Excellent Integrated System Limited

Stocking Distributor

Click to view price, real time Inventory, Delivery & Lifecycle Information:

Xilinx
HW-XGI-VIDEO-US

For any questions, you can email us directly:
sales@integrated-circuit.com

Video Input/Output Daughter Card

User Guide

UG235 (v1.2.1) October 31, 2007

Revision History

Video Input/Output Daughter Card
UG235 (v1.2.1) October 31, 2007

The following table shows the revision history for this document.

	Version	Revision
01/25/06	1.0	Initial Xilinx release.
02/13/06	1.1	Added two sentences to pages 17 and 43.
02/23/07	1.2	Corrected 3 pins and column 6 heading in Table A-2.
10/31/07	1.2.1	Defined the following acronyms on p. 48: EAV, CRC, NTSC, and PAL.

Contents

Preface: About This Guide

Guide Contents 11
Additional Resources 11
Conventions 12
Typographical 12
Online Document 13

Chapter 1: VIODC Overview

Introduction 15
Video Interface Support 16

Chapter 2: VIODC to ML402 Card Interface

VIOBUS Clocking 19
VIOBUS Signal Definitions 20

Chapter 3: Component and S-Video Interfaces

Overview 21
ADV7403 Video Decoder 21
ADV7321 Video Encoder 22
Video Signal Input and Output Conditioning 22
S-Video Input and Output 22
S-Video Input 22
S-Video Input Signal Conditioning 22
ADV7403 S-Video Input 24
S-Video Output 24
ADV7321 S-Video Output 24
S-Video Output Signal Conditioning 24
Composite Video Input and Output 24
Composite Video Input 25
Composite Video Input Conditioning Circuit 25
ADV7403 Composite Video Input 25
Composite Video Output 25
ADV7321A Composite Video Output 25
Composite Video Conditioning Circuit 25
Component Video Input and Output 26
Component Video Input 26
Input Signal Conditioning 26
ADV7403 Connection to FPGA 26
Component Video Output 27
FPGA to ADV7321 Connection 27
Analog Output Signal Conditioning 28
ADV7403 Configuration Modes 28
ADV7321A Configuration Modes 30

Chapter 4: DVI/VGA Input Interface

Interface Description . . . 33
- DVI Connectivity on VIODC . . . 33
- Signals . . . 33
- DVI Interface . . . 34
- VGA interface . . . 34
- Display Data Channel . . . 34

AD9887 Overview . . . 34
- Analog Interface . . . 34
- Digital Interface . . . 34

VGA Standard Overview . . . 35

Setting the PLL and Phase . . . 36
- Setting Black Levels . . . 37
- Setting Gain . . . 37

Bus Interface . . . 37
- DVI Input . . . 38
- I2C Initialization Table (in Hex) . . . 38
- DVI . . . 40

References to VGA, DVI Standards . . . 40

Chapter 5: DVI/VGA Output Interface

Overview . . . 41

TPF410 I2C Configuration . . . 42

Chapter 6: SDI Interface

Introduction . . . 43

Reference Clocks . . . 43

SDI Receiver . . . 44

PicoBlaze Controller for the ADV7321B Video Encoder . . . 45

SDI Transmitter . . . 48

References . . . 49

Chapter 7: Image Sensor Camera Interface

LVDS Camera Interface . . . 51

Camera Interface Signals . . . 51

Chapter 8: Attaching the VIODC to the ML40x Development Board

Appendix A: Reference Information

Schematic and Data Sheet Links . . . 55

VIOBUS Pinouts . . . 56

Appendix B: VSK I/O Connector Location Pictures

VIODC Connectors . . . 63

LVDS Camera 66
ML402 Board 67

Schedule of Figures

Chapter 1: VIODC Overview

Figure 1-1: **VIODC Attached to an ML402 Platform** 15
Figure 1-2: **VIODC Block Diagram** 16

Chapter 2: VIODC to ML402 Card Interface

Figure 2-1: **VIOBUS Clocking** 19

Chapter 3: Component and S-Video Interfaces

Figure 3-1: **S-Video, Composite, and Component Video Input and Output Block Diagram** 21
Figure 3-2: **S-Video, Composite, and Component Input and Output Signal Conditioning Circuit** 23
Figure 3-3: **Component Video Input** 26
Figure 3-4: **Connections from ADV7403 Video Decoder to XC2VP4 FGPA** 27
Figure 3-5: **Component Video Output Block Diagram** 27

Chapter 4: DVI/VGA Input Interface

Figure 4-1: **DVI Connectivity on VIODC Block Diagram** 33
Figure 4-2: **VGA Interface** 35
Figure 4-3: **Synchronization Signaling** 35
Figure 4-4: **Pixel Sampling** 36
Figure 4-5: **Ideal ADC Sampling Positions** 37

Chapter 5: DVI/VGA Output Interface

Figure 5-1: **DVI/VGA Video Output Interface Block Diagram** 41

Chapter 6: SDI Interface

Figure 6-1: **SDI Receiver Block Diagram** 44
Figure 6-2: **ADV7321B Debugger** 46
Figure 6-3: **SDI Transmitter Block Diagram** 48

Chapter 7: Image Sensor Camera Interface

Figure 7-1: **LVDS Camera Interface** 51
Figure 7-2: **Camera Clock** 52

Chapter 8: Attaching the VIODC to the ML40x Development Board

Figure 8-1: **Configuration Jumper Locations on the ML40x Bottom, Configured for VIODC Mounted to an ML402 Board** 54

Figure 8-2: **Configuration Jumper Locations on the VIODC Top, Configured for VIODC Mounted to an ML402 Board** 54

Appendix A: Reference Information

Appendix B: VSK I/O Connector Location Pictures

Figure B-1: **VIODC Rear View** 63
Figure B-2: **VIODC Left Side View** 64
Figure B-3: **VIODC Right Side View** 65
Figure B-4: **LVDS Camera** 66
Figure B-5: **ML402 Board** 67
Figure B-6: **ML402 Evaluation Platform** 68

Schedule of Tables

Chapter 1: VIODC Overview

Chapter 2: VIODC to ML402 Card Interface

Table 2-1: **VIOBUS Signal Definitions** 20

Chapter 3: Component and S-Video Interfaces

Table 3-1: **Configuration Modes for ADV7403 Video Decoder Chip** 28
Table 3-2: **Configuration Modes for ADV7321A Video Encoder Chip** 30

Chapter 4: DVI/VGA Input Interface

Table 4-1: **VGA Standards** 36
Table 4-2: **Analog VGA60** 38
Table 4-3: **Analog XGA60** 39
Table 4-4: **Analog SXGA60** 39
Table 4-5: **Analog UXGA60** 39
Table 4-6: **DVI** 40

Chapter 5: DVI/VGA Output Interface

Table 5-1: **Configuration Modes for TPF410 I2C Video Encoder Chip** 42

Chapter 6: SDI Interface

Table 6-1: **RocketIO Reference Clock Generation** 44
Table 6-2: **ADV7321B Register Settings for HD** 46
Table 6-3: **ADV7321B HD Mode Register 1 (0x10) Settings by Video Format** 47
Table 6-4: **ADV7321B Register Settings for NTSC** 47
Table 6-5: **ADV7321B Register Settings for PAL** 47

Chapter 7: Image Sensor Camera Interface

Table 7-1: **Camera Interface Signals** 51

Chapter 8: Attaching the VIODC to the ML40x Development Board

Table 8-1: **Required Jumper Positions** 53

Appendix A: Reference Information

Table A-1: **VIODC ICs** 55
Table A-2: **VIOBUS Signals XGI Header Connections** 56
Table A-3: **VIOBUS ML402 FPGA Connections** 58
Table A-4: **VIOBUS VIODC FPGA Connections** 60

Appendix B: VSK I/O Connector Location Pictures

About This Guide

This guide describes the Video Input and Output Daughter Card (VIODC), a standard video interface card that is compatible with the Xilinx ML401, ML402, and ML403 development platforms.

Guide Contents

This manual contains the following chapters:

- Chapter 1, "VIODC Overview" – provides an overview of the VIODC, interfaces, and I/Os.
- Chapter 2, "VIODC to ML402 Card Interface" – describes the VIODC to ML402 card interface.
- Chapter 3, "Component and S-Video Interfaces" – describes the High Definition (HD) and Standard Definition (SD) component video and S-video interfaces.
- Chapter 4, "DVI/VGA Input Interface" – provides an overview of the VGA and DVI input interface.
- Chapter 5, "DVI/VGA Output Interface"– provides an overview of the VGA and DVI output interface.
- Chapter 6, "SDI Interface"– provides an overview of the SDI video interface.
- Chapter 7, "Image Sensor Camera Interface"– describes the Irvine Sensors LVDS RGB camera interface.
- Chapter 8, "Attaching the VIODC to the ML40x Development Board" – provides information necessary for proper attachment of the VIODC to a ML40*x* development board.
- Appendix A, "Reference Information"– contains VIODC pinout information.
- Appendix B, "VSK I/O Connector Location Pictures" – shows I/O connection locations.

Additional Resources

To find additional documentation, see the Xilinx website at:

http://www.xilinx.com/literature/index.htm.

To search the Answer Database of silicon, software, and IP questions and answers, or to create a technical support Web Case, see the Xilinx website at:

http://www.xilinx.com/support.

Conventions

This document uses the following conventions. An example illustrates each convention.

Typographical

The following typographical conventions are used in this document:

Convention	Meaning or Use	Example
`Courier font`	Messages, prompts, and program files that the system displays	`speed grade: - 100`
`Courier bold`	Literal commands that you enter in a syntactical statement	**`ngdbuild`** *design_name*
Helvetica bold	Commands that you select from a menu	**File → Open**
	Keyboard shortcuts	**Ctrl+C**
Italic font	Variables in a syntax statement for which you must supply values	**`ngdbuild`** *design_name*
	References to other manuals	See the *Development System Reference Guide* for more information.
	Emphasis in text	If a wire is drawn so that it overlaps the pin of a symbol, the two nets are *not* connected.
Square brackets [ ]	An optional entry or parameter. However, in bus specifications, such as **`bus[7:0]`**, they are required.	**`ngdbuild`** [*option_name*] *design_name*
Braces { }	A list of items from which you must choose one or more	**`lowpwr =`**{**`on`**|**`off`**}
Vertical bar |	Separates items in a list of choices	**`lowpwr =`**{**`on`**|**`off`**}

Vertical ellipsis
.
.
.
	Repetitive material that has been omitted	`IOB #1: Name = QOUT'`
		`IOB #2: Name = CLKIN'`
		.
		.
		.

Horizontal ellipsis . . .	Repetitive material that has been omitted	**`allow block`** *block_name loc1 loc2 ... locn;*

Online Document

The following conventions are used in this document:

Convention	Meaning or Use	Example
Blue text	Cross-reference link to a location in the current document	See the section "Additional Resources" for details. Refer to "Title Formats" in Chapter 1 for details.
Red text	Cross-reference link to a location in another document	See Figure 2-5 in the *Handbook*.
Blue, underlined text	Hyperlink to a website (URL)	Go to http://www.xilinx.com for the latest speed files.

Chapter 1

VIODC Overview

Introduction

The Video Input and Output Daughter Card (VIODC) is a standard video interface card for Xilinx development platforms. It is compatible with ML401, ML402, and ML403 boards and other future Xilinx development platforms. The VIODC is shown in Figure 1-1 mounted on a ML402 platform. The VIODC provides access to high definition (HD) and standard definition (SD) video streams, as well as computer graphics video interfaces, such as VGA over DVI and SDI interfaces.

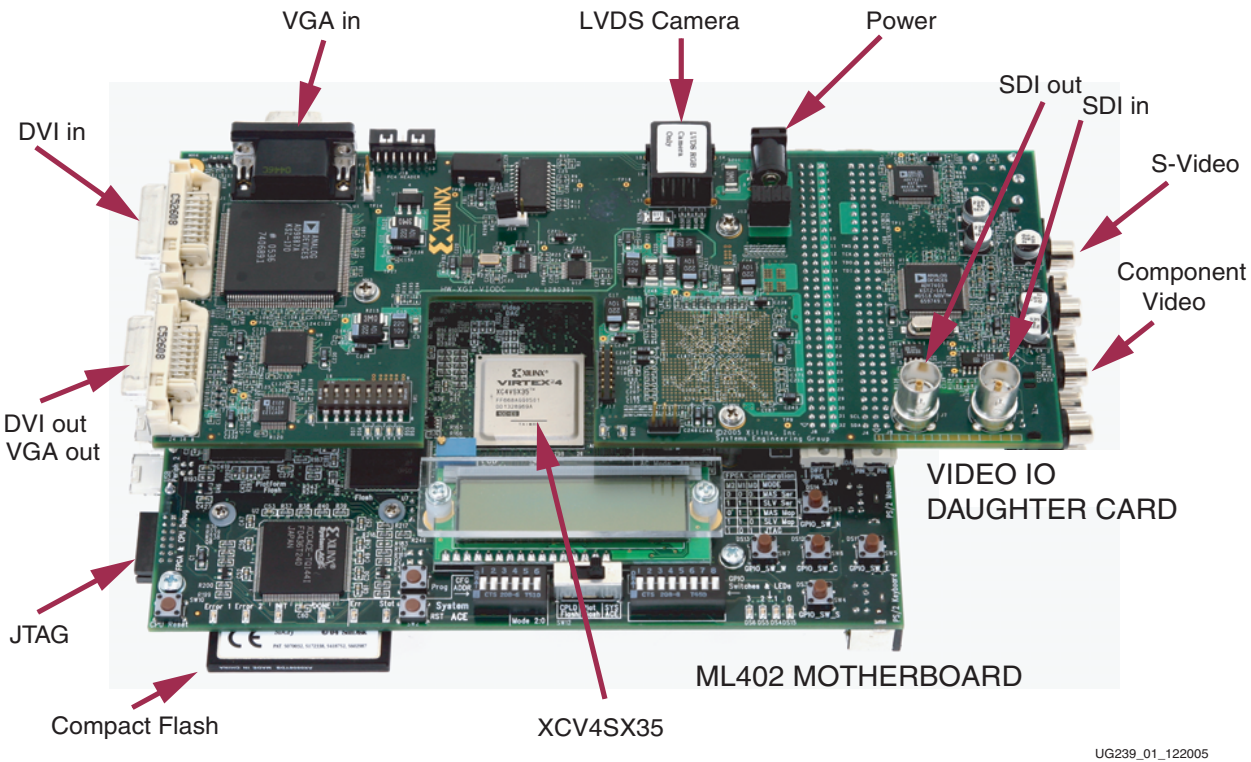

Figure 1-1: **VIODC Attached to an ML402 Platform**

Figure 1-2 shows a block diagram of the VIODC card. The VIODC consists of a number of video interface ICs connected to a Xilinx XCV2P7 FPGA. The VIODC is a daughter card which plugs onto a Xilinx ML40x FPGA platform via the XGI connector. The XGI connector provides a 64-signal bus between the ML40*x* and the VIODC. Collectively these signals are called the VIOBUS in this document.

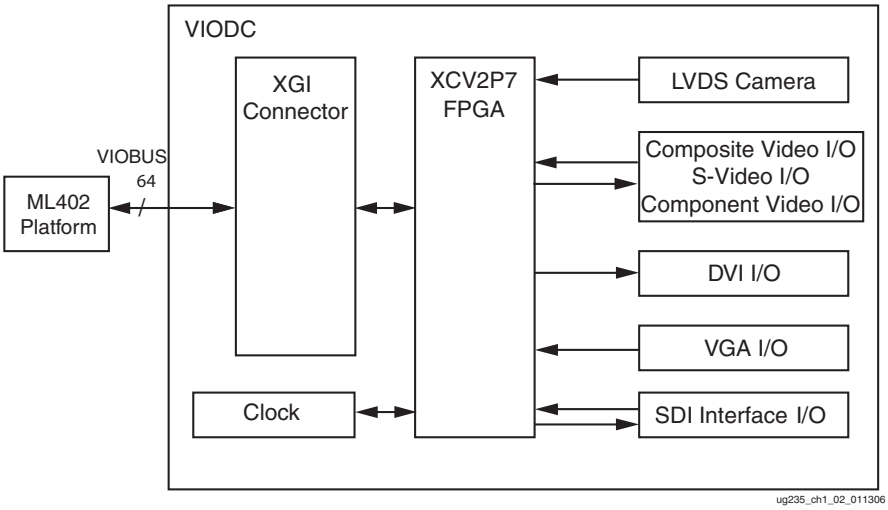

Figure 1-2: **VIODC Block Diagram**

Video Interface Support

The VIODC supports the following video interfaces:

- **LVDS Camera Input Port** – The LVDS camera input port supports the Irvine Sensors LVDS RGB camera with a Micron MT9V022 1/3 inch CMOS image sensor. The camera provides 752 x 480 pixels at 60 Hz progressive scan. It features low noise and very high dynamic range. The interface is implemented using LVDS signaling over standard Cat-6 Ethernet cables. Note that the LVDS camera interface is not compatible with Ethernet.
- **S-Video and Composite Video** – The VSK supports S-Video inputs and outputs. These interfaces can be configured to support NTSC, PAL, and virtually any other SD video format. The S-video input interface is supported by the ADV7403 decoder IC and output by the ADV7321 encoder IC. In addition to the encoder and decoder, analog filters are used to limit the video bandwidth.
- **Component Video I/O** – The component video I/O use standard RCA connectors to provide HD video the VSK. Component video is encoded as YPbPr video channels. The component video input on the supports 1080I, 720P, and 525P video standards. The Component video interface devices on the VSK support 10-bit digital video. Component video input is supported by the ADV7403 IC decoder IC and output by the ADV7321 encoder IC and analog filter sections.
- **DVI Digital Video I/O** – The VSK supports DVI video inputs and outputs. DVI is commonly used to interface to flat panel displays and computer graphics cards. The VSK DVI interfaces supports up to a pixel clock of up to 165 MHz. In addition to computer graphics, DVI is also used to carry HD video and is commonly found in high-end consumer video equipment, such as plasma displays, and can be found on some DVD players. The DVI ports can also be connected to HDMI interfaces by using a DVI/HDMI adapter. A TP410 IC is used to support DVI output and an AD9887 IC provides DVI input.
- **VGA Interface** – VGA input and outputs are is available on the VIODC card. The VGA output is routed to the analog output pins of the DVI output connector. It is sourced by an ADV7123 10-bit DAC. VGA input is captured by the AD9887 IC.

- **SDI Video Interface** – A complete SDI video interface capable of supporting both SD and HD video rates is available on the VSK. The SDI standard is a high-speed serial interface used to carry digital video over coax cable. It is generally used in a studio environment. The SDI system includes cable equalizers and Genlock circuitry. (The VSK is a demonstration platform only. For HD-SDI verification and compliance, Xilinx recommends using the Cook Technologies SDV board).
- **Clock Generator** – The clock generator section is used to generate standard video clock frequencies. It is based on an ICS 1523 clock generator IC.
- **XCV2P7 FPGA** – The VSK also includes a Xilinx XCV2P7 FPGA, which is used to interface to the various video interfaces, as well as the ML402 main board. It features Multi-Gigabit Transceivers (MGTs), which are used to support the SDI interface. It also enables the VIODC to be used in a stand-alone fashion.
- **XGI Connector**– The XGI connector is a standard connector interface used on XiIinx ML40*x* FPGA development platforms. The XGI connector is used to connect to the VIODC to a standard FPGA development platform, such as the ML402. The signals consist of 32 single-ended LVCMOS25 signals and 32 signals that can be configured as either 32 LVCMOS25 signals or 16 LVDS signal pairs. The LVDS pairs are length matched and routed as pairs on the PCB. In addition, 5V power is passed up to the VIODC over the XGI connector.
- **VIOBUS** – The Video Starter Kit (VSK) uses the VIODC as a Video I/O interface. For compatibility with the VSK, the 64 XGI signals have been specified as a bus named the VIOBUS. In this use, the signals on the VIODC XGI connector have been specified as a set of buses that transmit a 27-bit digital video channel from the VIODC to the FPGA development platform and a 27-bit bus to transmit a similar digital video channel from the ML40*x* to the VIODC. Each video channel consists of a 24-bit digital video bus, HSYNC, VSYNC and a clock enable signal. The pinout for the VIOBUS can be found in Appendix A, "Reference Information." This implementation of the interface runs synchronous to the interface clock supplied by the ML402 board. The VIOBUS also specifies an LVDS clock, a reset signal, an I2C interface, and a 4-pin serial bus.

Chapter 2

VIODC to ML402 Card Interface

When the VIODC is used as part of the Video Starter Kit (VSK) from Xilinx, the 64-pin XGI connector connects the VIODC to a ML402 card to communicate with the VIODC card. When the VIODC is used with the VSK, the 64 XGI signals are allocated to a bus named the VIOBUS, which serves the following functions:

- Transfers video data between the ML402 card and the VIODC card.
- Provides a clock to the VIODC card.
- Provides reset to the VIODC card.
- Provides a low-pin count serial bus to access registers on the VIODC.
- Provides an I2C bus (an industry standard 2-pin serial data bus used to communicate and configure ICs) to access registers on the VIODC video interface FPGA.

VIOBUS Clocking

The VIOBUS uses a simple synchronous interface running at 100 MHz (Figure 2-1).

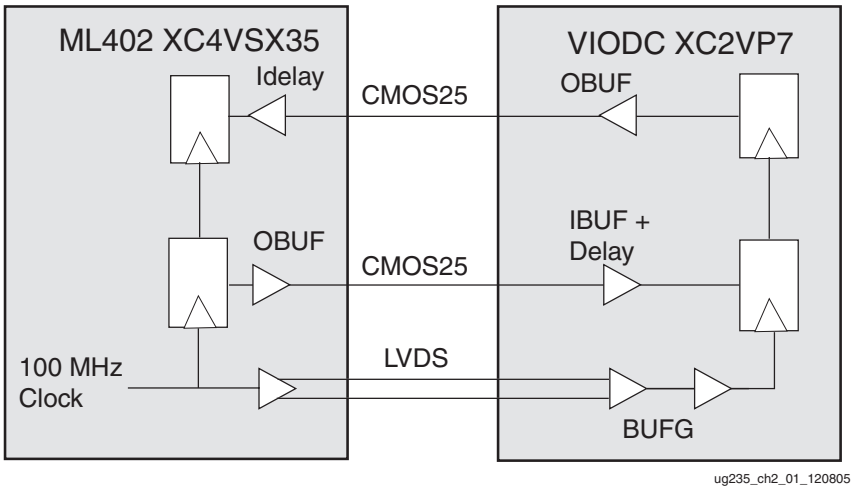

Figure 2-1: **VIOBUS Clocking**

A clock is passed from the ML402 FPGA to the VIODC using differential signaling. All data signals are single ended. The VIODC transmits data back to the ML402 FPGA using the received clock. Data returning back from the VIODC is clocked into the ML402 FPGA using the internal 100 MHz clock.

Future VIODC bus interfaces may implement a differential bus using the 16 differential pairs available on HDR2 and more sophisticated clocking.

VIOBUS Signal Definitions

Table 2-1: **VIOBUS Signal Definitions**

Signal	Description	nbits	Type	Target Speed	Source FPGA	XGI Pins
VIO Data Bus (a moderate-speed single-ended bus)						
vio_up[25:0]	Data bus to the VIODC	26	LVCMOS25	100 MHz	ML402	hdr1[20:2], hdr2[2:32]
vio_up_ena	Pixel enable for vio_up[25:0]	1	LVCMOS25	100 Mhz	ML402	hdr1[22]
vio_dn[25:0]	Data bus from the VIODC	26	LVCMOS25	100 MHz	VIODC	hdr1[42:24], hdr2[64:34]
vio_dn_ena	Pixel enable for vio_up[25:0]	1	LVCMOS25	100 MHz	VIODC	hdr1[44]
Sport Serial Bus (used to configure registers in the VIODC FPGA)						
vio_sport_up	Sport write data (16-bit data, 16-bit address)	1	LVCMOS25	10 MHz	ML402	hdr1[54]
vio_sport_dn	Sport return data	1	LVCMOS25	10 MHz	VIODC	hdr1[52]
vio_sport_sync	Sport sync pulse	1	LVCMOS25	10 MHz	ML402	hdr1[50]
vio_sport_clk	Sport clock	1	LVCMOS25	10 MHz	ML402	hdr1[48]
I2C Serial Bus (used to configure registers in the video devices)						
vio_i2c_sda_up	I2C write data	1	LVCMOS25	400 kHz	ML402	hdr1[60]
vio_i2c_sda_dn	I2C return data	1	LVCMOS25	400 kHz	VIODC	hdr1[58]
vio_i2c_scl_up	I2C clock signal	1	LVCMOS25	400 kHz	ML402	hdr1[56]
Miscellaneous						
vio_reset	Active High reset to VIODC	1	LVCMOS25	10 MHz	ML402	hdr1[46]
Clock						
vio_up_clk_lvds_P,N		1	LVDS25	400 MHz	ML402	hdr1[64:62]]

Refer to the VIOBUS pinout in Appendix A, "Reference Information" for signal locations.

Chapter 3

Component and S-Video Interfaces

Overview

The VIODC board supports input and output for S-video, composite, and component video. Figure 3-1 is a simplified block diagram of input and output.

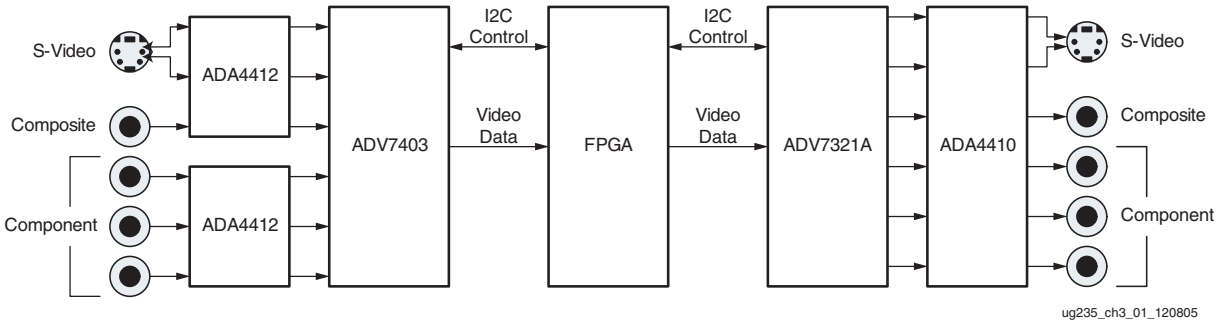

Figure 3-1: **S-Video, Composite, and Component Video Input and Output Block Diagram**

Input signals are conditioned by a combination of passive components and the ADA4412 device. The conditioned input signals are converted to digital signals by the ADV7403 video decoder device. Digital video output data stream from the FPGA is converted to analog signals by the ADV7321 video encoder device. The analog output signals are conditioned by the ADA4410 device.

The video decoder, ADV7403 from Analog Devices, is responsible for converting analog video signals into a representative digital video data stream. The video encoder, ADV7321A also from Analog Devices, is responsible for the generation of S-Video, composite, and component analog video signals from a digital video data stream. Both devices offer an I^2C control serial bus for control and ancillary data.

ADV7403 Video Decoder

The ADV7403 is a high quality, single chip, multiple format video decoder and graphics digitizer. This multiple format decoder automatically supports the conversion of PAL, NTSC, and SECAM standards in the form of composite or S-video into a digital ITU-R BT.656 format. The component processor is capable of decoding/digitizing a wide selection of video formats in any color space. Component video standards supported include: 525i, 625i, 525p, 625p, 720p, 1080i and many other HD standards, as well as graphic digitization from VGA to SXGA. Converted input signals are output to the output pixel port, which is connected directly to the FPGA. Under user control, the output pixel port is configurable to conform to multiple different standards. Selection of the format is done through commands written to the device over the I^2C bus and affects the pins

definitions. For complete details, refer to the Analog Devices data sheet found at www.analog.com

ADV7321 Video Encoder

The Analog Devices ADV7321 video encoder device is a single monolithic chip that performs multiple format digital-to-analog video encoder functions. Both standard and high definition input formats are supported including: SMPTE 293M (525p), BTA T-1004 EDTV2 (525p), CCIR-656, and SMPTE 274M. Multiple output standards for both SD and HD are also supported including: YPrPb HDTV (EIA 770.3), RGB, RGBHV, YPrPb progressive scan (EIA-770.1, EIA-770.2) and component YPrPb (SMPTE/EBU N10). 4:2:2 or 4:4:4 data format is supported for HDTV. For all standards, external horizontal, vertical and blanking signals or EAV/SAV timing codes control the insertion of appropriate synchronization signals into the digital data stream and, therefore, the analog output signal.

The ADV7321 provides user configuration options through an I^2C bus, which enables access to a large number of configuration registers. Under user control, the device pins are reconfigured to match the operation selected. For instance, SD 8-bit mode configuration only, the data input port S7-S0 would be used to transfer in a multiplexed fashion the digital video data stream into the device. The Y and C buses would not be used. Refer to Analog Devices ADV7321 data sheet for further details.

Video Signal Input and Output Conditioning

Each of the video input and output signals must be conditioned to ensure that the physical interfaces meet impedance and electrical specification for each individual video standard. Figure 3-3 illustrates the input and output conditioning circuits used for S-video, composite and component input and output signals.

S-Video Input and Output

S-Video Input

Connector J20 provides input and output of S-Video compatible signals. For the input, the Y (intensity) and C (color) signals are each conditioned and input into the ADV7403 video decoder to create a digital video data stream output, which is transferred to the Xilinx XC2VP4 FPGA for handling. Generation of S-Video output starts with a digital video stream coming from the FPGA, written into the ADV7321A video encoder to product the Y/C analog outputs, which are conditioned and output to the J20 S-Video connector.

S-Video Input Signal Conditioning

S-Video input signals are first conditioned using two identical circuits illustrated in Figure 3-2. This circuit contains both passive and active components, including the Analog Devices ADA4412 device. This conditioning circuit insures that the input signal impedance matches the S-Video (IEC 60933-5) specification and signal levels required by the ADV7403.

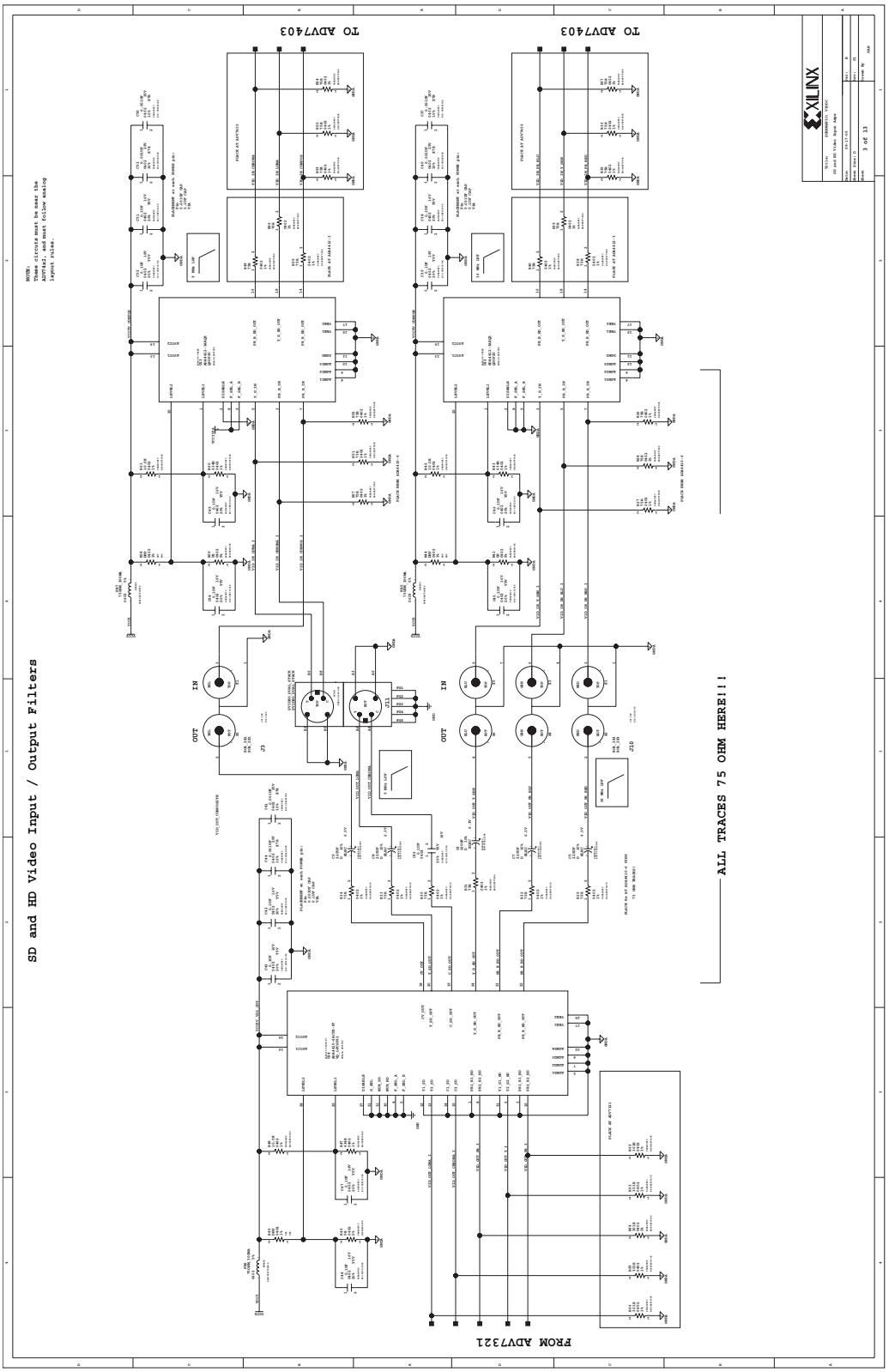

Figure 3-2: **S-Video, Composite, and Component Input and Output Signal Conditioning Circuit**

ADV7403 S-Video Input

The Y (intensity) and C (color) conditioned signals are input into the A12 and A10 of the ADV7403 twelve input analog multiplexer, which routes each of the selected input signals to one of the ADCs for conversion. Fully automatic detection and selection of all worldwide standards, (PAL, NTSC and SECAM) is provided as well as vertical blanking processing for Teletext, Closed Caption and wide screen signaling. For full details regarding the ADV7403 device refer to the Analog Devices datasheet.

The digital data stream generated from the conversion of the S-Video signals is available to the FPGA through a 41-bit data bus and 5-bit control. An I2C bus available on the ADV7403 provides control, status and ancillary data and is directly connected to the FPGA. For S-Video configuration there are 6 different interface configurations that use some of the lower 30 pins. The default configuration is to output YCrCb data on the 8-bit portion of the data bus from P19 to P12.

S-Video Output

Generation of S-Video output video from a digital video data stream is accomplished by the ADV7321 device. Video data is written from the XC2VP4 FPGA into the ADV7321 device, which converts from digital-to-analog values using DAC D and E. The analog output signals are conditioned to meet specification with the conditioned output going through connector J20.

ADV7321 S-Video Output

Data, video timing control and operations control bus connections between the FPGA and ADV7321 video encoder provide the digital video data stream and information needed to convert to generate analog S-Video Y/C signals. The FPGA writes the digital video data stream and control into the ADV7321, which then produces the appropriate analog output with complete video timing. The format of the data written is selectable the analog output is first conditioned and then placed on the output S-Video connector J20.

S-Video Output Signal Conditioning

Figure 3-2 details the implementation of the S-Video output conditioning circuit following the ADV7321 Y/C analog signal generation. This conditioning circuit is composed of both active and passive components, with the ADA4410 device providing active circuits and is designed to meet the specification IEC 60933-5 requirements for S-Video.

Composite Video Input and Output

Composite video is the format of an analog television (picture only) signal before it is combined with a sound signal and modulated onto an RF carrier. It is usually in a standard format such as NTSC, PAL, or SECAM. It is a composite of three source signals called Y, U and V with sync pulses. Y represents the brightness or *luminance* of the picture and includes synchronizing pulses, so that by itself it could be displayed as a monochrome picture. U and V between them carry the color information.

Composite video input and output is supported on the VIODC card through RCA type jack J18, this dual RCA jack has the composite video input on X1 and output on X2 and are color coded yellow. Input signals are conditioned and then presented to the ADV7403 for conversion to digital video data stream. The ADV7403 device automatically detects the video standard (PAL, NTSC, SECAM) and converts to the appropriate data stream with control information. The resulting data stream and control information is transferred to the

FPGA for further processing. A digital video data stream with control is converted to a composite video stream by the ADV7321A device and associated signal conditioning circuits. The data stream and control is supplied by the FPGA.

Composite Video Input

Composite video input on connector J18 X1 is first conditioned and then converted to the digital video data stream, which is passed to the XC2VP4 for further processing. The ADV7403 device is configurable under user control to select the format of the devices pixel output port. In SD composite video mode up to 3 10-bit data busses can be used to transfer the video data.

Composite Video Input Conditioning Circuit

To insure compatibility with the specification an input conditioning circuit is inserted before the ADV7403 analog input. Impedance matching for the input signal and level matching for the analog input are assured. Figure 3-2 details the implementation of this circuit.

ADV7403 Composite Video Input

The conditioned composite video input signal is input on the 11th input of the 12 input analog multiplexer. When configured properly by the user the input will be routed to an analog-to-digital converter and automatic format detection logic to generate the digital video data stream. Configuration of the ADV7403 device is through the I2C control bus and appropriate writes to a number of registers. As part of the configuration process the user will select the format of the output data. For programming details please refer to the Analog Devices ADV7403 data sheet.

Composite Video Output

Generation of composite video output starts with a digital video data stream being written from the XC2VP4 into the ADV7321A video encoder, which produces an analog output that is conditioned and presented on connector J18 X2.

ADV7321A Composite Video Output

The XC2VP4 Xilinx FPGA provides both the digital video data stream and the configuration to the ADV7321A. Configuration of the ADV7321A defines the interface connections and active pins for the connection from the XC2VP4 and to the ADV7321A. For composite video the input format can be configured for either 8/10-bit ITU-BT.656/601 or 16/20-bit YCrCb with embedded HS, VS and FIELD codes. The input digital video data stream is then converted to an analog composite video output signal that includes all timing and control signaling.

Composite Video Conditioning Circuit

The analog output of the ADV7321A is processed by a conditioning circuit that insures that the composite output signal meets composite video drive specifications. Figure 3-2 details the composite video output circuit.

Component Video Input and Output

Component Video Input

RCA style connector J19 (X1, X3 and X5) enable input of analog component video signals, which are then converted to the digital domain by the Analog Devices ADV7403 device. Either YPrPb or RGB analog input signals are accepted. The ADV7403 devices integrated 110 MHz ADCs, with 12-bit resolution, supporting HDTV for 525p, 625p, 720p and 1080i as well as RGB graphics support from VGA to SXGA at 60 frames per second. The digitized video output is connected directly to the Xilinx FPGA through a digital data, video timing control and I^2C control busses. Figure 3-3 illustrates the VIODC composite video input configuration, through ADV7403 video decoder to Xilinx XC2VP4 FPGA.

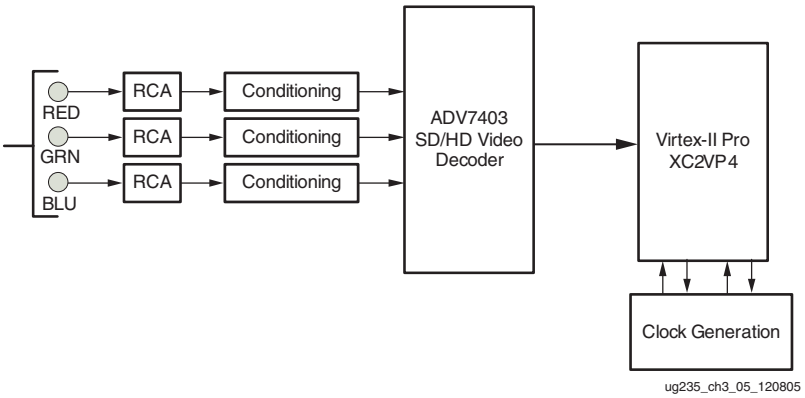

Figure 3-3: **Component Video Input**

Input Signal Conditioning

The three RCA jacks X1, X3 and X5 are color coded Red, Green and Blue respectively and form the physical connectors for the component video inputs. Conditioning of the analog input signal is done using circuit detailed in Figure 3-2.

ADV7403 Connection to FPGA

Digital connections from the ADV7403 to the Xilinx XC2VP4 FPGA consist of 42 data and 5 control signals. The data signals include three 12-bit data busses for, one for each of red, green and blue pixel values. Control signals include: horizontal and vertical frame

synchronization signals and field indicator. Figure 3-4 details the connections from the ADV7403 video decoder device to the XC2VP4 Xilinx FPGA.

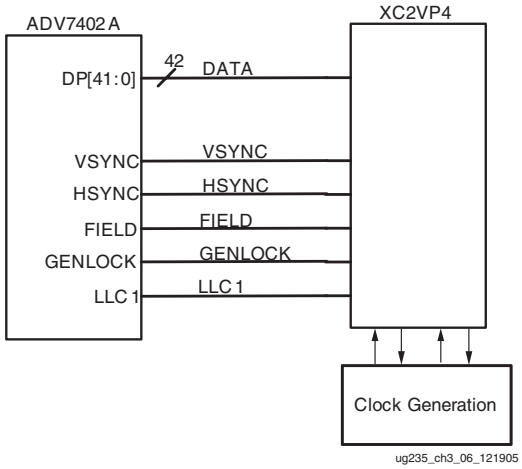

Figure 3-4: **Connections from ADV7403 Video Decoder to XC2VP4 FGPA**

Component Video Output

Compliant digital video streams are feed into the ADV7321 device by the Xilinx XC2VP4 FPGA, where it is converted to analog RGB or YPrPb using 12-bit DACs. The ADV7321 device, from Analog Devices, produces fully compliant SD/HD analog output signals, which are then conditioned and drive the RCA type jacks. The Analog Devices ADV7321 is used to generation all analog component RGB or YPrPb video output signals. Figure 3-5 is a block diagram of the component video output system on the VIODC board.

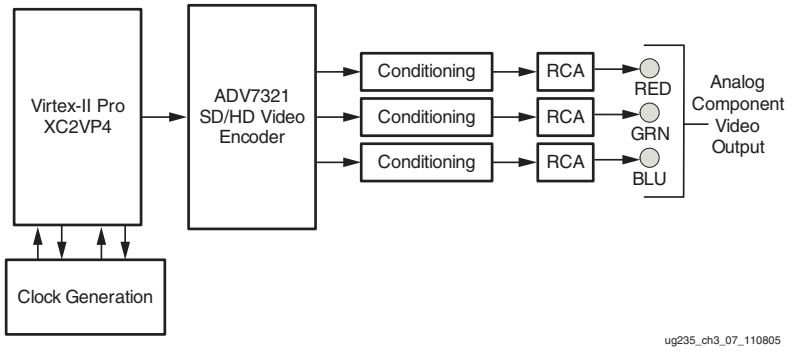

Figure 3-5: **Component Video Output Block Diagram**

FPGA to ADV7321 Connection

The Xilinx XC2VP4 FPGA drives digital video data streams, in either standard and/or high definition video format, onto three separate 10-bit wide digital input ports of the ADV7321. For all supported standards the ADV7321 generates all horizontal, vertical and blanking signals. Six high performance 12-bit digital to analog converters generate the analog output signals.

Analog Output Signal Conditioning

Output analog signals are first conditioned to meet standards requirements and then connected to the red, green and blue RCA type connectors X2, X4 and X6 respectively. Analog output is conditioned by a combination of passive and active circuits, as illustrated in Figure 3-3.

ADV7403 Configuration Modes

Refer to the ADV7403 data sheet for details configuring the ADV7403 device.

The ADV7403 is mapped to I2C address 0x40/0x41.

Table 3-1: **Configuration Modes for ADV7403 Video Decoder Chip**

Register Name	Register Address	Register Value	Description
525P			
Primary Mode	0x05	0x01	
Video Standard	0x06	0x08	[2:0] =PRIM_MODE
Enable XTAL	0x1D	0x47	[3:0]= VID_STD
ADC Power and PLL	0x3a	0x10	latch clock = 13-55 MHz
Bias Control	0x3b	0x80	External Bias Enable'
TLLC Control	0x3c	0x5c	PLL qpump
	0x6b	0xC2	[3:0]cpop_sel(1=20-bit,2=30-bit)
	0x85	0x18	Turn off SSPD as sync is on Y
	0x86	0x0b	ENABLE SDTI line count mode
	0xb3	0xfe	SDTI
ADC sw1	0xc3	0x54	[7:4]=adc1 [3:0]=adc0
ADC sw2	0xc4	0x86	[7]=sw_en, [6]=SOG [3:0]=adc2
	0x0e	0x80	Startup sequence
	0x52	0x46	
	0x54	0x00	
	0x0e	0x00	

Table 3-1: **Configuration Modes for ADV7403 Video Decoder Chip *(Continued)***

Register Name	Register Address	Register Value	Description
720P			
Primary Mode	0x05	0x01	
Video Standard	0x06	0x0a	[2:0] =PRIM_MODE
Enable XTAL	0x1D	0x47	[3:0]= VID_STD
ADC Power and PLL	0x3a	0x20	latch clock
Bias Control	0x3b	0x80	External Bias Enable'
TLLC Control	0x3c	0x5d	PLL qpump
	0x6b	0xC2	[3:0]cpop_sel(1=20-bit,2=30-bit)
	0x85	0x18	Turn off SSPD as sync is on Y
	0x86	0x0b	ENABLE SDTI line count mode
	0xb3	0xfe	SDTI
ADC sw1	0xc3	0x54	[7:4]=adc1 [3:0]=adc0
ADC sw2	0xc4	0x86	[7]=sw_en, [6]=SOG [3:0]=adc2
	0x0e	0x80	Startup sequence
	0x52	0x46	
	0x54	0x00	
	0x0e	0x00	
1080I			
Primary Mode	0x05	0x01	
Video Standard	0x06	0x0c	[2:0] =PRIM_MODE
Enable XTAL	0x1D	0x47	[3:0]= VID_STD
ADC power and PLL	0x3a	0x21	Latch clock
Bias Control	0x3b	0x80	External Bias Enable'

Table 3-1: **Configuration Modes for ADV7403 Video Decoder Chip *(Continued)***

Register Name	Register Address	Register Value	Description
TLLC control	0x3c	0x5d	PLL qpump
	0x6b	0xC2	[3:0]cpop_sel(1=20-bit,2=30-bit)
	0x85	0x18	Turn off SSPD as sync is on Y
	0x86	0x0b	ENABLE SDTI line count mode
	0xb3	0xfe	SDTI
ADC sw1	0xc3	0x54	[7:4]=adc1 [3:0]=adc0
ADC sw2	0xc4	0x86	[7]=sw_en, [6]=SOG [3:0]=adc2
	0x0e	0x80	Startup sequence
	0x52	0x46	
	0x54	0x00	
	0x0e	0x00	

Notes:

1. The ADV sw1 and sw2 are unique to the VIODC input configuration.

Refer to the ADV7403 data sheet for other video configurations.

ADV7321A Configuration Modes

Table 3-2, details the parameters setting for the internal registers of the ADV7321A Video Encoder device for each of the supported video standards.

The ADV7301 is mapped to I2C address 0x54/0x55.

Table 3-2: **Configuration Modes for ADV7321A Video Encoder Chip**

Register Name	Register Address	Register Value	Description
525P			
Power Mode	0x00	0xFE	[7]=DACA_composite [6]=DACB_luma [5]=DACC_chroma [4]=DACD_Y [3]=DACE_Pr [2]=DACF_Pb on [1]=pll_off(1=off) [0]=sleep(1=sleep)

Table 3-2: **Configuration Modes for ADV7321A Video Encoder Chip *(Continued)***

Register Name	Register Address	Register Value	Description
Input Mode	0x01	0x10	[6:4]=mode (7=PS54 6=SDHD 5=SDHD 4=10-bit 3=20-bit 2=hd, 1=ps, 0=sd,) [3]=clock_dly [2]=cb0, [0]=BTA compatibility
Mode	0x02	0x30	[5]=yuv_output [4]=rgb_out_sync [3]=use_rgb_matrix [2]=black bar'
HD Mode Reg 1	0x10	0x00	[7]macro_vision [6]blank_low [5]=720/1080i, [4]=625/525p [3:2]=sync_mode(0=hvsync,1=EAVcodes2=async), [1:0]=output_levels(
HD Mode Reg 2	0x11	0x01	[3]=tp_Field_en [2]=test_pattern_on [0]=data_valid_en
HD Mode Reg 4	0x13	0x04	[7]=dbuf [6]=4:2:2/4:4:4 [5]=SSAF [3]=sync_filter [2]=10-bit [0]=crcb
HD Mode Reg 6	0x15	0x00	[7:6]filter [5]gamma_en [4]=gamma_a/b [3]dac_swap [2]syncPrPb [1]=rgb_input
525PS			
Power Mode	0x00	0xFE	
Input Mode	0x01	0x20	
Mode	0x02	0x30	
HD Mode Reg 1	0x10	0x00	
HD Mode Reg 2	0x11	0x01	
HD Mode Reg 4	0x13	0x04	

Table 3-2: **Configuration Modes for ADV7321A Video Encoder Chip *(Continued)***

Register Name	Register Address	Register Value	Description
HD Mode Reg 6	0x15	0x00	
720P			
Power Mode	0x00	0xFE	
Input Mode	0x01	0x20	
Mode	0x02	0x30	
HD Mode Reg 1	0x10	0x20	
HD Mode Reg 2	0x11	0x01	
HD Mode Reg 4	0x13	0x24	
HD Mode Reg 6	0x15	0x00	
10801			
Power Mode	0x00	0xFE	
Input Mode	0x01	0x20	
Mode	0x02	0x30	
HD Mode Reg 1	0x10	0x68	
HD Mode Reg 2	0x11	0x01	
HD Mode Reg 4	0x13	0x04	
HD Mode Reg 6	0x15	0x00	

Refer to the ADV7321A data sheet for other video configurations.

Chapter 4

DVI/VGA Input Interface

This chapter describes the DVI and VGA input interface theory of operation. It covers the signals and presents an overview of the internal operating modes of the AD9887. Users can refer to the data sheet for more detailed information.

Interface Description

The VIODC DVI/VGA input interface allows standard PC video formats to be captured. This includes analog VGA formats and digital DVI up to 1600x1200 at 60 Hz.

DVI Connectivity on VIODC

The DVI/VGA input portion of the video input and output daughter card (VIODC) has two connectors. The first is a traditional HD15 as used by all older analog video cards and monitors. The second connector is a DVI-I connector which includes pins for both the analog VGA interface and the DVI digital interface. Note that the analog pins of the two connectors are tied together, so that only one or the other can be used at any time. See Figure 4-1.

VIODC uses Analog Devices AD9887A dual interface for flat panel displays. This part includes two very separate subsections: the analog (VGA) interface and the digital (DVI) interface. Via the I2C control bus, this part can be configured to receive either of the modes and output it in a parallel digital form to the FPGA.

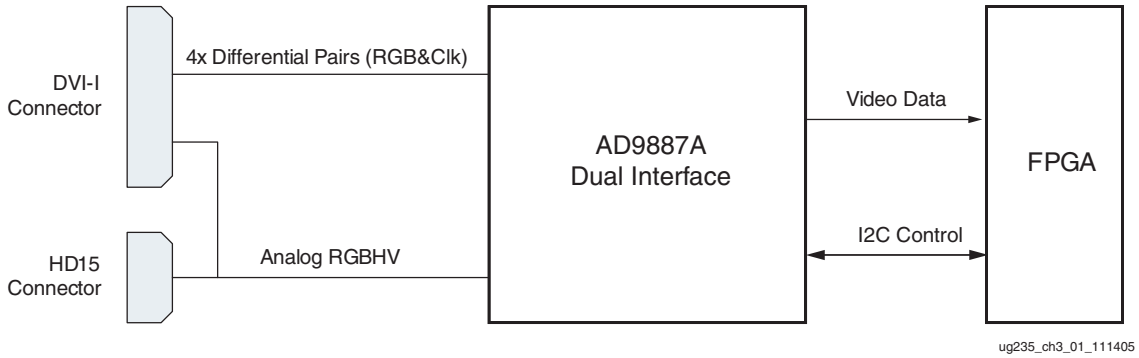

Figure 4-1: **DVI Connectivity on VIODC Block Diagram**

Signals

The AD9887A interface has a parallel digital bus interface to the FPGA for video data and an I2C control bus for configuration.

DVI Interface

The DVI interface is through the DVI-I connector. The video data is carried by four differential pairs, three data and a clock.

VGA interface

The analog VGA interface is through the either the HD15 connector or the analog pins of the DVI-I connector. The video data is carried by three analog signals along with up to two optional sync signals.

Display Data Channel

Both the DVI connector and the HD15 connector include SCL and SDA pins for the Display Data Channel (DDC). This is an I2C interface used by a computer to identify a monitor's capabilities. The graphics adapter reads the monitor's extended display identification data (EDID). This structure lists monitor manufacturer and model, supported resolutions, and other capabilities. If a graphics adapter cannot retrieve this EDID structure, it runs with a default resolution, typically 640 x 480 at 60 Hz analog. To allow higher resolutions or to enable the DVI interface, the receiver must report that it is capable of these modes. To support this, VIODC includes EEPROMs on the DDC (separate for each connector) that can be programmed with this structure.

The DDC is also used in the DVI connector for negotiating encryption keys when High-bandwidth Digital Content Protection (HDCP) is required. The AD9887A interface supports this functionality and has an EEPROM for storing these keys.

AD9887 Overview

The AD9887A dual interface includes both analog and digital interfaces. Refer to the AD9887A interface data sheet for further details.

Analog Interface

The AD9887A is a complete 8-bit 170 mega sample per second (MSPS) monolithic analog interface optimized for capturing RGB graphics signals from personal computers and workstations. Its 170 MSPS encode rate capability and full-power analog bandwidth of 330 MHz supports resolutions up to UXGA (1600 by 1200 at 60 Hz).

The analog interface includes a 170 MHz triple ADC with internal 1.25 V reference, a phase-locked loop (PLL), and programmable gain, offset, and clamp control. The user provides only a 3.3 V power supply, analog input, and HSYNC. Three-state CMOS outputs can be powered from 2.5 V to 3.3 V.

The AD9887A's on-chip PLL generates a pixel clock from HSYNC. Pixel clock output frequencies range from 12 MHz to 170 MHz. PLL clock jitter is typically 500 ps peak-to-peak at 170 MSPS. The AD9887A also offers full sync processing for composite sync and sync-on-green (SOG) applications.

Digital Interface

The AD9887A contains a DVI 1.0 compatible receiver and supports display resolutions up to UXGA (1600 _ 1200 at 60 Hz). The receiver operates with true color (24-bit) panels in 1 or

2 pixel(s)/clock mode and features an intra-pair skew tolerance of up to one full clock cycle.

With the inclusion of HDCP, displays can now receive encrypted video content. The AD9887A allows for authentication of a video receiver, decryption of encoded data at the receiver, and renewability of that authentication during transmission as specified by the HDCP v1.0 protocol.

VGA Standard Overview

The dominant video monitor interface standard for PCs has been analog VGA since its establishment in 1987. The VGA interface consists of RGB pixel levels and synchronization signaling (syncs). This interface was designed specifically for driving CRT monitors, so this interface is tailored to work with CRTs.

The video data is carried by three 75Ω transmission lines (coax), one each for red, green, and blue. When 75Ω terminated at the monitor, the voltage typically ranges from 0 mV - 700 mV. This voltage indicates 0% - 100% intensity on the associated color. In a CRT, these voltages adjust the intensity of the electron beam for each color.

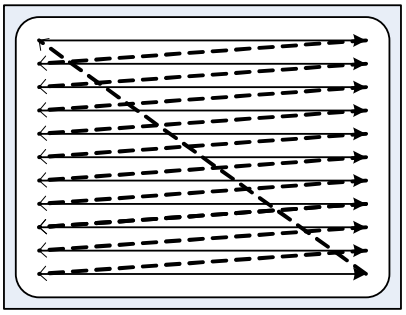

Figure 4-2: **VGA Interface**

Synchronization signaling is used to control the scanning of the electron beams, giving the stream of RGB video data positional significance. This signaling is a combination of two timing signals: horizontal synchronization and vertical synchronization (HSYNC and VSYNC). HSYNC resets the beam to the left edge of the screen, and when released it sweeps to the right. VSYNC resets the beam to the top of the screen, and when released it slowly sweeps downward. The monitor locks its vertical and horizontal sweep rates to the VSYNC and HSYNC frequencies, respectively.

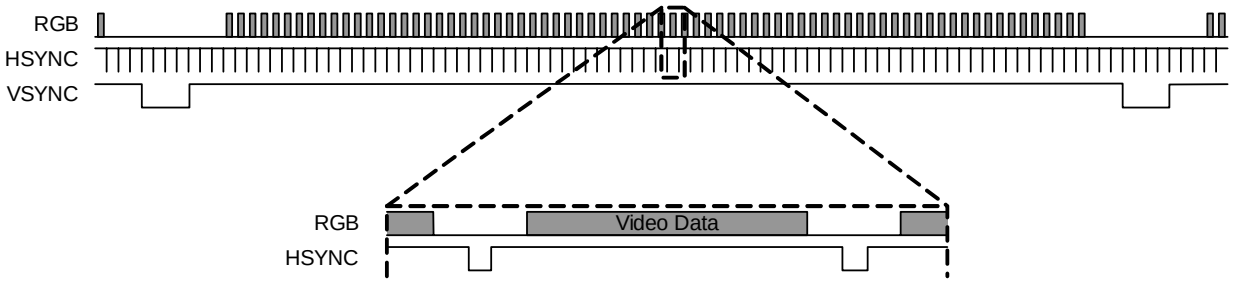

Figure 4-3: **Synchronization Signaling**

The HSYNC and VSYNC signals are critical to the VGA interface, but can be encoded in several ways. The most basic is with separate sync signals for each, increasing the number of signals to 5: RGBHV. These syncs can be active high or low, and different resolutions typically have different combinations of sync polarity. A second encoding is composite HV, with HSYNC and VSYNC combined onto a single signal. This is preformed through a logical XOR of the two signals. The end result looks like the original HSYNC signal, except that its polarity is inverted during VSYNC. This mode reduces the number of signals to 4. A third mode of encoding the sync signals is by combining the composite sync signal with the green data. This is referred to as "sync-on-green" (SOG). As mentioned previously, the typical signal levels are 0-700mV. SOG offsets this by 300mV to 300mV-1V. The drop from 300mV to 0 is used to indicate the composite sync.

Table 4-1: **VGA Standards**

	Pix Clock Freq (MHz)	HSync Freq (kHz)	Horizontal Timings (in clk cycles)				Vertical Timings (in Lines)			
			Front Porch	Sync	Back Porch	Active	Front Porch	Sync	Back Porch	Active
VGA60	40	37.9	40	128	88	800	1	4	23	600
XGA60	65	48.4	24	136	160	1024	3	6	29	768
SXGA60	108	64	48	112	248	1280	1	3	38	1024
UXGA60	162	75	64	192	304	1600	1	3	46	1200

Setting the PLL and Phase

The AD9887A digitizes the analog video waveforms using three 8-bit analog-to-digital converters. For this analog-to-digital conversion to operate properly, it must sample each pixel at the appropriate time (Figure 4-4).

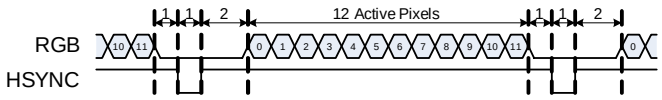

Figure 4-4: **Pixel Sampling**

To explain VGA sampling theory, it is useful to use a greatly simplified example. Figure 4-4 shows a single line from a frame with the horizontal front porch set to 1, the sync length set to 1, and the horizontal back porch set to 2. The line has 12 active pixels. In order to receive the video data, these are the only signals available.

Figure 4-5 illustrates the ideal ADC sampling positions for this line. To generate these sample times, the AD9887A includes a PLL that locks to the incoming HSYNC, multiplying the HSYNC frequency by a factor set by the feedback divider. The frequency multiplication factor is set to the total number of clock cycles per HSYNC period. For the Figure 4-5 example, the multiplication value is 1+1+2+12 = 16. The PLL in the AD9887A is free-running, so ADC samples occur during blanking (gray arrows) as well as active video

(black arrows). It is also important to note that the VCO frequency range and charge pump currents must be configured to match the expected frequency.

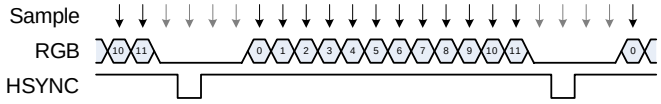

Figure 4-5: **Ideal ADC Sampling Positions**

Setting the feedback divider correctly results in the correct sample frequency, but the sample phase must also be set. Under ideal conditions, the phase delay would be T/2, to sample at the center of the pixels. This makes T/2 a reasonable default value. Under actual operating conditions, there can be skew between the HSYNC signal and the video data thus requiring phase adjustment.

The process above is quite straightforward if one knows the proper settings for a given source. VESA timing standards are a good place to start for standard resolutions. Unfortunately, graphics adapters often do not follow these exactly. In this case, the PLL divider value and phase must be adjusted to properly digitize the source. This process is beyond the scope of this document.

Setting Black Levels

The primary mechanism for setting the black level is the input clamp. This clamps the input to ground, resulting in a DC offset in the coupling capacitor (just outside the AD9887A) equal to the difference between the current input voltage and ground. After the clamp is released, this difference is effectively subtracted from the raw input signal, resulting in an input signal reference to ground. For this circuit to function properly, this clamp should only be enabled when the input data is known to be black, such as during the horizontal back porch. The AD9887A has a clamp placement register to control the start of the clamp, in cycles after the falling edge of HSYNC. The clamp duration register controls the length of the clamp, in cycles.

The AD9887A also includes offset registers to offset the black level of each color individually. This control is typically not needed.

Setting Gain

The AD9887A includes gain registers to adjust the input range of the ADC. Setting this too low results in a dim display, while setting it too high results in saturation and clipping of brighter colors.

Bus Interface

The AD9887A data output is synchronous to a differential clock, DATACK. The pixel data is 8 bits per color. There are two pixel buses, A and B. There are also three key sync signals – HS, VS, and DE.

The data bus can be operated in either single pixel mode or dual pixel mode. Single pixel mode only uses port A and the clock rate is equal to the pixel rate. In dual pixel mode, port A carries even pixels and port B carries odd pixels, and the clock rate is half the pixel rate. The single pixel more would be desirable to reduce the number of signals, but the AD9887A has a max DATACK frequency of 140 MHz, so for pixel rates greater than 140 MHz, the bus has to be operated in dual pixel mode.

The VSOUT output is an unmodified signal from the selected source. In analog mode with separate syncs, it is simply a passthrough from the VSYNC input. If the analog input is using a composite sync mode, VSOUT is the recovered vertical sync. In DVI mode, VSOUT is the decoded VSYNC from the serial digital stream.

In analog modes, the HSOUT is a reconstructed version of the HSYNC input or horizontal sync from the composite sync. Registers allow adjustment of the polarity and duration of this signal. In DVI mode, HSOUT is the decoded HSYNC from the serial digital stream.

DE is only available in DVI mode. This signal qualifies the pixel data as active pixels. In analog mode, this signal is high, and the pixels must be qualified elsewhere.

DVI Input

The Digital Visual Interface (DVI) is a digital replacement for analog VGA. It is simply a digital version of the above described analog interface, with the same scan order and timing. Instead of analog voltages for the video data, the data is serially encoded digital values. This method for high-speed serial data is called transition minimized differential signaling (TMDS). TMDS is a combination of the electrical signal specification and the encoding scheme. The electrical signal specifications are similar to LVDS. The encoding scheme results in 10-bit symbols for each 8-bit byte, thus the encoded bit rate is 10x the byte rate. Each pixel is encoded as a 24-bit value, 8-bits for each color. Just like VGA, each color is transferred separately, so each color has its own differential pair. This means that the encoded bit rate is 10x the pixel rate. DVI also requires a separate clock reference signal, increasing the number of differential pairs to 4. The 10-bit encoding also includes some special symbols, allowing the sync signals to be included with the green signal (a digital SOG).

Other than the digital encoding, there is one significant difference in interfacing to DVI. The TMDS encoding also allows a data enable (DE) signal to be carried with the data. This signal is very useful to digital systems, as it easily qualifies the data. In the analog VGA scheme, the only way to know when the data is valid is to know specific timing relationships with respect to the sync signals.

I2C Initialization Table (in Hex)

All I2C communications to the AD9887A is at address 0x9a/9B. See Table 4-2 through Table 4-6.

Table 4-2: **Analog VGA60**

Register Name	Register Address	Register Value	Description
Active Interface	0x12	0x81	Force selection of analog input
PLL Div MSB	0x01	0x41	PLL divider value. VGA60 has 1056 cycles per HSYNC period. 1056 -1 = 0x41F
PLL Div LSB	0x02	0xF0	
VCO/CPMP	0x03	0x8C	VCORNGE = 00, CURRENT = 011
Phase Adjust	0x04	0x80	Default phase = T/2
Clamp Placement	0x05	0x24	36 cycles after HSYNC
Clamp Duration	0x06	0x24	36 cycles in duration
HSOUT Pulse width	0x07	0x80	128 cycles in HSYNC

Table 4-3: **Analog XGA60**

Register Name	Register Address	Register Value	Description
Active Interface	0x12	0x81	Force selection of analog input
PLL Div MSB	0x01	0x53	PLL divider value. XGA60 has 1344 cycles per HSYNC period. 1344 -1 = 0x53F
PLL Div LSB	0x02	0xF0	
VCO/CPMP	0x03	0xB4	VCORNGE = 01, CURRENT = 101
Phase Adjust	0x04	0x80	Default phase = T/2
Clamp Placement	0x05	0x40	64 cycles after HSYNC
Clamp Duration	0x06	0x40	64 cycles in duration
HSOUT Pulse width	0x07	0x88	136 cycles in HSYNC

Table 4-4: **Analog SXGA60**

Register Name	Register Address	Register Value	Description
Active Interface	0x12	0x81	Force selection of analog input
PLL Div MSB	0x01	0x69	PLL divider value. SXGA60 has 1688 cycles per HSYNC period. 1688 -1 = 0x697
PLL Div LSB	0x02	0x70	
VCO/CPMP	0x03	0xD0	VCORNGE = 10, CURRENT = 100
Phase Adjust	0x04	0x80	Default phase = T/2
Clamp Placement	0x05	0x64	100 cycles after HSYNC
Clamp Duration	0x06	0x64	100 cycles in duration
HSOUT Pulse width	0x07	0x70	112 cycles in HSYNC

Table 4-5: **Analog UXGA60**

Register Name	Register Address	Register Value	Description
Active Interface	0x12	0x81	Force selection of analog input
PLL Div MSB	0x01	0x86	PLL divider value. UXGA60 has 2160 cycles per HSYNC period. 2160 -1 = 0x86F
PLL Div LSB	0x02	0xF0	
VCO/CPMP	0x03	0xD4	VCORNGE = 10, CURRENT = 101
Phase Adjust	0x04	0x80	Default phase = T/2
Clamp Placement	0x05	0x78	120 cycles after HSYNC

Table 4-5: **Analog UXGA60 *(Continued)***

Register Name	Register Address	Register Value	Description
Clamp Duration	0x06	0x78	120 cycles in duration
HSOUT Pulse width	0x07	0xC0	192 cycles in HSYNC

DVI

DVI does not require different settings for different resolutions.

Table 4-6: **DVI**

Register Name	Register Address	Register Value	Description
Active Interface	0x12	0xC1	Force selection of DVI input
Mode Control 2	0x10	0x74	Set bus to 2 pixels/clock

References to VGA, DVI Standards

Official VGA standards are available for purchase from the Video Electronics Standards Association at: http://www.vesa.org.

DVI specifications are freely available from the Digital Display Working Group at http://www.ddwg.org.

Chapter 5

DVI/VGA Output Interface

Overview

The VIODC supports both digital and analog outputs over the DVI output connector. The interface supports standards up to UXGA (165 MHz pixel clock). By connecting a VGA to the DVI adapter to the DVI output connector, VGA output is also supported.

Figure 5-1 shows a block diagram of the DVI and VGA output subsystem. A Texas Instruments TFP410 IC is used to generate the digital DVI interface. An Analog Devices ADV7123 triple DAC is used to generate the analog VGA compatible video signals. Both devices are connected to a common 30-bit digital video bus carrying RGB data, HSYNC and VSYNC. Independent pixel clocks are routed the devices. The triple DAC supports 10-bit/channel RGB video. DVI interface supports only 8-bit/channel video, and the video data ports are connected to the 8 MSBs of the video data bus. The DVI interface also requires a data enable to distinguish active video from inactive video pixels.

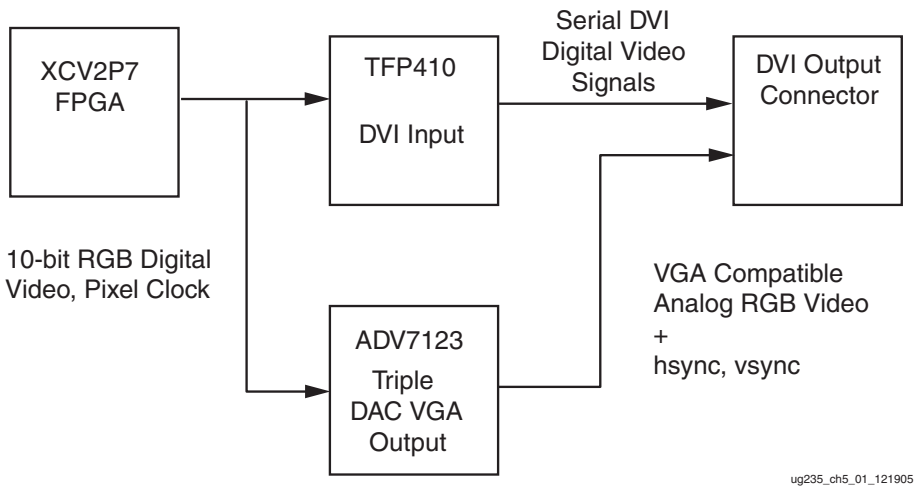

Figure 5-1: **DVI/VGA Video Output Interface Block Diagram**

The TPF410 is configured via as I2C bus interface. The DAC requires no configuration.

An I2C bus called DVI_OUT_DDC_SDA,SCL is routed from the XCV2P7 FPGA to support an I2C EPROM on the DVI output interface.

TPF410 I2C Configuration

The DVI device is initialized with a single write of 0x3f to location 0x08. The DVI I2C interface is mapped to location 0x70/0x71.

Table 5-1: **Configuration Modes for TPF410 I2C Video Encoder Chip**

Register Name	Register Address	Register Value	Description
ctl_1_mode	0x08	0x3F	[5]=vsyc_enable [4]=hsyc_enable [3]=dontcare [2]=24-bit operation [1]=posedge [0]=powerup

Chapter 6

SDI Interface

Introduction

This chapter describes how the demo is implemented. More detailed descriptions of how to implement SD-SDI and HD-SDI transmitters and receivers can be found in Xilinx application notes, XAPP683, XAPP684, XAPP579, and various other SDI-related Xilinx application notes. (The VSK is a demonstration platform only. For HD-SDI verification and compliance, Xilinx recommends using the Cook Technologies SDV board).

This chapter demonstrates the use of the SDI interfaces on the VIODC in both SD-SDI and HD-SDI modes and provides a basic demonstration of how to implement the SDI receiver and transmitter interface. The code provided can easily be modified to send the video received by the SDI receiver to different video interfaces or to the ML402 board for further processing. Likewise, the code can be modified so that the video source for the SDI transmitter comes from other video sources other than the internal video pattern generators.

The chapter also demonstrates how to use the ADV7321B video encoder device to convert digital video received by the SDI receiver into analog video. A PicoBlaze™ processor is used to control the ADV7321B video encoder through the I2C interface. The PicoBlaze processor code includes an interactive debugger allowing registers in the ADV7321B to be read or written through a ChipScope™ video input and output (VIO) console.

Reference Clocks

The source of reference clocks for the SDI transmitter and receiver is a combination of a voltage controlled crystal oscillator (VCXO) (PLL502) and a frequency synthesizer (ICS664-02).

The PLL502 VCXO is based on a 13.5 MHz crystal. The VCXO can multiply this crystal frequency by various multipliers. The control voltage for the VCXO comes from a Digital-to-Analog Converter (DAC). The DAC is controlled by the FPGA. In this demo, the DAC outputs a fixed voltage at about the 1.65 V.

The output of the VCXO is connected to the clock input of the ICS664-02 frequency synthesizer. The ICS664-02 can generate various different frequencies from the reference clock frequency supplied by the VCXO.

The RocketIO™ transceiver used in the SDI receiver needs three different clock frequencies: 74.25 MHz (to receive 1.485 Gb/s HD-SDI), 74.1758 MHz (to receive 1.4835 Gb/s HD-SDI), and 108 MHz (to receive 270 Mb/s SD-SDI). In the transmitter section, the RocketIO transceiver also needs three clock frequencies: 74.25 MHz, 74.1758 MHz, and 54 MHz.(1)

Table 6-1 shows the various frequencies produced by the PLL502 and ICS664-02 when configured for the three different SDI bit rates. In SD-SDI mode, the 54 MHz clock out of the ICS664-02 is multiplied by two by a Digital Clock Manager (DCM) to produce the 108 MHz reference clock needed by the RocketIO transceiver in the SDI receiver.

Table 6-1: **RocketIO Reference Clock Generation**

Bit Rate	VCXO Frequency	ICS664-02 Frequency	Rx REFCLK	Tx REFCLK
1.485 Gb/s	13.5 MHz	74.25 MHz	74.25 MHz	74.25 MHz
1.4835 Gb/s	13.5 MHz	74.1758 MHz	74.1758 MHz	74.1758 MHz
270 Mb/s	54 MHz	54 MHz	108 MHz	54 MHz

SDI Receiver

Figure 6-1 is a block diagram of the SDI receiver. Shaded blocks in the figure are external to the FPGA.

The serial bitstream enters the RocketIO receiver after passing through an SDI cable equalizer. The RocketIO receiver must be give a reference clock of the appropriate frequency depending on the bit rate being received. If the reference clock frequency doesn't match the bit rate of the input bitstream the receiver will not lock to the bitstream. If the demo is in Auto Rx mode, the automatic rate detection logic will sequence the RocketIO receiver through the three different bit rates supported by the demo until the receiver locks.

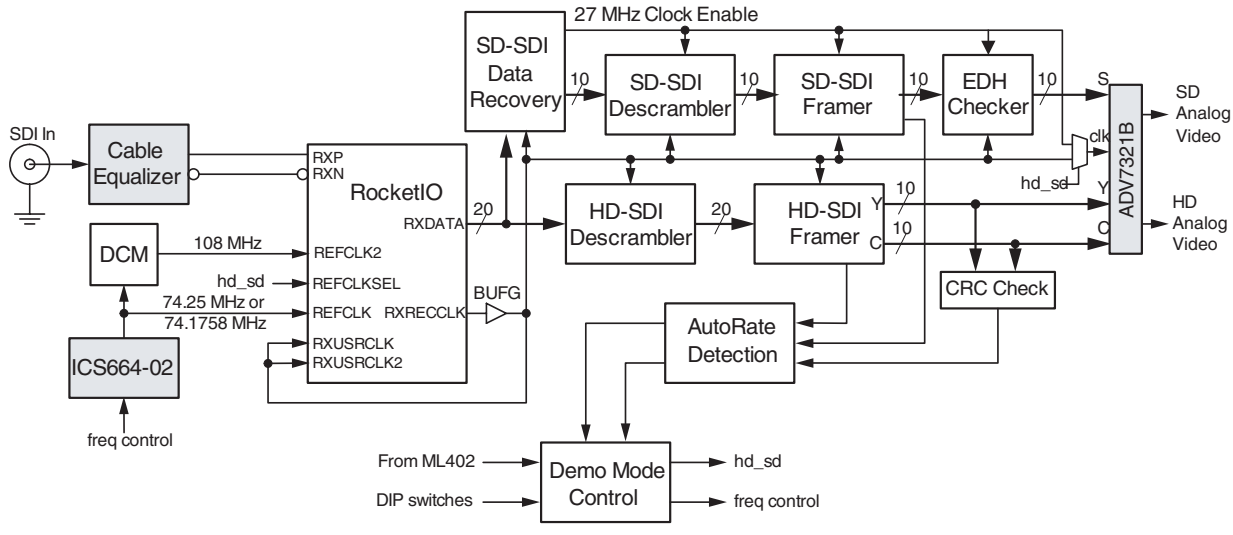

Figure 6-1: **SDI Receiver Block Diagram**

1. It is possible to use 108 MHz instead of 54 MHz for SD-SDI in the transmitter. However, because the ICS664-02 cannot directly generate 108 MHz, a DCM would be required to generate the 108 MHz clock resulting in more jitter on the output of the SDI transmitter due to higher jitter on the reference clock. The receiver section requires 108 MHz and cannot get by with 54 MHz. However, jitter on the RocketIO reference clock is not as important for the receiver.

In HD-SDI mode, there are two possible bit rates, 1.485 Gb/s and 1.4835 Gb/s. These require 74.25 MHz and 74.1758 MHz reference clock frequencies, respectively. The ICS664-02 frequency synthesizer can produce both of these frequencies. The RocketIO receiver uses this reference clock to "spin up" its clock and data recovery system. After locked to the bitstream, the RocketIO receiver produces a recovered clock on its RXRECCLK output port, running at the word rate of the recovered video data (either 74.25 MHz or 74.1578 MHz). Twenty bits of recovered data will be output on the RXDATA port every cycle of RXRECCLK.

RXRECCLK, the recovered clock produced by the RocketIO receiver, is buffered by a global clock buffer and used to clock the HD-SDI receiver logic in the FPGA. The data first passes through a descrambler to remove the NRZI encoding and scrambling. Next, the data is aligned to word boundaries by a framer. The output of the framer is two 10-bit ports, one for the luma (Y) channel information and the other the chroma (C) channel information. These two channels are checked for CRC errors and then output to the ADV7321B video encoder device to be converted to HD analog component video. Note that the HD-SDI receiver is capable of receiving some HD video formats not supported by the ADV7321B. In these cases, the ADV7321B's outputs are blanked.

The receiver works somewhat differently in SD-SDI mode. The 270 Mb/s bit rate of SD-SDI is below the minimum bit rates supported by the RocketIO receiver. For SD-SDI, the RocketIO receiver is used as an asynchronous oversampler and data recovery is not done in the RocketIO itself. The RocketIO receiver is given a 108 MHz reference clock. This is multiplied by 20 in the RocketIO receiver causing it to sample the bitstream at a 2.16 GHz rate, eight times faster than the 270 Mb/s bit rate. The RXRECCLK output clock from the RocketIO receiver is equal in frequency to the reference clock (108 MHz) and is not a true recovered clock. The RocketIO receiver outputs 20 bits of 8X oversampled data every cycle of RXRECCLK.

The oversampled data from the RocketIO receiver goes into a data recovery unit. This unit examines the oversampled data and recovers the actual data from the bitstream. It outputs the recovered data as 10-bit data words. Whenever the data recovery unit has 10-bits of recovered data ready, it asserts a data ready signal. This data ready signal is used as a clock enable to the receiver logic downstream from the data recovery unit. This clock enable signal is synchronous with the 108 MHz RXRECCLK from the RocketIO receiver. It is asserted, on average, one out of every four cycles of RXRECCLK to give an effective data rate of 27 MHz.

The recovered data passes through a SD-SDI descrambler and then a framer. The output of the framer is word aligned in the recovered video stream. The Y and C components of the video stream alternate every clock cycle on the output of the framer. The video stream is checked for errors by an EDH processor. Errors can only be detected if the incoming video has embedded EDH packets compliant with SMPTE RP 165. Finally, the video is sent to the ADV7321B video encoder and output from the VIODC as analog composite video. The data ready signal from the data recovery unit, used as a clock enable to the SDI receiver logic, is output to the ADV7321B encoder as a 27 MHz video clock for the SD video.

PicoBlaze Controller for the ADV7321B Video Encoder

A PicoBlaze soft processor in the Virtex-II Pro FPGA is used to setup and control the ADV7321B video encoder. The PicoBlaze controls the ADV7321B through the I^2C interface. The PicoBlaze processor monitors the SDI receiver status and whenever the receiver changes between HD and SD or whenever the video format being received changes, the PicoBlaze makes appropriate changes to the configuration of the ADV7321B encoder through the I^2C interface.

The code running on the PicoBlaze processor also provides an interactive debugging capability for viewing and changing the contents of the registers in the ADV7321B encoder. A ChipScope™ Pro VIO module is instantiated in the design to provide the user interface to this debugging capability.

With the demo design loaded in the Virtex-II Pro FPGA on the VIODC, the debugger can be activated by starting ChipScope Pro and loading the `adv_debugger.cpj` ChipScope project. This project opens a VIO console as shown in Figure 6-2.

To read a register in the ADV7321B video encoder, simply type the register number in hex in the Register Address field and click on the Read Strobe button. The data read from the register is displayed in the Read Data field.

To write a register, type the register number in hex in the Register Address field and the data to be written in hex in the Write Data field then click on the Write Strobe button. The debugger will always do a read of the register after it is written and display the updated value in the Read Data field.

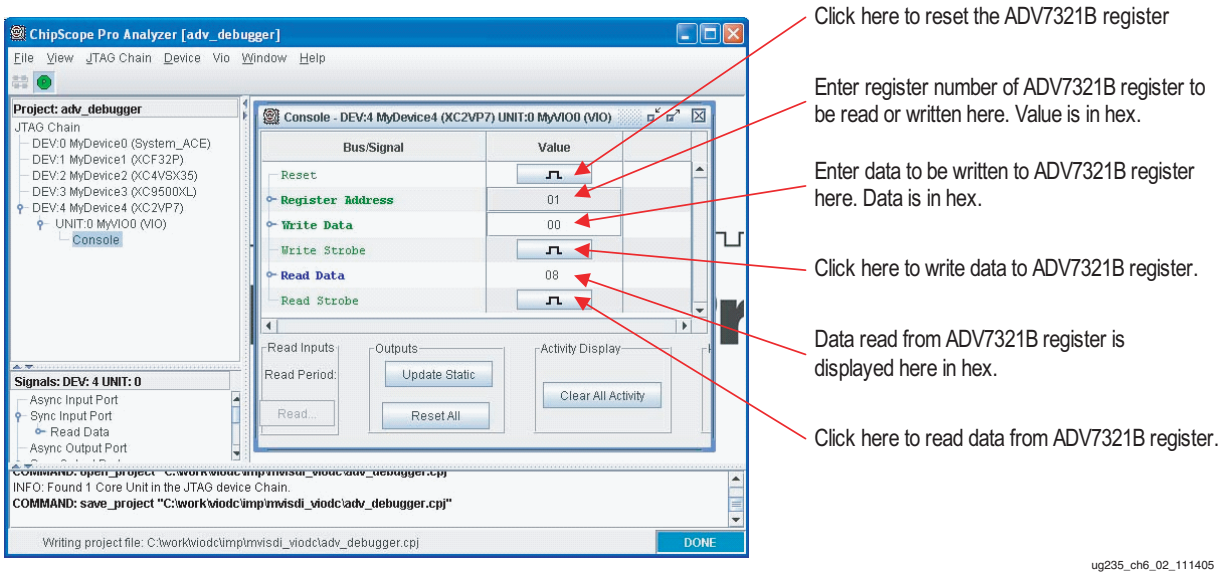

Figure 6-2: **ADV7321B Debugger**

Table 6-2 lists the settings of the ADV7321B registers when running in HD mode. Register 10 varies depending on the video format and is listed separately in Table 6-3.

Table 6-4 lists the settings used when the running in SD NTSC mode and Table 6-5 for SD PAL mode.

Table 6-2: **ADV7321B Register Settings for HD**

Register Name	Address	Value	Description
Power mode	0x00	0xFC	Enable DACs
Mode select	0x01	0x20	HD input only
HD mode 2	0x11	0x01	Pixel data valid bit set to 1
HD mode 3	0x12	0x00	Set all delays to 0
HD mode 4	0x13	0x44	Set for 4:2:2 sampling and 10-bit data input

Table 6-3: **ADV7321B HD Mode Register 1 (0x10) Settings by Video Format**

HD Video Format	Register 0x10 Value
720p60	0x2C
720p50	0x34
1080i30	0x6C
1080i25	0x74
All others	Not supported by ADV7321B

Table 6-4: **ADV7321B Register Settings for NTSC**

Register Name	Address	Value	Description
Power mode	0x00	0xFC	Enable DACs
Mode select	0x01	0x00	SD input only
SD mode 0	0x40	0x10	Select NTSC format
SD mode 1	0x42	0x40	SD pixel data valid
SD mode 3	0x44	0x00	Colorbars off
SD mode 6	0x48	0x10	10-bit input
SD timing 0	0x4A	0x08	SD timing mode 0 and blank disabled
SD Fsc 0	0x4C	0x16	
SD Fsc 1	0x4D	0x7c	
SD Fsc 2	0x4E	0xF0	
SD Fsc 3	0x4F	0x21	

Table 6-5: **ADV7321B Register Settings for PAL**

Register Name	Address	Value	Description
Power mode	0x00	0xFC	Enable DACs
Mode select	0x01	0x00	SD input only
SD mode 0	0x40	0x11	Select PAL B, D, G, H, I format
SD mode 1	0x42	0x40	SD pixel data valid
SD mode 3	0x44	0x00	Colorbars off
SD mode 6	0x48	0x10	10-bit input
SD timing 0	0x4A	0x08	SD timing mode 0 and blank disabled
SD Fsc 0	0x4C	0xCB	
SD Fsc 1	0x4D	0x8A	

Table 6-5: **ADV7321B Register Settings for PAL *(Continued)***

Register Name	Address	Value	Description
SD Fsc 2	0x4E	0x09	
SD Fsc 3	0x4F	0x2A	

SDI Transmitter

Figure 6-3 is a block diagram of the SDI transmitter. The ICS664-02 frequency synthesizer provides either 74.25 MHz or 74.1758 MHz in HD mode or 54 MHz in SD mode. This reference clock is connected directly to the REFCLK input of the RocketIO transceiver and also buffered by a global clock buffer and distributed as the transmitter clock to all portions of the transmitter section.

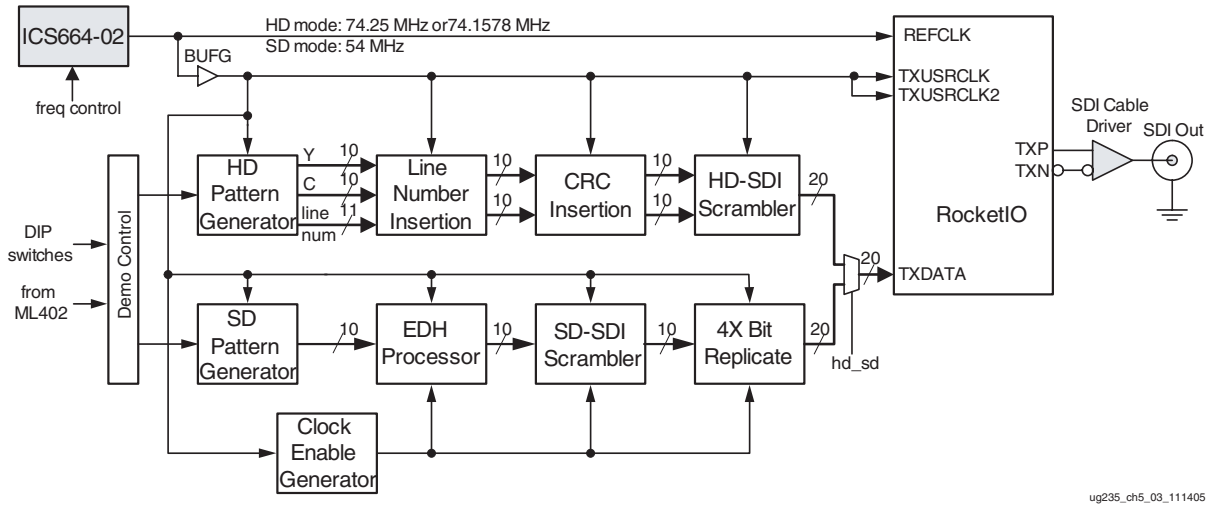

Figure 6-3: **SDI Transmitter Block Diagram**

The HD pattern generator produces four different video patterns for each of the eight HD video formats supported. The video pattern and format is selected either by the DIP switches on the VIODC or by the ML402 board. The HD pattern generator also produces an 11-bit line number value. The line number is inserted into the video stream after each End of Active Video (EAV) by the line number insertion logic. The Cyclic Redundancy Check (CRC) block generates CRCs for both the Y and C channels and inserts them into the video stream after the line number. Finally, the video is encoded for transmission by the HD-SDI scrambler and provided to the RocketIO transceiver where it is serialized and sent as an HD-SDI bitstream.

In SD mode, the clock from the ICS664-02 runs at 54 MHz. This is supplied to the RocketIO transceiver where it is multiplied by 20, so that the actual data rate of the transceiver's output is 1.08 Gb/s or 4X the 270 Mb/s SD-SDI bit rate. The SD pattern generator and the other elements of the SD-SDI transmitter data path need to run at 27 MHz, so a clock enable, asserted every other clock cycle, is generated and distributed to all elements of the SD-SDI transmitter data path.

The SD pattern generator produces either National Television System Committee (NTSC) or Phase Alternating Line (PAL) component 4:2:2 video. The EDH processor calculates the

error detection CRC values, forms them into an EDH packet, and inserts the packet into the appropriate place in the video stream. The video stream is encoded for transmission by the SD-SDI scrambler. Finally, every encoded bit from the scrambler is replicated so that four identical bits are fed to the RocketIO transmitter's data port for each encoded bit. This effectively slows the bit rate of the transmitter down to 270 Mb/s.

References

1. Xilinx application note XAPP683: *Multi-Rate HD/SD-SDI Transmitter Using Virtex-II Pro RocketIO Multi-Gigabit Transceivers*.
2. Xilinx application note XAPP684: *Multi-Rate HD/SD-SDI Receiver Using Virtex-II Pro RocketIO Multi-Gigabit Transceivers*.
3. Xilinx application note XAPP579: *Multi-Rate SDI integration Examples for the Serial Digital Video Demonstration Board*.

Chapter 7

Image Sensor Camera Interface

LVDS Camera Interface

The LVDS camera interfaces the Irvine Sensors LVDS RGB camera with a Micron MT9V022 1/3 inch CMOS image sensor. The camera provides 752 x 480 pixels at 60 Hz progressive scan. It features low noise and very high dynamic range. The interface is implemented using LVDS signaling over standard Cat-6 Ethernet cables. See Figure 7-1.

Note: The LVDS camera interface is not compatible with Ethernet.

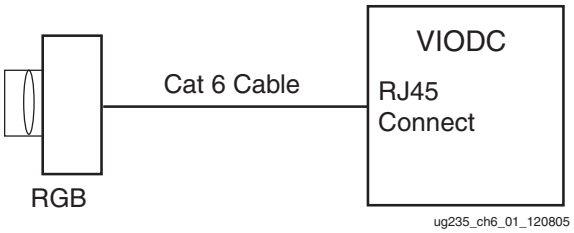

Figure 7-1: **LVDS Camera Interface**

Camera Interface Signals

The camera interface signals are shown in Table 7-1.

Table 7-1: **Camera Interface Signals**

Signal	RJ-45 Pin	Description
LVDS_DATA_P	1	Serial Camera data at 12X pixel rate
LVDS_DATA_N	2	Serial Camera data at 12X pixel rate
LVDS_CLK_P	3	Bit clock
LVDS_CLK_N	7	Bit clock
IIC_SCL	5	I2C interface clock
IIC_SDA	7	I2C interface data
+5V	5	5V power
GND	8	Ground

The LVDS camera (Figure 7-2) is available with a monochrome or RGB image sensor. The interface consists of a clock pair which runs at 12X the pixel rate. The pixel rate is 26.xx

MHz. Normally, the camera output serial data includes 8 data bits plus HSYNC and VSYNC, plus a START and STOP bit. Optionally, the output can be configured to include 10-bit data, with HSYNC and VSYNC encoded into the video data. Refer to the Micron MT9V022 data sheet for more details on configuration modes.

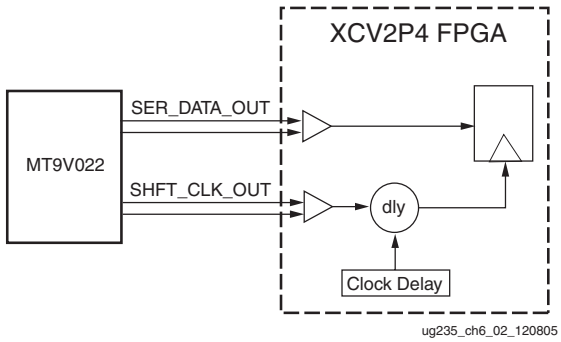

Figure 7-2: **Camera Clock**

The timing relationship between the clock and data is not specified, nor is the maximum cable rate. This requires the FPGA receiver to have the ability to adjust or skew the camera clock phase to clock in valid camera data. This is shown in Figure 7-1.

Chapter 8

Attaching the VIODC to the ML40x Development Board

The VIODC can be used in a standalone mode or mounted to a ML401, ML402, or ML403 development board. When the VIODC board is mounted on the ML40*x*, several jumpers are required to be configured properly for correct operation. The required jumper positions are detailed in Table 8-1 and shown in Figure 8-1 and Figure 8-2.

Table 8-1: **Required Jumper Positions**

Function	Jumper	Board	VIODC Standalone	ML402 Standalone	VIODC + ML40X
VIODC +5V Power	J13	VIODC	to Back Pins(1,2)	-	to Front Pins(2,3)
Bank 7 Voltage	J16	ML40x	-	Front=3.3V Pins(1,2)	Back=2.5V Pins(1,2)
JTAG TDO Source	TDO EXP	ML40x	-	Front=ML402 Pins(2,3)	Back=VIOD CPins(2,3)

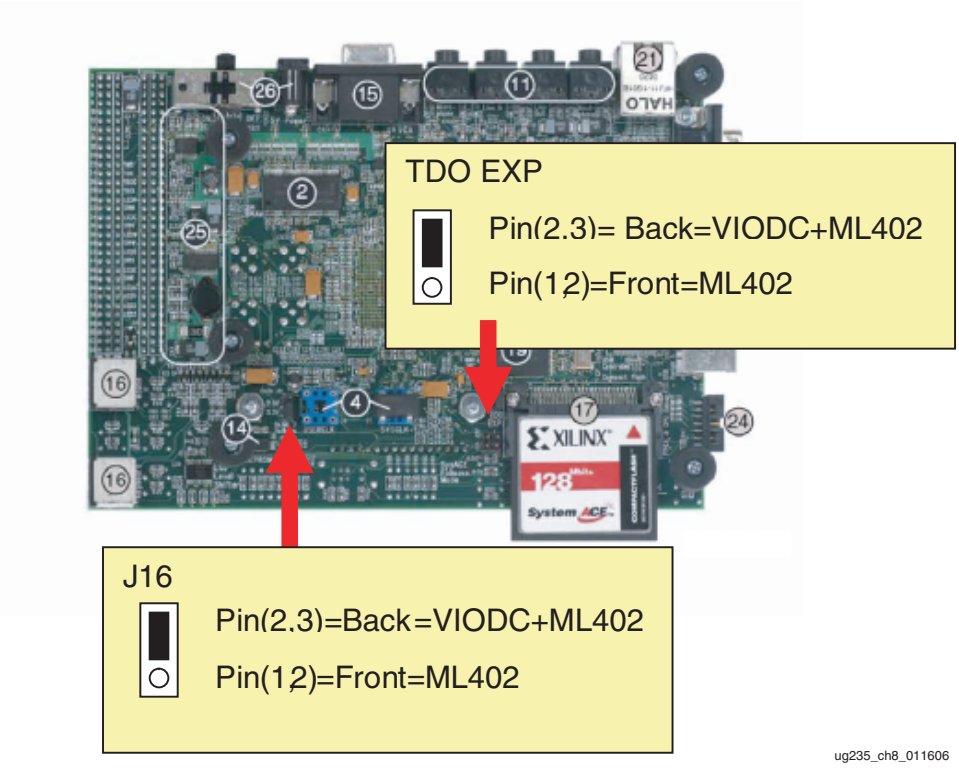

Figure 8-1: **Configuration Jumper Locations on the ML40x Bottom, Configured for VIODC Mounted to an ML402 Board**

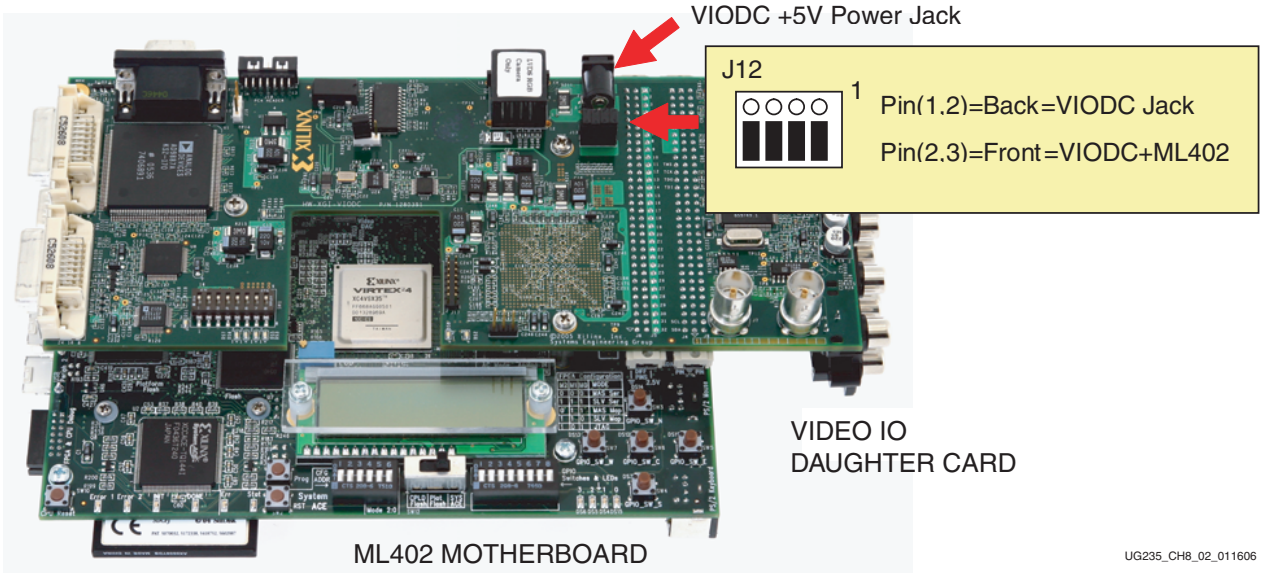

Figure 8-2: **Configuration Jumper Locations on the VIODC Top, Configured for VIODC Mounted to an ML402 Board**

XILINX®

Appendix A

Reference Information

Schematic and Data Sheet Links

Schematics	
VIODC schematic	VIODC
ML402 schematic	ML402

Table A-1: **VIODC ICs**

Manufacturer	Part Number	Function	Web Page	Data Sheet
ANALOG_DEVICES	AD9887AKS-170	DVI Receiver A&D	AD9887A	AD9887A
ANALOG_DEVICES	ADV7321AKST	Video Encoder	ADV7321A	ADV7321A
ANALOG_DEVICES	ADV7403AKSTZ-140	Video Decoder	ADV7403A	ADV7403A
ANALOG_DEVICES	ADV7123JST330	330 MHz Triple 10-Bit High Speed Video DAC	ADV7123	ADV7123
GENNUM	GS1524-CKD	Multi-Rate SDI Adaptive Cable Equalizer	GS1524	GS1524
GENNUM	GS1528-CKA	Multi-Rate SDI Dual Slew-Rate Cable Driver	GS1528	GS1528
ICS	ICS1523MLFT	Video Clock Synthesizer with I²C Programmable Delay	CS1523	CS1523
ICS	ICS664G-02LFTR	PECL Digital Video Clock Source	CS664	CS66402
MAXIM	MAX5206ACUB	16-bit DAC		MAX5206
Micron	M9T22V	CMOS Image Sensor	MT9V022	
PHASELINK	PLL502-37OCL-R	750 kHz – 800 MHz Low Phase Noise Multiplier VCXO	PLL502-37	PLL502
TI	TFP410PAP	Panel Bus DVI Transmitter 165 MHz	TFP410	TPF410

VIOBUS Pinouts

Table A-2: **VIOBUS Signals XGI Header Connections**

VIOBUS Single-Ended Mode Signal Name	VIOBUS Differential Mode Signal Name	XGI Header	Pin No.	ML402 XC4VSX35 Pin	VIODC XC2VP4 Pin
vio_up0	vio_up_lvds0_N	hdr2	2	Y18	A3
vio_up1	vio_up_lvds0_P	hdr2	4	AA18	B3
vio_up2	vio_up_lvds1_N	hdr2	6	W19	E9
vio_up3	vio_up_lvds1_P	hdr2	8	Y19	E8
vio_up4	vio_up_lvds2_N	hdr2	10	Y21	F9
vio_up5	vio_up_lvds2_P	hdr2	12	Y20	G9
vio_up6	vio_up_lvds3_N	hdr2	14	W24	C8
vio_up7	vio_up_lvds3_P	hdr2	16	W23	D8
vio_up8	vio_up_lvds4_N	hdr2	18	Y23	A8
vio_up9	vio_up_lvds4_P	hdr2	20	Y22	B8
vio_up10	vio_up_lvds5_N	hdr2	22	AA20	G14
vio_up11	vio_up_lvds5_P	hdr2	24	AA19	F14
vio_up12	vio_up_lvds6_N	hdr2	26	AA17	H15
vio_up13	vio_up_lvds6_P	hdr2	28	Y17	H14
vio_up14	vio_up_lvds7_N	hdr2	30	AC20	F15
vio_up15	vio_up_lvds7_P	hdr2	32	AB20	E15
vio_dn0	vio_dn_lvds0_N	hdr2	34	AD21	D15
vio_dn1	vio_dn_lvds0_P	hdr2	36	AE21	C15
vio_dn2	vio_dn_lvds1_N	hdr2	38	AD20	G18
vio_dn3	vio_dn_lvds1_P	hdr2	40	AE20	F18
vio_dn4	vio_dn_lvds2_N	hdr2	42	AC19	E19
vio_dn5	vio_dn_lvds2_P	hdr2	44	AD19	E18
vio_dn6	vio_dn_lvds3_N	hdr2	46	AB18	D19
vio_dn7	vio_dn_lvds3_P	hdr2	48	AC18	C19
vio_dn8	vio_dn_lvds4_N	hdr2	50	AE23	E20
vio_dn9	vio_dn_lvds4_P	hdr2	52	AF23	D20
vio_dn10	vio_dn_lvds5_N	hdr2	54	AF22	D21
vio_dn11	vio_dn_lvds5_P	hdr2	56	AF21	C21
vio_dn12	vio_dn_lvds6_N	hdr2	58	AF20	B19

Table A-2: **VIOBUS Signals XGI Header Connections *(Continued)***

VIOBUS Single-Ended Mode Signal Name	VIOBUS Differential Mode Signal Name	XGI Header	Pin No.	ML402 XC4VSX35 Pin	VIODC XC2VP4 Pin
vio_dn13	vio_dn_lvds6_P	hdr2	60	AF19	A19
vio_dn14	vio_dn_lvds7_N	hdr2	62	AE18	B24
vio_dn15	vio_dn_lvds7_P	hdr2	64	AF18	A24
vio_up16	vio_up0	hdr1	2	AA24	Y8
vio_up17	vio_up1	hdr1	4	V20	Y9
vio_up18	vio_up2	hdr1	6	AC25	Y13
vio_up9	vio_up3	hdr1	8	AC24	AA12
vio_up20	vio_up4	hdr1	10	W25	AA13
vio_up21	vio_up5	hdr1	12	AB24	AB8
vio_up22	vio_up6	hdr1	14	Y24	AB9
vio_up23	vio_up7	hdr1	16	AB23	AF8
vio_up24	vio_up8	hdr1	18	W26	AE8
vio_up25	vio_up9	hdr1	20	Y26	AB13
vio_up_clk_ena	vio_up_clk_ena	hdr1	22	Y25	AC13
vio_dn16	vio_dn0	hdr1	24	AA26	Y18
vio_dn17	vio_dn1	hdr1	26	AA23	AF19
vio_dn18	vio_dn2	hdr1	28	AC21	AE19
vio_dn19	vio_dn3	hdr1	30	AB26	AD15
vio_dn20	vio_dn4	hdr1	32	AC23	AC15
vio_dn21	vio_dn5	hdr1	34	AB25	AD19
vio_dn22	vio_dn6	hdr1	36	AD23	AB15
vio_dn23	vio_dn7	hdr1	38	AC26	AA15
vio_dn24	vio_dn8	hdr1	40	AD26	W15
vio_dn25	vio_dn9	hdr1	42	AC22	Y14
vio_dn_clk_ena	vio_dn_clk_ena	hdr1	44	V22	AC14
vio_reset	vio_reset	hdr1	46	V21	AB14
vio_sport_clk	vio_sport_clk	hdr1	48	W22	AC19
vio_sport_sync	vio_sport_sync	hdr1	50	AD25	AB19
vio_sport_dn	vio_sport_dn	hdr1	52	AB22	Y19
vio_sport_up	vio_sport_up	hdr1	54	W21	AB18
vio_i2c_scl_up	vio_i2c_scl_up	hdr1	56	W20	AA18

Table A-2: **VIOBUS Signals XGI Header Connections *(Continued)***

VIOBUS Single-Ended Mode Signal Name	VIOBUS Differential Mode Signal Name	XGI Header	Pin No.	ML402 XC4VSX35 Pin	VIODC XC2VP4 Pin
vio_i2c_sda_dn	vio_i2c_sda_dn	hdr1	58	AB21	AA14
vio_i2c_sda_up	vio_i2c_sda_up	hdr1	60	AD22	W14
vio_up_clk_lvds_N	vio_up_clk_lvds_N	hdr1	62	AE24	AD13
vio_up_clk_lvds_P	vio_up_clk_lvds_P	hdr1	64	AF24	AE13

Table A-3: **VIOBUS ML402 FPGA Connections**

VIOBUS Single-Ended Mode Signal Name	VIOBUS Differential Mode Signal Name	ML402 XC4VSX35 FPGA Pin Name	Pin	ML402 Schematic Signal Name
vio_up0	vio_up_lvds0_N	IO_L21N_7_Y18	Y18	HDR2_2
vio_up1	vio_up_lvds0_P	IO_L21P_7_AA18	AA18	HDR2_4
vio_up2	vio_up_lvds1_N	IO_L18N_7_W19	W19	HDR2_26
vio_up3	vio_up_lvds1_P	IO_L18P_7_Y19	Y19	HDR2_28
vio_up4	vio_up_lvds2_N	IO_L20N_VREF_7_Y21	Y21	HDR2_18
vio_up5	vio_up_lvds2_P	IO_L20P_7_Y20	Y20	HDR2_20
vio_up6	vio_up_lvds3_N	IO_L4N_VREF_7_W24	W24	HDR2_10
vio_up7	vio_up_lvds3_P	IO_L4P_7_W23	W23	HDR2_12
vio_up8	vio_up_lvds4_N	IO_L12N_VREF_7_Y23	Y23	HDR2_6
vio_up9	vio_up_lvds4_P	IO_L12P_7_Y22	Y22	HDR2_8
vio_up10	vio_up_lvds5_N	IO_L26N_SM2_7_AA20	AA20	HDR2_58_SYS_MON_VN2
vio_up11	vio_up_lvds5_P	IO_L26P_SM2_7_AA19	AA19	HDR2_60_SYS_MON_VP2
vio_up12	vio_up_lvds6_N	IO_L27N_SM3_7_AA17	AA17	HDR2_54_SYS_MON_VN3
vio_up13	vio_up_lvds6_P	IO_L27P_SM3_7_Y17	Y17	HDR2_56_SYS_MON_VP3
vio_up14	vio_up_lvds7_N	IO_L28N_VREF_7_AC20	AC20	HDR2_50
vio_up15	vio_up_lvds7_P	IO_L28P_7_AB20	AB20	HDR2_52
vio_dn0	vio_dn_lvds0_N	IO_L32N_SM7_7_AD21	AD21	HDR2_34_SYS_MON_VN7
vio_dn1	vio_dn_lvds0_P	IO_L32P_SM7_7_AE21	AE21	HDR2_36_SYS_MON_VP7
vio_dn2	vio_dn_lvds1_N	IO_L23N_VRP_7_AD20	AD20	HDR2_14
vio_dn3	vio_dn_lvds1_P	IO_L23P_VRN_7_AE20	AE20	HDR2_16
vio_dn4	vio_dn_lvds2_N	IO_L25N_CC_SM1_LC_7_AC19	AC19	HDR2_62_SYS_MON_VN1
vio_dn5	vio_dn_lvds2_P	IO_L25P_CC_SM1_LC_7_AD19	AD19	HDR2_64_SYS_MON_VP1
vio_dn6	vio_dn_lvds3_N	IO_L29N_SM4_7_AB18	AB18	HDR2_46_SYS_MON_VN4

Table A-3: **VIOBUS ML402 FPGA Connections *(Continued)***

VIOBUS Single-Ended Mode Signal Name	VIOBUS Differential Mode Signal Name	ML402 XC4VSX35 FPGA Pin Name	Pin	ML402 Schematic Signal Name
vio_dn7	vio_dn_lvds3_P	IO_L29P_SM4_7_AC18	AC18	HDR2_48_SYS_MON_VP4
vio_dn8	vio_dn_lvds4_N	IO_L19N_7_AE23	AE23	HDR2_22
vio_dn9	vio_dn_lvds4_P	IO_L19P_7_AF23	AF23	HDR2_24
vio_dn10	vio_dn_lvds5_N	IO_L30N_SM5_7_AF22	AF22	HDR2_42_SYS_MON_VN5
vio_dn11	vio_dn_lvds5_P	IO_L30P_SM5_7_AF21	AF21	HDR2_44_SYS_MON_VP5
vio_dn12	vio_dn_lvds6_N	IO_L17N_7_AF20	AF20	HDR2_30
vio_dn13	vio_dn_lvds6_P	IO_L17P_7_AF19	AF19	HDR2_32
vio_dn14	vio_dn_lvds7_N	IO_L31N_SM6_7_AE18	AE18	HDR2_38_SYS_MON_VN6
vio_dn15	vio_dn_lvds7_P	IO_L31P_SM6_7_AF18	AF18	HDR2_40_SYS_MON_VP6
vio_up16	vio_up0	IO_L8P_CC_LC_7_AA24	AA24	HDR1_28
vio_up17	vio_up1	IO_L5N_7_V20	V20	HDR1_42
vio_up18	vio_up2	IO_L9P_CC_LC_7_AC25	AC25	HDR1_36
vio_up9	vio_up3	IO_L16N_7_AC24	AC24	HDR1_2
vio_up20	vio_up4	IO_L2P_7_W25	W25	HDR1_52
vio_up21	vio_up5	IO_L7P_7_AB24	AB24	HDR1_32
vio_up22	vio_up6	IO_L8N_CC_LC_7_Y24	Y24	HDR1_26
vio_up23	vio_up7	IO_L14P_7_AB23	AB23	HDR1_12
vio_up24	vio_up8	IO_L2N_7_W26	W26	HDR1_50
vio_up25	vio_up9	IO_L6N_7_Y26	Y26	HDR1_38
vio_up_clk_ena	vio_up_clk_ena	IO_L6P_7_Y25	Y25	HDR1_40
vio_dn16	vio_dn0	IO_L10N_7_AA26	AA26	HDR1_22
vio_dn17	vio_dn1	IO_L14N_7_AA23	AA23	HDR1_10
vio_dn18	vio_dn2	IO_L24P_CC_LC_7_AC21	AC21	HDR1_60
vio_dn19	vio_dn3	IO_L10P_7_AB26	AB26	HDR1_24
vio_dn20	vio_dn4	IO_L16P_7_AC23	AC23	HDR1_4
vio_dn21	vio_dn5	IO_L7N_7_AB25	AB25	HDR1_30
vio_dn22	vio_dn6	IO_L15N_7_AD23	AD23	HDR1_6
vio_dn23	vio_dn7	IO_L9N_CC_LC_7_AC26	AC26	HDR1_34
vio_dn24	vio_dn8	IO_L11N_7_AD26	AD26	HDR1_18
vio_dn25	vio_dn9	IO_L13P_7_AC22	AC22	HDR1_16
vio_dn_clk_ena	vio_dn_clk_ena	IO_L1N_7_V22	V22	HDR1_54
vio_reset	vio_reset	IO_L1P_7_V21	V21	HDR1_56

Table A-3: **VIOBUS ML402 FPGA Connections *(Continued)***

VIOBUS Single-Ended Mode Signal Name	VIOBUS Differential Mode Signal Name	ML402 XC4VSX35 FPGA Pin Name	Pin	ML402 Schematic Signal Name
vio_sport_clk	vio_sport_clk	IO_L3N_7_W22	W22	HDR1_46
vio_sport_sync	vio_sport_sync	IO_L11P_7_AD25	AD25	HDR1_20
vio_sport_dn	vio_sport_dn	IO_L13N_7_AB22	AB22	HDR1_14
vio_sport_up	vio_sport_up	IO_L3P_7_W21	W21	HDR1_48
vio_i2c_scl_up	vio_i2c_scl_up	IO_L5P_7_W20	W20	HDR1_44
vio_i2c_sda_dn	vio_i2c_sda_dn	IO_L24N_CC_LC_7_AB21	AB21	HDR1_58
vio_i2c_sda_up	vio_i2c_sda_up	IO_L15P_7_AD22	AD22	HDR1_8
vio_up_clk_lvds_N	vio_up_clk_lvds_N	IO_L22N_7_AE24	AE24	HDR1_62
vio_up_clk_lvds_P	vio_up_clk_lvds_P	IO_L22P_7_AF24	AF24	HDR1_64

Table A-4: **VIOBUS VIODC FPGA Connections**

VIOBUS Single-Ended Mode Signal Name	VIOBUS Differential Mode Signal Name	VIODC XCV2P4 FPGA Pin Name	Pin	VIODC Schematic Signal Name
vio_up0	vio_up_lvds0_N	IO_L01N_1/VRP_1_A3	A3	V4_IOB_L21_N
vio_up1	vio_up_lvds0_P	IO_L01P_1/VRN_1_B3	B3	V4_IOB_L21_P
vio_up2	vio_up_lvds1_N	IO_L06N_1_E9	E9	V4_IOB_L18_N
vio_up3	vio_up_lvds1_P	IO_L06P_1_E8	E8	V4_IOB_L18_P
vio_up4	vio_up_lvds2_N	IO_L09N_1/VREF_1_F9	F9	V4_IOB_L20_N
vio_up5	vio_up_lvds2_P	IO_L09P_1_G9	G9	V4_IOB_L20_P
vio_up6	vio_up_lvds3_N	IO_L07N_1_C8	C8	V4_IOB_L4_N
vio_up7	vio_up_lvds3_P	IO_L07P_1_D8	D8	V4_IOB_L4_P
vio_up8	vio_up_lvds4_N	IO_L08N_1_A8	A8	V4_IOB_L12_N
vio_up9	vio_up_lvds4_P	IO_L08P_1_B8	B8	V4_IOB_L12_P
vio_up10	vio_up_lvds5_N	IO_L73N_0_G14	G14	V4_IOB_L26_N
vio_up11	vio_up_lvds5_P	IO_L73P_0_F14	F14	V4_IOB_L26_P
vio_up12	vio_up_lvds6_N	IO_L69N_0_H15	H15	V4_IOB_L27_N
vio_up13	vio_up_lvds6_P	IO_L69P_0/VREF_0_H14	H14	V4_IOB_L27_P
vio_up14	vio_up_lvds7_N	IO_L67N_0_F15	F15	V4_IOB_L28_N
vio_up15	vio_up_lvds7_P	IO_L67P_0_E15	E15	V4_IOB_L28_P
vio_dn0	vio_dn_lvds0_N	IO_L68N_0_D15	D15	V4_IOB_L32_N

Table A-4: **VIOBUS VIODC FPGA Connections *(Continued)***

VIOBUS Single-Ended Mode Signal Name	VIOBUS Differential Mode Signal Name	VIODC XCV2P4 FPGA Pin Name	Pin	VIODC Schematic Signal Name
vio_dn1	vio_dn_lvds0_P	IO_L68P_0_C15	C15	V4_IOB_L32_P
vio_dn2	vio_dn_lvds1_N	IO_L09N_0_G18	G18	V4_IOB_L23_N
vio_dn3	vio_dn_lvds1_P	IO_L09P_0/VREF_0_F18	F18	V4_IOB_L23_P
vio_dn4	vio_dn_lvds2_N	IO_L06N_0_E19	E19	V4_IOB_L25_N
vio_dn5	vio_dn_lvds2_P	IO_L06P_0_E18	E18	V4_IOB_L25_P
vio_dn6	vio_dn_lvds3_N	IO_L07N_0_D19	D19	V4_IOB_L29_N
vio_dn7	vio_dn_lvds3_P	IO_L07P_0_C19	C19	V4_IOB_L29_P
vio_dn8	vio_dn_lvds4_N	IO_L03N_0_E20	E20	V4_IOB_L19_N
vio_dn9	vio_dn_lvds4_P	IO_L03P_0/VREF_0_D20	D20	V4_IOB_L19_P
vio_dn10	vio_dn_lvds5_N	IO_L02N_0_D21	D21	V4_IOB_L30_N
vio_dn11	vio_dn_lvds5_P	IO_L02P_0_C21	C21	V4_IOB_L30_P
vio_dn12	vio_dn_lvds6_N	IO_L08N_0_B19	B19	V4_IOB_L17_N
vio_dn13	vio_dn_lvds6_P	IO_L08P_0_A19	A19	V4_IOB_L17_P
vio_dn14	vio_dn_lvds7_N	IO_L01N_0/VRP_0_B24	B24	V4_IOB_L31_N
vio_dn15	vio_dn_lvds7_P	IO_L01P_0/VRN_0_A24	A24	V4_IOB_L31_P
vio_up16	vio_up0	IO_L05_4/No_Pair_Y8	Y8	V4_IOB_L8_P_LC
vio_up17	vio_up1	IO_L09N_4_Y9	Y9	V4_IOB_L5_N
vio_up18	vio_up2	IO_L73N_4_Y13	Y13	V4_IOB_L9_P_LC
vio_up9	vio_up3	IO_L67N_4_AA12	AA12	V4_IOB_L16_N
vio_up20	vio_up4	IO_L73P_4_AA13	AA13	V4_IOB_L2_P
vio_up21	vio_up5	IO_L06N_4/VRP_4_AB8	AB8	V4_IOB_L7_P
vio_up22	vio_up6	IO_L06P_4/VRN_4_AB9	AB9	V4_IOB_L8_N_LC
vio_up23	vio_up7	IO_L08P_4_AF8	AD8	V4_IOB_L14_P
vio_up24	vio_up8	IO_L08N_4_AE8	AE8	V4_IOB_L2_N
vio_up25	vio_up9	IO_L74N_4/GCLK3S_AB13	AB13	V4_IOB_L6_N
vio_up_clk_ena	vio_up_clk_ena	IO_L74P_4/GCLK2P_AC13	AC13	V4_IOB_L6_P
vio_dn16	vio_dn0	IO_L09P_5_Y18	Y18	V4_IOB_L10_N
vio_dn17	vio_dn1	IO_L08N_5_AF19	AF19	V4_IOB_L14_N
vio_dn18	vio_dn2	IO_L08P_5_AE19	AE19	V4_IOB_L24_P_LC
vio_dn19	vio_dn3	IO_L68N_5_AD15	AD15	V4_IOB_L10_P
vio_dn20	vio_dn4	IO_L68P_5_AC15	AC15	V4_IOB_L16_P

Table A-4: **VIOBUS VIODC FPGA Connections *(Continued)***

VIOBUS Single-Ended Mode Signal Name	VIOBUS Differential Mode Signal Name	VIODC XCV2P4 FPGA Pin Name	Pin	VIODC Schematic Signal Name
vio_dn21	vio_dn5	IO_L07N_5/VREF_5_AD19	AD19	V4_IOB_L7_N
vio_dn22	vio_dn6	IO_L67N_5_AB15	AB15	V4_IOB_L15_N
vio_dn23	vio_dn7	IO_L67P_5_AA15	AA15	V4_IOB_L9_N_LC
vio_dn24	vio_dn8	IO_L06P_5/VRN_5_AB19	AB19	V4_IOB_L11P
vio_dn25	vio_dn9	IO_L73P_5_Y14	Y14	V4_IOB_L13_P
vio_dn_clk_ena	vio_dn_clk_ena	IO_L74N_5/GCLK5S_AC14	AC14	V4_IOB_L1_N
vio_reset	vio_reset	IO_L74P_5/GCLK4P_AB14	AB14	V4_IOB_L1_P
vio_sport_clk	vio_sport_clk	IO_L07P_5_AC19	AC19	V4_IOB_L3_N
vio_sport_sync	vio_sport_sync	IO_L69P_5_W15	W15	V4_IOB_L11_N
vio_sport_dn	vio_sport_dn	IO_L05_5/No_Pair_Y19	Y19	V4_IOB_L13_N
vio_sport_up	vio_sport_up	IO_L06N_5/VRP_5_AB18	AB18	V4_IOB_L3_P
vio_i2c_scl_up	vio_i2c_scl_up	IO_L09N_5/VREF_5_AA18	AA18	V4_IOB_L5_P
vio_i2c_sda_dn	vio_i2c_sda_dn	IO_L73N_5_AA14	AA14	V4_IOB_L24_N_LC
vio_i2c_sda_up	vio_i2c_sda_up	IO_L69N_5/VREF_5_W14	W14	V4_IOB_L15_P
vio_up_clk_lvds_N	vio_up_clk_lvds_N	IO_L75N_4/GCLK1S_AD13	AD13	V4_IOB_L22_N
vio_up_clk_lvds_P	vio_up_clk_lvds_P	IO_L75P_4/GCLK0P_AE13	AE13	V4_IOB_L22_P

Appendix B

VSK I/O Connector Location Pictures

VIODC Connectors

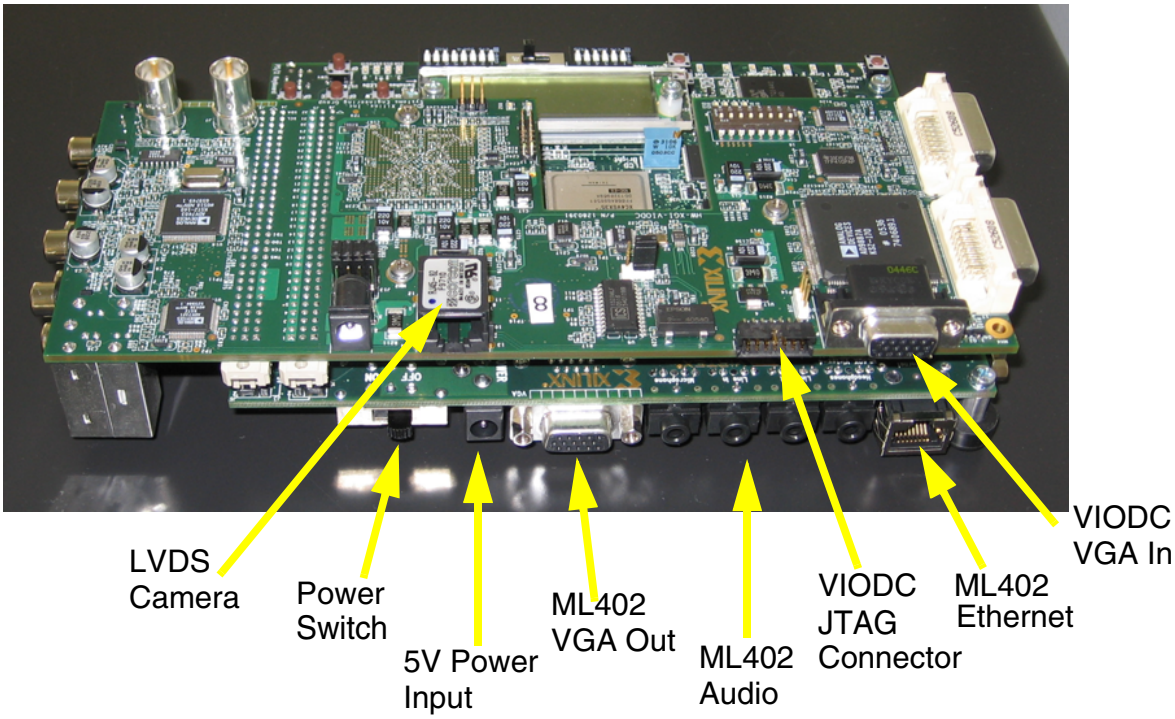

Figure B-1: **VIODC Rear View**

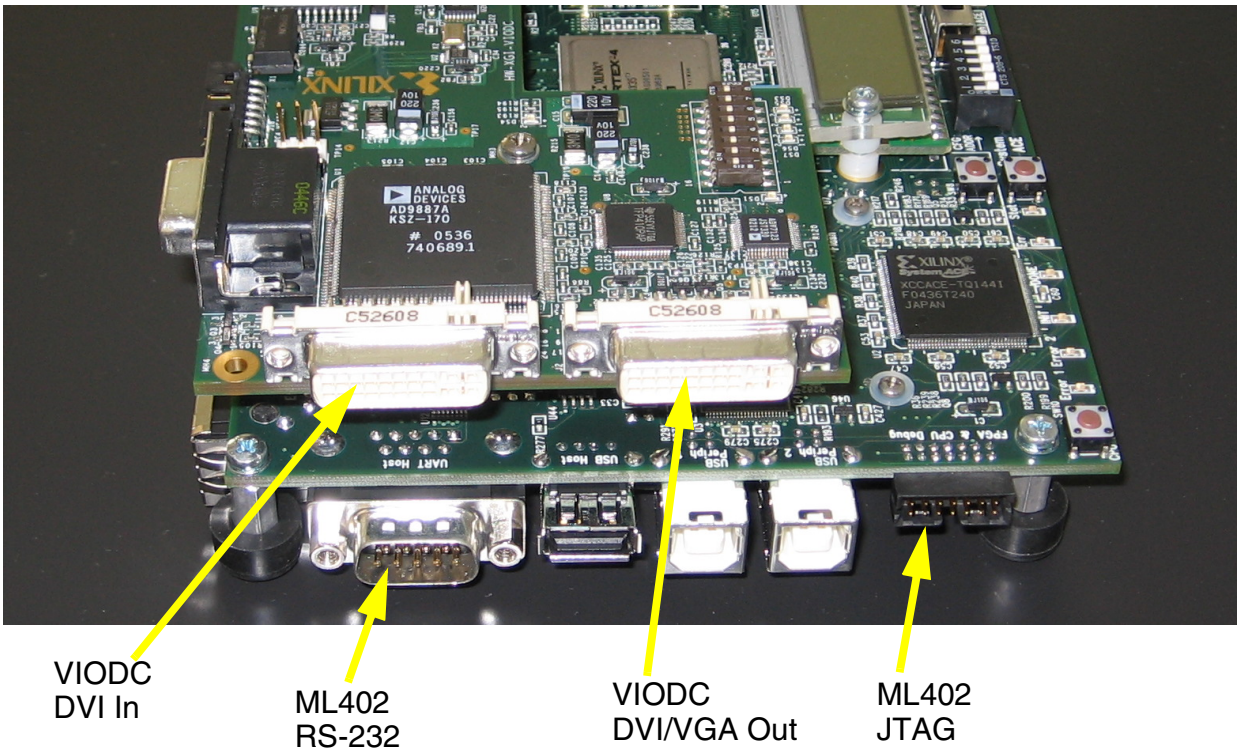

Figure B-2: **VIODC Left Side View**

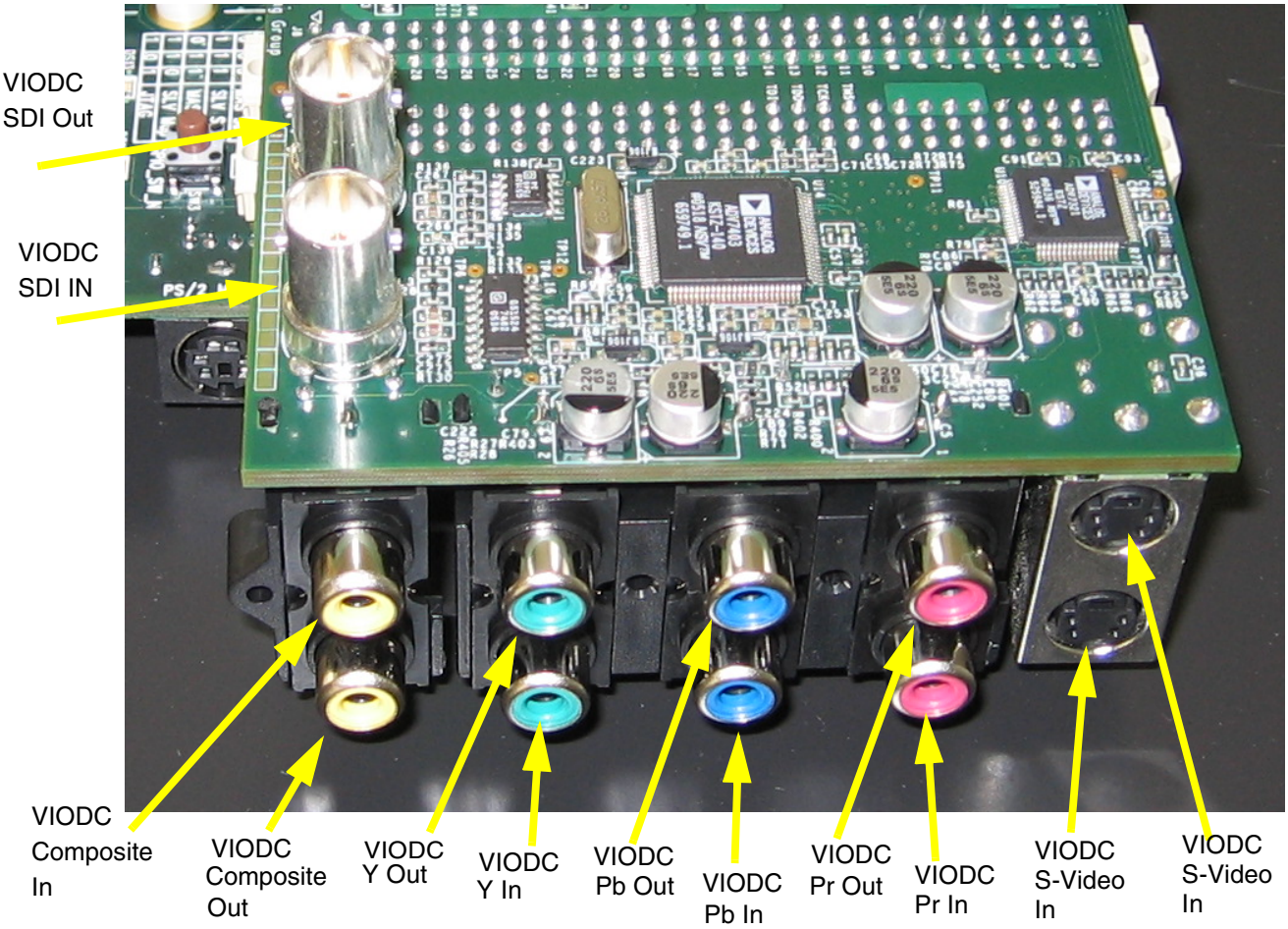

Figure B-3: **VIODC Right Side View**

LVDS Camera

Figure B-4: **LVDS Camera**

ML402 Board

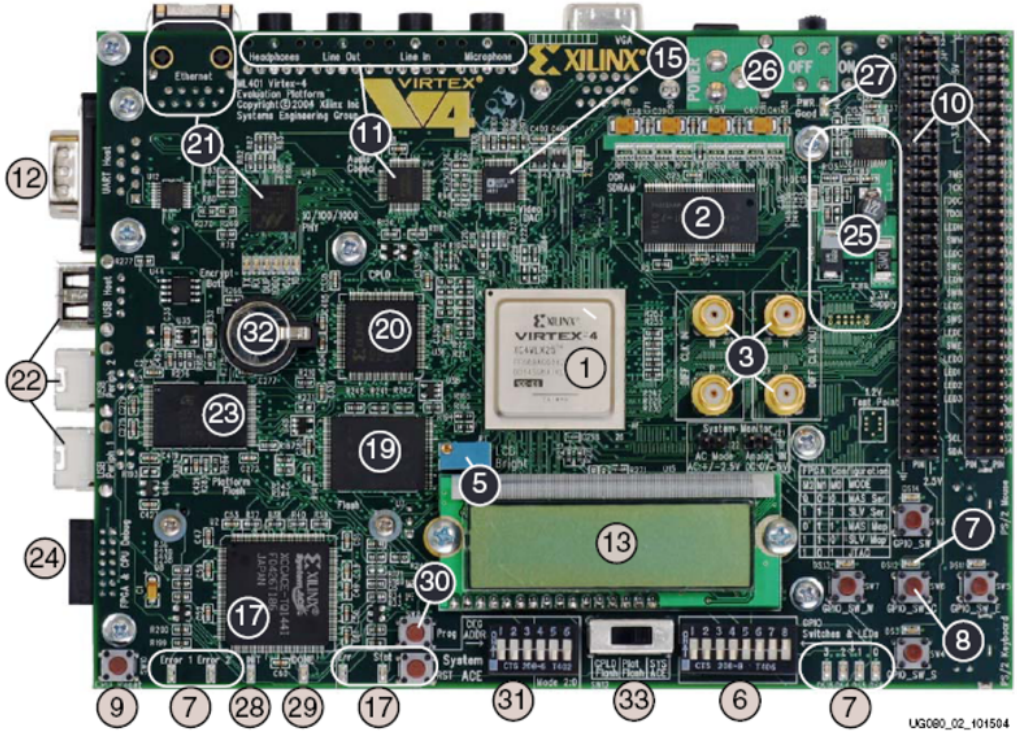

Figure B-5: **ML402 Board**

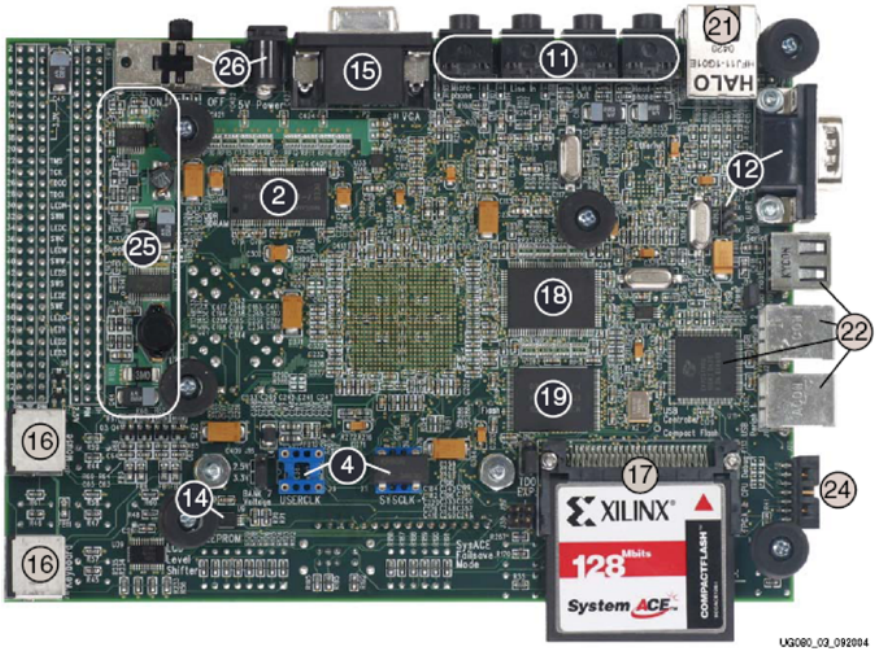

Figure B-6: **ML402 Evaluation Platform**