



Gamma Corrector IP Core - Lattice Radiant Software

User Guide

FPGA-IPUG-02122-1.2

November 2022

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS and with all faults, and all risk associated with such information is entirely with Buyer. Buyer shall not rely on any data and performance specifications or parameters provided herein. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. No Lattice products should be used in conjunction with mission- or safety-critical or any other application in which the failure of Lattice's product could create a situation where personal injury, death, severe property or environmental damage may occur. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Contents

Acronyms in This Document.....	5
1. Introduction.....	6
1.1. Quick Facts.....	6
1.2. Features.....	6
1.3. Conventions.....	7
1.3.1. Nomenclature.....	7
1.3.2. Signal Names.....	7
1.3.3. Host.....	7
1.3.4. Attribute.....	7
2. Functional Description.....	8
2.1. Overview.....	8
2.1.1. Gamma Correction Equation.....	8
2.1.2. Filling the Gamma LUT.....	9
2.1.3. Multi-Color Plane/Channel Mapping.....	9
2.1.4. Dynamically Loadable Gamma LUTs.....	10
2.1.5. Handshake I/O ports.....	10
2.2. Signal Description.....	10
2.3. Attributes Summary.....	12
2.4. Interfacing with the Gamma Corrector.....	15
2.4.1. Parallel and Sequential Architectures.....	15
2.4.2. Valid Output.....	15
2.5. Timing Diagrams.....	15
2.5.1. Parallel Architecture Timing.....	15
2.5.2. Sequential Architecture Timing.....	16
2.5.3. Dynamically Loadable Gamma LUT.....	17
3. IP Generation, Simulation, and Validation.....	19
3.1. Generating the IP.....	19
3.2. Running Functional Simulation.....	22
3.3. IP Evaluation.....	24
4. Ordering Part Number.....	25
Appendix A. Resource Utilization.....	26
References.....	28
Technical Support Assistance.....	29
Revision History.....	30

Figures

Figure 2.1. Top-level Interface Diagram for the Gamma Corrector IP Core	8
Figure 2.2. Original, Correction and After Correction Response Curves.....	9
Figure 2.3 Parallel Architecture	16
Figure 2.4 Sequential Architecture	16
Figure 2.5. Example of Dynamic Gamma Value Programming, Three Color Planes, Sequential Mode.....	17
Figure 2.6. Example of Dynamic Gamma Value Programming, Three Color Planes, Parallel Mode.....	17
Figure 2.7. Functionality of Programmable Gamma through Look-Up Table (LUT)	18
Figure 2.8. Functionality of Constant Gamma Value (= 2.2)	18
Figure 3.1. Module/IP Block Wizard	19
Figure 3.2. Configure User Interface of Gamma Corrector IP Core	20
Figure 3.3. Check Generating Result.....	21
Figure 3.4. Simulation Wizard.....	22
Figure 3.5. Adding and Reordering Source	23
Figure 3.6. Simulation Waveform	23

Tables

Table 1.1. Quick Facts	6
Table 2.1. Gamma Corrector IP Core Signal Description	10
Table 2.2. Attributes Table	12
Table 2.3. Attributes Descriptions	14
Table 3.1. Generated File List	21
Table A.1. Resource Utilization	26
Table A.2. Resource Utilization	26
Table A.3. Resource Utilization	26
Table A.4. Resource Utilization	27
Table A.5. Resource Utilization	27

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
FPGA	Field Programmable Gate Array
RTL	Register Transfer Level
RAM	Random Access Memory
EBR	Embedded Block RAM
GUI	Graphical User Interface
CRT	Cathode Ray Tube

1. Introduction

Gamma correction is a kind of pre-distortion correction applied on images or video frames to offset the non-linear behavior of display systems. CRT (Cathode Ray Tube) displays exhibit inherent non-linearity in mapping voltage to light intensity. The intensity of the displays is proportional to some power, referred to as *gamma*, of the signal amplitude. Gamma is usually greater than one and hence the displays have lower gain at low intensities and progressively larger gain at higher intensities. Gamma correction involves multiplying the input signal with the inverse of the display transfer function to achieve a linear intensity response for the display with respect to the original input signal.

The non-CRT displays such as Plasma, LCOS (Liquid Crystal on Silicon), and DLP (Digital Light Processing) have different transfer characteristics. Several gamma correction methods and values are used in television and display systems. Sometimes, the display itself can have linear characteristics, but a gamma transformation (usually called *degamma*) may be required because of an earlier gamma correction made to the incoming signal.

The Lattice Gamma Corrector IP core is a widely parameterizable and multi-color plane gamma correction system. It can support almost any custom gamma correction requirement.

1.1. Quick Facts

Table 1.1 presents a summary of the Gamma Corrector IP Core.

Table 1.1. Quick Facts

IP Requirements	Supported FPGA Families	CrossLink™-NX, Certus™-NX, CertusPro™-NX, MachXO5™-NX, and Lattice Avant
Resource Utilization	Targeted Devices	LIFCL-40, LIFCL-17, LFD2NX-40, LFD2NX-17, LFCPNX-100, LFMXO5-25, and LAV-AT-500E
	Resources	See Table A.1 , Table A.2 , Table A.3 , Table A.4 , and Table A.5
Design Tool Support	Lattice Implementation	IP Core v1.x.x – Lattice Radiant™ Software 2.1 or later
	Synthesis	Lattice Synthesis Engine Synopsys® Synplify Pro® for Lattice
	Simulation	For a list of supported simulators, see the Lattice Radiant Software User Guide .

1.2. Features

The key features of the Gamma Corrector IP Core include:

- Gamma correction characteristics can be specified as an equation using a gamma value or by the actual mapping values of the look-up table
- Configurable number of color planes- 1 to 3
- Configurable number of bits per color plane – 4 to 12
- Gamma correction look-up table can be run-time programmable
- Gamma correction enable/disable control
- Option for sequential or parallel architecture for area or throughput trade-off
- Optimized gamma look-up table memory when same gamma correction is used for multiple color planes
- Registered input option for input set-up time improvement

1.3. Conventions

1.3.1. Nomenclature

The nomenclature used in this document is based on Verilog HDL.

1.3.2. Signal Names

Signal Names that end with:

- *_j* are input signals
- *_o* are output signals

1.3.3. Host

The logic unit inside the FPGA interacts with the Gamma Corrector IP Core.

1.3.4. Attribute

The names of attributes in this document are formatted in title case and italicized (*Attribute Name*).

2. Functional Description

2.1. Overview

Gamma corrector is essentially a look-up-table (referred as *gamma LUT* in this document) that maps each input value to a corresponding output value having the same width. Some applications require simultaneous gamma correction of all the color components, for example, R, G, and B (Red, Green and Blue) require simultaneous correction. There may also be a need to reduce the memory utilization by performing the conversions sequentially for each color plane. Gamma correction for multiple color planes can also be used for gamma correction of multiple monochrome channels either parallel or sequentially. It is also useful to have the ability to dynamically load the gamma LUT values into the gamma corrector.

Figure 2.1 shows the interface diagram for the Gamma Corrector IP. The diagram shows all the available ports. Note that not all the I/O ports are available for a chosen configuration.

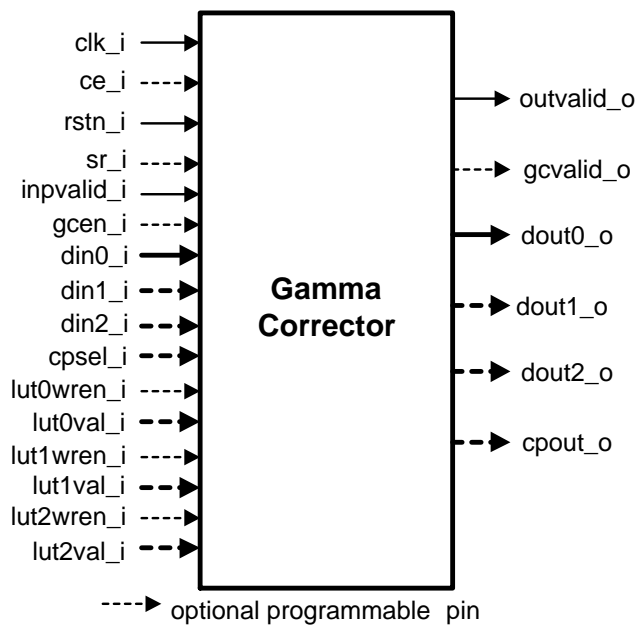


Figure 2.1. Top-level Interface Diagram for the Gamma Corrector IP Core

2.1.1. Gamma Correction Equation

Gamma correction is a non-linear exponent mapping applied to the normalized value of the input to result in a normalized corrected output. Gamma correction is defined by:

$$O_N = I_N^{\left(\frac{1}{\gamma}\right)} \quad (1)$$

where O_N is the normalized output and I_N is the normalized input, both normalized to the range [0,1] and γ is the gamma for the display that is being corrected.

The actual pixel mapping depends on the bit width of the pixel component. For a pixel width of b bits, we have:

$$O = \left(\frac{I_N}{2^b - 1}\right)^{1/\gamma} (2^b - 1) \quad (2)$$

Here O is the actual output pixel value represented as a b -bit binary number.

The original response of a CRT with gamma=2.2, gamma correction and the response of the CRT display for the gamma corrected input are shown in [Figure 2.2](#).

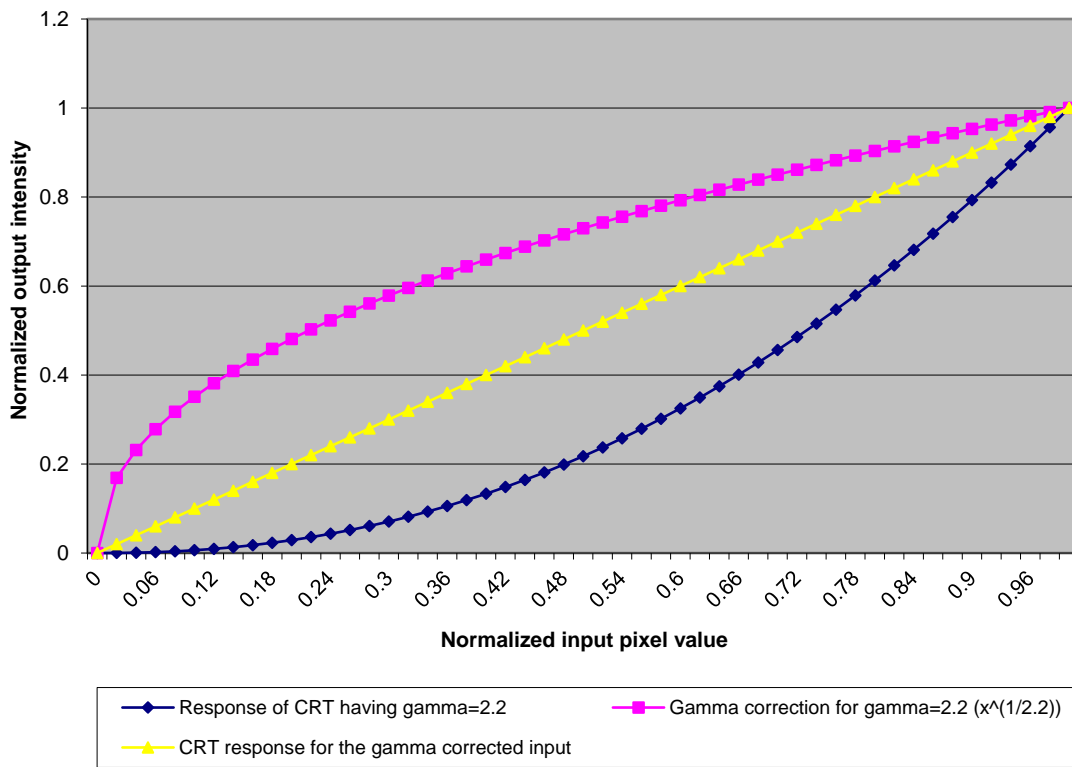


Figure 2.2. Original, Correction and After Correction Response Curves

2.1.2. Filling the Gamma LUT

Based on the above equation, the size of the gamma LUT is 2^b bits depth and b bits wide. The gamma LUT is, in most cases, pre-filled with the gamma correction value for each pixel value from 0 to 2^b-1 . The gamma correction function can be specified by the gamma value used in [Equation \(1\)](#) or the actual gamma mapping values for each input pixel value. This IP also supports reloading of the gamma LUT through input ports. It takes as many cycles as the number of locations in the gamma LUT to program it completely. While the LUTs are being programmed, gamma correction cannot take place. The IP core can have one or more gamma LUTs when used for mapping different color plane data or channels. If there are multiple gamma LUTs in the IP, for example, one for each of the 3 color planes, a selection input (cpsel_i) determines which color plane or channel is being processed at any time. The Gamma Corrector IP optimizes the number of physical gamma LUTs required depending on the value of the gamma for each color plane and the number of simultaneous i/o's (inputs/outputs) selected by the user.

2.1.3. Multi-Color Plane/Channel Mapping

This IP supports gamma correction of multiple color planes or channels and they are done either sequentially or in parallel. In sequential correction, only one color component data is mapped during a clock cycle. For example, if there are three color planes, the Gamma Corrector requires three clocks to process a color pixel. The input, cpsel_i, is used to identify which color plane LUT needs to be used for the current input. The same cpsel_i input can be used to time multiplex multiple channels (multiple displays) having different gamma correction requirements. In parallel correction, gamma correction of more than one color plane or channel is done simultaneously.

2.1.4. Dynamically Loadable Gamma LUTs

The gamma LUTs can be dynamically loaded or re-programmed after the IP is generated. During LUT programming, each of the possible input values is applied at the input port `dinx_i` and the corresponding correction value at the input `lutxval_i` while keeping the corresponding LUT write enable (`lutxwren_i`) signal high (x in the port names stands for 0,1 or 2). If the core is configured for the sequential mode, each of the LUTs corresponding to the color planes or channels is programmed sequentially. The value at the input port `cpse1_i` is used to identify the LUT that is programmed.

2.1.5. Handshake I/O ports

The input `gcen_i` is used to enable gamma correction. If this signal is low, the gamma correction is bypassed and the input value appears at the output after the usual latency for that configuration. The vector port `cpse1_i` identifies the color plane or channel number that is being processed or programmed. The output `cpout_o` gives out the color plane or channel number information for the current output(s) appearing at `doutx_o` (x stands for 0, 1, or 2) ports.

2.2. Signal Description

Table 2.1 lists the input and output signals for Gamma Corrector IP Core.

Table 2.1. Gamma Corrector IP Core Signal Description

Port	Bits	I/O	Description
All configurations			
<code>clk_i</code>	1	I	System clock (reference clock for input and output data).
<code>rstn_i</code>	1	I	System wide asynchronous active-low reset signal.
<code>inpsvald_i</code>	1	I	Input data valid. Indicates valid data is present on <code>din0_i</code> (also on <code>din1_i</code> and <code>din2_i</code> when present).
<code>din0_i</code>	4–12	I	Input Data. When the <i>sequential</i> architecture is selected, this port is used to give input data for all the color planes in sequence. When the <i>parallel</i> architecture is selected, this port is used to give input data for the first color plane.
<code>dout0_o</code>	4–12	O	Output Data. When the <i>sequential</i> architecture is selected, this port is used to give the output data for all the color planes in sequence. When the <i>parallel</i> architecture is selected, this port is used to give output data for the first color plane.
<code>outpsvald_o</code>	1	O	Output data valid. Indicates valid data is present on <code>dout0_o</code> (also on <code>dout1_o</code> and <code>dout2_o</code> when present). This valid data may correspond to the gamma corrected output or the bypassed input data, depending on the state of the <code>gcpsvald_o</code> signal.
When Add bypass function option is selected			
<code>gcen_i</code>	1	I	Gamma correction enable. This signal is valid only when <code>inpsvald_i</code> is asserted high. If <code>gcen_i</code> is high, gamma correction is performed for that input, otherwise no gamma correction is performed.
<code>gcpsvald_o</code>	1	O	Gamma corrected output. This signal is valid only when <code>outpsvald_o</code> is high. If <code>gcpsvald_o</code> is high, the output is a gamma corrected value, otherwise the output is a bypassed value, same as input. The output <code>gcpsvald_o</code> is a shifted version of the input signal <code>gcen_i</code> , the shift being equal to the output latency.
When parallel architecture is selected and Number of color planes is more than one			
<code>din1_i</code>	4–12	I	Input data for the second color plane.
<code>dout1_o</code>	4–12	O	Output data for the second color plane.
When parallel architecture is selected and Number of color planes is more than two			
<code>din2_i</code>	4–12	I	Input data for the third color plane.
<code>dout2_o</code>	4–12	O	Output data for the third color plane.
When first color plane gamma function is selected as programmable			
<code>lut0psvald_i</code>	4–12	I	Gamma value is programmed through <code>lut0psvald_i</code> port when: <ul style="list-style-type: none"> • <code>lut0pswren_i</code> is asserted high • Make all color plane same is unchecked • <i>Programmable Gamma LUT</i> is Checked on either Second Color plane or Third Color plane when <i>Number of color planes</i> is 2 or 3 and <i>Architecture</i> is

Port	Bits	I/O	Description
			<p>Sequential</p> <ul style="list-style-type: none"> • <i>Architecture</i> is Sequential and any one of the planes <i>Programmable Gamma LUT</i> is Checked • <i>Architecture</i> is Parallel and first color plane <i>Programmable Gamma LUT</i> is Checked <p>The value from lut0val_i port is written to the gamma LUT at the address provided at the din0_i input port.</p>
lut0wren_i	1	I	<p>Write enable for writing the gamma value for the first color plane when:</p> <ul style="list-style-type: none"> • <i>Make all color plane same</i> is unchecked • <i>Programmable Gamma LUT</i> is Checked on either Second Color plane or Third Color plane when Number of color planes is 2 or 3 and <i>Architecture</i> is Sequential. • <i>Architecture</i> is Sequential and any one of the planes <i>Programmable Gamma LUT</i> is Checked. • <i>Architecture</i> is Parallel and first color plane <i>Programmable Gamma LUT</i> is Checked.
When second color plane gamma function is selected as <i>programmable, parallel</i> architecture is selected and Number of color planes is more than 1			
lut1val_i	4–12	I	Gamma value is programmed through lut1val port when lut1wren_i is asserted high. The value from lut1val_i port is written to the gamma LUT at the address provided at the din1_i input port.
lut1wren_i	1	I	Write enable for writing the gamma value for the second color plane.
When third color plane gamma function is selected as <i>programmable, parallel</i> architecture is selected and Number of color planes is more than 2			
lut2val_i	4–12	I	Gamma value is programmed through lut2val_i port when lut2wren_i is asserted high. The value from lut2val_i port is written to the gamma LUT at the address provided at the din2_i input port.
lut2wren_i	1	I	Write enable for writing the gamma value for the second color plane.
When <i>sequential</i> architecture is selected			
cpsele_i	1–2	I	Input color plane select. The inputs at din0_i, lut0val_i, lut0wren_i, din1_i, lut1val_i, lut1wren_i, din2_i, lut2val_i, lut2wren_i apply to the color plane provided at cpsele_i.
cpout_o	1–2	O	Output color plane select. The output at dout0_o, dout1_o, dout2_o corresponds to the color plane provided at cpout_o.
Optional I/O			
ce_i	1	I	Clock Enable. While this is low, the core will ignore all other synchronous inputs and maintain its current state. This optional signal should be selected only when required as it leads to increased resource utilization.
sr_i	1	I	Synchronous Reset. This signal must be asserted for at least one clock period duration in order to re-initialize the core. After synchronous reset, all the internal registers are cleared and the outvalid_o goes low. This optional signal should be selected only when required as it leads to increased resource utilization.

2.3. Attributes Summary

The configurable attributes of the Gamma Corrector IP Core are shown in [Table 2.2](#) and are described in [Table 2.3](#). The attributes can be configured through the IP Catalog's Module/IP wizard of the Lattice Radiant software.

Table 2.2. Attributes Table

Attribute	Selectable Values	Default	Dependency on Other Attributes
Configuration			
Number of color planes	1, 2, 3	3	—
Make all color plane same	Checked, Unchecked	Checked	Editable when Number of color planes not equal to 1.
Architecture	Sequential, Parallel	Sequential	Editable when Number of color planes not equal to 1
First Color Plane			
Data width	4–12	8	—
Gamma function input method	Gamma Value, Through LUT file	Gamma Value	Selectable values depends on <i>Programmable Gamma LUT</i>
Gamma value	0.1–10.0	2.2	Enabled only when <i>Gamma function input method</i> is <i>Gamma value</i> .
Programmable Gamma LUT	Checked, Unchecked	Checked	—
Gamma function through LUT file	NA	NA	Enabled only when <i>Gamma function input method</i> is <i>Through LUT file</i> . The *.mem file used for all color planes should be the same when <i>Make all color planes same</i> = Checked and <i>Gamma function input method</i> = ThroughLUT file.
Second Color Plane			
Data width	4–12	8	Editable when <i>Make all color plane same</i> is disabled and when <i>Number of color planes</i> is greater than 1
Gamma function input method	Gamma Value, Through LUT file	Gamma Value	Editable when <i>Make all color plane same</i> is disabled and when <i>Number of color planes</i> is greater than 1. Selectable values depends on <i>Programmable Gamma LUT</i>
Gamma value	0.1–10.0	2.2	Editable when <i>Make all color plane same</i> is disabled and when <i>Number of color planes</i> is greater than 1
Programmable Gamma LUT	Checked, Unchecked	Checked	Editable when <i>Make all color plane same</i> is disabled and when <i>Number of color planes</i> is greater than 1
Gamma function through LUT file	NA	NA	Editable when <i>Make all color plane same</i> is disabled and when <i>Number of color planes</i> is greater than 1. Enabled only when <i>Gamma function input method</i> is <i>Through LUT file</i> . The *.mem file used for all color planes should be the same when <i>Make all color planes same</i> = Checked and <i>Gamma function input method</i> = ThroughLUT file.

Third Color Plane			
Data width	4–12	8	Editable when <i>Make all color plane same</i> is disabled and when <i>Number of color planes</i> is greater than 2
Gamma function input method	Gamma Value, Through LUT file	Gamma Value	Editable when <i>Make all color plane same</i> is disabled and when <i>Number of color planes</i> is greater than 2. Selectable values depends on <i>Programmable Gamma LUT</i>
Gamma value	0.1–10.0	2.2	Editable when <i>Make all color plane same</i> is disabled and when <i>Number of color planes</i> is greater than 2
Programmable Gamma LUT	Checked, Unchecked	Checked	Editable when <i>Make all color plane same</i> is disabled and when <i>Number of color planes</i> is greater than 2
Gamma function through LUT file	NA	NA	Editable when <i>Make all color plane same</i> is disabled and when <i>Number of color planes</i> is greater than 2. Enabled only when <i>Gamma function input method</i> is <i>Through LUT file</i> . The *.mem file used for all color planes should be the same when <i>Make all color planes same</i> = Checked and <i>Gamma function input method</i> = ThroughLUT file.
Implementation			
Add Bypass Function	Checked, Unchecked	Checked	—
Registered Input	Checked, Unchecked	Checked	—
Memory Type	EBR, Distributed	EBR	—
Optional Input and Output Ports			
ce	Checked, Unchecked	Unchecked	—
sr	Checked, Unchecked	Unchecked	—
Output Latency	3–6	6	Not editable. The value depends on <i>Number of color planes</i> , <i>Architecture</i> , <i>Add bypass function</i> , and <i>Registered input</i> .

Table 2.3. Attributes Descriptions

Attribute	Description
Configuration	
Number of color planes	Number of color planes for gamma correction
Make all color plane same	Selecting this option will make all the other color plane parameters the same as those of the first color plane.
Architecture	Selects between parallel and sequential implementation architectures
First Color plane	
Data width	The bit width for the color plane
Gamma function input method	The method for specifying the gamma function. The function can be specified by the value of the gamma in the gamma correction equation or by the actual gamma mapping values for the all the pixel values in the input range.
Gamma value	This gamma value is used to create the gamma LUT using Equation (1) . This parameter is available when Gamma function input method is selected as <i>Gamma value</i>
Programmable Gamma LUT	This parameter is used to indicate if the gamma LUT is also programmable through the input port.
Gamma function through LUT file	This browse button is enabled when Gamma function input method is set to <i>Through LUT file</i> . The gamma LUT values will be read from the text file specified. The *.mem file used for all color planes should be the same when <i>Make all color planes same</i> = Checked and <i>Gamma function input method</i> = ThroughLUT file.
Second Color plane	
Data width	The bit width for the color plane
Gamma function input method	The method for specifying the gamma function. The function can be specified by the value of the gamma in the gamma correction equation or by the actual gamma mapping values for the all the pixel values in the input range.
Gamma value	This gamma value is used to create the gamma LUT using Equation (1) . This parameter is available when Gamma function input method is selected as <i>Gamma value</i>
Programmable Gamma LUT	This parameter is used to indicate if the gamma LUT is also programmable through the input port.
Gamma function through LUT file	This browse button is enabled when Gamma function input method is set to <i>Through LUT file</i> . The gamma LUT values will be read from the text file specified. The *.mem file used for all color planes should be the same when <i>Make all color planes same</i> = Checked and <i>Gamma function input method</i> = ThroughLUT file.
Third Color plane	
Data width	The bit width for the color plane
Gamma function input method	The method for specifying the gamma function. The function can be specified by the value of the gamma in the gamma correction equation or by the actual gamma mapping values for the all the pixel values in the input range.
Gamma value	This gamma value is used to create the gamma LUT using Equation (1) . This parameter is available when Gamma function input method is selected as <i>Gamma value</i>
Programmable Gamma LUT	This parameter is used to indicate if the gamma LUT is also programmable through the input port.
Gamma function through LUT file	This browse button is enabled when Gamma function input method is set to <i>Through LUT file</i> . The gamma LUT values are read from the text file specified. The *.mem file used for all color planes should be the same when <i>Make all color planes same</i> = Checked and <i>Gamma function input method</i> = ThroughLUT file.
Implementation	
Add Bypass Function	Selecting this option will add the dynamic gamma correction bypass functionality. Input port <i>gcen_i</i> and output port <i>gvalid_i</i> are added to the Gamma Corrector IP.
Registered Input	The inputs are registered if this option is selected. The core inputs' set-up times will improve by registering the inputs. This option is useful when the input data is provided on the device pins.
Memory Type	This parameter influences the type of memory used to implement the gamma LUT. If EBR option is selected then the device's EBR (Embedded Block RAM) resources are used for the gamma LUT if the data width is greater than 4. If Distributed option is selected, distributed memory (realized using FPGA's LUTs) is used for the gamma LUT if data width is less than 9.

Attribute	Description
Optional Input and output ports	
ce	Optional clock enable input port ce_i is added to the IP core if this option is checked.
sr	Optional clock enable input port sr_i is added to the IP core if this option is checked.
Output Latency	This static display shows the output latency for the selected core configuration.

2.4. Interfacing with the Gamma Corrector

2.4.1. Parallel and Sequential Architectures

The Gamma Corrector IP offers the choice of two different architectures: parallel and sequential. In the parallel architecture, all the color plane data are applied at the same time. The output data for all the color planes are also available at the same time after a latency of a few clock cycles. In the sequential architecture, the input data for the color planes is applied in sequence, one after the other, using the same input port din0_i. The output data for the color planes is given out sequentially using the same output port dout0_o after a latency of a few clock cycles.

When sequential architecture is selected, the input port din0_i and output port dout0_o are shared between all the color planes. If the data widths are not the same for all color planes, then the highest data width must correspond to the first color plane. When the data width is less than the input port (din0_i and lut0val_i) size, the data must be left aligned and the unused LSBs must be driven with zeros. Similarly, the output data is left aligned with unused LSBs driven to zero, when the data width is less than the output port (dout0_o) size. The color plane or channel number provided on cpsel_i can be in order, allowing the processing of multi-rate channels.

2.4.2. Valid Output

The data output of the Gamma Corrector IP is valid after the output latency for the selected configuration and is indicated by outvalid_i going high. Output latency for Gamma Corrector IP, defined as the number of clock cycles between the sampling of the input data and the availability of the gamma corrected data at the output port, is from 3 to 6 clock cycles depending on the parameters selected.

2.5. Timing Diagrams

2.5.1. Parallel Architecture Timing

Figure 2.3 shows the timing diagram for the parallel architecture. The input data for all the color planes are applied simultaneously on the input ports din0_i, din1_i, and din2_i.

The signal inval_i is asserted to indicate a valid input data present on the input ports. After a latency of a few cycles, the output data for all the color planes appear on the output ports dout0_o, dout1_o, and dout2_o. The signal outvalid_o is asserted to indicate a valid output data present on the output ports. If signal gcn_i is asserted then the data from the input ports is gamma corrected and given at the output. A gamma corrected output at the output ports is indicated by a high gvalid_o signal. If the signal gcn_i is not asserted, the data from the input ports is passed on directly to the output ports. This is indicated by a low gvalid_o signal at the output.

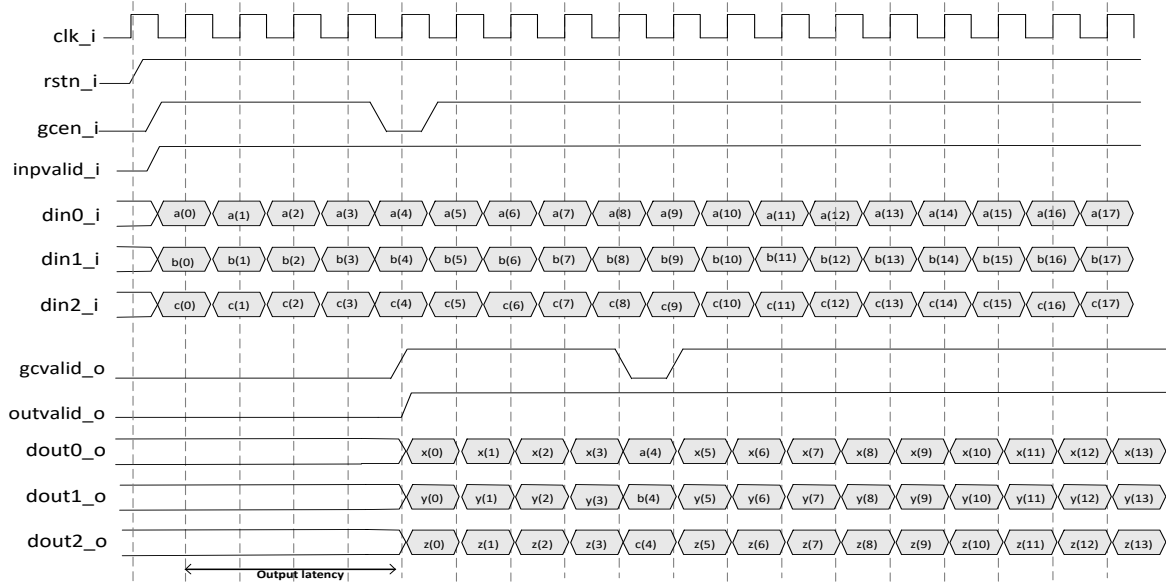


Figure 2.3 Parallel Architecture

2.5.2. Sequential Architecture Timing

Figure 2.4 shows the input and output signal timing for the sequential architecture. The input data for the three color planes are applied in sequence at the input port `din0_i`. The signal `inval_i` is asserted to indicate a valid data on `din0_i`. After a latency of a few cycles the output data for the first color plane appears on the output port `dout0_o`. In the following two cycles, the second and third color plane data appear on `dout0_o`. The signal `outvalid_o` is asserted to indicate a valid data on `dout0_o`.

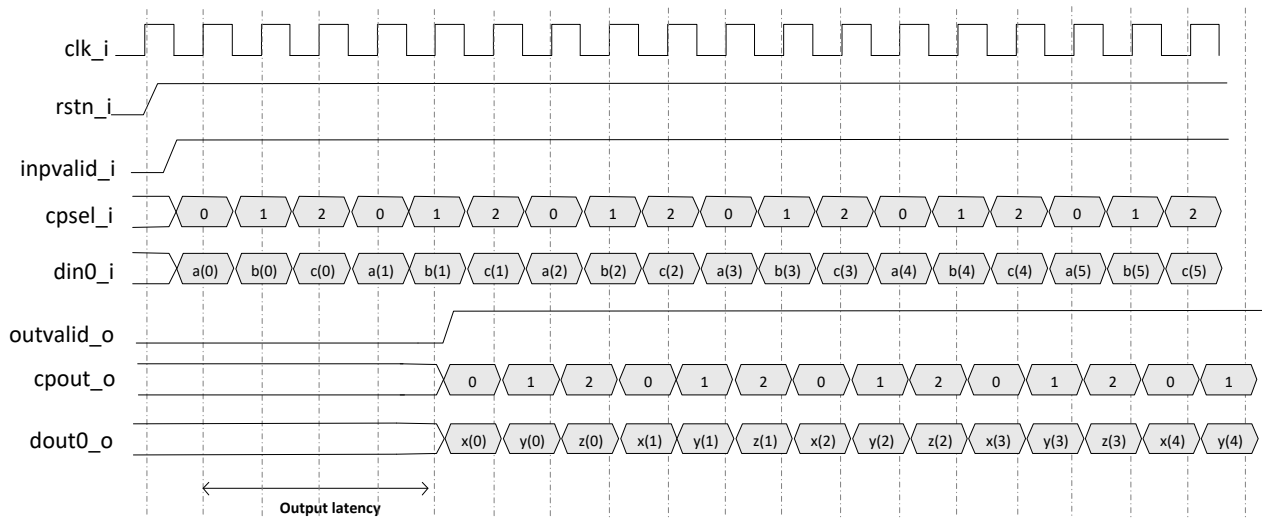


Figure 2.4 Sequential Architecture

2.5.3. Dynamically Loadable Gamma LUT

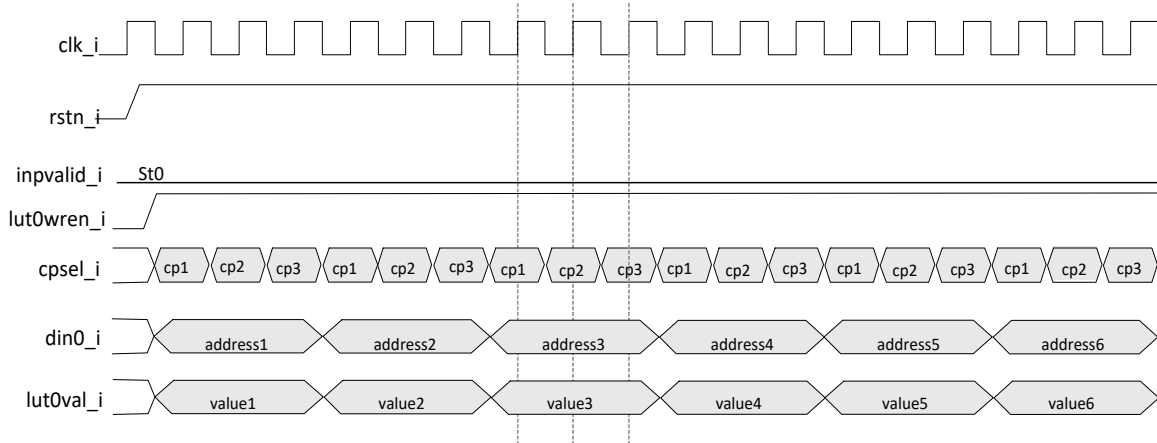


Figure 2.5. Example of Dynamic Gamma Value Programming, Three Color Planes, Sequential Mode

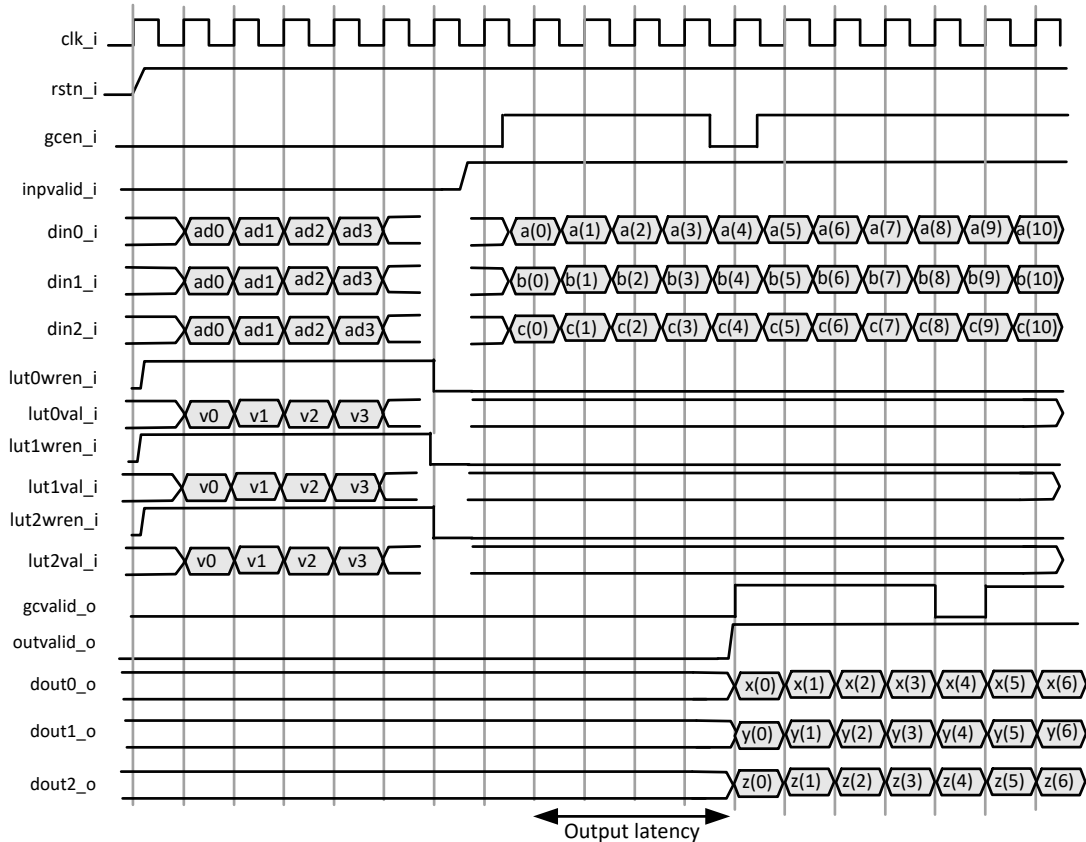


Figure 2.6. Example of Dynamic Gamma Value Programming, Three Color Planes, Parallel Mode

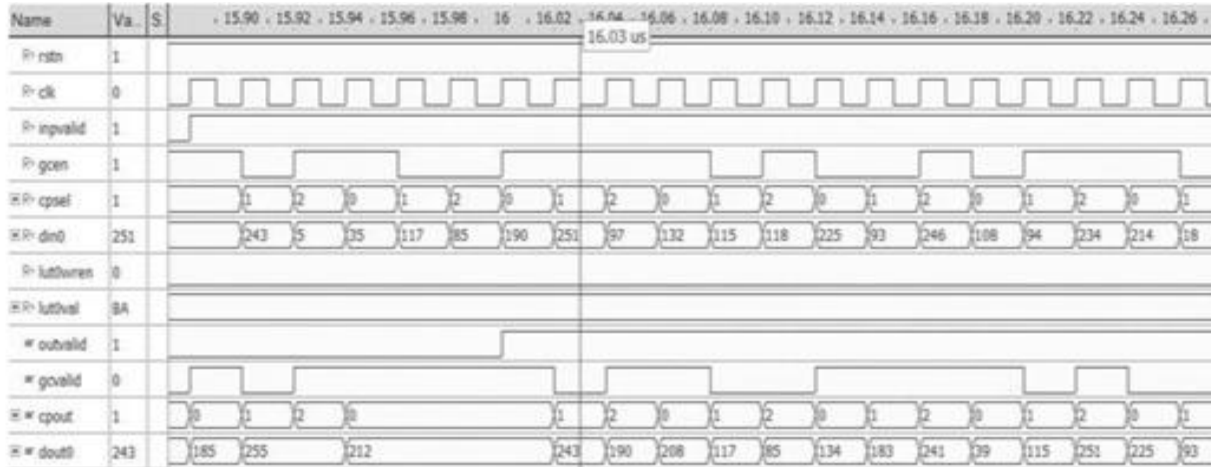


Figure 2.7. Functionality of Programmable Gamma through Look-Up Table (LUT)

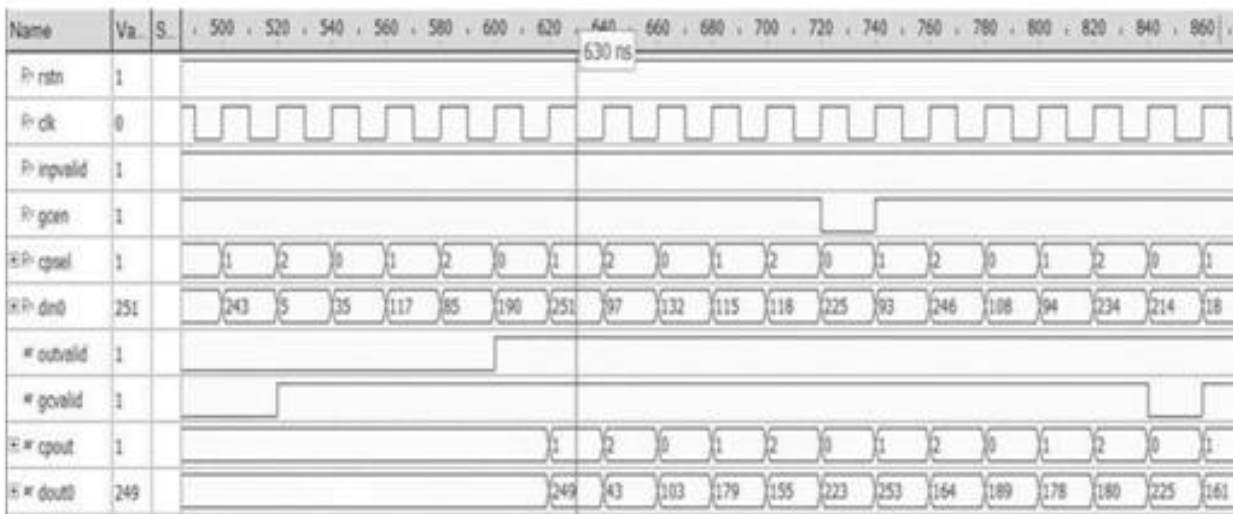


Figure 2.8. Functionality of Constant Gamma Value (= 2.2)

3. IP Generation, Simulation, and Validation

This section provides information on how to generate the IP Core using the Lattice Radiant software and how to run simulation and synthesis. For more details on the Lattice Radiant software, refer to the Lattice Radiant Software User Guide.

3.1. Generating the IP

The Lattice Radiant software allows user to customize and generate modules and IPs and integrate them into the device’s architecture. The procedure for generating the Gamma Corrector IP Core in Lattice Radiant software is described below.

To generate the Gamma Corrector IP Core:

1. Create a new Lattice Radiant software project or open an existing project
2. In the **IP Catalog** tab, double-click on **Gamma Corrector** under **IP, DSP** category. The **Module/IP Block Wizard** opens as shown in [Figure 3.1](#). Enter values in the **Component name** and the **Create in** fields and click **Next**.

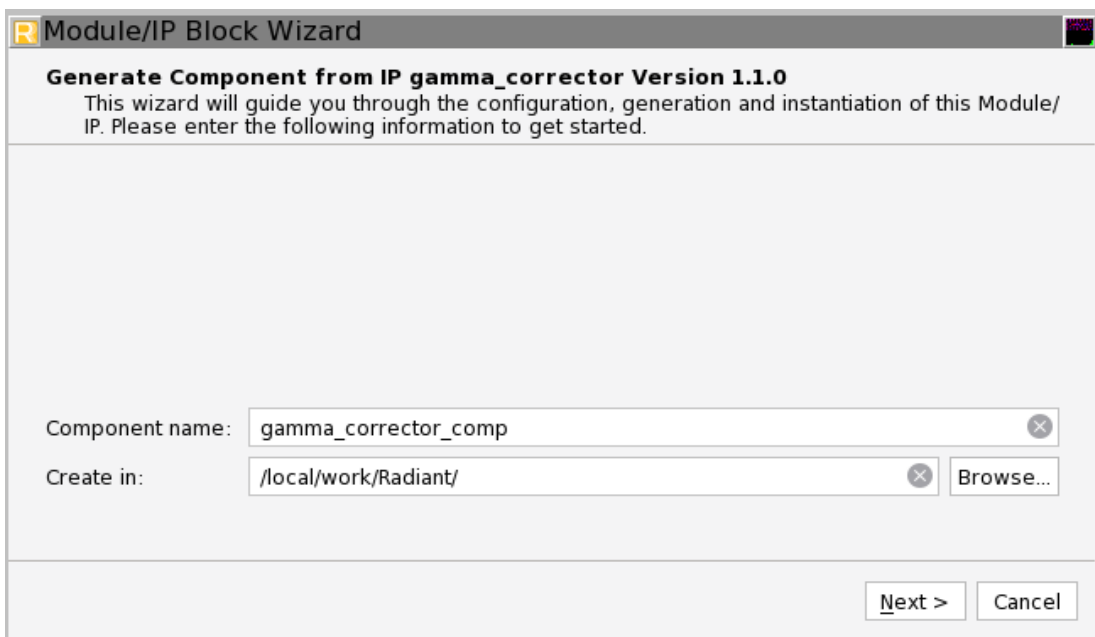


Figure 3.1. Module/IP Block Wizard

- In the module's dialog box of the **Module/IP Block Wizard** window, customize the selected Gamma Corrector IP Core using drop-down menus and check boxes. As a sample configuration, see [Figure 3.2](#). For configuration options, see the [Attributes Summary](#) section.

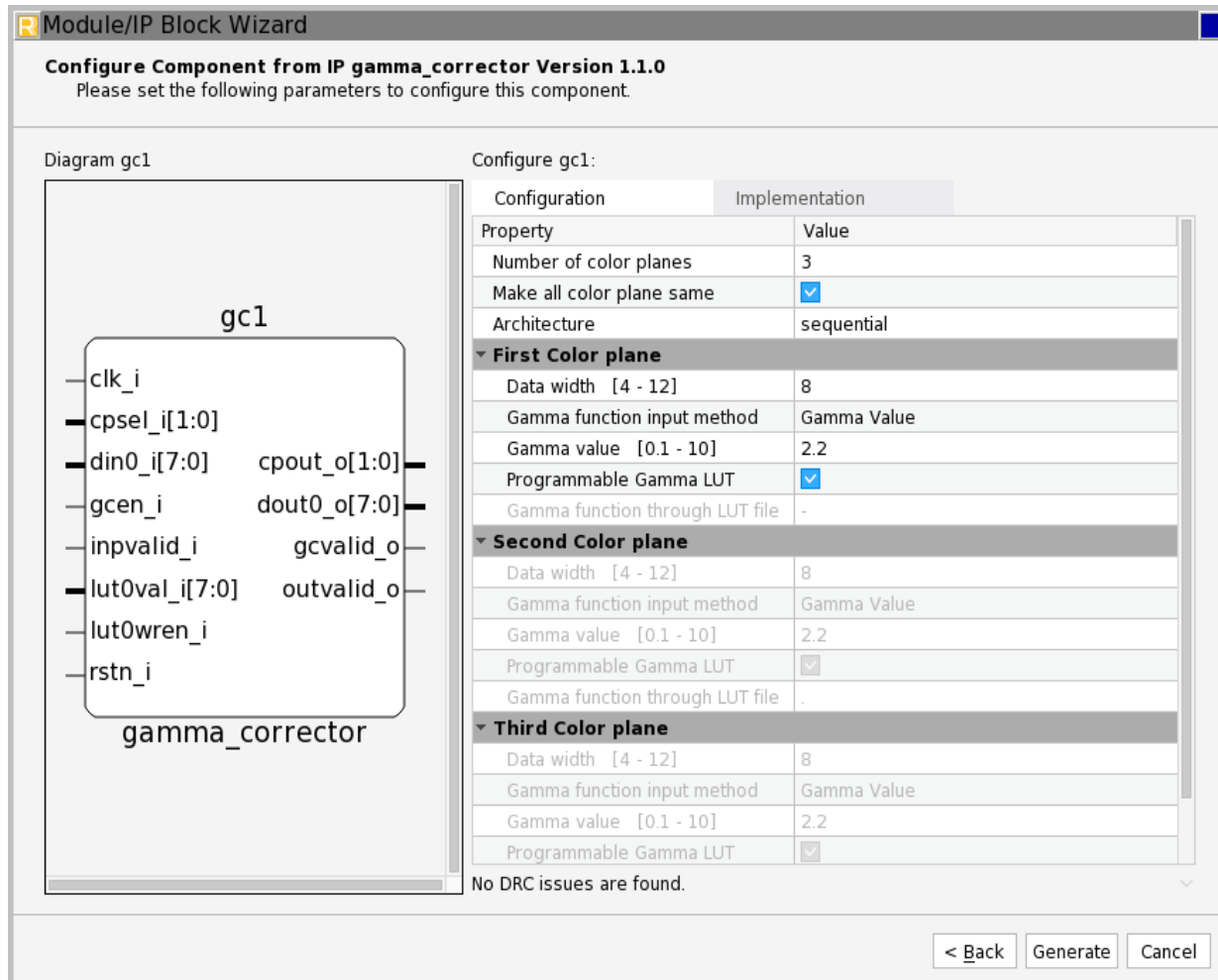


Figure 3.2. Configure User Interface of Gamma Corrector IP Core

- Click **Generate**. The **Check Generating Result** dialog box opens, showing design block messages and results as shown in [Figure 3.3](#).

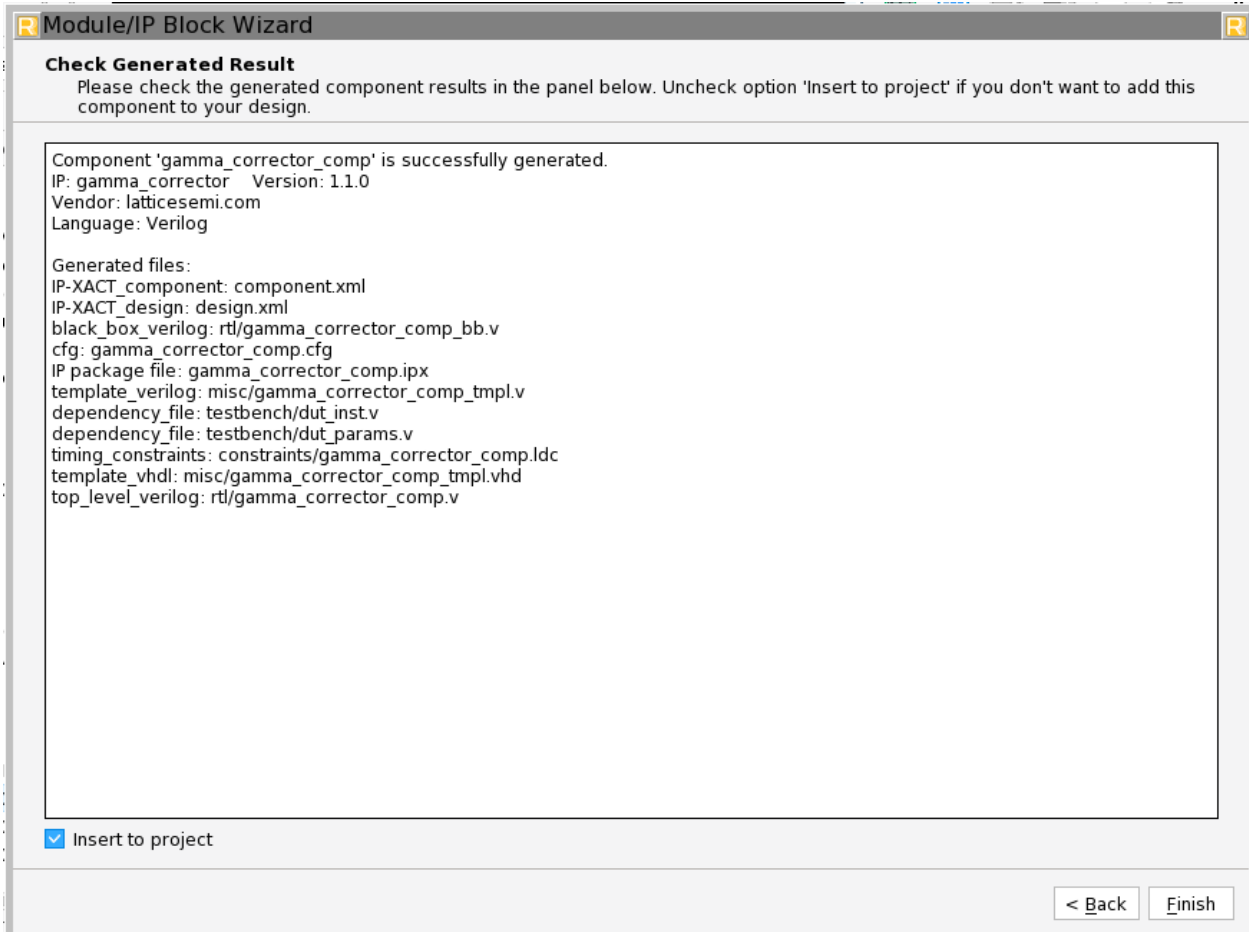


Figure 3.3. Check Generating Result


- Click the **Finish** button. All the generated files are placed under the directory paths in the **Create in** and the **Component name** fields shown in [Figure 3.1](#).

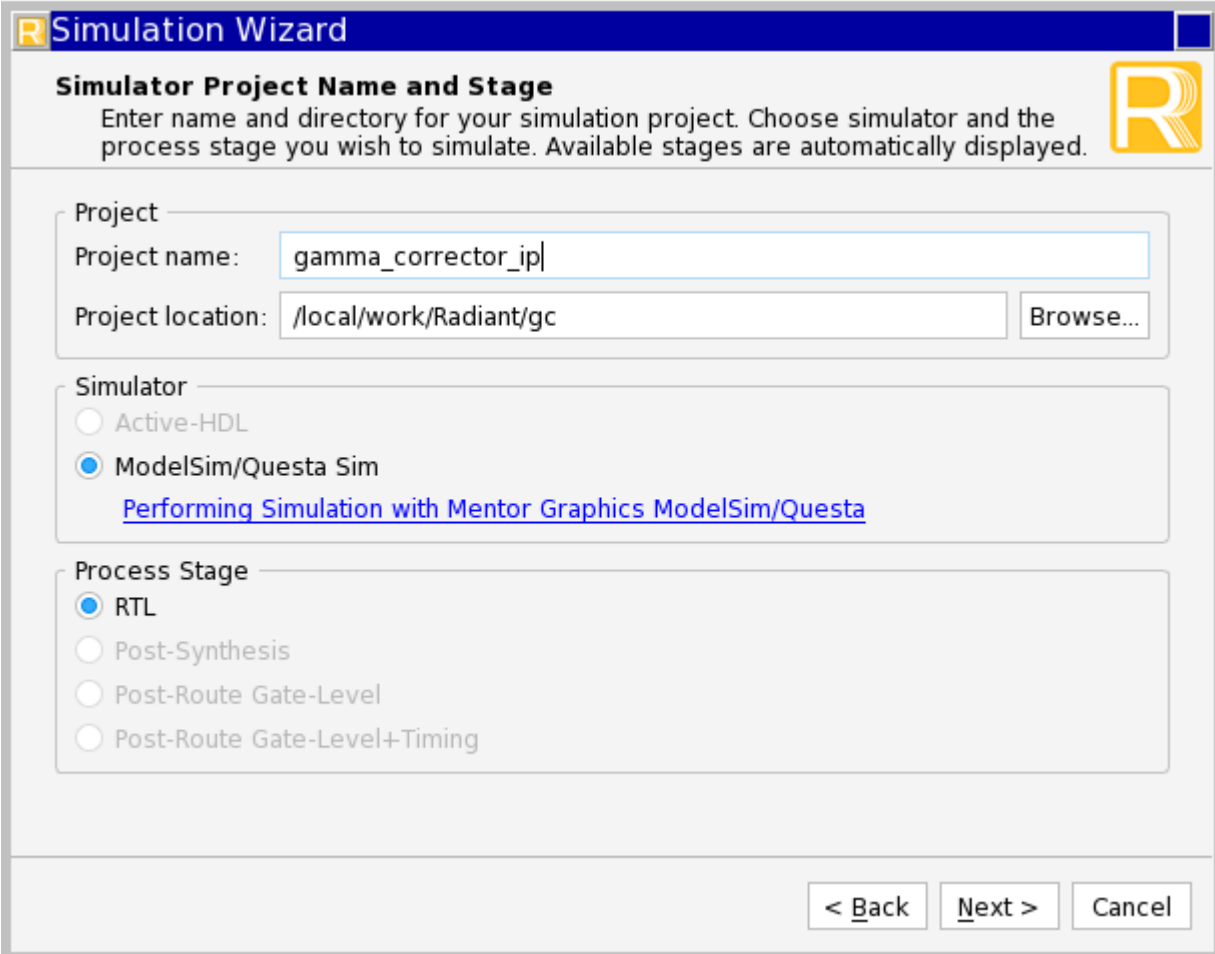
The generated Gamma Corrector IP Core package includes the black box (<Component name>_bb.v) and instance templates (<Component name>_tmpl.v/vhd) that can be used to instantiate the core in a top-level design. An example RTL top-level reference source file (<Component name>.v) that can be used as an instantiation template for the IP core is also provided. User may also use this top-level reference as the starting template for the top-level for their complete design. The generated files are listed in [Table 3.1](#).

Table 3.1. Generated File List

Attribute	Description
<Component name>.ipx	This file contains the information on the files associated to the generated IP.
<Component name>.cfg	This file contains the parameter values used in IP configuration.
component.xml	Contains the ipxact:component information of the IP.
design.xml	Documents the configuration parameters of the IP in IP-XACT 2014 format.
rtl/<Component name>.v	This file provides an example RTL top file that instantiates the IP core.
rtl/<Component name>_bb.v	This file provides the synthesis black box.
misc/<Component name>_tmpl.v misc /<Component name>_tmpl.vhd	These files provide instance templates for the IP core.

3.2. Running Functional Simulation

1. Click the  button located on the **Toolbar** to initiate the **Simulation Wizard** shown in [Figure 3.4](#).



Simulation Wizard

Simulator Project Name and Stage
Enter name and directory for your simulation project. Choose simulator and the process stage you wish to simulate. Available stages are automatically displayed.

Project

Project name:

Project location:

Simulator

Active-HDL

ModelSim/Questa Sim

[Performing Simulation with Mentor Graphics ModelSim/Questa](#)

Process Stage

RTL

Post-Synthesis

Post-Route Gate-Level

Post-Route Gate-Level+Timing

Figure 3.4. Simulation Wizard

2. Click **Next** to open the **Add and Reorder Source** window as shown in [Figure 3.5](#).

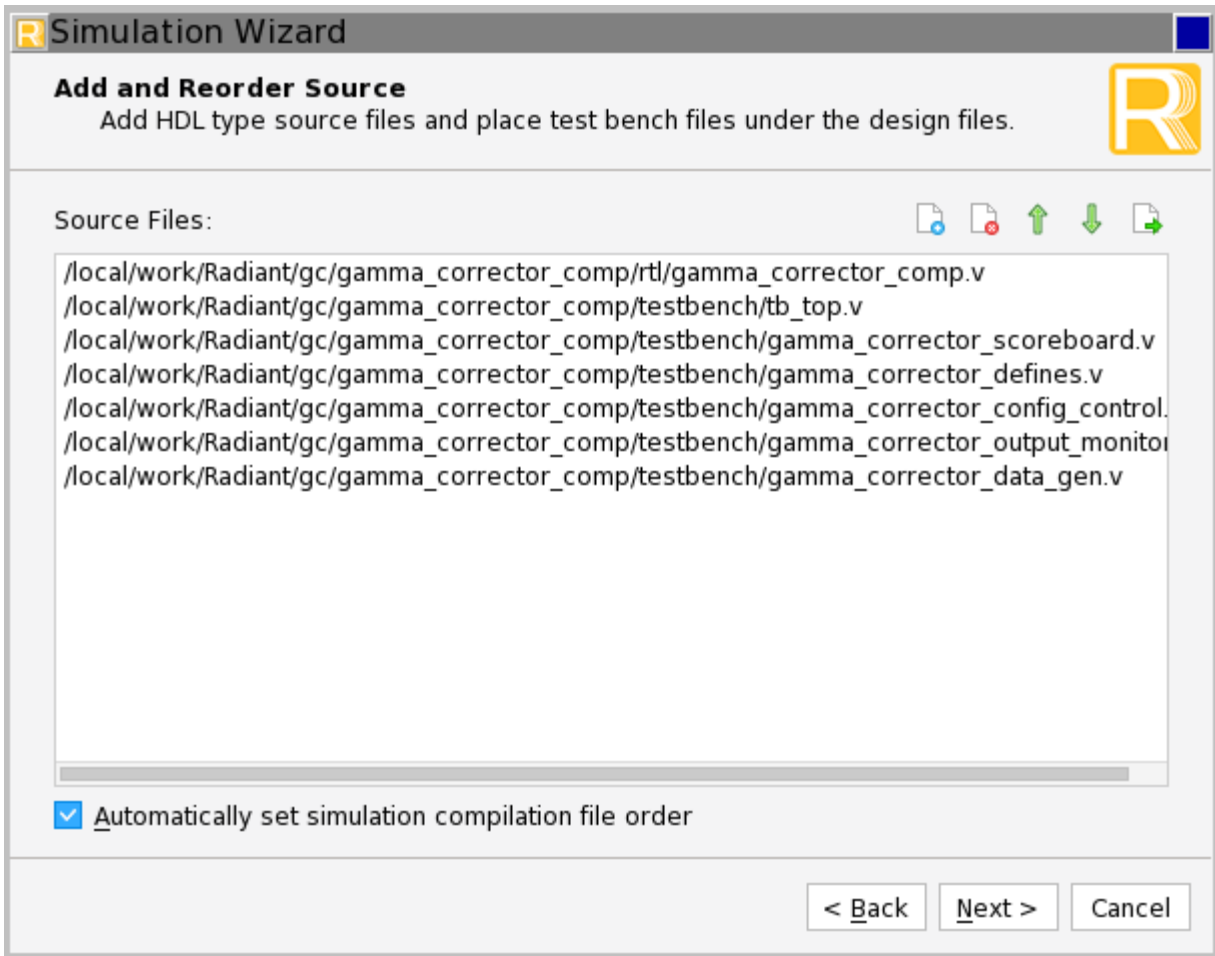


Figure 3.5. Adding and Reordering Source

3. Click **Next**. The **Summary** window is shown. Click **Finish** to run the simulation.

Note: It is necessary to follow the procedure above until it is fully automated in the Lattice Radiant software suite.

The results of the simulation in our example are provided [Figure 3.6](#).

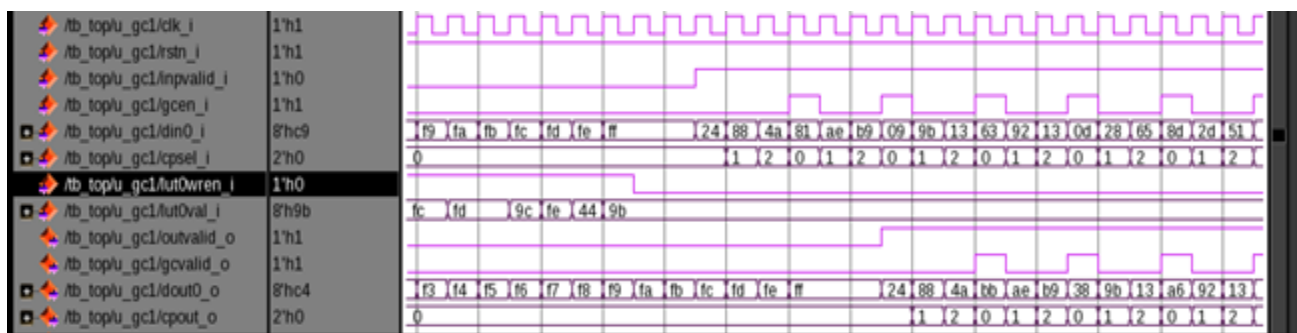


Figure 3.6. Simulation Waveform

3.3. IP Evaluation

The IP Core supports Lattice’s IP evaluation capability when used in the supported FPGA family and targeted device. This makes it possible to create versions of the IP core that operate in hardware for a limited period of time (approximately four hours) without requiring the purchase of an IP license. It may also be used to evaluate the core in hardware in user-defined designs. The IP evaluation capability may be enabled/disabled in the Strategy dialog box. It is disabled by default. To change this setting, go to Project > Active Strategy > Bitstream Settings.

4. Ordering Part Number

The Ordering Part Number (OPN) for this IP Core are the following:

- GAMMA-CNX-U – Gamma Corrector for CrossLink-NX - Single Design License
- GAMMA-CNX-UT – Gamma Corrector for CrossLink-NX - Site License
- GAMMA-CNX-US - Gamma Corrector for CrossLink-NX - 1 Year Subscription License
- GAMMA-CTNX-U – Gamma Corrector for Certus-NX - Single Design License
- GAMMA-CTNX-UT – Gamma Corrector for Certus-NX - Site License
- GAMMA-CTNX-US – Gamma Corrector for Certus-NX - 1 Year Subscription License
- GAMMA-CPNX-U – Gamma Corrector for CertusPro-NX - Single Design License
- GAMMA-CPNX-UT – Gamma Corrector for CertusPro-NX - Site License
- GAMMA-CPNX-US – Gamma Corrector for CertusPro-NX - 1 Year Subscription License
- GAMMA-XO5-U – Gamma Corrector for MachXO5-NX - Single Design License
- GAMMA-XO5-UT – Gamma Corrector for MachXO5-NX - Site License
- GAMMA-XO5-US – Gamma Corrector for MachXO5-NX - 1 Year Subscription License
- GAMMA-AVE-U – Gamma Corrector for Avant-E - Single Design License
- GAMMA-AVE-UT – Gamma Corrector for Avant-E - Site License
- GAMMA-AVE-US – Gamma Corrector for Avant-E - 1 Year Subscription License

Appendix A. Resource Utilization

Table A.1 shows configuration and resource utilization for LIFCL-40-9BG400I using Synplify Pro of Lattice Radiant software 2.1 or later.

Table A.1. Resource Utilization

Configuration	Clk Fmax (MHz)*	Registers	LUTs	EBRs	Programmable I/O
Configuration 1: Sequential architecture, 3 color planes, same color planes	200	130	27	2	35
Configuration 2: Parallel architecture, 3 color planes, same color planes	200	223	41	3	81
Configuration 3: Sequential architecture, 3 color planes, different color planes	200	132	31	3	35

Table A.2 shows configuration and resource utilization for LIFCL-17-7BG256I using Synplify Pro of Lattice Radiant software 2.1 or later.

Table A.2. Resource Utilization

Configuration	Clk Fmax (MHz)*	Registers	LUTs	EBRs	Programmable I/O
Configuration 1: Sequential architecture, 3 color planes, same color planes	200	130	27	2	35
Configuration 2: Parallel architecture, 3 color planes, same color planes	200	223	41	3	81
Configuration 3: Sequential architecture, 3 color planes, different color planes	200	132	31	3	35

Table A.3 shows configuration and resource utilization for LFD2NX-40-8BG256I using Synplify Pro of Lattice Radiant software 2.1 or later.

Table A.3. Resource Utilization

Configuration	Clk Fmax (MHz)*	Registers	LUTs	EBRs	Programmable I/O
Configuration 1: Sequential architecture, 3 color planes, same color planes	200	130	27	2	35
Configuration 2: Parallel architecture, 3 color planes, same color planes	200	223	41	3	81
Configuration 3: Sequential architecture, 3 color planes, different color planes	200	132	31	3	35

***Note:** Fmax is generated when the FPGA design only contains Gamma Corrector IP Core and the target Frequency is 100 MHz. These values may be reduced when user logic is added to the FPGA design.

Table A.4 shows configuration and resource utilization for LFMX05-25-9BBG400I using Synplify Pro of Lattice Radiant Software. Default configuration is used, and some attributes are changed from the default value to show the effect on the resource utilization.

Table A.4. Resource Utilization

Configuration	Clk Fmax (MHz)*	Registers	LUTs	EBRs
Configuration 1: Sequential architecture, 3 color planes, same color planes	200	105	19	1
Configuration 2: Parallel architecture, 3 color planes, same color planes	200	205	73	3
Configuration 3: Sequential architecture, 3 color planes, different color planes	200	105	19	1

***Note:** Fmax is generated when the FPGA design only contains Gamma Corrector IP Core and the target Frequency is 100 MHz. These values may be reduced when user logic is added to the FPGA design.

Table A.5 shows configuration and resource utilization for LAV-AT-500E-3LFG1156I using Synplify Pro of Lattice Radiant Software. Default configuration is used, and some attributes are changed from the default value to show the effect on the resource utilization.

Table A.5. Resource Utilization

Configuration	Clk Fmax (MHz)*	Registers	LUTs	EBRs
Configuration 1: Sequential architecture, 3 color planes, same color planes	201.45	97	18	1
Configuration 2: Parallel architecture, 3 color planes, same color planes	201.45	181	72	3
Configuration 3: Sequential architecture, 3 color planes, different color planes	201.45	97	18	1

***Note:** Fmax is generated when the FPGA design only contains Gamma Corrector IP Core and the target Frequency is 200 MHz. These values may be reduced when user logic is added to the FPGA design.

References

- <https://www.latticesemi.com/en/Products/FPGAandCPLD/CrossLink-NX>
- <https://www.latticesemi.com/en/Products/FPGAandCPLD/Certus-NX>
- <https://www.latticesemi.com/Products/FPGAandCPLD/CertusPro-NX>

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

Revision History

Revision 1.2, November 2022

Section	Change Summary
Introduction	<ul style="list-style-type: none"> Added MachXO5™-NX and Lattice Avant families in Table 1.1. Quick Facts. Add LFMXO5-25 and LAV-AT-500E devices in Table 1.1. Quick Facts.
IP Generation, Simulation, and Validation	<ul style="list-style-type: none"> Updated title of Section 3 from <i>Core Generation, Simulation, and Validation</i> to <i>IP Generation, Simulation, and Validation</i>. Deleted the <i>Licensing the IP</i> section. Updated Section 3.1 from <i>Generation and Synthesis</i> to <i>Generating the IP</i>.
Ordering Part Number	Updated OPNs for Gamma Corrector.
Resource Utilization	Added Table A.4. Resource Utilization and Table A.5. Resource Utilization .

Revision 1.1, June 2021

Section	Change Summary
Introduction	Updated Table 1.1 to add CertusPro-NX support.
Ordering Part Number	Added this section.
Appendix A. Resource Utilization	Updated section content.
References	Added CertusPro-NX web page.

Revision 1.0, December 2020

Section	Change Summary
All	Initial release



www.latticesemi.com