ispMACH 4000ZE Timing Model Design and Usage Guidelines

Introduction

When implementing a design into an ispMACH® 4000ZE device, it is often critical to understand how the placement of the design will affect the timing. ispMACH 4000ZE devices have numerous paths a signal can take, each of which affects the timing. The ispMACH 4000ZE timing model was created to more accurately describe these different paths.

Architecture Basics

The fundamental architecture of the ispMACH 4000ZE devices consists of multiple optimized Generic Logic Blocks (GLBs) interconnected by a Global Routing Pool (GRP). The GRP allows communication between GLBs and routes inputs to the GLBs. The Output Routing Pool (ORP) provides flexibility when assigning macrocells to I/O pins and further enhances routability. In an ispMACH 4000ZE device, all signals incur the same delays, regardless of the macrocell used. In addition to the product terms available in each cell, there is an expander term that can be added to support wider logic. This expander term adds five product terms at a time with a maximum of 20 product terms (four groups of five product terms). This feature allows the ispMACH 4000ZE devices to offer Speed Locking™ in which performance is design-independent and guaranteed. For functions wider than 20 product terms, a third type of path offers up to 80 product terms. The block diagram for the ispMACH 4000ZE GLB is shown in Figure 1.

Figure 1. ispMACH 4000ZE Block Diagram and PAL Structure
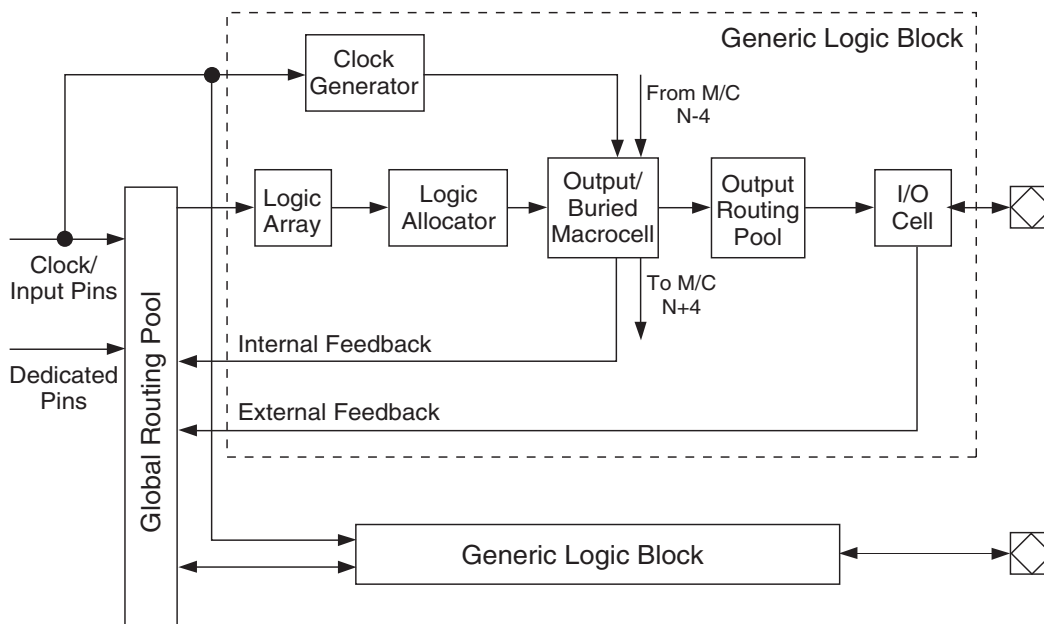
As indicated in Figure 1, any given macrocell output signal has two different feedback paths into the GRP. These two paths are referred to as internal feedback and external feedback. A signal is using internal feedback when it goes back into the GRP without going through the ORP and the I/O cell. When a signal is fed back into the GRP after having gone through the ORP and the I/O cell, it is using external feedback. Both feedback types are shown in Figure 2. For simplicity, the ORP and the I/O cell together are modeled as an output buffer.

Figure 2. ispMACH 4000ZE Signal Feedback Types
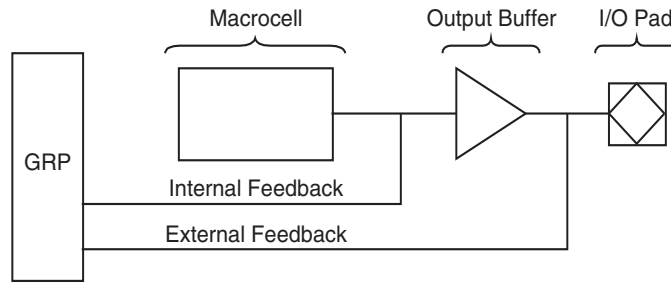
Timing Model

The primary focus of the ispMACH 4000ZE timing model is to represent the timing in an ispMACH 4000ZE device in an accurate and easy-to-understand manner. To ensure accuracy, the route of the signal must be known. To make the timing model easy to understand and use, timing is modularized such that each logic element in the signal path will have its own parameters. A diagram representing the ispMACH 4000ZE timing model is shown in Figure 3. A list of all base parameters used when calculating timing is given in the Internal Timing Parameters tables of the ispMACH 4000ZE Family Data Sheet. A list of adders and their respective base parameters is shown in Table 1. The adders are represented in italics in Figure 3. Refer to the ispMACH 4000ZE Family Data Sheet for the timing numbers associated with each parameter.
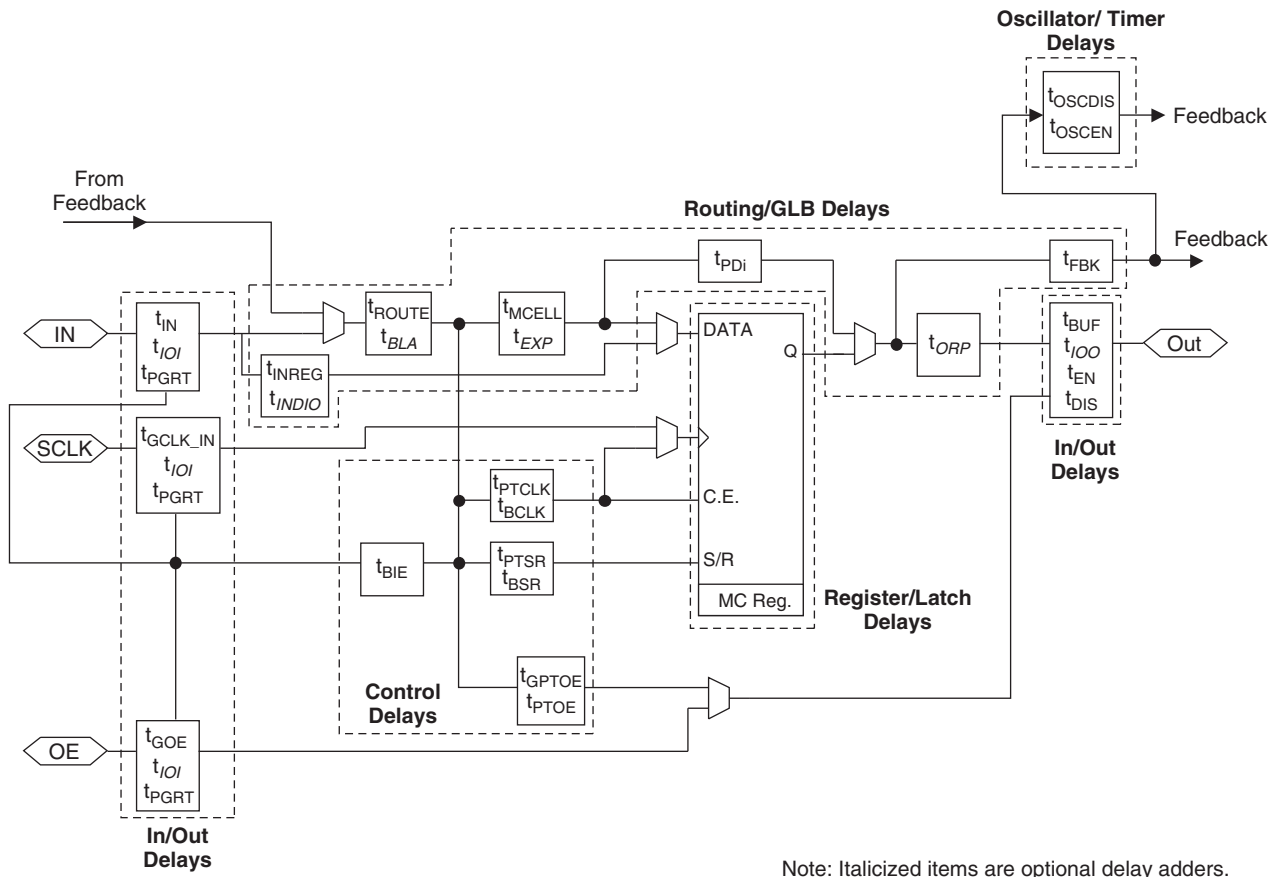
Figure 3. ispMACH 4000ZE Timing Model

Note: Italicized items are optional delay adders.

Table 1. ispMACH 4000ZE Family Timing Adders

Adder Type	Base Parameter	Description
Optional Delay Adders		
t_{BLA}	t_{ROUTE}	GLB loading adder
t_{EXP}	t_{MCELL}	Product term expander delay
t_{INDIO}	t_{INREG}	Additional delay for the input register for zero hold time
t_{IOI} Input Adjusters		
LVTTL_in	t_{IN}, t_{GCLK_IN}, t_{GOE}	Delay when input is configured as 3.3V TTL
LVCMOS15_in	t_{IN}, t_{GCLK_IN}, t_{GOE}	Delay when input is configured as 1.5V CMOS
LVCMOS18_in	t_{IN}, t_{GCLK_IN}, t_{GOE}	Delay when input is configured as 1.8V CMOS
LVCMOS25_in	t_{IN}, t_{GCLK_IN}, t_{GOE}	Delay when input is configured as 2.5V CMOS
LVCMOS33_in	t_{IN}, t_{GCLK_IN}, t_{GOE}	Delay when input is configured as 3.3V CMOS
t_{IOO} Output Adjusters		
LVTTL_out	t_{BUF}, t_{EN}, t_{DIS}	Delay when output is configured as 3.3V TTL
LVCMOS15_out	t_{BUF}, t_{EN}, t_{DIS}	Delay when output is configured as 1.5V CMOS
LVCMOS18_out	t_{BUF}, t_{EN}, t_{DIS}	Delay when output is configured as 1.8V CMOS
LVCMOS25_out	t_{BUF}, t_{EN}, t_{DIS}	Delay when output is configured as 2.5V CMOS
LVCMOS33_out	t_{BUF}, t_{EN}, t_{DIS}	Delay when output is configured as 3.3V CMOS
Slow Slew	t_{BUF}, t_{EN}	Delay when output is configured with slow slew. Slow slew may be used with any output type above.

Using the Timing Model

Timing in the ispMACH 4000ZE device family is calculated using timing delays associated with each of the blocks in the architecture. Register setup and hold times are calculated using the path delays on the data and clock signals into the register in conjunction with the inherent setup and hold times of the register itself. By using the internal path delays, accurate times are derived for timing with respect to the device input and output pins.

The Internal Timing Parameters tables of the [ispMACH 4000ZE Family Data Sheet](#) are split into several sections. The largest of the sections is the Routing Delays section, which defines all of the timing delays that are a result of a signal propagating through a particular architectural feature. As an example, t_{IN} represents the time it takes for a signal to propagate from the device I/O pad through the input buffer. In Figure 3, t_{IN} is shown in the same block with the parameter t_{IOI}. The parameter t_{IOI} appears in italics because it is an optional parameter. The same is true of all optional parameters in the model. The optional parameters are given for features, such as programmable I/O interface standards, that may affect timing but aren't always used or required.

When deriving external timing using the internal timing numbers, the following basic equations are used:

Setup time = Logic Delay + t_S - Clock Delay

Hold Time = Clock Delay + t_H - Logic Delay

Clock-to-Out Time = Clock Delay + t_{COi} + Output Path Delay

Below are examples of different setup, hold and clock-to-out times calculated using the ispMACH 4000ZE timing model. When the setup and hold times calculated are negative, the software will report them as being 0 ns.

Setup Times

Synchronous Setup $t_{SS} = (t_{IN} + t_{ROUTE} + t_{MCELL}) + t_S - (t_{GCK\_IN})$

Asynchronous Setup $t_{SA} = (t_{IN} + t_{ROUTE} + t_{MCELL}) + t_S - (t_{IN} + t_{ROUTE} + t_{PTCLK})$

Input Register Setup $t_{SIR} = (t_{IN} + t_{INREG}) + t_S - (t_{GCK\_IN})$

Hold Times

Synchronous Hold	$t_{HS} = (t_{GCK\_IN}) + t_H - (t_{IN} + t_{ROUTE} + t_{MCELL})$
Asynchronous Hold	$t_{HA} = (t_{IN} + t_{ROUTE} + t_{PTCLK}) + t_H - (t_{IN} + t_{ROUTE} + t_{MCELL})$
Input Register Hold	$t_{HIR} = (t_{GCK\_IN}) + t_H - (t_{IN} + t_{INREG})$

Clock-to-Out Time

$$t_{CO} = t_{GCK\_IN} + t_{COi} + t_{ORP} + t_{BUF}$$

Combinatorial Propagation Time

$$t_{PD} = t_{IN} + t_{ROUTE} + t_{MCELL} + t_{PDi} + t_{ORP} + t_{BUF}$$

The use of the ispMACH 4000ZE timing model will be demonstrated using two examples. The first example is a combinatorial logic design illustrating the use of internal feedback. The second example, a synchronous sequential logic design, demonstrates how to calculate f_{MAX}.

Example 1

This combinatorial logic design fits into an ispMACH 4000ZE device. A group of input signals are routed to Block A, which is in high power mode. Logic is generated in array "A" and allocated to macrocell A5, which is configured as a combinatorial path. This logic is routed to I/O 6, which is configured for a slow slew rate. The signal delay of this path would be:

$$t_{PD} = t_{IN} + t_{ROUTE} + t_{MCELL} + t_{PDi} + t_{ORP} + t_{BUF} + t_{IOO} \text{ (Slow Slew)}$$

This logic is also fed back to the central switch matrix via the internal feedback path and then routed to Block D. A second logic is generated in array "D" using the first logic along with another group of input signals. This second logic is allocated to macrocell D8, which is configured as a combinatorial path. This second logic is sent to pad I/O 31, which is in fast slew rate. The longest delay path of this design would be from Block A to I/O 31 and the delay $T_{CRITICAL}$ is:

$$T_{CRITICAL} = t_{IN} + t_{ROUTE} + t_{MCELL} + t_{PDi} + t_{FBK} + t_{ROUTE} + t_{MCELL} + t_{PDi} + t_{ORP} + t_{BUF}$$

When the number of product terms is increased beyond 20, the timing will change. T_{EXP} is used when more than 20 product terms are needed and the software settings allow for more than 20 product terms in a single function. The longest path would be for an input signal to use the expander feature of the ispMACH 4000ZE.

$$t_{PD} = t_{IN} + t_{ROUTE} + t_{MCELL} + t_{EXP} + t_{PDi} + t_{ORP} + t_{BUF}$$

Example 2

This synchronous sequential logic design has a 16-bit up-counter with load enable and reset. It fits into an isp-MACH 4256ZE device using 16 macrocells configured with T-type registers. Register inputs are defined by the device inputs and flip-flop output, which is internally fed back to the switch matrix. Under these conditions, the period t_{CNT} is limited by the internal delay from the flip-flop outputs through the internal feedback and logic to the flip-flop inputs.

$$t_{CNT} = t_{COi} + t_{FBK} + t_{ROUTE} + t_{MCELL} + t_{Si}$$

And the f_{MAX} is designated "f_{MAXINT}"

$$f_{MAXINT} = 1/ t_{CNT}$$

Conclusion

The ispMACH 4000ZE timing model provides an accurate, easy-to-understand timing calculation. It defines both internal and external feedback paths and simplifies the timing used for internal registers/latches. The timing model makes it easier to control the critical path timing in a high-speed design.

Technical Support Assistance

Hotline: 1-800-LATTICE (North America)

+1-503-268-8001 (Outside North America)

e-mail: techsupport@latticesemi.com

Internet: www.latticesemi.com

Revision History

Date	Version	Change Summary
April 2008	01.0	Initial release.
January 2010	01.1	Removed blank Internal Timing Parameters tables. Readers referred to ispMACH 4000ZE Family Data Sheet for internal timing information.