4. Configuration Devices for SRAM-Based LUT Devices Data Sheet

CF52005-2.4

Features

This chapter describes configuration devices for SRAM-based LUT Devices.

- Configuration device family for configuring Altera® ACEX® 1K, APEX™ 20K (including APEX 20K, APEX 20KC, and APEX 20KE), APEX II, Arria® GX, Cyclone®, Cyclone II, FLEX® 10K (including FLEX 10KE and FLEX 10KA) Mercury™, Stratix®, Stratix GX, Stratix II, and Stratix II GX devices.
- Easy-to-use 4-pin interface to Altera FPGAs
- Low current during configuration and near-zero standby current
- 5.0-V and 3.3-V operation
- Programming support with the Altera Programming Unit (APU) and programming hardware from Data I/O, BP Microsystems, and other third-party programmers
- Available in compact plastic packages
  - 8-pin plastic dual in-line package (PDIP)
  - 20-pin plastic J-lead chip carrier (PLCC) package
  - 32-pin plastic thin quad flat pack (TQFP) package
- EPC2 device has reprogrammable Flash configuration memory
  - 5.0-V and 3.3-V in-system programmability (ISP) through the built-in IEEE Std. 1149.1 JTAG interface
  - Built-in JTAG boundary-scan test (BST) circuitry compliant with IEEE Std. 1149.1
  - Supports programming through Serial Vector Format Files (**.svf**), Jam™ Standard Test and Programming Language (STAPL) Files (**.jam**), Jam STAPL Byte-Code Files (**.jbc**), and the Quartus® II and MAX+PLUS® II softwares via the USB Blaster, MasterBlaster™, ByteBlaster™ II, EthernetBlaster, or ByteBlasterMV™ download cable
  - *n*INIT_CONF pin allows INIT_CONF JTAG instruction to begin FPGA configuration
  - Can be programmed with Programmer Object Files (**.pof**) for EPC1 and EPC1441 devices
  - Available in 20-pin PLCC and 32-pin TQFP packages

For detailed information about enhanced configuration devices, refer to *Enhanced Configuration Devices (EPC4, EPC8 and EPC16) Data Sheet*.

For detailed information about serial configuration devices, refer to *Serial Configuration Devices (EPCS1, EPCS4, EPCS16, EPCS64 and EPCS128) Data Sheet*.

Functional Description

With SRAM-based devices, configuration data must be reloaded each time the device powers up, the system initializes, or when new configuration data is needed. Altera configuration devices store configuration data for SRAM-based ACEX 1K, APEX 20K, APEX II, Arria GX, Cyclone, Cyclone II, FLEX 10K, FLEX 6000, Mercury, Stratix, Stratix GX, Stratix II, and Stratix II GX devices. Table 4–1 lists Altera configuration devices and their features.

Table 4–1. Altera Configuration Devices

Device	Memory Size (Bits)	ISP Support	Cascaded Support	Reprogrammable	Operating Voltage
EPC1	1,046,496	No	Yes	No	5.0 or 3.3 V
EPC2	1,695,680	Yes	Yes	Yes	5.0 or 3.3 V
EPC1064	65,536	No	No	No	5.0 V
EPC1064V	65,536	No	No	No	3.3 V
EPC1213	212,942	No	Yes	No	5.0 V
EPC1441	440,800	No	No	No	5.0 or 3.3 V

Table 4–2 lists the supported configuration devices required to configure the ACEX 1K, APEX 20K, APEX 20KC, APEX 20KE, APEX II, Cyclone, Cyclone II, FLEX 10K, FLEX 10KA, FLEX 10KE, FLEX 6000/A , FLEX 8000A , Mercury, Stratix, Stratix GX, or Stratix II device.

Table 4–2. Configuration Devices Required (Part 1 of 4)

Family	Device	Data Size (Bits) *(1)*	EPC1064/ 1064V	EPC1213	EPC1441	EPC1	EPC2
ACEX 1K	EP1K10	159,160	—	—	1	1	1
	EP1K30	473,720	—	—	—	1	1
	EP1K50	784,184	—	—	—	1	1
	EP1K100	1,335,720	—	—	—	—	1
APEX 20K	EP20K100	993,360	—	—	—	1	1
	EP20K200	1,950,800	—	—	—	—	2
	EP20K400	3,880,720	—	—	—	—	3
APEX 20KC	EP20K200C	1,968,016	—	—	—	—	2
	EP20K400C	3,909,776	—	—	—	—	3
	EP20K600C	5,673,936	—	—	—	—	4
	EP20K1000C	8,960,016	—	—	—	—	6

Table 4–2. Configuration Devices Required (Part 2 of 4)

Family	Device	Data Size (Bits) *(1)*	EPC1064/ 1064V	EPC1213	EPC1441	EPC1	EPC2
APEX 20KE	EP20K30E	354,832	—	—	1	1	1
	EP20K60E	648,016	—	—	—	1	1
	EP20K100E	1,008,016	—	—	—	1	1
	EP20K160E	1,524,016	—	—	—	—	1
	EP20K200E	1,968,016	—	—	—	—	2
	EP20K300E	2,741,616	—	—	—	—	2
	EP20K400E	3,909,776	—	—	—	—	3
	EP20K600E	5,673,936	—	—	—	—	4
	EP20K1000E	8,960,016	—	—	—	—	6
	EP20K1500E	12,042,256	—	—	—	—	8
APEX II	EP2A15	1,168,688	—	—	—	—	3
	EP2A25	1,646,544	—	—	—	—	4
	EP2A40	2,543,016	—	—	—	—	6
	EP2A70	4,483,064	—	—	—	—	11
Cyclone	EP1C3	627,376	—	—	—	1	1
	EP1C4	925,000	—	—	—	1	1
	EP1C6	1,167,216	—	—	—	1 *(2)*	1
	EP1C12	2,326,528	—	—	—	—	1 *(2)*
	EP1C20	3,559,608	—	—	—	—	2 *(2)*
Cyclone II	EP2C5	1,265,792	—	—	—	—	1
	EP2C8	1,983,536	—	—	—	—	2
	EP2C20	3,892,496	—	—	—	—	3
	EP2C35	6,848,608	—	—	—	—	5
	EP2C50	9,951,104	—	—	—	—	6
	EP2C70	14,319,216	—	—	—	—	9
FLEX 10K	EPF10K10	118,000	—	—	1	1	1
	EPF10K20	231,000	—	—	1	1	1
	EPF10K30	376,000	—	—	1	1	1
	EPF10K40	498,000	—	—	—	1	1
	EPF10K50	621,000	—	—	—	1	1
	EPF10K70	892,000	—	—	—	1	1
	EPF10K100	1,200,000	—	—	—	—	1
FLEX 10KA	EPF10K10A	120,000	—	—	1	1	1
	EPF10K30A	406,000	—	—	1	1	1
	EPF10K50V	621,000	—	—	—	1	1
	EPF10K100A	1,200,000	—	—	—	—	1
	EPF10K130V	1,600,000	—	—	—	—	1
	EPF10K250A	3,300,000	—	—	—	—	2

Table 4–2. Configuration Devices Required (Part 3 of 4)

Family	Device	Data Size (Bits) *(1)*	EPC1064/ 1064V	EPC1213	EPC1441	EPC1	EPC2
FLEX 10KE	EPF10K30E	473,720	—	—	—	1	1
	EPF10K50E	784,184	—	—	—	1	1
	EPF10K50S	784,184	—	—	—	1	1
	EPF10K100B	1,200,000	—	—	—	—	1
	EPF10K100E	1,335,720	—	—	—	—	1
	EPF10K130E	1,838,360	—	—	—	—	2
	EPF10K200E	2,756,296	—	—	—	—	2
	EPF10K200S	2,756,296	—	—	—	—	2
FLEX 6000/A	EPF6010A	260,000	—	—	1	1	—
	EPF6016 (5.0V) / EPF6016A	260,000	—	—	1	1	—
	EPF6024A	398,000	—	—	1	1	—
FLEX 8000A	EPF8282A / EPF8282AV (3.3 V)	40,000	1	1	1	1	—
	EPF8452A	64,000	1	1	1	1	—
	EPF8636A	96,000	—	1	1	1	—
	EPF8820A	128,000	—	1	1	1	—
	EPF81188A	192,000	—	1	1	1	—
	EPF81500A	250,000	—	—	1	1	—
Mercury	EP1M120	1,303,120	—	—	—	—	1
	EP1M350	4,394,032	—	—	—	—	3
Stratix	EP1S10	3,534,640	—	—	—	—	3 *(3)*
	EP1S20	5,904,832	—	—	—	—	4
	EP1S25	7,894,144	—	—	—	—	5
	EP1S30	10,379,368	—	—	—	—	7
	EP1S40	12,389,632	—	—	—	—	8
	EP1S60	17,543,968	—	—	—	—	11
	EP1S80	23,834,032	—	—	—	—	15
Stratix GX	EP1SGX10	3,534,640	—	—	—	—	3
	EP1SGX25	7,894,144	—	—	—	—	5
	EP1SGX40	12,389,632	—	—	—	—	8

Table 4–2. Configuration Devices Required (Part 4 of 4)

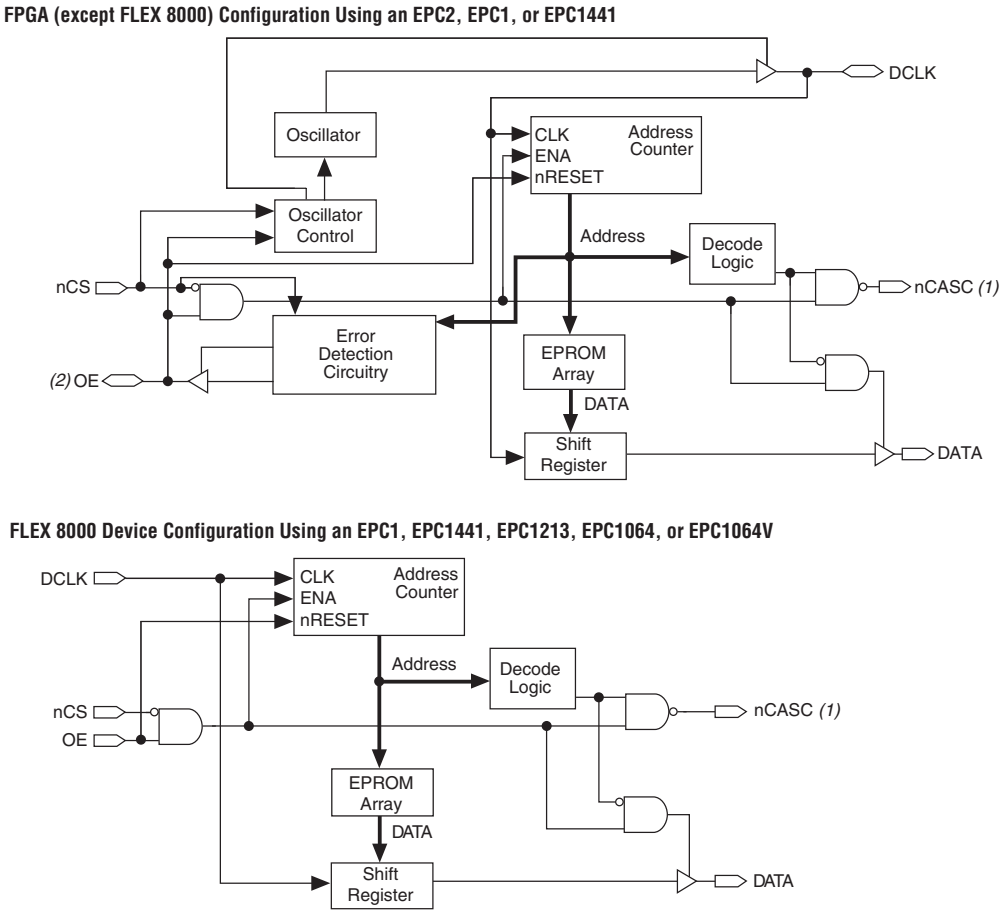
Family	Device	Data Size (Bits) *(1)*	EPC1064/ 1064V	EPC1213	EPC1441	EPC1	EPC2
Stratix II	EP2S15	5,000,000	—	—	—	—	3
	EP2S30	10,100,000	—	—	—	—	7
	EP2S60	17,100,000	—	—	—	—	11
	EP2S90	27,500,000	—	—	—	—	17
	EP2S130	39,600,000	—	—	—	—	24
	EP2S180	52,400,000	—	—	—	—	31

Notes to Table 4–2:

(1) Raw Binary Files (**.rbf**) were used to determine these sizes.
(2) This is with the Cyclone series compression feature enabled.
(3) EP1S10 ES devices requires four EPC2 devices.

Figure 4–1 shows the configuration device block diagram.

Figure 4–1. Configuration Device Block Diagram



Notes to Figure 4–1:

(1) The EPC1441 devices do not support data cascading. The EPC1, EPC2, and EPC1213 devices support data cascading.
(2) The OE pin is a bidirectional open-drain pin.

Device Configuration

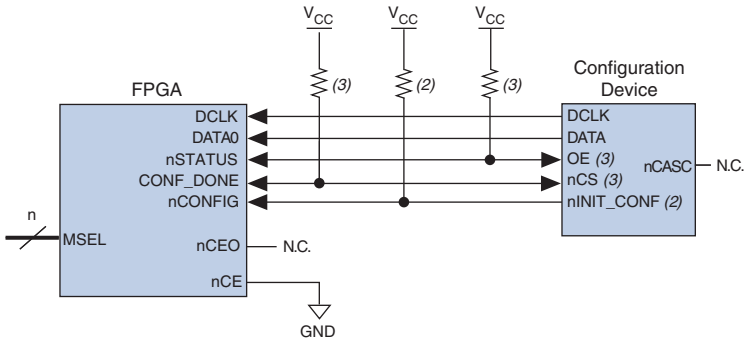
The EPC1, EPC2, and EPC1441 devices store configuration data in its EPROM array and serially clock data out using an internal oscillator. The OE, nCS, and DCLK pins supply the control signals for the address counter and the DATA output tri-state buffer. The configuration device sends a serial bitstream of configuration data to its DATA pin, which is routed to the DATA0 input of the FPGA.

The control signals for configuration devices (OE, nCS, and DCLK) interface directly with the FPGA control signals (nSTATUS, CONF_DONE, and DCLK, respectively). All Altera FPGAs can be configured by a configuration device without requiring an external intelligent controller.

An EPC2 device cannot configure FLEX 8000 or FLEX 6000 devices. Refer to Table 4–2 on page 4–2 for the configuration devices that support FLEX 8000 and FLEX 6000 devices.

Figure 4–2 shows the basic configuration interface connections between the configuration device and the Altera FPGA. For specific details about configuration interface connections, including pull-up resistor values, supply voltages and MSEL pin setting, refer to the appropriate FPGA family chapter in the *Configuration Handbook*.

Figure 4–2. Altera FPGA Configured Using an EPC1, EPC2, or EPC1441 Configuration Device *(Note 1)*



Notes to Figure 4–2:

(1) For specific details about configuration interface connections refer to the FPGA family chapter in the *Configuration Handbook*.

(2) The nINIT_CONF pin (available on EPC2 devices) has an internal pull-up resistor that is always active. This means an external pull-up resistor is not required on the nINIT_CONF/nCONFIG line. The nINIT_CONF pin does not need to be connected if its functionality is not used. If nINIT_CONF is not used or not available, nCONFIG must be pulled to V_{CC} either directly or through a resistor.

(3) EPC2 devices have internal programmable pull-up resistors on OE and nCS. If internal pull-up resistors are used, external pull-up resistors should not be used on these pins. The internal pull-up resistors are used by default in the Quartus II software. To turn off the internal pull-up resistors, check the **Disable nCS and OE pull-ups on configuration device** option when generating programming files.

The EPC2 device allows the user to begin configuration of the FPGA via an additional pin, nINIT_CONF. The nINIT_CONF pin of the EPC2 device can be connected to the nCONFIG of the FPGA, which allows the INIT_CONF JTAG instruction to begin FPGA configuration. The INIT_CONF JTAG instruction causes the EPC2 device to drive nINIT_CONF low, which in turn pulls nCONFIG low. Pulling nCONFIG low on the FPGA will reset the device. When the JTAG state machine exits this state, nINIT_CONF is released and pulled high by an internal 1-kΩ resistor, which in turn pulls nCONFIG high to begin configuration. If its functionality is not used, the nINIT_CONF pin does not need to be connected and nCONFIG of the FPGA must be pulled to V_{CC} either directly or through a resistor.

The EPC2 device's OE and *n*CS pins have internal programmable pull-up resistors. If internal pull-up resistors are used, external pull-up resistors should not be used on these pins. The internal pull-up resistors are used by default in the Quartus II software. To turn off the internal pull-up resistors, check the **Disable nCS** and **OE** pull-ups on configuration device option when generating programming files.

The configuration device's OE and *n*CS pins control the tri-state buffer on its DATA output pin, and enable the address counter and oscillator. When OE is driven low, the configuration device resets the address counter and tri-states its DATA pin. The *n*CS pin controls the DATA output of the configuration device. If *n*CS is held high after the OE reset pulse, the counter is disabled and the DATA output pin is tri-stated. If *n*CS is driven low after the OE reset pulse, the counter and DATA output pin are enabled. When OE is driven low again, the address counter is reset and the DATA output pin is tri-stated, regardless of the state of *n*CS.

If the FPGA's configuration data exceeds the capacity of a single EPC1 or EPC2 configuration device, multiple EPC1 or EPC2 devices can be cascaded together. If multiple EPC1 or EPC2 devices are required, the *n*CASC and *n*CS pins provide handshaking between the configuration devices.

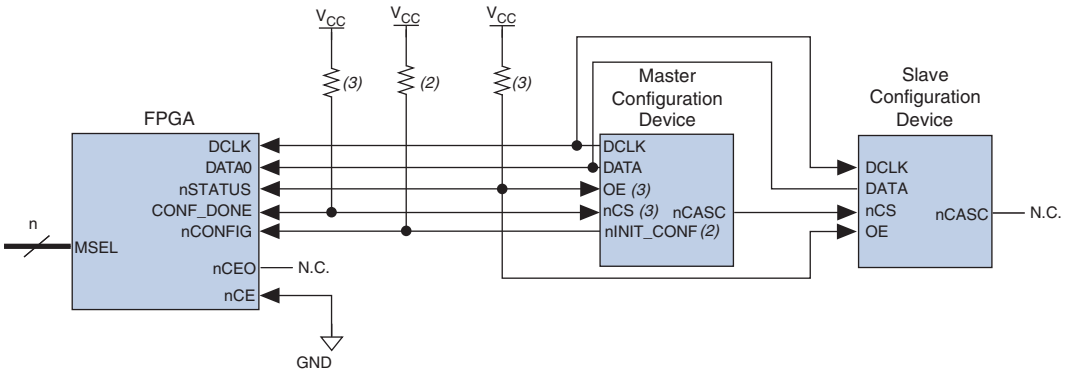
EPC1441 and EPC1064/V devices cannot be cascaded.

When configuring ACEX 1K, APEX 20K, APEX II, Arria GX, Cyclone, Cyclone II, FLEX 10K, Mercury, Stratix, Stratix GX, Stratix II, and Stratix II GX devices with cascaded EPC1 or EPC2 devices, the position of the EPC1 or EPC2 device in the chain determines its mode of operation. The first configuration device in the chain is the master, while subsequent configuration devices are slaves. The *n*INIT_CONF pin of the master EPC2 device can be connected to the *n*CONFIG of the FPGAs, which allows the INIT_CONF JTAG instruction to begin FPGA configuration. The *n*CS pin of the master configuration device is connected to the CONF_DONE of the FPGAs, while its *n*CASC pin is connected to *n*CS of the next slave configuration device in the chain. Additional EPC1 or EPC2 devices can be chained together by connecting *n*CASC to *n*CS of the next slave EPC1 or EPC2 device in the chain. The last device's *n*CS input comes from the previous device, while its *n*CASC pin is left floating. All other configuration pins (DCLK, DATA, and OE) are connected to every device in the chain.

Figure 4–3 shows the basic configuration interface connections between a configuration device chain and the Altera FPGA.

For specific details about configuration interface connections, including pull-up resistor values, supply voltages and MSEL pin setting, refer to the appropriate FPGA family chapter in the *Configuration Handbook*.

Figure 4–3. Altera FPGA Configured Using Two EPC1 or EPC2 Configuration Devices ***(Note 1)***



Notes to Figure 4–3:

(1) For specific details about configuration interface connections refer to the appropriate FPGA family chapter in the *Configuration Handbook*.

(2) The *n*INIT_CONF pin (available on EPC2 devices) has an internal pull-up resistor that is always active. This means an external pull-up resistor is not required on the *n*INIT_CONF/*n*CONFIG line. The *n*INIT_CONF pin does not need to be connected if its functionality is not used. If *n*INIT_CONF is not used or not available, *n*CONFIG must be pulled to V_{CC} either directly or through a resistor.

(3) EPC2 devices have internal programmable pull-up resistors on OE and *n*CS. If internal pull-up resistors are used, external pull-up resistors should not be used on these pins. The internal pull-up resistors are used by default in the Quartus II software. To turn off the internal pull-up resistors, check the **Disable nCS and OE pull-ups on configuration device** option when generating programming files.

When the first device in a configuration device chain is powered-up or reset, its *n*CS pin is driven low because it is connected to the CONF_DONE of the FPGA. Because both OE and *n*CS are low, the first device in the chain will recognize it is the master device and will control configuration. Since the slave devices' *n*CS pin is fed by the previous devices' *n*CASC pin, its *n*CS pin will be high upon power-up and reset. In the slave configuration devices, the DATA output is tri-stated and DCLK is an input. During configuration, the master device supplies the clock through DCLK to the FPGA and to any slave configuration devices. The master EPC1 or EPC2 device also provides the first stream of data to the FPGA during multi-device configuration. After the master EPC1 or EPC2 device finishes sending configuration data, it tri-states its DATA pin to avoid contention with other configuration devices. The master EPC1 or EPC2 device will also drive its *n*CASC pin low, which pulls the *n*CS pin of the next device low. This action signals the slave EPC1 or EPC2 device to start sending configuration data to the FPGAs.

The master EPC1 or EPC2 device clocks all slave configuration devices until configuration is complete. When all configuration data is transferred and the *n*CS pin on the master EPC1 or EPC2 device is driven high by the FPGA's CONF_DONE pin, the master EPC1 or EPC2 device then goes into zero-power (idle) state. The master EPC2 device drives DATA high and DCLK low, while the EPC1 and EPC1441 device tri-state DATA and drive DCLK low.

If *n*CS on the master EPC1 or EPC2 device is driven high before all configuration data is transferred, the master EPC1 or EPC2 device drives its OE signal low, which in turn drives the FPGA's *n*STATUS pin low, indicating a configuration error. Additionally, if the configuration device sends is generated its data and detects that CONF_DONE has not gone high, it recognizes that the FPGA has not configured successfully. EPC1 and EPC2 devices wait for 16 DCLK cycles after the last configuration bit was sent for

CONF_DONE to reach a high state. In this case, the configuration device pulls its OE pin low, which in turn drives the target device's nSTATUS pin low. Configuration automatically restarts if the **Auto-restart configuration on error** option is turned on in the Quartus II software from the **General** tab of the **Device & Pin Options** dialog box or the MAX+PLUS II software's **Global Project Device Options** dialog box (Assign menu).

For more information about FPGA configuration and configuration interface connections between configuration devices and Altera FPGAs, refer to the appropriate FPGA family chapter in the *Configuration Handbook*.

Power and Operation

This section describes Power-On Reset (POR) delay, error detection, and 3.3-V and 5.0-V operation of Altera configuration devices.

Power-On Reset (POR)

During initial power-up, a POR delay occurs to permit voltage levels to stabilize. When configuring an FPGA with an EPC1, EPC2, or EPC1441 device, the POR delay occurs inside the configuration device, and the POR delay is a maximum of 200 ms. When configuring a FLEX 8000 device with an EPC1213, EPC1064, or EPC1064V device, the POR delay occurs inside the FLEX 8000 device, and the POR delay is typically, 100 ms, with a maximum of 200 ms.

During POR, the configuration device drives its OE pin low. This low signal delays configuration because the OE pin is connected to the target FPGA's nSTATUS pin. When the configuration device completes POR, it releases its open-drain OE pin, which is then pulled high by a pull-up resistor.

The FPGA should be powered up before the configuration device exits POR to avoid the master configuration device from entering slave mode.

If the FPGA is not powered up before the configuration device exits POR, the CONF_DONE/nCS line will be high because of the pull-up resistor. When the configuration device exits POR and releases OE, it sees nCS high, which signals the configuration device to enter slave mode. Therefore, configuration will not begin (the DATA output is tri-stated and DCLK is an input pin in slave mode).

Error Detection Circuitry

The EPC1, EPC2, and EPC1441 configuration devices have built-in error detection circuitry for configuring ACEX 1K, APEX 20K, APEX II, Arria GX, Cyclone, Cyclone II, FLEX 10K, FLEX 6000, Mercury, Stratix, Stratix GX, Stratix II, or Stratix II GX devices.

Built-in error-detection circuitry uses the nCS pin of the configuration device, which monitors the CONF_DONE pin on the FPGA. If nCS on the master EPC1 or EPC2 device is driven high before all configuration data is transferred, the master EPC1 or EPC2 device drives its OE signal low, which in turn drives the FPGA's nSTATUS pin low, indicating a configuration error. Additionally, if the configuration device sends is generated its data and detects that CONF_DONE has not gone high, it recognizes that the

FPGA has not configured successfully. EPC1 and EPC2 devices wait for 16 DCLK cycles after the last configuration bit was sent for CONF_DONE to reach a high state. In this case, the configuration device pulls its OE pin low, which in turn drives the target device's nSTATUS pin low. Configuration automatically restarts if the **Auto-restart configuration on error** option is turned on in the Quartus II software from the **General** tab of the **Device & Pin Options** dialog box or the MAX+PLUS II software's **Global Project Device Options** dialog box (Assign menu).

In addition, if the FPGA detects a cyclical redundancy code (CRC) error in the received data, it will flag the error by driving nSTATUS low. This low signal on nSTATUS will drive the OE pin of the configuration device low, which will reset the configuration device. CRC checking is performed when configuring all Altera FPGAs.

3.3-V or 5.0-V Operation

The EPC1, EPC2, and EPC 1441 configuration device may be powered at 3.3 V or 5.0 V. For each configuration device, an option must be set for 5.0-V or 3.3-V operation.

For EPC1 and EPC1441 configuration devices, 3.3-V or 5.0-V operation is controlled by a programming bit in the **.pof**. The **Low-Voltage mode** option in the **Options** tab of the **Configuration Device Options** dialog box in the Quartus II software or the **Use Low-Voltage Configuration EPROM** option in the **Global Project Device Options** dialog box (Assign menu) in the MAX+PLUS II software sets this parameter. For example, EPC1 devices are programmed automatically to operate in 3.3-V mode when configuring FLEX 10KA devices, which have a V_{CC} voltage of 3.3 V. In this example, the EPC1 device's VCC pin is connected to a 3.3-V power supply.

For EPC2 devices, this option is set externally by the VCCSEL pin. In addition, the EPC2 device has an externally controlled option, set by the VPPSEL pin, to adjust the programming voltage to 5.0 V or 3.3 V. The functions of the VCCSEL and VPPSEL pins are described below. These pins are only available in the EPC2 devices.

- VCCSEL pin – For EPC2 configuration devices, 5.0-V or 3.3-V operation is controlled by the VCCSEL option pin. The device functions in 5.0-V mode when VCCSEL is connected to GND; the device functions in 3.3-V mode when VCCSEL is connected to V_{CC}.
- VPPSEL pin – The EPC2 V_{PP} programming power pin is normally tied to V_{CC}. For EPC2 devices operating at 3.3 V, it is possible to improve in-system programming times by setting V_{PP} to 5.0 V. For all other configuration devices, V_{PP} must be tied to V_{CC}. The EPC2 device's VPPSEL pin must be set in accordance with the EPC2 VPP pin. If the VPP pin is supplied by a 5.0-V supply, VPPSEL must be connected to GND; if the VPP pin is supplied by a 3.3-V power supply, VPPSEL must be connected to V_{CC}.

Table 4–3 describes the relationship between the V_{CC} and V_{PP} voltage levels and the required logic level for VCCSEL and VPPSEL. A logic level of high means the pin should be connected to V_{CC}, while a low logic level means the pin should be connected to GND.

Table 4–3. VCCSEL and VPPSEL Pin Functions on the EPC2

VCC Voltage Level (V)	VPP Voltage Level (V)	VCCSEL Pin Logic Level	VPPSEL Pin Logic Level
3.3	3.3	High	High
3.3	5.0	High	Low
5.0	5.0	Low	Low

At 3.3-V operation, all EPC2 inputs are 5.0-V tolerant, except DATA, DCLK, and *n*CASC. The DATA and DCLK pins are used only to interface between the EPC2 device and the FPGA it is configuring. The voltage tolerances of all EPC2 pins at 5.0 V and 3.3 V are listed in Table 4–4.

Table 4–4. EPC2 Input and Bidirectional Pin Voltage Tolerance

Pin	5.0-V Operation		3.3-V Operation	
	5.0-V Tolerant	3.3-V Tolerant	5.0-V Tolerant	3.3-V Tolerant
DATA	✓	✓	—	✓
DCLK	✓	✓	—	✓
*n*CASC	✓	✓	—	✓
OE	✓	✓	✓	✓
*n*CS	✓	✓	✓	✓
VCCSEL	✓	✓	✓	✓
VPPSEL	✓	✓	✓	✓
*n*INIT_CONF	✓	✓	✓	✓
TDI	✓	✓	✓	✓
TMS	✓	✓	✓	✓
TCK	✓	✓	✓	✓

If an EPC1, EPC2, or EPC1441 configuration device is powered at 3.3 V, the *n*STATUS and CONF_DONE pull-up resistors must be connected to 3.3 V. If these configuration devices are powered at 5.0 V, the *n*STATUS and CONF_DONE pull-up resistors can be connected to 3.3 V or 5.0 V.

Programming and Configuration File Support

The Quartus II and MAX+PLUS II softwares provide programming support for Altera configuration devices. During compilation, the Quartus II and MAX+PLUS II softwares automatically generates a **.pof**, which can be used to program the configuration devices. In a multi-device project, the software can combine the programming files for multiple ACEX 1K, APEX 20K, APEX II, Arria GX, Cyclone, Cyclone II, FLEX 10K, Mercury, Stratix, Stratix GX, Stratix II, and Stratix II GX devices into one or more configuration devices. The software allows you to select the appropriate configuration device to most efficiently store the data for each FPGA.

All Altera configuration devices are programmable using Altera programming hardware in conjunction with the Quartus II or MAX+PLUS II software. In addition, many third part programmers offer programming hardware that supports Altera configuration devices.

An EPC2 device can be programmed with a **.pof** generated for an EPC1 or EPC1441 device. An EPC1 device can be programmed using a **.pof** generated for an EPC1441 device.

EPC2 configuration devices can be programmed in-system through its industry-standard 4-pin JTAG interface. ISP capability in the EPC2 devices provides ease in prototyping and FPGA functionality. When programming multiple EPC2 devices in a JTAG chain, the Quartus II and MAX+PLUS II softwares and other programming methods employ concurrent programming to simultaneously program multiple devices and reduce programming time. EPC2 devices can be programmed and erased up to 100 times.

After programming an EPC2 device in-system, FPGA configuration can be initiated by the EPC2 `INIT_CONF` JTAG instruction. Refer to Table 4–6 on page 4–13.

For more information about programming and configuration support, refer to the following documents:

- *Altera Programming Hardware Data Sheet*
- *USB-Blaster Download Cable User Guide*
- *MasterBlaster Serial/USB Communications Cable User Guide*
- *ByteBlaster II Download Cable User Guide*
- *ByteBlasterMV Download Cable User Guide*
- *BitBlaster Serial Download Cable Data Sheet*

You can also program configuration devices using the Quartus II or MAX+PLUS II software with the APU, and the appropriate configuration device programming adapter.

Table 4–5 shows which programming adapter to use with each configuration device.

Table 4–5. Programming Adapters

Device	Package	Adapter
EPC2	20-pin J-Lead	PLMJ1213
	32-pin TQFP	PLMT1213
EPC1	8-pin DIP	PLMJ1213
	20-pin J-Lead	PLMJ1213
EPC1441	8-pin DIP	PLMJ1213
	20-pin J-Lead	PLMJ1213
	32-pin TQFP	PLMT1064

The following steps explain how to program Altera configuration devices using the Quartus II software and the APU:

1. Choose the **Quartus II Programmer** (Tools menu).
2. Load the appropriate **.pof** by clicking **Add**. The **Device** column displays the device for the current programming file.
3. Insert a blank configuration device into the programming adapter's socket.
4. Turn on the **Program/Configure**. You can also turn on **Verify** to verify the contents of a programmed device against the programming data loaded from a programming file.
5. Click **Start**.
6. After successful programming, you can place the configuration device on the PCB to configure the FPGA device.

The following steps explain how to program Altera configuration devices using the MAX+PLUS II software and the APU:

1. Open the MAX+PLUS II Programmer.
2. Load the appropriate **.pof** using the **Select Programming File** dialog box (File menu). By default, the **Programmer** loads the current project's **.pof**. The **Device** field displays the device for the current programming file.
3. Insert a blank configuration device into the programming adapter's socket.
4. Click **Program**.
5. After successful programming, you can place the configuration device on the PCB to configure the FPGA device.

If you are cascading EPC1 or EPC2 devices, you must generate multiple **.pof**. The first device **.pof** will have the same name as the project, while the second device **.pof** will have the same name as the first, but with a "_1" extension (e.g., **top_1.pof**).

IEEE Std. 1149.1 (JTAG) Boundary-Scan Testing

The EPC2 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. The EPC2 device supports the JTAG instructions shown in Table 4–6.

Table 4–6. EPC2 JTAG Instructions (Part 1 of 2)

JTAG Instruction	OPCODE	Description
SAMPLE/PRELOAD	00 0101 0101	Allows a snapshot of a signal at the device pins to be captured and examined during normal device operation, and permits an initial data pattern output at the device pins.
EXTEST	00 0000 0000	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	11 1111 1111	Places the 1-bit bypass register between the TDI and TDO pins, which allows the BST data to pass synchronously through a selected device to adjacent devices during normal device operation.

Table 4–6. EPC2 JTAG Instructions (Part 2 of 2)

JTAG Instruction	OPCODE	Description
IDCODE	00 0101 1001	Selects the device IDCODE register and places it between TDI and TDO, allowing the device IDCODE to be serially shifted out of TDO. The device IDCODE for the EPC2 configuration device is shown below: 0000 0001000000000010 00001101110 1
USERCODE	00 0111 1001	Selects the USERCODE register and places it between TDI and TDO, allowing the USERCODE to be serially shifted out of TDO. The 32-bit USERCODE is a programmable user-defined pattern.
INIT_CONF	00 0110 0001	This function initiates the FPGA re-configuration process by pulsing the *n*INIT_CONF pin low, which is connected to the FPGAs *n*CONFIG pins. After this instruction is updated, the *n*INIT_CONF pin is pulsed low when the JTAG state machine enters the Run-Test/Idle state. The *n*INIT_CONF pin is then released and nCONFIG is pulled high by the resistor after the JTAG state machine goes out of Run-Test/Idle state. The FPGA configuration starts after *n*CONFIG goes high. As a result, the FPGA is configured with the new configuration data stored in the configuration device. This function can be added to your programming file (**.pof**, **.jam**, **.jbc**) in the Quartus II software by enabling the **Initiate configuration after programming** option in the **Programmer options** window (Options menu). This instruction is also used by the MAX+PLUS II software, **.jam** files, and **.jbc** files.
ISP Instructions	—	These instructions are used when programming an EPC2 device via JTAG ports with a USB Blaster, MasterBlaster, ByteBlaster II, EthernetBlaster, or ByteBlaster MV download cable, or using a **.jam**, **.jbc**, or **.svf** via an embedded processor.

For more information, refer to *Application Note 39 (IEEE 1149.1 (JTAG) Boundary-Scan Testing in Altera Devices)* or the EPC2 BSDL files on the Altera web site.

Figure 4–4 shows the timing requirements for the JTAG signals.

Figure 4–4. EPC2 JTAG Waveforms

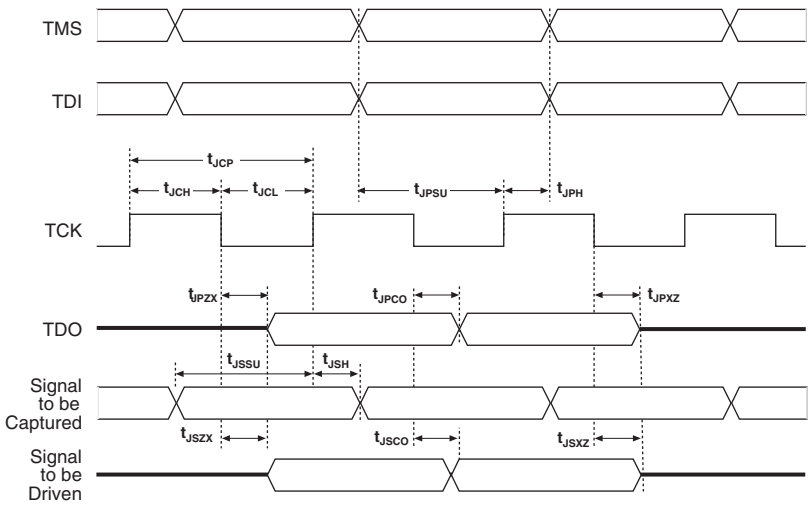


Table 4–7 shows the timing parameters and values for configuration devices.

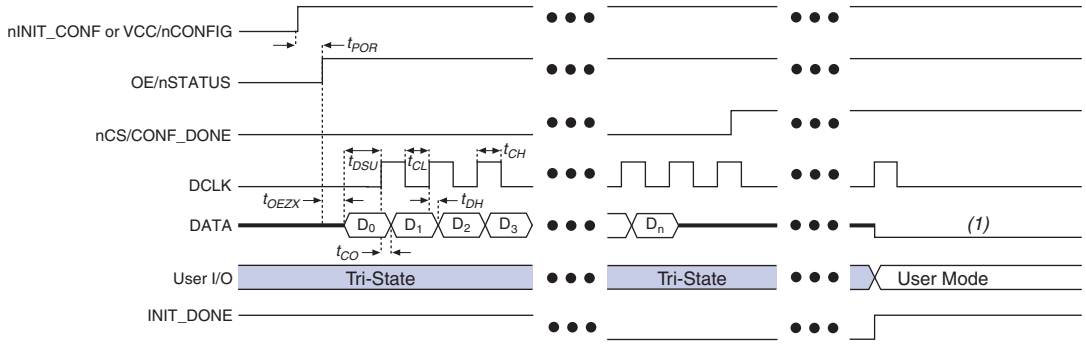
Table 4–7. JTAG Timing Parameters and Values

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

Timing Information

Figure 4–5 shows the timing waveform when using a configuration device.

Figure 4–5. Timing Waveform Using a Configuration Device



Note to Figure 4–5:

(1) The EPC2 device will drive DCLK low and DATA high after configuration. The EPC1 and EPC1441 device will drive DCLK low and tri-state DATA after configuration.

Table 4–8 defines the timing parameters when using EPC2 devices at 3.3 V.

Table 4–8. Timing Parameters when Using EPC2 devices at 3.3 V

Symbol	Parameter	Min	Typ	Max	Units
t_{POR}	POR delay *(1)*	—	—	200	ms
t_{OEZX}	OE high to DATA output enabled	—	—	80	ns
t_{CE}	OE high to first rising edge on DCLK	—	—	300	ns
t_{DSU}	Data setup time before rising edge on DCLK	30	—	—	ns
t_{DH}	Data hold time after rising edge on DCLK	0	—	—	ns
t_{CO}	DCLK to DATA out	—	—	30	ns
t_{CDOE}	DCLK to DATA enable/disable	—	—	30	ns
f_{CLK}	DCLK frequency	5	7.7	12.5	MHz
t_{MCH}	DCLK high time for the first device in the configuration chain	40	65	100	ns
t_{MCL}	DCLK low time for the first device in the configuration chain	40	65	100	ns
t_{SCH}	DCLK high time for subsequent devices	40	—	—	ns
t_{SCL}	DCLK low time for subsequent devices	40	—	—	ns
t_{CASC}	DCLK rising edge to *n*CASC	—	—	25	ns
t_{CCA}	*n*CS to nCASC cascade delay	—	—	15	ns
t_{OEW}	OE low pulse width (reset) to guarantee counter reset	100	—	—	ns
t_{OEC}	OE low (reset) to DCLK disable delay	—	—	30	ns
t_{NRCAS}	OE low (reset) to *n*CASC delay	—	—	30	ns

Note to Table 4–8:

(1) During initial power-up, a POR delay occurs to permit voltage levels to stabilize. Subsequent reconfigurations do not incur this delay.

Table 4–9 defines the timing parameters when using EPC1 and EPC1441 devices at 3.3 V.

Table 4–9. Timing Parameters when Using EPC1 and EPC1441 Devices at 3.3 V (Part 1 of 2)

Symbol	Parameter	Min	Typ	Max	Units
t_{POR}	POR delay *(1)*	—	—	200	ms
t_{OEZX}	OE high to DATA output enabled	—	—	80	ns
t_{CE}	OE high to first rising edge on DCLK	—	—	300	ns
t_{DSU}	Data setup time before rising edge on DCLK	30	—	—	ns
t_{DH}	Data hold time after rising edge on DCLK	0	—	—	ns
t_{CO}	DCLK to DATA out	—	—	30	ns
t_{CDOE}	DCLK to DATA enable/disable	—	—	30	ns
f_{CLK}	DCLK frequency	2	4	10	MHz
t_{MCH}	DCLK high time for the first device in the configuration chain	50	125	250	ns
t_{MCL}	DCLK low time for the first device in the configuration chain	50	125	250	ns
t_{SCH}	DCLK high time for subsequent devices	50	—	—	ns
t_{SCL}	DCLK low time for subsequent devices	50	—	—	ns

Table 4–9. Timing Parameters when Using EPC1 and EPC1441 Devices at 3.3 V (Part 2 of 2)

Symbol	Parameter	Min	Typ	Max	Units
t_{CASC}	DCLK rising edge to *n*CASC	—	—	25	ns
t_{CCA}	*n*CS to *n*CASC cascade delay	—	—	15	ns
t_{OEW}	OE low pulse width (reset) to guarantee counter reset	100	—	—	ns
t_{OEC}	OE low (reset) to DCLK disable delay	—	—	30	ns
t_{NRCAS}	OE low (reset) to *n*CASC delay	—	—	30	ns

Note to Table 4–9:

(1) During initial power-up, a POR delay occurs to permit voltage levels to stabilize. Subsequent reconfigurations do not incur this delay.

Table 4–10 defines the timing parameters when using EPC1, EPC2, and EPC1441 devices at 5.0 V.

Table 4–10. Timing Parameters when Using EPC1, EPC2 and EPC1441 Devices at 5.0 V

Symbol	Parameter	Min	Typ	Max	Units
t_{POR}	POR delay *(1)*	—	—	200	ms
t_{OEZX}	OE high to DATA output enabled	—	—	50	ns
t_{CE}	OE high to first rising edge on DCLK	—	—	200	ns
t_{DSU}	Data setup time before rising edge on DCLK	30	—	—	ns
t_{DH}	Data hold time after rising edge on DCLK	0	—	—	ns
t_{CO}	DCLK to DATA out	—	—	20	ns
t_{CDOE}	DCLK to DATA enable/disable	—	—	20	ns
f_{CLK}	DCLK frequency	6.7	10	16.7	MHz
t_{MCH}	DCLK high time for the first device in the configuration chain	30	50	75	ns
t_{MCL}	DCLK low time for the first device in the configuration chain	30	50	75	ns
t_{SCH}	DCLK high time for subsequent devices	30	—	—	ns
t_{SCL}	DCLK low time for subsequent devices	30	—	—	ns
t_{CASC}	DCLK rising edge to *n*CASC	—	—	20	ns
t_{CCA}	nCS to *n*CASC cascade delay	—	—	10	ns
t_{OEW}	OE low pulse width (reset) to guarantee counter reset	100	—	—	ns
t_{OEC}	OE low (reset) to DCLK disable delay	—	—	20	ns
t_{NRCAS}	OE low (reset) to *n*CASC delay	—	—	25	ns

Note to Table 4–10:

(1) During initial power-up, a POR delay occurs to permit voltage levels to stabilize. Subsequent reconfigurations do not incur this delay.

Table 4–11 defines the timing parameters when using EPC1, EPC1064, EPC1064V, EPC1213, and EPC1441 devices when configuring FLEX 8000 device.

Table 4–11. FLEX 8000 Device Configuration Parameters Using EPC1, EPC1064, EPC1064V, EPC1213, and EPC1441 Devices

Symbol	Parameter	EPC1064V		EPC1064 EPC1213		EPC1 EPC1441		Unit
		Min	Max	Min	Max	Min	Max	
t_{OEZX}	OE high to DATA output enabled	—	75	—	50	—	50	ns
t_{CSZX}	*n*CS low to DATA output enabled	—	75	—	50	—	50	ns
t_{CSXZ}	*n*CS high to DATA output disabled	—	75	—	50	—	50	ns
t_{CSS}	*n*CS low setup time to first DCLK rising edge	150	—	100	—	50	—	ns
t_{CSH}	*n*CS low hold time after DCLK rising edge	0	—	0	—	0	—	ns
t_{DSU}	Data setup time before rising edge on DCLK	75	—	50	—	50	—	ns
t_{DH}	Data hold time after rising edge on DCLK	0	—	0	—	0	—	ns
t_{CO}	DCLK to DATA out delay	—	100	—	75	—	75	ns
t_{CK}	Clock period	240	—	160	—	100	—	ns
f_{CK}	Clock frequency	—	4	—	6	—	8	MHz
t_{CL}	DCLK low time	120	—	80	—	50	—	ns
t_{CH}	DCLK high time	120	—	80	—	50	—	ns
t_{XZ}	OE low or *n*CS high to DATA output disabled	—	75	—	50	—	50	ns
t_{OEW}	OE pulse width to guarantee counter reset	150	—	100	—	100	—	ns
t_{CASC}	Last DCLK + 1 to *n*CASC low delay	—	90	—	60	—	50	ns
t_{CKXZ}	Last DCLK + 1 to DATA tri-state delay	—	75	—	50	—	50	ns
t_{CEOUT}	*n*CS high to *n*CASC high delay	—	150	—	100	—	100	ns

Operating Conditions

Table 4–12 through Table 4–19 provide information about absolute maximum ratings, recommended operating conditions, DC operating conditions, and capacitance for configuration devices.

Table 4–12. Absolute Maximum Ratings ***(Note 1)***

Symbol	Parameter	Conditions	Min	Max	Unit
V_{CC}	Supply voltage	With respect to ground *(2)*	–2.0	7.0	V
V_I	DC input voltage	With respect to ground *(2)*	–2.0	7.0	V
I_{MAX}	DC V_{CC} or ground current	—	—	50	mA
I_{OUT}	DC output current, per pin	—	–25	25	mA
P_D	Power dissipation	—	—	250	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 4–13. Recommended Operating Conditions

Symbol	Parameter	Conditions	Min	Max	Unit
V_{CC}	Supply voltage for 5.0-V operation	*(3), (4)*	4.75 (4.50)	5.25 (5.50)	V
	Supply voltage for 3.3-V operation	*(3), (4)*	3.0 (3.0)	3.6 (3.6)	V
V_I	Input voltage	With respect to ground	–0.3	V_{CC} + 0.3 *(5)*	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 4–14. DC Operating Conditions

Symbol	Parameter	Conditions	Min	Max	Unit
V_{IH}	High-level input voltage	—	2.0	V_{CC} + 0.3 *(5)*	V
V_{IL}	Low-level input voltage	—	–0.3	0.8	V
V_{OH}	5.0-V mode high-level TTL output voltage	I_{OH} = –4 mA DC *(6)*	2.4	—	V
	3.3-V mode high-level CMOS output voltage	I_{OH} = –0.1 mA DC *(6)*	V_{CC} – 0.2	—	V
V_{OL}	Low-level output voltage	I_{OL} = 4 mA DC *(6)*	—	0.4	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

Table 4–15. EPC1064, EPC1064V, and EPC1213 Device I_{CC} Supply Current Values

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
I_{CC0}	V_{CC} supply current (standby)	—	—	100	200	µA
I_{CC1}	V_{CC} supply current (during configuration)	—	—	10	50	mA

Table 4–16. EPC2 Device Values

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
I_{CC0}	V_{CC} supply current (standby)	V_{CC} = 5.0 V or 3.3 V	—	50	100	µA
I_{CC1}	V_{CC} supply current (during configuration)	V_{CC} = 5.0 V or 3.3 V	—	18	50	mA
R_{CONF}	Configuration pins	Internal pull up (OE, *n*CS, *n*INIT_CONF)	—	1	—	kΩ

Table 4–17. EPC1 Device I_{CC} Supply Current Values

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
I_{CC0}	V_{CC} supply current (standby)	—	—	50	100	µA
I_{CC1}	V_{CC} supply current (during configuration)	V_{CC} = 5.0 V	—	30	50	mA
		V_{CC} = 3.3 V	—	10	16.5	mA

Table 4–18. EPC1441 Device I_{CC} Supply Current Values

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
I_{CC0}	V_{CC} supply current (standby)	—	—	30	60	µA
I_{CC1}	V_{CC} supply current (during configuration)	V_{CC} = 5.0 V	—	15	30	mA
I_{CC1}	V_{CC} supply current (during configuration)	V_{CC} = 3.3 V	—	5	10	mA

Table 4–19. Capacitance *(Note 7)*

Symbol	Parameter	Conditions	Min	Max	Unit
C_{IN}	Input pin capacitance	V_{IN} = 0 V, f = 1.0 MHz	—	10	pF
C_{OUT}	Output pin capacitance	V_{OUT} = 0 V, f = 1.0 MHz	—	10	pF

Notes to Table 4–12 through Table 4–19:

(1) For more information, refer to the Operating Requirements for Altera Devices Data Sheet.

(2) The minimum DC input is -0.3 V. During transitions, the inputs may undershoot to -2.0 V or overshoot to 7.0 V for input currents less than 100 mA and periods shorter than 20 ns under no-load conditions.

(3) Numbers in parentheses are for industrial temperature range devices.

(4) Maximum V_{CC} rise time is 100 ms.

(5) Certain EPC2 pins may be driven to 5.75 V when operated with a 3.3-V V_{CC}. Refer to Table 4–4 on page 4–11.

(6) The I_{OH} parameter refers to high-level TTL or CMOS output current; the I_{OL} parameter refers to low-level TTL or CMOS output current.

(7) Capacitance is sample-tested only.

Pin Information

Table 4–20 describes EPC1, EPC2, and EPC1441 pin functions during device configuration.

For pin information about enhanced configuration devices, refer to the *Enhanced Configuration Devices (EPC4, EPC8 and EPC16) Data Sheet*.

For pin information about serial configuration devices, refer to the *Serial Configuration Devices (EPCS1, EPCS4, EPCS16, EPCS64 and EPCS128) Data Sheet*.

Table 4–20. EPC1, EPC2 and EPC1441 Pin Functions During Configuration (Part 1 of 3)

Pin Name	Pin Number			Pin Type	Description
	8-Pin PDIP *(1)*	20-Pin PLCC	32-Pin TQFP *(2)*		
DATA	1	2	31	Output	Serial data output. The DATA pin connects to the DATA0 of the FPGA. DATA is latched into the FPGA on the rising edge of DCLK. The DATA pin is tri-stated before configuration and when the *n*CS pin is high. After configuration, the EPC2 device will drive DATA high, while the EPC1 and EPC1441 device will tri-state DATA.
DCLK	2	4	2	Bidirectional	Clock output when configuring with a single configuration device or when the configuration device is the first (master) device in a chain. Clock input for the next (slave) configuration devices in a chain. The DCLK pin connects to the DCLK of the FPGA. Rising edges on DCLK increment the internal address counter and present the next bit of data on the DATA pin. The counter is incremented only if the OE input is held high, the *n*CS input is held low, and all configuration data has not been transferred to the target device. After configuration or when OE is low, the EPC1, EPC2 and EPC1441 device will drive DCLK low.
OE	3	8	7	Open-Drain Bidirectional	Output enable (active high) and reset (active low). The OE pin connects to the *n*STATUS of the FPGA. A low logic level resets the address counter. A high logic level enables DATA and the address counter to count. If this pin is low (reset) during configuration, the internal oscillator becomes inactive and DCLK drives low. Refer to "Error Detection Circuitry" on page 4–9. The OE pin has an internal programmable 1-kΩ resistor in EPC2 devices. If internal pull-up resistors are use, external pull-up resistors should not be used on these pins. The internal pull-up resistors can be disabled through the **Disable nCS and OE pull-ups on configuration device** option.

Table 4–20. EPC1, EPC2 and EPC1441 Pin Functions During Configuration (Part 2 of 3)

Pin Name	Pin Number			Pin Type	Description
	8-Pin PDIP *(1)*	20-Pin PLCC	32-Pin TQFP *(2)*		
*n*CS	4	9	10	Input	Chip select input (active low). The *n*CS pin connects to the CONF_DONE of the FPGA. A low input allows DCLK to increment the address counter and enables DATA to drive out. If the EPC1 or EPC2 is reset (OE pulled low) while *n*CS is low, the device initializes as the master device in a configuration chain. If the EPC1 or EPC2 device is reset (OE pulled low) while *n*CS is high, the device initializes as a slave device in the chain. The *n*CS pin has an internal programmable 1-kΩ resistor in EPC2 devices. If internal pull-up resistors are use, external pull-up resistors should not be used on these pins.The internal pull-up resistors can be disabled through the **Disable nCS and OE pull-ups on configuration device** option.
*n*CASC	6	12	15	Output	Cascade select output (active low). This output goes low when the address counter has reached its maximum value. When the address counter has reached its maximum value, the configuration device has sent all its configuration data to the FPGA. In a chain of EPC1 or EPC2 devices, the *n*CASC pin of one device is connected to the *n*CS pin of the next device, which permits DCLK to clock data from the next EPC1 or EPC2 device in the chain. For single EPC1 or EPC2 devices and the last device in the chain, *n*CASC is left floating. This pin is only available in EPC1 and EPC2 devices, which support data cascading.
*n*INIT_CONF	N/A	13	16	Open-Drain Output	Allows the INIT_CONF JTAG instruction to initiate configuration. The *n*INIT_CONF pin connects to the *n*CONFIG of the FPGA. If multiple EPC2 devices are used to configure a FPGA, the *n*INIT_CONF of the first EPC2 pin is tied to the FPGA's *n*CONFIG pin, while subsequent devices' *n*INIT_CONF pins are left floating. The INIT_CONF pin has an internal 1-kΩ pull-up resistor that is always active in EPC2 devices. This pin is only available in EPC2 devices.
TDI	N/A	11	13	Input	JTAG data input pin. Connect this pin to V_{CC} if the JTAG circuitry is not used. This pin is only available in EPC2 devices.
TDO	N/A	1	28	Output	JTAG data output pin. Do not connect this pin if the JTAG circuitry is not used. This pin is only available in EPC2 devices.

Table 4–20. EPC1, EPC2 and EPC1441 Pin Functions During Configuration (Part 3 of 3)

Pin Name	Pin Number			Pin Type	Description
	8-Pin PDIP *(1)*	20-Pin PLCC	32-Pin TQFP *(2)*		
TMS	N/A	19	25	Input	JTAG mode select pin. Connect this pin to V_{CC} if the JTAG circuitry is not used. This pin is only available in EPC2 devices.
TCK	N/A	3	32	Input	JTAG clock pin. Connect this pin to ground if the JTAG circuitry is not used. This pin is only available in EPC2 devices.
VCCSEL	N/A	5	3	Input	Mode select for V_{CC} supply. VCCSEL must be connected to ground if the device uses a 5.0-V power supply (V_{CC} = 5.0 V). VCCSEL must be connected to V_{CC} if the device uses a 3.3-V power supply (V_{CC} = 3.3 V). This pin is only available in EPC2 devices.
VPPSEL	N/A	14	17	Input	Mode select for V_{PP}. VPPSEL must be connected to ground if V_{PP} uses a 5.0-V power supply (V_{PP} = 5.0 V). VPPSEL must be connected to V_{CC} if V_{PP} uses a 3.3-V power supply (V_{PP} = 3.3 V). This pin is only available in EPC2 devices.
VPP	N/A	18	23	Power	Programming power pin. For the EPC2 device, this pin is normally tied to V_{CC}. If the EPC2 V_{CC} is 3.3 V, V_{PP} can be tied to 5.0 V to improve in-system programming times. For EPC1 and EPC1441 devices, V_{PP} must be tied to V_{CC}. This pin is only available in EPC2 devices.
VCC	7, 8	20	27	Power	Power pin.
GND	5	10	12	Ground	Ground pin. A 0.2-µF decoupling capacitor must be placed between the VCC and GND pins.

Notes to Table 4–20:

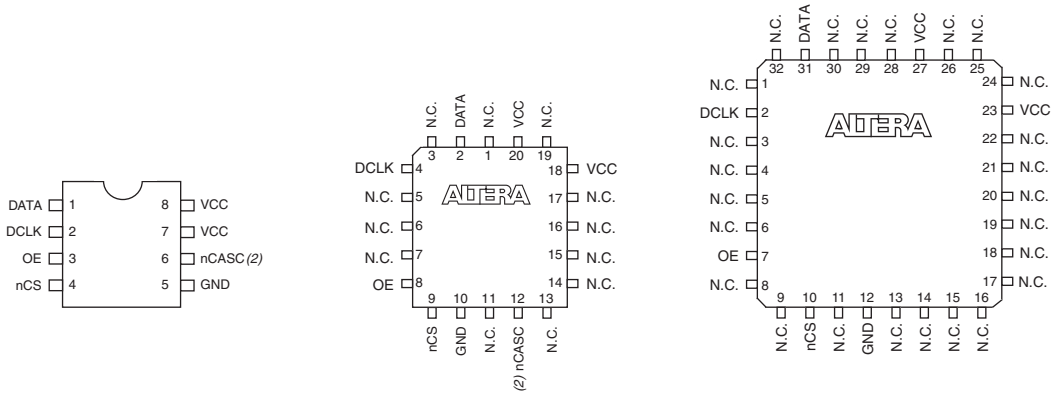
(1) This package is available for EPC1 and EPC1441 devices only.

(2) This package is available for EPC2 and EPC1441 devices only.

Package

Figure 4–6 and Figure 4–7 show the configuration device package pin-outs.

Figure 4–6. EPC1, EPC1064, EPC1064V, EPC1213, and EPC1441 Package Pin-Out Diagrams ***(Note 1)***



8-Pin PDIP

EPC1
EPC1441
EPC1213
EPC1064
EPC1064V

20-Pin PLCC

EPC1
EPC1441
EPC1213
EPC1064
EPC1064V

32-Pin TQFP

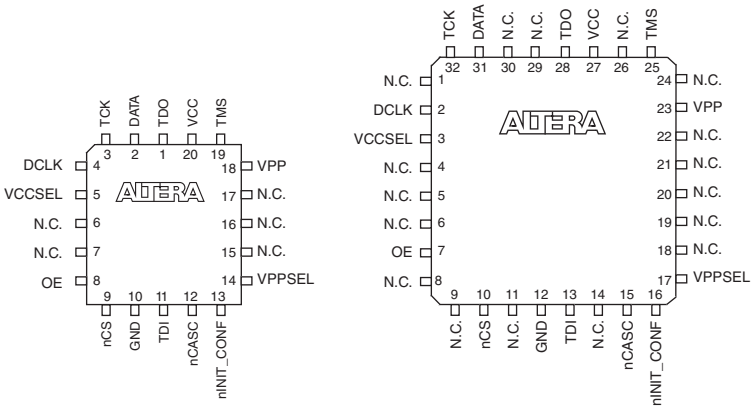
EPC1441
EPC1064
EPC1064V

Notes to Figure 4–6:

(1) EPC1 and EPC1441 devices are one-time programmable devices. ISP is not available in these devices.

(2) The nCASC pin is available on EPC1 devices, which allows them to be cascaded. On the EPC1441 devices, nCASC is a reserved pin and should be left unconnected.

Figure 4–7. EPC2 Package Pin-Out Diagrams



20-Pin PLCC

32-Pin TQFP

For package outlines and drawings, refer to the Altera Device Package Information Data Sheet.

Ordering Codes

Table 4–21. shows the ordering codes for the EPC1, EPC2, and EPC1441 configuration devices.

Table 4–21. Configuration Device Ordering Codes

Device	Package	Temperature	Ordering Code
EPC2	32-pin TQFP	Commercial	EPC2TC32
EPC2	32-pin TQFP	Industrial	EPC2TI32
EPC2	20-pin PLCC	Commercial	EPC2LC20
EPC2	20-pin PLCC	Industrial	EPC2LI20
EPC1	20-pin PLCC	Commercial	EPC1LC20
EPC1	20-pin PLCC	Industrial	EPC1LI20
EPC1	8-pin PDIP	Commercial	EPC1PC8
EPC1	8-pin PDIP	Industrial	EPC1PI8
EPC1441	32-pin TQFP	Commercial	EPC1441TC32
EPC1441	32-pin TQFP	Industrial	EPC1441TI32
EPC1441	20-pin PLCC	Commercial	EPC1441LC20
EPC1441	20-pin PLCC	Industrial	EPC1441LI20
EPC1441	8-pin PDIP	Commercial	EPC1441PC8
EPC1441	8-pin PDIP	Industrial	EPC1441PI8

Document Revision History

Table 4–22 lists the revision history for this chapter.

Table 4–22. Chapter Revision History

Date	Version	Changes
December 2009	2.4	■ Updated "Features" section. ■ Removed "Referenced Documents" section.
October 2008	2.3	■ Updated "Features" and "IEEE Std. 1149.1 (JTAG) Boundary-Scan Testing" sections. ■ Updated Table 5–2 and Table 5–16. ■ Added "Referenced Documents" section. ■ Updated new document format.
April 2007	2.2	Added document revision history.
July 2004	2.0	Added Stratix II and Cyclone II device information throughout chapter.
September 2003	1.0	Initial Release.