Lattice Semiconductor Corporation - ORSO42G5-1BMN484C Datasheet

Welcome to E-XFL.COM

Understanding Embedded - FPGAs (Field Programmable Gate Array)

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

Applications of Embedded - FPGAs

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

Details	
Product Status	Obsolete
Number of LABs/CLBs	-
Number of Logic Elements/Cells	10368
Total RAM Bits	113664
Number of I/O	204
Number of Gates	643000
Voltage - Supply	1.425V ~ 3.6V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 70°C (TA)
Package / Case	484-BBGA
Supplier Device Package	484-FPBGA (23x23)
Purchase URL	https://www.e-xfl.com/product-detail/lattice-semiconductor/orso42g5-1bmn484c

setup/hold and clock to out performance.

- New Double-Data Rate (DDR) and zero-bus turn-around (ZBT) memory interfaces support the latest high-speed memory interfaces.
- New 2x/4x uplink and downlink I/O capabilities interface high-speed external I/Os to reduced speed internal logic.
- ispLEVER™ development system software. Supported by industry-standard CAE tools for design entry, synthesis, simulation, and timing analysis.
- Meets Universal Test and Operations PHY Interface for ATM (UTOPIA) levels 1, 2, and 3; as well as POS-PHY3.

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: Embedded Block RAMs, MPI, PCMs, 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 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 ispLEVER Development System.

ORCA 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 supports 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.

Routing

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

Eight fully distributed primary clocks are routed on a low-skew, high-speed distribution network and may be sourced from dedicated I/O pads, PLLs, PLC logic or the Embedded Core. Secondary and edge-clock routing is available for fast regional clock or control signal routing for both internal regions and on device edges. Secondary clock routing can also be sourced from any I/O pin, PLLs, PLC logic or the Embedded Core.

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

System Level Features

The Series 4 also provides system-level functionality by means of its MicroProcessor Interface, Embedded System Bus, block-port Embedded Block RAMs, universal Programmable Phase-Locked Loops, and the addition of highly tuned networking specific Phase-Locked Loops. These functional blocks allow for easy glueless system interfacing and the capability to adjust to varying conditions in today's high-speed networking systems.

MicroProcessor Interface

The MPI provides a glueless interface between the FPGA and *PowerPC* microprocessors. Programmable in 8-, 16-, and 32-bit interfaces with optional parity to the *Motorola® PowerPC* 860 bus, it can be used for configuration and readback, as well as for FPGA control and monitoring of FPGA status. All MPI transactions utilize the Series 4 Embedded System Bus at 66 MHz performance.

The MicroProcessor Interface (MPI) provides a system-level interface, using the system bus, to the FPGA user-defined logic following configuration, including access to the Embedded Block RAM and general logic. The MPI supports burst data read and write transfers, allowing short, uneven transmission of data through the interface by including data FIFOs. Transfer accesses can be single beat (1 x 4 bytes or less), 4-beat (4 x 4 bytes), 8-beat (8 x 2 bytes), or 16-beat (16 x 1 bytes).

System Bus

An on-chip, multimaster, 32-bit system bus with 4-bit parity facilitates communication among the MPI, configuration logic, FPGA control, and status registers, Embedded Block RAMs, as well as user logic. The Embedded System Bus offers arbiter, decoder, master, and slave elements. Master and slave elements are also available for the user-logic and a slave interface is used for control and status of the embedded backplane transceiver portion of the ORSO42G5 and ORSO82G5.

The system bus control registers can provide control to the FPGA such as signaling for reprogramming, reset functions, and PLL programming. Status registers monitor INIT, DONE, and system bus errors. An interrupt controller is integrated to provide up to eight possible interrupt resources. Bus clock generation can be sourced from the MicroProcessor Interface clock, configuration clock (for slave configuration modes), internal oscillator, user clock from routing, or from the port clock (for JTAG configuration modes).

Phase-Locked Loops

Up to eight PLLs are provided on each Series 4 device, with four user PLLs generally provided for FPSCs, including the ORSO42G5 and ORSO82G5. In the FPSCs, these PLLs can only be driven by the FPGA resources. Programmable PLLs can be used to manipulate the frequency, phase, and duty cycle of a clock signal. Each PPLL is capable of manipulating and conditioning clock outputs from 15 MHz to 420 MHz. Frequencies can be adjusted from 1/8x to 8x, the input clock frequency. Each programmable PLL provides two outputs that have different multiplication factors but can have the same phase relationships. Duty cycles and phase delays can be adjusted in

There are sub-modes that can be derived by enabling or disabling certain functions through programmable register bits. Also, in cell mode, either the two-link alignment mode, for up to four alignment groups, or the eight-link alignment mode, where all eight links are combined into a single group, may be selected.

Data are processed in the transmit direction (FPGA to Backplane) as follows:

- In the **SERDES only mode**, there is the option to bypass all of the SONET and cell functions and pass raw 32-bit data from the FPGA into the 32:8 MUX block. In this mode, the user is responsible for providing an adequate ones transition density in the transmitted stream for clock and data recovery at the receive end of the link.
- In the **SONET mode**, a SONET frame is constructed around the input data and overhead bytes are inserted where appropriate. The 32-bits of data per channel are scrambled before being converted to 8-bits by the 32:8 MUX block and serialized by the SERDES.
- In the **cell mode**, 160 bits of data from the FPGA is sent to the Output Port Controller-8 block (for the ORSO82G5 only, the block is also referred to as OPC8 since it services eight links) or 40 bits of data to an Output Port Controller-2 block (referred to as the OPC2) which perform cell striping across the different SERDES links. The cells are then transferred to the SONET clock domain of 77.76 MHz through a Transmit FIFO. A SONET frame is constructed around the cell payload and overhead bytes inserted where appropriate before being sent to the MUX block. The data are then converted to 8 bits by the 32:8 MUX block before being serialized by the SERDES.

Data are processed in the receive direction (Backplane to FPGA) as follows:

- In the **SERDES only mode**, there is the option to bypass all of the SONET and cell functions and pass raw 32-bit data from the 8:32 DEMUX block into the FPGA interface.
- In the **SONET mode**, the descrambled data are sent to an alignment FIFO that performs lane-to-lane deskew and aligns data within an alignment group to a single clock domain and frame pulse. The SPE indicator is provided to the FPGA along with 32 bits of aligned data.There is an option to bypass the alignment FIFOs and pass data directly from the descrambler to the FPGA. This mode is programmable and can be controlled per channel.
- In the **cell mode**, the SONET framed data are descrambled and passed into a cell extractor which extracts cells from the payload portion of the SONET frame. The cells are passed through a FIFO which performs lane-to-lane deskew and a clock domain transfer from the SONET clock domain to the cell processing domain. The cells are passed into the input port controller block (referred to as IPC8 or IPC2 depending on whether eight or two links are serviced) which performs cell destriping before sending them to the FPGA interface. This cell processing feature makes the ORSO42G5 and ORSO82G5 ideal for interfacing devices with proprietary data formats across a backplane.

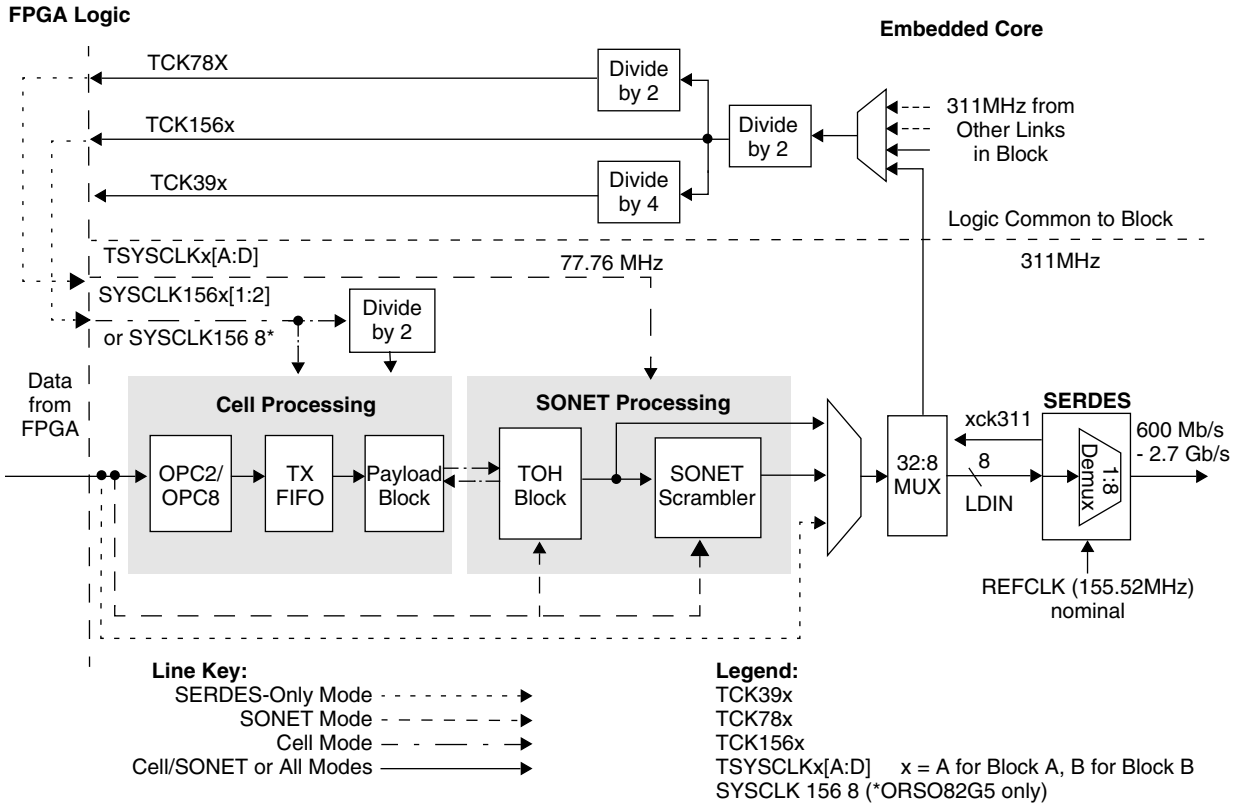
Embedded Core Functional Blocks - Overview

Each channel contains transmit path and receive path logic as shown in Figure 2. Data are processed on a channel by channel basis in the SERDES only and SONET modes. Channel by channel processing is also performed in the cell mode by the Input Port Controller (IPC) and Output Port Controller (OPC) blocks. Support for loopback is also provided but is not shown in Figure 2. The following sections will give an overview of the pseudo SONET protocol supported by the ORSO42G5 and ORSO82G5 and a top level overview of the macrocells, which provide the SERDES Only, SONET and cell mode functionality.

SERializer and DESerializer (SERDES)

Each SERDES block is a block transceiver containing two or four channels for serial data transmission, with a per-channel selectable data rate of 0.6-2.7 Gbps. Each SERDES block features high-speed CML interfaces and is designed to operate in SONET backplane applications. The transceiver is controlled and configured via an 8-bit slave interface on the system bus. Each channel has dedicated registers that are readable and writable. The device also contains global registers for control of common circuitry and functions. There are two SERDES blocks, A and B, in the embedded portion of the device. Each block supports full duplex serial links in the ORSO82G5 (Slice A contains channels AA, AB, AC, and AD while slice B contains channels BA, BB, BC, and BD). A similar naming

Figure 3. Top Level Overview, TX Path Logic, Single Channel



Receiver Architecture

The receiver section receives high-speed serial data at the external differential CML input pins. These data are fed to the clock recovery section which generates a recovered clock and retimes the data. Therefore the receive clocks are asynchronous between channels. The data are then optionally framed, reformatted, aligned and passed to the FPGA logic in various parallel data formats.

The top level receiver architecture is shown in Figure 4. The main logical blocks in the receive path are:

- Receive SERDES and 8:32 DEMUX.
- SONET processing logic.
- Input Port Controllers (IPCs) which contain the cell processing logic.

Depending on the mode of operation, the FPGA to backplane data path may include or bypass the various logical blocks.

of differential 0.6 to 2.7 Gbit/s links. At the FPGA/Embedded Core interface, the data are transferred across 32-bit buses. The SERDES blocks themselves are organized as two blocks. Each of the data paths is identified with a block and channel identifier (i.e., AC, AD, BC, BD or AA,...,BD).

Each channel has a 32-bit TX bus, 32-bit RX bus, a recovered clock, a transmit clock input and a transmit start signal.

Figure 5. Basic Data Flows - SERDES Only Mode

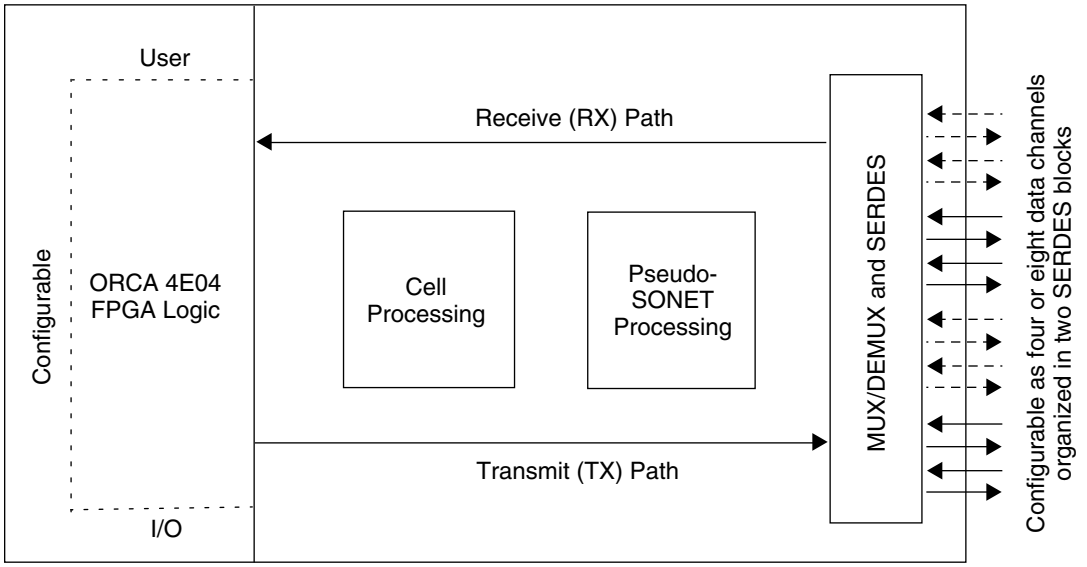


Figure 6 shows a block diagram of a single channel of the SERDES block. The transmitter section accepts either scrambled or un-scrambled data at the parallel input port. It also accepts the low-speed reference clock at the REFCLK input and uses this clock to synthesize the internal high-speed serial bit clock. The serialized data are available at the differential CML output terminated in 86 Ω to drive either an optical transmitter, coaxial media or a circuit board/backplane.

Figure 6. SERDES Functional Block Diagram for One Channel

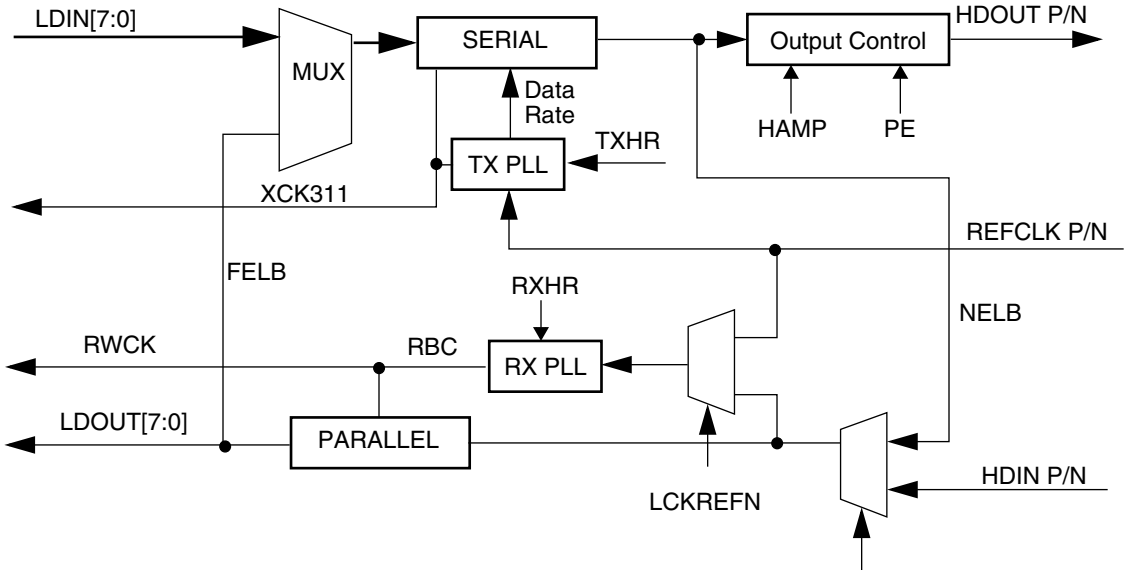
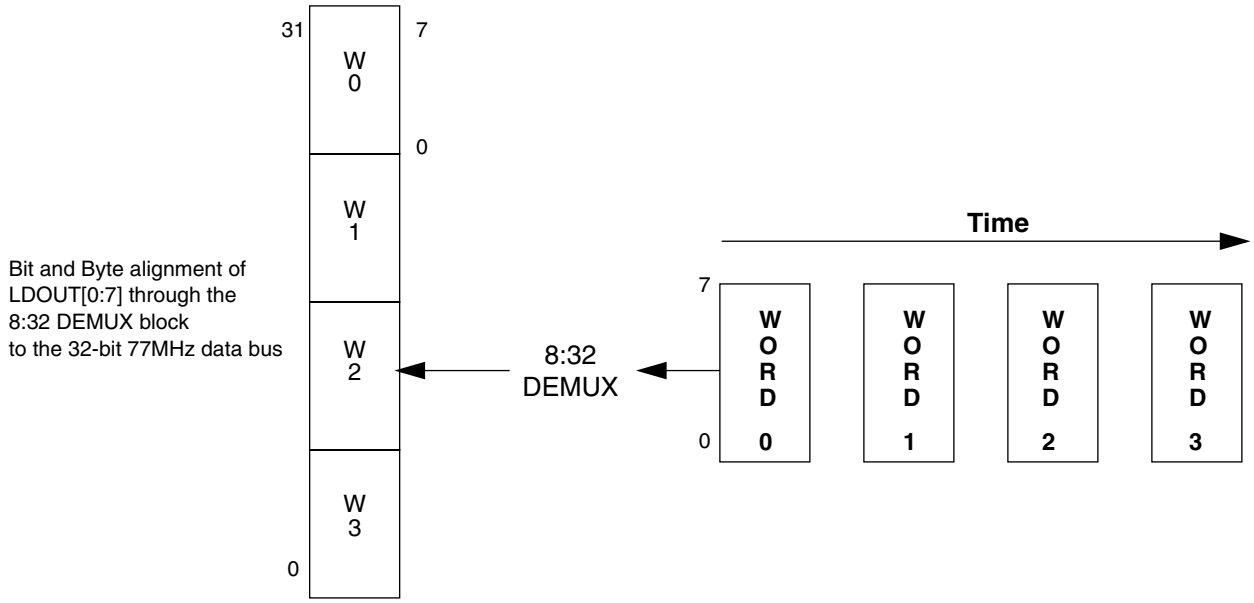


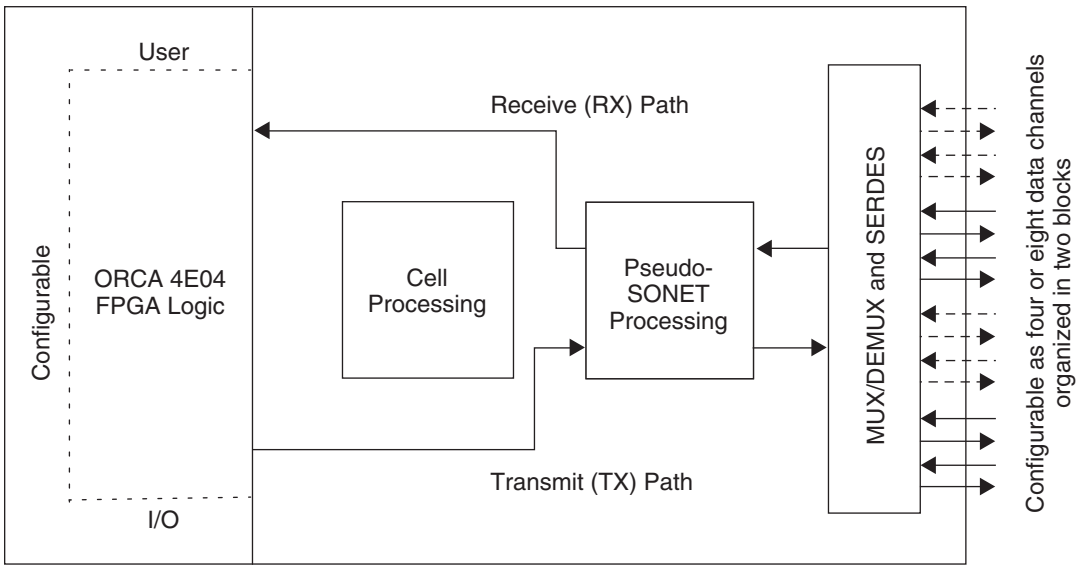
Figure 10. Bit and Byte Alignment for DEMUX Block



SONET Mode Operation – Detailed Description

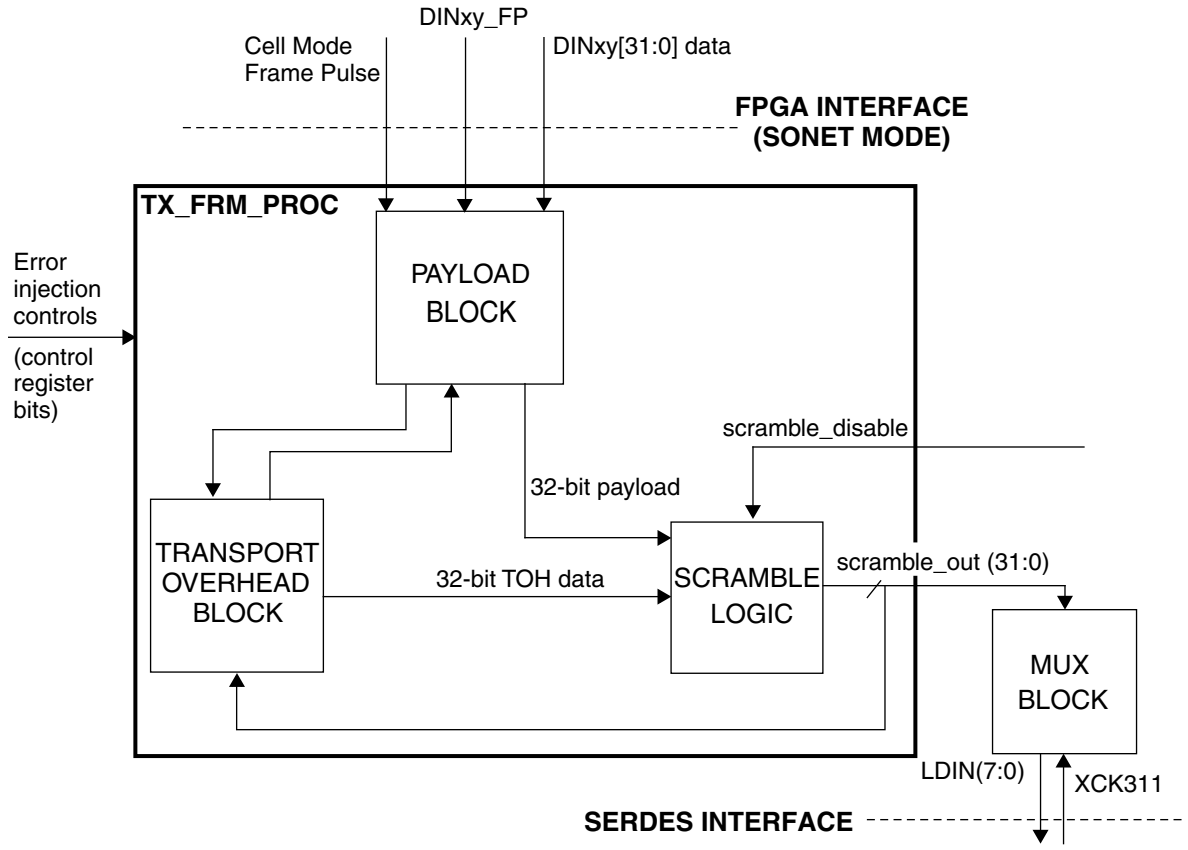
The following sections describe the data processing performed in the SONET logic blocks. The basic data flows in the SONET Mode are shown in Figure 11. At a top level, the descriptions are separated into processing in the transmit path (FPGA to serial link) and processing in the receive path (serial link to FPGA). In general, the descriptions in the next sections are written to describe SONET mode operation, although some of the "SONET logic blocks" are also used in cell mode operation. The various processing options are selected by setting bits in control registers and status information is written to status registers. Both types of registers can be written and/or read from the System Bus. Memory maps and descriptions for the registers are given in Table 21 through Table 36.

Figure 11. Basic Data Flows - SONET Mode



In the SONET mode, the transmit block receives 32-bit wide data from the FPGA (DINxx) on each of its channels along with a frame pulse (DINxx_FP) per channel and a transmit clock (TSYCLKxx). Typically this will represent a STS-48 stream on each link. The data are first passed through a TOH block which will generate all the timing pulses that are required to isolate individual overhead bytes (e.g., A1, A2, B1, D1-D3, etc.). The timing pulse gener-

Figure 16. TX Frame Processor (TFP) Block Diagram



Payload Sub-block

The Payload sub block is activated by the cell mode frame pulse (cell mode) or DINxx_FP from FPGA (SONET mode). A pulse on this signal indicates the start of a frame.

In SONET mode, only two types of data bytes are in each frame:

- TOH bytes
- SPE data bytes

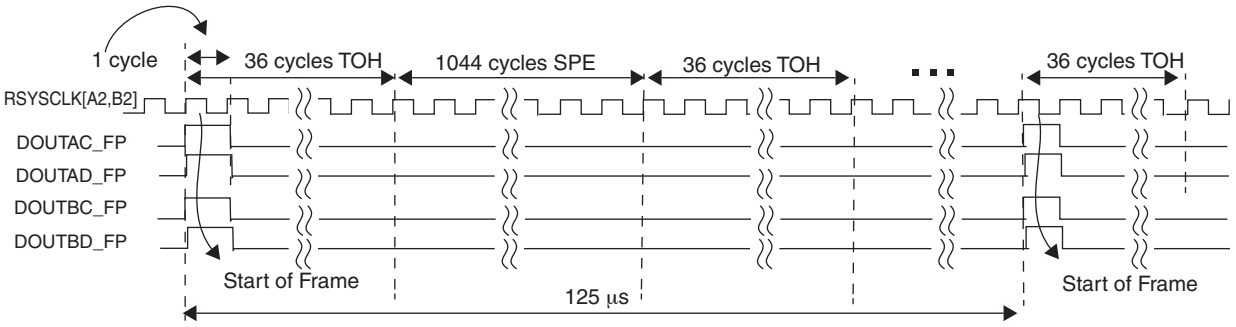
There are N x 3 (N = 48) bytes of TOH per row and there are a total of 9 rows in a SONET frame. The rest of the bytes in each row are SPE data bytes in SONET mode.

TOH Sub-block

This block is responsible formatting the 144 (48 x 3) bytes of TOH at the beginning of each row of the transport frame. All TOH bytes may be transmitted transparently from the FPGA logic using the transparent mode. Alternately, some or all TOH bytes may be inserted by the TOH block (AUTO_SOH and AUTO_TOH mode). The TOH data is transferred across the FPGA/core interface as 32-bit words, hence 36 clock cycles (12 x 3) are needed to transfer a TOH row. TOH insertion is controlled by software register bits as shown in the Register Map tables.

Figure 31 shows the quad alignment mode in the ORSO42G5.

Figure 31. Receive SONET Mode, Quad Alignment Mode – ORSO42G5



Clocks
RSYSCLKA2 and RSYSYSCLKB2 are sourced by RCK78A

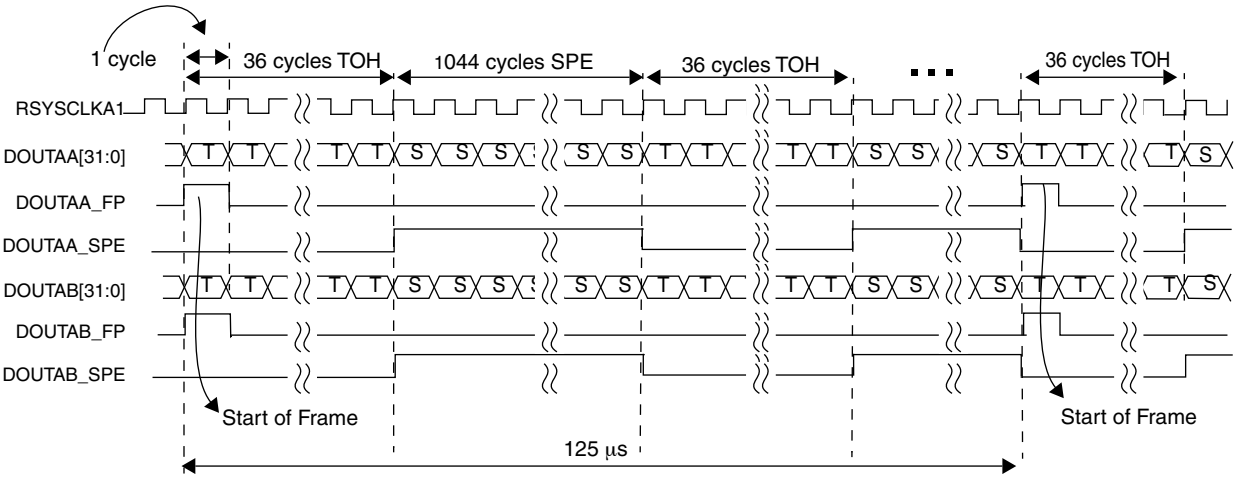
SONET Mode Receive Timing – ORSO82G5

This section contains timing diagrams for major interfaces of this block to the FPGA logic when SONET frames are to be transferred.

- When operating in SONET mode, the entire SONET frame is sent to the FPGA. In multi-channel alignment mode(s), data from all channels within an alignment group are aligned to the A1A2 framing bytes.
- Each SONET frame is 125µs. The frame starts with 36 clock cycles (77.76 MHz) of TOH followed by 1044 clock cycles of SPE, followed by 36 clock cycles of TOH, 1044 cycles of SPE.
- The DOUTxx_SPE signal indicates TOH or SPE in the data (low for TOH, high for SPE)
- Twin pairs are AA, AB (group A1), AC, AD (group A2), BA, BB (group B1) and BC, BD (group B2)

Figure 32 shows the SONET twin alignment mode timing for the ORSO82G5. The frame pulse and SPE indicators are show for each of the two channels (AA, AB) in twin alignment.

Figure 32. Receive Clocking Diagram for SONET Mode Twin Alignment in Block A – ORSO82G5



Clocks
RSYSCLKA1 is the read clock used for group A1
RSYSCLKA2 is the read clock used for group A2
RSYSCLKB1 is the read clock used for group B1
RSYSBLKB2 is the read clock used for group B2

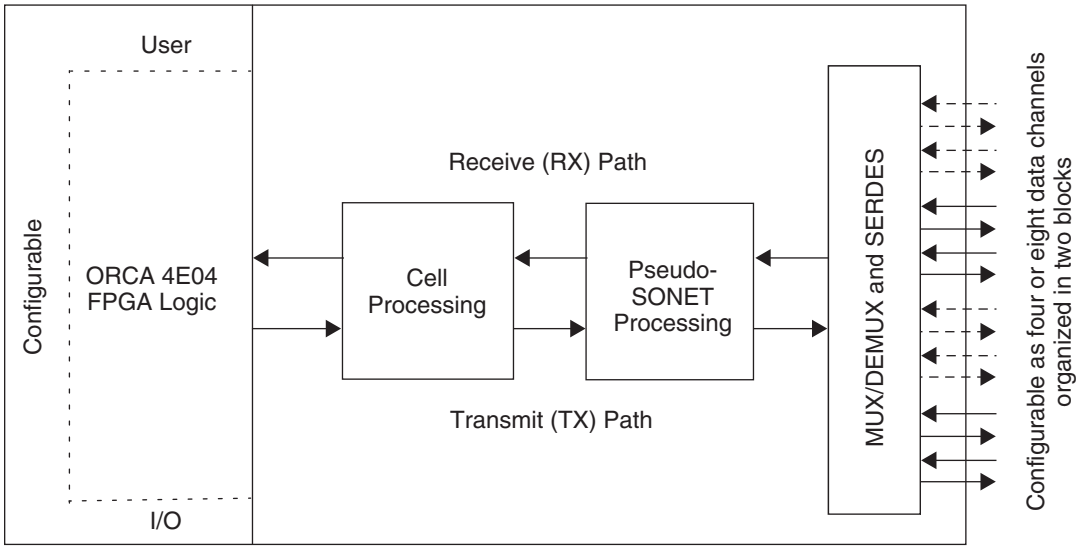
Data
T Represents TOH
S Represents SPE
DOUTxx-SPE is high for SPE, low for TOH

Cell Mode Detailed Description

A common application for the ORSO42G5 and ORSO82G5 is to provide a bridge between a port card and a cell-based switch fabric. In cell mode, the data in the Synchronous Payload Envelope (SPE) of the SONET frames is further formatted into fixed-length cells by the ORSO42G5 and ORSO82G5. The cell contents will typically be unique to specific port card and switch devices. The ORSO42G5 and ORSO82G5 supports this application with a "cell mode" of operation

The basic data flows in cell mode are shown in Figure 35. Data to be transmitted is received from the FPGA logic (see Table 11 and Table 12 for details of the core/FPGA signal assignments in the transmit direction which differ significantly from the SERDES only and SONET modes), inserted into the SPE of the SONET frame, scrambled and transmitted from the SERDES block. In cell mode, multiple SERDES links are used to achieve desired bandwidth. A two-link mode is supported in the ORSO42G5 and both two-link and eight-link cell modes are supported. For such interfaces, data are cell-striped in a round-robin fashion across multiple links by the transmitter.

Figure 35. Basic Data Flows - Cell Mode



In the receive direction, the framed data are received from the SERDES block, descrambled and are passed into a cell extractor which extracts individual cells from the payload portion of the SONET frame. The cells are then passed through a FIFO that performs lane-to-lane deskew and a clock domain transfer. The clock domain transfer is handled automatically using idle cell insertion and deletion.

The cells are passed into either the eight-link Input Port Controller IPC8 block (ORSO82G5 only) or to one of the two-link IPC2 block(s), which reassemble the cells back into a single cell stream (destriping) which is sent to the FPGA logic. (See Table 13 and Table 14 for details of the core/FPGA signal assignments in the receive direction. As with the transmit path, the cell mode assignments differ significantly from those for the SERDES only and SONET modes).

SERDES and SONET processing has been described in previous sections and only features unique to the cell mode will be discussed in the following sections. The cell format will be discussed first, followed by a description of the transmit path, which will include either a two-link or an eight-link Output Port Controller (OPC) block, and a description of the receive path, including the two-link or eight-link Input Port Controller (IPC) blocks.

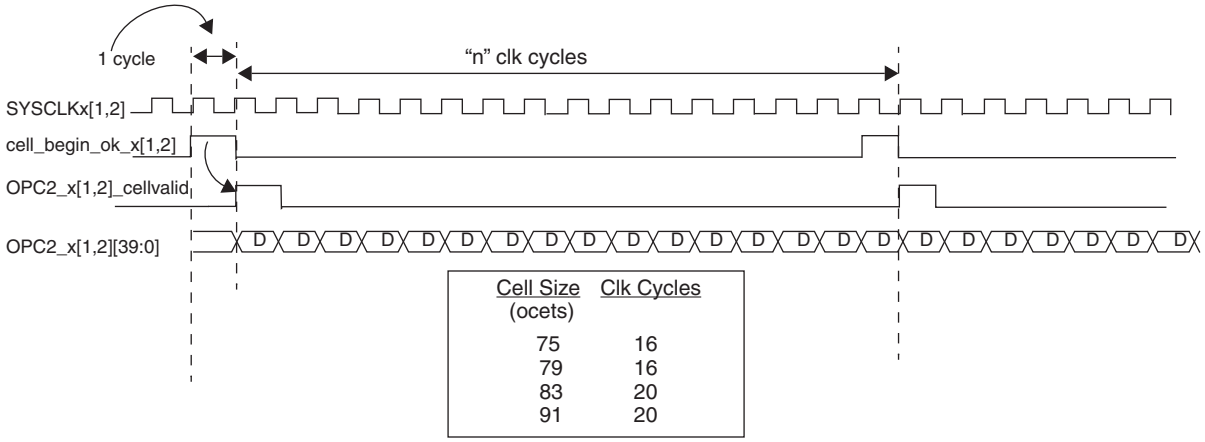
Cell Formats

Cells are arranged within a SONET (STS-48c) frame as shown in Figure 36. A SONET STS-48c frame has 4176 (87 x 48) columns of SPE and 9 rows that gives a total of 37,584 bytes. In this implementation, data in a SPE is limited to fixed size cells. Though four cell sizes are supported, only one cell size can be used at a time.

- If core FIFO cannot accept cells, cell_begin_ok will be low.
- If core FIFO is empty then cell_begin_ok will be asserted every 4 clock cycles until cellvalid is asserted by user to indicate valid cell data.

cellvalid: Clock-wide pulse asserted by user to indicate valid data. Asserted on the clock cycle following cell_begin_ok.

Figure 41. ORSO42G5 and ORSO82G5 Transmit FPGA Interface OPC2 Cell Mode

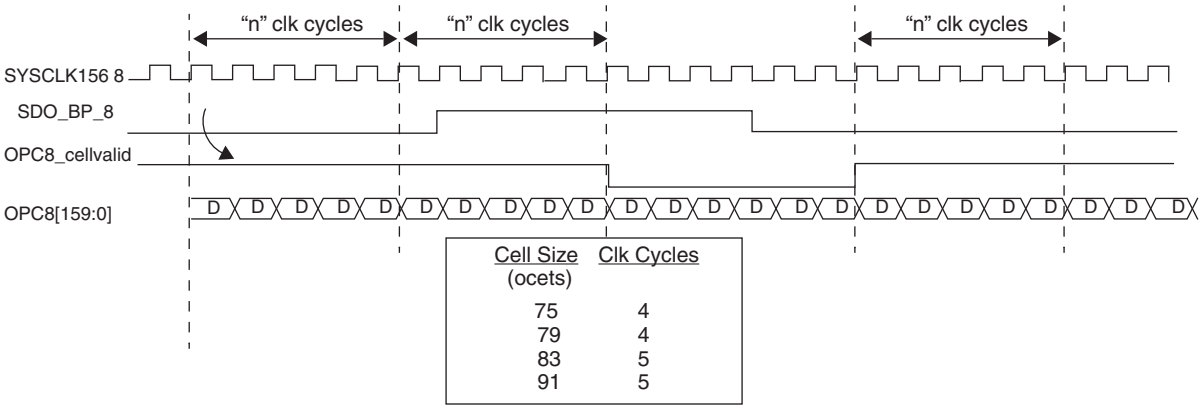


When operating in the eight-link cell mode, the OPC8 block passes user cells from FPGA to embedded core. Depending upon the configured CELL SIZE, cell transfers will take a variable number of SYSCLK156 cycles to be transmitted across the interface. Data are always transferred across a 160-bit bus (20 octets per clock cycle). Figure 42 shows five clock cycles for a cell transfer this corresponds to a user cell size of 91 octets. The two control signals in the figure are defined as:

sdo_bp_8: Backpressure signal from core instructing user to stop sending cell data. User should complete transmitting the current cell and can send one more cell before deasserting cellvalid.

cellvalid: Is high throughout a cell transfer to indicate valid cell data

Figure 42. ORSO82G5 Transmit FPGA Interface OPC8 Cell Mode



Link Header Detector

The Link Header detector determines when the next Link Header is coming in the frame and what the sequence number contained within it should be. If the value of the cell sequence counter is not equal to the expected value, then an error flag bit will set in the status registers and an error signal will be sent to the FPGA logic. The sequence counter will increment to the next sequence number. The sequence count value is NOT updated with the incorrect value, but is incremented each time a Link Header is received.

If excessive sequence errors are detected (three or more in a row), and the AUTO_REMOVE_[A:B] register bit is set, then the corresponding link will be treated as not valid. The RX_LINK_GOOD status bit will go low indicating the link is no longer receiving cells. If the AUTO_REMOVE_[A:B] register bit is not set, then the link is still valid when excessive sequence errors are detected. An OOF condition will also trigger the link to be removed from service if the AUTO_REMOVE_[A:B] register bit is set.

At startup or after a link has been removed from service (indicated by RX_LINK_GOOD going low), a link can be rejoined into the group. This is performed via the per block REJOIN_[A:B] register bit. When rejoining a link the RX FIFO will begin receiving cells. To cleanly rejoin a link into a group there are two methods to insure the RX FIFO begins loading correctly. The first method is to use the fast framing mode during the rejoin process. The can be done by setting the FFRM_EN_xx for the links that need to be rejoined before setting the REJOIN_[A:B] bit.

The second method is to issue a block reset to clear the FIFO once all links that have been selected to be rejoined are rejoined. This is done by first setting the REJOIN_[A:B] bit. Once the RX_LINK_GOOD status bit is high for the selected channels the GSWRST_[A:B] for the block should be set and cleared to reset the block. This method will disturb traffic on all links in the block during the GSWRST_[A:B] reset time.

Once all of the links in a group are rejoined and the traffic is again flowing the REJOIN_[A:B] bit should be cleared. If this bit is not cleared, a link may drop out using the AUTO_REMOVE mode and the channel may be rejoined incorrectly, causing errors on the entire group.

Receive FIFO

The main clock domain transfer for the data path is handled by the receive FIFO. A 16 x 161 FIFO is used in cell mode. The FIFO is implemented as a dual-port memory which will support simultaneous reads and writes. The receive FIFO block is written to at 77.76 MHz and read at 156 MHz.

The receive FIFO can allow for inter-link skew of about 800 ns (16 x 160 = 2560 bits, 400 ps per bit gives 1024 ns). The 160 LSBs in the memory are received data and the 161st bit indicates the start of a new cell. The FIFO write control logic indicates to the IPC, the start of a new frame of data. This signal will only be active for the A1 word of a frame.

Once frame synchronization has occurred and the IPC has responded with a FIFO enable signal, data will be written into the memory. Only the payload (cells) is written to the FIFO. The TOH bytes are not written into the FIFO. The cell octets immediately following the A1A2 bytes will be always written to the top of the FIFO.

Once a full cell has been written to the memory, the write control logic will send a control signal to the IPC8 or IPC2 block which will start the process of reading data from the FIFO. The IPC will read one whole cell at a time from each of the 8 FIFO blocks, if configured for the eight-link cell mode (ORSO82G5 only) or from each of 2 FIFO blocks if configured for the two-link cell mode.

A FIFO occupancy counter generates a RX_FIFO_OVRUN indication to the register interface if it detects a FIFO overflow condition. The cell mode allows for alignment of all eight-links or alignment of two-links. Thus there will be two IPC blocks for two pairs of channels per block.

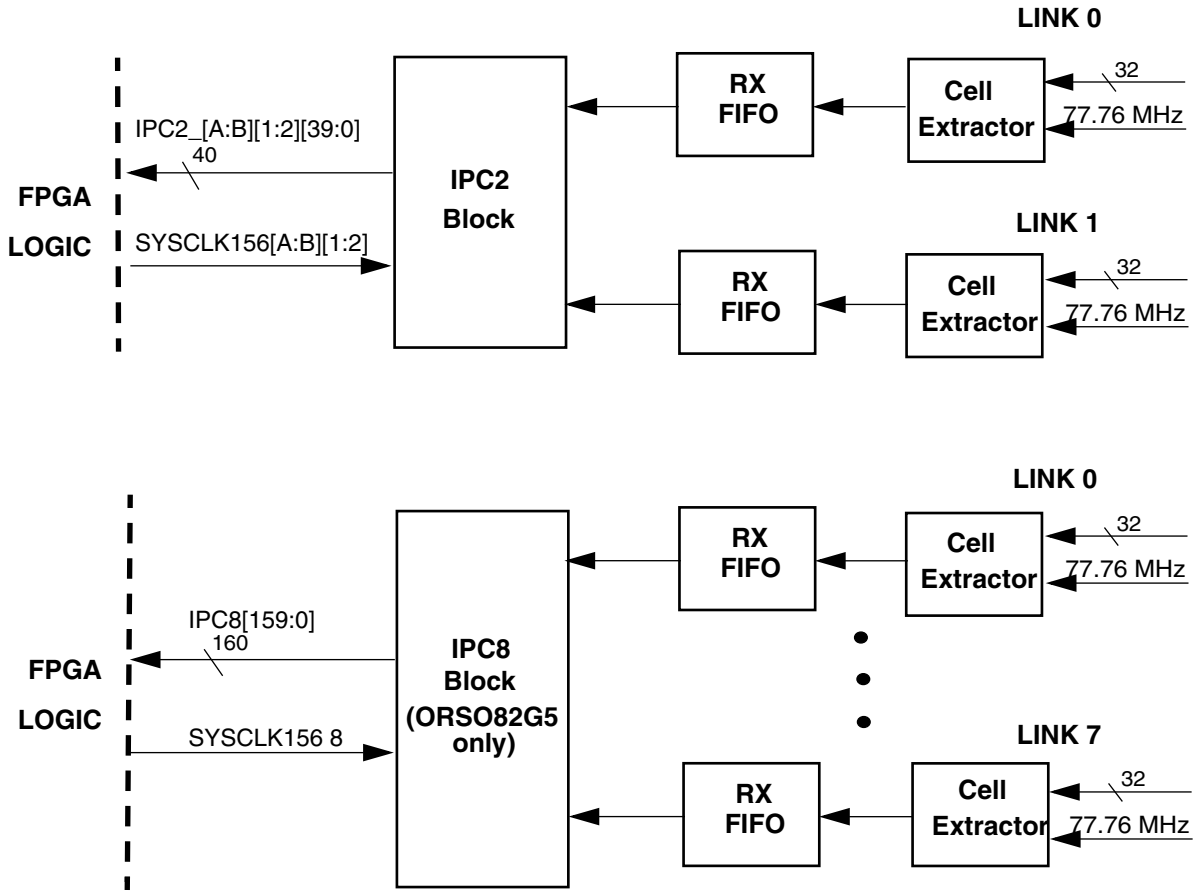
Input Port Controllers

The input port controllers (IPCs) are the block responsible for “directing traffic” for the receive traffic flow. The block diagrams for the 2-link and 8-link IPCs are shown in Figure 44. They provide the following essential functions.

- Determining when cell data can be read from the FIFOs of the individual links.

- Insuring group bundles are properly aligned.
- Scheduling reads from the RX FIFOs. Cells are read one at a time from the configured links.
- Parsing the cell data into payload data (along with selected header information). Cells which have errors that make them unusable (such as BIP or sequence number errors) are thrown away. This dropping of errored cells can be disabled through register bits CELL_BIP_INH_xx and CELL_SEQ_INX_xx.

Figure 44. IPC2 and IPC8 Block Diagrams



There are 5 IPC blocks in the embedded core. There is an IPC2 block for every channel pair:

- IPC2_A1 combines links from channels AA,AB (ORSO82G5 only)
- IPC2_A2 combines links from channels AC,AD
- IPC2_B1 combines links from channels BA,BB (ORSO82G5 only)
- IPC2_B2 combines links from channels BC,BD

The IPC8 block combines cells from all eight aligned links and transmits them to the FPGA logic (ORSO82G5 only).

Before an IPC can begin reading data from the Rx FIFOs and assembling cells, it must first align all FIFOs in a port bundle. This is accomplished by handshaking signals between the framer and the IPC. The framer indicates to the IPC that framing has been acquired. The framer does not start filling the FIFOs, however, until the next A1/A2 SONET signal.

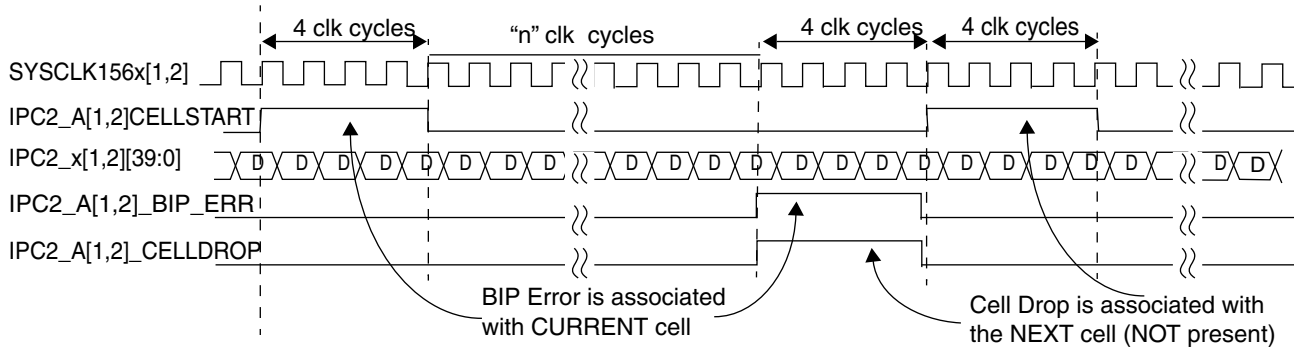
The behavior of the IPC is dependent on the AUTO_BUNDLE register bit. If AUTO_BUNDLE is set, the group will continue to operate even if a link (or several links) of the group is not valid (RX_LINK_GOOD is low). If AUTO_BUNDLE is not set the entire group must be valid (RX_LINK_GOOD is high) for the group to receive cells through the IPC.

The IPC must determine when FIFO reads may begin. Before reading data from a FIFO can begin, the FIFO must have a full cell available to be read. This is condition is indicated by a signal from each FIFO which is monitored by the IPC. The IPC then makes sure that the cells in a given port are received in the order that they are transmitted.

IPC Receive Cell Mode Timing Core/FPGA

This section contains timing diagrams for major interfaces of this block to the FPGA logic when cells are to be transferred. Figure 45 shows the cell twin-link mode timing. The number of clock cycles to transfer the cell data depends on the payload size selected. Error indications for CELL BIP errors and CELL DROP are also shown.

Figure 45. IPC2 Data Flow



CELL BIP ERROR
If a Cell BIP Error occurs,
the CELL_BIP_ERR signal reflects the occurrence, as shown in the Figure.
For 2-Link CELL MODE,
the CELL_BIP_ERR signal is asserted during the last 4 clock cycles of the receive cell.

CELL BIP ERROR
If a cell error occurs within the ASB and;
1. CELL_BIP_INH=0 ...Do not drop BIP errored cells (s/w selectable)
2. A BIP error occurs

The drop indicator will PRECEED the user cell that contains the BIP error. All data will be passed w/o modification.

When operating in CELL MODE, the IPC2 Block passes user cells as well as control and status signals to the user. Depending upon the configured CELL SIZE, cell transfers will take a variable number of SYSCLK156 cycles to be received across the interface. Data are always transferred across a 40-bit bus (5 octets per clock cycle). Figure 45 shows 16 clock cycles for a cell transfer. This corresponds to a User Cell size of 79 octets.

Figure 46 shows cell octal alignment mode timing for the ORSO82G5. When operating in CELL MODE, the IPC8 Block aligns all 8 channels of receive data on a FRAME basis. The IPC8 also passes user cells as well as control and status signals to the user. Depending upon the configured CELL SIZE, cell transfers will take a variable number of SYSCLK156 cycles to be received across the interface. Data are always transferred across an 160-bit bus (20 octets per clock cycle). Figure 46 shows 4 clock cycles for a cell transfer. This corresponds to a User Cell size of 79 octets.

Reference Clock Requirements

There are two pairs of reference clock inputs in the ORSO42G5 and ORSO82G5 devices. Each 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 46.

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 ±100 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 2.7 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 1/8th the bit rate of the data lane. Additionally, the MUX/DEMUX logic converts the data rate and bit-width so the FPGA core can run at 1/4th this frequency which gives a range of 18.8 to 84.4 MHz for the data in and out of the FPGA in SONET mode. In cell mode, there is a clock domain transfer to a 2x clock domain, which gives a range of 37.5 to 168.8 MHz for the data in and out of the FPGA.

Internal Clock Signals at the FPGA/Core Interface

There are several clock signals defined at the FPGA/Embedded Core interface in addition to the external reference clock for each SERDES block. All of the ORSO42G5 and ORSO82G5 clock signals are shown in Figure 47 and are described following the figure.

Table 17. Decoding of SCHAR_CHAN

SCHAR_CHAN0	SCHAR_CHAN1	Channel
0	0	BA
1	0	BB
0	1	BC
1	1	BD

The receive characterization test mode is entered when SCHAR_ENA=1 and SCHAR_TXSEL=0, In this mode, one of the channels of SERDES outputs is observed at chip ports as shown in Table 18. The channel that is observed is also based on the decoding of SCHAR_CHAN as shown in Table 18.

Table 18. SERDES Receive Characterization Mode

SERDES Output	Chip Port
LDOUTBx[9:0]	PSCHAR_LDIO[9:0]
RBC0Bx	PSCHAR_CKIO0
RBC1Bx	PSCHAR_CKIO1

Embedded Core Block RAM

There are two independent memory slices (labeled A and B) in the embedded core. Each memory slice has a capacity of 4K words by 36 bits. These are in addition to the block RAMs found in the FPGA portion of the ORSO42G5 and ORSO82G5. Although the memory slices are in the embedded core part of the chip, they do not interact with the rest of the embedded core circuits, but are standalone memories designed specifically to increase RAM capacity in the ORSO42G5 and ORSO82G5 chip. They can be used by the soft IP cores implemented in the FPGA portion of the FPSC.

A block diagram of a memory slice is shown in Figure 48. Each memory slice is organized into two sections (labeled SRAM A and SRAM B) and has one read port, one write port and four byte-write-enable (active-low) signals. Each byte has eight data bits and a control/parity bit. The control/parity bit responds to the same byte enable (BYTEWN_x[x]) as it's corresponding data. No special logic such as parity checking is performed on this bit by the core. The read data from the memory is registered so that it works as a pipelined synchronous memory block. The minimum timing specifications are shown in Figure 49 and Figure 50. Signal names and functions are summarized later in Table 19 and follow the general Series 4 naming conventions.

Table 26. Per-Channel Control Register Descriptions – ORSO42G5 (Continued)

(0x) Absolute Address	Bit	Name	Reset Value (0x)	Description	Mode
30823 - AC 30833 - AD 30923 - BC 30933 - BD	[0]	BYPASS_ALGN_FIFO_xx	00	Bypass alignment FIFO =1 in RX path in SONET mode. DOUT_xx data is clocked off RWCKxx.	SONET
	[1]	SERDES_ONLY_MODE_xx		SERDES-Only Mode. SERDES_ONLY_MODE =1 bypasses all the SONET and cell functions. Data is sent directly from the SERDES to the FPGA logic in the RX path and is passed directly from the FPGA logic to the SERDES in the TX path.	SERDES Only
	[2]	FORCE_B1_ERR_xx		Force B1 Error. FORCE_B1_ERR =1 forces a Section B1 error (Bit Interleaved Parity Error) in the TX path. Valid only when the corresponding AUTO_TOH_xx bit or AUTO_B1_xx bit is set to one.	Both
	[3]	FORCE_BIP8_ERR_xx		Force BIP8 Error, FORCE_BIP8_ERR =1 forces cell BIP8 error in the TX path.	Cell
	[4]	FORCE_A1A2_ERR_xx		Force A1A2 Error, FORCE_A1A2_ERR =1 forces an error in A1A2 bytes (framing error) in the TX path. Valid only when the corresponding AUTO_TOH_xx bit or AUTO_A1A2_xx bit is set to one.	Both
	[5]	FORCE_EX_SEQ_ERR_xx		Force Excessive Sequence Errors. FORCE_EXP_SEQ_ERR_xx = 1 forces sequence errors in the TX path by inverting the link header byte sequence number.	Cell
	[6]	FORCE_SEQ_ERR_xx		Force Sequence Error, FORCE_SEQ_ERR =1 forces one sequence error in the TX path. After this bit is set, the next Link Header byte's sequence number is inverted. The Link Header after the errored Link Header byte will have the correct sequence number	Cell
	[7]	FORCE_RDI_xx		Force Remote Defect Indication, FORCE_RDI =1 forces RDI errors in the TX path. The K2 byte in TOH is set to "00000110. Valid only when AUTO_TOH_xx bit is set to 1.	Cell

Table 27. Per-Channel Status Register Descriptions – ORSO42G5 (Continued)

(0x) Absolute Address	Bit	Name	Reset Value (0x)	Description	Mode
3082A - AC 3083A - AD 3092A - BC 3093A - BD	[0:4]	RSVD	00	Reserved	—
	[5]	STAT_CELL_ALIGN_ERR_xx		STAT_CELL_ALIGN_ERR_xx = 1 same as CELL_ALIGN_ERR_xx except that a 1 on this bit can NOT cause an interrupt	Cell
	[6]	STAT_TX_URUN_ERR_xx		STAT_TX_URUN_ERR_xx = 1 same as TX_URUN_ERR_xx except that a 1 on this bit can NOT cause an interrupt	Cell
	[7]	STAT_TX_ORUN_ERR_xx		STAT_TX_ORUN_ERR_xx = 1 same as TX_ORUN_ERR_xx except that a 1 on this bit can NOT cause an interrupt	Cell
3082B -AC 3083B - AD 3092B - BC 3093B - BD	[0]	RSVD	00	Reserved	—
	[1]	STAT_OOF_xx		STAT_OOF_xx = 1 same as OOF_xx except that a 1 on this bit can NOT cause an interrupt	Both
	[2]	STAT_EX_SEQ_ERR_xx		STAT_EX_SEQ_ERR_xx = 1 same as EX_SEQ_ERR_xx except that a 1 on this bit can NOT cause an interrupt	Cell
	[3]	STAT_SEQ_ERR_xx		STAT_SEQ_ERR_xx = 1 same as SEQ_ERR_xx except that a 1 on this bit can NOT cause an interrupt	Cell
	[4]	STAT_CELL_BIP_ERR_xx		STAT_CELL_BIP_ERR_xx = 1 same as CELL_BIP_ERR_xx except that a 1 on this bit can NOT cause an interrupt	Cell
	[5]	STAT_B1_ERR_xx		STAT_B1_ERR_xx = 1 same as B1_ERR_xx except that a 1 on this bit can NOT cause an interrupt	Both
	[6]	STAT_RX_FIFO_OVRUN_xx		STAT_RX_FIFO_OVRUN_xx = 1 same as RX_FIFO_OVRUN_xx except that a 1 on this bit can NOT cause an interrupt	Cell
	[7]	STAT_RDI_xx		STAT_RDI_xx = 1 same as RDI_xx except that a 1 on this bit can NOT cause an interrupt	Both
3082C - AC 3083C - AD 3092C - BC 3093C - BD	[0:7]	LINK_NUM_RX_xx	00	Link number received that is stored in the F1 byte position of the SONET TOH	Both
3082E - AC 3083E AD 3092E- BC 3093E - BD	[0:5]	RSVD	00	Reserved	—
	[6]	CH248_SYNC_xx		CH248_SYNC_xx = 1 indicates that A1A2 bytes have been detected by link xx for multi-channel alignment	SONET
	[7]	RX_LINK_GOOD_xx		RX_LINK_GOOD_xx = 1 indicates that frame has been acquired and the link has been stable. Goes high at an A1A2 boundary but can go low at any point in a frame due to excessive errors	Cell

Table 34. Per-Channel Control Register Descriptions – ORSO82G5 (Continued)

(0x) Absolute Address	Bit	Name	Reset Value (0x)	Description	Mode
30803 - AA 30813 - AB 30823 - AC 30833 - AD 30903 - BA 30913 - BB 30923 - BC 30933 - BD	[0]	BYPASS_ALGN_FIFO_xx	00	Bypass alignment FIFO =1 in RX path in SONET mode. DOUT_xx data is clocked off RWCKxx.	SONET
	[1]	SERDES_ONLY_MODE_xx		SERDES-Only Mode. SERDES_ONLY_MODE =1 bypasses all the SONET and cell functions. Data is sent directly from the SERDES to the FPGA logic in the RX path and is passed directly from the FPGA logic to the SERDES in the TX path.	SERDES Only
	[2]	FORCE_B1_ERR_xx		Force B1 Error. FORCE_B1_ERR =1 forces a Section B1 error (Bit Interleaved Parity Error) in the TX path. Valid only when the corresponding AUTO_TOH_xx bit or AUTO_B1_xx bit is set to one.	Both
	[3]	FORCE_BIP8_ERR_xx		Force BIP8 Error, FORCE_BIP8_ERR =1 forces cell BIP8 error in the TX path.	Cell
	[4]	FORCE_A1A2_ERR_xx		Force A1A2 Error, FORCE_A1A2_ERR =1 forces an error in A1A2 bytes (framing error) in the TX path. Valid only when the corresponding AUTO_TOH_xx bit or AUTO_A1A2_xx bit is set to one.	Both
	[5]	FORCE_EX_SEQ_ERR_xx		Force Excessive Sequence Errors. FORCE_EXP_SEQ_ERR_xx = 1 forces sequence errors in the TX path by inverting the link header byte sequence number.	Cell
	[6]	FORCE_SEQ_ERR_xx		Force Sequence Error, FORCE_SEQ_ERR =1 forces one sequence error in the TX path. After this bit is set, the next Link Header byte's sequence number is inverted. The Link Header after the errored Link Header byte will have the correct sequence number	Cell
	[7]	FORCE_RDI_xx		Force Remote Defect Indication, FORCE_RDI =1 forces RDI errors in the TX path. The K2 byte in TOH is set to '00000110'. Valid only when AUTO_TOH_xx bit is set to 1.	Cell

Table 36. Common Control Register Descriptions – ORSO82G5

(0x) Absolute Address	Bit	Name	Reset Value (0x)	Description	Mode
30A00	[0:1]	RCKSELB	00	“00” - Channel BA source for clock RCK78B “01” - Channel BB source for clock RCK78B “10” - Channel BC source for clock RCK78B “11” - Channel BD source for clock RCK78B	Both
	[2:3]	TCKSELB		“00” - Channel BA source for clock TCK78B “01” - Channel BB source for clock TCK78B “10” - Channel BC source for clock TCK78B “11” - Channel BD source for clock TCK78B	Both
	[4:5]	RCKSELA		“00” - Channel AA source for clock RCK78A “01” - Channel AB source for clock RCK78A “10” - Channel AC source for clock RCK78A “11” - Channel AD source for clock RCK78A	Both
	[6:7]	TCKSELA		“00” - Channel AA source for clock TCK78A “01” - Channel AB source for clock TCK78A “10” - Channel AC source for clock TCK78A “11” - Channel AD source for clock TCK78A	Both
30A01	[0:2]	CELL_SIZE	00	Cell Size, Three bits to set cell size. “000” - Cell size is 75 bytes, “001” - Cell size is 79 bytes, “010” - Cell size is 83 bytes, “011” - Cell size is 91 bytes These are the only supported cell sizes.	Cell
	[3:7]	RX_FIFO_MIN		Set Minimum threshold value for alignment FIFO in SONET mode. When the read address for the FIFO is below this value at the time when write address is zero, it indicates that the FIFO is near overflow. This event will go high only once during a frame when a framing byte has been detected by the aligner. The default threshold value is “00000”.	SONET
30A02	0	TX_DISABLE_ON_RDI	00	Transmitter Disable on RDI (Detection), If TX_DISABLE_ON_RDI = 1 - No cell data is transmitted on a link in which a RDI has been detected by the corresponding link’s receiver. If this bit is set to 0, cell data will be transmitted on a link irrespective of detection of a RDI.	Cell
	[1:3]	RSVD		Reserved	—
	4	SCHAR_ENA		SCHAR_ENA = 1 enables SERDES characterization of SERDES B. Refer to section	Factory Test
	5	SCHAR_TXSEL		SCHAR_TXSEL =1 is a Select Tx option which will cause chip ports to directly control the SERDES low-speed transmit ports of one of the channels selected by SCHAR_CHAN	Factory Test
	6:7	SCHAR_CHAN		“00” - Select channel BA to test “01” - Select channel BB to test “10” - Select channel BC to test “11” - Select channel BD to test	Factory Test

Table 52. ORSO82G5 680-Pin PBGAM (fpBGA) Pinout (Continued)

680-PBGAM	VDDIO Bank	VREF Group	I/O	Pin Description	Additional Function	680-PBGAM
U21	—	—	VDD15	VDD15	—	—
U22	—	—	VDD15	VDD15	—	—
V13	—	—	VDD15	VDD15	—	—
V14	—	—	VDD15	VDD15	—	—
V15	—	—	VDD15	VDD15	—	—
V20	—	—	VDD15	VDD15	—	—
V21	—	—	VDD15	VDD15	—	—
V22	—	—	VDD15	VDD15	—	—
W13	—	—	VDD15	VDD15	—	—
W14	—	—	VDD15	VDD15	—	—
W15	—	—	VDD15	VDD15	—	—
W20	—	—	VDD15	VDD15	—	—
W21	—	—	VDD15	VDD15	—	—
W22	—	—	VDD15	VDD15	—	—
Y16	—	—	VDD15	VDD15	—	—
Y17	—	—	VDD15	VDD15	—	—
Y18	—	—	VDD15	VDD15	—	—
Y19	—	—	VDD15	VDD15	—	—
T32	—	—	NC	NC	—	—
W32	—	—	NC	NC	—	—