



(<https://www.instructables.com/contest/classroom2023/?instructables>)

We're giving away
\$40,000
in gift card prizes

Learn more >



Getting Started With PetaLinux

By NAEastland (/member/NAEastland/) in Circuits (/circuits/) > Linux (/circuits/linux/projects/)

109,723

15

11



Download

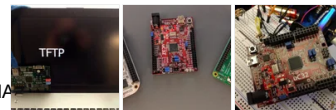
Favorite



(/member/NAEastland/)
By **NAEastland**
(/member/NAEastland/)

Follow

More by
the author:
(/member/NA



Interested in learning a little about embedded Linux? Have you worked with Xilinx FPGAs and want to explore some of the software related to their implementation? If you answered yes, then welcome! If you answered no (to at least the first question) then you should read through anyway, because this is pretty cool stuff.

Welcome to this Getting Started Guide (GSG) to PetaLinux!

PetaLinux is an embedded Linux development solution for Xilinx Zynq chips (an ARM processor with FPGA material, like the ones used [here](#)

[.https://reference.digilentinc.com/reference/programmable-logic/zybo/start?](https://reference.digilentinc.com/reference/programmable-logic/zybo/start?)

[redirect=1id=zybo/zybo\) and here](#)

[https://reference.digilentinc.com/reference/programmable-logic/zedboard/start?](https://reference.digilentinc.com/reference/programmable-logic/zedboard/start?redirect=1id=zedboard/zedboard)

[redirect=1id=zedboard/zedboard](https://reference.digilentinc.com/reference/programmable-logic/zedboard/start?redirect=1id=zedboard/zedboard))) as well as for MicroBlaze designs implemented in fully FPGA chips. This GSG will be using the 2015.4 versions of PetaLinux, SDK, and Vivado and will be targeting the [Zybo](http://store.digilentinc.com/zybo-zynq-7000-arm-fpga-soc-trainer-board/) (<http://store.digilentinc.com/zybo-zynq-7000-arm-fpga-soc-trainer-board/>) from Digilent, so we will be implementing a project on a Zynq target rather than the soft processor core of a MicroBlaze design. For some more info on MicroBlaze, check out [this guide](https://reference.digilentinc.com/learn/programmable-logic/tutorials/artty-getting-started-with-microblaze/start) (<https://reference.digilentinc.com/learn/programmable-logic/tutorials/artty-getting-started-with-microblaze/start>).

For this Instructable, the following prerequisites apply:

- Some familiarity with Linux
- A [Zybo](http://store.digilentinc.com/zybo-zynq-7000-arm-fpga-soc-trainer-board/) (<http://store.digilentinc.com/zybo-zynq-7000-arm-fpga-soc-trainer-board/>) or [Zedboard](http://store.digilentinc.com/zedboard-zynq-7000-arm-fpga-soc-development-board/) (<http://store.digilentinc.com/zedboard-zynq-7000-arm-fpga-soc-development-board/>) to deploy the project onto
- A Linux machine (VM or dual boot setup) of supported OS: Ubuntu 14.04, CentOS 7, SUSE Enterprise 12, RHEL 6.5/6.6/7. *These are for PetaLinux 2015.4. Newer versions may support more recent OS versions.
- At least 20GB of free hard drive space
- Have Vivado and Xilinx SDK [installed](https://reference.digilentinc.com/learn/software/tutorials/vivado-install-guide/start) (<https://reference.digilentinc.com/learn/software/tutorials/vivado-install-guide/start>) (the version you have installed should be the version of PetaLinux you will download)

*Note: While I was able to get PetaLinux 2015.4 running within Ubuntu 16.04, this is not a supported version and is not recommended.

This guide will walk you through the steps of initial setup for PetaLinux on your Linux machine and the implementation of a pre-built design on the Zybo. By the end of this Instructable, you will be able to begin tinkering with the design from either the hardware description language (HDL) side, or create custom applications you can run within the Linux system running on your board. Lets get started!

[Add Tip](#)

[Ask Question](#)

[Comment](#)

[Download](#)

Step 1: Download Installer

Vivado

Embedded Development

SDx Development Environments

ISE

Device Models

CAE Vendor Libraries

Version

2016.2

2016.1

2015.4

2015.3

2015.2

2015.1

2014.4

2014.2

2013.10

2013.04

2012.12

PetaLinux - 2015.4 Installation Files - 2015.4 Installation Files

Important Information

README

Download Type

Last Updated

Answers

Documentation

Installation Files

Dec 15, 2015

Release Notes and Known Issues

PetaLinux Tools Documentation

Download

PetaLinux 2015.4 License and copyrights info (TAR/GZIP - 3.54 MB)

MDS SUM Value:

7e9772ce396997d2898448ef52f25597

Download

PetaLinux 2015.4 Source code (TAR/GZIP - 778.1 MB)

MDS SUM Value:

a692ee5ce54ced935b25194edccc1036

Download

PetaLinux 2015.4 Installer (TAR/GZIP - 1.68GB)

MDS SUM Value:

74e752d382aec28f464eb3ce0d9cbc15

You will need to download the installer

(<http://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/embedded-design-tools.html>), for PetaLinux of the same version as the Vivado and SDK installation you should have on your system. Meaning if you have Vivado and SDK 2015.4 installed, you should download PetaLinux 2015.4. The download may take a while.

Once the download has completed, make a directory in which you would like the PetaLinux tools to be installed in. From your terminal, change directory (cd) into the directory the installer was downloaded into (likely *Downloads*) and run the installer with a specified path to the directory you just created.

Add Tip Ask Question Comment Download

Step 2: Install PetaLinux

```
nate@nate-N56JR: ~/Downloads
nate@nate-N56JR:~$ mkdir PetaLinux
nate@nate-N56JR:~$ cd Downloads
nate@nate-N56JR:~/Downloads$ ./petalinux-v2015.4-final-installer-dec.run ../PetaLinux/
```

Once the download has completed, make a directory in which you would like the PetaLinux tools to be installed in. In your terminal, change directory (cd) into the directory the installer was downloaded into (likely *Downloads*) and run the installer with a specified

path to the directory you just created. Starting from your home directory, enter the following commands (also in screenshot above).

```
mkdir PetaLinux
```

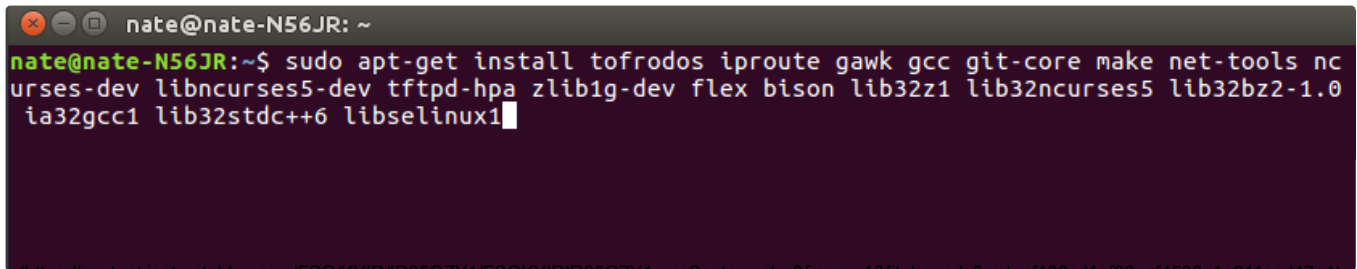
```
cd Downloads
```

```
./petalinux-v2015.4-final-installer-dec.run ../PetaLinux
```

This is just one option of installation location. You can install PetaLinux into any (typical) directory and it will work fine, as long as you have the tools sourced correctly, which we will cover. The PetaLinux tools have an end user licence agreement (EULA) as part of the tool usage, however PetaLinux does not require a license from Xilinx to run.

[Add Tip](#)[Ask Question](#)[Comment](#)[Download](#)

Step 3: Download Dependencies



```
nate@nate-N56JR: ~  
nate@nate-N56JR:~$ sudo apt-get install tofrodos iproute gawk gcc git-core make net-tools nc  
urses-dev libncurses5-dev tftpd-hpa zlib1g-dev flex bison lib32z1 lib32ncurses5 lib32bz2-1.0  
ia32gcc1 lib32stdc++6 libselinux1
```

As with many software development tools, there are a variety of dependencies that you will need to have in order for PetaLinux to operate. Many of the packages may already be installed on your computer, but some may not. A full list of the dependencies is included below (for Ubuntu).

- tofrodos
- iproute
- gawk
- gcc
- git-core
- make
- net-tools
- ncurses-dev
- libncurses5-dev
- tftpd*
- zlib1g-dev
- flex
- bison
- lib32z1
- lib32ncurses5
- lib32bz2-1.0
- ia32gcc1

- lib32stdc++6
- libselinux1

PetaLinux operates using dependencies on 32-bit libraries, so including those specific libraries indicated is required in order for it to operate correctly (even if some may seem redundant). In addition, PetaLinux can utilize a tftp server to streamline the development process, but it is not a requirement for it to function. The specified server package to use is tftpd, however I used tftpd-hpa and would suggest you do the same. If you are unfamiliar with the process of setting it up, you can take a look at this quick [Instructable](https://www.instructables.com/id/Setting-Up-TFTP-Server-for-PetaLinux/) (<https://www.instructables.com/id/Setting-Up-TFTP-Server-for-PetaLinux/>).

Downloading these is the same as any other module that you can get through apt-get, but when you make the following function call (or see image above) you can simply enter each module's name in the same line with spaces between.

```
sudo apt-get install tofrodos iproute gawk (etc)
```

The list of dependencies for other Linux distributions can also be found in the [PetaLinux documentation](http://www.xilinx.com/support/documentation/sw_manuals/petalinux2015_4/ug1144-petalinux-tools-reference-guide.pdf) (http://www.xilinx.com/support/documentation/sw_manuals/petalinux2015_4/ug1144-petalinux-tools-reference-guide.pdf), starting on page 10.

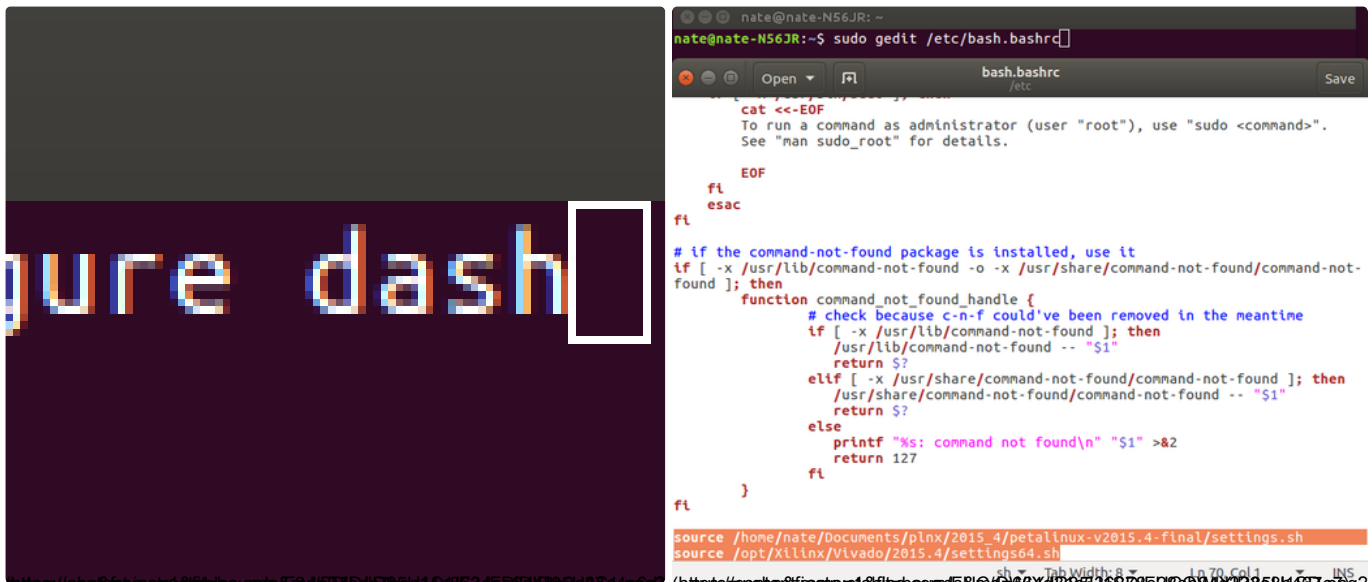
Add Tip

Ask Question

Comment

Download

Step 4: Set Up Bash and Source Settings



The PetaLinux tools require you to use 'bash' as your shell rather than 'dash', which is likely your default shell if you're running Ubuntu. To change this, just enter the following command, which will set your default from 'dash' to 'bash'.

```
sudo dpkg-reconfigure dash
```

The next thing to take care of will be to source the tools for PetaLinux to use within the terminal window. This includes the 'settings64.sh' and 'settings.sh' files in your Vivado and PetaLinux installation directories, respectively. To avoid needing to type the source commands into the shell every time, you can add a couple lines to the .bashrc script. To modify this system wide, use a text editor to open your .bashrc file. For Ubuntu, this will be bash.bashrc located in the /etc directory (see following command and/or first image above).

```
sudo gedit /etc/bash.bashrc
```

Once you have the script open, add the two commands for sourcing the appropriate files. Note that the path indicated here is simply where my installation directories are, so your specific file path may likely be different.

```
source /home/nate/Documents/plnx/2015_4/petalinux-v2015.4-final/settings.sh  
source /opt/Xilinx/Vivado/2015.4/settings64.sh
```

With the two lines added, save the changes and close the editor.

[Add Tip](#)[Ask Question](#)[Comment](#)[Download](#)

https://github.com/Xilinx/petalinux/blob/master/docs/creating_a_new_project.md

To use PetaLinux, you will need a PetaLinux project directory to work in. This can be done either by creating a totally new project or by using a reference design provided in a board support package (BSP). Creating a fresh project provides you with a basic template from which you can start your development. Just change to a directory you would like to create your project in and enter the following command.

```
petalinux-create --type project --template zynq --name test_01
```

The '--type' parameter should remain 'project', the '--template' parameter should be whatever supported architecture you are targeting (either zynq, zynqMP for Ultrascale chips, or microblaze for soft processors implemented in FPGA fabric), and the '--name' parameter can be whatever you want to name your project. Do note that this simply

provides a folder structure for PetaLinux to use and requires you to provide pretty much every part of the build, from the first stage boot loader to the file system, and is not suggested for those new to Linux development. New players should instead use a BSP!

[Add Tip](#)[Ask Question](#)[Comment](#)[Download](#)

Step 6: Create a Project - Board Support Package

(<https://content.instructables.com/ENP/HH10/ID85H404/ENP/HH10ID85H404.png?auto=webp&frame=1&fit=bounds&md=364ac5ccc4007c0cc873c6c4bc3d04c6>)

Creating a new project from a BSP is the simplest way to get started with PetaLinux, since it provides you with an already functioning and bootable Linux image that you start playing with. There are several BSPs available for download [from Xilinx](http://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/embedded-design-tools.html) (<http://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/embedded-design-tools.html>), as well as a [Digilent BSP](https://reference.digilentinc.com/learn/programmable-logic/tutorials/zybo-pmod-pack-demo/start) (<https://reference.digilentinc.com/learn/programmable-logic/tutorials/zybo-pmod-pack-demo/start>) for the [Zybo](http://store.digilentinc.com/zybo-zyng-7000-arm-fpga-soc-) (<http://store.digilentinc.com/zybo-zyng-7000-arm-fpga-soc->

trainer-board/). Once you have the BSP of your choosing downloaded (and extracted if it was a zip file), go to your terminal and change directory to where you would like to create your new PetaLinux project directory and enter the following command.

```
petalinux-create -t project -s
```

Here '-t' is equivalent to '--type' described in the previous step (its parameter should remain 'project') and '-s' is for source and should be followed with the absolute file path to the BSP you want to use. (i.e. /home/nate/Downloads/ZyboPmodPack.bsp).

[Add Tip](#)
[Ask Question](#)
[Comment](#)
[Download](#)

Step 7: Configure, Build, and Package

This step is very straight forward from an end user's perspective, but will require you to accept a bit of 'magic' in the background if you are not intimately familiar with the process of compiling a Linux image from scratch. Suffice it to say that by the end of the configure and build process in PetaLinux, you will have a kernel, file system, first stage and second stage boot loaders, and device tree compiled and ready to be deployed to your hardware target. To run configuration on the BSP project you just created, change directory into the directory that was made with the 'petalinux-create' command, and type in the following.

```
petalinux-config
```

This will initialize a configuration menu for your PetaLinux project (see first image above). Make sure your terminal window is at least its default dimensions or the menu will fail to launch. There are a variety of boot options available to you depending your application. Since the setup and operation of each of the different boot methods is a bit involved, it will be covered in a separate guide. If you are using the [Digilent BSP](https://reference.digilentinc.com/learn/programmable-logic/tutorials/zybo-pmod-pack-) (<https://reference.digilentinc.com/learn/programmable-logic/tutorials/zybo-pmod-pack->

demo/start#tutorial) then the default values in the configuration menu will be fine, so just select the 'Exit' option to leave the config menu. The configuration process will then continue on (this will take some time).

Once the configuration is complete, you will need to build your image by entering the following command.

```
petalinux-build
```

The execution of this command will also take a few minutes to complete, depending on your system. Once this is finished (again, running the Digilent BSP), enter the following command.

```
petalinux-package --boot --force --fsbl ./images/linux/zynq_fsbl.elf --fpga  
./images/linux/linux_bd_wrapper.bit --u-boot
```

After this completes you should have your BOOT.bin and U-boot files ready to go.

[Add Tip](#)[Ask Question](#)[Comment](#)[Download](#)

Step 8: Load Onto SD Card

The SD you use will need to have two partitions on it. I recommend an 8GB card with the first partition (your BOOT partition) formatted as a **File Allocation Table (FAT)** of **1GB**, and the second partition (your rootfs partition) formatted as an **ext4** for the remaining space on the card. The FAT partition will be where your BOOT.bin and image.ub will be stored, while the second partition will be where the file system for your Linux image will be. You can use a utility such as *fdisk* or *gparted* in your terminal to do the formatting for your SD card.

Copy these two files into the first partition of your SD card. You can do this from your PetaLinux project root directory with the following commands.

```
cp images/linux/BOOT.bin /media/BOOT
```

```
cp images/linux/image.ub /media/BOOT
```

Once your boot files have been copied into the BOOT partition of your SD card, copy the root file system into the second partition, your 'rootfs' partition, with the following command.

```
cp images/linux/rootfs.cpio /media/rootfs
```

*Note: the paths "/media/BOOT" and "/media/rootfs" may not be where your SD card is mounted, so you should modify the copy command accordingly.

Once the root file system and boot files have been copied to the two partitions of your SD card run the following PetaLinux command, still from the root directory of the PetaLinux project (also in the second image above).

```
petalinux-util --update-sdcard -d /media/nate/ZYBO_BOOT
```

With that last command entered in you can now unmount and eject your SD card from your computer and load it into your [Zybo](http://store.digilentinc.com/zybo-zynq-7000-arm-fpga-soc-trainer-board/) (<http://store.digilentinc.com/zybo-zynq-7000-arm-fpga-soc-trainer-board/>). You have now just completed this GSG for PetaLinux, congrats!

Add Tip

Ask Question

Comment

Download

Be the First to Share

Did you make this project? Share it with us!

I Made It!

Recommendations

(/Fear-Booth-Raspberry-Pi-Powered-Photo-Booth-With-a/)


Fear Booth - Raspberry Pi Powered Photo Booth With a Spooky Twist (/Fear-

(/The-Ultimate-Camera-RIG-With-Portable-Display/)

The Ultimate Camera RIG With Portable Display (/The-Ultimate-Camera-RIG-With-

Getting Started With PetaLinux	South East Asia: Sri Lankan Photo	South East Asia: Sri Lankan Photo	South East Asia: Sri Lankan Photo
	by NAEastland (/member/NAEastland/) 21 1.1K	by NAEastland (/member/NAEastland/) 3 734	by NAEastland (/member/NAEastland/) 3 734

[\(/contest/robotics2023/\)](/contest/robotics2023/) [\(/contest/classroom2023/\)](/contest/classroom2023/)



Add Tip

Ask Question

Post Comment

We have a **be nice** policy.
Please be positive and constructive.

Add Images

Post

11 Comments

[\(/member/ananthavidhya/\)](/member/ananthavidhya/) ananthavidhya [\(/member/ananthavidhya/\)](/member/ananthavidhya/) Question 1 year ago on Step 8

Answer

Upvote

I am getting error while building petalinux-build command;
Can you please tell me the cause?
I am using zynq ultrascale zcu106 Fpga.BSP is developed by someone .
I am trying to create compiling set up in my system using BSP in ubuntu18.04
using petalinux version 2020.2
I am struct at petalinu-build step with binutils,ncurses errors.
Where initially i thought it is network issue but i can clone respective drivers directly from git so it is may
not be network issue.
Kindly requesting to please help me in resolving this.
Thanks & Regards
A.N.V.Lavanya

(<https://content.instructables.com/FOIQ66WKZZKLM6/FOIQ66WKZZKLM6.png?auto=webp&fit=bounds&frame=1&height=1024&fit=bounds&frame=1&height=1024&width=1024>)

[\(/member/SriHarsha/\)](/member/SriHarsha/) SriHarsha [\(/member/SriHarsha/\)](/member/SriHarsha/) Question 1 year ago on Step 1

Answer

Upvote

Hello,
I have created a basic custom petalinux project (v2019.2). I did not experienced much problem in the
installation/creation/ run the project.

My project has been successfully created and I can create boot files and move them to my SD card boot partitions and also I can run them successfully.

Until here, I just customized only some rootfs settings but not much in configuration settings.

Later I want to changes my confirmation settings such as, I want to change the boot files system from INITRAMFS to eMMC/SD card.

If change these option, the petalinux project is not getting successful. I mean, booting is not successful through SD card. I have followed all the instructions perfectly.

In other case, I want to make my project as a OS file such that I want to create some files/folders in my OS and make them save permanently. That means they should not get deleted if I reboot/power off the ultrascale board.

What should I do to achieve these goals. Can anybody please help me to overcome these problems.

I could not understand where I am making mistakes. I am very new to this area.

Thanks and regards

(/member/gregcasey14910/) gregcasey14910 (/member/gregcasey14910/)
Question 2 years ago on Step 7

Answer

Upvote

1st -- NAEastland, the Petalinux is really, really complex to set up, with so many variant envnoirments, revisions --- I have been at a challenge to get it off the ground. You simple INSTRUCTIONS have REALLY help. I was amazed when I got the the Petlinux-build step and the number of processes and cores took off on my machine.

My current point, I am stuck at is here:

ERROR: Failed to package boot, FSBL file: ./images/linux/zynq_fsbl.elf doesn't exist.

Where is the .elf file come from ? I have seen that extension, when I was in WIINDOWS using Vivado, and VITIS loading FPGA/simple C++ application into the MINIZED, but I don't know how safe it is to just drag over an existing .elf file. any guidance for this? I just want to get linux to boot on the MINIZED

Thanks

Greg

(/member/ParikshitS6/) ParikshitS6 (/member/ParikshitS6/) 3 years ago

Reply

Upvote

I'm getting this error after running petalinux-build & the BOOT.bin file didn't get generated.

I tried 3-4 times but got the same error. Kindly guide me how to solve this.

Thanks a lot.

(https://content.instructables.com/FAD/0E1P/KAGSHR05FAD6E3P6AQSHR2D/FFDBLXMKAQSHR2D.jpg?auto=webp&fit=bounds&frame=1&width=1024) (https://content.instructables.com/FAD/0E1P/KAGSHR05FAD6E3P6AQSHR2D/FFDBLXMKAQSHR2D.jpg?auto=webp&fit=bounds&frame=1&width=1024)

(/member/JeromeC60/) JeromeC60 (/member/JeromeC60/) 5 years ago

Reply

Upvote

Hello,
I completed this set of instructions. Was well documented, compared with actual petalinux pdf which

seems to skip around.

I was not able to update SD card at the end, however every other step was ok. I think update my be removed.

Can you tell me what I should expect? I am using XILINX ZC702 evaluation board. When i insert into SD slot, nothing boots, nothing loads. Should I expect to see a linux OS come up? I am really not clear on what to expect and hot to verify if it is working properly.

Please let me know.

1 reply

(/member/PranayG11/) PranayG11 (/member/PranayG11/) 5 years ago

Reply

Upvote

well, I've done all things except the inclusion of 32bit dependencies. I tried
sudo apt-get install lib32ncurses5.... and also tried all 32 bit dependencies mentioned above, it is saying some "Failed to fetch: http://<some xyz link with .deb extension > Hash sum mismatch " error and unable to fetch some archives..... error and I'm unable sort that issue. I google and I tried everything but nothing worked. please anyone kindly help me to sort this issue, will be grateful for you.Thank you in advance. :)

(/member/JustinD137/) JustinD137 (/member/JustinD137/) 6 years ago

Reply

Upvote

The last petalinux-util --update-sdcard -d call is all you need at the end there. It does the copying for you. Also, you can specify the boot AND the rootfs file paths of the sd card to copy both the boot and file system.

1 reply

(/member/BrettN13/) BrettN13 (/member/BrettN13/) 6 years ago

Reply

Upvote

Thanks for the great intro.

Can anybody please explain how to include a third party application like openVPN to the linux image generated by petalinux? I know you can include openssl and necessary libraries in the petalinux-config menu, but I could not find openVPN.

Could someone please point me to a resource or tutorial that could help be get openVPN as an application in my petalinux image?

Thanks!

(/member/tytower/) tytower (/member/tytower/) 7 years ago

Reply

Upvote

I would suggest you try to explain in laymans terms what the heck you are talking about. What is it , where do I get it ,why would I want it.

I've been a linux only user for 30+ years and have never heard of it

Post Comment

Categories

 Circuits

(/circuits/)


 Workshop

(/workshop/)

 Craft (/craft/)


 Cooking

(/cooking/)

 Living (/living/)

 Outside

(/outside/)

 Teachers

(/teachers/)

About Us

Who We Are (/about/)

Why Publish?

(/create/)

Resources

Sitemap (/sitemap/)

Help (/how-to-write-a-great-instructable/)

Contact (/contact/)

(<https://www.instagram.com/instructables/>)

(<https://www.tiktok.com/@instructables>)

© 2023 Autodesk, Inc.

Terms of Service

(<http://usa.autodesk.com/adsk/servlet/item?siteID=123112&id=21959721>)

|

Privacy Statement

(<http://usa.autodesk.com/adsk/servlet/item?siteID=123112&id=21292079>)

|

Privacy
settings

|

Legal Notices & Trademarks

(<http://usa.autodesk.com/legal-notices-trademarks/>)

 (<http://www.autodesk.com>)