Building Jetson kernel

Wednesday, January 23, 2019   4:57 PM

Setting up with Auvidea J90 carrier board with no modifications (with Jetpack): https://devtalk.nvidia.com/default/topic/1045728/jetson-tx2/setting-the-tx2-up-with-auvide carrier-board/post/5306375/#5306375

Process to compile kernel for the Jetson TX2 with Auvidea carrier board:
You can use this as a general guide. Will refer to this as **Link1**

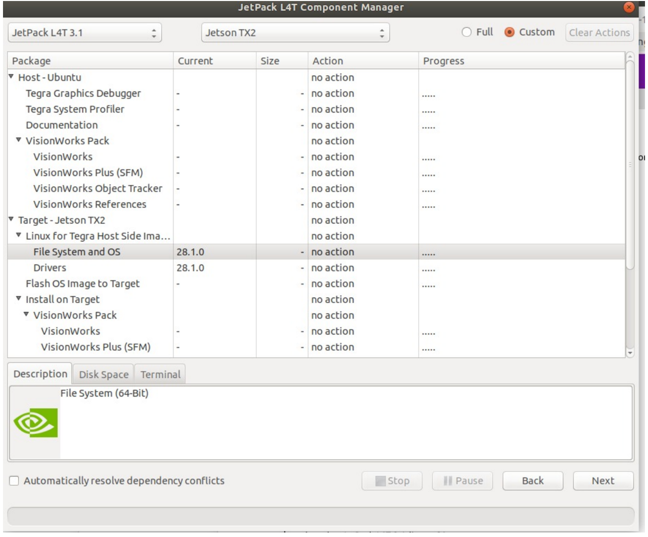
The default configuration file for the TX2 devboard is

Linux_for_Tegra/sources/hardware/nvidia/platform/t18x/quill/kernel-dts/tegra186-quill-p3310-1000-c03-00-base.dts

There is not much in this file because it takes most of it's configuration from:
Linux_for_Tegra/sources/hardware/nvidia/platform/t18x/quill/kernel-dts/tegra186-quill-p3310-1000-a00-00-base.dts

1. If running on Host Ubuntu 18.04, do the following. Skip if Ubuntu 16.04 or 14.04.
   a. Download Jetpack 'JetPack-L4T-3.1-linux-x64.run'
   b. chmod +x JetPack-L4T-3.1-linux-x64.run
   c. ./JetPack-L4T-3.1-linux-x64.run --noexec
   d. cd _installer
   e. Use editor(vi,code,nano,etc..) to open start_up.sh
   f. Change this line: if ["$os_version" == "14.04"] || ["$os_version" == "16.04"] ; then to
      if ["$os_version" == "18.04"] || ["$os_version" == "16.04"] ; then (line 46), if running Ubuntu 18.04. Else, leave it unchanged.
   g. ./start_up.sh

2. Choose the option Custom, and click on Clear Actions



JetPack L4T Component Manager

JetPack L4T 3.1 ⇕   Jetson TX2 ⇕   ○ Full ● Custom Clear Actions

Package	Current	Size	Action	Progress
▾ Host - Ubuntu			no action	
Tegra Graphics Debugger	-	-	no action	.....
Tegra System Profiler	-	-	no action	.....
Documentation	-	-	no action	.....
▾ VisionWorks Pack			no action	
VisionWorks	-	-	no action	.....
VisionWorks Plus (SFM)	-	-	no action	.....
VisionWorks Object Tracker	-	-	no action	.....
VisionWorks References	-	-	no action	.....
▾ Target - Jetson TX2			no action	
▾ Linux for Tegra Host Side Ima...			no action	
File System and OS	28.1.0	-	no action	.....
Drivers	28.1.0	-	no action	.....
Flash OS Image to Target	-	-	no action	.....
▾ Install on Target			no action	
▾ VisionWorks Pack			no action	
VisionWorks	-	-	no action	.....
VisionWorks Plus (SFM)	-	-	no action	.....

Description | Disk Space | Terminal

File System (64-Bit)

□ Automatically resolve dependency conflicts   Stop Pause Back Next

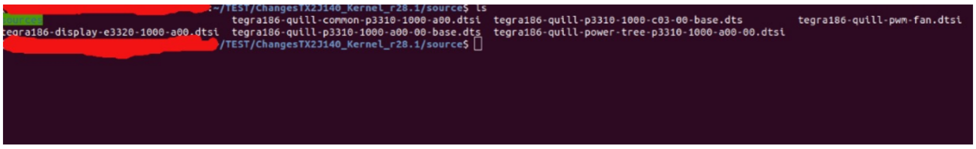
3. Select 'File System and OS' and 'Drivers' under Action, and choose install -> Next

4. After download, do the following to download kernel sources. The –k and –u parameters are the tag version which you can get from the Release notes for the specific version.
   a. cd 64_TX2/Linux_for_Tegra_tx2/
   b. sudo ./source_sync.sh -k tegra-l4t-r28.1 -u tegra-l4t-r28.1

5. Download Auvidea J90 firmware matching the Jetpack version, from their website: https://auvidea.eu/firmware/

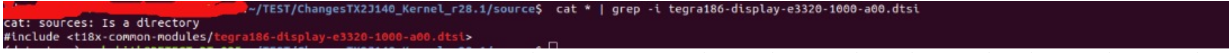
6. Extract the firmware zip file; in the case of L4T 28.1, it is called ChangesTX2J140_Kernel_r28.1_v1.5.tar.gz

7. Open the readme.txt, copy whatever files it mentions, **WITHOUT FLASHING.** You don't actually have to copy the compiled dtb files, because we'll be overwriting them make changes to the kernel.
   a. There is usually a folder called Sources as well. Use 'cat * | grep -I <filename>' to see which folder the file belongs to, and copy and paste it in that location. Exam steps below:
      i. cd ChangesTX2J140_Kernel_r28.1/source
      ii. ls



```
[redacted]:~/TEST/ChangesTX2J140_Kernel_r28.1/source$ ls
sources                                 tegra186-quill-common-p3310-1000-a00.dtsi  tegra186-quill-p3310-1000-c03-00-base.dts         tegra186-quill-pwm-fan.dtsi
tegra186-display-e3320-1000-a00.dtsi  tegra186-quill-p3310-1000-a00-00-base.dts  tegra186-quill-power-tree-p3310-1000-a00-00.dtsi
[redacted]/TEST/ChangesTX2J140_Kernel_r28.1/source$ 
```

      iii. cat * | grep -i tegra186-display-e3320-1000-a00.dtsi

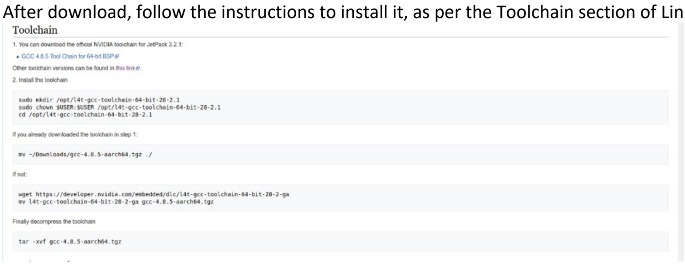


```
[redacted]:~/TEST/ChangesTX2J140_Kernel_r28.1/source$  cat * | grep -i tegra186-display-e3320-1000-a00.dtsi
cat: sources: Is a directory
#include <t18x-common-modules/tegra186-display-e3320-1000-a00.dtsi>
```

      iv. Which tells us that this file should be present in the folder 64_TX2/Linux_for_Tegra_tx2/sources/hardware/nvidia/platform/t18x/common/kernel-dts/t18x common-modules/
      v. Repeat for all the files.
      vi. Common possible locations are: common-modules, common-platforms, common-plugin-manager, and 64_TX2/Linux_for_Tegra_tx2/sources/hardware/nvidia/platform/t18x/quill/kernel-dts.
      vii. If the cat * command returns nothing or just the #include "filename", it usually implies file is in the quill/kernel-dts location.

viii. An alternative to this is to find the original files in the sources directory.

8. Download the Cross compile toolchain from here, or other sources, and set it up. The one that worked for me was: gcc-linaro-6.4.1-2017.08-x86_64_aarch64-linux-gnu you can find here
   a. After download, follow the instructions to install it, as per the Toolchain section of Link1.



Toolchain

1. You can download the official NVIDIA toolchain for JetPack 3.2.1:

- GCC 4.8.5 Tool Chain for 64-bit BSP

Other toolchain versions can be found in this link.

2. Install the toolchain

```
sudo mkdir /opt/l4t-gcc-toolchain-64-bit-28-2.1
sudo chown $USER:$USER /opt/l4t-gcc-toolchain-64-bit-28-2.1
cd /opt/l4t-gcc-toolchain-64-bit-28-2.1
```

If you already downloaded the toolchain in step 1:

```
mv ~/Downloads/gcc-4.8.5-aarch64.tgz ./
```

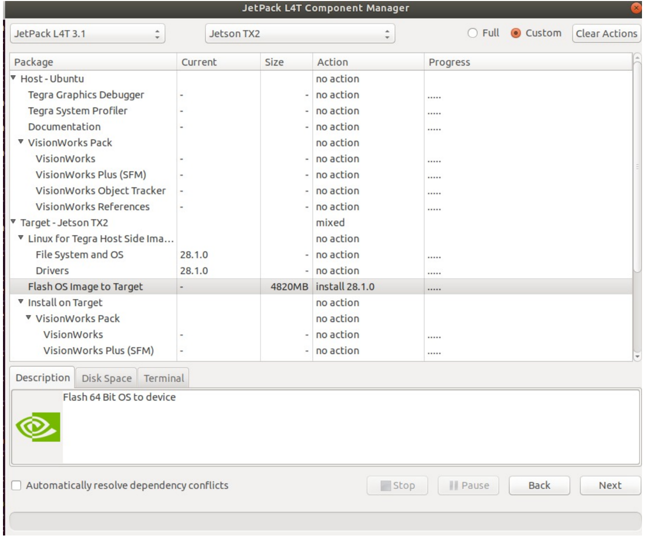
If not:

```
wget https://developer.nvidia.com/embedded/dlc/l4t-gcc-toolchain-64-bit-28-2-ga
mv l4t-gcc-toolchain-64-bit-28-2-ga gcc-4.8.5-aarch64.tgz
```

Finally decompress the toolchain

```
tar -xvf gcc-4.8.5-aarch64.tgz
```

9. The following will be for L4T 28.1, Jetpack 3.1.
   a. cd to your L4T directory (64_TX2/Linux_for_Tegra_tx2/)
   b. Download the script compile_kernel.sh. If that is not available, follow this
   c. If you're using the script, change the 'export CROSS_COMPILE' path to whatever you've installed the cross compile tool in (from 8).
   d. chmod +x compile_kernel.sh
   e. Run my script with:
      i. *sudo ./compile_kernel.sh initial* if running for the first time
      ii. *sudo ./compile_kernel.sh* if you've already run it at least once before
10. You can flash by either of two methods. I prefer the second one (10.b.) , because I can also easily install other software packages I want:
    a. `sudo ./flash.sh -d kernel/dtb/tegra186-quill-p3310-1000-c03-00-base.dtb -K kernel/Image jetson-tx2 mmcblk0p1` **OR**
    b. cd to the _installer folder created in step 1.c
       i. Run ./start_up.sh
       ii. Go upto the menu options like in step 2., but choose only 'Flash OS image to target' option, and proceed to flash. Remember to put it in *Force Recovery M* when the terminal prompt comes up and asks you to.



JetPack L4T Component Manager

JetPack L4T 3.1 | Jetson TX2 | ○ Full ● Custom | Clear Actions

Package	Current	Size	Action	Progress
▼ Host - Ubuntu			no action	
Tegra Graphics Debugger	-	-	no action	.....
Tegra System Profiler	-	-	no action	.....
Documentation	-	-	no action	.....
▼ VisionWorks Pack			no action	
VisionWorks	-	-	no action	.....
VisionWorks Plus (SFM)	-	-	no action	.....
VisionWorks Object Tracker	-	-	no action	.....
VisionWorks References	-	-	no action	.....
▼ Target - Jetson TX2			mixed	
▼ Linux for Tegra Host Side Ima...			no action	
File System and OS	28.1.0	-	no action	.....
Drivers	28.1.0	-	no action	.....
Flash OS Image to Target	-	4820MB	install 28.1.0	.....
▼ Install on Target			no action	
▼ VisionWorks Pack			no action	
VisionWorks	-	-	no action	.....
VisionWorks Plus (SFM)	-	-	no action	.....

Description | Disk Space | Terminal

Flash 64 Bit OS to device

□ Automatically resolve dependency conflicts

11. Connect ethernet and ssh into it --> shut it down from dev kit --> connect to carrier board