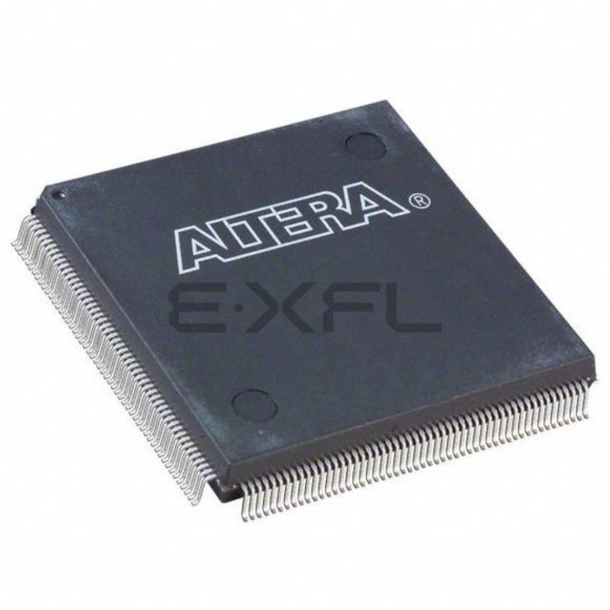



**Welcome to E-XFL.COM**

## 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 | 40960 |
| Number of I/O | 147 |
| Number of Gates | 199000 |
| Voltage - Supply | 2.375V ~ 2.625V |
| Mounting Type | Surface Mount |
| Operating Temperature | 0°C ~ 70°C (TA) |
| Package / Case | 208-BFQFP |
| Supplier Device Package | 208-PQFP (28x28) |
| Purchase URL | https://www.e-xfl.com/product-detail/intel/epf10k50sqc208-1x |

| Table 5. FLEX 10KE Performance | | | | | | |
|---|---|---|---|---|---|---|
| **Application** | **Resources Used** | | **Performance** | | | **Units** |
| | **LEs** | **EABs** | **-1 Speed Grade** | **-2 Speed Grade** | **-3 Speed Grade** | |
| 16-bit loadable counter | 16 | 0 | 285 | 250 | 200 | MHz |
| 16-bit accumulator | 16 | 0 | 285 | 250 | 200 | MHz |
| 16-to-1 multiplexer *(1)* | 10 | 0 | 3.5 | 4.9 | 7.0 | ns |
| 16-bit multiplier with 3-stage pipeline *(2)* | 592 | 0 | 156 | 131 | 93 | MHz |
| 256 × 16 RAM read cycle speed *(2)* | 0 | 1 | 196 | 154 | 118 | MHz |
| 256 × 16 RAM write cycle speed *(2)* | 0 | 1 | 185 | 143 | 106 | MHz |

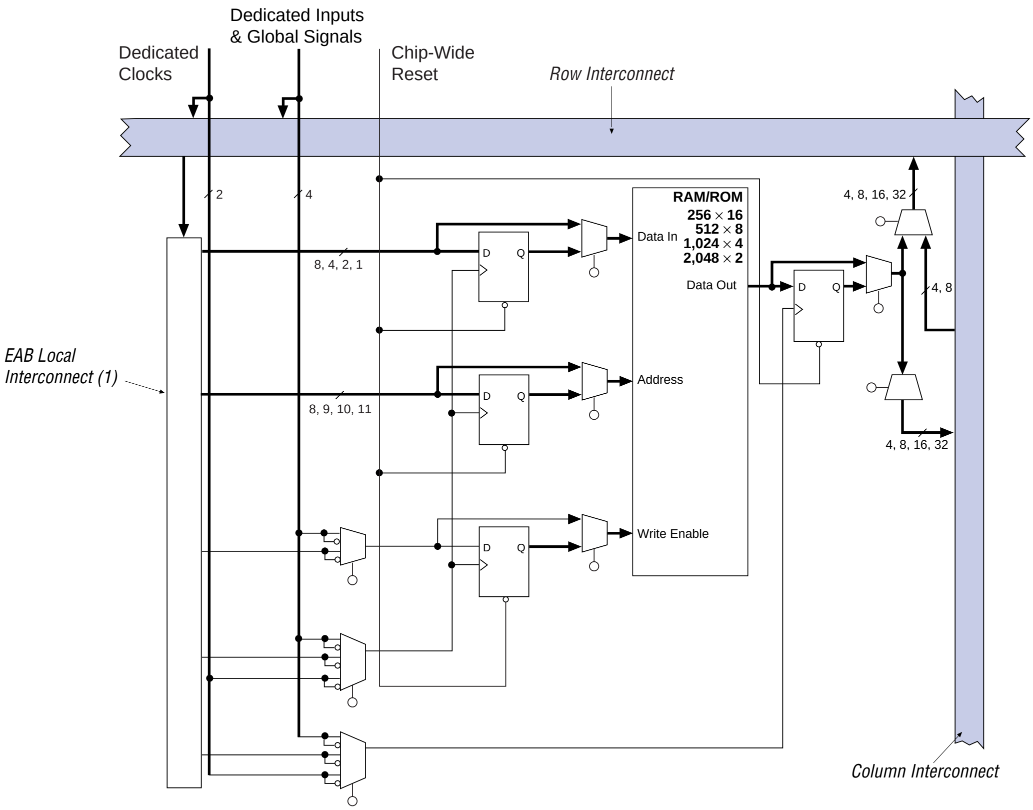
***Notes:***
(1) This application uses combinatorial inputs and outputs.
(2) This application uses registered inputs and outputs.

Table 6 shows FLEX 10KE performance for more complex designs. These designs are available as Altera MegaCore® functions.

| Table 6. FLEX 10KE Performance for Complex Designs | | | | | |
|---|---|---|---|---|---|
| **Application** | **LEs Used** | **Performance** | | | **Units** |
| | | **-1 Speed Grade** | **-2 Speed Grade** | **-3 Speed Grade** | |
| 8-bit, 16-tap parallel finite impulse response (FIR) filter | 597 | 192 | 156 | 116 | MSPS |
| 8-bit, 512-point fast Fourier transform (FFT) function | 1,854 | 23.4 | 28.7 | 38.9 | μs *(1)* |
| | | 113 | 92 | 68 | MHz |
| `a16450` universal asynchronous receiver/transmitter (UART) | 342 | 36 | 28 | 20.5 | MHz |

***Note:***
(1) These values are for calculation time. Calculation time = number of clocks required / $f_{max}$. Number of clocks required = ceiling [log 2 (points) / 2] × [points +14 + ceiling]

***Figure 4. FLEX 10KE Device in Single-Port RAM Mode***

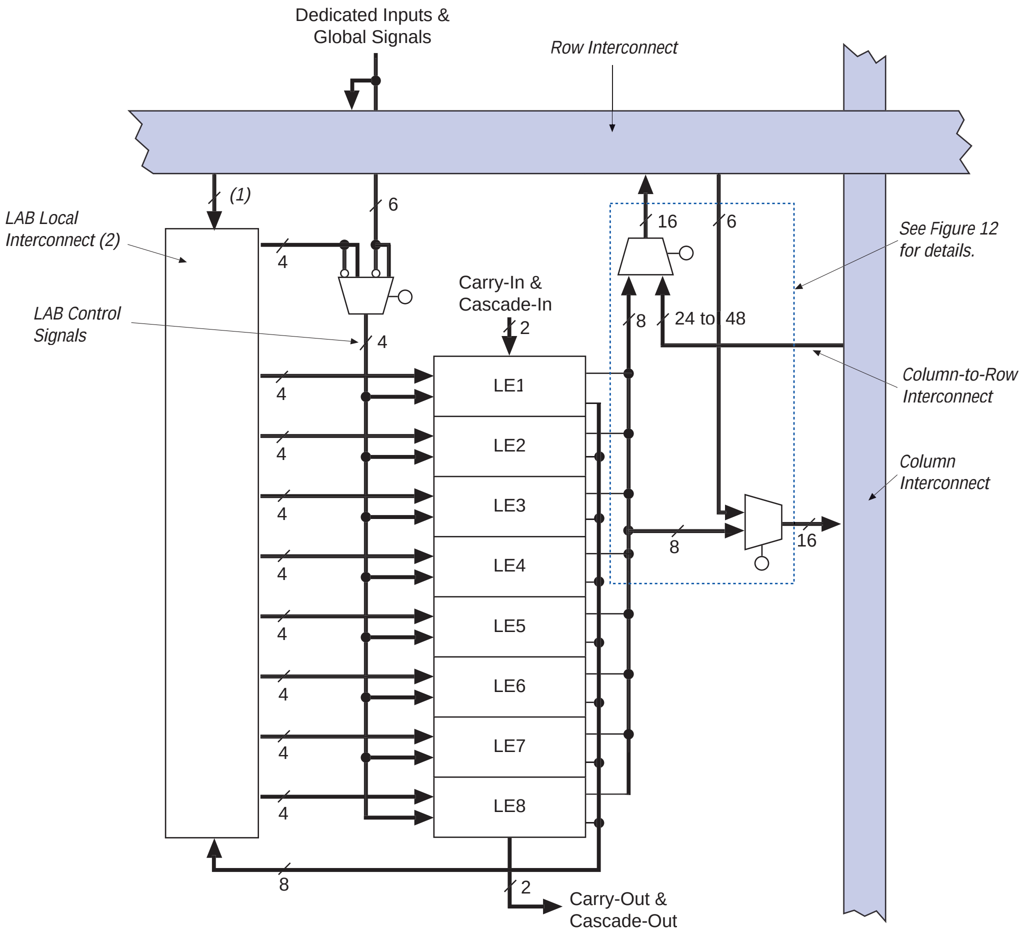



***Note:***

(1) EPF10K30E, EPF10K50E, and EPF10K50S devices have 88 EAB local interconnect channels; EPF10K100E, EPF10K130E, EPF10K200E, and EPF10K200S devices have 104 EAB local interconnect channels.

EABs can be used to implement synchronous RAM, which is easier to use than asynchronous RAM. A circuit using asynchronous RAM must generate the RAM write enable signal, while ensuring that its data and address signals meet setup and hold time specifications relative to the write enable signal. In contrast, the EAB's synchronous RAM generates its own write enable signal and is self-timed with respect to the input or write clock. A circuit using the EAB's self-timed RAM must only meet the setup and hold time specifications of the global clock.

***Figure 7. FLEX 10KE LAB***



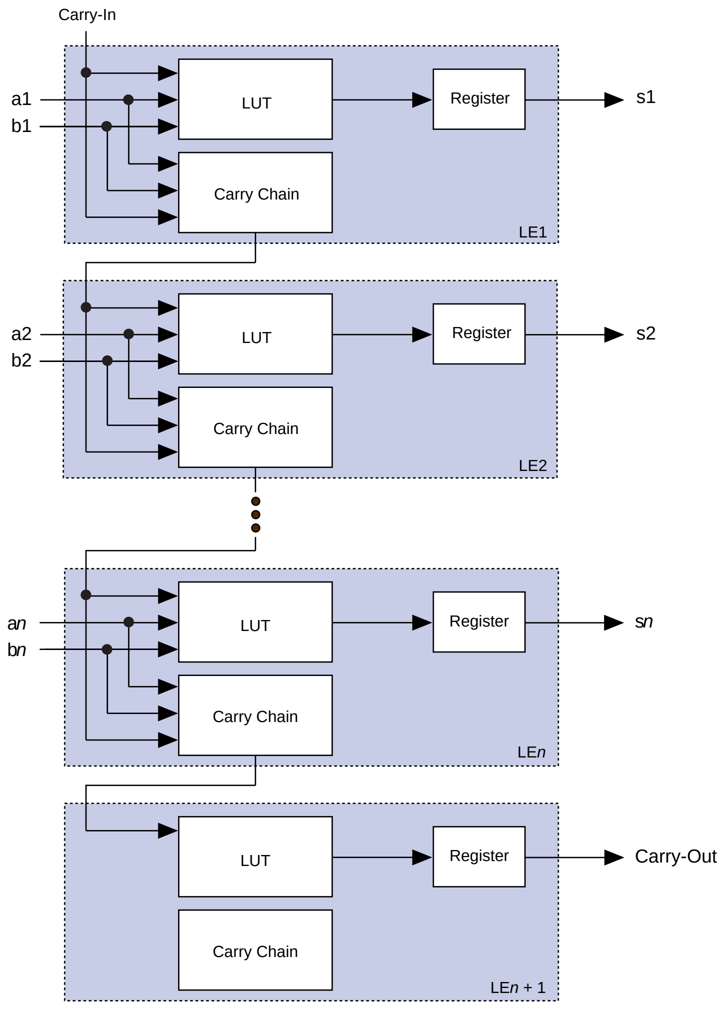

***Notes:***

(1) EPF10K30E, EPF10K50E, and EPF10K50S devices have 22 inputs to the LAB local interconnect channel from the row; EPF10K100E, EPF10K130E, EPF10K200E, and EPF10K200S devices have 26.

(2) EPF10K30E, EPF10K50E, and EPF10K50S devices have 30 LAB local interconnect channels; EPF10K100E, EPF10K130E, EPF10K200E, and EPF10K200S devices have 34.

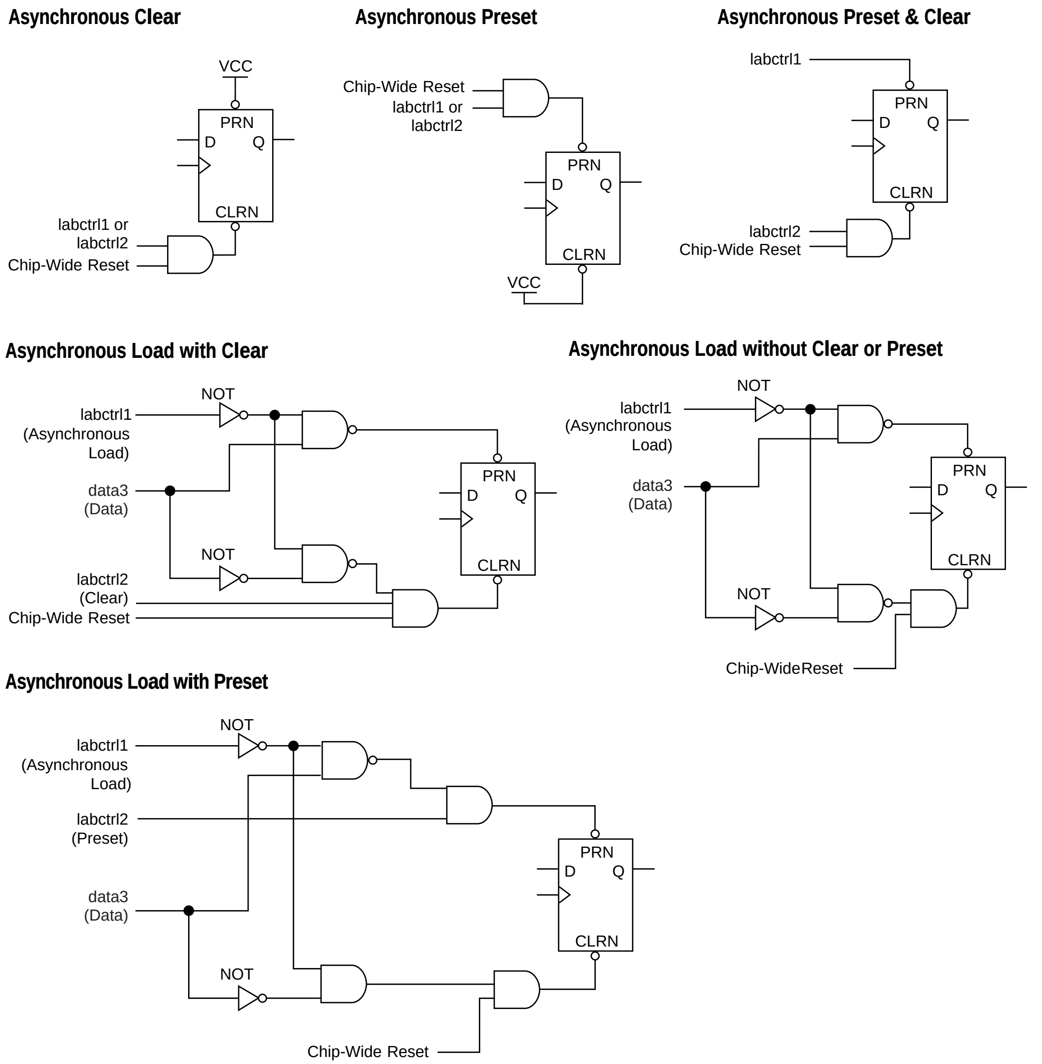
Figure 9 shows how an *n*-bit full adder can be implemented in *n* + 1 LEs with the carry chain. One portion of the LUT generates the sum of two bits using the input signals and the carry-in signal; the sum is routed to the output of the LE. The register can be bypassed for simple adders or used for an accumulator function. Another portion of the LUT and the carry chain logic generates the carry-out signal, which is routed directly to the carry-in signal of the next-higher-order bit. The final carry-out signal is routed to an LE, where it can be used as a general-purpose signal.

***Figure 9. FLEX 10KE Carry Chain Operation (n-Bit Full Adder)***



In addition to the six clear and preset modes, FLEX 10KE devices provide a chip-wide reset pin that can reset all registers in the device. Use of this feature is set during design entry. In any of the clear and preset modes, the chip-wide reset overrides all other signals. Registers with asynchronous presets may be preset when the chip-wide reset is asserted. Inversion can be used to implement the asynchronous preset. Figure 12 shows examples of how to setup the preset and clear inputs for the desired functionality.

*Figure 12. FLEX 10KE LE Clear & Preset Modes*



## FastTrack Interconnect Routing Structure

In the FLEX 10KE architecture, connections between LEs, EABs, and device I/O pins are provided by the FastTrack Interconnect routing structure, which is a series of continuous horizontal and vertical routing channels that traverses 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 routing structure 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 row. The column interconnect routes signals between rows and can drive I/O pins.

Row channels drive into the LAB or EAB local interconnect. The row signal is buffered at every LAB or EAB to reduce the effect of fan-out on delay. 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 a LAB drive the row interconnect.

Each column of LABs or EABs is served by a dedicated column interconnect. The column interconnect that serves the EABs has twice as many channels as other column interconnects. The column interconnect can then drive I/O pins or another row's interconnect to route the signals to other LABs or EABs in the device. A signal from the column interconnect, which can be either the output of a LE or an input from an I/O pin, must be routed to the row interconnect before it can enter a 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, a 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 flexibility enables routing resources to be used more efficiently (see Figure 13).

### *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 16).

**Figure 16. FLEX 10KE Row-to-IOE Connections**

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

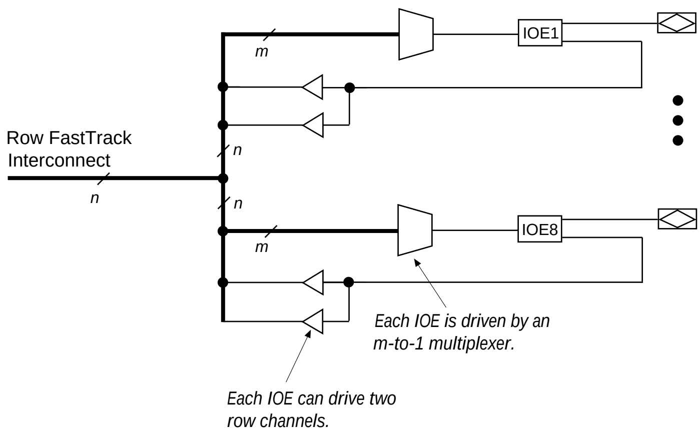



Table 10 lists the FLEX 10KE row-to-IOE interconnect resources.

**Table 10. FLEX 10KE Row-to-IOE Interconnect Resources**

| Device | Channels per Row *(n)* | Row Channels per Pin *(m)* |
|---|---|---|
| EPF10K30E | 216 | 27 |
| EPF10K50E<br>EPF10K50S | 216 | 27 |
| EPF10K100E | 312 | 39 |
| EPF10K130E | 312 | 39 |
| EPF10K200E<br>EPF10K200S | 312 | 39 |

### *Column-to-IOE Connections*

When an IOE is used as an input, it can drive up to two separate column channels. When an IOE is used as an output, the signal is driven by a multiplexer that selects a signal from the column channels. Two IOEs connect to each side of the column channels. Each IOE can be driven by column channels via a multiplexer. The set of column channels is different for each IOE (see Figure 17).

**Figure 17. FLEX 10KE Column-to-IOE Connections**

*The values for m and n are provided in Table 11.*

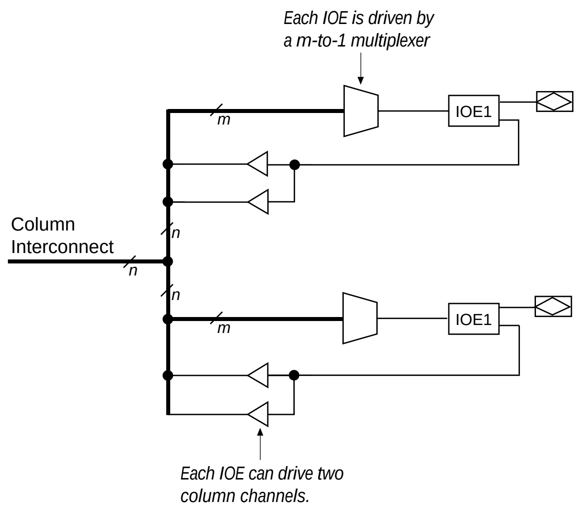



Table 11 lists the FLEX 10KE column-to-IOE interconnect resources.

**Table 11. FLEX 10KE Column-to-IOE Interconnect Resources**

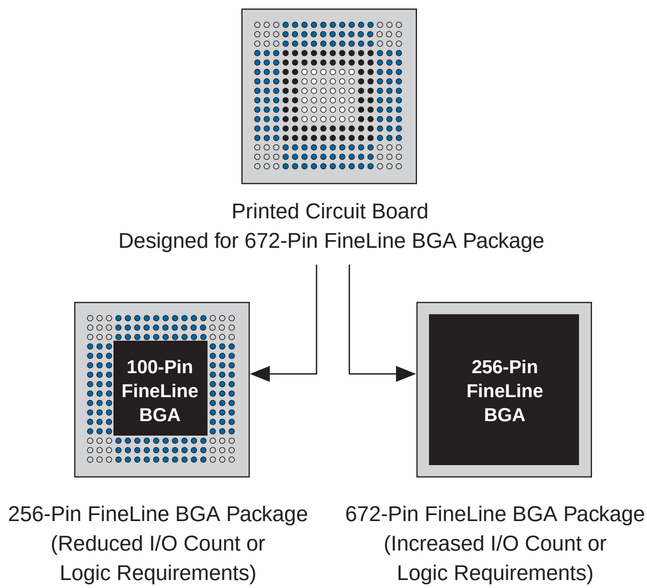
| Device | Channels per Column *(n)* | Column Channels per Pin *(m)* |
|---|---|---|
| EPF10K30E | 24 | 16 |
| EPF10K50E<br>EPF10K50S | 24 | 16 |
| EPF10K100E | 24 | 16 |
| EPF10K130E | 32 | 24 |
| EPF10K200E<br>EPF10K200S | 48 | 40 |

## 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 EPF10K30E device in a 256-pin FineLine BGA package to an EPF10K200S device in a 672-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 18).

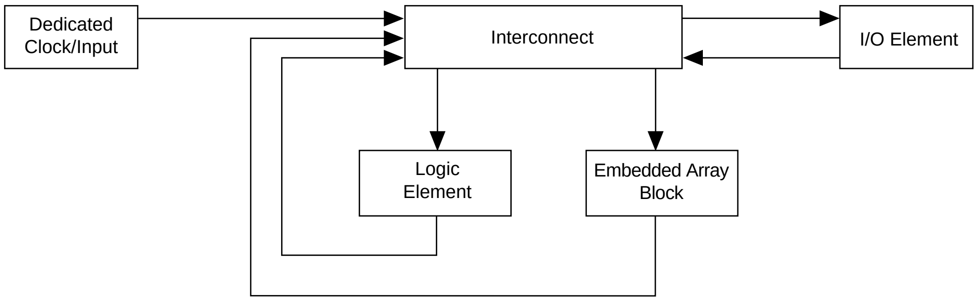
*Figure 18. SameFrame Pin-Out Example*



Timing simulation and delay prediction are available with the Altera Simulator and Timing Analyzer, or with industry-standard EDA tools. The Simulator offers both pre-synthesis functional simulation to evaluate logic design accuracy and post-synthesis timing simulation with 0.1-ns resolution. The Timing Analyzer provides point-to-point timing delay information, setup and hold time analysis, and device-wide performance analysis.

Figure 24 shows the overall timing model, which maps the possible paths to and from the various elements of the FLEX 10KE device.

***Figure 24. FLEX 10KE Device Timing Model***




Figures 25 through 28 show the delays that correspond to various paths and functions within the LE, IOE, EAB, and bidirectional timing models.

*Figure 26. FLEX 10KE Device IOE Timing Model*

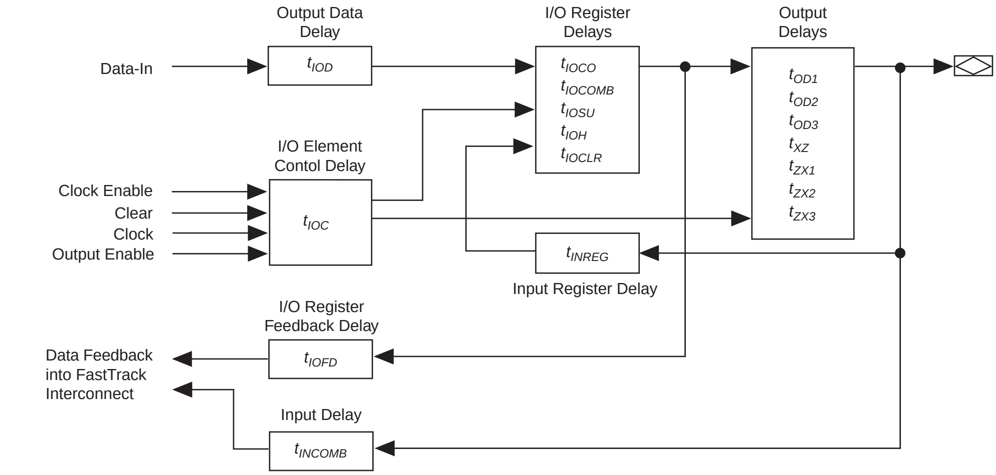



*Figure 27. FLEX 10KE Device EAB Timing Model*

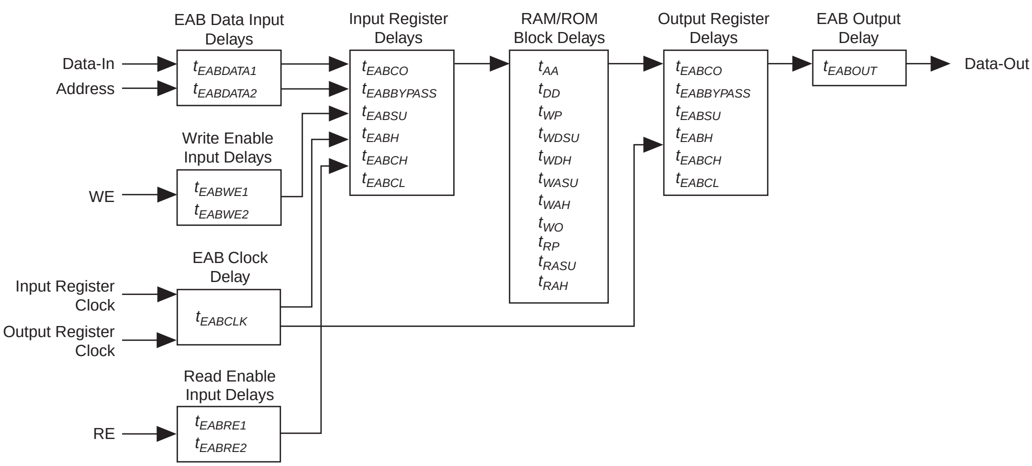
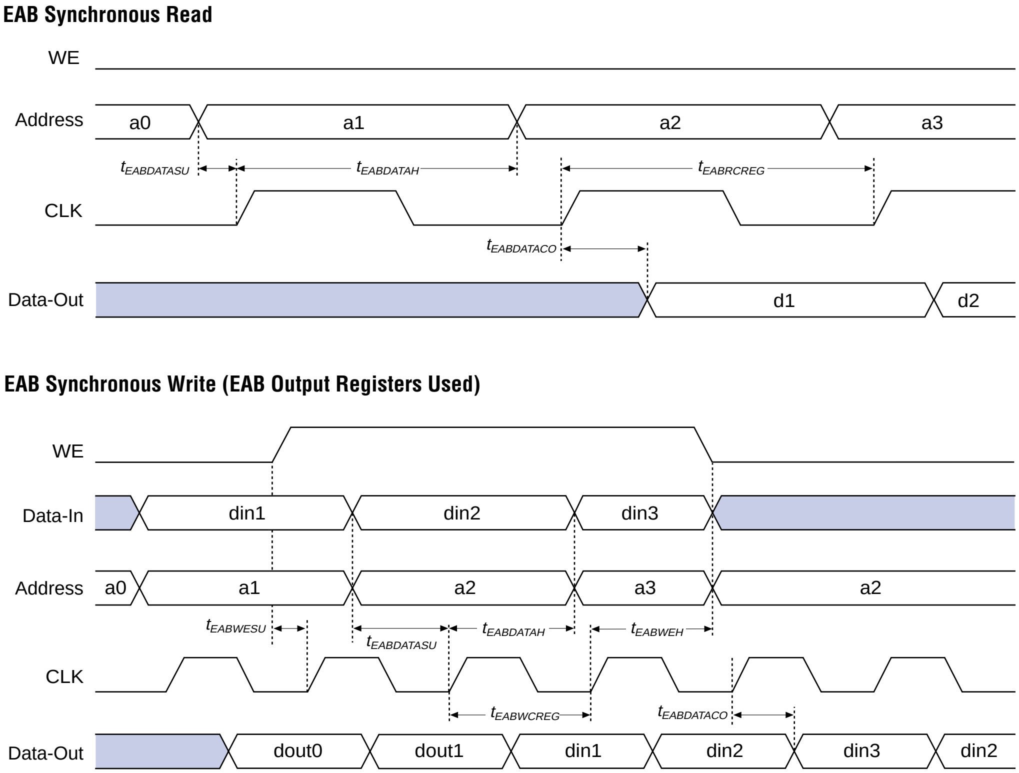


*Figure 30. EAB Synchronous Timing Waveforms*




Tables 31 through 37 show EPF10K30E device internal and external timing parameters.

*Table 31. EPF10K30E Device LE Timing Microparameters (Part 1 of 2) Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{LUT}$ | | 0.7 | | 0.8 | | 1.1 | ns |
| $t_{CLUT}$ | | 0.5 | | 0.6 | | 0.8 | ns |
| $t_{RLUT}$ | | 0.6 | | 0.7 | | 1.0 | ns |
| $t_{PACKED}$ | | 0.3 | | 0.4 | | 0.5 | ns |
| $t_{EN}$ | | 0.6 | | 0.8 | | 1.0 | ns |
| $t_{CICO}$ | | 0.1 | | 0.1 | | 0.2 | ns |
| $t_{CGEN}$ | | 0.4 | | 0.5 | | 0.7 | ns |

**Table 34. EPF10K30E Device EAB Internal Timing Macroparameters** *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{EABAA}$ | | 6.4 | | 7.6 | | 8.8 | ns |
| $t_{EABRCOMB}$ | 6.4 | | 7.6 | | 8.8 | | ns |
| $t_{EABRCREG}$ | 4.4 | | 5.1 | | 6.0 | | ns |
| $t_{EABWP}$ | 2.5 | | 2.9 | | 3.3 | | ns |
| $t_{EABWCOMB}$ | 6.0 | | 7.0 | | 8.0 | | ns |
| $t_{EABWCREG}$ | 6.8 | | 7.8 | | 9.0 | | ns |
| $t_{EABDD}$ | | 5.7 | | 6.7 | | 7.7 | ns |
| $t_{EABDATACO}$ | | 0.8 | | 0.9 | | 1.1 | ns |
| $t_{EABDATASU}$ | 1.5 | | 1.7 | | 2.0 | | ns |
| $t_{EABDATAH}$ | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{EABWESU}$ | 1.3 | | 1.4 | | 1.7 | | ns |
| $t_{EABWEH}$ | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{EABWDSU}$ | 1.5 | | 1.7 | | 2.0 | | ns |
| $t_{EABWDH}$ | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{EABWASU}$ | 3.0 | | 3.6 | | 4.3 | | ns |
| $t_{EABWAH}$ | 0.5 | | 0.5 | | 0.4 | | ns |
| $t_{EABWO}$ | | 5.1 | | 6.0 | | 6.8 | ns |

**Table 47. EPF10K100E Device EAB Internal Microparameters** *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{EABDATA1}$ | | 1.5 | | 2.0 | | 2.6 | ns |
| $t_{EABDATA1}$ | | 0.0 | | 0.0 | | 0.0 | ns |
| $t_{EABWE1}$ | | 1.5 | | 2.0 | | 2.6 | ns |
| $t_{EABWE2}$ | | 0.3 | | 0.4 | | 0.5 | ns |
| $t_{EABRE1}$ | | 0.3 | | 0.4 | | 0.5 | ns |
| $t_{EABRE2}$ | | 0.0 | | 0.0 | | 0.0 | ns |
| $t_{EABCLK}$ | | 0.0 | | 0.0 | | 0.0 | ns |
| $t_{EABCO}$ | | 0.3 | | 0.4 | | 0.5 | ns |
| $t_{EABBYPASS}$ | | 0.1 | | 0.1 | | 0.2 | ns |
| $t_{EABSU}$ | 0.8 | | 1.0 | | 1.4 | | ns |
| $t_{EABH}$ | 0.1 | | 0.1 | | 0.2 | | ns |
| $t_{EABCLR}$ | 0.3 | | 0.4 | | 0.5 | | ns |
| $t_{AA}$ | | 4.0 | | 5.1 | | 6.6 | ns |
| $t_{WP}$ | 2.7 | | 3.5 | | 4.7 | | ns |
| $t_{RP}$ | 1.0 | | 1.3 | | 1.7 | | ns |
| $t_{WDSU}$ | 1.0 | | 1.3 | | 1.7 | | ns |
| $t_{WDH}$ | 0.2 | | 0.2 | | 0.3 | | ns |
| $t_{WASU}$ | 1.6 | | 2.1 | | 2.8 | | ns |
| $t_{WAH}$ | 1.6 | | 2.1 | | 2.8 | | ns |
| $t_{RASU}$ | 3.0 | | 3.9 | | 5.2 | | ns |
| $t_{RAH}$ | 0.1 | | 0.1 | | 0.2 | | ns |
| $t_{WO}$ | | 1.5 | | 2.0 | | 2.6 | ns |
| $t_{DD}$ | | 1.5 | | 2.0 | | 2.6 | ns |
| $t_{EABOUT}$ | | 0.2 | | 0.3 | | 0.3 | ns |
| $t_{EABCH}$ | 1.5 | | 2.0 | | 2.5 | | ns |
| $t_{EABCL}$ | 2.7 | | 3.5 | | 4.7 | | ns |

**Table 48. EPF10K100E Device EAB Internal Timing Macroparameters** *(Part 1 of 2)* *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{EABAA}$ | | 5.9 | | 7.6 | | 9.9 | ns |
| $t_{EABRCOMB}$ | 5.9 | | 7.6 | | 9.9 | | ns |
| $t_{EABRCREG}$ | 5.1 | | 6.5 | | 8.5 | | ns |
| $t_{EABWP}$ | 2.7 | | 3.5 | | 4.7 | | ns |

Tables 52 through 58 show EPF10K130E device internal and external timing parameters.

**Table 52. EPF10K130E Device LE Timing Microparameters** *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{LUT}$ | | 0.6 | | 0.9 | | 1.3 | ns |
| $t_{CLUT}$ | | 0.6 | | 0.8 | | 1.0 | ns |
| $t_{RLUT}$ | | 0.7 | | 0.9 | | 0.2 | ns |
| $t_{PACKED}$ | | 0.3 | | 0.5 | | 0.6 | ns |
| $t_{EN}$ | | 0.2 | | 0.3 | | 0.4 | ns |
| $t_{CICO}$ | | 0.1 | | 0.1 | | 0.2 | ns |
| $t_{CGEN}$ | | 0.4 | | 0.6 | | 0.8 | ns |
| $t_{CGENR}$ | | 0.1 | | 0.1 | | 0.2 | ns |
| $t_{CASC}$ | | 0.6 | | 0.9 | | 1.2 | ns |
| $t_{C}$ | | 0.3 | | 0.5 | | 0.6 | ns |
| $t_{CO}$ | | 0.5 | | 0.7 | | 0.8 | ns |
| $t_{COMB}$ | | 0.3 | | 0.5 | | 0.6 | ns |
| $t_{SU}$ | 0.5 | | 0.7 | | 0.8 | | ns |
| $t_{H}$ | 0.6 | | 0.7 | | 1.0 | | ns |
| $t_{PRE}$ | | 0.9 | | 1.2 | | 1.6 | ns |
| $t_{CLR}$ | | 0.9 | | 1.2 | | 1.6 | ns |
| $t_{CH}$ | 1.5 | | 1.5 | | 2.5 | | ns |
| $t_{CL}$ | 1.5 | | 1.5 | | 2.5 | | ns |

**Table 53. EPF10K130E Device IOE Timing Microparameters** *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{IOD}$ | | 1.3 | | 1.5 | | 2.0 | ns |
| $t_{IOC}$ | | 0.0 | | 0.0 | | 0.0 | ns |
| $t_{IOCO}$ | | 0.6 | | 0.8 | | 1.0 | ns |
| $t_{IOCOMB}$ | | 0.6 | | 0.8 | | 1.0 | ns |
| $t_{IOSU}$ | 1.0 | | 1.2 | | 1.6 | | ns |
| $t_{IOH}$ | 0.9 | | 0.9 | | 1.4 | | ns |
| $t_{IOCLR}$ | | 0.6 | | 0.8 | | 1.0 | ns |
| $t_{OD1}$ | | 2.8 | | 4.1 | | 5.5 | ns |
| $t_{OD2}$ | | 2.8 | | 4.1 | | 5.5 | ns |

**Table 58. EPF10K130E External Bidirectional Timing Parameters** *Notes (1), (2)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{INSUBIDIR}$ *(3)* | 2.2 | | 2.4 | | 3.2 | | ns |
| $t_{INHBIDIR}$ *(3)* | 0.0 | | 0.0 | | 0.0 | | ns |
| $t_{INSUBIDIR}$ *(4)* | 2.8 | | 3.0 | | – | | ns |
| $t_{INHBIDIR}$ *(4)* | 0.0 | | 0.0 | | – | | ns |
| $t_{OUTCOBIDIR}$ *(3)* | 2.0 | 5.0 | 2.0 | 7.0 | 2.0 | 9.2 | ns |
| $t_{XZBIDIR}$ *(3)* | | 5.6 | | 8.1 | | 10.8 | ns |
| $t_{ZXBIDIR}$ *(3)* | | 5.6 | | 8.1 | | 10.8 | ns |
| $t_{OUTCOBIDIR}$ *(4)* | 0.5 | 4.0 | 0.5 | 6.0 | – | – | ns |
| $t_{XZBIDIR}$ *(4)* | | 4.6 | | 7.1 | | – | ns |
| $t_{ZXBIDIR}$ *(4)* | | 4.6 | | 7.1 | | – | ns |

***Notes to tables:***

(1) All timing parameters are described in Tables 24 through 30 in this data sheet.
(2) These parameters are specified by characterization.
(3) This parameter is measured without the use of the ClockLock or ClockBoost circuits.
(4) This parameter is measured with the use of the ClockLock or ClockBoost circuits.

Tables 59 through 65 show EPF10K200E device internal and external timing parameters.

**Table 59. EPF10K200E Device LE Timing Microparameters** *(Part 1 of 2)* *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{LUT}$ | | 0.7 | | 0.8 | | 1.2 | ns |
| $t_{CLUT}$ | | 0.4 | | 0.5 | | 0.6 | ns |
| $t_{RLUT}$ | | 0.6 | | 0.7 | | 0.9 | ns |
| $t_{PACKED}$ | | 0.3 | | 0.5 | | 0.7 | ns |
| $t_{EN}$ | | 0.4 | | 0.5 | | 0.6 | ns |
| $t_{CICO}$ | | 0.2 | | 0.2 | | 0.3 | ns |
| $t_{CGEN}$ | | 0.4 | | 0.4 | | 0.6 | ns |
| $t_{CGENR}$ | | 0.2 | | 0.2 | | 0.3 | ns |
| $t_{CASC}$ | | 0.7 | | 0.8 | | 1.2 | ns |
| $t_C$ | | 0.5 | | 0.6 | | 0.8 | ns |
| $t_{CO}$ | | 0.5 | | 0.6 | | 0.8 | ns |
| $t_{COMB}$ | | 0.4 | | 0.6 | | 0.8 | ns |
| $t_{SU}$ | 0.4 | | 0.6 | | 0.7 | | ns |

**Table 68. EPF10K50S Device EAB Internal Microparameters** *Note (1)*

| Symbol | -1 Speed Grade | | -2 Speed Grade | | -3 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{EABDATA1}$ | | 1.7 | | 2.4 | | 3.2 | ns |
| $t_{EABDATA2}$ | | 0.4 | | 0.6 | | 0.8 | ns |
| $t_{EABWE1}$ | | 1.0 | | 1.4 | | 1.9 | ns |
| $t_{EABWE2}$ | | 0.0 | | 0.0 | | 0.0 | ns |
| $t_{EABRE1}$ | | 0.0 | | 0.0 | | 0.0 | |
| $t_{EABRE2}$ | | 0.4 | | 0.6 | | 0.8 | |
| $t_{EABCLK}$ | | 0.0 | | 0.0 | | 0.0 | ns |
| $t_{EABCO}$ | | 0.8 | | 1.1 | | 1.5 | ns |
| $t_{EABBYPASS}$ | | 0.0 | | 0.0 | | 0.0 | ns |
| $t_{EABSU}$ | 0.7 | | 1.0 | | 1.3 | | ns |
| $t_{EABH}$ | 0.4 | | 0.6 | | 0.8 | | ns |
| $t_{EABCLR}$ | 0.8 | | 1.1 | | 1.5 | | |
| $t_{AA}$ | | 2.0 | | 2.8 | | 3.8 | ns |
| $t_{WP}$ | 2.0 | | 2.8 | | 3.8 | | ns |
| $t_{RP}$ | 1.0 | | 1.4 | | 1.9 | | |
| $t_{WDSU}$ | 0.5 | | 0.7 | | 0.9 | | ns |
| $t_{WDH}$ | 0.1 | | 0.1 | | 0.2 | | ns |
| $t_{WASU}$ | 1.0 | | 1.4 | | 1.9 | | ns |
| $t_{WAH}$ | 1.5 | | 2.1 | | 2.9 | | ns |
| $t_{RASU}$ | 1.5 | | 2.1 | | 2.8 | | |
| $t_{RAH}$ | 0.1 | | 0.1 | | 0.2 | | |
| $t_{WO}$ | | 2.1 | | 2.9 | | 4.0 | ns |
| $t_{DD}$ | | 2.1 | | 2.9 | | 4.0 | ns |
| $t_{EABOUT}$ | | 0.0 | | 0.0 | | 0.0 | ns |
| $t_{EABCH}$ | 1.5 | | 2.0 | | 2.5 | | ns |
| $t_{EABCL}$ | 1.5 | | 2.0 | | 2.5 | | ns |

## Power Consumption

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

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

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 can be calculated with the following equation:

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

Where:

$\mathbf{f_{MAX}}$ = Maximum operating frequency in MHz
N = Total number of LEs used in the device
$\mathbf{tog_{LC}}$ = Average percent of LEs toggling at each clock (typically 12.5%)
K = Constant

Table 80 provides the constant (K) values for FLEX 10KE devices.

*Table 80. FLEX 10KE K Constant Values*

| Device | K Value |
|---|---|
| EPF10K30E | 4.5 |
| EPF10K50E | 4.8 |
| EPF10K50S | 4.5 |
| EPF10K100E | 4.5 |
| EPF10K130E | 4.6 |
| EPF10K200E | 4.8 |
| EPF10K200S | 4.6 |

This calculation provides an $I_{CC}$ estimate based on typical conditions with no output load. The actual $I_{CC}$ should be verified during operation because this measurement is sensitive to the actual pattern in the device and the environmental operating conditions.

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

SRAM configuration elements allow FLEX 10KE 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 requires less than 85 ms and can be used to reconfigure an entire system dynamically. In-field upgrades can be performed by distributing new configuration files.

Before and during configuration, all I/O pins (except dedicated inputs, clock, or configuration pins) are pulled high by a weak pull-up resistor.

## Programming Files

Despite being function- and pin-compatible, FLEX 10KE devices are not programming- or configuration file-compatible with FLEX 10K or FLEX 10KA devices. A design therefore must be recompiled before it is transferred from a FLEX 10K or FLEX 10KA device to an equivalent FLEX 10KE device. This recompilation should be performed both to create a new programming or configuration file and to check design timing in FLEX 10KE devices, which has different timing characteristics than FLEX 10K or FLEX 10KA devices.

FLEX 10KE devices are generally pin-compatible with equivalent FLEX 10KA devices. In some cases, FLEX 10KE devices have fewer I/O pins than the equivalent FLEX 10KA devices. Table 81 shows which FLEX 10KE devices have fewer I/O pins than equivalent FLEX 10KA devices. However, power, ground, JTAG, and configuration pins are the same on FLEX 10KA and FLEX 10KE devices, enabling migration from a FLEX 10KA design to a FLEX 10KE design.

## Device Pin-Outs

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

## Revision History

The information contained in the *FLEX 10KE Embedded Programmable Logic Data Sheet* version 2.5 supersedes information published in previous versions.

### Version 2.5

The following changes were made to the *FLEX 10KE Embedded Programmable Logic Data Sheet* version 2.5:

- *Note (1)* added to Figure 23.
- Text added to "I/O Element" section on page 34.
- Updated Table 22.

### Version 2.4

The following changes were made to the *FLEX 10KE Embedded Programmable Logic Data Sheet* version 2.4: updated text on page 34 and page 63.