# Installation of RedHawk™ 7.5.2 Gold on the NVIDIA® Jetson AGX Xavier

## Release Notes

April 28th, 2020

0898003-7.5.2-Xavier

**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™ 7.5.2 on the Jetson AGX Xavier. 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 Xavier 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 7.5.2 for the Jetson AGX Xavier* 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 1.1.0-6343 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 AGX Xavier Developer Kit User Guide link to download the User's Guide PDF.

Place these downloaded files into a `xavier` directory in the home directory of the caurrent user.

## 3.2.    Install the downloaded SDK Manager package

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

```
$ cd ~/xavier
$ sudo apt install ./sdkmanager_1.1.0-6343_amd64.deb
```

## 3.3.    Use SDK Manager to flash the Xavier

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 AGX Xavier 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:
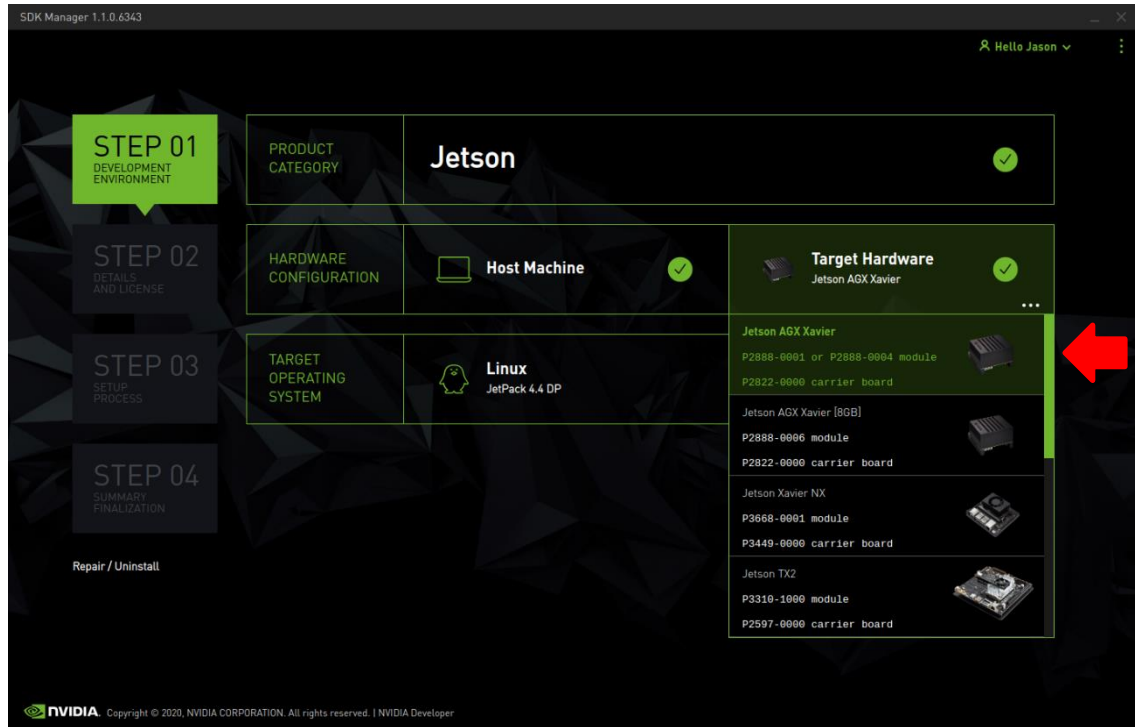


2. Click on the Jetson AGX Xavier (P2888-0001 or P2888-0004 module) target hardware option to make it the selected target, as shown in the following screenshot:
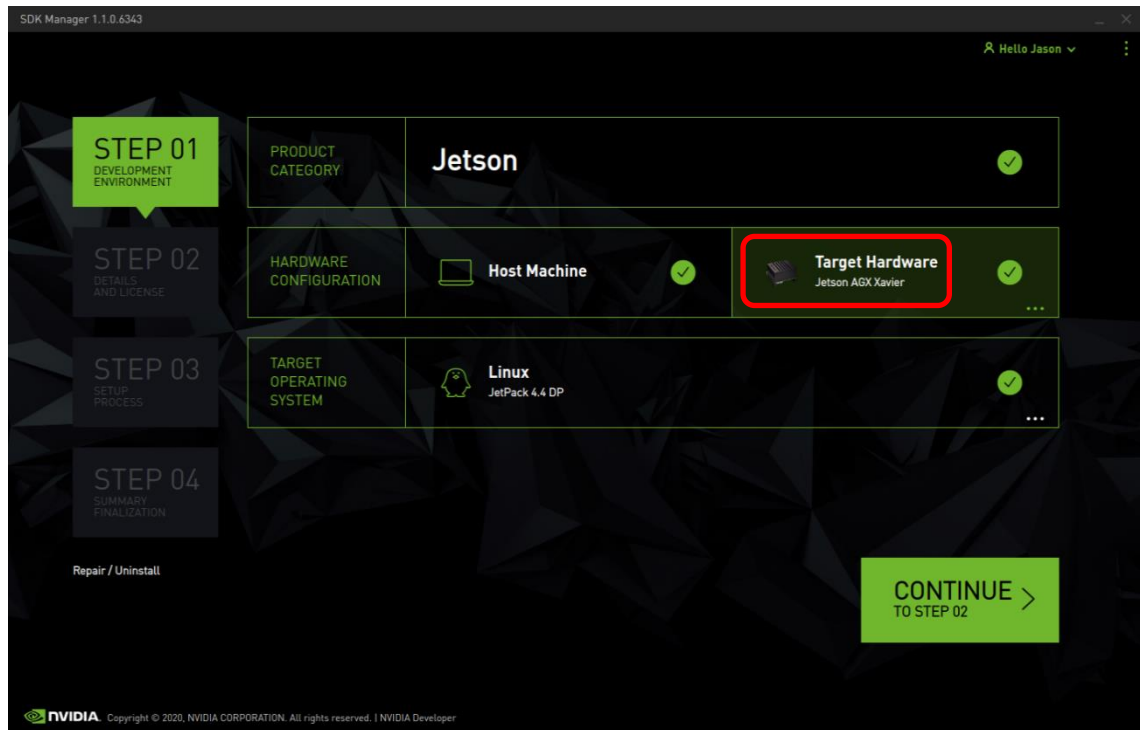


3. Click on the . . . in the Linux box to reveal the list of supported JetPack versions, as shown in the following screenshot:
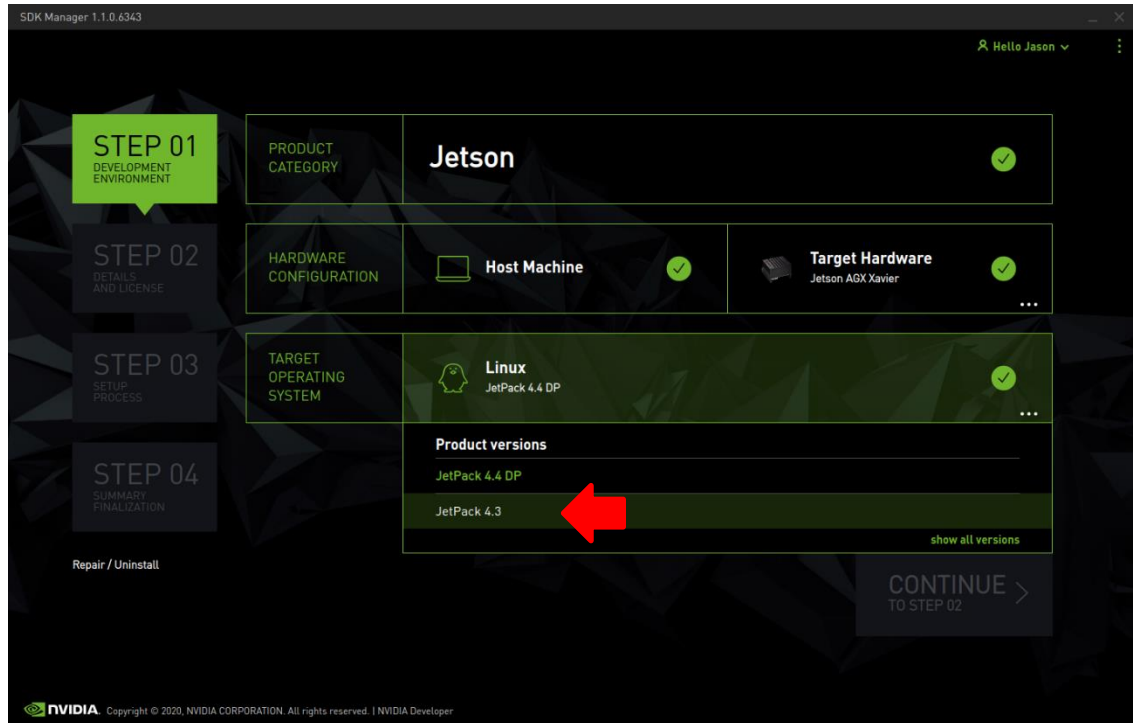
4. Click on the JetPack 4.3 version to make it the selected version, as shown in the following screenshot:
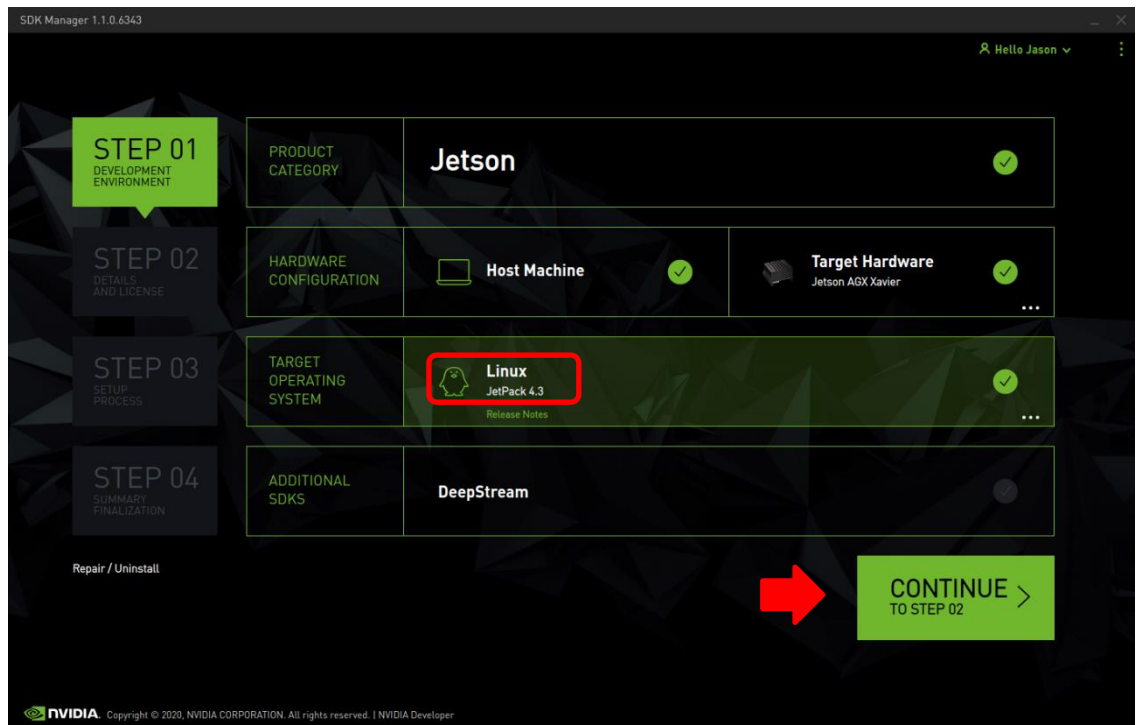


5. Press Continue to advance to the STEP 02 – DETAILS AND LICENSE screen and click I accept the terms and conditions of the license agreements at the bottom of the window to accept the licenses, as shown in the following screenshot:
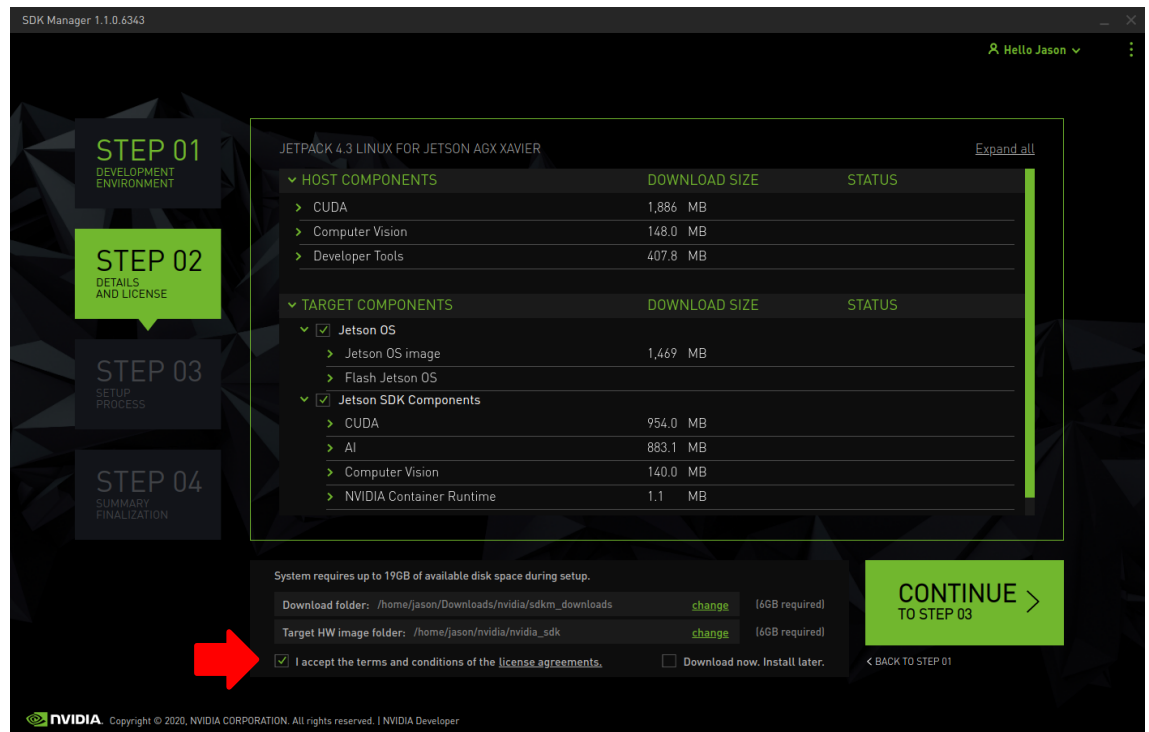
6.  Press Continue to advance to the STEP 03 – SETUP PROCESS screen and SDK Manager will begin downloading all the software components from NVIDIA, as shown below:
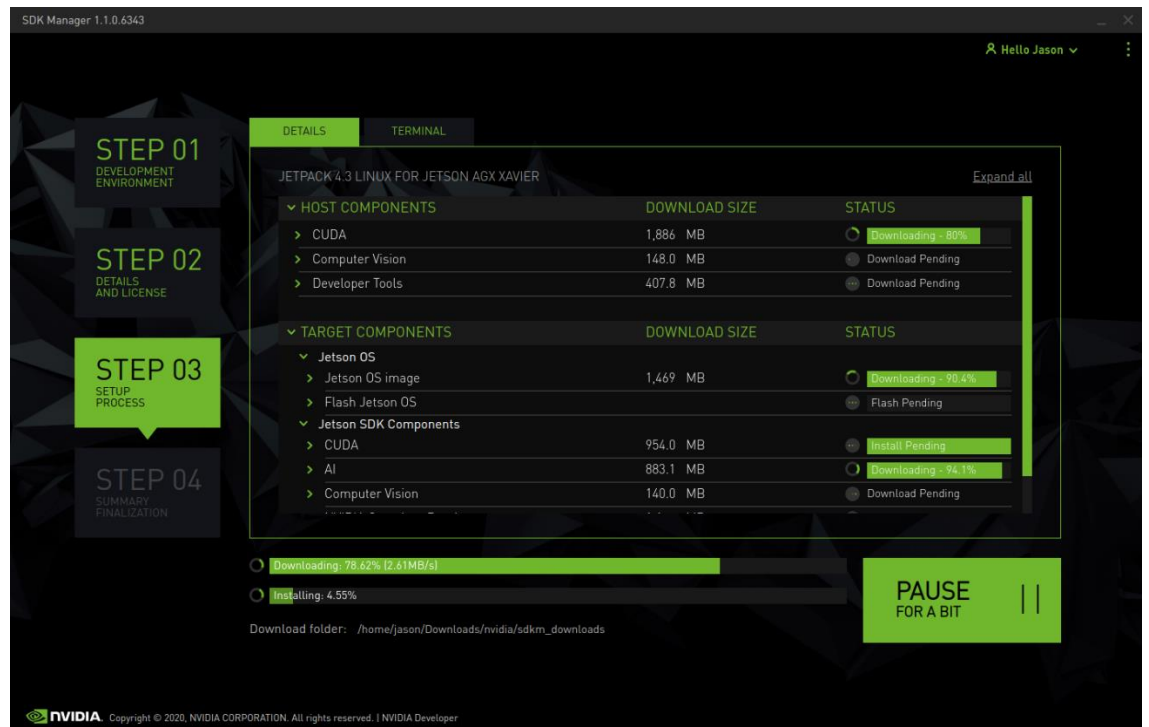


Depending on your network's speed, the download can take 30 minutes or more.  During the download and setup, a detailed output log can be viewed under the TERMINAL tab.

7.  Once the download and setup has completed, you will be presented with a Flash dialog. Notice the Automatic setup pulldown menu as shown in the following dialog:
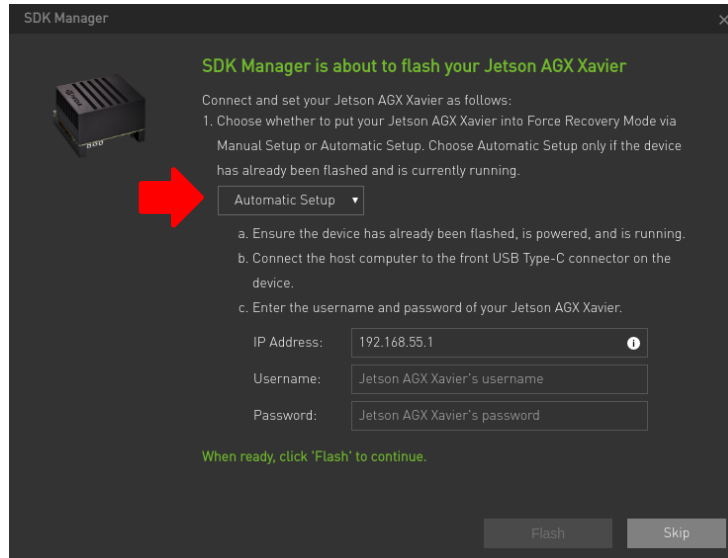
Click on Automatic setup and change the setting to Manual Setup as shown in the following dialog:



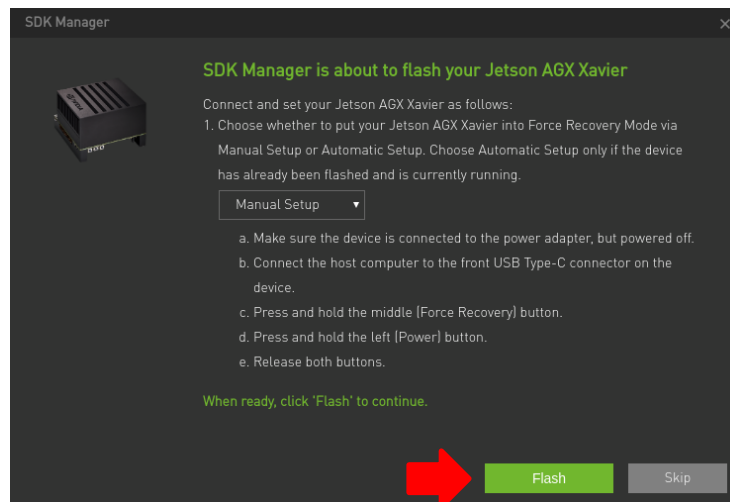Follow the steps listed on the Flash dialog to place the Jetson AGX Xavier into recovery mode and then click the Flash button to begin flashing.

**NOTE**

*Issue the `lsusb` command on the host to verify that a single `NVidia Corp.` entry with device ID `0955:7019` is displayed; this is the best indication that the Xavier has successfully entered recovery mode and is ready for flashing.*

8. Midway through flashing you will need to connect the Xavier to a monitor, keyboard and mouse to click through the System Configuration dialogs. Accept the license and then choose values for all settings that are appropriate for your site.

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.
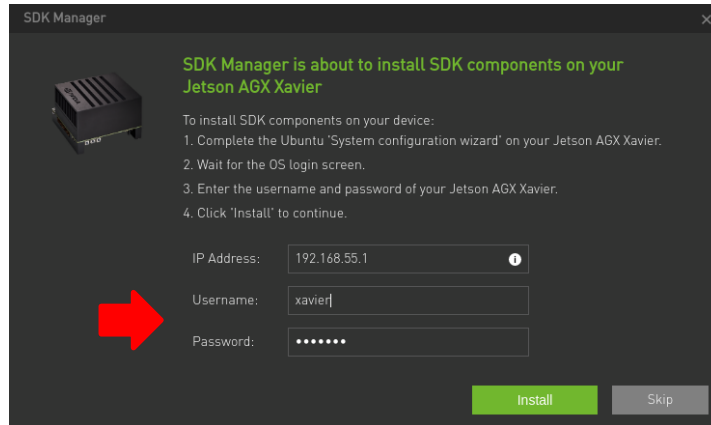


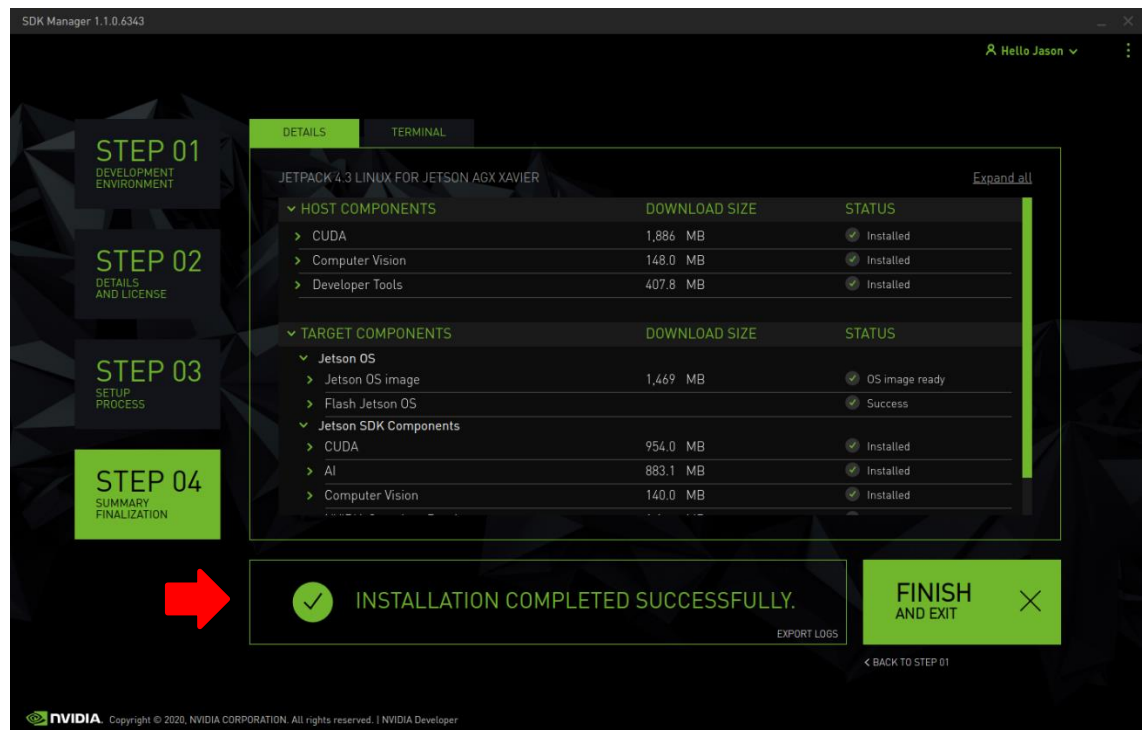Enter values for each field matching those entered into the System Configuration dialog in the previous step and then click Install to continue flashing.

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 Xavier, and while the Xavier is still booted with the NVIDIA stock kernel, log into the Xavier using the username and password chosen during flashing and run the following commands:

```
$ sudo nvpmodel -m 0
$ sudo nvpmodel -q
```

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

```
NV Power Mode: MAXN
0
```

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

## 3.5.     Copy RedHawk packages to the Xavier

**NOTE**

*This section assumes you are using the RedHawk 7.5.2 for Jetson AGX Xavier 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 Xavier, insert the *RedHawk 7.5.2 for Jetson AGX Xavier* optical disc into the Ubuntu host's optical drive and it should automatically mount under the `/media/$USER` directory (e.g. `/media/jane/RedHawk 7.5.2 aarch64`).

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

```
$ cd /media/jane/RedHawk 7.5.2 aarch64/Packages
$ scp *.deb user@host:/tmp
```

**NOTE**

*Use the username, hostname and password chosen during flashing.*

## 3.6.     Install RedHawk packages on the Xavier

To install the RedHawk packages, log into the Xavier (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:*

- *cryptsetup: WARNING: could not determine root device…*
- *Warning; /sbin/fsck.rootfs doesn't exist, can't install to…*
- *N: Can't drop privileges for downloading as file…*
- *insserv: warning: script nv-run-date, nv-run-once-install, …*
- *Warning: couldn't identify filesystem type for fsck hook…*
- */sbin/ldconfig.real: Warning: ignoring configuration file…*

*You can safely ignore these warnings as they are not relevant to RedHawk 7.5.2 on Jetson AGX Xavier embedded installations.*

At this point all the RedHawk packages, including the RedHawk kernels, should have installed successfully on the Xavier. 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

    LABEL primary
        MENU LABEL primary kernel
        LINUX /boot/Image
        INITRD /boot/initrd
        APPEND ${cbootargs} quiet
    ```

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-4.9.140-rt94-r32.3.1-tegra-RedHawk-7.5.2-trace
        INITRD /boot/initrd.img-4.9.140-rt94-r32.3.1-tegra-RedHawk-7.5.2-trace
        APPEND ${cbootargs} quiet
    ```

    **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 Xavier 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
    ```

Note that if your Xavier 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 Xavier 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:

```
4.9.140-rt94-r32.3.1-tegra-RedHawk-7.5.2-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.    Xavier hardware not optimized for real-time at boot

By default, at boot time the Xavier hardware is optimized for maximum throughput instead of maximum real-time performance; in this mode real-time applications will experience less determinism and higher latencies during execution.

To run real-time workloads on the Xavier with high determinism and significantly lower latencies, RedHawk provides a new /proc/rt_safe_mask control file to give users the ability to select the level of real-time performance on the Xavier.  For example, invoking the following command as the root user will maximize real-time performance:

```
# echo 255 > /proc/rt_safe_mask
```

The above command enables maximum real-time performance on all 8 CPU cores on the Xavier, and this is required to achieve the maximum real-time performance of the whole system. Note that 255 is a bit mask (i.e. hex 0xFF or binary 0b11111111) indicating that each of the 8 individual CPU cores is to be configured for maximum real-time performance.

Users may choose to exclude one or more CPU cores from the bit mask to sacrifice a degree of real-time performance in exchange for improved throughput; for example, invoking the following command as the root user excludes CPU core 0 from the bit mask:

```
# echo 254 > /proc/rt_safe_mask
```

The above command will still improve real-time performance compared to the default setting of 0 (where no CPU cores are optimized for real-time) but the improvement will not be as good as when all CPU cores are enabled for real-time.  However, in this mode throughput will improve.

Finally, users can also return to the default maximum throughput state of the Xavier (where no CPU cores are optimized for real-time) by invoking the following command as the root user:

```
# echo 0 > /proc/rt_safe_mask
```

## 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 Xavier has started with graphics displaying you can safely stop the service to improve real-time performance.

## 4.3. RedHawk kernels and OEM verification failures

Booting any of the RedHawk kernels on the Xavier will result in the following warnings being displayed almost immediately after the boot process begins:

```
[0014.855] I> Validate kernel ...
[0014.855] I> T19x: Authenticate kernel (bin_type: 37), max size 0x5000000
[0014.856] E> Stage2Signature validation failed with SHA2!!
[0014.856] C> OEM authentication of kernel header failed!
[0014.857] W> Failed to validate kernel binary (err=1077936152)
[0014.857] W> Security fuse not burned, continue to boot
```

These warnings can be safely ignored because OEM security fuses are not burned by default on Xavier hardware platforms.  For more information, refer to the *NVIDIA Jetson AGX Xavier Fuse Specification* available on the NVIDIA Developer website.

## 4.4. 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.

## 4.5.　　Kernel debugger not accessible via serial console

The KGDB kernel debugger for the Xavier is currently unable to communicate over the serial console interface (`/dev/ttyTCU0`).  Concurrent is working with NVIDIA to address this issue and hopefully it will be corrected in future NVIDIA kernels.

## 4.6.　　Warnings accessing `/dev/mem`

Attempting to directly access `/dev/mem` on the Xavier 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.

## 4.7.　　Change to PAM Capability module directory

The PAM Capability module directory that is documented in some versions of the *RedHawk User's Guide* is incorrect for the Xavier. The correct PAM Capability module location should be shown as:

```
/lib/aarch64-linux-gnu/security/pam_capability.so
```

If you wish to use the PAM Capability feature with services in the `/etc/pam.d` directory, then you should refer to the PAM Capability module as follows:

```
session required /lib/aarch64-linux-gnu/security/pam_capability.so
```

# 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`.

Use the following format for the repository definition of the RedHawk 7.5.2 for Jetson AGX Xavier product:

```
deb http://redhawk.concurrent-rt.com/ubuntu/login/passwd/redhawk/agx-xavier 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/agx-xavier 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 4.6 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.

## A.3.   Install RedHawk Packages

Now that the setup is done, simply issue the following command to install all the latest RedHawk 7.5.2 packages (and NightStar 4.6 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.


## 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.