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	Through Hole
Operating Temperature	-
Package / Case	599-BCPGA
Supplier Device Package	599-PGA (62.5x62.5)
Purchase URL	https://www.e-xfl.com/product-detail/intel/epf10k250egi599-3

General Description

Altera FLEX 10KE devices are an enhancement of the FLEX 10K device family. Based on reconfigurable CMOS SRAM elements, the Flexible Logic Element MatriX (FLEX) architecture incorporates all features necessary to implement common gate array megafunctions. With up to 250,000 gates, the FLEX 10KE family provides the density, speed, and features to integrate entire systems, including multiple 32-bit buses, into a single device.

FLEX 10KE devices are configurable, and they are 100% tested prior to shipment. As a result, the designer is not required to generate test vectors for fault coverage purposes. Instead, the designer can focus on simulation and design verification. In addition, the designer does not need to manage inventories of different gate array designs; FLEX 10KE devices can be configured on the board for the specific functionality required.

Table 5 shows FLEX 10KE performance for some common designs. All performance values shown were obtained with Synopsys DesignWare or LPM functions. Special design techniques are not required to implement the applications; the designer simply infers or instantiates a function in a Verilog HDL, VHDL, Altera Hardware Description Language (AHDL), or schematic design file.

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	196	182	143	MHz
16-bit accumulator	16	0	196	182	143	MHz
16-to-1 multiplexer, *Note (1)*	10	0	15	17	18	ns
16-bit multiplier with 3 stage pipeline, *Note (2)*	10	0	91	86	70	MHz
256 × 16 RAM read cycle speed, *Note (2)*	0	1	181	142	125	MHz
256 × 16 RAM write cycle speed, *Note (2)*	0	1	135	117	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

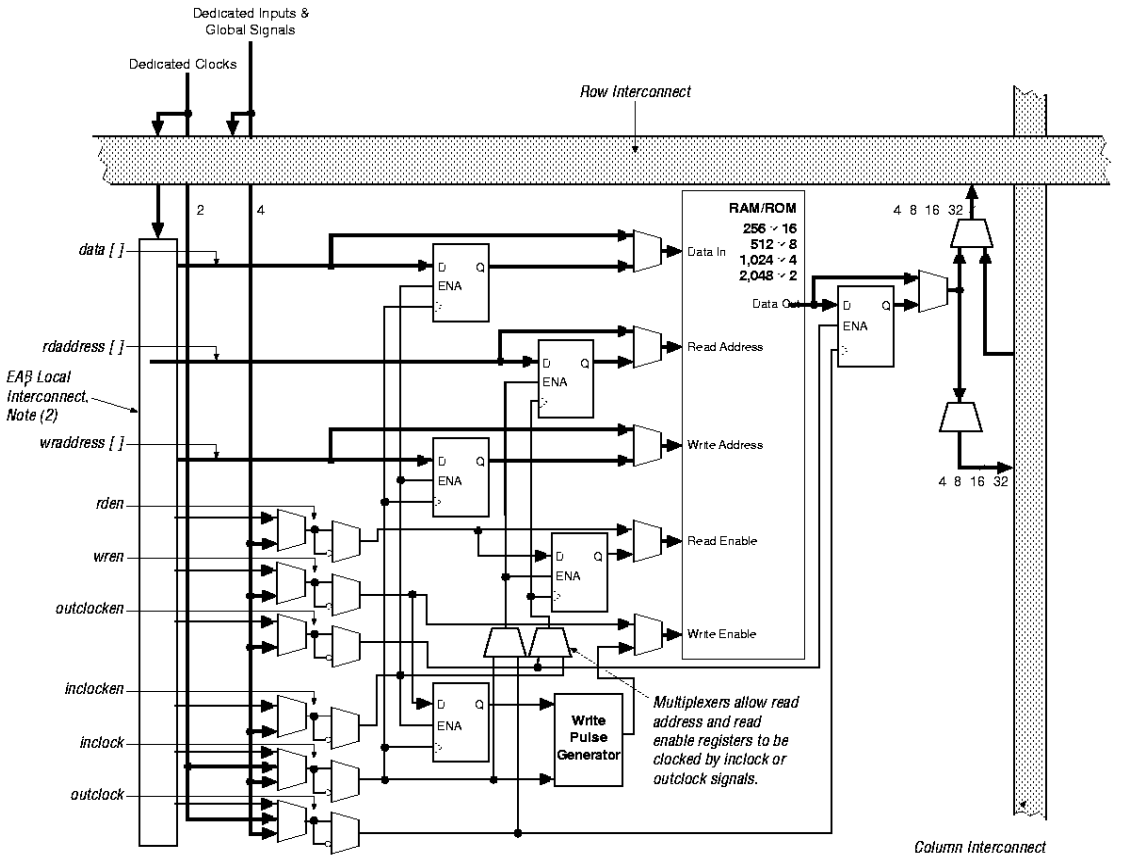
Application	LEs Used	Performance			Units
		-1 Speed Grade	-2 Speed Grade	-3 Speed Grade	
16-bit, 8-tap parallel finite impulse response (FIR) filter	592	138	128	103	MSPS
8-bit, 512-point fast Fourier transform (FFT) function	1,845	60	72	83	µs
		76	66	56	MHz
a16450 universal asynchronous receiver/transmitter (UART)	479	42	39	34	MHz

The FLEX 10KE architecture is similar to that of embedded gate arrays, which is the fastest-growing segment of the gate array market. As with standard gate arrays, embedded gate arrays implement general logic in a conventional "sea-of-gates" architecture. Additionally, embedded gate arrays have dedicated die areas for implementing large, specialized functions. By embedding functions in silicon, embedded gate arrays provide reduced die area and increased speed compared to standard gate arrays. While embedded megafunctions typically cannot be customized, FLEX 10KE devices are programmable, providing the designer with full control over embedded megafunctions and general logic while facilitating iterative design changes during debugging.

Each FLEX 10KE device contains an embedded array and a logic array. The embedded array is used to implement a variety of memory functions or complex logic functions, such as digital signal processing (DSP), microcontroller, wide-data-path manipulation, and data-transformation functions. The logic array performs the same function as the sea-of-gates in the gate array: it is used to implement general logic, such as counters, adders, state machines, and multiplexers. The combination of embedded and logic arrays provides the high performance and high density of embedded gate arrays, enabling designers to implement an entire system on a single device.

Alternatively, one clock and clock enable can be used to control the input registers of the EAB, while a different clock and clock enable control the output registers (see Figure 2).

Figure 2. FLEX 10KE Device in Dual-Port RAM Mode *Notes (1), (3)*

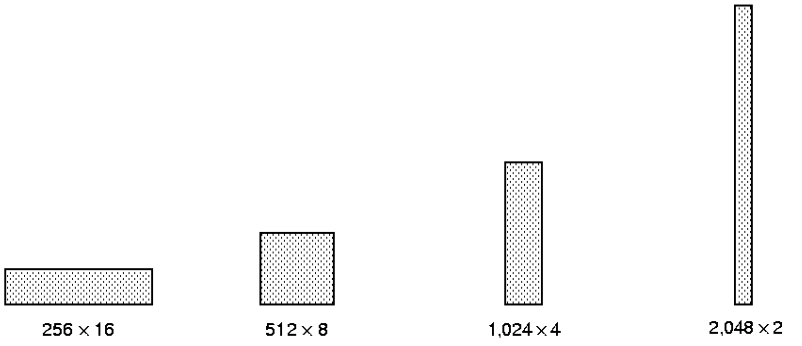


Notes:

(1) All registers can be asynchronously cleared by EAB local interconnect signals, global signals, or the chip-wide reset.
(2) EPF10K30E and EPF10K50E devices have 88 EAB local interconnect channels. EPF10K100E, EPF10K130E, EPF10K200E, and EPF10K250E devices have 104 EAB local interconnect channels.
(3) The EPF10K100B device does not offer dual-port RAM mode.

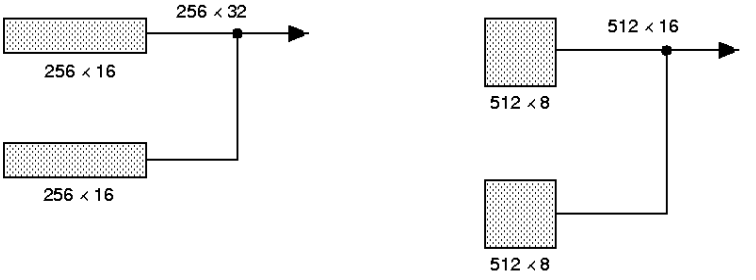
The FLEX 10KE EAB can be used in a single-port mode, which is useful for backward-compatibility with FLEX 10K designs (see Figure 3).

Figure 4. FLEX 10KE EAβ Memory Configurations



Larger blocks of RAM are created by combining multiple EABs. For example, two 256 × 16 RAM blocks can be combined to form a 256 × 32 block; two 512 × 8 RAM blocks can be combined to form a 512 × 16 block (see Figure 5).

Figure 5. Examples of Combining FLEX 10KE EABs



If necessary, all EABs in a device can be cascaded to form a single RAM block. EABs can be cascaded to form RAM blocks of up to 2,048 words without impacting timing. The Altera MAX+PLUS II software automatically combines EABs to meet a designer's RAM specifications.

EABs provide flexible options for driving and controlling clock signals. Different clocks and clock enables can be used for reading and writing to the EAB. Registers can be independently inserted on the data input, EAB output, write address, WE signals, read address, and RE signals. The global signals and the EAB local interconnect can drive WE, RE, and clock enable signals. The global signals, dedicated clock pins, and EAB local interconnect can drive the EAB clock signals. Because the LEs drive the EAB local interconnect, the LEs can control WE, RE, clear, clock, and clock enable signals.

An EAB is fed by a row interconnect and can drive out to row and column interconnects. Each EAB output can drive up to two row channels and up to two column channels; the unused row channel can be driven by other LEs. This feature increases the routing resources available for EAB outputs (see Figures 2 and 3). The column interconnect, which is adjacent to the EAB, has twice as many channels as other columns in the device.

Logic Array Block

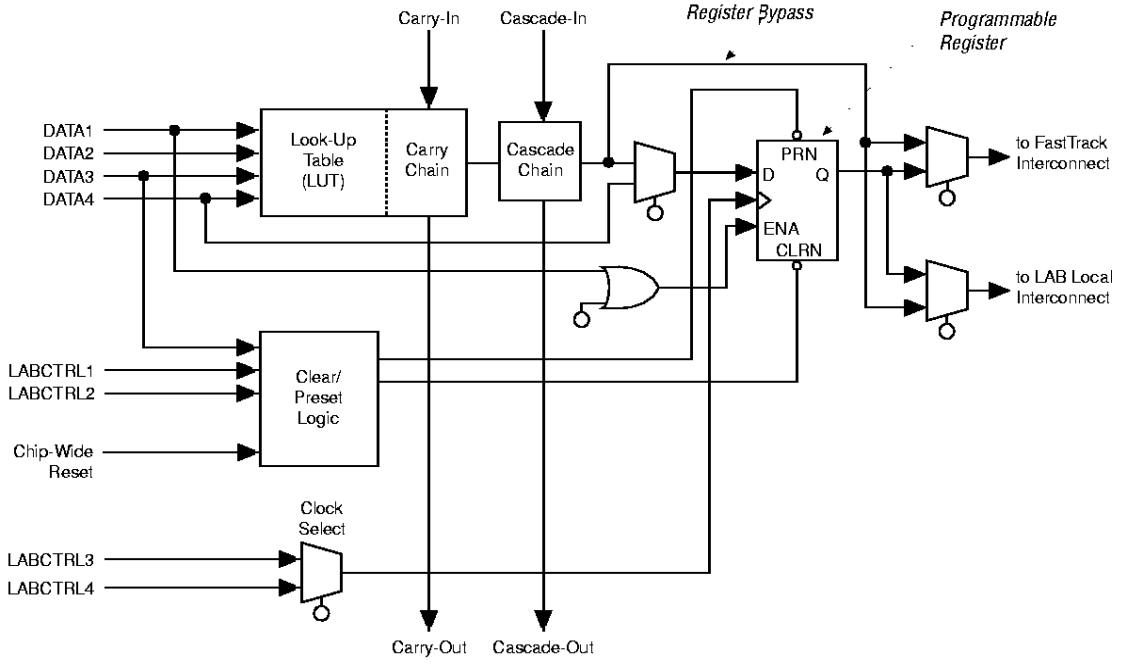
A LAB consists of eight LEs, their associated carry and cascade chains, LAB control signals, and the LAB local interconnect. The LAB provides the coarse-grained structure to the FLEX 10KE architecture, facilitating efficient routing with optimum device utilization and high performance (see Figure 6).

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 10KE 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 clock enable, a carry chain, and a cascade chain. Each LE drives both the local and the FastTrack Interconnect routing structure (see Figure 7).

Figure 7. FLEX 10KE Logic Element



Asynchronous Clear

The flipflop can be cleared by either LABCTRL1 or LABCTRL2. In this mode, the preset signal is tied to VCC to deactivate it.

Asynchronous Preset

An asynchronous preset is implemented as an asynchronous load, or with an asynchronous clear. If DATA3 is tied to VCC, asserting LABCTRL1 asynchronously loads a one into the register. Alternatively, the MAX+PLUS II software can provide preset control by using the clear and inverting the input and output of the register. Inversion control is available for the inputs to both LEs and IOEs. Therefore, if a register is preset by only one of the two LABCTRL signals, the DATA3 input is not needed and can be used for one of the LE operating modes.

Asynchronous Preset & Clear

When implementing asynchronous clear and preset, LABCTRL1 controls the preset and LABCTRL2 controls the clear. DATA3 is tied to VCC, so that asserting LABCTRL1 asynchronously loads a one into the register, effectively presetting the register. Asserting LABCTRL2 clears the register.

Asynchronous Load with Clear

When implementing an asynchronous load in conjunction with the clear, LABCTRL1 implements the asynchronous load of DATA3 by controlling the register preset and clear. LABCTRL2 implements the clear by controlling the register clear; LABCTRL2 does not have to feed the preset circuits.

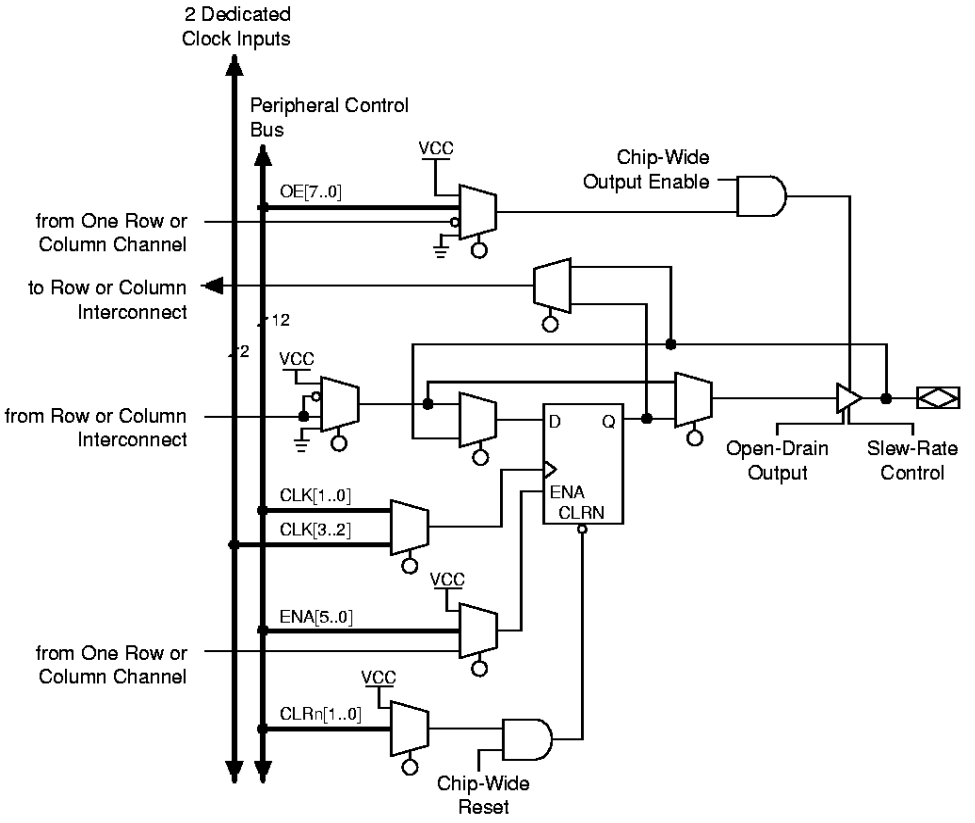
Asynchronous Load with Preset

When implementing an asynchronous load in conjunction with preset, the MAX+PLUS II software provides preset control by using the clear and inverting the input and output of the register. Asserting LABCTRL2 presets the register, while asserting LABCTRL1 loads the register. The MAX+PLUS II software inverts the signal that drives DATA3 to account for the inversion of the register's output.

Asynchronous Load without Preset or Clear

When implementing an asynchronous load without preset or clear, LABCTRL1 implements the asynchronous load of DATA3 by controlling the register preset and clear.

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

If more than six clock enable or eight output enable signals are required, each IOE on the device can be controlled by clock enable and output enable signals driven by specific LEs. In addition to the two clock signals available on the peripheral control bus, each IOE can use one of two dedicated clock pins. Each peripheral control signal can be driven by any of the dedicated input pins or the first LE of each LAB in a particular row. In addition, a LE in a different row can drive a column interconnect, which causes a row interconnect to drive the peripheral control signal. The chip-wide reset signal resets all IOE registers, overriding any other control signals.

When a dedicated clock pin drives IOE registers, it can be inverted for all IOEs in the device. All IOEs must use the same sense of the clock. For example, if any IOE uses the inverted clock, all IOEs must use the inverted clock and no IOE can use the non-inverted clock. However, LEs can still use the true or complement of the clock on a LAB-by-LAB basis.

The incoming signal may be inverted at the dedicated clock pin and will drive all IOEs. To use the true and complement of a clock to drive IOEs, drive it into both global clock pins. One global clock pin will supply the true, and the other will supply the complement.

When the true and complement of a dedicated input drives IOE clocks, then two signals on the peripheral control bus are consumed, one for each sense of the clock.

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.

ClockLock & ClockBoost Features

To support high-speed designs, FLEX 10KE devices offer optional ClockLock and ClockBoost circuitry containing a phase-locked loop (PLL) that is used to increase design speed and reduce resource usage. The ClockLock circuitry uses a synchronizing PLL that reduces the clock delay and skew within a device. This reduction minimizes clock-to-output and setup times while maintaining zero hold times. The ClockBoost circuitry, which provides a clock multiplier, allows the designer to enhance device area efficiency by resource sharing within the device. ClockBoost allows the designer to distribute a low-speed clock and multiply that clock on-device. Combined, the ClockLock and ClockBoost features provide significant improvements in system performance and bandwidth.

The ClockLock and ClockBoost features in FLEX 10KE devices are enabled through the MAX+PLUS II software. External devices are not required to use these features.

SameFrame Pin-Outs

FLEX 10KE devices support the SameFrame pin-out feature for FineLine BGA packages. SameFrame pin-out 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 anything from an EPF10K30E device in a 256-pin FineLine BGA package to an EPF10K250E device in a 672-pin FineLine BGA package.

The MAX+PLUS II software versions 9.1 and higher provide support to design PCBs with SameFrame pin-out devices. Devices can be defined for present and future use, and MAX+PLUS II software generates pin-outs describing how to lay out a board to take advantage of this migration (see Figure 17).

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 approximately 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 both the falling and rising edges of the output.

Open-Drain Output Option

FLEX 10KE devices provide an optional open-drain output (electrically equivalent to open-collector output) 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 (VCCINT), and another set for I/O output drivers (VCCIO).

The VCCINT pins must always be connected to a 2.5-V power supply. With a 2.5-V V_{CCINT} level, input voltages are compatible with 2.5-V, 3.3-V, and 5.0-V inputs. The VCCIO pins can be connected to either a 2.5-V or 3.3-V power supply, depending on the output requirements. When the VCCIO pins are connected to a 2.5-V power supply, the output levels are compatible with 2.5-V systems. When the VCCIO pins are connected to a 3.3-V power supply, the output high is at 3.3 V and is therefore compatible with 3.3-V or 5.0-V systems. Devices operating with V_{CCIO} levels higher than 3.0-V achieve a faster timing delay of t_{OD2} instead of t_{OD1}.

FLEX 10KE 2.5-V Device Recommended Operating Conditions

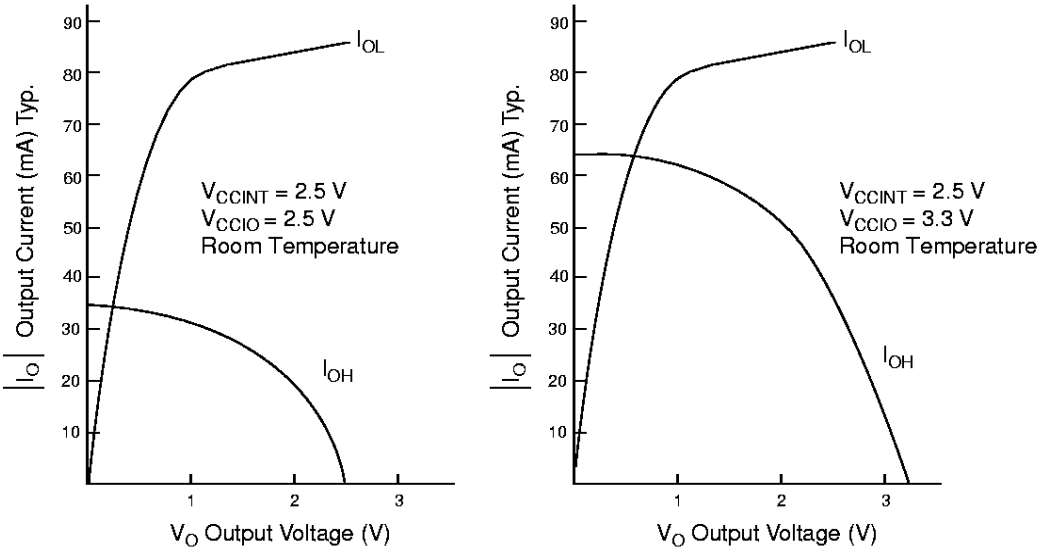
Symbol	Parameter	Conditions	Min	Max	Unit
V_{CCINT}	Supply voltage for internal logic and input buffers	*Notes (3), (4)*	2.30 (2.30)	2.70 (2.70)	V
V_{CCIO}	Supply voltage for output buffers, 3.3-V operation	*Notes (3), (4)*	3.00 (3.00)	3.60 (3.60)	V
	Supply voltage for output buffers, 2.5-V operation	*Notes (3), (4)*	2.30 (2.30)	2.70 (2.70)	V
V_I	Input voltage	*Note (5)*	0	5.3	V
V_O	Output voltage		0	V_{CCIO}	V
T_A	Ambient temperature	For commercial use	0	70	° C
		For industrial use	–40	85	° C
T_J	Operating temperature	For commercial use	0	85	° C
		For industrial use	–40	100	° C
t_R	Input rise time			40	ns
t_F	Input fall time			40	ns

FLEX 10KE 2.5-V Device DC Operating Conditions *Notes (6), (7)*

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{IH}	High-level input voltage		1.7, 0.5 × V_{CCIO}, *Note (8)*		5.3	V
V_{IL}	Low-level input voltage		–0.3		0.8, 0.3 × V_{CCIO}, *Note (8)*	V
V_{OH}	3.3-V high-level TTL output voltage	I_{OH} = –4 mA DC, V_{CCIO} = 3.00 V, *Note (9)*	2.4			V
	3.3-V high-level CMOS output voltage	I_{OH} = –0.1 mA DC, V_{CCIO} = 3.00 V, *Note (9)*	V_{CCIO} – 0.2			V
	3.3-V high-level PCI output voltage	I_{OH} = –0.5 mA DC, V_{CCIO} = 3.00 to 3.60 V, *Note (9)*	0.9 × V_{CCIO}			V
	2.5-V high-level output voltage	I_{OH} = –0.1 mA DC, V_{CCIO} = 2.30 V, *Note (9)*	2.1			V
		I_{OH} = –1 mA DC, V_{CCIO} = 2.30 V, *Note (9)*	2.0			V
		I_{OH} = –2 mA DC, V_{CCIO} = 2.30 V, *Note (9)*	1.7			V
V_{OL}	3.3-V low-level TTL output voltage	I_{OL} = 4 mA DC, V_{CCIO} = 3.00 V, *Note (10)*			0.45	V
	3.3-V low-level CMOS output voltage	I_{OL} = 0.1 mA DC, V_{CCIO} = 3.00 V, *Note (10)*			0.2	V
	3.3-V low-level PCI output voltage	I_{OL} = 1.5 mA DC, V_{CCIO} = 3.00 to 3.60 V, *Note (10)*			0.1 × V_{CCIO}	V
	2.5-V low-level output voltage	I_{OL} = 0.1 mA DC, V_{CCIO} = 2.30 V, *Note (10)*			0.2	V
		I_{OL} = 1 mA DC, V_{CCIO} = 2.30 V, *Note (10)*			0.4	V
		I_{OL} = 2 mA DC, V_{CCIO} = 2.30 V, *Note (10)*			0.7	V
I_I	Input pin leakage current	V_I = 5.3 to –0.3	–10		10	µA
I_{OZ}	Tri-stated I/O pin leakage current	V_O = 5.3 to –0.3	–10		10	µA
I_{CC0}	V_{CC} supply current (standby)	V_I = ground, no load, no toggling inputs, -1 speed grade		10		mA
		V_I = ground, no load, no toggling inputs, -2, -3 speed grades		5		mA
R_{CONF}	Value of I/O pin pull-up resistor before and during configuration	V_{CCIO} = 3.0 V, *Note (11)*	20		50	kΩ
		V_{CCIO} = 2.3 V, *Note (11)*	30		80	kΩ

Figure 21 shows the typical output drive characteristics of FLEX 10KE devices with 3.3-V and 2.5-V V_{CCIO}. The output driver is compatible with the 3.3-V ***PCI Local Bus Specification, Revision 2.1*** (with 3.3-V V_{CCIO}).

Figure 21. Output Drive Characteristics of FLEX 10KE Devices



Timing Model

Table 15 shows the external timing parameters of FLEX 10KE devices. Detailed timing information for these devices will be released as it is available.

Table 15. FLEX 10KE External Timing Parameters

Symbol	Parameter	Device	-1 Speed Grade		-2 Speed Grade		-3 Speed Grade		Unit
			Min	Max	Min	Max	Min	Max	
t_{DRR}	Register-to-register delay via 4 LEs, 3 row interconnects, and 4 local interconnects	EPF10K30E		8.5		10.0		13.5	ns
		EPF10K50E		8.5		10.0		13.5	ns
		EPF10K100B		11.0		12.0		14.5	ns
		EPF10K100E		10.0		12.0		16.0	ns
		EPF10K130E		10.0		12.0		16.0	ns
		EPF10K200E		10.0		12.0		16.0	ns
		EPF10K250E		11.0		13.5		17.0	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)* 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.

Table 19. FLEX 10KE Device Pin-Outs (Part 3 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
VCCIO (2.5 or 3.3 V)	5, 24, 45, 61, 71, 94, 115, 134	5, 22, 34, 42, 66, 84, 98, 110, 118, 138, 146, 165, 178, 194	16, 37, 57, 77, 112, 140, 160, 189, 205, 224	16, 37, 57, 77, 112, 140, 160, 189, 205, 224	A7, A23, B4, C15, D25, F4, H24, K5, M23, P2, T25, V2, W22, AB1, AC25, AD18, AF3, AF7, AF16	BC25	AR19
GNDINT	16, 57, 58, 84, 103, 127	21, 33, 49, 81, 82, 123, 129, 151, 185	10, 22, 32, 42, 52, 69, 85, 93, 104, 125, 135, 145, 155, 165, 176, 197, 216, 232	10, 22, 32, 42, 52, 69, 85, 93, 104, 125, 135, 145, 155, 165, 176, 197, 216, 232	A2, A10, A20, B1, B13, B22, B25, B26, C2, C9, C13, C25, H23, J26, K1, M1, N26, R1, R26, T1, U26, W1, AD2, AD14, AD20, AE1, AE2, AE7, AE25, AE26, AF11, AF19, AF25	E5, A3, A45, C1, C11, C19, C29, C37, C47, G25, L3, L45, W3, W45, AJ3, AJ45, AU3, AU45, BE1, BE11, BE19, BE29, BE37, BE47, BG3, BG45	AL3, AG5, AE4, AB5, Y2, U3, P5, M2, H1, B1, A11, B18, D24, F31, F35, K32, N34, T35, V32, AA33, AD35, AF32, AK35, AK31, AP24, AR18, AR11, E2, A19

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 BGA Device Pin-Outs (Part 1 of 4) *Note (1)*

Pin Name	256-Pin FineLine BGA EPF10K30E	256-Pin FineLine BGA EPF10K50E EPF10K100E EPF10K100B	672-Pin FineLine BGA EPF10K130E	672-Pin FineLine BGA 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 21. FLEX 10KE FineLine βGA Device Pin-Outs (Part 2 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
DATA5 *(4)*	D7	D7	D7	D7
DATA4 *(4)*	E7	E7	E7	E7
DATA3 *(4)*	F6	F6	F6	F6
DATA2 *(4)*	D5	D5	D5	D5
DATA1 *(4)*	E6	E6	E6	E6
DATA0 *(2)*, *(5)*	D4	D4	D4	D4
TDI *(2)*	F5	F5	F5	F5
TDO *(2)*	F19	F19	F19	F19
TCK *(2)*	E18	E18	E18	E18
TMS *(2)*	U18	U18	U18	U18
TRST *(2)*	V19	V19	V19	V19
Dedicated Inputs	E12, H11, R12, V11	E12, H11, R12, V11	E12, H11, R12, V11	E12, H11, R12, V11
Dedicated Clock Pins	D12, P11	D12, P11	D12, P11	D12, P11
DEV_CLRn *(3)*	G11	G11	G11	G11
DEV_OE *(3)*	F12	F12	F12	F12
VCCINT (2.5 V)	C11, C15, H14, J8, J10, J12, J15, L9, L10, L13, M10, M13, M14, N12, P8, P10, P12, P15, R14, V5, W21, Y8, AA12	C11, C15, H14, J8, J10, J12, J15, L9, L10, L13, M10, M13, M14, N12, P8, P10, P12, P15, R14, V5, W21, Y8, AA12	C11, C15, H14, J8, J10, J12, J15, L9, L10, L13, M10, M13, M14, N12, P8, P10, P12, P15, R14, V5, W21, Y8, AA12	C11, C15, H14, J8, J10, J12, J15, L9, L10, L13, M10, M13, M14, N12, P8, P10, P12, P15, R14, V5, W21, Y8, AA12
VCCIO (2.5 or 3.3 V)	A6, A13, B5, E1, G1, G15, H9, H20, J11, J13, K9, K11, K14, K20, L14, M9, N3, N9, N11, N14, N20, P13, R1, R9, T3, T15, T22, V22, AB13	A6, A13, B5, E1, G1, G15, H9, H20, J11, J13, K9, K11, K14, K20, L14, M9, N3, N9, N11, N14, N20, P13, R1, R9, T3, T15, T22, V22, AB13	A6, A13, B5, E1, G1, G15, H9, H20, J11, J13, K9, K11, K14, K20, L14, M9, N3, N9, N11, N14, N20, P13, R1, R9, T3, T15, T22, V22, AB13	A6, A13, B5, E1, G1, G15, H9, H20, J11, J13, K9, K11, K14, K20, L14, M9, N3, N9, N11, N14, N20, P13, R1, R9, T3, T15, T22, V22, AB13

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