

IEEE Std. 1149.1 (JTAG) Boundary-Scan Support

All Stratix® devices provide JTAG BST circuitry that complies with the IEEE Std. 1149.1a-1990 specification. JTAG boundary-scan testing can be performed either before or after, but not during configuration. Stratix devices can also use the JTAG port for configuration together with either the Quartus® II software or hardware using either Jam Files (.jam) or Jam Byte-Code Files (.jbc).

Stratix devices support IOE I/O standard setting reconfiguration through the JTAG BST chain. The JTAG chain can update the I/O standard for all input and output pins any time before or during user mode through the CONFIG_IO instruction. You can use this ability for JTAG testing before configuration when some of the Stratix pins drive or receive from other devices on the board using voltage-referenced standards. Since the Stratix device may not be configured before JTAG testing, the I/O pins may not be configured for appropriate electrical standards for chip-to-chip communication. Programming those I/O standards via JTAG allows you to fully test the I/O connection to other devices.

The enhanced PLL reconfiguration bits are part of the JTAG chain before configuration and after power-up. After device configuration, the PLL reconfiguration bits are not part of the JTAG chain.

The JTAG pins support 1.5-V/1.8-V or 2.5-V/3.3-V I/O standards. The TDO pin voltage is determined by the V_{CCIO} of the bank where it resides. The VCCSEL pin selects whether the JTAG inputs are 1.5-V, 1.8-V, 2.5-V, or 3.3-V compatible.

Stratix devices also use the JTAG port to monitor the logic operation of the device with the SignalTap® II embedded logic analyzer. Stratix devices support the JTAG instructions shown in [Table 3-1](#).

The Quartus II software has an Auto Usercode feature where you can choose to use the checksum value of a programming file as the JTAG user code. If selected, the checksum is automatically loaded to the USERCODE register. In the Settings dialog box in the Assignments menu, click **Device & Pin Options**, then **General**, and then turn on the **Auto Usercode** option.

<i>Table 3–1. Stratix JTAG Instructions</i>		
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 output at the device pins. Also used by the SignalTap II embedded logic analyzer.
EXTEST (1)	00 0000 0000	Allows the external circuitry 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, which allows 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 the USERCODE to be serially shifted out of TDO.
IDCODE	00 0000 0110	Selects the IDCODE register and places it between TDI and TDO, allowing the IDCODE to be serially shifted out of TDO.
HIGHZ (1)	00 0000 1011	Places the 1-bit bypass register between the TDI and TDO pins, which allows 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, which allows 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.
ICR instructions		Used when configuring an Stratix device via the JTAG port with a MasterBlaster™, ByteBlasterMV™, or ByteBlaster™ II download cable, or when using a Jam File or Jam Byte-Code File via an embedded processor or JRunner.
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 configuration of I/O standards through the JTAG chain for JTAG testing. Can be executed before, after, or during configuration. Stops configuration if executed during configuration. Once issued, the CONFIG_IO instruction will hold nSTATUS low to reset the configuration device. nSTATUS is held low until the device is reconfigured.
SignalTap II instructions		Monitors internal device operation with the SignalTap II embedded logic analyzer.

Note to Table 3–1:

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

The Stratix device instruction register length is 10 bits and the USERCODE register length is 32 bits. Tables 3–2 and 3–3 show the boundary-scan register length and device IDCODE information for Stratix devices.

Table 3–2. Stratix Boundary-Scan Register Length

Device	Boundary-Scan Register Length
EP1S10	1,317
EP1S20	1,797
EP1S25	2,157
EP1S30	2,253
EP1S40	2,529
EP1S60	3,129
EP1S80	3,777

Table 3–3. 32-Bit Stratix Device IDCODE

Device	IDCODE (32 Bits) (1)			
	Version (4 Bits)	Part Number (16 Bits)	Manufacturer Identity (11 Bits)	LSB (1 Bit) (2)
EP1S10	0000	0010 0000 0000 0001	000 0110 1110	1
EP1S20	0000	0010 0000 0000 0010	000 0110 1110	1
EP1S25	0000	0010 0000 0000 0011	000 0110 1110	1
EP1S30	0000	0010 0000 0000 0100	000 0110 1110	1
EP1S40	0000	0010 0000 0000 0101	000 0110 1110	1
EP1S60	0000	0010 0000 0000 0110	000 0110 1110	1
EP1S80	0000	0010 0000 0000 0111	000 0110 1110	1

Notes to Tables 3–2 and 3–3:

- (1) The most significant bit (MSB) is on the left.
- (2) The IDCODE's least significant bit (LSB) is always 1.

Figure 3–1 shows the timing requirements for the JTAG signals.

Figure 3–1. Stratix JTAG Waveforms

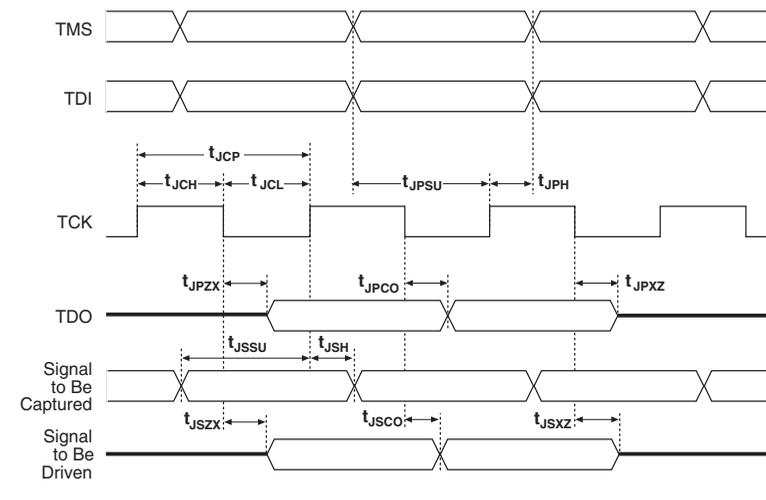


Table 3–4 shows the JTAG timing parameters and values for Stratix devices.

Symbol	Parameter	Min	Max	Unit
t_{JCP}	TCK clock period	100		ns
t_{JCH}	TCK clock high time	50		ns
t_{JCL}	TCK clock low time	50		ns
t_{JPSU}	JTAG port setup time	20		ns
t_{JPH}	JTAG port hold time	45		ns
t_{JPCO}	JTAG port clock to output		25	ns
t_{JPZX}	JTAG port high impedance to valid output		25	ns
t_{JPXZ}	JTAG port valid output to high impedance		25	ns
t_{JSSU}	Capture register setup time	20		ns
t_{JSH}	Capture register hold time	45		ns
t_{JSCO}	Update register clock to output		35	ns
t_{JSZX}	Update register high impedance to valid output		35	ns
t_{JSXZ}	Update register valid output to high impedance		35	ns



Stratix, Stratix II, Cyclone®, and Cyclone II devices must be within the first 17 devices in a JTAG chain. All of these devices have the same JTAG controller. If any of the Stratix, Stratix II, Cyclone, and Cyclone II devices are in the 18th or after they will fail configuration. This does not affect SignalTap II.



For more information on JTAG, see the following documents:

- *AN 39: IEEE Std. 1149.1 (JTAG) Boundary-Scan Testing in Altera Devices*
- *Jam Programming & Test Language Specification*

SignalTap II Embedded Logic Analyzer

Stratix devices feature the SignalTap II embedded logic analyzer, which monitors design operation over a period of time through the IEEE Std. 1149.1 (JTAG) circuitry. You can analyze internal logic at speed without bringing internal signals to the I/O pins. This feature is particularly important for advanced packages, such as FineLine BGA® packages, because it can be difficult to add a connection to a pin during the debugging process after a board is designed and manufactured.

Configuration

The logic, circuitry, and interconnects in the Stratix architecture are configured with CMOS SRAM elements. Altera® devices are reconfigurable. Because every device is tested with a high-coverage production test program, you do not have to perform fault testing and can focus on simulation and design verification.

Stratix devices are configured at system power-up with data stored in an Altera serial configuration device or provided by a system controller. Altera offers in-system programmability (ISP)-capable configuration devices that configure Stratix devices via a serial data stream. Stratix devices can be configured in under 100 ms using 8-bit parallel data at 100 MHz. The Stratix device's optimized interface allows microprocessors to configure it serially or in parallel, and synchronously or asynchronously. The interface also enables microprocessors to treat Stratix devices as memory and configure them by writing to a virtual memory location, making reconfiguration easy. After a Stratix device has been configured, it can be reconfigured in-circuit by resetting the device and loading new data. Real-time changes can be made during system operation, enabling innovative reconfigurable computing applications.

Operating Modes

The Stratix architecture uses SRAM configuration elements that require configuration data to be loaded each time the circuit powers up. The process of physically loading the SRAM data into the device is called configuration. During initialization, which occurs immediately after

configuration, the device resets registers, enables I/O pins, and begins to operate as a logic device. The I/O pins are tri-stated during power-up, and before and during configuration. Together, the configuration and initialization processes are called command mode. Normal device operation is called user mode.

SRAM configuration elements allow Stratix devices to be reconfigured in-circuit by loading new configuration data into the device. With real-time reconfiguration, the device is forced into command mode with a device pin. The configuration process loads different configuration data, reinitializes the device, and resumes user-mode operation. You can perform in-field upgrades by distributing new configuration files either within the system or remotely.

PORSEL is a dedicated input pin used to select POR delay times of 2 ms or 100 ms during power-up. When the PORSEL pin is connected to ground, the POR time is 100 ms; when the PORSEL pin is connected to V_{CC} , the POR time is 2 ms.

The nIO_PULLUP pin enables a built-in weak pull-up resistor to pull all user I/O pins to V_{CCIO} before and during device configuration. If nIO_PULLUP is connected to V_{CC} during configuration, the weak pull-ups on all user I/O pins are disabled. If connected to ground, the pull-ups are enabled during configuration. The nIO_PULLUP pin can be pulled to 1.5, 1.8, 2.5, or 3.3 V for a logic level high.

VCCSEL is a dedicated input that is used to choose whether all dedicated configuration and JTAG input pins can accept 1.5 V/1.8 V or 2.5 V/3.3 V during configuration. A logic low sets 3.3 V/2.5 V, and a logic high sets 1.8 V/1.5 V. VCCSEL affects the following pins: TDI, TMS, TCK, TRST, MSEL0, MSEL1, MSEL2, nCONFIG, nCE, DCLK, PLL_ENA, CONF_DONE, nSTATUS. The VCCSEL pin can be pulled to 1.5, 1.8, 2.5, or 3.3 V for a logic level high.

The VCCSEL signal does not control the dual-purpose configuration pins such as the DATA [7 . . 0] and PPA pins (nWS, nRS, CS, nCS, and RDYnBSY). During configuration, these dual-purpose pins will drive out voltage levels corresponding to the V_{CCIO} supply voltage that powers the I/O bank containing the pin. After configuration, the dual-purpose pins use I/O standards specified in the user design.

TDO and nCEO drive out at the same voltages as the V_{CCIO} supply that powers the I/O bank containing the pin. Users must select the V_{CCIO} supply for bank containing TDO accordingly. For example, when using the ByteBlaster™ MV cable, the V_{CCIO} for the bank containing TDO must be powered up at 3.3 V.

Configuring Stratix FPGAs with JRunner

JRunner is a software driver that configures Altera FPGAs, including Stratix FPGAs, through the ByteBlaster II or ByteBlasterMV cables in JTAG mode. The programming input file supported is in Raw Binary File (.rbf) format. JRunner also requires a Chain Description File (.cdf) generated by the Quartus II software. JRunner is targeted for embedded JTAG configuration. The source code is developed for the Windows NT operating system (OS), but can be customized to run on other platforms. For more information on the JRunner software driver, see the JRunner Software Driver: An Embedded Solution to the JTAG Configuration White Paper and the source files on the Altera web site (www.altera.com).

Configuration Schemes

You can load the configuration data for a Stratix device with one of five configuration schemes (see [Table 3–5](#)), chosen on the basis of the target application. You can use a configuration device, intelligent controller, or the JTAG port to configure a Stratix device. A configuration device can automatically configure a Stratix device at system power-up.

Multiple Stratix devices can be configured in any of five configuration schemes by connecting the configuration enable (nCE) and configuration enable output (nCEO) pins on each device.

Configuration Scheme	Data Source
Configuration device	Enhanced or EPC2 configuration device
Passive serial (PS)	MasterBlaster, ByteBlasterMV, or ByteBlaster II download cable or serial data source
Passive parallel asynchronous (PPA)	Parallel data source
Fast passive parallel	Parallel data source
JTAG	MasterBlaster, ByteBlasterMV, or ByteBlaster II download cable, a microprocessor with a Jam or JBC file, or JRunner

Partial Reconfiguration

The enhanced PLLs within the Stratix device family support partial reconfiguration of their multiply, divide, and time delay settings without reconfiguring the entire device. You can use either serial data from the logic array or regular I/O pins to program the PLL's counter settings in a serial chain. This option provides considerable flexibility for frequency

synthesis, allowing real-time variation of the PLL frequency and delay. The rest of the device is functional while reconfiguring the PLL. See the *Stratix Architecture* chapter of the *Stratix Device Handbook, Volume 1* for more information on Stratix PLLs.

Remote Update Configuration Modes

Stratix devices also support remote configuration using an Altera enhanced configuration device (e.g., EPC16, EPC8, and EPC4 devices) with page mode selection. Factory configuration data is stored in the default page of the configuration device. This is the default configuration that contains the design required to control remote updates and handle or recover from errors. You write the factory configuration once into the flash memory or configuration device. Remote update data can update any of the remaining pages of the configuration device. If there is an error or corruption in a remote update configuration, the configuration device reverts back to the factory configuration information.

There are two remote configuration modes: remote and local configuration. You can use the remote update configuration mode for all three configuration modes: serial, parallel synchronous, and parallel asynchronous. Configuration devices (for example, EPC16 devices) only support serial and parallel synchronous modes. Asynchronous parallel mode allows remote updates when an intelligent host is used to configure the Stratix device. This host must support page mode settings similar to an EPC16 device.

Remote Update Mode

When the Stratix device is first powered up in remote update programming mode, it loads the configuration located at page address "000." The factory configuration should always be located at page address "000," and should never be remotely updated. The factory configuration contains the required logic to perform the following operations:

- Determine the page address/load location for the next application's configuration data
- Recover from a previous configuration error
- Receive new configuration data and write it into the configuration device

The factory configuration is the default and takes control if an error occurs while loading the application configuration.

While in the factory configuration, the factory-configuration logic performs the following operations:

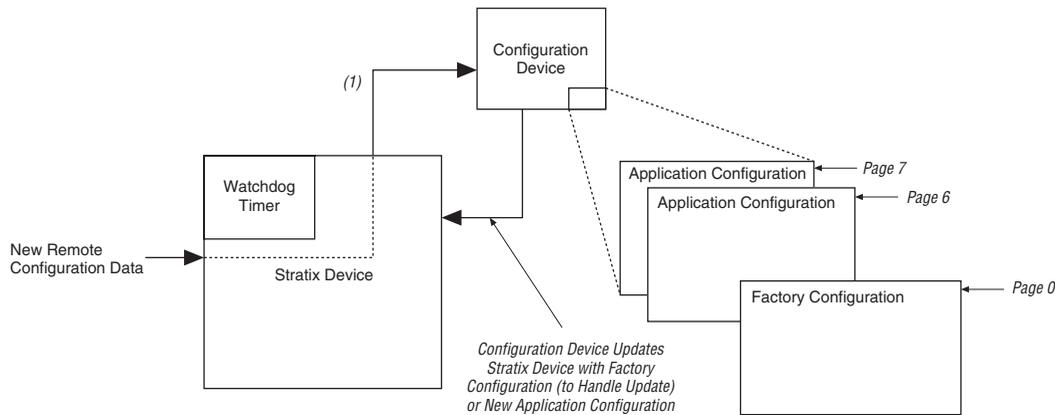
- Loads a remote update-control register to determine the page address of the new application configuration
- Determines whether to enable a user watchdog timer for the application configuration
- Determines what the watchdog timer setting should be if it is enabled

The user watchdog timer is a counter that must be continually reset within a specific amount of time in the user mode of an application configuration to ensure that valid configuration occurred during a remote update. Only valid application configurations designed for remote update can reset the user watchdog timer in user mode. If a valid application configuration does not reset the user watchdog timer in a specific amount of time, the timer updates a status register and loads the factory configuration. The user watchdog timer is automatically disabled for factory configurations.

If an error occurs in loading the application configuration, the configuration logic writes a status register to specify the cause of the error. Once this occurs, the Stratix device automatically loads the factory configuration, which reads the status register and determines the reason for reconfiguration. Based on the reason, the factory configuration will take appropriate steps and will write the remote update control register to specify the next application configuration page to be loaded.

When the Stratix device successfully loads the application configuration, it enters into user mode. The Stratix device then executes the main application of the user. Intellectual property (IP), such as a Nios® (16-bit ISA) and Nios® II (32-bit ISA) embedded processors, can help the Stratix device determine when remote update is coming. The Nios embedded processor or user logic receives incoming data, writes it to the configuration device, and loads the factory configuration. The factory configuration will read the remote update status register and determine the valid application configuration to load. [Figure 3-2](#) shows the Stratix remote update. [Figure 3-3](#) shows the transition diagram for remote update mode.

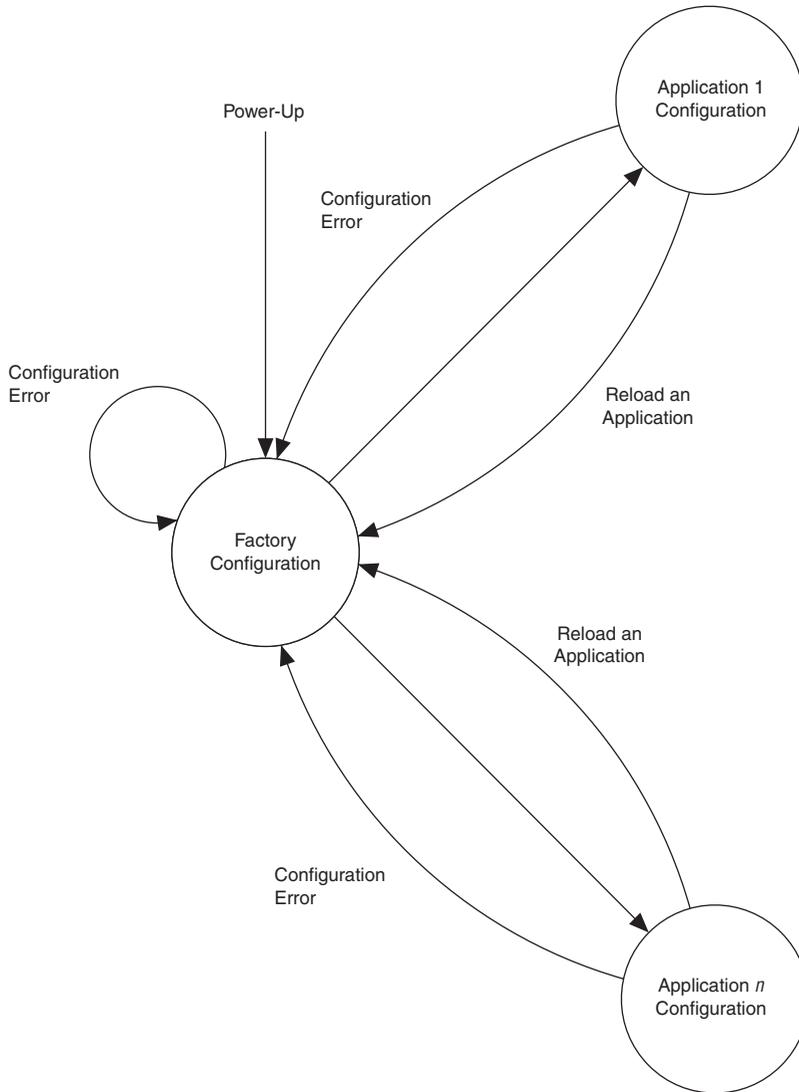
Figure 3–2. Stratix Device Remote Update



Note to Figure 3–2:

- (1) When the Stratix device is configured with the factory configuration, it can handle update data from EPC16, EPC8, or EPC4 configuration device pages and point to the next page in the configuration device.
-

Figure 3–3. Remote Update Transition Diagram Notes (1), (2)



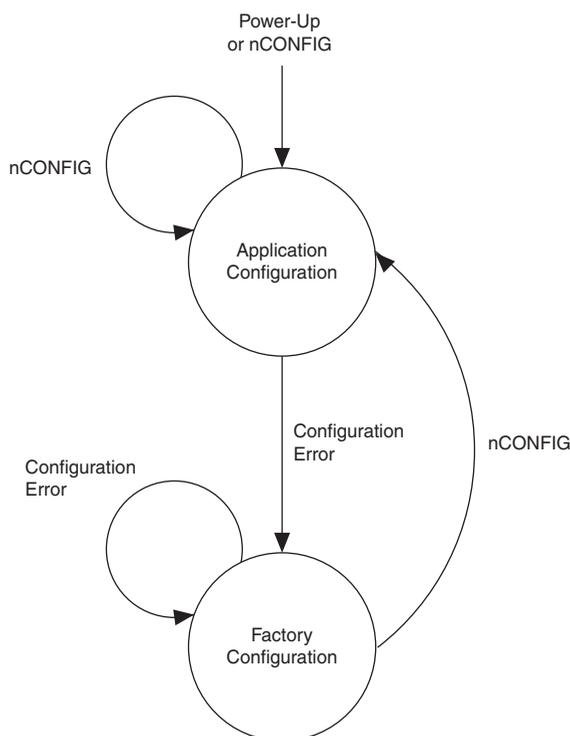
Notes to Figure 3–3:

- (1) Remote update of Application Configuration is controlled by a Nios embedded processor or user logic programmed in the Factory or Application configurations.
- (2) Up to seven pages can be specified allowing up to seven different configuration applications.

Local Update Mode

Local update mode is a simplified version of the remote update. This feature is intended for simple systems that need to load a single application configuration immediately upon power up without loading the factory configuration first. Local update designs have only one application configuration to load, so it does not require a factory configuration to determine which application configuration to use. Figure 3-4 shows the transition diagram for local update mode.

Figure 3-4. Local Update Transition Diagram



Stratix Automated Single Event Upset (SEU) Detection

Stratix devices offer on-chip circuitry for automated checking of single event upset (SEU) detection. FPGA devices that operate at high elevations or in close proximity to earth's North or South Pole require periodic checks to ensure continued data integrity. The error detection cyclic redundancy check (CRC) feature controlled by the **Device & Pin Options** dialog box in the Quartus II software uses a 32-bit CRC circuit to ensure data reliability and is one of the best options for mitigating SEU.

For Stratix, the CRC is computed by the Quartus II software and downloaded into the device as a part of the configuration bit stream. The `CRC_ERROR` pin reports a soft error when configuration SRAM data is corrupted, triggering device reconfiguration.

Custom-Built Circuitry

Dedicated circuitry is built in the Stratix devices to perform error detection automatically. You can use the built-in dedicated circuitry for error detection using CRC feature in Stratix devices, eliminating the need for external logic. This circuitry will perform error detection automatically when enabled. This error detection circuitry in Stratix devices constantly checks for errors in the configuration SRAM cells while the device is in user mode. You can monitor one external pin for the error and use it to trigger a re-configuration cycle. Select the desired time between checks by adjusting a built-in clock divider.

Software Interface

In the Quartus II software version 4.1 and later, you can turn on the automated error detection CRC feature in the **Device & Pin Options** dialog box. This dialog box allows you to enable the feature and set the internal frequency of the CRC between 400 kHz to 100 MHz. This controls the rate that the CRC circuitry verifies the internal configuration SRAM bits in the FPGA device.

For more information on CRC, see *AN 357: Error Detection Using CRC in Altera FPGA Devices*.

Temperature Sensing Diode

Stratix devices include a diode-connected transistor for use as a temperature sensor in power management. This diode is used with an external digital thermometer device such as a MAX1617A or MAX1619 from MAXIM Integrated Products. These devices steer bias current through the Stratix diode, measuring forward voltage and converting this reading to temperature in the form of an 8-bit signed number (7 bits plus sign). The external device's output represents the junction temperature of the Stratix device and can be used for intelligent power management.

The diode requires two pins (`tempdiodep` and `tempdioden`) on the Stratix device to connect to the external temperature-sensing device, as shown in [Figure 3-5](#). The temperature sensing diode is a passive element and therefore can be used before the Stratix device is powered.

Figure 3–5. External Temperature-Sensing Diode

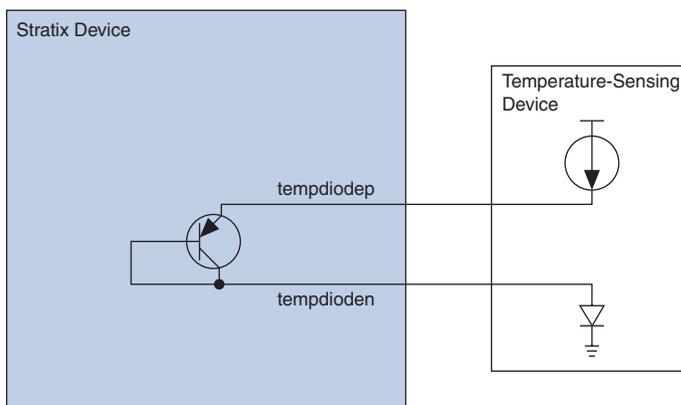


Table 3–6 shows the specifications for bias voltage and current of the Stratix temperature sensing diode.

Table 3–6. Temperature-Sensing Diode Electrical Characteristics

Parameter	Minimum	Typical	Maximum	Unit
$I_{\text{BIAS high}}$	80	100	120	μA
$I_{\text{BIAS low}}$	8	10	12	μA
$V_{\text{BP}} - V_{\text{BN}}$	0.3		0.9	V
V_{BN}		0.7		V
Series resistance			3	W

The temperature-sensing diode works for the entire operating range shown in [Figure 3-6](#).

Figure 3-6. Temperature vs. Temperature-Sensing Diode Voltage

