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	0°C ~ 70°C (TA)
Package / Case	680-BBGA
Supplier Device Package	680-FPBGA (35x35)
Purchase URL	https://www.e-xfl.com/product-detail/lattice-semiconductor/ort82g5-1fn680c

Embedded Function Features

- High-speed SERDES with programmable serial data rates over the range 0.6 to 3.7 Gbps. Operation has been demonstrated on design tolerance devices at 3.7 Gbps across 26 in. of FR-4 backplane and at 3.125 Gbps across 40 in. of FR-4 backplane across temperature and voltage specifications.
- Asynchronous operation per receive channel with the receiver frequency tolerance based on one reference clock per block channels (separate PLL per channel).
- Ability to select full-rate or half-rate operation per transmit or receive channel by setting the appropriate control registers.
- Programmable one-half amplitude transmit mode for reduced power in chip-to-chip application.
- Transmit preemphasis (programmable) for improved receive data eye opening.
- 32-bit (8b/10b) or 40-bit (raw data) parallel internal bus for data processing in FPGA logic.
- Provides a 10 Gbps backplane interface to switch fabric. Also supports multiple port cards at 2.5 Gbps.
- 3.125 Gbps SERDES compliant with XAUI serial data specification for 10 G Ethernet applications with protection.
- IEEE 802.3ae compliant XAUI transceiver. Includes embedded IEEE 802.3ae-based XAUI link state machine.
- Compliant to FC-0 specification for 1 Gbps, 2Gbps, 10 Gbps (FC-XAUI) modes. Includes Fibre Channel link state machine.
- High-Speed Interface (HSI) function for clock/data recovery serial backplane data transfer without external clocks.
- SERDES has low-power CML buffers. Support for 1.5V/1.8V I/Os. Allows use with optical transceiver, coaxial copper media, shielded twisted pair wiring or high-speed backplanes such as FR-4.
- Power down option of SERDES HSI receiver or transmitter on a per-channel basis.
- Automatic lock to reference clock in the absence of valid receive data.
- High-speed and low-speed loopback test modes.
- Requires no external component for clock recovery and frequency synthesis.
- SERDES characterization pins available to control/monitor the internal interface to one SERDES block (ORT82G5 only).
- SERDES HSI automatically recovers from loss-of-clock once its reference clock returns to normal operating state.
- Built-in boundary scan (IEEE ® 1149.1 and 1149.2 JTAG) for the programmable I/Os, not including the SERDES interface.
- FIFOs can align incoming data either across all eight channels (ORT82G5 only), across one or two groups of four channels, or across two or four groups of two channels. Alignment is done either using comma characters or by using the /A/ character in XAUI mode. Optionally, the alignment FIFOs can be bypassed for asynchronous operation between channels. (Each channel includes its own clock and frame pulse or comma detect.)
- Addition of two 4K x 36 dual-port RAMs with access to the programmable logic.
- The ORT82G5 is pinout compatible to the ORCA ORSO82G5 SONET backplane driver FPSC. The ORT42G5 is pin compatible to the ORSO42G5.

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, complied 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.

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.

Transmit Path (FPGA to Backplane) Logic

The transmitter section accepts four groups of either 8-bit unencoded data or 10-bit encoded data at the parallel interface to the FPGA logic. It also uses the reference clock, REFCLK[P:N]_[A:B] to synthesize an internal high-speed serial bit clock. The serialized transmitted data are available at the differential CML output pins to drive either an optical transmitters, coaxial media or a circuit board backplane.

As shown in Figure 3, the basic blocks in the transmit path include:

Embedded Core/FPGA interface and 4:1 multiplexer

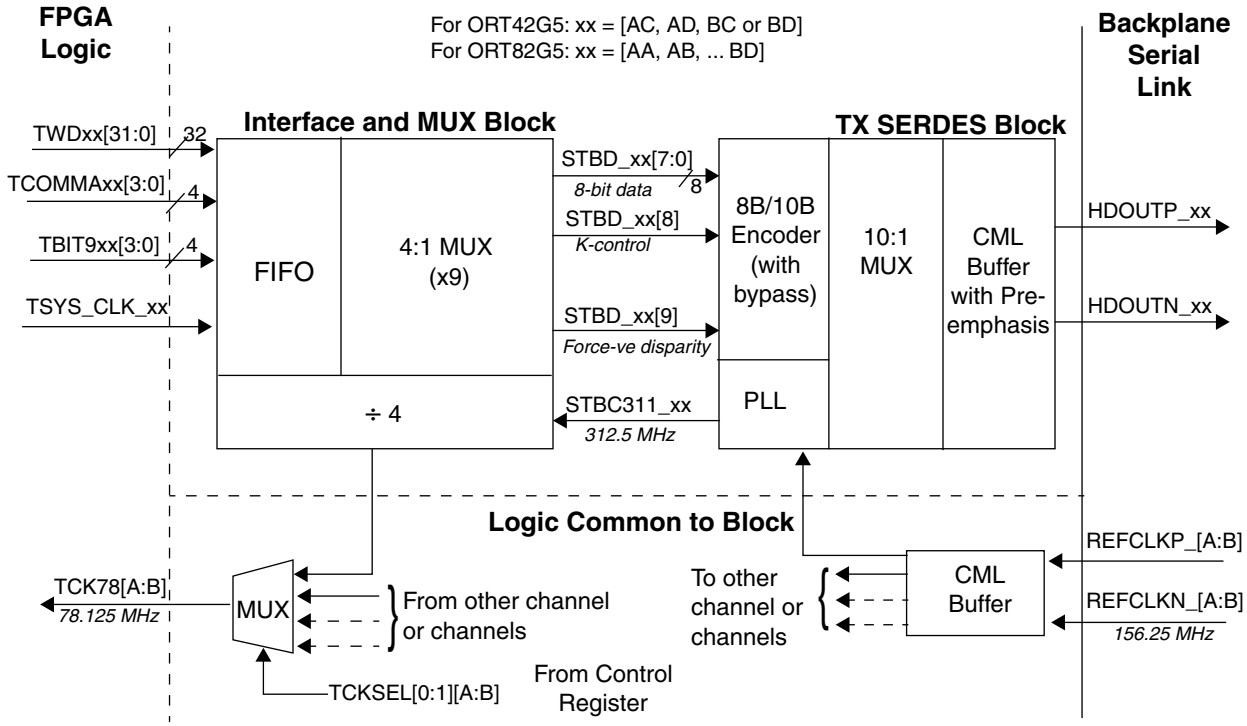
- Low speed parallel core/FPGA interface
- 4:1 multiplexer

Transmit SERDES

- 8b/10b Encoder
- 10:1 Multiplexer
- CML Output Buffer

Detailed descriptions of the logic blocks are given in following sections. Detailed descriptions of transmit clock distribution, including the transmit PLL are given in later sections of this data sheet.

Figure 3. Basic Logic Blocks, Transmit Path, Single Channel (Typical Reference Clock Frequency)



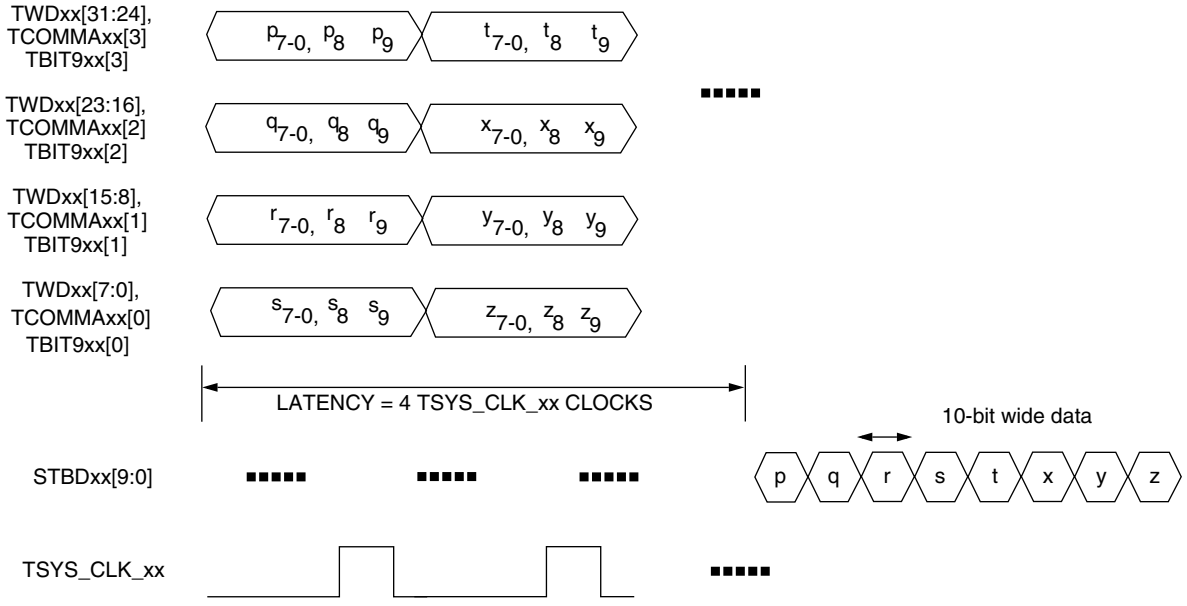
Embedded Core/FPGA Logic Interface and 4:1 Multiplexer

These blocks provide the data formatting and transmit data and clock signal transfers between the Embedded Core and the FPGA Logic. Control and status registers in the FPGA portion of the chip contain to control the transmit logic and record status. These bits are passed to the core using the FPGA System Bus and are described in later sections of this data sheet.

The low-speed transmit interface consists of a clock and 4 data bytes, each with an accompanying control bit. The data bytes are conveyed to the MUX via the TWDxx[31:0] ports (where xx represents the channel label [AA,...,BD] or [AC, AD, BC, BD]). The control bits are TCOMMAx[3:0] which define whether the input byte is to be interpreted as data or as a special character and TBIT9xx[3:0] which are used to force a negative disparity present state. The data and control signals are synchronized to the transmit clock, TSYS_CLK_xx. Both the data and control are strobed into the core on the rising edge of TSYS_CLK_xx. Note that each TBIT9xx[3:0] controls the disparity of the encoded version of its corresponding data byte. Setting bit TBIT9AC[3] to 1, for instance, will force the 8b/10b encoder to assess a current negative running disparity state. This will cause it to encode TWDAC[31:24] positively (more 1's than 0's). Setting TBIT9xx to 0 will leave the encoder free to alternate between positive and negative encoding to maintain a zero running disparity.

The MUX is responsible for taking 40 bits of data/control at the low-speed transmit interface and up-converting it to 10 bits of data/control at the SERDES transmit interface. The MUX has 2 clock domains - one based on the clock received from the SERDES block and a second that comes from the FPGA at 1/4 the frequency of the SERDES clock. The time sequence of interleaving data/control values is shown in Figure 4.

Figure 4. Transmit MUX Block Timing - Single Channel



SERDES Block

The SERDES block accepts either 8-bit data to be encoded or 10-bit unencoded data at the parallel input port from the MUX/DEMUX block. It also accepts the reference clock at the REFCLK_[A:B] input and uses this clock to synthesize the internal high-speed serial bit clock.

The internal STBC311xx clock is derived from the reference clock. The frequency of this clock depends on the setting of the half-rate/full-rate control bit setting the mode of the SERDES and the frequency of the REFCLK_[A:B] and/or that of the high-speed serial data. A falling edge on the STBC311xx clock port will cause a new data character to be transferred into the SERDES block. The latency from the SERDES block input to the high-speed serial output is 5 STBC311xx clock cycles, as shown in Figure 5.

Multi-channel Alignment

The alignment FIFO allows the transfer of all data to the system clock. The Multi-Channel Alignment block (Figure 6) allows the system to be configured to allow the frame alignment of multiple slightly varying data streams. This optional alignment ensures that matching SERDES streams will arrive at the FPGA end in perfect data synchronization.

Each channel is provided with a 24 word x 36-bit FIFO. The FIFO can perform two tasks: (1) to change the clock domain from receive clock to a clock from the FPGA side, and (2) to align the receive data over 2, 4, or 8 channels. This FIFO allows a timing budget of ±230.4 ns that can be allocated to skew between the data lanes and for transfer to the system clock. The input to the FIFO consists of 36 bits of demultiplexed data, RALIGN_xx[3:0], RWD_xx[31:0], and RWBIT8_xx[3:0].

The four RALIGN_xx bits are control signals, and can be the alignment character detect signals indicating the presence of a comma character in Fibre Channel mode and the /A/ character in XAUI mode. The other 32 RWD_xx bits are the 8-bit data bytes from the 8b/10b decoder. The alignment character, if present, is the MSB of the data. The RWBIT8_xx indicates the presence of a Km.n control character in the receive data byte. Only RWBIT8_xx and RWD_xx inputs are stored in the FIFO. During alignment process, RALIGN[3]_xx is used to synchronize multiple channels.

If a channel is not in any alignment group, it will set the FIFO-write-address to the beginning of the FIFO, and will set the FIFO-read-address to the middle of the FIFO, at the first assertion of RALIGN[3]_xx after reset or after the resync command.

The RX_FIFO_MIN_xx register bits can be used to control the threshold for minimum unused buffer space in the alignment FIFOs between read and write pointers before overflow (OVFL) status is flagged. The synchronization algorithm consists of a down counter which starts to count down by 1 from its initial value of 18 (decimal) when an alignment character from any channel within an alignment group has been received. Once all the alignment characters within the alignment group have been received, the count is decremented by 2 until 0 is reached. Data is then read from the FIFOs and output to the FPGA. This algorithm is not repeated after multi-channel alignment has been achieved; resynchronization must be forced by toggling the appropriate FMPU_RESYNC bit.

ORT42G5 Multi-channel Alignment

The ORT42G5 has a total of four channels. The incoming data of these channels can be synchronized in two ways or they can be independent of one other. Two channels, C and D, within either SERDES block can be aligned together to form a pair, as shown in Figure 11. Alternately, all four channels can be aligned together to form a communication channel with a bandwidth of 10 Gbps, as shown in Figure 12. Individual channels within an alignment group can be disabled (i.e., powered down) without disrupting other channels.

Figure 11. Dual Channel Alignment

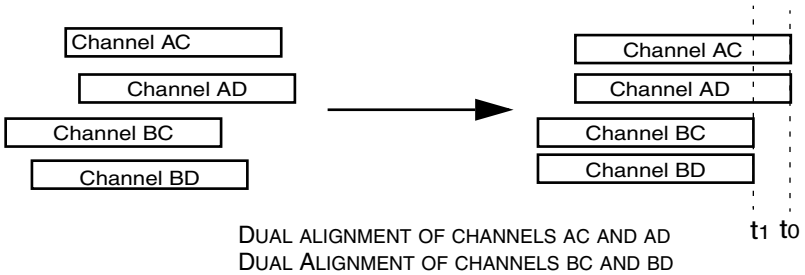
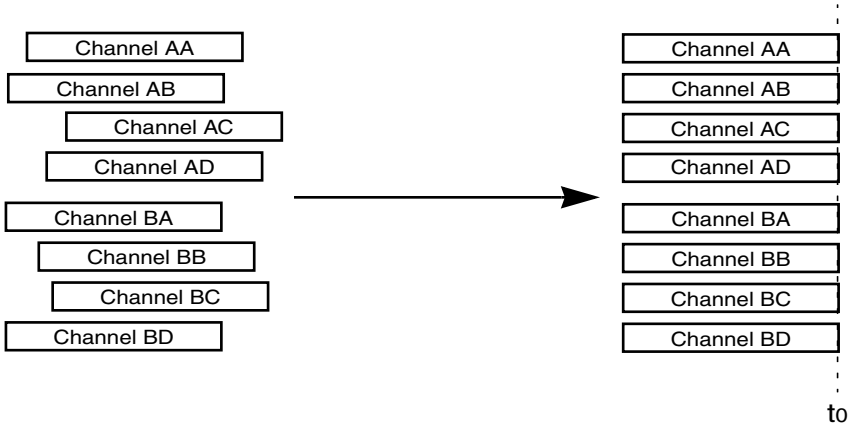


Figure 15. Alignment of all Eight SERDES Channels.



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.

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.

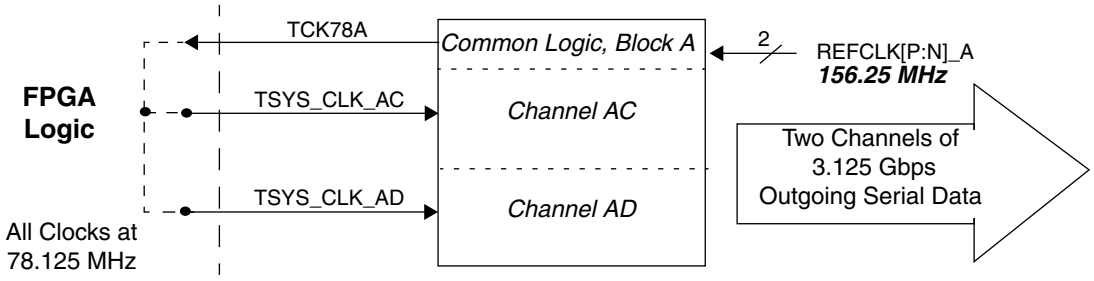
Table 14. TCK78[A:B] Source Selection

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

Recommended Transmit Clock Distribution for the ORT42G5

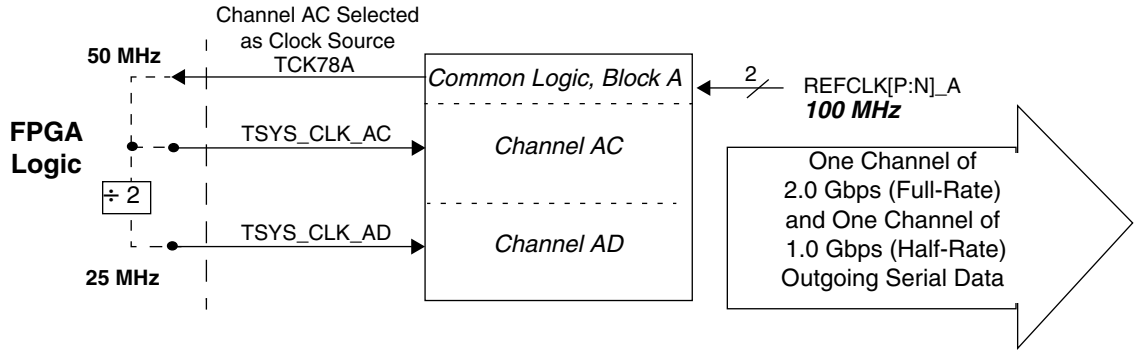
As an example of the recommended clock distribution approach, TSYS_CLK_A[C or D] can be sourced by TCK78A as shown in Figure 18 if the transmit line rate are common for both channels in a block. Similar clocking would be used for Block B.

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.

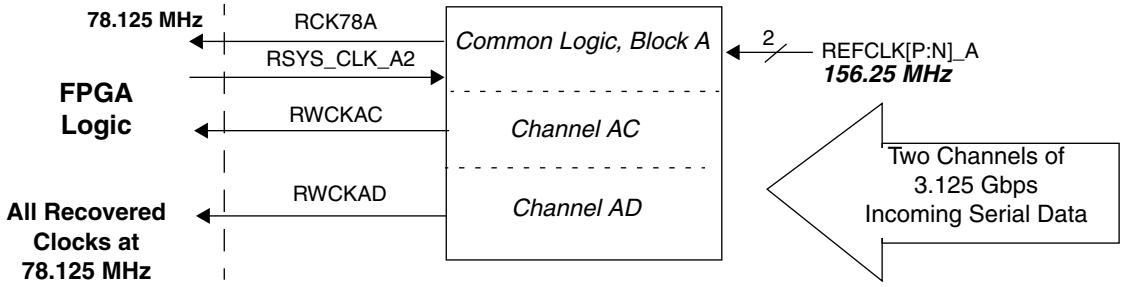
Table 15. RCK78[A:B] Source Selection

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.

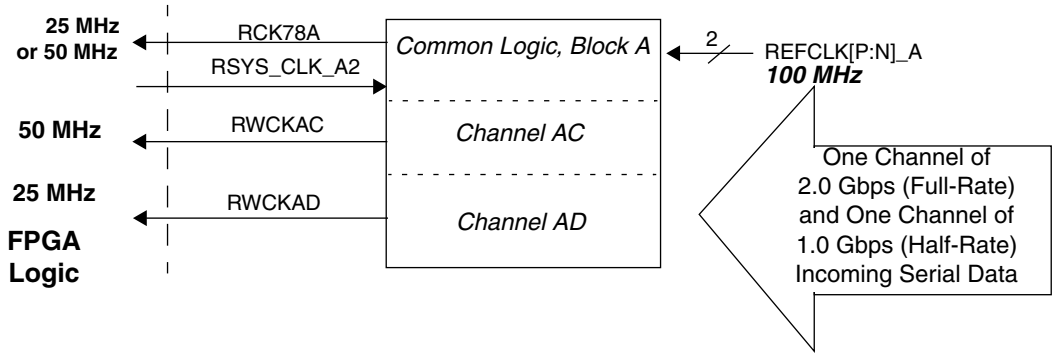
Figure 21 shows the recommended receive clocking for a single block.

Figure 20. Receive Clocking for a Single Block (Similar Connections Would Be Used for Block B)



The receive channel alignment bypass mode allows mixing of half and full line rates among the channels, as shown in Figure 21. The figure shows channel AC configured in full rate mode at 2.0 Gbps. Channel AD configured in half-rate mode at 1.0 Gbps. The receive alignment FIFO per channel cannot be used in this mode.

Figure 21. Receive Clocking for Mixed Line Rates



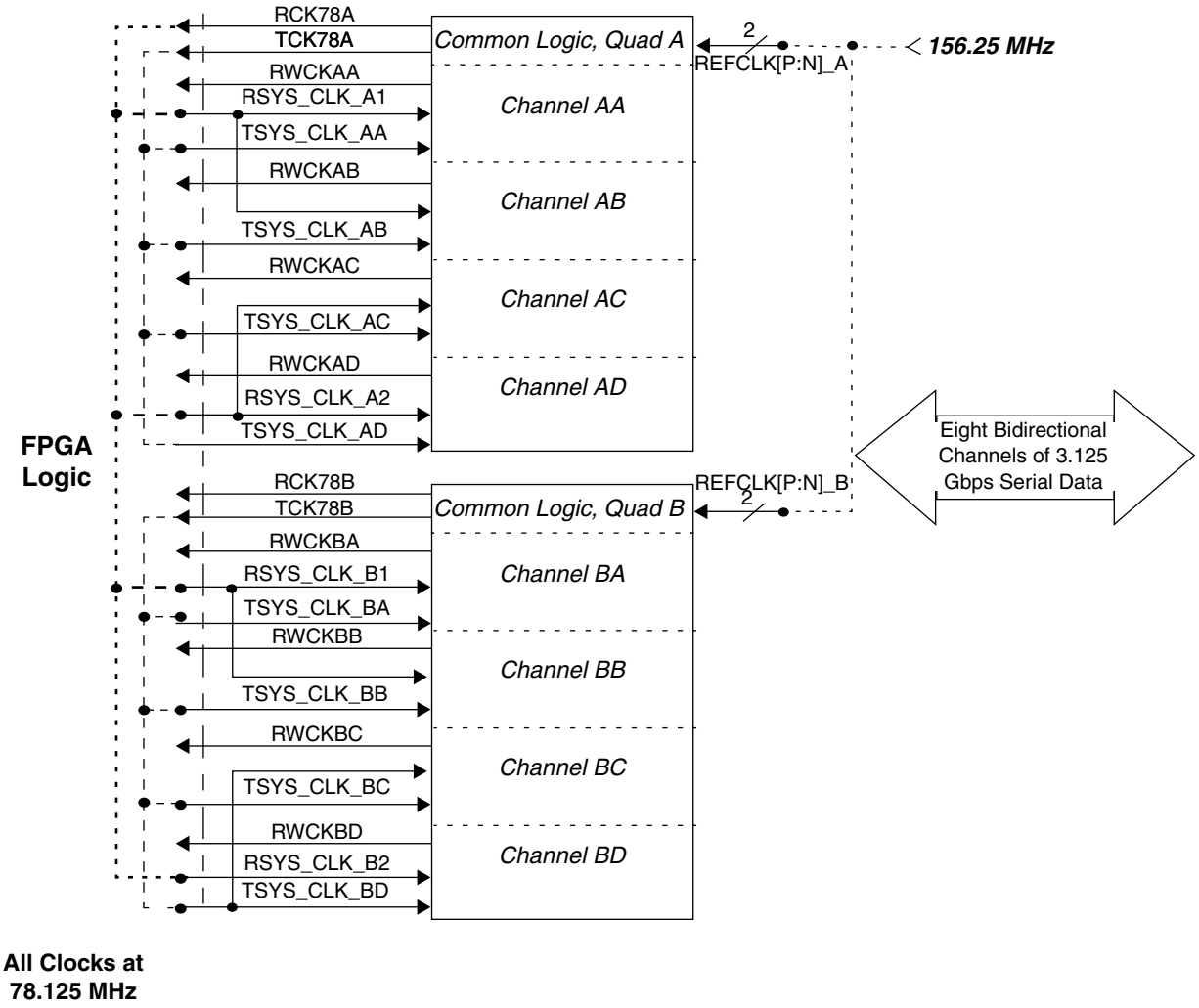
Each SERDES block can also be configured for any line rate (0.6 to 3.7 Gbps), since each block has its own reference clock input pins.

Multi-Channel Alignment Clocking Strategies for the ORT42G5

The data on the four channels in the ORT42G5 can be independent of each other or can be synchronized in two different ways. For example, two channels within a SERDES block can be aligned together, channel C and channel D. Alternatively, all four channels in a SERDES block 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. Clocking strategies for these various modes are described in the following paragraphs.

For dual alignment both channels must be sourced by the same clock. Either RWCKAC or RWCKAD can be connected to RSYS_CLK_A2. A clocking example for dual alignment is shown in Figure 22.

Figure 31. Clocking for Eight Channel Alignment



Reset Operation

The SERDES block can be reset in one of three different ways as follows: on power up, using the hardware reset, or via the microprocessor interface. The power up reset process begins when the power supply voltage ramps up to approximately 80% of the nominal value of 1.5V. Following this event, the device will be ready for normal operation after 3 ms.

A hardware reset is initiated by making the PASB_RESETN low for at least two microprocessor clock cycles. The device will be ready for operation 3 ms after the low to high transition of the PASB_RESETN. This reset function affects all SERDES channels and resets all microprocessor and internal registers and counters.

Using the software reset option, each channel can be individually reset by setting SWRST (bit 2) to a logic 1 in the channel configuration register. The device will be ready 3 ms after the SWRST bit is deasserted. Similarly, all four channels per quad SERDES can be reset by setting the global reset bit GSWRST. The device will be ready for normal operation 3 ms after the GSWRST bit is deasserted. Note that the software reset option resets only SERDES internal registers and counters. The microprocessor registers are not affected. It should also be noted that the embedded block cannot be accessed until after FPGA configuration is complete.

Table 28. ORT42G5 Memory Map (Continued)

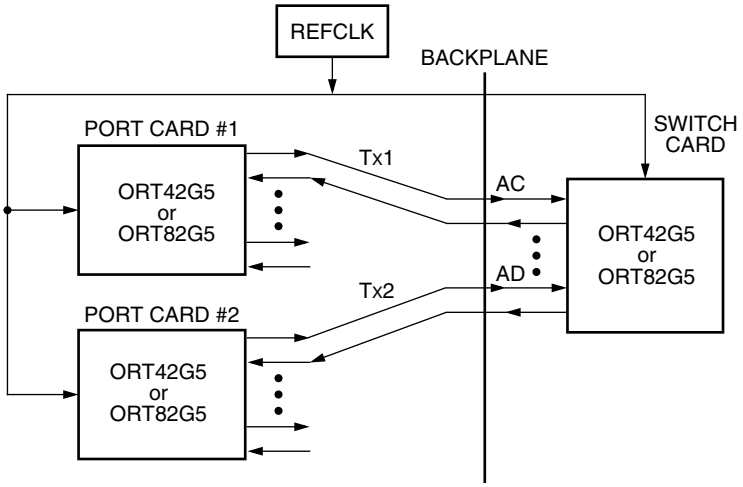
(0x) Absolute Address	Bit	Name	Reset Value (0x)	Description
SERDES Common Transmit and Receive Channel Configuration Registers (Read/Write), xx = [AC, AD, BC or BD]				
30024 - AC 30034 - AD 30124 - BC 30134 - BD	[0]	Reserved	See Bit Desc.	Reserved, must be 0. Set to 0 on device reset.
	[1]	MASK_xx		Transmit and Receive Alarm Mask Bit, Channel xx. When MASK_xx = 1, the transmit and receive alarms of a channel are prevented from generating an interrupt (i.e., they are masked or disabled). The MASK_xx bit overrides the individual alarm mask bits in the Alarm Mask Registers. MASK_xx = 1 on device reset.
	[2]	SWRST_xx		Transmit and Receive Software Reset Bit, Channel xx. When SWRST_ss = 1, this bit provides the same function as the hardware reset, except that all configuration register settings are unaltered. This is not a self-clearing bit. Once set, this bit must be manually set and cleared. SWRST = 0 on device reset.
	[3:6]	Not used		Not used
	[7]	TESTEN_xx		Transmit and Receive Test Enable Bit, Channel xx. When TESTEN_xx = 1, the transmit and receive sections are placed in test mode. The TestMode_[A:B][4:0] bits in the Global Control Registers specify the particular test, and must also be set. Note: When the global test enable bit GTESTEN_[A:B] = 0, the individual channel test enable bits are used to selectively place a channel in test or normal mode. When GTESTEN_[A:B] = 1, all channels in a block are set to test mode regardless of their TESTEN setting. TESTEN_xx = 0 on device reset.
SERDES Global Control Registers (Read Write) - Act on Both Channels in SERDES Block A or SERDES Block B.				
30005 - A 30105 - B	[0]	Reserved	See Bit Desc.	Reserved, must be 0. Set to 0 on device reset.
	[1]	GMASK_[A:B]		Global Mask. When GMASK_[A:B] = 1, the transmit and receive alarms of both channels in the SERDES block are prevented from generating an interrupt (i.e., they are masked or disabled). The GMASK_[A:B] bit overrides the individual MASK_xx bits. GMASK_[A:B] = 1 on device reset.
	[2]	GSWRST_[A:B]		Software reset bit. The GSWRST_[A:B] bit provides the same function as the hardware reset for the transmit and receive sections of both channels, except that the device configuration settings are not affected when GSWRST_[A:B] is asserted. This is not a self-clearing bit. Once set, this bit must be manually set and cleared. The GSWRST_[A:B] bit overrides the individual SWRST_xx bits. GSWRST_[A:B] = 0 on device reset.
	[3]	GPWRDNT_[A:B]		Powerdown Transmit Function. When GPWRDNT_[A:B] = 1, sections of the transmit hardware for both channels are powered down to conserve power. The GPWRDNT_[A:B] bit overrides the individual PWRDNT_xx bits. GPWRDNT_[A:B] = 0 on device reset.
	[4]	GPWRDNR_[A:B]		Powerdown Receive Function. When GPWRDNR_[A:B] = 1, sections of the receive hardware for both channels are powered down to conserve power. The GPWRDNR_[A:B] bit overrides the individual PWRDNR_xx bits. GPWRDNR_[A:B] = 0 on device reset.
	[5]	Reserved		Reserved, 1 on device reset.
	[6]	Not used		Not used
	[7]	GTESTEN_[A:B]		Test Enable Control. When GTESTEN_[A:B] = 1, the transmit and receive sections of both channels are placed in test mode. The GTESTEN_[A:B] bit overrides the individual TESTEN_xx bits. GTESTEN_[A:B] = 0 on device reset.
30006 - A 30106 - B	[0:4]	TestMode[A:B]	00	Test Mode - See Test Mode section for settings
	[5]	Not used		Not used
	[6:7]	Reserved		Reserved

Table 28. ORT42G5 Memory Map (Continued)

(0x) Absolute Address	Bit	Name	Reset Value (0x)	Description
30810 - Ax 30910 - Bx	[0]]	—	00	Reserved for future use
	[1]	—		Reserved for future use
	[2]	DOWDALIGN_xC		Word Realign Bit. When DOWDALIGN_xC transitions from 0 to 1, the receiver realigns on the next comma character for Channel xC. NOWDALIGN_xC=0 on device reset.
	[3]	DOWDALIGN_xC		Word Realign Bit. When DOWDALIGN_xC transitions from 0 to 1, the receiver realigns on the next comma character for Channel xC. NOWDALIGN_xC=0 on device reset.
	[4]	—		Reserved for future use. Set to zero.
	[5]	—		Reserved for future use. Set to zero.
	[6]	FMPU_STR_EN _xC		Enable multi-channel alignment for Channel xC. When FMPU_STR_EN _xC = 0, Channel xC is not part of a multi-channel alignment group When FMPU_STR_EN _xC = 1, Channel xC is part of a twin channel alignment (SERDES block A or B) or quad channel alignment (both SERDES blocks) group.
	[7]	FMPU_STR_EN _xD		Enable multi-channel alignment for Channel xD. When FMPU_STR_EN _xD = 0, Channel xD is not part of a multi-channel alignment group When FMPU_STR_EN _xD = 1, Channel xD is part of a twin channel alignment (SERDES block A or B) or quad channel alignment (both SERDES blocks) group.
30811 - Ax 30911 - Bx	[0:7]	FMPU_SYNMODE_ [A:B]	00	Sync mode for block [A:B] 00000000 = No channel alignment 00001010 = Twin channel alignment, SERDES block [A:B] 00001111 = Quad channel alignment (both SERDES blocks)
30820 - Ax 30920 - Bx	[0]	—	00	Reserved for future use.
	[1]	—		Reserved for future use.
	[2]	FMPU_RESYNC1_xC		Resync a Single Channel. When FMPU_RESYNC1_xC transitions from 0 to 1, the corresponding channel xC is resynchronized (the write and read pointers are reset). FMPU_STR_EN_xC=0 on device reset.
	[3]	FMPU_RESYNC1_xD		Resync a Single Channel. When FMPU_RESYNC1_xD transitions from 0 to 1, the corresponding channel xD is resynchronized (the write and read pointers are reset). FMPU_STR_EN_xD=0 on device reset.
	[4]	—		Reserved for future use.
	[5]	FMPU_RESYNC2[A:B]		Resync a Twin-Channel Group. When FMPU_RESYNC2[A:B] transitions from a 0 to a 1, the corresponding twin-channel group is resynchronized. FMPU_RESYNC2[A:B]=0 on device reset.
	[6]	—		Reserved for future use.
	[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, both channels in the SERDES block 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.

The disadvantage with this scheme is the fact that it is difficult to distribute a 156 MHz reference clock across a backplane. This may require expensive clock driver chips on the board to drive clocks to different destinations within the specified jitter limits for the reference clock.

Figure 38. Distributed Reference Clock to Rx And Tx Devices



High Speed Data Receiver

Table 35 specifies receiver parameters measured on devices with worst case process parameters and over the full range of operation conditions.

Table 35. External Data Input Specifications

Parameter	Conditions	Min.	Typ.	Max.	Units
Input Data					
Stream of Nontransitions	8b/10b encode/decode off	—	—	72	Bits
Sensitivity (differential), worst-case[1]	3.125 Gbps	80	—	—	mVp-p
Input Levels[2]	—	V_{SS} - 0.3	—	V_{DD_ANA} + 0.3	V
Internal Buffer Resistance (Each input to VDDIB)	—	40	50	60	Ω
PLL Lock Time[3]	—	—	—	Note 2	—

1. With PRBS 2^7-1 data pattern, all channels operating, FPGA logic active, REFCLK jitter of 30 ps., T_A = 0ºC to 85ºC, 1.425V to 1.575V supply.
2. Input level min + (input peak to peak swing)/2 ≤ common mode input voltage ≤ input level max - (input peak to peak swing)/2
3. The ORT42G5 and ORT82G5 SERDES receiver performs four levels of synchronization on the incoming serial data stream, providing first bit, then byte (character), then channel (32-bit word), and finally optional multi-channel alignment as described in TN1025. The PLL Lock Time is the time required for the CDR PLL to lock to the transitions in the incoming high-speed serial data stream. If the PLL is unable to lock to the serial data stream, it instead locks to REFCLK to stabilize the voltage-controlled oscillator (VCO), and periodically switches back to the serial data stream to again attempt synchronization.

Input Data Jitter Tolerance

A receiver's ability to tolerate incoming signal jitter is very dependent on jitter type. High speed serial interface standards have recognized the dependency on jitter type and have recently modified specifications to indicate tolerance levels for different jitter types as they relate to specific protocols (e.g XAUI, FC, Infiniband etc.). Sinusoidal jitter is considered to be a worst case jitter type. Table 36 shows receiver specifications with 10 MHz sinusoidal jitter injection. XAUI specific jitter tolerance measurements were measured in a separate experiment detailed in technical note TN1032, *SERDES Test Chip Jitter,* and are not reflected in these results.

Table 36. Receiver Sinusoidal Jitter Tolerance Specifications

Parameter	Conditions	Max.	Unit
Input Data			
Jitter Tolerance @3.125Gbps, Typical	600 mV diff eye[1]	0.75	UIP-P
Jitter Tolerance @3.125Gbps, Worst case	600 mV diff eye[1]	0.65	UIP-P
Jitter Tolerance @2.5Gbps,Typical	600 mV diff eye[1]	0.79	UIP-P
Jitter Tolerance @2.5Gbps, Worst case	600 mV diff eye[1]	0.67	UIP-P

1. With PRBS 2^7-1 data pattern, all channels operating, FPGA logic active, REFCLK jitter of 30 ps., T_A = 0ºC to 85ºC, 1.425V to 1.575V supply. Jitter measured with a Wavecrest SIA-3000.

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.
$\overline{\text{RESET}}$	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	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	$\overline{\text{PRGRM}}$ 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 $\overline{\text{INIT}}$ 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.
$\overline{\text{CFG\_IRQ}}$/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 $\overline{\text{INIT}}$. 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.

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:

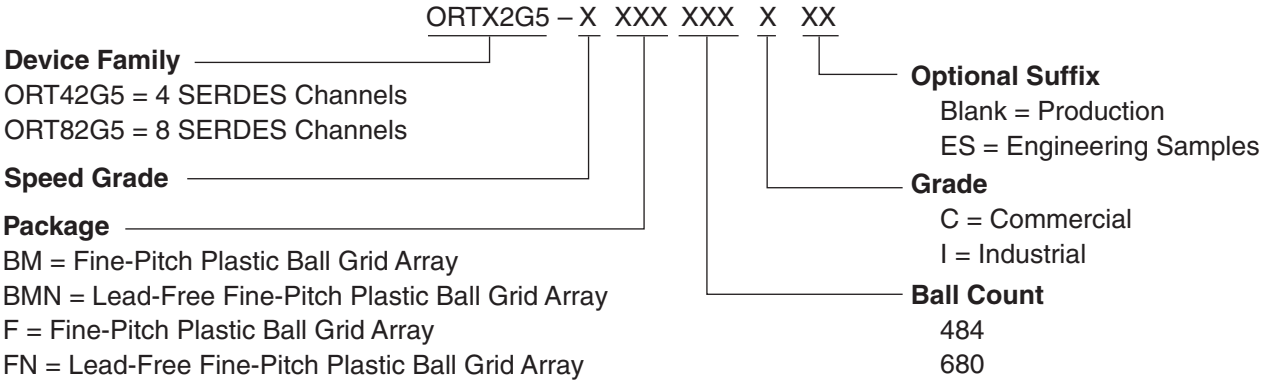
- _A1 indicates one ball between pairs.
- _A2 indicates two balls between pairs.
- _D0 indicates balls are diagonally adjacent.
- _D1 indicates balls are diagonally adjacent, separated by one physical ball.

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

680-PBGAM	VDDIO Bank	VREF Group	I/O	Pin Description	Additional Function	680-PBGAM
Y20	—	—	VSS	VSS	—	—
AG2	7 (CL)	8	IO	PL31B	—	L34C_D0
AH1	7 (CL)	8	IO	PL31A	—	L34T_D0
AF3	6 (BL)	1	IO	PL32D	D8	L1C_A0
AG3	6 (BL)	1	IO	PL32C	VREF_6_01	L1T_A0
AL7	6 (BL)	—	VDDIO6	VDDIO6	—	—
AE4	6 (BL)	1	IO	PL32B	—	L2C_A0
AF4	6 (BL)	1	IO	PL32A	—	L2T_A0
AE5	6 (BL)	1	IO	PL33D	D9	L3C_A0
AF5	6 (BL)	1	IO	PL33C	D10	L3T_A0
R21	—	—	VSS	VSS	—	—
AJ1	6 (BL)	2	IO	PL34D	—	L4C_D0
AH2	6 (BL)	2	IO	PL34C	VREF_6_02	L4T_D0
AM5	6 (BL)	—	VDDIO6	VDDIO6	—	—
AK1	6 (BL)	2	IO	PL34B	—	L5C_D0
AJ2	6 (BL)	2	IO	PL34A	—	L5T_D0
R22	—	—	VSS	VSS	—	—
AG4	6 (BL)	3	IO	PL35B	D11	L6C_D0
AH3	6 (BL)	3	IO	PL35A	D12	L6T_D0
AL1	6 (BL)	3	IO	PL36D	—	L7C_D0
AK2	6 (BL)	3	IO	PL36C	—	L7T_D0
AM9	6 (BL)	—	VDDIO6	VDDIO6	—	—
AM1	6 (BL)	3	IO	PL36B	VREF_6_03	L8C_D0
AL2	6 (BL)	3	IO	PL36A	D13	L8T_D0
AJ3	6 (BL)	4	IO	PL37D	—	—
T16	—	—	VSS	VSS	—	—
AJ4	6 (BL)	4	IO	PL37B	—	L9C_A0
AH4	6 (BL)	4	IO	PL37A	VREF_6_04	L9T_A0
AK3	6 (BL)	4	IO	PL38C	—	—
AN2	6 (BL)	—	VDDIO6	VDDIO6	—	—
AG5	6 (BL)	4	IO	PL38B	—	L10C_A0
AH5	6 (BL)	4	IO	PL38A	—	L10T_A0
AN1	6 (BL)	4	IO	PL39D	PLL_CK7C/HPPLL	L11C_D0
AM2	6 (BL)	4	IO	PL39C	PLL_CK7T/HPPLL	L11T_D0
T17	—	—	VSS	VSS	—	—
AL3	6 (BL)	4	IO	PL39B	—	L12C_D0
AK4	6 (BL)	4	IO	PL39A	—	L12T_D0
T18	—	—	VSS	VSS	—	—
AM3	—	—	I	PTEMP	PTEMP	—
AN3	6 (BL)	—	VDDIO6	VDDIO6	—	—
AJ5	—	—	IO	LVDS_R	LVDS_R	—
AL4	—	—	VDD33	VDD33	—	—
T19	—	—	VSS	VSS	—	—
AK5	—	—	VDD33	VDD33	—	—

Part Number Description



Device Type Options

Device	Voltage
ORT42G5	1.5V internal 3.3/2.5/1.8/ 1.5V I/O
ORT82G5	1.5V internal 3.3/2.5/1.8/ 1.5V I/O

Ordering Information

Conventional Packaging

Commercial[1]

Device Family	Part Number	Speed Grade	Package Type	Ball Count	Grade
ORT42G5	ORT42G5-3BM484C	3	PBGAM	484	C
	ORT42G5-2BM484C	2	PBGAM	484	C
	ORT42G5-1BM484C	1	PBGAM	484	C
ORT82G5	ORT82G5-3F680C	3	PBGAM (No Heat Spreader)	680	C
	ORT82G5-2F680C	2	PBGAM (No Heat Spreader)	680	C
	ORT82G5-1F680C	1	PBGAM (No Heat Spreader)	680	C
	ORT82G5-3BM680C2[2]	3	PBGAM (With Heat Spreader)	680	C
	ORT82G5-2BM680C[2]	2	PBGAM (With Heat Spreader)	680	C
	ORT82G5-1BM680C[2]	1	PBGAM (With Heat Spreader)	680	C

1. For all but the slowest commercial speed grade, the speed grades on these devices are dual marked. For example, the commercial speed grade -2XXXXXC is also marked with the industrial grade -1XXXXXI. The commercial grade is always one speed grade faster than the associated dual mark industrial grade. The slowest commercial speed grade is marked as commercial grade only.
2. BM680 package was converted to F680 via PCN#09A-08.

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.