Teledyne DALSA • 880 Rue McCaffrey • St-Laurent, Québec, H4T 2C7 • Canada
http://www.teledynedalsa.com/imaging/

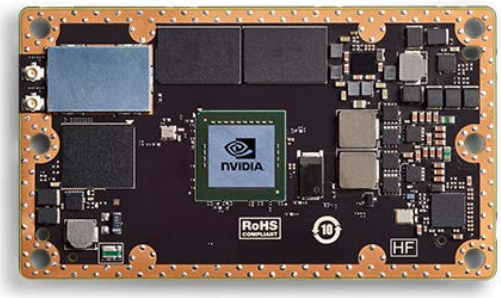
GVF-AN001-V1: GigE-V Framework for Linux App Note

Getting Started with GigE-V Framework for Linux and the NVIDIA Jetson (TK1/TX1/TX2) Embedded Platform

For Teledyne DALSA GigE-Vision cameras

Overview

Teledyne DALSA's Software SDK: **GigE-V Framework for Linux** is compatible with the NVIDIA Jetson platform running Linux for Tegra (L4T). This guide documents how to use the NVIDIA JetPack installer to flash the L4T OS (and optional developer tools) on the Jetson TK1/TX1/TX2 and install the Teledyne DALSA GigE-V Framework for Linux.



The Jetson platforms are pre-flashed with the L4T OS; if you are not reflashing the OS using the JetPack, refer to the section Installing GigE-V Framework on the Jetson.

If using the JetPack, refer to the section Using the NVIDIA JetPack Installer to Flash the Jetson with an L4T OS Image.

Installation using the JetPack and of the GigE-V Framework on the Jetson assumes that an internet connection is available to download the required packages and dependencies.

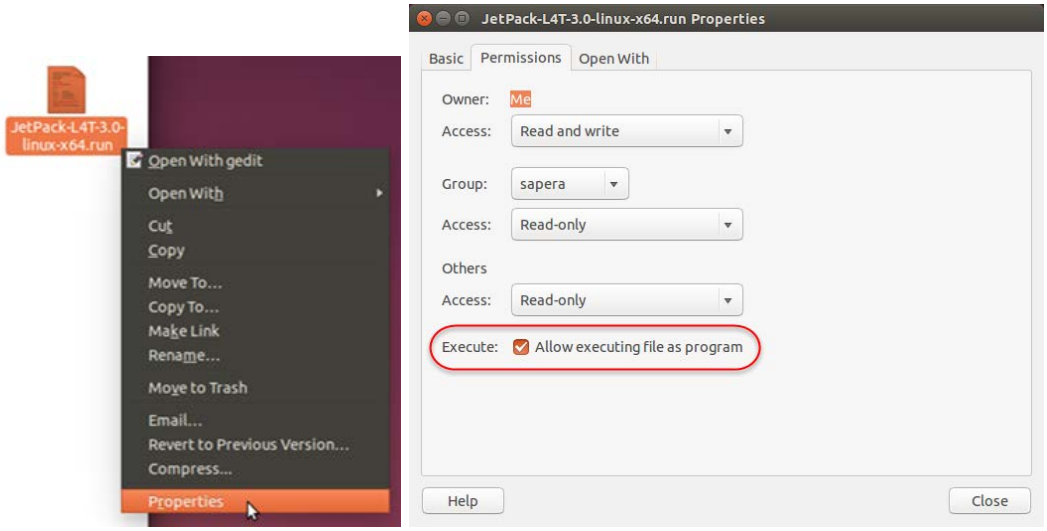
For documentation on the Teledyne DALSA GigE-V Framework API, refer to the GigE-V Framework for Linux 32/64-Bit Programmer's Manual.

Teledyne DALSA's **GigE-V Framework for Linux SDK** DALSA can be downloaded from the Teledyne DALSA web site:
https://www.teledynedalsa.com/imaging/support/downloads/sdks/

Using the NVIDIA JetPack Installer to Flash the Jetson with an L4T OS Image

Download the latest version of the JetPack from the Nvidia website to the host Linux development machine:
https://developer.nvidia.com/embedded/jetpack

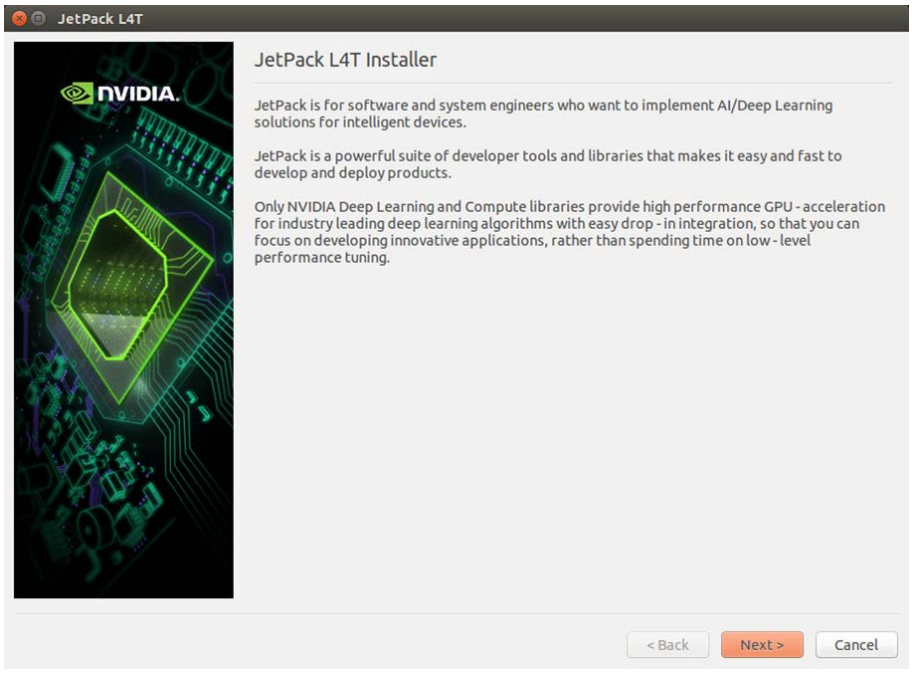
Right-click to open the Properties dialog of the *JetPack-L4T-3.0-linux-x64.run* file and change the Permissions settings to **Allow executing file as a program**.



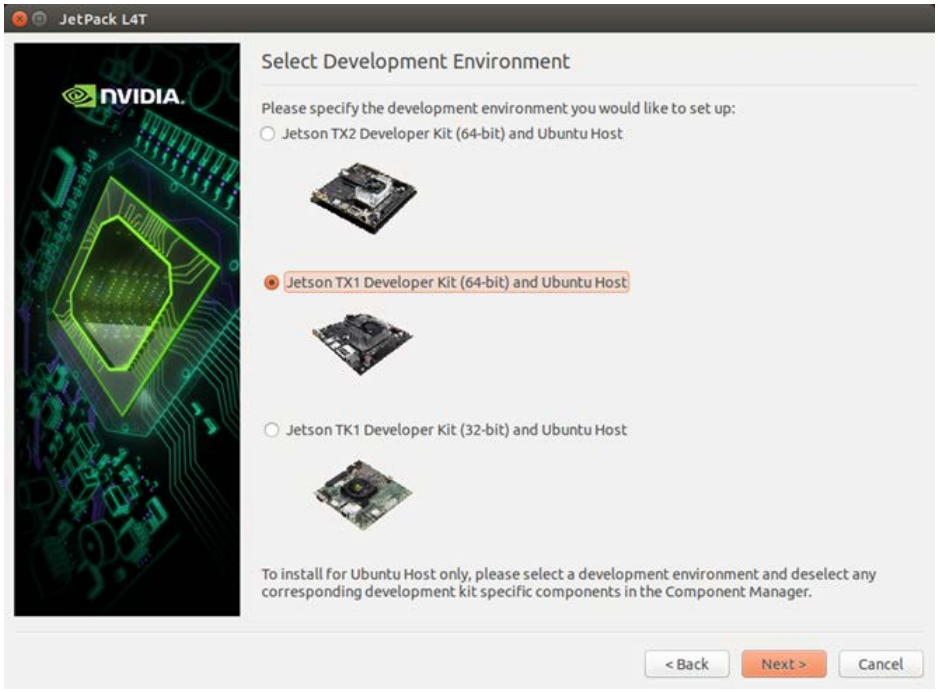
In a terminal window, navigate to the directory containing the *.run* file and use the following command:

```
sapera@sapera-X8ST3:~$ sudo ./JetPack-L4T-3.0-linux-x64.run
Creating directory _installer
Verifying archive integrity... All good.
Uncompressing JetPack  100%
```

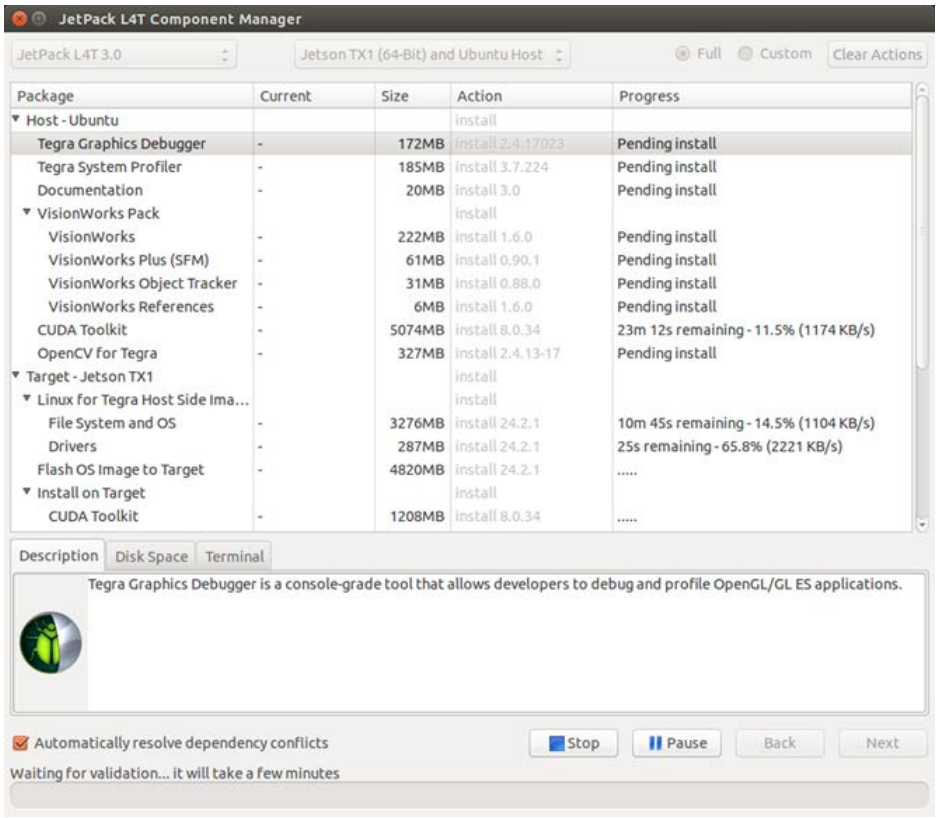
The Installer dialog will open. Follow the installation instructions presented.



Select the required development environment:

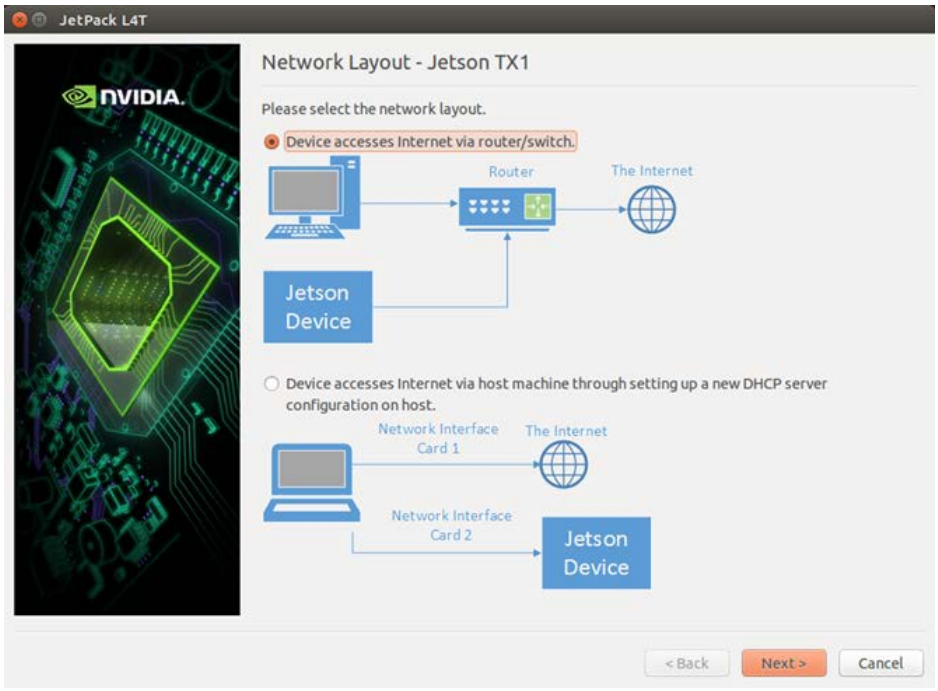


Select the required components to install:

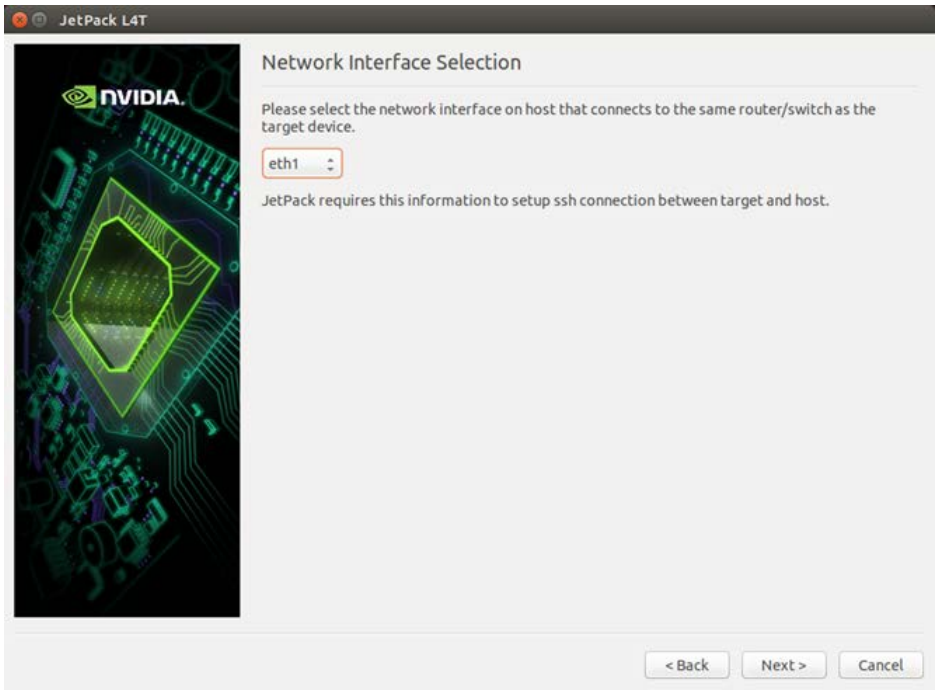


When the newly generated image is complete, proceed to flash the L4T OS on the target Jetson platform

Select the network layout used to connect the host PC to the Jetson.



Select the network interface on the host that connects to the Jetson.



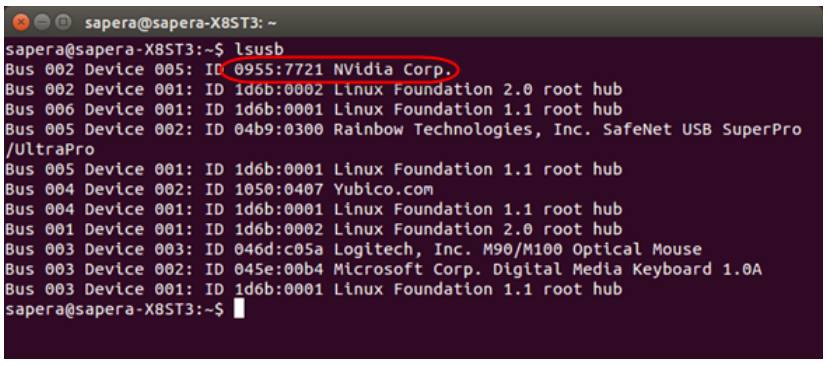
The Jetson must now be placed in Force USB Recovery Mode.

To set the board in recovery mode, follow these steps.

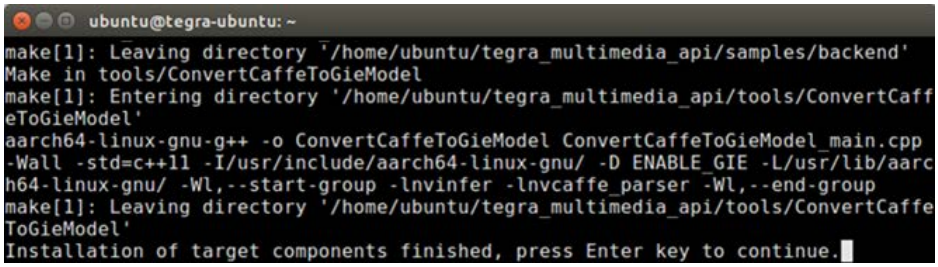
- Power down the device. If connected, remove the AC adapter from the device. The device must be powered off, not in a suspend or sleep state.
- Connect the Micro-B plug on the USB cable to the Recovery (USB Micro-B) Port on the device and the other end to an available USB port on the host PC.
- Connect the power adapter to the device.
- Press and release the Power button to power on the device.
- Press and hold the Force Recovery button: while pressing the Force Recovery button, press and release the Reset button; wait two seconds and relase the Force Recovery button.

To verify if the board is in recovery mode, run the command "lsusb" from the host computer. The board will be displayed as:

"Bus xxx Device xxx: ID 0955:xxxx NVidia Corp."



The terminal window on the host displays the progress of flashing the OS onto the target Jetson.



Installing GigE-V Framework on the Jetson

The GigE-V Framework for Linux is distributed as a compressed tar archive, with file type ".tar.gz". The naming convention e of this archive is:

GigE-V-Framework_<architecture>_<Version#>.<Build#>.tar.gz

For the Jetson, the following architectures are supported:

- **ARM AArch64**: 64-bit ARMv8
- **ARM hard float** : 32-bit ARMv7 with hardware floating point

The following table lists the required architecture for the Jetson model.

Model	Architecture
Jetson TX2	aarch64
Jetson TX1	aarch64
Jetson TK1	armhf

For example, the ARM files for version 2.02 build 0.0105 are:

- GigE-V-Framework_aarch64_2.02.0.0105.tar.gz,
- GigE-V-Framework_ARMhf_2.02.0.0105.tar.gz, and

At this time, only target systems configured for self-hosted development are supported. At installation time, parts of the API are compiled and linked to the run-time libraries found on the target system. This reduces the risk of an installation package failing to work with a target system due to mismatched versions of run-time libraries. As a consequence of this, certain pre-requisites are required for successful installation.

Required Packages

The following table contains a list of packages needed. In some cases the names are different or need to be searched for using a pattern due to distribution-dependent naming conventions.

Package Name	Purpose
libglade2-0 libglade2-dev	Load ".glade" UI definition files at application runtime
libgtk-3-dev	Compile and link GigE Vision Device Status tool
The following packages are typically already installed as part of the Linux for Tegra installation:	
gcc (top level package for C compiler) and g++ (top level package for C++ compilation)	S/W Development (Compilers/Linkers etc….)
libpcap0.8	Packet capture (for PF_PACKET interface support)
libx11-dev libxext-dev	Compile and Link Demos using X11 for Image display
libcap2 or libcap-ng0	Capabilities setting for CAP_NET_RAW and CAP_SYS_NICE support

For example,

```
sudo apt-get install <package name>
```

Note, if you are unable to locate a specific package, a regular expression can be used to try to find a suitable alternative package. For example,

```
sudo apt-get install libpcap*
```

GigE-V Framework Installation

To install the GigE-V Framework for Linux from its compressed tar archive file, start by copying it to a base directory, usually the HOME directory of the user installing it, and extracting the files.

For example:

```
cp GigE-V-Framework_aarch64_2.02.01.0128.tar.gz $HOME
cd $HOME
tar –zxf G GigE-V-Framework_aarch64_2.02.01.0128.tar.gz
```

Then, change to the directory DALSA and run the installer script.

```
cd DALSA
./corinstall
```

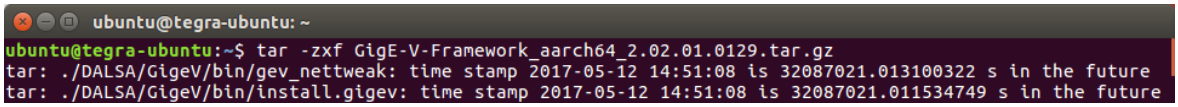
For detailed information on available examples, tools and the GigE-V Framework API, refer to the GigE-V Framework for Linux 32/64-Bit Programmer's Manual.

Updating the Jetson Date and Time

Note: Some computer systems do not retain time and date settings after power cycling. This is particularly true of embedded systems. Installation of the GigE-V Framework for Linux can be affected by misconfigured time and date settings if the files being installed are timestamped in the future when compared to the current system time.

In such instances, it may be necessary to install/enable an NTP (Network Time Protocol) capability in order to keep the time and date settings current.

For example, the following message indicates the timestamp of the file is in the future:



As an example, the ntpdate package can be installed and configured to use an available local or online NTP server to synchronize the system clock.

To install and configure the ntpdate package, use the following commands:

```
sudo apt-get install ntpdate
sudo ntpdate 140.165.161.1
```

It may be necessary to stop the service before initiating the update; for example:

```
sudo service ntp stop
sudo ntpdate time.nist.gov
sudo service ntp start
```

Additionally, the */etc/ntp.conf* file can be updated to include the required NTP server. For example, the following lines can be modified to add the NTP server:

```
# Use Ubuntu's ntp server as a fallback.
pool ntp.ubuntu.com
140.165.161.1
```