



System Design with New XC4000EX I/O Features

XAPP 056 September 17,1996 (Version 1.1)

Application Note by Lois Cartier

Summary

The XC4000EX FPGA family provides several new I/O features, including an additional latch on each input and an output multiplexer on each output. The output multiplexer can also be configured as a two-input function generator. Three different types of clock buffers allow system timing flexibility. These features are discussed, and examples show how to use them.

Xilinx Family

XC4000EX

Demonstrates

Clocking resources

Fast Capture latch on inputs

Output multiplexer/2-input function generator

Introduction

The XC4000EX FPGA family is intended for large designs starting at approximately 28,000 gates. In general, if no design changes were made in moving to a larger die, larger devices would be slower than their smaller counterparts, due to increased loading on the longer internal nets. This is true for clock lines as well as other internal signals.

A long delay from a clock pad to the internal registers can cause problems for a system designer, who often has to juggle input and output delays, taking advantage of every spare nanosecond. Short setup and clock-to-out times are a necessity.

The XC4000EX designers took several steps to avoid these typical "big-chip" timing problems. Special features in the XC4000EX designed to increase system performance include:

- Global Low-Skew buffers (BUFGLS), with access to the entire device
- Global Early buffers (BUFGE), driven by the same input signals as the Global Low-Skew buffers, but with shorter delays and limited access
- FastCLK™ buffers (BUFFCLK), offering the fastest possible path from pins to the clock inputs of nearby Input/Output Blocks (IOBs)

- Fast Capture latches, additional input latches clocked by Global Early or FastCLK buffer outputs, for reduced setup times
- Programmable data input delay: full, partial, or no delay
- Output multiplexers configurable as 2-input function generators, which in combination with the FastCLK buffers offer logic delays of as little as 6 nanoseconds, pin-to-pin.

Properly combined, these resources produce shorter setup times for inputs, shorter clock-to-pad output delays, and ultra-fast gated functions. The output multiplexer effectively doubles the number of available outputs.

XC4000EX Clocking Resources

To understand how to achieve these desirable results, it is first necessary to understand the different clock buffers and their capabilities.

Table 1 shows the three types of clock buffers provided in XC4000EX-family devices: Global Low-Skew buffers (BUFGLS), Global Early buffers (BUFGE), and FastCLK buffers (BUFFCLK). The different types of buffers have access to different portions of the device, and are intended for different uses.

Table 1: XC4000EX Clock Buffers

Direct Connections to:	BUFGLS	BUFGE (left & right)	BUFGE (top & bottom)	BUFFCLK
CLBs: All	X			
CLBs: Same Quadrant	X	X	X	
IOBs: Nearest Full Vertical Edge	X	X		
IOBs: Nearest Vertical Half-Edge	X	X	X	X
IOBs: Nearest Horizontal Full Edge	*			
IOBs: Nearest Horizontal Half Edge	*	X	*	

X Direct connection through dedicated lines

* Indirect connection through CLB array

Global Low-Skew buffers (BUFGLS) are comparable to Global Primary or Global Secondary buffers in XC4000E devices. They are the most generic type of clock buffer on the device—any of the eight buffers can drive the clock, function generator, or control pins of any Configurable Logic Block (CLB), and the clock or 3-state pins of any Input/Output Block (IOB) on the device. The path to IOB clock pins on the horizontal edges is through the CLB array.

There are two Global Low-Skew buffers in each corner of the die, as shown in [Figure 1](#). Use a Global Low-Skew buffer for any global clock where clock delay is not an important restriction.

Each Global Low-Skew buffer has a corresponding Global Early buffer (BUFGE). Each pair of buffers shares a common input, either an associated pad or an internal signal. The delay through the Global Early buffer is always less than the delay through the corresponding Low-Skew Buffer. The system designer can take advantage of this guaranteed behavior when clocking inputs into the device, as discussed in the “Shorter Setup Times” section below.

The shorter delay on the Global Early buffers is achieved in part by restricting the fanout of the clock output. Global Early clocks access only one-fourth of the CLBs on the device; those in the same quadrant as the buffer. Access to IOBs is restricted as well. Each BUFGE can drive the IOBs in the nearest vertical half-edge. Left and right BUFGEs have the additional capability of driving the nearest horizontal half-edge, as well as the entire left or right edge, respectively. Top and bottom BUFGEs access the nearest horizontal half edge through the CLB array. [Figure 2](#) and [Figure 3](#) best illustrate the areas of the device accessible from the Global Early buffers.

FastCLK buffers (BUFFCLK) provide the fastest possible connection between a pad and the dedicated clock lines paralleling the row of IOBs. FastCLK buffers are used to minimize setup times on inputs, or clock-to-out delays on outputs. In combination with the output multiplexers, they provide an extremely fast logic path, pin-to-pin. (See “Fast Gated Functions” on page 6.)

To minimize delay and skew, each FastCLK buffer accesses only the row of IOBs along half the chip edge. For example, the FastCLK buffer in the upper left edge of the device drives only the IOB clock lines along the upper half of the left edge, as shown in [Figure 4](#).

There are two FastCLK buffers on the left edge, and two on the right edge, of each XC4000EX device. There are no FastCLK buffers on the top and bottom edges of the die.

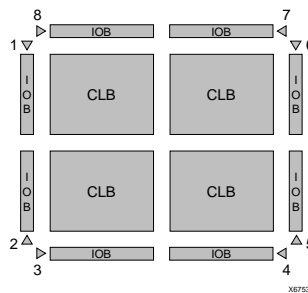


Figure 1: BUFGLS (any of #1 - #8)

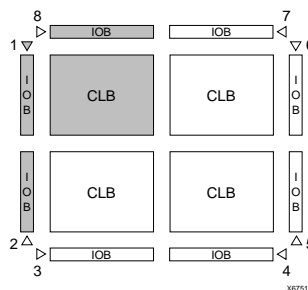


Figure 2: BUFGE #8 (top edge at left)

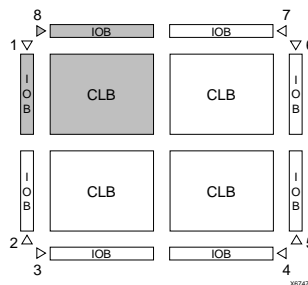


Figure 3: BUFGE #1 (left edge at top)

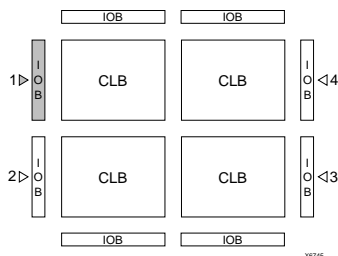


Figure 4: BUFFCLK #1 (top half of left edge)

Shorter Setup Times

Longer setup times at the device input pins mean less time available for the rest of the chip-to-chip path. Therefore, reducing the setup time required for a data input, relative to the system clock at the clock pin, is highly desirable.

A new feature is included in the XC4000EX expressly for this purpose—the Fast Capture latch. (See [Figure 5](#).) This transparent-Low latch is included in each IOB, in series with the regular transparent-High input latch or rising-edge triggered flip-flop. The latch enable is easily accessible from the dedicated IOB clock lines, but need not be sourced by the same clock as the regular latch or flip-flop. Instead, the Fast Capture latch can be enabled by one of the faster clock paths, through the Global Early or Fast-CLK buffers.

The regular latch or flip-flop is still necessary. It should be clocked by the same clock used in the internal logic, to synchronize the input data to the internal clock.

Adjusting the Delay on the Input Path

All XC4000-Series devices have an optional delay in the input path, which increases the setup time but guarantees a zero hold. This delay is included by default. Positive hold time requirements can cause problems in system design, because it is difficult to guarantee the length of time an incoming signal will remain stable after the active edge of the system clock. Therefore, the ideal situation is to decrease the input setup time, while retaining a guaranteed zero hold time.

To decrease the setup time, the delay on the input path must be decreased, relative to the clock delay. To retain a zero hold time, it cannot be decreased too much, as too short a delay would introduce a positive hold time requirement. The XC4000EX IOB therefore includes a programmable 2-tap delay element, as shown in [Figure 5](#). The standard delay can be used with the Global Low-Skew clock, and a zero hold time is guaranteed. Since a BUFGE or BUFFCLK has a shorter delay, a zero hold time is guaranteed for them as well. Similarly, a partial delay can be used with the Global Early clock, and a zero hold time is also guaranteed, for both BUFGE and BUFFCLK. However, if no delay is inserted, a zero hold time is *not* guaranteed with respect to the FastCLK output..

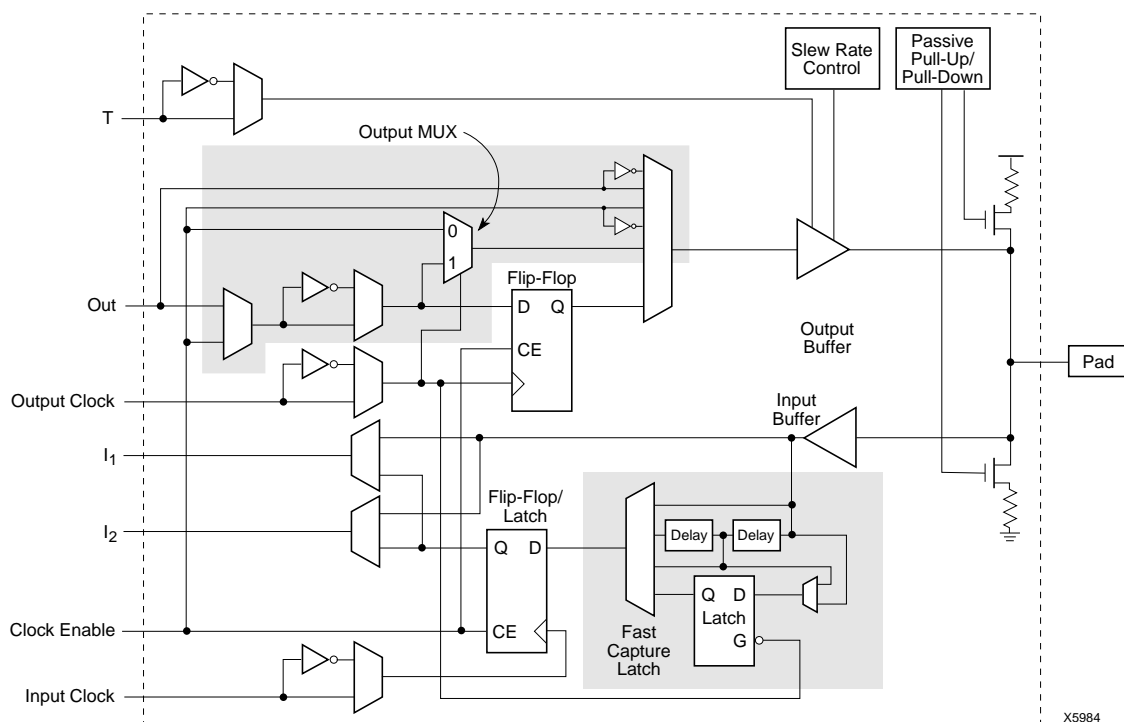


Figure 5: XC4000EX IOB with Fast Capture Latch and Output Multiplexer (Mux)

The full delay on the input path when using the Global Low-Skew buffer requires significant over-compensation for the clock delay, to guarantee a zero hold time under the least favorable operating conditions. The partial delay needed to make the same guarantee for the Global Early buffer requires substantially less over-compensation. Therefore, the setup time is shorter relative to the Global Early buffer.

The 2-tap delay can also be used with the standard latch or flip-flop alone, to adjust the setup and hold times relative to the system clock. For example, a Global Early buffer can be used with a standard input flip-flop or latch and a partial delay on the input, *as long as the same Global Early clock is used for the related internal logic*.

Preferred Usage

The Fast Capture latch (FCL) was primarily designed to be used as follows (see the top example in Figure 6).

- 1. A single input, a system clock pad, drives both a BUFGLS and BUFGE.
- 2. The BUFGE provides the latch enable (clock) for the Fast Capture latch.
- 3. The BUFGLS provides the clock for the standard latch or flip-flop in the IOB. It is also used as the clock for related logic in the CLBs.
- 4. The input to the Fast Capture latch is by default assigned a partial delay, and therefore has a guaranteed zero hold time relative to the system clock at the pad.
- 5. The Fast Capture latch output provides the data input to the standard latch or flip-flop.
- 6. The output of the standard latch or flip-flop is fully synchronized to the internal clock.

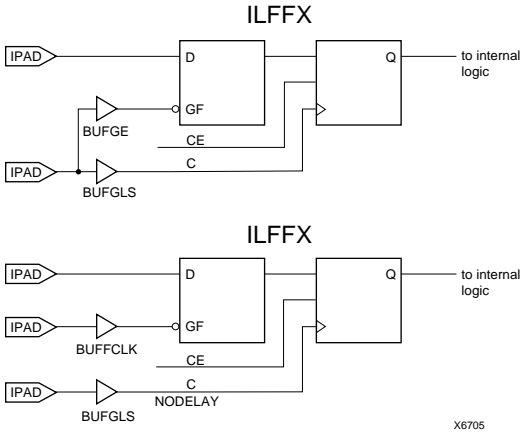


Figure 6: Examples Using Fast Capture Latch

This method may be preferred to using the FastCLK for the Fast Capture latch. Partially, this preference is because of the guaranteed zero hold time. Less obvious, perhaps, is the advantage that the BUFGE and BUFGLS can be driven by the same input pad. Therefore, there is no possibility of external skew between two different clock pads. This external skew *must* be taken into account if using the FastCLK with the Fast Capture latch.

Recommended configurations for the Fast Capture feature are shown in Table 2. Other variations can be used, but only with careful timing analysis. Setup and hold times for all variations are specified in the *XC4000 Series Field Programmable Gate Arrays* data sheet, to assist with such an analysis.

Table 2: Supported Configurations for XC4000EX IOB Input Paths

	FCL Clock	Delay on Input	Standard Latch/FF Clock	Internal Logic Clock	Comments
Most Cases	not used	Full Delay	BUFGLS	BUFGLS	Safe
Earlier Internal Clock Needed	not used	Partial Delay	BUFGE	BUFGE	Safe
Shorter Setup Required	BUFGE	Partial Delay	BUFGLS	BUFGLS	Safe, assuming shared buffer input
Urgent Need for Minimum Setup	BUFFCLK	No Delay	BUFGLS	BUFGLS	Needs careful timing analysis: possible positive hold time, possible external clock skew
Urgent Need for Minimum Setup, Earlier Internal Clock	BUFFCLK	No Delay	BUFGE	BUFGE	Needs careful timing analysis: possible positive hold time, possible external clock skew

Table 3: Fast Capture Library Symbols

Library Symbol	Standard Storage Element	Clock Enable on Standard Storage Element?
ILFFX	Flip-Flop	no
ILFLX	Latch	no
ILFFXE	Flip-Flop	yes
ILFLXE	Latch	yes

Controlling Input Configuration in a Design

Four different library symbols represent the Fast Capture latch, as shown in [Table 3](#). Each symbol includes both a Fast Capture latch (FCL) and a standard storage element, because the FCL cannot be used without the standard element. The symbols vary by whether the standard storage element is a flip-flop or a latch, and whether or not the standard element has a clock enable input. The FCL does not have a clock enable input.

The input delay is controlled, if desired, by placing an attribute or property on the Fast Capture latch symbol (see [Table 4](#)). By default, if the Fast Capture latch is used, the software assumes that a Global Early buffer sources the enable signal, and inserts a partial delay. If any other delay is desired, the addition of the appropriate attribute or property overrides the default assignment.

If timing information is specified for the design, it is not necessary to decide which clock buffer to use. The software interprets the generic library symbol BUFG as an invitation to use the clock buffer most appropriate to achieve the specified timing requirements. Therefore, for example, a single clock input can be routed through a BUFG symbol, and the BUFG output used to drive both the Fast Capture latch enable and the standard flip-flop or latch clock. The software takes care of the rest. It does not, however, place a Fast Capture latch unless specified in the schematic or instantiated in the HDL code.

Table 4: Programmable Input Delay Element

Amount of Delay	Attribute or Property	Primary Use	Zero Hold with . . .
Full Delay	none added (default)	BUFGLS	BUFGLS BUFGE BUFFCLK
Partial Delay	MEDDELAY	BUFGE	BUFGE BUFFCLK
No Delay	NODELAY	BUFFCLK	none

Faster Clock-to-Out

The new XC4000EX global buffers also reduce the delay from the active edge of the system clock to the output appearing at the output pin. The total clock-to-out delay, measured from pin to pin, includes the delay from the external clock pin to the output flip-flop clock pin, the clock-to-out delay of the output flip-flop, the delay from the flip-flop output to the output buffer, and the delay through the output buffer to the output pin. With the exception of the clock-to-out delay through the flip-flop, all of these factors are under user control.

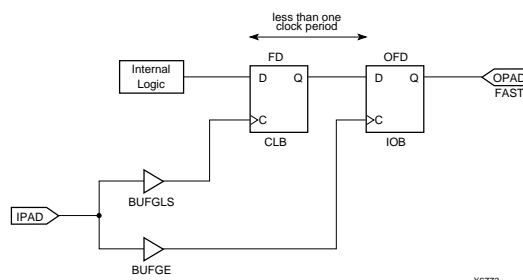
Delay From Clock Pin to Internal Flip-Flop

The delay from the clock pin to the output flip-flop is determined by the choice of clock buffer, and the loading on the clock net.

The most straightforward approach is to use the same clock signal, driven by the same clock buffer, for both the internal logic creating the output signal, and the output flip-flop. In this way, the maximum clock rate reported by timing analysis also applies to the output path.

Clocking the output flip-flop from a buffer with a shorter delay than the internal clock buffer speeds up the output path but reduces the available time between the last two flip-flops on the path. For example, as shown in [Figure 7](#), a single clock signal can be routed through both a BUFGLS and a BUFGE, with the BUFGLS signal used for the internal logic, and the BUFGE signal used for the output flip-flop. For the fastest possible clock-to-out path, route the clock through a BUFFCLK to the output flip-flop.

It is very important to remember that whenever either of these approaches is used, the time available to travel from the internal logic to the output flip-flop is shorter than the internal clock period. This path may prove to be the critical path in the design. Therefore, an early clock on the output flip-flop must be taken into consideration when calculating the internal clock speed for the design. Careful timing analysis is required.


Figure 7: Reducing Delay on Output Path (careful timing analysis is required)

Delay From Flip-Flop to Output Buffer

The XC4000EX, like all XC4000-Series devices, includes flip-flops in the IOB for the shortest possible clock-to-output delay. As in the XC4000E, these output flip-flops have clock enables, making them more generally useful than those in previous generations of FPGAs. The clock enable pins in XC4000EX IOBs have excellent access from the IOB local nets and longlines, so routing delay on the clock enable signal should not be a limiting factor.

The XC4000EX software may automatically place flip-flops in either the CLBs or the IOBs. To ensure that the IOB flip-flop is used, place a special library symbol such as OFD in the design.

Delay Through Output Buffer

All XC4000-Series devices offer an optional fast output. By default, the slew rate of each output is reduced, to minimize power transients when switching non-critical signals. Use the optional fast output mode for critical signals, bearing in mind that the transient output slew rate is significantly increased. All relevant timing is separately specified for fast and slew-rate limited outputs.

Select the fast output option by attaching a FAST attribute or property to the output buffer, flip-flop, or pad.

Fast Gated Functions

Some applications require the fastest possible path from one input pin, through simple combinatorial logic, to an output pin. One such case is an address-qualified read or write strobe from a processor.

For example, consider a system with a 16-bit address bus. It may be necessary to decode the sixteen bits, then wait for the qualifying write strobe. When the strobe occurs, the output must be generated with the shortest possible delay. This section describes how to implement this function in the XC4000EX.

As shown in Figure 5, the XC4000EX IOB includes a multiplexer on the output path. By placing the appropriate library element, any 2-input function can be implemented, including AND, NAND, OR, NOR, XOR, and XNOR, as well as a simple pass-gate. Or, of course, it can simply be used to multiplex two outputs, as described in the next section, "Increasing Effective I/O."

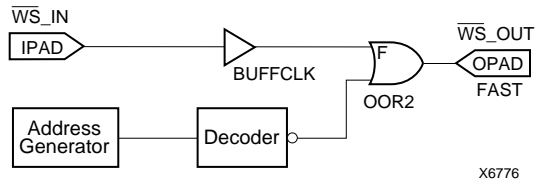


Figure 8: Fast Write Strobe Generator

The delay through the output multiplexer is different for each input, as can be seen from the diagram in Figure 5. Sometimes a single input even has different delays, depending on the path through the multiplexer. To simplify the use of the output multiplexer, the special library symbols denoting the multiplexer have an "F" marked on one input. (See Figure 8.) This input comes from the output clock (OK) pin of the IOB, and always has the shortest delay through the multiplexer.

The library symbols themselves have the same names as their CLB counterparts, but with an "O" (for Output) prefix. For example, an 2-input OR-gate implemented in the output multiplexer is called "OOR2." Library symbols with inverted ("bubbled") inputs are not available, in order to reduce library size, but an inverter placed on the input path is absorbed into the multiplexer. Logic is not moved into the IOB function generator unless one of these special symbols is used.

To implement the very fast write strobe circuit, the decode logic is mapped into CLBs or a wide edge decoder. In this example, the decode path is not critical.

The write strobe enters the device through a FastCLK buffer. As previously mentioned, the FastCLK buffers provide the fastest possible input path from a pad to the IOB clock lines. The buffered signal is ORed together with the decoded address line in the output multiplexer. The write strobe is attached to the pin labeled "F." Since the "F" pin of the OOR2 comes from the output clock pin, it is easily and quickly accessed from the IOB clock lines.

The result of the OR function is fed onto the output pad. The output pad is assigned a FAST attribute or property to remove the default slew-rate limitation.

The resulting delay, pin-to-pin, of the write strobe signal is less than 6 nanoseconds, a performance normally seen only in CPLDs or PALs.

Increasing Effective I/O

Many applications, particularly in large devices such as the XC4000EX family, require hundreds of input/output pins. This pinout proliferation does not come cheaply. Larger packages with more pins cost more, and PC boards are also more costly and complex. One method for overcoming these problems is multiplexing the I/Os. The number of I/Os effectively increases without increasing the number of package pins or board traces.

Microprocessors have used time-division multiplexing for years. Addresses and data are typically multiplexed on a single bus, for both read and write cycles. First the address is supplied, and is stored in the destination device. In the second part of the cycle, data is placed on the bus and a strobe initiates the read or write at the stored address.

Multiplexed I/Os transfer data more slowly than non-multiplexed I/Os. However, not all signals are speed-critical. I/Os with non-critical timing can benefit from this technique.

Multiplexing Outputs

Multiplexed outputs are easily created using the XC4000EX output multiplexer shown in [Figure 5](#).

Consider the multiplexed address/data example shown in [Figure 9](#). When the clock (select line) is High, the address is placed on the output pin; when Low, the data appears. To create this output, simply run both address and data bits into the output multiplexer (OMUX2). Use the output clock (OK) to select the address bit when High, the data bit when low.

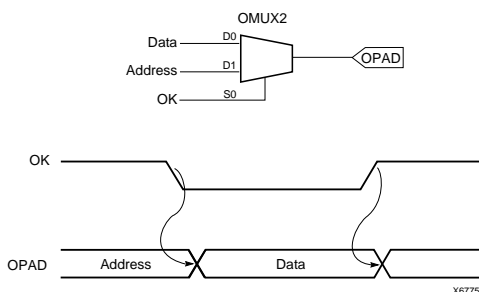


Figure 9: Multiplexing XC4000EX Outputs

De-Multiplexing Inputs

All XC4000-Series devices can de-multiplex inputs. (See [Figure 10](#).) An input signal can be routed in parallel, both through the input flip-flop or latch, and directly to the internal logic. The two input paths can be used for two different input signals.

In the address/data write cycle example, the input storage element is configured as a transparent-High latch (ILD). An address bit is presented at the input pad when the input clock (IK) is High. When IK goes Low, the address bit is latched at the latch output (I2).

The data bit is presented while the input clock is Low, and passes immediately through the direct input (I1) to the internal logic. The address is available on I2, and the data is available at I1, when the clock (IK) goes High. The rising clock edge registers the new data at the specified address.

Another Example

A second example is shown in [Figure 11](#). In this example, two input signals, A and B, are de-multiplexed from a single pin.

The Fast Capture latch captures the A input, which is then registered in the regular IOB flip-flop. The B input is registered in a CLB. Two simple logical functions are performed, and the result is multiplexed onto a single output pin as in the previous example.

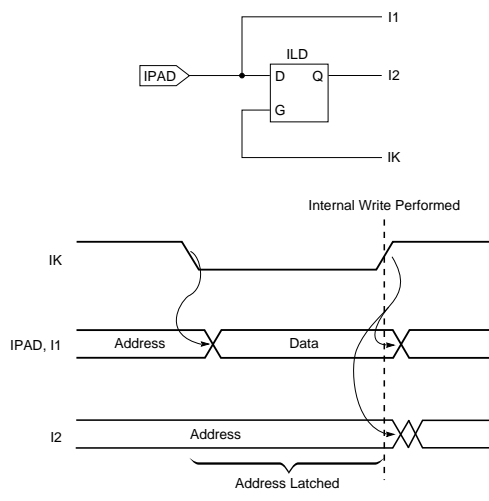


Figure 10: De-Multiplexing Inputs

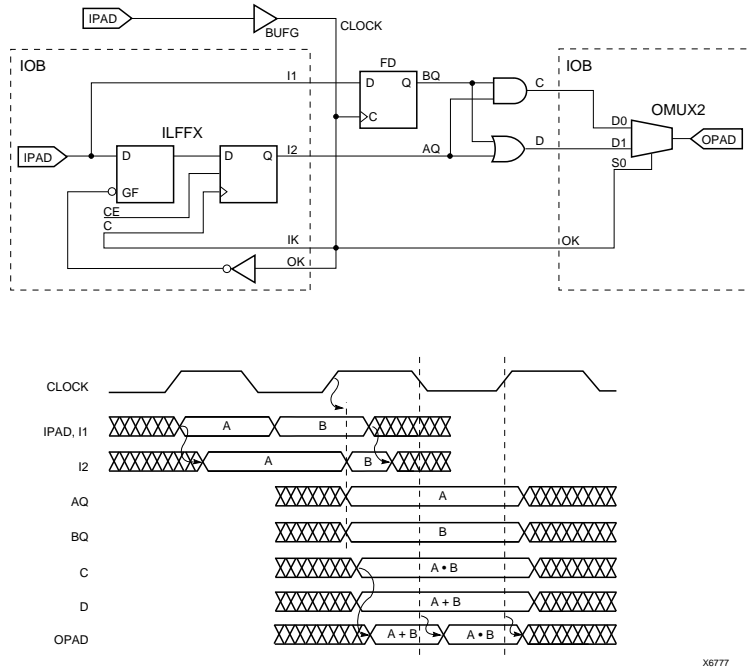
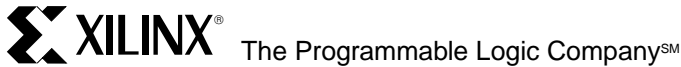


Figure 11: Simple De-Multiplexing/Multiplexing Example for the XC4000EX Family

X6777



Headquarters

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
U.S.A.

Tel: 1 (800) 255-7778
or 1 (408) 559-7778
Fax: 1 (800) 559-7114

Net: hotline@xilinx.com
Web: http://www.xilinx.com

North America

Irvine, California
(714) 727-0780

Englewood, Colorado
(303) 220-7541

Sunnyvale, California
(408) 245-9850

Schaumburg, Illinois
(847) 605-1972

Nashua, New Hampshire
(603) 891-1098

Raleigh, North Carolina
(919) 846-3922

West Chester, Pennsylvania
(610) 430-3300

Dallas, Texas
(214) 960-1043

Europe

Xilinx Sarl
Jouy en Josas, France
Tel: (33) 1-34-63-01-01
Net: frhelp@xilinx.com

Xilinx GmbH
Aschheim, Germany
Tel: (49) 89-99-1549-01
Net: dlhelp@xilinx.com

Xilinx, Ltd.
Byfleet, United Kingdom
Tel: (44) 1-932-349401
Net: ukhelp@xilinx.com

Japan

Xilinx, K.K.
Tokyo, Japan
Tel: (03) 3297-9191

Asia Pacific

Xilinx Asia Pacific
Hong Kong
Tel: (852) 2424-5200
Net: hongkong@xilinx.com

© 1996 Xilinx, Inc. All rights reserved. The Xilinx name and the Xilinx logo are registered trademarks, all XC-designated products are trademarks, and the Programmable Logic Company is a service mark of Xilinx, Inc. All other trademarks and registered trademarks are the property of their respective owners.

Xilinx, Inc. does not assume any liability arising out of the application or use of any product described herein; nor does it convey any license under its patent, copyright or maskwork rights or any rights of others. Xilinx, Inc. reserves the right to make changes, at any time, in order to improve reliability, function or design and to supply the best product possible. Xilinx, Inc. cannot assume responsibility for the use of any circuitry described other than circuitry entirely embodied in its products. Products are manufactured under one or more of the following U.S. Patents: (4,847,612; 5,012,135; 4,967,107; 5,023,606; 4,940,909; 5,028,821; 4,870,302; 4,706,216; 4,758,985; 4,642,487; 4,695,740; 4,713,557; 4,750,155; 4,821,233; 4,746,822; 4,820,937; 4,783,607; 4,855,669; 5,047,710; 5,068,603; 4,855,619; 4,835,418; and 4,902,910. Xilinx, Inc. cannot assume responsibility for any circuits shown nor represent that they are free from patent infringement or of any other third party right. Xilinx, Inc. assumes no obligation to correct any errors contained herein or to advise any user of this text of any correction if such be made.