

EMBEDDED JTAG

FOR

BOUNDARY-SCAN TEST

TECHNICAL OVERVIEW

BY ALAN SGUIGNA





Alan Sguigna – Vice President of Sales & Customer Service

Alan has more than 20 years of experience in senior-level general management, marketing, engineering, sales, manufacturing, finance and customer service positions. Before joining ASSET, he worked in the telecom industry. He has had profit and loss responsibility for a \$150 million division of Spirent Communications, a supplier of test products and services. Prior to his tenure with Spirent, Mr. Sguigna also served in business development positions with Nortel Networks, overseeing the growth of its voice over Internet protocol (VoIP) products.

Table of Contents

Executive Summary	4
Test Coverage and Fault Detection/Diagnosis.....	5
Boundary-Scan Test.....	5
Design for Test Requirements	5
Technical Description	5
Action Types:.....	6
Action Players:.....	6
Resource Requirements	7
Hardware Signal Interface:	7
FPGA Resources:.....	8
Conclusion	9
Learn More.....	9

Table of Figures

Figure 1: IEEE 1149.1 Tap Controller.....	4
---	---

© 2017 ASSET InterTech, Inc.

ASSET, ScanWorks and SourcePoint are registered trademarks, and the ScanWorks logo are trademarks of ASSET InterTech, Inc. All other trade and service marks are the properties of their respective owners.

Executive Summary

ASSET InterTech develops and delivers JTAG/boundary-scan based test tools for a wide variety of board designs of various architectures where key silicon components implement IEEE Std 1149.1 and related embedded instrumentation accessible via the standard test access port (TAP) (Figure 1). Based on its long experience with JTAG, ASSET provides for the embedding of boundary scan test utilities resident within OEMs' printed circuit boards. These test utilities (ASSET's ScanWorks Embedded Diagnostics for Boundary-Scan Test, or SED for BST) are used by test and debug engineers during the lifecycle of the product, unencumbered by external hardware controllers, cables and fixturing. This provides substantial productivity gains and cost savings to OEMs in the forms of improved yields, test capital cost reduction, test time compression, and enhanced diagnosability of repair depot and field failures.

This document is a brief description of the technology and implementation for SED for BST.

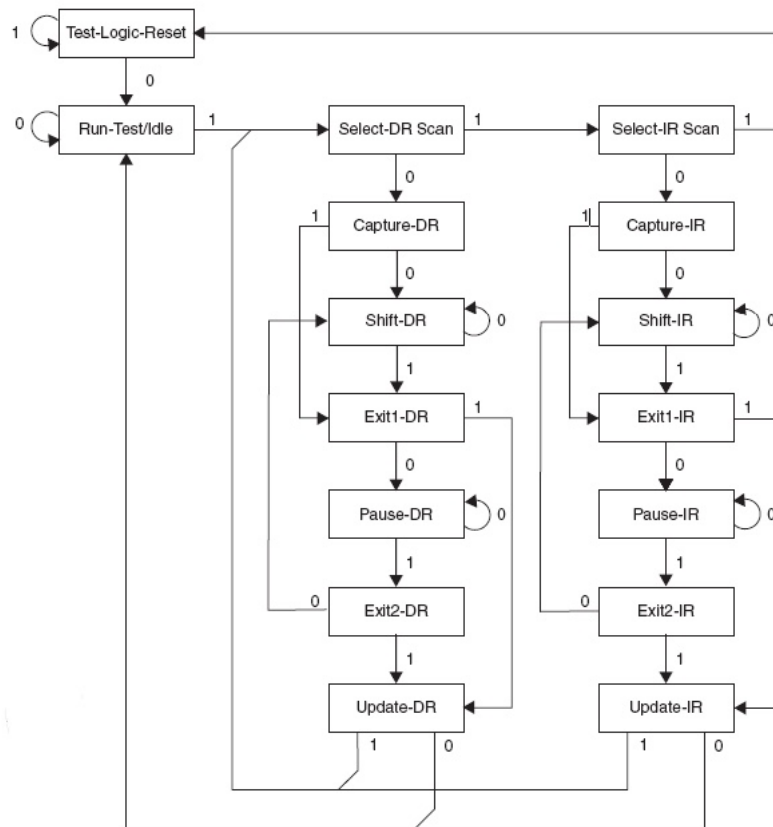


Figure 1: IEEE 1149.1 Tap Controller

Test Coverage and Fault Detection/Diagnosis

Boundary-Scan Test

Boundary-scan test (BST) is a static vector-based test technology that uses the on-chip JTAG (aka IEEE 1149.x) embedded instrument within commercial silicon to perform structural testing. As such, it performs shorts/opens/stuck-at fault testing and detection very quickly. Diagnostics are to the device and net/pin level.

BST provides a very effective means of detecting and diagnosing in-situ assembly defects. The boundary-scan application, being software-based, is usable throughout the lifecycle of the product. By embedding BST, thereby removing any external hardware (controller/pod, cabling, fixturing) dependencies, it can be applied at a system-level and in a parallel fashion in such environments as manufacturing production lines, ESS, repair/return depots, or even in the field.

Design for Test Requirements

Total boundary-scan test coverage is related to the boundary scan access that is available on the board. A key criterion for consideration is synchronization of test vectors to encompass as many JTAG-compliant devices on the board as possible. This can be achieved by having all 1149.1/1149.6 capable devices in a monolithic board chain that can be accessed from a single TAP from the embedded JTAG controller (such as from and/or within a service processor, baseboard management controller (BMC), with or without an FPGA implemented for this purpose). Concatenation, level shifting and multiplexing to achieve this functionality are the responsibility of the OEM.

Technical Description

SED for Test enables the means to execute ScanWorks actions (tests) in an embedded environment. These actions (tests) are developed using the ScanWorks Development software on a Windows PC. The ScanWorks Development software provides a data format for each supported action that allows the action to be run by the embedded action players. The action data is then either embedded on each unit under test (UUT) by storing the associated data in on-board flash memory, or downloaded to the UUT via an off-board link, preferably Ethernet.

The customer is responsible for ensuring their embedded system has adequate processor real-time capacity and memory footprint to accommodate ASSET's IP, the requirements for which are described below.

Action Types:

The following action types are supported:

- Scan Path Verify
- IEEE 1149.1 Interconnect testing
- IEEE 1149.6 Interconnect testing
- Memory Access Verification
- CPLD/FPGA Programming/Configuration
- Flash Programming

An action consists of the necessary data and/or code specific to a UUT to perform a test.

Action Players:

The action players reside on the UUT target and provide the runtime environment. They allow for flow control (looping, branching and advanced coding) and support advanced diagnostics and flexible test capabilities. ScanWorks actions run via the embedded action player.

The action player and actions as specified herein can be non-resident. That is, they can be downloaded over Ethernet, and executed dynamically at run-time. Action players support all supported action types on all UUTs. Actions themselves are specific to the UUT and an action type.

Each action as specified above is a standalone application that can be run on the UUT via a command line in a telnet (serial port) session. Pass results are piped to the console. Fail notifications are sent to the console, and diagnostic data is uploaded via Ethernet. BST failure diagnostic data needs to be uploaded into a ScanWorks benchtop license for diagnostic processing and reporting. BST applications access the embedded SED library and execute IRScan and DRScan functions.

The OEM is responsible for downloading the action player and appropriate actions to the applicable UUTs, running/stopping tests, uploading pass/fail status and diagnostic results, parsing results data, cleaning up the file system when complete, and other management functions.

Resource Requirements

The action players occupy ≤ 2 MB of flash footprint. The action data size is dependent upon the application. BST SPV and BST MAV action data are negligibly small. BST interconnect actions can be $> \sim 1$ MB for large designs, with a typical range of ~ 200 kB - 300 kB. The runtime memory needed to run STAPL files is entirely dependent upon the size of the file.

The IP that implements the IEEE 1149.1 (JTAG) TAP Controller signals is a proven high-performance design that is utilized in several of ASSET's off-the shelf-hardware offerings. For hardware board designs supporting an FPGA that has real-estate to support the JTAG mastering function, an implementation for the TAP controller written in VHDL will be provided (in obfuscated form with exception of the top level entity) and included for each customer design.

Hardware Signal Interface:

The current interface to the SED Controller is a simple VHDL port that behaves like a standard Wishbone slave interface. The SED VHDL port comprises the following (example only – may be subject to minor change and/or modified as suited for the targeted hardware design):

- rst_i: Reset input
- clk_i: Clock input (100MHz)
- addr_i[8:0]: D-word (32-bit) aligned byte address
- sel_i[3:0]: Select input array
- cyc_i: Cycle input
- stb_i: Strobe input
- lock_i: Lock input
- we_i: Write enable input
- dat_i[31:0]: Data input array
- ack_o: Acknowledge output

dat_o[31:0]: Data output array

jtag_clk: JTAG source clock input (see Note)

TRST: Test Logic Reset output from embedded TAP Controller

TCK: Test Clock output from embedded TAP Controller

TMS: Test Mode Select output from embedded TAP Controller

TDI: Test Data In (UUT referenced) output from embedded TAP Controller

TDO: Test Data Out (UUT referenced) input to embedded TAP Controller

oe_n: Output enable for all outbound SED signals

Note: The TAP controller IP is designed to operate with jtag_clk up to 50MHz and has no minimum frequency requirement, although a jtag_clk must be present for the TAP controller IP to operate.

The use of a system clock as well as a jtag_clk allows reading and writing internal FIFOs and registers at much faster rates than the TCK is operating. This improves performance and frees the controlling processor to perform other tasks.

FPGA Resources:

The resources required by the TAP Controller IP are small when compared to today's FPGAs. The IP will reside in fewer than 4,000 Lookup Tables (LUTs) (or equivalent) and require only 74k-bits of memory. These memory requirements are based on send and receive FIFOs sized at 2k by 18-bits. The following is a small excerpt from a place and route report of a similar implementation in a Xilinx Virtex-5 xc5v1x50t:

Number of Slice LUTS	3,682 out of 28,800	12%
----------------------	---------------------	-----

Number of Slice LUT Flip-Flop pairs	5,177 out of 28,800	17%
-------------------------------------	---------------------	-----

The address space required by the SED Controller IP is 256 32-bit locations (8 address bits) of contiguous addresses within the FPGA.

Two global clock circuits will be required for routing the system and JTAG clock sources within the FPGA.

Conclusion

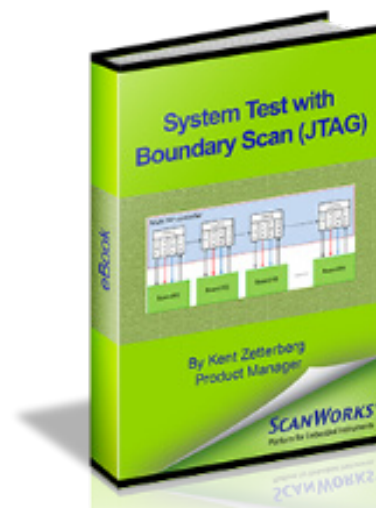
Mission-critical systems are expected to operate 24 X 7 with close to zero outage downtime. To achieve this level of performance, such systems must combine high reliability with rapid field repair and replacement.

ASSET provides embedded firmware solutions for structural/functional testing and CPU run-control for many industries, including military/aerospace, servers, storage, telecom, and others. This third-party Intellectual Property (IP) reduces development costs and accelerates time-to-market.

With embedded boundary-scan test, OEMs and their customers can debug intermittent, irreproducible hardware defects in the lab and in the field, that are often the source of costly NTFs and customer dissatisfaction. The focus can then shift to product functionality, product differentiation, and high levels of customer service and satisfaction.

Learn More

*You might be interested in
learning more about our
system-level test using JTAG.*



Register Today!