Understanding FLEX 6000 Timing

May 1999, ver. 2 — Application Note 92

Introduction

Altera® devices provide predictable performance that is consistent from simulation to application. Before configuring a device, you can determine the worst-case timing delays for any design by using the MAX+PLUS® II Timing Analyzer. You can also calculate the propagation delays by using the timing model provided in this application note along with the timing parameters listed in the *FLEX 6000 Programmable Logic Device Family Data Sheet* in this data book.

For the most precise timing results, you should use the MAX+PLUS II Timing Analyzer, which accounts for the effects of secondary factors such as placement and fan-out.

This application note defines FLEX® 6000 device internal and external timing parameters, and illustrates the timing model for the FLEX 6000 device family.

Familiarity with the FLEX 6000 architecture and characteristics is assumed. Refer to the *FLEX 6000 Programmable Logic Device Family Data Sheet* for a complete description of the FLEX 6000 architecture and for specific values for timing parameters listed in this application note.

Internal Timing Microparameters

The timing delays contributed by individual FLEX 6000 architectural elements are called internal timing microparameters, which cannot be measured explicitly. All internal timing microparameters are shown in italic type. The following sections define the internal timing microparameters for the FLEX 6000 device family.

I/O Element Timing Microparameters

The following list describes the I/O element (IOE) timing microparameters for the FLEX 6000 device family:

t_{OD1}	Output buffer and pad delay with the slow slew rate logic option turned off and $V_{CCIO} = V_{CCINT}$.
t_{OD2}	Output buffer and pad delay with the slow slew rate logic option turned off and V_{CCIO} = low voltage.

A-AN-092-02

t_{OD3}	Output buffer and pad delay with the slow slew rate logic option turned on.
t_{XZ}	Output buffer disable delay. The delay required for high impedance to appear at the output pin after the tri-state buffer's enable control is disabled.
t_{ZX1}	Output buffer enable delay with the slow slew rate logic option turned off and $V_{CCIO} = V_{CCINT}$. The delay required for the output signal to appear at the output pin after the tri-state buffer's enable control is enabled.
t_{ZX2}	Output buffer enable delay with the slow slew rate logic option turned off and V_{CCIO} = low voltage. The delay required for the output signal to appear at the output pin after the tri-state buffer's enable control is enabled.
t_{ZX3}	Output buffer enable delay with the slow slew rate logic option turned on. The delay required for the output signal to appear at the output pin after the tri-state buffer's enable control is enabled.
t_{IOE}	Output enable control delay. The delay for a signal used to control the output enable of the IOE's tri-state buffer.
t_{IN}	Input pad and buffer to FastTrack® Interconnect delay. The time required for a signal on an I/O pin, used as an input, to reach a row or column channel of the FastTrack Interconnect.
t_{IN_DELAY}	Input pad and buffer to FastTrack Interconnect delay with additional delay turned on. The time required for a signal on an I/O pin, used as an input, to reach a row or column channel of the FastTrack Interconnect.

Interconnect Timing Microparameters

The following list describes the routing timing microparameters for the FLEX 6000 device family:

t_{LOCAL}	Logic array block (LAB) local interconnect delay. The delay incurred by a signal entering an LAB or by a signal routed between logic elements (LEs) by local routing.

t_{ROW}	Row interconnect routing delay. The delay incurred by a signal that requires routing through a row channel in the FastTrack Interconnect. The t_{ROW} delay is a function of fan-out and the distance between the source and destination LEs. The value shown in the *FLEX 6000 Programmable Logic Device Family Data Sheet* is the longest delay possible for an LE with a fan-out of four LEs. However, the value generated by the MAX+PLUS II Timing Analyzer is more accurate because it includes fan-out considerations and the relative locations of the source and destination LEs of the design. This parameter is a worst-case value for typical applications. Post-compilation timing simulation and timing analysis are required to determine actual worst-case performance.
t_{COL}	Column interconnect routing delay. The delay incurred by a signal that requires routing through a column channel in the FastTrack Interconnect. The value shown in the *FLEX 6000 Programmable Logic Device Family Data Sheet* is the longest delay possible for an LE with a fan-out of four LEs. However, the value generated by the MAX+PLUS II Timing Analyzer is more accurate because it includes fan-out considerations and the relative locations of the source and destination LEs of the design. This parameter is a worst-case value for typical applications. Post-compilation timing simulation and timing analysis are required to determine actual worst-case performance.
t_{DIN_D}	Dedicated input to LE data delay. The time required for a signal, used as a data input, to reach an LE from a dedicated input pin. The t_{DIN_D} delay is a function of fan-out and the distance between the source pin and destination LEs. The value shown in the *FLEX 6000 Programmable Logic Device Family Data Sheet* is the longest delay possible for a dedicated input with a fan-out of four LEs. However, the value generated by the MAX+PLUS II Timing Analyzer is more accurate because it includes fan-out considerations and the relative locations of the source and destination LEs of the design.

This parameter is a worst-case value for typical applications. Post-compilation timing simulation and timing analysis are required to determine actual worst-case performance.

t_{DIN_C}	Dedicated input to LE control delay. The delay of a signal coming from a dedicated input pin that is used as an LE register control signal. These LE register control signals include the clock, clear, and preset inputs to the LE register.
$t_{LEGLOBAL}$	LE output to LE control via internally generated global signal delay. The delay incurred by a signal that routes from the output of an LE to an LE register control signal via an internally generated global signal. These LE register control signals include the clock, clear, and preset inputs to the LE register.

This parameter is a worst-case value for typical applications. Post-compilation timing simulation and timing analysis are required to determine actual worst-case performance.

$t_{LABCARRY}$	Routing delay for the carry-out signal of an LE driving the carry-in signal of a different LE in a different LAB. A carry chain longer than one LAB skips either from one even-numbered LAB to another even-numbered LAB, or from one odd-numbered LAB to another odd-numbered LAB.
$t_{LABCASC}$	Routing delay for the cascade-out signal of an LE driving the cascade-in signal of a different LE in a different LAB. A cascade chain longer than one LAB skips either from one even-numbered LAB to another even-numbered LAB, or from one odd-numbered LAB to another odd-numbered LAB.

Logic Element Timing Microparameters

The following list describes the LE timing microparameters for the FLEX 6000 device family:

$t_{REG\_TO\_REG}$	Look-up table (LUT) delay for LE register feedback in counter mode. The delay incurred by a signal that is routed from the data output of an LE register back to the data input of the same LE register. The first LE of a counter implemented with carry chains incurs a $t_{REG\_TO\_REG}$ delay.
$t_{CASC\_TO\_REG}$	Cascade-in to register delay. The delay from the cascade-in signal to the data input of the LE register.
$t_{CARRY\_TO\_REG}$	Carry-in to register delay. The delay from the carry-in signal to the data input of the LE register.
$t_{DATA\_TO\_REG}$	LE input to register delay. The delay from the LE data input signal through the LUT to the data input of the LE register.
$t_{CASC\_TO\_OUT}$	Cascade-in to LE output delay. The delay from the cascade-in signal to the output of the LE.
$t_{CARRY\_TO\_OUT}$	Carry-in to LE output delay. The delay from the carry-in signal to the output of the LE.
$t_{DATA\_TO\_OUT}$	LE input to LE output delay. The time required for an LE data input signal to propagate through the LUT, bypass the register, and arrive at the output of the LE.
$t_{REG\_TO\_OUT}$	Register output to LE output delay. The delay from the output of the LE register to the output of the LE.
t_{SU}	LE register setup time before clock. The minimum time that the data input signal is required to be stable at the LE register input before the register clock's rising edge to ensure that the register correctly stores the input data. t_{SU} is also the minimum recovery time between deasserting the clear signal and the rising edge of the clock.
t_H	LE register hold time after clock. The minimum time that a signal is required to be stable at the LE register input after the register clock's rising edge to ensure that the register correctly stores the input data.

t_{CO}	LE register clock-to-output delay. The delay from the rising edge of the LE register's clock to the time the data appears at the register output.
t_{CLR}	LE register clear delay. The delay from the assertion of the LE register's asynchronous clear input to the time the register output stabilizes at a logic low.
t_{C}	LE register control signal delay. The time required for a signal to be routed to the clock, preset, or clear input of an LE register.
t_{LD_CLR}	Synchronous load or clear delay in counter mode. The delay from the input of the LAB-wide synchronous clear or load signal to the data input of the register in counter mode.
$t_{CARRY\_TO\_CARRY}$	Carry-in to carry-out delay. The delay incurred by generating a carry-out signal that uses the carry-in signal from the previous LE.
$t_{REG\_TO\_CARRY}$	Register output to carry-out delay. The delay incurred by generating a carry-out signal from the output of the register.
$t_{DATA\_TO\_CARRY}$	LE input to carry-out delay. The delay incurred by generating a carry-out signal from a data input signal of the LE.
$t_{CARRY\_TO\_CASC}$	Carry-in to cascade-out delay. The delay incurred by generating a cascade-out signal from the carry-in signal from the previous LE.
$t_{CASC\_TO\_CASC}$	Cascade-in to cascade-out delay. The delay incurred by generating a cascade-out signal that uses the cascade-in signal from the previous LE.
$t_{REG\_TO\_CASC}$	Register-out to cascade-out delay. The delay from the output of the LE register to the cascade-out signal.
$t_{DATA\_TO\_CASC}$	LE input to cascade-out delay. The delay from a data input of the LE to the cascade-out signal.
t_{CH}	Minimum LE register clock high time. The minimum time that the register's clock input must remain at a stable logic high state before the falling edge of the clock.

t_{CL}	Minimum LE register clock low time. The minimum time that the register's clock input must remain at a stable logic low state before the rising edge of the clock.

External Timing Parameters

External timing parameters represent actual pin-to-pin timing characteristics. Each external timing parameter consists of a combination of internal delay elements. They are worst-case values, derived from extensive performance measurements, and they are ensured by device testing or characterization. All external timing parameters are shown in bold type. For example, $\mathbf{t_1}$ is the AC operating specification. Other external timing parameters can be estimated by using the timing model or the equations in "Calculating Timing Delays" on page 928.

External Reference Timing Parameters

The following list describes the external reference timing parameters for the FLEX 6000 device family:

$\mathbf{t_1}$	This timing parameter shows the delay of a register-to-register test pattern. There are 12 LEs including the source and destination registers. The row and column interconnects between the registers have various lengths. This timing parameter is used to determine the speed grade of FLEX 6000 devices.

External Timing Parameters

The following list describes the external timing parameters for the FLEX 6000 device family.

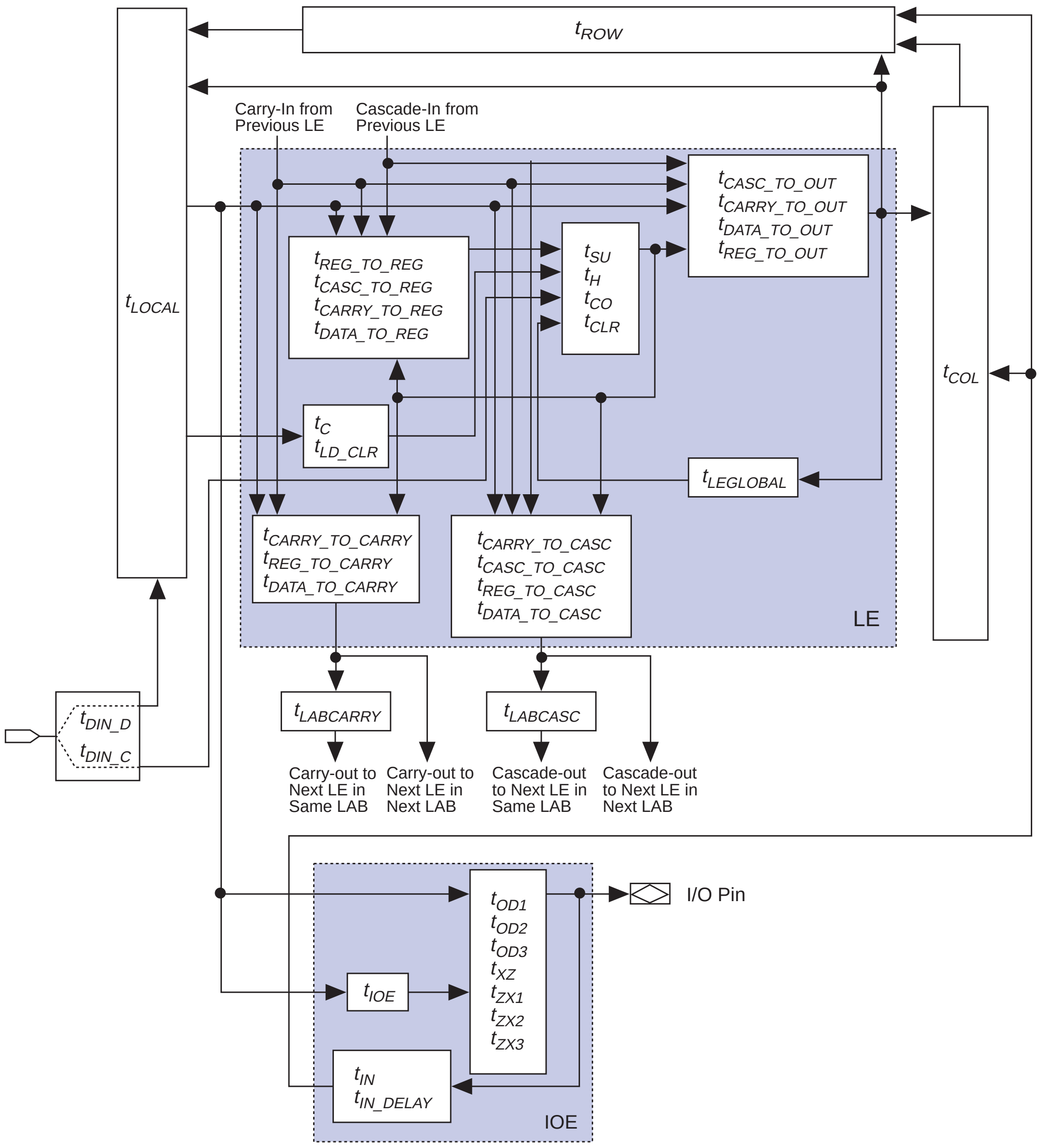
$\mathbf{t_{INSU}}$	Setup time with global clock into LE register on the same row in column nearest to the I/O. The time required for a signal to be stable at the input pin before a rising edge is applied to the global clock pin to ensure that the register correctly stores the input data. The data input of the LE register is driven by an input pin in the same row.
$\mathbf{t_{INH}}$	Hold time with global clock from an I/O pin to any LE register. The time required for a signal to be stable at the input pin after a rising edge is applied to the global clock pin to ensure that the register correctly stores the input data. This parameter does not include the optional increased input delay.

t_{OUTCO}	Clock-to-output delay with global clock with LE register using a FastFLEX™ I/O pin. The delay from when a rising edge is applied to the global clock pin to the time the data appears at the FastFLEX I/O pin. A FastFLEX I/O pin is a row or column output pin that receives its data signals from the adjacent local interconnect driven by an adjacent LE.
t_{ODH}	Output data hold time after clock. The minimum time a registered output pin will remain at its previous value after a rising edge is applied to the clock input pin. This parameter applies for both global and non-global clocking of LE registers.

FLEX 6000 Timing Model

Timing models are simplified block diagrams that illustrate the propagation delays through Altera devices. Logic can be implemented on different paths. You can trace the actual paths used in your device by examining the equations listed in the MAX+PLUS II Report File (**.rpt**) for the project. Add the appropriate internal timing parameters to calculate the approximate propagation delays through the device. However, the MAX+PLUS II Timing Analyzer provides the most accurate timing information. Figure 1 shows the FLEX 6000 timing model.

Figure 1. FLEX 6000 Timing Model



Calculating Timing Delays

You can calculate approximate pin-to-pin timing delays for FLEX 6000 devices using the timing model shown in Figure 1 along with the internal timing parameters in the *FLEX 6000 Programmable Logic Device Family Data Sheet* in this data book. Each external timing parameter is calculated from a combination of internal timing parameters.

Figure 2 shows the FLEX 6000 device family external timing parameters. For simplicity, the external parameters that contain output pin delays are shown with FastFLEX I/O pins. A FastFLEX I/O pin is a row or column output pin that receives its data signals from the adjacent local interconnect driven by an adjacent LE. To calculate the routing delays for pins that do not use the FastFLEX I/O feature, add the following delays:

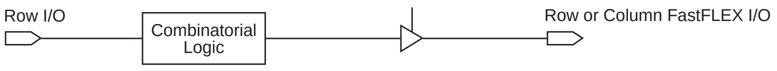
- Add a t_{ROW} delay for row output pins driven by non-adjacent LEs in the same row. This delay also applies to column output pins driven by non-adjacent LEs in the nearest row.
- Add a $t_{COL} + t_{ROW}$ delay for all other routing paths.

To calculate the delay for a signal that follows a different path, refer to the timing model to determine which internal timing parameters to sum.

Figure 2. Logic Element External Timing Parameters (Part 1 of 4)

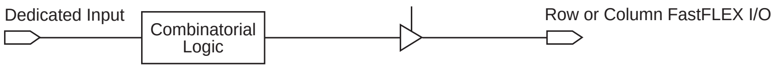
Combinatorial Delay

From Row I/O Inputs:



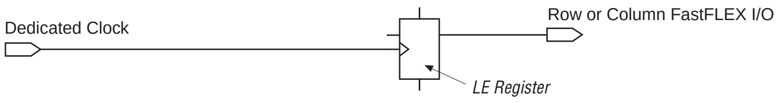
$\mathbf{t_{COMB}} = t_{IN} + t_{ROW} + t_{LOCAL} + t_{DATA\_TO\_OUT} + t_{LOCAL} + t_{OD1}$

From Dedicated Inputs:



$\mathbf{t_{COMB}} = t_{DIN\_D} + t_{LOCAL} + t_{DATA\_TO\_OUT} + t_{LOCAL} + t_{OD1}$

Clock-to-Output Delay from a Global Clock to Any Output



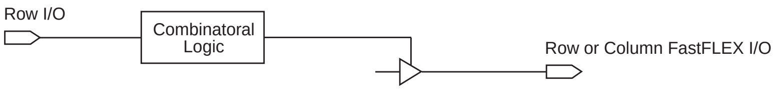
$\mathbf{t_{CO}} = t_{DIN\_C} + t_{CO} + t_{REG\_TO\_OUT} + t_{LOCAL} + t_{OD1}$

Figure 2. Logic Element External Timing Parameters (Part 2 of 4)

Tri-State Enable/Disable Delay

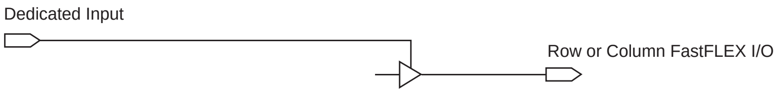
t_{XZ} or t_{ZX}

From Row I/O Inputs through Logic:



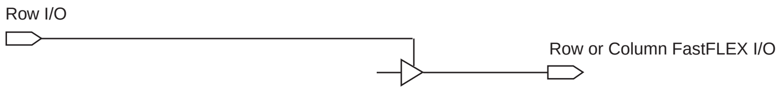
t_{XZ}, t_{ZX} = $t_{IN} + t_{ROW} + t_{LOCAL} + t_{DATA\_TO\_OUT} + t_{LOCAL} + t_{IOE} + (t_{XZ}$ or $t_{ZX1})$

Directly from Dedicated Inputs:



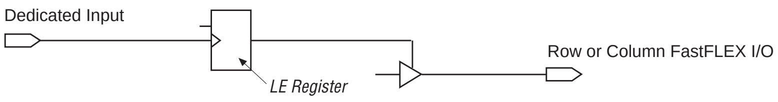
t_{XZ}, t_{ZX} = $t_{DIN\_D} + t_{LOCAL} + t_{IOE} + (t_{XZ}$ or $t_{ZX1})$

Directly from Row I/O Inputs:



t_{XZ}, t_{ZX} = $t_{IN} + t_{ROW} + t_{LOCAL} + t_{IOE} + (t_{XZ}$ or $t_{ZX1})$

From an LE Register:



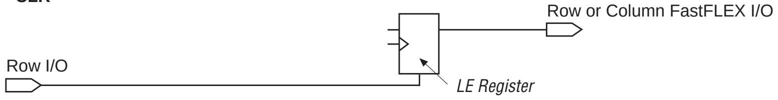
t_{XZ}, t_{ZX} = $t_{DIN\_C} + t_{CO} + t_{REG\_TO\_OUT} + t_{LOCAL} + t_{IOE} + (t_{XZ}$ or $t_{ZX1})$

Figure 2. Logic Element External Timing Parameters (Part 3 of 4)

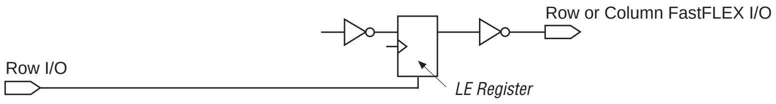
LE Register Clear & Preset (NOT Gate Push-Back) Time

From Row I/O Inputs to Locally Routed Row or Column Outputs:

$\mathbf{t_{CLR}}$



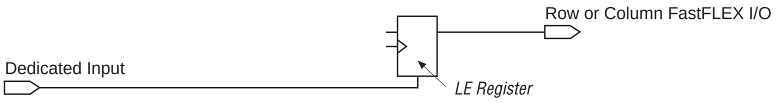
$\mathbf{t_{PRE}}$



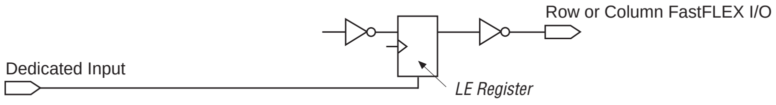
$$t_{CLR} = t_{PRE} = t_{IN} + t_{ROW} + t_{LOCAL} + t_{C} + t_{CLR} + t_{REG\_TO\_OUT} + t_{LOCAL} + t_{OD1}$$

From Dedicated Inputs to Locally Routed Row or Column Outputs:

$\mathbf{t_{CLR}}$



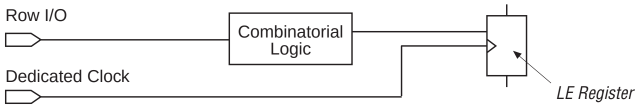
$\mathbf{t_{PRE}}$



$$t_{CLR} = t_{PRE} = t_{DIN\_C} + t_{CLR} + t_{REG\_TO\_OUT} + t_{LOCAL} + t_{OD1}$$

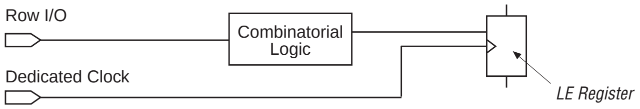
Figure 2. Logic Element External Timing Parameters (Part 4 of 4)

Setup Time from a Global Clock & Row I/O Data Input



$t_{SU} \quad = \quad [(t_{IN} + t_{ROW} + t_{LOCAL} + t_{DATA\_TO\_REG}) - (t_{DIN\_C} + t_C)] + t_{SU}$

Hold Time from a Global Clock & Row I/O Data Input



$t_H \quad = \quad [(t_{DIN\_C} + t_C) - (t_{IN} + t_{ROW} + t_{LOCAL} + t_{DATA\_TO\_REG})] + t_H$

Timing Model vs. MAX+PLUS II Timing Analyzer

Hand calculations based on the timing model provide a good estimate of a design's performance. However, the MAX+PLUS II Timing Analyzer always provides the most accurate information on the performance of a design because it takes into account three secondary factors that influence the routing microparameters:

- Fan-out for each signal in the delay path
- Positions of other loads relative to the source and destination
- Distance between signal source and destination

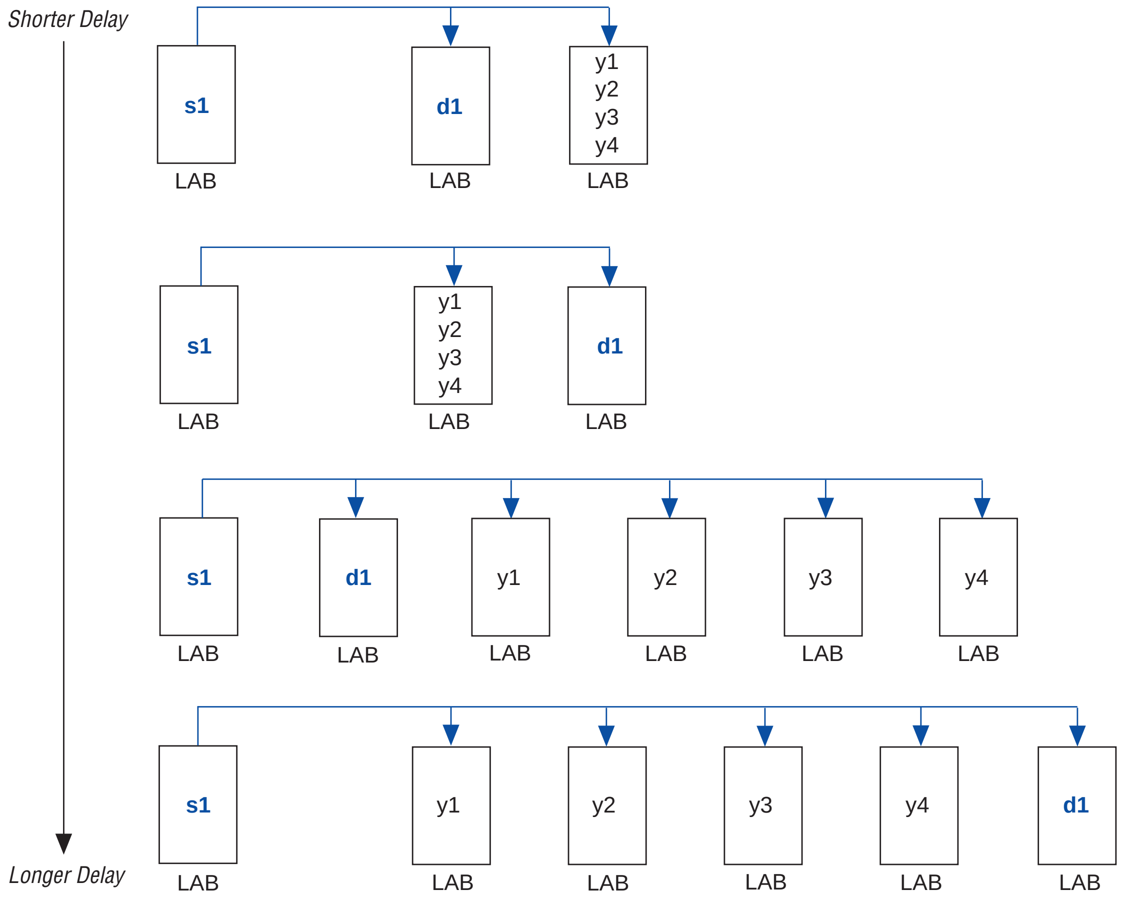
Fan-Out

The more loads a signal has to drive, the longer the delay across t_{ROW}, t_{COL}, and t_{DIN_D}. These delays are functions of the number of LABs that a signal source must drive.

Load Distribution

The load distribution relative to the source and destination also affects the t_{ROW}, t_{COL}, and t_{DIN_D} delays. Consider a signal `s1` that feeds destination `d1` and logic elements `y[4..1]`. If `y[4..1]` are in different LABs, `s1` has four additional loads. However, if the LEs are all in the same LAB, `s1` has one shorter-delay load. Therefore, the row interconnect delay from `s1` to `d1` is greater when each load `y[4..1]` is in a different LAB. Figure 3 illustrates how variations in the position of `d1` and the distribution of `y[4..1]` change the routing delay.

Figure 3. Delay from s1 to d1 as a Function of Relative Position & Load Distribution



Distance

The distance between the source and destination LEs also affects the t_{ROW}, t_{COL}, and t_{DIN_D} parameters. For example, if `s1` drives an LE in the same row, the delay from `s1` to the LE increases as the distance from `s1` to the LE increases.

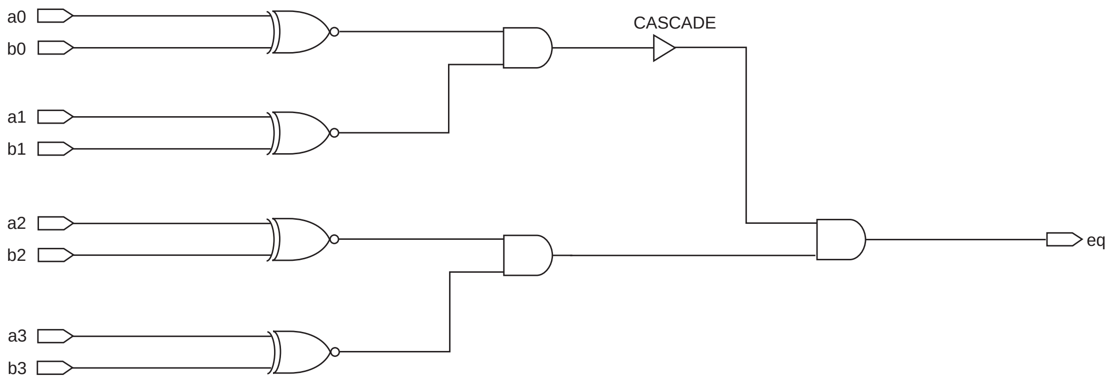
Examples

The following examples show how to use internal timing microparameters to estimate the delays for real applications.

Example 1: 4-Bit Equality Comparator with a Cascade Chain

You can analyze the timing delays for circuits that have been subjected to minimization and logic synthesis. The synthesized equations are available in your project's MAX+PLUS II Report File (**.rpt**). These equations are structured so that you can quickly determine the logic implementation of any signal. For example, Figure 4 shows a 4-bit equality comparator.

Figure 4. 4-Bit Equality Comparator Circuit

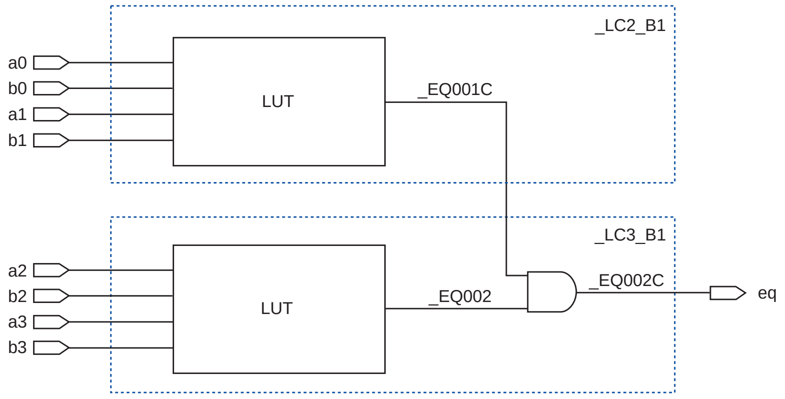


The MAX+PLUS II Report File for the circuit shown in Figure 4 gives the equations for eq, the output of the comparator:

```
eq       =  _LC3_B1;
_LC3_B1 =  LCELL( _EQ002C);
_EQ002C =  _EQ002 & CASCADE( _EQ001C);
_EQ002   =  a2 & a3 & b2 & b3
            # a2 & !a3 & b2 & !b3
            # !a2 & a3 & !b2 & b3
            # !a2 & !a3 & !b2 & !b3;
_LC2_B1 =  LCELL( _EQ001C);
_EQ001C =  _EQ001;
_EQ001   =  a0 & a1 & b0 & b1
            # a0 & !a1 & b0 & !b1
            # !a0 & a1 & !b0 & b1
            # !a0 & !a1 & !b0 & !b1;
```

Figure 5 shows a synthesized 4-bit equality comparator.

Figure 5. Synthesized 4-Bit Equality Comparator



The output pin eq is the output of the second LE of a cascade chain. The LUT of _LC2_B1 implements the comparison of the first two bits. The comparison of the second two bits is implemented in the LUT of _LC3_B1. The outputs of these two LUTs are then cascaded together to form the output of _LC3_B1.

If a2 and eq are both row I/O pins, the timing delay from a2 to eq can be estimated by adding the following microparameters:

$t_{IN} + t_{ROW} + t_{LOCAL} + t_{DATA\_TO\_OUT} + t_{ROW} + t_{LOCAL} + t_{OD1}$

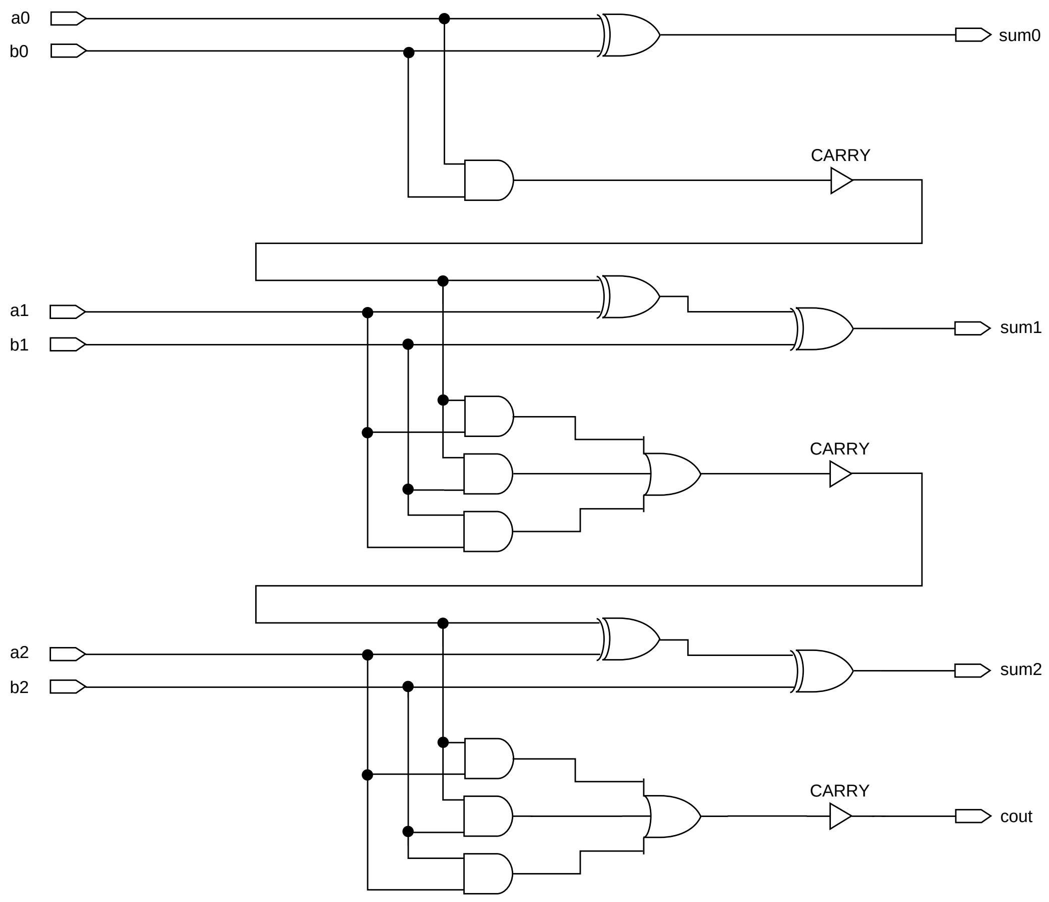
If a0 is a row I/O pin, the timing delay from a0 to eq can be estimated by adding the following microparameters:

$t_{IN} + t_{ROW} + t_{LOCAL} + t_{DATA\_TO\_CASC} + t_{CASC\_TO\_OUT} + t_{ROW} + t_{LOCAL} + t_{OD1}$

Example 2: 3-Bit Adder Using a Carry Chain

FLEX 6000 devices have specialized resources that implement complex arithmetic functions. For instance, adders and counters require a carry function to determine whether or not to increment the next significant bit. The FLEX 6000 architecture has a built-in carry chain that performs this function. This example explains how to estimate the delay for a 3-bit adder that uses a carry chain (see Figure 6).

Figure 6. 3-Bit Adder Implemented with a Carry Chain

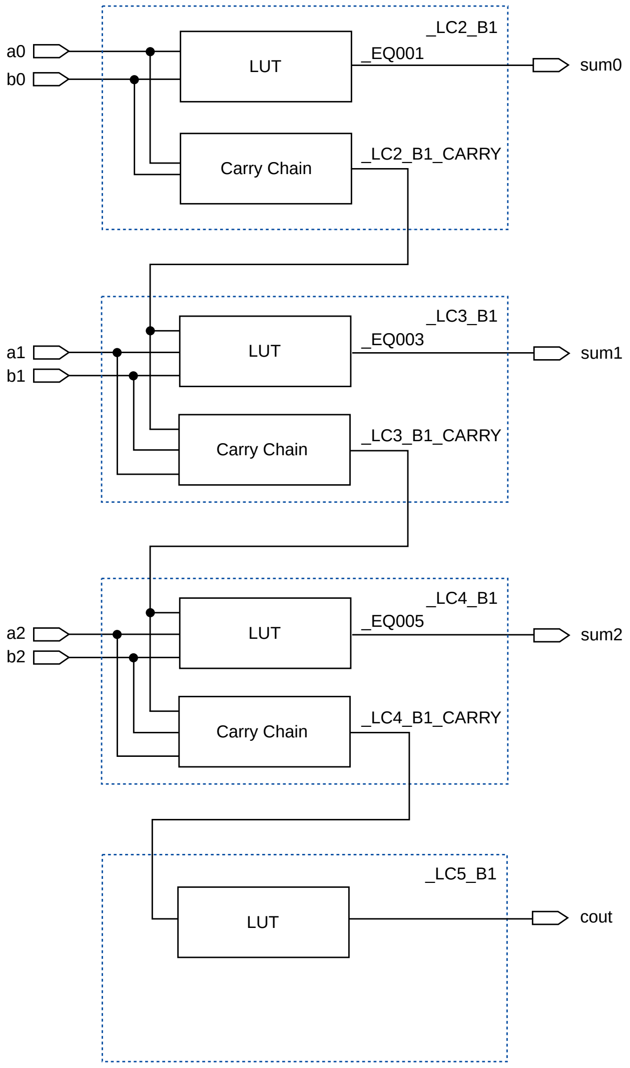


The MAX+PLUS II Report File contains the following equations for the 3-bit adder in Figure 6:

```
cout            = _LC5_B1;
sum0            = _LC2_B1;
sum1            = _LC3_B1;
sum2            = _LC4_B1;
_LC2_B1         = LCELL( _EQ001);
_EQ001          = !a0 & b0
                  # a0 & !b0;
_LC2_B1_CARRY   = CARRY( _EQ002);
_EQ002          = a0 & b0;
_LC3_B1         = LCELL( _EQ003);
_EQ003          = a1 & !b1 & !_LC2_B1_CARRY
                  # !a1 & !b1 & _LC2_B1_CARRY
                  # a1 & b1 & _LC2_B1_CARRY
                  # !a1 & b1 & !_LC2_B1_CARRY;
_LC3_B1_CARRY   = CARRY( _EQ004);
_EQ004          = a1 & _LC2_B1_CARRY
                  # a1 & b1
                  # b1 & _LC2_B1_CARRY;
_LC4_B1         = LCELL( _EQ005);
_EQ005          = a2 & !b2 & !_LC3_B1_CARRY
                  # !a2 & !b2 & _LC3_B1_CARRY
                  # a2 & b2 & _LC3_B1_CARRY
                  # !a2 & b2 & !_LC3_B1_CARRY;
_LC5_B1         = LCELL( _LC4_B1_CARRY);
_LC4_B1_CARRY   = CARRY( _EQ006);
_EQ006          = a2 & _LC3_B1_CARRY
                  # a2 & b2
                  # b2 & _LC3_B1_CARRY;
```

Figure 7 shows a synthesized 3-bit adder.

Figure 7. Synthesized 3-Bit Adder



In Figure 7, LE `_LC2_B1` generates `sum0` and a carry-out signal (`_LC2_B1_CARRY`) that feeds the carry-in of `_LC3_B1`. LE `_LC3_B1` generates `sum1` and a carry-out signal (`_LC3_B1_CARRY`) that feeds the carry-in of `_LC4_B1`. LE `_LC4_B1` generates `sum2` and `cout` using `a2`, `b2`, and `_LC3_B1_CARRY`. The `cout` signal must pass through `_LC5_B1` because a carry buffer cannot directly feed a pin.

If `a0` and `sum1` are row I/O pins, the timing delay from `a0` to `sum1` can be estimated by adding the following microparameters:

$t_{IN} + t_{ROW} + t_{LOCAL} + t_{DATA\_TO\_CARRY} + t_{CARRY\_TO\_OUT} + t_{ROW} + t_{LOCAL} + t_{OD1}$

If `a0` and `cout` are row I/O pins, the timing delay from `a0` to `cout` can be estimated by adding the following microparameters:

$t_{IN} + t_{ROW} + t_{LOCAL} + t_{DATA\_TO\_CARRY} + t_{CARRY\_TO\_CARRY} + t_{CARRY\_TO\_CARRY} + t_{CARRY\_TO\_OUT} + t_{ROW} + t_{LOCAL} + t_{OD1}$

Conclusion

The FLEX 6000 device architecture has predictable internal timing delays that can be estimated based on signal synthesis and placement. The MAX+PLUS II Timing Analyzer provides the most accurate timing information. However, you can also use the FLEX 6000 timing model along with the timing parameters listed in the *FLEX 6000 Programmable Logic Device Family Data Sheet* in this data book to estimate a design's performance before compilation. Both methods enable you to accurately predict your design's in-system timing performance.