Video Remapper v1.00a

DS879 October 16, 2012 **Product Specification**

Introduction

The Xilinx LogiCORE™ IP Video Remapper core is an easy-to-use IP core for assigning inbound video color component channels to different outbound component channels on AXI4-Stream interfaces. It allows for up to four inbound video to be remapped to different outbound video component channels. This core allows reordering, duplicating and width-changing all color components.

Features

- Independently selectable inbound and outbound component widths of 8, 10, 12, and 16 bits.
- Independently configurable number of inbound and outbound components up to 4 components per interface.
- Configurable outbound component values either from inbound or constant value.
- Allows for same inbound component to be mapped to multiple outbound components on AXI4-Stream interfaces.

LogiCORE IP Facts Table	
Core Specifics	
Supported Device Family(1)	Zynq™-7000 (2), Artix™-7, Virtex®-7, Kintex™-7, Virtex-6, Spartan®-6
Supported User Interfaces	AXI4-Stream(3)
Resources	N/A
Provided with Core	
Design Files	Verilog Source Code
Example Design	Not Provided
Test Bench	Not Provided
Constraints File	Not Provided
Simulation Model	Verilog Source Code
Supported S/W Driver (4)	N/A
Tested Design Flows(5)	
Design Entry	ISE Design Suite Embedded Edition v14.3,
Simulation	Mentor Graphics ModelSim, Xilinx® ISim
Synthesis	Xilinx Synthesis Technology (XST)
Support	
Provided by Xilinx @ www.xilinx.com/support	

Notes:

1. For a complete list of supported derivative devices, see Embedded Edition Derivative Device Support.
2. Supported in ISE Design Suite implementations only.
3. Video protocol as defined in the *Video IP: AXI Feature Adoption* section of (UG761) *AXI Reference Guide* [Ref 1].
4. Standalone driver details can be found in the EDK or SDK directory (<*install_directory*>/doc/usenglish/xilinx_drivers.htm). Linux OS and driver support information is available from //wiki.xilinx.com.
5. For the supported versions of the tools, see the Xilinx Design Tools: Release Notes Guide.

Overview

The Video Remapper core allows for any video color component within the inbound AXI4-Stream interface to be mapped onto any video color component channel within the outbound AXI4-Stream interface. Video color component channel widths of 8, 10, 12, and 16 are supported, and 1-, 2-, 3- or 4-color-component channels are supported. If the outbound color component channel data width is different than the inbound data width, then the outbound color component channels are truncated or padded according to the rules in Component Data Widths.

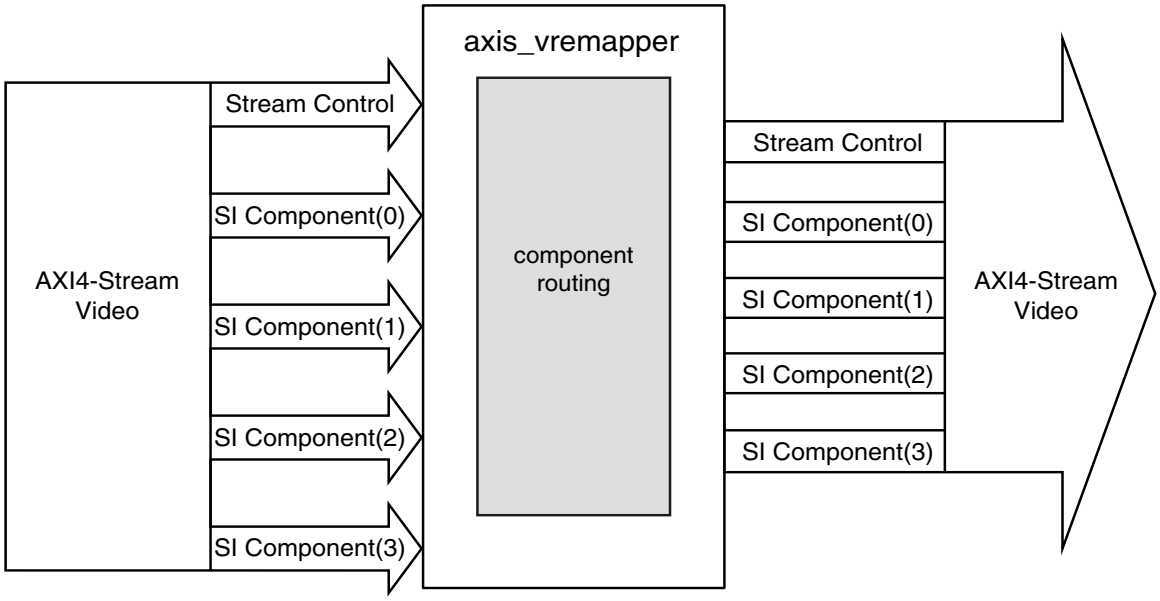

Figure 1: **Video Remapper Core**

The Video Remapper core is compliant with the AXI4-Stream Video Protocol standard. Refer to the *Video IP: AXI Feature Adoption* section of the *AXI Reference Guide (UG761)* [Ref 1]for additional information.

Functional Description

The Video Remapper core provides a method to redefine the static routing of the SI Components to different MI Component without any buffering.

Latency

The Video Remapper core does not introduce any latency to the channel because it is implemented as wire connections.

Throughput

The Video Remapper core does not affect the channel throughput because it is implemented as wire connections.

Parameterization

Table 1 describes the core parameters.

Table 1: **Video Remapper Parameters**

Parameter Name	Default Value	Allowable Values	Description
C_S_COMPONENT_WIDTH	8	8,10,12,16	Width of the each of the components on the inbound slave interface
C_NUM_S_COMPONENTS	1	1 - 4	Number of components represented in the inbound video stream
C_M_COMPONENT_WIDTH	8	8,10,12,16	Width of the each of the components on the outbound master interface
C_NUM_M_COMPONENTS	1	1 -4	Number of components represented in the outbound video stream
C_M0_COMPONENT_MAP	1	1 - 5	Selects which inbound component drives outbound component 0. A value of 5 selects the constant value of C_M0_CONSTANT_VALUE to be driven.
C_M1_COMPONENT_MAP	1	1 - 5	This parameter is only valid when C_NUM_M_COMPONENTS > 1. Selects which inbound component drives outbound component 1. A value of 5 selects the constant value of C_M1_CONSTANT_VALUE to be driven.
C_M2_COMPONENT_MAP	2	1 - 5	This parameter is only valid when C_NUM_M_COMPONENTS > 2.Selects which inbound component drives outbound component 2. A value of 5 selects the constant value of C_M2_CONSTANT_VALUE to be driven.
C_M3_COMPONENT_MAP	3	1 - 5	This parameter is only valid when C_NUM_M_COMPONENTS > 3.Selects which inbound component drives outbound component 3. A value of 5 selects the constant value of C_M3_CONSTANT_VALUE to be driven.
C_M0_CONSTANT_VALUE	0x0000	0x0000-0xFFFF	Value of constant value to be driven on outbound component 0 when C_M0_COMPONENT_MAP equals 5.
C_M1_CONSTANT_VALUE	0x0000	0x0000-0xFFFF	This parameter is only valid when C_NUM_M_COMPONENTS > 1. Value of constant value to be driven on outbound component 1 when C_M1_COMPONENT_MAP equals 5.
C_M2_CONSTANT_VALUE	0x0000	0x0000-0xFFFF	This parameter is only valid when C_NUM_M_COMPONENTS > 2. Value of constant value to be driven on outbound component 2 when C_M2_COMPONENT_MAP equals 5.
C_M3_CONSTANT_VALUE	0x0000	0x0000-0xFFFF	This parameter is only valid when C_NUM_M_COMPONENTS > 3. Value of constant value to be driven on outbound component 3 when C_M3_COMPONENT_MAP equals 5.

Core Interfaces

Port Descriptions

The Video Remapper core is compliant with the AXI4-Stream Video Protocol standard. Figure 2 illustrates an I/O diagram of the Video Remapper core.

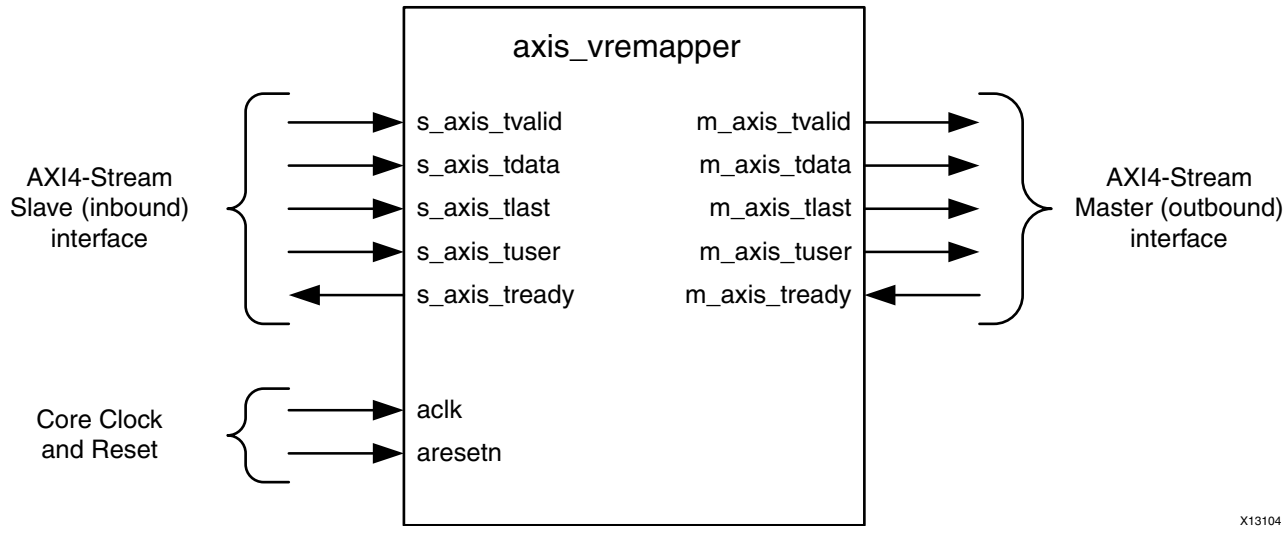

Figure 2: **Video Remapper Core Top-Level Signaling Interface**

Common Interface Signals

Table 2 summarizes the signals which are either shared by, or not part of the dedicated AXI4-Stream data control interfaces.

Table 2: **Common Interface Signals**

Signal Name	Direction	Width	Description
aclk	In	1	Video Core Clock
aresetn	In	1	Video Core Active Low Synchronous Reset

The `aclk` and `aresetn` signals are shared between the core and the AXI4-Stream data interfaces.

aclk

The AXI4-Stream interface must be synchronous to the core clock signal `aclk`. All AXI4-Stream interface's input signals are sampled on the rising edge of `aclk`. All AXI4-Stream output signal's changes occur after the rising edge of `aclk`.

aresetn

The `aresetn` signal must be synchronous to the aclk `and` must be held low for a minimum of 32 clock cycles of the slowest clock.

Data Interface

The Video Remapper core receives and transmits data using AXI4-Stream interfaces that implement a video protocol as defined in the *Video IP: AXI Feature Adoption* section of the (UG761) *AXI Reference Guide* [Ref 1].

AXI4-Stream Signal Names and Descriptions

Table 3 describes the AXI4-Stream inbound signal names and descriptions.

Table 3: **AXI4-Stream Data Interface Signal Descriptions**

Signal Name	Direction	Width	Description
s_axis_tdata	In	Automatically calculated based on C_NUM_S_COMPONENTS and C_S_AXIS_COMPONENT_WIDTH	Input Video Data
s_axis_tvalid	In	1	Input Video Valid Signal
s_axis_tready	Out	1	Input Ready
s_axis_tuser	In	1	Input Video Start Of Frame
s_axis_tlast	In	1	Input Video End Of Line

Table 4 describes the AXI4-Stream outbound signal names and descriptions.

Table 4: **AXI4-Stream Data Interface Signal Descriptions**

Signal Name	Direction	Width	Description
m_axis_tdata	Out	Automatically calculated based on C_NUM_M_COMPONENTS and C_M_AXIS_COMPONENT_WIDTH	Interface Output Video Data
m_axis_tvalid	Out	1	Interface Output Video Valid
m_axis_tready	In	1	Interface Output Video Ready
m_axis_tuser	Out	1	Interface Output Video Start of Frame
m_axis_tlast	Out	1	Interface Output Video End of Line

Video Data

The AXI4-Stream interface specification restricts `TDATA` widths to integer multiples of 8 bits. The Video Remapper `*_axis_tdata` is packed and padded to multiples of 8 bits as necessary. Zero padding the most significant bits is only necessary for 10 and 12 bit wide data.

Component Data Widths

If the outbound component data width (`C_M_COMPONENT_WIDTH`) is greater than the inbound component data width (`C_S_COMPONENT_WIDTH`), then the outbound component data is zero padded on the least significant bits. If the outbound component data width (`C_M_COMPONENT_WIDTH`) is less than the inbound component data width (`C_S_COMPONENT_WIDTH`), then the outbound component data is truncated on the least significant bits.

READY/VALID Handshake

A valid transfer occurs whenever `READY`, `VALID`, and `aresetn` are high at the rising edge of `aclk`, as seen in Figure 3. During valid transfers, `DATA` only carries active video data. Blank periods and ancillary data packets are not transferred through the AXI4-Stream video protocol.

Guidelines on Driving s_axis_tvalid

Once s_axis_tvalid is asserted, no interface signals (except the Video Remapper core driving s_axis_tready) may change value until the transaction completes (s_axis_tready and s_axis_tvalid are high on the rising edge of aclk). Once asserted, s_axis_tvalid may only be de-asserted after a transaction has completed. Transactions may not be retracted or aborted. In any cycle following a transaction, s_axis_tvalid can either be de-asserted or remain asserted to initiate a new transfer.

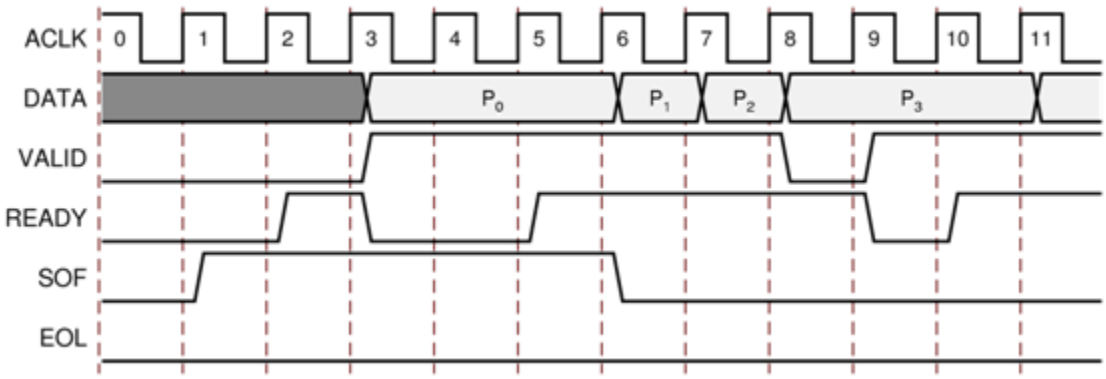

Figure 3: **Example of READY/VALID Handshake, Start of a New Frame**

Guidelines on Driving m_axis_tready

The m_axis_tready signal may be asserted before, during or after the cycle in which the Video Remapper core asserted m_axis_tvalid. The assertion of m_axis_tready may be dependent on the value of m_axis_tvalid. A slave that can immediately accept data qualified by m_axis_tvalid, should pre-assert its m_axis_tready signal until data is received. Alternatively, m_axis_tready can be registered and driven the cycle following VALID assertion. It is recommended that the AXI4-Stream slave should drive READY independently, or pre-assert READY to minimize latency.

Start of Frame Signals - m_axis_tuser, s_axis_tuser

The Start-Of-Frame (SOF) signal, physically transmitted over the AXI4-Stream TUSER[0] signal, marks the first pixel of a video frame. The SOF pulse is 1 valid transaction wide, and must coincide with the first pixel of the frame, as seen in Figure 3. The SOF signal serves as a frame synchronization signal, which allows downstream cores to re-initialize, and detect the first pixel of a frame. The SOF signal may be asserted an arbitrary number of aclk cycles before the first pixel value is presented on DATA, as long as a VALID is not asserted.

End of Line Signals - m_axis_tlast, s_axis_tlast

The End-Of-Line (EOL) signal, physically transmitted over the AXI4-Stream TLAST signal, marks the last pixel of a line. The EOL pulse is 1 valid transaction wide, and must coincide with the last pixel of a scan-line.

Support

Xilinx provides technical support for this LogiCORE IP product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support of product if implemented in devices that are not defined in the documentation, if customized beyond that allowed in the product documentation, or if changes are made to any section of the design labeled *DO NOT MODIFY*.

Licensing and Ordering Information

This Xilinx LogiCORE IP module is provided at no additional cost with the Xilinx ISE Design Suite Embedded Edition tools under the terms of the Xilinx End User License.

Information about this and other Xilinx LogiCORE IP modules is available at the Xilinx Intellectual Property page. For information on pricing and availability of other Xilinx LogiCORE IP modules and tools, contact your local Xilinx sales representative.

References

1. UG761 AXI Reference Guide

Revision History

The following table shows the revision history for this document:

Date	Version	Description of Revisions
10/16/2012	1.0	Initial Xilinx release.

Notice of Disclaimer