



Welcome to [E-XFL.COM](https://www.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	624
Number of Logic Elements/Cells	4992
Total RAM Bits	49152
Number of I/O	191
Number of Gates	257000
Voltage - Supply	2.375V ~ 2.625V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 70°C (TA)
Package / Case	256-BGA
Supplier Device Package	256-FBGA (17x17)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/intel/epf10k100efc256-2aa">https://www.e-xfl.com/product-detail/intel/epf10k100efc256-2aa</a>



For more information on FLEX device configuration, 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*
- *MasterBlaster Download Cable Data Sheet*
- *Application Note 116 (Configuring APEX 20K, FLEX 10K, & FLEX 6000 Devices)*

FLEX 10KE devices are supported by the Altera development systems, which are 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 system includes DesignWare functions that are optimized for the FLEX 10KE architecture.

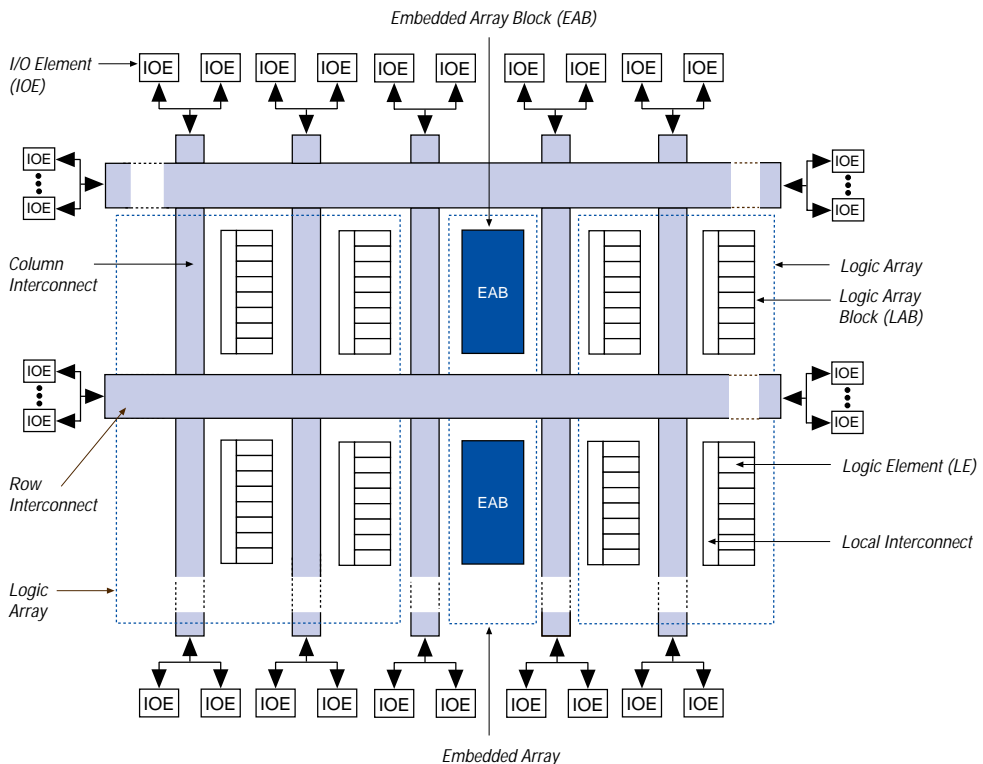
The Altera development system runs on Windows-based PCs and Sun SPARCstation, and HP 9000 Series 700/800.



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

Figure 1 shows a block diagram of the FLEX 10KE architecture. Each group of LEs is combined into an LAB; groups of LABs are arranged into rows and columns. Each row also contains a single EAB. The LABs and EABs are interconnected by the FastTrack Interconnect routing structure. IOEs are located at the end of each row and column of the FastTrack Interconnect routing structure.

Figure 1. FLEX 10KE Device Block Diagram



FLEX 10KE devices provide six dedicated inputs that drive the flipflops' control inputs and ensure the efficient distribution of high-speed, low-skew (less than 1.5 ns) control signals. These signals use dedicated routing channels that provide shorter delays and lower skews than the FastTrack Interconnect routing structure. Four of the dedicated inputs drive four global signals. These four global signals can also be driven by internal logic, providing an ideal solution for a clock divider or an internally generated asynchronous clear signal that clears many registers in the device.

The programmable flipflop in the LE can be configured for D, T, JK, or SR operation. The clock, clear, and preset control signals on the flipflop can be driven by global signals, general-purpose I/O pins, or any internal logic. For combinatorial functions, the flipflop is bypassed and the output of the LUT drives the output of the LE.

The LE has two outputs that drive the interconnect: one drives the local interconnect and the other drives either the row or column FastTrack Interconnect routing structure. The two outputs can be controlled independently. For example, the LUT can drive one output while the register drives the other output. This feature, called register packing, can improve LE utilization because the register and the LUT can be used for unrelated functions.

The FLEX 10KE architecture provides two types of dedicated high-speed data paths that connect adjacent LEs without using local interconnect paths: carry chains and cascade chains. The carry chain supports high-speed counters and adders and the cascade chain implements wide-input functions with minimum delay. Carry and cascade chains connect all LEs in a LAB as well as all LABs in the same row. Intensive use of carry and cascade chains can reduce routing flexibility. Therefore, the use of these chains should be limited to speed-critical portions of a design.

### *Carry Chain*

The carry chain provides a very fast (as low as 0.2 ns) carry-forward function between LEs. The carry-in signal from a lower-order bit drives forward into the higher-order bit via the carry chain, and feeds into both the LUT and the next portion of the carry chain. This feature allows the FLEX 10KE architecture to implement high-speed counters, adders, and comparators of arbitrary width efficiently. Carry chain logic can be created automatically by the Altera Compiler during design processing, or manually by the designer during design entry. Parameterized functions such as LPM and DesignWare functions automatically take advantage of carry chains.

Carry chains longer than eight LEs are automatically implemented by linking LABs together. For enhanced fitting, a long carry chain skips alternate LABs in a row. A carry chain longer than one LAB skips either from even-numbered LAB to even-numbered LAB, or from odd-numbered LAB to odd-numbered LAB. For example, the last LE of the first LAB in a row carries to the first LE of the third LAB in the row. The carry chain does not cross the EAB at the middle of the row. For instance, in the EPF10K50E device, the carry chain stops at the eighteenth LAB and a new one begins at the nineteenth LAB.

*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 Altera 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 that use a specific LE operating mode for optimal performance.

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

### Normal Mode

The normal mode is suitable for general logic applications and wide decoding functions that can take advantage of a cascade chain. In normal mode, four data inputs from the LAB local interconnect and the carry-in are inputs to a four-input LUT. The Altera Compiler automatically selects the carry-in or the `DATA3` signal as one of the inputs to the LUT. The LUT output can be combined with the cascade-in signal to form a cascade chain through the cascade-out signal. Either the register or the LUT can be used to drive both the local interconnect and the FastTrack Interconnect routing structure at the same time.

The LUT and the register in the LE can be used independently (register packing). To support register packing, the LE has two outputs; one drives the local interconnect, and the other drives the FastTrack Interconnect routing structure. The `DATA4` signal can drive the register directly, allowing the LUT to compute a function that is independent of the registered signal; a three-input function can be computed in the LUT, and a fourth independent signal can be registered. Alternatively, a four-input function can be generated, and one of the inputs to this function can be used to drive the register. The register in a packed LE can still use the clock enable, clear, and preset signals in the LE. In a packed LE, the register can drive the FastTrack Interconnect routing structure while the LUT drives the local interconnect, or vice versa.

### Arithmetic Mode

The arithmetic mode offers 2 three-input LUTs that are ideal for implementing adders, accumulators, and comparators. One LUT computes a three-input function; the other generates a carry output. As shown in [Figure 11](#) on [page 22](#), the first LUT uses the carry-in signal and two data inputs from the LAB local interconnect to generate a combinatorial or registered output. For example, in an adder, this output is the sum of three signals: `a`, `b`, and carry-in. The second LUT uses the same three signals to generate a carry-out signal, thereby creating a carry chain. The arithmetic mode also supports simultaneous use of the cascade chain.

### Up/Down Counter Mode

The up/down counter mode offers counter enable, clock enable, synchronous up/down control, and data loading options. These control signals are generated by the data inputs from the LAB local interconnect, the carry-in signal, and output feedback from the programmable register. Use 2 three-input LUTs: one generates the counter data, and the other generates the fast carry bit. A 2-to-1 multiplexer provides synchronous loading. Data can also be loaded asynchronously with the clear and preset register control signals without using the LUT resources.

Figure 13. FLEX 10KE LAB Connections to Row & Column Interconnect

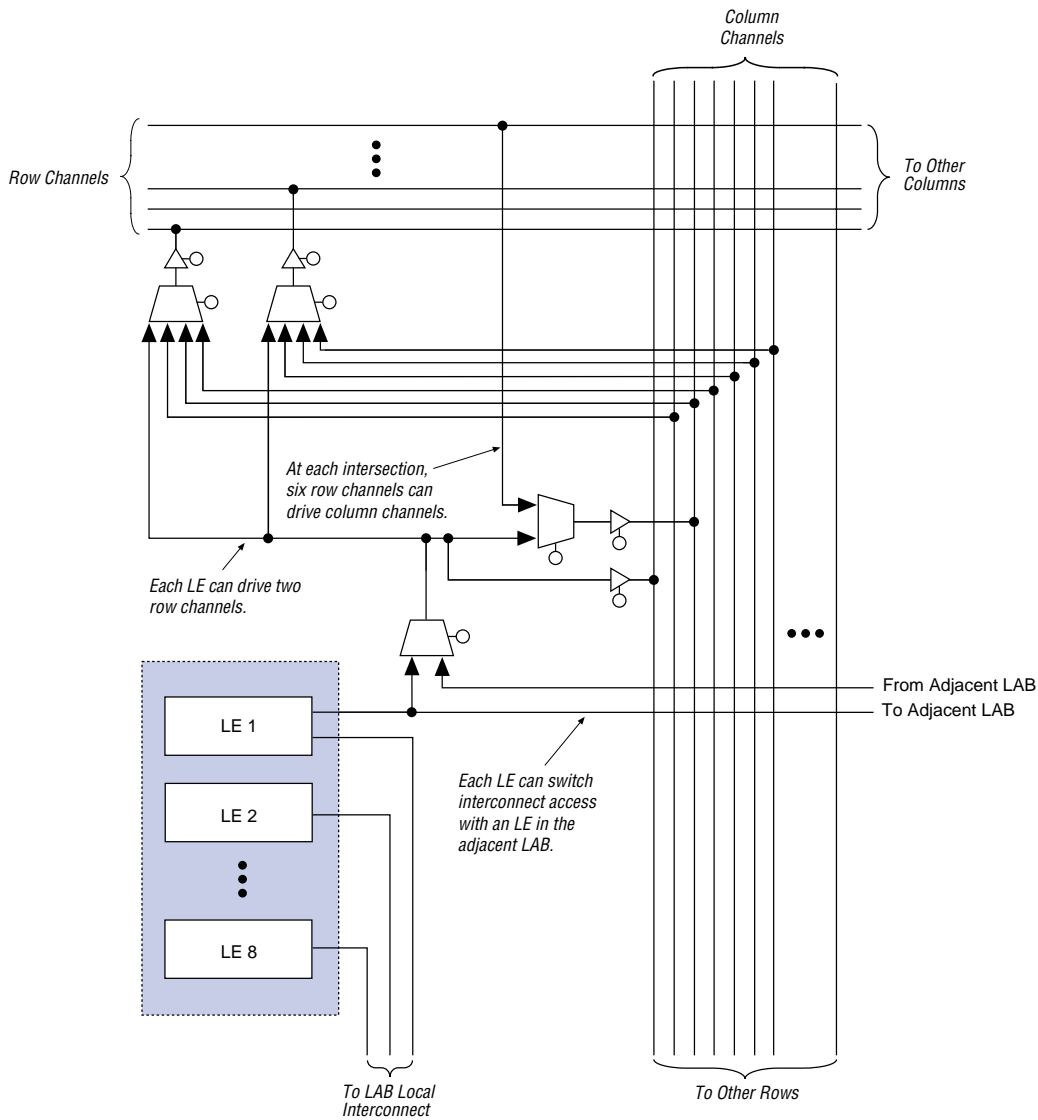
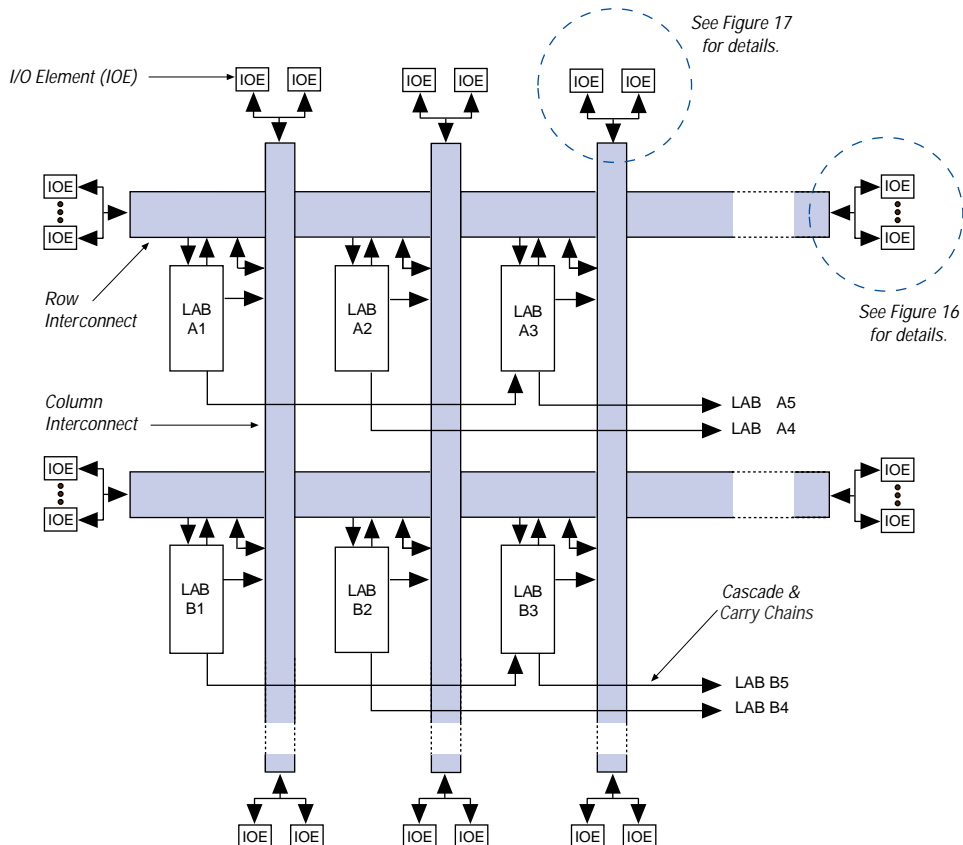


Figure 14. FLEX 10KE Interconnect Resources



## I/O Element

An IOE contains a bidirectional I/O buffer and a register that can be used either as an input register for external data that requires a fast setup time, or as an output register for data that requires fast clock-to-output performance. In some cases, using an LE register for an input register will result in a faster setup time than using an IOE register. IOEs can be used as input, output, or bidirectional pins. For bidirectional registered I/O implementation, the output register should be in the IOE, and the data input and output enable registers should be LE registers placed adjacent to the bidirectional pin. The Altera Compiler uses the programmable inversion option to invert signals from the row and column interconnect automatically where appropriate. [Figure 15](#) shows the bidirectional I/O registers.



When dedicated inputs drive non-inverted and inverted peripheral clears, clock enables, and output enables, two signals on the peripheral control bus will be used.

Tables 8 and 9 list the sources for each peripheral control signal, and show how the output enable, clock enable, clock, and clear signals share 12 peripheral control signals. The tables also show the rows that can drive global signals.

**Table 8. Peripheral Bus Sources for EPF10K30E, EPF10K50E & EPF10K50S Devices**

Peripheral Control Signal	EPF10K30E	EPF10K50E EPF10K50S
OE0	Row A	Row A
OE1	Row B	Row B
OE2	Row C	Row D
OE3	Row D	Row F
OE4	Row E	Row H
OE5	Row F	Row J
CLKENA0/CLK0/GLOBAL0	Row A	Row A
CLKENA1/OE6/GLOBAL1	Row B	Row C
CLKENA2/CLR0	Row C	Row E
CLKENA3/OE7/GLOBAL2	Row D	Row G
CLKENA4/CLR1	Row E	Row I
CLKENA5/CLK1/GLOBAL3	Row F	Row J

### 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.

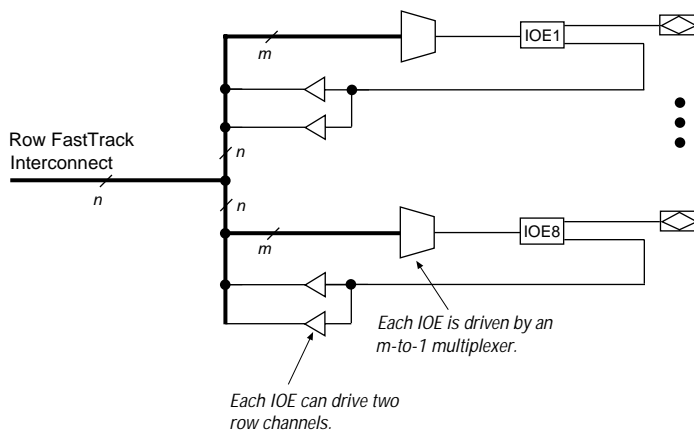


Table 10 lists the FLEX 10KE row-to-IOE interconnect resources.

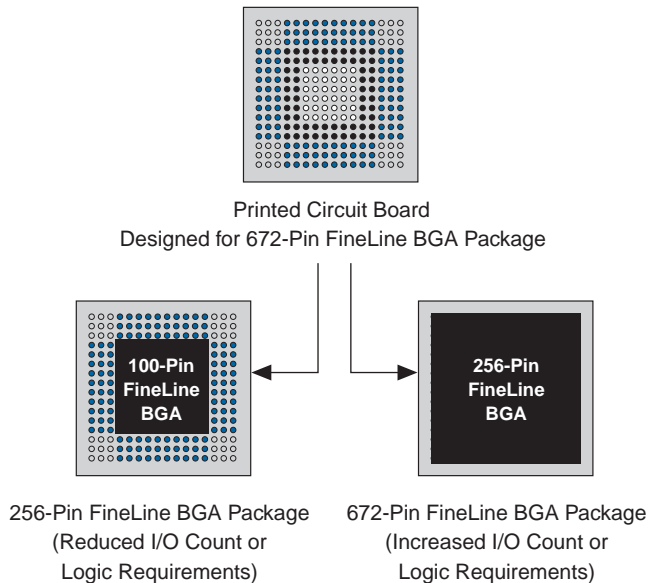
Table 10. FLEX 10KE Row-to-IOE Interconnect Resources		
Device	Channels per Row ( $n$ )	Row Channels per Pin ( $m$ )
EPF10K30E	216	27
EPF10K50E EPF10K50S	216	27
EPF10K100E	312	39
EPF10K130E	312	39
EPF10K200E EPF10K200S	312	39

## 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*



## PCI Pull-Up Clamping Diode Option

FLEX 10KE devices have a pull-up clamping diode on every I/O, dedicated input, and dedicated clock pin. PCI clamping diodes clamp the signal to the  $V_{CCIO}$  value and are required for 3.3-V PCI compliance. Clamping diodes can also be used to limit overshoot in other systems.

Clamping diodes are controlled on a pin-by-pin basis. When  $V_{CCIO}$  is 3.3 V, a pin that has the clamping diode option turned on can be driven by a 2.5-V or 3.3-V signal, but not a 5.0-V signal. When  $V_{CCIO}$  is 2.5 V, a pin that has the clamping diode option turned on can be driven by a 2.5-V signal, but not a 3.3-V or 5.0-V signal. Additionally, a clamping diode can be activated for a subset of pins, which would allow a device to bridge between a 3.3-V PCI bus and a 5.0-V device.

## Slew-Rate Control

The output buffer in each IOE has an adjustable output slew rate that can be configured for low-noise or high-speed performance. A slower slew rate reduces system noise and adds a maximum delay of 4.3 ns. The fast slew rate should be used for speed-critical outputs in systems that are adequately protected against noise. Designers can specify the slew rate pin-by-pin or assign a default slew rate to all pins on a device-wide basis. The slow slew rate setting affects the falling edge of the output.

## Open-Drain Output Option

FLEX 10KE devices provide an optional open-drain output (electrically equivalent to open-collector output) for each I/O pin. This open-drain output enables the device to provide system-level control signals (e.g., interrupt and write enable signals) that can be asserted by any of several devices. It can also provide an additional wired-OR plane.

## MultiVolt I/O Interface

The FLEX 10KE device architecture supports the MultiVolt I/O interface feature, which allows FLEX 10KE devices in all packages to interface with systems of differing supply voltages. These devices have one set of  $V_{CC}$  pins for internal operation and input buffers ( $V_{CCINT}$ ), and another set for I/O output drivers ( $V_{CCIO}$ ).

**Table 20. 2.5-V EPF10K50E & EPF10K200E Device Recommended Operating Conditions**

Symbol	Parameter	Conditions	Min	Max	Unit
V <sub>CCINT</sub>	Supply voltage for internal logic and input buffers	(3), (4)	2.30 (2.30)	2.70 (2.70)	V
V <sub>CCIO</sub>	Supply voltage for output buffers, 3.3-V operation	(3), (4)	3.00 (3.00)	3.60 (3.60)	V
	Supply voltage for output buffers, 2.5-V operation	(3), (4)	2.30 (2.30)	2.70 (2.70)	V
V <sub>I</sub>	Input voltage	(5)	−0.5	5.75	V
V <sub>O</sub>	Output voltage		0	V <sub>CCIO</sub>	V
T <sub>A</sub>	Ambient temperature	For commercial use	0	70	° C
		For industrial use	−40	85	° C
T <sub>J</sub>	Operating temperature	For commercial use	0	85	° C
		For industrial use	−40	100	° C
t <sub>R</sub>	Input rise time			40	ns
t <sub>F</sub>	Input fall time			40	ns

**Table 21. 2.5-V EPF10K30E, EPF10K50S, EPF10K100E, EPF10K130E & EPF10K200S Device Recommended Operating Conditions**

Symbol	Parameter	Conditions	Min	Max	Unit
V <sub>CCINT</sub>	Supply voltage for internal logic and input buffers	(3), (4)	2.375 (2.375)	2.625 (2.625)	V
V <sub>CCIO</sub>	Supply voltage for output buffers, 3.3-V operation	(3), (4)	3.00 (3.00)	3.60 (3.60)	V
	Supply voltage for output buffers, 2.5-V operation	(3), (4)	2.375 (2.375)	2.625 (2.625)	V
V <sub>I</sub>	Input voltage	(5)	−0.5	5.75	V
V <sub>O</sub>	Output voltage		0	V <sub>CCIO</sub>	V
T <sub>A</sub>	Ambient temperature	For commercial use	0	70	° C
		For industrial use	−40	85	° C
T <sub>J</sub>	Operating temperature	For commercial use	0	85	° C
		For industrial use	−40	100	° C
t <sub>R</sub>	Input rise time			40	ns
t <sub>F</sub>	Input fall time			40	ns

Table 26. EAB Timing Microparameters *Note (1)*

Symbol	Parameter	Conditions
$t_{EABDATA1}$	Data or address delay to EAB for combinatorial input	
$t_{EABDATA2}$	Data or address delay to EAB for registered input	
$t_{EABWE1}$	Write enable delay to EAB for combinatorial input	
$t_{EABWE2}$	Write enable delay to EAB for registered input	
$t_{EABRE1}$	Read enable delay to EAB for combinatorial input	
$t_{EABRE2}$	Read enable delay to EAB for registered input	
$t_{EABCLK}$	EAB register clock delay	
$t_{EABCO}$	EAB register clock-to-output delay	
$t_{EABYPASS}$	Bypass register delay	
$t_{EABSU}$	EAB register setup time before clock	
$t_{EABH}$	EAB register hold time after clock	
$t_{EABCLR}$	EAB register asynchronous clear time to output delay	
$t_{AA}$	Address access delay (including the read enable to output delay)	
$t_{WP}$	Write pulse width	
$t_{RP}$	Read pulse width	
$t_{WDSU}$	Data setup time before falling edge of write pulse	(5)
$t_{WDH}$	Data hold time after falling edge of write pulse	(5)
$t_{WASU}$	Address setup time before rising edge of write pulse	(5)
$t_{WAH}$	Address hold time after falling edge of write pulse	(5)
$t_{RASU}$	Address setup time with respect to the falling edge of the read enable	
$t_{RAH}$	Address hold time with respect to the falling edge of the read enable	
$t_{WO}$	Write enable to data output valid delay	
$t_{DD}$	Data-in to data-out valid delay	
$t_{EABOUT}$	Data-out delay	
$t_{EABCH}$	Clock high time	
$t_{EABCL}$	Clock low time	

Table 40. EPF10K50E 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		2.0		2.7	ns
$t_{EABDATA1}$		0.6		0.7		0.9	ns
$t_{EABWE1}$		1.1		1.3		1.8	ns
$t_{EABWE2}$		0.4		0.4		0.6	ns
$t_{EABRE1}$		0.8		0.9		1.2	ns
$t_{EABRE2}$		0.4		0.4		0.6	ns
$t_{EABCLK}$		0.0		0.0		0.0	ns
$t_{EABCO}$		0.3		0.3		0.5	ns
$t_{EABYPASS}$		0.5		0.6		0.8	ns
$t_{EABSU}$	0.9		1.0		1.4		ns
$t_{EABH}$	0.4		0.4		0.6		ns
$t_{EABCLR}$	0.3		0.3		0.5		ns
$t_{AA}$		3.2		3.8		5.1	ns
$t_{WP}$	2.5		2.9		3.9		ns
$t_{RP}$	0.9		1.1		1.5		ns
$t_{WDSU}$	0.9		1.0		1.4		ns
$t_{WDH}$	0.1		0.1		0.2		ns
$t_{WASU}$	1.7		2.0		2.7		ns
$t_{WAH}$	1.8		2.1		2.9		ns
$t_{RASU}$	3.1		3.7		5.0		ns
$t_{RAH}$	0.2		0.2		0.3		ns
$t_{WO}$		2.5		2.9		3.9	ns
$t_{DD}$		2.5		2.9		3.9	ns
$t_{EABOUT}$		0.5		0.6		0.8	ns
$t_{EABCH}$	1.5		2.0		2.5		ns
$t_{EABCL}$	2.5		2.9		3.9		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
$t_{EABYPASS}$		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



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

Symbol	-1 Speed Grade		-2 Speed Grade		-3 Speed Grade		Unit
	Min	Max	Min	Max	Min	Max	
$t_{EABWCOMB}$	5.9		7.7		10.3		ns
$t_{EABWCREG}$	5.4		7.0		9.4		ns
$t_{EABDD}$		3.4		4.5		5.9	ns
$t_{EABDATACO}$		0.5		0.7		0.8	ns
$t_{EABDATASU}$	0.8		1.0		1.4		ns
$t_{EABDATAH}$	0.1		0.1		0.2		ns
$t_{EABWESU}$	1.1		1.4		1.9		ns
$t_{EABWEH}$	0.0		0.0		0.0		ns
$t_{EABWDSU}$	1.0		1.3		1.7		ns
$t_{EABWDH}$	0.2		0.2		0.3		ns
$t_{EABWASU}$	4.1		5.2		6.8		ns
$t_{EABWAH}$	0.0		0.0		0.0		ns
$t_{EABWO}$		3.4		4.5		5.9	ns

Table 49. EPF10K100E Device Interconnect Timing Microparameters *Note (1)*

Symbol	-1 Speed Grade		-2 Speed Grade		-3 Speed Grade		Unit
	Min	Max	Min	Max	Min	Max	
$t_{DIN2IOE}$		3.1		3.6		4.4	ns
$t_{DIN2LE}$		0.3		0.4		0.5	ns
$t_{DIN2DATA}$		1.6		1.8		2.0	ns
$t_{DCLK2IOE}$		0.8		1.1		1.4	ns
$t_{DCLK2LE}$		0.3		0.4		0.5	ns
$t_{SAMELAB}$		0.1		0.1		0.2	ns
$t_{SAMEROW}$		1.5		2.5		3.4	ns
$t_{SAMECOLUMN}$		0.4		1.0		1.6	ns
$t_{DIFFROW}$		1.9		3.5		5.0	ns
$t_{TROWROWS}$		3.4		6.0		8.4	ns
$t_{LEPERIPH}$		4.3		5.4		6.5	ns
$t_{LABCARRY}$		0.5		0.7		0.9	ns
$t_{LABCASC}$		0.8		1.0		1.4	ns

Table 71. EPF10K50S External Timing Parameters *Note (1)*

Symbol	-1 Speed Grade		-2 Speed Grade		-3 Speed Grade		Unit
	Min	Max	Min	Max	Min	Max	
$t_{\text{DRR}}$		8.0		9.5		12.5	ns
$t_{\text{INSU}}^{(2)}$	2.4		2.9		3.9		ns
$t_{\text{INH}}^{(2)}$	0.0		0.0		0.0		ns
$t_{\text{OUTCO}}^{(2)}$	2.0	4.3	2.0	5.2	2.0	7.3	ns
$t_{\text{INSU}}^{(3)}$	2.4		2.9				ns
$t_{\text{INH}}^{(3)}$	0.0		0.0				ns
$t_{\text{OUTCO}}^{(3)}$	0.5	3.3	0.5	4.1			ns
$t_{\text{PCISU}}$	2.4		2.9		—		ns
$t_{\text{PCIH}}$	0.0		0.0		—		ns
$t_{\text{PCICO}}$	2.0	6.0	2.0	7.7	—	—	ns

Table 72. EPF10K50S External Bidirectional Timing Parameters *Note (1)*

Symbol	-1 Speed Grade		-2 Speed Grade		-3 Speed Grade		Unit
	Min	Max	Min	Max	Min	Max	
$t_{\text{INSUBIDIR}}^{(2)}$	2.7		3.2		4.3		ns
$t_{\text{INHBIDIR}}^{(2)}$	0.0		0.0		0.0		ns
$t_{\text{INHBIDIR}}^{(3)}$	0.0		0.0		—		ns
$t_{\text{INSUBIDIR}}^{(3)}$	3.7		4.2		—		ns
$t_{\text{OUTCOBIDIR}}^{(2)}$	2.0	4.5	2.0	5.2	2.0	7.3	ns
$t_{\text{XZBIDIR}}^{(2)}$		6.8		7.8		10.1	ns
$t_{\text{ZXBIDIR}}^{(2)}$		6.8		7.8		10.1	ns
$t_{\text{OUTCOBIDIR}}^{(3)}$	0.5	3.5	0.5	4.2	—	—	
$t_{\text{XZBIDIR}}^{(3)}$		6.8		8.4		—	ns
$t_{\text{ZXBIDIR}}^{(3)}$		6.8		8.4		—	ns

**Notes to tables:**

- (1) All timing parameters are described in [Tables 24 through 30](#).
- (2) This parameter is measured without use of the ClockLock or ClockBoost circuits.
- (3) This parameter is measured with use of the ClockLock or ClockBoost circuits

Table 74. EPF10K200S Device IOE Timing Microparameters (Part 2 of 2) *Note (1)*

Symbol	-1 Speed Grade		-2 Speed Grade		-3 Speed Grade		Unit
	Min	Max	Min	Max	Min	Max	
$t_{ZX2}$		4.5		4.8		6.6	ns
$t_{ZX3}$		6.6		7.6		10.1	ns
$t_{INREG}$		3.7		5.7		7.7	ns
$t_{IOFD}$		1.8		3.4		4.0	ns
$t_{INCOMB}$		1.8		3.4		4.0	ns

Table 75. EPF10K200S Device EAB Internal Microparameters *Note (1)*

Symbol	-1 Speed Grade		-2 Speed Grade		-3 Speed Grade		Unit
	Min	Max	Min	Max	Min	Max	
$t_{EABDATA1}$		1.8		2.4		3.2	ns
$t_{EABDATA1}$		0.4		0.5		0.6	ns
$t_{EABWE1}$		1.1		1.7		2.3	ns
$t_{EABWE2}$		0.0		0.0		0.0	ns
$t_{EABRE1}$		0		0		0	ns
$t_{EABRE2}$		0.4		0.5		0.6	ns
$t_{EABCLK}$		0.0		0.0		0.0	ns
$t_{EABCO}$		0.8		0.9		1.2	ns
$t_{EABYPASS}$		0.0		0.1		0.1	ns
$t_{EABSU}$	0.7		1.1		1.5		ns
$t_{EABH}$	0.4		0.5		0.6		ns
$t_{EABCLR}$	0.8		0.9		1.2		ns
$t_{AA}$		2.1		3.7		4.9	ns
$t_{WP}$	2.1		4.0		5.3		ns
$t_{RP}$	1.1		1.1		1.5		ns
$t_{WDSU}$	0.5		1.1		1.5		ns
$t_{WDH}$	0.1		0.1		0.1		ns
$t_{WASU}$	1.1		1.6		2.1		ns
$t_{WAH}$	1.6		2.5		3.3		ns
$t_{RASU}$	1.6		2.6		3.5		ns
$t_{RAH}$	0.1		0.1		0.2		ns
$t_{WO}$		2.0		2.4		3.2	ns
$t_{DD}$		2.0		2.4		3.2	ns
$t_{EABOUT}$		0.0		0.1		0.1	ns
$t_{EABCH}$	1.5		2.0		2.5		ns
$t_{EABCL}$	2.1		2.8		3.8		ns

## 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**.



101 Innovation Drive  
San Jose, CA 95134  
(408) 544-7000  
<http://www.altera.com>  
**Applications Hotline:**  
(800) 800-EPLD  
**Literature Services:**  
[lit\\_req@altera.com](mailto:lit_req@altera.com)

Copyright © 2003 Altera Corporation. All rights reserved. Altera, The Programmable Solutions Company, the stylized Altera logo, specific device designations, and all other words and logos that are identified as trademarks and/or service marks are, unless noted otherwise, the trademarks and service marks of Altera Corporation in the U.S. and other countries. All other product or service names are the property of their respective holders. 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.

