Jetson Nano Developer Kit 40-Pin Expansion Header Configuration

Application Note

DA_09565-001 | July 2019

Document History

Doc_Number

Version	Date	Authors	Description of Change
1.0	July 9, 2019	jsachs, jonathanh, twarren	Initial release

Table of Contents

Customizing the Jetson Nano 40-pin Expansion Header

The Jetson Nano Developer Kit carrier board includes a 40-pin expansion header. By default, all interface signal pins are configured as GPIO inputs, except pins 3 and 5 and pins 27 and 28, which are I2C SDA and SCL, and pins 8 and 10, which are UART TX and RX.

This application note describes how to alter the function of pins on the 40-pin header by using the Jetson Nano Developer Kit pinmux spreadsheet. Note that the pinmux actually configures the SoC on the Jetson module, which ultimately routes the SoC signals to the carrier board's 40-pin header.

If you want to configure other SoC pins, see the full Jetson Nano module pinmux spreadsheet and the NVIDIA L4T Development Guide for complete documentation.

Prerequisites

- A Jetson Nano Developer Kit.
- A computer running Linux (a **Linux host**) which has the GCC toolchain installed that is recommended for building L4T. For more details, see the topic "The L4T Toolchain" section in the *L4T Development Guide*.
- A computer running Microsoft Windows (a **Windows host**) with Microsoft Excel installed.

Download and Customize the Pinmux Spreadsheet

1. On the Windows host, download the Jetson Nano Developer Kit Pinmux spreadsheet from the Jetson Download Center.
2. Open the file in Microsoft Excel and ensure that:
    - The spreadsheet file is writable.
    - The Excel option "Enable Editing" or "Enable Content" is selected. You may need to set one of these options if Excel displays warnings such as "PROTECTED VIEW" or "SECURITY WARNING."
    - The spreadsheet macros are enabled. The spreadsheet needs them to generate device tree source files.
3. Modify columns AR ("Customer Usage") and AS ("Pin Direction") to change the function of individual pins on the developer kit's 40-pin header.
4. Check the following columns through BA to see if any other values must be adjusted.

You need not modify column BB ("IO Block Voltage," concerning SoC connector voltages), since all signals on the carrier board's 40-pin expansion header use 3.3V.

5. Click the Generate DT file button to export your pinmux configuration. When prompted, enter the board name `jetson-nano-sd`. The spreadsheet creates two device tree source files, which you will use in the section "Update the CBoot Pinmux" below.
6. Save the spreadsheet in CSV (comma-delimited) format. You will use this CSV file in To update the CBoot pinmux.

   To save the spreadsheet in CSV format:

   1. Under the File menu in Excel, select Save As.
   2. Enter the filename `jetson-nano-sd.csv`.
   3. Select CSV UTF-8 (comma-delimited) (*.csv) from the menu of filetypes.
   4. Click Save.

Download the L4T Driver Package and Source Files

On your Linux host, download and extract the L4T Jetson Driver Package.

Go to the directory `Linux_for_Tegra` and run the script `source_sync.sh` to download the various source trees. When prompted, enter the correct tag for the L4T version you are using. The release tag name can be found in the L4T Release Notes. For example, for L4T release 32.1, enter the tag `tegra-l4t-r32.1`.

```
$ cd Linux_for_Tegra/
$ ./source_sync.sh
```

Update the U-Boot Pinmux

The pinmux configuration programmed by the U-Boot bootloader is stored in one of the U-Boot header files. Therefore, you must update the U-Boot header, rebuild U-Boot, and reflash. Note that the header file is generated using the `tegra-pinmux-scripts` tool to parse the CSV version of the pinmux spreadsheet. To generate the header file and update the U-Boot bootloader:

1. On your Linux host, obtain a copy of `tegra-pinmux-scripts` by cloning the following Git tree:

   ```
   $ git clone https://github.com/NVIDIA/tegra-pinmux-scripts.git
   ```

2. On your Linux host, place a copy the CSV file in the `tegra-pinmux-scripts/csv` directory.

   ```
   $ cd tegra-pinmux-scripts
   $ mkdir csv
   $ cp <path-to-csv>/jetson-nano-sd.csv csv/p3450-porg.csv
   ```

Where `<path-to-csv>` is the pathname of the directory that contains the CSV file you created in To download and customize the pinmux spreadsheet. This assumes that the CSV file's directory is accessible from the Linux host. If it is not, copy the file by some other means.

3. Import the pinmux CSV file into the `tegra-pinmux-script` internal format:

```
$ ./csv-to-board.py p3450-porg
```

If the following error occurs, it is most likely because the filetype was set to "CSV..." instead of "CSV UTF-8... " when you exported the spreadsheet to CSV.

```
UnicodeDecodeError: 'utf-8' codec can't decode byte 0x96 in position
5617: invalid start byte
```

Make sure that the spreadsheet is exported using the correct CSV format.

4. Generate the U-Boot pinmux header file:

```
$ ./board-to-uboot.py p3450-port > pinmux-config-p3450-porg.h
```

5. Rebuild the U-Boot bootloader.
   1. Copy the U-Boot header file generated in the previous step to this directory:

```
$ cd Linux_for_Tegra/sources/u-boot/
$ cp <path-to-pinmux-scripts>/tegra-pinmux-scripts/pinmux-config-
p3450-porg.h board/nvidia/p3450-porg/
```

   2. Set the build environment:

```
$ export CROSS_COMPILE=<toolchain_install_path>/bin/aarch64-linux-gnu
```

   3. Build U-Boot by running the commands:

```
$ make distclean
$ make p3450-porg_defconfig
$ make
```

   4. Copy the new U-Boot binary to the L4T tree, where it will be found and used when flashing:

```
$ cp u-boot.bin ../../bootloader/t210ref/p3450-porg/
```

Update the CBoot Pinmux

The CBoot bootloader uses device tree files generated by the Excel spreadsheet to configure the pinmux. The pinmux settings in device tree files are only applied by CBoot, and not re-applied by the Linux kernel. To use the updated device tree files you must rebuild the device tree image for Jetson Nano. To update the device tree image:

1. Determine the Jetson Nano device tree version. You can determine the version from the device tree source file name, which can be found by executing this command on Jetson Nano:

   ```
   $ cat /proc/device-tree/nvidia,dtsfilename
   ```

   If the file name is `tegra210-p3448-0000-p3449-0000-a02.dts`, the version is a02. If it is `tegra210-p3448-0000-p3449-0000-b00.dts`, the version is b00.

2. Copy the newly generated device tree files from your Windows host to your Linux host and place them in the following locations under the L4T Driver Package.

   ```
   $ cd Linux_for_Tegra/sources/hardware/nvidia/platform/t210/porg/kernel-
   dts/porg-platforms/
   $ cp <path-to-new-dt-files>/tegra210-jetson-nano-sd-pinmux.dtsi tegra210-
   porg-pinmux-p3448-0000-<nano-dt-version>.dtsi
   $ cp <path-to-new-dt-files>/tegra210-jetson-nano-sd-gpio-default.dtsi
   tegra210-porg-gpio-p3448-0000-<nano-dt-version>.dtsi
   ```

   Where `<nano-dt-version>` is the version that you determined in the previous step.

3. Set the build environment:

   ```
   $ export CROSS_COMPILE=<toolchain_install_path>/bin/aarch64-linux-gnu
   ```

4. Re-build the device tree image:

   ```
   $ cd Linux_for_Tegra/sources/kernel/kernel-4.9/
   $ make ARCH=arm64 tegra_defconfig
   $ make ARCH=arm64 dtbs
   ```

5. Copy the updated device-tree image to the L4T tree:

   ```
   $ cp arch/arm64/boot/dts/tegra210-p3448-0000-p3449-0000-<nano-dt-
   version>.dtb ../../../kernel/dtb/
   ```

Flash Jetson Nano

Use your Linux host to flash the updated U-Boot bootloader and device tree image to Jetson Nano. Remember that Jetson Nano must first be placed in Force Recovery mode.

```
$ cd Linux_for_Tegra/
$ sudo ./flash.sh jetson-nano-qspi-sd mmcblk0p1
```

Notice

The information provided in this document is believed to be accurate and reliable as of the date provided. However, NVIDIA Corporation ("NVIDIA") does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This publication supersedes and replaces all other similar publications for the product that may have been previously supplied.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and other changes to this document at any time, and/or to discontinue any product or service without notice. Customer should obtain the latest relevant documentation before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer. NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document.

NVIDIA products are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer's own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use without further testing or modification. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer's sole responsibility to ensure the product is suitable and fit for the application planned by customer and to do the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer's product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA does not accept any liability related to any default, damage, costs or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document, or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA. Reproduction of information in this document is permissible only if reproduction is approved by NVIDIA in writing, is made without alteration, and is accompanied by all associated conditions, limitations, and notices.

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, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE.

Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA's aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the NVIDIA terms and conditions of sale for the product.

Trademarks

NVIDIA, Tegra, and the NVIDIA logo are trademarks and/or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright

© 2019 NVIDIA Corporation. All rights reserved.

NVIDIA Corporation | 2788 San Tomas Expressway, Santa Clara, CA 95051
http://www.nvidia.com