Patrick Jul 4

Since when does Armbian not expand the ROOTFS?

Franco_Gaetan Jul 4

Well when I've imaged from their website, booted from MicroSD card the first thing it says is that it has no space.

I was looking at your patches, I've been documenting everything on my github. Specifcally the k1-x file and how many options are not enabled. When reading the the documentation for u-boot there's a lot that's not there like:

CONFIG_USE_PREBOOT=y CONFIG_PREBOOT="pci enum; usb start; scsi scan; nvme scan; virtio scan"

GitHub - rcman/BPI-F3: Working with my banana-pi F3 board and figuring it out

Working with my banana-pi F3 board and figuring it out - rcman/BPI-F3

Franco

Patrick Jul 4

If you use the extlinux patch it won't require the k1-x env file. It will basically make the unit boot like any other board and also adds overlay support. Of course there are things I haven't tested U-BOOT wise, so I can't speak to every feature that may be available in that ENV file.

Franco_Gaetan Jul 4

Thanks.

Here's the docs from this link: **Booting from TPL/SPL — Das U-Boot unknown version documentation**

NVMe

This methods load the image from an NVMe drive. Required configuration settings include:

- CONFIG_SPL_PCI=y
- CONFIG_SPL_PCI_PNP=y
- CONFIG_SPL_NVME=y
- CONFIG_SPL_NVME_PCI=y
- CONFIG_SPL_NVME_BOOT_DEVICE (number of the NVMe device)
- CONFIG_SYS_NVME_BOOT_PARTITION (partition to read from)

To load from a file system use:

- CONFIG_SPL_FS_FAT=y or CONFIG_SPL_FS_EXT=y
- CONFIG_SPL_FS_LOAD_PAYLOAD_NAME=""

I wanted to know if this worked, that's the path I was going down.

Franco

Patrick **Jul 4**

I have no clue. Best I was able to do using the ENV file was boot from a partition on a SD or EMMC and have the ROOTFS located on the NVMe.

From what I can tell there is no way to just straight boot the NVMe using that file. Maybe it requires flashing to SPI? Which is something I haven't figured out yet.

Franco_Gaetan **Jul 4**

Thank you so much for your help. I just wanted to clarify some things. I can understand the changes to the source that are in the patches. So when you re-compile u-boot with the changes it will give you the files again like bootinfo_emmc.bin and FSBL.bin.

Are they still written to dev/mmvcblk2boot0 as they are in the docs like below?

echo 0 | sudo tee /sys/block/mmcblk2boot0/force_ro sudo dd if=bootinfo_emmc.bin of=/dev/mmcblk2boot0 sudo dd if=FSBL.bin of=/dev/mmcblk2boot0 seek=512 bs=1 sync

Thanks Franco

Patrick **Jul 4**

There seems to be diff ways and combinations of flashing the bins, but this is how I do it.

EMMC

```
echo 0 > /sys/block/mmcblk2boot0/force_ro
sleep .50
dd if="bootinfo_emmc.bin" of="/dev/mmcblk2boot0" bs=512 conv=notrunc
dd if="FSBL.bin" of="/dev/mmcblk2boot0" bs=512 seek=1 conv=notrunc
dd if="FSBL.bin" of="/dev/mmcblk2boot0" bs=512 seek=512 conv=notrunc
dd if="fw_dynamic.itb" of="/dev/mmcblk2" bs=512 seek=1280 conv=notrunc
dd if="u-boot.itb" of="/dev/mmcblk2" bs=512 seek=2048 conv=notrunc
sync
```

SDCARD

```
dd if="bootinfo_emmc.bin" of="/dev/mmcblk0" bs=512 conv=notrunc
dd if="FSBL.bin" of="/dev/mmcblk0" bs=512 seek=1 conv=notrunc
dd if="FSBL.bin" of="/dev/mmcblk0" bs=512 seek=512 conv=notrunc
dd if="fw_dynamic.itb" of="/dev/mmcblk0" bs=512 seek=1280 conv=notrunc
dd if="u-boot.itb" of="/dev/mmcblk0" bs=512 seek=2048 conv=notrunc
sync
```

EDIT: and yes, it will produce the same files using those patches.

---

lu_zero Luca Barbato **Jul 5**

If you want ubuntu from nvme, you may take a recent bianbu (or build one using the instructions **here** , copy the bootloader to the emmc and replace the u-boot configuration with one that sets `root=/nvme0n1p{}` in the final bootargs.

Once the upstreaming efforts complete things will get more smooth.

Franco_Gaetan **Jul 5**

Patrick I have a few questions.

- the patching files, it seems some files are patched and some not. Are these already part of a build tree because I am having issues running the diff git commands. When I've dug down more I see there are other directories that were created like patches/spacemit and so on. Can I not download the changed files already?

Next thing is during my build it never creates FW_PAYLOAD I specify FW_PAYLOAD=y

when make is run, is it because all the patches have not been applied?.

Your system, are you booting from EMMC and using NVME or are you booting from SD card and using NVME?

Sorry for all the questions. Just need to understand it all.

Thanks Franco

Patrick **Jul 5**

My patches are meant to be patched against mainline; opensbi, u-boot and linux. If you are using the BPI or BL sources, you don't need most of the patches I'm using. To make ur life easier I would either pull mainline `u-boot v2022.10` or git clone the BL sources and only apply patches 020 thru 023.

BL: `https://gitee.com/bianbu-linux`

I have the eMMC partitioned with a 100MB offset and flashed with u-boot.

```
sudo patrick@bananapif3:~$ sudo fdisk -l
Disk /dev/mmcblk2: 14.56 GiB, 15634268160 bytes, 30535680 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x4787fb22

Device         Boot  Start      End  Sectors  Size Id Type
/dev/mmcblk2p1      204800 30535679 30330880 14.5G 83 Linux

Disk /dev/nvme0n1: 238.47 GiB, 256060514304 bytes, 500118192 sectors
Disk model: KBG40ZNS256G NVMe KIOXIA 256GB
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x7ffb5003

Device         Boot   Start       End   Sectors  Size Id Type
/dev/nvme0n1p1       204800   1048575    843776  412M 83 Linux
/dev/nvme0n1p2      1048576 500117503 499068928  238G 83 Linux
```

Franco_Gaetan **Jul 5**

Thanks for the info!

Much appreciated. Franco

Patrick **Jul 5**

Precompiled bins if you want them; **https://github.com/pyavitz/binary/releases/download/060420/bpi-f3-u-boot-syslinux.tar.xz**

Franco_Gaetan **Jul 5**

Thanks you're awesome! Franco

Franco_Gaetan **Jul 5**

Hi,

I've been trying to do that but not having any luck. I'm trying to boot a kernel from TFTP. Even though I've make the uImage from the kernel with mkimage it says it can't read the kernel format. The kernel is not compressed. I really wish I didn't have to deal with all this keys stuff for secure boot. I could care less about the box being hacked since I'm doing all the work.

I just want to the box to boot, SUPER EASY! I don't care how I force the boot to work the way I want. When I see the partition structure of bianbu it just makes me mad. Who the heck make their partition in that way? No One!

I should just be able to run the installer from micro sd card and install what ever I want where I want to. Not have to deal with the constant issues that I'd dealing with.

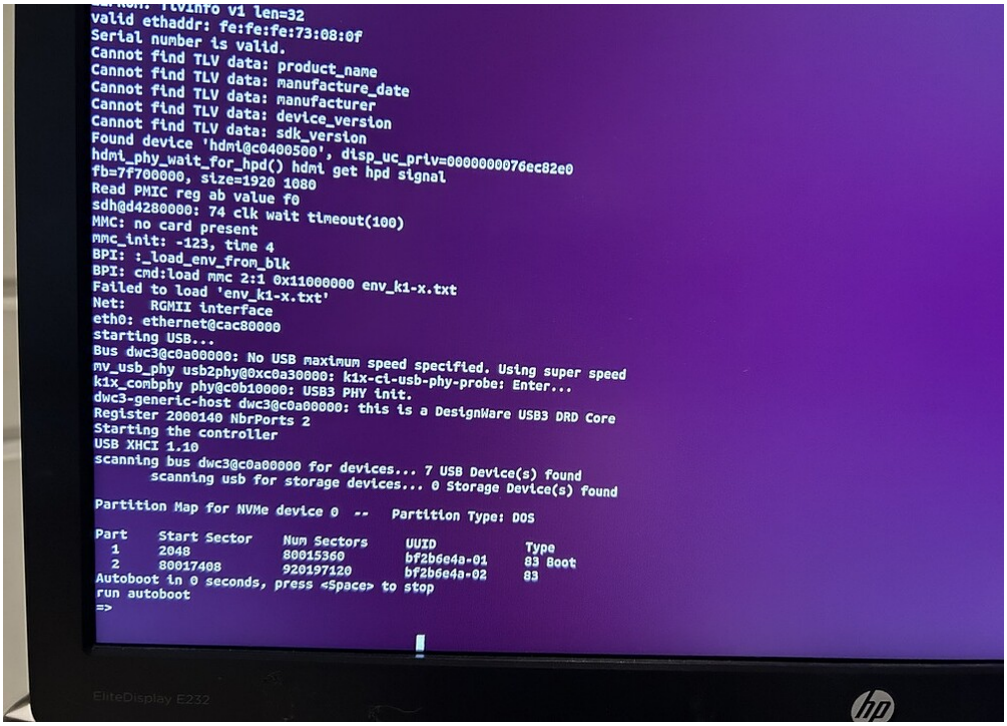
Franco

Franco_Gaetan **Jul 6**

Hi again,

I'm making progress on booting and it shows my nvme partitions. I think my env files wrong.

Can you share what's in your file? Thanks Franco



Patrick **Jul 6**

It is looking for a /boot/extlinux/extlinux.conf file. I left an example of one inside the precompiled u-boot archive I linked above. It appears you have BOOT partition, so on the BOOT partition create a dir “extlinux” and inside place the file “extlinux.conf”. Edit the file to fit your needs.

Franco_Gaetan **Jul 8**

Well I can’t say I’m having any luck booting from NVME. I have started from scratch so many times but nothing yet.

I do have a few more questions. Does the root partition need to be labeled rootfs or does not matter? I noticed in your example for extlinux.conf you have root=UUID which I have also tried.

Right now my NVME has 2 partitions. Partition 1 is boot and partition 2 us root.

I also did out find something interesting. You know when you image bianbu to anything it creates partitions 1 through 4 and each is labeled. Whell instead if the traditional DD command with seek and bs I basically DD the 4 files to their respective partitions and a boot menu appears which is very interesting.

Also while investigating extlinux I found an app call u-boot-menu which sets all the menu options up for you, another interesting find.

Just wanted to let you know that. Franco

Patrick **Jul 8**

It doesn't matter what the label is. You can actually use `root=LABEL=` in place of `root=PARTUUID=` if you prefer.

Yeap. SYSLINUX is basically like a `low rent` GRUB with very basic menu options.

As for the Bianbu and BPI images, I've only ever looked them over, I've never booted one before.

I did recently make some changes to my patching and added BOOT SCRIPT support to it. So if you understand boot.scr thats now an option.

```
Starting the controller
USB XHCI 1.10
scanning bus dwc3@c0a00000 for devices... 4 USB Device(s) found
       scanning usb for storage devices... 0 Storage Device(s) found

Partition Map for NVMe device 0  --   Partition Type: DOS

Part    Start Sector    Num Sectors UUID        Type
  1 204800      843776      7ffb5003-01 83
  2 1048576     499068928   7ffb5003-02 83
Autoboot in 0 seconds, press <Space> to stop
Loading ...
1642 bytes read in 15 ms (106.4 KiB/s)
## Executing script at 02000000
34006068 bytes read in 2084 ms (15.6 MiB/s)
101542 bytes read in 24 ms (4 MiB/s)
6319892 bytes read in 398 ms (15.1 MiB/s)
26 bytes read in 21 ms (1000 Bytes/s)
Loaded overlay.txt: k1-uart2 k1-qspi
Overlaying k1-uart2 ...
263 bytes read in 21 ms (11.7 KiB/s)
Overlaying k1-qspi ...
262 bytes read in 22 ms (10.7 KiB/s)
Booting BananaPi F3 from mmc 0:1 ...
Bad Linux RISCV Image magic!
Trying bootm ...

## Loading kernel from FIT Image at 10000000 ...
   Using 'conf-default' configuration
   Verifying Hash Integrity ... OK
```

```
   Trying 'kernel' kernel subimage
     Description:  Linux 6.1.97
     Type:         Kernel Image
     Compression:  uncompressed
     Data Start:   0x100000bc
     Data Size:    34004480 Bytes = 32.4 MiB
     Architecture: RISC-V
     OS:           Linux
     Load Address: 0x01400000
     Entry Point:  0x01400000
     Hash algo:    crc32
     Hash value:   582f5d9d
   Verifying Hash Integrity ... crc32+ OK
## Flattened Device Tree blob at 31000000
   Booting using the fdt blob at 0x31000000
   Loading Kernel Image
   Loading Ramdisk to 76ead000, end 76ead106 ... OK
   Loading Device Tree to 0000000076e91000, end 0000000076eacfff ... OK

Starting kernel ...
```

Franco_Gaetan **Jul 8**

Actually that's where I get to as well. How long should I wait for the kernel to boot?

I've waited a bit.

Franco

Patrick **Jul 8**

You should see something happening after that right away, unless you have logging and whatnots off. Are you using a custom kernel or something?

Could try removing `console=both` from the extlinux cmdline and set logging to 7.