XAPP741 (v1.3) April 14, 2014

Designing High-Performance Video Systems in 7 Series FPGAs with the AXI Interconnect

Authors: Pankaj Kumbhare and Vamsi Krishna

Summary

This application note covers the design considerations of a video system using the performance features of the LogiCORE™ IP Advanced eXtensible Interface (AXI) Interconnect core. The design focuses on high system throughput using approximately 80% of DDR memory bandwidth through the AXI Interconnect core with F_{MAX} and area optimizations in certain portions of the design.

The design uses eight AXI video direct memory access (AXI VDMA) engines to simultaneously move 16 streams (eight transmit video streams and eight receive video streams), each in 1920 x 1080 pixel format at 60 Hz refresh rate, and 24 data bits per pixel. This design also has additional video equivalent AXI traffic generated from four LogiCORE AXI Traffic Generator (ATG) cores configured for 1080p video mode. The ATG core generates continuous AXI traffic based on its configuration. In this design, ATG is configured to generate AXI4 video traffic in 1080p mode. This pushes the system throughput requirement to approximately 80% of DDR bandwidth. Each AXI VDMA is driven from a LogiCORE IP Test Pattern Generator (AXI TPG) core. AXI VDMA is configured to operate in free running mode. Data read by each AXI VDMA is sent to a common Video On-Screen Display (AXI OSD) core capable of multiplexing or overlaying multiple video streams to a single output video stream. The output of the AXI OSD core drives the onboard high-definition media interface (HDMI™ technology) video display interface through the RGB to YCrCb Color Space Converter core and LogicCORE IP Chroma Resampler core. A LogiCore Video Timing Controller (AXI VTC) generates the required timing signals.

The LogiCORE AXI Performance Monitor core is added to capture DDR performance metrics. DDR traffic is passed through the AXI Interconnect to move 16 video streams over 8 VDMA pipelines and 8 video streams of traffic from the four ATG cores configured for 1080p video mode. All 16 video streams moved by the AXI VDMA blocks are buffered through a shared DDR3 SDRAM memory and are controlled by a MicroBlaze™ processor. The ATG cores continuously generate video traffic to be stored in DDR memory.

The reference system is targeted for the Kintex®-7 FPGA XC7K325TFFG900-2 on the Xilinx KC705 evaluation board, revision 1.1. (See the *KC705 Evaluation Board for the Kintex-7 FPGA User Guide* (UG810) [Ref 1].)

Included Systems

The reference design is created and built using Vivado® IP Integrator (IPI) 2013.4, which is part of Vivado® Design Suite: System Edition. The Vivado IP integrator provides a device and platform aware, interactive environment that supports intelligent auto-connection of key IP interfaces, one-click IP subsystem generation, real time DRCs, and interface change propagation, combined with a powerful debug capability. Creation of the reference design using the Vivado tools logic design flow is described in detail in Building Hardware, page 14. The design also includes software built using the Xilinx Software Development Kit (SDK). The software runs on the MicroBlaze™ processor subsystem and implements control, status, and monitoring functions. Complete Vivado and SDK project files are provided with this application note to allow you to examine and rebuild the design or to use it as a template for starting a new design.

Introduction

High-performance video systems can be created using Xilinx AXI IP. The use of AXI Interconnect, Memory Interface Generator (MIG), and VDMA IP blocks can form the core of video systems capable of handling multiple video streams and frame buffers sharing a common DDR3 SDRAM. AXI is a standardized IP interface protocol based on the Advanced Microcontroller Bus Architecture (AMBA®) specification. The AXI interfaces used in the reference design consist of AXI4, AXI4-Lite, and AXI4-Stream interfaces as described in the AMBA AXI4 specifications [Ref 2]. These interfaces provide a common IP interface protocol framework around which to build the design.

Together, the AXI Interconnect and AXI MIG implement a high-bandwidth, multi-ported memory controller (MPMC) for use in applications where multiple devices share a common memory controller. This is a requirement in many video, embedded, and communications applications where data from multiple sources moves through a common memory device, typically DDR3 SDRAM.

AXI VDMA implements a high-performance, video-optimized DMA engine with frame buffering, scatter gather, and two-dimensional (2D) DMA features. AXI VDMA transfers video data streams to or from memory and operates under dynamic software control or static configuration modes.

A clock generator and processor system reset block supplies clocks and resets throughout the system. High-level control of the system is provided by an embedded MicroBlaze processor subsystem containing I/O peripherals and processor support IP. To optimize the system to balance performance and area, multiple AXI Interconnect blocks are used to implement segmented/hierarchical AXI Interconnect networks with each AXI Interconnect block individually tuned and optimized.

Hardware Requirements

The hardware requirements for this reference system are:

- Xilinx KC705 evaluation board (revision 1.1)
- One USB Type-A to Mini-B 5-pin cable
- One USB Type-A to Micro-B 5-pin cable
- High-quality HDMI to DVI cable (colors are not displayed properly otherwise)
- Display monitor supporting 1920 x 1080 pixel resolution up to 60 frames/sec (the reference design was tested using a Dell P2210T monitor)

The installed software tool required for building and downloading this reference system is Vivado Design Suite 2013.4.

Reference Design Specifics

In addition to the MicroBlaze processor, the reference design includes these cores:

- MDM
- LMB block RAM
- AXI_INTERCONNECT
- CLOCK GENERATOR
- PROC_SYS_RESET
- AXI_UARTLITE
- AXI IIC
- AXI_INTC
- MIG
- AXI_BRAM
- AXI_VTC
- AXI_TPG

- AXI_VDMA
- AXI_PERFORMANCE_MONITOR
- AXI_OSD
- AXI4-Stream to Video Out
- RGB2YCrCB Converter
- Chroma Resampler
- AXI Traffic Generator
- HDMI_OUT IP

Figure 1 and Table 1 show a block diagram and address map of the system, respectively.

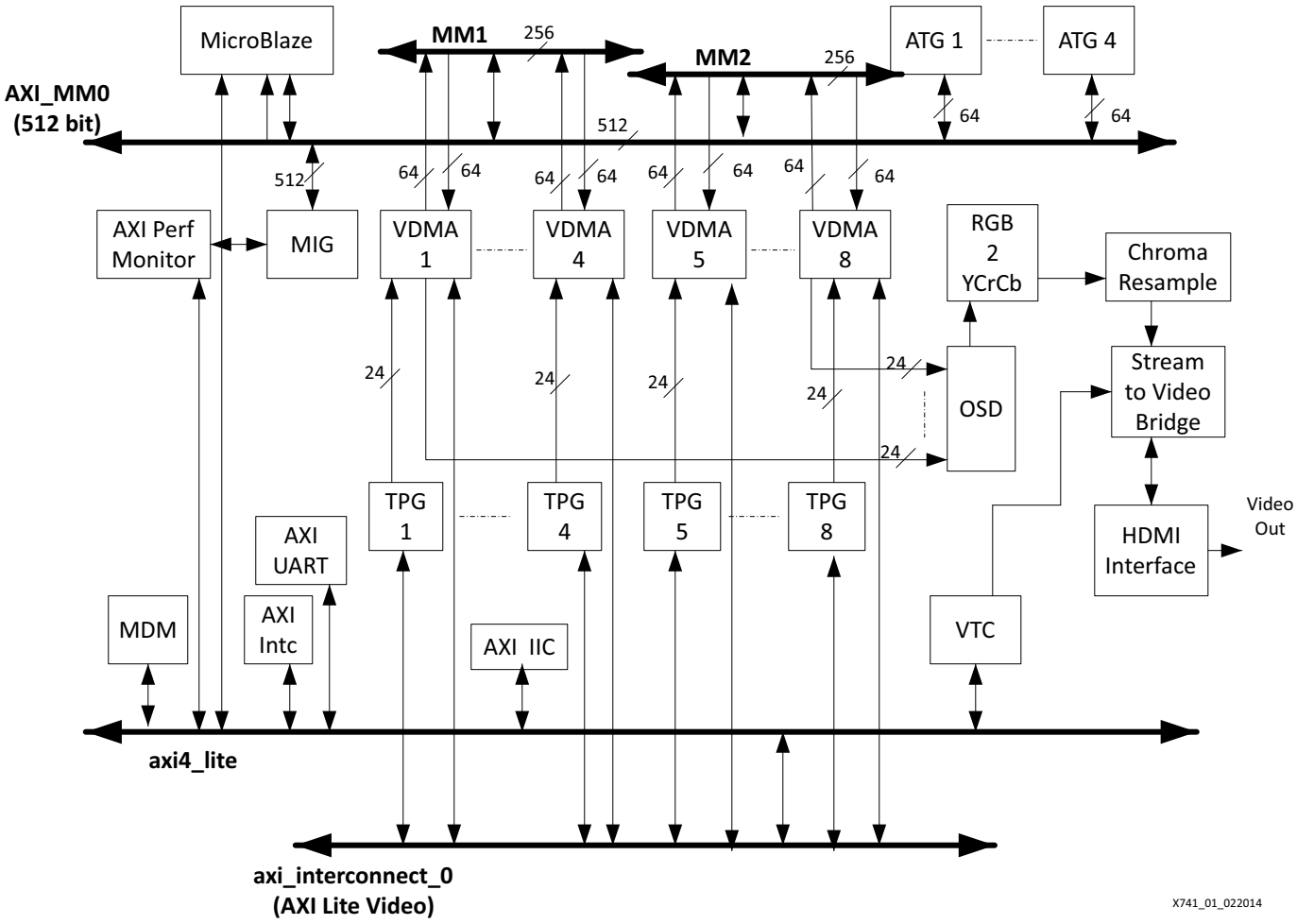

Figure 1: **Reference System Block Diagram**

Table 1: **Reference System Address Map**

Peripheral	Instance	Base Address	High Address
AXI interrupt controller	axi_intc_1	0x41200000	0x4120FFFF
LMB block RAM controller	microblaze_0_local_memory/dlmb_bram_if_cntlr	0x00000000	0x0001FFFF
LMB block RAM controller	microblaze_0_local_memory/ilmb_bram_if_cntlr	0x00000000	0x0001FFFF
MDM	mdm_1	0x41400000	0x4140FFFF

Table 1: **Reference System Address Map *(Cont'd)***

Peripheral	Instance	Base Address	High Address
AXI UartLite	axi_uartlite_1	0x40600000	0x4060FFFF
MIG	mig_1	0x80000000	0xBFFFFFFF
Video timing controller	v_tc_0	0x44AA0000	0x44AAFFFF
AXI IIC	axi_iic_1	0x40800000	0x4080FFFF
AXI TPG	v_tpg_0	0x74A10000	0x74A1FFFF
	v_tpg_1	0x74A30000	0x74A3FFFF
	v_tpg_2	0x74A50000	0x74A5FFFF
	v_tpg_3	0x74A70000	0x74A7FFFF
	v_tpg_4	0x74A90000	0x74A9FFFF
	v_tpg_5	0x74AB0000	0x74ABFFFF
	v_tpg_6	0x74AD0000	0x74ADFFFF
	v_tpg_7	0x74AF0000	0x74AFFFFF
AXI performance monitor	axi_perf_mon_0	0x44A10000	0x44A1FFFF
AXI On-Screen display	v_osd_0	0x44A00000	0x44A0FFFF
AXI VDMA	axi_vdma_0	0x74A00000	0x74A0FFFF
	axi_vdma_1	0x74A20000	0x74A2FFFF
	axi_vdma_2	0x74A40000	0x74A4FFFF
	axi_vdma_3	0x74A60000	0x74A6FFFF
	axi_vdma_4	0x74A80000	0x74A8FFFF
	axi_vdma_5	0x74AA0000	0x74AAFFFF
	axi_vdma_6	0x74AC0000	0x74ACFFFF
	axi_vdma_7	0x74AE0000	0x74AEFFFF

Hardware System Specifics

This section describes the high-level features of the reference design, including how to configure the main IP blocks. Information about useful IP features, performance/area trade-offs, and other configuration information is also provided. This information is applied to a video system, but the principles used to optimize the system performance apply to a wide range of high-performance AXI systems. For information about AXI system optimization and design trade-offs, see *AXI Reference Guide* (UG761) [Ref 3].

This application note assumes you have some general knowledge of the Vivado IP integrator feature. See *Vivado Design Suite Tutorial: Designing IP Subsystems Using IP Integrator* (UG995) [Ref 4] *and Vivado Design Suite User Guide: Designing IP Subsystems Using IP Integrator* (UG994) [Ref 5] for more information about the Vivado IP integrator feature.

Video-Related IP

The reference design implements eight video pipelines each running at a 1920 x 1080 pixel format at 60 frames/sec. Four ATG cores are added to generate additional video traffic to push throughput to ~ 80% of DDR bandwidth. Each picture consists of three bytes per pixel. Each video pipeline running at 60 frames/sec requires a bandwidth of 373.2 MB/s (~4 Gb/s). In addition, four ATG cores generate video traffic on DDR to push DDR to maximum bandwidth utilization.

The video traffic is generated by AXI TPG IP cores and displayed by the AXI OSD core. The four ATG cores generate one Read and one Write stream each. So there are 8 video streams for four ATG cores. Thus the total video streams is 24.The total aggregate read/write bandwidth generated is equivalent to that from 8 AXI VDMAs and 8 video streams from 4 ATG cores 9.355 GB/s (74.85 Gb/s).

The total aggregate read/write bandwidth generated is equivalent to 24 video streams requiring 8.96 GB/s (71.68 Gb/s). Because AXI VDMAs are operating in Free Running mode, write throughput is slightly greater than read throughput. So actual achieved throughput is greater than the abovementioned theoretical throughput.

This application note demonstrates AXI system performance using 24 (16 AXI VDMA and 8 ATG) high-definition video streams. At a minimum, video systems must include a source, some internal processing, and a destination. There can be multiple stages internally using several IP modules. The canonical video system in Figure 2 shows that most video systems consist of input, pre-processing, main processing, post-processing, and output stages. Many of the video stages illustrated require memory access at video rates. Video data goes in or out of memory according to the requirements of internal processing stages. In this application note, a series of test pattern generators create the internal IP block memory traffic to simulate typical conditions.

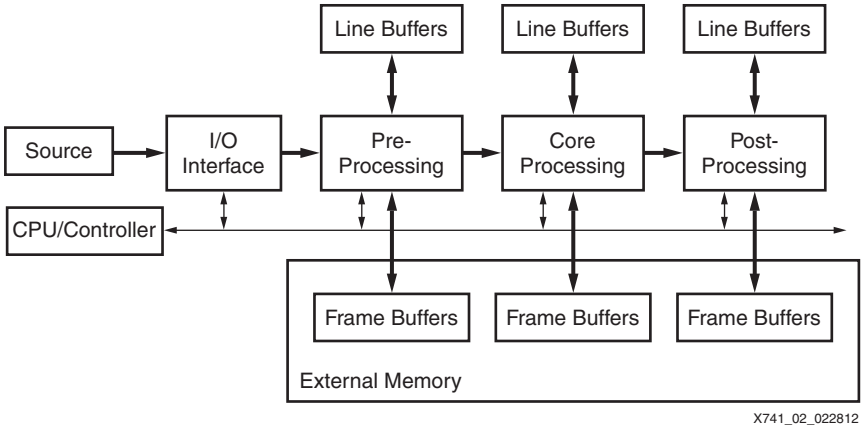

Figure 2: **Typical Video System**

AXI Interconnects

This design contains multiple AXI Interconnects, each tuned to balance for throughput, area, and timing considerations (see *LogiCORE IP AXI Interconnect Product Guide for Vivado Design Suite* (PG059) [Ref 6]). The AXI_MM0, AXI_MM1, and AXI_MM2 instances are used for high-speed masters and slaves that include high throughput and high F_{MAX} optimizations. The AXI_MM0, AXI_MM1, and AXI_MM2 interconnects are optimized for higher throughput. They are used to buffer frame data generated by the TPG and to access the same data from the buffer through the VDMA to display on the LCD. The axi4_lite and axi_interconnect_0 interconnect instances are generally optimized for area. They are used by the processor to access slave registers and to write to the VDMA register space for control of the AXI VDMA. The AXI VDMA operation and its register descriptions are described in detail in *LogiCORE IP AXI Video Direct Memory Access Product Guide for Vivado Design Suite* (PG020) [Ref 7].

AXI Interconnect (AXI_MM Instance)

This AXI Interconnect instance provides the highest F_{MAX} and throughput for the design by having a 512-bit core data width and running at 200 MHz. The AXI Interconnect core data width and clock frequency match the capabilities of the attached AXI MIG so that width and clock converters between them are not needed. Sizing the AXI Interconnect core data width and clock frequency below the native width and clock frequency of the memory controller creates a system bandwidth bottleneck in the system. To help meet the timing requirements of a 512-bit

AXI interface at 200 MHz, a bank of register slices are enabled between AXI_MM interconnect and AXI MIG. Together, AXI Interconnect and AXI MIG form a 22-port AXI MPMC connected to MicroBlaze processor instruction cache (ICache) and data cache (DCache) ports, four ATG ports, eight AXI VDMA MM2S ports, and eight AXI VDMA S2MM ports. The configuration of this AXI Interconnect is consistent with the system performance optimization recommendations for an AXI MPMC based system as described in the *AXI Reference Guide* (UG761) [Ref 3].

AXI VDMA Instances

The AXI VDMA core is designed to provide video read/write transfer capabilities from the AXI4 domain to the AXI4-Stream domain, and vice versa. The AXI VDMA provides high-speed data movement between system memory and AXI4-Stream based target video IP. AXI4 interfaces are used for the high-speed data movement and buffer descriptor fetches across the AXI Interconnect.

The AXI VDMA core incorporates video-specific functionality, that is, Gen-Lock and Frame Sync, for fully synchronized frame DMA operations and 2D DMA transfers. In addition to synchronization, frame store numbers and scatter gather or register direct mode operations are available for ease-of-control by the central processor.

In this design, the AXI VDMA scatter gather feature is not used because the system could be implemented sufficiently using the simpler register direct mode of AXI VDMA, which would remove the area cost of the scatter gather feature. Scatter gather should only be enabled if the system requires relatively complex software control of the AXI VDMA operations.

Initialization, status, and management registers in the AXI VDMA core are accessed through an AXI4-Lite slave interface.

This design uses eight instances of AXI VDMA, each using two 64-bit interfaces toward the AXI4 memory map and two 24-bit interfaces toward the streaming side. The upsizer in the VDMA is used to convert 24-bit transactions from the streaming side to 64-bit wide transactions to the memory map side of the VDMA core. Similarly, downsizers are used to convert 64-bit memory-mapped transactions to 24-bit streaming side transactions.

The 64-bit wide MM2S and S2MM interfaces from the AXI VDMA instances are connected to the AXI_MM instance of the AXI Interconnect. The upsizers in the interconnects MM1 and MM2 convert 64-bit VDMA MM interfaces to the native 256 bit widths of the interconnects. The masters run at 150 MHz (60 Hz frame rate) (video clock), which requires asynchronous clock converters to the 200 MHz AXI Interconnect core frequency. See Datapath Optimization to understand optimized settings between the VDMA and interconnects (MM0, MM1, and MM2).

For maximum throughput for the AXI VDMA instances, the maximum burst length is set to 256. In addition, AXI Interconnects are configured for maximized performance, which enables data FIFO on the master and slave side of the interconnect. These settings all follow performance recommendations for AXI endpoint masters as described in the *AXI Reference Guide* (UG761) [Ref 3].

In addition, line buffers inside the AXI VDMA for the read and write sides are set to 1K deep to improve system performance and reduce the risk of system throttling. See *LogiCORE IP AXI Video Direct Memory Access Product Guide for Vivado Design Suite* (PG020) [Ref 7] for more information.

To achieve optimal utilization of the memory controller, transactions from the VDMA master interfaces need to occur in different banks and must be aligned to KB/MB boundaries. With video designs, frame buffers need to be accessed in different banks, and the overlapping of banks must be minimized while the video design is running.

This design demonstrates 1080p60 frames (1920 x 1080) with three bytes per pixel. Each horizontal line is about 8 KB (1920 x 3 = 5760), so the AXI VDMA line stride is set to 8 KB boundaries.

The start of every new line is aligned on an 8 KB boundary. The vertical lines of each frame (1080 lines per frame) are aligned to a 2 KB boundary for each frame. Therefore, each frame buffer in this design is aligned on 16 MB boundaries (8 KB x 2 KB).

The previous calculation is taken care of while defining the frame buffer for VDMA in the application provided with this application note.

Datapath Optimization

A possible datapath configuration between the VDMA and DDR3 controller using balanced hierarchical interconnects is shown in Figure 3. Transaction data width conversion is done at the VDMA and at the AXI4 Interconnects (MM0, MM1, and MM2).

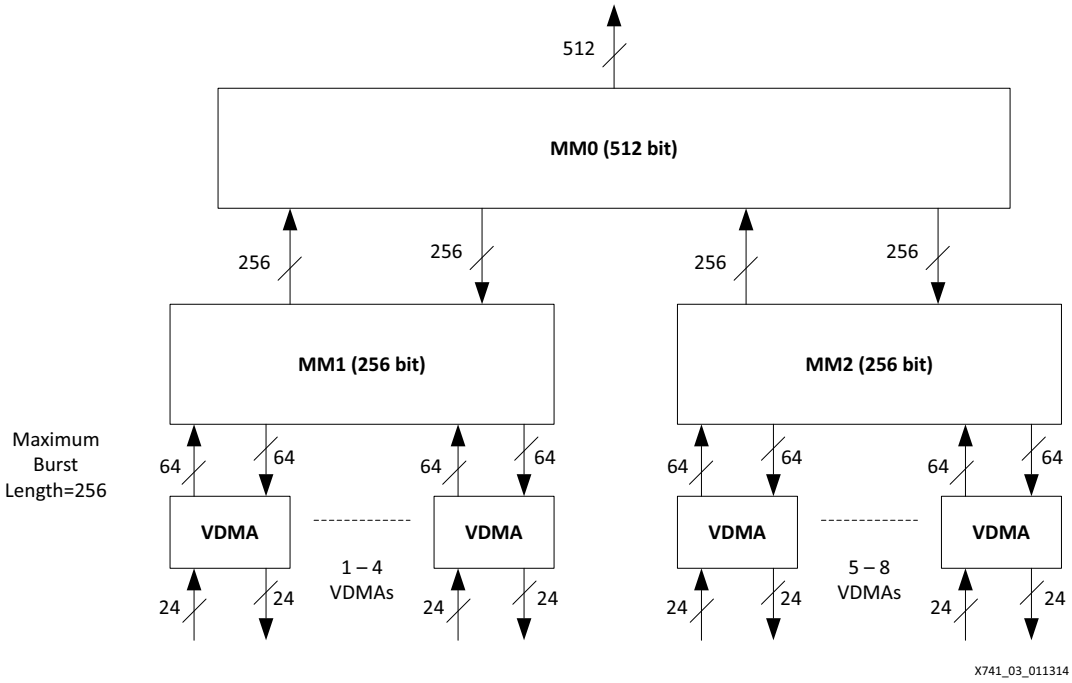

Figure 3: **Balanced Interconnect Configuration**

MicroBlaze Processor ICache and DCache

The MicroBlaze processor ICache and DCache masters are connected to the AXI Interconnect and run at 100 MHz because the MicroBlaze processor runs a software application from main memory that sets up and monitors the video pipelines. Running the MicroBlaze processor at this frequency helps timing and area.

See the *Vivado Design Suite User Guide: Embedded Processor Hardware Design* (UG898) [Ref 8] for more information. The 100 MHz clock setting ensures that synchronous integer ratio clock converters in the AXI Interconnect can be used, which offers lower latency and less area than asynchronous converters.

Memory Interface Generator (MIG)

The single slave connected to the AXI Interconnect is the MIG. The memory controller AXI interface is 512 bits wide running at 200 MHz and disables narrow burst support for optimal throughput and timing. This configuration matches the native AXI interface clock and width corresponding to a 64-bit DDR3 DIMM at 800 MHz memory clock, which is the maximum performance of the memory controller for a Kintex-7 device in -2 speed grade.

The slave interface has a read/write issuance of eight. Register slices are enabled to ensure that the interface meets timing at 200 MHz. These settings help ensure that a high degree of transaction pipelining is active to improve system throughput. See the *Zynq-7000 SoC and 7 Series Devices Memory Interface Solutions Data Sheet* (DS176) [Ref 9] for more information about the memory controller.

AXI Interconnect (axi4_lite, axi_interconnect_0)

The MicroBlaze processor data peripheral (DP) interface master writes and reads to all AXI4-Lite slave registers in the design for control and status information.

These interconnects are 32 bits and do not require high F_{MAX} and throughput. Therefore, they are connected to a slower F_{MAX} portion of the design by a separate AXI Interconnect.

Because there are more than 16 AXI4-Lite slave interfaces in the design, additional AXI Interconnect instances are connected in cascaded form to allow the processor to access all the AXI4-Lite interfaces in the system.

The axi4_lite and axi_interconnect_0 interconnect blocks are configured for shared-access mode because high throughput is not required in this portion of the design. Therefore, area can be optimized over performance on these interconnect blocks. Also, these two interconnects are clocked at 50 MHz to ensure that synchronous integer ratio clock converters in the AXI Interconnect can be used, which offer lower latency and less area than asynchronous clock converters.

axi4_lite Interconnect

The slaves on the AXI_Lite interconnect are for MDM, AXI_UARTLITE, AXI_IIC, AXI_INTC, AXI_VTC AXI OSD, PERF_MON and the connectors to the axi_interconnect_0 interconnect.

axi_interconnect_0 Interconnect

The eighth slave of axi4_lite interconnect is connected as master to this interconnect. The slaves on this AXI Interconnect are AXI_TPG (eight instances) and the AXI VDMA slave interface (eight instances).

Video Timing Controller

The LogiCORE IP Video Timing Controller (AXI VTC) is a general purpose video timing generator and detector. The input side of this core automatically detects horizontal and vertical synchronization pulses, polarity, blanking timing, and active video pixels. The output side of the core generates the horizontal and vertical blanking and synchronization pulses used in a standard video system including support for programmable pulse polarity.

The AXI VTC contains an AXI4-Lite Interface to access slave registers from a processor. For more information about the AXI VTC, see the *LogiCORE IP Video Timing Controller Product Guide for Vivado Design Suite* (PG016) [Ref 10].

In this design, the AXI VTC instance is used without detection. The AXI VTC instance is used for the AXI OSD, which is the read portion of the video pipelines.

Test Pattern Generator

The LogiCORE IP Test Pattern Generator (AXI TPG) contains an AXI4-Lite interface to access slave control registers from a processor.

In this reference design, the video traffic to DDR3 memory is generated by a series of AXI TPGs. Each AXI TPG block can generate several video test patterns that are commonly used in the video industry for verification and testing. In the reference design, the AXI TPG is used as a replacement for other video IP because only the amount of traffic generated to demonstrate the performance of the system is of interest. The control software demonstrates generation of flat colors, color bars, horizontal and vertical burst patterns, and the generation of zone plates.

No matter which test pattern is chosen, the amount of data generated is the same, namely, 1080p HD video. For example, an RGB (24-bit), 1080p60 pattern generates 373.2 MB/s, which is a nearly 4 Gb/s data stream.

Several operating modes are accessible through software control. In this application note, the AXI TPG always generates a test pattern that could be one of flat colors, color bars, horizontal ramp, vertical ramp, or zoneplates. These patterns are meant for testing purposes only and are not calibrated to broadcast industry standards.

Streaming to Video Bridge

The LogiCORE IP AXI4-Stream in to Video Out core (see *LogiCORE IP AXI4-Stream to Video Out Product Guide for Vivado Design Suite* (PG044) [Ref 11]) is used to convert data from the AXI4-Stream video protocol interface to video domain interface. The Streaming to Video core works in conjunction with the timing generator portion of the AXI VTC core. This core provides a bridge between the AXI4 Stream video input from AXI OSD with video output (synchronization signals or active video with either syncs, blanks, or both) interfaces.

The core is configured in slave mode as it receives a streaming video signal from AXI OSD core and timing signals from AXI VTC core to generate Video out signals to the HDMI interface.

AXI Traffic Generator

The AXI Traffic Generator (ATG) core can be used to generate heavy traffic in several pre-defined protocols like Video, Ethernet, and PCIe®. In this application, the ATG core is configured to generate video traffic in 1080p 60 Hz format to put additional load on DDR to handle 24 video streams

RGB to YCrCb Converter

This IP core is used to convert the incoming video stream in RGB format to YCrCb format which can be given as input to the HDMI interface.

Video On-Screen Display

The LogiCORE IP Video On-Screen Display (AXI OSD) provides a flexible video-processing block for alpha blending, compositing up to eight independent layers, and generating simple text and graphics capable of handling images up to 4K x 4K sizes in YUV 422,YUV444, YUVA,YUV420 RGB, and RGBA image formats in 8, 10, or 12 bits per color component. In this application note, the AXI OSD blends the eight video streams as separate display layers. Because the video streams generated by the AXI TPG cores are enabled through software control, the display shows the blended layers on top of each other. Figure 4 shows a three-level block diagram of the AXI OSD core.

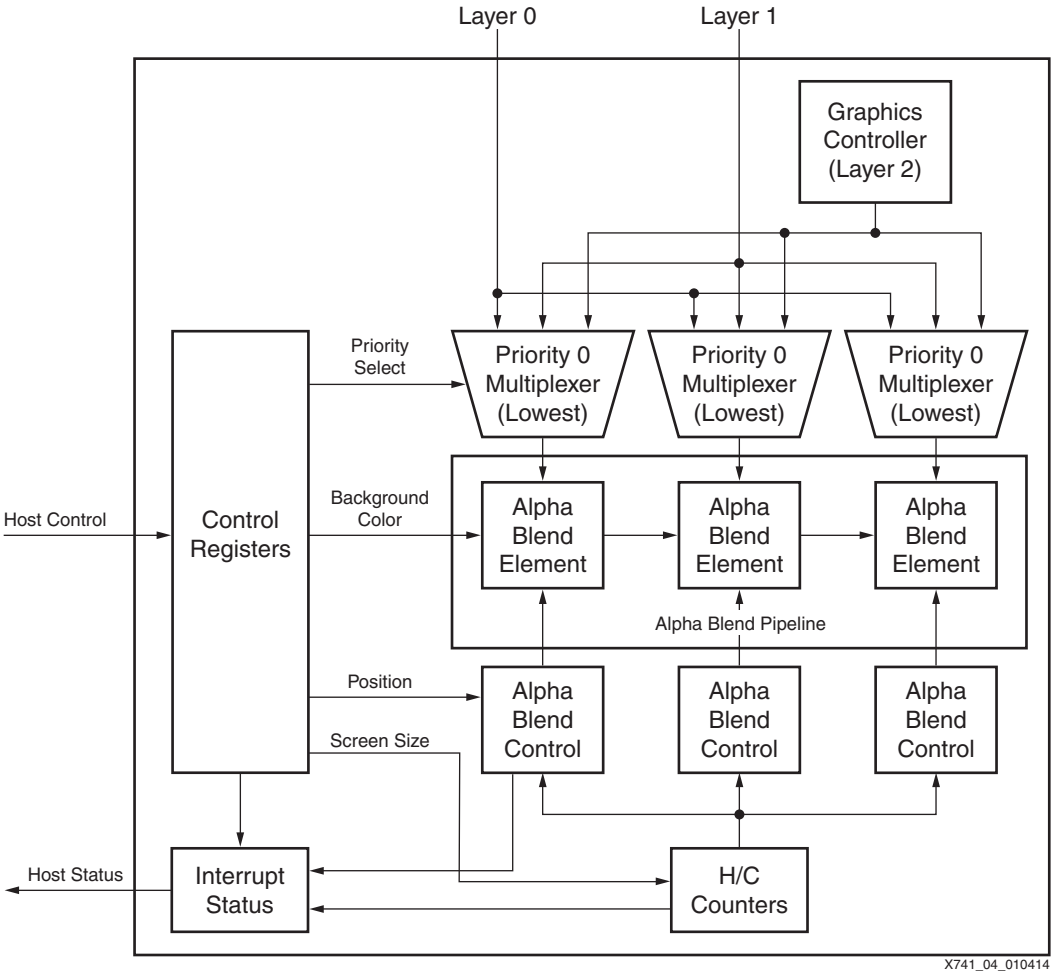

Figure 4: **Sample Three-Layer OSD Core Block Diagram**

The AXI OSD contains an AXI4-Lite interface to access the slave registers from a processor. For more information about the AXI OSD, see the *LogiCORE IP Video On-Screen Display Product Guide for Vivado Design Suite* (PG010) [Ref 12].

Chroma Resampler

This IP core is used to convert between different video formats such as YUV422, YUV444, and YUV420. In this application, the Chroma Resampler core converts the incoming video in YUV444 format to YUV422 format.

AXI Performance Monitor

The AXI performance monitor core (AXI PERFORMANCE MONITOR) measures throughput for a DDR3 memory connected to the AXI Interconnect. The processor accesses the AXI performance monitor core registers through a slave AXI4-Lite interface contained in the core. The AXI performance monitor core only monitors the read and write channels between the AXI slave and the AXI Interconnect. The core does not modify or change any of the AXI transactions it is monitoring.

Several signals must be connected in the system to measure the throughput. The DDR slave interconnect (AXI_MM0) is connected to one slot of the monitor. In addition to these, the AXI_Lite bus interface is connected to access the core registers by the processor. In addition to the signals of these two bus interfaces, the core clock (the higher of the two bus interface clock frequencies) must be connected.

The core can measure performance metrics such as total read byte count, write byte count, read requests, write requests, and write responses. Count start and count end conditions come from the processor through the register interface. The global clock counter of the core measures the number of clocks between the count start and count end events. The counters used for the performance monitor can be configured for 32 or 64 bits through the register interface. Final user-selectable metrics can also be read through the register interface.

In this application note, the DDR3 slave is connected to one of the slots of the AXI performance monitor core to measure the throughput of the core. Valid, ready, strobe, and other AXI signals connected to the performance monitor slots are used to enable various counters for measuring events on the bus.

Software Applications

AXI VDMA DISPLAY Application

The software application starts up the video pipelines, allowing you to examine bandwidth in real time and display separate layers or alpha blend all layers on the LCD screen.

Application-level software for controlling the system is written in C using the provided drivers for each IP. The programmer model for each IP describes the particular API used by the drivers. Alternatively, application software can be written to use the IP control registers directly and handle the interrupts at the application level, but using the provided drivers and a layer of control at the application level is a far more convenient option.

The application software in the reference design performs these actions:

1. The software application first resets the HDMI technology port on the KC705 board through the IIC interface.
2. The AXI TPG instances are set to write a default color bar pattern that does not start until the AXI VTC instances are started.
3. The AXI VDMA instances are started, which consists of the processor writing into its registers. The program then starts the read/write channels to begin the transfers for the VDMA instances.
4. The AXI VTC instances are started with 1920 x 1080 pixels (60 Hz) timing configuration.
5. The AXI OSD is configured for 1920 x 1080 resolution output.

   The eight AXI TPG instances in the design are configured to write:

   - Color bars (layer 0)
   - Zone plate patterns (layer 1)
   - Vertical bars (layer 2)
   - Horizontal bars (layer 3)
   - Tartan bars (layer 4)
   - Flat red (layer 5)
   - Flat green (layer 6)
   - Flat blue (layer 7)

After the initial setup sequence, you can choose to view a certain layer by selecting a number (option 0–7). When the number of a particular layer is selected, the AXI OSD registers are modified to make the alpha blending on that particular layer be the highest value, while the others are at the smallest. When option 8 is selected (alpha blending all layers), different values are given to the alpha blending register for each layer to show all layers on the LCD screen at the same time. Option 9 reads performance metrics from the core. The four ATG cores are always enabled and continuously generating video traffic on DDR.

Executing the Reference Design in Hardware

This section provides instructions to execute the reference design in hardware. This reference design runs on the KC705 board shown in Figure 5.

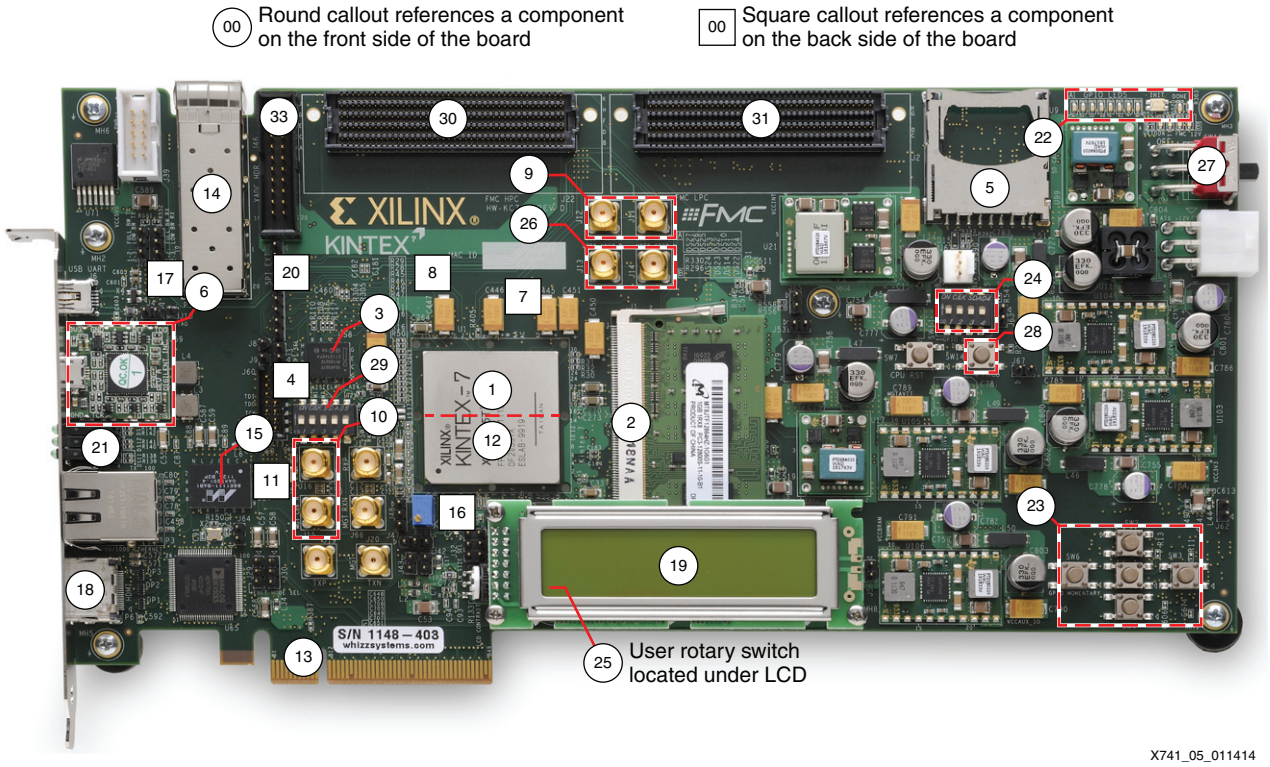

Figure 5: **KC705 Board**

In these instructions, numbers in parentheses correspond to callout numbers in Figure 5. Not all callout numbers are referenced.

Note: See the *KC705 Evaluation Board for the Kintex-7 FPGA User Guide* (UG810) [Ref 1] for a complete description of all the callout numbers.

1. Connect a USB cable from the host PC to the USB JTAG port (6). Ensure the appropriate device drivers are installed.
2. Connect a second USB cable from the host PC to the USB UART port. Ensure that the USB UART drivers described in Hardware Requirements, page 2 have been installed.
3. Connect the KC705 HDMI technology connector (18) to a video monitor capable of displaying a 1920 x 1080 resolution and displaying up to 60 Hz video signal.
4. Connect a power supply cable.
5. Set power ON (27).
6. Start a terminal program (for example, HyperTerminal) on the host PC with these settings:
   - Baud Rate: 9600
   - Data Bits: 8
   - Parity: None
   - Stop Bits: 1
   - Flow Control: None

Executing the Reference System Using the Pre-Built Bitstream and the Compiled Software Application

These are the steps to execute the system using files in the `ready_for_download` directory of the `<unzip_dir>/kc705_video_8x_pipeline/` directory:

1. In a Xilinx command shell or terminal window, change directories to the `ready_for_download` directory:

   % cd <unzip dir>/kc705_video_8x_pipeline/ready_for_download
2. Invoke the Xilinx Microprocessor Debugger (XMD) tool:

   % xmd
3. Download the bitstream inside XMD:

   XMD% fpga -f download_ipi.bit
4. Connect to the processor inside XMD:

   XMD% connect mb mdm
5. Download the processor code (ELF) file:

   XMD% dow axi_vdma_display.elf
6. Run the software:

   XMD% run

Results from Running Hardware and Software

The Dell P2210T LCD monitor connected to the KC705 board displays a color bar pattern, and the HyperTerminal screen displays the output shown in Figure 6.

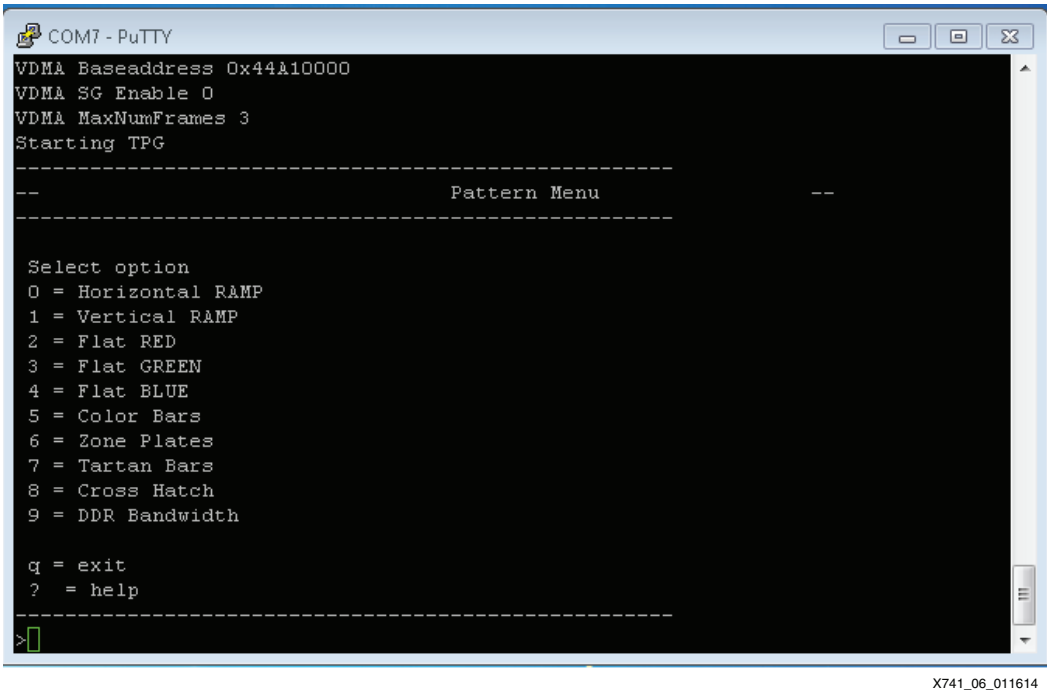

Figure 6: **HyperTerminal Output**

You can choose one of the eleven options displayed on the HyperTerminal screen:

0 = Color bars (layer 0)

1 = Zoneplate patterns (layer 1)

2 = Vertical ramp (layer 2)

3 = Horizontal ramp (layer 3)

4 = Tartan bars (layer 4)

5 = Flat red (layer 5)

6 = Flat green (layer 6)

7 = Flat blue (layer 7)

8 = Alpha blend of all layers simultaneously (layers 0–7)

9 = Real time system performance (one second of transfers)

Performance

The AXI_MM interconnect is 512 bits running at 200 MHz. The theoretical maximum bandwidth on each channel is 12.8 GB/s.

The DDR3 PHY is set for 64 bits with a memory clock frequency of 800 MHz. The theoretical throughput on DDR3 is 12.8 GB/s, which is the total bandwidth available in the design.

Using option 9 of the software application should show this output (the numbers can vary slightly from the values shown):

```
---------DDR3, AXI4 Slave Profile Summary........
Theoretical DDR Bandwidth                   = 12800 MB/sec
Practical DDR bandwidth                     = 9580 MB/sec
Percentage of DDDR Bandwidth consumed
   by eight Video Pipelines (Approx.)       = 74.85%
```

The total bandwidth is approximately 9,580 MB/s out of 12,800 MB/s, which is around 75% of the total theoretical bandwidth of the main memory.

Note: The numbers can vary slightly from the values shown.

Building Hardware

This section covers rebuilding the hardware design using the Vivado tools design flows. Before rebuilding the project, you must ensure that full system hardware evaluation or full licenses for AXI OSD and Chroma Resampler are installed. To obtain evaluation licenses for the AXI OSD or Chroma Resampler, see these websites:

- Xilinx Chroma Resampler LogiCORE IP product page [Ref 13] and *LogiCORE IP Chroma Resampler Product Guide for Vivado Design Suite* (PG012) [Ref 14]
- Xilinx Video On-Screen Display product page [Ref 15] and *LogiCORE IP Video On-Screen Display Product Guide* (PG010) [Ref 12]

Vivado Design Flow

To open and rebuild the design:

1. Install the Vivado Design Suite 2013.4 (requires Logic Edition at a minimum).
2. Unzip the reference design files accompanying this application note into a local folder (referred to as `<unzip dir>`).
3. Go to the `<unzip_dir>/kc705_video_8x_pipeline/HW/project_1` folder.
4. Open the Vivado tools in Windows by selecting **Start > Xilinx Design Tools > Vivado 2013.4 > Vivado** or enter the vivado command in Linux after setting up the Xilinx tools.

5. In the Vivado tools, select **Getting Started > Open Project** (Figure 7).

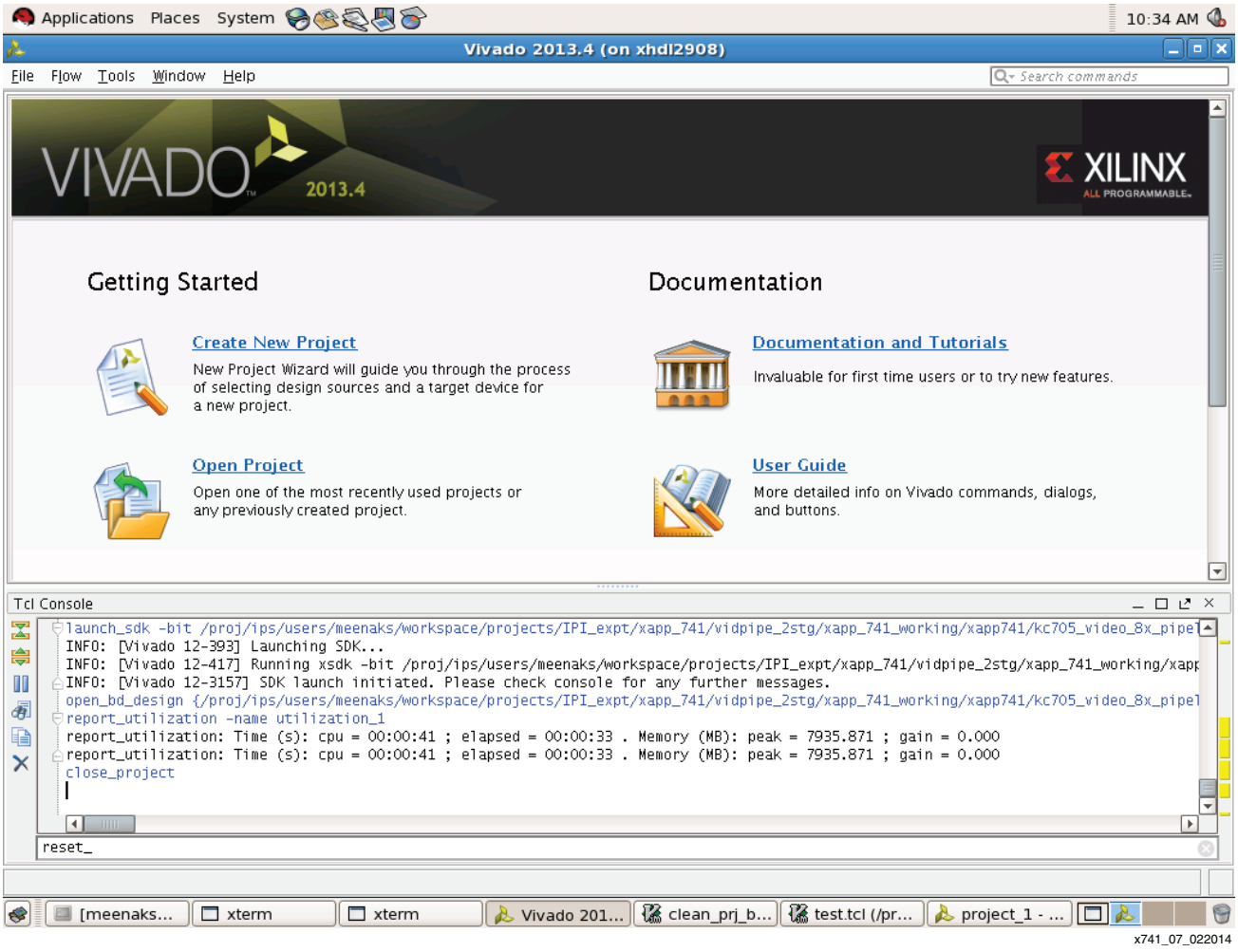

Figure 7: **Vivado Tools Getting Started Window (Open Project)**

6. Select `<unzip dir>/HW/project_1/project_1.xpr` and click **OK** (Figure 8).

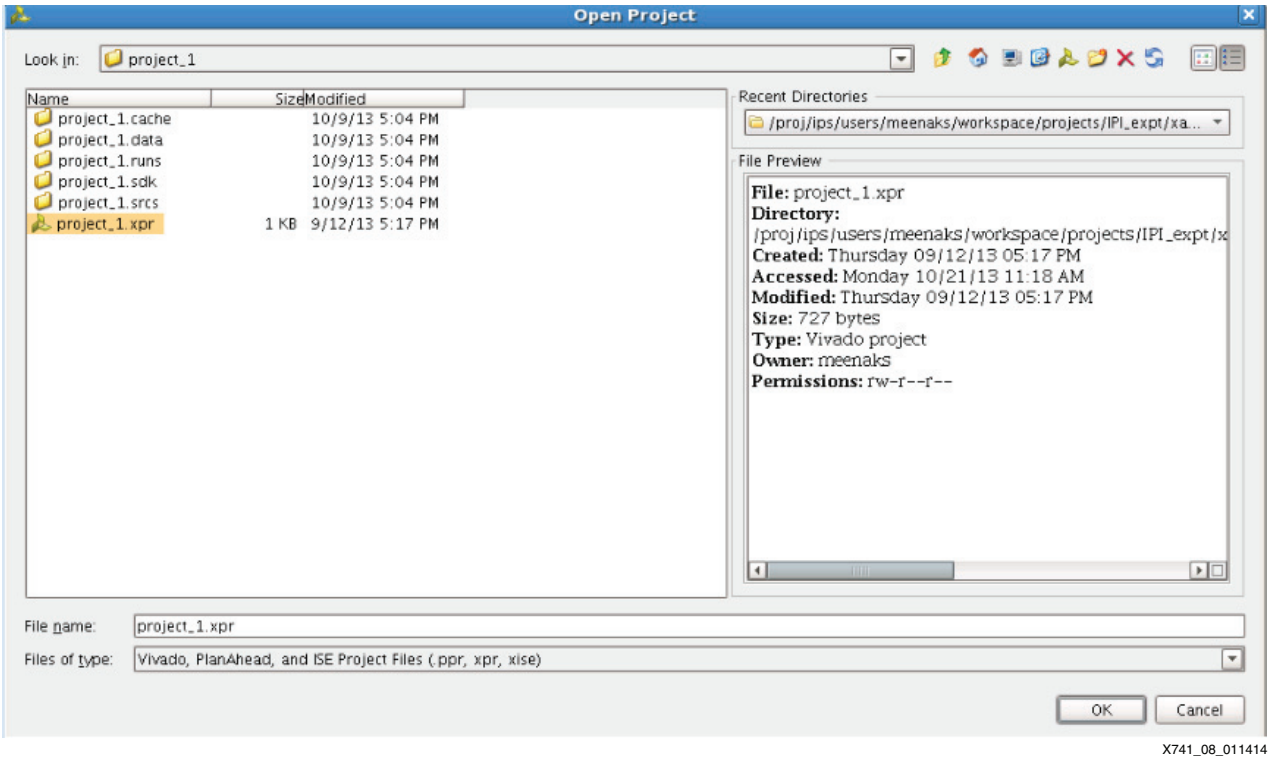

Figure 8: **Open Project Window**

7. Select **Flow > Generate Bitstream** (Figure 9) or press the **Generate Bitstream** icon next to **Program and Debug** in the Flow Navigator window pane (Figure 10).

   Note: If a dialog box is displayed asking to run Synthesis and Implementation, click **Yes**. The process of running synthesis and implementation on this design can take one hour or more to complete.

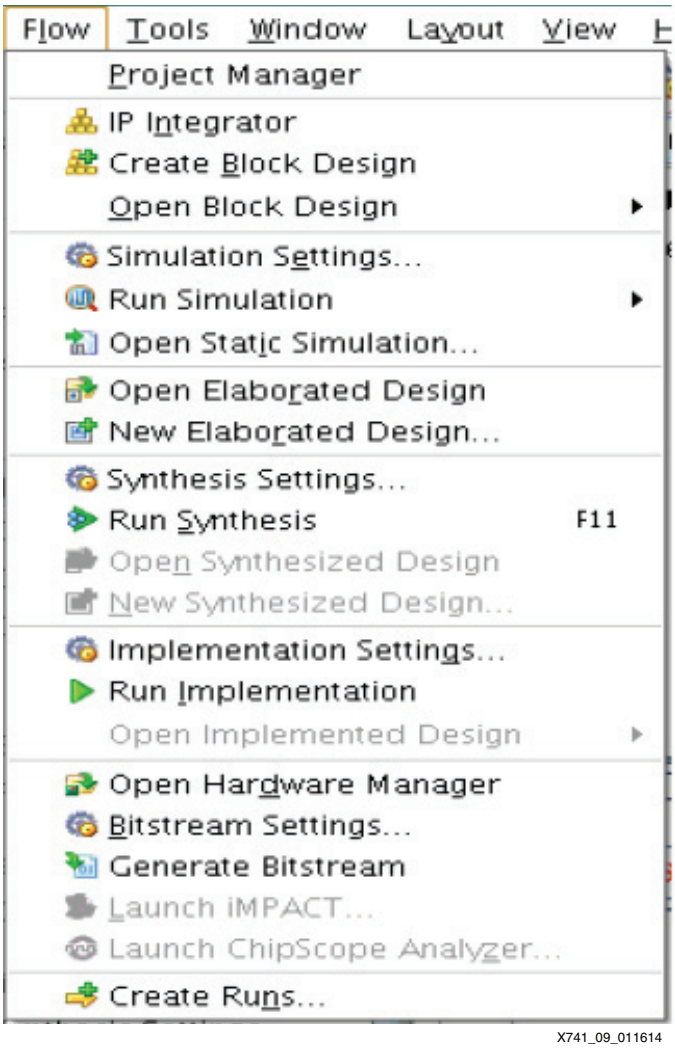

Figure 9: **Starting Bitstream Generation Using Menu Options**

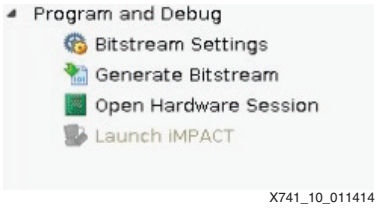

Figure 10: **Starting Bitstream Generation Using Icon**

8. Synthesis, implementation, and bitstream generation operations are performed on the design. When the “Bitstream Generation Completed” window appears, click **OK** (Figure 11).

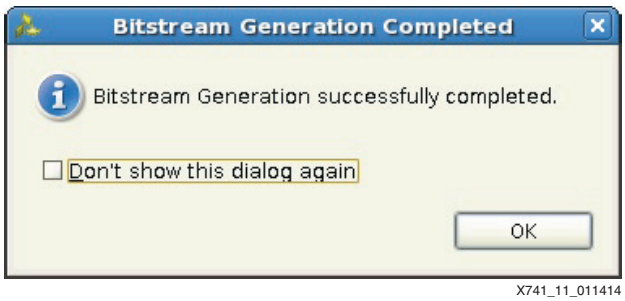

Figure 11: **Bitstream Generation Completed Dialog**

Compiling Software in SDK

1. Start SDK. In Linux, type **`xsdk`** to start SDK.
2. In the workspace launcher, select this workspace: `<unzip dir>/kc705_video_8x_pipeline/SW/SDK_Workspace`
3. Click **OK**.
4. Import the board support package (BSP), hardware platform, and software applications by selecting **File > Import > General > Existing Projects** into the workspace.
5. Click **Next**, then browse to `<unzip dir>/kc705_video_8x_pipeline/SW`.
6. Click **OK**.
7. Ensure that all checkboxes are selected (including **axi_vdma_display** and **hw_platform_0**).
8. Ensure that the associated software applications are selected.
9. Click **Finish**.

The BSP and software applications compile at this step. The process takes 2 to 5 minutes. You can now modify existing software applications and create new software applications in SDK.

Running the Hardware and Software through SDK

1. Select **Xilinx Tools > Program FPGA**.

   Note: Ensure bootloop is used for microblaze_0.
2. Click **Program**.
3. In the Project Explorer window, right click and select **axi_vdma_display > Run As > Launch on Hardware**.

Design Characteristics

The reference design is implemented in a Kintex-7 FPGA (XC7K325TFFG900-2) using the Vivado Design Suite 2013.4.

The resources used are:

- Total LUTs used: 96219 out of 203,800 (47%)
- Total I/Os used: 141 out of 500 (28%)
- Total internal memory used:
  - RAMB36E1s: 272 out of 445 (61%)
  - RAMB18E1s: 159 out of 890 (18%)

The resource utilization details of the design using the Vivado tools flows are shown in Table 4.

Note: Device resource utilization results depend on the implementation tool versions. Exact results can vary. These numbers should be used as a guideline.

Reference Design

The reference design has been fully verified and tested on hardware. The design includes details on the various functions of the different modules. The interface has been successfully placed and routed at 200 MHz on the main AXI interfaces to the memory controller using the Vivado Design Suite 2013.4.

The reference design files for this application note can be downloaded from:

https://secure.xilinx.com/webreg/clickthrough.do?cid=184421

The reference design matrix is shown in Table 2.

Table 2: **Reference Design Matrix**

Parameter	Description
General	
Developer name	Pankaj Kumbhare, Vamsi Krishna
Target devices (stepping level, ES, production, speed grades)	Kintex-7 FPGAs
Source code provided	Yes
Source code format	VHDL/Verilog (some sources encrypted)
Design uses code/IP from existing Xilinx application note/reference designs or third party.	Reference designs provided for Vivado IP 2013.4
Simulation	
Functional simulation performed	N/A
Timing simulation performed	N/A
Test bench used for functional and timing simulations	N/A
Test bench format	N/A
Simulator software/version used	N/A
SPICE/IBIS simulations	N/A
Implementation	
Synthesis software tools/version used	Vivado Design Suite 2013.4
Implementation software tools/versions used	Vivado Design Suite 2013.4
Static timing analysis performed	Yes (Timing passed with Vivado Design Suite implementation.)
Hardware Verification	
Hardware verified	Yes
Hardware platform used for verification	KC705 board

Utilization and Performance

Table 3 shows device utilization information.

Table 3: **Device Utilization Report on XC7K325T-FFG900-2**

Component	Vivado IPI Tools Flow
Slice Logic	
Slice Register	110947 (27%)

Table 3: **Device Utilization Report on XC7K325T-FFG900-2 *(Cont'd)***

Component	Vivado IPI Tools Flow
Occupied Slices	39806 (78%)
Slice LUTs	96219 (47%)
IOBs	
I/Os	141 (28%)
Memory	
RAMB36E1s	272 (61%)
RAMB18E1s	159 (18%)
Clocking	
BUFGCTRL	9 (28%)
MMCME2_ADV	2 (20%)
PLLE2_ADV	1 (10%)
General	
Run Time	117 minutes
Timing Violations	None

Device resource utilization is detailed in Table 4 for the IP cores shown in Figure 1. The information in Table 4 is taken from the utilization_1 tab in Vivado that opens when the **Implemented Design > report_utilization** command is executed. The utilization information is approximate due to cross-boundary logic optimizations and logic sharing between modules.

Table 5 summarizes the bandwidth calculations for the physical memory interface.

Table 4: **Module Level Resource Utilization**

IP Core	Instance Name	Slices	Slice Registers	LUTs	LUT RAM	Block RAM/ FIFO	DSP	MMCM ADV/ PLL	BUFG CTRL
MIG	mig_1	4,808	10,526	12,927	2,325	0	0	2	2
AXI Interconnect	axi4_lite	324	817	546	98	0	0	0	0
	axi_interconnect_0	295	137	569	0	0	0	0	0
	AXI_MM0	9,039	20,286	21,627	1,335	104	0	0	0
	MM1	3,559	8,542	8,273	722	41	0	0	0
	MM2	3,533	8,542	8,397	722	41	0	0	0
AXI VTC	v_tc_0	925	2,730	1,176	22	0	0	0	0
AXI TPG	v_tpg_0	606	1,757	1,181	21	1	3	0	0
	v_tpg_1	653	1,757	1,176	22	1	3	0	0
	v_tpg_2	667	1,757	1,175	22	1	3	0	0
	v_tpg_3	600	1,757	1,172	22	1	3	0	0
	v_tpg_4	614	1,757	1,173	22	1	3	0	0
	v_tpg_5	621	1,757	1,173	22	1	3	0	0
	v_tpg_6	606	1,757	1,175	22	1	3	0	0
	v_tpg_7	629	1,757	1,175	22	1	3	0	0

Table 4: **Module Level Resource Utilization *(Cont'd)***

IP Core	Instance Name	Slices	Slice Registers	LUTs	LUT RAM	Block RAM/ FIFO	DSP	MMCM ADV/ PLL	BUFG CTRL
AXI VDMA + AXI Traffic Generator	axi_vdma_0	1,195	3,840	2,952	192	13	0	0	0
	axi_vdma_1	1,380	3,840	2,589	192	13	0	0	0
	axi_vdma_2	1,309	3,840	2,592	192	13	0	0	0
	axi_vdma_3	1,271	3,840	2,592	192	13	0	0	0
	axi_vdma_4	1,293	3,840	2,590	193	13	0	0	0
	axi_vdma_5	1,282	3,840	2,592	192	13	0	0	0
	axi_vdma_6	1,267	3,840	2,592	192	13	0	0	0
	axi_vdma_7	1,253	3,840	2,587	192	13	0	0	0
	axi_traffic_gen_0	66	145	150	0	0	0	0	0
	axi_traffic_gen_1	70	147	148	0	0	0	0	0
	axi_traffic_gen_2	68	147	148	0	0	0	0	0
	axi_traffic_gen_3	72	149	148	0	0	0	0	0
OSD (Includes associated glue logic and display driver)	v_osd_0	2,685	8,331	5,033	332	0	24	0	0
	v_rgb2ycrcb_0	126	307	312	74	0	4	0	0
	v_cresample_0	128	332	327	64	0	0	0	0
	v_axi4s_vid_out_0	83	197	95	0	5	0	0	0
	hdmi_interface_0	19	38	0	0	0	0	0	0
Clock and reset logic	clk_wiz_1	0	0	0	0	0	0	1	5
	proc_sys_reset_1	15	37	20	1	0	0	0	0
MicroBlaze Processor Subsystem (Includes local memory and debug module for JTAG-based debug)	microblaze_0	756	1,545	1,632	192	20	0	0	0
	mdm_1	51	134	99	12	0	0	0	2
	lmb_v10_1	1	1	0	0	0	0	0	0
	lmb_v10_2	1	1	0	0	0	0	0	0
	dlmb_bram_if_ctrl	4	2	6	0	0	0	0	0
	ilmb_bram_if_ctrl	3	2	2	0	0	0	0	0
	microblaze_0_local_memory	63	6	144	0	32	0	0	0
	microblaze_0_axi_intc	72	177	193	32	0	0	0	0
AXI IIC	axi_iic_0	147	349	405	10	0	0	0	0
AXI UartLite	axi_uartlite_0	43	91	96	10	0	0	0	0
AXI Performance Monitor	axi_perf_mon_0	1,209	2,453	3,247	6	2	0	0	0
Total	39319	39,806	110,947	96,219	7,655	352	52	3	9

Table 5: **DDR3 Memory Physical Interface Maximum Theoretical Bandwidth**

Data Width	Data Rate	Maximum Theoretical Bandwidth
64 bits (SODIMM)	1600 Mb/s	12.8 GB/s (102.4 Gb/s)

Table 6 summarizes the total bandwidth of video data moved through memory.

Table 6: **Average Bandwidth Used for Video Traffic**

Frame Resolution	Refresh Rate (Hz)	Bits Per Pixel	Number of Video Streams	Total Video Bandwidth
1920 x 1080	60	24	16 (VDMA)+8(ATG)	9.355 GB/s (74.84 Gb/s)

Table 7 summarizes the percentage of the maximum theoretical bandwidth used by the video streams.

Table 7: **Percentage of the Maximum Theoretical Bandwidth Used**

Total Aggregate Video Bandwidth	Frame Rate (Hz)	Maximum Theoretical Bandwidth	Percentage of the Maximum Theoretical Bandwidth Used(1)
9.355 GB/s (74.84 Gb/s)	60	12.8 GB/s (102.4 Gb/s)	74.85

Notes:

1. Since VDMA is in free running mode, the write throughout is greater than read throughput. So the actual throughput achieved is slightly greater than the Theoretical BW

Conclusion

This application note describes a video system using an AXI Interconnect core configured to operate at a bandwidth of approximately 10 GB/s. Eight video pipelines, each processing high-definition video streams of 1920 x 1080 pixels at 60 frames/sec are connected to the DDR memory through the AXI Interconnect. To meet high-performance design requirements, the DDR3 controller (DDR memory with an 800 MHz clock and 64-bit data width) is configured to utilize approximately eighty percent of its available bandwidth.

References

This application note uses the following references:

1. *KC705 Evaluation Board for the Kintex-7 FPGA User Guide* (UG810)
2. AMBA AXI4 specifications
3. *AXI Reference Guide* (UG761)
4. *Vivado Design Suite Tutorial: Designing IP Subsystems Using IP Integrator* (UG995)
5. *Vivado Design Suite User Guide: Designing IP Subsystems Using IP Integrator* (UG994)
6. *LogiCORE IP AXI Interconnect Product Guide for Vivado Design Suite* (PG059)
7. *LogiCORE IP AXI Video Direct Memory Access Product Guide for Vivado Design Suite* (PG020)
8. *Vivado Design Suite User Guide: Embedded Processor Hardware Design* (UG898)
9. *Zynq-7000 SoC and 7 Series Devices Memory Interface Solutions Data Sheet* (DS176)
10. *LogiCORE IP Video Timing Controller Product Guide for Vivado Design Suite* (PG016)
11. *LogiCORE IP AXI4-Stream to Video Out Product Guide for Vivado Design Suite* (PG044)
12. *LogiCORE IP Video On-Screen Display Product Guide for Vivado Design Suite* (PG010)
13. Xilinx Chroma Resampler LogiCORE IP
14. *LogiCORE IP Chroma Resampler Product Guide for Vivado Design Suite* (PG012)
15. Xilinx Video On-Screen Display LogiCORE IP
16. Vivado Design Suite 2013.4 documentation
17. *Kintex-7 FPGA KC705 Evaluation Kit Getting Started Guide* (UG883)
18. Xilinx Video Timing Controller LogiCORE IP
19. *LogiCORE IP AXI Performance Monitor Product Guide for Vivado Design Suite* (PG037)

20. *LogiCORE IP AXI Traffic Generator Product Guide for Vivado Design Suite* (PG125)
21. *LogiCORE IP RGB to YCrCb Color Space Converter Product Guide for Vivado Design Suite* (PG013)

Revision History

The following table shows the revision history for this document.

Date	Version	Description of Revisions
03/23/2012	1.0	Initial Xilinx release.
04/30/2012	1.1	Updated Hardware Requirements. Added AXI VTC block to Figure 1. In Video-Related IP, changed 7.96 Gb/s to 79.6 Gb/s. Replaced references to LogiCORE IP AXI Video Direct Memory Access (axi_vdma) Product Specification (v3.01.a) with LogiCORE IP AXI Video Direct Memory Access v5.00.a Product Guide in AXI Interconnects, AXI VDMA Instances, and References. In AXI Interconnect (AXI_MM Instance), replaced the word “rank” with “bank.” Updated Executing the Reference Design in Hardware. Updated RAMB36E1s and RAMB18E1s in Design Characteristics.
12/17/2012	1.2	Updated software version from 13.4 to 14.3 and added Vivado tools throughout. Revised Figure 1. In AXI VDMA Instances, removed procedure to follow when C_PRMRY_IS_ACLK_ASYNC is 1. Added Datapath Optimization and Streaming to Video Bridge. Updated step 3 of Executing the Reference System Using the Pre-Built Bitstream and the Compiled Software Application. In Performance, updated bandwidth percentage from 77% to 78%. Added *XPS Design Flow* and Vivado Design Flow. Updated Design Characteristics and these tables: Device Utilization Report on XC7K325T-FFG900-1 and Module Level Resource Utilization.
04/14/2014	1.3	Updated for Vivado Design Suite 2013.4. Revised USB cable types in Hardware Requirements. Removed AXI2AXI connector, AXI_7SERIES_DDRX, and csc_rgb_to_ycrcb422 cores and added MIG, RGB2YCrCB Converter, Chroma Resampler, and AXI Traffic Generator to Reference Design Specifics. Updated entire Table 1 Reference System Address Map. Kintex-7 device XC7K325T-FFG900-1 was changed to XC7K325T-FFG900-2. Updated Video-Related IP. Removed Configuration 2 figure and table. Clarified VDMA information in the AXI VDMA Instances section. The Memory Interface Generator (MIG) section was added and speed grade changed from -1 to -2 in that section. The heading *AXI VTC* was changed to Video Timing Controller. The heading *AXI TPG* was changed to Test Pattern Generator. The heading AXI OSD was changed to Video On-Screen Display. Instructions in Executing the Reference System Using the Pre-Built Bitstream and the Compiled Software Application changed. Option **d** is no longer available in Results from Running Hardware and Software. The XPS Design Flow section was removed from Building Hardware. The Compiling Software in SDK procedure was updated. Resource usage numbers were updated in Design Characteristics. Updated the reference design files and data in these tables in the Reference Design section: Reference Design Matrix, Device Utilization Report on XC7K325T-FFG900-2, Module Level Resource Utilization, Average Bandwidth Used for Video Traffic, and Percentage of the Maximum Theoretical Bandwidth Used. Updated References.

DISCLAIMER