JTAG Interface : Simple Introduction

amt_ann004 (v1.0) Application Note

OVERVIEW

In today's complex systems, testability is an increasing concern in almost every application and in every area of application development. Manufacturers that thoroughly address the issue of testability at the device, board, and system levels deliver more consistently reliable and cost-effective products to the marketplace.

This means building in **test capabilities** in every phase of development and deployment, including design verification, hardware and software integration, manufacturing, and in the field.

INDEX

JTAG HISTORY

In the 1980s, the **Joint Test-Action Group (JTAG)** formed by representatives from makers and users of components and boards, recognized that only a cooperative effort could address the mounting testability problems in a coordinated way. Its mandate was to propose design structures that semiconductor makers would incorporate into device designs to aid in testing boards and systems. In 1990 the IEEE adopted the proposal as **IEEE Standard 1149.1**. Its stated purpose was to test interconnections between Integrated Circuits (ICs) installed on boards, modules, hybrids, and other substrates. Manufacturers adopting the standard could also test the IC itself.

JTAG ARCHITECTURE

Architecture IEEE Standard 1149.1 is a testing standard. However it is described as a collection of design rules applied principally at the IC level that allow software to alleviate the growing cost of designing and producing digital systems. The primary benefit of the standard is its ability to transform extremely difficult printed circuit board testing problems that could be attacked with ad-hoc testing methods into well-structured problems that software can easily and swiftly deal with. To conform to the boundary-scan standard IEEE 1149.1, a device must contain the following:

- **Test Access Port Controller (TAP),**
- **Scan Instruction Register,**
- **Scan Data Registers.**

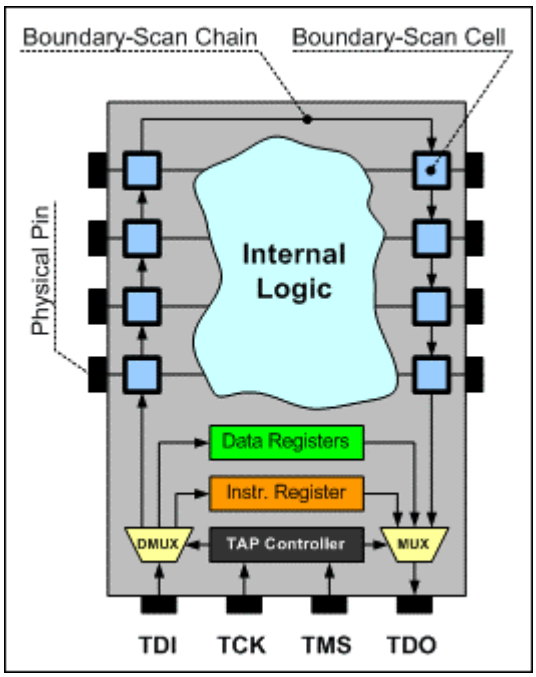

Figure 1: JTAG structure

JTAG TAP CONTROLLER

The TAP Controller is a **16-state finite state machine** added on the IC die itself. It recognizes the communication protocol and generates internal control signals used by the remainder of the Boundary Scan logic. **The TAP controller is driven by TCK and TMS only**; no other signals affect TAP controller. They program the TAP Controller as a 16-state machine, generating clock and control signals for the instruction and data registers. Only two events can trigger a change of controller state:

- a test-clock rising edge,
- system power-up.

Movement through the state machine is controlled by the value of TMS, a set-up time prior to the **rising edge of TCK**. The 1s and 0s adjacent to each state transition arc show the value that must be present on TMS at the time of the next rising edge of TCK.

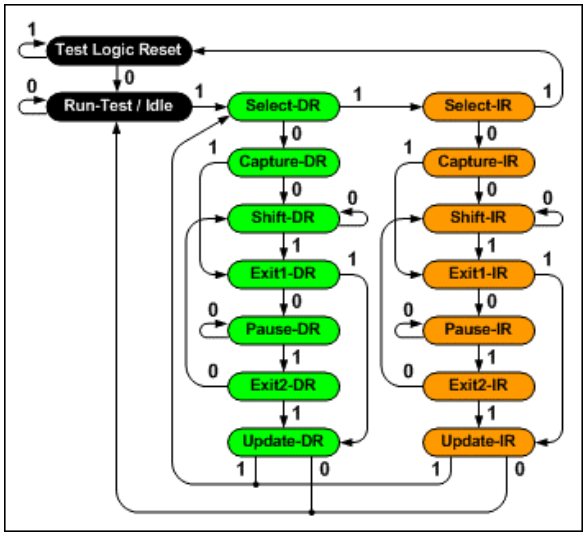

Figure 2: JTAG TAP Controller

The different state can be divided in 4 groups:

- Reset,
- BIST,
- Data Register Update,
- Instruction Register,
- Update.

JTAG TAP State denominations

IEEE JTAG Spec.	SVF Spec.	Amontec Code	
Exit2-DR	DREXIT2	"0000"	hex0
Exit1-DR	DREXIT1	"0001"	hex1
Shift-DR	DRSHIFT	"0010"	hex2
Pause-DR	DRPAUSE	"0011"	hex3
Select-IR	IRSELECT	"0100"	hex4
Update-DR	DRUPDATE	"0101"	hex5
Capture-DR	DRCAPTURE	"0110"	hex6
Select-DR	DRSELECT	"0111"	hex7
Exit2-IR	IREXIT2	"1000"	hex8
Exit1-IR	IREXIT1	"1001"	hex9
Shift-IR	IRSHIFT	"1010"	hexA
Pause-IR	IRPAUSE	"1011"	hexB
Run-Test / Idle	IDLE	"1100"	hexC
Update-IR	IRUPDATE	"1101"	hexD
Capture-DR	IRCAPTURE	"1110"	hexE
Test Logic Reset	RESET	"1111"	hexF

JTAG TAP State definitions

Test-Logic-Reset

For a target device in the stable Test-Logic-Reset state, the test logic is reset and is disabled so that the normal logic function of the device is performed. The instruction register is reset to an opcode that selects the optional IDCODE instruction, if supported, or the BYPASS instruction. Certain data registers also can be reset to their power-up values.

Run-Test/Idle

For a target device, Run-Test/Idle is a stable state in which the test logic can be actively running a test or can be idle.

Select-DR-Scan, Select-IR-Scan

For a target device, no specific function is performed in the Select-DR-Scan and Select-IR-Scan states, and the TAP controller exits either of these states on the next TCK cycle. These states allow the selection of either data-register scan or instruction-register scan.

Capture-DR

For a target device in the Capture-DR state, the selected data register can capture a data value as specified by the current instruction. Such capture operations occur on the rising edge of TCK, upon which the Capture-DR state is exited.

Shift-DR

For a target device, upon entry to the Shift-DR state, the selected data register is placed in the scan path between TDI and TDO, and on the first falling edge of TCK, TDO goes from the high-impedance state to an active state. TDO outputs the logic level present in the least-significant bit of the selected data register. While in the stable Shift-DR state,

data is shifted serially through the selected data register on each TCK cycle.

Exit1-DR, Exit2-DR

For a target device, the Exit1-DR and Exit2-DR states are temporary states that end a data-register scan. It is possible to return to the Shift-DR state from either Exit1-DR or Exit2-DR without recapturing the data register. On the first falling edge of TCK after entry to Exit1-DR, TDO goes from the active state to the high-impedance state.

Pause-DR

For a target device, no specific function is performed in the stable Pause-DR state. The Pause-DR state suspends and resumes data-register scan operations without loss of data.

Update-DR

For a target device, if the current instruction calls for the selected data register to be updated with current data, such update occurs on the falling edge of TCK, following entry to the Update-DR state.

Capture-IR

For a target device in the Capture-IR state, the instruction register captures its current status value. This capture operation occurs on the rising edge of TCK, upon which the Capture-IR state is exited.

Shift-IR

For a target device, upon entry to the Shift-IR state, the instruction register is placed in the scan path between TDI and TDO, and on the first falling edge of TCK, TDO goes from the high-impedance state to an active state. TDO outputs the logic level present in the least-significant bit of the instruction register. While in the stable Shift-IR state, instruction data is shifted serially through the instruction register on each TCK cycle.

Exit1-IR, Exit2-IR

For a target device, the Exit1-IR and Exit2-IR states are temporary states that end an instruction-register scan. It is possible to return to the Shift-IR state from either Exit1-IR or Exit2-IR without recapturing the instruction register. On the first falling edge of TCK after entry to Exit1-IR, TDO goes from the active state to the high-impedance state.

Pause-IR

For a target device, no specific function is performed in the stable Pause-IR state, in which the TAP controller can remain indefinitely. The Pause-IR state suspends and resumes instruction-register scan operations without loss of data.

Update-IR

For a target device, the current instruction is updated and takes effect on the falling edge of TCK, following entry to the Update-IR state.

All JTAG operations shift data into or out of JTAG instruction and data registers. The TAP Controller provides direct access to all of these registers. There are two classes of JTAG registers: the Instruction register (only one) and Data registers (many). Access to the Instruction Register is provided through the Shift-IR state, while access to the Data Register is provided through the Shift-DR state.

To shift data through these registers, the TAP Controller of the target device must be moved to the corresponding state. For example, to shift data into the Instruction Register, the TAP Controller must be moved to the Shift-IR state, and the data shifted in, LSB first.

IEEE STANDARD 1149.1 COMPLIANT DEVICE

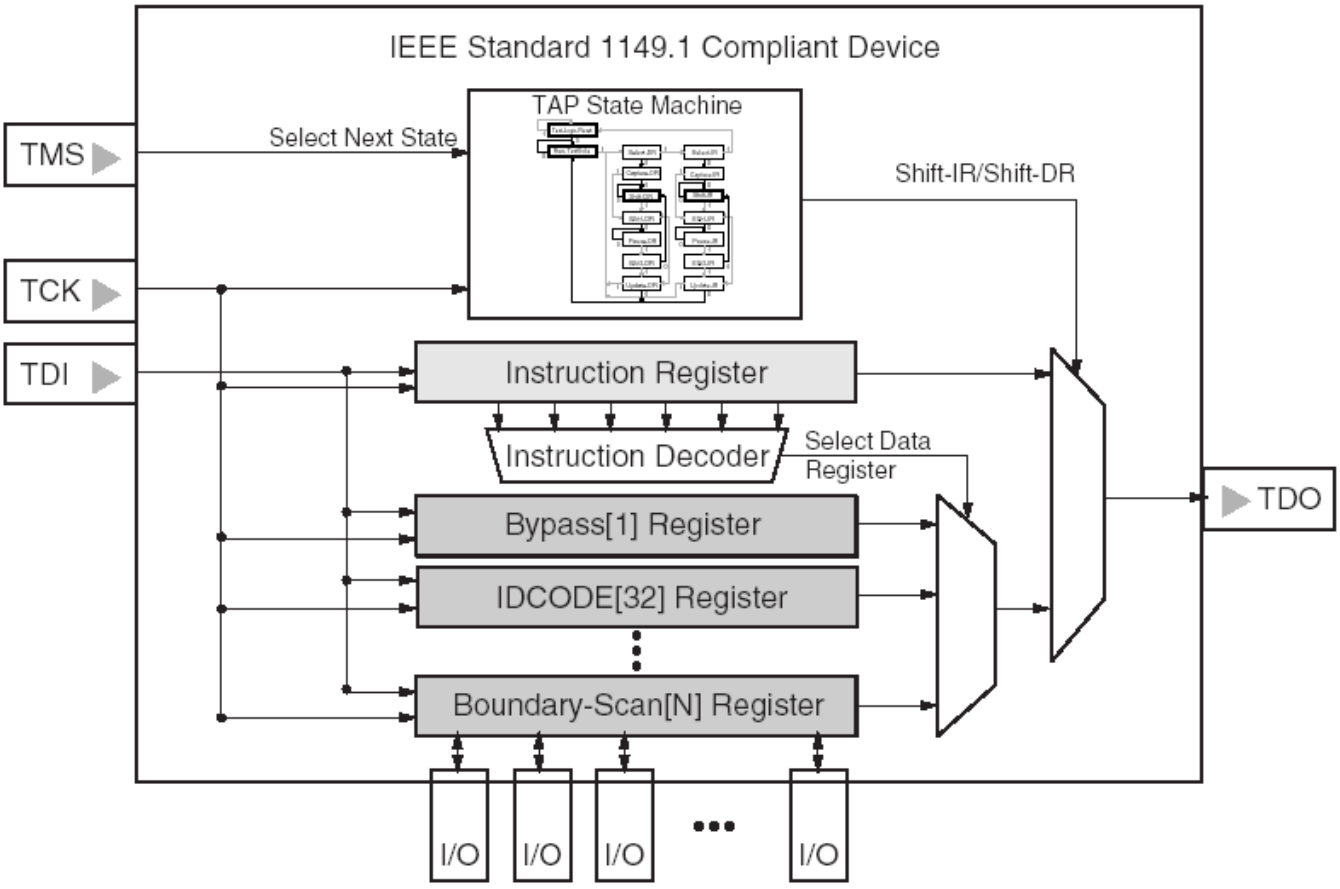

Figure 3: Typical JTAG Device Architecture

JTAG SIGNALS

The Boundary Scan Register and other test features of the device are accessed through a standard interface - the JTAG Test Access Port (TAP). According to the standard, the TAP must contain four signals, each available through a dedicated device pin and they may not be shared with any other function:

- Test Data Input (**TDI**): it is used to shift data and instruction tests into the Boundary Scan register.
- Test Data Output (**TDO**): this pin provides data from the Boundary Scan register or other internal register.
- Test Clock (**TCK**): this input controls test-logic timing independent of clocks that normal system operations employ. The TDI shifts values into the appropriate register on the rising edge of TCK. Selected register contents shift out onto TDO during the TCK's falling edge.
- Test-Mode select (**TMS**): this input, which also clocks through on the rising edge of TCK, determines the state of the TAP controller.

Note:

An optional active-low Test-Reset Pin (**TRST#**) permits an asynchronous TAP controller initialization without affecting other device or system logic. Asserting this pin inactivates the boundary-scan register and places the device in normal operating mode. These pins are used with a simple protocol to communicate with on-chip Boundary-Scan logic.

The key timing relationships include:

- TMS and TDI are sampled on the rising edge of TCK.
- A new TDO value appears after the falling edge of TCK.

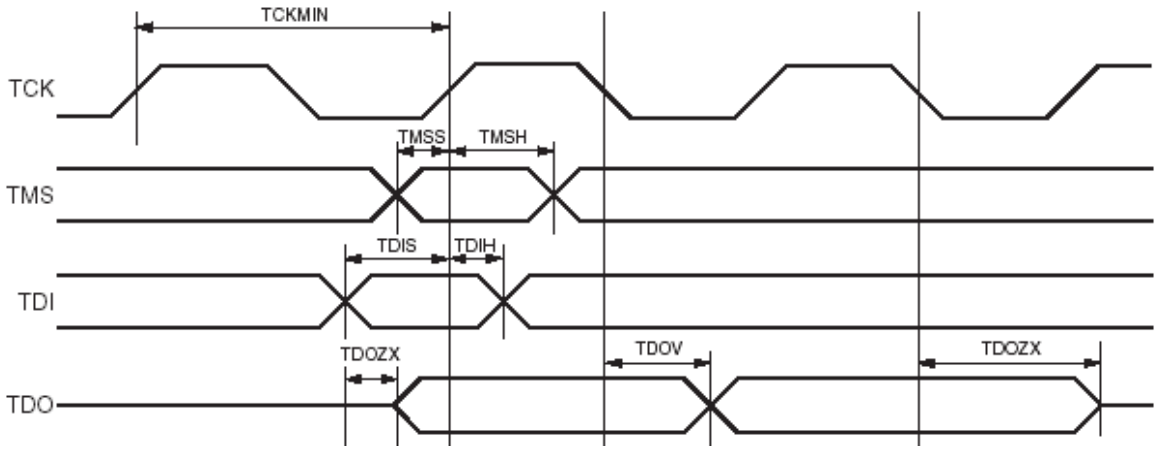

Figure 4: Test Access Port Timing

JTAG INSTRUCTION REGISTERS

The normal way of performing a JTAG test operation is to enter an instruction which specifies the type of test to be performed next, and the Data Register to be used during this test, into the Instruction Register (by means of running the TAP through an "ir path"), and then to use the Data Register to perform the test (by means of running the TAP through one or more "dr paths").

There are private and public instructions. Public instructions are documented by the chip manufacturers and available for general use. Private instructions are not. The IEEE-1149 standard defines a mandatory set of public instructions that must be present in all compliant JTAG implementations. This mandatory set contains the following instructions:

- **BYPASS**: Here the TDI and TDO lines are connected to single-bit pass-through register (which passes to TDI to the TDO with a single-clock delay). This instruction allows the testing of other devices connected to the same test-loop.
- **EXTEST**: With this command the boundary scan register (BSR) is connected between the TDI and the TDO signals. The chip's pin states are sampled and captured by the BSR cells at the entry to the "capture dr" state (see TAP state transition diagram above). The contents of the BSR register are shifted out via the TDO line at exits from the "shift dr" state. As the contents of the BSR (the captured data) are shifted out, new data are sifted in at the entries to the "shift dr" state. The new contents of the BSR are applied to the chip's pins during the "update dr" state.
- **IDCODE**: The ID register is connected between the TDI and the TDO. At the entry to the "capture dr" state are the Device ID Code (a hard-wired identification number containing the manufacturer code, that part number, and the revision code) is parallel-loaded into the register. This number is shifted out at the exits of the "shift dr" states.
- **INTEST**: With this command the boundary scan register (BSR) is connected between the TDI and the TDO signals. The chip's internal core-logic signals are sampled and captured by the BSR cells at the entry to the "capture dr" state (see TAP state transition diagram above). The contents of the BSR register are shifted out via the TDO line at exits from the "shift dr" state. As the contents of the BSR (the captured data) are shifted out, new data are sifted in at the entries to the "shift dr" state. The new contents of the BSR are applied to the chip's core-logic signals during the "update dr" state.

JTAG DATA REGISTERS

Several different data registers can be built into boundary-scan components. All Boundary-Scan instructions set operational modes that place a selected data register between TDI and TDO. This register is referred to as the target register. This preserves a fundamental notion of Boundary-Scan: TDI and TDO always form the ends of a shift register. The function of this register is dictated by the effective TAP instruction.

Two Data Registers are always required to be present on a 1149.1 component:

- the **Boundary-Scan Register**,
- the **Bypass Register**.

Several others are described by the standard, such as an Identification Register, but are optional.

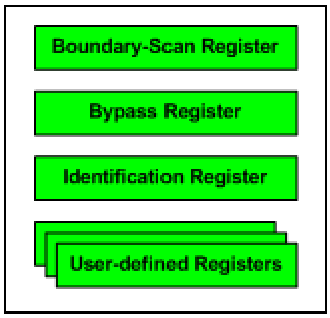

Figure 5: Data Register structure

The Test Data Register loads the data in parallel on the rising edge of TCK in the TAP Controller state CAPTURE-DR (triggering test results), puts the data on the output parallel latch on the falling edge of TCK in the TAP Controller state UPDATE-DR (new test pattern generation), and shifts the serial data through TDI to TDO during the TAP Controller state SHIFT-DR.

JTAG APPLICATIONS

The Boundary-Scan Standard allows for ICs to be linked into chains by linking the TDO pin of one IC with the TDI pin of the next. For example, the 1149.1 ICs on a board may all be linked together by their TDO-TDI pins in succession. Several distinct chains may exist on a board if they do not share any TAP signals.

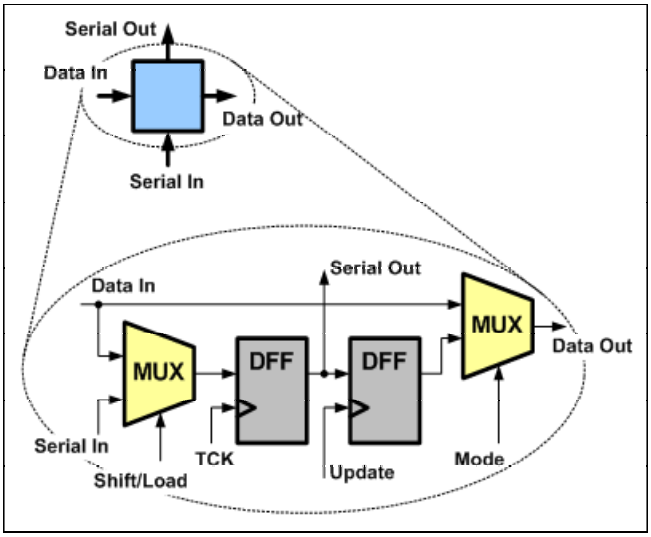

Figure 6: Boundary –Scan Cell

The 1149.1 Standard allows us to exploit the Boundary-Scan to test a board composed of Boundary-Scan chips.

Interconnect tests look for shorts between boundary-scan nodes and conventional nodes with bed-of-nails or edge-connector access, as well as opens between tester nails and boundary-scan input pins. The EXTEST instruction latches boundary-scan nodes to the state that permits easier in-circuit back drive (a logic 1 to TTL). The tester then looks for node movement when it forces non-boundary-scan nodes to their opposite states.

Applying this technique to a single conventional node places a 1 on that node and scans out boundary-scan-input states, then injects a 0 onto the test node and scans again. A short between the test node and a boundary-scan node will show up as a failure. An open connection will cause both scanning operations to produce exactly the same output pattern.

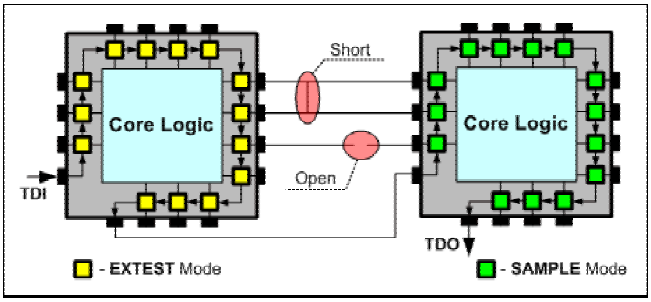

Figure 7: JTAG Board Application

Some shorts other than the one between test and boundary-scan nodes can cause this operation to fail. Only rarely, however, will such a faulty node follow the test node at both the latched-1 and latched-0 states. Changing test-node states several times and declaring a short only when the suspect boundary-scan node exactly follows these transitions further improves the likelihood of a correct diagnosis.

NEED MORE DETAIL ABOUT JTAG

Please feel you free to visit our Amontec website on http://www.amontec.com/.

FIGURE AND TABLE INDEX

CONTENT REVISIONS

This document contains the following changes to content, causing it to differ from previous versions:

Version	Date	Changes
V1.0	05-FEB-2005	First version (by Laurent Gauch)
V1.1	16-OCT-2005	Adding TAP State Denomination

CONTACT AMONTEC

Amontec, Gauch
Logic Road 123
1641 Vuippens
Switzerland
http://www.amontec.com/
mailto:info@amonte.com