

**Jetson Device Flashing and Configuration Guide**

**Flashing Support Overview**

* [**flash.sh**](http://flash.sh): Used to flash a Jetson device with the bootloader, kernel, and optionally the root file system to internal or external storage.
* **l4t\_initrd\_flash.sh**: Leverages the recovery initrd for faster flashing to both internal and external media.

**Prerequisites**

Ensure the following directories exist:

* bootloader: Bootloader and flashing tools (TegraFlash, CFG, BCT)
* kernel: Kernel image, DTB files, kernel modules
* rootfs: Root file system (initially empty, to be populated with the sample file system)
* nv\_tegra: User space binaries and sample applications

**Connection:** Connect your host computer to the Jetson device’s recovery port via USB before running flash commands.

**Basic Flashing Script Usage**

* **Display usage info:**

$ sudo ./flash.sh –h

* **Basic command:**

$ sudo ./flash.sh [options] <board> <rootdev>

* + <board>: Target device configuration (see Jetson Modules and Configurations)
  + <rootdev>: Device to be flashed (e.g., mmcblk0p1 for eMMC/SD card)

**Flashing Procedures**

**1. Flashing the Target Device**

* Put device in Force Recovery Mode:
  + Power on, hold RECOVERY button, press RESET button.
* Run flash script:

$ sudo ./flash.sh <board> <rootdev>

* + Example:

$ sudo ./flash.sh <board> mmcblk0p1

**2. Flash Using NVIDIA’s Convenient Script**

* Automatically detects carrier board:

$ sudo ./nvsdkmanager\_flash.sh [--storage <storage>]

* + --storage options: nvme0n1p1 (NVMe SSD), sda1 (USB drive)

**3. Flash with Rootfs by UUID**

* **Internal storage:**

$ sudo ./flash.sh <board> internal

* **External storage:**

$ sudo ./flash.sh <board> external

**4. Flash by Partition Device Name**

* **USB device:**

$ sudo ./flash.sh <board> sda<x>

* **NVMe device:**

$ sudo ./flash.sh <board> nvme0n1p<x>

**Cloning, Backup, and Restore**

**Clone and Flash**

1. Clone system image:

$ sudo ./flash.sh -r -k APP -G <clone> <board> mmcblk0p1

1. Copy and flash image:

$ sudo cp <clone>.img bootloader/system.img  
$ sudo ./flash.sh -r -k APP <board> mmcblk0p1

* + For a new device, flash all partitions:

$ sudo ./flash.sh -r <board> mmcblk0p1

**Backup and Restore**

* Tools located at /Linux\_for\_Tegra/tools/backup-restore/
* Instructions: README\_backup\_restore.txt

**Flashing to Different Storage Media**

**USB Drive**

1. **Manual Setup:**
   * Identify device: $ sudo lsblk -p -d | grep sd
   * Create GPT: $ sudo parted /dev/<sdx> mklabel gpt
   * Add partition: $ sudo parted /dev/<sdx> mkpart APP 0GB <size>
   * Format and mount:

$ sudo mkfs.ext4 /dev/<sdx>1  
$ sudo mount /dev/<sdx>1 /mnt

1. **Generate rootfs without flashing:**

$ cd Linux\_for\_Tegra/  
$ sudo BOOTDEV=sda1 ./flash.sh --no-flash <board> <sdxn>  
$ sudo mkdir tmp\_system  
$ sudo mount bootloader/system.img.raw ./tmp\_system  
$ sudo rsync -axHAWX --numeric-ids --info=progress2 --exclude=/proc ./tmp\_system/ /mnt

* + Unmount and plug into target device.

**NVMe Drive**

1. **Manual Setup:**
   * Identify device: $ lsblk -d -p | grep nvme | cut -d\ -f 1
   * Create GPT: $ sudo parted /dev/<nvmeXn1> mklabel gpt
   * Add partition: $ sudo parted /dev/<nvmeXn1> mkpart APP 0GB <size>
   * Format and mount:

$ sudo mkfs.ext4 /dev/<nvmeXn1>p1  
$ sudo mount /dev/<nvmeXn1>p1 /mnt

1. **Generate rootfs without flashing:**

$ cd Linux\_for\_Tegra/  
$ sudo BOOTDEV=nvme0n1p1 ./flash.sh --no-flash <board> nvme0n1p1  
$ sudo mkdir tmp\_system  
$ sudo mount bootloader/system.img.raw ./tmp\_system  
$ sudo rsync -axHAWX --numeric-ids --info=progress2 --exclude=/proc ./tmp\_system/ /mnt

* + Unmount and plug into target device.
  + Set boot order in U-Boot if required.

**SD Card (Jetson Xavier NX Only)**

1. **Create image:**

$ ./jetson-disk-image-creator.sh -o <blob\_name> -b <board>

1. **Flash image:**
   * With Etcher: Use GUI to select image and SD card.
   * With dd:

$ sudo dd if=<sd\_blob\_name> of=/dev/mmcblk<n> bs=1M oflag=direct

1. **Resize root partition:** Handled by oem-config on first boot.

**Flashing to an External Storage Device (Initrd)**

* Tools and workflows: Linux\_for\_Tegra/tools/kernel\_flash/
* See README\_initrd\_flash.txt for details.
* Devices: /dev/sd\* (USB), /dev/nvme\*n\* (NVMe)

**Flashing a Specific Partition**

* Use -k option:

$ sudo ./flash.sh -k <partition\_name> [--image <image\_name>] <board> <rootdev>

**Flashing for NFS as Root**

* **Preparation:** Set up NFS root file system and configure NFS on Linux host.
* **Flash command:**

$ sudo ./flash.sh -N <ip\_addr>:<root\_path> <board> eth0

**Flashing with Initrd**

1. **Requirements:** High-quality USB cable, NetworkManager, dependencies (libxml2-utils, simg2img, network-manager, abootimg, sshpass, device-tree-compiler)
2. **Disable automount:** $ systemctl stop udisks2.service
3. **Flash command:**

$ cd Linux\_for\_Tegra  
$ sudo ./tools/kernel\_flash/l4t\_initrd\_flash.sh <board-name> <rootdev>

**Mass Flashing**

* Tool and instructions: Linux\_for\_Tegra/README\_Massflash.txt

**Enlarging Internal Memory Partition**

* Use -S <size> option:

$ sudo ./flash.sh -S <size> <board> <rootdev>

**Verifying Driver Update Success**

* On target device:

$ sha1sum –c /etc/nv\_tegra\_release

* + Success: Displays "OK" after file name.

**Reconfiguring with oem-config**

* Runs automatically on first boot after flashing with SDK Manager.
* Can be re-enabled manually for custom configurations.

**To Re-enable oem-config**

1. **Install packages:**

$ sudo apt-get install --no-install-recommends ubiquity oem-config oem-config-gtk

1. **Remove NVIDIA oem-config:**

$ sudo dpkg --purge nvidia-l4t-oem-config

1. **Apply binaries and set default target:**

$ cd Linux\_for\_Tegra  
$ sudo ./apply\_binaries.sh -r <root>  
$ cd $root/etc/systemd/system  
$ sudo rm default.target  
$ sudo ln -s /lib/systemd/system/nv-oem-config.target default.target

1. **Make sparse image and flash.**

**Communication Through Debugging Port**

* Default: Host’s tty device ↔ Jetson’s flashing port (USB)
* To use 40-pin UART: Edit <top>/etc/nv-oem-config.conf.t194, set nv-oem-config-uart-port=ttyTHS0

**Headless Mode Flow in oem-config**

* Use a terminal program (e.g., putty, screen) on host.
* Follow on-screen prompts to set locale, username, password, network, and other options.
* Optionally create a swap file and set APP partition size.

**Skipping oem-config**

* Use l4t\_create\_default\_user.sh before flashing:

$ l4t\_create\_default\_user.sh [-u <user>] [-p <pswd>] [-n <host>] [-a] [--accept-license] [-h]

* + Example:

$ l4t\_create\_default\_user.sh -u nvidia -p NDZjMWM4

**Modifying Jetson RAM Disk**

1. **Unpack initrd:**

$ sudo su  
$ cp /boot/initrd /tmp  
$ mkdir /tmp/temp  
$ cd /tmp/temp  
$ gunzip -c /tmp/initrd | cpio -i

1. **Modify files as needed.**
2. **Repack initrd:**

$ cd /tmp/temp  
$ find . | cpio -H newc -o | gzip -9 -n > ../initrd  
$ cp /tmp/initrd /boot/initrd

*This document provides a comprehensive reference for flashing, cloning, backing up, and configuring NVIDIA Jetson devices using official tools and procedures. For detailed workflows and advanced options, refer to the official NVIDIA documentation and the README files included in the Linux\_for\_Tegra package.*

⁂