# Customizable HSB Sensor Interfaces on Lattice CertusPro-NX Platforms

## Reference Design

FPGA-RD-02326-1.7

September 2026

## 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, with all faults, and all associated risk is the responsibility entirely of the Buyer. The information provided herein is for informational purposes only and may contain technical inaccuracies or omissions, and may be otherwise rendered inaccurate for many reasons, and Lattice assumes no obligation to update or otherwise correct or revise this information. 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. LATTICE PRODUCTS AND SERVICES ARE NOT DESIGNED, MANUFACTURED, OR TESTED FOR USE IN LIFE OR SAFETY CRITICAL SYSTEMS, HAZARDOUS ENVIRONMENTS, OR ANY OTHER ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE, INCLUDING ANY APPLICATION IN WHICH THE FAILURE OF THE PRODUCT OR SERVICE COULD LEAD TO DEATH, PERSONAL INJURY, SEVERE PROPERTY DAMAGE OR ENVIRONMENTAL HARM (COLLECTIVELY, "HIGH-RISK USES"). FURTHER, BUYER MUST TAKE PRUDENT STEPS TO PROTECT AGAINST PRODUCT AND SERVICE FAILURES, INCLUDING PROVIDING APPROPRIATE REDUNDANCIES, FAIL-SAFE FEATURES, AND/OR SHUT-DOWN MECHANISMS. LATTICE EXPRESSLY DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS OF THE PRODUCTS OR SERVICES FOR HIGH-RISK USES. 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.

## Inclusive Language

This document was created consistent with Lattice Semiconductor’s inclusive language policy. In some cases, the language in underlying tools and other items may not yet have been updated. Please refer to Lattice’s inclusive language FAQ 6878 for a cross reference of terms. Note in some cases such as register names and state names it has been necessary to continue to utilize older terminology for compatibility.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# Contents

Contents ..... 3
Abbreviations in This Document ..... 9
1. Introduction ..... 11
1.1. Reference Design: Quick Facts ..... 13
1.1.1. CertusPro-NX Versa Board with Jetson AGX Orin/Thor ..... 13
1.1.2. CertusPro-NX Sensor-to-Ethernet Bridge Board with Jetson AGX Orin ..... 14
1.2. Quick-Start Guides ..... 15
1.3. Features ..... 15
1.4. Naming Conventions ..... 16
1.4.1. Nomenclature ..... 16
1.4.2. Signal Names ..... 16
2. Directory Structure and File Overview ..... 17
3. Functional Description ..... 19
3.1. NVIDIA Host Developer Kits ..... 19
3.1.1. Jetson AGX Orin Developer Kit ..... 19
3.1.2. Jetson AGX Thor Developer Kit ..... 20
3.2. Lattice-Powered Holoscan Sensor Bridge Platforms ..... 21
3.2.1. CertusPro-NX Versa Board ..... 21
3.2.2. CertusPro-NX Sensor-to-Ethernet Bridge Board ..... 22
3.3. Hololink IP and Customizable Interfaces ..... 22
3.3.1. Overview ..... 22
3.3.2. Sensor Interface ..... 23
3.3.3. Host Interface through Ethernet Link ..... 27
3.4. Multi Camera Frame Synchronization ..... 29
3.5. Dual-Boot Feature ..... 29
3.6. Clocking and Reset Scheme ..... 30
3.6.1. CertusPro-NX Versa Board ..... 30
3.6.2. CertusPro-NX Sensor-to-Ethernet Bridge Board ..... 33
4. Customizing the Reference Design ..... 34
4.1. Customization Steps for the Reference Design ..... 34
4.1.1. Selecting the Holoscan Sensor Bridge Board ..... 34
4.1.2. Generating the IP ..... 34
4.1.3. Configuring the HDL Parameter and DEFINE Switch ..... 44
4.1.4. Configuring the Platform Design Constraint ..... 46
4.1.5. Verifying the Reference Design Configuration ..... 48
4.1.6. Configuring the MIPI CSI-2 D-PHY IP at Runtime ..... 49
4.1.7. Enabling ISP or MJPEG Feature at Runtime ..... 50
4.1.8. Configuring the Driver and Applying the Patch ..... 50
4.2. Customizing the Reference Design for the CertusPro-NX Versa Board ..... 51
4.3. Customizing the Reference Design for the CertusPro-NX Sensor-to-Ethernet Bridge Board ..... 54
5. Compiling the Reference Design ..... 57
5.1. Configuring the Reference Design Parameters ..... 57
5.2. Assigning a Revision ID for Each Bitstream ..... 57
5.3. Configuring the Lattice Radiant Software to Compile the Reference Design ..... 58
5.4. Generating the Bitstream ..... 60
6. Implementing the Reference Design ..... 61
6.1. Hardware Requirements ..... 61
6.1.1. CertusPro-NX Versa Board and NVIDIA Jetson AGX Orin ..... 61
6.1.2. CertusPro-NX Versa Board and NVIDIA Jetson AGX Thor ..... 63
6.1.3. CertusPro-NX Sensor-to-Ethernet Bridge Board and NVIDIA Jetson AGX Orin ..... 64
6.2. Software Requirements ..... 65
6.2.1. Jetson AGX Orin ..... 65

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

6.2.2. Jetson AGX Thor ..... 65
6.3. NVIDIA Platform Host Setup ..... 65
6.3.1. Jetson AGX Orin Developer Kit ..... 65
6.3.2. Jetson AGX Thor Developer Kit ..... 74
6.4. Testing the System ..... 83
6.4.1. Test Setup on the CertusPro-NX Versa Board ..... 83
6.4.2. Test Setup on the CertusPro-NX Sensor-to-Ethernet Bridge Board ..... 94
7. Resource Utilization ..... 104
7.1. Resource Utilization Using the CertusPro-NX Versa Board ..... 104
7.1.1. Resource Utilization with Ethernet 1GbE ..... 104
7.1.2. Resource Utilization with Ethernet 2.5GbE ..... 105
7.1.3. Resource Utilization with Ethernet 10GbE ..... 106
7.2. Resource Utilization Using the CertusPro-NX Sensor-to-Ethernet Bridge Board ..... 107
7.2.1. Resource Utilization with Ethernet 10GbE ..... 107
8. Debug Methodology ..... 108
8.1. CertusPro-NX Versa Board ..... 108
8.1.1. Powering Up the Board ..... 109
8.1.2. Verifying the Ethernet Connection ..... 110
8.1.3. Running the Streaming Script (linux_imx258_player.py) ..... 112
8.2. CertusPro-NX Sensor-to-Ethernet Bridge Board ..... 114
8.2.1. Powering Up the Board ..... 115
8.2.2. Verifying the Ethernet Connection ..... 116
8.2.3. Running the Streaming Script (linux_imx274_player.py) ..... 118
9. Reference Design Configuration Tool ..... 120
9.1. Installation ..... 120
9.1.1. Windows Defender SmartScreen Warning ..... 120
9.2. Key Concepts ..... 121
9.2.1. Preview vs. Save Behavior ..... 121
9.2.2. MIPI Configuration Modes ..... 121
9.2.3. ISP and MJPEG Dependency ..... 122
9.3. User Interface ..... 122
9.3.1. Welcome Screen ..... 122
9.3.2. Main Workspace ..... 122
9.3.3. Menu Bar and Keyboard Shortcuts ..... 124
9.4. Project Loading ..... 126
9.5. Configuration Workflow ..... 126
9.6. Board-Specific Guidance ..... 127
9.7. Save Behavior and Backups ..... 127
9.8. Validation Messages ..... 127
9.8.1. Diagnostic Codes ..... 128
Appendix A: Known Issues ..... 129
References ..... 130
Technical Support Assistance ..... 131
Revision History ..... 132

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# Figures

Figure 1.1. Overview of Customizable HSB Sensor Interface RD with NVIDIA ecosystem ..... 12
Figure 2.1. Directory Structure ..... 17
Figure 3.1. Jetson AGX Orin Developer Kit ..... 19
Figure 3.2. Jetson AGX Thor Developer Kit ..... 20
Figure 3.3. CertusPro-NX Versa Board ..... 21
Figure 3.4. CertusPro-NX Sensor-to-Ethernet Bridge Board ..... 22
Figure 3.5. Holoscan Sensor Bridge IP in FPGA ..... 23
Figure 3.6. Overview block diagram of MIPI CSI-2 Sensor Interface ..... 24
Figure 3.7. LVDS DDR Receiver Interface on the CertusPro-NX Sensor-to-Ethernet Bridge Board ..... 27
Figure 3.8. Reset and Clock Domain Block Diagram for Ethernet 1GbE ..... 30
Figure 3.9. Reset and Clock Domain Block Diagram for Ethernet 2.5GbE ..... 30
Figure 3.10. Reset and Clock Domain Block Diagram for Ethernet 10GbE ..... 31
Figure 3.11. Reset and Clock Domain Block Diagram ..... 33
Figure 4.1. Steps Flow to Customize and Verify the Reference Design ..... 34
Figure 4.2. Steps to Generate ISP Lattice Semiconductor Reference Design IP ..... 36
Figure 4.3. Generated ISP IP Folder Structure ..... 37
Figure 4.4. LSCC ISP Builder Linux Orin GUI. ..... 38
Figure 4.5. ISP builder GUI with Live Image Streaming ..... 39
Figure 4.6. LSCC ISP BuilderGUI preview loading the configured JSON file. ..... 40
Figure 4.7. Generate Configuration window ..... 40
Figure 4.8. MIPI CSI-2 RX D-PHY IP Generation Module ..... 41
Figure 4.9. Ethernet 1GbE (MAC + SGMII SERDES) IP Generation ..... 42
Figure 4.10. Ethernet 2.5GbE (MAC + PHY) IP Generation ..... 43
Figure 4.11. Ethernet 10GbE MAC IP Generation ..... 43
Figure 4.12. Ethernet 10GbE PCS IP Generation ..... 44
Figure 4.13. Define Switch Setting in HOLOLINK_def.svh File ..... 45
Figure 4.14. PDC File Setting for CertusPro-NX Versa Board ..... 46
Figure 4.15. Screenshot of Lattice Radiant Software Hierarchy File List ..... 48
Figure 4.16. Preprocessor Macro to Enable Runtime MIPI CSI-2 D-PHY IP ..... 49
Figure 4.17. Configuring Runtime MIPI using Software Driver API ..... 49
Figure 4.18. CertusPro-NX Versa Board Block Diagram ..... 51
Figure 4.19. Required HDL Parameter and DEFINE Switch Configuration ..... 52
Figure 4.20. PDC Configuration File ..... 53
Figure 4.21. Lattice Radiant Software Hierarchy File List ..... 54
Figure 4.22. CertusPro-NX Sensor-to-Ethernet Bridge Board Block Diagram ..... 54
Figure 4.23. Required HDL Parameter and DEFINE Switch Configuration ..... 55
Figure 4.24. Lattice Radiant Software Hierarchy File List ..... 56
Figure 5.1. Revision ID assignment (hsb_cpnx_versa_top.sv) ..... 57
Figure 5.2. Revision ID assignment (hsb_cpnx_2chip_top.sv) ..... 57
Figure 5.3. Revision ID Date Code Representation ..... 58
Figure 5.4. Lattice Radiant Software ..... 58
Figure 5.5. Open Project File for CertusPro-NX Versa Board ..... 59
Figure 5.6. Place and Route Design Strategy ..... 59
Figure 5.7. Generate and Export Bitstream File ..... 60
Figure 6.1. Hardware Setup ..... 62
Figure 6.2. Camera Sensor Connection ..... 62
Figure 6.3. Setup with AGX Thor for 10GbE Ethernet ..... 63
Figure 6.4. Hardware Setup for using CertusPro-NX Sensor-to-Ethernet Bridge Board with AGX Orin ..... 64
Figure 6.5. Ubuntu Host PC and Jetson AGX Orin Developer Kit Setup ..... 65
Figure 6.6. Recovery and Reset Button ..... 66
Figure 6.7. SDK Manager Login ..... 66
Figure 6.8. Jetson AGX Orin [64 GB Developer Kit Version] ..... 67

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Figure 6.9. Development Environment Selection ..... 68
Figure 6.10. Terms and Conditions – License Agreement ..... 68
Figure 6.11. Administrative Right ..... 69
Figure 6.12. Download Operation and OS Image Creation ..... 69
Figure 6.13. SDK Manager ..... 70
Figure 6.14. Summary Finalization ..... 70
Figure 6.15. MAXN Power Mode ..... 72
Figure 6.16. Hardware Setup for Flash Jetson AGX Thor ..... 74
Figure 6.17. AGX Thor Recovery, Reset, and Power Button ..... 75
Figure 6.18. SDK Manager Login ..... 75
Figure 6.19. Device Selection ..... 76
Figure 6.20. STEP 01 – Development Environment ..... 76
Figure 6.21. STEP 02 – Details and License ..... 77
Figure 6.22. Administrative Right Request ..... 77
Figure 6.23. STEP 03 – Setup Process ..... 78
Figure 6.24. Recovery Mode Selection ..... 78
Figure 6.25. Pre-flash Operation Setup ..... 79
Figure 6.26. Post-flash Installation Dialog ..... 79
Figure 6.27. STEP 04 – Summary Finalization ..... 80
Figure 6.28. MAXN Power Mode ..... 81
Figure 6.29. Hardware Setup for Bit File Programming ..... 84
Figure 6.30. Radiant Programmer Window ..... 84
Figure 6.31. Select Cable Settings ..... 85
Figure 6.32. Load Bitstream File ..... 85
Figure 6.33. Program Device Toolbar Icon ..... 86
Figure 6.34. Message on Successful Programming ..... 86
Figure 6.35. Console Output — Remote Firmware Update ..... 87
Figure 6.36. Radiant Deployment Tool – Getting Started ..... 87
Figure 6.37. Select Input File ..... 88
Figure 6.38. Dual Boot Options ..... 88
Figure 6.39. Select Output File ..... 89
Figure 6.40. Generate Deployment ..... 89
Figure 6.41. Camera Video Streaming Test Setup ..... 91
Figure 6.42. Output Image under Different Video Streaming Format ..... 92
Figure 6.43. Live Body Pose Estimation Output ..... 93
Figure 6.44. Live TAO PeopleNet Output ..... 94
Figure 6.45. Bit File Programming Hardware Setup ..... 94
Figure 6.46. Lattice Programming Cable Pin Configuration ..... 95
Figure 6.47. Radiant Programmer – Getting Started ..... 95
Figure 6.48. External Flash Configuration ..... 95
Figure 6.49. External Flash Memory (SPI flash) Device Properties – CertusPro-NX Device ..... 96
Figure 6.50. External Flash Memory (SPI flash) Device Properties – CrossLink-NX Device ..... 97
Figure 6.51. Output Window ..... 97
Figure 6.52. Console Output – Remote Firmware Update ..... 98
Figure 6.53. Radiant Deployment Tool – Getting Started ..... 99
Figure 6.54. Select Input File ..... 99
Figure 6.55. Dual Boot Options ..... 100
Figure 6.56. Select Output File ..... 100
Figure 6.57. Generate Deployment ..... 101
Figure 6.58. Dual IMX274 Camera Video Streaming Test Setup ..... 102
Figure 6.59. Live Body Pose Estimation Output ..... 103
Figure 6.60. Live TAO PeopleNet Output ..... 103
Figure 8.1. Debug Methodology Flow Chart For RD ..... 108
Figure 8.2. Position of the LED on CertusPro-NX Versa Board ..... 109

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Figure 8.3. Ethernet Link Establishment Status ..... 110
Figure 8.4. Data Packet Received by Host ..... 111
Figure 8.5. Sample Output ..... 111
Figure 8.6. LAN Light Indication on Host ..... 112
Figure 8.7. Revision ID Version ..... 112
Figure 8.8. Camera I2C Configuration Fail Log ..... 113
Figure 8.9. Debug Methodology Flow Chart For RD ..... 114
Figure 8.10. Position of the Status LED on CertusPro-NX Sensor-to-Ethernet Bridge Board ..... 115
Figure 8.11. Ethernet Link Establishment Status ..... 116
Figure 8.12. Data Packet Received by Host ..... 117
Figure 8.13. Sample Output ..... 117
Figure 8.14. LAN Light Indication on Host ..... 118
Figure 8.15. Revision ID Version ..... 118
Figure 8.16. Camera I2C Configuration Fail Log ..... 119
Figure 9.1. Initial SmartScreen Prompt ..... 120
Figure 9.2. Click Run Anyway ..... 121
Figure 9.3. Welcome Screen ..... 122
Figure 9.4. Main Workspace – CertusPro-NX Versa Board ..... 123
Figure 9.5. File Menu ..... 124
Figure 9.6. Application Settings Dialog Box ..... 125
Figure 9.7. About Dialog Box ..... 125
Figure 9.8. Board Variant Dialog Box ..... 126
Figure 9.9. Close Project Dialog Box ..... 127

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# Tables

Table 1.1. Summary of CertusPro-NX Versa Board with NVIDIA Jetson AGX Orin ..... 13
Table 1.2. Summary of CertusPro-NX Versa Board with NVIDIA Jetson AGX Thor ..... 14
Table 1.3. Summary of CertusPro-NX Sensor-to-Ethernet Bridge Board with NVIDIA Jetson AGX Orin ..... 14
Table 2.1. Folder Structure Description ..... 17
Table 3.1. Supported Bayer processing and converter steps ..... 25
Table 3.2. Supported RGB processing and converter steps ..... 25
Table 3.3. Supported YUV444 processing and converter steps ..... 25
Table 3.4. Clock Domain Distribution for Reference design with Ethernet 1GbE ..... 31
Table 3.5. Clock Domain Distribution for Reference design with Ethernet 2.5GbE ..... 31
Table 3.6. Clock Domain Distribution for Reference design with Ethernet 10GbE ..... 32
Table 3.7. Reset Distribution applicable for both Ethernet 1GbE, 2.5GbE and 10GbE ..... 32
Table 3.8. Clock Domain Distribution for CertusPro-NX Sensor-to-Ethernet Bridge Board ..... 33
Table 3.9. Reset Distribution for CertusPro-NX Sensor-to-Ethernet Bridge Board ..... 33
Table 4.1. Selected Runtime Configuration for MIPI CSI-2 RX D-PHY IP ..... 34
Table 4.2. MIPI CSI-2 RX D-PHY IP configuration options ..... 35
Table 4.3. Generated ISP IP Folder Structure Description ..... 37
Table 4.4. PDC Template Files ..... 46
Table 4.5. PDC Pin Assignment for 2D Array for MIPI Data Lane ..... 47
Table 4.6. ISP or MJPEG Parameters Enablement in Hardware ..... 50
Table 4.7. Configuration for the reference design to fit into the CertusPro-NX Versa Board ..... 51
Table 4.8. Configuration for the reference design to fit into the CertusPro-NX Sensor-to-Ethernet Bridge Board ..... 55
Table 5.1. Board ID Representation in Hexadecimal ..... 57
Table 6.1. CertusPro-NX Platform and Jetson Module Compatibility ..... 61
Table 6.2. IMX258 Test Command Parameters and Default Values ..... 90
Table 7.1. Logic Consumption for Each Instance Available in the Reference Design ..... 104
Table 7.2. Comparison Between the Reference Design Utilization and the CertusPro-NX Resources Available ..... 104
Table 7.3. Logic Consumption for Each Instance Available in the Reference Design ..... 105
Table 7.4. Comparison Between the Reference Design Utilization and the CertusPro-NX Resources Available ..... 105
Table 7.5. Logic Consumption for Each Instance Available in the Reference Design ..... 106
Table 7.6. Comparison Between the Reference Design Utilization and the CertusPro-NX Resources Available ..... 106
Table 7.7. Logic Consumption for Each Instance Available in the Reference Design ..... 107
Table 7.8. Comparison Between the Reference Design Utilization and the CertusPro-NX Resources Available ..... 107
Table 8.1. LED Status upon Board Power-up ..... 109
Table 8.2. Expected Status LED Behavior when running linux_imx258_player.py.script ..... 113
Table 8.3. LED Status upon Board Power-up ..... 115
Table 9.1. Runtime MIPI Configuration Modes ..... 121
Table 9.2. Supported MIPI Configuration Modes by Lane Rate ..... 121
Table 9.3. Supported File and Folder Drop Location ..... 122
Table 9.4. Board connectivity ..... 123
Table 9.5. Cameras MIPI ..... 123
Table 9.6. Advanced features ..... 123
Table 9.7. Board identity ..... 124
Table 9.8. Editor tabs and status ..... 124
Table 9.9. Menu Bar Actions and Keyboard Shortcuts ..... 124
Table 9.10. Application Settings Option and Description ..... 125
Table 9.11. About Dialog Content ..... 125
Table 9.12. Project Loading Behavior for Multiple Board Variants ..... 126
Table 9.13. Message Indicators and Recommended Action ..... 126
Table 9.14. Board-Specific Features and Constraints ..... 127
Table 9.15. Configuration Data Saved to Each File ..... 127
Table 9.16. Message Status Indicator Definitions ..... 127
Table 9.17. Diagnostic Codes, Causes, and Recommended Fixes ..... 128

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# Abbreviations in This Document

A list of abbreviations used in this document.

| Abbreviations | Definition |
|---|---|
| APB | Advanced Peripheral Bus |
| AXI | Advanced eXtensible Interface |
| BOOTP | Bootstrap Protocol |
| CDC | Clock Domain Crossing |
| CRC | Cyclic Redundancy Check |
| CMOS | Complementary Metal-Oxide-Semiconductor |
| CUDA | Compute Unified Device Architecture |
| DCT | Discrete Cosine Transform |
| DDR | Dual Data Rate |
| D-PHY | Digital Physical Layer |
| EBR | Embedded Block RAM |
| ECB | Ethernet Control Bus |
| FP4 | 4-bit Floating-point |
| FPGA | Field-Programmable Gate Array |
| FW | Firmware |
| GUI | Graphical User Interface |
| HDL | Hardware Description Language |
| Hex | Hexadecimal |
| HIF | Host Interface |
| HSB | Holoscan Sensor Bridge |
| I2C | Inter-Integrated Circuit |
| ICMP | Internet Control Message Protocol |
| IP | Intellectual Property (Core) |
| ISP | Image Signal Processing |
| JSON | JavaScript Object Notation |
| LSCC | Lattice Semiconductor Company |
| MAC | Media Access Control |
| MIPI | Mobile Industry Processor Interface |
| MJPEG | Motion Joint Photographic Experts Group |
| MPCS | Multi-Protocol Communication Stack |
| NGC | NVIDIA GPU Cloud |
| PCB | Printed Circuit Board |
| PCS | Physical Coding Sublayer |
| PDC | Platform Design Constraint |
| PFU | Programmable Function Unit |
| PLL | Phase-Locked Loop |
| PTP | Precision Time Protocol |
| PMA | Physical Medium Attachment |
| PROM | Programmable Read-Only Memory |
| QSFP28 | Quad Small Form-factor Pluggable 28 |
| RTL | Register-Transfer level |
| SDK | Software Development Kit |
| SERDES | Serializer/Deserializer |
| SFP | Small Form-factor Pluggable |
| SPI | Serial Peripheral Interface |
| STA | Static Timing Analysis |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

| Abbreviations | Definition |
|---|---|
| TFLOPS | Tera Floating Point Operations Per Second |
| TOPS | Tera Operations Per Second |
| UDP | User Datagram Protocol |
| YAML | YAML Ain’t Markup Language |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# 1. Introduction

This user guide describes the reference design for customizable HSB sensor interfaces, on the Lattice CertusPro™-NX platforms. The supported platforms include the Lattice CertusPro-NX Versa Board and the Lattice CertusPro-NX Sensor-to-Ethernet Bridge Board. These boards function as the Holoscan Sensor Bridge between connected sensors and the NVIDIA host. The customizable interfaces in this reference design are divided into two interface categories.

- **Sensor Interface**
  This reference design targets a MIPI CSI-2 camera sensor. The sensor interface is designed to be flexible and scalable, allowing you to integrate multiple camera sensors based on your requirements. The system supports various configurations to meet diverse imaging needs.
- **Host Interface**
  The host interface focuses on Ethernet connectivity and supports multiple link speeds to accommodate varying data throughput requirements. This enables seamless communication between the sensor bridge and host systems.

**Note:**

- On the CertusPro-NX Versa Board, one Ethernet port and one camera connector are available for testing.
- On the CertusPro-NX Sensor-to-Ethernet Bridge Board, one Ethernet port and two camera connectors are available for testing.

The primary objective of the reference design is to establish a robust and reliable bridge between the HSB interfaces and the NVIDIA host. The reference design leverages NVIDIA's Holoscan Sensor Bridge solution and integrates smoothly within the broader NVIDIA ecosystem, enabling high-performance sensor data processing and transmission. An overview of the key components of the NVIDIA Holoscan Sensor Bridge solution and ecosystem is as follows:

1. Sensors
   - Targets the MIPI CSI-2 camera sensor.
2. Holoscan Sensor Bridge
   - CertusPro-NX Versa Board and CertusPro-NX Sensor-to-Ethernet Bridge Board function as the Holoscan Sensor Bridge solution.
   - Performs sensor data pre-processing to reduce computational load on the host system.
   - The reference design is implemented in the Lattice FPGA targeting LFCPNX-100 speed grade 9 device.
     The key components of the reference design include:
     - Sensor Interface
       - Supports MIPI CSI-2 camera sensors which are highly configurable to accommodate various camera sensors with different MIPI lanes and lane rates.
       - Supports an LVDS DDR receiver interface connection to an external FPGA device capable of handling multiple MIPI CSI-2 (D-PHY) channels for high-speed sensor input. For example, on the CertusPro-NX Sensor-to-Ethernet Bridge Board, the CrossLink-NX device serves as an external interface to the MIPI CSI-2 camera sensor and uses LVDS DDR receiver interface to transfer raw CSI-2 data to the CertusPro-NX device for processing.
     - Hololink IP
       - Acts as a bridge between the sensor interface and the host system.
       - Handles data encapsulation, synchronization, and buffering to ensure smooth transmission of high-throughput sensor data.
     - Host Interface (Ethernet link)
       - Provides a high-speed Ethernet link to the host, supporting 1GbE, 2.5GbE, or 10GbE speeds.
       - Enables low-latency, and high-bandwidth communication.
3. Host (NVIDIA Jetson AGX Orin/Thor Developer Kit)
   - Runs the Holoscan SDK to process incoming data.
   - Performs real-time AI inference, visualization, and decision-making.
   - Strems processed data to the cloud or other edge devices.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

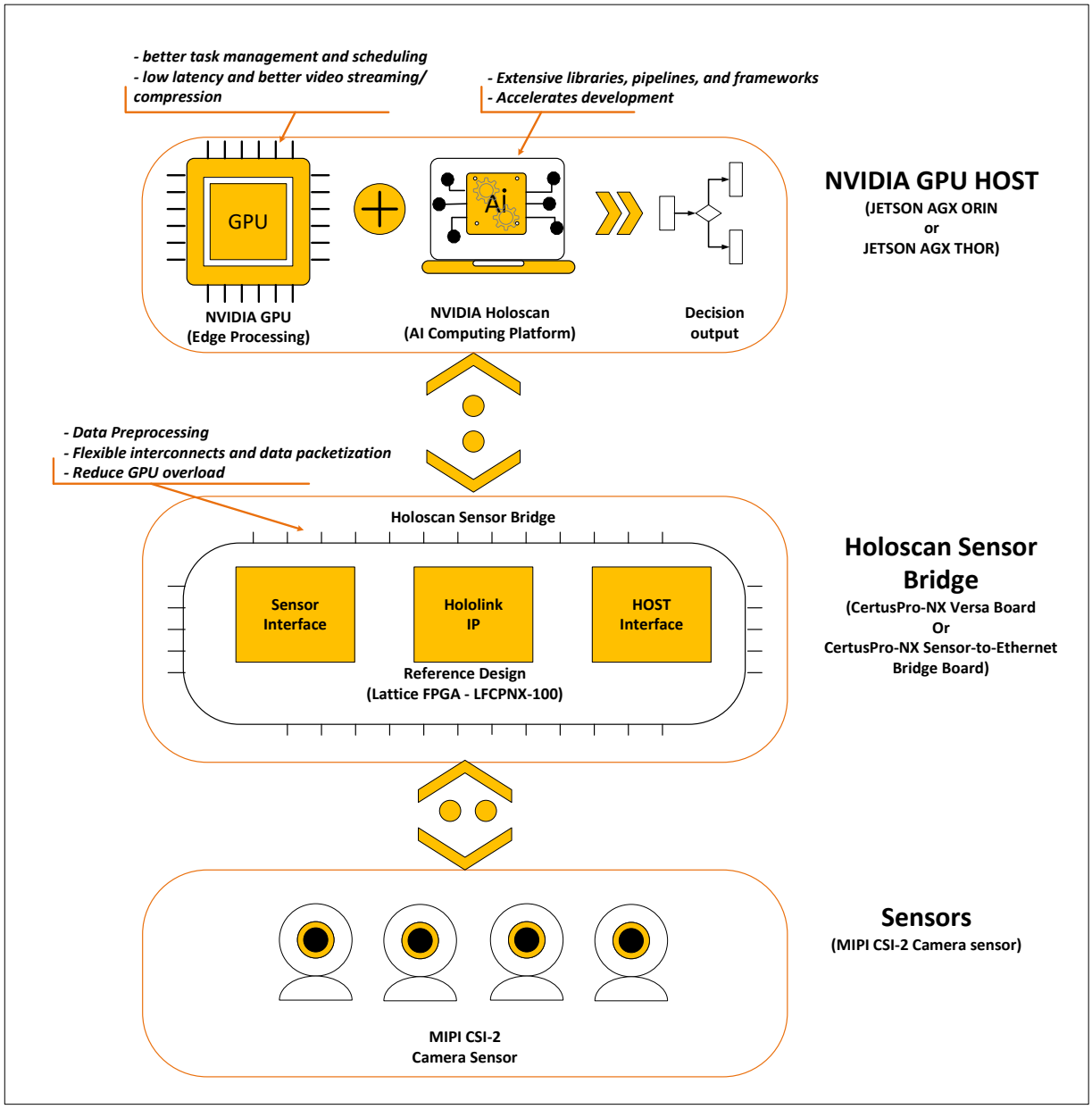



**Figure 1.1. Overview of Customizable HSB Sensor Interface RD with NVIDIA ecosystem**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

## 1.1. Reference Design: Quick Facts

### 1.1.1. CertusPro-NX Versa Board with Jetson AGX Orin/Thor

**Table 1.1. Summary of CertusPro-NX Versa Board with NVIDIA Jetson AGX Orin**

| | | |
|---|---|---|
| **General** | Target Devices | CertusPro-NX FPGA (LFCPNX-100-9LFG672I) |
| | Source code format | Verilog and SystemVerilog |
| **Simulation** | Functional simulation | N/A |
| | Timing simulation | N/A |
| | Testbench | N/A |
| | Testbench format | N/A |
| **Software Requirements** | Software tool and version | Lattice Radiant™ software, version 2026.1.0.37.0 |
| | IP version (if applicable) | • CSI-2/DSI D-PHY Rx IP v2.2.0<br>• OSC IP v1.4.0<br>• PLL IP v1.9.1<br>• 10GbE Ethernet PCS IP Core v1.3.0<br>• 10GbE Ethernet MAC IP Core v1.1.0<br>• Tri-Speed Ethernet v2.1.0 (MAC + SGMII (SERDES))<br>• 2.5GbE Ethernet (MAC + PHY IP) Core v1.2.0<br>• ISP Lattice Semiconductor Reference Design IP v1.1.1<br>• Launch LSCC ISP Builder Linux Orin v1.1.0<br>• MJPEG Lattice Semiconductor Reference Design IP<br>• Hololink IP v2606 |
| | Host Software (AGX Orin) | • JetPack 7.2<br>• NVIDIA Holoscan SDK 4.4.0<br>• NVIDIA Jetson Linux 39.2<br>• HSB SDK 2.7.0 |
| **Hardware Requirements** | Board | • CertusPro-NX Versa Board<br>• IMX258 camera module<br>• Ethernet 10GBase-T (Cat 6 RJ-45) SFP+ module (10GbE Ethernet)<br>• Ethernet 2.5GBase-T (Cat 6 RJ-45) SFP-2.5G-T-1 module (2.5GbE Ethernet)<br>• 10/100/1000BASE-T SFP module (1GbE Ethernet)<br>• Display Monitor |
| | Host | • NVIDIA Jetson AGX Orin Developer Kit<br>• M.2 2280 NVMe SSD 512GB |
| | Cable | • DisplayPort cable<br>• Ethernet Cat 6 cable |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

**Table 1.2. Summary of CertusPro-NX Versa Board with NVIDIA Jetson AGX Thor**

| | | |
|---|---|---|
| **General** | Target Devices | CertusPro-NX FPGA (LFCPNX-100-9LFG672I) |
| | Source code format | Verilog and SystemVerilog |
| **Simulation** | Functional simulation | N/A |
| | Timing simulation | N/A |
| | Testbench | N/A |
| | Testbench format | N/A |
| **Software Requirements** | Software tool and version | Lattice Radiant software, version 2026.1.0.37.0 |
| | IP version (if applicable) | • CSI-2/DSI D-PHY Rx IP v2.2.0<br>• OSC IP v1.4.0<br>• PLL IP v1.9.1<br>• 10GbE Ethernet PCS IP Core v1.3.0<br>• 10GbE Ethernet MAC IP Core v1.1.0<br>• ISP Lattice Semiconductor Reference Design IP v1.1.1<br>• Launch LSCC ISP Builder Linux Thor v1.1.0<br>• MJPEG Lattice Semiconductor Reference Design IP<br>• Hololink IP v2606 |
| | Host Software (AGX Thor) | • JetPack 7.2<br>• NVIDIA Holoscan SDK 4.4.0<br>• NVIDIA Jetson Linux 39.2<br>• HSB SDK 2.7.0 |
| **Hardware Requirements** | Board | • CertusPro-NX Versa Board<br>• IMX258 camera module<br>• Display Monitor |
| | Host | • NVIDIA Jetson AGX Thor Developer Kit |
| | Cable | • DisplayPort cable<br>• 100G QSFP28 to 4×25G SFP28 Breakout Active Optical Cable (AOC) |

### 1.1.2. CertusPro-NX Sensor-to-Ethernet Bridge Board with Jetson AGX Orin

**Table 1.3. Summary of CertusPro-NX Sensor-to-Ethernet Bridge Board with NVIDIA Jetson AGX Orin**

| | | |
|---|---|---|
| **General** | Target Devices | CertusPro-NX FPGA (LFCPNX-100-9LFG672C)<br>CrossLink-NX FPGA (LIFCL-17-9BG256C) |
| | Source code format | Verilog and SystemVerilog |
| **Simulation** | Functional simulation | N/A |
| | Timing simulation | N/A |
| | Testbench | N/A |
| | Testbench format | N/A |
| **Software Requirements** | Software tool and version | Lattice Radiant software, version 2026.1.0.37.0 |
| | IP version (if applicable) | • LVDS DDR Receiver IP v1.7.0<br>• OSC IP v1.4.0<br>• PLL IP v1.9.1<br>• 10GbE Ethernet PCS IP Core v1.3.0<br>• 10GbE Ethernet MAC IP Core v1.1.0<br>• Hololink IP v2606<br>• CrossLink-NX FPGA bitstream |
| | Host Software (AGX Orin) | • JetPack 7.2<br>• NVIDIA Holoscan SDK 4.4.0<br>• NVIDIA Jetson Linux 39.2<br>• HSB SDK 2.7.0 |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

| | | |
|---|---|---|
| **Hardware Requirements** | Board | • CertusPro-NX Sensor-to-Ethernet Bridge Board<br>• IMX274 camera module<br>• Ethernet 10GBase-T (Cat 6 RJ-45) SFP + module (10GbE Ethernet)<br>• Display Monitor |
| | Host | • NVIDIA Jetson AGX Orin Developer Kit<br>• M.2 2280 NVMe SSD 512GB |
| | Cable | • DisplayPort cable<br>• Ethernet Cat 6 cable |

## 1.2. Quick-Start Guides

IMX258 Camera Streaming Setup – see Implementing the Reference Design section

Instructions for setting up and streaming video from the IMX258 camera module to the NVIDIA Jetson AGX Orin/Thor developer kit over a 1GbE/2.5GbE/10GbE Ethernet connection.

IMX274 Camera Streaming Setup – see Implementing the Reference Design section

Instructions for setting up and streaming video from the IMX274 camera module to the NVIDIA Jetson AGX Orin developer kit over a 10GbE Ethernet connection.

Jetson AGX Orin Developer Kit Setup

Steps for setting up the NVIDIA Jetson AGX Orin developer kit.

Jetson AGX Thor Developer Kit Setup

Steps for setting up the NVIDIA Jetson AGX Thor developer kit.

## 1.3. Features

Key features of the reference design include:

- Sensor Interface
  - Supports one or more camera sensors.
  - Designed to be customizable, allowing flexible configuration of the MIPI CSI-2 RX D-PHY interface.
  - Enables integration of various camera types.
  - The interface type can be configured to support either the MIPI CSI-2 RX interface or the LVDS DDR receiver interface.
  - Performs data pre-processing, supporting raw MIPI CSI-2 data capture as well as optional image processing such as ISP-based image enhancement and MJPEG image decoding.
- Hololink IP
  - Acts as a bridge between the sensor interface and the host system.
  - Implements Ethernet packetization and supports streaming DMA, control interfaces, and transport abstraction.
  - Enables low-latency, high-throughput data transfer from sensors to compute platforms.
  - Supports Precision Time Protocol (PTP) per IEEE 1588-2019 specification. PTP synchronizes the Hololink IP internal time with the host time, enabling the host to act as the Time Transmitter and send PTP packets.
- Host Interface (Ethernet link)
  - Supports Ethernet channels with a link speed of 1GbE, 2.5GbE, or 10GbE.
  - Uses UDP over Ethernet to stream sensor data directly to GPU memory.
  - Enables real-time AI inference and visualization with minimal latency.
  - In testing, the CertusPro-NX Versa Board was used with a single Ethernet port configured for either 1GbE, 2.5GbE or 10GbE.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Features are enabled and implemented according to the target platform configured as a Holoscan Sensor Bridge. For detailed instructions, see the Customizing the Reference Design for CertusPro-NX Versa Board section and the Customizing the Reference Design for a CertusPro-NX Sensor to Ethernet Bridge Board section. You can tune the reference design to meet your own platform requirement.

## 1.4. Naming Conventions

### 1.4.1. Nomenclature

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

### 1.4.2. Signal Names

- *_n* are active low (asserted when value is logic 0)
- *_i* are input signals
- *_o* are output signals

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# 2. Directory Structure and File Overview

The following figure illustrates the directory structure of the reference design.

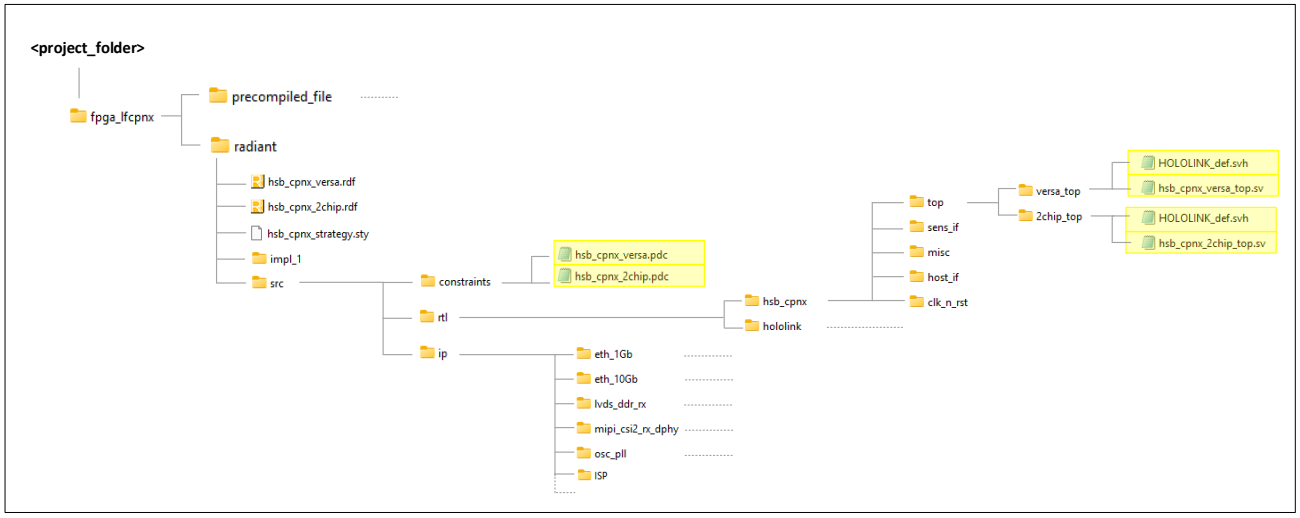



**Figure 2.1. Directory Structure**

**Table 2.1. Folder Structure Description**

| Reference Design Specific | Folder/File | Description |
|---|---|---|
| Common Folders/files | radiant/src/rtl/hsb_cpnx/ | This folder contains HDL code for the sensor interface, host interface, top-level design and other components. |
| | radiant/src/rtl/hololink/ | This folder contains the encrypted Hololink IP. The current version is 2606. |
| | radiant/ip/ | This folder contains IP cores used in the reference design, including the oscillator, PLLs, MIPI CSI-2 RX D-PHY IP, Ethernet 10GbE IP, Ethernet 2.5GbE IP, Tri-Speed Ethernet (MAC + SGMII/SERDES), and the CrossLink-NX (LIFCL-17) FPGA as an external device, along with other supporting components. |
| | radiant/ip/ISP/ | This folder contains ISP and MJPEG Lattice Semiconductor Reference design IP. |
| | radiant/hsb_cpnx_strategy.sty | This strategy file is shared by both reference designs. It includes custom settings optimized for improved place-and-route results. |
| | radiant/imp_1 | This folder contains reports and logs generated during FPGA bitstream-generation compilation. |
| | precompiled_file/ | The folder contains the FPGA-compiled bitstream for each reference design. Each reference design generates its own bitstream. For the CertusPro-NX Sensor-to-Ethernet Bridge Board, two FPGA bitstreams are produced: one for the CertusPro-NX device and one for the CrossLink-NX device. |
| CertusPro-NX Versa Board | radiant/hsb_cpnx_versa.rdf | This file is the Lattice Radiant software project file for the CertusPro-NX Versa reference design. |
| | radiant/src/constraints/hsb_cpnx_versa.pdc | This folder contains the PDC files for the CertusPro-NX Versa Board. |
| | radiant/src/rtl/hsb_cpnx/top/versa_top/hsb_cpnx_versa_top.sv | This file is the top-level module file for the CertusPro-NX Versa Board reference design. |
| | radiant/src/rtl/hsb_cpnx/top/versa_top/HOLOLINK_def.svh | This file contains the define switches used in the CertusPro-NX Versa Board reference design. |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

| Reference Design Specific | Folder/File | Description |
|---|---|---|
| CertusPro-NX Sensor to Ethernet Bridge Board | radiant/hsb_cpnx_2chip.rdf | This file is the Lattice Radiant software project file for the CertusPro-NX Sensor-to-Ethernet Bridge Board reference design. |
| | radiant/src/constraints/hsb_cpnx_versa.pdc | This folder contains the PDC files for the CertusPro-NX Sensor-to-Ethernet Bridge Board. |
| | radiant/src/rtl/hsb_cpnx/top/2chip_top/ hsb_cpnx_2chip_top.sv | This file is the top-level module file for the CertusPro-NX Sensor-to-Ethernet Bridge Board reference design. |
| | radiant/src/rtl/hsb_cpnx/top/2chip_top/ HOLOLINK_def.svh | This file contains the define switches used in the CertusPro-NX Sensor-to-Ethernet Bridge Board reference design. |

The project folder structure and files are shared between the two reference designs targeted for the CertusPro-NX Versa Board and the CertusPro-NX Sensor-to-Bridge Board. Some folders and files are common to both reference designs, while others are specific to each design.

The files highlighted in Figure 2.1 (radiant/src/rtl/hsb_cpnx/top/versa_top/HOLOLINK_def.svh, radiant/src/rtl/hsb_cpnx/top/2chip_top/HOLOLINK_def.svh, radiant/src/rtl/hsb_cpnx/top/versa_top/hsb_cpnx_versa_top.sv, radiant/src/rtl/hsb_cpnx/top/2chip_top/hsb_cpnx_2chip_top.sv, radiant/src/constraints/pdc/hsb_cpnx_versa.pdc and radiant/src/constraints/hsb_cpnx_2chip.pdc) are important. You must modify these files when developing your custom reference design.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# 3. Functional Description

## 3.1. NVIDIA Host Developer Kits

This section provides an overview of the NVIDIA host developer kit, highlighting key features of the Jetson AGX Orin and Jetson AGX Thor developer kits. For additional information, refer to the NVIDIA Developer web page and search for Jetson Developer Kits.

### 3.1.1. Jetson AGX Orin Developer Kit



**Figure 3.1. Jetson AGX Orin Developer Kit**

The NVIDIA Jetson AGX Orin is a compact, high-performance edge AI platform for workloads that require substantial on-device AI processing. It delivers 200–275 INT8 TOPS of compute through an Ampere-architecture GPU with 1,792–2,048 CUDA cores and 56–64 Tensor Cores. The system integrates either an 8-core or 12-core Arm Cortex-A78AE CPU running up to 2.2 GHz and 32 GB or 64 GB of LPDDR5 memory with 204.8 GB/s bandwidth. These resources enable efficient execution of complex AI pipelines and high-throughput data streams.

Jetson AGX Orin supports NVIDIA software stack, including Isaac for robotics, DeepStream for vision AI, and Riva for conversational AI, allowing you to build applications across multiple AI and robotics domains.

Jetson AGX Orin provides a single Multi-Gigabit Ethernet (MGBE) interface. Depending on the carrier board and PHY implementation, the interface typically supports 1GbE, 2.5GbE, 5GbE, and 10GbE Ethernet. This bandwidth is sufficient for receiving UDP-based sensor data from a Holoscan Sensor Bridge, although it operates at lower throughput than NVIDIA Thor 25GbE Ethernet interfaces. In the current reference design, only the 1GbE and 10GbE Ethernet link speeds are enabled.

Jetson AGX Orin supports Precision Time Protocol (PTP) through its integrated MGBE interface, enabling accurate time synchronization with attached sensors. The platform fully supports UDP transport, providing low-latency, real-time data streaming. Although its Ethernet bandwidth is lower than that of NVIDIA Thor, Orin can accommodate multiple sensor inputs as long as the combined throughput remains within the capacity of the MGBE interface.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### 3.1.2. Jetson AGX Thor Developer Kit



**Figure 3.2. Jetson AGX Thor Developer Kit**

NVIDIA Jetson AGX Thor is a next-generation robotics and physical-AI computing platform built for highly demanding real-time sensor workloads. It provides up to 2,070 FP4 TFLOPS of AI compute powered by the NVIDIA Blackwell GPU architecture, offering more than 7.5X greater AI performance and substantially improved energy efficiency compared to Jetson AGX Orin.

The system integrates 128 GB of LPDDR5X memory with 273 GB/s of bandwidth, enabling efficient handling of large-scale multimodal data streams, including high-resolution camera, LiDAR, and radar inputs. To meet real-time robotics requirements, Jetson AGX Thor includes a 14-core Arm Neoverse-V3AE CPU that provides deterministic compute performance for sensor fusion, perception, and rapid decision-making. When combined with the NVIDIA Holoscan framework, the platform can execute complex sensor pipelines with sub-10 ms latency, supporting advanced autonomous-system and robotics workloads.

Jetson AGX Thor provides high-bandwidth Ethernet connectivity for integration with the Holoscan Sensor Bridge. The system includes a QSFP28 interface supporting four 25 Gbps Ethernet links, delivering a combined throughput of 100 Gbps. This capacity allows the host to ingest multiple high-rate UDP sensor streams from imaging and ranging devices without performance bottlenecks. Thor also incorporates a 5GbE RJ45 Ethernet port that can be used for system control, secondary data paths, or redundancy, depending on system requirements.

All high-speed Ethernet interfaces on Jetson AGX Thor support IEEE 1588 Precision Time Protocol, enabling accurate alignment of sensor timestamps when used with the Holoscan Sensor Bridge. This capability is essential for synchronized streaming across multiple sensors, including cameras, LiDAR, radar, and microphones. Thor also provides full UDP transport support, offering low-latency, non-blocking data delivery suitable for real-time systems. With its 25GbE Ethernet links, the platform can maintain continuous high-throughput UDP streams, ensuring reliable frame-level sensor data reception even under heavy system load.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

## 3.2. Lattice-Powered Holoscan Sensor Bridge Platforms

This section provides an overview of the Lattice-powered Holoscan Sensor Bridge platforms. The supported CertusPro-NX platforms include the CertusPro-NX Versa Board and the CertusPro-NX Sensor-to-Ethernet Bridge Board.

### 3.2.1. CertusPro-NX Versa Board

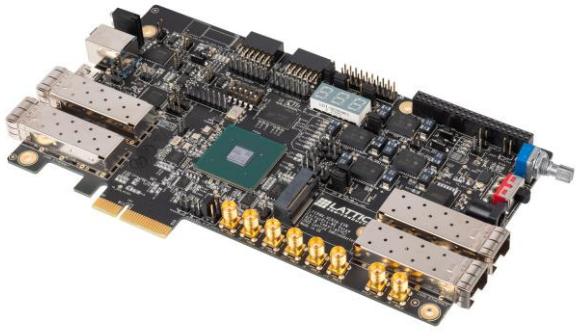


**Figure 3.3. CertusPro-NX Versa Board**

The CertusPro-NX Versa Board is a flexible FPGA evaluation and development platform based on the CertusPro-NX (LFCPNX-100-9LFG672I device). It enables designers to prototype and validate a wide range of high-performance connectivity and processing use cases. The board extends the FPGA's functionality through multiple industry-standard interfaces, including dual SFP ports supporting 10-Gigabit Ethernet and 1-Gigabit SGMII, two SERDES channels, PCIe Gen3 x4, and a MIPI CSI-2 camera connector, providing robust options for data acquisition and high-speed communication.

The CertusPro-NX Versa Board provides essential development features, including LPDDR4 memory, GPIO indicators, push-buttons, DIP switches, onboard boot flash for configuration storage, and an FT2232H USB-to-JTAG interface for FPGA programming and debugging. These capabilities make the platform well-suited for designing and validating systems that incorporate video bridging, sensor interfaces, networking protocols, and PCIe-based acceleration.

Engineers can use extensive expandability through SFP cages, PMOD connectors, Raspberry Pi HAT support, and SERDES access ports, making the Versa Board a versatile hardware platform for edge-AI and sensor-processing applications. The board supports both endpoint and root-port modes for PCIe Gen3, enabling flexible FPGA integration in host-centric systems or deployment as an intelligent controller within distributed architectures. For additional information, refer to the CertusPro-NX Versa Board User Guide (FPGA-EB-02053).

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### 3.2.2. CertusPro-NX Sensor-to-Ethernet Bridge Board



**Figure 3.4. CertusPro-NX Sensor-to-Ethernet Bridge Board**

The CertusPro-NX Sensor-to-Ethernet Bridge Board is an FPGA-based connectivity platform that links a variety of sensor interfaces to high-speed Ethernet networks. It integrates both CertusPro-NX and CrossLink™-NX FPGA devices, enabling flexible sensor data ingestion through AXI-Stream and APB control buses and converting this data into Ethernet packets for downstream processing.

This board is specifically optimized for low-latency, high-throughput sensor streaming, and it integrates with the NVIDIA Holoscan Sensor Bridge, providing a complete end-to-end solution for real-time sensor ingestion. The reference design available with the board ensures that data captured at the FPGA's sensor interface reliably arrives in system DRAM, and it supports advanced features such as streaming DMA, transport abstraction, Linux sockets, and ConnectX SmartNIC acceleration using GPUDirect RDMA.

To support high-bandwidth networking, the board provides up to two 10-Gigabit Ethernet channels, enabling the transport of multiple high-rate sensor streams over UDP or other low-latency protocols to hosts such as Jetson AGX Orin or NVIDIA IGX Orin. The platform open-source software stack, configurable FPGA IP blocks, and programmable system-control framework allow rapid customization of sensor inputs, making the bridge board well-suited for AI-enabled edge systems that require robust sensor fusion. For additional details, refer to the CertusPro-NX Sensor-to-Ethernet Board User Guide (FPGA-EB-02069).

## 3.3. Hololink IP and Customizable Interfaces

### 3.3.1. Overview

The NVIDIA Holoscan Sensor Bridge FPGA IP works with the Holoscan software to provide a sensor-agnostic platform that transfers data from various sensors to an Ethernet-connected host.

The Holoscan Sensor Bridge FPGA IP comprises three main components: the Sensor Interface IP, the Hololink IP, and the Host Interface IP. The block diagram of the Holoscan Sensor Bridge FPGA IP is illustrated in Figure 3.5. The Sensor Interface and Host (Ethernet) Interface blocks are implemented using FPGA vendor-specific logic. In this reference design, Lattice Semiconductor is responsible for the integrating and maintaining the IP for both the Sensor Interface and Host Interface components.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

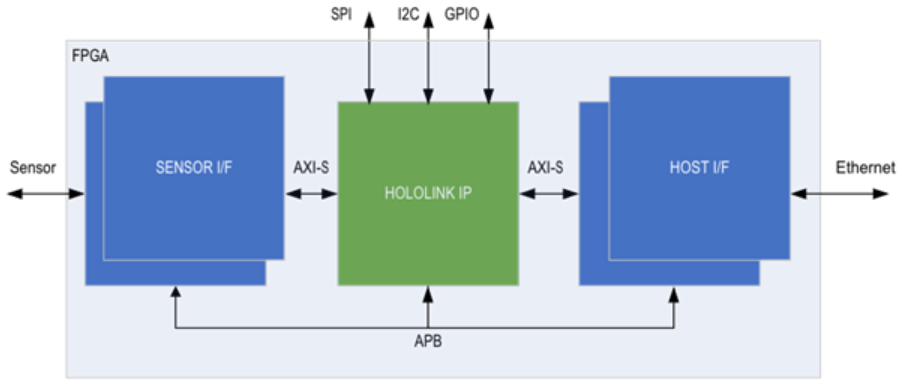



**Figure 3.5. Holoscan Sensor Bridge IP in FPGA**

The Holoscan Sensor Bridge FPGA IP streamlines FPGA development, and provides scalability and configurability to support a wide range of sensor-to-host applications.

Key functions of the Holoscan Sensor Bridge IP include:

- Encapsulation of sensor data: Converts AXI-Stream sensor data into Ethernet UDP AXI-Stream format for host-side processing.
- Network protocol support: Implements BOOTP, ICMP, and NVIDIA-defined Ethernet Control Bus (ECB) protocols.
- Event and control packet transmission: Sends enumeration and control event packets based on predefined conditions.
- Peripheral interface control: Manages SPI, I2C, and GPIO interfaces to configure sensors and other onboard components.

A detailed explanation on the NVIDIA Hololink IP is provided in the NVIDIA documentation webpage. Search for *Holoscan Sensor Bridge,* and then select the respective HSB SDK release. This section provides only an overview of the IP.

## 3.3.2. Sensor Interface

### 3.3.2.1. MIPI CSI-2 RX D-PHY Sensor Interface

The customizable MIPI CSI-2 camera sensor interface has two capabilities:

- **Scalability**
  Refers to the ability of the MIPI CSI-2 sensor interface to increase or decrease the number of sensor interfaces based on user platform requirements.
- **Configurability**
  Refers to the ability to configure each camera sensor interface individually, such as setting different MIPI CSI-2 sensor lane and lane rates.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### MIPI CSI-2 Payload Data to AXI-Stream Payload

Figure 3.6 shows the block diagram of the MIPI CSI-2 sensor interface, which outputs either MIPI CSI-2 RAW data, MJPEG bitstream or ISP YUV422 data format.

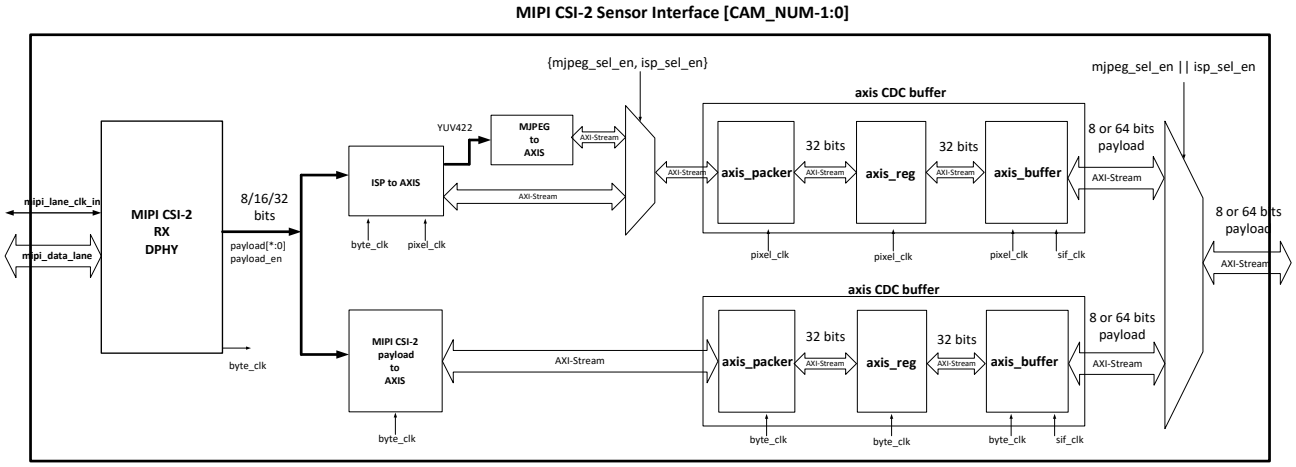



**Figure 3.6. Overview block diagram of MIPI CSI-2 Sensor Interface**

The MIPI CSI-2 sensor interface includes five primary components: MIPI CSI-2 RX D-PHY instance, MIPI CSI-2 raw10 to AXIS, ISP to AXIS, MJPEG to AXIS and AXIS CDC buffer. The number of MIPI CSI-2 sensor interfaces varies based on the assigned number of cameras.

1. **MIPI CSI-2 RX D-PHY Instance**

   The MIPI CSI-2 RX D-PHY instances serve as the primary interface for MIPI CSI-2 image sensors. The reference design targets the LFCPNX-100 device package, therefore only the soft MIPI CSI-2/DSI RX D-PHY IP is supported. This configurable IP block receives MIPI CSI-2 data streams and converts them into 8-bit, 16-bit, or 32-bit parallel data, depending on the lane configuration selected during IP generation.

   The IP allows you to set the MIPI CSI-2 lane rate based on the camera sensor's requirements. It also generates an output byte clock derived from the data type and forwards it to the AXIS blocks. If the sensor provides a continuous D-PHY clock, you should disable the RX_FIFO and drive clk_byte_fr_i directly using clk_byte_hs_o.

   Compared to the hardened version, the soft MIPI CSI-2 RX D-PHY IP has performance limitations that vary by FPGA family and package. For detailed specifications and configuration guidance, see the CSI-2/DSI D-PHY Rx IP Core User Guide (FPGA-IPUG-02081).

   MIPI CSI-2 RX D-PHY instance can be configured to support either PRESET or RUNTIME changeable MIPI configuration. The MIPI configuration refers to the MIPI lane rate and MIPI lane number. The following section describes the steps to configure the MIPI configuration.

2. **MIPI CSI-2 RAW10 to AXIS**

   The MIPI CSI-2 RAW10 payload data is converted into an AXI4-Stream (AXIS) interface. The module extracts image data packets from the CSI-2 stream and reformats them into AXIS compliant payloads for downstream processing. This conversion supports parallel RAW10 output in 8-bit, 16-bit, or 32-bit formats, as supported by the MIPI CSI-2 RX D-PHY IP.

3. **ISP to AXIS Instance**

   The ISP to AXIS instance consists of two main functional components:

   a. Byte-to-Pixel Conversion module

   b. ISP IP module

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

The Byte-to-Pixel Conversion module performs clock-domain crossing between the MIPI CSI-2 byte clock domain and the pixel clock domain. Incoming MIPI CSI-2 RAW10 payload data is unpacked and converted into a pixel-formatted data stream suitable for ISP processing. This pixel-formatted data is then processed by the image processing functions enabled within the ISP IP. All ISP processing runs in the pixel clock domain, with the pixel clock set to 150 MHz. The current Byte-to-Pixel Conversion module supports only MJPEG operation; the supported resolution is 1920 × 1080.

The ISP IP allows you to select an ISP processing configuration that best suits your application requirements. Three processing domains (Bayer, RGB, and YUV) are supported within the ISP pipeline. These domains form a sequential processing pipeline, not parallel paths. Color-space conversion stages (Debayer, RGB-to-YUV, and YUV444-to-YUV422) determine the transition points between processing domains. Regardless of the selected ISP functions, the final output of the ISP IP is always YUV422 for the Holoscan application.

The current version of the ISP IP supports a limited set of ISP algorithms and processing steps in three color spaces including Bayer, RGB, and YUV. The supported ISP steps are detailed in the following tables.

**Table 3.1. Supported Bayer processing and converter steps**

| Bayer Processing | Descriptions | Available Configurations |
|---|---|---|
| Black Level Correction (BLC) | Correct per-plane black offset on Bayer data. | Per-channel black level offsets. |
| Digital Gain (DG) | Apply gain on Bayer domain. | Per-channel gains. |
| Image Statistics | Collect image statistics (passthrough step). | Output Min/Max/Avg, bright/dark count thresholds. |
| **Bayer converter** | **Descriptions** | **Available Configurations** |
| Debayer (Linear 3x3) | Convert Bayer to RGB. | None |

**Table 3.2. Supported RGB processing and converter steps**

| RGB Processing | Descriptions | Available Configurations |
|---|---|---|
| Black Level Correction (BLC) | Apply RGB-domain black-level correction. | Per-channel offsets. |
| Digital Gain (DG) | Apply RGB gain. | R/G/B gain, per-channel mode. |
| AWB (Gray World) | White-balance correction using the gray world algorithm | None |
| Color Correction Matrix (CCM) | Color correction using 3x3 transform. | Matrix coefficients. |
| 1-Dimensional Look-Up-Table (LUT1D) | Tone and gamma curve remapping. | Curve type and curve parameters via LUT dialog. |
| **RGB converter** | **Descriptions** | **Available Configurations** |
| RGB2YUV | Convert RGB to YUV444. | None |

**Table 3.3. Supported YUV444 processing and converter steps**

| YUV444 Processing | Descriptions | Available Configurations |
|---|---|---|
| Chroma Saturation | Adjust U/V saturation strength. | Saturation gain. |
| Sharpen | Luma sharpening with unsharp mask. | Gaussian window size, sharpen amount, threshold. |
| **YUV444 converter** | **Descriptions** | **Available Configurations** |
| YUV444toYUV422 | Convert YUV444 to YUV422. | None |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

The output payload format of the ISP IP is fixed to YUV422. Based on the MUX control signal, the YUV422 payload is either routed to the MJPEG IP or forwarded to the Hololink IP after being converted into an AXI-Stream (AXIS)-compliant payload.

**4. MJPEG to AXIS**

When the MJPEG mux control signal is selected, the YUV422 payload output from the ISP instance is routed to the MJPEG IP for compression. The MJPEG IP encodes the incoming video stream as a sequence of independent JPEG frames. Each frame is processed using the standard JPEG compression pipeline, including DCT, quantization, and Huffman encoding. However, unlike conventional JPEG mode, the Huffman table is not embedded in each frame header; instead, the encoder uses the standard JPEG Huffman table internally.

By excluding the Huffman table from every frame header, MJPEG mode reduces header size and minimizes overhead, making it well suited for continuous video streaming applications such as camera interfaces and real-time data transmission. Despite this optimization, the encoding process and output format remain JPEG-compliant, with each frame properly terminated by an End-of-Image (EOI) marker. This approach is particularly efficient in systems where the decoder already has knowledge of the standard Huffman tables and does not require them to be transmitted repeatedly.

The resulting MJPEG bitstream is output after being formatted into an AXI-Stream (AXIS) compliant interface. The MJPEG IP operates with a pixel clock of 150 MHz and supports 1920 × 1080 (Full HD) resolution. When the four-lane MIPI sensor is used, the MIPI lane rate must not exceed 375 Mbps. This MJPEG IP is a Lattice Semiconductor proprietary reference design provided in encrypted form and is integrated as part of the Holoscan reference design.

**5. AXIS CDC Buffer**

- AXIS Packer Module:
  A key feature of this module is its handling of the TKEEP signal, which indicates the validity of individual bytes within each data word. The AXIS Packer filters out invalid TKEEP bits, ensuring that only valid image data is included in the output stream. This helps maintain data integrity and prevents downstream modules from processing corrupted or incomplete payloads.
- AXIS Reg Module
  The AXIS Reg module acts as a simple register stage within the AXIS data path. Its primary function is to shift and hold one cycle of AXIS data, which helps manage timing, pipeline depth, and synchronization between modules.
- AXIS Buffer Module
  The AXIS Buffer module receives AXI4-Stream data from the AXIS Reg module and transfers it through a dual-clock FIFO (DC_FIFO) to support clock domain crossing (CDC). This enables reliable data movement between components that operate in separate clock domains.
  Key functions include:
  - Data buffering: Temporarily stores AXIS data and control signals to manage timing differences.
  - Clock-domain crossing: Uses a DC_FIFO to safely transfer data between asynchronous clock domains.
  - Flow control: Maintains AXIS protocol integrity using TVALID, TREADY, and TLAST signals across domains.

  The AXIS Buffer module is essential in systems where the CSI-2 receiver and downstream AXIS processing blocks operate on separate clocks, ensuring smooth and reliable data streaming.

#### 3.3.2.2. LVDS DDR Receiver Interface

The CertusPro-NX Sensor-to-Ethernet Bridge design uses the Lattice LVDS DDR Receiver IP to acquire high-speed LVDS-serialized image data from the camera sensor. The IP deserializes the DDR LVDS data stream and outputs pixel-domain parallel video data, which is then passed through the internal processing pipeline to the Ethernet transmission path. LVDS offers reduced signal swing and lower power consumption compared to traditional signaling methods. These characteristics make LVDS a common interface in modern CMOS image sensors. The interface includes one differential clock pair and multiple differential data pairs, with the clock center-aligned to the data.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

In the Holoscan Sensor Bridge architecture, sensor data can enter through two distinct ingestion paths:

- MIPI CSI-2 path: CSI-2 sensors interface with the CrossLink-NX FPGA, which uses its integrated MIPI D-PHY Hard IP to receive and decode CSI-2 data streams.
- LVDS path: LVDS-based sensors connect directly to the CertusPro-NX FPGA, which incorporates the LVDS DDR Receiver IP for high-speed deserialization, word alignment, and RAW10/RAW12 data reconstruction.

Both video streams, whether sourced from the CSI-2 or LVDS interface, are ultimately delivered to the CertusPro-NX, where they are packetized and transmitted over a 10GbE Ethernet link to an NVIDIA host for Holoscan processing. This functional split enables the Sensor Bridge to support multiple sensor interface standards while maximizing FPGA resource efficiency. Figure 3.7 shows the LVDS DDR receiver interface used in the CertusPro-NX Sensor-to-Ethernet Bridge board.

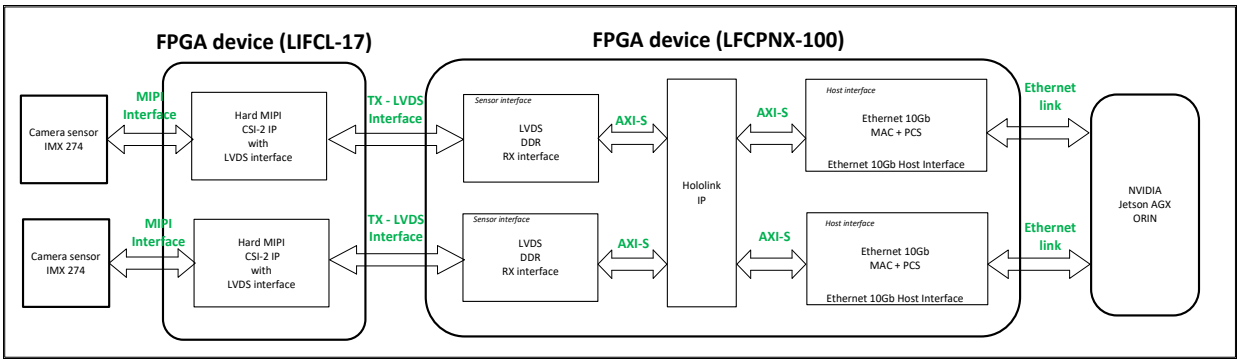



**Figure 3.7. LVDS DDR Receiver Interface on the CertusPro-NX Sensor-to-Ethernet Bridge Board**

## 3.3.3. Host Interface through Ethernet Link

The CertusPro-NX device provides high-speed Ethernet connectivity using integrated IP cores and external transceiver modules. This section describes the available 1GbE, 2.5GbE, and 10GbE Ethernet configurations, along with their key components, integration methods, and functional behavior.

### 3.3.3.1. Ethernet 1GbE Support with 10/100/1000 Mbps SFP Transceiver

The CertusPro-NX device supports 1GbE Ethernet using 10/100/1000BASE-T SFP transceivers. This configuration leverages the Lattice Tri-Speed Ethernet IP, which provides a flexible, robust solution for Ethernet connectivity at 10 Mbps, 100 Mbps, and 1GbE speeds.

Key Components:

- **Tri-Speed Ethernet IP Core**
  Supports MAC + SGMII (SERDES) mode for high-speed serial data transfer. Compliant with IEEE 802.3 and Cisco SGMII standards, it enables auto-negotiation for speed and duplex and includes statistics counters for monitoring.
- **High-Level Block Diagram Components**
  - MAC Core: Handles Ethernet frame processing (CRC checks, filtering) and interfaces with user logic via AXI4-Stream for both transmit and receive paths.
  - SGMII PCS (SERDES): Converts GMII data into serialized differential signals using 8b/10b encoding at 1.25 Gbps. Supports auto-negotiation and link establishment.
  - Host Interface: Configurable via AXI4-Lite, APB, or AHB-Lite for MAC and PCS settings, including speed, duplex mode, and interrupt handling.

The MAC core processes Ethernet frames and passes GMII data to the SGMII PCS, which serializes it for transmission through the SFP transceiver. This architecture ensures reliable, efficient Ethernet communication for embedded applications.

### 3.3.3.2. Ethernet 2.5GbE Support with 2.5GBase-T SFP Transceiver

The CertusPro-NX device supports 2.5GbE Ethernet connectivity, providing a cost-effective and power-efficient solution for moderate-to-high bandwidth applications such as industrial networking, embedded vision, and edge devices. The implementation leverages standard Ethernet IP cores and flexible PHY interfaces to support copper or optical media.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

**Key Components:**

- MAC IP Core
  Handles Ethernet frame-level processing, including frame encapsulation, CRC generation and checking, flow control, and statistics collection. The MAC interfaces with user logic through AXI or Avalon streaming interfaces, enabling seamless integration with the FPGA system fabric.
- PHY IP Core (PHY Interface)
  Implements the physical-layer communication logic required for 2.5GbE operation and connects the MAC to an external PHY or optical module. Common interface options include SGMII or 2500BASE-X, depending on the selected media.
  Features:
  - Encoding/Decoding: Supports 8b/10b encoding for reliable serial data transmission.
  - Auto-Negotiation: Enables automatic speed and duplex configuration.
  - Clock Recovery and Synchronization: Ensures proper alignment between transmit and receive data paths.
  - Link Training and Status Monitoring: Maintains stable Ethernet connectivity.
  - Error Detection: Monitors link integrity and frame errors.
- 2.5GBase-T SFP Transceiver Module
  Provides the physical Ethernet connection to the network.
  Features:
  - Media Support: Compatible with 2.5GBASE-T copper (Cat 5e/ Cat 6) or optical modules using 2500BASE-X.
  - Extended Reach: Up to 100 meters over copper cabling; longer distances supported with fiber.
  - Power Efficient: Lower power consumption compared to 10GbE solutions.
  - Diagnostics: Supports link status, speed indication, and error reporting.

#### 3.3.3.3. Ethernet 10GbE Support with 10GBase-T SFP+ Transceiver

The CertusPro-NX device supports 10GbE Ethernet using SFP+ transceivers, enabling high-speed data communication for bandwidth-intensive applications. The solution integrates multiple IP cores and hardware components.

**Key Components:**

- MAC IP Core
  Manages Ethernet frame-level operations, including encapsulation, CRC generation, and flow control. Interfaces with system logic via AXI or Avalon, depending on the FPGA architecture.
- PCS IP Core (Physical Coding Sublayer)
  Handles low-level physical layer functions and interfaces directly with the SFP+ module.
  Features:
  - Encoding/Decoding: Supports 64b/66b schemes.
  - Scrambling/Descrambling: Maintains signal integrity.
  - Clock Data Recovery (CDR): Aligns incoming data with system clock.
  - Link Synchronization: Establishes and maintains stable link.
  - Error Handling: Optional Forward Error Correction (FEC) for enhanced reliability.
- 10GBase-T SFP+ Transceiver Module
  A compact, hot-swappable module for 10GbE connections.
  Features:
  - Hot-Swappable: Enables module replacement without shutting down the system.
  - Media Support: Compatible with copper and fiber (such as SR for short-range, LR for long-range).
  - Diagnostics: Supports real-time monitoring of temperature, power, and signal quality.
  - Transmission range: Varies from a few meters (copper) to tens of kilometers (fiber).

The MAC and PCS IP cores work together to prepare and transmit Ethernet frames through the SFP+ module. This setup ensures high-speed, low-latency communication suitable for advanced networking applications.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

## 3.4. Multi Camera Frame Synchronization

Multiple camera frame synchronization is achieved through a common synchronized signal generated by the FPGA. Camera sensors and modules from different vendors typically allow an external triggerable signal, FSIN/FSYNC/XVS/XTRIG, to be connected to start capturing the video frame. Synchronizing cameras allows two or more cameras to achieve stereo vision and makes depth computation possible.

In the Holoscan Sensor Bridge FPGA, a module named vsync_gen is provided to generate an IEEE 1588 PTP time-stamping-based signal source to be connected to the external trigger signal of the camera to achieve synchronization.

The PTP Time Stamping synchronized signal allows multiple cameras in the Holoscan Sensor Bridge environment to achieve sub-microsecond synchronization.

The vsync_gen module is clocked by ptp_clk in the system and so all the counters run at a resolution of approximately 10 ns.

Below are the parameters and configurations allowable in the vsync_gen modules, programmable through HSB SDK.

- PERIOD (10, 30, 60, 90, 120 Hz)
- PULSE EXPOSURE (pulse width) programmable in a resolution of 10 ns
- DELAY START programmable in a resolution of 10 ns
- START VALUE (polarity), active high or active low

## 3.5. Dual-Boot Feature

Dual Boot allows an FPGA to store and load multiple configuration bitstreams (images) from an external SPI flash. This capability provides a reliable fallback mechanism, ensuring the system remains operational even if a primary or alternate configuration becomes corrupted. In a typical deployment, the FPGA contains a primary bitstream (default boot image) and a golden bitstream (safe recovery image).

**Key functional behaviors are described below:**

- Normal boot flow
  Upon power-up, the device attempts to load the primary bitstream. If loading fails, the FPGA automatically falls back to the golden bitstream, ensuring a valid configuration is always active.
- Fallback and robustness
  If the FPGA fails to load an alternate bitstream, such as when a bitstream is corrupted, it automatically falls back to the golden bitstream.
- External flash address mapping
  Each bitstream image is assigned to a fixed starting address in SPI flash. These addresses must match those defined in the design (for example boot_addr parameters) and in the Lattice Radiant Deployment Tool configuration.

For more details on the multi-boot feature on the Nexus platform, refer to the Multi-Boot User Guide for Nexus Platform (FPGA-TN-02145).

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# 3.6. Clocking and Reset Scheme

## 3.6.1. CertusPro-NX Versa Board

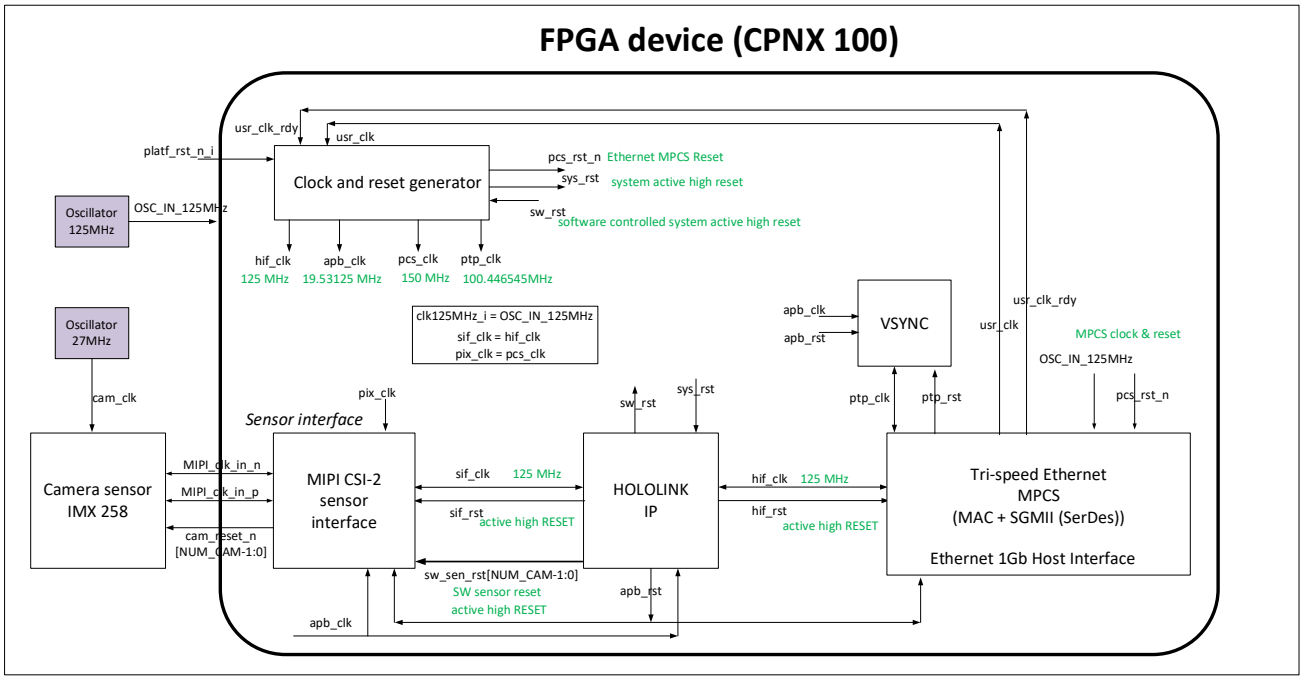



**Figure 3.8. Reset and Clock Domain Block Diagram for Ethernet 1GbE**

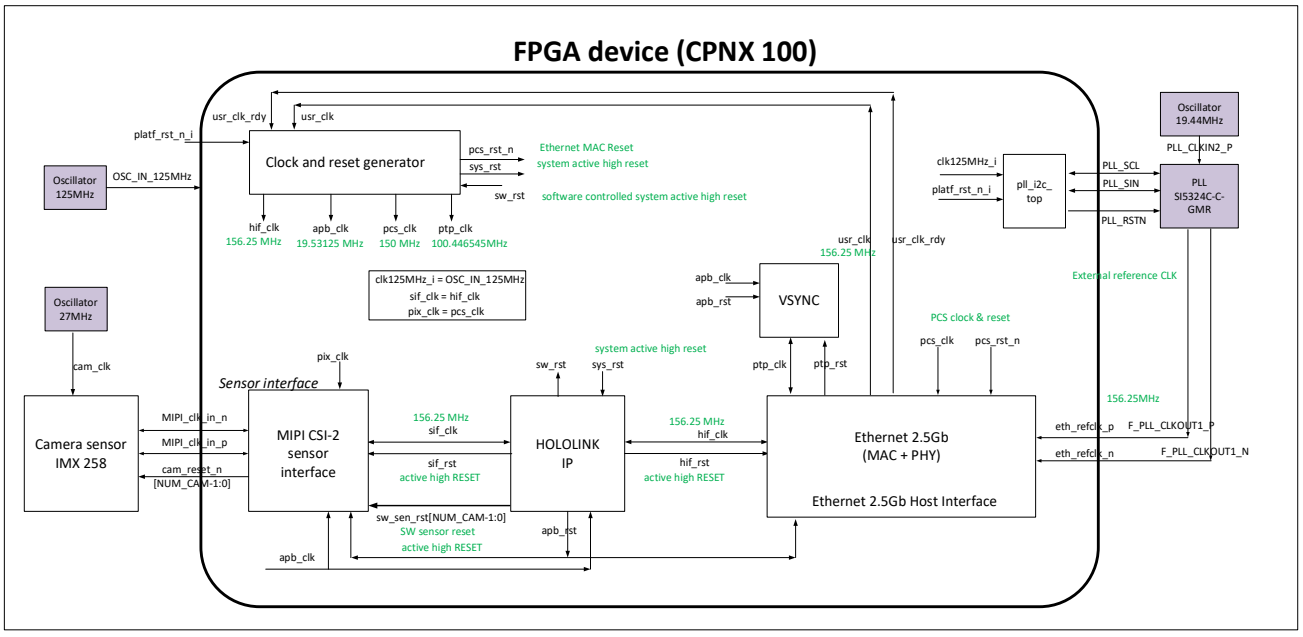



**Figure 3.9. Reset and Clock Domain Block Diagram for Ethernet 2.5GbE**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

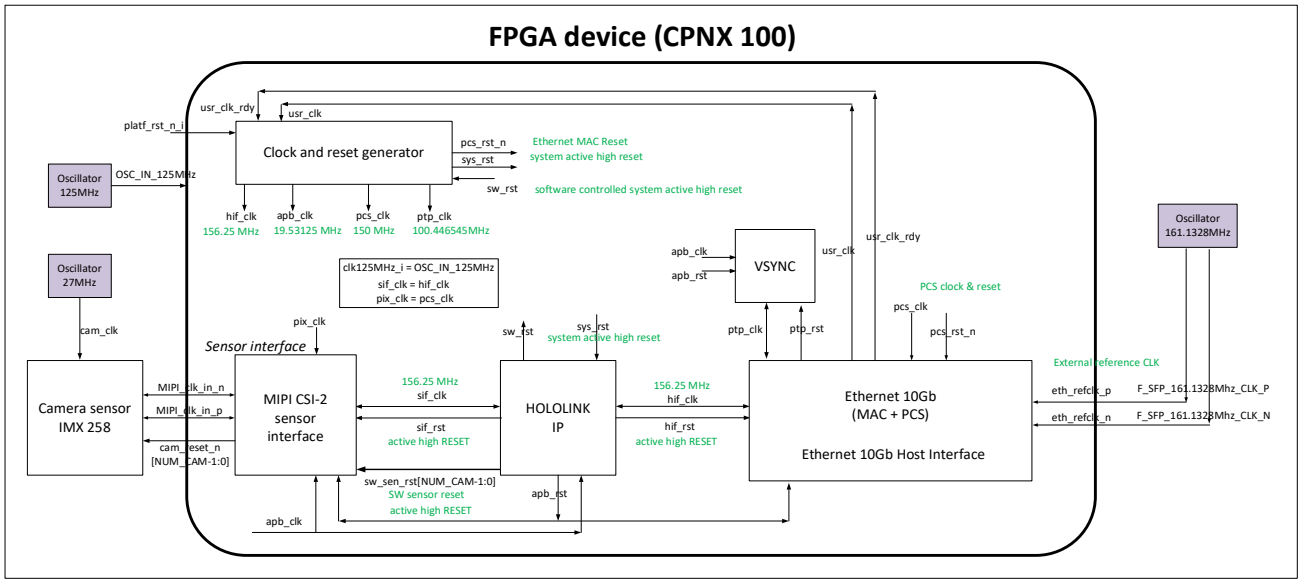



**Figure 3.10. Reset and Clock Domain Block Diagram for Ethernet 10GbE**

The following tables list the clock frequencies and their distribution.

**Table 3.4. Clock Domain Distribution for Reference design with Ethernet 1GbE**

| Clocks | Frequency (MHz) | Description |
|---|---|---|
| usr_clk | 125 | PCS output clock is used as reference clock for the PLL to generate system clocks (such as, apb_clk and hif_clk). |
| OSC_IN_125MHz | 125 | Platform clock input used for MPCS IP and debug hook signals. |
| hif_clk | 125 | Host interface clock used as AXIS clock for host interface. |
| sif_clk | 125 | Directly assigned from hif_clk and is used for the AXIS clock for sensor interface. |
| apb_clk | 19.5313 | APB bus clock. |
| ptp_clk | 100.45 | Precision time protocol clock. |

**Table 3.5. Clock Domain Distribution for Reference design with Ethernet 2.5GbE**

| Clocks | Frequency (MHz) | Description |
|---|---|---|
| usr_clk | 156.25 | PCS output clock is used as reference clock for the PLL to generate system clocks (such as, apb_clk and hif_clk). |
| pcs_clk | 150 | PCS calibration clock. |
| OSC_IN_125MHz | 125 | Platform clock input used for debug hook signals. |
| eth_refclk_p | 156.25 | External ethernet reference clock. |
| hif_clk | 156.25 | Host interface clock used as AXIS clock for host interface. |
| sif_clk | 156.25 | Directly assigned from hif_clk and is used as the AXIS clock for sensor interface. |
| apb_clk | 19.5313 | APB bus clock. |
| ptp_clk | 100.45 | Precision time protocol clock. |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

**Table 3.6. Clock Domain Distribution for Reference design with Ethernet 10GbE**

| Clocks | Frequency (MHz) | Description |
|---|---|---|
| usr_clk | 322.266 | PCS output clock is used as reference clock for the PLL to generate system clocks (such as, apb_clk and hif_clk). |
| pcs_clk | 150 | PCS calibration clock. |
| OSC_IN_125MHz | 125 | Platform clock input used for debug hook signals. |
| eth_refclk_p | 161.1326 | External ethernet reference clock. |
| hif_clk | 156.25 | Host interface clock used as AXIS clock for host interface. |
| sif_clk | 156.25 | Directly assigned from hif_clk and is used as the AXIS clock for sensor interface. |
| apb_clk | 19.5313 | APB bus clock. |
| ptp_clk | 100.45 | Precision time protocol clock. |

**Table 3.7. Reset Distribution applicable for both Ethernet 1GbE, 2.5GbE and 10GbE**

| Clocks | Description |
|---|---|
| sw_sen_rst[NUM_CAM-1:0] | Register-controlled sensor reset. The reset signal width increases based on the number of assigned cameras. Details are discussed in the Customizing the Reference Design section. |
| sw_sys_rst | Register-controlled system reset. Used to reset system-level logic. Also triggers reset for hif_rst, apb_rst, and sif_rst. |
| apb_rst | Reset APB logic. |
| hif_rst | Reset Host logic. |
| sif_rst | Reset Sensor logic. |
| ptp_rst | Precision time protocol reset. |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

## 3.6.2. CertusPro-NX Sensor-to-Ethernet Bridge Board

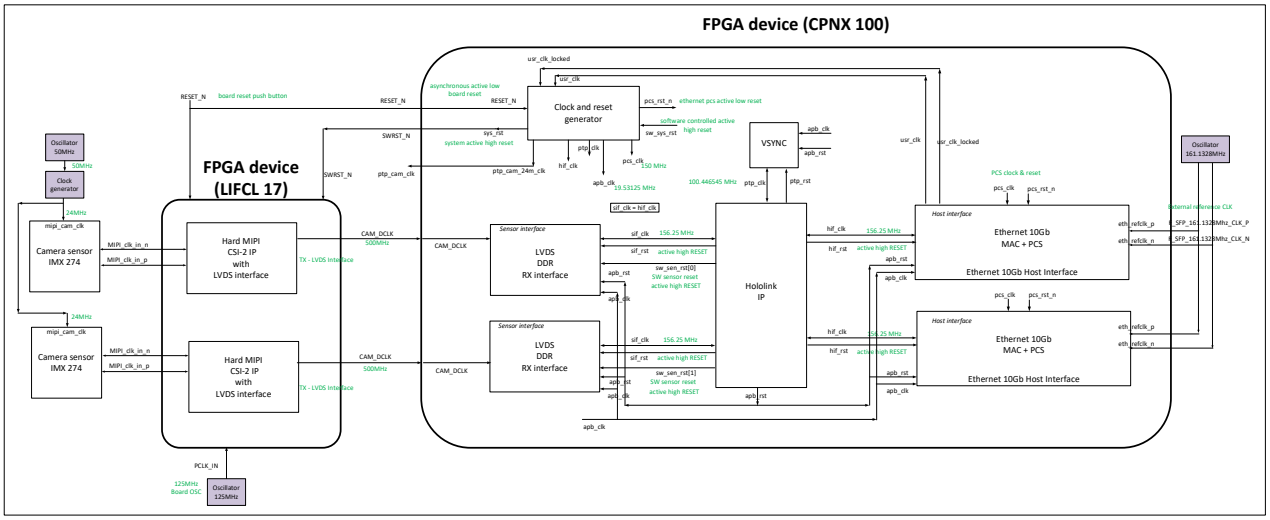



**Figure 3.11. Reset and Clock Domain Block Diagram**

The following table lists the clock frequencies and their distribution.

**Table 3.8. Clock Domain Distribution for CertusPro-NX Sensor-to-Ethernet Bridge Board**

| Clocks | Frequency (MHz) | Description |
|---|---|---|
| mipi_cam_clk | 24 | Camera sensor clock is generated by the platform oscillator and divider. |
| PCLK_IN | 125 | Platform input clock. |
| CAM_DCLK[1:0] | 500 | CAM_DCLK is the LVDS data clock which is generated in CrossLink-NX device via PLL with the reference clock = PCLK_IN. |
| usr_clk | 322.266 | PCS output clock is used as reference clock for the PLL to generate system clocks (such as, apb_clk and hif_clk). |
| pcs_clk | 150 | PCS calibration clock. |
| hif_clk | 156.25 | Host interface clock used as AXIS clock for host interface. |
| sif_clk | 156.25 | Directly assigned from hif_clk and is used as the AXIS clock for sensor interface. |
| apb_clk | 19.5313 | APB bus clock. |
| ptp_clk | 100.45 | Precision time protocol clock. |

**Table 3.9. Reset Distribution for CertusPro-NX Sensor-to-Ethernet Bridge Board**

| Reset | Description |
|---|---|
| RESET_N | Platform push button reset. |
| SWRST_N | System reset which resets the CrossLink-NX device. |
| sw_sys_rst | Register-controlled software system reset. Used to reset system-level logic. Also triggers reset for hif_rst, apb_rst, and sif_rst. |
| apb_rst | Reset APB logic. |
| hif_rst | Reset Host logic. |
| sif_rst | Reset Sensor logic. |
| ptp_rst | Precision time protocol reset. |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# 4. Customizing the Reference Design

## 4.1. Customization Steps for the Reference Design

This section provides a detailed walkthrough of the steps required to configure and to verify the customized reference design. Figure 4.1 illustrates the steps involved to enable customization and support for various customer-specific configurations. However, details on the reference design compilation and implementation are described in the Compiling the Reference Design and Implementing the Reference Design sections.

The screenshots provided in this section are for reference only. The details may vary depending on the version of the IP or software being used.

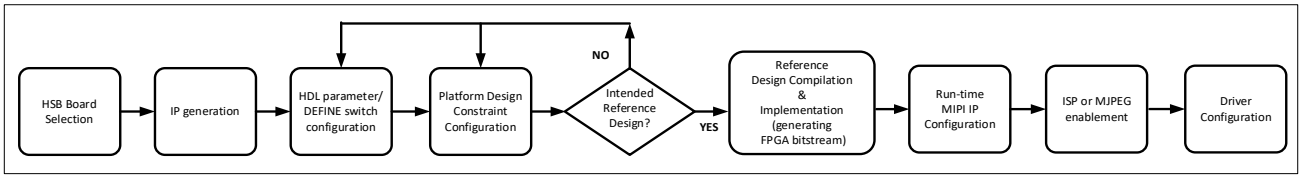



**Figure 4.1. Steps Flow to Customize and Verify the Reference Design**

### 4.1.1. Selecting the Holoscan Sensor Bridge Board

Customization of the reference design depends on the target hardware platform. The design can be implemented on Lattice-provided platforms, such as the CertusPro-NX Versa Board or the CertusPro-NX Sensor-to-Ethernet Bridge Board or on a user-designed custom platform. In this reference design, several HDL parameters are preset for specific platform to simplify integration; these mappings are described in later sections. If you are developing a custom platform, you can define additional parameters as needed to meet your system requirements.

### 4.1.2. Generating the IP

#### 4.1.2.1. MIPI CSI-2 RX D-PHY IP Generation

The reference design enables the MIPI CSI-2 RX D-PHY IP to interface with MIPI CSI-2 camera sensors. Two configuration methods are supported:

**Method 1: Runtime configuration of the MIPI CSI-2 IP**

To support runtime configuration, the MIPI CSI-2 RX D-PHY IP is initially set to a high-speed default configuration. This ensures that the IP can be reconfigured to lower operating settings during system runtime. Table 4.1 lists the default parameters applied for runtime configuration.

A key limitation of this approach is the longer FPGA compilation time. Because the design must accommodate the highest-speed MIPI CSI-2 RX D-PHY settings, the Place-and-Route process requires significantly more time to close timing, resulting in longer bitstream-generation cycles.

**Table 4.1. Selected Runtime Configuration for MIPI CSI-2 RX D-PHY IP**

| MIPI Lane Rate (Mbps) | MIPI Lane Number | MIPI RX D-PHY IP name |
|---|---|---|
| 1,500 | 4 lanes | mipi_csi2_rx_dphy_flex.ipx |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

**Method 2: Preset configuration of the MIPI CSI-2 IP**

The MIPI CSI-2 RX D-PHY IP is generated based on the specifications of the camera sensor being used. The key parameters to configure include:

- Number of data lanes
- MIPI lane rate
- Gear Ratio (fixed to 8)
- MIPI synchronizer clock frequency (fixed to 150 MHz)

These parameters must be correctly set before generating the IP to ensure compatibility with the camera sensor. The gear ratio for the CertusPro-NX family device is fixed at 8, and the MIPI synchronizer clock frequency in the reference design is set to 150 MHz. The reference design also provides preconfigured MIPI CSI-2 RX D-PHY IPs for you to choose from. Table 4.2 lists the available IP configuration options that you can select. The IP is selected based on user settings, as described in the HDL Parameter and DEFINE Switch Configuration section.

**Table 4.2. MIPI CSI-2 RX D-PHY IP configuration options**

| MIPI Lane Rate (Mbps) | MIPI Lane Number | MIPI RX D-PHY IP name |
|---|---|---|
| 720 | 1 lane | mipi_csi2_rx_DPHY_720Mbps_1L.ipx |
| | 2 lanes | mipi_csi2_rx_DPHY_720Mbps_2L.ipx |
| | 4 lanes | mipi_csi2_rx_DPHY_720Mbps_4L.ipx |
| 960 | 2 lanes | mipi_csi2_rx_DPHY_960Mbps_2L.ipx |
| | 4 lanes | mipi_csi2_rx_DPHY_960Mbps_4L.ipx |
| 1,440 | 4 lanes | mipi_csi2_rx_DPHY_1440Mbps_4L.ipx |
| 1,500 | 4 lanes | mipi_csi2_rx_DPHY_1500Mbps_4L.ipx |

#### 4.1.2.2. Byte-to-Pixel Conversion IP

The Byte-to-Pixel conversion IP is a Lattice Semiconductor reference design IP provided in encrypted form and integrated as part of the reference design. The IP is located under the ./radiant/src/ip/ISP/b2p_raw10raw8/ directory.

This module performs clock-domain crossing between the MIPI CSI-2 byte clock domain and the pixel clock domain. Incoming MIPI CSI-2 RAW10 payload data is unpacked and converted into a pixel-formatted data stream suitable for ISP processing.

Limitation: The current implementation supports at a fixed resolution of 1920 × 1080.

#### 4.1.2.3. ISP Reference Design IP

The ISP IP is a Lattice Semiconductor proprietary reference design IP provided in encrypted form and integrated as part of the reference design. The IP is located in the radiant/src/ip/ISP/mISP/ directory.

**Specifications:**

- Input format: Pixel-formatted data stream at 1920 × 1080 (Full HD)
- Output format: YUV422 or YUV422 with AXI-Stream–compliant data
- Clock domain: Pixel clock, fixed at 150 MHz

The ISP IP sub-functionality is user-configurable. To optimize image quality for your environment, tune the ISP configuration and regenerate the ISP IP. Figure 4.2 illustrates the steps involved in developing and deploying the ISP reference design IP in a Holoscan application.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

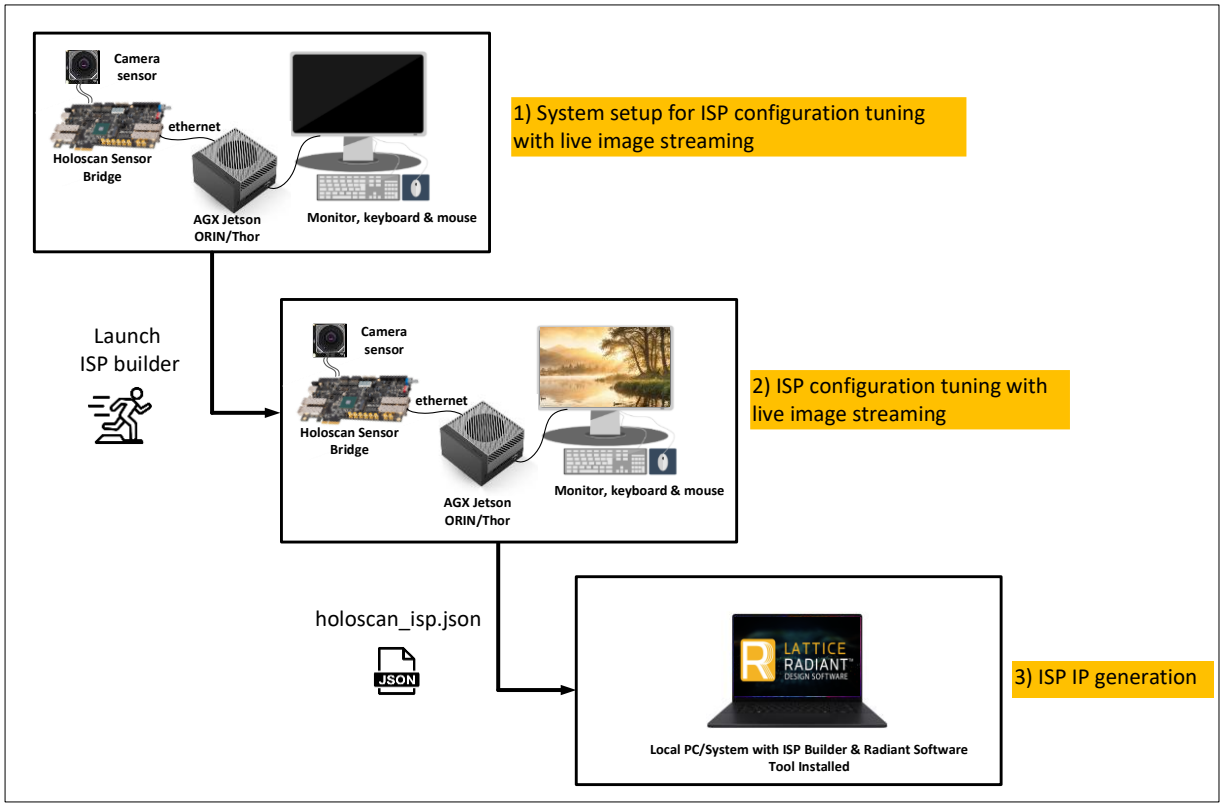



**Figure 4.2. Steps to Generate ISP Lattice Semiconductor Reference Design IP**

#### 4.1.2.4.1 System Setup for ISP Configuration Tuning with Live Image Streaming

Before proceeding, ensure the following prerequisites are met:

- The system must first be prepared for Holoscan system testing using the CertusPro-NX Versa platform with NVIDIA Jetson AGX Orin or Jetson AGX Thor Host. Follow the procedure in the Implementing the Reference Design section of this document.
- The generated FPGA bitstream includes support for the MIPI runtime configuration feature and the ISP hardware module. You can use a demo FPGA bitstream which is provided in the CertusPro-NX Customizable HSB Sensor Interfaces Kit and Reference Design – Source Code.
- ISP tuning is not supported for CertusPro-NX Sensor-to-Ethernet Bridge platform

**Note:** Verify that the MIPI CSI-2 image streams successfully on the Jetson AGX Orin or Thor host before proceeding.

In addition to the system setup for ISP configuration tuning, the LSCC ISP Builder Linux Orin or LSCC ISP Builder Linux Thor software is required. Copy the tool to the NVIDIA Jetson AGX Orin or Thor Host. These tools are included with the reference design. The software is tested on NVIDIA Jetson AGX Orin or Thor Host, which runs on Linux OS. The software tool (lscc_isp_builder_linux_orin) is located in the radiant/src/ip/ISP/mISP/lscc_isp_builder/ directory of the Customizable HSB Sensor Interface reference design.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

#### 4.1.2.4.2 ISP Configuration Tuning with Live Image Streaming

**Step 1. Launch LSCC ISP Builder Linux Orin or LSCC ISP Builder Linux Thor software**

To run the newly copied LSCC ISP build on AGX Orin/Thor for the first time, you must first grant executable permissions using the command below.

```
chmod +x lscc_isp_builder/lscc_isp_builder_linux_orin
chmod +x lscc_isp_builder/lscc_isp_builder_linux_thor
```

By default, the reference design includes a pre-generated ISP IP with JSON configuration file. Use this file as a baseline configuration to improve image streaming quality. The JSON file is in the directory shown Figure 4.3 and described in Table 4.3.

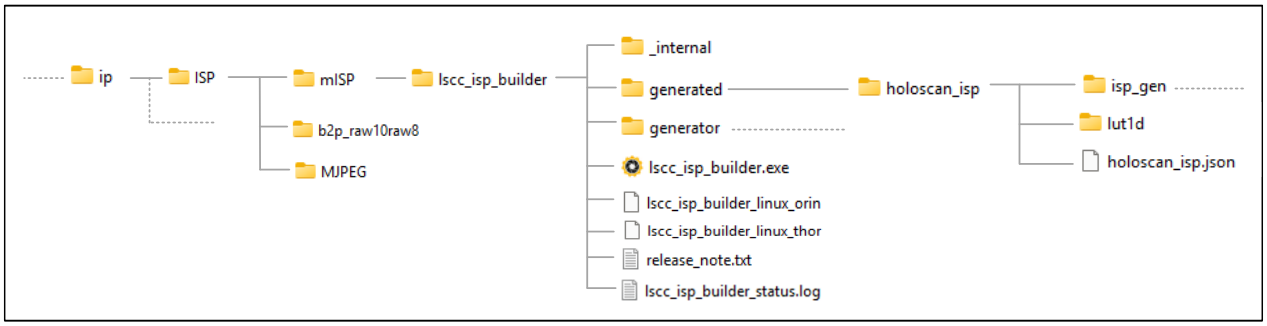



**Figure 4.3. Generated ISP IP Folder Structure**

**Table 4.3. Generated ISP IP Folder Structure Description**

| Folder/File | Description |
|---|---|
| lscc_isp_builder/release_note.txt | Release note about the ISP builder includes version number and quick start steps to execute ISP builder |
| lscc_isp_builder/lscc_isp_builder_linux_orin | Linux executable file used to tune ISP configuration with live image streaming on NVIDIA Jetson AGX Orin Host |
| lscc_isp_builder/lscc_isp_builder_linux_thor | Linux executable file used to tune ISP configuration with live image streaming on NVIDIA Jetson AGX Thor Host |
| lscc_isp_builder/lscc_isp_builder.exe | Executable file used to generate ISP IP using JSON config file as input |
| lscc_isp_builder/generator/ | Folder consists of the ISP IP generator related files |
| lscc_isp_builder/generated/holoscan_isp/isp_gen | The path that defined by user when generating the ISP IP. This folder consists of the generated ISP IP files. |
| lscc_isp_builder/generated/holoscan_isp/holoscan_isp.json | This JSON file will be generated based on the JSON configuration file that is used to generate ISP IP. |

Run the LSCC ISP Builder Linux Orin or LSCC ISP Builder Linux Thor software on the host. When the GUI launches, change the camera streaming option to Network Stream to enable ISP live stream tuning. Load the pre-generated JSON configuration file as the initial configuration (see Step 1 in Figure 4.4).

**Input/Output specification:**

- Input: Bayer-patterned RAW10 data
- Output: YUV422 format
- Target resolution: 1920 × 1080 (Full HD)

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

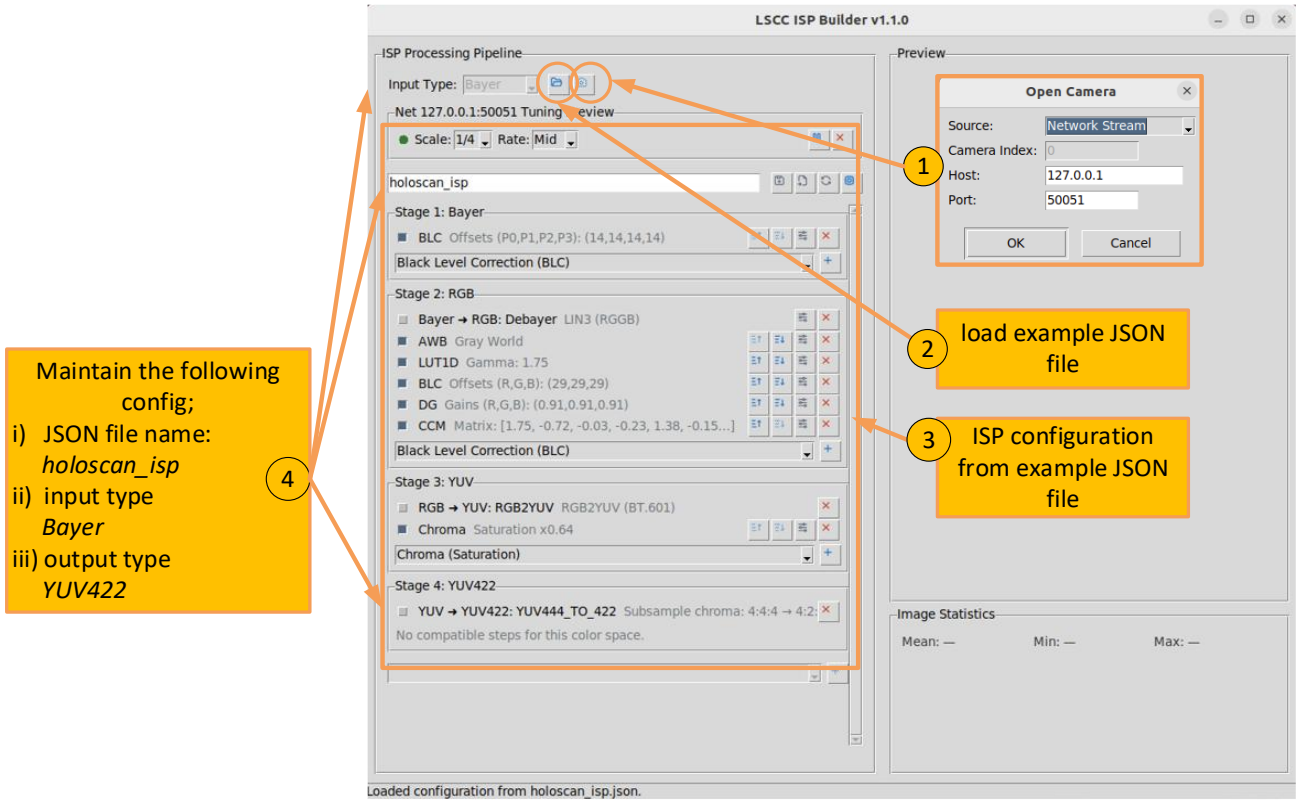



**Figure 4.4. LSCC ISP Builder Linux Orin GUI.**

### Step 2. Enable Live Image Streaming Preview

Ensure Step 1. is complete before proceeding. Run the following commands on the Jetson AGX Orin or Thor host terminal:

```
cd holoscan-sensor-bridge
sh docker/build.sh --igpu
xhost +
sh docker/demo.sh
python3 ./examples/linux_socketop_imx258.py --focus=-200
```

A live image streaming preview is displayed on the GUI as shown in Figure 4.5.

### Step 3. Tune ISP Configuration

Use the GUI to add or remove ISP processing functions and adjust the parameters of each ISP function using the available function icons (see Figure 4.5):

**Configure:** Click the configure button ( ) beside a selected step to adjust its parameters.

**Reorder:** C the Move Up ( )and Move Down ( ) controls to modify processing order.

**Add:** Select a step from the drop-down menu at the bottom of each color space and click Add ( ).

**Remove:** Click the Remove ( ) button to delete an unneeded step.

The live streaming image serves as the benchmark for evaluating and optimizing ISP processing functionality as enhancements are applied and tuned.

**Note:** A very low frame rate is used during ISP image streaming tuning. Once the desired ISP configuration is finalized, save the ISP settings to a JSON file for IP generation.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

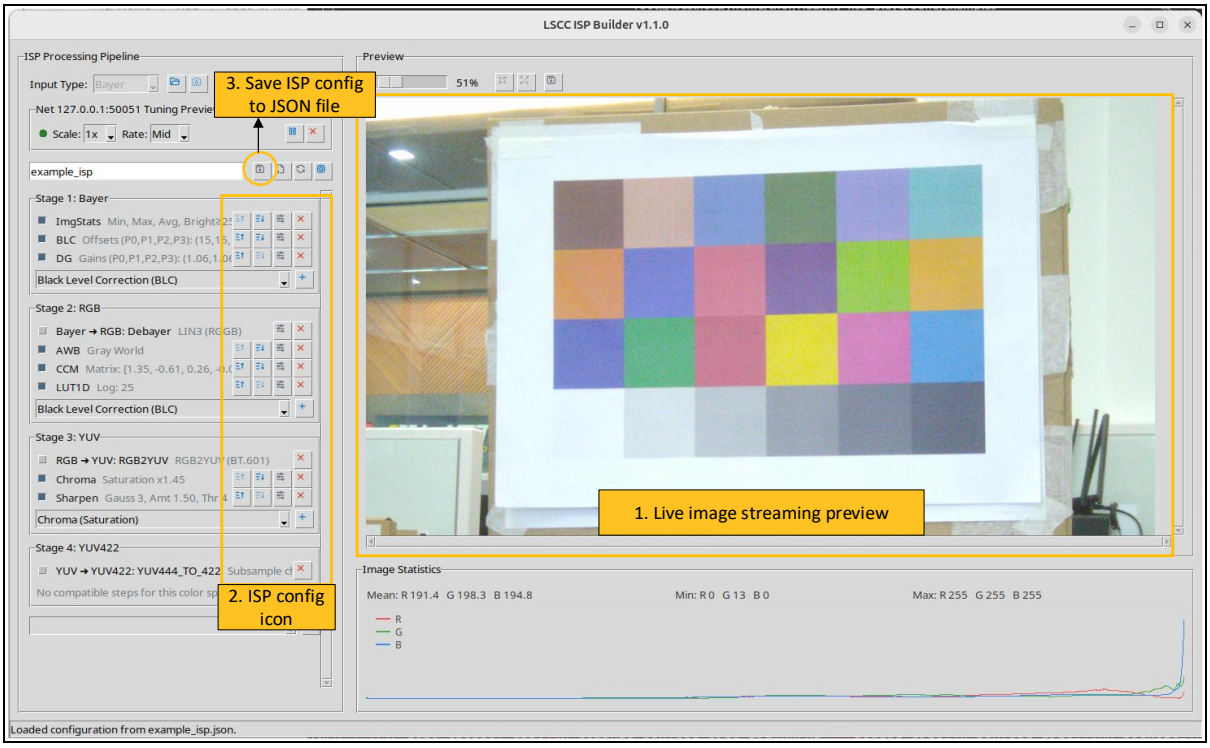



**Figure 4.5. ISP builder GUI with Live Image Streaming**

#### 4.1.2.4.3 ISP IP Generation

**Note:** The LSCC ISP Builder (lscc_isp_builder.exe) tool must run on a local PC with Windows OS and the Lattice Radiant software installed. The tool is included as part of the reference design and is located in: `/radiant/src/rtl/ip/ISP/lsc_isp_builder/` directory.

**To generate the ISP IP:**

1. Launch lscc_isp_builder.exe on the local PC.
2. When the GUI opens, load the configured JSON file (holoscan_isp.json). The configured ISP settings are displayed in the GUI, as shown in Figure 4.6.
3. Click **Generate Configuration**. A pop-up window appears as shown in Figure 4.7.
4. Specify the Lattice Radiant software installation path and the output directory for the generated ISP IP. The recommended output path is `radiant\src\ip\ISP\mISP\lscc_isp_builder\generated\holoscan_isp\isp_gen\`.
5. Click **Generate RTL** to generate the ISP IP.

**Note:** Choosing a path other than the recommended directory requires changes to the Lattice Radiant project file.

The generated ISP IP is produced in encrypted format. Output log files, and the JSON file used for generation are placed in the generated folder as shown in Figure 4.3

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

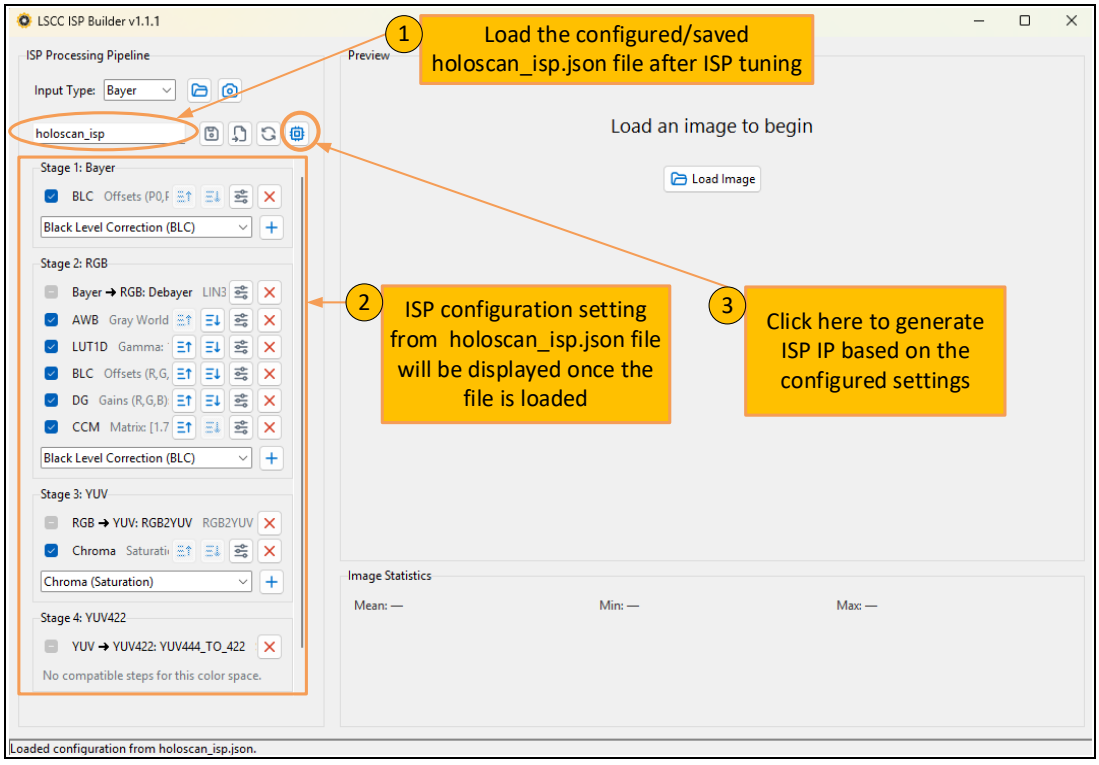



**Figure 4.6. LSCC ISP BuilderGUI preview loading the configured JSON file.**

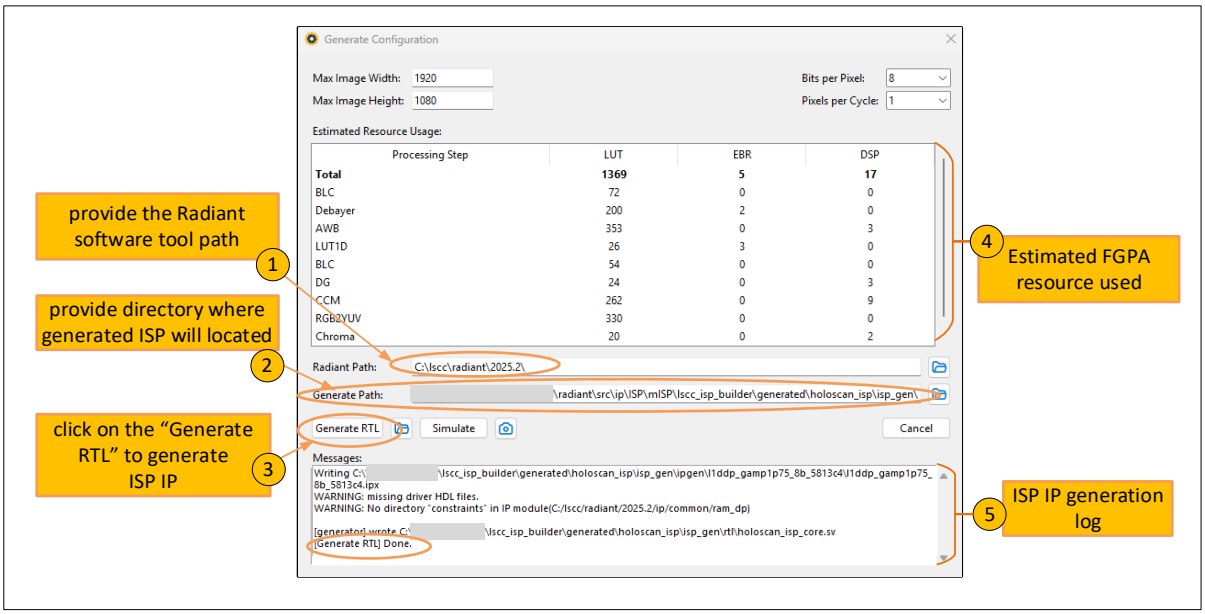



**Figure 4.7. Generate Configuration window**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

#### 4.1.2.4. MJPEG Reference Design IP

The MJPEG IP is a Lattice Semiconductor reference design IP provided in encrypted form and integrated as part of the reference design. Its purpose is to compress YUV422 video streams to reduce the overall video bandwidth for systems with low throughput requirement. The RTL is located in the radiant/src/ip/ISP/MJPEG/ directory.

**Specifications and constraints:**

- Input format: YUV422 only
- Clock domain: Single pixel clock, fixed at 150 MHz
- Default resolution: 1920 × 1080 (Full HD)
- Compression: Baseline JPEG using standard quantization and Huffman tables per JPEG ITU-T Recommendation T.81.
- Compression ratio: Approximately 40:1 for natural scenes.

**Note:** The underlying theory of MJPEG operation is outside the scope of this document.

Due to system clock limitations on the targeted FPGA device, the pixel clock is fixed at 150 MHz. When using a four-lane MIPI sensor, the MIPI lane rate must be 375 MHz or less to support Full HD resolution.

**Note:** The MJPEG format is currently limited to an output frame rate of 30 fps in this release.

#### 4.1.2.5. LVDS Image Sensor Receiver Interface IP

The Lattice LVDS (Low Voltage Differential Signaling) DDR Receiver IP is specifically used for the CertusPro-NX Sensor-to-Ethernet Bridge board. The Lattice LVDS DDR Receiver IP configuration is shown in Figure 4.8.

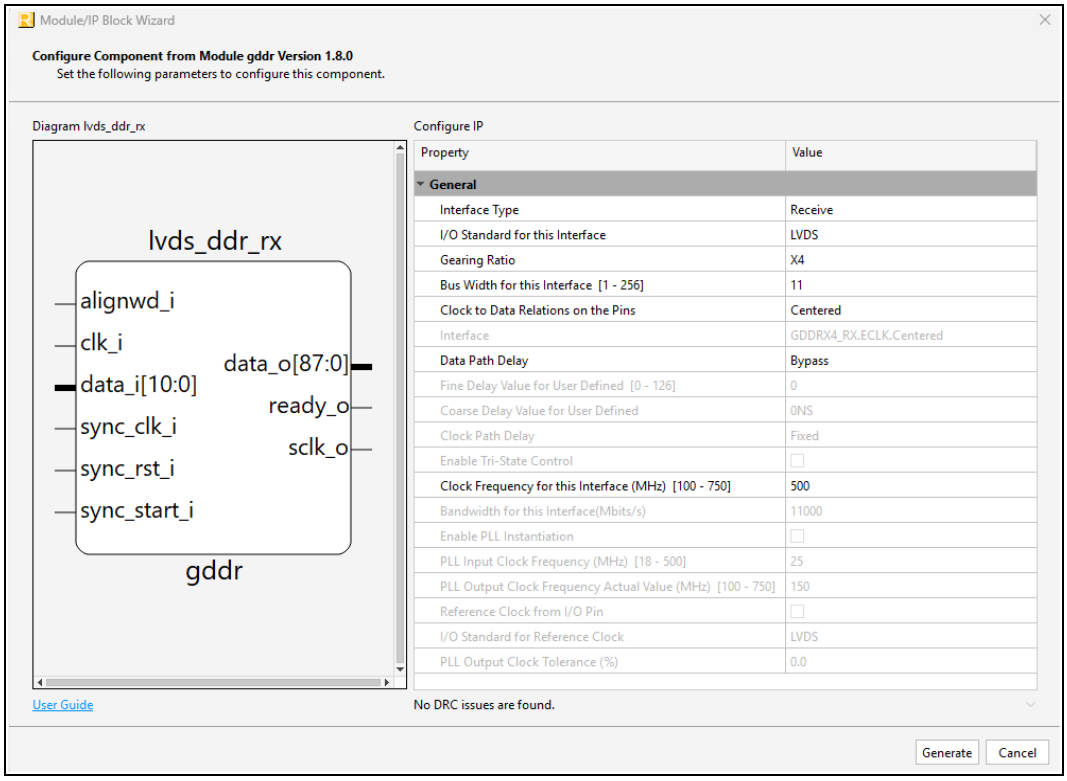



**Figure 4.8. MIPI CSI-2 RX D-PHY IP Generation Module**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

#### 4.1.2.6. Hololink IP

The Hololink IP is owned and maintained by NVIDIA. The latest Hololink IP source code is available on GitHub at the following link, GitHub - nvidia-holoscan/holoscan-sensor-bridge at 2.7.0. Currently, the reference design integrates the latest available Hololink IP, which is version 2606.

#### 4.1.2.7. Ethernet IP Generation

Specific Ethernet IPs are required to support 1GbE, 2.5GbE, or 10GbE link speeds in the CertusPro-NX FPGA device. To support a 1GbE link speed, the Tri-Speed Ethernet v2.1.0 (MAC + SGMII (SERDES)) IP is used. To support Ethernet 2.5GbE link speed, the Ethernet (MAC + PHY) IP v1.2.0 is used. To support Ethernet 10GbE link speed, the Ethernet PCS IP Core v1.3.0 and MAC IP Core v1.2.0 are used. The Ethernet Lane ID must be set to 6 (Lane ID = 6) to meet the platform requirements for the CertusPro-NX Versa Evaluation Board before generating the IP. For the CertusPro-NX Sensor-to-Ethernet Bridge board, the Ethernet PCS lane ID must be set to 2 and 3 respectively as two Ethernet ports are used. Figure 4.9 shows the Tri-speed Ethernet v2.1.0 (MAC + SGMII (SERDES)) IP generation for 1GbE Ethernet link-speed support. Figure 4.10 shows the Ethernet 2.5GbE (MAC + PHY) IP generation. Figure 4.11 shows the Ethernet 10GbE MAC IP generation, and Figure 4.12 shows the Ethernet 10GbE PCS IP generation module GUI.

**Note:** IP versions may vary as they are updated periodically. The figures below are for illustration purposes.

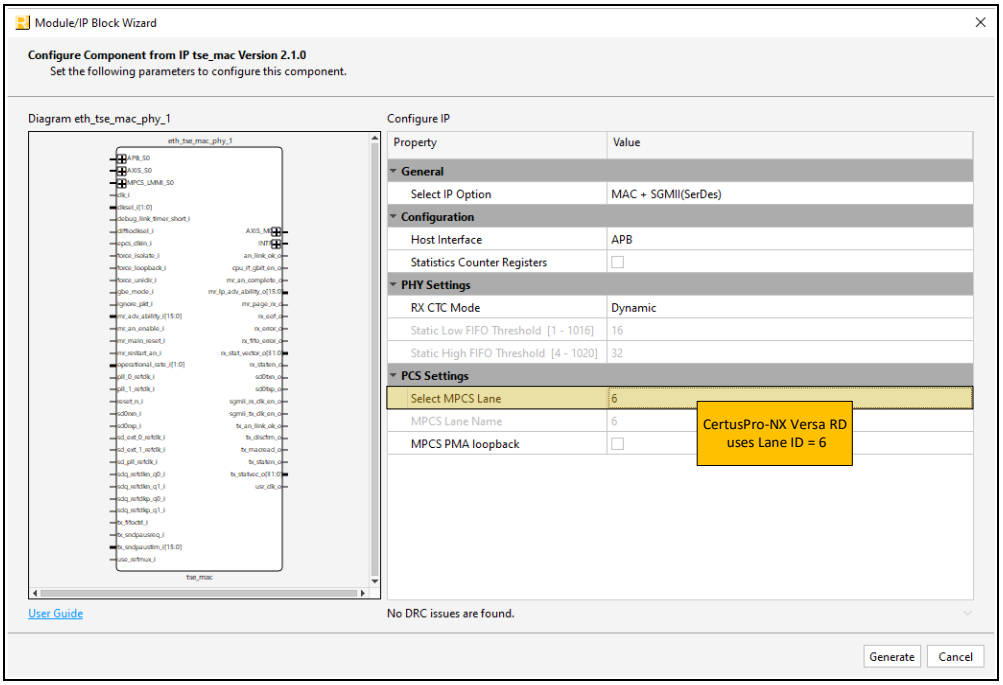



**Figure 4.9. Ethernet 1GbE (MAC + SGMII SERDES) IP Generation**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

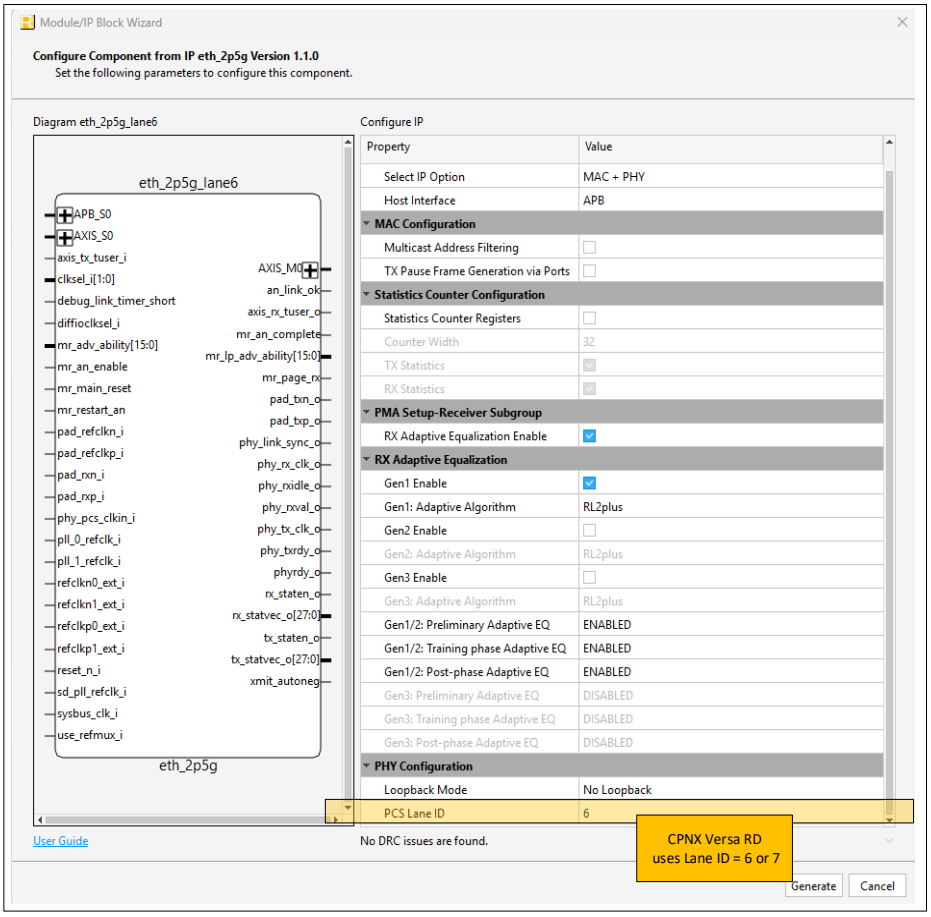



**Figure 4.10. Ethernet 2.5GbE (MAC + PHY) IP Generation**

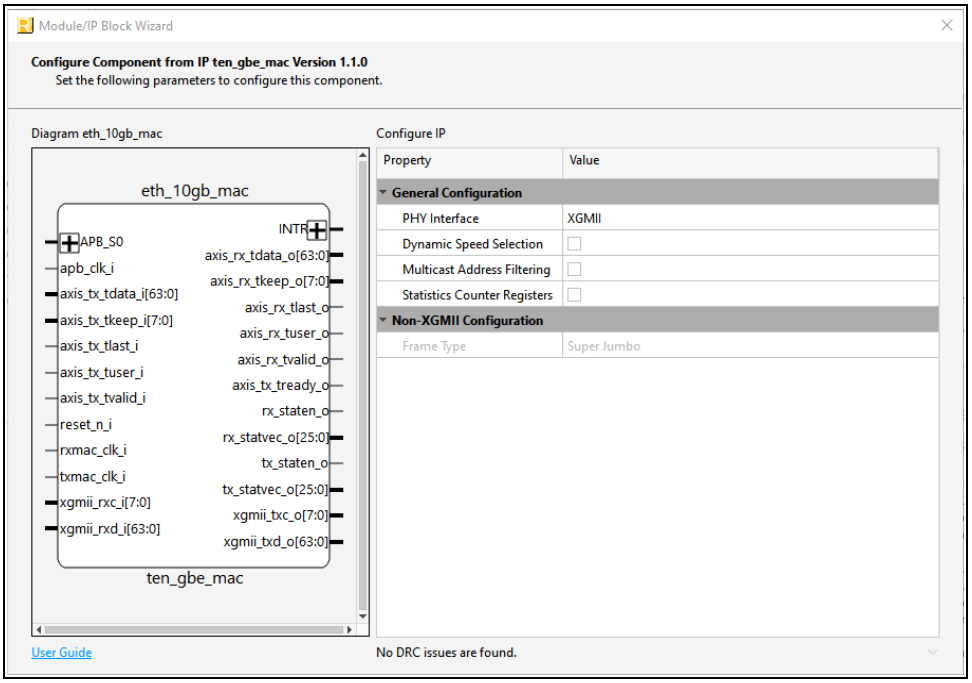



**Figure 4.11. Ethernet 10GbE MAC IP Generation**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

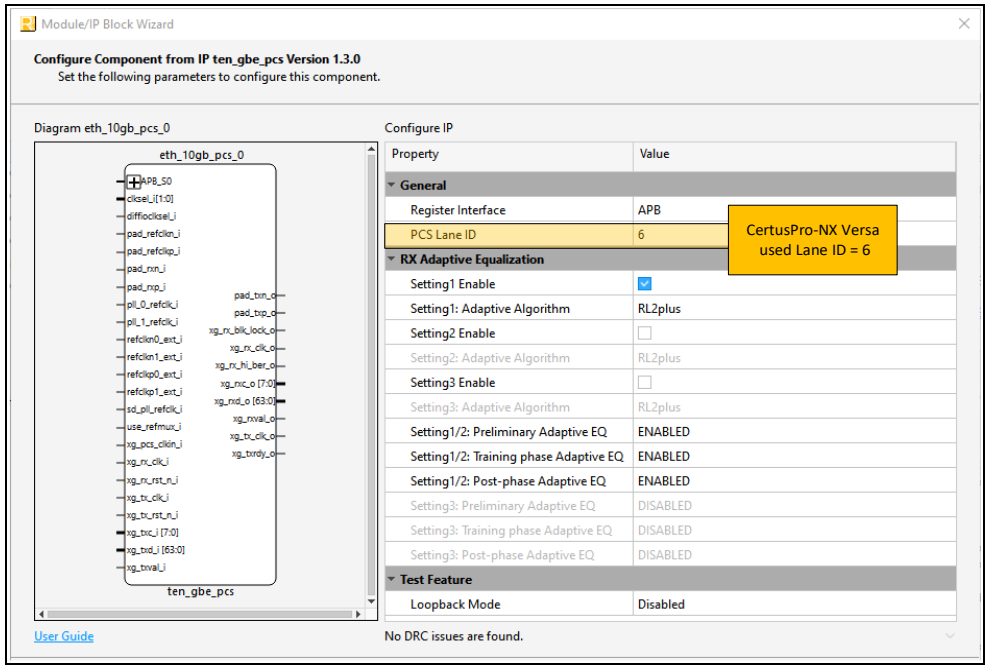



**Figure 4.12. Ethernet 10GbE PCS IP Generation**

### 4.1.3. Configuring the HDL Parameter and DEFINE Switch

Configure the HDL parameters and DEFINE switches to reflect the desired hardware setup. These configurations modify the HDL design prior to compilation and affect the following aspects:

- Number of camera sensor interfaces
- MIPI CSI-2 configuration
  - Method 1: Runtime configuration of the MIPI CSI-2 IP
  - Method 2: Preset configuration of the MIPI CSI-2 IP (configure number of lanes per camera sensor and MIPI lane rate used for each camera sensor)
- Image preprocessing
  - ISP decoding or MJPEG compression is enabled by default
- Number of camera reset outputs
- I2C control peripherals for each camera sensor
- AXI-Stream ports connecting the sensor interface to the Hololink IP
- AXI-Stream ports connecting the Hololink IP to the host interface (Ethernet link)
- Ethernet link speed: 1GbE, 2.5GbE, or 10GbE
- Number of Ethernet links used as host interface
- Ethernet PCS lane ID configuration

These changes must be made in the DEFINE file to ensure proper integration. The DEFINE file differs for each HSB platform and is located at the following paths:

```
<project_name>/fpga_lfcpnx/radiant/src/rtl/hsb_cpnx/top/versa_top/HOLOLINK_def.svh
<project_name>/fpga_lfcpnx/radiant/src/rtl/hsb_cpnx/top/2chip_top/HOLOLINK_def.svh
```

Configurations such as cameras reset outputs, I2C camera controls, and AXI-Stream ports for the sensor/host interface are automatically configured based on the number of camera sensors and assigned Ethernet 1GbE, 2.5GbE or 10GbE links. Figure 4.13 shows the DEFINE switches that need to be changed in HOLOLINK_def.svh file. These settings must be carefully configured to ensure the HDL design reflects the intended hardware setup and supports the required camera sensor interfaces.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.




**Figure 4.13. Define Switch Setting in HOLOLINK_def.svh File**

**Note:** Figure 4.13 screenshot shows an example setting of four camera sensors. If more or less cameras are required for the intended reference design, the array size for USR_NUM_RX_LANE[`NUM_CAM-1:0] and USR_MIPI_RATE_Mbps[`NUM_CAM-1:0] change. Therefore, the data assigned to it are trimmed or added based on the intended camera number usage.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### 4.1.4. Configuring the Platform Design Constraint

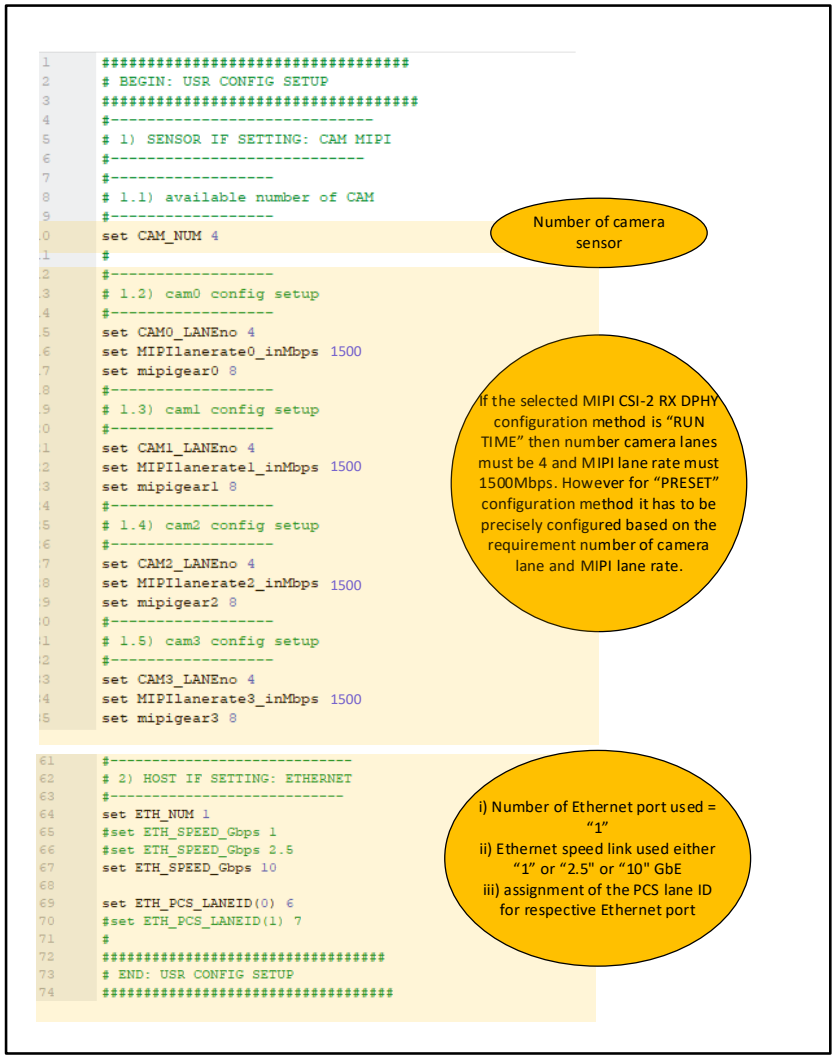
Once the HDL design configuration is finalized, platform design constraints in the PDC file are updated. The reference design includes PDC template files; their locations are listed in Table 4.4.

**Table 4.4. PDC Template Files**

| Reference design | PDC template file location |
|---|---|
| CertusPro-NX Versa Board | <project_name>/fpga_lfcpnx/radiant/src/constraints/hsb_cpnx_versa.pdc |
| CertusPro-NX Sensor-to-Ethernet Bridge Board | <project_name>/fpga_lfcpnx/radiant/src/constraints/hsb_cpnx_2chip.pdc |

Use the runtime MIPI CSI-2 RX D-PHY configuration method only when the PDC file sets the MIPI parameters as follows:

- MIPI lane rate: 1,500 Mbps
- MIPI lane count: 4




```
#####################################
# BEGIN: USR CONFIG SETUP
#####################################
#-----------------------------
# 1) SENSOR IF SETTING: CAM MIPI
#-----------------------------
#------------------
# 1.1) available number of CAM
#------------------
set CAM_NUM 4
#
#------------------
# 1.2) cam0 config setup
#------------------
set CAM0_LANEno 4
set MIPIlanerate0_inMbps 1500
set mipigear0 8
#------------------
# 1.3) cam1 config setup
#------------------
set CAM1_LANEno 4
set MIPIlanerate1_inMbps 1500
set mipigear1 8
#------------------
# 1.4) cam2 config setup
#------------------
set CAM2_LANEno 4
set MIPIlanerate2_inMbps 1500
set mipigear2 8
#------------------
# 1.5) cam3 config setup
#------------------
set CAM3_LANEno 4
set MIPIlanerate3_inMbps 1500
set mipigear3 8

#-----------------------------
# 2) HOST IF SETTING: ETHERNET
#-----------------------------
set ETH_NUM 1
#set ETH_SPEED_Gbps 1
#set ETH_SPEED_Gbps 2.5
set ETH_SPEED_Gbps 10

set ETH_PCS_LANEID(0) 6
#set ETH_PCS_LANEID(1) 7
#
#####################################
# END: USR CONFIG SETUP
#####################################
```

**Figure 4.14. PDC File Setting for CertusPro-NX Versa Board**

Figure 4.14 shows the PDC file settings for the number of camera sensors, data lane, and lane rates used for each camera sensor, as well as the number of Ethernet ports and link speed, configurable to 1GbE, 2.5GbE, or 10GbE. If fewer cameras are used, the lane numbers for undefined cameras will be ignored. For example, if the number of cameras is 1, the lane number values set for cameras 2, 3, and 4 will be ignored.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

The steps to update the platform design constraints include:

- Pin assignment at the top level of the HDL design
- Defining timing constraints

Proper configuration of these constraints is essential to produce the intended reference design. However, the PDC file template provided with the reference design allows users to choose 1, 2, 3, or 4 camera sensors.

#### 4.1.4.1. Pin Assignment for MIPI CSI-2 Camera Sensor Interface

This section covers only the MIPI CSI-2 camera sensor interface pin assignments. The pins that need to be updated at the HDL top level are the camera MIPI clock lane, camera MIPI data lane, camera sensor reset, and camera I2C control. Pay special attention to the MIPI data lane assignment due to the use of two-dimensional array coding method at the HDL top-level design.

MIPI clock and data lanes are differential signals, and the pins assigned to them must be high-speed differential I/O types. For CertusPro-NX FPGAs, the typical I/O type used for MIPI clock and data lanes is LVDS (Low Voltage Differential Signaling) I/O pairs. Special handling is required when setting pin assignments for MIPI data lanes, as the reference design's top level uses a two-dimensional array coding method. This affects how the assignments are defined and mapped in PDC file. Table 4.5 illustrates the correct pin assignment approach under these conditions in the PDC file up to four cameras.

**Table 4.5. PDC Pin Assignment for 2D Array for MIPI Data Lane**

| Camera | SystemVerilog Pin Name | PDC Mapped Pin Name |
|---|---|---|
| Camera 0 | mipi_cam_data_p[0][0] | mipi_cam_data_p[0] |
| | mipi_cam_data_n[0][0] | mipi_cam_data_n[0] |
| | mipi_cam_data_p[0][1] | mipi_cam_data_p[1] |
| | mipi_cam_data_n[0][1] | mipi_cam_data_n[1] |
| | mipi_cam_data_p[0][2] | mipi_cam_data_p[2] |
| | mipi_cam_data_n[0][2] | mipi_cam_data_n[2] |
| | mipi_cam_data_p[0][3] | mipi_cam_data_p[3] |
| | mipi_cam_data_n[0][3] | mipi_cam_data_n[3] |
| Camera 1 | mipi_cam_data_p[1][0] | mipi_cam_data_p[4] |
| | mipi_cam_data_n[1][0] | mipi_cam_data_n[4] |
| | mipi_cam_data_p[1][1] | mipi_cam_data_p[5] |
| | mipi_cam_data_n[1][1] | mipi_cam_data_n[5] |
| | mipi_cam_data_p[1][2] | mipi_cam_data_p[6] |
| | mipi_cam_data_n[1][2] | mipi_cam_data_n[6] |
| | mipi_cam_data_p[1][3] | mipi_cam_data_p[7] |
| | mipi_cam_data_n[1][3] | mipi_cam_data_n[7] |
| Camera 2 | mipi_cam_data_p[2][0] | mipi_cam_data_p[8] |
| | mipi_cam_data_n[2][0] | mipi_cam_data_n[8] |
| | mipi_cam_data_p[2][1] | mipi_cam_data_p[9] |
| | mipi_cam_data_n[2][1] | mipi_cam_data_n[9] |
| | mipi_cam_data_p[2][2] | mipi_cam_data_p[10] |
| | mipi_cam_data_n[2][2] | mipi_cam_data_n[10] |
| | mipi_cam_data_p[2][3] | mipi_cam_data_p[11] |
| | mipi_cam_data_n[2][3] | mipi_cam_data_n[11] |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

| Camera | SystemVerilog Pin Name | PDC Mapped Pin Name |
|---|---|---|
| Camera 3 | mipi_cam_data_p[3][0] | mipi_cam_data_p[12] |
| | mipi_cam_data_n[3][0] | mipi_cam_data_n[12] |
| | mipi_cam_data_p[3][1] | mipi_cam_data_p[13] |
| | mipi_cam_data_n[3][1] | mipi_cam_data_n[13] |
| | mipi_cam_data_p[3][2] | mipi_cam_data_p[14] |
| | mipi_cam_data_n[3][2] | mipi_cam_data_n[14] |
| | mipi_cam_data_p[3][3] | mipi_cam_data_p[15] |
| | mipi_cam_data_n[3][3] | mipi_cam_data_n[15] |

#### 4.1.4.2. Timing Constraint for MIPI CSI-2 Camera Sensor Interface

Using the PDC template from reference design, you can configure the timing constraint for the MIPI CSI-2 camera sensor interface by assigning the MIPI lane rate, as shown in Figure 4.14. In the PDC file, this value will be used to assign the timing constraint required for the MIPI clock input.

### 4.1.5. Verifying the Reference Design Configuration

Verify that the configured settings produce the target reference design. There are two steps required for you to verify: first, review the Lattice Radiant software hierarchy list to confirm the selected camera number and camera configuration. Second, check the clock summary details in the timing analysis report to ensure the assigned timing constraint matches the target configuration. Figure 4.15 shows a screenshot of the reference design implemented on the CertusPro-NX Versa Board. If the verified design is not the target reference design, review the HDL parameters, DEFINE switch configuration, and PDC configuration.

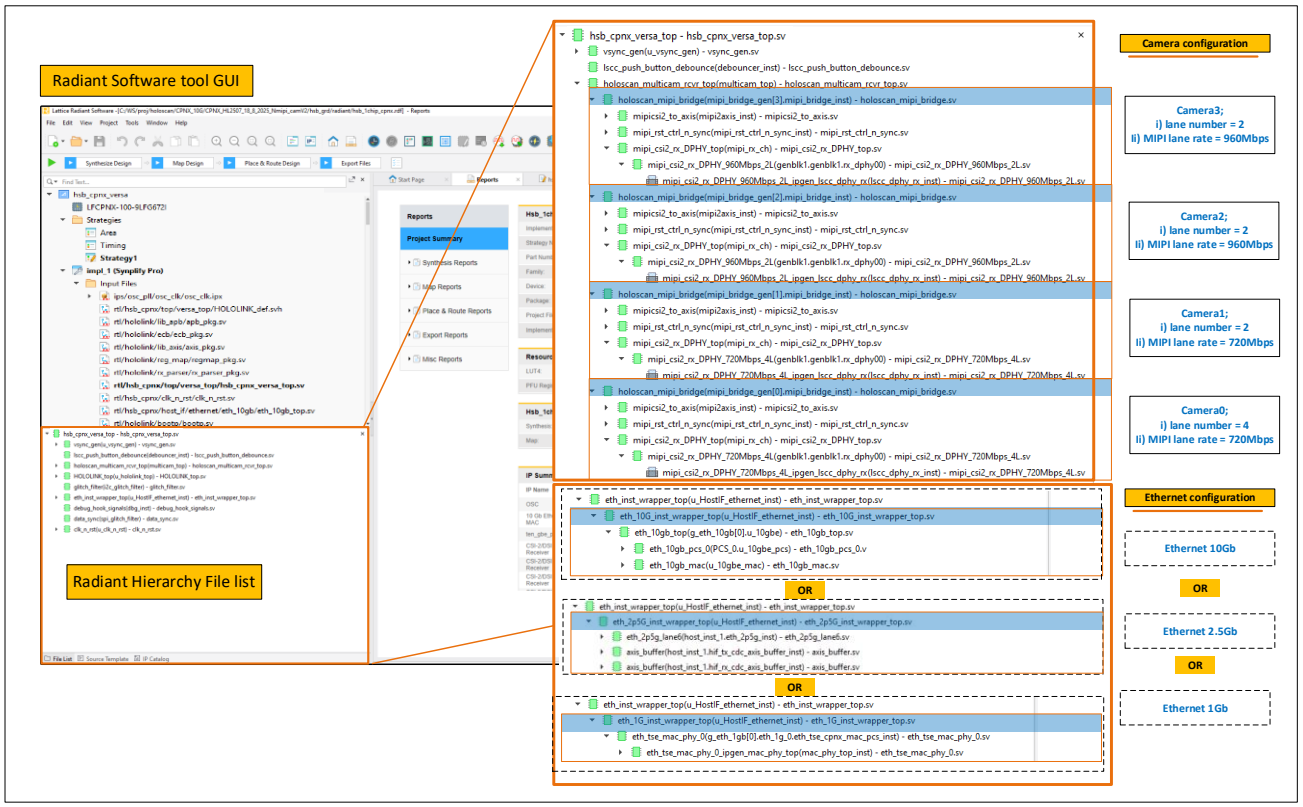



**Figure 4.15. Screenshot of Lattice Radiant Software Hierarchy File List**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### 4.1.6. Configuring the MIPI CSI-2 D-PHY IP at Runtime

The MIPI CSI-2 RX D-PHY IP supports runtime configuration of the MIPI lane count and lane rate through the software driver API. Use the following steps to configure the runtime MIPI CSI-2 RX D-PHY IP:

1. Enabling Runtime Configuration

   Enable runtime control by adding the RUNTIME_MIPI preprocessor macro to:

   `<project_name>/fpga_lfcpnx/radiant/src/rtl/hsb_cpnx/top/versa_top/HOLOLINK_def.svh`

   See Figure 4.16.

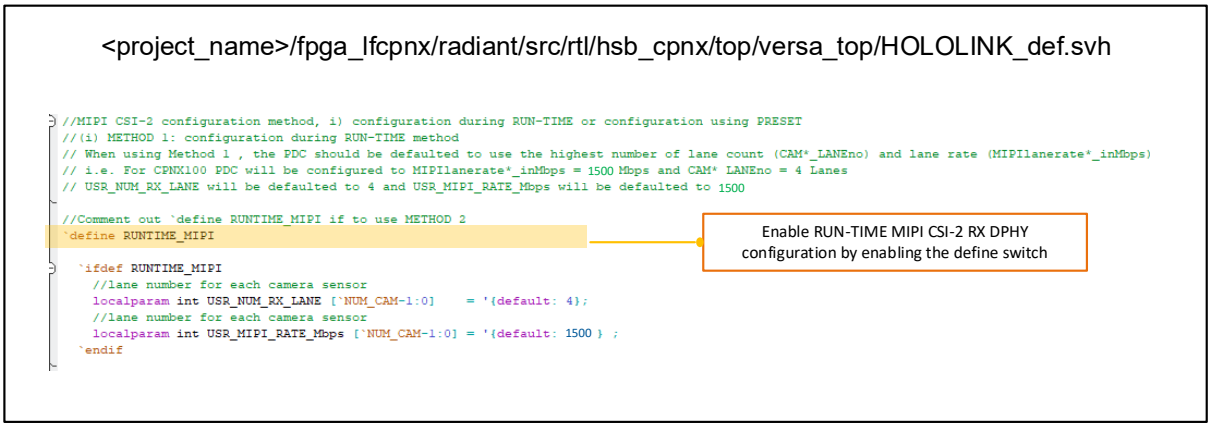



**Figure 4.16. Preprocessor Macro to Enable Runtime MIPI CSI-2 D-PHY IP**

2. Configuring Runtime Parameters

   After runtime configuration is enabled, use the configure_mipi_lane driver API to set the desired lane count and lane rate through the Control and Status Register (CSR). Figure 4.17 shows an example configuration for a four-lane, 720-Mbps setup using the script located at:

   `<holoscan-sensor-bridge>/examples/linux_imx258_player.py.`

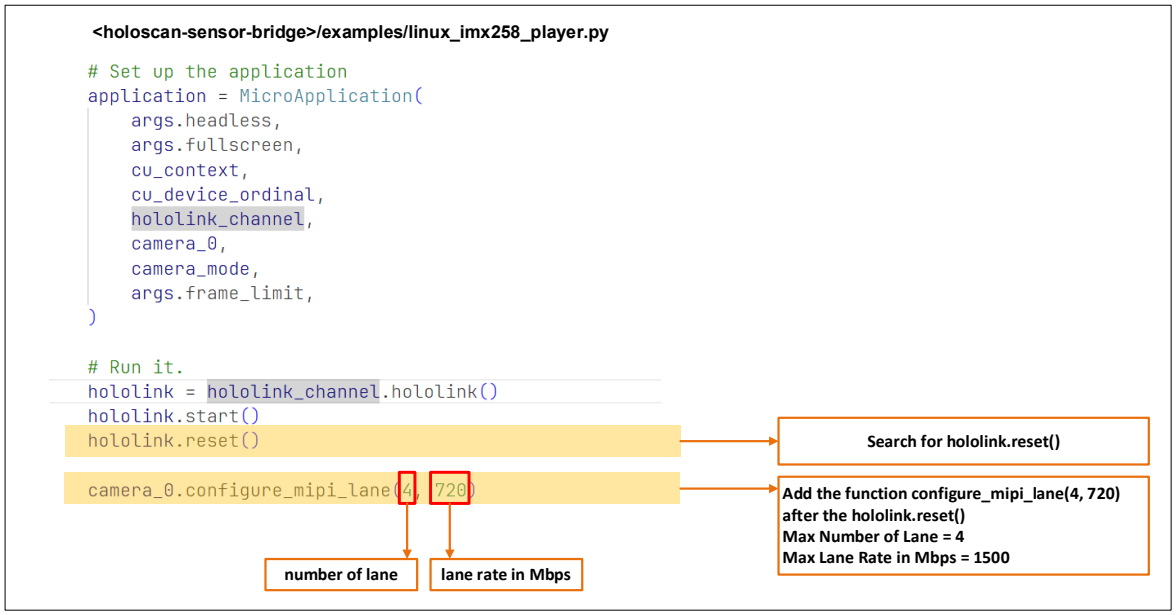



**Figure 4.17. Configuring Runtime MIPI using Software Driver API**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### 4.1.7. Enabling ISP or MJPEG Feature at Runtime

By default, ISP processing and MJPEG compression are supported in hardware. You can confirm this by checking that the parameters ISP_EN and MJPEG_EN are set to 1 in the HOLOLINK_def.svh file. To conserve FPGA resources, you can disable these parameters (set to 0) to meet your application requirements.

The ISP_EN parameter controls whether the ISP hardware instance is enabled, while MJPEG_EN controls the MJPEG compression hardware. Table 4.6 shows the relationship between these parameters.

**Table 4.6. ISP or MJPEG Parameters Enablement in Hardware**

| Configuration Mode | ISP_EN | MJPEG_EN | Description |
|---|---|---|---|
| ISP only | 1 | 0 | Enables ISP processing only. MJPEG compression is disabled. |
| ISP + MJPEG | 1 | 1 | Enables both ISP processing and MJPEG compression. MJPEG depends on ISP and requires it to be enabled. |
| ISP disabled | 0 | X (disabled) | Disables ISP. MJPEG is automatically unavailable and its setting is ignored. |

When the ISP processing is enabled in hardware (ISP_EN = 1), with or without MJPEG (MJPEG_EN = 0 or 1), the software controls ISP and MJPEG operation through control registers. Provide the appropriate argument when running the player command-line tool. Further details on command-line options, refer to the Testing the System section.

### 4.1.8. Configuring the Driver and Applying the Patch

The driver configuration must be adapted for the specific reference design. Because each camera sensor uses distinct I2C control signals, the driver must be updated to correctly detect and communicate with the intended sensor, ensuring proper initialization and operation.

Configuration support for the IMX258 sensor is described in the Building the Holoscan Sensor Bridge Demo Container section. To enable this support, follow the procedure outlined from Configuring the AGX Orin Developer Kit through Building the Holoscan Sensor Bridge Demo Container section.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

## 4.2. Customizing the Reference Design for the CertusPro-NX Versa Board

Figure 4.18 shows an overview of a configured CertusPro-NX Versa board block diagram.

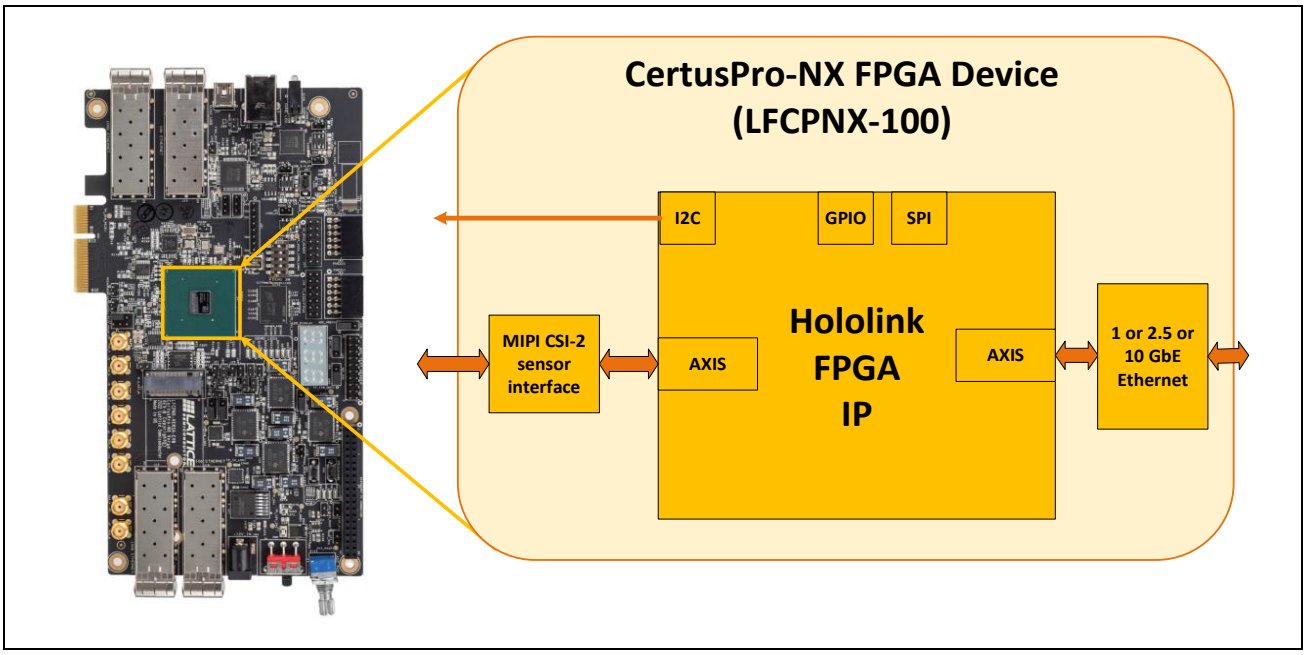



**Figure 4.18. CertusPro-NX Versa Board Block Diagram**

Table 4.7 shows the configuration required for the reference design to run on the CertusPro-NX Versa Board.

**Table 4.7. Configuration for the reference design to fit into the CertusPro-NX Versa Board**

| Sensor / Host Interface | Features | | Configuration |
|---|---|---|---|
| Camera sensor interface | Number of camera sensors | | 1 |
| | Preset MIPI configuration method | | — |
| | 1 | Camera 0: MIPI lane number | 4 |
| | 2 | Camera 0: MIPI lane rate | 720 Mbps |
| Ethernet Link | Number of Ethernet port | | 1 |
| | Ethernet link speed | | 1GbE, 2.5GbE, or 10GbE |

To customize the reference design, follow the steps below.

**Note:** This reference design is already integrated with a preset MIPI RX D-PHY IP configuration, latest Hololink IP (version 2606), and Ethernet IPs.

1. Customize the reference design to match the CertusPro-NX Versa board specifications.
2. Configure HDL Parameter and DEFINE Switch.

   The reference design needs to be configured by changing the settings of the HDL parameter and DEFINE switch to meet the CertusPro-NX Versa board requirement using this file:

   ```
   <project_name>/fpga_lfcpnx/radiant/src/rtl/hsb_cpnx/versa_top/HOLOLINK_def.svh
   ```

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

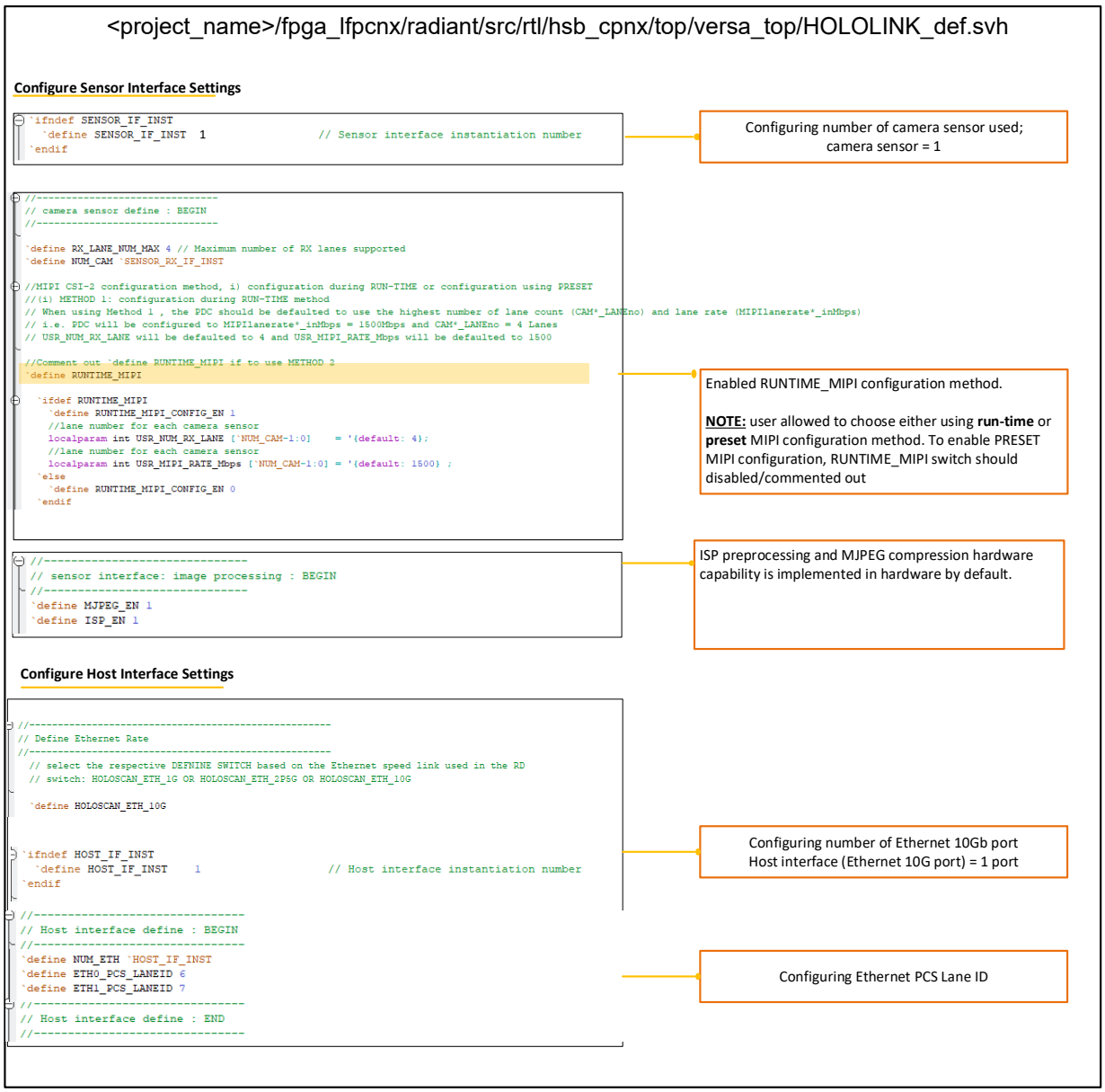
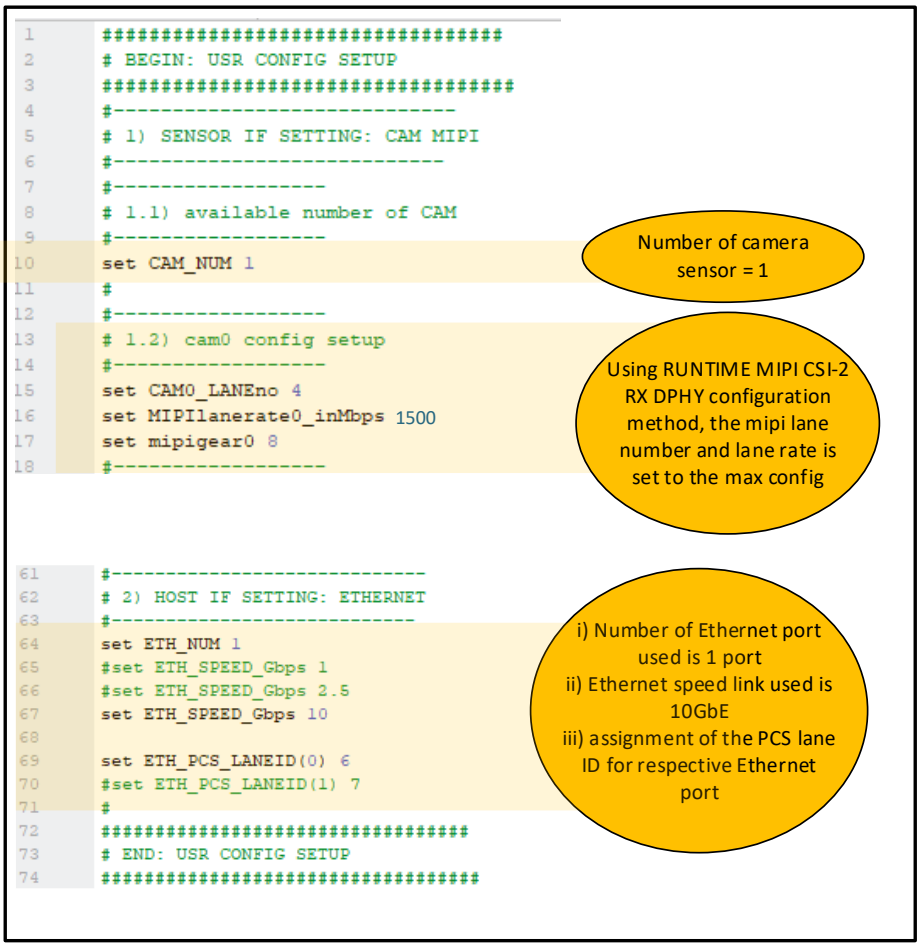



**Figure 4.19. Required HDL Parameter and DEFINE Switch Configuration**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

3. Review the PDC configuration file.

   Figure 4.20 shows a screenshot of the *<project_name>/fpga_lfcpnx/radiant/constraints/hsb_cpnx_versa.pdc* file.




```
1  ####################################
2  # BEGIN: USR CONFIG SETUP
3  ####################################
4  #-----------------------------
5  # 1) SENSOR IF SETTING: CAM MIPI
6  #-----------------------------
7  #------------------
8  # 1.1) available number of CAM
9  #------------------
10 set CAM_NUM 1
11 #
12 #------------------
13 # 1.2) cam0 config setup
14 #------------------
15 set CAM0_LANEno 4
16 set MIPIlanerate0_inMbps 1500
17 set mipigear0 8
18 #------------------


61 #-----------------------------
62 # 2) HOST IF SETTING: ETHERNET
63 #-----------------------------
64 set ETH_NUM 1
65 #set ETH_SPEED_Gbps 1
66 #set ETH_SPEED_Gbps 2.5
67 set ETH_SPEED_Gbps 10
68
69 set ETH_PCS_LANEID(0) 6
70 #set ETH_PCS_LANEID(1) 7
71 #
72 ####################################
73 # END: USR CONFIG SETUP
74 ####################################
```

**Figure 4.20. PDC Configuration File**

4. Configure the driver. For details on how to configure the driver, refer to the Building the Holoscan Sensor Bridge Demo Container section. To enable driver configuration support, follow the steps from Configuring the AGX Orin Developer Kit section through the Building the Holoscan Sensor Bridge Demo Container section.
5. Verify the reference design configuration. Review the Lattice Radiant software hierarchy file list and the clock summary from the place-and-route timing report. Figure 4.21 shows the reference design in the Lattice Radiant software hierarchy file list. The reference design is now ready for compilation and implementation.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

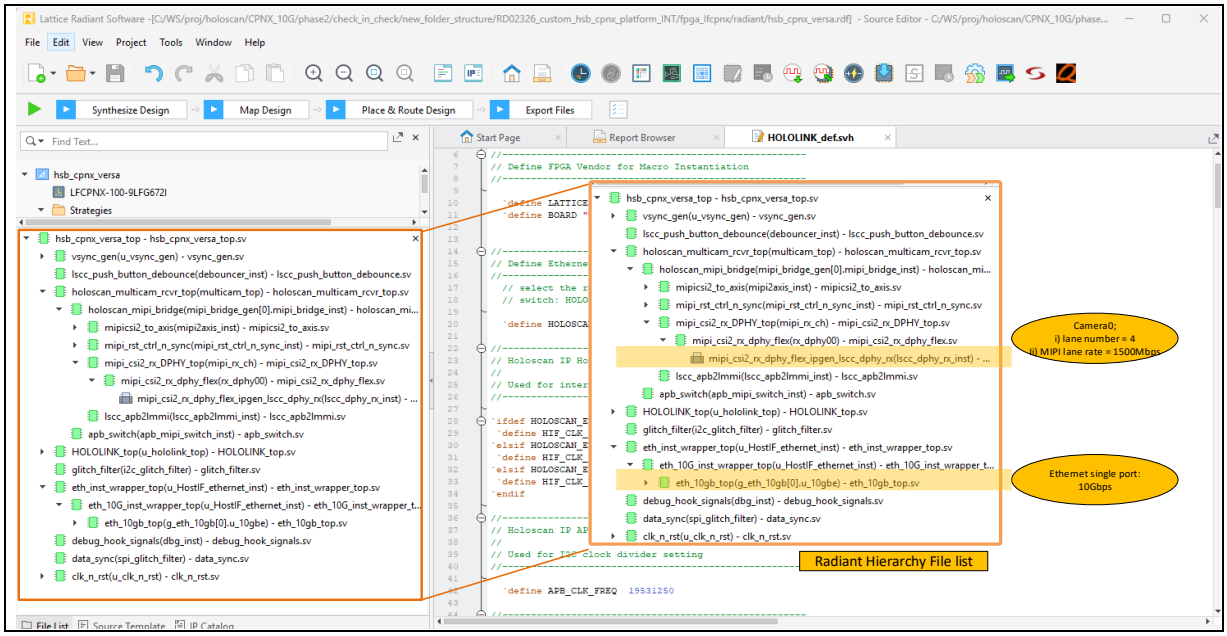



**Figure 4.21. Lattice Radiant Software Hierarchy File List**

## 4.3. Customizing the Reference Design for the CertusPro-NX Sensor-to-Ethernet Bridge Board

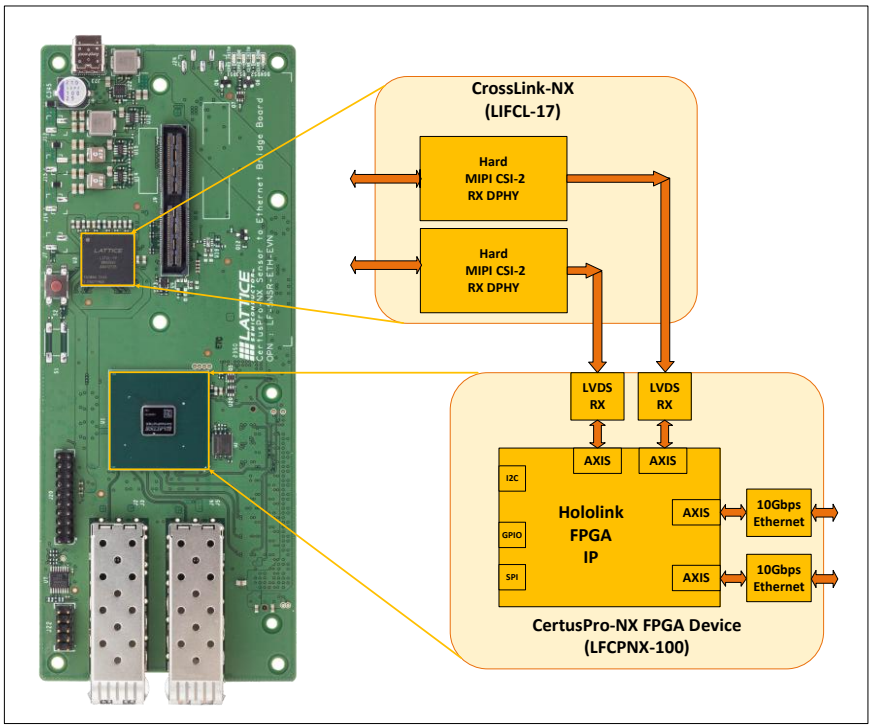



**Figure 4.22. CertusPro-NX Sensor-to-Ethernet Bridge Board Block Diagram**

Table 4.8 shows the configuration required for the reference design to run on the CertusPro-NX Sensor-to-Ethernet Bridge Board. The configuration for this reference design is fixed because the board is designed for a specific use.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

**Table 4.8. Configuration for the reference design to fit into the CertusPro-NX Sensor-to-Ethernet Bridge Board**

| Sensor/Host interface | Features | Configuration |
|---|---|---|
| CrossLink-NX external device | 2 MIPI CSI-2 Hard D-PHY | 1 prebuilt bitstream |
| LVDS DDR receiver interface | Number of LVDS DDR receiver | 2 |
| Ethernet Link | Number of Ethernet port | 2 |
| | Ethernet link speed | 10GbE |

To customize the reference design, follow the steps below.

**Note:** This reference design includes a prebuilt FPGA bitstream for the CrossLink-NX device that supports two MIPI CSI-2 Hard D-PHY interfaces, LVDS DDR RX IP, the latest Hololink IP (version 2606), and Ethernet IPs.

1. Customize the reference design to match the CertusPro-NX Sensor-to-Ethernet Bridge board specifications.
2. Program the prebuilt FPGA bitstream for the CrossLink-NX device on the platform.
3. Configure the HDL Parameter and DEFINE Switch.

   The reference design needs to be configured by changing the HDL parameter and DEFINE switch settings to meet the CertusPro-NX Sensor-to-Ethernet Bridge board requirements using this file:

   `<project_name>/fpga_lfcpnx/radiant/src/rtl/hsb_cpnx/top/2chip_top/HOLOLINK_def.svh`

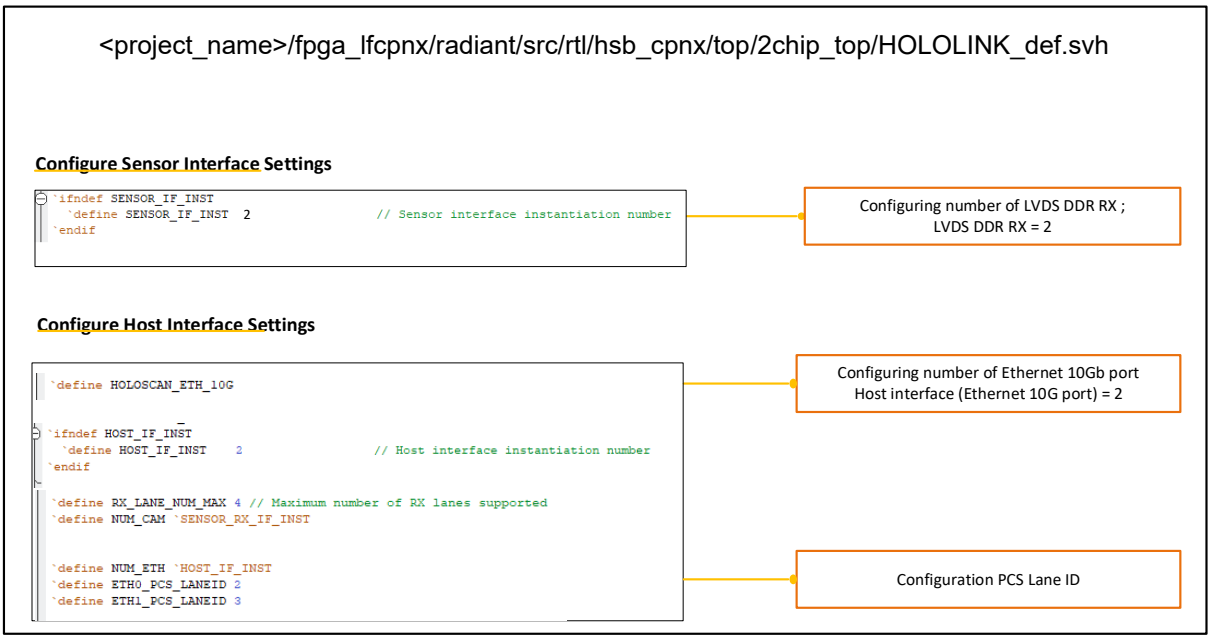



**Figure 4.23. Required HDL Parameter and DEFINE Switch Configuration**

4. Review the PDC configuration file, *<project_name>/fpga_lfcpnx/radiant/constraints/hsb_cpnx_2chip.pdc* and make the changes accordingly.
5. Verify the reference design configuration. Review the Lattice Radiant software hierarchy file list and the clock summary from the place-and-route timing report. Figure 4.24 shows the reference design in the Lattice Radiant software hierarchy file list. The reference design is now ready for compilation and implementation.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

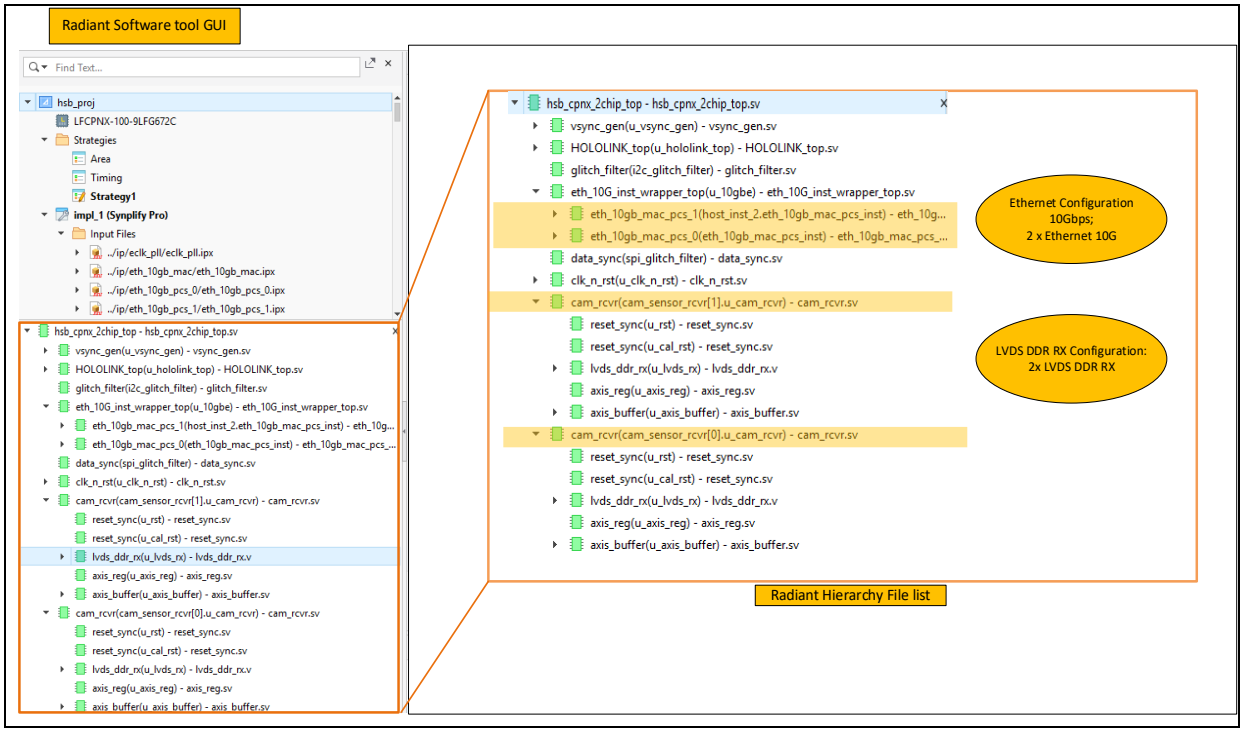



**Figure 4.24. Lattice Radiant Software Hierarchy File List**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# 5. Compiling the Reference Design

This section describes how to compile the reference design using Lattice Radiant software. For more details on the Lattice Radiant software, refer to the Lattice Radiant Software User Guide. The screenshots in this section are for reference only; details may vary depending on the software version and the FW ID, Board ID, and date code used.

## 5.1. Configuring the Reference Design Parameters

To configure the reference design parameters, refer to the Customizing the Reference Design section.

## 5.2. Assigning a Revision ID for Each Bitstream

To assign a revision ID for each bitstream, update the top-level HDL file hsb_cpnx_versa_top.sv for the CertusPro-NX Versa Board as shown in Figure 5.1 or hsb_cpnx_2chip_top.sv for the CertusPro-NX Sensor-to-Ethernet Bridge Board as shown in Figure 5.2.

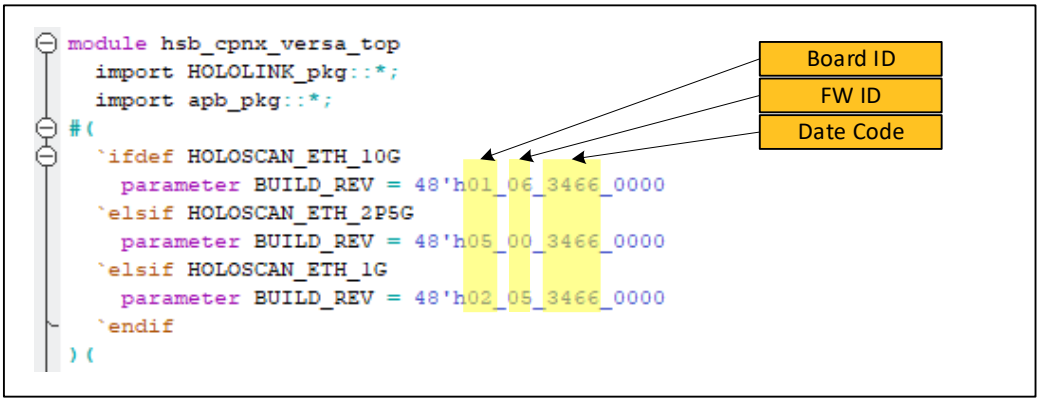



**Figure 5.1. Revision ID assignment (hsb_cpnx_versa_top.sv)**

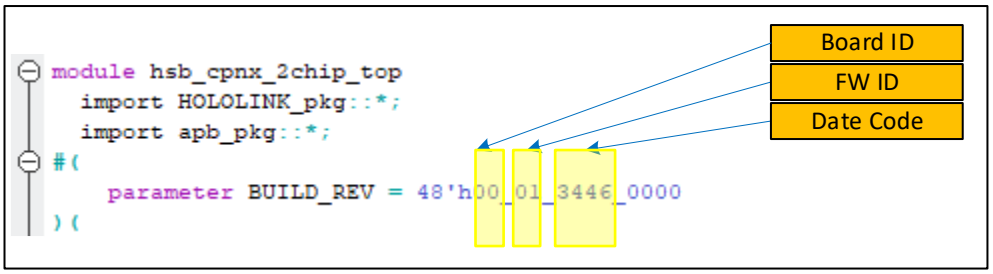



**Figure 5.2. Revision ID assignment (hsb_cpnx_2chip_top.sv)**

The Board ID is stored in a 1-byte hexadecimal value determined by the HSB platform and the configured Ethernet link speed. Table 5.1 lists the Board ID values for each supported HSB platform and link-speed combination. The firmware (FW) ID is also stored in a 1-byte hexadecimal value and increments with each bitstream release.

**Table 5.1. Board ID Representation in Hexadecimal**

| HSD Board with Ethernet Link Speed | Board ID |
|---|---|
| CertusPro-NX Versa board with Ethernet speed 1 Gbps | 0x02 |
| CertusPro-NX Versa board with Ethernet speed 10 Gbps | 0x01 |
| CertusPro-NX Versa board with Ethernet speed 2.5 Gbps | 0x05 |
| CertusPro-NX Sensor-to-Ethernet Bridge Board with Ethernet speed 10 Gbps | 0x00 |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

The date code indicates when the release bitstream was generated. It is stored in a 2-byte hexadecimal format, as shown in Figure 5.3. The figure provides an example of converting the date November 4, 2025 from decimal to binary and hexadecimal formats.

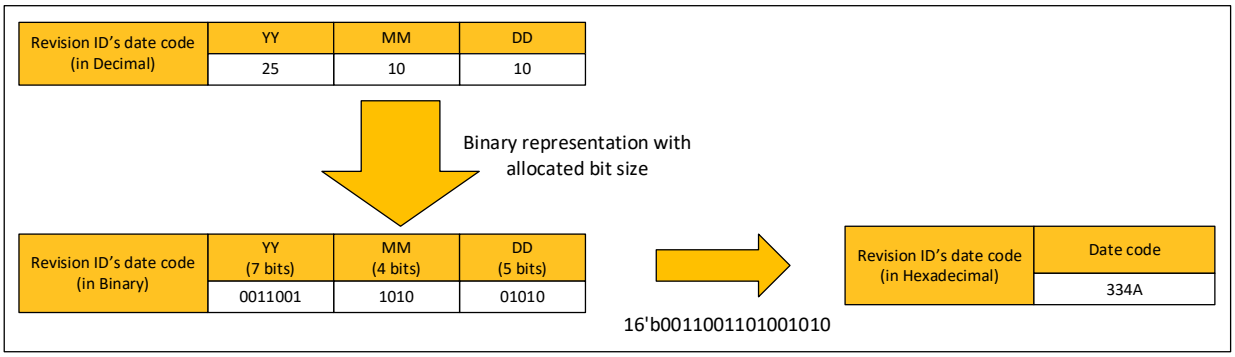



**Figure 5.3. Revision ID Date Code Representation**

## 5.3. Configuring the Lattice Radiant Software to Compile the Reference Design

1. Launch the Lattice Radiant software, as shown in Figure 5.4. The project is compiled using the Lattice Radiant software.

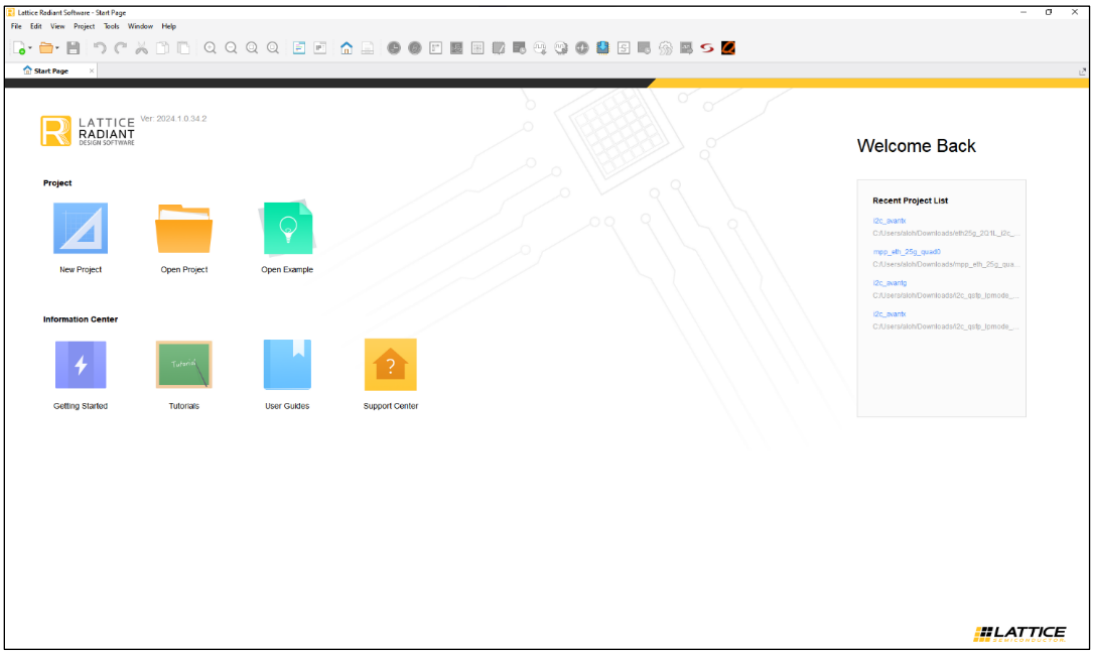



**Figure 5.4. Lattice Radiant Software**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

2. Click **File** > **Open Project.** From the project database, open the Lattice Radiant software project file. To open the CertusPro-NX Versa board project file, use hsb_cpnx_versa.rdf, or to open the CertusPro-NX Sensor-to-Ethernet Bridge board project file, use hsb_cpnx_2chip.rdf, as shown in Figure 5.5.




**Figure 5.5. Open Project File for CertusPro-NX Versa Board**

3. Modify the Place and Route design strategy details under **Strategy1 > Place & Route Design**. Refer to Figure 5.6. The seed run count is set to 4 for this reference design, allowing you to select the reference design with the best place-and-route result and the best STA report. Specify multiple seeds for placement and routing during implementation[1].

**Note:**

1. This is controlled by the Placement Iteration setting under the Place and Route Design strategy in the Lattice Radiant software. This setting determines how many placement and routing iterations are performed. Each iteration uses a different cost table, resulting in a unique placement strategy and generating a distinct Uniform Database (.udb) file. By exploring multiple placement options, you increase the likelihood of achieving better timing closure and overall design performance. Using multiple seeds is beneficial for complex designs as it provides alternative implementations that may meet timing more effectively.

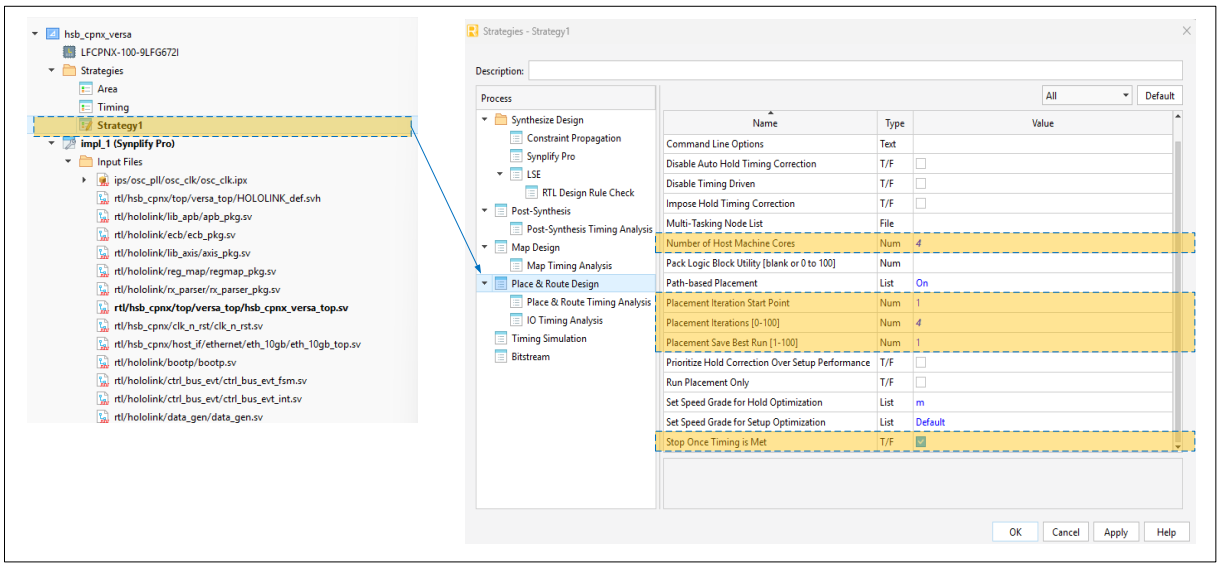



**Figure 5.6. Place and Route Design Strategy**

For further details on configuring the Lattice Radiant software, refer to the Lattice Radiant Software User Guide.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

## 5.4. Generating the Bitstream

To create the FPGA bitstream file, click **Export Files** to generate the bitstream file as shown in Figure 5.7. View the log messages in the Export Reports folder for the generated bitstream.

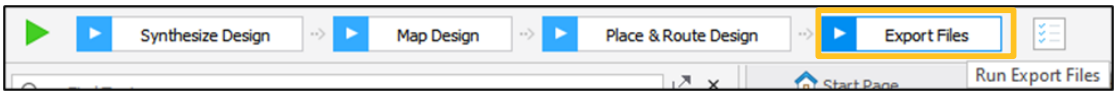



**Figure 5.7. Generate and Export Bitstream File**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# 6. Implementing the Reference Design

The following table provides an overview of the compatibility of various CertusPro-NX board types with different NVIDIA Jetson modules and their respective Ethernet speed capabilities.

**Table 6.1. CertusPro-NX Platform and Jetson Module Compatibility**

| Board Type | Jetson Module | |
|---|---|---|
| | AGX Orin | AGX Thor |
| CertusPro-NX Versa Board | 1GbE, 2.5GbE, and 10GbE | 10GbE |
| CertusPro-NX Sensor-to-Ethernet Bridge Board | 10GbE | - |

## 6.1. Hardware Requirements

### 6.1.1. CertusPro-NX Versa Board and NVIDIA Jetson AGX Orin

- NVIDIA Jetson AGX Orin developer kit
- Lattice CertusPro-NX Versa Board
- IMX258 camera sensor module – contact Lattice Sales for more information
- Ethernet 10GBase-T (Cat 6 RJ-45) SFP+ Module (10GbE support) – tested with 6COM 10GBASE SFP+ RJ-45 Module
- Ethernet 2.5GBase-T (Cat 6 RJ-45) SFP-2.5G-T-1 Module (2.5GbE support) – tested with 6COM 2.5GBASE SFP RJ-45 Module
- 10/100/1000BASE-T SFP module (Ethernet 1GbE support) – tested with FS 10/100/1000BASE-T RJ45 SFP Module
- CAT6 Ethernet cable (support 1GbE/2.5GbE/10GbE)
- M.2 2280 NVMe SSD 512GB
- Type-C USB cable
- DisplayPort cable (for AGX Orin)
- Monitor with DisplayPort support (for AGX Orin)
- Keyboard (for AGX Orin)
- Mouse (for AGX Orin)
- Mini USB Type-A cable for programming
- 12 V power supply
- Jetson AGX Orin power adapter

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Figure 6.1 shows the hardware setup for the Jetson AGX Orin host and Lattice CertusPro-NX Versa Board.

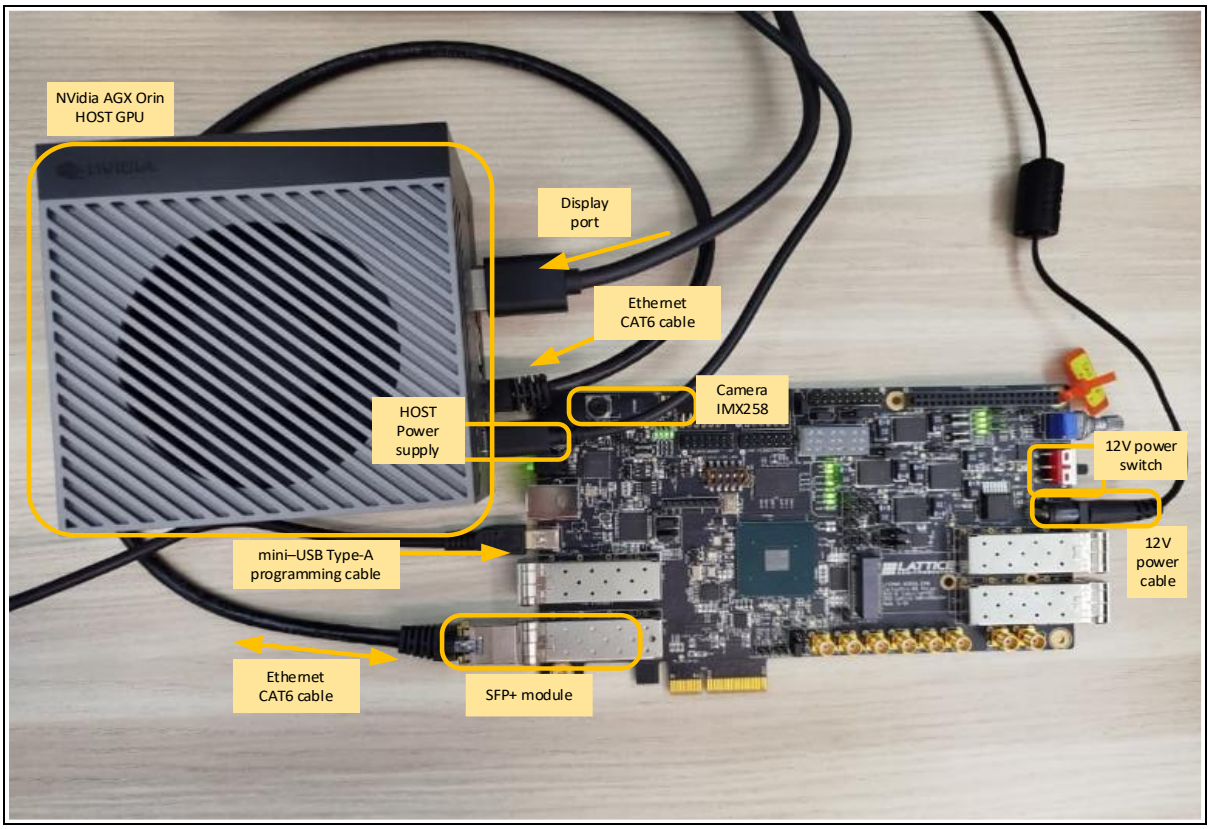



**Figure 6.1. Hardware Setup**

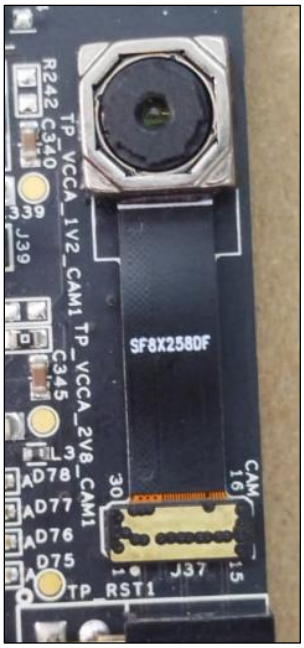



**Figure 6.2. Camera Sensor Connection**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### 6.1.2. CertusPro-NX Versa Board and NVIDIA Jetson AGX Thor

- NVIDIA Jetson AGX Thor developer kit
- Lattice CertusPro-NX Versa Board
- 13 MP IMX258 camera module – contact Lattice Sales for more information
- 100G QSFP28 to 4×25G SFP28 Breakout Active Optical Cable (AOC) – tested with FS 100G QSFP28 to 4×25G Breakout AOC
- Type-C to Type-A USB Cable
- DisplayPort cable (for AGX Thor)
- Monitor with DisplayPort support (for AGX Thor)
- Keyboard (for AGX Thor)
- Mouse (for AGX Thor)
- Mini USB to USB-A cable (for programming)
- 12 V power supply (for CertusPro-NX Versa Board)
- Power adapter (for AGX Thor)

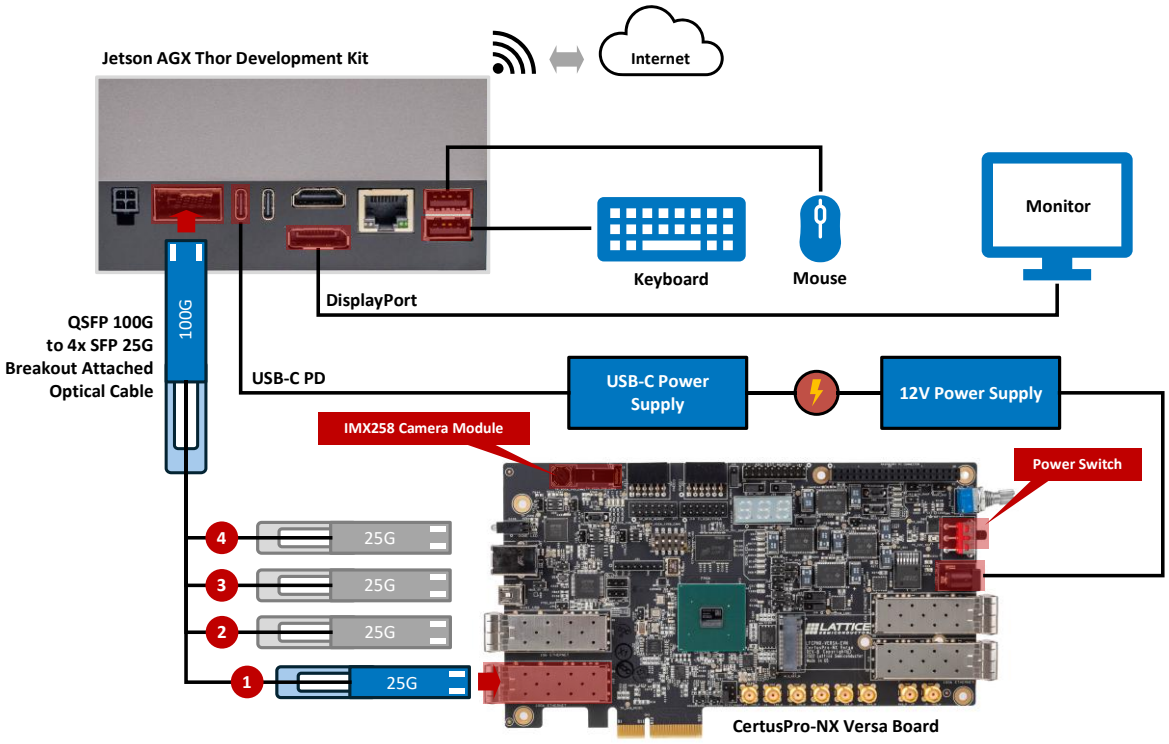



**Figure 6.3. Setup with AGX Thor for 10GbE Ethernet**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### 6.1.3. CertusPro-NX Sensor-to-Ethernet Bridge Board and NVIDIA Jetson AGX Orin

- NVIDIA Jetson AGX Orin developer kit
- CertusPro-NX Sensor-to-Ethernet Bridge Board
- Leopard Imaging IMX274 Dual Camera Sensor Module – contact Lattice Sales for more information
- Ethernet 10GBase-T (Cat 6 RJ-45) SFP+ Module (Ethernet 10GbE support) – tested with 6COM 10GBASE SFP+ RJ-45 Module
- Lattice Programming Cables
- CAT6 Ethernet cable (support 10GbE)
- M.2 2280 NVMe SSD 512GB
- Type-C to Type-A USB Cable
- DisplayPort cable (for AGX Orin)
- Monitor with DisplayPort support (for AGX Orin)
- Keyboard (for AGX Orin)
- Mouse (for AGX Orin)
- Mini USB to USB-A cable (for programming)
- USB-C power adapter (for CertusPro-NX Sensor-to-Ethernet Bridge Board)
- Power adapter (for AGX Orin)

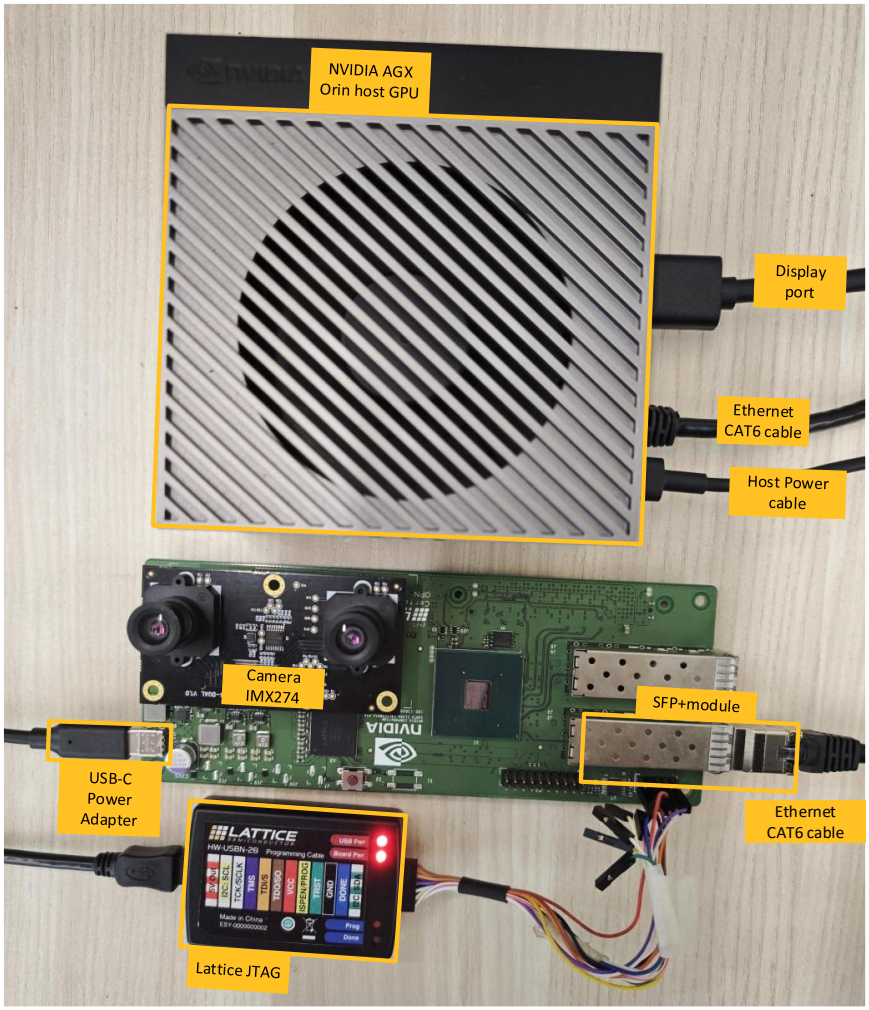



**Figure 6.4. Hardware Setup for using CertusPro-NX Sensor-to-Ethernet Bridge Board with AGX Orin**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

## 6.2. Software Requirements

### 6.2.1. Jetson AGX Orin

- JetPack 7.2
- NVIDIA Jetson Linux 39.2
- NVIDIA Holoscan SDK 4.4.0
- Holoscan Sensor Bridge (HSB) SDK 2.7.0

### 6.2.2. Jetson AGX Thor

- JetPack 7.2
- NVIDIA Jetson Linux 39.2
- NVIDIA Holoscan SDK 4.4.0
- Holoscan Sensor Bridge (HSB) SDK 2.7.0

## 6.3. NVIDIA Platform Host Setup

### 6.3.1. Jetson AGX Orin Developer Kit

#### 6.3.1.1. Setting Up the Jetson AGX Orin Developer Kit

To set up the Jetson AGX Orin Developer Kit, follow the steps below. Execute these steps on an Ubuntu host PC terminal.

1. Set up an Ubuntu 24.04 host PC with at least 8 GB of RAM and an internet connection. Connect the Ubuntu host to the Jetson AGX Orin using a USB cable as shown in Figure 6.5.

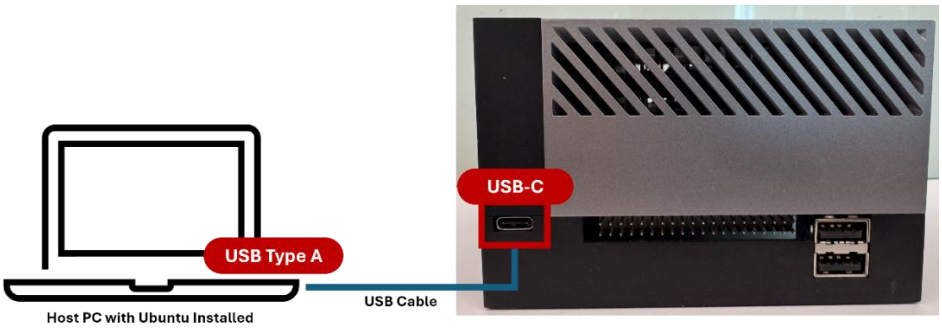



**Figure 6.5. Ubuntu Host PC and Jetson AGX Orin Developer Kit Setup**

2. Download and install the NVIDIA SDK Manager:

```
wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2404/x86_64/cuda-keyring_1.1-1_all.deb
sudo dpkg -i cuda-keyring_1.1-1_all.deb
sudo apt-get update
sudo apt-get -y install sdkmanager
```

3. Place the AGX Orin into recovery mode by pressing and holding the recovery button as shown in Figure 6.6. While holding the recovery button, press the reset button and release it. Wait at least three seconds, then release the recovery button.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.




**Figure 6.6. Recovery and Reset Button**

4. Launch SDK Manager from the terminal, type:
   ```
   sdkmanager
   ```
5. Sign up for an NVIDIA Developer account at https://developer.nvidia.com if you do not have one.
6. After the SDK Manager is launched, click Login as shown in Figure 6.7. A browser window opens, where you sign in using your NVIDIA Developer account credentials.

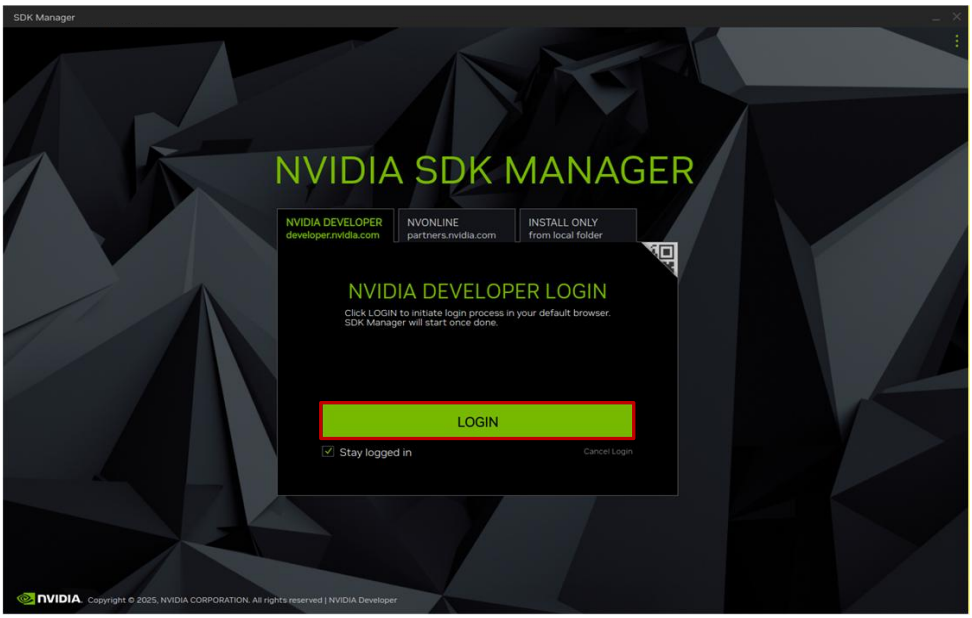



**Figure 6.7. SDK Manager Login**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

7. Check your email for the authentication verification link and click it to complete the login process. Once verified, the SDK Manager launches automatically.
8. After the SDK Manager window appears, select **Jetson AGX Orin [64 GB Developer Kit version]** as the target hardware. Then click **OK**.

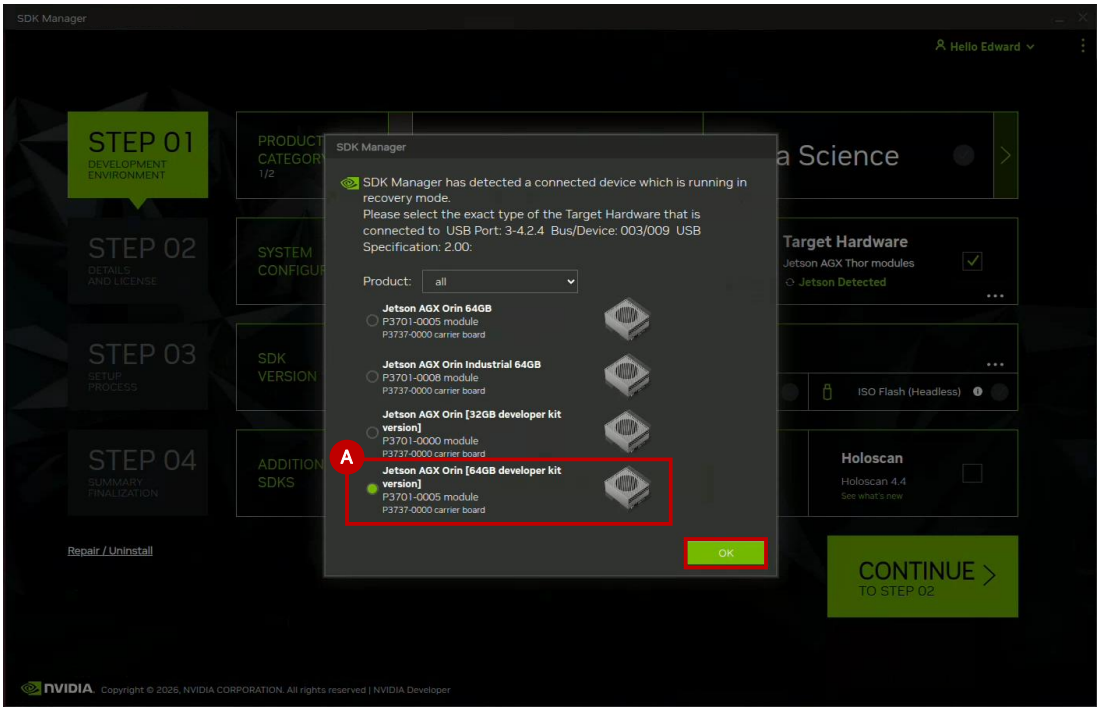



**Figure 6.8. Jetson AGX Orin [64 GB Developer Kit Version]**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

9. If the module is not detected, repeat the Recovery mode step. Click the **…** icon on bottom right of **(A)** to select the correct target hardware as shown in Figure 6.9. Click the **…** icon on bottom right of **(B)** and select **JetPack 7.2**. Select the check box as indicated in **(C)** to select Direct Flash. Then, click **CONTINUE** to proceed.

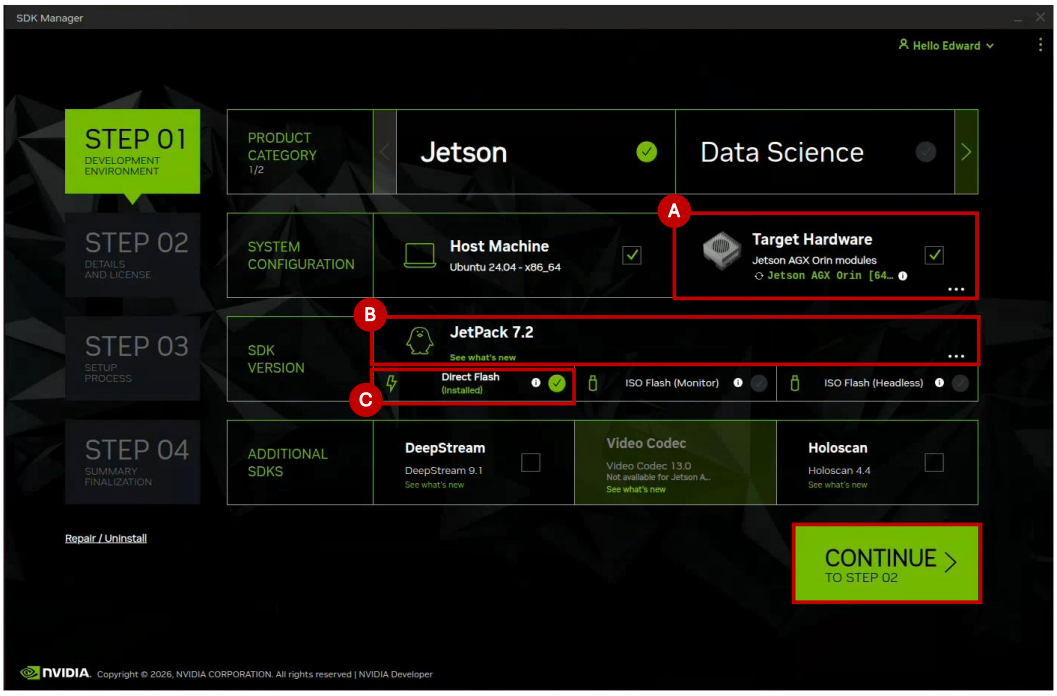



**Figure 6.9. Development Environment Selection**

10. Leave all settings as default, and select **I accept the terms and conditions of the license agreements** at **(A)**. Then, click **CONTINUE** to proceed with the installation process.

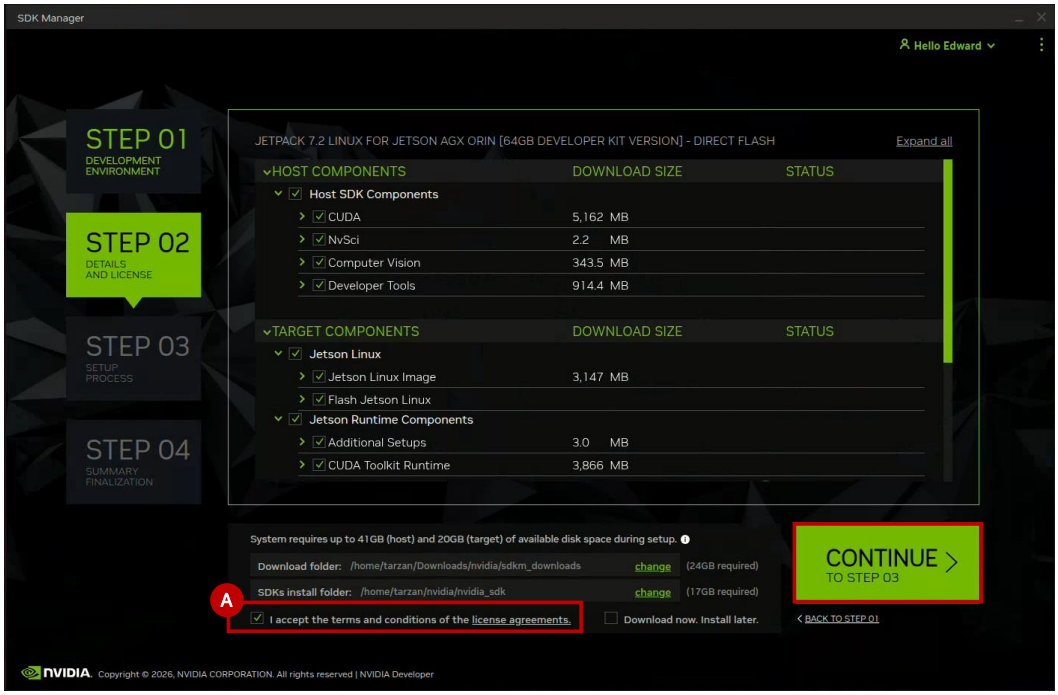



**Figure 6.10. Terms and Conditions – License Agreement**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

11. In **STEP 03 – SETUP PROCESS,** as shown in Figure 6.12, wait for the download operation and OS image creation process to complete.

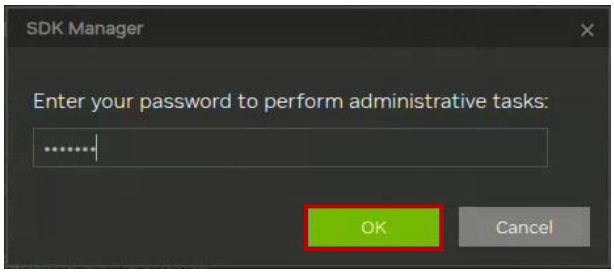



**Figure 6.11. Administrative Right**

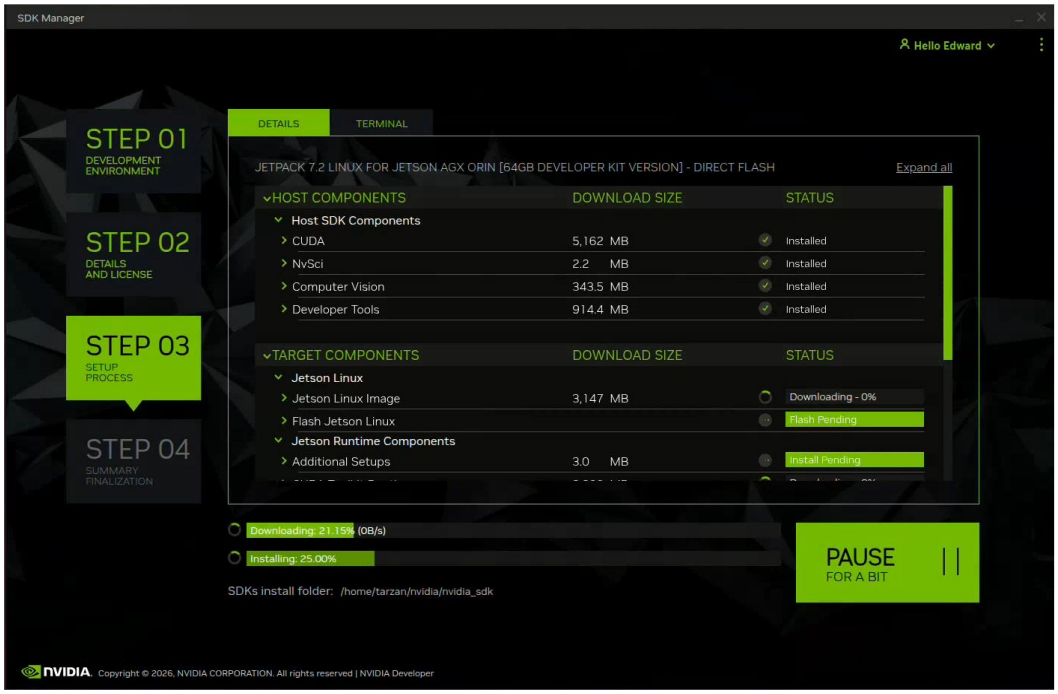



**Figure 6.12. Download Operation and OS Image Creation**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

12. When prompted for a flash setup, select Pre-Config from the dropdown at (A). Enter the desired username to access the AGX Orin system (see letter **(B)** in Figure 6.13); along with the password at **(C)**. Select NVMe as the storage device at **(D)**. Then, click **Flash** to proceed.

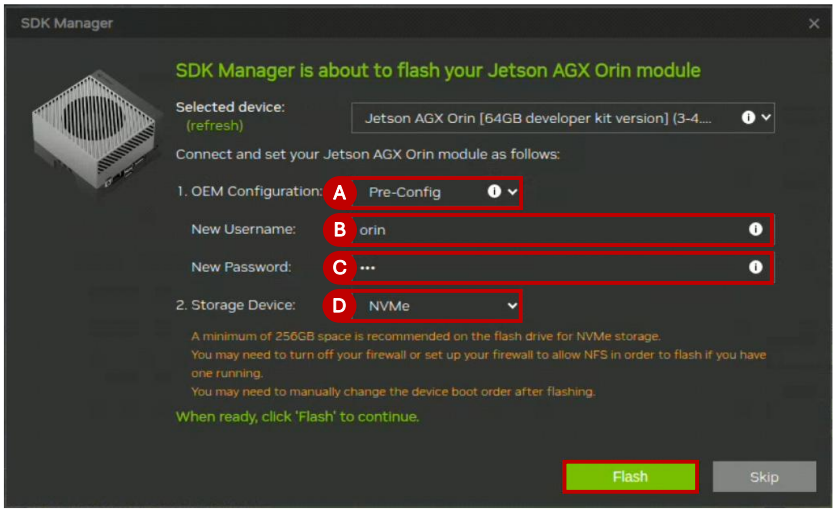



**Figure 6.13. SDK Manager**

13. Complete **STEP 04 – SUMMARY FINALIZATION** to finish the setup. Reset the Jetson AGX Orin system by pressing the reset button as shown in Figure 6.6. Then proceed to the next section to configure the Jetson AGX Orin itself.




**Figure 6.14. Summary Finalization**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

#### 6.3.1.2. Configuring the AGX Orin Developer Kit

The customizable HSB sensor interface reference design is supported by the Jetson AGX Orin system running JetPack 7.2. The following steps are one-time setup procedures and should be executed directly on the AGX Orin terminal, outside the Holoscan Sensor Bridge (HSB) Docker container. All commands assume the HSB is connected through end0 (the Ethernet interface on the Jetson AGX Orin).

1. Install git-lfs (some data files in the HSB source repository use it).
```
sudo apt-get update
sudo apt-get install -y git-lfs nano
```
2. Grant the user permission to the docker subsystem.
```
sudo usermod -aG docker $USER
```
3. Reboot the AGX Orin to apply the changes.
```
sudo reboot
```
4. Increase the network receive buffer size to support Linux sockets.
```
echo 'net.core.rmem_max = 31326208' | sudo tee /etc/sysctl.d/52-hololink-rmem_max.conf
sudo sysctl -p /etc/sysctl.d/52-hololink-rmem_max.conf
```
5. Assign a static IP address (192.168.0.101) to the onboard network port.
```
EN0=end0
sudo nmcli con add con-name hololink-$EN0 ifname $EN0 type ethernet ip4 192.168.0.101/24
sudo nmcli connection up hololink-$EN0
```
6. Power-on the CertusPro-NX Versa board and ensure it is properly connected. Then, ping to check connectivity.
```
ping 192.168.0.2
```
7. Isolate a processor core from the Linux kernel when running Linux socket-based examples[1].

   To isolate that core, edit the /boot/extlinux/extlinux.conf file.
```
sudo apt install nano
sudo apt update
sudo nano /boot/extlinux/extlinux.conf
```
8. Add the setting isolcpus=2 to the end of the line that begins with **APPEND.**

   Your file should resemble the following:
```
TIMEOUT 30
DEFAULT primary

MENU TITLE L4T boot options

LABEL primary
      MENU LABEL primary kernel
      LINUX /boot/Image
      ...
      APPEND ${cbootargs} ...<other-settings>... isolcpus=2
```
   Save and exit by pressing **Ctrl + O**, then **Enter**, followed by **Ctrl + X.**

   ***Note:*** *The sensor bridge applications can run the network receiver process on a different core by setting the environment variable* ***HOLOLINK_AFFINITY*** *to the desired core*.

   *The command below is shown for demonstrating the use of HOLOLINK_AFFINITY and is not required for system setup.*
```
HOLOLINK_AFFINITY=0 python3 examples/linux_imx258_player.py
```

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

9. Run the **jetson_clocks** tool at startup to set the core clocks to their maximum.

```
JETSON_CLOCKS_SERVICE=/etc/systemd/system/jetson_clocks.service
cat <<EOF | sudo tee $JETSON_CLOCKS_SERVICE >/dev/null
[Unit]
Description=Jetson Clocks Startup
After=nvpmodel.service

[Service]
Type=oneshot
ExecStart=/usr/bin/jetson_clocks

[Install]
WantedBy=multi-user.target
EOF
sudo chmod u+x $JETSON_CLOCKS_SERVICE
sudo systemctl enable jetson_clocks.service
```

10. Set the Jetson AGX Orin power mode to **MAXN** for optimal performance. You can change this setting using the power dropdown menu in the upper-right corner of the screen, as indicated by **(A)**. Click **(B)** to select the **MAXN** option.

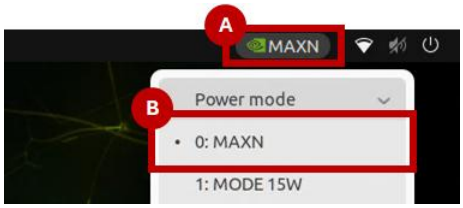



**Figure 6.15. MAXN Power Mode**

11. Reboot Jetson AGX Orin to apply changes.

```
sudo reboot
```

12. Enable PTP for **$EN0** to synchronize timestamps in received data with the host system time. Install the **linuxptp** tool, then create a systemd service file to run **phc2sys** at boot time, ensuring that **$EN0** clock synchronizes with the system clock.

```
sudo apt update && sudo apt install -y linuxptp
EN0=end0
PHC2SYS_SERVICE=/etc/systemd/system/phc2sys-$EN0.service
cat <<EOF | sudo tee $PHC2SYS_SERVICE >/dev/null
[Unit]
Description=Copy system time to $EN0
Requires=NetworkManager.service
After=NetworkManager.service
After=timemaster.service

[Service]
Type=simple
ExecStartPre=timeout 3m bash -c "\
until [ \"\$(nmcli -g GENERAL.STATE device show $EN0)\" = \"100 (connected)\" ]; do \
     sleep 1; \
done"
ExecStart=/usr/sbin/phc2sys -c $EN0 -s CLOCK_REALTIME -O 0 -S 0.0001

[Install]
WantedBy=multi-user.target
EOF
```

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

13. Configure service for execution at startup, and start it now.

```
sudo chmod u+x $PHC2SYS_SERVICE
sudo systemctl enable phc2sys-$EN0.service
sudo systemctl start phc2sys-$EN0.service
```

14. Next, run **ptp4l** to send PTP SYNC messages to **$EN0**.

```
cat <<EOF | sudo tee /etc/linuxptp/hsb-ptp.conf >/dev/null
# This configuration is appropriate for NVIDIA Holoscan sensor bridge
# applications, where PTP messages are sent over L2 and a 1/2 second interval.
[global]
logSyncInterval -1
logMinDelayReqInterval -1
network_transport L2
EOF
```

15. Create a systemd service to run **ptp4l**.

```
PTP4L_SERVICE=/etc/systemd/system/ptp4l-$EN0.service
cat <<EOF | sudo tee $PTP4L_SERVICE >/dev/null
[Unit]
Description=Send PTP SYNC messages to $EN0
After=phc2sys-$EN0.service

[Service]
Type=simple
ExecStart=/usr/sbin/ptp4l -i $EN0 -f /etc/linuxptp/hsb-ptp.conf

[Install]
WantedBy=multi-user.target
EOF
```

```
sudo chmod u+x $PTP4L_SERVICE
sudo systemctl enable ptp4l-$EN0.service
sudo systemctl start ptp4l-$EN0.service
```

**Note:**

1. For high-bandwidth applications, such as 4K video acquisition, isolate the network receiver core to ensure optimal performance. Set processor affinity for the example program to the isolated core to improve performance and reduce latency. By default, the sensor bridge software runs the time-critical background network receiver process on CPU 2 (the third core). Ensure CPU 2 is isolated from Linux scheduling so that no other processes run on it without explicit user assignment, thereby improving reliability and performance.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

#### 6.3.1.3. Building the Holoscan Sensor Bridge Demo Container

The MIPI IMX258 camera reference design is based on the HSB version 2.7.0.

1. Fetch the HSB source code version 2.7.0 from Github:
   ```
   git clone https://github.com/nvidia-holoscan/holoscan-sensor-bridge -b 2.7.0
   ```
2. Navigate to the directory at the same level where the `holoscan-sensor-bridge` repository is cloned, copy the patch file **lattice_sensor_bridge_2.7.0.patch** into this directory, and apply the patch to support the IMX258 camera module.
   ```
   patch -p0 < lattice_sensor_bridge_2.7.0.patch
   ```
3. Verify that **linux_imx258_player.py** exists after applying **lattice_sensor_bridge_2.7.0.patch** to confirm that the patch was applied correctly.
   ```
   ls holoscan-sensor-bridge/examples/linux_imx258_player.py
   ```
   Expected output is as follows:
   ```
   holoscan-sensor-bridge/examples/linux_imx258_player.py
   ```
4. Log in to NVIDIA GPU Cloud (NGC) using the browser on AGX Orin.
   - Register for a developer account for NGC at https://catalog.ngc.nvidia.com/
   - Create an API key at: https://ngc.nvidia.com/setup/api-key
   - Use the API key to log in to **nvcr.io.** Run the command *docker login nvcr.io*
     ```
     docker login nvcr.io
     Username: $oauthtoken
     Password: <Your token key to NGC>
     WARNING! Your password will be stored unencrypted in /home/<user>/.docker/config.json.
     Configure a credential helper to remove this warning. See
     https://docs.docker.com/engine/reference/commandline/login/#credentials-store

     Login Succeeded
     ```
5. Build the HSB demo container for iGPU (default for AGX Orin).
   ```
   cd holoscan-sensor-bridge
   sh docker/build.sh --igpu
   ```
6. To validate the setup, run the video streaming test on AGX Orin as described in Testing Camera Video Streaming.

### 6.3.2. Jetson AGX Thor Developer Kit

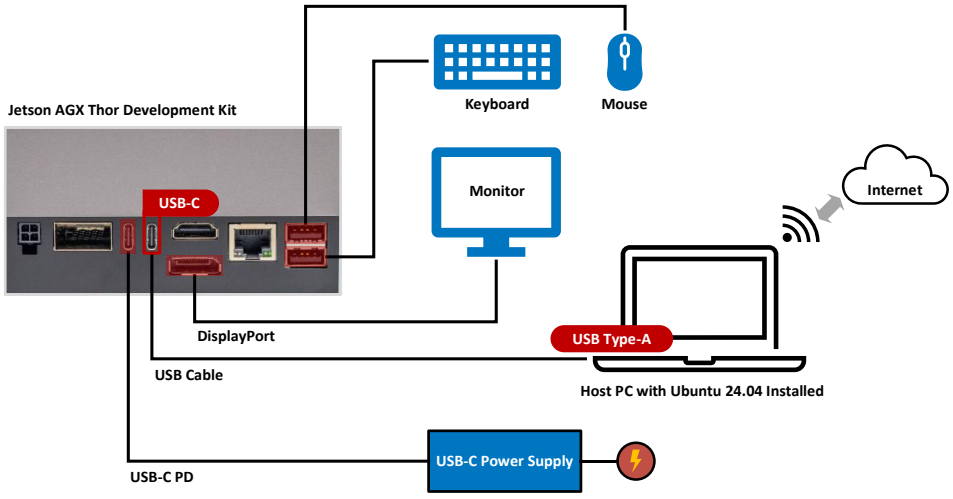



**Figure 6.16. Hardware Setup for Flash Jetson AGX Thor**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

#### 6.3.2.1. Flashing the Jetson AGX Thor using the SDK Manager

1. Ensure the hardware setup matches Figure 6.16. Also ensure the Ubuntu host PC is powered up and internet connection is enabled.
2. Set the AGX Thor to recovery mode by following the below steps.

   **Recovery mode steps**

   a. Connect the Type-C power cable to the AGX Thor. The power supply must be ON.
   b. Press and hold the **Recovery Button** (see Figure 6.17 for location).
   c. While holding the **Recovery Button**, press the Reset Button.
   d. Release both buttons simultaneously.

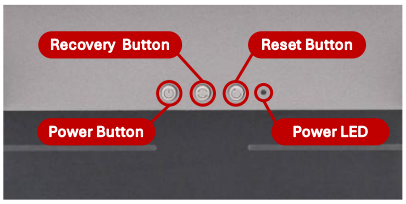



**Figure 6.17. AGX Thor Recovery, Reset, and Power Button**

**Note:** Run all commands in this section from the terminal of the Ubuntu host PC.

3. In the Ubuntu Host PC, open a terminal window and execute the following commands to download the SDK Manager.

```
wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2404/x86_64/cuda-keyring_1.1-1_all.deb
sudo dpkg -i cuda-keyring_1.1-1_all.deb
sudo apt-get update
sudo apt-get -y install sdkmanager
```

4. Run the SDK Manager command.

```
sdkmanager
```

5. Sign up for an NVIDIA Developer account at NVIDIA Developer if you do not have one.
6. After the SDK Manager is launched, click Login as shown in Figure 6.18. A browser window will open, where you sign in using your NVIDIA Developer account credentials.
7. Check your email for the authentication verification link and click it to complete the login process. Once verified, the SDK Manager will launch automatically.

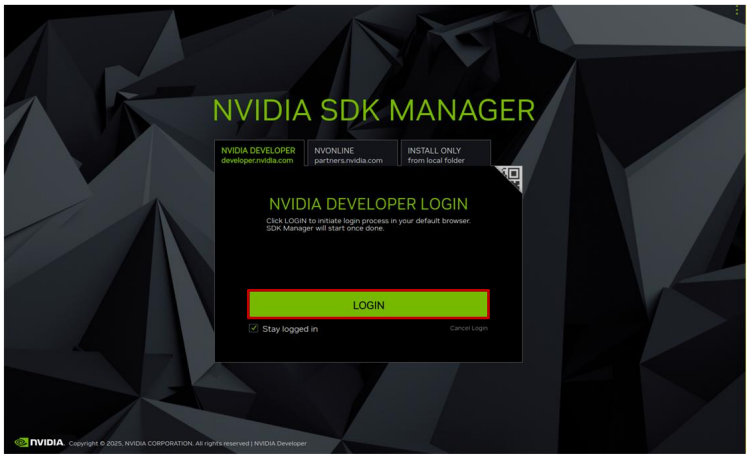



**Figure 6.18. SDK Manager Login**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

8. Select **Jetson AGX Thor Development** as the **Target Hardware** (see letter **A** in Figure 6.20) on the device selection pop up after the SDK Manager starts. Then click OK.

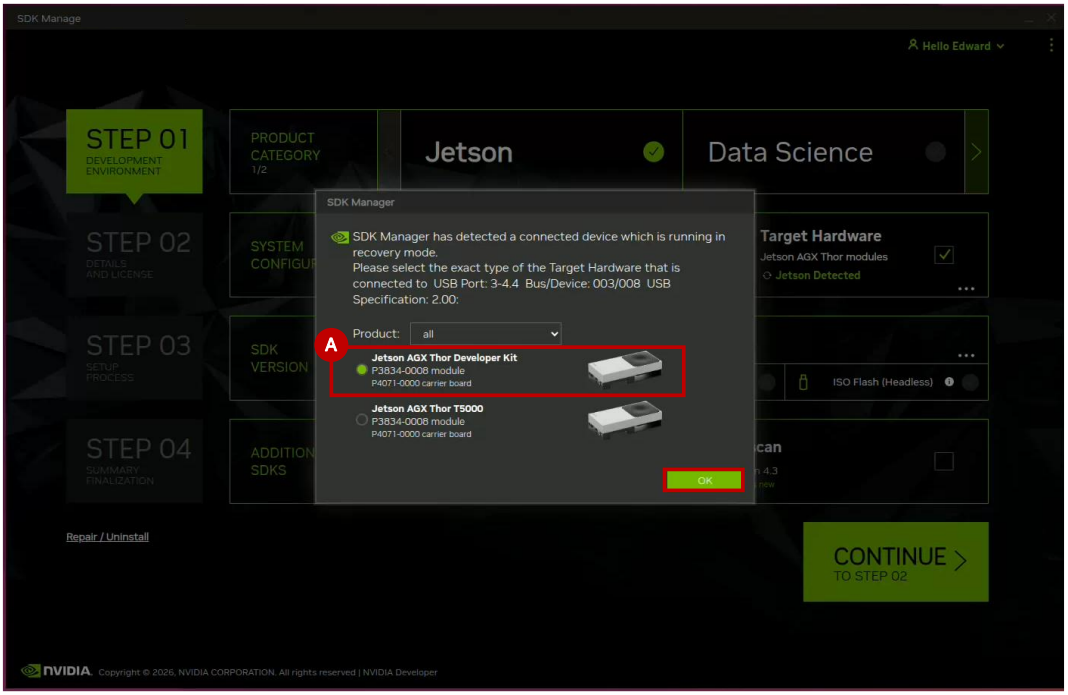



**Figure 6.19. Device Selection**

9. If the module is not detected, repeat the Recovery mode step. Click the … icon on bottom right of **(A)** to select the correct target hardware as shown in Figure 6.20. Click the … icon on bottom right of **(B)** and select JetPack 7.2. Select the check box as indicated in **(C)** to select Direct Flash. Click **CONTINUE** to proceed to **STEP 02** – **Details and License**.

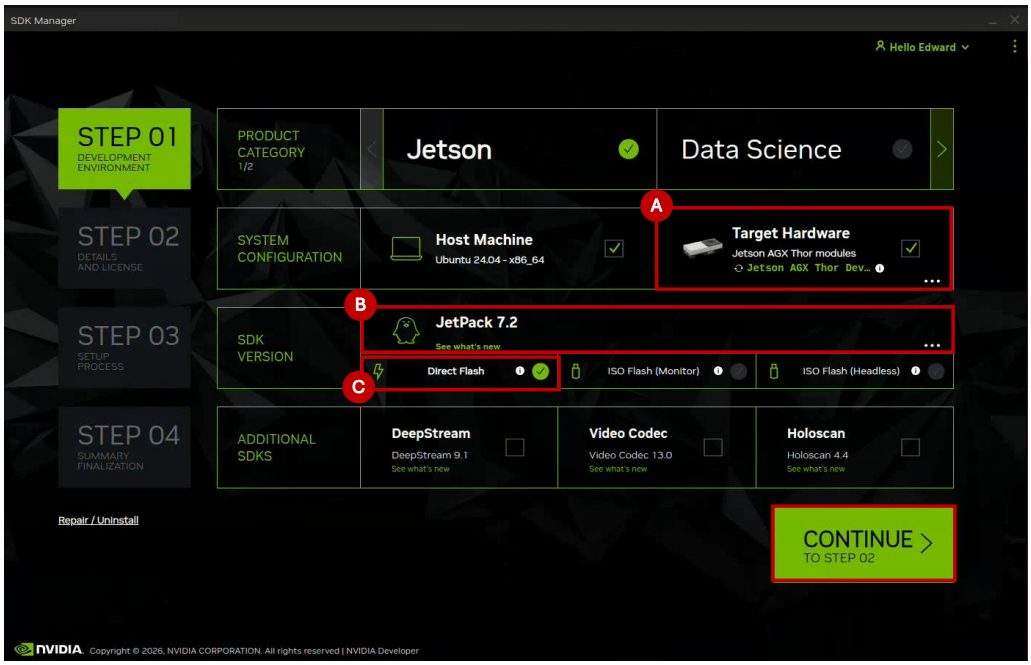



**Figure 6.20. STEP 01 – Development Environment**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

10. In **STEP 02**, select **I accept the terms and conditions of the license agreements** at **(A)** as shown in Figure 6.21. Leave all settings under **TARGET COMPONENTS** as their default values. Click **CONTINUE** to proceed with the installation process.

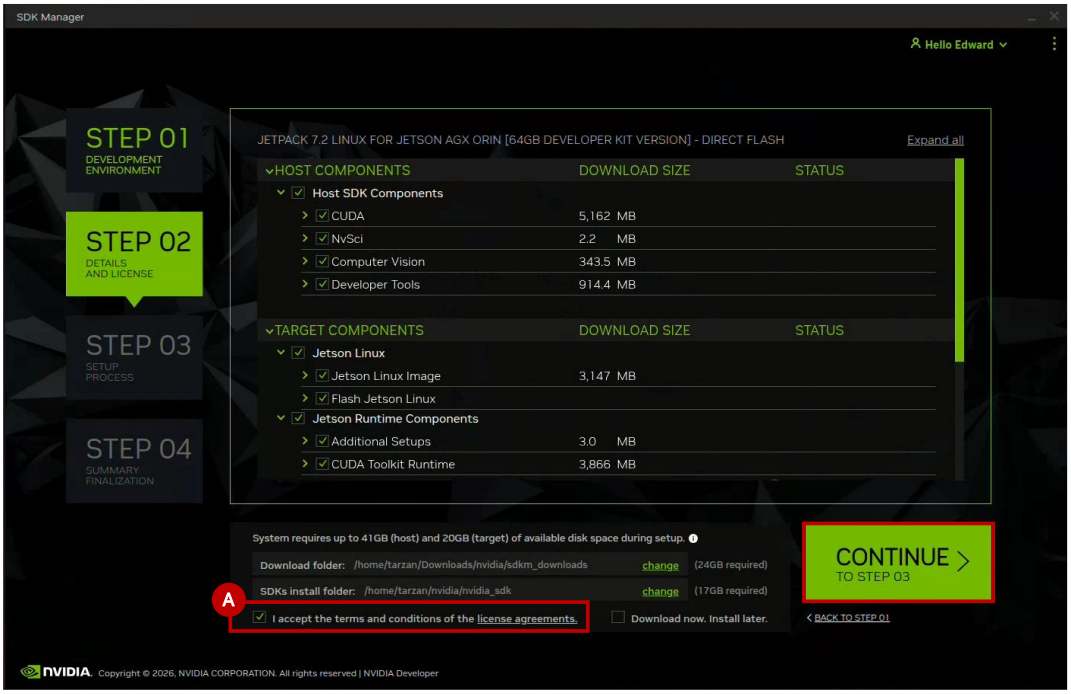



**Figure 6.21. STEP 02 – Details and License**

11. In STEP 03, the **SDK Manager** prompts for a sudo password. Enter the password and click **OK** in Figure 6.22. The software download and installation process begins. The progress is shown in Figure 6.23.

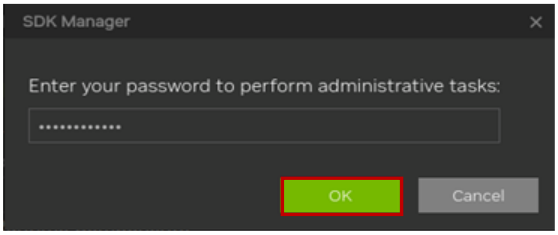



**Figure 6.22. Administrative Right Request**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

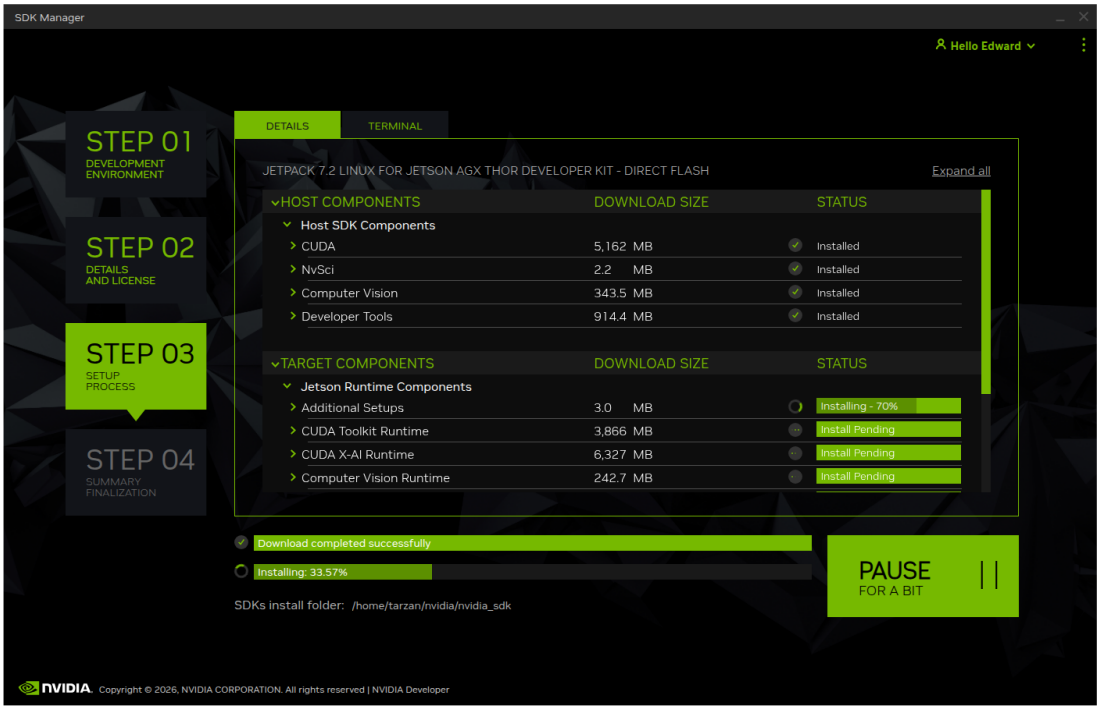



**Figure 6.23. STEP 03 – Setup Process**

12. After the installation completes, the SDK Manager displays a dialog box to flash the target device. Select **Manual Setup – Jetson AGX Thor Developer Kit** (see letter **A**) as shown in Figure 6.24 to begin flashing the target device. Refer to the Recovery mode step for details.

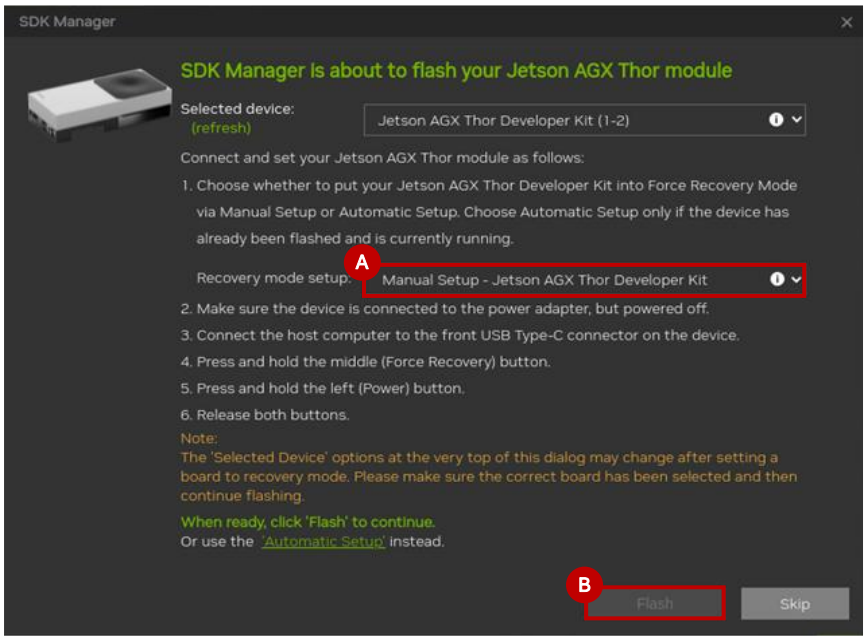



**Figure 6.24. Recovery Mode Selection**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

13. In the setup configuration screen, select **Pre-Config** under the **OEM Configuration** (letter **A**) as shown in Figure 6.25. Enter the desired **username** (letter B) and **password** (letter **C**) to create the new user account for AGX Thor. Leave the **Storage Device** setting as the default. Click **Flash** to begin the flashing process.

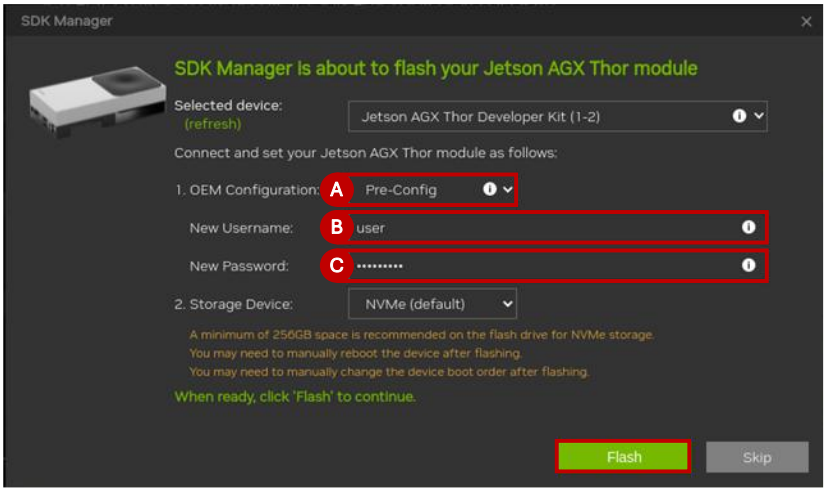



**Figure 6.25. Pre-flash Operation Setup**

14. After flashing is complete, perform a full reboot; power the AGX Thor off, then on (see Figure 6.17 for the power button location).
15. Allow the SDK Manager to detect the device and begin the initial setup process. When the AGX Thor reaches the operating system login screen, enter the created **username** (letter **A**) and **password** (letter **B**) as shown in Figure 6.26. Click **Install** to start installing the SDK component.

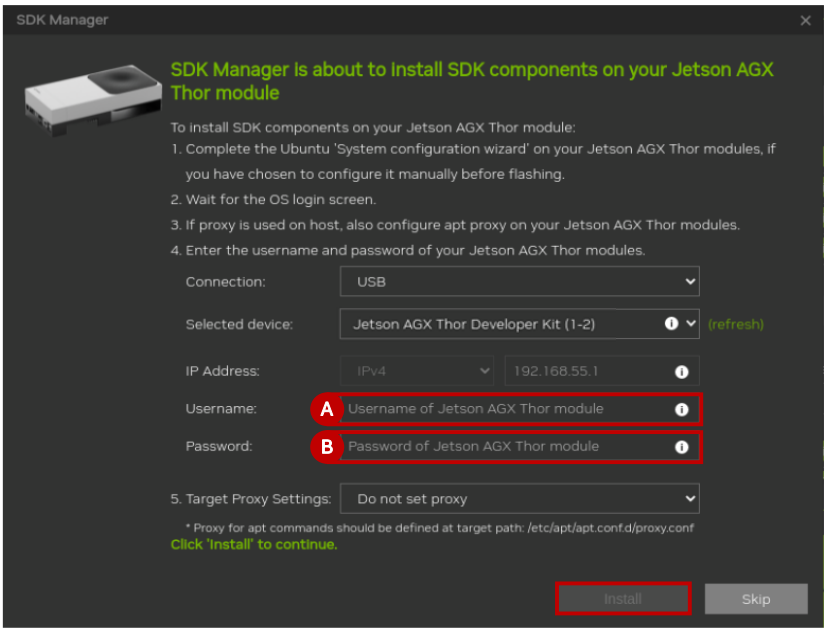



**Figure 6.26. Post-flash Installation Dialog**

16. After the SDK component installation completes, the **STEP 04 – Summary Finalization** panel appears. Review any warnings or errors displayed in the summary panel. Click **FINISH AND EXIT** to complete the installation process.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

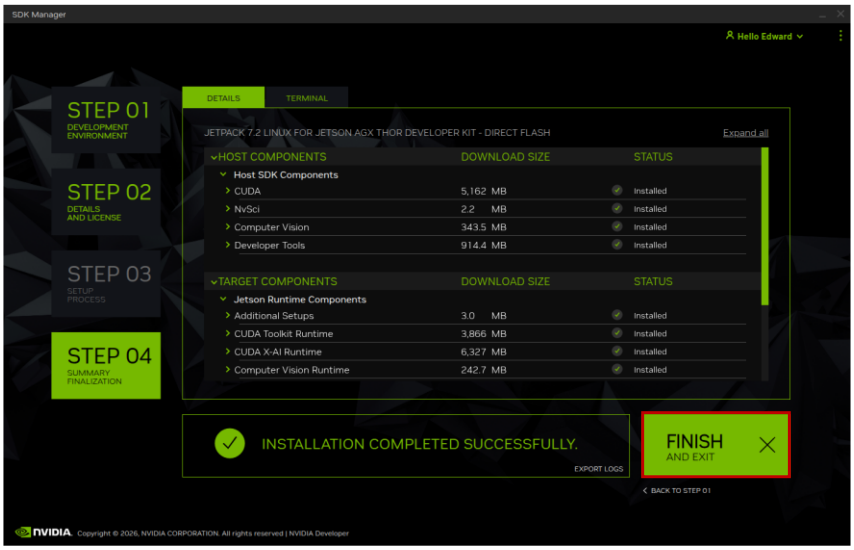



**Figure 6.27. STEP 04 – Summary Finalization**

### 6.3.2.2. Configuring the AGX Thor Developer Kit

**Note:** All commands in this section must be executed in the AGX Thor terminal and only need to be performed once.

1. Install the Holoscan SDK.

```
sudo apt update
sudo apt install holoscan-cuda-13=4.4*
```

2. Install the Holoscan Sensor Bridge dependencies.

```
. /etc/os-release

UBUNTU_VERSION="`echo ${VERSION_ID} | sed 's/\.//g'`"

sudo apt-get install ca-certificates gpg wget

wget https://developer.download.nvidia.com/compute/nvcomp/5.2.0/local_installers/nvcomp-local-repo-\
ubuntu${UBUNTU_VERSION}-5.2.0_5.2.0-1_arm64.deb

sudo dpkg -i nvcomp-local-repo-ubuntu${UBUNTU_VERSION}-5.2.0_5.2.0-1_arm64.deb

sudo cp /var/nvcomp-local-repo-ubuntu${UBUNTU_VERSION}-5.2.0/nvcomp-*-keyring.gpg /usr/share/keyrings/

rm nvcomp-local-repo-ubuntu${UBUNTU_VERSION}-5.2.0_5.2.0-1_arm64.deb

test -f /usr/share/doc/kitware-archive-keyring/copyright || wget -O - \
https://apt.kitware.com/keys/kitware-archive-latest.asc 2>/dev/null | gpg --dearmor - | sudo tee \
/usr/share/keyrings/kitware-archive-keyring.gpg >/dev/null

echo "deb [signed-by=/usr/share/keyrings/kitware-archive-keyring.gpg] https://apt.kitware.com/ubuntu/ \
${VERSION_CODENAME} main" | sudo tee /etc/apt/sources.list.d/kitware.list >/dev/null

sudo apt-get update

test -f /usr/share/doc/kitware-archive-keyring/copyright || sudo rm /usr/share/keyrings/kitware-archive\
-keyring.gpg

sudo apt-get install kitware-archive-keyring
sudo apt-get update

sudo apt install -y cmake=3.31.11* cmake-data=3.31.11* libfmt-dev libssl-dev libcurlpp-dev \
libyaml-cpp-dev python3-dev nvcomp-cuda-13
```

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

3. Add the user to the Docker subsystem.
```
sudo usermod -aG docker $USER
```
4. Increase the Linux network receive buffer size.
```
echo 'net.core.rmem_max = 31326208' | sudo tee /etc/sysctl.d/52-hololink-rmem_max.conf
sudo sysctl -p /etc/sysctl.d/52-hololink-rmem_max.conf
```
5. Isolate a processor core from the Linux kernel when running Linux socket-based examples:
Open **/boot/extlinux/extlinux.conf** for editing.
```
sudo apt update
sudo apt install nano
sudo nano /boot/extlinux/extlinux.conf
```
6. Add the setting **`isolcpus=2`** to the end of the line that begins with **APPEND**.
Your file should resemble the following:
```
TIMEOUT 30
DEFAULT primary

MENU TITLE L4T boot options

LABEL primary
    MENU LABEL primary kernel
    LINUX /boot/Image
    ...
    APPEND ${cbootargs} ...<other-settings>... isolcpus=2
```
Save and exit by pressing **Ctrl + O**, then Enter, followed by **Ctrl + X**.

**Note:** The sensor bridge applications can run the network receiver process on a different core by setting the environment variable **HOLOLINK_AFFINITY** to the desired core. (Complete Steps 1–6 and the procedure in Building the Holoscan Sensor Bridge Demo Container before continuing with Steps 7–13).

*The command below is shown for demonstrating the use of HOLOLINK_AFFINITY and is not required for system setup.*
```
HOLOLINK_AFFINITY=0 python3 examples/linux_imx258_player.py
```
7. Enable the network interface (assuming a camera IP address 192.168.0.2).
```
EN0=mgbe0_0
sudo nmcli con add con-name hololink-$EN0 ifname $EN0 type ethernet ip4 192.168.0.101/24
sudo nmcli connection modify hololink-$EN0 +ipv4.routes 192.168.0.2/32
sudo nmcli connection up hololink-$EN0
```
8. Set the AGX Thor power mode to **MAXN** for optimal performance. You can change this setting using the power dropdown menu in the upper-right corner of the screen, as indicated by **(A)**. Click on Power Mode dropdown and select **(B) MAXN**

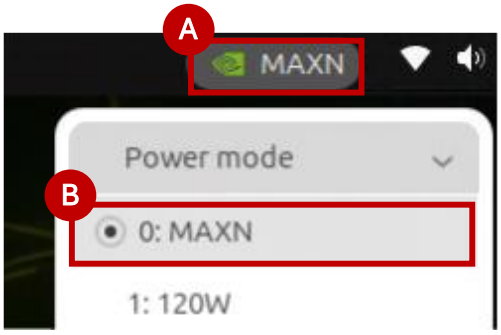



**Figure 6.28. MAXN Power Mode**

9. Reboot the AGX Thor to apply changes.
```
sudo reboot
```

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

10. Enable PTP for **$EN0** to synchronize timestamps in received data with the host system time. Install the **linuxptp** tool, then create a systemd service to run **phc2sys** at boot, ensuring the clock in **$EN0** is synchronized with the system clock.

```
sudo apt update && sudo apt install -y linuxptp
EN0=mgbe0_0
PHC2SYS_SERVICE=/etc/systemd/system/phc2sys-$EN0.service
cat <<EOF | sudo tee $PHC2SYS_SERVICE >/dev/null
[Unit]
Description=Copy system time to $EN0
Requires=NetworkManager.service
After=NetworkManager.service
After=timemaster.service

[Service]
Type=simple
ExecStartPre=timeout 3m bash -c "\
until [ \"\$(nmcli -g GENERAL.STATE device show $EN0)\" = \"100 (connected)\" ]; do \
     sleep 1; \
done"
ExecStart=/usr/sbin/phc2sys -c $EN0 -s CLOCK_REALTIME -O 0 -S 0.0001

[Install]
WantedBy=multi-user.target
EOF
```

11. Configure the service to run at startup, and start it.

```
sudo chmod u+x $PHC2SYS_SERVICE
sudo systemctl enable phc2sys-$EN0.service
sudo systemctl start phc2sys-$EN0.service
```

12. Run **ptp4l** to send PTP SYNC messages to **$EN0**.

```
cat <<EOF | sudo tee /etc/linuxptp/hsb-ptp.conf >/dev/null
# This configuration is appropriate for NVIDIA Holoscan sensor bridge
# applications, where PTP messages are sent over L2 and a 1/2 second interval.
[global]
logSyncInterval -1
logMinDelayReqInterval -1
network_transport L2
EOF
```

13. Create a systemd service to run **ptp4l.**

```
PTP4L_SERVICE=/etc/systemd/system/ptp4l-$EN0.service
cat <<EOF | sudo tee $PTP4L_SERVICE >/dev/null
[Unit]
Description=Send PTP SYNC messages to $EN0
After=phc2sys-$EN0.service

[Service]
Type=simple
ExecStart=/usr/sbin/ptp4l -i $EN0 -f /etc/linuxptp/hsb-ptp.conf

[Install]
WantedBy=multi-user.target
EOF

sudo chmod u+x $PTP4L_SERVICE
sudo systemctl enable ptp4l-$EN0.service
sudo systemctl start ptp4l-$EN0.service
```

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

#### 6.3.2.3. Building the Holoscan Sensor Bridge Demo Container

1. Fetch the Holoscan Sensor Bridge source code from NVIDIA GitHub.
   ```
   git clone https://github.com/nvidia-holoscan/holoscan-sensor-bridge -b 2.7.0
   ```
2. Navigate to the directory at the same level where the `holoscan-sensor-bridge` repository is cloned, copy the patch file **lattice_sensor_bridge_2.7.0.patch** into this directory, and apply the patch to support IMX258 camera module.
   ```
   patch -p0 < lattice_sensor_bridge_2.7.0.patch
   ```
3. Verify that **linux_imx258_player.py** exists after applying **lattice_sensor_bridge_2.7.0.patch** to confirm that the patch was applied correctly.
   ```
   ls holoscan-sensor-bridge/examples/linux_imx258_player.py
   ```
   Expected output is as follows:
   ```
   holoscan-sensor-bridge/examples/linux_imx258_player.py
   ```
4. Log in to NVIDIA GPU Cloud (NGC) using the browser on AGX Thor.
   - Register for a developer account for NGC at https://catalog.ngc.nvidia.com/
   - Create an API key at: https://ngc.nvidia.com/setup/api-key
   - Use the API key to log in to **nvcr.io.** Run the command *docker login nvcr.io*
     ```
     docker login nvcr.io
     Username: $oauthtoken
     Password: <Your token key to NGC>
     WARNING! Your password will be stored unencrypted in /home/<user>/.docker/config.json.
     Configure a credential helper to remove this warning. See
     https://docs.docker.com/engine/reference/commandline/login/#credentials-store

     Login Succeeded
     ```
5. Rebuild the Holoscan Sensor Bridge demo container using the following command.
   ```
   cd holoscan-sensor-bridge
   sh docker/build.sh --igpu
   ```
6. To validate the setup, run the video streaming test on AGX Thor as described in Testing Camera Video Streaming.

## 6.4. Testing the System

This section describes the procedures for testing the reference design and programming the bitstream on both the CertusPro-NX Versa Board and the CertusPro-NX Sensor-to-Ethernet Bridge Board. The reference design testing includes the following functionalities:

- Video streaming
  - ISP and MJPEG features (supported only on CertusPro-NX Versa Board)
- Body pose estimation

### 6.4.1. Test Setup on the CertusPro-NX Versa Board

#### 6.4.1.1. Programming the Bit File

This section outlines the steps for uploading the *.bit* file. There are two methods to update the reference design firmware on the CertusPro-NX Versa Board.

1. Local firmware update through USB interface.

   Flash the firmware directly into the board using a USB connection, independent of the Holoscan ecosystem. This method is ideal for initial setup, debugging, and hands-on development.
2. Remote firmware update through Ethernet

   Update the firmware remotely over an Ethernet connection using the Holoscan ecosystem. This method is suited for field updates and production environments, where remote access and automation are essential.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

The following sections describe how to upload the **.bit** file using each method.

### Local Firmware Update through USB Interface

1. Connect the board to the PC using a mini-USB Type-A cable, as shown in Figure 6.29. Connect the power adaptor to the board power port and turn the switch **ON.** The board is now ready to be programmed.

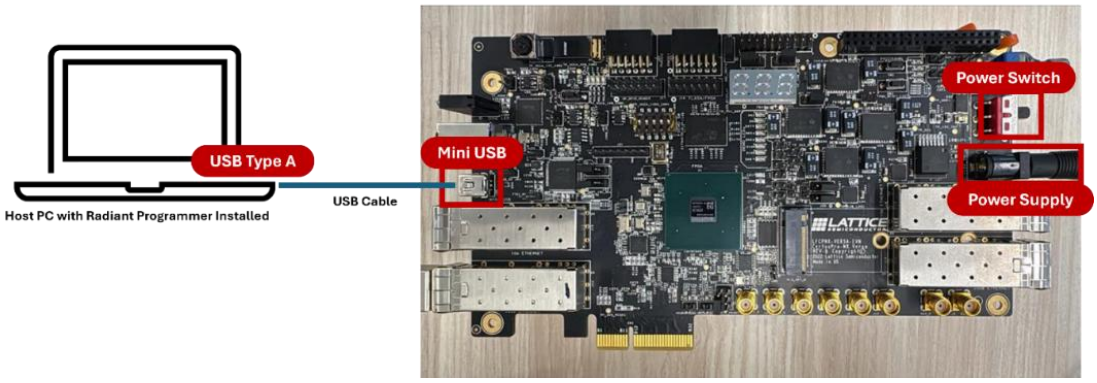



**Figure 6.29. Hardware Setup for Bit File Programming**

2. Click the Radiant Programmer icon ( ) if the project is already open in the Lattice Radiant software, otherwise, launch the standalone Radiant Programmer. Enter the **Project Location**, **Project Name**, then click **OK**.

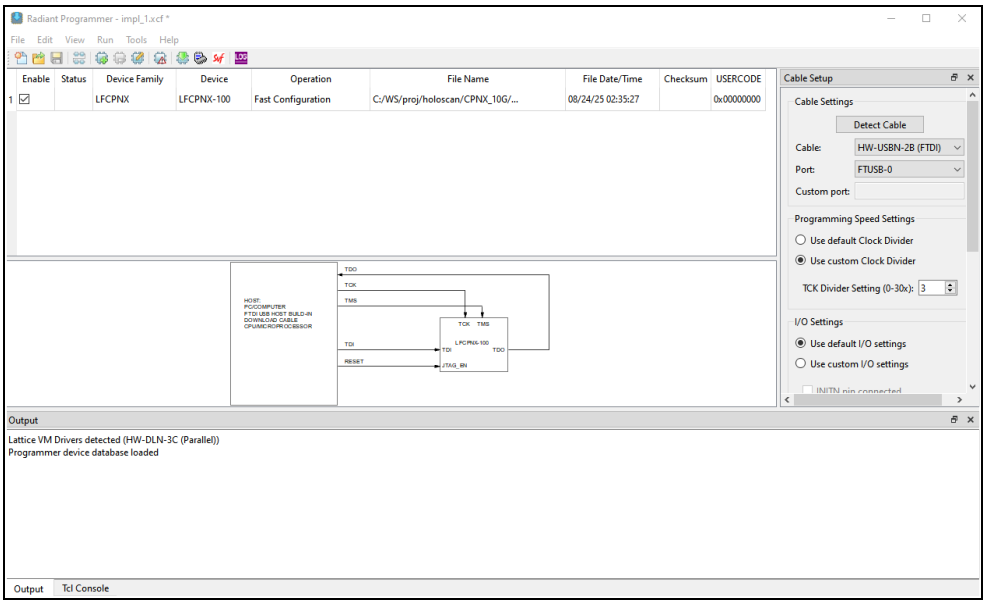



**Figure 6.30. Radiant Programmer Window**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

3. Click **Detect Cable** in the cable setup section, select **FTUSB-0**. Then click **OK** to proceed.

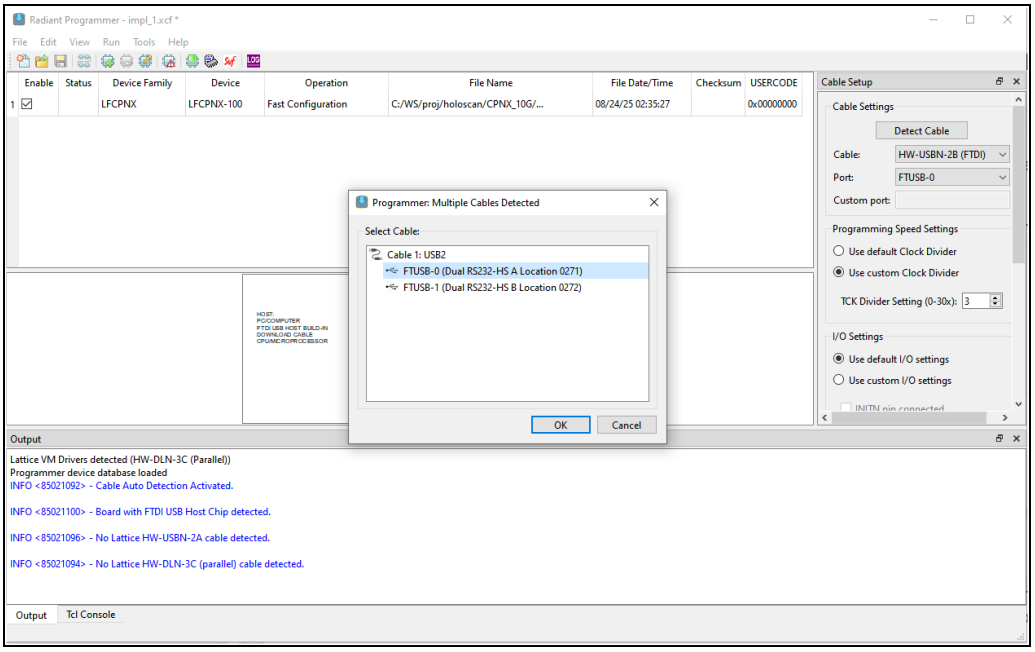



**Figure 6.31. Select Cable Settings**

4. Select the appropriate SPI flash configuration to flash directly to external SPI flash. Refer to Figure 6.32 for the expected settings. Click the **…** button under the file name bar and choose the bitstream file to program.

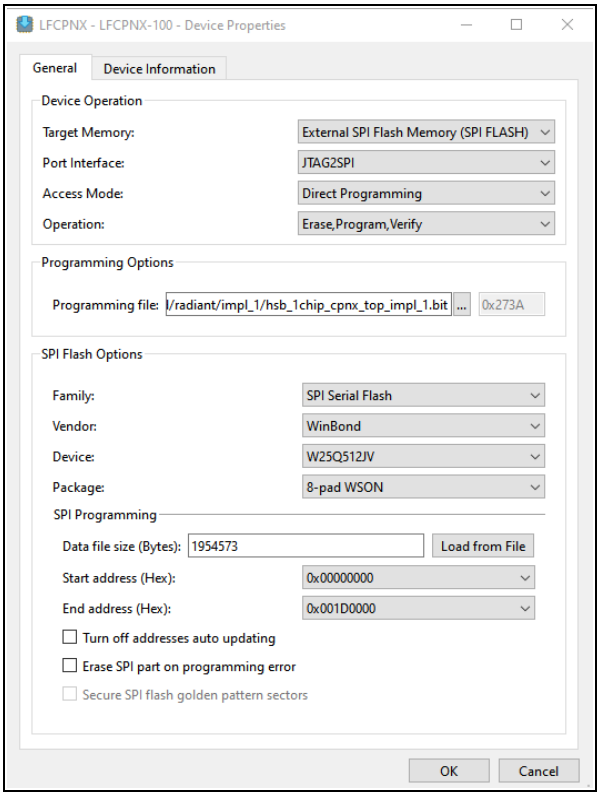



**Figure 6.32. Load Bitstream File**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

5. Click the **Program Device** toolbar icon to load the bitstream into the board, as shown in Figure 6.33.



**Figure 6.33. Program Device Toolbar Icon**

6. Verify that the programming status is **Operation: successful** in the output window, as shown in Figure 6.34. To activate the flashed bitstream, power cycle the CertusPro-NX Versa board by switching it **OFF**, then **ON**.

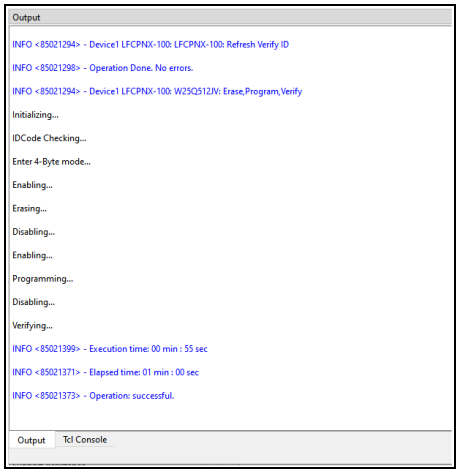



**Figure 6.34. Message on Successful Programming**

### Remote Firmware Update through Ethernet

Before using this method, note the following prerequisites:

- If you are using the CertusPro-NX Versa board for the first time, perform a Local Firmware Update through USB. After the initial update, you may use the remote firmware update via Ethernet for future updates.
- For firmware changes involving a switch between 1GbE[1], 2.5GbE[1], and 10GbE bitstreams on the same board, use the Local Firmware Update through USB.
- Before proceeding, complete the steps in Building the Holoscan Sensor Bridge Demo Container.

If all prerequisites are met, follow the steps below:

1. Connect the AGX Orin/Thor ethernet port to the CertusPro-NX Versa board 10GbE/1GbE Ethernet port (SFP Module). Refer to Figure 6.1 for the setup.
2. Select and copy the following 1GbE[1]/2.5GbE[1]/10GbE FPGA bitstream file into the holoscan-sensor-bridge directory.
   - Bitstream file supporting 10GbE Ethernet: fpga_cpnx_versa_0108_2606.bit
   - Bitstream file supporting 2.5GbE Ethernet: fpga_cpnx_versa_0502_2606.bit
   - Bitstream file supporting 1GbE Ethernet: fpga_cpnx_versa_0207_2606.bit
3. Open the terminal on the AGX Orin.
4. Change directory to the holoscan-sensor-bridge directory.
   ```
   cd holoscan-sensor-bridge
   ```
5. Update the FPGA bitstream for 1GbE[1], 2.5GbE[1], or 10GbE Ethernet using the command below. The process takes 20 to 30 minutes to complete.
   - Using bitstream that supports 10GbE Ethernet:
     ```
     program_lattice_cpnx_versa scripts/manifest_cpnx_versa_10g.yaml
     ```
   - Using bitstream that supports 2.5GbE Ethernet:
     ```
     program_lattice_cpnx_versa scripts/manifest_cpnx_versa_2p5g.yaml
     ```
   - Using bitstream that supports 1GbE[1] Ethernet:
     ```
     program_lattice_cpnx_versa scripts/manifest_cpnx_versa_1g.yaml
     ```

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

6. Wait for the process to complete. When prompted, power cycle the Versa board, then press *Enter* to exit the program. Refer to Figure 6.35 for the expected output.

```
INFO spi_flash_base.cpp:169 status tid=0xa0 -- CPNX: verify address=0x100000
INFO spi_flash_base.cpp:169 status tid=0xa0 -- CPNX: verify address=0x110000
INFO spi_flash_base.cpp:169 status tid=0xa0 -- CPNX: verify address=0x120000
INFO spi_flash_base.cpp:169 status tid=0xa0 -- CPNX: verify address=0x130000
INFO spi_flash_base.cpp:169 status tid=0xa0 -- CPNX: verify address=0x140000
INFO spi_flash_base.cpp:169 status tid=0xa0 -- CPNX: verify address=0x150000
INFO spi_flash_base.cpp:169 status tid=0xa0 -- CPNX: verify address=0x160000
INFO spi_flash_base.cpp:169 status tid=0xa0 -- CPNX: verify address=0x170000
INFO spi_flash_base.cpp:169 status tid=0xa0 -- CPNX: verify address=0x180000
INFO spi_flash_base.cpp:169 status tid=0xa0 -- CPNX: verify address=0x190000
INFO spi_flash_base.cpp:169 status tid=0xa0 -- CPNX: verify address=0x1A0000
INFO spi_flash_base.cpp:169 status tid=0xa0 -- CPNX: verify address=0x1B0000
INFO spi_flash_base.cpp:169 status tid=0xa0 -- CPNX: verify address=0x1C0000
INFO spi_flash_base.cpp:169 status tid=0xa0 -- CPNX: verify address=0x1D0000
You must now physically power cycle the sensor bridge device.
Press <Enter> to continue:
root@ubuntu-agx:/home/agx/hsb/holoscan-sensor-bridge-2.2.0#
```

**Figure 6.35. Console Output — Remote Firmware Update**

7. Power cycle the Versa board by switching it **off**, and then **on** to activate the newly flashed bitstream.

**Note:**

1. Only Jetson AGX Orin supports 1GbE and 2.5GbE Ethernet.

#### 6.4.1.2. Programming the Golden FPGA Bitstream with a Jump Table

To enable dual-boot operation, you must include both a primary FPGA bitstream (the default boot image) and a golden FPGA bitstream (a reliable fallback for recovery). Generate a PROM file by combining the primary and golden bitstreams with the required jump table. Use this PROM file to program the external SPI flash device.

1. Generate the PROM file.
   a. Open the Lattice Radiant Programmer.
   b. In Radiant Programmer, select **Tools > Deployment Tool**. The window shown in Figure 6.36 opens.

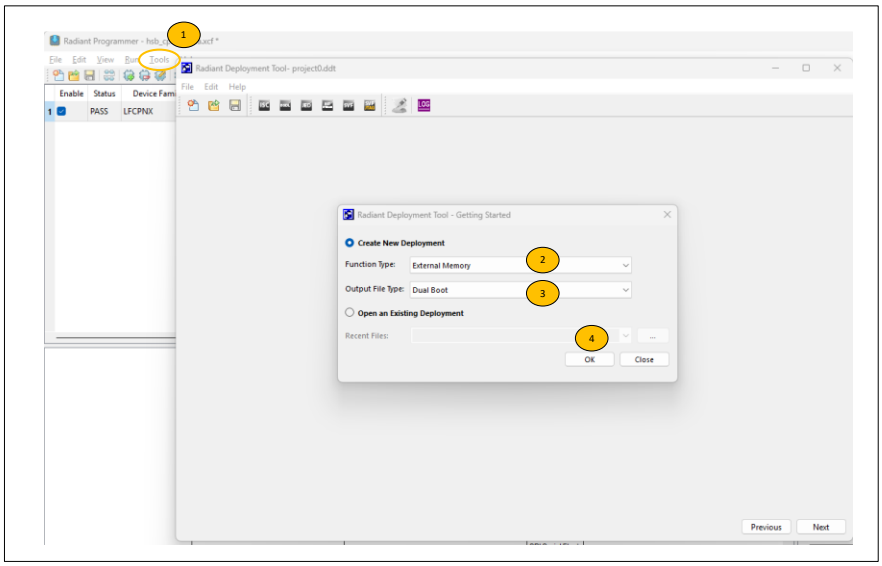



**Figure 6.36. Radiant Deployment Tool – Getting Started**

   c. Set Function to External Memory and Output File Type to Dual Boot, and then click OK.
   d. Add your primary and golden bitstream files, regardless of orientation.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

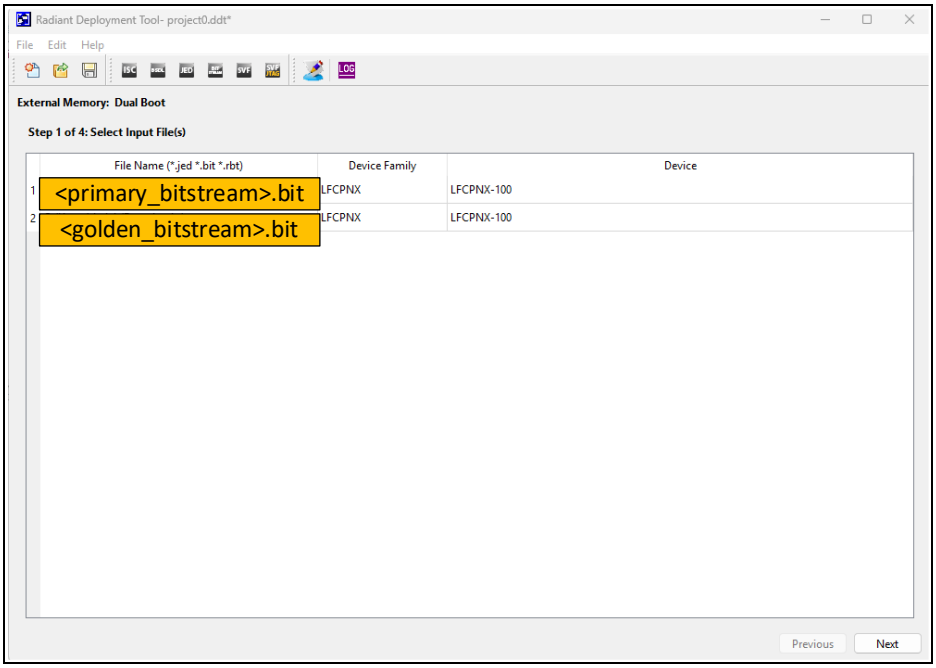



**Figure 6.37. Select Input File**

e. Configure the dual-boot option as required; ensure the golden and primary FPGA bitstream files are correctly selected.

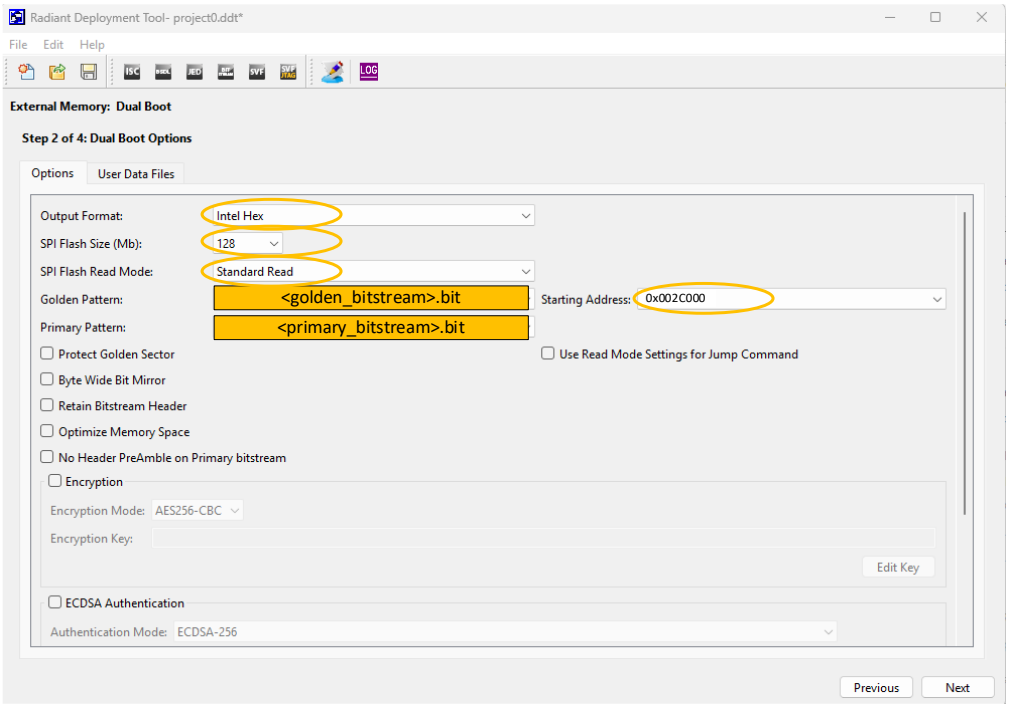



**Figure 6.38. Dual Boot Options**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

f. Select the output file name and the save location.

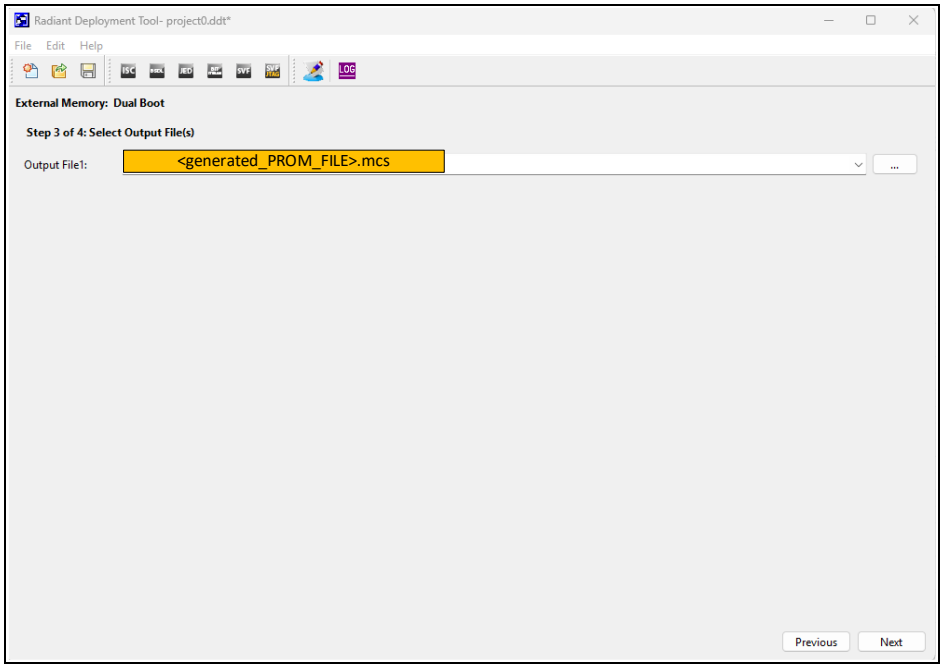



**Figure 6.39. Select Output File**

g. Generate the PROM file and confirm that the log shows a successful generation message.

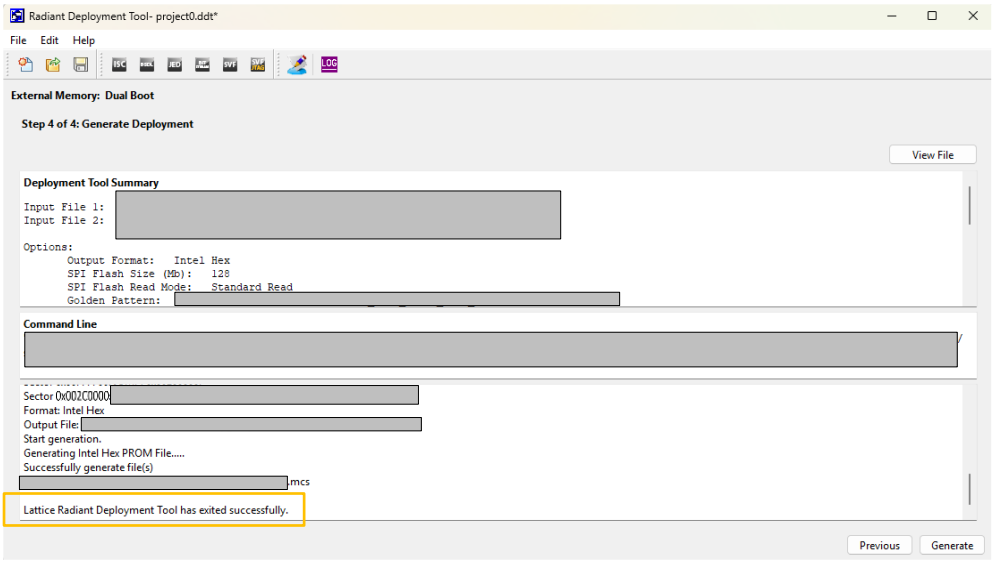



**Figure 6.40. Generate Deployment**

2. Program the PROM (.mcs) file.

   Programming the PROM file is the same as programming the bitstream file described in the Local Firmware Update through USB Interface section. The only difference is that the input is the PROM file with the *.mcs* extension.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

#### 6.4.1.3. Testing the Camera Video Streaming

Table 6.2 lists the parameter settings used in the Python scripts for video streaming, body pose estimation, and TAO PeopleNet evaluations. These configurations were applied to all testing activities and are tailored for the IMX258 camera.

**Table 6.2. IMX258 Test Command Parameters and Default Values**

| Parameter | Parameter Value and Description | Default Value |
|---|---|---|
| `--runtime-mipi` | **`0`**: Used for fixed lane rate and number MIPI IPK<br>**`1`**: Used for runtime MIPI IPK | 1 |
| `--format` | **`raw10`**: Camera streaming data will be in RAW10 format<br>**`mjpeg`**: Camera streaming data will be in MJPEG format<br>**`yuv422`**: Camera streaming data will be in YUV422 format | raw10 |
| `--focus` | Integer value from `-450` to `450` to set the camera focus distance | -200 |
| `--camera-mode` | **`0`**: Set camera mode to 1920x1080 60FPS<br>**`1`**: Set camera mode to 1920x1080 30FPS | 0 |

**Note:** The MJPEG format is currently limited to an output frame rate of 30 fps in this release.

Follow these steps:

1. Run the Holoscan Sensor Bridge demo container.
   ```
   cd /<path>/<to>/<holoscan-sensor-bridge>
   xhost +
   sh docker/demo.sh
   ```
   Note: After running the demo.sh script on the terminal, the demo container starts automatically, and the command prompt switches to the container environment.
2. Run the video streaming demo for the IMX258 camera. Select the appropriate command for your Ethernet configuration:
   - 10GbE Ethernet:
     ```
     python3 examples/linux_imx258_player.py
     ```
   - 2.5GbE[1] Ethernet:
     ```
     python3 examples/linux_imx258_player.py
     ```
   - 1GbE[1] Ethernet:
     ```
     python3 examples/linux_imx258_player.py --camera-mode=1
     ```
     **Note:** Set the camera mode to 1. The 1GbE Ethernet configuration supports 1920 × 1080 at 30 fps only.

**Note:**
1. Jetson AGX Orin only.

Figure 6.41 shows the camera video streaming test setup with AGX Orin. The video is captured in an office environment under lighting conditions between 135 lux and 140 lux. The test environment is identical for both 1GbE[1], 2.5GbE[1] , and 10GbE Ethernet configurations. Image quality remains consistent across all Ethernet configurations.

The primary objective of this reference design is to establish a robust and reliable bridge between the HSB interfaces and the NVIDIA host. Image tuning is not included. This process typically involves defining image quality parameters, configuring processing pipelines, and optimizing the image signal processor (ISP). These tasks are highly specific to the camera sensor and require dedicated resources.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

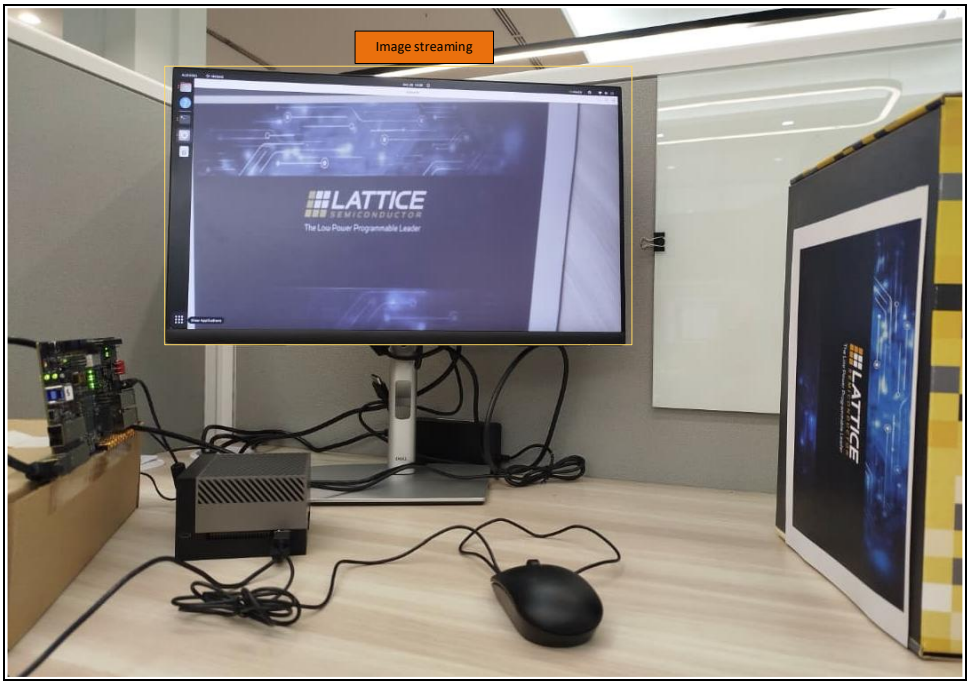



**Figure 6.41. Camera Video Streaming Test Setup**

#### 6.4.1.4. Testing the Camera with ISP and MJPEG Features

In this example, the system is configured using a 10GbE Ethernet interface, with the camera operating at a resolution of 1920 × 1080 and 60 fps. The video streaming function supports three output formats: RAW10, YUV422, and MJPEG. Refer to Table 6.2 for details.

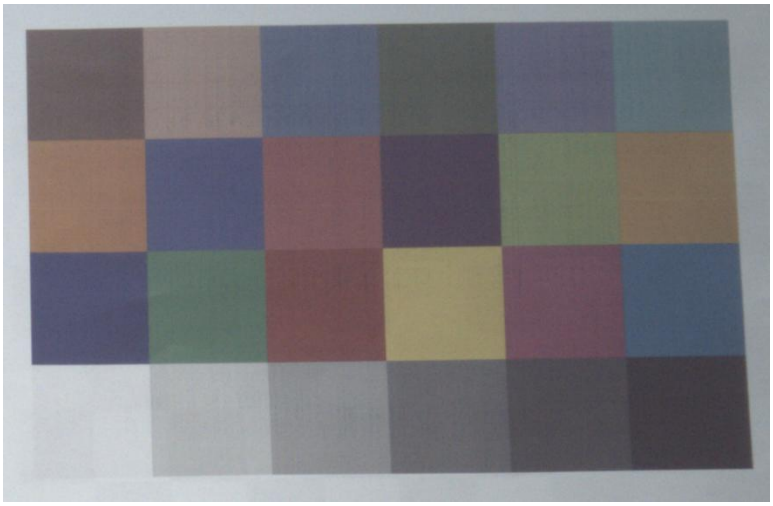
1. Run the Holoscan Sensor Bridge demo container.

```
cd /<path>/<to>/<holoscan-sensor-bridge>
xhost +
sh docker/demo.sh
```

   **Note:** After running the *demo.sh* script on the terminal, the demo container starts automatically, and the command prompt switches to the container environment.

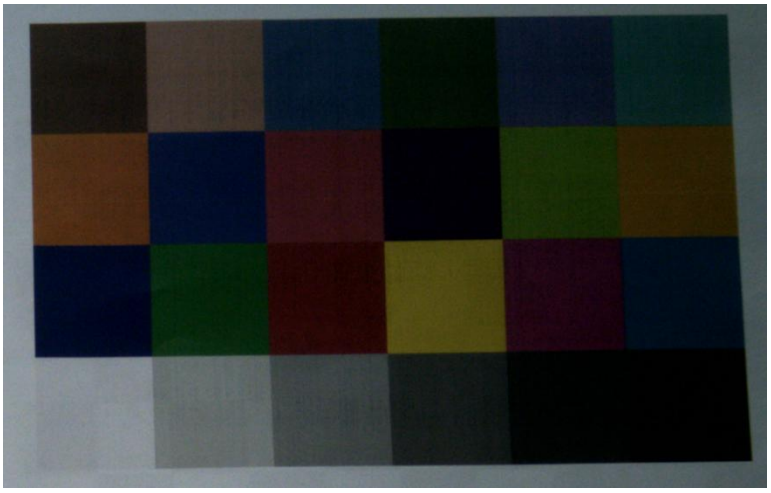
2. Run the video streaming example. To select a streaming format, use the format parameter with mjpeg, raw10, or yuv422. Adjust the lens focus as needed using the focus parameter.

```
python3 examples/linux_imx258_player.py --format=mjpeg --focus=-200
```



RAW10 – Output using built-in NVIDIA ISP and visualizer.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.



MJPEG – ISP output (YUV422) converts to MJPEG



YUV422 – ISP enabled output with custom tuning settings for IMX258

**Figure 6.42. Output Image under Different Video Streaming Format**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

#### 6.4.1.5. Testing the Camera Body Pose

1. Run the Holoscan Sensor Bridge demo container.

```
cd /<path>/<to>/<holoscan-sensor-bridge>
xhost +
sh docker/demo.sh
```

**Note:** After running the *demo.sh* script on the terminal, the demo container starts automatically, and the command prompt switches to the container environment.

2. Complete the one-time prerequisite setup for Body Pose Estimation. Run the following commands inside the container environment.

```
apt-get update && apt-get install -y ffmpeg
pip3 install ultralytics onnx
cd examples
yolo export model=yolov8n-pose.pt format=onnx
trtexec --onnx=yolov8n-pose.onnx --saveEngine=yolov8n-pose.engine.fp32
cd -
```

**Note:** This setup only needs to be performed once.

3. Run the body pose estimation example to display live pose tracking in the video, as shown in Figure 6.43. The detailed parameter configurations are listed in Table 6.2.

```
python3 examples/linux_body_pose_estimation_imx258.py --format=raw10
```

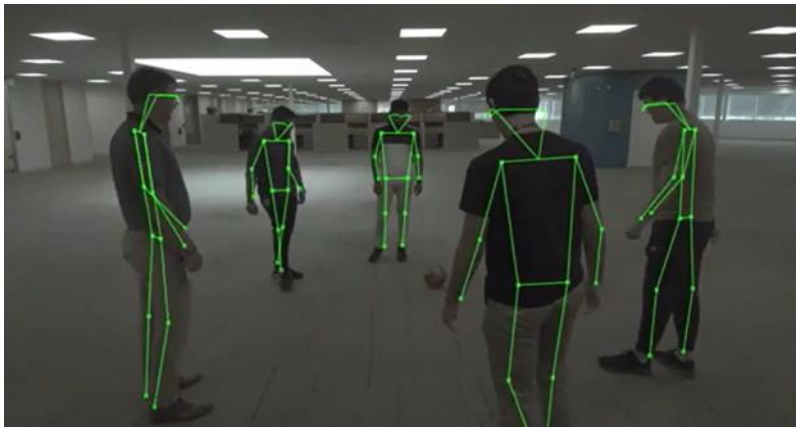


**Figure 6.43. Live Body Pose Estimation Output**

#### 6.4.1.6. Testing the Camera TAO PeopleNet

1. Run the following commands outside the container environment to download the TAO PeopleNet model required by the Holoscan Sensor Bridge demo:

```
cd /<path>/<to>/<holoscan-sensor-bridge>
wget --content-disposition \
"https://api.ngc.nvidia.com/v2/models/org/nvidia/team/tao/peoplenet/pruned_quantized_decrypted_v2.3.3\
/files?redirect=true&path=resnet34_peoplenet_int8.onnx" -O examples/resnet34_peoplenet_int8.onnx
```

**Note:** This setup only needs to be performed once. Launch the Holoscan Sensor Bridge demo container..

```
xhost +
sh docker/demo.sh
```

**Note:** After running the *demo.sh* script on the terminal, the demo container starts automatically and the command prompt switches to the container environment.

3. Run the TAO PeopleNet example to display live person detection in the video, as shown in Figure 6.44. The detailed parameter configurations are listed in Table 6.2.

```
python3 examples/linux_tao_peoplenet_imx258.py --format=raw10
```

**Note:** On the first run, the script may take at least 10 minutes to complete the setup.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.



**Figure 6.44. Live TAO PeopleNet Output**

## 6.4.2. Test Setup on the CertusPro-NX Sensor-to-Ethernet Bridge Board

### 6.4.2.1. Programming the Bit File

This section outlines the steps for uploading the .bit file. There are two methods to update the reference design firmware on the CertusPro-NX Sensor-to-Ethernet Bridge Board.

1. Local firmware update through USB interface.

Flash the firmware directly into the board using a USB connection, independent of the Holoscan ecosystem. This method is ideal for initial setup, debugging, and hands-on development.

2. Remote firmware update over Ethernet.

Update the firmware remotely over an Ethernet connection using the Holoscan ecosystem. This method is suited for field updates and production environments, where remote access and automation are essential.

The following sections describe in detail the steps of uploading the *.bit* file using both methods.

**Local Firmware Update through JTAG**

1. Use the Lattice Radiant Programmer to program the board. Launch it as a standalone tool or from a Lattice Radiant project. Set up the hardware as shown in Figure 6.45, and connect the Lattice Programming Cable to header J22 (Figure 6.46).

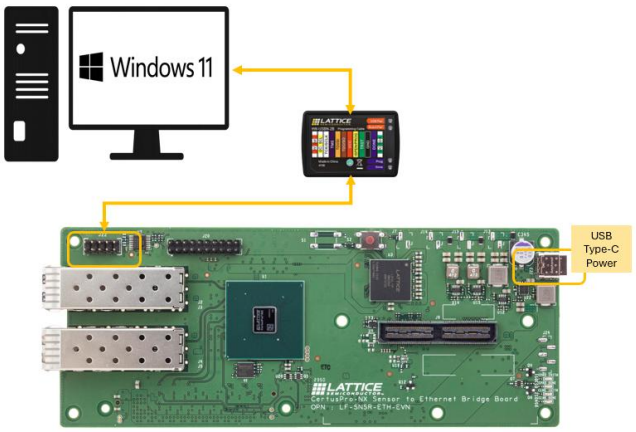
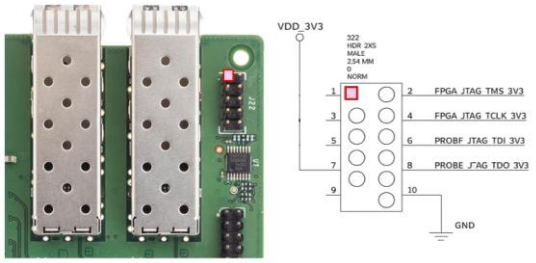



**Figure 6.45. Bit File Programming Hardware Setup**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.




| LF-SNSR-ETH-EVN | J22 NET NAME | Lattice Programming Cable | Cable Color |
|---|---|---|---|
| J22.2 | FPGA_JTAG_TMS_3V3 | TMS | Purple |
| J22.4 | FPGA_JTAG_TCLK_3V3 | TCK/SCLK | White |
| J22.6 | PROBE_JTAG_TDI_3V3 | TDI/SI | Orange |
| J22.7 | VDD_3V3 | VCC | Red |
| J22.8 | PROBE_JTAG_TDO_3V3 | TDO/SO | Brown |
| J22.10 | GND | GND | Black |

**Figure 6.46. Lattice Programming Cable Pin Configuration**

2. Power on the board by connecting a USB Type-C power source.
3. Start a programming session by launching the tool and running a board scan (Figure 6.47).

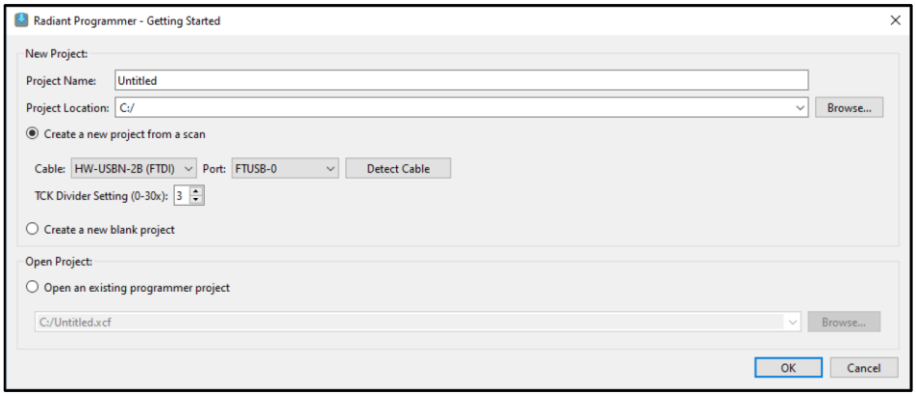



**Figure 6.47. Radiant Programmer – Getting Started**

4. When the board is detected, the window shown in Figure 6.48 opens. This interface allows you to enter the file name.

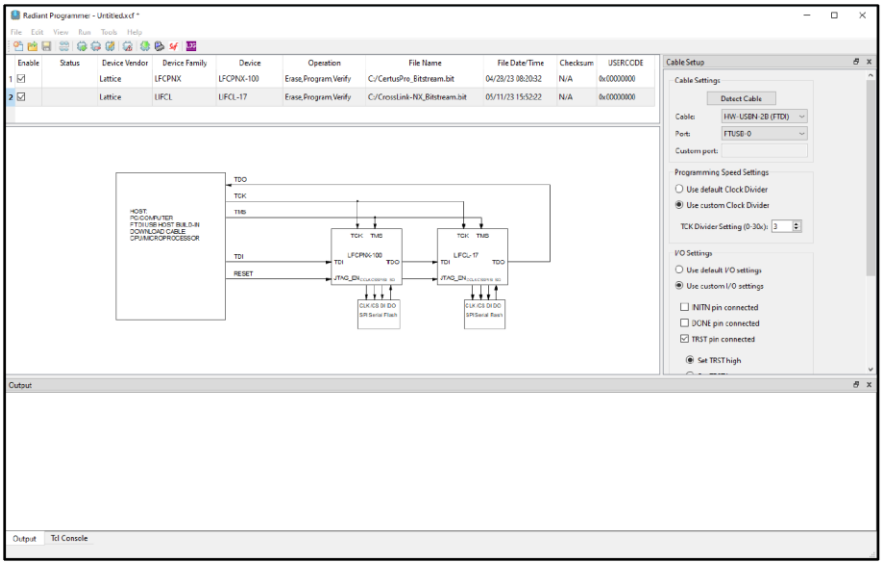


**Figure 6.48. External Flash Configuration**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

5. Double-click the Operation field and select the appropriate programming mode. In this example, select Erase, Program, Verify for External SPI Flash Memory (SPI flash) (Figure 6.49 and Figure 6.50).

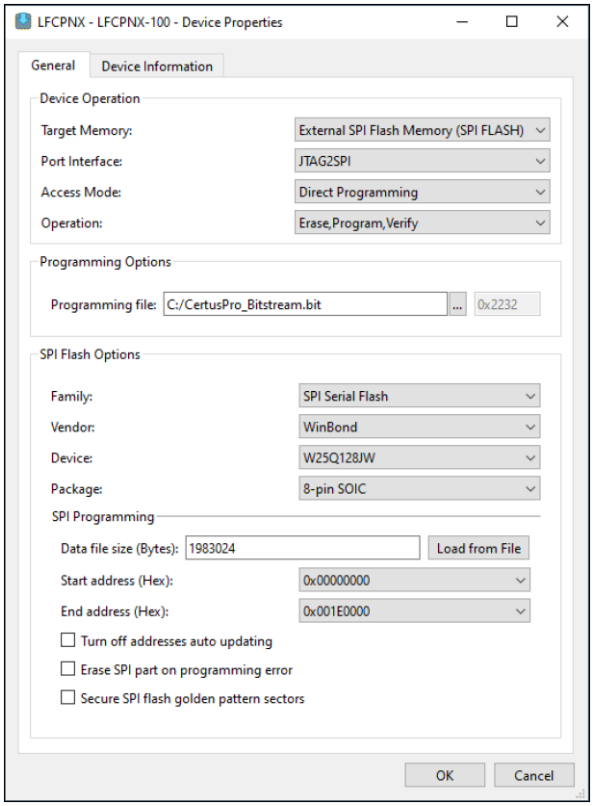



**Figure 6.49. External Flash Memory (SPI flash) Device Properties – CertusPro-NX Device**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

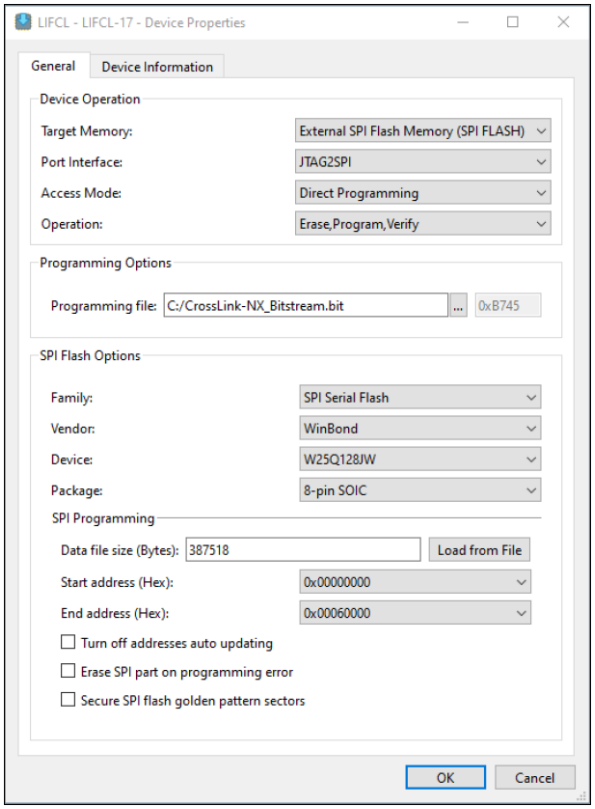



**Figure 6.50. External Flash Memory (SPI flash) Device Properties – CrossLink-NX Device**

6. Back in the main window, click Program to program both flashes on the CertusPro-NX Sensor-to-Ethernet Bridge board.
7. Verify the programming result in the Output console at the bottom of Lattice Radiant Programmer. You should see **Operation: Successful** (Figure 6.51).

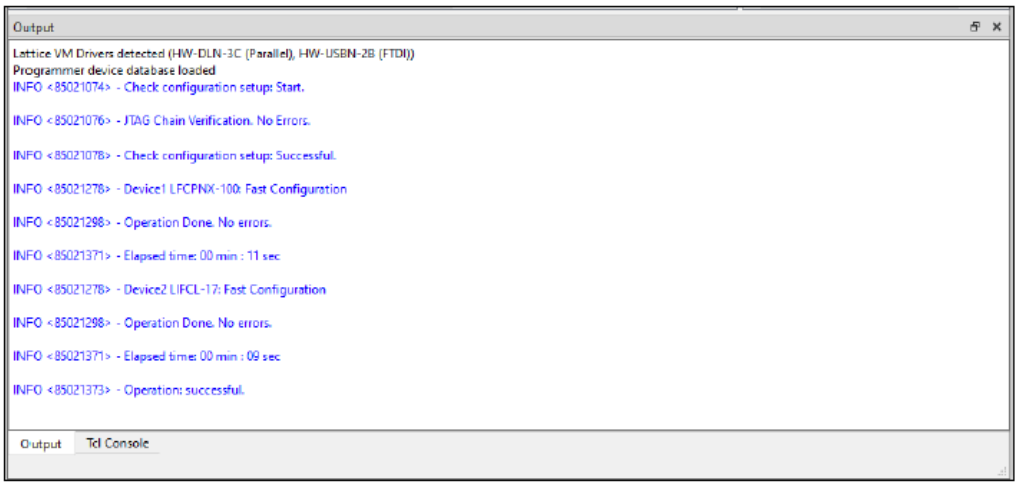



**Figure 6.51. Output Window**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

**Remote Firmware Update through Ethernet**

Before using this method, note the following prerequisites:

- If you are using the CertusPro-NX Sensor-to-Ethernet Bridge Board for the first time, perform a Local Firmware Update through JTAG.
- After the initial update, you may use the remote firmware update through Ethernet for future updates.

If all prerequisites are met, follow the steps below:

1. Connect the AGX Orin Ethernet port to the 10GbE Ethernet port (SFP module) on the CertusPro-NX Sensor-to-Ethernet Bridge Board. See Figure 6.4 for the setup.
2. Select and copy the following 10GbE FPGA bitstream into the holoscan-sensor-bridge directory.
   - Bitstream that supports 10GbE Ethernet: fpga_cpnx_2chip_0003_2606.bit
3. Open a terminal on the AGX Orin.
4. Change to the holoscan-sensor-bridge directory.
   ```
   cd holoscan-sensor-bridge
   ```
5. Update the FPGA bitstream for 10GbE Ethernet using the following command. The process takes 20 to 30 minutes to complete.

   Using the bitstream that supports 10GbE Ethernet:
   ```
   program_lattice_cpnx100 scripts/manifest_cpnx_2chip_10g.yaml
   ```
6. Wait for the process to complete. When prompted, power-cycle the Sensor-to-Ethernet Bridge board, then press Enter to exit the program. See Figure 6.52 for the expected output.

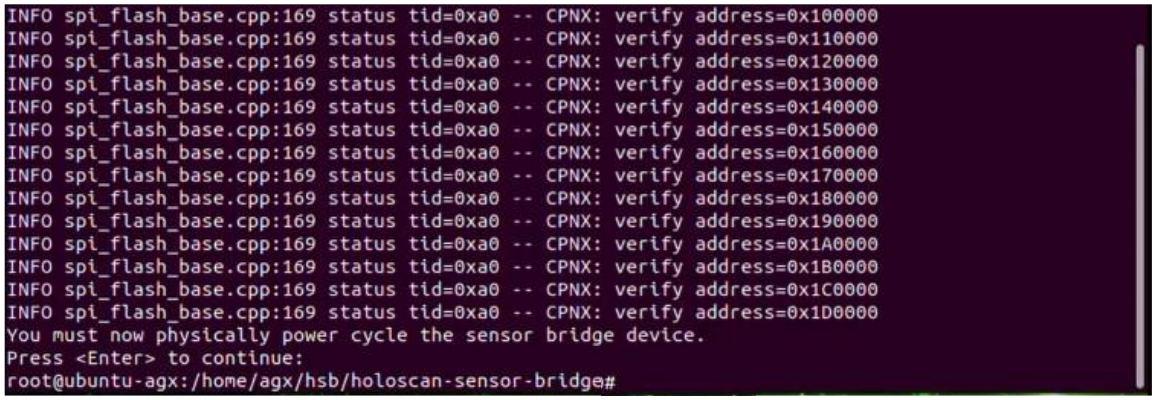



**Figure 6.52. Console Output – Remote Firmware Update**

7. Power-cycle the Sensor-to-Ethernet Bridge board: turn it off, then on to activate the newly flashed bitstream.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

#### 6.4.2.2. Programming the Golden FPGA Bitstream with a Jump Table

To enable dual-boot operation, you must include both a primary FPGA bitstream (the default boot image) and a golden FPGA bitstream (a reliable fallback for recovery). You generate a PROM file by combining the primary and golden bitstreams with the required jump table. Use this PROM file to program the external SPI flash device.

1. Generate the PROM file.
   a. Open the Lattice Radiant Programmer.
   b. In Radiant Programmer, select **Tools > Deployment Tool**. The window shown in Figure 6.53 opens.

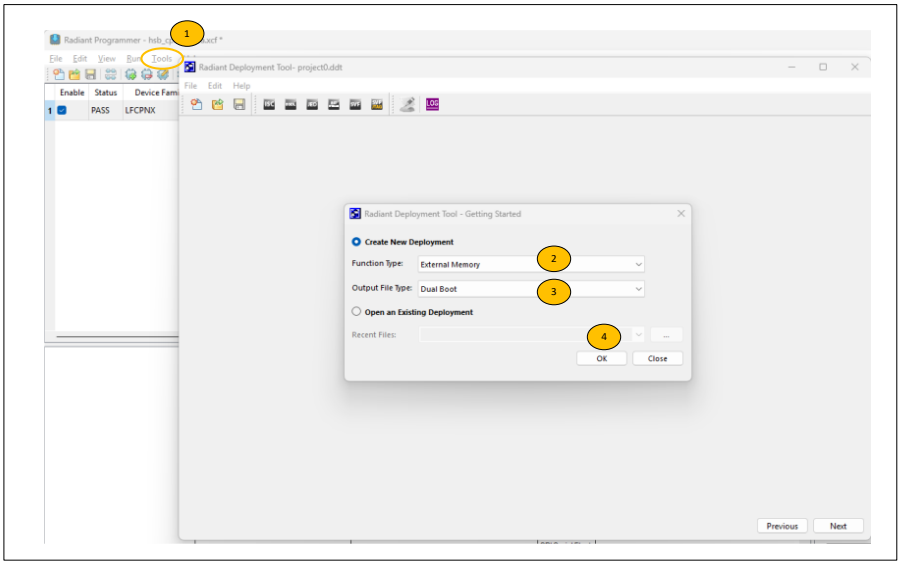



**Figure 6.53. Radiant Deployment Tool – Getting Started**

   c. Set Function to External Memory and Output File Type to Dual Boot, and then click OK.
   d. Add your primary and golden bitstream files, regardless of orientation.

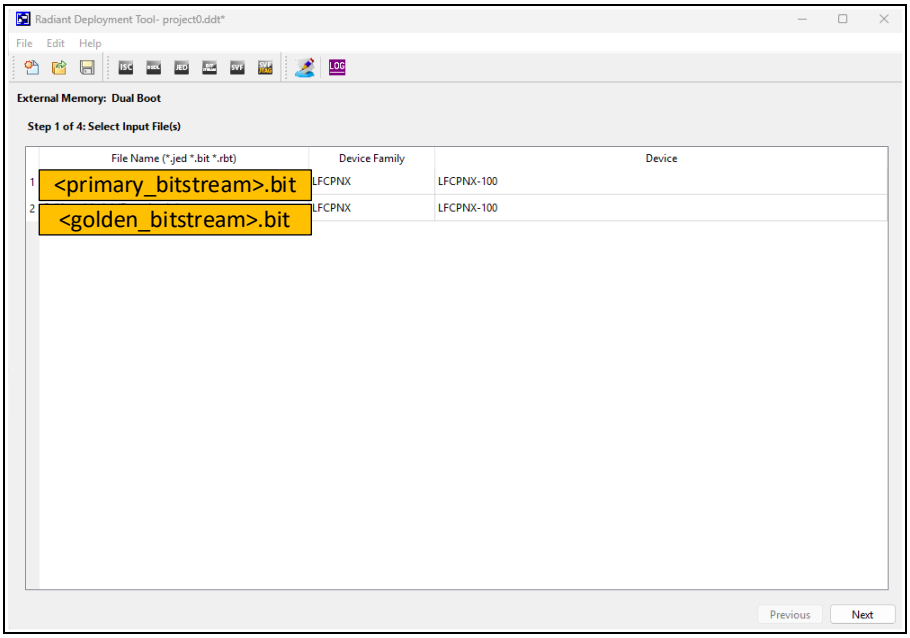



**Figure 6.54. Select Input File**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

e. Configure the dual-boot option as required; ensure the golden and primary FPGA bitstream files are correctly selected.

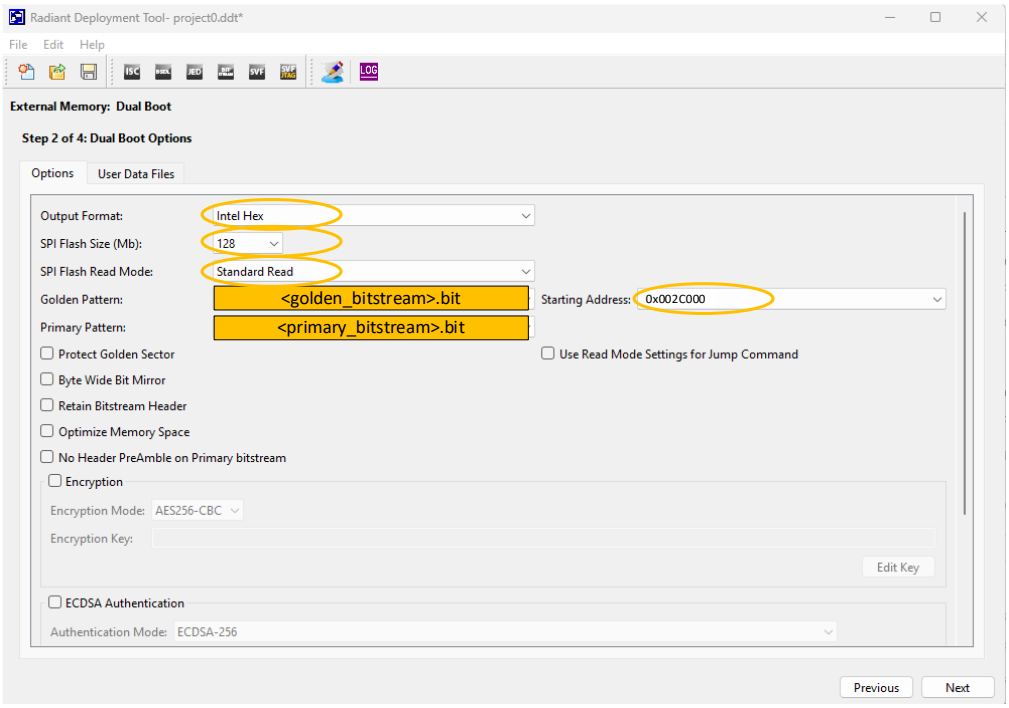



**Figure 6.55. Dual Boot Options**

f. Select the output file name and the save location.

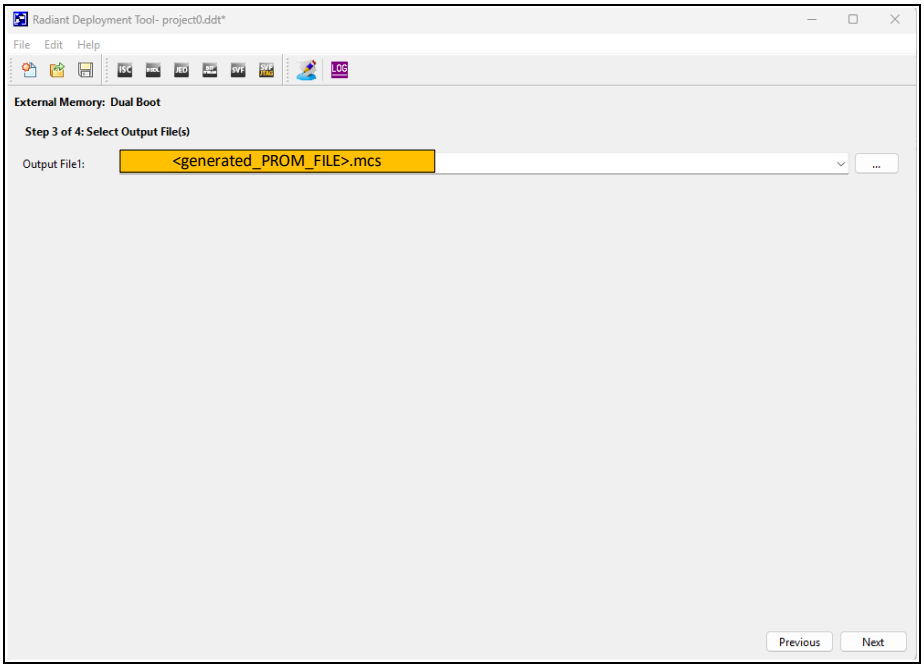



**Figure 6.56. Select Output File**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

g. Generate the PROM file and confirm that the log shows a successful generation message.

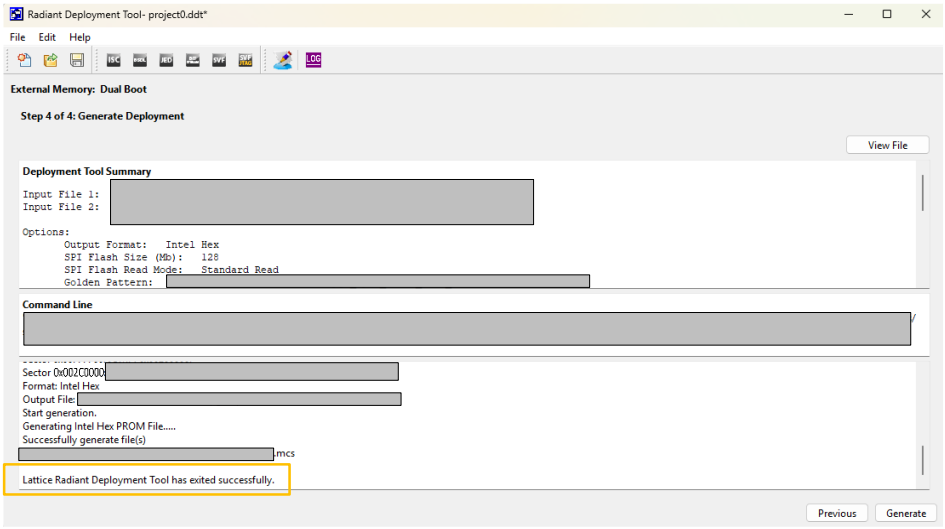



**Figure 6.57. Generate Deployment**

2. Program the PROM (.mcs) file.

   Programming the PROM file is the same as programming the bitstream file described in the Local Firmware Update through JTAG section. The only difference is that the input is the PROM file with the .mcs extension.

#### 6.4.2.3. Testing the Camera Video Streaming

1. Run the Holoscan Sensor Bridge demo container.

```
cd /<path>/<to>/<holoscan-sensor-bridge>
xhost +
sh docker/demo.sh
```

   **Note:** After running the *demo.sh* script on the terminal, the demo container starts automatically, and the command prompt switches to the container environment.

2. Run the video streaming demo for the IMX274 camera. Select the appropriate command for your Ethernet configuration:
   - 10GbE Ethernet using single camera:

```
python3 examples/linux_imx274_player.py
```

   - 10GbE Ethernet using dual camera:

```
python3 examples/linux_single_network_stereo_imx274_player.py
```

Figure 6.58 shows the camera video streaming test setup with AGX Orin. The video is captured in an office environment under lighting conditions between 135 lux and 140 lux. The test environment uses a 10GbE Ethernet configuration.

The primary objective of this reference design is to establish a robust and reliable bridge between the HSB interfaces and the NVIDIA host. Image tuning is not included. This process typically involves defining image quality parameters, configuring processing pipelines, and optimizing the image signal processor (ISP). These tasks are highly specific to the camera sensor and require dedicated resources.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

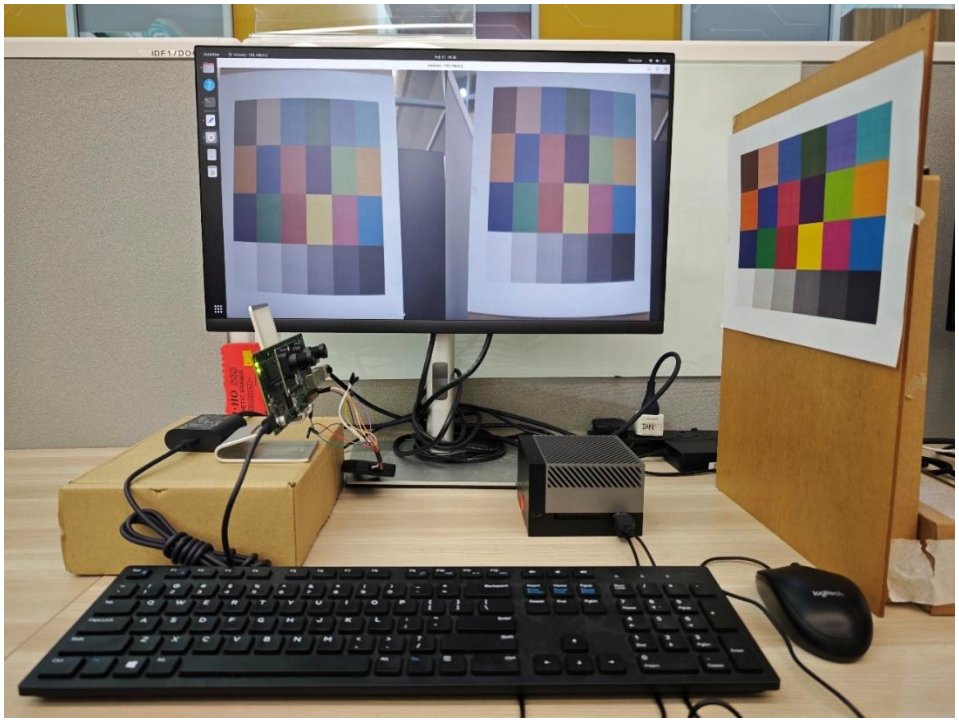


**Figure 6.58. Dual IMX274 Camera Video Streaming Test Setup**

#### 6.4.2.4. Testing the Camera Body Pose

1. Run the Holoscan Sensor Bridge demo container.

```
cd /<path>/<to>/<holoscan-sensor-bridge>
xhost +
sh docker/demo.sh
```

**Note:** After running the *demo.sh* script on the terminal, the demo container starts automatically, and the command prompt switches to the container environment.

2. Complete the one-time prerequisite setup for Body Pose Estimation. Run the following commands inside the container environment.

```
apt-get update && apt-get install -y ffmpeg
pip3 install ultralytics onnx
cd examples
yolo export model=yolov8n-pose.pt format=onnx
trtexec --onnx=yolov8n-pose.onnx --saveEngine=yolov8n-pose.engine.fp32
cd -
```

**Note:** This setup only needs to be performed once.

3. Run the body pose estimation example to display live pose tracking in the video, as shown in Figure 6.59.

```
python3 examples/linux_body_pose_estimation.py
```

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

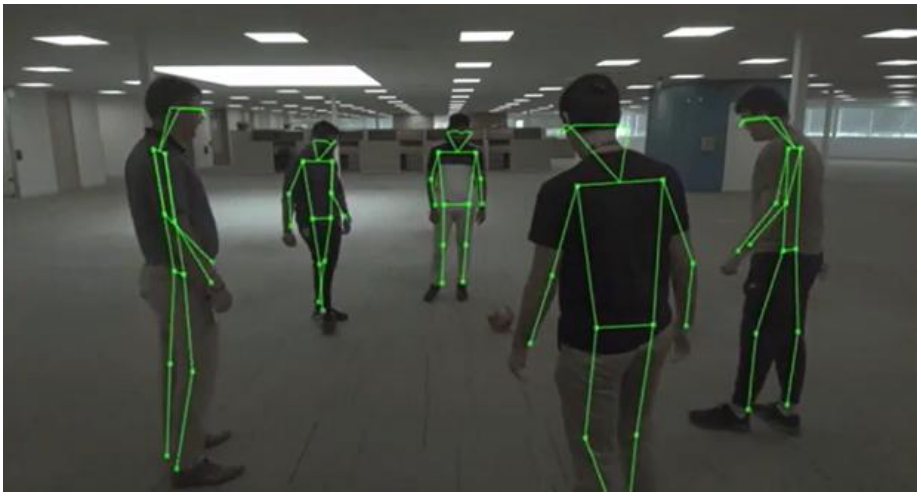


**Figure 6.59. Live Body Pose Estimation Output**

#### 6.4.2.5. Testing Camera TAO PeopleNet

1. 1. Run the following commands outside the container environment to download the TAO PeopleNet model required by the Holoscan Sensor Bridge demo:

```
cd /<path>/<to>/<holoscan-sensor-bridge>
wget --content-disposition \
"https://api.ngc.nvidia.com/v2/models/org/nvidia/team/tao/peoplenet/pruned_quantized_decrypted_v2.3.3\
/files?redirect=true&path=resnet34_peoplenet_int8.onnx" -O examples/resnet34_peoplenet_int8.onnx
```

   **Note:** This setup only needs to be performed once.

2. Launch the Holoscan Sensor Bridge demo container.

```
xhost +
sh docker/demo.sh
```

   **Note:** After running the *demo.sh* script on the terminal, the demo container starts automatically, and the command prompt switches to the container environment.

3. Run the TAO PeopleNet example to display live person detection in the video, as shown in Figure 6.60.

```
python3 examples/linux_tao_peoplenet.py
```

   **Note:** On the first run, the script may take at least 10 minutes to complete the setup.

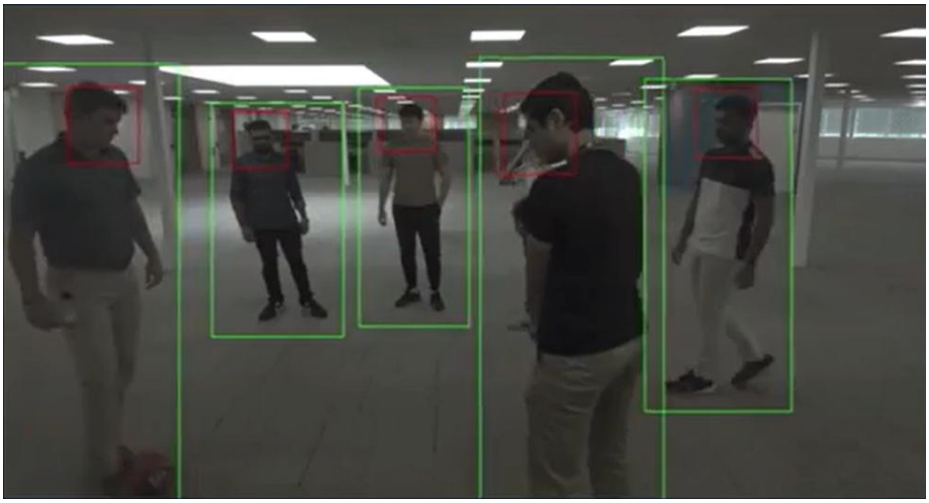


**Figure 6.60. Live TAO PeopleNet Output**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# 7. Resource Utilization

The section shows the resource utilization of the reference design for the CertusPro-NX FPGA device using the Synplify Pro® tool in the Lattice Radiant software. Refer to the Compiling the Reference Design section for configuration details. Note that changes in design attributes may affect resource utilization, and values may vary over time due to feature updates, bug fixes, or other modifications.

Note: By default, RUNTIME_MIPI configuration method is selected and both ISP and MJPEG hardware modules are implemented in the reference design.

## 7.1. Resource Utilization Using the CertusPro-NX Versa Board

This section summarizes FPGA resource utilization across configurations that use 1GbE, 2.5GbE, and 10GbE Ethernet links. It highlights how the primary design blocks map onto the available logic resources of the CertusPro-NX (LFCPNX-100-9LFG672I) device, providing a clear view of implementation efficiency across different Ethernet options.

### 7.1.1. Resource Utilization with Ethernet 1GbE

Table 7.1 shows one camera sensor interface configured with four MIPI lanes at 1,500 Mbps paired with a single-port 1GbE Ethernet host interface.

**Table 7.1. Logic Consumption for Each Instance Available in the Reference Design**

| Instance name | LUT4 Logic | LUT4 Distributed RAM | LUT4 Ripple Logic | PFU Registers | DSP MULT | EBR |
|---|---|---|---|---|---|---|
| multicam_top (camera sensor interface) | 6,031 | 600 | 1,980 | 7,695 | 36 | 60 |
| Hololink_top (Hololink IP) | 13,229 | 468 | 3,060 | 18,696 | 4 | 59 |
| Ethernet_inst (Ethernet 1GbE- host interface) | 2,258 | 0 | 800 | 3,285 | 0 | 4 |
| Clk_rst | 2 | 0 | 0 | 38 | 0 | 0 |
| Miscellaneous | 179 | 48 | 118 | 441 | 0 | 0 |
| Total | 21,699 | 1,116 | 5,958 | 30,155 | 40 | 123 |

**Note:** Other submodules are categorized as miscellaneous in this table.

Table 7.2 compares the reference design logic consumption with the available resources of the CertusPro-NX (LFCPNX-100-9LFG672I) device, based on the configuration described in Table 7.1.

**Table 7.2. Comparison Between the Reference Design Utilization and the CertusPro-NX Resources Available**

| | LUT | PFU registers | DSP Mult | EBR |
|---|---|---|---|---|
| LFCPNX-100 device logic resource | 79,872 | 79,872 | 156 | 208 |
| Reference design logic resource consumed | 28,773 | 30,155 | 40 | 123 |
| Reference design logic resource consumed (%) | 36.02% | 37.75% | 25.64% | 59.13% |

From the place-and-route report, the device SLICE utilization summary after the final SLICE packing is 58% used (23,319 out of 39,936 slices). The $F_{MAX}$ used is 125 MHz.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### 7.1.2. Resource Utilization with Ethernet 2.5GbE

Table 7.3 shows one camera sensor interface configured with four MIPI lanes at 1,500 Mbps paired with a single-port 2.5GbE Ethernet host interface.

**Table 7.3. Logic Consumption for Each Instance Available in the Reference Design**

| Instance name | LUT4 Logic | LUT4 Distributed RAM | LUT4 Ripple Logic | PFU Registers | DSP MULT | EBR |
|---|---|---|---|---|---|---|
| multicam_top<br>(camera sensor interface) | 6,175 | 600 | 1,948 | 7,981 | 36 | 54 |
| Hololink_top<br>(Hololink IP) | 15,452 | 468 | 2,958 | 21,257 | 4 | 49 |
| Ethernet_inst<br>(Ethernet 2.5GbE- host interface) | 4,237 | 24 | 484 | 5,196 | 0 | 62 |
| Clk_rst | 4 | 0 | 0 | 38 | 0 | 0 |
| Miscellaneous | 294 | 48 | 176 | 538 | 0 | 1 |
| Total | 26,162 | 1,140 | 5,566 | 35,010 | 40 | 166 |

**Note:** Other submodules are categorized as miscellaneous in this table.

Table 7.4 compares the logic consumption of the reference design with the available resources of the CertusPro-NX (LFCPNX-100-9LFG672I) device, based on the configuration described in Table 7.3.

**Table 7.4. Comparison Between the Reference Design Utilization and the CertusPro-NX Resources Available**

| | LUT | PFU registers | DSP Mult | EBR |
|---|---|---|---|---|
| LFCPNX-100 device logic resource | 79,872 | 79,872 | 156 | 208 |
| Reference design logic resource consumed | 32,868 | 35,010 | 40 | 166 |
| Reference design logic resource consumed (%) | 41.15% | 43.83% | 25.64% | 79.81% |

From the place-and-route report, the device SLICE utilization summary after the final SLICE packing is 75% used (29,839 out of 39,936 slices). The $F_{MAX}$ used is 156.25 MHz.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### 7.1.3. Resource Utilization with Ethernet 10GbE

Table 7.5 shows one camera sensor interface configured with four MIPI lanes at 1,500 Mbps and a single-port 10GbE Ethernet host interface.

**Table 7.5. Logic Consumption for Each Instance Available in the Reference Design**

| Instance name | LUT4 Logic | LUT4 Distributed RAM | LUT4 Ripple Logic | PFU Registers | DSP MULT | EBR |
|---|---|---|---|---|---|---|
| multicam_top<br>(camera sensor interface) | 6,180 | 600 | 1,948 | 7,971 | 36 | 54 |
| Hololink_top<br>(Hololink IP) | 15,651 | 468 | 2,958 | 20,864 | 4 | 49 |
| Ethernet_inst<br>(Ethernet 10GbE- host interface) | 3,863 | 0 | 110 | 2,976 | 0 | 4 |
| Clk_rst | 4 | 0 | 0 | 38 | 0 | 0 |
| Miscellaneous | 166 | 48 | 118 | 380 | 0 | 0 |
| Total | 25,864 | 1,116 | 5,134 | 32,229 | 40 | 107 |

**Note:** Other submodules are categorized as miscellaneous in this table.

Table 7.6 compares the reference design logic consumption with the available resources of the CertusPro-NX (LFCPNX-100-9LFG672I) device, based on the configuration described in Table 7.5.

**Table 7.6. Comparison Between the Reference Design Utilization and the CertusPro-NX Resources Available**

| | LUT | PFU registers | DSP Mult | EBR |
|---|---|---|---|---|
| LFCPNX-100 device logic resource | 79,872 | 79,872 | 156 | 208 |
| Reference design logic resource Consumed | 32,114 | 32,229 | 40 | 107 |
| Reference design logic resource consumed (%) | 40.21% | 40.35% | 25.64% | 51.44% |

From the place-and-route report, the device SLICE utilization summary after the final SLICE packing is 62% used (24,789 out of 39,936 slices). The FMAX used is 156.25 MHz.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

## 7.2. Resource Utilization Using the CertusPro-NX Sensor-to-Ethernet Bridge Board

### 7.2.1. Resource Utilization with Ethernet 10GbE

Table 7.7 reports resource utilization only for the CertusPro-NX FPGA. The CrossLink™-NX (LIFCL-17) is treated as external; its resource usage is not included here. Table 7.8 compares the logic usage of the reference design with the available resources of the CertusPro-NX (LFCPNX-100-9LFG672C), using the configuration in Table 7.7.

**Table 7.7. Logic Consumption for Each Instance Available in the Reference Design**

| Instance name | LUT4 Logic | LUT4 Distributed RAM | LUT4 Ripple Logic | PFU Registers | DSP MULT | EBR |
|---|---|---|---|---|---|---|
| cam_sensor_rcvr[0]<br>(LVDS DDR RX interface) | 1,001 | 0 | 58 | 765 | 0 | 3 |
| cam_sensor_rcvr[1]<br>(LVDS DDR RX interface) | 1,008 | 0 | 58 | 764 | 0 | 3 |
| Hololink_top<br>(Hololink IP) | 30,411 | 3,048 | 4,516 | 39,041 | 4 | 109 |
| Ethernet_inst<br>(Ethernet 10GbE- host interface X2) | 7,709 | 0 | 220 | 5,952 | 0 | 8 |
| Clk_rst | 3 | 0 | 0 | 38 | 0 | 0 |
| Miscellaneous | 1,434 | 48 | 446 | 2,222 | 0 | 2 |
| Total | 41,566 | 3,096 | 5,298 | 48,782 | 4 | 125 |

**Note:** Other submodules are categorized as miscellaneous in this table

**Table 7.8. Comparison Between the Reference Design Utilization and the CertusPro-NX Resources Available**

| | LUT | PFU registers | DSP Mult | EBR |
|---|---|---|---|---|
| LFCPNX-100 device logic resource | 79,872 | 79,872 | 156 | 208 |
| Reference design logic resource consumed | 49,960 | 48,782 | 4 | 125 |
| Reference design logic resource consumed (%) | 62.55% | 61.08% | 2.56% | 60.10% |

From the place-and-route report, the device SLICE utilization summary after the final SLICE packing is 92% used (36,551 out of 39,936 slices). The $F_{MAX}$ used is 156.25 MHz.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# 8. Debug Methodology

This section outlines the recommended steps for debugging the camera video streaming setup in the Holoscan Sensor Bridge reference design. By following this procedure, you can verify system functionality and identify potential issues during integration and testing.

## 8.1. CertusPro-NX Versa Board

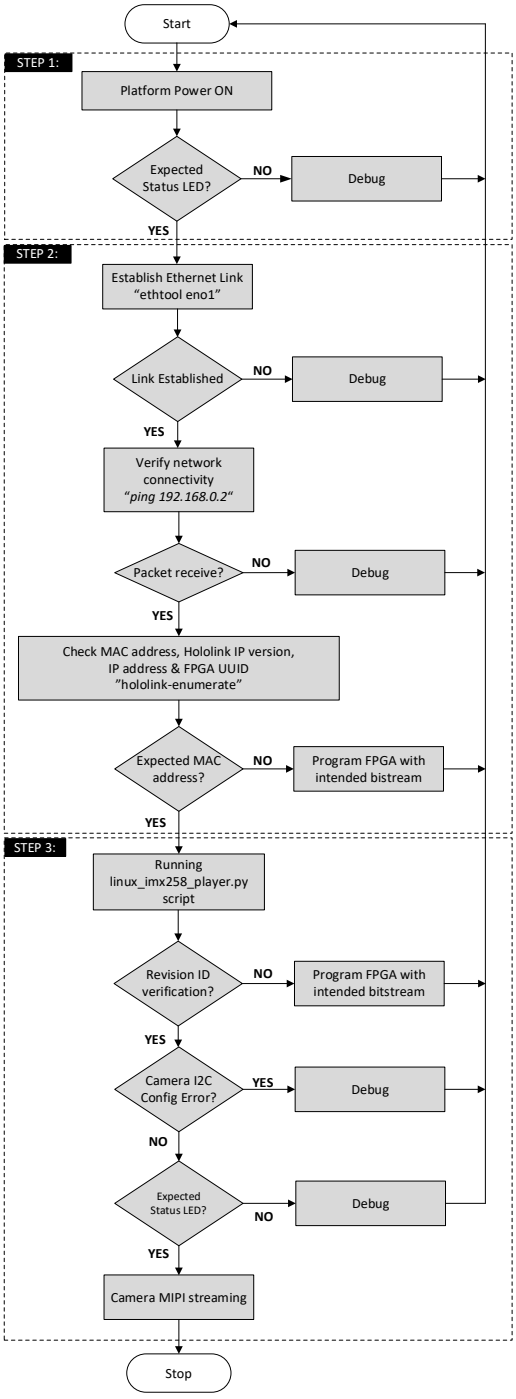



**Figure 8.1. Debug Methodology Flow Chart For RD**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### 8.1.1. Powering Up the Board

1. Power cycle the board after programming.
   After programming the FPGA firmware into the external SPI flash, fully power down the platform, then power it back up. This power cycle ensures the FPGA correctly loads and configures the new firmware.
2. Verify LED status.
   After powering up, check the status LEDs on the board. Refer to Figure 8.2 for LED locations. These LEDs indicate the board operational status. Table 8.1 explains each LED behavior and meaning.
   **Note**: This step assumes the HSB Docker container is already set up on the host system.

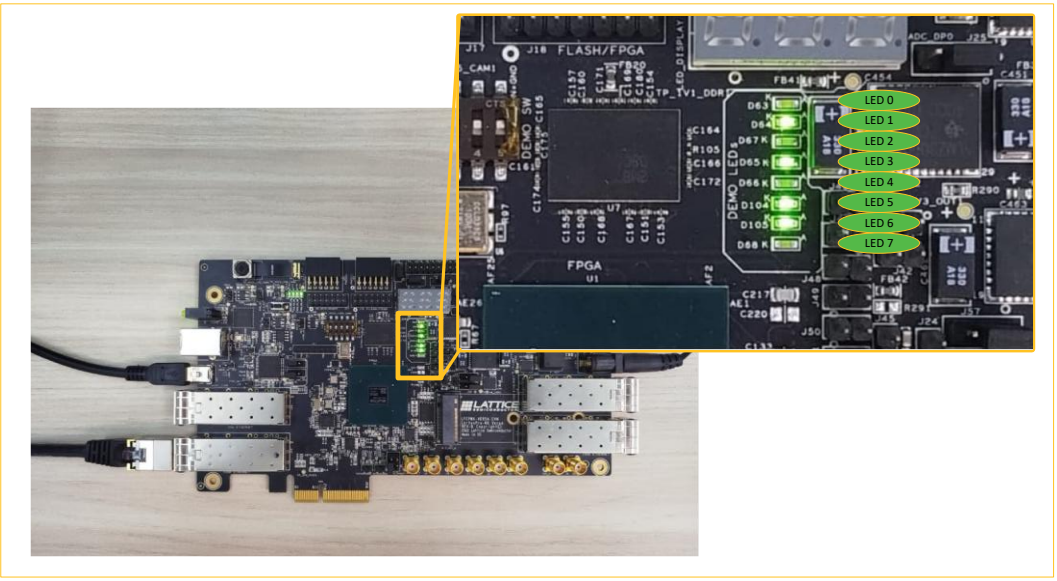



**Figure 8.2. Position of the LED on CertusPro-NX Versa Board**

Table 8.1 shows the LED status upon power-up.

**Table 8.1. LED Status upon Board Power-up**

| LED | Connected Signal | Board Power UP |
|---|---|---|
| LED0 | pcs_clk_heart_beat | LED toggles |
| LED1 | hif_clk_heart_beat | LED toggles |
| LED2 | byte_clk_heart_beat | OFF |
| LED3 | plat_rst_n (platform reset) | ON |
| LED4 | sif_rst | OFF |
| LED5 | sw_sen_rst | ON |
| LED6 | cam_rst_n | ON |
| LED7 | hif_rst | OFF |

#### 8.1.1.1. Troubleshooting Common Issues

- **Issue A: No status LEDs turn on at power-up**.
  - Workaround 1: Check that the CertusPro-NX Versa board power supply is connected and the 12 V switch is turned on.
  - Workaround 2: Power cycle the board using 12 V switch.
  - Workaround 3: Reprogram the FPGA to confirm the correct bitstream is used.
  - Workaround 4: If the issue persists, submit a technical support case through Lattice Technical Support.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

- **Issue B: LED is not toggling**.
  Possible cause: Ethernet IP configuration issue.
  - Workaround 1: Regenerate the Ethernet IP with the correct settings. Recompile and reprogram the FPGA device with the newly compiled FPGA bitstream.
  - Workaround 2: If the issue persists, submit a technical support case through Lattice Technical Support.
- **Issue C: Reset status LED behave unexpectedly**.
  Possible cause: Custom changes to HDL source code or software patches.
  - Review your modifications to the reference design HDL source code and any applied software patches.

### 8.1.2. Verifying the Ethernet Connection

1. Check the Ethernet link status.
   On the host terminal, run *ethtool end0* command. This verifies whether the Ethernet link is established, and confirms that the link speed is as expected. If successful, the terminal will display the link status, as shown in Figure 8.3.

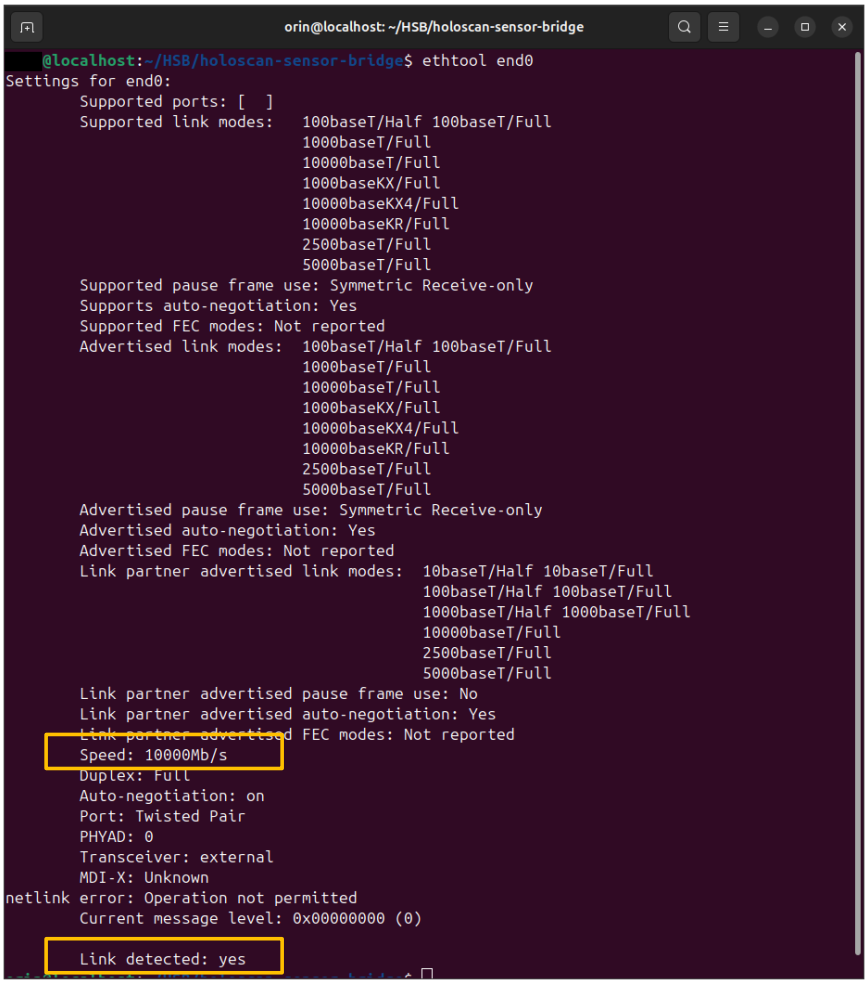



**Figure 8.3. Ethernet Link Establishment Status**

2. Test the Ethernet connectivity.
   Run *ping 192.168.0.2* command on the host terminal. A successful response confirms that the network connection is active. You should see messages indicating that packets are being received. Refer to Figure 8.4 for an example of successful ping response from the evaluation board.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

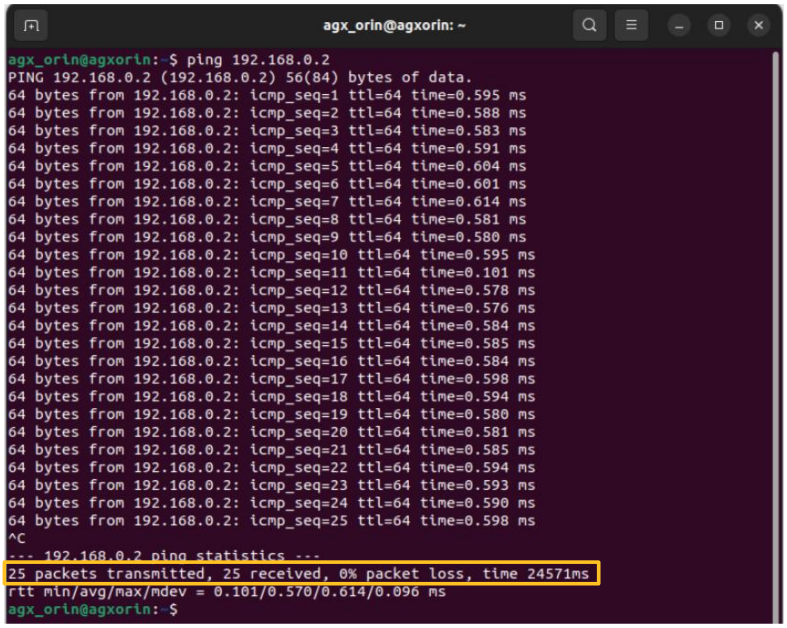



**Figure 8.4. Data Packet Received by Host**

3. Verify the device information.
   On the host HSB container prompt, run *hololink-enumerate* command. This command displays the MAC address, Hololink IP version, IP address, and FPGA UUID. See Figure 8.5 for a sample output.

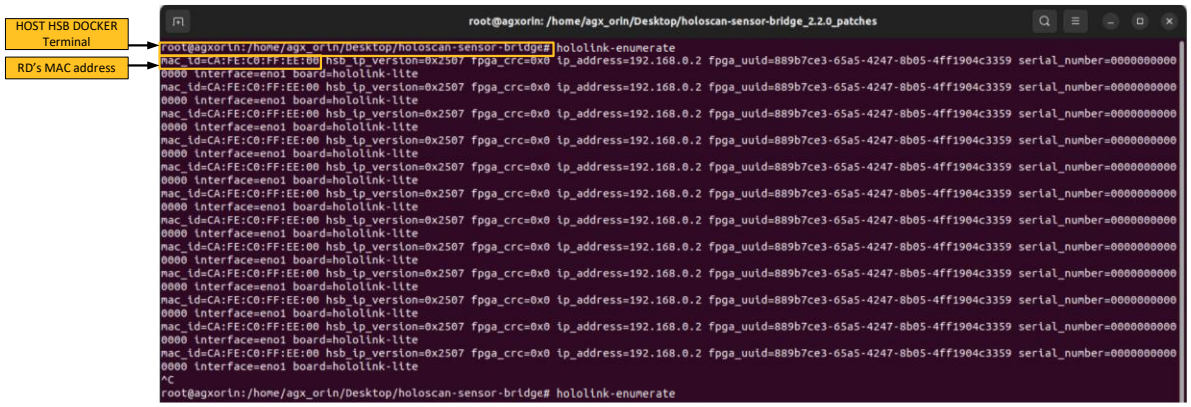



**Figure 8.5. Sample Output**

#### 8.1.2.1. Troubleshooting Connection Issues

If you encounter issues during steps 1 or 2, try the following:

- Workaround 1: Ensure the SFP module is securely locked into the SFP cage.
- Workaround 2: Confirm that the LAN cable is properly connected to both the SFP module and the host LAN port. If the connection is correct, the LAN indicator light should be **ON**, as shown in Figure 8.6.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

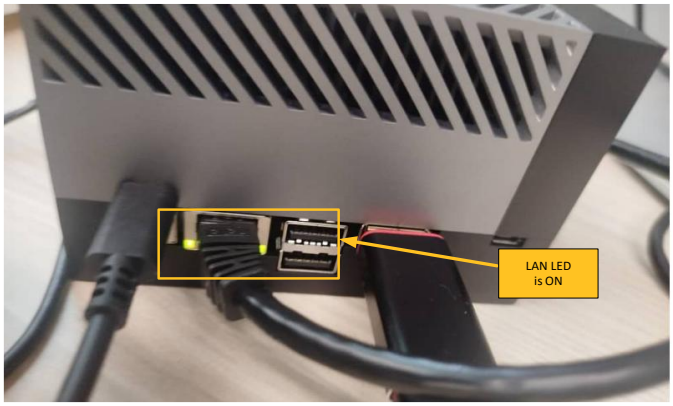



**Figure 8.6. LAN Light Indication on Host**

## 8.1.3. Running the Streaming Script (linux_imx258_player.py)

### 8.1.3.1. Preparing the Streaming Script

Run the script. Follow the steps in the Camera Video Streaming Test section to run the *linux_imx258_player.py* command on the host HSB docker container prompt.

### 8.1.3.2. Monitoring the Output of the Streaming Script

Once the script is running, observe the following:

1. Verify the Revision ID.
   Check the log for the Revision ID to confirm the FPGA firmware version. Figure 8.7 shows the Revision ID version from the host HSB docker container prompt, which represents the bitstream version for the CertusPro-NX with 10GbE Ethernet.

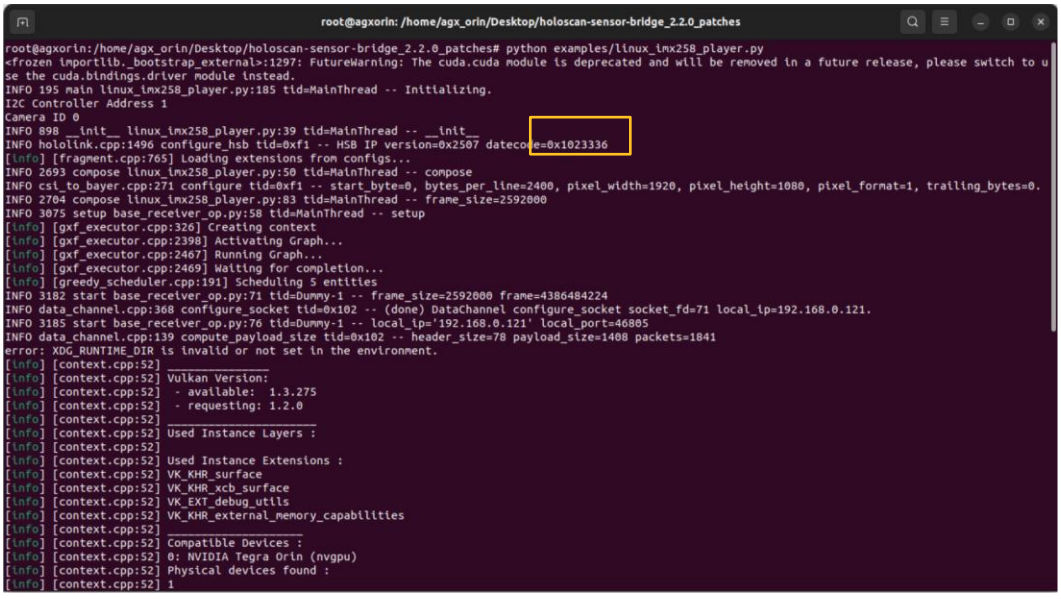



**Figure 8.7. Revision ID Version**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

2. Check the camera for I2C configuration errors.
   Look for any I2C configuration errors in the log, when the *linux_imx258_player.py* script is executed. Figure 8.8 shows an example of a failed camera I2C configuration.

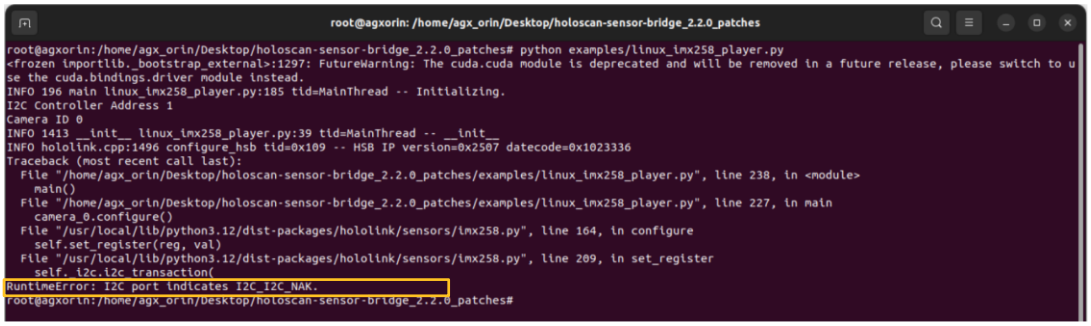



**Figure 8.8. Camera I2C Configuration Fail Log**

3. Observe the status LED.
   Refer to Table 8.2 for expected LED behavior. The byte clock heartbeat LED should toggle, indicating that the camera sensor is transmitting MIPI data.

**Table 8.2. Expected Status LED Behavior when running linux_imx258_player.py.script**

| LED | Connected Signal | MIPI camera streaming |
|---|---|---|
| LED0 | pcs_clk_heart_beat | LED toggles |
| LED1 | hif_clk_heart_beat | LED toggles |
| LED2 | byte_clk_heart_beat | LED toggles |
| LED3 | plat_rst_n (platform reset) | ON |
| LED4 | sif_rst | OFF |
| LED5 | sw_sen_rst | ON |
| LED6 | cam_rst_n | ON |

4. Confirm MIPI Camera Streaming
   If no errors are detected, the MIPI camera stream should display successfully on the host monitor.

#### 8.1.3.3. Troubleshooting Common Issues

- **Issue A: Revision ID version mismatch.**
  - Workaround: Reprogram the HSB board with the correct FPGA bitstream version.
- **Issue B: Camera I2C Configuration error.**
  - Workaround 1: Ensure the *IMX258* camera sensor is properly connected.
  - Workaround 2: If you are using your own reference design HDL code, verify that the camera reset signal is not asserted.
  - Workaround 3: If you are using your own software patch, confirm the I2C configuration port is correctly targeted.
  - Workaround 4: If you are using your own software patch, ensure the camera sensor register settings are correct.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

## 8.2. CertusPro-NX Sensor-to-Ethernet Bridge Board

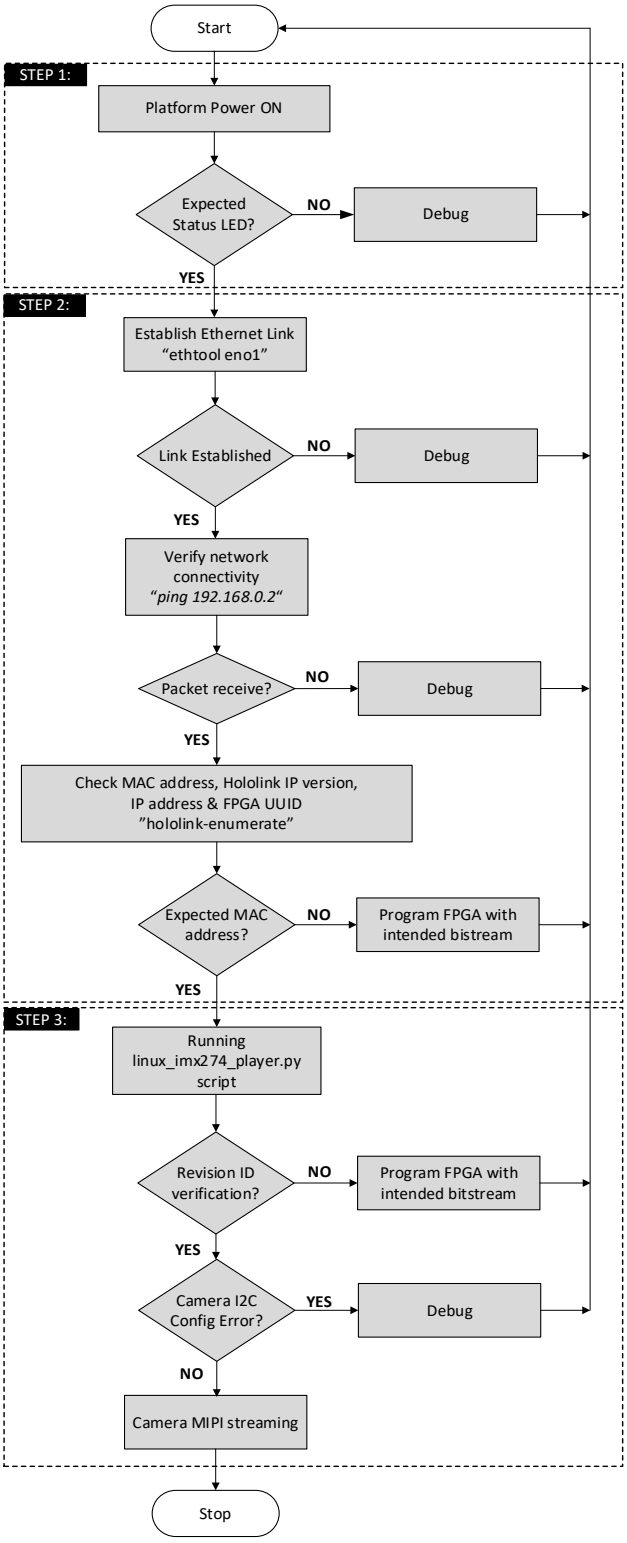



**Figure 8.9. Debug Methodology Flow Chart For RD**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### 8.2.1. Powering Up the Board

1. Power cycle the board after programming the FPGA bitstream into the external SPI flash.
   After programming the FPGA firmware into the external SPI flash, fully power down the platform, then power it back up. This power cycle ensures the FPGA correctly loads and configures the new firmware.
2. Verify LED status.
   After powering up, check the status LEDs on the board. Refer to Figure 8.10 for LED locations. These LEDs indicate the board operational status. Table 8.3 explains each LED behavior and meaning.
   **Note**: This step assumes the HSB Docker container is already set up on the host system.

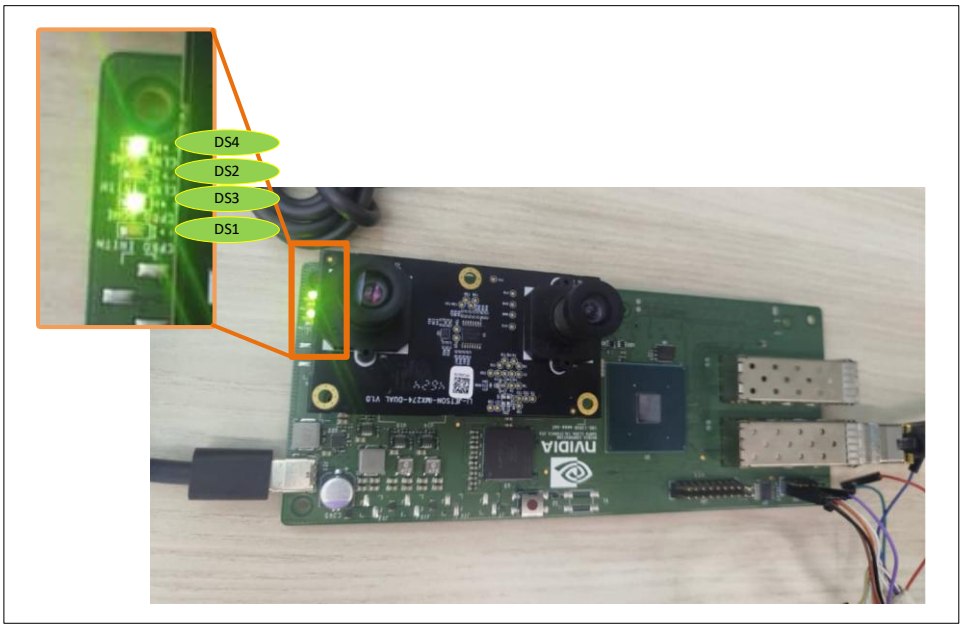



**Figure 8.10. Position of the Status LED on CertusPro-NX Sensor-to-Ethernet Bridge Board**

Table 8.3 shows the LED status upon power-up.

**Table 8.3. LED Status upon Board Power-up**

| LED | Connected Signal | Board Power UP |
|---|---|---|
| DS1 | CPRO INITN | OFF |
| DS2 | CLNK INITN | OFF |
| DS3 | CPRO DONE | ON |
| DS4 | CLNK DONE | ON |

Use the INITN status pin to monitor errors during configuration. After configuration completes successfully, DONE asserts and the FPGA enters User Mode.

#### 8.2.1.1. Troubleshooting Common Issues

- **The status LEDs do not behave as described in** Table 8.3.
  - Workaround 1: Check that the CertusPro-NX Sensor-to-Ethernet Bridge Board power supply is connected and is turned on.
  - Workaround 2: Power cycle the board.
  - Workaround 3: Reprogram the FPGA to confirm the correct bitstream is used.
  - Workaround 4: If the issue persists, submit a technical support case through Lattice Technical Support.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### 8.2.2. Verifying the Ethernet Connection

1. Check the Ethernet link status.
   On the host terminal, run *ethtool end0* command. This verifies whether the Ethernet link is established, and confirms that the link speed is as expected. If successful, the terminal will display the link status, as shown in Figure 8.11.

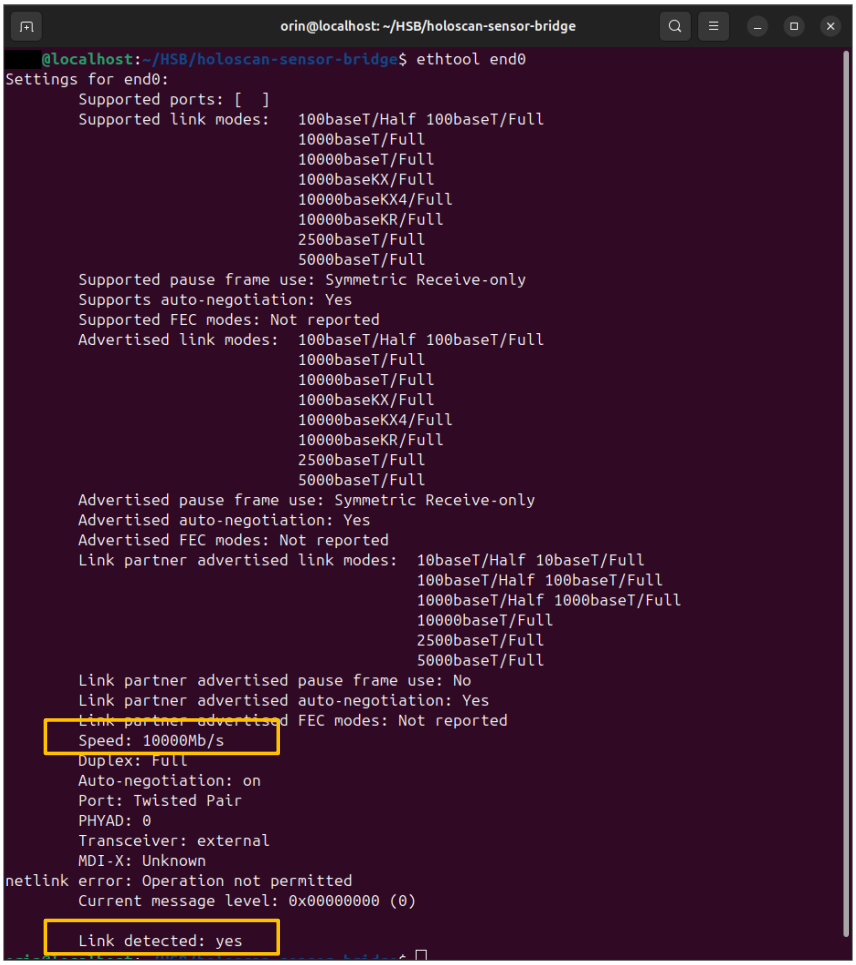



**Figure 8.11. Ethernet Link Establishment Status**

2. Test the Ethernet connectivity.
   Run *ping 192.168.0.2* command on the host terminal. A successful response confirms that the network connection is active. You should see messages indicating that packets are being received. Refer to Figure 8.12 for an example of successful ping response from the evaluation board.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

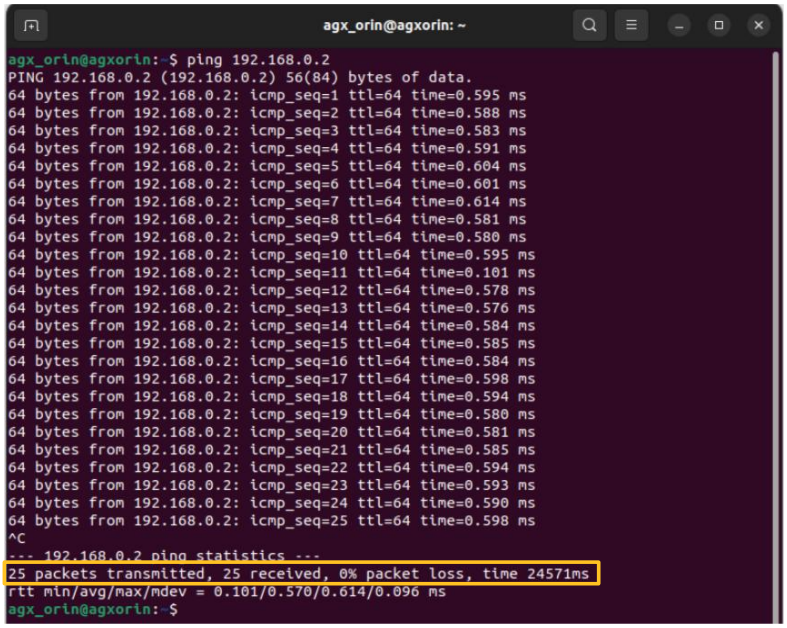



**Figure 8.12. Data Packet Received by Host**

3. Verify the device information.
   On the host HSB container prompt, run *hololink-enumerate* command. This command displays the MAC address, Hololink IP version, IP address, and FPGA UUID. See Figure 8.13 for a sample output.

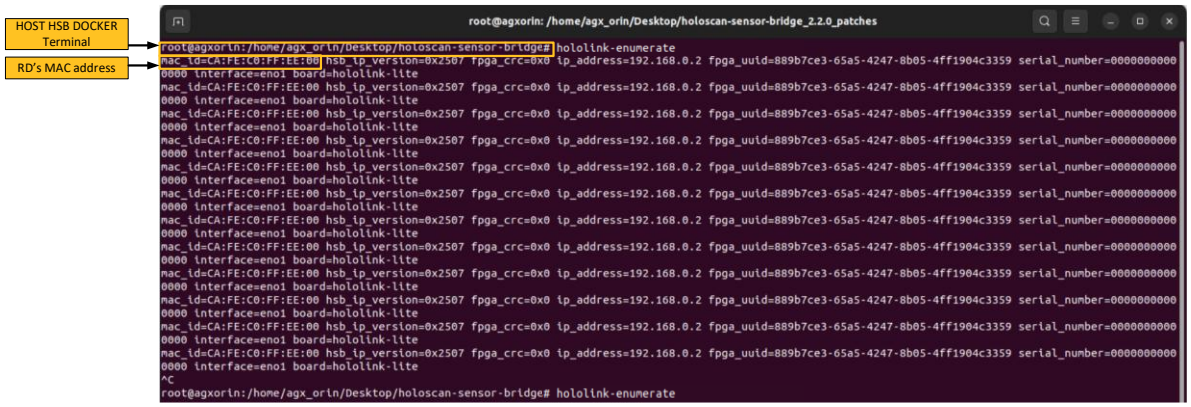



**Figure 8.13. Sample Output**

#### 8.2.2.1. Troubleshooting Connection Issues

If you encounter issues during steps 1 or 2, try the following:

- Workaround 1: Ensure the SFP module is securely locked into the SFP cage.
- Workaround 2: Confirm that the LAN cable is properly connected to both the SFP module and the host LAN port. If the connection is correct, the LAN indicator light should be on, as shown in Figure 8.14.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

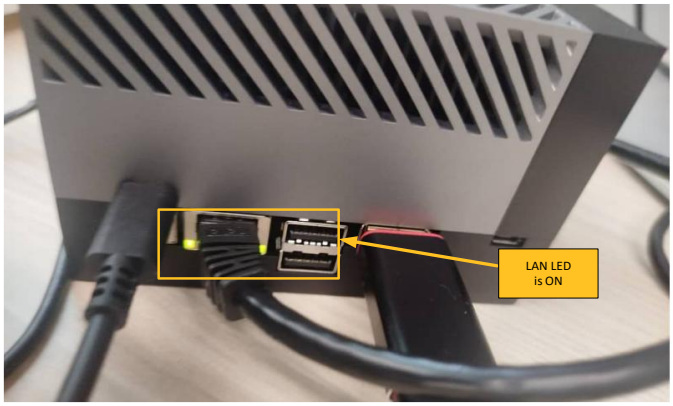



**Figure 8.14. LAN Light Indication on Host**

## 8.2.3. Running the Streaming Script (linux_imx274_player.py)

### 8.2.3.1. Preparing the Streaming Script

Run the script. Follow the steps in the Camera Video Streaming Test section to run the *linux_imx274_player.py* command on the host HSB docker container prompt.

### 8.2.3.2. Monitoring the Output of the Streaming Script

Once the script is running, observe the following:

1. Verify the Revision ID.
   Check the log for the Revision ID to confirm the FPGA firmware version. Figure 8.15 shows the Revision ID version from the host HSB docker container prompt, which represents the bitstream version for the CertusPro-NX with 10GbE Ethernet.

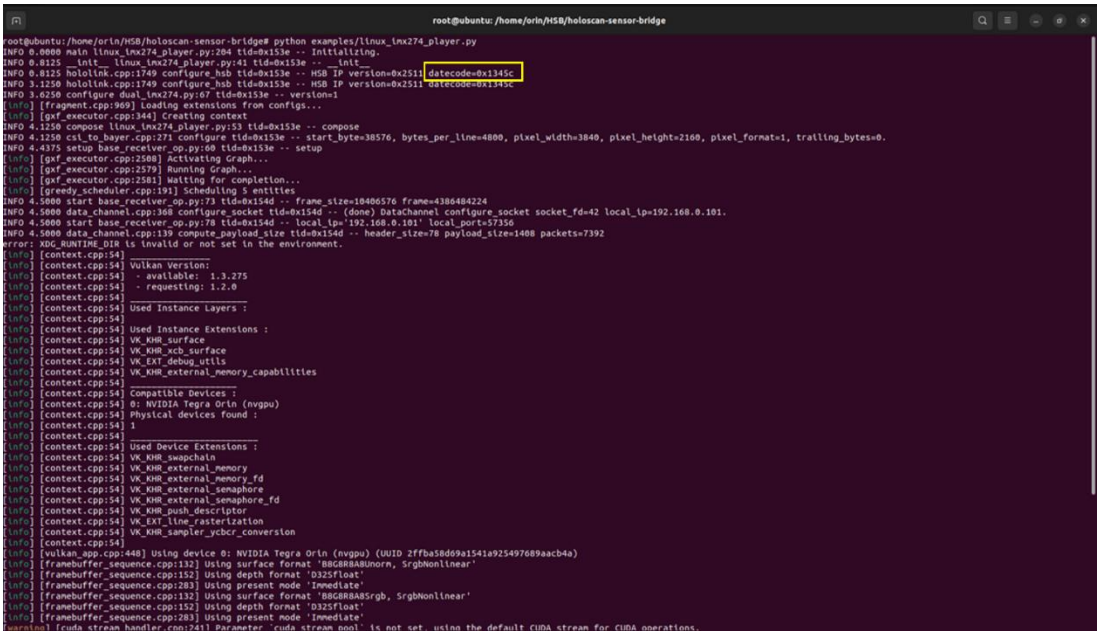



**Figure 8.15. Revision ID Version**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

2. Check the camera for I2C configuration errors.
   Look for any I2C configuration errors in the log, when the *linux_imx274_player.py* script is executed. Figure 8.16 shows an example of a failed camera I2C configuration.

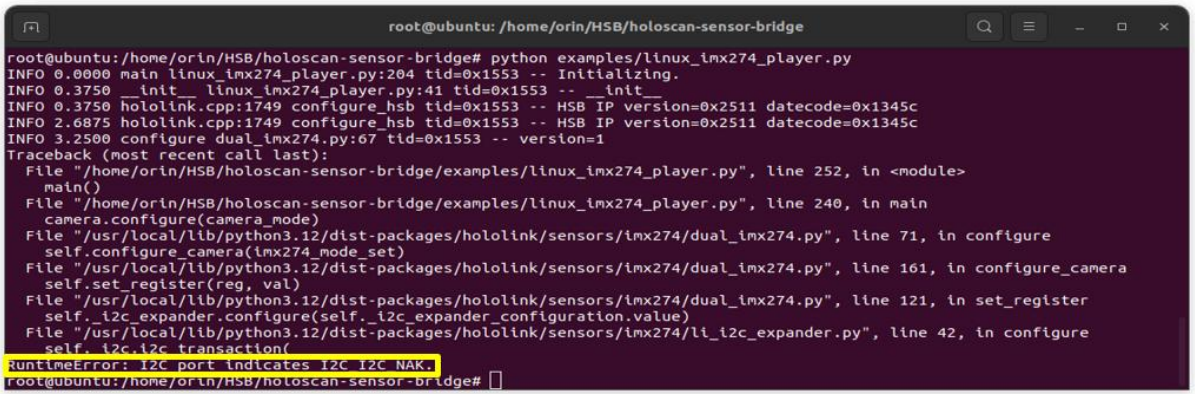



**Figure 8.16. Camera I2C Configuration Fail Log**

3. Confirm MIPI Camera Streaming.
   If no errors are detected, the MIPI camera stream should display successfully on the host monitor.

#### 8.2.3.3. Troubleshooting Common Issues

- **Issue A: Revision ID version mismatch.**
  - Workaround: Reprogram the HSB board with the correct FPGA bitstream version.
- **Issue B: Camera I2C Configuration Error.**
  - Workaround 1: Ensure the *IMX274* camera sensor is properly connected.
  - Workaround 2: If you are using your own reference design HDL code, verify that the camera reset signal is not asserted.
  - Workaround 3: If you are using your own software patch, confirm the I2C configuration port is correctly targeted.
  - Workaround 4: If you are using your own software patch, ensure the camera sensor register settings are correct.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# 9. Reference Design Configuration Tool

When customizing the reference design, two files sit at every configuration:

- HOLOLINK_def.svh: SystemVerilog header defining the HDL parameters and DEFINE switches for camera count, MIPI configuration, Ethernet speed, ISP/MJPEG enables, and board identity.
- Board .pdc file: Platform Design Constraint file governing pin and placement constraints.

Configuring the HDL Parameter and DEFINE switch section and Configuring the Platform Design Constraint section walk through how to configure these files manually, editing raw text, matching array sizes, and ensuring both files remain consistent with each other. While this approach offers full visibility into every setting, manual errors become more likely, particularly as the number of cameras or Ethernet speed variants grows.

The Reference Design configuration tool addresses exactly that requirement. Rather than opening these files in a text editor, the tool presents the same parameters through a structured graphical interface; enforces valid combinations automatically, and writes both files together in a single *Save* operation, producing the same correctly configured output with far less risk of a manual mistake.

**Note:** The tool does not launch the Lattice Radiant software or generate bitstreams. After saving the configuration, compile the reference design in the Lattice Radiant software. For details, refer to the Compiling the Reference Design section.

## 9.1. Installation

**Requirements:** Windows 10 or 11 (64-bit). Read and write access to the RD project folder.

- Download the Reference Design source zip from the HSB Sensor Interface customization website and extract it fully.
- If the *misc* folder is a separate zip, extract it as well.
- Open the *misc* folder and double-click *rd_config.exe*. No Python or Lattice Radiant software is required.

**Caution:** Note that the Reference Design configuration tool is designed to work exclusively with its corresponding Reference Design release. Compatibility with other Reference Design versions is not supported.

### 9.1.1. Windows Defender SmartScreen Warning

When launching *rd_config.exe* for the first time, Windows Defender SmartScreen may display a security prompt. This is expected behavior and does not indicate a problem with the tool.

**Steps to Bypass Windows Defender SmartScreen and Launch the Tool**

1. On the initial SmartScreen prompt, click **More info**.

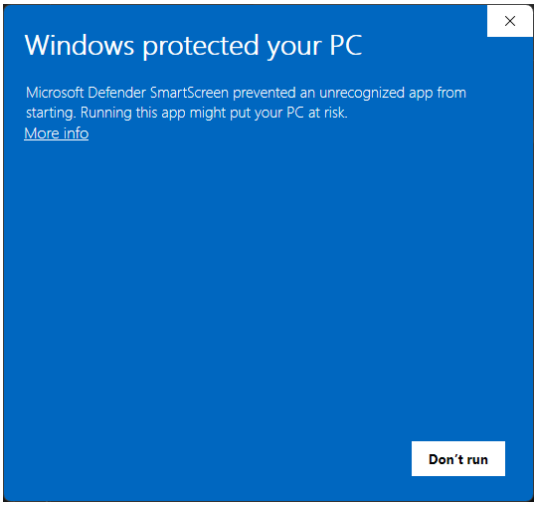



**Figure 9.1. Initial SmartScreen Prompt**

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

2. After clicking **More info**, click **Run anyway** to launch the tool.

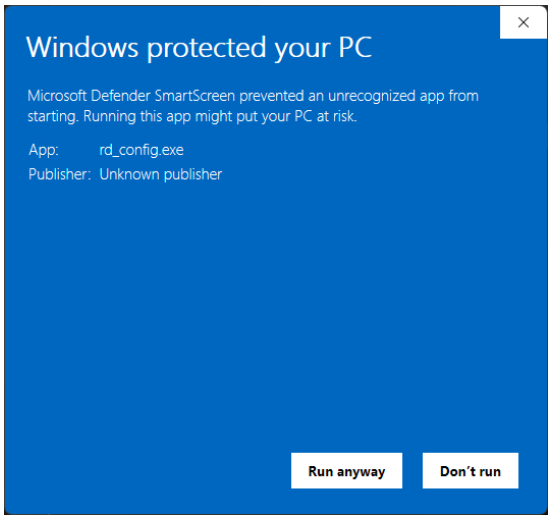



**Figure 9.2. Click Run Anyway**

**Note:** This prompt typically appears only on the first launch. If your organization policy prevents running unsigned executables, contact your IT administrator.

## 9.2. Key Concepts

### 9.2.1. Preview vs. Save Behavior

Every change in the options pane updates the *HOLOLINK_def.svh* and *board .pdc* preview in the memory only. Project files on the disk are unchanged until you click *Save*. Closing without saving loses all changes.

### 9.2.2. MIPI Configuration Modes

The *Enable Runtime MIPI* checkbox selects between two MIPI modes:

**Table 9.1. Runtime MIPI Configuration Modes**

| Mode | Enable Runtime MIPI | MIPI lane/rate |
|---|---|---|
| Runtime MIPI | On | Fixed, 4 lanes/1,500 Mbps per camera |
| Preset MIPI | Off | Selectable per camera |

Disabling **Runtime MIPI** switches the interface to **Preset MIPI** mode, where the MIPI lane count and lane rate can be independently configured for each camera.

**Table 9.2. Supported MIPI Configuration Modes by Lane Rate**

| Rate (Mbps) | Allowed lanes |
|---|---|
| 720 | 1, 2, 4 |
| 960 | 2, 4 |
| 1,440 | 4 |
| 1,500 | 4 |

When the camera count increases, each newly added camera row defaults to 1 lane/720 Mbps. Adjust the lane and rate for each new camera as required.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### 9.2.3. ISP and MJPEG Dependency

MJPEG requires ISP to be on. Turning ISP off clears MJPEG automatically.

## 9.3. User Interface

### 9.3.1. Welcome Screen

When the *rd_config.exe* launches, the welcome screen appears with a large dashed-border drop zone. Drag a project folder or *HOLOLINK_def.svh* into the zone to begin.

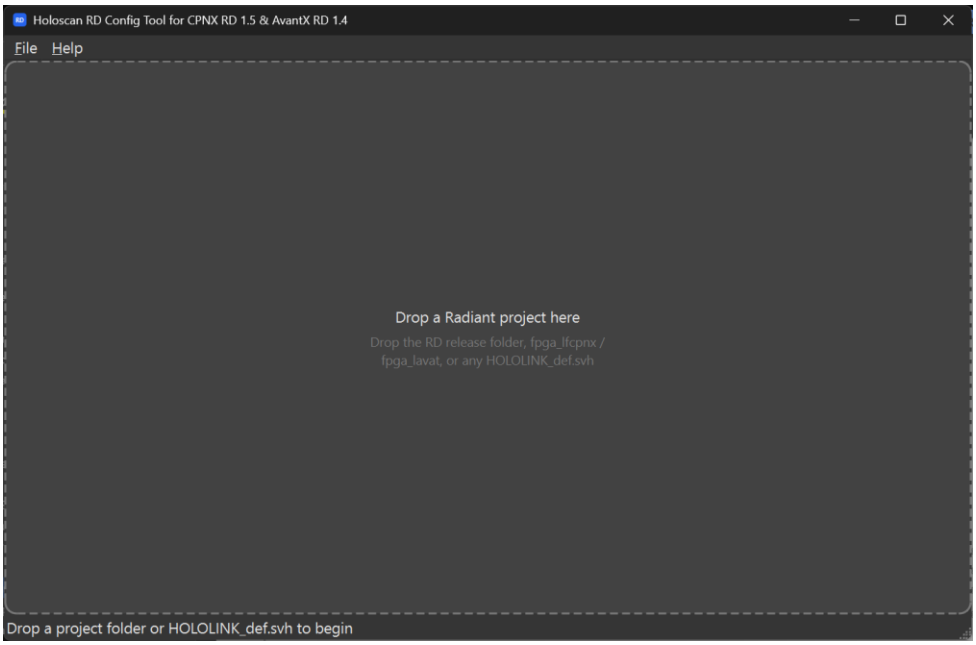



**Figure 9.3. Welcome Screen**

The drop zone accepts both files and folders. Table 9.3 describes the supported file and folder drop locations.

**Table 9.3. Supported File and Folder Drop Location**

| Drop target | Notes |
|---|---|
| RD project folder | Top-level folder extracted from the RD source zip |
| <RD project folder>/fpga_lfcpnx | CertusPro-NX FPGA project root |
| <RD project-folder>/fpga_lfcpnx/radiant/src/rtl/hsb_cpnx/top/… / HOLOLINK_def.svh | Skips most selection dialogs |

### 9.3.2. Main Workspace

The *options pane* on the left contains all configurable parameters. The *Editor tab* on the right display a live in-memory preview of *HOLOLINK_def.svh* and the *board .pdc* file. The *Save* button is at the top-left of the toolbar.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

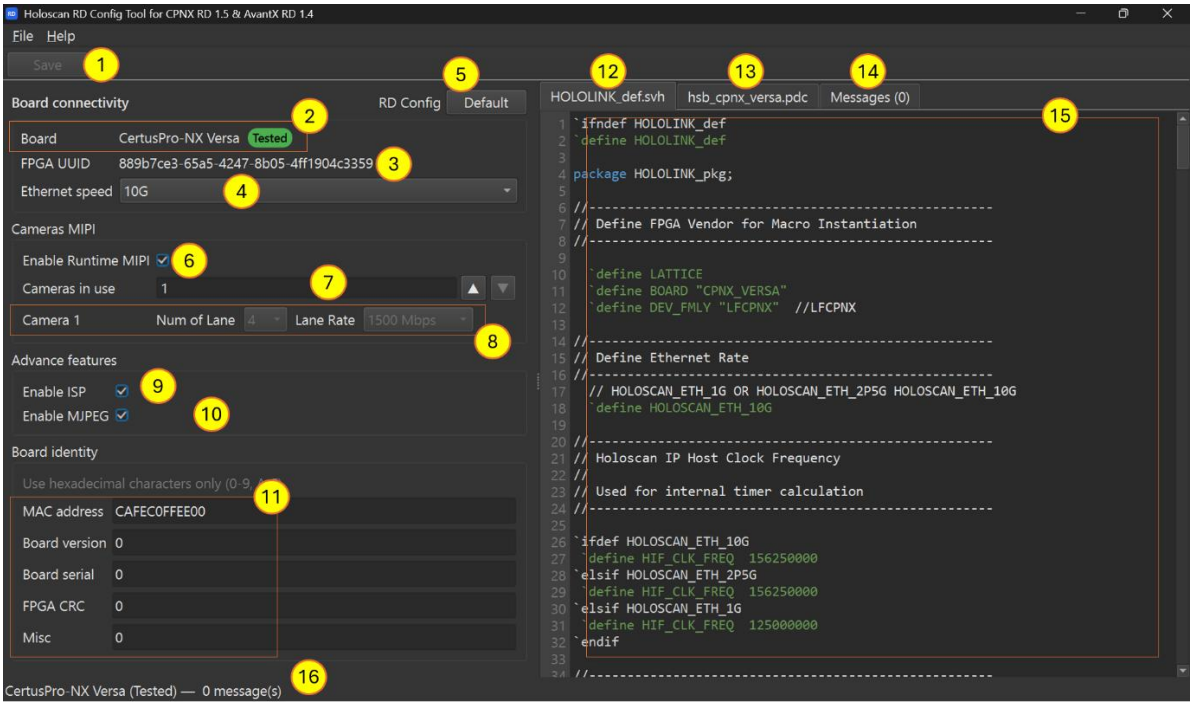



**Figure 9.4. Main Workspace – CertusPro-NX Versa Board**

The main workspace is described in Table 9.4 through Table 9.8.

**Table 9.4. Board connectivity**

| | Label | Description |
|---|---|---|
| 1 | Save | Commits all in-memory changes to disk. Active only when there are unsaved changes. |
| 2 | Board name and badge | Detected board name, read-only. Displays a Tested (green) or Custom (yellow) badge depending on camera count. |
| 3 | FPGA UUID | Universally Unique Identifier (UUID) read from HOLOLINK_def.svh, read-only. The UUID differs across Ethernet speed variants (for example, 1G, 2.5G, 10G). |
| 4 | Ethernet speed | Set this first, it drives the Ethernet macro, PDC Ethernet field, and all downstream camera and MIPI limits. |
| 5 | RD config / default | Resets all option pane settings to the pack defaults. |

**Table 9.5. Cameras MIPI**

| | Label | Description |
|---|---|---|
| 6 | Enable Runtime MIPI | When checked, MIPI is fixed at 4 lanes/1,500 Mbps and lane/rate controls are grayed out. When unchecked (Preset MIPI), lane and rate are set per camera manually. |
| 7 | Cameras in use | Spin control for the number of active camera interfaces. |
| 8 | Lane and rate | Per-camera MIPI lane count and data rate.<br>Valid lanes: 1, 2, 4;<br>Valid rates: 720, 960, 1,440, 1,500 Mbps (Preset MIPI only). |

**Table 9.6. Advanced features**

| | Label | Description |
|---|---|---|
| 9 | Enable ISP | Enables the Image Signal Processing block. |
| 10 | Enable MJPEG | Enables Motion JPEG compression. Requires enable ISP to be on. |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

**Table 9.7. Board identity**

| | Label | Description |
|---|---|---|
| 11 | Board identity fields | MAC address, board version, board serial, FPGA CRC, and misc.<br>Identify the physical board to the host system. Each field must contain valid hex characters only. |

**Note:** All fields accept hexadecimal characters only (0–9, A–F). The tool writes these values to *HOLOLINK_def.svh* on Save.

**Table 9.8. Editor tabs and status**

| | Label | Description |
|---|---|---|
| 12 | SVH preview tab | Live in-memory preview of HOLOLINK_def.svh. Read-only in the tool. |
| 13 | PDC tab | Live preview of the board Platform Design Constraint file. |
| 14 | Messages tab | Displays validation results. Yellow or red indicator appears when attention is needed. |
| 15 | Live SVH preview | Shows the exact text the tool will write on Save, and updates it in real time. |
| 16 | Status bar | Shows the current board name, badge, and message count. |

### 9.3.3. Menu Bar and Keyboard Shortcuts

The menu bar provides two menus: **File** and **Help**, covering all project and application actions.

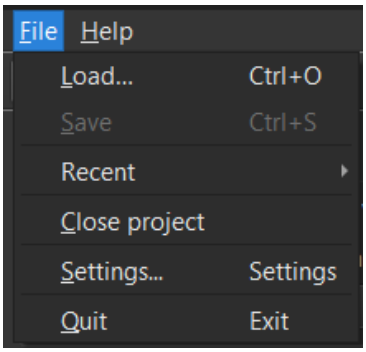



**Figure 9.5. File Menu**

**Table 9.9. Menu Bar Actions and Keyboard Shortcuts**

| Action | Menu path | Shortcut |
|---|---|---|
| Open project | File → Load… | Ctrl+O |
| Save | File → Save | Ctrl+S |
| Application settings | File → Settings… | Ctrl+, |
| Close project | File → Close project | — |
| Recent projects | File → Recent | — |
| Version/supported RD | Help → About | — |
| Quit | File → Quit | — |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

#### 9.3.3.1. Application Settings

Open through **File → Settings…**. These preferences control how the tool looks and behaves. They do not affect any project files.

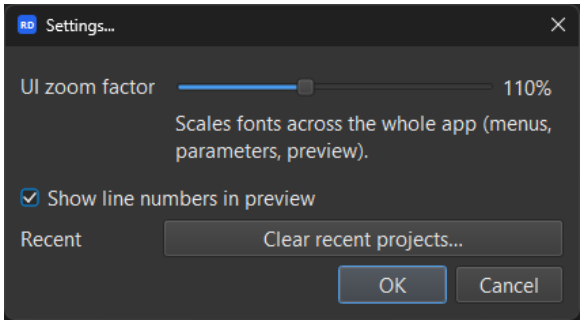



**Figure 9.6. Application Settings Dialog Box**

**Table 9.10. Application Settings Option and Description**

| Setting | Description |
|---|---|
| UI zoom factor | Scales fonts across the whole application. Range: 50–200%. |
| Show line numbers in preview | Toggles line number display in the SVH and PDC editor tabs. |
| Recent → Clear recent projects… | Removes all entries from the File → Recent list. |

#### 9.3.3.2. About

Open through **Help → About**. Displays the tool name, supported reference design platforms and versions, and a brief description.

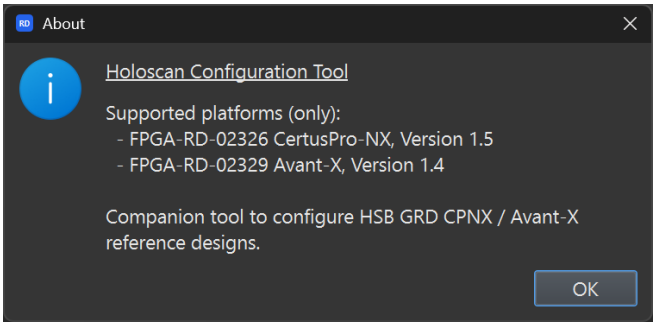



**Figure 9.7. About Dialog Box**

**Table 9.11. About Dialog Content**

| Field | Description |
|---|---|
| Supported platforms | Lists the reference design families and versions the tool support, example, FPGA-RD-02326 CertusPro-NX v1.5 and FPGA-RD-02329 Avant-X v1.4. |
| Description | Confirms the tool is a companion for configuring HSB GRD CertusPro-NX / Avant-X reference designs. |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

## 9.4. Project Loading

Drag an RD project folder or HOLOLINK_def.svh onto the welcome screen (see Table 9.3 for supported locations) or choose **File → Load…** (Ctrl+O). If a selection dialog appears, pick the correct option.

**Table 9.12. Project Loading Behavior for Multiple Board Variants**

| Dialog | When it appears | What to pick |
|---|---|---|
| Board variant | Multiple board-top folders detected | CertusPro-NX Versa board or CertusPro-NX Sensor-to-Ethernet Bridge board |

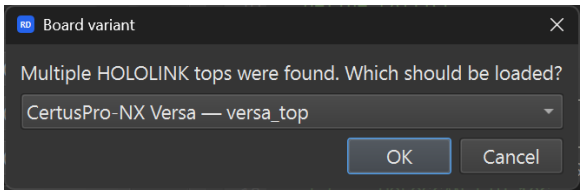



**Figure 9.8. Board Variant Dialog Box**

After the project loads, verify the board name and the FPGA UUID in the options pane. Review the message tab for any warnings or errors that occurred during loading.

**Tip:** Use **File → Recent** to reopen a recently used project without browsing to the project folder again.

## 9.5. Configuration Workflow

Follow this order for all Versa boards. The Sensor-to-Ethernet Bridge board is view-only, no configuration steps apply.

1. Set the Ethernet speed first.

   This parameter drives generation of the Ethernet macro in HOLOLINK_def.svh, populates the corresponding PDC Ethernet field and FPGA UUID, and serves as the source configuration for all downstream camera and MIPI settings.
2. Choose MIPI mode.
   - Runtime MIPI on: The MIPI is fixed at 4 lanes/1,500 Mbps. Set the camera count only.
   - Runtime MIPI off (Preset MIPI): Set the camera count, then tune lane and rate per camera.
3. Set the cameras in use.

   Increase the count with the spin control. The badge changes from *Tested* (one camera) to *Custom* (two or more). Set the camera count before tuning the MIPI rows.
4. Tune per-camera MIPI (Preset MIPI only)

   Select *Lane* and *Rate* from the dropdowns for each camera. The user-interface filters invalid combinations automatically.
5. Enable *ISP/MJPEG* if needed.

   Toggle as required. After each change, recheck *Messages*.
6. Check *Messages*, refer to Table 9.13, then click *save (Ctrl+S)*. After saving, open the project in the Lattice Radiant software to build the bitstream.

**Table 9.13. Message Indicators and Recommended Action**

| Messages state | Action |
|---|---|
| Clear (no messages) | Proceed to Save |
| Yellow warning | Review; Save is allowed with confirmation |
| Red error | Tool prompts Save anyway? Confirm to save with errors, or cancel and fix first |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

## 9.6. Board-Specific Guidance

The Reference Design configuration tool supports multiple target platforms. Available configuration options and constraints vary by target platform. Refer to Table 9.14 for details.

**Table 9.14. Board-Specific Features and Constraints**

| Feature | CertusPro-NX Versa board | Sensor-to-Ethernet Bridge board |
|---|---|---|
| Configuration mode | Full configuration | View only, no configuration steps apply |
| Ethernet speed | 1G, 2.5G, or 10G | Fixed at 10G |
| MIPI mode | Runtime MIPI or Preset MIPI | Fixed, all controls are read-only |
| Camera count | Multi-camera supported | Fixed two-camera LVDS bridge configuration |
| ISP/MJPEG | Both supported | Not applicable |
| Save | Enabled | Disabled |
| Board badge | Tested (one camera) / Custom (two or more cameras) | Tested |

## 9.7. Save Behavior and Backups

Clicking **Save** writes three files listed in Table 9.15.

**Table 9.15. Configuration Data Saved to Each File**

| File | What changes |
|---|---|
| HOLOLINK_def.svh | Ethernet macro, MIPI arrays, ISP/MJPEG, board identity |
| board .pdc | Camera count, per-camera MIPI, Ethernet speed key |
| .rd_config.json | Board/Reference Design identification metadata |

Closing without saving. Click **File** → **Close project** (or quitting) prompts **Save/Discard/Cancel**. Discard reverts to the on-disk files.

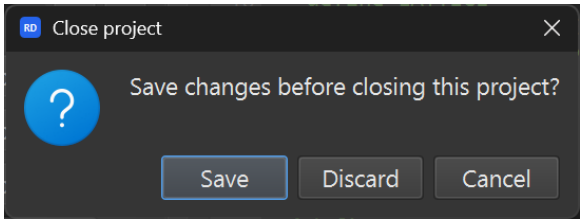



**Figure 9.9. Close Project Dialog Box**

## 9.8. Validation Messages

**Table 9.16. Message Status Indicator Definitions**

| Indicator | Meaning |
|---|---|
| Clear | No issues |
| Yellow | Warning, for example *Untested Design* |
| Red | Error, invalid setting |

When an Untested Design warning appears in three places: the triangle beside the custom badge, the yellow icon on the message tab, and the warning text in the message panel. Click the *message* to jump to the related line in SVH or PDC.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

### 9.8.1. Diagnostic Codes

**Table 9.17. Diagnostic Codes, Causes, and Recommended Fixes**

| Code | Severity | Cause | Fix |
|---|---|---|---|
| untested_design | Warning | This configuration has not been fully validated. High-bandwidth interface configurations may encounter timing closure issues, which can result in longer-than-expected FPGA build times. | Reduce the camera count to tested or accept and save |
| preset_config_invalid | Error | Camera/MIPI/ISP/MJPEG combo not in preset matrix | Change one parameter at a time until the combination is valid |
| runtime_mipi_invalid | Error | Too many cameras for Runtime MIPI tier | Reduce the camera count or switch to Preset MIPI |
| eth_macro_invalid | Error | Ethernet macro invalid for this board | Re-select a supported Ethernet speed |
| eth_svh_pdc_mismatch | Error | SVH and PDC disagree on Ethernet speed | Re-select the Ethernet speed to synchronize both files |
| pdc_eth_invalid | Error | PDC Ethernet value not allowed | Re-select a supported Ethernet speed to update the PDC value |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# Appendix A: Known Issues

1. When using certain types of 1000BASE-T RJ-45 SFP module, the network link may fail to establish, and no link LED activity is observed during this condition.
2. When using certain types of 10GBASE-T SFP+ modules, the network link may fail to establish, and no link LED activity is observed during this condition. Additionally, the Ethernet MAC may reset unexpectedly. This issue has been identified and is scheduled to be addressed in a future software release.
3. When building with the Ethernet 2.5GbE configuration, you will observe a longer place-and-route duration compared to the Ethernet 10GbE and Ethernet 1GbE configurations.
4. When building a 2.5G Ethernet design, Lattice Radiant software reports a warning that no set_clock_uncertainty constraint is defined for a PLL clock output. This is a known issue and will be addressed in a future release.
5. Timing Analysis report a false Minimum Pulse Width (MPW) timing violation for the CertusPro-NX MPCS module due to a timing model issue in the current Lattice Radiant software release. This reported violation does not reflect actual silicon behavior, has no impact on hardware functionality, and can be safely ignored. This issue affects only the CertusPro-NX 10G interface configurations. A fix is planned for future release.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# References


- CertusPro-NX Versa Board User Guide (FPGA-EB-02053)
- CertusPro-NX Sensor to Ethernet Bridge Board User Guide (FPGA-EB-02069)
- CSI-2/DSI D-PHY Receiver IP Core - User Guide (FPGA-IPUG-02081)
- Multi-Boot User Guide for Nexus Platform (FPGA-TN-02145)
- Lattice Radiant Software User Guide
- CertusPro-NX Customizable HSB Sensor Interface Kit and Reference Design – Source Code
- CertusPro-NX Versa Board web page
- CertusPro-NX Sensor-to-Ethernet Bridge Board web page
- CertusPro-NX Versa Holoscan Kit web page
- CSI-2/DSI D-PHY Receiver IP Core web page
- CSI-2/DSI D-PHY Transmitter IP Core web page
- NVIDIA Holoscan - NVIDIA Docs NVDIA Holoscan Docs Hub
- NVIDIA Jetson Developer Kits web page
- Lattice Radiant FPGA design software
- Lattice Insights for Lattice Semiconductor training courses and learning plans
- Information Technology — Digital Compression and Coding of Continuous-tone Still Images — Requirements and Guidelines, ITU-T Recommendation T.81, Sept. 1992. [Online]. Available: w3.org


© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# Technical Support Assistance

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

For frequently asked questions, refer to the Lattice Answer Database at www.latticesemi.com/Support/AnswerDatabase.

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

# Revision History

**Revision 1.7, September 2026**

| Section | Change Summary |
|---|---|
| Implementing the Reference Design | Added (\) line continuation to Linux commands in this section to mitigate PDF conversion spacing and line-break issues. |

**Revision 1.6, September 2026**

| Section | Change Summary |
|---|---|
| All | • Updated the Lattice Radiant software version to 2026.1.0.37.0 globally.<br>• Minor editorial fixes. |
| Introduction | • Updated the following in Table 1.1. Summary of CertusPro-NX Versa Board with NVIDIA Jetson AGX Orin, Table 1.2. Summary of CertusPro-NX Versa Board with NVIDIA Jetson AGX Thor, and Table 1.3. Summary of CertusPro-NX Sensor-to-Ethernet Bridge Board with NVIDIA Jetson AGX Orin.<br>  ◦ Updated the Hololink IP version to *2606.*<br>  ◦ Updated the JetPack version to *7.2*.<br>  ◦ Updated the NVIDIA Holoscan SDK version to *4.4.0*.<br>  ◦ Updated the NVIDIA Jetson Linux version to *39.2*.<br>  ◦ Updated the HSB SDK version to *2.7.0*.<br>• Updated the *CSI-2/DSI D-PHY Rx IP version to 2.2.0* in Table 1.1. Summary of CertusPro-NX Versa Board with NVIDIA Jetson AGX Orin and Table 1.2. Summary of CertusPro-NX Versa Board with NVIDIA Jetson AGX Thor.<br>• Updated the 2.5GbE Ethernet (MAC + PHY IP) Core to *v1.2.0* in Table 1.1. Summary of CertusPro-NX Versa Board with NVIDIA Jetson AGX Orin.<br>• Added an *M.2 2280 NVMe SSD 512GB* host component in Table 1.1. Summary of CertusPro-NX Versa Board with NVIDIA Jetson AGX Orin and in Table 1.3. Summary of CertusPro-NX Sensor-to-Ethernet Bridge Board with NVIDIA Jetson AGX Orin. |
| Directory Structure and File Overview | Updated the Hololink IP version referenced in the radiant/src/rtl/hololink/folder description to *2606.* |
| Customizing the Reference Design | • Updated the runtime MIPI CSI-2 D-PHY configuration command from *python* to *pyhton3.*<br>• Updated the Hololink IP version to *2606.*<br>• Updated the HSB SDK version to *2.7.0*.<br>• Added Jetpack 7.2 permission requirement under the ISP Configuration Tuning with Live Image Streaming section. |
| Implementing the Reference Design | • Added an *M.2 2280 NVMe SSD 512GB* host component to the hardware requirement lists for both the CertusPro-NX Versa Board and the CertusPro-NX Sensor-to-Ethernet Bridge Board.<br>• Updated the following for both Jetson AGX Orin and Jetson AGX Thor in the Software Requirements section.<br>  ◦ Updated the JetPack version to *7.2.*<br>  ◦ Updated the NVIDIA Jetson Linux version to *39.2.*<br>  ◦ Updated the NVIDIA Holoscan SDK version to *4.4.0.*<br>  ◦ Updated the HSB SDK version to *2.7.0.*<br>• Replaced the JetPack 6.2.1 confirmation screenshot in the AGX Orin flashing walkthrough with a new screenshot and step showing selection of JetPack 7.2 (Figure 6.9. Development Environment Selection).<br>• Added a new screenshot and step documenting an administrative-permission prompt in the AGX Orin flashing walkthrough (Figure 6.11. Administrative Right).<br>• Updated the Ubuntu version to 24.04 in Figure 6.16. Hardware Setup for Flash Jetson AGX Thor.<br>• Added a new screenshot and step documenting the target-hardware selection pop-up in the AGX Thor flashing walkthrough (Figure 6.19. Device Selection). |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

| Section | Change Summary |
|---|---|
| | • Updated Ubuntu version to *24.04*.<br>• Updated the steps in Setting-up the Jetson AGX Orin Developer Kit.<br>• Updated the Configuring the AGX Orin Developer Kit section.<br>  • Added a step to install the nano text editor.<br>  • Updated the *network interface reference* from *EN0=eno1* to *EN0=end0* and the *isolated-core reference* from *core3* to *CPU2*.<br>  • Updated the board IP from *192.168.0.101* to *192.168.0.2*.<br>• Updated HSB source code version to *2.7*.0.<br>• Updated the steps in Flashing the Jetson AGX Thor using the SDK Manager section.<br>• Updated *step 1*, *step 2,* and added *step 8* in Configuring the AGX Thor Developer Kit section.<br>• Added *2.5GbE coverage* to the Remote Firmware Update through Ethernet procedure (under CertusPro-NX Versa Board) and its accompanying *note 1*, and updated the *bitstream description* to cover *10GbE*, *2.5GbE*, and *1GbE*.<br>• Removed the *–runtime-mipi 1* argument and updated the interpreter from *python* to *python3* in the Testing the Camera with ISP and MJPEG Features and in Testing the Camera Body Pose sections under the CertusPro-NX Versa Board.<br>• Updated the interpreter from *python* to *python3* in Testing the Camera Body Pose section under the CertusPro-NX Sensor-to-Ethernet Bridge Board.<br>• Updated the steps in *Testing the Camera TAO PeopleNet* section for both *CertusPro-NX Versa Board* and *CertusPro-NX Sensor-to-Ethernet Bridge Board* to use the python3 interpreter.<br>• Added a path to holoscan-sensor-bridge before the *wget* command under the following:<br>  • Testing the Camera TAO PeopleNet (CertusPro-NX Versa Board)<br>  • Testing the Camera TAO PeopleNet (CertusPro-NX Sensor-to-Ethernet Board) |
| Resource Utilization | • Updated the resource utilization values in the following sections.<br>  • Resource Utilization with Ethernet 1GbE<br>  • Resource Utilization with Ethernet 2.5GbE<br>  • Resource Utilization with Ethernet 10GbE using the CertusPro-NX Versa Board<br>  • Resource Utilization with Ethernet 10GbE using the CertusPro-NX Sensor-to-Ethernet Bridge Board |
| Debug Methodology | Updated command from *eno1* to *end0.* |
| Reference Design Configuration Tool | • Updated the Preset MIPI mode description to clarify that changing the Ethernet speed updates the camera count limit and valid lane/rate options while preserving existing per-camera MIPI lane and rate values under the MIPI Configuration Mode section.<br>• Updated the Configuration Workflow section.<br>  • Updated the Ethernet speed configuration description to clarify that the parameter generates the Ethernet macro in HOLOLINK_def.svh, populates the corresponding PDC Ethernet field and FPGA UUID, and serves as the source configuration for all downstream camera and MIPI settings.<br>  • Removed the note stating that enabling ISP and MJPEG reduces the maximum allowed camera count.<br>• Replaced *Congested* with *Untested* in Table 9.16. Message Status Indicator Definitions and in the entire Validation Messages section.<br>• Changed *congested_design* to *untested_design* and updated description of the entire row under the untested_design code in Table 9.17. Diagnostic Codes, Causes, and Recommended Fixes. |
| Appendix A. Known Issues | • Added a note that when building a 2.5G Ethernet design, Lattice Radiant software reported a warning that no set_clock_uncertainty constraint was defined for a PLL clock output.<br>• Added a note that the reported MPCS MPW timing violation affects only CertusPro-NX 10G interface configurations, is caused by a timing model issue, and has no impact on hardware functionality. |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

**Revision 1.5, July 2026**

| Section | Change Summary |
|---|---|
| RD Config Tool | Added this section. |

**Revision 1.4, June 2026**

| Section | Change Summary |
|---|---|
| All | • Updated the Lattice Radiant software version from *2025.2.0.48.0* to *2025.2.1.321.0* globally.<br>• Minor editorial fixes. |
| Abbreviations in This Document | Added the following abbreviations: CUDA, DCT, FP4, ISP, JSON, LSCC, MJPEG, QSFP28, RTL, TFLOPS, and TOPS. |
| Introduction | • Added 2.5GbE as a supported host Ethernet link speed.<br>• Added the following in Table 1.1. Summary of CertusPro-NX Versa Board with NVIDIA Jetson AGX Orin.<br>  • 2.5GbE Ethernet (MAC + PHY IP) Core v1.1.0.<br>  • ISP Lattice Semiconductor Reference Design IP v1.1.1.<br>  • Launch LSCC ISP Builder Linux Orin v1.1.0.<br>  • MJPEG Lattice Semiconductor Reference Design IP.<br>  • Ethernet 2.5GBase-T SFP-2.5G-T-1 module.<br>• Added the following in Table 1.2. Summary of CertusPro-NX Versa Board with NVIDIA Jetson AGX Thor.<br>  • ISP Lattice Semiconductor Reference Design IP v1.1.1 and removed the Tri-Speed Ethernet IP.<br>  • Launch LSCC ISP Builder Linux Thor v1.1.0.<br>  • MJPEG Lattice Semiconductor Reference Design IP.<br>• Added 2.5GbE to the IMX258 camera streaming connection options in the Quick-Start Guides.<br>• Updated the Features section to add ISP-based image enhancement and MJPEG image decoding to the Sensor Interface and to add 2.5GbE to the supported Host Interface link speeds. |
| Directory Structure and File Overview | Updated Table 2.1. Folder Structure Description to restructure the project directory paths and to add the radiant/ip/ISP/ folder (ISP and MJPEG Lattice Semiconductor Reference design IP) and the Ethernet 2.5GbE IP. |
| Functional Description | • Updated the NVIDIA Hololink documentation reference from the *v2.5.0 release* to the respective *HSB SDK release* in the Overview section.<br>• Expanded the Sensor Interface section to add ISP processing and MJPEG compression, increasing the MIPI CSI-2 sensor interface from two to five primary components and adding MJPEG bitstream and ISP YUV422 output formats.<br>• Updated Figure 3.6. Overview block diagram of MIPI CSI-2 Sensor Interface to show the new ISP-to-AXIS, MJPEG-to-AXIS, byte-to-pixel, and dual clock-domain CDC paths with MUX-based output selection.<br>• Added Table 3.1. Supported Bayer processing and converter steps, Table 3.2. Supported RGB processing and converter steps, and Table 3.3. Supported YUV444 processing and converter steps.<br>• Added a 2.5GbE Ethernet support subsection covering the MAC IP Core, PHY IP Core, and 2.5GBase-T SFP transceiver module to the Host Interface through Ethernet Link section.<br>• Added a Multi Camera Frame Synchronization section describing the vsync_gen module, IEEE 1588 PTP-based synchronization, and the programmable PERIOD, PULSE EXPOSURE, DELAY START, and START VALUE parameters.<br>• Updated the Clocking and Reset Scheme section to add the following:<br>  • 2.5GbE coverage (a new Clock Domain Distribution for Ethernet 2.5GbE table).<br>  • A ptp_clk (100.45 MHz) row to the 1GbE and 10GbE clock-domain tables and to expand the reset distribution to cover 1GbE, 2.5GbE, and 10GbE.<br>• Updated Figure 3.8. Reset and Clock Domain Block Diagram for Ethernet 1GbE to add a |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

| Section | Change Summary |
|---|---|
| | VSYNC block and PTP clock path.<br>• Added Figure 3.9. Reset and Clock Domain Block Diagram for Ethernet 2.5GbE.<br>• Updated Figure 3.10. Reset and Clock Domain Block Diagram for Ethernet 10GbE to add a VSYNC block and PTP clock path. |
| Customizing the Reference Design | • Updated Figure 4.1. Steps Flow to Customize and Verify the Reference Design to add HSB board selection, run-time MIPI configuration, and ISP or MJPEG enablement steps.<br>• Added the following in the Generating the IP section.<br>  • Byte-to-Pixel Conversion IP section<br>  • ISP Reference Design IP section<br>  • MJPEG Reference Design IP section<br>  • 2.5GbE in the Ethernet IP Generation section<br>• Added Figure 4.2. Steps to Generate ISP Lattice Semiconductor Reference Design IP through Figure 4.7. Generate Configuration window illustrating the ISP IP generation and tuning workflow.<br>• Added Figure 4.10. Ethernet 2.5GbE (MAC + PHY) IP Generation.<br>• Added Table 4.3. Generated ISP IP Folder Structure Description.<br>• Added a new Enabling ISP or MJPEG Feature at Runtime section describing the ISP_EN and MJPEG_EN parameters.<br>• Added Table 4.6. ISP or MJPEG Parameters Enablement in Hardware.<br>• Updated Table 4.8. Configuration for the reference design to fit into the CertusPro-NX Sensor-to-Ethernet Bridge Board to a 1,500 Mbps MIPI lane rate and added 2.5GbE to the Ethernet link speed options. |
| Compiling the Reference Design | Updated Table 5.1. Board ID Representation in Hexadecimal to add the 0x05 Board ID for the CertusPro-NX Versa Board at 2.5GbE Ethernet speed. |
| Implementing the Reference Design | • Updated Table 6.1. CertusPro-NX Platform and Jetson Module Compatibility to list supported Ethernet speeds per platform and to add 2.5GbE support for the CertusPro-NX Versa Board with AGX Orin.<br>• Updated the Jetson AGX Orin Developer Kit and Jetson AGX Thor Developer Kit sections.<br>  • Consolidated the patch application and HSB demo container build steps into a single section workflow for *Building the Holoscan Sensor Bridge Demo Container*.<br>  • Removed the separate *Applying the Patch* subsection.<br>  • Added NGC login procedure and a validation step referencing the video streaming test.<br>• Added Table 6.2. IMX258 Test Command Parameters and Default Values in the Testing the Camera Video Streaming section.<br>• Added three new application test sections:<br>  • Testing the Camera with ISP and MJPEG Features<br>  • Testing the Camera with Body Pose<br>  • Testing the Camera TAO PeopleNet |
| Resource Utilization | • Updated the Resource Utilization section to change the MIPI lane rate from 720 Mbps to 1,500 Mbps and to note that the RUNTIME_MIPI configuration is selected by default and both ISP and MJPEG hardware modules are implemented.<br>• Updated the resource utilization values in the following sections.<br>  • Resource Utilization with Ethernet 1GbE<br>  • Resource Utilization with Ethernet 10GbE using the CertusPro-NX Versa Board<br>  • Resource Utilization with Ethernet 10GbE using the CertusPro-NX Sensor-to-Ethernet Bridge Board<br>• Added a Resource Utilization with Ethernet 2.5GbE section. |
| Appendix A. Known Issues | Added a known issue noting that building the Ethernet 2.5GbE configuration results in a longer place-and-route duration than the 10GbE and 1GbE configurations. |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

| Section | Change Summary |
|---|---|
| References | • Added the following to this section:<br>  • CertusPro-NX Customizable HSB Sensor Interface Kit and Reference Design – Source Code<br>  • ITU-T Recommendation T.81 |

**Revision 1.3, April 2026**

| Section | Change Summary |
|---|---|
| All | • Corrected HSB SDK version from *2.5.0-EA* to *2.5.0.*<br>• Minor editorial fixes. |
| Introduction | • Updated Tri-speed Ethernet version to *2.1.0* in Table 1.2. Summary of CertusPro-NX Versa Board with NVIDIA Jetson AGX Thor.<br>• Updated LVDS DDR Receiver version to *1.7.0* in Table 1.3. Summary of CertusPro-NX Sensor-to-Ethernet Bridge Board with NVIDIA Jetson AGX Orin. |
| Functional Description | • Updated the following *key components* in the Ethernet 10GbE Support with 10GBase-T SFP+ Transceiver section.<br>  • Updated encoding/decoding to support *64b/66b schemes*.<br>  • Removed *1GbE* support in the SFP+ transceiver module. |
| Implementing the Reference Design | • Updated the Test Setup on the CertusPro-NX Versa Board section.<br>  • Updated the reference of step 2 to *Local Firmware Update through USB Interface* in the Programming the Golden FPGA Bitstream with a Jump Table subsection.<br>• Updated the Test Setup on the CertusPro-NX Sensor-to-Ethernet Bridge Board section.<br>  • Updated the reference of step 2 to *Local Firmware Update through JTAG* in the Programming the Golden FPGA Bitstream with a Jump Table subsection. |

**Revision 1.2, March 2026**

| Section | Change Summary |
|---|---|
| All | • Changed the document title from *CertusPro-NX Versa Board – Customizable HSB Sensor Interfaces* to *Customizable HSB Sensor Interfaces on Lattice CertusPro-NX Platforms.*<br>• Added the *CertusPro-NX Sensor-to-Ethernet Bridge Board platform* in this document. |
| Abbreviations in This Document | Added CMOS, DDR, Hex, PMA, and PROM in this section. |
| Introduction | Reworked section content. |
| Directory Structure and File Overview | Reworked section content. |
| Functional Description | Reworked section content. |
| Customizing the Reference Design | Reworked section content. |
| Compiling the Reference Design | Reworked section content. |
| Implementing the Reference Design | Reworked section content. |
| Resource Utilization | • Updated the resource utilization values for CertusPro-NX Versa Board.<br>• Added resource utilization for CertusPro-NX Sensor-to-Ethernet Bridge Board. |
| Debug Methodology | Reworked section content. |
| Appendix A. Known Issues | Added known issue regarding *10GBASE-T SFP+ modules*. |
| References | Added the following:<br>• CertusPro-NX Versa Board User Guide (FPGA-EB-02053)<br>• CertusPro-NX Sensor to Ethernet Bridge Board User Guide (FPGA-EB-02069)<br>• Multi-Boot User Guide for Nexus Platform (FPGA-TN-02145)<br>• CertusPro-NX Versa Board web page<br>• CertusPro-NX Sensor-to-Ethernet Bridge Board web page<br>• CertusPro-NX Versa Holoscan Kit web page |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

| Section | Change Summary |
|---|---|
| | • NVIDIA Jetson Developer Kits web page |

**Revision 1.1, March 2026**

| Section | Change Summary |
|---|---|
| All | • Updated Hololink IP version from *2507* to *2511*.<br>• Added Thor platform. |
| Introduction | • Added the *Thor* platform in this section.<br>• Updated the following in Table 1.1. Summary of the Reference Design<br>  • Lattice Radiant software version to *2025.2.0.48.0*.<br>  • NVIDIA Holoscan SDK from *3.3.0* to *3.7.0*.<br>  • HSB SDK from *2.2.0* to *2.5.0-EA.*<br>• Added the following in Table 1.1. Summary of the Reference Design<br>• Added *Host Software* (*AGX Thor).*<br>  • Added AGX Thor cable 100G QSFP28 to 4×25G SFP28 Breakout Active Optical Cable (AOC). |
| Functional Description | Updated Holoscan Sensor Bridge version from *2.2.0* to *2.5.0-EA* in the Hololink IP Overview section. |
| Customizing the Reference Design | • Added a disclaimer in this section that the screenshots are for reference only.<br>• Updated the GitHub link in the Hololink IP section. |
| Compiling the Reference Design | • Added a disclaimer in this section that the screenshots are for reference only.<br>• Updated Lattice Radiant software version to *2025.2.0.48.0* in the Configuring the Lattice Radiant Software to Compile the Reference Design section. |
| Implementing the Reference Design on the CertusPro-NX Versa Board | • Added hardware and software requirements for NVIDIA Thor.<br>• Updated the HSB SDK version to *2.5.0-EA.*<br>• Added Figure 6.3. Setup with AGX Thor for 10G Ethernet.<br>• Added NVIDIA Platform Host Setup section and moved the Jetson AGX Orin Developer Kit under this section.<br>• Minor updates on the commands in the Configuring the AGX Orin Developer Kit section.<br>• Added the Jetson AGX Thor Developer Kit section.<br>• Added a note indicating that only Jetson AGX Orin supports 1G Ethernet in the Remote Firmware Updated via Ethernet section and in the Camera Video Streaming Test section. |
| Resource Utilization | • Reworked Table 7.1. Logic Consumption for each Instance Available in the Reference Design and in Table 7.3. Logic Consumption for each Instance Available in the Reference Design.<br>• Reworked Table 7.2. Comparison between Reference Design Utilization and CertusPro-NX Resources Available and Table 7.4. Comparison between Reference Design Utilization and CertusPro-NX Resources Available.<br>• Updated the *SLICE packaging percentage* in this section. |

**Revision 1.0, November 2025**

| Section | Change Summary |
|---|---|
| All | Initial release. |

© 2026 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.




www.latticesemi.com