ScanWorks® Platform for Embedded Instruments

Figure 24: Testing FPGAs with Boundary Scan – Configured

Turning to the case for testing configured FPGAs. The BSDL file supplied by the device manufacturer is for the unconfigured device. Once the device is configured the BSDL needs to be modified.

The design team should provide you with the configuration file. The FPGA vendor should have tools to generate a post-configuration BSDL file leveraging the configuration file. So, the first disadvantage of testing configured FPGAs is that a custom BSDL must be created.

Testing FPGAs with Boundary Scan – Multiple Configurations


Testing FPGAs with Boundary Scan
Configured FPGA
FPGA
Bank
Config 1
Config 2
Which configuration at test time?
SourcePoint®
Platform for Software Debug and Trace
26
© 2023, ASSET InterTech, Inc.
ScanWorks®
Platform for Embedded Instruments

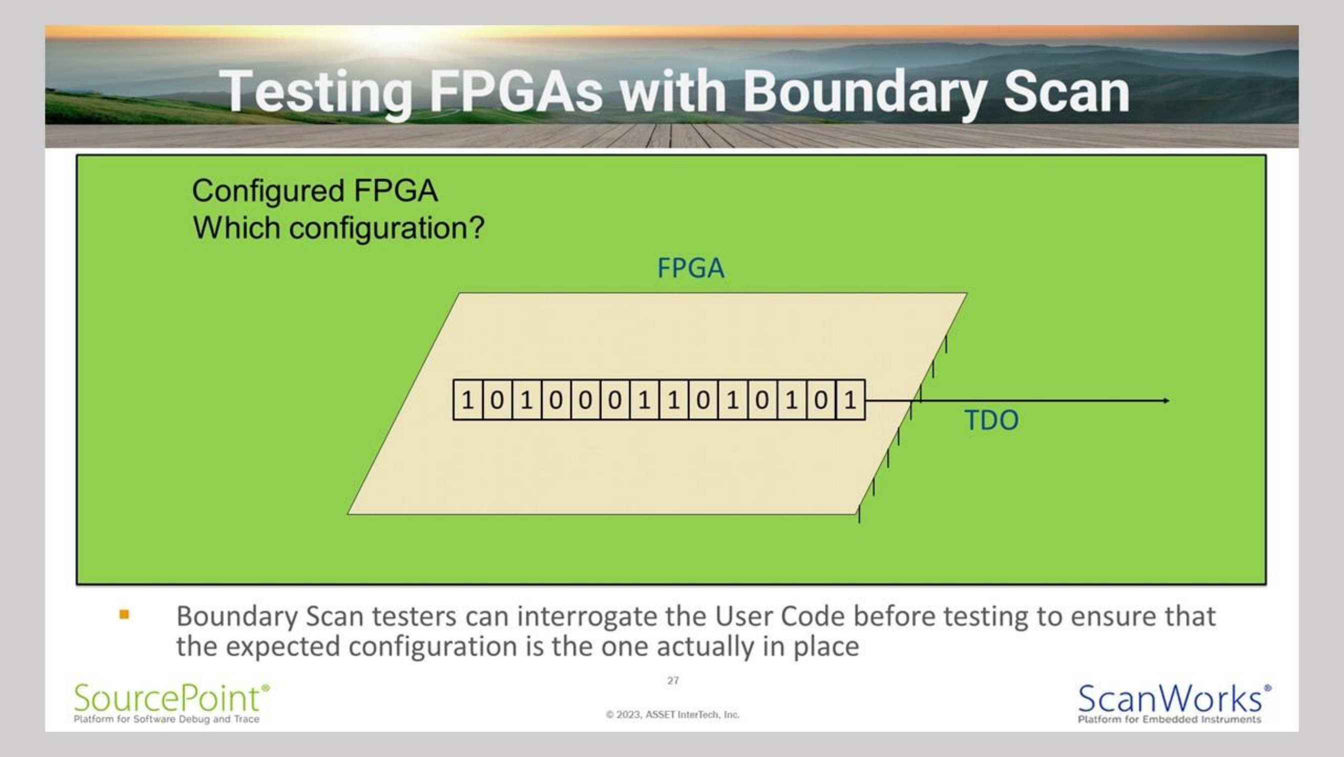
- Many designs have multiple configurations for the same FPGA

Figure 25: Testing FPGAs with Boundary Scan – Multiple Configurations

Many designs have multiple configurations for the same FPGA. Configuration details may also change from time to time (field upgrades, etc).

So, the second disadvantage of testing configured FPGAs is controlling which configuration is present in the FPGA.

Testing FPGAs with Boundary Scan – User Code



Testing FPGAs with Boundary Scan

Configured FPGA
Which configuration?

FPGA
1 0 1 0 0 0 1 1 0 1 0 1 0 1
TDO

- Boundary Scan testers can interrogate the User Code before testing to ensure that the expected configuration is the one actually in place

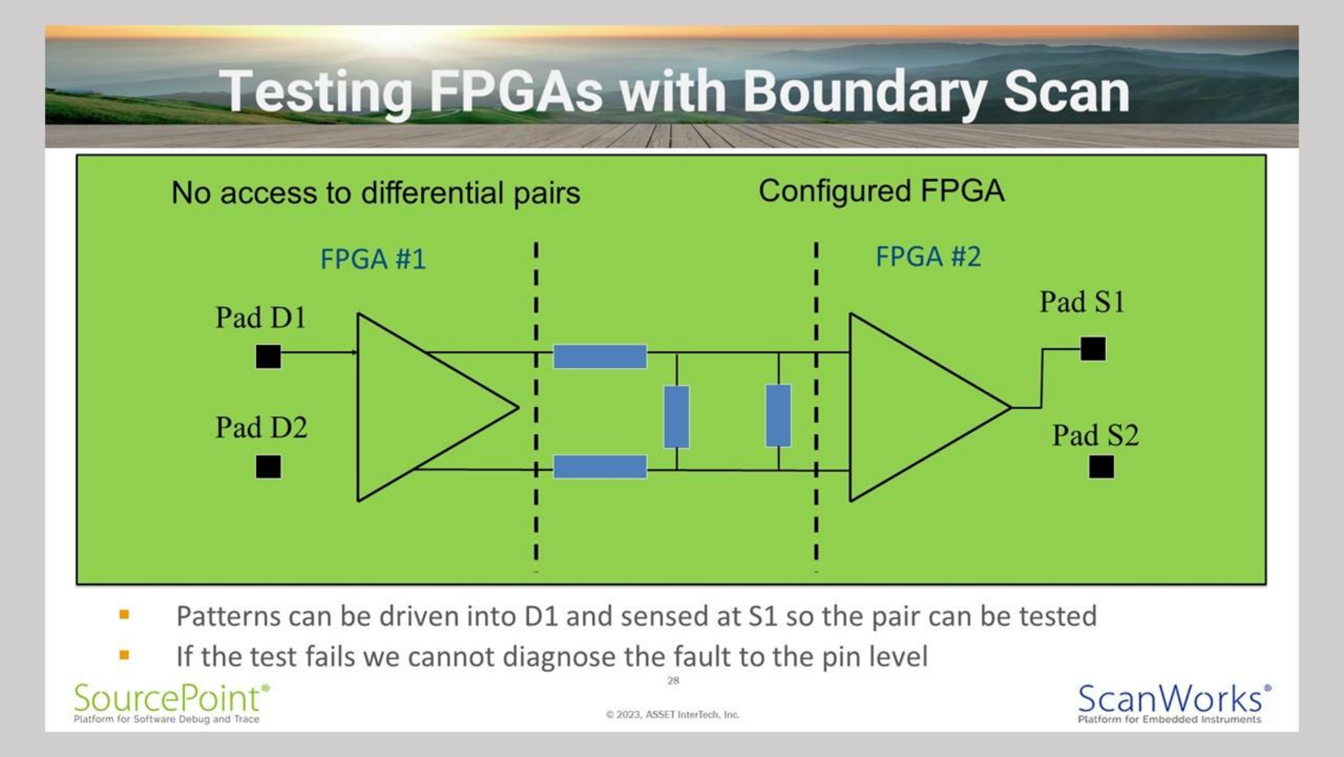
27
SourcePoint®
Platform for Software Debug and Trace
ScanWorks®
Platform for Embedded Instruments

Figure 26: Testing FPGAs with Boundary Scan – User Code

Boundary Scan testers can interrogate the User Code before testing to ensure that the expected configuration is the one actually in place. User codes are like the IDCODE, but they are under your control.

User Codes should be unique for every version of every configuration. If you currently implement User Codes, you may see testing with post-configured BSDLs as an advantage.

Testing FPGAs with Boundary Scan – Configured Driver and Receiver



Testing FPGAs with Boundary Scan

No access to differential pairs
Configured FPGA
FPGA #1
FPGA #2
Pad D1
Pad D2
Pad S1
Pad S2

- Patterns can be driven into D1 and sensed at S1 so the pair can be tested
- If the test fails we cannot diagnose the fault to the pin level

28

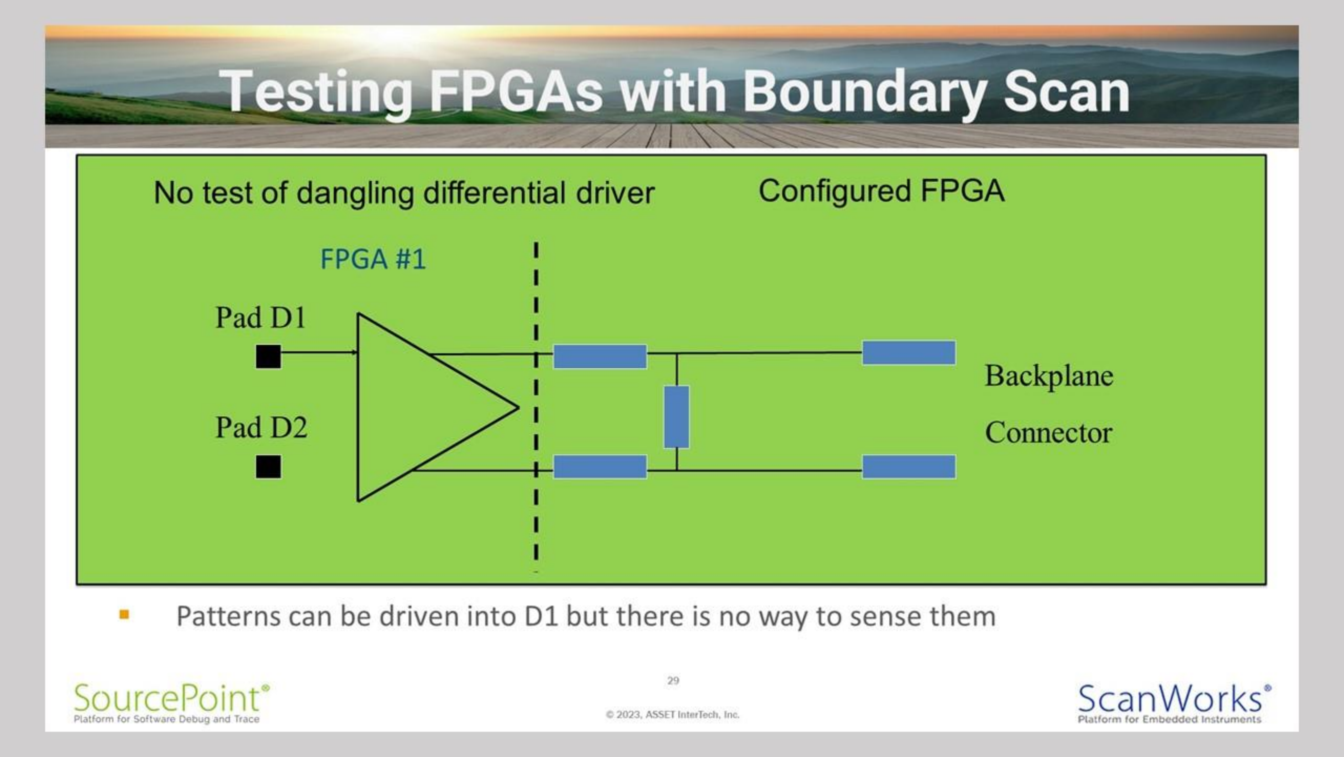
© 2023, ASSET InterTech, Inc.

Figure 27: Testing FPGAs with Boundary Scan – Configured Driver and Receiver

This is a differential pair after configuration. Notice that the drive and sense amplifiers are now in place. These sit between the pads that Boundary Scan accesses and the pins. Notice that D2 and S2 are no longer connected to anything. We can drive patterns into D1 and sense them at S1 so we can test the pair.

But, if the test fails, we cannot diagnose the fault to the pin level. We would not know whether to rework FPGA #1 or FPGA #2.

Testing FPGAs with Boundary Scan – Dangling Differential Driver



Testing FPGAs with Boundary Scan

No test of dangling differential driver

Configured FPGA

- Patterns can be driven into D1 but there is no way to sense them

SourcePoint® Platform for Software Debug and Trace

29

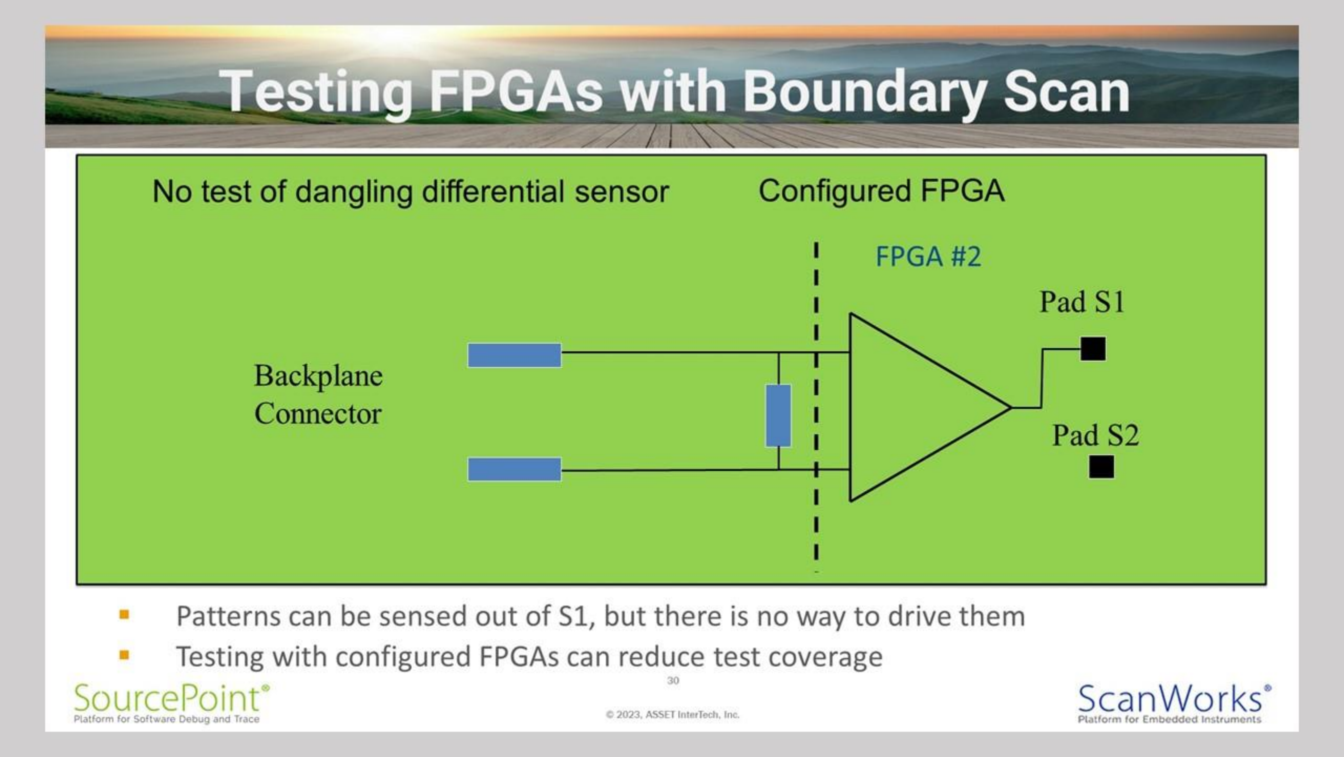
© 2023, ASSET InterTech, Inc.

ScanWorks® Platform for Embedded Instruments

Figure 28: Testing FPGAs with Boundary Scan – Dangling Differential Driver

If we only have access to one end of the pair, we cannot test it at all. Patterns can be driven into D1, but we have no way to sense them.

Testing FPGAs with Boundary Scan – Dangling Differential Receiver



Testing FPGAs with Boundary Scan

No test of dangling differential sensor

Configured FPGA

FPGA #2
Pad S1
Backplane Connector
Pad S2

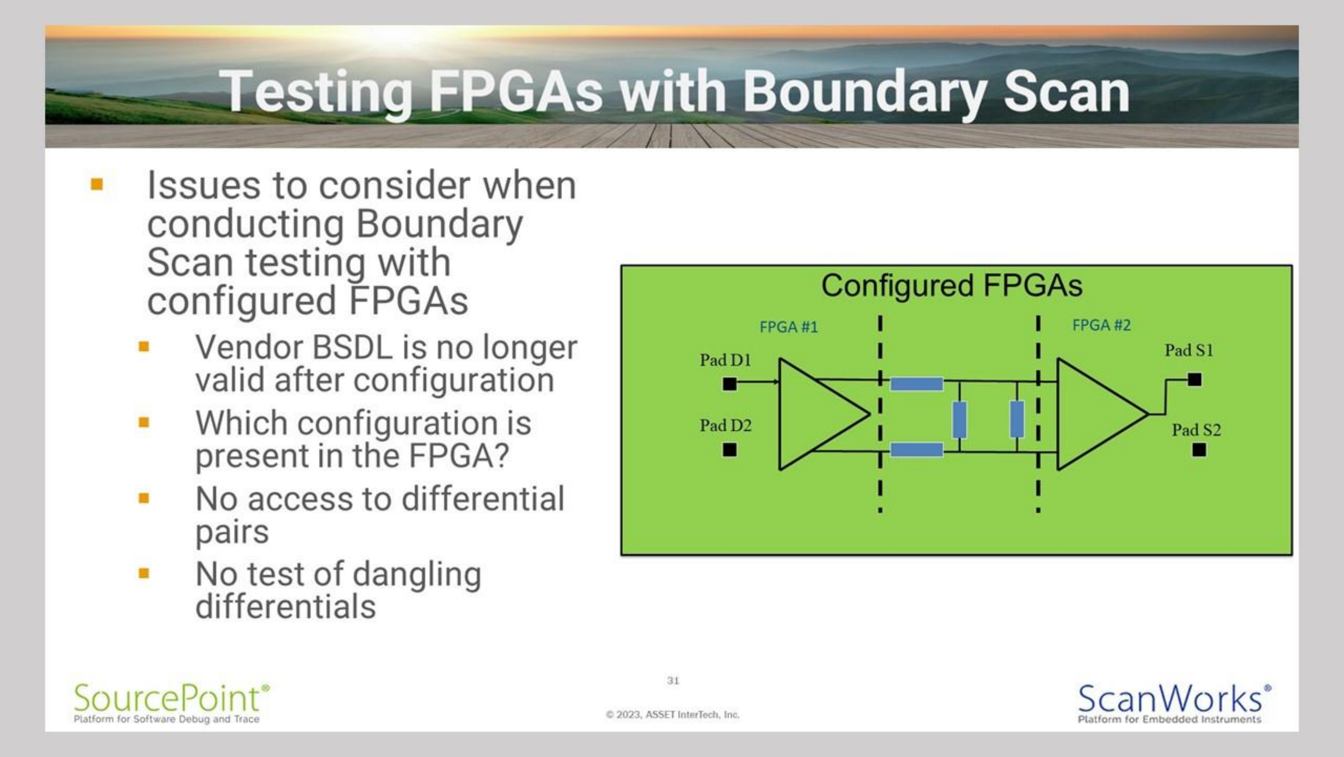
- Patterns can be sensed out of S1, but there is no way to drive them
- Testing with configured FPGAs can reduce test coverage

SourcePoint®
Platform for Software Debug and Trace
30
ScanWorks®
Platform for Embedded Instruments

Figure 29: Testing FPGAs with Boundary Scan – Dangling Differential Receiver

The same is true at the sense end. We could sense patterns out of S1, but we have no way to drive them. Testing with configured FPGAs can reduce test coverage.

Testing FPGAs with Boundary Scan – Configured Considerations



Testing FPGAs with Boundary Scan

- Issues to consider when conducting Boundary Scan testing with configured FPGAs
  - Vendor BSDL is no longer valid after configuration
  - Which configuration is present in the FPGA?
  - No access to differential pairs
  - No test of dangling differentials

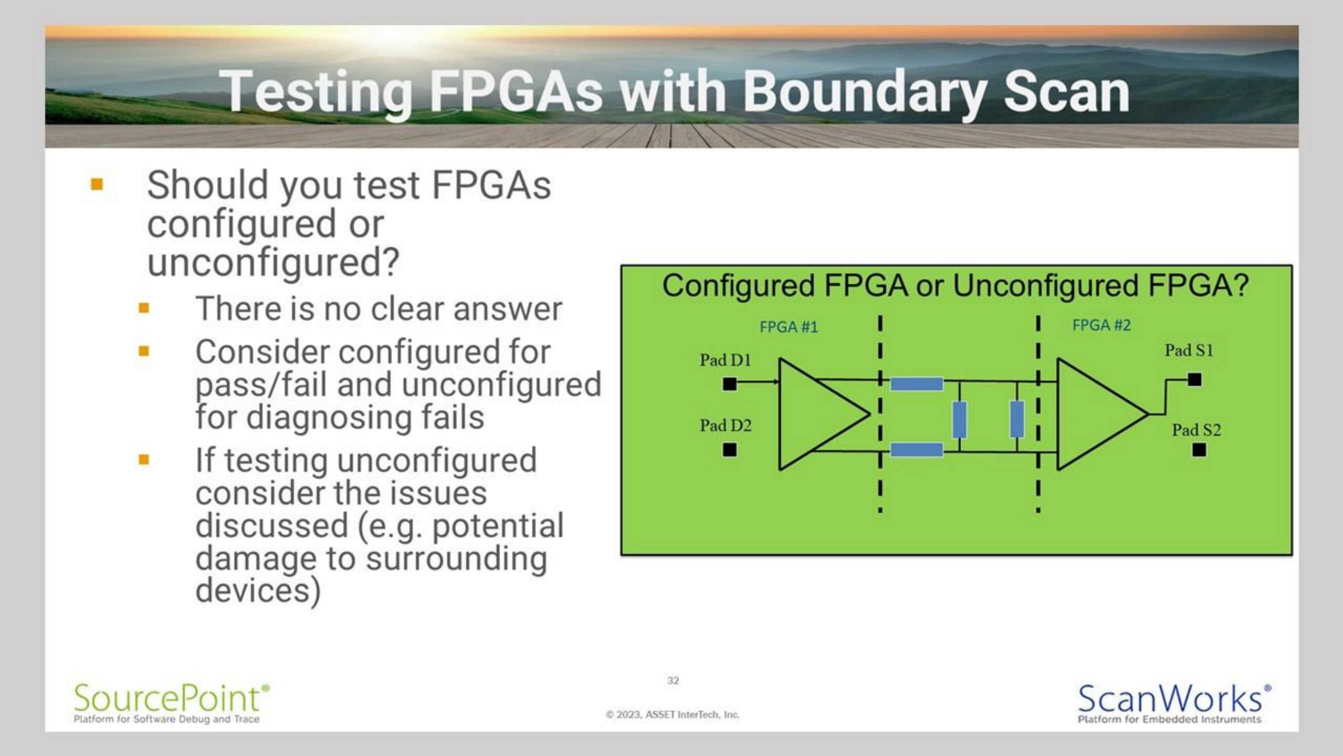
31

© 2023, ASSET InterTech, Inc.

Figure 30: Testing FPGAs with Boundary Scan – Configured Considerations

To summarize these are the issues to consider when testing with configured FPGAs.

Testing FPGAs with Boundary Scan – Configured vs Unconfigured Summary



Testing FPGAs with Boundary Scan

- Should you test FPGAs configured or unconfigured?
  - There is no clear answer
  - Consider configured for pass/fail and unconfigured for diagnosing fails
  - If testing unconfigured consider the issues discussed (e.g. potential damage to surrounding devices)

Configured FPGA or Unconfigured FPGA?

SourcePoint® Platform for Software Debug and Trace

32

© 2023, ASSET InterTech, Inc.

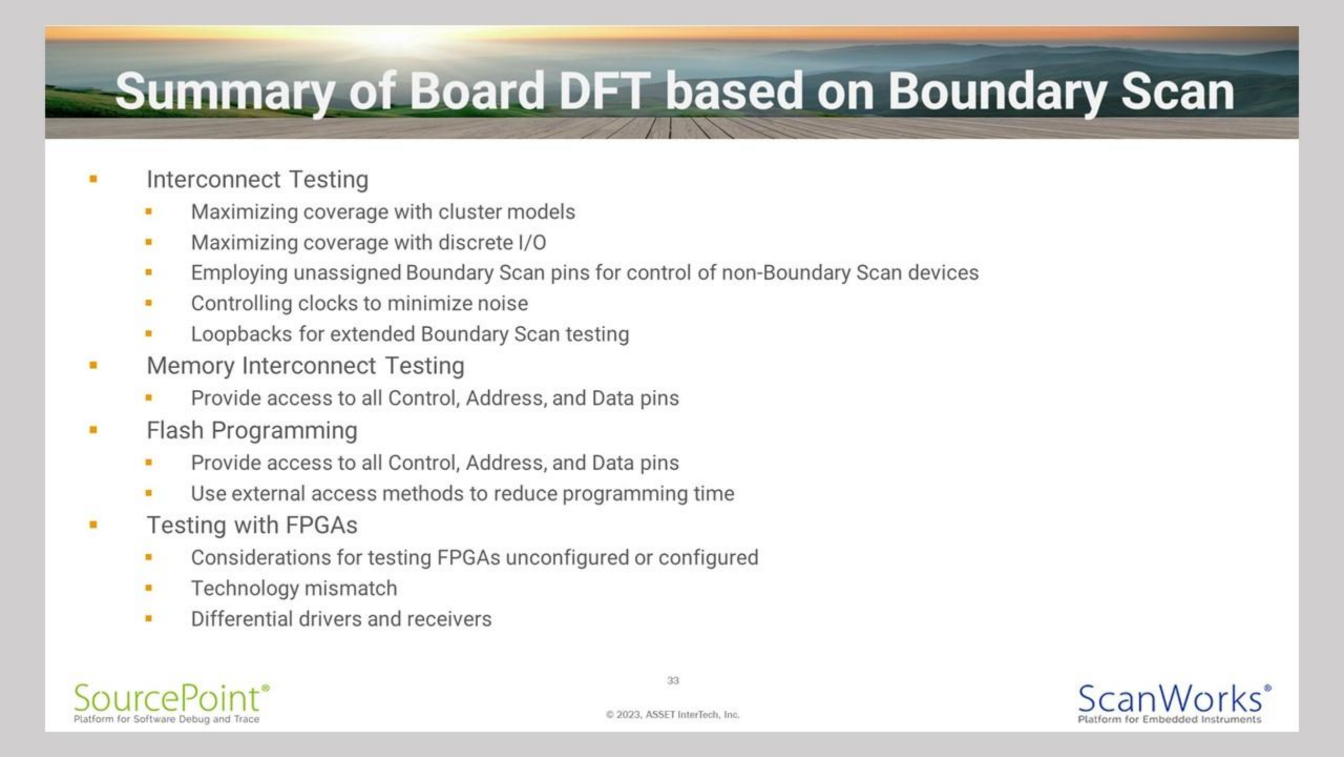
ScanWorks® Platform for Embedded Instruments

Figure 31: Testing FPGAs with Boundary Scan – Configured or Unconfigured Summary

So, should you test FPGAs configured or unconfigured? There is no clear answer, it depends on your circumstances. Testing configured is probably the safest strategy to reduce possible damage to surrounding devices. Testing configured might also be necessary to validate interconnects for low-voltage memories.

Testing unconfigured is probably best if you use differential circuits or if configuration versions are difficult to predict (e.g., field test/returns).

Summary of Board DFT based on Boundary Scan – Volume 2



Summary of Board DFT based on Boundary Scan

- Interconnect Testing
  - Maximizing coverage with cluster models
  - Maximizing coverage with discrete I/O
  - Employing unassigned Boundary Scan pins for control of non-Boundary Scan devices
  - Controlling clocks to minimize noise
  - Loopbacks for extended Boundary Scan testing
- Memory Interconnect Testing
  - Provide access to all Control, Address, and Data pins
- Flash Programming
  - Provide access to all Control, Address, and Data pins
  - Use external access methods to reduce programming time
- Testing with FPGAs
  - Considerations for testing FPGAs unconfigured or configured
  - Technology mismatch
  - Differential drivers and receivers

SourcePoint® Platform for Software Debug and Trace

33

© 2023, ASSET InterTech, Inc.

ScanWorks® Platform for Embedded Instruments

Figure 32: Summary of Board DFT based on Boundary Scan – Volume 2

So, in summary. These are the DFT guidelines we've covered today.

Conclusion

In this eBook, we examined Design for Test (DFT) guidelines specific to the design of boards to be tested through the Boundary Scan registers of IEEE 1149.1-compliant devices. Boards designed implementing Boundary Scan DFT guidelines are capable of being tested during the manufacturing process with a Boundary Scan test tool. The most important guideline (presented in volume 1 of this eBook) is that where possible, use 1149.1-compliant devices on the board in lieu of non-Boundary Scan devices. Without Boundary Scan devices on the board, no degree of Boundary Scan coverage exists. Next, to ensure testability through the Boundary Scan registers, the 1149.1-compliant devices must be connected TDI-to-TDO. The other device TAP signals, TMS, TCK, and TRST* (if applicable) are also connected in a distributed fashion. This configuration of the 1149.1-compliant devices is known as a scan chain. Boundary Scan DFT guidelines state that access to the TAP signals be provided to the Boundary Scan test tool. The more Boundary Scan registers that are accessible on the board, the higher the Boundary Scan test coverage for detection of open and/or short structural faults.

As Boundary Scan technology matured, it is now used for testing memory devices such as SRAM and DDR, programming flash devices, and configuration of programmable devices such as FPGAs. There are also DFT guidelines that decrease the programming time of flash devices via Boundary Scan by using external hardware to access the WE RDY/BSY signals. With proper DFT access to these signals, programming time can be decreased by upwards of 50%.

When testing with FPGAs, the Test Engineer must consider if testing with the FPGA configured or unconfigured with satisfy all test requirements. Testing the FPGA unconfigured allows more test coverage because Boundary Scan has access to all IO. However, testing my damage the surrounding devices. If the FPGA is tested configured, the surrounding devices are safer, because the voltages match, however there is some loss of test coverage.

Maximizing board test coverage is imperative to improving manufacturing yields, increasing product quality, and reducing product returns. The prime reasons for developing a board test strategy are to find defects, diagnose the cause of the faults, and repair them quickly. The obtained fault data can be used to improve the processes that produced the board and alleviate conditions which create board faults.