



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

Understanding Embedded - FPGAs (Field Programmable Gate Array)

Embedded - FPGAs, or Field Programmable Gate Arrays, are advanced integrated circuits that offer unparalleled flexibility and performance for digital systems. Unlike traditional fixed-function logic devices, FPGAs can be programmed and reprogrammed to execute a wide array of logical operations, enabling customized functionality tailored to specific applications. This reprogrammability allows developers to iterate designs quickly and implement complex functions without the need for custom hardware.

Applications of Embedded - FPGAs

The versatility of Embedded - FPGAs makes them indispensable in numerous fields. In telecommunications.

Details

Product Status	Obsolete
Number of LABs/CLBs	-
Number of Logic Elements/Cells	10368
Total RAM Bits	113664
Number of I/O	204
Number of Gates	643000
Voltage - Supply	1.425V ~ 3.6V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 70°C (TA)
Package / Case	484-BBGA
Supplier Device Package	484-FPBGA (23x23)
Purchase URL	https://www.e-xfl.com/product-detail/lattice-semiconductor/ort42g5-3bm484c

Package Thermal Characteristics	
Summary	114
Θ_{JA}	114
Ψ_{JC}	114
Ψ_{JC}	115
Ψ_{JB}	115
FPSC Maximum Junction Temperature	115
Package Thermal Characteristics	115
Heat Sink Vendors for BGA Packages.....	115
Package Parasitics.....	116
Package Outline Drawings.....	116
Ordering Information	117

Programmable Features

- High-performance programmable logic:
 - 0.16 μm 7-level metal technology.
 - Internal performance of >250 MHz.
 - Over 400K usable system gates.
 - Meets multiple I/O interface standards.
 - 1.5V operation (30% less power than 1.8V operation) translates to greater performance.
 - Traditional I/O selections:
 - LVTTTL (3.3V) and LVCMOS (2.5V and 1.8V) I/Os.
 - Per pin-selectable I/O clamping diodes provide 3.3V 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 and II), HSTL (Class I, III, IV), ZBT, and DDR.
 - Double-ended: LVDS, bused-LVDS, and LVPECL. Programmable (on/off) internal parallel termination (100 Ω) is also supported for these I/Os.
 - New capability to (de)multiplex I/O signals:
 - New DDR on both input and output at rates up to 350 MHz (700 MHz effective rate).
 - New 2x and 4x downlink and uplink capability per I/O (i.e., 50 MHz internal to 200 MHz I/O).
 - Enhanced twin-block 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 operations.
 - 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 \rightarrow 1 MUX, new 8 \rightarrow 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.
 - Soft-Wired 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-wide, byte-wide, 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 a 10-bit decoder, and PAL[®]-like AND-OR-Invert (AOI) in each programmable logic cell.
 - New 200 MHz embedded block-port RAM blocks, two read ports, two write ports, and two sets of byte lane enables. Each embedded RAM block can be configured as:
-

-
- 1—512 x 18 (block-port, two read/two write) with optional built in arbitration.
 - 1—256 x 36 (dual-port, one read/one write).
 - 1—1K x 9 (dual-port, one read/one write).
 - 2—512 x 9 (dual-port, one read/one write for each).
 - 2 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).
- 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.
 - Built-in testability:
 - Full boundary scan (IEEE 1149.1 and Draft 1149.2 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.
 - 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. Multiplication of the input frequency up to 64x and division of the input frequency down to 1/64x possible.
 - New cycle stealing capability allows a typical 15% to 40% internal speed improvement after final place and route. This feature also enables compliance with many setup/hold and clock to out I/O specifications and may provide reduced ground bounce for output buses by allowing flexible delays of switching output buffers.
 - Per channel Pseudo-Random Bit Sequence (PRBS) generator and checker in FPGA logic.

Programmable Logic System Features

- PCI local bus compliant for FPGA I/Os.
 - Improved PowerPC[®] 860 and PowerPC II high-speed synchronous microprocessor 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 system bus facilitates communication among the microprocessor interface, configuration logic, Embedded Block RAM, FPGA logic, and embedded standard cell blocks.
 - Variable size based readback of configuration data capability with the built-in microprocessor interface and system bus.
 - Internal, 3-state, and bidirectional buses with simple control provided by the SLIC.
 - New clock routing structures for global and local clocking significantly increases speed and reduces skew (<200 ps for OR4E04).
 - New local clock routing structures allow creation of localized clock trees.
 - Two new edge clock routing structures allow up to six high-speed clocks on each edge of the device for improved setup/hold and clock to out performance.
 - New Double-Data Rate (DDR) and Zero-Bus Turn-around (ZBT) memory interfaces support the latest high-speed memory interfaces.
 - New 2x/4x uplink and downlink I/O capabilities interface high-speed external I/Os to reduced speed internal logic.
-

Description

What is an FPSC?

FPSCs, or field-programmable system chips, are devices that combine field-programmable logic with ASIC or mask-programmed logic on a single device. FPSCs provide the time to market and the flexibility of FPGAs, the design effort savings of using soft Intellectual Property (IP) cores, and the speed, design density, and economy of ASICs.

FPSC Overview

Lattice's Series 4 FPSCs are created from Series 4 ORCA FPGAs. To create a Series 4 FPSC, several columns of Programmable Logic Cells (see FPGA Logic Overview section for FPGA logic details) are added to an embedded logic core. Other than replacing some FPGA gates with ASIC gates, at greater than 10:1 efficiency, none of the FPGA functionality is changed—all of the Series 4 FPGA capability is retained including: the Embedded Block RAMs, MicroProcessor Interface (MPI), boundary scan, etc. The columns of programmable logic are replaced at the right of the device, allowing pins from the replaced columns to be used as I/O pins for the embedded core. The remainder of the device pins retain their FPGA functionality.

FPSC Gate Counting

The total gate count for an FPSC is the sum of its embedded core (standard-cell/ASIC gates) and its FPGA gates. Because FPGA gates are generally expressed as a usable range with a nominal value, the total FPSC gate count is sometimes expressed in the same manner. Standard-cell ASIC gates are, however, 10 to 25 times more silicon-area efficient than FPGA gates. Therefore, an FPSC with an embedded function is gate equivalent to an FPGA with a much larger gate count.

FPGA/Embedded Core Interface

The interface between the FPGA logic and the embedded core has been enhanced to allow for a greater number of interface signals than on previous FPSC architectures. Compared to bringing embedded core signals off-chip, this on-chip interface is much faster and requires less power. All of the delays for the interface are precharacterized and accounted for in the Lattice ispLEVER™ System software.

Series 4 based FPSCs expand this interface by providing a link between the embedded block and the multi-master 32-bit system bus in the FPGA logic. This system bus allows the core easy access to many of the FPGA logic functions including the Embedded Block RAMs and the microprocessor interface.

Clock spines also can pass across the FPGA/embedded core boundary. This allows for fast, low-skew clocking between the FPGA and the embedded core. Many of the special signals from the FPGA, such as DONE and global set/reset, are also available to the embedded core, making it possible to fully integrate the embedded core with the FPGA as a system.

For even greater system flexibility, FPGA configuration RAMs are available for use by the embedded core. This allows for user-programmable options in the embedded core, in turn allowing for greater flexibility. Multiple embedded core configurations may be designed into a single device with user-programmable control over which configurations are implemented, as well as the capability to change core functionality simply by reconfiguring the device.

FPSC Design Kit

Development is facilitated by an FPSC design kit which, together with ispLEVER System software and third-party synthesis and simulation engines, provides all software and documentation required to design and verify an FPSC implementation. Included in the kit are the FPSC configuration manager, compiled Verilog simulation models, HSPICE and/or IBIS models for I/O buffers, and complete online documentation. The kit's software coupled with the design environment, provides a seamless FPSC design environment. More information can be obtained by visiting the Lattice web site at www.latticesemi.com or contacting a local sales office.

The transceivers are controlled and configured through the system bus in the FPGA logic and through the external 8-bit microprocessor interface of the FPGA. Each channel has associated dedicated registers that are readable and writable. There are also global registers for control of common circuitry and functions.

The SERDES performs 8b/10b encoding and decoding for each channel. The 8b/10b transmission code can support either Ethernet or Fibre Channel specifications for serial encoding/decoding, special characters, and error detection.

The user can disable the 8b/10b decoder to receive raw 10-bit words which will be rate reduced by the SERDES. If this mode is chosen, the user must bypass the multi-channel alignment FIFOs.

The SERDES block contains its own dedicated PLLs for both transmit and receive clock generation. The user provides a reference clock of the appropriate frequency. The receiver PLLs extract the clock from the serial input data and retime the data with the recovered clock.

MUX/DEMUX Block

The MUX/DEMUX block converts the data format for the high speed serial links to a wide, low-speed format for crossing the CORE/FPGA interface. The intermediate interface to the SERDES macrocell runs at 1/10th the bit rate of the data lane. The MUX/DEMUX converts the data rate and bus width so the interface to the FPGA core can run at 1/4th this intermediate frequency, giving a range of 25.0-92.5 MHz for the data rates into and out of the FPGA logic.

Multi-channel Alignment FIFOs

In the ORT82G5, the eight incoming data channels (four per SERDES block) can be independent of each other or can be synchronized in several ways. Two channels within a SERDES block can be aligned together; channels A and B and/or channels C and D. Alternatively, four channels in a SERDES block can be aligned together to form a communication channel with a bandwidth of 10 Gbps. Finally, the alignment can be extended across both SERDES blocks to align all eight channels. Individual channels within an alignment group can be disabled (i.e., powered down) without disrupting other channels.

In the ORT42G5, the four incoming data channels (two per SERDES block) can be independent of each other or can be synchronized in two ways. Two channels, channels C and D, within either SERDES block can be aligned together. Alternatively, all four channels can be aligned together to form a communication channel with a bandwidth of 10 Gbps. Individual channels within an alignment group can be disabled (i.e., powered down) without disrupting other channels.

XAUI and Fibre Channel Link State Machines

Two separate link state machines are included in the architecture. A XAUI link state machine is included in the embedded core modeled after the IEEE 802.3ae standard. A separate state machine for Fibre Channel is also implemented.

FPGA/Embedded Core Interface

In 8b/10b mode, the FPGA logic will receive/transmit 32-bits of data (up to 92.5 MHz) and 4 K_CTRL bits from/to the embedded core. There are 8 data streams in each direction plus additional timing, status and control signals.

Data sent to the FPGA can be aligned using comma (/K/) characters or /A/ character as specified either by Fibre Channel or by IEEE 802.3ae for XAUI based interfaces. The alignment character is made available to the FPGA along with the data. The special characters K28.1, K28.5 and K28.7 are treated as valid comma characters by the SERDES.

If the receive channel alignment FIFOs are bypassed, then each channel will provide its own receive clock in addition to data and comma character detect signals. If the 8b/10b decoders are bypassed, then 40-bit data streams are passed to the FPGA logic. No channel alignment can be done in 8b/10b bypass mode.

Dual Port RAMs

In addition to the backplane interface blocks, there are two independent memory blocks in the ASB. Each memory block has a capacity of 4k words by 36 bits. It has one read port, one write port, and four byte-write-enable (active-low) signals. The read data from the memory block is registered so that it works as a pipelined synchronous memory block.

FPSC Configuration

Configuration of the ORT42G5 and ORT82G5 occurs in two stages: FPGA bitstream configuration and embedded core setup.

Prior to becoming operational, the FPGA goes through a sequence of states, including power up, initialization, configuration, start-up, and operation. The FPGA logic is configured by standard FPGA bit stream configuration means as discussed in the Series 4 FPGA data sheet.

After the FPGA configuration is complete, the options for the embedded core are set based on the contents of registers that are accessed through the FPGA system bus.

The system bus itself can be driven by an external PowerPC compliant microprocessor via the MPI block or via a user master interface in FPGA logic. A simple IP block that drives the system by using the user register interface and very little FPGA logic is available in the MPI/System Bus Technical Note. This IP block sets up the embedded core via a state machine and allows the ORT42G5 and ORT82G5 to work in an independent system without an external microprocessor interface.

Backplane Transceiver Core Detailed Description

The following sections describe the various logic blocks in the Embedded Core portion of the FPSC. The FPGA section of the FPSC is identical to an ORCA OR4E04 FPGA except that the pads on one edge of the FPGA chip are replaced by the Embedded Core. For a detailed description of the programmable logic functions, please see the ORCA Series 4 FPGA Data Sheet and related application and technical notes.

The major functional blocks in the Embedded Core include:

- Two SERializer-DESerializer (SERDES) blocks and Clock and Data Recovery (CDR) circuitry
- 8b/10b encoder/decoders
- Transmit pre-emphasis circuitry
- 4-to-1 multiplexers (MUX) and 1-to-4 demultiplexers (DEMUX)
- Fibre channel synchronization state machine
- XAUI link alignment state machine
- Alignment FIFOs
- Embedded 4K x 36 RAM blocks (independent from transceiver logic).

A top level block diagram of the Embedded Core Logic is shown in Figure 2. The Embedded RAM blocks are not shown. The external pins for the Embedded Core are listed later in this data sheet in Table 41 and the signals at the Transceiver Embedded Core/FPGA interface for the ORT42G5 are listed in Table 8, Table 9 and Table 11; and for the ORT82G5, in Table 8, Table 10 and Table 12.

Following the definitions and conventions used in defining the 8b/10b coding rules, each valid coded character has a name corresponding to its 8-bit binary value:

- Dxx.y for data characters
- Kxx.y for special characters
- xx = the 5-bit input value, base 10, for bits ABCDE
- y = the 3-bit input value, base 10, for bits FGH

An 8b/10b encoder is designed to maintain a neutral average disparity. Disparity is the difference between the number of 1s and 0s in the encoded word. Neutral disparity indicates the number of 1s and 0s are equal. Positive disparity indicates more 1s than 0s. Negative disparity indicates more 0s than 1s. The average disparity determines the DC component of the signals on the serial line. Running disparity is a record of the cumulative disparity of every encoded word, and is tracked by the encoder.

In order to maintain neutral disparity, two different codings are defined for each data value. The 8b/10b encoder in the transmit path selects between (+) and (-) encoded word based on calculated disparity of the present data to maintain neutral disparity

In the receive path, the clock and data recovery blocks retiming the incoming data and 8b/10b decoders generate 8-bit data based on the received 10-bit data. A sequence of valid 8b/10b coded characters has a maximum run length of 5 bits (i.e., 5 consecutive ones or 5 consecutive zeros before a mandatory bit transition). This assures adequate transitions for robust clock recovery.

The recovered data is aligned on a 10-bit boundaries by detecting and aligning to special characters in the incoming data stream. Data is word aligned using the comma (/K/) character. A comma character is a special character that contains a unique pattern (0011111 or its complement 1100000) in the 10-bit space that makes it useful for delimiting word boundaries. The special characters K28.1, K28.5 and K28.7 contain this comma sequence and are treated as valid comma characters by the SERDES.

The following table shows all of the valid special characters. All of the special characters are made available to the FPGA logic; however only the comma characters are used by the SERDES logic. The different codings that are possible for each data value are shown as encoded word (+) and encoded word (-). The table also illustrates the 8b/10b bit labeling convention. The bit positions of the 8-bit characters are labeled as H,G,F,E,D,C,B and A and the bit positions of the 10-bit encoded characters are labeled as a, b, c, d, e, i, f, g, h, and j. The encoded words are transmitted serially with bit 'a' transmitted first and bit 'j' transmitted last.

Table 2. Valid Special Characters

K Character	HGF EDCBA 765 43210	K Control	Encoded Word (-)	Encoded Word (+)
			abcdei fghj	abcdei fghj
K28.0	000 11100	1	001111 0100	110000 1011
K28.1 /comma/	001 11100	1	001111 1001	110000 0110
K28.2	010 11100	1	001111 0101	110000 1010
K28.3 /A/	011 11100	1	001111 0011	110000 1100
K28.4	100 11100	1	001111 0010	110000 1101
K28.5 /comma/	101 11100	1	001111 1010	110000 0101
K28.6	110 11100	1	001111 0110	110000 1001
K28.7 /comma/	111 11100	1	001111 1000	110000 0111
K23.7	111 10111	1	111010 1000	000101 0111
K27.7	111 11011	1	110110 1000	001001 0111
K29.7	111 11101	1	101110 1000	010001 0111
K30.7	111 11110	1	011110 1000	100001 0111

alignment will only occur when the communication channel is synchronized. When there is no synchronization of the link, the 1:4 DEMUX will continue to output 4-byte words at some arbitrary, but constant, boundary.

There are 2 control register bits available for each channel for word alignment. They are DOWDALGN_xx and NOWDALGN_xx. The DOWDALGN_xx bit is positive edge triggered. Writing a 0 followed by a 1 to this register bit will cause the corresponding DEMUX to look for a new comma character and align the 32-bit word such that the comma is in the most significant byte position. It is important that the comma is in the most significant byte position since the multi-channel aligner looks for comma in the most significant byte only.

Typically, it is not necessary to set the DOWDALGN_xx bit. When the link state machine loses synchronization (DEMUXWAS_xx register bit is 0), the DEMUX block automatically looks for a new comma character irrespective of whether the DOWDALGN_xx bit is set or not. However, as discussed earlier, the comma character may become misaligned without the Fibre Channel link state machine indicating a loss of synchronization. In such cases, the DOWDALGN_xx bit must be toggled to force resynchronization.

The NOWDALGN_xx bit is a level-sensitive bit. If it is a 1, then the DEMUX does not dynamically alter the word boundary based on comma and SWDSYNC_xx output of the SERDES. This might be useful if a channel were configured to bypass the multi-channel alignment FIFO and raw 40-bits of data are directed from SERDES to FPGA.

In Fibre Channel mode, the default setting (NOWDALGN_xx = 0) causes the word boundary to be set as soon as the SERDES SWDSYNC_xx output is a 1 and a comma character has been detected. The character that is the comma becomes the most-significant portion of the demultiplexed word. When the SERDES loses link synchronization it will drop SWDSYNC_xx low. The DEMUX will begin search for word alignment as soon as SWDSYNC_xx goes to 1 again.

The DEMUX passes on to the channel alignment FIFO block a set of control signals that indicate the location of the synchronizing event. RALIGN_xx[3:0] are these indicators. If there is no link synchronization, all of the RALIGN_xx[3:0] bits will be zeros independent of synchronizing events that come in. When the link is synchronized, then the bit that corresponds to the time of the synchronization event will be set to a 1.

The relationship between a time sequence of values input at SRBDxx[7:0] to the values output at RWD_xx[39:0] is shown in Figure 8. A parallel relationship exists between SRBDx[8] and RWBIT8_xx[3:0] as well as between SRBD_xx[9] and RWBIT9_xx[3:0].

- FMPU_SYNMODE_B = 11111111 (Register Location 30911)

To enable/disable multi-channel alignment of individual channels within a multi-channel alignment group:

- FMPU_STR_EN_xx = 1 enabled
- FMPU_STR_EN_xx = 0 disabled
- (Register Location 30810 and 30910, where xx is one of AC, AD, BC or BD.)

To resynchronize a multichannel alignment group set the following bit to zero, and then set it to one.

- FMPU_RESYNC4 for four channels, AC, AD, BC and BD. (Register Location 30A02, bit 2)
- FMPU_RESYNC2A for dual channels, AC and AD. (Register Location 30820, bit 5)
- FMPU_RESYNC2B for block channels, BC and BD. (Register Location 30920, bit 5)

To resynchronize an independent channel (resetting the write and the read pointer of the FIFO) set the following bit to zero, and then set it to one.

FMPU_RESYNC1_xx (Register Locations 30820 and 30920, bits 2 and 3, where xx is one of AC, AD, BC or BD).

ORT82G5 Configuration

Register settings for multi-channel alignment are shown in Table 7.

Table 7. Multi-channel Alignment Modes

Register Bits FMPU_SYNMODE_xx[0:1]	Mode
00	No multi-channel alignment.
10	Twin channel alignment.
01	Quad channel alignment.
11	Eight channel alignment.

Note: Where xx is one of A[A:D] and B[A:D].

To align all eight channels:

- FMPU_SYNMODE_A[A:D] = 11
- FMPU_SYNMODE_B[A:D] = 11

To align all four channels in SERDES A:

- FMPU_SYNMODE_A[A:D] = 01

To align two channels in SERDES A:

- FMPU_SYNMODE_A[A:B] = 10 for channel AA and AB
- FMPU_SYNMODE_A[C:D] = 10 for channel AC and AD

A similar alignment can be defined for SERDES B.

To enable/disable synchronization signal of individual channel within a multi-channel alignment group:

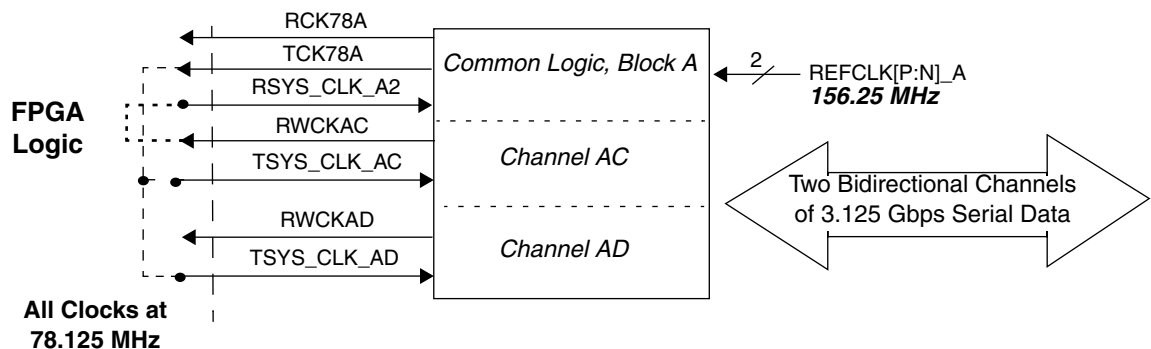
- FMPU_STR_EN_xx = 1 enabled
- FMPU_STR_EN_xx = 0 disabled

where xx is one of A[A:D] and B[A:D].

To resynchronize a multi-channel alignment group set the following bit to zero, and then set it to one:

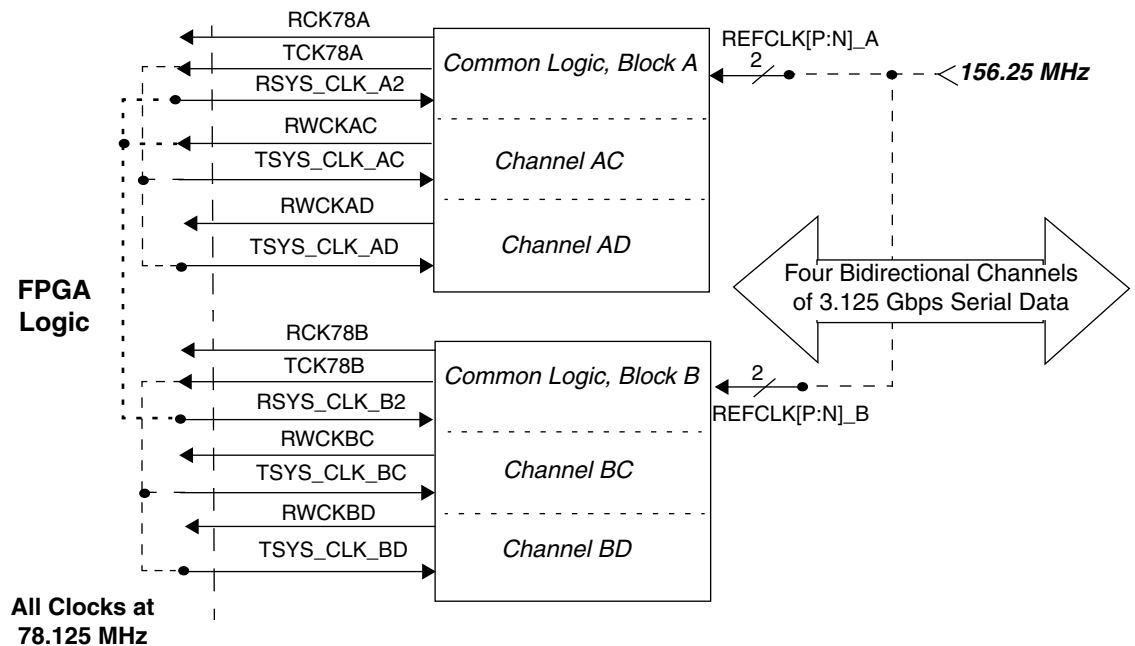
- FMPU_RESYNC8 for eight channel A[A:D] and B[A:D]
- FMPU_RESYNC4A for quad channel A[A:D]
- FMPU_RESYNC2A1 for twin channel A[A:B]

Figure 22. Receive Clocking for a Dual Alignment in a Single Block (Similar Connections Would Be Used for Block B)



For quad alignment, either RCK78A or RCK78B can be used to source RSYS_CLK_[A:B]2 as shown in Figure 23.

Figure 23. Clocking for Quad Alignment



As mentioned earlier, both sections of a slice can be written independently / simultaneously, due to the independent CSW per section.

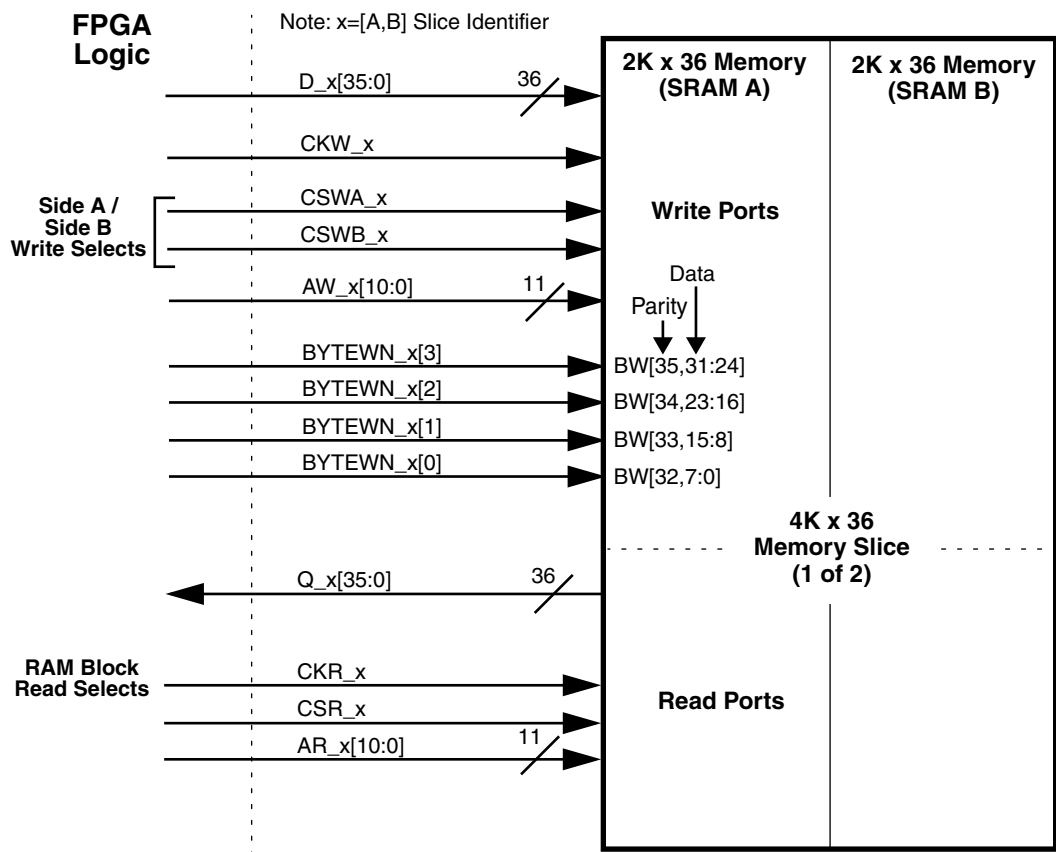
The same signal illustration above applies to slice B by changing `_A` to `_B`.

SDRAM A and SDRAM B in Figure 34 refer to the built-in sections A and B of one EAC RAM slice.

These SDRAMs should not be confused with the FPGA SDRAMs, which are generated through Module Generator in ispLEVER. The EAC SDRAMs are always built-in to the embedded core section of the ORT82G5/42G5 and their pins are accessed through the EAC interface. In order for these pins to be available at the interface in the generated HDL models from ispLEVER, the “Use the Extra Memory in FPSC Core” checkbox needs to be checked in the customization window (after hitting the "customize" button) in Module Generator, while generating the ORT82G5/42G5 core HDL. These signals will not otherwise show in the interface model.

Figure 35 and Figure 36 show, per slice, timing diagrams for both write and read accesses. These figures do not include the `_x` section, which refers to either slice A or B, even though this is implied. Signal names and functions are summarized in Table 26 and follow the general ORCA Series 4 naming conventions.

Figure 34. Block Diagram, Embedded Core Memory Slice



which follows the Power PC convention where address bit 0 is the MSb and address bit 31 is the LSb. The MPI maps bits MPI_ADDR[14:31] to bits [17:0] of the system address bus. The User Master Interface (UMI) has an 18-bit address bus and uses the opposite notation, where address line 17 is the MSb and address line 0 is the LSb. The UMI maps bits um_addr[17:0] to bits [17:0] of the system address bus. Because of the address mapping done by the MPI and UMI, the same hexadecimal address value is valid for both interfaces.

The UMI, internal and microprocessor interface data buses have both 32-bit data and 4-bit parity fields and the data fields are mapped 1:1 to each other, i.e., bit 0 is bit 0 for all three buses. The bit ordering is specific to the targeted functional block. In the memory map, only bits [0:7] are specified and the convention followed for sub-field descriptions is to map the bits in the description directly to the bit order given in the bit column. For example, to select channel C as the source for the transmit and receive clocks, the register at location 30A00 should have bits 0, 2, 4 and 6 set to zero and bits 1, 3, 5 and 7 set to one.

In the example in the previous paragraph, the bits being set are control bits and are independent of the MSb/LSb convention used. The resulting bit pattern 01010101 maps to the hexadecimal value AA if the left-most bit is considered the LSb and to 55 if the right-most bit is considered the LSb. In some cases, however, the data represents the value of a specific parameter, such as a size or threshold level, and the value may be stored at more than one address location, since each location can hold only 8 bits of data. For a given register, either the MSb or the LSb bit position is specified explicitly in the memory map. If the parameter value extends over multiple register locations, the relative bit or byte ordering is also specified. For additional information on the MPI and the system bus, see Technical Note TN1017, *ORCA Series 4 MPI/System Bus*.

Table 28. ORT42G5 Memory Map

(0x) Absolute Address	Bit	Name	Reset Value (0x)	Description
SERDES Alarm Registers (Read Only, Clear on Read), xx = [AC, AD, BC or BD]				
30020 - AC	[0]	Reserved	00	Reserved
30030 - AD	[1]	LKI_xx		Receive PLL Lock Indication, Channel xx. LKI_xx = 1 indicates the receive PLL is locked.
30120 - BC	[2]	Not used		Reserved
30130 - BD	[3]	Not used		Reserved
	[4:7]	Not used		Not used
SERDES Alarm Mask Registers (Read/Write), xx = [AC, AD, BC or BD]				
30021 - AC	[0]	Reserved	FF	Reserved, must be set to 1. Set to 1 on device reset.
30031 - AD	[1]	MLKI_xx		Mask Receive PLL Lock Indication, Channel xx.
30121 - BC	[2]	Reserved		Reserved. Must be set to 1. Set to 1 on device reset.
30131 - BD	[3]	Reserved		Reserved. Must be set to 1. Set to 1 on device reset.
	[4]	Reserved		Reserved. Must be set to 1. Set to 1 on device reset.
	[5]	Reserved		Reserved. Must be set to 1. Set to 1 on device reset.
	[6]	Reserved		Reserved. Must be set to 1. Set to 1 on device reset.
	[7]	Reserved		Reserved. Must be set to 1. Set to 1 on device reset.

Table 30. ORT82G5 Memory Map (Continued)

(0x) Absolute Address	Bit	Name	Reset Value (0x)	Description
30810 - Ax 30910 - Bx	[0]xA [1]xB [2]xC [3]xD	DOWDALIGN_xx	00	Word Realign Bit. When DOWDALIGN_xx transitions from 0 to 1, the receiver realigns on the next comma character for Channel xx. NOWDALIGN_xx=0 on device reset.
	[4]xA [5]xB [6]xC [7]xD	FMPU_STR_EN_xx		Enable multi-channel alignment for Channel xx. When FMPU_STR_EN_xx=1, the corresponding channel participates in multi-channel alignment. FMPU_STR_EN_xx=0 on device reset.
30811 - Ax 30911 - Bx	[0:1] xA [2:3] xB [4:5] xC [6:7] xD	FMPU_SYNMOD E_xx[0:1]	00	Sync mode for xx 00 = No channel alignment 10 = Twin channel alignment 01 = Quad channel alignment 11 = Eight channel alignment
30820 - Ax 30920 - Bx	[0]xA [1]xB [2]xC [3]xD	FMPU_RESYNC1_xx	00	Resync a Single Channel. When FMPU_RESYNC1_xx transitions from 0 to 1, the corresponding channel is resynchronized (the write and read pointers are reset). FMPU_STR_EN_xx=0 on device reset.
	[4] xA & xB [5] xC & xD	FMPU_RESYNC2_x[1:2]		Resync a Pair of Channels. When FMPU_RESYNC2_[A:B][1:2] transitions from a 0 to a 1, the corresponding channel pair is resynchronized. FFMPU_RESYNC2_[A:B][1:2]=0 on device reset.
	[6]	FMPU_RESYNC4[A:B]		Resync a Four-Channel Group. When FMPU_RESYNC4[A:B] transitions from a 0 to a 1, the corresponding four-channel group is resynchronized. FMPU_RESYNC4[A:B]=0 on device reset.
	[7]	XAUI_MODE[A:B]		Controls use of XAUI link state machine in place of Fibre-Channel state machine. When XAUI_MODE[A:B]=1, all four channels in the SERDES quad enable their XAUI link state machines. (LINKSM_xx bits are ignored). XAUI_MODE[A:B]=0 on device reset.
30821 - A 30921 - B	[0]	NOCHALGN [A:B]	00	Bypass channel alignment. NOCHALGN [A:B] =1 causes bypassing of multi-channel alignment FIFOs for the corresponding SERDES quad. NOCHALGN [A:B] =0 on device reset.
	[1:7]	Reserved for future use.		
30933	[0:3]	Reserved for future use.		
	[4:5]	SCHAR_CHAN[0:1]	00	Select channel to test 00 = Channel BA 10 = Channel BB 01 =Channel BC 11 = Channel BD
	[6]	SCHAR_TXSEL		1=Select TX option 0=Select RX option
	[7]	SCHAR_ENA		1=Enable Characterization of SERDES B
Status Registers (Read Only), xx=[AA,...,BD]				
30804 - Ax 30904 - Bx	[0:1] xA [2:3] xB [4:5] xC [6:7] xD	XAUISTAT_xx[0:1]	00	XAUI Status Register. Status of XAUI link state machine for Channel xx 00 – No synchronization. 10 – Synchronization done. 11 – Not used. 01 – no_comma (see XAUI state machine) and at least one CV detected XAUISTAT_xx[0:1] = 00 on device reset.

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. If any pin is not used (or not bonded to a package pin), it is also 3-stated with an internal pull-up resistor after configuration. The pin descriptions in Table and throughout this data sheet show active-low signals with an overscore. The package pinout tables that follow, show this as a signal ending with _N. For example $\overline{\text{LDC}}$ and LDC_N are equivalent.

Table 40. Pin Descriptions

Symbol	I/O	Description
Dedicated Pins		
VDD33	—	3.3V positive power supply. This power supply is used for 3.3V configuration RAMs and internal PLLs. When using PLLs, this power supply should be well isolated from all other power supplies on the board for proper operation.
VDD15	—	1.5V positive power supply for internal logic.
VDDIO	—	Positive power supply used by I/O banks.
VSS	—	Ground.
PTEMP	I	Temperature sensing diode pin. Dedicated input.
RESET	I	During configuration, $\overline{\text{RESET}}$ forces the restart of configuration and a pull-up is enabled. After configuration, 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	O	In the master and asynchronous peripheral modes, CCLK is an output which strobes configuration data in.
	I	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. ¹
	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.
PRGRM	I	PRGRM 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.
RD_CFG	I	This pin must be held high during device initialization until the INIT pin goes high. This pin always has an active pull-up. During configuration, RD_CFG is an active-low input that activates the TS_ALL function and 3-states all of the I/O. After configuration, 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 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.
$\overline{\text{CFG_IRQ}}$ /MPI_IRQ	O	During JTAG, slave, master, and asynchronous peripheral configuration assertion 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, when the MPI is used.
LVDS_R	—	Reference resistor connection for controlled impedance termination of Series 4 FPGA LVDS inputs.
Special-Purpose Pins		
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. ¹
PLL_CK[0:7][TC]	I	Semi-dedicated PLL clock pins. During configuration they are 3-stated with a pull up.
	I/O	These pins are user-programmable I/O pins if not used by PLLs after configuration.
P[TBLR]CLK[1:0][TC]	I	Pins dedicated for the primary clock. Input pins on the middle of each side with differential pairing.
	I/O	After configuration these pins are user programmable I/O, if not used for clock inputs.

- Example connections are shown in Figure 40. The naming convention for the power supply sources shown in the figure are as follows:
 - Supply_1.5V – Tx-Rx digital, auxiliary power pins.
 - Supply_VDDIB – Input Rx buffer power pins.
 - Supply_VDDOB – Output Tx buffer power pins.
 - Supply_VDDANA – Tx analog power pins, Rx analog power pins, guard band power pins.

Figure 40. Power Supply Filtering

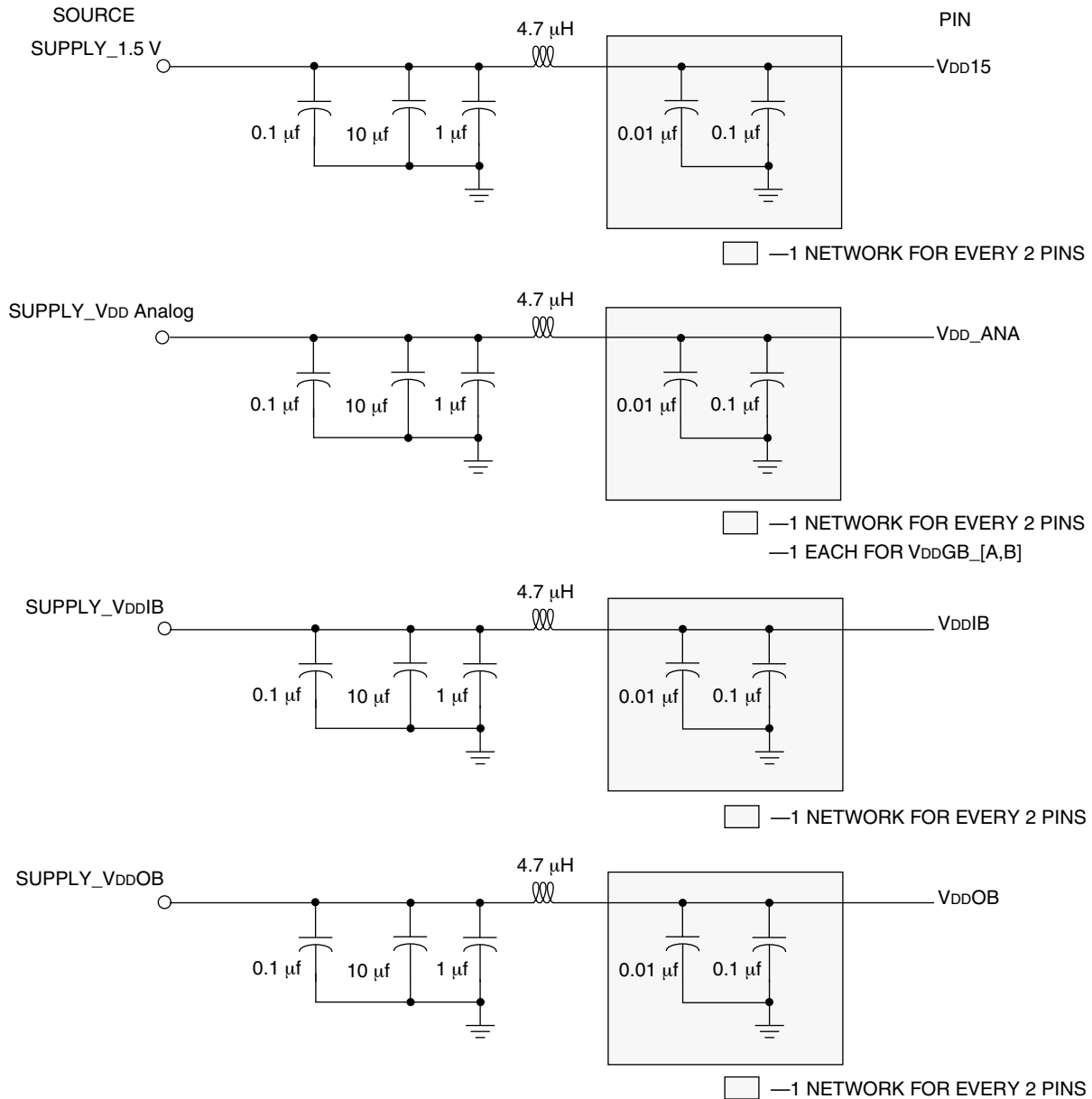


Table 44. ORT42G5 484-pin PBGAM (fpBGA) Pinout (Continued)

484-PBGAM	VDDIO Bank	VREF Group	I/O	Pin Description	Additional Function	484-PBGAM
L11	-	-	VSS	VSS	-	-
N15	-	-	VDD15	VDD15	-	-
D10	1 (TC)	5	IO	PT18D	PTCK1C	L68C
C10	1 (TC)	5	IO	PT18C	PTCK1T	L68T
A12	1 (TC)	5	IO	PT17D	PTCK0C	L69C
B12	1 (TC)	5	IO	PT17C	PTCK0T	L69T
P6	-	-	VDD15	VDD15	-	-
A11	1 (TC)	5	IO	PT16D	VREF_1_05	L70C
B11	1 (TC)	5	IO	PT16C	-	L70T
L12	-	-	VSS	VSS	-	-
D9	1 (TC)	6	IO	PT15D	-	L71C
C9	1 (TC)	6	IO	PT15C	-	L71T
G15	1 (TC)	-	VDDIO1	VDDIO1	-	-
B10	1 (TC)	6	IO	PT14D	-	L72C
A10	1 (TC)	6	IO	PT14C	VREF_1_06	L72T
B9	0 (TL)	1	IO	PT13D	MPI_RTRY_N	L73C
A9	0 (TL)	1	IO	PT13C	MPI_ACK_N	L73T
D8	0 (TL)	1	IO	PT12D	M0	L74C
C8	0 (TL)	1	IO	PT12C	M1	L74T
A22	-	-	VSS	VSS	-	-
B8	0 (TL)	2	IO	PT12B	MPI_CLK	L75C
A8	0 (TL)	2	IO	PT12A	A21/MPI_BURST_N	L75T
C7	0 (TL)	2	IO	PT11D	M2	L76C
D7	0 (TL)	2	IO	PT11C	M3	L76T
E9	0 (TL)	-	VDDIO0	VDDIO0	-	-
E6	0 (TL)	2	IO	PT11A	MPI_TEA_N	-
F6	-	-	VDD15	VDD15	-	-
B7	0 (TL)	3	IO	PT9D	VREF_0_03	L77C
A7	0 (TL)	3	IO	PT9C	-	L77T
A6	0 (TL)	3	IO	PT8D	D0	L78C
B6	0 (TL)	3	IO	PT8C	TMS	L78T
C6	0 (TL)	4	IO	PT7D	A20/MPI_BDIP_N	L79C
D6	0 (TL)	4	IO	PT7C	A19/MPI_TSZ1	L79T
B1	-	-	VSS	VSS	-	-
A5	0 (TL)	4	IO	PT6D	A18/MPI_TSZ0	L80C
B5	0 (TL)	4	IO	PT6C	D3	L80T
C5	0 (TL)	5	IO	PT5D	D1	L81C
D5	0 (TL)	5	IO	PT5C	D2	L81T
B2	-	-	VSS	VSS	-	-
A4	0 (TL)	5	IO	PT4D	TDI	L82C
B4	0 (TL)	5	IO	PT4C	TCK	L82T
E10	0 (TL)	-	VDDIO0	VDDIO0	-	-
B22	-	-	VSS	VSS	-	-
C4	0 (TL)	6	IO	PT2D	PLL_CK1C/PPLL	L83C

Table 45. ORT82G5 680-Pin PBGAM (fpBGA) Pinout (Continued)

680-PBGAM	VDDIO Bank	VREF Group	I/O	Pin Description	Additional Function	680-PBGAM
AA32	—	—	VDD_ANA	VDD_ANA	—	—
AF30	—	—	—	REXT_B	—	—
AF31	—	—	—	REXTN_B	—	—
AE30	—	—	I	REFCLKN_B	—	—
AE31	—	—	I	REFCLKP_B	—	—
AB32	—	—	VSS	VSS	—	—
AD30	—	—	VDDIB	VDDIB_BA	—	—
AD32	—	—	VDD_ANA	VDD_ANA	—	—
AF33	—	—	I	HDINN_BA	—	—
AC32	—	—	VSS	VSS	—	—
AF34	—	—	I	HDINP_BA	—	—
AE32	—	—	VDD_ANA	VDD_ANA	—	—
AD31	—	—	VSS	VSS	—	—
K32	—	—	VDD_ANA	VDD_ANA	—	—
AE33	—	—	O	HDOUTN_BA	—	—
AF32	—	—	VSS	VSS	—	—
AE34	—	—	O	HDOUTP_BA	—	—
AC30	—	—	VDDOB	VDDOB_BA	—	—
AG30	—	—	VSS	VSS	—	—
AB30	—	—	VDDIB	VDDIB_BB	—	—
AD33	—	—	I	HDINN_BB	—	—
AG31	—	—	VSS	VSS	—	—
AD34	—	—	I	HDINP_BB	—	—
AC31	—	—	VSS	VSS	—	—
AC33	—	—	O	HDOUTN_BB	—	—
AG32	—	—	VSS	VSS	—	—
AC34	—	—	O	HDOUTP_BB	—	—
AB31	—	—	VDDOB	VDDOB_BB	—	—
AG33	—	—	VSS	VSS	—	—
AA30	—	—	VDDIB	VDDIB_BC	—	—
AB33	—	—	I	HDINN_BC	—	—
AG34	—	—	VSS	VSS	—	—
AB34	—	—	I	HDINP_BC	—	—
AA31	—	—	VSS	VSS	—	—
Y30	—	—	VDDOB	VDDOB_BC	—	—
AA33	—	—	O	HDOUTN_BC	—	—
H30	—	—	VSS	VSS	—	—
AA34	—	—	O	HDOUTP_BC	—	—
Y31	—	—	VDDOB	VDDOB_BC	—	—
H31	—	—	VSS	VSS	—	—
W30	—	—	VDDIB	VDDIB_BD	—	—
Y33	—	—	I	HDINN_BD	—	—
H32	—	—	VSS	VSS	—	—
Y34	—	—	I	HDINP_BD	—	—

Table 45. ORT82G5 680-Pin PBGAM (fpBGA) Pinout (Continued)

680-PBGAM	VDDIO Bank	VREF Group	I/O	Pin Description	Additional Function	680-PBGAM
W31	—	—	VSS	VSS	—	—
V30	—	—	VDDOB	VDDOB_BD	—	—
W33	—	—	O	HDOUTN_BD	—	—
H33	—	—	VSS	VSS	—	—
W34	—	—	O	HDOUTP_BD	—	—
V31	—	—	VDDOB	VDDOB_BD	—	—
H34	—	—	VSS	VSS	—	—
J32	—	—	VSS	VSS	—	—
U31	—	—	VDDOB	VDDOB_AD	—	—
T34	—	—	O	HDOUTP_AD	—	—
M32	—	—	VSS	VSS	—	—
T33	—	—	O	HDOUTN_AD	—	—
U30	—	—	VDDOB	VDDOB_AD	—	—
T31	—	—	VSS	VSS	—	—
R34	—	—	I	HDINP_AD	—	—
N32	—	—	VSS	VSS	—	—
R33	—	—	I	HDINN_AD	—	—
T30	—	—	VDDIB	VDDIB_AD	—	—
U32	—	—	VSS	VSS	—	—
R31	—	—	VDDOB	VDDOB_AC	—	—
P34	—	—	O	HDOUTP_AC	—	—
U33	—	—	VSS	VSS	—	—
P33	—	—	O	HDOUTN_AC	—	—
R30	—	—	VDDOB	VDDOB_AC	—	—
P31	—	—	VSS	VSS	—	—
N34	—	—	I	HDINP_AC	—	—
U34	—	—	VSS	VSS	—	—
N33	—	—	I	HDINN_AC	—	—
P30	—	—	VDDIB	VDDIB_AC	—	—
V32	—	—	VSS	VSS	—	—
M34	—	—	O	HDOUTP_AB	—	—
V33	—	—	VSS	VSS	—	—
M33	—	—	O	HDOUTN_AB	—	—
N31	—	—	VDDOB	VDDOB_AB	—	—
M31	—	—	VSS	VSS	—	—
L34	—	—	I	HDINP_AB	—	—
V34	—	—	VSS	VSS	—	—
L33	—	—	I	HDINN_AB	—	—
N30	—	—	VDDIB	VDDIB_AB	—	—
K34	—	—	O	HDOUTP_AA	—	—
K33	—	—	O	HDOUTN_AA	—	—
M30	—	—	VDDOB	VDDOB_AA	—	—
L32	—	—	VDD_ANA	VDD_ANA	—	—
L31	—	—	VSS	VSS	—	—

Table 45. ORT82G5 680-Pin PBGA (fpBGA) Pinout (Continued)

680-PBGAM	VDDIO Bank	VREF Group	I/O	Pin Description	Additional Function	680-PBGAM
B29	1 (TC)	8	IO	PT33D	—	L1C_A0
C29	1 (TC)	8	IO	PT33C	VREF_1_08	L1T_A0
B15	1 (TC)	—	VDDIO1	VDDIO1	—	—
E27	1 (TC)	8	IO	PT32D	—	L2C_A0
E26	1 (TC)	8	IO	PT32C	—	L2T_A0
AP34	—	—	Vss	Vss	—	—
A30	1 (TC)	8	IO	PT32B	—	—
A29	1 (TC)	9	IO	PT31D	—	L3C_D3
E25	1 (TC)	9	IO	PT31C	VREF_1_09	L3T_D3
B17	1 (TC)	—	VDDIO1	VDDIO1	—	—
E24	1 (TC)	9	IO	PT31A	—	—
B28	1 (TC)	9	IO	PT30D	—	L4C_A0
C28	1 (TC)	9	IO	PT30C	—	L4T_A0
B2	—	—	Vss	Vss	—	—
D28	1 (TC)	9	IO	PT30A	—	—
C27	1 (TC)	9	IO	PT29D	—	L5C_A0
D27	1 (TC)	9	IO	PT29C	—	L5T_A0
E23	1 (TC)	9	IO	PT29B	—	L6C_A0
E22	1 (TC)	9	IO	PT29A	—	L6T_A0
D26	1 (TC)	1	IO	PT28D	—	L7C_A0
D25	1 (TC)	1	IO	PT28C	—	L7T_A0
B33	—	—	Vss	Vss	—	—
D24	1 (TC)	1	IO	PT28B	—	L8C_A0
D23	1 (TC)	1	IO	PT28A	—	L8T_A0
C26	1 (TC)	1	IO	PT27D	VREF_1_01	L9C_A0
C25	1 (TC)	1	IO	PT27C	—	L9T_A0
D11	1 (TC)	—	VDDIO1	VDDIO1	—	—
E21	1 (TC)	1	IO	PT27B	—	L10C_A0
E20	1 (TC)	1	IO	PT27A	—	L10T_A0
D22	1 (TC)	2	IO	PT26D	—	L11C_A0
D21	1 (TC)	2	IO	PT26C	VREF_1_02	L11T_A0
E34	—	—	Vss	Vss	—	—
A28	1 (TC)	2	IO	PT26B	—	—
B26	1 (TC)	2	IO	PT25D	—	L12C_A0
B25	1 (TC)	2	IO	PT25C	—	L12T_A0
D13	1 (TC)	—	VDDIO1	VDDIO1	—	—
B27	1 (TC)	2	IO	PT25B	—	—
A27	1 (TC)	3	IO	PT24D	—	L13C_A0
A26	1 (TC)	3	IO	PT24C	VREF_1_03	L13T_A0
N13	—	—	Vss	Vss	—	—
C24	1 (TC)	3	IO	PT24B	—	—
C22	1 (TC)	3	IO	PT23D	—	L14C_A0
C23	1 (TC)	3	IO	PT23C	—	L14T_A0
D15	1 (TC)	—	VDDIO1	VDDIO1	—	—

Table 46. Heat Sink Vendors

Vendor	Location	Phone
Aavid Thermalloy	Concord, NH	(603) 224-9988
Chip Coolers	Warwick, RI	(800) 227-0254
IERC	Burbank, CA	(818) 842-7277
R-Theta	Buffalo, NY	(800) 388-5428
Sanyo Denki	Torrance, CA	(310) 783-5400
Wakefield Thermal Solutions	Pelham, NH	(800) 325-1426

Package Parasitics

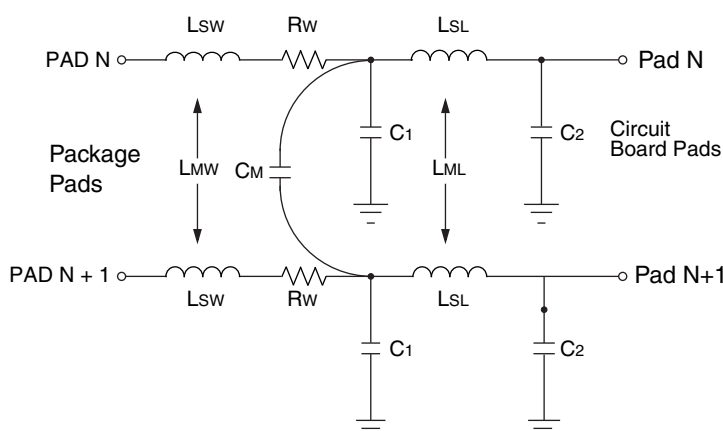
The electrical performance of an IC package, such as signal quality and noise sensitivity, is directly affected by the package parasitics. Table 47 lists eight parasitics associated with the ORCA packages. These parasitics represent the contributions of all components of a package, which include the bond wires, all internal package routing, and the external leads.

Four inductances in nH are listed: LSW and LSL, the self-inductance of the lead; and LMW and LML, 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: CM, the mutual capacitance of the lead to the nearest neighbor lead; and C1 and C2, 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 47 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 C1 and C2 capacitors.

Table 47. ORCA Typical Package Parasitics

LSW	LMW	RW	C1	C2	CM	LSL	LML
3.8	1.3	250	1.0	1.0	0.3	2.8-5	0.5 -1

Figure 41. Package Parasitics

Package Outline Drawings

Package Outline Drawings for the 484-ball PBGA (fpBGA) used for the ORT42G5 and 680-ball PBGA (fpBGA) used for the ORT82G5 are available in the Package Diagrams section of the Lattice Semiconductor web site at www.latticesemi.com.