NVIDIA Jetson Linux

Release Notes

Version 39.2.0 GA

RN_10698-r39.2.0 | June 2026

Table of Contents

1. About This Release 2
    1.1. Platform and Release Information 2
    1.2. Login Credentials 3
    1.3. What's New 3
    1.4. OE Builds 4
2. Using Jetson AGX Thor Developer Kit to Emulate T4000 Module 4
3. Known Issues 5
    3.1. General System Usability 5
    3.2. Camera 9
    3.3. Multimedia 11
    3.4. Graphics 11
    3.5. Connectivity 11
    3.6. Display 12
    3.7. Compute Stack 12
    3.8. Multi-Instance GPU (MIG) 13
4. Fixed Issues 14
5. Implementation Details 18
    5.1. Camera 18
        5.1.1. SIPL Framework for Camera Streaming 18
        5.1.2. Camera Serial Interface 20
        5.1.3. Device Registration 22
        5.1.4. Camera Use Cases for CoE and GMSL and SIPL Example Applications 24
    5.2. Device Tree Change for nvethernet 24

1. About This Release

The NVIDIA® Jetson® Linux 39.2.0 General Availability (GA) release includes the Linux Kernel 6.8, the UEFI-based Bootloader, the Ubuntu 24.04-based root file system, NVIDIA drivers, the necessary firmware, toolchain, and more. This release supports the NVIDIA Jetson Thor™ and Jetson Orin platforms™.

Note: This GA release can be used for production purposes.

1.1. Platform and Release Information

Description	Supported Version
Host machine Linux distribution for flashing software on the hardware platforms supported by this release.	Ubuntu 24.04 and 22.04
Sample rootfs that was derived from Ubuntu to run on Tegra devices that are supported in this release.	Ubuntu 24.04
Supported Linux kernel version.	6.8
Supported Arm® architecture.	aarch64
Name of the configuration file used in flashing. **Note:** When you flash a configuration file with `flash.sh`, specify the configuration's basename; for example, the file name without the `.conf` suffix.	For details, refer to Quick Start in the Jetson Linux Developer Guide.
Toolchain for cross-compilation.	GCC 13.2
Release tag on source server (https://gitlab.com/nvidia/nv-tegra).	jetson_39.2_GA

1.2. Login Credentials

To create your login credentials, at the first boot, follow the system prompts.

1.3. What's New

Jetson Linux 39.2.0 is the latest Jetson Linux release that supports NVIDIA JetPack™ 7.2. With Jetson Linux 39.2.0, Jetson software aligns with the Server Base System Architecture (SBSA), positioning Jetson Thor alongside industry-standard ARM server design. The following are the highlights for this release:

- Adds support for the Jetson Orin product family within JetPack 7 releases.
- Updates the `nv-load-display-modules` service to manage driver differences between Jetson Orin and Jetson Thor at run time. For more information, refer to nv-load-display-modules Service in the Jetson Linux Developer Guide.
- Introduces native support for single-command NemoClaw installation on developer kits.
- Adds support for Jetson AGX Orin 32GB Super Mode (MAXN_SUPER).
- Introduces a unified ISO-based installation method for both Jetson Orin and Jetson Thor developer kits.
- Provides official Yocto recipes for Jetson Orin and Jetson Thor platforms through the OE4T GitHub repository.
- Enables Multi-Instance GPU (MIG) on Jetson Thor T5000 as a technology preview feature.
- Includes Jetson SIPL API Package v2.0.0 for JetPack 7.2 GA, a major release that expands SIPL camera support and introduces a new unified camera framework for both GMSL and CoE use cases.
    - Key updates include production GMSL driver support, new unified `nvsipl_camera` and `nvsipl_query` sample applications, stereo pipeline support for GMSL and CoE sensors, expanded UDDF driver interfaces, new notification and power diagnostic events, additional CSI and pixel format support, and improved camera/module query capabilities.
    - This release also includes significant changes from JetPack 7.1 that affect migration, including API namespace and type renames, updated JSON configuration requirements, new driver database loading behavior, changed package and installation paths, renamed sample apps, and ABI changes requiring JetPack 7.1 UDDF drivers to be rebuilt against JetPack 7.2 headers.

For information about a variety of implementation details, refer to Implementation Details in these release notes.

1.4. OE Builds

Beginning with JetPack 7.2 and Jetson Linux 39.2, we are releasing official NVIDIA Open Embedded/Yocto recipes. These recipes use the sources of the same BSP as JetPack 7.2. You can use them to build customized targeted images for your application. Recipes will be posted at https://github.com/OE4T.

2. Using Jetson AGX Thor Developer Kit to Emulate T4000 Module

This section provides information on how to emulate the T4000 module by using the Jetson AGX Thor developer kit with this BSP.

1. On the Jetson AGX Thor dev kit, append `mem=64G` in the `Append` specification in `/boot/extlinux/extlinux.conf` to modify the memory size in the kernel command line. After modification, save the file and reboot the unit to reflect the change.
2. Depending on your needs, switch to 90W mode or 70W mode.

```
Shell
sudo nvpmodel -m 2 # to emulate MAXN mode of T4000
sudo nvpmodel -m 3 # to emulate 70W mode of T4000
```

Note: T4000 emulation using the Jetson AGX Thor dev kit is applicable for CPU, GPU, and memory performance emulation. The other engines, such as NVDEC and NVENC, are not emulated.

3. Known Issues

This section provides details about issues that were discovered during development and QA but have not yet been resolved in this release.

3.1. General System Usability

The following general system usability-related issues are noted in this release.

<table>
<tr><th>Issue</th><th>Description</th></tr>
<tr><td>6266271</td><td>When installing ISO on Orin devices with older QSPI images, it is important to allow Capsule update of the QSPI image. To trigger this, press “y” when prompted. ISO installation continues after Capsule update is complete. Skipping this step causes installation issues due to incompatibility of new ISO images with older QSPI images.</td></tr>
<tr><td>6279443</td><td>Jetson Orin Nano units that are updated to JetPack 7.2 using ISO install continue to use the same profile that was set before update. Units will not default to “Super” mode after the update. To use “Super” mode, you must flash the target using a Linux host or SDKM.</td></tr>
<tr><td>6236259</td><td>On Jetson Orin platforms, reducing the EMC to a value below Fmax (such as 3199 MHz) with the following power modes by using <code>nvpmodel.service</code> during systemd initialization can cause system crashes upon reboot. This issue is especially noticeable when a display is connected, regardless of its resolution. The following platforms and power modes are affected:
<table>
<tr><th>Platform Freq</th><th>MAXN EMC Freq</th><th>Power Mode</th><th>Its EMC Freq</th></tr>
<tr><td>AGX Orin 64GB</td><td>~3200 MHz</td><td>15W</td><td>2133 MHz</td></tr>
<tr><td>AGX Orin 32GB</td><td>~3200 MHz</td><td>15W</td><td>2133 MHz</td></tr>
<tr><td>AGX Orin Industrial</td><td>~3200 MHz</td><td>15W</td><td>2133 MHz</td></tr>
<tr><td>Orin NX 16GB</td><td>~3200 MHz</td><td>10W</td><td>2133 MHz</td></tr>
<tr><td>Orin NX 8GB</td><td>~3200 MHz</td><td>10W</td><td>2133 MHz</td></tr>
<tr><td>Orin Nano 8GB</td><td>~3200 MHz</td><td>7W</td><td>2133 MHz</td></tr>
</table>
</td></tr>
</table>

Issue	Description
	To prevent this problem, 1. Switch to MAXN power mode that restores the EMC to Fmax before rebooting 2. After the system restarts, apply the desired power mode. 3. If the system is inadvertently rebooted while in a problematic power mode, disconnect the display before booting to allow the system to start with the selected mode, and reconnect the display after it finishes initializing.

5748062	The TPM-based hardware random-number generator (hwrng) contributes multiple hundreds of microseconds in real-time latencies. To mitigate this issue, choose one of the following options.
	Option 1:
	Disable the TMP-based hwrng kernel config by setting `CONFIG_HW_RANDOM_TPM=N`.
	If the kernel config `CONFIG_HW_RANDOM_OPTEE` is set either to `M` or `Y`, hwrng is replaced by the OP-TEE–based HWRNG.
	Option 2:
	1. Identify the hwrng kernel thread PID:
	`ps aux | grep hwrng`
	2. Pin the thread to a non-RT CPU core. For example, if the PID of the hwrng kernel thread is 100, use the following:
	`taskset -p -c 0 100`
	The `-c 0` option specifies CPU core 0.

5733592	For Jetson Thor, BSP assumes that your NVMe is larger than 234 GiB. If your NVMe is smaller, do the following to avoid flashing failures. When you run the initrd flash script, add `EXT_NUM_SECTORS=<NVMe storage size in bytes> / 512` as an environment variable: `$ sudo EXT_NUM_SECTORS="<NVMe storage size in bytes> / 512" ./tools/kernel_flash/l4t_initrd_flash.sh`

Issue	Description
5727037	We addressed the hardware issue "SMMU-3 Incomplete invalidation of TLB structures" by creating a software workaround. A side-effect of the workaround is degradation of SMMU UnMap performance. Any use case that includes SMMU UnMap in its performant path will see degradation in performance.
5437763	When a debug cable is removed and re-inserted, minicom becomes inaccessible. Restarting minicom fixes the issue.
5680083	USB Host mode does not work with a T4000 module on a Jetson AGX Thor baseboard with a USB Type-C RCM port.
5251623	On Jetson AGX Thor, some peripherals (like RealSense Camera or Ethernet adapter) do not get enumerated when plugged into a USB Type C port. To enable USB host mode on recovery port, do the following: 1. Enter `echo none > /sys/class/usb_role/usb2-0-role-switch/role`. 2. Connect the USB gadget to the port. 3. Set the data role to host by entering `echo host > /sys/class/usb_role/usb2-0-role-switch/role`.
5424568	ISO installation fails when a USB drive with the image is plugged into a “USB3.0 4-port Portable Hub Model UH400” USB hub. Other USB sticks or hubs work as expected.
5437041	During the OEM configuration process, navigating back to previous pages to reenter the computer name can throw an error saying that the name is not unique even though the configuration is not yet saved.
5463861	When installing ISO in headless mode without a display connected, the UEFI console `/dev/ttyACM1` is filled with a lot of output, which makes the ISO installation difficult. This issue is resolved after QSPI is updated to GA (38.2) release. Issue is also not seen when a display is connected to the target.
5460707	Flashing to target sometimes fails when a Trendnet TU2-ET100 USB-to-Ethernet dongle is connected. We did not observe issues with other USB-to-Ethernet dongles.
5116642	For Jetson Thor, the OC event counter (reported by `/sys/class/hwmon/hwmon*/oc*event_cnt`) starts at 1 even when no OC event has occurred.

Issue	Description
5406663	Currently, you can monitor GPU utilization by using `nvidia-smi dmon`. However, because of design changes, integration of GPU utilization in Jetson Power GUI is still under evaluation**.**
5130696	In Jetson Thor, ASPM for PCIe devices might not be enabled due to the low-power state exit latency of PCIe controllers, specifically Gen5 speed. To avoid the issue, configure the speed of the PCIe controller to lower Gen.
5435196	Jetson Thor can sometimes get stuck during continuous boot stress cycles and will need a cold reset to recover.
5661165	After stopping the vLLM/SGLang servers or stopping the Docker container, memory usage might stay high. Use the following command to free the cached memory: `sudo sh -c "sync; echo 3 > /proc/sys/vm/drop_caches"`
5699079	If excessively large CUDA memory allocation is attempted, the device might reboot. Ensure that CUDA or other applications do not request more memory than physically available. Another workaround is to launch CUDA processes with higher OOM scores to avoid killing of system processes.
5787211	The Vision model `RedHatAI/Llama-3.2-11B-Vision-Instruct-quantized.w4a16` fails to run on the latest NGC VLLM container because of incompatible packages.
6235055	Jetson Orin Nano sometimes sees DCE aborts during SC7 suspend/resume transitions, and triggers a system watchdog reset.
6229695	Flashing Jetson ISO over http devices is not currently supported.
6244887	Flashing Jetson AGX Orin with Real-Time (RT) kernel with display connected could lead to a system crash. Updating to RT kernel using Over-the-Air (OTA) updates after flashing non-RT kernel resolves the issue. Working in headless mode (without display) also resolves the issue.

3.2. Camera

The following camera-related issues are noted in this release.

Issue	Description

6236205	Jetson IO tool fails on Jetson Orin Nano Super devices when flashed using ISO images. Units flashed with other methods do not see the same issue. The following steps work around this issue:
	1. Find the correct DTB that matches the target by running the following:
	Shell
	`dtb=$(for f in /boot/*.dtb; do [ "$(fdtget "$f" / compatible 2>/dev/null \| xargs)" = "$(tr '\0' ' ' < /proc/device-tree/compatible | xargs)" ] && [ "$(fdtget "$f" / model 2>/dev/null | xargs)" = "$(tr '\0' ' ' < /proc/device-tree/model \| xargs)" ] && echo "$f" && break done)`
	`echo "$dtb"`
	If this step prints nothing, no exact matching DTB was found in `/boot`; do not run step 2.
	2. Copy the matching DTB to `/boot/dtb/` with the `kernel_` prefix:
	Shell
	`sudo mkdir -p /boot/dtb`
	`sudo cp "$dtb" "/boot/dtb/kernel_$(basename "$dtb")"`
	3. Run Jetson-IO again:
	Shell
	`sudo /opt/nvidia/jetson-io/jetson-io.py`

5383804	Flashing of FPGA FW might fail if setup is not properly configured. To prevent FPGA FW failure, ensure the following steps are followed: 1. Verify that the camera is connected and that you can ping it. 2. Use `hololink-enumerate` to capture your current FPGA FW version.

Issue	Description
	3. Ensure that you are using the correct flash script and FW version for the type of camera you are using, per the user guide instructions. 4. When flashing starts, do not interrupt the process.
5449309	The error `NOTIF_ERROR_ICP_BAD_INPUT_STREAM` might be observed for the first few frames. This is expected because HSB generates sequence numbers that need to be matched and thus skips frames as required.
5500614	VB1940 might generate corrupted images during overnight long-run tests in preview mode.
5476402	The first two or three frames generated from VB1940 are corrupted. Image capture should be done after that.
5783937	SIPL GStreamer pipelines might fail after installing the SIPL package because of symbol lookup errors in UDDF driver libraries, causing undefined-symbol failures. Apply the following workaround to help fix the issue: 1. `sudo cp /usr/lib/nvsipl_uddf/libnvuddf_eagle_library.so /tmp/libnvuddf_eagle_library.so_orig` 2. `cd /usr/src/jetson_sipl_api/sipl/` 3. `sudo mkdir build` 4. `cd build` 5. `sudo cmake ..` 6. `sudo make nvsipl_coe_camera` 7. `sudo cp ./samples/coe_camera/nvsipl_coe_camera ~/` 8. `sudo cp /tmp/libnvuddf_eagle_library.so_orig /usr/lib/nvsipl_uddf/libnvuddf_eagle_library.so` 9. Run the `nvsiplsrc` pipeline.
6241469	Jetson AGX Thor when connected to display via DP-alt over USBC could sometimes cause a kernel crash with the following errors `top-cbb-fabric@0x8100800000 TIMEOUT_ERR`

3.3. Multimedia

The following issues related to multimedia are noted in this release.

Issue	Description
5388059	USB camera preview fails with OSS Cheese application because of conflicts between OSS and NVIDIA JPEG library.
4506985	The `emc_log` tool is not showing correct memory bandwidth for iGPU-based NVJPG HW engine on Jetson Thor.

3.4. Graphics

The following issues related to graphics are noted in this release.

Issue	Description
—	—

3.5. Connectivity

The following connectivity issues are noted in this release.

Issue	Description
5392554	MGBE on the Jetson AGX Thor developer kit shows lower performance on the 25G interface. The 10G interface has been validated on copper cable; to increase 10G MGBE performance, switch to threaded mode. For example, for MGBE0, use the following command: `echo 1 > /sys/devices/platform/bus@0/a808a10000.ethernet/net/mgbe_0/threaded` Adjust the interface number as needed (0–3). The Jetson Thor T4000 module supports three interfaces of the MGBE (0–2).

3.6. Display

The following display issues are noted in this release.

Issue	Description
5525468	When the DCE is suspended, if a wake event tries to interrupt the suspend process before the device completely enters the suspend state, the device can hang or cause a system crash because the DCE can't reset and resume normally.
5424504	UEFI menu is slow to respond during ISO installation on some 4K monitors. This issue does not affect the functionality.
5380828	Toggling fractional scaling options in display settings on desktop could lead to some flickering on the background.
5422032	In a dual-monitor setup, enabling desktop scaling might cause window corruption or visual artifacts because of a GNOME bug.

3.7. Compute Stack

The following issues related to various compute components are noted in this release.

Issue	Description

5092683	Fisheye Python samples fail to run because of OpenCV compatibility issues. To work around this issue, install the following:
	`sudo python3 -m pip install opencv-python==4.8.0.74;`
	`sudo python3 -m pip install pillow numpy==1.26;`

5228336 5462083	For details, refer to the VPI Release Note.

3.8. Multi-Instance GPU (MIG)

Note that MIG is a technology preview feature. Support for MIG is still a work in progress in the current release, and the following known issues related to this feature have been identified.

Issue	Description
6230230 6230231	MIG is currently not supported on Jetson T4000 or on the Jetson Orin Series of products.
6237160 6238288	MIG on Jetson AGX Thor is recommended only for profiles 78 and 83 with this initial release.
6162096	Output of `nvidia-smi` is incorrect when using MIG.

4. Fixed Issues

The following issues were fixed in this release.

Issue	Description

5656972	In some conditions, the following messages repeat in the kernel log:
	`[ 373.487744] nvethernet a808e10000.ethernet: [xpcs_lane_bring_up][808][type:0x4][loga-0x0] PCS block lock SUCCESS`
	`[ 374.543752] nvethernet a808e10000.ethernet: [xpcs_lane_bring_up][808][type:0x4][loga-0x0] PCS block lock SUCCESS`
	To stop them, bring down MGBEinterfaces that are not in use by using the following commands:
	`#sudo su`
	`#ifconfig mgbe0_0 down`
	`#ifconfig mgbe1_0 down`
	`#ifconfig mgbe2_0 down`
	`#ifconfig mgbe3_0 down`

5701579	Immediately after flashing, in case of any issues with connecting to 6 GHz in WPA3 mode, reset the device and try again.
5426982	Sometimes in a busy environment, not all Wi-Fi access points are observed due to limited buffer size for scan results. As a workaround, run `wpa_cli set bss_max_count 500` to increase the `wpa_supplicant` buffer size.

5748062	When building Real-Time Kernel with the sources, make the following changes to the `generic_rt_build.sh` script:
	`@@ -62,7 +62,10 @@ enable_rt()`
	`--enable PREEMPT_RT --disable DEBUG_PREEMPT\`
	`--disable KVM\`
	`--enable EMBEDDED\`
	`+ --enable EXPERT\`
	`--enable NAMESPACES\`
	`@@ -72,7 +75,10 @@ enable_rt()`
	`--enable PREEMPT_RT --disable DEBUG_PREEMPT\`
	`--disable KVM\`
	`--enable EMBEDDED\`
	`+ --enable EXPERT\`

Issue	Description
	`--enable NAMESPACES\` Without this change, the built kernel does not have `PREEMT_RT` enabled.
5763680	When a display is connected (regardless of resolution), attempting to set the EMC clock frequency to a value other than Fmax (for example, 4266 MHz) via `nvpmodel.service` during the systemd initialization stage can fail and result in a system crash. To mitigate this issue, always configure the EMC clock to Fmax in the `/etc/nvpmodel.conf` file. After the system has fully booted to the Ubuntu console, you can then apply the desired EMC clock cap manually. Use the following command to set the EMC clock cap: `$ echo <val> > /sys/class/devfreq/bwmgr/max_freq`
5367003	Suspend/resume cycle could sometimes lead to reboot of the unit.
5461056 5398396	Argus_crash can be seen while running preview/capture use cases for E3333[OV5693], E3331[IMX318], or IMX185, due to the opening sequence of streaming only. Use `override.cfg` files for the sensors to avoid this issue.
5461150	Failed to stream with AR1335. nvv4l2camerasrc: Internal data stream error AR1335 uses a discrete framerate not supported by nvv4l2camerasrc. Use v4l2src + nvvidconv to avoid the issue.
5436810	CUDA BL memory is not supported for encode and decode use cases.
5451745	De-interlace decode is not supported.
5437529	Slice-level encoding is not supported.
5236938	Init time delay for multi-instance decode use cases.
5437402	When two displays are connected to target one over HDMI and another over DP, the NVIDIA logo gets clipped during boot up and reboot.
5452012	GStreamer `nvsiplsrc` plug-in requires `nvvidconv` to copy the buffers because it does not work directly with the encoder.
5416070	DVFS is not fully optimal for multimedia use cases
5226667 5465140	Wi-Fi 6-GHz routers that use MBSSID are not supported.

Issue	Description
5456784	Using Jetson IO to configure Jetson Camera Hawk-Owl p3762 module could sometimes lead to the display being stuck.
5451920	GStreamer `nv3dsink` plug-in hangs in DRC use cases.
5461913	Image captures with IMX274 in WDR mode and SDR mode are blurry at times. This is due to tuning of noise parameters that is required in this mode. The issue is fixed with this release.
5504026 5470960	Issues with Multi HSB streaming are fixed. Can now support up to a maximum of 16 cameras (through 8x HSBs).
5445933 5472056 5446087	Intermittent failures could be observed when pinging cameras over Ethernet. When running SIPL example applications, random I2C errors and capture failures are seen because of UDDF driver header incompatibility. Beginning with this release, the following workaround is no longer needed. 1. `sudo cp /usr/lib/nvsipl_uddf/libnvuddf_eagle_library.so /tmp/libnvuddf_eagle_library.so_orig` 2. `cd /usr/src/jetson_sipl_api/sipl/` 3. `sudo mkdir build` 4. `cd build` 5. `sudo cmake ..` 6. `sudo make nvsipl_coe_camera` 7. `sudo cp ./samples/coe_camera/nvsipl_coe_camera ~/` 8. `sudo cp /tmp/libnvuddf_eagle_library.so_orig /usr/lib/nvsipl_uddf/libnvuddf_eagle_library.so` 9. Run the app `nvsipl_coe_camera`.
5455397	Fix segmentation fault in `nvsipl_query_sample`.

5454877	The `nvsipl_coe_camera` application fails to run inside a Docker container. Use the following workaround to fix the problem.
	
	In the file `/etc/nvidia-container-runtime/host-files-for-container.d/drivers.csv`, add the following lines:
	
	`lib, /usr/lib/nvsipl_uddf/libnvuddf_eagle_library.so`
	`lib, /usr/lib/nvsipl_drv/libnvsipl_qry_vb1940.so`
	`lib, /var/nvidia/nvcam/settings/sipl/vb1940.nito`

4755448	Fix PCIe completion timeout issues during the PTP operation.
5406304	Fix issues related to UART where standard baudrates were not supported.

Issue	Description
5440020	Bluetooth reconnections for LE keyboard and mice do not work on the initial pair. They will work after device reboot or Bluetooth reboot or toggling the connected device from settings page.
5699174	Hang is observed during video playback when using the GStreamer nv3dsink graphics plug-in with CUDA array memory type for the block-linear use case.
5561962	The multiplanar CUDA Array to NvRm Surface Array DMABUF block-linear interoperability test fails for smaller resolutions.
5466051	The NVIDIA FFMPEG HW acceleration is currently not supported on Jetson Thor. If your applications require FFMPEG, you can use the open source version with the following command: `sudo apt-get install -y ffmpeg`

5. Implementation Details

This section provides information about implementation details.

5.1. Camera

The following are the camera integration changes compared to previous Jetson Linux releases.

5.1.1. SIPL Framework for Camera Streaming

Jetson SIPL API Package 2.0.0 for JetPack 7.2 is a major Jetson SIPL API release. It introduces production GMSL driver support, unified camera sample applications, expanded UDDF interfaces, stereo pipeline support, and updated sensor/module configuration handling. This release contains breaking changes that affect API, ABI, JSON schema, package layout, and driver loading.

Release Highlights

- Added production GMSL support for MAX96712, MAX96724, MAX96717, MAX9295, IMX728, IMX623, AR0234, BR24H128, GT24C64D, MAX20087, NativeGpio, and CoSerDes modules including AR0234HAWK, R0SIM623, and R0SIM728.
- Added unified `nvsipl_camera` sample application for GMSL and CoE camera capture and display, including WFD support.
- Added unified `nvsipl_query` sample tool for hierarchical camera configuration queries, JSON output, and per-module or per-transport inspection.
- Added GMSL and CoE stereo pipeline support using CNvFSensorGroupPipeline.
- Added support for external FSYNC in GMSL stereo configurations.
- Added CoE stereo support for VB1940, including new VB1940_Stereo configuration.
- Added dynamic CoE sensor count support using vector-based sensor lists.
- Added 3-lane CSI support: CSI_AB3, CSI_CD3, CSI_EF3, and CSI_GH3.
- Added new pixel format support, including RAW12TP, 24 bpp, and RGB/BGR pixel orders.
- Added new notifications, power interrupt status codes, query methods, and ISP/autocontrol configuration options.

New Public APIs and Interfaces

- Added 19 UDDF DDI/CDI headers for GMSL component interfaces, GPIO, diagnostics, configuration objects, and sensor characterization mode.
- Added `FrameProcessingRate` enum for configurable AE, AWB, and ISP processing rates.
- Added query methods `GetModuleConfig()`, `GetTransportSettings()`, and `GetVersion()`.
- Added `GetIspOverrideSettings()` optional method in `INvSiplControlAuto`.
- Added `UserDefDTCaptureSetting` for user-defined datatype capture.
- Added `MAX_VIRTUAL_CHANNELS_PER_SENSOR` constant.
- Added WF display headers under `include/WF/`.
- Added relocated `capture/include/` directory.

Breaking Changes and Migration Notes

- Camera configuration types moved from `nvsipl::` to `nvsipl::sensorconfig::`.
- `CameraSystemConfig` has been renamed to `SensorSystemConfig`; related types were also renamed, including `CameraConfig` to `ModuleConfig`, `CoECamera` to `CoEModule`, and `GmslCamera` to `GmslModule`.
- `INvSIPLCameraQuery::GetCameraSystemConfig()` has been renamed to `GetSensorSystemConfig()`.
- `CameraDeviceInfoList` has been restructured from per-component lists to module-level lists.
- Deprecated `ImageMetaData` ISP statistics fields have been removed. Applications should use `INvSIPLISPStatCustomInterface::GetIspStatsInfo()`.
- Deprecated notifications `NOTIF_INFO_ISP_PROCESSING_DONE` and `NOTIF_INFO_ACP_PROCESSING_DONE` have been removed.
- `MAX_CAMERAMODULES_PER_BLOCK` increased from 4 to 8.
- UDDF headers were renamed, including `ISensorControl.hpp` to `ICameraSensorControl.hpp` and `HSLDynamicSequence.hpp` to `IHSLDynamicSequence.hpp`.
- SIPL package directory changed from `camera-public/sipl/` to `camera-public/sipl-public/`.
- Install paths changed: drivers now install to `/usr/lib/nvsipl_drv/`, and applications now install to `/usr/sbin/`.
- Build targets changed: `eagleAIO` is now `eagle`, and `samples` is now `all_drivers`.

- Sample apps changed: `nvsipl_coe_camera` is replaced by `nvsipl_camera`, and `nvsipl_coe_query_test` is replaced by `nvsipl_query`.

Important Driver and Configuration Changes

- `INvSIPLQuery::ParseDatabase()` must now be called before `ParseJsonFile()`. In JetPack 7.2, `ParseDatabase()` loads the built-in driver database from `/usr/lib/nvsipl_drv/libnvsipl_qry_*.so`, and `ParseJsonFile()` overlays custom IP, MAC, and configuration data. Code that only calls `ParseJsonFile()` must be updated.
- JSON module and transport `name` fields are now case-sensitive and must match driver-exported names.
- JSON virtual-channel `inputFormat` value `raw10packed` has been renamed to `raw10tp`.
- CoE stereo configuration no longer uses `cameraConfigs[].CoECamera.isStereo`; stereo is derived from `sensorInfo.sensorGroup > 0` and `transportSettings[].sync_sensors`.
- UDDF interface UUIDs were bumped for `IGmslDeserializer`, `IGmslModuleControl`, `IGmslSerializer`, and `ICoEBridgeControl`. JetPack 7.1 drivers must be rebuilt against JetPack 7.2 headers.
- ICoEBridgeControl.hpp changed from fixed-size sensor arrays to vector-based sensor lists.
- NOVA0 camera module has been renamed to `AR0234CS_HAWK`, with corresponding directory, JSON, class, library, and target name updates.
- HAWK AR0234 JSON configuration changed: `fsyncMode` value `osc_manual` is now `external`, new sensor fields are required, and `hsb_metadata_lines` was removed.

5.1.2. Camera Serial Interface

The following are changes to the camera serial interface.

Note: We recommend that all camera drivers be packages like Loadable Kernel Module (LKM) for JetPack 6 and later.

- **Sensor kernel drivers**

  Driver source code for the supported sensors are located in the `<TOP>/kernel/nvidia-oot/drivers/media/i2c/` directory. To obtain a complete understanding of the driver, examine this source file.

- **LKM**

  The camera and sensor drivers are loadable modules. The rebuilt sensor driver will be in `<OUT>/l4t-generic-release-aarch64/nvidia//kernel-noble/kernel-nvidia-oot/nvidia-oot/drivers/media/i2c/<camera>.ko`.

  On root file system, the loadable modules are located in the following directories:

  - Camera sensor driver: `/lib/modules/6.8.12-tegra/updates/drivers/media/i2c/`.
  - Camera driver: `/lib/modules/6.8.12-tegra/updates/drivers/media/platform/tegra/camera/`.
  - RTCPU driver: `/lib/modules/6.8.12-tegra/updates/drivers/platform/tegra/rtcpu/`.
  - Nvhost VI driver: `/lib/modules/6.8.12-tegra/updates/drivers/video/tegra/host/vi/`.
  - Nvhost CSI driver: `/lib/modules/6.8.12-tegra/updates/drivers/video/tegra/host/nvcsi/`.

- **Device registration**

  After driver development is complete, you must add the new device information to the system kernel device tree so it can be registered (instantiated) when the kernel boots. To register your device, use one of the following methods:

  - **Device-tree overlay**

    You need to create a device-tree overlay file to register the camera module. If your camera module has on-board EEPROM and is programmed with a valid camera ID, you can use that to apply the overlay for a specific camera module and update the device-tree entries with proper information at runtime. Using DTB overlays with EEPROM-ID allows a single system image to support multiple camera devices.

    To change camera modules, power down the device, replace the camera module, and then reboot. The new module works automatically.

  - **Create and apply a DTB overlay file**

    1. Add your `.dtsi` file to the camera configuration `.dtsi` file. For Jetson AGX Orin, the file is `tegra264-p3971-camera-modules.dtsi`.

    2. Set the status of your device tree nodes to `disabled`.

       ```
       imx274_cam0: imx274_a@1a {
              status = "disabled";
       };
       ```

3. Add the overlay information as fragments below to a new `.dts` file. You can also refer to the camera DTB overlay files provided with the current release: `<top>/hardware/nvidia/t264/nv-public/overlay/camera-overlay-file.dts`.

4. Update the .dts file with proper overlay information and a compatible string:

```
/{
    overlay-name = "Jetson Camera Dual-IMX274";
    jetson-header-name = "Jetson AGX CSI Connector";
    compatible = "nvidia,p3971-0089+p3834-0008",
"nvidia,p3971-0050+p3834-0005", "nvidia,p3971-0080+p3834-0008",
"nvidia,p3834-0008", "nvidia,tegra264";

    / {
        overlay-name = "Jetson Camera Dual-IMX274";
        jetson-header-name = "Jetson AGX CSI Connector";
        compatible = JETSON_COMPATIBLE;
        fragment@0 {
             target-path = "/";
             board_config {
                 ids = "LPRD-dual-imx274-002";
                 sw-modules = "kernel";
             };

        __overlay__ {
...
```

5. Compile the `.dts` file to generate a `.dtbo` file.

6. *Before you flash,* move the `.dtbo` file to `flash_folder/kernel/dtb/`.

7. Add the following line to the `<board>.conf` file, which is used to flash the device. For the Jetson AGX Orin board, the config file is `p3834-0008-p3971-0000.conf`.

```
OVERLAY_DTB_FILE="${OVERLAY_DTB_FILE},tegra264-p3971-camera-dual-i
mx274-overlay.dtbo"
```

5.1.3. Device Registration

After you complete driver development, you *must* add the new device's information to the system kernel device tree so it can be registered (instantiated) when the kernel boots. This section describes how to register a new device.

Before you begin, ensure that you obtain the kernel source files.

Because UEFI boot is enabled in this release, the plug-in manager is no longer supported. You must create a *device tree overlay* (DTB overlay or `.dtbo`) file to register the camera module.

If your camera module has an on-board EEPROM, and is programmed with a valid camera ID, at runtime, you can use the device tree overlay file to apply the overlay for a specific camera module and update the device tree entries with proper information. Using a device tree overlay with an EEPROM ID allows a system image to support multiple camera devices. To select a different camera, power down the device, replace the camera module, and reboot. The new module works automatically.

To create and apply a device tree overlay file:

1. Add the `.dtsi` file to the camera configuration `.dtsi` file.
2. Set the status of your device tree nodes to disabled.

   ```
   imx274_a@1a {
       status = "disabled";
   };
   ```

3. Add the overlay information as fragments to a new `.dts` file.

   ```
   <top>/hardware/nvidia/t264/nv-public/overlay/tegra264-p3971-camera-o
   verlay-file.dts
   ```

   You can also see the camera DTB overlay files that are provided with the current release for examples.

4. Update the .dts file with the correct overlay information and a compatible string.

   ```
   / {
            overlay-name = "Jetson Camera Dual-IMX274";
            jetson-header-name = "Jetson AGX CSI Connector";
            compatible = "nvidia,p3971-0089+p3834-0008";

   fragment@0 {
        target-path = "/";
        board_config {
            ids = "LPRD-dual-imx274-002" ;
            sw-modules = "kernel";
        };
         __overlay__ {
             status = "okay";
        };
   ```

```
                };
                fragment@1 {
                        . . .
                };

};
```

5. To generate a `.dtbo` file, compile the `.dts` file.
6. Before flashing, move the `.dtbo` file to `flash_folder/kernel/dtb/`.
7. Add the following line to the `<board>.conf` file, which is used to flash the device.

   ```
   OVERLAY_DTB_FILE="${OVERLAY_DTB_FILE},tegra264-p3971-camera-dual-imx274-overlay.dtbo";
   ```

   This line causes the following tasks to completed:

   - If a specific camera board is found when the kernel boots, the override data is applied to that camera board's tree nodes.
   - The tree nodes are made available for the system to use.

5.1.4. Camera Use Cases for CoE and GMSL and SIPL Example Applications

All Implementation details for Camera-over-Ethernet (CoE) use cases are available in CoE Camera Development Guide in the Jetson Linux Developer Guide.

5.2. Device Tree Change for nvethernet

In the previous release, JetPack 7.0, the restart lane bring-up logic was controlled through the device tree flag `nvidia,pcs-rx-eq-sw-ovrd`.

Beginning with JetPack 7.1, the flag is renamed `nvidia,force-restart-lane-bringup`. To maintain backward compatibility with older device trees, the driver now checks for both flags. If either flag is present, restart lane bring-up is executed.

The legacy flag `nvidia,pcs-rx-eq-sw-ovrd` is deprecated and will be removed in a later release. If applicable, update the device tree in the QSPI firmware to use `nvidia,force-restart-lane-bringup`.

Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. NVIDIA Corporation (“NVIDIA”) makes no representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Material (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice.

Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer (“Terms of Sale”). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer’s own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer’s sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer’s product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability related to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, “MATERIALS”) ARE BEING PROVIDED “AS IS.” NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA’s aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

Trademarks

NVIDIA, the NVIDIA logo, are trademarks and/or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

NVIDIA Corporation | 2788 San Tomas Expressway, Santa Clara, CA 95051
http://www.nvidia.com

VESA DisplayPort
DisplayPort and DisplayPort Compliance Logo, DisplayPort Compliance Logo for Dual-mode Sources, and DisplayPort Compliance Logo for Active Cables are trademarks owned by the Video Electronics Standards Association in the United States and other countries.

HDMI
HDMI, the HDMI logo, and High-Definition Multimedia Interface are trademarks or registered trademarks of HDMI Licensing LLC.

Arm
Arm, AMBA, and ARM Powered are registered trademarks of Arm Limited. Cortex, MPCore, and Mali are trademarks of Arm Limited. All other brands or product names are the property of their respective holders. ″Arm″ is used to represent ARM Holdings plc; its operating company Arm Limited; and the regional subsidiaries Arm Inc.; Arm KK; Arm Korea Limited.; Arm Taiwan Limited; Arm France SAS; Arm Consulting (Shanghai) Co. Ltd.; Arm Germany GmbH; Arm Embedded Technologies Pvt. Ltd.; Arm Norway, AS, and Arm Sweden AB.

OpenCL
OpenCL is a trademark of Apple Inc. used under license to the Khronos Group Inc.

Copyright
© 2026 NVIDIA Corporation & Affiliates. All rights reserved.

NVIDIA Corporation | 2788 San Tomas Expressway, Santa Clara, CA 95051
http://www.nvidia.com