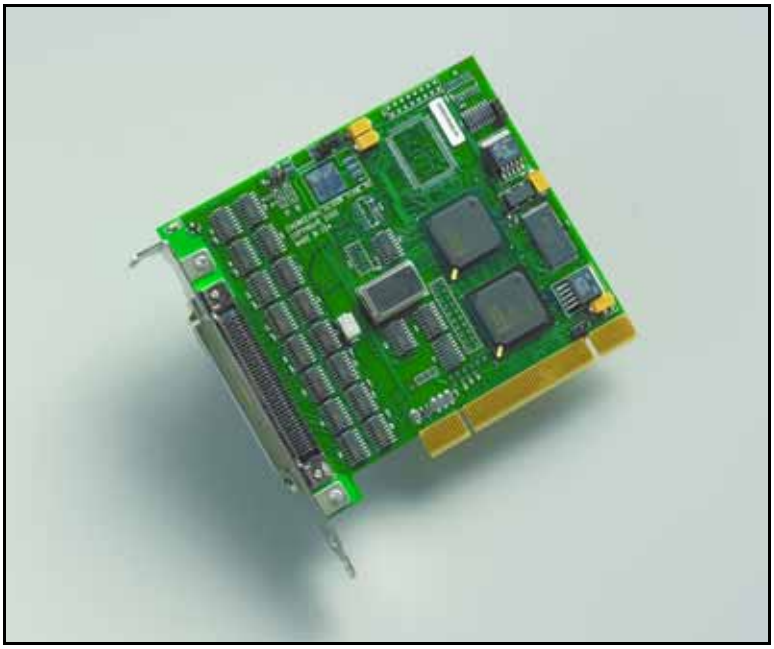
User's Guide Addendum

SSD8IO-MF2

Configuration Package



Synchronous serial device with 8 I/O (multi-function 2) for use with EDT PCI CDa interface

Doc. 008-04025-01
Rev. 2011 August 17

Contact

Sky Blue Microsystems GmbH
Geisenhausenerstr. 18
81379 Munich
Germany
www.skyblue.de
info@skyblue.de
Tel. +49 (0) 89 - 780297 0

Terms of Use Agreement

Definitions. This agreement, between Engineering Design Team, Inc. (“Seller”) and the user or distributor (“Buyer”), covers the use and distribution of the following items provided by Seller: a) the binary and all provided source code for any and all device drivers, software libraries, utilities, and example applications (collectively, “Software”); b) the binary and all provided source code for any and all configurable or programmable devices (collectively, “Firmware”); and c) the computer boards and all other physical components (collectively, “Hardware”). Software, Firmware, and Hardware are collectively referred to as “Products.” This agreement also covers Seller’s published Limited Warranty (“Warranty”) and all other published manuals and product information in physical, electronic, or any other form (“Documentation”).

License. Seller grants Buyer the right to use or distribute Seller’s Software and Firmware Products solely to enable Seller’s Hardware Products. Seller’s Software and Firmware must be used on the same computer as Seller’s Hardware. Seller’s Products and Documentation are furnished under, and may be used only in accordance with, the terms of this agreement. By using or distributing Seller’s Products and Documentation, Buyer agrees to the terms of this agreement, as well as any additional agreements (such as a nondisclosure agreement) between Buyer and Seller.

Export Restrictions. Buyer will not permit Seller’s Software, Firmware, or Hardware to be sent to, or used in, any other country except in compliance with applicable U.S. laws and regulations. For clarification or advice on such laws and regulations, Buyer should contact: U.S. Department of Commerce, Export Division, Washington, D.C., 20230, U.S.A.

Limitation of Rights. Seller grants Buyer a royalty-free right to modify, reproduce, and distribute executable files using the Seller’s Software and Firmware, provided that: a) the source code and executable files will be used only with Seller’s Hardware; b) Buyer agrees to indemnify, hold harmless, and defend Seller from and against any claims or lawsuits, including attorneys’ fees, that arise or result from the use or distribution of Buyer’s products containing Seller’s Products. Seller’s Hardware may not be copied or recreated in any form or by any means without Seller’s express written consent.

No Liability for Consequential Damages. In no event will Seller, its directors, officers, employees, or agents be liable to Buyer for any consequential, incidental, or indirect damages (including damages for business interruptions, loss of business profits or information, and the like) arising out of the use or inability to use the Products, even if Seller has been advised of the possibility of such damages. Because some jurisdictions do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitations may not apply to Buyer. Seller’s liability to Buyer for actual damages for any cause whatsoever, and regardless of the form of the action (whether in contract, product liability, tort including negligence, or otherwise) will be limited to fifty U.S. dollars ($50.00).

Limited Hardware Warranty. Seller warrants that the Hardware it manufactures and sells shall be free of defects in materials and workmanship for a period of 12 months from date of shipment to initial Buyer. This warranty does not apply to any product that is misused, abused, repaired, or otherwise modified by Buyer or others. Seller’s sole obligation for breach of this warranty shall be to repair or replace (F.O.B. Seller’s plant, Beaverton, Oregon, USA) any goods that are found to be non-conforming or defective as specified by Buyer within 30 days of discovery of any defect. Buyer shall bear all installation and transportation expenses, and all other incidental expenses and damages.

Limitation of Liability. *In no event shall Seller be liable for any type of special consequential, incidental, or penal damages, whether such damages arise from, or are a result of, breach of contract, warranty, tort (including negligence), strict liability, or otherwise.* All references to damages herein shall include, but not be limited to: loss of profit or revenue; loss of use of the goods or associated equipment; costs of substitute goods, equipment, or facilities; downtime costs; or claims for damages. Seller shall not be liable for any loss, claim, expense, or damage caused by, contributed to, or arising out of the acts or omissions of Buyer, whether negligent or otherwise.

No Other Warranties. Seller makes no other warranties, express or implied, including without limitation the implied warranties of merchantability and fitness for a particular purpose, regarding Seller’s Products or Documentation. Seller does not warrant, guarantee, or make any representations regarding the use or the results of the use of the Products or Documentation or their correctness, accuracy, reliability, currentness, or otherwise. All risk related to the results and performance of the Products and Documentation is assumed by Buyer. The exclusion of implied warranties is not permitted by some jurisdictions. The above exclusion may not apply to Buyer.

Disclaimer. Seller’s Products and Documentation, including this document, are subject to change without notice. Documentation does not represent a commitment from Seller.

Contents

SSD8IO-MF2 Configuration Package

Overview

The SSD8IO-MF2 (Synchronous Serial Device, 8 I/O - Multifunction 2) Configuration Package is a package that enables the EDT PCI CDa board to transfer eight channels of synchronous serial input/output (I/O) between an external device and a PCI computer.

NOTE The SSD8IO-MF2 firmware was developed using the Xilinx Project Navigator. The VHDL source is available and the project is set up for you to use, if you wish; for details, contact tech@edt.com.

Channels 0–3 provide additional handshaking and clocking signals, while channels 4–7 provide the same functionality as the FPGA files in the SSD16IO configuration package (see Related Resources on page 1).

Channels 0–3 support both sender- and receiver-initiated flow control. The sender can pause transmission at any time by holding the clock high; the receiver can pause transmission at any time by deasserting the ready signal. The receiver will safely accept at least 16 bits after suspending transmission. Transmission or reception can be paused manually via the 0x3F User Pause register.

Channels 0–1 are output channels; each has a frame valid (active low) output signal and a device ready (active high) input signal in addition to the normal clock and data signals. Output clock signals can be sourced from either an internal PLL or an external clock input via the connector pinout. The transmitted frame length in bytes is programmable via the configuration registers (see 0x39 – 3B Channel 0 Transmit Frame Length and 0x3C – 3E Channel 1 Transmit Frame Length).

Channels 2–3 are input channels; each has a frame valid (active low) input signal and a ready (active high) output signal in addition to the normal clock and data input signals. Data is latched only when the frame valid input signal is true. If a frame ends on a non-4byte boundary, the receiver will fill the remaining bytes as zero.

Channels 4–7 can be configured as either input or output channels in pairs. No additional handshaking signals are provided; however, the output clock can be sourced independently from channels 0 and 1.

Related Resources

These EDT resources may prove helpful or necessary for your applications.

Resource	*EDT weblink*
SSD16IO Addendum -User's Guide	www.edt.com/manuals/PCD/ssd16.pdf
PCI CD / CDa User's Guide	www.edt.com/manuals/PCD/pcicd.pdf
Application Programming Interface	www.edt.com/manuals.html
Installation packages / software downloads for supported operating systems (Windows, Linux, etc.)	www.edt.com/software.html

Installation

The EDT installation disk provides installation packages for all supported operating systems (Windows, Linux, Solaris, MacOS).

However, to get the most current package and avoid version issues later, for new applications we recommend downloading the latest EDT installation package from our website (as instructed below).

NOTE For existing applications, avoid version issues by updating only if you have a specific reason to do so.

To download the latest EDT installation package:

1. Go to www.edt.com/software.html and find the correct package for your system.
2. Install the Pcd driver software by doing one of the following:
   — For a new application, download the latest package for your operating system.
   — For an existing application, use the package that was used to build it (from your own or EDT's archives), or recompile / relink the application with the latest installation package download.
3. Install the board assembly in the host computer as specified by the computer manufacturer.
4. To configure the board, at the command prompt, enter:

   `initpcd -u` *`unit number`* `-f` *`configuration file`*

   ...substituting appropriate values for the placeholder text (indicated in *italic*).

   For example, to configure board 0 with the sample configuration file provided, enter:

   `initpcd -u` *`0`* `-f` *`pcd_config/ssd8io_mf2.cfg`*

About the Software and Firmware

Your EDT installation package includes software and firmware for the SSD8IO-MF2 configuration package. This software and firmware includes at least the files listed below.

`ssd8io_mf2.bit`	FPGA configuration file for the UI FPGA on the PCI CDa board (LVDS or RS422); implements the multi-function firmware capabilities.
`ssd8io_mf2.cfg`	Sample software initialization file to enable operation with `ssd8io_mf2.bit`. All sample software initialization files (editable text files that you can customize for your own applications) are in the `pcd_config` subdirectory of your EDT installation package.
`cda16_classic.bit`	FPGA configuration file for the PCI FPGA on the PCI CDa board.

To load the `cda16_classic.bit` file (required), go to the directory installed on the driver and run...

```
pciload –u 0 cda16_classic.bit
```

FPGA Configuration Files

The PCI CDa board implements the DMA interface using two field-programmable gate arrays (FPGAs), referred to as the PCI FPGA and the UI (user interface) FPGA:

The *PCI FPGA* communicates with the host computer over the PCI bus. It implements the DMA engine, which transfers data between the board and the host computer, and loads its firmware on powerup from flash ROM located on the main board.

The *UI FPGA* transfers data between the user device and the PCI FPGA; in some instances, it also sends the data to the mezzanine board. The UI FPGA or mezzanine board may also process the data in some manner, depending on the application.

FPGA configuration files define the firmware required for the PCI FPGA and the UI FPGA.

- PCI FPGA configuration files are in the `flash` subdirectory of the EDT installation package.
- UI FPGA configuration files are in the `bitfiles` subdirectory of the EDT installation package.

The PCD Device Driver

The PCD device driver is the software running on the host computer that allows the host operating system to communicate with the SSD8IO-MF2 configuration package. The driver is loaded into the kernel upon installation, and thereafter runs as a kernel module. The driver name and subdirectory is specific to each supported operating system; the installation script handles those details for you, automatically installing the correct device driver in the correct operating system-specific manner.

Sample Program Files

Along with the driver, the FPGA configuration files, and the software initialization files, the EDT installation package includes applications and utilities that you can use to initialize and configure the board, access registers, or test the board. For many of these programs, C source is also provided, so that you can use them as starting points to write your own applications. The most commonly useful are described below.

Application Files

`rd16`	Performs simple ring buffer input on a specified channel.
`wr16`	Performs simple ring buffer output on a specified channel.
`simple_read`	Performs DMA input without using ring buffers. Data is therefore subject to interruptions, depending on system performance.
`simple_write`	Performs DMA output without using ring buffers. Data is therefore subject to interruptions, depending on system performance.
`simple_getdata`	Serves as an example of a variety of DMA-related operations, including reading the data from the connector interface and writing it to a file, as well as measuring input rate.
`simple_putdata`	Serves as an example of a variety of DMA-related operations, including reading data from a file and writing it out to the connector interface.

Utility Files

`initpcd`	A utility for initializing and configuring the SSD8IO.
`pciload`	A utility for loading the PCI FPGA firmware.
`pdb`	A utility that enables interactive reading and writing of the PCI CDa UI FPGA registers.

Test Files

Various C source, executable, and FPGA configuration files are available for testing (see Testing on page 4), including at least:

`ssd8io_mf2_test.c` Tests the PCI CDa multifunction configuration.

Building Applications

Executable and PCD source files are at the top level of your EDT installation package. If you need to rebuild an application, therefore, run `make` in this top-level directory.

To install a compiler:

- With Linux, you can use the `gcc` compiler typically included with the Linux installation.
- With Windows or Solaris, you must either install a C compiler (EDT recommends Microsoft Visual C for Windows, or Sun WorkShop C for Solaris) or contact tech@edt.com for instructions on using `gcc`.

After building an application, use the `--help` command line option for a list of usage options and descriptions.

Testing

After installing a loopback connector (described in the Pinouts section), run the test program...

```
ssd8io_mf2_test -u N
```

...where `N` is the unit number of the PCI CDa board being tested. The test is menu-driven, and it includes tests to send data across the loopback cable, set the buffer size, and adjust the enable signal block size and the number of inter-block clocks.

For details, see `ssd8io_mf2_test.c`.

Registers

In the SSD8IO-MF2 configuration package, the following registers are modified, as shown below, from those in the SSD16IO configuration package.

0x21 Ungated Channels Clock Select

Access / Notes: 8-bit read-write / EDT_SS_CLK_SEL

Selects output clock timing source. Internal clock is the default. External clocks let you select an input channel's clock to serve as the output transmit clock for channels 4-7.

Bit	Name	Description			
7–0	[no name]	0x00 Internal from PLL1	0x0A External, ch. 9 input clock		
		0x01 External, ch. 0 input clock	0x0B External, ch. 10 input clock		
		0x02 External, ch. 1 input clock	0x0C External, ch. 11 input clock		
		0x03 External, ch. 2 input clock	0x0D External, ch. 12 input clock		
		0x04 External, ch. 3 input clock	0x0E External, ch. 13 input clock		
		0x05 External, ch. 4 input clock	0x0F External, ch. 14 input clock		
		0x06 External, ch. 5 input clock	0x10 External, ch. 15 input clock		
		0x07 External, ch. 6 input clock	0x20 External, EXTCLKIN input clock		Not
		0x08 External, ch. 7 input clock	0x40 Enable PLL0 out on EXTCLKIN		available
		0x09 External, ch. 8 input clock	for board under test (testing only)		for ECL

0x37 Gated Channels Clock Select

Access / Notes: 8-bit read-write / SSD8IO_MF_CLK_SEL

Selects output clock timing source for channels 0 and 1. The internal clock is the default. External clocks let you select an input channel's clock to serve as the output transmit clock for channels 0 and 1.

Bit	Name	Description			
7–0	[no name]	0x00 Internal from PLL1	0x0A External, ch. 9 input clock		
		0x01 External, ch. 0 input clock	0x0B External, ch. 10 input clock		
		0x02 External, ch. 1 input clock	0x0C External, ch. 11 input clock		
		0x03 External, ch. 2 input clock	0x0D External, ch. 12 input clock		
		0x04 External, ch. 3 input clock	0x0E External, ch. 13 input clock		
		0x05 External, ch. 4 input clock	0x0F External, ch. 14 input clock		
		0x06 External, ch. 5 input clock	0x10 External, ch. 15 input clock		
		0x07 External, ch. 6 input clock	0x20 External, EXTCLKIN input clock		Not
		0x08 External, ch. 7 input clock	0x40 Enable PLL0 out on EXTCLKIN		available
		0x09 External, ch. 8 input clock	for board under test (testing only)		for ECL

0x38 Custom Receive Channel 32-byte Pad Enable

Access / Notes: 8-bit read-write / [no access name]

Enables extra padding logic on specified receive channel(s).

Enabling this functionality pads a DMA'd frame to a 32-byte boundary..

Bit	Name	Description
7–4	[no name]	Not used.
3–2	[no name]	Enable padding per custom receive channel.
1–0	[no name]	Not used.

0x39 – 3B Channel 0 Transmit Frame Length

Access / Notes: 24-bit read-write / [no access name]

Sets the expected transmit frame length in bytes for channel 0.

Bit	Name	Description
23–0	[no name]	Transmit frame length (in bytes).

0x3C – 3E Channel 1 Transmit Frame Length

Access / Notes: 24-bit read-write / [no access name]

Sets the expected transmit frame length in bytes for channel 1.

Bit	Name	Description
23–0	[no name]	Transmit frame length (in bytes).

0x3F User Pause

Access / Notes: 4-bit read-write / [no access name]

Forces a channel to pause (i.e., not be ready) so it will not transmit or receive.

Bit	Name	Description
7–4	[no name]	Reserved.
3–0	[no name]	For each channel (3-0), manually pause.

Pinouts

The SSD8IO-MF2 configuration package connects your device to the PCI CDa main board via the 80-pin connector, as shown below. Signals labeled “not used” are connected, and can be accessed by your firmware.

Pin	Signal		Loopback	Pin	Signal	
1	ground			41	ground	
2	not used			42	not used	
3	CH2D+	IN		43	CH0D+	OUT
4	CH2D-			44	CH0D-	
5	CH2CLK+	IN		45	CH0CLK+	OUT
6	CH2CLK-			46	CH0CLK-	
7	CH3D+	IN		47	CH1D+	OUT
8	CH3D-			48	CH1D-	
9	CH3CLK+	IN		49	CH1CLK+	OUT
10	CH3CLK-			50	CH1CLK-	
11	CH6D+			51	CH4D+	
12	CH6D-			52	CH4D-	
13	CH6CLK+			53	CH4CLK+	
14	CH6CLK-			54	CH4CLK-	
15	CH7D+			55	CH5D+	
16	CH7D-			56	CH5D-	
17	CH7CLK+			57	CH5CLK+	
18	CH7CLK-			58	CH5CLK-	
19	EXTCLKIN+			59	EXTCLKIN-	
20	not used			60	not used	
21	not used			61	not used	
22	not used			62	not used	
23	not used			63	not used	
24	CH8D+	Ch 0 - OUT		64	CH10D+	Ch 2 - IN
25	CH8D-	FRM_L		65	CH10D-	FRM_L
26	not used			66	not used	
27	not used			67	not used	
28	CH9D+	Ch 1 - OUT		68	CH11D+	Ch 3 - IN
29	CH9D-	FRM_L		69	CH11D-	FRM_L
30	not used			70	not used	
31	not used			71	not used	
32	CH12D+	Ch 0 - IN		72	CH14D+	CH 2 - OUT
33	CH12D-	RDY		73	CH14D-	RDY
34	not used			74	not used	
35	not used			75	not used	
36	CH13D+	Ch 1 - IN		76	CH15D+	CH 3 - OUT
37	CH13D-	RDY		77	CH15D-	RDY
38	not used			78	not used	
39	not used			79	not used	
40	ground			80	ground	

Revision Log

Below is a history of modifications to this guide for the SSD8IO-MF2 configuration package.

Date	Rev	By	Pp	Detail
20110817	01	PH, SB	6	• Added register 0x38.
20110601	00	PH	All	• Created new guide.