Lattice SDI Quad-view

June 2015 Reference Design RD1211

Introduction

This document describes a Lattice design solution for HD SDI digital video Quad-view recorder that enables SDI video capture to DDR3 memory and playback with HDMI interface from DDR3 memory. The design works with the LatticeECP3 FPGA and uses its built-in SERDER transceivers with associated PCS logic and Lattice Tri-Rate SDI PHY IP core for acquisition of SDI video data. The design has been validated on hardware using the Sparrowhawk FX development board with SDI add-on daughter card.

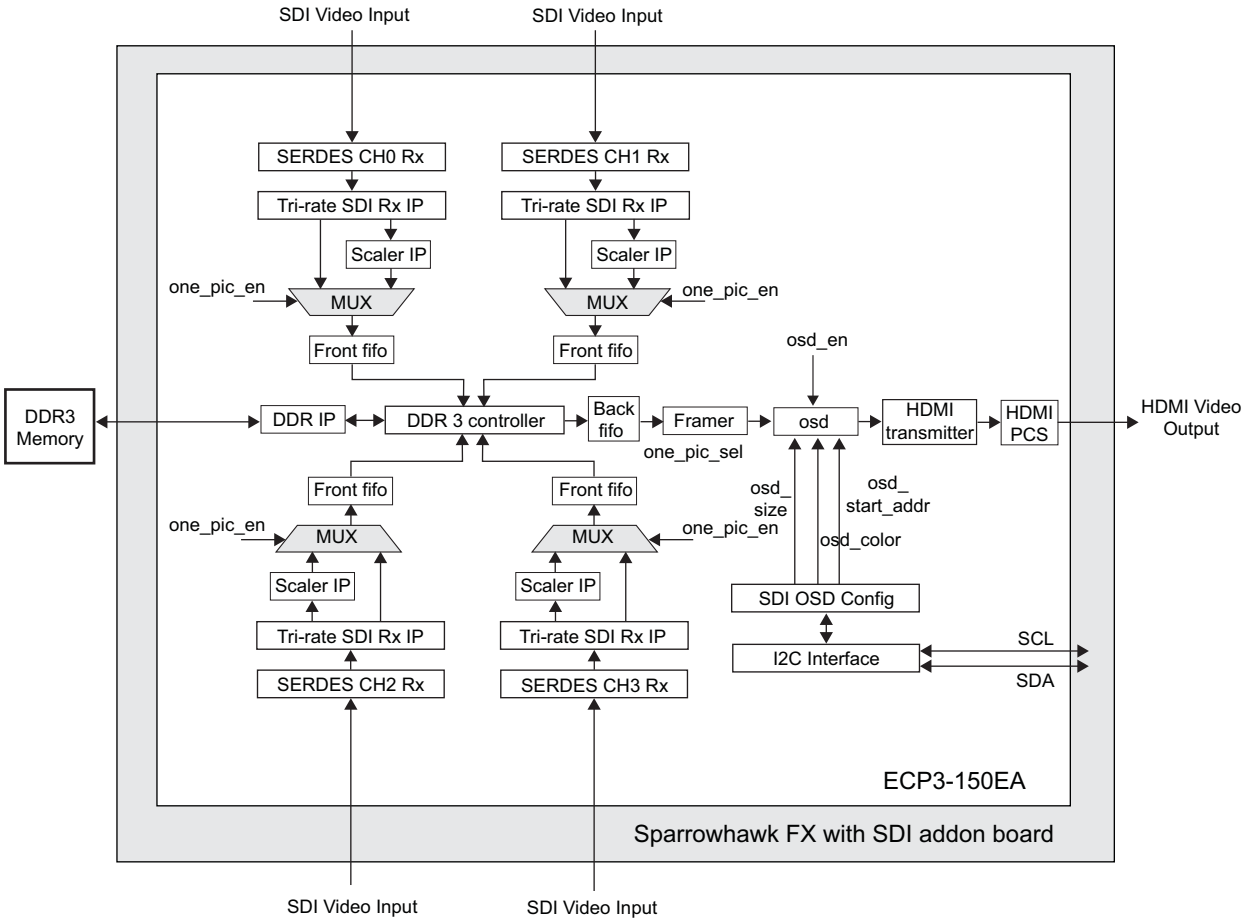
Features

- Support for the following video standards:
  - SMPTE 292M
  - SMPTE 296M
- Support for the following input video formats:
  - 720P60/M
  - 720p50/M
  - 720p30/M
  - 720p25/M
  - 720p24/M
- Support for video formats with fractional frame rate (M factor = 1.001)
- Four independent SDI video receiver channels
- Selectable video stream display with HDMI interface
- OSD function
- 4-channel SDI status monitor
- DDR3 as the external frame buffer

Functional Description

The SDI DVR solution uses LatticeECP3 FPGA to implement a low cost, low power, high performance HD-SDI-DVR solution with a high speed DDR3 interface. The functionality implemented on the LatticeECP3 FPGA includes On-Screen Display, HD Scalar and Multi-SDI Quad View, Video Multiplexer and simple OSD.

Figure 1. Block Diagram



Video Interface

The four SDI serial video input streams are received by the LatticeECP3 SERDES quads A and C and descrambled and word-aligned by the four Tri-Rate SDI PHY IP core receivers. The supported video format for each SDI input is 1280 x 720 HD only.

Table 1. SDI input SERDES mapping

BNC Connector	Quad	Serdes RX Channel
J1	PCSC	Channel #1
J2	PCSC	Channel #2
J3	PCSA	Channel #1
J4	PCSA	Channel #2

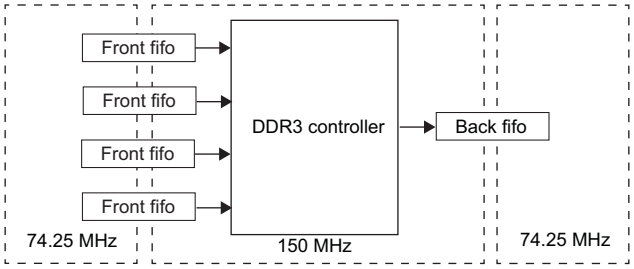
Depending on the mode of operation the received parallel video data is either sent to the scalar IP core or just passed through to four front FIFOs without scaling.

There are two modes of operation, selectable through DIP switch #2 on the Sparrowhawk FX board (marked as SW1).

- **Mode 1 (one picture mode)** – only 1 out of 4 SDI inputs is passed to the HDMI output. Inputs are selectable through the DIP switches #3 and #4.
- **Mode 2 (quad view)** – each of the 4 SDI video inputs is downscaled to 640 x 360, combined to one 720P video and then output to HDMI interface.

Data is written to the front FIFOs and read from the back FIFO synchronous to the 74.25 MHz clock. Data from the front FIFOs is written into the DDR3 memory synchronously to 150 MHz clock. DDR controller reads the data from the memory and writes it to the back FIFO synchronously to 150 MHz clock. FIFOs are used for clock domain crossing.

Figure 2. Clock Domains



Framer core controls the read from the back FIFO, generates the HDMI timing control signals, and converts the YCbCr data from the back FIFO to the RGB data. RGB data is transferred to the OSD core which then performs a simple OSD function to show channel number only. OSD can be disabled through DIP switch #0.

OSD parameters as start address, color and size are set to their default reset values, but can be changed through I2C interface.

The video pixel data, hsync/vsync and other timing control signals from the OSD core are encoded by the HDMI transmitter core into three 10-bit TMDS data. LatticeECP3 SERDES quad B serializes the data for high-speed transmission through three SERDES channels. An additional SERDES channel is used to transmit the pixel clock.

Clocking Requirements

Table 2. Clocking Requirements for Lattice SDI Quad-view Reference Design

Clocks	Description	Frequency
osc_clk_in	100 MHz clock oscillator input to the FPGA	100 MHz
k_clk	System output clock from DDR3 SDRAM memory controller PLL. Data is read from front FIFO, sent to DDR3 memory controller and written to back FIFO in this clock domain.	150 MHz
clk_148	Locally generated clock with the frequency of 148.5 MHz, used as a reference clock for the SDI receiver	148.5 MHz
clk_74	Locally generated clock with the frequency of 74.25 MHz, used as a reference clock for the HDMI transmitter	74.25 MHz
clk_27	Locally generated clock with the frequency of 27 MHz	27 MHz

Parameter Definitions

The sdi_dvr_par.v module contains parameters used in the I2C controller and register configuration parameters for the SDI Quad-view reference design. Table 10 lists the parameters included in this file.

Table 3. Definitions in the sdi_dvr_par.v Parameters File

Parameter	Value	Description
DEVICE_ID	0x30	Device I2C address
ADDR_OSD_START_ADDR	0x31	OSD_START_ADDR register address
ADDR_OSD_CTRL	0x32	OSD control register address
ADDR_SDI_RX_LOS	0x11	SDI rx loss of signal register address (read only)
ADDR_SDI_RX_LOL	0x12	SDI rx loss of lock register address (read only)
ADDR_SDI_ACT	0x13	SDI active register address (read only)
ADDR_LN0_CHAR_0	0x00	LN0_CHAR_0 register address
ADDR_LN0_CHAR_1	0x01	LN0_CHAR_1 register address
ADDR_LN0_CHAR_2	0x02	LN0_CHAR_2 register address
ADDR_LN0_CHAR_3	0x03	LN0_CHAR_3 register address
ADDR_LN0_CHAR_4	0x04	LN0_CHAR_4 register address
ADDR_LN0_CHAR_5	0x05	LN0_CHAR_5 register address
ADDR_LN0_CHAR_6	0x06	LN0_CHAR_6 register address
LN0_CHAR_NUM	0x07	LN0_CHAR_NUM register address

Table 4. Registers Description

Register	Width	Description
OSD_START_ADDR	8	osd_start_addr[7:4]: the value (4* osd_start_addr[7:4]+8) is the row number of the first text; osd_start_addr[3:0]: the value (4* osd_start_addr[3:0]+20) is the column number of the first text;
OSD_CTRL	6	[5:4] osd size: 00 - smallest, 11 - largest, default - 00 [3:0] osd color: 0x0 - r = 0xff, g = 0x00, b = 0x00; 0x1 - r = 0x00, g = 0xff, b = 0x00; 0x2 - r = 0x00, g = 0x00, b = 0xff; 0x3 - r = 0xff, g = 0xff, b = 0x00; 0x4 - r = 0xff, g = 0x00, b = 0xff; 0x5 - r = 0x00, g = 0xff, b = 0xff; 0x6 - r = 0x10, g = 0x10, b = 0x10; 0x7 - r = 0x20, g = 0x20, b = 0x20; 0x8 - r = 0x30, g = 0x30, b = 0x30; 0x9 - r = 0x40, g = 0x40, b = 0x40; 0xa - r = 0x50, g = 0x50, b = 0x50; 0xb - r = 0x60, g = 0x60, b = 0x60; 0xc - r = 0x70, g = 0x70, b = 0x70; 0xd - r = 0x80, g = 0x80, b = 0x80; 0xe - r = 0x90, g = 0x90, b = 0x90; 0xf - r = 0xa0, g = 0xa0, b = 0xa0;
SDI_RX_LOS	4	SDI rx loss of signal [3:0] - SDI in3, SDI in2, SDI in1, SDI in0
SDI_RX_LOL	0x12	SDI rx loss of lock [3:0] - SDI in3, SDI in2, SDI in1, SDI in0
SDI_ACT	0x13	SDI active register, indicates that the receiver is locked to a valid video [3:0] - SDI in3, SDI in2, SDI in1, SDI in0

Sparrowhawk FX Implementation

The SDI Quad-view reference design demonstration runs on Sparrowhawk FX board with SDI add-on card connected to one or more SDI video input sources and HDMI video output.

Prerequisities

The hardware, software, cable, general requirements and setup procedure for this demonstration are given in the following sections.

Hardware Requirements

This demonstration requires the following hardware components:

- Sparrowhawk FX board with SDI addon
- 12 V DC power supply
- PC

Software Requirements

This demonstration requires the following software components:

- Lattice Diamond Programmer tool
- ORCAstra software for user control interface - optional

Cable Requirements

This demonstration requires the following cable components:

- One to four SDI (BNC to BNC) video cables for input
- 1x DVI or DVI-to-HDMI video cable for output
- Lattice USB Programming Cable (HW-USBN-2B)

Setup Procedure

1. Check prerequisites.
2. Setup hardware.
3. Download FPGA bitstream.
4. Run the demo.

Limitations

The bitstream included with the demo design has a time-out restriction. It will only allow evaluation operation for about four hours because the Tri-Rate SDI Phy IP core evaluation version contains a built-in timer. A license is required to generate bitstream that do not include the hardware evaluation timeout limitation.

Hardware Setup

Important parts for the setup are outlined in red on the Sparrowhawk FX board layout.

- **JTAG** – used for download of FPGA designs. Connect Lattice JTAG cable here when downloading.
- **Power connector and switch** – used to power the board.
- **DVI Out #0** – mixed video output. Connect to the DVI connector of the monitor.

Figure 3. Sparrowhawk FX Board

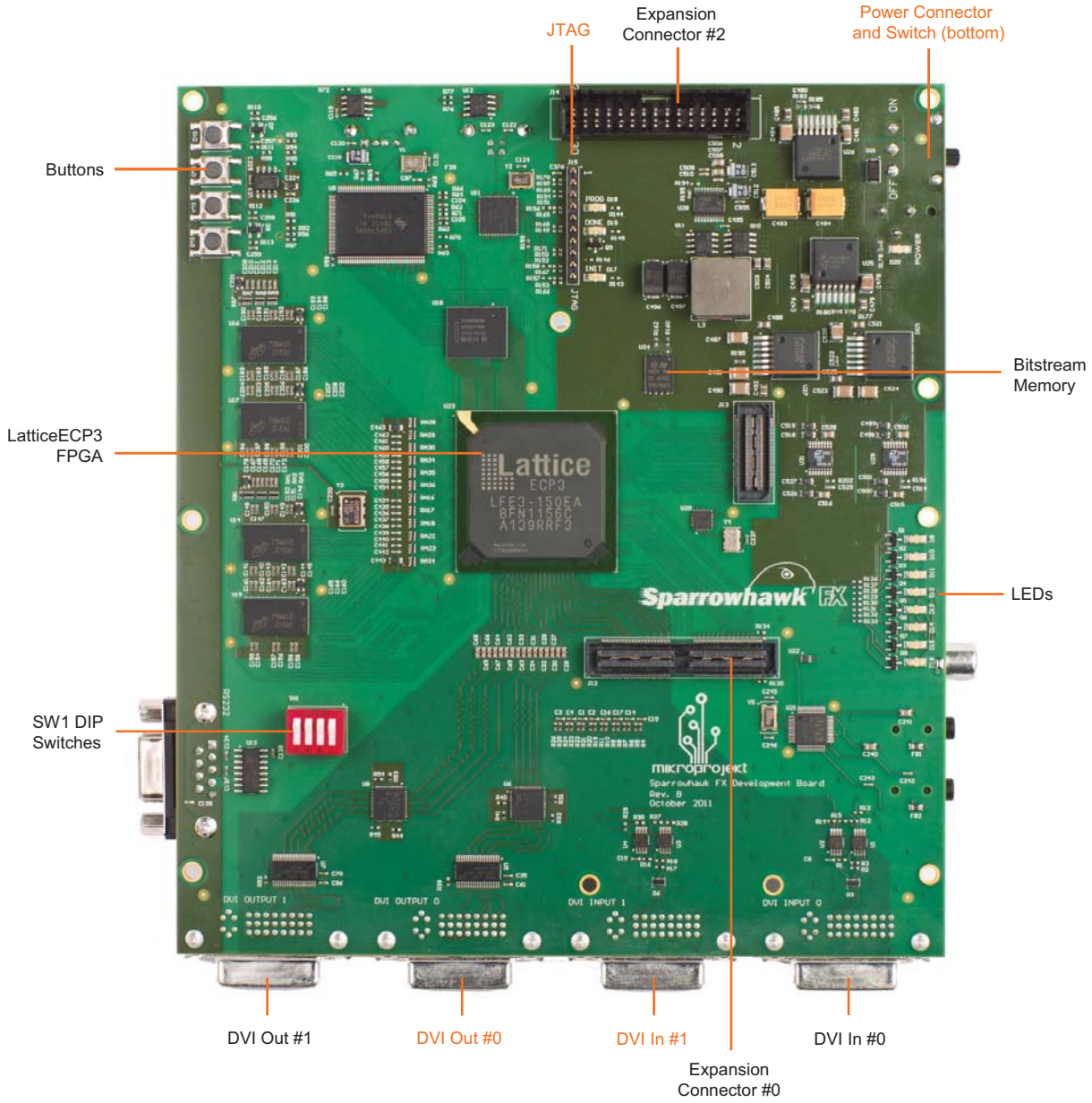


Figure 4. SDI Addon Board

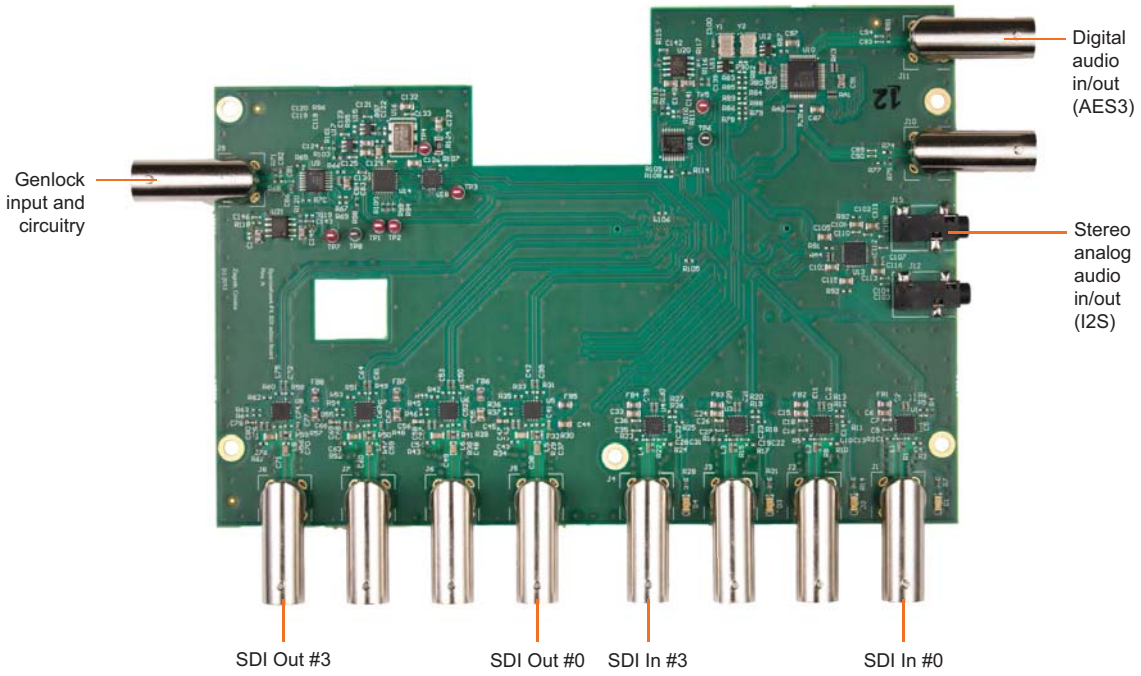
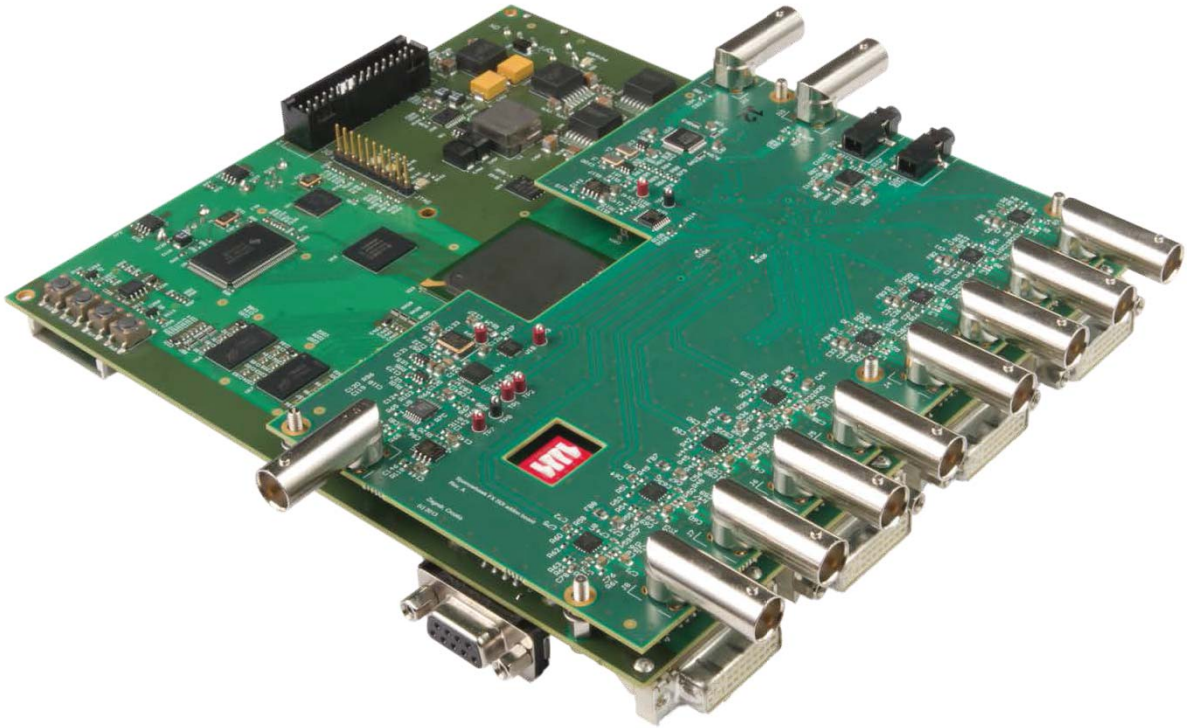


Figure 5. Sparrowhawk FX with Mounted SDI Addon

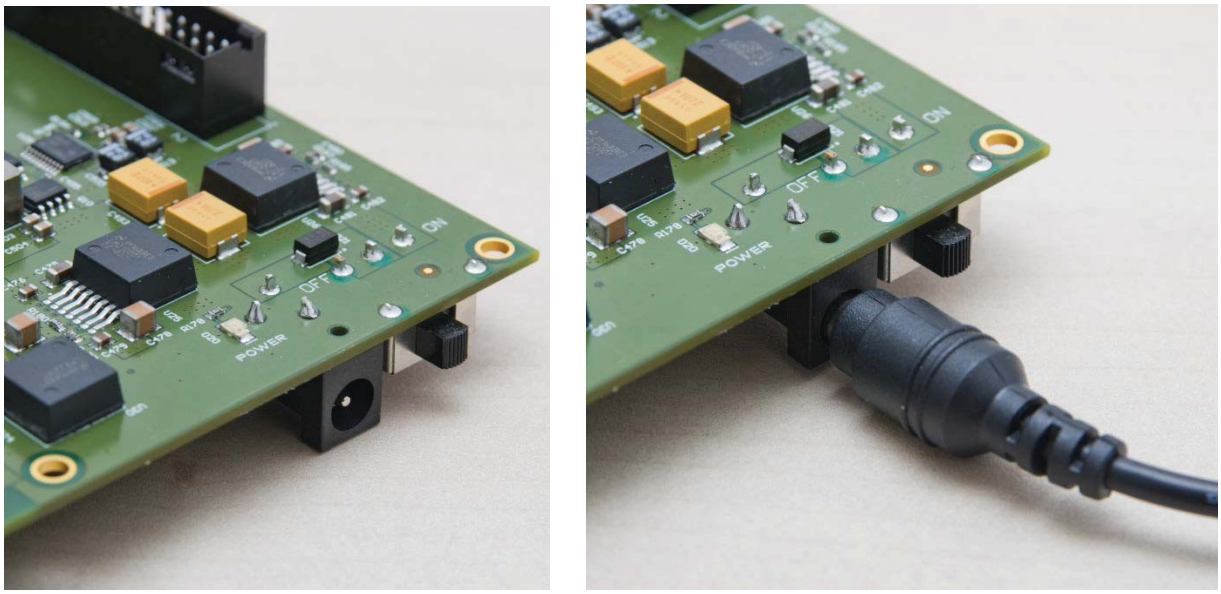


Powering Up the Board

The Sparrowhawk FX with SDI addon can be powered by the 12 V DC Wall-mount Converter provided with the board, or any type of DC supply source, providing 12 V DC and a minimum of 18 W.

The 12 V DC power supply should be connected to the connector J16 on the bottom side of the board. The Sparrowhawk FX is protected by the diode D12 from the reverse power connection. The board is turned on/off by toggling the switch SW6, with *ON* and *OFF* marked in the silkscreen on the top of the board.

Figure 6. Sparrowhawk FX 12 V DC Power Connection



Connecting Video Sources and Sinks

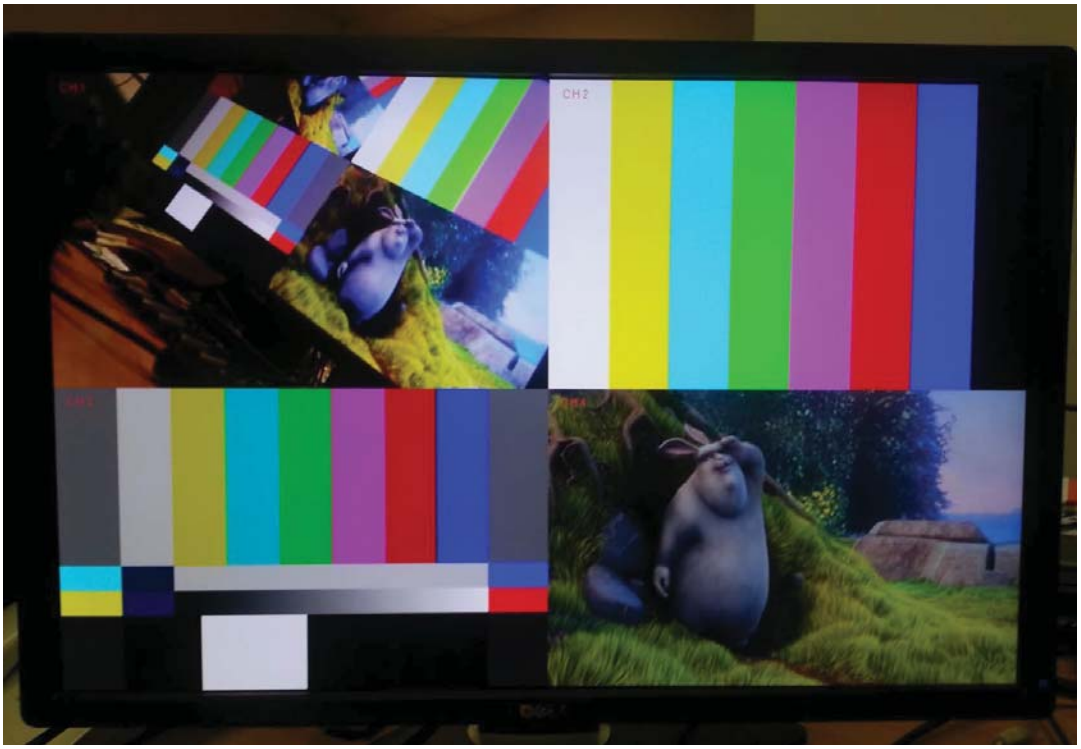
The SDI video sources and the DVI video sink are connected as shown in the Figure 7 for Mode 2 (see Table 10 for Mode switching options).

If Mode 1 is selected, SDI video input should be connected to the input #0, #1, #2 or #3 depending on which input is passed through to the HDMI output (DIP switches #3 and #4).

Note: SDI video inputs support ONLY 1280 x 720 resolution.

Figure 7. 4xSDI Input, DVI Output

Figure 8. 4xSDI Input, DVI Output on Monitor with OSD



Pinout Maps and Hardware Assignments

There are two modes of operation for SDI Quad-view reference design, one picture mode and quad view, selectable by the DIP switches listed in the following table.

Table 5. DIP Switch Control Signals Mapping

DIP Switch	Control Signal	Description
#1	osd_en	0 – Disable OSD 1 – Enable OSD
#2	one_pic_en	0 – Mode 2 is enabled (quad view) 1 – Mode 1 is enabled (one picture mode)
#3	one_pic_sel_0	Selects which input is passed to the HDMI output if one picture mode is enabled
#4	one_pic_sel_1	00 – select SDI video input 0 01 – select SDI video input 1 10 – select SDI video input 2 11 – select SDI video input 3

Push buttons SW2 and SW3 can be used to reset the SDI PCS.

Figure 9. SW Push Button for Resetting the SDI PCS

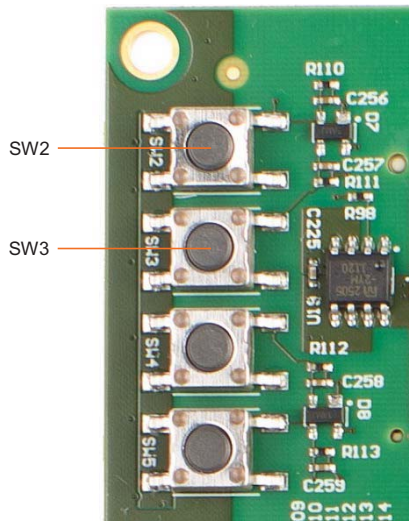


Table 6. Orcastra and Push Button Reset PCS Mapping

FPGA LatticeECP3 PCS	Orcastra	Function	Connector	Reset
PCSC	ECP3 PCS 0 channel 1	SDI RX	J1 on SDI addon	SW2
PCSC	ECP3 PCS 0 channel 2	SDI RX	J2 on SDI addon	SW2
PCSA	ECP3 PCS 2 channel 1	SDI RX	J3 on SDI addon	SW3
PCSA	ECP3 PCS 2 channel 2	SDI RX	J4 on SDI addon	SW3
PCSB	ECP3 PCS 1	HDMI TX	DVI output #0 on SHFX board	—

There are eight status LEDs on Sparrowhawk FX board that indicate the SDI channel status.

Table 7. LED Assignments

LEDs	Description	Silkscreen
led[0]	SDI_in0 loss of lock	D9
led[1]	SDI_in1 loss of lock	D10
led[2]	SDI_in2 loss of lock	D11
led[3]	SDI_in3 loss of lock	D12
led[4]	hdmi_tx_pll loss of lock	D13
led[5]	not used	D14
led[6]	not used	D15
led[7]	not used	D16

The device pinout is summarized in Table 8.

Table 8. Pinout

Ports	I/O	LatticeECP3 BGA Ball	SysIO Bank	Description
osc_clk_100	In	U6	7	100 MHz input clock
hdinp_ch0	In	AL24	PCSC	SDI high-speed PCS input, positive, channel 0
hdinn_ch0	In	AK24	PCSC	SDI high-speed PCS input, negative, channel 0
hdinp_ch1	In	AL23	PCSC	SDI high-speed PCS input, positive, channel 1
hdinn_ch1	In	AK23	PCSC	SDI high-speed PCS input, negative, channel 1
hdinp_ch2	In	AL20	PCSA	SDI high-speed PCS output, positive, channel 2
hdinb_ch2	In	AK20	PCSA	SDI high-speed PCS output, negative, channel 2
hdinp_ch3	In	AL19	PCSA	SDI high-speed PCS input, positive, channel 3
hdinn_ch3	In	AK19	PCSA	SDI high-speed PCS input, negative, channel 3
hdoutp0	Out	AP15	PCSB	HDMI high-speed PCS output, positive, channel 0
hdoutn0	Out	AN15	PCSB	HDMI high-speed PCS output, negative, channel 0
hdoutp1	Out	AP16	PCSB	HDMI high-speed PCS output, positive, channel 1
hdoutn1	Out	AN16	PCSB	HDMI high-speed PCS output, negative, channel 1
hdoutp2	Out	AP17	PCSB	HDMI high-speed PCS output, positive, channel 2
hdoutn2	Out	AN17	PCSB	HDMI high-speed PCS output, negative, channel 2
hdoutp3	Out	AP14	PCSB	HDMI high-speed PCS output, positive, channel 3
hdoutn3	Out	AN14	PCSB	HDMI high-speed PCS output, negative, channel 3
scl	In/Out	B28	1	I2C clock line
sda	In/Out	A28	1	I2C data line
em_ddr_reset_n	Out	AD10	6	Asynchronous reset signal from the memory controller to the memory device
em_ddr_data[0]	In/Out	V2	6	Memory bidirectional data bus
em_ddr_data[1]	In/Out	V4	6	Memory bidirectional data bus
em_ddr_data[2]	In/Out	U1	6	Memory bidirectional data bus
em_ddr_data[3]	In/Out	V8	6	Memory bidirectional data bus
em_ddr_data[4]	In/Out	V1	6	Memory bidirectional data bus
em_ddr_data[5]	In/Out	V9	6	Memory bidirectional data bus
em_ddr_data[6]	In/Out	U2	6	Memory bidirectional data bus
em_ddr_data[7]	In/Out	V3	6	Memory bidirectional data bus

Ports	I/O	LatticeECP3 BGA Ball	SysIO Bank	Description
em_ddr_data[8]	In/Out	Y2	6	Memory bidirectional data bus
em_ddr_data[9]	In/Out	W2	6	Memory bidirectional data bus
em_ddr_data[10]	In/Out	Y8	6	Memory bidirectional data bus
em_ddr_data[11]	In/Out	W8	6	Memory bidirectional data bus
em_ddr_data[12]	In/Out	Y1	6	Memory bidirectional data bus
em_ddr_data[13]	In/Out	W4	6	Memory bidirectional data bus
em_ddr_data[14]	In/Out	W9	6	Memory bidirectional data bus
em_ddr_data[15]	In/Out	W3	6	Memory bidirectional data bus
em_ddr_dqs[0]	In/Out	V5	6	Memory bidirectional data strobe
em_ddr_dqs[1]	In/Out	W6	6	Memory bidirectional data strobe
em_ddr_clk	Out	AE2	6	Up to 400 MHz memory clock generated by the controller
em_ddr_cke	Out	AE4	6	Memory clock enable generated by the controller
em_ddr_ras_n	Out	AL4	6	Memory row address strobe
em_ddr_cas_n	Out	AK4	6	Memory column address strobe
em_ddr_we_n	Out	AK3	6	Memory write enable
em_ddr_cs_n	Out	AN1	6	Memory chip select
em_ddr_odt	Out	AL3	6	Memory on-die termination control
em_ddr_dm[0]	Out	W7	6	DDR3 memory write data mask – to mask the byte lanes for byte-level write
em_ddr_dm[1]	Out	W1	6	DDR3 memory write data mask – to mask the byte lanes for byte-level write
em_ddr_ba[0]	Out	AP2	6	Memory bank address
em_ddr_ba[1]	Out	AC8	6	Memory bank address
em_ddr_ba[2]	Out	AN2	6	Memory bank address
em_ddr_addr[0]	Out	AN3	6	Memory address bus
em_ddr_addr[1]	Out	AC9	6	Memory address bus
em_ddr_addr[2]	Out	AM3	6	Memory address bus
em_ddr_addr[3]	Out	AP3	6	Memory address bus
em_ddr_addr[4]	Out	AD8	6	Memory address bus
em_ddr_addr[5]	Out	AM4	6	Memory address bus
em_ddr_addr[6]	Out	AD9	6	Memory address bus
em_ddr_addr[7]	Out	AM5	6	Memory address bus
em_ddr_addr[8]	Out	AC10	6	Memory address bus
em_ddr_addr[9]	Out	AL5	6	Memory address bus
em_ddr_addr[10]	Out	AJ4	6	Memory address bus
em_ddr_addr[11]	Out	AB10	6	Memory address bus
em_ddr_addr[12]	Out	AJ5	6	Memory address bus
hdmi_out_oe_n_0	Out	Y25	3	HDMI level shifter chip STHDLS101T TMDS output enable pin dedicated to the DVI output #0 (connector J3, PCSB SERDES quad)
hdmi_out_ddc_en_0	Out	Y26	3	HDMI level shifter chip STHDLS101T DDC and HPD enable pin dedicated to the DVI output #0 (connector J3, PCSB SERDES quad)
sw_dip[0]	In	A33	8	DIP switch
sw_dip[1]	In	A32	8	DIP switch

Ports	I/O	LatticeECP3 BGA Ball	SysIO Bank	Description
sw_dip[2]	In	B32	8	DIP switch
sw_dip[3]	In	C32	8	DIP switch
sw_push[0]	In	E11	0	Push button
sw_push[1]	In	E10	0	Push button
sw_push[2]	In	F10	0	Push button
sw_push[3]	In	F12	0	Push button
led[0]	Out	G23	1	LED
led[1]	Out	H23	1	LED
led[2]	Out	H22	1	LED
led[3]	Out	G21	1	LED
led[4]	Out	G26	1	LED
led[5]	Out	H26	1	LED
led[6]	Out	G25	1	LED
led[7]	Out	H25	1	LED

I2C interface is connected to the pins 5 and 6 on expansion connector #2 on Sparrowhawk FX board (J14). I2C slave interface in the design has a fixed 7-bit I2C address 0x30 (defined in the sdi_dvr_par.v parameters file).

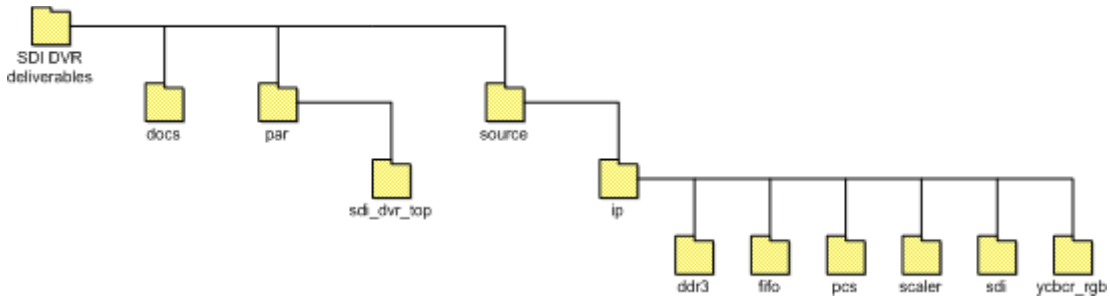
Table 9. I2C Mapping On Expansion Connector #2

LEDs	Description	BGA Ball	Expansion Connector #2 Pin
scl	I2C clock	B28	5
sda	I2C data	A28	6

Reference Design Demo

List of Files

Figure 10. Directory Structure of the SDI_DVR Deliverables



Description of directories in the deliverables folder (not all directories and the belonging files are listed here; this description is for orientation only):

- \docs: contains the following PDF documents delivered with the SDI Quad-view reference design
  - SDI Quad-view Reference Design User Guide (this document)
  - Sparrowhawk FX user's manual
  - SDI addon board user's manual
- \par: Diamond project directory, contains an .ldf Diamond design project file, .lpf constraint file, all netlist NGO files required for the FPGA design, and bitstream file containing FPGA design (in the \par\sdi_dvr_top subdirectory).
- \source: RTL code directory, contains all files pertaining to the FPGA design (top level file, clock generation, etc.). Separate subdirectory \source\ip contains Lattice IP cores with netlist and simulation files.

  The following RTL Verilog files are contained in the \source directory:
  - sdi_dvr_top.v (top level RTL file)
  - sdi_dvr_par.v
  - sdi_dvr_config_reg.v
  - scaler_intf_gen.v
  - pll_100_148.v
  - pll_100_27.v
  - osd.v
  - orcastra.v
  - i2c_if.v
  - hdmi_transmitter_bb.v
  - framer.v
  - frame_manage.v
  - front_fifo_ctrl.v
  - ddr_if.v
  - back_fifo_ctrl.v

Programming the Design

The demonstration requires deployment of FPGA design usually to the SPI FLASH memory. This chapter will discuss downloading bitstream files (.bit) to the non-volatile memory using Diamond Programmer.

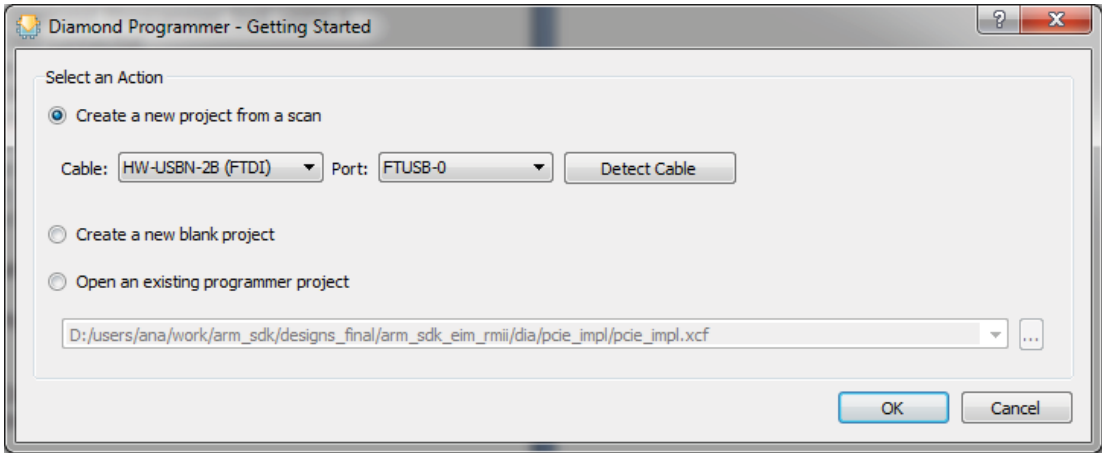
Deployment Using Diamond Programmer

SPI FLASH can be programmed using Lattice's Diamond Programmer tool, using a Lattice Programming Cable connected to the JTAG connector.

To download the bitstream file to the SPI FLASH on the board:

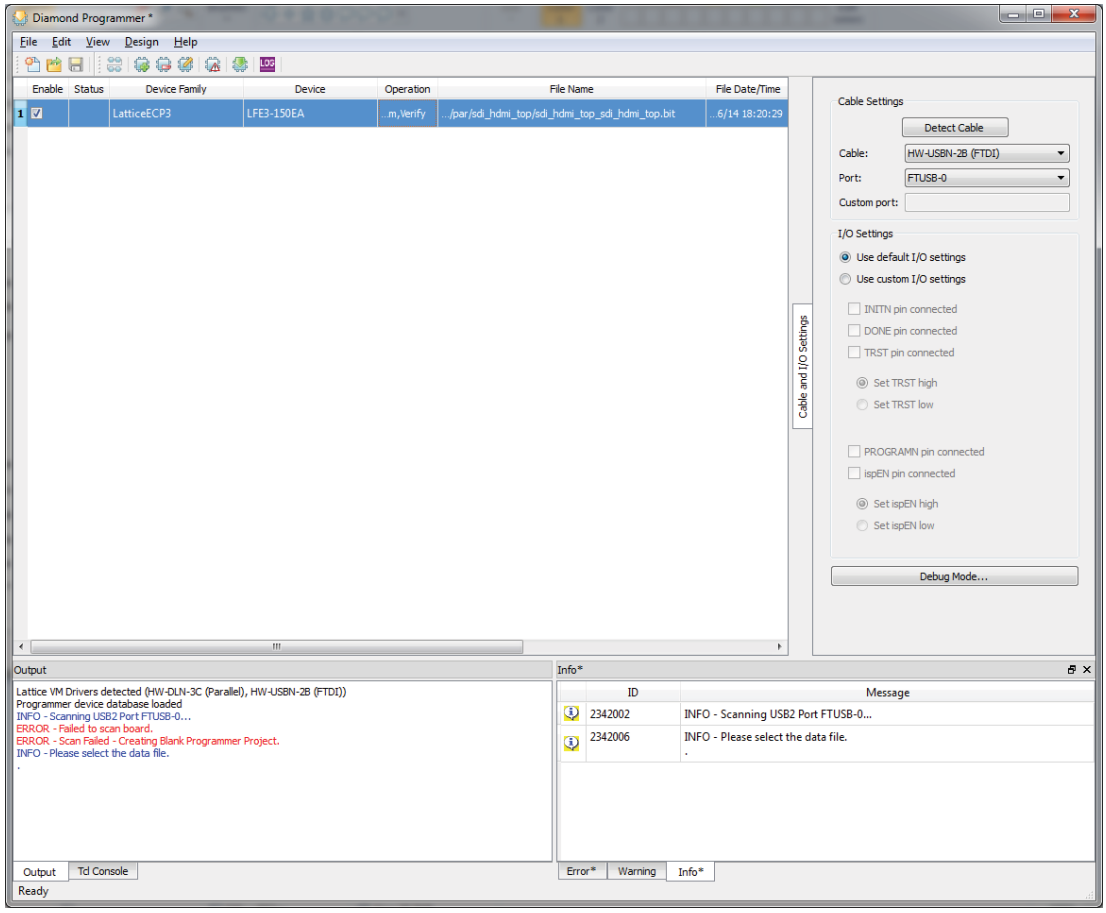
1. Remove any Lattice USB Programming cables from your system.
2. Connect the power supply to the Sparrowhawk FX board.
3. Connect a USB cable from your computer to the Sparrowhawk FX board. Give the computer a few seconds to detect the USB device.
4. In Diamond, choose **Tools > Programmer**.
5. In the Getting Started dialog box, choose **Create a new project from a scan**.
6. Click **Detect Cable**.
   a. In the Cable box, select **HW-USBN-2B (FTDI)**.
   b. In the Port box, choose **FTUSB-0**.
7. Click **OK**.

Figure 11. Getting Started Dialog Box in Diamond Programmer Tool



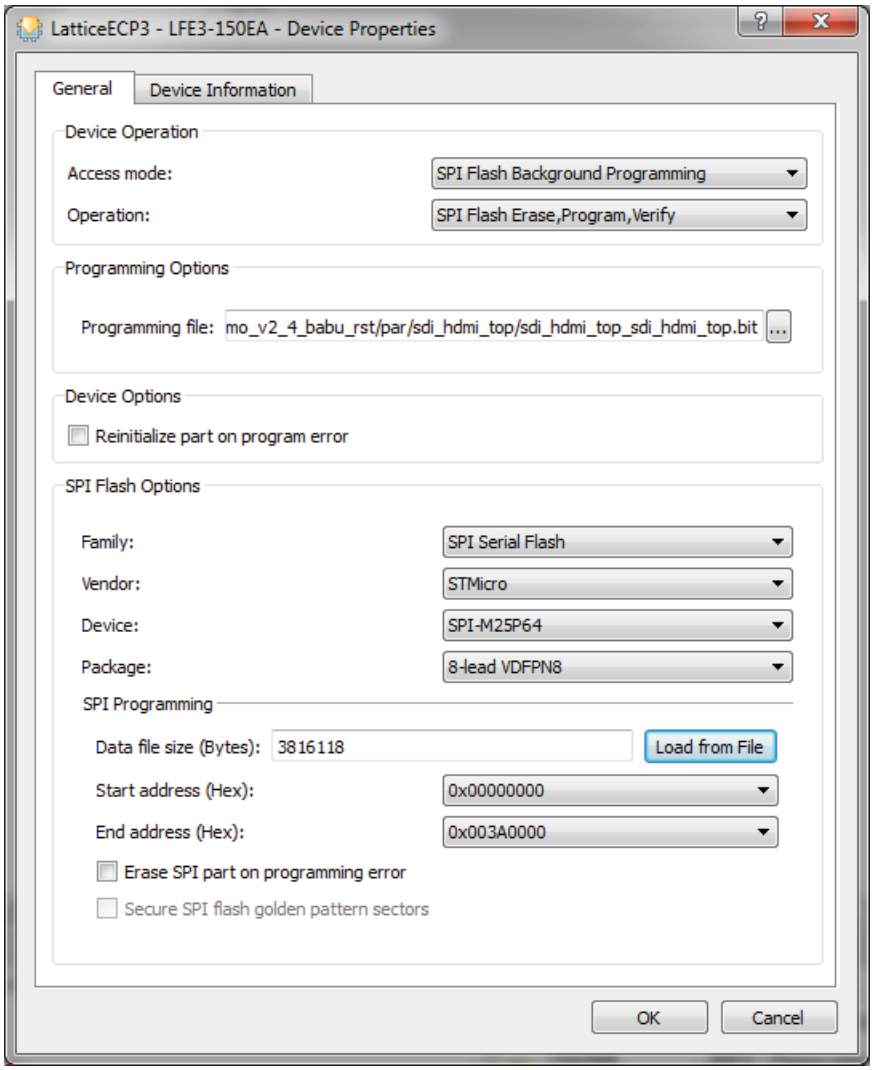
8. Programmer view is displayed in Diamond Programmer. Under the Device Family choose **LatticeECP3**.
9. Under the Device choose **LFE3-150EA**.

Figure 12. Programmer View in Diamond Programmer Tool



10. Double-click the **Operation** field.
    a. Under the Access mode, choose **SPI Flash Background Programming**.
    b. Under Operation, choose **SPI Flash Erase, Program, Verify**.
    c. In the Programming Options, choose the programming file by browsing to the directory with .bit file you wish to download.
    d. In the SPI Flash options choose **SPI Serial Flash, STMicro, SPI-M25P64, 8-lead VDFN8**.
    e. Click **Load from File**.
    f. Click **OK**.

Figure 13. Configuring SPI Flash Options in Diamond Programmer Tool



11. Click the **Program** button  on the Programmer toolbar to initiate the download.
12. If the programming process succeeded, you will see a green-shaded PASS in the Programmer Status column. Check the Programmer output console to see if the download passed.
13. Reboot the board by toggling the switch SW6.

Resource Utilization on LatticeECP3

The device resource utilization is summarized in Table 10.

Table 10. Resource Utilization for Different Modules

Modules	Slices	LUTs	Registers
sdi_dvr_config_reg	74	90	98
sdi_rx0	485	703+2*23	412
sdi_rx1	485	703+2*23	412
sdi_rx2	485	703+2*23	412
sdi_rx3	485	703+2*23	412
scaler0	860	708+2*228	1085
scaler1	860	708+2*228	1085
scaler2	860	708+2*228	1085
scaler3	860	708+2*228	1085
osd	193	189+2*80	139
i2c_itf	55	97	39
hdmi_transmitter	306	403	312
front_fifo_ctrl0	136	231	160
front_fifo_ctrl1	136	231	160
front_fifo_ctrl2	136	231	160
front_fifo_ctrl3	136	231	160
framer	341	283+2*117	460
ddr3_sdram_ctrl	1755	2145+2*122	1948
ddr_if	477	666+2*112	370
Total	10088	14746	11172

Technical Support Assistance

Submit a technical support case via www.latticesemi.com/techsupport.

Revision History

Date	Version	Change Summary
June 2015	1.0	Initial release.