Enhanced Configuration Devices

(EPC4, EPC8, & EPC16)

September 2001, ver. 1.0 Data Sheet

Features

- Enhanced configuration devices include EPC4, EPC8, and EPC16 devices
- 4-, 8-, and 16-Mbit Flash memory devices that configure APEX™ II, APEX 20K, Mercury™, ACEX™ 1K, and FLEX® 10K devices
  - Compression increases effective configuration density of these devices up to 7, 15, or 30 Mbits
- Available in the 100-pin plastic quad flat pack (PQFP) package and the 88-pin Ultra FineLine BGA™ package
- Standard Flash die and a controller die combined into one package
- V_{CCINT} and V_{CCIO} are both 3.3 V
- Supports true N-bit (N = 1, 2, 4, and 8) programmable logic device (PLD) concurrent configuration mode
  - Configures multiple PLDs in parallel
  - Supports an 8-bit parallel data output on every DCLK cycle

Preliminary Information

- Pin-selectable 2-ms or 100-ms power-on reset (POR) time
- Programmable clock speed with three clock modes for faster configuration time
  - Internal oscillator defaults to 10 MHz
  - Programmable internal oscillator for higher frequencies of 33, 50, and 66 MHz
  - External clock source with frequencies up to 133 MHz
- EPC16 configuration device allows PLD or processor to access unused Flash memory locations via external flash interface
- Flash memory can hold up to eight pages of configuration files, enabling systems to reconfigure PLDs with different configuration files
- Flash block/sector protection capability (EPC16 configuration devices only)
- Compliant with IEEE Std. 1532 in-system programmability (ISP) specification
- Supports ISP via Jam™ Standard Test and Programming Language (STAPL)
- Supports Joint Test Action Group (JTAG) boundary scan
- nINIT_CONF pin allows private JTAG instruction to initiate PLD configuration
- Programmable configuration done error detection capability
- Internal programmable weak pull-ups on nCS and OE pins, Flash address, and control lines, and bus hold on data line
- Standby mode with reduced power consumption

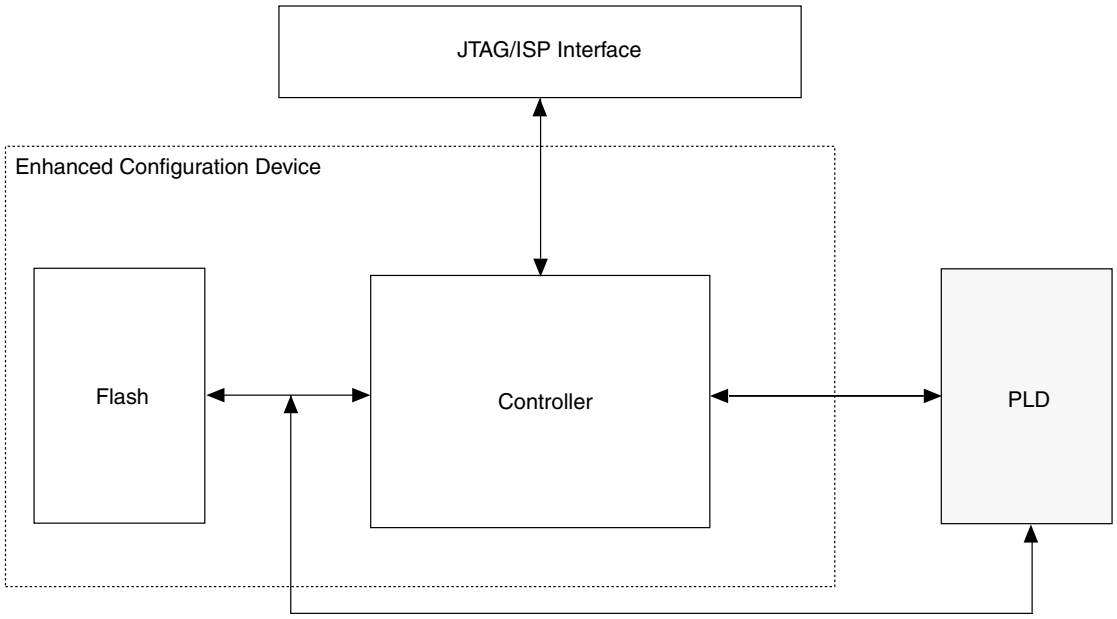
Architecture Description

The Altera® enhanced configuration devices support a single-device solution for very high-density PLDs while decreasing configuration time. The core of an enhanced configuration device is divided into two major blocks, the controller and the Flash memory. The Flash memory can be used for APEX II, APEX 20K, Mercury, ACEX, and FLEX 10K device configuration, and its unused locations can be used as memory storage for the PLD or processor.

All references to the direct Flash interface in this document are for EPC16 configuration devices only. For information on using Flash memory interface in the EPC4 or EPC8 configuration devices, please contact Altera Applications.

Figure 1 shows a block diagram of the enhanced configuration device's core blocks, their connection to the PLD, and their interface with the JTAG/ISP interface.

Figure 1. Enhanced Configuration Device Block Diagram



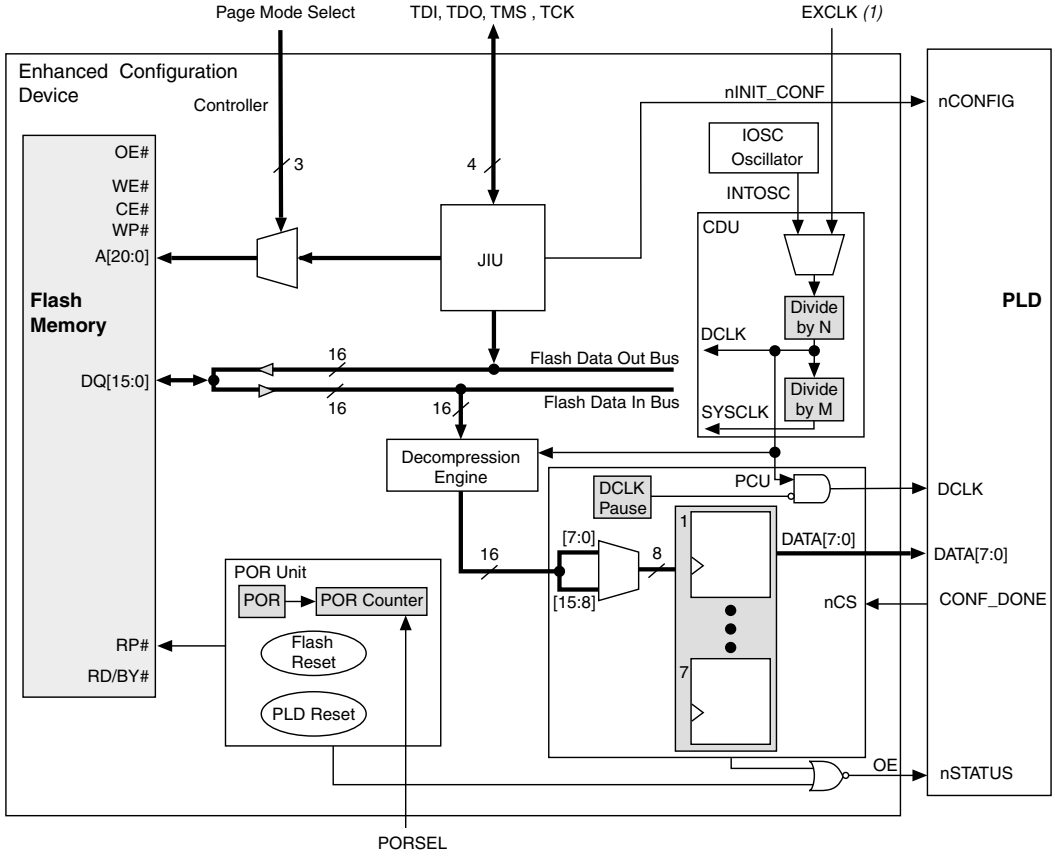
Enhanced Configuration Device Controller Unit

The controller unit of the enhanced configuration device has a 3.3-V core and an I/O interface. The controller is a synchronous system that includes the following:

- Power-on reset circuitry (POR)
- Internal oscillator (IOSC)
- Clock divider unit (CDU)
- Decompression engine
- PLD configuration unit (PCU)
- JTAG interface unit (JIU)

Figure 2 shows a block diagram of the enhanced configuration device controller unit.

Figure 2. Enhanced Configuration Device Controller Unit Block Diagram



Note to Figure 2:
(1) EXCLK should be connected to VCC or GND if it is not being used.

Power-On Reset Unit

The POR circuit keeps the system in reset until the power supply voltage levels have stabilized. The enhanced configuration device has two options for the POR time: the user can either keep the POR time at the 100-ms default value or reduce the POR time through the selectable input pin to 2 ms for applications that require fast power-up. The PORSEL input pin controls the POR reduction time from 100 ms to 2 ms. See Table 7 on page 28 for more information.

The POR unit manages the controller's reset scheme. When the POR counter expires, the POR unit releases the OE pin. The POR time can be further extended from an external source by driving the OE pin low.

Do not execute JTAG or ISP instructions until POR is complete.

The enhanced configuration device reset can be divided into three categories:

- The POR reset starts at initial power-up reset during V_{CC} ramp or if V_{CC} drops anytime after V_{CC} has stabilized.
- The PLD initiates re-configuration by driving nSTATUS low, which occurs if the PLD detects a cyclic redundancy check (CRC) error or if the nCONFIG input pin is asserted in the PLD.
- The controller detects an error and asserts the OE to initiate re-configuration of APEX II, APEX 20K, Mercury, ACEX 1K, and FLEX 10K devices when the auto restart upon error option is enabled in software.

Internal Oscillator

Frequencies for the internal oscillator (IOSC) of the enhanced configuration device, which supports four modes of internal clock frequencies, are shown in Table 1. The user can program the oscillator, which is controlled by option bits through the software.

Table 1. Internal Oscillator Frequencies

Mode	Min (MHz)	Typ (MHz)	Max (MHz)
A	6.4	8.0	10.0
B	21.0	26.5	33.0
C	32.0	40.0	50.0
D	42.0	53.0	66.0

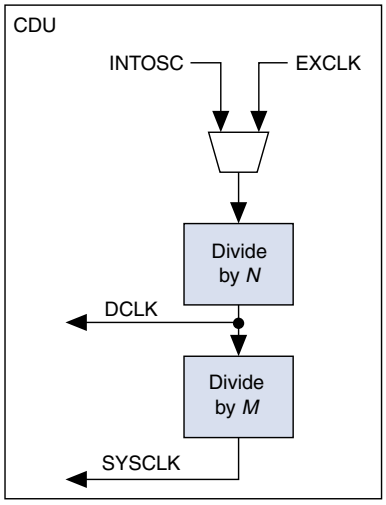
Clock Divider Unit

The CDU generates SYSCLK and DCLK for the controller by dividing the internal oscillator clock (INTOSC) or external clock (EXCLK). The CDU's clock division architecture has two dividers. The first divider (*N*) divides down the selected reference clock to generate DCLK. The second divider (*M*) divides down DCLK to generate SYSCLK. Each divider contains a 1 to 16 integer divider. Both a 1.5 divider and a 2.5 divider are also implemented in the first divider (*N*), but the second divider (*M*) can only divide integers. As a default from power-up, the INTOSC is in mode A, the first divider is set to divide by one to generate the DCLK, and the second divider is set to divide by two to generate the SYSCLK (see Figure 3).

The default duty cycle for all clock divisions other than non-integer divisions is 50% (for the non-integer dividers, the duty cycle will not be 50%). For integer divisions, the CDU allows the duty cycle of DCLK and SYSCLK to be programmable by setting appropriate option bits through the software. The DCLK frequency is limited by the maximum DCLK frequency of the PLD, but the SYSCLK frequency is limited by the maximum Flash performance (about 10 MHz). Therefore, DCLK and SYSCLK might run at different frequencies. See Figure 3 for details on the CDU.

The maximum DCLK frequency for each PLD family is specified in *Application Note 116 (Configuring SRAM-Based LUT Devices)*.

Figure 3. Clock Divider Unit



Decompression Engine

Enhanced configuration devices support decompression. Configuration data is compressed by the Quartus™ II software and stored in the enhanced configuration device. During configuration, the decompression engine inside the enhanced configuration device will decompress or expand data. This feature increases the effective configuration density of the enhanced configuration device up to 7, 15, or 30 Mbits in EPC4, EPC8, and EPC16, respectively.

The enhanced configuration device also supports a parallel data bus to the PLD to reduce configuration time. However, in some cases, the PLD data transfer is limited by the Flash data transfer rate. With parallel programming mode in the PLD (when *N* = 8 and the DCLK frequency is 66 MHz), the data output bandwidth to the PLD is faster than the data input bandwidth reading from the Flash. Because configuration time depends on the ratio of data bits read and the bandwidth used, the compression will improve configuration time. The decompression engine decompresses the configuration data before sending it to the PLD configuration unit (PCU) for PLD configuration.

PLD Configuration Unit

The function of the PCU is to transmit decompressed data to the PLD, depending on the configuration mode. The enhanced configuration device supports four concurrent configuration modes, with *N* = 1, 2, 4, or 8. Depending on the data width, the PCU shifts the data to transmit appropriate data to the valid data pins. Unused data pins drive low.

In addition to transmitting data to the PLD, the PCU is responsible for delaying the DCLK to the PLD whenever there is insufficient decompressed data, i.e., when waiting for the decompression engine to decompress data. This technique is called "Pausing DCLK."

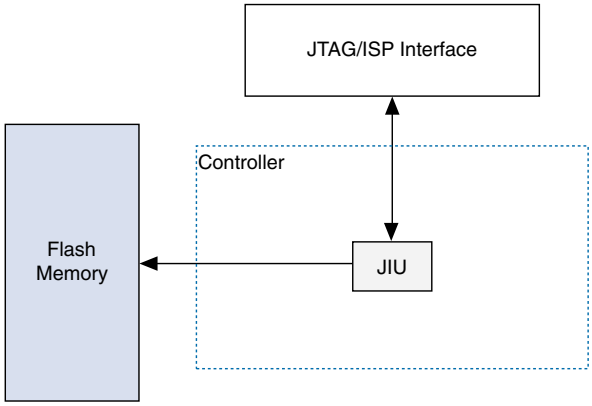
The PCU manages the CONF_DONE error detection logic. A CONF_DONE error occurs when nCS is not de-asserted within a certain number of clock cycles after the last data bit is transmitted to the PLD. When a CONF_DONE error is detected, the PCU asserts signals to the POR unit, which asserts the OE pin to start re-configuring the PLD. This is done only when the auto-restart configuration upon frame error option is enabled in software.

JTAG Interface Unit

The IEEE Std. 1149.1 JTAG Boundary Scan is implemented in enhanced configuration devices to facilitate the testing of its interconnection and functionality. Enhanced configuration devices also support the ISP mode. The enhanced configuration device's ISP is compliant with the IEEE Std. 1532 draft 2.0 specification. In addition to programming, erasing, and verifying the Flash, enhanced configuration devices (EPC16 configuration devices) also support block/sector protection through IEEE Std. 1532-compliant instructions.

The JTAG interface unit (JIU) communicates directly with the Flash memory (see Figure 4). The JIU operates at the maximum JTAG TCK frequency of 10 MHz.

Figure 4. JTAG/ISP Interface



Before the JTAG/ISP interface programs the Flash memory, a JTAG instruction (PENDCFG) asserts the PLD's nCONFIG pin (connected to nINIT_CONF pin), which will terminate any access to Flash. When the ISP mode starts, the JIU takes over the Flash memory. If the ISP mode starts during PLD configuration, the configuration terminates immediately.

Flash Memory

The Flash memory in EPC4, EPC8, and EPC16 devices is 4, 8, or 16 Mbits, respectively, with the boot block at the bottom. The Flash memory is divided into three types of blocks: the boot block, parameter block, and main block. Each block has protection capability and can be erased individually. The enhanced configuration device can also program and erase the Flash lock bits through the JTAG interface. The lock bits protect Flash against an inadvertent erase; a block cannot be erased when the lock bit is set.

Boot Block

The boot block, which is 8K words on the EPC16 configuration device, can replace a dedicated boot PROM for a microprocessor (as found in Excalibur™ embedded processor solutions). It can also store other system data, but not configuration data. The boot block features hardware-controllable write protection to protect the crucial microprocessor boot code from accidental modification using a combination of RP# and WP# pins and a block lock bit. Each block contains a lock bit that disables a program or an erase operation on the block.

To use the bottom boot block in the Flash memory, WP# should be connected to VCC. If WP# is connected to GND, the bottom boot block is locked so it cannot be programmed or erased. WP# only exists in the bottom boot blocks; the other blocks are not affected.

When using the Quartus II software versions 1.0 and 1.1, WP# should be connected to VCC, otherwise, the Quartus II software cannot successfully program the device.

Parameter Block

The parameter block is used for storing small, frequently updated parameters. In EPC16 devices, there are six parameter blocks of 4K words. Parameter block protection is controlled by using a combination of RP# and block lock bits.

Main Block

The main block fills the remainder of the Flash memory and contains configuration and user memory space. In EPC16 configuration devices, there are 31 blocks of 32K words. Similar to the parameter block, the protection of the main block is controlled using a combination of RP# and block lock bits.

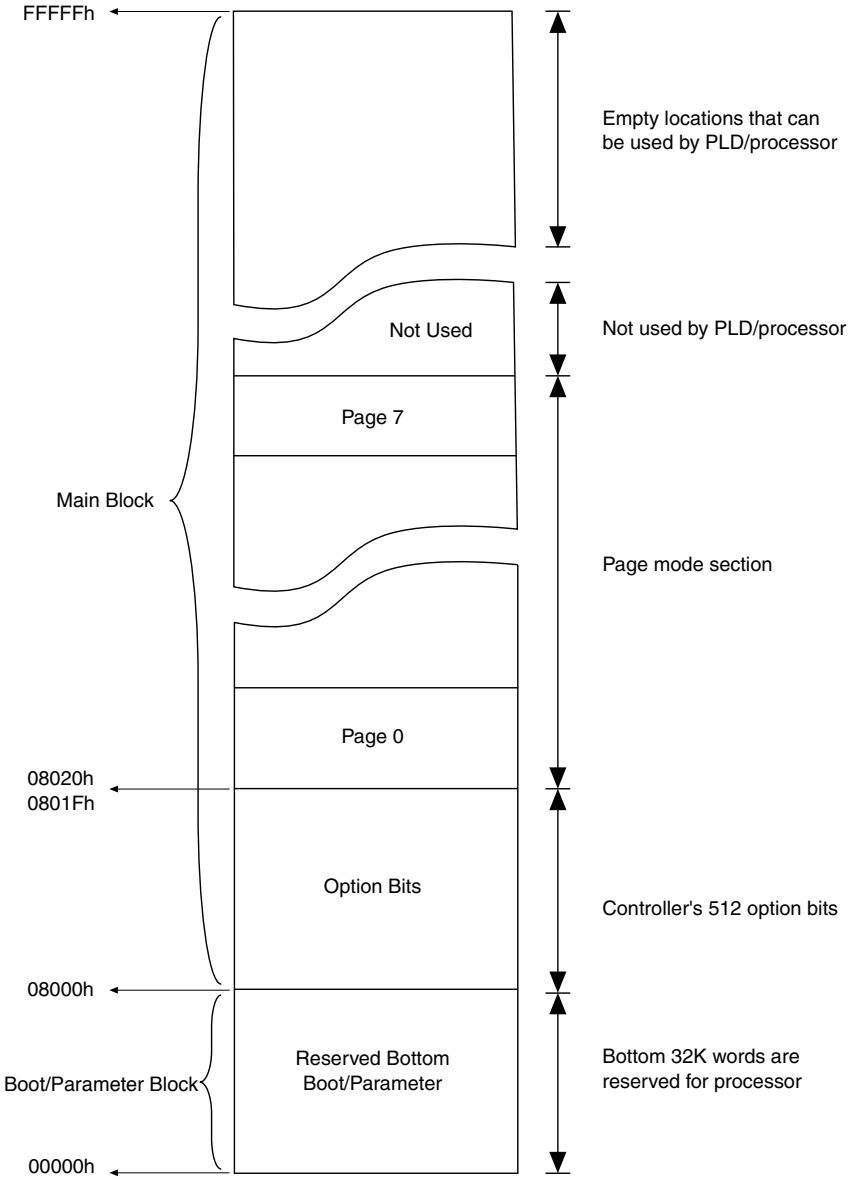
Memory Map

The EPC16 configuration device memory map can be divided into two main sections: controller memory space and user memory space.

The controller memory space consists of the controller's option bits and a maximum of eight pages of configuration data. The memory space starts with address 08000h (after 32K words of boot/parameter blocks) and continues upward. 512 bits reside from address 08000h to 00801fh, and they are reserved for option bits.

Figure 5 shows the 16-Mbit Flash memory map for EPC16 devices.

Figure 5. Flash Memory in EPC16 Configuration Device (Bottom Boot)



In EPC16 configuration devices, the boot blocks and the parameter blocks are located at the bottom of the 32K-word blocks. Due to the lock bit flexibility available with 4K blocks and the WP# feature in the two 4K boot blocks, the user may want to use the bottom 32K-word block (boot and parameter blocks) for PLD or processor memory space. Altera recommends using the bottom 32K-word blocks (boot/parameter blocks). However, if the PLD or processor will boot from the top 32K-word blocks, the user should re-map the address to the bottom block by using glue logic. In systems that do not use PLD or processor memory space, the system user can use the bottom 32K-word blocks for configuration data memory space.

Page Mode Selection

The Page Mode Selection feature allows the enhanced configuration device to store up to eight different designs per PLD. The user chooses which design will configure the PLD at configuration. The Page Mode Selection will enable designers to switch the functionality of a PLD (or PLDs) by switching PGM[ ] pins.

Each page mode can have up to 8-bit concurrent configuration of devices.

Three input pins (PGM[2:0]) select one of eight pages of the configuration files that configure the PLDs. Page 0 is defined as the default page (see Figure 5 on page 10). Connect these pins on the board to select the user-specified page in the Quartus II software when generating the EPC4, EPC8, or EPC16 POF file. PGM2 is the most significant bit (MSB).

Operating Modes

The operating modes define the enhanced configuration device's process flow of data and control signals. The data process flow explains how data is transferred between blocks during read and write cycles. The control process flow explains how control signals handshake between the blocks to facilitate data transfer. The main modes are normal mode and programming mode.

Normal Mode

The Normal mode controls the PLD configuration process using compressed data in the Flash memory. The process involves reading data from the Flash memory, data decompression, and sending data to the PLD.

Upon power-up, the POR unit generates all the reset signals. The POR unit resets all enhanced configuration device's control units using the 10-MHz default internal clock as the main clock source. After the POR counter expires, the POR unit de-asserts OE. By holding OE low, the POR time can be extended. Upon start of the configuration process, the device samples the PGM[] select pins to determine which page of the configuration files in Flash memory should be used for PLD configuration. The CDU will switch the internal clock to the new clock settings according to the option bit setting. The device starts to read the Flash configuration data. When OE goes high, the PCU starts the DCLK and configures the PLD.

When the last configuration data bits have been read from the Flash memory, the page counter expires and the PCU stops reading from the Flash memory. If no error is detected on CONF_DONE, DCLK will continue toggling until nCS goes high, indicating a successful PLD configuration cycle. If CONF_DONE error detection detects an error, the POR unit will assert OE and start a new PLD reconfiguration.

After the PLD configuration process is complete, PCU stops DCLK. To keep the Flash memory in an idle state, the device enables pull-ups, pull-downs, and/or bus-keepers to the Flash interface pins.

Programming Mode

During ISP mode, the JTAG interface accesses the Flash memory. The controller processes the ISP instructions to access the Flash memory through the JIU. After receiving an ISP instruction, the JIU decodes the instruction and performs the necessary Flash bus cycle. At the end of the programming mode, the JIU interfaces with the PLD to initiate a PLD reconfiguration cycle. When the JTAG interface takes the bus-mastership, it starts to reconfigure the PLD. During PLD configuration, the JTAG interface should not be used, as using it may interfere with the PLD configuration. After the re-configuration cycle is successfully completed, the PLD asserts CONF_DONE high. Upon this assertion, DCLK drives low, and DATA[7:0] remains in the last logic state.

Device Configuration

The control signals from the enhanced configuration device (DATA[], DCLK, nCS, nINIT_CONF, and OE) interface directly with the APEX II, APEX 20K, Mercury, ACEX, or FLEX 10K devices' control signals.

For more information on parallel configuration, refer to *Application Note 116 (Configuring SRAM-Based LUT Devices)*.

The DCLK pin, which is driven from the enhanced configuration device to the PLD, acts as the configuration cycle reference clock. It functions as the configuration data "write-enable" strobe signal. The OE pin is an open-drain pin and is driven low when POR is not complete. A built-in 2-ms or 100-ms counter holds the release of OE during the initial power-up to permit voltage level stabilization. After POR expires, the POR time can be extended externally by driving OE low. When OE is driven low, the enhanced configuration device resets the address counters.

The nCS pin of the enhanced configuration device is connected to the CONF_DONE pin of the PLD. The nCS pin checks for a successful PLD configuration after the last configuration data has been transmitted to the PLD. The PLD always drives the nCS low when the OE is pulled low. Both the nCS and OE pins have a programmable weak internal pull-up resistor.

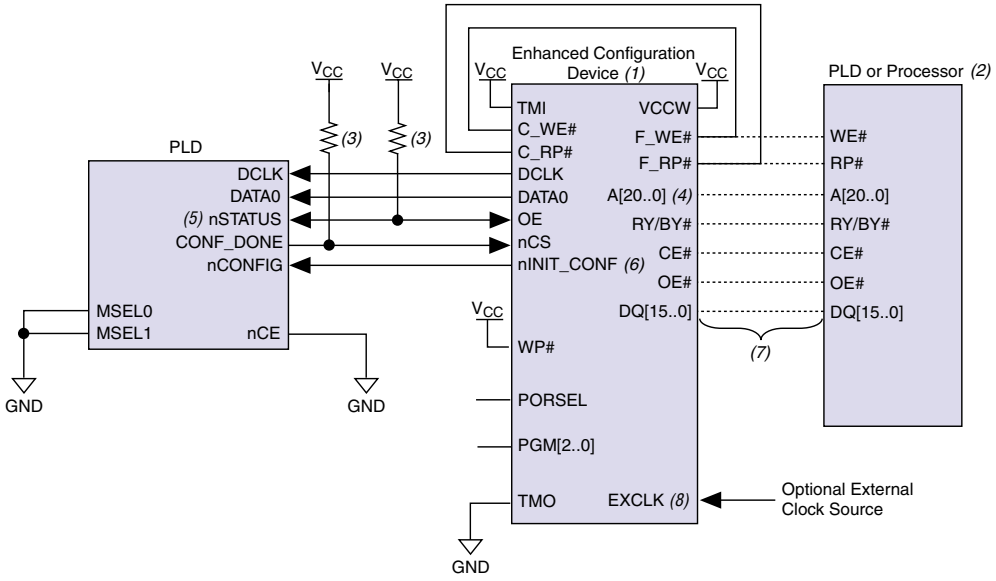
The enhanced configuration device allows the user to initiate configuration of APEX II, APEX 20K, Mercury, ACEX, or FLEX 10K devices via the nINIT_CONF pin, which can be tied to the nCONFIG pin of LUT-based PLDs. A JTAG instruction causes the enhanced configuration device to drive nINIT_CONF low, which, in turn, pulls nCONFIG low. The enhanced configuration device then drives nINIT_CONF high to start configuration. When the JTAG state machine exits this stage, nINIT_CONF releases the nCONFIG, and the PLD configuration is initiated.

Serial Configuration Mode

APEX II, APEX 20K, Mercury, ACEX, and FLEX 10K devices can be configured through the enhanced configuration device in the serial programming mode. In this mode, the enhanced configuration device sends a serial bit-stream of configuration data to its DATA0 pin, which is routed to the DATA0 input pin on LUT-based PLDs.

Figure 6 shows APEX II, APEX 20K, Mercury, ACEX, or FLEX 10K devices configured with an enhanced configuration device in the serial programming mode.

Figure 6. Serial Configuration Mode



***Notes to** Figure 6:*

(1) If the direct Flash interface is not used in an enhanced configuration device, then the Flash pins should be left unconnected because they are internally connected to the controller unit. The only pins that need external connection are WP#, WE#, and RP#, which are shown in Figure 13 on page 29. If Flash is being used as an external memory source (in EPC16 configuration devices), then the Flash pins should be connected. For more information, consult the LHF16J06 Data Sheet on the Altera web site (**http://www.altera.com**).

(2) For Flash interface in EPC4 and EPC8 configuration devices, please contact Altera Applications.

(3) The OE, nCS, and nINIT_CONF pins on enhanced configuration devices have internal pull-up resistors. The internal pull-up resistor on the nINIT_CONF pin is always active. However, on OE and nCS pins, the user has the option of turning these resistors on or off through the software.

(4) Pin A20 in EPC16 devices, pins A20 and A19 in EPC8 devices, and pins A20, A19, and A18 in EPC4 devices should be left floating. These pins should not be connected to any signal, i.e., they are no-connect pins.

(5) nSTATUS should not have an external pull-up resistor for ACEX or FLEX devices. Instead, the programmable internal resistor on OE should be used.

(6) nINIT_CONF has an internal pull-up resistor that is always active. If nINIT_CONF is not available or not used, nCONFIG must be pulled to V_{CC} either directly or through a 1-kΩ resistor. When configuring an APEX 20KE device, an external pull-up should not be used. For more information, refer to *Application Note 116 (Configuring SRAM-Based LUT Devices)*.

(7) The Flash interface exists only in EPC16 configuration devices and is a tri-state interface. The signals displayed as dotted lines should not be driven when the Flash interface is not available. For Flash interface availability, refer to Table 2 on page 15.

(8) EXCLK is an input only. In the Quartus II software, the user can select EXCLK or internal oscillator as the clock source.

In the enhanced configuration device, the Flash memory stores the configuration data, and the controller transfers the configuration data through the DATA0 pin to LUT-based PLDs. DATA0, DCLK, nCS, nINIT_CONF, and OE pins interface the enhanced configuration device to the PLD.

External Flash Memory Interface

In EPC16 configuration devices, the unused memory portion of the main block (i.e., memory that was not used for the configuration file) can be used by an external source such as a microprocessor or PLD. This external source uses the unused Flash memory to store application codes. The address, data, and control ports of the Flash memory are internally connected to the enhanced configuration device controller and to external device pins. An external source can drive these external device pins to access the Flash memory when the interface to Flash is available, (i.e., when the controller is not accessing Flash memory).

When the controller accesses Flash memory while configuring a PLD or programming an enhanced configuration device, the processor must tri-state the Flash interface pins to avoid contention. When the controller is not accessing Flash memory, the interface pins tri-state and allow the processor or PLD to access the Flash memory.

Flash memory access is available after successful configuration of the PLD, as indicated in Table 2, which lists the signals that indicate when the Flash memory is available.

Table 2. Enhanced Configuration Devices Interface Signals

nINIT_CONF/ nCONFIG	OE/ nSTATUS	nCS/ CONF_DONE	External Flash Interface	PLD State
0	X	X	Not available	PLD or enhanced configuration device is in power-on reset (POR) mode, or nCONFIG is asserted by external source (PLD or processor), or private JTAG initiates configuration instruction.
1	0	0	Not available	PLD or enhanced configuration device is in POR mode or has failed configuration.
1	1	0 (DCLK active)	Not available	Configuration is in process when DCLK is toggling.
1	1	0 (DCLK inactive)	Available	Enhanced configuration device is blank when CONF-DONE is low and DCLK is not toggling.
1	1	1	Available	PLD is configured.

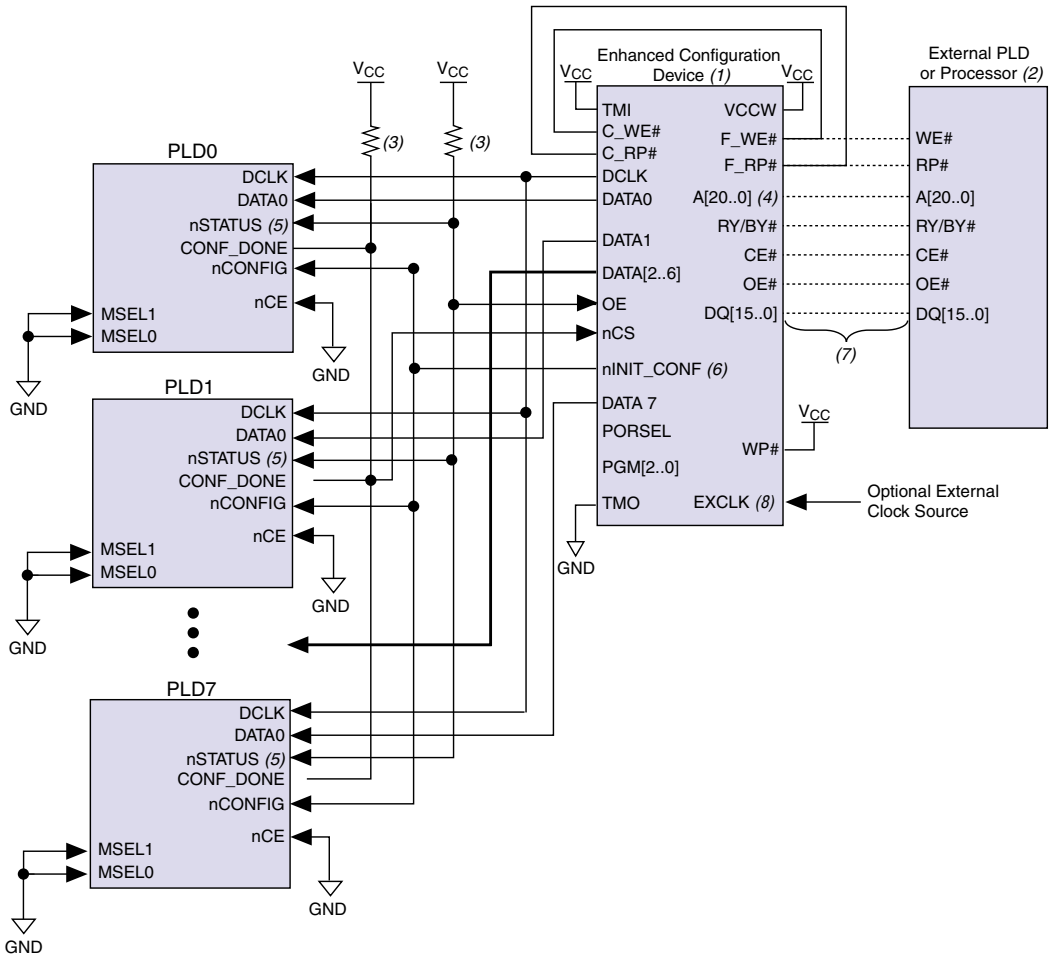
When using an external source (processor or PLD) to access Flash memory, the following considerations should be made:

- User cannot force the enhanced configuration device's controller to relinquish Flash access to the external source (processor or PLD). The external source (processor or PLD) must wait until configuration is complete, or when CONF_DONE goes high, before accessing the Flash memory.
- If the Auto_Restart configuration option is enabled and corrupted programming data is in the Flash memory, enhanced configuration devices will continuously try to configure the PLD. In such cases, the external source (processor or PLD) cannot access the Flash memory until a valid programming file is downloaded to the enhanced configuration device.
- The external source (processor or PLD) can cause the configuration process to restart by releasing control of the interface and then toggling nCONFIG.

Multiple Device Configuration in Serial Mode

The enhanced configuration device supports parallel configuration of multiple devices in serial configuration mode (see Figure 7). The enhanced configuration device can simultaneously output 1, 2, 4, or 8 parallel DATA outputs to multiple LUT-based PLDs. The user selects the configuration modes via the software.

Figure 7. Concurrent Configuration of Multiple Devices in Serial Mode (Different Data with N = 8)



Notes to Figure 7:

(1) If the direct Flash interface is not used in an enhanced configuration device, then the Flash pins should be left unconnected because they are internally connected to controller unit. The only pins that need external connection are WP#, WE#, and RP#, which are shown in Figure 13 on page 29. If Flash is being used as an external memory source (in EPC16 configuration devices), then the Flash pins should be connected. For more information, consult the LHF16J06 Data Sheet on the Altera web site (**http://www.altera.com**).

(2) For Flash interface in EPC4 and EPC8 configuration devices, please contact Altera Applications.

(3) The OE, nCS, and nINIT_CONF pins on enhanced configuration devices have internal pull-up resistors. The internal pull-up resistor on the nINIT_CONF pin is always active. However, on OE and nCS pins, the user has the option of turning these resistors on or off through the software.

(4) Pin A20 in EPC16 devices, pins A20 and A19 in EPC8 devices, and pins A20, A19, and A18 in EPC4 devices should be left floating. These pins should not be connected to any signal, i.e., they are no-connect pins.

(5) nSTATUS should not have an external pull-up resistor for ACEX or FLEX devices. Instead, the programmable internal resistor on OE should be used.

(6) nINIT_CONF has an internal pull-up resistor which is always active. If nINIT_CONF is not available or not used, an external pull-up should not be used. When configuring an APEX 20KE device, an external pull-up should not be used. For more information, refer to *Application Note 116 (Configuring SRAM-Based LUT Devices)*.
(7) The Flash interface exists only in EPC16 configuration devices and is a tri-state interface. The signals displayed as dotted lines should not be driven when the Flash interface is not available. For Flash interface availability, refer to Table 2 on page 15.
(8) EXCLK is an input only. In the Quartus II software, the user can select the EXCLK or the internal oscillator as the clock source.

Table 3 summarizes the configuration modes in the enhanced configuration device.

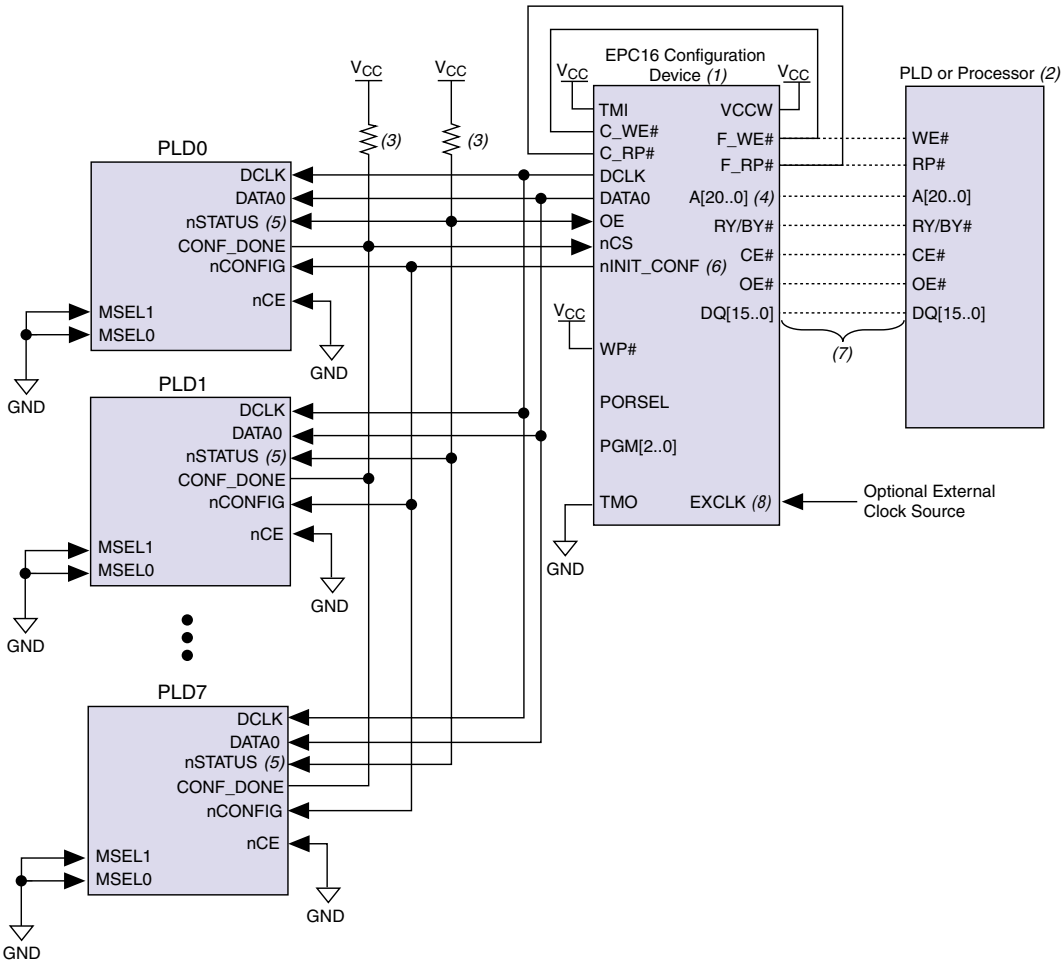
Table 3. Enhanced Configuration Device Mode

Mode Name	Mode (1)	Used Outputs	Unused Outputs
Passive Serial Mode	1	DATA0	DATA[7..1] drive out
Passive Multi-Device Parallel Synchronous Mode	2	DATA[1..0]	DATA[7..2] drive out
Passive Multi-Device Parallel Synchronous Mode	4	DATA[3..0]	DATA[7..4] drive out
Passive Multi-Device Parallel Synchronous Mode	8	DATA[7..0]	–

***Note to** Table 3:*
(1) The mode category gives the number of valid DATA outputs at each configuration mode.

Figure 8 shows parallel configuration of multiple devices in serial mode with the same DATA.

Figure 8. Concurrent Configuration of Multiple Devices in Serial Mode (Same Data with N = 1)



***Notes to** Figure 8:*

(1) If the direct Flash interface is not used in an enhanced configuration device, then the Flash pins should be left unconnected because they are internally connected to controller unit. The only pins that need external connection are WP#, WE#, and RP#, which are shown in Figure 13 on page 29. If Flash is being used as an external memory source (in external EPC16 configuration devices), then the Flash pins should be connected. For more information, consult the LHF16J06 Data Sheet on the Altera web site (**http://www.altera.com**).

(2) For Flash interface in EPC4 and EPC8 configuration devices, please contact Altera Applications.

(3) The OE, nCS, and nINIT_CONF pins on enhanced configuration devices have internal pull-up resistors. The internal pull-up resistor on the nINIT_CONF pin is always active. However, on OE and nCS pins, the user has the option of turning these resistors on or off through the software.

(4) Pin A20 in EPC16 devices, pins A20 and A19 in EPC8 devices, and pins A20, A19, and A18 in EPC4 devices should be left floating. These pins should not be connected to any signal, i.e., they are no-connect pins.

(5) nSTATUS should not have an external pull-up resistor for ACEX or FLEX devices. Instead, the programmable internal resistor on OE should be used.

(6) nINIT_CONF has an internal pull-up resistor that is always active. If nINIT_CONF is not available or not used, nCONFIG must be pulled to V_{CC} either directly or through a 1-kΩ resistor. When configuring an APEX 20KE device, an external pull-up should not be used. For more information, refer to *Application Note 116 (Configuring SRAM-Based LUT Devices)*.

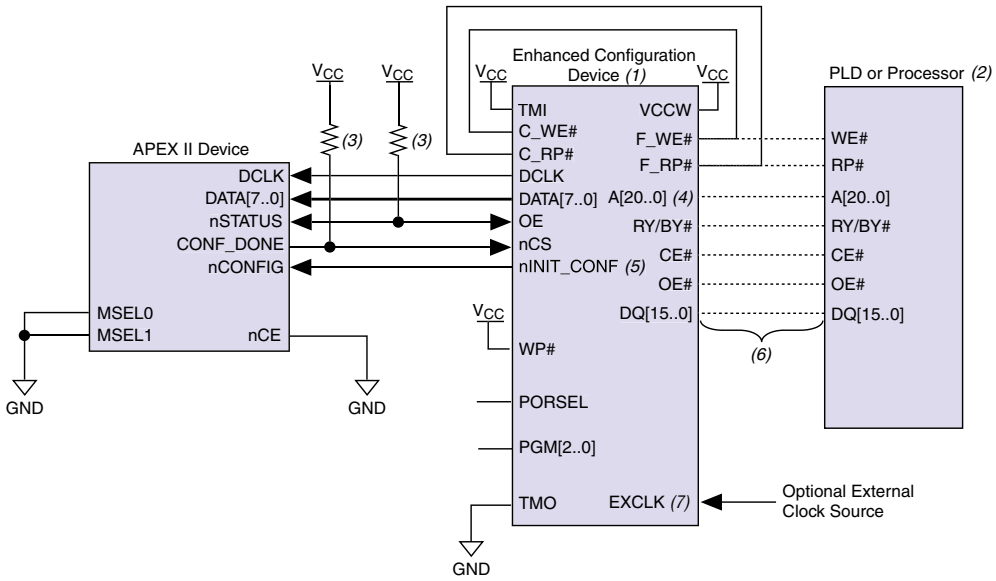
(7) The Flash interface exists only in EPC16 configuration devices and is a tri-state interface. The signals displayed as dotted lines should not be driven when the Flash interface is not available. For Flash interface availability, refer to Table 2 on page 15.

(8) EXCLK is an input only. In the Quartus II software, the user can select the EXCLK or the internal oscillator as the clock source.

Parallel Configuration Mode

APEX II devices can be configured through enhanced configuration devices in the fast parallel configuration mode. In this mode, the enhanced configuration device sends a byte of data to the DATA[7..0] pins, which route to the DATA[7..0] input pins in the APEX II device. APEX II devices receive byte-wide configuration data per each clock cycle. Figure 9 shows the enhanced configuration device parallel configuration mode.

Figure 9. Parallel Configuration Mode



Notes to Figure 9:

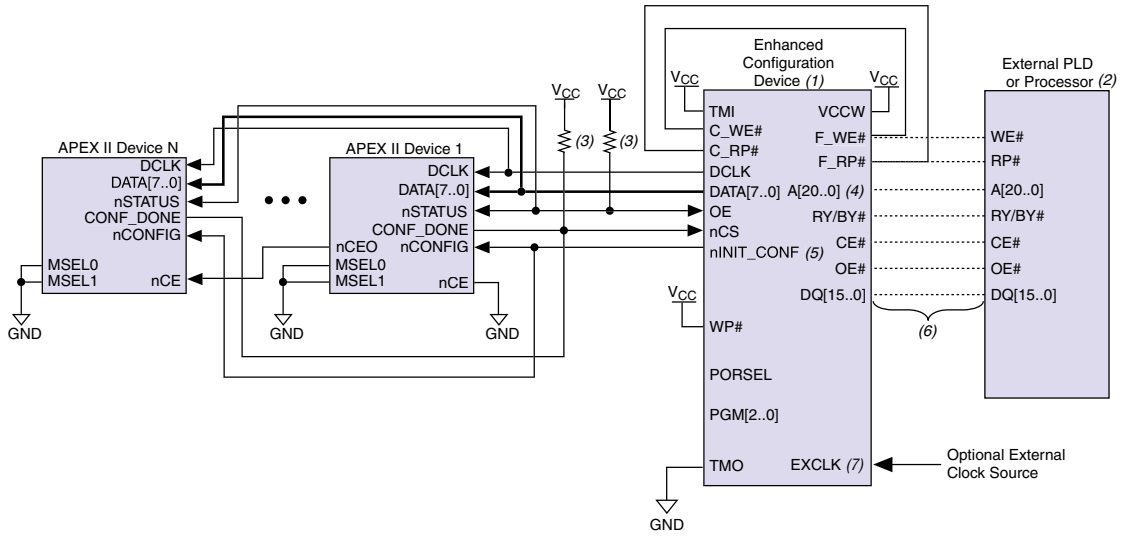
(1) If the direct Flash interface is not used in an enhanced configuration device, then the Flash pins should be left unconnected because they are internally connected to the controller unit. The only pins that need external connection are WP#, WE#, and RP#, which are shown in Figure 13 on page 29. If Flash is being used as an external memory source (in EPC16 configuration devices), then the Flash pins should be connected. For more information, consult the LHF16J06 Data Sheet on the Altera web site (**http://www.altera.com**).

(2) For Flash interface in EPC4 and EPC8 configuration devices, please contact Altera Applications.
(3) The OE, nCS, and nINIT_CONF pins on enhanced configuration devices have internal pull-up resistors. The internal pull-up resistor on the nINIT_CONF pin is always active. However, on OE and nCS pins, the user has the option of turning these resistors on or off through the software. All pull-up resistors are 1-kΩ.
(4) Pin A20 in EPC16 devices, pins A20 and A19 in EPC8 devices, and pins A20, A19, and A18 in EPC4 devices should be left floating. These pins should not be connected to any signal, i.e., they are no-connect pins.
(5) nINIT_CONF has an internal pull-up resistor that is always active. If nINIT_CONF is not available or not used, nCONFIG must be pulled to V_{CC} either directly or through a 1-kΩ resistor. For more information, refer to *Application Note 116 (Configuring SRAM-Based LUT Devices)*.
(6) The Flash interface exists only in EPC16 configuration devices and is a tri-state interface. The signals displayed as dotted lines should not be driven when the Flash interface is not available. For Flash interface availability, refer to Table 2 on page 15.
(7) EXCLK is an input only. In the Quartus II software, the user can select EXCLK or internal oscillator as the clock source.

Figure 10 shows a diagram of multiple APEX II device configuration with an enhanced configuration device in parallel programming mode. In this mode, multiple APEX II devices are cascaded together. After the first APEX II device completes configuration, its nCEO pin activates the second APEX II device's nCE pin. This pin activation prompts the second device to start configuration. (See Figure 10.)

Because CONF_DONE pins are tied together, all devices initialize and simultaneously enter user mode. If the enhanced configuration device detects an error, the configuration stops for the whole chain because nSTATUS pins are tied together.

Figure 10. Parallel Configuration of Multiple Devices in a Chain



***Notes to** Figure 10:*

(1) If the direct Flash interface is not used in an enhanced configuration device, then the Flash pins should be left unconnected because they are internally connected to the controller unit. The only pins that need external connection are WP#, WE#, and RP#, which are shown in Figure 13 on page 29. If Flash is being used as an external memory source (in EPC16 configuration devices), then the Flash pins should be connected. For more information, consult the LHF16J06 Data Sheet on the Altera web site (**http://www.altera.com**).

(2) For Flash interface in EPC4 and EPC8 configuration devices, please contact Altera Applications.

(3) The OE, nCS, and nINIT_CONF pins on enhanced configuration devices have internal pull-up resistors. The internal pull-up resistor on the nINIT_CONF pin is always active. However, on OE and nCS pins, the user has the option of turning these resistors on or off through the software. All pull-up resistors are 1-kΩ.

(4) Pin A20 in EPC16 devices, pins A20 and A19 in EPC8 devices, and pins A20, A19, and A18 in EPC4 devices should be left floating. These pins should not be connected to any signal, i.e., they are no-connect pins.

(5) nINIT_CONF has an internal pull-up resistor that is always active. If nINIT_CONF is not available or not used, nCONFIG must be pulled to V_{CC} either directly or through a 1-kΩ resistor. For more information, refer to *Application Note 116 (Configuring SRAM-Based LUT Devices)*.

(6) The Flash interface exists only in EPC16 configuration devices and is a tri-state interface. The signals displayed as dotted lines should not be driven when the Flash interface is not available. For Flash interface availability, refer to Table 2 on page 15.

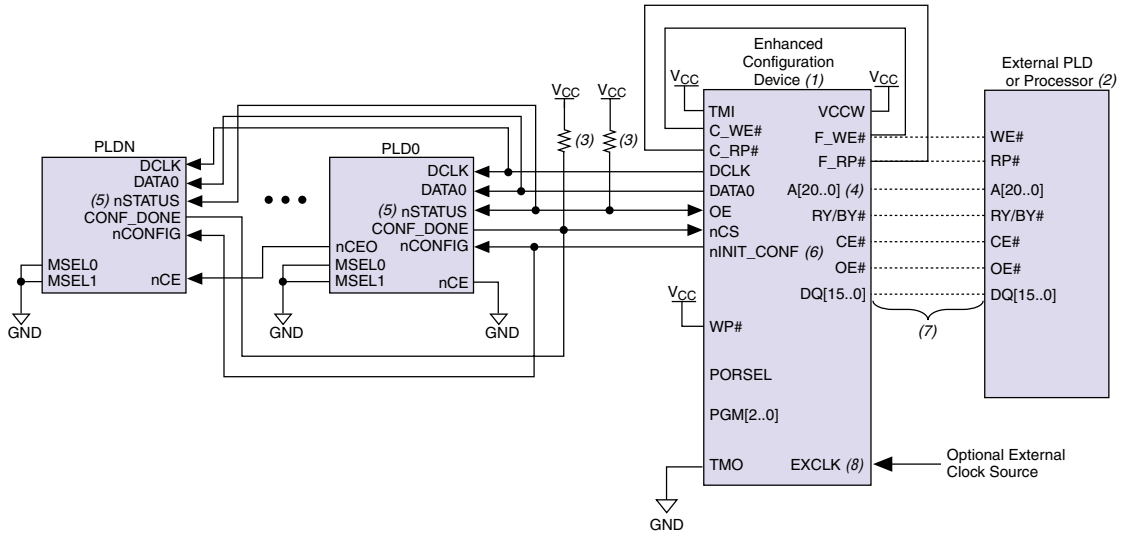
(7) EXCLK is an input only. In the Quartus II software, the user can select EXCLK or internal oscillator as the clock source.

Serial Configuration of Multiple Devices in a Chain

Because enhanced configuration devices contain a significant amount of Flash memory, the user does not need to cascade multiple enhanced configuration devices to configure large devices.

An enhanced configuration device can configure a chain of PLDs that are cascaded together. Figure 11 shows the enhanced configuration device configuring a chain of multiple PLDs in serial mode.

Figure 11. Serial Configuration of Multiple Devices in a Chain



Notes to Figure 11:

(1) If the direct Flash interface is not used in an enhanced configuration device, then the Flash pins should be left unconnected because they are internally connected to controller unit. The only pins that need external connection are WP#, WE#, and RP#, which are shown in Figure 13 on page 29. If Flash is being used as an external memory source (in external EPC16 configuration devices), then the Flash pins should be connected. For more information, consult the LHF16J06 Data Sheet on the Altera web site (**http://www.altera.com**).

(2) For Flash interface in EPC4 and EPC8 configuration devices, please contact Altera Applications.

(3) The OE, nCS, and nINIT_CONF pins on enhanced configuration devices have internal pull-up resistors. The internal pull-up resistor on the nINIT_CONF pin is always active. However, on OE and nCS pins, the user has the option of turning these resistors on or off through the software.

(4) Pin A20 in EPC16 devices, pins A20 and A19 in EPC8 devices, and pins A20, A19, and A18 in EPC4 devices should be left floating. These pins should not be connected to any signal, i.e., they are no-connect pins.

(5) nSTATUS should not have an external pull-up resistor for ACEX or FLEX devices. Instead, the programmable internal resistor on OE should be used.

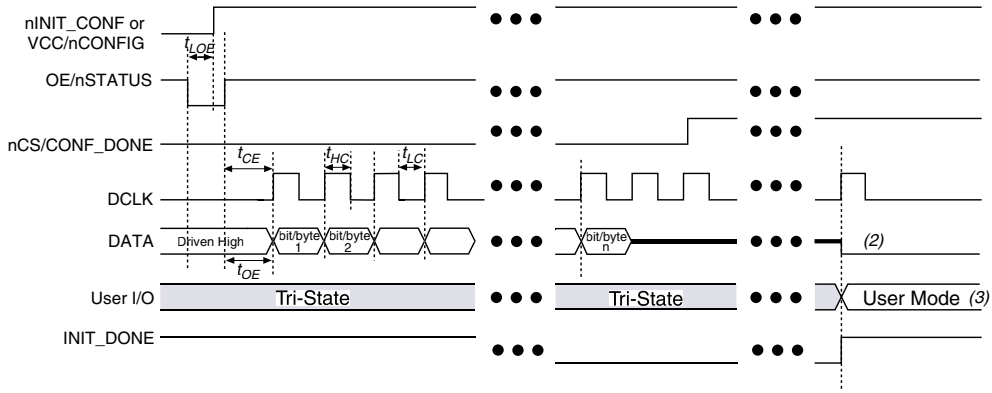
(6) nINIT_CONF has an internal pull-up resistor that is always active. If nINIT_CONF is not available or not used, nCONFIG must be pulled to V_{CC} either directly or through a 1-kΩ resistor. When configuring an APEX 20KE device, an external pull-up should not be used. For more information, refer to *Application Note 116 (Configuring SRAM-Based LUT Devices)*.

(7) The Flash interface exists only in EPC16 configuration devices and is a tri-state interface. The signals displayed as dotted lines should not be driven when the Flash interface is not available. For Flash interface availability, refer to Table 2 on page 15.

(8) EXCLK is an input only. In the Quartus II software, the user can select the EXCLK or the internal oscillator as the clock source.

Figure 12 shows the timing waveform for the enhanced configuration device scheme.

Figure 12. Enhanced Configuration Device Scheme Timing Waveform *Note (1)*



Notes to Figure 12:

(1) For timing information, refer to the Table 4 on page 25.
(2) The configuration device will drive DATA low after configuration.
(3) APEX II and APEX 20K devices enter user mode 40 clock cycles after CONF_DONE goes high. Mercury devices enter user mode 136 clock cycles after CONF_DONE goes high. ACEX 1K, FLEX 10K, and FLEX 6000 devices enter user mode 10 clock cycles after CONF_DONE goes high.

Table 4 defines the enhanced configuration device timing parameters when using enhanced configuration devices at 3.3 V.

Table 4. Enhanced Configuration Device Configuration Parameters (PLD Interface)

Symbol	Parameter	Condition	Min	Typ	Max	Unit
f_{DCLK}	DCLK frequency	40% duty cycle			66.7	MHz
t_{DCLK}	DCLK period		15			ns
t_{HC}	DCLK duty cycle high time	40% duty cycle	6			ns
t_{LC}	DCLK duty cycle low time	40% duty cycle	6			ns
t_{CE}	OE to first DCLK delay		40			µs
t_{OE}	OE to first DATA available		40			µs
t_{CAC}	DCLK rising edge to DATA change			0	(2)	ns
t_{CF} (3)	OE assert to DCLK disable delay		277			ns
t_{DF} (3)	OE assert to DATA disable delay		277			ns
t_{OH}	DATA hold time from last DCLK rising edge		(2)	0		ns
t_{RE} (1)	DCLK rising edge to OE		60			ns
t_{LOE}	OE assert time to assure reset		60			ns
f_{ECLK}	EXCLK input frequency	40% duty cycle			133.3	MHz
t_{ECLK}	EXCLK input period		7.5			ns
t_{ECLKH}	EXCLK input duty cycle high time	40% duty cycle	3.375			ns
t_{ECLKL}	EXCLK input duty cycle low time	40% duty cycle	3.375			ns
t_{ECLKR}	EXCLK input rise time	133 MHz			3	ns
t_{ECLKF}	EXCLK input fall time	133 MHz			3	ns
t_{POR} (4)	POR time	2 ms	1	2	3	ms
		100 ms	50	100	120	ms

Notes to Table 4:

(1) This parameter is used for CONF_DONE error detection by the enhanced configuration device.
(2) Contact Altera Applications for detailed information.
(3) This parameter is used in cyclic redundancy check (CRC) error detection by CPLD.
(4) The VCC ramp time should be less than 1 ms for 2-ms POR, and it should be less than 100 ms for 100-ms POR.

Power Sequencing

Altera recommends that you power-up the PLD before the enhanced configuration device's POR expires. The pin-selectable POR time feature is useful for ensuring this power-up sequence. The enhanced configuration device has two POR settings, 2 ms and 100 ms. For more margin, the 100-ms setting can be selected to allow the PLD to power-up before configuration is attempted.

Enhanced Configuration Device Pin-Outs

Tables 5 through 7 describe pin definitions for the enhanced configuration device. These tables include PLD interface pins, Flash interface pins, JTAG interface pins, and other pins.

Table 5. PLD Interface Pins with Respect to Controller

Pin Name	Pin Type	Description
DATA[7..0]	Output	This is the PLD configuration output data bus. DATA changes on each rising edge of DCLK.
DCLK	Output	The DCLK pin is always an output. The enhanced configuration device drives the DCLK signal to the PLD as the configuration clock.
nCS	Input	The nCS pin is an input to the enhanced configuration device and is used to input the PLD's CONF_DONE signal for error detection after the last configuration data is transmitted to the PLD. The PLD will always drive nCS low when OE is asserted. This pin contains a programmable internal weak pull-up.
nINIT_CONF	Output	The nINIT_CONF pin can be connected to the nCONFIG pin on LUT-based PLDs to initiate configuration for the enhanced configuration device via a private JTAG instruction. This pin contains a programmable internal weak pull-up.
OE	Open-Drain I/O	This pin is driven low when POR is not complete. A user-selectable 2-ms or 100-ms counter holds off the release of OE during initial power up to permit voltage levels to stabilize. POR time can be extended externally by driving OE low. After the enhanced configuration device controller releases OE, it waits for OE to go high before starting the PLD configuration process. This pin contains a programmable internal weak pull-up.

Table 6. Flash Interface Pins *Note (1)*

Pin Name	Pin Type	Description
A[20:0] *(2)*	Input	These pins are the address input to the Flash memory for read and write operations. The addresses are internally latched during a write cycle.
DQ[15:0]	Input/Output	These pins are a Data bus that interface with the Flash memory and the controller. The controller or an external source drives DQ[15:0] during the Flash command and the data write bus cycles. During the data read cycle, Flash memory drives the DQ[15:0] to the controller.
CE# *(3)*	Input	When asserted, it activates the Flash memory. When it is high, it deselects the device and reduces power consumption to standby levels.
RP# *(3)*, *(4)*	Input	When asserted, it resets the Flash memory. When high, it enables normal operation. When low, it inhibits write operation in the Flash memory, which provides data protection during power transitions.
OE# *(3)*	Input	The controller asserts this pin during Flash read cycles. When asserted, it enables the drivers of Flash output pins.
WE# *(3)*, *(5)*	Input	The controller asserts WE# during Flash write cycle. When asserted, it controls writes to the Flash memory. In the Flash memory, addresses and data are latched on the rising edge of the WE# pulse.
WP# *(3)*, *(5)*	Input	This pin is usually tied to VCC or ground on the board. The controller does not drive this pin because it could cause contention.
VCCW	Supply	Block erase, full chip erase, word write, or lock bit configuration power supply.
RY/BY# *(3)*	Output	Flash asserts this pin when a write or erase operation is complete. This is a Flash only pin.

Notes to** Table 6**:

(1) If the direct Flash interface is not used in an enhanced configuration device, then the Flash pins should be left unconnected because they are internally connected to controller unit. The only pins that need external connection are WP#, WE#, and RP#, which are shown in Figure 13 on page 29. If Flash is being used as an external memory source (in EPC16 configuration devices), then the Flash pins should be connected. For more information, consult the LHF16J06 Data Sheet on the Altera web site (**http://www.altera.com**).

(2) Pin A20 in EPC16 devices, pins A20 and A19 in EPC8 devices, and pins A20, A19, and A18 in EPC4 devices are floating. These pins should not be connected to any signal, i.e., they are no-connect pins.

(3) The # symbol means active low.

(4) These pins can be driven to 12 V during Flash testing. Because the controller cannot tolerate the 12-V level, connection on these pins from the controller to the Flash will not be bonded internally in the package and they will be available as two separate pins. The user is required to connect the two pins at the board level (for example, on the PCB, connect the WE# pin from controller to WE# pin from the Flash memory, as shown in Figure 13 on page 29).

(5) WP# should be connected to VCC on the board when using the Quartus II software versions 1.0 and 1.1.

Table 7. JTAG Interface Pins and Other Pins with Respect to Controller

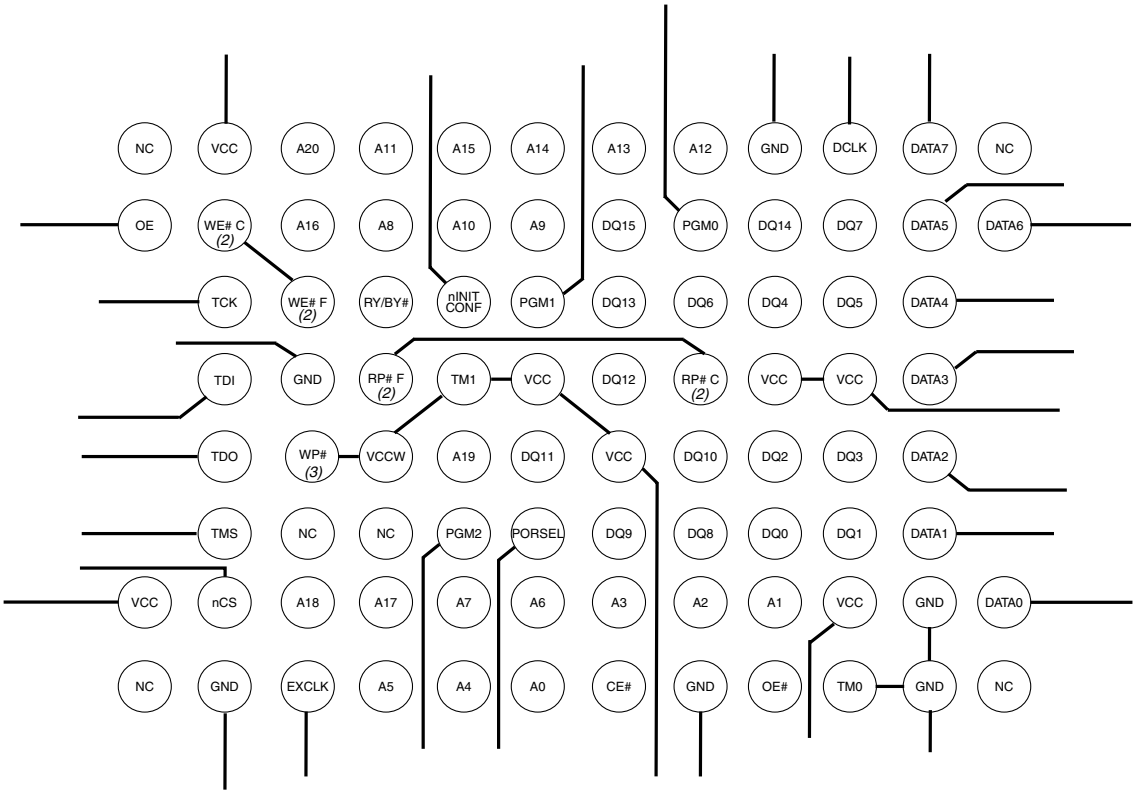
Pin Name	Pin Type	Description
TDI	Input	This is a JTAG data input pin. Connect this pin to VCC if the JTAG circuitry is not used.
TDO	Output	This is a JTAG data output pin. Do not connect this pin if the JTAG circuitry is not used.
TCK	Input	This is a JTAG clock pin. Connect this pin to ground if the JTAG circuitry is not used.
TMS	Input	This is a JTAG mode select pin. Connect this pin to VCC if the JTAG circuitry is not used.
PGM[2..0]	Input	These three input pins select one of the eight pages of the configuration files to configure the PLD. Connect these pins on the board to select the page specified by the designer in the Quartus II software when generating the enhanced configuration device POF file. PGM[2] is the MSB.
EXCLK	Input	During Normal mode, the EXCLK pin operates as the external clock source.
PORSEL	Input	This pin selects a 2-ms or 100-ms POR counter delay during power up. When PORSEL is Low, POR time is 100 ms. When PORSEL is High, POR time is 2 ms.
TMO	Input	Test mode pin selects different test modes. In operating mode, this pin should be connected to GND
TMI	Input	Test mode pin selects different test modes. In operating mode, this pin should be connected to VCC

Package

The EPC16 configuration device is available in both the 88-pin Ultra FineLine BGA package and the 100-pin PQFP package. The Ultra FineLine BGA package, which is based on 0.8-mm pitch, maximizes board space efficiency. A board can be laid out for this package using only one PCB layer. The EPC8 and EPC4 devices are available in the 100-pin PQFP package.

Figure 13 shows the PCB routing for the 88-pin Ultra FineLine BGA package. The Gerber file for this layout is on the Altera web site at **http://www.altera.com**.

Figure 13. PCB Routing for 88-Pin Ultra FineLine BGA Package *Note (1)*



Notes to** Figure 13**:

(1) If the direct Flash interface is not used in an enhanced configuration device, then the Flash pins should be left unconnected because they are internally connected to controller unit. The only pins that need external connection are WP#, WE#, and RP#. If Flash is being used as an external memory source (in EPC16 configuration devices), then the Flash pins should be connected. For more information, consult the LHF16J06 Data Sheet on the Altera web site (**http://www.altera.com**).

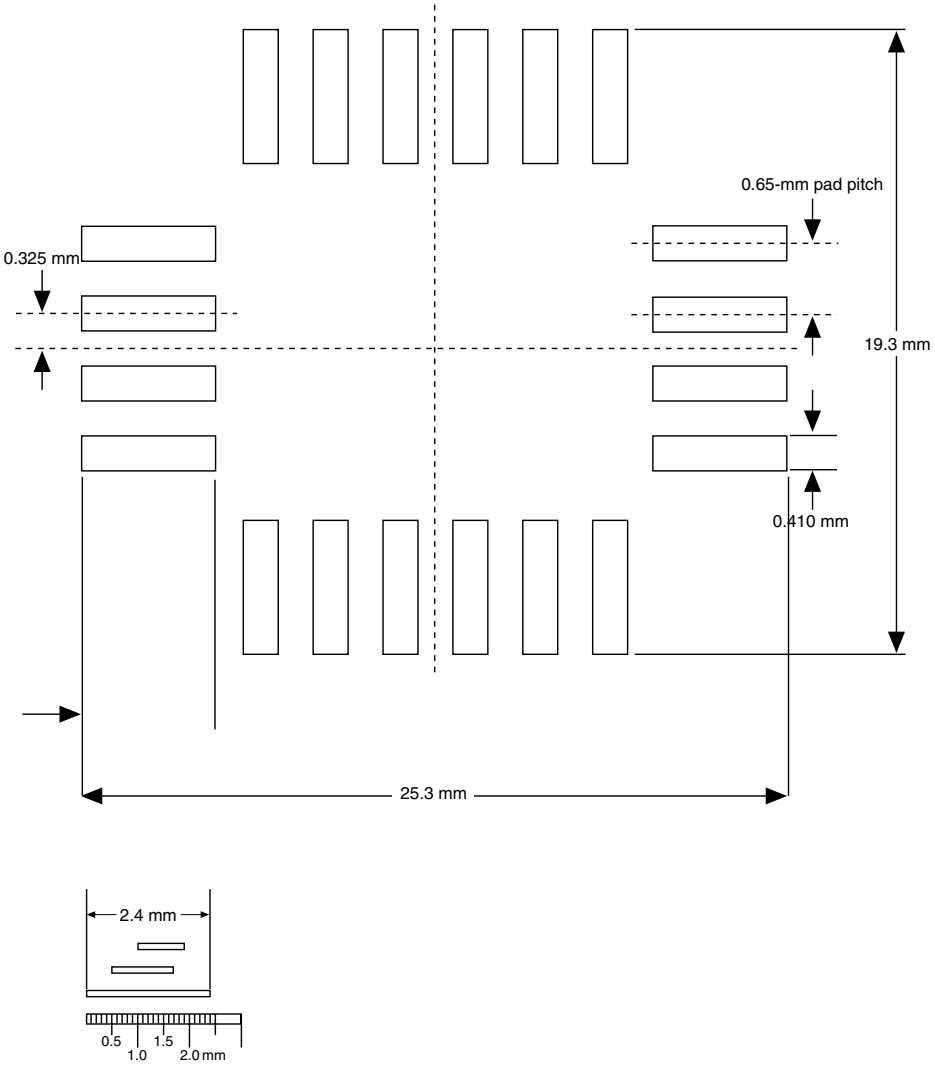
(2) RP#F and WE#F are pins on the Flash die. RP#C and WE#C are pins on the controller die. WE#C and WE#F should be connected together on the PCB. RP#F and RP#C should also be connected together on the PCB.

(3) WP# should be connected to 3.3 V to be able to program the bottom boot block, which is required when programming the device from the Quartus II software.

Package Layout Recommendation

Enhanced configuration devices in 100-pin PQFP packages have different package dimension than other 100-pin PQFP devices. Figure 14 shows the 100-pin PQFP PCB footprint specifications for enhanced configuration devices. These footprint dimensions are based on vendor-supplied package outline diagrams.

Figure 14. Enhanced Configuration Device PCB Footprint Specifications for 100-Pin PQFP Packages
Notes (1), (2), (3)



Notes to Figure 14:
(1) Used 0.5-mm increase for front and back of nominal foot length
(2) Used 0.3-mm increase to maximum foot width.
(3) Diagrams are based on vendor-supplied drawings.

ISP Programming & Configuration File Support

The Quartus II development software provides programming support for the enhanced configuration device and automatically generates the programming files for the EPC4, EPC8, and EPC16 configuration devices. In a multi-device project, the software can combine the programming files for multiple APEX II, APEX 20K, Mercury, ACEX, or FLEX 10K devices into one EPC4, EPC8, or EPC16 configuration device.

Enhanced configuration devices can be programmed in-system through its industry-standard 4-pin JTAG interface. ISP in the enhanced configuration device provides ease in prototyping and updating APEX II, APEX 20K, Mercury, ACEX, or FLEX device functionality. Enhanced configuration devices can also be programmed by third-party Flash programmers.

After programming an enhanced configuration device in-system, LUT-based PLD configuration can be initiated by including the enhanced configuration device's JTAG `INIT_CONF` instruction. See Table 8.

The ISP circuitry in the enhanced configuration device is compliant with the IEEE Std. 1532 specification. The IEEE Std. 1532 is a standard that allows concurrent ISP between devices from multiple vendors.

Table 8. Enhanced Configuration Device JTAG Instructions

JTAG Instruction	Description
SAMPLE/PRELOAD	Allows a snapshot of the state of the enhanced configuration device pins to be captured and examined during normal device operation and permits an initial data pattern output at the device pins.
EXTEST	Allows the external circuitry and board-level interconnections to be tested by forcing a test pattern at the output pins and capturing results at the input pins.
BYPASS	Places the 1-bit bypass register between the TDI and the TDO pins, which allow the BST data to pass synchronously through a selected device to adjacent devices during normal device operation.
IDCODE *(1)*	Selects the device IDCODE register and places it between TDI and TDO, allowing the device IDCODE to be serially shifted out to TDO. The device IDCODE for the enhanced configuration device is shown below: 0100A0DDh
USERCODE	Selects the USERCODE register and places it between TDI and TDO, allowing the USERCODE to be serially shifted out the TDO. The 32-bit USERCODE is a programmable user-defined pattern.
INIT_CONF	This function initiates the PLD re-configuration process by tying nINIT_CONF to the PLD(s) nCONFIG pin(s). After this instruction is updated, the nINIT_CONF is released and starts the PLD configuration.
PENDCFG	This function asserts nINIT_CONF before accessing the Flash memory, if the external PLD/processor is connected to the Flash. This avoids a Flash bus contention when both JTAG/ISP and external PLD/processor want to access the Flash. Before JTAG/ISP can access the Flash, the external PLD/processor needs to be reset by asserting nINIT_CONF, which puts the external PLD/processor in a “reset” state and waits for the de-assertion of the INIT_CONF.

Note to** Table 8**:
(1) For enhanced configuration devices, instruction register length is 10 and boundary scan length is 174.

IEEE Std. 1149.1 (JTAG) Boundary-Scan Testing

The enhanced configuration device provides JTAG BST circuitry that complies with the IEEE Std. 1149.1-1990 specification. JTAG boundary-scan testing can be performed before or after configuration, but not during configuration. Table 9 shows the timing parameters and values for the enhanced configuration device.

Table 9. JTAG Timing Parameters & Values

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 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		25	ns
t_{JSZX}	Update register high-impedance to valid output		25	ns
t_{JSXZ}	Update register valid output to high impedance		25	ns

Operating Conditions

Tables 10 through 15 provide information on absolute maximum ratings, recommended operating conditions, DC operating conditions, supply current values, capacitance, and configuration parameters for the enhanced configuration device.

Table 10. Enhanced Configuration Device Absolute Maximum Rating

Symbol	Parameter	Condition	Min	Max	Unit
V_{CC}	Supply voltage	With respect to ground	–0.5	4.6	V
V_I	DC input voltage	With respect to ground	–0.5	3.6	V
I_{MAX}	DC V_{CC} or ground current			100	mA
I_{OUT}	DC output current, per pin		–25	25	mA
P_D	Power dissipation			360	mW
T_{STG}	Storage temperature	No bias	–65	150	˚ C
T_{AMB}	Ambient temperature	Under bias	–65	135	˚ C
T_J	Junction temperature	Under bias		135	˚ C

Table 11. Enhanced Configuration Device Recommended Operating Conditions

Symbol	Parameter	Condition	Min	Max	Unit
V_{CC}	Supplies voltage for 3.3-V operation		3.0	3.6	V
V_I	Input voltage	With respect to ground	–0.3	V_{CC} + 0.3	V
V_O	Output voltage		0	V_{CC}	V
T_A	Operating temperature	For commercial use	0	70	˚ C
		For industrial use	–40	85	˚ C
T_R	Input rise time			20	ns
T_F	Input fall time			20	ns

Table 12. Enhanced Configuration Device DC Operating Conditions

Symbol	Parameter	Condition	Min	Typ	Max	Unit
V_{CC}	Supplies voltage to core		3.0	3.3	3.6	V
V_{IH}	High-level input voltage		2.0		V_{CC} + 0.3	V
V_{IL}	Low-level input voltage				0.8	V
V_{OH}	3.3-V mode high-level TTL output voltage	I_{OH} = –4 mA	2.4			V
	3.3-V mode high-level CMOS output voltage	I_{OH} = –0.1 mA	V_{CC} – 0.2			V
V_{OL}	Low-level output voltage TTL	I_{OL} = –4 mA DC			0.45	V
	Low-level output voltage CMOS	I_{OL} = –0.1 mA DC			0.2	V
I_I	Input leakage current	$V_I = V_{CC}$ or ground	–10		10	µA
I_{OZ}	Tri-state output off-state current	$V_O = V_{CC}$ or ground	–10		10	µA
R_{CONF}	Configuration pins	Internal pull up (OE, nCS, nINIT, CONF)		6		kΩ

Table 13. Enhanced Configuration Device I_{CC} Supply Current Values

Symbol	Parameter	Condition	Min	Typ	Max	Unit
I_{CC0}	V_{CC} supply current (standby)			50	100	µA
I_{CC1}	V_{CC} supply current (during configuration)			50	100	mA

Table 14. Enhanced Configuration Device Capacitance

Symbol	Parameter	Condition	Min	Max	Unit
C_{IN}	Input pin capacitance			10	pF
C_{OUT}	Output pin capacitance			10	pF

Table 15. Enhanced Configuration Device Configuration Parameters (Flash Interface)

Symbol	Parameter	Condition	Min	Typ	Max	Unit
f_{SCLK}	SYSCLK frequency		–		10	MHz
t_{SCLK}	SYSCLK period		100		–	ns
t_{SCLKH}	SYSCLK duty cycle high time		50			ns
t_{SCLKL}	SYSCLK duty cycle low time		50			ns
$t_{AVQV(F)}$	Flash address to data DQ[15:0] delay		–		85	ns
$t_{GLQV(F)}$	Flash OE# to data DQ[15:0] delay		–		40	ns
$t_{WLWH(F)}$	Flash WE# pulse width		50		–	ns
$t_{WHR0(F)}$	Flash WE# high to SR7 ready		–		100	ns

Device Pin-Outs

See the Altera web site (**http://www.altera.com**) or the *Altera Digital Library* for pin-out information.

101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
http://www.altera.com
Applications Hotline:
(800) 800-EPLD
Customer Marketing:
(408) 544-7104
Literature Services:
lit_req@altera.com