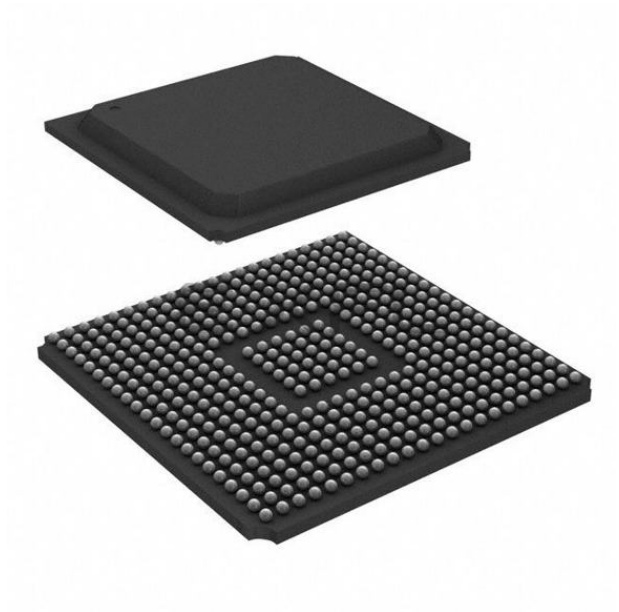
AMD Xilinx - XC3S1000-4FGG456I Datasheet



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	Active
Number of LABs/CLBs	1920
Number of Logic Elements/Cells	17280
Total RAM Bits	442368
Number of I/O	333
Number of Gates	1000000
Voltage - Supply	1.14V ~ 1.26V
Mounting Type	Surface Mount
Operating Temperature	-40°C ~ 100°C (TJ)
Package / Case	456-BBGA
Supplier Device Package	456-FBGA (23x23)
Purchase URL	https://www.e-xfl.com/product-detail/xilinx/xc3s1000-4fgg456i

Spartan-3 FPGA Family: Introduction and Ordering Information

DS099 (v3.0) October 29, 2012 **Product Specification**

Introduction

The Spartan®-3 family of Field-Programmable Gate Arrays is specifically designed to meet the needs of high volume, cost-sensitive consumer electronic applications. The eight-member family offers densities ranging from 50,000 to 5,000,000 system gates, as shown in Table 1.

The Spartan-3 family builds on the success of the earlier Spartan-IIE family by increasing the amount of logic resources, the capacity of internal RAM, the total number of I/Os, and the overall level of performance as well as by improving clock management functions. Numerous enhancements derive from the Virtex®-II platform technology. These Spartan-3 FPGA enhancements, combined with advanced process technology, deliver more functionality and bandwidth per dollar than was previously possible, setting new standards in the programmable logic industry.

Because of their exceptionally low cost, Spartan-3 FPGAs are ideally suited to a wide range of consumer electronics applications, including broadband access, home networking, display/projection and digital television equipment.

The Spartan-3 family is a superior alternative to mask programmed ASICs. FPGAs avoid the high initial cost, the lengthy development cycles, and the inherent inflexibility of conventional ASICs. Also, FPGA programmability permits design upgrades in the field with no hardware replacement necessary, an impossibility with ASICs.

Features

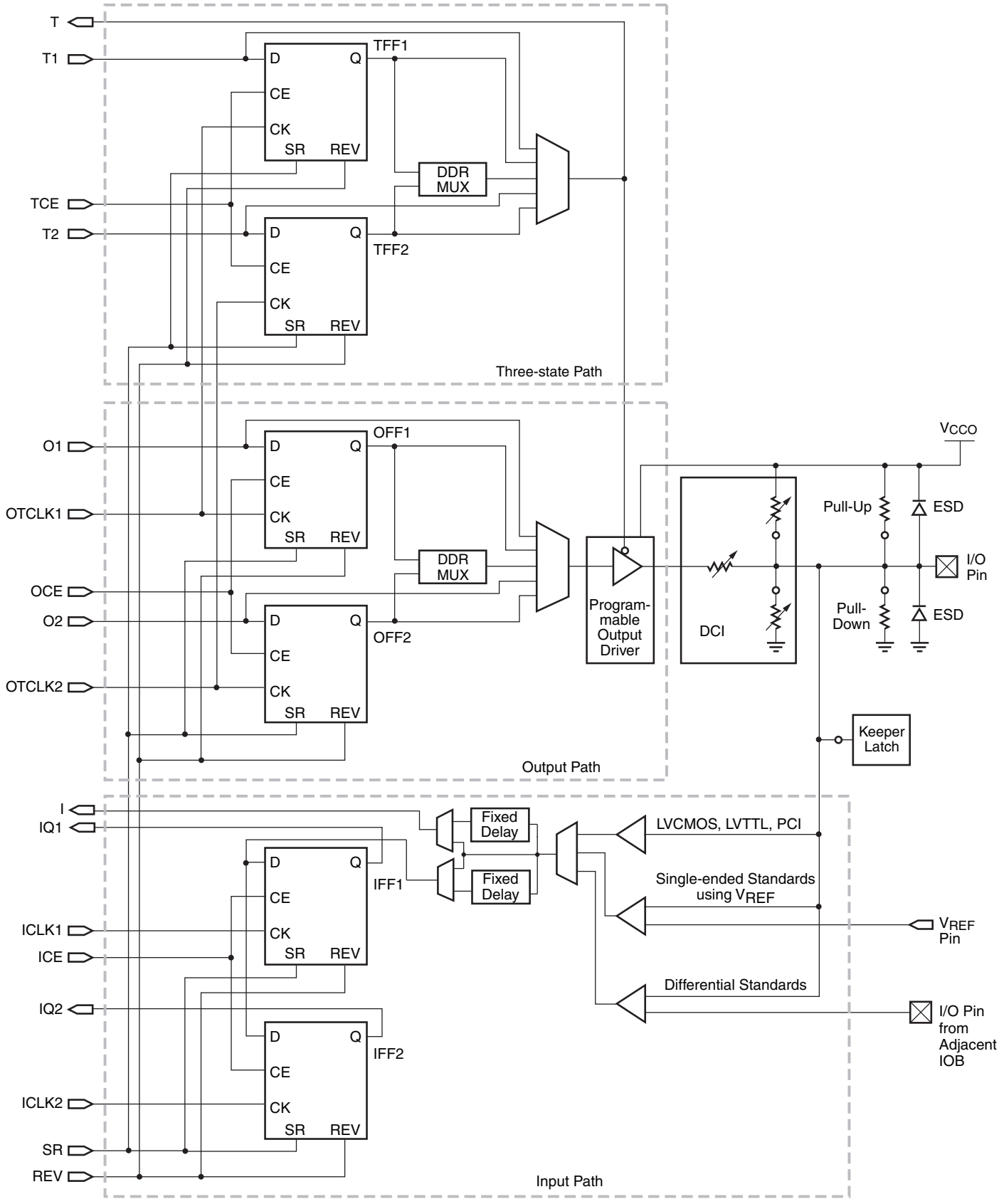
- Low-cost, high-performance logic solution for high-volume, consumer-oriented applications
  - Densities up to 74,880 logic cells
- SelectIO™ interface signaling
  - Up to 633 I/O pins
  - 622+ Mb/s data transfer rate per I/O
  - 18 single-ended signal standards
  - 8 differential I/O standards including LVDS, RSDS
  - Termination by Digitally Controlled Impedance
  - Signal swing ranging from 1.14V to 3.465V
  - Double Data Rate (DDR) support
  - DDR, DDR2 SDRAM support up to 333 Mb/s
- Logic resources
  - Abundant logic cells with shift register capability
  - Wide, fast multiplexers
  - Fast look-ahead carry logic
  - Dedicated 18 x 18 multipliers
  - JTAG logic compatible with IEEE 1149.1/1532
- SelectRAM™ hierarchical memory
  - Up to 1,872 Kbits of total block RAM
  - Up to 520 Kbits of total distributed RAM
- Digital Clock Manager (up to four DCMs)
  - Clock skew elimination
  - Frequency synthesis
  - High resolution phase shifting
- Eight global clock lines and abundant routing
- Fully supported by Xilinx ISE® and WebPACK™ software development systems
- MicroBlaze™ and PicoBlaze™ processor, PCI®, PCI Express® PIPE Endpoint, and other IP cores
- Pb-free packaging options
- Automotive Spartan-3 XA Family variant

Table 1: **Summary of Spartan-3 FPGA Attributes**

Device	System Gates	Equivalent Logic Cells[1]	CLB Array (One CLB = Four Slices)			Distributed RAM Bits (K=1024)	Block RAM Bits (K=1024)	Dedicated Multipliers	DCMs	Max. User I/O	Maximum Differential I/O Pairs
			Rows	Columns	Total CLBs						
XC3S50[2]	50K	1,728	16	12	192	12K	72K	4	2	124	56
XC3S200[2]	200K	4,320	24	20	480	30K	216K	12	4	173	76
XC3S400[2]	400K	8,064	32	28	896	56K	288K	16	4	264	116
XC3S1000[2]	1M	17,280	48	40	1,920	120K	432K	24	4	391	175
XC3S1500	1.5M	29,952	64	52	3,328	208K	576K	32	4	487	221
XC3S2000	2M	46,080	80	64	5,120	320K	720K	40	4	565	270
XC3S4000	4M	62,208	96	72	6,912	432K	1,728K	96	4	633	300
XC3S5000	5M	74,880	104	80	8,320	520K	1,872K	104	4	633	300

Notes:

1. Logic Cell = 4-input Look-Up Table (LUT) plus a 'D' flip-flop. "Equivalent Logic Cells" equals "Total CLBs" x 8 Logic Cells/CLB x 1.125 effectiveness.
2. These devices are available in Xilinx Automotive versions as described in DS314: *Spartan-3 Automotive XA FPGA Family.*



Note: All IOB signals originating from the FPGA's internal logic have an optional polarity inverter.

DS099-2_01_091410

Figure 7: **Simplified IOB Diagram**

The DCI feature operates independently for each of the device's eight banks. Each bank has an 'N' reference pin (VRN) and a 'P' reference pin, (VRP), to calibrate driver and termination resistance. Only when using a DCI standard on a given bank do these two pins function as VRN and VRP. When not using a DCI standard, the two pins function as user I/Os. As shown in Figure 9, add an external reference resistor to pull the VRN pin up to V_{CCO} and another reference resistor to pull the VRP pin down to GND. Also see Figure 42, page 116. Both resistors have the same value—commonly 50Ω—with one-percent tolerance, which is either the characteristic impedance of the line or twice that, depending on the DCI standard in use. Standards having a symbol name that contains the letters "DV2" use a reference resistor value that is twice the line impedance. DCI adjusts the output driver impedance to match the reference resistors' value or half that, according to the standard. DCI always adjusts the on-chip termination resistors to directly match the reference resistors' value.

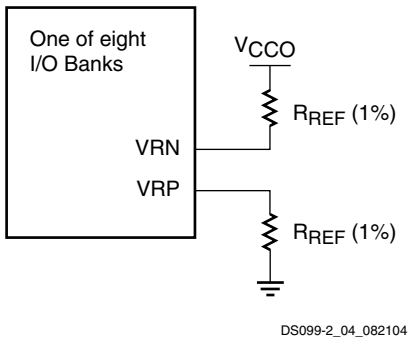


Figure 9: **Connection of Reference Resistors (R_{REF})**

The rules guiding the use of DCI standards on banks are as follows:

- No more than one DCI I/O standard with a Single Termination is allowed per bank.
- No more than one DCI I/O standard with a Split Termination is allowed per bank.
- Single Termination, Split Termination, Controlled- Impedance Driver, and Controlled-Impedance Driver with Half Impedance can co-exist in the same bank.

See also The Organization of IOBs into Banks, immediately below, and DCI: User I/O or Digitally Controlled Impedance Resistor Reference Input, page 115.

The Organization of IOBs into Banks

IOBs are allocated among eight banks, so that each side of the device has two banks, as shown in Figure 10. For all packages, each bank has independent V_{REF} lines. For example, V_{REF} Bank 3 lines are separate from the V_{REF} lines going to all other banks.

For the Very Thin Quad Flat Pack (VQ), Plastic Quad Flat Pack (PQ), Fine Pitch Thin Ball Grid Array (FT), and Fine Pitch Ball Grid Array (FG) packages, each bank has dedicated V_{CCO} lines. For example, the V_{CCO} Bank 7 lines are separate from the V_{CCO} lines going to all other banks. Thus, Spartan-3 devices in these packages support eight independent V_{CCO} supplies.

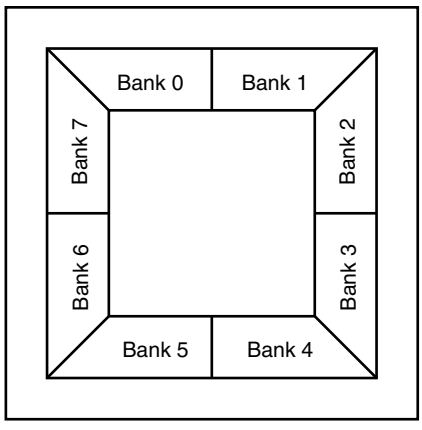


Figure 10: **Spartan-3 FPGA I/O Banks (Top View)**

Table 13: **Block RAM Port Signals *(Cont'd)***

Signal Description	Port A Signal Name	Port B Signal Name	Direction	Function
Data Output Bus	DOA	DOB	Output	Basic data access occurs whenever WE is inactive. The DO outputs mirror the data stored in the addressed memory location. Data access with WE asserted is also possible if one of the following two attributes is chosen: WRITE_FIRST and READ_FIRST. WRITE_FIRST simultaneously presents the new input data on the DO output port and writes the data to the address RAM location. READ_FIRST presents the previously stored RAM data on the DO output port while writing new data to RAM. A third attribute, NO_CHANGE, latches the DO outputs upon the assertion of WE. It is possible to configure a port's total data path width (w) to be 1, 2, 4, 9, 18, or 36 bits. This selection applies to both the DI and DO paths. See the DI signal description.
Parity Data Output(s)	DOPA	DOPB	Output	Parity inputs represent additional bits included in the data input path to support error detection. The number of parity bits "p" included in the DI (same as for the DO bus) depends on a port's total data path width (w). See Table 14.
Write Enable	WEA	WEB	Input	When asserted together with EN, this input enables the writing of data to the RAM. In this case, the data access attributes WRITE_FIRST, READ_FIRST or NO_CHANGE determines if and how data is updated on the DO outputs. See the DO signal description. When WE is inactive with EN asserted, read operations are still possible. In this case, a transparent latch passes data from the addressed memory location to the DO outputs.
Clock Enable	ENA	ENB	Input	When asserted, this input enables the CLK signal to synchronize Block RAM functions as follows: the writing of data to the DI inputs (when WE is also asserted), the updating of data at the DO outputs as well as the setting/resetting of the DO output latches. When de-asserted, the above functions are disabled.
Set/Reset	SSRA	SSRB	Input	When asserted, this pin forces the DO output latch to the value that the SRVAL attribute is set to. A Set/Reset operation on one port has no effect on the other ports functioning, nor does it disturb the memory's data contents. It is synchronized to the CLK signal.
Clock	CLKA	CLKB	Input	This input accepts the clock signal to which read and write operations are synchronized. All associated port inputs are required to meet setup times with respect to the clock signal's active edge. The data output bus responds after a clock-to-out delay referenced to the clock signal's active edge.

Port Aspect Ratios

On a given port, it is possible to select a number of different possible widths ($w - p$) for the DI/DO buses as shown in Table 14. These two buses always have the same width. This data bus width selection is independent for each port. If the data bus width of Port A differs from that of Port B, the Block RAM automatically performs a bus-matching function. When data are written to a port with a narrow bus, then read from a port with a wide bus, the latter port will effectively combine "narrow" words to form "wide" words. Similarly, when data are written into a port with a wide bus, then read from a port with a narrow bus, the latter port will divide "wide" words to form "narrow" words. When the data bus width is eight bits or greater, extra parity bits become available. The width of the total data path (w) is the sum of the DI/DO bus width and any parity bits (p).

The width selection made for the DI/DO bus determines the number of address lines according to the relationship expressed below:

$$r = 14 - [\log(w-p)/\log(2)] \qquad \textit{Equation 1}$$

In turn, the number of address lines delimits the total number (n) of addressable locations or depth according to the following equation:

$$n = 2^r \qquad \textit{Equation 2}$$

- **Phase Shifting:** The DCM provides the ability to shift the phase of all its output clock signals with respect to its input clock signal.

The DCM has four functional components: the Delay-Locked Loop (DLL), the Digital Frequency Synthesizer (DFS), the Phase Shifter (PS), and the Status Logic. Each component has its associated signals, as shown in Figure 19.

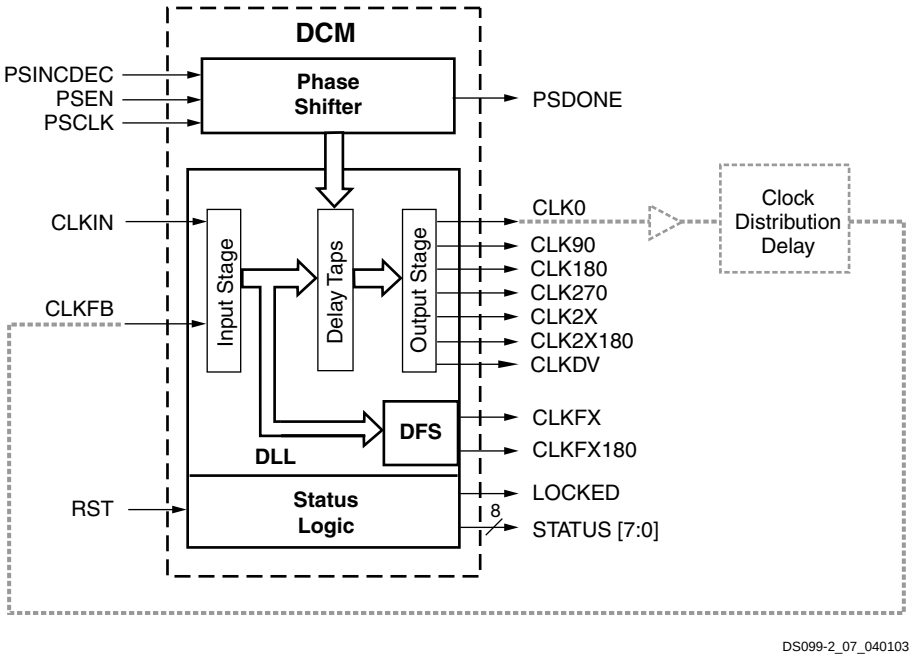


Figure 19: **DCM Functional Blocks and Associated Signals**

Delay-Locked Loop (DLL)

The most basic function of the DLL component is to eliminate clock skew. The main signal path of the DLL consists of an input stage, followed by a series of discrete delay elements or *taps,* which in turn leads to an output stage. This path together with logic for phase detection and control forms a system complete with feedback as shown in Figure 20.

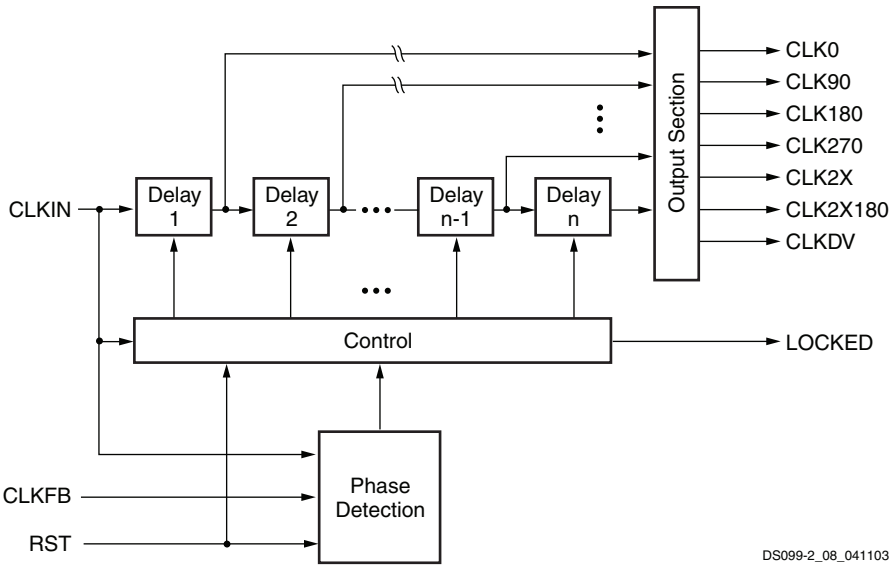


Figure 20: **Simplified Functional Diagram of DLL**

The output frequency (f_{CLKFX}) can be expressed as a function of the incoming clock frequency (f_{CLKIN}) as follows:

$$f_{CLKFX} = f_{CLKIN}(CLKFX\_MULTIPLY/CLKFX\_DIVIDE) \quad \textit{Equation 3}$$

Regarding the two attributes, it is possible to assign any combination of integer values, provided that two conditions are met:

- The two values fall within their corresponding ranges, as specified in Table 18.
- The f_{CLKFX} frequency calculated from the above expression accords with the DCM's operating frequency specifications.

For example, if CLKFX_MULTIPLY = 5 and CLKFX_DIVIDE = 3, then the frequency of the output clock signal would be 5/3 that of the input clock signal.

DFS Frequency Modes

The DFS supports two operating modes, High Frequency and Low Frequency, with each specified over a different clock frequency range. The DFS_FREQUENCY_MODE attribute chooses between the two modes. When the attribute is set to LOW, the Low Frequency mode permits the two DFS outputs to operate over a low-to-moderate frequency range. When the attribute is set to HIGH, the High Frequency mode allows both these outputs to operate at the highest possible frequencies.

DFS With or Without the DLL

The DFS component can be used with or without the DLL component:

Without the DLL, the DFS component multiplies or divides the CLKIN signal frequency according to the respective CLKFX_MULTIPLY and CLKFX_DIVIDE values, generating a clock with the new target frequency on the CLKFX and CLKFX180 outputs. Though classified as belonging to the DLL component, the CLKIN input is shared with the DFS component. This case does not employ feedback loop; therefore, it cannot correct for clock distribution delay.

With the DLL, the DFS operates as described in the preceding case, only with the additional benefit of eliminating the clock distribution delay. In this case, a feedback loop from the CLK0 output to the CLKFB input must be present.

The DLL and DFS components work together to achieve this phase correction as follows: Given values for the CLKFX_MULTIPLY and CLKFX_DIVIDE attributes, the DLL selects the delay element for which the output clock edge coincides with the input clock edge whenever mathematically possible. For example, when CLKFX_MULTIPLY = 5 and CLKFX_DIVIDE = 3, the input and output clock edges will coincide every three input periods, which is equivalent in time to five output periods.

Smaller CLKFX_MULTIPLY and CLKFX_DIVIDE values achieve faster lock times. With no factors common to the two attributes, alignment will occur once with every number of cycles equal to the CLKFX_DIVIDE value. Therefore, it is recommended that the user reduce these values by factoring wherever possible. For example, given CLKFX_MULTIPLY = 9 and CLKFX_DIVIDE = 6, removing a factor of three yields CLKFX_MULTIPLY = 3 and CLKFX_DIVIDE = 2. While both value-pairs will result in the multiplication of clock frequency by 3/2, the latter value-pair will enable the DLL to lock more quickly.

Table 18: **DFS Attributes**

Attribute	Description	Values
DFS_FREQUENCY_MODE	Chooses between High Frequency and Low Frequency modes	Low, High
CLKFX_MULTIPLY	Frequency multiplier constant	Integer from 2 to 32
CLKFX_DIVIDE	Frequency divisor constant	Integer from 1 to 32

Table 19: **DFS Signals**

Signal	Direction	Description
CLKFX	Output	Multiplies the CLKIN frequency by the attribute-value ratio (CLKFX_MULTIPLY/CLKFX_DIVIDE) to generate a clock signal with a new target frequency.
CLKFX180	Output	Generates a clock signal with same frequency as CLKFX, only shifted 180° out-of-phase.

DFS Clock Output Connections

There are two basic cases that determine how to connect the DFS clock outputs: on-chip and off-chip, which are illustrated in sections [a] and [c], respectively, of Figure 21. This is similar to what has already been described for the DLL component. See DLL Clock Output and Feedback Connections, page 34.

In the on-chip case, it is possible to connect either of the DFS's two output clock signals through general routing resources to the FPGA's internal registers. Either a Global Clock Buffer (BUFG) or a BUFGMUX affords access to the global clock network. The optional feedback loop is formed in this way, routing CLK0 to a global clock net, which in turn drives the CLKFB input.

In the off-chip case, the DFS's two output clock signals, plus CLK0 for an optional feedback loop, can exit the FPGA using output buffers (OBUF) to drive a clock network plus registers on the board. The feedback loop is formed by feeding the CLK0 signal back into the FPGA using an IBUFG, which directly accesses the global clock network, or an IBUF. Then, the global clock net is connected directly to the CLKFB input.

Phase Shifter (PS)

The DCM provides two approaches to controlling the phase of a DCM clock output signal relative to the CLKIN signal: First, there are nine clock outputs that employ the DLL to achieve a desired phase relationship: CLK0, CLK90, CLK180, CLK270, CLK2X, CLK2X180, CLKDV CLKFX, and CLKFX180. These outputs afford "coarse" phase control.

The second approach uses the PS component described in this section to provide a still finer degree of control. The PS component is only available when the DLL is operating in its low-frequency mode. The PS component phase shifts the DCM output clocks by introducing a "fine phase shift" (T_{PS}) between the CLKFB and CLKIN signals inside the DLL component. The user can control this fine phase shift down to a resolution of 1/256 of a CLKIN cycle or one tap delay (DCM_TAP), whichever is greater. When in use, the PS component shifts the phase of all nine DCM clock output signals together. If the PS component is used together with a DCM clock output such as the CLK90, CLK180, CLK270, CLK2X180 and CLKFX180, then the fine phase shift of the former gets added to the coarse phase shift of the latter.

PS Component Enabling and Mode Selection

The CLKOUT_PHASE_SHIFT attribute enables the PS component for use in addition to selecting between two operating modes. As described in Table 20, this attribute has three possible values: NONE, FIXED and VARIABLE. When CLKOUT_PHASE_SHIFT is set to NONE, the PS component is disabled and its inputs, PSEN, PSCLK, and PSINCDEC, must be tied to GND. The set of waveforms in section [a] of Figure 22 shows the disabled case, where the DLL maintains a zero-phase alignment of signals CLKFB and CLKIN upon which the PS component has no effect. The PS component is enabled by setting the attribute to either the FIXED or VARIABLE values, which select the Fixed Phase mode and the Variable Phase mode, respectively. These two modes are described in the sections that follow

Determining the Fine Phase Shift

The user controls the phase shift of CLKFB relative to CLKIN by setting and/or adjusting the value of the PHASE_SHIFT attribute. This value must be an integer ranging from –255 to +255. The PS component uses this value to calculate the desired fine phase shift (T_{PS}) as a fraction of the CLKIN period (T_{CLKIN}). Given values for PHASE-SHIFT and T_{CLKIN}, it is possible to calculate T_{PS} as follows:

$$T_{PS} = T_{CLKIN}(PHASE\_SHIFT/256) \qquad \textit{Equation 4}$$

Both the Fixed Phase and Variable Phase operating modes employ this calculation. If the PHASE_SHIFT value is zero, then CLKFB and CLKIN will be in phase, the same as when the PS component is disabled. When the PHASE_SHIFT value is positive, the CLKFB signal will be shifted later in time with respect to CLKIN. If the attribute value is negative, the CLKFB signal will be shifted earlier in time with respect to CLKIN.

The Fixed Phase Mode

This mode fixes the desired fine phase shift to a fraction of the T_{CLKIN}, as determined by Equation 4 and its user-selected PHASE_SHIFT value P. The set of waveforms insection [b] of Figure 22 illustrates the relationship between CLKFB and CLKIN in the Fixed Phase mode. In the Fixed Phase mode, the PSEN, PSCLK and PSINCDEC inputs are not used and must be tied to GND. Fixed phase shift requires ISE software version 10.1.03 or later.

Table 22: **Status Logic Signals**

Signal	Direction	Description
RST	Input	A High resets the entire DCM to its initial power-on state. Initializes the DLL taps for a delay of zero. Sets the LOCKED output Low. This input is asynchronous.
STATUS[7:0]	Output	The bit values on the STATUS bus provide information regarding the state of DLL and PS operation
LOCKED	Output	Indicates that the CLKIN and CLKFB signals are in phase by going High. The two signals are out-of-phase when Low.

Table 23: **DCM STATUS Bus**

Bit	Name	Description
0	Phase Shift Overflow	A value of 1 indicates a phase shift overflow when one of two conditions occurs: Incrementing (or decrementing) TPS beyond 255/256 of a CLKIN cycle. The DLL is producing its maximum possible phase shift (i.e., all delay taps are active).(1)
1	CLKIN Input Stopped Toggling	A value of 1 indicates that the CLKIN input signal is not toggling. A value of 0 indicates toggling. This bit functions only when the CLKFB input is connected.(2)
2	CLKFX/CLKFX180 Output Stopped Toggling	A value of 1 indicates that the CLKFX or CLKFX180 output signals are not toggling. A value of 0 indicates toggling. This bit functions only when using the Digital Frequency Synthesizer (DFS).
3:7	Reserved	–

Notes:

1. The DLL phase shift with all delay taps active is specified as the parameter FINE_SHIFT_RANGE.
2. If only the DFS clock outputs are used, but none of the DLL clock outputs, this bit will not go High when the CLKIN signal stops.

Table 24: **Status Attributes**

Attribute	Description	Values
STARTUP_WAIT	Delays transition from configuration to user mode until lock condition is achieved.	TRUE, FALSE

Stabilizing DCM Clocks Before User Mode

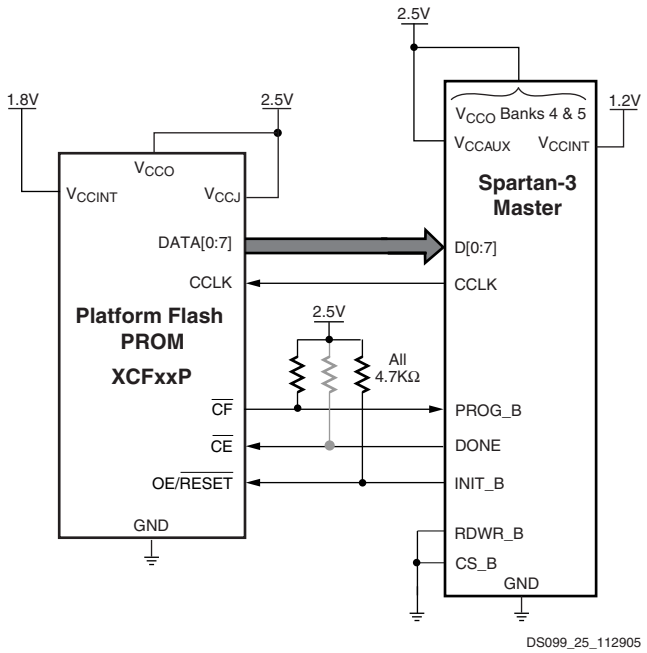
It is possible to delay the completion of device configuration until after the DLL has achieved a lock condition using the STARTUP_WAIT attribute described in Table 24. This option ensures that the FPGA does not enter user mode—i.e., begin functional operation—until all system clocks generated by the DCM are stable. In order to achieve the delay, it is necessary to set the attribute to TRUE as well as set the BitGen option LCK_cycle to one of the six cycles making up the Startup phase of configuration. The selected cycle defines the point at which configuration will halt until the LOCKED output goes High.

Global Clock Network

Spartan-3 devices have eight Global Clock inputs called GCLK0 - GCLK7. These inputs provide access to a low-capacitance, low-skew network that is well-suited to carrying high-frequency signals. The Spartan-3 FPGAs clock network is shown in Figure 23. GCLK0 through GCLK3 are located in the center of the bottom edge. GCLK4 through GCLK7 are located in the center of the top edge.

Eight Global Clock Multiplexers (also called BUFGMUX elements) are provided that accept signals from Global Clock inputs and route them to the internal clock network as well as DCMs. Four BUFGMUX elements are located in the center of the bottom edge, just above the GCLK0 - GCLK3 inputs. The remaining four BUFGMUX elements are located in the center of the top edge, just below the GCLK4 - GCLK7 inputs.

Pairs of BUFGMUX elements share global inputs, as shown in Figure 24. For example, the GCLK4 and GCLK5 inputs both potentially connect to BUFGMUX4 and BUFGMUX5 located in the upper right center. A differential clock input uses a pair of GCLK inputs to connect to a single BUFGMUX element.



Notes:

1. There are two ways to use the DONE line. First, one may set the BitGen option DriveDone to "Yes" only for the last FPGA to be configured in the chain shown above (or for the single FPGA as may be the case). This enables the DONE pin to drive High; thus, no pull-up resistor is necessary. DriveDone is set to "No" for the remaining FPGAs in the chain. Second, DriveDone can be set to "No" for all FPGAs. Then all DONE lines are open-drain and require the pull-up resistor shown in grey. In most cases, a value between 3.3KΩ to 4.7KΩ is sufficient. However, when using DONE synchronously with a long chain of FPGAs, cumulative capacitance may necessitate lower resistor values (e.g. down to 330Ω) in order to ensure a rise time within one clock cycle.

Figure 28: **Connection Diagram for Master Parallel Configuration**

Master Parallel Mode

In this mode, the FPGA configures from byte-wide data, and the FPGA supplies the CCLK configuration clock. In Master configuration modes, CCLK behaves as a bidirectional I/O pin. Timing is similar to the Slave Parallel mode except that CCLK is supplied by the FPGA. The device connections are shown in Figure 28.

Boundary-Scan (JTAG) Mode

In Boundary-Scan mode, dedicated pins are used for configuring the FPGA. The configuration is done entirely through the IEEE 1149.1 Test Access Port (TAP). FPGA configuration using the Boundary-Scan mode is compatible with the IEEE Std 1149.1-1993 standard and IEEE Std 1532 for In-System Configurable (ISC) devices.

Configuration through the boundary-scan port is always available, regardless of the selected configuration mode. In some cases, however, the mode pin setting may affect proper programming of the device due to various interactions. For example, if the mode pins are set to Master Serial or Master Parallel mode, and the associated PROM is already programmed with a valid configuration image, then there is potential for configuration interference between the JTAG and PROM data. Selecting the Boundary-Scan mode disables the other modes and is the most reliable mode when programming via JTAG.

Configuration Sequence

The configuration of Spartan-3 devices is a three-stage process that occurs after Power-On Reset or the assertion of PROG_B. POR occurs after the V_{CCINT}, V_{CCAUX}, and V_{CCO} Bank 4 supplies have reached their respective maximum input threshold levels (see Table 29, page 59). After POR, the three-stage process begins.

First, the configuration memory is cleared. Next, configuration data is loaded into the memory, and finally, the logic is activated by a start-up process. A flow diagram for the configuration sequence of the Serial and Parallel modes is shown in Figure 29. The flow diagram for the Boundary-Scan configuration sequence appears in Figure 30.

Table 46: **Timing for the IOB Three-State Path**

Symbol	Description	Conditions	Device	Speed Grade		Units
				-5	-4	
				Max[3]	Max[3]	
Synchronous Output Enable/Disable Times						
T_{IOCKHZ}	Time from the active transition at the OTCLK input of the Three-state Flip-Flop (TFF) to when the Output pin enters the high-impedance state	LVCMOS25, 12 mA output drive, Fast slew rate	All	0.74	0.85	ns
T_{IOCKON}[2]	Time from the active transition at TFF's OTCLK input to when the Output pin drives valid data		All	0.72	0.82	ns
Asynchronous Output Enable/Disable Times						
T_{GTS}	Time from asserting the Global Three State (GTS) net to when the Output pin enters the high-impedance state	LVCMOS25, 12 mA output drive, Fast slew rate	XC3S200 XC3S400	7.71	8.87	ns
			XC3S50 XC3S1000 XC3S1500 XC3S2000 XC3S4000 XC3S5000	8.38	9.63	ns
Set/Reset Times						
T_{IOSRHZ}	Time from asserting TFF's SR input to when the Output pin enters a high-impedance state	LVCMOS25, 12 mA output drive, Fast slew rate	All	1.55	1.78	ns
T_{IOSRON}[2]	Time from asserting TFF's SR input at TFF to when the Output pin drives valid data		XC3S200 XC3S400	2.24	2.57	ns
			XC3S50 XC3S1000 XC3S1500 XC3S2000 XC3S4000 XC3S5000	2.91	3.34	ns

Notes:

1. The numbers in this table are tested using the methodology presented in Table 48 and are based on the operating conditions set forth in Table 32 and Table 35.
2. This time requires adjustment whenever a signal standard other than LVCMOS25 with 12 mA drive and Fast slew rate is assigned to the data Output. When this is true, *add* the appropriate Output adjustment from Table 47.
3. For minimums, use the values reported by the Xilinx timing analyzer.

Table 47: **Output Timing Adjustments for IOB**

Convert Output Time from LVCMOS25 with 12mA Drive and Fast Slew Rate to the Following Signal Standard (IOSTANDARD)	Add the Adjustment Below		Units
	Speed Grade		
	-5	-4	
Single-Ended Standards			
GTL	0	0.02	ns
GTL_DCI	0.13	0.15	ns
GTLP	0.03	0.04	ns
GTLP_DCI	0.23	0.27	ns
HSLVDCI_15	1.51	1.74	ns
HSLVDCI_18	0.81	0.94	ns

Table 48: **Test Methods for Timing Measurement at I/Os *(Cont'd)***

Signal Standard (IOSTANDARD)	Inputs			Outputs		Inputs and Outputs
	V_{REF} (V)	V_L (V)	V_H (V)	R_T (Ω)	V_T (V)	V_M (V)
DIFF_SSTL2_II	-	V_{ICM} – 0.75	V_{ICM} + 0.75	50	1.25	V_{ICM}
DIFF_SSTL2_II_DCI						

Notes:

1. Descriptions of the relevant symbols are as follows:
   VREF – The reference voltage for setting the input switching threshold
   VICM – The common mode input voltage
   VM – Voltage of measurement point on signal transition
   VL – Low-level test voltage at Input pin
   VH – High-level test voltage at Input pin
   RT – Effective termination resistance, which takes on a value of 1MW when no parallel termination is required
   VT – Termination voltage
2. The load capacitance (CL) at the Output pin is 0 pF for all signal standards.
3. According to the PCI specification.

The capacitive load (C_L) is connected between the output and GND. *The Output timing for all standards, as published in the speed files and the data sheet, is always based on a C_L value of zero.* High-impedance probes (less than 1 pF) are used for all measurements. Any delay that the test fixture might contribute to test measurements is subtracted from those measurements to produce the final timing numbers as published in the speed files and data sheet.

Using IBIS Models to Simulate Load Conditions in Application

IBIS Models permit the most accurate prediction of timing delays for a given application. The parameters found in the IBIS model (V_{REF}, R_{REF}, and V_{MEAS}) correspond directly with the parameters used in Table 48, V_T, R_T, and V_M. Do not confuse V_{REF} (the termination voltage) from the IBIS model with V_{REF} (the input-switching threshold) from the table. A fourth parameter, C_{REF}, is always zero. The four parameters describe all relevant output test conditions. IBIS models are found in the Xilinx development software as well as at the following link.

http://www.xilinx.com/support/download/index.htm

Simulate delays for a given application according to its specific load conditions as follows:

1. Simulate the desired signal standard with the output driver connected to the test setup shown in Figure 35. Use parameter values V_T, R_T, and V_M from Table 48. C_{REF} is zero.
2. Record the time to V_M.
3. Simulate the same signal standard with the output driver connected to the PCB trace with load. Use the appropriate IBIS model (including V_{REF}, R_{REF}, C_{REF}, and V_{MEAS} values) or capacitive value to represent the load.
4. Record the time to V_{MEAS}.
5. Compare the results of steps 2 and 4. The increase (or decrease) in delay should be added to (or subtracted from) the appropriate Output standard adjustment (Table 47) to yield the worst-case delay of the PCB trace.

Table 59: **Switching Characteristics for the DLL**

Symbol	Description	Frequency Mode / FCLKIN Range	Device	Speed Grade				Units
				-5		-4		
				Min	Max	Min	Max	
Output Frequency Ranges								
CLKOUT_FREQ_1X_LF	Frequency for the CLK0, CLK90, CLK180, and CLK270 outputs	Low	All	18	167	18	167	MHz
CLKOUT_FREQ_1X_HF	Frequency for the CLK0 and CLK180 outputs	High		48	280	48	280	MHz
CLKOUT_FREQ_2X_LF(3)	Frequency for the CLK2X and CLK2X180 outputs	Low		36	334	36	334	MHz
CLKOUT_FREQ_DV_LF	Frequency for the CLKDV output	Low		1.125	110	1.125	110	MHz
CLKOUT_FREQ_DV_HF		High		3	185	3	185	MHz
Output Clock Jitter(4)								
CLKOUT_PER_JITT_0	Period jitter at the CLK0 output	All	All	–	±100	–	±100	ps
CLKOUT_PER_JITT_90	Period jitter at the CLK90 output			–	±150	–	±150	ps
CLKOUT_PER_JITT_180	Period jitter at the CLK180 output			–	±150	–	±150	ps
CLKOUT_PER_JITT_270	Period jitter at the CLK270 output			–	±150	–	±150	ps
CLKOUT_PER_JITT_2X	Period jitter at the CLK2X and CLK2X180 outputs			–	±200	–	±200	ps
CLKOUT_PER_JITT_DV1	Period jitter at the CLKDV output when performing integer division			–	±150	–	±150	ps
CLKOUT_PER_JITT_DV2	Period jitter at the CLKDV output when performing non-integer division			–	±300	–	±300	ps
Duty Cycle								
CLKOUT_DUTY_CYCLE_DLL(5)	Duty cycle variation for the CLK0, CLK90, CLK180, CLK270, CLK2X, CLK2X180, and CLKDV outputs	All	XC3S50	–	±150	–	±150	ps
			XC3S200	–	±150	–	±150	ps
			XC3S400	–	±250	–	±250	ps
			XC3S1000	–	±400	–	±400	ps
			XC3S1500	–	±400	–	±400	ps
			XC3S2000	–	±400	–	±400	ps
			XC3S4000	–	±400	–	±400	ps
			XC3S5000	–	±400	–	±400	ps
Phase Alignment								
CLKIN_CLKFB_PHASE	Phase offset between the CLKIN and CLKFB inputs	All	All	–	±150	–	±150	ps
CLKOUT_PHASE	Phase offset between any two DLL outputs (except CLK2X and CLK0)			–	±140	–	±140	ps
	Phase offset between the CLK2X and CLK0 outputs			–	±250	–	±250	ps

Table 80: **Bitstream Options Affecting Spartan-3 Device Pins *(Cont'd)***

Affected Pin Name(s)	Bitstream Generation Function	Option Variable Name	Values (Default)
CCLK	After configuration, this bitstream option either pulls CCLK to VCCAUX via a pull-up resistor, or allows CCLK to float.	CclkPin	• **Pullup** • Pullnone
CCLK	For Master configuration modes, this option sets the approximate frequency, in MHz, for the internal silicon oscillator.	ConfigRate	• 3, **6**, 12, 25, 50
PROG_B	A pull-up resistor to VCCAUX exists on PROG_B during configuration. After configuration, this bitstream option either pulls PROG_B to VCCAUX via a pull-up resistor, or allows PROG_B to float.	ProgPin	• **Pullup** • Pullnone
DONE	After configuration, this bitstream option either pulls DONE to VCCAUX via a pull-up resistor, or allows DONE to float. See also DriveDone option.	DonePin	• **Pullup** • Pullnone
DONE	If set to Yes, this option allows the FPGA's DONE pin to drive High when configuration completes. By default, the DONE is an open-drain output and can only drive Low. Only single FPGAs and the last FPGA in a multi-FPGA daisy-chain should use this option.	DriveDone	• **No** • Yes
M2	After configuration, this bitstream option either pulls M2 to VCCAUX via a pull-up resistor, to ground via a pull-down resistor, or allows M2 to float.	M2Pin	• **Pullup** • Pulldown • Pullnone
M1	After configuration, this bitstream option either pulls M1 to VCCAUX via a pull-up resistor, to ground via a pull-down resistor, or allows M1 to float.	M1Pin	• **Pullup** • Pulldown • Pullnone
M0	After configuration, this bitstream option either pulls M0 to VCCAUX via a pull-up resistor, to ground via a pull-down resistor, or allows M0 to float.	M0Pin	• **Pullup** • Pulldown • Pullnone
HSWAP_EN	After configuration, this bitstream option either pulls HSWAP_EN to VCCAUX via a pull-up resistor, to ground via a pull-down resistor, or allows HSWAP_EN to float.	HswapenPin	• **Pullup** • Pulldown • Pullnone
TDI	After configuration, this bitstream option either pulls TDI to VCCAUX via a pull-up resistor, to ground via a pull-down resistor, or allows TDI to float.	TdiPin	• **Pullup** • Pulldown • Pullnone
TMS	After configuration, this bitstream option either pulls TMS to VCCAUX via a pull-up resistor, to ground via a pull-down resistor, or allows TMS to float.	TmsPin	• **Pullup** • Pulldown • Pullnone
TCK	After configuration, this bitstream option either pulls TCK to VCCAUX via a pull-up resistor, to ground via a pull-down resistor, or allows TCK to float.	TckPin	• **Pullup** • Pulldown • Pullnone
TDO	After configuration, this bitstream option either pulls TDO to VCCAUX via a pull-up resistor, to ground via a pull-down resistor, or allows TDO to float.	TdoPin	• **Pullup** • Pulldown • Pullnone

Setting Bitstream Generator Options

Refer to the "BitGen" chapter in the Xilinx ISE® software documentation.

User I/Os by Bank

Table 97 indicates how the available user-I/O pins are distributed between the eight I/O banks on the FT256 package.

Table 97: **User I/Os Per Bank in FT256 Package**

Package Edge	I/O Bank	Maximum I/O	All Possible I/O Pins by Type				
			I/O	DUAL	DCI	VREF	GCLK
Top	0	20	13	0	2	3	2
	1	20	13	0	2	3	2
Right	2	23	18	0	2	3	0
	3	23	18	0	2	3	0
Bottom	4	21	8	6	2	3	2
	5	20	7	6	2	3	2
Left	6	23	18	0	2	3	0
	7	23	18	0	2	3	0

Table 100: **FG456 Package Pinout *(Cont'd)***

Bank	3S400 Pin Name	3S1000, 3S1500, 3S2000 Pin Name	FG456 Pin Number	Type
7	IO_L23N_7	IO_L23N_7	F2	I/O
7	IO_L23P_7	IO_L23P_7	F3	I/O
7	IO_L24N_7	IO_L24N_7	H5	I/O
7	IO_L24P_7	IO_L24P_7	G5	I/O
7	N.C. (◆)	IO_L26N_7	G3	I/O
7	N.C. (◆)	IO_L26P_7	G4	I/O
7	IO_L27N_7	IO_L27N_7	G1	I/O
7	IO_L27P_7/VREF_7	IO_L27P_7/VREF_7	G2	VREF
7	N.C. (◆)	IO_L28N_7	H1	I/O
7	N.C. (◆)	IO_L28P_7	H2	I/O
7	N.C. (◆)	IO_L29N_7	J4	I/O
7	N.C. (◆)	IO_L29P_7	H4	I/O
7	N.C. (◆)	IO_L31N_7	J5	I/O
7	N.C. (◆)	IO_L31P_7	J6	I/O
7	N.C. (◆)	IO_L32N_7	J1	I/O
7	N.C. (◆)	IO_L32P_7	J2	I/O
7	N.C. (◆)	IO_L33N_7	K5	I/O
7	N.C. (◆)	IO_L33P_7	K6	I/O
7	IO_L34N_7	IO_L34N_7	K3	I/O
7	IO_L34P_7	IO_L34P_7	K4	I/O
7	IO_L35N_7	IO_L35N_7	K1	I/O
7	IO_L35P_7	IO_L35P_7	K2	I/O
7	IO_L38N_7	IO_L38N_7	L5	I/O
7	IO_L38P_7	IO_L38P_7	L6	I/O
7	IO_L39N_7	IO_L39N_7	L3	I/O
7	IO_L39P_7	IO_L39P_7	L4	I/O
7	IO_L40N_7/VREF_7	IO_L40N_7/VREF_7	L1	VREF
7	IO_L40P_7	IO_L40P_7	L2	I/O
7	VCCO_7	VCCO_7	H3	VCCO
7	VCCO_7	VCCO_7	H6	VCCO
7	VCCO_7	VCCO_7	J7	VCCO
7	VCCO_7	VCCO_7	K7	VCCO
7	VCCO_7	VCCO_7	L7	VCCO
N/A	GND	GND	A1	GND
N/A	GND	GND	A22	GND
N/A	GND	GND	AA2	GND
N/A	GND	GND	AA21	GND
N/A	GND	GND	AB1	GND
N/A	GND	GND	AB22	GND
N/A	GND	GND	B2	GND

Table 103: **FG676 Package Pinout *(Cont'd)***

Bank	XC3S1000 Pin Name	XC3S1500 Pin Name	XC3S2000 Pin Name	XC3S4000 Pin Name	XC3S5000 Pin Name	FG676 Pin Number	Type
0	VCCO_0	VCCO_0	VCCO_0	VCCO_0	VCCO_0	H9	VCCO
0	VCCO_0	VCCO_0	VCCO_0	VCCO_0	VCCO_0	H10	VCCO
0	VCCO_0	VCCO_0	VCCO_0	VCCO_0	VCCO_0	J11	VCCO
0	VCCO_0	VCCO_0	VCCO_0	VCCO_0	VCCO_0	J12	VCCO
0	VCCO_0	VCCO_0	VCCO_0	VCCO_0	VCCO_0	J13	VCCO
0	VCCO_0	VCCO_0	VCCO_0	VCCO_0	VCCO_0	K13	VCCO
1	IO	IO	IO	IO	IO	A14	I/O
1	IO	IO	IO	IO	IO	A22	I/O
1	IO	IO	IO	IO	IO	A23	I/O
1	IO	IO	IO	IO	IO	D16	I/O
1	IO	IO	IO	IO	IO_L17P_1(3)	E18	I/O
1	IO	IO	IO	IO	IO	F14	I/O
1	IO	IO	IO	IO	IO	F20	I/O
1	IO	IO	IO	IO	IO	G19	I/O
1	IO/VREF_1	IO/VREF_1	IO/VREF_1	IO/VREF_1	IO/VREF_1	C15	VREF
1	IO/VREF_1	IO/VREF_1	IO/VREF_1	IO/VREF_1	IO/VREF_1	C17	VREF
1	N.C. (◆)	IO/VREF_1	IO/VREF_1	IO/VREF_1	IO_L17N_1/VREF_1(3)	D18	VREF
1	IO_L01N_1/VRP_1	IO_L01N_1/VRP_1	IO_L01N_1/VRP_1	IO_L01N_1/VRP_1	IO_L01N_1/VRP_1	D22	DCI
1	IO_L01P_1/VRN_1	IO_L01P_1/VRN_1	IO_L01P_1/VRN_1	IO_L01P_1/VRN_1	IO_L01P_1/VRN_1	E22	DCI
1	IO_L04N_1	IO_L04N_1	IO_L04N_1	IO_L04N_1	IO_L04N_1	B23	I/O
1	IO_L04P_1	IO_L04P_1	IO_L04P_1	IO_L04P_1	IO_L04P_1	C23	I/O
1	IO_L05N_1	IO_L05N_1	IO_L05N_1	IO_L05N_1	IO_L05N_1	E21	I/O
1	IO_L05P_1	IO_L05P_1	IO_L05P_1	IO_L05P_1	IO_L05P_1	F21	I/O
1	IO_L06N_1/VREF_1	IO_L06N_1/VREF_1	IO_L06N_1/VREF_1	IO_L06N_1/VREF_1	IO_L06N_1/VREF_1	B22	VREF
1	IO_L06P_1	IO_L06P_1	IO_L06P_1	IO_L06P_1	IO_L06P_1	C22	I/O
1	IO_L07N_1	IO_L07N_1	IO_L07N_1	IO_L07N_1	IO_L07N_1	C21	I/O
1	IO_L07P_1	IO_L07P_1	IO_L07P_1	IO_L07P_1	IO_L07P_1	D21	I/O
1	IO_L08N_1	IO_L08N_1	IO_L08N_1	IO_L08N_1	IO_L08N_1	A21	I/O
1	IO_L08P_1	IO_L08P_1	IO_L08P_1	IO_L08P_1	IO_L08P_1	B21	I/O
1	IO_L09N_1	IO_L09N_1	IO_L09N_1	IO_L09N_1	IO_L09N_1	D20	I/O
1	IO_L09P_1	IO_L09P_1	IO_L09P_1	IO_L09P_1	IO_L09P_1	E20	I/O
1	IO_L10N_1/VREF_1	IO_L10N_1/VREF_1	IO_L10N_1/VREF_1	IO_L10N_1/VREF_1	IO_L10N_1/VREF_1	A20	VREF
1	IO_L10P_1	IO_L10P_1	IO_L10P_1	IO_L10P_1	IO_L10P_1	B20	I/O
1	N.C. (◆)	IO_L11N_1	IO_L11N_1	IO_L11N_1	IO_L11N_1	E19	I/O
1	N.C. (◆)	IO_L11P_1	IO_L11P_1	IO_L11P_1	IO_L11P_1	F19	I/O
1	N.C. (◆)	IO_L12N_1	IO_L12N_1	IO_L12N_1	IO_L12N_1	C19	I/O
1	N.C. (◆)	IO_L12P_1	IO_L12P_1	IO_L12P_1	IO_L12P_1	D19	I/O
1	IO_L15N_1	IO_L15N_1	IO_L15N_1	IO_L15N_1	IO_L15N_1	A19	I/O
1	IO_L15P_1	IO_L15P_1	IO_L15P_1	IO_L15P_1	IO_L15P_1	B19	I/O
1	IO_L16N_1	IO_L16N_1	IO_L16N_1	IO_L16N_1	IO_L16N_1	F18	I/O
1	IO_L16P_1	IO_L16P_1	IO_L16P_1	IO_L16P_1	IO_L16P_1	G18	I/O
1	N.C. (◆)	IO_L18N_1	IO_L18N_1	IO_L18N_1	IO(3)	B18	I/O

Table 103: **FG676 Package Pinout *(Cont'd)***

Bank	XC3S1000 Pin Name	XC3S1500 Pin Name	XC3S2000 Pin Name	XC3S4000 Pin Name	XC3S5000 Pin Name	FG676 Pin Number	Type
3	IO_L19N_3	IO_L19N_3	IO_L19N_3	IO_L19N_3	IO_L19N_3	W26	I/O
3	IO_L19P_3	IO_L19P_3	IO_L19P_3	IO_L19P_3	IO_L19P_3	W25	I/O
3	IO_L20N_3	IO_L20N_3	IO_L20N_3	IO_L20N_3	IO_L20N_3	U20	I/O
3	IO_L20P_3	IO_L20P_3	IO_L20P_3	IO_L20P_3	IO_L20P_3	V20	I/O
3	IO_L21N_3	IO_L21N_3	IO_L21N_3	IO_L21N_3	IO_L21N_3	V23	I/O
3	IO_L21P_3	IO_L21P_3	IO_L21P_3	IO_L21P_3	IO_L21P_3	V22	I/O
3	IO_L22N_3	IO_L22N_3	IO_L22N_3	IO_L22N_3	IO_L22N_3	V25	I/O
3	IO_L22P_3	IO_L22P_3	IO_L22P_3	IO_L22P_3	IO_L22P_3	V24	I/O
3	IO_L23N_3	IO_L23N_3	IO_L23N_3	IO_L23N_3	IO_L23N_3	U22	I/O
3	IO_L23P_3/VREF_3	IO_L23P_3/VREF_3	IO_L23P_3/VREF_3	IO_L23P_3/VREF_3	IO_L23P_3/VREF_3	U21	VREF
3	IO_L24N_3	IO_L24N_3	IO_L24N_3	IO_L24N_3	IO_L24N_3	U24	I/O
3	IO_L24P_3	IO_L24P_3	IO_L24P_3	IO_L24P_3	IO_L24P_3	U23	I/O
3	IO_L26N_3	IO_L26N_3	IO_L26N_3	IO_L26N_3	IO_L26N_3	U26	I/O
3	IO_L26P_3	IO_L26P_3	IO_L26P_3	IO_L26P_3	IO_L26P_3	U25	I/O
3	IO_L27N_3	IO_L27N_3	IO_L27N_3	IO_L27N_3	IO_L27N_3	T20	I/O
3	IO_L27P_3	IO_L27P_3	IO_L27P_3	IO_L27P_3	IO_L27P_3	T19	I/O
3	IO_L28N_3	IO_L28N_3	IO_L28N_3	IO_L28N_3	IO_L28N_3	T22	I/O
3	IO_L28P_3	IO_L28P_3	IO_L28P_3	IO_L28P_3	IO_L28P_3	T21	I/O
3	IO_L29N_3	IO_L29N_3	IO_L29N_3	IO_L29N_3	IO_L29N_3	T26	I/O
3	IO_L29P_3	IO_L29P_3	IO_L29P_3	IO_L29P_3	IO_L29P_3	T25	I/O
3	IO_L31N_3	IO_L31N_3	IO_L31N_3	IO_L31N_3	IO_L31N_3	R20	I/O
3	IO_L31P_3	IO_L31P_3	IO_L31P_3	IO_L31P_3	IO_L31P_3	R19	I/O
3	IO_L32N_3	IO_L32N_3	IO_L32N_3	IO_L32N_3	IO_L32N_3	R22	I/O
3	IO_L32P_3	IO_L32P_3	IO_L32P_3	IO_L32P_3	IO_L32P_3	R21	I/O
3	IO_L33N_3	IO_L33N_3	IO_L33N_3	IO_L33N_3	IO_L33N_3	R24	I/O
3	IO_L33P_3	IO_L33P_3	IO_L33P_3	IO_L33P_3	IO_L33P_3	T23	I/O
3	IO_L34N_3	IO_L34N_3	IO_L34N_3	IO_L34N_3	IO_L34N_3	R26	I/O
3	IO_L34P_3/VREF_3	IO_L34P_3/VREF_3	IO_L34P_3/VREF_3	IO_L34P_3/VREF_3	IO_L34P_3/VREF_3	R25	VREF
3	IO_L35N_3	IO_L35N_3	IO_L35N_3	IO_L35N_3	IO_L35N_3	P20	I/O
3	IO_L35P_3	IO_L35P_3	IO_L35P_3	IO_L35P_3	IO_L35P_3	P19	I/O
3	IO_L38N_3	IO_L38N_3	IO_L38N_3	IO_L38N_3	IO_L38N_3	P22	I/O
3	IO_L38P_3	IO_L38P_3	IO_L38P_3	IO_L38P_3	IO_L38P_3	P21	I/O
3	IO_L39N_3	IO_L39N_3	IO_L39N_3	IO_L39N_3	IO_L39N_3	P24	I/O
3	IO_L39P_3	IO_L39P_3	IO_L39P_3	IO_L39P_3	IO_L39P_3	P23	I/O
3	IO_L40N_3/VREF_3	IO_L40N_3/VREF_3	IO_L40N_3/VREF_3	IO_L40N_3/VREF_3	IO_L40N_3/VREF_3	P26	VREF
3	IO_L40P_3	IO_L40P_3	IO_L40P_3	IO_L40P_3	IO_L40P_3	P25	I/O
3	VCCO_3	VCCO_3	VCCO_3	VCCO_3	VCCO_3	P17	VCCO
3	VCCO_3	VCCO_3	VCCO_3	VCCO_3	VCCO_3	P18	VCCO
3	VCCO_3	VCCO_3	VCCO_3	VCCO_3	VCCO_3	R18	VCCO
3	VCCO_3	VCCO_3	VCCO_3	VCCO_3	VCCO_3	T18	VCCO
3	VCCO_3	VCCO_3	VCCO_3	VCCO_3	VCCO_3	T24	VCCO
3	VCCO_3	VCCO_3	VCCO_3	VCCO_3	VCCO_3	U19	VCCO
3	VCCO_3	VCCO_3	VCCO_3	VCCO_3	VCCO_3	V19	VCCO

Table 107: **FG900 Package Pinout *(Cont'd)***

Bank	XC3S2000 Pin Name	XC3S4000, XC3S5000 Pin Name	FG900 Pin Number	Type
7	IO_L01N_7/VRP_7	IO_L01N_7/VRP_7	C1	DCI
7	IO_L01P_7/VRN_7	IO_L01P_7/VRN_7	C2	DCI
7	IO_L02N_7	IO_L02N_7	D3	I/O
7	IO_L02P_7	IO_L02P_7	D4	I/O
7	IO_L03N_7/VREF_7	IO_L03N_7/VREF_7	D1	VREF
7	IO_L03P_7	IO_L03P_7	D2	I/O
7	IO_L04N_7	IO_L04N_7	E1	I/O
7	IO_L04P_7	IO_L04P_7	E2	I/O
7	IO_L05N_7	IO_L05N_7	F5	I/O
7	IO_L05P_7	IO_L05P_7	E4	I/O
7	IO_L06N_7	IO_L06N_7	F2	I/O
7	IO_L06P_7	IO_L06P_7	F3	I/O
7	IO_L07N_7	IO_L07N_7	G3	I/O
7	IO_L07P_7	IO_L07P_7	G4	I/O
7	IO_L08N_7	IO_L08N_7	G1	I/O
7	IO_L08P_7	IO_L08P_7	G2	I/O
7	IO_L09N_7	IO_L09N_7	H7	I/O
7	IO_L09P_7	IO_L09P_7	G6	I/O
7	IO_L10N_7	IO_L10N_7	H5	I/O
7	IO_L10P_7/VREF_7	IO_L10P_7/VREF_7	H6	VREF
7	IO_L11N_7	IO_L11N_7	H3	I/O
7	IO_L11P_7	IO_L11P_7	H4	I/O
7	IO_L13N_7	IO_L13N_7	H1	I/O
7	IO_L13P_7	IO_L13P_7	H2	I/O
7	IO_L14N_7	IO_L14N_7	J4	I/O
7	IO_L14P_7	IO_L14P_7	J5	I/O
7	IO_L15N_7	IO_L15N_7	J1	I/O
7	IO_L15P_7	IO_L15P_7	J2	I/O
7	IO_L16N_7	IO_L16N_7	K9	I/O
7	IO_L16P_7/VREF_7	IO_L16P_7/VREF_7	J8	VREF
7	IO_L17N_7	IO_L17N_7	K6	I/O
7	IO_L17P_7	IO_L17P_7	K7	I/O
7	IO_L19N_7/VREF_7	IO_L19N_7/VREF_7	K2	VREF
7	IO_L19P_7	IO_L19P_7	K3	I/O
7	IO_L20N_7	IO_L20N_7	L10	I/O
7	IO_L20P_7	IO_L20P_7	K10	I/O
7	IO_L21N_7	IO_L21N_7	L7	I/O
7	IO_L21P_7	IO_L21P_7	L8	I/O
7	IO_L22N_7	IO_L22N_7	L5	I/O
7	IO_L22P_7	IO_L22P_7	L6	I/O

Table 107: **FG900 Package Pinout *(Cont'd)***

Bank	XC3S2000 Pin Name	XC3S4000, XC3S5000 Pin Name	FG900 Pin Number	Type
7	IO_L23N_7	IO_L23N_7	L3	I/O
7	IO_L23P_7	IO_L23P_7	L4	I/O
7	IO_L24N_7	IO_L24N_7	L1	I/O
7	IO_L24P_7	IO_L24P_7	L2	I/O
7	N.C. (◆)	IO_L25N_7	M6	I/O
7	N.C. (◆)	IO_L25P_7	M7	I/O
7	IO_L26N_7	IO_L26N_7	M3	I/O
7	IO_L26P_7	IO_L26P_7	M4	I/O
7	IO_L27N_7	IO_L27N_7	M1	I/O
7	IO_L27P_7/VREF_7	IO_L27P_7/VREF_7	M2	VREF
7	IO_L28N_7	IO_L28N_7	N10	I/O
7	IO_L28P_7	IO_L28P_7	M10	I/O
7	IO_L29N_7	IO_L29N_7	N8	I/O
7	IO_L29P_7	IO_L29P_7	N9	I/O
7	IO_L31N_7	IO_L31N_7	N1	I/O
7	IO_L31P_7	IO_L31P_7	N2	I/O
7	IO_L32N_7	IO_L32N_7	P9	I/O
7	IO_L32P_7	IO_L32P_7	P10	I/O
7	IO_L33N_7	IO_L33N_7	P6	I/O
7	IO_L33P_7	IO_L33P_7	P7	I/O
7	IO_L34N_7	IO_L34N_7	P2	I/O
7	IO_L34P_7	IO_L34P_7	P3	I/O
7	IO_L35N_7	IO_L35N_7	R9	I/O
7	IO_L35P_7	IO_L35P_7	R10	I/O
7	IO_L37N_7	IO_L37N_7	R7	I/O
7	IO_L37P_7/VREF_7	IO_L37P_7/VREF_7	R8	VREF
7	IO_L38N_7	IO_L38N_7	R5	I/O
7	IO_L38P_7	IO_L38P_7	R6	I/O
7	IO_L39N_7	IO_L39N_7	R3	I/O
7	IO_L39P_7	IO_L39P_7	R4	I/O
7	IO_L40N_7/VREF_7	IO_L40N_7/VREF_7	R1	VREF
7	IO_L40P_7	IO_L40P_7	R2	I/O
7	N.C. (◆)	IO_L46N_7	M8	I/O
7	N.C. (◆)	IO_L46P_7	M9	I/O
7	N.C. (◆)	IO_L49N_7	N6	I/O
7	N.C. (◆)	IO_L49P_7	M5	I/O
7	N.C. (◆)	IO_L50N_7	N4	I/O
7	N.C. (◆)	IO_L50P_7	N5	I/O
7	VCCO_7	VCCO_7	E3	VCCO
7	VCCO_7	VCCO_7	J3	VCCO

Table 110: **FG1156 Package Pinout *(Cont'd)***

Bank	XC3S4000 Pin Name	XC3S5000 Pin Name	FG1156 Pin Number	Type
5	IO_L04P_5	IO_L04P_5	AL6	I/O
5	IO_L05N_5	IO_L05N_5	AP6	I/O
5	IO_L05P_5	IO_L05P_5	AN6	I/O
5	IO_L06N_5	IO_L06N_5	AK7	I/O
5	IO_L06P_5	IO_L06P_5	AJ7	I/O
5	IO_L07N_5	IO_L07N_5	AG10	I/O
5	IO_L07P_5	IO_L07P_5	AF10	I/O
5	IO_L08N_5	IO_L08N_5	AJ10	I/O
5	IO_L08P_5	IO_L08P_5	AH10	I/O
5	IO_L09N_5	IO_L09N_5	AM10	I/O
5	IO_L09P_5	IO_L09P_5	AL10	I/O
5	IO_L10N_5/VRP_5	IO_L10N_5/VRP_5	AP10	DCI
5	IO_L10P_5/VRN_5	IO_L10P_5/VRN_5	AN10	DCI
5	IO_L11N_5/VREF_5	IO_L11N_5/VREF_5	AP11	VREF
5	IO_L11P_5	IO_L11P_5	AN11	I/O
5	IO_L12N_5	IO_L12N_5	AF12	I/O
5	IO_L12P_5	IO_L12P_5	AE12	I/O
5	IO_L13N_5	IO_L13N_5	AJ12	I/O
5	IO_L13P_5	IO_L13P_5	AH12	I/O
5	IO_L14N_5	IO_L14N_5	AL12	I/O
5	IO_L14P_5	IO_L14P_5	AK12	I/O
5	IO_L15N_5	IO_L15N_5	AP12	I/O
5	IO_L15P_5	IO_L15P_5	AN12	I/O
5	IO_L16N_5	IO_L16N_5	AE13	I/O
5	IO_L16P_5	IO_L16P_5	AD13	I/O
5	IO_L17N_5	IO_L17N_5	AH13	I/O
5	IO_L17P_5	IO_L17P_5	AG13	I/O
5	IO_L18N_5	IO_L18N_5	AM13	I/O
5	IO_L18P_5	IO_L18P_5	AL13	I/O
5	IO_L19N_5	IO_L19N_5	AG14	I/O
5	IO_L19P_5/VREF_5	IO_L19P_5/VREF_5	AF14	VREF
5	IO_L20N_5	IO_L20N_5	AJ14	I/O
5	IO_L20P_5	IO_L20P_5	AH14	I/O
5	IO_L21N_5	IO_L21N_5	AM14	I/O
5	IO_L21P_5	IO_L21P_5	AL14	I/O
5	IO_L22N_5	IO_L22N_5	AP14	I/O
5	IO_L22P_5	IO_L22P_5	AN14	I/O
5	IO_L23N_5	IO_L23N_5	AF15	I/O
5	IO_L23P_5	IO_L23P_5	AE15	I/O
5	IO_L24N_5	IO_L24N_5	AJ15	I/O