



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	256
Number of Logic Elements/Cells	608
Total RAM Bits	8192
Number of I/O	113
Number of Gates	6000
Voltage - Supply	4.5V ~ 5.5V
Mounting Type	Surface Mount
Operating Temperature	-40°C ~ 100°C (TJ)
Package / Case	144-LQFP
Supplier Device Package	144-TQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/xilinx/xc4006e-4tq144i

Table 1: XC4000E and XC4000X Series Field Programmable Gate Arrays

Device	Logic Cells	Max Logic Gates (No RAM)	Max. RAM Bits (No Logic)	Typical Gate Range (Logic and RAM)*	CLB Matrix	Total CLBs	Number of Flip-Flops	Max. User I/O
XC4002XL	152	1,600	2,048	1,000 - 3,000	8 x 8	64	256	64
XC4003E	238	3,000	3,200	2,000 - 5,000	10 x 10	100	360	80
XC4005E/XL	466	5,000	6,272	3,000 - 9,000	14 x 14	196	616	112
XC4006E	608	6,000	8,192	4,000 - 12,000	16 x 16	256	768	128
XC4008E	770	8,000	10,368	6,000 - 15,000	18 x 18	324	936	144
XC4010E/XL	950	10,000	12,800	7,000 - 20,000	20 x 20	400	1,120	160
XC4013E/XL	1368	13,000	18,432	10,000 - 30,000	24 x 24	576	1,536	192
XC4020E/XL	1862	20,000	25,088	13,000 - 40,000	28 x 28	784	2,016	224
XC4025E	2432	25,000	32,768	15,000 - 45,000	32 x 32	1,024	2,560	256
XC4028EX/XL	2432	28,000	32,768	18,000 - 50,000	32 x 32	1,024	2,560	256
XC4036EX/XL	3078	36,000	41,472	22,000 - 65,000	36 x 36	1,296	3,168	288
XC4044XL	3800	44,000	51,200	27,000 - 80,000	40 x 40	1,600	3,840	320
XC4052XL	4598	52,000	61,952	33,000 - 100,000	44 x 44	1,936	4,576	352
XC4062XL	5472	62,000	73,728	40,000 - 130,000	48 x 48	2,304	5,376	384
XC4085XL	7448	85,000	100,352	55,000 - 180,000	56 x 56	3,136	7,168	448

* Max values of Typical Gate Range include 20-30% of CLBs used as RAM.

Note: All functionality in low-voltage families is the same as in the corresponding 5-Volt family, except where numerical references are made to timing or power.

Description

XC4000 Series devices are implemented with a regular, flexible, programmable architecture of Configurable Logic Blocks (CLBs), interconnected by a powerful hierarchy of versatile routing resources, and surrounded by a perimeter of programmable Input/Output Blocks (IOBs). They have generous routing resources to accommodate the most complex interconnect patterns.

The devices are customized by loading configuration data into internal memory cells. The FPGA can either actively read its configuration data from an external serial or byte-parallel PROM (master modes), or the configuration data can be written into the FPGA from an external device (slave and peripheral modes).

XC4000 Series FPGAs are supported by powerful and sophisticated software, covering every aspect of design from schematic or behavioral entry, floor planning, simulation, automatic block placement and routing of interconnects, to the creation, downloading, and readback of the configuration bit stream.

Because Xilinx FPGAs can be reprogrammed an unlimited number of times, they can be used in innovative designs

where hardware is changed dynamically, or where hardware must be adapted to different user applications. FPGAs are ideal for shortening design and development cycles, and also offer a cost-effective solution for production rates well beyond 5,000 systems per month.

Taking Advantage of Re-configuration

FPGA devices can be re-configured to change logic function while resident in the system. This capability gives the system designer a new degree of freedom not available with any other type of logic.

Hardware can be changed as easily as software. Design updates or modifications are easy, and can be made to products already in the field. An FPGA can even be re-configured dynamically to perform different functions at different times.

Re-configurable logic can be used to implement system self-diagnostics, create systems capable of being re-configured for different environments or operations, or implement multi-purpose hardware for a given application. As an added benefit, using re-configurable FPGA devices simplifies hardware design and debugging and shortens product time-to-market.

Input Thresholds

The input thresholds of 5V devices can be globally configured for either TTL (1.2 V threshold) or CMOS (2.5 V threshold), just like XC2000 and XC3000 inputs. The two global adjustments of input threshold and output level are independent of each other. The XC4000XL family has an input threshold of 1.6V, compatible with both 3.3V CMOS and TTL levels.

Global Signal Access to Logic

There is additional access from global clocks to the F and G function generator inputs.

Configuration Pin Pull-Up Resistors

During configuration, these pins have weak pull-up resistors. For the most popular configuration mode, Slave Serial, the mode pins can thus be left unconnected. The three mode inputs can be individually configured with or without weak pull-up or pull-down resistors. A pull-down resistor value of 4.7 k Ω is recommended.

The three mode inputs can be individually configured with or without weak pull-up or pull-down resistors after configuration.

The PROGRAM input pin has a permanent weak pull-up.

Soft Start-up

Like the XC3000A, XC4000 Series devices have "Soft Start-up." When the configuration process is finished and the device starts up, the first activation of the outputs is automatically slew-rate limited. This feature avoids potential ground bounce when all outputs are turned on simultaneously. Immediately after start-up, the slew rate of the individual outputs is, as in the XC4000 family, determined by the individual configuration option.

XC4000 and XC4000A Compatibility

Existing XC4000 bitstreams can be used to configure an XC4000E device. XC4000A bitstreams must be recompiled for use with the XC4000E due to improved routing resources, although the devices are pin-for-pin compatible.

Additional Improvements in XC4000X Only

Increased Routing

New interconnect in the XC4000X includes twenty-two additional vertical lines in each column of CLBs and twelve new horizontal lines in each row of CLBs. The twelve "Quad Lines" in each CLB row and column include optional repowering buffers for maximum speed. Additional high-performance routing near the IOBs enhances pin flexibility.

Faster Input and Output

A fast, dedicated early clock sourced by global clock buffers is available for the IOBs. To ensure synchronization with the regular global clocks, a Fast Capture latch driven by the early clock is available. The input data can be initially loaded into the Fast Capture latch with the early clock, then transferred to the input flip-flop or latch with the low-skew global clock. A programmable delay on the input can be used to avoid hold-time requirements. See "IOB Input Signals" on page 20 for more information.

Latch Capability in CLBs

Storage elements in the XC4000X CLB can be configured as either flip-flops or latches. This capability makes the FPGA highly synthesis-compatible.

IOB Output MUX From Output Clock

A multiplexer in the IOB allows the output clock to select either the output data or the IOB clock enable as the output to the pad. Thus, two different data signals can share a single output pad, effectively doubling the number of device outputs without requiring a larger, more expensive package. This multiplexer can also be configured as an AND-gate to implement a very fast pin-to-pin path. See "IOB Output Signals" on page 23 for more information.

Additional Address Bits

Larger devices require more bits of configuration data. A daisy chain of several large XC4000X devices may require a PROM that cannot be addressed by the eighteen address bits supported in the XC4000E. The XC4000X Series therefore extends the addressing in Master Parallel configuration mode to 22 bits.

Set/Reset

An asynchronous storage element input (SR) can be configured as either set or reset. This configuration option determines the state in which each flip-flop becomes operational after configuration. It also determines the effect of a Global Set/Reset pulse during normal operation, and the effect of a pulse on the SR pin of the CLB. All three set/reset functions for any single flip-flop are controlled by the same configuration data bit.

The set/reset state can be independently specified for each flip-flop. This input can also be independently disabled for either flip-flop.

The set/reset state is specified by using the INIT attribute, or by placing the appropriate set or reset flip-flop library symbol.

SR is active High. It is not invertible within the CLB.

Global Set/Reset

A separate Global Set/Reset line (not shown in Figure 1) sets or clears each storage element during power-up, re-configuration, or when a dedicated Reset net is driven active. This global net (GSR) does not compete with other routing resources; it uses a dedicated distribution network.

Each flip-flop is configured as either globally set or reset in the same way that the local set/reset (SR) is specified. Therefore, if a flip-flop is set by SR, it is also set by GSR. Similarly, a reset flip-flop is reset by both SR and GSR.

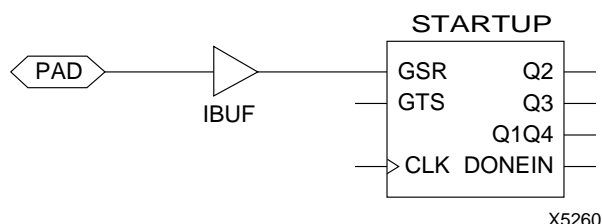


Figure 2: Schematic Symbols for Global Set/Reset

GSR can be driven from any user-programmable pin as a global reset input. To use this global net, place an input pad and input buffer in the schematic or HDL code, driving the GSR pin of the STARTUP symbol. (See Figure 2.) A specific pin location can be assigned to this input using a LOC attribute or property, just as with any other user-programmable pad. An inverter can optionally be inserted after the input buffer to invert the sense of the Global Set/Reset signal.

Alternatively, GSR can be driven from any internal node.

Data Inputs and Outputs

The source of a storage element data input is programmable. It is driven by any of the functions F', G', and H', or by the Direct In (DIN) block input. The flip-flops or latches drive the XQ and YQ CLB outputs.

Two fast feed-through paths are available, as shown in Figure 1. A two-to-one multiplexer on each of the XQ and YQ outputs selects between a storage element output and any of the control inputs. This bypass is sometimes used by the automated router to repower internal signals.

Control Signals

Multiplexers in the CLB map the four control inputs (C1 - C4 in Figure 1) into the four internal control signals (H1, DIN/H2, SR/H0, and EC). Any of these inputs can drive any of the four internal control signals.

When the logic function is enabled, the four inputs are:

- EC — Enable Clock
- SR/H0 — Asynchronous Set/Reset or H function generator Input 0
- DIN/H2 — Direct In or H function generator Input 2
- H1 — H function generator Input 1.

When the memory function is enabled, the four inputs are:

- EC — Enable Clock
- WE — Write Enable
- D0 — Data Input to F and/or G function generator
- D1 — Data input to G function generator (16x1 and 16x2 modes) or 5th Address bit (32x1 mode).

Using FPGA Flip-Flops and Latches

The abundance of flip-flops in the XC4000 Series invites pipelined designs. This is a powerful way of increasing performance by breaking the function into smaller subfunctions and executing them in parallel, passing on the results through pipeline flip-flops. This method should be seriously considered wherever throughput is more important than latency.

To include a CLB flip-flop, place the appropriate library symbol. For example, FDCE is a D-type flip-flop with clock enable and asynchronous clear. The corresponding latch symbol (for the XC4000X only) is called LDCE.

In XC4000 Series devices, the flip flops can be used as registers or shift registers without blocking the function generators from performing a different, perhaps unrelated task. This ability increases the functional capacity of the devices.

The CLB setup time is specified between the function generator inputs and the clock input K. Therefore, the specified CLB flip-flop setup time includes the delay through the function generator.

Using Function Generators as RAM

Optional modes for each CLB make the memory look-up tables in the F' and G' function generators usable as an array of Read/Write memory cells. Available modes are level-sensitive (similar to the XC4000/A/H families), edge-triggered, and dual-port edge-triggered. Depending on the selected mode, a single CLB can be configured as either a 16x2, 32x1, or 16x1 bit array.

Additional Input Latch for Fast Capture (XC4000X only)

The XC4000X IOB has an additional optional latch on the input. This latch, as shown in [Figure 16](#), is clocked by the output clock — the clock used for the output flip-flop — rather than the input clock. Therefore, two different clocks can be used to clock the two input storage elements. This additional latch allows the very fast capture of input data, which is then synchronized to the internal clock by the IOB flip-flop or latch.

To use this Fast Capture technique, drive the output clock pin (the Fast Capture latching signal) from the output of one of the Global Early buffers supplied in the XC4000X. The second storage element should be clocked by a Global Low-Skew buffer, to synchronize the incoming data to the internal logic. (See [Figure 17](#).) These special buffers are described in “Global Nets and Buffers (XC4000X only)” on [page 37](#).

The Fast Capture latch (FCL) is designed primarily for use with a Global Early buffer. For Fast Capture, a single clock signal is routed through both a Global Early buffer and a Global Low-Skew buffer. (The two buffers share an input pad.) The Fast Capture latch is clocked by the Global Early buffer, and the standard IOB flip-flop or latch is clocked by the Global Low-Skew buffer. This mode is the safest way to use the Fast Capture latch, because the clock buffers on both storage elements are driven by the same pad. There is no external skew between clock pads to create potential problems.

To place the Fast Capture latch in a design, use one of the special library symbols, ILFFX or ILFLX. ILFFX is a transparent-Low Fast Capture latch followed by an active-High input flip-flop. ILFLX is a transparent-Low Fast Capture latch followed by a transparent-High input latch. Any of the clock inputs can be inverted before driving the library element, and the inverter is absorbed into the IOB. If a single BUFG output is used to drive both clock inputs, the software automatically runs the clock through both a Global Low-Skew buffer and a Global Early buffer, and clocks the Fast Capture latch appropriately.

[Figure 16 on page 21](#) also shows a two-tap delay on the input. By default, if the Fast Capture latch is used, the Xilinx software assumes a Global Early buffer is driving the clock, and selects MEDDELAY to ensure a zero hold time. Select

the desired delay based on the discussion in the previous subsection.

IOB Output Signals

Output signals can be optionally inverted within the IOB, and can pass directly to the pad or be stored in an edge-triggered flip-flop. The functionality of this flip-flop is shown in [Table 11](#).

An active-High 3-state signal can be used to place the output buffer in a high-impedance state, implementing 3-state outputs or bidirectional I/O. Under configuration control, the output (OUT) and output 3-state (T) signals can be inverted. The polarity of these signals is independently configured for each IOB.

The 4-mA maximum output current specification of many FPGAs often forces the user to add external buffers, which are especially cumbersome on bidirectional I/O lines. The XC4000E and XC4000EX/XL devices solve many of these problems by providing a guaranteed output sink current of 12 mA. Two adjacent outputs can be interconnected externally to sink up to 24 mA. The XC4000E and XC4000EX/XL FPGAs can thus directly drive buses on a printed circuit board.

By default, the output pull-up structure is configured as a TTL-like totem-pole. The High driver is an n-channel pull-up transistor, pulling to a voltage one transistor threshold below V_{cc}. Alternatively, the outputs can be globally configured as CMOS drivers, with p-channel pull-up transistors pulling to V_{cc}. This option, applied using the bitstream generation software, applies to all outputs on the device. It is not individually programmable. In the XC4000XL, all outputs are pulled to the positive supply rail.

Table 11: Output Flip-Flop Functionality (active rising edge is shown)

Mode	Clock	Clock Enable	T	D	Q
Power-Up or GSR	X	X	0*	X	SR
Flip-Flop	X	0	0*	X	Q
		1*	0*	D	D
	X	X	1	X	Z
	0	X	0*	X	Q

Legend:

X

Don't care

Rising edge

SR

Set or Reset value. Reset is default.

0*

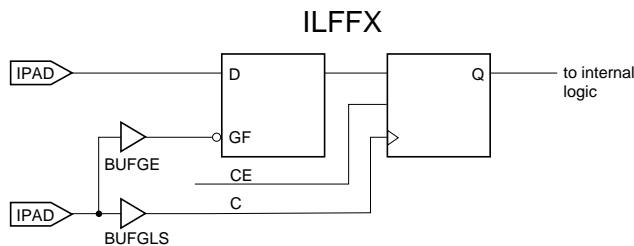
Input is Low or unconnected (default value)

1*

Input is High or unconnected (default value)

Z

3-state



X9013

Figure 17: Examples Using XC4000X FCL

or clear on reset and after configuration. Other than the global GSR net, no user-controlled set/reset signal is available to the I/O flip-flops. The choice of set or clear applies to both the initial state of the flip-flop and the response to the Global Set/Reset pulse. See [“Global Set/Reset” on page 11](#) for a description of how to use GSR.

JTAG Support

Embedded logic attached to the IOBs contains test structures compatible with IEEE Standard 1149.1 for boundary scan testing, permitting easy chip and board-level testing. More information is provided in [“Boundary Scan” on page 42](#).

Three-State Buffers

A pair of 3-state buffers is associated with each CLB in the array. (See [Figure 27 on page 30](#).) These 3-state buffers can be used to drive signals onto the nearest horizontal longlines above and below the CLB. They can therefore be used to implement multiplexed or bidirectional buses on the horizontal longlines, saving logic resources. Programmable pull-up resistors attached to these longlines help to implement a wide wired-AND function.

The buffer enable is an active-High 3-state (i.e. an active-Low enable), as shown in [Table 13](#).

Another 3-state buffer with similar access is located near each I/O block along the right and left edges of the array. (See [Figure 33 on page 34](#).)

The horizontal longlines driven by the 3-state buffers have a weak keeper at each end. This circuit prevents undefined floating levels. However, it is overridden by any driver, even a pull-up resistor.

Special longlines running along the perimeter of the array can be used to wire-AND signals coming from nearby IOBs or from internal longlines. These longlines form the wide edge decoders discussed in [“Wide Edge Decoders” on page 27](#).

Three-State Buffer Modes

The 3-state buffers can be configured in three modes:

- Standard 3-state buffer
- Wired-AND with input on the I pin
- Wired OR-AND

Standard 3-State Buffer

All three pins are used. Place the library element BUFT. Connect the input to the I pin and the output to the O pin. The T pin is an active-High 3-state (i.e. an active-Low enable). Tie the T pin to Ground to implement a standard buffer.

Wired-AND with Input on the I Pin

The buffer can be used as a Wired-AND. Use the WAND1 library symbol, which is essentially an open-drain buffer. WAND4, WAND8, and WAND16 are also available. See the *XACT Libraries Guide* for further information.

The T pin is internally tied to the I pin. Connect the input to the I pin and the output to the O pin. Connect the outputs of all the WAND1s together and attach a PULLUP symbol.

Wired OR-AND

The buffer can be configured as a Wired OR-AND. A High level on either input turns off the output. Use the WOR2AND library symbol, which is essentially an open-drain 2-input OR gate. The two input pins are functionally equivalent. Attach the two inputs to the I0 and I1 pins and tie the output to the O pin. Tie the outputs of all the WOR2ANDs together and attach a PULLUP symbol.

Three-State Buffer Examples

[Figure 21](#) shows how to use the 3-state buffers to implement a wired-AND function. When all the buffer inputs are High, the pull-up resistor(s) provide the High output.

[Figure 22](#) shows how to use the 3-state buffers to implement a multiplexer. The selection is accomplished by the buffer 3-state signal.

Pay particular attention to the polarity of the T pin when using these buffers in a design. Active-High 3-state (T) is identical to an active-Low output enable, as shown in [Table 13](#).

Table 13: Three-State Buffer Functionality

IN	T	OUT
X	1	Z
IN	0	IN



Figure 21: Open-Drain Buffers Implement a Wired-AND Function

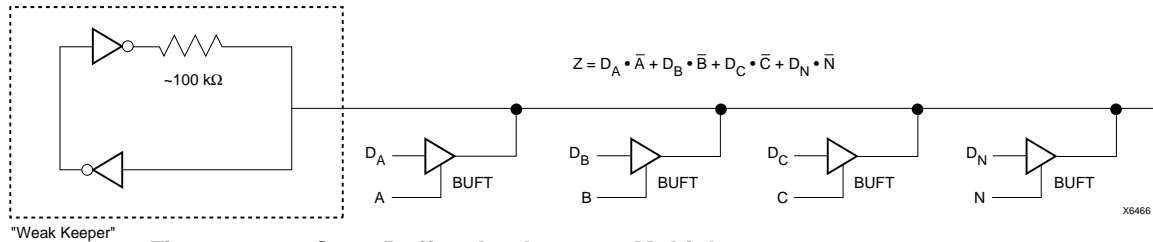


Figure 22: 3-State Buffers Implement a Multiplexer

Wide Edge Decoders

Dedicated decoder circuitry boosts the performance of wide decoding functions. When the address or data field is wider than the function generator inputs, FPGAs need multi-level decoding and are thus slower than PALs. XC4000 Series CLBs have nine inputs. Any decoder of up to nine inputs is, therefore, compact and fast. However, there is also a need for much wider decoders, especially for address decoding in large microprocessor systems.

An XC4000 Series FPGA has four programmable decoders located on each edge of the device. The inputs to each decoder are any of the IOB I1 signals on that edge plus one local interconnect per CLB row or column. Each row or column of CLBs provides up to three variables or their complements., as shown in Figure 23. Each decoder generates a High output (resistor pull-up) when the AND condition of the selected inputs, or their complements, is true. This is analogous to a product term in typical PAL devices.

Each of these wired-AND gates is capable of accepting up to 42 inputs on the XC4005E and 72 on the XC4013E. There are up to 96 inputs for each decoder on the XC4028X and 132 on the XC4052X. The decoders may also be split in two when a larger number of narrower decoders are required, for a maximum of 32 decoders per device.

The decoder outputs can drive CLB inputs, so they can be combined with other logic to form a PAL-like AND/OR structure. The decoder outputs can also be routed directly to the chip outputs. For fastest speed, the output should be on the same chip edge as the decoder. Very large PALs can be emulated by ORing the decoder outputs in a CLB. This decoding feature covers what has long been considered a weakness of older FPGAs. Users often resorted to external PALs for simple but fast decoding functions. Now, the dedicated decoders in the XC4000 Series device can implement these functions fast and efficiently.

To use the wide edge decoders, place one or more of the WAND library symbols (WAND1, WAND4, WAND8, WAND16). Attach a DECODE attribute or property to each WAND symbol. Tie the outputs together and attach a PUL-

LUP symbol. Location attributes or properties such as L (left edge) or TR (right half of top edge) should also be used to ensure the correct placement of the decoder inputs.

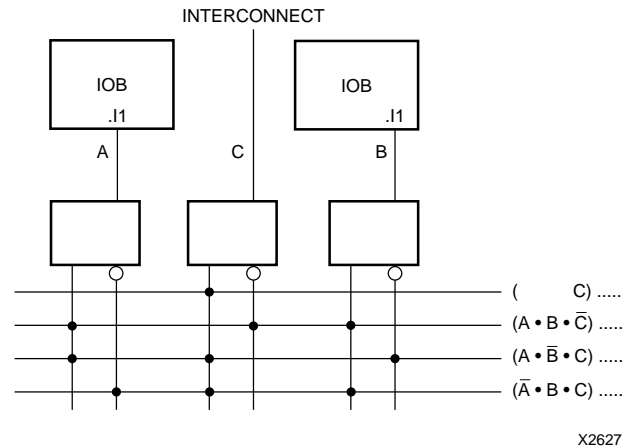


Figure 23: XC4000 Series Edge Decoding Example

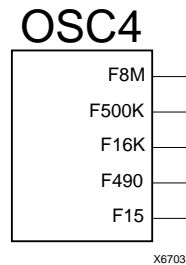


Figure 24: XC4000 Series Oscillator Symbol

On-Chip Oscillator

XC4000 Series devices include an internal oscillator. This oscillator is used to clock the power-on time-out, for configuration memory clearing, and as the source of CCLK in Master configuration modes. The oscillator runs at a nominal 8 MHz frequency that varies with process, Vcc, and temperature. The output frequency falls between 4 and 10 MHz.



x5994

Figure 25: High-Level Routing Diagram of XC4000 Series CLB (shaded arrows indicate XC4000X only)

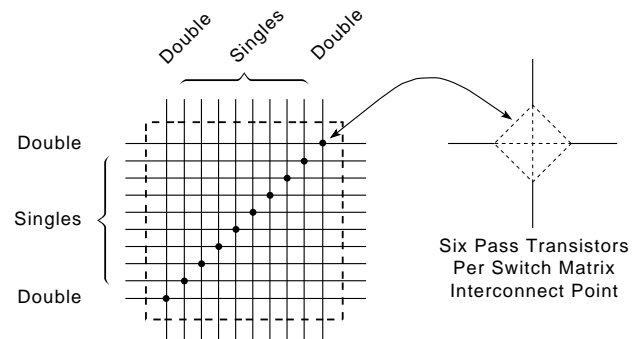
Table 14: Routing per CLB in XC4000 Series Devices

	XC4000E		XC4000X	
	Vertical	Horizontal	Vertical	Horizontal
Singles	8	8	8	8
Doubles	4	4	4	4
Quads	0	0	12	12
Longlines	6	6	10	6
Direct Connects	0	0	2	2
Globals	4	0	8	0
Carry Logic	2	0	1	0
Total	24	18	45	32

Programmable Switch Matrices

The horizontal and vertical single- and double-length lines intersect at a box called a programmable switch matrix (PSM). Each switch matrix consists of programmable pass transistors used to establish connections between the lines (see [Figure 26](#)).

For example, a single-length signal entering on the right side of the switch matrix can be routed to a single-length line on the top, left, or bottom sides, or any combination thereof, if multiple branches are required. Similarly, a double-length signal can be routed to a double-length line on any or all of the other three edges of the programmable switch matrix.



X6600

Figure 26: Programmable Switch Matrix (PSM)

Single-Length Lines

Single-length lines provide the greatest interconnect flexibility and offer fast routing between adjacent blocks. There are eight vertical and eight horizontal single-length lines associated with each CLB. These lines connect the switching matrices that are located in every row and a column of CLBs.

Single-length lines are connected by way of the programmable switch matrices, as shown in [Figure 28](#). Routing connectivity is shown in [Figure 27](#).

Single-length lines incur a delay whenever they go through a switching matrix. Therefore, they are not suitable for routing signals for long distances. They are normally used to conduct signals within a localized area and to provide the branching for nets with fanout greater than one.

circuit prevents undefined floating levels. However, it is overridden by any driver, even a pull-up resistor.

Each XC4000E longline has a programmable splitter switch at its center, as does each XC4000X longline driven by TBUFs. This switch can separate the line into two independent routing channels, each running half the width or height of the array.

Each XC4000X longline not driven by TBUFs has a buffered programmable splitter switch at the 1/4, 1/2, and 3/4 points of the array. Due to the buffering, XC4000X longline performance does not deteriorate with the larger array sizes. If the longline is split, the resulting partial longlines are independent.

Routing connectivity of the longlines is shown in [Figure 27 on page 30](#).

Direct Interconnect (XC4000X only)

The XC4000X offers two direct, efficient and fast connections between adjacent CLBs. These nets facilitate a data flow from the left to the right side of the device, or from the top to the bottom, as shown in [Figure 30](#). Signals routed on the direct interconnect exhibit minimum interconnect propagation delay and use no general routing resources.

The direct interconnect is also present between CLBs and adjacent IOBs. Each IOB on the left and top device edges has a direct path to the nearest CLB. Each CLB on the right and bottom edges of the array has a direct path to the nearest two IOBs, since there are two IOBs for each row or column of CLBs.

The place and route software uses direct interconnect whenever possible, to maximize routing resources and minimize interconnect delays.



Figure 30: XC4000X Direct Interconnect

I/O Routing

XC4000 Series devices have additional routing around the IOB ring. This routing is called a VersaRing. The VersaRing facilitates pin-swapping and redesign without affecting board layout. Included are eight double-length lines spanning two CLBs (four IOBs), and four longlines. Global lines and Wide Edge Decoder lines are provided. XC4000X devices also include eight octal lines.

A high-level diagram of the VersaRing is shown in [Figure 31](#). The shaded arrows represent routing present only in XC4000X devices.

[Figure 33 on page 34](#) is a detailed diagram of the XC4000E and XC4000X VersaRing. The area shown includes two IOBs. There are two IOBs per CLB row or column, therefore this diagram corresponds to the CLB routing diagram shown in [Figure 27 on page 30](#). The shaded areas represent routing and routing connections present only in XC4000X devices.

Octal I/O Routing (XC4000X only)

Between the XC4000X CLB array and the pad ring, eight interconnect tracks provide for versatility in pin assignment and fixed pinout flexibility. (See [Figure 32 on page 33](#).)

These routing tracks are called octals, because they can be broken every eight CLBs (sixteen IOBs) by a programmable buffer that also functions as a splitter switch. The buffers are staggered, so each line goes through a buffer at every eighth CLB location around the device edge.

The octal lines bend around the corners of the device. The lines cross at the corners in such a way that the segment most recently buffered before the turn has the farthest distance to travel before the next buffer, as shown in [Figure 32](#).

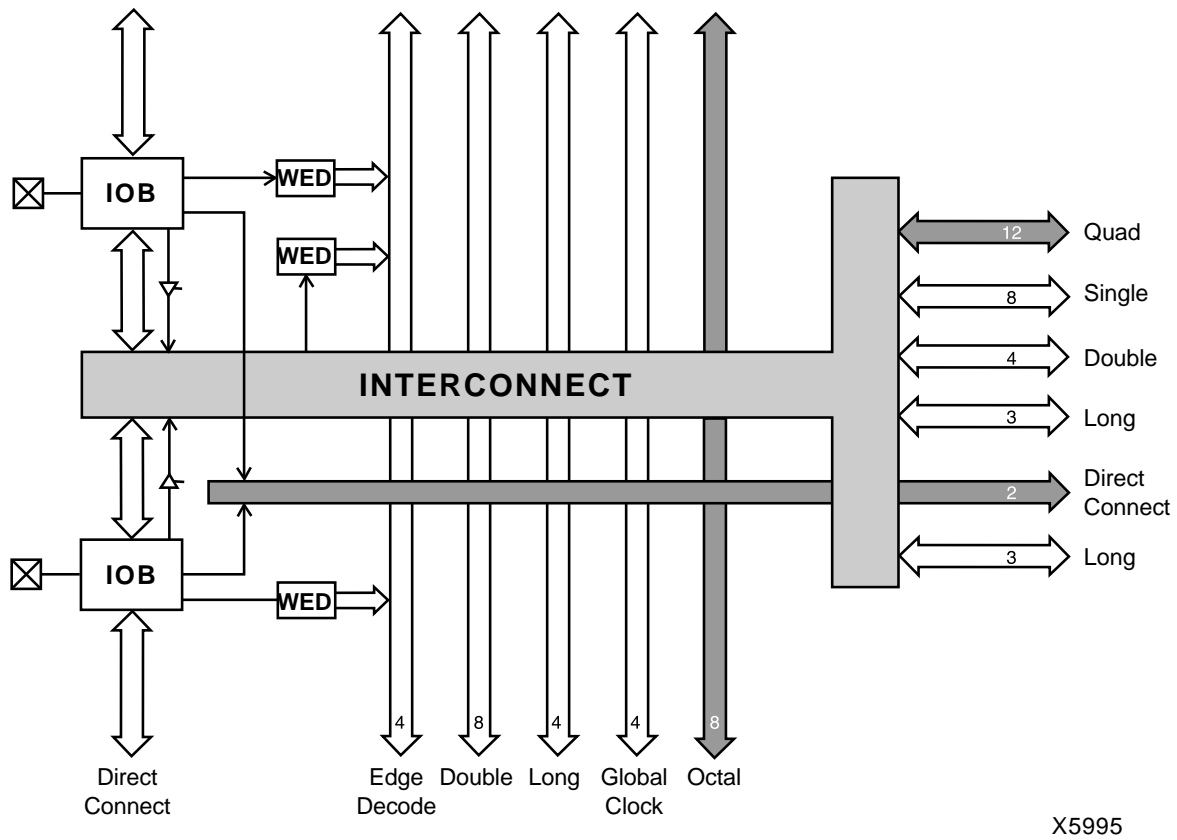


Figure 31: High-Level Routing Diagram of XC4000 Series VersaRing (Left Edge)
WED = Wide Edge Decoder, IOB = I/O Block (shaded arrows indicate XC4000X only)

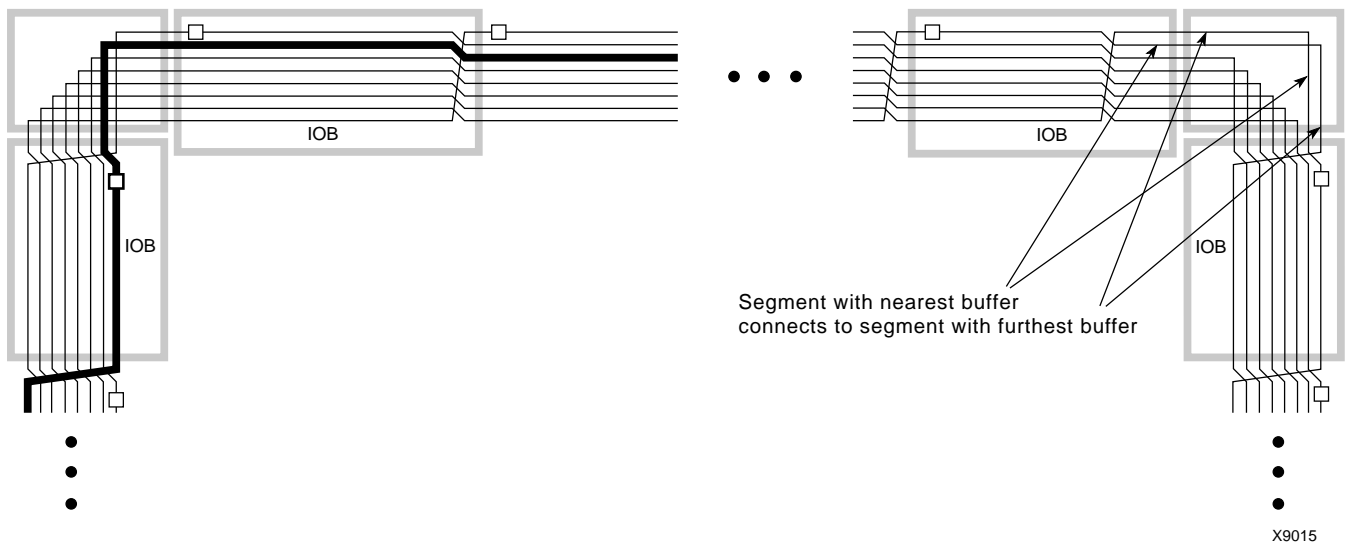


Figure 32: XC4000X Octal I/O Routing

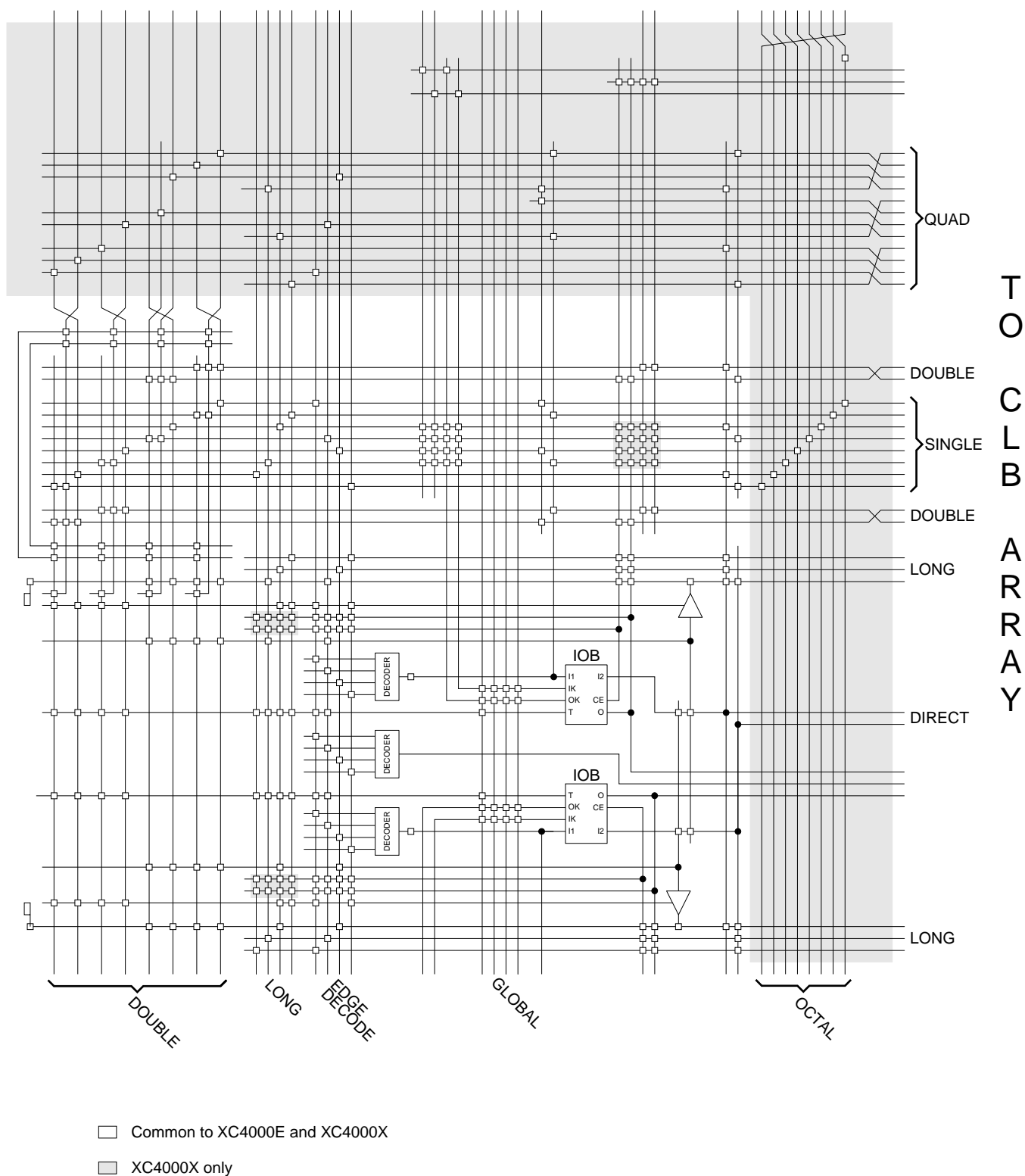


Figure 33: Detail of Programmable Interconnect Associated with XC4000 Series IOB (Left Edge)

IOB inputs and outputs interface with the octal lines via the single-length interconnect lines. Single-length lines are also used for communication between the octals and double-length lines, quads, and longlines within the CLB array.

Segmentation into buffered octals was found to be optimal for distributing signals over long distances around the device.

Global Nets and Buffers

Both the XC4000E and the XC4000X have dedicated global networks. These networks are designed to distribute clocks and other high fanout control signals throughout the devices with minimal skew. The global buffers are described in detail in the following sections. The text descriptions and diagrams are summarized in [Table 15](#). The table shows which CLB and IOB clock pins can be sourced by which global buffers.

In both XC4000E and XC4000X devices, placement of a library symbol called BUFG results in the software choosing the appropriate clock buffer, based on the timing requirements of the design. The detailed information in these sections is included only for reference.

Global Nets and Buffers (XC4000E only)

Four vertical longlines in each CLB column are driven exclusively by special global buffers. These longlines are in addition to the vertical longlines used for standard interconnect. The four global lines can be driven by either of two types of global buffers. The clock pins of every CLB and IOB can also be sourced from local interconnect.

Two different types of clock buffers are available in the XC4000E:

- Primary Global Buffers (BUFGP)
- Secondary Global Buffers (BUFGS)

Four Primary Global buffers offer the shortest delay and negligible skew. Four Secondary Global buffers have slightly longer delay and slightly more skew due to potentially heavier loading, but offer greater flexibility when used to drive non-clock CLB inputs.

The Primary Global buffers must be driven by the semi-dedicated pads. The Secondary Global buffers can be sourced by either semi-dedicated pads or internal nets.

Each CLB column has four dedicated vertical Global lines. Each of these lines can be accessed by one particular Primary Global buffer, or by any of the Secondary Global buffers, as shown in [Figure 34](#). Each corner of the device has one Primary buffer and one Secondary buffer.

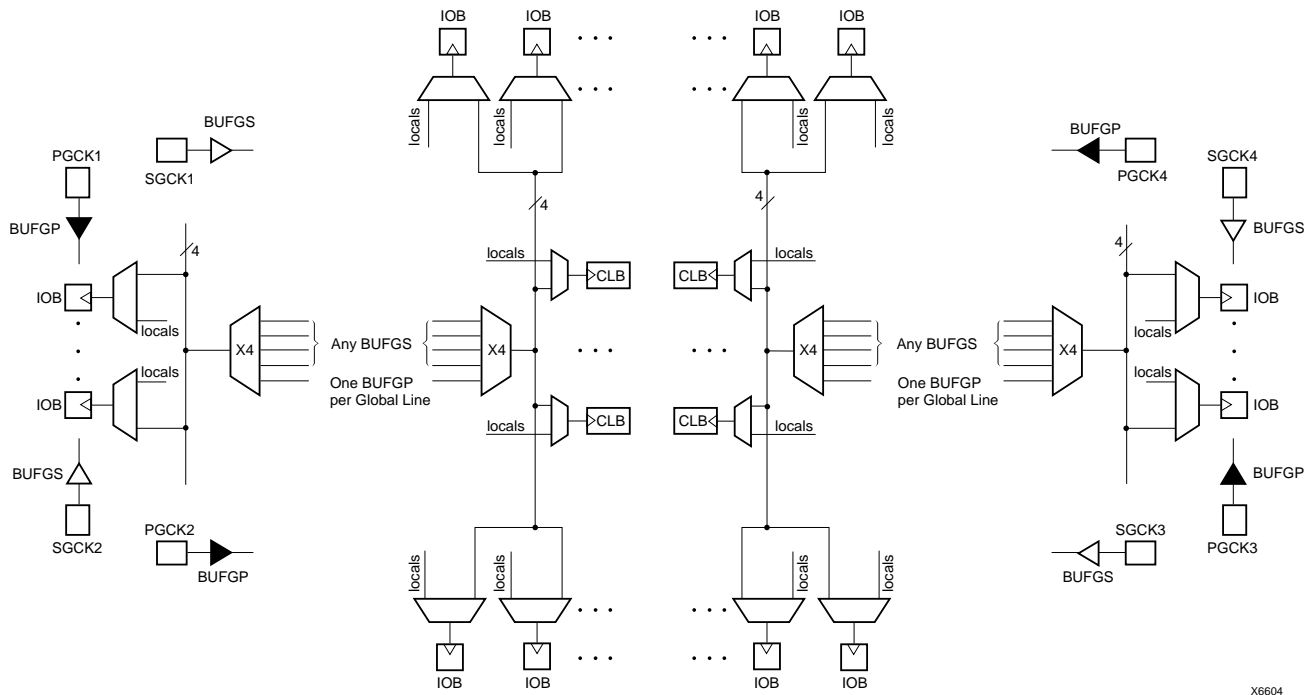
IOBs along the left and right edges have four vertical global longlines. Top and bottom IOBs can be clocked from the global lines in the adjacent CLB column.

A global buffer should be specified for all timing-sensitive global signal distribution. To use a global buffer, place a BUFGP (primary buffer), BUFGS (secondary buffer), or BUFG (either primary or secondary buffer) element in a schematic or in HDL code. If desired, attach a LOC attribute or property to direct placement to the designated location. For example, attach a LOC=L attribute or property to a BUFGS symbol to direct that a buffer be placed in one of the two Secondary Global buffers on the left edge of the device, or a LOC=BL to indicate the Secondary Global buffer on the bottom edge of the device, on the left.

Table 15: Clock Pin Access

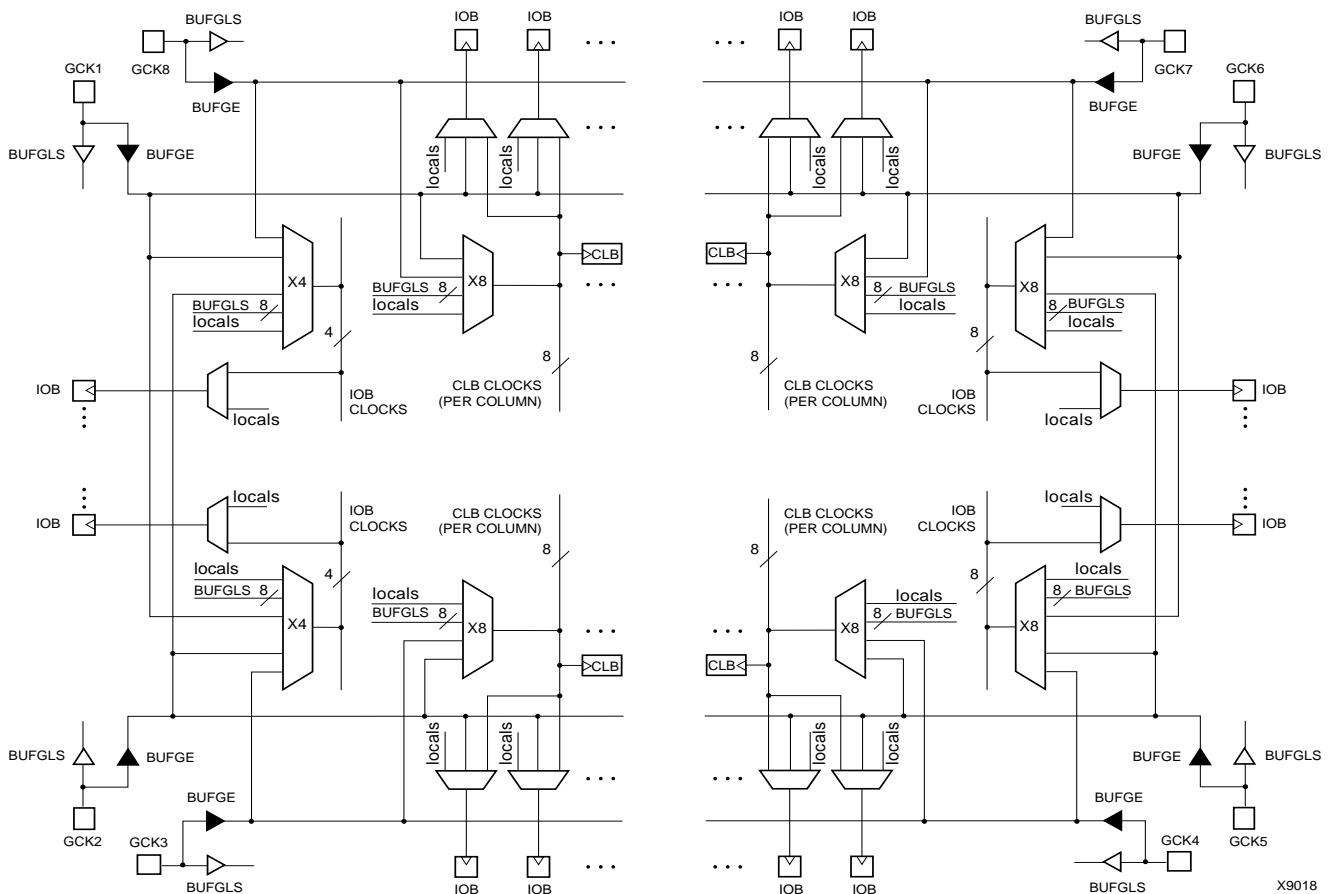
	XC4000E		XC4000X			Local Inter-connect
	BUFGP	BUFGS	BUFGLS	L & R BUFGE	T & B BUFGE	
All CLBs in Quadrant	√	√	√	√	√	√
All CLBs in Device	√	√	√			√
IOBs on Adjacent Vertical Half Edge	√	√	√	√	√	√
IOBs on Adjacent Vertical Full Edge	√	√	√	√		√
IOBs on Adjacent Horizontal Half Edge (Direct)				√		√
IOBs on Adjacent Horizontal Half Edge (through CLB globals)	√	√	√	√	√	√
IOBs on Adjacent Horizontal Full Edge (through CLB globals)	√	√	√			√

L = Left, R = Right, T = Top, B = Bottom



X6604

Figure 34: XC4000E Global Net Distribution



X9018

Figure 35: XC4000X Global Net Distribution

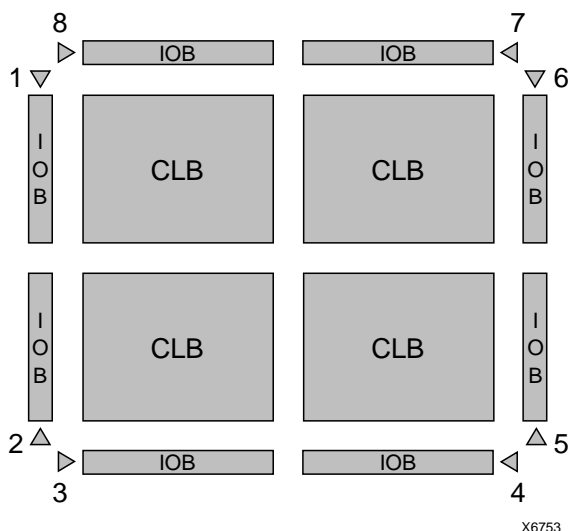


Figure 36: Any BUFGLS (GCK1 - GCK8) Can Drive Any or All Clock Inputs on the Device

Global Early Buffers

Each corner of the XC4000X device has two Global Early buffers. The primary purpose of the Global Early buffers is to provide an earlier clock access than the potentially heavily-loaded Global Low-Skew buffers. A clock source applied to both buffers will result in the Global Early clock edge occurring several nanoseconds earlier than the Global Low-Skew buffer clock edge, due to the lighter loading.

Global Early buffers also facilitate the fast capture of device inputs, using the Fast Capture latches described in **"IOB Input Signals"** on page 20. For Fast Capture, take a single clock signal, and route it through both a Global Early buffer and a Global Low-Skew buffer. (The two buffers share an input pad.) Use the Global Early buffer to clock the Fast Capture latch, and the Global Low-Skew buffer to clock the normal input flip-flop or latch, as shown in **Figure 17** on page 23.

The Global Early buffers can also be used to provide a fast Clock-to-Out on device output pins. However, an early clock in the output flip-flop IOB must be taken into consideration when calculating the internal clock speed for the design.

The Global Early buffers at the left and right edges of the chip have slightly different capabilities than the ones at the top and bottom. Refer to **Figure 37**, **Figure 38**, and **Figure 35** on page 36 while reading the following explanation.

Each Global Early buffer can access the eight vertical Global lines for all CLBs in the quadrant. Therefore, only one-fourth of the CLB clock pins can be accessed. This restriction is in large part responsible for the faster speed of the buffers, relative to the Global Low-Skew buffers.

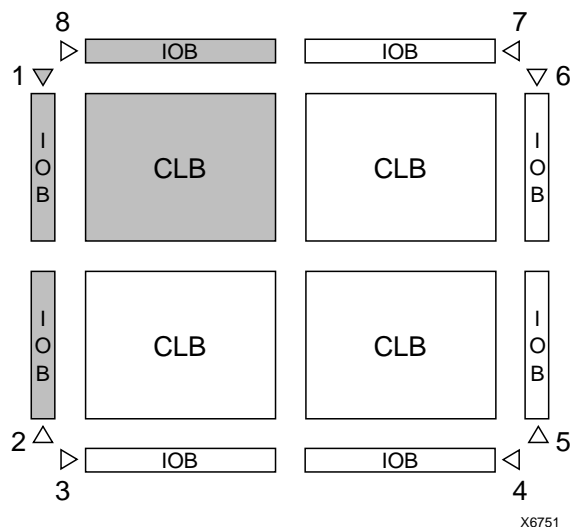


Figure 37: Left and Right BUFGEs Can Drive Any or All Clock Inputs in Same Quadrant or Edge (GCK1 is shown. GCK2, GCK5 and GCK6 are similar.)

The left-side Global Early buffers can each drive two of the four vertical lines accessing the IOBs on the entire left edge of the device. The right-side Global Early buffers can each drive two of the eight vertical lines accessing the IOBs on the entire right edge of the device. (See **Figure 37**.)

Each left and right Global Early buffer can also drive half of the IOBs along either the top or bottom edge of the device, using a dedicated line that can only be accessed through the Global Early buffers.

The top and bottom Global Early buffers can drive half of the IOBs along either the left or right edge of the device, as shown in **Figure 38**. They can only access the top and bottom IOBs via the CLB global lines.

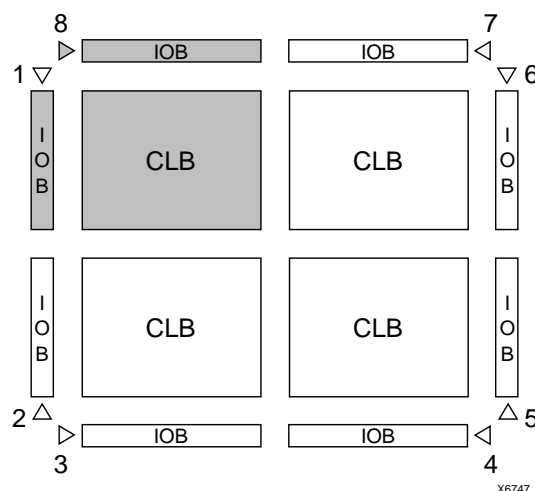


Figure 38: Top and Bottom BUFGEs Can Drive Any or All Clock Inputs in Same Quadrant (GCK8 is shown. GCK3, GCK4 and GCK7 are similar.)

Figure 41 on page 44 is a diagram of the XC4000 Series boundary scan logic. It includes three bits of Data Register per IOB, the IEEE 1149.1 Test Access Port controller, and the Instruction Register with decodes.

XC4000 Series devices can also be configured through the boundary scan logic. See “Readback” on page 55.

Data Registers

The primary data register is the boundary scan register. For each IOB pin in the FPGA, bonded or not, it includes three bits for In, Out and 3-State Control. Non-IOB pins have appropriate partial bit population for In or Out only. PROGRAM, CCLK and DONE are not included in the boundary scan register. Each EXTEST CAPTURE-DR state captures all In, Out, and 3-state pins.

The data register also includes the following non-pin bits: TDO.T, and TDO.O, which are always bits 0 and 1 of the

data register, respectively, and BSCANT.UPD, which is always the last bit of the data register. These three boundary scan bits are special-purpose Xilinx test signals.

The other standard data register is the single flip-flop BYPASS register. It synchronizes data being passed through the FPGA to the next downstream boundary scan device.

The FPGA provides two additional data registers that can be specified using the BSCAN macro. The FPGA provides two user pins (BSCAN.SEL1 and BSCAN.SEL2) which are the decodes of two user instructions. For these instructions, two corresponding pins (BSCAN.TDO1 and BSCAN.TDO2) allow user scan data to be shifted out on TDO. The data register clock (BSCAN.DRCK) is available for control of test logic which the user may wish to implement with CLBs. The NAND of TCK and RUN-TEST-IDLE is also provided (BSCAN.IDLE).

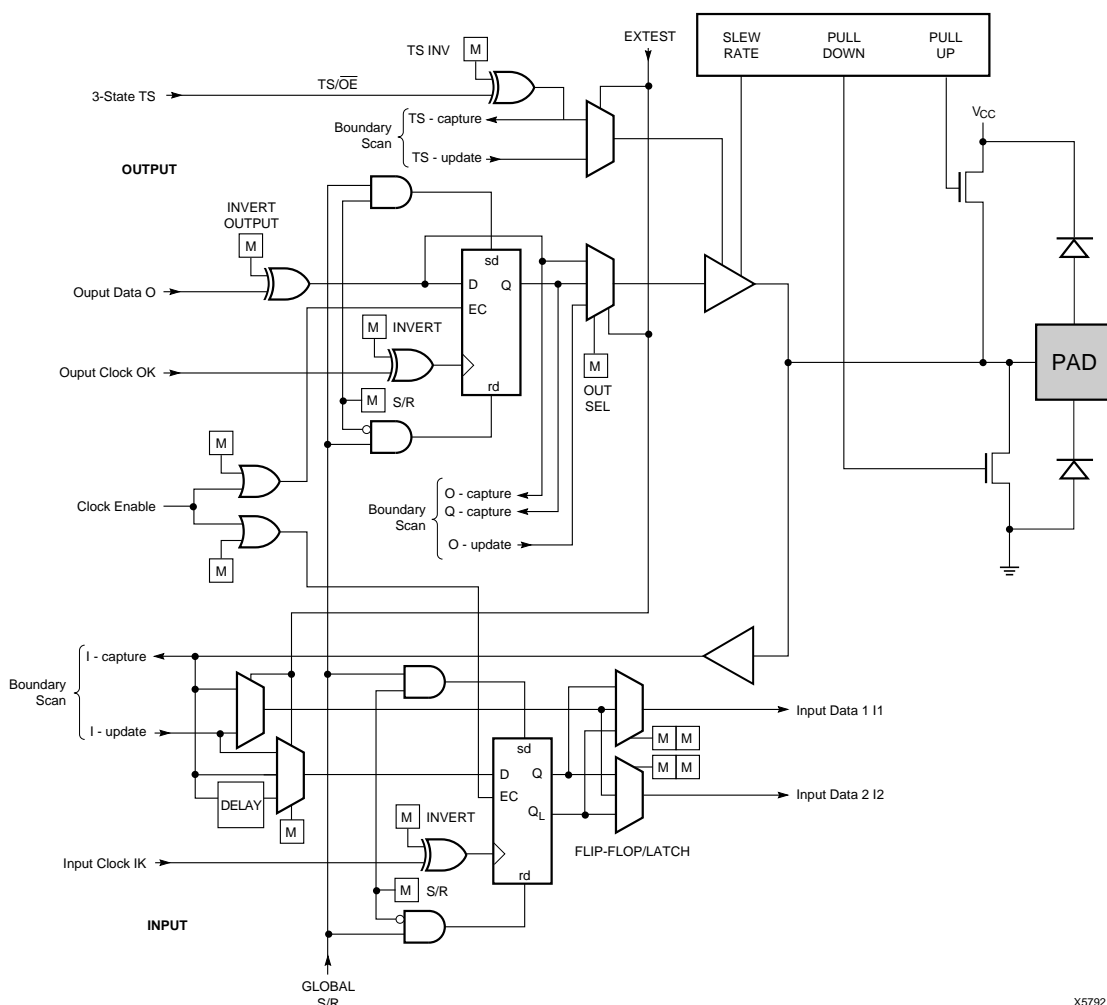


Figure 40: Block Diagram of XC4000E IOB with Boundary Scan (some details not shown). XC4000X Boundary Scan Logic is Identical.

Configuration Modes

XC4000E devices have six configuration modes. XC4000X devices have the same six modes, plus an additional configuration mode. These modes are selected by a 3-bit input code applied to the M2, M1, and M0 inputs. There are three self-loading Master modes, two Peripheral modes, and a Serial Slave mode, which is used primarily for daisy-chained devices. The coding for mode selection is shown in [Table 18](#).

Table 18: Configuration Modes

Mode	M2	M1	M0	CCLK	Data
Master Serial	0	0	0	output	Bit-Serial
Slave Serial	1	1	1	input	Bit-Serial
Master Parallel Up	1	0	0	output	Byte-Wide, increment from 00000
Master Parallel Down	1	1	0	output	Byte-Wide, decrement from 3FFFF
Peripheral Synchronous*	0	1	1	input	Byte-Wide
Peripheral Asynchronous	1	0	1	output	Byte-Wide
Reserved	0	1	0	—	—
Reserved	0	0	1	—	—

* Can be considered byte-wide Slave Parallel

A detailed description of each configuration mode, with timing information, is included later in this data sheet. During configuration, some of the I/O pins are used temporarily for the configuration process. All pins used during configuration are shown in [Table 22 on page 58](#).

Master Modes

The three Master modes use an internal oscillator to generate a Configuration Clock (CCLK) for driving potential slave devices. They also generate address and timing for external PROM(s) containing the configuration data.

Master Parallel (Up or Down) modes generate the CCLK signal and PROM addresses and receive byte parallel data. The data is internally serialized into the FPGA data-frame format. The up and down selection generates starting addresses at either zero or 3FFFF (3FFFFFF when 22 address lines are used), for compatibility with different microprocessor addressing conventions. The Master Serial mode generates CCLK and receives the configuration data in serial form from a Xilinx serial-configuration PROM.

CCLK speed is selectable as either 1 MHz (default) or 8 MHz. Configuration always starts at the default slow frequency, then can switch to the higher frequency during the first frame. Frequency tolerance is -50% to +25%.

Additional Address lines in XC4000 devices

The XC4000X devices have additional address lines (A18-A21) allowing the additional address space required to daisy-chain several large devices.

The extra address lines are programmable in XC4000EX devices. By default these address lines are not activated. In the default mode, the devices are compatible with existing XC4000 and XC4000E products. If desired, the extra address lines can be used by specifying the address lines option in bitgen as 22 (bitgen -g AddressLines:22). The lines (A18-A21) are driven when a master device detects, via the bitstream, that it should be using all 22 address lines. Because these pins will initially be pulled high by internal pull-ups, designers using Master Parallel Up mode should use external pull down resistors on pins A18-A21. If Master Parallel Down mode is used external resistors are not necessary.

All 22 address lines are always active in Master Parallel modes with XC4000XL devices. The additional address lines behave identically to the lower order address lines. If the Address Lines option in bitgen is set to 18, it will be ignored by the XC4000XL device.

The additional address lines (A18-A21) are not available in the PC84 package.

Peripheral Modes

The two Peripheral modes accept byte-wide data from a bus. A RDY/BUSY status is available as a handshake signal. In Asynchronous Peripheral mode, the internal oscillator generates a CCLK burst signal that serializes the byte-wide data. CCLK can also drive slave devices. In the synchronous mode, an externally supplied clock input to CCLK serializes the data.

Slave Serial Mode

In Slave Serial mode, the FPGA receives serial configuration data on the rising edge of CCLK and, after loading its configuration, passes additional data out, resynchronized on the next falling edge of CCLK.

Multiple slave devices with identical configurations can be wired with parallel DIN inputs. In this way, multiple devices can be configured simultaneously.

Serial Daisy Chain

Multiple devices with different configurations can be connected together in a "daisy chain," and a single combined bitstream used to configure the chain of slave devices.

To configure a daisy chain of devices, wire the CCLK pins of all devices in parallel, as shown in [Figure 51 on page 60](#). Connect the DOUT of each device to the DIN of the next. The lead or master FPGA and following slaves each passes resynchronized configuration data coming from a single source. The header data, including the length count,

Low. During this time delay, or as long as the $\overline{\text{PROGRAM}}$ input is asserted, the configuration logic is held in a Configuration Memory Clear state. The configuration-memory frames are consecutively initialized, using the internal oscillator.

At the end of each complete pass through the frame addressing, the power-on time-out delay circuitry and the level of the $\overline{\text{PROGRAM}}$ pin are tested. If neither is asserted, the logic initiates one additional clearing of the configuration frames and then tests the $\overline{\text{INIT}}$ input.

Initialization

During initialization and configuration, user pins $\overline{\text{HDC}}$, $\overline{\text{LDC}}$, $\overline{\text{INIT}}$ and $\overline{\text{DONE}}$ provide status outputs for the system interface. The outputs $\overline{\text{LDC}}$, $\overline{\text{INIT}}$ and $\overline{\text{DONE}}$ are held Low and $\overline{\text{HDC}}$ is held High starting at the initial application of power.

The open drain $\overline{\text{INIT}}$ pin is released after the final initialization pass through the frame addresses. There is a deliberate delay of 50 to 250 μs (up to 10% longer for low-voltage devices) before a Master-mode device recognizes an inactive $\overline{\text{INIT}}$. Two internal clocks after the $\overline{\text{INIT}}$ pin is recognized as High, the FPGA samples the three mode lines to determine the configuration mode. The appropriate interface lines become active and the configuration preamble and data can be loaded. Configuration

The 0010 preamble code indicates that the following 24 bits represent the length count. The length count is the total number of configuration clocks needed to load the complete configuration data. (Four additional configuration clocks are required to complete the configuration process, as discussed below.) After the preamble and the length count have been passed through to all devices in the daisy chain, $\overline{\text{DOUT}}$ is held High to prevent frame start bits from reaching any daisy-chained devices.

A specific configuration bit, early in the first frame of a master device, controls the configuration-clock rate and can increase it by a factor of eight. Therefore, if a fast configuration clock is selected by the bitstream, the slower clock rate is used until this configuration bit is detected.

Each frame has a start field followed by the frame-configuration data bits and a frame error field. If a frame data error is detected, the FPGA halts loading, and signals the error by pulling the open-drain $\overline{\text{INIT}}$ pin Low. After all configuration frames have been loaded into an FPGA, $\overline{\text{DOUT}}$ again follows the input data so that the remaining data is passed on to the next device.

Delaying Configuration After Power-Up

There are two methods of delaying configuration after power-up: put a logic Low on the $\overline{\text{PROGRAM}}$ input, or pull the bidirectional $\overline{\text{INIT}}$ pin Low, using an open-collector (open-drain) driver. (See [Figure 46 on page 50](#).)

A Low on the $\overline{\text{PROGRAM}}$ input is the more radical approach, and is recommended when the power-supply

rise time is excessive or poorly defined. As long as $\overline{\text{PROGRAM}}$ is Low, the FPGA keeps clearing its configuration memory. When $\overline{\text{PROGRAM}}$ goes High, the configuration memory is cleared one more time, followed by the beginning of configuration, provided the $\overline{\text{INIT}}$ input is not externally held Low. Note that a Low on the $\overline{\text{PROGRAM}}$ input automatically forces a Low on the $\overline{\text{INIT}}$ output. The XC4000 Series $\overline{\text{PROGRAM}}$ pin has a permanent weak pull-up.

Using an open-collector or open-drain driver to hold $\overline{\text{INIT}}$ Low before the beginning of configuration causes the FPGA to wait after completing the configuration memory clear operation. When $\overline{\text{INIT}}$ is no longer held Low externally, the device determines its configuration mode by capturing its mode pins, and is ready to start the configuration process. A master device waits up to an additional 250 μs to make sure that any slaves in the optional daisy chain have seen that $\overline{\text{INIT}}$ is High.

Start-Up

Start-up is the transition from the configuration process to the intended user operation. This transition involves a change from one clock source to another, and a change from interfacing parallel or serial configuration data where most outputs are 3-stated, to normal operation with I/O pins active in the user-system. Start-up must make sure that the user-logic 'wakes up' gracefully, that the outputs become active without causing contention with the configuration signals, and that the internal flip-flops are released from the global Reset or Set at the right time.

[Figure 47](#) describes start-up timing for the three Xilinx families in detail. The configuration modes can use any of the four timing sequences.

To access the internal start-up signals, place the STARTUP library symbol.

Start-up Timing

Different FPGA families have different start-up sequences.

The XC2000 family goes through a fixed sequence. $\overline{\text{DONE}}$ goes High and the internal global Reset is de-activated one CCLK period after the I/O become active.

The XC3000A family offers some flexibility. $\overline{\text{DONE}}$ can be programmed to go High one CCLK period before or after the I/O become active. Independent of $\overline{\text{DONE}}$, the internal global Reset is de-activated one CCLK period before or after the I/O become active.

The XC4000 Series offers additional flexibility. The three events — $\overline{\text{DONE}}$ going High, the internal Set/Reset being de-activated, and the user I/O going active — can all occur in any arbitrary sequence. Each of them can occur one CCLK period before or after, or simultaneous with, any of the others. This relative timing is selected by means of software options in the bitstream generation software.

The default option, and the most practical one, is for DONE to go High first, disconnecting the configuration data source and avoiding any contention when the I/Os become active one clock later. Reset/Set is then released another clock period later to make sure that user-operation starts from stable internal conditions. This is the most common sequence, shown with heavy lines in [Figure 47](#), but the designer can modify it to meet particular requirements.

Normally, the start-up sequence is controlled by the internal device oscillator output (CCLK), which is asynchronous to the system clock.

XC4000 Series offers another start-up clocking option, UCLK_NOSYNC. The three events described above need not be triggered by CCLK. They can, as a configuration option, be triggered by a user clock. This means that the device can wake up in synchronism with the user system.

When the UCLK_SYNC option is enabled, the user can externally hold the open-drain DONE output Low, and thus stall all further progress in the start-up sequence until DONE is released and has gone High. This option can be used to force synchronization of several FPGAs to a common user clock, or to guarantee that all devices are successfully configured before any I/Os go active.

If either of these two options is selected, and no user clock is specified in the design or attached to the device, the chip could reach a point where the configuration of the device is complete and the Done pin is asserted, but the outputs do not become active. The solution is either to recreate the bit-stream specifying the start-up clock as CCLK, or to supply the appropriate user clock.

Start-up Sequence

The Start-up sequence begins when the configuration memory is full, and the total number of configuration clocks

received since $\overline{\text{INIT}}$ went High equals the loaded value of the length count.

The next rising clock edge sets a flip-flop Q0, shown in [Figure 48](#). Q0 is the leading bit of a 5-bit shift register. The outputs of this register can be programmed to control three events.

- The release of the open-drain DONE output
- The change of configuration-related pins to the user function, activating all IOBs.
- The termination of the global Set/Reset initialization of all CLB and IOB storage elements.

The DONE pin can also be wire-ANDed with DONE pins of other FPGAs or with other external signals, and can then be used as input to bit Q3 of the start-up register. This is called "Start-up Timing Synchronous to Done In" and is selected by either CCLK_SYNC or UCLK_SYNC.

When DONE is not used as an input, the operation is called "Start-up Timing Not Synchronous to DONE In," and is selected by either CCLK_NOSYNC or UCLK_NOSYNC.

As a configuration option, the start-up control register beyond Q0 can be clocked either by subsequent CCLK pulses or from an on-chip user net called STARTUP.CLK. These signals can be accessed by placing the STARTUP library symbol.

Start-up from CCLK

If CCLK is used to drive the start-up, Q0 through Q3 provide the timing. Heavy lines in [Figure 47](#) show the default timing, which is compatible with XC2000 and XC3000 devices using early DONE and late Reset. The thin lines indicate all other possible timing options.

Configuration Timing

The seven configuration modes are discussed in detail in this section. Timing specifications are included.

Slave Serial Mode

In Slave Serial mode, an external signal drives the CCLK input of the FPGA. The serial configuration bitstream must be available at the DIN input of the lead FPGA a short setup time before each rising CCLK edge.

The lead FPGA then presents the preamble data—and all data that overflows the lead device—on its DOUT pin.

There is an internal delay of 0.5 CCLK periods, which means that DOUT changes on the falling CCLK edge, and the next FPGA in the daisy chain accepts data on the subsequent rising CCLK edge.

Figure 51 shows a full master/slave system. An XC4000 Series device in Slave Serial mode should be connected as shown in the third device from the left.

Slave Serial mode is selected by a <111> on the mode pins (M2, M1, M0). Slave Serial is the default mode if the mode pins are left unconnected, as they have weak pull-up resistors during configuration.

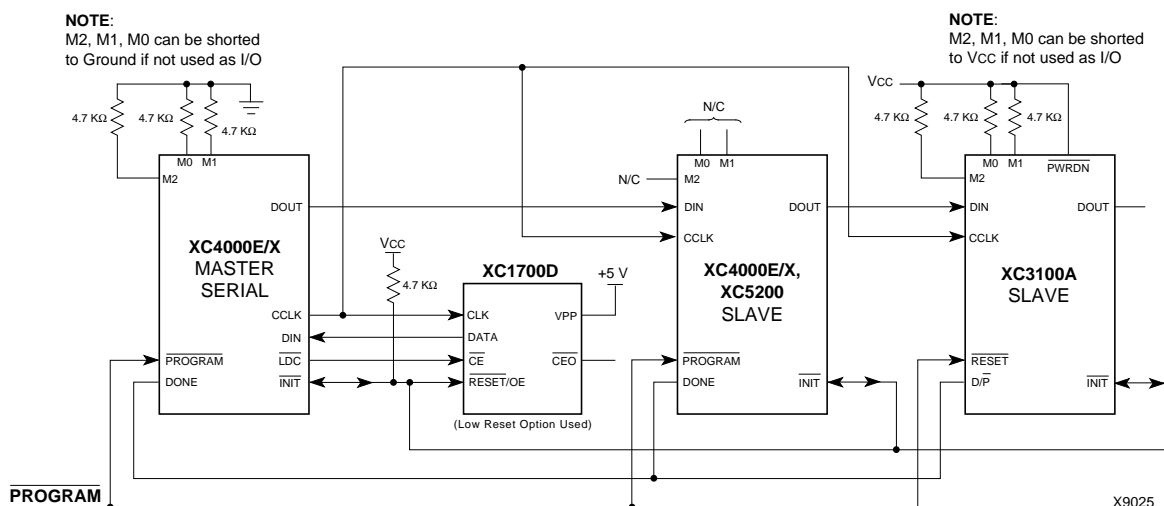
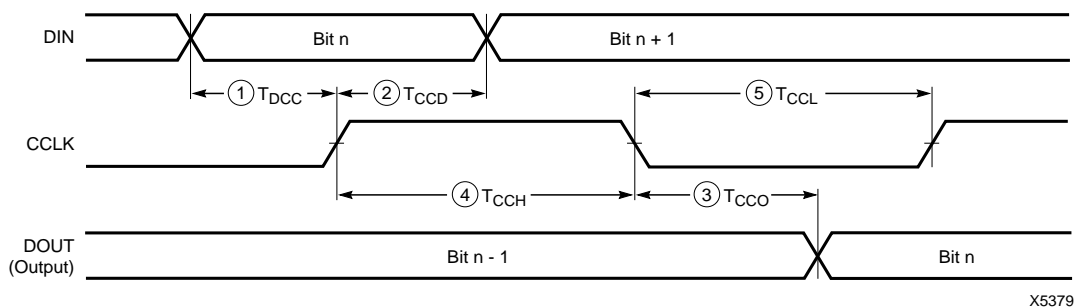


Figure 51: Master/Slave Serial Mode Circuit Diagram



	Description	Symbol	Min	Max	Units
CCLK	DIN setup	1 T_{DCC}	20		ns
	DIN hold	2 T_{CCD}	0		ns
	DIN to DOUT	3 T_{CCO}		30	ns
	High time	4 T_{CCH}	45		ns
	Low time	5 T_{CCL}	45		ns
	Frequency	F_{CC}		10	MHz

Note: Configuration must be delayed until the INIT pins of all daisy-chained FPGAs are High.

Figure 52: Slave Serial Mode Programming Switching Characteristics

Master Serial Mode

In Master Serial mode, the CCLK output of the lead FPGA drives a Xilinx Serial PROM that feeds the FPGA DIN input. Each rising edge of the CCLK output increments the Serial PROM internal address counter. The next data bit is put on the SPROM data output, connected to the FPGA DIN pin. The lead FPGA accepts this data on the subsequent rising CCLK edge.

The lead FPGA then presents the preamble data—and all data that overflows the lead device—on its DOUT pin. There is an internal pipeline delay of 1.5 CCLK periods, which means that DOUT changes on the falling CCLK edge, and the next FPGA in the daisy chain accepts data on the subsequent rising CCLK edge.

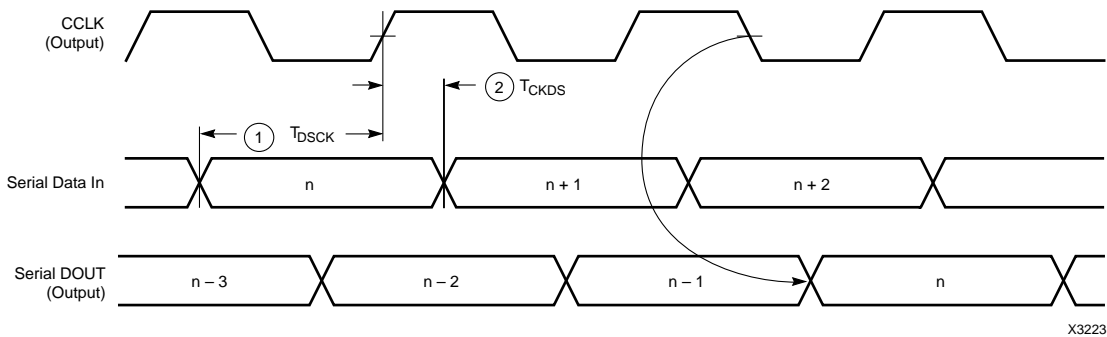
In the bitstream generation software, the user can specify Fast ConfigRate, which, starting several bits into the first frame, increases the CCLK frequency by a factor of eight.

For actual timing values please refer to “**Configuration Switching Characteristics**” on page 68. Be sure that the serial PROM and slaves are fast enough to support this data rate. XC2000, XC3000/A, and XC3100A devices do not support the Fast ConfigRate option.

The SPROM CE input can be driven from either $\overline{\text{LDC}}$ or DONE. Using $\overline{\text{LDC}}$ avoids potential contention on the DIN pin, if this pin is configured as user-I/O, but $\overline{\text{LDC}}$ is then restricted to be a permanently High user output after configuration. Using DONE can also avoid contention on DIN, provided the early DONE option is invoked.

Figure 51 on page 60 shows a full master/slave system. The leftmost device is in Master Serial mode.

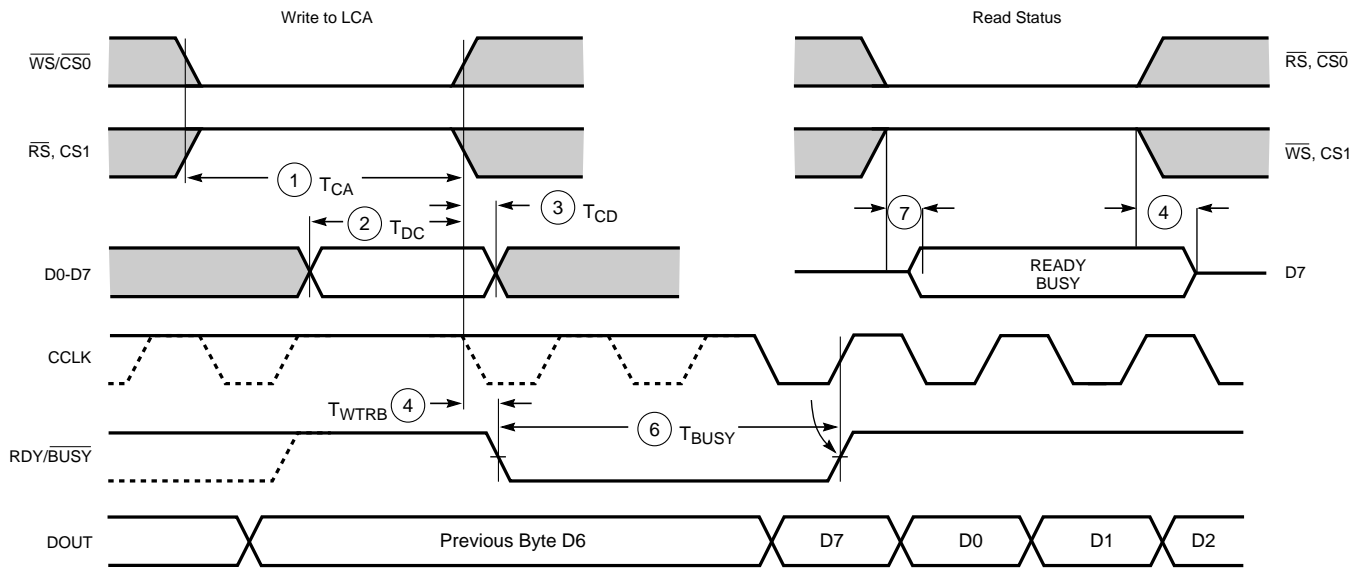
Master Serial mode is selected by a <000> on the mode pins (M2, M1, M0).



	Description	Symbol	Min	Max	Units
CCLK	DIN setup	1 T_{DSCK}	20		ns
	DIN hold	2 T_{CKDS}	0		ns

Notes: 1. At power-up, Vcc must rise from 2.0 V to Vcc min in less than 25 ms, otherwise delay configuration by pulling PROGRAM Low until Vcc is valid.
2. Master Serial mode timing is based on testing in slave mode.

Figure 53: Master Serial Mode Programming Switching Characteristics



X6097

	Description	Symbol	Min	Max	Units
Write	Effective Write time (CS0, WS=Low; RS, CS1=High)	1 T_{CA}	100		ns
	DIN setup time	2 T_{DC}	60		ns
	DIN hold time	3 T_{CD}	0		ns
RDY	RDY/BUSY delay after end of Write or Read	4 T_{WTRB}		60	ns
	RDY/BUSY active after beginning of Read	7		60	ns
	RDY/BUSY Low output (Note 4)	6 T_{BUSY}	2	9	CCLK periods

- Notes:
1. Configuration must be delayed until the \overline{INIT} pins of all daisy-chained FPGAs are High.
 2. The time from the end of \overline{WS} to CCLK cycle for the new byte of data depends on the completion of previous byte processing and the phase of the internal timing generator for CCLK.
 3. CCLK and DOUT timing is tested in slave mode.
 4. T_{BUSY} indicates that the double-buffered parallel-to-serial converter is not yet ready to receive new data. The shortest T_{BUSY} occurs when a byte is loaded into an empty parallel-to-serial converter. The longest T_{BUSY} occurs when a new word is loaded into the input register before the second-level buffer has started shifting out data.

This timing diagram shows very relaxed requirements. Data need not be held beyond the rising edge of \overline{WS} . RDY/BUSY will go active within 60 ns after the end of \overline{WS} . A new write may be asserted immediately after RDY/BUSY goes Low, but write may not be terminated until RDY/BUSY has been High for one CCLK period.

Figure 59: Asynchronous Peripheral Mode Programming Switching Characteristics