CertusPro-NX Multi-Object Detection on SOM Board Demo User Guide

User Guide

FPGA-UG-02247-1.0

January 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.

Contents

Figures

Abbreviations in This Document

A list of abbreviations used in this document.

Abbreviation	Definition
AC/DC	Alternating Current/Direct Current
AI	Artificial Intelligence
CM5	Compute Module Five
DHCP	Dynamic Host Configuration Protocol
ESD	Electrostatic Discharge
FPGA	Field-Programmable Gate Array
HUB	Host-accelerated Unified Bridge
I2C	Inter-Integrated Circuit
IP Address	Internet Protocol Address
MOD	Multi-Object Detection
OPN	Ordering Part Number
OS	Operating System
PC	Personal Computer
RAM	Random Access Memory
SOM	System On Module
SPI	Serial Peripheral Interface
SSH	Secure Shell
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus
VNC	Virtual Network Computing

1. Introduction

Multi-object detection (MOD) plays a crucial role in enabling machines to perceive and interpret complex visual environments. From autonomous vehicles and surveillance systems to smart retail and robotics, the ability to accurately detect and classify multiple objects in real-time is foundational to many modern AI applications.

2. Functional Description

This document describes MOD running on a system-on-module (SOM) hosting a Lattice CertusPro™-NX device. The aim is to showcase the capabilities of the solution in identifying and localizing various objects within images or video streams.

A key focus of this implementation is the deployment of MOD on small form factor devices, such as edge AI modules, embedded systems, and mobile platforms. These compact yet powerful devices enable real-time inference at the edge, reduce latency, preserve privacy, and minimize the need for cloud-based processing.

3. Demo Setup

3.1. Hardware Requirement

The following hardware components are required to run the MOD demo on the CertusPro-NX SOM:

- CertusPro-NX SOM Board:
  - OPN: LFCPNX-SOM-EVN
  - CertusPro-NX Package: LFCPNX-100-9ASG256I
  - MachXO3D™ Package: LCMXO3D-9400
- CertusPro-NX Carrier Module (LF-GEN-CR-PCBA)
- Raspberry Pi Compute Module 5 (CM5)
- USB-A/USB-C to USB-C cable (Used for data connection between the Raspberry Pi CM5 and a host PC)
- USB-A to micro-USB cable (Used for programming the bitstream, firmware, and ensuring proper terminal prints)
- IMX219 Camera Module: Raspberry Pi Camera Module 2/Arducam 8 MP IMX219 Camera with four-lane support.
- Power adapter for board power: 5 V 25 W AC/DC external wall mount (class II) adapter multi-blade (sold separately) input.

Refer to the following figures for the views of each hardware module and the complete board assembly.

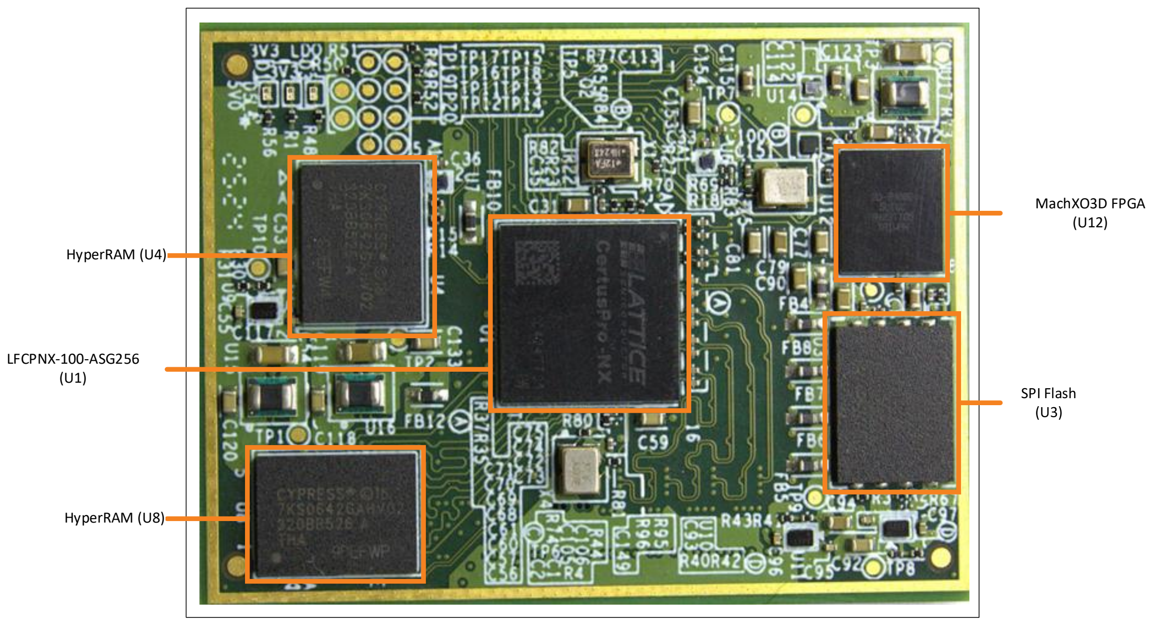


Figure 3.1. CertusPro-NX SOM Board

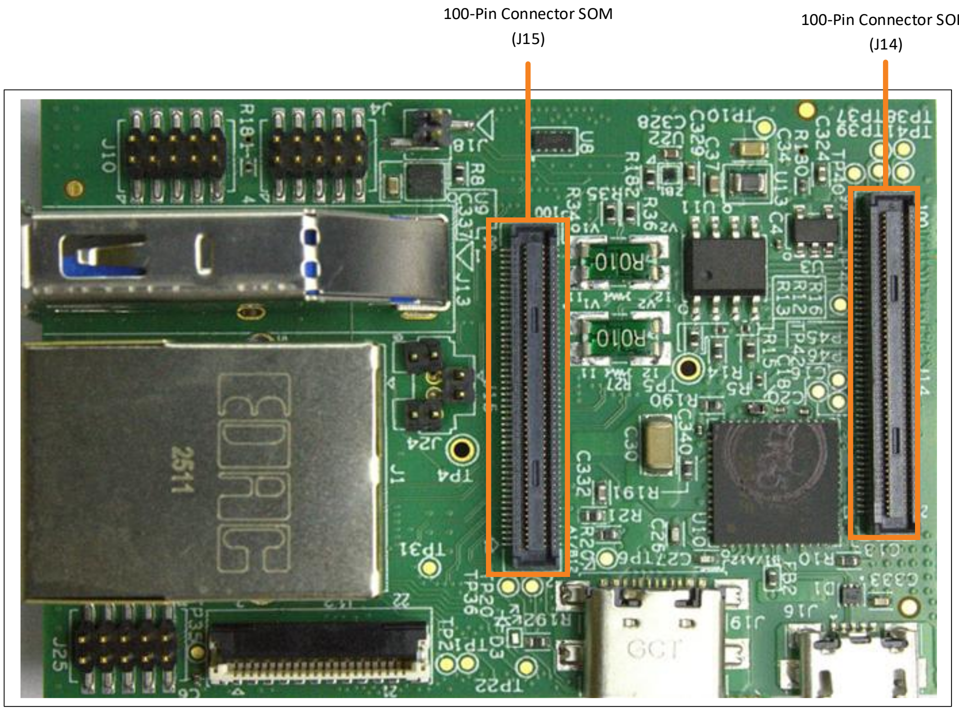


Figure 3.2. CertusPro-NX Carrier Module

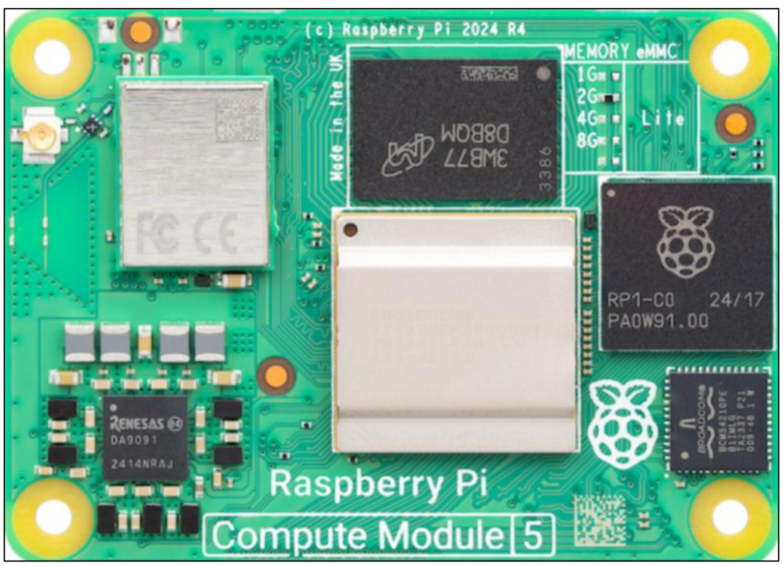


Figure 3.3. Raspberry Pi CM5

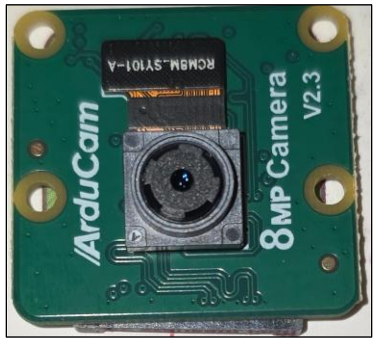


Figure 3.4. IMX219 Camera Module

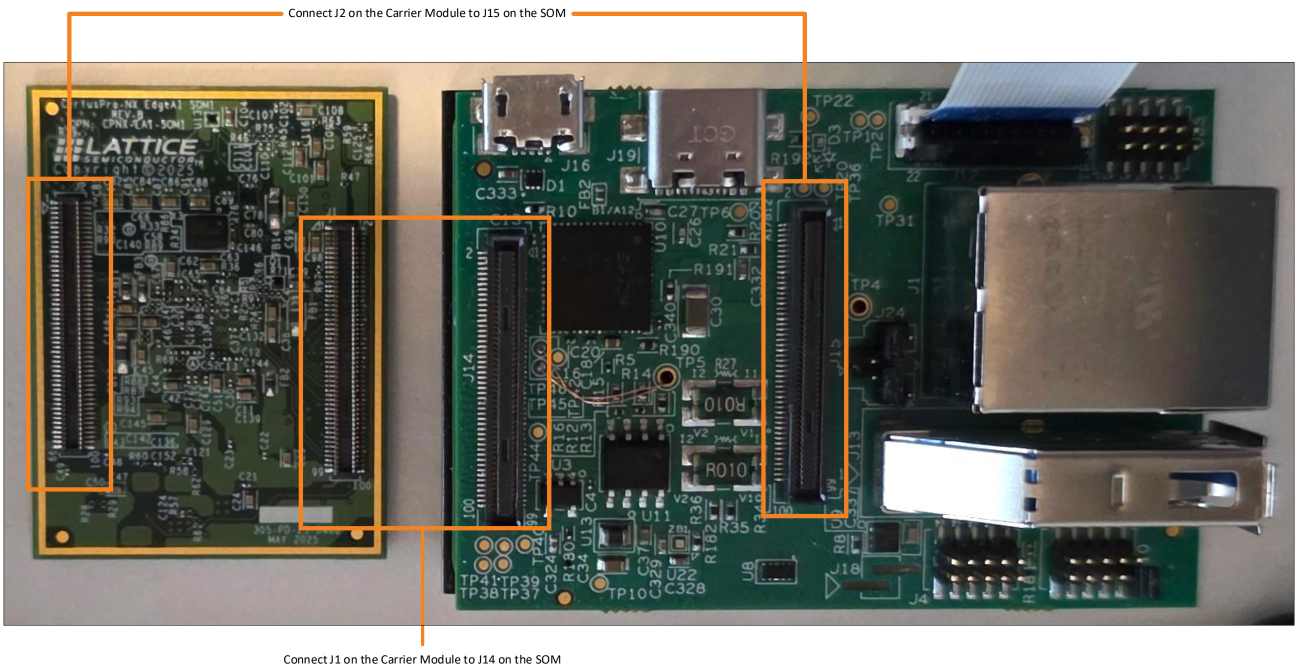


Figure 3.5. Bottom of SOM and Top of Carrier Board with Connection Details

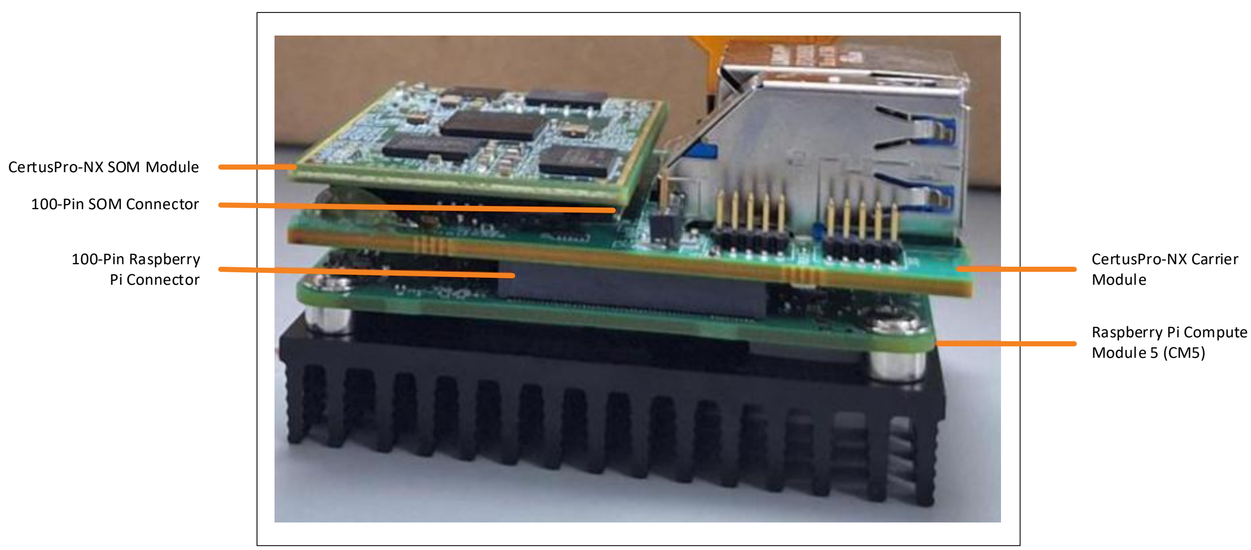


Figure 3.6. SOM-Carrier-Raspberry Pi Assembly

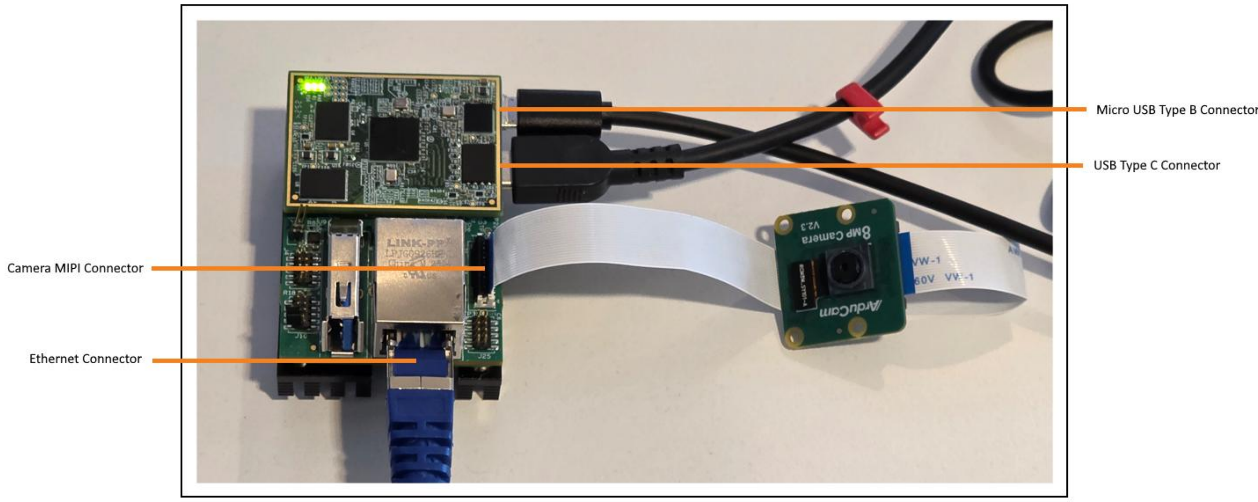


Figure 3.7. SOM-Carrier-Raspberry Pi Assembly with USB-C Power and Ethernet Connected for Normal Operation

3.2. Software Requirements

The following software tools and files are required to run the MOD demo on the CertusPro-NX SOM:

- Lattice Radiant™ Programmer (version 2025.1 or later).
- Raspberry Pi Software
- rpiboot
- Demo package containing:
    - Raspberry Pi OS Image: <date>-CPNX-HUB-1.5-edgeHUB-<version>.img.xz
    - Bitstream: mod_top_torna_revb_cpnx100_impl_1-<version>.bit @ 0x00000000
    - FwRoot: fwroot_<version>.bin @ 0x00300000

4. Programming the Demo

This section focuses on programming the components of the SOM to run the MOD demo.

```
∨ BITSTREAM
    mod_top_torna_revb_cpnx100_impl_1-2025-11-24.bit
∨ CAMERA_CONFIG
    Sony-IMX219-4K-Camera-Configuration-Custom.bin
> docs
∨ FIRMWARE
    gard_fw_320109.bin
∨ MODELS
    mod-cpnx-8.2.0-RGB.bin
∨ ROOT_IMAGE
    fwroot_320109.bin
∨ RPI_OS
    2025-12-17-CPNX-HUB-1.5-edgeHUB-0.2.8.img.xz
```

Figure 4.1. Demo Package Folder Structure

4.1. Setting Up Raspberry Pi CM5

This section describes how to download and image the Raspberry Pi CM5 operating system with preinstalled software and configurations.

4.1.1. Preparing Hardware for Imaging

Warning: The SOM assembly and associated circuitry are delicate high-speed equipment and highly susceptible to ESD. Ensure that you use proper ESD protection measures before handling.

As shown in the figure below, on the SOM assembly, add a 1.27 mm jumper on pins 1 and 2 (Jumper J10).

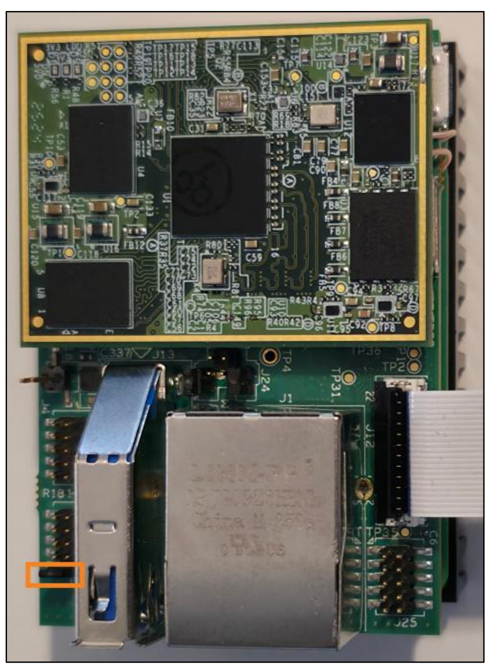


Figure 4.2. Jumper Location on the Carrier Module at J10

4.1.2. Installing rpiboot on Windows PC

rpiboot installs the drivers and boot tool. Do not close any driver installation windows that appear during the installation process.

By default, this package installs at the *C:\Program Files (x86)\Raspberry Pi* directory.

Follow these steps:

1. Install the provided **rpiboot_setup.exe**.
2. Double-click **rpiboot.exe** to run it.

4.1.3. Installing Raspberry Pi Imager on Windows PC

1. Install the provided **imager** executable.
2. Double-click the installer to run it.
3. Select and agree with the policies, then continue with the installation.

4.1.4. Imaging Raspberry Pi CM5

1. Connect the USB-C cable from the device to a Windows PC (use the USB-C port next to the micro-USB port).
   Note: Ensure that the PC is able to provide enough power for imaging the Raspberry Pi CM5. Do not power the device from a PC during normal operation.

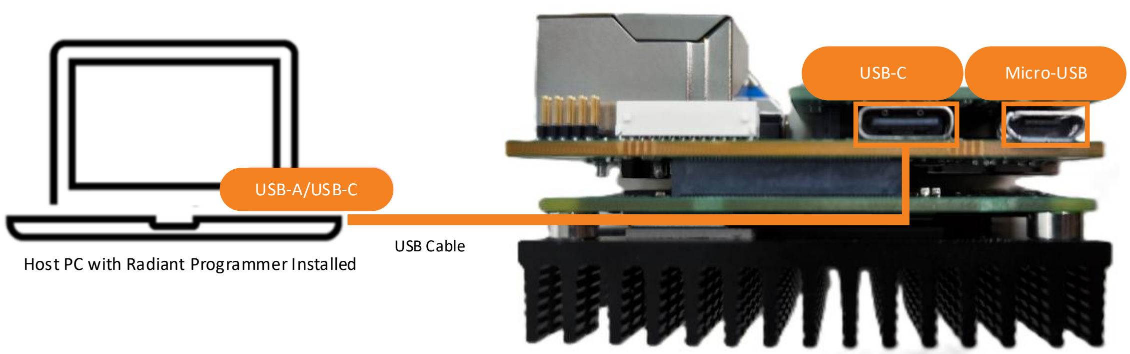


Figure 4.3. Raspberry Pi CM5 to Windows PC Connection for Imaging

2. From Windows, run rpiboot:
```
C:\Program Files (x86)\Raspberry Pi\rpiboot.exe
```

   The Raspberry Pi CM5 is detected by the script, and a storage device appears in Explorer.

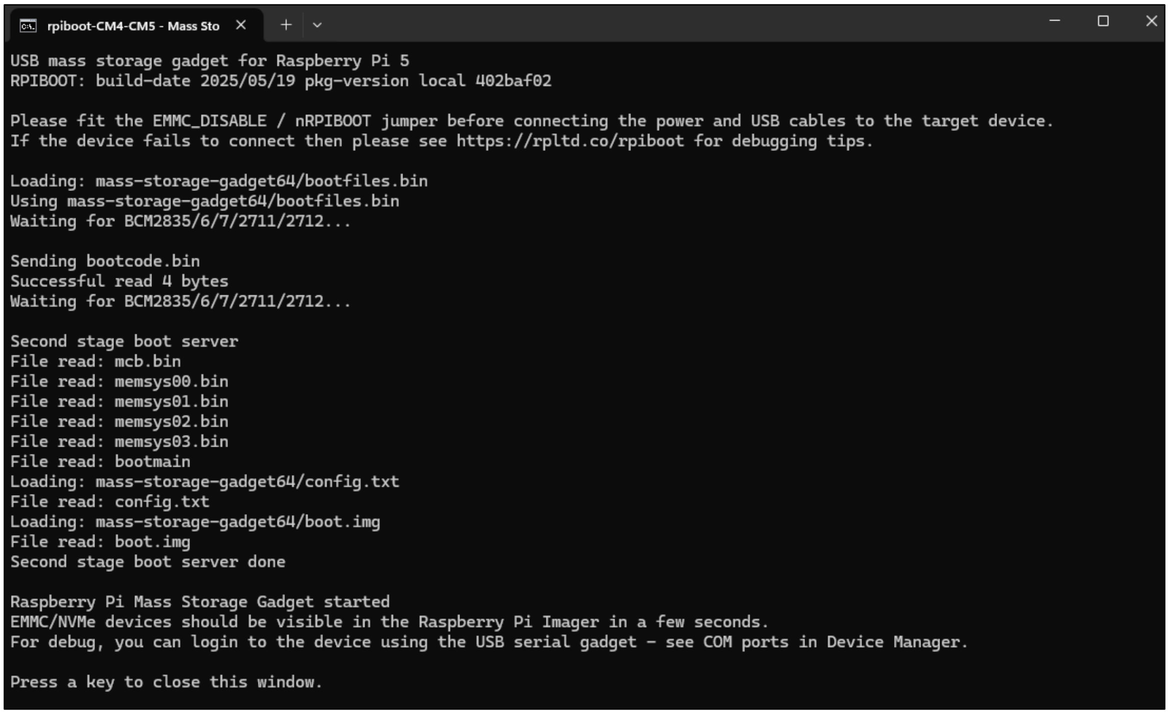


Figure 4.4. rpiboot.exe Successfully Sets the Raspberry Pi CM5 to Mass Storage

3. Open Raspberry Pi Imager:
    a. Click **Choose Device**, then select the Raspberry Pi CM5 option.
    b. Click **Choose OS**, then select the .img file from the demo package.
    c. Click **Choose Storage**, then select the Raspberry Pi CM5 storage device (if no other storage devices are attached, it shows as the only available one).
    d. When prompted for **Customization**, select **No**.
    e. Click **Next** or **WRITE** to write the image to the storage.
    f. Once complete, a pop-up window indicates that the process is complete and the device can be safely disconnected.
4. Remove the USB-C cable and the jumper.
5. Connect the USB-C power adapter.

4.1.5. Configuring Raspberry Pi CM5 Network

The HUB Raspberry Pi OS image comes pre-installed with an ISC DHCP Server for serving dynamic IP addresses to devices connected to the Ethernet port (*eth0*). The DHCP server starts automatically on boot with a static IP address.

To set up and use the Ethernet connection for the SOM Raspberry Pi CM5 after flashing:

- Your machine should have an Ethernet port with dynamic IP address configured.
- Connect an Ethernet LAN cable from your machine to the SOM Ethernet port.
- Connect the power supply to the SOM.
- Optionally, as described in the previous section, use PuTTY to access the serial debug UART and confirm the IP address.

A static IP address *10.10.1.1/24* is automatically assigned to the SOM Raspberry Pi CM5. If this IP address is pre-allotted, a subsequent IP in the range *10.10.1.100* to *10.10.1.200* is assigned.

Run the following command to check the IP address:

```
hostname -I
```

The IP address is now accessible within the network.

In case of failure, power cycle the SOM for a seamless reboot and IP allocation.

The following figure shows an example of accessing the SOM assembly over SSH. Use the default credentials to log in.

Default credentials:

- **Username**: *lattice*
- **Password**: *lattice*

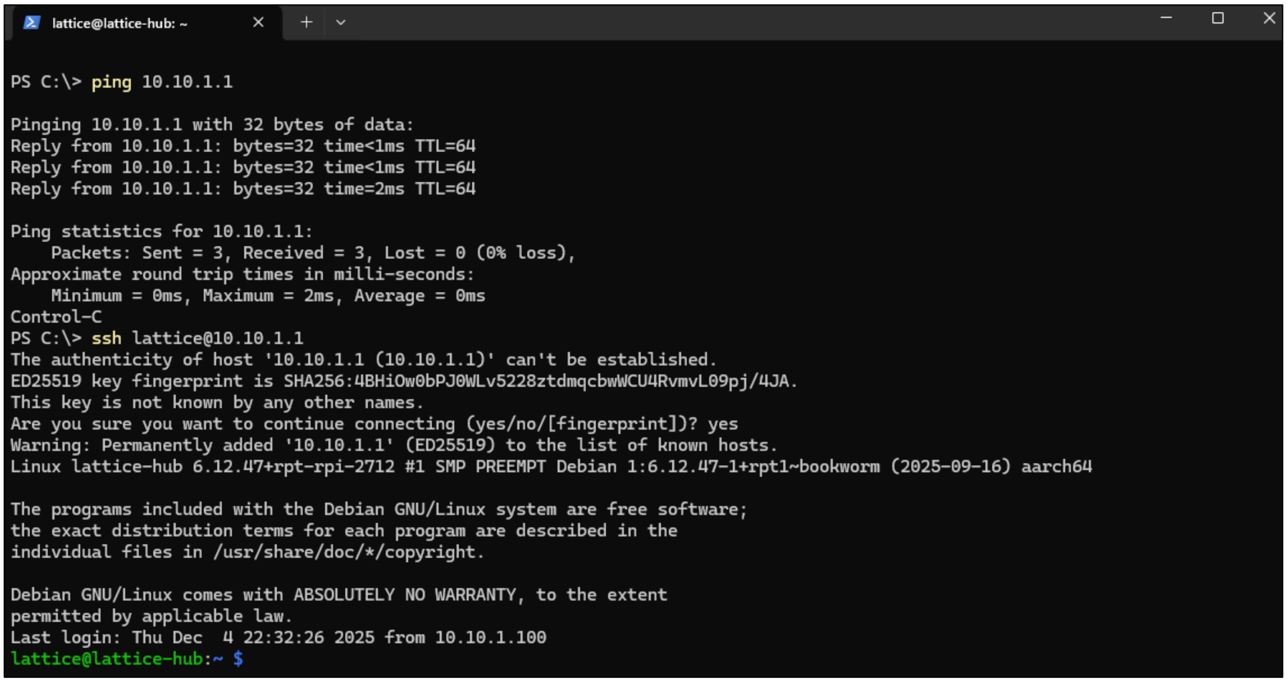


Figure 4.5. Windows Terminal Accessing SOM through Ethernet

4.1.5.1. Enabling Static IPv4 Address

This configuration enables the SOM to be accessed over the LAN network. To set a static IP for the Raspberry Pi CM5 on the SOM assembly, run the following command. This needs to be configured only once and will automatically reconnect if available:

```
nmcli connection modify <connection-name> ifname eth0 type ethernet ipv4.addresses <ip-addr>/24 \
ipv4.gateway <gateway> ipv4.method manual autoconnect yes
```

Where:

- **connection-name** - *Wired connection 1* (already set up on Raspberry Pi CM5)
- **ip-addr** – IPv4 address
- **gateway** – IPv4 gateway

Note: Root privileges are required. If you encounter a *permission denied* error, run the command using sudo or with a user account that has elevated privileges.

4.2. Updating HUB Package

If a new package is provided without a full Raspberry Pi OS update:

1. Remote into the Raspberry Pi CM5.
2. Copy the *lscc_hub-X.X.X-arm64.deb* package to the Downloads folder on the Raspberry Pi CM5 using a USB key.
3. Install the package in Terminal:
   a. Open Terminal.
   b. Run:

```
sudo apt install lscc-hub_X.X.X_arm64.deb -y
```

(Example: lscc_hub-1.5.0-arm64.deb)

4.3. Updating Software Package

If a new package is provided without a full Raspberry Pi OS update:

1. Uninstall the previous package in Terminal:
```
sudo apt remove lscc-hub -y
```

2. Install the latest package in Terminal:
```
sudo apt install lscc-hub_X.X.X_arm64.deb -y
```

(Example: lscc_hub-1.5.0-arm64.deb)

4.4. Programming MachXO3D FPGA Using Radiant Programmer

If the SOM is not detected by the Radiant Programmer tool as an LFCPNX-100 device, you likely need to program the MachXO3D FPGA. For detailed programming instructions, refer to the RTL User Guide available on the sensAI Reference Design GitHub page.

4.5. Programming CertusPro-NX SOM with MOD Demo Using Radiant Programmer

1. Install the Radiant Programmer software in its default location: *C:\lscc\programmer\radiant\2025.1*.
2. Run the Radiant Programmer software:
   a. Click **Scan Device** to ensure the SOM is detected as an LFCPNX-100 device. If it is not detected, verify the connection or replace the micro-USB cable. Otherwise, refer to the Programming MachXO3D FPGA Using Radiant Programmer section.
   b. Enable **LFCPNX** as the listed device.

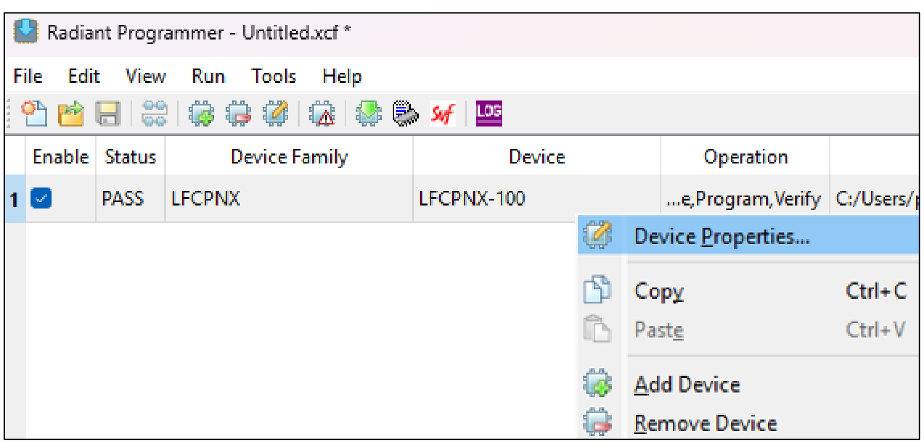


Figure 4.6. CertusPro-NX (LFCPNX-100) Device Listed in Radiant Programmer Software

   c. Right-click the device and select **Device Properties**.

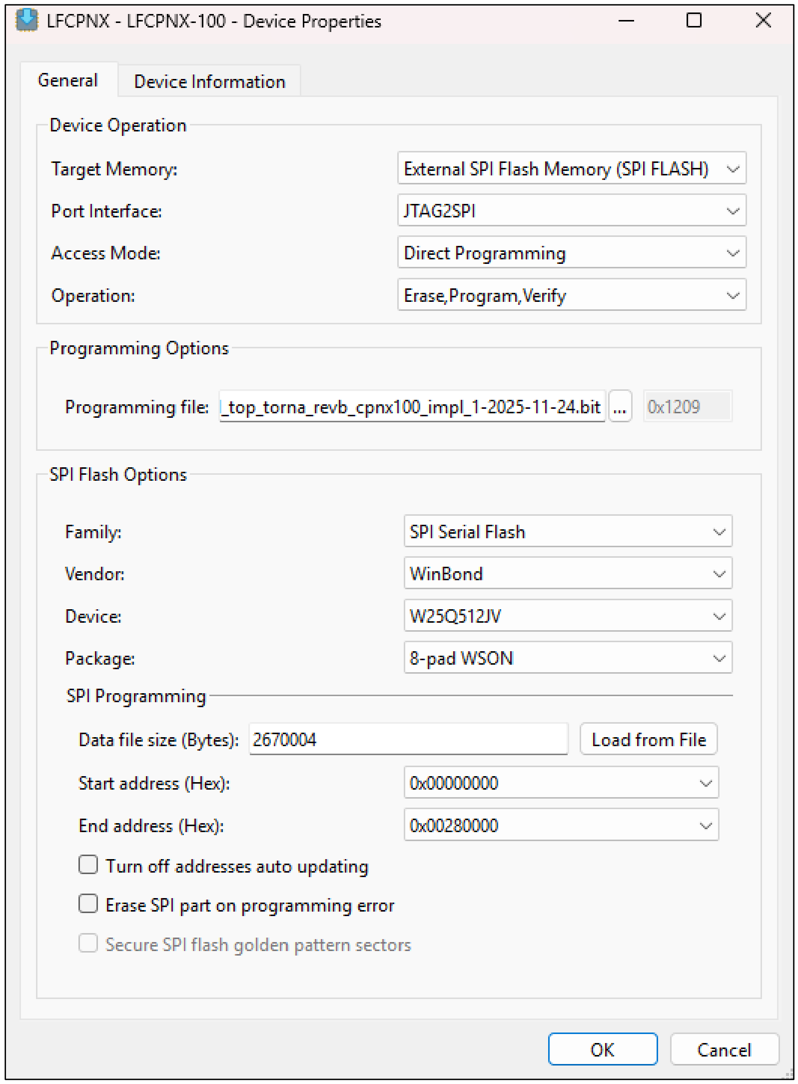


Figure 4.7. Radiant Programmer SPI Flash Options for Bitstream File Programming

d. For **Target Memory**, select **External SPI Flash**. This expands the options.

e. For **Programming file**, select the *.bit* file you downloaded (for example, *mod_top_torna_revb_cpnx100_impl_1-2025-11-24 1.bit*).

f. Click the Load from File button next to Data file size.

g. For the **Start address (Hex)**, select *0x00000000*.

h. Click **OK**.

i. From the toolbar menu, click **Run**, then select **Program Device**.

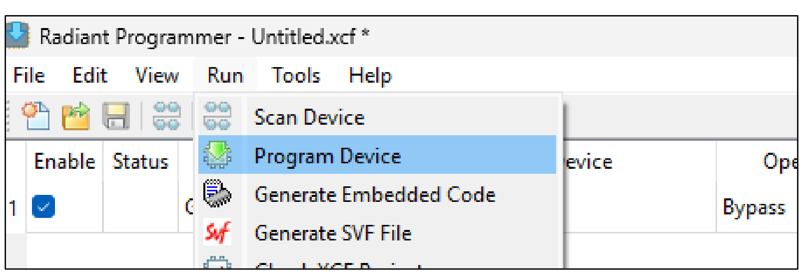


Figure 4.8. Programming FPGA from Radiant Programmer Software

3. This process takes approximately 20 minutes. When successful, the following output appears.

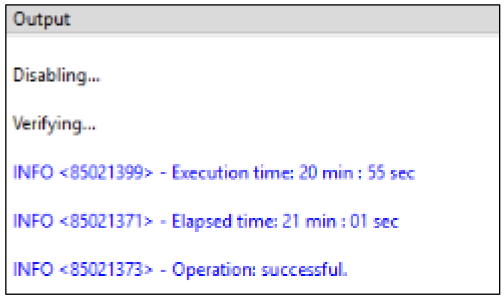


Figure 4.9. Successful Operation Output

4. Repeat this process with *FwRoot.bin* and set the start address to *0x00300000*.

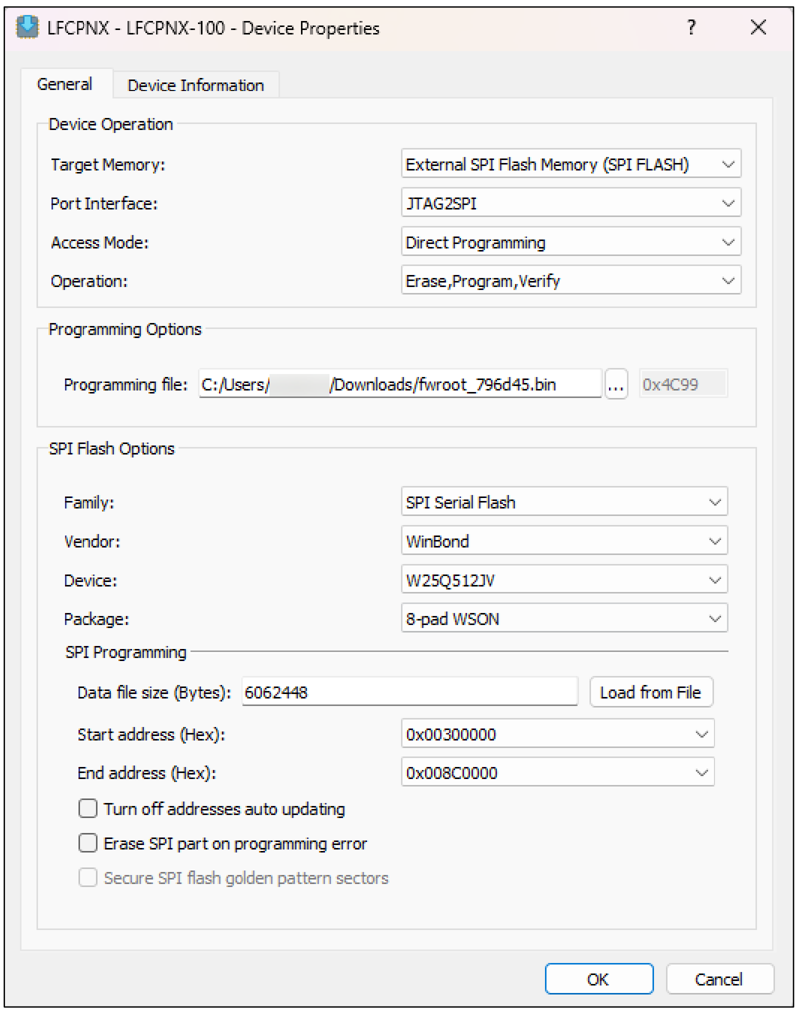


Figure 4.10. Radiant Programmer SPI Flash Options for Binary File Programming

5. Running the Demo

5.1. Preparing the Device

The Raspberry Pi image comes with a static IP address set to *10.10.1.1*. Follow these steps to connect:

1. Connect an Ethernet cable from the SOM to your computer.
2. Connect power to the Raspberry Pi CM5 through the USB-C port next to the micro USB port. If the device is already powered on, power cycle the device by unplugging the power and plugging it back in.

5.2. Starting Scripts in VNC

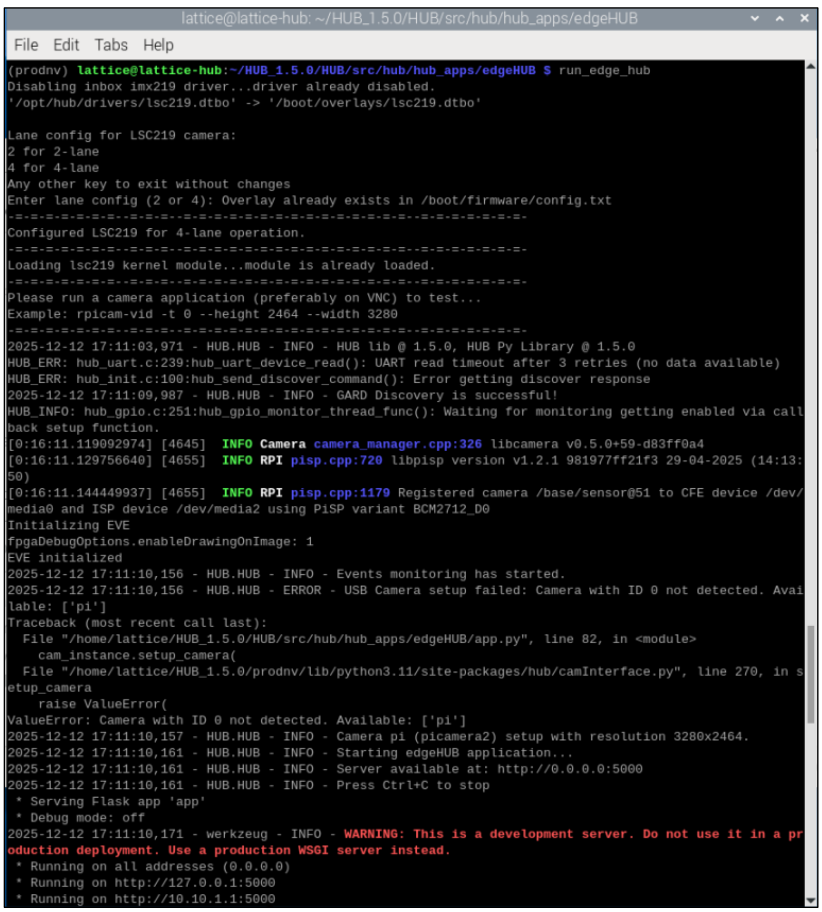
5.2.1. Connecting RealVNC Viewer to IP Address

1. Open RealVNC Viewer.
2. Enter the IP address of the Raspberry Pi CM5 retrieved in the Preparing the Device section.
3. Enter the username and password.
4. The Raspberry Pi OS desktop appears.

5.2.2. Running Scripts

From the Raspberry Pi OS desktop, open Terminal, and run:

```
run_edge_hub
```



```
lattice@lattice-hub: ~/HUB_1.5.0/HUB/src/hub/hub_apps/edgeHUB
File Edit Tabs Help
(prodnv) lattice@lattice-hub:~/HUB_1.5.0/HUB/src/hub/hub_apps/edgeHUB $ run_edge_hub
Disabling inbox imx219 driver...driver already disabled.
'/opt/hub/drivers/lsc219.dtbo' -> '/boot/overlays/lsc219.dtbo'

Lane config for LSC219 camera:
2 for 2-lane
4 for 4-lane
Any other key to exit without changes
Enter lane config (2 or 4): Overlay already exists in /boot/firmware/config.txt
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
Configured LSC219 for 4-lane operation.
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
Loading lsc219 kernel module...module is already loaded.
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
Please run a camera application (preferably on VNC) to test...
Example: rpicam-vid -t 0 --height 2464 --width 3280
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
2025-12-12 17:11:03,971 - HUB.HUB - INFO - HUB lib @ 1.5.0, HUB Py Library @ 1.5.0
HUB_ERR: hub_uart.c:239:hub_uart_device_read(): UART read timeout after 3 retries (no data available)
HUB_ERR: hub_init.c:100:hub_send_discover_command(): Error getting discover response
2025-12-12 17:11:09,987 - HUB.HUB - INFO - GARD Discovery is successful!
HUB_INFO: hub_gpio.c:251:hub_gpio_monitor_thread_func(): Waiting for monitoring getting enabled via call
back setup function.
[0:16:11.119092974] [4645]  INFO Camera camera_manager.cpp:326 libcamera v0.5.0+59-d83ff0a4
[0:16:11.129756640] [4655]  INFO RPI pisp.cpp:720 libpisp version v1.2.1 981977ff21f3 29-04-2025 (14:13:
50)
[0:16:11.144449937] [4655]  INFO RPI pisp.cpp:1179 Registered camera /base/sensor@51 to CFE device /dev/
media0 and ISP device /dev/media2 using PiSP variant BCM2712_D0
Initializing EVE
fpgaDebugOptions.enableDrawingOnImage: 1
EVE initialized
2025-12-12 17:11:10,156 - HUB.HUB - INFO - Events monitoring has started.
2025-12-12 17:11:10,156 - HUB.HUB - ERROR - USB Camera setup failed: Camera with ID 0 not detected. Avai
lable: ['pi']
Traceback (most recent call last):
  File "/home/lattice/HUB_1.5.0/HUB/src/hub/hub_apps/edgeHUB/app.py", line 82, in <module>
    cam_instance.setup_camera(
  File "/home/lattice/HUB_1.5.0/prodnv/lib/python3.11/site-packages/hub/camInterface.py", line 270, in s
etup_camera
    raise ValueError(
ValueError: Camera with ID 0 not detected. Available: ['pi']
2025-12-12 17:11:10,157 - HUB.HUB - INFO - Camera pi (picamera2) setup with resolution 3280x2464.
2025-12-12 17:11:10,161 - HUB.HUB - INFO - Starting edgeHUB application...
2025-12-12 17:11:10,161 - HUB.HUB - INFO - Server available at: http://0.0.0.0:5000
2025-12-12 17:11:10,161 - HUB.HUB - INFO - Press Ctrl+C to stop
 * Serving Flask app 'app'
 * Debug mode: off
2025-12-12 17:11:10,171 - werkzeug - INFO - WARNING: This is a development server. Do not use it in a pr
oduction deployment. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http://10.10.1.1:5000
```

Figure 5.1. Console Output from HUB when a Client is Connected

Note: A few errors may appear because the system attempts to identify available interfaces, such as UART, I2C, and cameras.

5.3. Using HUB

1. Open a web browser on Windows.
2. Enter the IP address used above and append the port *:5000* in the address bar (for example, *10.10.1.1:5000*).
3. Navigate to the **Live Streaming** tab.
4. Click on the **Start Stream** button.
5. Ensure the **Display AI Metadata** checkbox is checked. If not, check it.
6. The model has been trained to the eight classification types below. Point the camera to an object or of a high quality printout of an object and observe the camera feed displaying a bounding box and identification. The eight classification types:
    - Person
    - Bicycles
    - Cars
    - Motorcycles
    - Buses
    - Trucks
    - Traffic lights
    - Stop signs

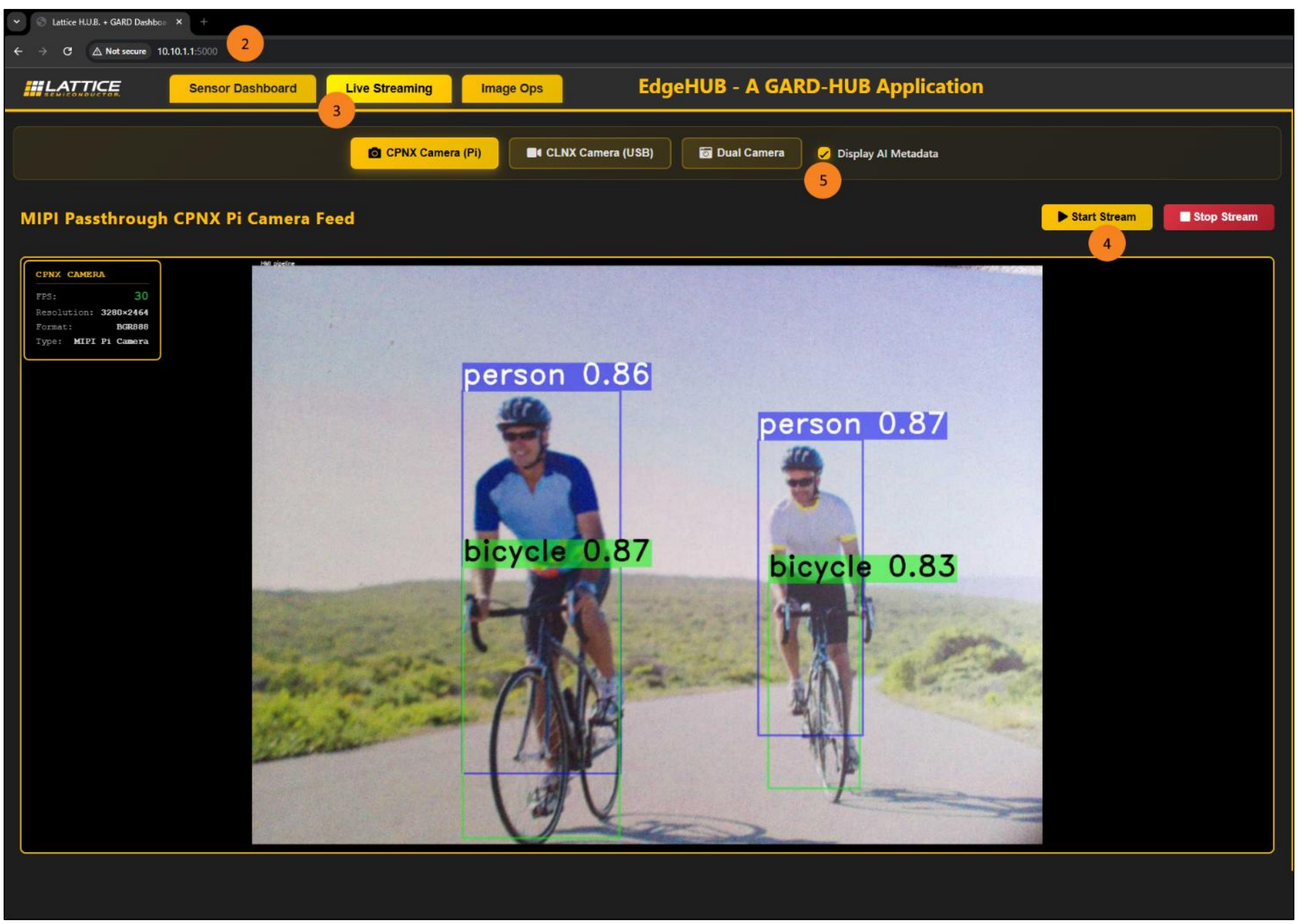


Figure 5.2. Web Application Visualizer – AI Metadata Overlay with Detection of 2 Persons and 2 Bicycles

References

- sensAI GARD Application Reference Design User Guide (FPGA-RD-02332)
- Lattice sensAI Solution Stack web page
- CertusPro-NX web page
- MachXO3D web page
- Lattice Radiant Software web page
- Lattice Solutions IP Cores web page
- Lattice Solutions Reference Designs web page
- Lattice Insights web page for Lattice Semiconductor training courses and learning plans

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.

Revision History

Revision 1.0, January 2026

Section	Change Summary
All	Initial release.

www.latticesemi.com