Welcome to 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	120
Number of Logic Elements/Cells	480
Total RAM Bits	-
Number of I/O	81
Number of Gates	6000
Voltage - Supply	4.75V ~ 5.25V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 85°C (TJ)
Package / Case	100-BQFP
Supplier Device Package	100-PQFP (20x14)
Purchase URL	https://www.e-xfl.com/product-detail/xilinx/xc5204-6pq100c

Matrix through a hierarchy of different-length metal segments in both the horizontal and vertical directions. A pro-

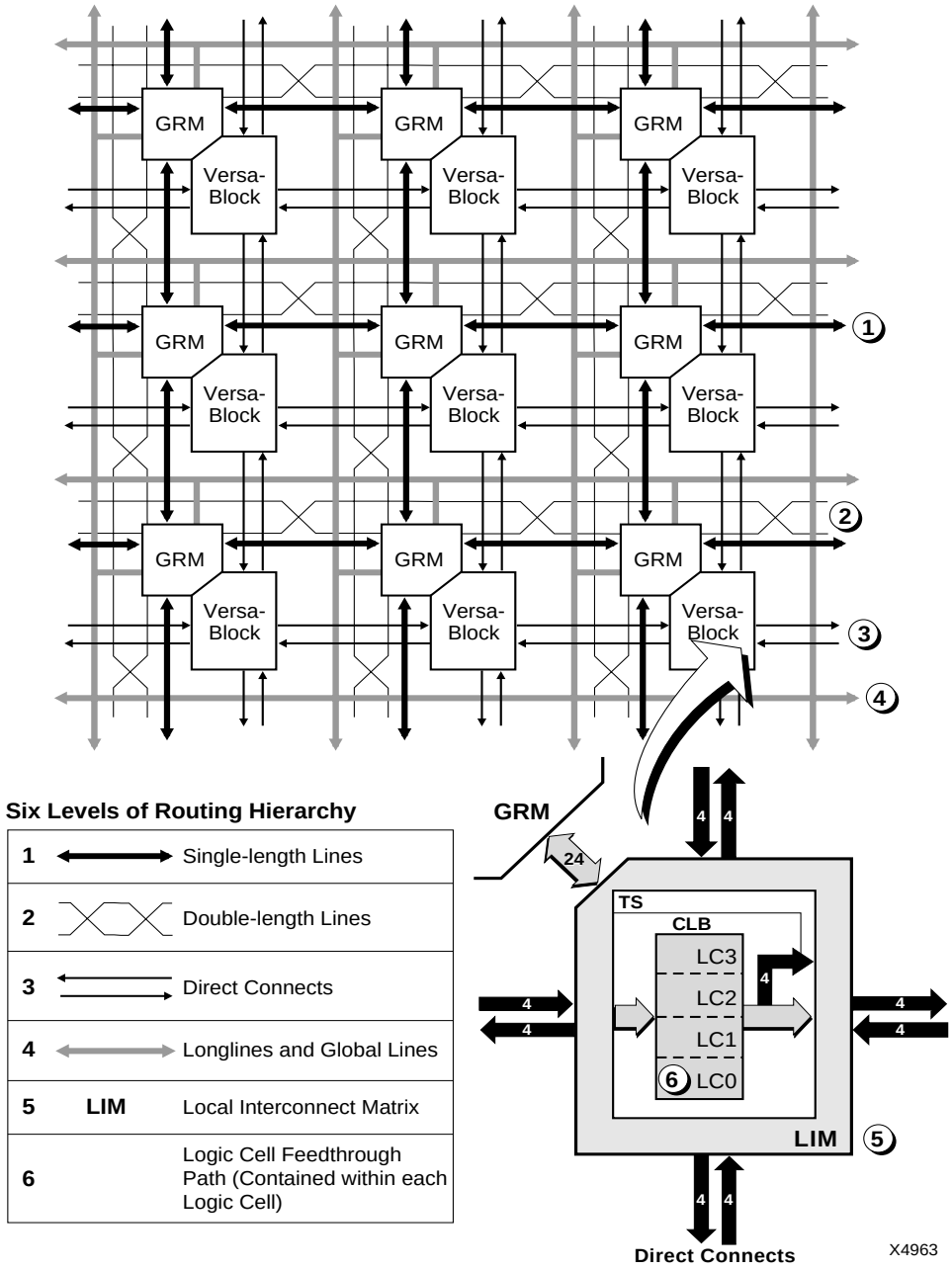


Figure 15: XC5200 Interconnect Structure

grammable interconnect point (PIP) establishes an electrical connection between two wire segments. The PIP, consisting of a pass transistor switch controlled by a memory element, provides bidirectional (in some cases, unidirectional) connection between two adjoining wires. A collection of PIPs inside the General Routing Matrix and in the Local Interconnect Matrix provides connectivity between various types of metal segments. A hierarchy of PIPs and associated routing segments combine to provide a powerful interconnect hierarchy:

- Forty bidirectional single-length segments per CLB provide ten routing channels to each of the four neighboring CLBs in four directions.
- Sixteen bidirectional double-length segments per CLB provide four routing channels to each of four other (non-neighboring) CLBs in four directions.
- Eight horizontal and eight vertical bidirectional Longline

segments span the width and height of the chip, respectively.

Two low-skew horizontal and vertical unidirectional global-line segments span each row and column of the chip, respectively.

Single- and Double-Length Lines

The single- and double-length bidirectional line segments make up the bulk of the routing channels. The double-length lines hop across every other CLB to reduce the propagation delays in speed-critical nets. Regenerating the signal strength is recommended after traversing three or four such segments. Xilinx place-and-route software automatically connects buffers in the path of the signal as necessary. Single- and double-length lines cannot drive onto Longlines and global lines; Longlines and global lines can, however, drive onto single- and double-length lines. As a general rule, Longline and global-line connections to the general routing matrix are unidirectional, with the signal direction from these lines toward the routing matrix.

Longlines

Longlines are used for high-fan-out signals, 3-state busses, low-skew nets, and faraway destinations. Row and column splitter PIPs in the middle of the array effectively double the total number of Longlines by electrically dividing them into two separated half-lines. Longlines are driven by the 3-state buffers in each CLB, and are driven by similar buffers at the periphery of the array from the VersaRing I/O Interface.

Bus-oriented designs are easily implemented by using Longlines in conjunction with the 3-state buffers in the CLB and in the VersaRing. Additionally, weak keeper cells at the periphery retain the last valid logic level on the Longlines when all buffers are in 3-state mode.

Longlines connect to the single-length or double-length lines, or to the logic inside the CLB, through the General Routing Matrix. The only manner in which a Longline can be driven is through the four 3-state buffers; therefore, a Longline-to-Longline or single-line-to-Longline connection through PIPs in the General Routing Matrix is not possible. Again, as a general rule, long- and global-line connections to the General Routing Matrix are unidirectional, with the signal direction from these lines toward the routing matrix.

The XC5200 family has no pull-ups on the ends of the Longlines sourced by TBUFs, unlike the XC4000 Series. Consequently, wired functions (i.e., WAND and WORAND) and wide multiplexing functions requiring pull-ups for undefined states (i.e., bus applications) must be implemented in a different way. In the case of the wired functions, the same functionality can be achieved by taking advantage of the carry/cascade logic described above, implementing a wide logic function in place of the wired function. In the case of 3-state bus applications, the user must insure that all states of the multiplexing function are defined. This process is as simple as adding an additional TBUF to drive the bus High when the previously undefined states are activated.

Global Lines

Global buffers in Xilinx FPGAs are special buffers that drive a dedicated routing network called Global Lines, as shown in Figure 16. This network is intended for high-fanout clocks or other control signals, to maximize speed and minimize skewing while distributing the signal to many loads.

The XC5200 family has a total of four global buffers (BUFG symbol in the library), each with its own dedicated routing channel. Two are distributed vertically and two horizontally throughout the FPGA.

The global lines provide direct input only to the CLB clock pins. The global lines also connect to the General Routing Matrix to provide access from these lines to the function generators and other control signals.

Four clock input pads at the corners of the chip, as shown in Figure 16, provide a high-speed, low-skew clock network to each of the four global-line buffers. In addition to the dedicated pad, the global lines can be sourced by internal logic. PIPs from several routing channels within the VersaRing can also be configured to drive the global-line buffers.

Details of all the programmable interconnect for a CLB is shown in Figure 17.

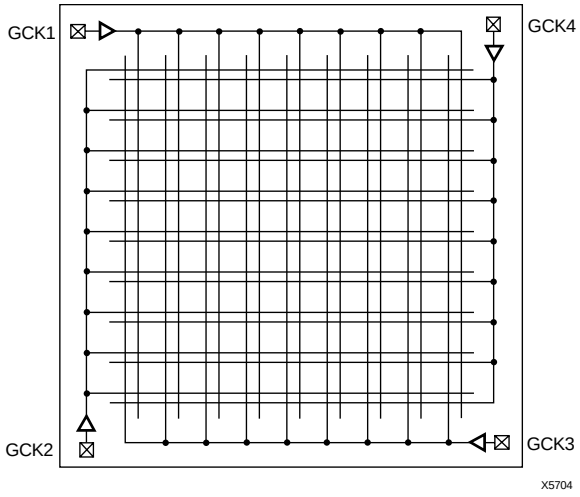


Figure 16: Global Lines

VersaRing Input/Output Interface

The VersaRing, shown in Figure 18, is positioned between the core logic and the pad ring; it has all the routing resources of a VersaBlock without the CLB logic. The VersaRing decouples the core logic from the I/O pads. Each VersaRing Cell provides up to four pad-cell connections on one side, and connects directly to the CLB ports on the other side.

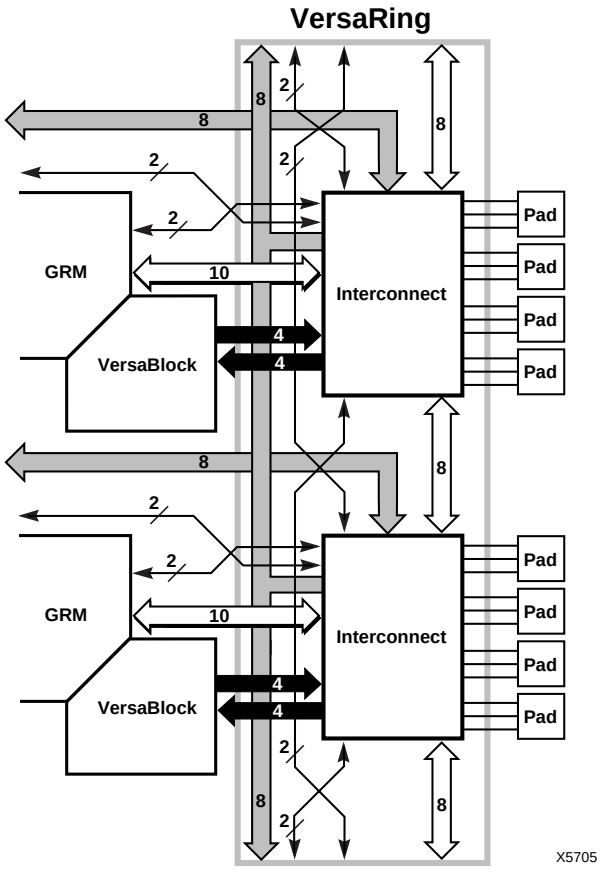


Figure 18: VersaRing I/O Interface

Boundary Scan

The “bed of nails” has been the traditional method of testing electronic assemblies. This approach has become less appropriate, due to closer pin spacing and more sophisticated assembly methods like surface-mount technology and multi-layer boards. The IEEE boundary scan standard 1149.1 was developed to facilitate board-level testing of electronic assemblies. Design and test engineers can imbed a standard test logic structure in their device to achieve high fault coverage for I/O and internal logic. This structure is easily implemented with a four-pin interface on any boundary scan-compatible IC. IEEE 1149.1-compatible devices may be serial daisy-chained together, connected in parallel, or a combination of the two.

XC5200 devices support all the mandatory boundary-scan instructions specified in the IEEE standard 1149.1. A Test Access Port (TAP) and registers are provided that implement the EXTEST, SAMPLE/PRELOAD, and BYPASS instructions. The TAP can also support two USERCODE instructions. When the boundary scan configuration option is selected, three normal user I/O pins become dedicated inputs for these functions. Another user output pin becomes the dedicated boundary scan output.

Boundary-scan operation is independent of individual IOB configuration and package type. All IOBs are treated as independently controlled bidirectional pins, including any unbonded IOBs. Retaining the bidirectional test capability after configuration provides flexibility for interconnect testing.

Also, internal signals can be captured during EXTEST by connecting them to unbonded IOBs, or to the unused outputs in IOBs used as unidirectional input pins. This technique partially compensates for the lack of INTEST support.

The user can serially load commands and data into these devices to control the driving of their outputs and to examine their inputs. This method is an improvement over bed-of-nails testing. It avoids the need to over-drive device outputs, and it reduces the user interface to four pins. An optional fifth pin, a reset for the control logic, is described in the standard but is not implemented in Xilinx devices.

The dedicated on-chip logic implementing the IEEE 1149.1 functions includes a 16-state machine, an instruction register and a number of data registers. The functional details can be found in the IEEE 1149.1 specification and are also discussed in the Xilinx application note XAPP 017: *“Boundary Scan in XC4000 and XC5200 Series devices”*

Figure 19 on page 99 is a diagram of the XC5200-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.

The public boundary-scan instructions are always available prior to configuration. After configuration, the public instructions and any USERCODE instructions are only available if specified in the design. While SAMPLE and BYPASS are available during configuration, it is recommended that boundary-scan operations not be performed during this transitory period.

In addition to the test instructions outlined above, the boundary-scan circuitry can be used to configure the FPGA device, and to read back the configuration data.

All of the XC4000 boundary-scan modes are supported in the XC5200 family. Three additional outputs for the User-Register are provided (Reset, Update, and Shift), repre-

Table 9: 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. In this case, they must be called out by special schematic definitions. To use these pins, 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 kΩ - 10 kΩ 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 50 to 250 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.
GCK1 - GCK4	Weak Pull-up	I or I/O	Four 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 GCK1-GCK4 pins provide the shortest path to the four Global Buffers. Any input pad symbol connected directly to the input of a BUFG symbol is automatically placed on one of these pins.
$\overline{\text{CS0}}$, $\overline{\text{CS1}}$, $\overline{\text{WS}}$, $\overline{\text{RS}}$	I	I/O	These four inputs are used in Asynchronous Peripheral mode. The chip is selected when $\overline{\text{CS0}}$ is Low and CS1 is High. While the chip is selected, a Low on Write Strobe ($\overline{\text{WS}}$) loads the data present on the D0 - D7 inputs into the internal data buffer. A Low on Read Strobe ($\overline{\text{RS}}$) changes D7 into a status output — High if Ready, Low if Busy — and drives D0 - D6 High. In Express mode, CS1 is used as a serial-enable signal for daisy-chaining. $\overline{\text{WS}}$ and $\overline{\text{RS}}$ should be mutually exclusive, but if both are Low simultaneously, the Write Strobe overrides. After configuration, these are user-programmable I/O pins.
A0 - A17	O	I/O	During Master Parallel configuration, these 18 output pins address the configuration EPROM. After configuration, they are user-programmable I/O pins.
D0 - D7	I	I/O	During Master Parallel, Peripheral, and Express configuration, these eight input pins receive configuration data. After configuration, they are user-programmable I/O pins.
DIN	I	I/O	During Slave Serial or Master Serial configuration, DIN is the serial configuration data input receiving data on the rising edge of CCLK. During Parallel configuration, DIN is the D0 input. After configuration, DIN is a user-programmable I/O pin.
DOUT	O	I/O	During configuration in any mode but Express mode, DOUT is the serial configuration data output that can drive the DIN of daisy-chained slave FPGAs. DOUT data changes on the falling edge of CCLK. In Express mode, DOUT is the status output that can drive the CS1 of daisy-chained FPGAs, to enable and disable downstream devices. After configuration, DOUT is a user-programmable I/O pin.

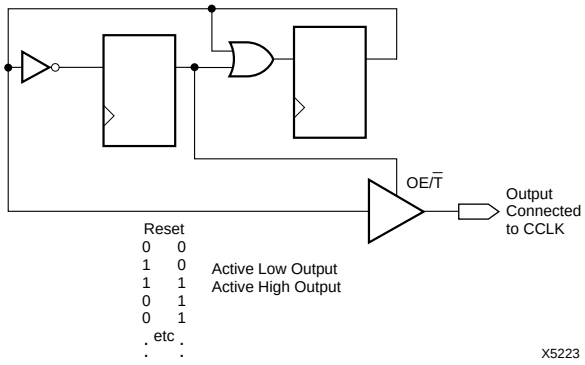


Figure 22: CCLK Generation for XC3000 Master Driving an XC5200-Series Slave

Express Mode

Express mode is similar to Slave Serial mode, except the data is presented in parallel format, and is clocked into the target device a byte at a time rather than a bit at a time. The data is loaded in parallel into eight different columns: it is not internally serialized. Eight bits of configuration data are loaded with every CCLK cycle, therefore this configuration mode runs at eight times the data rate of the other six modes. In this mode the XC5200 family is capable of supporting a CCLK frequency of 10 MHz, which is equivalent to an 80 MHz serial rate, because eight bits of configuration data are being loaded per CCLK cycle. An XC5210 in the Express mode, for instance, can be configured in about 2 ms. The Express mode does not support CRC error checking, but does support constant-field error checking. A length count is not used in Express mode.

In the Express configuration mode, an external signal drives the CCLK input(s). The first byte of parallel configuration data must be available at the D inputs of the FPGA devices a short set-up time before the second rising CCLK edge. Subsequent data bytes are clocked in on each consecutive rising CCLK edge. See Figure 38 on page 123.

Bitstream generation currently generates a bitstream sufficient to program in all configuration modes except Express. Extra CCLK cycles are necessary to complete the configuration, since in this mode data is read at a rate of eight bits per CCLK cycle instead of one bit per cycle. Normally the entire start-up sequence requires a number of bits that is equal to the number of CCLK cycles needed. An additional five CCLKs (equivalent to 40 extra bits) will guarantee completion of configuration, regardless of the start-up options chosen.

Multiple slave devices with identical configurations can be wired with parallel D0-D7 inputs. In this way, multiple devices can be configured simultaneously.

Pseudo Daisy Chain

Multiple devices with different configurations can be connected together in a pseudo daisy chain, provided that all of the devices are in Express mode. A single combined bitstream is used to configure the chain of Express mode devices, but the input data bus must drive D0-D7 of each device. Tie High the CS1 pin of the first device to be configured, or leave it floating in the XC5200 since it has an internal pull-up. Connect the DOUT pin of each FPGA to the CS1 pin of the next device in the chain. The D0-D7 inputs are wired to each device in parallel. The DONE pins are wired together, with one or more internal DONE pull-ups activated. Alternatively, a 4.7 kΩ external resistor can be used, if desired. (See Figure 37 on page 122.) CCLK pins are tied together.

The requirement that all DONE pins in a daisy chain be wired together applies only to Express mode, and only if all devices in the chain are to become active simultaneously. All devices in Express mode are synchronized to the DONE pin. User I/O for each device become active after the DONE pin for that device goes High. (The exact timing is determined by options to the bitstream generation software.) Since the DONE pin is open-drain and does not drive a High value, tying the DONE pins of all devices together prevents all devices in the chain from going High until the last device in the chain has completed its configuration cycle.

The status pin DOUT is pulled LOW two internal-oscillator cycles (nominally 1 MHz) after $\overline{\text{INIT}}$ is recognized as High, and remains Low until the device's configuration memory is full. Then DOUT is pulled High to signal the next device in the chain to accept the configuration data on the D7-D0 bus. All devices receive and recognize the six bytes of preamble and length count, irrespective of the level on CS1; but subsequent frame data is accepted only when CS1 is High and the device's configuration memory is not already full.

Setting CCLK Frequency

For Master modes, CCLK can be generated in one of three frequencies. In the default slow mode, the frequency is nominally 1 MHz. In fast CCLK mode, the frequency is nominally 12 MHz. In medium CCLK mode, the frequency is nominally 6 MHz. The frequency range is -50% to +50%. The frequency is selected by an option when running the bitstream generation software. If an XC5200-Series Master is driving an XC3000- or XC2000-family slave, slow CCLK mode must be used. Slow mode is the default.

Table 11: XC5200 Bitstream Format

Data Type	Value	Occurrences
Fill Byte	11111111	Once per bitstream
Preamble	11110010	
Length Counter	COUNT(23:0)	
Fill Byte	11111111	

Table 11: XC5200 Bitstream Format

Data Type	Value	Occurrences
Start Byte	11111110	Once per data frame
Data Frame *	DATA(N-1:0)	
Cyclic Redundancy Check or Constant Field Check	CRC(3:0) or 0110	
Fill Nibble	1111	
Extend Write Cycle	FFFFFF	
Postamble	11111110	Once per device
Fill Bytes (30)	FFFF…FF	
Start-Up Byte	FF	Once per bitstream
*Bits per Frame (N) depends on device size, as described for table 11.		

Data Stream Format

The data stream ("bitstream") format is identical for all configuration modes, with the exception of Express mode. In Express mode, the device becomes active when DONE goes High, therefore no length count is required. Additionally, CRC error checking is not supported in Express mode.

The data stream formats are shown in Table 11. Express mode data is shown with D0 at the left and D7 at the right. For all other modes, bit-serial data is read from left to right, and byte-parallel data is effectively assembled from this serial bitstream, with the first bit in each byte assigned to D0.

The configuration data stream begins with a string of eight ones, a preamble code, followed by a 24-bit length count and a separator field of ones (or 24 fill bits, in Express mode). This header is followed by the actual configuration data in frames. The length and number of frames depends on the device type (see Table 12). Each frame begins with a start field and ends with an error check. In all modes except Express mode, a postamble code is required to signal the end of data for a single device. In all cases, additional start-up bytes of data are required to provide four clocks for the startup sequence at the end of configuration. Long daisy chains require additional startup bytes to shift the last data through the chain. All startup bytes are don't-cares; these bytes are not included in bitstreams created by the Xilinx software.

In Express mode, only non-CRC error checking is supported. In all other modes, a selection of CRC or non-CRC error checking is allowed by the bitstream generation software. The non-CRC error checking tests for a designated end-of-frame field for each frame. For CRC error checking, the software calculates a running CRC and inserts a unique four-bit partial check at the end of each frame. The 11-bit CRC check of the last frame of an FPGA includes the last seven data bits.

Detection of an error results in the suspension of data loading and the pulling down of the $\overline{\text{INIT}}$ pin. In Master modes, CCLK and address signals continue to operate externally. The user must detect $\overline{\text{INIT}}$ and initialize a new configuration by pulsing the $\overline{\text{PROGRAM}}$ pin Low or cycling Vcc.

Table 12: Internal Configuration Data Structure

Device	VersaBlock Array	PROM Size (bits)	Xilinx Serial PROM Needed
XC5202	8 x 8	42,416	XC1765E
XC5204	10 x 12	70,704	XC17128E
XC5206	14 x 14	106,288	XC17128E
XC5210	18 x 18	165,488	XC17256E
XC5215	22 x 22	237,744	XC17256E

Bits per Frame = (34 x number of Rows) + 28 for the top + 28 for the bottom + 4 splitter bits + 8 start bits + 4 error check bits + 4 fill bits * + 24 extended write bits
= (34 x number of Rows) + 100
* In the XC5202 (8 x 8), there are 8 fill bits per frame, not 4

Number of Frames = (12 x number of Columns) + 7 for the left edge + 8 for the right edge + 1 splitter bit
= (12 x number of Columns) + 16

Program Data = (Bits per Frame x Number of Frames) + 48 header bits + 8 postamble bits + 240 fill bits + 8 start-up bits
= (Bits per Frame x Number of Frames) + 304

PROM Size = Program Data

Cyclic Redundancy Check (CRC) for Configuration and Readback

The Cyclic Redundancy Check is a method of error detection in data transmission applications. Generally, the transmitting system performs a calculation on the serial bitstream. The result of this calculation is tagged onto the data stream as additional check bits. The receiving system performs an identical calculation on the bitstream and compares the result with the received checksum.

Each data frame of the configuration bitstream has four error bits at the end, as shown in Table 11. If a frame data error is detected during the loading of the FPGA, the configuration process with a potentially corrupted bitstream is terminated. The FPGA pulls the $\overline{\text{INIT}}$ pin Low and goes into a Wait state.

During Readback, 11 bits of the 16-bit checksum are added to the end of the Readback data stream. The checksum is computed using the CRC-16 CCITT polynomial, as shown in Figure 23. The checksum consists of the 11 most significant bits of the 16-bit code. A change in the checksum indicates a change in the Readback bitstream. A comparison to a previous checksum is meaningful only if the readback data is independent of the current device state. CLB outputs should not be included (Read Capture option not used). Statistically, one error out of 2048 might go undetected.

Configuration

The length counter begins counting immediately upon entry into the configuration state. In slave-mode operation it is important to wait at least two cycles of the internal 1-MHz clock oscillator after $\overline{\text{INIT}}$ is recognized before toggling CCLK and feeding the serial bitstream. Configuration will not begin until the internal configuration logic reset is released, which happens two cycles after $\overline{\text{INIT}}$ goes High. A master device's configuration is delayed from 32 to 256 µs to ensure proper operation with any slave devices driven by the master device.

The 0010 preamble code, included for all modes except Express mode, 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. In Express mode, the length count bits are ignored, and DOUT is held Low, to disable the next device in the pseudo daisy chain.

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. In Express mode, when the first device is fully programmed, DOUT goes High to enable the next device in the chain.

Delaying Configuration After Power-Up

To delay master mode configuration after power-up, pull the bidirectional $\overline{\text{INIT}}$ pin Low, using an open-collector (open-drain) driver. (See Figure 12.)

Using an open-collector or open-drain driver to hold $\overline{\text{INIT}}$ Low before the beginning of master mode 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 at the right time.

Figure 25 describes start-up timing for the three Xilinx families in detail. Express mode configuration always uses either CCLK_SYNC or UCLK_SYNC timing, the other 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/XC5200 Series offers additional flexibility. The three events — DONE going High, the internal 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 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 25, 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/XC5200 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 bitstream 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 26. 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 25 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.

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

In all configuration modes except Express mode, XC5200-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} * $ 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.

In Express mode, there is no length count. The DONE pin for each device goes High when the device has received its quota of configuration data. Wiring the DONE pins of several devices together delays start-up of all devices until all are fully configured.

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 20 kΩ - 100 kΩ pull-up. The delay from

Note that in XC5200-Series devices, configuration data is *not* inverted with respect to configuration as it is in XC2000 and XC3000 families.

Readback of Express mode bitstreams results in data that does not resemble the original bitstream, because the bitstream format differs from other modes.

XC5200-Series Readback does not use any dedicated pins, but uses four internal nets (RDBK.TRIG, RDBK.DATA, RDBK.RIP and RDBK.CLK) that can be routed to any IOB. To access the internal Readback signals, place the READBACK library symbol and attach the appropriate pad symbols, as shown in Figure 27.

After Readback has been initiated by a Low-to-High transition on RDBK.TRIG, the RDBK.RIP (Read In Progress) output goes High on the next rising edge of RDBK.CLK. Subsequent rising edges of this clock shift out Readback data on the RDBK.DATA net.

Readback data does not include the preamble, but starts with five dummy bits (all High) followed by the Start bit (Low) of the first frame. The first two data bits of the first frame are always High.

Each frame ends with four error check bits. They are read back as High. The last seven bits of the last frame are also read back as High. An additional Start bit (Low) and an 11-bit Cyclic Redundancy Check (CRC) signature follow, before RDBK.RIP returns Low.

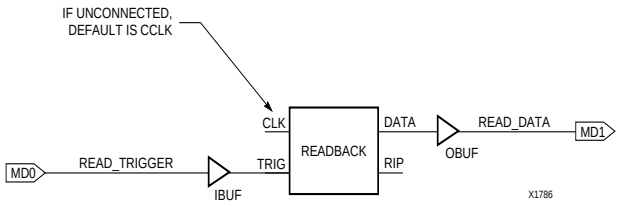


Figure 27: Readback Schematic Example

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 CLB outputs and the IOB output and input signals. Note that while the bits describing configuration (interconnect and function generators) 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.

The readback signals are located in the lower-left corner of the device.

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.

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 11 and Table 12.

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.

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 28 shows a full master/slave system. An XC5200-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.

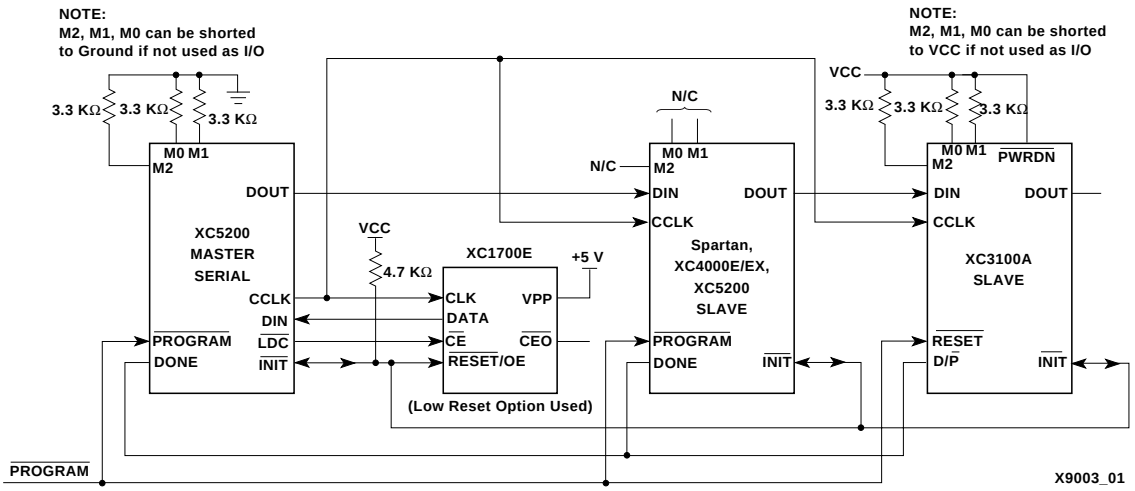
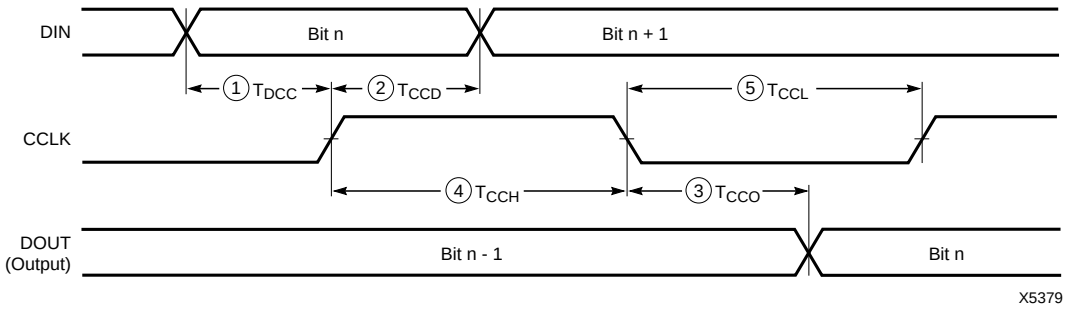


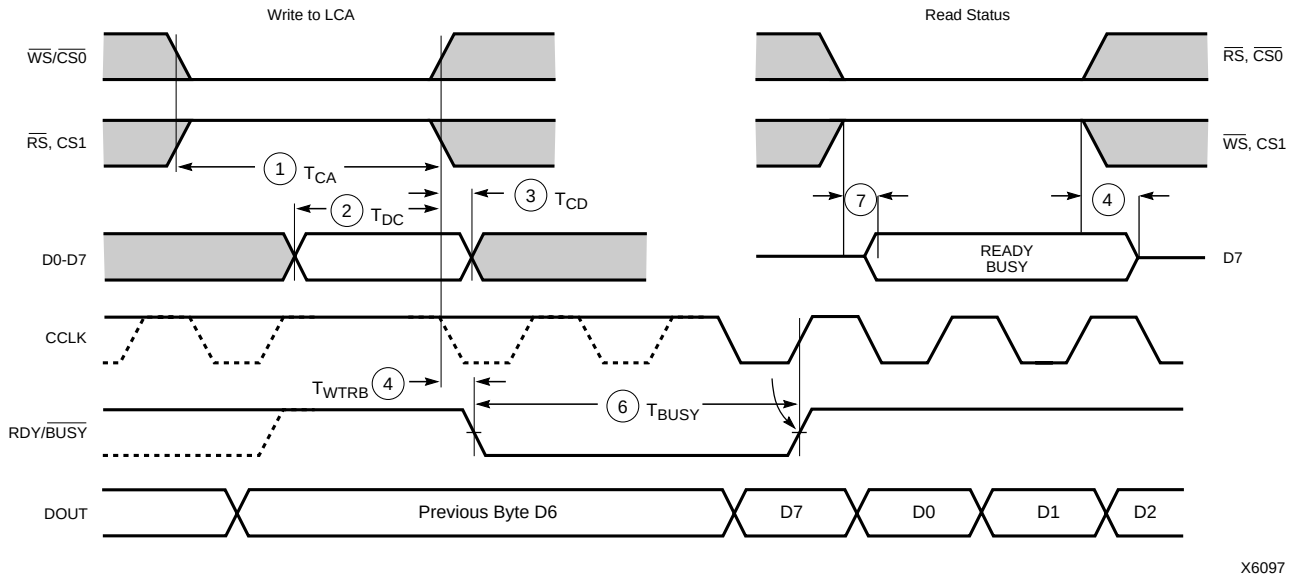
Figure 28: Master/Slave Serial Mode Circuit Diagram



	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 $\overline{INIT}$ pins of all daisy-chained FPGAs are High.

Figure 29: Slave Serial Mode Programming Switching Characteristics

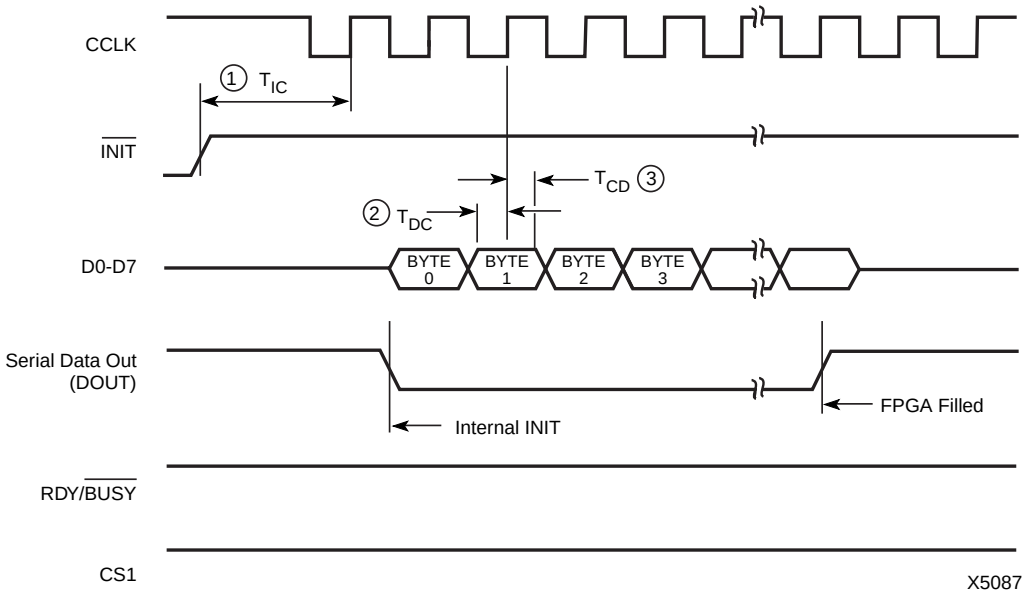


	Description	Symbol		Min	Max	Units
Write	Effective Write time ($\overline{CSO}$, $\overline{WS}$=Low; $\overline{RS}$, CS1=High	1	T_{CA}	100		ns
	DIN setup time	2	T_{DC}	60		ns
	DIN hold time	3	T_{CD}	0		ns
RDY	RDY/$\overline{BUSY}$ delay after end of Write or Read	4	T_{WTRB}		60	ns
	RDY/$\overline{BUSY}$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

Notes:
1. Configuration must be delayed until $\overline{INIT}$ 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 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/$\overline{BUSY}$ will go active within 60 ns after the end of $\overline{WS}$. A new write may be asserted immediately after RDY/$\overline{BUSY}$ goes Low, but write may not be terminated until RDY/$\overline{BUSY}$ has been High for one CCLK period.

Figure 36: Asynchronous Peripheral Mode Programming Switching Characteristics



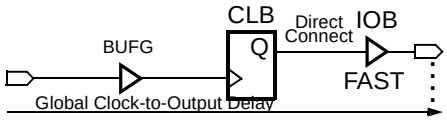
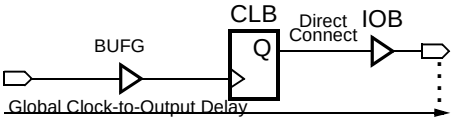
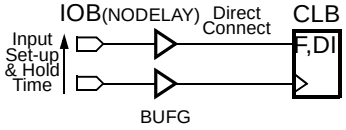
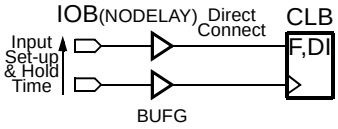
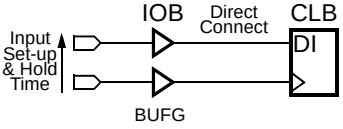
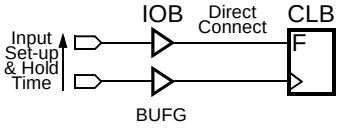
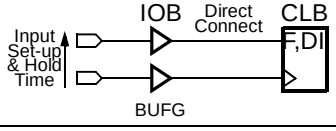
	Description	Symbol		Min	Max	Units
CCLK	INIT (High) Setup time required	1	T_{IC}	5		µs
	DIN Setup time required	2	T_{DC}	30		ns
	DIN hold time required	3	T_{CD}	0		ns
	CCLK High time		T_{CCH}	30		ns
	CCLK Low time		T_{CCL}	30		ns
	CCLK frequency		F_{CC}		10	MHz

Note: If not driven by the preceding DOUT, CS1 must remain high until the device is fully configured.

Figure 38: Express Mode Programming Switching Characteristics

XC5200 Guaranteed Input and Output Parameters (Pin-to-Pin)

All values listed below are tested directly, and guaranteed over the operating conditions. The same parameters can also be derived indirectly from the Global Buffer specifications. The delay calculator uses this indirect method, and may overestimate because of worst-case assumptions. When there is a discrepancy between these two methods, the values listed below should be used, and the derived values should be considered conservative overestimates.

		Speed Grade	-6	-5	-4	-3
Description	**Symbol**	**Device**	**Max (ns)**	**Max (ns)**	**Max (ns)**	**Max (ns)**
Global Clock to Output Pad (fast) 	T_{ICKOF} (Max)	XC5202	16.9	15.1	10.9	9.8
		XC5204	17.1	15.3	11.3	9.9
		XC5206	17.2	15.4	11.9	10.8
		XC5210	17.2	15.4	12.8	11.2
		XC5215	19.0	17.0	12.8	11.7
Global Clock to Output Pad (slew-limited) 	T_{ICKO} (Max)	XC5202	21.4	18.7	12.6	11.5
		XC5204	21.6	18.9	13.3	11.9
		XC5206	21.7	19.0	13.6	12.5
		XC5210	21.7	19.0	15.0	12.9
		XC5215	24.3	21.2	15.0	13.1
Input Set-up Time (no delay) to CLB Flip-Flop 	T_{PSUF} (Min)	XC5202	2.5	2.0	1.9	1.9
		XC5204	2.3	1.9	1.9	1.9
		XC5206	2.2	1.9	1.9	1.9
		XC5210	2.2	1.9	1.9	1.8
		XC5215	2.0	1.8	1.7	1.7
Input Hold Time (no delay) to CLB Flip-Flop 	T_{PHF} (Min)	XC5202	3.8	3.8	3.5	3.5
		XC5204	3.9	3.9	3.8	3.6
		XC5206	4.4	4.4	4.4	4.3
		XC5210	5.1	5.1	4.9	4.8
		XC5215	5.8	5.8	5.7	5.6
Input Set-up Time (with delay) to CLB Flip-Flop DI Input 	T_{PSU}	XC5202	7.3	6.6	6.6	6.6
		XC5204	7.3	6.6	6.6	6.6
		XC5206	7.2	6.5	6.4	6.3
		XC5210	7.2	6.5	6.0	6.0
		XC5215	6.8	5.7	5.7	5.7
Input Set-up Time (with delay) to CLB Flip-Flop F Input 	T_{PSUL} (Min)	XC5202	8.8	7.7	7.5	7.5
		XC5204	8.6	7.5	7.5	7.5
		XC5206	8.5	7.4	7.4	7.4
		XC5210	8.5	7.4	7.4	7.3
		XC5215	8.5	7.4	7.4	7.2
Input Hold Time (with delay) to CLB Flip-Flop 	T_{PH} (Min)	XC52xx	0	0	0	0

Note: 1. These measurements assume that the CLB flip-flop uses a direct interconnect to or from the IOB. The INREG/ OUTREG properties, or XACT-Performance, can be used to assure that direct connects are used. t_{PSU} applies only to the CLB input DI that bypasses the look-up table, which only offers direct connects to IOBs on the left and right edges of the die. t_{PSUL} applies to the CLB inputs F that feed the look-up table, which offers direct connect to IOBs on all four edges, as do the CLB Q outputs.

2. When testing outputs (fast or slew-limited), half of the outputs on one side of the device are switching.

Pin	Description		PC84	PQ100	VQ100	TQ144	PG156	PQ160	Boundary Scan Order
14.	I/O		-	-	-	141	D3	157	129
15.	I/O (A14)		9	1	98	142	B1	158	138
16.	I/O (A15)		10	2	99	143	B2	159	141
	VCC		11	3	100	144	C3	160	-
	GND		12	4	1	1	C4	1	-
17.	GCK1 (A16, I/O)		13	5	2	2	B3	2	150
18.	I/O (A17)		14	6	3	3	A1	3	153
19.	I/O		-	-	-	4	A2	4	159
20.	I/O		-	-	-	5	C5	5	162
21.	I/O (TDI)		15	7	4	6	B4	6	165
22.	I/O (TCK)		16	8	5	7	A3	7	171
	GND		-	-	-	8	C6	10	-
23.	I/O		-	-	-	9	B5	11	174
24.	I/O		-	-	-	10	B6	12	177
25.	I/O (TMS)		17	9	6	11	A5	13	180
26.	I/O		18	10	7	12	C7	14	183
27.	I/O		-	-	-	13	B7	15	186
28.	I/O		-	11	8	14	A6	16	189
29.	I/O		19	12	9	15	A7	17	195
30.	I/O		20	13	10	16	A8	18	198
	GND		21	14	11	17	C8	19	-
	VCC		22	15	12	18	B8	20	-
31.	I/O		23	16	13	19	C9	21	201
32.	I/O		24	17	14	20	B9	22	207
33.	I/O		-	18	15	21	A9	23	210
34.	I/O		-	-	-	22	B10	24	213
35.	I/O		25	19	16	23	C10	25	219
36.	I/O		26	20	17	24	A10	26	222
37.	I/O		-	-	-	25	A11	27	225
38.	I/O		-	-	-	26	B11	28	231
	GND		-	-	-	27	C11	29	-
39.	I/O		27	21	18	28	B12	32	234
40.	I/O		-	22	19	29	A13	33	237
41.	I/O		-	-	-	30	A14	34	240
42.	I/O		-	-	-	31	C12	35	243
43.	I/O		28	23	20	32	B13	36	246
44.	I/O		29	24	21	33	B14	37	249
45.	M1 (I/O)		30	25	22	34	A15	38	258
	GND		31	26	23	35	C13	39	-
46.	M0 (I/O)		32	27	24	36	A16	40	261
	VCC		33	28	25	37	C14	41	-
47.	M2 (I/O)		34	29	26	38	B15	42	264
48.	GCK2 (I/O)		35	30	27	39	B16	43	267
49.	I/O (HDC)		36	31	28	40	D14	44	276
50.	I/O		-	-	-	41	C15	45	279
51.	I/O		-	-	-	42	D15	46	282
52.	I/O		-	32	29	43	E14	47	288
53.	I/O (LDC)		37	33	30	44	C16	48	291
54.	I/O		-	-	-	-	E15	49	294
55.	I/O		-	-	-	-	D16	50	300
	GND		-	-	-	45	F14	51	-
56.	I/O		-	-	-	46	F15	52	303

Pin	Description		PC84	PQ100	VQ100	TQ144	PG156	PQ160	Boundary Scan Order
57.	I/O		-	-	-	47	E16	53	306
58.	I/O		38	34	31	48	F16	54	312
59.	I/O		39	35	32	49	G14	55	315
60.	I/O		-	36	33	50	G15	56	318
61.	I/O		-	37	34	51	G16	57	324
62.	I/O		40	38	35	52	H16	58	327
63.	I/O ($\overline{ERR}$, $\overline{INIT}$)		41	39	36	53	H15	59	330
	VCC		42	40	37	54	H14	60	-
	GND		43	41	38	55	J14	61	-
64.	I/O		44	42	39	56	J15	62	336
65.	I/O		45	43	40	57	J16	63	339
66.	I/O		-	44	41	58	K16	64	348
67.	I/O		-	45	42	59	K15	65	351
68.	I/O		46	46	43	60	K14	66	354
69.	I/O		47	47	44	61	L16	67	360
70.	I/O		-	-	-	62	M16	68	363
71.	I/O		-	-	-	63	L15	69	366
	GND		-	-	-	64	L14	70	-
72.	I/O		-	-	-	-	N16	71	372
73.	I/O		-	-	-	-	M15	72	375
74.	I/O		48	48	45	65	P16	73	378
75.	I/O		49	49	46	66	M14	74	384
76.	I/O		-	-	-	67	N15	75	387
77.	I/O		-	-	-	68	P15	76	390
78.	I/O		50	50	47	69	N14	77	396
79.	I/O		51	51	48	70	R16	78	399
	GND		52	52	49	71	P14	79	-
	DONE		53	53	50	72	R15	80	-
	VCC		54	54	51	73	P13	81	-
	PROG		55	55	52	74	R14	82	-
80.	I/O (D7)		56	56	53	75	T16	83	408
81.	GCK3 (I/O)		57	57	54	76	T15	84	411
82.	I/O		-	-	-	77	R13	85	420
83.	I/O		-	-	-	78	P12	86	423
84.	I/O (D6)		58	58	55	79	T14	87	426
85.	I/O		-	59	56	80	T13	88	432
	GND		-	-	-	81	P11	91	-
86.	I/O		-	-	-	82	R11	92	435
87.	I/O		-	-	-	83	T11	93	438
88.	I/O (D5)		59	60	57	84	T10	94	444
89.	I/O ($\overline{CS0}$)		60	61	58	85	P10	95	447
90.	I/O		-	62	59	86	R10	96	450
91.	I/O		-	63	60	87	T9	97	456
92.	I/O (D4)		61	64	61	88	R9	98	459
93.	I/O		62	65	62	89	P9	99	462
	VCC		63	66	63	90	R8	100	-
	GND		64	67	64	91	P8	101	-
94.	I/O (D3)		65	68	65	92	T8	102	468
95.	I/O ($\overline{RS}$)		66	69	66	93	T7	103	471
96.	I/O		-	70	67	94	T6	104	474
97.	I/O		-	-	-	95	R7	105	480
98.	I/O (D2)		67	71	68	96	P7	106	483

Pin Locations for XC5206 Devices

The following table may contain pinout information for unsupported device/package combinations. Please see the availability charts elsewhere in the XC5200 Series data sheet for availability information.

Pin	Description	PC84	PQ100	VQ100	TQ144	PQ160	TQ176	PG191	PQ208	Boundary Scan Order
	VCC	2	92	89	128	142	155	J4	183	-
1.	I/O (A8)	3	93	90	129	143	156	J3	184	87
2.	I/O (A9)	4	94	91	130	144	157	J2	185	90
3.	I/O	-	95	92	131	145	158	J1	186	93
4.	I/O	-	96	93	132	146	159	H1	187	99
5.	I/O	-	-	-	-	-	160	H2	188	102
6.	I/O	-	-	-	-	-	161	H3	189	105
7.	I/O (A10)	5	97	94	133	147	162	G1	190	111
8.	I/O (A11)	6	98	95	134	148	163	G2	191	114
9.	I/O	-	-	-	135	149	164	F1	192	117
10.	I/O	-	-	-	136	150	165	E1	193	123
	GND	-	-	-	137	151	166	G3	194	-
11.	I/O	-	-	-	-	152	168	C1	197	126
12.	I/O	-	-	-	-	153	169	E2	198	129
13.	I/O (A12)	7	99	96	138	154	170	F3	199	138
14.	I/O (A13)	8	100	97	139	155	171	D2	200	141
15.	I/O	-	-	-	140	156	172	B1	201	150
16.	I/O	-	-	-	141	157	173	E3	202	153
17.	I/O (A14)	9	1	98	142	158	174	C2	203	162
18.	I/O (A15)	10	2	99	143	159	175	B2	204	165
	VCC	11	3	100	144	160	176	D3	205	-
	GND	12	4	1	1	1	1	D4	2	-
19.	GCK1 (A16, I/O)	13	5	2	2	2	2	C3	4	174
20.	I/O (A17)	14	6	3	3	3	3	C4	5	177
21.	I/O	-	-	-	4	4	4	B3	6	183
22.	I/O	-	-	-	5	5	5	C5	7	186
23.	I/O (TDI)	15	7	4	6	6	6	A2	8	189
24.	I/O (TCK)	16	8	5	7	7	7	B4	9	195
25.	I/O	-	-	-	-	8	8	C6	10	198
26.	I/O	-	-	-	-	9	9	A3	11	201
	GND	-	-	-	8	10	10	C7	14	-
27.	I/O	-	-	-	9	11	11	A4	15	207
28.	I/O	-	-	-	10	12	12	A5	16	210
29.	I/O (TMS)	17	9	6	11	13	13	B7	17	213
30.	I/O	18	10	7	12	14	14	A6	18	219
31.	I/O	-	-	-	-	-	15	C8	19	222
32.	I/O	-	-	-	-	-	16	A7	20	225
33.	I/O	-	-	-	13	15	17	B8	21	234
34.	I/O	-	11	8	14	16	18	A8	22	237
35.	I/O	19	12	9	15	17	19	B9	23	246
36.	I/O	20	13	10	16	18	20	C9	24	249
	GND	21	14	11	17	19	21	D9	25	-
	VCC	22	15	12	18	20	22	D10	26	-
37.	I/O	23	16	13	19	21	23	C10	27	255
38.	I/O	24	17	14	20	22	24	B10	28	258
39.	I/O	-	18	15	21	23	25	A9	29	261
40.	I/O	-	-	-	22	24	26	A10	30	267
41.	I/O	-	-	-	-	-	27	A11	31	270

Pin	Description	PC84	PQ100	VQ100	TQ144	PQ160	TQ176	PG191	PQ208	Boundary Scan Order
87.	I/O	-	-	-	-	72	80	P17	94	468
88.	I/O	48	48	45	65	73	81	N16	95	471
89.	I/O	49	49	46	66	74	82	T17	96	480
90.	I/O	-	-	-	67	75	83	R17	97	483
91.	I/O	-	-	-	68	76	84	P16	98	486
92.	I/O	50	50	47	69	77	85	U18	99	492
93.	I/O	51	51	48	70	78	86	T16	100	495
	GND	52	52	49	71	79	87	R16	101	-
	DONE	53	53	50	72	80	88	U17	103	-
	VCC	54	54	51	73	81	89	R15	106	-
	PROG	55	55	52	74	82	90	V18	108	-
94.	I/O (D7)	56	56	53	75	83	91	T15	109	504
95.	GCK3 (I/O)	57	57	54	76	84	92	U16	110	507
96.	I/O	-	-	-	77	85	93	T14	111	516
97.	I/O	-	-	-	78	86	94	U15	112	519
98.	I/O (D6)	58	58	55	79	87	95	V17	113	522
99.	I/O	-	59	56	80	88	96	V16	114	528
100.	I/O	-	-	-	-	89	97	T13	115	531
101.	I/O	-	-	-	-	90	98	U14	116	534
	GND	-	-	-	81	91	99	T12	119	-
102.	I/O	-	-	-	82	92	100	U13	120	540
103.	I/O	-	-	-	83	93	101	V13	121	543
104.	I/O (D5)	59	60	57	84	94	102	U12	122	552
105.	I/O ($\overline{CS0}$)	60	61	58	85	95	103	V12	123	555
106.	I/O	-	-	-	-	-	104	T11	124	558
107.	I/O	-	-	-	-	-	105	U11	125	564
108.	I/O	-	62	59	86	96	106	V11	126	567
109.	I/O	-	63	60	87	97	107	V10	127	570
110.	I/O (D4)	61	64	61	88	98	108	U10	128	576
111.	I/O	62	65	62	89	99	109	T10	129	579
	VCC	63	66	63	90	100	110	R10	130	-
	GND	64	67	64	91	101	111	R9	131	-
112.	I/O (D3)	65	68	65	92	102	112	T9	132	588
113.	I/O ($\overline{RS}$)	66	69	66	93	103	113	U9	133	591
114.	I/O	-	70	67	94	104	114	V9	134	600
115.	I/O	-	-	-	95	105	115	V8	135	603
116.	I/O	-	-	-	-	-	116	U8	136	612
117.	I/O	-	-	-	-	-	117	T8	137	615
118.	I/O (D2)	67	71	68	96	106	118	V7	138	618
119.	I/O	68	72	69	97	107	119	U7	139	624
120.	I/O	-	-	-	98	108	120	V6	140	627
121.	I/O	-	-	-	99	109	121	U6	141	630
	GND	-	-	-	100	110	122	T7	142	-
122.	I/O	-	-	-	-	111	123	U5	145	636
123.	I/O	-	-	-	-	112	124	T6	146	639
124.	I/O (D1)	69	73	70	101	113	125	V3	147	642
125.	I/O ($\overline{RCLK-BUSY/RDY}$)	70	74	71	102	114	126	V2	148	648
126.	I/O	-	-	-	103	115	127	U4	149	651
127.	I/O	-	-	-	104	116	128	T5	150	654
128.	I/O (D0, DIN)	71	75	72	105	117	129	U3	151	660
129.	I/O (DOUT)	72	76	73	106	118	130	T4	152	663

Pin	Description	PC84	TQ144	PQ160	TQ176	PQ208	PG223	BG225	PQ240	Boundary Scan Order
95.	I/O	-	-	-	-	-	H15	P7	85	519
96.	I/O	-	50	56	62	74	H18	R7	86	522
97.	I/O	-	51	57	63	75	J18	L7	87	528
98.	I/O	40	52	58	64	76	J17	N8	88	531
99.	I/O ($\overline{\text{ERR}}$, $\overline{\text{INIT}}$)	41	53	59	65	77	J16	P8	89	534
	VCC	42	54	60	66	78	J15	VCC*	90	-
	GND	43	55	61	67	79	K15	GND*	91	-
100.	I/O	44	56	62	68	80	K16	L8	92	540
101.	I/O	45	57	63	69	81	K17	P9	93	543
102.	I/O	-	58	64	70	82	K18	R9	94	546
103.	I/O	-	59	65	71	83	L18	N9	95	552
104.	I/O	-	-	-	72	84	L17	M9	96	555
105.	I/O	-	-	-	73	85	L16	L9	97	558
106.	I/O	-	-	-	-	-	L15	R10	99	564
107.	I/O	-	-	-	-	-	M15	P10	100	567
	VCC	-	-	-	-	-	-	VCC*	101	-
108.	I/O	46	60	66	74	86	M18	N10	102	570
109.	I/O	47	61	67	75	87	M17	K9	103	576
110.	I/O	-	62	68	76	88	N18	R11	104	579
111.	I/O	-	63	69	77	89	P18	P11	105	588
	GND	-	64	70	78	90	M16	GND*	106	-
112.	I/O	-	-	-	-	-	N15	M10	107	591
113.	I/O	-	-	-	-	-	P15	N11	108	600
114.	I/O	-	-	-	-	91	N17	R12	109	603
115.	I/O	-	-	-	-	92	R18	L10	110	606
116.	I/O	-	-	71	79	93	T18	P12	111	612
117.	I/O	-	-	72	80	94	P17	M11	112	615
118.	I/O	48	65	73	81	95	N16	R13	113	618
119.	I/O	49	66	74	82	96	T17	N12	114	624
120.	I/O	-	67	75	83	97	R17	P13	115	627
121.	I/O	-	68	76	84	98	P16	K10	116	630
122.	I/O	50	69	77	85	99	U18	R14	117	636
123.	I/O	51	70	78	86	100	T16	N13	118	639
	GND	52	71	79	87	101	R16	GND*	119	-
	DONE	53	72	80	88	103	U17	P14	120	-
	VCC	54	73	81	89	106	R15	VCC*	121	-
	PROG	55	74	82	90	108	V18	M12	122	-
124.	I/O (D7)	56	75	83	91	109	T15	P15	123	648
125.	GCK3 (I/O)	57	76	84	92	110	U16	N14	124	651
126.	I/O	-	77	85	93	111	T14	L11	125	660
127.	I/O	-	78	86	94	112	U15	M13	126	663
128.	I/O	-	-	-	-	-	R14	N15	127	666
129.	I/O	-	-	-	-	-	R13	M14	128	672
130.	I/O (D6)	58	79	87	95	113	V17	J10	129	675
131.	I/O	-	80	88	96	114	V16	L12	130	678
132.	I/O	-	-	89	97	115	T13	M15	131	684
133.	I/O	-	-	90	98	116	U14	L13	132	687
134.	I/O	-	-	-	-	117	V15	L14	133	690
135.	I/O	-	-	-	-	118	V14	K11	134	696
	GND	-	81	91	99	119	T12	GND*	135	-
136.	I/O	-	-	-	-	-	R12	L15	136	699

Pin	Description	PC84	TQ144	PQ160	TQ176	PQ208	PG223	BG225	PQ240	Boundary Scan Order
137.	I/O	-	-	-	-	-	R11	K12	137	708
138.	I/O	-	82	92	100	120	U13	K13	138	711
139.	I/O	-	83	93	101	121	V13	K14	139	714
	VCC	-	-	-	-	-	-	VCC*	140	-
140.	I/O (D5)	59	84	94	102	122	U12	K15	141	720
141.	I/O ($\overline{CS0}$)	60	85	95	103	123	V12	J12	142	723
142.	I/O	-	-	-	104	124	T11	J13	144	726
143.	I/O	-	-	-	105	125	U11	J14	145	732
144.	I/O	-	86	96	106	126	V11	J15	146	735
145.	I/O	-	87	97	107	127	V10	J11	147	738
146.	I/O (D4)	61	88	98	108	128	U10	H13	148	744
147.	I/O	62	89	99	109	129	T10	H14	149	747
	VCC	63	90	100	110	130	R10	VCC*	150	-
	GND	64	91	101	111	131	R9	GND*	151	-
148.	I/O (D3)	65	92	102	112	132	T9	H12	152	756
149.	I/O ($\overline{RS}$)	66	93	103	113	133	U9	H11	153	759
150.	I/O	-	94	104	114	134	V9	G14	154	768
151.	I/O	-	95	105	115	135	V8	G15	155	771
152.	I/O	-	-	-	116	136	U8	G13	156	780
153.	I/O	-	-	-	117	137	T8	G12	157	783
154.	I/O (D2)	67	96	106	118	138	V7	G11	159	786
155.	I/O	68	97	107	119	139	U7	F15	160	792
	VCC	-	-	-	-	-	-	VCC*	161	-
156.	I/O	-	98	108	120	140	V6	F14	162	795
157.	I/O	-	99	109	121	141	U6	F13	163	798
158.	I/O	-	-	-	-	-	R8	G10	164	804
159.	I/O	-	-	-	-	-	R7	E15	165	807
	GND	-	100	110	122	142	T7	GND*	166	-
160.	I/O	-	-	-	-	-	R6	E14	167	810
161.	I/O	-	-	-	-	-	R5	F12	168	816
162.	I/O	-	-	-	-	143	V5	E13	169	819
163.	I/O	-	-	-	-	144	V4	D15	170	822
164.	I/O	-	-	111	123	145	U5	F11	171	828
165.	I/O	-	-	112	124	146	T6	D14	172	831
166.	I/O (D1)	69	101	113	125	147	V3	E12	173	834
167.	I/O ($\overline{RCLK}$-$\overline{BUSY}$/RDY)	70	102	114	126	148	V2	C15	174	840
168.	I/O	-	103	115	127	149	U4	D13	175	843
169.	I/O	-	104	116	128	150	T5	C14	176	846
170.	I/O (D0, DIN)	71	105	117	129	151	U3	F10	177	855
171.	I/O (DOUT)	72	106	118	130	152	T4	B15	178	858
	CCLK	73	107	119	131	153	V1	C13	179	-
	VCC	74	108	120	132	154	R4	VCC*	180	-
172.	I/O (TDO)	75	109	121	133	159	U2	A15	181	-
	GND	76	110	122	134	160	R3	GND*	182	-
173.	I/O (A0, $\overline{WS}$)	77	111	123	135	161	T3	A14	183	9
174.	GCK4 (A1, I/O)	78	112	124	136	162	U1	B13	184	15
175.	I/O	-	113	125	137	163	P3	E11	185	18
176.	I/O	-	114	126	138	164	R2	C12	186	21
177.	I/O (CS1, A2)	79	115	127	139	165	T2	A13	187	27
178.	I/O (A3)	80	116	128	140	166	N3	B12	188	30
179.	I/O	-	-	-	-	-	P4	F9	189	33

Pin	Description	PQ160	HQ208	HQ240	PG299	BG225	BG352	Boundary Scan Order
146.	I/O	-	-	-	R17	-	AD6	750
147.	I/O	-	-	-	T18	-	AC7	756
148.	I/O	73	95	113	U19	R13	AF4	759
149.	I/O	74	96	114	V19	N12	AF3	768
150.	I/O	75	97	115	R16	P13	AD5	771
151.	I/O	76	98	116	T17	K10	AE3	774
152.	I/O	77	99	117	U18	R14	AD4	780
153.	I/O	78	100	118	X20	N13	AC5	783
	GND	79	101	119	W20	GND*	GND*	-
	DONE	80	103	120	V18	P14	AD3	-
	VCC	81	106	121	X19	VCC*	VCC*	-
	PROG	82	108	122	U17	M12	AC4	-
154.	I/O (D7)	83	109	123	W19	P15	AD2	792
155.	GCK3 (I/O)	84	110	124	W18	N14	AC3	795
156.	I/O	85	111	125	T15	L11	AB4	804
157.	I/O	86	112	126	U16	M13	AD1	807
158.	I/O	-	-	127	V17	N15	AA4	810
159.	I/O	-	-	128	X18	M14	AA3	816
160.	I/O	-	-	-	U15	-	AB2	819
161.	I/O	-	-	-	T14	-	AC1	828
162.	I/O (D6)	87	113	129	W17	J10	Y3	831
163.	I/O	88	114	130	V16	L12	AA2	834
164.	I/O	89	115	131	X17	M15	AA1	840
165.	I/O	90	116	132	U14	L13	W4	843
166.	I/O	-	117	133	V15	L14	W3	846
167.	I/O	-	118	134	T13	K11	Y2	852
168.	I/O	-	-	-	W16	-	Y1	855
169.	I/O	-	-	-	W15	-	V4	858
	GND	91	119	135	X16	GND*	GND*	-
170.	I/O	-	-	136	U13	L15	V3	864
171.	I/O	-	-	137	V14	K12	W2	867
172.	I/O	92	120	138	W14	K13	U4	870
173.	I/O	93	121	139	V13	K14	U3	876
	VCC	-	-	140	X15	VCC*	VCC*	-
174.	I/O (D5)	94	122	141	T12	K15	V2	879
175.	I/O ($\overline{CS0}$)	95	123	142	X14	J12	V1	882
176.	I/O	-	-	-	X13	-	T1	888
177.	I/O	-	-	-	V12	-	R4	891
178.	I/O	-	124	144	W12	J13	R3	894
179.	I/O	-	125	145	T11	J14	R2	900
180.	I/O	96	126	146	X12	J15	R1	903
181.	I/O	97	127	147	U11	J11	P3	906
182.	I/O (D4)	98	128	148	V11	H13	P2	912
183.	I/O	99	129	149	W11	H14	P1	915
	VCC	100	130	150	X10	VCC*	VCC*	-
	GND	101	131	151	X11	GND*	GND*	-
184.	I/O (D3)	102	132	152	W10	H12	N2	924
185.	I/O ($\overline{RS}$)	103	133	153	V10	H11	N4	927
186.	I/O	104	134	154	T10	G14	N3	936
187.	I/O	105	135	155	U10	G15	M1	939
188.	I/O	-	136	156	X9	G13	M2	942
189.	I/O	-	137	157	W9	G12	M3	948