CONDUANT

StreamStor LVDS16 Daughter Board

User Manual

Copyright and Trademarks

The information in this document is subject to change without notice.

Printed in the United States.

Manual version: 8.11

Publication date: April 18, 2008

TABLE OF CONTENTS

License Agreement and Limited Warranty

IMPORTANT: CAREFULLY READ THE TERMS AND CONDITIONS OF THIS AGREEMENT BEFORE USING THE PRODUCT. By installing or otherwise using the StreamStor Product, you agree to be bound by the terms of this Agreement. If you do not agree to the terms of this Agreement, do not install or use the StreamStor Product and return it to Conduant Corporation.

GRANT OF LICENSE. In consideration for your purchase of the StreamStor Product, Conduant Corporation hereby grants you a limited, non-exclusive, revocable license to use the software and firmware which controls the StreamStor Product (hereinafter the "Software") solely as part of and in connection with your use of the StreamStor Product. If you are authorized to resell the StreamStor Product, Conduant Corporation hereby grants you a limited non-exclusive license to transfer the Software only in conjunction with a sale or transfer by you of the StreamStor Product controlled by the Software, provided you retain no copies of the Software and the recipient agrees to be bound by the terms of this Agreement and you comply with the RESALE provision herein.

NO REVERSE ENGINEERING. You may not cause or permit, and must take all appropriate and reasonable steps necessary to prevent, the reverse engineering, decompilation, reverse assembly, modification, reconfiguration or creation of derivative works of the Software, in whole or in part.

OWNERSHIP. The Software is a proprietary product of Conduant Corporation which retains all title, rights and interest in and to the Software, including, but not limited to, all copyrights, trademarks, trade secrets, know-how and other proprietary information included or embodied in the Software. The Software is protected by national copyright laws and international copyright treaties.

TERM. This Agreement is effective from the date of receipt of the StreamStor Product and the Software. This Agreement will terminate automatically at any time, without prior notice to you, if you fail to comply with any of the provisions hereunder. Upon termination of this Agreement for any reason, you must return the StreamStor Product and Software in your possession or control to Conduant Corporation.

LIMITED WARRANTY. This Limited Warranty is void if failure of the StreamStor Product or the Software is due to accident, abuse or misuse.

Hardware: Conduant's terms of warranty on all manufactured products is one year from the date of shipment from our offices. After the warranty period, product support and repairs are available on a fee paid basis. Warranty on all third party materials sold through Conduant, such as chassis, disk drives, PCs, bus extenders, and drive carriers, is passed through with the original manufacturer's warranty. Conduant will provide no charge service for 90 days to replace or handle repair returns on third party materials. Any charges imposed by the original manufacturer will be passed through to the customer. After 90 days, Conduant will handle returns on third party material on a time and materials basis.

Software: The warranty on all software products is 90 days from the date of shipment from Conduant's offices. After 90 days, Conduant will provide product support and upgrades on a fee paid basis. Warranties on all third party software are passed through with the original manufacturer's warranty. Conduant will provide no charge service for 90 days to replace or handle repair returns on third party software. Any charges imposed by the manufacturer will be passed through to the customer.

DISCLAIMER OF WARRANTIES. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CONDUANT CORPORATION DISCLAIMS ALL OTHER WARRANTIES AND CONDITIONS, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, AND NONINFRINGEMENT, WITH REGARD TO THE STREAMSTOR PRODUCT AND THE SOFTWARE.

SOLE REMEDIES. If the StreamStor Product or the Software do not meet Conduant Corporation's Limited Warranty and you return the StreamStor Product and the Software to Conduant Corporation, Conduant

Corporation's entire liability and your exclusive remedy shall be at Conduant Corporation 's option, either (a) return of the price paid, if any, or (b) repair or replacement of the StreamStor Product or the Software. Any replacement Product or Software will be warranted for the remainder of the original warranty period.

LIMITATION OF LIABILITIES. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL CONDUANT CORPORATION BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR ANY OTHER PECUNIARY LOSS) ARISING OUT OF THE USE OF OR INABILITY TO USE THE STREAMSTOR PRODUCT AND THE SOFTWARE. IN ANY CASE, CONDUANT CORPORATION'S ENTIRE LIABILITY UNDER ANY PROVISION OF THIS AGREEMENT SHALL BE LIMITED TO THE AMOUNT ACTUALLY PAID BY YOU FOR THE STREAMSTOR PRODUCT AND THE SOFTWARE. BECAUSE SOME STATES AND JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF LIABILITY, THE ABOVE LIMITATION MAY NOT APPLY TO YOU.

RESALE. If you are authorized to resell the StreamStor Product, you must distribute the StreamStor Product only in conjunction with and as part of your product that is designed, developed and tested to operate with and add significant functionality to the StreamStor Product; you may not permit further distribution or transfer of the StreamStor Product by your end-user customer; you must agree to indemnify, hold harmless and defend Conduant Corporation from and against any claims or lawsuits, including attorneys' fees, that arise or result from the use or distribution of your product; and you may not use Conduant Corporation's name, logos or trademarks to market your product without the prior written consent of Conduant Corporation.

ENTIRE AGREEMENT; SEVERABILITY. This Agreement constitutes the complete and exclusive agreement between you and Conduant Corporation with respect to the subject matter hereof and supersedes all prior written or oral agreements, understandings or communications. If any provision of this Agreement is deemed invalid under any applicable law, it shall be deemed modified or omitted to the extent necessary to comply with such law and the remainder of this Agreement shall remain in full force and effect.

GOVERNING LAW. This Agreement is governed by the laws of the State of Colorado, without giving effect to the choice of law provisions therein. By accepting this Agreement, you hereby consent to the exclusive jurisdiction of the state and federal courts sitting in the State of Colorado.

About This Manual

This manual is intended to serve the following purposes:

- to provide an overview of the StreamStor LVDS16 daughter board;
- to act as a reference for the operator; and
- to provide guidance on software capabilities and choices.

It is suggested that you periodically check the Conduant web site for the most recent software updates, application notes, and technical bulletins.

If you are unable to locate the information you need, please feel free to contact us by e-mail or phone.

Overview

The StreamStor LVDS16 daughter board is a mezzanine IO board that can be used with some StreamStor controllers such as the Amazon. Adding this daughter board provides a high-speed data interface that allows the simple front panel data port (FPDP) high speed data interface to be extended over longer cable lengths utilizing low voltage differential signals and receiver channel-to-channel skew cancellation. A 16 bit LVDS record and playback interface operates with FPDP style hardware flow control similar to the receive master and transmit master ports found in the FPDP specification, ANSI/VITA 17-1998.

The LVDS daughter board provides two independent interfaces. One interface is the record port. The other interface is the playback port. The receive port and transmit port are independent. The transmit port is optional and takes a second PCI slot for cabling.

In reading the following sections on using this daughter board, it is important to be familiar with the American National Standard entitled "Electrical Characteristics of Low Voltage Differential Signaling (LVDS) Interface Circuits" (ANSI/TIA/EIA-644-A-2201). For information, please visit www.tiaonline.org.

Interface Electronics

Interface electronics and termination values on StreamStor are those recommended by the ANSI standard, though some signals and terminations can be electronically connected or isolated with crossbar switching devices in order to support electronic reconfiguration.

Data Formats

LVDS provides a point-to-point 16-bit LVDS unframed data stored as pairs stacked to 32-bit words.

PIO Signals

Programmable user IO signals are not stored with recorded data. For application specific use of these, such as event recording, please contact Conduant.

Connector / Cabling

Standard 12 lane and 4 lane Infiniband cables with jack screws can be used to connect a data source to the LVDS recorder. For loopback testing of one LVDS Amazon play port to an LVDS record port, a straight through 4 lane Infiniband style cable is required. The following table defines the pinout of the 12 and 4 lane connectors.

SIGNAL	TYPE	CONN	PIN	DIR	NOTE
R_DATA0	LVDS	12	1, 2	IN	16 bits of data driven by source. Each nibble has an independent valid but a common clock. 16 bit recording should have all RDVLD signals low.
R_DATA1	LVDS	12	3, 4	IN	
R_DATA2	LVDS	12	5, 6	IN	
R_DATA3	LVDS	12	7, 8	IN	
R_DATA4	LVDS	12	17, 18	IN	
R_DATA5	LVDS	12	19, 20	IN	
R_DATA6	LVDS	12	21, 22	IN	
R_DATA7	LVDS	12	23, 24	IN	
R_DATA8	LVDS	12	25, 26	IN	
R_DATA9	LVDS	12	27, 28	IN	
R_DATA10	LVDS	12	29, 30	IN	
R_DATA11	LVDS	12	31, 32	IN	
R_DATA12	LVDS	12	41, 42	IN	
R_DATA13	LVDS	12	43, 44	IN	
R_DATA14	LVDS	12	45, 46	IN	
R_DATA15	LVDS	12	47, 48	IN	
RDATA_CLK	LVDS	12	9, 10	IN	Data clock, 200MHz max
RDVLD0	LVDS	12	11, 12	IN	Synchronous to clock. Signals should be low for normal recording.
RDVLD1	LVDS	12	13, 14	IN	
RDVLD2	LVDS	12	35, 36	IN	
RDVLD3	LVDS	12	37, 38	IN	
TRIG_LVDS	LVDS	4	3, 4	IN	Start of transfer signals. Must remain true for 5 clocks. Use is optional and must be enabled in software. (opt)
TRIG_RS422	RS422	4	9, 10	IN	
SYNCIN	LVDS	4	1, 2	IN	Event capture signal (opt)
NRDY	LVDS	4	11, 12	OUT	Recorder not ready
SUSPEND	LVDS	4	13, 14	OUT	Data suspend (stop data flow)

Programming

The LVDS daughter board is designed for maximum flexibility and can be configured to run as a receiver or transmitter at various clock frequencies.

Configuring LVDS with XLRSetDBMode

The XLRSetDBMode function provided by the StreamStor API is used to set most of LVDS parameters and options (see the SDK User Manual). Table 1 lists the supported modes for this daughter board. Note that the board must be in a compatible mode for the StreamStor operation being requested (i.e., receive mode for XLRRecord, transmit mode for XLRPlayback).

TABLE 1 – FPDP Modes

XLRSetDBMode Mode	Description
SS_LVDS16MODE_RECV	Receive data, receive master termination applied.
SS_LVDS16MODE_XMIT	Transmit data and clocks.

Table 2 details the supported LVDS daughter board options. These options may be combined using a Boolean "or" operation before passing them into the XLRSetDBMode function.

TABLE 2 – FPDP Options

XLRSetDBMode Option	Description
SS_DBOPT_LVDS16_FLOWCONTROL	Enables flow control. By default, it is not enabled.
SS_DBOPT_LVDS16_DATAVALID_GLOBAL	Enables the LVDS global DATA VALID starting pulse. By default, it is not enabled.
SS_DBOPT_LVDS16_DATAVALID_RS422	Enables the LVDS RS-422 global DATA VALID starting pulse. By default, it is not enabled.

Setting up Channels

The StreamStor system can be configured in various ways to record or playback from the LVDS ports or the PCI bus.

Setting the Channel Mode

The function `XLRSetMode` is used to set the input/output path on the StreamStor. The only valid channel mode for an LVDS daughter board is `SS_MODE_SINGLE_CHANNEL`. This is the default mode. In this mode, data is sent and received over a single channel.

Binding Input and Output Channels

Each of the possible input or output ports is considered a channel and must be bound into the StreamStor controller before recording or playback will occur from that port. By default, the PCI bus is bound as both the input and output port in single channel mode. The API functions `XLRBindInputChannel` and `XLRBindOutputChannel` must be called to define an input and output port if other than the PCI bus. Table 3 defines the constants to use to select the appropriate channel for your application.

TABLE 3 – Channel definition

Channel Number	Channel Description
`0`	PCI
`30`	LVDS record port
`31`	LVDS playback port

Defining Phase Relationships with XLRSetDBBitPhase

The `XLRSetDBBitPhase` function provided by the StreamStor API is used to redefine the phase relationship of each data bit with respect to its associated nibble clock. You pass `XLRSetDBBitPhase` the bit to be adjusted and the direction in which to adjust.

For example, if bit 0 of the `adjust` parameter is set, then bit 0 of the data bus will be adjusted in the direction specified in bit 0 of the `direction` parameter. Likewise if `adjust’s` bit 3 is set, then bit 3 will be adjusted in the direction specified in bit 3 of `direction`.

Selecting a Clock Edge with XLRSetDBEdge

The `XLRSetDBEdge` function provided by the StreamStor API is used to select the positive or negative clock edge for sampling each signal. The `edgeSelect` parameter of `XLRSetDBEdge` is a 32 bit integer that is used to specify the positive or negative edge of the clock to sample data. Table 4 lists each bit of `edgeSelect` and its purpose.

Table 4 – Edge Selector Bits

Bit Position	Signal Name
31	Reserved
30	not used
29	not used
28	not used
27	not used
26	not used
25	dValid_RS422
24	dValid_LVDS
23	not used
22	dValid(3)
21	Record data(15)
20	Record data(14)
19	Record data(13)
18	Record data(12)
17	PIO(2)
16	dValid(2)
15	Record data(11)
14	Record data(10)
13	Record data(9)
12	Record data(8)
11	PIO(1)
10	dValid(1)
9	Record data(7)
8	Record data(6)
7	Record data(5)
6	Record data(4)
5	Sync
4	dValid(0)
3	Record data(3)
2	Record data(2)
1	Record data(1)
0	Record data(0)

Setting Clock Speeds

The LVDS daughter board provides programmable clock speeds at the record and playback interfaces. When setting a frequency, it is applied to both the record and playback ports.

Predefined Frequencies using XLRSetPortClock

The `XLRSetPortClock` function is used to set a predefined frequency for recording and playing back. Table 5 lists the available clock settings.

TABLE 5 – Predefined Clock Settings

XLRSetPortClock clock	Actual clock Speed (MHZ)
`SS_LVDSCLOCK_20MHZ`	20
`SS_LVDSCLOCK_31_25MHZ`	31.25
`SS_LVDSCLOCK_62_5MHZ`	62.5
`SS_LVDSCLOCK_95MHZ`	95
`SS_LVDSCLOCK_100MHZ`	100
`SS_LVDSCLOCK_125MHZ`	125
`SS_LVDSCLOCK_150MHZ`	150
`SS_LVDSCLOCK_160MHZ`	160
`SS_LVDSCLOCK_190MHZ`	190
`SS_LVDSCLOCK_200MHZ`	200

Custom Frequencies using XLRSetLVDSClock

The `XLRSetLVDSClock` function is used to set a frequency other than one of those that are predefined by the StreamStor API. Custom frequencies are applied only to recording. (I.e., for playback, you must use a predefined frequency.) `XLRSetLVDSClock` requires you specify a PLL (phase lock loop) for which you are setting the frequency. Table 6 lists the available PLL values.

TABLE 6 – PLL Values

PLL	Record Nibble Set
`SS_LVDS_RECORD_CLOCK_0`	Clock 0
`SS_LVDS_RECORD_CLOCK_1`	Clock 1
`SS_LVDS_RECORD_CLOCK_2`	Clock 2
`SS_LVDS_RECORD_CLOCK_3`	Clock 3

`XLRSetLVDSClock` also requires you specify a frequency array. The frequency array is of type `S_LVDS_CONFIG`, which is an array of six unsigned integer values. These six values correspond to the bit stream value that can be obtained from Altera's design tools.

The bit stream is described in Altera's Application Note 367, "Implementing PLL Reconfiguration in Stratix II Devices." It can be downloaded from www.altera.com/literature/an/an367.pdf. The values to place in the array can be obtained by using Altera's Quartus II design software, which is downloadable from the Altera web site, www.altera.com.

Function Reference

The following functions are applicable only to the LVDS daughter board.

- `XLRSetDBBitPhase`
- `XLRSetDBEdge`
- `XLRSetLVDSClock`

Manual pages for these functions are at the end of this section. Manual pages for all other commands (i.e., those that are not specific to the LVDS daughter board, such as `XLRSetPortClock` and `XLRSetDBMode`) are in the SDK User's Guide.

XLRSetDBBitPhase

Syntax:

```
XLR_RETURN_CODE XLRSetDBBitPhase( SSHANDLE xlrDevice,
   USHORT adjust, USHORT direction )
```

Description:

`XLRSetDBBitPhase` sets the phase relationship of each data bit with respect to its associated nibble clock.

Parameters:

- *`xlrDevice`* - the device handle returned from a previous call to `XLROpen`.
- *`adjust`* - specifies the bit to be adjusted.
- *`direction`* - specifies the direction of the adjustment.

For example, if bit 0 of `adjust` is set, then bit 0 will be adjusted in the direction specified in bit 0 of `direction`. Likewise if `adjust's` bit 3 is set, then bit 3 will be adjusted in the direction specified in bit 3 of `direction`.

Return Value:

On success, this function returns XLR_SUCCESS.
On failure, this function returns XLR_FAIL.

Usage:

```
#define BIT3  0x04
SSHANDLE      xlrDevice;
USHORT        dbits;
USHORT        direction;

if ( XLROpen( 1, &xlrDevice ) != XLR_SUCCESS) {
  printf (“Cannot open card.\n”);
  exit(1);
}

// Adjust bit 3 forward.
dbits = dbits | BIT3;
direction = direction | BIT3;

if (XLRSetDBBitPhase( xlrDevice, dbits, direction ) {
      printf (“Bit 3 adjusted forward.\n”);
}
```

See Also:

`XLRSetDBEdge` and `XLRSetLVDSClock`.

XLRSetDBEdge

Syntax:

```
XLR_RETURN_CODE XLRSetDBEdge( SSHANDLE xlrDevice,
     UINT edgeSelect )
```

Description:

XLRSetDBEdge sets the clock edge for data capture of each bit.

Parameters:

- *xlrDevice* - the device handle returned from a previous call to XLROpen.
- *edgeSelect* – the positive or negative edge of the clock to sample data. A zero in a bit position indicates negative. A one in a bit position indicates positive. Refer to the "Edge Selector Bits" table in the LVDS User Manual for a description of each bit.

Return Value:

On success, this function returns XLR_SUCCESS.
On failure, this function returns XLR_FAIL.

Usage:

```
SSHANDLE     xlrDevice;

if ( XLROpen( 1, &xlrDevice ) != XLR_SUCCESS) {
  printf (“Cannot open card.\n”);
  exit(1);
}

// Configure to use the rising edge for everything and to
// not use global data Valids.
if ( XLRSetDBEdge( xlrDevice, 0x00FFFFFF ) != XLR_SUCCESS ) {
  printf (“Cannot set DB edge.\n”);
  exit(1);
}
```

See Also:

XLRSetDBBitPhase and XLRSetLVDSClock.

XLRSetLVDSClock

Syntax:

```
XLR_RETURN_CODE XLRSetLVDSClock( SSHANDLE xlrDevice, USHORT pll,
S_LVDS_CONFIG configArray[] )
```

Description:

`XLRSetLVDS` sets the playback frequency on the LVDS daughter board.

Parameters:

- *`xlrDevice`* - the device handle returned from a previous call to `XLROpen`.
- *`pll`* - the PLL (phase lock loop) for which you are setting the frequency. *`pll`* is one of:
  - SS_LVDS_RECORD_CLOCK_0
  - SS_LVDS_RECORD_CLOCK_1
  - SS_LVDS_RECORD_CLOCK_2
  - SS_LVDS_RECORD_CLOCK_3
- *`configArray`* – the six element array holding the edge selector bit values. Note that the array holds 192 bits. The Altera bit stream is 172 bits, so you must pad the upper bits of the last word in the array.

Return Value:

On success, this function returns XLR_SUCCESS.
On failure, this function returns XLR_FAIL.

Usage:

```
SSHANDLE     xlrDevice;
UINT         i;

if ( XLROpen( 1, &xlrDevice ) != XLR_SUCCESS) {
  printf (“Cannot open card.\n”);
  exit(1);
}

// Set a custom LVDS clock frequency
for( i = 0; i < 4; i++ ) {
   sLVDS[0] = 0x00000C06;
   sLVDS[1] = 0x80402004;
   sLVDS[2] = 0x04020100;
   sLVDS[3] = 0x40201008;
   sLVDS[4] = 0x00000100;
   sLVDS[5] = 0x00000008;
   if( XLRSetLVDSClock( xlrDevice, (SS_LVDS_RECORD_CLOCK_0 + i),
       &sLVDS ) != XLR_SUCCESS ){
      printf( "ERROR: Set LVDS clock failed.\n" );
   }
}
```

See Also:

`XLRSetDBBitPhase` and `XLRSetDBEdge`.

Technical Support

Conduant wants to be sure that your StreamStor system works correctly and stays working correctly. In the event, however, that you are unable to get your system to work properly, or if a working system ceases to function, we will do all that we can to get your system back online.

Solving the problem is largely a matter of data collection and steps that must be taken one at a time. In order for us to better serve you, we ask that you take the time to perform the following steps prior to calling us. This way, you can provide us with the most meaningful information possible that will help us solve the problem.

Is the problem one that obviously requires replacement parts due to physical damage to the system? If yes, then please gather the information described below and report the problem to tech support, by phone or through the Conduant web site.

Have you confirmed that no cabling has been inadvertently disconnected or damaged while working around the equipment?

Is the StreamStor card properly seated in the PCI (CPCI/PXI) slot?

Do all the systems have good power connections and voltages?

Does the confidence test sscfg.exe (on Windows) or ssopen/sstest (on Linux) run OK?

Has the software installation been corrupted? Try re-installing software.

Have you checked the Conduant web site for technical bulletins?

Have you recently installed a new Linux kernel or compiler or a new Windows Service Pack?

If the above steps did not resolve the problem, then please initiate a trouble ticket on the support section of the Conduant website at www.conduant.com. Click on “Support” and then click on “Submit a Ticket.” Please provide as much information about your system and the problem as possible. We will do all that we can to resolve the problem as quickly as possible.

Contacting Technical Support

E-mail: support@conduant.com

Web: www.conduant.com

Mail: Conduant Corporation
Technical Support
1501 South Sunset Street, Suite C
Longmont, CO 80501