An Embedded Technique For At-Speed Interconnect Testing

Benoit Nadeau-Dostie, Jean-Francois Cote, Harry Hulvershorn and Stephen Pateras

LogicVision Canada, Inc.
Ottawa, Ontario

LogicVision, Inc.
San Jose, CA

Abstract

A new embedded test technique which provides full at-speed testing of board level interconnect is described. The proposed technique is fully compatible with the IEEE 1149.1 boundary scan standard. The technique extends the standard's architecture to provide for synchronized at-speed timing control of the boundary scan cells so that test data can be applied and captured across the interconnect at system speeds.

1. Introduction

Rapid advances in silicon, packaging and board technologies have created new opportunities and challenges for equipment manufacturers. Indeed, the complexity and speed of today's boards are creating severe manufacturing test challenges. This is particularly true when it comes to testing for delay related defects in the board interconnect.

Many companies today perform interconnect testing using in-circuit test (ICT) techniques [10]. Testers that provide this capability typically contain hundreds to thousands of physical probes (often called a "bed-of-nails") that are used to contact chip pins and board test points to perform the interconnect tests. However, the continued practicality of ICT is severely challenged by advances in packaging technology. Dense, double-sided boards require thousands of probes, and modern surface-mount chip packages leave most chip pins physically inaccessible. In addition, testing for interconnect delay faults is not practical due to the inability of the tester hardware to apply at-speed signals through the probes.

ICT's physical access problem drove the creation and subsequent industry-wide acceptance of the IEEE 1149.1 Boundary Scan Standard [1-9] as a foundation for probeless interconnect testing. Unfortunately, the IEEE 1149.1 standard is not designed for applying and capturing data across interconnects at application speed. Because of the standard's serial control approach, a minimum of 2.5 test clock (TCK) cycles are required between the time test data can be launched from one chip and captured at another. So for example, with a TCK operating frequency of 25 MHz, the effective interconnect test frequency available from 1149.1 becomes 10 MHz, clearly much lower that typical board application speeds.

In light of the above limitations, the method most commonly used to achieve at-speed interconnect testing is through functional testing at the board or system level. Functional testing though has some strong limitations. The time and resource required to develop the functional tests can be very high. Perhaps even worse is the time needed to diagnose the physical location of a fault once it is detected.

In this paper, a new embedded test technique which provides full at-speed testing of board level interconnect is described. The proposed technique builds upon the standard IEEE 1149.1 test approach. The technique uses new at-speed boundary scan cells as well as embedded timing control for these cells. A key feature of the proposed technique is that it is compatible to IEEE 1149.1 based testing. A board design can incorporate a mixture of chips incorporating the technique described in this paper and chips simply compliant to the IEEE 1149.1 standard. Any existing boundary scan test hardware as well as test and diagnostic software tools can still be used without modification.

The remainder of this paper is organized as follows. Section 2 describes the high level architectures at both the chip and board levels required to implement the proposed technique. Section 3 describes the new at-speed boundary scan cells while the at-speed timing controller is described in section 4. Section 5 describes the use model of the proposed technique in the context of boundary scan based testing. Section 6 concludes the paper.

2. High Level Architectures

The diagram in Figure 1 illustrates the chip level architecture required to implement the proposed technique. Shown are the standard IEEE 1149.1 TAP and controller together with at-speed boundary scan cells and an at-speed interconnect (ASI) controller. The at-speed boundary scan cells consist of IEEE

0-7803-5753-1/99 $10.00

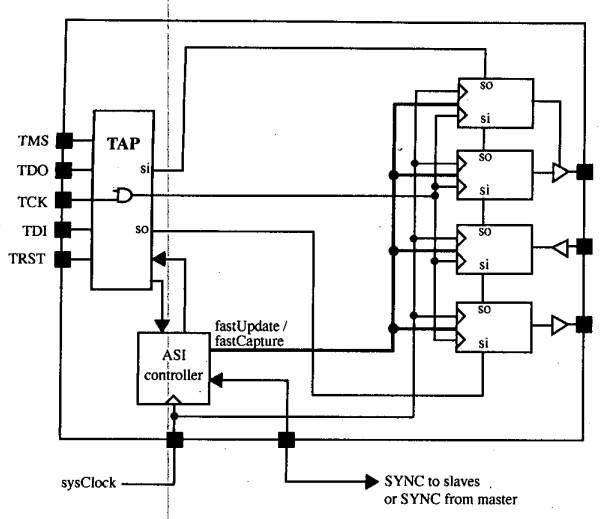

Figure 1: Chip Level Architecture for At-Speed Interconnect Testing

1149.1 compliant boundary scan cells augmented with specialized logic and system clocked flip-flops. A detailed description of these cells is provided in section 3. Although clocked by the system clock, these new boundary scan cells are still clocked by a gated version of TCK for scanning. However instead of receiving the 1149.1 standard update and capture signals (referred to by the standard as UpdateDR and ClockDR respectively) by the TAP controller, the boundary scan cells receive high speed versions of these signals, fastUpdate and fastCapture, generated by the ASI controller.

The ASI controller can also source or sink a synchronization (SYNC) signal connected to a chip bidirectional pin. This SYNC signal is used to synchronize all ASI controllers on the board involved in the at-speed interconnect test. One of the ASI controllers acts as a master and sources the SYNC signals at the appropriate time, while all other ASI controllers act as slaves and wait to receive the generated SYNC signal before proceeding with the at-speed test.

2.1 Board Level Architectures

The simple scenario shown in Figure 2 illustrates how chips implementing the proposed technique are to be assembled at the board level. Notice that in this example there are two separate clock domains, CK1 and CK2. In general, any number of system clocks, synchronous or asynchronous to each other, can be supported. The only requirement is that a chip must possess a separate ASI controller for each clock domain that possesses at-speed boundary scan cells. Each of these ASI controllers only controls those at-speed boundary scan cells which exist in the same clock domain. Similarly,

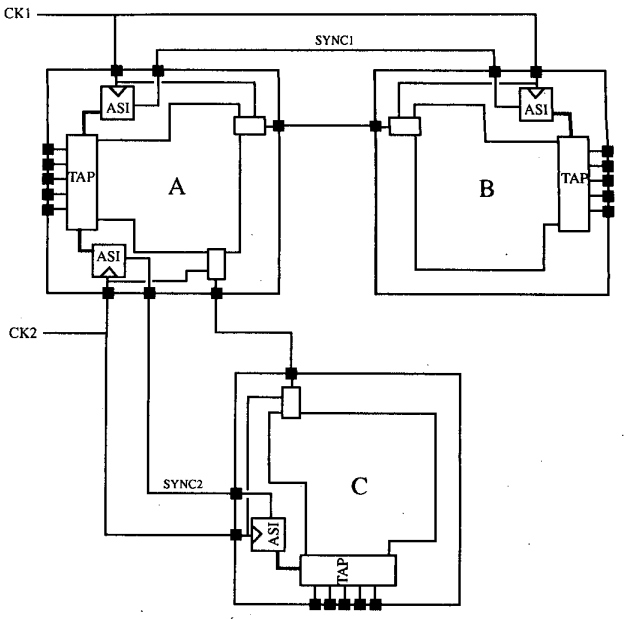

Figure 2: Sample Board Scenario

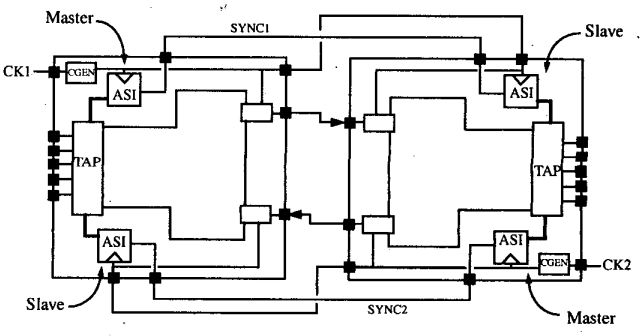

Figure 3: Board With Source Synchronous Clocks

each ASI controller on a given chip only interacts with ASI controllers on other chips which are in the same clock domain. For example, in Figure 2, chip A is fed by both clocks CK1 and CK2 and thus requires two ASI controllers. The chip's top most ASI controller connects through a common SYNC line to the ASI controller on chip B as these exist in the same clock domain (CK1). Similarly, chip A's bottom most ASI controller connects through a common SYNC line to the ASI controller on chip C as these two both exist in clock domain CK2.

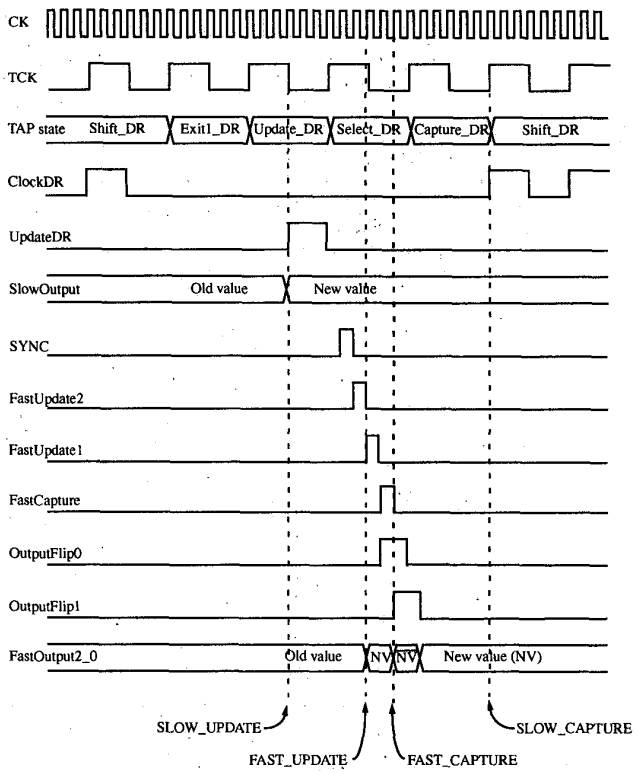

Figure 4: Timing Diagram for At-Speed Interconnect Testing

Figure 3 illustrates the board level case where the clocks are not routed synchronously to all chips, but rather clock and data are sent together from one chip to the next. This is referred to as a source synchronous interface. Handling such an interface requires that the ASI controller on the chip sourcing the clock always act as Master. Notice that a chip can both source and receive clocks as is the case in Figure 3. In this situation, the chip requires a separate ASI controller acting as a Master for each clock domain in which the clock is sourced, and a separate ASI controller acting as a Slave for each clock domain in which the clock is received.

2.2 At-Speed Interconnect Test Timing

Given the above board level architectures, an interconnect test pattern can be applied at-speed from one chip to another based on the timing diagram shown in Figure 4. The timing is designed such that chips implementing the proposed technique (*Fast* chips) and chips implementing the 1149.1 standard (*Slow* chips) can exchange test data safely. After serially loading test data in the boundary scan chain using the ClockDR signal, the UpdateDR signal is generated by the Slow chips to update their outputs (SLOW_UPDATE event in Figure 4). The Fast chips wait for the SYNC signal to be generated before they begin the process of updating their outputs and capturing their inputs. The different at-speed update signals (FastUpdate*N*) are used for different groups of outputs depending on the number of system clock cycles allowed for the propagation of those outputs to the inputs of other fast chips. The diagram in Figure 4 shows two at-speed update signals, FastUpdate1 and FastUpdate2, which will support respectively either one or two cycle interconnect paths. In general, interconnect paths of any system cycle length can be supported by having the ASI controller generate an at-speed update signal sufficiently early.

In the proposed implementation, all Fast chips capture their inputs at the same time using the *FastCapture* signal. It would be possible to provide delayed versions of the *FastCapture* signal in order to add even more flexibility in supporting multi-cycle interconnect paths.

For outputs that have the capability, the *OutputFlip* signals are used to invert the outputs around the time of the capture. This feature allows to test for hold time violations. In Figure 4, the *OutputFlip0* and *OutputFlip1* signals provide for a 0 and 1 cycle hold time test respectively. More delayed *OutputFlip* signals can also be generated if testing for longer hold times is necessary. The flipped output data is then inverted back to its original value at the next cycle so that it can propagate again to other Slow chips. Note that sequential circuits connected to outputs supporting the output flip capability might be upset by this procedure. The feature therefore needs to be

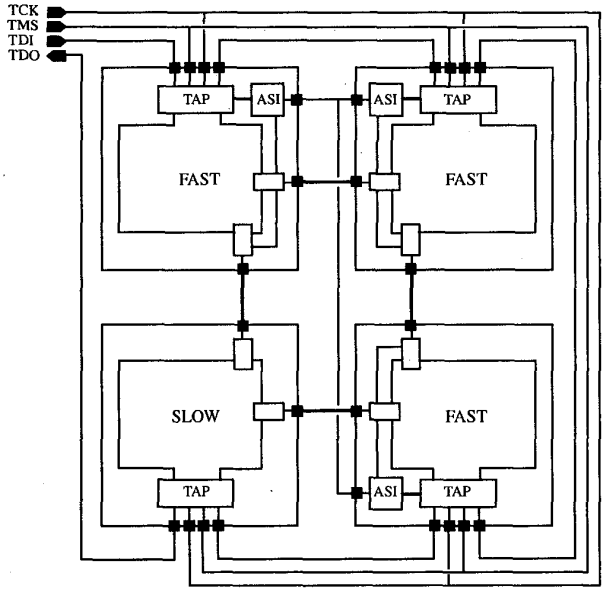

Figure 5: Fast and Slow Chips Can Be Freely Intermixed

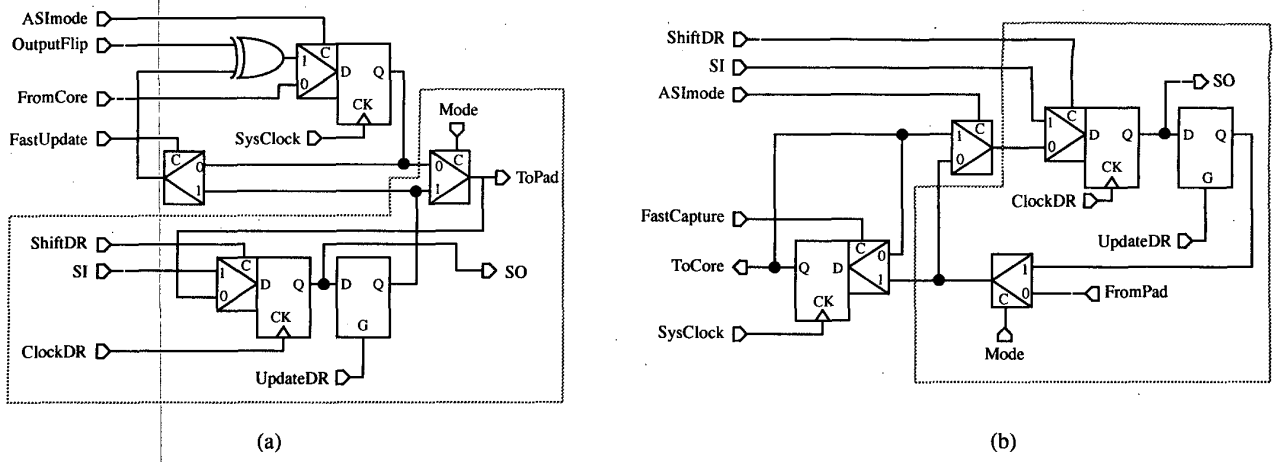

Figure 6: At-Speed Boundary Scan Cells for Registered I/O

turned off if the output response needs to be the same as if the same test pattern was applied using the standard 1149.1 sequence.

To minimize overhead, the number of system clock cycles allowed for test data to propagate from a specific output to other Fast chips is fixed. However, a refinement to the proposed technique would allow for a programmable number of cycles for data propagation. A two-bit register could be used to select which at-speed update signal should be used for a particular bus for example. A similar improvement is to provide a register to program the number of clock cycles of hold time.

The key thing to notice with this timing is that the at-speed update and capture sequence occurs in between the slow update and capture events. This makes the proposed technique fully compatible and transparent to 1149.1 operation. Indeed, a board design can incorporate a mixture of Fast and Slow chips as illustrated in Figure 5. Both chip types can be placed on the same 1149.1 test bus and receive the same test clocking and control. Any existing boundary scan test hardware as well as test generation and diagnostic tools can still be used with both chip types present without modification. Of course, in this scenario only the interconnect between Fast chips is tested at-speed, while the remaining interconnect is tested at standard TCK rates.

3. At-Speed Boundary Scan Cells

Enhancements to the standard boundary scan cells are of course needed in order to support the timing described in the previous section.

3.1 Boundary Scan Cells for Registered I/O

One possible implementation of these enhancements is shown in Figure 6. An output boundary scan cell is illustrated in Figure 6a as well as an input boundary scan cell in Figure 6b. Enable and bidirectional cells can easily be derived from the two cells shown. The cells in Figure 6 are for registered I/Os. That is, a functional flip-flop exists at the I/O boundary. In this case, the flip-flop serves as the at-speed portion of the boundary scan cell in test mode.

Refer to the output cell in Figure 6a. That portion of the cell lying within the dotted outline consists of a standard 1149.1 compliant output boundary scan cell. From this, the cell is enhanced to also include the existing functional flip-flop as well as an exor gate and two multiplexers (muxes). The mux added to the flip-flop chooses between functional and at-speed test operation. The second mux is used to control when the output test data value scanned into the boundary scan cell is to be updated into the functional flip-flop and thus sent to the output pin (for at-speed testing, the *Mode* signal is set to logical 0). This mux is therefore controlled by the at-speed *FastUpdate* signal. Finally, the exor gate, together with the feedback loop created by the *FastUpdate* signal being inactive (logical 0), is used to invert the value stored in the functional flip-flop whenever the *OutputFlip* signal is forced active (logical 1).

The input boundary scan cell in Figure 6b contains enhancements similar to those for the output cell. Once again, the portion of the cell within the dotted outline consists of a standard 1149.1 cell. In addition to the existing functional flip-flop, two additional muxes are needed to provide the at-speed capabilities. The top most mux chooses between standard 1149.1 and at-speed test operation. The mux added to the func-

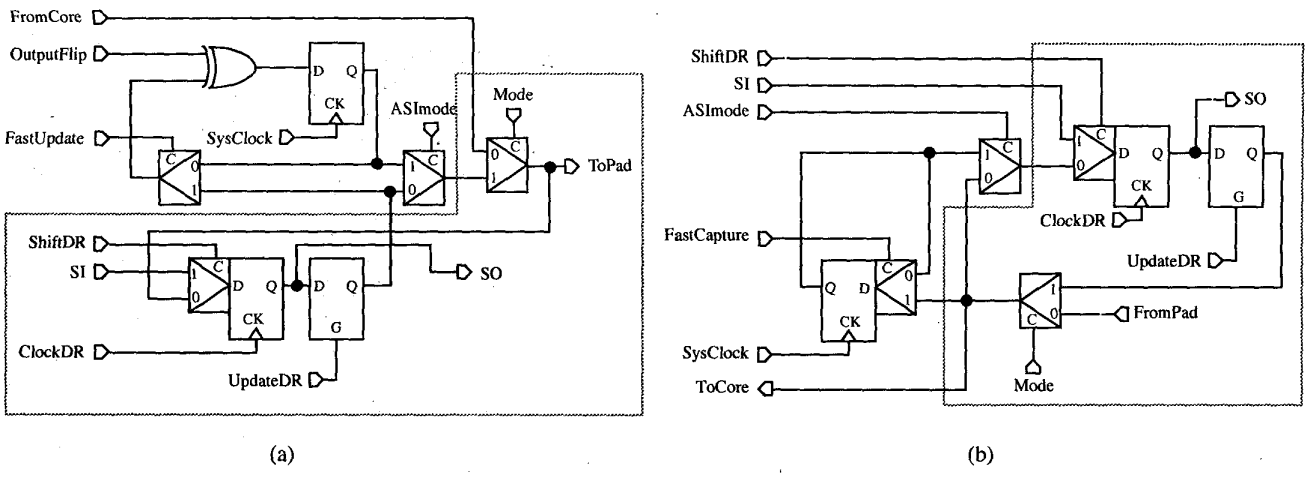

Figure 7: At-Speed Boundary Scan Cells for Non-Registered I/O

tional flip-flop is controlled by the at-speed *FastCapture* signal and therefore performs the at-speed capture of the test data received from the input pin. Note that the *FastCapture* signal is always kept active (logical 1) during normal functional operation.

3.2 Boundary Scan Cells for Non-Registered I/O

Not all designs have registered I/Os and therefore at-speed boundary scan cells which support non-registered I/Os are also needed. An implementation of these cells is provided in Figure 7. An output cell is shown in Figure 7a and an input cell in Figure 7b. As can be seen, these cells are very similar to those in Figure 6. The key difference is that for the cells in Figure 7, the flip-flop driven by the system clock does not serve as a functional flip-flop. In the case of the output cell in Figure 7a, this is apparent by the fact that the *FromCore* signal no longer feeds the flip-flop but rather directly feeds the output mux instead. Similarly, for the input cell in Figure 7b, the *ToCore* signal is no longer fed by the flip-flop, but is driven directly from the input mux.

There are a couple of drawbacks when forced to use the cells in Figure 7. The most obvious is that the flip-flop driven by the system clock is no longer used functionally and therefore becomes additional overhead. The second drawback is that the interconnect path being tested at-speed is not a true functional path. Because the I/Os are not registered, there is some functional logic in between the I/Os and the functional flip-flops driving or driven by the I/Os. This implies that the interconnect path between these I/Os is only a portion of the overall system path. Unfortunately a full system clock cycle is used to test this interconnect path, creating some slack in the timing. Nevertheless, this slack will typically be small, and will still allow for the detection of a large range of delay defects.

An important thing to notice regarding all of the boundary scan cells in both Figures 6 and 7 is that they can be easily created by using the standard boundary scan cells as a basis. The only modification to the standard cells is that an internal net needs to be cut in order to insert the mux controlled by the *ASImode* signal.

Finally, a standard boundary scan description language (BSDL) [2,9] file can still be used to describe a boundary scan chain composed of the enhanced boundary scan cells. BSDL is typically required as input to standard 1149.1 based test systems.

4. At-Speed Interconnect Test Controller

As described in section 2, each chip taking part in at-speed interconnect testing requires one ASI controller per clock domain. The function of each ASI controller is to generate the at-speed update and capture signals for the boundary scan cells within the same clock domain. In addition, when set-up to be a master, an ASI controller will generate the SYNC signal to be sent to all other ASI controllers on the board residing in the same clock domain. A block diagram of the ASI controller is provided in Figure 8.

User control of the ASI controller is provided through a scannable configuration register. This register contains a minimum of two bits. One is an Enable bit and must be set in order for the ASI controller to operate. If the ASI controller is disabled, the boundary scan cells will behave as standard 1149.1 cells. The other required bit selects between Master and Slave mode. Recall that within a given clock domain on the board,

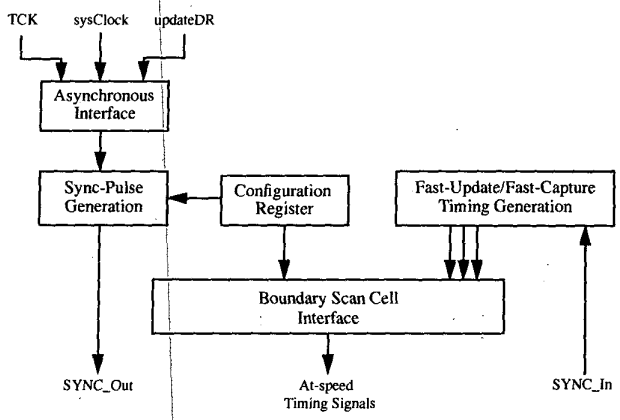

Figure 8: Block Diagram of the ASI Controller

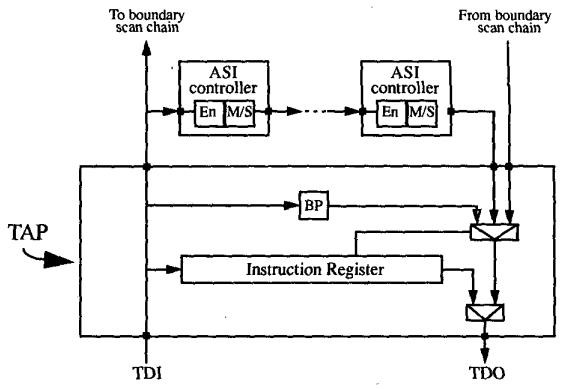

Figure 9: TAP Access of the ASI Controllers

there must be exactly one ASI controller operating as a Master, and all others must operate as Slaves. The configuration register could contain other bits as well. One example is a bit which disables the output flip capability.

When configured as a Master, the ASI controller must generate the SYNC signal as described in section 2. To generate this signal, an asynchronous interface is used to monitor the updateDR signal from the TAP. The rising edge of TCK following this event triggers the generation of the SYNC signal which is fed to a bidirectional registered I/O pin. This pin must be bidirectional as an ASI controller can be configured at run time to be either a Master or a Slave. In either case, the SYNC signal received from the registered I/O is fed to the Timing Generation block. This block generates the sequence of FastUpdate and FastCapture signals described in Figure 4. If configured to do so, it will also generate the OutputFlip signals at the appropriate times.

The Boundary Scan Interface block then drives the timing signals to the boundary scan cells across the chip. In high speed designs, it may be necessary to pipeline these signals in order to meet cycle times and control skew. If this is the case, the Timing Generation block will have to generate these signals a sufficient number of clock cycles early to counteract the pipeline delay.

5. Use Model

Figure 9 illustrates how one or more ASI controllers are hooked up to a standard 1149.1 TAP. Recall that more than one ASI controller are required on a chip if the chip's at-speed boundary scan cells reside in more than one clock domain.

As shown, the configuration registers of all ASI controllers are placed together in a scan chain. This scan chain is accessed as a separate Test Data Register from the TAP. A specific TAP instruction is used to select this register and hence provide access to the Master/Slave and Enable bits of each of the ASI controllers.

Given this architecture, running an at-speed interconnect test is straightforward. The necessary steps are:

1. For all fast chips, load the appropriate instruction into the TAP's instruction register to select the internal ASI controller chain. Load the Bypass instruction into the Slow chips.
2. Scan in the proper values for the Master/Slave and Enable bits for each of the ASI controllers within each of the Fast

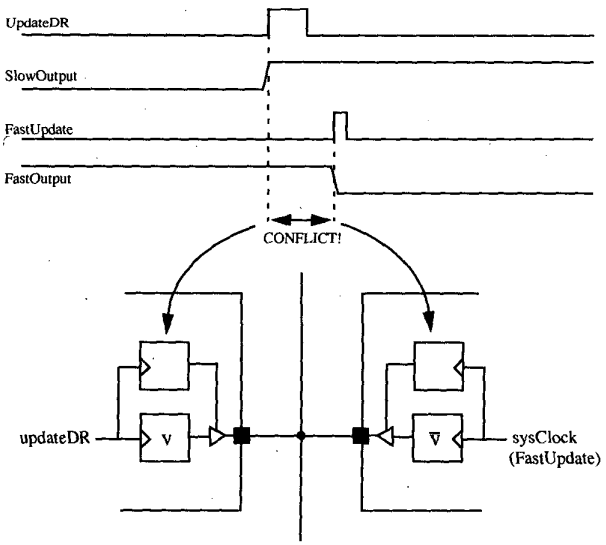

Figure 10: Potential Conflict Between 1149.1 and At-Speed Drivers

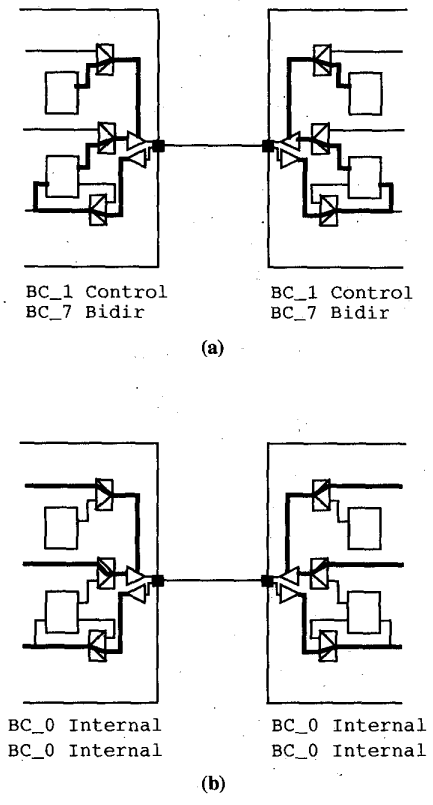

Figure 11: BSDL Requirements for (a) 1149.1 Testing and (b) At-Speed Testing

chips. Recall that within each clock domain on the board, one of the ASI controllers must act as a Master and all others must act as Slaves.

3. Proceed with standard 1149.1 interconnect test steps.

The implication of the above use model is that hardware and software tools used in standard boundary scan testing can still be used in conjunction with the proposed technique. There are nevertheless some issues to be aware of.

5.1 Fast and Slow Chip Driver Interaction

The first of these issues is depicted in Figure 10. Shown is the case where tri-state outputs from both a Fast and a Slow chip are driving the same net. A driver conflict can occur when a given test pattern enables the output on the Slow chip and disables the output on the Fast chip. This is due to the fact that the FastUpdate signal to the enable boundary scan cell on the Fast chip is generated some time after the UpdateDR signal to the enable boundary scan cell on the Slow chip, resulting in both drivers being on during that period of time. A conflict will exist if the outputs are driving different values. The time span of the conflict state is typically very small and may not be a concern. If it is however, a simple solution to this problem is to always provide an additional pattern which disables both outputs before turning one of them on.

Note that this problem can also exist with standard 1149.1 boundary scan cells in that TCK skew between chips will result in the same contention problem.

5.2 Testing Clocks and SYNC Signals

Another issue has to do with system clocks and SYNC signals used to apply the at-speed tests. Since these signals are not explicitly tested during at-speed testing, it is useful to at least test them using an 1149.1 test. This requires that boundary scan cells be associated with these signals during 1149.1 testing and not during at-speed testing. This duality is illustrated in Figure 11 for a SYNC signal. In Figure 11a, boundary scan cells are used for the bidirectional I/O on both ends of the signal. However, as shown in Figure 11b, these cells are bypassed during at-speed testing. The fact that the cells are not used must be reflected in the BSDL. This can be accomplished by defining the cells as INTERNAL and specifying that their capture value is X (accomplished using the BC_0 BSDL cell type). This of course results in the need for two BSDL files, one for 1149.1 testing and one for at-speed testing.

There are, additionally, some diagnostic benefits to applying both 1149.1 and at-speed tests. If only at-speed tests are performed, then it becomes impossible to differentiate between stuck-at and delay faults. The defect mechanisms associated with these two fault types are quite different and it is therefore useful to be able to tell them apart. A fault which is detected by an at-speed test but not by an 1149.1 test can be diagnosed as being a delay fault.

6. Conclusion

In this paper, a technique has been described that provides full at-speed testing of board level interconnect. The technique is fully compatible with the IEEE 1149.1 boundary scan standard. Standard boundary scan cells are enhanced to make use of the system clock so that test data can be applied and captured across the interconnect at system speeds. Small embedded controllers are used to provide the at-speed timing signals required by the enhanced cells.

Boundary scan tester hardware and associated test generation and diagnostic software tools used for standard 1149.1 testing can still be used with the proposed at-speed technique. This significantly reduces the effort required to migrate to the proposed technique from the standard 1149.1 approach.

References

[1] IEEE Standard 1149.1a-1993, Standard Test Access Port and Boundary-Scan Architecture, IEEE Standards Board, New York, October 1993.

[2] Supplement to IEEE Std 1149.1-1990, IEEE Standard Test Access Port and Boundary-Scan Architecture, IEEE Standards Board, New York, March 1995.

[3] D. Bhavsar, "An Architecture for Extending the IEEE Standard 1149.1 Test Access Port to System Backplanes", Proc. International Test Conference, 1991, pp. 768-776.

[4] L. Whetsel, "A Proposed Method of Accessing 1149.1 in a Backplane Environment", Proc. International Test Conference, 1992, pp. 206-216.

[5] N. Jarwala, C. Yau, P. Stiling and E. Tammaru, "A Framework for Boundary-Scan Based System Test and Diagnosis", Proc. International Test Conference, 1992, pp. 993-998.

[6] R.E. Tulloss, "IEEE 1149 Standards-Changing Testing, Silicon to Systems", Proc. International Test Conference, 1993, pp. 399-408

[7] C. Pyron and W.C. Bruce, "Implementing 1149.1 in the PowerPC™ RISC Microprocessor Family", Proc. International Test Conference, 1995, pp. 844-850.

[8] L. Whetsel, "Improved Boundary Scan Design", Proc. International Test Conference, 1995, pp. 851-860.

[9] H. Singh, G. Patankar and J. Beausang, "A Symbolic Simulation-based ANSI/IEEE Std 1149.1 Compliance Checker and BSDL Generator", Proc. International Test Conference, 1997, pp. 256-264.

[10] S. Scheiber, *Building a Successful Board-Test Strategy*, ButterWorth-Heinemann, October 1994.