Installation of RedHawk™ 9.4 Gold on NVIDIA® Jetson AGX Orin series, Orin NX series, and Orin Nano series systems

Release Notes

November 20th, 2024

0898003-9.4-Orin

READ ME BEFORE INSTALLING THIS PRODUCT

concurrent
REAL-TIME

This page intentionally left blank

1. Introduction

This document describes the process of installing the ARM64 version of RedHawk Linux™ 9.4 Gold on the Jetson AGX Orin, Orin NX, and Orin Nano. The directions in this document supersede all others – they are specific to installing the software on Concurrent Real-Time’s RedHawk systems.

2. Requirements

- Jetson AGX Orin, Orin NX, or Orin Nano with Internet access
- An x86_64 host system running Ubuntu 22.04 with Internet access; note that the host system should be up to date with the latest available Ubuntu package updates
- The *RedHawk 9.4 Gold for Jetson AGX Orin, Orin NX, and Orin Nano* optical media disc

3. Installation

3.1. Download SDK Manager onto the host system

On an x86_64 host system running Ubuntu 22.04, download version 2.2.0-12021 of SDK Manager from the NVIDIA website via this URL:

https://developer.nvidia.com/jetpack

Click on the NVIDIA SDK Manager link to download the SDK Manager package (login may be required).

3.2. Install the downloaded SDK Manager package

Open a terminal window and issue the following commands to install the SDK Manager:

```
$ cd ~/Downloads
$ sudo apt install ./sdkmanager_2.2.0-12021_amd64.deb
```

3.3. Connect the Orin device to the host system

The Orin must be connected to the host system for SDK Manager to be able to detect it and successfully flash it. Two different USB cables are required to connect the Orin to the host system. Refer to the cables and documentation included with your Orin.

3.4. Put the Orin device into recovery mode

Newer versions of SDK Manager work best when the Orin is initially placed into recovery mode. For the AGX Orin, if it is powered on and booted, issue the following command to enter recovery mode:

```
reboot --force forced-recovery
```

Alternatively, refer to the NVIDIA documentation for manual instructions to put your specific Orin device into recovery mode. In recovery mode, running `lsusb` on the host will show an `NVIDIA Corp.` entry specific to one of the following devices:

`0955:7023` (AGX Orin 64 GB)	`0955:7323` (Orin NX 16 GB)	`0955:7523` (Orin Nano 8 GB)
`0955:7223` (AGX Orin 32 GB)	`0955:7423` (Orin NX 8 GB)	`0955:7623` (Orin Nano 4 GB)

This indicates the Orin is properly in recovery mode.

3.5. Use SDK Manager to flash the Orin device

Invoke the following command as the current user to begin using the SDK Manager:

```
$ sdkmanager
```

The first action required by the SDK Manager is to log into the nvidia.com website using your developer or partner account email address and password. Enter the appropriate information and press LOGIN to continue.

NOTE

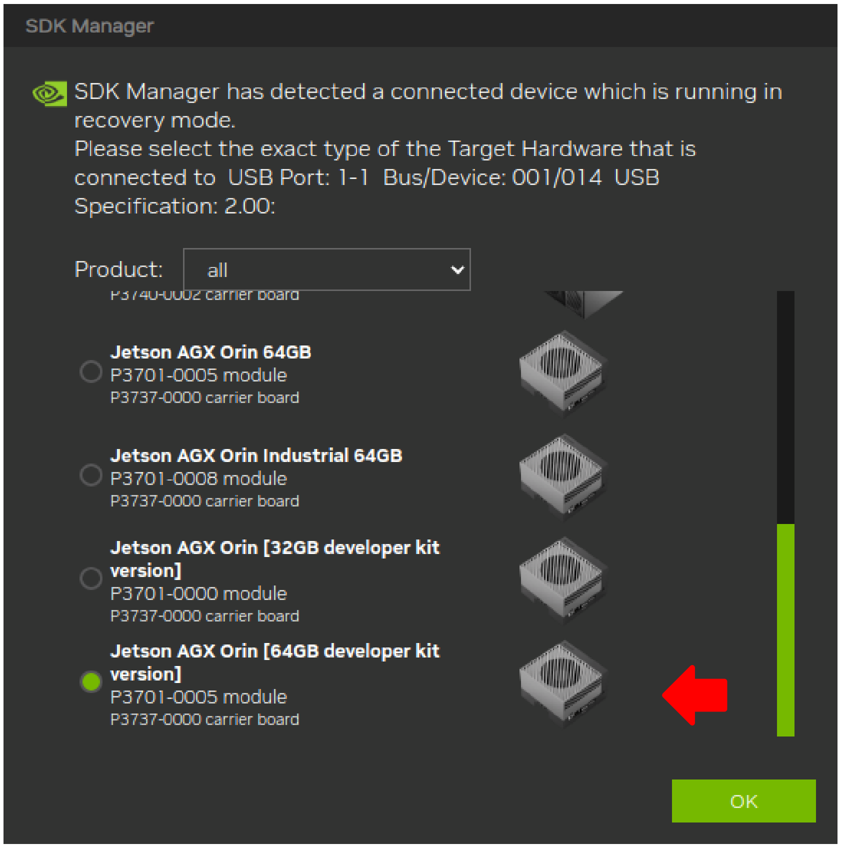
RedHawk 9.4 has only been validated with JetPack version 6.1.
Other versions are incompatible with RedHawk 9.4 and must not be used.

At this point, you can generally follow the instructions in the *Jetson AGX Orin Developer Kit User Guide* along with the following RedHawk-specific instructions.

NOTE

Screenshots for the AGX Orin 64 GB Developer Kit are provided below, but the installation process outlined here also applies to Orin NX and Nano devices.

1. The following dialog will immediately appear on the screen if the Orin is connected and properly placed into recovery mode at the time when the SDK Manager is first invoked:



SDK Manager

SDK Manager has detected a connected device which is running in recovery mode.
Please select the exact type of the Target Hardware that is connected to USB Port: 1-1 Bus/Device: 001/014 USB Specification: 2.00:

Product: all

P3740-0002 carrier board

○ **Jetson AGX Orin 64GB**
P3701-0005 module
P3737-0000 carrier board

○ **Jetson AGX Orin Industrial 64GB**
P3701-0008 module
P3737-0000 carrier board

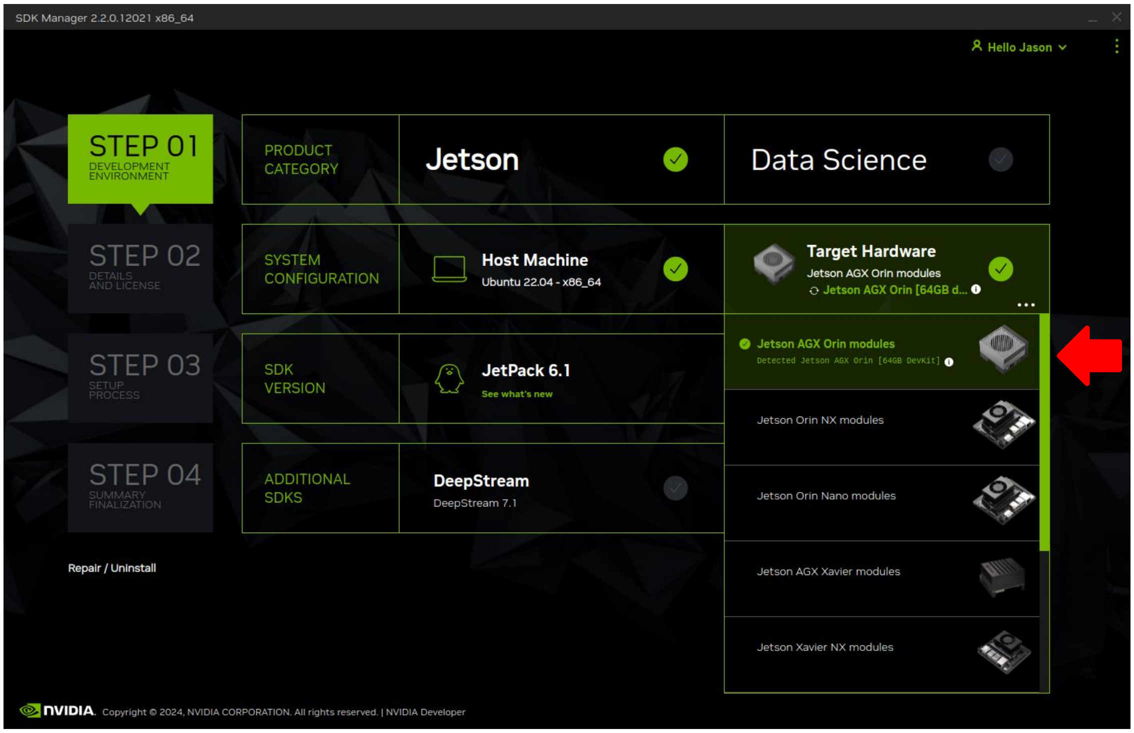
○ **Jetson AGX Orin [32GB developer kit version]**
P3701-0000 module
P3737-0000 carrier board

● **Jetson AGX Orin [64GB developer kit version]**
P3701-0005 module
P3737-0000 carrier board

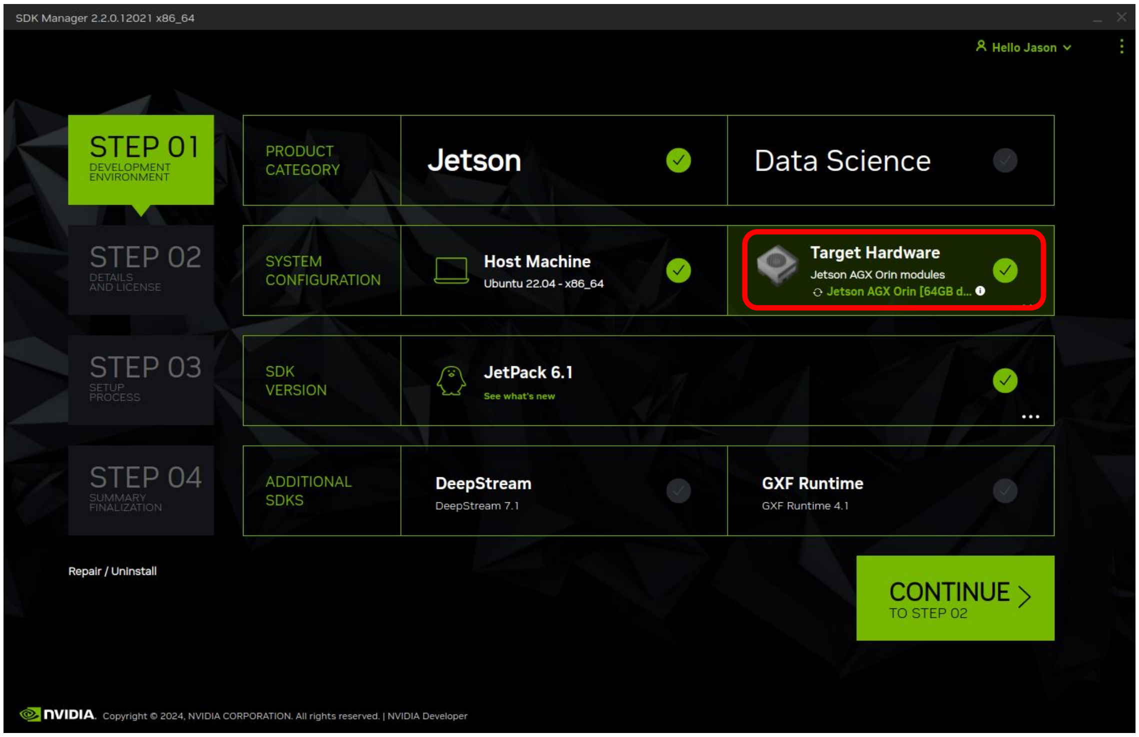
Choose Jetson AGX Orin (64GB developer kit version) and press OK.

If the Orin was not detected by the SDK Manager, proceed to the next section to select the hardware model you are installing.

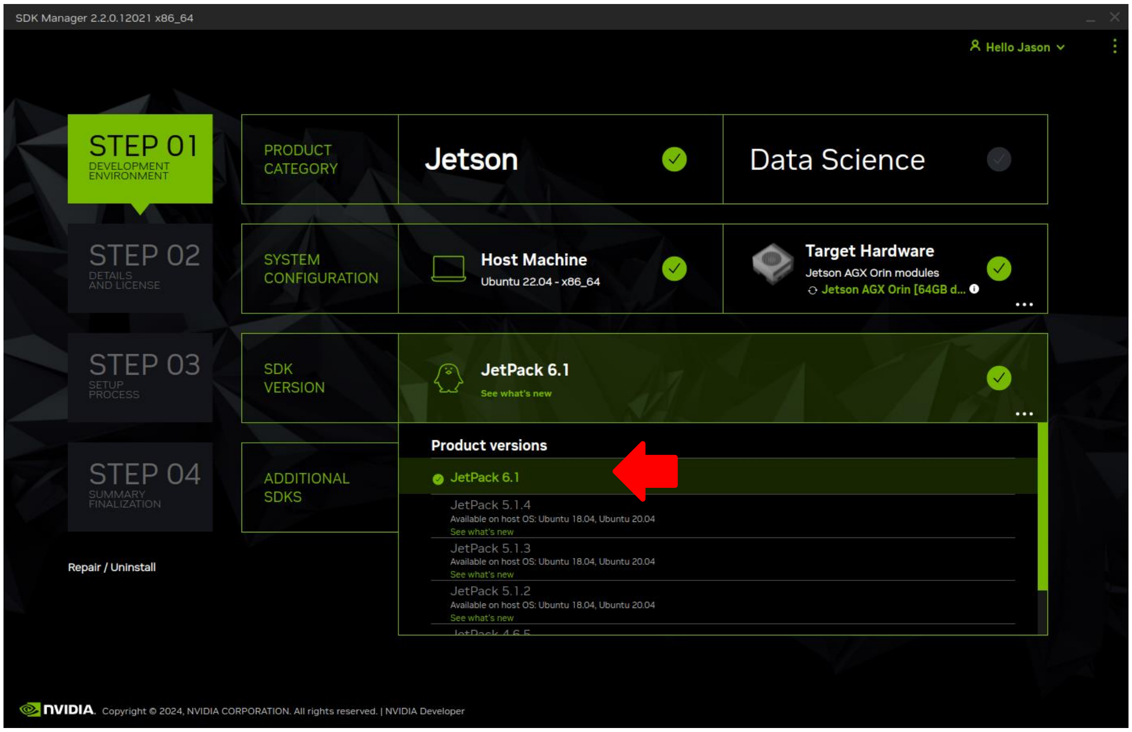
2. On the STEP 01 - DEVELOPMENT ENVIRONMENT screen, click on the . . . icon in the Target Hardware area to reveal the list of supported hardware configurations, as shown in the following screenshot:



3. Click on the Jetson AGX Orin modules target hardware option to make it the selected target, as shown in the following screenshot:



4. Click on the . . . in the JetPack area to reveal the list of supported JetPack versions, as shown in the following screenshot:



PRODUCT CATEGORY: Jetson ✓

SYSTEM CONFIGURATION: Host Machine Ubuntu 22.04 - x86_64 ✓ | Target Hardware Jetson AGX Orin modules Jetson AGX Orin [64GB d... ✓

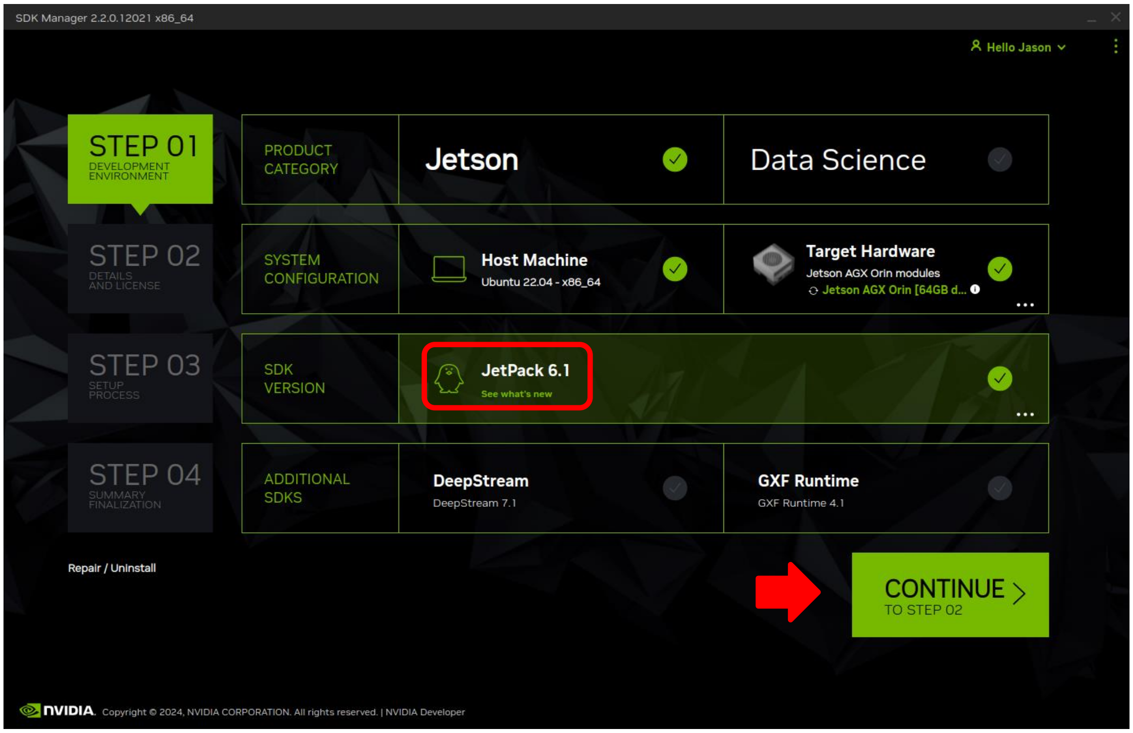
SDK VERSION: JetPack 6.1 ✓

Product versions

- ✓ JetPack 6.1
- JetPack 5.1.4 Available on host OS: Ubuntu 18.04, Ubuntu 20.04 See what's new
- JetPack 5.1.3 Available on host OS: Ubuntu 18.04, Ubuntu 20.04 See what's new
- JetPack 5.1.2 Available on host OS: Ubuntu 18.04, Ubuntu 20.04 See what's new
- JetPack 4.6.5

NVIDIA. Copyright © 2024, NVIDIA CORPORATION. All rights reserved. | NVIDIA Developer

5. Click on the JetPack 6.1 version to make it the selected version, as shown in the following screenshot:



PRODUCT CATEGORY: Jetson ✓

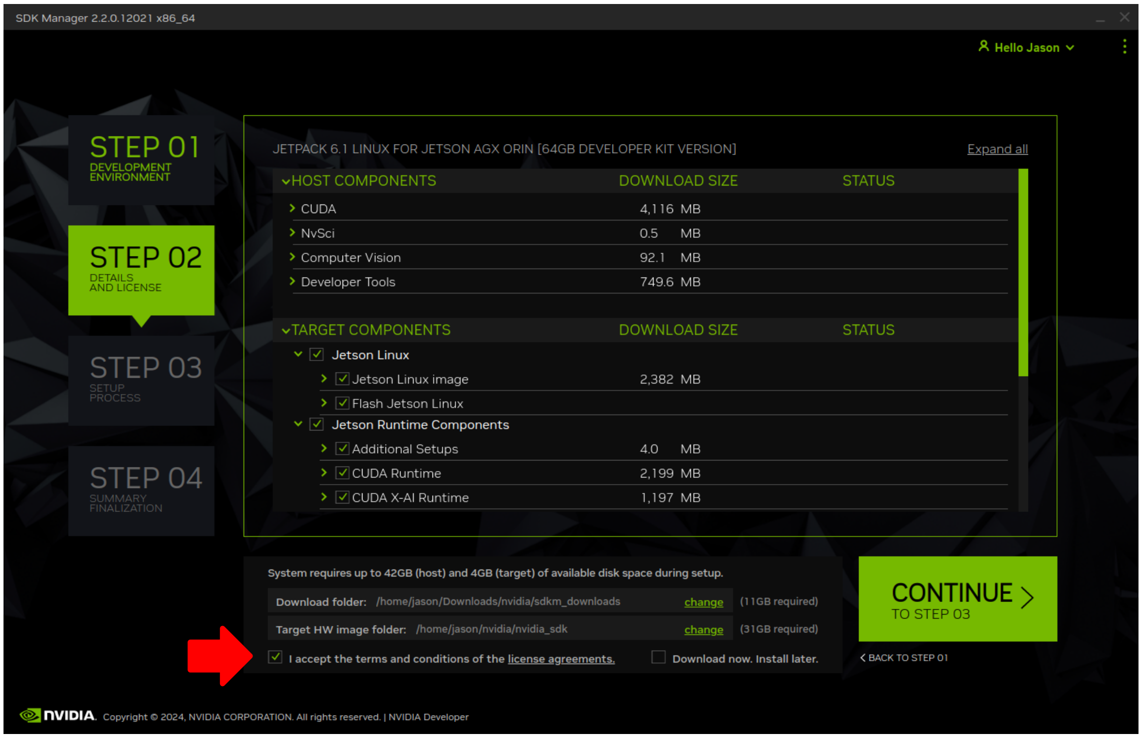
SYSTEM CONFIGURATION: Host Machine Ubuntu 22.04 - x86_64 ✓ | Target Hardware Jetson AGX Orin modules Jetson AGX Orin [64GB d... ✓

SDK VERSION: JetPack 6.1 ✓

ADDITIONAL SDKS: DeepStream DeepStream 7.1 | GXF Runtime GXF Runtime 4.1

NVIDIA. Copyright © 2024, NVIDIA CORPORATION. All rights reserved. | NVIDIA Developer

6. Press Continue to advance to the STEP 02 – DETAILS AND LICENSE screen and click I accept the terms and conditions of the license agreements at the bottom of the window to accept the licenses, as shown in the following screenshot:


SDK Manager 2.2.0.12021 x86_64
Hello Jason
STEP 01 DEVELOPMENT ENVIRONMENT
STEP 02 DETAILS AND LICENSE
STEP 03 SETUP PROCESS
STEP 04 SUMMARY FINALIZATION
Expand all
CONTINUE TO STEP 03
BACK TO STEP 01

JETPACK 6.1 LINUX FOR JETSON AGX ORIN [64GB DEVELOPER KIT VERSION]

HOST COMPONENTS	DOWNLOAD SIZE	STATUS
CUDA	4,116 MB	
NvSci	0.5 MB	
Computer Vision	92.1 MB	
Developer Tools	749.6 MB	

TARGET COMPONENTS	DOWNLOAD SIZE	STATUS
☑ Jetson Linux		
☑ Jetson Linux image	2,382 MB	
☑ Flash Jetson Linux		
☑ Jetson Runtime Components		
☑ Additional Setups	4.0 MB	
☑ CUDA Runtime	2,199 MB	
☑ CUDA X-AI Runtime	1,197 MB	

System requires up to 42GB (host) and 4GB (target) of available disk space during setup.

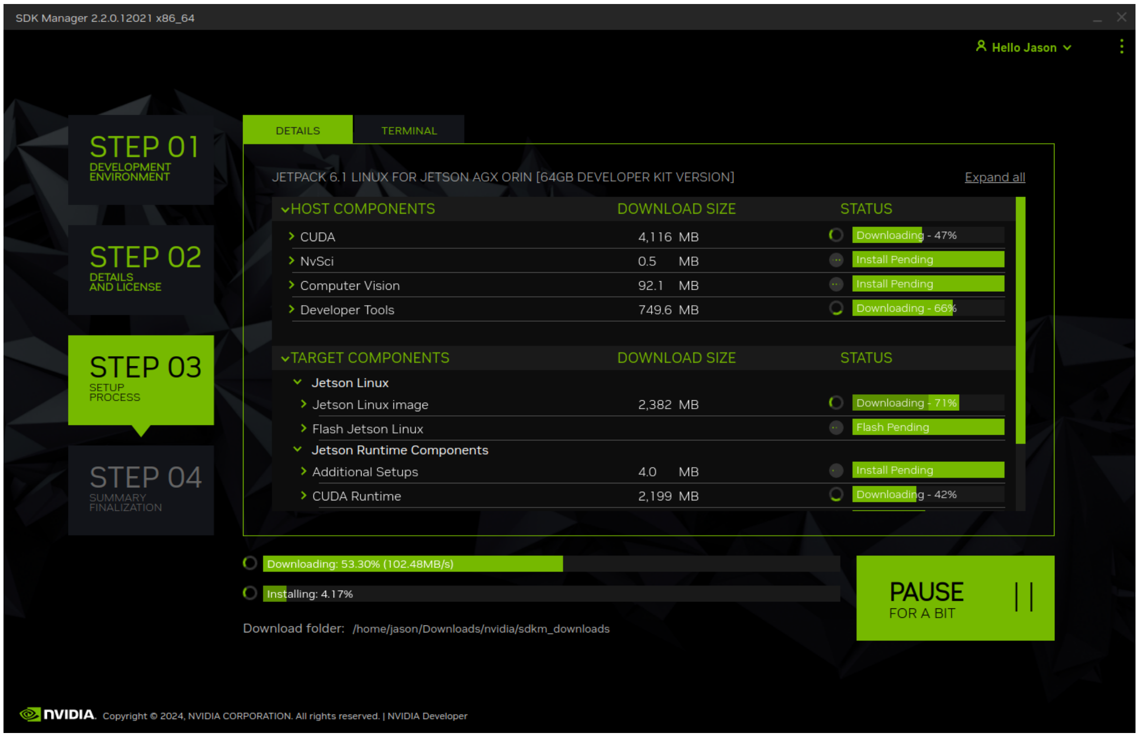
Download folder: /home/jason/Downloads/nvidia/sdkm_downloads change (11GB required)

Target HW image folder: /home/jason/nvidia/nvidia_sdk change (31GB required)

☑ I accept the terms and conditions of the license agreements. ☐ Download now. Install later.

NVIDIA. Copyright © 2024, NVIDIA CORPORATION. All rights reserved. | NVIDIA Developer

7. Press Continue to advance to the STEP 03 – SETUP PROCESS screen and SDK Manager will begin downloading all the software components from NVIDIA, as shown below:


SDK Manager 2.2.0.12021 x86_64
Hello Jason
STEP 01 DEVELOPMENT ENVIRONMENT
STEP 02 DETAILS AND LICENSE
STEP 03 SETUP PROCESS
STEP 04 SUMMARY FINALIZATION
DETAILS
TERMINAL
Expand all
PAUSE FOR A BIT

JETPACK 6.1 LINUX FOR JETSON AGX ORIN [64GB DEVELOPER KIT VERSION]

HOST COMPONENTS	DOWNLOAD SIZE	STATUS
CUDA	4,116 MB	Downloading - 47%
NvSci	0.5 MB	Install Pending
Computer Vision	92.1 MB	Install Pending
Developer Tools	749.6 MB	Downloading - 66%

TARGET COMPONENTS	DOWNLOAD SIZE	STATUS
Jetson Linux		
Jetson Linux image	2,382 MB	Downloading - 71%
Flash Jetson Linux		Flash Pending
Jetson Runtime Components		
Additional Setups	4.0 MB	Install Pending
CUDA Runtime	2,199 MB	Downloading - 42%

Downloading: 53.30% (102.48MB/s)

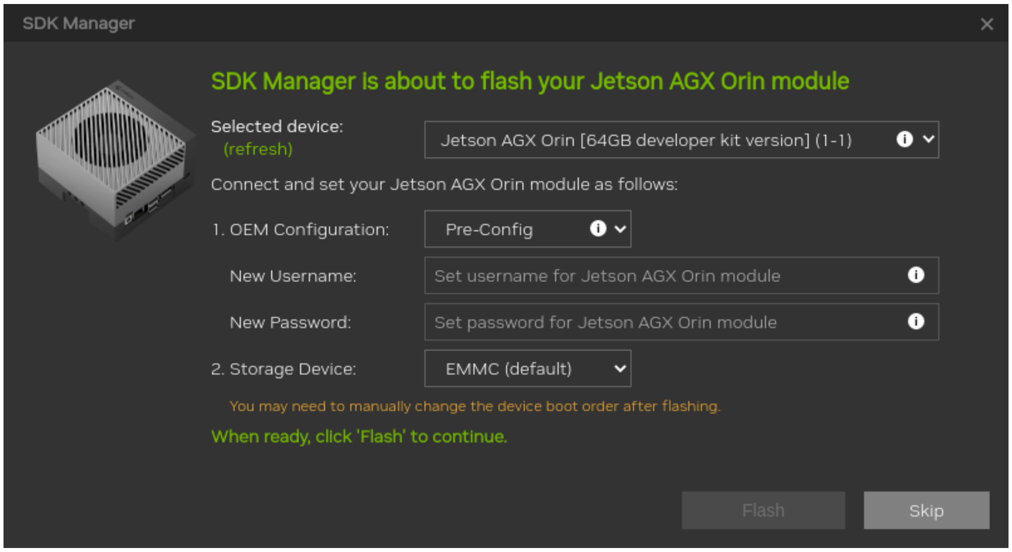
Installing: 4.17%

Download folder: /home/jason/Downloads/nvidia/sdkm_downloads

NVIDIA. Copyright © 2024, NVIDIA CORPORATION. All rights reserved. | NVIDIA Developer

Depending on your network's speed, the download can take 30 minutes or more. During the download and setup, a detailed output log can be viewed under the TERMINAL tab.

8. Once the download and setup has completed, you will be presented with a Flash dialog.



SDK Manager

SDK Manager is about to flash your Jetson AGX Orin module

Selected device: (refresh) Jetson AGX Orin [64GB developer kit version] (1-1)

Connect and set your Jetson AGX Orin module as follows:

1. OEM Configuration: Pre-Config

New Username: Set username for Jetson AGX Orin module

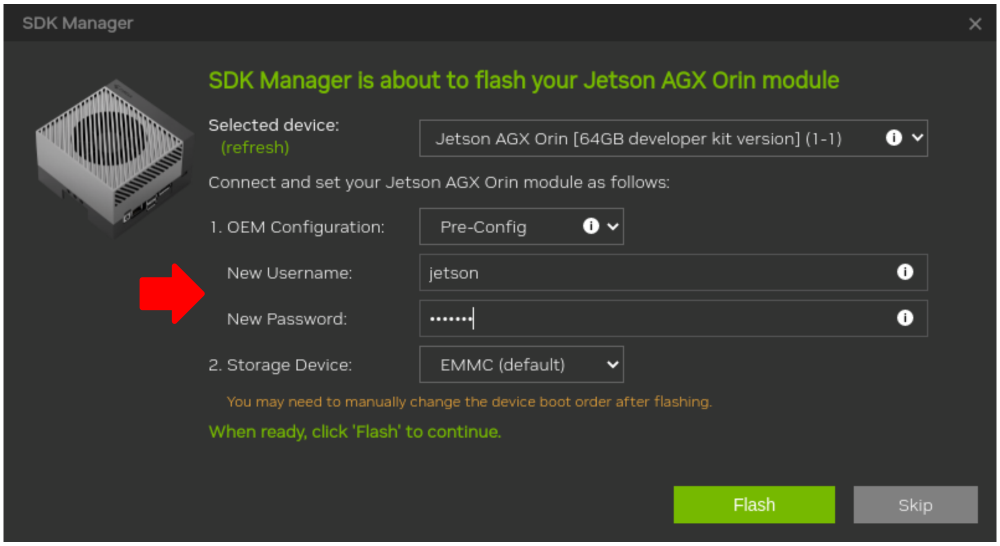
New Password: Set password for Jetson AGX Orin module

2. Storage Device: EMMC (default)

You may need to manually change the device boot order after flashing.

When ready, click 'Flash' to continue.

Fill in the New Username and New Password fields for the default user that will be created on the Orin, as shown in the following dialog:



SDK Manager

SDK Manager is about to flash your Jetson AGX Orin module

Selected device: (refresh) Jetson AGX Orin [64GB developer kit version] (1-1)

Connect and set your Jetson AGX Orin module as follows:

1. OEM Configuration: Pre-Config

New Username: jetson

New Password: •••••••

2. Storage Device: EMMC (default)

You may need to manually change the device boot order after flashing.

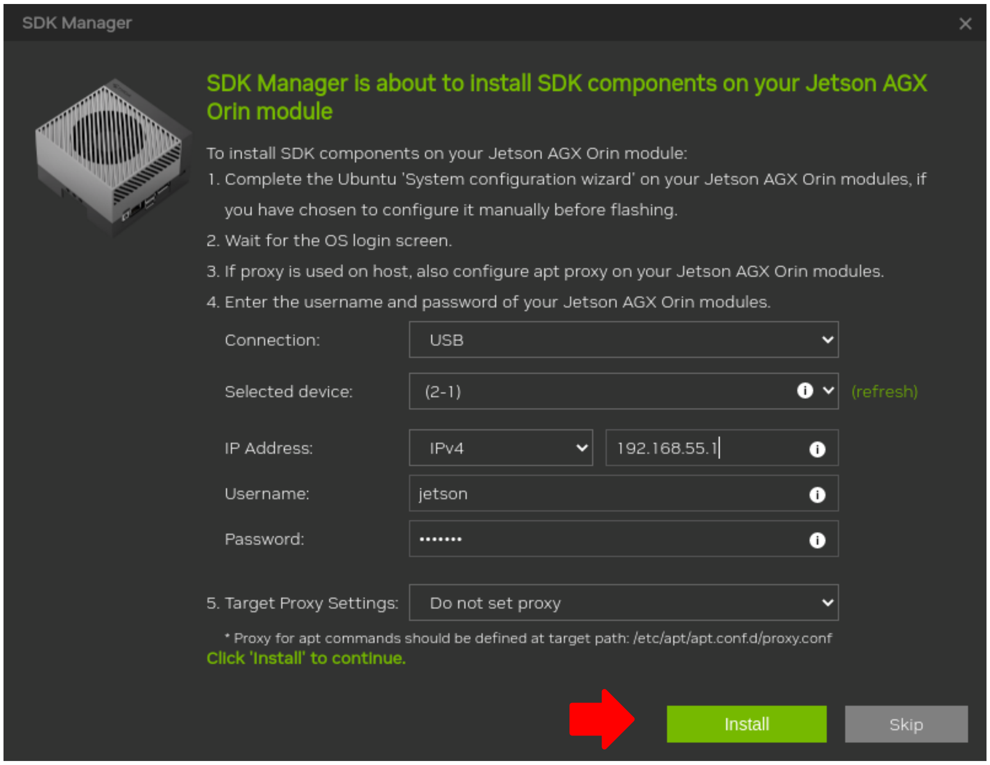
When ready, click 'Flash' to continue.

Click Flash to begin flashing the Orin.

NOTE

Fashing may occasionally fail for unknown reasons. If the flash fails, unplug the Orin and then plug it in while placing it into recovery mode. Also, disconnect the USB cables connecting the Orin to the host and then re-connect them. Subsequent flashing attempts will generally succeed.

9. During flashing the SDK Manager will again prompt you to fill in Username and Password fields as shown in the following dialog:



SDK Manager is about to install SDK components on your Jetson AGX Orin module

To install SDK components on your Jetson AGX Orin module:

1. Complete the Ubuntu 'System configuration wizard' on your Jetson AGX Orin modules, if you have chosen to configure it manually before flashing.
2. Wait for the OS login screen.
3. If proxy is used on host, also configure apt proxy on your Jetson AGX Orin modules.
4. Enter the username and password of your Jetson AGX Orin modules.

Connection: USB

Selected device: (2-1) (refresh)

IP Address: IPv4 192.168.55.1

Username: jetson

Password: •••••••

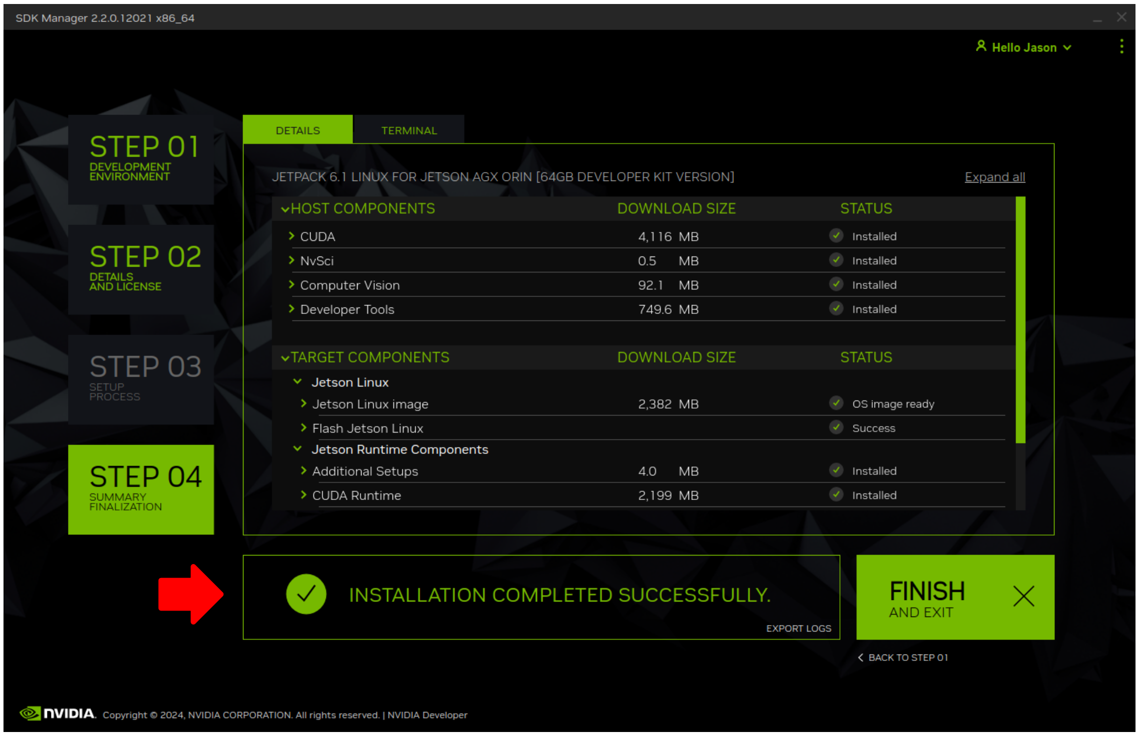
5. Target Proxy Settings: Do not set proxy

* Proxy for apt commands should be defined at target path: /etc/apt/apt.conf.d/proxy.conf

Click 'Install' to continue.

These fields default to the values specified in the previous step, so simply click Install to continue the software installation.

10. Once the flashing and software installation have completed you will automatically proceed to the STEP 04 – SUMMARY FINALIZATION screen.



JETPACK 6.1 LINUX FOR JETSON AGX ORIN [64GB DEVELOPER KIT VERSION]

HOST COMPONENTS	DOWNLOAD SIZE	STATUS
CUDA	4,116 MB	✓ Installed
NvSci	0.5 MB	✓ Installed
Computer Vision	92.1 MB	✓ Installed
Developer Tools	749.6 MB	✓ Installed

TARGET COMPONENTS	DOWNLOAD SIZE	STATUS
Jetson Linux		
Jetson Linux image	2,382 MB	✓ OS image ready
Flash Jetson Linux		✓ Success
Jetson Runtime Components		
Additional Setups	4.0 MB	✓ Installed
CUDA Runtime	2,199 MB	✓ Installed

INSTALLATION COMPLETED SUCCESSFULLY.

Copyright © 2024, NVIDIA CORPORATION. All rights reserved. | NVIDIA Developer

Click Finish and Exit to exit the SDK Manager

3.6. Set NVIDIA Power Model for optimal performance

After SDK Manager has completed the Ubuntu installation on the Orin, and while the Orin is still booted with the NVIDIA stock kernel, log into the Orin using the username and password chosen during flashing and run the following command:

```
$ sudo nvpmodel -m number
```

Where *number* is:

	3	for Orin AGX 64 GB (50 Watt)	3	for Orin AGX 32 GB (40 Watt)
	3	for Orin NX 16 GB (25 Watt)	3	for Orin NX 8 GB (20 Watt)
	0	for Orin Nano 8 GB (15 Watt)	0	for Orin Nano 4 GB (10 Watt)

Output similar to the following will then be displayed:

```
NVPM WARN: Golden image context is already created
NVPM WARN: Reboot required for changing to this power mode: 3
NVPM WARN: DO YOU WANT TO REBOOT NOW? enter YES/yes to confirm:
```

Type `yes` and press ENTER to reboot the Orin and then log in again and issue the following command:

```
$ sudo nvpmodel -q
```

Verify that the specific Orin device has been correctly set to the values indicated above.

Note that once the Orin's power mode has been set it will persist across reboots until it is explicitly changed again using the `nvpmodel` command.

3.7. Disable SOC Display Hand-Off Mode (Required)

The RedHawk kernel that will be installed in the next sections requires that SOC Display Hand-Off Mode be fully disabled (i.e., set to Never) in the Orin's firmware settings. Follow the steps in this section to disable display hand-off.

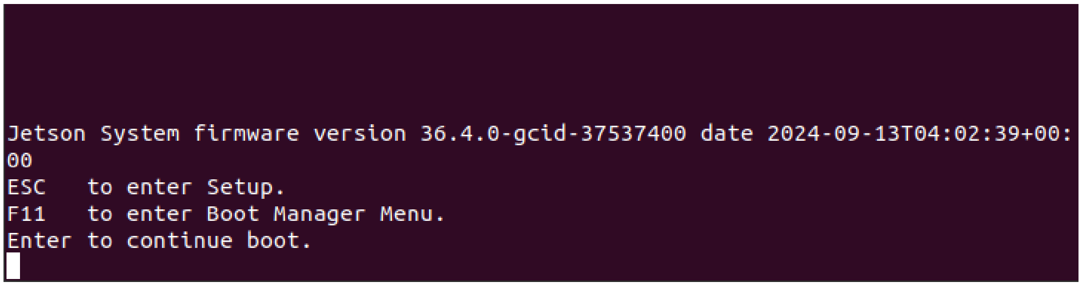
1. Connect to the Orin's serial console from the host machine. For example, if you choose to install and use `minicom` to access the serial console, invoke the following command:

```
$ sudo minicom --color=on -b 115200 -D /dev/ttyACM1
```

NOTE

You can alternatively use a monitor and keyboard connected to the Orin.

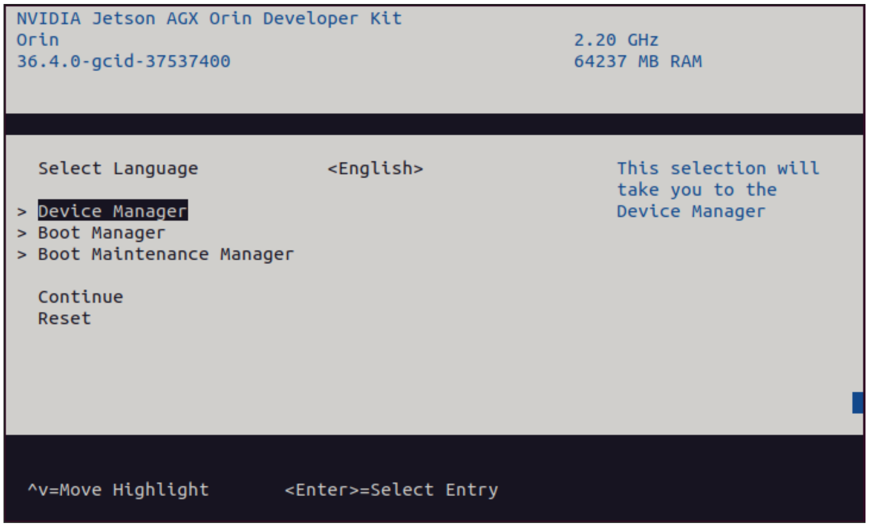
2. Reboot the Orin and wait for the following text to appear at the bottom of the serial console:



```
Jetson System firmware version 36.4.0-gcid-37537400 date 2024-09-13T04:02:39+00:
00
ESC   to enter Setup.
F11   to enter Boot Manager Menu.
Enter to continue boot.
```

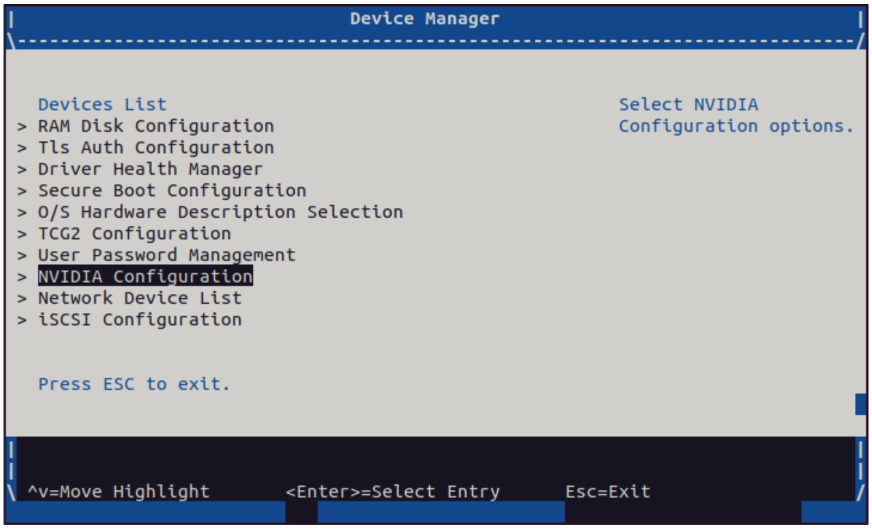
When this appears, press the ESC key to enter the firmware setup menu.

3. In the firmware setup menu, arrow down to select Device Manager (as shown below) and then press Enter:



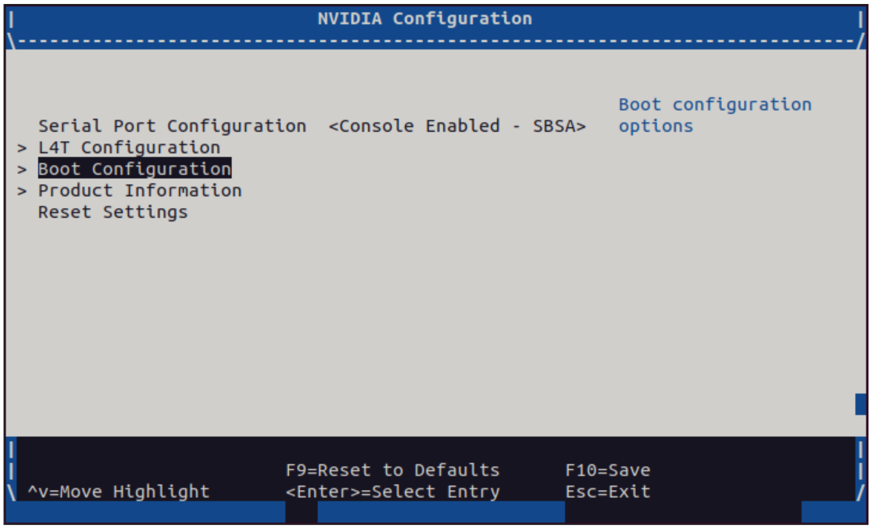
```
NVIDIA Jetson AGX Orin Developer Kit
Orin                                                2.20 GHz
36.4.0-gcid-37537400                                64237 MB RAM


  Select Language            <English>           This selection will
                                                 take you to the
> Device Manager                                 Device Manager
> Boot Manager
> Boot Maintenance Manager

  Continue
  Reset


 ^v=Move Highlight      <Enter>=Select Entry
```

4. In the Device Manager menu, arrow down to select NVIDIA Configuration (as shown below) and then press Enter.



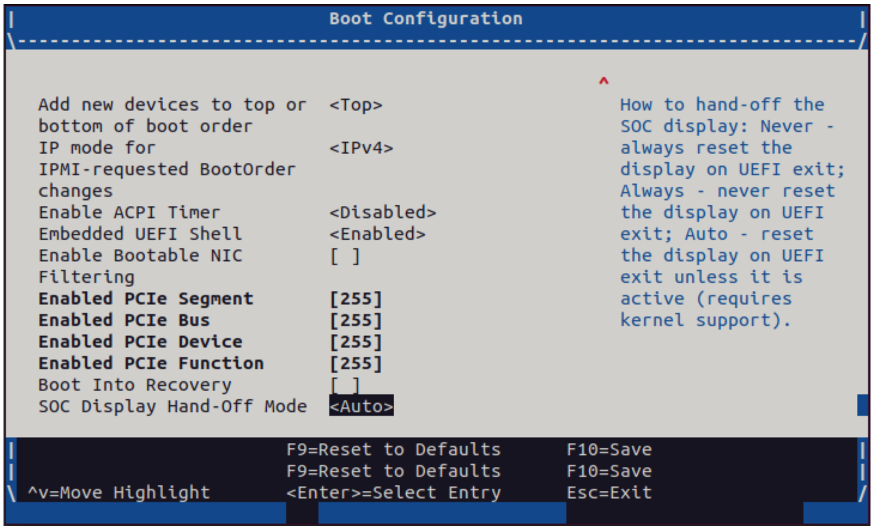
```
|                             Device Manager                             |
\------------------------------------------------------------------------/


  Devices List                                    Select NVIDIA
> RAM Disk Configuration                          Configuration options.
> Tls Auth Configuration
> Driver Health Manager
> Secure Boot Configuration
> O/S Hardware Description Selection
> TCG2 Configuration
> User Password Management
> NVIDIA Configuration
> Network Device List
> iSCSI Configuration


  Press ESC to exit.

|
|
\ ^v=Move Highlight       <Enter>=Select Entry      Esc=Exit             /
```

5. In the NVIDIA Configuration menu, arrow down to select Boot Configuration (as shown below) and then press Enter:



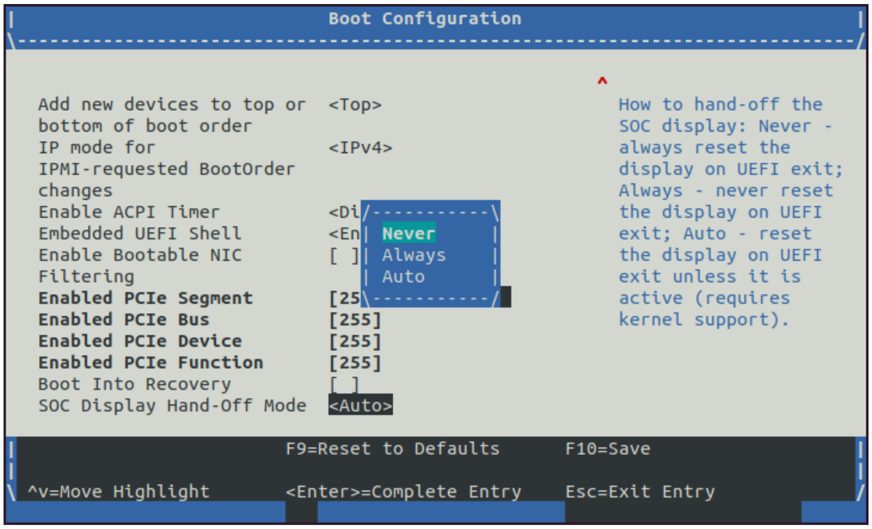
```
|                          NVIDIA Configuration                          |
\------------------------------------------------------------------------/


                                                  Boot configuration
  Serial Port Configuration  <Console Enabled - SBSA>  options
> L4T Configuration
> Boot Configuration
> Product Information
  Reset Settings

|
|                         F9=Reset to Defaults      F10=Save
\ ^v=Move Highlight       <Enter>=Select Entry      Esc=Exit             /
```

6. In the Boot Configuration menu, arrow down to select SOC Display Hand-Off Mode (as shown below) and then press Enter:



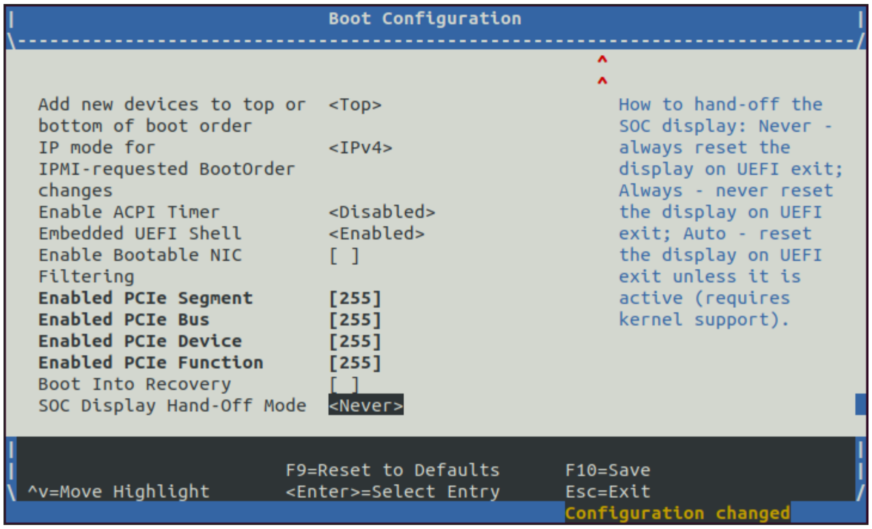
```
|                          Boot Configuration                              |
\--------------------------------------------------------------------------/

                                                       ^
  Add new devices to top or  <Top>                       How to hand-off the
  bottom of boot order                                   SOC display: Never -
  IP mode for                <IPv4>                      always reset the
  IPMI-requested BootOrder                               display on UEFI exit;
  changes                                                Always - never reset
  Enable ACPI Timer          <Disabled>                  the display on UEFI
  Embedded UEFI Shell        <Enabled>                   exit; Auto - reset
  Enable Bootable NIC        [ ]                         the display on UEFI
  Filtering                                              exit unless it is
  Enabled PCIe Segment       [255]                       active (requires
  Enabled PCIe Bus           [255]                       kernel support).
  Enabled PCIe Device        [255]
  Enabled PCIe Function      [255]
  Boot Into Recovery         [ ]
  SOC Display Hand-Off Mode  <Auto>

|                       F9=Reset to Defaults      F10=Save                 |
|                       F9=Reset to Defaults      F10=Save                 |
\ ^v=Move Highlight     <Enter>=Select Entry      Esc=Exit                 /
```

7. In the SOC Display Hand-Off Mode menu, arrow up to select Never (as shown below) and then press Enter:



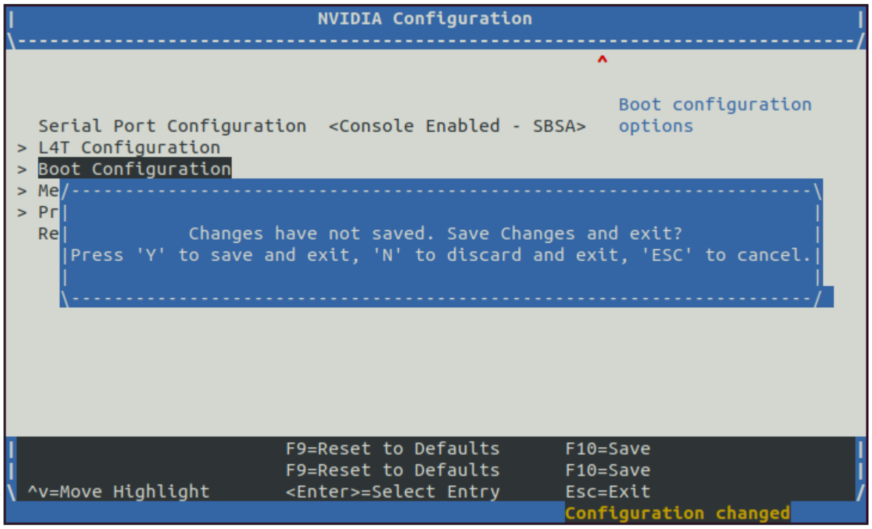
```
|                          Boot Configuration                              |
\--------------------------------------------------------------------------/

                                                       ^
  Add new devices to top or  <Top>                       How to hand-off the
  bottom of boot order                                   SOC display: Never -
  IP mode for                <IPv4>                      always reset the
  IPMI-requested BootOrder                               display on UEFI exit;
  changes                                                Always - never reset
  Enable ACPI Timer          <Di/-----------\            the display on UEFI
  Embedded UEFI Shell        <En| Never      |           exit; Auto - reset
  Enable Bootable NIC        [ ]| Always     |           the display on UEFI
  Filtering                     | Auto       |           exit unless it is
  Enabled PCIe Segment       [25\-----------/            active (requires
  Enabled PCIe Bus           [255]                       kernel support).
  Enabled PCIe Device        [255]
  Enabled PCIe Function      [255]
  Boot Into Recovery         [ ]
  SOC Display Hand-Off Mode  <Auto>

|                       F9=Reset to Defaults      F10=Save                 |
|                                                                          |
\ ^v=Move Highlight     <Enter>=Complete Entry    Esc=Exit Entry           /
```

8. Confirm that Never has been selected (as shown below) and press ESC to exit the menu:



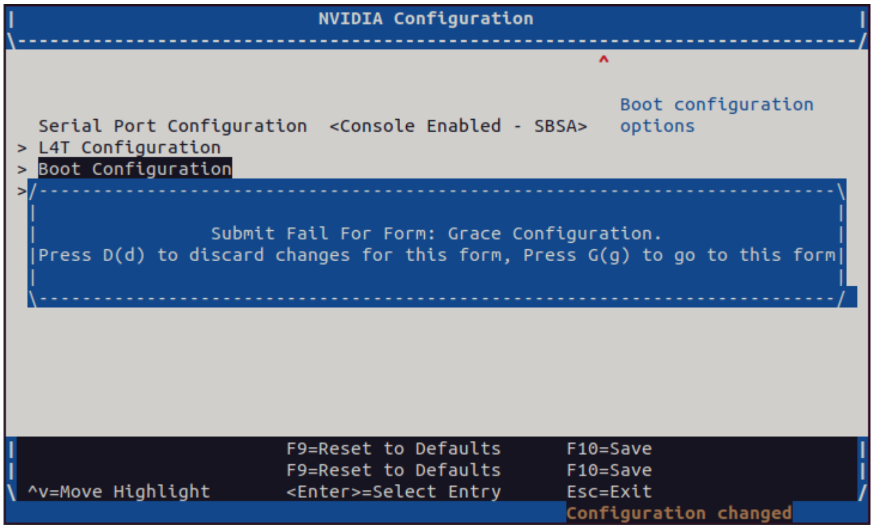
```
|                          Boot Configuration                              |
\--------------------------------------------------------------------------/
                                                       ^
                                                       ^
  Add new devices to top or  <Top>                       How to hand-off the
  bottom of boot order                                   SOC display: Never -
  IP mode for                <IPv4>                      always reset the
  IPMI-requested BootOrder                               display on UEFI exit;
  changes                                                Always - never reset
  Enable ACPI Timer          <Disabled>                  the display on UEFI
  Embedded UEFI Shell        <Enabled>                   exit; Auto - reset
  Enable Bootable NIC        [ ]                         the display on UEFI
  Filtering                                              exit unless it is
  Enabled PCIe Segment       [255]                       active (requires
  Enabled PCIe Bus           [255]                       kernel support).
  Enabled PCIe Device        [255]
  Enabled PCIe Function      [255]
  Boot Into Recovery         [ ]
  SOC Display Hand-Off Mode  <Never>

|                                                                          |
|                       F9=Reset to Defaults      F10=Save                 |
\ ^v=Move Highlight     <Enter>=Select Entry      Esc=Exit                 /
                                                  Configuration changed
```

9. Press ESC twice and you will then be prompted to either save or discard changes (as shown below). Press Y to save the changes.



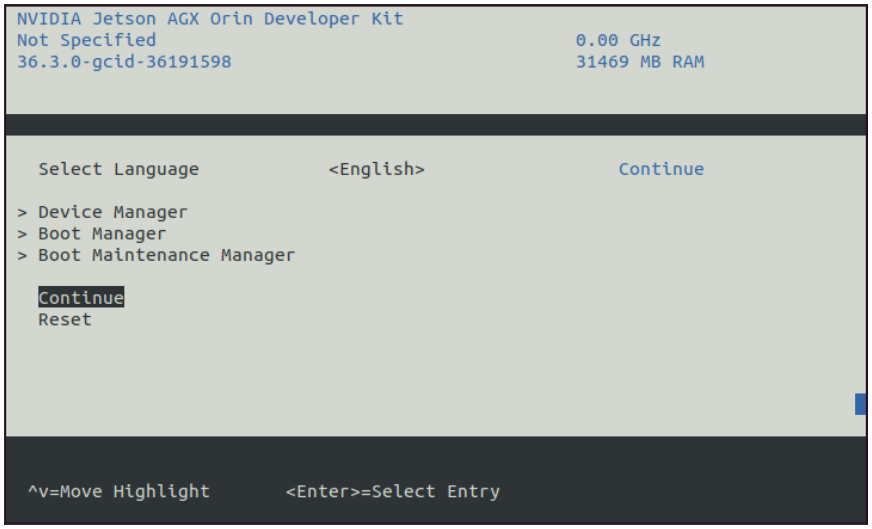
```
|                       NVIDIA Configuration                           |
\----------------------------------------------------------------------/
                                                  ^
                                                      Boot configuration
  Serial Port Configuration  <Console Enabled - SBSA>   options
> L4T Configuration
> Boot Configuration
> Me/----------------------------------------------------------------\
> Pr|                                                                |
  Re|          Changes have not saved. Save Changes and exit?        |
    |Press 'Y' to save and exit, 'N' to discard and exit, 'ESC' to cancel.|
    |                                                                |
    \----------------------------------------------------------------/

|                       F9=Reset to Defaults      F10=Save           |
|                       F9=Reset to Defaults      F10=Save           |
\ ^v=Move Highlight     <Enter>=Select Entry      Esc=Exit           /
                                                  Configuration changed
```

10. At this point an erroneous dialog may appear indicating that changes to the Grace Configuration form failed to submit. Press D to discard and exit the menu.



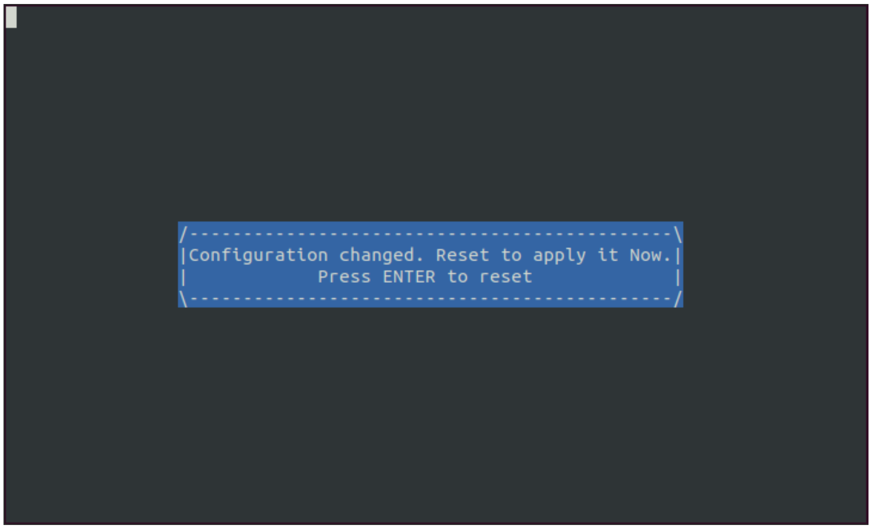
```
|                       NVIDIA Configuration                           |
\----------------------------------------------------------------------/
                                                  ^
                                                      Boot configuration
  Serial Port Configuration  <Console Enabled - SBSA>   options
> L4T Configuration
> Boot Configuration
>/-------------------------------------------------------------------\
 |                                                                   |
 |             Submit Fail For Form: Grace Configuration.            |
 |Press D(d) to discard changes for this form, Press G(g) to go to this form|
 |                                                                   |
 \-------------------------------------------------------------------/

|                       F9=Reset to Defaults      F10=Save           |
|                       F9=Reset to Defaults      F10=Save           |
\ ^v=Move Highlight     <Enter>=Select Entry      Esc=Exit           /
                                                  Configuration changed
```

11. Press ESC again to return to the top-level firmware settings menu, then arrow down to select Continue (as shown below) and then press Enter:



```
NVIDIA Jetson AGX Orin Developer Kit
Not Specified                                       0.00 GHz
36.3.0-gcid-36191598                                31469 MB RAM


  Select Language            <English>                 Continue

> Device Manager
> Boot Manager
> Boot Maintenance Manager

  Continue
  Reset



 ^v=Move Highlight       <Enter>=Select Entry
```

12. Finally, press Enter when prompted to reset and reboot the Orin, as shown below:



```
/---------------------------------------------\
|Configuration changed. Reset to apply it Now.|
|            Press ENTER to reset             |
\---------------------------------------------/
```

3.8. Copy RedHawk packages to the Orin

NOTE

This section assumes you are using the RedHawk 9.4 Gold for Jetson AGX Orin, Orin NX, and Orin Nano optical media disc. Alternatively, if you wish to install RedHawk over the network, refer to Appendix A, "Install RedHawk from Concurrent Real-Time Repositories" at the end of this document. Once installed, continue with section 3.9 below to complete installation.

After SDK Manager has completed the Ubuntu installation on the Orin, insert the *RedHawk 9.4 Gold for Jetson AGX Orin, Orin NX, and Orin Nano* optical disc into the Ubuntu host's optical drive and it should automatically mount under the `/media/$USER` directory (e.g., `/media/jane/RedHawk-9.4-aarch64`).

Once mounted, issue the following command to copy the full contents of the disc into the Orin's `/tmp` directory:

```
$ scp -r /media/jane/RedHawk-9.4-aarch64 user@Orin-IP-Address:/tmp
```

NOTE

Use the username and password chosen during flashing.

3.9. Install RedHawk packages on the Orin

To install the RedHawk packages, log into the Orin (either via `ssh` or via the graphical console) and issue the following commands:

```
$ cd /tmp/RedHawk-9.4-aarch64
$ sudo ./install-redhawk
```

Enter the password for the current user if the `sudo` command prompts for it.

NOTE

Installation will take approximately 10 minutes to complete, as packages are installed and NVIDIA kernel modules are compiled.

NOTE

Some errors and warnings will be displayed during package installation, including:

- *debconf: delaying package configuration, since apt-utils is not installed*
- *dpkg: linux-libc-dev:arm64: dependency problems, but removing anyway*
- *Warning: couldn’t identify type of root file system for fsck hook*
- *update-alternatives: warning: skip creation because associated file doesn't exist*
- *Error: Unable to read grub2 configuration file "/boot/grub/grub.cfg"*
- *[Errno 2] No such file or directory: '/boot/grub/grub.cfg'*

You can safely ignore these messages as they are not relevant to RedHawk on Jetson platform embedded installations.

At this point, all RedHawk software should be successfully installed on the Orin, however additional steps in the next section are required before you can boot RedHawk kernels.

3.10. Add RedHawk entries to the `extlinux.conf` file

Perform the following steps on the Orin to create boot entries for the RedHawk kernels:

1. Create a backup of the `/boot/extlinux/extlinux.conf` file.

```
$ cd /boot/extlinux
$ sudo cp extlinux.conf extlinux.bak
```

2. Open the `extlinux.conf` file in your preferred editor. For example, if using `vi`:

```
$ sudo vi extlinux.conf
```

3. Duplicate all the `primary kernel` entry text lines to create a second identical entry below the initial entry. For example:

```
TIMEOUT 30
DEFAULT primary

MENU TITLE L4T boot options

LABEL primary
    MENU LABEL primary kernel
    LINUX /boot/Image
    INITRD /boot/initrd
    APPEND ...many boot options...

LABEL primary
    MENU LABEL primary kernel
    LINUX /boot/Image
    INITRD /boot/initrd
    APPEND ...many boot options...
```

4. Customize the *first* kernel entry to point to the desired RedHawk kernel. For example, to create an entry for the RedHawk trace kernel, modify the `LABEL`, `MENU LABEL`, `LINUX`, `INITRD`, and `APPEND` lines as follows:

```
LABEL redhawk-trace-graphics
    MENU LABEL redhawk-trace kernel with graphics
    LINUX /boot/vmlinuz-6.6.49-rt41-RedHawk-9.4-trace
    INITRD /boot/initrd.img-6.6.49-rt41-RedHawk-9.4-trace
```

Important: Make sure to add the `nvidia_drm.fbdev=0` and `nvidia_drm.modeset=0` boot options to the `APPEND` line for only RedHawk kernels as follows:

```
APPEND ...many boot options... nvidia_drm.fbdev=0 nvidia_drm.modeset=0
```

NOTE

These additional boot options are required for NVIDIA graphics to work with RedHawk 9.4 kernels, but these options will hide early boot messages and also prevent successfully switching into run-level 3 (multi-user no graphics). Refer to Known Issues section 4.1 for more information and workarounds.

NOTE

Modifying the first kernel entry will allow the NVIDIA kernel to be booted as a fallback if any typos were entered during the customization of the first entry. Failure to follow this guideline can render the Orin unbootable and require re-flashing if a typo is accidentally introduced during editing.

5. Change the default kernel to the newly added RedHawk kernel by changing the `DEFAULT` entry at the top of the file:

```
TIMEOUT 30
DEFAULT redhawk-trace-graphics
```

NOTE

If the Orin has been set up with a serial console, you may choose to skip this step and instead interactively decide which kernel to boot when you are presented with the boot menu on the serial console.

3.11. Verify RedHawk kernel installation

Upon reboot, the Orin should now be running the selected RedHawk kernel. To verify this, issue the following command:

```
$ uname -r
```

You should see output displayed like the following:

```
6.6.49-rt41-RedHawk-9.4-trace
```

Please contact Concurrent Real-Time technical support if you had any problems during this installation (support@concurrent-rt.com or 1-800-245-6453).

4. Known Issues

Special consideration should be given to the following areas.

4.1. NVIDIA frame buffer incompatibilities and workarounds

JetPack 6.1 provides an older NVIDIA graphics driver that has frame buffer incompatibilities with newer Linux kernels. NVIDIA is aware of this issue and it is expected that future JetPack updates will eventually correct this, but until then RedHawk will require special workarounds described below. All of these workarounds require first disabling the SOC Display Hand-Off mode in the firmware settings, described in section 3.7 above.

4.1.1. Booting into graphical mode (run-level 5)

For NVIDIA graphics to work with RedHawk, the `extlinux.conf` entry requires additional `nvidia_drm.fbdev=0` and `nvidia_drm.modeset=0` boot options, as shown below:

```
LABEL redhawk-trace-graphics
   MENU LABEL redhawk-trace kernel with graphics
   LINUX /boot/vmlinuz-6.6.49-rt41-RedHawk-9.4-trace
   INITRD /boot/initrd.img-6.6.49-rt41-RedHawk-9.4-trace
   APPEND ...many boot options... nvidia_drm.fbdev=0 nvidia_drm.modeset=0
```

In this graphical mode, you cannot successfully switch to run-level 1 or run-level 3 because the text will be displayed on a frame buffer that is incompatible with modern Linux kernels.

4.1.2. Booting into text mode (run-level 3)

To successfully boot into run-level 3 (multi-user no graphics), the `extlinux.conf` entry requires an additional `3` boot option, as shown below:

```
LABEL redhawk-trace-text-mode
   MENU LABEL redhawk-trace kernel in text mode
   LINUX /boot/vmlinuz-6.6.49-rt41-RedHawk-9.4-trace
   INITRD /boot/initrd.img-6.6.49-rt41-RedHawk-9.4-trace
   APPEND ...many boot options... 3
```

In this text mode, you cannot successfully switch to run-level 5 to start graphics because the graphics will be displayed on a frame buffer that is incompatible with modern Linux kernels.

4.1.3. Booting into hybrid mode (all run-levels, limited CUDA)

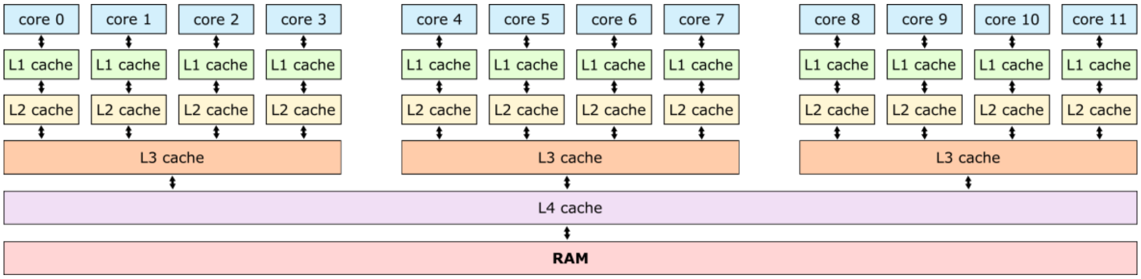
A third hybrid option is available that supports switching between all run levels, however with this option CUDA applications that utilize the frame buffer or display graphics cannot be run. However, CUDA applications that utilize CUDA for non-graphical workloads can be run successfully in this mode, so this mode may be appropriate for some applications.

To successfully boot into this hybrid mode, no additional boot options are required, simply replace the `/etc/X11/xorg.conf` file with a copy of the `/etc/X11/xorg.conf.ccur-failsafe` file. The X configuration in the failsafe file allows graphics to work with RedHawk along with run-level switching but suffers from the previously mentioned CUDA limitations.

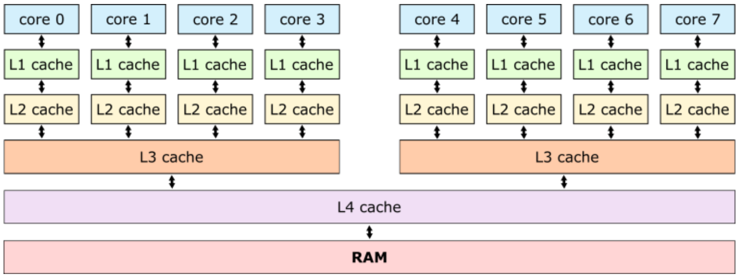
4.2. Cores in clusters can interfere with shielding

All Jetson AGX Orin series, Orin NX series, and Orin Nano series devices have cores grouped into clusters that utilize a 4-tier cache as shown in the following diagrams:

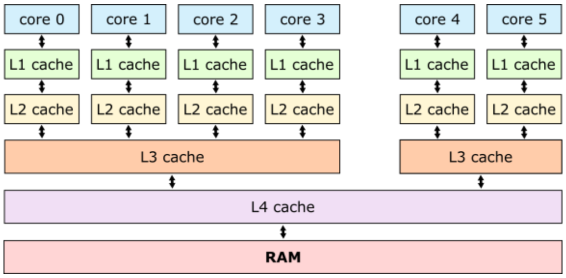
AGX Orin 64 GB / AGX Orin Industrial



AGX Orin 32 GB / Orin NX 16 GB



Orin NX 8 GB / Orin Nano 8 GB / Orin Nano 4 GB



Each core has its own L1 and L2 caches, but L3 caches are always shared between two or four other cores in the same cluster. For example, in all the above diagrams, cores 0, 1, 2, and 3 are always sharing an L3 cache which is isolated from the L3 caches of other clusters.

Note that the L4 cache in these diagrams is often called a “system cache” because the GPU also uses this cache, though for the GPU it is an L3 cache.

When shielding a core on the AGX Orin, Orin NX, or Orin Nano for a real-time application, activity on the other cores in the cluster can modify the shared L3 cache and impact the determinism and performance of the real-time application. To achieve maximum real-time performance, all other cores in the same cluster should be also shielded, with the unused cores kept idle. Alternatively, the other cores in the same cluster can be taken offline to protect real-time performance while also reducing power consumption.

4.3. Maximizing clock speeds

NVIDIA provides the `jetson_clocks` utility to control the speeds of various hardware clocks on Orin devices. This utility provides several options and must be run as the root user. Invoke the utility with the `--show` option to view all current clock speeds. For example (AGX Orin):

```
root@ubuntu:~# jetson_clocks --show 2>/dev/null
SOC family:tegra234  Machine:Jetson AGX Orin
Online CPUs: 0-11
cpu0: Online= Governor=schedutil MinFreq=729600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=1 c7=1
cpu1: Online= Governor=schedutil MinFreq=729600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=1 c7=1
cpu10: Online= Governor=schedutil MinFreq=729600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=1 c7=1
cpu11: Online= Governor=schedutil MinFreq=729600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=1 c7=1
cpu2: Online= Governor=schedutil MinFreq=729600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=1 c7=1
cpu3: Online= Governor=schedutil MinFreq=729600 MaxFreq=2201600 CurrentFreq=1420800 IdleStates: WFI=1 c7=1
cpu4: Online= Governor=schedutil MinFreq=729600 MaxFreq=2201600 CurrentFreq=1728000 IdleStates: WFI=1 c7=1
cpu5: Online= Governor=schedutil MinFreq=729600 MaxFreq=2201600 CurrentFreq=1728000 IdleStates: WFI=1 c7=1
cpu6: Online= Governor=schedutil MinFreq=729600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=1 c7=1
cpu7: Online= Governor=schedutil MinFreq=729600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=1 c7=1
cpu8: Online= Governor=schedutil MinFreq=729600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=1 c7=1
cpu9: Online= Governor=schedutil MinFreq=729600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=1 c7=1
GPU MinFreq=306000000 MaxFreq=1300500000 CurrentFreq=306000000
EMC MinFreq=204000000 MaxFreq=3199000000 CurrentFreq=2133000000 FreqOverride=0
DLA0_CORE:   Online=1 MinFreq=0 MaxFreq=1600000000 CurrentFreq=1600000000
DLA0_FALCON: Online=1 MinFreq=0 MaxFreq=844800000 CurrentFreq=844800000
DLA1_CORE:   Online=1 MinFreq=0 MaxFreq=1600000000 CurrentFreq=1600000000
DLA1_FALCON: Online=1 MinFreq=0 MaxFreq=844800000 CurrentFreq=844800000
PVA0_VPS0: Online=1 MinFreq=0 MaxFreq=1152000000 CurrentFreq=1152000000
PVA0_AXI:  Online=1 MinFreq=0 MaxFreq=832000000 CurrentFreq=832000000
FAN Dynamic Speed control=active hwmon2_pwm=56
NV Power Mode: MAXN
root@ubuntu:~#
```

Notice that, by default, some of the current frequency values above are significantly lower than the maximum frequencies supported by the hardware.

To achieve the best performance, set all clock speeds to their maximum values by invoking the `jetson_clocks` utility without any options. Then, to verify that all clock speeds have been maximized, run the `jetson_clocks` utility again with the `--show` option. For example:

```
root@ubuntu:~# jetson_clocks
root@ubuntu:~# jetson_clocks --show 2>/dev/null
SOC family:tegra234  Machine:Jetson AGX Orin
Online CPUs: 0-11
cpu0: Online= Governor=schedutil MinFreq=2201600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=0 c7=0
cpu1: Online= Governor=schedutil MinFreq=2201600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=0 c7=0
cpu10: Online= Governor=schedutil MinFreq=2201600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=0 c7=0
cpu11: Online= Governor=schedutil MinFreq=2201600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=0 c7=0
cpu2: Online= Governor=schedutil MinFreq=2201600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=0 c7=0
cpu3: Online= Governor=schedutil MinFreq=2201600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=0 c7=0
cpu4: Online= Governor=schedutil MinFreq=2201600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=0 c7=0
cpu5: Online= Governor=schedutil MinFreq=2201600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=0 c7=0
cpu6: Online= Governor=schedutil MinFreq=2201600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=0 c7=0
cpu7: Online= Governor=schedutil MinFreq=2201600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=0 c7=0
cpu8: Online= Governor=schedutil MinFreq=2201600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=0 c7=0
cpu9: Online= Governor=schedutil MinFreq=2201600 MaxFreq=2201600 CurrentFreq=2201600 IdleStates: WFI=0 c7=0
GPU MinFreq=1300500000 MaxFreq=1300500000 CurrentFreq=1300500000
EMC MinFreq=204000000 MaxFreq=3199000000 CurrentFreq=3199000000 FreqOverride=1
DLA0_CORE:   Online=1 MinFreq=0 MaxFreq=1600000000 CurrentFreq=1600000000
DLA0_FALCON: Online=1 MinFreq=0 MaxFreq=844800000 CurrentFreq=844800000
DLA1_CORE:   Online=1 MinFreq=0 MaxFreq=1600000000 CurrentFreq=1600000000
DLA1_FALCON: Online=1 MinFreq=0 MaxFreq=844800000 CurrentFreq=844800000
PVA0_VPS0: Online=1 MinFreq=0 MaxFreq=1152000000 CurrentFreq=1152000000
PVA0_AXI:  Online=1 MinFreq=0 MaxFreq=832000000 CurrentFreq=832000000
FAN Dynamic Speed control=active hwmon2_pwm=56
NV Power Mode: MAXN
root@ubuntu:~#
```

Notice that all the current frequency values above have been configured to their maximum speeds.

NOTE

Unlike `nvpmodel`, clock speed changes made by `jetson_clocks` do not persist across a reboot. Thus, you must always run `jetson_clocks` again after a reboot to ensure that all clock speeds are maximized.

4.4. Maximizing fan speed during heavy loads

By default, the speed of Orin's CPU cooling fan will be dynamically adjusted, as needed, to always ensure proper cooling. This default dynamic fan behavior should be more than adequate for almost all use cases.

However, for periods of prolonged benchmarking and/or stress testing, Concurrent recommends maximizing the CPU cooling fan speed before starting tests to prevent fast temperature rises. Temperatures that rise too quicky may result in slowing down CPU speeds, which can negatively impact benchmarking.

Invoke `jetson_clocks` with the `--fan` option to maximize the speed of the CPU cooling fan.

4.5. Entropy gathering daemon interferes with real-time

Ubuntu 22.04 enables the Hardware Volatile Entropy Gathering daemon (`haveged`) that continuously samples various hardware resources (e.g., cache, branch predictions, memory translation tables) to collect entropy and improve random number generation.

This daemon's constant hardware probing activity can negatively impact real-time performance even on shielded CPU cores, and the daemon should be stopped before starting real-time workloads by invoking the following command as the root user:

```
# systemctl stop haveged.service
```

Do not disable the entropy gathering service because various Ubuntu startup services require a minimum level of entropy to be gathered before they will start; this includes services required for the X Server to properly initialize. However, after the Orin has started with graphics displaying you can safely stop the service to improve real-time performance.

4.6. APT upgrades may change default kernel

Performing APT package upgrades may result in a non-RedHawk kernel being selected as the default kernel. Make sure to review the `/boot/extlinux/extlinux.conf` file after any APT package upgrading is performed.

4.7. Perform APT upgrades running the L4T kernel

It is recommended to reboot to the L4T kernel (non-RedHawk) before performing APT package upgrades. If APT package upgrades are attempted while a RedHawk kernel is booted, the operation may fail due to various NVIDIA-provided packages (e.g., the bootloader) executing fragile post-installation scripts that get confused by non-L4T kernels.

4.8. Infinite looping during boot with RedHawk kernels

The firmware settings change documented in section 3.7 above is absolutely required for booting RedHawk kernels, along with the additional boot options documented in section 3.10 above. Failure to make the firmware settings change will result in RedHawk kernels infinite looping during boot such that the Jetson will appear to have crashed. However, messages will be continuously displayed over the serial console similar to the following:

```
[    9.852776] CPU:0, Error: dce-fabric@0xde00000, irq=32
[    9.859844] **************************************
[    9.864784] CPU:0, Error:dce-fabric, Errmon:2
[    9.869273] CBB registers returning all 1's which is invalid
[    9.875141] CPU:0, Error: cbb-fabric@0x13a00000, irq=33
[    9.880527] **************************************
...
[    9.968482] WARNING: CPU: 0 PID: 0 at drivers/soc/tegra/cbb/tegra234-cbb.c:
[    9.968573] CPU: 0 PID: 0 Comm: swapper/0 Tainted: G  OE
[    9.968576] Hardware name: NVIDIA NVIDIA Jetson AGX Orin Developer Kit
```

If this happens, you must disconnect the power from the Jetson and then perform the required changes documented in sections 3.7 and 3.10.

4.9. NightView known issues with Orin platforms

At the time of publication, there are some known issues with NightView support for the Orin platforms that are currently being investigated, including:

- Debugging a program that uses CUDA may occasionally result in a segmentation violation (SIGSEGV) during CUDA initialization.

- Single stepping in programs that use CUDA may occasionally result in a segmentation violation (SIGSEGV).

These issues are actively being investigated and are expected to be resolved in a future release of NightView.

4.10. New procedures for building custom kernels

RedHawk 9.4 uses a new procedure for building custom kernels that is significantly different from previous RedHawk kernels. Follow the steps below to build custom kernels for Orin platforms.

NOTE

Building custom kernels requires a significant amount storage space and it is recommended to use Orins that have been installed onto larger NVME drives.

For example, to build a custom kernel without kernel tracing enabled:

1. Install additional packages that are needed for building kernels:

   ```
   $ sudo apt install libssl-dev debhelper gawk qtbase5-dev
   ```

2. Create a custom kernel config based upon the standard kernel:

   ```
   $ cd /usr/src/linux-6.6.49-RedHawk-9.4
   $ sudo cp configs/arm64/standard .config
   ```

3. Optionally, make further customizations to the kernel config:

   ```
   $ sudo make xconfig
   ```

   Or via text mode:

   ```
   $ sudo make menuconfig
   ```

4. Build the custom kernel and also package it into a Deb file:

```
$ sudo make deb-pkg
```

NOTE

Warnings may be displayed during the kernel build, including:

- *warning: building a source package without cleaning*
- *warning: native package version may not have a revision*
- *warning: missing information for output field Standards-Version*

These and other warnings are not relevant and can be safely ignored.

5. Install the newly built custom kernel packages:

```
$ sudo apt install ./debian-pkgs/*custom_9.4*.deb
```

6. Important: Build and install all of the NVIDIA out-of-tree drivers for the custom kernel:

```
$ sudo dkms install ccur-nvidia-oot/36.4.0 -k 6.6.49-rt41-RedHawk-9.4-custom
```

7. Update the custom kernel's initramfs to include the NVIDIA out-of-tree drivers:

```
$ sudo update-initramfs -u -k 6.6.49-rt41-RedHawk-9.4-custom
```

After the custom kernel is built and installed you will need to edit the `extlinux.conf` file to add an entry for the custom kernel. For example:

```
LABEL redhawk-custom
   MENU LABEL redhawk-custom
   LINUX /boot/vmlinuz-6.6.49-rt41-RedHawk-9.4-custom
   INITRD /boot/initrd.img-6.6.49-rt41-RedHawk-9.4-custom
   APPEND ...many boot options... nvidia_drm.fbdev=0 nvidia_drm.modeset=0
```

Once the entry has been added, you can either change the `DEFAULT` in `extlinux.conf` to be `redhawk-custom` or you can interactively choose the custom kernel via the early boot menu.

A. Install RedHawk from Concurrent Real-Time Repositories

RedHawk can be installed over the network utilizing the Concurrent Real-Time software repositories. This has the advantage that it will install the latest updates that have been released, however some up-front setup is required to perform the installation; follow the steps below to complete this setup.

A.1. Install Concurrent Real-Time Public Keys

Concurrent Real-Time signs its APT repositories with a GPG-generated DSA and ElGamal key pair. You need to install Concurrent Real-Time's public key so that the APT system can authorize usage of the repositories.

You can download the ccur-public-keys file and import the key file as needed into APT's key ring. You should import the key file before attempting to access software from Concurrent Real-Time's repositories.

The following commands will download the public key and install it.

```
wget -q http://redhawk.concurrent-rt.com/network/ccur-public-keys
apt-key add ccur-public-keys
```

If you wish to check the authenticity of the public keys file you downloaded from Concurrent Real-Time's web site, request the key fingerprints from Concurrent Software Support and compare them to the fingerprints of the keys you downloaded.

You can obtain the fingerprints of the keys you downloaded with the following command:

```
gpg --with-fingerprint ccur-public-keys
```

A.2. Install Repository Definition Files

Repository definition files tell the APT installation and update system how to access specific repositories. The definition files must reside in the `/etc/apt/sources.list.d/` directory and the file name must end `.list`; e.g. `/etc/apt/sources.list.d/ccur.list`.

For example, use the following format for the repository definition of the RedHawk 9.4 Gold for Jetson AGX Orin product:

```
deb [arch=arm64] http://redhawk.concurrent-rt.com/ubuntu/login/passwd/redhawk/orin 9.4.n core
```

For example, the following entry substitutes a fictional login and password:

```
deb [arch=arm64] http://redhawk.concurrent-rt.com/ubuntu/L12345/xyzzy/redhawk/orin 9.4.n core
```

Note that if the NightStar tools have been purchased for this machine, then you should also add a repository definition for the NightStar tools below the previously added line. For example:

```
deb [arch=arm64] http://redhawk.concurrent-rt.com/ubuntu/L12345/xyzzy/nightstar 5.2 rt ubu22
```

Once the repository definition file is in place the repository setup is complete. Run the following command as root to ensure that the software repositories have been added correctly:

```
sudo apt update
```

If you see errors or warnings produced that reference the newly added `ccur.list` file, then verify that all the above steps have all been completed as documented.

A.3. Install RedHawk Packages

Now that the setup is done, simply issue the following command to install all the latest RedHawk 9.4 packages (and NightStar 5.2 packages if purchased) onto the current system:

```
apt install 'ccur-*'
```

Once this command completes the RedHawk packages are installed. Continue with Section 3.8, “Add RedHawk entries to the `extlinux.conf` file," to finalize the RedHawk installation.

A.4. Support

If you need assistance, please contact the Concurrent Real-Time Software Support Center at our toll-free number 1-800-245-6453. For calls outside the continental United States, the number is 1-954-283-1822. The Software Support Center operates Monday through Friday from 8 a.m. to 5 p.m., Eastern Standard Time.

You may also submit a request for assistance at any time by using the Concurrent Real-Time website at http://concurrent-rt.com/support or by sending an email to support@concurrent-rt.com.