



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	42
Number of Logic Elements/Cells	336
Total RAM Bits	-
Number of I/O	120
Number of Gates	4000
Voltage - Supply	4.75V ~ 5.25V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 70°C (TA)
Package / Case	160-BQFP
Supplier Device Package	160-PQFP (28x28)
Purchase URL	https://www.e-xfl.com/product-detail/intel/epf8452aqc160-3

FLEX 8000 devices provide a large number of storage elements for applications such as digital signal processing (DSP), wide-data-path manipulation, and data transformation. These devices are an excellent choice for bus interfaces, TTL integration, coprocessor functions, and high-speed controllers. The high-pin-count packages can integrate multiple 32-bit buses into a single device. [Table 3](#) shows FLEX 8000 performance and LE requirements for typical applications.

Table 3. FLEX 8000 Performance

Application	LEs Used	Speed Grade			Units
		A-2	A-3	A-4	
16-bit loadable counter	16	125	95	83	MHz
16-bit up/down counter	16	125	95	83	MHz
24-bit accumulator	24	87	67	58	MHz
16-bit address decode	4	4.2	4.9	6.3	ns
16-to-1 multiplexer	10	6.6	7.9	9.5	ns

All FLEX 8000 device packages provide four dedicated inputs for synchronous control signals with large fan-outs. Each I/O pin has an associated register on the periphery of the device. As outputs, these registers provide fast clock-to-output times; as inputs, they offer quick setup times.

The logic and interconnections in the FLEX 8000 architecture are configured with CMOS SRAM elements. FLEX 8000 devices are configured at system power-up with data stored in an industry-standard parallel EPROM or an Altera serial configuration devices, or with data provided by a system controller. Altera offers the EPC1, EPC1213, EPC1064, and EPC1441 configuration devices, which configure FLEX 8000 devices via a serial data stream. Configuration data can also be stored in an industry-standard 32 K × 8 bit or larger configuration device, or downloaded from system RAM. After a FLEX 8000 device has been configured, it can be reconfigured in-circuit by resetting the device and loading new data. Because reconfiguration requires less than 100 ms, real-time changes can be made during system operation. For information on how to configure FLEX 8000 devices, go to the following documents:

- [Configuration Devices for APEX & FLEX Devices Data Sheet](#)
- [BitBlaster Serial Download Cable Data Sheet](#)
- [ByteBlasterMV Parallel Port Download Cable Data Sheet](#)
- [Application Note 33 \(Configuring FLEX 8000 Devices\)](#)
- [Application Note 38 \(Configuring Multiple FLEX 8000 Devices\)](#)

FLEX 8000 devices contain an optimized microprocessor interface that permits the microprocessor to configure FLEX 8000 devices serially, in parallel, synchronously, or asynchronously. The interface also enables the microprocessor to treat a FLEX 8000 device as memory and configure the device by writing to a virtual memory location, making it very easy for the designer to create configuration software.

The FLEX 8000 family is supported by Altera's MAX+PLUS II development system, a single, integrated package that offers schematic, text—including the Altera Hardware Description Language (AHDL), VHDL, and Verilog HDL—and waveform design entry, compilation and logic synthesis, simulation and timing analysis, and device programming. The MAX+PLUS II software provides EDIF 2 0 0 and 3 0 0, library of parameterized modules (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 runs on Windows-based PCs and Sun SPARCstation, HP 9000 Series 700/800, and IBM RISC System/6000 workstations.

The MAX+PLUS II software interfaces 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 8000 architecture.



For more information on the MAX+PLUS II software, go to the [*MAX+PLUS II Programmable Logic Development System & Software Data Sheet*](#).

Functional Description

The FLEX 8000 architecture incorporates a large matrix of compact building blocks called logic elements (LEs). Each LE contains a 4-input LUT that provides combinatorial logic capability and a programmable register that offers sequential logic capability. The fine-grained structure of the LE provides highly efficient logic implementation.

Eight LEs are grouped together to form a logic array block (LAB). Each FLEX 8000 LAB is an independent structure with common inputs, interconnections, and control signals. The LAB architecture provides a coarse-grained structure for high device performance and easy routing.

The FLEX 8000 architecture provides two dedicated high-speed data paths—carry chains and cascade chains—that connect adjacent LEs without using local interconnect paths. The carry chain supports high-speed counters and adders; the cascade chain implements wide-input functions with minimum delay. Carry and cascade chains connect all LEs in an LAB and all LABs in the same row. Heavy use of carry and cascade chains can reduce routing flexibility. Therefore, the use of carry and cascade chains should be limited to speed-critical portions of a design.

Carry Chain

The carry chain provides a very fast (less than 1 ns) carry-forward function between LEs. The carry-in signal from a lower-order bit moves 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 8000 architecture to implement high-speed counters and adders of arbitrary width. The MAX+PLUS II Compiler can create carry chains automatically during design processing; designers can also insert carry chain logic manually during design entry.

Figure 4 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 is typically bypassed for simple adders, but can be used for an accumulator function. Another portion of the LUT and the carry chain logic generate 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 another LE, where it can be used as a general-purpose signal. In addition to mathematical functions, carry chain logic supports very fast counters and comparators.

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 signal are the inputs to a 4-input LUT. Using a configurable SRAM bit, the MAX+PLUS II Compiler automatically selects the carry-in or the DATA3 signal as an input. The LUT output can be combined with the cascade-in signal to form a cascade chain through the cascade-out signal. The LE-Out signal—the data output of the LE—is either the combinatorial output of the LUT and cascade chain, or the data output (Q) of the programmable register.

Arithmetic Mode

The arithmetic mode offers two 3-input LUTs that are ideal for implementing adders, accumulators, and comparators. One LUT provides a 3-bit function; the other generates a carry bit. As shown in Figure 6, 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 bits: a, b, and the 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 a cascade chain.

Up/Down Counter Mode

The up/down counter mode offers counter 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. Two 3-input LUTs are used: 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.

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, and the output of this multiplexer is ANDed with a synchronous clear.

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 signals select the signal that drives the bus. However, if multiple output enable signals are active, contending signals can be driven onto the bus. Conversely, if no output enable 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 is used to asynchronously load signals into a register. 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 asynchronous modes, which are chosen during design entry. LPM functions that use registers will automatically use the correct asynchronous mode. See [Figure 7](#).

- Clear only
- Preset only
- Clear and preset
- Load with clear
- Load with preset
- Load without clear or preset

Asynchronous Clear

A register is cleared by one of the two LABCTRL signals. When the CLRn port receives a low signal, the register is set to zero.

Asynchronous Preset

An asynchronous preset is implemented as either an asynchronous load or an asynchronous clear. If DATA3 is tied to VCC, asserting LABCTRL1 asynchronously loads a 1 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 Clear & Preset

When implementing asynchronous clear and preset, LABCTRL1 controls the preset and LABCTRL2 controls the clear. The DATA3 input is tied to VCC; therefore, asserting LABCTRL1 asynchronously loads a 1 into the register, effectively presetting the register. Asserting LABCTRL2 clears the register.

Asynchronous Load with Clear

When implementing an asynchronous load 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.

Asynchronous Load with Preset

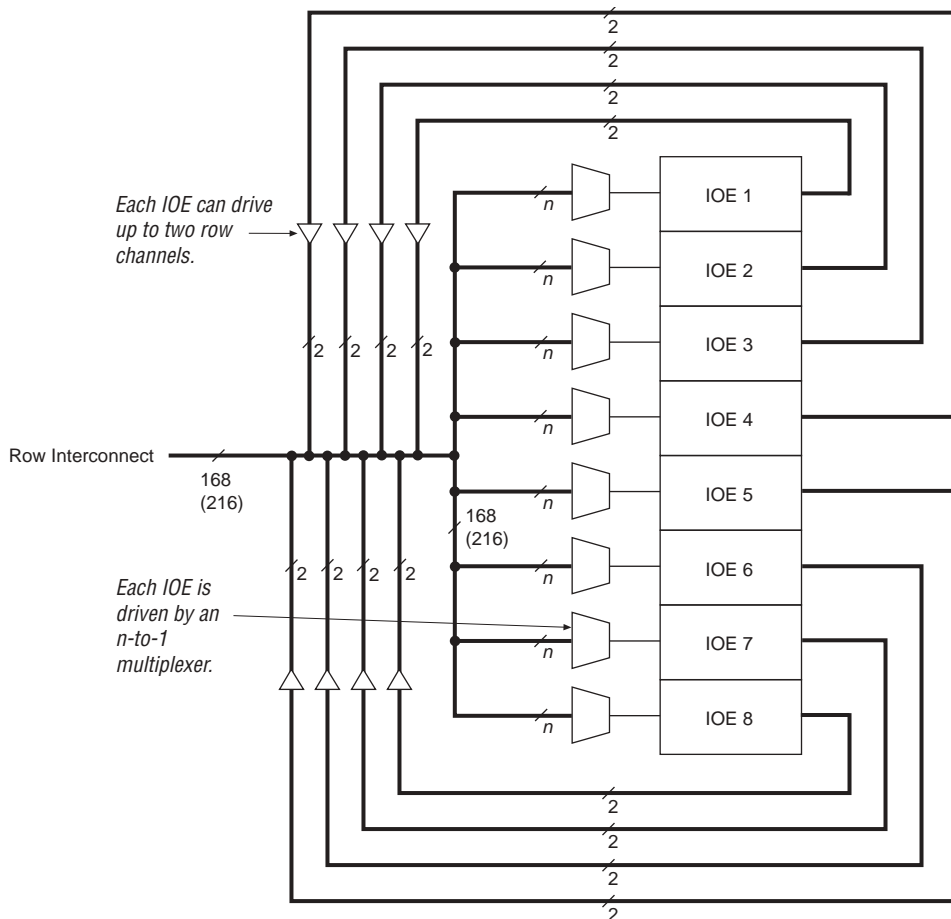
When implementing an asynchronous load in conjunction with a preset, the MAX+PLUS II software provides preset control by using the clear and inverting the input and output of the register. Asserting LABCTRL2 clears the register, while asserting LABCTRL1 loads the register. The MAX+PLUS II software inverts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

Asynchronous Load without Clear or Preset

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

Figure 11. FLEX 8000 Row-to-IOE Connections

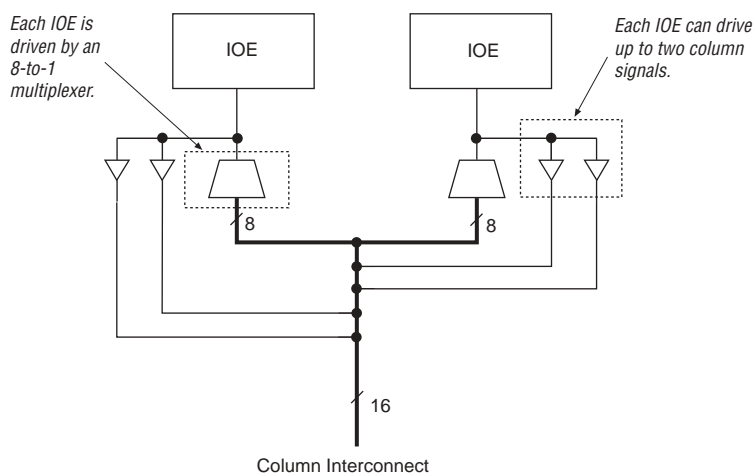
Numbers in parentheses are for EPF81500A devices. See [Note \(1\)](#).

**Note:**

- (1) $n = 13$ for EPF8282A and EPF8282AV devices.
 $n = 21$ for EPF8452A, EPF8636A, EPF8820A, and EPF81188A devices.
 $n = 27$ for EPF81500A devices.

Column-to-IOE Connections

Two IOEs are located at the top and bottom of the column channels (see [Figure 12](#)). When an IOE is used as an input, it can drive up to two separate column channels. The output signal to an IOE can choose from 8 of the 16 column channels through an 8-to-1 multiplexer.

Figure 12. FLEX 8000 Column-to-IOE Connections

In addition to general-purpose I/O pins, FLEX 8000 devices have four dedicated input pins. These dedicated inputs provide low-skew, device-wide signal distribution, and are typically used for global clock, clear, and preset control signals. The signals from the dedicated inputs are available as control signals for all LABs and I/O elements 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.

Signals enter the FLEX 8000 device either from the I/O pins that provide general-purpose input capability or from the four dedicated inputs. The IOEs are located at the ends of the row and column interconnect channels.

I/O pins can be used as input, output, or bidirectional pins. Each I/O pin has a register that can be used either as an input register for external data that requires fast setup times, or as an output register for data that requires fast clock-to-output performance. The MAX+PLUS II Compiler uses the programmable inversion option to invert signals automatically from the row and column interconnect when appropriate.

The clock, clear, and output enable controls for the IOEs are provided by a network of I/O control signals. These signals can be supplied by either the dedicated input pins or by internal logic. The IOE control-signal paths are designed to minimize the skew across the device. All control-signal sources are buffered onto high-speed drivers that drive the signals around the periphery of the device. This “peripheral bus” can be configured to provide up to four output enable signals (10 in EPF81500A devices), and up to two clock or clear signals. [Figure 13 on page 22](#) shows how two output enable signals are shared with one clock and one clear signal.

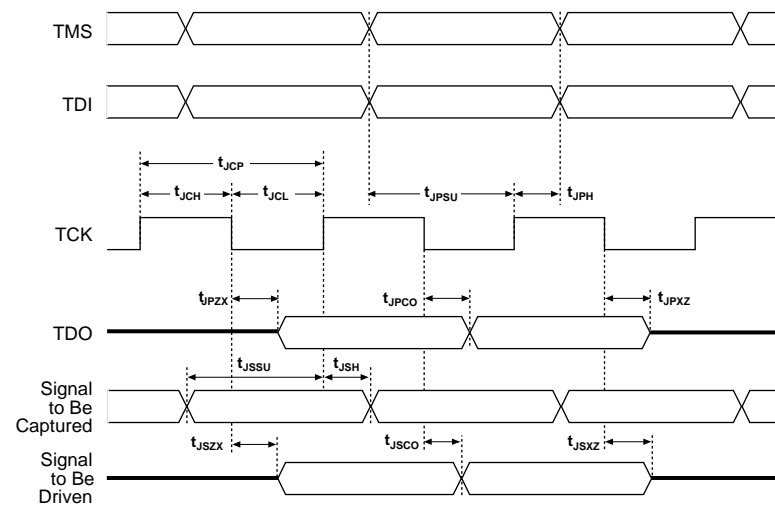
The instruction register length for FLEX 8000 devices is three bits. [Table 7](#) shows the boundary-scan register length for FLEX 8000 devices.

Table 7. FLEX 8000 Boundary-Scan Register Length

Device	Boundary-Scan Register Length
EPF8282A, EPF8282AV	273
EPF8636A	417
EPF8820A	465
EPF81500A	645

FLEX 8000 devices that support JTAG include weak pull-ups on the JTAG pins. [Figure 14](#) shows the timing requirements for the JTAG signals.

Figure 14. EPF8282A, EPF8282AV, EPF8636A, EPF8820A & EPF81500A JTAG Waveforms



[Table 8](#) shows the timing parameters and values for EPF8282A, EPF8282AV, EPF8636A, EPF8820A, and EPF81500A devices.

Table 10. FLEX 8000 5.0-V Device Recommended Operating Conditions

Symbol	Parameter	Conditions	Min	Max	Unit
V_{CCINT}	Supply voltage for internal logic and input buffers	(3), (4)	4.75 (4.50)	5.25 (5.50)	V
V_{CCIO}	Supply voltage for output buffers, 5.0-V operation	(3), (4)	4.75 (4.50)	5.25 (5.50)	V
	Supply voltage for output buffers, 3.3-V operation	(3), (4)	3.00 (3.00)	3.60 (3.60)	V
V_I	Input voltage		-0.5	$V_{CCINT} + 0.5$	V
V_O	Output voltage		0	V_{CCIO}	V
T_A	Operating temperature	For commercial use	0	70	° C
		For industrial use	-40	85	° C
t_R	Input rise time			40	ns
t_F	Input fall time			40	ns

Table 11. FLEX 8000 5.0-V Device DC Operating Conditions Notes (5), (6)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{IH}	High-level input voltage		2.0		$V_{CCINT} + 0.5$	V
V_{IL}	Low-level input voltage		-0.5		0.8	V
V_{OH}	5.0-V high-level TTL output voltage	$I_{OH} = -4$ mA DC (7) $V_{CCIO} = 4.75$ V	2.4			V
	3.3-V high-level TTL output voltage	$I_{OH} = -4$ mA DC (7) $V_{CCIO} = 3.00$ V	2.4			V
	3.3-V high-level CMOS output voltage	$I_{OH} = -0.1$ mA DC (7) $V_{CCIO} = 3.00$ V	$V_{CCIO} - 0.2$			V
V_{OL}	5.0-V low-level TTL output voltage	$I_{OL} = 12$ mA DC (7) $V_{CCIO} = 4.75$ V			0.45	V
	3.3-V low-level TTL output voltage	$I_{OL} = 12$ mA DC (7) $V_{CCIO} = 3.00$ V			0.45	V
	3.3-V low-level CMOS output voltage	$I_{OL} = 0.1$ mA DC (7) $V_{CCIO} = 3.00$ V			0.2	V
I_I	Input leakage current	$V_I = V_{CC}$ or ground	-10		10	μA
I_{OZ}	Tri-state output off-state current	$V_O = V_{CC}$ or ground	-40		40	μA
I_{CC0}	V_{CC} supply current (standby)	$V_I =$ ground, no load		0.5	10	mA

Figure 19. FLEX 8000 Timing Model

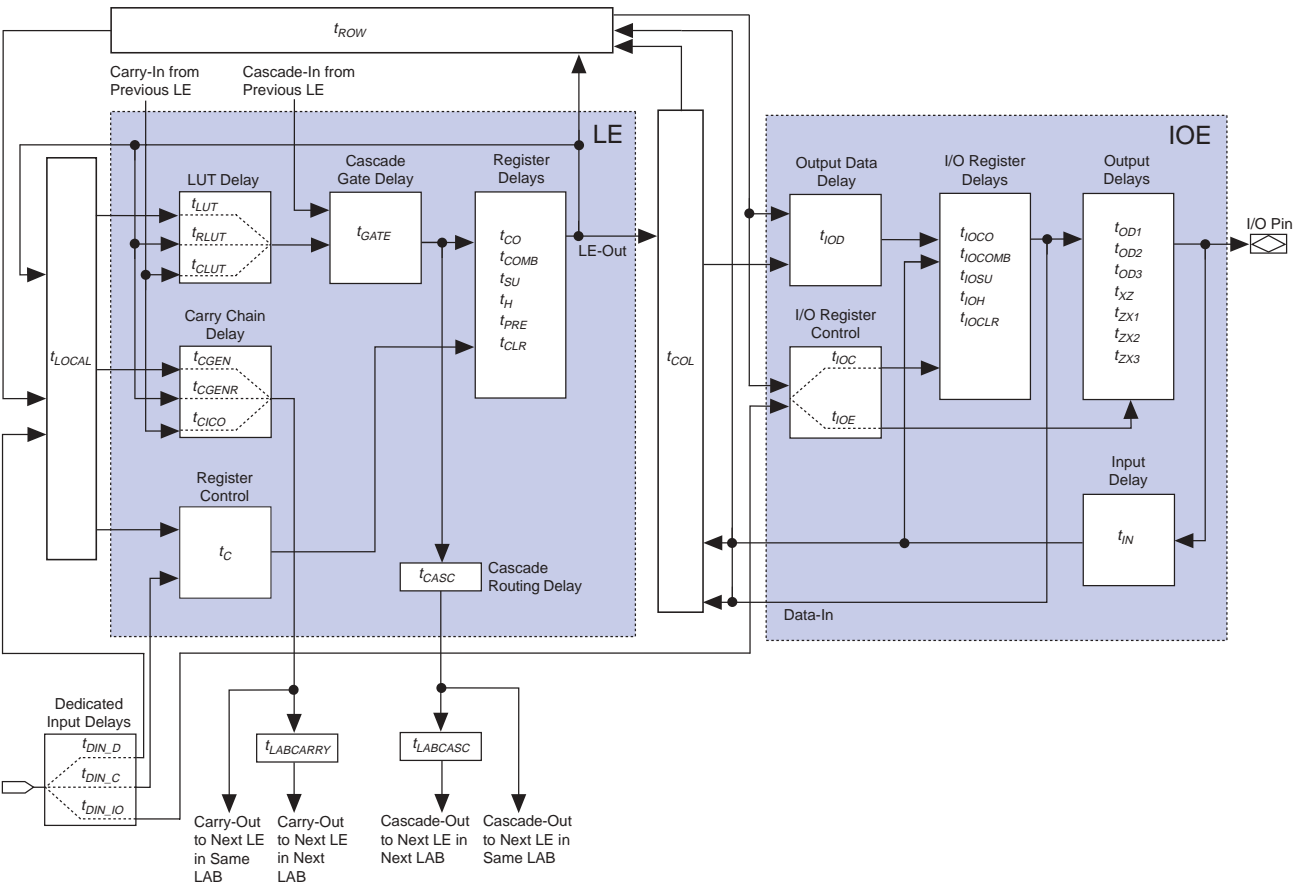


Table 21. FLEX 8000 Timing Model Interconnect Paths

Source	Destination	Total Delay
LE-Out	LE in same LAB	t_{LOCAL}
LE-Out	LE in same row, different LAB	$t_{ROW} + t_{LOCAL}$
LE-Out	LE in different row	$t_{COL} + t_{ROW} + t_{LOCAL}$
LE-Out	IOE on column	t_{COL}
LE-Out	IOE on row	t_{ROW}
IOE on row	LE in same row	$t_{ROW} + t_{LOCAL}$
IOE on column	Any LE	$t_{COL} + t_{ROW} + t_{LOCAL}$

Tables 22 through 49 show the FLEX 8000 internal and external timing parameters.

Table 22. EPF8282A Internal I/O Element Timing Parameters

Symbol	Speed Grade						Unit
	A-2		A-3		A-4		
	Min	Max	Min	Max	Min	Max	
t_{IOD}		0.7		0.8		0.9	ns
t_{IOC}		1.7		1.8		1.9	ns
t_{IOE}		1.7		1.8		1.9	ns
t_{IOCO}		1.0		1.0		1.0	ns
t_{IOCOMB}		0.3		0.2		0.1	ns
t_{IOSU}	1.4		1.6		1.8		ns
t_{IOH}	0.0		0.0		0.0		ns
t_{IOCLR}		1.2		1.2		1.2	ns
t_{IN}		1.5		1.6		1.7	ns
t_{OD1}		1.1		1.4		1.7	ns
t_{OD2}		—		—		—	ns
t_{OD3}		4.6		4.9		5.2	ns
t_{XZ}		1.4		1.6		1.8	ns
t_{ZX1}		1.4		1.6		1.8	ns
t_{ZX2}		—		—		—	ns
t_{ZX3}		4.9		5.1		5.3	ns

Table 23. EPF8282A Interconnect Timing Parameters

Symbol	Speed Grade						Unit
	A-2		A-3		A-4		
	Min	Max	Min	Max	Min	Max	
$t_{LABCASC}$		0.3		0.3		0.4	ns
$t_{LABCARRY}$		0.3		0.3		0.4	ns
t_{LOCAL}		0.5		0.6		0.8	ns
t_{ROW}		4.2		4.2		4.2	ns
t_{COL}		2.5		2.5		2.5	ns
t_{DIN_C}		5.0		5.0		5.5	ns
t_{DIN_D}		7.2		7.2		7.2	ns
t_{DIN_IO}		5.0		5.0		5.5	ns

Table 30. EPF8452A I/O Element Timing Parameters

Symbol	Speed Grade						Unit
	A-2		A-3		A-4		
	Min	Max	Min	Max	Min	Max	
t_{IOD}		0.7		0.8		0.9	ns
t_{IOC}		1.7		1.8		1.9	ns
t_{IOE}		1.7		1.8		1.9	ns
t_{IOCO}		1.0		1.0		1.0	ns
t_{IOCOMB}		0.3		0.2		0.1	ns
t_{IOSU}	1.4		1.6		1.8		ns
t_{IOH}	0.0		0.0		0.0		ns
t_{IOCLR}		1.2		1.2		1.2	ns
t_{IN}		1.5		1.6		1.7	ns
t_{OD1}		1.1		1.4		1.7	ns
t_{OD2}		—		—		—	ns
t_{OD3}		4.6		4.9		5.2	ns
t_{XZ}		1.4		1.6		1.8	ns
t_{ZX1}		1.4		1.6		1.8	ns
t_{ZX2}		—		—		—	ns
t_{ZX3}		4.9		5.1		5.3	ns

Table 31. EPF8452A Interconnect Timing Parameters

Symbol	Speed Grade						Unit
	A-2		A-3		A-4		
	Min	Max	Min	Max	Min	Max	
$t_{LABCASC}$		0.3		0.4		0.4	ns
$t_{LABCARRY}$		0.3		0.4		0.4	ns
t_{LOCAL}		0.5		0.5		0.7	ns
t_{ROW}		5.0		5.0		5.0	ns
t_{COL}		3.0		3.0		3.0	ns
t_{DIN_C}		5.0		5.0		5.5	ns
t_{DIN_D}		7.0		7.0		7.5	ns
t_{DIN_IO}		5.0		5.0		5.5	ns

Table 32. EPF8452A LE Timing Parameters

Symbol	Speed Grade						Unit
	A-2		A-3		A-4		
	Min	Max	Min	Max	Min	Max	
t_{LUT}		2.0		2.3		3.0	ns
t_{CLUT}		0.0		0.2		0.1	ns
t_{RLUT}		0.9		1.6		1.6	ns
t_{GATE}		0.0		0.0		0.0	ns
t_{CASC}		0.6		0.7		0.9	ns
t_{CICO}		0.4		0.5		0.6	ns
t_{CGEN}		0.4		0.9		0.8	ns
t_{CGENR}		0.9		1.4		1.5	ns
t_C		1.6		1.8		2.4	ns
t_{CH}	4.0		4.0		4.0		ns
t_{CL}	4.0		4.0		4.0		ns
t_{CO}		0.4		0.5		0.6	ns
t_{COMB}		0.4		0.5		0.6	ns
t_{SU}	0.8		1.0		1.1		ns
t_H	0.9		1.1		1.4		ns
t_{PRE}		0.6		0.7		0.8	ns
t_{CLR}		0.6		0.7		0.8	ns

Table 33. EPF8452A External Timing Parameters

Symbol	Speed Grade						Unit
	A-2		A-3		A-4		
	Min	Max	Min	Max	Min	Max	
t _{DRR}		16.0		20.0		25.0	ns
t _{ODH}	1.0		1.0		1.0		ns

Table 36. EPF8636A LE Timing Parameters

Symbol	Speed Grade						Unit
	A-2		A-3		A-4		
	Min	Max	Min	Max	Min	Max	
t_{LUT}		2.0		2.3		3.0	ns
t_{CLUT}		0.0		0.2		0.1	ns
t_{RLUT}		0.9		1.6		1.6	ns
t_{GATE}		0.0		0.0		0.0	ns
t_{CASC}		0.6		0.7		0.9	ns
t_{CICO}		0.4		0.5		0.6	ns
t_{CGEN}		0.4		0.9		0.8	ns
t_{CGENR}		0.9		1.4		1.5	ns
t_C		1.6		1.8		2.4	ns
t_{CH}	4.0		4.0		4.0		ns
t_{CL}	4.0		4.0		4.0		ns
t_{CO}		0.4		0.5		0.6	ns
t_{COMB}		0.4		0.5		0.6	ns
t_{SU}	0.8		1.0		1.1		ns
t_H	0.9		1.1		1.4		ns
t_{PRE}		0.6		0.7		0.8	ns
t_{CLR}		0.6		0.7		0.8	ns

Table 37. EPF8636A External Timing Parameters

Symbol	Speed Grade						Unit
	A-2		A-3		A-4		
	Min	Max	Min	Max	Min	Max	
t _{DRR}		16.0		20.0		25.0	ns
t _{ODH}	1.0		1.0		1.0		ns

Power Consumption

The supply power (P) for FLEX 8000 devices can be calculated with the following equation:

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

Typical $I_{CCSTANDBY}$ values are shown as I_{CC0} in Table 11 on page 28 and Table 15 on page 30. 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 \(Evaluating Power for Altera Devices\)](#). The $I_{CCACTIVE}$ value depends on the switching frequency and the application logic. This value can be calculated based on the amount of current that each LE typically consumes.

The following equation shows the general formula for calculating $I_{CCACTIVE}$:

$$I_{CCACTIVE} = K \times f_{MAX} \times N \times \text{tog}_{LC} \times \frac{\mu A}{\text{MHz} \times \text{LE}}$$

The parameters in this equation are shown below:

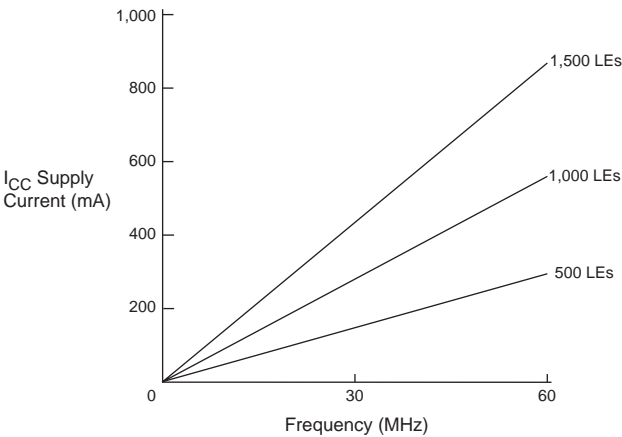
- f_{MAX} = Maximum operating frequency in MHz
- N = Total number of logic cells used in the device
- tog_{LC} = Average percentage of logic cells toggling at each clock
- K = Constant, shown in [Table 50](#)

Table 50. Values for Constant K	
Device	K
5.0-V FLEX 8000 devices	75
3.3-V FLEX 8000 devices	60

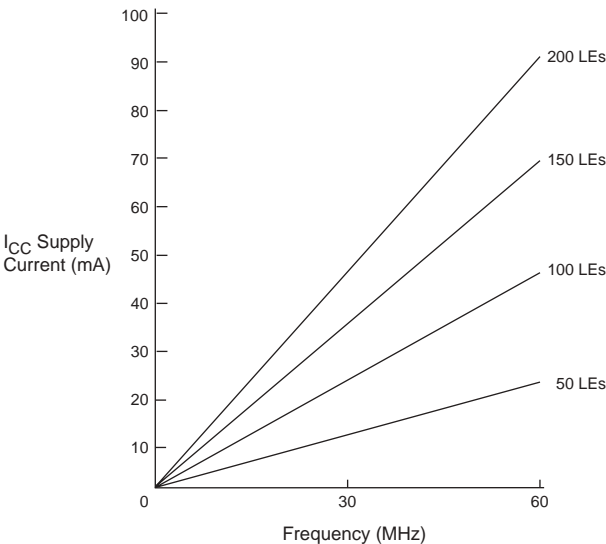
This calculation provides an I_{CC} estimate based on typical conditions with no output load. The actual I_{CC} value should be verified during operation because this measurement is sensitive to the actual pattern in the device and the environmental operating conditions.

[Figure 20](#) shows the relationship between I_{CC} and operating frequency for several LE utilization values.

Figure 20. FLEX 8000 $I_{CCACTIVE}$ vs. Operating Frequency
5.0-V FLEX 8000 Devices



3.3-V FLEX 8000 Devices



Configuration & Operation



The FLEX 8000 architecture supports several configuration schemes to load a design into the device(s) on the circuit board. This section summarizes the device operating modes and available device configuration schemes.

For more information, go to [Application Note 33 \(Configuring FLEX 8000 Devices\)](#) and [Application Note 38 \(Configuring Multiple FLEX 8000 Devices\)](#).

Device Pin-Outs

Tables 52 through 54 show the pin names and numbers for the dedicated pins in each FLEX 8000 device package.

Table 52. FLEX 8000 84-, 100-, 144- & 160-Pin Package Pin-Outs (Part 1 of 3)

Pin Name	84-Pin PLCC EPF8282A	84-Pin PLCC EPF8452A EPF8636A	100-Pin TQFP EPF8282A EPF8282AV	100-Pin TQFP EPF8452A	144-Pin TQFP EPF8820A	160-Pin PGA EPF8452A	160-Pin PQFP EPF8820A (1)
nSP (2)	75	75	75	76	110	R1	1
MSEL0 (2)	74	74	74	75	109	P2	2
MSEL1 (2)	53	53	51	51	72	A1	44
nSTATUS (2)	32	32	24	25	37	C13	82
nCONFIG (2)	33	33	25	26	38	A15	81
DCLK (2)	10	10	100	100	143	P14	125
CONF_DONE (2)	11	11	1	1	144	N13	124
nWS	30	30	22	23	33	F13	87
nRS	48	48	42	45	31	C6	89
RDCLK	49	49	45	46	12	B5	110
nCS	29	29	21	22	4	D15	118
CS	28	28	19	21	3	E15	121
RDYnBUSY	77	77	77	78	20	P3	100
CLKUSR	50	50	47	47	13	C5	107
ADD17	51	51	49	48	75	B4	40
ADD16	36	55	28	54	76	E2	39
ADD15	56	56	55	55	77	D1	38
ADD14	57	57	57	57	78	E1	37
ADD13	58	58	58	58	79	F3	36
ADD12	60	60	59	60	83	F2	32
ADD11	61	61	60	61	85	F1	30
ADD10	62	62	61	62	87	G2	28
ADD9	63	63	62	64	89	G1	26
ADD8	64	64	64	65	92	H1	22
ADD7	65	65	65	66	94	H2	20
ADD6	66	66	66	67	95	J1	18
ADD5	67	67	67	68	97	J2	16
ADD4	69	69	68	70	102	K2	11
ADD3	70	70	69	71	103	K1	10
ADD2	71	71	71	72	104	K3	8
ADD1	76	72	76	73	105	M1	7

Notes to tables:

- (1) Perform a complete thermal analysis before committing a design to this device package. See [Application Note 74 \(Evaluating Power for Altera Devices\)](#) for more information.
- (2) This pin is a dedicated pin and is not available as a user I/O pin.
- (3) SDOUT will drive out during configuration. After configuration, it may be used as a user I/O pin. By default, the MAX+PLUS II software will not use SDOUT as a user I/O pin; the user can override the MAX+PLUS II software and use SDOUT as a user I/O pin.
- (4) If the device is not configured to use the JTAG BST circuitry, this pin is available as a user I/O pin.
- (5) JTAG pins are available for EPF8636A devices only. These pins are dedicated user I/O pins.
- (6) If this pin is used as an input in user mode, ensure that it does not toggle before or during configuration.
- (7) TRST is a dedicated input pin for JTAG use. This pin must be grounded if JTAG BST is not used.
- (8) Pin 52 is a V_{CC} pin on EPF8452A devices only.
- (9) The user I/O pin count includes dedicated input pins and all I/O pins.
- (10) Unused dedicated inputs should be tied to ground on the board.
- (11) SDOUT does not exist in the EPF8636GC192 device.
- (12) These pins are no connect (N.C.) pins for EPF8636A devices only. They are user I/O pins in EPF8820A devices.
- (13) EPF8636A devices have 132 user I/O pins; EPF8820A devices have 148 user I/O pins.
- (14) For EPF81500A devices, these pins are dedicated JTAG pins and are not available as user I/O pins. If JTAG BST is not used, TDI, TCK, TMS, and TRST should be tied to GND.

Revision History

The information contained in the *FLEX 8000 Programmable Logic Device Family Data Sheet* version 11.1 supersedes information published in previous versions. The *FLEX 8000 Programmable Logic Device Family Data Sheet* version 11.1 contains the following change: minor textual updates.