Adapting JTAG for AC Interconnect Testing

Lee Whetsel
Texas Instruments

Abstract

The use of AC coupled interconnects to provide communication paths between devices is increasing. The existing IEEE 1149.1 boundary scan standard [1] (JTAG) has limitations that hinder it from being able to effectively test all AC coupled interconnects. This paper describes a simple enhancement to the JTAG architecture enabling it to operate in new modes facilitating AC interconnect testing.

Description

Figure 1 illustrates a DC interconnect being tested using the existing JTAG standard. The DC interconnect includes an example termination element (R). In functional mode, a core output of a first device passes through a JTAG boundary cell and output buffer to be transmitted through the external DC interconnect to a core input of a second device, via an input buffer and JTAG boundary cell. In test mode, and during the JTAG Extest instruction, the boundary cells are controlled by the JTAG test access port (TAP) and instruction register to operate independent of the IC's core circuitry to allow testing of the DC interconnect. The Extest instruction and its operation within the JTAG architecture [1] is well known.

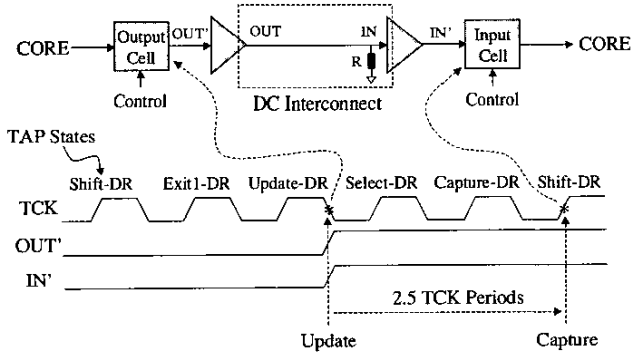

Figure 1

In the timing diagram of Figure 1, the Extest operation is seen to include the steps of shifting data during the Shift-DR TAP state, updating data during the Update-DR TAP state, then capturing data during the Capture-DR TAP state. The data is updated from the output boundary cell on the falling edge of TCK in the Update-DR state and is captured in the input boundary cell on the rising edge of TCK in the Capture-DR state. The update and capture operations are separated by 2.5 TCK periods. This separation does not interfere with the Extest operation, since the data value updated and driven from the output boundary cell is held at the input boundary cell beyond the capture operation. Thus the Extest instruction can test to see if the interconnect is structurally correct.

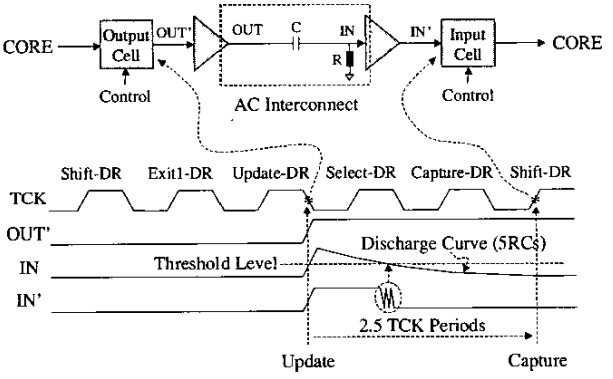

Figure 2

Figure 2 illustrates the example of Figure 1 with an AC interconnect used in place of the DC interconnect. The AC interconnect includes the resistive termination element along with a capacitor (C) located in the connection between the output of the output buffer (OUT) and input of the input buffer (IN). The capacitor serves to block the DC component of the signal transmitted through the interconnect while passing the AC component.

The timing diagram of Figure 2 indicates the problem the Extest instruction has in testing

the AC interconnect. As with the timing example of Figure 1, data is shifted during the Shift-DR TAP state, updated during the Update-DR TAP state, then captured during the Capture-DR TAP state. In this example, a logic one is updated from the output buffer (OUT') on the falling TCK edge of the Update-DR state to drive the input of the input buffer (IN'). Following update, the RC network in the AC interconnect starts to discharge as the pull down resistor bleeds off the voltage from the capacitor to ground. After 5 RC time constants, the voltage present at the input buffer will be approaching ground potential. It is assumed that the 5 RC time constants occur within the 2.5 TCK periods between Update-DR and Capture-DR. Thus the input to the input boundary cell will be at a logic zero during the capture operation, nullifying the test.

From this description it is seen that AC interconnects with small RC time constants will be rendered untestable by the JTAG Extest instruction. This problem is known and certain solutions are being developed by IEEE Standard 1149.6 [2]. Some of the solutions being looked at in IEEE 1149.6 require a significant amount of circuitry be added to boundary input cells of the JTAG architecture. The solution suggested in this paper attempts to solve the problem without adding much, if any, circuitry to the boundary cells of the JTAG architecture. The solution presented is based on three new test instructions that can be added to the JTAG instruction set; a Propagation Test instruction, a Decay Test instruction, and a Cycle Test instruction.

Propagation Test Instruction

The Propagation Test instruction enables testing the propagation of a signal from a device output to a device input through an AC coupled interconnect. Figure 3 illustrates the AC interconnect of Figure 2 being tested using the Propagation Test instruction. The Propagation Test instruction does not require any additional circuitry to be added to the boundary scan cells, i.e. conventional boundary scan cells may be used. The Propagation Test instruction enables use of an addition clock signal within the JTAG architecture, referred to as the Capture Strobe (CS) in the timing diagram of Figure 3.

As seen in the timing diagram of Figure 3, the Propagation Test instruction operates output boundary cells the same way as the Extest instruction of Figures 1 and 2. However, the Propagation Test instruction modifies the way the input boundary cells are operated. In the timing diagram of Figure 3, the Capture Strobe (CS) signal is shown to become active to control the input boundary cells within a programmable window of time shortly after when data is updated from the output boundary cells. Thus the Capture Strobe provides the ability to capture the transient of the propagated signal prior to it being discharged to ground via the RC network. The normal JTAG capture, in the Capture-DR state, is forced to a no-operation (NO-OP) state by the Propagation Test instruction to prevent the normal JTAG capture from overwriting the data captured by the Capture Strobe signal.

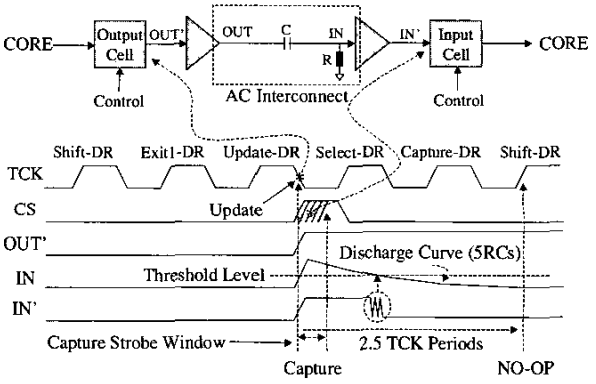

Figure 3

Figure 4 illustrates the JTAG architecture modified to support the Propagation Test instruction. The modifications include adding Capture Strobe circuitry consisting of ClockDR gating and a Delay circuit. When the Propagation Test instruction is loaded into the instruction register, new control signals are output from the instruction register control bus. One signal from the instruction register is a Capture Enable (CE) signal that enables gating of a delayed TCK signal to the ClockDR input of the boundary scan register during the Update-DR state (U). Another signal from the instruction register to the TAP is used to disable the TAP's normal JTAG capture that occurs in the Capture-DR state. The Delay circuit is programmable (via JTAG) to allow varying the placement of the Capture Strobe (CS)

signal edge. During the normal Extest instruction the added circuitry is disabled to provide the normal JTAG Extest method of testing.

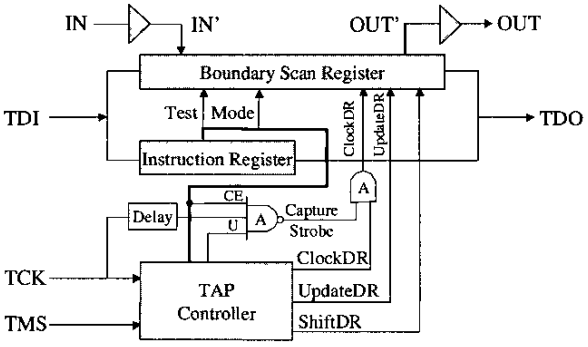

Figure 4

interconnect to capture the steady state response of the interconnect to the updated signal.

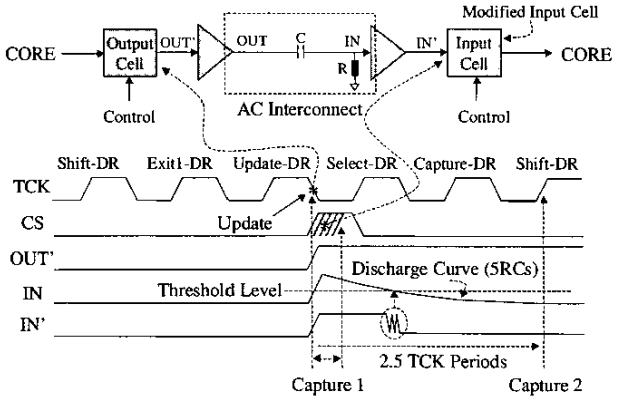

Figure 5

Decay Test Instruction

The Decay Test instruction enables testing the decay of a signal propagated from a device output to a device input through an AC coupled interconnect. Figure 5 illustrates the AC interconnect of Figure 2 being tested using the Decay Test instruction. As seen in the timing diagram of Figure 5, the Decay Test instruction operates identically to the Propagation Test instruction timing diagram of Figure 4 with the exception that it allows the normal JTAG capture operation to occur in the Capture-DR state instead of forcing the NO-OP. The Decay Test instruction reuses Propagation Test Capture Strobe circuitry of Figure 4. As indicated, the Decay Test instruction does require a modification to input boundary cells.

As seen in the timing diagram of Figure 5, data updated from output boundary cells is captured into input boundary cells during the Capture Strobe window as previously described in the Propagation Test instruction. However, unlike the Propagation Test instruction, the Decay Test instruction maintains use of the normal JTAG capture operation in the Capture-DR state. The Decay Test instruction includes the steps of; (1) updating a signal to the input of an AC interconnect, (2) performing a first capture (Capture 1) at the output of the AC interconnect to capture the transient response of the interconnect to the updated signal, and (3) performing a second capture (Capture 2) at the output of the AC

In the timing diagram of Figure 5 it is assumed that 2.5 TCK periods is sufficient time for the AC interconnect to arrive at a steady state prior to the second capture. However, if 2.5 TCKs is not enough time to allow the interconnect to arrive at a steady state, the TAP may be transitioned into the Run Test/Idle state to add more TCK periods between the first and second captures.

Figure 6 illustrates an example of a modified "Control & Observe" input boundary cell that can be used by the Decay Test instruction. The difference between the modified "Control & Observe" input cell of Figure 6 and a normal JTAG "Control & Observe" input cell is that the modified cell contains additional circuitry to detect differences in the logic levels captured during the first and second capture operations. The modified cell operates as normal JTAG input cell during standard JTAG instructions. Modifications to a JTAG "Observe Only" input cell is the same as for the "Control & Observe" input cell. For 'Observe Only" input cells, the modification block is the input cell.

The additional circuitry consists of an XOR gate and a mux coupled to the normal JTAG mux and FF. When the Test input, a new signal from the instruction register, is low, as a result of loading the standard Extest instruction, the input cell operates in the normal JTAG capture and shift modes. When the Test input is high, as a result of loading the Decay Test instruction, the operation

of the input cell is modified to allow the output of the XOR to be captured into the FF during the Capture-DR state instead of the cell's input (IN'). As seen, the output of the XOR is the sum of the present state of the FF and the data value present on the cell's input.

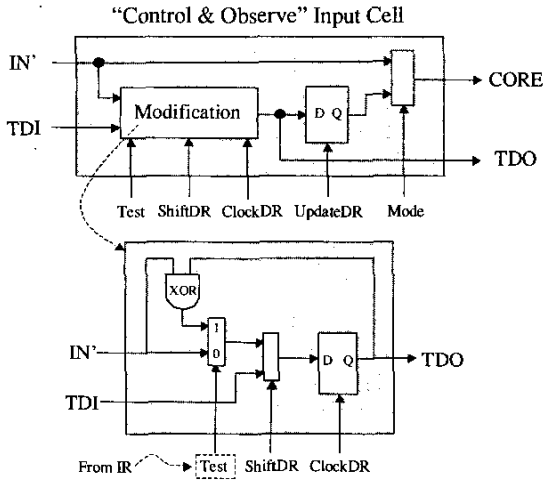

Figure 6

The following describes the sequence of steps for setting up and executing the Decay Test instruction. First a Sample/Preload instruction is loaded and executed to establish initial test data in the input and output boundary cells. Next the Decay Test instruction is loaded and executed to; (a) shift test data into the cells, (b) perform an update and first capture operation, and (c) perform a second capture operation. The first capture operation captures the transient of the updated signal and the second capture operation captures the steady state of the updated signal. If the steady state signal value captured is opposite the transient signal value captured the test passes, otherwise it fails.

In the first step, data preloaded into the input and output cells will be data that initializes the cells for the first test data pattern to be transmitted through an AC interconnect. For example, if the AC interconnect of Figure 5 is being tested, the output cell will be initialized to output a logic zero to the AC interconnect whenever the Decay Test is loaded. Outputting a logic zero to this particular AC interconnect places the RC network in the expected steady state condition (i.e. a logic zero will be input to the input boundary scan cell). By establishing the steady state logic zero condition in the interconnect, a logic one can be transmitted through the interconnect from the output cell to the input cell during the update operation. While a logic zero is the steady state for this particular interconnect, a logic one may be the steady state for a another interconnect type.

The first capture that follows the update operation captures the XOR of the updated logic one transient (a logic one) and the scanned in present state of the input cell FF (a logic zero), making the value in the FF a logic one. The second capture that occurs in the Capture-DR state captures the XOR of the updated logic one steady state (logic zero) and the previously captured state of the FF (logic one), making the value in the FF remain a logic one. Scanning out a logic one from the input cell FF verifies that the transient and steady state captures were of opposite values and the test passed. Scanning out a logic zero from the input cell FF indicates that the transient and steady state captures were of the same value and the test fails. If the test fails, the Decay Test can be repeated with a longer time interval placed between the first (transient) and second (steady state) captures by entering the Run Test/Idle state for a number of TCKs to increase the separation between the two captures.

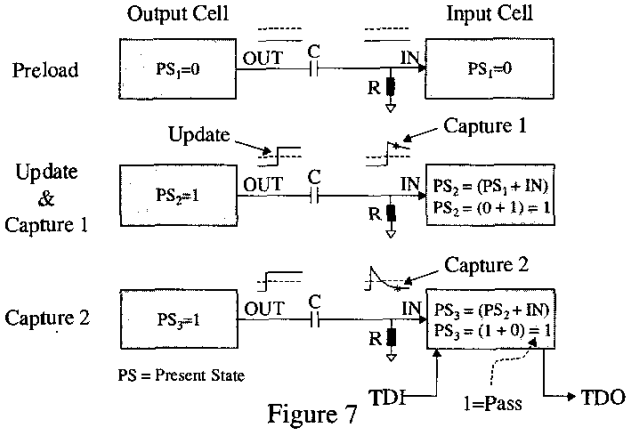

Figure 7

Figure 7 illustrates an example flow of the Decay Test instruction being used to test a good AC interconnect between an output and input cell. In Figure 7, the progression of the Decay Test operation is shown to occur in steps. The first step is to preload the cells using Sample/Preload

to establish an initial present state (PS) test condition. The second step loads the Decay Test instruction, shifts in test data to be updated, and performs the update and first capture (Update & Capture 1) operation. The next step performs the second capture (Capture 2) operation then shifts out the test results. Since the input (IN) to the input cell during first and second capture operations is different the test passes.

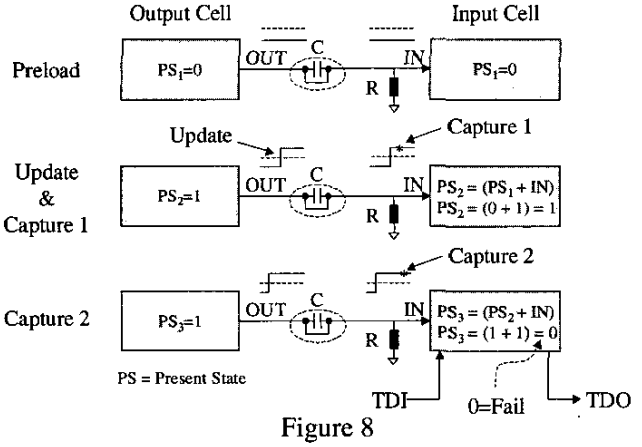

Figure 8

Figure 8 illustrates an example flow of the Decay Test instruction being used to test a faulty AC interconnect between an output and input cell. The fault in the interconnect is the short circuit across the capacitor C. Other than the short circuit fault the interconnect of Figure 8 is the same as the one in Figure 7. The same test steps of Figure 7 are repeated. As seen, with the short across the capacitor the interconnect cannot arrive at the expected the steady state and the test fails since the input cell captures the same value during the first and second capture operations.

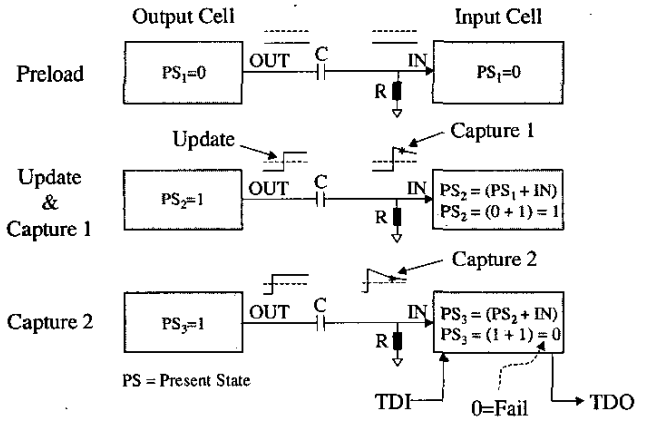

Figure 9

Figure 9 illustrates an example flow of the Decay Test instruction being used to test an AC interconnect between an output and input cell. The fault in the interconnect is due to wrong valued R and/or C components. Again, the same test steps of Figure 7 are repeated. As seen, and due to a wrong R and/or C value, the expected steady state condition at the input (IN) of the input cell is not achieved, and the test fails because the input cells capture the same value during the first and second capture operations. Re-testing and increasing the delay between the first and second capture operations, via entry into the Run Test/Idle state, will allow the test to pass and reveal the reason for the failure.

Cycle Test Instruction

The Cycle Test instruction enables testing of signal cycles propagated from a device output to a device input through an AC coupled interconnect.

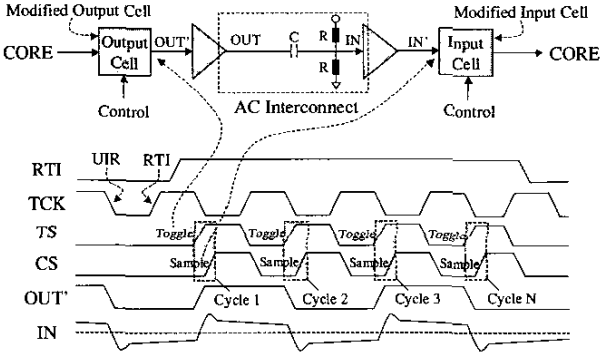

Figure 10

Figure 10 illustrates an AC interconnect being tested using the Cycle Test instruction. The AC interconnect differs from the previous AC interconnect in that it contains two termination resistors, one to supply and the other to ground. The steady state voltage of the node (IN) between the two resistors will be a voltage less than the supply but greater than ground, as determined by the resistor values. As seen, the Cycle Test instruction requires modifications to both input and output boundary cells. The output boundary cell is modified to include a toggle output mode and the input boundary cell is modified to include the previously described Decay Test modification, plus a pass/fail flag memory. These cell

modifications are described in more detail in regard to Figures 12 and 13.

As seen in the timing diagram of Figure 10, the Cycle Test instruction begins operating when the TAP enters the Run Test/Idle (RTI) state, indicated by RTI going high. The RTI signal comes from the TAP and indicates when the TAP is in the Run Test/Idle state. Prior to entering the RTI state, a Sample/Preload operation will have initialized the input and output cells. In the timing diagram, a logic zero will be output from the output cells when the Cycle Test instruction is updated from the instruction register in the Update-IR (UIR) state.

During Cycle Test instruction operation, the output cell is enabled by a Toggle Strobe (TS) signal to alternate (toggle) the input of the interconnect, while the input cell is enabled by the Capture Strobe signal to capture (sample) the output of the interconnect. The waveform on IN of the timing diagram is representative of the expected output response of the interconnect. As seen in the diagram, the sample occurs during the transient portion of each toggled signal.

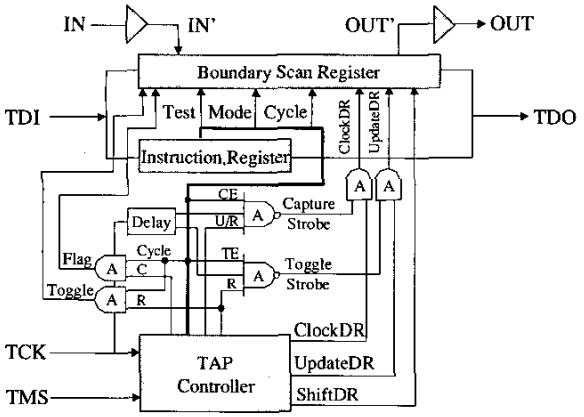

Figure 11

Figure 11 illustrates the JTAG architecture modified to support the Cycle Test instruction. The modifications include adding gating for the Toggle Strobe to be applied to the UpdateDR input of the boundary scan register, and gating to provide Flag and Toggle control inputs to the boundary scan register. When the Cycle Test instruction is loaded into the instruction register, new control signals are output on the instruction register control bus. One signal is a Toggle Enable (TE) signal that enables gating a delayed TCK signal to the UpdateDR input of the boundary scan register during the Run Test/Idle state (R). Another signal is a Cycle signal used to enable the Flag and Toggle inputs to the boundary scan register during the Capture-DR (C) and Run Test/Idle (R) states, respectively. Also, the previously described Capture Enable (CE) signal is set to enable the Capture Strobe signal to drive the ClockDR input of the boundary scan register during Update-DR and Run Test/Idle states (U/R). The Delay circuit provides a longer delay on the TCK driving the Capture Strobe gating than the delay driving the Toggle Strobe gating, to insure the toggle operation precedes the capture operation. During standard JTAG instructions the added circuitry is disabled to provide the normal JTAG operation of the boundary scan register.

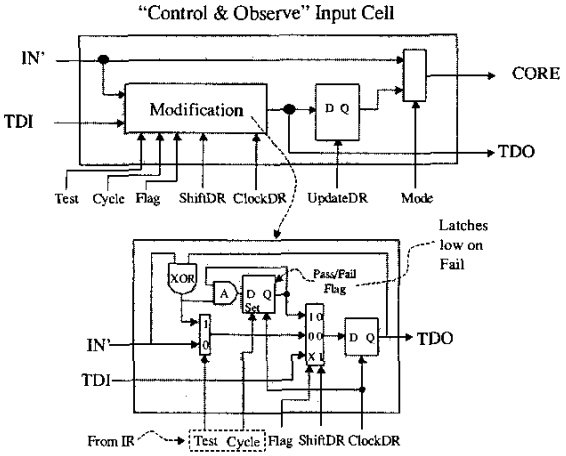

Figure 12

Figure 12 illustrates an example of a modified "Control & Observe" input boundary scan cell for the Cycle Test. The input cell includes the Decay Test modifications (i.e. XOR and mux of Figure 6) plus a Pass/Fail Flag memory. Also the normal JTAG mux is increased to accept a third input from the Pass/Fail Flag. The FF of the Pass/Fail Flag receives the Cycle signal from the instruction register on its Set input to initialize the Flag to a logic one (pass condition) prior to performing the Cycle Test. The Test signal is low during Cycle Test to allow the input cell FF to receive the IN' input. During Cycle Test and while the TAP is in Run Test/Idle,

the Flag FF is operated by Capture Strobe, via ClockDR, to store the XOR of the present state of the input cell FF and the logic level on IN'. As in Figure 6, for "Observe Only" input cells, the modification is the cell.

During Cycle Test, as long as opposite logic values are input at the cell's IN' input, the Flag FF will maintain a logic one (pass) value. If the logic values on the IN' input remain the same, the Flag FF will latch to a logic zero (fail) value. After the Cycle Test is complete, the TAP leaves Run Test/Idle to perform a data register scan operation. During the data register scan operation the value in the Flag FF is selected for capture during the Capture-DR state by the Flag signal input from the Flag gate of Figure 11. Again, a captured logic one indicates a pass and a logic zero indicates a fail.

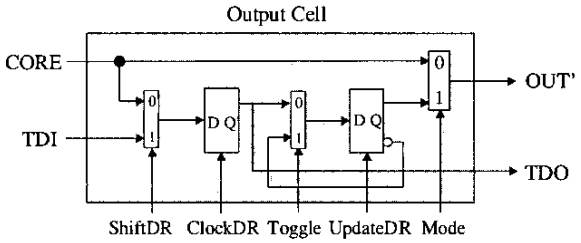

Figure 13

Figure 13 illustrates an example of a modified output boundary scan cell for the Cycle Test. The output cell includes an additional mux between the shift and update FFs. During standard JTAG instructions, the Toggle signal from the Toggle gate of Figure 11 is set to couple the shift FF to the update FF. During Cycle Test and while the TAP is in Run Test/Idle, the Toggle signal is set to couple the inverted output of the update FF to the input of the update FF, enabling it to toggle during each Toggle Strobe input on the UpdateDR signal. This output toggle operation is shown in the timing diagram of Figure 10. When the Cycle Test is completed and the TAP enters the data register shift operation the output cell operates as a normal JTAG output boundary cell to capture, shift and update data.

Testing Differential AC Interconnects

Figure 14 illustrates an example of testing a differential AC coupled interconnect using a conventional JTAG input and output cell. Using a first Propagation Test operation, the output cell updates a logic one (from a zero) to the differential output buffer to stimulate the differential AC interconnect. The input cell captures the output of the differential input buffer during the transient state of the differential input signals IN & IN*, as in Figure 3. Using a second Propagation Test operation, the output cell updates a logic zero (from a one) to the differential output buffer to stimulate the differential AC interconnect. The input cell captures the output of the differential input buffer during the transient state of the differential signals IN & lN*. Decay and Cycle Test operations may also be performed, using appropriately modified input and output cells.

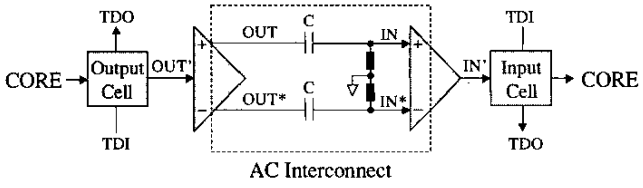

Figure 14

Figure 15 illustrates an example of testing a differential AC coupled interconnect using a conventional JTAG output cell and three input cells.

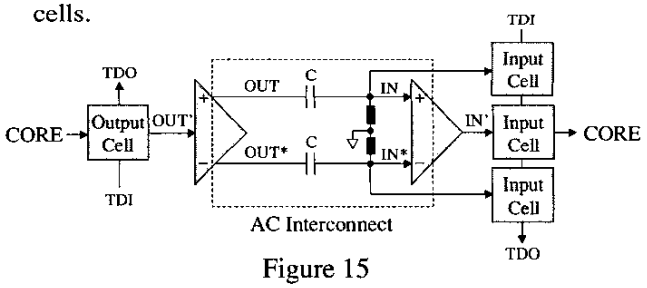

Figure 15

As in Figure 14, one input cell is coupled to the output of the differential input buffer. The two other cells are coupled to the individual leads (IN & IN*) of the differential interconnect. Using a first Propagation Test operation, the output cell updates a logic one to the differential output buffer to stimulate the differential AC interconnect. The input cells capture the IN lead signal, the IN* lead signal, and the output of the differential input buffer during the transient state. Using a second Propagation Test operation, the output cell updates a logic zero to the differential

output buffer to stimulate the differential AC interconnect. The input cells capture the IN lead signal, the IN* lead signal, and the output of the differential input buffer during the transient state.

The additional input cells on the IN and IN* leads allow the testing of the differential interconnect to be independent of the differential input buffer. For example, the Propagation Test described in Figure 14 is dependent on the differential input buffer being good. So the advantage of the three input cell approach is that it allows direct testing of the differential interconnect paths and the ability to detect a faulty differential input buffer. Decay and Cycle Test operations may also be performed, using appropriately modified input and output cells.

IEEE 1149.6 Testing Example

Figure 16 illustrates an example of how AC coupled differential interconnects may be tested using the IEEE 1149.6 standard (2). The 1149.6 standard is based on detecting signal transitions to test AC coupled interconnects between ICs. The transition detection method requires use of a different type of input test cell, as shown in Figure 16. The details of the 1149.6 input test cell are given in a paper by Eklow, Barhardt, and Parker (2). 1149.6 also suggests use of an output test cell adapted for stimulating AC coupled interconnects during test, again detailed in the paper (2).

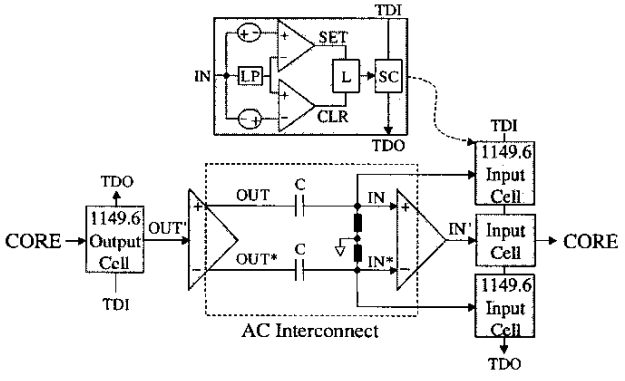

Figure 16

According to 1149.6, an Extest_Pulse instruction may be loaded and operated to cause the 1149.6 output cell to input stimulus signals on the OUT and OUT* inputs to the differential interconnect of Figure 16. During the operation of the 1149.6 output cell, the 1149.6 input cells operate to detect signal transitions on the IN and IN* outputs of the differential interconnect. A high going transition will cause a latch (L) within the 1149.6 input cell to be set, and a low going transition will cause the latch to be cleared. After the Extest_Pulse operation is complete, a TAP controlled data scan operation is performed to load the latch value into a scan cell (SC) to be shifted out for inspection.

1149.6 also provides an Extest_Train instruction that allows P1149.6 output cells to operate to stream signals into the OUT and OUT* inputs of the differential interconnect while 1149.6 input cells operate to detect signal transitions at the IN and IN* outputs of the AC interconnect. The Extest_Train instruction is targeted for testing special case AC interconnects that must be energized with signal activity before they can operate properly.

1149.6 is advantageously based on 1149.1 and is compliant with all test modes and instructions of that legacy test standard. The design of 1149.6 input cells and their ability to robustly detect signal transitions on a stimulated AC coupled interconnect is the key contribution of 1149.6.

One limitation that appears to exist in 1149.6 is the inability to determine when the signal transition is actually detected. For example, in the timing diagram of Figure 11 of the Eklow paper (2), the final stimulation of the AC interconnect occurs on the falling edge of TCK in the SelectDR state. In response to this final stimulation, the latch (L) of the 1149.6 input cell is either set (for a rising transition) or cleared (for a falling transition). On the rising edge of TCK in the CaptureDR state the scan cell (SC) of the 1149.6 input cell captures the transition value from the latch (L) to be shifted out. The transition that established the latched value may occur anywhere within the time frame between the falling edge of TCK in the SelectDR state (final stimulation point) and the rising edge of TCK in the CaptureDR state (scan cell (SC) capture point). The interval of this 1½ TCK time frame is governed by the frequency of the TCK. For

example, a 20Mhz TCK produces a 75ns interval, a 10Mhz produces a 150ns interval, and a 5Mhz TCK produces a 300ns interval.

Figure 17 illustrates an AC interconnect to be tested using an 1149.6 output cell and input cell. For simplification, the input and output buffers are not shown. The output cell outputs final stimulus to the AC interconnect and the latch of the input cell stores the transition response. Two different timings (Time1 & Time2) are shown whereby the response transition is latched (L) following the stimulus. In each case the latching occurs within the 1½ TCK time frame, so both Time1 and Time2 tests pass. However, for proper functional operation of the circuits connected to the AC interconnect the transition needs to occur within the timing indicated by the dotted line box. Since the transition of the passing Time1 test is within the box, the test insures the circuits will functionally operate. However since the transition of the passing Time2 test is not within the box, the test does not insure that the circuits will functionally operate. Thus 1149.6, like 1149.1, only tests for the structural correctness of circuit interconnects. The ability of the circuit interconnects to operate at functional speeds is not tested.

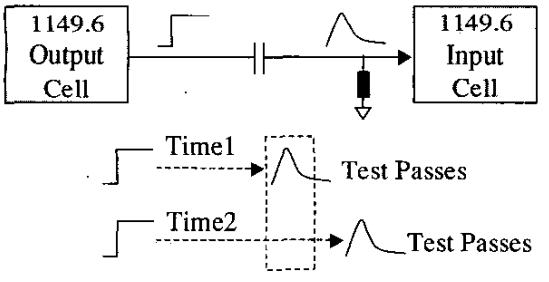

Figure 17

Since the approach described in this paper uses clock edge placement to test AC coupled interconnects, it has the potential to test not only the structural correctness of an interconnect, but also the ability of the interconnect to meet the functional timing required by the circuits communicating over the interconnect.

Figure 18 illustrates the AC interconnect of Figure 17 being tested using the Propagation Test instruction and conventional JTAG cells. It is assumed that the TCK Delay circuit of Figure 4 has been designed to allow programming the placement of the Capture Strobe clock edge (ClockDR). Thus the Capture Strobe edge may be positioned for testing that a signal output from the output cell arrives at the input cell within the functionally required dotted box time frame. With the Capture Strobe edge placed to occur within the dotted box time frame an interconnect exhibiting the timing of Time1 will pass while an interconnect exhibiting the timing of Time2 will fail. Thus the advantage of using an edge operated test as described in this paper over a transition operated test as described in 1149.6 is the ability of the edge operated test to test with timing that verifies both the structural and functional correctness of an AC interconnect.

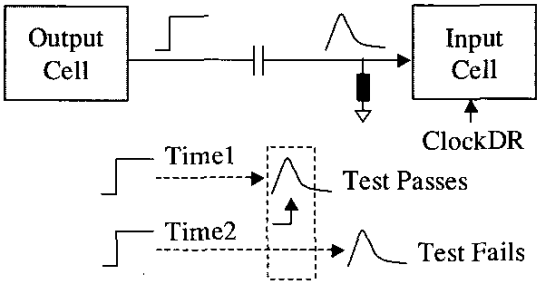

Figure 18

As seen in Figure 19, devices may have different AC interconnect busses A, B, and C, each with different functional timing box requirements.

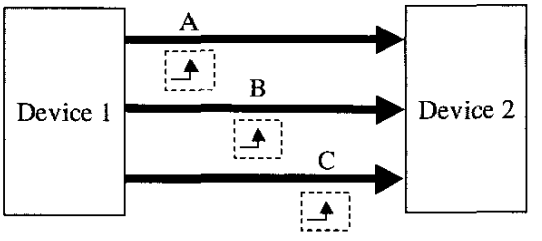

Figure 19

To test the example of Figure 19 for structural and functional correctness using the Propagation Test instruction requires three separate tests to be executed. The first test is executed with the edge placed in the timing box of bus A, the second test is executed with the edge placed in the timing box of bus B, and the third

test is executed with the edge placed in the timing box of bus C. Separate testing of each bus A, B, and C may also be required for structural only testing if the edge must be positioned differently for each bus.

Testing of the example in Figure 19 using 1149.6 can be accomplished with a single Extest_Pulse test operation since 1149.6 testing is based on structural testing using transition detection. Thus the advantage 1149.6 has over the approach of this paper, when only structural interconnect testing is required, is that a single Extest_Pulse operation can be used to test all AC interconnects between devices at the same time.

In Figure 19, if Device 1 contains only 1149.1 circuitry and Device 2 contains 1149.1 circuitry plus the additional circuitry mentioned in this paper, it is possible to operate the Propagation and Decay Test instructions in Device 2 while operating an Extest instruction in Device 1. This is possible since Propagation and Decay test operations in Device 2 only require an Extest update operation from Device 1. Thus Propagation and Decay Test instructions are compatible with existing 1149.1 devices.

Summary and Conclusion

Each of the test instructions described in this paper provides a simple and unique way of testing AC interconnects.

The described Propagation Test instruction enables testing of an AC interconnect's ability to pass signal transients between a driving and receiving device. This instruction requires only the Capture Strobe producing circuitry be added to the JTAG architecture. Conventional boundary scan cells are used by this instruction. The Propagation Test instruction in one device is test plug and play compatible with an Extest instruction in another device.

The described Decay Test instruction enables testing an AC interconnect's ability to pass signal transients from a driving device to a receiving device, as well as testing the interconnect's ability to arrive at an expected steady state during each signal being passed. This instruction requires the Capture Strobe producing circuitry and modification to input boundary cells that are coupled to AC interconnects. In addition to testing the interconnect, this instruction also verifies that correct value R and C components exist within the interconnect. The Decay Test instruction in one device is test plug and play compatible with an Extest instruction in another device.

The described Cycle Test instruction enables testing an AC interconnect's ability to continuously pass signal transients from a driving device to a receiving device. This instruction requires the Capture Strobe and Toggle Strobe producing circuitry and modification of both input and output boundary cells coupled to AC interconnects. Since the Cycle Test can be set to run continuously in the Run Test/Idle state, it is more likely to detect the more difficult of faults in an AC interconnect, i.e. those occurring at random as a result of noise, cross-talk, voltage fluctuations, temperature, etc.

The design of the programmable Delay circuit of Figure 4 is key to making these test instructions work properly. The Delay circuit needs to be able to place the Capture Strobe edge of any of the instructions within a timing window that allows sampling the expected AC interconnect output response to a given input stimulus. Programming of the Delay circuit can be achieved by simply providing an instruction to select a data register in the Delay circuit, then scanning the data register to select a desired delay tap point to use for the test instructions.

References

(1) IEEE Std 1149.1, A Standard Test Access Port and Boundary Scan Architecture.

(2) Eklow, Barnhardt, Parker, IEEE P1149.6: A Boundary Scan Standard for Advanced Digital Networks, Paper 37.1 2002 IEEE International Test Conference, pp 1056-1065.