

Welcome to [E-XFL.COM](https://www.e-xfl.com)

Understanding Embedded - FPGAs (Field Programmable Gate Array)

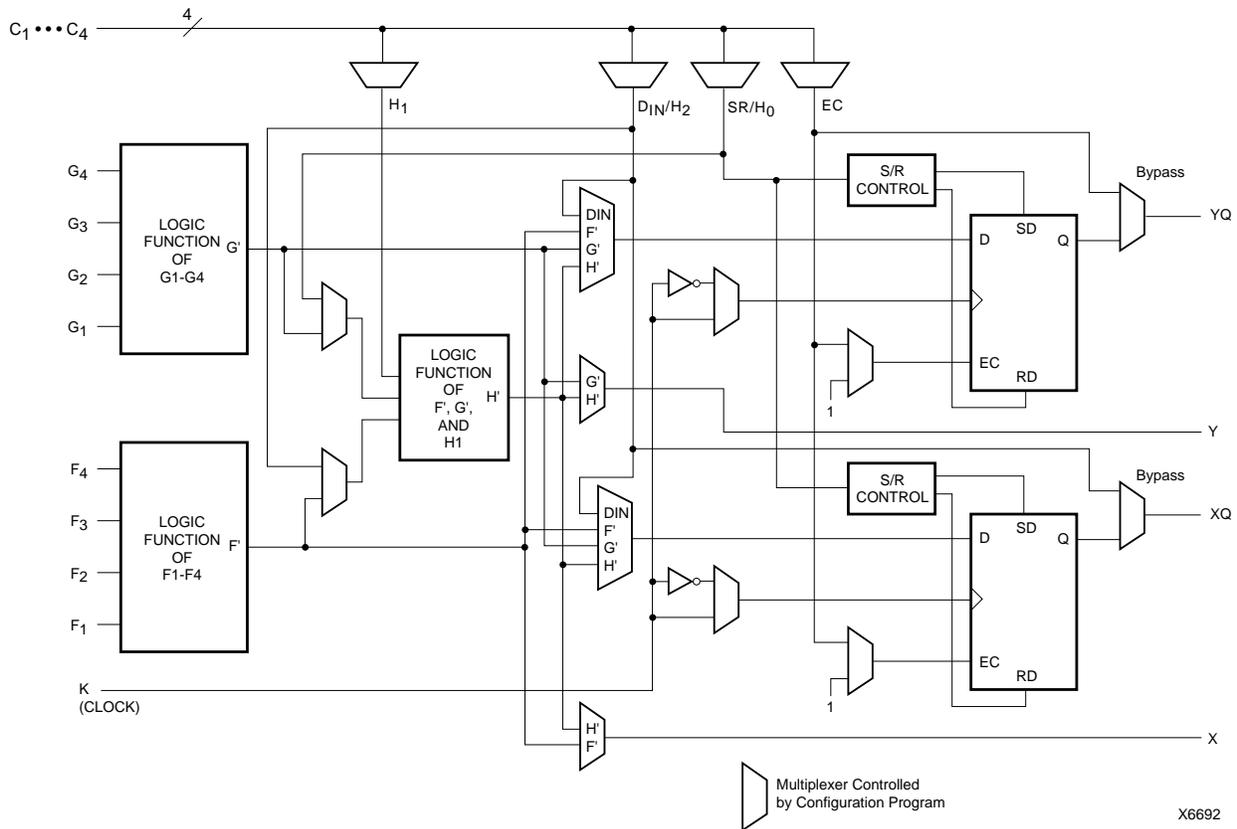
Embedded - FPGAs, or Field Programmable Gate Arrays, are advanced integrated circuits that offer unparalleled flexibility and performance for digital systems. Unlike traditional fixed-function logic devices, FPGAs can be programmed and reprogrammed to execute a wide array of logical operations, enabling 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 | 324 |
| Number of Logic Elements/Cells | 770 |
| Total RAM Bits | 10368 |
| Number of I/O | 129 |
| Number of Gates | 8000 |
| Voltage - Supply | 4.75V ~ 5.25V |
| Mounting Type | Surface Mount |
| Operating Temperature | 0°C ~ 85°C (Tj) |
| Package / Case | 160-BQFP |
| Supplier Device Package | 160-PQFP (28x28) |
| Purchase URL | https://www.e-xfl.com/product-detail/xilinx/xc4008e-3pq160c |



X6692

Figure 1: Simplified Block Diagram of XC4000 Series CLB (RAM and Carry Logic functions not shown)

Flip-Flops

The CLB can pass the combinational output(s) to the interconnect network, but can also store the combinational results or other incoming data in one or two flip-flops, and connect their outputs to the interconnect network as well.

The two edge-triggered D-type flip-flops have common clock (K) and clock enable (EC) inputs. Either or both clock inputs can also be permanently enabled. Storage element functionality is described in [Table 2](#).

Latches (XC4000X only)

The CLB storage elements can also be configured as latches. The two latches have common clock (K) and clock enable (EC) inputs. Storage element functionality is described in [Table 2](#).

Clock Input

Each flip-flop can be triggered on either the rising or falling clock edge. The clock pin is shared by both storage elements. However, the clock is individually invertible for each storage element. Any inverter placed on the clock input is automatically absorbed into the CLB.

Clock Enable

The clock enable signal (EC) is active High. The EC pin is shared by both storage elements. If left unconnected for either, the clock enable for that storage element defaults to the active state. EC is not invertible within the CLB.

Table 2: CLB Storage Element Functionality (active rising edge is shown)

| Mode | K | EC | SR | D | Q |
|-----------------|---|----|----|---|----|
| Power-Up or GSR | X | X | X | X | SR |
| Flip-Flop | X | X | 1 | X | SR |
| | | 1* | 0* | D | D |
| Latch | 0 | X | 0* | X | Q |
| | 1 | 1* | 0* | X | Q |
| Both | 0 | 1* | 0* | D | D |
| | X | 0 | 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)

tions of the CLB, with the exception of the redefinition of the control signals. In 16x2 and 16x1 modes, the H' function generator can be used to implement Boolean functions of F', G', and D1, and the D flip-flops can latch the F', G', H', or D0 signals.

Single-Port Edge-Triggered Mode

Edge-triggered (synchronous) RAM simplifies timing requirements. XC4000 Series edge-triggered RAM operates like writing to a data register. Data and address are presented. The register is enabled for writing by a logic High on the write enable input, WE. Then a rising or falling clock edge loads the data into the register, as shown in Figure 3.

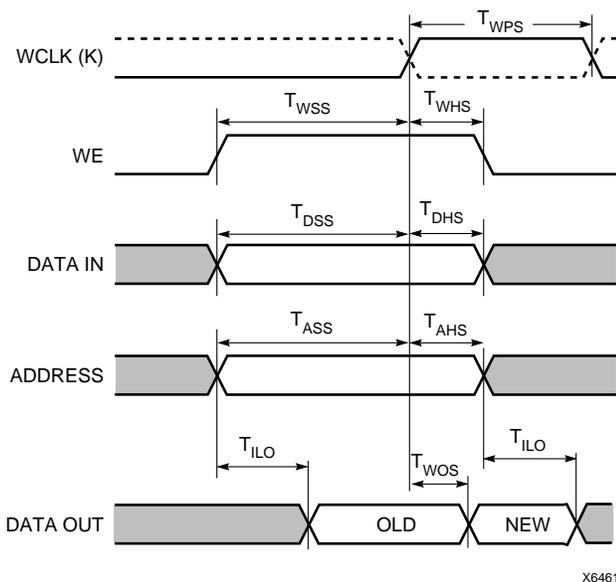


Figure 3: Edge-Triggered RAM Write Timing

Complex timing relationships between address, data, and write enable signals are not required, and the external write enable pulse becomes a simple clock enable. The active edge of WCLK latches the address, input data, and WE sig-

nals. An internal write pulse is generated that performs the write. See Figure 4 and Figure 5 for block diagrams of a CLB configured as 16x2 and 32x1 edge-triggered, single-port RAM.

The relationships between CLB pins and RAM inputs and outputs for single-port, edge-triggered mode are shown in Table 5.

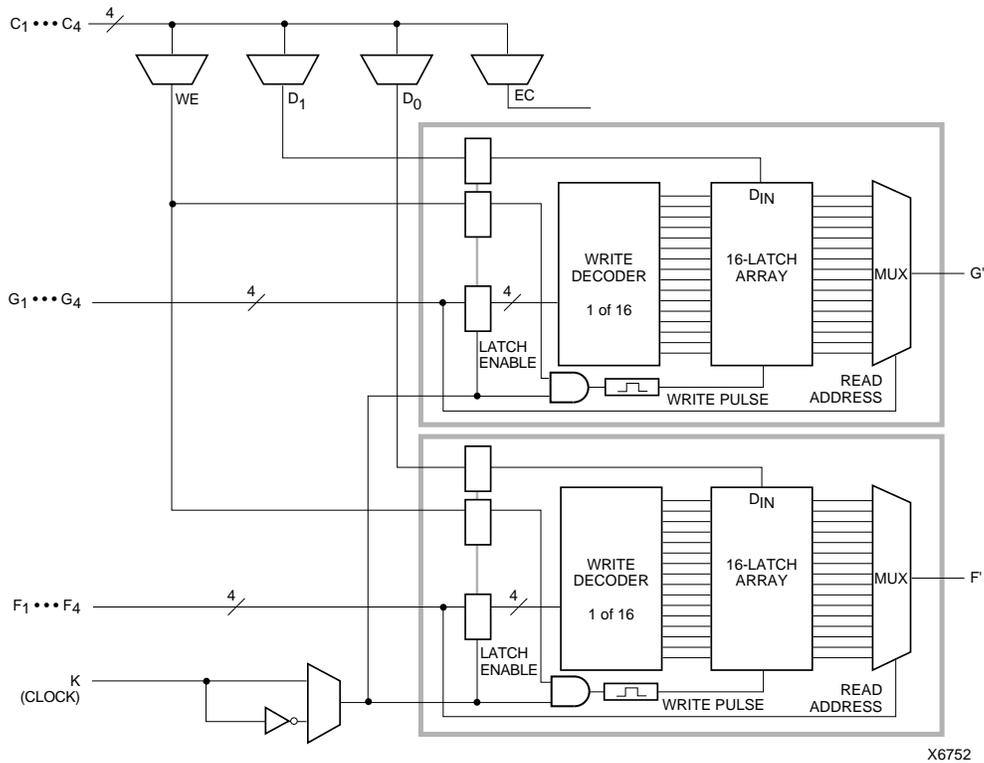
The Write Clock input (WCLK) can be configured as active on either the rising edge (default) or the falling edge. It uses the same CLB pin (K) used to clock the CLB flip-flops, but it can be independently inverted. Consequently, the RAM output can optionally be registered within the same CLB either by the same clock edge as the RAM, or by the opposite edge of this clock. The sense of WCLK applies to both function generators in the CLB when both are configured as RAM.

The WE pin is active-High and is not invertible within the CLB.

Note: The pulse following the active edge of WCLK (T_{WPS} in Figure 3) must be less than one millisecond wide. For most applications, this requirement is not overly restrictive; however, it must not be forgotten. Stopping WCLK at this point in the write cycle could result in excessive current and even damage to the larger devices if many CLBs are configured as edge-triggered RAM.

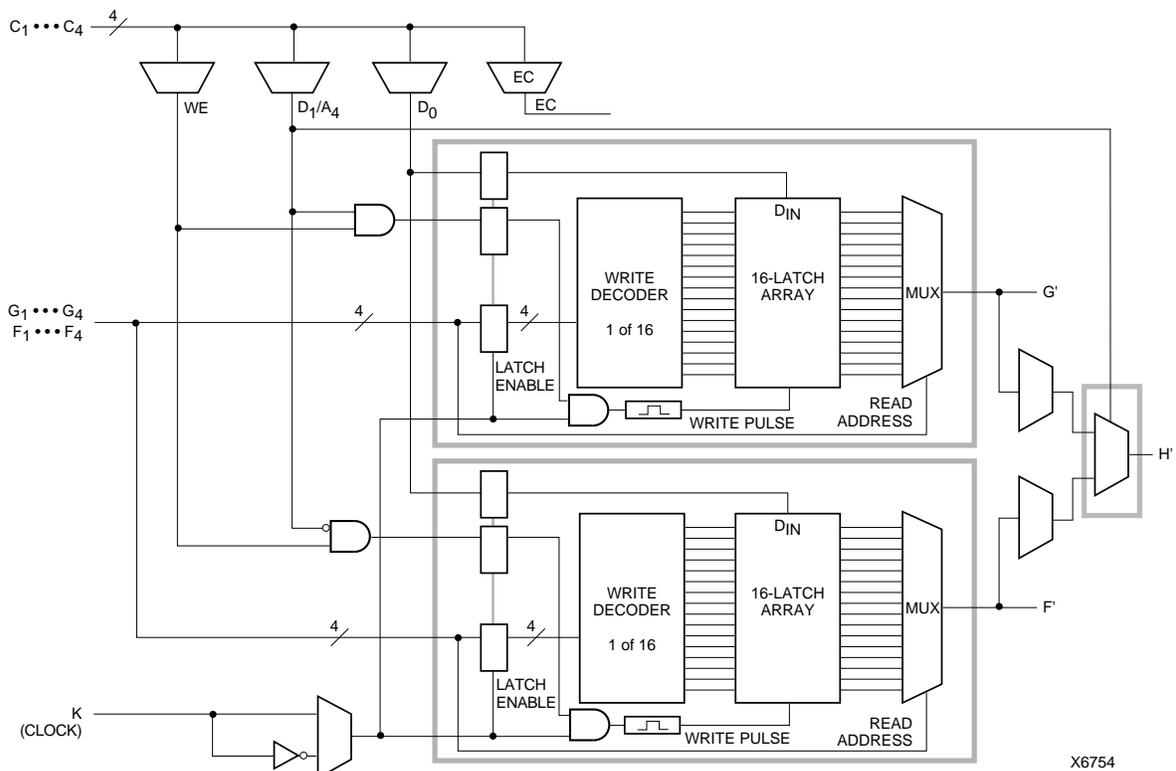
Table 5: Single-Port Edge-Triggered RAM Signals

| RAM Signal | CLB Pin | Function |
|----------------|----------------------------------|----------------------------|
| D | D0 or D1 (16x2, 16x1), D0 (32x1) | Data In |
| A[3:0] | F1-F4 or G1-G4 | Address |
| A[4] | D1 (32x1) | Address |
| WE | WE | Write Enable |
| WCLK | K | Clock |
| SPO (Data Out) | F' or G' | Single Port Out (Data Out) |



X6752

Figure 4: 16x2 (or 16x1) Edge-Triggered Single-Port RAM



X6754

Figure 5: 32x1 Edge-Triggered Single-Port RAM (F and G addresses are identical)

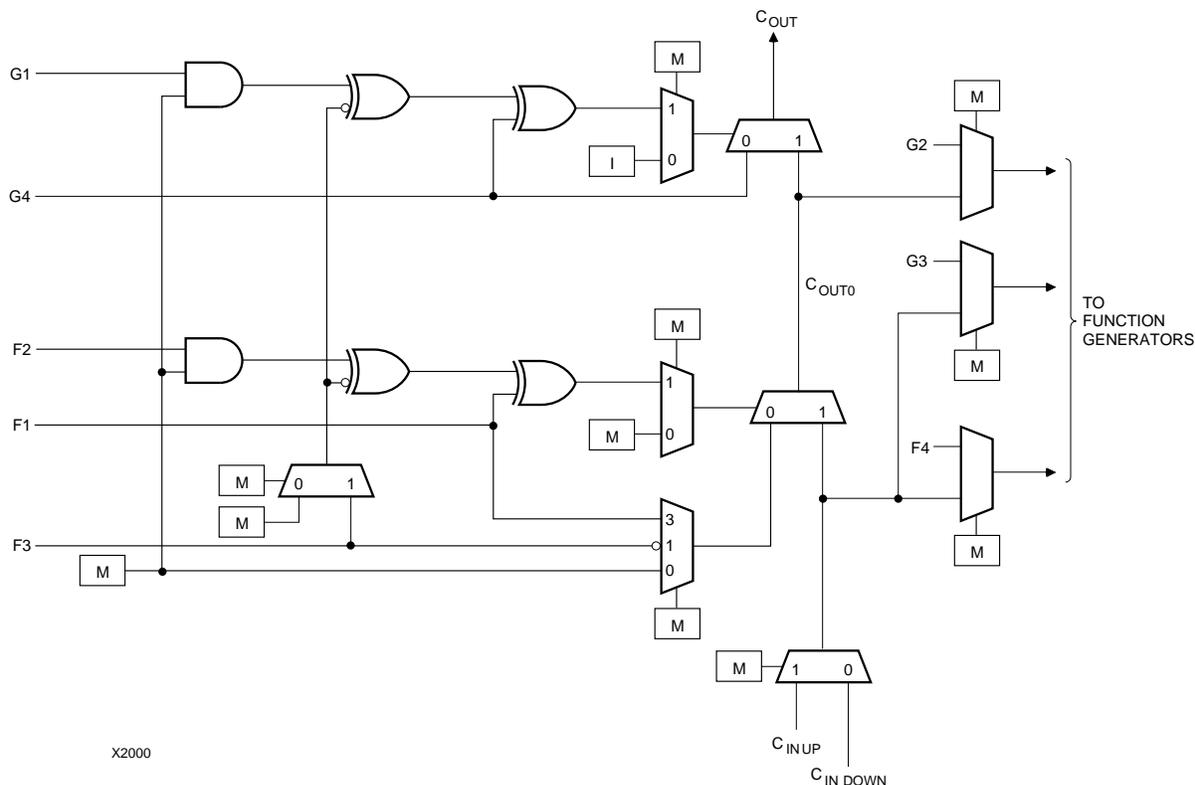


Figure 14: Detail of XC4000E Dedicated Carry Logic

Input/Output Blocks (IOBs)

User-configurable input/output blocks (IOBs) provide the interface between external package pins and the internal logic. Each IOB controls one package pin and can be configured for input, output, or bidirectional signals.

Figure 15 shows a simplified block diagram of the XC4000E IOB. A more complete diagram which includes the boundary scan logic of the XC4000E IOB can be found in Figure 40 on page 43, in the “Boundary Scan” section.

The XC4000X IOB contains some special features not included in the XC4000E IOB. These features are highlighted in a simplified block diagram found in Figure 16, and discussed throughout this section. When XC4000X special features are discussed, they are clearly identified in the text. Any feature not so identified is present in both XC4000E and XC4000X devices.

IOB Input Signals

Two paths, labeled I1 and I2 in Figure 15 and Figure 16, bring input signals into the array. Inputs also connect to an input register that can be programmed as either an edge-triggered flip-flop or a level-sensitive latch.

The choice is made by placing the appropriate library symbol. For example, IFD is the basic input flip-flop (rising edge triggered), and ILD is the basic input latch (transparent-High). Variations with inverted clocks are available, and some combinations of latches and flip-flops can be implemented in a single IOB, as described in the *XACT Libraries Guide*.

The XC4000E inputs can be globally configured for either TTL (1.2V) or 5.0 volt CMOS thresholds, using an option in the bitstream generation software. There is a slight input hysteresis of about 300mV. The XC4000E output levels are also configurable; the two global adjustments of input threshold and output level are independent.

Inputs on the XC4000XL are TTL compatible and 3.3V CMOS compatible. Outputs on the XC4000XL are pulled to the 3.3V positive supply.

The inputs of XC4000 Series 5-Volt devices can be driven by the outputs of any 3.3-Volt device, if the 5-Volt inputs are in TTL mode.

Supported sources for XC4000 Series device inputs are shown in Table 8.

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 |

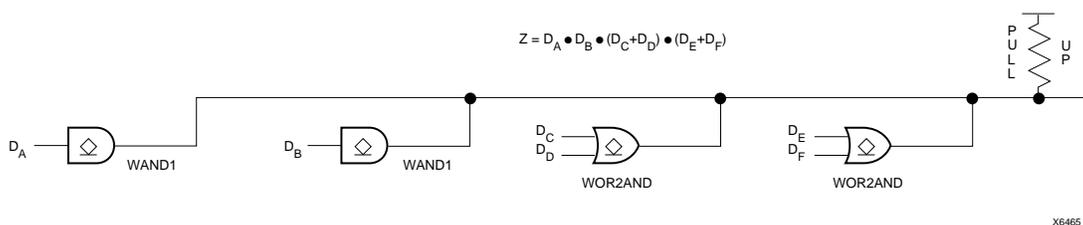


Figure 21: Open-Drain Buffers Implement a Wired-AND Function

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.

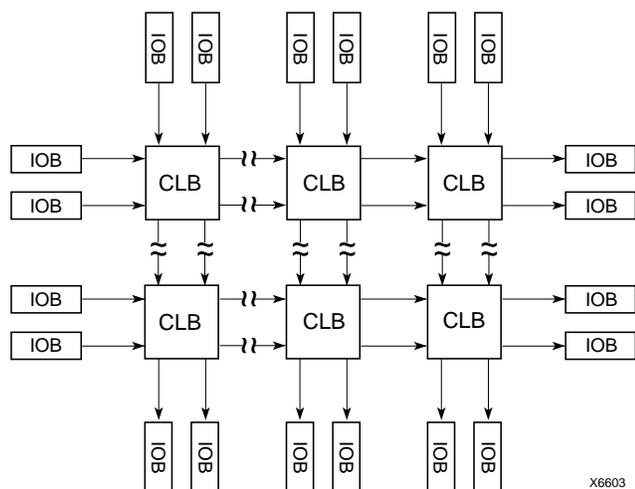


Figure 30: XC4000X Direct Interconnect

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](#).

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 Interconnect |
|---|---------|-------|---------|-------------|-------------|--------------------|
| | BUFGP | BUFGS | BUFGLS | L & R BUFGS | T & B BUFGS | |
| 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

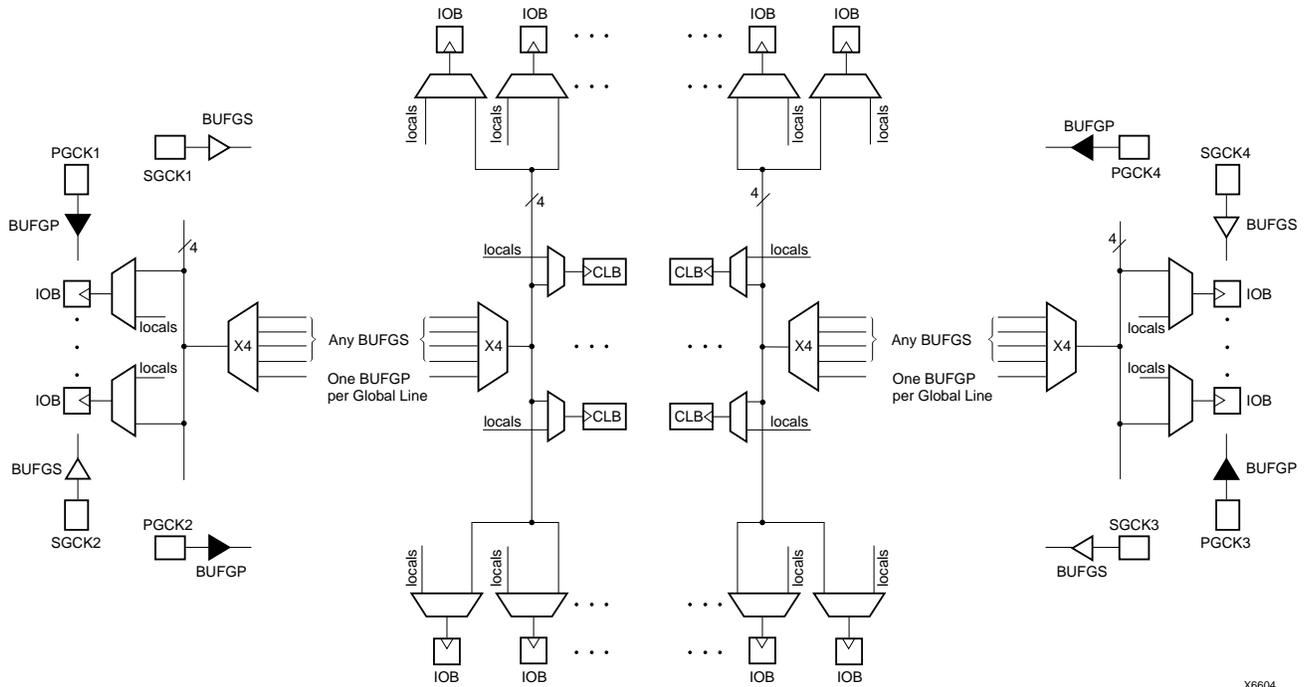


Figure 34: XC4000E Global Net Distribution

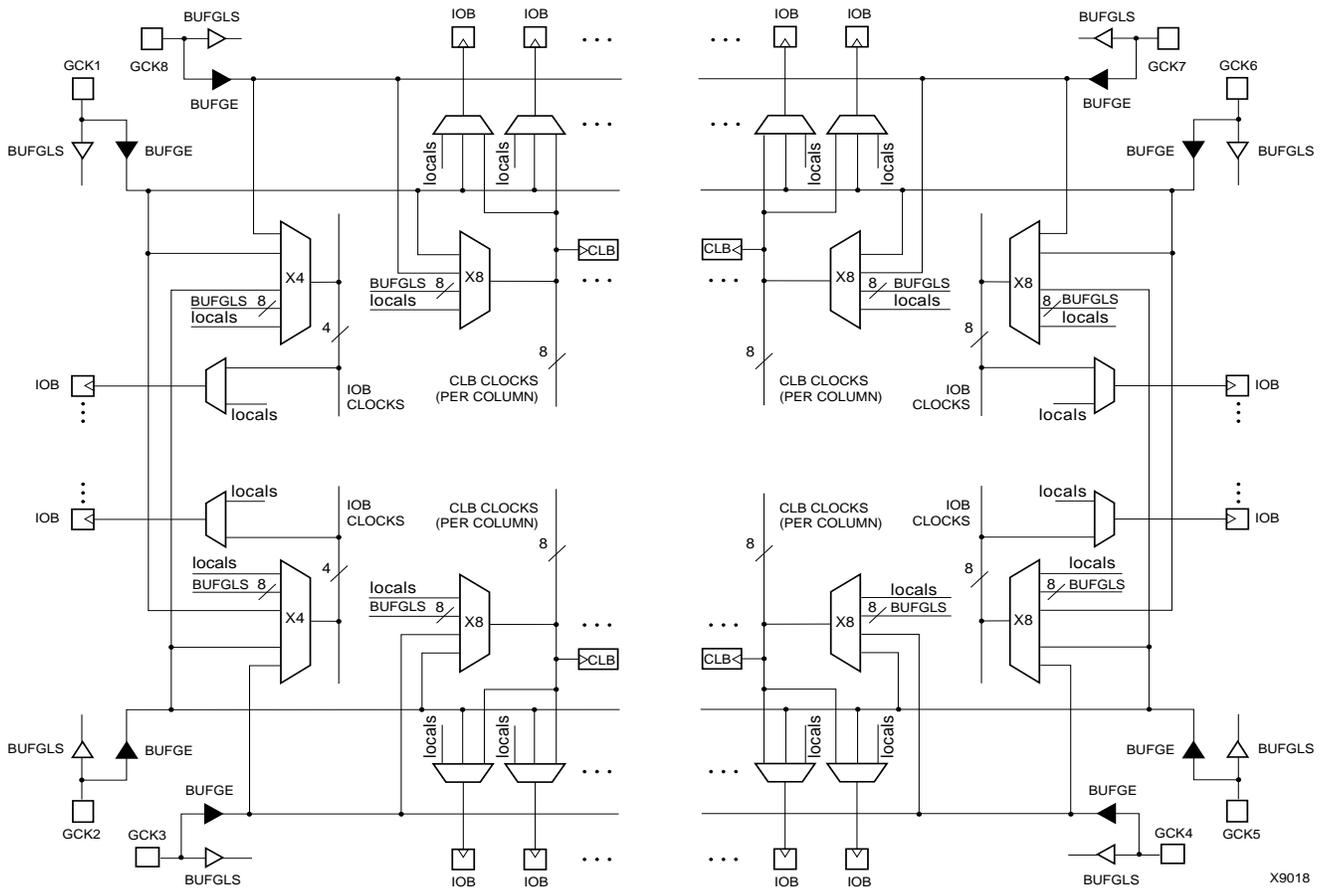


Figure 35: XC4000X Global Net Distribution

Global Nets and Buffers (XC4000X only)

Eight vertical longlines in each CLB column are driven by special global buffers. These longlines are in addition to the vertical longlines used for standard interconnect. The global lines are broken in the center of the array, to allow faster distribution and to minimize skew across the whole array. Each half-column global line has its own buffered multiplexer, as shown in [Figure 35](#). The top and bottom global lines cannot be connected across the center of the device, as this connection might introduce unacceptable skew. The top and bottom halves of the global lines must be separately driven — although they can be driven by the same global buffer.

The eight global lines in each CLB column can be driven by either of two types of global buffers. They can also be driven by internal logic, because they can be accessed by single, double, and quad lines at the top, bottom, half, and quarter points. Consequently, the number of different clocks that can be used simultaneously in an XC4000X device is very large.

There are four global lines feeding the IOBs at the left edge of the device. IOBs along the right edge have eight global lines. There is a single global line along the top and bottom edges with access to the IOBs. All IOB global lines are broken at the center. They cannot be connected across the center of the device, as this connection might introduce unacceptable skew.

IOB global lines can be driven from two types of global buffers, or from local interconnect. Alternatively, top and bottom IOBs can be clocked from the global lines in the adjacent CLB column.

Two different types of clock buffers are available in the XC4000X:

- Global Low-Skew Buffers (BUFGLS)
- Global Early Buffers (BUFGE)

Global Low-Skew Buffers are the standard clock buffers. They should be used for most internal clocking, whenever a large portion of the device must be driven.

Global Early Buffers are designed to provide a faster clock access, but CLB access is limited to one-fourth of the device. They also facilitate a faster I/O interface.

[Figure 35](#) is a conceptual diagram of the global net structure in the XC4000X.

Global Early buffers and Global Low-Skew buffers share a single pad. Therefore, the same IPAD symbol can drive one buffer of each type, in parallel. This configuration is particularly useful when using the Fast Capture latches, as described in [“IOB Input Signals” on page 20](#). Paired Global

Early and Global Low-Skew buffers share a common input; they cannot be driven by two different signals.

Choosing an XC4000X Clock Buffer

The clocking structure of the XC4000X provides a large variety of features. However, it can be simple to use, without understanding all the details. The software automatically handles clocks, along with all other routing, when the appropriate clock buffer is placed in the design. In fact, if a buffer symbol called BUFG is placed, rather than a specific type of buffer, the software even chooses the buffer most appropriate for the design. The detailed information in this section is provided for those users who want a finer level of control over their designs.

If fine control is desired, use the following summary and [Table 15 on page 35](#) to choose an appropriate clock buffer.

- The simplest thing to do is to use a Global Low-Skew buffer.
- If a faster clock path is needed, try a BUFG. The software will first try to use a Global Low-Skew Buffer. If timing requirements are not met, a faster buffer will automatically be used.
- If a single quadrant of the chip is sufficient for the clocked logic, and the timing requires a faster clock than the Global Low-Skew buffer, use a Global Early buffer.

Global Low-Skew Buffers

Each corner of the XC4000X device has two Global Low-Skew buffers. Any of the eight Global Low-Skew buffers can drive any of the eight vertical Global lines in a column of CLBs. In addition, any of the buffers can drive any of the four vertical lines accessing the IOBs on the left edge of the device, and any of the eight vertical lines accessing the IOBs on the right edge of the device. (See [Figure 36 on page 38](#).)

IOBs at the top and bottom edges of the device are accessed through the vertical Global lines in the CLB array, as in the XC4000E. Any Global Low-Skew buffer can, therefore, access every IOB and CLB in the device.

The Global Low-Skew buffers can be driven by either semi-dedicated pads or internal logic.

To use a Global Low-Skew buffer, instantiate a BUFGLS 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=T attribute or property to direct that a BUFGLS be placed in one of the two Global Low-Skew buffers on the top edge of the device, or a LOC=TR to indicate the Global Low-Skew buffer on the top edge of the device, on the right.

Table 16: Pin Descriptions (Continued)

| Pin Name | I/O During Config. | I/O After Config. | Pin Description |
|---|--------------------|-------------------|--|
| TDI, TCK, TMS | I | I/O or I (JTAG) | If boundary scan is used, these pins are Test Data In, Test Clock, and Test Mode Select inputs respectively. They come directly from the pads, bypassing the IOBs. These pins can also be used as inputs to the CLB logic after configuration is completed. If the BSCAN symbol is not placed in the design, all boundary scan functions are inhibited once configuration is completed, and these pins become user-programmable I/O. The pins can be used automatically or user-constrained. To use them, use "LOC=" or place the library components TDI, TCK, and TMS instead of the usual pad symbols. Input or output buffers must still be used. |
| HDC | O | I/O | High During Configuration (HDC) is driven High until the I/O go active. It is available as a control output indicating that configuration is not yet completed. After configuration, HDC is a user-programmable I/O pin. |
| $\overline{\text{LDC}}$ | O | I/O | Low During Configuration ($\overline{\text{LDC}}$) is driven Low until the I/O go active. It is available as a control output indicating that configuration is not yet completed. After configuration, $\overline{\text{LDC}}$ is a user-programmable I/O pin. |
| $\overline{\text{INIT}}$ | I/O | I/O | Before and during configuration, $\overline{\text{INIT}}$ is a bidirectional signal. A 1 k Ω - 10 k Ω external pull-up resistor is recommended. As an active-Low open-drain output, $\overline{\text{INIT}}$ is held Low during the power stabilization and internal clearing of the configuration memory. As an active-Low input, it can be used to hold the FPGA in the internal WAIT state before the start of configuration. Master mode devices stay in a WAIT state an additional 30 to 300 μs after $\overline{\text{INIT}}$ has gone High. During configuration, a Low on this output indicates that a configuration data error has occurred. After the I/O go active, $\overline{\text{INIT}}$ is a user-programmable I/O pin. |
| PGCK1 - PGCK4 (XC4000E only) | Weak Pull-up | I or I/O | Four Primary Global inputs each drive a dedicated internal global net with short delay and minimal skew. If not used to drive a global buffer, any of these pins is a user-programmable I/O. The PGCK1-PGCK4 pins drive the four Primary Global Buffers. Any input pad symbol connected directly to the input of a BUF _{GP} symbol is automatically placed on one of these pins. |
| SGCK1 - SGCK4 (XC4000E only) | Weak Pull-up | I or I/O | Four Secondary Global inputs each drive a dedicated internal global net with short delay and minimal skew. These internal global nets can also be driven from internal logic. If not used to drive a global net, any of these pins is a user-programmable I/O pin. The SGCK1-SGCK4 pins provide the shortest path to the four Secondary Global Buffers. Any input pad symbol connected directly to the input of a BUF _{GS} symbol is automatically placed on one of these pins. |
| GCK1 - GCK8 (XC4000X only) | Weak Pull-up | I or I/O | Eight inputs can each drive a Global Low-Skew buffer. In addition, each can drive a Global Early buffer. Each pair of global buffers can also be driven from internal logic, but must share an input signal. If not used to drive a global buffer, any of these pins is a user-programmable I/O. Any input pad symbol connected directly to the input of a BUF _{GLS} or BUF _{GE} symbol is automatically placed on one of these pins. |
| FCLK1 - FCLK4 (XC4000XLA and XC4000XV only) | Weak Pull-up | I or I/O | Four inputs can each drive a Fast Clock (FCLK) buffer which can deliver a clock signal to any IOB clock input in the octant of the die served by the Fast Clock buffer. Two Fast Clock buffers serve the two IOB octants on the left side of the die and the other two Fast Clock buffers serve the two IOB octants on the right side of the die. On each side of the die, one Fast Clock buffer serves the upper octant and the other serves the lower octant. If not used to drive a Fast Clock buffer, any of these pins is a user-programmable I/O. |

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 HDC , $\overline{\text{LDC}}$, $\overline{\text{INIT}}$ and DONE provide status outputs for the system interface. The outputs $\overline{\text{LDC}}$, $\overline{\text{INIT}}$ and DONE are held Low and 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, 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, 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. 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. DONE can be programmed to go High one CCLK period before or after the I/O become active. Independent of 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 — 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.

Start-up from a User Clock (STARTUP.CLK)

When, instead of CCLK, a user-supplied start-up clock is selected, Q1 is used to bridge the unknown phase relationship between CCLK and the user clock. This arbitration causes an unavoidable one-cycle uncertainty in the timing of the rest of the start-up sequence.

DONE Goes High to Signal End of Configuration

XC4000 Series devices read the expected length count from the bitstream and store it in an internal register. The length count varies according to the number of devices and the composition of the daisy chain. Each device also counts the number of CCLKs during configuration.

Two conditions have to be met in order for the DONE pin to go high:

- the chip's internal memory must be full, and
- the configuration length count must be met, *exactly*.

This is important because the counter that determines when the length count is met begins with the very first CCLK, not the first one after the preamble.

Therefore, if a stray bit is inserted before the preamble, or the data source is not ready at the time of the first CCLK, the internal counter that holds the number of CCLKs will be one ahead of the actual number of data bits read. At the end of configuration, the configuration memory will be full, but the number of bits in the internal counter will not match the expected length count.

As a consequence, a Master mode device will continue to send out CCLKs until the internal counter turns over to zero, and then reaches the correct length count a second time. This will take several seconds [$2^{24} * \text{CCLK period}$] — which is sometimes interpreted as the device not configuring at all.

If it is not possible to have the data ready at the time of the first CCLK, the problem can be avoided by increasing the number in the length count by the appropriate value. The *XACT User Guide* includes detailed information about manually altering the length count.

Note that DONE is an open-drain output and does not go High unless an internal pull-up is activated or an external pull-up is attached. The internal pull-up is activated as the default by the bitstream generation software.

Release of User I/O After DONE Goes High

By default, the user I/O are released one CCLK cycle after the DONE pin goes High. If CCLK is not clocked after DONE goes High, the outputs remain in their initial state — 3-stated, with a 50 k Ω - 100 k Ω pull-up. The delay from DONE High to active user I/O is controlled by an option to the bitstream generation software.

Release of Global Set/Reset After DONE Goes High

By default, Global Set/Reset (GSR) is released two CCLK cycles after the DONE pin goes High. If CCLK is not clocked twice after DONE goes High, all flip-flops are held in their initial set or reset state. The delay from DONE High to GSR inactive is controlled by an option to the bitstream generation software.

Configuration Complete After DONE Goes High

Three full CCLK cycles are required after the DONE pin goes High, as shown in [Figure 47 on page 53](#). If CCLK is not clocked three times after DONE goes High, readback cannot be initiated and most boundary scan instructions cannot be used.

Configuration Through the Boundary Scan Pins

XC4000 Series devices can be configured through the boundary scan pins. The basic procedure is as follows:

- Power up the FPGA with $\overline{\text{INIT}}$ held Low (or drive the $\overline{\text{PROGRAM}}$ pin Low for more than 300 ns followed by a High while holding $\overline{\text{INIT}}$ Low). Holding $\overline{\text{INIT}}$ Low allows enough time to issue the CONFIG command to the FPGA. The pin can be used as I/O after configuration if a resistor is used to hold $\overline{\text{INIT}}$ Low.
- Issue the CONFIG command to the TMS input
- Wait for $\overline{\text{INIT}}$ to go High
- Sequence the boundary scan Test Access Port to the SHIFT-DR state
- Toggle TCK to clock data into TDI pin.

The user must account for all TCK clock cycles after INIT goes High, as all of these cycles affect the Length Count compare.

For more detailed information, refer to the Xilinx application note XAPP017, "*Boundary Scan in XC4000 Devices*." This application note also applies to XC4000E and XC4000X devices.

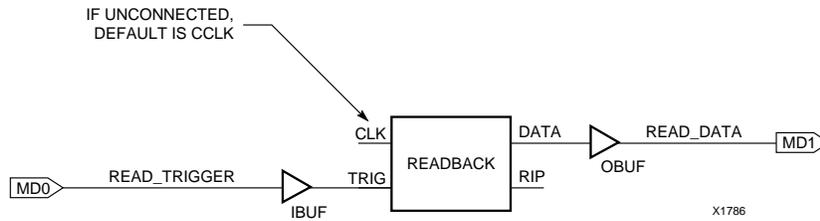


Figure 49: Readback Schematic Example

Readback Options

Readback options are: Read Capture, Read Abort, and Clock Select. They are set with the bitstream generation software.

Read Capture

When the Read Capture option is selected, the readback data stream includes sampled values of CLB and IOB signals. The rising edge of RDBK.TRIG latches the inverted values of the four CLB outputs, the IOB output flip-flops and the input signals I1 and I2. Note that while the bits describing configuration (interconnect, function generators, and RAM content) are *not* inverted, the CLB and IOB output signals *are* inverted.

When the Read Capture option is not selected, the values of the capture bits reflect the configuration data originally written to those memory locations.

If the RAM capability of the CLBs is used, RAM data are available in readback, since they directly overwrite the F and G function-table configuration of the CLB.

RDBK.TRIG is located in the lower-left corner of the device, as shown in [Figure 50](#).

Read Abort

When the Read Abort option is selected, a High-to-Low transition on RDBK.TRIG terminates the readback operation and prepares the logic to accept another trigger.

After an aborted readback, additional clocks (up to one readback clock per configuration frame) may be required to re-initialize the control logic. The status of readback is indicated by the output control net RDBK.RIP. RDBK.RIP is High whenever a readback is in progress.

Clock Select

CCLK is the default clock. However, the user can insert another clock on RDBK.CLK. Readback control and data are clocked on rising edges of RDBK.CLK. If readback must be inhibited for security reasons, the readback control nets are simply not connected.

RDBK.CLK is located in the lower right chip corner, as shown in [Figure 50](#).

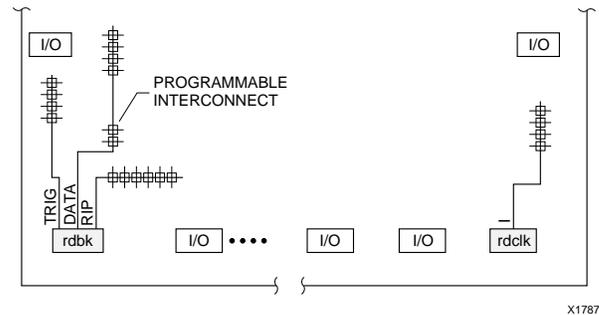


Figure 50: READBACK Symbol in Graphical Editor

Violating the Maximum High and Low Time Specification for the Readback Clock

The readback clock has a maximum High and Low time specification. In some cases, this specification cannot be met. For example, if a processor is controlling readback, an interrupt may force it to stop in the middle of a readback. This necessitates stopping the clock, and thus violating the specification.

The specification is mandatory only on clocking data at the end of a frame prior to the next start bit. The transfer mechanism will load the data to a shift register during the last six clock cycles of the frame, prior to the start bit of the following frame. This loading process is dynamic, and is the source of the maximum High and Low time requirements.

Therefore, the specification only applies to the six clock cycles prior to and including any start bit, including the clocks before the first start bit in the readback data stream. At other times, the frame data is already in the register and the register is not dynamic. Thus, it can be shifted out just like a regular shift register.

The user must precisely calculate the location of the readback data relative to the frame. The system must keep track of the position within a data frame, and disable interrupts before frame boundaries. Frame lengths and data formats are listed in [Table 19](#), [Table 20](#) and [Table 21](#).

Readback with the XChecker Cable

The XChecker Universal Download/Readback Cable and Logic Probe uses the readback feature for bitstream verification. It can also display selected internal signals on the PC or workstation screen, functioning as a low-cost in-circuit emulator.

Table 22: Pin Functions During Configuration

| CONFIGURATION MODE <M2:M1:M0> | | | | | | |
|-------------------------------|--------------------------|------------------------------|-------------------------------|---------------------------------|-------------------------------|----------------|
| SLAVE SERIAL <1:1:1> | MASTER SERIAL <0:0:0> | SYNCH. PERIPHERAL <0:1:1> | ASYNCH. PERIPHERAL <1:0:1> | MASTER PARALLEL DOWN <1:1:0> | MASTER PARALLEL UP <1:0:0> | USER OPERATION |
| M2(HIGH) (I) | M2(LOW) (I) | M2(LOW) (I) | M2(HIGH) (I) | M2(HIGH) (I) | M2(HIGH) (I) | (I) |
| M1(HIGH) (I) | M1(LOW) (I) | M1(HIGH) (I) | M1(LOW) (I) | M1(HIGH) (I) | M1(LOW) (I) | (O) |
| M0(HIGH) (I) | M0(LOW) (I) | M0(HIGH) (I) | M0(HIGH) (I) | M0(LOW) (I) | M0(LOW) (I) | (I) |
| HDC (HIGH) | HDC (HIGH) | HDC (HIGH) | HDC (HIGH) | HDC (HIGH) | HDC (HIGH) | I/O |
| LDC (LOW) | LDC (LOW) | LDC (LOW) | LDC (LOW) | LDC (LOW) | LDC (LOW) | I/O |
| INIT | INIT | INIT | INIT | INIT | INIT | I/O |
| DONE | DONE | DONE | DONE | DONE | DONE | DONE |
| PROGRAM (I) | PROGRAM (I) | PROGRAM (I) | PROGRAM (I) | PROGRAM (I) | PROGRAM (I) | PROGRAM |
| CCLK (I) | CCLK (O) | CCLK (I) | CCLK (O) | CCLK (O) | CCLK (O) | CCLK (I) |
| | | RDY/BUSY (O) | RDY/BUSY (O) | RCLK (O) | RCLK (O) | I/O |
| | | | RS (I) | | | I/O |
| | | | CS0 (I) | | | I/O |
| | | DATA 7 (I) | DATA 7 (I) | DATA 7 (I) | DATA 7 (I) | I/O |
| | | DATA 6 (I) | DATA 6 (I) | DATA 6 (I) | DATA 6 (I) | I/O |
| | | DATA 5 (I) | DATA 5 (I) | DATA 5 (I) | DATA 5 (I) | I/O |
| | | DATA 4 (I) | DATA 4 (I) | DATA 4 (I) | DATA 4 (I) | I/O |
| | | DATA 3 (I) | DATA 3 (I) | DATA 3 (I) | DATA 3 (I) | I/O |
| | | DATA 2 (I) | DATA 2 (I) | DATA 2 (I) | DATA 2 (I) | I/O |
| | | DATA 1 (I) | DATA 1 (I) | DATA 1 (I) | DATA 1 (I) | I/O |
| DIN (I) | DIN (I) | DATA 0 (I) | DATA 0 (I) | DATA 0 (I) | DATA 0 (I) | I/O |
| DOUT | DOUT | DOUT | DOUT | DOUT | DOUT | SGCK4-GCK6-I/O |
| TDI | TDI | TDI | TDI | TDI | TDI | TDI-I/O |
| TCK | TCK | TCK | TCK | TCK | TCK | TCK-I/O |
| TMS | TMS | TMS | TMS | TMS | TMS | TMS-I/O |
| TDO | TDO | TDO | TDO | TDO | TDO | TDO-(O) |
| | | | WS (I) | A0 | A0 | I/O |
| | | | | A1 | A1 | PGCK4-GCK7-I/O |
| | | | CS1 | A2 | A2 | I/O |
| | | | | A3 | A3 | I/O |
| | | | | A4 | A4 | I/O |
| | | | | A5 | A5 | I/O |
| | | | | A6 | A6 | I/O |
| | | | | A7 | A7 | I/O |
| | | | | A8 | A8 | I/O |
| | | | | A9 | A9 | I/O |
| | | | | A10 | A10 | I/O |
| | | | | A11 | A11 | I/O |
| | | | | A12 | A12 | I/O |
| | | | | A13 | A13 | I/O |
| | | | | A14 | A14 | I/O |
| | | | | A15 | A15 | SGCK1-GCK8-I/O |
| | | | | A16 | A16 | PGCK1-GCK1-I/O |
| | | | | A17 | A17 | I/O |
| | | | | A18* | A18* | I/O |
| | | | | A19* | A19* | I/O |
| | | | | A20* | A20* | I/O |
| | | | | A21* | A21* | I/O |
| | | | | | | ALL OTHERS |

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.

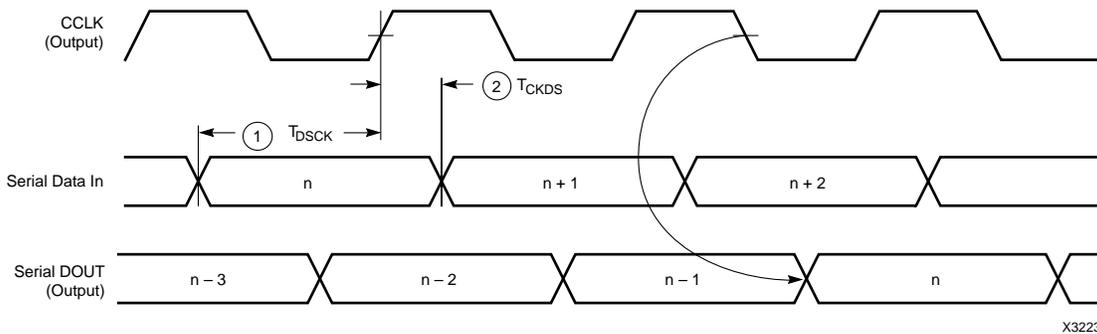
In the bitstream generation software, the user can specify Fast ConfigRate, which, starting several bits into the first frame, increases the CCLK frequency by a factor of eight.

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{LDC} or DONE. Using \overline{LDC} avoids potential contention on the DIN pin, if this pin is configured as user-I/O, but \overline{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, V_{CC} must rise from 2.0 V to V_{CC} min in less than 25 ms, otherwise delay configuration by pulling PROGRAM Low until V_{CC} is valid.
 2. Master Serial mode timing is based on testing in slave mode.

Figure 53: Master Serial Mode Programming Switching Characteristics

Synchronous Peripheral Mode

Synchronous Peripheral mode can also be considered Slave Parallel mode. An external signal drives the CCLK input(s) of the FPGA(s). The first byte of parallel configuration data must be available at the Data inputs of the lead FPGA a short setup time before the rising CCLK edge. Subsequent data bytes are clocked in on every eighth consecutive rising CCLK edge.

The same CCLK edge that accepts data, also causes the RDY/ $\overline{\text{BUSY}}$ output to go High for one CCLK period. The pin name is a misnomer. In Synchronous Peripheral mode it is really an ACKNOWLEDGE signal. Synchronous operation does not require this response, but it is a meaningful signal for test purposes. Note that RDY/ $\overline{\text{BUSY}}$ is pulled High with a high-impedance pullup prior to $\overline{\text{INIT}}$ going High.

The lead FPGA serializes the data and presents the preamble data (and all data that overflows the lead device) on its DOUT pin. There is an internal 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 order to complete the serial shift operation, 10 additional CCLK rising edges are required after the last data byte has been loaded, plus one more CCLK cycle for each daisy-chained device.

Synchronous Peripheral mode is selected by a <011> on the mode pins (M2, M1, M0).

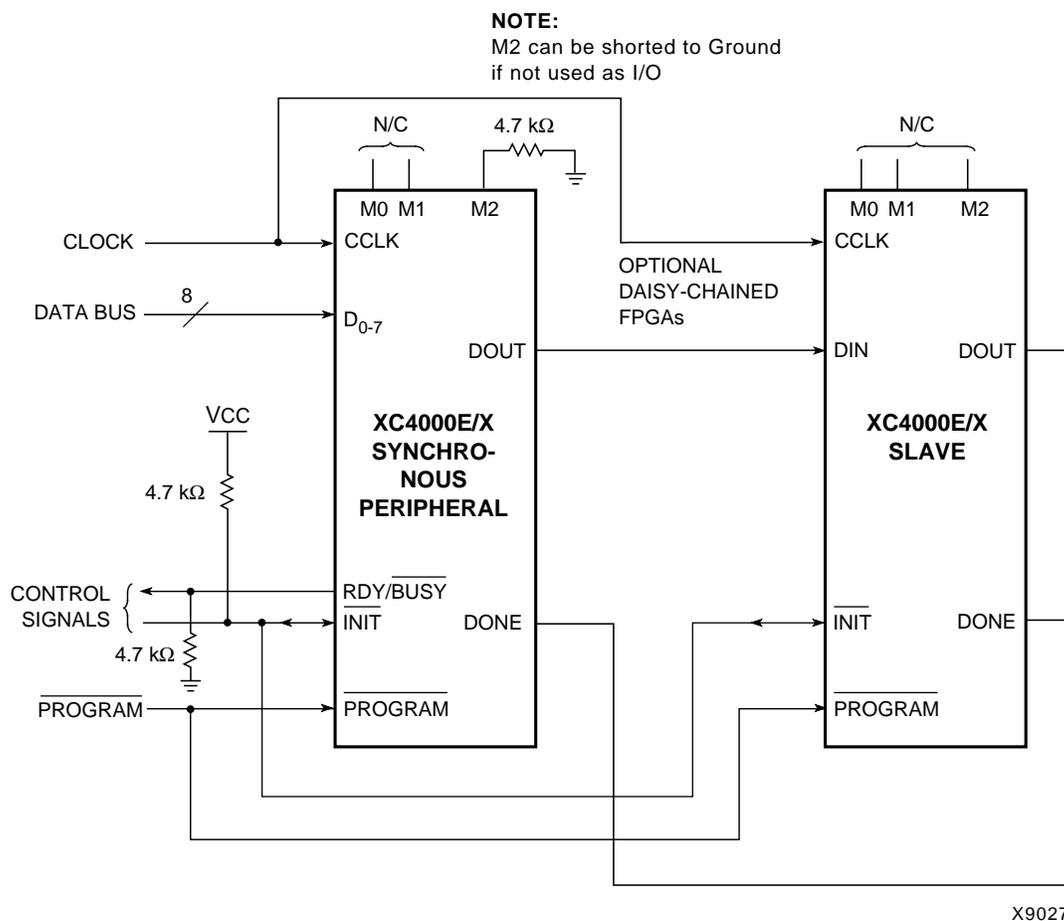
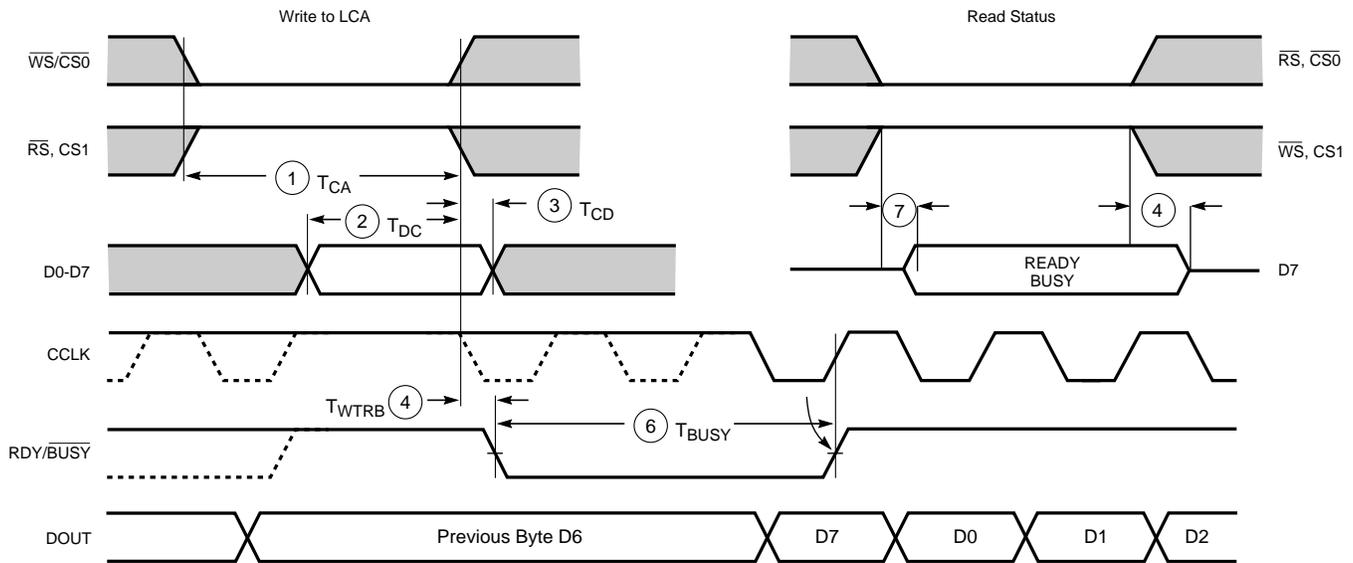


Figure 56: Synchronous Peripheral Mode Circuit Diagram



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 $\overline{\text{INIT}}$ pins of all daisy-chained FPGAs are High.
 2. The time from the end of $\overline{\text{WS}}$ 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 $\overline{\text{WS}}$. RDY/BUSY will go active within 60 ns after the end of $\overline{\text{WS}}$. 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

Product Availability

Table 24, Table 25, and Table 26 show the planned packages and speed grades for XC4000-Series devices. Call your local sales office for the latest availability information, or see the Xilinx website at <http://www.xilinx.com> for the latest revision of the specifications.

Table 24: Component Availability Chart for XC4000XL FPGAs

| | PINS | | | | | | | | | | | | | | | | | | | | | | |
|----------|------|-------------|-------------|-------------|-------------|-----------------|----------------|-------------|-------------|-----------------|----------------|-------------|----------------|-------------|------------|------------|----------------|------------|------------|------------|------------|------------|------------|
| | | 84 | 100 | 100 | 144 | 144 | 160 | 160 | 176 | 176 | 208 | 208 | 240 | 240 | 256 | 299 | 304 | 352 | 411 | 432 | 475 | 559 | 560 |
| | | Plast. PLCC | Plast. PQFP | Plast. VQFP | Plast. TQFP | High-Perf. TQFP | High-Perf. QFP | Plast. PQFP | Plast. TQFP | High-Perf. TQFP | High-Perf. QFP | Plast. PQFP | High-Perf. QFP | Plast. PQFP | Plast. BGA | Ceram. PGA | High-Perf. QFP | Plast. BGA | Ceram. PGA | Plast. BGA | Ceram. PGA | Ceram. PGA | Plast. BGA |
| CODE | PC84 | PQ100 | VQ100 | TQ144 | HT144 | HQ160 | PQ160 | TQ176 | HT176 | HQ208 | PQ208 | HQ240 | PQ240 | BG256 | PG299 | HQ304 | BG352 | PG411 | BG432 | PG475 | PG559 | BG560 | |
| XC4002XL | -3 | C | I | C | | | | | | | | | | | | | | | | | | | |
| | -2 | C | I | C | C | | | | | | | | | | | | | | | | | | |
| | -1 | C | I | C | C | | | | | | | | | | | | | | | | | | |
| | -09C | C | C | C | | | | | | | | | | | | | | | | | | | |
| XC4005XL | -3 | C | I | C | C | | | | | | | | | | | | | | | | | | |
| | -2 | C | I | C | C | | | | | | | | | | | | | | | | | | |
| | -1 | C | I | C | C | | | | | | | | | | | | | | | | | | |
| | -09C | C | C | C | C | | | | | | | | | | | | | | | | | | |
| XC4010XL | -3 | C | I | C | | | | | | | | | | | | | | | | | | | |
| | -2 | C | I | C | | | | | | | | | | | | | | | | | | | |
| | -1 | C | I | C | | | | | | | | | | | | | | | | | | | |
| | -09C | C | C | | C | | | | | | | | | | | | | | | | | | |
| XC4013XL | -3 | | | | | | | | | | | | | | | | | | | | | | |
| | -2 | | | | | | | | | | | | | | | | | | | | | | |
| | -1 | | | | | | | | | | | | | | | | | | | | | | |
| | -09C | | | | | | | | | | | | | | | | | | | | | | |
| XC4020XL | -3 | | | | | | | | | | | | | | | | | | | | | | |
| | -2 | | | | | | | | | | | | | | | | | | | | | | |
| | -1 | | | | | | | | | | | | | | | | | | | | | | |
| | -09C | | | | | | | | | | | | | | | | | | | | | | |
| XC4028XL | -3 | | | | | | | | | | | | | | | | | | | | | | |
| | -2 | | | | | | | | | | | | | | | | | | | | | | |
| | -1 | | | | | | | | | | | | | | | | | | | | | | |
| | -09C | | | | | | | | | | | | | | | | | | | | | | |
| XC4036XL | -3 | | | | | | | | | | | | | | | | | | | | | | |
| | -2 | | | | | | | | | | | | | | | | | | | | | | |
| | -1 | | | | | | | | | | | | | | | | | | | | | | |
| | -09C | | | | | | | | | | | | | | | | | | | | | | |
| XC4044XL | -3 | | | | | | | | | | | | | | | | | | | | | | |
| | -2 | | | | | | | | | | | | | | | | | | | | | | |
| | -1 | | | | | | | | | | | | | | | | | | | | | | |
| | -09C | | | | | | | | | | | | | | | | | | | | | | |
| XC4052XL | -3 | | | | | | | | | | | | | | | | | | | | | | |
| | -2 | | | | | | | | | | | | | | | | | | | | | | |
| | -1 | | | | | | | | | | | | | | | | | | | | | | |
| | -09C | | | | | | | | | | | | | | | | | | | | | | |
| XC4062XL | -3 | | | | | | | | | | | | | | | | | | | | | | |
| | -2 | | | | | | | | | | | | | | | | | | | | | | |
| | -1 | | | | | | | | | | | | | | | | | | | | | | |
| | -09C | | | | | | | | | | | | | | | | | | | | | | |
| XC4085XL | -3 | | | | | | | | | | | | | | | | | | | | | | |
| | -2 | | | | | | | | | | | | | | | | | | | | | | |
| | -1 | | | | | | | | | | | | | | | | | | | | | | |
| | -09C | | | | | | | | | | | | | | | | | | | | | | |

1/29/99

C = Commercial $T_J = 0^\circ$ to $+85^\circ\text{C}$
 I = Industrial $T_J = -40^\circ\text{C}$ to $+100^\circ\text{C}$

User I/O Per Package

Table 27, Table 28, and Table 29 show the number of user I/Os available in each package for XC4000-Series devices. Call your local sales office for the latest availability information, or see the Xilinx website at <http://www.xilinx.com> for the latest revision of the specifications.

Table 27: User I/O Chart for XC4000XL FPGAs

| Device | Max I/O | Maximum User Accessible I/O by Package Type | | | | | | | | | | | | | | | | | | | | | | |
|----------|---------|---|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--|
| | | PC84 | PQ100 | VQ100 | TQ144 | HT144 | HQ160 | PQ160 | TQ176 | HT176 | HQ208 | PQ208 | HQ240 | PQ240 | BG256 | PG299 | HQ304 | BG352 | PG411 | BG432 | PG475 | PG559 | BG560 | |
| XC4002XL | 64 | 61 | 64 | 64 | | | | | | | | | | | | | | | | | | | | |
| XC4005XL | 112 | 61 | 77 | 77 | 112 | | | 112 | | | 112 | | | | | | | | | | | | | |
| XC4010XL | 160 | 61 | 77 | | 113 | | | 129 | 145 | | 160 | | | 160 | | | | | | | | | | |
| XC4013XL | 192 | | | | | 113 | | 129 | | 145 | 160 | | 192 | 192 | | | | | | | | | | |
| XC4020XL | 224 | | | | | 113 | | 129 | | 145 | 160 | | 192 | 205 | | | | | | | | | | |
| XC4028XL | 256 | | | | | | 129 | | | | 160 | | 193 | 205 | 256 | 256 | 256 | | | | | | | |
| XC4036XL | 288 | | | | | | 129 | | | | 160 | | 193 | | | 256 | 288 | 288 | 288 | | | | | |
| XC4044XL | 320 | | | | | | 129 | | | | 160 | | 193 | | | 256 | 289 | 320 | 320 | | | | | |
| XC4052XL | 352 | | | | | | | | | | | 193 | | | | 256 | | 352 | 352 | | | | 352 | |
| XC4062XL | 384 | | | | | | | | | | | 193 | | | | 256 | | | 352 | 384 | | | 384 | |
| XC4085XL | 448 | | | | | | | | | | | | | | | | | 352 | | | 448 | 448 | | |

1/29/99

Table 28: User I/O Chart for XC4000E FPGAs

| Device | Max I/O | Maximum User Accessible I/O by Package Type | | | | | | | | | | | | | | | |
|---------|---------|---|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| | | PC84 | PQ100 | VQ100 | PG120 | TQ144 | PG156 | PQ160 | PG191 | HQ208 | PQ208 | PG223 | BG225 | HQ240 | PQ240 | PG299 | HQ304 |
| XC4003E | 80 | 61 | 77 | 77 | 80 | | | | | | | | | | | | |
| XC4005E | 112 | 61 | 77 | | | 112 | 112 | 112 | | | 112 | | | | | | |
| XC4006E | 128 | 61 | | | | 113 | 125 | 128 | | | 128 | | | | | | |
| XC4008E | 144 | 61 | | | | | | 129 | 144 | | 144 | | | | | | |
| XC4010E | 160 | 61 | | | | | | 129 | 160 | 160 | 160 | | 160 | | | | |
| XC4013E | 192 | | | | | | | 129 | | 160 | 160 | 192 | 192 | 192 | 192 | | |
| XC4020E | 224 | | | | | | | | | 160 | | 192 | | 193 | | | |
| XC4025E | 256 | | | | | | | | | | | 192 | | 193 | | 256 | 256 |

1/29/99

Table 29: User I/O Chart for XC4000EX FPGAs

| Device | Max I/O | Maximum User Accessible I/O by Package Type | | | | | | |
|----------|---------|---|-------|-------|-------|-------|-------|-------|
| | | HQ208 | HQ240 | PG299 | HQ304 | BG352 | PG411 | BG432 |
| XC4028EX | 256 | 160 | 193 | 256 | 256 | 256 | | |
| XC4036EX | 288 | | 193 | | 256 | 288 | 288 | 288 |

1/29/99