XILINX ®

Using the XC9500/XL/XV JTAG Boundary Scan Interface

XAPP069 (v3.0) June 3, 2002

Summary

This application note explains the XC9500™/XL/XV Boundary Scan interface and demonstrates the software available for programming and testing XC9500/XL/XV CPLDs. An appendix summarizes the iMPACT software operations and provides an overview of the additional operations supported by XC9500/XL/XV CPLDs for in-system programming.

Introduction

IEEE Boundary Scan Standard 1149.1, also known as JTAG, is a testing standard that uses software to reduce costs. The primary benefit of the standard is its ability to transform difficult printed circuit board testing problems into well-structured, efficient solutions that are easily performed in software. The standard defines a hardware architecture and the mechanisms for its use.

The JTAG standard itself defines instructions that can be used to perform functional and interconnect tests as well as built-in self test procedures. Vendor-specific extensions to the standard allow execution of maintenance and diagnostic applications as well as permit programming algorithms for reconfigurable parts.

Connecting Devices in a Boundary Scan Chain

All devices in the chain share the TCK and TMS signals. The system TDI signal is connected to the TDI input of the first device in the Boundary Scan chain. The TDO signal from that first device is connected to the TDI input of the second device in the chain and so on. The last device in the chain has its TDO output connected to the system TDO pin. This configuration is illustrated in Figure 1.
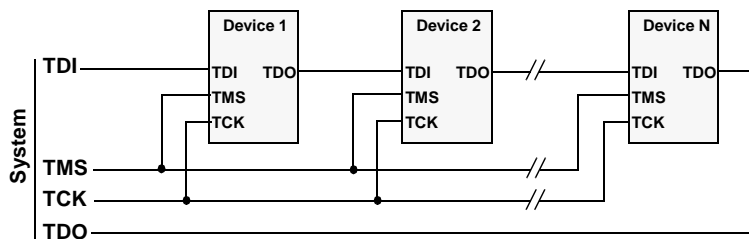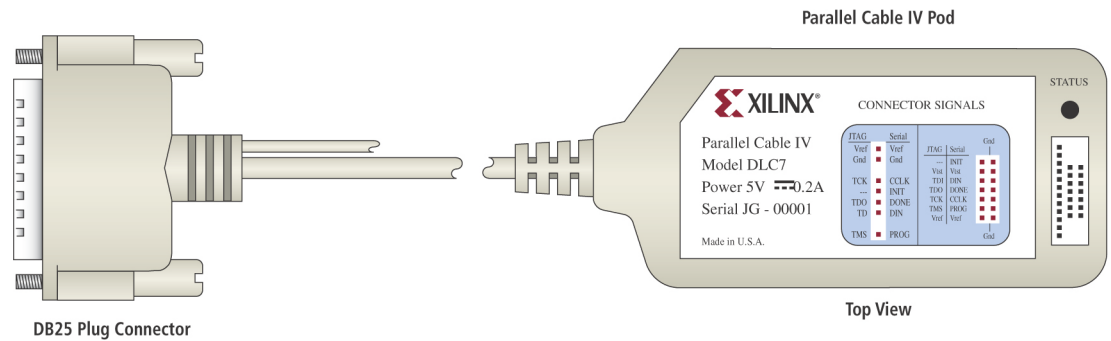

Figure 1: **Single-Port Serial Boundary Scan Chain**

Downloading a Design File

The Parallel Cable IV, shown in Figure 2, connects to the parallel printer port of any Windows-based PC. The cable contains drivers to buffer the signals as they are driven into the system, and the power for the drivers is derived from the target system. The cable's V_{REF} and GND wires are connected to power and ground on the target system, and the remaining four wires are connected to the corresponding TAP inputs on the target system. The cable pins are clearly labeled. The optional test reset (TRST) is not supported by the Xilinx cables and if any parts in the system have a TRST, this pin should be attached to V_{CC} through a pull-up resistor.

Note: Xilinx devices do not have the optional TRST pin.

www.xilinx.com
1-800-255-7778

Figure 2: **Parallel Cable IV**

Figure 3 shows how the cable is connected to the printed circuit board for programming. Connect all six flying leads to the target board and observe the power sequencing recommendations.
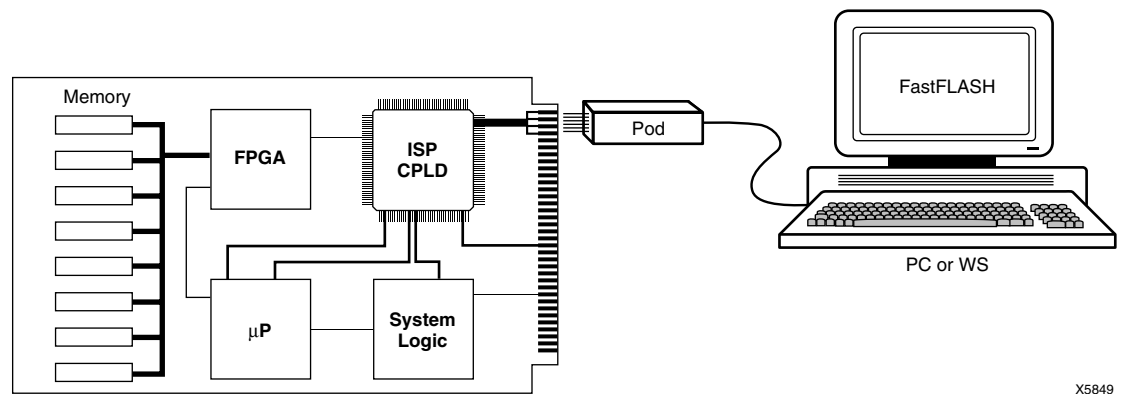
Figure 3: **Target PCB Connected for Program/Test**

Power Sequencing

Cable protection ensures that the parallel port cannot be damaged through normal cable operation. For increased safety, ensure that the PC is always powered up before the target system.

When powering down, turn off the target system first, and then turn off the host PC.

iMPACT Download Software

Upon initiation of the iMPACT download software, the parallel port is queried to verify the connection of the Parallel Cable IV. The cable must be attached and powered for proper verification. If an error message is returned, stating that the cable could not be found or indicating a cable other than the Parallel Cable IV was identified, check the cable connections to the PC parallel port. Figure 4 shows the iMPACT software user interface.

Using the iMPACT Download Software

See the *iMPACT User Guide* for detailed instructions. The following steps outline the downloading procedure:

1. Invoke the iMPACT software from the Project Navigator.

2. Assign the files for each device in the chain ordered from system TDI to TDO (use JEDEC files for XC9500/XL/XV devices or BSDL files for other JTAG-compatible devices).

3. Select the target XC9500/XL/XV devices to program.

4. Select the **Operation → Program** menu item with the Erase, before the programming and verify options. Detailed information regarding the downloading progress and any failure conditions will be displayed in the iMPACT log view.
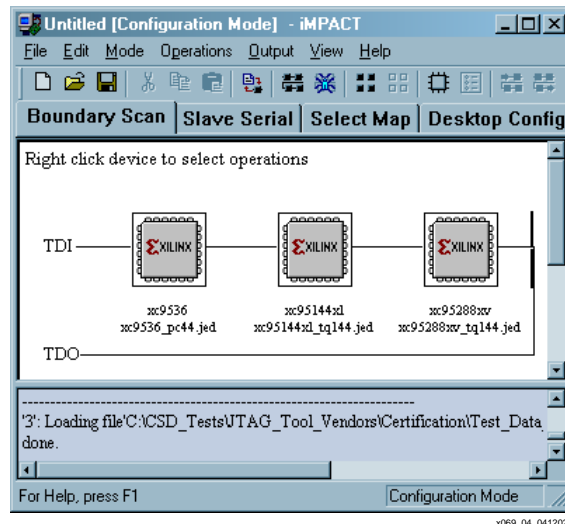

Figure 4: **iMPACT User Interface**

Debugging a Boundary Scan Chain

If you are unable to establish communications with the Boundary Scan chain, the DEBUG CHAIN dialog can assist in the debugging of this problem. To access this dialog select **File → Debug Chain** ... You will then see the dialog shown in Figure 5. You can control the values of the TMS, TDI, and the number of clock pulses as you step through the finite state machine of the test access port (TAP) controller. You will need to use an oscilloscope or logic probe to monitor TDO.
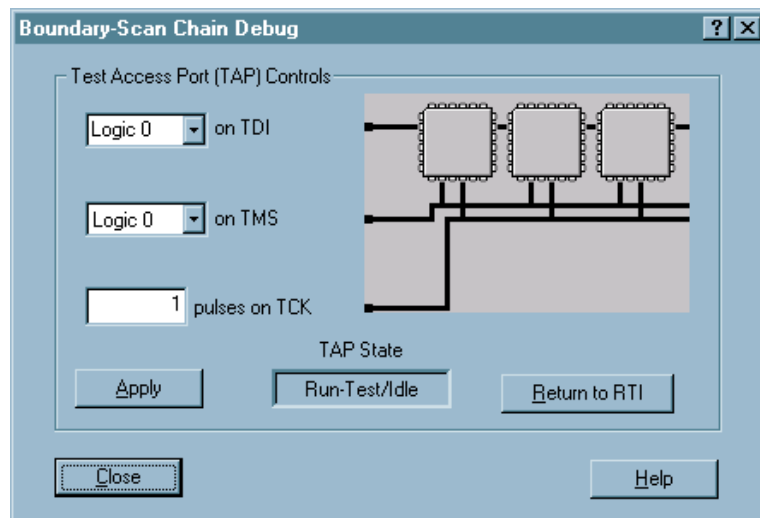
Figure 5: **Boundary Scan Chain Debug**

A good fundamental check of your Boundary Scan chain is to execute a simple integrity test. The boundary scan integrity check sequences the TAP through a TRST sequence (TMS set to 1, TCK pulsed 5 times) and then transitions all devices to the RunTest/Idle state (TMS set to 0, TCK pulsed once). Then, all parts are run through a CAPTURE -IR sequence while TDI is set to 1 (so that "1"s will be shifted in). To do this set TMS to 1 and pulse TCK twice to enter the Select-IR-Scan state. Then set TMS to 0 and pulse TCK twice to enter the Shift-IR state.

If you look in the device BSDL files you will see the expected capture sequence defined in the "INSTRUCTION_CAPTURE" field. This is the expected response shifted out at TDO in the Shift-IR state. For all XC9500/XL/XV parts, this sequence is a "1" followed by seven zeros. You should therefore see the "1" on TDO after the falling edge of the fourth TCK pulse after the TRST sequence. On the next TCK pulse TDO should return to zero. As you continue pulsing TCK in the Shift-IR state you will see the 1 to 0 transitions of the capture values. When all the capture values have been shifted out you will see the sequence of logic "1"s input on TDI.

See Appendix 1 for more details on the specific JTAG features supported.

Interfacing to Third-Party Boundary Scan Test Tools

BSDL files are required for interfacing to third-party Boundary Scan board test equipment (ATE), automatic test pattern generation software (ADAPT) and JTAG-based development and debugging systems. The BSDL files for all package variations of the available XC9500/XL/XV devices can be found in the Xilinx software "<device>data" directory, where <device> is either XC9500, XC9500XL, or XC9500XV. The BSDL files re also available on the web:

http://support.xilinx.com/support/sw_bsdl.htm

In some circumstances silicon is updated and revised.

Notes:
1.   The Xilinx iMPACT software is compatible with all version of the XC9500XL and XC9500XV families. However, the iMPACT software is compatible with only version 2 or later of the XC9500 CPLD family.

The Xilinx BSDL files follow a standard naming convention: `device_package.bsd`. For example, the BSDL file for the XC9536 in the PC44 package is `xc9536_pc44.bsd` and the xc95144 in the TQ144 package is `xc95144_tq144.bsd`.

Appendix A JTAG Details

The top level schematic of the test logic defined by IEEE Std. 1149.1 includes several key blocks as shown in Figure 6.
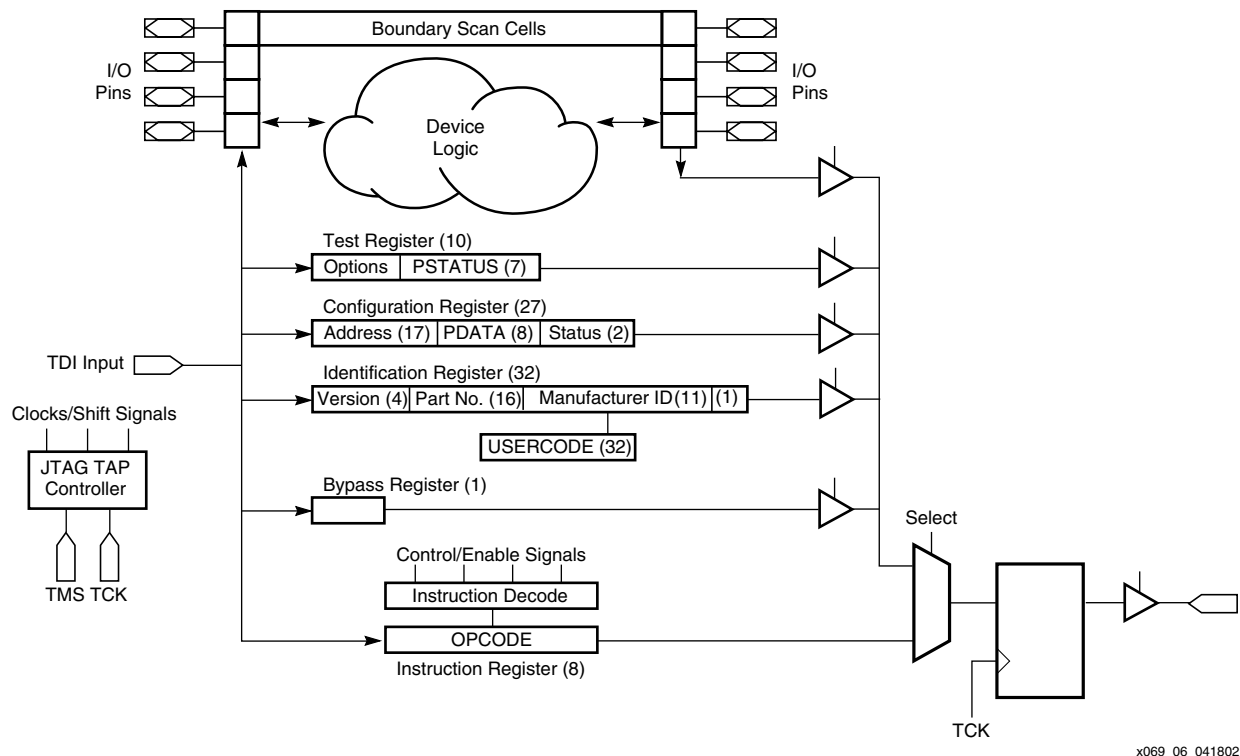

Figure 6: **JTAG Architecture**

The TAP Controller

The TAP controller responds to control sequences supplied through the test access port (TAP) and generates the clocks and control signals required by the other circuit blocks.

The Instruction Register

The instruction register is a shift register-based circuit and is serially loaded with instructions that select an operation to be performed.

The Data Registers

The data registers are a bank of shift registers. The stimuli required by an operation are serially loaded into the data registers, selected by the current instruction. Following execution of the operation, results can be shifted out for examination.

The JTAG Test Access Port

The JTAG Test Access Port (TAP) has four pins that drive the circuit blocks and control specific operations. The TAP loads and unloads instructions and data. The four TAP pins are: TMS, TCK, TDI and TDO. The function of each TAP pin is:

*   **TMS** - Test Mode Select is the mode input signal to the TAP Controller. The TAP controller is a 16-state finite state machine (FSM) that controls the JTAG engine. At the rising edge of TCK, TMS determines the TAP controller state sequence. TMS has an internal pull-up resistor to provide a logic 1 to the system if TMS is not driven.

*   **TCK** - JTAG Test Clock sequences the TAP controller as well as all JTAG registers.

*   **TDI** -Test Data Input is the serial data input to all JTAG instruction and data registers. The TAP controller state and instruction register contents determine which register is fed by TDI for any operation. TDI has an internal pull-up resistor to provide a logic 1 to the system if TDI is not driven. TDI is loaded into the JTAG registers on TCK's rising edge.

*   **TDO** - Test Data Out is the serial data output for all JTAG instruction and data registers. The TAP controller state and instruction register contents determine which register feeds TDO for a specific operation. Only one register (instruction or data) is connected between TDI and TDO for any JTAG operation. TDO changes state on TCK's falling edge and is only active during the shifting of data through the device. TDO is in a 3-state condition at all other times.

*   **TRST** - [Optional] Asynchronous Test Reset for the JTAG TAP. The TRST pin is active-low. When reset, the TAP state machine is put into the Test-Logic-Reset state, and the IDCODE instruction is loaded as the active instruction. Note that the BYPASS instruction is the default active instruction if the device does not support an IDCODE instruction.

JTAG TAP Controller

The TAP Controller is a 16-state FSM, that controls the loading of data into the various JTAG registers. A state diagram of the TAP controller is shown in Figure 7.

The state of TMS at the rising edge of TCK determines the sequence of state transitions. There are basically two state transition paths for sampling the signal at TDI: one for shifting information to the instruction register and one for shifting data into the data register.

JTAG TAP Controller States

Test-Logic-Reset

This state is entered on device power-up when at least five TCK clocks occur with TMS held high. Entry into this state resets all JTAG logic so that it does not interfere with the normal component logic, and loads the IDCODE instruction into the instruction register.
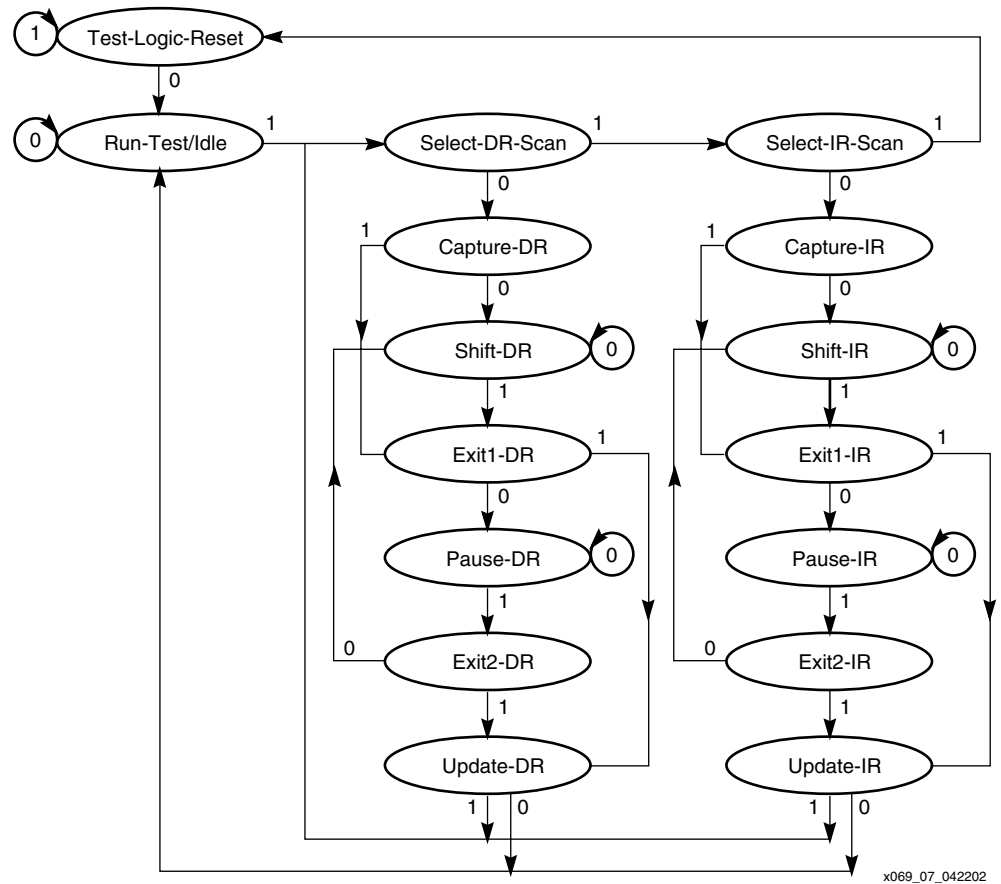
Figure 7: **TAP Controller State Diagram**

Run-Test-Idle

In this state, certain operations can occur depending on the current instruction. For the XC9500/XL/XV family, "Run-Test-Idle" causes generation of the program, verify, erase, and POR (Power-On-Reset) pulses when the associated ISP instruction is active.

Select-DR-Scan

This is a transitional state entered prior to performing a scan operation on a data register or in passing to the Select-IR-Scan state.

Select-IR-Scan

This is a transitional state entered prior to performing a scan operation on the instruction register or in returning to the Test-Logic-Reset state.

Capture-DR

This state allows data to be loaded from parallel inputs into the data register selected by the current instruction at the rising edge of TCK. If the selected data register has no parallel inputs, the register retains its state.

Shift-DR

In this state data is shifted by one stage in the currently selected register from TDI towards TDO by on each rising edge of TCK.

Exit1-DR

This is a transitional state allowing the option of passing to the Pause- DR state or transitioning directly to the Update-DR state.

Pause-DR

This is a wait state that allows shifting of data to be temporarily halted.

Exit2-DR

This is a transitional state allowing the option of passing to the Update-DR state or returning to the Shift-DR state to continue accepting data.

Update-DR

In this state the data contained in the currently selected data register is loaded into a latched parallel output (for registers that have such a latch) on the falling edge of TCK after entering this state. The parallel latch prevents changes at the parallel register output from occurring during the shifting process.

Capture-IR

In this state data is loaded from parallel inputs into the instruction register on the rising edge of TCK. The least two significant bits of the parallel inputs must have the value 01, and the remaining 6 bits are either hard-coded or used for monitoring the security and data protect bits.

Shift-IR

In this state instruction register values are shifted one stage towards TDO on each rising TCK edge.

Exit1-IR

Exit1-IR is a transitional state allowing the option of transitioning to the Pause-IR state or the Update-IR state.

Pause-IR

Pause-IR allows shifting of the instruction to be temporarily halted.

Exit2-IR

Exit2-IR is a transitional state allowing the option of passing to the Update-IR state or returning to the Shift-IR state to continue shifting in data.

Update-IR

In this state instruction register values are parallel latched on the falling edge of TCK. The parallel latch prevents changes at the parallel output of the instruction register from occurring during the shifting process.

JTAG Instructions Supported in XC9500/XL/XV Parts

Mandatory Boundary Scan Instructions

BYPASS

The BYPASS instruction configures the device to bypass the scan registers and pass immediately to TDO.

SAMPLE/PRELOAD

The SAMPLE/PRELOAD instruction allows a snapshot of the normal operation of a component to be taken and examined. It also allows data values to be loaded onto the latched parallel outputs of the boundary scan shift register prior to the selection of other Boundary Scan test instructions.

EXTEST

The EXTEST instruction allows testing of off-chip circuitry and board level interconnections.

XC9500/XL/XV Additional Boundary Scan Instructions

INTEST

The INTEST instruction allows testing of the on-chip system logic while the component is already on the board.

HIGHZ

HIGHZ permits automatic placement of all outputs on the XC9500/XL/XV part to high impedance (3-state) mode. This condition can be beneficial for board testing strategies.

IDCODE

The IDCODE instruction allows blind interrogation of the components assembled onto a printed circuit board to determine what components exist in a system.

USERCODE

The USERCODE instruction allows a user-programmable identification code to be shifted out for examination. This allows the programmed function of the component to be determined.

XC9500/XL/XV Reconfiguration Instructions

ISPEN

ISPEN activates the XC9500/XL/XV device for In-System Programming.

FPGM

FPGM programs bits at specified addresses.

FERASE

FERASE erases a sectors of programming cells.

FVFY

FVFY verifies the programming at specified addresses.

ISPEX

ISPEX transfers the XC9500/XL/XV memory cell contents to internal low power configuration latches.

FPGMI

FPGMI programs bits at an internally generated address.

FVFYI

FVFYI verifies the programming at an internally generated address.

FBULK

FBULK erases several sectors of programming cells.

Device Operations

The programming information is extracted from the JEDEC file generated by the fitter software. The JEDEC file name is defaulted to *design_name*.jed.

Device operation options available to users are:

Erase, Program, Verify

Download contents of the JEDEC file to the device programming registers. Erase the device, configure it, and read back the contents of device programming registers and compare them to the JEDEC file. Report any differences to the user.

Verify

Read back contents of the device programming registers and compare them with the JEDEC file.

Erase

Clear the device configuration information.

Functional Test

Apply user-specified functional vectors from the JEDEC file to the device using the JTAG INTEST instruction, and compare the results obtained with expected values. Report any differences.

Read Device ID

Read and display the contents of the JTAG IDCODE register.

Read User Signature

The signature value is set by the user at programming time. It is valid only after programming. This function reads the contents of the JTAG USERCODE register and displays the result.

Readback

Extracts contents of device programming registers and creates a new JEDEC file with the results.

Checksum

Extract the contents of device programming registers and calculate a checksum for comparison with the expected value

BSDL Description Summary

The Boundary Scan Description Language (BSDL) describes the boundary scan features of a component. The system looks for BSDL files along the Xilinx\data path and in the current working directory. A BSDL file must be specified for each non-XC9500/XL/XV device in the JTAG chain.

The name of the BSDL file is assumed to be *device_name*.bsd.

JEDEC Description Summary

The JEDEC file is an ASCII file containing the configuration information and optionally the vectors that can be used to verify the functional behavior of the configured part. A JEDEC file must be specified for each XC9500/XL/XV device in the JTAG chain; one JEDEC file is generated for each XC9500/XL/XV device in the system by the fitter software.

The name of the JEDEC file is assumed to be *design_name*.jed.

References

1. IEEE Std. 1149.1a 1993 Standard Test Access Port and Boundary Scan Architecture. 1993

2. The Boundary Scan Handbook, Ken Parker, Klewer Academic Publishers, 1992

3. JEDEC Standard, Standard Data Transfer Format Between Data Preparation System and Programmable Logic Device Programmer JESD3-C, June 1994.

4. IEEE Std. 1149.1b Supplement (B) to Standard Test Access Port & Boundary Scan Architecture, IEEE Std. 1149.1 - 1990, 1994.

5. Boundary Scan Test: A Practical Approach, Harry Bleeker et al., Kluwer Academic Publishers, 1993.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
02/01/98	2.0	Revised release.
06/03/02	3.0	Revised release.