Welcome to E-XFL.COM

Understanding Embedded - FPGAs (Field Programmable Gate Array)

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

Applications of Embedded - FPGAs

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

Details	
Product Status	Obsolete
Number of LABs/CLBs	-
Number of Logic Elements/Cells	10368
Total RAM Bits	113664
Number of I/O	372
Number of Gates	643000
Voltage - Supply	1.425V ~ 3.6V
Mounting Type	Surface Mount
Operating Temperature	-40°C ~ 85°C (TA)
Package / Case	680-BBGA
Supplier Device Package	680-FPBGA (35x35)
Purchase URL	https://www.e-xfl.com/product-detail/lattice-semiconductor/ort82g5-2fn680i

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

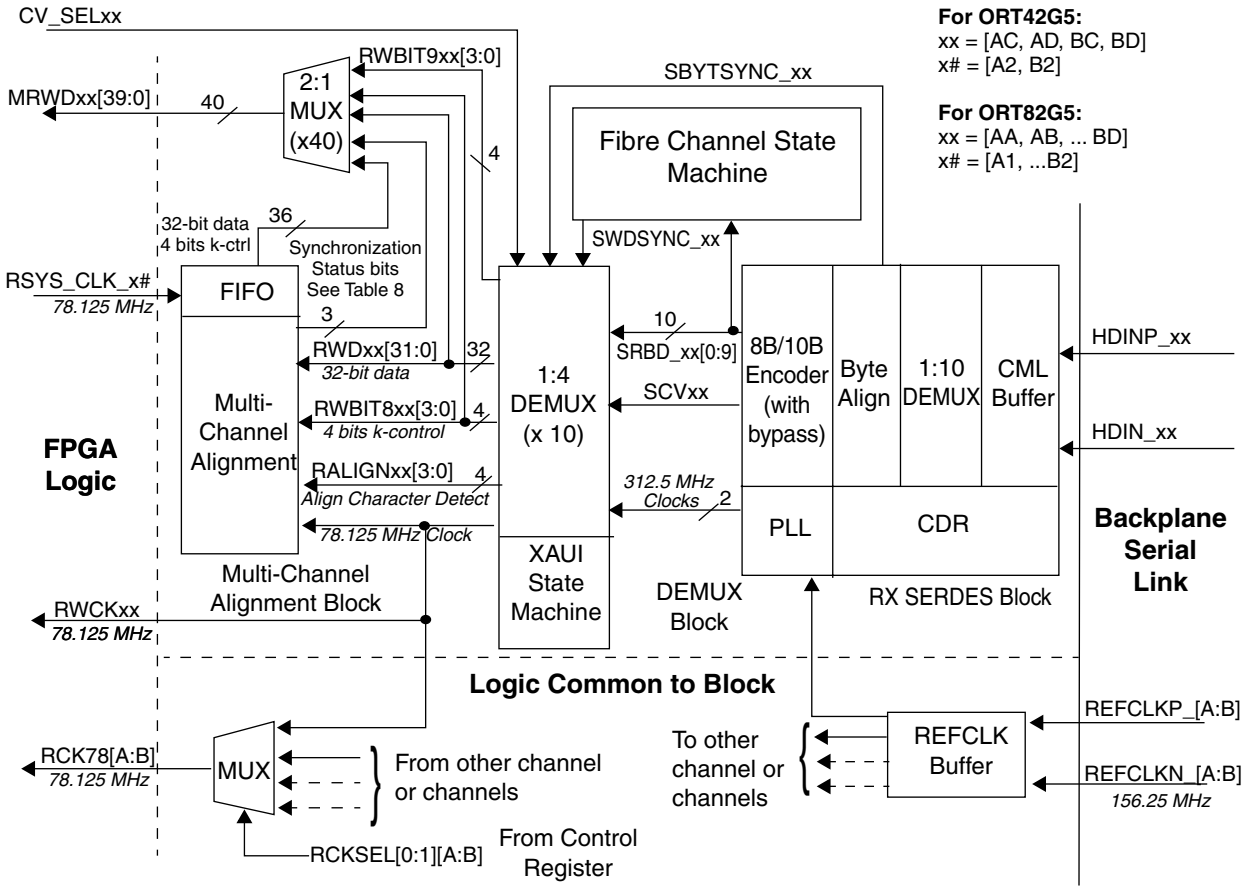
The following sections describe the various logic blocks in the Embedded Core portion of the FPSC. The FPGA section of the FPSC is identical to an ORCA OR4E04 FPGA except that the pads on one edge of the FPGA chip are replaced by the Embedded Core. For a detailed description of the programmable logic 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.

Figure 6. Basic Logic Blocks, Receive Path, Single Channel (Typical Reference Clock Frequency)



Each channel provides its own received clock, received data and K-character detect signals to the FPGA logic. Incoming data from multiple channels can be aligned using comma (/K/) characters or /A/ character (as specified either in Fibre Channel specifications or in IEEE 802.3ae for XAUI based interfaces). 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 this 8b/10b bypass mode.

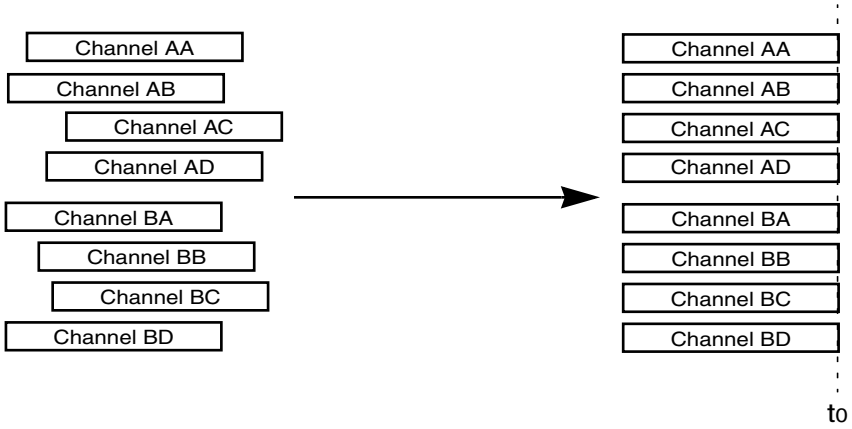
Detailed descriptions of data synchronization, of the SERDES, DEMUX and Multi-Channel Alignment blocks and of the Fibre Channel and XAUI state machines are given in following sections. Receive clock distribution is described in a later section of this data sheet.

Synchronization

The SERDES RX logic performs four levels of synchronization on the incoming serial data stream. Each level builds upon the previous, providing first bit, then byte (character), then channel (32-bit word), and finally multi-channel alignment. Each step is described functionally in the following paragraphs. The details of the logical implementations are described in subsequent sections.

Bit alignment is the task of the Clock/Data Recovery (CDR) block. This block utilizes a PLL that locks to the transitions in the incoming high-speed serial data stream, and outputs the extracted clock as well as the data. If the PLL is unable to lock to the serial data stream, it instead locks to REFCLK[A:B] to stabilize the voltage-controlled oscillator (VCO), and periodically switches back to the serial data stream to again attempt synchronization. This process continues until a valid input data stream is detected and lock is achieved. The CDR can maintain lock on data as long as the input data stream contains an adequate data “eye” (i.e., jitter is within specification) and the maximum data stream run length is not exceeded.

Figure 15. Alignment of all Eight SERDES Channels.



Note that any channel within an alignment group can be removed from that alignment group by setting FMPU_STR_EN_xx to 0. The disabling of any channel(s) within an alignment group will not affect the operation of the remaining active channels. If the active channels are synchronized, that synchronization will be maintained and no data loss will occur.

For every alignment group, there are both an OVFL and an OOS status register bit. The OVFL bit is set when alignment FIFO overflow occurs. The OOS bit is flagged when the down counter in the synchronization algorithm has reached a value of 0 and alignment characters from all channels within an alignment group have not been received. In the memory map section for the ORT42G5 the bits indicating OOS and OVFL are referred to as SYNC2_[A:B]_OOS and SYNC4_OOS and the bits indicating OVFL are SYNC2_[A:B]_OVFL and SYNC4_OVFL.

In the memory map section for the ORT82G5, the bits indicating OOS and OVFL are referred to as SYNC2_[A1,A2,B1,B2]_OOS, SYNC4_[A:B]_OOS and SYNC8_OOS and the bits indicating OVFL are SYNC2_[A1,A2,B1,B2]_OVFL, SYNC4_[A:B]_OVFL and SYNC8_OVFL.

Alignment can also be done between the receive channels on two ORT82G5 devices. Each of the two devices needs to provide its aligned K_CTRL or other alignment character to the other device, which will delay reading from a second alignment FIFO until all channels requesting alignment on the current device and all channels requesting alignment on the other device are aligned (as indicated on the K_CTRL character). These second alignment FIFOs will be implemented in FPGA logic on the ORT82G5. This scheme also requires that the reference clock for both devices be driven by the same signal.

XAUI Lane Alignment Function (Lane Deskew)

In XAUI mode, the receive section in each lane uses the /A/ code group to compensate for lane-to-lane skew. The mechanism restores the timing relationship between the 4 lanes by lining up the /A/ characters into a column. Figure 16 shows the alignment of four lanes based on /A/ character. A minimum spacing of 16 code-groups implies that at least ± 80 bits of skew compensation capability should be provided, which the devices significantly exceed.

- 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]

Table 10. Definition of Status Bits of MRWDxx that Vary for Different Channels for the ORT82G5 (Continued)

Channel Index	Bit Index	Name	Description
BA	29	CV_BA_OR	Code violation in one or more of the received 10-bit groups for channel BA
BA	19	SYNC2_B1_OOS	Dual channel synchronization of channels BA and BB not successful if 1
BB	29	CV_BB_OR	Code violation in one or more of the received 10-bit groups for channel BB
BB	19	SYNC4_B_OOS	Quad channel synchronization of SERDES quad B not successful if 1
BC	29	CV_BC_OR	Code violation in one or more of the received 10-bit groups for channel BC
BC	19	SYNC2_B2_OOS	Dual channel synchronization of channels BC and BD not successful if 1
BD	29	CV_BD_OR	Code violation in one or more of the received 10-bit groups for channel BD
BD	19	SYNC8_OOS	Eight channel synchronization not successful if 1

For the ORT82G5, the SYNC2_[A1,A2,B1,B2]_OOS, SYNC4_[A:B]_OOS,and SYNC8_OOS signals can be used with CH248_SYNC_xx to determine if the desired multi-channel alignment was successful. If, when CH248_SYNC_xx goes high the corresponding OOS signal remains low, the data being transferred across the core/FPGA interface is correctly aligned between channels. Note that only the signals corresponding to the selected alignment mode will be meaningful.

For both devices, the code violation signals will only be valid if the corresponding CV_SELxx = 1. (If 8b10bR=0, CV_SEL should also be zero. The CV_xx_OR signals are obtained by ORing four code violation signals from the 1:4 DEMUX block. These are primarily indicators of received signal quality since a single code violation will not force a loss of sync (LOS) state in the word alignment state machines. Since these signals come from the DEMUX block, if multi-channel alignment is enabled, the code violation signals correspond to data that must still be multi-channel aligned. Hence these signals provide advance notification of detected violations in data that will appear at the core/FPGA interface several clock cycles later. The exact number of clock cycles that the data is delayed depends on the skew between the incoming data for the different channels.

Transceiver FPGA/Embedded Core Signals

Table 12 summarizes the interface signals between the FPGA logic and the core. In the table, an input refers to a signal flowing into the embedded core and an output refers to a signal flowing out of the embedded core.

Table 11. Transceiver Embedded Core/FPGA Interface Signal Description for the ORT42G5

FPGA/Embedded Core Interface Signal Name (xx = [AC, AD, BC or BD])	Input (I) to or Output (O) from Core	Signal Description
Transmit Path Signals		
TWDxx[31:0]	I	Transmit data – channel xx.
TCOMMAxx[3:0]	I	Transmit comma character – channel xx.
TBIT9xx[3:0]	I	Transmit force negative disparity – channel xx
TSYS_CLK_xx	I	Transmit low-speed clock to the FPGA – channel xx
TCK78[A:B]	O	Transmit low-speed clock to the FPGA – SERDES Quad [A:B].
Receive Path Signals		
MRWDxx[39:0]	O	Receive data – Channel xx (see Table 8 and Table 9).
RWCKxx	O	Low-speed receive clock—Channel xx.
RCK78[A:B]	O	Receive low-speed clock to FPGA—SERDES Quad [A:B].
RSYS_CLK_A2	I	Low-speed receive FIFO clock for channels AC, AD
RSYS_CLK_B2	I	Low-speed receive FIFO clock for channels BC, BD
CV_SELxx	I	Enable detection of code violations in the incoming data
SYS_RST_N	I	Synchronous reset of the channel alignment blocks.

Table 12. Transceiver Embedded Core/FPGA Interface Signal Description for the ORT82G5

FPGA/Embedded Core Interface Signal Name xx=... line remain (xx = [AA, ..., BD]	Input (I) to or Output (O) from Core	Signal Description
Transmit Path Signals		
TWDxx[31:0]	I	Transmit data – channel xx.
TCOMMAxx[3:0]	I	Transmit comma character – channel xx.
TBIT9xx[3:0]	I	Transmit force negative disparity – channel xx
TSYS_CLK_xx	I	Transmit low-speed clock to the FPGA – channel xx
TCK78[A:B]	O	Transmit low-speed clock to the FPGA – SERDES Quad [A:B].
Receive Path Signals		
MRWDxx[39:0]	O	Receive data – Channel xx (see Table 8 and Table).
RWCKxx	O	Low-speed receive clock—Channel xx.
RCK78[A:B]	O	Receive low-speed clock to FPGA—SERDES Quad [A:B].
RSYS_CLK_A1	I	Low-speed receive FIFO clock for channels AA, AB
RSYS_CLK_A2	I	Low-speed receive FIFO clock for channels AC, AD
RSYS_CLK_B1	I	Low-speed receive FIFO clock for channels BA, BB
RSYS_CLK_B2	I	Low-speed receive FIFO clock for channels BC, BD
CV_SELxx	I	Enable detection of code violations in the incoming data
SYS_RST_N	I	Synchronous reset of the channel alignment blocks.

Reference Clocks and Internal Clock Distribution

Reference Clock Requirements

There are two pairs of reference clock inputs on the ORT42G5 and ORT82G5. The differential reference clock is distributed to all channels in a block. Each channel has a differential buffer to isolate the clock from the other channels. The input clock is preferably a differential signal; however, the device can operate with a single-ended input. The input reference clock directly impacts the transmit data eye, so the clock should have low jitter. In particular, jitter components in the DC to 5 MHz range should be minimized. The required electrical characteristics for the reference clock are given in Table 38.

Note: In sections of this data sheet, the differential clocks are simply referred to as the reference clock as REFCLK_[A:B].

Synthesized and Recovered Clocks

The SERDES Embedded Core block contains its own dedicated PLLs for transmit and receive clock generation. The user provides a reference clock of the appropriate frequency, as described in the previous section. The transmitter PLL uses the REFCLK_[A,B] inputs to synthesize the internal high-speed serial bit clocks. The receiver PLLs extract the clock from the serial input data and retime the data with the recovered clock.

The receive PLL for each channel has two modes of operation - lock to reference and lock to data with retiming. When no data or invalid data is present on the HDINP_xx and HDINN_xx pins, the receive VCO will not lock to data and its frequency can drift outside of the nominal ±350 ppm range. Under this condition, the receive PLL will lock to REFCLK_[A,B] for a fixed time interval and then will attempt to lock to receive data. The process of attempting to lock to data, then locking to clock will repeat until valid input data exists. There is also a control register bit per channel to force the receive PLL to always lock to the reference clock.

The high-speed transmit and receive serial data links can run at 0.6 to 3.7 Gbps, depending on the frequency of the reference clock and the state of the control bits from the internal transmit control register. The interface to the serializer/deserializer block runs at 1/10th the bit rate of the data lane. Additionally, the MUX/DEMUX logic converts the

TCK78[A:B]:
This is a muxed output from the core to the FPGA across the core-FPGA interface of one of the 2 transmit SERDES clocks per block operating at up to 92.5 MHz in the embedded core. There is one clock output per SERDES block.

TSYS_CLK[AC, AD, BC, BD]:
These clocks are inputs to the SERDES block A and B respectively from the FPGA. These are used by each channel to control the timing of the Transmit Data Path. To guarantee correct transmit operation theses clocks must be frequency locked within 0 ppm to TCK78[A:B].

Transmit and Receive Clock Rates

Table 13 shows typical relationship between the data rates, the reference clock, the transmit TCK78[A:B] clock and the receive RCK78[A:B] clock. The selection of full-rate or half-rate for a given reference clock speed is set by bits in the transmit and receive control registers and can be set per channel.

Table 13. Transmit Data and Clock Rates

Data Rate	Reference Clock	TCK78[A: B] and RCK78[A:B] Clocks	Rate of Channel Selected as Clock Source
0.6 Gbps	60 MHz	15 MHz	Half
1.0 Gbps	100 MHz	25 MHz	Half
1.25 Gbps	125 MHz	31.25 MHz	Half
2.0 Gbps	100 MHz	50 MHz	Full
2.5 Gbps	125 MHz	62.5 MHz	Full
3.125 Gbps	156 MHz	78 MHz	Full
3.7 Gbps	185 MHz	92.5 MHz	Full

Besides taking in a TSYS_CLK_xx from the FPGA logic for each channel, the transmit path logic sends back a clock of the same frequency, but arbitrary phase. This clock, TCK78[A:B], is derived from the MUX block of one of the 2 channels in its SERDES block. The MUX blocks provide the potential source for TCK78[A:B] by a divide-by-4 of the SERDES STBC311xs clock used in synchronizing the transmit data words in the STBC311xx clock domain. The STBC311xx clocks are internal to the core and are not brought across the core/FPGA interface

The receiver section receives high-speed serial data at its differential CML input port and sends in to the Clock and Data Recovery (CDR) block. The CDR block then generates a recovered clock (RWCKxx) and retimes the data. Thus, the recovered receive clocks are asynchronous between channels.

Transmit Clock Source Selection

The TCKSEL[A:B] bit select the source channel of TCK78[A:B]. The selection of the source for TCK78[A:B] is controlled by this bit as shown in Table 14.

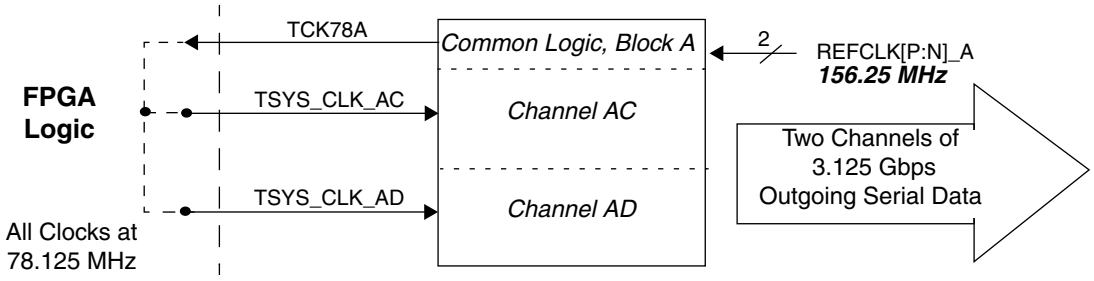
Table 14. TCK78[A:B] Source Selection

TCKSEL[A:B]	Clock Source
0	Channel C
1	Channel D

Recommended Transmit Clock Distribution for the ORT42G5

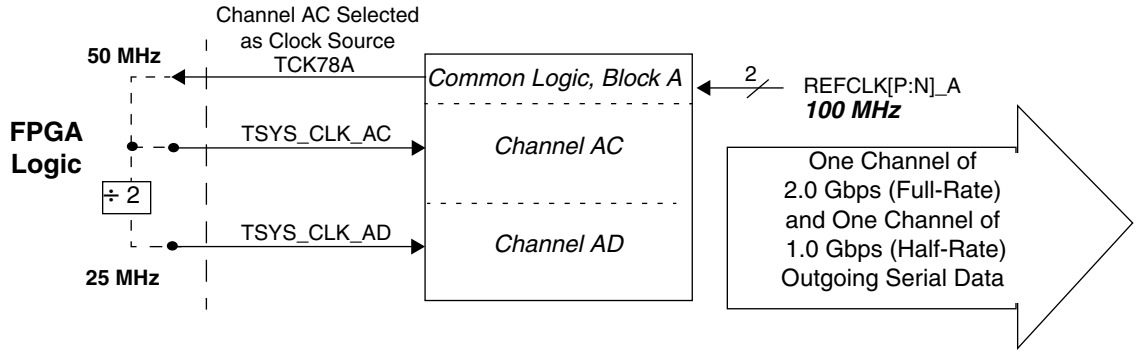
As an example of the recommended clock distribution approach, TSYS_CLK_A[C or D] can be sourced by TCK78A as shown in Figure 18 if the transmit line rate are common for both channels in a block. Similar clocking would be used for Block B.

Figure 18. Transmit Clocking for a Single Block (Similar Connections Would Be Used for Block B)



If the transmit line rate is mixed between half and full rate among the channels, then the scheme shown in Figure 19 can be used. The figure shows TSYS_CLK_AC being sourced by TCK78A and TSYS_CLK_AD being sourced by TCK78A/2 (the division is done in FPGA logic). Similar clocking would be used for Block B.

Figure 19. Mixed Rate Transmit Clocking for a Single Block (Similar Connections Would Be Used for Block B)



Receive Clock Source Selection and Recommended Clock Distribution

In the receive path, one clock per block of two channels, called RCK78[A:B], is sent to the FPGA logic. The control register bits RCKSEL[A:B] is used to select the clock source for these clocks. The selection of the source for RCK78[A:B] is controlled by this bit as shown in Table 15.

Table 15. RCK78[A:B] Source Selection

RCKSEL[A:B]	Clock Source
0	Channel C
1	Channel D

In the receive channel alignment bypass mode the data and recovered clocks for the four channels are independent. The data for each channel are synchronized to the recovered clock from that channel.

Figure 21 shows the recommended receive clocking for a single block.

Start Up Sequence for the ORT42G5

The following sequence is required by the ORT42G5 device. For information required for simulation that may be different than this sequence, see the ORT42G5 Design Kit.

1. Initiate a hardware reset by making PASB_RESETN low. Keep this low during FPGA configuration of the device. The device will be ready for operation 3 ms after the low to high transition of PASB_RESETN.

2. At startup, the legacy SERDES channel logic must be powered down and removed from any multi-channel alignment groups:

   - Setting bit 1 to one in registers at locations 30002, 30012, 30102, 30112, 30003, 30013, 30103 and 30113 powers down the legacy logic. (Note that the reset value for these bits is 0.)
   - Setting bits 4 and 5 to zero (reset condition) in the register at locations 30810 and 30910 removes the legacy logic from any alignment group.

3. Configure the following SERDES internal and external registers. Note that after device initialization, all alarm and status bits should be read once to clear them. A subsequent read will provide the valid state.

   Set the following bits in register 30800:
   – Bits LCKREFN_[AC and AD] to 1, which implies lock to data.
   – Bits ENBYSYNC_[AC and AD] to 1 which enables dynamic alignment to comma.

   Set the following bits in register 30801:
   – Bits LOOPENB_[AC and AD] to 1 if high-speed serial loopback is desired.

   Set the following bits in register 30900:
   – Bits LCKREFN_[BC and BD] to 1 which implies lock to data.
   – Bits ENBYSYNC_[BC and BD] to 1 which enables dynamic alignment to comma.

   Set the following bits in register 30901:
   – Bits LOOPENB_[BC and BD] to 1 if high-speed serial loopback is desired.

   Set the following bits in registers 30022, 30032, 30122, 30132:
   – TXHR set to 1 if TX half-rate is desired.
   – 8b10bT set to 1 if 8b10b encoding is desired.

   Set the following bits in registers 30023, 30033, 30123, 30133:
   – RXHR Set to 1 if RX half-rate is desired.
   – 8b10bR set to 1 if 8b10b decoding is desired.
   – LINKSM set to 1 if the Fibre Channel state machine is desired.

   Assert GSWRST bit by writing 1's to both SERDES blocks. Deassert GSWRST bit by writing 0's to both SERDES blocks. Wait 3 ms. If higher speed serial loopback has been selected, the receive PLLs will use this time to lock to the new serial data.

   Monitor the following alarm bits in registers 30020, 30030, 30120, 30130:
   – LKI, PLL lock indicator. 1 indicates that PLL has achieved lock.

4. If 8b/10b mode is enabled, enable link synchronization by periodically sending the following sequence three times:
   – K28.5 D21.4 D21.5 D21.5 or any other idle ordered set (starting with a /comma/) in FC mode.
   – /comma/ characters for the XAUI state machine and /A/ characters for word and channel alignment in XAUI mode.

Test Modes

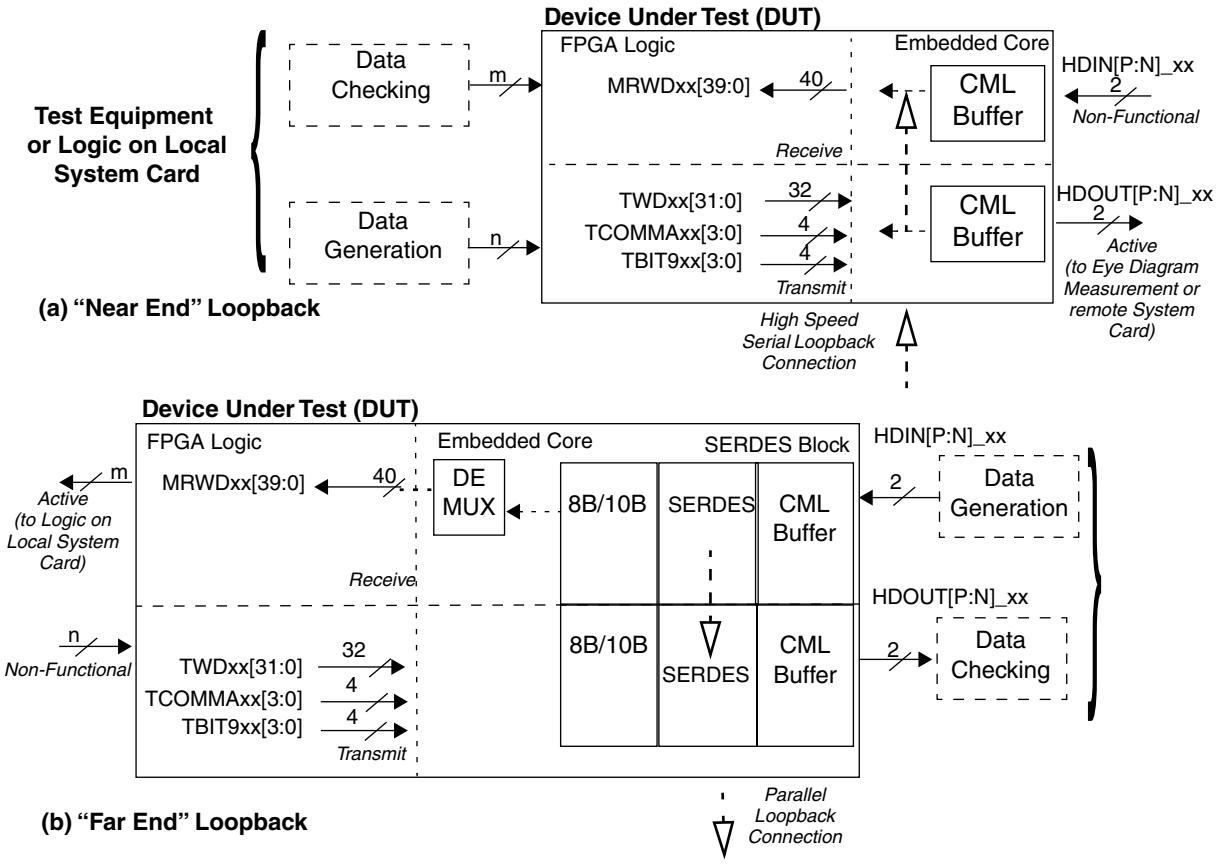
In addition to the operational logic described in the preceding sections, the Embedded Core contains logic to support various test modes - both for device validation and evaluation and for operating system level tests. The following sections discuss two of the test support logic blocks, supporting various loopback modes and SERDES characterization.

Loopback Testing

Loopback testing is performed by looping back (either internal to the Embedded Core, by configuring the FPGA logic or by external connections) transmitted data to the corresponding receiver inputs, or received data to the transmitter output. The loopback path may be either serial or parallel.

In general, loopback tests can be classified as “near end” or “far end.” In “near end” loopback (Figure 32(a)), data is generated and checked locally, i.e. by logic on, or connection of, test equipment to the same card as the FPSC. In “far end” loopback (Figure 32(b)), the generating and checking functions are performed remotely, either by test equipment or a remote system card.

Figure 32. “Near End” vs. “Far End” Loopback



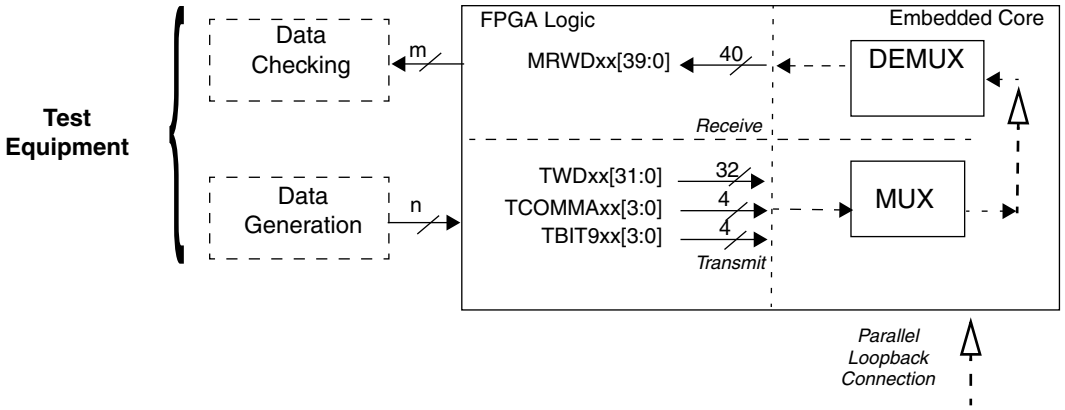
The loopback mode can also be characterized by the physical location of the loopback connection. There are three possible loopback modes supported by the Embedded Core logic:

- High-speed serial loopback at the CML buffer interface (near end)
- Parallel loopback at the SERDES boundary (far end)

Parallel Loopback at MUX/DEMUX Boundary, Excluding SERDES

This is a low-frequency test mode used to test the MUX/DEMUX logic block. As with the mode described in the previous section, the loopback path is at the interface between the SERDES blocks and the MUX and DEMUX blocks and uses the parallel 10-bit buses at these interfaces (see Figure 33). However, the loopback connection is made such that the output signals from the TX MUX block are used as the input signals to the RX SERDES block. In this loopback mode the MRWDxx[39:0], TWDxx[31:0], TCOMMAxx[3:0] and TBIT9xx[3:0] signal lines function normally and the high-speed serial input and output buffers are not used. Use of this mode also requires configuration of the FPGA logic to connect the MRWDxx[39:0], TWDxx[31:0], TCOMMAxx[3:0] and TBIT9xx[3:0] signal lines to external pins. The basic loopback path is shown in Figure 33.

Figure 33. Parallel Loopback at MUX/DEMUX Boundary, Excluding SERDES



This test mode is enabled by setting the pin PLOOP_TEST_ENN to 0. PASB_TESTCLK must be running in this mode at 4x frequency of RSYS_CLK[A2, B2] or TSYS_CLK_[AC, AD, BC, BD] for the ORT42G5 and RSYS_CLK[A1,A2,B1,B2] or TSYS_CLK_[AA, AB... BD] for the ORT82G5.

SERDES Characterization Test Mode (ORT82G5 Only)

The SERDES characterization mode is a test mode that allows for direct control and observation of the transmit and receive SERDES interfaces at chip ports. With these modes the SERDES logic and I/O can be tested one channel at a time in either the receive or transmit modes. The SERDES characterization mode is available for only one quad (quad B) of the ORT82G5.

The characterization test mode is configured by setting bits in the control registers via the system bus. There are four bits that set up the test mode. The transmit characterization test mode is entered when SCHAR_ENA=1 and SCHAR_TXSEL=1. Entering this mode will cause chip port inputs to directly control the SERDES low-speed transmit ports of one of the channels as shown in Table 23.

Table 23. SERDES Transmit Characterization Mode

Chip Port	SERDES Input
PSCHAR_CKIO0	TBCBx
PSCHAR_LDIO[9:0]	LDINBx[9:0]

The x in the table will be a single channel in SERDES quad B, selected by the SCHAR_CHAN control bits. The decoding of SCHAR_CHAN is shown in Table 24.

Table 28. ORT42G5 Memory Map (Continued)

(0x) Absolute Address	Bit	Name	Reset Value (0x)	Description
30933	[0:3]	—	00	Reserved for future use.
	[4:5]	—		Reserved for future use.
	[6]	—		Reserved for future use.
	[7]	—		Reserved for future use.
Status Registers (Read Only, Clear on Read), xx = [AC, AD, BC or BD]				
30804 - Ax 30904 - Bx	[0:1]	—	00	Reserved for future use.
	[2:3]	—		Reserved for future use.
	[4:5]	XAUISTAT_xC		XAUI Status Register. Status of XAUI link state machine for Channel xC 00 – No synchronization, 10 – Synchronization done, 11 – Not used, 01 – no_comma (see XAUI state machine) and at least one CV detected. XAUISTAT_xC[0:1] = 00 on device reset.
	[6:7]	XAUISTAT_xD		XAUI Status Register. Status of XAUI link state machine for Channel xD 00 – No synchronization, 10 – Synchronization done, 11 – Not used, 01 – no_comma (see XAUI state machine) and at least one CV detected. XAUISTAT_xD[0:1] = 00 on device reset.
30805 - Ax 30905 - Bx	[0]]	—	00	Reserved for future use.
	[1]	—		Reserved for future use.
	[2]	DEMUXWAS_xC		Status of Word Alignment. When DEMUX_WAS_xC=1, word alignment is achieved for Channel xC. DEMUX_WAS_xC=0 on device reset.
	[3]	DEMUXWAS_xD		Status of Word Alignment. When DEMUX_WAS_xD=1, word alignment is achieved for Channel xD. DEMUX_WAS_xD=0 on device reset.
	[4]	—		Reserved for future use.
	[5]	—		Reserved for future use.
	[6]	CH24_SYNC_xC		Status of Channel Alignment. When CH24_SYNC_xC=1, multi-channel alignment is achieved for Channel xC. CH24_SYNC_xC=0 on device reset.
	[7]	CH24_SYNC_xD		Status of Channel Alignment. When CH24_SYNC_xD=1, multi-channel alignment is achieved for Channel xD. CH24_SYNC_xD=0 on device reset.
30814 - A 30914 - B	[0]	—	00	Reserved for future use.
	[1]	SYNC2_[A:B]_OVFL		Multi-Channel Overflow Status. When SYNC2_[A:B]_OVFL=1, twin channel synchronization FIFO overflow has occurred. SYNC2_[A:B]_OVFL=0 on device reset.
	[2:3]	—		Reserved for future use.
	[4]	SYNC2_[A:B]_OOS		Multi-Channel Out-Of-Sync Status. When SYNC2_[A:B]_OOS=1, twin channel synchronization has failed. SYNC2_[A:B]_OOS=0 on device reset.
	[5:7]	—		Reserved for future use.

Table 40. Pin Descriptions (Continued)

Symbol	I/O	Description
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 if boundary scan is not used.[1]
RDY/$\overline{\text{BUSY}}$/RCLK	O	During configuration in asynchronous 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. During the master parallel configuration mode, RCLK is a read output signal to an external memory. This output is not normally used.
	I/O	After configuration this pin is a user-programmable I/O pin.[1]
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.[1]
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.[1]
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.[1]
$\overline{\text{CS0}}$, 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, if MPI is not used, these pins are user-programmable I/O pins.[1]
$\overline{\text{RD}}$/$\overline{\text{MPI\_STRB}}$	I	$\overline{\text{RD}}$ is used in the asynchronous peripheral configuration mode. A low on $\overline{\text{RD}}$ changes D[7:3] into a status output. $\overline{\text{WR}}$ 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. As a status indication, a high indicates ready, and a low indicates busy.
$\overline{\text{WR}}$/MPI_RW	I	$\overline{\text{WR}}$ is used in asynchronous peripheral mode. A low on $\overline{\text{WR}}$ transfers data on D[7:0] to the FPGA. In MPI mode, a high on MPI_RW allows a read from the data bus, while a low causes a write transfer to the FPGA.
	I/O	After configuration, if the MPI is not used, $\overline{\text{WR}}$/MPI_RW is a user-programmable I/O pin.[1]
PPC_A[14:31]	I	During MPI mode the PPC_A[14:31] are used as the address bus driven by the PowerPC bus master utilizing the least-significant bits of the PowerPC 32-bit address.
MPI_BURST	I	$\overline{\text{MPI\_BURST}}$ is driven low to indicate a burst transfer is in progress in MPI mode. Driven high indicates that the current transfer is not a burst.
MPI_BDIP	I	$\overline{\text{MPI\_BDIP}}$ is driven by the PowerPC processor in MPI mode. 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[0:1]	I	MPI_TSZ[0:1] signals are driven by the bus master in MPI mode to indicate the data transfer size for the transaction. Set 01 for byte, 10 for half-word, and 00 for word.
A[21:0]	O	During master parallel mode A[21:0] address the configuration EPROMs up to 4M bytes.
	I/O	If not used for MPI these pins are user-programmable I/O pins after configuration.[1]
MPI_ACK	O	In MPI mode this is driven low indicating the MPI received the data on the write cycle or returned data on a read cycle.
	I/O	If not used for MPI these pins are user-programmable I/O pins after configuration.[1]

Table 41. FPSC Function Pin Descriptions (Continued)

Symbol	I/O	Description
HDOUTP_AB (ORT82G5 only)	O	High-speed CML transmit data output – SERDES quad A, channel B.
HDOUTN_AC	O	High-speed CML transmit data output – SERDES quad A, channel C.
HDOUTP_AC	O	High-speed CML transmit data output – SERDES quad A, channel C.
HDOUTN_AD	O	High-speed CML transmit data output – SERDES quad A, channel D.
HDOUTP_AD	O	High-speed CML transmit data output – SERDES quad A, channel D.
HDOUTN_BA (ORT82G5 only)	O	High-speed CML transmit data output – SERDES quad B, channel A.
HDOUTP_BA (ORT82G5 only)	O	High-speed CML transmit data output – SERDES quad B, channel A.
HDOUTN_BB (ORT82G5 only)	O	High-speed CML transmit data output – SERDES quad B, channel B.
HDOUTP_BB (ORT82G5 only)	O	High-speed CML transmit data output – SERDES quad B, channel B.
HDOUTN_BC	O	High-speed CML transmit data output – SERDES quad B, channel C.
HDOUTP_BC	O	High-speed CML transmit data output – SERDES quad B, channel C.
HDOUTN_BD	O	High-speed CML transmit data output – SERDES quad B, channel D.
HDOUTP_BD	O	High-speed CML transmit data output – SERDES quad B, channel D.
Power and Ground		
VDDIB_AA (ORT82G5 only)	—	1.8V/1.5V power supply for high-speed serial input buffers.
VDDIB_AB (ORT82G5 only)	—	1.8V/1.5V power supply for high-speed serial input buffers.
VDDIB_AC	—	1.8V/1.5V power supply for high-speed serial input buffers.
VDDIB_AD	—	1.8V/1.5V power supply for high-speed serial input buffers.
VDDIB_BA (ORT82G5 only)	—	1.8V/1.5V power supply for high-speed serial input buffers.
VDDIB_BB (ORT82G5 only)	—	1.8V/1.5V power supply for high-speed serial input buffers.
VDDIB_BC	—	1.8V/1.5V power supply for high-speed serial input buffers.
VDDIB_BD	—	1.8V/1.5V power supply for high-speed serial input buffers.
VDDOB_AA (ORT82G5 only)	—	1.8V/1.5V power supply for high-speed serial output buffers.
VDDOB_AB (ORT82G5 only)	—	1.8V/1.5V power supply for high-speed serial output buffers.
VDDOB_AC	—	1.8V/1.5V power supply for high-speed serial output buffers.
VDDOB_AD	—	1.8V/1.5V power supply for high-speed serial output buffers.
VDDOB_BA (ORT82G5 only)	—	1.8V/1.5V power supply for high-speed serial output buffers.
VDDOB_BB (ORT82G5 only)	—	1.8V/1.5V power supply for high-speed serial output buffers.
VDDOB_BC	—	1.8V/1.5V power supply for high-speed serial output buffers.
VDDOB_BD	—	1.8V/1.5V power supply for high-speed serial output buffers.
VDDGB_A	—	1.5V guard band power supply.
VDDGB_B	—	1.5V guard band power supply.
VDD_ANA	—	1.5V power supply for SERDES analog receive and transmit circuitry.

1. Should be externally connected on board to 3.3V pull-up resistor.

Package Information

Package Pinouts

Table 43 provides the number of user-programmable I/Os available for each package.

Table 43. I/O Summary

Device	ORT42G5	ORT82G5
User programmable I/O	204	372
Available programmable differential pair pins	166	330
FPGA configuration pins	7	7
FPGA dedicated function pins	2	2
Core function pins	32	71
VDD15	49	63
VDD33	8	10
VDDIO	34	32
VSS	112	91
VDDGB	2	2
VDDIB	4	8
VDDOB	8	12
VDD_ANA	22	8
No connect	0	2
Total package pins	484	680

Table 44 and Table 45 provide the package pin and pin function for the ORT42G5 and ORT82G5 FPSC and packages. The bond pad name is identified in the PIO nomenclature used in the ispLEVER System software design editor. The Bank column provides information as to which output voltage level bank the given pin is in. The Group column provides information as to the group of pins the given pin is in. This is used to show which VREF pin is used to provide the reference voltage for single-ended limited-swing I/Os. If none of these buffer types (such as SSTL, GTL, HSTL) are used in a given group, then the VREF pin is available as an I/O pin.

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 column for the FPGA. The tables provide no information on unused pads.

As shown in the pair columns in Table 38, differential pairs and physical locations are numbered within each bank (e.g., L19C-A0 is the nineteenth pair in an associated bank). A ‘C’ indicates complementary differential, whereas a ‘T’ indicates true differential. An _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 balls are diagonally adjacent, separated by one physical ball.

VREF pins, shown in the Pin Description columns in Table 44 and Table 45, 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. ORT42G5 484-pin PBGAM (fpBGA) Pinout (Continued)

484-PBGAM	VDDIO Bank	VREF Group	I/O	Pin Description	Additional Function	484-PBGAM
T22	-	-	I	HDINP_BC	-	HSP_2
J19	-	-	VDD_ANA	VDD_ANA	-	-
F20	-	-	VSS	VSS	-	-
K16	-	-	VDD_ANA	VDD_ANA	-	-
R20	-	-	VDDOB	VDDOB_BC	-	-
R21	-	-	O	HDOUTN_BC	-	HSN_3
G19	-	-	VSS	VSS	-	-
R22	-	-	O	HDOUTP_BC	-	HSP_3
P21	-	-	VDDOB	VDDOB_BC	-	-
H16	-	-	VSS	VSS	-	-
P22	-	-	VDDIB	VDDIB_BD	-	-
K17	-	-	VDD_ANA	VDD_ANA	-	-
N22	-	-	I	HDINN_BD	-	HSN_4
H17	-	-	VSS	VSS	-	-
N21	-	-	I	HDINP_BD	-	HSP_4
K18	-	-	VDD_ANA	VDD_ANA	-	-
H18	-	-	VSS	VSS	-	-
K19	-	-	VDD_ANA	VDD_ANA	-	-
P20	-	-	VDDOB	VDDOB_BD	-	-
M22	-	-	O	HDOUTN_BD	-	HSN_5
H19	-	-	VSS	VSS	-	-
M21	-	-	O	HDOUTP_BD	-	HSP_5
N20	-	-	VDDOB	VDDOB_BD	-	-
L16	-	-	VSS	VSS	-	-
L17	-	-	VSS	VSS	-	-
M20	-	-	VDDOB	VDDOB_AD	-	-
L22	-	-	O	HDOUTP_AD	-	HSP_6
L18	-	-	VSS	VSS	-	-
L21	-	-	O	HDOUTN_AD	-	HSN_6
L20	-	-	VDDOB	VDDOB_AD	-	-
N16	-	-	VDD_ANA	VDD_ANA	-	-
L19	-	-	VSS	VSS	-	-
N17	-	-	VDD_ANA	VDD_ANA	-	-
K22	-	-	I	HDINP_AD	-	HSP_7
M16	-	-	VSS	VSS	-	-
K21	-	-	I	HDINN_AD	-	HSN_7
N18	-	-	VDD_ANA	VDD_ANA	-	-
K20	-	-	VDDIB	VDDIB_AD	-	-
M17	-	-	VSS	VSS	-	-
J20	-	-	VDDOB	VDDOB_AC	-	-
J21	-	-	O	HDOUTP_AC	-	HSP_8
M18	-	-	VSS	VSS	-	-
J22	-	-	O	HDOUTN_AC	-	HSN_8
H20	-	-	VDDOB	VDDOB_AC	-	-

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

680-PBGAM	VDDIO Bank	VREF Group	I/O	Pin Description	Additional Function	680-PBGAM
AM4	6 (BL)	5	IO	PB2A	DP2	L13T_D0
AL5	6 (BL)	5	IO	PB2B	—	L13C_D0
AN7	6 (BL)	—	VDDIO6	VDDIO6	—	—
AP3	6 (BL)	5	IO	PB2C	PLL_CK6T/PPLL	L14T_A0
AP4	6 (BL)	5	IO	PB2D	PLL_CK6C/PPLL	L14C_A0
AN4	6 (BL)	5	IO	PB3B	—	—
U16	—	—	VSS	VSS	—	—
AK6	6 (BL)	5	IO	PB3C	—	L15T_A0
AK7	6 (BL)	5	IO	PB3D	—	L15C_A0
AL6	6 (BL)	5	IO	PB4A	VREF_6_05	L16T_A0
AM6	6 (BL)	5	IO	PB4B	DP3	L16C_A0
AP1	6 (BL)	—	VDDIO6	VDDIO6	—	—
AN5	6 (BL)	6	IO	PB4C	—	L17T_A0
AP5	6 (BL)	6	IO	PB4D	—	L17C_A0
AK8	6 (BL)	6	IO	PB5B	—	—
U17	—	—	VSS	VSS	—	—
AP6	6 (BL)	6	IO	PB5C	VREF_6_06	L18T_D0
AP7	6 (BL)	6	IO	PB5D	D14	L18C_D0
AM7	6 (BL)	6	IO	PB6A	—	L19T_D0
AN6	6 (BL)	6	IO	PB6B	—	L19C_D0
AP2	6 (BL)	—	VDDIO6	VDDIO6	—	—
AL8	6 (BL)	7	IO	PB6C	D15	L20T_A0
AL9	6 (BL)	7	IO	PB6D	D16	L20C_A0
AK9	6 (BL)	7	IO	PB7B	—	—
U18	—	—	VSS	VSS	—	—
AN8	6 (BL)	7	IO	PB7C	D17	L21T_A0
AM8	6 (BL)	7	IO	PB7D	D18	L21C_A0
AN9	6 (BL)	7	IO	PB8A	—	L22T_D0
AP8	6 (BL)	7	IO	PB8B	—	L22C_D0
AK10	6 (BL)	7	IO	PB8C	VREF_6_07	L23T_A0
AL10	6 (BL)	7	IO	PB8D	D19	L23C_A0
AP9	6 (BL)	8	IO	PB9B	—	—
U19	—	—	VSS	VSS	—	—
AM10	6 (BL)	8	IO	PB9C	D20	L24T_A0
AM11	6 (BL)	8	IO	PB9D	D21	L24C_A0
AK11	6 (BL)	8	IO	PB10B	—	—
AN10	6 (BL)	8	IO	PB10C	VREF_6_08	L25T_A0
AP10	6 (BL)	8	IO	PB10D	D22	L25C_A0
AN11	6 (BL)	9	IO	PB11A	—	L26T_A0
AP11	6 (BL)	9	IO	PB11B	—	L26C_A0
V16	—	—	VSS	VSS	—	—
AL12	6 (BL)	9	IO	PB11C	D23	L27T_A0
AK12	6 (BL)	9	IO	PB11D	D24	L27C_A0
AN12	6 (BL)	9	IO	PB12A	—	L28T_A0

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

680-PBGAM	VDDIO Bank	VREF Group	I/O	Pin Description	Additional Function	680-PBGAM
AM28	5 (BC)	7	IO	PB31D	—	L26C_A0
AN30	5 (BC)	7	IO	PB32B	—	—
R14	—	—	VSS	VSS	—	—
AK25	5 (BC)	7	IO	PB32C	—	L27T_D0
AL26	5 (BC)	7	IO	PB32D	—	L27C_D0
AN17	5 (BC)	—	VDDIO5	VDDIO5	—	—
AL27	5 (BC)	8	IO	PB33C	—	L28T_A0
AL28	5 (BC)	8	IO	PB33D	VREF_5_08	L28C_A0
AN31	5 (BC)	8	IO	PB34B	—	—
R15	—	—	VSS	VSS	—	—
AK26	5 (BC)	8	IO	PB34D	—	—
AM30	5 (BC)	9	IO	PB35B	—	—
AL29	5 (BC)	9	IO	PB35D	VREF_5_09	—
AK27	5 (BC)	9	IO	PB36B	—	—
R20	—	—	VSS	VSS	—	—
AL30	5 (BC)	9	IO	PB36C	—	L29T_D0
AK29	5 (BC)	9	IO	PB36D	—	L29C_D0
AK28	—	—	VDD33	VDD33	—	—
AA16	—	—	VDD15	VDD15	—	—
AP32	—	—	IO	PSCHAR_LDIO9	—	—
AP33	—	—	IO	PSCHAR_LDIO8	—	—
AN32	—	—	IO	PSCHAR_LDIO7	—	—
AM31	—	—	IO	PSCHAR_LDIO6	—	—
AA17	—	—	VDD15	VDD15	—	—
AM32	—	—	VDD33	VDD33	—	—
AL31	—	—	IO	PSCHAR_LDIO5	—	—
AM33	—	—	IO	PSCHAR_LDIO4	—	—
AA18	—	—	VDD15	VDD15	—	—
AK30	—	—	IO	PSCHAR_LDIO3	—	—
AL32	—	—	IO	PSCHAR_LDIO2	—	—
AA19	—	—	VDD15	VDD15	—	—
AB16	—	—	VDD15	VDD15	—	—
AK31	—	—	VDD33	VDD33	—	—
AJ30	—	—	IO	PSCHAR_LDIO1	—	—
AK33	—	—	IO	PSCHAR_LDIO0	—	—
AK34	—	—	IO	PSCHAR_CKIO1	—	—
AJ31	—	—	IO	PSCHAR_CKIO0	—	—
AJ33	—	—	IO	PSCHAR_XCK	—	—
AJ34	—	—	IO	PSCHAR_WDSYNC	—	—
AH30	—	—	IO	PSCHAR_CV	—	—
AH31	—	—	IO	PSCHAR_BYTSYNC	—	—
AH32	—	—	O	ATMOUT_B (no connect)	—	—
AH33	—	—	VSS	VSS	—	—
AH34	—	—	VDDGB_B	VDDGB_B	—	—

Technical Support Assistance

Hotline: 1-800-LATTICE (North America)
+1-503-268-8001 (Outside North America)
e-mail: techsupport@latticesemi.com
Internet: www.latticesemi.com

Revision History

Date	Version	Change Summary
—	—	Previous Lattice releases.
July 2008	07.0	BM680 conversion to F680 per PCN#09A-08.