

FMC-IMAGEON IP CORES for Vivado

Version 1.0

Revision History

Version	Description	Date
1.0	FMC-IMAGEON IP CORES for Vivado <ul style="list-style-type: none">Vivado 2013.3 version	Mar 06, 2014

Table of Contents

Revision History	2
Table of Contents	3
Table of Figures	4
Table of Tables	5
About this Guide	- 1 -
Additional Documentation	- 2 -
Additional Support Resources	- 3 -
Introduction	- 4 -
Archive Contents	- 6 -
FMC-IMAGEON – IP Cores	- 6 -
FMC-IMAGEON – TCL Scripts	- 6 -
FMC-IMAGEON – Software Drivers	- 6 -
FMC-IMAGEON – Application Code Examples	- 6 -
FMC-IMAGEON – IP Cores	- 7 -
fmc_imageon_hdmi_in (2.0)	- 8 -
Functionality	- 8 -
Parameters	- 9 -
Ports	- 9 -
fmc_imageon_hdmi_out (2.0)	- 10 -
Functionality	- 11 -
Parameters	- 12 -
Ports	- 12 -
fmc_imageon_vita_receiver (2.2)	- 14 -
Functionality	- 14 -
Parameters	- 17 -
Ports	- 17 -
Registers	- 18 -
FMC-IMAGEON – TCL Scripts	- 22 -
fmc_imageon_hdmii_yuv422.tcl	- 23 -
fmc_imageon_hdmii_rgb.tcl	- 23 -
fmc_imageon_hdmio_y.tcl	- 24 -
fmc_imageon_hdmio_yuv422.tcl	- 24 -
fmc_imageon_hdmio_rgb.tcl	- 25 -
fmc_imageon_vita_raw.tcl	- 25 -

fmc_imageon_vita_mono.tcl	- 26 -
fmc_imageon_vita_color.tcl	- 27 -
FMC-IMAGEON – Software Libraries & Drivers.....	- 29 -
fmc_iic_sw (2.03 a)	- 29 -
fmc_ipmi_sw (2.02 a).....	- 29 -
fmc_imageon_sw (2.01 a).....	- 31 -
fmc_imageon_vita_receiver (2.01 a).....	- 33 -
FMC-IMAGEON – Application Code Examples	- 35 -
HDMI Pass-Through	- 35 -
VITA Pass-Through	- 36 -
HDMI Display Controller	- 37 -
References	- 39 -
Known Issues and Limitations	- 41 -
HDMI Output – HDMIIO_CLK – reducing radiated emissions.....	- 41 -
Troubleshooting.....	- 42 -
ERROR: ADV7611 has not locked on incoming video, aborting !	- 42 -

Table of Figures

Figure 1 – ON Semiconductor Image Sensor with HDMI Input/Output FMC Bundle.....	- 4 -
Figure 2 – Connectivity Diagram	- 5 -
Figure 3 – Block Diagram	- 5 -
Figure 4 – fmc_imageon_hdmi_in – Block Diagram.....	- 9 -
Figure 5 – fmc_imageon_hdmi_out – Block Diagram	- 11 -
Figure 6 – fmc_imageon_vita_receiver – Block Diagram.....	- 15 -
Figure 7 – fmc_imageon_vita_core – Block Diagram.....	- 16 -
Figure 8 – fmc_imageon_vita_receiver – SYNCGEN Video Timing Diagram.....	- 21 -
Figure 9 – HDMI Input – 16 bits YUV 4:2:2 video format	- 23 -
Figure 10 – HDMI Input – 24 bits RGB video format.....	- 23 -
Figure 11 – HDMI Output – 8 bits monochrome video format.....	- 24 -
Figure 12 – HDMI Output – 16 bits YUV 4:2:2 video format	- 24 -
Figure 13 – HDMI Output – 24 bits RGB video format.....	- 25 -
Figure 14 – VITA Receiver – Raw Pixels	- 25 -

Figure 15 – VITA Receiver – Monochrome Sensor.....	- 26 -
Figure 16 – Defective Pixel Correction	- 26 -
Figure 17 – Defective Pixel Acceptance Criteria	- 27 -
Figure 18 – VITA Receiver – Color Sensor	- 28 -
Figure 19 – Color Filters in Bayer Format	- 28 -
Figure 20 –HDMI Pass-Through – 16 bits YUV 4:2:2	- 35 -
Figure 21 –HDMI Pass-Through – 24 bits RGB	- 36 -
Figure 22 –VITA Pass-Through – Color Sensor.....	- 36 -
Figure 23 –VITA Pass-Through – Monochrome Sensor	- 37 -
Figure 24 – HDMI Display Controller.....	- 37 -

Table of Tables

Table 1 – FMC-IMAGEON – IP Cores Overview	- 7 -
Table 2 – fmc_imageon_hdmi_in – Parameter List.....	- 9 -
Table 3 – fmc_imageon_hdmi_in – Port List	- 10 -
Table 4 – fmc_imageon_hdmi_in – Debug Port	- 10 -
Table 5 – fmc_imageon_hdmi_out – Parameter List.....	- 12 -
Table 6 – fmc_imageon_hdmi_out – Port List.....	- 13 -
Table 7 – fmc_imageon_hdmi_in – Debug Port.....	- 13 -
Table 8 – fmc_imageon_vita_receiver – Parameter List.....	- 17 -
Table 9 – fmc_imageon_vita_receiver – Port List	- 18 -
Table 10 – fmc_imageon_vita_receiver – Registers	- 21 -
Table 11 – FMC-IMAGEON – TCL Scripts Overview.....	- 22 -
Table 12 – IP Repository – Software Libraries Overview.....	- 29 -
Table 13 – fmc_imageon_sw – Function List	- 32 -
Table 14 – fmc_imageon_vita_receiver – Function List.....	- 34 -
Table 15 – FMC-IMAGEON – Application Code Examples	- 35 -
Table 16 – CECE913 I2C Register Settings for SSC.....	- 41 -

About this Guide

This user guide describes Vivado compliant IP Cores and related collateral that are provided for the FMC-IMAGEON FMC module.

This manual contains the following chapters:

- Chapter “**Introduction**” provides an overview and features of the FMC-IMAGEON hardware.
- The “**FMC-IMAGEON – IP Cores**” chapter provide a detailed description of the Vivado compliant IP cores for the FMC-IMAGEON.
- The “**FMC-IMAGEON – TCL Scripts**” chapter provides a detailed description of the TCL scripts for the FMC-IMAGEON, which accelerate the creation of hardware designs in Vivado IP Integrator.
- The “**FMC-IMAGEON – Software Libraries & Drivers**” chapter provides a detailed description of the Software Libraries and Drivers for the FMC-IMAGEON module.
- The “**FMC-IMAGEON – Application Code Examples**” chapter provides a description of the application code examples provided for use with the FMC-IMAGEON module.
- Appendix “**References**” provides a list of references to documentation related to the FMC module.
- Appendix “**Known Issues and Limitations**” provides a list of known issues and limitations with this reference design.
- Appendix “**Troubleshooting**” provides a list of troubleshooting suggestions for this reference design.

Additional Documentation

For more information on the VITA-2000 image sensor, please visit the following resources.



- **VITA2000: 2.3 Megapixel, 92 FPS, Global Shutter CMOS Image Sensor**
 - VITA-2000 Product Information
<http://www.onsemi.com/PowerSolutions/product.do?id=VITA2000>

For more information on the Analog Devices HDMI devices, please visit the following resources.



- **ADV7511 HDMI Transmitter**
 - ADV7511 Product Information
<http://www.analog.com/adv7511>
 - ADV7511 Technical Resources on EngineerZone
<http://ez.analog.com/docs/DOC-1740>
- **ADV7611 HDMI Receiver**
 - ADV7611 Product Information
<http://www.analog.com/adv7611>
 - ADV7611 Technical Resources on EngineerZone
<http://ez.analog.com/docs/DOC-1745>

Additional Support Resources

To access the most current collateral for the FMC-IMAGEON FMC module, please visit the product website at:

<http://www.em.avnet.com/fmc-imageon>

To access the most current collateral for the On Semi Image Sensor FMC bundle, which includes this FMC module, please visit the product website at:

<http://www.em.avnet.com/fmc-imageon-v2000c>

Once on the product website:

To access the latest documentation and designs, click on the following link:

Support Files & Downloads

To access technical support, click on the following link:

Online Technical Support

To access the technical forums, visit the following web link:

<http://community.em.avnet.com/>

To search the database of silicon and software questions and answers or to create a technical support case in WebCase, see the Xilinx website at:

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

Introduction

The ON Semiconductor Image Sensor FMC bundle provides several high-definition video interfaces for Xilinx® FMC-enabled baseboards. The FMC module has on-board HDMI input/output interfaces. The ON Semiconductor VITA-2000-color image sensor module provides a high definition camera supporting high frame rates and featuring a global shutter.

This FMC bundle is ideal for developing application for machine vision, motion monitoring, and high-end security and surveillance.



Figure 1 – ON Semiconductor Image Sensor with HDMI Input/Output FMC Bundle

As illustrated in Figure 2 and Figure 3, the FMC module connects to an FMC carrier, and provides the following interfaces:

- HDMI Input
- HDMI Output
- LCEDI Interface for VITA Image Sensor modules

The following block diagram illustrates the connectivity of the FMC module.

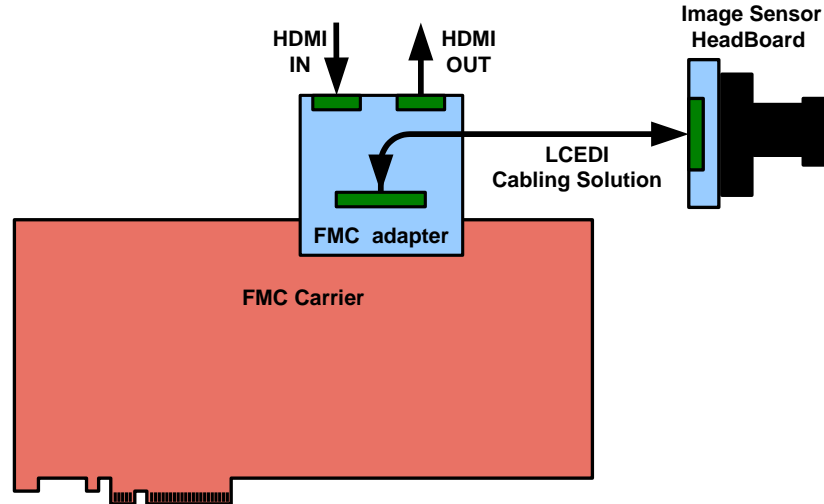


Figure 2 – Connectivity Diagram

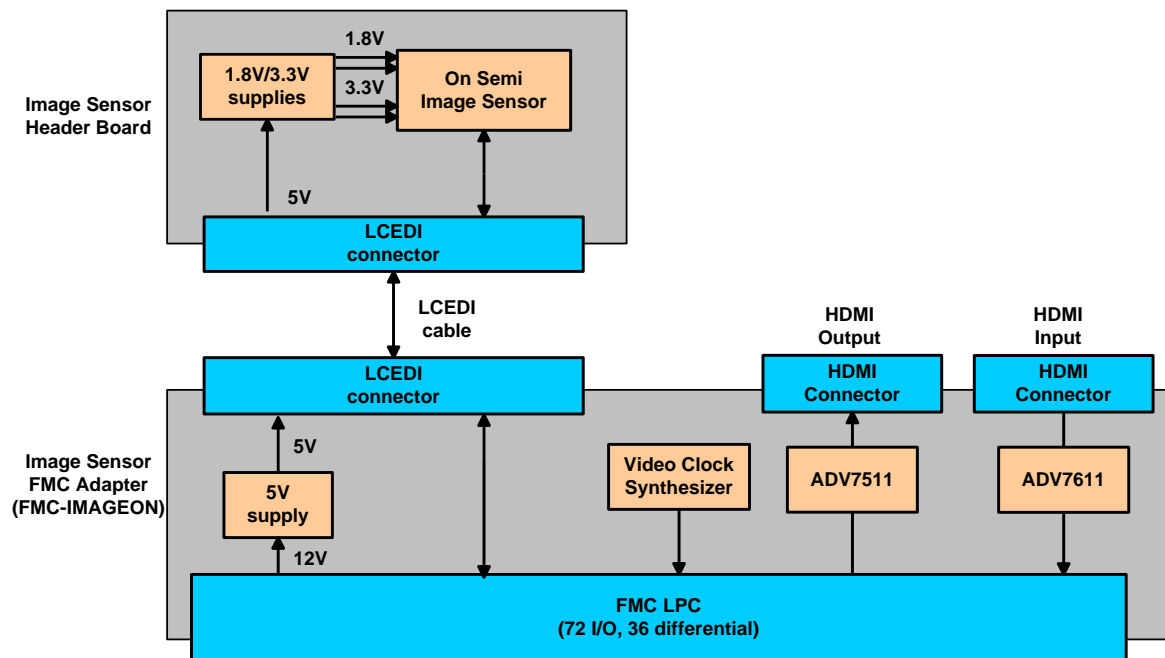


Figure 3 – Block Diagram

Archive Contents

This archive contains the following reusable components, which are described in more detail in the following sections.

FMC-IMAGEON – IP Cores

Vivado compliant IP cores are provided for the three video interfaces on the FMC module.

- FMC-IMAGEON – HDMI Input
- FMC-IMAGEON – HDMI Output
- FMC-IMAGEON – VITA Receiver

More details on these IP cores can be found in section **FMC-IMAGEON – IP Cores**.

FMC-IMAGEON – TCL Scripts

In order to assist the user to quickly create video designs based on the FMC-IMAGEON module, TCL scripts for Vivado IP Integrator have been provided. These scripts will create sub-modules which instantiate the FMC-IMAGEON IP cores, as well as the Xilinx video IP cores that they are typically connected to.

FMC-IMAGEON – Software Drivers

The following software drivers are provided for the FMC-IMAGEON module

- FMC-IMAGEON – Driver of peripherals on the I2C chain
- FMC-IMAGEON – VITA Receiver Driver

FMC-IMAGEON – Application Code Examples

The following example application code is provided for the FMC-IMAGEON module

- FMC-IMAGEON – HDMI Pass-Through
- FMC-IMAGEON – VITA Pass-Through
- FMC-IMAGEON – HDMI Display Controller

FMC-IMAGEON – IP Cores

The following table contains the complete list of IP Cores that are provided for the FMC-IMAGEON.

IP Core	Version	Description
FMC-IMAGEON HDMI Cores		
fmc_imageon_hdmii_in	2.0	The fmc_imageon_hdmii_in IP CORE provides a connection to the ADV7611 HDMI Receiver on the FMC-IMAGEON module.
fmc_imageon_hdmii_out	2.0	The fmc_imageon_hdmii_out IP CORE provides a connection to the ADV7511 HDMI Transmitter on the FMC-IMAGEON module.
FMC-IMAGEON HDMI Cores		
fmc_imageon_vita_receiver	2.2	The fmc_imageon_vita_receiver IP CORE implements the Serial LVDS receiver for the VITA image sensor.

Table 1 – FMC-IMAGEON – IP Cores Overview

The next sections describe the IP Cores in greater detail.

fmc_imageon_hdmi_in (2.0)

The fmc_imageon_hdmi_in IP CORE provides a connection to the ADV7611 HDMI Receiver chip on the FMC-IMAGEON module.

The IP Core can be found in the following location:

FMC_IMAGEON\2013_3\avnet_fmc_imageon_cores\
fmc_imageon_hdmi_input_v2

The Vivado project that can be used to re-generate or modify the IP Core can be found in the following location:

FMC_IMAGEON\2013_3\avnet_fmc_imageon_cores\ip_projects\
edit_fmc_imageon_hdmi_in.xpr

Functionality

The following figure illustrates the functionality of the fmc_imageon_hdmi_in IP Core.

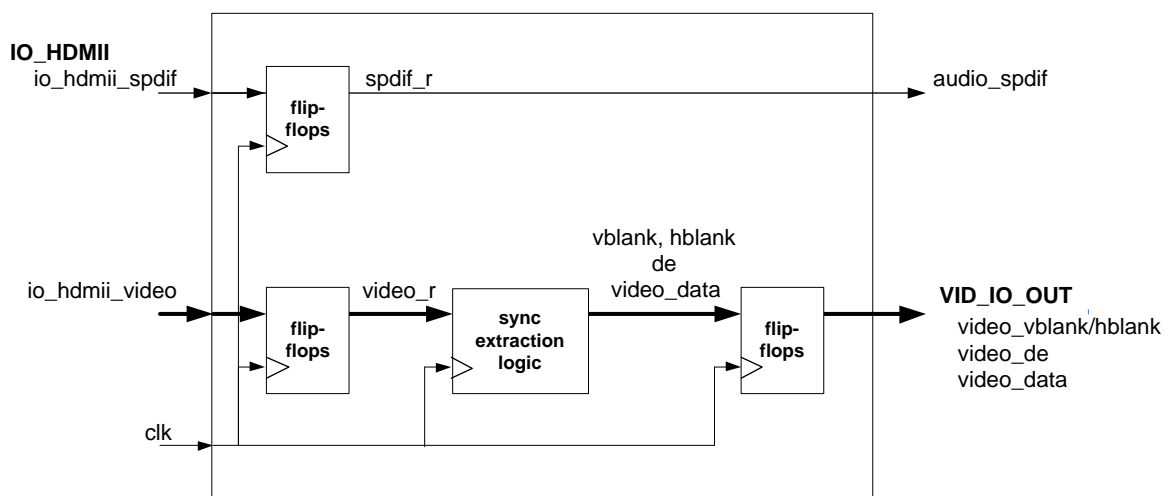


Figure 4 – fmc_imageon_hdmi_in – Block Diagram

The ADV7611 provides a 1 bit audio in SPDIF format which is simply passed through for the internal hardware design.

The ADV7611 provides a 16 bit video in YCbCr 4:2:2 format, with embedded syncs. The IP Core extracts the sync signals and creates a generic timed video (VID_IO_OUT) bus interface for the hardware design.

Parameters

Parameter	Description
C_FAMILY	FPGA Family, determines implementation. Supported values are : zynq kintex7
C_DATA_WIDTH	Video data width. Supported values are: 16 (default)
C_DEBUG_PORT	This option will make the debug port visible and accessible for debug purposes

Table 2 – fmc_imageon_hdmi_in – Parameter List

Ports

Port	Direction	Bus Interface	Description
clk	I	-	Clock input (pixel rate)
io_hdmii_spdif	I	IO_HDMII	Audio Data - SPDIF
io_hdmii_video[15:0]	I	IO_HDMII	Video input from ADV7611 - 16 bit YCbCr 4:2:2 - embedded syncs
audio_spdif	O	-	Audio input from ADV7611 - SPDIF
video_vblank	O	VID_IO_OUT	Vertical Sync strobe
video_hblank	O	VID_IO_OUT	Horizontal Sync strobe
video_de	O	VID_IO_OUT	Data Enable strobe
video_data[15:0]	O	VID_IO_OUT	Video pixel data - 16 bit YCbCr 4:2:2
debug_o[23:0]	O	-	Debug port

Table 3 – fmc_imageon_hdmi_in – Port List

A bus interface called VID_IO_OUT has been defined for the fmc_imageon_hdmi_in IP Core outputs. In IP Integrator, these outputs can either be accessed individually by their port names or as a group by their bus interface name.

A bus interface called IO_HDMI has also been defined, which groups all of the I/O pins for the HDMI input interface.

The debug port is visible when the C_DEBUG_PORT is enabled, and contains the following signals:

Index	Signal	Description
[15:0]	video_r	registered version of io_hdmi_video port
[16]	spdif_r	registered version of io_hdmi_spdif port
[17]	de	internal de signal, generated by sync extraction logic
[18]	hblank	internal hblank signal, generated by sync extraction logic
[19]	vblank	internal vblank signal, generated by sync extraction logic
[20]	sav_va	strobe indicating the detection of a SAV sync code
[21]	sav_sb	strobe indicating the detection of a SAV sync code
[22]	eav_va or eav_vb	strobe indicating the detection of a EAV sync code
[23]	sync_code	strobe indicating the detection of a sync code

Table 4 – fmc_imageon_hdmi_in – Debug Port

fmc_imageon_hdmi_out (2.0)

The fmc_imageon_hdmi_out IP CORE provides a connection to the ADV7511 HDMI Transmitter on the FMC-IMAGEON module.

The IP Core can be found in the following location:

FMC_IMAGEON\2013_3\avnet_fmc_imageon_cores\
fmc_imageon_hdmi_output_v2

The Vivado project that can be used to re-generate or modify the IP Core can be found in the following location:

FMC_IMAGEON\2013_3\avnet_fmc_imageon_cores\ip_projects\
edit_fmc_imageon_hdmi_out.xpr

Functionality

The following figure illustrates the functionality of the fmc_imageon_hdmi_out IP CORE.

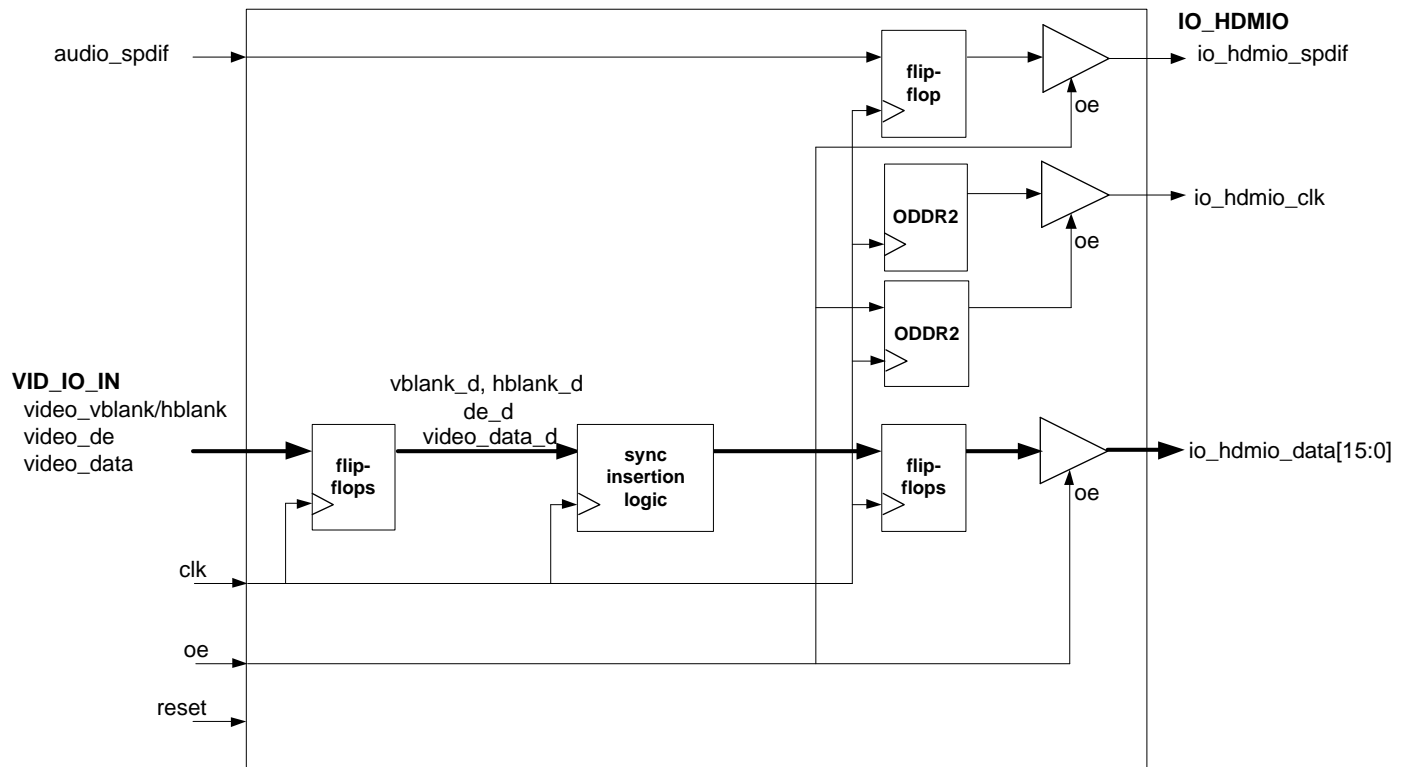


Figure 5 – fmc_imageon_hdmi_out – Block Diagram

A 1 bit audio signal in SPDIF format is passed through to the ADV7511 device.

The IP CORE receives a generic timed video (VID_IO_IN) Interface in 16 bit YCbCr 4:2:2 format, embeds the vblank, hblank, and de synchronization signals, before driving to the ADV7511 HDMI transmitter.

The clock sent to the ADV7511 output interface is generated with ODDR2 registers in the IOB fabric.

All outputs are enabled with the oe port. This allows the outputs to the FMC connector to be tri-stated until the FMC module has been correctly identified.

Parameters

Parameter	Description
C_FAMILY	FPGA Family, determines implementation. Supported values are : zynq kintex7
C_DATA_WIDTH	Video data width. Supported values are: 16 (default)
C_DEBUG_PORT	This option will make the debug port visible and accessible for debug purposes

Table 5 – fmc_imageon_hdmi_out – Parameter List

Ports

Port	Direction	Bus Interface	Description
clk	I	-	Clock input (pixel rate)
reset	I	-	Reset
oe	I	-	Output Enable (can be used to disable all outputs to FMC connector)
embed_syncs	I	-	Enable embedded sync generation
audio_spdif	I	-	Audio input - SPDIF
video_vblank	I	VID_IO_IN	Vertical Sync strobe
video_hblank	I	VID_IO_IN	Horizontal Sync strobe
video_de	I	VID_IO_IN	Data Enable strobe
video_data[15:0]	I	VID_IO_IN	Video pixel data - 16 bit YCbCr 4:2:2
io_hdmio_spdif	O	IO_HDMIO	Audio output to ADV7511
io_hdmio_clk	O	IO_HDMIO	Clock output to ADV7511
io_hdmio_video[15:0]	O	IO_HDMIO	Video output to ADV7511 - 16 bit YCbCr 4:2:2 - embedded syncs

debug_o[39:0]	O	-	Debug port
---------------	---	---	------------

Table 6 – fmc_imageon_hdmi_out – Port List

A bus interface called VID_IO_OUT has been defined for the fmc_hdmi_hdmi_out IP CORE inputs. In IP Integrator, these outputs can either be accessed individually by their port names or as a group by their bus interface name.

A bus interface called IO_HDMIO has also been defined, which groups all of the I/O pins for the HDMI output interface.

The debug port is visible when the C_DEBUG_PORT is enabled, and contains the following signals:

Index	Signal	Description
[15:0]	video_r	early version of video_data port
[16]	spdif_r	early version of audio_spdif port
[17]	'0'	unused
[18]	'0'	unused
[19]	embed_syncs	the embed_syncs port
[36:20]	video_data_r	registered version of video_data port
[36]	de_d	registered version of video_de port
[37]	hblank_d	registered version of video_hblank port
[38]	vblank_d	registered version of video_vblank port
[39]	'0'	unused

Table 7 – fmc_imageon_hdmo_in – Debug Port

fmc_imageon_vita_receiver (2.2)

The IP Core can be found in the following location:

FMC_IMAGEON\2013_3\avnet_fmc_imageon_cores\
fmc_imageon_vita_receiver_v2

The Vivado project that can be used to re-generate or modify the IP Core can be found in the following location:

FMC_IMAGEON\2013_3\avnet_fmc_imageon_cores\ip_projects\
edit_fmc_imageon_vita_receiver.xpr

Functionality

The following figure illustrates the functionality of the fmc_imageon_vita_receiver IP CORE. Memory mapped registers are used to configure the VITA receiver. These registers are implemented directly in the AXI4-Lite Slave Logic in the top level.

.

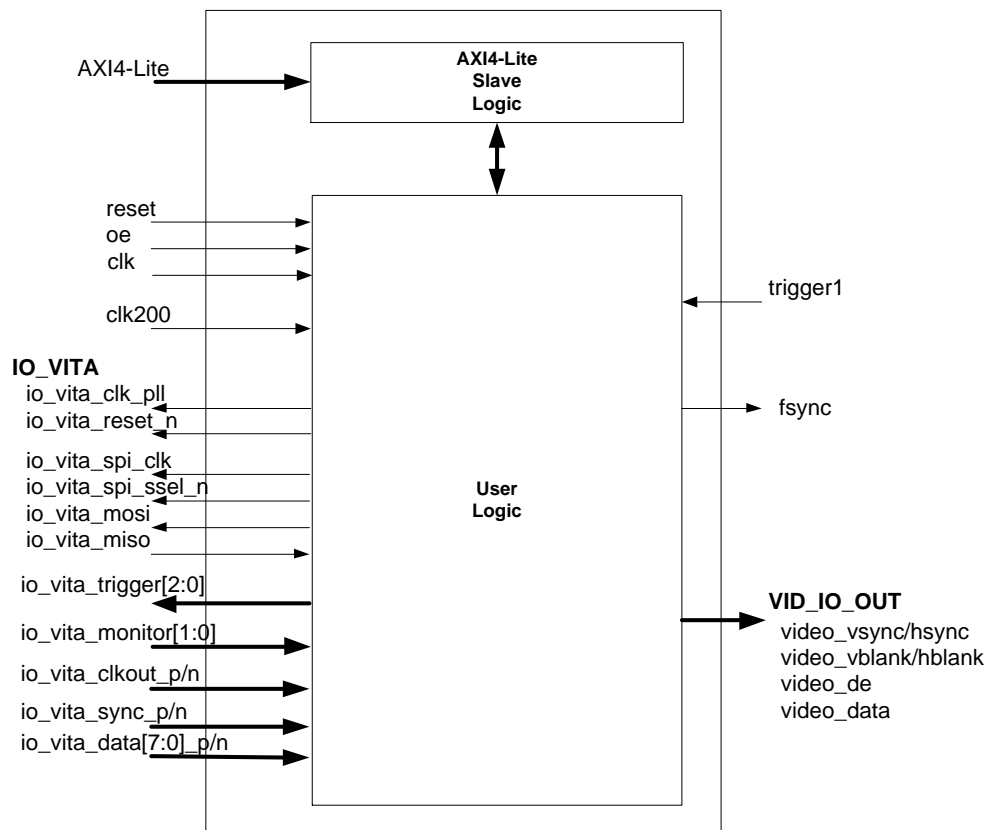


Figure 6 – fmc_imageon_vita_receiver – Block Diagram

The following figure illustrates the `fmc_imageon_vita_core` sub-module.

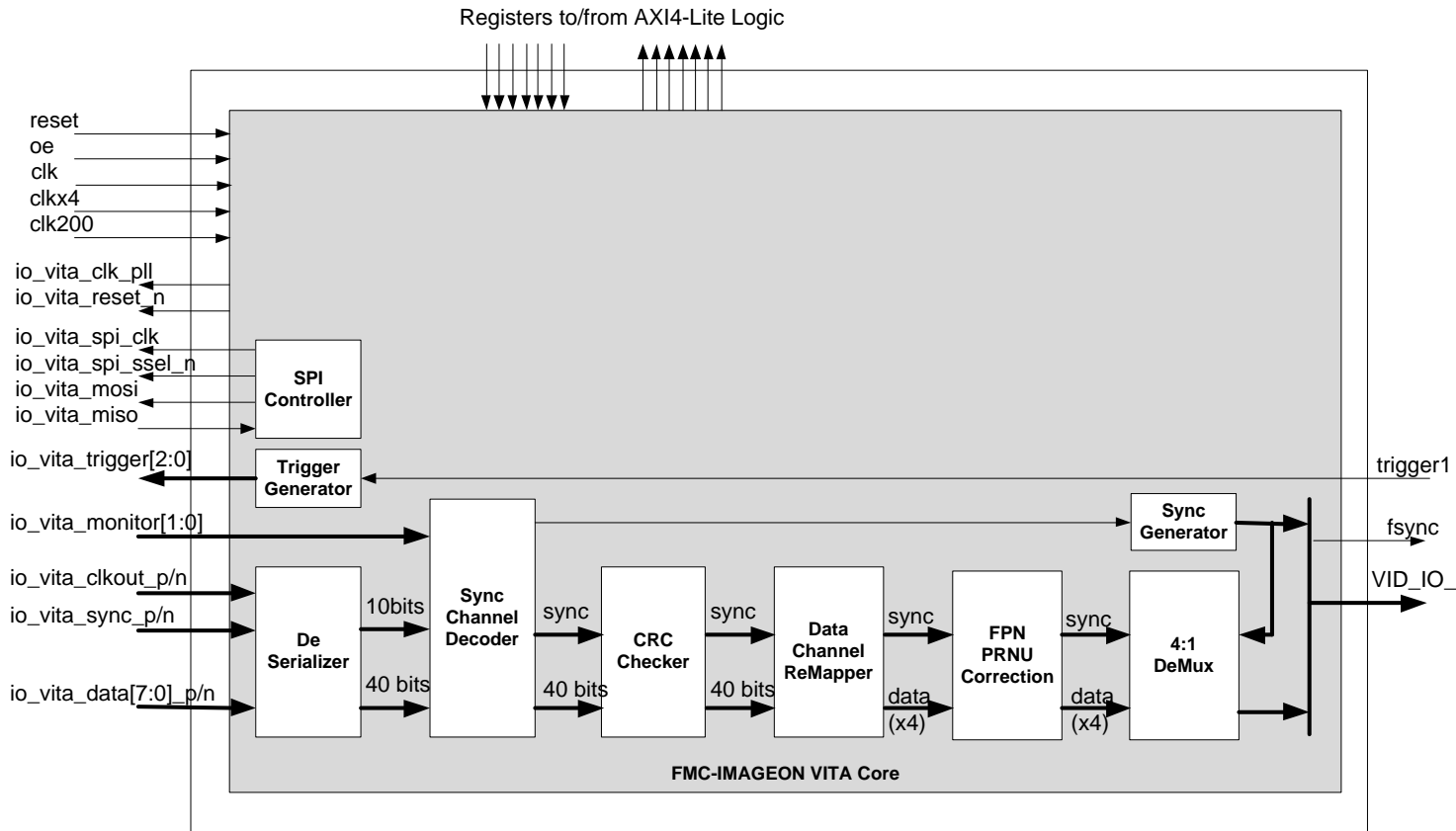


Figure 7 – `fmc_imageon_vita_core` – Block Diagram

The `fmc_imageon_vita_core` module implements the vita receiver. This module can be reused by itself if the AXI wrapper is not needed or desired.

The “De-Serializer” module implements the Serial LVDS receiver.

The “Sync Channel Decoder” module interprets the SYNC channel and generates appropriate synchronization signals.

The “CRC Checker” verifies the integrity of the Serial LVDS link by validating the CRC checksums that are sent with each line of video.

The “Sync Generator” generates the standard VSYNC/HSYNC, VBLANK/HBLANK, DE synchronization signals needed to create the generic timed video interface (VID_IO_OUT).

The “4:1 DeMux” module takes the 4 parallel pixel values synchronous to clock “clk”, and re-generates a single stream of pixels synchronous to clock “clkx4”. This module contains a FIFO that can store up to 6 lines of active video.

The fsync_o port generates a pulse that is active for 1 cycle at the start of each new frame. It can be used to synchronize video DMA transfers.

Parameters

Parameter	Description
C_FAMILY	FPGA Family, determines implementation. Supported values are : zynq
C_VIDEO_DATA_WIDTH	Video data width for VID_IO_OUT bus interface. Supported values are: 8 => contains 8 most significant bits of pixel 10 (default) => contains full 10 bit pixel 16 => bottom byte contains 8 bit pixel, top byte is 0x80 (no chroma) 24 => each byte contains 8 bit pixel 40 64
C_VIDEO_DIRECT_OUTPUT	When active, the video data contains 4 parallel pixel values: 0 (default) 1
C_DEBUG_PORT	This option will make the debug ports visible and accessible for debug purposes

Table 8 – fmc_imageon_vita_receiver – Parameter List

In default mode, the C_VIDEO_DIRECT_OUTPUT will be 0, in which case, the incoming video data will be de-multiplexed, before being sent to the VID_IO_OUT bus interface. In this case, the clk port is expected to be 4x the internal clock (ie. io_vita_clk_pll).

When the C_VIDEO_DIRECT_OUTPUT is set to 1, the incoming video data will be output directly to the VID_IO_OUT bus interface. In this case, the clk port will be the same frequency as the internal clock (ie. io_vita_clk_pll)..

Ports

Port	Direction	Bus Interface	Description
*	*	S_AXI	AXI4-Lite Slave Port
reset	I	-	Reset
oe	I	-	Output enable
clk	I	-	Clock input (video pixel rate)
clk200	I	-	200 MHz input clock
xsvi_active_video_o	O	VID_IO_OUT	Data Enable strobe
xsvi_vsync_o	O	VID_IO_OUT	Vertical Sync strobe
xsvi_hsync_o	O	VID_IO_OUT	Horizontal Sync strobe
xsvi_vblank_o	O	VID_IO_OUT	Vertical Sync strobe
xsvi_hblank_o	O	VID_IO_OUT	Horizontal Sync strobe
xsvi_video_data_o[]	O	VID_IO_OUT	Video pixel data
io_vita_clk_pll	O	IO_VITA	reference clock to VITA sensor
io_vita_reset_n	O	IO_VITA	Active low reset to VITA sensor
io_vita_trigger[2:0]	O	IO_VITA	Trigger strobes to VITA sensor
io_vita_monitor[1:0]	I	IO_VITA	Monitor strobes from VITA sensor
io_vita_spi_sclk	O	IO_VITA	SPI clock to VITA sensor
io_vita_spi_ssel_n	O	IO_VITA	SPI select to VITA sensor
io_vita_spi_mosi	O	IO_VITA	SPI data to VITA sensor
io_vita_spi_miso	I	IO_VITA	SPI data from VITA sensor
io_vita_clk_out_p/n	I	IO_VITA	Differential clock from VITA sensor
io_vita_sync_p/n	I	IO_VITA	Differential sync from VITA sensor
io_vita_data[7:0]_p/n	I	IO_VITA	Differential data from VITA sensor

Table 9 – fmc_imageon_vita_receiver – Port List

Registers

Offset	Register	R/W	Description
0x00000000	SPI_CONTROL	read/write	[0] VITA_RESET [1] SPI_RESET [8] SPI_STATUS_BUSY [9] SPI_STATUS_EROR [16] SPI_TXFIFO_FULL [23] SPI_RXFIFO_EMPTY
0x00000004	SPI_TIMING	read/write	[[15:0] SPI_TIMING
0x00000008	SPI_TXFIFO_DATA	read/write	[31:0]
0x0000000C	SPI_RXFIFO_DATA	read/write	[31:0]
0x00000010	ISERDES_CONTROL		[0] ISERDES_RESET [1] ISERDES_AUTO_ALIGN [2] ISERDES_ALIGN_START [3] ISERDES_FIFO_ENABLE [8] ISERDES_CLK_READY [9] ISERDES_ALIGN_BUSY [10] ISERDES_ALIGNED

0x00000014	ISERDES_TRAINING	read/write	[9:0] ISERDES_TRAINING
0x00000018	ISERDES_MANUAL_TAP	read/write	[9:0] ISERDES_MANUAL_TAP
0x0000001C	-	-	-
0x00000020	DECODER_CONTROL	read/write	[0] DECODER_RESET [1] DECODER_ENABLE
0x00000024	DECODER_STARTODDEVEN	read/write	[31:0] DECODER_STARTODDEVEN
0x00000028	DECODER_CODES_LS_LE	read/write	[15:0] CODE_LS (line start) [31:16] CODE_LE (line end)
0x0000002C	DECODER_CODES_FS_FE	read/write	[15:0] CODE_FS (frame start) [31:16] CODE_FE (frame end)
0x00000030	DECODER_CODES_BL_IMG	read/write	[15:0] CODE_BL (black line) [31:16] CODE_IMG (image line)
0x00000034	DECODER_CODES_TR_CRC	read/write	[15:0] CODE_TR (training) [31:16] CODE_CRC (crc)
0x00000038	DECODER_CNT_BLACK_LINES	read	[31:0] CNT_BLACK_LINES
0x0000003C	DECODER_CNT_IMAGE_LINES	read	[31:0] CNT_IMAGE_LINES
0x00000040	DECODER_CNT_BLACK_PIXELS	read	[31:0] CNT_BLACK_PIXELS
0x00000044	DECODER_CNT_IMAGE_PIXELS	read	[31:0] CNT_IMAGE_PIXELS
0x00000048	DECODER_CNT_FRAMES	read	[31:0] CNT_FRAMES
0x0000004C	DECODER_CNT_WINDOWS	read	[31:0] CNT_WINDOWS
0x00000050	DECODER_CNT_CLOCKS	read	[31:0] CNT_CLOCKS
0x00000054	DECODER_CNT_START_LINES	read	[31:0] CNT_START_LINES
0x00000058	DECODER_CNT_END_LINES	read	[31:0] CNT_END_LINES
0x0000005C	SYNCGEN_DELAY	read/write	[15:0] DELAY (# of clk cycles)
0x00000060	SYNCGEN_HTIMING1	read/write	[15:0] HACTIVE (active video) [31:0] HFPOrch (front porch)
0x00000064	SYNCGEN_HTIMING2	read/write	[14:0] HSYNCW (sync width) [15] HSYNCP (sync polarity) [31:16] HBPOrch (back porch)
0x00000068	SYNCGEN_VTIMING1	read/write	[15:0] VACTIVE (active video) [31:0] VFPOrch (front porch)
0x0000006C	SYNCGEN_VTIMING2	read/write	[14:0] VSYNCW (sync width) [15] VSYNCP (sync polarity) [31:16] VBPOrch (back porch)
0x00000070	CRC_CONTROL	read/write	[0] CRC_RESET [1] CRC_INITVALUE
0x00000074	CRC_STATUS	read/write	[3:0] CRC_STATUS
0x00000078	REMAPPER_CONTROL	read/write	[2:0] REMAPPER_WRITE_CFG [6:4] REMAPPER_MODE
0x0000007C	-	-	-
0x00000080	FPN_PRNU_VALUES[31:0]	read/write	[7:0] PRNU_0 [15:8] FPN_0 [23:16] PRNU_1 [31:24] FPN_1
0x00000084	FPN_PRNU_VALUES[63:32]	read/write	[7:0] PRNU_2 [15:8] FPN_2

			[23:16] PRNU_3 [31:24] FPN_3
0x00000088	FPN_PRNU_VALUES[95:64]	read/write	[7:0] PRNU_4 [15:8] FPN_4 [23:16] PRNU_5 [31:24] FPN_5
0x0000008C	FPN_PRNU_VALUES[127:96]	read/write	[7:0] PRNU_6 [15:8] FPN_6 [23:16] PRNU_7 [31:24] FPN_7
0x00000090	FPN_PRNU_VALUES[159:128]	read/write	[7:0] PRNU_8 [15:8] FPN_8 [23:16] PRNU_9 [31:24] FPN_9
0x00000094	FPN_PRNU_VALUES[191:160]	read/write	[7:0] PRNU_10 [15:8] FPN_10 [23:16] PRNU_11 [31:24] FPN_11
0x00000098	FPN_PRNU_VALUES[223:192]	read/write	[7:0] PRNU_12 [15:8] FPN_12 [23:16] PRNU_13 [31:24] FPN_13
0x0000009C	FPN_PRNU_VALUES[255:224]	read/write	[7:0] PRNU_14 [15:8] FPN_14 [23:16] PRNU_15 [31:24] FPN_15
0x000000A0	-	-	-
0x000000A4	-	-	-
0x000000A8	-	-	-
0x000000AC	-	-	-
0x000000B0	-	-	-
0x000000B4	-	-	-
0x000000B8	-	-	-
0x000000BC	-	-	-
0x000000C0	DECODER_CNT_MONITOR0_HIGH	read	[31:0] MONITOR0_HIGH
0x000000C4	DECODER_CNT_MONITOR0_LOW	read	[31:0] MONITOR0_LOW
0x000000C8	DECODER_CNT_MONITOR1_HIGH	read	[31:0] MONITOR1_HIGH
0x000000CC	DECODER_CNT_MONITOR1_LOW	read	[31:0] MONITOR1_LOW
0x000000D0	-	-	-
0x000000D4	-	-	-
0x000000D8	-	-	-
0x000000DC	TRIGGEN_EXT_DEBOUNCE	read/write	[31:0] EXT_DEBOUNCE
0x000000E0	TRIGGEN_CONTROL	read/write	[2:0] TRIGGEN_ENABLE [6:4] SYNC2READOUT [8] READOUTTRIGGER [16] EXT_POLARITY [24] CNT_UPDATE

			[30:28] GEN_POLARITY
0x000000E4	TRIGGEN_DEFAULT_FREQ	read/write	[31:0] TRIG_DEFAULT_FREQ
0x000000E8	TRIGGEN_TRIG0_HIGH	read/write	[31:0] TRIG0_HIGH
0x000000EC	TRIGGEN_TRIG0_LOW	read/write	[31:0] TRIG0_LOW
0x000000F0	TRIGGEN_TRIG1_HIGH	read/write	[31:0] TRIG1_HIGH
0x000000F4	TRIGGEN_TRIG1_LOW	read/write	[31:0] TRIG1_LOW
0x000000F8	TRIGGEN_TRIG2_HIGH	read/write	[31:0] TRIG2_HIGH
0x000000FC	TRIGGEN_TRIG2_LOW	read/write	[31:0] TRIG2_LOW

Table 10 – fmc_imageon_vita_receiver – Registers

The SYNCGEN_* registers define the settings for the video sync generator's video timing. The following figure illustrates how the synchronization signals will be generated from these settings.

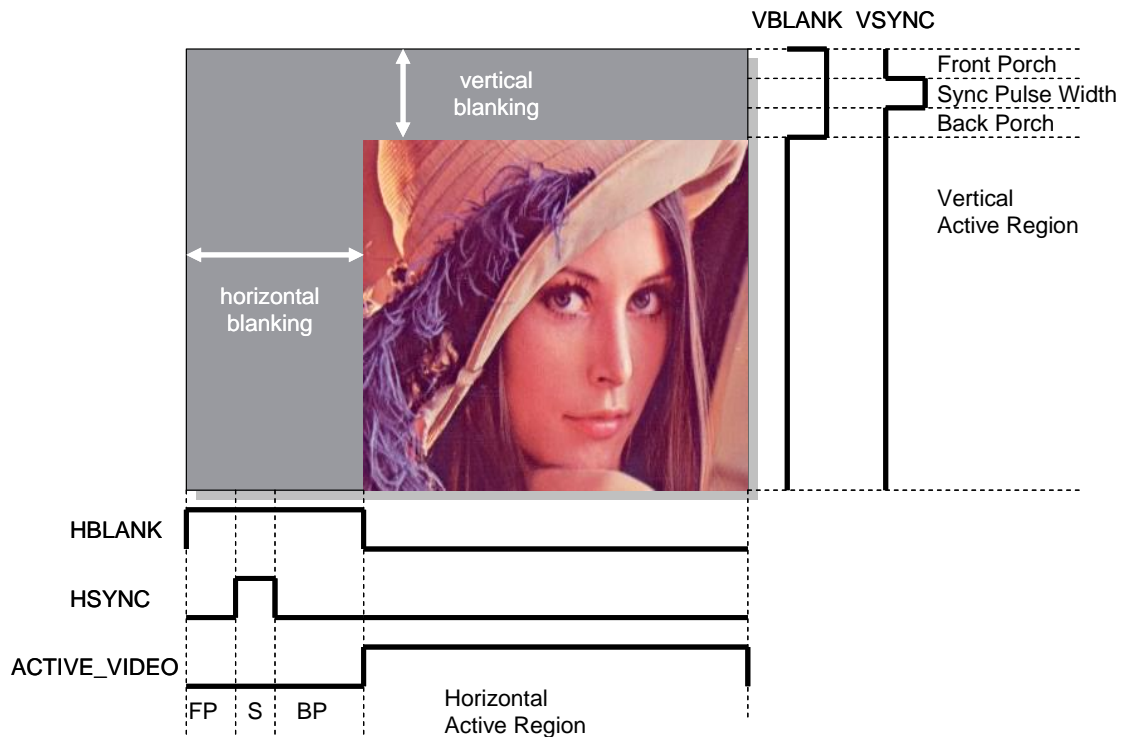


Figure 8 – fmc_imageon_vita_receiver – SYNCGEN Video Timing Diagram

FMC-IMAGEON – TCL Scripts

In order to assist the user to quickly create video designs based on the FMC-IMAGEON module, TCL scripts for Vivado IP Integrator have been provided. These scripts will create sub-modules which instantiate the FMC-IMAGEON IP cores, as well as the Xilinx video IP cores that they are typically connected to.

The following table contains the complete list of TCL Scripts that are provided for the FMC-IMAGEON FMC module.

TCL Script	Description
FMC-IMAGEON – HDMI Input	
fmc_imageon_hdmii_yuv422.tcl	Provides 16 bit YUV 4:2:2 video format to user
fmc_imageon_hdmii_rgb.tcl	Provides 24 bits RGB video format to user
FMC-IMAGEON – HDMI Output	
fmc_imageon_hdmio_y.tcl	Provides 8 bit grayscale video format to user
fmc_imageon_hdmio_yuv422.tcl	Provides 16 bit YUV 4:2:2 video format to user
fmc_imageon_hdmio_rgb.tcl	Provides 24 bits RGB video format to user
FMC-IMAGEON – VITA Receiver	
fmc_imageon_vita_raw.tcl	Provides 8 bits raw pixels to user
fmc_imageon_vita_mono.tcl	Provides 8 bits grayscale video format to user
fmc_imageon_vita_color.tcl	Provides 24 bits RGB video format to user

Table 11 – FMC-IMAGEON – TCL Scripts Overview

The next sections describe the TCL Scripts in greater detail.

fmc_imageon_hdmii_yuv422.tcl

This TCL script will create the following Vivado IP Integrator sub-module, which instantiates the “HDMI Input” and “Video Input to AXI4-Stream” IP cores.

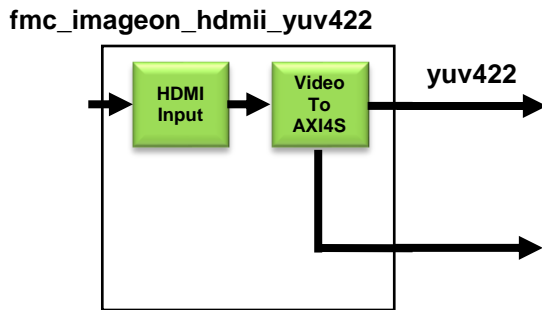


Figure 9 – HDMI Input – 16 bits YUV 4:2:2 video format

fmc_imageon_hdmii_rgb.tcl

This TCL script will create the following Vivado IP Integrator sub-module, which instantiates the “HDMI Input” and “Video Input to AXI4-Stream” IP cores, as well as the “Chroma Resampler” and “YCbCr to RGB” IP cores.

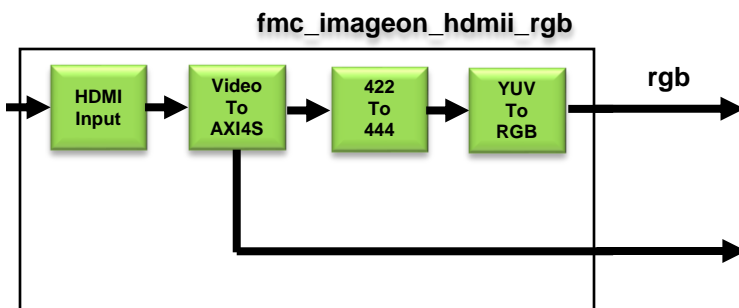


Figure 10 – HDMI Input – 24 bits RGB video format

fmc_imageon_hdmio_y.tcl

This TCL script will create the following Vivado IP Integrator sub-module, which instantiates the “AXI4-Stream to Video Output” and “HDMI Output” IP cores.

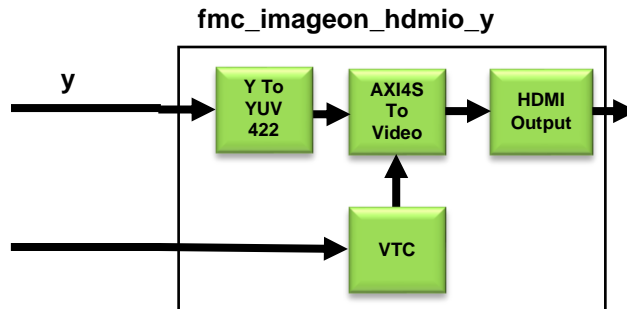


Figure 11 – HDMI Output – 8 bits monochrome video format

fmc_imageon_hdmio_yuv422.tcl

This TCL script will create the following Vivado IP Integrator sub-module, which instantiates the “AXI4-Stream to Video Output” and “HDMI Output” IP cores.

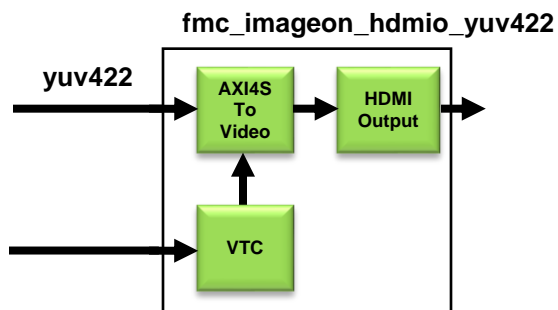


Figure 12 – HDMI Output – 16 bits YUV 4:2:2 video format

fmc_imageon_hdmio_rgb.tcl

This TCL script will create the following Vivado IP Integrator sub-module, which instantiates the “AXI4-Stream to Video Output” and “HDMI Output” IP cores, as well as the “Chroma Resampler” and “RGB to YCbCr” IP cores

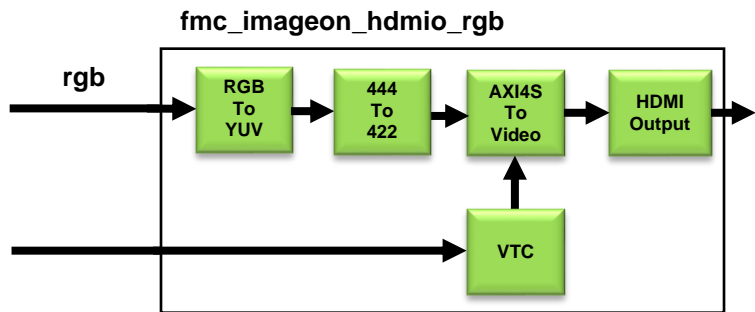


Figure 13 – HDMI Output – 24 bits RGB video format

fmc_imageon_vita_raw.tcl

This TCL script will create the following Vivado IP Integrator sub-module, which instantiates the “VITA Receiver” and “Video Input to AXI4-Stream” IP cores, without any additional video IP cores.

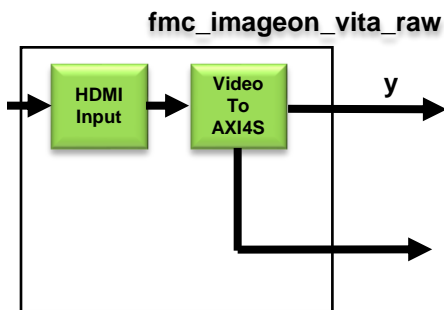


Figure 14 – VITA Receiver – Raw Pixels

fmc_imageon_vita_mono.tcl

This TCL script will create the following Vivado IP Integrator sub-module, which instantiates the “VITA Receiver” and “Video Input to AXI4-Stream” IP cores, as well as the “Defect Pixel Correction” (DPC) IP core.

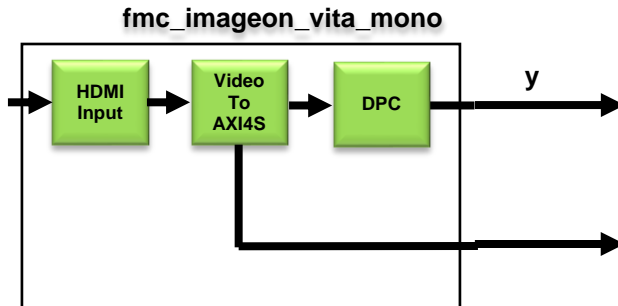


Figure 15 – VITA Receiver – Monochrome Sensor

The Defect Pixel Correction (DPC) core is used to identify and correct defective pixels, as shown in the following figure

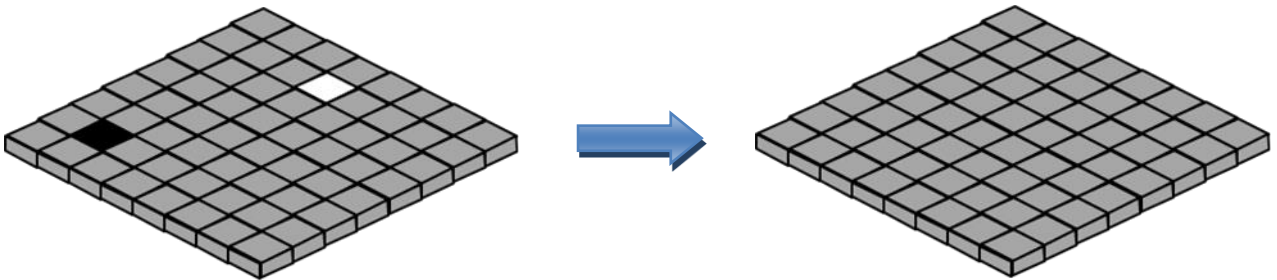


Figure 16 – Defective Pixel Correction

Defective pixels are typical for image sensors. Image sensor manufacturers tolerate certain types of defective pixels, assuming that they can be corrected downstream by an image processing pipeline.

The following image illustrates typical acceptance criteria for defective pixels.

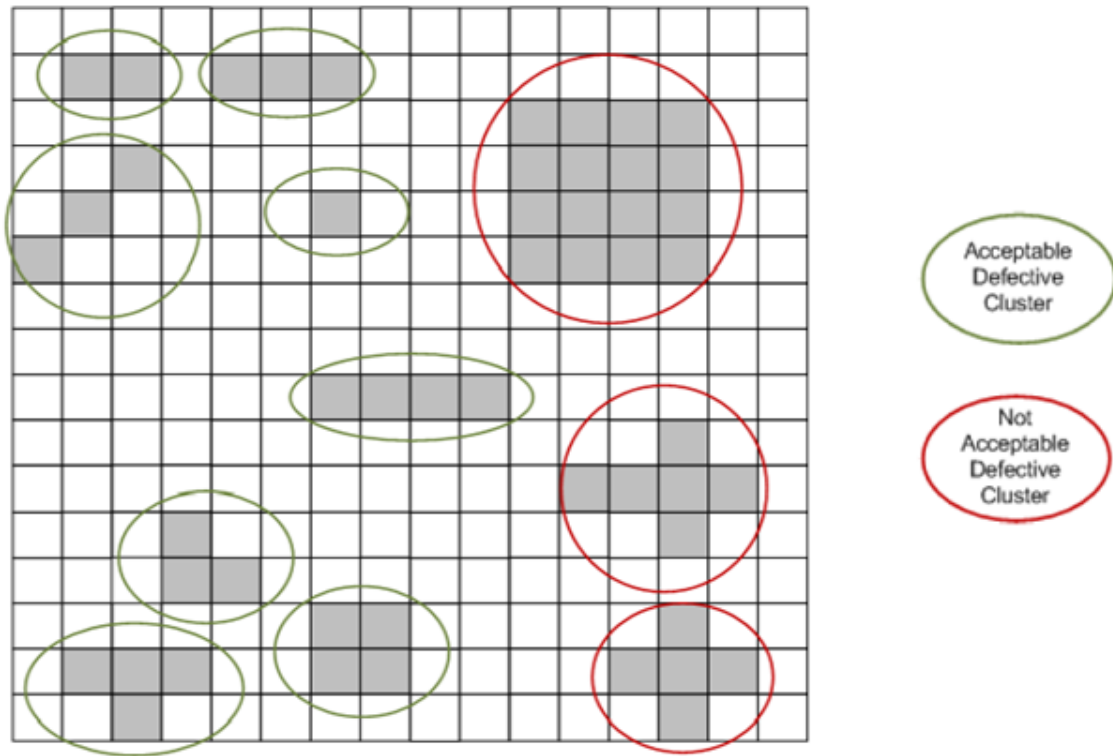


Figure 17 – Defective Pixel Acceptance Criteria

fmc_imageon_vita_color.tcl

This TCL script will create the following Vivado IP Integrator sub-module, which instantiates the “VITA Receiver” and “Video Input to AXI4-Stream” IP cores, as well as the “Defect Pixel Correction” (DPC) and “Color Filter Array Interpolation” (CFA) IP cores.

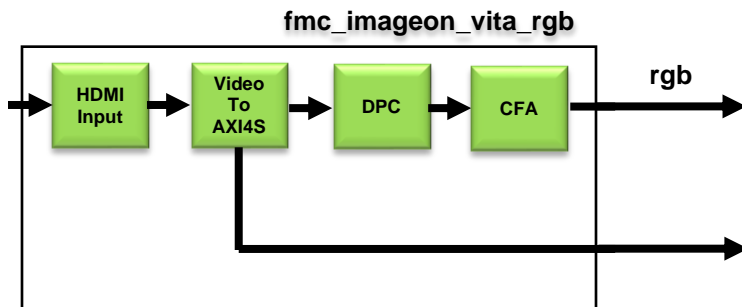


Figure 18 – VITA Receiver – Color Sensor

The most widespread and cost effective implementation of color image sensors involves placing color filters on top of each pixel in a Bayer pattern arrangement.

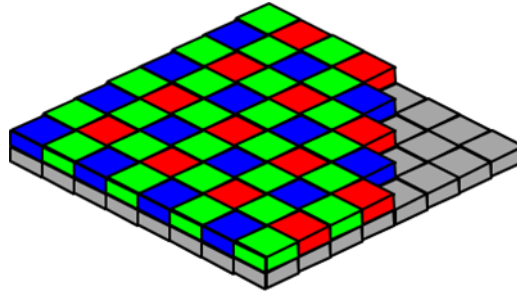


Figure 19 – Color Filters in Bayer Format

With this arrangement of color filters, each pixel captures only one of the three primary colors. The Color Filter Array interpolation (CFA) core restores the missing two colors based on neighboring pixels.

FMC-IMAGEON – Software Libraries & Drivers

The following table contains the complete list of software libraries and drivers provided in the archive and where to find documentation:

Driver	Version	Description
Generic Software Libraries		
fmc_iic_sw	2.03 a	Library which provides a generic I2C controller API.
fmc_ipmi_sw	2.02 a	Library which provides an API to access the contents of the FMC IPMI EEPROMs.
FMC-IMAGEON specific Software Library		
fmc_imageon_sw	2.01 a	Library which provides an API to perform I2C configuration of the FMC-IMAGEON module's HDMI input, HDMI output, and Video clock synthesizer peripherals.
VITA Receiver specific Drivers		
fmc_imageon_vita_receiver	2.01 a	Library which provides an API to perform SPI configuration of the FMC-IMAGEON module's VITA receiver.

Table 12 – IP Repository – Software Libraries Overview

The next sections describe the TCL Scripts in greater detail.

fmc_iic_sw (2.03 a)

The documentation for the FMC-IIC software library can be found in the following location:

```
FMC_IMAGEON\2013_3\avnet_fmc_imageon_cores\sw_services\
fmc_iic_sw_v2_03_a\doc\FMC-IIC_Software_Guide_1_03.pdf
```

fmc_ipmi_sw (2.02 a)

The documentation for the FMC-IPMI software library can be found in the following location:

FMC_IMAGEON\2013_3\avnet_fmc_imageon_cores\sw_services\
fmc_ipmi_sw_v2_02_a\doc\FMC-IPMI_Software_Guide_1_01.pdf

fmc_imageon_sw (2.01 a)

The FMC-IMAGEON software library can be found in the following location:

```
FMC_IMAGEON\2013_3\avnet_fmc_imageon_cores\sw_services\
fmc_imageon_sw_v1_05_a\fmc_imageon.c/.h
```

The FMC-IMAGEON software library consists of the following functions.

Function	Description
General Functions	
fmc_imageon_init	Initialize instance of FMC-IMAGEON software library.
I2C Multiplexer Function	
fmc_imageon_iic_mux_reset	Reset FMC-IMAGEON's I2C multiplexer
fmc_imageon_iic_mux	Select the I2C multiplexer's active ports.
I2C Configuration Functions	
fmc_imageon_iic_config2	Perform generic I2C configuration using specified register/value matrix
fmc_imageon_iic_config3	Perform generic I2C configuration using specified device/register/value matrix
Video Clock Synthesizer (CDCE913) Functions	
fmc_imageon_vclk_init	Initialize video clock synthesizer.
fmc_imageon_vclk_config	Set video clock synthesizer to specified frequency.
HDMI Input (ADV7611) Functions	
fmc_imageon_hdmii_init	Initialize HDMI input interface
fmc_imageon_hdmii_set_hpd	Set HDMI input interface's Hot Plug Detect (HPD) pin
fmc_imageon_hdmii_set_rst	Set HDMI input interface's Reset pin
fmc_imageon_hdmii_get_int	Get HDMI input interface's Interrupt (INT1) pin status
fmc_imageon_hdmii_get_lock	Get HDMI input interface's lock status
fmc_imageon_hdmii_get_timing	Get HDMI input interface's detected video timing
HDMI Output (ADV7511) Functions	
fmc_imageon_hdmio_init	Initialize HDMI output interface.
fmc_imageon_hdmio_set_pd	Set HDMI output interface's Power Down (PD)
fmc_imageon_hdmio_get_hpd	Get HDMI output interface's Hot Plug Detect (HPD) pin status

DDC/EDID EEPROM Functions	
fmc_imageon_hdmi_read_edid	Read HDMI input interface's EDID EEPROM.
fmc_imageon_hdmi_write_edid	Write HDMI input interface's EDID EEPROM.
fmc_imageon_hdmi_read_edid	Read HDMI output interface's EDID EEPROM.
Delay Functions	
fmc_imageon_wait_usec	Wait for specified number of micro-seconds. Requires and calls external usleep() function.

Table 13 – fmc_imageon_sw – Function List

For more information on each of these functions, please refer to the software library's source code.

fmc_imageon_vita_receiver (2.01 a)

The FMC-IMAGEON VITA Receiver driver can be found in the following location

```
FMC_IMAGEON\2013_3\avnet_fmc_imageon_cores\drivers\
fmc_imageon_vita_receiver_v2_01_a\src\fmc_imageon_vita_receiver.c/.h
```

The FMC-IMAGEON VITA Receiver driver consists of the following functions.

Function	Description
General Functions	
fmc_imageon_vita_receiver_init	Initialize instance of FMC-IMAGEON VITA receiver driver.
Low-Level Functions	
fmc_imageon_vita_receiver_reg_read	Read value of specified VITA receiver register.
fmc_imageon_vita_receiver_reg_write	Write to specified VITA receiver register.
fmc_imageon_vita_receiver_reset	Set value of reset to VITA sensor.
fmc_imageon_vita_receiver_spi_config	Configure the VITA receiver's SPI controller
fmc_imageon_vita_receiver_spi_read	Perform SPI transaction to read VITA sensor register
fmc_imageon_vita_receiver_spi_write	Perform SPI transaction to write VITA sensor register
fmc_imageon_vita_receiver_write_sequence	Perform sequence of SPI transactions.
fmc_imageon_vita_receiver_display_sequence	Display sequence of SPI transactions (for debug purposes)
High-Level Functions	
fmc_imageon_vita_receiver_sensor_initialize	Initialize the VITA sensor and VITA Receiver IP Core (using Global Master Mode)
fmc_imageon_vita_receiver_get_status	Get status of VITA receiver
fmc_imageon_vita_receiver_sensor_1080P60	Initialize the VITA-2000 sensor and the VITA Receiver IP Core for 1080P60 resolution (using Triggered Slave Mode)
fmc_imageon_vita_receiver_set_analog_gain	Set VITA sensor's analog gain
fmc_imageon_vita_receiver_set_digital_gain	Set VITA sensor's digital gain

fmc_imageon_vita_receiver_set_exposure_time	Set VITA sensor's exposure time
---	---------------------------------

Table 14 – fmc_imageon_vita_receiver – Function List

For more information on each of these functions, please refer to the driver's source code.

FMC-IMAGEON – Application Code Examples

The following table contains the complete list of example application code provided for the FMC-IMAGEON module. Each of these examples is accompanied by a tutorial describing how to create the entire design.

Application Code	Description
HDMI Pass-Through	Application Code that configures the HDMI input (ADV7611) and HDMI output (ADV7511) interfaces for a HDMI pass-through design.
VITA Pass-Through	
HDMI Display Controller	

Table 15 – FMC-IMAGEON – Application Code Examples

The next sections describe the application code examples in greater detail.

HDMI Pass-Through

The HDMI Pass-Through example code initializes the HDMI input and HDMI output interfaces on the FMC-IMAGEON module for the following two pass-through hardware designs.

One pass-through will make the 16 bits YUV 4:4:2 video format available for the user, which corresponds to the video format of the external HDMI input (ADI ADV7611) and HDMI output (ADI ADV7511) devices

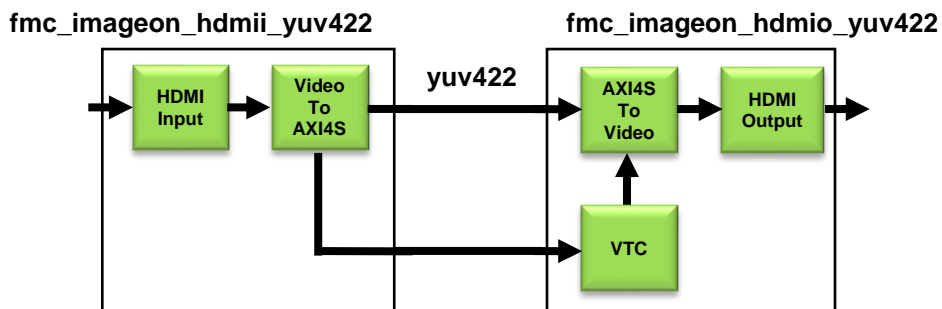


Figure 20 –HDMI Pass-Through – 16 bits YUV 4:2:2

An alternate pass-through makes use of the Chroma Resample (422 To 444 / 444 To 422) and Color Space Conversion (YUV to RGB / RGB to YUV) cores to make the 24 bits RGB video format available to the user.

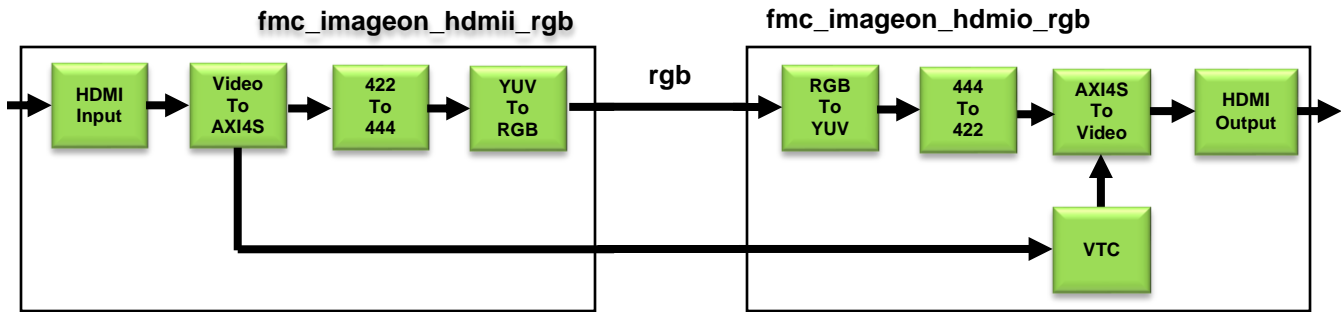


Figure 21 –HDMI Pass-Through – 24 bits RGB

Refer to the **FMC-IMAGEON HDMI Pass-Through Tutorial** document for detailed instructions on how to build this design.

VITA Pass-Through

The VITA Pass-Through example code initializes the VITA-2000 image sensor, the VITA receiver, and the HDMI output interface on the FMC-IMAGEON module for the following two pass-through hardware designs.

One pass-through can be used for a color image sensor.

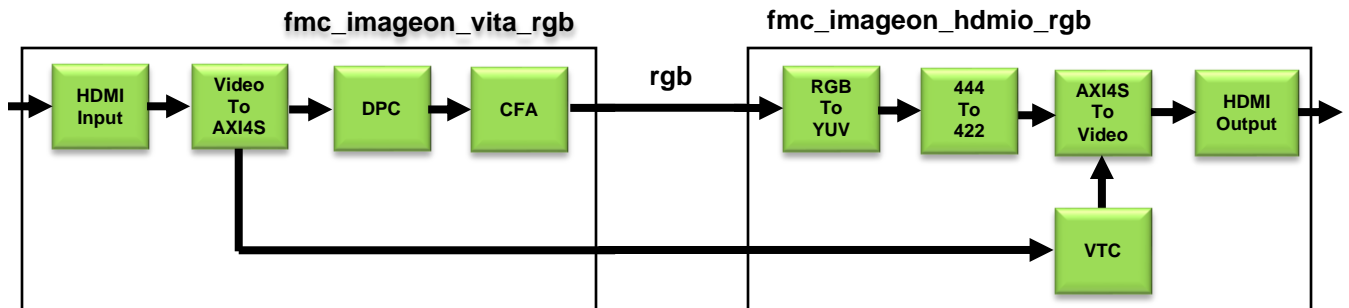


Figure 22 –VITA Pass-Through – Color Sensor

An alternate pass-through can be used for a monochrome image sensor. A monochrome sensor will only have the Y (or luma) component, which is intensity. The AXI4-Stream Protocol Converter core can be used to set the interleaved UV components (or chroma) to 0x80, effectively creating color-less pixels in the 16 bits YUV 4:2:2 video format.

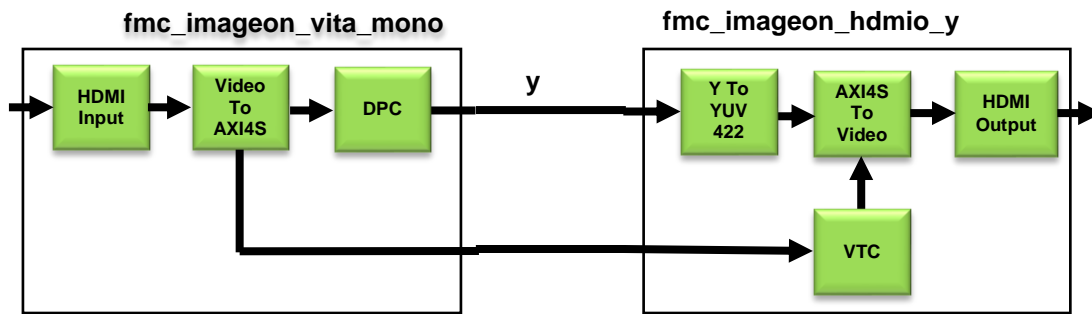


Figure 23 –VITA Pass-Through – Monochrome Sensor

HDMI Display Controller

The HDMI display controller will make use of the AXI Video DMA IP core, as shown below.

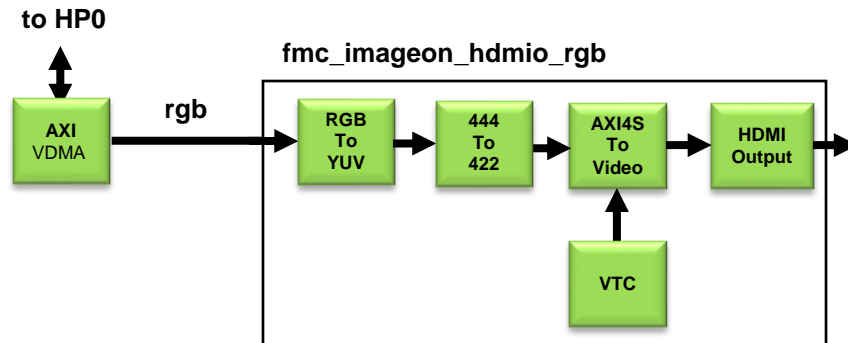


Figure 24 – HDMI Display Controller

The AXI Video DMA core will fetch video content from the Zynq's external memory via one of the High-Performance Ports (HP0), and produce video content in the AXI4-Stream protocol.

Refer to the **FMC-IMAGEON HDMI Display Controller Tutorial** document for detailed instructions on how to build this design.

References

All documentation supporting the ON Semiconductor Image Sensor with HDMI Input/Output FMC Bundle is available on the Avnet Design Resource Center (DRC):

<http://www.em.avnet.com/fmc-imageon-v2000c>

1. Getting Started with the HDMI Input/Output FMC Module
<http://www.em.avnet.com/fmc-imageon> → Support Files & Downloads
2. Avnet FMC-IMAGEON – Hardware User Guide
<http://www.em.avnet.com/fmc-imageon> → Support Files & Downloads
3. Getting Started with the ON Semiconductor Image Sensor with HDMI Input/output FMC Bundle
<http://www.em.avnet.com/fmc-imageon-v2000c> → Support Files & Downloads

The following reference provides links to documentation supporting video application notes and video intellectual property (IP).

4. Color Filter Array Interpolation
<http://www.xilinx.com/products/ipcenter/EF-DI-CFA.htm>
5. Color Correction Matrix
<http://www.xilinx.com/products/ipcenter/EF-DI-CCM.htm>
6. Defective Pixel Correction
<http://www.xilinx.com/products/ipcenter/EF-DI-DEF-PIX-CORR.htm>
7. Gamma Correction
<http://www.xilinx.com/products/ipcenter/EF-DI-GAMMA.htm>
8. Image Edge Enhancement
<http://www.xilinx.com/products/ipcenter/EF-DI-IMG-ENHANCE.htm>
9. Image Noise Reduction
<http://www.xilinx.com/products/ipcenter/EF-DI-IMG-NOISE.htm>
10. Image Statistics Engine
<http://www.xilinx.com/products/ipcenter/EF-DI-IMG-STATS.htm>
11. Motion Adaptive Noise Reduction
<http://www.xilinx.com/products/ipcenter/EF-DI-IMG-MA-NOISE.htm>

12. RGB to YCrCb Color-Space Converter

http://www.xilinx.com/products/ipcenter/RGB_to_YCrCb.htm

13. YCrCb to RGB Color-Space Converter

http://www.xilinx.com/products/ipcenter/YCrCb_to_RGB.htm

14. Video Direct Memory Access (DMA)

<http://www.xilinx.com/products/ipcenter/EF-DI-VID-DMA.htm>

15. Video On Screen Display (OSD)

<http://www.xilinx.com/products/ipcenter/EF-DI-OSD.htm>

16. Video Scaler

<http://www.xilinx.com/products/ipcenter/EF-DI-VID-SCALER.htm>

17. Video Timing Controller

<http://www.xilinx.com/products/ipcenter/EF-DI-VID-TIMING.htm>

The following reference provides links to documentation for AXI interconnect.

18. UG761 - AXI Reference Guide

19. PG065 – AXI4-Stream Infrastructure

Known Issues and Limitations

The following issues are known to exist. When applicable, the workaround used is described.

HDMI Output – HDMIO_CLK – reducing radiated emissions

Spread Spectrum Clocking (SSC) is used on the FMC-IMAGEON module's HDMI output interface (HDMIO_CLK) to reduce radiated emissions to industry approved levels. This can be implemented using the SSC feature of the on-board TI CDCE913 video clock synthesizer.

The recommended setting is to modulate the clock using down-spread clocking by an amount of -0.75%. This technique significantly lowers the radiated emissions, while maintaining functionality on the HDMI output interface.

The following I2C registers settings will program the CDCE913 for down-spread clocking of -0.75%.

Setting	Register	Value	Description
SSC1DC	0x16[7]	0	PLL1 SSC down/center selection = down
SSC1	0x10[2:0]	011	PLL1 SSC Selection (Modulation Amount) = -0.75%

Table 16 – CECE913 I2C Register Settings for SSC

Troubleshooting

ERROR: ADV7611 has not locked on incoming video, aborting !

If you get the following output from the serial console:

```
Hello World

-----
--          FMC-IMAGEON HDMI Pass-Through          --
-----

FMC-IMAGEON Initialization ...
HDMI Input Initialization ...
Waiting for ADV7611 to locked on incoming video ...
      ERROR : ADV7611 has NOT locked on incoming video, aborting !
```

Your video input is not connected or not active. Please re-verify your video input source by connecting it directly to your HDMI/DVI monitor.