Introduction

RGB888 to 8-bit YCbCr converter converts RGB color space to YCbCr 4:2:2 color space. To facilitate easy insertion to practical video systems, this design example takes video stream control signals (H_SYNC, V_SYNC, and DEN) and delays them appropriately, so that control signals can be easily synchronized with the output video stream. This document provides a brief description of RGB888 to YcbCr 8-bit Converter and its implementation.

The design is implemented in VHDL. The Lattice iCEcube2™ Place and Route tool integrated with the Synopsys Synplify Pro® synthesis tool is used for the implementation of the design. The design can be targeted to other iCE FPGA family devices.

Features

- RGB88 mode input and 8-bit YCbCr 4:2:2 mode output
- Pipelined implementation
- Latency of 5 cycles
- H_SYNC, V_SYNC and DEN control signals for video synchronization

Functional Description

*Figure 1. Functional Description*
**Signal Description**

**Table 1. Signal Description**

<table>
<thead>
<tr>
<th>Signal</th>
<th>Width</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>i_CLK</td>
<td>1</td>
<td>Input</td>
<td>Input Pixel Clock</td>
</tr>
<tr>
<td>i_RST</td>
<td>1</td>
<td>Input</td>
<td>Asynchronous Active High System Reset</td>
</tr>
<tr>
<td>i_PIX_DEN</td>
<td>1</td>
<td>Input</td>
<td>Data Enable (RGB valid) synchronized with pixel clock</td>
</tr>
<tr>
<td>i_H_SYNC</td>
<td>1</td>
<td>Input</td>
<td>Horizontal Sync</td>
</tr>
<tr>
<td>i_V_SYNC</td>
<td>1</td>
<td>Input</td>
<td>Vertical Sync</td>
</tr>
<tr>
<td>i_RED</td>
<td>8</td>
<td>Input</td>
<td>Red component of pixel</td>
</tr>
<tr>
<td>i_GREEN</td>
<td>8</td>
<td>Input</td>
<td>Green component of pixel</td>
</tr>
<tr>
<td>i_BLUE</td>
<td>8</td>
<td>Input</td>
<td>Blue component of pixel</td>
</tr>
<tr>
<td>o_PIX_DEN</td>
<td>1</td>
<td>Output</td>
<td>YCbCr valid data synchronized with pixel clock</td>
</tr>
<tr>
<td>o_H_SYNC</td>
<td>1</td>
<td>Output</td>
<td>Pipelined Horizontal Sync</td>
</tr>
<tr>
<td>o_V_SYNC</td>
<td>1</td>
<td>Output</td>
<td>Pipelined Vertical Sync</td>
</tr>
<tr>
<td>o_Y</td>
<td>8</td>
<td>Output</td>
<td>Converted Y component</td>
</tr>
<tr>
<td>o_Cb</td>
<td>8</td>
<td>Output</td>
<td>Converted Cb component</td>
</tr>
<tr>
<td>o_Cr</td>
<td>8</td>
<td>Output</td>
<td>Converted Cr component</td>
</tr>
</tbody>
</table>

**Design Module Description**

*Figure 2. Functional Block Diagram*

**Configurable parameter**

None
Register Map
This design does not have any user accessible registers or memory.

Design Details
This module converts RGB to YCbCr, consisting of one luma component(Y) representing brightness, and two chroma components (Cb and Cr) as per the following conversion expressions:

- Y = 16 + (0.2567890625 * Red) + (0.50412890625 * Green) + (0.09790625 * Blue)
- Cb = 128 + (0.14822265625 * Red) + (0.2909921875 * Green) + (0.43921484375 * Blue)
- Cr = 128 + (0.43921484375 * Red) + (0.3677890625 * Green) + (0.071442578125 * Blue)

The implementation comprises of a set of constant coefficient multipliers implemented as shift and add adders. This is a fully synchronous design and all the modules listed in the block diagram generate registered outputs, clocked by input pixel clock. Considering the large amount of data path involved here, a pipelined implementation is provided to improve the performance. Computed Y, Cb and Cr values are clipped and limited to maximum/minimum permissible range. To facilitate easy insertion to practical video systems, the design conveniently pipelines the video control signals H_SYNC, V_SYNC, and DEN by introducing a latency of 5 clock cycles.

Initialization Conditions
This design does not have any user specific initialization conditions.

Timing Diagram

Figure 3. Timing Diagram
RGB888 to YCbCr 8-Bit Converter

Simulation Waveforms

Figure 4. Simulation Waveforms

Usage Examples

RGB to YCbCr converters are useful in applications like JPEG and MPEG image encoders, which is used in DVDs, digital TV and Video CDs, where images are coded in YCbCr format. YCbCr is also the most preferred format for hue and saturation control of images.

Simulation setup comprises of a testbench which provides input RGB666 values for various colors like red, blue, green, white etc… The DUT generated output YCbCr 8-bit values are compared against the corresponding known YCbCr values.

Implementation

This design is implemented in VHDL. When using this design in a different device, density, speed or grade, performance and utilization may vary.

Performance and Resource Utilization

Table 2. Performance and Resource Utilization

References

- iCE40 Family Handbook
Technical Support Assistance
Hotline:  1-800-LATTICE (North America)
        +1-503-268-8001 (Outside North America)
E-mail:  techsupport@latticesemi.com
Internet: www.latticesemi.com

Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Change Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>April 2013</td>
<td>01.0</td>
<td>Initial release.</td>
</tr>
</tbody>
</table>