



NVIDIA Jetson Linux

Release Notes

Version 36.3 GA

Table of Contents

1. About this Release	2
1.1. Login Credentials	3
1.2. What's New	3
2. Known Issues	5
2.1. General System Usability	5
2.2. Flashing	8
2.3. Camera	8
2.4. Multimedia	9
2.5. Display	9
2.6. Compute Stack	11
2.7. Deepstream	11
3. Fixed Issues	12
4. Implementation Details	14
4.1. Camera	14
4.2. Device Registration	16
4.2.1. Device Tree Overlay	16
4.2.2. Using the Jetson IO Tool	18
4.3. UEFI	18

1. About this Release

The NVIDIA® Jetson™ Linux 36.3 General Availability (GA) release includes the Linux Kernel 5.15, the Ubuntu 22.04-based root file system, the UEFI-based Bootloader, NVIDIA drivers, the necessary firmware, toolchain, and more. This release supports all Jetson Orin production modules and Developer Kits.



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

Platform and Release Information

Description	Supported Version
Host machine Linux distribution for flashing software onto Jetson devices	Ubuntu x64 20.04 or 22.04 (x64 distribution)
Sample rootfs derived from the Ubuntu operating system to run on Jetson devices.	Ubuntu 22.04
The supported Linux kernel version.	5.15 LTS
The supported ARM architecture.	aarch64

Description	Supported Version
<p>Name of the configuration file used in flashing.</p> <p>Note: When you flash a configuration file with the flash.sh script, specify the configuration's basename, for example, the file name without the .conf suffix. Some of the products require flashing through initrd instead of flash.sh</p> <p>For a complete description of supported platforms and configuration names, see the <i>Jetson Modules and Configurations</i> table in Environment Variables.</p>	<p><i>jetson-agx-orin-devkit-industrial.conf</i>: Flashes the Jetson Orin industrial module (P3701-0008), which is attached to a Jetson Orin reference carrier board (P3737-0000).</p> <p><i>jetson-orin-nano-devkit.conf</i>: Flashes one of the following modules that is attached to a Jetson Orin Nano Carrier board (P3768-0000):</p> <ul style="list-style-type: none"> • Jetson Orin Nano developer kit module with SD Card (P3767-0005) • Jetson Orin Nano 8GB module (P3767-0003) • Jetson Orin Nano 4GB module (P3767-0004) • Jetson Orin NX 16GB module (P3767-0000) • Jetson Orin NX 8GB module (P3767-0001) <p><i>jetson-agx-orin-devkit.conf</i>: Flashes one of the following modules that is attached to a Jetson AGX Orin Developer Kit (P3730-0000) reference carrier board (P3737-0000):</p> <ul style="list-style-type: none"> • Jetson AGX Orin developer kit module (P3701-0000) • Jetson AGX Orin 32GB module (P3701-0004) • Jetson AGX Orin 64GB module (P3701-0005)
Board names, module names, and revision numbers	Refer to the Jetson FAQ for a detailed list of Jetson device information.
Toolchain for cross-compilation	<p>Bootlin GCC 11.3</p> <p>https://developer.nvidia.com/embedded/jetson-linux</p>
Release Tag	jetson_36.3

1.1. Login Credentials

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

1.2. What's New

Jetson Linux 36.3 is the first production quality Jetson Linux release that supports NVIDIA JetPack™ 6. Here are the highlights for this release:

- Production-ready software stack

- Preview of [upgradable compute stack](#) to upgrade CUDA, Tensorrt, CUDNN, DLA and VPI without upgrading Jetson Linux. Refer to JetPack documentation for running compute stack from this production release on JetPack 6 Developer Preview BSP.
- Enabled PREEMPT_RT in the pre-built kernel Debian package.
- Added support for [upgrading Jetson Orin Nano Developer Kit](#) from JetPack 5 to JetPack 6 without needing a host machine.
- Over-The-Air (OTA) Update
 - Added support for the [OTA tool](#) to upgrade NVIDIA Jetson Orin based devices from JetPack 5.
- Security
 - Enabled support for [security](#) features, bringing parity with JetPack 5.
- Power and Performance
 - Enabled support for [power](#) features, bringing parity with JetPack 5.
 - Added support for the [Power Estimator](#) tool in JetPack 6.
- Display
 - Added support for framebuffer console.
 - Added support for suspend/resume in Display Core Engine (DCE).
- Multimedia
 - Added support for H.264 Constrained_Baseline and Constrained_High profiles.
 - Added support for Dynamic Bitrate Rate and Dynamic FrameRate in the AV1 encoder.
 - Added support for Dynamic Resolution Change (DRC) in H.264, H.265 and AV1 encoders.
 - Added video transform support for UYVY BT.709 and BT.2020 colorimetry.

Here is some additional information:

- Jetson Linux Sources are available on Git in addition to the Jetson Linux page (refer to [Working with Sources](#) for more information).

For more information about the adaptation and bring up process for your custom carrier boards, refer to [Jetson Module Adaptation and Bringup](#) for the Jetson AGX Orin, Orin NX, and Orin Nano platforms.

- Refer to the [Jetson Linux Developer Guide](#) for Jetson Linux documentation and [Implementation Details](#) for more information about implementation details that cover a variety of topics.

2. Known Issues

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

2.1. General System Usability

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

Issue	Description
4185596	Jetson AGX Orin Developer Kit and Jetson AGX Industrial modules might intermittently fail to resume after suspend.
4323950	<p>All Jetson devices sometimes see an intermittent TLP packet Malform issue with the rt18822ce WiFi driver when you see the following messages in the error log:</p> <pre>[163.849288] rt188x2ce 0001:01:00.0: AER: can't recover (no error_detected callback)</pre> <p>When this issue occurs, the WiFi will not be functional and ethernet might also get dropped. To recover from this issue, reboot the device.</p>
4191812	<p>On NVIDIA IGX, when the bandwidth request from Display/VI comes before the nvpmode EMC clock capping request, the request from VI/Display takes precedence because the VI/Display cannot handle the latency of a DRAM switch without a fatal error.</p> <p>When the issue occurs, the EMC clock is uncapped and power might go high. You need to select the nvpmode power mode with the appropriate EMC clock setting.</p>
4480028	<p>With some USB cables that are shipped with Jetson AGX Orin, you might see the following errors in the logs:</p> <pre>usb usb2-port2: Cannot enable. Maybe the USB cable is bad? usb usb2-port2: config error</pre> <p>To stop the error messages, swap the USB cables.</p>
4617111 4620917	The TSEC CBB errors might be observed on Jetson AGX Orin during bootup, shutdown, suspend, and resume when the target is connected to a display with HDCP 2.2 support. This issue is the result of the missing synchronization between Tsec and DCE/display engines. The HDCP can be disabled by following patch which prevents the CBB errors.

Issue	Description
	<pre> --- a/nv-soc/tegra234-soc-display-overlay.dtsi +++ b/nv-soc/tegra234-soc-display-overlay.dtsi @@ -176,7 +176,6 @@ "dpaux0_reset", "dsi_core_reset", "mipi_cal_reset"; - hdcp_enabled; status = "disabled"; memory-region = <fb0_reserved>; nvidia,disp-sw-soc-chip-id = <0x2350>; </pre>
4435272	<p>The GPIO sysfs node has been deprecated, and /sys/class/gpio cannot be used to control the GPIO state.</p> <p>To use GPIO from the userspace, use the new GPIO character device API (for example, libgpiod).</p>
4201479	<p>Although we support flashing Jetson-Linux BSP in multiple boot media like USB and NVMe, the media should have the same version of BSP. An attempt to flash different BSP versions to multiple boot media will lead to a system crash in the UEFI because different overlays are flashed in the UEFI partition.</p>
4185596	<p>Waking up from Deep Sleep state (SC7) by USB events is not supported in the NVIDIA JetPack 6.0 GA release for the Jetson Orin Series of products. This functionality will be added in a future release.</p>
4450559	<p>On the Jetson AGX Orin series of products, enabling the Wake-On-Lan (WOL) feature is not recommended when users connect to these products using SSH or when the target is booting from NFS.</p> <p>In both cases, users will lose connection with the target.</p>
4414147	<p>Suspending the resume cycle on Jetson AGX Orin with USB-C devices connected might lead to kernel exceptions.</p> <p>This issue will be fixed in the next release.</p>
4412365	<p>On Jetson AGX Orin with external NVMe connected flashed using SDK Manager could see the following message during debian update process:</p> <p>FuPluginUefiCapsule cannot find default ESP: More than one available ESP WARNING: UEFI ESP partition not detected or configured</p>

Issue	Description
	<p>See https://github.com/fwupd/fwupd/wiki/PluginFlag:esp-not-found for more information.</p> <p>To recover from this issue :</p> <ol style="list-style-type: none"> 1. Stop the Debian update. 2. Run the following commands: <pre>\$ sudo mkdir /boot/efi \$ sudo mount /dev/nvme0n1p10 /boot/efi \$ sudo vim /etc/fwupd/uefi_capsule.conf change the OverrideESPMountPoint=/boot/efi</pre> 3. To trigger the OTA again, run the following command. <pre>\$ sudo apt dist-upgrade</pre>

2.2. Flashing

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

Issue	Description
4357750	<p>On the Ubuntu 18.04 Linux host, the <code>sudo ./apply_binaries.sh</code> installation step shows the following messages:</p> <p>qemu: Unsupported syscall: 293</p> <p>Reinstalling the <code>qemu-user-static</code> package on the Linux host sometimes eliminates the messages. These messages are typically harmless and everything still functions properly.</p>
4229251	<p>On some Linux hosts during flashing, the following message appears in <code>dmesg</code> logs, followed by flashing failures:</p> <p>Cannot enable. Maybe the USB cable is bad?</p> <p>To work around the issue, try connecting to a different USB port on your host machine. If this step does not help, change the USB cable or reboot the host machine.</p>

2.3. Camera

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

Issue	Description
4264284	<p>On Jetson AGX Orin, images captured using argus cameras with DOL HDR Sensors might show a marginal increase in noise.</p> <p>A fix for this issue will be provided in the next release.</p>
3692128	<p>The following issues were observed on E3331 (IMx318 - CPHY sensor) on Jetson AGX Orin:</p> <ul style="list-style-type: none">• E3331 (IMX318-CPHY) sensor fails to stream by using the <code>v4l2-ctl</code> command.• Streaming using <code>argus_camera</code> and sample apps works, but it might intermittently output corrupted frames.

Issue	Description
3539001	On Jetson Orin AGX,the getSharpnessScore () Argus API, which provides the focuser stats information, is not supported for this release.
4510846	The initial probe might fail for one of the hawk sensors that are connected through GMSL because the v4l2 video node (/dev/video*) failed to register.To workaround the issue, try to insmod the nv_hawk_owl driver after the boot completes.
4588292	<p>With this release, cam i2cbus-8 runs at a default frequency of 100khz as compared to previous JetPack 5.x release where it was set to 400khz. This will result in the slow programming of secondary devices that are connected to i2cbus-8.</p> <p>We recommend that you create an overlay with the i2c-frequency change for i2cbus-8 to 400khz.</p>

2.4. Multimedia

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

Issue	Description
4146738	The deInterlace feature is not supported in this release.
4401805	<p>Starting a video_drc_drm with the disable-video option leaves the rendered in a bad state.</p> <p>To resolve this issue, reboot the system.</p>

2.5. Display

The following Display issues are noted in this release.

Issue	Description
4385047	Hotplugging the display using DP to Jetson AGX Orin after the bootup might lead to a corrupted screen.

Issue	Description
4324714	The secondary display that is connected to Jetson AGX Orin and Jetson Orin NX/Nano shows as being connected on the xrandr but gdm is not rendered on the screen.
4618846 4618626	The Cablecreation branded DP-to-HDMI converter does not work as expected with Jetson AGX Orin. We recommend that you use a different DP-to-HDMI converter.

2.6. Compute Stack

The following Deep Learning-related issues are noted in this release.

Issue	Description
4564075	To run VPI PVA algorithms in a docker container, the same VPI version has to be installed on the docker host.

2.7. Deepstream

Issue	Description
4325898	The pipeline gets stuck for multiframe_src when using <code>nvv4l2decoder</code> . DS developers use the pipeline to run decode and infer jpeg images.

3. Fixed Issues

This section provides details about issues that were noted as issues in the previous releases but fixed in the current release.

Issue	Description
4385287	<p>When only one IMX219 is connected to Jetson Orin NX 16GB/8GB, and a dual IMX219 overlay is applied, the preview and capture with <code>argus_samples</code> and <code>argus_camera</code> fails.</p> <p>To work around this issue, connect the dual IMX219 cameras when applying dual overlay support.</p>
4327644	<p>When an IMX219 or an IMX477 camera is connected to Jetson Orin NX and Jetson Orin Nano devices through the CAM0 interface, to correctly complete a camera recording, users need to explicitly specify <code>sensor-id=1</code> instead of the default <code>sensor-id=0</code> in the <code>gststreamer nvarguscamera</code> element.</p>
4389380	<p>A half-preview image of IMX477 is observed when you run <code>argus_camera</code> for Orin Nano 4GB SKU4.</p>
4389112	<p>Multimedia sample compilation fails due to a missing header file, and the following message is displayed:</p> <p>NvJpegDecoder.cpp:36:10: fatal error: jpegint.h: No such file or directory</p> <p>This issue will be fixed by using apt update after the release.</p>
4359668	<p>Applying the 3840x2160@30/24Hz resolution on a display that is connected to Jetson Orin NX makes the display go blank and the following error message is displayed:</p> <p>No VSI InfoFrame exists on two video fields.</p>
4349663	<p>When building TensorRT engines for DLA, there is a known issue where the entire DLA subgraphs that are listed in <i>Layers Running on DLA</i> (seen with TensorRT's verbose mode) cannot be built and eventually falls back to the GPU with the following message:</p> <p>{ForeignNode[...] } cannot be compiled by DLA, falling back to GPU.</p> <p>This has been observed with the following ResNet-based models:</p> <ul style="list-style-type: none">• PeopleNet v2.6• TrafficCamNet from TAO. <p>In both cases, to fix this issue, change TensorRT's default DLA SRAM pool size from 1 MiB to 0.5 MiB. When using <code>trtexec</code> to build the TensorRT engine, add the</p>

Issue	Description
	<p>--memPoolSize=dlaSRAM:0.5 argument. For other TensorRT applications that directly call TensorRT APIs, refer to this code section in trtexec.</p>
4391619	<p>Resnet-10 batch_size=1 GPU-only inferencing is broken on Jetson AGX Orin.</p> <p>Currently, syncpoint submission is updated to the end of cudaEventRecord() after all the operations have finished, so that after syncpoint is complete, there should be no pending work on the marker.</p> <p>However, when the application thread invokes cudaEventElapsedTime, but the GPU is still busy and has not yet completed its work, it will fail.</p> <p>To work around this issue and avoid the application execution failure, invoke the jetson_clocks utility to max out the SOC clocks and speed up the execution. This workaround only improves the symptom and is not guaranteed to work every time.</p>
4361621	<p>DRC support in gst pipelines is a work in progress and will require some changes in the Video Codec.</p>
4297071	<p>The new Gstreamer version might have some issues with the RTP stack and might lead to intermittent segmentation faults.</p>

4. Implementation Details

This section provides information about implementation details.

4.1. Camera

Here are the camera integration changes compared to previous Jetson Linux 35.x releases:



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 the `<OUT>/l4t-generic-release-aarch64/nvidia/kernel-jammy-src/kernel-nvidia-oot/nvidia-oot/drivers/media/i2c/<camera>.ko` directory.
- **On root file system, the loadable modules are located in the following directories:**
 - **Camera sensor driver:**
`/lib/modules/5.15.116-release-tegra/extra/drivers/media/i2c/.`
 - **Camera driver:**
`/lib/modules/5.15.116-release-tegra/extra/drivers/media/platform/tegra/camera.`
 - **RTCPU driver:**
`/lib/modules/5.15.116-release-tegra/extra/extra/drivers/platform/tegra/rtcpu/.`
 - **Nvhost VI driver:**
`/lib/modules/5.15.116-release-tegra/extra/drivers/video/tegra/host/vi/.`
 - **Nvhost CSI driver:**
`/lib/modules/5.15.116-release-tegra/extra/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.

tegra234-p3737-camera-modules.dtsi is for Jetson AGX-orin.

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/t23x/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,p3737-0000+p3701-0000",
        "nvidia,p3737-0000+p3701-0004", "nvidia,p3737-0000+p3701-0005",
        "nvidia,p3737-0000+p3701-0008";

    fragment@0 {
        target-path = "/bus@0/i2c@3180000/tca9546@70/i2c@0/imx274_a@1a";
        board_config {
            ids = "LPRD-dual-imx274-002";
            sw-modules = "kernel";
        };
        __overlay__ {
            status = "okay";
        };
    };

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

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 lines to the <board>.conf file, which is used to flash the device. For Jetson AGX-orin board, the config file is p3737-0000-p3701-0000.conf.
OVERLAY_DTB_FILE="\${OVERLAY_DTB_FILE}, tegra234-p3737-camera-dual-imx274-overlay.dtbo".

4.2. Device Registration

After you complete the 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. The following sections describe ways to register a new device.

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

4.2.1. Device Tree Overlay

Because UEFI boot is enabled in this release, the plugin 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.

```
imx185_cam0: imx185_a@1a {  
    status = "disabled";  
};
```

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

```
<top>/hardware/nvidia/platform/t19x/common/kernel-dts/t19x-common-modules/tegra194-camera-overlay-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 Xavier CSI Connector";
    compatible = "nvidia,p2822-0000+p2888-0001";

    fragment@0 {
        target= "<&imx185_cam0>";
        board_config {
            ids = "LPRD-dual-imx274-002" ;
            sw-modules = "kernel";
        };
        __overlay__ {
            status = "okay";
        };
        fragment@1 {
            . . .
        };
    };
};
```

1. To generate a .dtbo file, compile the .dts file.
2. **Before flashing**, move the .dtbo file to flash_folder/kernel/dtb/.
3. Add the following line to the <board>.conf file, which is used to flash the device.

```
OVERLAY_DTB_FILE="${OVERLAY_DTB_FILE},tegra 194-camera-overlay-file.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.

4.2.2. Using the Jetson IO Tool

If your camera module does not have an on-board EEPROM, you can use the same DTB overlay file to statically configure the board for the attached camera.

1. After you attach the camera module, apply the camera module's DTB overlay using the Jetson-IO tool, and reboot.

The new module will work immediately after Jetson Linux starts.

Note: You might have to delete the `board_config{}` node from the fragments in the DTB overlay file.

2. After you compile the `.dts` file to generate a `.dtbo` file, move the `.dtbo` file to `/boot` on the Jetson device, so that the Jetson-IO tool can recognize it.
3. Launch the Jetson-IO tool and configure the DTB overlay.

4.3. UEFI

For fixes that were made in the UEFI sources after the release, go to [UEFI GitHub](#).

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.

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

© 2024 NVIDIA Corporation & Affiliates. All rights reserved.

