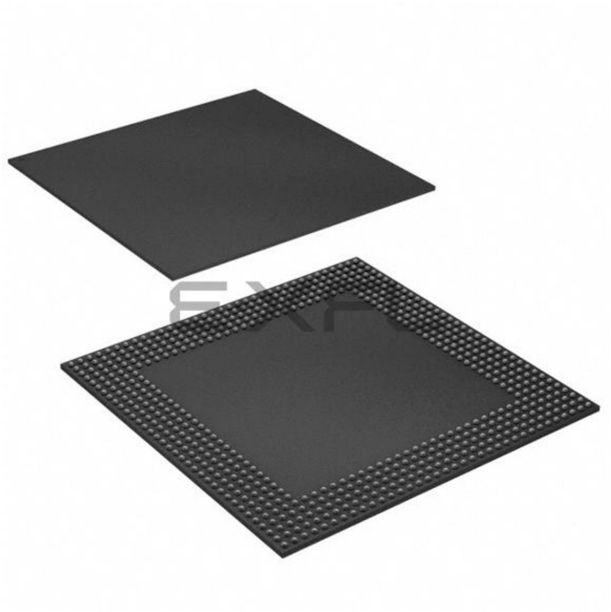
Intel - EPF10K250EBC600-2 Datasheet



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	-
Number of Logic Elements/Cells	-
Total RAM Bits	-
Number of I/O	-
Number of Gates	-
Voltage - Supply	-
Mounting Type	Surface Mount
Operating Temperature	-
Package / Case	600-BGA
Supplier Device Package	600-BGA (45x45)
Purchase URL	https://www.e-xfl.com/product-detail/intel/epf10k250ebc600-2

- Software design support and automatic place-and-route provided by the Altera MAX+PLUS® II development system for 486- and Pentium-based PCs, Sun SPARCstation, HP 9000 Series 700/800, and IBM RISC System/6000 workstations
- Flexible package options
  - Available in a variety of packages with 144 to 672 pins, including the innovative FineLine BGA™ packages (see Tables 3 and 4)
  - SameFrame™ pin-compatibility (with other FLEX 10KE devices) across device densities and pin counts
- Additional design entry and simulation support provided by electronic design interchange format (EDIF) 2 0 0 and 3 0 0 netlist files, library of parameterized modules (LPM), DesignWare components, Verilog HDL, VHDL, and other interfaces to popular electronic design automation (EDA) tools from manufacturers such as Cadence, Exemplar Logic, Mentor Graphics, OrCAD, Synopsys, Synplicity, VeriBest, and Viewlogic

Table 3. FLEX 10KE Package Options & I/O Pin Count *Notes (1), (2), (3)*

Device	144-Pin TQFP	208-Pin PQFP	240-Pin PQFP	256-Pin FineLine βGA	356-Pin βGA	484-Pin FineLine βGA	599-Pin PGA	600-Pin βGA	672-Pin FineLine βGA
EPF10K30E	102	147		176		220			
EPF10K50E	102	147	189	191		254			
EPF10K100B		147	189	191					
EPF10K100E		147	189	191	274	338			
EPF10K130E			186			369			413
EPF10K200E							470	470	470
EPF10K250E							470	470	470

Notes:

(1) Contact Altera Customer Marketing for up-to-date information on package availability.

(2) FLEX 10KE device package types include thin quad flat pack (TQFP), plastic quad flat pack (PQFP), pin-grid array (PGA), and ball-grid array (BGA).

(3) Devices in the same package are pin-compatible, although some devices have more I/O pins than others. When planning device migration, use the I/O pins that are common to all devices. MAX+PLUS II software versions 9.1 and higher provide features to help use only the common pins.

Table 4. FLEX 10KE Package Sizes

Device	144-Pin TQFP	208-Pin PQFP	240-Pin PQFP	256-Pin FineLine βGA	356-Pin βGA	484-Pin FineLine βGA	599-Pin PGA	600-Pin βGA	672-Pin FineLine βGA
Pitch (mm)	0.50	0.50	0.50	1.0	1.27	1.0	–	1.27	1.0
Area (mm^2)	484	936	1,197	289	1,225	529	3,904	2,025	729

FLEX 10KE devices are configured at system power-up with data stored in an Altera serial Configuration EPROM device or provided by a system controller. Altera offers the EPC1, EPC2, and EPC1441 Configuration EPROMs, which configure FLEX 10KE devices via a serial data stream. Configuration data can also be downloaded from system RAM or from the Altera BitBlaster™ serial download cable, ByteBlaster™ parallel port download cable, or ByteBlasterMV™ parallel port download cable. After a FLEX 10KE device has been configured, it can be reconfigured in-circuit by resetting the device and loading new data. Because reconfiguration requires less than 330 ms, real-time changes can be made during system operation.

FLEX 10KE devices contain an optimized interface that permits microprocessors to configure FLEX 10KE devices serially or in parallel, and synchronously or asynchronously. The interface also enables microprocessors to treat a FLEX 10KE device as memory and configure the device by writing to a virtual memory location, making it very easy for the designer to reconfigure the device.

For more information, go to the following documents:

- *Configuration EPROMs for FLEX Devices Data Sheet*
- *BitBlaster Serial Download Cable Data Sheet*
- *ByteBlaster Parallel Port Download Cable Data Sheet*
- *ByteBlasterMV Parallel Port Download Cable Data Sheet*

FLEX 10KE devices are supported by the Altera MAX+PLUS II development system, a single, integrated package that offers schematic, text—including AHDL—and waveform design entry; compilation and logic synthesis; full simulation and worst-case timing analysis; and device configuration. The MAX+PLUS II 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 MAX+PLUS II software works easily with common gate array EDA tools for synthesis and simulation. For example, the MAX+PLUS II software can generate Verilog HDL files for simulation with tools such as Cadence Verilog-XL. Additionally, the MAX+PLUS II 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 MAX+PLUS II development system includes DesignWare functions that are optimized for the FLEX 10KE architecture.

Embedded Array Block

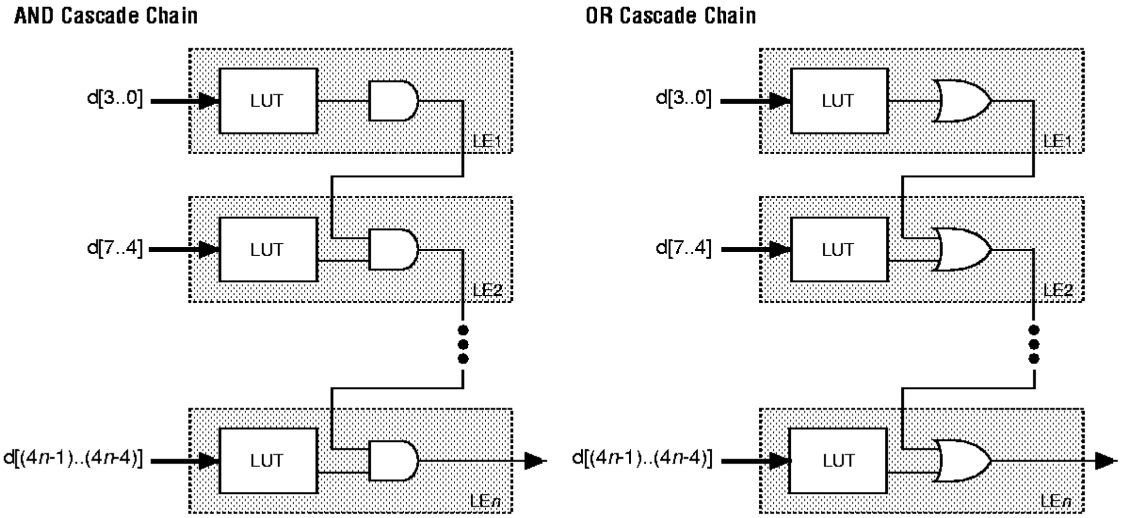
The EAB is a flexible block of RAM with registers on the input and output ports, and is used to implement common gate array megafunctions. Because it is large and flexible, the EAB is suitable for functions such as multipliers, vector scalars, and error correction circuits. These functions can be combined in applications such as digital filters and microcontrollers.

Logic functions are implemented by programming the EAB with a read-only pattern during configuration, thereby creating a large LUT. With LUTs, combinatorial functions are implemented by looking up the results, rather than by computing them. This implementation of combinatorial functions can be faster than using algorithms implemented in general logic, a performance advantage that is further enhanced by the fast access times of EABs. The large capacity of EABs enables designers to implement complex functions in one logic level without the routing delays associated with linked LEs or field-programmable gate array (FPGA) RAM blocks. For example, a single EAB can implement a 5 × 4 multiplier or any function with nine inputs and nine outputs. Parameterized functions such as LPM functions can take advantage of the EAB automatically.

The FLEX 10KE EAB provides advantages over FPGAs which implement on-board RAM as arrays of small, distributed RAM blocks. These small FPGA RAM blocks must be connected together to make usably-sized RAM blocks. The RAM blocks are connected together using multiplexers implemented with more logic blocks. These extra multiplexers cause extra delay, which slows down the RAM block. FPGA RAM blocks are also prone to routing problems because small blocks of RAM must be connected together to make larger blocks. In contrast, EABs can be used to implement large, dedicated blocks of RAM that eliminate these timing and routing concerns.

The FLEX 10KE enhanced EAB adds dual-port capability to the existing EAB structure. The dual-port structure is ideal for FIFO buffers with one or two clocks. The FLEX 10KE EAB can also support up to 16-bit-wide RAM blocks and is backward-compatible with any design containing FLEX 10K EABs. The FLEX 10KE EAB can act in dual-port or single-port mode. When in dual-port mode, separate clocks may be used for EAB read and write sections, which allows the EAB to be written and read at different rates. It also has separate synchronous clock enable signals for the EAB read and write sections, which allow independent control of these sections.

Figure 9. FLEX 10KE Cascade Chain Operation



LE Operating Modes

The FLEX 10KE LE can operate in the following four modes:

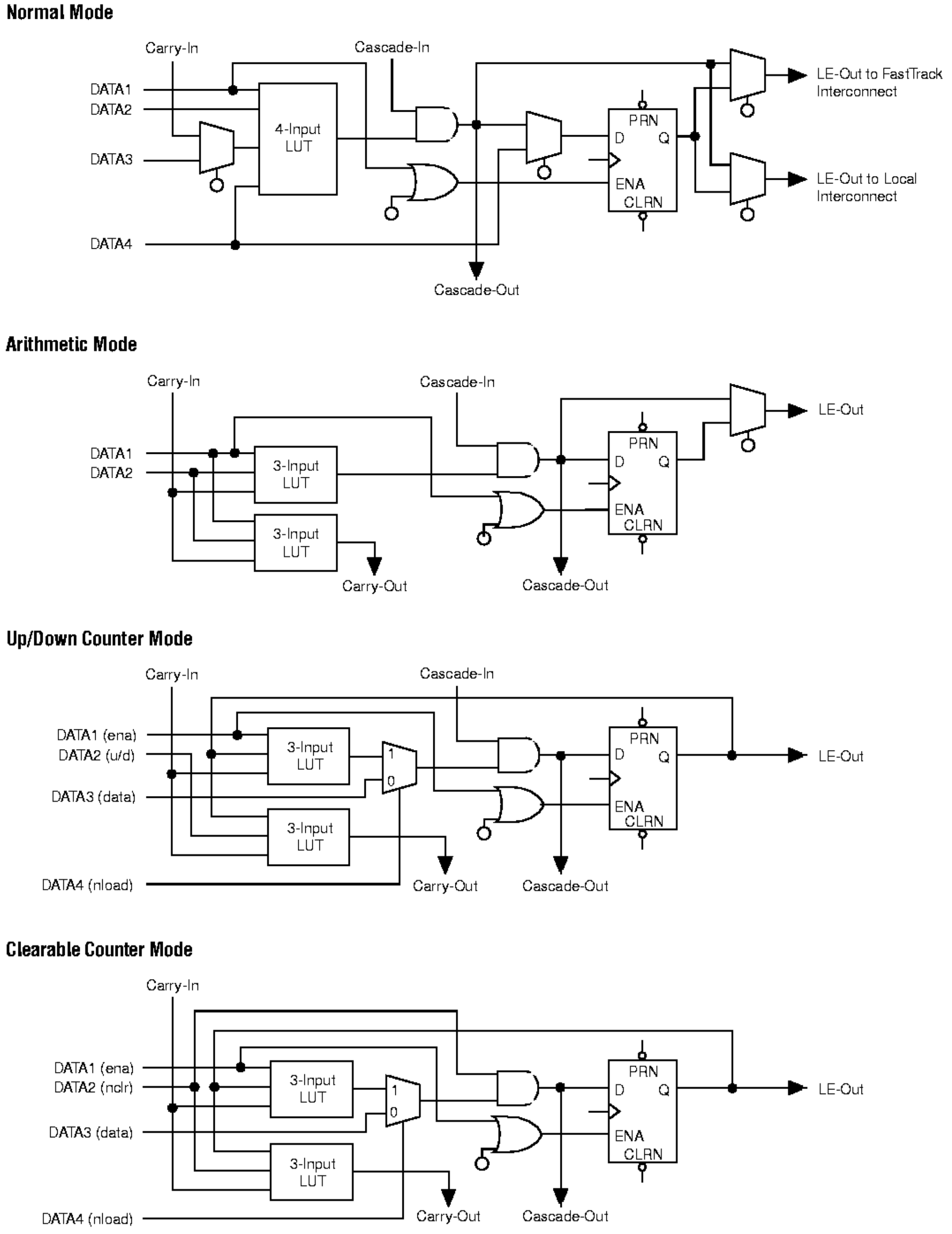
- Normal mode
- Arithmetic mode
- Up/down counter mode
- Clearable counter mode

Each of these modes uses LE resources differently. In each mode, seven available inputs to the LE—the four data inputs from the LAB local interconnect, the feedback from the programmable register, and the carry-in and cascade-in from the previous LE—are directed to different destinations to implement the desired logic function. Three inputs to the LE provide clock, clear, and preset control for the register. The MAX+PLUS II software, in conjunction with parameterized functions such as LPM and DesignWare functions, automatically chooses the appropriate mode for common functions such as counters, adders, and multipliers. If required, the designer can also create special-purpose functions to use an LE operating mode for optimal performance.

The architecture provides a synchronous clock enable to the register in all four modes. The MAX+PLUS II software can set `DATA1` to enable the register synchronously, providing easy implementation of fully synchronous designs.

Figure 10 shows the LE operating modes.

Figure 10. FLEX 10KE LE Operating Modes



Clearable Counter Mode

The clearable counter mode is similar to the up/down counter mode, but supports a synchronous clear instead of the up/down control. The clear function is substituted for the cascade-in signal in the up/down counter mode. Two 3-input LUTs are used: one generates the counter data, and the other generates the fast carry bit. Synchronous loading is provided by a 2-to-1 multiplexer. The output of this multiplexer is ANDed with a synchronous clear signal.

Internal Tri-State Emulation

Internal tri-state emulation provides internal tri-stating without the limitations of a physical tri-state bus. In a physical tri-state bus, the tri-state buffers' output enable (OE) signals select which signal drives the bus. However, if multiple OE signals are active, contending signals can be driven onto the bus. Conversely, if no OE signals are active, the bus will float. Internal tri-state emulation resolves contending tri-state buffers to a low value and floating buses to a high value, thereby eliminating these problems. The MAX+PLUS II software automatically implements tri-state bus functionality with a multiplexer.

Clear & Preset Logic Control

Logic for the programmable register's clear and preset functions is controlled by the DATA3, LABCTRL1, and LABCTRL2 inputs to the LE. The clear and preset control structure of the LE asynchronously loads signals into a register. Either LABCTRL1 or LABCTRL2 can control the asynchronous clear. Alternatively, the register can be set up so that LABCTRL1 implements an asynchronous load. The data to be loaded is driven to DATA3; when LABCTRL1 is asserted, DATA3 is loaded into the register.

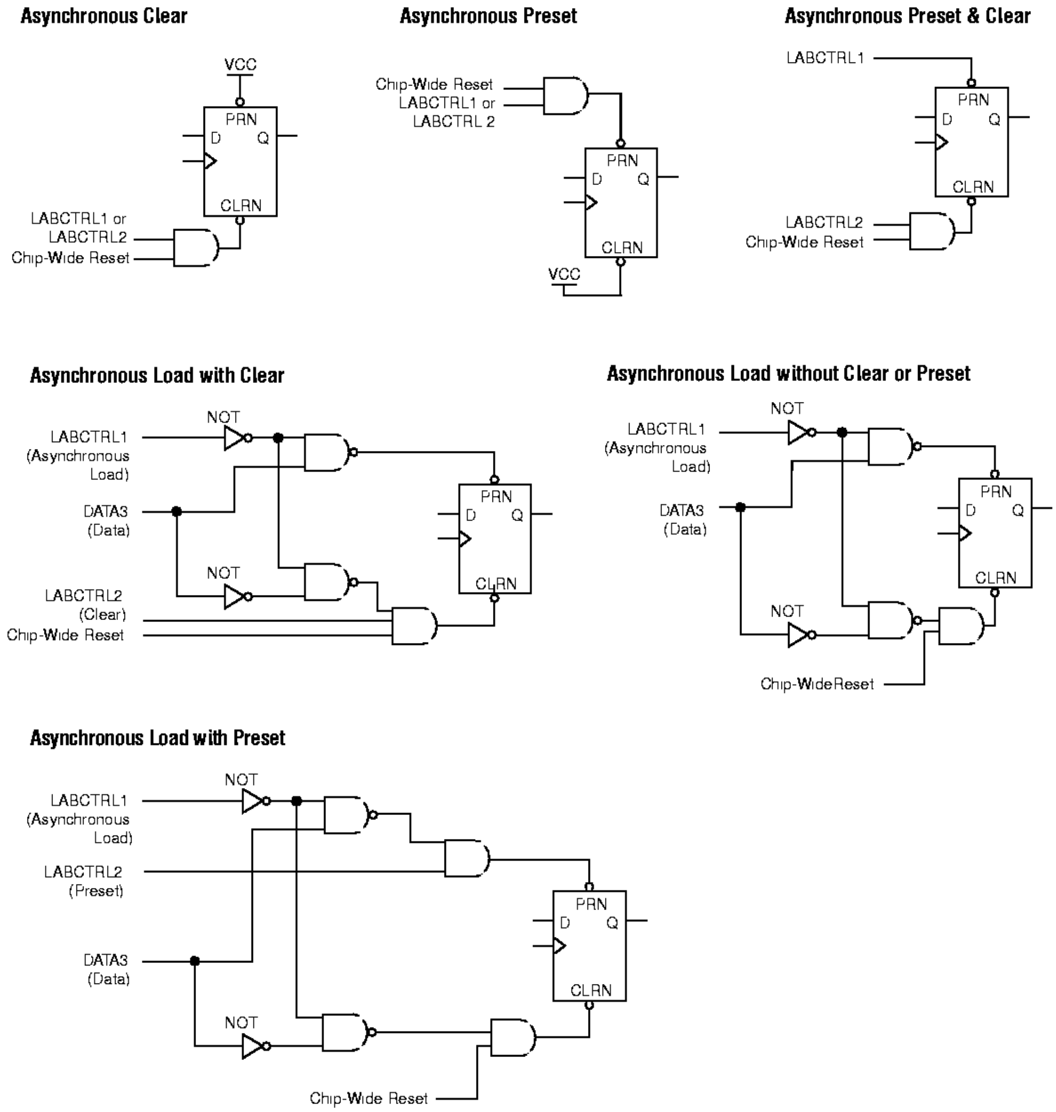
During compilation, the MAX+PLUS II Compiler automatically selects the best control signal implementation. Because the clear and preset functions are active-low, the Compiler automatically assigns a logic high to an unused clear or preset.

The clear and preset logic is implemented in one of the following six modes chosen during design entry:

- Asynchronous clear
- Asynchronous preset
- Asynchronous clear and preset
- Asynchronous load with clear
- Asynchronous load with preset
- Asynchronous load without clear or preset

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 11 shows examples of how to enter a design for the desired functionality.

Figure 11. FLEX 10KE LE Clear & Preset Modes



Asynchronous Clear

Asynchronous Preset

Asynchronous Preset & Clear

Asynchronous Load with Clear

Asynchronous Load without Clear or Preset

Asynchronous Load with Preset

For improved routing, the row interconnect consists of a combination of full-length and half-length channels. The full-length channels connect to all LABs in a row; the half-length channels connect to the LABs in half of the row. The EAB can be driven by the half-length channels in the left half of the row and by the full-length channels. The EAB drives out to the full-length channels. In addition to providing a predictable, row-wide interconnect, this architecture provides increased routing resources. Two neighboring LABs can be connected using a half-row channel, thereby saving the other half of the channel for the other half of the row.

Table 7 summarizes the FastTrack Interconnect routing structure resources available in each FLEX 10KE device.

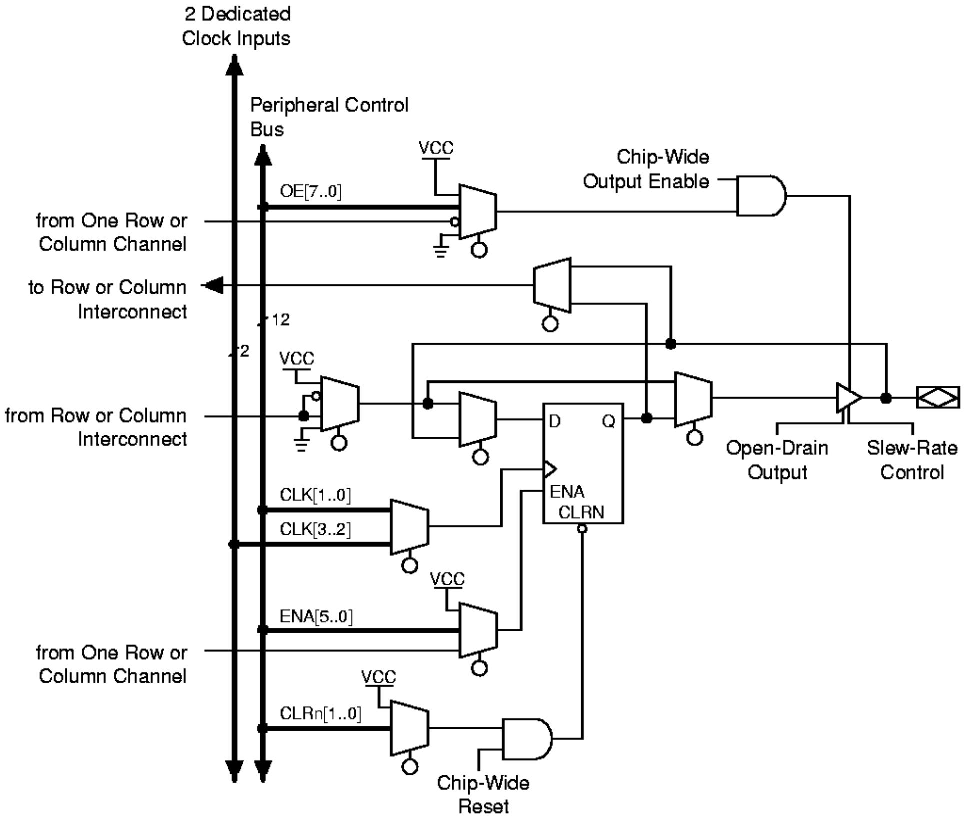
Table 7. FLEX 10KE FastTrack Interconnect Resources

Device	Rows	Channels per Row	Columns	Channels per Column
EPF10K30E	6	216	36	24
EPF10K50E	10	216	36	24
EPF10K100B EPF10K100E	12	312	52	24
EPF10K130E	16	312	52	32
EPF10K200E	24	312	52	48
EPF10K250E	20	456	76	40

In addition to general-purpose I/O pins, FLEX 10KE devices have six dedicated input pins that provide low-skew signal distribution across the device. These six inputs can be used for global clock, clear, preset, and peripheral output enable and clock enable control signals. These signals are available as control signals for all LABs and IOEs in the device. The dedicated inputs can also be used as general-purpose data inputs because they can feed the local interconnect of each LAB in the device.

Figure 13 shows the interconnection of adjacent LABs and EABs, with row, column, and local interconnects, as well as the associated cascade and carry chains. Each LAB is labeled according to its location: a letter represents the row and a number represents the column. For example, LAB B3 is in row B, column 3.

Figure 14. FLEX 10KE I/O Element



Each IOE selects the clock, clear, clock enable, and output enable controls from a network of I/O control signals called the peripheral control bus. The peripheral control bus uses high-speed drivers to minimize signal skew across devices; it provides up to 12 peripheral control signals that can be allocated as follows:

- Up to eight output enable signals
- Up to six clock enable signals
- Up to two clock signals
- Up to two clear signals

Table 9. EPF10K100β, EPF10K100E, EPF10K130E, EPF10K200E & EPF10K250E Peripheral βus Sources

Peripheral Control Signal	EPF10K100β EPF10K100E	EPF10K130E	EPF10K200E	EPF10K250E
OE0	Row A	Row C	Row G	Row E
OE1	Row C	Row E	Row I	Row G
OE2	Row E	Row G	Row K	Row I
OE3	Row L	Row N	Row R	Row P
OE4	Row I	Row K	Row O	Row M
OE5	Row K	Row M	Row Q	Row O
CLKENA0/CLK0/GLOBAL0	Row F	Row H	Row L	Row J
CLKENA1/OE6/GLOBAL1	Row D	Row F	Row J	Row H
CLKENA2/CLR0	Row B	Row D	Row H	Row F
CLKENA3/OE7/GLOBAL2	Row H	Row J	Row N	Row L
CLKENA4/CLR1	Row J	Row L	Row P	Row N
CLKENA5/CLK1/GLOBAL3	Row G	Row I	Row M	Row K

Signals on the peripheral control bus can also drive the four global signals, referred to as GLOBAL0 through GLOBAL3 in Tables 8 and 9. An internally generated signal can drive a global signal, providing the same low-skew, low-delay characteristics as a signal driven by an input pin. An LE drives the global signal by driving a row line that drives the peripheral bus, which then drives the global signal. This feature is ideal for internally generated clear or clock signals with high fan-out. However, internally driven global signals offer no advantage over the general-purpose interconnect for routing data signals.

The chip-wide output enable pin is an active-low pin that can be used to tri-state all pins on the device. This option can be set in the MAX+PLUS II software. The registers in the IOE can also be reset by the chip-wide reset pin.

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)* in the ***1998 Data Book***.

☞ 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 16 provides the constant (K) values for EPF10K50E, EPF10K100B, and EPF10K200E devices. K factors for other FLEX 10KE devices will be released as they become available.

Table 16. FLEX 10KE K Constant Values

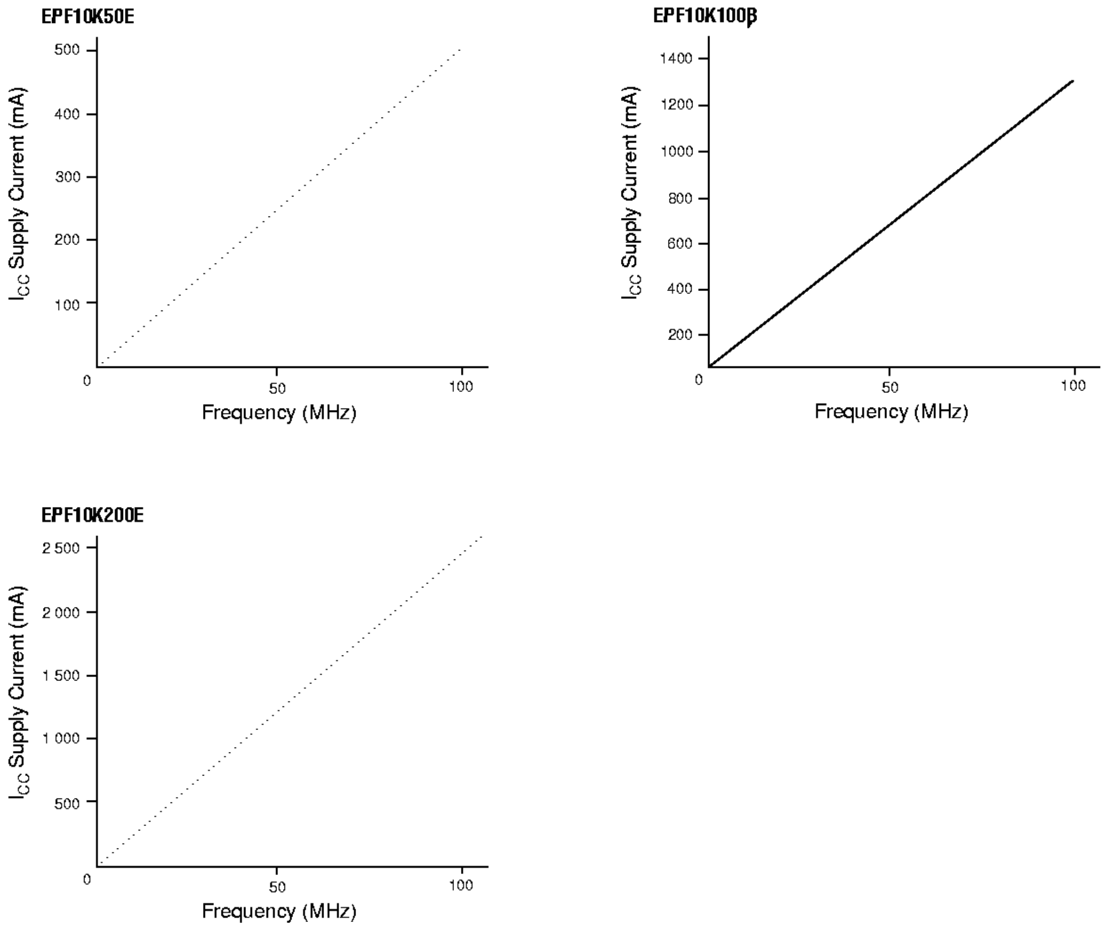
Device	K Value
EPF10K50E	14
EPF10K100B	19
EPF10K200E	19

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.

To better reflect actual designs, the power model (and the constant K in the power calculation equations) for continuous interconnect FLEX devices assume that LEs drive FastTrack Interconnect channels. In contrast, the power model of segmented FPGAs assume that all LEs drive only one short interconnect segment. This assumption may lead to inaccurate results when compared to measured power consumption for actual designs in segmented FPGAs.

Figure 22 shows the relationship between the current and operating frequency of EPF10K50E, EPF10K100B, and EPF10K200E devices. For other FLEX 10KE devices, contact Altera Applications at (800) 800-EPLD.

Figure 22. FLEX 10KE $I_{CCACTIVE}$ vs. Operating Frequency



Configuration & Operation

The FLEX 10KE architecture supports several configuration schemes. This section summarizes the device operating modes and available device configuration schemes.

Operating Modes

The FLEX 10KE architecture uses SRAM configuration elements that require configuration data to be loaded every time the circuit powers up. The process of physically loading the SRAM data into the device is called *configuration*. Before configuration, as V_{CC} rises, the device initiates a Power-On Reset (POR). This POR event clears the device and prepares it for configuration. The FLEX 10KE POR time does not exceed 50 µs; however, when configuring with a Configuration EPROM, the Configuration EPROM imposes a 100 µs delay that allows system power to stabilize before configuration.

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

SRAM configuration elements allow 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 330 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 the faster FLEX 10KE device.

FLEX 10KE devices are generally pin-compatible with the equivalent FLEX 10KA device. In some cases, FLEX 10KE devices have fewer I/O pins than FLEX 10KA devices. Table 17 shows which FLEX 10KE devices have fewer I/Os. 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.

Table 17. I/O Count on FLEX 10KA & FLEX 10KE Devices

FLEX 10KA		FLEX 10KE	
Device	I/O Count	Device	I/O Count
EPF10K30AF256	191	EPF10K30EF256	176
EPF10K30AF484	244	EPF10K30EF484	218
EPF10K30AB356	246	EPF10K30EB356	220
EPF10K50VF484	300	EPF10K50EF484	254
EPF10K50VB356	274	EPF10K50EB356	256
EPF10K100AF484	366	EPF10K100EF484	337
EPF10K130VB600	470	EPF10K130EB600	426

Configuration Schemes

The configuration data for a FLEX 10KE device can be loaded with one of five configuration schemes (see Table 18), chosen on the basis of the target application. An EPC2, EPC1, or EPC1441 Configuration EPROM, intelligent controller, or the JTAG port can be used to control the configuration of a FLEX 10KE device, allowing automatic configuration on system power-up.

Multiple FLEX 10KE devices can be configured in any of the five configuration schemes by connecting the configuration enable (nCE) and configuration enable output (nCEO) pins on each device. Additional FLEX 10K, FLEX 10KA, FLEX 10KE, and FLEX 6000 devices can be configured in the same serial chain.

Table 18. Data Sources for FLEX 10KE Configuration

Configuration Scheme	Data Source
Configuration EPROM	EPC1, EPC2, or EPC1441 Configuration EPROM
Passive serial (PS)	BitBlaster, ByteBlaster, or ByteBlasterMV download cables, or serial data source
Passive parallel asynchronous (PPA)	Parallel data source
Passive parallel synchronous (PPS)	Parallel data source
JTAG	BitBlaster, ByteBlaster, or ByteBlasterMV download cables, or microprocessor with a Jam File

Device Pin-Outs

Table 19 shows the dedicated pin-outs for FLEX 10KE devices in 144-pin TQFP, 208-pin PQFP, 240-pin PQFP, 356-pin BGA, 599-pin PGA, and 600-pin BGA packages.

Table 19. FLEX 10KE Device Pin-Outs (Part 1 of 4) Note (1)

Pin Name	144-Pin TQFP EPF10K30E EPF10K50E	208-Pin PQFP EPF10K30E EPF10K50E EPF10K100E EPF10K100β	240-Pin PQFP EPF10K50E EPF10K100E EPF10K100β	240-Pin PQFP EPF10K130E	356-Pin βGA EPF10K100E	599-Pin PGA EPF10K200E EPF10K250E	600-Pin βGA EPF10K200E EPF10K250E
MSEL0 (2)	77	108	124	124	D4	F6	F5
MSEL1 (2)	76	107	123	123	D3	C3	C1
nSTATUS (2)	35	52	60	60	D24	E43	D32
nCONFIG (2)	74	105	121	121	D2	B4	D4
DCLK (2)	107	155	179	179	AC5	BE5	AP1
CONF_DONE (2)	2	2	2	2	AC24	BC43	AM32
INIT_DONE (3)	14	19	26	26	T24	AM40	AE32
nCE (2)	106	154	178	178	AC2	BB6	AN2
nCEO (2)	3	3	3	3	AC22	BF44	AP35
nWS (4)	142	206	238	238	AE24	BB40	AR29
nRS (4)	141	204	236	236	AE23	BA37	AM28

Table 20 shows the dedicated pin-outs for FLEX 10KE devices in 256-pin FineLine BGA and 672-pin FineLine BGA packages.

Table 20. FLEX 10KE FineLine βGA Device Pin-Outs (Part 1 of 4) Note (1)

Pin Name	256-Pin FineLine βGA EPF10K30E	256-Pin FineLine βGA EPF10K50E EPF10K100E EPF10K100β	672-Pin FineLine βGA EPF10K130E	672-Pin FineLine βGA EPF10K200E EPF10K250E
MSEL0 (2)	P1	P1	W6	W6
MSEL1 (2)	R1	R1	Y6	Y6
nSTATUS (2)	T16	T16	AA21	AA21
nCONFIG (2)	N4	N4	V9	V9
DCLK (2)	B2	B2	G7	G7
CONF_DONE (2)	C15	C15	H20	H20
INIT_DONE (3)	G16	G16	M21	M21
nCE (2)	B1	B1	G6	G6
nCEO (2)	B16	B16	G21	G21
nWS (4)	B14	B14	G19	G19
nRS (4)	C14	C14	H19	H19
nCS (4)	A16	A16	F21	F21
CS (4)	A15	A15	F20	F20
RDYnBSY (4)	G14	G14	M19	M19
CLKUSR (4)	D15	D15	J20	J20
DATA7 (4)	B5	B5	G10	G10
DATA6 (4)	D4	D4	J9	J9
DATA5 (4)	A4	A4	F9	F9
DATA4 (4)	B4	B4	G9	G9
DATA3 (4)	C3	C3	H8	H8
DATA2 (4)	A2	A2	F7	F7
DATA1 (4)	B3	B3	G8	G8
DATA0 (2), (5)	A1	A1	F6	F6
TDI (2)	C2	C2	H7	H7
TDO (2)	C16	C16	H21	H21
TCK (2)	B15	B15	G20	G20
TMS (2)	P15	P15	W20	W20
TRST (2)	R16	R16	Y21	Y21
Dedicated Inputs	B9, E8, M9, R8	B9, E8, M9, R8	Y13, U14, G14, K13	Y13, U14, G14, K13
Dedicated Clock Pins	A9, L8	A9, L8	T13, F14	T13, F14

Table 20. FLEX 10KE FineLine βGA Device Pin-Outs (Part 2 of 4) *Note (1)*

Pin Name	256-Pin FineLine βGA EPF10K30E	256-Pin FineLine βGA EPF10K50E EPF10K100E EPF10K100β	672-Pin FineLine βGA EPF10K130E	672-Pin FineLine βGA EPF10K200E EPF10K250E
DEV_CLRn *(3)*	D8	D8	J13	J13
DEV_OE *(3)*	C9	C9	H14	H14
VCCINT (2.5 V)	E11, F5, F7, F9, F12, H6, H7, H10, J7, J10, J11, K9, L5, L7, L9, L12, M11, R2	E11, F5, F7, F9, F12, H6, H7, H10, J7, J10, J11, K9, L5, L7, L9, L12, M11, R2	E13, E17, H2, H25, K16, L10, L12, L14, L17, M2, M25, N11, N12, N15, P12, P15, P16, R14, T2, T10,T12, T14, T17, T25, U16, Y7, AA23, AB10, AC14	E13, E17, H2, H25, K16, L10, L12, L14, L17, M2, M25, N11, N12, N15, P12, P15, P16, R14, T2, T10,T12, T14, T17, T25, U16, Y7, AA23, AB10, AC14
VCCIO (2.5 or 3.3 V)	D12, E6, F8, F10, G6, G8, G11, H11, J6, K6, K8, K11, L10, M6, N12	D12, E6, F8, F10, G6, G8, G11, H11, J6, K6, K8, K11, L10, M6, N12	C8, C15, D7, G3, J3, J17, K11, K22, L13, L15, M11, M13, M16, M22, N16, P11, R5, R11, R13, R16, R22, T15, U3, U11, V5, V17, V24, Y2, Y24, AA26, AD15	C8, C15, D7, G3, J3, J17, K11, K22, L13, L15, M11, M13, M16, M22, N16, P11, R5, R11, R13, R16, R22, T15, U3, U11, V5, V17, V24, Y2, Y24, AA26, AD15
GND	E5, E12, F6, F11, G7, G9, G10, H8, H9, J8, J9, K7, K10, L6, L11, M5, M12, T8	E5, E12, F6, F11, G7, G9, G10, H8, H9, J8, J9, K7, K10, L6, L11, M5, M12, T8	A2, A25, B2, B25, C3, C10, C24, D3, D4, D19, D23, D24, E4, E23, G23, J5, J23, K4, K10, K17, L11, L16, L22, M5, M12, M14, M15, N13, N14, P13, P14, P22, R12, R15, T11, T16, U10, U17, U24, V3, Y5, AA13, AA22, AB3, AB4, AB5, AB23, AB24, AC3, AC8, AC24, AD13, AD18, AE2, AE25, AF2, AF25	A2, A25, B2, B25, C3, C10, C24, D3, D4, D19, D23, D24, E4, E23, G23, J5, J23, K4, K10, K17, L11, L16, L22, M5, M12, M14, M15, N13, N14, P13, P14, P22, R12, R15, T11, T16, U10, U17, U24, V3, Y5, AA13, AA22, AB3, AB4, AB5, AB23, AB24, AC3, AC8, AC24, AD13, AD18, AE2, AE25, AF2, AF25

Table 20. FLEX 10KE FineLine βGA Device Pin-Outs (Part 4 of 4) *Note (1)*

Pin Name	256-Pin FineLine βGA EPF10K30E	256-Pin FineLine βGA EPF10K50E EPF10K100E EPF10K100β	672-Pin FineLine βGA EPF10K130E	672-Pin FineLine βGA EPF10K200E EPF10K250E
Total User I/O Pins *(7)*	176	191	413	470

Notes:

(1) All pins that are not listed are user I/O pins.
(2) This pin is a dedicated pin; it is not available as a user I/O pin.
(3) This pin can be used as a user I/O pin if it is not used for its device-wide or configuration function.
(4) This pin can be used as a user I/O pin after configuration.
(5) This pin is tri-stated in user mode.
(6) The optional JTAG pin `TRST` is not used in the 144-pin TQFP package.
(7) The user I/O pin count includes dedicated input pins, dedicated clock pins, and all I/O pins.

Table 21 shows the dedicated pin-outs for FLEX 10KE devices in 484-pin FineLine BGA packages.

Table 21. FLEX 10KE FineLine βGA Device Pin-Outs (Part 1 of 5) *Note (1)*

Pin Name	484-Pin FineLine βGA EPF10K30E	484-Pin FineLine βGA EPF10K50E	484-Pin FineLine βGA EPF10K100E	484-Pin FineLine βGA EPF10K130E
`MSEL0` *(2)*	U4	U4	U4	U4
`MSEL1` *(2)*	V4	V4	V4	V4
`nSTATUS` *(2)*	W19	W19	W19	W19
`nCONFIG` *(2)*	T7	T7	T7	T7
`DCLK` *(2)*	E5	E5	E5	E5
`CONF_DONE` *(2)*	F18	F18	F18	F18
`INIT_DONE` *(3)*	K19	K19	K19	K19
`nCE` *(2)*	E4	E4	E4	E4
`nCEO` *(2)*	E19	E19	E19	E19
`nWS` *(4)*	E17	E17	E17	E17
`nRS` *(4)*	F17	F17	F17	F17
`nCS` *(4)*	D19	D19	D19	D19
`CS` *(4)*	D18	D18	D18	D18
`RDYnBSY` *(4)*	K17	K17	K17	K17
`CLKUSR` *(4)*	G18	G18	G18	G18
`DATA7` *(4)*	E8	E8	E8	E8
`DATA6` *(4)*	G7	G7	G7	G7

Table 21. FLEX 10KE FineLine βGA Device Pin-Outs (Part 5 of 5) *Note (1)*				
Pin Name	484-Pin FineLine βGA EPF10K30E	484-Pin FineLine βGA EPF10K50E	484-Pin FineLine βGA EPF10K100E	484-Pin FineLine βGA EPF10K130E
GNDIO	–	–	–	–
Total User I/O Pins *(7)*	220	254	338	369

Notes to Tables:

(1) All pins that are not listed are user I/O pins.
(2) This pin is a dedicated pin; it is not available as a user I/O pin.
(3) This pin can be used as a user I/O pin if it is not used for its device-wide or configuration function.
(4) This pin can be used as a user I/O pin after configuration.
(5) This pin is tri-stated in user mode.
(6) The optional JTAG pin TRST is not used in the 144-pin TQFP package.
(7) The user I/O pin count includes dedicated input pins, dedicated clock pins, and all I/O pins.

Table 22 shows pin compatibility between FLEX 10KE devices.

Table 22. FLEX 10KE Device Pin Compatibility								
Device	144-Pin TQFP	208-Pin PQFP	240-Pin PQFP	256-Pin FineLine βGA	484-Pin FineLine βGA	599-Pin PGA	600-Pin βGA	672-Pin FineLine βGA
EPF10K30E	*Note (1)*	*Note (1)*		*Note (2)*	*Note (2)*			
EPF10K50E	*Note (1)*	*Note (1)*	*Note (2)*	*Note (2)*	*Note (2)*			
EPF10K100B		*Note (1)*	*Note (2)*	*Note (2)*				
EPF10K100E		*Note (1)*	*Note (2)*	*Note (2)*	*Note (2)*			
EPF10K130E			*Note (2)*		*Note (2)*			*Note (2)*
EPF10K200E						*Note (1)*	*Note (1)*	*Note (2)*
EPF10K250E						*Note (1)*	*Note (1)*	*Note (2)*

Notes:

(1) Devices in the same package are pin-compatible and have the same number of I/O pins.
(2) Devices in the same package are pin-compatible, although some devices have more I/O pins than others. When planning device migration, use the I/O pins that are common to all devices. MAX+PLUS II software versions 9.1 and higher provide features to help use only the common pins.

ALTERA.

101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
http://www.altera.com
Applications Hotline:
(800) 800-EPLD
Customer Marketing:
(408) 544-7104
Literature Services:
(888) 3-ALTERA
lit_req@altera.com

Altera, MAX, MAX+PLUS, MAX+PLUS II, FLEX, Jam, ClockLock, ClockBoost, SameFrame, FLEX 10K, FLEX 10KA, FLEX 10KE, EPF10K30E, EPF10K50E, EPF10K100B, EPF10K100E, EPF10K130E, EPF10K200E, EPF10K250E, FLEX 6000, MultiVolt, EPC2, EPC1, EPC1441, MegaCore, BitBlaster, ByteBlaster, ByteBlasterMV, FastTrack Interconnect, and FineLine BGA are trademarks and/or service marks of Altera Corporation in the United States and other countries. Altera acknowledges the trademarks of other organizations for their respective products or services mentioned in this document. Altera products are protected under numerous U.S. and foreign patents and pending applications, maskwork rights, and copyrights. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera Corporation. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

I.S. EN ISO 9001