AN 047: Using the Gamma Correction Example Design

AN047-v1.0
May 2022
www.efinixinc.com

Contents

Introduction

Gamma correction is a nonlinear processing of the luminance in videos or digital images. You can apply gamma correction to properly display brightness on screens.

Efinix provides an example that implements a simple gamma correction design. The example design also includes a testbench with customizable parameters so you can simulate the gamma correction design.

Features

- Fixed input gamma factor of 1.0
- Supports 2.0, 2.1, 2.2, 2.3, and 2.4 gamma curve correction
- Supports 8 bit or 10 bit data width
- RTL design in Verilog HDL
- Includes simulation testbench
- Supported in Titanium and Trion FPGAs

Resource Utilization and Performance

FPGA	Logic Utilization (LUTs)	Memory Blocks	f_{MAX} (MHz)	Language	Efinity Version
Ti60 F225 C4	37	4	1,040	Verilog HDL	2021.2
T20 BGA169 C4	66	8	357		

Functional Description

The gamma correction example involves multiplying the input signal with the display transfer function as shown in the following equation:

$D_{OUT} = D_{IN}^{\gamma}$

where:

- D_{OUT} represents output image data (RGB). Represented in the range of 0 to 1.
- D_{IN} represents input image data (RGB). Represented in the range of 0 to 1.
- γ represents gamma correction factor.

Figure 1: Gamma Correction Example Block Diagram

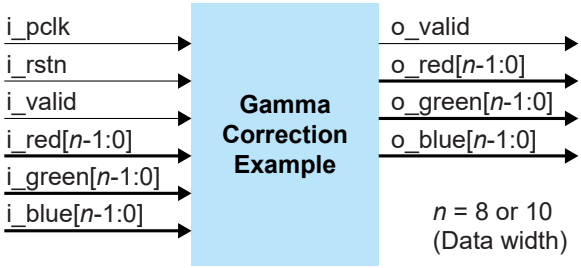


Table 1: Gamma Correction Example Ports

n = 8 or 10

Port Name	Direction	Description
i_pclk	Input	System clock.
i_rstn	Input	Active low reset signal.
i_valid	Input	Input data valid signal. Assert this signal to indicate input data is valid.
i_red[*n*-1:0]	Input	Input red pixel data.
i_green[*n*-1:0]	Input	Input green pixel data.
i_blue[*n*-1:0]	Input	Input blue pixel data.
o_valid	Output	Output data valid signal. The design asserts this signal when the output data is valid.
o_red[*n*-1:0]	Output	Output red pixel data.
o_green[*n*-1:0]	Output	Output green pixel data.
o_blue[*n*-1:0]	Output	Output blue pixel data.

Customizing the Example Design

The example design has parameters so you can customize its function. You define the parameters in the **gamma_correction.v** file.

Table 2: Design Example Paramters

Parameter	Option	Description
DATA_WIDTH	8, 10 (Default)	Represents input and output data width in bit.
GAMMA_CURVE	2P0.mem, 2P1.mem, 2P2.mem (Default), 2P3.mem, 2P4.mem	Targeted gamma correction curve.

Example Design Testbench

The example design includes a simulation testbench which allows you to simulate the gamma correction example. It also includes the simulation run file, **gamma_correction_tb.do**.

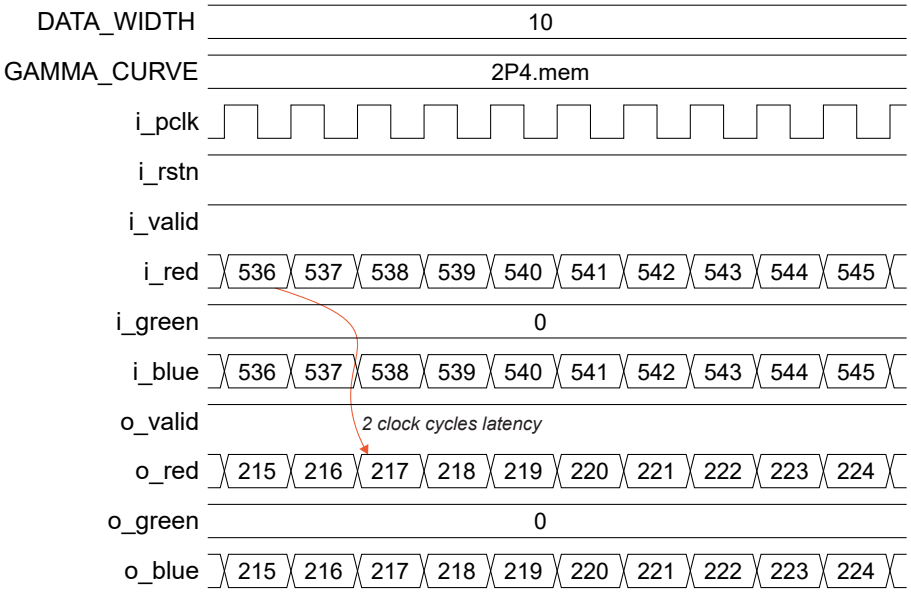
The testbench file, **gamma_correction_tb.v**, defines the example design parameters and other additional video stream parameters as the input data, then calculates the output values. The following table lists the testbench parameters.

Table 3: Testbench Parameters

Parameter	Option	Description
MAX_HRES	Up to 4096	Horizontal resolution. Default: 1024
MAX_VRES	Up to 4096	Vertical resolution. Default: 512
HSP	Up to 2048	Horizontal Sync Pulse Width. Default: 44
HBP	Up to 2048	Horizontal Back Porch Default: 148
HFP	Up to 8192	Horizontal Front Porch Default: 88
VSP	Up to 2048	Vertical Sync Pulse Width Default: 5
VBP	Up to 2048	Vertical Back Porch Default: 36
VFP	Up to 8192	Vertical Front Porch Default: 4
DATA_WIDTH	8, 10	Input and output data width. Default: 10

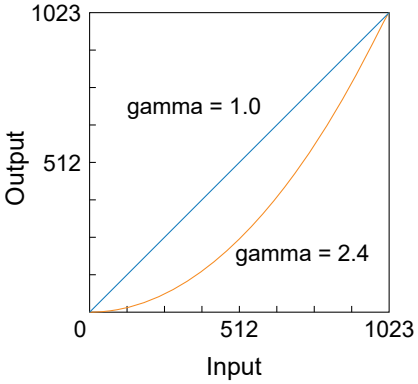
The following figure shows the simulation waveform. The gamma correction starts when `i_valid` signal set to high. The gamma correction output is available at the output after two clock cycles.

Figure 2: Simulation Waveform



The following figure illustrates the gamma curve correction response of 2.4.

Figure 3: Simulation Gamma Correction Curve Response



Revision History

Table 4: Revision History

Date	Version	Description
May 2022	1.0	Initial release.