Preliminary Data Sheet
December 2000

microelectronics group

Lucent Technologies
Bell Labs Innovations

ORCA® Series 4
Field-Programmable Gate Arrays

Programmable Features

- High-performance platform design.
  - — 0.13 µm seven-level metal technology.
  - — Internal performance of >250 MHz (four logic levels).
  - — I/O performance of >416 MHz for all user I/Os.
  - — Over 1.5 million usable system gates.
  - — Meets multiple I/O interface standards.
  - — 1.5 V operation (30% less power than 1.8 V operation) translates to greater performance.
  - — Embedded block RAM (EBR) for onboard storage and buffer needs.
  - — Built-in system components including an internal system bus, eight PLLs, and microprocessor interface.
- Traditional I/O selections.
  - — LVTTL and LVCMOS (3.3 V, 2.5 V, and 1.8 V) I/Os.
  - — Per pin-selectable I/O clamping diodes provide 3.3 V PCI compliance.
  - — Individually programmable drive capability. 24 mA sink/12 mA source, 12 mA sink/6 mA source, or 6 mA sink/3 mA source.
  - — Two slew rates supported (fast and slew-limited).
  - — Fast-capture input latch and input flip-flop (FF)/latch for reduced input setup time and zero hold time.
  - — Fast open-drain drive capability.
  - — Capability to register 3-state enable signal.
  - — Off-chip clock drive capability.
  - — Two-input function generator in output path.
- New programmable high-speed I/O.
  - — Single-ended: GTL, GTL+, PECL, SSTL3/2 (class I & II), HSTL (Class I, III, IV), zero-bus turn-around (*ZBT**), and double data rate (DDR).
  - — Double-ended: LDVS, bused-LVDS, LVPECL.
  - — Customer defined: Ability to substitute arbitrary standard-cell I/O to meet fast moving standards.
- New capability to (de)multiplex I/O signals.
  - — New DDR on both input and output at rates up to 311 MHz (622 MHz effective rate).
  - — Used to implement emerging *RapidIO*† backplane interface specification.
  - — New 2x and 4x downlink and uplink capability per I/O (i.e., 104 MHz internal to 416 MHz I/O).
- Enhanced twin-quad programmable function unit (PFU).
  - — Eight 16-bit look-up tables (LUTs) per PFU.
  - — Nine user registers per PFU, one following each LUT and organized to allow two nibbles to act independently, plus one extra for arithmetic carry/borrow operations.

* *ZBT* is a trademark of Integrated Device Technologies Inc.
† *RapidIO* is a trademark of Motorola, Inc.

Table 1. *ORCA* Series 4—Available FPGA Logic

Device	Columns	Rows	PFUs	User I/O	LUTs	EBR Blocks	EBR Bits (k)	Usable† Gates (k)
OR4E2	26	24	624	400	4992	8	74	260—470
OR4E4	36	36	1296	576	10368	12	111	400—720
OR4E6	46	44	2024	720	16,192	16	148	530—970
OR4E10	60	56	3360	928	26,880	20	185	740—1350
OR4E14	70	66	4620	1088	36,960	24	222	930—1700

† The usable gate counts range from a logic-only gate count to a gate count assuming 20% of the PFUs/SLICs being used as RAMs. The logic-only gate count includes each PFU/SLIC (counted as 108 gates/PFU), including 12 gates per LUT/FF pair (eight per PFU), and 12 gates per SLIC/FF pair (one per PFU). Each of the four PIO groups are counted as 16 gates (three FFs, fast-capture latch, output logic, CLK, and I/O buffers). PFUs used as RAM are counted at four gates per bit, with each PFU capable of implementing a 32 x 4 RAM (or 512 gates) per PFU. Embedded block RAM (EBR) is counted as four gates per bit plus each block has an additional 25k gates. 7k gates are used for each PLL and 50k gates for the embedded system bus and microprocessor interface logic. Both the EBR and PLLs are conservatively utilized in the gate count calculations.

Note: Devices are not pinout compatible with *ORCA* Series 2/3.

Table of Contents

Table of Contents (continued)

Programmable Features (continued)

- — New register control in each PFU has two independent programmable clocks, clock enables, local set/reset, and data selects.
- — New LUT structure allows flexible combinations of LUT4, LUT5, new LUT6, 4-to-1 MUX, new 8-to-1 MUX, and ripple mode arithmetic functions in the same PFU.
- — 32 x 4 RAM per PFU, configurable as single- or dual-port. Create large, fast RAM/ROM blocks (128 x 8 in only eight PFUs) using the supplemental logic and interconnect cell (SLIC) decoders as bank drivers.
- — Softwired LUTs (SWL) allow fast cascading of up to three levels of LUT logic in a single PFU through fast internal routing which reduces routing congestion and improves speed.
- — Flexible fast access to PFU inputs from routing.
- — Fast-carry logic and routing to all four adjacent PFUs for nibble-, bytewide, or longer arithmetic functions, with the option to register the PFU carry-out.

- Abundant high-speed buffered and nonbuffered routing resources provide 2x average speed improvements over previous architectures.
- Hierarchical routing optimized for both local and global routing with dedicated routing resources. This results in faster routing times with predictable and efficient performance.
- SLIC provides eight 3-statable buffers, up to 10-bit decoder, and *PAL*[1]-like and-or-invert (AOI) in each programmable logic cell.
- New 200 MHz embedded quad-port RAM blocks, two read ports, two write ports, and two sets of byte lane enables. Each embedded RAM block can be configured as:
  - — One 512 x 18 (quad-port, two read/two write) with optional built-in arbitration.
  - — One 256 x 36 (dual-port, one read/one write).
  - — One 1K x 9 (dual-port, one read/one write).
  - — Two 512 x 9 (dual-port, one read/one write for each).
  - — Two RAMs with arbitrary number of words whose sum is 512 or less by 18 (dual-port, one read/one write).
  - — Supports joining of RAM blocks.
  - — Two 16 x 8-bit content addressable memory (CAM) support.
  - — FIFO 512 x 18, 256 x 36, 1K x 9 or dual 512 x 9.
  - — Constant multiply (8 x 16 or 16 x 8).
  - — Dual-variable multiply (8 x 8).
- Built-in testability.
  - — Full boundary-scan (*IEEE*[2] 1149.1 and Draft 1149.2 joint test access group (JTAG)).
  - — Programming and readback through boundary-scan port compliant to *IEEE* Draft 1532:D1.7.
  - — TS_ALL testability function to 3-state all I/O pins.
  - — New temperature-sensing diode used to determine device junction temperature.

System Features

- PCI local bus compliant.
- Improved *PowerPC*[3]860 and *PowerPC* II high-speed (66 MHz) synchronous MPI interface can be used for configuration, readback, device control, and device status, as well as for a general-purpose interface to the FPGA logic, RAMs, and embedded standard-cell blocks. Glueless interface to synchronous *PowerPC* processors with user-configurable address space provided.
- New embedded *AMBA*[4] specification 2.0 AHB system bus (*ARM*[4]processor) facilitates communication among the microprocessor interface, configuration logic, EBR, FPGA logic, and embedded standard-cell blocks. Embedded 32-bit internal system bus plus 4-bit parity interconnects FPGA logic, microprocessor interface (MPI), embedded RAM blocks, and embedded standard-cell blocks with 100 MHz bus performance. Included are built-in system registers that act as the control and status center for the device.
- New network phase-locked loops (PLLs) meet ITU-T G.811 specifications and provide clock conditioning for DS-1/E-1 and STS-3/STM-1 applications.
- Flexible general-purpose programmable PLLs offer clock multiply (up to 8x), divide (down to 1/8x), phase shift, delay compensation, and duty cycle adjustment combined. Improved built-in clock management with programmable phase-locked loops (PPLLs) provide optimum clock modification and conditioning for phase, frequency, and duty cycle from 20 MHz up to 420 MHz. Each PPLL provides two separate clock outputs.

1. *PAL* is a trademark of Advanced Micro Devices, Inc.
2. *IEEE* a is registered trademark of The Institute of Electrical and Electronics Engineers, Inc.
3. *PowerPC* is a registered trademark of International Business Machines, Corporation.
4. *AMBA* and *ARM* are trademarks of ARM Limited.

System Features (continued)

- Variable-size bused readback of configuration data capability with the built-in MPI and system bus.
- Internal, 3-state, bidirectional buses with simple control provided by the SLIC.
- Meets universal test and operations PHY interface for ATM (UTOPIA) Levels 1, 2, and 3. Also meets proposed specifications for UTOPIA Level 4 for 10 Gbits/s interfaces.
- New clock routing structures for global and local clocking significantly increases speed and reduces skew (<200 ps for OR4E4).
- New local clock routing structures allow creation of localized clock trees anywhere on the device.
- New DDR, QDR, and *ZBT* memory interfaces support the latest high-speed memory interfaces.
- New 2x/4x uplink and downlink I/O shift registers capabilities interface high-speed external I/Os to reduced internal logic speed.
- *ORCA* Foundry 2000 development system software. Supported by industry-standard CAE tools for design entry, synthesis, simulation, and timing analysis.

Table 2. System Performance

Function	No. PFUs	2	Unit
16-bit loadable up/down counter	2	282	MHz
16-bit accumulator	2	282	MHz
8 x 8 Parallel Multiplier			
Multiplier mode, unpipelined [1]	11.5	72	MHz
ROM mode, unpipelined [2]	8	175	MHz
Multiplier mode, pipelined [3]	15	197	MHz
32 x 16 RAM (synchronous)			
Single port, 3-state bus [4]	4	264	MHz
Dual-port [5]	4	340	MHz
128 x 8 RAM (synchronous)			
Single port, 3-state bus [4]	8	264	MHz
Dual-port, 3-state bus [5]	8	264	MHz
Address Decode			
8-bit internal, LUT-based	0.25	1.37	ns
8-bit internal, SLIC-based [6]	0	0.73	ns
32-bit internal, LUT-based	2	4.68	ns
32-bit internal, SLIC-based [7]	0	2.08	ns
36-bit Parity Check (internal)	2	4.68	ns

1. Implemented using 8 x 1 multiplier mode (unpipelined), register-to-register, two 8-bit inputs, one 16-bit output.
2. Implemented using two 32 x 4 RAMs and one 12-bit adder, one 8-bit input, one fixed operand, one 16-bit output.
3. Implemented using 8 x 1 multiplier mode (fully pipelined), two 8-bit inputs, one 16-bit output (seven of 15 PFUs contain only pipelining registers).
4. Implemented using 32 x 4 RAM mode with read data on 3-state buffer to bidirectional read/write bus.
5. Implemented using 32 x 4 dual-port RAM mode.
6. Implemented in one partially occupied SLIC, with decoded output setup to CE in the same PLC.
7. Implemented in five partially occupied SLICs.

Product Description

Architecture Overview

The *ORCA* Series 4 architecture is a new generation of SRAM-based programmable devices from Lucent Technologies Microelectronics Group. It includes enhancements and innovations geared toward today's high-speed systems on a single chip. Designed with networking applications in mind, the Series 4 family incorporates system-level features that can further reduce logic requirements and increase system speed. *ORCA* Series 4 devices contain many new patented enhancements and are offered in a variety of packages and speed grades.

The hierarchical architecture of the logic, clocks, routing, RAM, and system-level blocks create a seamless merge of FPGA and ASIC designs. Modular hardware and software technologies enable system-on-chip integration with true plug-and-play design implementation.

The architecture consists of four basic elements: programmable logic cells (PLCs), programmable input/output cells (PIOs), embedded block RAMs (EBRs), and system-level features. These elements are interconnected with a rich routing fabric of both global and local wires. An array of PLCs and its associated resources are surrounded by common interface blocks (CIBs) that provide an abundant interface to the adjacent PIOs or system blocks. Routing congestion around these critical blocks is eliminated by the use of the same routing fabric implemented within the programmable logic core. PICs provide the logical interface to the PIOs which provide the boundary interface off and onto the device. Also, the interquad routing blocks (hIQ, vIQ) separate the quadrants of the PLC array and provide the global routing and clocking elements. Each PLC contains a PFU, SLIC, local routing resources, and configuration RAM. Most of the FPGA logic is performed in the PFU, but decoders, *PAL*-like functions, and 3-state buffering can be performed in the SLIC. The PIOs provide device inputs and outputs and can be used to register signals and to perform input demultiplexing, output multiplexing, uplink and downlink functions, and other functions on two output signals.

The Series 4 architecture integrates macrocell blocks of memory known as EBR. The blocks run horizontally across the PLC array and provide flexible memory functionality. Large blocks of 512 x 18 quad-port RAM complement the existing distributed PFU memory. The RAM blocks can be used to implement RAM, ROM, FIFO, multiplier, and CAM.

System-level functions such as a microprocessor interface, PLLs, embedded system bus elements (located in the corners of the array), the routing resources, and configuration RAM are also integrated elements of the architecture.

Product Description (continued)

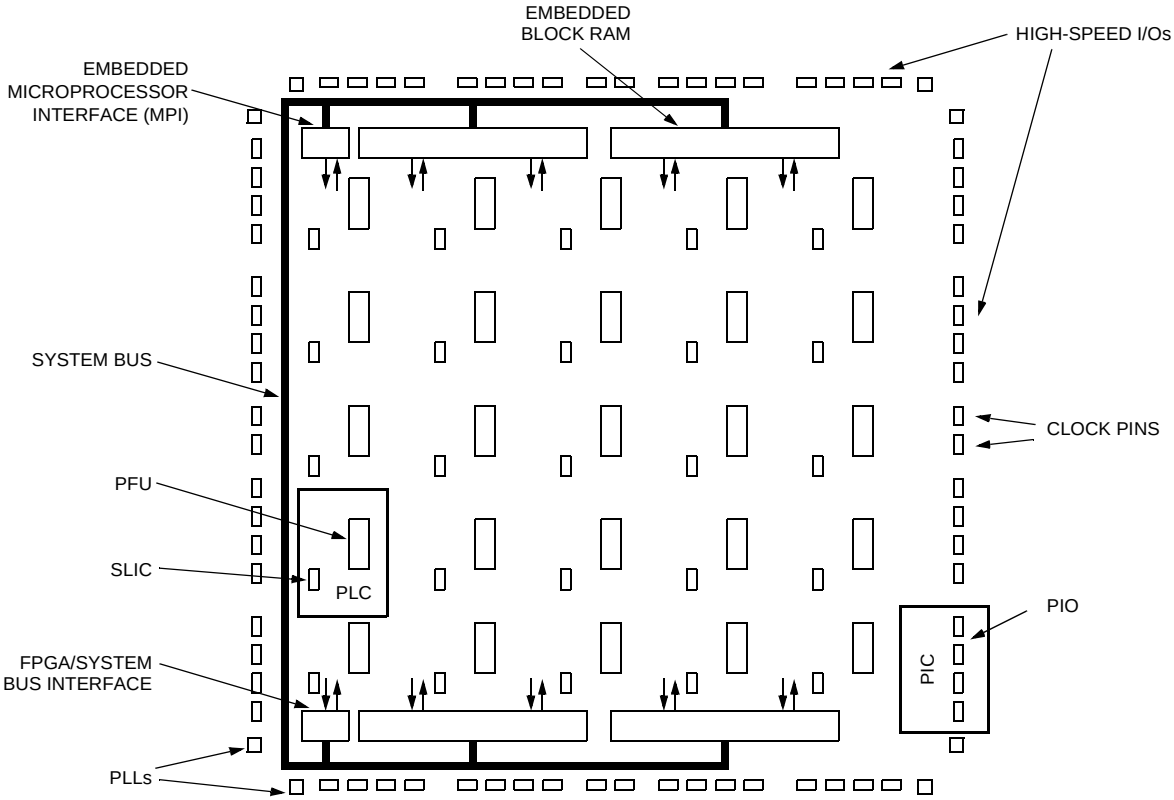


Figure 1. Series 4 Top-Level Diagram

Programmable Logic Cells

The PLCs are arranged in an array of rows and columns. The location of a PLC is indicated by its row and column so that a PLC in the second row and the third column is R2C3. The array of actual PLCs for every device begins with R3C2 in all Series 4 generic FPGAs.

The PLC consists of a PFU, SLIC, and routing resources. Each PFU within a PLC contains eight 4-input (16-bit) LUTs, eight latches/FFs, and one additional FF that may be used independently or with arithmetic functions. The PFU is the main logic element of the PLC, containing elements for both combinatorial and sequential logic. Combinatorial logic is done in LUTs located in the PFU. The PFU can be used in different modes to meet different logic requirements. The LUTs twin-quad architecture provides a configurable medium-/large-grain architecture that can be used to implement from one to eight independent combinatorial logic functions or a large number of complex logic functions using multiple LUTs. The flexibility of the LUT to handle wide input functions, as well as multiple smaller input functions, maximizes the gate count per PFU while increasing system speed.

Programmable Logic Cells (continued)

The PFU is organized in a twin-quad fashion: two sets of four LUTs and FFs that can be controlled independently. Each PFU has two independent programmable clocks, clock enables, local set/reset, and data selects with one available per set of quad FFs.

LUTs may also be combined for use in arithmetic functions using fast-carry chain logic in either 4-bit or 8-bit modes. The carry-out of either mode may be registered in the ninth FF for pipelining. Each PFU may also be configured as a synchronous 32 x 4 single- or dual-port RAM or ROM. The FFs (or latches) may obtain input from LUT outputs or directly from invertible PFU inputs, or they can be tied high or tied low. The FFs also have programmable clock polarity, clock enables, and local set/reset.

The LUTs can be programmed to operate in one of three modes: combinatorial, ripple, or memory. In combinatorial mode, the LUTs can realize any 4-, 5-, or 6-input logic function and many multilevel logic functions using *ORCA*'s SWL connections. In ripple mode, the high-speed carry logic is used for arithmetic functions, comparator functions, or enhanced data path functions. In memory mode, the LUTs can be used as a 32 x 4 synchronous RAM or ROM, in either single- or dual-port mode.

The SLIC is connected from PLC routing resources and from the outputs of the PFU. It contains eight 3-state, bidirectional buffers and logic to perform up to a 10-bit AND function for decoding, or an AND-OR with optional INVERT to perform *PAL*-like functions. The 3-state drivers in the SLIC and their direct connections from the PFU outputs make fast, true 3-state buses possible within the FPGA.

Programmable Function Unit

The PFUs are used for logic. Each PFU has 53 external inputs and 20 outputs and can operate in several modes. The functionality of the inputs and outputs depends on the operating mode.

The PFU uses 36 data input lines for the LUTs, eight data input lines for the latches/FFs, eight control inputs (CLK[1:0], CE[1:0], LSR[1:0], SEL[1:0]), and a carry input (CIN) for fast arithmetic functions and general-purpose data input for the ninth FF. There are eight combinatorial data outputs (one from each LUT), eight latched/registered outputs (one from each latch/FF), a carry-out (COUT), and a registered carry-out (REGCOUT) that comes from the ninth FF. The carry-out signals are used principally for fast arithmetic functions. There are also two dedicated F6 mode outputs which are for the 6-input LUT function and 8-to-1 MUX.

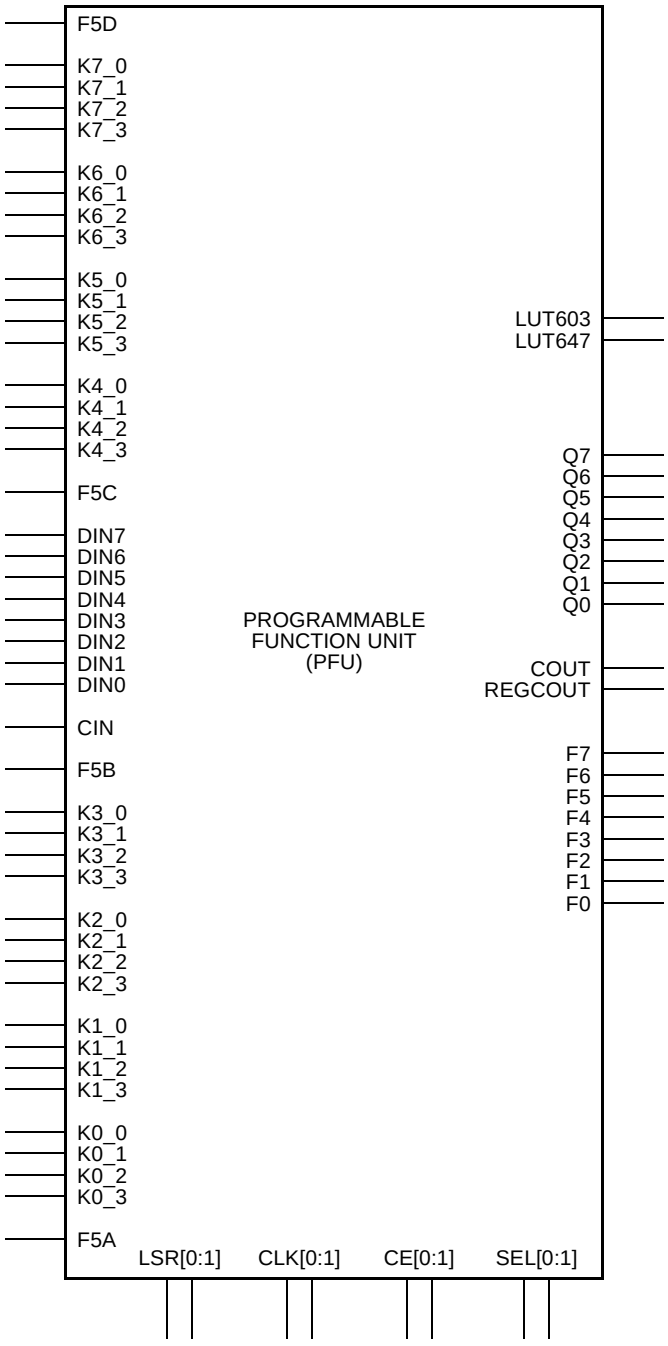
Figure 2 and Figure 3 show high-level and detailed views of the ports in the PFU, respectively. The eight sets of LUT inputs are labeled as K0 through K7 with each of the four inputs to each LUT having a suffix of _x, where x is a number from 0 to 3.

There are four F5 inputs labeled A through D. These are used for additional LUT inputs for 5- and 6-input LUTs or as a selector for multiplexing two 4-input LUTs. Four adjacent LUT4s can also be multiplexed together with a 4-to-1 MUX to create a 6-input LUT. The eight direct data inputs to the latches/FFs are labeled as DIN[7:0]. Registered LUT outputs are shown as Q[7:0], and combinatorial LUT outputs are labeled as F[7:0].

The PFU implements combinatorial logic in the LUTs and sequential logic in the latches/FFs. The LUTs are static random access memory (SRAM) and can be used for read/write or ROM.

Each latch/FF can accept data from its associated LUT. Alternatively, the latches/FFs can accept direct data from DIN[7:0], eliminating the LUT delay if no combinatorial function is needed. Additionally, the CIN input can be used as a direct data source for the ninth FF. The LUT outputs can bypass the latches/FFs, which reduces the delay out of the PFU. It is possible to use the LUTs and latches/FFs more or less independently, allowing, for instance, a comparator function in the LUTs simultaneously with a shift register in the FFs.

Programmable Logic Cells (continued)

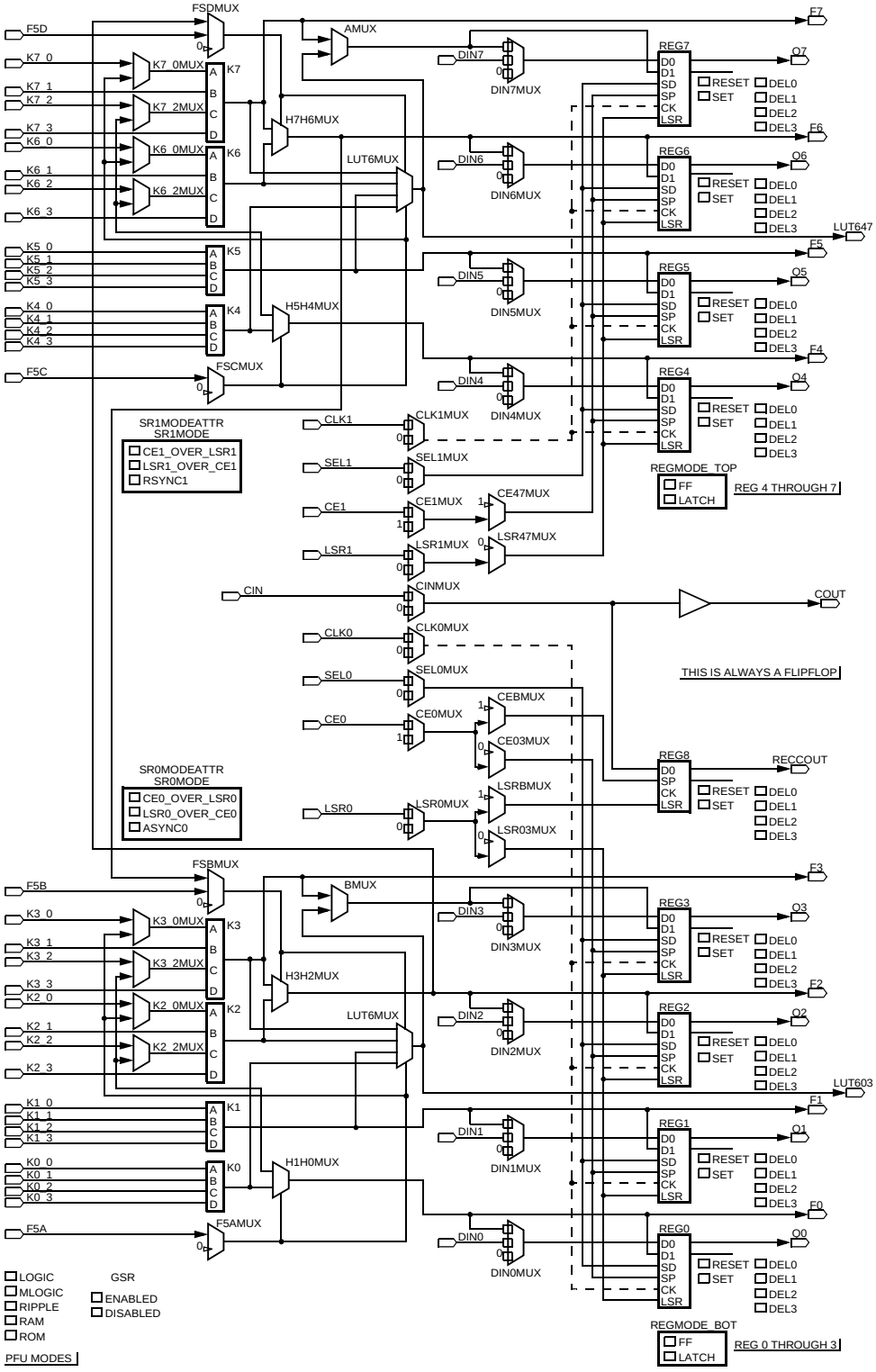


5-5752(F)a

Figure 2. PFU Ports

The PFU can be configured to operate in four modes: logic mode, half-logic mode, ripple mode, and memory (RAM/ROM) mode. In addition, ripple mode has four submodes and RAM mode can be used in either a single- or dual-port memory fashion. These submodes of operation are discussed in the following sections.

Programmable Logic Cells (continued)



5-9714(F)

Note: All multiplexers without select inputs are configuration selector multiplexers.

Figure 3. Simplified PFU Diagram

Programmable Logic Cells (continued)

Look-Up Table Operating Modes

The operating mode affects the functionality of the PFU input and output ports and internal PFU routing. For example, in some operating modes, the DIN[7:0] inputs are direct data inputs to the PFU latches/FFs. In memory mode, the same DIN[7:0] inputs are used as a 4-bit write data input bus and a 4-bit write address input bus into LUT memory.

Table 3 lists the basic operating modes of the LUT. Figure 4—Figure 7 show block diagrams of the LUT operating modes. The accompanying descriptions demonstrate each mode's use for generating logic.

Table 3. Look-Up Table Operating Modes

Mode	Function
Logic	4-, 5-, and 6-input LUTs; softwired LUTs; latches/FFs with direct input or LUT input; CIN as direct input to ninth FF or as pass through to COUT.
Half Logic/ Half Ripple	Upper four LUTs and latches/FFs in logic mode; lower four LUTs and latches/FFs in ripple mode; CIN and ninth FF for logic or ripple functions.
Ripple	All LUTs combined to perform ripple-through data functions. Eight LUT registers available for direct-in use or to register ripple output. Ninth FF dedicated to ripple out, if used. The submodes of ripple mode are adder/subtractor, counter, multiplier, and comparator.
Memory	All LUTs and latches/FFs used to create a 32x4 synchronous dual-port RAM. Can be used as single-port or as ROM.

PFU Control Inputs

Each PFU has eight routable control inputs and an active-low, asynchronous global set/reset (GSRN) signal that affects all latches and FFs in the device. The eight control inputs are CLK[1:0], LSR[1:0], CE[1:0], and SEL[1:0], and their functionality for each logic mode of the PFU is shown in Table 4. The clock signal to the PFU is CLK, CE stands for clock enable, which is its primary function. LSR is the local set/reset signal that can be configured as synchronous or asynchronous. The selection of set or reset is made for each latch/FF and is not a function of the signal itself. SEL is used to dynamically select between direct PFU input and LUT output data as the input to the latches/FFs.

All of the control signals can be disabled and/or inverted via the configuration logic. A disabled clock enable indicates that the clock is always enabled. A disabled LSR indicates that the latch/FF never sets/resets (except from GSRN). A disabled SEL input indicates that DIN[7:0] PFU inputs or the LUT outputs are always input to the latches/FFs.

Table 4. Control Input Functionality

Mode	CLK[1:0]	LSR[1:0]	CE[1:0]	SEL[1:0]
Logic	CLK to all latches/ FFs	LSR to all latches/FFs, enabled per nibble and for ninth FF	CE to all latches/FFs, selectable per nibble and for ninth FF	Select between LUT input and direct input for eight latches/FFs
Half Logic/ Half Ripple	CLK to all latches/ FFs	LSR to all latches/FF, enabled per nibble and for ninth FF	CE to all latches/FFs, selectable per nibble and for ninth FF	Select between LUT input and direct input for eight latches/FFs
Ripple	CLK to all latches/ FFs	LSR to all latches/FFs, enabled per nibble and for ninth FF	CE to all latches/FFs, selectable per nibble and for ninth FF	Select between LUT input and direct input for eight latches/FFs
Memory (RAM)	CLK to RAM	LSR0 port enable 2	CE1 RAM write enable CE0 Port enable 1	Not used
Memory (ROM)	Optional for synchronous outputs	Not used	Not used	Not used

Programmable Logic Cells (continued)

Logic Mode

The PFU diagram of Figure 3 represents the logic mode of operation. In logic mode, the eight LUTs are used individually or in flexible groups to implement user logic functions. The latches/FFs may be used in conjunction with the LUTs or separately with the direct PFU data inputs. There are three basic submodes of LUT operation in PFU logic mode: F4 mode, F5 mode, and the F6 mode. Combinations of the submodes are possible in each PFU.

F4 mode, shown simplified in Figure 4, illustrates the uses of the basic 4-input LUTs in the PFU. The output of an F4 LUT can be passed out of the PFU, captured at the LUTs associated latch/FF, or multiplexed with the adjacent F4 LUT output using one of the F5[A:D] inputs to the PFU (not shown). Only adjacent LUT pairs (K_0 and K_1, K_2 and K_3, K_4 and K_5, K_6 and K_7) can be multiplexed, and the output always goes to the even-numbered output of the pair.

The F5 submode of the LUT operation, shown simplified in Figure 4, indicates the use of 5-input LUTs to implement logic. 5-input LUTs are created from two 4-input LUTs and a multiplexer. The F5 LUT is the same as the multiplexing of two F4 LUTs described previously with the constraint that the inputs to both F4 LUTs be the same. The F5[A:D] input is then used as the fifth LUT input. The equations for the two F4 LUTs will differ by the assumed value for the F5[A:D] input, one F4 LUT assuming that the F5[A:D] input is zero, and the other assuming it is a one. The selection of the appropriate F4 LUT output in the F5 MUX by the F5[A:D] signal creates a 5-input LUT.

Two 6-input LUTs are created by shorting together the inputs of four 4-input LUTs (K0:3 and K4:7) which are multiplexed together. The F5 inputs of the adjacent F4 LUTs derive the fifth and sixth inputs of the F6 mode as shown in Figure 5. The F6 outputs, LUT603 and LUT647, are dedicated to the F6 mode or can be used as the outputs of MUX8x1. MUX8x1 modes as shown in Figure 7 are created by programming adjacent 4-input LUTs to 2x1 MUXs and multiplexing down to create MUX8x1. Other functions can be implemented from the configuration shown in Figure 5 where the four LUT4s drive the 4x1 MUX in each half of the PFU if the LUT4 inputs are not tied to the same inputs. Both F6 mode and MUX8x1 are available in the upper and lower PFU nibbles.

Any combination of F4 and F5 LUTs is allowed per PFU using the eight 16-bit LUTs. Examples are eight F4 LUTs, four F5 LUTs, a combination of four F4 plus two F5 LUTs, a combination of two F4, one F5, plus one F6, or a combination of one F5, one MUX21 of two LUT4s, and one MUX41 of four LUT4s.

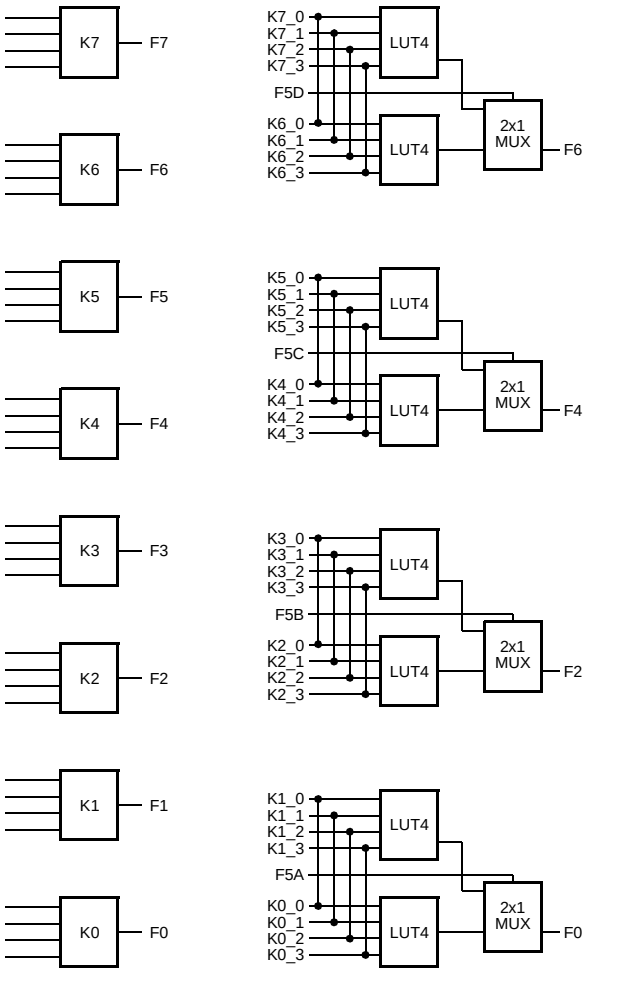


Figure 4. Simplified F4 and F5 Logic Modes

Programmable Logic Cells (continued)

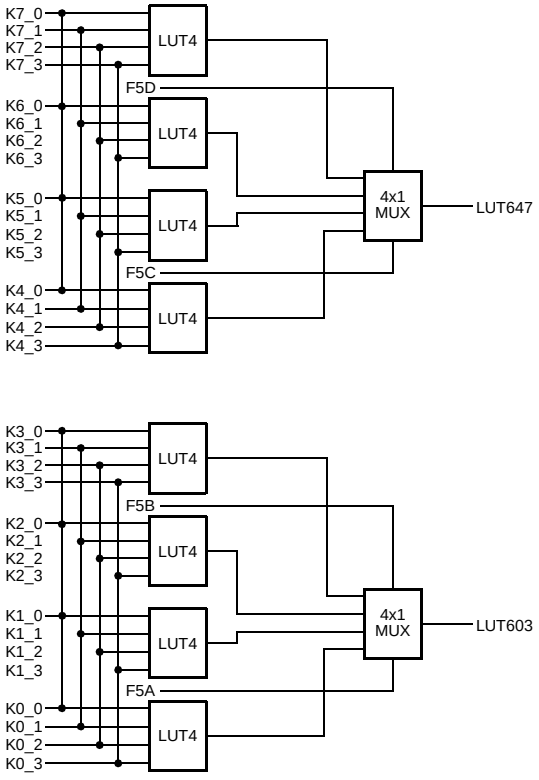


Figure 5. Simplified F6 Logic Modes

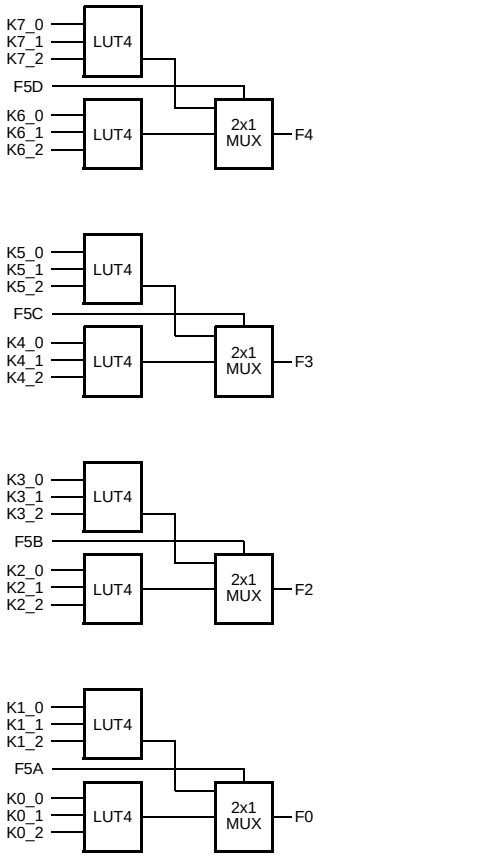
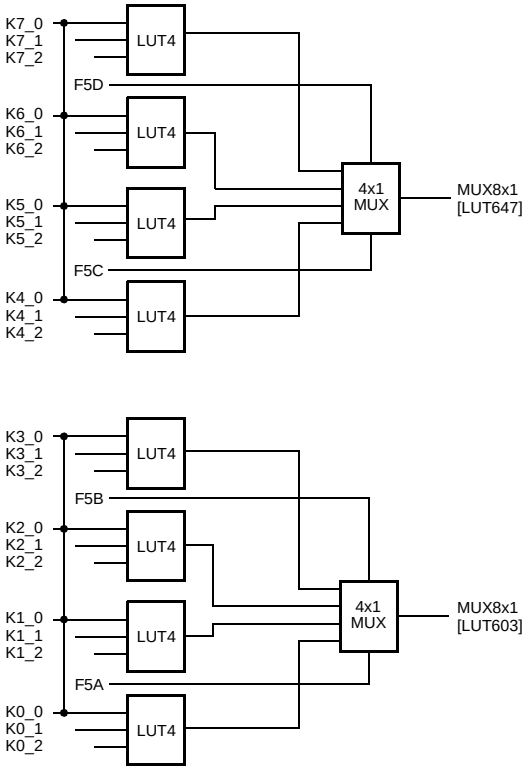


Figure 6. MUX 4x1

Programmable Logic Cells (continued)



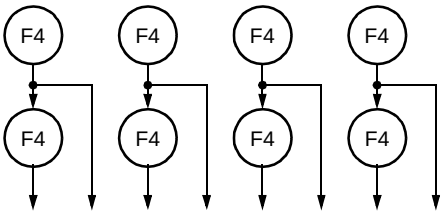
5-9736(F)a

Figure 7. MUX 8x1

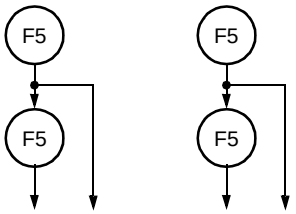
Softwired LUT capability uses F4, F5, and F6 LUTs, along with MUX21 and MUX41 blocks together with internal PFU feedback routing, to generate complex logic functions up to three LUT levels deep. Multiplexers can be independently configured to route certain LUT outputs to the input of other LUTs. In this manner, very complex logic functions, some of up to 22 inputs, can be implemented in a single PFU at greatly enhanced speeds.

It is important to note that an LUT output that is fed back for softwired use is still available to be registered or output from the PFU. This means, for instance, that a logic equation that is needed by itself and as a term in a larger equation need only be generated once, and PLC routing resources will not be required to use it in the larger equation.

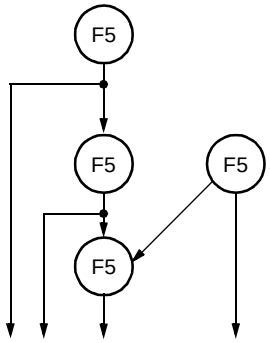
Programmable Logic Cells (continued)



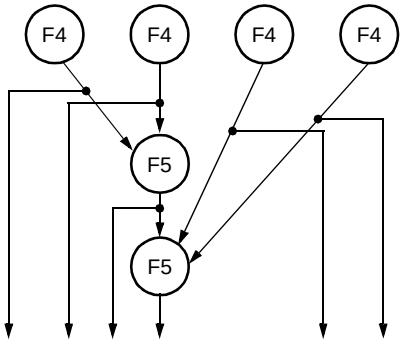
FOUR 7-INPUT FUNCTIONS IN ONE PFU



TWO 9-INPUT FUNCTIONS IN ONE PFU

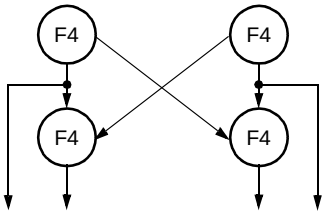


ONE 17-INPUT FUNCTION IN ONE PFU

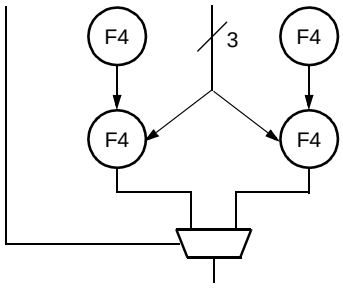


ONE 21-INPUT FUNCTION IN ONE PFU

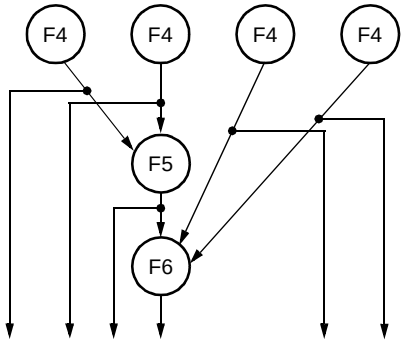
5-5753 (F)



TWO 10-INPUT FUNCTIONS IN ONE PFU



ONE OF TWO 21-INPUT FUNCTIONS IN ONE PFU



ONE 22-INPUT FUNCTION IN ONE PFU

5-5754 (F)

Figure 8. Softwired LUT Topology Examples

Programmable Logic Cells (continued)

Half-Logic Mode

Series 4 FPGAs are based upon a twin-quad architecture in the PFUs. The bytewide nature (eight LUTs, eight latches/FFs) may just as easily be viewed as two nibbles (two sets of four LUTs, four latches/FFs). The two nibbles of the PFU are organized so that any nibblewide feature (excluding some softwired LUT topologies) can be swapped with any other nibblewide feature in another PFU. This provides for very flexible use of logic and for extremely flexible routing. The half-logic mode of the PFU takes advantage of the twin-quad architecture and allows half of a PFU, $K_{[7:4]}$ and associated latches/FFs, to be used in logic mode while the other half of the PFU, $K_{[3:0]}$ and associated latches/FFs, is used in ripple mode. In half-logic mode, the ninth FF may be used as a general-purpose FF or as a register in the ripple mode carry chain.

Ripple Mode

The PFU LUTs can be combined to do bytewide ripple functions with high-speed carry logic. Each LUT has a dedicated carry-out net to route the carry to/from any adjacent LUT. Using the internal carry circuits, fast arithmetic, counter, and comparison functions can be implemented in one PFU. Similarly, each PFU has carry-in (CIN, FCIN) and carry-out (COUT, FCOUT) ports for fast-carry routing between adjacent PFUs.

The ripple mode is generally used in operations on two data buses. A single PFU can support an 8-bit ripple function. Data buses of 4 bits and less can use the nibblewide ripple chain that is available in half-logic mode. This nibblewide ripple chain is also useful for longer ripple chains where the length modulo 8 is four or less. For example, a 12-bit adder (12 modulo 8 = 4) can be implemented in one PFU in ripple mode (8 bits) and one PFU in half-logic mode (4 bits), freeing half of a PFU for general logic mode functions.

Each LUT has two operands and a ripple (generally carry) input, and provides a result and ripple (generally carry) output. A single bit is rippled from the previous LUT and is used as input into the current LUT. For LUT K_0, the ripple input is from the PFU CIN or FCIN port. The CIN/FCIN data can come from either the fast-carry routing (FCIN) or the PFU input (CIN), or it can be tied to logic 1 or logic 0.

In the following discussions, the notations LUT K_7/K_3 and F[7:0]/F[3:0] are used to denote the LUT that provides the carry-out and the data outputs for full PFU ripple operation (K_7, F[7:0]) and half-logic ripple operation (K_3, F[3:0]), respectively. The ripple mode diagram (Figure 9) shows full PFU ripple operation, with half-logic ripple connections shown as dashed lines.

The result output and ripple output are calculated by using generate/propagate circuitry. In ripple mode, the two operands are input into $K_Z[1]$ and $K_Z[0]$ of each LUT. The result bits, one per LUT, are F[7:0]/F[3:0] (see Figure 9). The ripple output from LUT K_7/K_3 can be routed on dedicated carry circuitry into any of four adjacent PLCs, and it can be placed on the PFU COUT/FCOUT outputs. This allows the PLCs to be cascaded in the ripple mode so that nibblewide ripple functions can be expanded easily to any length.

Result outputs and the carry-out may optionally be registered within the PFU. The capability to register the ripple results, including the carry output, provides for improved counter performance and simplified pipelining in arithmetic functions.

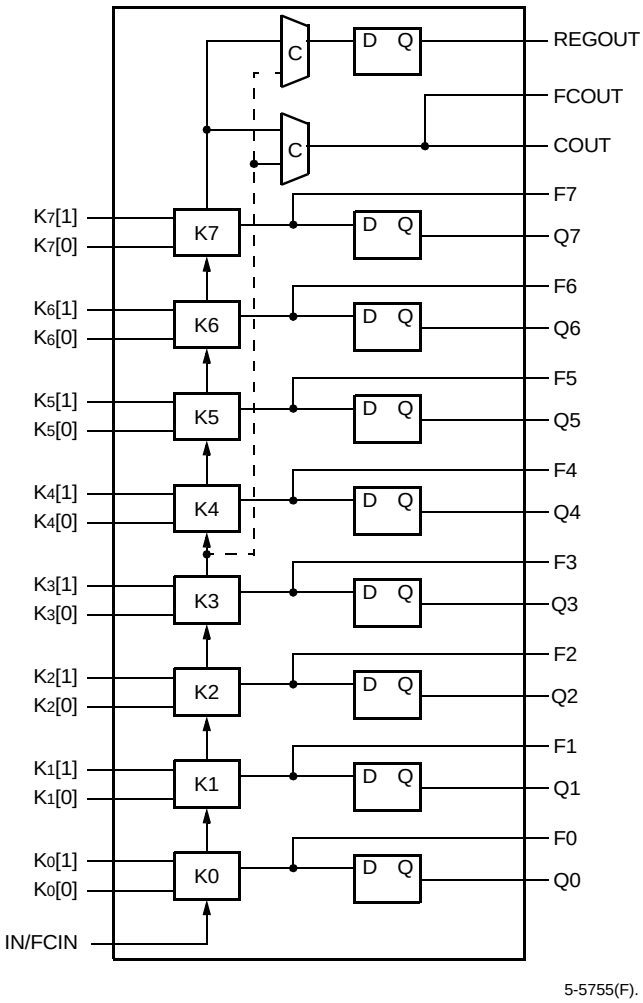


Figure 9. Ripple Mode

Programmable Logic Cells (continued)

The ripple mode can be used in one of four submodes. The first of these is **adder-subtractor submode**. In this submode, each LUT generates three separate outputs. One of the three outputs selects whether the carry-in is to be propagated to the carry-out of the current LUT or if the carry-out needs to be generated. If the carry-out needs to be generated, this is provided by the second LUT output. The result of this selection is placed on the carry-out signal, which is connected to the next LUT carry-in or the COUT/FCOUT signal, if it is the last LUT (K_7/K_3). Both of these outputs can be any equation created from Kz[1] and Kz[0], but in this case, they have been set to the propagate and generate functions.

The third LUT output creates the result bit for each LUT output connected to F[7:0]/F[3:0]. If an adder/subtractor is needed, the control signal to select addition or subtraction is input on F5A/F5C inputs. These inputs generate the controller input AS. When AS = 0, this function performs the adder, A + B. When AS = 1, the function performs the subtractor, A – B. The result bit is created in one-half of the LUT from a single bit from each input bus Kz[1:0], along with the ripple input bit.

The second submode is the **counter submode** (see Figure 10). The present count, which may be initialized via the PFU DIN inputs to the latches/FFs, is supplied to input Kz[0], and then output F[7:0]/F[3:0] will either be incremented by one for an up counter or decremented by one for a down counter. If an up/down counter is needed, the control signal to select the direction (up or down) is input on F5A and F5C. When F5[A:C], respectively per nibble, is a logic 1, this indicates a down counter and a logic 0 indicates an up counter.

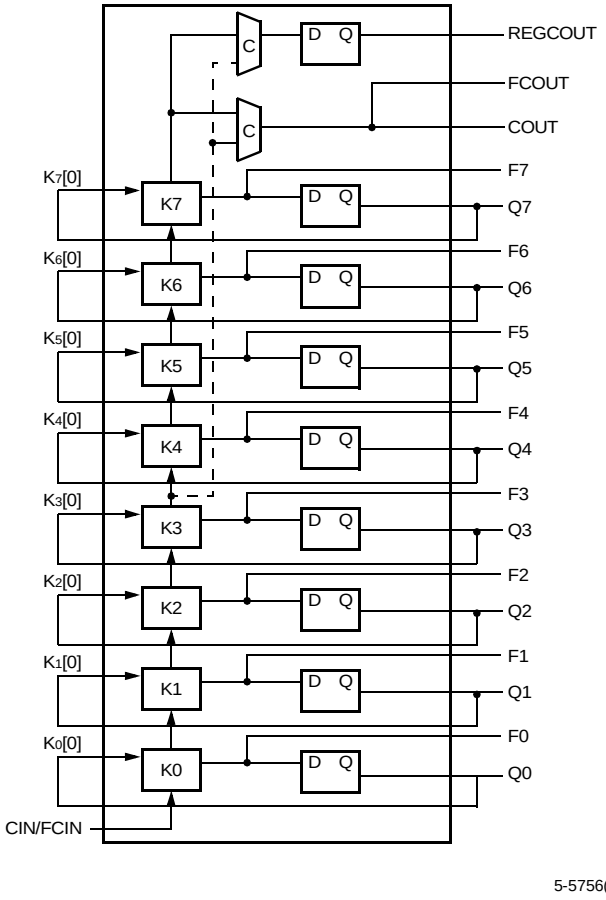


Figure 10. Counter Submode

Programmable Logic Cells (continued)

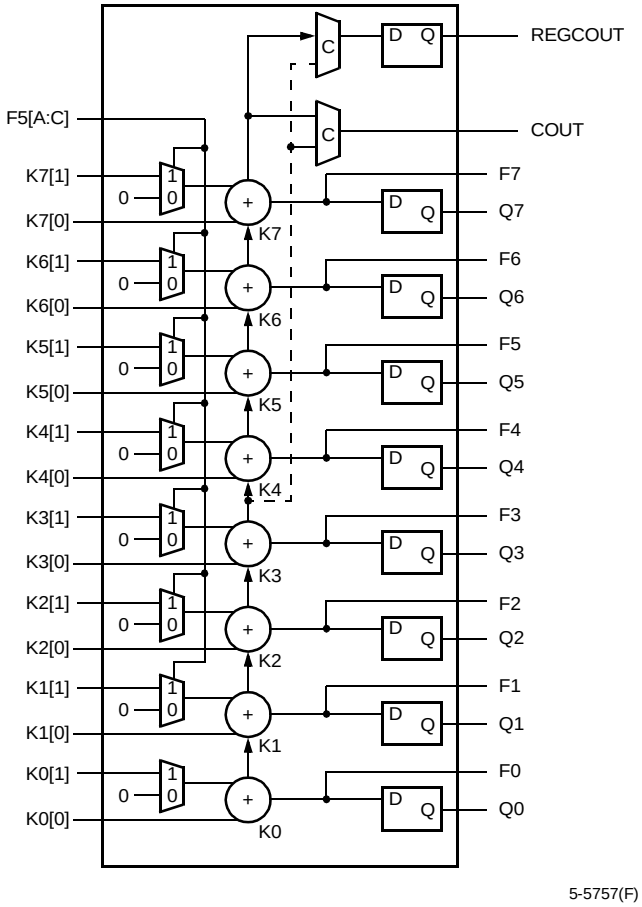
In the third submode, **multiplier submode**, a single PFU can affect an 8x1 bit (4x1 for half-ripple mode) multiply and sum with a partial product (see Figure 11). The multiplier bit is input at F5[A:C], respectively per nibble, and the multiplicand bits are input at Kz[1], where Kz[1] is the most significant bit (MSB). Kz[0] contains the partial product (or other input to be summed) from a previous stage. If F5[A:C] is logical 1, the multiplicand is added to the partial product. If F5[A:C] is logical 0, 0 is added to the partial product, which is the same as passing the partial product. CIN/FCIN can bring the carry-in from the less significant PFUs if the multiplicand is wider than 8 bits, and COUT/FCOUT holds any carry-out from the multiplication, which may then be used as part of the product or routed to another PFU in multiplier mode for multiplicand width expansion.

Ripple mode's fourth submode features **equality comparators**. The functions that are explicitly available are A ≥ B, A ≠ B, and A ≤ B, where the value for A is input on Kz[0], and the value for B is input on Kz[1]. A value of 1 on the carry-out signals valid argument. For example, a carry-out equal to 1 in AB submode indicates that the value on Kz[0] is greater than or equal to the value on Kz[1]. Conversely, the functions A ≤ B, A + B, and A > B are available using the same functions but with a 0 output expected. For example, A > B with a 0 output indicates A ≤ B. Table 5 shows each function and the output expected.

If larger than 8 bits, the carry-out signal can be cascaded using fast-carry logic to the carry-in of any adjacent PFU. The use of this submode could be shown using Figure 9, except that the CIN/FCIN input for the least significant PFU is controlled via configuration.

Table 5. Ripple Mode Equality Comparator Functions and Outputs

Equality Function	*ORCA* Foundry Submode	True, if Carry-Out Is:
A ≥ B	A ≥ B	1
A ≤ B	A ≤ B	1
A ≠ B	A ≠ B	1
A < B	A > B	0
A > B	A < B	0
A = B	A ≠ B	0



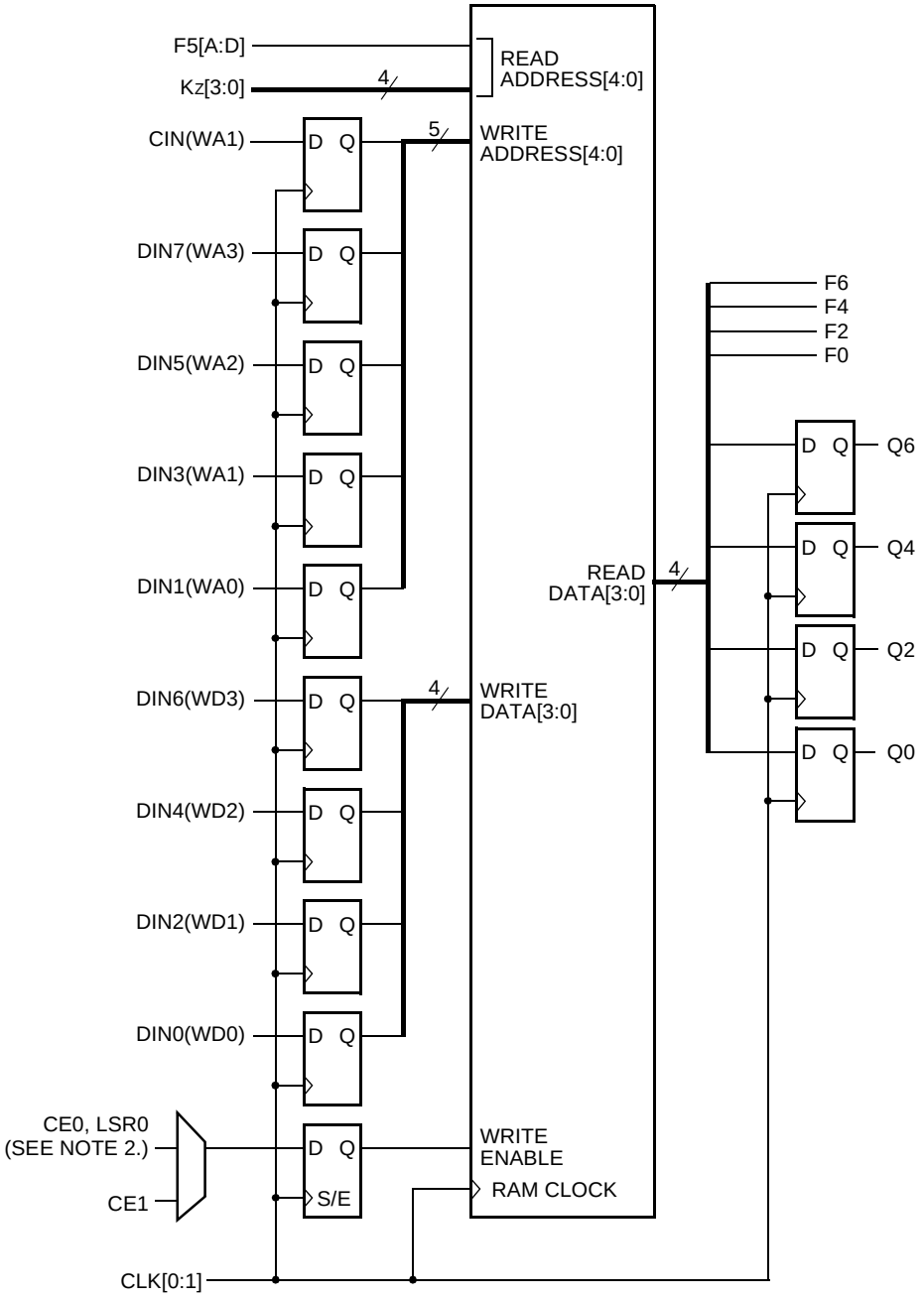
Key: C = configuration data.
Note: F5[A:C] shorted together.

Figure 11. Multiplier Submode

Programmable Logic Cells (continued)

Memory Mode

The Series 4 PFU can be used to implement a 32 x 4 (128-bit) synchronous, dual-port RAM. A block diagram of a PFU in memory mode is shown in Figure 12. This RAM can also be configured to work as a single-port memory and because initial values can be loaded into the RAM during configuration, it can also be used as a ROM.



5-5969(F)a

1. CLK[0:1] are commonly connected in memory mode.
2. CE1 = write enable = wren; wren = 0 (no write enable); wren = 1 (write enabled).
   CE0 = write port enable 0; CE0 = 0, wren = 0; CE0 = 1, wren = CE1.
   LSR0 = write port enable 1; LSR0 = 0, wren = CE0; LSR0 = 1, wren = CE1.

Figure 12. Memory Mode

Programmable Logic Cells (continued)

The PFU memory mode uses all LUTs and latches/FFs including the ninth FF in its implementation as shown in Figure 12. The read address is input at the Kz[3:0] and F5[A:D] inputs where Kz[0] is the LSB and F5[A:D] is the MSB, and the write address is input on CIN (MSB) and DIN[7, 5, 3, 1], with DIN[1] being the LSB. Write data is input on DIN[6, 4, 2, 0], where DIN[6] is the MSB, and read data is available combinatorially on F[6, 4, 2, 0] and registered on Q[6, 4, 2, 0] with F[6] and Q[6] being the MSB. The write enable controlling ports are input on CE0, CE1, and LSR0. CE1 is the active-high write enable (CE1 = 1, RAM is write enabled). The first write port is enabled by CE0. The second write port is enabled with LSR0. The PFU CLK (CLK0) signal is used to synchronously write the data. The polarities of the clock, write enable, and port enables are all programmable. Write-port enables may be disabled if they are not to be used.

Data is written to the write data, write address, and write enable registers on the active edge of the clock, but data is not written into the RAM until the next clock edge one-half cycle later. The read port is actually asynchronous, providing the user with read data very quickly after setting the read address, but timing is also provided so that the read port may be treated as fully synchronous for write then read applications. If the read and write address lines are tied together (maintaining MSB to MSB, etc.), then the dual-port RAM operates as a synchronous single-port RAM. If the write enable is disabled, and an initial memory contents are provided at configuration time, the memory acts as a ROM (the write data and write address ports and write port enables are not used).

Wider memories can be created by operating two or more memory mode PFUs in parallel, all with the same address and control signals, but each with a different nibble of data. To increase memory word depth above 32, two or more PLCs can be used. Figure 10 shows a 128 x 8 dual-port RAM that is implemented in eight PLCs. This figure demonstrates data path width expansion by placing two memories in parallel to achieve an 8-bit data path. Depth expansion is applied to achieve 128 words deep using the 32-word deep PFU memories. In addition to the PFU in each PLC, the SLIC (described in the next section) in each PLC is used for read address decodes and 3-state drivers. The 128 x 8 RAM shown could be made to operate as a single-port RAM by tying (bit-for-bit) the read and write addresses.

To achieve depth expansion, one or two of the write address bits (generally the MSBs) are routed to the write port enables as in Figure 10. For 2 bits, the bits select which 32-word bank of RAM of the four available from a decode of two WPE inputs is to be written. Similarly, 2 bits of the read address are decoded in the SLIC and are used to control the 3-state buffers through which the read data passes. The write data bus is common, with separate nibbles for width expansion, across all PLCs, and the read data bus is common (again, with separate nibbles) to all PLCs at the output of the 3-state buffers.

Figure 13 also shows the capability to provide a read enable for RAMs/ROMs using the SLIC cell. The read enable will 3-state the read data bus when inactive, allowing the write data and read data buses to be tied together if desired.

Programmable Logic Cells (continued)

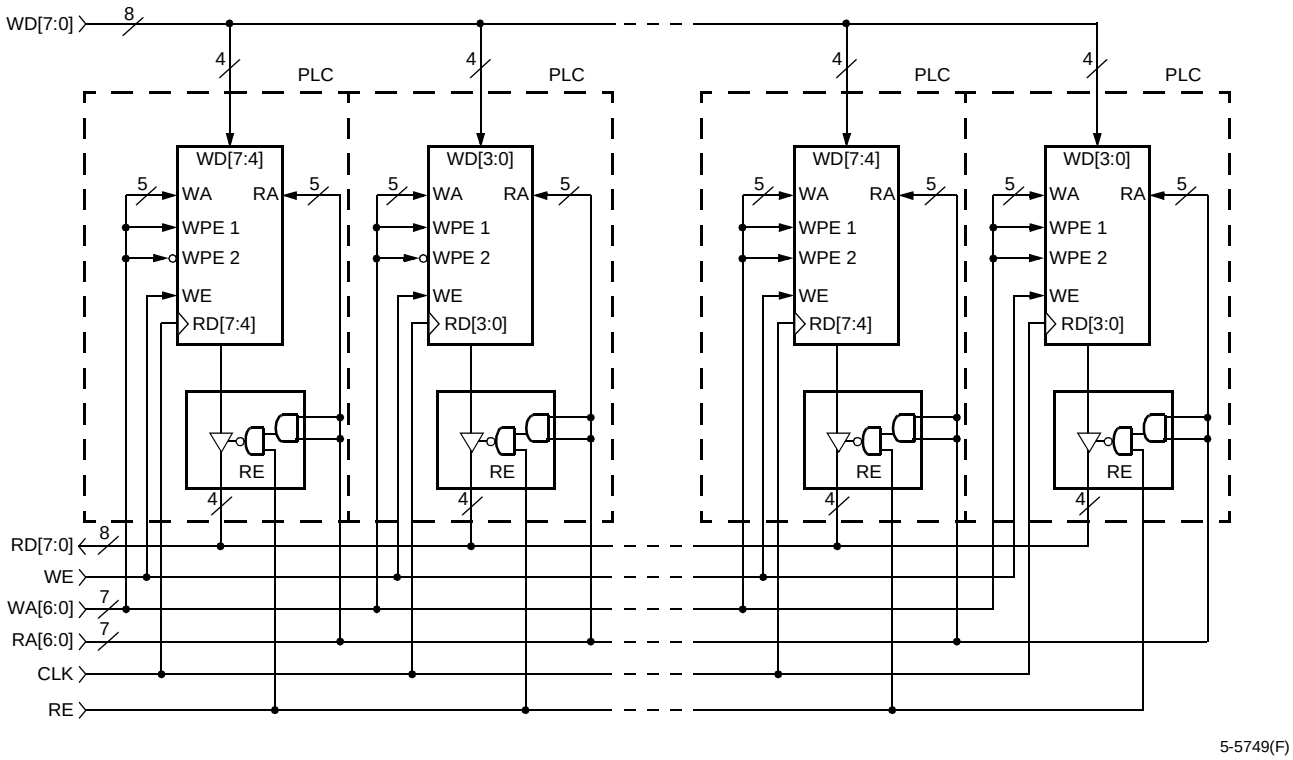


Figure 13. Memory Mode Expansion Example—128 x 8 RAM

Supplemental Logic and Interconnect Cell

Each PLC contains a SLIC embedded within the PLC routing, outside of the PFU. As its name indicates, the SLIC performs both logic and interconnect (routing) functions. Its main features are 3-statable, bidirectional buffers, and a *PAL*-like decoder capability. Figure 14 shows a diagram of a SLIC with all of its features shown. All modes of the SLIC are not available at one time.

The ten SLIC inputs can be sourced directly from the PFU or from the general routing fabric. SI[0:9] inputs can come from the horizontal or vertical routing, and I[0:9} comes from the PFU outputs O[9:0]. These inputs can also be tied to a logical 1 or 0 constant. The inputs are twin-quad in nature and are segregated into two groups of four nibbles and a third group of two inputs for control. Each input nibble groups also have 3-state capability; however, the third pair does not.

There is one 3-state control (TRI) for each SLIC, with the capability to invert or disable the 3-state control for each group of four BIDIs. Separate 3-state control for each nibblewide group is achievable by using the SLICs decoder (DEC) output, driven by the group of two BIDIs, to control the 3-state of one BIDI nibble while using the TRI signal to control the 3-state of the other BIDI nibble. Figure 15 shows the SLIC in buffer mode with available 3-state control from the TRI and DEC signals. If the entire SLIC is acting in a buffer capacity, the DEC output may be used to generate a constant logic 1 (V_{HI}) or logic 0 (V_{LO}) signal for general use.

The SLIC may also be used to generate *PAL*-like AND-OR with optional INVERT (AOI) functions or a decoder of up to 10 bits. Each group of buffers can feed into an AND gate (4-input AND for the nibble groups and 2-input AND for the other two buffers). These AND gates then feed into a 3-input gate that can be configured as either an AND gate or an OR gate. The output of the 3-input gate is invertible and is output at the DEC output of the SLIC. Figure 19 shows the SLIC in full decoder mode.

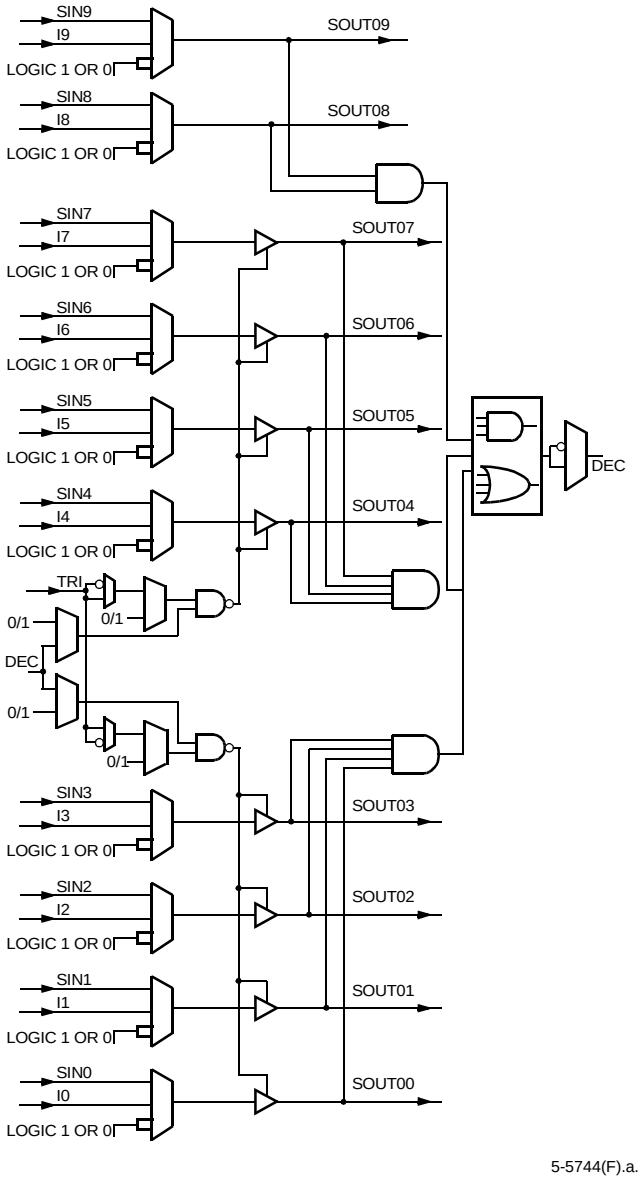
The functionality of the SLIC is parsed by the two nibblewide groups and the 2-bit buffer group. Each of these groups may operate independently as BIDI buffers (with or without 3-state capability for the nibblewide groups) or as a *PAL*/decoder.

Programmable Logic Cells (continued)

As discussed in the Memory Mode section on page 19, if the SLIC is placed into one of the modes where it contains both buffers and a decode or AOI function (e.g., BUF_BUF_DEC mode), the DEC output can be gated with the 3-state input signal. This allows up to a 6-input decode (e.g., BUF_DEC_DEC mode) plus the 3-state input to control the enable/disable of up to four buffers per SLIC. Figure 15—Figure 19 show several configurations of the SLIC, while Table 6 shows all of the possible modes.

Table 6. SLIC Modes

Mode No.	Mode	BUF [3:0]	BUF [7:4]	BUF [9:8]
1	BUFFER	Buffer	Buffer	Buffer
2	BUF_BUF_DEC	Buffer	Buffer	Decoder
3	BUF_DEC_BUF	Buffer	Decoder	Buffer
4	BUF_DEC_DEC	Buffer	Decoder	Decoder
5	DEC_BUF_BUF	Decoder	Buffer	Buffer
6	DEC_BUF_DEC	Decoder	Buffer	Decoder
7	DEC_DEC_BUF	Decoder	Decoder	Buffer
8	DECODER	Decoder	Decoder	Decoder



5-5744(F).a.

Figure 14. SLIC All Modes Diagram

Programmable Logic Cells (continued)

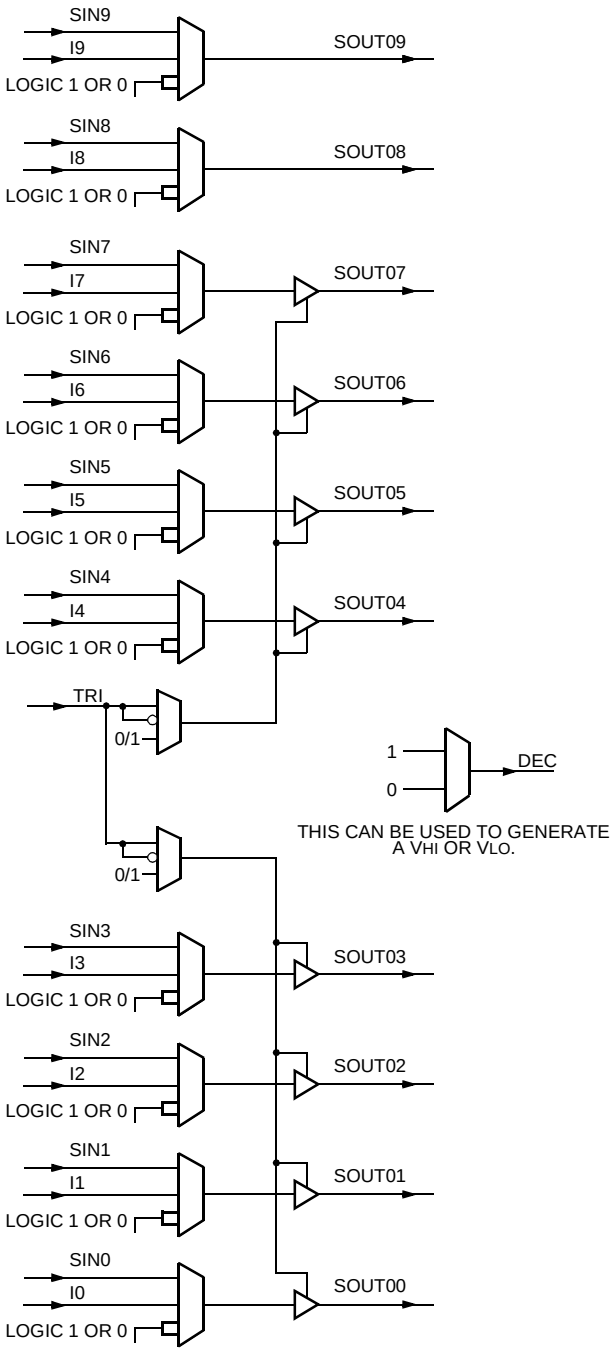


Figure 15. Buffer Mode

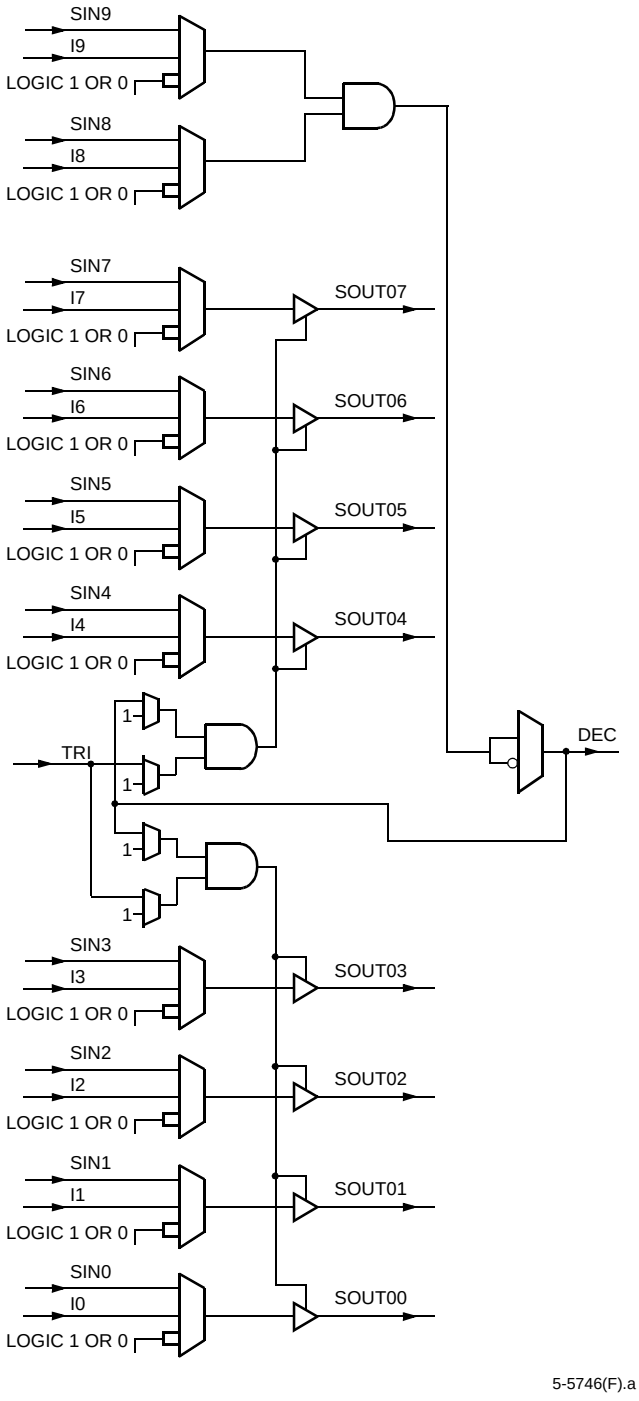


Figure 16. Buffer-Buffer-Decoder Mode

Programmable Logic Cells (continued)

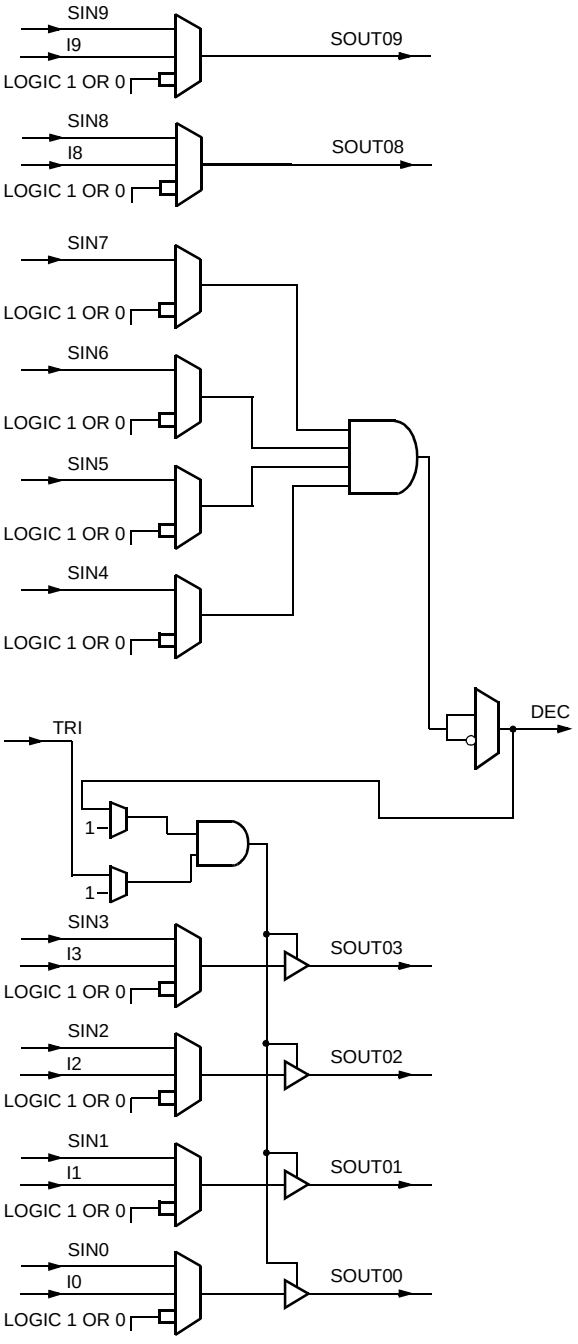


Figure 17. Buffer-Decoder-Buffer Mode

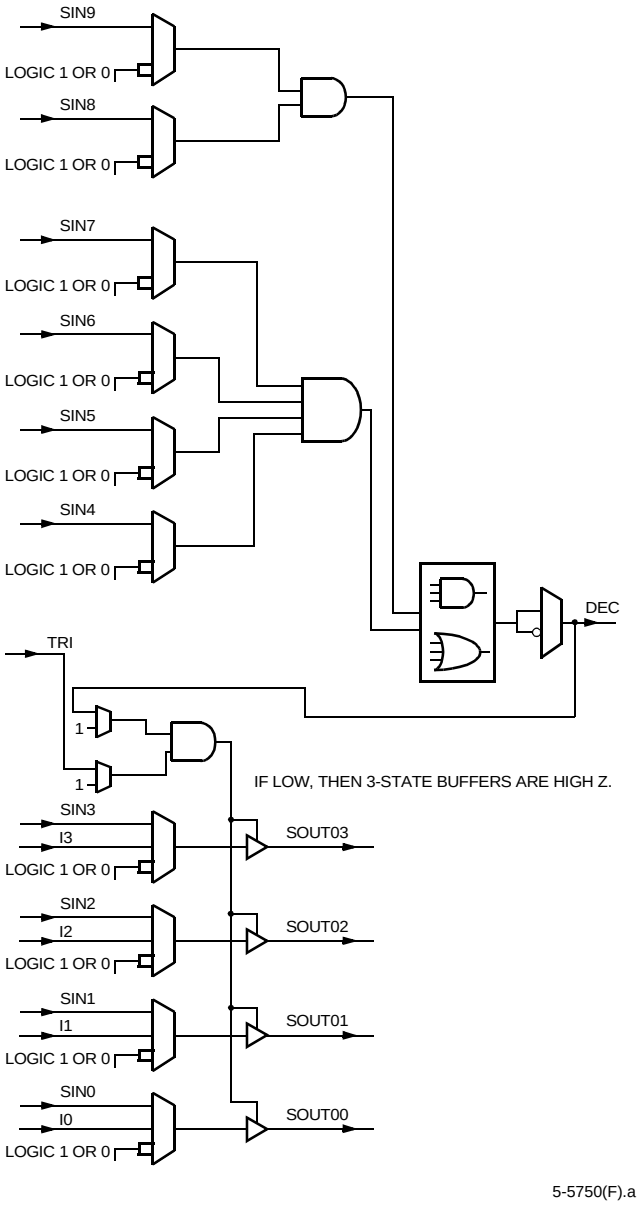


Figure 18. Buffer-Decoder-Decoder Mode

Programmable Logic Cells (continued)

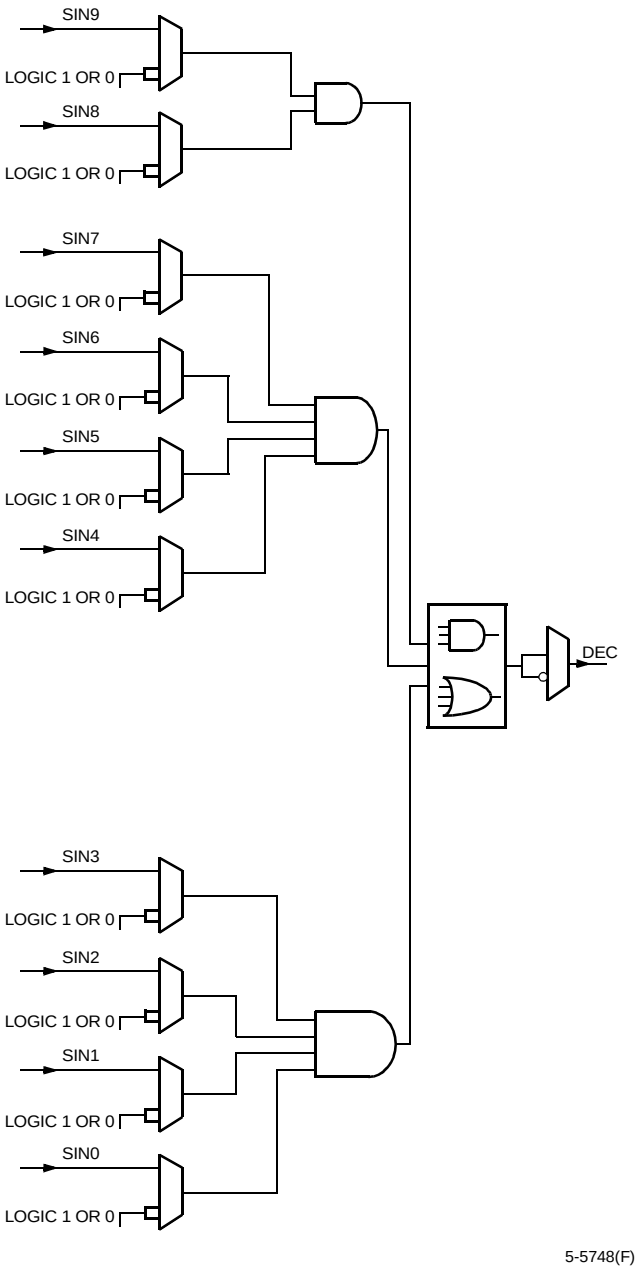


Figure 19. Decoder Mode

PLC Latches/Flip-Flops

The eight general-purpose latches/FFs in the PFU can be used in a variety of configurations. In some cases, the configuration options apply to all eight latches/FFs in the PFU and some apply to the latches/FFs on a nibblewide basis where the ninth FF is considered independently. For other options, each latch/FF is independently programmable. In addition, the ninth FF can be used for a variety of functions.

Table 7 summarizes these latch/FF options. The latches/FFs can be configured as either positive- or negative-level sensitive latches, or positive or negative edge-triggered FFs (the ninth register can only be a FF). All latches/FFs in a given nibble of a PFU share the same clock, and the clock to these latches/FFs can be inverted. The input into each latch/FF is from either the corresponding LUT output (F[7:0]) or the direct data input (DIN[7:0]). The latch/FF input can also be tied to logic 1 or to logic 0, which is the default.

Table 7. Configuration RAM Controlled Latch/ Flip-Flop Operation

Function	Options
Common to All Latches/FFs in PFU	
Enable GSRN	GSRN enabled or has no effect on PFU latches/FFs.
Set Individually in Each Latch/FF in PFU	
Set/Reset Mode	Set or reset.
By Group (Latch/FF[3:0], Latch/FF[7:4], and FF[8])	
Clock Enable	CE or none.
LSR Control	LSR or none.
Clock Polarity	Noninverted or inverted.
Latch/FF Mode	Latch or FF.
LSR Operation	Asynchronous or synchronous.
Front-end Select*	Direct (DIN[7:0]) or from LUT (F[7:0]).
LSR Priority	Either LSR or CE has priority.

* Not available for FF[8].

Each PFU has two independent programmable clocks, clock enable CE[1:0], local set/reset LSR[1:0], and front-end data selects SEL[1:0]. When CE is disabled, each latch/FF retains its previous value when clocked. The clock enable, LSR, and SEL inputs can be inverted to be active-low.

Programmable Logic Cells (continued)

The set/reset operation of the latch/FF is controlled by two parameters: reset mode and set/reset value. When the GSRN and local set/reset (LSR) signals are not asserted, the latch/FF operates normally. The reset mode is used to select a synchronous or asynchronous LSR operation. If synchronous, LSR has the option to be enabled only if clock enable (CE) is active or for LSR to have priority over the clock enable input, thereby setting/resetting the FF independent of the state of the clock enable. The clock enable is supported on FFs, not latches. It is implemented by using a 2-input multiplexer on the FF input, with one input being the previous state of the FF and the other input being the new data applied to the FF. The select of this 2-input multiplexer is clock enable (CE), which selects either the new data or the previous state. When the clock enable is inactive, the FF output does not change when the clock edge arrives.

The GSRN signal is only asynchronous, and it sets/resets all latches/FFs in the FPGA based upon the set/reset configuration bit for each latch/FF. The set/reset value determines whether GSRN and LSR are set or reset inputs. The set/reset value is independent for each latch/FF. An option is available to disable the GSRN function per PFU after initial device configuration.

The latch/FF can be configured to have a data front-end select. Two data inputs are possible in the front-end select mode, with the SEL signal used to select which data input is used. The data input into each latch/FF is from the output of its associated LUT, F[7:0], or direct from DIN[7:0], bypassing the LUT. In the front-end data select mode, both signals are available to the latches/FFs.

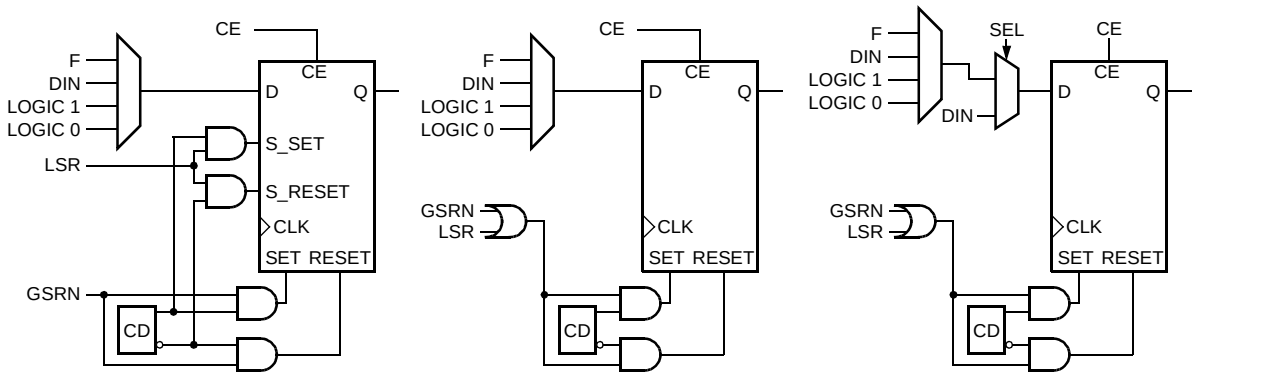
If either or both of these inputs is unused or is unavailable, the latch/FF data input can be tied to a logic 0 or logic 1 instead (the default is logic 0).

The latches/FFs can be configured in three basic modes:

- Local synchronous set/reset: the input into the PFU's LSR port is used to synchronously set or reset each latch/FF.
- Local asynchronous set/reset: the input into LSR asynchronously sets or resets each latch/FF.
- Latch/FF with front-end select, LSR either synchronous or asynchronous: the data select signal selects the input into the latches/FFs between the LUT output and direct data in.

For all three modes, each latch/FF can be independently programmed as either set or reset. Figure 20 provides the logic functionality of the front-end select, global set/reset, and local set/reset operations.

The ninth PFU FF, which is generally associated with registering the carry-out signal in ripple mode functions, can be used as a general-purpose FF. It is only an FF and is not capable of being configured as a latch. Because the ninth FF is not associated with an LUT, there is no front-end data select. The data input to the ninth FF is limited to the CIN input, logic 1, logic 0, or the carry-out in ripple and half-logic modes.



Key: CD = configuration data.

Figure 20. Latch/FF Set/Reset Configurations

Embedded Block RAM

The *ORCA* Series 4 devices complement the distributed PFU RAM with large blocks of memory macrocells. The memory is available in 512 words by 18 bits/word blocks with two write and two read ports. Two byte lane enables also operate with quad-port functionality. Additional logic has been incorporated for FIFO, multiplier, and CAM implementations. The RAM blocks are organized along the PLC rows and are added in proportion to the FPGA array sizes as shown in Table 8. The contents of the RAM blocks may be optionally initialized during FPGA configuration.

Table 8. *ORCA* Series 4— Available Embedded Block RAM

Device	Number of Blocks	Number of EBR Bits
OR4E2	8	74K
OR4E4	12	111K
OR4E6	16	148K
OR4E10	20	185K
OR4E14	24	222K

Each highly flexible 512 x 18 (quad-port, two read/two write) RAM block can be programmed by the user to meet their particular function. Each of the EBR configurations use the physical signals as shown in Table 9. Quad-port addressing permits simultaneous read and write operations.

The EBR ports are written synchronously on the positive edge of CKW. Synchronous read operations use the positive edge of CKR. Options are available to use synchronous read address registers and read output registers, or to bypass these registers and have the RAM read operate asynchronously.

EBR Features

Quad-Port Modes (Two Read/Two Write)

- 512 x 18 with optional built-in arbitration between write ports.
- 1024 x 18 built on two blocks with built-in decode logic for simplified implementation and increased speed.

Dual-Port Modes (One Read/One Write)

- One 256 x 36.
- One 1K x 9.
- Two 512 x 9 built in one EBR with two separate read, write clocks and enables for independent operation.
- Two RAMs with a user customized number of words whose sum is 512 (or less) by 18.

The joining of RAM blocks is supported to create wider and deeper memories. The adjacent routing interface provided by the CIBs allow the cascading of blocks together with minimal penalties due to routing delays.

FIFO Modes

FIFOs can be configured to 256, 512, or 1K depths and 36, 18, or 9 widths respectively or two-512 x 9 but also can be expanded using multiple blocks. FIFO works synchronously with the same read and write clock where the read port can be registered on the output or not registered. It can also be optionally configured asynchronously with different read and write clocks.

Integrated flags allow the user the ability to fully utilize the EBR for FIFO, without the need to dedicate an address for providing distinct full/empty status. There are four programmable flags provided for each FIFO: Empty, partially empty, full, and partially full FIFO status. The partially empty and partially full flags are programmable with the flexibility to program the flags to any value from the full or empty threshold. The programmed values can be set to a fixed value through the bit stream, or a dynamic value can be controlled by input pins of the EBR FIFO.

Multiplier Modes

The *ORCA* EBR supports two variations of multiplier functions. Constant coefficient MULTIPLY [KCM] mode will produce a 24-bit output of a fixed 8-bit constant multiply of a 16-bit number or a fixed 16-bit constant multiply of an 8-bit number. This KCM multiplies a constant times a 16- or 8-bit number and produces a product as a 24-bit result. The coefficient and multiplication tables are stored in memory. Both the input and outputs can be configured to be registered for pipelining. Both write ports are available during MULTIPLY mode so that the user logic can update and modify the coefficients for dynamic coefficient updates.

An 8 x 8 MULTIPLY mode is configurable to either a pipelined or combinatorial multiplier function of two 8-bit numbers. Two 8-bit operands are multiplied to yield a 16-bit product. The input and outputs can be registered in pipeline mode.

Embedded Block RAM (continued)

CAM Mode

The CAM block is a content address memory that provides fast address searches by receiving data input and returning addresses that contain the data. Implemented in each EBR are two 16-word x 8-bit CAM function blocks.

The CAM has three modes: single match, multiple match, and clear, which are all achieved in one clock cycle. In single-match mode, an 8-bit data input is internally decoded and reports a match when data is present in a particular RAM address. Its result is reported by a corresponding single address bit. In multiple match, the same occurs with the exception of multiple address lines report the match. Clear mode is used to clear the CAM contents in one clock cycle by erasing all locations.)

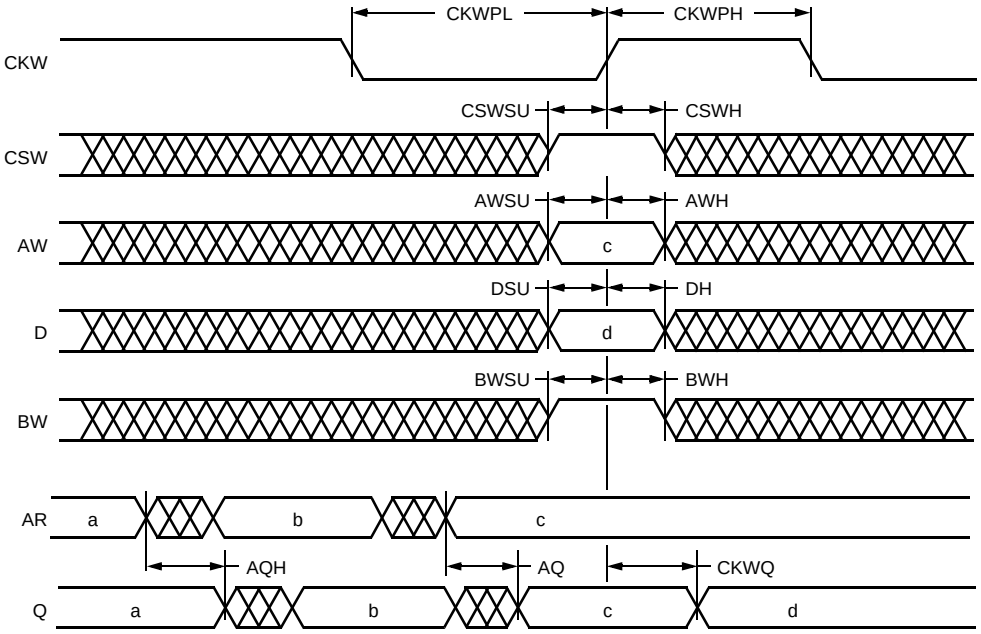
Arbitration logic is optionally programmed by the user to signal occurrences of data collisions as well as to block both ports from writing at the same time. The arbitration logic prioritizes PORT1. When utilizing the arbiter, the signal BUSY indicates data is being written to PORT1. This BUSY output signals PORT1 activity by driving a high output. The arbitration default is enabled; however, the user may disable the arbiter in configuration. If the arbiter is turned off, both ports could be written at the same time and the data would be corrupt. In this scenario, the BUSY signal will indicate a possible error.

There is also a user option which dedicates PORT 1 to communications to the system bus. In this mode, the user logic only has access to PORT0 and arbitration logic is enabled. The system bus utilizes the priority given to it by the arbiter; therefore, the system bus will always be able to write to the EBR.

Table 9. RAM Signals

Port Signals	I/O	Function
PORT 0		
AR0[#:0]	I	Address to be read.
AW0[#:0]	I	Address to be written.
BW0<1:0>	I	Byte-write enable. Byte = 8 bits + parity bit. <1> = bits[17, 15:9] <0> = bits[16, 7:0]
CKR0	I	Positive-edge asynchronous read clock.
CKW0	I	Positive-edge synchronous write clock.
CSR0	I	Enables read to output. Active-high.
CSW0	I	Enables write to occur. Active-high.
D [#:0]	I	Input data to be written to RAM.
Q [#:0]	O	Output data of memory contents at referenced address.
PORT 1		
AR1[#:0]	I	Address to be read.
AW1[#:0]	I	Address to be written.
BW1<1:0>	I	Byte-write enable. Byte = 8 bits + parity bit. <1> = bits[17, 15:9] <0> = bits[16, 7:0]
CKR1	I	Positive-edge asynchronous read clock.
CKW1	I	Positive-edge synchronous write clock.
CSR1	I	Enables read to output. Active-high.
CSW1	I	Enables write to occur. Active-high.
D [#:0]	I	Input data to be written to RAM.
Q [#:0]	O	Output data of memory contents at referenced address.
Control		
BUSY	O	PORT1 writing. Active-high.
RESET	I	Data output registers cleared. Memory contents unaffected. Active-low.

Embedded Block RAM (continued)



0308 (F)

Figure 21. EBR Read and Write Cycles with Write Through

Table 10. FIFO Signals

Port Signals	I/O	Function
AR(1:0)[9:0]	I	Programs FIFO flags. Used for partially empty flag size.
AW(1:0)[9:0]	I	Programs FIFO flags. Used for partially full flag size.
FF	O	Full flag.
PFF	O	Partially full flag.
PEF	O	Partially empty flag.
EF	O	Empty flag.
D0[17:0]	I	Data inputs for all configurations.
D1[17:0]	I	Data inputs for 256 x 36 configurations only.
CKW[0:1]	I	Positive-edge write port clock. Port 1 only used for 256 x 36 configurations.
CKR[0:1]	I	Positive-edge read port clock. Port 1 only used for 256 x 36 configurations.
CSW[1:0]	I	Active-high write enable. Port 1 only used for 256 x 36 configurations.
CSR[1:0]	I	Active-high read enable. Port 1 only used for 256 x 36 configurations.
RESET	I	Active-low. Resets FIFO pointers.
Q0[17:0]	O	Data outputs for all configurations.
Q1[17:0]	O	Data outputs for 256 x 36 configurations.

Embedded Block RAM (continued)

Table 11. Constant Multiplier Signals

Port Signals	I/O	Function
AR0[15:0]	I	Data input—operand.
AW(1:0)[8:0]	I	Address bits.
D(1:0)[17:0]	I	Data inputs to load memory or change coefficient.
CKW[0:1]	I	Positive-edge write port clock.
CKR[0:1]	I	Positive-edge read port clock. Used for synchronous multiply mode.
CSW[1:0]	I	Active-high write enable.
CSR[1:0]	I	Active-high read enable.
Q[23:0]	O	Data outputs—product result.

Table 12. 8 x 8 Multiplier Signals

Port Signals	I/O	Function
AR0[7:0]	I	Data input—multiplicand.
AR1[7:0]	I	Data input—multiplier.
AW(1:0)[8:0]	I	Address bits for memory.
D(1:0)[17:0]	I	Data inputs to load memory.
CKW[0:1]	I	Positive-edge write port clock.
CKR[0:1]	I	Positive-edge read port clock. Used for synchronous multiply mode.
CSW[1:0]	I	Active-high write enable.
CSR[1:0]	I	Active-high enables. For enabling address registers.
BW(1:0)[1:0]	I	Byte-lane write for loading memory.
Q[15:0]	O	Data outputs—product.

Table 13. CAM Signals

Port Signals	I/O	Function
AR(1:0)[7:0]	I	Data match.
AW(1:0)[8:0]	I	Data write.
D(1:0)[17]	I	Clear data active-high.
D(1:0)[16]	I	Single match active-high.
D(1:0)[3:0]	I	CAM address for data write.
CSW[1:0]	I	Active-high write enable. Enable for CAM data write.
CSR[1:0]	I	Active-high enable data registers. Enable for CAM data registers.
Q(1:0)15:0]	O	Decoded data outputs. 1 corresponds to a data match at that address location.

Routing Resources

The abundant routing resources of the Series 4 architecture are organized to route signals individually or as buses with related control signals. Both local and global signals utilize high-speed buffered and nonbuffered routes. One PLC segmented (x1), six PLC segmented (x6), and bused half-chip (xHL) routes are patterned together to provide high connectivity with fast software routing times and high-speed system performance.

x1 routes cross width of one PLC and provide local connectivity to PFU and SLIC inputs and outputs. x6 lines cross width of six PLCs and are unidirectional and buffered with taps in the middle and on the end. Segments allow connectivity to PFU/SLIC outputs (driven at one endpoint), other x6 lines (at endpoints), and x1 lines for access to PFU/SLIC inputs. xH lines run vertically and horizontally the distance of half the device and are useful for driving medium-/long-distance 3-state routing.

The improved routing resources offer great flexibility in moving signals to and from the logic core. This flexibility translates into an improved capability to route designs at the required speeds even when the I/O signals have been locked to specific pins.

Generally, the *ORCA* Foundry Development System is used to automatically route interconnections. Interactive routing with the *ORCA* Foundry design editor (EPIC) is also available for design optimization.

The routing resources consist of switching circuitry and metal interconnect segments. Generally, the metal lines which carry the signals are designated as routing segments. The switching circuitry connects the routing segments, providing one or more of three basic functions: signal switching, amplification, and isolation. A net running from a PFU or PIO output (source) to a PLC or PIO input (destination) consists of one or more routing segments, connected by switching circuitry called configurable interconnect points (CIPs).

Clock Distribution Network

Primary Clock Nets

The Series 4 FPGAs provide eight fully distributed global primary net routing resources. These eight primary nets can only drive clock signals. The scheme dedicates four of the eight resources to provide fast primary nets and four are available for general primary nets. The fast primary nets are targeted toward low-skew and small injection times while the general primary nets are also targeted toward low-skew but have more source location flexibility. Fast access to the global primary nets can be sourced from two pairs of pads located in the center of each side of the device, from the programmable PLLs, and dedicated network PLLs located in the corners, or from PLC logic. The I/O pads are dedicated in pairs for use of differential I/O clocking or single-ended I/O clock sources. However, if these pads are not needed to source the clock network, they can be utilized for general I/O. The clock routing scheme is patterned using vertical and horizontal routes which provide connectivity to all PLC columns.

Secondary Clock and Control Nets

Secondary spines provide flexible clocking and control signaling for local regions. Secondary nets usually have high fan-outs. The Series 4 utilizes a spine and branches that use additional x6 segments. This strategy provides a flexible connectivity and routes can be sourced from any I/O pin, all PLLs, or from PLC logic.

Edge Clock Nets

Routes are distributed around the edges and are available for every four PIOs (one per PIC). All PIOs and PLLs can drive the edge clocks and are used in conjunction with the secondary spines discussed above to drive the same edge clock signal into the internal logic array. The edge clocks provide fast injection to the PLC array and I/O registers. Many edge clock nets are provided on each side of the device.

Programmable Input/Output Cells

Programmable I/O

The Series 4 PIO addresses the demand for the flexibility to select I/O that meets system interface requirements. I/Os can be programmed in the same manner as in previous *ORCA* devices with the addition of new features that allow the user the flexibility to select new I/O types that support high-speed interfaces.

Each PIC contains up to four programmable I/O pads and are interfaced through a common interface block to the FPGA array. The PIO group is split into two pairs of I/O pads with each pair having independent clock enables, local set/reset, and global set/reset.

On the input side, each PIO contains a programmable latch/FF which enables very fast latching of data from any pad. The combination provides for very low setup requirements and zero hold times for signals coming on-chip. It may also be used to demultiplex an input signal, such as a multiplexed address/data signal, and register the signals without explicitly building a demultiplexer with a PFU.

Programmable Input/Output Cells

(continued)

On the output side of each PIO, an output from the PLC array can be routed to each output FF, and logic can be associated with each I/O pad. The output logic associated with each pad allows for multiplexing of output signals and other functions of two output signals.

The output FF, in combination with output signal multiplexing, is particularly useful for registering address signals to be multiplexed with data, allowing a full clock cycle for the data to propagate to the output. The output buffer signal can be inverted, and the 3-state control can be made active-high, active-low, or always enabled. In addition, this 3-state signal can be registered or nonregistered.

The Series 4 I/O logic has been enhanced to include modes for speed uplink and downlink capabilities. These modes are supported through shift register logic which divides down incoming data rates or multiplies up outgoing data rates. This new logic block also supports high-speed DDR mode requirements where data is clocked into and out of the I/O buffers on both edges of the clock.

The new programmable I/O cell allows designers to select I/Os that meet many new communication standards permitting the device to hook up directly without any external interface translation. They support traditional FPGA standards as well as high-speed single-ended and differential pair signaling (as shown in Table 14). Based on a programmable, bank-oriented I/O ring architecture, designs can be implemented using 3.3 V, 2.5 V, 1.8 V, and 1.5 V output levels.

Table 14. Series 4 Programmable I/O Standards

Standard	V_{DDIO} (V)	V_{REF} (V)	Interface Usage
LVTTL	3.3	NA	General purpose.
LVCMOS2	2.5	NA	
LVCMOS1.8	1.8	NA	
PCI	3.3	NA	PCI.
LVDS	2.5	NA	Point to point and multidrop backplanes, high noise immunity.
Bused-LVDS	2.5	NA	Network backplanes, high noise immunity, bus architecture backplanes.
LVPECL	2.5	NA	Network backplanes, differential 100 MHz+ clocking, optical transceiver, high-speed networking.
PECL	3.3	2.0	Backplanes.
GTL	3.3	0.8	Backplane or processor interface.
GTL+	3.3	1.0	
HSTL-Class I	1.5	0.75	High-speed SRAM and networking interfaces.
HTSL-Class III and IV	1.5	0.9	
STTL3-Class I and II	3.3	1.5	Synchronous DRAM interface.
SSTL2-Class I and II	2.5	1.25	

Note: Interfaces to DDR and *ZBT* memories are supported through the interface standards shown above.

Programmable Input/Output Cells

(continued)

The PIOs are located along the perimeter of the device. The PIO name is represented by a two-letter designation to indicate on which side of the device it is located followed by a number to indicate in which row or column it is located. The first letter, P, designates that the cell is a PIO and not a PLC. The second letter indicates the side of the array where the PIO is located. The four sides are left (L), right (R), top (T), and bottom (B). The individual I/O pad is indicated by a single letter (either A, B, C, or D) placed at the end of the PIO name. As an example, PL10A indicates a pad located on the left side of the array in the tenth row.

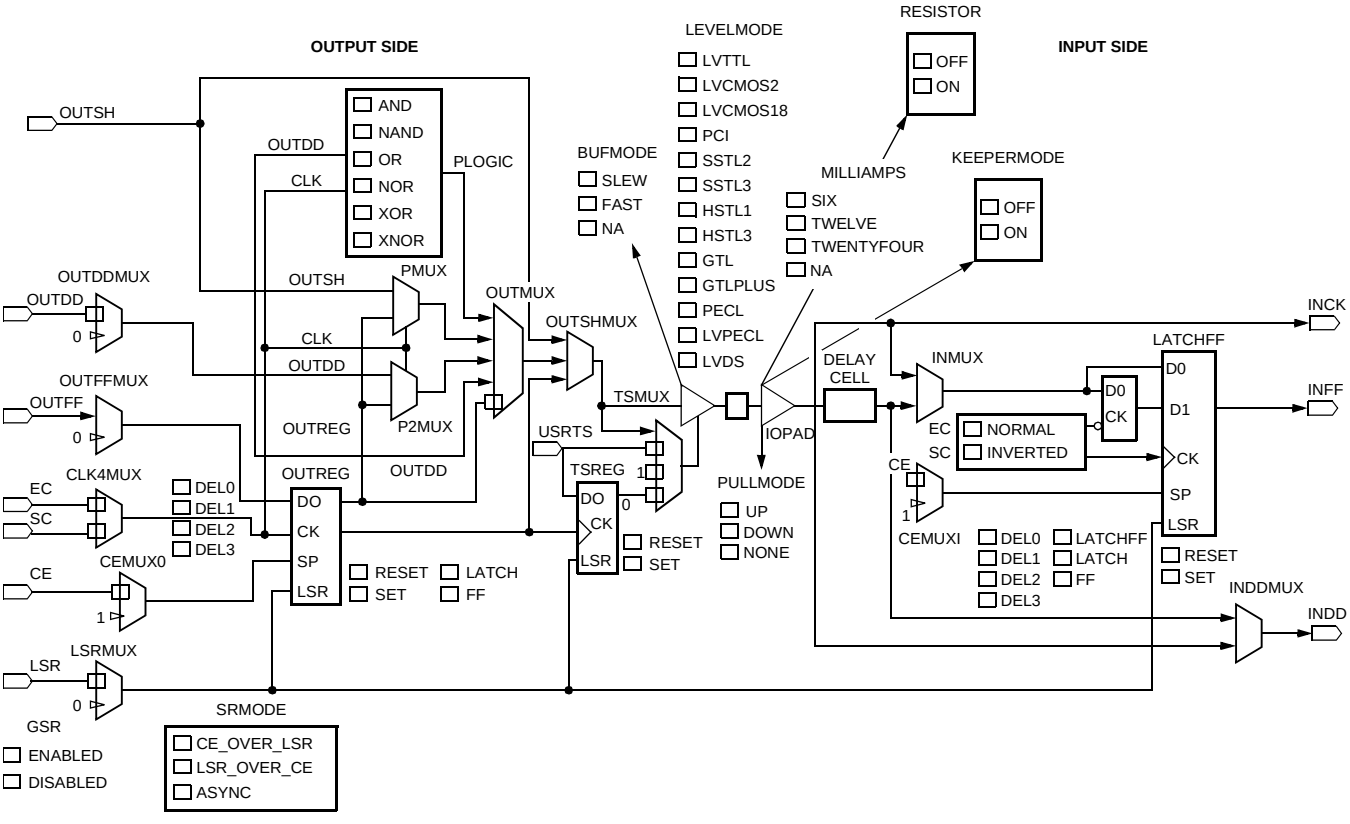
Each PIC interfaces to four bond pads and contains the necessary routing resources to provide an interface between I/O pads and the PLCs. Each PIC is composed of four programmable I/Os and significant routing resources. Each PIC contains input buffers, output buffers, routing resources, latches/FFs, and logic and can be configured as an input, output, or bidirectional I/O. Any PIO is capable of supporting the I/O standard listed in Table 12 and supporting DDR and *ZBT* specifications.

The I/O on the OR4Exxx Series devices allows compliance with PCI Local Bus (Rev. 2.2) 3.3 V signaling environments. The signaling environment used for each input buffer can be selected on a per-pin basis. The selection provides the appropriate I/O clamping diodes for PCI compliance.

The CIBs that bound the PIOs have significant local routing resources, similar to routing in the PLCs. This new routing increases the ability to fix user pinouts prior to placement and routing of a design and still maintain routability. The flexibility provided by the routing also provides for increased signal speed due to a greater variety of optimal signal paths.

Included in the PIO routing interface is a fast path from the input pins to the PFU logic. This feature allows for input signals to be very quickly processed by the SLIC decoder function and used on-chip or sent back off of the FPGA. Also, the Series 4 PIOs include latches and FFs and options for using fast, dedicated secondary, and edge clocks.

A diagram of a single PIO is shown in Figure 22, and Table 15 provides an overview of the programmable functions in an I/O cell.



5-9732(F)

Figure 22. Series 4 PIO Image from *ORCA* Foundry

Programmable Input/Output Cells

(continued)

Inputs

There are many major options on the PIO inputs that can be selected in the *ORCA* Foundry tools listed in Table 15. Inputs may have a pull-up or pull-down resistor selected on an input for signal stabilization and power management. A weak keeper circuit is also available on inputs. Input signals in a PIO are passed to CIB routing and/or a fast route into the clock routing system.

There is also a programmable delay available on the input. When enabled, this delay affects the INFF and INDD signals of each PIO, but not the clock input. The delay allows any signal to have a guaranteed zero hold time when input. This feature is discussed subsequently.

Inputs should have transition times of less than 500 ns and should not be left floating. If any pin is not used, it is 3-stated with an internal pull-up resistor enabled automatically after configuration.

Floating inputs increase power consumption, produce oscillations, and increase system noise. The inputs have a typical hysteresis of approximately 250 mV to reduce sensitivity to input noise. The PIC contains input circuitry that provides protection against latch-up and electrostatic discharge.

The other features of the PIO inputs relate to the latch/FF structure in the input path. In latch mode, the input signal is fed to a latch that is clocked by either the primary, secondary, or edge clock signal. The clock may be inverted or noninverted. There is also a local set/reset signal to the latch. The senses of these signals are also programmable and have the capability to enable or disable the global set/reset signal and select the set/reset priority. The same control signals may also be used to control the input latch/FF when it is configured as a FF instead of a latch, with the addition of another control signal used as a clock enable. The PIOs are paired together and have independent CE, set/reset, and GSRN control signals for the pair. Note that these control signals are paired to the same pair of pins used for differential signaling.

The input path is also capable of accepting data from any pad using a fast capture feature. This feature can be programmed as a latch or FF referenced to any clock. There are two options for zero-hold input capture in the PIO. If input delay mode is selected to delay the signal from the input pin, data can be either registered or latched with guaranteed zero-hold time in the PIO using a system clock. To further improve setup time, the fast zero-hold mode of the PIO input takes advantage of the latch/FF combination and sources the input FF data from a dedicated latch that is clocked by a fast edge clock from the dedicated clock pads or any local pad. The input FF is then driven by a primary clock sourced from a dedicated input pin designed for fast, low-skew operation at the I/Os. These dedicated pads are located in pairs in the center of each side of the array and if not utilized by the clock spine can be used as general user I/O. The clock inputs to both the dedicated fast capture latch and the input FF can also be driven by the on-chip PLLs.

The combination of input register capability provides for input signal demultiplexing without any additional resources such as for address and data arriving on the same pins. On the positive edge of the clock, the data would come from the pad to latch. The PIO input signal is sent to both the input latch and directly to INDD. The signal is latched on the falling edge of the clock and output to routing at INFF. The address and data are then both available at the rising edge of the clock. These signals may be registered or otherwise processed in the PLCs.

Programmable Input/Output Cells

(continued)

Table 15. PIO Options

Input	Option
Input Level	LVTTL, LVCMOS 2, LVCMOS 1.8, 3.3 V PCI Compliant.
Input Speed	Fast, Delayed.
Float Value	Pull-up, Pull-down, None.
Register Mode	Latch, FF, Fast Zero Hold FF, None (direct input).
Clock Sense	Inverted, Noninverted.
Input Selection	Input 1, Input 2, Clock Input.
Keeper Mode	On, Off.
LVDS Resistor	On, Off.
Output	**Option**
Output Drive Current	12 mA/6 mA or 6 mA/3 mA 24 mA/12 mA.
Output Function	Normal, Fast Open Drain.
Output Speed	Fast, Slew.
Output Source	FF Direct-out, General Routing.
Output Sense	Active-high, Active-low.
3-State Sense	Active-high, Active-low (3-state).
FF Clocking	Edge Clock, System Clock.
Clock Sense	Inverted, Noninverted.
Logic Options	See Table 17.
I/O Controls	**Option**
Clock Enable	Active-high, Active-low, Always Enabled.
Set/Reset Level	Active-high, Active-low, No Local Reset.
Set/Reset Type	Synchronous, Asynchronous.
Set/Reset Priority	CE over LSR, LSR over CE.
GSR Control	Enable GSR, Disable GSR.

Outputs

The PIO's output drivers for TTL/CMOS outputs have programmable drive capability and slew rates. Two propagation delays (fast, slewlim) are available on output drivers. There are three combinations of programmable drive currents (24 mA sink/12 mA source, 12 mA sink/6 mA, and 6 mA sink/3 mA source). At powerup, the output drivers are in slewlim mode and 12 mA sink/6 mA source. If an output is not to be driven in the selected configuration mode, it is 3-stated.

The output buffer signal can be inverted, and the 3-state control signal can be made active-high, active-low, or always enabled. In addition, this 3-state signal can be registered or nonregistered. Additionally, there is a fast, open-drain output option that directly connects the output signal to the 3-state control, allowing the output buffer to either drive to a logic 0 or 3-state, but never to drive to a logic 1.

The PIO has both input and output shift register capabilities. This ability allows the data rate to be reduced from the pad or increased to the pad by two or four times. The shift register block (SRB) is available in groups of four PIO. Both the input and output shift registers are controlled by the same clock and can operate at the same time at the same speed as long as the SRB is not connected to the same pads.The output control signals are similar to the input control signals in that they are per pair of PIOs.

Bus Hold

Each PIO can be programmed with a KEEPERMODE feature. This element is user programmed for bus hold requirements. This mode retains the last known state of a bus when the bus goes into 3-state. It prevents floating buses and saves system power.

PIO Register Control Signals

The PIO latches/FFs have various clock, clock enable (CE), local set/reset (LSR), and GSRN controls. Table 16 provides a summary of these control signals and their effect on the PIO latches/FFs. Note that all control signals are optionally invertible. The output control signals are similar to the input control signals in that they are per pair of PIOs.

Table 16. PIO Register Control Signals

Control Signal	Effect/Functionality
Edge Clock (ECLK)	Clocks input fast-capture latch; optionally clocks output FF, or 3-state FF.
System Clock (SCLK)	Clocks input latch/FF; optionally clocks output FF, or 3-state FF.
Clock Enable (CE)	Optionally enables/disables input FF (not available for input latch mode); optionally enables/disables output FF; separate CE inversion capability for input and output.
Local Set/Reset (LSR)	Option to disable; affects input latch/FF, output FF, and 3-state FF if enabled.
Global Set/Reset (GSRN)	Option to enable or disable per PIO (the input FF, output FF, and 3-state FF) after initial configuration.
Set/Reset Mode	The input latch/FF, output FF, and 3-state FF are individually set or reset by both the LSR and GSRN inputs.

Programmable Input/Output Cells

(continued)

The PIO output FF can perform output data multiplexing with no PLC resources required. This type of scheme is necessary for DDR applications which require data clocking out of the I/O on both edges of the clock. In this scheme, the output of OUTFF and OUTDD are serialized and shifted out on both the positive and negative edges of the clock using the shift registers.

The PIC logic block can also generate logic functions based on the signals on the OUTDD and CLK ports of the PIO. The functions are AND, NAND, OR, NOR, XOR, and XNOR. Table 17 is provided as a summary of the PIO logic options.

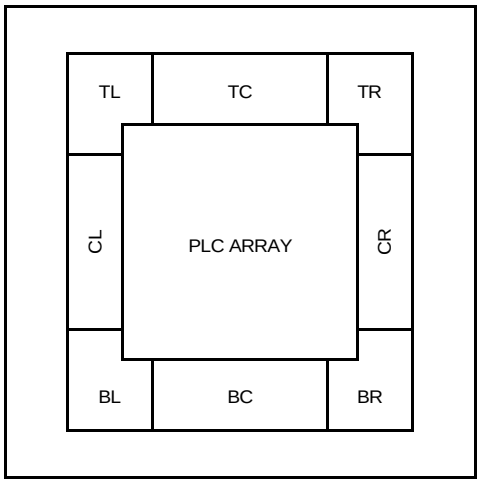
Table 17. PIO Logic Options

Option	Description
AND	Output logical AND of signals on OUTFF and clock.
NAND	Output logical NAND of signals on OUTFF and clock.
OR	Output logical OR of signals on OUTFF and clock.
NOR	Output logical NOR of signals on OUTFF and clock.
XOR	Output logical XOR of signals on OUTFF and clock.
XNOR	Output logical XNOR of signals on OUTFF and clock.

Flexible I/O features allow the user to select I/O to meet different high-speed interface requirements. These I/Os require different input references or supply voltages. The perimeter of the device is divided into groups of PIOs or buffer banks. For each bank, there is a separate $V_{DD}IO$. Every device is equally broken up into eight I/O banks. The $V_{DD}IO$ supplies the correct output voltage for a particular standard. The user must supply the appropriate power supply to the $V_{DD}IO$ pin. Within a bank, several I/O standards may be mixed as long as they use a common $V_{DD}IO$. Also, some interface standards require a specified threshold voltage known as V_{REF}. In these modes, where a particular V_{REF} is required, the device is automatically programmed to dedicate a pin for the appropriate V_{REF} which must be supplied by the user. The V_{REF} is dedicated exclusively to the bank and cannot be intermixed with other signaling requiring other V_{REF} voltages. However, pins not requiring V_{REF} can be mixed in the bank. The V_{REF} pad is then no longer available to the user for general use. See Table 14 for a list of the I/O standards supported.

Table 18. Compatible Mixed I/O Standards

$V_{DD}IO$ BANK Voltage	Compatible Standards
3.3 V	LVTTL, SSTL3-I, SSTL3-II, GTL, GTL+, PECL
2.5 V	LVCMOS2, SSTL2-I, SSTL2-II, LVDS, LVPECL
1.8 V	LVCMOS18
1.5 V	HSTL I, HSTL III, HSTL IV



0205(F).

Figure 23*. ORCA* High-Speed I/O Banks

High-Speed Memory Interfaces

PIO features allow high-speed interfaces to external SRAM and/or DRAM devices. Series 4 I/Os provide 200 MHz *ZBT* requirements when switching between write and read cycles. *ZBT* allows 100% use of bus cycles during back-to-back read/write and write/read cycles. However, this maximum utilization of the bus increases probability of bus contention when the interfaced devices attempt to drive the bus to opposite logic values. The LVTTL I/O interfaces directly with commercial *ZBT* SRAMs signaling and allows the versatility to program the FPGA drive strengths from 6 mA to 24 mA.

DDR allows data to be read or written on both the rising and the falling edge of the clock which delivers twice the bandwidth. QDR (quad data rate) are similar, but have separate read and write parts for over double the bandwidth. The DDR capability in the PIO also allows double the bandwidth per pin for generic transfer of data between two devices. DDR doubles the memory speed from SDRAMs without the need to increase clock frequency. The flexibility of the PIO allows 133 MHz/266 Mbits per second performance using the SSTL I/O features of the Series 4. All DDR interface functions are built into the PIO.

Programmable Input/Output Cells (continued)

LVDS I/O

The LVDS differential pair I/O standard allows for high-speed, low-voltage swing and low-power interfaces defined by industry standards: *ANSI**/TIA/*EIA*†-644 and *IEEE* 1596.3 SSI-LVDS. The general-purpose standard is supplied without the need for an input reference supply and uses a low switching voltage which translates to low ac power dissipation.

The *ORCA* LVDS I/O provides an integrated 100 Ω matching resistor used to provide a differential voltage across the inputs of the receiver. The on-chip integration provides termination of the LVDS receiver without the need of discrete external board resistors. The user has the programmable option to enable termination per receiver pair for point-to-point applications or, in multipoint interfaces, limit the use of termination to bused pairs. If the user chooses to terminate any differential receiver, a single LVDS_R pin is dedicated to connect a single 100 Ω resistor to VSS, which will provide a balance termination to all of the LVDS receiver pairs programmed to termination. See Table 20 for the LVDS termination pin location.

Table 19 provides the dc specifications for the *ORCA* LVDS solution.

Table 19. LVDS I/O Specifications

Parameter	Min	Typical	Max	Unit
Built-in Receiver Differential Input Resistor	95	100	105	Ω
Receiver Input Voltage	0.0	—	2.4	V
Differential Input Threshold	–100	—	100	mV
Output Common-mode Voltage	1.125	1.25	1.375	V
Input Common-mode Voltage	0.2	1.25	2.2	V

Table 20. LVDS Termination Pin

Dedicated Chip LVDS External Termination Pin (LVDS_R) Per Package		
BA352	**BC432**	**BM680**
AC3	AH29	AL1

PIO Downlink/Uplink

Each group of four PIO have access to an input/output shift register as shown in Figure 24. This feature allows high-speed input data to be divided down by 1/2 or 1/4, and output data can be multiplied by 2x or 4x its internal speed. Both the input and output shift can be programmed to operate at the same time. However, the same PIO cannot be used for both input and output shift registers at the same time.

For input shift mode, the data from INDD from the PIO is connected to the input shift register. The input data is divided down and is returned to the routing through the INSH nodes. In 4x mode, all the INSH nodes are used. 2x mode uses INSH4 and INSH3. Similarly, the output shift register brings data into the register from dedicated OUTSH nodes. 4x mode uses all the OUTSH signals. However, only OUTSH2 and OUTSH1 are used for 2x mode.

* *ANSI* is a registered trademark of American National Standards Institute, Inc.
† *EIA* is a registered trademark of Electronic Industries Association.

Programmable Input/Output Cells (continued)

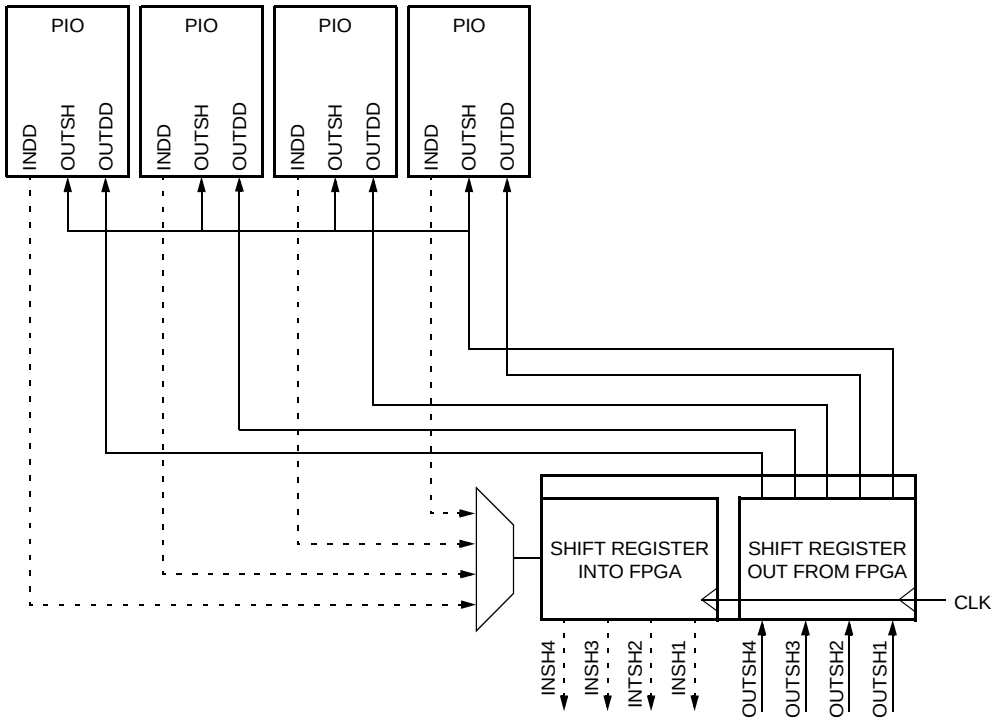


Figure 24. PIO Shift Register

Special Function Blocks

Internal Oscillator

The internal oscillator resides in the upper left corner of the FPGA array. It has output clock frequencies of 1.25 MHz and 10 MHz. The internal oscillator is the source of the internal CCLK used for configuration. It may also be used after configuration as a general-purpose clock signal.

Global Set/Reset (GSRN)

The GSRN logic resides in the lower-right corner of the FPGA. GSRN is an invertible (default active-low) signal that is used to reset all of the user-accessible latches/FFs on the device. GSRN is automatically asserted at powerup and during configuration of the device.

The timing of the release of GSRN at the end of configuration can be programmed in the start-up logic described below. Following configuration, GSRN may be connected to the $\overline{\text{RESET}}$ pin via dedicated routing, or it may be connected to any signal via normal routing. Within each PFU and PIO, individual FFs and latches can be programmed to either be set or reset when GSRN is asserted. Series 4 allows individual PFUs and PIOs to turn off the GSRN signal to its latches/FFs after configuration.

The $\overline{\text{RESET}}$ input pad has a special relationship to GSRN. During configuration, the $\overline{\text{RESET}}$ input pad always initiates a configuration abort, as described in the FPGA States of Operation section. After configuration, the GSRN can either be disabled (the default), directly connected to the $\overline{\text{RESET}}$ input pad, or sourced by a lower-right corner signal. If the $\overline{\text{RESET}}$ input pad is not used as a global reset after configuration, this pad can be used as a normal input pad.

Start-Up Logic

The start-up logic block can be configured to coordinate the relative timing of the release of GSRN, the activation of all user I/Os, and the assertion of the DONE signal at the end of configuration. If a start-up clock is used to time these events, the start-up clock can come from CCLK, or it can be routed into the start-up block using lower-right corner routing resources.

Special Function Blocks (continued)

Temperature Sensing

The built-in temperature-sensing diodes allow junction temperature to be measured during device operation. A physical pin (PTEMP) is dedicated for monitoring device junction temperature. PTEMP works by forcing a 10 µA current in the forward direction, and then measuring the resulting voltage. The voltage decreases with increasing temperature at approximately –1.69 mV/°C. A typical device with a 85 °C device temperature will measure approximately 630 mV.

Table 21. Dedicated Temperature Sensing

Dedicated Temperature Sensing Diode Pin Per Package		
BA352	**BC432**	**BM680**
AB3	AH31	AK4

Boundary-Scan

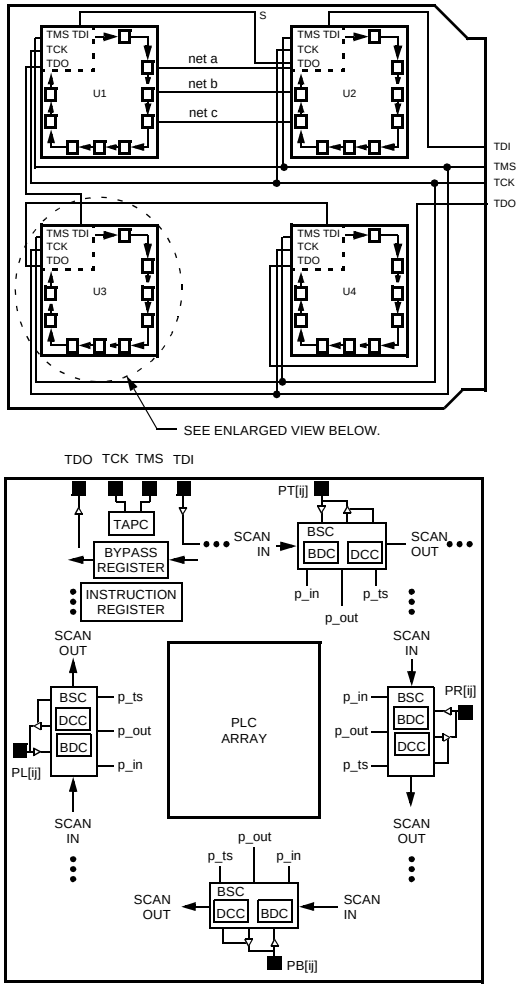
The *IEEE* standards 1149.1 and 1149.2 (*IEEE* Standard test access port and boundary-scan architecture) are implemented in the *ORCA* series of FPGAs. It allows users to efficiently test the interconnection between integrated circuits on a PCB as well as test the integrated circuit itself. The *IEEE* 1149 standard is a well-defined protocol that ensures interoperability among boundary-scan (BSCAN) equipped devices from different vendors.

Series 4 FPGAs are also compliant to *IEEE* standard 1532/D1. This standard for boundary-scan based in-system configuration of programmable devices provides a standardized programming access and methodology for FPGAs. A device, or set of devices, implementing this standard may be programmed, read back, erased verified, singly or concurrently, with a standard set of resources.

The *IEEE* 1149 standards define a test access port (TAP) that consists of a four-pin interface with an optional reset pin for boundary-scan testing of integrated circuits in a system. The *ORCA* Series FPGA provides four interface pins: test data in (TDI), test mode select (TMS), test clock (TCK), and test data out (TDO). The $\overline{\text{PRGM}}$ pin used to reconfigure the device also resets the boundary-scan logic.

The user test host serially loads test commands and test data into the FPGA through these pins to drive outputs and examine inputs. In the configuration shown in Figure 26, where boundary-scan is used to test ICs, test data is transmitted serially into TDI of the first BSCAN device (U1), through TDO/TDI connections between BSCAN devices (U2 and U3), and out TDO of the last BSCAN device (U4). In this configuration, the TMS and TCK signals are routed to all boundary-scan ICs in parallel so that all boundary-scan components operate in the same state. In other configurations, multiple scan paths are used instead of a single ring. When multiple scan paths are used, each ring is independently controlled by its own TMS and TCK signals.

Figure 26 provides a system interface for components used in the boundary-scan testing of PCBs. The three major components shown are the test host, boundary-scan support circuit, and the devices under test (DUTs). The DUTs shown here are *ORCA* Series FPGAs with dedicated boundary-scan circuitry. The test host is normally one of the following: automatic test equipment (ATE), a workstation, a PC, or a microprocessor.



5-5972(F)

Key: BSC = boundary-scan cell, BDC = bidirectional data cell, and DCC = data control cell.

Figure 25. Printed-Circuit Board with Boundary-Scan Circuitry

Special Function Blocks (continued)

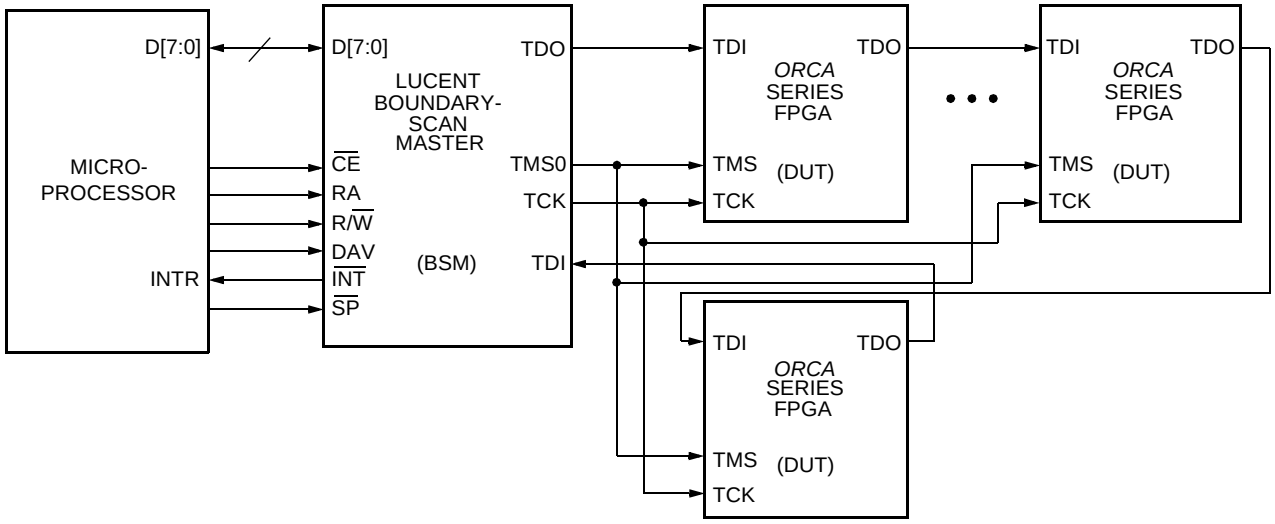


Figure 26. Boundary-Scan Interface

The boundary-scan support circuit shown in Figure 26 is the 497Aa boundary-scan master (BSM). The BSM off-loads tasks from the test host to increase test throughput. To interface between the test host and the DUTs, the BSM has a general MPI and provides parallel-to-serial/serial-to-parallel conversion, as well as three 8K data buffers. The BSM also increases test throughput with a dedicated automatic test-pattern generator and with compression of the test response with a signature analysis register. The PC-based boundary-scan test card/software allows a user to quickly prototype a boundary-scan test setup.

Boundary-Scan Instructions

The Series 4 boundary-scan circuitry includes ten *IEEE* 1149.1, 1149.2, and 1532/D1 instructions and six *ORCA*-defined instructions. These also include one *IEEE* 1149.3 optional instruction. A 6-bit wide instruction register supports all the instructions listed in Table 22. The BYPASS instruction passes data internally from TDI to TDO after being clocked by TCK.

Table 22. Boundary-Scan Instructions

Code	Instruction
000000	EXTEST
000001	SAMPLE
000011	PRELOAD
000100	RUNBIST
000101	IDCODE
000110	USERCODE
001000	ISC_ENABLE
001001	ISC_PROGRAM
001010	ISC_NOOP
001011	ISC_DISABLE
001101	ISC_PROGRAM_USERCODE
001110	ISC_READ
010001	PLC_SCAN_RING1
010010	PLC_SCAN_RING2
010011	PLC_SCAN_RING3
010100	RAM_WRITE
010101	RAM_READ
111111	BYPASS

Special Function Blocks (continued)

The external test (EXTEST) instruction allows the interconnections between ICs in a system to be tested for opens and stuck-at faults. If an EXTEST instruction is performed for the system shown in Figure 25, the connections between U1 and U2 (shown by nets a, b, and c) can be tested by driving a value onto the given nets from one device and then determining whether this same value is seen at the other device. This is determined by shifting 2 bits of data for each pin (one for the output value and one for the 3-state value) through the BSR until each one aligns to the appropriate pin. Then, based upon the value of the 3-state signal, either the I/O pad is driven to the value given in the BSR, or the BSR is updated with the input value from the I/O pad, which allows it to be shifted out TDO.

The SAMPLE and PRELOAD instructions are useful for system debugging and fault diagnosis by allowing the data at the FPGA's I/Os to be observed during normal operation or written during test operation. The data for all of the I/Os is captured simultaneously into the BSR, allowing them to be shifted-out TDO to the test host. Since each I/O buffer in the PIOs is bidirectional, two pieces of data are captured for each I/O pad: the value at the I/O pad and the value of the 3-state control signal. For preload operation, data is written from the BSR to all of the I/Os simultaneously.

There are six *ORCA*-defined instructions. The PLC scan rings 1, 2, and 3 (PSR1, PSR2, PSR3) allow user-defined internal scan paths using the PLC latches/FFs and routing interface. The RAM_Write Enable (RAM_W) instruction allows the user to serially configure the FPGA through TDI. The RAM_Read Enable (RAM_R) allows the user to read back RAM contents on TDO after configuration. The IDCODE instruction allows the user to capture a 32-bit identification code that is unique to each device and serially output it at TDO. The IDCODE format is shown in Table 23.

An optional *IEEE* 1149.3 instruction RUNBIST has been implemented. This instruction is used to invoke the built-in self-test (BIST) of regular structures like RAMs, ROMs, FIFOs, etc., and the surrounding RANDOM logic in the circuit.

Also implemented in Series 4 devices is the *IEEE* 1532/D1 standards for in-system configuration for programmable logic devices. Included are four mandatory and two optional instructions defined in the standards. ISC_ENABLE, ISC_PROGRAM, ISC_NOOP, and ISC_DISABLE are the four mandatory instructions. ISC_ENABLE initializes the devices for all subsequent ISC instructions. The ISC_PROGRAM instruction is similar to the RAM_WRITE instruction implemented in all *ORCA* devices where the user must monitor the PINITN pin for a high indicating the end of initialization and a successful configuration can be started. The ISC_PROGRAM instruction is used to program the configuration memory through a dedicated ISC_Pdata register. The ISC_NOOP instruction is used when programming multiple devices in parallel. During this mode, TDI and TDO behave like BYPASS. The data shifted through TDI is shifted out through TDO. However, the output pins remain in control of the BSR, unlike BYPASS where they are driven by the system logic. The ISC_DISABLE is used upon completion of the ISC programming. No new ISC instructions will be operable without another ISC_ENABLE instruction.

Optional 1532/D1 instructions include ISC_PROGRAM_USERCODE. When this instruction is loaded, the user shifts all 32 bits of a user-defined ID (LSB first) through TDI. This overwrites any ID previously loaded into the ID register. This ID can then be read back through the USERCODE instruction defined in *IEEE* 1149.2.

ISC_READ is similar to the *ORCA* RAM_Read instruction which allows the user to read back the configuration RAM contents serially out on TDO. Both must monitor the PDONE signal to determine weather or not configuration is completed. ISC_READ used a 1-bit register to synchronously read back data coming from the configuration memory. The readback data is clocked into the ISC_READ data register and then clocked out TDO on the falling edge or TCK.

Table 23. Series 4E Boundary-Scan Vendor-ID Codes

Device	Version (4-bit)	Part* (10-bit)	Family (6-bit)	Manufacturer (11-bit)	LSB (1-bit)
OR4E2	0000	0011100000	001000	00000011101	1
OR4E4	0000	0001010000	001000	00000011101	1
OR4E6	0000	0000110000	001000	00000011101	1
OR4E10	0000	0011110000	001000	00000011101	1
OR4E14	0000	0010001000	001000	00000011101	1

* PLC array size of FPGA, reverse bit order.

Note: Table assumes version 0.

Special Function Blocks (continued)

ORCA Boundary-Scan Circuitry

The *ORCA* Series boundary-scan circuitry includes a test access port controller (TAPC), instruction register (IR), boundary-scan register (BSR), and bypass register. It also includes circuitry to support the 18 predefined instructions.

Figure 27 shows a functional diagram of the boundary-scan circuitry that is implemented in the *ORCA* Series. The input pins' (TMS, TCK, and TDI) locations vary depending on the part, and the output pin is the dedicated TDO/RD_DATA output pad. Test data in (TDI) is the serial input data. Test mode select (TMS) controls the boundary-scan TAPC. Test clock (TCK) is the test clock on the board.

The BSR is a series connection of boundary-scan cells (BSCs) around the periphery of the IC. Each I/O pad on the FPGA, except for CCLK, DONE, and the boundary-scan pins (TCK, TDI, TMS, and TDO), is included in the BSR. The first BSC in the BSR (connected to TDI) is located in the first PIO I/O pad on the left of the top side of the FPGA (PTA PIO). The BSR proceeds clockwise around the top, right, bottom, and left sides of the array. The last BSC in the BSR (connected to TDO) is located on the top of the left side of the array (PL1D).

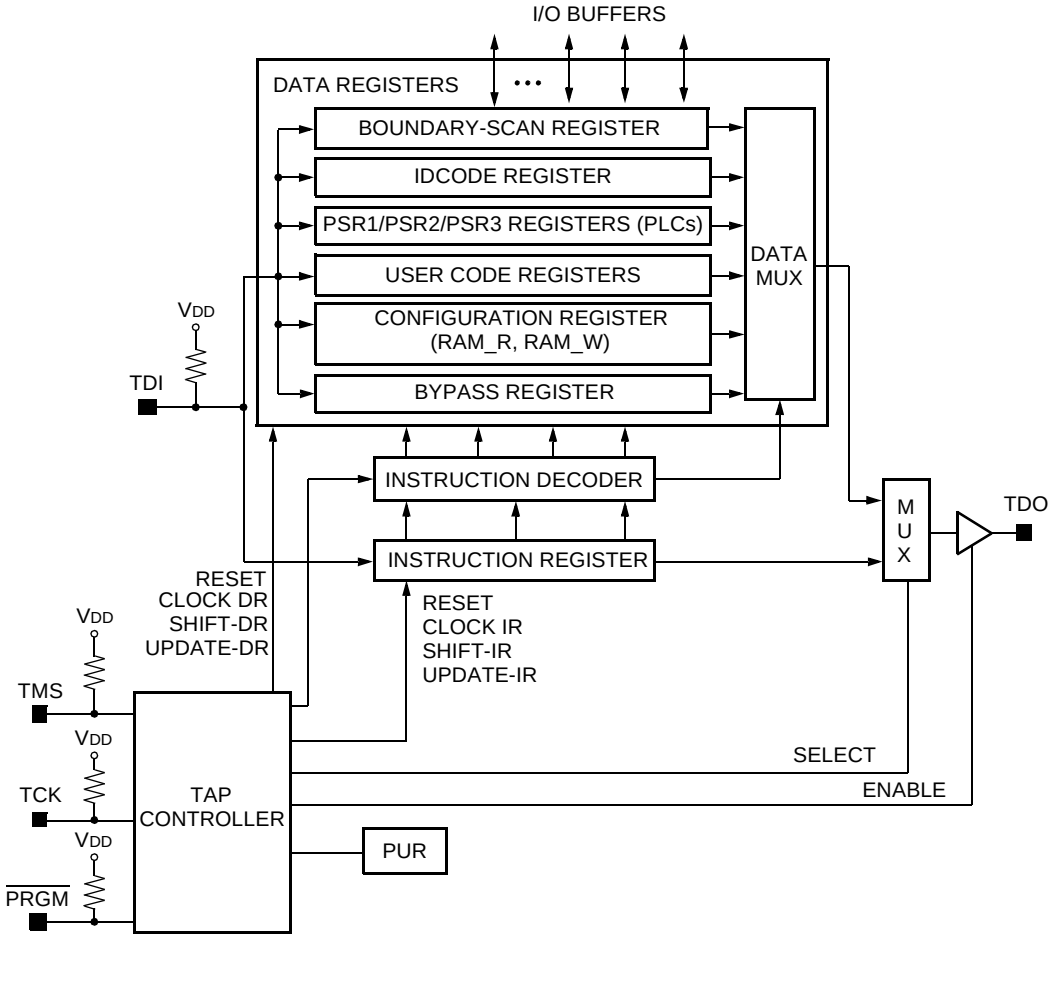
The bypass instruction uses a single FF, which resynchronizes test data that is not part of the current scan operation. In a bypass instruction, test data received on TDI is shifted out of the bypass register to TDO. Since the BSR (which requires a two FF delay for each pad) is bypassed, test throughput is increased when devices that are not part of a test operation are bypassed.

The boundary-scan logic is enabled before and during configuration. After configuration, a configuration option determines whether or not boundary-scan logic is used.

The 32-bit boundary-scan identification register contains the manufacturer's ID number, unique part number, and version (as described earlier). The identification register is the default source for data on TDO after RESET if the TAP controller selects the shift-data-register (SHIFT-DR) instruction. If boundary scan is not used, TMS, TDI, and TCK become user I/Os, and TDO is 3-stated or used in the readback operation.

An optional USERCODE is available. The USERCODE is a 32-bit value that the user can set during device configuration and can be written to and read from the FPGA via the boundary-scan logic.

Special Function Blocks (continued)



5-5768(F)

Figure 27. *ORCA* Series Boundary-Scan Circuitry Functional Diagram

ORCA Series TAP Controller

The *ORCA* Series TAP controller is a 1149 compatible TAPC. The 16 JTAG state assignments from the *IEEE* 1149 specification are used. The TAPC is controlled by TCK and TMS. The TAPC states are used for loading the IR to allow three basic functions in testing: providing test stimuli (Update-DR), providing test execution (Run-Test/Idle), and obtaining test responses (Capture-DR). The TAPC allows the test host to shift in and out both instructions and test data/results. The inputs and outputs of the TAPC are provided in the table below. The outputs are primarily the control signals to the instruction register and the data register.

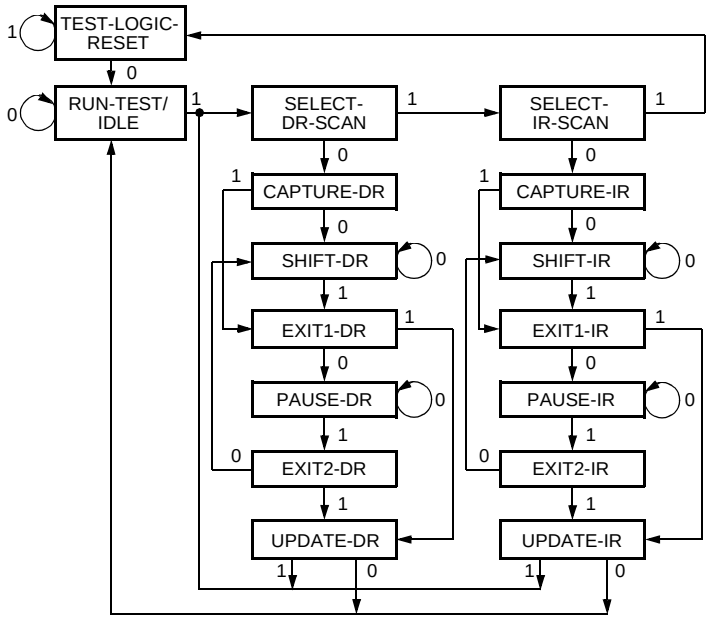
Table 24. TAP Controller Input/Outputs

Symbol	I/O	Function
TMS	I	Test Mode Select
TCK	I	Test Clock
PUR	I	Powerup Reset
$\overline{\text{PRGM}}$	I	BSCAN Reset
TRESET	O	Test Logic Reset
Select	O	Select IR (High); Select-DR (Low)
Enable	O	Test Data Out Enable
Capture-DR	O	Capture/Parallel Load-DR
Capture-IR	O	Capture/Parallel Load-IR
Shift-DR	O	Shift Data Register
Shift-IR	O	Shift Instruction Register
Update-DR	O	Update/Parallel Load-DR
Update-IR	O	Update/Parallel Load-IR

Special Function Blocks (continued)

The TAPC generates control signals that allow capture, shift, and update operations on the instruction and data registers. In the capture operation, data is loaded into the register. In the shift operation, the captured data is shifted out while new data is shifted in. In the update operation, either the instruction register is loaded for instruction decode, or the boundary-scan register is updated for control of outputs.

The test host generates a test by providing input into the *ORCA* Series TMS input synchronous with TCK. This sequences the TAPC through states in order to perform the desired function on the instruction register or a data register. Figure 28 provides a diagram of the state transitions for the TAPC. The next state is determined by the TMS input value.



5-5370(F)

Figure 28. TAP Controller State Transition Diagram

Boundary-Scan Cells

Figure 29 is a diagram of the boundary-scan cell (BSC) in the *ORCA* series PIOs. There are four BSCs in each PIO: one for each pad, except as noted above. The BSCs are connected serially to form the BSR. The BSC controls the functionality of the in, out, and 3-state signals for each pad.

The BSC allows the I/O to function in either the normal or test mode. Normal mode is defined as when an output buffer receives input from the PLC array and provides output at the pad or when an input buffer provides input from the pad to the PLC array. In the test mode, the BSC executes a boundary-scan operation, such as shifting in scan data from an upstream BSC in the BSR, providing test stimuli to the pad, capturing test data at the pad, etc.

The primary functions of the BSC are shifting scan data serially in the BSR and observing input (p_in), output (p_out), and 3-state (p_ts) signals at the pads. The BSC consists of two circuits: the bidirectional data cell is used to access the input and output data, and the direction control cell is used to access the 3-state value. Both cells consist of a FF used to shift scan data which feeds a FF to control the I/O buffer. The bidirectional data cell is connected serially to the direction control cell to form a boundary-scan shift register.

The TAPC signals (capture, update, shiftn, treset, and TCK) and the MODE signal control the operation of the BSC. The bidirectional data cell is also controlled by the high out/low in (HOLI) signal generated by the direction control cell. When HOLI is low, the bidirectional data cell receives input buffer data into the BSC. When HOLI is high, the BSC is loaded with functional data from the PLC.

Special Function Blocks (continued)

The MODE signal is generated from the decode of the instruction register. When the MODE signal is high (EXTEST), the scan data is propagated to the output buffer. When the MODE signal is low (BYPASS or SAMPLE), functional data from the FPGA's internal logic is propagated to the output buffer.

The boundary-scan description language (BSDL) is provided for each device in the *ORCA* Series of FPGAs on the *ORCA* Foundry CD. The BSDL is generated from a device profile, pinout, and other boundary-scan information.

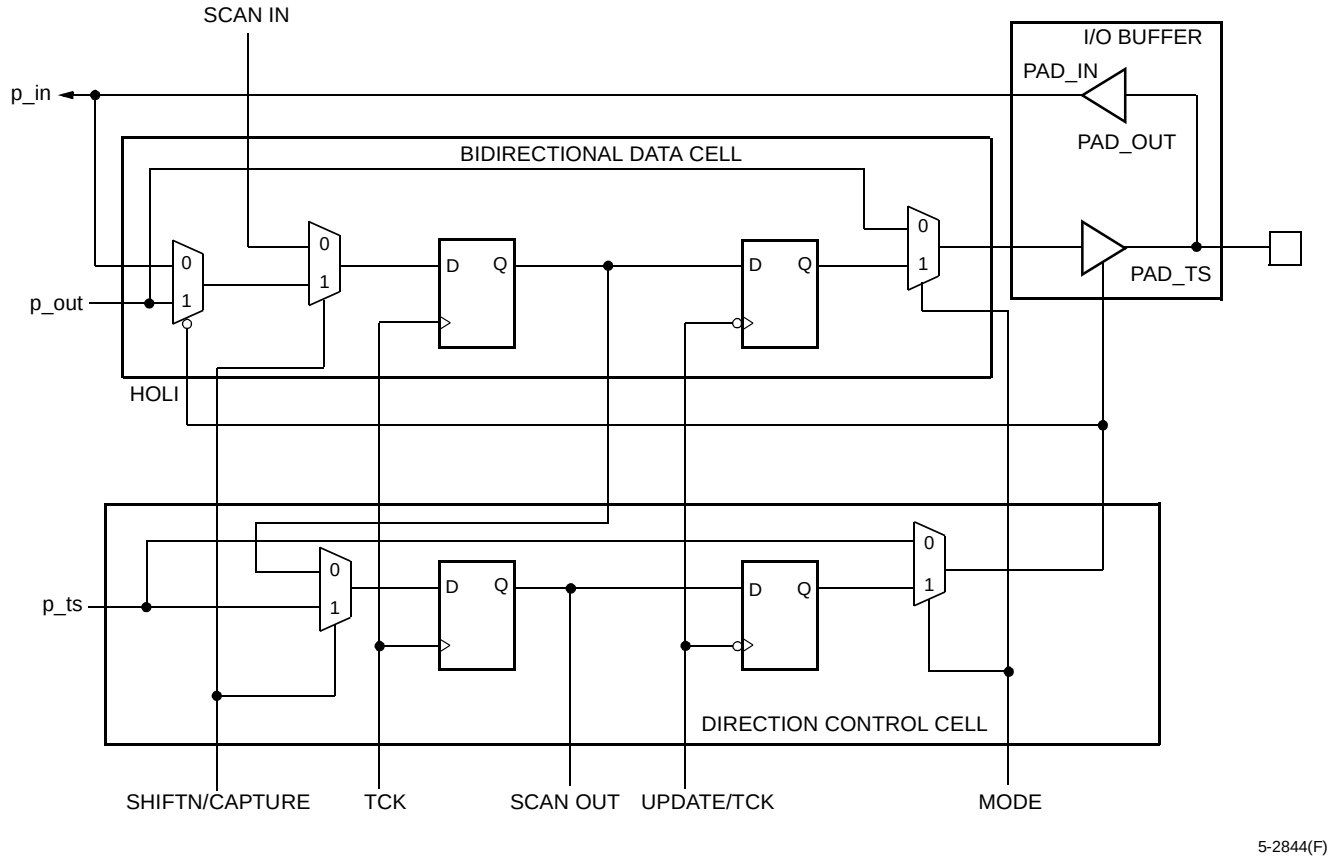


Figure 29. Boundary-Scan Cell

Boundary-Scan Timing

To ensure race-free operation, data changes on specific clock edges. The TMS and TDI inputs are clocked in on the rising edge of TCK, while changes on TDO occur on the falling edge of TCK. In the execution of an EXTEST instruction, parallel data is output from the BSR to the FPGA pads on the falling edge of TCK. The maximum frequency allowed for TCK is 10 MHz.

Figure 30 shows timing waveforms for an instruction scan operation. The diagram shows the use of TMS to sequence the TAPC through states. The test host (or BSM) changes data on the falling edge of TCK, and it is clocked into the DUT on the rising edge.

Special Function Blocks (continued)

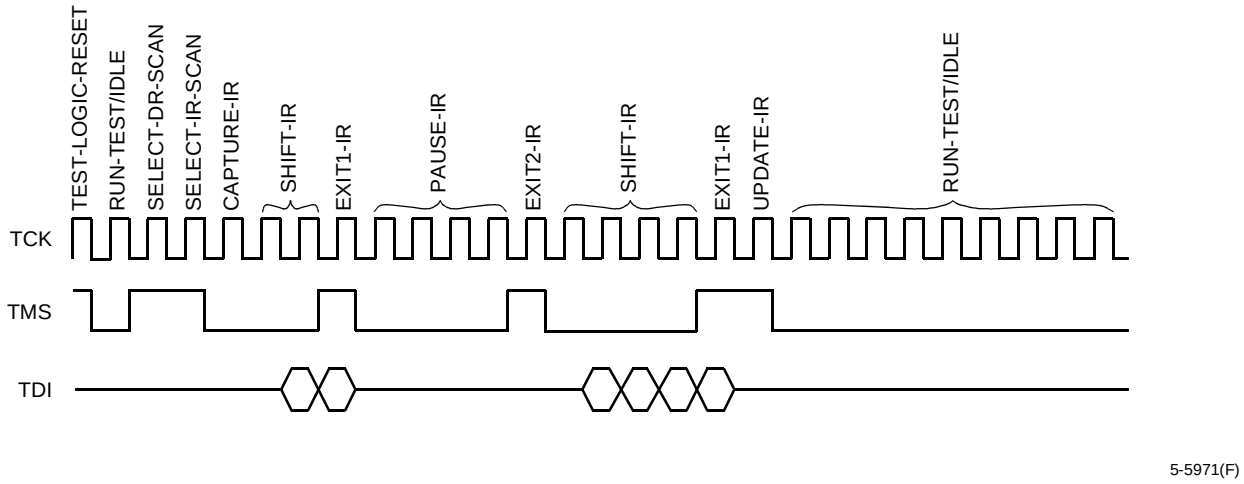


Figure 30. Instruction Register Scan Timing Diagram

Readback Logic

The readback logic can be enabled via a bit stream option or by instantiation of a library readback component.

Readback is used to read back the configuration data and, optionally, the state of all PFU and PIO FF outputs. A readback operation can be done while the FPGA is in normal system operation. The readback operation can be daisy-chained. To use readback, the user selects options in the bit stream generator in the *ORCA* Foundry development system.

Table 25 provides readback options selected in the bit stream generator tool. The table provides the number of times that the configuration data can be read back. This is intended primarily to give the user control over the security of the FPGA's configuration program. The user can prohibit readback (0), allow a single readback (1), or allow unrestricted readback (U).

Table 25. Readback Options

Option	Function
0	Prohibit Readback
1	Allow One Readback Only
U	Allow Unrestricted Number of Readbacks

Readback can be performed via the Series 4 MPI or by using dedicated FPGA readback controls. If the MPI is enabled, readback via the dedicated FPGA readback logic is disabled. Readback using the MPI is discussed in the MPI section.

The pins used for dedicated readback are readback data (RD_DATA), read configuration ($\overline{\text{RD\_CFG}}$), and configuration clock (CCLK). A readback operation is initiated by a high-to-low transition on $\overline{\text{RD\_CFG}}$. The $\overline{\text{RD\_CFG}}$ input must remain low during the readback operation. The readback operation can be restarted at frame 0 by driving the $\overline{\text{RD\_CFG}}$ pin high, applying at least two rising edges of CCLK, and then driving $\overline{\text{RD\_CFG}}$ low again. One bit of data is shifted out on RD_DATA at the rising edge of CCLK. The first start bit of the readback frame is transmitted out several cycles after the first rising edge of CCLK after $\overline{\text{RD\_CFG}}$ is input low (see the readback timing characteristics table in the timing characteristics section). To be certain of the start of the readback frame, the data can be monitored for the 01 frame start bit pair.

Special Function Blocks (continued)

Readback can be initiated at an address other than frame 0 via the new MPI control registers (see the Microprocessor Interface section for more information). In all cases, readback is performed at sequential addresses from the start address.

It should be noted that the RD_DATA output pin is also used as the dedicated boundary-scan output pin, TDO. If this pin is being used as TDO, the RD_DATA output from readback can be routed internally to any other pin desired. The $\overline{\text{RD\_CFG}}$ input pin is also used to control the global 3-state (TS_ALL) function. Before and during configuration, the TS_ALL signal is always driven by the $\overline{\text{RD\_CFG}}$ input and readback is disabled. After configuration, the selection as to whether this input drives the readback or global 3-state function is determined by a set of bit stream options. If used as the $\overline{\text{RD\_CFG}}$ input for readback, the internal TS_ALL input can be routed internally to be driven by any input pin.

The readback frame contains the configuration data and the state of the internal logic. During readback, the value of all registered PFU and PIO outputs can be captured. The following options are allowed when doing a capture of the PFU outputs:

- Do not capture data (the data written to the RAMs, usually 0, will be read back).
- Capture data upon entering readback.
- Capture data based upon a configurable signal internal to the FPGA. If this signal is tied to logic 0, capture RAMs are written continuously.
- Capture data on either options two or three above.

The readback frame has an identical format to that of the configuration data frame, which is discussed later in the Configuration Data Format section. If LUT memory is not used as RAM and there is no data capture, the readback data (not just the format) will be identical to the configuration data for the same frame. This eases a bitwise comparison between the configuration and readback data. The configuration header, including the length count field, is not part of the readback frame. The readback frame contains bits in locations not used in the configuration. These locations need to be masked out when comparing the configuration and readback frames. The development system optionally provides a readback bit stream to compare to readback data from the FPGA. Also note that if any of the LUTs are used as RAM and new data is written to them, these bits will not have the same values as the original configuration data frame either.

Global 3-State Control (TS_ALL)

To increase the testability of the *ORCA* Series FPGAs, the global 3-state function (TS_ALL) disables the device. The TS_ALL signal is driven from either an external pin or an internal signal. Before and during configuration, the TS_ALL signal is driven by the input pad $\overline{\text{RD\_CFG}}$. After configuration, the TS_ALL signal can be disabled, driven from the $\overline{\text{RD\_CFG}}$ input pad, or driven by a general routing signal in the upper right corner. Before configuration, TS_ALL is active-low; after configuration, the sense of TS_ALL can be inverted.

The following occur when TS_ALL is activated:

- All of the user I/O output buffers are 3-stated.
- The TDO/RD_DATA output buffer is 3-stated.
- The $\overline{\text{RD\_CFG}}$, $\overline{\text{RESET}}$, and $\overline{\text{PRGM}}$ input buffers remain active with a pull-up.
- The DONE output buffer is 3-stated, and the input buffer is pulled up.

Microprocessor Interface (MPI)

The Series 4 FPGAs have a dedicated synchronous MPI function block. The MPI is programmable to operate with *PowerPC* MPC860/MPC8260 series microprocessors. The pin listing is shown in Table 26. The MPI implements an 8-, 16-, or 32-bit interface with 4-bit parity to the host processor (*PowerPC*) that can be used for configuration and readback of the FPGA as well as for user-defined data processing and general monitoring of FPGA functions. In addition to dedicated-function registers, the MPI bridges to the *AMBA* embedded system bus through which the *PowerPC* bus master can access the FPGA configuration logic, EBR, and other user logic. There is also capability to interrupt the host processor either by a hard interrupt or by having the host processor poll the MPI and the embedded system bus.

The control portion of the MPI is available following powerup of the FPGA if the mode pins specify MPI mode, even if the FPGA is not yet configured. The width of the data port is selectable among 8-, 16-, or 32-bit and the parity bus can be 1-, 2-, or 4- bit. In configuration mode, the data bus width and parity are related to the state of the M[0:3] mode pins. For postconfiguration use, the MPI must be included in the configuration bit stream by using an MPI library element in your design from the *ORCA* macro library, or by setting the bit of the MPI configuration control register prior to the start of configuration. The user can also enable and disable the parity bus through the configuration bit stream. These pads can be used as general I/O when they are not needed for MPI use.

The *ORCA* FPGA is a memory-mapped peripheral to the *PowerPC* processor. The MPI interfaces to the user-programmable FPGA logic using the *AMBA* embedded system bus. The MPI has access to a series of addressable registers made accessible by the *AMBA* system bus that provide FPGA control and status, configuration and readback data transfer, FPGA device identification, and a dedicated user scratchpad register. All registers are 8 bits wide. The address map for these registers and the user-logic address space utilize the same registers as the *AMBA* embedded system bus. The internal *AMBA* bus is 32 bits wide and the proper transformation of 8-, 16-, or 32-bit data of the MPI is done when transferring data between the MPI and ESB.

Table 26. MPC 860 to *ORCA* MPI Interconnection

***PowerPC* Signal**	***ORCA* Pin Name**	**MPI I/O**	**Function**
D[*n*:0]	D[31:0]	I/O	8-, 16-, 32-bit data bus.
DP[*m*:0]	DP[3:0]	I/O	Selectable parity bus width from 1-, 2-, and 4-bit.
A[14:31]	A[17:0]	I	32-bit MPI address bus.
$\overline{\text{TS}}$	$\overline{\text{MPI\_STRB}}$	I	Transfer start signal.
$\overline{\text{BURST}}$	$\overline{\text{MPI\_BURST}}$	I	Active-low indicates burst transfer in-progress/high indicates current transfer not a burst.
—	$\overline{\text{CS0}}$	I	Active-low MPI select.
—	CS1	I	Active-high MPI select.
CLKOUT	MPI_CLK	I	*PowerPC* interface clock.
RD/$\overline{\text{WR}}$	MPI_RW	I	Read (high)/write (low) signal.
$\overline{\text{TA}}$	$\overline{\text{MPI\_ACK}}$	O	Active-low transfer acknowledge signal.
$\overline{\text{BDIP}}$	$\overline{\text{MPI\_BDIP}}$	O	Active-low burst transfer in progress signal indicates that the second beat in front of the current one is requested by the master. Negated before the burst transfer ends to abort the burst data phase.
$\overline{\text{Any of IRQ[7:0]}}$	$\overline{\text{MPI\_IRQ}}$	O	Active-low interrupt request signal.
$\overline{\text{TEA}}$	$\overline{\text{MPI\_TEA}}$	O	Active-low indicates MPI detects a bus error on the internal system bus for current transaction.
$\overline{\text{RETRY}}$	$\overline{\text{MPI\_RTRY}}$	O	Requests the MPC860 to relinquish the bus and retry the cycle.

Embedded System Bus (ESB)

Implemented using the open standard, on-chip bus *AMBA*-AHB 2.0 specification, the Series 4 devices connects all the FPGA elements together with a standardized bus framework. The ESB facilitates communication among MPI, configuration, EBRs, and user logic in all the generic FPGA devices. AHB serves the need for high-performance SoC as well as aligning with current synthesis design flows. Multiple bus masters optimize system performance by sharing resources between different bus masters such as the MPI and configuration logic. The wide data bus configuration of 32 bits with 4-bit parity supports the high-bandwidth of data-intensive applications of using the wide on-chip memory. *AMBA* enhances a reusable design methodology by defining a common backbone for IP modules.

The ESB is a synchronous bus that is driven by either the MPI clock, internal oscillator, CCLK (slave configuration modes), TCK (JTAG configuration modes), or by a user clock from routing. During initial configuration and reconfiguration, the bus clock is defaulted to the configuration clock. The postconfiguration clock source is set during configuration. The user has the ability to program several slaves through the user logic interface. Embedded block RAM also interfaces seamlessly to the AHB bus.

A single bus arbiter controls the traffic on the bus by ensuring that only one master has access to the bus at any time. The arbiter monitors a number of different requests to use the bus and decides which request is currently the highest priority. The configuration modes have the highest priority and overrides all normal user modes. Priority can be programmed between MPI and user logic at configuration in generic FPGAs. If no priority is set, a round-robin approach is used by granting the next requesting master in a rotating fixed order.

Several interfaces exist between the ESB and other FPGA elements. The MPI interface acts as a bridge between the external microprocessor bus and ESB. The MPI may have different clock domains than the ESB if the ESB clock is not sourced from the external microprocessor clock. Pipelined operation allows high-speed memory interface to the EBR and peripheral access without the requirement for additional cycles on the bus. Burst transfers allow optimal use of the memory interface by giving advance information of the nature of the transfers.

Table 27 is a listing of the ESB register file and brief descriptions. Table 28 shows the system interrupt registers, and Table 29 and Table 30 show the FPGA status and command registers, all with brief descriptions.

Embedded System Bus (ESB) (continued)

Table 27. Embedded System Bus/MPI Registers

Register	Byte	Read/Write	Initial Value	Description
00	03—00	RO	—	32-bit device ID
01	07—04	R/W	—	Scratchpad register
02	0B—08	R/W	—	Command register
03	0F—0C	RO	—	Status register
04	13	R/W	—	Interrupt enable register—MPI
	12	R/W	—	Interrupt enable register—USER
	11	R/W	—	Interrupt enable register—FPSC
	10	RO	—	Interrupt cause register
05	17—14	R/W	—	Readback address register (14 bits)
06	1B—18	RO	—	Readback data register
07	1F—1C	R/W	—	Configuration data register
08	23—20	RO	—	Reserved
09	27—24	RO	—	Bus error address register
0A	2B—28	RO	—	Interrupt vector 1 predefined by configuration bit stream
0B	2F—2C	RO	—	Interrupt vector 2 predefined by configuration bit stream
0C	33—30	RO	—	Interrupt vector 3 predefined by configuration bit stream
0D	37—34	RO	—	Interrupt vector 4 predefined by configuration bit stream
0E	3B—38	RO	—	Interrupt vector 5 predefined by configuration bit stream
0F	3F—3C	RO	—	Interrupt vector 6 predefined by configuration bit stream
10	43—40	—	—	Top-left PPLL control/status
11	47—44	—	—	Top-left HPLL control/status
14	53—50	—	—	Top-right PPLL control/status
18	63—60	—	—	Bottom-left PPLL control/status
19	67—64	—	—	Bottom-left HPLL control/status
1C	73—70	—	—	Bottom-right PPLL control/status

Table 28. Interrupt Register Space Assignments

Byte	Bit	Read/Write	Description
13	7—0	R/W	Interrupt enable register—MPI
12	7—0	R/W	Interrupt enable register—USER
11	7—0	R/W	Interrupt enable register—FPSC
10	Interrupt cause registers		
	7	RO	USER_IRQ_GENERAL;
	6	RO	USER_IRQ_SLAVE;
	5	RO	USER_IRQ_MASTER;
	4	RO	CFG_IRQ_DATA;
	3	RO	ERR_FLAG 1
	2	RO	MPI_IRQ
	1	RO	FPSC_IRQ_SLAVE;
	0	RO	FPSC_IRQ_MASTER

Embedded System Bus (ESB) (continued)

Table 29. Status Register Space Assignments

Byte	Bit	Read/Write	Description
0F	7:0	—	Reserved
0E	7:0	—	Reserved
OD	7	RO	Configuration write data acknowledge
	6	RO	Readback data ready
	5	RO	Unassigned (zero)
	4	RO	Unassigned (zero)
	3	RO	FPSC_BIT_ERR
	2	RO	RAM_BIT_ERR
	1	RO	Configuration write data size (1, 2, or 4 bytes)
	0	RO	Use with above for HSIZE[1:0] (byte, half-word, word)
0C	7	RO	Readback addresses out of range
	6	RO	Error response received by CFG from system bus
	5	RO	Error responses received by CFG from system bus
	4	RO	Unassigned (zero)
	3	RO	Unassigned (zero)
	2	RO	Unassigned (zero)
	1	RO	ERR_FLAG 1
	0	RO	ERR_FLAG 0

Table 30. Command Register Space Assignments

Byte	Bit	Description
08	7	Bus reset from MPI > drives HRESETn
	6	Bus reset from USER > drives HRESETn
	5	Bus reset from FPSC > drives HRESETn
	4	SYS_DAISY
	3	REPEAT_RDBK (Don't increment readback address.)
	2	MPI_USR_ENABLE
	1	Readback data size (1, 2, or 4 bytes)
	0	Use with above for HSIZE[1:0]
09	7	R/W SYS_GSR (GSR Input)
	6	SYS_RD_CFG (similar to FPGA pin RD_CFGN, but active-high)
	5	PRGM from MPI > (similar to FPGA pin, but active-high)
	4	PRGM from USER > (similar to FPGA pin, but active-high)
	3	PRGM from FPSC > (similar to FPGA pin, but active-high)
	2	LOCK from MPI
	1	LOCK from USER
	0	LOCK from FPSC
0A	7:0	Reserved
0B	7:0	Reserved

Phase-Locked Loops

There are eight PLLs available to perform many clock modification and clock conditioning functions on the Series 4 FPGAs. Six of the PLLs are programmable allowing the user the flexibility to configure the PLL to manipulate the frequency, phase, and duty cycle of a clock signal. Four of the programmable PLLs are capable of manipulating and conditioning clocks from 20 MHz to 200 MHz and two others are capable of manipulating and conditioning clocks from 60 MHz to 420 MHz. Frequencies can be adjusted from 1/8x to 8x the input clock frequency. Each programmable PLL provides two outputs that have different multiplication factors with the same phase relationships. Duty cycles and phase delays can be adjusted in 12.5% of the clock period increments. An input buffer delay compensation mode is available for phase delay. Each PPLL provides two outputs (MCLK, NCLK) that can have programmable (12.5% steps) phase differences.

The PPLLs can be utilized to eliminate skew between the clock input pad and the internal clock inputs across the entire device. The PPLLs can drive onto the primary, secondary, and edge clock networks inside the FPGA. Each PPLL can take a clock input from the dedicated pad or differential pair of pads in its corner or from general routing resources.

Functionality of the PPLLs is programmed during operation through a read/write interface to the internal system bus command and status registers or via the configuration bit stream. There is also a PLL output signal, LOCK, that indicates a stable output clock state. Unlike Series 3, this signal does not have to be intergrated before use.

Table 31. PPLL Specifications

Parameter		Min	Nom	Max	Unit
V_{DD}1.5		1.425	1.5	1.575	V
V_{DD}3.3		3.0	3.3	3.6	V
Operating Temp		–40	25	125	°C
Input Clock Voltage		1.425	1.5	1.575	V
Output Clock Voltage		1.425	1.5	1.575	V
Input Clock Frequency (no division)	PPLL	20	—	200	MHz
	HPPLL	60	—	420	
Output Clock Frequency	PPLL	20	—	200	MHz
	HPPLL	60	—	420	
Input Duty Cycle Tolerance		30	—	70	%
Output Duty Cycle		45	50	55	%
dc Power		—	28	—	mW
Total On Current		—	8.5	—	mA
Total Off Current		—	30	—	pA
Cycle to Cycle Jitter (p-p)		—	<0.02	—	UIp-p
Lock Time		—	<50	—	µs
Frequency Multiplication		1x, 2x, 3x, 4x, 5x, 6x, 7x, 8x,			—
Frequency Division		1/8, 1/7, 1/6, 1/5, 1/4, 1/3, 1/2			—
Duty Cycle Adjust of Output Clock		12.5, 25, 37.5, 50, 62.5, 75, 87.5			%
Delay Adjust of Output Clock		0, 12.5, 25, 37.5, 50, 62.5, 75, 87.5			%
Phase Shift Between MCLK & NCLK		0, 45, 90, 135, 180, 225, 270, 315			degree

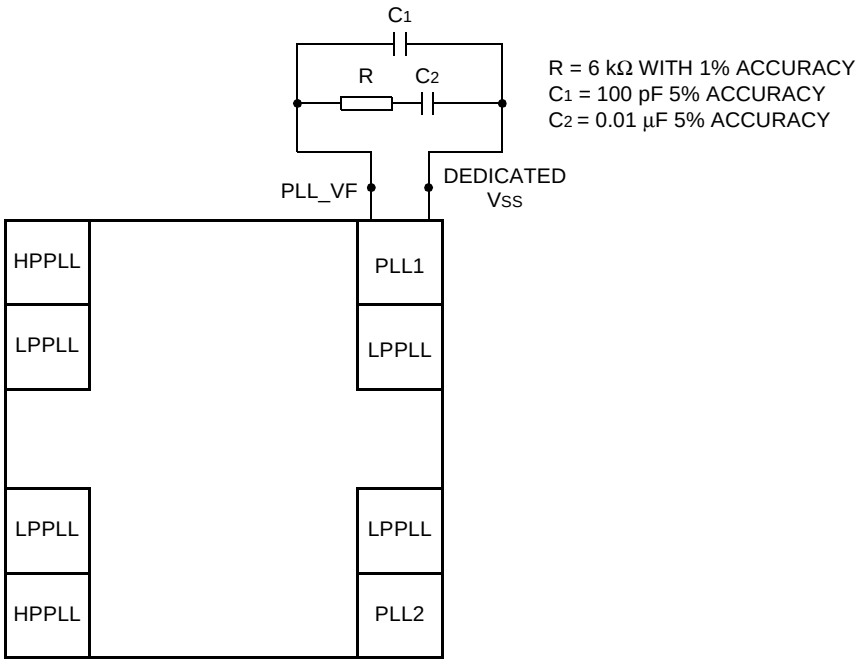
Additional highly tuned and characterized dedicated phase-locked loops (DPLLs) are included to ease system designs. These DPLLs meet ITU-T G.811 primary clocking specifications and enable system designers to target very tightly specified clock conditioning not available in the universal PPLLs. DPLLs are targeted to low-speed networking DS1 and E1 and high-speed SONET/SDH networking STS-3 and STM-1 systems.

Phase-Locked Loops (continued)

Table 32. DPLL DS-1/E-1 Specifications

Parameter	Min	Nom	Max	Unit
V_{DD}1.5	1.425	1.5	1.575	V
V_{DD}3.3	3.0	3.3	3.6	V
Operating Temp	–40	25	125	°C
Input Clock Voltage	1.425	1.5	1.575	V
Output Clock Voltage	1.425	1.5	1.575	V
Input Clock Frequency	1.0	—	2.5	MHz
Output Clock Frequency	—	1.544	—	MHz
	—	2.048	—	
Input Duty Cycle Tolerance	30	—	70	%
Output Duty Cycle	47	50	53	%
dc Power	—	20	—	mW
Total On Current	—	2.5	—	mA
Total Off Current	—	40	—	pA
Cycle to Cycle Jitter (p-p)	0.015 at 1.544 MHz			UIp-p
	0.05 at 2.048 MHz			
Lock Time	—	<1200	—	µs

A dedicated pin PLL_VF is needed for externally connecting a low-pass filter circuit, as shown in Table 33. This provides the specified DS–1/E–1 PLL operating condition.



1001(F).

Figure 31. PLL_VF External Requirements

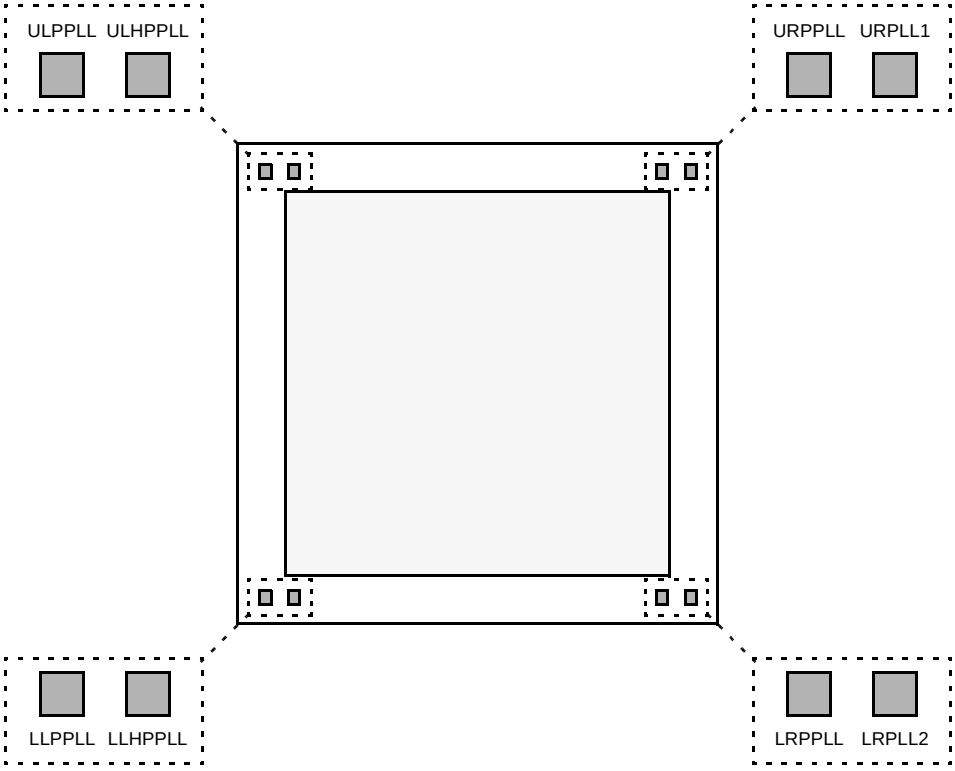
Phase-Locked Loops (continued)

Table 33. Dedicated Pin Per Package

Dedicated PLL_VF Pin Per Package		
BA352	**BC432**	**BM680**
B24	C4	D30

Table 34. STS-3/STM-1 DPLL Specifications

Parameter	Min	Nom	Max	Unit
Input Clock Frequency	—	155.52	—	MHz
Output Clock Frequency	—	155.52	—	MHz
Input Duty Cycle Tolerance	30	—	70	%
Output Duty Cycle	47	50	53	%
dc Power	—	50	—	mW
Total On Current	—	2.4	—	mA
Total Off Current	—	30	—	pA
Cycle to Cycle Jitter (p-p)		0.02		UIp-p
Lock Time	—	<50	—	µs



0045(F)

Figure 32. PLL Naming Scheme

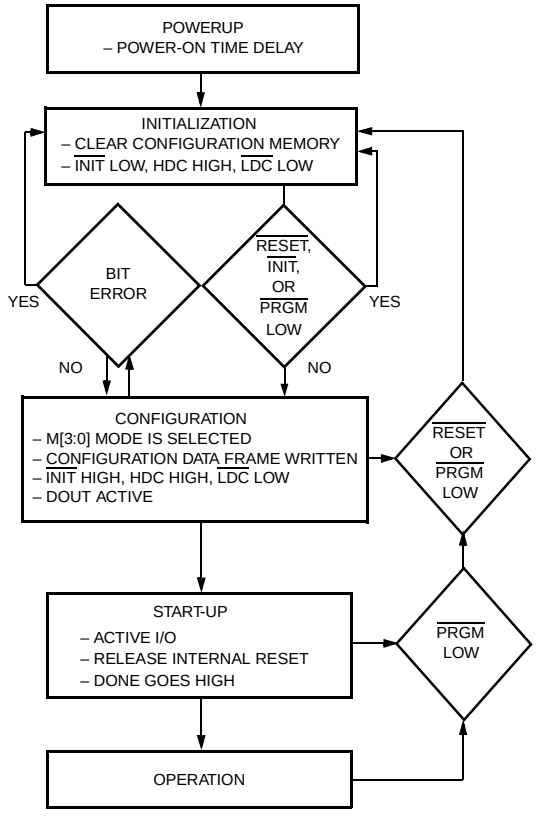
Phase-Locked Loops (continued)

Table 35. Phase-Lock Loops Index

Name	Description
[UL][LL][UR][LR]PPLL	Universal user-programmable PLL (20 MHz—200 MHz)
[UL][LL]HPPLL	Universal user-programmable PLL (60 MHz—420 MHz)
URPLL1	DS-1/E-1 dedicated PLL
LRPLL2	STS-1/STM-1 dedicated PLL

FPGA States of Operation

Prior to becoming operational, the FPGA goes through a sequence of states, including initialization, configuration, and start-up. Figure 33 outlines these three states.



5-4529(F).

Figure 33. FPGA States of Operation

FPGA States of Operation (continued)

Initialization

Upon powerup, the device goes through an initialization process. First, an internal power-on-reset circuit is triggered when power is applied. Dedicated power pins called $V_{DD}33$ are used by the configuration logic. When $V_{DD}33$ reaches the voltage at which portions of the FPGA begin to operate (2.0 V), the I/Os are configured based on the configuration mode, as determined by the mode select inputs M[2:0]. A time-out delay is initiated when $V_{DD}33$ reaches between 2.7 V to 3.0 V to allow the power supply voltage to stabilize. The $\overline{\text{INIT}}$ and DONE outputs are low. At powerup, if $V_{DD}33$ does not rise from 2.0 V to $V_{DD}33$ in less than 25 ms, the user should delay configuration by inputting a low into $\overline{\text{INIT}}$, $\overline{\text{PRGM}}$, or $\overline{\text{RESET}}$ until $V_{DD}33$ is greater than the recommended minimum operating voltage.

At the end of initialization, the default configuration option is that the configuration RAM is written to a low state. This prevents shorts prior to configuration. As a configuration option, after the first configuration (i.e., at reconfiguration), the user can reconfigure without clearing the internal configuration RAM first. The active-low, open-drain initialization signal $\overline{\text{INIT}}$ is released and must be pulled high by an external resistor when initialization is complete. To synchronize the configuration of multiple FPGAs, one or more $\overline{\text{INIT}}$ pins should be wire-ANDed. If $\overline{\text{INIT}}$ is held low by one or more FPGAs or an external device, the FPGA remains in the initialization state. $\overline{\text{INIT}}$ can be used to signal that the FPGAs are not yet initialized. After $\overline{\text{INIT}}$ goes high for two internal clock cycles, the mode lines (M[3:0]) are sampled, and the FPGA enters the configuration state.

The high during configuration (HDC), low during configuration ($\overline{\text{LDC}}$), and DONE signals are active outputs in the FPGA's initialization and configuration states. HDC, $\overline{\text{LDC}}$, and DONE can be used to provide control of external logic signals such as reset, bus enable, or PROM enable during configuration. For parallel master configuration modes, these signals provide PROM enable control and allow the data pins to be shared with user logic signals.

If configuration has begun, an assertion of $\overline{\text{RESET}}$ or $\overline{\text{PRGM}}$ initiates an abort, returning the FPGA to the initialization state. The $\overline{\text{PRGM}}$ and $\overline{\text{RESET}}$ pins must be pulled back high before the FPGA will enter the configuration state. During the start-up and operating states, only the assertion of $\overline{\text{PRGM}}$ causes a reconfiguration.

In the master configuration modes, the FPGA is the source of configuration clock (CCLK). In this mode, the initialization state is extended to ensure that, in daisy-chain operation, all daisy-chained slave devices are ready. Independent of differences in clock rates, master mode devices remain in the initialization state an additional six internal clock cycles after $\overline{\text{INIT}}$ goes high.

When configuration is initiated, a counter in the FPGA is set to 0 and begins to count configuration clock cycles applied to the FPGA. As each configuration data frame is supplied to the FPGA, it is internally assembled into data words. Each data word is loaded into the internal configuration memory. The configuration loading process is complete when the internal length count equals the loaded length count in the length count field, and the required end of configuration frame is written.

During configuration, the PIO and PLC latches/FFs are held set/reset and the internal BIDI buffers are 3-stated. The combinatorial logic begins to function as the FPGA is configured. Figure 34 shows the general waveform of the initialization, configuration, and start-up states.

Configuration

The *ORCA* Series FPGA functionality is determined by the state of internal configuration RAM. This configuration RAM can be loaded in a number of different modes. In these configuration modes, the FPGA can act as a master or a slave of other devices in the system. The decision as to which configuration mode to use is a system design issue. Configuration is discussed in detail, including the configuration data format and the configuration modes used to load the configuration data in the FPGA, following a description of the start-up state.

Start-Up

After configuration, the FPGA enters the start-up phase. This phase is the transition between the configuration and operational states and begins when the number of CCLKs received after $\overline{\text{INIT}}$ goes high is equal to the value of the length count field in the configuration frame and when the end of configuration frame has been written. The system design issue in the start-up phase is to ensure the user I/Os become active without inadvertently activating devices in the system or causing bus contention. A second system design concern is the timing of the release of global set/reset of the PLC latches/FFs.

FPGA States of Operation (continued)

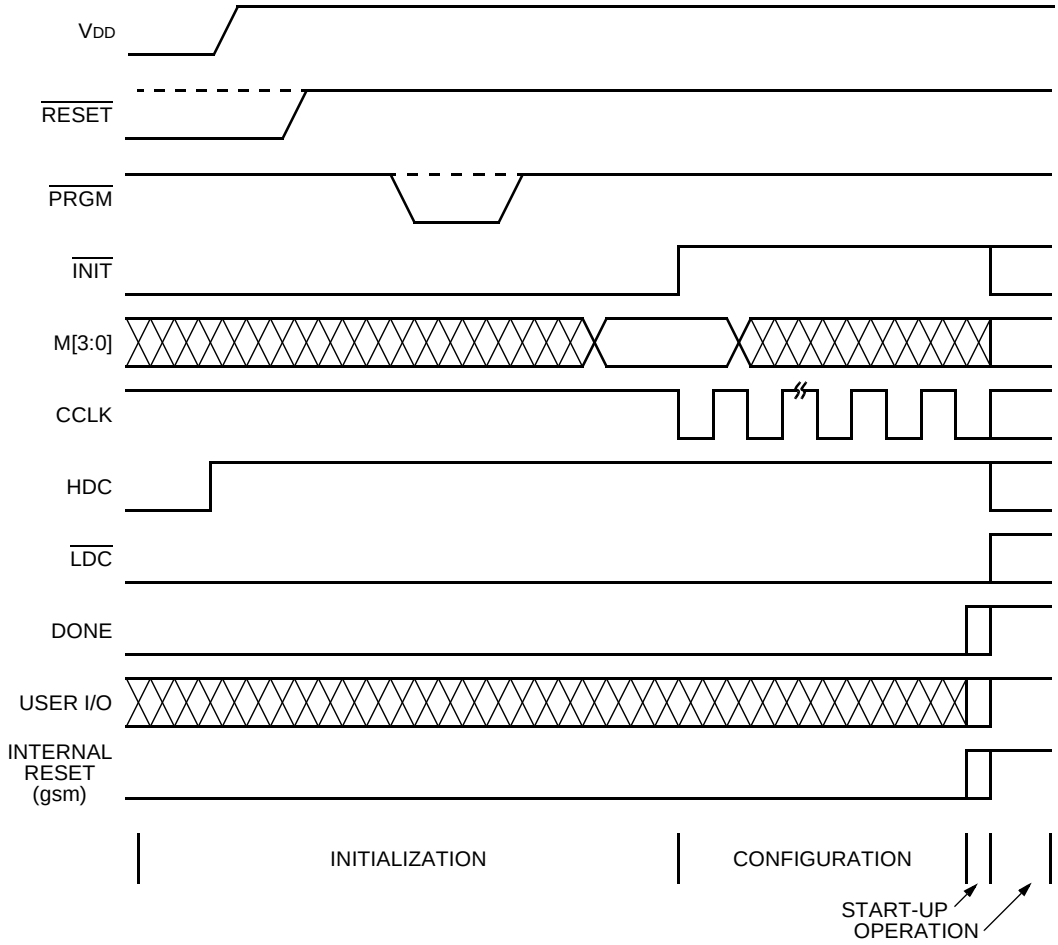


Figure 34. Initialization/Configuration/Start-Up Waveforms

FPGA States of Operation (continued)

There are configuration options that control the relative timing of three events: DONE going high, release of the set/reset of internal FFs, and user I/Os becoming active. Figure 35 shows the start-up timing for *ORCA* FPGAs. The system designer determines the relative timing of the I/Os becoming active, DONE going high, and the release of the set/reset of internal FFs. In the *ORCA* Series FPGA, the three events can occur in any arbitrary sequence. This means that they can occur before or after each other, or they can occur simultaneously.

There are four main start-up modes: CCLK_NOSYNC, CCLK_SYNC, UCLK_NOSYNC, and UCLK_SYNC. The only difference between the modes starting with CCLK and those starting with UCLK is that for the UCLK modes, a user clock must be supplied to the start-up logic. The timing of start-up events is then based upon this user clock, rather than CCLK. The difference between the SYNC and NOSYNC modes is that for SYNC mode, the timing of two of the start-up events, release of the set/reset of internal FFs, and the I/Os becoming active is triggered by the rise of the external DONE pin followed by a variable number of rising clock edges (either CCLK or UCLK). For the NOSYNC mode, the timing of these two events is based only on either CCLK or UCLK.

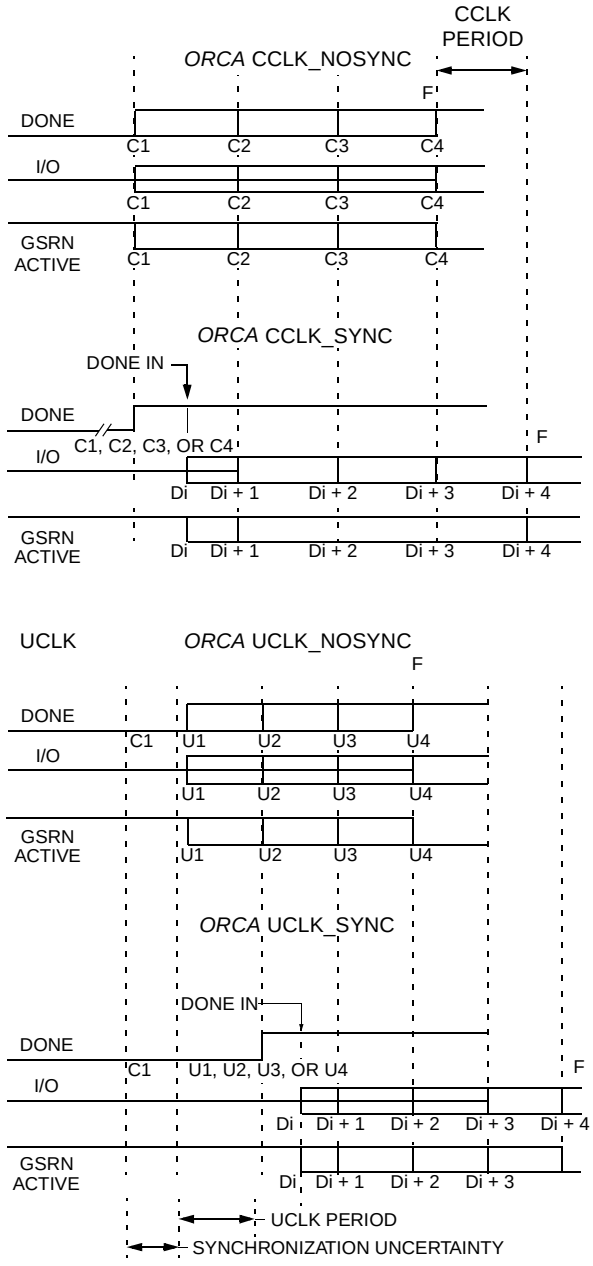
DONE is an open-drain bidirectional pin that may include an optional (enabled by default) pull-up resistor to accommodate wired ANDing. The open-drain DONE signals from multiple FPGAs can be tied together (ANDed) with a pull-up (internal or external) and used as an active-high ready signal, an active-low PROM enable, or a reset to other portions of the system. When used in SYNC mode, these ANDed DONE pins can be used to synchronize the other two start-up events, since they can all be synchronized to the same external signal. This signal will not rise until all FPGAs release their DONE pins, allowing the signal to be pulled high.

The default for *ORCA* is the CCLK_SYNC synchronized start-up mode where DONE is released on the first CCLK rising edge, C1 (see Figure 35). Since this is a synchronized start-up mode, the open-drain DONE signal can be held low externally to stop the occurrence of the other two start-up events. Once the DONE pin has been released and pulled up to a high level, the other two start-up events can be programmed individually to either happen immediately or after up to four rising edges of CCLK (Di, Di + 1, Di + 2, Di + 3, Di + 4). The default is for both events to happen immediately after DONE is released and pulled high.

A commonly used design technique is to release DONE one or more clock cycles before allowing the I/O to become active. This allows other configuration devices, such as PROMs, to be disconnected using the DONE signal so that there is no bus contention when the I/Os become active. In addition to controlling the FPGA during start-up, other start-up techniques that avoid contention include using isolation devices between the FPGA and other circuits in the system, reassigning I/O locations, and maintaining I/Os as 3-stated outputs until contentions are resolved.

Each of these start-up options can be selected during bit stream generation in *ORCA* Foundry, using Advanced Options. For more information, please see the *ORCA* Foundry documentation.

FPGA States of Operation (continued)



F = FINISHED, NO MORE CLKS REQUIRED.

5-2761(F)

Figure 35. Start-Up Waveforms

FPGA States of Operation (continued)

Reconfiguration

To reconfigure the FPGA when the device is operating in the system, a low pulse is input into PRGM or a program command is sent to the system bus. The configuration data in the FPGA is cleared, and the I/Os not used for configuration are 3-stated. The FPGA then samples the mode select inputs and begins reconfiguration. When reconfiguration is complete, DONE is released, allowing it to be pulled high.

Partial Reconfiguration

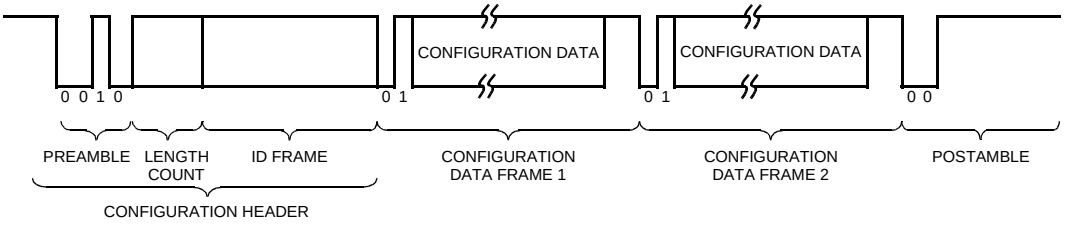
All *ORCA* device families have been designed to allow a partial reconfiguration of the FPGA at any time. This is done by setting a bit stream option in the previous configuration sequence that tells the FPGA to not reset all of the configuration RAM during a reconfiguration. Then only the configuration frames that are to be modified need to be rewritten, thereby reducing the configuration time.

Other bit stream options are also available that allow one portion of the FPGA to remain in operation while a partial reconfiguration is being done. If this is done, the user must be careful to not cause contention between the two configurations (the bit stream resident in the FPGA and the partial reconfiguration bit stream) as the second reconfiguration bit stream is being loaded.

Other Configuration Options

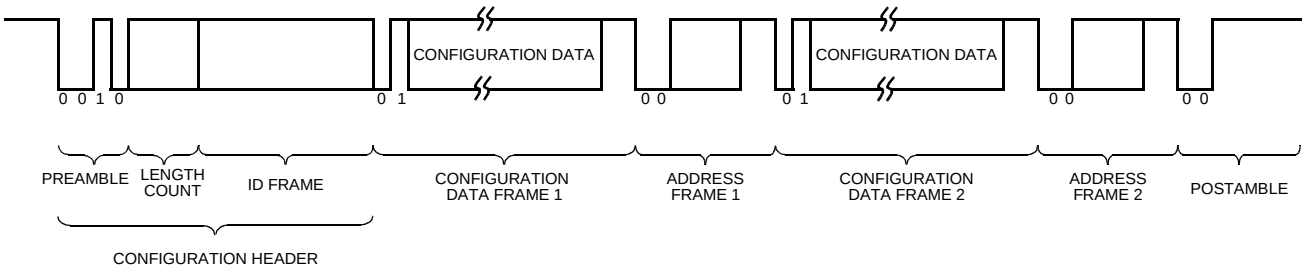
There are many other configuration options available to the user that can be set during bit stream generation in *ORCA* Foundry. These include options to enable boundary scan and/or the MPI and/or the programmable PLL blocks, readback options, and options to control and use the internal oscillator after configuration.

Other useful options that affect the next configuration (not the current configuration process) include options to disable the global set/reset during configuration, disable the 3-state of I/Os during configuration, and disable the reset of internal RAMs during configuration to allow for partial configurations (see above). For more information on how to set these and other configuration options, please see the *ORCA* Foundry documentation.



5-5759(F)

Figure 36. Serial Configuration Data Format—Autoincrement Mode



5-5760(F)

Figure 37. Serial Configuration Data Format—Explicit Mode

FPGA States of Operation (continued)

Table 36A. Configuration Frame Format and Contents

Frame	Contents	Description
Header	11110010	Preamble for generic FPGA.
	24-bit length count	Configuration bit stream length.
	11111111	8-bit trailing header.
ID Frame	0101 1111 1111 1111	ID frame header.
	44 reserved bits	Reserved bits set to 0.
	Part ID	20-bit part ID.
	Checksum	8-bit checksum.
	11111111	8 stop bits (high) to separate frames.
FPGA Header	1111 0010	This is a new mandatory header for generic portion.
	11111111	8 stop bits (high) to separate frames.
FPGA Address Frame	00	Address frame header.
	14-bit address	14-bit address of generic FPGA.
	Checksum	8-bit checksum.
	11111111	Eight stop bits (high) to separate frames.
FPGA Data Frame	01	Data frame header, same as generic.
	Alignment bits	String of 0 bits added to frame to reach a byte boundary.
	Data bits	Number of data bits depends upon device.
	Checksum	8-bit checksum.
	11111111	Eight stop bits (high) to separate frames.
Postamble for Generic FPGA	00 or 10	Postamble header, 00 = finish, 10 = more bits coming.
	11111111 111111	Dummy address.
	11111111 11111111	16 stop bits (high).

Table 36B. Configuration Frame Format and Contents for Embedded Block RAM

Frame	Contents	Description
RAM Header	11110001	A mandatory header for RAM bit stream portion.
	11111111	8 stop bits (high) to separate frames.
RAM Address Frame	00	Address frame header, same as generic.
	6-bit address	6-bit address of RAM blocks.
	Checksum	8-bit checksum.
	11111111	Eight stop bits (high) to separate frames.
RAM Data Frame	01	Data frame header, same as generic.
	000000	Six of 0 bits added to reach a byte boundary.
	512x18 data bits	Exact number of bits in a RAM block.
	Checksum	8-bit checksum.
	11111111	Eight stop bits (high) to separate frames.
Postamble for RAM	00 or 10	Postamble header. 00 = finish, 10 = more bits coming.
	111111	Dummy address.
	11111111 11111111	16 stop bits (high).

FPGA States of Operation (continued)

Table 37. Configuration Frame Size

Devices	OR4E2	OR4E4	OR4E6	OR4E10	OR4E14
Number of Frames	1796	2436	3076	3972	4356
Data Bits/Frame	900	1284	1540	1924	2372
Maximum Configuration Data (Number of bits/frame x Number of frames)	1,610,400	3,127,824	4,737,040	7,642,128	10,332,432
Maximum PROM Size (bits) (add configuration header and postamble)	1,161,648	3,128,072	4,737,288	7,642,376	10,332,680

Bit Stream Error Checking

There are three different types of bit stream error checking performed in the *ORCA* Series 4 FPGAs: ID frame, frame alignment, and CRC checking.

The ID data frame is sent to a dedicated location in the FPGA. This ID frame contains a unique code for the device for which it was generated. This device code is compared to the internal code of the FPGA. Any differences are flagged as an ID error. This frame is automatically created by the bit stream generation program in *ORCA* Foundry.

Each data and address frame in the FPGA begins with a frame start pair of bits and ends with eight stop bits set to 1. If any of the previous stop bits were a 0 when a frame start pair is encountered, it is flagged as a frame alignment error.

Error checking is also done on the FPGA for each frame by means of a checksum byte. If an error is found on evaluation of the checksum byte, then a checksum/parity error is flagged. The checksum is the XOR of all the data bytes, from the start of frame up to and including the bytes before the checksum. It applies to the ID, address, and data frames.

When any of the three possible errors occur, the FPGA is forced into an idle state, forcing $\overline{\text{INIT}}$ low. The FPGA will remain in this state until either the $\overline{\text{RESET}}$ or $\overline{\text{PRGM}}$ pins are asserted. Also the pin $\overline{\text{CFQ\_IRQ/MPI\_IRQ}}$ is forced low to signal the error and the specific type of bit stream error is written to one of the system bus registers by the FPGA configuration logic. The $\overline{\text{PGRM}}$ bit of the system bus control register can also be used to reset out of the error condition and restart configuration.

FPGA Configuration Modes

There are twelve methods for configuring the FPGA. Eleven of the configuration modes are selected on the M0, M1, and M2 inputs. The twelfth configuration mode is accessed through the boundary-scan interface. A fourth input, M3, is used to select the frequency of the internal oscillator, which is the source for CCLK in some configuration modes. The nominal frequencies of the internal oscillator are 1.25 MHz and 10 MHz. The 1.25 MHz frequency is selected when the M3 input is unconnected or driven to a high state.

There are three basic FPGA configuration modes: master, slave, and peripheral. The configuration data can be transmitted to the FPGA serially or in parallel bytes. As a master, the FPGA provides the control signals out to strobe data in. As a slave device, a clock is generated externally and provided into the CCLK input. In the three peripheral modes, the FPGA acts as a microprocessor peripheral. Table 38 lists the functions of the configuration mode pins.

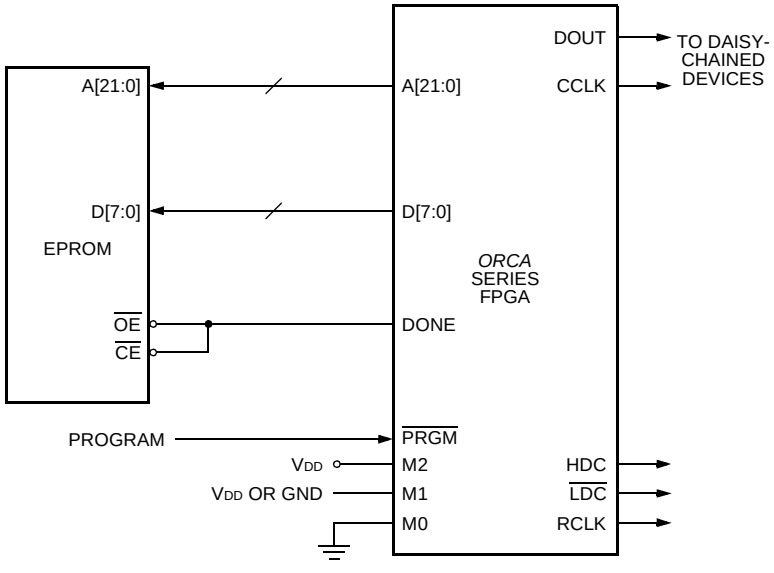
FPGA Configuration Modes (continued)

Table 38. Configuration Modes

M3	M2	M1	M0	CCLK	Configuration Mode	Data
0	0	0	0	Output. High-frequency.	Master Serial	Serial
0	1	0	0	Output. High-frequency.	Master Parallel	8-bit
0	1	0	1	Output. High-frequency.	Asynchronous Peripheral	8-bit
0	1	1	1	NA.	Reserved	NA
1	0	0	0	Output. Low-frequency.	Master Serial	Serial
1	0	0	1	Input.	Slave Parallel	8-bit
1	0	1	0	Output.	MPC860 MPI	8-bit
1	0	1	1	Output.	MPC860 MPI	16-bit
1	1	0	0	Output. Low-frequency.	Master Parallel	8-bit
1	1	0	1	Output. Low-frequency.	Asynchronous Peripheral	8-bit
1	1	1	0	Output.	MPC860 MPI	32-bit
1	1	1	1	Input.	Slave Serial	Serial

Master Parallel Mode

The master parallel configuration mode is generally used to interface to industry-standard, bytewide memory. Figure 38 provides the connections for master parallel mode. The FPGA outputs an 18-bit address on A[17:0] to memory and reads 1 byte of configuration data on the rising edge of RCLK. The parallel bytes are internally serialized starting with the least significant bit, D0. D[7:0] of the FPGA can be connected to D[7:0] of the microprocessor only if a standard prom file format is used. If a .bit or .rbt file is used from *ORCA* Foundry, then the user must mirror the bytes in the .bit or .rbt file **or** leave the .bit or .rbt file unchanged and connect D[7:0] of the FPGA to D[0:7] of the microprocessor.



5-9738(F)

Figure 38. Master Parallel Configuration Schematic

FPGA Configuration Modes (continued)

In master parallel mode, the starting memory address is 00000 hex, and the FPGA increments the address for each byte loaded.

One master mode FPGA can interface to the memory and provide configuration data on DOUT to additional FPGAs in a daisy-chain. The configuration data on DOUT is provided synchronously with the falling edge of CCLK. The frequency of the CCLK output is eight times that of RCLK.

Master Serial Mode

In the master serial mode, the FPGA loads the configuration data from an external serial ROM. The configuration data is either loaded automatically at start-up or on a $\overline{\text{PRGM}}$ command to reconfigure. Serial PROMs can be used to configure the FPGA in the master serial mode.

Configuration in the master serial mode can be done at powerup and/or upon a configure command. The system or the FPGA must activate the serial ROM's $\overline{\text{RESET}}$/OE and $\overline{\text{CE}}$ inputs. At powerup, the FPGA and serial ROM each contain internal power-on reset circuitry that allows the FPGA to be configured without the system providing an external signal. The power-on reset circuitry causes the serial ROM's internal address pointer to be reset. After powerup, the FPGA automatically enters its initialization phase.

The serial ROM/FPGA interface used depends on such factors as the availability of a system reset pulse, availability of an intelligent host to generate a configure command, whether a single serial ROM is used or multiple serial ROMs are cascaded, whether the serial ROM contains a single or multiple configuration programs, etc. Because of differing system requirements and capabilities, a single FPGA/serial ROM interface is generally not appropriate for all applications.

Data is read in the FPGA sequentially from the serial ROM. The DATA output from the serial ROM is connected directly into the DIN input of the FPGA. The CCLK output from the FPGA is connected to the CLK input of the serial ROM. During the configuration process, CCLK clocks one data bit on each rising edge.

Since the data and clock are direct connects, the FPGA/serial ROM design task is to use the system or FPGA to enable the $\overline{\text{RESET}}$/OE and $\overline{\text{CE}}$ of the serial ROM(s). There are several methods for enabling the serial ROM's $\overline{\text{RESET}}$/OE and $\overline{\text{CE}}$ inputs. The serial ROM's $\overline{\text{RESET}}$/OE is programmable to function with RESET active-high and $\overline{\text{OE}}$ active-low or $\overline{\text{RESET}}$ active-low and OE active-high.

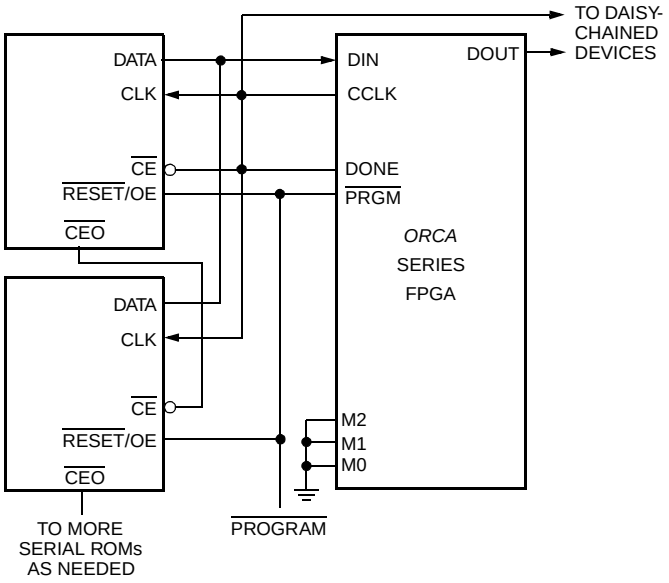
In Figure 39, serial ROMs are cascaded to configure multiple daisy-chained FPGAs. The host generates a 500 ns low pulse into the FPGA's $\overline{\text{PRGM}}$ input. The FPGA's $\overline{\text{INIT}}$ input is connected to the serial ROMs' $\overline{\text{RESET}}$/OE input, which has been programmed to function with $\overline{\text{RESET}}$ active-low and OE active-high. The FPGA DONE is routed to the $\overline{\text{CE}}$ pin. The low on DONE enables the serial ROMs. At the completion of configuration, the high on the FPGA's DONE disables the serial ROM.

Serial ROMs can also be cascaded to support the configuration of multiple FPGAs or to load a single FPGA when configuration data requirements exceed the capacity of a single serial ROM. After the last bit from the first serial ROM is read, the serial ROM outputs $\overline{\text{CEO}}$ low and 3-states the DATA output. The next serial ROM recognizes the low on $\overline{\text{CE}}$ input and outputs configuration data on the DATA output. After configuration is complete, the FPGA's DONE output into $\overline{\text{CE}}$ disables the serial ROMs.

This FPGA/serial ROM interface is not used in applications in which a serial ROM stores multiple configuration programs. In these applications, the next configuration program to be loaded is stored at the ROM location that follows the last address for the previous configuration program. The reason the interface in Figure 39 will not work in this application is that the low output on the $\overline{\text{INIT}}$ signal would reset the serial ROM address pointer, causing the first configuration to be reloaded.

In some applications, there can be contention on the FPGA's DIN pin. During configuration, DIN receives configuration data, and after configuration, it is a user I/O. If there is contention, an early DONE at start-up (selected in *ORCA* Foundry) may correct the problem. An alternative is to use $\overline{\text{LDC}}$ to drive the serial ROM's $\overline{\text{CE}}$ pin. In order to reduce noise, it is generally better to run the master serial configuration at 1.25 MHz (M3 pin tied high), rather than 10 MHz, if possible.

FPGA Configuration Modes (continued)



5-4456(F)

Figure 39. Master Serial Configuration Schematic

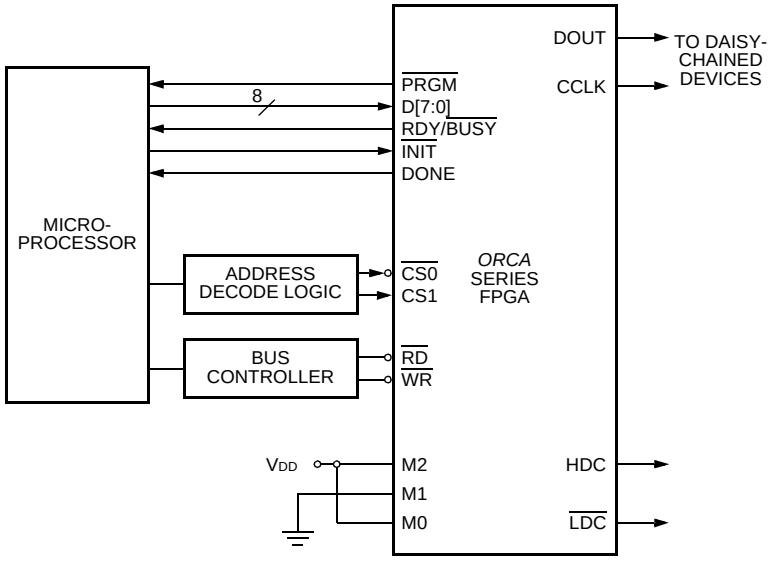
Asynchronous Peripheral Mode

Figure 40 shows the connections needed for the asynchronous peripheral mode. In this mode, the FPGA system interface is similar to that of a microprocessor-peripheral interface. The microprocessor generates the control signals to write an 8-bit byte into the FPGA. The FPGA control inputs include active-low $\overline{CS0}$ and active-high CS1 chip selects and $\overline{WR}$ and $\overline{RD}$ inputs. The chip selects can be cycled or maintained at a static level during the configuration cycle. Each byte of data is written into the FPGA's D[7:0] input pins. D[7:0] of the FPGA can be connected to D[7:0] of the microprocessor only if a standard prom file format is used. If a .bit or .rbt file is used from *ORCA* Foundry, then the user must mirror the bytes in the .bit or .rbt file **or** leave the .bit or .rbt file unchanged and connect D[7:0] of the FPGA to D[0:7] of the microprocessor.

The FPGA provides an RDY/$\overline{BUSY}$ status output to indicate that another byte can be loaded. A low on RDY/$\overline{BUSY}$ indicates that the double-buffered hold/shift registers are not ready to receive data, and this pin must be monitored to go high before another byte of data can be written. The shortest time RDY/$\overline{BUSY}$ is low occurs when a byte is loaded into the hold register and the shift register is empty, in which case the byte is immediately transferred to the shift register. The longest time for RDY/$\overline{BUSY}$ to remain low occurs when a byte is loaded into the holding register and the shift register has just started shifting configuration data into configuration RAM.

The RDY/$\overline{BUSY}$ status is also available on the D7 pin by enabling the chip selects, setting $\overline{WR}$ high, and applying $\overline{RD}$ low, where the $\overline{RD}$ input provides an output enable for the D7 pin when $\overline{RD}$ is low. The D[6:0] pins are not enabled to drive when $\overline{RD}$ is low and, therefore, only act as input pins in asynchronous peripheral mode. Optionally, the user can ignore the RDY/$\overline{BUSY}$ status and simply wait until the maximum time it would take for the RDY/$\overline{BUSY}$ line to go high, indicating the FPGA is ready for more data, before writing the next data byte.

FPGA Configuration Modes (continued)



5-9739(F)

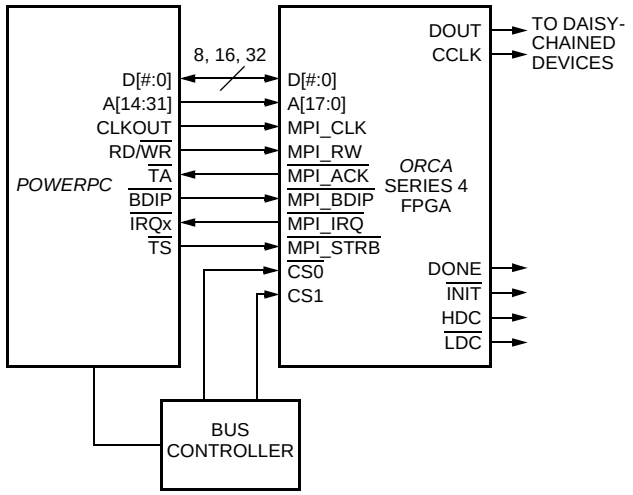
Figure 40. Asynchronous Peripheral Configuration

Microprocessor Interface Mode

The built-in MPI in Series 4 FPGAs is designed for use in configuring the FPGA. Figure 41 shows the glueless interface for FPGA configuration and readback from the *PowerPC* processor. When enabled by the mode pins, the MPI handles all configuration/readback control and handshaking with the host processor. For single FPGA configuration, the host sets the configuration control register $\overline{\text{PRGM}}$ bit to zero then back to a one and, after reading that the configuration write data acknowledge register is high, transfers data 8, 16, or 32 bits at a time to the FPGA's D[#:0] input pins. If configuring multiple FPGAs through daisy-chain operation is desired, the SYS_DAISY bit must be set in the configuration control register of the MPI.

There are two options for using the host interrupt request in configuration mode. The configuration control register offers control bits to enable the interrupt on either a bit stream error or to notify the host processor when the FPGA is ready for more configuration data. The MPI status register may be used in conjunction with, or in place of, the interrupt request options. The status register contains a 2-bit field to indicate the bit stream error status. As previously mentioned, there is also a bit to indicate the MPI's readiness to receive another byte of configuration data. A flow chart of the MPI configuration process is shown in Figure 42.

FPGA Configuration Modes (continued)

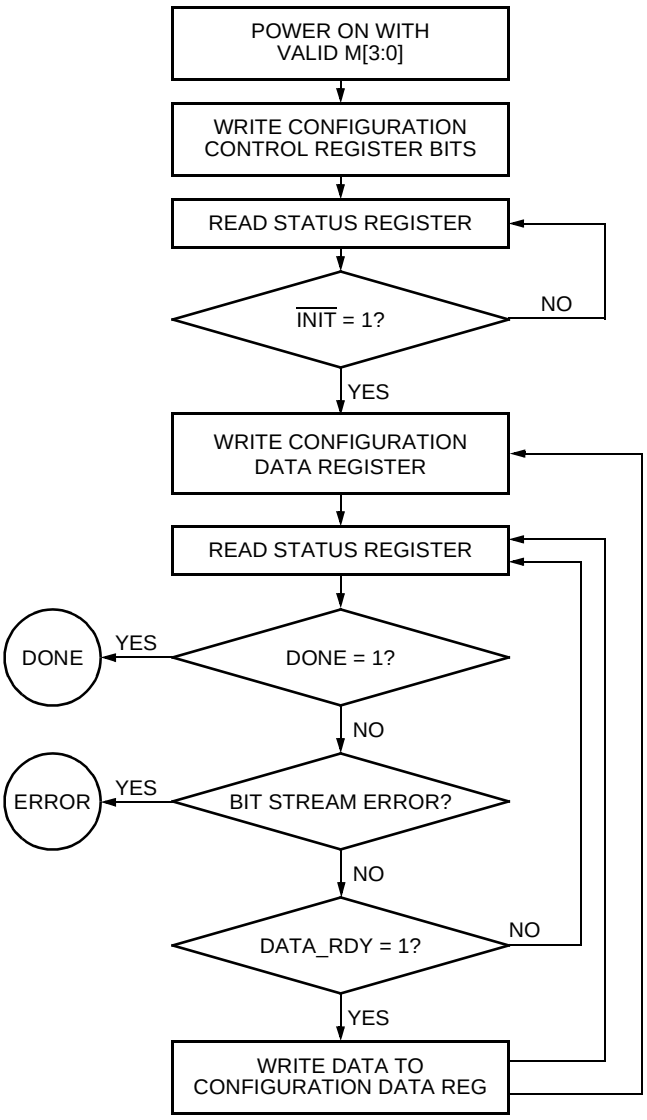


5-5761(F)

Figure 41. *PowerPC*/MPI Configuration Schematic

Configuration readback can also be performed via the MPI when it is in user mode. The MPI is enabled in user mode by setting the MPI_USER_ENABLE bit to 1 in the configuration control register prior to the start of configuration or through a configuration option. To perform readback, the host processor writes the 14-bit readback start address to the readback address registers and sets the RD_CFG bit to 0 in the configuration control register. Readback data is returned 8 bits at a time to the readback data register and is valid when the DATA_RDY bit of the status register is 1. There is no error checking during readback. A flow chart of the MPI readback operation is shown in Figure 43. The RD_DATA pin used for dedicated FPGA readback is invalid during MPI readback.

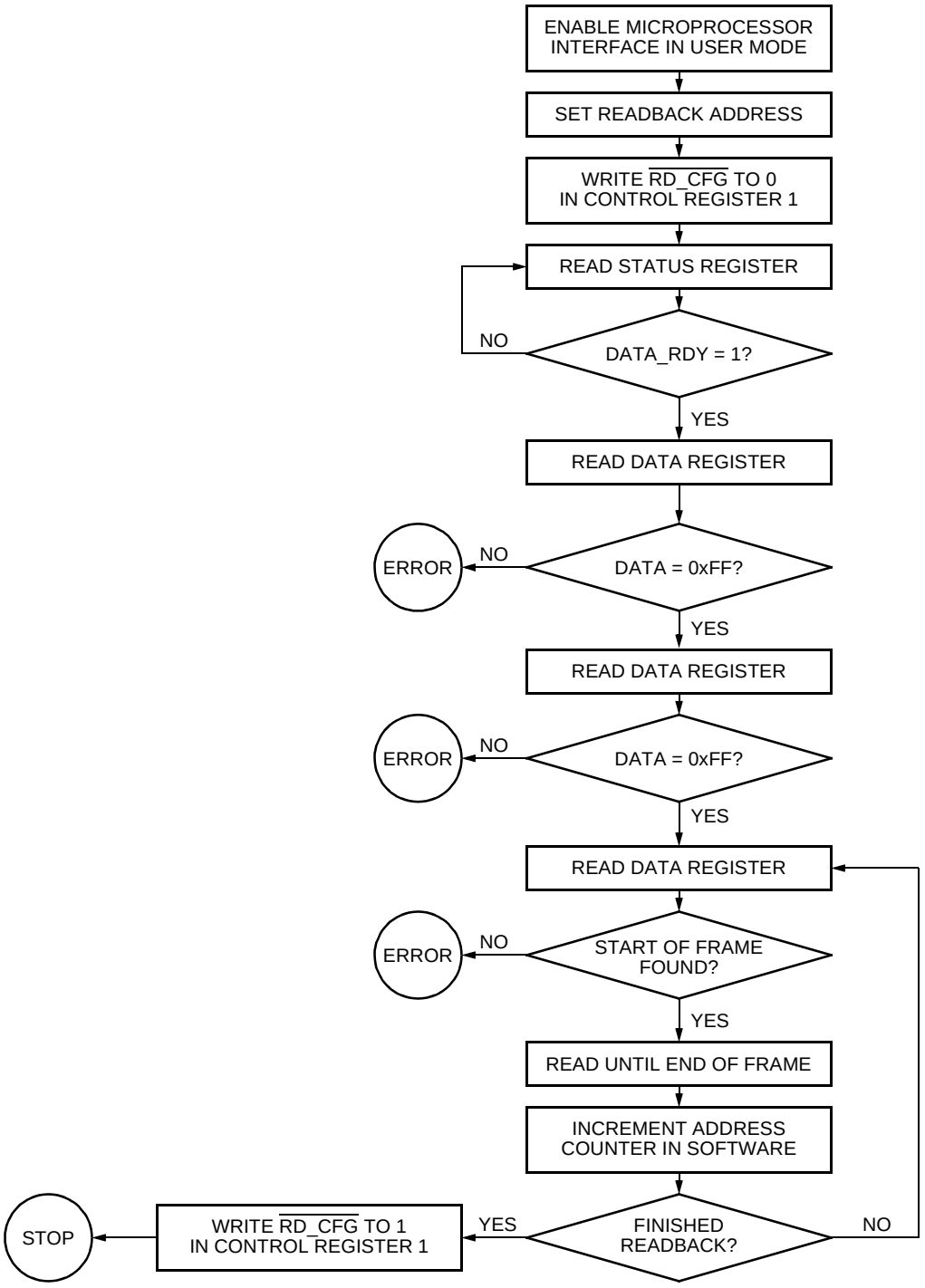
FPGA Configuration Modes (continued)



5-5763(F)

Figure 42. Configuration Through MPI

FPGA Configuration Modes (continued)



5-5764(F)

Figure 43. Readback Through MPI

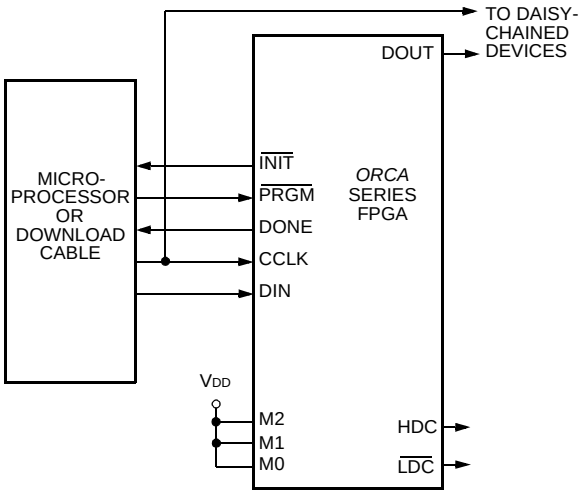
FPGA Configuration Modes (continued)

Slave Serial Mode

The slave serial mode is primarily used when multiple FPGAs are configured in a daisy chain (see the Daisy Chaining section). It is also used on the FPGA evaluation board that interfaces to the download cable. A device in the slave serial mode can be used as the lead device in a daisy chain. Figure 44 shows the connections for the slave serial configuration mode.

The configuration data is provided into the FPGA's DIN input synchronous with the configuration clock CCLK input. After the FPGA has loaded its configuration data, it retransmits the incoming configuration data on DOUT. CCLK is routed into all slave serial mode devices in parallel.

Multiple slave FPGAs can be loaded with identical configurations simultaneously. This is done by loading the configuration data into the DIN inputs in parallel.



5-4485(F)

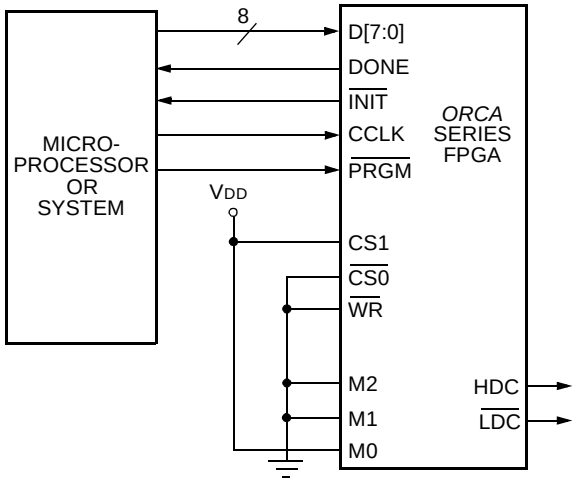
Figure 44. Slave Serial Configuration Schematic

Slave Parallel Mode

The slave parallel mode is essentially the same as the slave serial mode except that 8 bits of data are input on pins D[7:0] for each CCLK cycle. Due to 8 bits of data being input per CCLK cycle, the DOUT pin does not contain a valid bit stream for slave parallel mode. As a result, the lead device cannot be used in the slave parallel mode in a daisy-chain configuration.

Figure 45 is a schematic of the connections for the slave parallel configuration mode. $\overline{WR}$ and $\overline{CS0}$ are active-low chip select signals, and CS1 is an active-high chip select signal. These chip selects allow the user to configure multiple FPGAs in slave parallel mode using an 8-bit data bus common to all of the FPGAs. These chip selects can then be used to select the FPGAs to be configured with a given bit stream. The chip selects must be active for each valid CCLK cycle until the device has been completely programmed. They can be inactive between cycles but must meet the setup and hold times for each valid positive CCLK. D[7:0] of the FPGA can be connected to D[7:0] of the microprocessor only if a standard prom file format is used. If a .bit or .rbt file is used from *ORCA* Foundry, then the user must mirror the bytes in the .bit or .rbt file **or** leave the .bit or .rbt file unchanged and connect D[7:0] of the FPGA to D[0:7] of the microprocessor.

FPGA Configuration Modes (continued)



5-4487(F)

Figure 45. Slave Parallel Configuration Schematic

Daisy Chaining

Multiple FPGAs can be configured by using a daisy chain of the FPGAs. Daisy chaining uses a lead FPGA and one or more FPGAs configured in slave serial mode. The lead FPGA can be configured in any mode except slave parallel mode.

All daisy-chained FPGAs are connected in series. Each FPGA reads and shifts the preamble and length count in on positive CCLK and out on negative CCLK edges.

An upstream FPGA that has received the preamble and length count outputs a high on DOUT until it has received the appropriate number of data frames so that downstream FPGAs do not receive frame start bit pairs. After loading and retransmitting the preamble and length count to a daisy chain of slave devices, the lead device loads its configuration data frames. The loading of configuration data continues after the lead device has received its configuration data if its internal frame bit counter has not reached the length count. When the configuration RAM is full and the number of bits received is less than the length count field, the FPGA shifts any additional data out on DOUT.

The configuration data is read into DIN of slave devices on the positive edge of CCLK, and shifted out DOUT on the negative edge of CCLK. Figure 46 shows the connections for loading multiple FPGAs in a daisy-chain configuration.

The generation of CCLK for the daisy-chained devices that are in slave serial mode differs depending on the configuration mode of the lead device. A master parallel mode device uses its internal timing generator to produce an internal CCLK at eight times its memory address rate (RCLK). The asynchronous peripheral mode device outputs eight CCLKs for each write cycle. If the lead device is configured in slave mode, CCLK must be routed to the lead device and to all of the daisy-chained devices.

FPGA Configuration Modes (continued)

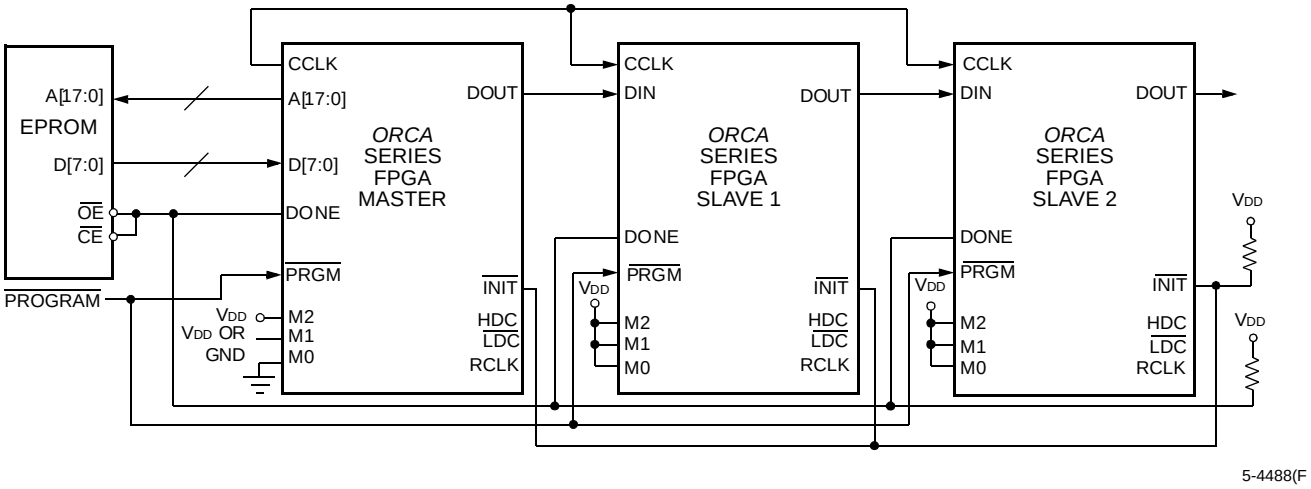


Figure 46. Daisy-Chain Configuration Schematic

As seen in Figure 46, the $\overline{\text{INIT}}$ pins for all of the FPGAs are connected together. This is required to guarantee that powerup and initialization will work correctly. In general, the DONE pins for all of the FPGAs are also connected together as shown to guarantee that all of the FPGAs enter the start-up state simultaneously. This may not be required, depending upon the start-up sequence desired.

Daisy-Chaining with Boundary Scan

Multiple FPGAs can be configured through the JTAG ports by using a daisy chain of the FPGAs. This daisy-chaining operation is available upon initial configuration after powerup, after a power-on reset, after pulling the program pin to reset the chip, or during a reconfiguration if the EN_JTAG RAM has been set.

All daisy-chained FPGAs are connected in series. Each FPGA reads and shifts the preamble and length count in on the positive TCK and out on the negative TCK edges.

An upstream FPGA that has received the preamble and length count outputs a high on TDO until it has received the appropriate number of data frames so that downstream FPGAs do not receive frame start bit pairs. After loading and retransmitting the preamble and length count to a daisy chain of downstream devices, the lead device loads its configuration data frames.

The loading of configuration data continues after the lead device had received its configuration read into TDI of downstream devices on the positive edge of TCK, and shifted out TDO on the negative edge of TCK.

Absolute Maximum Ratings

Stresses in excess of the absolute maximum ratings can cause permanent damage to the device. These are absolute stress ratings only. Functional operation of the device is not implied at these or any other conditions in excess of those given in the operations sections of this data sheet. Exposure to absolute maximum ratings for extended periods can adversely affect device reliability.

The *ORCA* Series FPGAs include circuitry designed to protect the chips from damaging substrate injection currents and to prevent accumulations of static charge. Nevertheless, conventional precautions should be observed during storage, handling, and use to avoid exposure to excessive electrical stress.

Absolute Maximum Ratings (continued)

Table 39. Absolute Maximum Ratings

Parameter	Symbol	Min	Max	Unit
Storage Temperature	T_{stg}	–65	150	°C
Power Supply Voltage with Respect to Ground	$V_{DD}3$	—	≤4.2	V
	$V_{DD}15$	—	2	V
Input Signal with Respect to Ground	—	V_{SS} – 0.3	$V_{DD}IO$ + 0.3	V
Signal Applied to High-impedance Output	—	V_{SS} – 0.3	$V_{DD}IO$ + 0.3	V
Maximum Package Body Temperature	—	—	220	°C

Recommended Operating Conditions

Table 40. Recommended Operating Conditions

Parameter	Symbol	Min	Max	Unit
Power Supply Voltage with Respect to Ground	$V_{DD}3$	2.7	3.6	V
	$V_{DD}15$	1.4	1.6	V
Input Voltages	V_{IN}	V_{SS} – 0.3	$V_{DD}IO$ + 0.3	V
Junction Temperature	T_J	–40	125	°C

Note: The maximum recommended junction temperature (T_J) during operation is 125 °C.

Electrical Characteristics

Table 41. Electrical Characteristics

Parameter	Symbol	Test Conditions	Min	Max	Unit
Input Leakage Current	I_L	V_{DD} = max, V_{IN} = V_{SS} or V_{DD}	–10	10	µA
Standby Current:	I_{DDSB}	T_A = 25 °C, V_{DD} = 3.3 V			
OR4E2		internal oscillator running, no output loads,	—	TBD	mA
OR4E4		inputs V_{DD} or GND (after configuration)	—	TBD	mA
OR4E6			—	TBD	mA
OR4E10			—	TBD	mA
OR4E14			—	TBD	mA
Standby Current:	I_{DDSB}	T_A = 25 °C, V_{DD} = 3.3 V			
OR4E2		internal oscillator stopped, no output loads,	—	TBD	mA
OR4E4		inputs V_{DD} or GND (after configuration)	—	TBD	mA
OR4E6			—	TBD	mA
OR4E10			—	TBD	mA
OR4E14			—	TBD	mA
Powerup Current:	Ipp	Power supply current at approximately 1 V, within			
OR4E2		a recommended power supply ramp rate of	TBD	—	mA
OR4E4		1 ms—200 ms	TBD	—	mA
OR4E6			TBD	—	mA
OR4E10			TBD	—	mA
OR4E14			TBD	—	mA

* The pull-up resistor will externally pull the pin to a level 1.0 V below $V_{DD}IO$.

Electrical Characteristics (continued)

Table 41. Electrical Characteristics (continued)

Parameter	Symbol	Test Conditions	Min	Max	Unit
Data Retention Voltage ($V_{DD}33$)	V_{DR}	T_A = 25 °C	2.3	—	V
Input Capacitance	C_{IN}	T_A = 25 °C, V_{DD} = 3.3 V Test frequency = 1 MHz	—	6	pF
Output Capacitance	C_{OUT}	T_A = 25 °C, V_{DD} = 3.3 V Test frequency = 1 MHz	—	6	pF
DONE Pull-up Resistor*	R_{DONE}	—	100	—	kΩ
M[3:0] Pull-up Resistor*	R_M	—	100	—	kΩ
I/O Pad Static Pull-up Current*	I_{PU}	$V_{DD}IO$ = 3.6 V, V_{IN} = V_{SS}, T_A = 0 °C	14.4	50.9	µA
I/O Pad Static Pull-down Current	I_{PD}	$V_{DD}IO$ = 3.6 V, V_{IN} = V_{SS}, T_A = 0 °C	26	103	µA
I/O Pad Pull-up Resistor*	R_{PU}	$V_{DD}IO$ = all, V_{IN} = V_{SS}, T_A = 0 °C	100	—	kΩ
I/O Pad Pull-down Resistor	R_{PD}	$V_{DD}IO$ = all, V_{IN} = V_{DD}, TA = 0 °C	50	—	kΩ
DONE Pull-up Resistor*	R_{DONE}	—	100	—	kΩ
M[3:0] Pull-up Resistor*	R_M	—	100	—	kΩ
I/O Pad Static Pull-up Current*	I_{PU}	$V_{DD}IO$ = 3.6 V, V_{IN} = V_{SS}, T_A = 0 °C	14.4	50.9	µA
I/O Pad Static Pull-down Current	I_{PD}	$V_{DD}IO$ = 3.6 V, V_{IN} = V_{SS}, T_A = 0 °C	26	103	µA
I/O Pad Pull-up Resistor*	R_{PU}	$V_{DD}IO$ = all, V_{IN} = V_{SS}, T_A = 0 °C	100	—	kΩ
I/O Pad Pull-down Resistor	R_{PD}	$V_{DD}IO$ = all, V_{IN} = V_{DD}, T_A = 0 °C	50	—	kΩ

* The pull-up resistor will externally pull the pin to a level 1.0 V below $V_{DD}IO$.

Pin Information

Pin Descriptions

This section describes the pins found on the Series 4 FPGAs. Any pin not described in this table is a user-programmable I/O. During configuration, the user-programmable I/Os are 3-stated with an internal pull-up resistor enabled. If any pin is not used (or not bonded to a package pin), it is also 3-stated with an internal pull-up resistor enabled after configuration.

Table 42. Pin Descriptions

Symbol	I/O	Description
Dedicated Pins		
$V_{DD}33$	—	3 V positive power supply.
$V_{DD}15$	—	1.5 V positive power supply for internal logic.
V_{DDIO}	—	Positive power supply used by I/O banks.
GND	—	Ground supply.
PLL_VF	—	Dedicated pins for PLL filtering.
PTEMP	I	Temperature-sensing diode pin. Dedicated input.
$\overline{\text{RESET}}$	I	During configuration, $\overline{\text{RESET}}$ forces the restart of configuration and a pull-up is enabled. After configuration, $\overline{\text{RESET}}$ can be used as a general FPGA input or as a direct input, which causes all PLC latches/FFs to be asynchronously set/reset.
CCLK	I O	In the master and asynchronous peripheral modes, CCLK is an output which strobes configuration data in. In the slave or readback after configuration, CCLK is input synchronous with the data on DIN or D[7:0]. CCLK is an output for daisy-chain operation when the lead device is in master, peripheral, or system bus modes.
DONE	I	As an input, a low level on DONE delays FPGA start-up after configuration.*
	O	As an active-high, open-drain output, a high level on this signal indicates that configuration is complete. DONE has an optional pull-up resistor.
$\overline{\text{PRGM}}$	I	$\overline{\text{PRGM}}$ is an active-low input that forces the restart of configuration and resets the boundary-scan circuitry. This pin always has an active pull-up.
$\overline{\text{RD\_CFG}}$	I	This pin must be held high during device initialization until the $\overline{\text{INIT}}$ pin goes high. This pin always has an active pull-up. During configuration, $\overline{\text{RD\_CFG}}$ is an active-low input that activates the TS_ALL function and 3-states all of the I/O. After configuration, $\overline{\text{RD\_CFG}}$ can be selected (via a bit stream option) to activate the TS_ALL function as described above, or, if readback is enabled via a bit stream option, a high-to-low transition on $\overline{\text{RD\_CFG}}$ will initiate readback of the configuration data, including PFU output states, starting with frame address 0.
RD_DATA/TDO	O	RD_DATA/TDO is a dual-function pin. If used for readback, RD_DATA provides configuration data out. If used in boundary-scan, TDO is test data out.
$\overline{\text{CFG\_IRQ}}$/$\overline{\text{MPI\_IRQ}}$	O	During JTAG, slave, master, and asynchronous peripheral configuration assertion by the FPGA on this $\overline{\text{CFG\_IRQ}}$ (active-low) indicates an error or errors for block RAM or FPSC initialization. MPI active-low interrupt request output.

* The FPGA States of Operation section contains more information on how to control these signals during start-up. The timing of DONE release is controlled by one set of bit stream options, and the timing of the simultaneous release of all other configuration pins (and the activation of all user I/Os) is controlled by a second set of options.

Pin Information (continued)

Table 42. Pin Descriptions (continued)

Symbol	I/O	Description
Special-Purpose Pins (Can also be used as a general I/O)		
M[3:0]	I	During powerup and initialization, M0—M3 are used to select the configuration mode with their values latched on the rising edge of INIT. During configuration, a pull-up is enabled.
	I/O	After configuration, these pins are user-programmable I/O.*
PLL_CK[0:7]	I/O	Dedicated PCM clock pins. These pins are a user-programmable I/O pins if not used by PLLs.
P[TBTR]CLK[1:0][TC]	I/O	Pins dedicated for the primary clock. Input pins on the middle of each side with differential pairing. They may be used as general I/O pins if not needed for clocking purposes.
TDI, TCK, TMS	I	If boundary-scan is used, these pins are test data in, test clock, and test mode select inputs. If boundary-scan is not selected, all boundary-scan functions are inhibited once configuration is complete. Even if boundary-scan is not used, either TCK or TMS must be held at logic 1 during configuration. Each pin has a pull-up enabled during configuration.
	I/O	After configuration, these pins are user-programmable I/O.*
RDY/$\overline{\text{BUSY}}$/RCLK	O	During configuration in peripheral mode, RDY/RCLK indicates another byte can be written to the FPGA. If a read operation is done when the device is selected, the same status is also available on D7 in asynchronous peripheral mode. After configuration, if the MPI is not used, this pin is a user-programmable I/O pin.*
	I/O	During the master parallel configuration mode, RCLK is a read output signal to an external memory. This output is not normally used.
HDC	O	High during configuration is output high until configuration is complete. It is used as a control output, indicating that configuration is not complete.
	I/O	After configuration, this pin is a user-programmable I/O pin.*
$\overline{\text{LDC}}$	O	$\overline{\text{Low during configuration}}$ is output low until configuration is complete. It is used as a control output, indicating that configuration is not complete.
	I/O	After configuration, this pin is a user-programmable I/O pin.*
$\overline{\text{INIT}}$	I/O	$\overline{\text{INIT}}$ is a bidirectional signal before and during configuration. During configuration, a pull-up is enabled, but an external pull-up resistor is recommended. As an active-low, open-drain output, $\overline{\text{INIT}}$ is held low during power stabilization and internal clearing of memory. As an active-low input, $\overline{\text{INIT}}$ holds the FPGA in the wait-state before the start of configuration. After configuration, this pin is a user-programmable I/O pin.*
$\overline{\text{CS0}}$, CS1	I	$\overline{\text{CS0}}$ and CS1 are used in the asynchronous peripheral, slave parallel, and microprocessor configuration modes. The FPGA is selected when $\overline{\text{CS0}}$ is low and CS1 is high. During configuration, a pull-up is enabled.
	I/O	After configuration, these pins are user-programmable I/O pins.*
$\overline{\text{RD}}$/$\overline{\text{MPI\_STRB}}$	I	$\overline{\text{RD}}$ is used in the asynchronous peripheral configuration mode. A low on $\overline{\text{RD}}$ changes D7 into a status output. As a status indication, a high indicates ready, and a low indicates busy. $\overline{\text{WR}}$ and $\overline{\text{RD}}$ should not be used simultaneously. If they are, the write strobe overrides. This pin is also used as the MPI data transfer strobe.
	I/O	After configuration, if the MPI is not used, this pin is a user-programmable I/O pin.*

* The FPGA States of Operation section contains more information on how to control these signals during start-up. The timing of DONE release is controlled by one set of bit stream options, and the timing of the simultaneous release of all other configuration pins (and the activation of all user I/Os) is controlled by a second set of options.

Pin Information (continued)

Table 42. Pin Descriptions (continued)

Symbol	I/O	Description
Special-Purpose Pins (continued)		
A[17:0]	I	During MPI mode, the A[17:0] are used as the address bus driven by the *PowerPC* bus master utilizing the least significant bits of the *PowerPC* 32-bit address.
	O	During master parallel configuration mode, A[17:0] address the configuration EPROM. In MPI mode, many of the A[n] pins have alternate uses as described below. See the Special Function Blocks section for more MPI information. During configuration, if not in master parallel or an MPI configuration mode, these pins are 3-stated with a pull-up enabled.
$\overline{\text{MPI\_BURST}}$	I	A[21] is used as the $\overline{\text{MPI\_BURST}}$. It is driven low to indicate a burst transfer is in progress. Driven high indicates that the current transfer is not a burst.
$\overline{\text{MPI\_BDIP}}$	I	A[22] is used as the $\overline{\text{MPI\_BDIP}}$. It is driven by the *PowerPC* processor. Assertion of this pin indicates that the second beat in front of the current one is requested by the master. Negated before the burst transfer ends to abort the burst data phase.
MPI_TSZ[1:0]	I	A[19:18] are used as the MPI_TSZ[1:0] signals and are driven by the bus master to indicate the data transfer size for the transaction. Set 01 for byte, 10 for half-word, and 00 for word. During master parallel mode A[21:0], address the configuration EPROMs up to 4M bytes.
A[21:0]	O	If not used for MPI, these pins are user-programmable I/O pins.*
$\overline{\text{MPI\_ACK}}$	O	In *PowerPC* mode MPI operation, this is driven low indicating the MPI received the data on the write cycle or returned data on a read cycle.
$\overline{\text{MPI\_CLK}}$	I	This is the *PowerPC* synchronous, positive-edge bus clock used for the MPI interface. It can be a source of the clock for the embedded system bus. If MPI is used, this can be the *AMBA* bus clock.
$\overline{\text{MPI\_TEA}}$	O	A low on the MPI transfer error acknowledge indicates that the MPI detects a bus error on the internal system bus for the current transaction.
$\overline{\text{MPI\_RTRY}}$	O	This pin requests the MPC860 to relinquish the bus and retry the cycle.
D[31:0]	I/O	Selectable data bus width from 8, 16, 32 bits. Driven by the bus master in a write transaction. Driven by MPI in a read transaction.
	I	D[7:0] receive configuration data during master parallel, peripheral, and slave parallel configuration modes and each pin has a pull-up enabled. During serial configuration modes, D0 is the DIN input. D[7:3] output internal status for asynchronous peripheral mode when RD is low. After configuration, the pins are user-programmable I/O pins.*
DP[3:0]	I/O	Selectable parity bus width from 1, 2, 4-bit, DP[0] for D[7:0], DP[1] for D[15:8], DP[2] for D[23:16], and DP[3] for D[32:24]. After configuration, this pin is a user-programmable I/O pin.*
DIN	I	During slave serial or master serial configuration modes, DIN accepts serial configuration data synchronous with CCLK. During parallel configuration modes, DIN is the D0 input. During configuration, a pull-up is enabled.
	I/O	After configuration, this pin is a user-programmable I/O pin.*
DOUT	O	During configuration, DOUT is the serial data output that can drive the DIN of daisy-chained slave devices. Data out on DOUT changes on the rising edge of CCLK.
	I/O	After configuration, DOUT is a user-programmable I/O pin.*

* The FPGA States of Operation section contains more information on how to control these signals during start-up. The timing of DONE release is controlled by one set of bit stream options, and the timing of the simultaneous release of all other configuration pins (and the activation of all user I/Os) is controlled by a second set of options.

Pin Information (continued)

Package Compatibility

Table 43 provides the number of user I/Os available for the *ORCA* Series 4 FPGAs for each available package. Each package has seven dedicated configuration pins.

Table 44 through Table 46 provide the package pin and pin function for the *ORCA* Series 4 FPGAs and packages. The bond pad name is identified in the PIO nomenclature used in the *ORCA* Foundry design editor.

When the number of FPGA bond pads exceeds the number of package pins, bond pads are unused. When the number of package pins exceeds the number of bond pads, package pins are left unconnected (no connects). When a package pin is to be left as a no connect for a specific die, it is indicated as a note in the device pad column for the FPGA. The tables provide no information on unused pads.

Table 43. *ORCA* I/Os Summary

Device	352 PBGA	432 EBGA	680 PBGAM1
OR4E2/OR4E4/OR4E6			
User I/O Single Ended	262	306	466
Available Differential Pairs (LVDS, LVPECL)	128	150	196
Configuration	7	7	7
Dedicated Function	3	3	3
$V_{DD}15$	16	40	48
$V_{DD}33$	8	8	8
$V_{DD}IO$	24	24	60
V_{SS}	68	44	88

Pin Information (continued)

As shown in the Pair column, differential pairs and physical locations are numbered within each bank (e.g., L19C_A0 is the nineteenth pair in an associated bank). The C indicates complementary differential whereas a T indicates true differential. The _A0 indicates the physical location of adjacent balls in either the horizontal or vertical direction. Other physical indicators are as follows:

- _A1 indicates one ball between pairs.
- _A2 indicates two balls between pairs.
- _D0 indicates balls are diagonally adjacent.
- _D1 indicates diagonally adjacent separated by one physical ball.

VREF pins, shown in the Additional Function column, are associated to the bank and group (e.g., VREF_TL_01 is the VREF for group one of the top left (TL) bank).

Table 44. 352-Pin PBGA Pinout

352 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
D12	TL	1	PT11D	PT13D	PT18D	$\overline{\text{MPI\_RTRY}}$	L1C_D2	COMPLEMENT
B10	TL	1	PT11C	PT13C	PT18C	$\overline{\text{MPI\_ACK}}$	L1T_D2	TRUE
A10	TL	1	PT10D	PT12D	PT16D	M0	L2C_A2	COMPLEMENT
D10	TL	1	PT10C	PT12C	PT16C	M1	L2T_A2	TRUE
B9	TL	2	PT10B	PT12B	PT15D	MPI_CLK	L3C_D0	COMPLEMENT
C10	TL	2	PT10A	PT12A	PT15C	A21/$\overline{\text{MPI\_BURST}}$	L3T_D0	TRUE
A9	TL	2	PT9D	PT11D	PT14D	M2	L4C_D0	COMPLEMENT
B8	TL	2	PT9C	PT11C	PT14C	M3	L4T_D0	TRUE
A8	TL	2	PT9B	PT11B	PT13D	VREF_TL_02	L5C_D1	COMPLEMENT
C9	TL	2	PT9A	PT11A	PT13C	$\overline{\text{MPI\_TEA}}$	L5T_D1	TRUE
B7	TL	3	PT8B	PT9D	PT11D	VREF_TL_03	—	COMPLEMENT
D8	TL	3	PT7D	PT8D	PT10D	D0	L6C_D2	COMPLEMENT
A7	TL	3	PT7C	PT8C	PT10C	TMS	L6T_D2	TRUE
C8	TL	4	PT7B	PT7D	PT9D	A20/$\overline{\text{MPI\_BDIP}}$	L7C_D2	COMPLEMENT
B6	TL	4	PT7A	PT7C	PT9C	A19/MPI_TSZ1	L7T_D2	TRUE
D7	TL	4	PT6D	PT6D	PT8D	A18/MPI_TSZ0	L8C_D2	COMPLEMENT
A6	TL	4	PT6C	PT6C	PT8C	D3	L8T_D2	TRUE
B5	TL	5	PT5D	PT5D	PT6D	D1	L9C_A0	COMPLEMENT
A5	TL	5	PT5C	PT5C	PT6C	D2	L9T_A0	TRUE
C6	TL	5	PT4D	PT4D	PT4D	TDI	L10C_D2	COMPLEMENT
B4	TL	5	PT4C	PT4C	PT4C	TCK	L10T_D2	TRUE
D5	TL	6	PT2D	PT2D	PT2D	PLL_CK1C/PPLL	L11C_D2	COMPLEMENT
A4	TL	6	PT2C	PT2C	PT2C	PLL_CK1T/PPLL	L11T_D2	TRUE
E2	TL	7	PL2D	PL2D	PL2D	PLL_CK0C/ HPPLL	L12C_A1	COMPLEMENT
E4	TL	7	PL2C	PL2C	PL2C	PLL_CK0T/ HPPLL	L12T_A1	TRUE
E3	TL	7	PL3D	PL4D	PL4D	D5	L13C_A1	COMPLEMENT
E1	TL	7	PL3C	PL4C	PL4C	D6	L13T_A1	TRUE
F2	TL	8	PL4D	PL5D	PL6D	HDC	L14C_D1	COMPLEMENT

Pin Information (continued)

Table 44. OR4E6 352-Pin PBGA Pinout (continued)

352 BGA Ball	$V_{DD}IO$ Bank	V_{REF} Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
G4	TL	8	PL4C	PL5C	PL6C	$\overline{LDC}$	L14T_D1	TRUE
F3	TL	9	PL5D	PL6D	PL8D	—	L15C_A1	COMPLEMENT
F1	TL	9	PL5C	PL6C	PL8C	D7	L15T_A1	TRUE
G1	TL	9	PL5B	PL7D	PL9D	V_{REF}_TL_09	L16C_A1	COMPLEMENT
G3	TL	9	PL5A	PL7C	PL9C	A17	L16T_A1	TRUE
H2	TL	9	PL6D	PL8D	PL10D	$\overline{CS0}$	L17C_D1	COMPLEMENT
J4	TL	9	PL6C	PL8C	PL10C	CS1	L17T_D1	TRUE
H1	TL	10	PL7D	PL10D	PL12D	$\overline{INIT}$	L18C_A1	COMPLEMENT
H3	TL	10	PL7C	PL10C	PL12C	DOUT	L18T_A1	TRUE
J2	TL	10	PL7B	PL11D	PL13D	V_{REF}_TL_10	L19C_A0	COMPLEMENT
J1	TL	10	PL7A	PL11C	PL13C	A16	L19T_A0	TRUE
B1	TL	—	V_{DD}33	V_{DD}33	V_{DD}33	—	—	—
C2	TL	—	PRD_DATA	PRD_DATA	PRD_DATA	TDO	—	—
C1	TL	—	$\overline{PRESET}$	$\overline{PRESET}$	$\overline{PRESET}$	—	—	—
D2	TL	—	$\overline{PRD\_CFG}$	$\overline{PRD\_CFG}$	$\overline{PRD\_CFG}$	—	—	—
D3	TL	—	$\overline{PPRGRM}$	$\overline{PPRGRM}$	$\overline{PPRGRM}$	—	—	—
D1	TL	—	V_{DD}IO_TL	V_{DD}IO_TL	V_{DD}IO_TL	—	—	—
G2	TL	—	V_{DD}IO_TL	V_{DD}IO_TL	V_{DD}IO_TL	—	—	—
C11	TL	—	V_{DD}IO_TL	V_{DD}IO_TL	V_{DD}IO_TL	—	—	—
C7	TL	—	V_{DD}IO_TL	V_{DD}IO_TL	V_{DD}IO_TL	—	—	—
C5	TL	—	PCFG_MPI_IRQ	PCFG_MPI_IRQ	PCFG_MPI_IRQ	$\overline{CFG\_IRQ}$/ $\overline{MPI\_IRQ}$	—	—
B3	TL	—	PCCLK	PCCLK	PCCLK	—	—	—
C4	TL	—	PDONE	PDONE	PDONE	—	—	—
A3	TL	—	V_{DD}33	V_{DD}33	V_{DD}33	—	—	—
A1	TL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
A2	TL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
A26	TL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AC13	TL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AC18	TL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AC23	TL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AC4	TL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AC8	TL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AD24	TL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AA23	TL	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
AA4	TL	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
A19	TC	1	PT21D	PT28D	PT35D	—	L1C_A1	COMPLEMENT
C19	TC	1	PT21C	PT28C	PT35C	—	L1T_A1	TRUE
B18	TC	1	PT20D	PT27D	PT34D	V_{REF}_TC_01	L2C_A0	COMPLEMENT
A18	TC	1	PT20C	PT27C	PT34C	—	L2T_A0	TRUE
B17	TC	1	PT20B	PT27B	PT33D	—	L3C_D0	COMPLEMENT
C18	TC	1	PT20A	PT27A	PT33C	—	L3T_D0	TRUE
A17	TC	2	PT19D	PT26D	PT32D	—	L4C_A2	COMPLEMENT
D17	TC	2	PT19C	PT26C	PT32C	V_{REF}_TC_02	L4T_A2	TRUE

Pin Information (continued)

Table 44. OR4E6 352-Pin PBGA Pinout (continued)

352 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
B16	TC	2	PT18D	PT25D	PT30D	—	L5C_D0	COMPLEMENT
C17	TC	2	PT18C	PT25C	PT30C	—	L5T_D0	TRUE
B15	TC	3	PT18B	PT24D	PT29D	—	L6C_A0	COMPLEMENT
A15	TC	3	PT18A	PT24C	PT29C	VREF_TC_03	L6T_A0	TRUE
C16	TC	3	PT17D	PT23D	PT28D	—	L7C_D1	COMPLEMENT
B14	TC	3	PT17C	PT23C	PT28C	—	L7T_D1	TRUE
D15	TC	4	PT16D	PT21D	PT26D	—	L8C_D2	COMPLEMENT
A14	TC	4	PT16C	PT21C	PT26C	—	L8T_D2	TRUE
C15	TC	4	PT15D	PT19D	PT24D	—	L9C_D1	COMPLEMENT
B13	TC	4	PT15C	PT19C	PT24C	VREF_TC_04	L9T_D1	TRUE
A13	TC	5	PT14D	PT18D	PT23D	PTCK1C	L10C_D1	COMPLEMENT
C14	TC	5	PT14C	PT18C	PT23C	PTCK1T	L10T_D1	TRUE
B12	TC	5	PT13D	PT17D	PT22D	PTCK0C	L11C_D0	COMPLEMENT
C13	TC	5	PT13C	PT17C	PT22C	PTCK0T	L11T_D0	TRUE
A12	TC	5	PT13B	PT16D	PT21D	VREF_TC_05	L12C_D0	COMPLEMENT
B11	TC	5	PT13A	PT16C	PT21C	—	L12T_D0	TRUE
C12	TC	6	PT12B	PT14D	PT19D	—	L13C_D1	COMPLEMENT
A11	TC	6	PT12A	PT14C	PT19C	VREF_TC_06	L13T_D1	TRUE
A16	TC	—	VDDIO_TC	VDDIO_TC	VDDIO_TC	—	—	—
D13	TC	—	VDDIO_TC	VDDIO_TC	VDDIO_TC	—	—	—
R15	TC	—	VSS	VSS	VSS	—	—	—
R16	TC	—	VSS	VSS	VSS	—	—	—
T11	TC	—	VSS	VSS	VSS	—	—	—
T12	TC	—	VSS	VSS	VSS	—	—	—
T13	TC	—	VSS	VSS	VSS	—	—	—
T14	TC	—	VSS	VSS	VSS	—	—	—
T15	TC	—	VSS	VSS	VSS	—	—	—
T16	TC	—	VSS	VSS	VSS	—	—	—
T23	TC	—	VDD15	VDD15	VDD15	—	—	—
T4	TC	—	VDD15	VDD15	VDD15	—	—	—
J24	TR	1	PR8C	PR11C	PR13C	—	L1T_D1	TRUE
G25	TR	1	PR8D	PR11D	PR13D	VREF_TR_01	L1C_D1	COMPLEMENT
H23	TR	1	PR7A	PR10C	PR12C	—	L2T_D2	TRUE
G26	TR	1	PR7B	PR10D	PR12D	—	L2C_D2	COMPLEMENT
H24	TR	1	PR7C	PR9C	PR11C	—	L3T_D1	TRUE
F25	TR	1	PR7D	PR9D	PR11D	—	L3C_D1	COMPLEMENT
G23	TR	2	PR6A	PR8C	PR10C	—	L4T_D2	TRUE
F26	TR	2	PR6B	PR8D	PR10D	—	L4C_D2	COMPLEMENT
E25	TR	2	PR6C	PR7C	PR9C	VREF_TR_02	L5T_A0	TRUE
E26	TR	2	PR6D	PR7D	PR9D	—	L5C_A0	COMPLEMENT
F24	TR	3	PR5C	PR6C	PR7C	—	L6T_D1	TRUE
D25	TR	3	PR5D	PR6D	PR7D	VREF_TR_03	L6C_D1	COMPLEMENT

Pin Information (continued)

Table 44. OR4E6 352-Pin PBGA Pinout (continued)

352 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
E23	TR	3	PR4C	PR5C	PR5C	—	L7T_D2	TRUE
D26	TR	3	PR4D	PR5D	PR5D	—	L7C_D2	COMPLEMENT
E24	TR	4	PR3C	PR3C	PR3C	PLL_CK3T/ PLL1(1.554/ 2.048 MHz)	L8T_D1	TRUE
C25	TR	4	PR3D	PR3D	PR3D	PLL_CK3C/ PLL1(1.554/ 2.048 MHz)	L8C_D1	COMPLEMENT
A24	TR	5	PT27D	PT37D	PT47D	PLL_CK2C/PPLL	L9C_A0	COMPLEMENT
B23	TR	5	PT27C	PT37C	PT47C	PLL_CK2T/PPLL	L9T_A0	TRUE
C23	TR	5	PT26D	PT36D	PT45D	VREF_TR_05	L10C_A1	COMPLEMENT
A23	TR	5	PT26C	PT36C	PT45C	—	L10T_A1	TRUE
B22	TR	6	PT26B	PT35B	PT43D	—	L11C_A1	COMPLEMENT
D22	TR	6	PT26A	PT35A	PT43C	—	L11T_A1	TRUE
C22	TR	6	PT25D	PT34D	PT42D	VREF_TR_06	L12C_A1	COMPLEMENT
A22	TR	6	PT25C	PT34C	PT42C	—	L12T_A1	TRUE
B21	TR	7	PT24D	PT33D	PT40D	—	L13C_D1	COMPLEMENT
D20	TR	7	PT24C	PT33C	PT40C	VREF_TR_07	L13T_D1	TRUE
A21	TR	7	PT24B	PT32D	PT39D	—	L14C_D0	COMPLEMENT
B20	TR	7	PT24A	PT32C	PT39C	—	L14T_D0	TRUE
A20	TR	8	PT23D	PT31D	PT38D	—	L15C_A1	COMPLEMENT
C20	TR	8	PT23C	PT31C	PT38C	VREF_TR_08	L15T_A1	TRUE
B19	TR	8	PT22D	PT29D	PT36D	—	L16C_D1	COMPLEMENT
D18	TR	8	PT22C	PT29C	PT36C	—	L16T_D1	TRUE
G24	TR	—	VDDIO_TR	VDDIO_TR	VDDIO_TR	—	—	—
D24	TR	—	VDDIO_TR	VDDIO_TR	VDDIO_TR	—	—	—
C26	TR	—	VDD33	VDD33	VDD33	—	—	—
A25	TR	—	VDD33	VDD33	VDD33	—	—	—
B24	TR	—	PLL_VF	PLL_VF	PLL_VF	—	—	—
C21	TR	—	VDDIO_TR	VDDIO_TR	VDDIO_TR	—	—	—
P12	TR	—	VSS	VSS	VSS	—	—	—
P13	TR	—	VSS	VSS	VSS	—	—	—
P14	TR	—	VSS	VSS	VSS	—	—	—
P15	TR	—	VSS	VSS	VSS	—	—	—
P16	TR	—	VSS	VSS	VSS	—	—	—
R11	TR	—	VSS	VSS	VSS	—	—	—
R12	TR	—	VSS	VSS	VSS	—	—	—
R13	TR	—	VSS	VSS	VSS	—	—	—
R14	TR	—	VSS	VSS	VSS	—	—	—
L23	TR	—	VDD15	VDD15	VDD15	—	—	—
L4	TR	—	VDD15	VDD15	VDD15	—	—	—
V25	CR	1	PR20C	PR29C	PR35C	—	L1T_A0	TRUE

Pin Information (continued)

Table 44. OR4E6 352-Pin PBGA Pinout (continued)

352 BGA Ball	$V_{DD}IO$ Bank	V_{REF} Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
V26	CR	1	PR20D	PR29D	PR35D	—	L1C_A0	COMPLEMENT
U25	CR	1	PR19C	PR28C	PR33C	V_{REF}_CR_01	L2T_D0	TRUE
V24	CR	1	PR19D	PR28D	PR33D	—	L2C_D0	COMPLEMENT
U23	CR	2	PR18C	PR26A	PR31C	—	L3T_D1	TRUE
T25	CR	2	PR18D	PR26B	PR31D	V_{REF}_CR_02	L3C_D1	COMPLEMENT
U24	CR	2	PR17A	PR25A	PR30C	—	L4T_D1	TRUE
T26	CR	2	PR17B	PR25B	PR30D	—	L4C_D1	COMPLEMENT
R25	CR	3	PR17C	PR25C	PR29C	—	L5T_A0	TRUE
R26	CR	3	PR17D	PR25D	PR29D	V_{REF}_CR_03	L5C_A0	COMPLEMENT
T24	CR	4	PR16C	PR23C	PR27C	PRCK1T	L6T_D1	TRUE
P25	CR	4	PR16D	PR23D	PR27D	PRCK1C	L6C_D1	COMPLEMENT
R23	CR	4	PR15A	PR22C	PR26C	—	L7T_D2	TRUE
P26	CR	4	PR15B	PR22D	PR26D	V_{REF}_CR_04	L7C_D2	COMPLEMENT
N25	CR	5	PR15C	PR21C	PR25C	—	L8T_A1	TRUE
N23	CR	5	PR15D	PR21D	PR25D	—	L8C_A1	COMPLEMENT
N26	CR	5	PR14A	PR20C	PR24C	PRCK0T	L9T_D1	TRUE
P24	CR	5	PR14B	PR20D	PR24D	PRCK0C	L9C_D1	COMPLEMENT
M25	CR	5	PR14C	PR19C	PR23C	V_{REF}_CR_05	L10T_D0	TRUE
N24	CR	5	PR14D	PR19D	PR23D	—	L10C_D0	COMPLEMENT
M26	CR	6	PR13C	PR17C	PR21C	—	L11T_D0	TRUE
L25	CR	6	PR13D	PR17D	PR21D	V_{REF}_CR_06	L11C_D0	COMPLEMENT
M24	CR	6	PR12A	PR16C	PR20C	—	L12T_D1	TRUE
L26	CR	6	PR12B	PR16D	PR20D	—	L12C_D1	COMPLEMENT
K25	CR	7	PR12C	PR15C	PR19C	—	L13T_D0	TRUE
L24	CR	7	PR12D	PR15D	PR19D	—	L13C_D0	COMPLEMENT
K26	CR	7	PR11B	PR14B	PR18D	—	—	COMPLEMENT
K23	CR	7	PR11C	PR14C	PR17C	V_{REF}_CR_07	L14T_D1	TRUE
J25	CR	7	PR11D	PR14D	PR17D	—	L14C_D1	COMPLEMENT
K24	CR	8	PR10C	PR13C	PR15C	—	L15T_D1	TRUE
J26	CR	8	PR10D	PR13D	PR15D	—	L15C_D1	COMPLEMENT
H25	CR	8	PR9C	PR12C	PR14C	V_{REF}_CR_08	L16T_A0	TRUE
H26	CR	8	PR9D	PR12D	PR14D	—	L16C_A0	COMPLEMENT
U26	CR	—	V_{DD}IO_CR	V_{DD}IO_CR	V_{DD}IO_CR	—	—	—
R24	CR	—	V_{DD}IO_CR	V_{DD}IO_CR	V_{DD}IO_CR	—	—	—
M23	CR	—	V_{DD}IO_CR	V_{DD}IO_CR	V_{DD}IO_CR	—	—	—
M16	CR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
N11	CR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
N12	CR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
N13	CR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
N14	CR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
N15	CR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
N16	CR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—

Pin Information (continued)

Table 44. OR4E6 352-Pin PBGA Pinout (continued)

352 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
P11	CR	—	VSS	VSS	VSS	—	—	—
F23	CR	—	VDD15	VDD15	VDD15	—	—	—
F4	CR	—	VDD15	VDD15	VDD15	—	—	—
AE20	BR	1	PB22A	PB30C	PB37C	—	L1T_D1	TRUE
AC19	BR	1	PB22B	PB30D	PB37D	—	L1C_D1	COMPLEMENT
AF20	BR	1	PB22C	PB31C	PB38C	VREF_BR_01	L2T_D1	TRUE
AD19	BR	1	PB22D	PB31D	PB38D	—	L2C_D1	COMPLEMENT
AE21	BR	1	PB23A	PB32C	PB39C	—	L3T_D1	TRUE
AC20	BR	1	PB23B	PB32D	PB39D	—	L3C_D1	COMPLEMENT
AD20	BR	2	PB23C	PB33C	PB40C	—	L4T_D1	TRUE
AE22	BR	2	PB23D	PB33D	PB40D	VREF_BR_02	L4C_D1	COMPLEMENT
AF22	BR	2	PB24C	PB34C	PB42C	—	—	TRUE
AD21	BR	3	PB25A	PB35A	PB43A	—	—	TRUE
AE23	BR	3	PB25C	PB35C	PB44C	—	L5T_D1	TRUE
AC22	BR	3	PB25D	PB35D	PB44D	VREF_BR_03	L5C_D1	COMPLEMENT
AF23	BR	3	PB26C	PB36C	PB45C	—	L6T_D1	TRUE
AD22	BR	3	PB26D	PB36D	PB45D	—	L6C_D1	COMPLEMENT
AE24	BR	4	PB27C	PB37C	PB47C	PLL_CK5T/PPLL	L7T_D0	TRUE
AD23	BR	4	PB27D	PB37D	PB47D	PLL_CK5C/PPLL	L7C_D0	COMPLEMENT
AD26	BR	5	PR26A	PR38A	PR46C	PLL_CK4T/PLL2 (155.52 MHz)	L8T_D0	TRUE
AC25	BR	5	PR26B	PR38B	PR46D	PLL_CK4C/PLL2 (155.52 MHz)	L8C_D0	COMPLEMENT
AC24	BR	5	PR25A	PR37C	PR44C	VREF_BR_05	L9T_A1	TRUE
AC26	BR	5	PR25B	PR37D	PR44D	—	L9C_A1	COMPLEMENT
AB25	BR	6	PR25C	PR36C	PR43C	—	L10T_A1	TRUE
AB23	BR	6	PR25D	PR36D	PR43D	—	L10C_A1	COMPLEMENT
AB26	BR	6	PR24C	PR35C	PR41C	VREF_BR_06	L11T_D0	TRUE
AA25	BR	6	PR24D	PR35D	PR41D	—	L11C_D0	COMPLEMENT
Y23	BR	7	PR23A	PR34C	PR40C	—	L12T_D0	TRUE
AA24	BR	7	PR23B	PR34D	PR40D	—	L12C_D0	COMPLEMENT
AA26	BR	7	PR23C	PR33C	PR39C	—	L13T_D0	TRUE
Y25	BR	7	PR23D	PR33D	PR39D	VREF_BR_07	L13C_D0	COMPLEMENT
Y26	BR	7	PR22A	PR32C	PR38C	—	L14T_A1	TRUE
Y24	BR	7	PR22B	PR32D	PR38D	—	L14C_A1	COMPLEMENT
W25	BR	8	PR22C	PR31C	PR37C	—	L15T_D1	TRUE
V23	BR	8	PR22D	PR31D	PR37D	VREF_BR_08	L15C_D1	COMPLEMENT
W26	BR	8	PR21C	PR30C	PR36C	—	L16T_A1	TRUE
W24	BR	8	PR21D	PR30D	PR36D	—	L16C_A1	COMPLEMENT
AF21	BR	—	VDDIO_BR	VDDIO_BR	VDDIO_BR	—	—	—
AF24	BR	—	VDD33	VDD33	VDD33	—	—	—
AE26	BR	—	VDD33	VDD33	VDD33	—	—	—

Pin Information (continued)

Table 44. OR4E6 352-Pin PBGA Pinout (continued)

352 BGA Ball	$V_{DD}IO$ Bank	V_{REF} Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
AD25	BR	—	$V_{DD}IO\_BR$	$V_{DD}IO\_BR$	$V_{DD}IO\_BR$	—	—	—
AB24	BR	—	$V_{DD}IO\_BR$	$V_{DD}IO\_BR$	$V_{DD}IO\_BR$	—	—	—
L13	BR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
L14	BR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
L15	BR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
L16	BR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
M11	BR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
M12	BR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
M13	BR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
M14	BR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
M15	BR	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
D21	BR	—	$V_{DD}15$	$V_{DD}15$	$V_{DD}15$	—	—	—
D6	BR	—	$V_{DD}15$	$V_{DD}15$	$V_{DD}15$	—	—	—
AD11	BC	1	PB13A	PB17C	PB21C	—	L1T_D1	TRUE
AE13	BC	1	PB13B	PB17D	PB21D	—	L1C_D1	COMPLEMENT
AC12	BC	1	PB13C	PB18C	PB22C	$V_{REF}\_BC\_01$	L2T_D2	TRUE
AF13	BC	1	PB13D	PB18D	PB22D	—	L2C_D2	COMPLEMENT
AD12	BC	2	PB14C	PB19C	PB23C	PBCK0T	L3T_D1	TRUE
AE14	BC	2	PB14D	PB19D	PB23D	PBCK0C	L3C_D1	COMPLEMENT
AF14	BC	2	PB15C	PB20C	PB24C	$V_{REF}\_BC\_02$	L4T_D1	TRUE
AD13	BC	2	PB15D	PB20D	PB24D	—	L4C_D1	COMPLEMENT
AE15	BC	3	PB16C	PB21C	PB26C	—	L5T_D0	TRUE
AD14	BC	3	PB16D	PB21D	PB26D	$V_{REF}\_BC\_03$	L5C_D0	COMPLEMENT
AF15	BC	3	PB17A	PB22C	PB27C	—	L6T_D0	TRUE
AE16	BC	3	PB17B	PB22D	PB27D	—	L6C_D0	COMPLEMENT
AD15	BC	3	PB17C	PB23C	PB28C	PBCK1T	L7T_D1	TRUE
AF16	BC	3	PB17D	PB23D	PB28D	PBCK1C	L7C_D1	COMPLEMENT
AC15	BC	4	PB18A	PB24C	PB29C	—	L8T_D1	TRUE
AE17	BC	4	PB18B	PB24D	PB29D	—	L8C_D1	COMPLEMENT
AF17	BC	4	PB18C	PB25C	PB30C	—	L9T_A2	TRUE
AC17	BC	4	PB18D	PB25D	PB30D	$V_{REF}\_BC\_04$	L9C_A2	COMPLEMENT
AE18	BC	5	PB19C	PB26C	PB32C	—	L10T_D0	TRUE
AD17	BC	5	PB19D	PB26D	PB32D	$V_{REF}\_BC\_05$	L10C_D0	COMPLEMENT
AF18	BC	5	PB20C	PB27C	PB34C	—	L11T_D0	TRUE
AE19	BC	5	PB20D	PB27D	PB34D	—	L11C_D0	COMPLEMENT
AF19	BC	6	PB21A	PB28C	PB35C	—	L12T_D1	TRUE
AD18	BC	6	PB21B	PB28D	PB35D	$V_{REF}\_BC\_06$	L12C_D1	COMPLEMENT
AC14	BC	—	$V_{DD}IO\_BC$	$V_{DD}IO\_BC$	$V_{DD}IO\_BC$	—	—	—
AD16	BC	—	$V_{DD}IO\_BC$	$V_{DD}IO\_BC$	$V_{DD}IO\_BC$	—	—	—
H4	BC	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
J23	BC	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
N4	BC	—	V_{SS}	V_{SS}	V_{SS}	—	—	—

Pin Information (continued)

Table 44. OR4E6 352-Pin PBGA Pinout (continued)

352 BGA Ball	$V_{DD}IO$ Bank	V_{REF} Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
P23	BC	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
V4	BC	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
W23	BC	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
L11	BC	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
L12	BC	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
D11	BC	—	$V_{DD}15$	$V_{DD}15$	$V_{DD}15$	—	—	—
D16	BC	—	$V_{DD}15$	$V_{DD}15$	$V_{DD}15$	—	—	—
Y1	BL	1	PL22D	PL32D	PL38D	D8	L1C_D1	COMPLEMENT
W3	BL	1	PL22C	PL32C	PL38C	V_{REF}_BL_01	L1T_D1	TRUE
AA2	BL	1	PL22B	PL33D	PL39D	D9	L2C_D1	COMPLEMENT
Y4	BL	1	PL22A	PL33C	PL39C	D10	L2T_D1	TRUE
AA1	BL	2	PL23C	PL34C	PL40C	V_{REF}_BL_02	—	TRUE
AB2	BL	3	PL24D	PL35B	PL42D	D11	L3C_A0	COMPLEMENT
AB1	BL	3	PL24C	PL35A	PL42C	D12	L3T_A0	TRUE
AA3	BL	3	PL25D	PL36B	PL44D	V_{REF}_BL_03	L4C_D1	COMPLEMENT
AC2	BL	3	PL25C	PL36A	PL44C	D13	L4T_D1	TRUE
AB4	BL	4	PL27D	PL39D	PL47D	PLL_CK7C/ HPPLL	L5C_D2	COMPLEMENT
AC1	BL	4	PL27C	PL39C	PL47C	PLL_CK7T/ HPPLL	L5T_D2	TRUE
AE3	BL	5	PB2A	PB2A	PB2A	DP2	—	TRUE
AF3	BL	5	PB2C	PB2C	PB2C	PLL_CK6T/PPLL	L6T_A0	TRUE
AE4	BL	5	PB2D	PB2D	PB2D	PLL_CK6C/PPLL	L6C_A0	COMPLEMENT
AD4	BL	5	PB3C	PB4A	PB4C	V_{REF}_BL_05	L7T_A1	TRUE
AF4	BL	5	PB3D	PB4B	PB4D	DP3	L7C_A1	COMPLEMENT
AE5	BL	6	PB4C	PB5C	PB6C	V_{REF}_BL_06	L8T_A1	TRUE
AC5	BL	6	PB4D	PB5D	PB6D	D14	L8C_A1	COMPLEMENT
AF5	BL	7	PB5C	PB6C	PB8C	D15	L9T_D0	TRUE
AE6	BL	7	PB5D	PB6D	PB8D	D16	L9C_D0	COMPLEMENT
AC7	BL	7	PB6A	PB7C	PB9C	D17	L10T_D0	TRUE
AD6	BL	7	PB6B	PB7D	PB9D	D18	L10C_D0	COMPLEMENT
AF6	BL	7	PB6C	PB8C	PB10C	V_{REF}_BL_07	L11T_D0	TRUE
AE7	BL	7	PB6D	PB8D	PB10D	D19	L11C_D0	COMPLEMENT
AF7	BL	8	PB7A	PB9C	PB11C	D20	L12T_A1	TRUE
AD7	BL	8	PB7B	PB9D	PB11D	D21	L12C_A1	COMPLEMENT
AE8	BL	8	PB7C	PB10C	PB12C	V_{REF}_BL_08	L13T_D1	TRUE
AC9	BL	8	PB7D	PB10D	PB12D	D22	L13C_D1	COMPLEMENT
AF8	BL	9	PB8C	PB11C	PB13C	D23	L14T_A1	TRUE
AD8	BL	9	PB8D	PB11D	PB13D	D24	L14C_A1	COMPLEMENT
AE9	BL	9	PB9C	PB12C	PB14C	V_{REF}_BL_09	L15T_A0	TRUE
AF9	BL	9	PB9D	PB12D	PB14D	D25	L15C_A0	COMPLEMENT
AE10	BL	10	PB10C	PB13C	PB16C	D26	L16T_D0	TRUE

Pin Information (continued)

Table 44. OR4E6 352-Pin PBGA Pinout (continued)

352 BGA Ball	$V_{DD}IO$ Bank	V_{REF} Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
AD9	BL	10	PB10D	PB13D	PB16D	D27	L16C_D0	COMPLEMENT
AC10	BL	10	PB11C	PB14C	PB18C	V_{REF}_BL_10	L17T_D1	TRUE
AE11	BL	10	PB11D	PB14D	PB18D	D28	L17C_D1	COMPLEMENT
AD10	BL	11	PB12A	PB15C	PB19C	D29	L18T_D1	TRUE
AF11	BL	11	PB12B	PB15D	PB19D	D30	L18C_D1	COMPLEMENT
AE12	BL	11	PB12C	PB16C	PB20C	V_{REF}_BL_11	L19T_A0	TRUE
AF12	BL	11	PB12D	PB16D	PB20D	D31	L19C_A0	COMPLEMENT
Y3	BL	—	$V_{DD}IO$_BL	$V_{DD}IO$_BL	$V_{DD}IO$_BL	—	—	—
AB3	BL	—	PTEMP	PTEMP	PTEMP	—	—	—
AD2	BL	—	$V_{DD}IO$_BL	$V_{DD}IO$_BL	$V_{DD}IO$_BL	—	—	—
AC3	BL	—	LVDS_R	LVDS_R	LVDS_R	—	—	—
AD1	BL	—	V_{DD}33	V_{DD}33	V_{DD}33	—	—	—
AF2	BL	—	V_{DD}33	V_{DD}33	V_{DD}33	—	—	—
AD5	BL	—	$V_{DD}IO$_BL	$V_{DD}IO$_BL	$V_{DD}IO$_BL	—	—	—
AF10	BL	—	$V_{DD}IO$_BL	$V_{DD}IO$_BL	$V_{DD}IO$_BL	—	—	—
B25	BL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
B26	BL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
C24	BL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
C3	BL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
D14	BL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
D19	BL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
D23	BL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
D4	BL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
D9	BL	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AC21	BL	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
AC6	BL	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
K2	CL	1	PL8D	PL12D	PL14D	A15	L1C_D0	COMPLEMENT
J3	CL	1	PL8C	PL12C	PL14C	A14	L1T_D0	TRUE
K1	CL	1	PL9D	PL13D	PL16D	V_{REF}_CL_01	L2C_A2	COMPLEMENT
K4	CL	1	PL9C	PL13C	PL16C	D4	L2T_A2	TRUE
L2	CL	2	PL10D	PL14D	PL18D	RDY/$\overline{\text{BUSY}}$/RCLK	L3C_D0	COMPLEMENT
K3	CL	2	PL10C	PL14C	PL18C	V_{REF}_CL_02	L3T_D0	TRUE
M2	CL	2	PL10B	PL15D	PL19D	A13	L4C_A0	COMPLEMENT
M1	CL	2	PL10A	PL15C	PL19C	A12	L4T_A0	TRUE
L3	CL	3	PL11B	PL17D	PL21D	A11	L5C_D1	COMPLEMENT
N2	CL	3	PL11A	PL17C	PL21C	V_{REF}_CL_03	L5T_D1	TRUE
M4	CL	4	PL13D	PL19D	PL23D	$\overline{\text{RD}}$/$\overline{\text{MPI\_STRB}}$	L6C_D2	COMPLEMENT
N1	CL	4	PL13C	PL19C	PL23C	V_{REF}_CL_04	L6T_D2	TRUE
M3	CL	4	PL14D	PL20D	PL24D	PLCK0C	L7C_D1	COMPLEMENT
P2	CL	4	PL14C	PL20C	PL24C	PLCK0T	L7T_D1	TRUE
P1	CL	5	PL15D	PL21D	PL25D	A10	L8C_D1	COMPLEMENT

Pin Information (continued)

Table 44. OR4E6 352-Pin PBGA Pinout (continued)

352 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
N3	CL	5	PL15C	PL21C	PL25C	A9	L8T_D1	TRUE
R2	CL	5	PL16D	PL22D	PL26D	A8	L9C_D0	COMPLEMENT
P3	CL	5	PL16C	PL22C	PL26C	VREF_CL_05	L9T_D0	TRUE
R1	CL	6	PL17D	PL24D	PL28D	PLCK1C	L10C_D0	COMPLEMENT
T2	CL	6	PL17C	PL24C	PL28C	PLCK1T	L10T_D0	TRUE
R3	CL	6	PL17B	PL25D	PL29D	VREF_CL_06	L11C_D1	COMPLEMENT
T1	CL	6	PL17A	PL25C	PL29C	A7	L11T_D1	TRUE
R4	CL	6	PL18D	PL26D	PL30D	A6	L12C_D1	COMPLEMENT
U2	CL	6	PL18C	PL26C	PL30C	A5	L12T_D1	TRUE
U1	CL	7	PL19D	PL27D	PL32D	$\overline{WR}$/MPI_RW	L13C_A2	COMPLEMENT
U4	CL	7	PL19C	PL27C	PL32C	VREF_CL_07	L13T_A2	TRUE
V2	CL	8	PL20D	PL28D	PL34D	A4	L14C_D1	COMPLEMENT
U3	CL	8	PL20C	PL28C	PL34C	VREF_CL_08	L14T_D1	TRUE
V1	CL	8	PL20B	PL29D	PL35D	A3	L15C_D0	COMPLEMENT
W2	CL	8	PL20A	PL29C	PL35C	A2	L15T_D0	TRUE
W1	CL	8	PL21D	PL30D	PL36D	A1	L16C_D1	COMPLEMENT
V3	CL	8	PL21C	PL30C	PL36C	A0	L16T_D1	TRUE
Y2	CL	8	PL21B	PL31D	PL37D	DP0	L17C_D1	COMPLEMENT
W4	CL	8	PL21A	PL31C	PL37C	DP1	L17T_D1	TRUE
L1	CL	—	VDDIO_CL	VDDIO_CL	VDDIO_CL	—	—	—
P4	CL	—	VDDIO_CL	VDDIO_CL	VDDIO_CL	—	—	—
T3	CL	—	VDDIO_CL	VDDIO_CL	VDDIO_CL	—	—	—
AD3	CL	—	VSS	VSS	VSS	—	—	—
AE1	CL	—	VSS	VSS	VSS	—	—	—
AE2	CL	—	VSS	VSS	VSS	—	—	—
AE25	CL	—	VSS	VSS	VSS	—	—	—
AF1	CL	—	VSS	VSS	VSS	—	—	—
AF25	CL	—	VSS	VSS	VSS	—	—	—
AF26	CL	—	VSS	VSS	VSS	—	—	—
B2	CL	—	VSS	VSS	VSS	—	—	—
AC11	CL	—	VDD15	VDD15	VDD15	—	—	—
AC16	CL	—	VDD15	VDD15	VDD15	—	—	—

Pin Information (continued)

As shown in the Pair column, differential pairs and physical locations are numbered within each bank (e.g., L19C_A0 is the nineteenth pair in an associated bank). The C indicates complementary differential whereas a T indicates true differential. The _A0 indicates the physical location of adjacent balls in either the horizontal or vertical direction. Other physical indicators are as follows:

- _A1 indicates one ball between pairs.
- _A2 indicates two balls between pairs.
- _D0 indicates balls are diagonally adjacent.
- _D1 indicates diagonally adjacent separated by one physical ball.

VREF pins, shown in the Additional Function column, are associated to the bank and group (e.g., VREF_TL_01 is the VREF for group one of the top left (TL) bank).

Table 45. 432-Pin EBGA

432 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
B19	0	1	PT11D	PT13D	PT18D	$\overline{\text{MPI\_RTRY}}$	L1C_A0	COMPLEMENT
C19	0	1	PT11C	PT13C	PT18C	$\overline{\text{MPI\_ACK}}$	L1T_A0	TRUE
D19	0	1	PT11B	PT13B	PT17D	—	L2C_D0	COMPLEMENT
C20	0	1	PT11A	PT13A	PT17C	VREF_TL_01	L2T_D0	TRUE
A21	0	1	PT10D	PT12D	PT16D	M0	L3C_A0	COMPLEMENT
B21	0	1	PT10C	PT12C	PT16C	M1	L3T_A0	TRUE
A22	0	2	PT9D	PT11D	PT14D	M2	L5C_A0	COMPLEMENT
B22	0	2	PT9C	PT11C	PT14C	M3	L5T_A0	TRUE
C22	0	2	PT9B	PT11B	PT13D	VREF_TL_02	L6C_D1	COMPLEMENT
A23	0	2	PT9A	PT11A	PT13C	$\overline{\text{MPI\_TEA}}$	L6T_D1	TRUE
D20	0	2	PT10B	PT12B	PT15D	MPI_CLK	L4C_D0	COMPLEMENT
C21	0	2	PT10A	PT12A	PT15C	A21/$\overline{\text{MPI\_BURST}}$	L4T_D0	TRUE
B23	0	3	PT8B	PT9D	PT11D	VREF_TL_03	L7C_D1	COMPLEMENT
D22	0	3	PT8A	PT9C	PT11C	—	L7T_D1	TRUE
C23	0	3	PT7D	PT8D	PT10D	D0	L8C_D0	COMPLEMENT
B24	0	3	PT7C	PT8C	PT10C	TMS	L8T_D0	TRUE
C24	0	4	PT7B	PT7D	PT9D	A20/$\overline{\text{MPI\_BDIP}}$	L9C_D0	COMPLEMENT
D23	0	4	PT7A	PT7C	PT9C	A19/MPI_TSZ1	L9T_D0	TRUE
A25	0	4	PT6D	PT6D	PT8D	A18/MPI_TSZ0	L10C_A0	COMPLEMENT
B25	0	4	PT6C	PT6C	PT8C	D3	L10T_A0	TRUE
D24	0	5	PT5D	PT5D	PT6D	D1	L11C_D2	COMPLEMENT
A26	0	5	PT5C	PT5C	PT6C	D2	L11T_D2	TRUE
B26	0	5	PT4D	PT4D	PT4D	TDI	L12C_A0	COMPLEMENT
C26	0	5	PT4C	PT4C	PT4C	TCK	L12T_A0	TRUE
A27	0	6	PT3D	PT3D	PT3D	—	L13C_A0	COMPLEMENT
B27	0	6	PT3C	PT3C	PT3C	VREF_TL_06	L13T_A0	TRUE
C27	0	6	PT2D	PT2D	PT2D	PLL_CK1C	L14C_D0	COMPLEMENT
D26	0	6	PT2C	PT2C	PT2C	PLL_CK1T	L14T_D0	TRUE
F31	0	7	PL3D	PL4D	PL4D	D5	L17C_D2	COMPLEMENT
H28	0	7	PL3C	PL4C	PL4C	D6	L17T_D2	TRUE
E30	0	7	PL2D	PL2D	PL2D	PLL_CK0C	L15C_A0	COMPLEMENT
E31	0	7	PL2C	PL2C	PL2C	PLL_CK0T	L15T_A0	TRUE

Pin Information (continued)

Table 45. OR4E6 432-Pin EBGA (continued)

432 BGA Ball	$V_{DD}IO$ Bank	V_{REF} Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
F29	0	7	PL2B	PL3D	PL3D	—	L16C_A0	COMPLEMENT
F30	0	7	PL2A	PL3C	PL3C	V_{REF}_TL_07	L16T_A0	TRUE
G29	0	8	PL4D	PL5D	PL6D	HDC	L18C_A0	COMPLEMENT
G30	0	8	PL4C	PL5C	PL6C	$\overline{\text{LDC}}$	L18T_A0	TRUE
K28	0	9	PL6D	PL8D	PL10D	$\overline{\text{CS0}}$	L21C_D1	COMPLEMENT
J30	0	9	PL6C	PL8C	PL10C	CS1	L21T_D1	TRUE
G31	0	9	PL5D	PL6D	PL8D		L19C_D2	COMPLEMENT
J28	0	9	PL5C	PL6C	PL8C	D7	L19T_D2	TRUE
H30	0	9	PL5B	PL7D	PL9D	V_{REF}_TL_09	L20C_D0	COMPLEMENT
J29	0	9	PL5A	PL7C	PL9C	A17	L20T_D0	TRUE
K30	0	10	PL7D	PL10D	PL12D	$\overline{\text{INIT}}$	L23C_A0	COMPLEMENT
K31	0	10	PL7C	PL10C	PL12C	DOUT	L23T_A0	TRUE
L29	0	10	PL7B	PL11D	PL13D	V_{REF}_TL_10	L24C_D0	COMPLEMENT
M28	0	10	PL7A	PL11C	PL13C	A16	L24T_D0	TRUE
J31	0	10	PL6B	PL9D	PL11D	—	L22C_D1	COMPLEMENT
K29	0	10	PL6A	PL9C	PL11C	—	L22T_D1	TRUE
A12	0	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
A16	0	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
A2	0	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
A20	0	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
A24	0	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
A29	0	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
H29	0	—	V_{DD}IO0_TL	V_{DD}IO0_TL	V_{DD}IO0_TL	—	—	—
E29	0	—	V_{DD}IO0_TL	V_{DD}IO0_TL	V_{DD}IO0_TL	—	—	—
C25	0	—	V_{DD}IO0_TL	V_{DD}IO0_TL	V_{DD}IO0_TL	—	—	—
B20	0	—	V_{DD}IO0_TL	V_{DD}IO0_TL	V_{DD}IO0_TL	—	—	—
E28	0	—	V_{DD}33	V_{DD}33	V_{DD}33	—	—	—
D27	0	—	V_{DD}33	V_{DD}33	V_{DD}33	—	—	—
A1	0	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
A31	0	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
AA28	0	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
AA4	0	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
AE28	0	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
D30	0	—	$\overline{\text{PRESET}}$	$\overline{\text{PRESET}}$	$\overline{\text{PRESET}}$	—	—	—
D29	0	—	PRD_DATA	PRD_DATA	PRD_DATA	TDO	—	—
D31	0	—	$\overline{\text{PRD\_CFG}}$	$\overline{\text{PRD\_CFG}}$	$\overline{\text{PRD\_CFG}}$	—	—	—
F28	0	—	$\overline{\text{PPRGRM}}$	$\overline{\text{PPRGRM}}$	$\overline{\text{PPRGRM}}$	—	—	—
C28	0	—	PDONE	PDONE	PDONE	—	—	—
A28	0	—	PCFG_MPI_IRQ	PCFG_MPI_IRQ	PCFG_MPI_IRQ	$\overline{\text{CFG\_IRQ}}$/$\overline{\text{MPI\_IRQ}}$	—	—
B28	0	—	PCCLK	PCCLK	PCCLK	—	—	—
A9	1	1	PT21D	PT28D	PT35D	—	L1C_D1	COMPLEMENT
C10	1	1	PT21C	PT28C	PT35C	—	L1T_D1	TRUE
B10	1	1	PT20D	PT27D	PT34D	V_{REF}_TC_01	L2C_A0	COMPLEMENT
A10	1	1	PT20C	PT27C	PT34C	—	L2T_A0	TRUE

Pin Information (continued)

Table 45. OR4E6 432-Pin EBGA (continued)

432 BGA Ball	$V_{DD}IO$ Bank	V_{REF} Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
C11	1	1	PT20B	PT27B	PT33D	—	L3C_D0	COMPLEMENT
D12	1	1	PT20A	PT27A	PT33C	—	L3T_D0	TRUE
B11	1	2	PT19D	PT26D	PT32D	—	L4C_A0	COMPLEMENT
A11	1	2	PT19C	PT26C	PT32C	V_{REF}_TC_02	L4T_A0	TRUE
C12	1	2	PT19B	PT26B	PT31D	—	L5C_D0	COMPLEMENT
D13	1	2	PT19A	PT26A	PT31C	—	L5T_D0	TRUE
B12	1	2	PT18D	PT25D	PT30D	—	L6C_D0	COMPLEMENT
C13	1	2	PT18C	PT25C	PT30C	—	L6T_D0	TRUE
D14	1	3	PT18B	PT24D	PT29D	—	L7C_D2	COMPLEMENT
A13	1	3	PT18A	PT24C	PT29C	V_{REF}_TC_03	L7T_D2	TRUE
C14	1	3	PT17D	PT23D	PT28D	—	L8C_A0	COMPLEMENT
B14	1	3	PT17C	PT23C	PT28C	—	L8T_A0	TRUE
A14	1	4	PT16D	PT21D	PT26D	—	L9C_D1	COMPLEMENT
C15	1	4	PT16C	PT21C	PT26C	—	L9T_D1	TRUE
B15	1	4	PT15D	PT19D	PT24D	—	L10C_A0	COMPLEMENT
A15	1	4	PT15C	PT19C	PT24C	V_{REF}_TC_04	L10T_A0	TRUE
D16	1	5	PT14D	PT18D	PT23D	PTCK1C	L11C_A1	COMPLEMENT
B16	1	5	PT14C	PT18C	PT23C	PTCK1T	L11T_A1	TRUE
A17	1	5	PT13D	PT17D	PT22D	PTCK0C	L12C_A0	COMPLEMENT
B17	1	5	PT13C	PT17C	PT22C	PTCK0T	L12T_A0	TRUE
C17	1	5	PT13B	PT16D	PT21D	V_{REF}_TC_05	L13C_D1	COMPLEMENT
A18	1	5	PT13A	PT16C	PT21C	—	L13T_D1	TRUE
B18	1	6	PT12D	PT15D	PT20D	—	L14C_A0	COMPLEMENT
C18	1	6	PT12C	PT15C	PT20C	—	L14T_A0	TRUE
A19	1	6	PT12B	PT14D	PT19D	—	L15C_D2	COMPLEMENT
D18	1	6	PT12A	PT14C	PT19C	V_{REF}_TC_06	L15T_D2	TRUE
M31	1	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
T1	1	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
T31	1	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
Y1	1	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
Y31	1	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
C16	1	—	V_{DD}IO_TC	V_{DD}IO_TC	V_{DD}IO_TC	—	—	—
B13	1	—	V_{DD}IO_TC	V_{DD}IO_TC	V_{DD}IO_TC	—	—	—
L4	1	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
R28	1	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
R4	1	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
R4	1	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
U28	1	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
J1	2	1	PR8D	PR11D	PR13D	V_{REF}_TR_01	L1C_D1	COMPLEMENT
K3	2	1	PR8C	PR11C	PR13C	—	L1T_D1	TRUE
H2	2	1	PR7D	PR9D	PR11D	—	L3C_D0	COMPLEMENT
J3	2	1	PR7C	PR9C	PR11C	—	L3T_D0	TRUE
K4	2	1	PR7B	PR10D	PR12D	—	L2C_D1	COMPLEMENT
J2	2	1	PR7A	PR10C	PR12C	—	L2T_D1	TRUE
G3	2	2	PR6D	PR7D	PR9D	—	L5C_A0	COMPLEMENT

Pin Information (continued)

Table 45. OR4E6 432-Pin EBGA (continued)

432 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
G2	2	2	PR6C	PR7C	PR9C	VREF_TR_02	L5T_A0	TRUE
J4	2	2	PR6B	PR8D	PR10D	—	L4C_D0	COMPLEMENT
H3	2	2	PR6A	PR8C	PR10C	—	L4T_D0	TRUE
F1	2	2	PR5B	PR6B	PR8D	—	L6C_D2	COMPLEMENT
H4	2	2	PR5A	PR6A	PR8C	—	L6T_D2	TRUE
F3	2	3	PR5D	PR6D	PR7D	VREF_TR_03	L7C_A0	COMPLEMENT
F2	2	3	PR5C	PR6C	PR7C	—	L7T_A0	TRUE
F4	2	3	PR4D	PR4D	PR5D	—	L9C_D0	COMPLEMENT
E3	2	3	PR4C	PR4C	PR5C	—	L9T_D0	TRUE
E2	2	3	PR4B	PR5D	PR6D	—	L8C_A0	COMPLEMENT
E1	2	3	PR4A	PR5C	PR6C	—	L8T_A0	TRUE
D2	2	4	PR3D	PR3D	PR3D	PLL_CK3C	L10C_A0	COMPLEMENT
D1	2	4	PR3C	PR3C	PR3C	PLL_CK3T	L10T_A0	TRUE
B4	2	5	PT27D	PT37D	PT47D	PLL_CK2C	L11C_A0	COMPLEMENT
A4	2	5	PT27C	PT37C	PT47C	PLL_CK2T	L11T_A0	TRUE
D6	2	5	PT26D	PT36D	PT45D	VREF_TR_05	L12C_D0	COMPLEMENT
C5	2	5	PT26C	PT36C	PT45C	—	L12T_D0	TRUE
B5	2	6	PT26B	PT35B	PT43D	—	L13C_A0	COMPLEMENT
A5	2	6	PT26A	PT35A	PT43C	—	L13T_A0	TRUE
C6	2	6	PT25D	PT34D	PT42D	VREF_TR_06	L14C_A0	COMPLEMENT
B6	2	6	PT25C	PT34C	PT42C	—	L14T_A0	TRUE
A6	2	7	PT25B	PT34B	PT41D	—	L15C_D2	COMPLEMENT
D8	2	7	PT25A	PT34A	PT41C	—	L15T_D2	TRUE
C7	2	7	PT24D	PT33D	PT40D	—	L16C_A0	COMPLEMENT
B7	2	7	PT24C	PT33C	PT40C	VREF_TR_07	L16T_A0	TRUE
D9	2	7	PT24B	PT32D	PT39D	—	L17C_D0	COMPLEMENT
C8	2	7	PT24A	PT32C	PT39C	—	L17T_D0	TRUE
B8	2	8	PT23D	PT31D	PT38D	—	L18C_D0	COMPLEMENT
C9	2	8	PT23C	PT31C	PT38C	VREF_TR_08	L18T_D0	TRUE
D10	2	8	PT22D	PT29D	PT36D	—	L19C_D1	COMPLEMENT
B9	2	8	PT22C	PT29C	PT36C	—	L19T_D1	TRUE
C2	2	—	VSS	VSS	VSS	—	—	—
C30	2	—	VSS	VSS	VSS	—	—	—
C31	2	—	VSS	VSS	VSS	—	—	—
H1	2	—	VSS	VSS	VSS	—	—	—
H31	2	—	VSS	VSS	VSS	—	—	—
M1	2	—	VSS	VSS	VSS	—	—	—
D3	2	—	VDDIO_TR	VDDIO_TR	VDDIO_TR	—	—	—
G1	2	—	VDDIO_TR	VDDIO_TR	VDDIO_TR	—	—	—
A7	2	—	VDDIO_TR	VDDIO_TR	VDDIO_TR	—	—	—
E4	2	—	VDD33	VDD33	VDD33	—	—	—
D5	2	—	VDD33	VDD33	VDD33	—	—	—
D4	2	—	VDD15	VDD15	VDD15	—	—	—
D7	2	—	VDD15	VDD15	VDD15	—	—	—

Pin Information (continued)

Table 45. OR4E6 432-Pin EBGA (continued)

432 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
G28	2	—	VDD15	VDD15	VDD15	—	—	—
G4	2	—	VDD15	VDD15	VDD15	—	—	—
L28	2	—	VDD15	VDD15	VDD15	—	—	—
C4	2	—	PLL_VF	PLL_VF	PLL_VF	—	—	—
AB1	3	1	PR20D	PR29D	PR35D	—	L1C_A0	COMPLEMENT
AB2	3	1	PR20C	PR29C	PR35C	—	L1T_A0	TRUE
Y4	3	1	PR19D	PR28D	PR33D	—	L2C_D0	COMPLEMENT
AA3	3	1	PR19C	PR28C	PR33C	VREF_CR_01	L2T_D0	TRUE
Y2	3	2	PR18D	PR26B	PR31D	VREF_CR_02	L4C_D1	COMPLEMENT
W4	3	2	PR18C	PR26A	PR31C	—	L4T_D1	TRUE
AA1	3	2	PR18B	PR27B	PR32D	—	L3C_A0	COMPLEMENT
AA2	3	2	PR18A	PR27A	PR32C	—	L3T_A0	TRUE
W2	3	2	PR17B	PR25B	PR30D	—	L5C_A0	COMPLEMENT
W3	3	2	PR17A	PR25A	PR30C	—	L5T_A0	TRUE
W1	3	3	PR17D	PR25D	PR29D	VREF_CR_03	L6C_D2	COMPLEMENT
V4	3	3	PR17C	PR25C	PR29C	—	L6T_D2	TRUE
V2	3	4	PR16D	PR23D	PR27D	PRCK1C	L7C_A0	COMPLEMENT
V3	3	4	PR16C	PR23C	PR27C	PRCK1T	L7T_A0	TRUE
U3	3	4	PR15B	PR22D	PR26D	VREF_CR_04	L8C_D1	COMPLEMENT
V1	3	4	PR15A	PR22C	PR26C	—	L8T_D1	TRUE
T3	3	5	PR15D	PR21D	PR25D	—	L9C_D1	COMPLEMENT
U1	3	5	PR15C	PR21C	PR25C	—	L9T_D1	TRUE
R2	3	5	PR14D	PR19D	PR23D	—	L11C_A0	COMPLEMENT
R1	3	5	PR14C	PR19C	PR23C	VREF_CR_05	L11T_A0	TRUE
T2	3	5	PR14B	PR20D	PR24D	PRCK0C	L10C_A1	COMPLEMENT
T4	3	5	PR14A	PR20C	PR24C	PRCK0T	L10T_A1	TRUE
P1	3	5	PR13B	PR18D	PR22D	—	L12C_D1	COMPLEMENT
R3	3	5	PR13A	PR18C	PR22C	—	L12T_D1	TRUE
P3	3	6	PR13D	PR17D	PR21D	VREF_CR_06	L13C_A0	COMPLEMENT
P2	3	6	PR13C	PR17C	PR21C	—	L13T_A0	TRUE
P4	3	6	PR12B	PR16D	PR20D	—	L14C_D2	COMPLEMENT
N1	3	6	PR12A	PR16C	PR20C	—	L14T_D2	TRUE
M2	3	7	PR12D	PR15D	PR19D	—	L15C_D0	COMPLEMENT
N3	3	7	PR12C	PR15C	PR19C	—	L15T_D0	TRUE
L2	3	7	PR11D	PR14D	PR17D	—	L17C_A0	COMPLEMENT
L1	3	7	PR11C	PR14C	PR17C	VREF_CR_07	L17T_A0	TRUE
M3	3	7	PR11B	PR14B	PR18D	—	L16C_D0	COMPLEMENT
N4	3	7	PR11A	PR14A	PR18C	—	L16T_D0	TRUE
K2	3	8	PR9D	PR12D	PR14D	—	L19C_A0	COMPLEMENT
K1	3	8	PR9C	PR12C	PR14C	VREF_CR_08	L19T_A0	TRUE
L3	3	8	PR10D	PR13D	PR15D	—	L18C_D0	COMPLEMENT
M4	3	8	PR10C	PR13C	PR15C	—	L18T_D0	TRUE
B1	3	—	VSS	VSS	VSS	—	—	—
B29	3	—	VSS	VSS	VSS	—	—	—

Pin Information (continued)

Table 45. OR4E6 432-Pin EBGA (continued)

432 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
B3	3	—	VSS	VSS	VSS	—	—	—
B31	3	—	VSS	VSS	VSS	—	—	—
C1	3	—	VSS	VSS	VSS	—	—	—
N2	3	—	VDDIO_CR	VDDIO_CR	VDDIO_CR	—	—	—
U2	3	—	VDDIO_CR	VDDIO_CR	VDDIO_CR	—	—	—
Y3	3	—	VDDIO_CR	VDDIO_CR	VDDIO_CR	—	—	—
D15	3	—	VDD15	VDD15	VDD15	—	—	—
D17	3	—	VDD15	VDD15	VDD15	—	—	—
D21	3	—	VDD15	VDD15	VDD15	—	—	—
D25	3	—	VDD15	VDD15	VDD15	—	—	—
D28	3	—	VDD15	VDD15	VDD15	—	—	—
AJ8	4	1	PB23B	PB32D	PB39D	—	L3C_A0	COMPLEMENT
AK8	4	1	PB23A	PB32C	PB39C	—	L3T_A0	TRUE
AJ9	4	1	PB22D	PB31D	PB38D	—	L2C_D0	COMPLEMENT
AH10	4	1	PB22C	PB31C	PB38C	VREF_BR_01	L2T_D0	TRUE
AK9	4	1	PB22B	PB30D	PB37D	—	L1C_A0	COMPLEMENT
AL9	4	1	PB22A	PB30C	PB37C	—	L1T_A0	TRUE
AL6	4	2	PB24C	PB34C	PB42C	—	—	TRUE
AH8	4	2	PB24B	PB34B	PB41D	—	L5C_D0	COMPLEMENT
AJ7	4	2	PB24A	PB34A	PB41C	—	L5T_D0	TRUE
AK7	4	2	PB23D	PB33D	PB40D	VREF_BR_02	L4C_A0	COMPLEMENT
AL7	4	2	PB23C	PB33C	PB40C	—	L4T_A0	TRUE
AJ5	4	3	PB26D	PB36D	PB45D	—	L7C_A0	COMPLEMENT
AK5	4	3	PB26C	PB36C	PB45C	—	L7T_A0	TRUE
AL5	4	3	PB25D	PB35D	PB44D	VREF_BR_03	L6C_D1	COMPLEMENT
AJ6	4	3	PB25C	PB35C	PB44C	—	L6T_D1	TRUE
AK6	4	3	PB25A	PB35A	PB43A	—	—	TRUE
AJ4	4	4	PB27D	PB37D	PB47D	PLL_CK5C	L9C_A0	COMPLEMENT
AK4	4	4	PB27C	PB37C	PB47C	PLL_CK5T	L9T_A0	TRUE
AL4	4	4	PB27B	PB37B	PB46D	VREF_BR_04	L8C_D2	COMPLEMENT
AH6	4	4	PB27A	PB37A	PB46C	—	L8T_D2	TRUE
AH1	4	5	PR26B	PR38B	PR46D	PLL_CK4C	L10C_A0	COMPLEMENT
AH2	4	5	PR26A	PR38A	PR46C	PLL_CK4T	L10T_A0	TRUE
AG3	4	5	PR25B	PR37D	PR44D	—	L11C_D0	COMPLEMENT
AF4	4	5	PR25A	PR37C	PR44C	VREF_BR_05	L11T_D0	TRUE
AG1	4	6	PR25D	PR36D	PR43D	—	L12C_A0	COMPLEMENT
AG2	4	6	PR25C	PR36C	PR43C	—	L12T_A0	TRUE
AE3	4	6	PR24D	PR35D	PR41D	—	L14C_D0	COMPLEMENT
AD4	4	6	PR24C	PR35C	PR41C	VREF_BR_06	L14T_D0	TRUE
AF2	4	6	PR24B	PR35B	PR42D	—	L13C_A0	COMPLEMENT
AF3	4	6	PR24A	PR35A	PR42C	—	L13T_A0	TRUE
AD3	4	7	PR23D	PR33D	PR39D	VREF_BR_07	L16C_D0	COMPLEMENT
AC4	4	7	PR23C	PR33C	PR39C	—	L16T_D0	TRUE
AE1	4	7	PR23B	PR34D	PR40D	—	L15C_A0	COMPLEMENT
AE2	4	7	PR23A	PR34C	PR40C	—	L15T_A0	TRUE

Pin Information (continued)

Table 45. OR4E6 432-Pin EBGA (continued)

432 BGA Ball	$V_{DD}IO$ Bank	V_{REF} Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
AC3	4	7	PR22B	PR32D	PR38D	—	L17C_D0	COMPLEMENT
AD2	4	7	PR22A	PR32C	PR38C	—	L17T_D0	TRUE
AC2	4	8	PR22D	PR31D	PR37D	V_{REF}_BR_08	L18C_D1	COMPLEMENT
AB4	4	8	PR22C	PR31C	PR37C	—	L18T_D1	TRUE
AB3	4	8	PR21D	PR30D	PR36D	—	L19C_D1	COMPLEMENT
AC1	4	8	PR21C	PR30C	PR36C	—	L19T_D1	TRUE
AL20	4	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AL24	4	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AL29	4	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AL3	4	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AL30	4	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AL8	4	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AF1	4	—	V_{DD}IO_BR	V_{DD}IO_BR	V_{DD}IO_BR	—	—	—
AH3	4	—	V_{DD}IO_BR	V_{DD}IO_BR	V_{DD}IO_BR	—	—	—
AH9	4	—	V_{DD}IO_BR	V_{DD}IO_BR	V_{DD}IO_BR	—	—	—
AG4	4	—	V_{DD}33	V_{DD}33	V_{DD}33	—	—	—
AH5	4	—	V_{DD}33	V_{DD}33	V_{DD}33	—	—	—
B2	4	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
B30	4	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
C29	4	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
C3	4	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
D11	4	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
AK17	5	1	PB13D	PB18D	PB22D	—	L2C_A0	COMPLEMENT
AJ17	5	1	PB13C	PB18C	PB22C	V_{REF}_BC_01	L2T_A0	TRUE
AL18	5	1	PB13B	PB17D	PB21D	—	L1C_A0	COMPLEMENT
AK18	5	1	PB13A	PB17C	PB21C	—	L1T_A0	TRUE
AL15	5	2	PB15D	PB20D	PB24D	—	L4C_D0	COMPLEMENT
AK16	5	2	PB15C	PB20C	PB24C	V_{REF}_BC_02	L4T_D0	TRUE
AJ16	5	2	PB14D	PB19D	PB23D	PBCK0C	L3C_D1	COMPLEMENT
AL17	5	2	PB14C	PB19C	PB23C	PBCK0T	L3T_D1	TRUE
AL13	5	3	PB17D	PB23D	PB28D	PBCK1C	L7C_D1	COMPLEMENT
AJ14	5	3	PB17C	PB23C	PB28C	PBCK1T	L7T_D1	TRUE
AK14	5	3	PB17B	PB22D	PB27D	—	L6C_A0	COMPLEMENT
AL14	5	3	PB17A	PB22C	PB27C	—	L6T_A0	TRUE
AJ15	5	3	PB16D	PB21D	PB26D	V_{REF}_BC_03	L5C_A0	COMPLEMENT
AK15	5	3	PB16C	PB21C	PB26C	—	L5T_A0	TRUE
AL11	5	4	PB19B	PB26B	PB31D	—	L10C_D1	COMPLEMENT
AJ12	5	4	PB19A	PB26A	PB31C	—	L10T_D1	TRUE
AH13	5	4	PB18D	PB25D	PB30D	V_{REF}_BC_04	L9C_D1	COMPLEMENT
AK12	5	4	PB18C	PB25C	PB30C	—	L9T_D1	TRUE
AK13	5	4	PB18B	PB24D	PB29D	—	L8C_A0	COMPLEMENT
AH14	5	4	PB18A	PB24C	PB29C	—	L8T_A0	TRUE
AL10	5	5	PB20D	PB27D	PB34D	—	L12C_D1	COMPLEMENT
AJ11	5	5	PB20C	PB27C	PB34C	—	L12T_D1	TRUE
AH12	5	5	PB19D	PB26D	PB32D	V_{REF}_BC_05	L11C_D1	COMPLEMENT

Pin Information (continued)

Table 45. OR4E6 432-Pin EBGA (continued)

432 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
AK11	5	5	PB19C	PB26C	PB32C	—	L11T_D1	TRUE
AJ10	5	6	PB21B	PB28D	PB35D	VREF_BC_06	L13C_A0	COMPLEMENT
AK10	5	6	PB21A	PB28C	PB35C	—	L13T_A0	TRUE
AK3	5	—	VSS	VSS	VSS	—	—	—
AK31	5	—	VSS	VSS	VSS	—	—	—
AL12	5	—	VSS	VSS	VSS	—	—	—
AL16	5	—	VSS	VSS	VSS	—	—	—
AL2	5	—	VSS	VSS	VSS	—	—	—
AJ13	5	—	VDDIO_BC	VDDIO_BC	VDDIO_BC	—	—	—
AH16	5	—	VDDIO_BC	VDDIO_BC	VDDIO_BC	—	—	—
AJ3	5	—	VDD15	VDD15	VDD15	—	—	—
AK2	5	—	VDD15	VDD15	VDD15	—	—	—
AK30	5	—	VDD15	VDD15	VDD15	—	—	—
AL1	5	—	VDD15	VDD15	VDD15	—	—	—
AL31	5	—	VDD15	VDD15	VDD15	—	—	—
AC29	6	1	PL22D	PL32D	PL38D	D8	L1C_D0	COMPLEMENT
AD30	6	1	PL22C	PL32C	PL38C	VREF_BL_01	L1T_D0	TRUE
AD29	6	1	PL22B	PL33D	PL39D	D9	L2C_D0	COMPLEMENT
AC28	6	1	PL22A	PL33C	PL39C	D10	L2T_D0	TRUE
AE31	6	2	PL23D	PL34D	PL40D	—	L3C_A0	COMPLEMENT
AE30	6	2	PL23C	PL34C	PL40C	VREF_BL_02	L3T_A0	TRUE
AF30	6	3	PL25D	PL36B	PL44D	VREF_BL_03	L5C_A0	COMPLEMENT
AF29	6	3	PL25C	PL36A	PL44C	D13	L5T_A0	TRUE
AD28	6	3	PL24D	PL35B	PL42D	D11	L4C_D2	COMPLEMENT
AF31	6	3	PL24C	PL35A	PL42C	D12	L4T_D2	TRUE
AG29	6	4	PL27D	PL39D	PL47D	PLL_CK7C	L7C_D1	COMPLEMENT
AF28	6	4	PL27C	PL39C	PL47C	PLL_CK7T	L7T_D1	TRUE
AG31	6	4	PL26D	PL37B	PL45D	—	L6C_A0	COMPLEMENT
AG30	6	4	PL26C	PL37A	PL45C	VREF_BL_04	L6T_A0	TRUE
AJ27	6	5	PB3D	PB4B	PB4D	DP3	L9C_D0	COMPLEMENT
AH26	6	5	PB3C	PB4A	PB4C	VREF_BL_05	L9T_D0	TRUE
AL28	6	5	PB2D	PB2D	PB2D	PLL_CK6C	L8C_A0	COMPLEMENT
AK28	6	5	PB2C	PB2C	PB2C	PLL_CK6T	L8T_A0	TRUE
AJ28	6	5	PB2A	PB2A	PB2A	DP2	—	TRUE
AH24	6	6	PB5B	PB6B	PB7D	—	L12C_D2	COMPLEMENT
AL26	6	6	PB5A	PB6A	PB7C	—	L12T_D2	TRUE
AK26	6	6	PB4D	PB5D	PB6D	D14	L11C_A0	COMPLEMENT
AJ26	6	6	PB4C	PB5C	PB6C	VREF_BL_06	L11T_A0	TRUE
AL27	6	6	PB4B	PB4D	PB5D	—	L10C_A0	COMPLEMENT
AK27	6	6	PB4A	PB4C	PB5C	—	L10T_A0	TRUE
AJ23	6	7	PB6D	PB8D	PB10D	D19	L15C_D0	COMPLEMENT
AK24	6	7	PB6C	PB8C	PB10C	VREF_BL_07	L15T_D0	TRUE
AJ24	6	7	PB6B	PB7D	PB9D	D18	L14C_D0	COMPLEMENT
AH23	6	7	PB6A	PB7C	PB9C	D17	L14T_D0	TRUE
AL25	6	7	PB5D	PB6D	PB8D	D16	L13C_A0	COMPLEMENT

Pin Information (continued)

Table 45.OR4E6 432-Pin EBGA (continued)

432 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
AK25	6	7	PB5C	PB6C	PB8C	D15	L13T_A0	TRUE
AJ22	6	8	PB7D	PB10D	PB12D	D22	L17C_D1	COMPLEMENT
AL23	6	8	PB7C	PB10C	PB12C	VREF_BL_08	L17T_D1	TRUE
AK23	6	8	PB7B	PB9D	PB11D	D21	L16C_D1	COMPLEMENT
AH22	6	8	PB7A	PB9C	PB11C	D20	L16T_D1	TRUE
AH20	6	9	PB9D	PB12D	PB14D	D25	L19C_D0	COMPLEMENT
AJ21	6	9	PB9C	PB12C	PB14C	VREF_BL_09	L19T_D0	TRUE
AL22	6	9	PB8D	PB11D	PB13D	D24	L18C_A0	COMPLEMENT
AK22	6	9	PB8C	PB11C	PB13C	D23	L18T_A0	TRUE
AJ19	6	10	PB11D	PB14D	PB18D	D28	L21C_D0	COMPLEMENT
AK20	6	10	PB11C	PB14C	PB18C	VREF_BL_10	L21T_D0	TRUE
AJ20	6	10	PB11B	PB14B	PB17D	—		COMPLEMENT
AL21	6	10	PB10D	PB13D	PB16D	D27	L20C_A0	COMPLEMENT
AK21	6	10	PB10C	PB13C	PB16C	D26	L20T_A0	TRUE
AJ18	6	11	PB12D	PB16D	PB20D	D31	L23C_D1	COMPLEMENT
AL19	6	11	PB12C	PB16C	PB20C	VREF_BL_11	L23T_D1	TRUE
AH18	6	11	PB12B	PB15D	PB19D	D30	L22C_D1	COMPLEMENT
AK19	6	11	PB12A	PB15C	PB19C	D29	L22T_D1	TRUE
AJ1	6	—	VSS	VSS	VSS	—	—	—
AJ2	6	—	VSS	VSS	VSS	—	—	—
AJ30	6	—	VSS	VSS	VSS	—	—	—
AJ31	6	—	VSS	VSS	VSS	—	—	—
AK1	6	—	VSS	VSS	VSS	—	—	—
AK29	6	—	VSS	VSS	VSS	—	—	—
AH19	6	—	VDDIO_BL	VDDIO_BL	VDDIO_BL	—	—	—
AJ25	6	—	VDDIO_BL	VDDIO_BL	VDDIO_BL	—	—	—
AH30	6	—	VDDIO_BL	VDDIO_BL	VDDIO_BL	—	—	—
AE29	6	—	VDDIO_BL	VDDIO_BL	VDDIO_BL	—	—	—
AH27	6	—	VDD33	VDD33	VDD33	—	—	—
AG28	6	—	VDD33	VDD33	VDD33	—	—	—
AH25	6	—	VDD15	VDD15	VDD15	—	—	—
AH28	6	—	VDD15	VDD15	VDD15	—	—	—
AH4	6	—	VDD15	VDD15	VDD15	—	—	—
AH7	6	—	VDD15	VDD15	VDD15	—	—	—
AJ29	6	—	VDD15	VDD15	VDD15	—	—	—
AH31	6	—	PTEMP	PTEMP	PTEMP	—	—	—
AH29	6	—	LVDS_R	LVDS_R	LVDS_R	—	—	—
M29	7	1	PL9D	PL13D	PL16D	VREF_7_01	L2C_D0	COMPLEMENT
N28	7	1	PL9C	PL13C	PL16C	D4	L2T_D0	TRUE
L30	7	1	PL8D	PL12D	PL14D	A15	L1C_A0	COMPLEMENT
L31	7	1	PL8C	PL12C	PL14C	A14	L1T_A0	TRUE
M30	7	2	PL9B	PL13B	PL17D	—	—	COMPLEMENT
N29	7	2	PL10D	PL14D	PL18D	RDY/$\overline{\text{BUSY}}$/RCLK	L3C_A0	COMPLEMENT
N30	7	2	PL10C	PL14C	PL18C	VREF_7_02	L3T_A0	TRUE

Pin Information (continued)

Table 45. OR4E6 432-Pin EBGA (continued)

432 BGA Ball	$V_{DD}IO$ Bank	V_{REF} Group	General-Purpose User I/O			Additional Function	Pair	Differential
			OR4E2	OR4E4	OR4E6			
N31	7	2	PL10B	PL15D	PL19D	A13	L4C_D1	COMPLEMENT
P29	7	2	PL10A	PL15C	PL19C	A12	L4T_D1	TRUE
P30	7	3	PL11D	PL16D	PL20D	—	L5C_A0	COMPLEMENT
P31	7	3	PL11C	PL16C	PL20C	—	L5T_A0	TRUE
R29	7	3	PL11B	PL17D	PL21D	A11	L6C_A0	COMPLEMENT
R30	7	3	PL11A	PL17C	PL21C	V_{REF}_7_03	L6T_A0	TRUE
T28	7	4	PL14D	PL20D	PL24D	PLCK0C	L8C_A1	COMPLEMENT
T30	7	4	PL14C	PL20C	PL24C	PLCK0T	L8T_A1	TRUE
R31	7	4	PL13D	PL19D	PL23D	$\overline{RD}$/$\overline{MPI\_STRB}$	L7C_D1	COMPLEMENT
T29	7	4	PL13C	PL19C	PL23C	V_{REF}_7_04	L7T_D1	TRUE
V31	7	5	PL16D	PL22D	PL26D	A8	L10C_A0	COMPLEMENT
V30	7	5	PL16C	PL22C	PL26C	V_{REF}_7_05	L10T_A0	TRUE
U30	7	5	PL15D	PL21D	PL25D	A10	L9C_A0	COMPLEMENT
U29	7	5	PL15C	PL21C	PL25C	A9	L9T_A0	TRUE
W29	7	6	PL18D	PL26D	PL30D	A6	L13C_D0	COMPLEMENT
Y30	7	6	PL18C	PL26C	PL30C	A5	L13T_D0	TRUE
V29	7	6	PL17D	PL24D	PL28D	PLCK1C	L11C_D1	COMPLEMENT
W31	7	6	PL17C	PL24C	PL28C	PLCK1T	L11T_D1	TRUE
V28	7	6	PL17B	PL25D	PL29D	V_{REF}_7_06	L12C_D1	COMPLEMENT
W30	7	6	PL17A	PL25C	PL29C	A7	L12T_D1	TRUE
AA31	7	7	PL19D	PL27D	PL32D	$\overline{WR}$/MPI_RW	L14C_A0	COMPLEMENT
AA30	7	7	PL19C	PL27C	PL32C	V_{REF}_7_07	L14T_A0	TRUE
Y29	7	7	PL18B	PL26B	PL31D	—	—	COMPLEMENT
AB29	7	8	PL21D	PL30D	PL36D	A1	L17C_D1	COMPLEMENT
AC31	7	8	PL21C	PL30C	PL36C	A0	L17T_D1	TRUE
AC30	7	8	PL21B	PL31D	PL37D	DP0	L18C_D1	COMPLEMENT
AB28	7	8	PL21A	PL31C	PL37C	DP1	L18T_D1	TRUE
Y28	7	8	PL20D	PL28D	PL34D	A4	L15C_D0	COMPLEMENT
AA29	7	8	PL20C	PL28C	PL34C	V_{REF}_7_08	L15T_D0	TRUE
AB31	7	8	PL20B	PL29D	PL35D	A3	L16C_A0	COMPLEMENT
AB30	7	8	PL20A	PL29C	PL35C	A2	L16T_A0	TRUE
A3	7	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
A30	7	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
A8	7	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AD1	7	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
AD31	7	—	V_{SS}	V_{SS}	V_{SS}	—	—	—
W28	7	—	V_{DD}IO_CL	V_{DD}IO_CL	V_{DD}IO_CL	—	—	—
U31	7	—	V_{DD}IO_CL	V_{DD}IO_CL	V_{DD}IO_CL	—	—	—
P28	7	—	V_{DD}IO_CL	V_{DD}IO_CL	V_{DD}IO_CL	—	—	—
AE4	7	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
AH11	7	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
AH15	7	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
AH17	7	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—
AH21	7	—	V_{DD}15	V_{DD}15	V_{DD}15	—	—	—

Pin Information (continued)

As shown in the Pair column, differential pairs and physical locations are numbered within each bank (e.g., L19C_A0 is the nineteenth pair in an associated bank). The C indicates complementary differential whereas a T indicates true differential. The _A0 indicates the physical location of adjacent balls in either the horizontal or vertical direction. Other physical indicators are as follows:

- _A1 indicates one ball between pairs.
- _A2 indicates two balls between pairs.
- _D0 indicates balls are diagonally adjacent.
- _D1 indicates diagonally adjacent separated by one physical ball.

VREF pins, shown in the Additional Function column, are associated to the bank and group (e.g., VREF_TL_01 is the VREF for group one of the top left (TL) bank).

Table 46. 680-Pin PBGAM Pinout

680 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
E16	TL	1	PT13D	PT18D	$\overline{\text{MPI\_RTRY}}$	L1C_D1	COMPLEMENT
D14	TL	1	PT13C	PT18C	$\overline{\text{MPI\_ACK}}$	L1T_D1	TRUE
C14	TL	1	PT13B	PT17D	—	L2C_D0	COMPLEMENT
D13	TL	1	PT13A	PT17C	VREF_TL_01	L2T_D0	TRUE
A12	TL	1	PT12D	PT16D	M0	L3C_A0	COMPLEMENT
B12	TL	1	PT12C	PT16C	M1	L3T_A0	TRUE
E15	TL	2	PT12B	PT15D	MPI_CLK	L4C_D3	COMPLEMENT
B11	TL	2	PT12A	PT15C	A21/ $\overline{\text{MPI\_BURST}}$	L4T_D3	TRUE
C11	TL	2	PT11D	PT14D	M2	L5C_D2	COMPLEMENT
E14	TL	2	PT11C	PT14C	M3	L5T_D2	TRUE
D12	TL	2	PT11B	PT13D	VREF_TL_02	L6C_A0	COMPLEMENT
D11	TL	2	PT11A	PT13C	$\overline{\text{MPI\_TEA}}$	L6T_A0	TRUE
A10	TL	3	PT10D	PT12D	—	L7C_A0	COMPLEMENT
B10	TL	3	PT10C	PT12C	—	L7T_A0	TRUE
C10	TL	3	PT10A	PT12A	—	—	TRUE
C9	TL	3	PT9D	PT11D	VREF_TL_03	L8C_D0	COMPLEMENT
D10	TL	3	PT9C	PT11C	—	L8T_D0	TRUE
E13	TL	3	PT9A	PT11A	—	—	TRUE
B9	TL	3	PT8D	PT10D	D0	L9C_A0	COMPLEMENT
A9	TL	3	PT8C	PT10C	TMS	L9T_A0	TRUE
D9	TL	4	PT7D	PT9D	A20/$\overline{\text{MPI\_BDIP}}$	L10C_D2	COMPLEMENT
A8	TL	4	PT7C	PT9C	A19/MPI_TSZ1	L10T_D2	TRUE
B8	TL	4	PT6D	PT8D	A18/MPI_TSZ0	L11C_D3	COMPLEMENT
E12	TL	4	PT6C	PT8C	D3	L11T_D3	TRUE
C8	TL	4	PT6B	PT7D	VREF_TL_04	L12C_A0	COMPLEMENT
D8	TL	4	PT6A	PT7C	—	L12T_A0	TRUE
E11	TL	5	PT5D	PT6D	D1	L13C_D3	COMPLEMENT

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
A7	TL	5	PT5C	PT6C	D2	L13T_D3	TRUE
A6	TL	5	PT5B	PT5D	—	L14C_D0	COMPLEMENT
B7	TL	5	PT5A	PT5C	VREF_TL_05	L14T_D0	TRUE
C7	TL	5	PT4D	PT4D	TDI	L15C_A0	COMPLEMENT
D7	TL	5	PT4C	PT4C	TCK	L15T_A0	TRUE
E10	TL	5	PT4B	PT4B	—	L16C_D4	COMPLEMENT
A5	TL	5	PT4A	PT4A	—	L16T_D4	TRUE
B6	TL	6	PT3D	PT3D	—	L17C_D2	COMPLEMENT
E9	TL	6	PT3C	PT3C	VREF_TL_06	L17T_D2	TRUE
A4	TL	6	PT3B	PT3B	—	L18C_D0	COMPLEMENT
B5	TL	6	PT3A	PT3A	—	L18T_D0	TRUE
D6	TL	6	PT2D	PT2D	PLL_CK1C/ PPLL	L19C_A0	COMPLEMENT
C6	TL	6	PT2C	PT2C	PLL_CK1T/ PPLL	L19T_A0	TRUE
C5	TL	6	PT2B	PT2B	—	L20C_D1	COMPLEMENT
E8	TL	6	PT2A	PT2A	—	L20T_D1	TRUE
D1	TL	7	PL2D	PL2D	PLL_CK0C/ HPPLL	L21C_D2	COMPLEMENT
F4	TL	7	PL2C	PL2C	PLL_CK0T/ HPPLL	L21T_D2	TRUE
F3	TL	7	PL3D	PL3D	—	L22C_D0	COMPLEMENT
G4	TL	7	PL3C	PL3C	VREF_TL_07	L22T_D0	TRUE
E2	TL	7	PL4D	PL4D	D5	L23C_D2	COMPLEMENT
H5	TL	7	PL4C	PL4C	D6	L23T_D2	TRUE
E1	TL	8	PL4B	PL5D	—	L24C_D0	COMPLEMENT
F2	TL	8	PL4A	PL5C	VREF_TL_08	L24T_D0	TRUE
J5	TL	8	PL5D	PL6D	HDC	L25C_D3	COMPLEMENT
F1	TL	8	PL5C	PL6C	$\overline{\text{LDC}}$	L25T_D3	TRUE
H4	TL	8	PL5B	PL7D	—	L26C_D0	COMPLEMENT
G3	TL	8	PL5A	PL7C	—	L26T_D0	TRUE
H3	TL	9	PL6D	PL8D	—	L27C_D0	COMPLEMENT
G2	TL	9	PL6C	PL8C	D7	L27T_D0	TRUE
K5	TL	9	PL7D	PL9D	VREF_TL_09	L28C_D3	COMPLEMENT
G1	TL	9	PL7C	PL9C	A17	L28T_D3	TRUE
J4	TL	9	PL8D	PL10D	$\overline{\text{CS0}}$	L29C_D1	COMPLEMENT
L5	TL	9	PL8C	PL10C	CS1	L29T_D1	TRUE
J3	TL	10	PL9D	PL11D	—	L30C_D0	COMPLEMENT

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	$V_{DD}IO$ Bank	V_{REF} Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
H2	TL	10	PL9C	PL11C	—	L30T_D0	TRUE
K4	TL	10	PL9A	PL11A	—	—	TRUE
H1	TL	10	PL10D	PL12D	$\overline{INIT}$	L31C_D0	COMPLEMENT
J2	TL	10	PL10C	PL12C	DOUT	L31T_D0	TRUE
J1	TL	10	PL11D	PL13D	V_{REF}_TL_10	L32C_D1	COMPLEMENT
K3	TL	10	PL11C	PL13C	A16	L32T_D1	TRUE
M5	TL	10	PL11A	PL13A	—	—	TRUE
F5	TL	—	V_{DD}33	V_{DD}33	—	—	—
E4	TL	—	PRD_DATA	PRD_DATA	—	—	—
E3	TL	—	$\overline{PRESET}$	$\overline{PRESET}$	—	—	—
D2	TL	—	$\overline{PRD\_CFG}$	$\overline{PRD\_CFG}$	—	—	—
G5	TL	—	$\overline{PPRGRM}$	$\overline{PPRGRM}$	—	—	—
E7	TL	—	PCFG_MPI_IRQ	PCFG_MPI_IRQ	$\overline{CFG\_IRQ}$/ $\overline{MPI\_IRQ}$	—	—
E6	TL	—	PCCLK	PCCLK	—	—	—
B4	TL	—	PDONE	PDONE	—	—	—
D5	TL	—	V_{DD}33	V_{DD}33	—	—	—
A1	TL	—	V_{SS}	V_{SS}	—	—	—
A2	TL	—	V_{SS}	V_{SS}	—	—	—
A18	TL	—	V_{SS}	V_{SS}	—	—	—
A33	TL	—	V_{SS}	V_{SS}	—	—	—
A34	TL	—	V_{SS}	V_{SS}	—	—	—
B1	TL	—	V_{SS}	V_{SS}	—	—	—
B2	TL	—	V_{SS}	V_{SS}	—	—	—
B33	TL	—	V_{SS}	V_{SS}	—	—	—
B34	TL	—	V_{SS}	V_{SS}	—	—	—
C3	TL	—	V_{SS}	V_{SS}	—	—	—
C13	TL	—	V_{SS}	V_{SS}	—	—	—
N16	TL	—	V_{DD}15	V_{DD}15	—	—	—
N17	TL	—	V_{DD}15	V_{DD}15	—	—	—
N18	TL	—	V_{DD}15	V_{DD}15	—	—	—
N19	TL	—	V_{DD}15	V_{DD}15	—	—	—
P16	TL	—	V_{DD}15	V_{DD}15	—	—	—
P17	TL	—	V_{DD}15	V_{DD}15	—	—	—
A3	TL	—	V_{DD}IO_TL	V_{DD}IO_TL	—	—	—
B3	TL	—	V_{DD}IO_TL	V_{DD}IO_TL	—	—	—
C1	TL	—	V_{DD}IO_TL	V_{DD}IO_TL	—	—	—
C2	TL	—	V_{DD}IO_TL	V_{DD}IO_TL	—	—	—

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
C4	TL	—	VDDIO_TL	VDDIO_TL	—	—	—
D3	TL	—	VDDIO_TL	VDDIO_TL	—	—	—
E5	TL	—	VDDIO_TL	VDDIO_TL	—	—	—
C25	TC	1	PT28D	PT35D	—	L1C_D1	COMPLEMENT
E24	TC	1	PT28C	PT35C	—	L1T_D1	TRUE
D24	TC	1	PT28A	PT35A	—	L2T_D2	TRUE
A25	TC	1	PT28B	PT35B	—	L2C_D2	COMPLEMENT
D23	TC	1	PT27D	PT34D	VREF_TC_01	L3C_D1	COMPLEMENT
B25	TC	1	PT27C	PT34C	—	L3T_D1	TRUE
C24	TC	1	PT27B	PT33D	—	L4C_D1	COMPLEMENT
E23	TC	1	PT27A	PT33C	—	L4T_D1	TRUE
B24	TC	2	PT26D	PT32D	—	L5C_D1	COMPLEMENT
D22	TC	2	PT26C	PT32C	VREF_TC_02	L5T_D1	TRUE
E22	TC	2	PT26B	PT31D	—	L6C_D0	COMPLEMENT
D21	TC	2	PT26A	PT31C	—	L6T_D0	TRUE
B23	TC	2	PT25D	PT30D	—	L7C_A0	COMPLEMENT
B22	TC	2	PT25C	PT30C	—	L7T_A0	TRUE
A23	TC	3	PT24D	PT29D	—	L8C_D1	COMPLEMENT
C21	TC	3	PT24C	PT29C	VREF_TC_03	L8T_D1	TRUE
E21	TC	3	PT24A	PT29A	—	—	TRUE
D20	TC	3	PT23D	PT28D	—	L9C_D2	COMPLEMENT
A22	TC	3	PT23C	PT28C	—	L9T_D2	TRUE
E20	TC	3	PT23A	PT28A	—	—	TRUE
A21	TC	3	PT22D	PT27D	—	L10C_A0	COMPLEMENT
B21	TC	3	PT22C	PT27C	—	L10T_A0	TRUE
D19	TC	3	PT22A	PT27A	—	—	TRUE
B20	TC	4	PT21D	PT26D	—	L11C_A0	COMPLEMENT
A20	TC	4	PT21C	PT26C	—	L11T_A0	TRUE
B19	TC	4	PT20D	PT25D	—	L12C_A0	COMPLEMENT
C19	TC	4	PT20C	PT25C	—	L12T_A0	TRUE
E19	TC	4	PT19D	PT24D	—	L13C_D0	COMPLEMENT
D18	TC	4	PT19C	PT24C	VREF_TC_04	L13T_D0	TRUE
B18	TC	4	PT19A	PT24A	—	L14T_A0	TRUE
C18	TC	4	PT19B	PT24B	—	L14C_A0	COMPLEMENT
B17	TC	5	PT18D	PT23D	PTCK1C	L15C_D0	COMPLEMENT
C17	TC	5	PT18C	PT23C	PTCK1T	L15T_D0	TRUE
D17	TC	5	PT18A	PT23A		L16T_D2	TRUE
A16	TC	5	PT18B	PT23B		L16C_D2	COMPLEMENT

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
B16	TC	5	PT17D	PT22D	PTCK0C	L17C_A0	COMPLEMENT
C16	TC	5	PT17C	PT22C	PTCK0T	L17T_A0	TRUE
D16	TC	5	PT17A	PT22A	—	—	TRUE
E18	TC	5	PT16D	PT21D	VREF_TC_05	L18C_D3	COMPLEMENT
A15	TC	5	PT16C	PT21C	—	L18T_D3	TRUE
B15	TC	5	PT16A	PT21A	—	—	TRUE
D15	TC	6	PT15D	PT20D	—	L19C_D2	COMPLEMENT
A14	TC	6	PT15C	PT20C	—	L19T_D2	TRUE
B14	TC	6	PT15A	PT20A	—	—	TRUE
E17	TC	6	PT14D	PT19D	—	L20C_D3	COMPLEMENT
A13	TC	6	PT14C	PT19C	VREF_TC_06	L20T_D3	TRUE
B13	TC	6	PT14A	PT19A	—	—	TRUE
C22	TC	—	VSS	VSS	—	—	—
C32	TC	—	VSS	VSS	—	—	—
D4	TC	—	VSS	VSS	—	—	—
D31	TC	—	VSS	VSS	—	—	—
N3	TC	—	VSS	VSS	—	—	—
N13	TC	—	VSS	VSS	—	—	—
N14	TC	—	VSS	VSS	—	—	—
N15	TC	—	VSS	VSS	—	—	—
N20	TC	—	VSS	VSS	—	—	—
N21	TC	—	VSS	VSS	—	—	—
N22	TC	—	VSS	VSS	—	—	—
P18	TC	—	VDD15	VDD15	—	—	—
P19	TC	—	VDD15	VDD15	—	—	—
R16	TC	—	VDD15	VDD15	—	—	—
R17	TC	—	VDD15	VDD15	—	—	—
R18	TC	—	VDD15	VDD15	—	—	—
R19	TC	—	VDD15	VDD15	—	—	—
A11	TC	—	VDDIO_TC	VDDIO_TC	—	—	—
A17	TC	—	VDDIO_TC	VDDIO_TC	—	—	—
A19	TC	—	VDDIO_TC	VDDIO_TC	—	—	—
A24	TC	—	VDDIO_TC	VDDIO_TC	—	—	—
C12	TC	—	VDDIO_TC	VDDIO_TC	—	—	—
C15	TC	—	VDDIO_TC	VDDIO_TC	—	—	—
C20	TC	—	VDDIO_TC	VDDIO_TC	—	—	—
C23	TC	—	VDDIO_TC	VDDIO_TC	—	—	—
J34	TR	1	PR11B	PR13B	—	L1C_A0	COMPLEMENT

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
H34	TR	1	PR11A	PR13A	—	L1T_A0	TRUE
J33	TR	1	PR11C	PR13C	—	L2T_A1	TRUE
J31	TR	1	PR11D	PR13D	VREF_TR_01	L2C_A1	COMPLEMENT
J32	TR	1	PR10C	PR12C	—	L3T_D1	TRUE
G34	TR	1	PR10D	PR12D	—	L3C_D1	COMPLEMENT
M30	TR	1	PR9A	PR11A	—	—	TRUE
H33	TR	1	PR9C	PR11C	—	L4T_A0	TRUE
H32	TR	1	PR9D	PR11D	—	L4C_A0	COMPLEMENT
L30	TR	1	PR8A	PR10A	—	—	TRUE
H31	TR	2	PR8C	PR10C	—	L5T_D1	TRUE
G33	TR	2	PR8D	PR10D	—	L5C_D1	COMPLEMENT
F34	TR	2	PR7A	PR9A	—	—	TRUE
F33	TR	2	PR7C	PR9C	VREF_TR_02	L6T_D0	TRUE
G32	TR	2	PR7D	PR9D	—	L6C_D0	COMPLEMENT
K30	TR	2	PR6A	PR8C	—	L7T_D2	TRUE
G31	TR	2	PR6B	PR8D	—	L7C_D2	COMPLEMENT
E34	TR	3	PR6C	PR7C	—	L8T_D2	TRUE
J30	TR	3	PR6D	PR7D	VREF_TR_03	L8C_D2	COMPLEMENT
D34	TR	3	PR5A	PR6A	—	—	TRUE
F32	TR	3	PR5C	PR6C	—	L9T_A0	TRUE
F31	TR	3	PR5D	PR6D	—	L9C_A0	COMPLEMENT
E33	TR	3	PR4C	PR5C	—	L10T_A0	TRUE
D33	TR	3	PR4D	PR5D	—	L10C_A0	COMPLEMENT
H30	TR	4	PR3A	PR4C	—	L11T_D2	TRUE
E32	TR	4	PR3B	PR4D	VREF_TR_04	L11C_D2	COMPLEMENT
E31	TR	4	PR3C	PR3C	PLL_CK3T/ PLL1(1.554/ 2.048 MHz)	L12T_A0	TRUE
G30	TR	4	PR3D	PR3D	PLL_CK3C/ PLL1(1.554/ 2.048 MHz)	L12C_A0	COMPLEMENT
C30	TR	5	PT37D	PT47D	PLL_CK2C/ PPLL	L13C_D0	COMPLEMENT
B31	TR	5	PT37C	PT47C	PLL_CK2T/ PPLL	L13T_D0	TRUE
E28	TR	5	PT37B	PT46D	—	L14C_D2	COMPLEMENT
B30	TR	5	PT37A	PT46C	—	L14T_D2	TRUE
D29	TR	5	PT36D	PT45D	VREF_TR_05	L15C_D2	COMPLEMENT
A31	TR	5	PT36C	PT45C	—	L15T_D2	TRUE

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	V_{DDIO} Bank	V_{REF} Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
D28	TR	6	PT35D	PT44D	—	L16C_D1	COMPLEMENT
B29	TR	6	PT35C	PT44C	—	L16T_D1	TRUE
E27	TR	6	PT35B	PT43D	—	L17C_D1	COMPLEMENT
C29	TR	6	PT35A	PT43C	—	L17T_D1	TRUE
A30	TR	6	PT34D	PT42D	V_{REF}_TR_06	L18C_D3	COMPLEMENT
E26	TR	6	PT34C	PT42C	—	L18T_D3	TRUE
A29	TR	7	PT34B	PT41D	—	L19C_D2	COMPLEMENT
D27	TR	7	PT34A	PT41C	—	L19T_D2	TRUE
C28	TR	7	PT33D	PT40D	—	L20C_A0	COMPLEMENT
C27	TR	7	PT33C	PT40C	V_{REF}_TR_07	L20T_A0	TRUE
B28	TR	7	PT32D	PT39D	—	L21C_D2	COMPLEMENT
E25	TR	7	PT32C	PT39C	—	L21T_D2	TRUE
A28	TR	8	PT31D	PT38D	—	L22C_D2	COMPLEMENT
D26	TR	8	PT31C	PT38C	V_{REF}_TR_08	L22T_D2	TRUE
C26	TR	8	PT30D	PT37D	—	—	COMPLEMENT
B27	TR	8	PT30A	PT37A	—	—	TRUE
D25	TR	8	PT29D	PT36D	—	L23C_D2	COMPLEMENT
A27	TR	8	PT29C	PT36C	—	L23T_D2	TRUE
A26	TR	8	PT29A	PT36A	—	L24T_A0	TRUE
B26	TR	8	PT29B	PT36B	—	L24C_A0	COMPLEMENT
F30	TR	—	V_{DD}33	V_{DD}33	—	—	—
E29	TR	—	V_{DD}33	V_{DD}33	—	—	—
D30	TR	—	PLL_VF	PLL_VF	—	—	—
N32	TR	—	V_{SS}	V_{SS}	—	—	—
P13	TR	—	V_{SS}	V_{SS}	—	—	—
P14	TR	—	V_{SS}	V_{SS}	—	—	—
P15	TR	—	V_{SS}	V_{SS}	—	—	—
P20	TR	—	V_{SS}	V_{SS}	—	—	—
P21	TR	—	V_{SS}	V_{SS}	—	—	—
P22	TR	—	V_{SS}	V_{SS}	—	—	—
R13	TR	—	V_{SS}	V_{SS}	—	—	—
R14	TR	—	V_{SS}	V_{SS}	—	—	—
R15	TR	—	V_{SS}	V_{SS}	—	—	—
R20	TR	—	V_{SS}	V_{SS}	—	—	—
T13	TR	—	V_{DD}15	V_{DD}15	—	—	—
T14	TR	—	V_{DD}15	V_{DD}15	—	—	—
T15	TR	—	V_{DD}15	V_{DD}15	—	—	—

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
T20	TR	—	VDD15	VDD15	—	—	—
T21	TR	—	VDD15	VDD15	—	—	—
T22	TR	—	VDD15	VDD15	—	—	—
A32	TR	—	VDDIO_TR	VDDIO_TR	—	—	—
B32	TR	—	VDDIO_TR	VDDIO_TR	—	—	—
C31	TR	—	VDDIO_TR	VDDIO_TR	—	—	—
C33	TR	—	VDDIO_TR	VDDIO_TR	—	—	—
C34	TR	—	VDDIO_TR	VDDIO_TR	—	—	—
D32	TR	—	VDDIO_TR	VDDIO_TR	—	—	—
E30	TR	—	VDDIO_TR	VDDIO_TR	—	—	—
AE32	CR	1	PR29A	PR35C	—	L1T_D1	TRUE
AC30	CR	1	PR29B	PR35D	—	L1C_D1	COMPLEMENT
AD31	CR	1	PR28A	PR34A	—	—	TRUE
AE33	CR	1	PR29C	PR34C	—	L2T_D1	TRUE
AC31	CR	1	PR29D	PR34D	—	L2C_D1	COMPLEMENT
AE34	CR	1	PR28B	PR33B	—	—	COMPLEMENT
AD32	CR	1	PR28C	PR33C	VREF_CR_01	L3T_D1	TRUE
AB30	CR	1	PR28D	PR33D	—	L3C_D1	COMPLEMENT
AD33	CR	2	PR27D	PR32B	—	—	COMPLEMENT
AB31	CR	2	PR27A	PR32C	—	L4T_D0	TRUE
AA30	CR	2	PR27B	PR32D	—	L4C_D0	COMPLEMENT
AA31	CR	2	PR27C	PR31A	—	—	TRUE
AC33	CR	2	PR26A	PR31C	—	L5T_A0	TRUE
AB33	CR	2	PR26B	PR31D	VREF_CR_02	L5C_A0	COMPLEMENT
AC34	CR	2	PR26C	PR30A	—	—	TRUE
AA32	CR	2	PR25A	PR30C	—	L6T_D1	TRUE
Y30	CR	2	PR25B	PR30D	—	L6C_D1	COMPLEMENT
Y31	CR	3	PR24B	PR29B	—	—	COMPLEMENT
AB34	CR	3	PR25C	PR29C	—	L7T_D3	TRUE
W30	CR	3	PR25D	PR29D	VREF_CR_03	L7C_D3	COMPLEMENT
AA34	CR	3	PR24A	PR28A	—	—	TRUE
AA33	CR	3	PR24C	PR28C	—	L8T_D1	TRUE
W31	CR	3	PR24D	PR28D	—	L8C_D1	COMPLEMENT
Y33	CR	4	PR23A	PR27A	—	—	TRUE
Y34	CR	4	PR23C	PR27C	PRCK1T	L9T_D0	TRUE
W33	CR	4	PR23D	PR27D	PRCK1C	L9C_D0	COMPLEMENT
W32	CR	4	PR22A	PR26A	—	—	TRUE

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	V_{DDIO} Bank	V_{REF} Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
V30	CR	4	PR22C	PR26C	—	L10T_A0	TRUE
V31	CR	4	PR22D	PR26D	V_{REF}_CR_04	L10C_A0	COMPLEMENT
V33	CR	5	PR21C	PR25C	—	L11T_A0	TRUE
V32	CR	5	PR21D	PR25D	—	L11C_A0	COMPLEMENT
U33	CR	5	PR20A	PR24A	—	L12T_A0	TRUE
U32	CR	5	PR20B	PR24B	—	L12C_A0	COMPLEMENT
T34	CR	5	PR20C	PR24C	PRCK0T	L13T_D2	TRUE
U31	CR	5	PR20D	PR24D	PRCK0C	L13C_D2	COMPLEMENT
T33	CR	5	PR19A	PR23A	—	—	TRUE
T32	CR	5	PR19C	PR23C	V_{REF}_CR_05	L14T_A0	TRUE
T31	CR	5	PR19D	PR23D	—	L14C_A0	COMPLEMENT
U30	CR	5	PR18A	PR22A	—	—	TRUE
R31	CR	5	PR18C	PR22C	—	L15T_D1	TRUE
R34	CR	5	PR18D	PR22D	—	L15C_D1	COMPLEMENT
R33	CR	5	PR17A	PR21A	—	—	TRUE
P34	CR	6	PR17C	PR21C	—	L16T_A1	TRUE
P32	CR	6	PR17D	PR21D	V_{REF}_CR_06	L16C_A1	COMPLEMENT
T30	CR	6	PR16A	PR20A	—	—	TRUE
P31	CR	6	PR16C	PR20C	—	L17T_A1	TRUE
P33	CR	6	PR16D	PR20D	—	L17C_A1	COMPLEMENT
R30	CR	6	PR16B	PR19B	—	—	COMPLEMENT
N33	CR	7	PR15C	PR19C	—	L18T_A1	TRUE
N31	CR	7	PR15D	PR19D	—	L18C_A1	COMPLEMENT
N34	CR	7	PR15B	PR18B	—	—	COMPLEMENT
M31	CR	7	PR14A	PR18C	—	L19T_A1	TRUE
M33	CR	7	PR14B	PR18D	—	L19C_A1	COMPLEMENT
P30	CR	7	PR15A	PR17A	—	—	TRUE
M34	CR	7	PR14C	PR17C	V_{REF}_CR_07	L20T_D1	TRUE
L32	CR	7	PR14D	PR17D	—	L20C_D1	COMPLEMENT
L31	CR	8	PR13B	PR16D	—	—	COMPLEMENT
L33	CR	8	PR13A	PR15A	—	—	TRUE
K34	CR	8	PR13C	PR15C	—	L21T_A0	TRUE
K33	CR	8	PR13D	PR15D	—	L21C_A0	COMPLEMENT
K32	CR	8	PR12A	PR14A	—	—	TRUE
N30	CR	8	PR12C	PR14C	V_{REF}_CR_08	L22T_D2	TRUE
K31	CR	8	PR12D	PR14D	—	L22C_D2	COMPLEMENT
R21	CR	—	V_{SS}	V_{SS}	—	—	—

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
R22	CR	—	VSS	VSS	—	—	—
T16	CR	—	VSS	VSS	—	—	—
T17	CR	—	VSS	VSS	—	—	—
T18	CR	—	VSS	VSS	—	—	—
T19	CR	—	VSS	VSS	—	—	—
U16	CR	—	VSS	VSS	—	—	—
U17	CR	—	VSS	VSS	—	—	—
U18	CR	—	VSS	VSS	—	—	—
U19	CR	—	VSS	VSS	—	—	—
V1	CR	—	VSS	VSS	—	—	—
U13	CR	—	VDD15	VDD15	—	—	—
U14	CR	—	VDD15	VDD15	—	—	—
U15	CR	—	VDD15	VDD15	—	—	—
U20	CR	—	VDD15	VDD15	—	—	—
U21	CR	—	VDD15	VDD15	—	—	—
U22	CR	—	VDD15	VDD15	—	—	—
L34	CR	—	VDDIO_CR	VDDIO_CR	—	—	—
M32	CR	—	VDDIO_CR	VDDIO_CR	—	—	—
R32	CR	—	VDDIO_CR	VDDIO_CR	—	—	—
U34	CR	—	VDDIO_CR	VDDIO_CR	—	—	—
W34	CR	—	VDDIO_CR	VDDIO_CR	—	—	—
Y32	CR	—	VDDIO_CR	VDDIO_CR	—	—	—
AC32	CR	—	VDDIO_CR	VDDIO_CR	—	—	—
AD34	CR	—	VDDIO_CR	VDDIO_CR	—	—	—
AM26	BR	1	PB30A	PB37A	—	—	TRUE
AP27	BR	1	PB30C	PB37C	—	L1T_A0	TRUE
AN27	BR	1	PB30D	PB37D	—	L1C_A0	COMPLEMENT
AK25	BR	1	PB31C	PB38C	VREF_BR_01	L2T_D0	TRUE
AL26	BR	1	PB31D	PB38D	—	L2C_D0	COMPLEMENT
AM27	BR	1	PB32C	PB39C	—	L3T_D1	TRUE
AK26	BR	1	PB32D	PB39D	—	L3C_D1	COMPLEMENT
AP28	BR	2	PB33C	PB40C	—	L4T_A0	TRUE
AN28	BR	2	PB33D	PB40D	VREF_BR_02	L4C_A0	COMPLEMENT
AL27	BR	2	PB34A	PB41C	—	L5T_A0	TRUE
AL28	BR	2	PB34B	PB41D	—	L5C_A0	COMPLEMENT
AK27	BR	2	PB34C	PB42C	—	—	TRUE
AM28	BR	3	PB35A	PB43A	—	—	TRUE
AN29	BR	3	PB35B	PB43D	—	—	COMPLEMENT

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
AK28	BR	3	PB35C	PB44C	—	L6T_D1	TRUE
AM29	BR	3	PB35D	PB44D	VREF_BR_03	L6C_D1	COMPLEMENT
AP29	BR	3	PB36B	PB45B	—	L7C_A2	COMPLEMENT
AL29	BR	3	PB36A	PB45A	—	L7T_A2	TRUE
AP30	BR	3	PB36C	PB45C	—	L8T_A0	TRUE
AN30	BR	3	PB36D	PB45D	—	L8C_A0	COMPLEMENT
AK29	BR	4	PB37A	PB46C	—	L9T_D1	TRUE
AM30	BR	4	PB37B	PB46D	VREF_BR_04	L9C_D1	COMPLEMENT
AL30	BR	4	PB37C	PB47C	PLL_CK5T/ PPLL	L10T_D2	TRUE
AP31	BR	4	PB37D	PB47D	PLL_CK5C/ PPLL	L10C_D2	COMPLEMENT
AJ30	BR	5	PR38A	PR46C	PLL_CK4T/PLL2 (155.52 MHz)	L11T_D1	TRUE
AK32	BR	5	PR38B	PR46D	PLL_CK4C/PLL2 (155.52 MHz)	L11C_D1	COMPLEMENT
AL33	BR	5	PR38C	PR45C	—	L12T_D2	TRUE
AH30	BR	5	PR38D	PR45D	—	L12C_D2	COMPLEMENT
AL34	BR	5	PR37C	PR44C	VREF_BR_05	L13T_D2	TRUE
AJ31	BR	5	PR37D	PR44D	—	L13C_D2	COMPLEMENT
AJ32	BR	6	PR36C	PR43C	—	L14T_D0	TRUE
AH31	BR	6	PR36D	PR43D	—	L14C_D0	COMPLEMENT
AK33	BR	6	PR35A	PR42C	—	L15T_D2	TRUE
AG30	BR	6	PR35B	PR42D	—	L15C_D2	COMPLEMENT
AK34	BR	6	PR35C	PR41C	VREF_BR_06	L16T_D0	TRUE
AJ33	BR	6	PR35D	PR41D	—	L16C_D0	COMPLEMENT
AF30	BR	7	PR34A	PR40A	—	—	TRUE
AJ34	BR	7	PR34C	PR40C	—	L17T_D2	TRUE
AG31	BR	7	PR34D	PR40D	—	L17C_D2	COMPLEMENT
AH32	BR	7	PR33A	PR39A	—	—	TRUE
AG32	BR	7	PR33C	PR39C	—	L18T_D0	TRUE
AH33	BR	7	PR33D	PR39D	VREF_BR_07	L18C_D0	COMPLEMENT
AE30	BR	7	PR32A	PR38A	—	—	TRUE
AH34	BR	7	PR32C	PR38C	—	L19T_D2	TRUE
AF31	BR	7	PR32D	PR38D	—	L19C_D2	COMPLEMENT
AF32	BR	8	PR31A	PR37A	—	—	TRUE
AG33	BR	8	PR31C	PR37C	—	L20T_D1	TRUE
AE31	BR	8	PR31D	PR37D	VREF_BR_08	L20C_D1	COMPLEMENT
AG34	BR	8	PR30A	PR36A	—	—	TRUE

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
AF33	BR	8	PR30B	PR36B	—	—	COMPLEMENT
AD30	BR	8	PR30C	PR36C	—	L21T_D3	TRUE
AF34	BR	8	PR30D	PR36D	—	L21C_D3	COMPLEMENT
AN31	BR	—	VDD33	VDD33	—	—	—
AK31	BR	—	VDD33	VDD33	—	—	—
V16	BR	—	VSS	VSS	—	—	—
V17	BR	—	VSS	VSS	—	—	—
V18	BR	—	VSS	VSS	—	—	—
V19	BR	—	VSS	VSS	—	—	—
V34	BR	—	VSS	VSS	—	—	—
W16	BR	—	VSS	VSS	—	—	—
W17	BR	—	VSS	VSS	—	—	—
W18	BR	—	VSS	VSS	—	—	—
W19	BR	—	VSS	VSS	—	—	—
Y13	BR	—	VSS	VSS	—	—	—
Y14	BR	—	VSS	VSS	—	—	—
V13	BR	—	VDD15	VDD15	—	—	—
V14	BR	—	VDD15	VDD15	—	—	—
V15	BR	—	VDD15	VDD15	—	—	—
V20	BR	—	VDD15	VDD15	—	—	—
V21	BR	—	VDD15	VDD15	—	—	—
V22	BR	—	VDD15	VDD15	—	—	—
AK30	BR	—	VDDIO_BR	VDDIO_BR	—	—	—
AL32	BR	—	VDDIO_BR	VDDIO_BR	—	—	—
AM31	BR	—	VDDIO_BR	VDDIO_BR	—	—	—
AM33	BR	—	VDDIO_BR	VDDIO_BR	—	—	—
AM34	BR	—	VDDIO_BR	VDDIO_BR	—	—	—
AN32	BR	—	VDDIO_BR	VDDIO_BR	—	—	—
AP32	BR	—	VDDIO_BR	VDDIO_BR	—	—	—
AN15	BC	1	PB17A	PB21A	—	—	TRUE
AN16	BC	1	PB17C	PB21C	—	L1T_D2	TRUE
AK17	BC	1	PB17D	PB21D	—	L1C_D2	COMPLEMENT
AL16	BC	1	PB18A	PB22A	—	—	TRUE
AM16	BC	1	PB18C	PB22C	VREF_BC_01	L2T_A1	TRUE
AP16	BC	1	PB18D	PB22D	—	L2C_A1	COMPLEMENT
AN17	BC	2	PB19A	PB23A	—	L3T_A1	TRUE
AL17	BC	2	PB19B	PB23B	—	L3C_A1	COMPLEMENT
AM17	BC	2	PB19C	PB23C	PBCK0T	L4T_A0	TRUE

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
AM18	BC	2	PB19D	PB23D	PBCK0C	L4C_A0	COMPLEMENT
AN18	BC	2	PB20B	PB24B	—	L5C_A1	COMPLEMENT
AL18	BC	2	PB20A	PB24A	—	L5T_A1	TRUE
AL19	BC	2	PB20C	PB24C	VREF_BC_02	L6T_D0	TRUE
AK18	BC	2	PB20D	PB24D	—	L6C_D0	COMPLEMENT
AM19	BC	2	PB21A	PB25C	—	L7T_A0	TRUE
AN19	BC	2	PB21B	PB25D	—	L7C_A0	COMPLEMENT
AP20	BC	3	PB21C	PB26C	—	L8T_A0	TRUE
AN20	BC	3	PB21D	PB26D	VREF_BC_03	L8C_A0	COMPLEMENT
AL20	BC	3	PB22A	PB27A	—	—	TRUE
AP21	BC	3	PB22C	PB27C	—	L9T_A0	TRUE
AN21	BC	3	PB22D	PB27D	—	L9C_A0	COMPLEMENT
AK19	BC	3	PB23A	PB28A	—	—	TRUE
AM21	BC	3	PB23C	PB28C	PBCK1T	L10T_A0	TRUE
AL21	BC	3	PB23D	PB28D	PBCK1C	L10C_A0	COMPLEMENT
AK20	BC	3	PB24A	PB29A	—	—	TRUE
AP22	BC	4	PB24C	PB29C	—	L11T_A0	TRUE
AN22	BC	4	PB24D	PB29D	—	L11C_A0	COMPLEMENT
AK21	BC	4	PB25A	PB30A	—	—	TRUE
AL22	BC	4	PB25C	PB30C	—	L12T_A0	TRUE
AL23	BC	4	PB25D	PB30D	VREF_BC_04	L12C_A0	COMPLEMENT
AK22	BC	4	PB26A	PB31C	—	L13T_D2	TRUE
AN23	BC	4	PB26B	PB31D	—	L13C_D2	COMPLEMENT
AP23	BC	5	PB26C	PB32C	—	L14T_A3	TRUE
AK23	BC	5	PB26D	PB32D	VREF_BC_05	L14C_A3	COMPLEMENT
AN24	BC	5	PB27A	PB33C	—	L15T_A0	TRUE
AM24	BC	5	PB27B	PB33D	—	L15C_A0	COMPLEMENT
AL24	BC	5	PB27C	PB34C	—	L16T_D2	TRUE
AP25	BC	5	PB27D	PB34D	—	L16T_D2	COMPLEMENT
AN25	BC	6	PB28A	PB35A	—	—	TRUE
AK24	BC	6	PB28C	PB35C	—	L17T_D3	TRUE
AP26	BC	6	PB28D	PB35D	VREF_BC_06	L17C_D3	COMPLEMENT
AN26	BC	6	PB29A	PB36A	—	—	TRUE
AL25	BC	6	PB29C	PB36C	—	L18T_A0	TRUE
AM25	BC	6	PB29D	PB36D	—	L18C_A0	COMPLEMENT
Y15	BC	—	VSS	VSS	—	—	—
Y20	BC	—	VSS	VSS	—	—	—
Y21	BC	—	VSS	VSS	—	—	—

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
Y22	BC	—	VSS	VSS	—	—	—
AA13	BC	—	VSS	VSS	—	—	—
AA14	BC	—	VSS	VSS	—	—	—
AA15	BC	—	VSS	VSS	—	—	—
AA20	BC	—	VSS	VSS	—	—	—
AA21	BC	—	VSS	VSS	—	—	—
AA22	BC	—	VSS	VSS	—	—	—
AB3	BC	—	VSS	VSS	—	—	—
W13	BC	—	VDD15	VDD15	—	—	—
W14	BC	—	VDD15	VDD15	—	—	—
W15	BC	—	VDD15	VDD15	—	—	—
W20	BC	—	VDD15	VDD15	—	—	—
W21	BC	—	VDD15	VDD15	—	—	—
W22	BC	—	VDD15	VDD15	—	—	—
AM12	BC	—	VDDIO_BC	VDDIO_BC	—	—	—
AM15	BC	—	VDDIO_BC	VDDIO_BC	—	—	—
AM20	BC	—	VDDIO_BC	VDDIO_BC	—	—	—
AM23	BC	—	VDDIO_BC	VDDIO_BC	—	—	—
AP11	BC	—	VDDIO_BC	VDDIO_BC	—	—	—
AP17	BC	—	VDDIO_BC	VDDIO_BC	—	—	—
AP19	BC	—	VDDIO_BC	VDDIO_BC	—	—	—
AP24	BC	—	VDDIO_BC	VDDIO_BC	—	—	—
AF1	BL	1	PL32D	PL38D	D8	L1C_A0	COMPLEMENT
AF2	BL	1	PL32C	PL38C	VREF_BL_01	L1T_A0	TRUE
AE4	BL	1	PL32A	PL38A	—	—	TRUE
AF3	BL	1	PL33D	PL39D	D9	L2C_A0	COMPLEMENT
AF4	BL	1	PL33C	PL39C	D10	L2T_A0	TRUE
AE5	BL	2	PL34D	PL40D	—	L3C_D3	COMPLEMENT
AG1	BL	2	PL34C	PL40C	VREF_BL_02	L3T_D3	TRUE
AG2	BL	2	PL34B	PL41D	—	L4C_D2	COMPLEMENT
AF5	BL	2	PL34A	PL41C	—	L4T_D2	TRUE
AG3	BL	3	PL35B	PL42D	D11	L5C_A0	COMPLEMENT
AG4	BL	3	PL35A	PL42C	D12	L5T_A0	TRUE
AH1	BL	3	PL36D	PL43D	—	L6C_A1	COMPLEMENT
AH3	BL	3	PL36C	PL43C	—	L6T_A1	TRUE
AH4	BL	3	PL36B	PL44D	VREF_BL_03	L7C_D0	COMPLEMENT
AG5	BL	3	PL36A	PL44C	D13	L7T_D0	TRUE
AH2	BL	4	PL37D	PL44B	—	—	COMPLEMENT

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	V_{DDIO} Bank	V_{REF} Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
AJ2	BL	4	PL37B	PL45D	—	L8C_D2	COMPLEMENT
AH5	BL	4	PL37A	PL45C	V_{REF}_BL_04	L8T_D2	TRUE
AJ3	BL	4	PL38C	PL45A	—	—	TRUE
AJ4	BL	4	PL38B	PL46D	—	—	COMPLEMENT
AJ1	BL	4	PL38A	PL46A	—	—	TRUE
AK1	BL	4	PL39D	PL47D	PLL_CK7C/ HPPLL	L9C_A0	COMPLEMENT
AK2	BL	4	PL39C	PL47C	PLL_CK7T/ HPPLL	L9T_A0	TRUE
AJ5	BL	4	PL39B	PL47B	—	L10C_D1	COMPLEMENT
AK3	BL	4	PL39A	PL47A	—	L10T_D1	TRUE
AL5	BL	5	PB2A	PB2A	DP2	L11T_A0	TRUE
AM5	BL	5	PB2B	PB2B	—	L11C_A0	COMPLEMENT
AN4	BL	5	PB2C	PB2C	PLL_CK6T/ PPLL	L12T_D2	TRUE
AK7	BL	5	PB2D	PB2D	PLL_CK6C/ PPLL	L12C_D2	COMPLEMENT
AP4	BL	5	PB3A	PB3A	—	—	TRUE
AL6	BL	5	PB3C	PB3C	—	L13T_A0	TRUE
AM6	BL	5	PB3D	PB3D	—	L13C_A0	COMPLEMENT
AL7	BL	5	PB4A	PB4C	V_{REF}_BL_05	L14T_D1	TRUE
AN5	BL	5	PB4B	PB4D	DP3	L14C_D1	COMPLEMENT
AK8	BL	6	PB4C	PB5C	—	L15T_D3	TRUE
AP5	BL	6	PB4D	PB5D	—	L15C_D3	COMPLEMENT
AN6	BL	6	PB5C	PB6C	V_{REF}_BL_06	L16T_D2	TRUE
AK9	BL	6	PB5D	PB6D	D14	L16C_D2	COMPLEMENT
AP6	BL	6	PB6A	PB7C	—	L17T_D2	TRUE
AL8	BL	6	PB6B	PB7D	—	L17C_D2	COMPLEMENT
AM7	BL	7	PB6C	PB8C	D15	L18T_A0	TRUE
AM8	BL	7	PB6D	PB8D	D16	L18C_A0	COMPLEMENT
AN7	BL	7	PB7A	PB9A	—	—	TRUE
AK10	BL	7	PB7C	PB9C	D17	L19T_D3	TRUE
AP7	BL	7	PB7D	PB9D	D18	L19C_D3	COMPLEMENT
AL9	BL	7	PB8A	PB10A	—	—	TRUE
AK11	BL	7	PB8C	PB10C	V_{REF}_BL_07	L20T_D1	TRUE
AM9	BL	7	PB8D	PB10D	D19	L20C_D1	COMPLEMENT
AN8	BL	8	PB9A	PB11A	—	—	TRUE
AL10	BL	8	PB9C	PB11C	D20	L21T_D2	TRUE
AP8	BL	8	PB9D	PB11D	D21	L21C_D2	COMPLEMENT

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	VDDIO Bank	VREF Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
AN9	BL	8	PB10A	PB12A	—	—	TRUE
AP9	BL	8	PB10C	PB12C	VREF_BL_08	L22T_D1	TRUE
AM10	BL	8	PB10D	PB12D	D22	L22C_D1	COMPLEMENT
AK12	BL	9	PB11A	PB13A	—	L23T_D0	TRUE
AL11	BL	9	PB11B	PB13B	—	L23C_D0	COMPLEMENT
AN10	BL	9	PB11C	PB13C	D23	L24T_A0	TRUE
AP10	BL	9	PB11D	PB13D	D24	L24C_A0	COMPLEMENT
AN11	BL	9	PB12A	PB14A	—	L25T_A0	TRUE
AM11	BL	9	PB12B	PB14B	—	L25C_A0	COMPLEMENT
AK13	BL	9	PB12C	PB14C	VREF_BL_09	L26T_D0	TRUE
AL12	BL	9	PB12D	PB14D	D25	L26C_D0	COMPLEMENT
AN12	BL	9	PB13A	PB15C	—	L27T_D2	TRUE
AK14	BL	9	PB13B	PB15D	—	L27C_D2	COMPLEMENT
AP12	BL	10	PB13C	PB16C	D26	L28T_A0	TRUE
AP13	BL	10	PB13D	PB16D	D27	L28C_A0	COMPLEMENT
AL13	BL	10	PB14A	PB17C	—	L29T_A1	TRUE
AN13	BL	10	PB14B	PB17D	—	L29C_A1	COMPLEMENT
AP14	BL	10	PB14C	PB18C	VREF_BL_10	L30T_D3	TRUE
AK15	BL	10	PB14D	PB18D	D28	L30C_D3	COMPLEMENT
AN14	BL	11	PB15A	PB19A	—	—	TRUE
AM14	BL	11	PB15C	PB19C	D29	L31T_D1	TRUE
AK16	BL	11	PB15D	PB19D	D30	L31C_D1	COMPLEMENT
AL14	BL	11	PB16A	PB20A	—	—	TRUE
AP15	BL	11	PB16C	PB20C	VREF_BL_11	L32T_A2	TRUE
AL15	BL	11	PB16D	PB20D	D31	L32C_A2	COMPLEMENT
AK4	BL	—	PTEMP	PTEMP	—	—	—
AL1	BL	—	LVDS_R	LVDS_R	—	—	—
AL2	BL	—	VDD33	VDD33	—	—	—
AK6	BL	—	VDD33	VDD33	—	—	—
AB13	BL	—	VSS	VSS	—	—	—
AB14	BL	—	VSS	VSS	—	—	—
AB15	BL	—	VSS	VSS	—	—	—
AB20	BL	—	VSS	VSS	—	—	—
AB21	BL	—	VSS	VSS	—	—	—
AB22	BL	—	VSS	VSS	—	—	—
AB32	BL	—	VSS	VSS	—	—	—
AL4	BL	—	VSS	VSS	—	—	—
AL31	BL	—	VSS	VSS	—	—	—

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	$V_{DD}IO$ Bank	V_{REF} Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
AM3	BL	—	V_{SS}	V_{SS}	—	—	—
AM13	BL	—	V_{SS}	V_{SS}	—	—	—
Y16	BL	—	$V_{DD}15$	$V_{DD}15$	—	—	—
Y17	BL	—	$V_{DD}15$	$V_{DD}15$	—	—	—
Y18	BL	—	$V_{DD}15$	$V_{DD}15$	—	—	—
Y19	BL	—	$V_{DD}15$	$V_{DD}15$	—	—	—
AA16	BL	—	$V_{DD}15$	$V_{DD}15$	—	—	—
AA17	BL	—	$V_{DD}15$	$V_{DD}15$	—	—	—
AK5	BL	—	$V_{DD}IO\_BL$	$V_{DD}IO\_BL$	—	—	—
AL3	BL	—	$V_{DD}IO\_BL$	$V_{DD}IO\_BL$	—	—	—
AM1	BL	—	$V_{DD}IO\_BL$	$V_{DD}IO\_BL$	—	—	—
AM2	BL	—	$V_{DD}IO\_BL$	$V_{DD}IO\_BL$	—	—	—
AM4	BL	—	$V_{DD}IO\_BL$	$V_{DD}IO\_BL$	—	—	—
AN3	BL	—	$V_{DD}IO\_BL$	$V_{DD}IO\_BL$	—	—	—
AP3	BL	—	$V_{DD}IO\_BL$	$V_{DD}IO\_BL$	—	—	—
L4	CL	1	PL12D	PL14D	A15	L1C_D1	COMPLEMENT
K2	CL	1	PL12C	PL14C	A14	L1T_D1	TRUE
K1	CL	1	PL12B	PL15D	—	L2C_D0	COMPLEMENT
L2	CL	1	PL12A	PL15C	—	L2T_D0	TRUE
L3	CL	1	PL13D	PL16D	$V_{REF}\_CL\_01$	L3C_D1	COMPLEMENT
N5	CL	1	PL13C	PL16C	D4	L3T_D1	TRUE
M4	CL	2	PL13B	PL17D	—	L4C_A1	COMPLEMENT
M2	CL	2	PL13A	PL17C	—	L4T_A1	TRUE
P5	CL	2	PL14D	PL18D	RDY/$\overline{BUSY}$/ RCLK	L5C_D3	COMPLEMENT
M1	CL	2	PL14C	PL18C	$V_{REF}\_CL\_02$	L5T_D3	TRUE
N1	CL	2	PL15D	PL19D	A13	L6C_A2	COMPLEMENT
N4	CL	2	PL15C	PL19C	A12	L6T_A2	TRUE
N2	CL	3	PL16D	PL20D	—	L7C_D0	COMPLEMENT
P1	CL	3	PL16C	PL20C	—	L7T_D0	TRUE
R5	CL	3	PL16A	PL20A	—	—	TRUE
P2	CL	3	PL17D	PL21D	A11	L8C_A0	COMPLEMENT
P3	CL	3	PL17C	PL21C	$V_{REF}\_CL\_03$	L8T_A0	TRUE
T5	CL	3	PL17A	PL21A	—	—	TRUE
P4	CL	3	PL18D	PL22D	—	L9C_D2	COMPLEMENT
R1	CL	3	PL18C	PL22C	—	L9T_D2	TRUE
R2	CL	3	PL18A	PL22A	—	L10T_A1	TRUE
R4	CL	3	PL18B	PL22B	—	L10C_A1	COMPLEMENT

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	$V_{DD}IO$ Bank	V_{REF} Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
U5	CL	4	PL19D	PL23D	$\overline{RD}/\overline{MPI\_STRB}$	L11C_D0	COMPLEMENT
T4	CL	4	PL19C	PL23C	V_{REF}_CL_04	L11T_D0	TRUE
T1	CL	4	PL19A	PL23A	—	L12T_D3	TRUE
V5	CL	4	PL19B	PL23B	—	L12C_D3	COMPLEMENT
T2	CL	4	PL20D	PL24D	PLCK0C	L13C_A0	COMPLEMENT
T3	CL	4	PL20C	PL24C	PLCK0T	L13T_A0	TRUE
U4	CL	4	PL20B	PL24B	—	L14C_A0	COMPLEMENT
U3	CL	4	PL20A	PL24A	—	L14T_A0	TRUE
U2	CL	5	PL21D	PL25D	A10	L15C_A0	COMPLEMENT
V2	CL	5	PL21C	PL25C	A9	L15T_A0	TRUE
V3	CL	5	PL21B	PL25B	—	L16C_A0	COMPLEMENT
V4	CL	5	PL21A	PL25A	—	L16T_A0	TRUE
W5	CL	5	PL22D	PL26D	A8	L17C_A2	COMPLEMENT
W2	CL	5	PL22C	PL26C	V_{REF}_CL_05	L17T_A2	TRUE
W3	CL	5	PL23D	PL27D	—	L18C_D1	COMPLEMENT
Y1	CL	5	PL23C	PL27C	—	L18T_D1	TRUE
Y2	CL	5	PL23A	PL27A	—	—	TRUE
W4	CL	6	PL24D	PL28D	PLCK1C	L19C_D2	COMPLEMENT
AA1	CL	6	PL24C	PL28C	PLCK1T	L19T_D2	TRUE
AA2	CL	6	PL24A	PL28A	—	—	TRUE
Y5	CL	6	PL25D	PL29D	V_{REF}_CL_06	L20C_A0	COMPLEMENT
Y4	CL	6	PL25C	PL29C	A7	L20T_A0	TRUE
AA3	CL	6	PL25A	PL29A	—	—	TRUE
AA5	CL	6	PL26D	PL30D	A6	L21C_D3	COMPLEMENT
AB1	CL	6	PL26C	PL30C	A5	L21T_D3	TRUE
AB2	CL	7	PL26B	PL31D	—	—	COMPLEMENT
AA4	CL	7	PL27D	PL32D	$\overline{WR}$/MPI_RW	L22C_A0	COMPLEMENT
AB4	CL	7	PL27C	PL32C	V_{REF}_CL_07	L22T_A0	TRUE
AB5	CL	7	PL27B	PL33D	—	L23C_D3	COMPLEMENT
AC1	CL	7	PL27A	PL33C	—	L23T_D3	TRUE
AC2	CL	8	PL28D	PL34D	A4	L23C_A2	COMPLEMENT
AC5	CL	8	PL28C	PL34C	V_{REF}_CL_08	L23T_A2	TRUE
AD2	CL	8	PL29D	PL35D	A3	L23C_A0	COMPLEMENT
AD3	CL	8	PL29C	PL35C	A2	L23T_A0	TRUE
AC4	CL	8	PL29A	PL35A	—	—	TRUE

Pin Information (continued)

Table 46. OR4E6 680-Pin PBGAM Pinout (continued)

680 BGA Ball	$V_{DD}IO$ Bank	V_{REF} Group	General-Purpose User I/O		Additional Function	Pair	Differential
			OR4E4	OR4E6			
AE1	CL	8	PL30D	PL36D	A1	L24C_A0	COMPLEMENT
AE2	CL	8	PL30C	PL36C	A0	L24T_A0	TRUE
AD4	CL	8	PL31D	PL37D	DP0	L25C_D0	COMPLEMENT
AE3	CL	8	PL31C	PL37C	DP1	L25T_D0	TRUE
AD5	CL	8	PL31A	PL37A	—	—	TRUE
AM22	CL	—	V_{SS}	V_{SS}	—	—	—
AM32	CL	—	V_{SS}	V_{SS}	—	—	—
AN1	CL	—	V_{SS}	V_{SS}	—	—	—
AN2	CL	—	V_{SS}	V_{SS}	—	—	—
AN33	CL	—	V_{SS}	V_{SS}	—	—	—
AN34	CL	—	V_{SS}	V_{SS}	—	—	—
AP1	CL	—	V_{SS}	V_{SS}	—	—	—
AP2	CL	—	V_{SS}	V_{SS}	—	—	—
AP18	CL	—	V_{SS}	V_{SS}	—	—	—
AP33	CL	—	V_{SS}	V_{SS}	—	—	—
AP34	CL	—	V_{SS}	V_{SS}	—	—	—
AA18	CL	—	$V_{DD}15$	$V_{DD}15$	—	—	—
AA19	CL	—	$V_{DD}15$	$V_{DD}15$	—	—	—
AB16	CL	—	$V_{DD}15$	$V_{DD}15$	—	—	—
AB17	CL	—	$V_{DD}15$	$V_{DD}15$	—	—	—
AB18	CL	—	$V_{DD}15$	$V_{DD}15$	—	—	—
AB19	CL	—	$V_{DD}15$	$V_{DD}15$	—	—	—
L1	CL	—	$V_{DD}IO\_CL$	$V_{DD}IO\_CL$	—	—	—
M3	CL	—	$V_{DD}IO\_CL$	$V_{DD}IO\_CL$	—	—	—
R3	CL	—	$V_{DD}IO\_CL$	$V_{DD}IO\_CL$	—	—	—
U1	CL	—	$V_{DD}IO\_CL$	$V_{DD}IO\_CL$	—	—	—
W1	CL	—	$V_{DD}IO\_CL$	$V_{DD}IO\_CL$	—	—	—
Y3	CL	—	$V_{DD}IO\_CL$	$V_{DD}IO\_CL$	—	—	—
AC3	CL	—	$V_{DD}IO\_CL$	$V_{DD}IO\_CL$	—	—	—
AD1	CL	—	$V_{DD}IO\_CL$	$V_{DD}IO\_CL$	—	—	—

Package Thermal Characteristics Summary

There are three thermal parameters that are in common use: Θ_{JA}, ψ_{JC}, and Θ_{JC}. It should be noted that all the parameters are affected, to varying degrees, by package design (including paddle size) and choice of materials, the amount of copper in the test board or system board, and system airflow.

Θ_{JA}

This is the thermal resistance from junction to ambient (theta-JA, R-theta, etc.).

$$\Theta_{JA} = \frac{T_J - T_A}{Q}$$

where T_J is the junction temperature, T_A, is the ambient air temperature, and Q is the chip power.

Experimentally, Θ_{JA} is determined when a special thermal test die is assembled into the package of interest, and the part is mounted on the thermal test board. The diodes on the test chip are separately calibrated in an oven. The package/board is placed either in a JEDEC natural convection box or in the wind tunnel, the latter for forced convection measurements. A controlled amount of power (Q) is dissipated in the test chip's heater resistor, the chip's temperature (T_J) is determined by the forward drop on the diodes, and the ambient temperature (T_A) is noted. Note that Θ_{JA} is expressed in units of °C/W.

ψ_{JC}

This JEDEC designated parameter correlates the junction temperature to the case temperature. It is generally used to infer the junction temperature while the device is operating in the system. It is not considered a true thermal resistance, and it is defined by:

$$\psi_{JC} = \frac{T_J - T_C}{Q}$$

where T_C is the case temperature at top dead center, T_J is the junction temperature, and Q is the chip power. During the Θ_{JA} measurements described above, besides the other parameters measured, an additional temperature reading, T_C, is made with a thermocouple attached at top-dead-center of the case. ψ_{JC} is also expressed in units of °C/W.

Θ_{JC}

This is the thermal resistance from junction to case. It is most often used when attaching a heat sink to the top of the package. It is defined by:

$$\Theta_{JC} = \frac{T_J - T_C}{Q}$$

The parameters in this equation have been defined above. However, the measurements are performed with the case of the part pressed against a water-cooled heat sink to draw most of the heat generated by the chip out the top of the package. It is this difference in the measurement process that differentiates Θ_{JC} from ψ_{JC}. Θ_{JC} is a true thermal resistance and is expressed in units of °C/W.

Θ_{JB}

This is the thermal resistance from junction to board (Θ_{JB}). It is defined by:

$$\Theta_{JB} = \frac{T_J - T_B}{Q}$$

where T_B is the temperature of the board adjacent to a lead measured with a thermocouple. The other parameters on the right-hand side have been defined above. This is considered a true thermal resistance, and the measurement is made with a water-cooled heat sink pressed against the board to draw most of the heat out of the leads. Note that Θ_{JB} is expressed in units of °C/W, and that this parameter and the way it is measured are still in JEDEC committee.

Package Thermal Characteristics

Table 47. *ORCA* Series 4 FPGAs Plastic Package Thermal Guidelines

Package	ΘJA (°C/W)			T = 70 °C Max, TJ = 125 °C Max 0 fpm (W)
	0 fpm	200 fpm	500 fpm	
352-Pin PBGA	19.0	16.0	15.0	2.9
432-Pin EBGA	11.0	8.5	7.5	5.0
680-Pin PBGAM	14.5	TBD	TBD	3.8

Package Coplanarity

The coplanarity limits of the Lucent packages are as follows:

- PBGA: 8.0 mils
- EBGA: 8.0 mils
- PBGAM: 8.0 mils

Package Parasitics

The electrical performance of an IC package, such as signal quality and noise sensitivity, is directly affected by the package parasitics. Table 48 lists eight parasitics associated with the *ORCA* packages. These parasitics represent the contributions of all components of a package, which include the bond wires, all internal package routing, and the external leads.

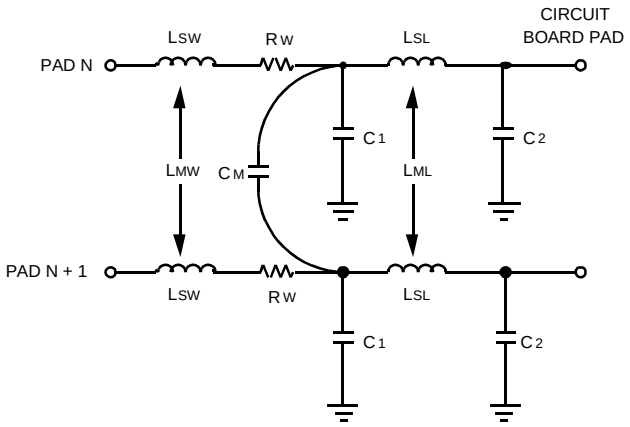
Four inductances in nH are listed: L_{SW} and L_{SL}, the self-inductance of the lead; and L_{MW} and L_{ML}, the mutual inductance to the nearest neighbor lead. These parameters are important in determining ground bounce noise and inductive crosstalk noise. Three capacitances in pF are listed: C_M, the mutual capacitance of the lead to the nearest neighbor lead; and C_1 and C_2, the total capacitance of the lead to all other leads (all other leads are assumed to be grounded). These parameters are important in determining capacitive crosstalk and the capacitive loading effect of the lead. Resistance values are in mΩ.

The parasitic values in Table 48 are for the circuit model of bond wire and package lead parasitics. If the mutual capacitance value is not used in the designer's model, then the value listed as mutual capacitance should be added to each of the C_1 and C_2 capacitors.

Table 48. *ORCA* Series 4 FPGAs Package Parasitics

Package Type	L_{SW}	L_{MW}	R_W	C_1	C_2	C_M	L_{SL}	L_{ML}
352-Pin PBGA	5	2	220	1.5	1.5	1.5	7—12	3—6
432-Pin EBGA	4.0	1.5	500	1.0	1.0	0.3	3.0—5.5	0.5—1
680-Pin PBGAM	3.8	1.3	250	1.0	1.0	0.3	2.8—5	0.5—1

Package Parasitics (continued)



5-3862(C)r2

Figure 47. Package Parasitics

Package Outline Diagrams

Terms and Definitions

Basic Size (BSC): The basic size of a dimension is the size from which the limits for that dimension are derived by the application of the allowance and the tolerance.

Design Size: The design size of a dimension is the actual size of the design, including an allowance for fit and tolerance.

Typical (TYP): When specified after a dimension, this indicates the repeated design size if a tolerance is specified or repeated basic size if a tolerance is not specified.

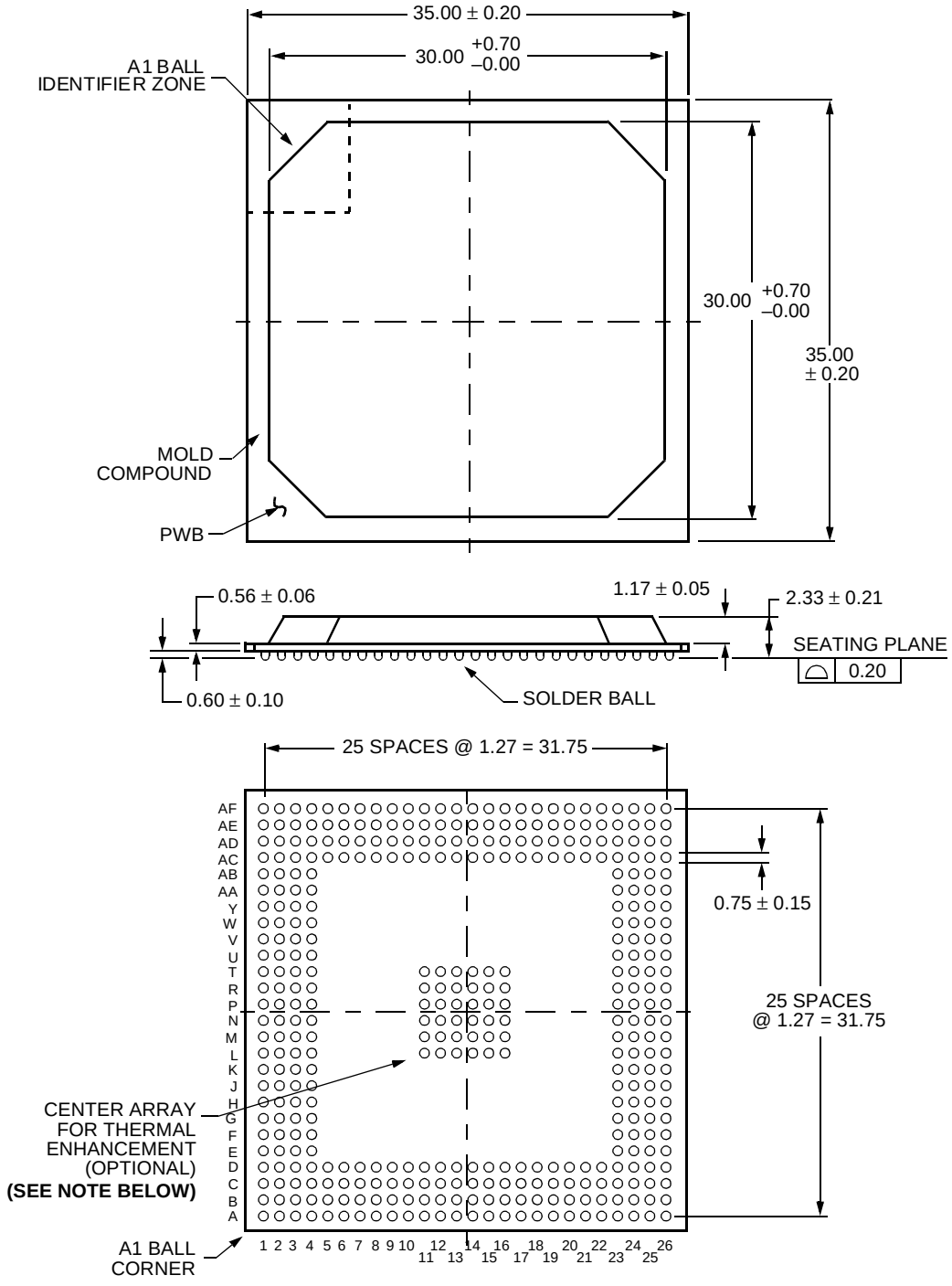
Reference (REF): The reference dimension is an untoleranced dimension used for informational purposes only. It is a repeated dimension or one that can be derived from other values in the drawing.

Minimum (MIN) or Maximum (MAX): Indicates the minimum or maximum allowable size of a dimension.

Package Outline Drawings

352-Pin PBGA

Dimensions are in millimeters.



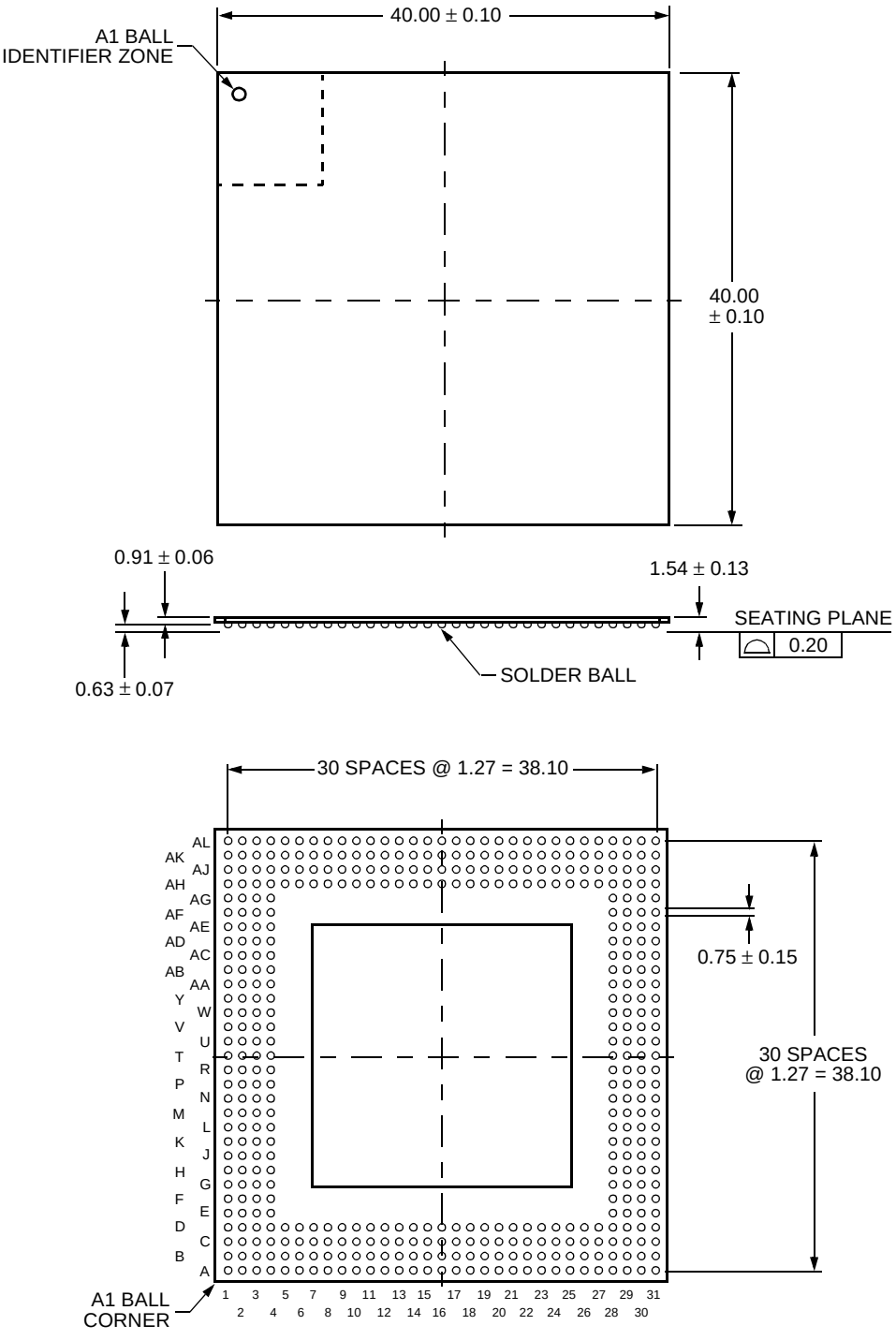
5-4407(F)

Note: Although the 36 thermal enhancement balls are stated as an option, they are standard on the 352 FPGA package.

Package Outline Drawings (continued)

432-Pin EBGA

Dimensions are in millimeters.

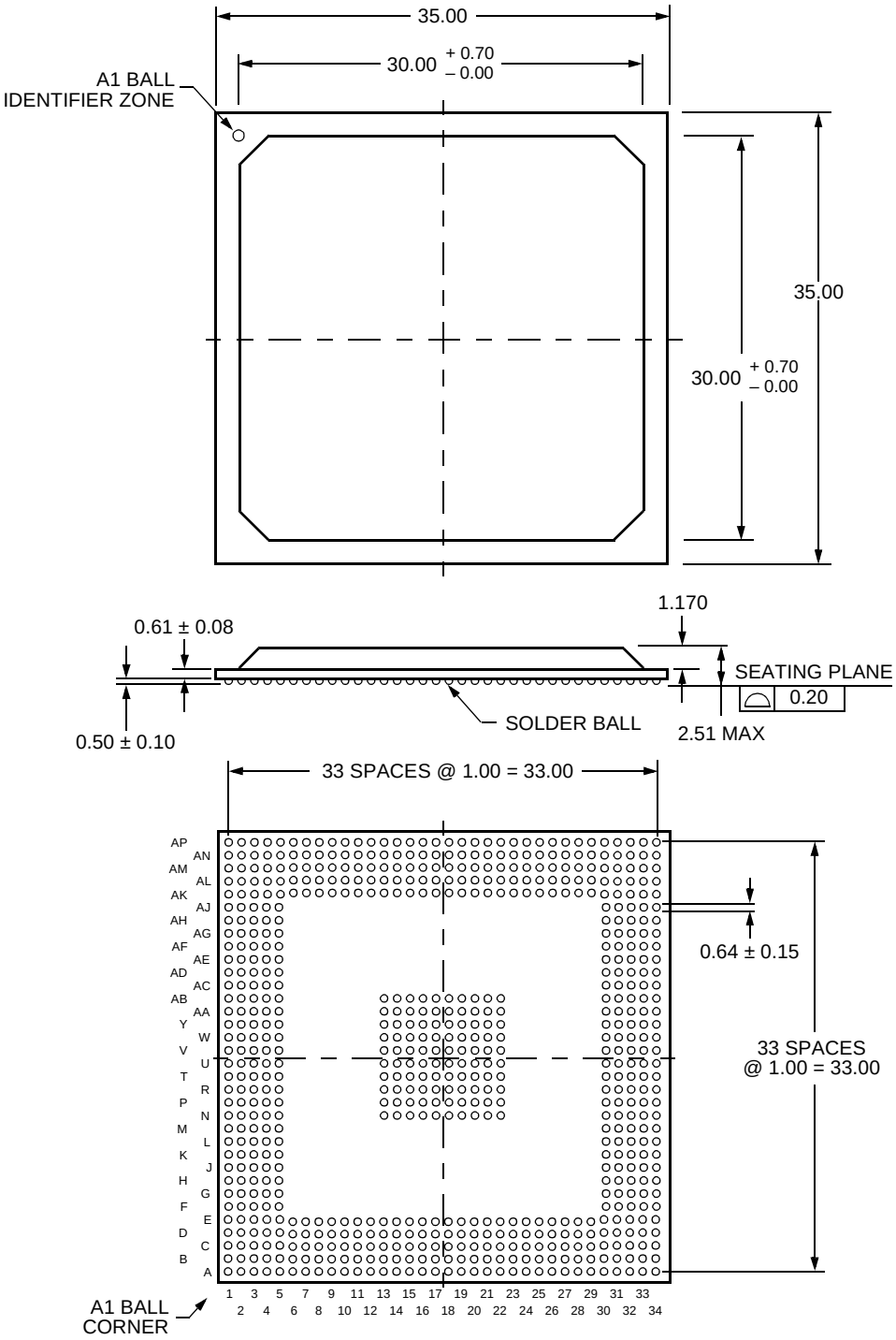


5-4409(F)

Package Outline Drawings (continued)

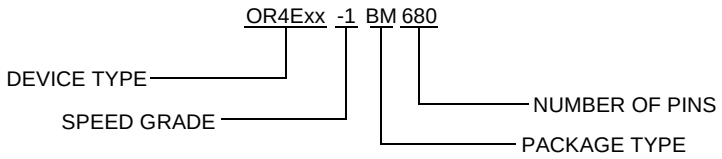
680-Pin PBGAM

Dimensions are in millimeters.



5-4406(F)

Ordering Information



5-6435 (F)

Table 49. Series 4 Package Matrix (Speed Grades)

Packages	352-Pin PBGA 1.27 mm	432-Pin EBGA 1.27 mm	680-Pin PBGAM 1 mm
OR4E2	-1/-2	-1/-2	—
OR4E4	-1/-2	-1/-2	-1/-2
OR4E6	-1/-2	-1/-2	-1/-2
OR4E10	—	—	-1/-2

Table 50. Package Options

Symbol	Description
BA	Plastic Ball Grid Array (PBGA)
BC	Enhanced Ball Grid Array (EBGA)
BM	Plastic Multilayer Ball Grid Array (PBGAM)

For additional information, contact your Microelectronics Group Account Manager or the following:
INTERNET: **http://www.lucent.com/micro**, or for FPGA information, **http://www.lucent.com/orca**
E-MAIL: **docmaster@micro.lucent.com**
N. AMERICA: Microelectronics Group, Lucent Technologies Inc., 555 Union Boulevard, Room 30L-15P-BA, Allentown, PA 18109-3286
1-800-372-2447, FAX 610-712-4106 (In CANADA: **1-800-553-2448**, FAX 610-712-4106)
ASIA PACIFIC: Microelectronics Group, Lucent Technologies Singapore Pte. Ltd., 77 Science Park Drive, #03-18 Cintech III, Singapore 118256
Tel. (65) 778 8833, FAX (65) 777 7495
CHINA: Microelectronics Group, Lucent Technologies (China) Co., Ltd., A-F2, 23/F, Zao Fong Universe Building, 1800 Zhong Shan Xi Road, Shanghai 200233 P. R. China **Tel. (86) 21 6440 0468**, **ext. 325**, FAX (86) 21 6440 0652
JAPAN: Microelectronics Group, Lucent Technologies Japan Ltd., 7-18, Higashi-Gotanda 2-chome, Shinagawa-ku, Tokyo 141, Japan
Tel. (81) 3 5421 1600, FAX (81) 3 5421 1700
EUROPE: Data Requests: MICROELECTRONICS GROUP DATALINE: **Tel. (44) 7000 582 368**, FAX (44) 1189 328 148
Technical Inquiries: GERMANY: **(49) 89 95086 0** (Munich), UNITED KINGDOM: **(44) 1344 865 900** (Ascot),
FRANCE: **(33) 1 40 83 68 00** (Paris), SWEDEN: **(46) 8 594 607 00** (Stockholm), FINLAND: **(358) 9 3507670** (Helsinki),
ITALY: **(39) 02 6608131** (Milan), SPAIN: **(34) 1 807 1441** (Madrid)

December 2000
DS01-024NCIP (Replaces DS00-221FPGA)

microelectronics group

Lucent Technologies
Bell Labs Innovations