
Welcome to E-XFL.COM

## Understanding Embedded - FPGAs (Field Programmable Gate Array)

Embedded - FPGAs, or Field Programmable Gate Arrays, are advanced integrated circuits that offer unparalleled flexibility and performance for digital systems. Unlike traditional fixed-function logic devices, FPGAs can be programmed and reprogrammed to execute a wide array of logical operations, enabling customized functionality tailored to specific applications. This reprogrammability allows developers to iterate designs quickly and implement complex functions without the need for custom hardware.

## Applications of Embedded - FPGAs

The versatility of Embedded - FPGAs makes them indispensable in numerous fields. In telecommunications.

| Details | |
|---|---|
| Product Status | Obsolete |
| Number of LABs/CLBs | 360 |
| Number of Logic Elements/Cells | 2880 |
| Total RAM Bits | 20480 |
| Number of I/O | 189 |
| Number of Gates | 116000 |
| Voltage - Supply | 4.75V ~ 5.25V |
| Mounting Type | Surface Mount |
| Operating Temperature | 0°C ~ 70°C (TA) |
| Package / Case | 240-BFQFP Exposed Pad |
| Supplier Device Package | 240-RQFP (32x32) |
| Purchase URL | https://www.e-xfl.com/product-detail/intel/epf10k50rc240-3 |

For more information, see the following documents:

- *Configuration Devices for APEX & FLEX Devices Data Sheet*
- *BitBlaster Serial Download Cable Data Sheet*
- *ByteBlasterMV Parallel Port Download Cable Data Sheet*
- *Application Note 116 (Configuring APEX 20K, FLEX 10K & FLEX 6000 Devices)*

FLEX 10K devices are supported by Altera development systems; single, integrated packages that offer schematic, text (including AHDL), and waveform design entry, compilation and logic synthesis, full simulation and worst-case timing analysis, and device configuration. The Altera software provides EDIF 2 0 0 and 3 0 0, LPM, VHDL, Verilog HDL, and other interfaces for additional design entry and simulation support from other industry-standard PC- and UNIX workstation-based EDA tools.

The Altera software works easily with common gate array EDA tools for synthesis and simulation. For example, the Altera software can generate Verilog HDL files for simulation with tools such as Cadence Verilog-XL. Additionally, the Altera software contains EDA libraries that use device-specific features such as carry chains which are used for fast counter and arithmetic functions. For instance, the Synopsys Design Compiler library supplied with the Altera development systems include DesignWare functions that are optimized for the FLEX 10K architecture.

The Altera development systems run on Windows-based PCs and Sun SPARCstation, and HP 9000 Series 700/800 workstations.

See the *MAX+PLUS II Programmable Logic Development System & Software Data Sheet* for more information.

## Functional Description

Each FLEX 10K device contains an embedded array to implement memory and specialized logic functions, and a logic array to implement general logic.

The embedded array consists of a series of EABs. When implementing memory functions, each EAB provides 2,048 bits, which can be used to create RAM, ROM, dual-port RAM, or first-in first-out (FIFO) functions. When implementing logic, each EAB can contribute 100 to 600 gates towards complex logic functions, such as multipliers, microcontrollers, state machines, and DSP functions. EABs can be used independently, or multiple EABs can be combined to implement larger functions.

Each LAB provides four control signals with programmable inversion that can be used in all eight LEs. Two of these signals can be used as clocks; the other two can be used for clear/preset control. The LAB clocks can be driven by the dedicated clock input pins, global signals, I/O signals, or internal signals via the LAB local interconnect. The LAB preset and clear control signals can be driven by the global signals, I/O signals, or internal signals via the LAB local interconnect. The global control signals are typically used for global clock, clear, or preset signals because they provide asynchronous control with very low skew across the device. If logic is required on a control signal, it can be generated in one or more LEs in any LAB and driven into the local interconnect of the target LAB. In addition, the global control signals can be generated from LE outputs.

## Logic Element

The LE, the smallest unit of logic in the FLEX 10K architecture, has a compact size that provides efficient logic utilization. Each LE contains a four-input LUT, which is a function generator that can quickly compute any function of four variables. In addition, each LE contains a programmable flipflop with a synchronous enable, a carry chain, and a cascade chain. Each LE drives both the local and the FastTrack Interconnect. See Figure 6.

*Figure 6. FLEX 10K Logic Element*

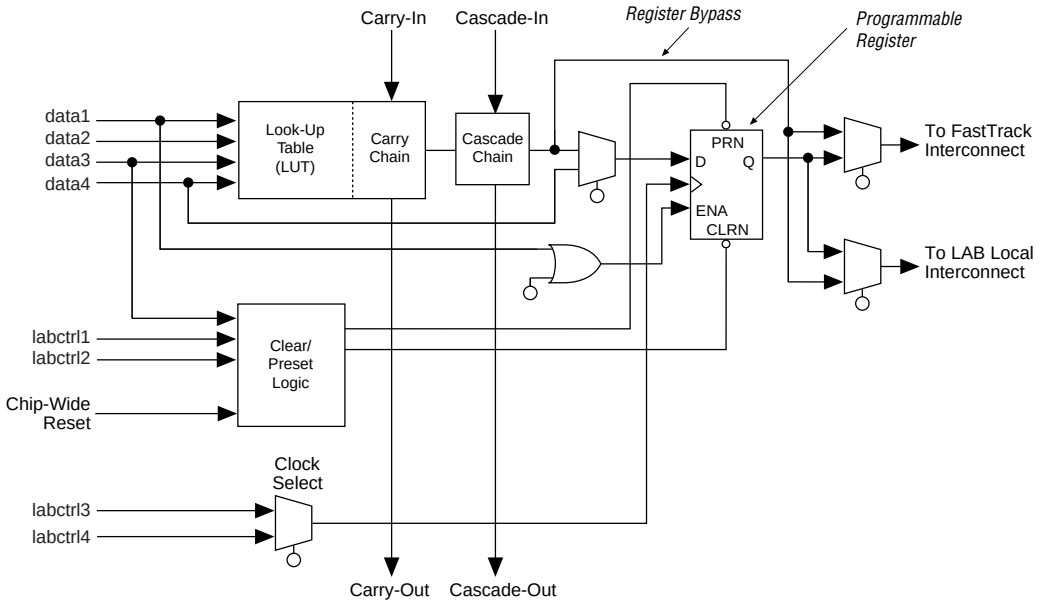
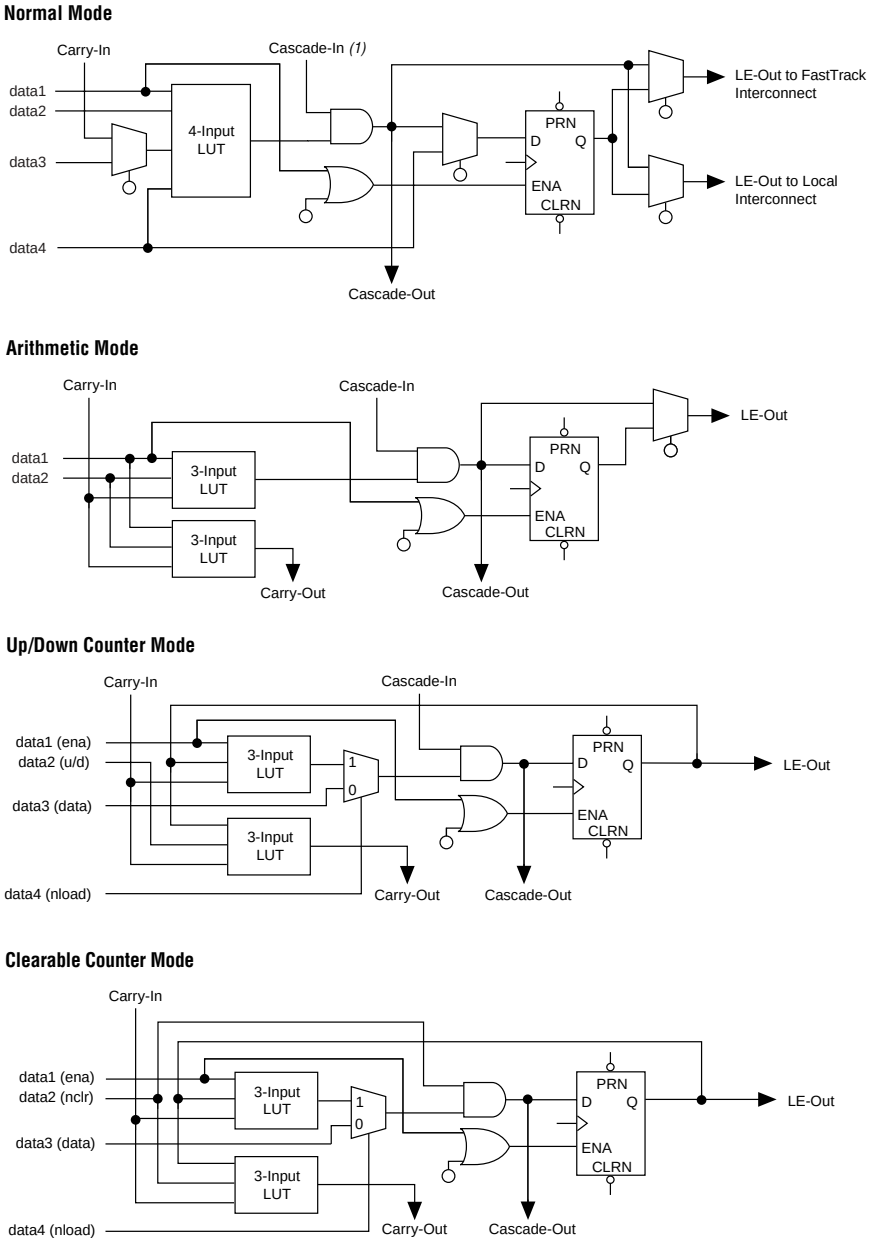


*Figure 9. FLEX 10K LE Operating Modes*




*Note:*
(1) Packed registers cannot be used with the cascade chain.

## FastTrack Interconnect

In the FLEX 10K architecture, connections between LEs and device I/O pins are provided by the FastTrack Interconnect, which is a series of continuous horizontal and vertical routing channels that traverse the device. This global routing structure provides predictable performance, even in complex designs. In contrast, the segmented routing in FPGAs requires switch matrices to connect a variable number of routing paths, increasing the delays between logic resources and reducing performance.

The FastTrack Interconnect consists of row and column interconnect channels that span the entire device. Each row of LABs is served by a dedicated row interconnect. The row interconnect can drive I/O pins and feed other LABs in the device. The column interconnect routes signals between rows and can drive I/O pins.

A row channel can be driven by an LE or by one of three column channels. These four signals feed dual 4-to-1 multiplexers that connect to two specific row channels. These multiplexers, which are connected to each LE, allow column channels to drive row channels even when all eight LEs in an LAB drive the row interconnect.

Each column of LABs is served by a dedicated column interconnect. The column interconnect can then drive I/O pins or another row's interconnect to route the signals to other LABs in the device. A signal from the column interconnect, which can be either the output of an LE or an input from an I/O pin, must be routed to the row interconnect before it can enter an LAB or EAB. Each row channel that is driven by an IOE or EAB can drive one specific column channel.

Access to row and column channels can be switched between LEs in adjacent pairs of LABs. For example, an LE in one LAB can drive the row and column channels normally driven by a particular LE in the adjacent LAB in the same row, and vice versa. This routing flexibility enables routing resources to be used more efficiently. See Figure 11.

Signals on the peripheral control bus can also drive the four global signals, referred to as GLOBAL0 through GLOBAL3 in Tables 8 and 9. The internally generated signal can drive the global signal, providing the same low-skew, low-delay characteristics for an internally generated signal as for a signal driven by an input. This feature is ideal for internally generated clear or clock signals with high fan-out. When a global signal is driven by internal logic, the dedicated input pin that drives that global signal cannot be used. The dedicated input pin should be driven to a known logic state (such as ground) and not be allowed to float.

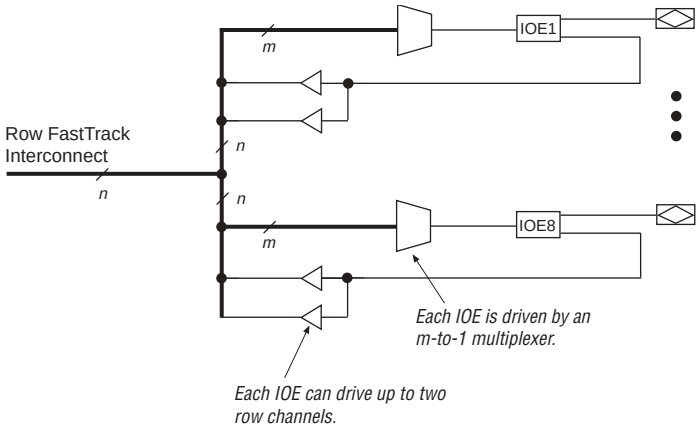
When the chip-wide output enable pin is held low, it will tri-state all pins on the device. This option can be set in the Global Project Device Options menu. Additionally, the registers in the IOE can be reset by holding the chip-wide reset pin low.

### *Row-to-IOE Connections*

When an IOE is used as an input signal, it can drive two separate row channels. The signal is accessible by all LEs within that row. When an IOE is used as an output, the signal is driven by a multiplexer that selects a signal from the row channels. Up to eight IOEs connect to each side of each row channel. See Figure 14.

***Figure 14. FLEX 10K Row-to-IOE Connections***

*The values for m and n are provided in Table 10.*



## SameFrame Pin-Outs

FLEX 10KE devices support the SameFrame pin-out feature for FineLine BGA packages. The SameFrame pin-out feature is the arrangement of balls on FineLine BGA packages such that the lower-ball-count packages form a subset of the higher-ball-count packages. SameFrame pin-outs provide the flexibility to migrate not only from device to device within the same package, but also from one package to another. A given printed circuit board (PCB) layout can support multiple device density/package combinations. For example, a single board layout can support a range of devices from an EPF10K10A device in a 256-pin FineLine BGA package to an EPF10K100A device in a 484-pin FineLine BGA package.

The Altera software provides support to design PCBs with SameFrame pin-out devices. Devices can be defined for present and future use. The Altera software generates pin-outs describing how to lay out a board to take advantage of this migration (see Figure 16).

*Figure 16. SameFrame Pin-Out Example*

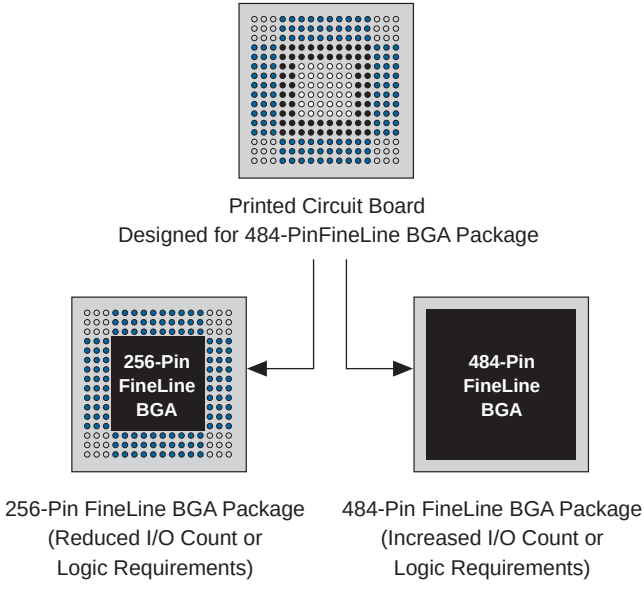


Figure 18 shows the timing requirements for the JTAG signals.

*Figure 18. JTAG Waveforms*

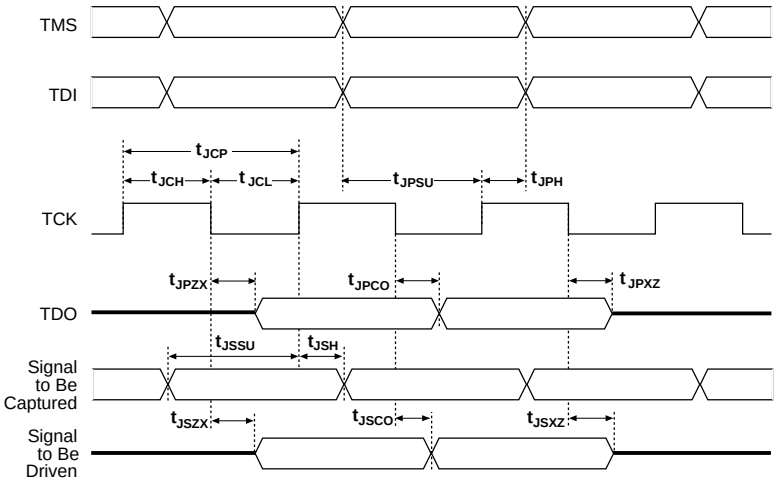
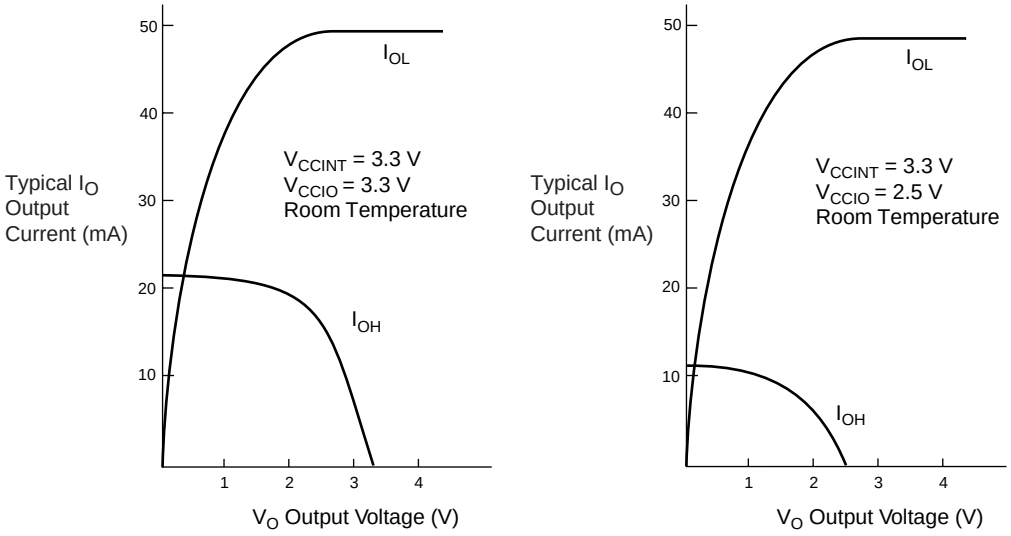



Table 16 shows the timing parameters and values for FLEX 10K devices.

*Table 16. 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 to output | | 25 | ns |
| $t_{JPZX}$ | JTAG port high impedance to valid output | | 25 | ns |
| $t_{JPXZ}$ | JTAG port valid output to high impedance | | 25 | ns |
| $t_{JSSU}$ | Capture register setup time | 20 | | ns |
| $t_{JSH}$ | Capture register hold time | 45 | | ns |
| $t_{JSCO}$ | Update register clock to output | | 35 | ns |
| $t_{JSZX}$ | Update register high-impedance to valid output | | 35 | ns |
| $t_{JSXZ}$ | Update register valid output to high impedance | | 35 | ns |

*Figure 23. Output Drive Characteristics for EPF10K250A Device*




## Timing Model

The continuous, high-performance FastTrack Interconnect routing resources ensure predictable performance and accurate simulation and timing analysis. This predictable performance contrasts with that of FPGAs, which use a segmented connection scheme and therefore have unpredictable performance.

Device performance can be estimated by following the signal path from a source, through the interconnect, to the destination. For example, the registered performance between two LEs on the same row can be calculated by adding the following parameters:

- LE register clock-to-output delay ($t_{CO}$)
- Interconnect delay ($t_{SAMEROW}$)
- LE look-up table delay ($t_{LUT}$)
- LE register setup time ($t_{SU}$)

The routing delay depends on the placement of the source and destination LEs. A more complex registered path may involve multiple combinatorial LEs between the source and destination LEs.

*Table 72. EPF10K50V Device IOE Timing Microparameters* *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | -4 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | Min | Max | |
| $t_{IOD}$ | | 1.2 | | 1.6 | | 1.9 | | 2.1 | ns |
| $t_{IOC}$ | | 0.3 | | 0.4 | | 0.5 | | 0.5 | ns |
| $t_{IOCO}$ | | 0.3 | | 0.3 | | 0.4 | | 0.4 | ns |
| $t_{IOCOMB}$ | | 0.0 | | 0.0 | | 0.0 | | 0.0 | ns |
| $t_{IOSU}$ | 2.8 | | 2.8 | | 3.4 | | 3.9 | | ns |
| $t_{IOH}$ | 0.7 | | 0.8 | | 1.0 | | 1.4 | | ns |
| $t_{IOCLR}$ | | 0.5 | | 0.6 | | 0.7 | | 0.7 | ns |
| $t_{OD1}$ | | 2.8 | | 3.2 | | 3.9 | | 4.7 | ns |
| $t_{OD2}$ | | – | | – | | – | | – | ns |
| $t_{OD3}$ | | 6.5 | | 6.9 | | 7.6 | | 8.4 | ns |
| $t_{XZ}$ | | 2.8 | | 3.1 | | 3.8 | | 4.6 | ns |
| $t_{ZX1}$ | | 2.8 | | 3.1 | | 3.8 | | 4.6 | ns |
| $t_{ZX2}$ | | – | | – | | – | | – | ns |
| $t_{ZX3}$ | | 6.5 | | 6.8 | | 7.5 | | 8.3 | ns |
| $t_{INREG}$ | | 5.0 | | 5.7 | | 7.0 | | 9.0 | ns |
| $t_{IOFD}$ | | 1.5 | | 1.9 | | 2.3 | | 2.7 | ns |
| $t_{INCOMB}$ | | 1.5 | | 1.9 | | 2.3 | | 2.7 | ns |

*Table 75. EPF10K50V Device Interconnect Timing Microparameters* *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | -4 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | Min | Max | |
| $t_{DIN2IOE}$ | | 4.7 | | 6.0 | | 7.1 | | 8.2 | ns |
| $t_{DIN2LE}$ | | 2.5 | | 2.6 | | 3.1 | | 3.9 | ns |
| $t_{DIN2DATA}$ | | 4.4 | | 5.9 | | 6.8 | | 7.7 | ns |
| $t_{DCLK2IOE}$ | | 2.5 | | 3.9 | | 4.7 | | 5.5 | ns |
| $t_{DCLK2LE}$ | | 2.5 | | 2.6 | | 3.1 | | 3.9 | ns |
| $t_{SAMELAB}$ | | 0.2 | | 0.2 | | 0.3 | | 0.3 | ns |
| $t_{SAMEROW}$ | | 2.8 | | 3.0 | | 3.2 | | 3.4 | ns |
| $t_{SAMECOLUMN}$ | | 3.0 | | 3.2 | | 3.4 | | 3.6 | ns |
| $t_{DIFFROW}$ | | 5.8 | | 6.2 | | 6.6 | | 7.0 | ns |
| $t_{TWOROWS}$ | | 8.6 | | 9.2 | | 9.8 | | 10.4 | ns |
| $t_{LEPERIPH}$ | | 4.5 | | 5.5 | | 6.1 | | 7.0 | ns |
| $t_{LABCARRY}$ | | 0.3 | | 0.4 | | 0.5 | | 0.7 | ns |
| $t_{LABCASC}$ | | 0.0 | | 1.3 | | 1.6 | | 2.0 | ns |

*Table 76. EPF10K50V Device External Timing Parameters* *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | -4 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | Min | Max | |
| $t_{DRR}$ | | 11.2 | | 14.0 | | 17.2 | | 21.1 | ns |
| $t_{INSU}$ *(2), (3)* | 5.5 | | 4.2 | | 5.2 | | 6.9 | | ns |
| $t_{INH}$ *(3)* | 0.0 | | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{OUTCO}$ *(3)* | 2.0 | 5.9 | 2.0 | 7.8 | 2.0 | 9.5 | 2.0 | 11.1 | ns |

*Table 77. EPF10K50V Device External Bidirectional Timing Parameters* *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | -4 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | Min | Max | |
| $t_{INSUBIDIR}$ | 2.0 | | 2.8 | | 3.5 | | 4.1 | | ns |
| $t_{INHBIDIR}$ | 0.0 | | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{OUTCOBIDIR}$ | 2.0 | 5.9 | 2.0 | 7.8 | 2.0 | 9.5 | 2.0 | 11.1 | ns |
| $t_{XZBIDIR}$ | | 8.0 | | 9.8 | | 11.8 | | 14.3 | ns |
| $t_{ZXBIDIR}$ | | 8.0 | | 9.8 | | 11.8 | | 14.3 | ns |

*Table 86. EPF10K10A Device IOE Timing Microparameters* *Note (1)* *(Part 2 of 2)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{IOH}$ | 0.8 | | 1.0 | | 1.3 | | ns |
| $t_{IOCLR}$ | | 1.2 | | 1.4 | | 1.9 | ns |
| $t_{OD1}$ | | 1.2 | | 1.4 | | 1.9 | ns |
| $t_{OD2}$ | | 2.9 | | 3.5 | | 4.7 | ns |
| $t_{OD3}$ | | 6.6 | | 7.8 | | 10.5 | ns |
| $t_{XZ}$ | | 1.2 | | 1.4 | | 1.9 | ns |
| $t_{ZX1}$ | | 1.2 | | 1.4 | | 1.9 | ns |
| $t_{ZX2}$ | | 2.9 | | 3.5 | | 4.7 | ns |
| $t_{ZX3}$ | | 6.6 | | 7.8 | | 10.5 | ns |
| $t_{INREG}$ | | 5.2 | | 6.3 | | 8.4 | ns |
| $t_{IOFD}$ | | 3.1 | | 3.8 | | 5.0 | ns |
| $t_{INCOMB}$ | | 3.1 | | 3.8 | | 5.0 | ns |

| *Table 95. EPF10K30A Device EAB Internal Timing Macroparameters* *Note (1)* | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Symbol** | **-1 Speed Grade** | | **-2 Speed Grade** | | **-3 Speed Grade** | | **Unit** |
| | **Min** | **Max** | **Min** | **Max** | **Min** | **Max** | |
| $t_{EABAA}$ | | 9.7 | | 11.6 | | 16.2 | ns |
| $t_{EABRCCOMB}$ | 9.7 | | 11.6 | | 16.2 | | ns |
| $t_{EABRCREG}$ | 5.9 | | 7.1 | | 9.7 | | ns |
| $t_{EABWP}$ | 3.8 | | 4.5 | | 5.9 | | ns |
| $t_{EABWCCOMB}$ | 4.0 | | 4.7 | | 6.3 | | ns |
| $t_{EABWCREG}$ | 9.8 | | 11.6 | | 16.6 | | ns |
| $t_{EABDD}$ | | 9.2 | | 11.0 | | 16.1 | ns |
| $t_{EABDATACO}$ | | 1.7 | | 2.1 | | 3.4 | ns |
| $t_{EABDATASU}$ | 2.3 | | 2.7 | | 3.5 | | ns |
| $t_{EABDATAH}$ | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{EABWESU}$ | 3.3 | | 3.9 | | 4.9 | | ns |
| $t_{EABWEH}$ | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{EABWDSU}$ | 3.2 | | 3.8 | | 5.0 | | ns |
| $t_{EABWDH}$ | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{EABWASU}$ | 3.7 | | 4.4 | | 5.1 | | ns |
| $t_{EABWAH}$ | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{EABWO}$ | | 6.1 | | 7.3 | | 11.3 | ns |

*Table 102. EPF10K100A Device EAB Internal Timing Macroparameters* *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{EABAA}$ | | 6.8 | | 7.8 | | 9.2 | ns |
| $t_{EABRCCOMB}$ | 6.8 | | 7.8 | | 9.2 | | ns |
| $t_{EABRCREG}$ | 5.4 | | 6.2 | | 7.4 | | ns |
| $t_{EABWP}$ | 3.2 | | 3.7 | | 4.4 | | ns |
| $t_{EABWCCOMB}$ | 3.4 | | 3.9 | | 4.7 | | ns |
| $t_{EABWCREG}$ | 9.4 | | 10.8 | | 12.8 | | ns |
| $t_{EABDD}$ | | 6.1 | | 6.9 | | 8.2 | ns |
| $t_{EABDATACO}$ | | 2.1 | | 2.3 | | 2.9 | ns |
| $t_{EABDATASU}$ | 3.7 | | 4.3 | | 5.1 | | ns |
| $t_{EABDATAH}$ | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{EABWESU}$ | 2.8 | | 3.3 | | 3.8 | | ns |
| $t_{EABWEH}$ | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{EABWDSU}$ | 3.4 | | 4.0 | | 4.6 | | ns |
| $t_{EABWDH}$ | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{EABWASU}$ | 1.9 | | 2.3 | | 2.6 | | ns |
| $t_{EABWAH}$ | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{EABWO}$ | | 5.1 | | 5.7 | | 6.9 | ns |

*Table 103. EPF10K100A Device Interconnect Timing Microparameters* *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{DIN2IOE}$ | | 4.8 | | 5.4 | | 6.0 | ns |
| $t_{DIN2LE}$ | | 2.0 | | 2.4 | | 2.7 | ns |
| $t_{DIN2DATA}$ | | 2.4 | | 2.7 | | 2.9 | ns |
| $t_{DCLK2IOE}$ | | 2.6 | | 3.0 | | 3.5 | ns |
| $t_{DCLK2LE}$ | | 2.0 | | 2.4 | | 2.7 | ns |
| $t_{SAMELAB}$ | | 0.1 | | 0.1 | | 0.1 | ns |
| $t_{SAMEROW}$ | | 1.5 | | 1.7 | | 1.9 | ns |
| $t_{SAMECOLUMN}$ | | 5.5 | | 6.5 | | 7.4 | ns |
| $t_{DIFFROW}$ | | 7.0 | | 8.2 | | 9.3 | ns |
| $t_{TWOROWS}$ | | 8.5 | | 9.9 | | 11.2 | ns |
| $t_{LEPERIPH}$ | | 3.9 | | 4.2 | | 4.5 | ns |
| $t_{LABCARRY}$ | | 0.2 | | 0.2 | | 0.3 | ns |
| $t_{LABCASC}$ | | 0.4 | | 0.5 | | 0.6 | ns |

*Table 104. EPF10K100A Device External Timing Parameters* *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{DRR}$ | | 12.5 | | 14.5 | | 17.0 | ns |
| $t_{INSU}$ *(2)*, *(3)* | 3.7 | | 4.5 | | 5.1 | | ns |
| $t_{INH}$ *(3)* | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{OUTCO}$ *(3)* | 2.0 | 5.3 | 2.0 | 6.1 | 2.0 | 7.2 | ns |

*Table 105. EPF10K100A Device External Bidirectional Timing Parameters* *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{INSUBIDIR}$ | 4.9 | | 5.8 | | 6.8 | | ns |
| $t_{INHBIDIR}$ | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{OUTCOBIDIR}$ | 2.0 | 5.3 | 2.0 | 6.1 | 2.0 | 7.2 | ns |
| $t_{XZBIDIR}$ | | 7.4 | | 8.6 | | 10.1 | ns |
| $t_{ZXBIDIR}$ | | 7.4 | | 8.6 | | 10.1 | ns |

*Table 110. EPF10K250A Device Interconnect Timing Microparameters* *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{DIN2IOE}$ | | 7.8 | | 8.5 | | 9.4 | ns |
| $t_{DIN2LE}$ | | 2.7 | | 3.1 | | 3.5 | ns |
| $t_{DIN2DATA}$ | | 1.6 | | 1.6 | | 1.7 | ns |
| $t_{DCLK2IOE}$ | | 3.6 | | 4.0 | | 4.6 | ns |
| $t_{DCLK2LE}$ | | 2.7 | | 3.1 | | 3.5 | ns |
| $t_{SAMELAB}$ | | 0.2 | | 0.3 | | 0.3 | ns |
| $t_{SAMEROW}$ | | 6.7 | | 7.3 | | 8.2 | ns |
| $t_{SAMECOLUMN}$ | | 2.5 | | 2.7 | | 3.0 | ns |
| $t_{DIFFROW}$ | | 9.2 | | 10.0 | | 11.2 | ns |
| $t_{TWOROWS}$ | | 15.9 | | 17.3 | | 19.4 | ns |
| $t_{LEPERIPH}$ | | 7.5 | | 8.1 | | 8.9 | ns |
| $t_{LABCARRY}$ | | 0.3 | | 0.4 | | 0.5 | ns |
| $t_{LABCASC}$ | | 0.4 | | 0.4 | | 0.5 | ns |

*Table 111. EPF10K250A Device External Reference Timing Parameters* *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{DRR}$ | | 15.0 | | 17.0 | | 20.0 | ns |
| $t_{INSU}$ *(2), (3)* | 6.9 | | 8.0 | | 9.4 | | ns |
| $t_{INH}$ *(3)* | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{OUTCO}$ *(3)* | 2.0 | 8.0 | 2.0 | 8.9 | 2.0 | 10.4 | ns |

*Table 112. EPF10K250A Device External Bidirectional Timing Parameters* *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{INSUBIDIR}$ | 9.3 | | 10.6 | | 12.7 | | ns |
| $t_{INHBIDIR}$ | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{OUTCOBIDIR}$ | 2.0 | 8.0 | 2.0 | 8.9 | 2.0 | 10.4 | ns |
| $t_{XZBIDIR}$ | | 10.8 | | 12.2 | | 14.2 | ns |
| $t_{ZXBIDIR}$ | | 10.8 | | 12.2 | | 14.2 | ns |

*Table 113. ClockLock & ClockBoost Parameters (Part 2 of 2)*

| Symbol | Parameter | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $f_{CLKDEV1}$ | Input deviation from user specification in MAX+PLUS II (ClockBoost clock multiplication factor equals 1) *(1)* | | | ±1 | MHz |
| $f_{CLKDEV2}$ | Input deviation from user specification in MAX+PLUS II (ClockBoost clock multiplication factor equals 2) *(1)* | | | ±0.5 | MHz |
| $t_{INCLKSTB}$ | Input clock stability (measured between adjacent clocks) | | | 100 | ps |
| $t_{LOCK}$ | Time required for ClockLock or ClockBoost to acquire lock *(2)* | | | 10 | µs |
| $t_{JITTER}$ | Jitter on ClockLock or ClockBoost-generated clock *(3)* | | | 1 | ns |
| $t_{OUTDUTY}$ | Duty cycle for ClockLock or ClockBoost-generated clock | 40 | 50 | 60 | % |

*Notes:*

(1) To implement the ClockLock and ClockBoost circuitry with the MAX+PLUS II software, designers must specify the input frequency. The MAX+PLUS II software tunes the PLL in the ClockLock and ClockBoost circuitry to this frequency. The $f_{CLKDEV}$ parameter specifies how much the incoming clock can differ from the specified frequency during device operation. Simulation does not reflect this parameter.

(2) During device configuration, the ClockLock and ClockBoost circuitry is configured before the rest of the device. If the incoming clock is supplied during configuration, the ClockLock and ClockBoost circuitry locks during configuration, because the $t_{LOCK}$ value is less than the time required for configuration.

(3) The $t_{JITTER}$ specification is measured under long-term observation.

## Power Consumption

The supply power (P) for FLEX 10K devices can be calculated with the following equation:

$$P = P_{INT} + P_{IO} = (I_{CCSTANDBY} + I_{CCACTIVE}) \times V_{CC} + P_{IO}$$

Typical $I_{CCSTANDBY}$ values are shown as $I_{CC0}$ in the FLEX 10K device DC operating conditions tables on pages 46, 49, and 52 of this data sheet. The $I_{CCACTIVE}$ value depends on the switching frequency and the application logic. This value is calculated based on the amount of current that each LE typically consumes. The $P_{IO}$ value, which depends on the device output load characteristics and switching frequency, can be calculated using the guidelines given in *Application Note 74 (Evaluating Power for Altera Devices)*.

Compared to the rest of the device, the embedded array consumes a negligible amount of power. Therefore, the embedded array can be ignored when calculating supply current.

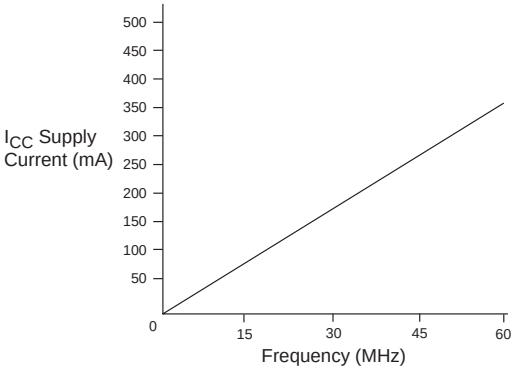
The $I_{CCACTIVE}$ value is calculated with the following equation:

$$I_{CCACTIVE} = K \times \mathbf{f_{MAX}} \times N \times \mathbf{tog_{LC}} \times \frac{\mu A}{MHz \times LE}$$

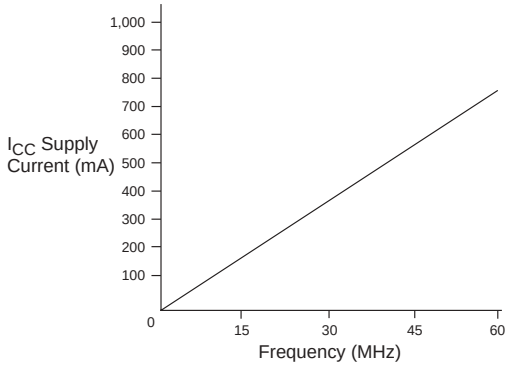
The parameters in this equation are shown below:

*Figure 32. $I_{CCACTIVE}$ vs. Operating Frequency (Part 1 of 3)*

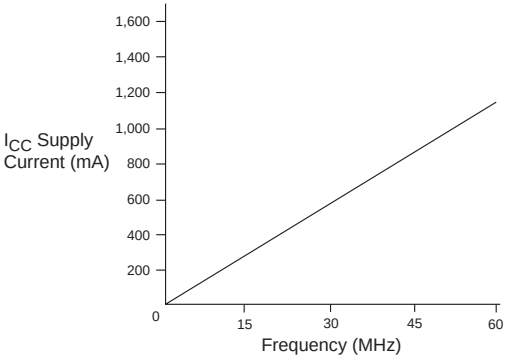
**EPF10K10**



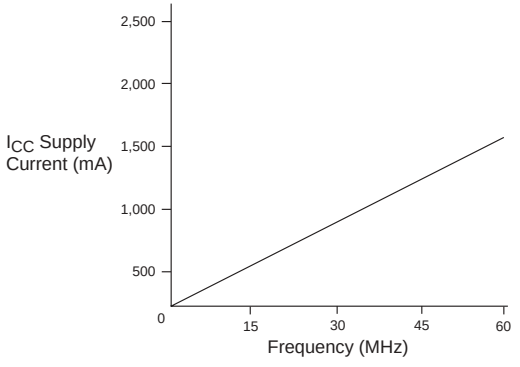

**EPF10K20**



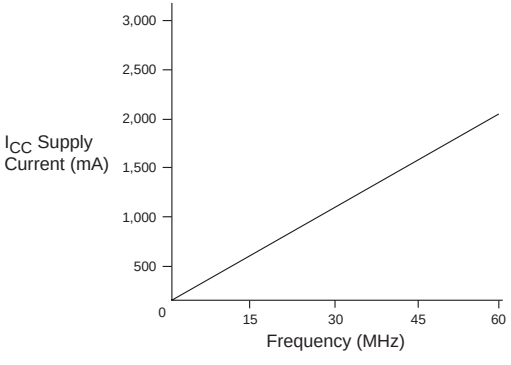

**EPF10K30**




**EPF10K40**




**EPF10K50**




**EPF10K70**

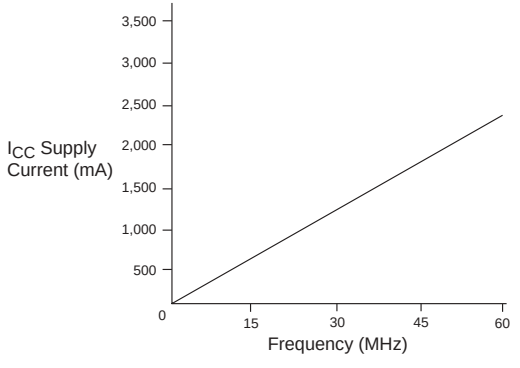
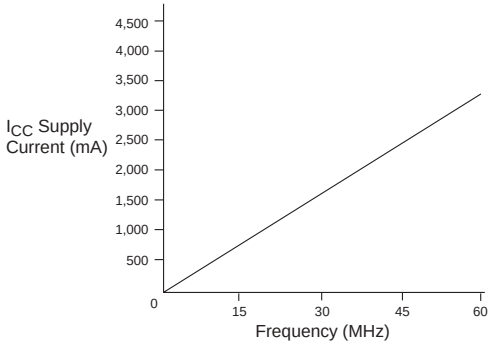
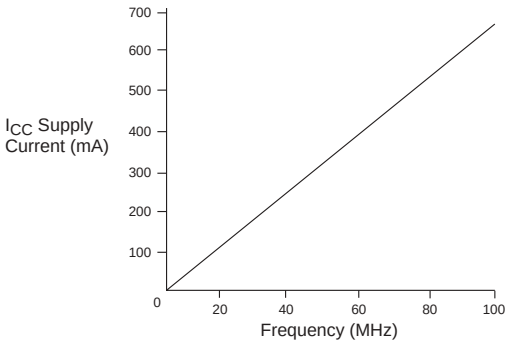


*Figure 32. $I_{CCACTIVE}$ vs. Operating Frequency (Part 2 of 3)*

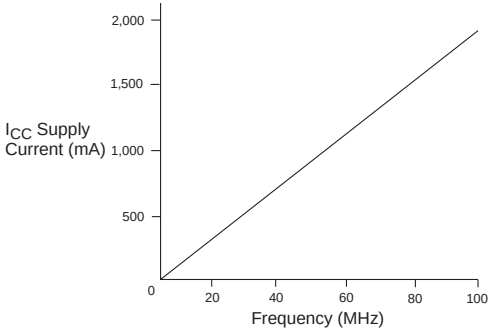
**EPF10K100**



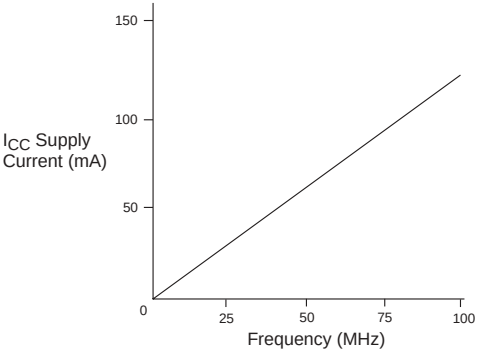

**EPF10K50V**



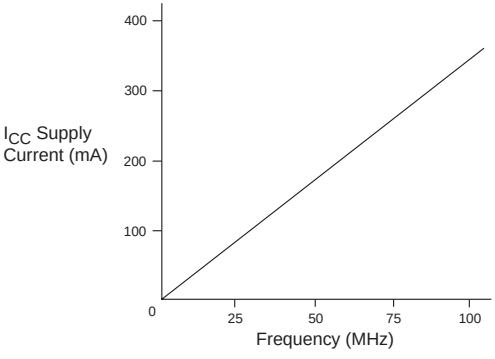

**EPF10K130V**



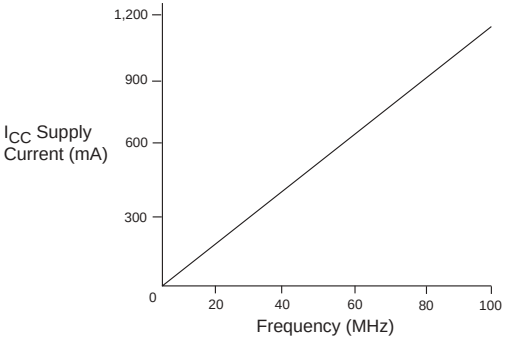

**EPF10K10A**




**EPF10K30A**




**EPF10K100A**



SRAM configuration elements allow FLEX 10K devices to be reconfigured in-circuit by loading new configuration data into the device. Real-time reconfiguration is performed by forcing the device into command mode with a device pin, loading different configuration data, reinitializing the device, and resuming user-mode operation.

The entire reconfiguration process may be completed in less than 320 ms using an EPF10K250A device with a DCLK frequency of 10 MHz. This process can be used to reconfigure an entire system dynamically. In-field upgrades can be performed by distributing new configuration files.

Refer to the configuration device data sheet to obtain the POR delay when using a configuration device method.

## Programming Files

Despite being function- and pin-compatible, FLEX 10KA and FLEX 10KE devices are not programming- or configuration-file compatible with FLEX 10K devices. A design should be recompiled before it is transferred from a FLEX 10K device to an equivalent FLEX 10KA or FLEX 10KE device. This recompilation should be performed to create a new programming or configuration file and to check design timing on the faster FLEX 10KA or FLEX 10KE device. The programming or configuration files for EPF10K50 devices can program or configure an EPF10K50V device. However, Altera recommends recompiling a design for the EPF10K50V device when transferring it from the EPF10K50 device.

## Configuration Schemes

The configuration data for a FLEX 10K device can be loaded with one of five configuration schemes (see Table 116), chosen on the basis of the target application. An EPC1, EPC2, EPC16, or EPC1441 configuration device, intelligent controller, or the JTAG port can be used to control the configuration of a FLEX 10K device, allowing automatic configuration on system power-up.

*Notes:*