XAPP742 (v1.2) February 26, 2014

AXI VDMA Reference Design

Authors: Pankaj Kumbhare and Vamsi Krishna

Summary

This application note demonstrates the creation of video systems by using Xilinx native video IP cores such as AXI Video Direct Memory Access (VDMA), Video Timing Controller (VTC), test pattern generator (TPG), and the DDR3 memory controller to process configurable frame rates and resolutions in Kintex®-7 FPGAs. The reference design focuses on run time configuration of an onboard clock generator for a video pixel clock and video IP cores for running selected combinations of video resolution and frame rate.

The system also displays system-level bandwidth utilization for each combination of frame rate and resolution, which are common metrics used by system designers. This application note discusses the configuration of each video IP in detail, helping designers make effective use of video IP cores for processing various video features.

The video datapath of the current design includes video TPG, VDMA (streaming to memory-mapped), DDR, VDMA (memory-mapped to streaming), and video onscreen display (OSD) IP blocks. Each of these video IP blocks are configured dynamically to process various combinations of frame rate and resolution. An onboard configurable clock generator (SI570) is used to generate the video clock for the desired refresh rate and resolution. The VDMA is driven from a TPG which has an AXI4-Stream Interface to VDMA. The VDMA core operates in free running mode. Data read by the AXI VDMA is sent to the OSD. The output of the OSD core drives an onboard High Definition Media Interface (HDMI™) video display through the color space converter with a VTC block to generate the necessary timing signals.

A performance monitor block is added to capture the performance of DDR memory. Video frame data moved by the AXI VDMA blocks are buffered through a shared DDR3 SDRAM and are controlled by a MicroBlaze™ processor.

The reference design is targeted for the Kintex-7 FPGA XC7K325TFFG900-2 on the Xilinx KC705 evaluation board (Rev 1.1).

Included Systems

The reference design is created and built using Vivado® IP Integrator 2013.4, which is part of Vivado® System Edition (SE). The IP Integrator (IPI) is an interactive design and verification environment enabling creation and verification of a hierarchical system by graphically connecting IP provided by Xilinx, third parties, or proprietary IP using interface level connections. It 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 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 tools and SDK project files are provided with the reference design to allow you to examine and rebuild the design or use it as a template for starting a new design.

Introduction

Xilinx IP cores implement various functions for many video applications. The use of AXI interconnect, the memory interface generator (MIG) tool, VDMA, onboard configurable clock generator, VTC, and OSD IP blocks can form the core of video systems capable of handling the various combinations of frame rates and resolution. 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 1]. These interfaces provide a common IP interface protocol framework around which to build the design.

AXI VDMA implements a high-performance, video-optimized DMA engine with frame buffering, and two-dimensional (2D) DMA features. Together, the AXI interconnect and AXI MIG implement a multi-port memory controller (MPMC) for sharing data from multiple sources through a common memory device, typically DDR3 SDRAM. The AXI VDMA transfers the buffered video data streams to and 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. An onboard SI570 clock generator is used dynamically to change the video pixel clock frequency at run time for varying video frame rates. High-level control of the system containing I/O peripherals and processor support IP is provided by an embedded MicroBlaze processor. To optimize the system to balance performance and area, multiple AXI interconnect blocks are used to implement segmented or cascaded AXI interconnect networks with each AXI interconnect block individually tuned and optimized.

Hardware and Software Requirements

The hardware requirements for this reference system are:

- Xilinx KC705 evaluation board (Rev 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 cable (colors are not displayed properly otherwise)
- Display monitors supporting configurable resolutions (up to 75 frames/sec, tested on a Dell P2210T monitor)

The installed tool requirements for building and downloading this reference system are:

- Vivado Design Suite System Edition 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
- Memory Interface Generator (MIG)
- Video Timing Controller
- AXI_TPG
- AXI_VDMA
- AXI_PERFOMANCE_MONITOR

- AXI_OSD
- AXI4-Stream to Video Out
- HDMI_Interface IP cores
- RGB2YCRCB color space converter
- Chroma resampler

An onboard configurable clock generator (SI570) is used to supply the video clock to the design.

Figure 1 shows a block diagram of the reference system.

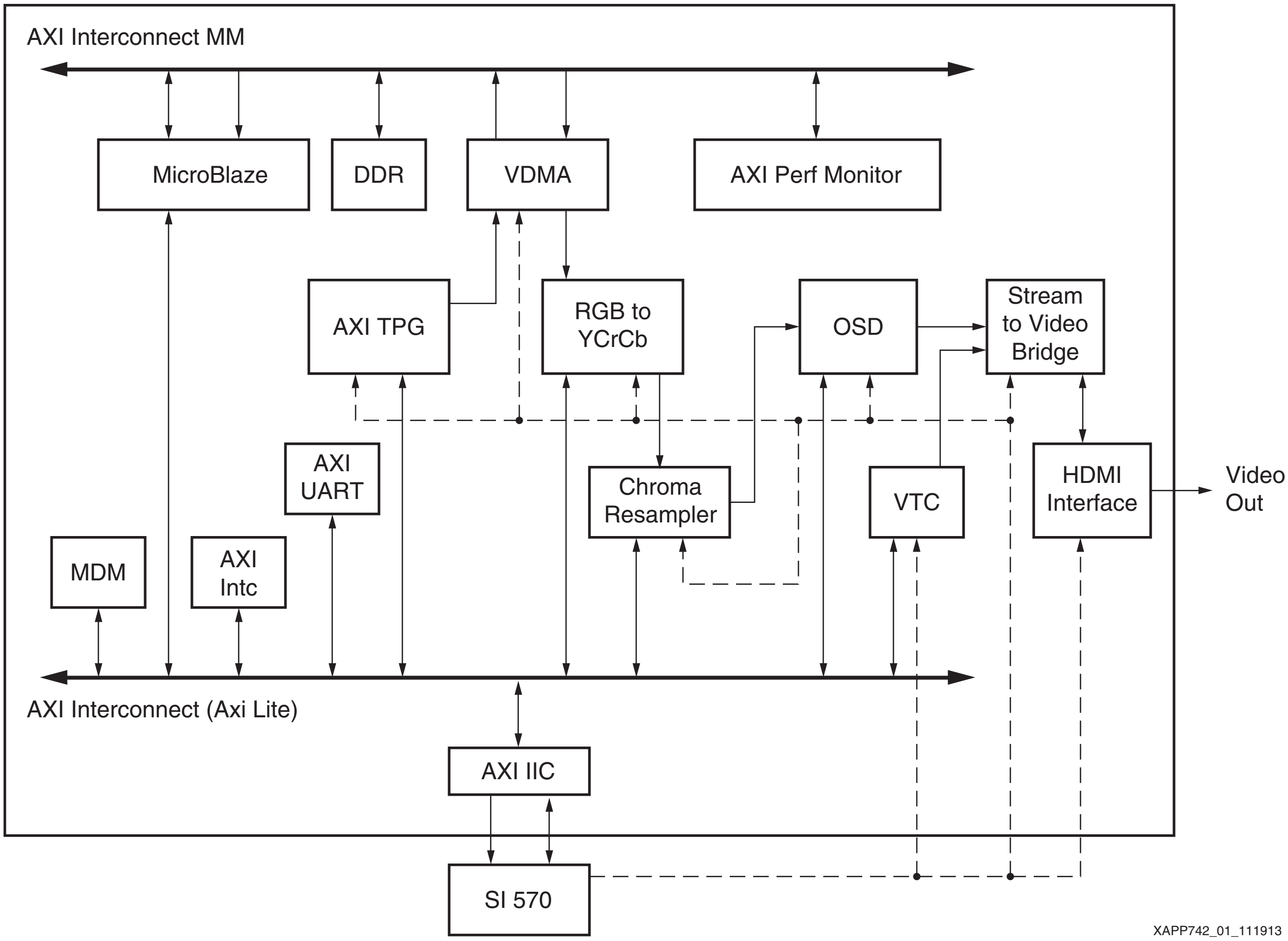


Figure 1: **Reference System Block Diagram**

Table 1 shows the address map of the reference system.

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 UART Lite	axi_uartlite_1	0x40600000	0x4060FFFF
MIG	mig_1	0x80000000	0xBFFFFFFF
Video Timing Controller	v_tc_1	0x44A30000	0x44A3FFFF
AXI TPG	v_tpg_1	0x44A00000	0x44AFFFF
AXI performance monitor	axi_perf_mon_0	0x44A50000	0x44A5FFFF
AXI onscreen display	v_osd_1	0x44A20000	0x44A2FFFF
AXI VDMA	axi_vdma_1	0x44A10000	0x44A1FFFF
Chroma resampler	v_cresample_0	0x44A60000	0x44A6FFFF
Color-Space converter	v_rgb2ycrcb_0	0x44A40000	0x44A4FFFF

Table 2 describes the detailed configurable frame rates and resolutions.

Table 2: **Configuration Frame Rates and Resolution**

Resolution	Horizontal Active Video	Vertical Active Video	Video Clock Frequency (MHz)
1	640	480	25.2
2	720	480	27.2
3	800	600	39.7
4	1024	768	64.9
5	1280	720	74.2
6	1280	1024	108.2
7	1920	1080	148.5
8	1600	1200	162

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 can be applied to a wide range of embedded systems.

This application note assumes you have some general knowledge of IP Integrator feature. See *Designing IP Subsystems Using IP Integrator* (UG995) [Ref 2] for more information about the Vivado IP Integrator feature.

Video-Related IP

The reference design implements a video system that can run the desired video resolutions and frame rates. Each picture consists of three bytes per pixel. The video system and its IP cores run with a total aggregate read/write bandwidth as described in Table 3. The video traffic of the system is generated by the TPG block and displayed by the OSD core.

Table 3: **Frame Resolution Versus Bandwidth of Video Streams**

Resolution	Horizontal Active Video	Vertical Active Video	Frame Rate	Bandwidth/ One Stream (MB)[1]	Bandwidth/ Two Streams (MB)[2]
1	640	480	60	55.29	110.59
			75	69.12	138.24
2	720	480	60	62.21	124.42
			75	77.76	155.52
3	800	600	60	86.4	172.80
			75	108	216.0
4	1024	768	60	141.56	283.12
			75	176.945	353.89
5	1280	720	60	165.89	331.78
			75	207.36	414.72
6	1280	1024	60	235.93	471.86
			75	294.91	589.82
7	1920	1080	60	373.25	746.5
			75	466.56	933.12
8	1600	1200	60	345.6	691.20
			75	432	864.0

Notes:

1. The bandwidth for one stream is calculated as Vertical Active Video x Horizontal Active Video x Bytes per Pixel x Frame Rate.
2. Since VDMA is running in Free Running mode, the write throughput is higher than read throughput. So, the aggregate bandwidth is greater than or equal to that mentioned under Bandwidth/Two Streams column.

This application note demonstrates a video system using two high-definition video streams of configurable resolutions and frame rates. Video systems typically include a source, some internal processing, and a destination. There can be multiple stages internally using many 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 and out of memory according to the requirements of the internal processing stages. In this application note, a TPG creates the internal IP block memory traffic to simulate typical conditions.

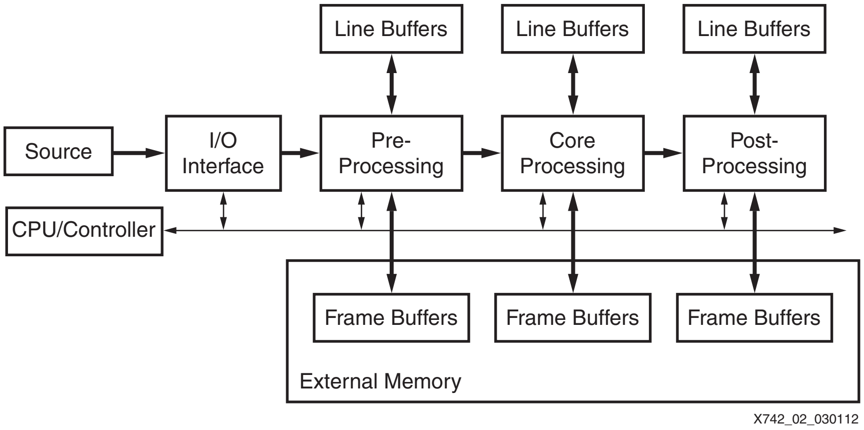


Figure 2: **Typical Video System**

Onboard Configurable Clock Generator

In the reference design, clocks for the processor, DDR memory, and other slaves of the processor are derived from the internal mixed-mode clock manager (MMCM) of the FPGA. After the design is implemented, these frequencies cannot be changed at run time. However, to change dynamically the resolution and frame rates of the system, the video pixel clock must be configurable at run time. The onboard configurable clock generator (SI570), which is connected through the IIC bus, is used to generate the video pixel clock at run time. After selecting an option through HyperTerminal, the SI570 is configured to generate the desired video pixel clock at run time.

AXI Interconnect

This design contains two AXI interconnects, each targeted to balance throughput, area, and timing considerations (see *LogiCORE IP AXI Interconnect Product Guide* (PG059) [Ref 3]). The AXI_MM instance is used for high-speed masters and slaves that include high throughput and high F_{MAX} optimizations. The AXI_Lite is generally optimized for area and is used for the processor to access slave registers of various IP cores.

AXI Interconnect can be configured for crossbar width and clock frequency. In addition, it also has parameters for enabling FIFO buffering to improve throughput and register slicing to improve timing.

For this design, the interconnect is configured for maximum performance by setting crossbar width of 512 and clock frequency of 200 Mz. Also fifo buffering and register slicing is enabled to improve throughput and performance.

AXI VDMA

The AXI VDMA core is designed to provide video read/write transfer capabilities from the AXI4 memory-mapped 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. The AXI VDMA core incorporates video-specific functionality, for example, Gen-Lock and Frame Sync, for fully synchronized frame DMA operations and 2D DMA transfers. In addition to synchronization, frame store numbers and register direct or scatter gather mode operations are available for ease of control by the central processor.

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

This design uses a single instance of AXI VDMA with three frame buffers using the AXI4 MM2S, AXI4 S2MM, AXI4-Stream MM2S, and AXI4-Stream S2MM interfaces.

AXI VDMA can be configured to process multiple resolutions through specifying horizontal and vertical resolutions in its register space. The 24-bit wide MM2S and S2MM interfaces from the AXI VDMA instance are connected to the AXI_MM instance of the AXI interconnect. The masters run off the video clock (onboard clock generator) which requires asynchronous clock converters to the 200 MHz AXI interconnect core frequency. Upsizers in the AXI interconnect are used to convert 24-bit transactions from the AXI VDMA to the transaction width of the AXI interconnect core.

In addition, line buffers inside the AXI VDMA for the read and write sides are set to 1K deep. The pixel data from the streaming interface is temporarily stored in line buffers before it is supplied on the MM interface. This allows the VDMA MM interface to handle width conversion without any loss of data. See *LogiCORE IP AXI Video Direct Memory Access (axi_vdma) Product Guide* (PG020) [Ref 4] for more information.

MicroBlaze Processor ICache and DCache

The MicroBlaze processor ICache and DCache masters run at 100 MHz because the MicroBlaze processor runs an application from main memory that sets up and monitors the video datapath. 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. Setting the MicroBlaze processor to run at 100 MHz synchronous improves design timing closure and reduces area.

Memory Interface Generator (MIG)

The single slave connected to the interconnect is the MIG. The memory controller AXI Interface is 256 bits wide, running at 200 MHz, and disables narrow burst support for optimal throughput and timing. See the *7 Series FPGAs Memory Interface Solutions* (DS176) [Ref 5] for more information about the memory controller.

AXI Interconnect (AXI_Lite)

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.

A single AXI4-Lite interconnect is sufficient for one video pipeline design. It is possible to extend the number of interconnects by cascading interconnects if there are more than 16 AXI4-Lite slaves.

The AXI_Lite interconnect block is 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, this interconnect is 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.

AXI_Lite Interconnect

The slaves on the AXI_Lite interconnect are for MDM, AXI_UARTLITE, AXI_IIC, AXI_INTC, TPG, VTC, RGB2YCRCB, CRESAMPLE, AXI_PERFORMANCE_MONITOR, AXI OSD, and AXI_VDMA.

Video Timing Controller

The VTC LogiCORE™ IP 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 and includes support for programmable pulse polarity.

The VTC contains an AXI4-Lite interface to access slave registers from a processor. For more information about the VTC, see the *LogiCORE IP Video Timing Controller Product Guide* (PG016) [Ref 6]. The VTC instance is used for the AXI OSD, which is the output portion of the video pipeline.

AXI TPG

The 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 TPG. The 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 TPG is used as a replacement to a video source because only the amount of traffic generated to demonstrate the performance of the system is of interest. The control software demonstrates generation of nine possible video patterns, such as color bars, horizontal and vertical burst patterns, flat colors, and zone plates. No matter which test pattern is chosen, the amount of data generated is the same for a particular resolution and frame rate.

Several operating modes are accessible through software control. In this application note, the TPG always generates one of nine possible test patterns through user input. 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 to Video Out IP core (see *LogiCORE IP AXI4-Stream to Video Out Product Guide* (PG044) [Ref 7]) is used to convert data from the AXI4-Stream video protocol interface to the video domain interface. This core works in conjunction with the timing generator portion of the VTC core.

This core provides a bridge between the AXI4-Stream video input from 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 streaming video signal from the OSD core and timing signals from the VTC core to generate Video out signals to the HDMI Interface.

RGB2YCrCb 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 HDMI interface.

AXI OSD

The Video On-Screen Display LogiCORE IP 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 other image formats in 8, 10, or 12 bits per color component. In this application note, the OSD is configured to display one video layer.

The AXI OSD contains an AXI4-Lite interface allowing access to core configuration registers from a processor. For more information about the AXI OSD, see the *LogiCORE IP Video On-Screen Display Product Guide* (PG010) [Ref 8].

Chroma Resampler

This IP core is used to convert between different video formats YUV422, YUV444, YUV420, and so on. 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 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. DDR slave interconnect (AXI_MM) is connected to one of the four slots of the monitor. In addition to the signals of the DDR slave interconnect and AXI4-Lite bus interface, the core clock (the higher of two bus interface clock frequencies) must also be connected.

The core can also measure various 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 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 AXI performance monitor core is connected to the DDR3 memory controller to measure the throughput of the core. Valid, ready, strobe, and other AXI signals of the DDR3 slave are used to enable various counters for measuring events on the bus.

Software Applications

The software application starts up the video pipeline allowing you to change resolution and frame rate and examine bandwidth in real time.

Application-level software for controlling the system is written in C using the provided drivers for each IP. The programmer’s model for each IP describes the particular application program interface (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 system, caches, UART, and VDMA are initialized.
2. The HDMI port is initialized.
3. The onboard SI570 clock generator is configured through the IIC interface.
4. The onboard SI570 clock generator is configured to generate the video clock for the desired frame rate and resolution.
5. The TPG instances are set to write a default color bar pattern that does not start until the VTC instances are started. TPG and VTC instances are configured to generate the desired frame resolutions.
6. The VTC, Chroma resampler, and RGB2YCrCb instances are started for the desired resolution.
7. The AXI OSD is configured for the desired resolution.
8. The AXI VDMA instance is started by configuring VDMA read and write channels to begin the transfers for the VDMA instances.
9. The TPG instance in the design is configured to write one of the 9 possible test patterns:
   - Vertical ramp
   - Horizontal ramp
   - Flat red
   - Flat green
   - Flat blue

- Color bars
- Zone plates
- Tartan bars
- Cross hatch

After the initial setup sequence, you can enter an option through HyperTerminal to select one of the combinations of resolution and frame rate (**0–7** for resolutions, **a** or **b** for frame rates). Because OSD is configured for a single channel, its input data is directly transferred to its output. Video patterns output from the OSD are routed to the HDMI port through a color space (RGB to YCrCb) converter. For multiple video pipelines, OSD registers can also be configured to blend input channels to the required level, and different values are given to the alpha blending register for each layer to show all layers on the LCD screen at the same time.

After configuring the video pipeline for the desired frame rate and resolution, you can select one of the 9 possible video patterns by entering a number from **0** through **8**. Option **9** sets up the performance monitoring IP instance to measure the performance of the DDR memory.

Executing the Reference Design on Hardware

This section provides instructions to execute the reference design in hardware. The reference design runs on the KC705 board shown in Figure 3. In these instructions, numbers in parentheses correspond to the callout numbers in Figure 3. Not all callout numbers are referenced.

1. Connect a USB cable from the host PC to the USB JTAG port (item (6) in Figure 3). Ensure that the appropriate device drivers are installed (see the *Kintex-7 FPGA KC705 Evaluation Kit Getting Started Guide* (UG883) [Ref 9] for additional information).
2. Connect a second USB cable from the host PC to the USB UART port (6). Ensure that the USB-UART drivers described in Hardware and Software Requirements, page 2 have been installed.
3. Connect the KC705 HDMI™ connector (18) to a video monitor. To display the higher frame rate configurations, the monitor should be capable of displaying frame rates up to 120 Hz video signal.
4. Connect a power supply cable.
5. Turn the 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

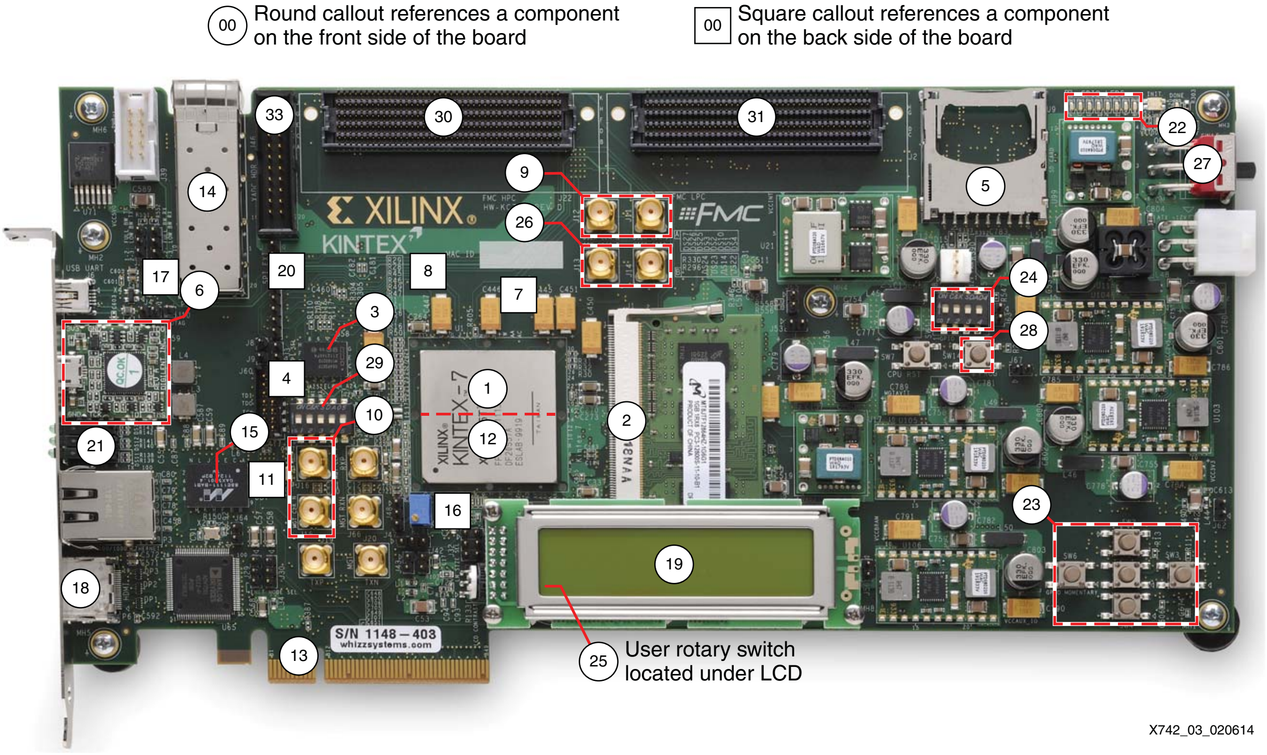

00 Round callout references a component on the front side of the board
00 Square callout references a component on the back side of the board
33
30
31
22
27
9
14
5
26
17
6
20
8
7
24
28
3
29
4
1
10
2
15
12
21
11
16
23
19
18
13
25 User rotary switch located under LCD
X742_03_020614

Figure 3: **KC705 Board**

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>/vdma_ref_design/` directory:

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

   ```
   % cd <unzip dir>/vdma_ref_design/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:

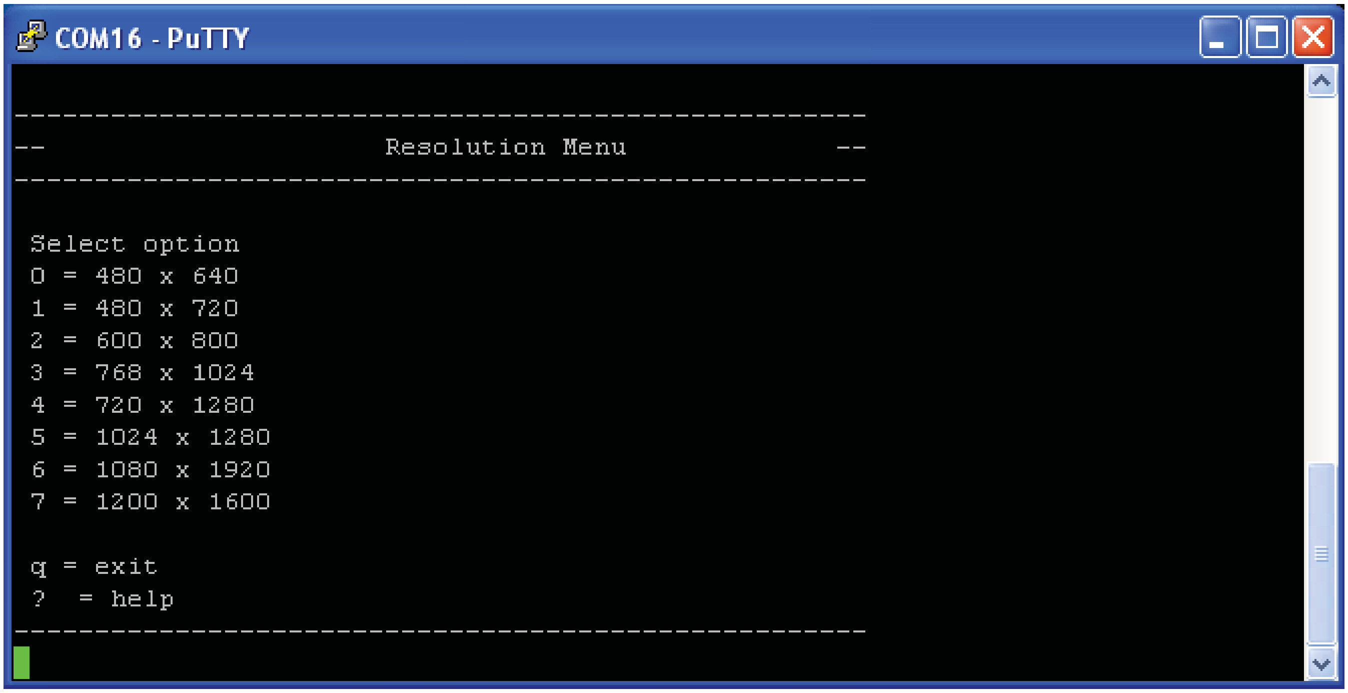
   ```
   XMD% dow vdma_display.elf
   ```

6. Run the application:

   ```
   XMD% run
   ```

Results from Running Hardware and Software

The LCD monitor connected to the KC705 board displays a color bar pattern, and the HyperTerminal screen displays the output shown in Figure 4.

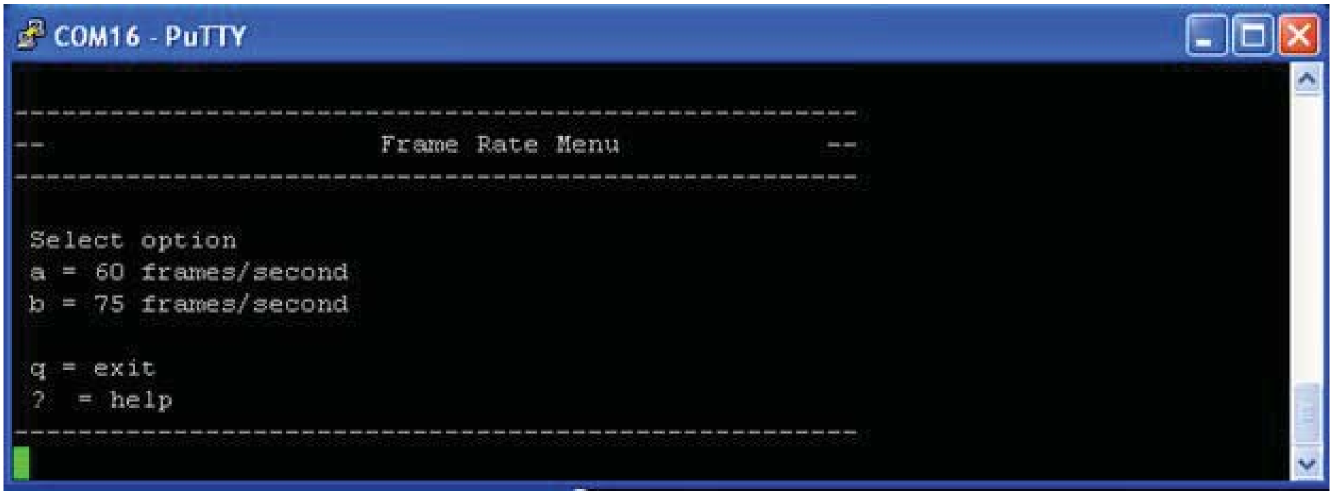


```
-------------------------------------------------------
--                    Resolution Menu                --
-------------------------------------------------------

 Select option
 0 = 480 x 640
 1 = 480 x 720
 2 = 600 x 800
 3 = 768 x 1024
 4 = 720 x 1280
 5 = 1024 x 1280
 6 = 1080 x 1920
 7 = 1200 x 1600

 q = exit
 ?  = help
-------------------------------------------------------
```

X742_04_020614

Figure 4: **HyperTerminal Menu for Selecting Resolution**

You can select between the 60 and 75 frames/sec options, as shown in Figure 5.



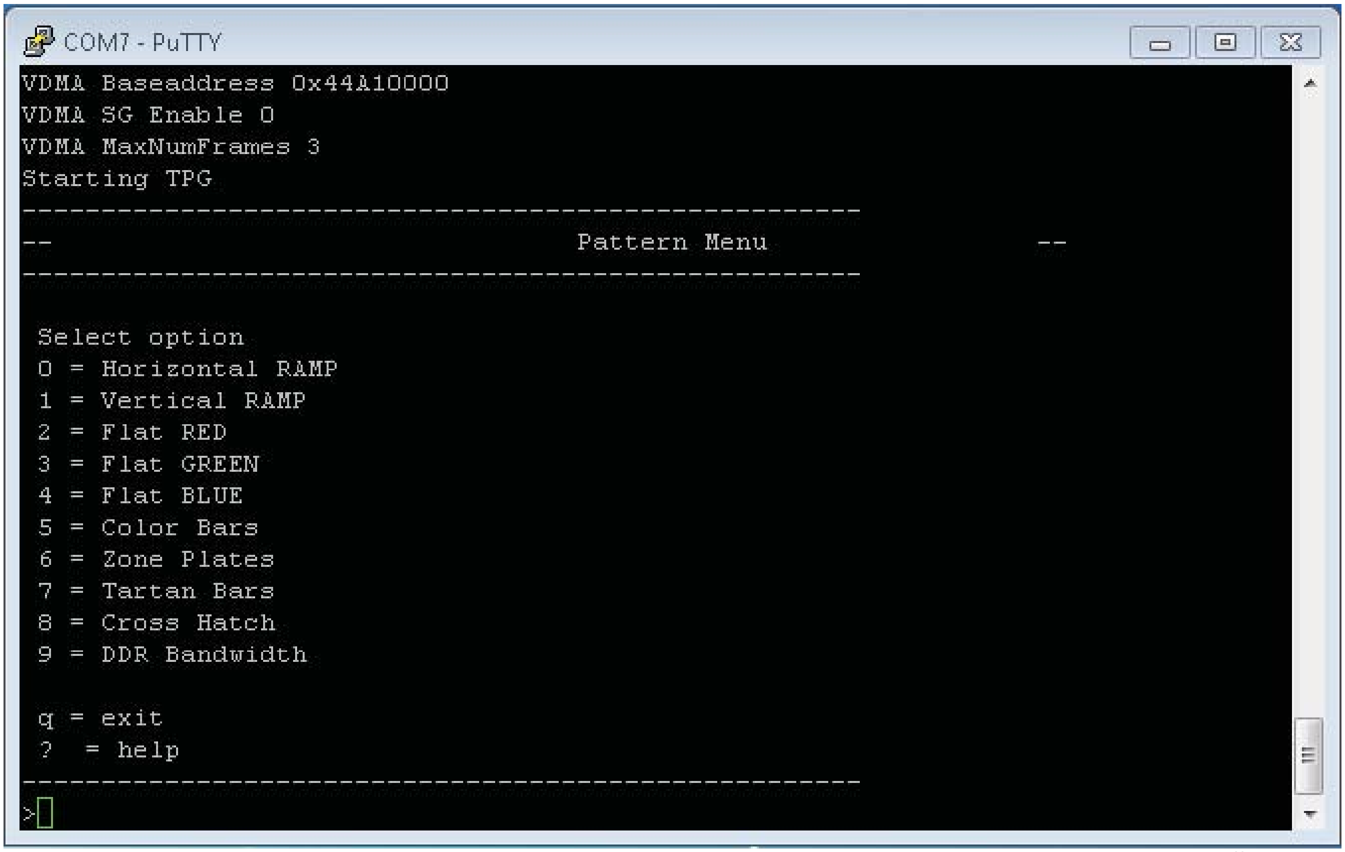
```
-------------------------------------------------------
--                    Frame Rate Menu                --
-------------------------------------------------------

 Select option
 a = 60 frames/second
 b = 75 frames/second

 q = exit
 ?  = help
-------------------------------------------------------
```

X742_05_020614

Figure 5: **HyperTerminal Menu for Selecting Frame Rate**

You can choose from nine pattern options displayed on the HyperTerminal screen, as shown in Figure 6:

0 = Horizontal ramp
1 = Vertical ramp
2 = Flat red
3 = Flat green
4 = Flat blue
5 = Color bars
6 = Zone plates
7 = Tartan bars
8 = Cross hatch



```
VDMA Baseaddress 0x44A10000
VDMA SG Enable 0
VDMA MaxNumFrames 3
Starting TPG
-----------------------------------------------------
--                                 Pattern Menu                 --
-----------------------------------------------------

 Select option
 0 = Horizontal RAMP
 1 = Vertical RAMP
 2 = Flat RED
 3 = Flat GREEN
 4 = Flat BLUE
 5 = Color Bars
 6 = Zone Plates
 7 = Tartan Bars
 8 = Cross Hatch
 9 = DDR Bandwidth

 q = exit
 ?  = help
-----------------------------------------------------
>
```

X742_06_020614

Figure 6: **HyperTerminal Menu for Selecting Video Pattern**

Performance

The DDR3 PHY is set for 64 bits with a memory clock frequency of 400 MHz. The theoretical throughput on the DDR3 memory is 6.4 GB/s, which is the total bandwidth available in the design.

Selecting option **9** causes the application to display this output (the numbers might vary slightly from the values shown):

```
---------DDR3, AXI4 Slave Profile Summary........

Theoretical DDR Bandwidth = 6400000000 bytes/sec

Practical DDR Bandwidth = 829192864 bytes/sec

Percentage of DDR Bandwidth consumed by frame of resolution
(1920x1080 @ 60 Hz) (approx.) = 12.72
```

Building Hardware

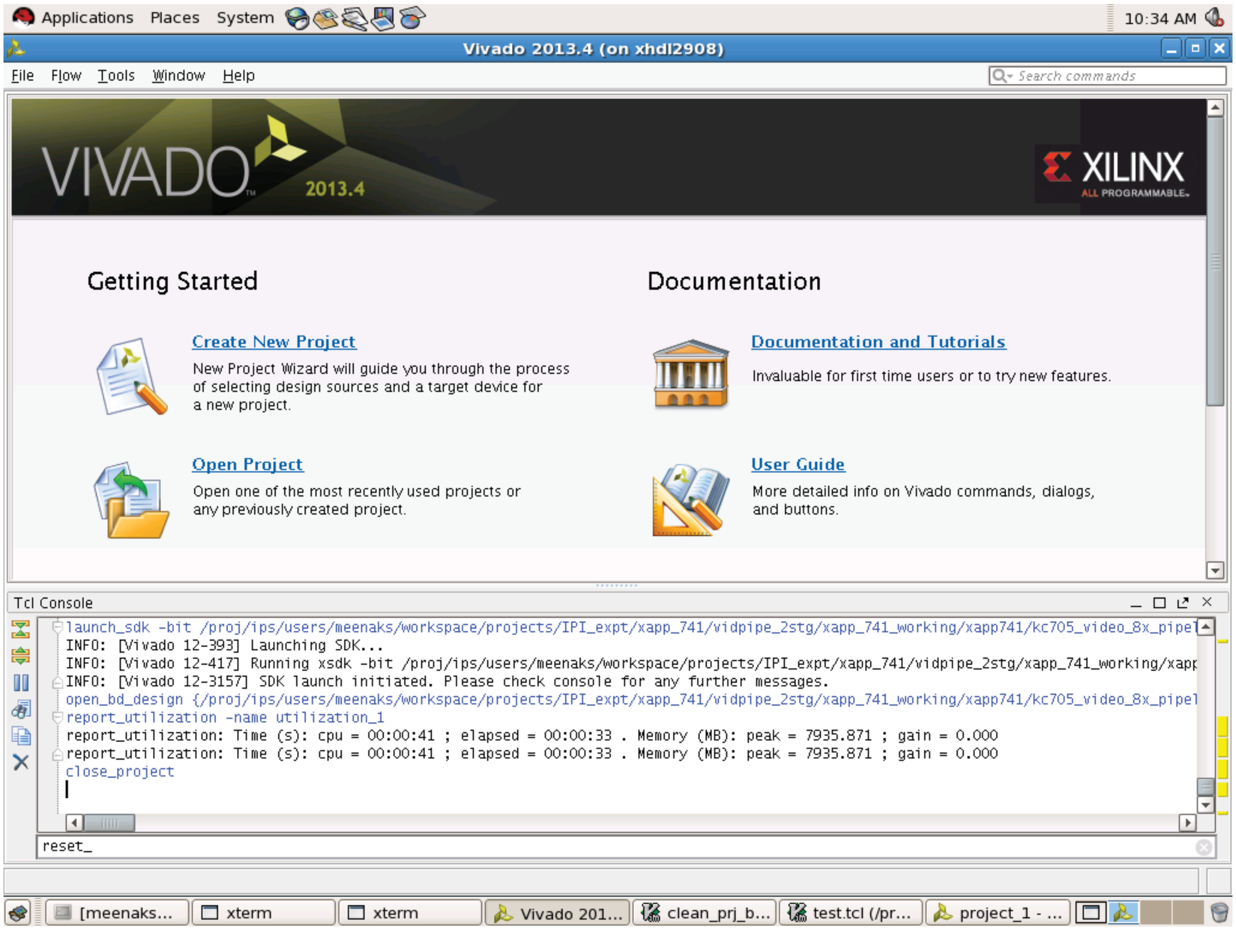
This section covers rebuilding the hardware design using the Vivado IP Integrator tools design flows. Before rebuilding the project, you must ensure that the full system hardware evaluation or full licenses for AXI OSD and Chroma Resampler are installed. To obtain evaluation licenses for the AXI OSD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

- Xilinx Chroma Resampler LogiCORE IP product page [Ref 10] and *LogiCORE IP Chroma Resampler Product Guide for Vivado Design Suite* (PG012) [Ref 11]
- Video On-Screen Display product page [Ref 12] and *LogiCORE IP Video On-Screen Display Product Guide* (PG010) [Ref 8]

Vivado Tools Design Flow

To open and rebuild the design:

1. Install the Vivado Design Suite 2013.4 (requires System Edition at a minimum).
2. Unzip the reference design files accompanying this application note into a local folder (referred to as `<unzip dir>`).
3. 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.
4. In the Vivado tools, select **Getting Started > Open Project** (Figure 7).



Applications Places System
10:34 AM
Vivado 2013.4 (on xhdl2908)
File Flow Tools Window Help
Search commands
VIVADO 2013.4
XILINX ALL PROGRAMMABLE.

Getting Started

Create New Project

New Project Wizard will guide you through the process of selecting design sources and a target device for a new project.

Open Project

Open one of the most recently used projects or any previously created project.

Documentation

Documentation and Tutorials

Invaluable for first time users or to try new features.

User Guide

More detailed info on Vivado commands, dialogs, and buttons.

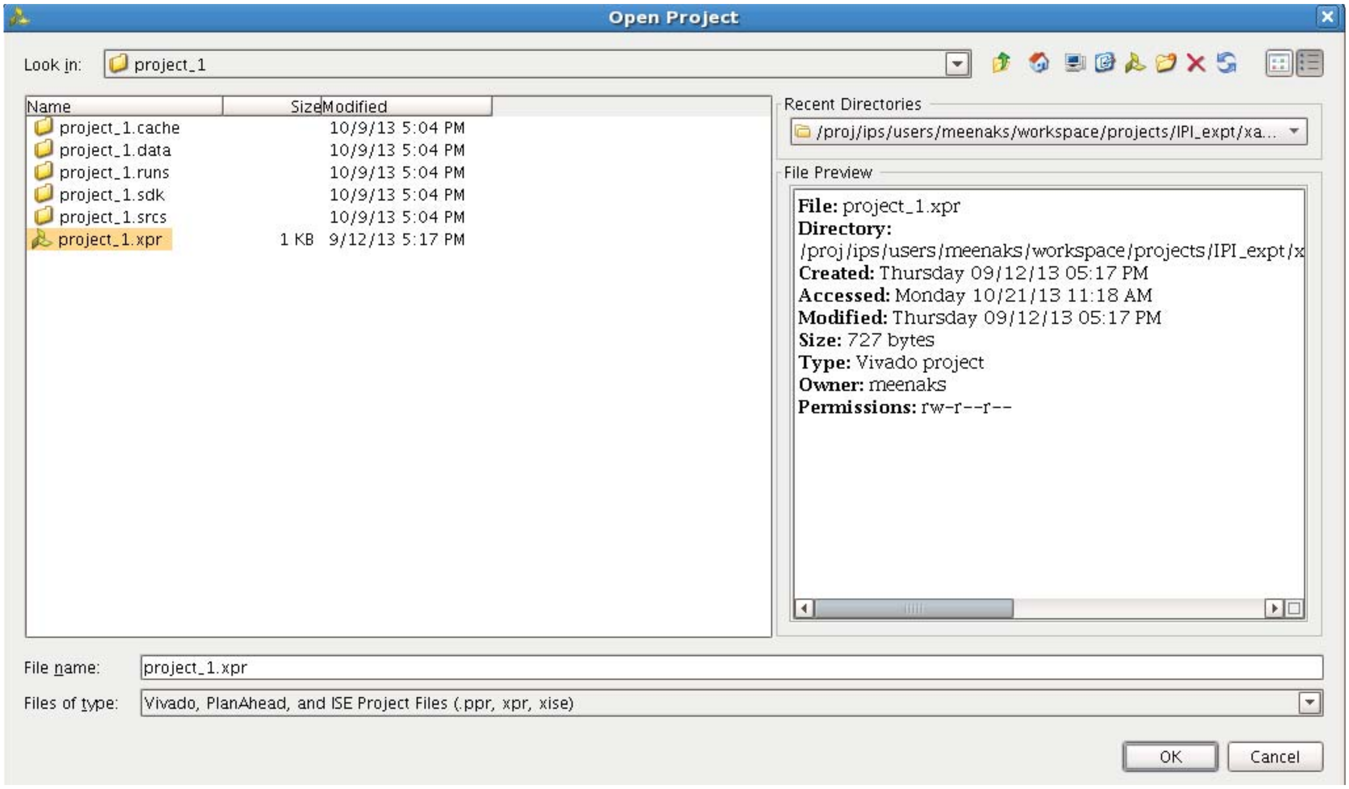
```
launch_sdk -bit /proj/ips/users/meenaks/workspace/projects/IPI_expt/xapp_741/vidpipe_2stg/xapp_741_working/xapp741/kc705_video_8x_pipel
INFO: [Vivado 12-393] Launching SDK...
INFO: [Vivado 12-417] Running xsdk -bit /proj/ips/users/meenaks/workspace/projects/IPI_expt/xapp_741/vidpipe_2stg/xapp_741_working/xapp
INFO: [Vivado 12-3157] SDK launch initiated. Please check console for any further messages.
open_bd_design {/proj/ips/users/meenaks/workspace/projects/IPI_expt/xapp_741/vidpipe_2stg/xapp_741_working/xapp741/kc705_video_8x_pipel
report_utilization -name utilization_1
report_utilization: Time (s): cpu = 00:00:41 ; elapsed = 00:00:33 . Memory (MB): peak = 7935.871 ; gain = 0.000
report_utilization: Time (s): cpu = 00:00:41 ; elapsed = 00:00:33 . Memory (MB): peak = 7935.871 ; gain = 0.000
close_project
```

```
reset_
```

X742_07_020614

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

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



Look in: project_1

Name	Size	Modified
project_1.cache		10/9/13 5:04 PM
project_1.data		10/9/13 5:04 PM
project_1.runs		10/9/13 5:04 PM
project_1.sdk		10/9/13 5:04 PM
project_1.srcs		10/9/13 5:04 PM
project_1.xpr	1 KB	9/12/13 5:17 PM

Recent Directories: /proj/ips/users/meenaks/workspace/projects/IPI_expt/xa...

File Preview

File: project_1.xpr
Directory:
/proj/ips/users/meenaks/workspace/projects/IPI_expt/x
Created: Thursday 09/12/13 05:17 PM
Accessed: Monday 10/21/13 11:18 AM
Modified: Thursday 09/12/13 05:17 PM
Size: 727 bytes
Type: Vivado project
Owner: meenaks
Permissions: rw-r--r--

File name: project_1.xpr

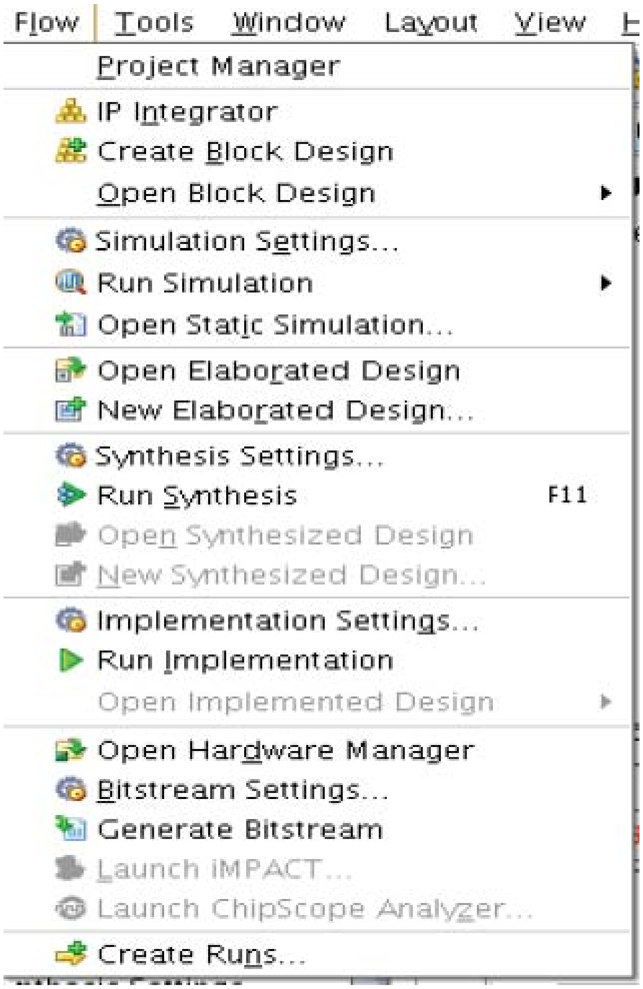
Files of type: Vivado, PlanAhead, and ISE Project Files (.ppr, xpr, xise)

X742_08_020614

Figure 8: **Open Project Window**

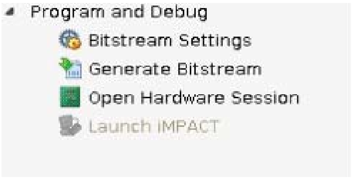
6. Select **Flow > Generate Bitstream** (Figure 9) or click the button next to **Program and Debug > Generate Bitstream** in the Flow Navigator window pane (Figure 10).

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



x742_09_020614

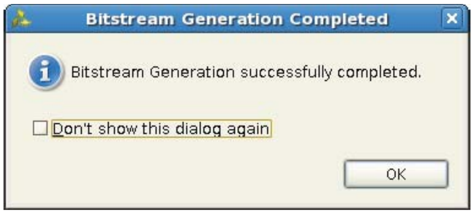
Figure 9: **Starting Bitstream Generation Using Menu Options**



X742_10_020614

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

7. Synthesis, implementation, and bitstream generation operations are performed on the design. When the `Bitstream Generation Completed` message appears, click **OK** (Figure 11).



Bitstream Generation Completed

Bitstream Generation successfully completed.

□ Don't show this dialog again

X742_11_020614

Figure 11: **Bitstream Generation Completed Dialog**

8. The generated bitstream is at `<unzipdir>/vdma_ref_design/HW/project_1/project_1.runs/impl_1/design_1_wrapper.bit.`

Compiling Software and Running Design through SDK

Compiling Software in SDK

1. Start SDK. (In Linux, type **`xsdk`** to start SDK.)
2. In the Workspace Launcher, select Workspace `<unzip dir>/vdma_ref_design/SW/SDK_Workspace.`
3. Click **OK**.
4. If the SDK projects are not visible in Workspace, import the SDK project by selecting **File>Import** .
5. Select **General> Existing Projects into Workspace.**
6. Click **Next.**
7. Change the root directory field to `unzip_dir/vdma_ref_design/SW/SDK_Workspace`.
8. Click **Finish**.

   The board support package (BSP) and software applications compile at this step. The process takes two to five minutes. You can now modify existing applications and create new applications in SDK. (The following steps are only needed if the goal is to rebuild the hardware as described in Building Hardware.)
9. Right-click on the **`hw_platform_0`** and select **Change Hardware Platform Specification**.
10. Click **Yes**.
11. Select the XPS SDK export `design_1.xml` file (`<unzip dir>/vdma_ref_design/HW/project_1/project_1.sdk/SDK/SDK_Export/hw/design_1.xml`).
12. Click **OK**.

The BSP and software applications compile again at this step. The process takes two to five minutes.

Running the Hardware and Software through SDK

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

   Ensure bootloop is used for microblaze_0.
2. Click **Program**.

3. In the Project Explorer window, right-click **vdma_display > Run As > Launch on Hardware**.

Design Characteristics

This reference design is implemented in a Kintex-7 FPGA XC7K325TFFG900-2 using the Vivado Design Suite System Edition 2013.4.

The resources used are:

- Total LUTs used: 39,565 out of 203,800 (19%)
- Total I/Os used: 143 out of 582 (25%)
- Total internal memory used:
    - RAMB36E1s: 84 out of 445 (19%)
    - RAMB18E1s: 55 out of 890 (6%)

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=185348

The reference design matrix is shown in Table 4.

Table 4: **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 and IP from existing Xilinx application note and reference designs or third-party software	Reference designs provided for Vivado IPI 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 tools/version used	Vivado design tools
Implementation tools/versions used	Vivado System Edition 2013.4

Table 4: **Reference Design Matrix *(Cont'd)***

Parameter	Description
Static timing analysis performed	Yes (passing timing in Place & Route)
Hardware Verification	
Hardware verified	Yes
Hardware platform used for verification	KC705 board

Utilization and Performance

Table 5 shows the device and utilization information.

Table 5: **XC7K325T-FFG900-2 Device Utilization Report**

Component	Vivado IP Integrator Tool Flow
Slice Logic	
Slice Registers	45,170 (11%)
Occupied Slices	107,352 (21%)
Slice LUTs	39,565 (19%)
IOBs	
I/Os	143 (29%)
Memory	
RAMB36E1s	84 (19%)
RAMB18E1s	55 (6%)
Clocking	
BUFGCTRL	8 (25%)
MMCME2_ADV	2 (20%)
PLLE2_ADV	1 (10%)
General	
Run Time	70 minutes
Timing Violations	None

Device resource utilization is detailed in Table 5 for the IP cores shown in Figure 1. The information in Table 6 is taken from the utilization_1 tab in the Vivado GUI 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 6: **Module Level Resource Utilization**

IP Core	Instance Name	Slices	Slice Registers	LUTs	LUT RAM	F7 MUX	F8 MUX	Block RAM FIFO	DSP	MMCM ADV/PLL
MIG	mig_1	32,962	10,485	12,962	2,321	336	2	0	0	2
AXI interconnect	axi_interconnect_1	2,096	824	704	98	2	0	0	0	0
	axi_interconnect_2	28,735	12,272	9,686	1,414	78	0	56	0	0
Video timing controller	v_tc_1	5,312	2,796	1,645	7	108	0	0	0	0
AXI TPG	v_tpg_1	4,329	2,204	1,494	31	40	0	1	3	0
AXI VDMA	axi_vdma_1	7,074	3,589	2,533	188	25	0	7	0	0
AXI OSD and associated interconnect logic	v_rgb2ycrcb_0	3,092	1,656	1,008	76	21	3	0	4	0
	v_cresample_0	6,709	3,483	2,164	81	64	0	0	0	0
	v_osd_1	4,213	2,507	1,002	35	40	0	0	2	0
	v_axi4s_vid_out_1	383	197	95	0	3	0	5	0	0
	hdmi_interface_0	73	40	0	0	0	0	0	0	0
Clk, reset, and miscellaneous system logic	clk_wiz_1	0	0	0	0	0	0	0	0	1
	proc_sys_reset_1	59	38	18	1	0	0	0	0	0
MicroBlaze Processor Subsystem (includes local memory and debug module for JTAG-based debug)	microblaze_1	4,843	1,613	1,842	192	129	4	10	3	0
	mdm_1	322	138	102	12	0	0	0	0	0
	lmb_v10_1	1	1	0	0	0	0	0	0	0
	lmb_v10_2	1	1	0	0	0	0	0	0	0
	lmb_bram_if_ctrl_1	16	2	6	0	0	0	0	0	0
	lmb_bram_if_ctrl_2	5	2	2	0	0	0	0	0	0
	blk_mem_gen_1	342	6	144	0	64	0	32	0	0
	axi_intc_1	541	195	230	32	0	0	0	0	0
AXI IIC	axi_iic_1	1,003	351	415	10	5	0	0	0	0
AXI UartLite	axi_uartlite_1	239	91	96	10	1	0	0	0	0
AXI perf mon	axi_perf_mon_0	7,786	2,446	3,163	6	512	0	2	0	0
Total		107,352	45,170	39,565	4,501	1,445	5	112	12	3

Notes:

1. Slices can be packed with basic elements from multiple IP cores and hierarchies. Therefore, a slice is counted in every hierarchical module that each of its packed basic elements belong to. This results in some double counting of slice counts when adding up the slice counts across modules.

Table 7 summarizes the bandwidth calculations for the physical memory interface.

Table 7: **Physical DDR3 Memory Interface Maximum Theoretical Bandwidth**

Data Width	Data Rate	Maximum Theoretical Bandwidth
64 bits (SODIMM)	800 Mb/s	6.40 GB/s (51.2 Gb/s)

Table 8 summarizes the total bandwidth of video data moved through memory (resolution = 1920 x 1080 and frame rate = 60 Hz).

Table 8: **Average Bandwidth Used for Video Traffic**

Frame Resolution	Refresh Rate	Bits Per Pixel	Number of Video Streams	Video Throughput (Total Available Bandwidth)
1920 x 1080	60 Hz	24	2	0.746 GB/s (5.97 Gb/s)

Table 9 summarizes the percentage of the maximum theoretical bandwidth used by the video streams for various resolutions.

Table 9: **Percentage of the Maximum Theoretical Bandwidth Used**

Resolution	Horizontal Active Video x Vertical Active Video	Frame Rate (Hz)	Maximum Theoretical Bandwidth (MB/s)	Percentage of the Maximum Theoretical Bandwidth Used
1	640 x 480	60	110.59	1.93
		75	138.24	2.41
2	720 x 480	60	124.42	2.23
		75	155.52	2.79
3	800 x 600	60	172.80	3.20
		75	216.0	4.0
4	1024 x 768	60	283.12	5.23
		75	353.89	6.54
5	1280 x 720	60	331.78	6.05
		75	414.72	7.56
6	1280 x 1024	60	471.86	8.73
		75	589.82	10.91
7	1920 x 1080	60	746.5	12.72
		75	933.12	15.95
8	1600 x 1200	60	691.20	12.95
		75	864.0	16.19

Notes:

1. Because VDMA is in free running mode, write throughput is always more than read throughput. So the percentage of DDR bandwidth used is slightly more than the theoretical value.

Conclusion

The reference design accompanying this application note demonstrates the configuration of Xilinx video IP cores through their register interface for processing various desired combinations of video resolutions and frame rates. Video clocks for processing different frame rates and resolutions of the design are supplied by the onboard configurable clock generator. The AXI performance monitor IP core connected to DDR memory is used to measure bandwidth and latencies of the system for various resolutions and frame rates. This application note thus details the effective configuration of Xilinx video IP cores to implement various functions for many video applications.

References

This application note uses the following references:

1. AMBA AXI4 specifications
2. UG995, *Vivado Design Suite Tutorial: Designing IP Subsystems Using IP Integrator*
3. PG059, *LogiCORE IP AXI Interconnect Product Guide*
4. PG020, *LogiCORE IP AXI Video Direct Memory Access (axi_vdma) Product Guide*
5. DS176, *7 Series FPGAs Memory Interface Solutions*
6. PG016, *LogiCORE IP Video Timing Controller Product Guide*
7. PG044, *LogiCORE IP AXI4-Stream to Video Out Product Guide*
8. PG010, *LogiCORE IP Video On-Screen Display Product Guide*
9. UG883, *Kintex-7 FPGA KC705 Evaluation Kit Getting Started Guide*
10. Xilinx Chroma Resampler LogiCORE IP
11. PG012, *LogiCORE IP Chroma Resampler Product Guide for Vivado Design Suite*
12. Xilinx Video On-Screen Display LogiCORE IP
13. UG081, *MicroBlaze Processor Reference Guide: Embedded Development Kit*
14. UG761, *AXI Reference Guide*
15. UG994, *Vivado Design Suite User Guide: Designing IP Subsystems Using IP Integrator*
16. UG810, *KC705 Evaluation Board for the Kintex-7 FPGA User Guide*
17. PG013, *LogiCORE IP RGB to YCrCb Color Space Converter Product Guide for Vivado Design Suite*

Revision History

The following table shows the revision history for this document.

Date	Version	Description of Revisions
05/03/2012	1.0	Initial Xilinx release.
10/05/2012	1.1	Updated software version from 13.4 to 14.2 throughout. Updated Included Systems. Added Vivado Design Suite to Hardware and Software Requirements. Added Video block to Figure 1. Added Streaming to Video Bridge. Removed workaround for C_PRMRY_IS_ACLK_ASYNC. Removed note in Executing the Reference System Using the Pre-Built Bitstream and the Compiled Software Application. Updated first paragraph of Building Hardware. Added Vivado Tools Design Flow. Updated step 9 and step 11 in Compiling Software and Running Design through SDK. Updated resource usage in Design Characteristics. Updated utilization numbers in Table 5 and Table 6. Added *Video In to AXI4-Stream v1.0 Product Guide* to References.
02/26/2014	1.2	Updated for Vivado Design Suite 2013.4 and the revision 1.1 KC705 evaluation board. XC7K325TFFG900-1 became XC7K325TFFG900-2. Updated Figure 1. Updated Table 1, Reference System Address Map. Updated Table 3, Frame Resolution Versus Bandwidth of Video Streams, changing all bandwidth values. Added sections RGB2YCrCb Converter, page 8 and Chroma Resampler, page 8. Removed former Figure 3, Sample Three-Layer OSD Core Block Diagram. Changed the AXI VDMA interface bus widths from 32-bit to 24-bit. Changed use of AXI 7 series DDRX to Memory Interface Generator (MIG) throughout. Updated Figure 6. Obsoleted XPS Design Flow procedure. Updated Figure 7, Figure 8, and Figure 9. Modified Compiling Software in SDK procedure. Modified resources used in Design Characteristics. Updated Table 4, Reference Design Matrix. Replaced XPS Flow column in Table 5, Device Utilization Report, with Vivado IP Integrator Tool Flow utilization column. Replaced Table 6, Module Level Resource Utilization. Updated Table 8, Average Bandwidth Used for Video Traffic. Updated all values in Table 9, Percentage of the Maximum Theoretical Bandwidth Used. Updated the References section.

Disclaimer

The information disclosed to you hereunder (the “Materials”) is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of Xilinx’s limited warranty, please refer to Xilinx’s Terms of Sale which can be viewed at www.xilinx.com/legal.htm#tos; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such critical applications, please refer to Xilinx’s Terms of Sale which can be viewed at www.xilinx.com/legal.htm#tos.