

Embedded Instrumentation Integration Using IEEE Nexus 5001 and 1149.7

Neal Stollon, HDL Dynamics

New capabilities for improving embedded control and visibility for chip level analysis and design for debug logic and interfaces to an IC are needed to enable real time functional control, test, and observation of embedded and otherwise not easily accessible operational features [1]. IEEE 5001, also known as Nexus provides a standard method and architecture for embedded instrument interfaces. New debug and instrumentation capabilities, being introduced by Nexus working groups, include support for IEEE 1149.7, which defines the next generation for (JTAG) Debug and Test Control [2].

Given a need for embedded instrumentation, the advantages of adopting standards based solutions are compelling. In addition to reducing the design times required for the instrumentation interface, standards based solutions provide consistency of features and have support from a variety of tools vendors. As an example, over half dozen tools vendors support a range of Nexus solutions today. Instrumentation based debug tools have evolved from the earliest days of processor application to provide powerful and compact methods of hardware and software analysis. The most widely used instrumentation interface today is IEEE 1149.1 or JTAG, which provides a test interface in the majority of digital ICs. JTAG provides a simple and well understood interface supported by many embedded systems tools vendors. JTAG, as IC and board level test interface, has limitations when used as an instrumentation interface. Since JTAG has a single bidirectional serial channel (**figure 1**), bandwidth is limited; which for communication intensive instrumentation applications such as tracing of embedded signals, can be a serious limitation. Similarly the serial nature of JTAG limits the number of JTAG based instruments that it can practically communicate with, which is a performance limitation barrier for multi-core designs at the SoC or board level. The simplicity of the JTAG interface similarly presents a significant performance limitation in many instrumentation applications. JTAG, as a relatively simple state controlled interface, lacks any native features for security, power management, and other factors important to modern SoC. Despite these limitations, JTAG has remained the leading instrumentation interface, in part due to a lack of alternatives that also support more mainstream test functions.

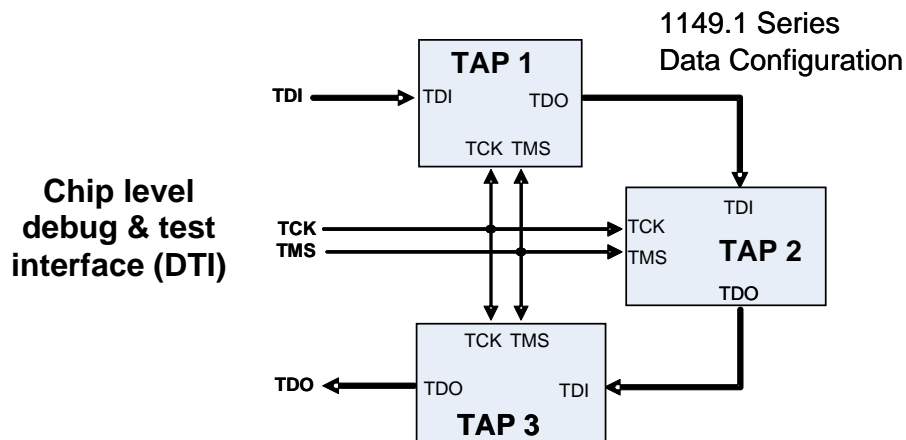


Figure 1: 1149.1 JTAG Interface

Nexus functions

Nexus is an alternative instrumentation interface developed to address these limitations of JTAG for instrumentation. Nexus was developed (and standardized as IEEE 5001) in 1999 as an

instrumentation and processor debug architecture that includes IO ports for improved bandwidth and a standardized protocol that supports a variety of instrument types and both inter and intra-chip multi-core integration and communication. [3,5]. A basic Nexus interface, as shown in **figure 2**, includes 2 sets of interfaces; a JTAG interface and parallel input and output data interfaces, referred to as AUX ports., Nexus interfaces can be configured in 3 modes:

- a). JTAG by itself for both control and data transfer
- b) JTAG (mainly for control and lower speed and bandwidth operations) plus AUX interfaces (mainly for higher bandwidth data transfer)
- c) AUX ports by themselves for both control and data transfer.

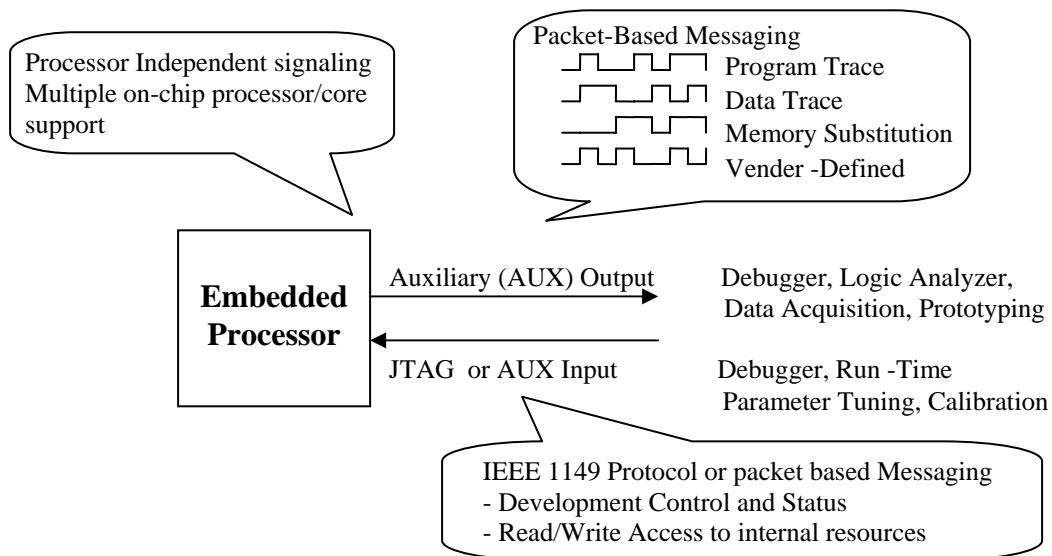


Figure 2: Nexus IEEE 5001 Interfaces

Nexus relies on the AUX ports for higher speed and bandwidth interfaces than can be supported by JTAG. The tradeoff of this is that Nexus requires dedicated pins to support transfer during normal operations and pins are a rationed resource in increasingly pin limited devices such as most complex SoC. Nexus addresses operation with limited pins by only implementing AUX ports where they are required. So, as a typical example, for trace or other write IO intensive operations, the output bandwidth required is addressed by an AUX output port, whereas the lower bandwidth setup and control inputs to the trace instruments can be supported by JTAG. Alternately, operations such as calibration or memory substitution (a Nexus configuration of replacing accesses to on-chip memory with accesses to external memory transferred through the Nexus interface), are typically more read IO intensive and may require input AUX ports to meet bandwidth requirements.

1149.7 functions

The addition of Nexus interfaces, improves the instrument interface bandwidth by using the AUX ports and a higher performance instrumentation interface protocol. The 1149.1 JTAG interfaces, which were not designed with instrumentation in mind, can still introduce limitations. An updated and expanded interface based on JTAG has recently been introduced which addresses many of the limitations of 1147.1 JTAG. IEEE 1149.7 interfaces are backward compatible with 1149.1 JTAG, but support an architecture that provides a more compact and powerful means for adding

embedded instruments to digital processors and SoC devices. Support for 1149.7 is included in the IEEE 5001-2009 release, planned for later this year.

IEEE 1149.7 expands upon key features of 1149.1 JTAG in several important ways. 1149.7 features are provided in a progressive series of 6 Access Port classes (T0-T5 as shown in **figure 3**). Support for any class of capabilities implies the support for the features at the previous class. It should be noted that Nexus is compatible with all class levels, and that, having a packet based protocol where data packet information is stored or accessed from Nexus defined registers [4], is operating at a different level of hierarchy compared to the more physical layer oriented 1149.7 interfaces. As such, different Nexus implementations may adopt 1149.7 features up to any class level, without being required to support the entire set of features.

The costs of IEEE 1149.7 are in adding additional levels of control hierarchy to a test and debug port. While IEEE 1149.1 JTAG was designed for reduced logic in a time when gates count per device were smaller and more expensive, in many current ICs, additional logic and complexity are justified for increased features and a more flexible and reduced pin interface;

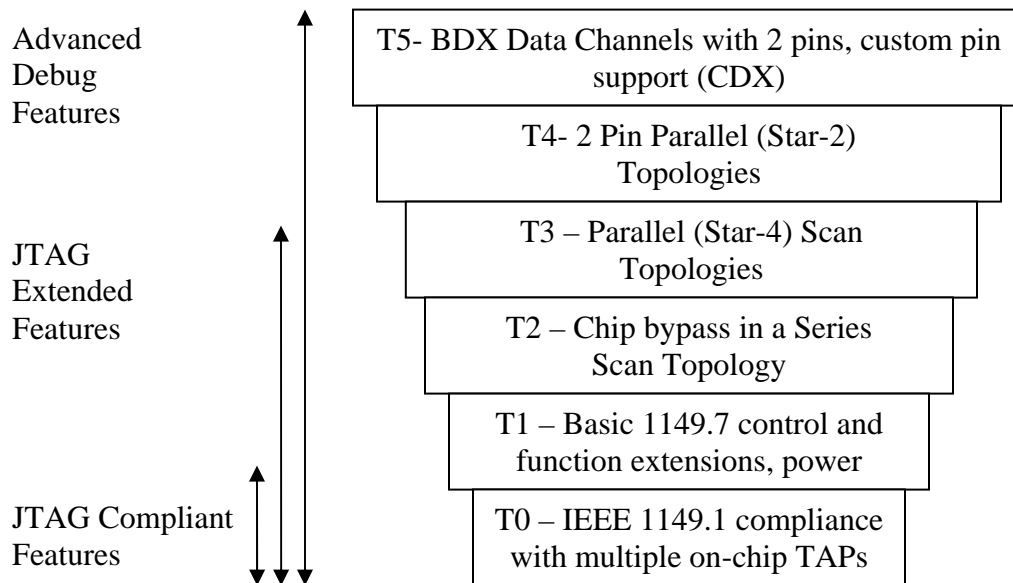


Figure 3: IEEE 1149.7 Classes

The most basic class T0 capability provides a backward compatible 1149.1 emulation interface. This allows legacy compatibility with existing JTAG test systems, or where more advanced features are not required.

Class T1 capability provides 4 user selectable power control modes (based on power down states and power down after delay) Whereas 1149.1 JTAG has a single "always on" state, IEEE 1149.7 selectable power mode controls may be propagated into Nexus logic to enable standardized means to power down test and debug features for low power devices

Class T2 capability provides coupling mechanisms for reduced bypass delay for the case of multiple TAPs on a chip or a system. This allows shorter latency for serial scan operations, which can be significant for devices with a larger number of TAPs and instruments.

Class T3 capability provides for parallel data input (TDI) and output (TDO) configurations in addition to the JTAG serial data chain interfaces. 1149.7 refers to this parallel data interface as a Star-4 topology (**figure 4a**). This allows simpler and more direct access to on-chip TAPs and instruments in a manner to similar to many on-chip bus interfaces and to Nexus AUX ports.

Class T4 supports multiple test/debug configuration functions, the most notable of which is operation in a 2 wire TAP (Star-2) configuration (**figure 4b**) This allows operation with a reduced pin interface, a key requirement in pin limited systems such as mobile devices.

Class T5 supports BDX (Background Data Transport) and CDX (custom data transport) transfer modes for higher data channel performance. BDX allows transfers during idle control periods. CDX transfers are, as implied, user defined, and can include Nexus optional functions or extensions.

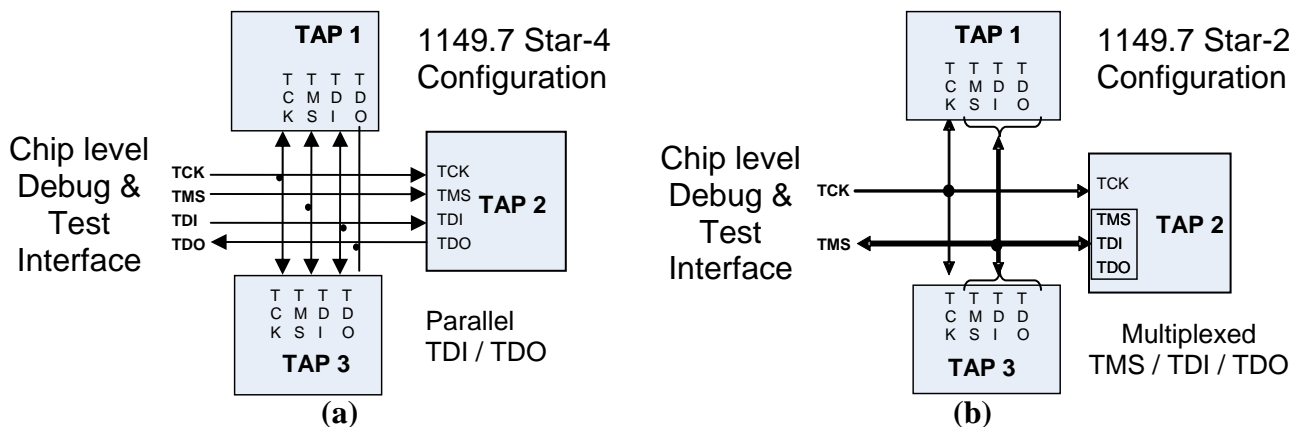


Figure 4: 1149.7 Star-4 and Star-2 configurations

IEEE 1149.7 / Nexus Integration

While IEEE 1149.7 provides new port level features and improvements in latency for transport of embedded instrument control and data compared to 1149.1 JTAG, it does not address the issues of improving bandwidth in the underlying communication with the instrument. As in JTAG, there is typically only a single data input and output interface. Since IO bandwidth is often the key limitation in performance of applications such as trace and calibration, this remains a serious limitation. Even at its most advanced (T5) level, IEEE 1149.7 interface only uses a single pin for instrument use. Having a scalable and extendable data interface for test and debug was one of the reasons for the creation of Nexus and one of its most notable features is the definition of input and output AUX ports that increase the effective number of data channels to significantly increase transport bandwidth.

Figure 5 shows the IEEE 1149.7 STAR-4 configuration integrated with both AUX IN and AUX OUT ports. Depending on requirements, only one of the AUX ports may be required. A similar configuration can be used to implement Nexus with a 2 wire Star-2 configuration. IEEE 5001 also defines, in the 2009 revision, the ability to implement increased (order of magnitude) bandwidth by replacing the AUX channels with a corresponding number of SerDes ports allowing Gigabit data transfers.

While other high performance debug interfaces exist (ARM ETM/Coresight and MIPS EJTAG being notable), they are typically vendor specific and closely tied to a given processor, making system level embedded instrument integration an ongoing challenge. Nexus provides a standard based integration scheme that supports heterogeneous core and debug interfaces neutrally. Nexus

packet based transfers include ID fields an specific transfer operations for accessing and assigning ID information that allow sequential and multiplexed access to multiple instruments within a single Nexus interface or across multiple Nexus instantiations.

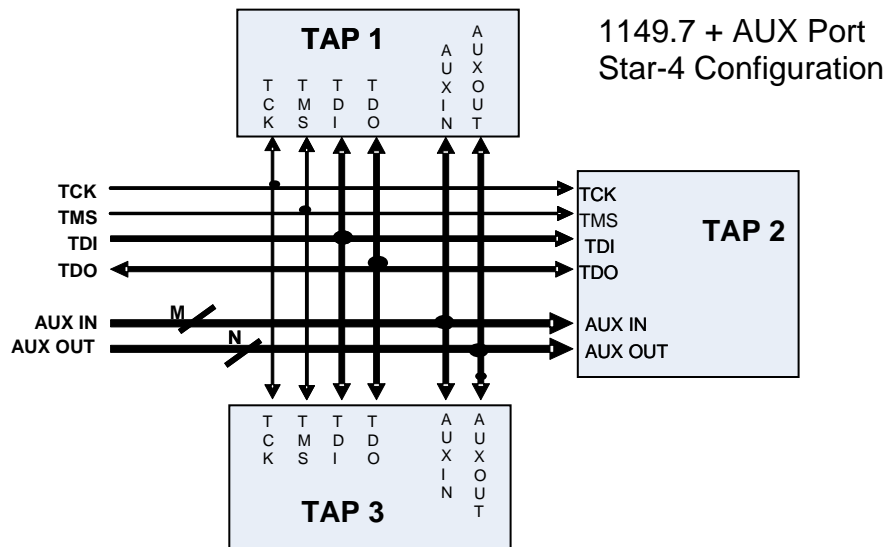


Figure 5: IEEE 1149.7 STAR-4 and Nexus AUX port interfaces

In a typical mode of operation, a Nexus interface may consist of a lower bandwidth JTAG interface providing command and configuration inputs and a higher bandwidth AUX OUT port outputting debug data. Since the JTAG chain may have greater latency than the AUX ports, this can in some cases introduce communication complexity and reduced performance. IEEE 1149.7, in particular in the T3 and T4 Star configurations, can reduce this latency. The AUX ports, being parallel data channels, can be configured in a related star output interface to synchronize operations and signaling.

In summary, Nexus has adopted IEEE 1149.7 as a successor to the 1149.1 JTAG interface port that is an interface feature in the IEEE 5001-2003 Nexus specification release. Since 1149.7 is backward compatible to 1149.1 JTAG, it does not impact any legacy systems that use JTAG as a debug interface. Its value for Nexus is in new features that make debug more efficient, including

- The ability to quickly access a specific TAP in a system with multiple TAPs, either on chip or in different devices. By implementing a system level bypass, the scan chain is drastically shorter, which directly improves debugging performance.
- The ability to control debug logic power consumption in an industry standard way
- The introduction of a Star-4 (4 wire JTAG parallel interface to multiple TAPs) connectivity to complement the 1149.1 JAG serial TAP connections. A star configuration allows simpler test connection and simplified physical connections that are compatible with Nexus data interfaces.
- A 2-wire TAP option (Star-2) that replaces the four-wire TAP, and which provides pin limited ICs, with size or package constraints, with test and debug features at a lower pin cost.

- [1] “Processor and System Bus On-Chip Instruments”, *Embedded Systems Conf (ESC) Proceedings*, April 2003
- [2] “IEEE 1149.7: Expanding and improving JTAG”, Nov. 2008
<http://www.embedded-computing.com/articles/id/?3687>
- [3] “Nexus Based Multi-Core Debug”, DesignCon 2006 Proceedings, Jan 2006
http://www.designcon.com/media/2006/3-WP1--Neal_Stollon.pdf
- [4] “Multi-core analysis made easy with the Nexus 5001 debug spec” March, 2008
<http://www.embedded.com/design/multicore/206901752>
- [5] Nexus 5001 Forum Website <http://www.nexus5001.org>