Jetson TX1 Developer Kit User Guide

October 2015

Notice

ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OR CONDITION OF TITLE, MERCHANTABILITY, SATISFACTORY QUALITY, FITNESS FOR A PARTICULAR PURPOSE AND ON-INFRINGEMENT, ARE HEREBY EXCLUDED TO THE MAXIMUM EXTENT PERMITTED BY LAW.

Information furnished is believed to be accurate and reliable. However, NVIDIA Corporation assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent or patent rights of NVIDIA Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. NVIDIA Corporation products are not authorized for use as critical components in life support devices or systems without express written approval of NVIDIA Corporation.

Trademarks

Copyright

NVIDIA Corporation | 2701 San Tomas Expressway | Santa Clara, CA 95050 | +1 408 486-2000 | www.nvidia.com

Contents

Introduction

The NVIDIA® Jetson TX1 Developer Kit is a full-featured development platform for visual computing. It is ideal for applications requiring high computational performance in a low power envelope. The Jetson TX1 Developer kit is designed to get you up and running quickly: It comes pre-flashed with a Linux environment, includes support for many common APIs, and is supported by NVIDIAs complete development tool chain. The board exposes many standard hardware interfaces, enabling a highly flexible and extensible platform.

Go to http://developer.nvidia.com/embedded-computing for access to software updates and the developer SDK supporting the OS image and host development platform that you want to use. The SDK includes an OS image that you will load onto your device, developer tools, supporting documentation, and code samples to help you get started.

Getting Started

Individual development efforts will vary and may result in modifications to the system configuration. It is recommended that you begin with the basic system configuration (as shipped) to ensure proper system operation prior to any further development.

Assumptions

- You have a Jetson TX1 Tegra Developer System, equipped with the NVIDIA® Tegra® TX1 processor.
- Your developer system should be cabled as follows:
  - Serial cable plugged into the serial port on the target connected to your Linux host directly or through a serial-to-USB converter. (To setup serial console on the Linux host.)
  - (Not included in the developer kit) To connect USB peripherals such as keyboard, mouse, and [optional] USB/Ethernet adapter (for network connection), a USB hub should be connected to the working USB port on the Jetson TX1 system.
  - An HDMI cable plugged into the target which is connected to an external HDMI display.
  - An Ethernet cable plugged into the on board Ethernet port.

Powering Up the Tegra Device

1. Connect a USB keyboard to the USB Type A connector of your device.
2. Connect an HDMI-compatible display to the HDMI connector on your device.
3. Connect the AC adapter supplied in your kit to the power connector of your device. Use the supplied AC adapter since it is appropriately rated for your kit.
4. Plug the power adapter into an appropriately rated electrical outlet.
5. They system should power on. If not, press and release the power button on the device.

Login Credentials

> Username: `ubuntu`

> Password: `ubuntu`

Force Recovery Mode

To update your system, you will need to be in Force USB Recovery Mode so you can transfer system software to the developer board. When in Force USB Recovery Mode, you are able to update system software and write the boot loader, boot configuration table (BCT), and partition configuration to the Tegra device.

See the Developer SDK documentation for OS specific instructions when updating system software on your developer board.

To place system in Force USB Recovery Mode:

1. 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.
2. Connect the Micro-B plug on the USB cable to the Recovery (USB Micro-B) Port on the device [4] and the other end to an available USB port on the host PC.
3. Connect the power adapter to the device.
4. With the system powered on, press and release the POWER button, if necessary; press and hold the RECOVERY FORCE button; while depressing the RECOVERY FORCE button, press and release the RESET button; wait two seconds and release the RECOVERY FORCE button.

Flashing the Boot Loader and Kernel

This section describes the steps required to flash and boot the target Tegra device. It also provides usage information for the `flash.sh` helper script.

Flash Procedure

First, flash the board with the boot loader and kernel, and, optionally, flash the rootfs to internal eMMC.

Prerequisites

The following directories must be present:

- /bootloader—boot loader plus flashing tools (NvFlash, CFG, BCTs, etc.)
- /kernel—a kernel zImage /vmlinux.uimg, DTB files, and kernel modules
- /rootfs—the root file system that you download (This directory starts empty and you populate it with the sample file system.)
- /nv_tegra—NVIDIA® Tegra® user space binaries and sample applications

You must also have the USB cable connected to the recovery port prior to running the commands listed in the procedure. For more information, see the Requirements topic in this section.

To flash the boot loader and kernel

1. Put the target board into reset/recovery mode. Do so by first powering on the board and then holding the recovery button, and then pressing the reset button as described in the *Quick Start Guide* for the board.

2. Run the `flash.sh` script that is in the top level directory of this release. The script must be supplied with the target board (`jetson-tx1`) for the root file system:

   ```
   $ sudo ./flash.sh <platform> <rootdev>
   ```

   - If the root file system will be on a USB disk, execute the script as follows:

     ```
     $ sudo ./flash.sh <platform> sda1
     ```

     Note: If a SATA device is connected, that device enumerates as `sda1`.

   - If the root file system will be on an SD card, execute the script as follows:

     ```
     $ sudo ./flash.sh <platform> mmcblk1p1
     ```

   - If the root file system will be on the internal eMMC, execute the script as follows:

     ```
     $ sudo ./flash.sh <platform> mmcblk0p1
     ```

     Where <platform> is `jetson-tx1`.

   The above examples are for u-boot. For fastboot, add the following argument:

```
-L <PATH_TO_FASTBOOT_BIN_FILE>
```

For example:

```
$ sudo ./flash.sh –L bootloader/<platform>/fastboot.bin <platform>
<rootdev>
```

The boot loader and kernel will load.

For more information on U-Boot, see the U-Boot Guide chapter of this document.

Flash Script Usage

You can find the most up-to-date usage information by running `flash.sh –h` (using the `flash.sh` script included in the release). The basic usage information is as follows.

Usage

```
sudo ./flash.sh [options] <platform> <rootdev>
```

Where you specify the required parameters and one or more of the options shown in the following table.

Parameters	Description	
<platform>	Is `jetson-tx1`.	
<rootdev>	Is one of following:	
	mmcblk0p1	Specifies internal eMMC.
	mmcblk1p1	Specifies external SDCARD.
	sda1	Specifies external USB device (such as, USB memory stick or HDD).
	eth0	Specifies nfsroot via external USB Ethernet interface.
Options	**Description**	
-h	Specifies to print this usage information.	
-b <bctfile>	Specifies the NvFlash Boot Configuration Table (BCT) file.	
-c <cfgfile>	Specifies the NvFlash configuration file.	
-d <dtbfile>	Optionally specifies a device tree file to use instead of the default.	
-e <emmc_file>	Specifies the eMMC size of the target device.	
-f <flashapp>	Specifies the path to flash application: nvflash or tegra-rcm.	
-i	Specifies to pass the user kernel command line to the kernel as-is.	
-k <partition id>	Specifies the kernel partition ID to be updated (minimum = 5).	
-n <nfs args>	Specifies the static NFS network assignments: `<Client IP>:<Server IP>:<Gateway IP>:<Netmask>`	
-o <odmdata>	Specifies the ODM data value.	
-p	Total eMMC HW boot partition size.	
-r	Specifies to skip building and reuse existing `system.img`.	

-s <ubootscript>	Specifies the boot script file for U-Boot.
-C <cmdline>	Specifies the kernel command line. **Warning**: Each option in this kernel command-line gets higher precedence over the same option from fastboot. In case of NFS booting, this script adds NFS booting related arguments if the -i option is omitted.
-F <flasher>	Specifies the flash server, such as `fastboot.bin`.
-I <initrd>	Specifies `initrd` file. Null `initrd` is the default.
-K <kernel>	Specifies the kernel image, such as zImage.
-L <bootloader>	Specifies the full path to the boot loader, such as `fastboot.bin` or `u-boot.bin`.
-P <end_of_PPT_plus_1>	Specifies the sum of the primary GPT start address, the size of PPT, plus 1.
-R <rootfs dir>	Specifies the sample rootfs directory.
-N <nfsroot>	Specifies the nfsroot, for example: `<my IP addr>:/my/exported/nfs/rootfs`
-S <size>	Specifies the rootfs size in bytes. This is valid only for internal rootdev. KiB, MiB, GiB style shorthand is allowed. For example, 1GiB signifies 1024 * 1024 * 1024 bytes.
-T <ITS file>	ITS file name. Valid only for u-boot.

JetPack

This section is intended to help you get familiar with installing Jetson TX1 Developer Pack (JetPack TX1), using the tools and running sample code.

Before you Begin

> You have a Jetson TX1 Tegra Developer Kit equipped with the NVIDIA Tegra TX1 processor.

> You have a host machine that is running Linux.

> Your developer system is cabled as follows:

- Serial cable plugged into the serial port J1A2 UART4 on the target connected to your Linux host directly or through a serial-to-USB converter. (To setup serial console on the Linux host.)
- USB Micro-B cable connecting Jetson TX1 (J1E1 USB0) to your Linux host for flashing.
- (Not included in the developer kit) To connect USB peripherals such as keyboard, mouse, and [optional] USB/Ethernet adapter (for network connection), a USB hub should be connected to the working USB port (J1C2 USB2) on the Jetson TX1 system.
- An HDMI cable plugged into "J1C1 HDMI1" on the target which is connected to an external HDMI display.
- An Ethernet cable plugged into the J1D1 on board Ethernet port.

> Download latest JetPack TX1

- The latest version of Jetson TX1 Development Pack (JetPack TX1) is available at NVIDIA Developer.
- All available JetPack TX1 downloads here.

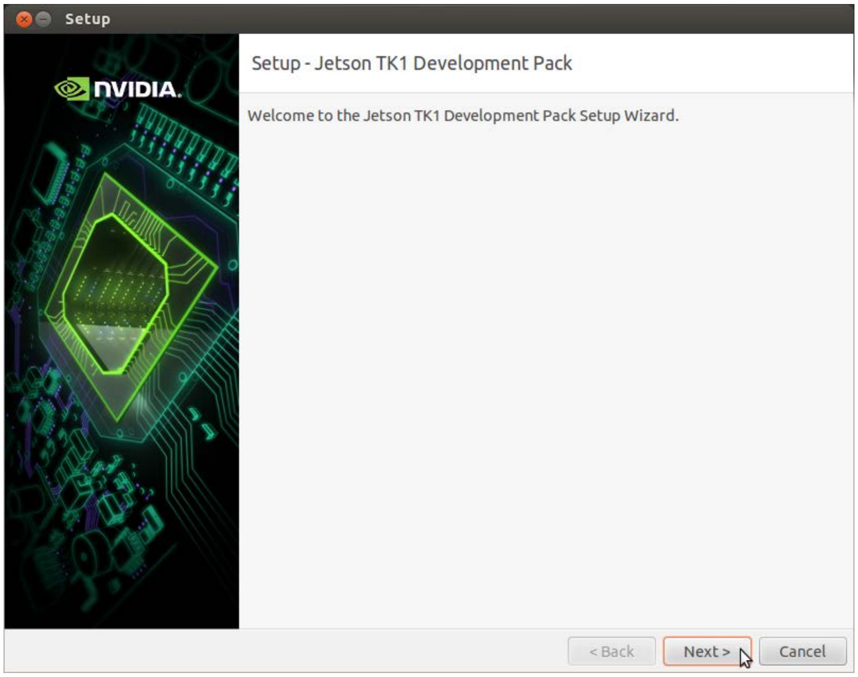
Installing JetPack TX1

Assumes you downloaded the latest JetPack TX1 version, `jetpack-${VERSION}.run`. `${VERSION}` refers to the version string for the installer you got.

1. Add execute permissions for the `jetpack-${VERSION}.run`

```
chmod +x jetpack-${VERSION}.run
```

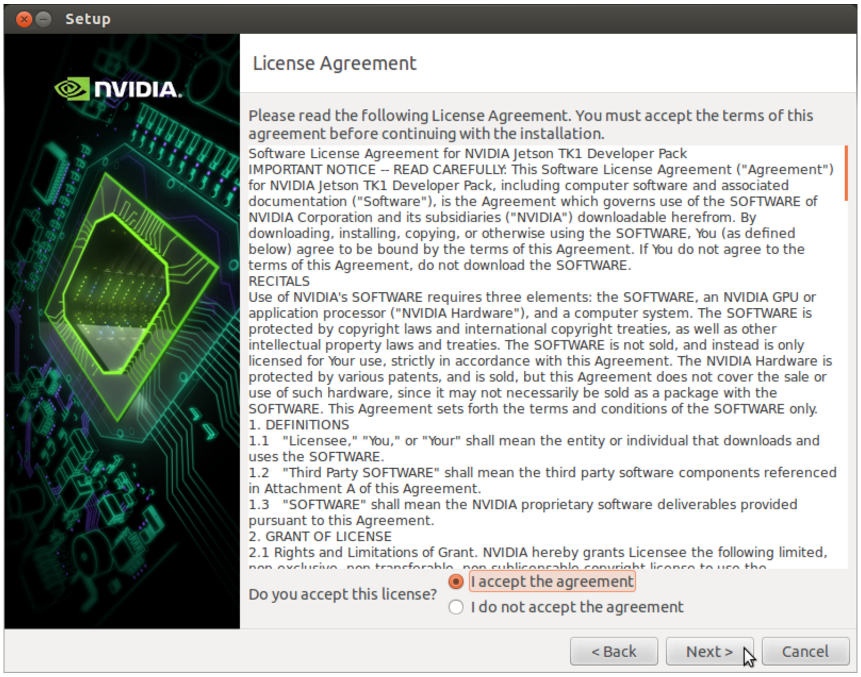
2. Run the `jetpack-${VERSION}.run` by double-clicking on it in the file browser.



Setup - Jetson TK1 Development Pack

Welcome to the Jetson TK1 Development Pack Setup Wizard.

3. Accept the license agreement to continue



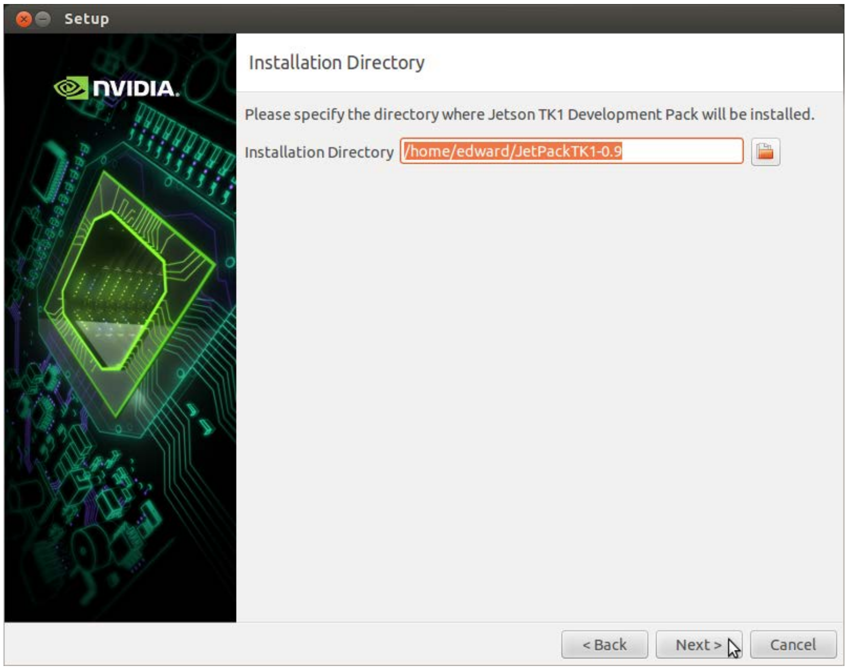
License Agreement

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

Software License Agreement for NVIDIA Jetson TK1 Developer Pack
IMPORTANT NOTICE -- READ CAREFULLY: This Software License Agreement ("Agreement") for NVIDIA Jetson TK1 Developer Pack, including computer software and associated documentation ("Software"), is the Agreement which governs use of the SOFTWARE of NVIDIA Corporation and its subsidiaries ("NVIDIA") downloadable herefrom. By downloading, installing, copying, or otherwise using the SOFTWARE, You (as defined below) agree to be bound by the terms of this Agreement. If You do not agree to the terms of this Agreement, do not download the SOFTWARE.
RECITALS
Use of NVIDIA's SOFTWARE requires three elements: the SOFTWARE, an NVIDIA GPU or application processor ("NVIDIA Hardware"), and a computer system. The SOFTWARE is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. The SOFTWARE is not sold, and instead is only licensed for Your use, strictly in accordance with this Agreement. The NVIDIA Hardware is protected by various patents, and is sold, but this Agreement does not cover the sale or use of such hardware, since it may not necessarily be sold as a package with the SOFTWARE. This Agreement sets forth the terms and conditions of the SOFTWARE only.
1. DEFINITIONS
1.1 "Licensee," "You," or "Your" shall mean the entity or individual that downloads and uses the SOFTWARE.
1.2 "Third Party SOFTWARE" shall mean the third party software components referenced in Attachment A of this Agreement.
1.3 "SOFTWARE" shall mean the NVIDIA proprietary software deliverables provided pursuant to this Agreement.
2. GRANT OF LICENSE
2.1 Rights and Limitations of Grant. NVIDIA hereby grants Licensee the following limited, non-exclusive, non-transferable, non-sublicensable copyright license to use the

Do you accept this license? ● I accept the agreement ○ I do not accept the agreement

4. Enter or navigate to desired installation directory.

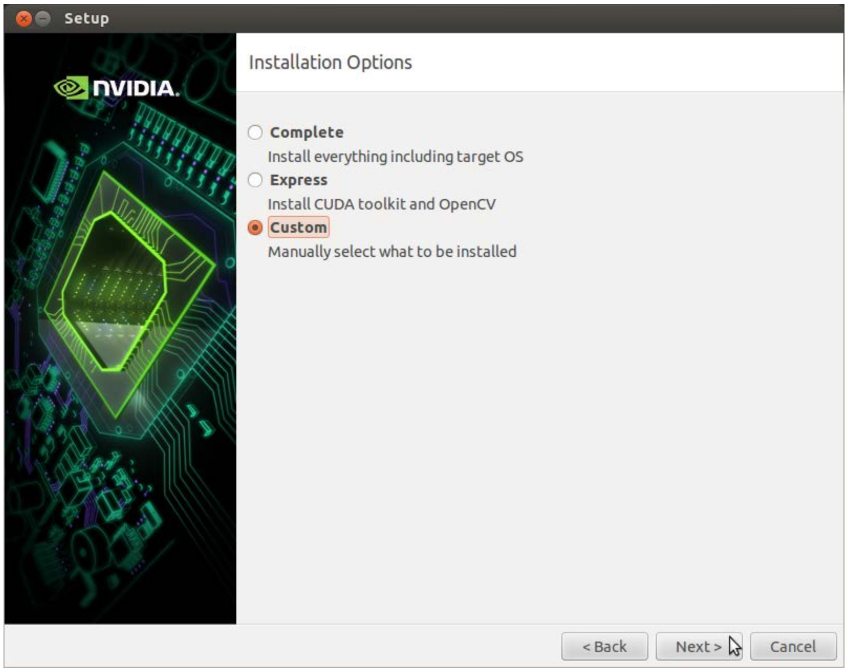


Installation Directory

Please specify the directory where Jetson TK1 Development Pack will be installed.

Installation Directory: /home/edward/JetPackTK1-0.9

5. Select installation type.



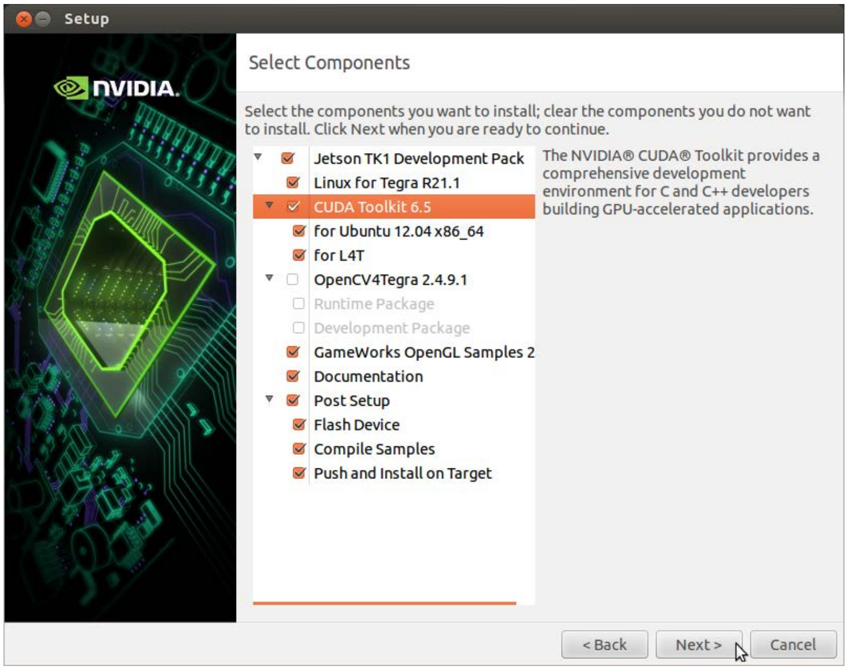
Installation Options

○ **Complete**
Install everything including target OS

○ **Express**
Install CUDA toolkit and OpenCV

● **Custom**
Manually select what to be installed

6. Select components to install.



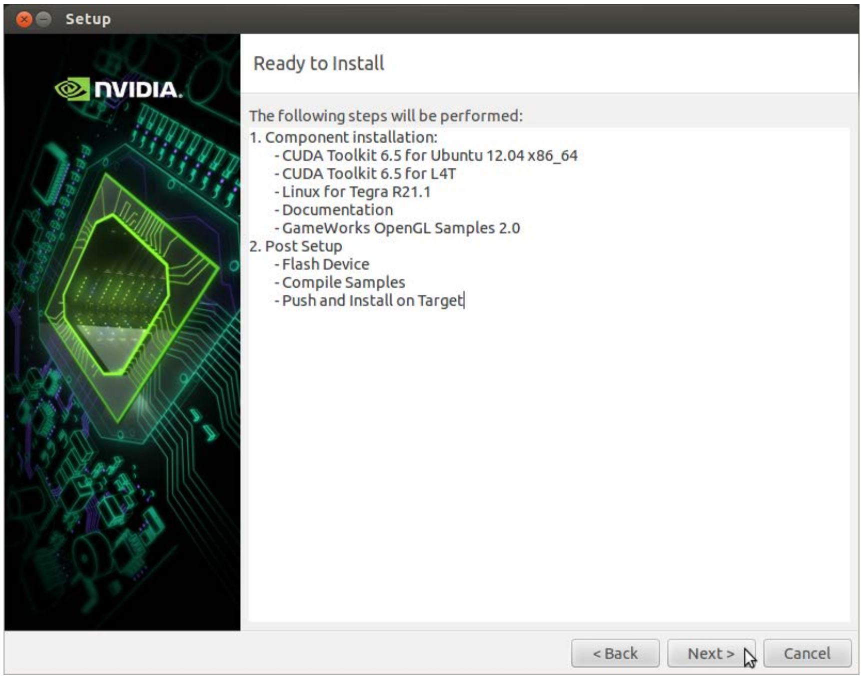
Select Components

Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

- ☑ Jetson TK1 Development Pack
  - ☑ Linux for Tegra R21.1
  - ☑ CUDA Toolkit 6.5
    - ☑ for Ubuntu 12.04 x86_64
    - ☑ for L4T
  - ☐ OpenCV4Tegra 2.4.9.1
    - ☐ Runtime Package
    - ☐ Development Package
  - ☑ GameWorks OpenGL Samples 2
  - ☑ Documentation
  - ☑ Post Setup
    - ☑ Flash Device
    - ☑ Compile Samples
    - ☑ Push and Install on Target

The NVIDIA® CUDA® Toolkit provides a comprehensive development environment for C and C++ developers building GPU-accelerated applications.

7. Review Setup and click the Next button proceed with setup or click Back button to return to the Select Components screen.

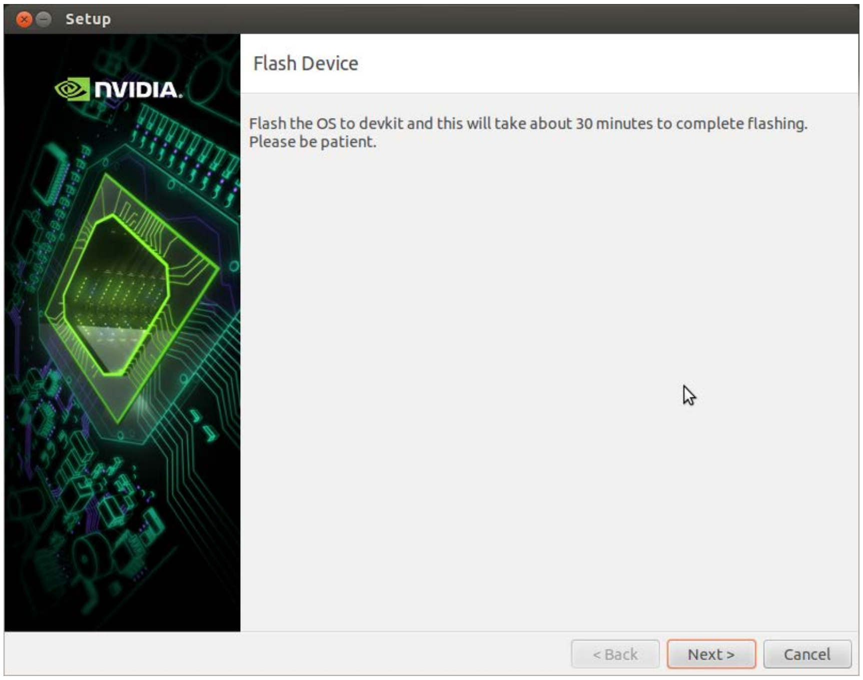


Ready to Install

The following steps will be performed:

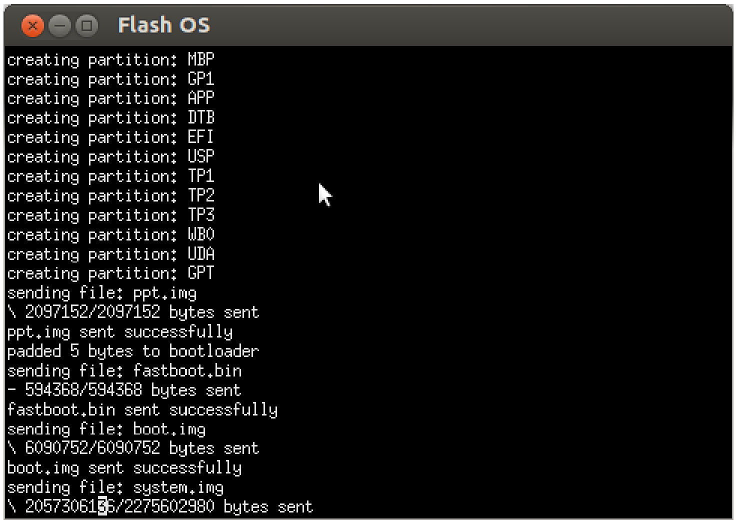
```
1. Component installation:
    - CUDA Toolkit 6.5 for Ubuntu 12.04 x86_64
    - CUDA Toolkit 6.5 for L4T
    - Linux for Tegra R21.1
    - Documentation
    - GameWorks OpenGL Samples 2.0
2. Post Setup
    - Flash Device
    - Compile Samples
    - Push and Install on Target
```

8. Click next and follow instructions on popup window to put device into recovery mode, then begin flashing device.



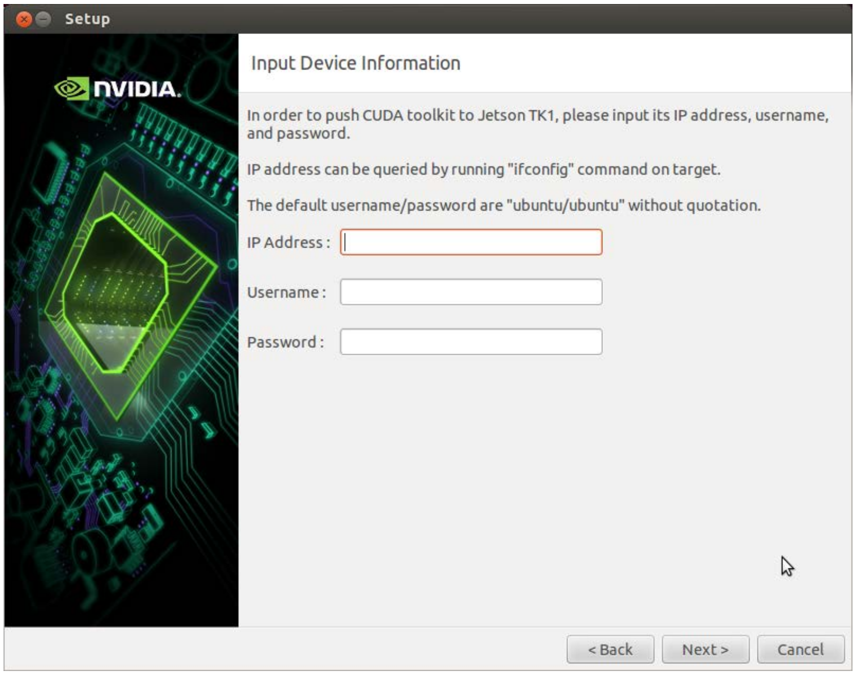
Flash the OS to devkit and this will take about 30 minutes to complete flashing.
Please be patient.

Sample output when flashing OS on Jetson.



```
creating partition: MBP
creating partition: GP1
creating partition: APP
creating partition: DTB
creating partition: EFI
creating partition: USP
creating partition: TP1
creating partition: TP2
creating partition: TP3
creating partition: WB0
creating partition: UDA
creating partition: GPT
sending file: ppt.img
\ 2097152/2097152 bytes sent
ppt.img sent successfully
padded 5 bytes to bootloader
sending file: fastboot.bin
- 594368/594368 bytes sent
fastboot.bin sent successfully
sending file: boot.img
\ 6090752/6090752 bytes sent
boot.img sent successfully
sending file: system.img
\ 2057306136/2275602980 bytes sent
```

9. After successfully flashing device, enter your device information in the dialog that comes up.



Input Device Information

In order to push CUDA toolkit to Jetson TK1, please input its IP address, username, and password.

IP address can be queried by running "ifconfig" command on target.

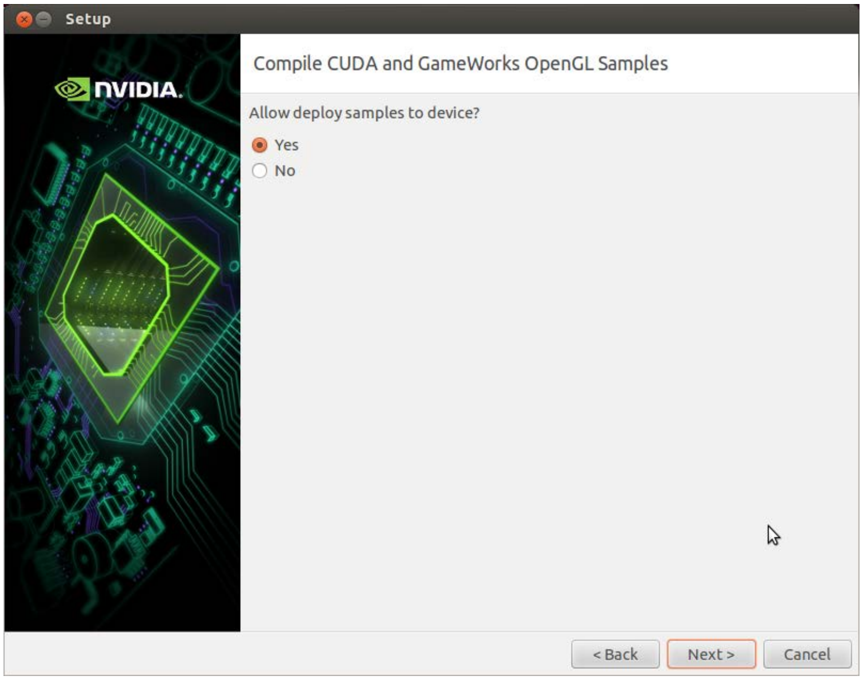
The default username/password are "ubuntu/ubuntu" without quotation.

IP Address :

Username :

Password :

10. After installing CUDA on the device, the installer will compile CUDA and GameWorks OpenGL samples. You can choose whether to deploy the compiled binaries to device.

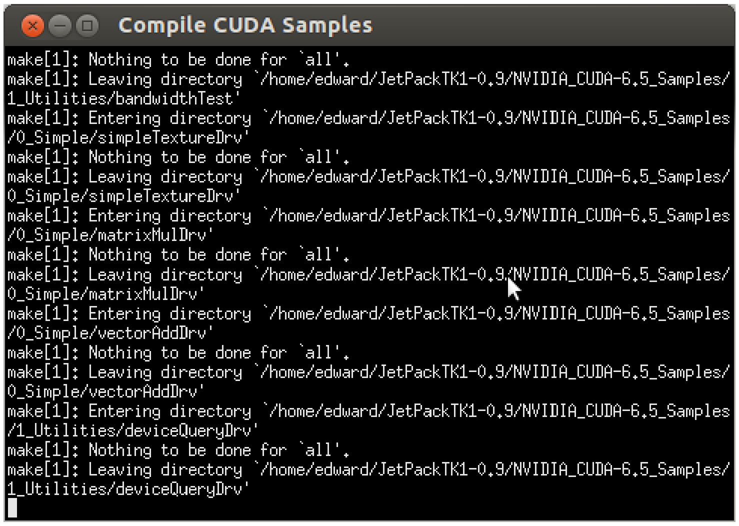


Compile CUDA and GameWorks OpenGL Samples

Allow deploy samples to device?

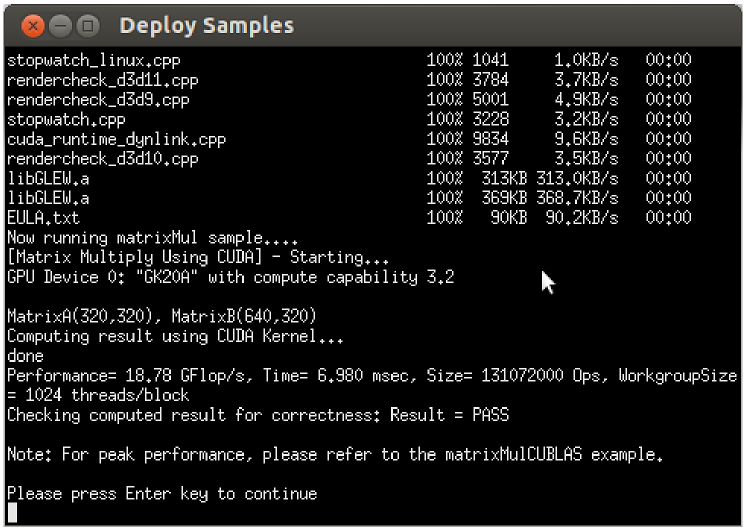
● Yes
○ No

Sample output when compiling CUDA samples.



```
make[1]: Nothing to be done for `all'.
make[1]: Leaving directory `/home/edward/JetPackTK1-0.9/NVIDIA_CUDA-6.5_Samples/
1_Utilities/bandwidthTest'
make[1]: Entering directory `/home/edward/JetPackTK1-0.9/NVIDIA_CUDA-6.5_Samples
/0_Simple/simpleTextureDrv'
make[1]: Nothing to be done for `all'.
make[1]: Leaving directory `/home/edward/JetPackTK1-0.9/NVIDIA_CUDA-6.5_Samples/
0_Simple/simpleTextureDrv'
make[1]: Entering directory `/home/edward/JetPackTK1-0.9/NVIDIA_CUDA-6.5_Samples
/0_Simple/matrixMulDrv'
make[1]: Nothing to be done for `all'.
make[1]: Leaving directory `/home/edward/JetPackTK1-0.9/NVIDIA_CUDA-6.5_Samples/
0_Simple/matrixMulDrv'
make[1]: Entering directory `/home/edward/JetPackTK1-0.9/NVIDIA_CUDA-6.5_Samples
/0_Simple/vectorAddDrv'
make[1]: Nothing to be done for `all'.
make[1]: Leaving directory `/home/edward/JetPackTK1-0.9/NVIDIA_CUDA-6.5_Samples/
0_Simple/vectorAddDrv'
make[1]: Entering directory `/home/edward/JetPackTK1-0.9/NVIDIA_CUDA-6.5_Samples
/1_Utilities/deviceQueryDrv'
make[1]: Nothing to be done for `all'.
make[1]: Leaving directory `/home/edward/JetPackTK1-0.9/NVIDIA_CUDA-6.5_Samples/
1_Utilities/deviceQueryDrv'
```

11. Copy CUDA samples to device. Run `matrixMul` sample remotely on device.



```
stopwatch_linux.cpp                              100% 1041     1.0KB/s   00:00
rendercheck_d3d11.cpp                            100% 3784     3.7KB/s   00:00
rendercheck_d3d9.cpp                             100% 5001     4.9KB/s   00:00
stopwatch.cpp                                    100% 3228     3.2KB/s   00:00
cuda_runtime_dynlink.cpp                         100% 9834     9.6KB/s   00:00
rendercheck_d3d10.cpp                            100% 3577     3.5KB/s   00:00
libGLEW.a                                        100%  313KB 313.0KB/s   00:00
libGLEW.a                                        100%  369KB 368.7KB/s   00:00
EULA.txt                                         100%   90KB  90.2KB/s   00:00
Now running matrixMul sample....
[Matrix Multiply Using CUDA] - Starting...
GPU Device 0: "GK20A" with compute capability 3.2

MatrixA(320,320), MatrixB(640,320)
Computing result using CUDA Kernel...
done
Performance= 18.78 GFlop/s, Time= 6.980 msec, Size= 131072000 Ops, WorkgroupSize
= 1024 threads/block
Checking computed result for correctness: Result = PASS

Note: For peak performance, please refer to the matrixMulCUBLAS example.

Please press Enter key to continue
```

Compiling

JetPack TX1 automatically compiles all samples if “Compile Samples” was checked during components selection. If you selected CUDA components, CUDA samples will be found in the following directory:

```
 <JetPack_Install_Dir>/NVIDIA_CUDA-<version>_Samples directory.
```

You could recompile the samples by running:

```
OPENMPDIR=/usr/arm-linux-gnueabihf/lib EXTRA_LDFLAGS=--unresolved-
symbols=ignore-in-shared-libs ARMv7=1 GCC=arm-linux-gnueabihf-g++ make
```

If you selected GameWorks OpenGL samples, GameWorks OpenGL samples will be available in the following directory:

```
< JetPack_Install_Dir >/GameWorksOpenGLSamples
```

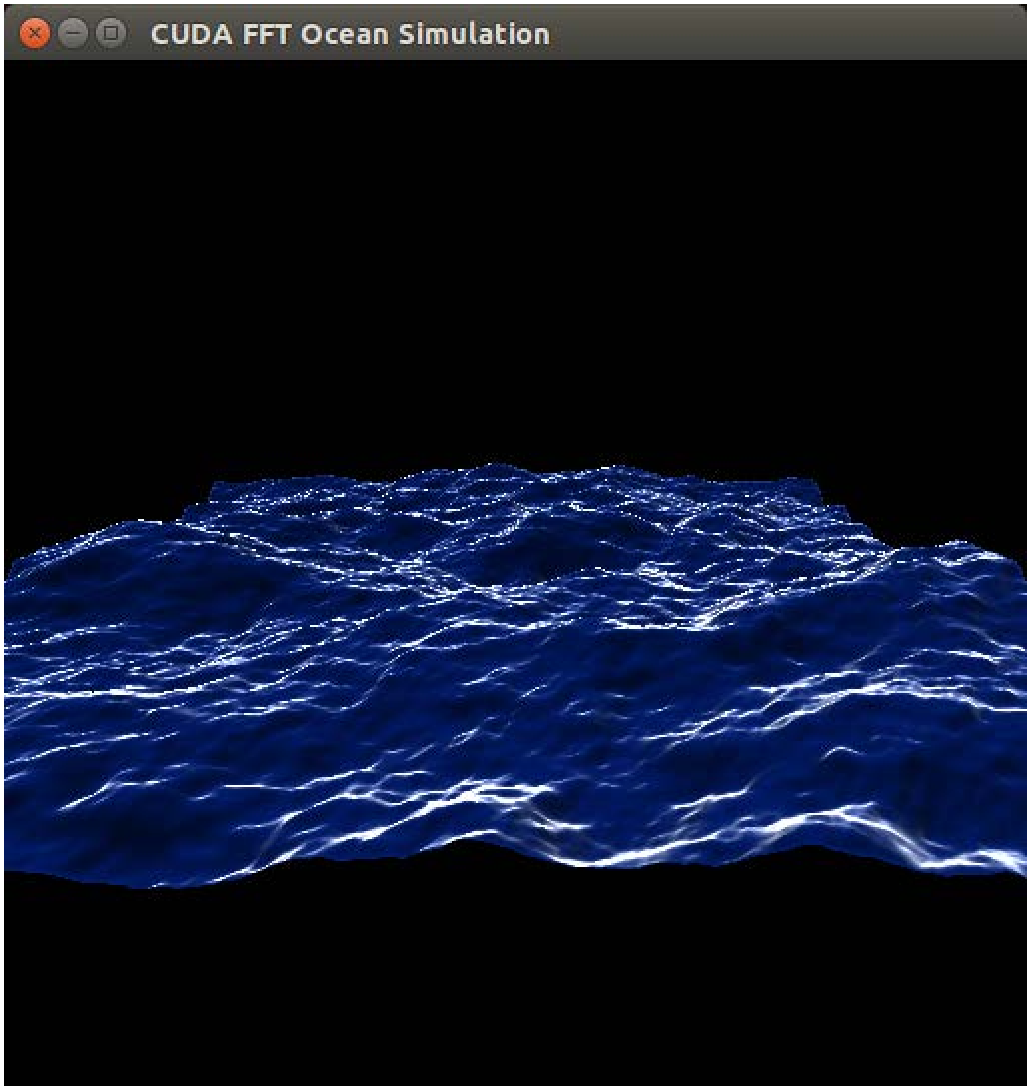
You could cross compile them by running the “`make`” command under the “`samples/build/linux-arm32`” subfolder.

Run JetPack TX1 Samples

The CUDA samples directory is copied to home directory on the device by JetPack TX1. The built binaries can be found in the following directory:

```
/home/ubuntu/NVIDIA_CUDA-6.5_Samples/bin/armv7l/linux/release/gnueabihf/
```

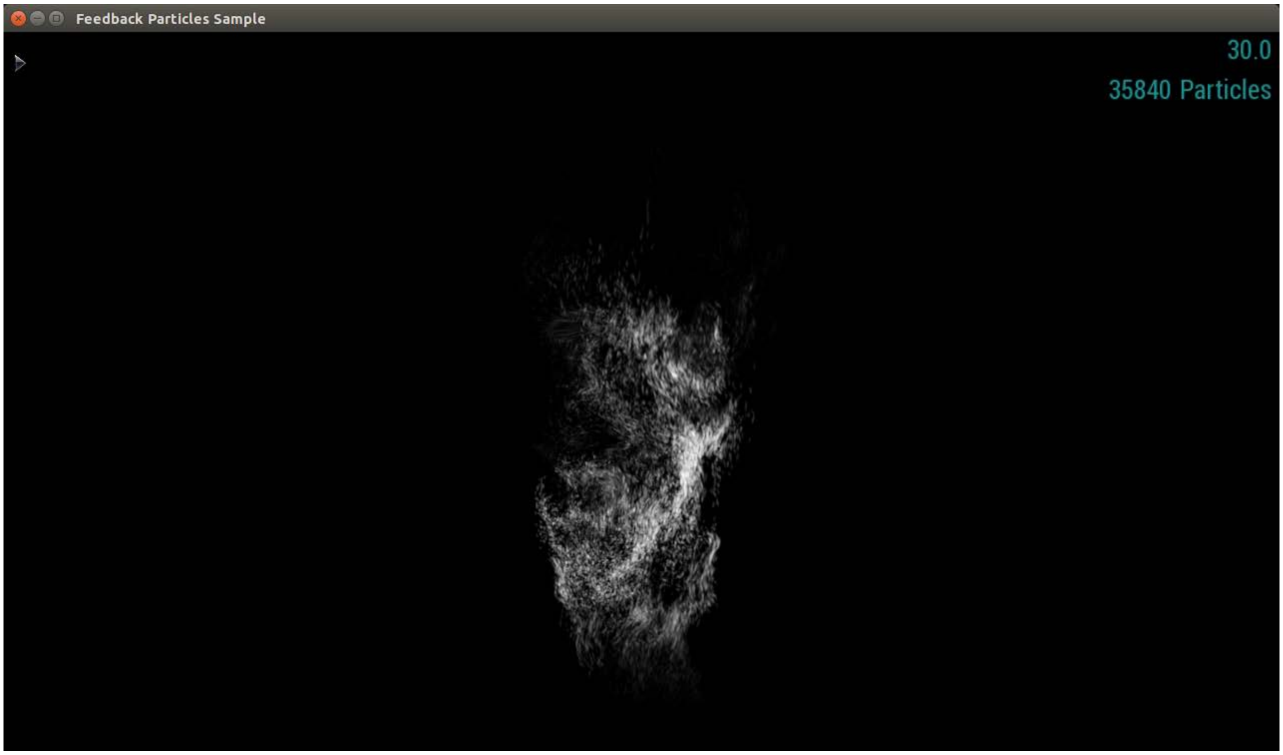
Run them by calling them in terminal or double clicking on them in the file browser. For example, when you run the `oceanFFT` sample, the following screen will be displayed.



GameWorks OpenGL samples will be copied to home directory on device. The built binaries can be found in the following directory:

```
/home/ubuntu/GameWorksOpenGLSamples/samples/bin/linux-arm32
```

Run them by calling them in terminal or double clicking on them in the file browser. For example, when you run `FeedbackParticlesApp`, you will see smoke particles rising.



Compliance

The NVIDIA® Jetson TX1 Developer Kit is compliant with the regulations listed in this section. Compliance marks, including the FCC and IC ID numbers, can be found http://developer.nvidia.com/embedded-computing

United States

Federal Communications Commission (FCC)

FCC ID: VOB-P2180

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including any interference that may cause undesired operation of the device.

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation.

If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

FCC Warning: The FCC requires that you be notified that any changes or modifications to this device not expressly approved by the manufacturer could void the user’s authority to operate the equipment.

RF Radiation Exposure Statement

This equipment complies with FCC RF radiation exposure limits set forth for an uncontrolled environment. This equipment should be installed and operated with a minimum distance of 20 centimeters between the radiator and your body.

Only those antennas with same type and lesser/equal gain filed under this FCC ID number can be used with this device.

Underwriters Laboratories (UL)

I.T.E E204896

Canada

Industry Canada (IC)

IC: 7361A-P2180

CAN ICES-3(B)/NMB-3(B)

This device complies with *Industry Canada's licence-exempt RSSs* of the Industry Canada Rules. . Operation is subject to the following two conditions: (1) this device may not cause interference, and (2) this device must accept any interference, including interference that may cause undesired operation of the device.

5150–5250 MHz is only for indoor use to reduce the potential for harmful interference to co-channel mobile satellite systems;

Ce dispositif est conforme à la norme RSS-247 d'Industrie Canada applicable aux appareils radio exempts de licence. Son fonctionnement est sujet aux deux conditions suivantes: (1) le dispositif ne doit pas produire de brouillage préjudiciable, et (2) ce dispositif doit accepter tout brouillage reçu, y compris un brouillage susceptible de provoquer un fonctionnement indésirable.

RF Radiation Exposure Statement:

Jetson Dev Kit has been tested and complies with IC RSS 102 RF radiation exposure limits set forth for an uncontrolled environment when used with the NVIDIA accessories supplied or designated for this product. To satisfy IC exposure requirements, a separation distance of at least 20 cm must be maintained between the antenna of this device and persons during device operation.. The use of any other accessories may not ensure compliance with IC RSS 102RF exposure guidelines.

Déclaration d'exposition aux radiations:

La Jetson Dev Kit a ete testee conformemment aux normes d'exposition d'emission RF de la IC RSS 102 pour un environement non controle lors d'utilisation avec les accessoires fournis or recommendes par NVIDIA. Pour satisfaire aux exigences d'exposition IC, une distance de séparation d'au moins 20 cm doit être maintenue entre l'antenne de cet appareil et des personnes pendant le fonctionnement de l'appareil. L'utilisation d'accessoires autres que ceux recommendes par NVIDIA ne guarantis pas la compatibilite avec les normes d'emission RF de la IC RSS 102.

European Union

European Conformity; Conformité Européenne (CE)

This device bears the CE mark and class-2 identifier in accordance with Directive 1999/5/EC

This device complies with the following Directives:

R&TTE Directive for radio equipment

Low Voltage Directive for electrical safety

RoHS Directive for hazardous substances

A copy of the Declaration of Conformity to the essential requirements may be obtained directly from NVIDIA GmbH (Floessergasse 2, 81369 Munich, Germany).

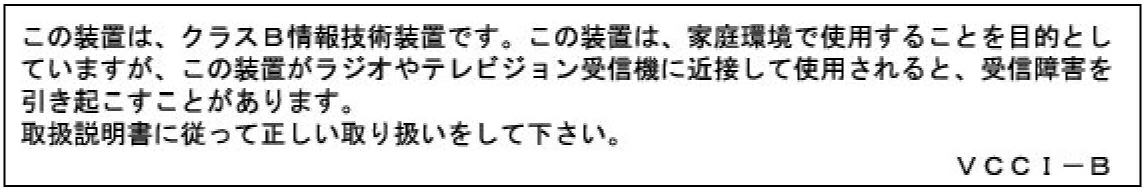
Australia and New Zealand

Australian Communications and Media Authority (RCM)

This product meets the applicable EMC requirements for Class B, I.T.E equipment and applicable radio equipment requirements

Japan

Voluntary Control Council for Interference (VCCI)



この装置は、クラスＢ情報技術装置です。この装置は、家庭環境で使用することを目的としていますが、この装置がラジオやテレビジョン受信機に近接して使用されると、受信障害を引き起こすことがあります。
取扱説明書に従って正しい取り扱いをして下さい。

ＶＣＣＩ－Ｂ

Radio/ Telecommunications Certification

R XXXAAAYYYY

W52/W53 are indoor use only

South Korea

Radio Research Agency (RRA)

Korean Agency for Technology and Standards (KATS)

MSIP-RMM-NVA-P2180

MSIP-CMM-NVA-P2597

B 급 기기 (가정용 방송 통신기자재)	이 기기는 가정용(B 급) 전자파적합기기로서 주로 가정에서 사용하는 것을 목적으로 하며, 모든 지역에서 사용할 수 있습니다.

Taiwan

National Communications Commission

國家通訊傳播委員會

CCXXxxYYyyyZzW

注意！

依據 低功率電波輻射性電機管理辦法

第十二條

經型式認證合格之低功率射頻電機，非經許可，公司、商號或使用者均不得擅自變更頻率、加大功率或變更原設計之特性及功能。

第十四條

低功率射頻電機之使用不得影響飛航安全及干擾合法通信；經發現有干擾現象時，應立即停用，並改善至無干擾時方得繼續使用。

前項合法通信，指依電信法規定作業之無線電通信。

低功率射頻電機須忍受合法通信或工業、科學及醫療用電波輻射性電機設備之干擾。

本無線資訊傳輸設備具主控運作模式故不得使用 5.600-5.650 GHz 頻帶

CHINA

CMIT ID:XXXXXX

本设备包含型号核准代码为 CMIIT ID: XXXXYYZZZ 的无线电发射模块

ISRAEL