

Installation of RedHawk™ 7.5.4 Gold on the NVIDIA® Jetson TX2

Release Notes

April 14th, 2021

0898003-7.5.4-TX2



This page intentionally left blank

1. Introduction

This document describes the process of installing the ARM64 version of RedHawk™ 7.5.4 on the Jetson TX2. 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 TX2 with Internet access
- An x86_64 host system running Ubuntu 16.04 or 18.04 with Internet access; note that the host system should be up to date with the latest available Ubuntu package updates
- The *RedHawk 7.5.4 for the Jetson TX2* optical media disk

NOTE

Make sure that /usr/bin/python exists on the Ubuntu host system; install the python package if /usr/bin/python does not exist.

3. Installation

3.1. Download SDK Manager onto the host system

On an x86_64 host system running Ubuntu 16.04 or 18.04, download version 1.5.0-7774 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). You should also click on the Jetson TX2 Developer Kit User Guide link to download the User's Guide PDF.

Place these downloaded files into a `tx2` directory in the home directory of the current user.

3.2. Install the downloaded SDK Manager package

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

```
$ cd ~/tx2
$ sudo apt install ./sdkmanager_1.5.0-7774_amd64.deb
```

3.3. Use SDK Manager to flash the TX2

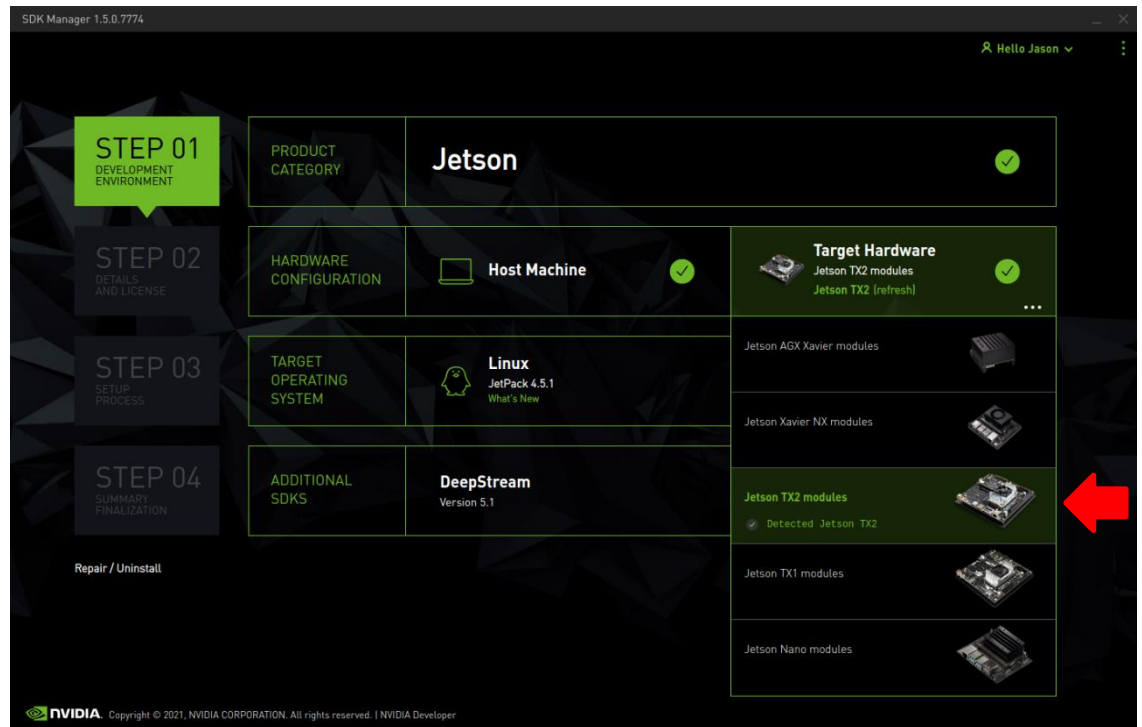
Invoke the following command as the current user to begin using the SDK Manager:

```
$ sdkmanager
```

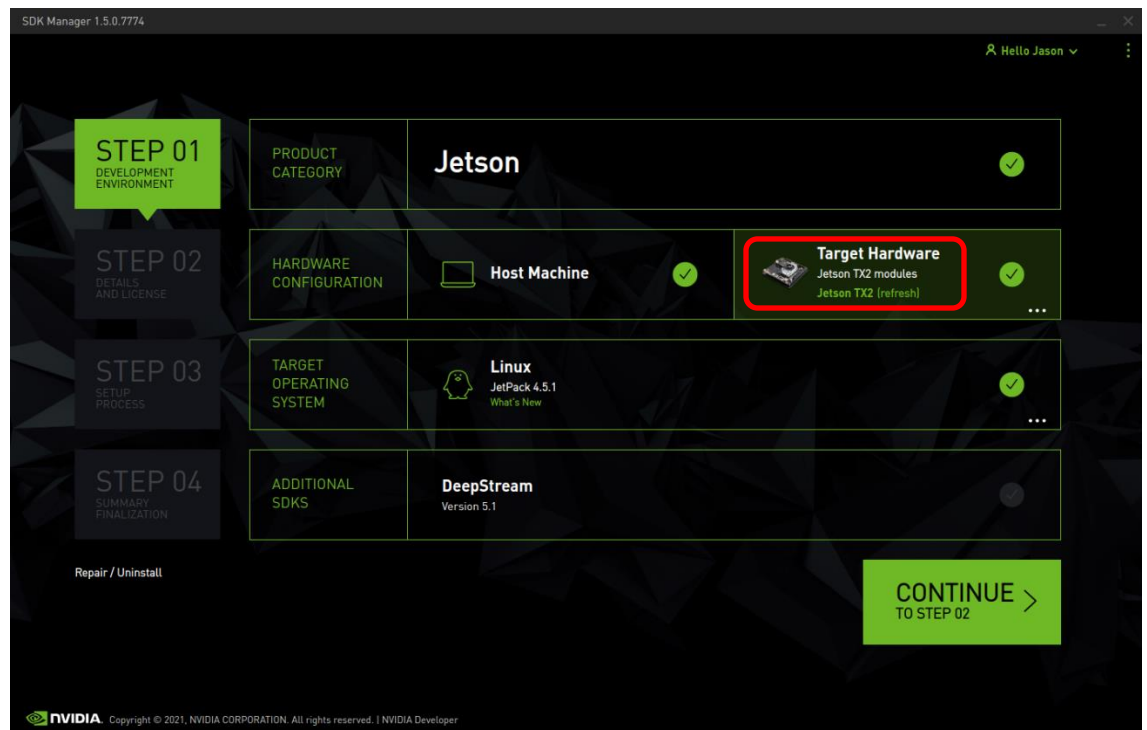
The first action required by the SDK Manager is to log into the `nvidia.com` website using your developer or partner account email address and password. Enter the appropriate information and press `LOGIN` to continue.

At this point, you can generally follow the instructions in the *Jetson TX2 Developer Kit User Guide* along with the following RedHawk-specific instructions:

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



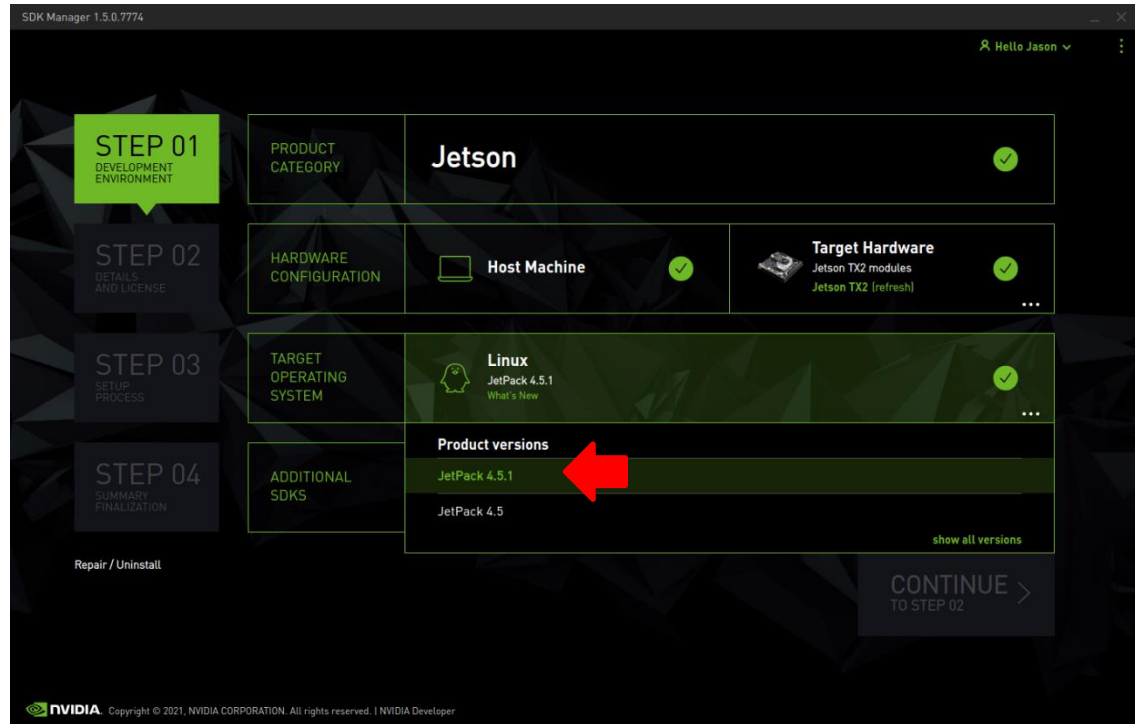
2. Click on the Jetson TX2 (3310-1000 module) target hardware option to make it the selected target, as shown in the following screenshot:



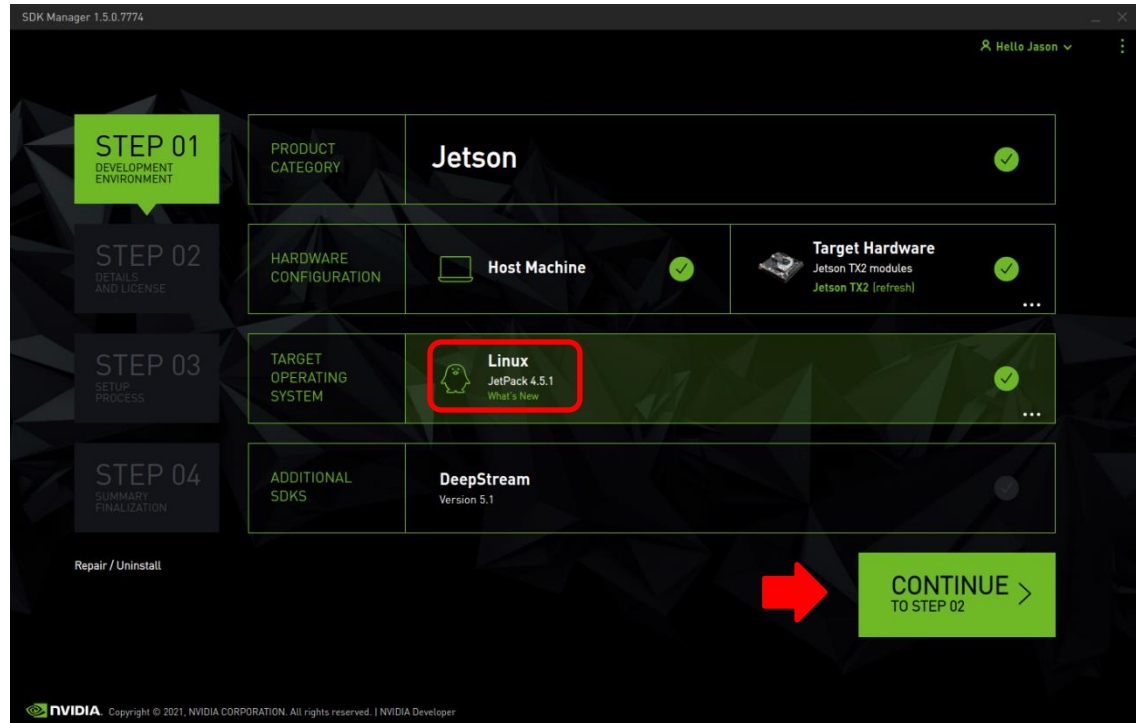
NOTE

RedHawk 7.5.4 also fully supports the Jetson TX2i (3489-0000 module); choose the Jetson TX2i module instead if you are flashing Jetson TX2i hardware.

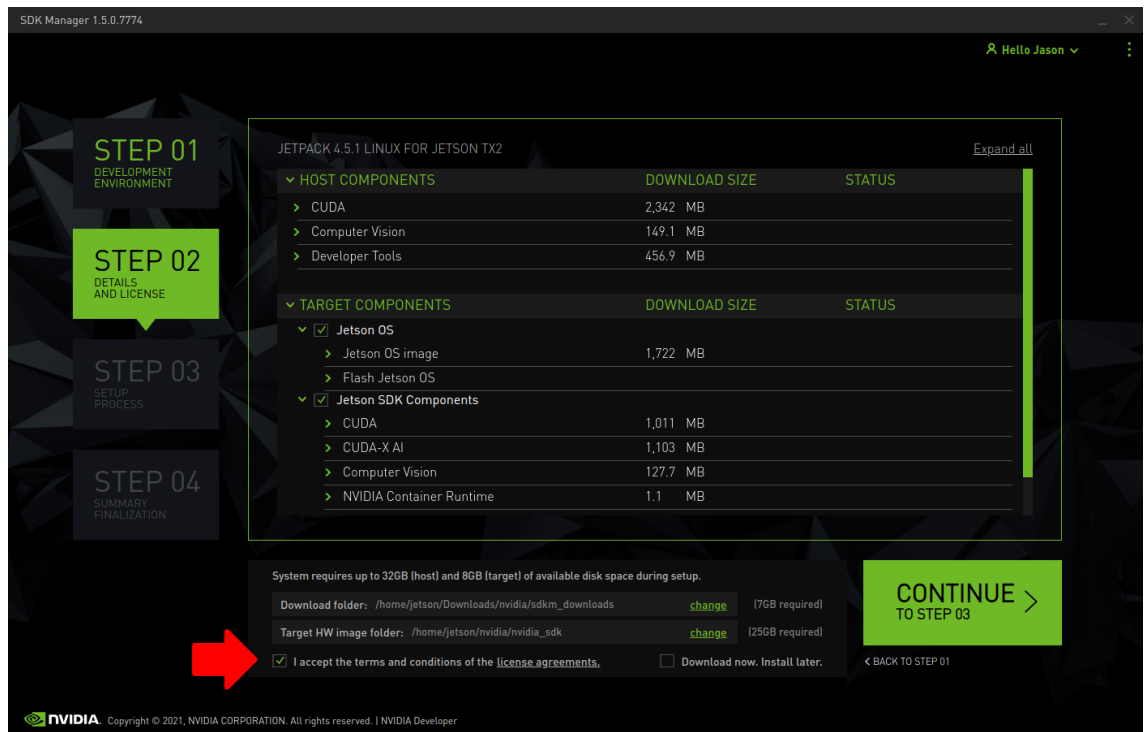
3. Click on the . . . in the Linux box to reveal the list of supported JetPack versions, as shown in the following screenshot:



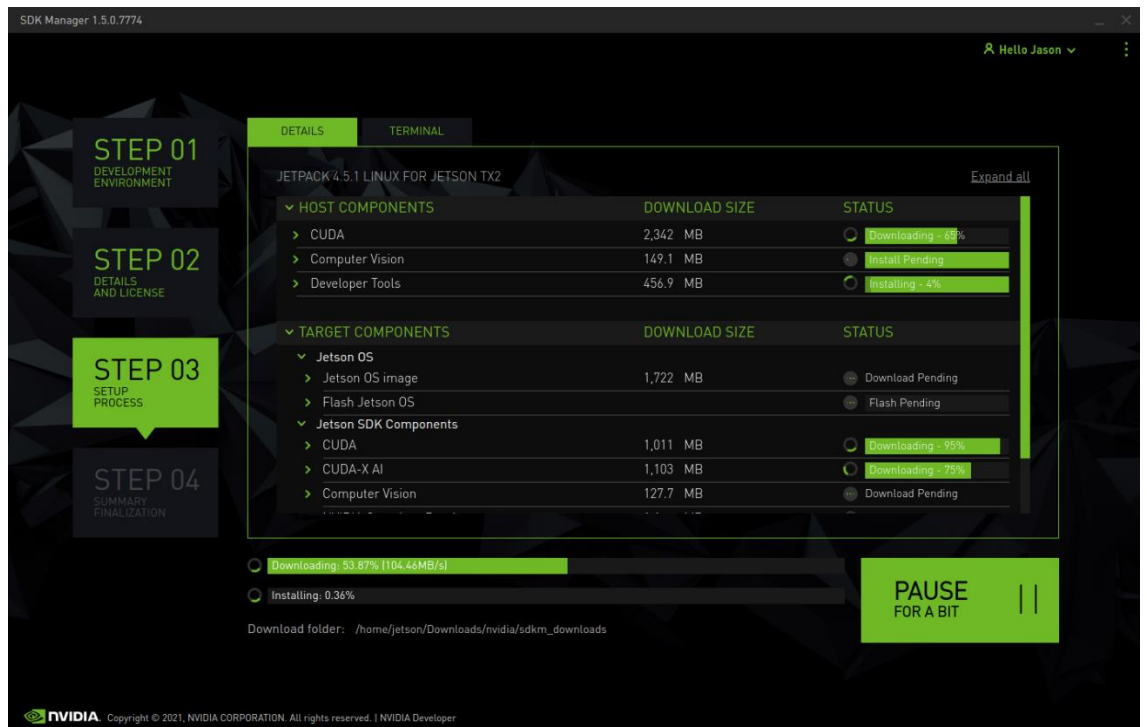
4. Click on the JetPack 4.5.1 version to make it the selected version, as shown in the following screenshot:



5. 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:



- 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:



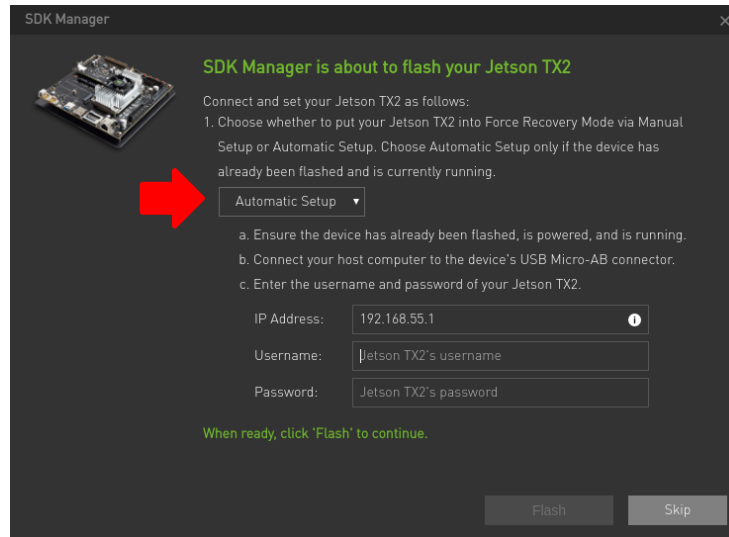
Depending on your network's speed, the download can take 30 minutes or more. During the download and setup, a detailed output log can be viewed under the TERMINAL tab.

- Once the download and setup has completed, you may be presented with the Flash dialog as shown in this section.

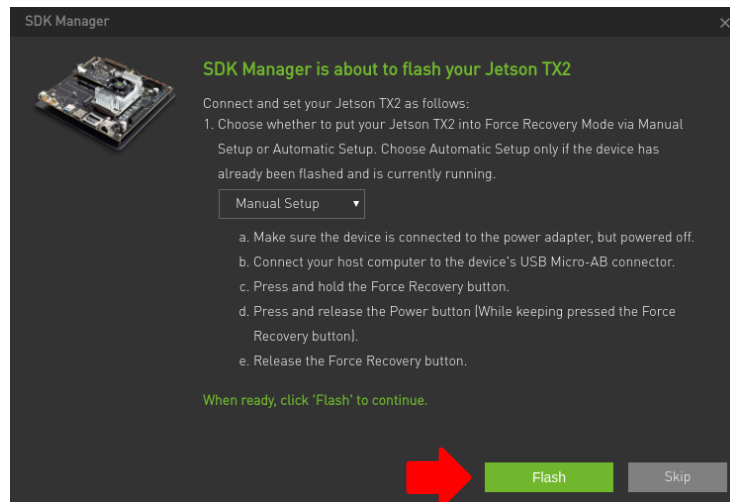
NOTE

If the Jetson TX2 is already in recovery mode and connected to the host system, then this step will be skipped, and flashing will start automatically.

Notice the Automatic setup pulldown menu as shown in the following dialog:



Click on Automatic setup and change the setting to Manual Setup as shown in the following dialog:



Follow the steps listed on the Flash dialog to place the Jetson TX2 into recovery mode and then click the Flash button to begin flashing.

NOTE

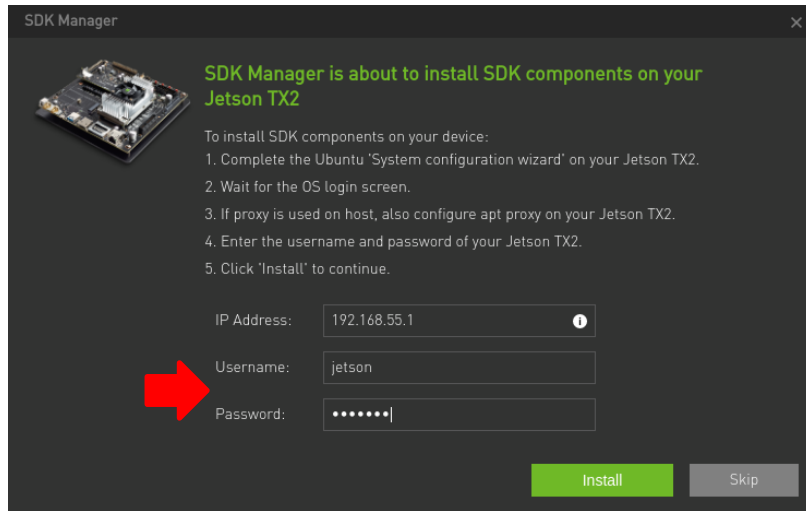
Issue the `lsusb` command on the host to verify that a single `NVIDIA Corp.` entry with device ID `0955:7c18` is displayed; this is the best indication that the TX2 has successfully entered recovery mode and is ready for flashing.

- Midway through flashing, make sure the TX2 is connected to a monitor, keyboard and mouse in order to click through the System Configuration dialogs. Accept the license and then choose values for all settings that are appropriate for your site.

During the system configuration you will be presented with the Select Nvpmode Mode dialog; accept the default because the correct mode will be set after flashing.

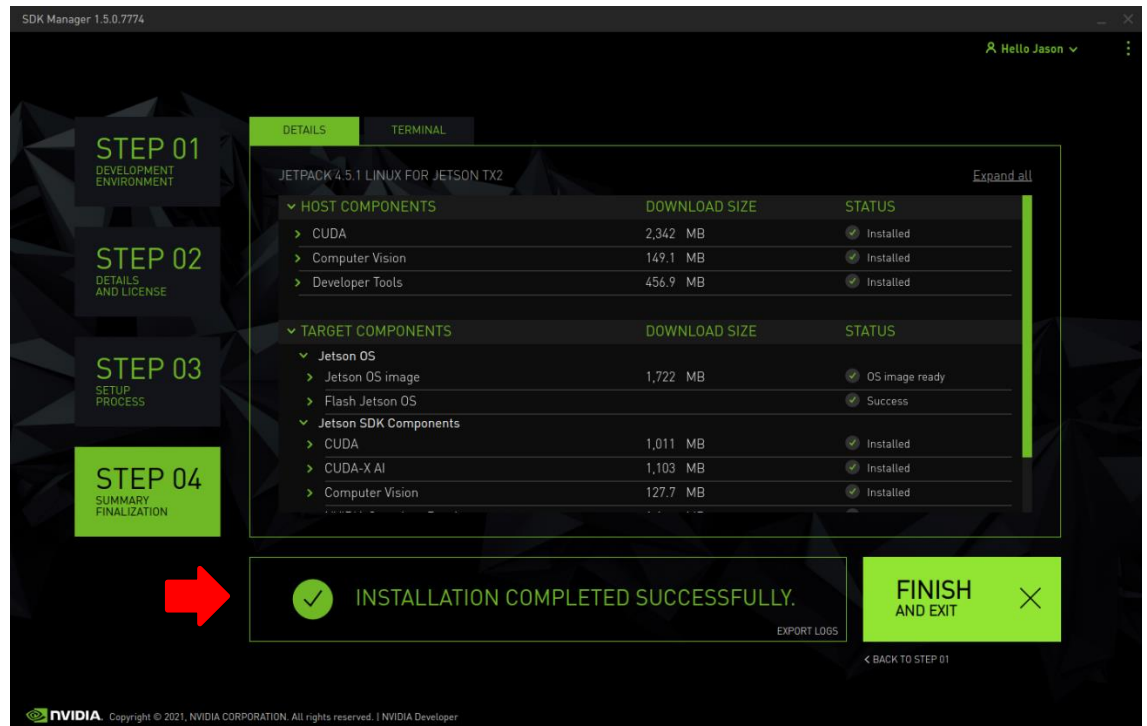
Make sure to remember the username and password chosen because this information will be needed by the SDK Manager in the following step. Press Continue to exit the System Configuration dialogs and continue with flashing.

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



Enter values for each field matching those entered into the System Configuration dialog in the previous step and then click Install to continue flashing.

10. Once the flashing has completed you will automatically proceed to the STEP 04 - SUMMARY FINALIZATION screen.



Click Finish and Exit to exit the SDK Manager

3.4. Set NVIDIA Power Model to Maximum Performance

After SDK Manager has completed the Ubuntu installation on the TX2, and while the TX2 is still booted with the NVIDIA stock kernel, log into the TX2 using the username and password chosen during flashing and run the following commands:

```
$ sudo nvpmode1 -m 0
$ sudo nvpmode1 -q
```

The second command invocation above should display the selected power mode as follows:

```
NVPM WARN: fan mode is not set!
NV Power Mode: MAXN
0
```

This confirms that the TX2 has been correctly configured for maximum performance.

NOTE

The `nvpmode1 -q` command always generates an erroneous fan warning, regardless of the current fan mode setting; this warning can be safely ignored.

3.5. Copy RedHawk packages to the TX2

NOTE

This section assumes you are using the RedHawk 7.5.4 for Jetson TX2 optical media disc. Alternatively, if you wish to install RedHawk from the network refer to Appendix A, “Install RedHawk from Concurrent Real-Time Repositories.” Once installed, continue with section 3.6 below to finalize the installation.

After SDK Manager has completed the Ubuntu installation on the TX2, insert the *RedHawk 7.5.4 for Jetson TX2* optical disc into the Ubuntu host’s optical drive and it should automatically mount under the `/media/$USER` directory (e.g., `/media/jane/RedHawk 7.5.4 aarch64`).

Change to the mounted directory on the host and use networking to copy the RedHawk packages into the newly installed TX2, as illustrated by the following commands:

```
$ cd /media/jane/RedHawk 7.5.4 aarch64/Packages
$ scp *.deb user@host:/tmp
```

NOTE

Use the username, hostname and password chosen during flashing.

3.6. Install RedHawk packages on the TX2

To install the RedHawk packages, log into the TX2 (either via `ssh` or via the graphical console) and issue the following commands:

```
$ cd /tmp
$ sudo apt install ./ccur*.deb
```

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

NOTE

Several warnings may be displayed during package installation, including:

- *Warning: couldn't identify filesystem type for fsck hook...*
- *l: The initramfs will attempt to resume from /dev/zram5*
- */sbin/ldconfig.real: Warning: ignoring configuration file...*
- */etc/ld.so.conf.d/aarch-linux-gnu_EGL.conf: No such file...*
- *N: Can't drop privileges for downloading as file...*

You can safely ignore these warnings as they are not relevant to RedHawk 7.5.4 on Jetson TX2 embedded installations.

At this point all the RedHawk packages, including the RedHawk kernels, should have installed successfully on the TX2. However, additional steps in the next section are required before you can boot into the RedHawk kernels.

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

Perform the following steps on the Jetson 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 ${cbootargs} quiet ...many other boot options... isolcpus=1-2

LABEL primary
    MENU LABEL primary kernel
    LINUX /boot/Image
    INITRD /boot/initrd
    APPEND ${cbootargs} quiet ...many other boot options... isolcpus=1-2
```

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 lines as follows:

```
LABEL redhawk-trace
    MENU LABEL redhawk-trace kernel
    LINUX /boot/Image-4.9.201-rt134-r32.5.1-tegra-RedHawk-7.5.4-trace
    INITRD /boot/initrd.img-4.9.201-rt134-r32.5.1-tegra-RedHawk-7.5.4-trace
    APPEND ${cbootargs} quiet ...many other boot options... isolcpus=1-2
```

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 TX2 unbootable and require re-flashing if a typo is accidentally introduced during editing.

5. New with JetPack 4.5.1: Modify the isolcpus boot option setting for all RedHawk kernels to change it from this:

```
isolcpus=1-2
```

to this:

```
isolcpus=
```

NOTE

Failure to perform this step can negatively impact the real-time performance achievable with RedHawk kernels.

The final boot entry for a RedHawk kernel should now look as follows:

```
LABEL redhawk-trace
MENU LABEL redhawk-trace kernel
LINUX /boot/Image-4.9.201-rt134-r32.5.1-tegra-RedHawk-7.5.4-trace
INITRD /boot/initrd.img-4.9.201-rt134-r32.5.1-tegra-RedHawk-7.5.4-trace
APPEND ${cbootargs} quiet ...many other boot options... isolcpus=
```

6. 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
```

Note that if your TX2 has been set up with a serial console, you may choose to skip step 5 and instead interactively decide which kernel you wish to boot each time the system restarts.

3.8. Verify RedHawk kernel installation

The TX2 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:

```
4.9.201-rt134-r32.5.1-tegra-RedHawk-7.5.4-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. Denver cores impact system performance

The NVIDIA-designed Denver cores (core 1 and 2) can negatively impact the real-time performance of the Jetson TX2 and TX2i. When enabled, these cores can occasionally add hundreds of microseconds of latency to real-time operations. Their negative impact can be eliminated by disabling the Denver cores with the following `cpu` command:

```
# cpu -d 1,2
```

Because of this real-time impact, the Denver cores are brought down by RedHawk at system initialization time. If desired, the cores can be enabled again for non-real-time workloads by issuing the following `cpu` command:

```
# cpu -u 1,2
```

See the `cpu (1)` man page for additional information.

4.2. Entropy gathering daemon interferes with real-time

Ubuntu 18.04 enables a new 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
```

Note that disabling the entropy gathering service is not recommended because some 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 TX2 has started with graphics displaying you can safely stop the service to improve real-time performance.

4.3. Lockdep is turned off shortly after boot

The kernel code provided by NVIDIA has not been optimized for use with debug kernels, and this is most evident when the kernel's `CONFIG_LOCKDEP_SUPPORT` feature is enabled. Because RedHawk debug kernels (`debug` and `prt-debug`) are built with this feature, lockdep informational messages will occasionally appear in the system's message journal. For example:

```
INFO: possible recursive locking detected
...
INFO: lockdep is turned off
```

These messages are solely informational, and they do not appear to cause any real-time or behavioral problems with the system. Concurrent is working with NVIDIA to address these warnings and hopefully they will be eliminated in future NVIDIA kernels.

4.4. Warnings accessing /dev/mem

Attempting to directly access /dev/mem on the TX2 is not permitted and will result in warnings output to the system console. For example, invoking the `dmidecode` command while a RedHawk kernel is running will output an error like the following:

```
Program dmidecode tried to access /dev/mem
```

These messages are informational and can be safely ignored.

Note that if the stock NVIDIA kernel is running, issuing `dmidecode` will crash the system and cause it to automatically reboot.

4.5. Error reserving Flattened Device Tree memory region

During early boot of both the NVIDIA stock kernel and the RedHawk kernels, the following messages may appear multiple times on the serial console:

```
ERROR: reserving fdt memory region failed (addr=0 size=0)
```

NVIDIA is aware of these messages, but the messages are common to all TX2 hardware and do not impact the normal operation of the system; these messages can be safely ignored.

A. Serial Console Setup

Concurrent recommends setting up a serial console when using RedHawk on the Jetson TX1 and TX2 development boards. This appendix provides details and tips for connecting a serial console.

A.1. Purchase a Raspberry Pi serial cable

First, you will need to purchase a special cable that attaches to specific pins on the Jetson motherboard to provide a USB serial connection suitable for a desktop PC. Concurrent recommends using the following Raspberry Pi cable, but other similar products may also work:

- Adafruit Industries 954 USB-to-TTL Serial Cable, Raspberry Pi
<http://www.amazon.com/ADAFRUIT-INDUSTRIES-954-SERIAL-RASPBERRY/dp/B00DJUHGH1>



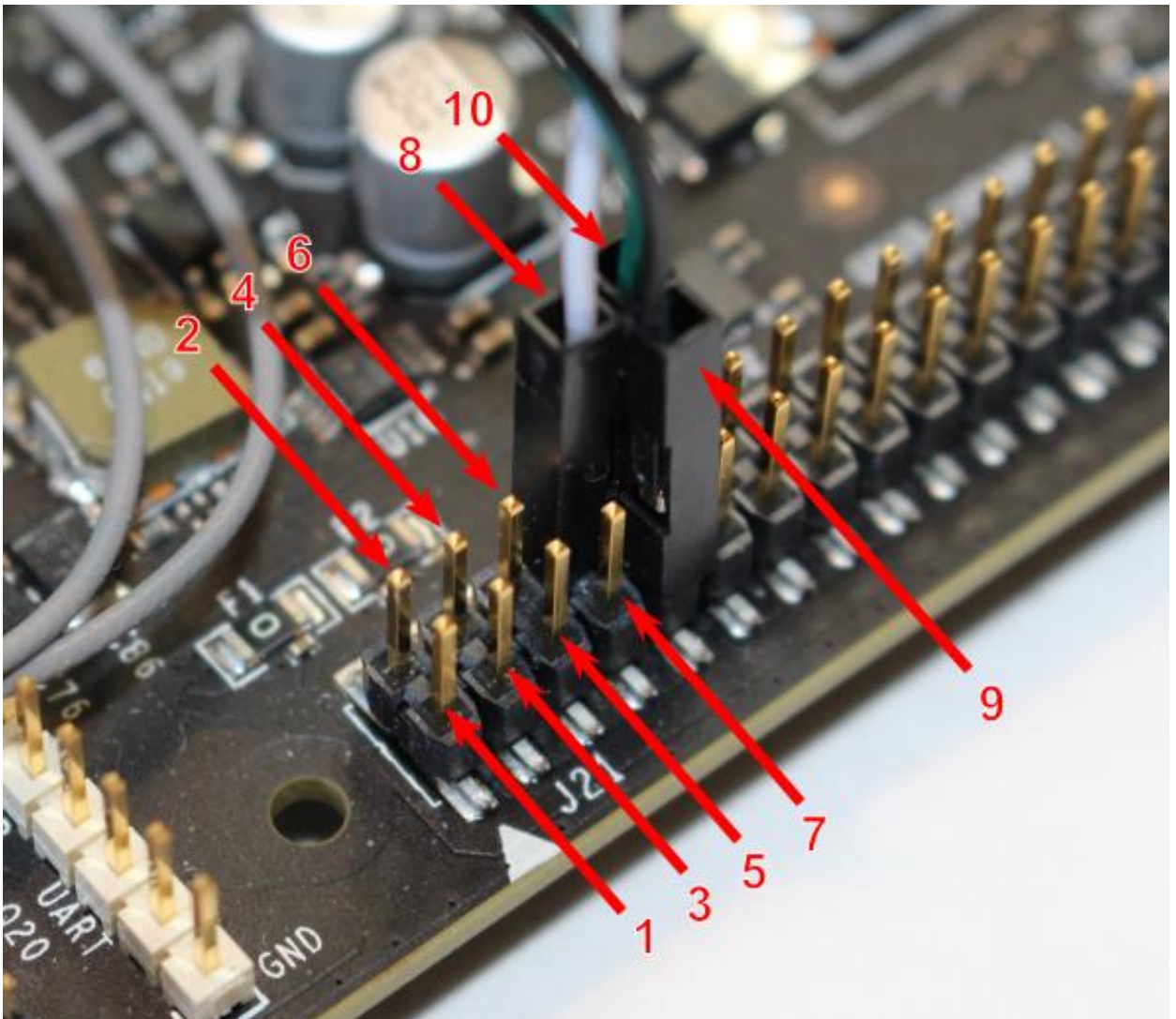
A.2. Connect the serial cable to Jetson and PC

Once you have the correct serial cable, follow the instructions in the video on this page to attach the cable to either the Jetson TX1, TX2 or TX2i and then to a desktop PC:

- Serial Console – NVIDIA Jetson TX1
<http://jetsonhacks.com/2015/12/01/serial-console-nvidia-jetson-tx1/>

Here is a brief summary of the important information of this video:

- The header pins are marked J21 with an arrow pointing to pin 1. Each pin is numbered as shown in the following image:



The serial cable's white wired connector goes to pin 8, green wired connector goes to pin 10 and black wired connector goes to pin 9. Note that the red wired connector is not used.

- The minicom settings should be configured as shown below:

```
+-----+
| A -   Serial Device       : /dev/ttyUSB0
| C -   Callin Program      :
| D -   Callout Program     :
| E -   Bps/Par/Bits        : 115200 8N1
| F -   Hardware Flow Control : No
| G -   Software Flow Control : Yes
|
|   Change which setting? █
+-----+
```

B. 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.

B.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
```

B.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`.

Use the following format for the repository definition of the RedHawk 7.5.4 for Jetson TX2 product:

```
deb http://redhawk.concurrent-rt.com/ubuntu/login/passwd/redhawk/tx2 7.5.n core
```

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

```
deb http://redhawk.concurrent-rt.com/ubuntu/LR12345/t0ps3cr3t/redhawk/tx2 7.5.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 http://redhawk.concurrent-rt.com/ubuntu/LR12345/t0ps3cr3t/nightstar 4.8 rt
```

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.

B.3. Install RedHawk Packages

Now that the setup is done, simply issue the following command to install all the latest RedHawk 7.5.4 packages (and NightStar 5.0 packages if purchased) onto the current system:

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

Once this command completes the RedHawk packages are installed. Continue with Section 3.5, "Flash desired RedHawk kernel using bootburn.sh," in order to finalize the RedHawk installation.

B.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.