

# iCE65 as RGB888 to 8-bit YCbCr Converter

## Overview

RGB888 to 8-bit YCbCr converter converts RGB color space to YCbCr 4:2:0 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.

## Features

- RGB888 mode input and 8-bit YCbCr 4:2:0 mode output
- Pipelined implementation
- Latency of 5 cycles
- H\_SYNC, V\_SYNC and DEN control signals for video synchronization
- IP-XACT version 1.2 compliant

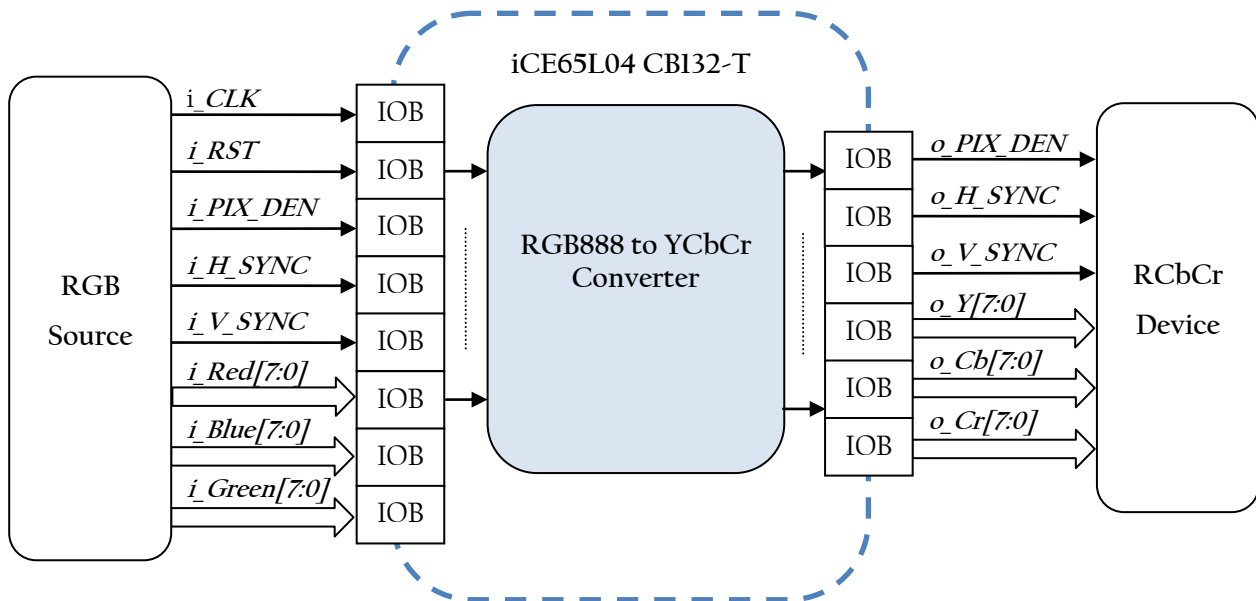
## Resource Utilization

Table 1: Resource Utilization

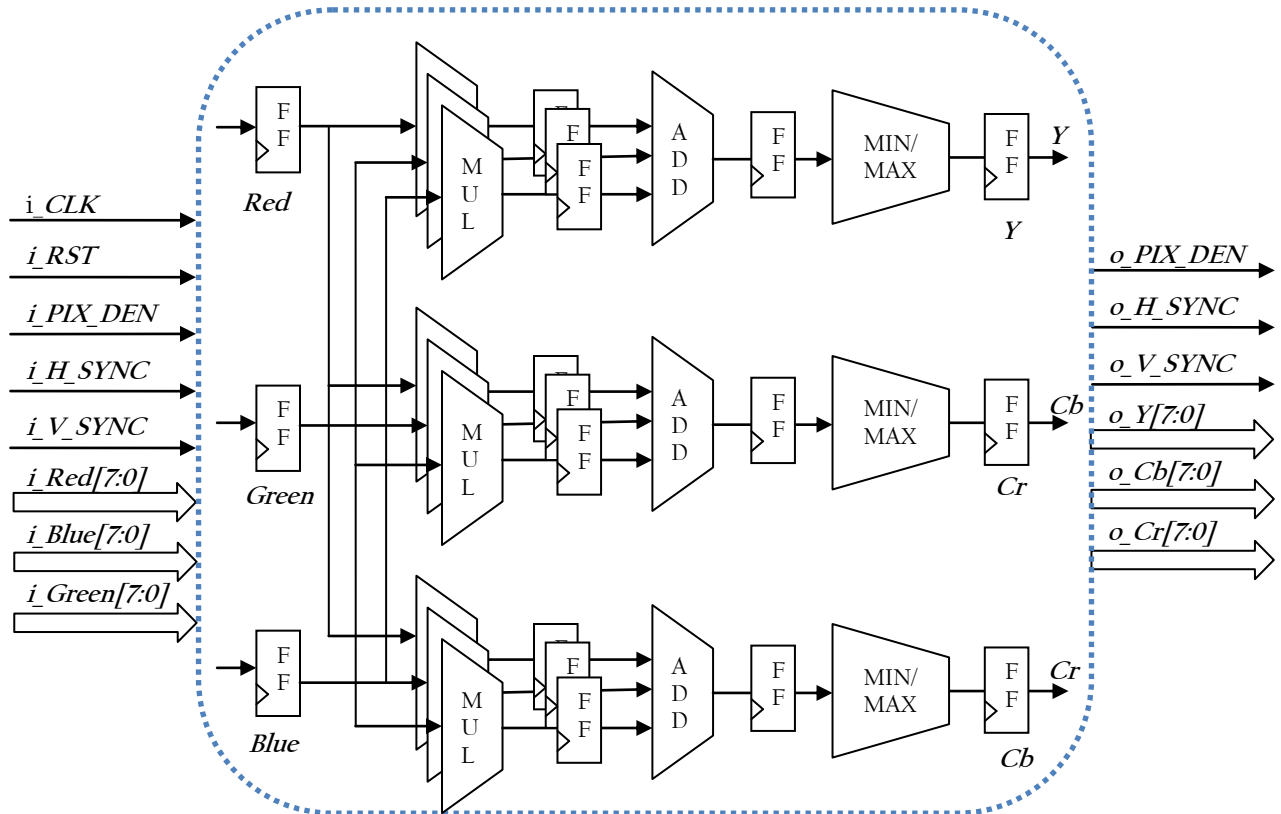
LUTs	Registers	Memory	I/Os	GBs
422	306	0	0	0

Note: Resource Utilization is based on iCECube 2010.12.14671 release.

## System Block Diagram



## Functional Block Diagram



## Design Interface

Signal Name	Pin Type	Signal Description
<b>i_CLK</b>	Input	Input Pixel Clock
<b>i_RST</b>	Input	Asynchronous active high system reset
<b>i_PIX_DEN</b>	Input	Data Enable(RGB valid) synchronized with pixel clock
<b>i_H_SYNC</b>	Input	Horizontal Sync
<b>i_V_SYNC</b>	Input	Vertical Sync
<b>i_Red[7:0]</b>	Input	Red component of Pixel
<b>i_Green[7:0]</b>	Input	Green component of Pixel
<b>i_Blue[7:0]</b>	Input	Blue component of Pixel
<b>o_PIX_DEN</b>	Output	YCbCr Data valid synchronized with pixel clock
<b>o_H_SYNC</b>	Output	Pipelined Horizontal Sync
<b>o_V_SYNC</b>	Output	Pipelined Vertical Sync
<b>o_Y[7:0]</b>	Output	Converted Y Component
<b>o_Cb[7:0]</b>	Output	Converted Cb Component
<b>o_Cr[7:0]</b>	Output	Converted Cr Component

## Configurable Parameters

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:

$$Y = 16 + (0.2567890625 * Red) + (0.50412890625 * Green) + (0.09790625 * Blue)$$

$$Cb = 128 + (0.14822265625 * Red) + (0.2909921875 * Green) + (0.43921484375 * Blue)$$

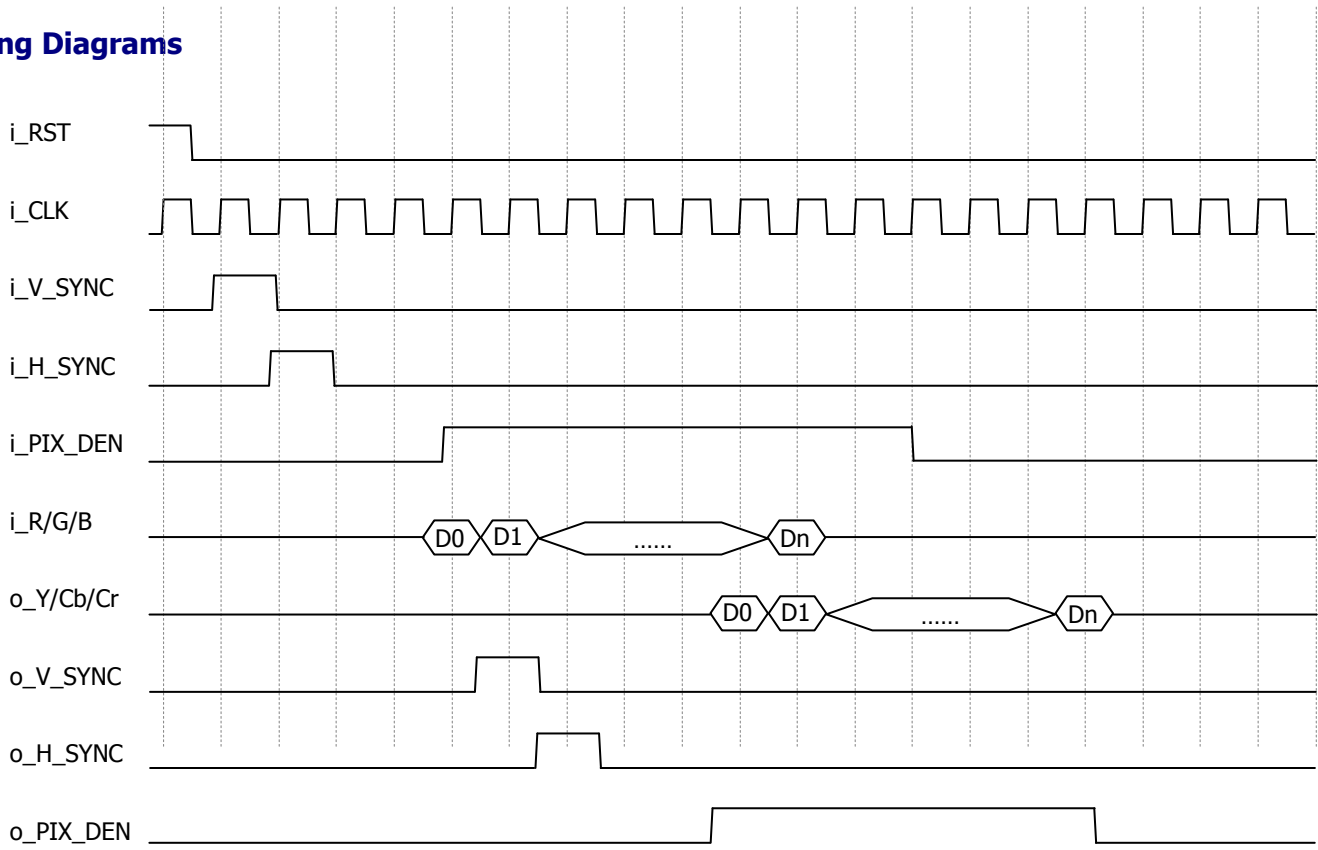
$$Cr = 128 + (0.43921484375 * Red) + (0.3677890625 * Green) + (0.071442578125 * Blue)$$

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 Diagrams



Note: Signal naming here in the waveform is same as that listed in Pin Description table

## 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 RGB888 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.

## System Designer Flow

RGB888ToYCbCr8Bit is compatible with System Designer/IP-XACT 1.2

The System Designer flow is as follows,

1. Launch the System Designer from Synplify Pro using menu 'Import -> Launch System Designer'.
2. Create a new project (open an existing old project, as necessary) and import the IP-XACT XML file
3. Drag and place the component from the 'Library' pane to the 'Design' pane
4. Click on the "Generate Files" button, which generates the necessary files required for synthesis and simulation.

5. Go to Synplify Pro and click on the “Run” button to synthesize the System Designer generated files. Synplify Pro generates all the necessary files for P&R in iCECube.

## References

The following references were used in the creation of this design:

- SiliconBlue Technologies, Inc. “iCE65 Ultra Low-Power mobile FPGA Family” datasheet (26-MAY-2010).
- Wikipedia : <http://en.wikipedia.org/wiki/YCbCr>

## Revision History

Version	Date	Description
1.0	09-SEP-2010	Initial draft document
1.1	07-DEC-2010	IP-XACT format Update

## Disclaimer

Copyright © 2007–2009 by SiliconBlue Technologies LTD. All rights reserved. SiliconBlue is a registered trademark of SiliconBlue Technologies LTD in the United States. Specific device designations, and all other words and logos that are identified as trademarks are, unless noted otherwise, the trademarks of SiliconBlue Technologies LTD. All other product or service names are the property of their respective holders. SiliconBlue products are protected under numerous United States and foreign patents and pending applications, maskwork rights, and copyrights. SiliconBlue warrants performance of its semiconductor products to current specifications in accordance with SiliconBlue's standard warranty, but reserves the right to make changes to any products and services at any time without notice. SiliconBlue assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by SiliconBlue Technologies LTD. SiliconBlue customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.



3255 Scott Blvd.,  
Building 7, Suite 101  
Santa Clara, CA 95054

Tel: 408-727-6101  
Fax: 408-727-6085

[www.SiliconBlueTech.com](http://www.SiliconBlueTech.com)