Installation of RedHawk™ 7.5 on the NVIDIA® DRIVE™ PX 2 AutoChauffeur (P2379)

Release Notes

February 21st, 2019

0898003-7.5-PX

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 on the NVIDIA® DRIVE™ PX 2 AutoChauffeur (P2379). The directions in this document supersede all others – they are specific to installing the software on Concurrent Real-Time's RedHawk systems.

2. Requirements

- DRIVE PX 2 AutoChauffeur (P2379) with Internet access
- An x86_64 host system running Ubuntu 14.04 or 16.04 with Internet access
- *RedHawk Revision 7.5 for the DRIVE PX 2* optical media disk

3. Installation

3.1.   Download DriveInstall on the host system

On an x86_64 host system running Ubuntu 14.04 or 16.04, download version 5.0.5.0bL of DriveInstall from the NVIDIA website via this URL (login is required):

https://developer.nvidia.com/nvidia-drive-downloads

NOTE

DriveInstall version 5.0.5.0bL can be downloaded via the Downloads Archive link located near the bottom of this web page in the Additional Resources section. Do not download and install the newer SDK Manager because it installs firmware that impacts real-time performance and it is not supported by RedHawk.

Place the downloaded file named `DriveInstall_5.0.5.0bL_SDK_b3.run` into a `drive` directory in the home directory of the current user. Note that logging in as the root user is *not* required to run the DriveInstall script.

3.2.   Perform a full DriveInstall installation of the DRIVE PX 2

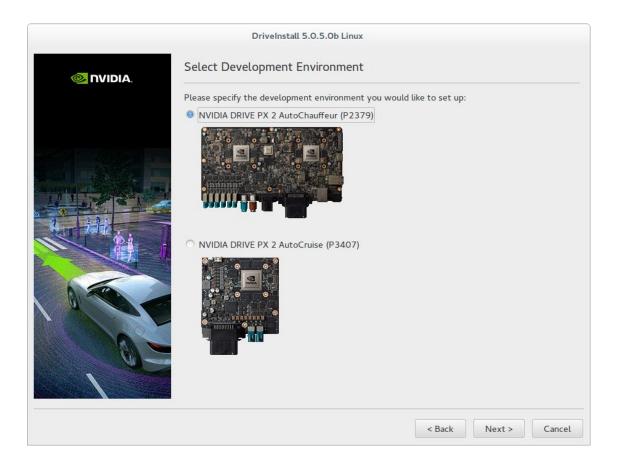Change into the `drive` directory and run the DriveInstall tool as follows:

```
$ chmod +x DriveInstall_5.0.5.0bL_SDK_b3.run
$ ./DriveInstall_5.0.5.0bL_SDK_b3.run
```

At this point, you can generally follow the DriveInstall instructions that are available on the NVIDIA website via this URL:

https://docs.nvidia.com/drive/driveinstall_docs/

However, make sure to follow the directions listed below during the installation:

1. On the Select Development Environment screen, select the NVIDIA DRIVE PX 2 AutoChauffeur (P2379) development environment option as shown in the following screenshot:

2. Press Next to continue to the next screen and ensure that the DriveInstall Linux
   Components Manager is configured to flash a <u>Full</u> installation as shown below before
   proceeding:

Note that if you are performing the DriveInstall installation process for the first time, DriveInstall
will download a large amount of software to perform the full installation.

Also, throughout the installation process you will be prompted for interactive input in dialog
windows and the terminal window that is spawned from the DriveInstall tool; make sure to pay
attention to these prompts as the release will pause until the instructions in the prompts are
performed.

Complete all remaining steps in the installation process and then proceed to the next section in
this document to begin RedHawk product installation.

NOTE

*Once DriveInstall flashing has completed, make sure to connect to both the Tegra A
and Tegra B graphical displays and click through each of the Start-Up Wizard dialogs.
Failure to click through the Wizards completely will cause them to restart upon
each reboot and this may impact the DRIVE PX 2's real-time performance.*

3.3.   Copy RedHawk packages to the DRIVE PX 2

NOTE

This section assumes you are using the RedHawk 7.5 for DRIVE PX 2 optical media disc. Alternatively, if you wish to install RedHawk from the network refer to Appendix B, "Install RedHawk from Concurrent Real-Time Repositories."
Once installed continue with section 3.5 below to finalize the installation.

After DriveInstall has completed the Ubuntu installation on the DRIVE PX 2, insert the *RedHawk 7.5 for DRIVE PX 2* 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 aarch64`).

Change to the mounted directory on the host and use networking to copy the RedHawk packages into both Tegra A and Tegra B in the DRIVE PX 2, as illustrated by the following commands:

```
$ cd /media/jane/RedHawk 7.5 aarch64/Packages
$ scp *.deb nvidia@tegraA:/tmp
$ scp *.deb nvidia@tegraB:/tmp
```

NOTE

During the DriveInstall installation, two users are automatically created: ubuntu and nvidia. The password for each user is the matching user name (e.g. login nvidia, password nvidia).

Replace `tegraA` and `tegraB` with the actual IP address or hostname of Tegra A and Tegra B. The RedHawk packages will now be sitting in the `/tmp` directory on both Tegra A and Tegra B.

3.4.   Install RedHawk packages on the DRIVE PX 2

To install the RedHawk packages, log into Tegra A (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-…', etc.*

You can safely ignore these warnings as they are not relevant to RedHawk 7.5 on DRIVE PX 2 embedded installations.

Repeat these commands while logged into Tegra B to install RedHawk on Tegra B as well.

At this point all the RedHawk packages, including the RedHawk kernels, should have installed successfully on Tegra A and Tegra B in the DRIVE PX 2. However, additional steps in the next section are required before you can boot into the RedHawk kernels.

3.5. Flash desired RedHawk kernel using `bootburn.sh`

The RedHawk kernels were placed into the `/boot` directory during the package installation in the previous step, however the RedHawk kernel image files cannot be booted directly from `/boot`.

Perform the following steps on the x86_64 host machine running Ubuntu 14.04 or 16.04 to flash the desired RedHawk kernel into one of the Tegra systems in the DRIVE PX 2.

NOTE

The example instructions below flash the RedHawk Trace kernel into Tegra A.

1. Change into the DriveSDK *kernel* directory:

   ```
   $ cd ~/NVIDIA/Drive/5050bL_SDK/DriveSDK/drive-t186ref-linux/kernel
   ```

2. Copy over the desired RedHawk kernel boot image from Tegra A:

   ```
   $ scp nvidia@tegraA:/boot/Image-4.9.38-rt25-5.0.5.0bL-RedHawk-7.5-trace .
   ```

   Replace *tegraA* with the actual IP address or hostname of Tegra A.

3. Change into the DriveSDK foundation directory:

   ```
   $ cd ~/NVIDIA/Drive/5050bL_SDK/DriveSDK/drive-t186ref-foundation
   ```

4. Open the storage file for Tegra A in a text editor:

   ```
   $ vi ./pct/p2379c01-t186a/linux/linux_storage.cfg
   ```

   Replace `p2379c01-t186a` with `p2379c01-t186b` to edit the storage file for Tegra B.

5. Update the `imagepath` variable to point to the selected kernel.

   Search for the `imagepath` variable and change the final directory and kernel image name from these values:

   ```
   imagepath=.../kernel-rt_patches/Image
   ```

   To these values:

   ```
   imagepath=.../kernel/Image-4.9.38-rt25-5.0.5.0bL-RedHawk-7.5-trace
   ```

   NOTE

   Make sure to leave all the leading path elements unchanged.

   Save the updated storage file and exit the text editor.

6. Prepare an updated flash binary for Tegra A incorporating the selected kernel:

   ```
   $ make -f Makefile.bind BOARD=p2379c01-t186a
   ```

   Replace `p2379c01-t186a` with `p2379c01-t186b` to prepare for Tegra B flashing.

7. Change into the bootburn script directory:

```
$ cd ./utils/scripts/bootburn
```

8. Use the `bootburn.sh` script to flash the selected kernel onto Tegra A:

```
$ sudo ./bootburn.sh -b p2379c01-t186a -u 1_kernel 1_ramdisk -x /dev/ttyUSB1
```

Replace `p2379c01-t186a` with `p2379c01-t186b` to flash the kernel onto Tegra B.

The flashing process will run for several minutes before it is complete. During the flash both Tegra A and Tegra B in the DRIVE PX 2 will be reset, even though you are only flashing one of them at a time during the above steps.

Repeat these steps from the beginning, replacing all Tegra A references with Tegra B, to flash a RedHawk kernel into Tegra B. Once both Tegra A and Tegra B have been flashed with RedHawk kernels, the RedHawk installation on the DRIVE PX 2 is complete.

NOTE

After RedHawk installation, it is recommended that modified `linux_storage.cfg` files are restored to their original states with their original `imagepath` variable settings, and step 6 above is repeated for both Tegra A and Tegra B to prepare original flash binaries; this will ensure that subsequent DriveInstall flashing works as intended.

3.6.   Verify RedHawk kernel installation

The DRIVE PX 2 should now be running the selected RedHawk kernel. To verify this, issue the following command:

```
$ uname -r
```

On Tegra A and Tegra B you should see a line displayed like the following:

```
4.4.38-rt25-5.0.5.0bL-RedHawk-7.5-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 Tegra A and Tegra B. 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:

```
$ sudo 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:

```
$ sudo cpu -u 1,2
```

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

4.2.　Lockdep warnings with debug kernels

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 to address these warnings and they will be eliminated in future versions of RedHawk.

4.3.　Warnings accessing `/dev/mem`

Attempting to directly access `/dev/mem` on the Tegra A and Tegra B is not permitted and will result in errors output to the systems console. For example, invoking the `dmidecode` application while a RedHawk kernel is running will output an error like the following:

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

Note that if the stock NVIDIA kernel is running, you may see hundreds of lines of `SError` messages output to the console, similar to the following:

```
CPU4: SError detected, daif=140, spsr=0x20000000, mpidr=800001020
CPU4: SError detected, daif=140, spsr=0x40000145, mpidr=800001020
ROC:IOB Machine Check Error:
Address Type = Secure DRAM
Address = 0x0 (Unknown Device)
...
```

These errors are only informational and can be ignored.

4.4.   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 DRIVE PX 2. The correct PAM Capability module location for Tegra A and Tegra B 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
```

4.5.   Missing package sources with new host installations

Now that Ubuntu 18.04 has been released, new Ubuntu 16.04 installations may fail to include all the necessary package sources for DriveInstall to function correctly.

Before using DriveInstall, verify that the following package source entries are include in the system's `/etc/apt/sources.list` file:

```
deb http://us.archive.ubuntu.com/ubuntu/ xenial-updates main restricted
deb http://us.archive.ubuntu.com/ubuntu/ xenial-security main restricted
deb-src http://us.archive.ubuntu.com/ubuntu/ xenial universe
```

If any of these package source entries are missing on your Ubuntu 16.04 system, then you will need to add them for DriveInstall to work correctly.

4.6.   Resource temporarily unavailable during DriveInstall

Some versions of the kernel running on Ubuntu 16.04 host systems are not able to correctly communicate with the PX2 serial consoles. For example, DriveInstall may abort during flashing when kernel version 4.15.0-45 is running on the host, whereas kernel version 4.15.0-29 does not appear to have any problems.

If DriveInstall exits with an error during flashing, check the log to see if any *Resource temporarily unavailable* messages appear; if these messages are present, revert to an earlier kernel version on the host (e.g. 4.15.0-29), reboot the host system and then attempt to run DriveInstall again.

4.7.   Stop Firefox from automatically starting at boot

DriveInstall configures Firefox to automatically start at boot time in both Tegra A and Tegra B. To disable this behavior, issue the following command on both Tegra A and Tegra B:

```
$ sudo rm /etc/xdg/autostart/WelcomeHTML.desktop
```

Note that you will need to repeat this operation every time DriveInstall is used to perform a full installation of the DRIVE PX 2.

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 for DRIVE PX 2 product:

```
deb [arch=arm64] http://redhawk.concurrent-rt.com/ubuntu/login/passwd/redhawk/px2 7.5 core
```

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

```
deb [arch=arm64] http://redhawk.concurrent-rt.com/ubuntu/LR12345/t0ps3cr3t/redhawk/px2 7.5 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/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 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.