A Unified DFT Architecture for use with IEEE 1149.1 and VSIA/IEEE P1500 Compliant Test Access Controllers

Bulent I. Dervisoglu
Cadence Design Systems
2655 Seely Avenue
San Jose, CA 95134
(408)-895 2184

bulent@cadence.com

ABSTRACT

This paper discusses some of the critical issues that may prevent IEEE P1500 from becoming an acceptable standard and offers some suggestions for their solution. In particular, the inadequacy of the proposed P1500 and the VSIA solutions in handling hierarchical implementations is addressed. Support for hierarchical implementations is seen as an essential feature in a test access methodology that is intended for use in System on a Chip (SoC) designs. The author is actively pursuing some of these solutions through the working groups.

1. INTRODUCTION

IEEE P1500 Standard for Embedded Core Test (SECT) has been proposed [1,2,3] as a standard for testing IP (Intellectual Property) cores in a SoC (System on a Chip) environment. To this end, the IEEE P1500 Working Group was formed in 1995. Separately, the Virtual Socket Interface Alliance (VSIA) has been working to develop a common methodology for test development to facilitate easier transfer of test related information between the Virtual Component (VC) provider and the VC integrator. Working groups of the P1500 and the VSIA have been coordinating their efforts in order to arrive at a solution that satisfies VSIA member requirements as well as can be endorsed as an IEEE standard. Due to close cooperation between these two organizations, the VSIA Test Access Architecture [4] and the P1500 SECT [5] are similar in their technical strategy and details.

IEEE P1500 is aimed at developing a standard for embedded core test that enables core designers and integrators to develop and use IP from different sources. To this end, the P1500 Working Group is developing a common Serial Interface Layer (SIL) architecture. Despite their many common features, there are certain differences between the VSIA and P1500 proposals. Basic concepts in both architectures are derived from the IEEE 1149.1 standard (including work done by the IEEE P1149.2 WG [6,7]). However, both solutions are incomplete in terms of providing a self-contained methodology for use under the IEEE 1149.1 framework. This is a major concern since incomplete buy-in from or difficulty to work together with an existing/related IEEE standard may delay acceptance and deployment of the new methodology. The VSIA Test Control Block (TCB) can be seen as a subset of the P1500 SIL. Whereas the P1500 SIL is mandatory the VSIA TCB is an optional feature. In applications where the VSIA TCB may not be implemented a standardized set of internal test signals (which are the outputs of TCB) must be provided as test control inputs to that core. Thus, issue of mandatory versus optional TCB becomes mostly an issue of overall implementation philosophy. This issue is not covered here. Furthermore, this paper addresses only the test access mechanism made possible by the proposed architectures but does not discuss methods for testing the individual cores or the overall SoC component.

This paper presents the author's own views and proposals that are not necessarily the official position of the VSIA Test DWG or the Working Groups of IEEE P1500 TECS.

2. BACKGROUND ON IEEE P1500

The P1500 Working Group has organized its efforts along two separate but closely linked activities. The first activity is known as the Scalable Test Architecture and is being developed by the Core Test Action Group (CTAG) sub-team of the P1500 Working Group. The second activity is aimed at developing a Core Test Language (CTL) which will be used to describe the DFT architecture and the test data (i.e. patterns) that have been developed to test the target core. The CTAG sub-team has adopted the objective that the P1500 Scalable Architecture Standardization shall be applicable to hierarchical cores and shall also define how cores with 1149.1 can become P1500 compliant.

The Serial Interface Layer defined by CTAG provides a standard access mechanism to control the test features of a P1500-compliant core. The SIL provides architectural features that are similar to those in the IEEE 1149.1 standard [1] but it excludes the familiar 16-state Finite State Machine (FSM) that is a mandatory element within IEEE 1149.1. The SIL contains the following architectural elements:

- Mandatory Wrapper Instruction Register (WIR) which is used to select/enable particular test features,
- Mandatory Wrapper Bypass Register (WBYPASS Register) to facilitate pass-through access through individual cores,
- Mandatory Wrapper Interface Port (WIP) that comprises control and data signals for accessing the DFT features of the target core.
- Optional Wrapper Boundary Register (WBR) to control and observe core logic input/output signals. The presence (or absence) of the WBR is used to differentiate between the two

levels of compliance, called *IEEE P1500 Wrapped Core* and *IEEE P1500 Unwrapped Core*, respectively.

In essence, the P1500 SIL architecture has been derived from the architecture of the IEEE 1149.1 TAP Controller by removing the TAP FSM as had been proposed earlier by the IEEE 1149.2 Working Group [7]. In particular, the P1500 SIL does not employ a finite state machine (FSM) to identify the test controller state and contains several test control signals that enable specific actions in a manner that mimics the actions of the IEEE 1149.1 TAP FSM. Thus, test data actions in the P1500 SIL are categorized as either Wrapper Data-Register (WDR) or Wrapper Instruction-Register (WIR) actions. Basic test actions comprise Capture, Shift or Update using the indicated register. Figure 1 shows the block diagram of the P1500 SIL architecture and demonstrates its relationship to the IEEE 1149.1 TAP Controller. Whereas the P1500 SIL is targeted at individual cores in a target SoC, the architecture scales to work with multiple cores and the component-level test controller (i.e. the TAP Controller). This paper addresses the problems associated in linking multiple P1500 SILs together and shows that, without modifications, the existing P1500 architecture is inadequate to satisfy some needs. Specific extensions to the P1500 SIL are proposed below.

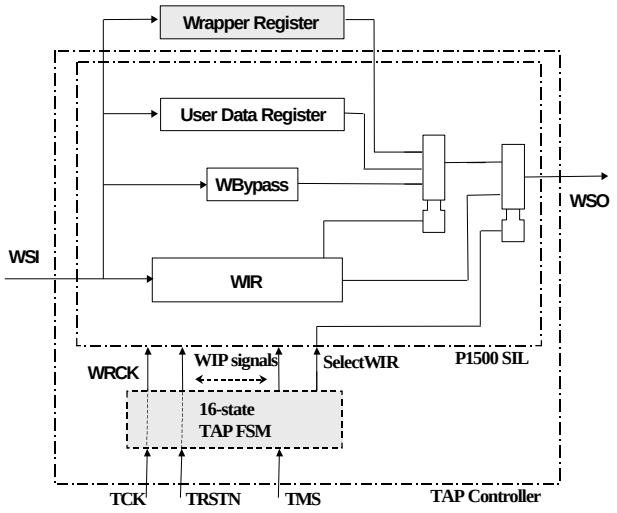

Figure 1. P1500 SIL and the TAP Controller architecture

There are many architectural similarities between the P1500 SIL and the VSIA Test Control Block (TCB). Whereas the P1500 SIL has eliminated the IEEE 1149.1 TAP FSM block, the VSIA TCB goes one step further and makes the WIR become an optional element. Thus VSIA seeks a standardization of the control signals instead of developing standard instructions that represent codified (i.e. instruction opcode) versions of the control signals that become shared among multiple VC's.

3. THE P1500 and VSIA TIMING MODELS

Both the P1500 and VSIA are vague about the use of clock signals to synchronize test activities inside the SIL or the TCB. P1500 requires a dedicated test clock (WCRK) to control the WIR and WBYPASS registers and allows other clocks to be used to control the WBR or other WDR elements. This is in contrast to the VSIA specification where a single test clock (TCLK) controls the TCB, which contains only the Instruction Register. Other registers, including the Bypass and the Wrapper can be clocked by a different signal even though VSIA recommends using the TCLK to operate these registers also. Furthermore, the VSIA specification adopts the same timing rule as in IEEE 1149.1 whereby serial output changes occur only on the falling-edge of the test clock (TCLK). This is required in the P1500 only for the WIR and the WBYPASS but not for any other WDR (including the Wrapper Register). Whereas both the P1500 and the VSIA make reference to a single, dedicated, test clock (WRCK) a *de-facto standard* timing model for P1500 can be stated even where multiple clock signals are used. This can be achieved by defining separate clock events (clock edges or logic states of multiple clock pulses) that correspond to the rising-edge and falling-edge of *WRCK*, respectively. Whereas present P1500 rules do not state exact timing relationships among the WIP signals, the following definitions are supplied to provide timing-correct behavior of multiple-SIL (or SIL and TAP) configurations.

- *CaptureWR:* Causes data to be captured into the target WDR (which is identified by the presently active instruction) or the WIR, on the next rising-edge of *WRCK*.

- *ShiftWR:* Causes present contents of the target WDR (which is identified by the presently active instruction) or the WIR, to be shifted in direction from the Wrapper Serial Input (*WSI)* towards Wrapper Serial Output (*WSO)* on the next rising-edge of *WRCK*.

- *UpdateWR:* Causes data to be updated into the parallel-update stages of the target WDR (which is identified by the presently active instruction) or the WIR, on the next falling-edge of *WRCK*.

- *SelectWIR:* Selects the WIR or the target WDR (which is identified by the presently active instruction) for action as controlled by the *CaptureWR, ShiftWR* or the *UpdateWR* signals, as described above.

Additionally, it is necessary to establish set-up and hold conditions that must be satisfied among the various WIP signals. This is an area where present rules in P1500 need to be tightened and made more specific. For example, since the clock input for the WIR may include the term "*WRCK* & *ShiftWR*" it is necessary that *ShiftWR* must be ready (i.e. set-up) before the next rising-edge of *WRCK* and its value should not change (i.e. hold) within the vicinity of the rising-edge of *WRCK*. In general, the following strategy can be employed:

- WIP signals CaptureWR, ShiftWR and SelectWR shall change only on the falling-edge of the WRCK and UpdateWR shall change only on the rising-edge of WRCK. Additionally, it is recommended that WRCK should be a 50% duty-cycle, free-running clock signal that may also be used as the TCK signal for the chip-level TAP Controller.
- The *WRSTN* signal causes an asynchronous reset of the SIL and also enables the SIL for synchronous behavior when *WRSTN* becomes de-asserted. Since, following a reset

condition, the first synchronous activity in the SIL may either be a capture or shift operation, *WRSTN must be de-asserted on (or before) the falling-edge of WRCK*.

4. CHIP-LEVEL ACCESS to the P1500 SIL

The IEEE P1500 SIL has been designed to make it possible to connect multiple SILs in series by cascading their WSI and WSO terminals and using a shared set of *WRSTN, WRCK, ShiftWR, CaptureWR, UpdateWR* and the *SelectWIR* signals to control their combined operation. The serial connection among several SILs creates a composite SIL that remains P1500 compliant. Due to similarities between the P1500 and 1149.1 TAP architectures, it is also possible to connect the (composite) SIL in series with a component-level TAP. However, whereas the resulting arrangement of registers can be made to operate together, this configuration also violates the IEEE 1149.1 standard rules for the top-level IC component. This is a serious issue since, unless it is remedied, it prevents using IEEE P1500 compliant cores to build IC components that are compliant with IEEE 1149.1 standard.

To understand why the serial connection between the P1500 SIL and the 1149.1 TAP breaks the standard operation of the IEEE 1149.1 TAP Controller, first assume that the (optional) IDCODE instruction has been provided in the top-level 1149.1 test controller. This requires that the serial connection between the SIL and the TAP should be made such that the TAP controller comes after the SIL. This is necessary so that the contents of the TAP's registers appear at the component's TDO output pin first. Then, considering their serial operation, there are two possible ways to control the TAP and the SIL registers in tandem.

a) The SIL and the TAP are operated together so that their respective Instruction Registers (i.e. the WIR and the IR) perform jointly. This implies that the *SelectWIR* signal is asserted whenever the TAP FSM is in any one of the states where IR is selected (e.g. *Capture-IR, Shift-IR, Update-IR).* The corollary is that when *SelectWIR* is de-asserted the Data Registers in the SIL and the TAP must be operated jointly. This causes the effective lengths of the TAP Data Registers appear longer as seen external to the IC component. In the first place, this makes it necessary to reflect the details of the SIL to become visible through the component-level BSDL [8] so that external EDA tools can deal with the longer length of the composite Data Registers. This is highly undesirable, especially when the target SoC device contains several cores, each with its individual SIL. In addition to causing unnecessary detail to be revealed at each level of integration this causes additional work in test pattern reuse. Furthermore, whereas it may be possible to modify the component's BSDL file to specify the *effective* length for some of its registers this is not possible for the component's BYPASS and the IDCODE registers since these registers are required to contain only 1 and 32 bits, respectively.

b) SIL registers may be operated upon as Data Registers within the 1149.1 architecture. In this case, the WIR of the SIL will be updated when the TAP FSM is in *Update-DR* state. Since updating the WIR with a new instruction may select a different Wrapper Data Register such that, as seen through the 1149.1 TAP, the selected data register and its length may change even though a new instruction has not been (serially) loaded into the TAP's IR. This is a violation of the IEEE 1149.1 standard.

The above discussion shows that, without some changes, it is not possible to connect and operate the P1500 SIL in series with a component's top-level TAP Controller and remain compliant with the IEEE 1149.1 standard at the IC component level. This situation is avoided in the VSIA architecture by using a separate pair of serial input/output ports for the Wrapper (WP_SI/WP_SO) and the TCB (TC_SI/TC_SO), respectively. In this case, loading a different instruction into the TCB does not cause its register length to change since all data registers (including the bypass register) appear between WP_SI/WP_SO ports instead of TC_SI/TC_SO. However, like the P1500 SIL, the VSIA TCB architecture does not explicitly support hierarchical connections among multiple test controllers.

5. ENHANCING the P1500 SIL

This section proposes enhancements to the proposed IEEE P1500 SIL architecture to allow its use together with an existing component-level IEEE 1149.1 TAP Controller. Furthermore, the proposed changes add flexibility to the P500 SIL and allow its usage in hierarchical implementations that may also include other (legacy) embedded TAP Controllers.

5.1 The P1500 Reset Issue

In its present form, the P1500 SIL architecture does not provide differentiating between the *Reset* state and the presence of the W*Bypass* instruction in the WIR. This is due to the fact that the SIL does not contain an FSM to indicate different test controller states, including the *Test-Logic-Reset* state that is present in the 1149.1 TAP FSM. Consequently, the P1500 *WBypass* instruction serves double duty as both the *normal* state of the core as well as being used as a benign test instruction. Indeed, originally, the P1500 Working Group had accepted the notion of a *Core Enable* instruction which, apparently later has become merged with the *WBypass* instruction. The Test Control Block (TCB) for the VSIA Test Access Architecture also defines the *Normal* mode and states that while the TCB is in the *normal* mode *None of the Design For Testability (DFT) structures in the VC is activated*. Thus, whereas the VSIA test architecture makes the distinction between *normal* mode and all other *DFT-enabled* modes the same is not true for the P1500 SIL architecture. The P1500 SIL contains the *WRSTN* signal to assert an asynchronous reset condition. It is possible to use a common signal, e.g. the TAP's TRSTN, to reset all test controllers in the target SoC device. However, it is not immediately obvious how to generate a reset signal from the present SIL so it affects only those SILs which are directly under it (i.e. within a component's hierarchy) without issuing a chip-level reset through the 1149.1 TAP Controller. The presence of the *WBypass* instruction inside the WIR alone is not sufficient to conclude that a reset signal should be generated by the present SIL. Furthermore, if the reset signal is made conditional on the *WBypass* instruction then there is no way to de-assert the reset signal without loading a different instruction into the WIR. The difficulty in asserting/de-asserting a reset signal from a given SIL may create difficulties in implementing SoC-level DFT architectures where multiple cores are operated together to perform various different test functions. This is especially true if/when some of the embedded test controllers may be based on IEEE 1149.1 TAP Controller architecture.

5.2 Support for Hierarchy

Despite the adoption of a basic principle that the P1500 standard should be applicable to hierarchical cores the P1500 SIL architecture, in its present form, provides only for a serial connection among multiple SILs even though the core logic may have been implemented in a hierarchical manner. Thus, multiple P1500 SILs are connected to form a composite SIL much the same as multiple IEEE 1149.1 compatible IC components are connected in series to form a composite component on a Printed Circuit Board (PCB) assembly. A series connection among the SILs may be appropriate for cores at the same level of hierarchy but is not well suited for linking together SILs, which are at different levels of hierarchy.

Present thinking within the IEEE 1500 CTAG is that when multiple SILs are connected in series their SelectWIR lines should be operated together so that their WIR are accesses in series as a single, composite Instruction Register. A direct implication of this requirement is that internal details of cores must become visible at every level of the hierarchy. For example, consider a top-level core containing a SIL that has a n-bit WIR and assume that a particular opcode has been assigned to enable a test instruction that is specific and unique to that core. Now, further assume that another VC (such as an additional RAM block) has been embedded in that core. In this case, the effective WIR length for the SIL of the top-level core becomes "n+k", where "k" is the WIR length for the embedded VC. This means that all opcode assignments for the SIL of the top-level core must reflect not only the length but also the bit assignments for the WIR of the embedded core. Apart from the obviously undesirable side effects that this creates there is yet another issue that may render this solution unacceptable, as described below.

Direct reset of test controllers for hierarchically embedded cores, which have been connected in serial fashion, can be achieved using WRSTN as the common reset signal. Typically, the WRSTN signal is driven in common to all SILs and the chip-level TAP controller. Synchronous reset of the SILs can be achieved by creating a synchronous event, such as the loading of the *WBypass* instruction into all of the WIRs. However, this makes it impossible to "park" the core in a predetermined test mode and still bypass its SIL in order to access other SILs. This illustrates the difficulty in generating a localized reset signal, which is necessary in hierarchical implementations.

In comparison, the Hierarchical Test Access Controller (H-TAP) described in [9] offers an overall reset mechanism so that whenever a parent Test Access controller becomes reset (i.e. TAP FSM is in *Run_Test/Idle* state) all of its child TAP controllers also become reset. Since the highest-level controller is the component-level TAP, reset of the TAP causes all embedded controllers also to become reset and allows IEEE 1149.1 compliant behavior of the component until a specific instruction is executed to access the next-level embedded test controller. This operation is compliant with the IEEE 1149.1 at the top-level of the IC component while allowing different behavior at the lower levels of the hierarchy. The H-TAP can be easily linked to the P1500 SIL (or a series of SILs) by decoding the outputs from its internal FSM and implementing a special instruction that enables the SIL. This is illustrated in figure 2.

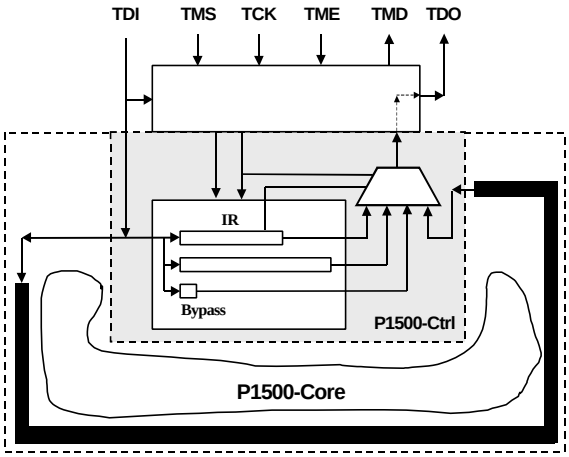

Figure 2. Architecture of Hierarchical TAP

6. The HIERARCHICAL SIL (H-SIL)

The present thinking within the P1500 CTAG Working Group is that all embedded cores within the target SoC can (should) be connected as a single chain regardless of their hierarchical nature. Furthermore, due to the similarities of between the IEEE 1149.1 TAP controller and the P1500 SIL, it is thought that cores that contain TAP controllers can be connected in series with P1500 compliant cores. This requires that an external (e.g. chip-level) controller should drive the WIP signals in the correct manner to allow such co-operation. Whereas this is achievable, it leads to the unintended conclusion that *the entire serial chain must operate following the IEEE 1149.1 protocol if even a single TAP controller exists along its path*. This fact defeats the purpose in removing the 16-state TAP FSM in the P1500 SIL since the SIL must still be controlled such that it behaves as if the 16-state FSM is present. This problem can be avoided if the chip is broken into a number of separately operable serial paths such that test access controllers that implement different access protocols are placed along different paths. For example, test access controllers (e.g. P1500 SIL, IEEE 1149.1 TAP, VSIA TCB, etc) may be connected such that they are accessed at different levels of hierarchy. This way, when a different level in the hierarchy is enabled to access the test controller at that level, other controllers would become transparent and would not affect the access protocol for the enabled test controller.

In order to support hierarchy, first it is necessary for a core's SIL to be modified so it has an operational mode where only registers from the present SIL can be inserted between the externally visible WSI and WSO terminals. That is, whereas connecting an upper-level core's SIL in series with the SIL of its child(ren) should be permitted, this should be done in such a way that, in a basic operational mode, just the upper-level core's SIL registers (such as WBYPASS Register) should be visible between the WSI and WSO terminals. Second, serially connected SILs should be operated such that when the WIR in any SIL is enabled for shifting this shall cause all WIRs in all SILs also to become enabled. This means that loading a new instruction into the WIR for the top-level SIL shall require instruction registers of all other

serially connected SILs also to be updated with a new (or the same as the present) instruction opcode. This raises the question of what instruction opcode should be loaded into all of the child SILs when the intent is only to load the top-level SIL. One answer to this question is to define and load the *Normal* opcode into the child SILs where the *Normal* instruction places the core into its mission mode and may shut down all of its test circuitry. Alternately, it is possible to load the *WBypass* instruction which inserts the WBypass register between the core's WSI and WSO terminals and assures that the test circuits do not interfere with the mission function of the core. However, the *WBypass* instruction does not guarantee that the mission logic can be executed; that decision/capability is based on actual design implementation. Defining the *Normal* instruction as a new, mandatory instruction for the P1500 architecture solves several initialization, reset and test issues and should be considered for adoption by the P1500 WG. The *Normal* instruction also makes it possible to generate a reset condition to reset test logic inside the entire hierarchy of SILs below the present-level SIL. The WBypass instruction can still be used if the top-level SIL is to be enabled for mission mode operation without causing a reset of the lower level SILs.

Next, we should consider implementing different sets of instructions that select just the present (e.g. top-level) SILs own registers or the registers of the present SIL concatenated with registers from other SILs that are connected in series with it. For example, consider the following instructions:

Instruction	Selected Register	Opcode
Wpreload	WBR and all serial Cores	001
Wclamp	WBYPASS and all serial Cores	010
Safestate	WBYPASS and all serial Cores	011
Wextest	WBR and all serial Cores	100
Coretest	WBR and all serial Cores	101
WBypass	WBypass and all serial Cores	110
Cnormal	Straight through present Core only	000
Cbypass	WBYPASS of present Core only	111

Table 1. Sample P1500 instruction set to support hierarchy

Even greater flexibility can be provided by assigning a unique bit within the WIR to enable just the present-level or the present-level plus the next level of SILs that are connected in series. This allows accessing the lower-level SILs while the present-level SIL remains enabled using any one of its possible instructions. This way hierarchical cores can be accessed so that SILs at either or both levels of the hierarchy can be selected for operation.

7. CHIP-LEVEL ACCESS to the P1500 SIL

The same strategy described above can be used to modify an existing TAP controller so that either the TAP's own registers or those registers plus the next-level P1500 SIL registers are accessed. However, this requires modifying (i.e. adding new instructions to) existing TAP design. An alternative approach may be to connect the TAP and the next-level SILs in series and to implement the lower-level SILs such that when in the *Normal* mode each SIL introduces a 0-length (i.e. no flip-flops) bypass path between its WSI and WSO terminals. This allows programming the TAP with the usual 1149.1 instructions while loading the *Normal* mode into the SILs in order to achieve IEEE 1149.1 compliant behavior for the IC component. Access to internal cores would be provided by loading an instruction into the SIL while the TAP IR is loaded with the BYPASS instruction or any other instruction that may be desirable.

If the above approach is used with multiple SILs in 0-length bypass mode then the path-delay through them may become too long and/or difficult to characterize. This issue can be mitigated with the following hierarchy of test controllers.

Level 0 (IC Component level) ……TAP Controller

Level 1 (Next-level in hierarchy) ……SIL_1

Level 2 and up ……SIL_2, and up

Here, SIL_1 is a dummy SIL that does not belong to any internal core of the IC. Its only purpose is to act as the interface between the TAP and the other cores. Level_2 consists of any number of SILs that are at the same level of hierarchy and are connected in series. This is illustrated in Figure 3.

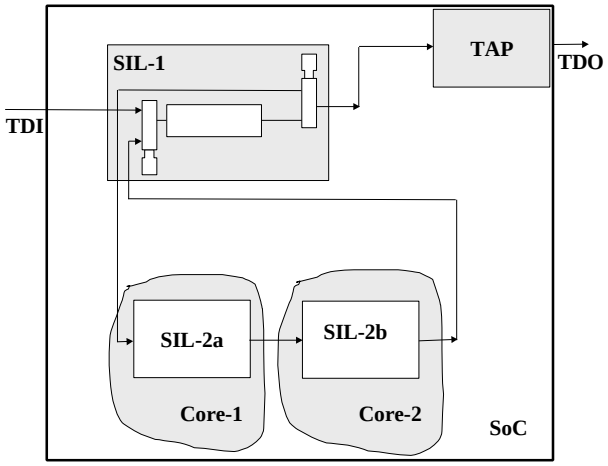

Figure 3. Connecting multiple SILs in a hierarchy

8. CONCLUSIONS

The VSIA TST DWG and the IEEE P1500 CTAG have developed initial proposals for test access in cores (i.e. VCs). The two proposals are very similar but they also possess some notable differences. In general, both solutions have adopted some basic operating principles that have their roots in the IEEE 1149.1 standard. However, the issue of exactly how to interface the VSIA TCB and/or the P1500 SIL with a chip-level IEEE 1149.1 TAP Controller has not been discussed in detail by either group. The approach taken by the VSIA TST DWG has been to define test mode requirements without codifying these as formal. By doing so, the VSIA TST DWG has left it to the IEEE P1500 WG to develop a standard that meets the VSIA requirements. The P1500 WG is developing architectural elements and formal rules that define their characteristics.

One solution that had been developed earlier through the work done by the IEEE P1149.2 WG [6] to handle similar situations can be reused here. Indeed, it has been recommended that both the VSIA and P1500 should consider including the 1149.1 TAP Controller as a baseline and allow bypassing the TAP FSM in a particular communications mode. This would allow a common, serial interface (plus existing software) to link all cores and still allow the more specialized/flexible test/communications access

offered by the present VSIA and P1500 solutions. It is unfortunate that such proposals have been rejected by the P1500 WG with religion-like opposition that has not been based on sound technical rationale.

One of the notable differences between the VSIA and P1500 solutions is that the TCB block in the VSIA proposal is optional whereas the WIR is a mandatory feature of the P1500 SIL. Thus, the VSIA DWG has focused on defining a set of common test control inputs to the core logic and define their actions. The intent of this approach was to (a) define a lightweight set of requirements for DFT, and (b) to allow sharing of a common TCB among multiple VCs. In contrast, the P1500 solution defines a mandatory Instruction Register in the SIL and also defines the set of mandatory and optional instructions without specifying the test control signals that are driven by the SIL in order to place the core into the various test modes. Obviously, it is necessary to reconcile the different choice of mandatory architectural elements (e.g. the Instruction Register) between the two approaches so that a common approach can be recommended to the VSI members.

A second area of difference between the two solutions is that the VSIA solution uses separate serial-in/serial-out ports for the Wrapper (WP_SI/WP_SO) from the serial-in/serial-out ports for the TCB (TC_SI/TC_SO). This allows connecting the VSIA TCB directly in series with a core or chip-level TAP controller without creating protocol violations of the IEEE 1149.1 standard. Furthermore, P1500 needs to solidify its position regarding edge-triggered operation of the SIL under the control of a single-phase test clock. It is not easy to assure interoperability among multiple SILs that may not have been developed using simple, straightforward timing model.

Finally, both of the VSIA and the P1500 approaches need to be improved to provide capability to handle hierarchy within the cores (VCs) as well as between the top-level IC component and the next-level cores under it. Here, P1500 CTAG has taken a strong position that can be paraphrased as:

P1500 SIL addresses hierarchy by offering a serial connection among the SILs at different levels of the hierarchy and leaves it up to the system integrator to develop the access methodology from the chip-level TAP Controller to the individual P1500 SILs.

This is an undesirable solution (or non-solution!) to a real problem. Simple extensions to the P1500 SIL architecture can be implemented to address these problems, including:

- New *CNormal* instruction which disables test circuitry and places the core into its mission mode.
- New *CBypasss* instruction which places only the present core's *WBypass* register between the core's *externally visible* WSI and WSO terminals. Interpret the *WBypass* instruction to place the WBYPASS register of the present core in series with presently selected registers in all other SILs that have been connected in series with it.
- Provide a new *WRClock_Enable* input to enable placing a core into a *Wpause* state.
  Note: This simple and innocuous feature has been presented to the P1500 WG and was shown to be a key feature necessary for implementing hierarchical P1500 SIL [10]. Unfortunately, this proposal was turned down by the P1500 CTAG WG primarily due to a lack of interest among its members in developing a hierarchical SIL architecture.

It is further recommended that the above features should be implemented using a dedicated bit in the WIR to select between the present SIL alone or all serially connected SILs for access between WSI and WSO. This allows accessing a child SIL while the parent SIL may remain preloaded with an arbitrary instructions (i.e. not just the *WBypass* instruction). These features allow implementing a chip-level hierarchical structure whereby an interface SIL can be used as a stepping stone to link the top-level TAP Controller with other SILs that have been connected in series to form a composite SIL.

Both of the VSIA test DWG and the IEEE P1500 CTAG aim at addressing test access requirements for individual cores. More technical work and improved coordination efforts are needed to bring the VSIA TST DWG and IEEE P1500 CTAG solutions together so that a single standard emerges to address user requirements. Furthermore, it is necessary to structure a new effort that focuses on SoC requirements to develop methodology that ensures easy integration and use of test structures implemented by the cores.

9. REFERENCES

[1] Anon., IEEE Standard Test Access Port and Boundary Scan Architecture, *IEEE Std. 1149.1a-1993.*

[2] B. I. Dervisoglu, Shared I/O-Cell Structures: A Framework for Extending the IEEE 1149.1 Boundary-Scan Standard, *Proc. ITC'98, pp. 980-989, 1998.*

[3] D. Bhavsar, An Architecture for Extending the IEEE Standard 1149.1 Test Access Port to System Backplanes, *Proc. ITC'91, pp. 768-776, 1991.*

[4] Anon. VSI Alliance Test Architecture, *V0.11, 2000.*

[5] E. Marinissen, Y. Zorian, R. Kapur, T. Taylor, L. Whetsel, *Towards a Standard for Embedded Core Test: An Example,* Conf. Proceedings, 24.1, pp. 616-627, ITC'99.

[6] B. Dervisoglu, *Status and Summary of IEEE P1149.2,* Design & Test of Computers, pp. 79 – 81, Sep. 1992.

[7] B. I. Dervisoglu, M. Ricchetti and B. Eklow, *Shared I/O-Cell Structures: A Framework for Extending the IEEE 1149.1 Boundary-Scan Standard,* ITC'98, Session 38.3, October 18-22, 1998, Washington, D.C.

[8] Anon., *Supplement to IEEE Std 1149.1-1990 IEEE Standard Test Access Port and Boundary-Scan Architecture, IEEE Std. 1149.1b-1994.*

[9] B. Dervisoglu, J. Swamy, *A Novel Approach for Designing a Hierarchical Test Access Controller for Embedded Core Designs in an SoC Environment,* 4th International Workshop of Testing Embedded Core-based System-Chips, 1.4-1, May 3-4, Montreal, TECS'00.

[10] B. Dervisoglu, *Proposal to Handle Embedded Cores,* formal proposal to P1500 CTAG, members-only URL, http://grouper.ieee.org/groups/1500/ctag_private/00/12.05.dervisoglu.html.