

This chapter describes the boundary-scan test (BST) features that are supported in Stratix® V devices.

Stratix V devices support IEEE Std. 1149.1 and IEEE Std. 1149.6. The IEEE Std. 1149.6 is only supported on the high-speed serial interface (HSSI) transceivers in Stratix V devices. IEEE Std. 1149.6 enables board-level connectivity checking between transmitters and receivers that are AC coupled (connected with a capacitor in series between the source and destination).

This chapter includes the following sections:

- “IEEE Std. 1149.6 Boundary-Scan Register” on page 11–2
- “BST Operation Control” on page 11–3
- “I/O Voltage Support in a JTAG Chain” on page 11–5
- “Boundary-Scan Description Language Support” on page 11–6



For more information about the following IEEE Std. 1149.1 BST features, refer to the *IEEE 1149.1 (JTAG) Boundary-Scan Testing in Stratix III Devices* chapter in volume 1 of the *Stratix III Device Handbook*:

- IEEE Std. 1149.1 BST architecture and circuitry
- IEEE Std. 1149.1 boundary-scan register
- IEEE Std. 1149.1 BST guidelines
- Test access port (TAP) controller state-machine

IEEE Std. 1149.6 Boundary-Scan Register

The boundary-scan cells (BSCs) for HSSI transmitters (GXB_TX [p, n]) and receivers/input clock buffer (GXB_RX [p, n])/(REFCLK [p, n]) in Stratix V devices are different from the BSCs for the I/O pins.

Figure 11-1 shows the Stratix V HSSI transmitter BSC.

Figure 11-1. HSSI Transmitter BSC with IEEE Std. 1149.6 BST Circuitry for Stratix V Devices

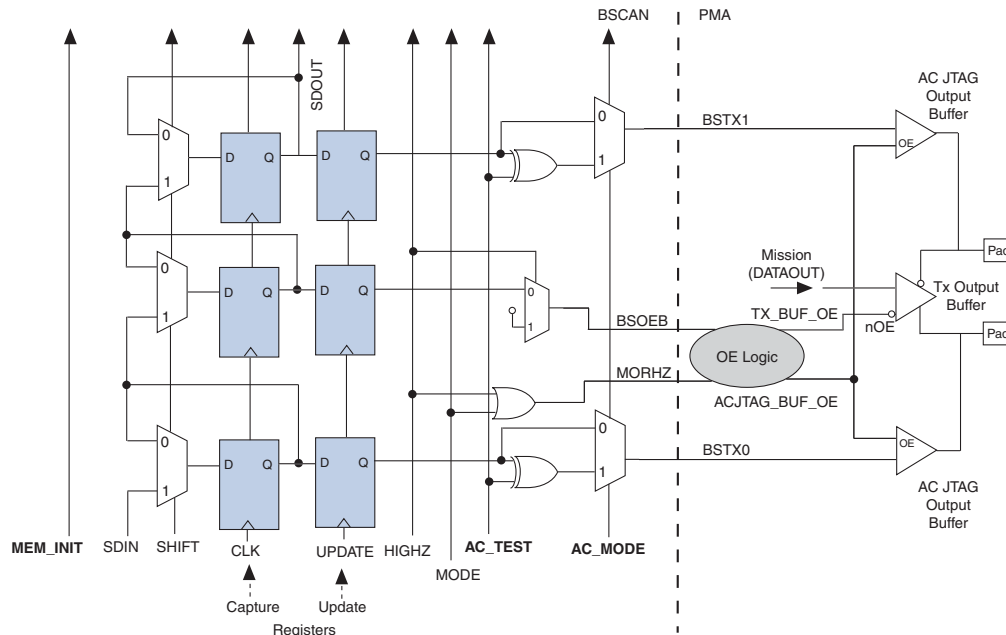
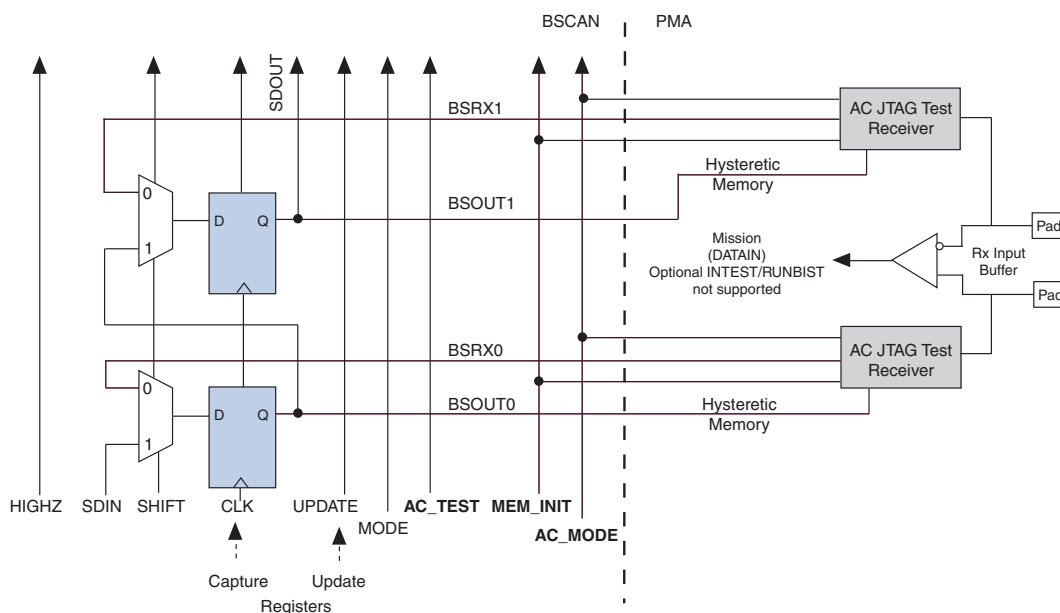


Figure 11-2 shows the Stratix V HSSI receiver/input clock buffer BSC.

Figure 11-2. HSSI Receiver/Input Clock Buffer BSC with IEEE Std. 1149.6 BST Circuitry for Stratix V Devices



BST Operation Control

Table 11-1 lists the IDCODE information for Stratix V devices.

Table 11-1. 32-Bit IDCODE Information for Stratix V Devices—Preliminary

Family	Device	IDCODE (32 Bits) ⁽¹⁾			
		Version (4 Bits)	Part Number (16 Bits)	Manufacturer Identity (11 Bits)	LSB (1 Bit) ⁽²⁾
Stratix V GX	5SGXA3	0000	0010 1001 0100 0111	000 1101 1101	1
	5SGXA4	0000	0010 1001 0010 0111	000 1101 1101	1
	5SGXA5	0000	0010 1001 0001 0011	000 1101 1101	1
	5SGXA7	0000	0010 1001 0000 0011	000 1101 1101	1
	5SGXA9	0000	0010 1001 0001 0101	000 1101 1101	1
	5SGXAB	0000	0010 1001 0000 0101	000 1101 1101	1
	5SGXB5	0000	0010 1001 0001 0010	000 1101 1101	1
Stratix V GT	5SGXB6	0000	0010 1001 0000 0010	000 1101 1101	1
	5SGTC5	0000	0010 1001 0010 0011	000 1101 1101	1
Stratix V GS	5SGTC7	0000	0010 1001 0100 0011	000 1101 1101	1
	5SGSD3	0000	0010 1001 0001 0001	000 1101 1101	1
	5SGSD4	0000	0010 1001 0000 0001	000 1101 1101	1
	5SGSD5	0000	0010 1001 0000 0111	000 1101 1101	1
	5SGSD6	0000	0010 1001 0001 0100	000 1101 1101	1
Stratix V E	5SGSD8	0000	0010 1001 0000 0100	000 1101 1101	1
	5SEE9	0000	0010 1001 0100 0101	000 1101 1101	1
	5SEEB	0000	0010 1001 0010 0101	000 1101 1101	1

Notes to Table 11-1:

- (1) The MSB is on the left.
- (2) The LSB of the IDCODE is always 1.

Table 11-2 lists the JTAG instructions that are supported by Stratix V devices.

Table 11-2. JTAG Instruction Supported by Stratix V Devices (Part 1 of 2)

JTAG Instruction	Instruction Code	Description
SAMPLE/PRELOAD	00 0000 0101	Allows a snapshot of signals at the device pins to be captured and examined during normal device operation and permits an initial data pattern to be an output at the device pins. Also used by the SignalTap™ II Embedded Logic Analyzer.
EXTEST ⁽¹⁾	00 0000 1111	Allows the external circuit and board-level interconnects to be tested by forcing a test pattern at the output pins and capturing test results at the input pins.
BYPASS	11 1111 1111	Places the 1-bit bypass register between the TDI and TDO pins, allowing the BST data to pass synchronously through selected devices to adjacent devices during normal device operation.
USERCODE	00 0000 0111	Selects the 32-bit USERCODE register and places it between the TDI and TDO pins, allowing USERCODE to be serially shifted out of TDO.

Table 11–2. JTAG Instruction Supported by Stratix V Devices (Part 2 of 2)

JTAG Instruction	Instruction Code	Description
IDCODE	00 0000 0110	Selects the IDCODE register and places it between the TDI and TDO pins, allowing IDCODE to be serially shifted out of TDO. IDCODE is the default instruction at power up and in the TAP RESET state.
HIGHZ ⁽¹⁾	00 0000 1011	Places the 1-bit bypass register between the TDI and TDO pins, allowing the BST data to pass synchronously through selected devices to adjacent devices during normal device operation while tri-stating all of the I/O pins.
CLAMP ⁽¹⁾	00 0000 1010	Places the 1-bit bypass register between the TDI and TDO pins, allowing the BST data to pass synchronously through selected devices to adjacent devices during normal device operation while holding I/O pins to a state defined by the data in the boundary-scan register.
PULSE_NCONFIG	00 0000 0001	Emulates pulsing the nCONFIG pin low to trigger reconfiguration even though the physical pin is unaffected.
CONFIG_IO	00 0000 1101	Allows I/O reconfiguration through JTAG ports using IOCSR for JTAG testing. This is executed after or during configurations. The nSTATUS pin must go high before you can issue the CONFIG_IO instruction.
LOCK	01 1111 0000	Disables the access by JTAG instructions and places the device into JTAG secure mode.
UNLOCK	11 0011 0001	Enables access for JTAG instructions that are not disabled by security fuses.
EXTEST_PULSE	00 1000 1111	Enables board-level connectivity checking between the transmitters and receivers that are AC coupled by generating three output transitions: <ul style="list-style-type: none"> ■ Driver drives data on the falling edge of TCK in the UPDATE_IR/DR state. ■ Driver drives inverted data on the falling edge of TCK after entering the RUN_TEST/IDLE state. ■ Driver drives data on the falling edge of TCK after leaving the RUN_TEST/IDLE state.
EXTEST_TRAIN	00 0100 1111	Behaves the same as the EXTEST_PULSE instruction except that the output continues to toggle on the TCK falling edge as long as the TAP controller is in the RUN_TEST/IDLE state.

Note to Table 11–2:

(1) Bus hold and weak pull-up resistor features override the high-impedance state of HIGHZ, CLAMP, and EXTEST.

If any of the security fuses is blown, some of the JTAG instructions are disabled. UNLOCK instruction enables the JTAG instructions that are not blocked by these fuses. LOCK instruction disables these JTAG instructions, returning to the “JTAG secure” state. You can only issue the LOCK or UNLOCK instruction from the JTAG core access.

The IDCODE instruction is the default instruction when the TAP controller is in the reset state. Without loading any instructions, you can go to the SHIFT_DR state and shift out the JTAG device ID.



If the device is in reset state, when the nCONFIG or nSTATUS signal is low, the device IDCODE might not be read correctly. To read the device IDCODE correctly, you must issue the IDCODE JTAG instruction only when the nCONFIG and nSTATUS signals are high.

IEEE Std. 1149.6 mandates the addition of two new instructions—EXTEST_PULSE and EXTEST_TRAIN. These two instructions enable edge-detecting behavior on the signal path containing the HSSI pins. These instructions implement new test behaviors for the HSSI pins and simultaneously behave identically to the IEEE Std. 1149.1 EXTEST instruction for non-HSSI pins.



If you use DC coupling on the HSSI signals, execute the EXTEST instruction. If you use AC coupling on the HSSI signals, execute the EXTEST_PULSE instruction.

I/O Voltage Support in a JTAG Chain

A device operating in BST mode uses four required pins—TDI, TDO, TMS, TCK, and one optional pin, TRST. The TCK pin has an internal weak pull-down resistor, while the TDI, TMS, and TRST pins have internal weak pull-up resistors. The TDO output pin and all the JTAG input pins are powered by the 2.5-V/3.0-V V_{CCPD} supply of I/O bank 3A. All user I/O pins are tri-stated during JTAG configuration.

The JTAG chain can support several different devices. However, use caution if the chain contains devices that have different V_{CCIO} levels. The output voltage level of the TDO pin must meet the specification of the TDI pin it drives.

Table 11-3 lists board design recommendations to ensure proper JTAG chain operation.

Table 11-3. Supported TDO and TDI Voltage Combinations



Device	TDI Input Buffer Power	Stratix V TDO V_{CCPD}	
		$V_{CCPD} = 3.0\text{ V}$ (1)	$V_{CCPD} = 2.5\text{ V}$ (2)
Stratix V	$V_{CCPD} = 3.0\text{ V}$	✓	✓
	$V_{CCPD} = 2.5\text{ V}$	✓	✓
Non-Stratix V	$V_{CC} = 3.3\text{ V}$	✓ (3)	✓ (4)
	$V_{CC} = 2.5\text{ V}$	✓ (3)	✓ (4)
	$V_{CC} = 1.8\text{ V}$	✓ (3)	✓ (4)
	$V_{CC} = 1.5\text{ V}$	✓ (3)	✓ (4)

Notes to Table 11-3:

- (1) The TDO output buffer meets $V_{OH}(\text{MIN}) = 2.4\text{ V}$.
- (2) The TDO output buffer meets $V_{OH}(\text{MIN}) = 2.0\text{ V}$.
- (3) Input buffer must be 3.0-V tolerant.
- (4) Input buffer must be 2.5-V tolerant.

Boundary-Scan Description Language Support

The boundary-scan description language (BSDL), a subset of VHDL, provides a syntax that allows you to describe the features of an IEEE Std. 1149.6 BST-capable device that can be tested. You can test software development systems, then use the BSDL files for test generation, analysis, and failure diagnostics.

-  For more information about BSDL files for IEEE Std. 1149.6-compliant Stratix V devices, refer to the [IEEE 1149.6 BSDL Files](#) page on the Altera website.
-  You can also generate BSDL files (pre-configuration and post-configuration) for IEEE Std. 1149.6-compliant Stratix V devices with the Quartus® II software version 10.0 SP1 and later. For more information about the procedure to generate BSDL files using the Quartus II software, refer to the [BSDL Files Generation in QII](#) page on the Altera website.

Document Revision History

[Table 11-4](#) lists the revision history for this chapter.

Table 11-4. Document Revision History

Date	Version	Changes
November 2011	1.3	Updated Table 11-1 and Table 11-2 .
May 2011	1.2	<ul style="list-style-type: none"> ■ Chapter moved to volume 2 for the 11.0 release. ■ Updated Table 11-1.
December 2010	1.1	No changes to the content of this chapter for the Quartus II software 10.1.
July 2010	1.0	Initial release.