

Packages - Packaging Request #1965

Depthcharge bootable kernel for Librebooted Asus Chromebook C201

2018-08-25 10:12 AM - CBotulinum

Status:	fixed	% Done:	100%
Priority:	discussion		
Assignee:	Megver83		
Category:			
Description			
None of the kernel images offered by Parabola are bootable by the Depthcharge bootloader, which is used by the Libreboot project to boot kernels. That means anyone using the C201 cannot run an up-to-date kernel unless they compile it and package it - some of us don't have the means to do that. The other option is using the kernel that is packaged by ArchLinux ARM - which carries serious freedom issues with it. So, is it possible to offer a Depthcharge-bootable kernel?			

History

#1 - 2018-08-25 12:15 PM - bill-auger

- Priority changed from bug to wish
- Assignee set to Megver83

#2 - 2018-08-27 03:49 AM - Megver83

- Subject changed from Depthcharge bootable kernel for Librebooted ARM devices to Depthcharge bootable kernel for Librebooted Asus Chromebook C201

You should have written to our mailing list first asking is its possible to boot Parabola on a C201. Now, if you know how and have the solution, then that's a reason for creating a bug report. I say this, because in the past there have been many attempts to get Parabola working on this machine with a libre kernel. However, no one has achieved nothing up to now ...

<https://lists.parabola.nu/pipermail/dev/2016-April/003939.html>
<https://lists.parabola.nu/pipermail/dev/2016-April/003940.html>
<https://libreboot.org/docs/hardware/c201.html#parabola-gnulinux>

The device needs a kernel with specific drives for its hardware, that are not integrated into Linus Torvald's version of Linux, the upstream Linux kernel.

Instead, CrOS systems use a downstream version of Linux, with changes for hardware support. The source code for it is available as part of the ChromiumOS project and it can be rebuilt from source. Until the upstream kernel is ready, this downstream kernel is the best candidate to get a working free system on the device.

I've the skills and time to research and probably create a compatible linux-libre kernel with the C201, but pitifully I do not have a C201.

What you could do meanwhile, is to get that kernel source, deblob it with the deblobbing scripts, and see if it works.

#3 - 2018-08-27 03:49 AM - Megver83

- Priority changed from wish to discussion

#4 - 2018-09-11 05:34 PM - CBotulinum

So, I've managed to successfully boot Linux-Libre.

Here are the instructions I followed to generate an image that Depthcharge can boot, I hope this can be used to make a specific Depthcharge-bootable kernel package:

<https://notabug.org/nuclearkev/dragora-c201#creating-a-bootable-kernel>

All software dependencies of this methods are already present in Parabola (vboot-utils and uboot-tools), and all the other needed files are present on the linked repository.

#5 - 2018-12-06 11:06 AM - CBotulinum

Any news or progress?

I've been following the process I described above quite successfully for a while now, and the kernel boots without further modification.

#6 - 2018-12-07 05:02 PM - Megver83

Sorry for not answering. These are really exciting news. I'll work on linux-libre-chromebook soon.

#7 - 2018-12-09 04:18 PM - Megver83

Btw, did you use arch-kernel.its or kernel.its? (from the repo you built the kernel)

According to what I see, arch-kernel.its is the same as

<https://raw.githubusercontent.com/archlinuxarm/PKGBUILDS/master/core/linux-armv7/kernel.its>, which needs to be built from kernel source, while kernel.its can be built with the linux-libre image installed in /boot, however there are some other differences (I'm not an expert in ARM and device trees) which idk if they are important.

I've uploaded chromebook-kernel to [libre-testing]

<https://git.parabola.nu/abslibre.git/commit/?id=22d17f153183fa86b44816581c1677ca57cc1a79>

which uses kernel.its from dragora-c201.git, I'll later upload linux-libre-chromebook which is the same but using arch-kernel.its (so it will be in the linux-libre 4.19.6 PKGBUILD at [libre-testing] soon). After I upload that one, I'd like you to try both and tell me which one works, and if both works, chromebook-kernel will go to [libre] as linux-libre-chromebook.

#8 - 2018-12-10 08:46 AM - CBotulinum

I use kernel.its.

I'm gonna test the package now, will update soon.

#9 - 2018-12-10 05:01 PM - Megver83

I've uploaded linux-libre-chromebook to [libre-testing]. The only difference with chromebook-kernel is that it uses arch-kernel.its (the same one from ALARM).

Btw, this sign has to be updated every time linux-libre is updated, right?

#10 - 2018-12-10 05:45 PM - CBotulinum

Yes, it has to be updated every time, IIRC.

Sorry that I didn't reply, I just didn't manage to install it, I forgot if it is possible to install a package from [libre-testing] without pulling all the other packages from there instead of [libre], and anyhow I hadn't had much time today, I'll hop on the IRC tomorrow (around the same time as now-ish) and check all this out/update the issue as needed.

#11 - 2018-12-10 06:09 PM - Megver83

You can use pacman -U <package-url> if you don't want to enable [libre-testing]

Remember to try both, chromebook-kernel and linux-libre-chromebook. If you are asked to remove your-freedom, do it (until I get the working kernel in [libre])

#12 - 2018-12-11 07:46 AM - CBotulinum

Okay, thanks for the info, in about 10-ish hours I'll test both packages and report back - got a bit of a busy day today...

#13 - 2018-12-11 10:24 PM - CBotulinum

So, I've hit an unrelated snag.

I tried to update the system beforehand (to minimize variables), but I forgot to run the needed scripts to make the kernel bootable, effectively burning away whatever amount of free time I would have had to test those kernels.

I'll try again tomorrow (IF I have the time), if not, it'll have to wait to Thursday/the weekend.

Really sorry, I think I shouldn't try to mess around with stuff like this after a long and tiring day...

#14 - 2018-12-12 11:16 AM - CBotulinum

Okay, so chromebook-kernel doesn't work, even though I had successfully managed to boot linux-libre 4.19.6 while manually running the scripts to generate the image and flash it.

So, I'll restore from a backup and try the other kernel, in around two hours-ish I should have an update about linux-libre-chromebook.

#15 - 2018-12-12 03:10 PM - CBotulinum

UPDATE:

Just installed linux-libre-chromebook, and it works!

Thanks a lot for the effort!

#16 - 2018-12-12 08:02 PM - Megver83

I suspected chromebook-kernel wasn't going to work. I'll move linux-libre-chromebook to [libre] and make an announce in the news, since this is the

first time someone reports to successfully boot Parabola in a C201 with linux-libre kernel

#17 - 2018-12-13 07:34 AM - CBotulinum

Again, thanks a lot for the effort, this is going to help a lot of people trying to run Parabola on fully free (yet still somewhat modern) systems.

#18 - 2018-12-13 06:48 PM - Megver83

- % Done changed from 0 to 90
- Priority changed from discussion to feature
- Status changed from open to in progress

#19 - 2018-12-14 02:15 AM - Megver83

- % Done changed from 90 to 100
- Status changed from in progress to fixed

The package is now in [libre], updated to linux-libre 4.19.8

#20 - 2018-12-24 09:15 PM - 3p7kp4xn

linux-libre-chromebook seems does not work for me, or maybe not works on some of C201. (I'm using 2GB RAM version)

At first I tried to use this kernel to boot void-musl-arm, got white screen.
then I tried to install parabola, successfully boot parabola with cros kernel, but always got white screen with linux-libre-chromebook.

currently only two kernel works for me:

1. built followed by this https://wiki.gentoo.org/wiki/Asus_Chromebook_C201#4.13.2B_kernel_configuration (at least it can boot voidlinux), but can't recognize partitions on internal storage and rockchip audio device etc.
2. extracted from devuan-arm, change cmdline and repack with vbutil_kernel, but freeze at random time

I was also tried to change kernel cmdline to add debug or change loglevel to linux-libre-chromebook kernel, but no luck, always got white screen.

does linux-libre-chromebook recognize internal storage (those partitions show in lsblk) and rockchip audio device ROCKCHIP-I2C well?

BTW the way I install linux-libre-chromebook is just download the package and extract it, dd **vmlinux.kpart** to /dev/sda1 (sda1 is the first partition on my USB stick, sda2 is the root partition)

#21 - 2018-12-29 10:31 PM - Megver83

Did you try installing ALARM and then migrating to Parabola? IMO that's the easiest way to install Parabola in ARM devices

3p7kp4xn wrote:

does linux-libre-firmware recognize internal storage (those partitions show in lsblk) and rockchip audio device ROCKCHIP-I2C well?

Why should it? It's just firmware for non-critical stuff, like some atheros wifi cards and things like that, but all the rest (like audio and filesystem drivers) are in-kernel.

#22 - 2018-12-29 10:50 PM - bill-auger

isn't the ARM tarball the easiest way to install? - it was intended to be

#23 - 2019-01-02 06:43 PM - Megver83

bill-auger wrote:

isn't the ARM tarball the easiest way to install? - it was intended to be

I know, but to be honest, I think that it was unnecessary. Having ALARM tarballs and the migration guide (which is easy) makes our tarballs pointless. Plus, our tarball is generic, while ALARM creates tarballs for specific devices and SoCs.

Different to x86, an install ISO lets you test the system before installing it, while on ARM you must directly install the OS (at least in most ARM devices), so in that case there's no reason to bother on creating our own rootfs tarballs if they can easily migrate to Parabola.

With this I'm not saying that we should delete them, but think it twice if it's a good idea before going ahead implementing new stuff. This is my conclusion, at least, based on my experience with ARM.

#24 - 2019-01-02 08:18 PM - bill-auger

that does sound reasonable, but is there any non-free software on the basic ALARM system? - if there is, then we can not actually recommend it

#25 - 2019-01-02 10:33 PM - Megver83

Of course there is (it's a base system anyways), afaiK only linux kernel and linux-firmware. But it wouldn't be such a terrible thing if we give the advice of installing your-freedom (and by migrating you completely clean your system from non-free crap).

#26 - 2019-01-03 10:01 PM - bill-auger

the kernel for sure, i was mainly wondering how much work it would be to explode each of the ALARM tarballs, replace the kernels and whatever else, set the mirrors to parabola, and re-pack them; or would that be more trouble than re-building them in a VM

#27 - 2019-01-04 02:50 AM - 3p7kp4xn

- File dmesg_linux-libre-chromebook_parabola_2019-01-04T02_10_16Z.log added

no I didn't try migrating from ALARM to Parabola before, but I just tried.

what I really want to ask is, does linux-libre-chromebook the kernel can recognize partitions of internal storage? cuz I never build a mainline kernel can recognize partitions of internal storage.

after migrated to parabola from archlinux and installed linux-libre-chromebook then reboot. (I was about give up and just leave the white screen there and write this) after several minutes, it shows text of systemd-service start up info and **finally booted into tty!** I checked output of lsblk, it does not recognize partitions of internal storage.

then I rebooted it, it cost 9 min on white screen and 2 min waiting for systemd service or something then boot to tty.

but I can't boot this kernel on voidlinux, still white screen after wait for 20 min.

BTW it's the output of dmesg (the second time I boot this kernel on parabola on c201).

#28 - 2019-01-04 08:30 PM - Megver83

Are you booting from SD Card or USB? did you tried different storage devices?

3p7kp4xn wrote:

but I can't boot this kernel on voidlinux, still white screen after wait for 20 min.

Of course you cannot boot it, you must install linux-libre (the kernel) and then linux-libre-chromebook (the kernel **sign**, which dd's to the kernel partition)

Please send us more info, like lsblk and fdisk outputs. Also, how was the performance with ALARM? did it have the same issues of Parabola?

Also, after reading the dmesg log, I searched a bit and found [this](#) (check it, as it may be the solution)

P.S.: I see that the [NanoPi K2 had the same issue](#) and it was solved by modifying the dts file. However it would be strange for the C201 since [CBotulinum](#) could boot Parabola correctly without having to patch the kernel.

#29 - 2019-01-06 08:34 AM - CBotulinum

If you want I can try to test stuff here, what does the issue seem to be? Or in other words what would you want me to test?

#30 - 2019-01-06 08:14 PM - Megver83

CBotulinum wrote:

If you want I can try to test stuff here, what does the issue seem to be? Or in other words what would you want me to test?

If everything works for you, we don't need you to test anything for now. However, it would be nice if you (or we) could make a wiki page explaining step-by-step how to install Parabola in a Librebooted C201.

Btw, I didn't ask you, did you migrated from ALARM or directly installed Parabola in an SD Card/USB? (this is one of the things I'd write in this future wiki page)

#31 - 2019-01-07 07:38 AM - CBotulinum

I did actually migrate, was much simpler to do so (plus, I had used ALARM on different devices a few times before, so...)

#32 - 2019-01-07 01:45 PM - Megver83

CBotulinum -

OK, so you basically installed ALARM as explained [here](#) and then migrated as explained in our Wiki, right?

#33 - 2019-01-07 06:42 PM - 3p7kp4xn

- File `voidlinux_musl_c201_dmesg_fdisk_lsblk.txt` added
- File `parabola_migrated_from_alarm_dmesg_fdisk_lsblk.txt` added
- File `parabola_installed_from_tarball_dmesg_fdisk_lsblk.txt` added

i am booting from a Kingston USB stick, I always remove the microSD before boot from USB. no I didn't and may won't try to install parabola to my microSD, I don't have an extra microSD to install it to.

since the kernel I built by following gentoo wiki can boot my voidlinux without kernel modules, I thought this kernel could also boot without installing kernel modules. but seems I was wrong.

I just tried to extract kernel modules from linux-libre and write `vmlinux.kpart` to the boot partition.

voidlinux (microSD): cost about 617 seconds to boot to tty, touchpad does not works (tried on `i3wm`), does not report error while `fdisk -l`

parabola installed from tarball (USB): 715s to boot to tty, kernel report errors while `fdisk -l`

parabola migrated from alarm (USB): 970s to boot to tty, kernel report errors while `fdisk -l`

Don't have any performance issues with alarm nor parabola, but only tried console. didn't try X.

I think these I/O error was because this kernel can't recognize `mmcblk2` (the internal storage) well, not a hardware issue.

the old mainline kernel (4.13.0-rc1) built by follow the [\[\[gentoo wiki\]\]](https://wiki.gentoo.org/wiki/[[gentoo wiki]])(

https://wiki.gentoo.org/wiki/Asus_Chromebook_C201#4.13.2B_kernel_configuration) works for me (it just can't recognize any partitions of internal storage, or can't boot sometimes but don't know why it could most most of times nowadays).

BTW CrOS kernel from alarm or devuan-arm can make it freeze at random time (at least on voidlinux and devuan-arm, or just the touchpad and keyboard suddently unusable? IDK but I can only solve that by hard reboot).

I don't want to build kernel for c201 recently, it's too painful (got white screen or can't boot for so many times before!).

maybe it's because of different version of c201?

mine is C201PA-DS01 made in 2015-06 (2GB RAM + 16GB ROM).

[CBotulinum](#) can you provide this info? change `mmcblk0` to other if it's not your internal storage. here is mine. and the size of RAM and internal storage?

```
$ cat /sys/class/block/mmcblk0/device/uevent
DRIVER=mmcblk
MMC_TYPE=MMC
MMC_NAME=HAG2e
MODALIAS=mmc:block
$ cat /sys/class/block/mmcblk0/device/date
04/2015
```

i also installed ALARM as explained here and then migrated as explained in our Wiki

#34 - 2019-01-08 04:22 AM - libreantho

Hi I have manage to install on parabola on sd card, works great! But I'm am not able to transfer it to emmc.

when i `fdisk` I get:

The primary GPT table is corrupt, but the backup appears OK, so that will be used.

I've tried `dd` the sd card to the emmc but it boot into recovery.

When I `dd`, it give "error Buffer I/O error on dev `mmcblk2`, logical block 0", lost async page write on 9 block and 4: `print_req_error: I/O error, dev mmcblk2, sector XXXX`

Anyone manage to put it on emmc?

#35 - 2019-01-08 04:33 AM - Megver83

libreantho -

This may help

https://unix.stackexchange.com/questions/423923/installing-archlinuxarm-in-internal-storage-on-asus-chromebook-c201?noredirect=1#comment763885_423923

https://wiki.debian.org/InstallingDebianOn/Asus/C201#Installing_to_internal_memory_from_SD_card

^here at "reconfigure the kernel" skip the `vboot_util` part since the `vmlinuz.signed` file is provided in `linux-libre-chromebook` as `/boot/vmlinuz.kpart`

#36 - 2019-01-09 02:26 AM - libreantho

link is dead

https://unix.stackexchange.com/questions/423923/installing-archlinuxarm-in-internal-storage-on-asus-chromebook-c201?noredirect=1#comment763885_423923

so I found information here (urjaman comment was the one that worked best for me):

<https://github.com/altreact/archbk/issues/3>

1. unmount all partitions on mmcblk2*, if your system automounts things.
2. shrink & move ROOT-B (mmcblk0p5)


```
cgpt add -i 5 -b 282624 -s 32 /dev/mmcblk2
```
3. shrink & move STATE (mmcblk0p1)


```
cgpt add -i 1 -b 282656 -s 32 /dev/mmcblk2
```
4. move & enlarge ROOT-A (mmcblk2p3)


```
cgpt add -i 3 -b 282688 -s 30470080 /dev/mmcblk2
```
5. make a new filesystem on ROOT-A


```
partx -a /dev/mmcblk2
mkfs -t ext4 /dev/mmcblk2p3
```
6. make mountpoints


```
mkdir /tmp/{old,new}root
```
7. mounts for a clean copy


```
mount --bind / /tmp/oldroot
mount /dev/mmcblk2p3 /tmp/newroot
```
8. copy system and unmount target


```
cd /tmp/oldroot
cp -a * ../newroot
umount /tmp/newroot
```
9. install kernel to KERN-A (mmcblk2p2)


```
dd if=/boot/vmlinuz.kpart of=/dev/mmcblk2p2
```
10. set mmcblk0p2 to the high priority and successful


```
cgpt add -i 2 -P 10 -S 1 /dev/mmcblk2
```
11. set mmcblk0p4 to low priority so it is not used


```
cgpt add -i 4 -P 0 -S 0 /dev/mmcblk2
```

unfortunately I get white screen with both parabola and archlinux-arm-chromebook (mainline kernel). I tried with the latest veyron image which did half-boot(system-md hang, it does exactly the same on sd card)

Maybe because of the issue mentioned in the link below? I haven't got an older image to test with downgraded system-md:
<https://archlinuxarm.org/forum/viewtopic.php?fi=15&t=13249&p=59997&hilit=emmc+c201+emmc+c100p#p59997>

So I haven't managed to get this to work on emmc... My latest guess would have been to try from an old veyron image(with the 3.14 kernel and unupgraded systemmd) install it on emmc and upgrade to parabola afterward. But I haven't managed to get an old image to test.

Maybe I'm doing something wrong and somebody can help.

#37 - 2019-01-09 11:25 AM - CBotulinum

Megver83 wrote:

OK, so you basically installed ALARM as explained [here](#) and then migrated as explained in our Wiki, right?

Yes, that is exactly what I did.

As for the touchpad, I can also say that on my device it doesn't work too, I mean it does get recognized by lsusb, but I have no idea why I can't get it to work (not that I've put much effort or really care about that at the moment).

#38 - 2019-01-09 03:48 PM - libreantho

does this works for the touchpad:

```
pacman -S xf86-input-synaptics
```

source: <https://calvin.me/arch-linux-samsung-arm-chromebook/>

#39 - 2019-01-09 04:35 PM - libreantho

I followed the wiki instruction to the letter I was able to boot with the ArchLinuxARM-armv7-chromebook-latest.tar.gz but not with ArchLinuxARM-veyron-latest.tar.gz. Migrating from ArchLinuxARM-armv7-chromebook-latest to parabola was straightforward and worked. What I could not do was to get a booting image of ALARM on the EMMC then migrating to parabola. I could also not install parabola to the EMMC from parabola by copying the root and the kernel on the right partition of the EMMC. I managed to get ArchLinuxARM-veyron-latest on the EMMC, But System-md hang and I cannot get to the command prompt. Does anyone managed to install parabola on the EMMC? If so was it with the Veyron image? Was it an old image?

Thanks

#40 - 2019-01-10 06:29 AM - CBotulinum

IIRC, the touchpad situation wasn't remedied by installing xf86-input-synaptics, anyhow, I **might** try to get to the bottom of this tomorrow - if I have enough time, since my server has been facing some odd failures in different subsystems, which would probably take a while to debug.

#41 - 2019-01-10 06:06 PM - libreantho

have you modified xorg and changed the mouse preference as in the tutorial link.

Then set-up the config file at `/etc/X11/xorg.conf.d/70-synaptics.conf`

```
Section "InputClass"
    Identifier "touchpad"
    Driver "synaptics"
    MatchIsTouchpad "on"
    Option "FingerHigh" "5"
    Option "FingerLow" "5"
    Option "TapButton1" "1"
    Option "TapButton2" "3"
    Option "TapButton3" "2"
    Option "HorizTwoFingerScroll" "on"
    Option "VertTwoFingerScroll" "on"
EndSection
```

if it does not work, maybe try this with `xf86-input-libinput?`:

<https://unix.stackexchange.com/questions/324627/touchpad-not-working-properly-after-upgrade-arch-on-chromebook>

#42 - 2019-01-29 04:40 PM - Time4Tea

Hi, I've been following this thread and I'm just wondering if anyone has had any success with installing a system to the internal eMMC? Some of the earlier comments suggested it was proving problematic.

#43 - 2019-01-30 01:21 AM - Time4Tea

I've been digging and I've found out some things about the eMMC:

It appears that the reason the eMMC partitions are not visible in Linux is because the primary GPT header signature has been set to 'IGNOREME'. Apparently, the standard signature for GPT is 'EFI PART', and this signature is required for Linux to recognize the partition table. I installed the 'vboot-utils' package, which includes the `cgpt` tool, which can apparently be used to manage Chrome OS GPT tables. The command `cgpt legacy -e <device>` is supposed to change the header signature back to 'EFI PART'. However, I tried that and it didn't work (just gave me I/O errors).

This suggested some sort of write-protection, so I also downloaded and compiled the 'mmc-utils' package from [here](#). This apparently allows querying/probing of MMC devices. I entered the command `mmc writeprotect user get <device>`, which shows the write-protect status of the device and got the following output:

```
# ./mmc writeprotect user get /dev/mmcblk2
Write Protect Group size in blocks/bytes: 16384/8388608
Write Protect Groups 0-0 (Blocks 0-16383), Permanent Write Protection
Write Protect Groups 1-1878 (Blocks 16384-30785535), No Write Protection
```

So, it seems that permanent write protection is enabled on blocks 0-16383, which covers the PMBR, primary GPT header and table, as well as a partition that seems to contain some 'Chrome OS firmware'. I'm not 100% sure what this 'permanent write protection' means; however, judging by [this document](#), it looks like it may be implemented in the hardware and irreversible(!).

If that is the case, then the eMMC seems to be completely locked-down, with seemingly no possibility to alter the primary GPT. So, unless someone can come up with a clever way of using just the secondary GPT table (which doesn't seem to be locked), I'm not sure that the eMMC will be very usable.

I hate when people do stuff like this ... :-(

#44 - 2019-01-30 10:10 AM - CBotulinum

I honestly never tried to install to the internal eMMC, nor is that a thing I wish to do.

Having said that though, Debian has a guide for installation on the internal eMMC, you could try that:

https://wiki.debian.org/InstallingDebianOn/Asus/C201#Installing_to_internal_memory_from_SD_card.

Though I think you need a patched kernel for it, plus, it is kind of inconvenient (eMMC is tiny on this device).

#45 - 2019-01-30 11:06 AM - Time4Tea

[CBotulinum](#): ok, I will give it a try, although I don't see how it would work, if the partition table has been permanently write-protected. As I said above, I tried altering the GPT signature with `'cgpt legacy -e'` and it couldn't do that.

I agree the eMMC isn't huge (~16GB), but it's big enough for a root partition (e.g. for a backup system). It would be nice to be able to use the space and wipe Chrome OS off completely.

#46 - 2019-01-30 01:13 PM - CBotulinum

That partition could be housing some of the Libreboot files IIRC, maybe that means one would have to remove the write-protect screw? But that might ruin the bootloader then.

Try asking in `#libreboot` on freenode, I think they've mapped those partitions out.

#47 - 2019-01-30 01:16 PM - Time4Tea

I tried the instructions on that Debian page, but got the following when I tried to run 'cgpt add':

```
# cgpt add -i 5 -s 32 /dev/mmcblk2
WARNING: Primary GPT header is being ignored
ERROR: Entries' crc corrupted
ERROR: -i 5 -s 32
```

I suspect it is linked to the write-protection that I mentioned before. I will try again after flashing Libreboot and see if anything changes.

#48 - 2019-01-30 11:46 PM - libreantho

For those who wish to install to the EMMC.

PrawnOs has implemented a logic to install to emmc(debian) you can check the code here:

<https://github.com/SolidHal/PrawnOS/blob/master/scripts/InstallScripts/InstallToInternal.sh>

#49 - 2019-01-31 12:37 AM - Time4Tea

[libreantho](#): Interesting. I will give it a try and report back in a couple of days. Thanks!

#50 - 2019-02-01 12:10 PM - Time4Tea

I migrated from ALARM to Parabola (using the instructions [here](#)). It worked, but there were a handful of minor issues. I will open a new bug report to flag them. I actually now have both ALARM and Parabola installed on different partitions on the microSD (so I can switch between the two for troubleshooting).

The touchpad does not work for me with Parabola either. To me, this looks like a hardware driver issue, not to do with X. There is no 'mouse0' device present in either /dev/input or the /sys device tree. During the ALARM boot, there is a line in the dmesg log that says the 'Elan Touchpad' was recognized; however, that line is not present in the boot log for Parabola. I compared the output of 'lsmod' between the two systems - ALARM had two modules that Parabola didn't: 'Mousedev' and 'Joydev'. I modprobed those into Parabola, but the touchpad still didn't work.

Those were the only modules that ALARM had that Parabola didn't that looked like they could have been relevant to the touchpad (the others were all crypto stuff). So, that suggests to me that the problem may either be a built-in kernel setting that is configured differently in the libre kernel, or perhaps this touchpad is not supported by a free driver? I noticed in ALARM that it seems to be connected to the i2c bus - perhaps there is some 'i2c-hid' option that is not configured in the kernel?

Does anyone have any other ideas?

#51 - 2019-02-01 01:03 PM - Time4Tea

Definitely a kernel issue. The touchpad works fine if I boot the ALARM kernel & modules with the Parabola root partition.

#52 - 2019-02-01 03:41 PM - Megver83

as this is definitively a kernel issue, please try with Parabola + ALARM non-free kernel (packages: linux-armv7, linux-armv7-chromebook)

you'll have to delete your-freedom for this

P.S: just for commenting. To me it looks like linux-libre doesn't have a good support for ARM hardware (not their fault anyways). Because I remember that in the past I attempted to install Libreboot in a T400 without success, and by using Parabola in a Banana Pi M1. The problem was that the OS didn't detect the /dev/spidev* devices, no matter how many kernel modules I loaded, I could never get it. And my old problem with SPI devices looks to be the same with mouse* devices.

P.S. 2: I use the same Arch{32,ARM} configs for kernels. The only changes I make are to disable options that don't work with blobs.

#53 - 2019-02-01 06:43 PM - Time4Tea

[Megver83](#): I just installed those ALARM linux packages into Parabola and the touchpad now works. So, from what you said about using the same kernel config as ALARM, does that imply that the touchpad is not supported by the free kernel/drivers?

#54 - 2019-02-02 01:48 AM - Megver83

I fear yes, or at least not OOTB. However, idk if [CBotulinum](#) had this problem.

The other option is to check Debian. See if the touchpad works without non-free software and apply any change they make to their libre kernel to ours.

#55 - 2019-02-02 12:06 PM - Time4Tea

[Megver83](#): ok, I will try the Debian kernel at some point over the next few days.

#56 - 2019-02-02 12:44 PM - CBotulinum

IDK really about the touchpad, as I never liked the concept at all (on my Thinkpad I've disabled it altogether and use the Trackpoint/a USB mouse),

though after some digging around, I think I found something that might help:

<https://github.com/hugegreenbug/xf86-input-cmt>

#57 - 2019-02-05 05:23 PM - Megver83

- Priority changed from feature to discussion

it's clear that this turned into a discussion - I'm changing the issue's priority as such

#58 - 2019-02-07 03:18 AM - Time4Tea

Regarding the touchpad: I downloaded the sources for the latest ALARM and linux-libre kernels and then 'grepped' them both for the string "Elan Touchpad". All of the same files that contain that string still appear to be present in linux-libre. The main driver source file seems to be `drivers/input/mouse/elan_i2c_core.c`, the start of which says it is free software (GPLv2) (although it also says 'copyright of ELAN Microelectronics Corp.').

So, if the driver source file is present in the linux-libre source, then doesn't that point to the problem being configuration-related, if that driver isn't being included in the kernel?

#59 - 2019-02-07 12:01 PM - Time4Tea

It does appear to be the configuration. I checked the libre kernel config and literally the only mouse option that wasn't enabled is 'elan_i2c_touchpad_support'. If I have time over the next couple of days, I'll see if I can re-compile it with that option and try it out.

(I have also checked and the `elan_i2c` module is built in to the ARCH kernel)

#60 - 2019-02-08 03:53 AM - Time4Tea

I compiled the `elan_i2c` module; loaded it and it works.

#61 - 2019-02-16 05:15 AM - Megver83

crap, sorry for disabling it. I did because the `elan_i2c.h` file tries to load a blob

[here](#) you can see the script I use to remove "blobbed" config lines from the kernel.

<https://git.parabola.nu/~megver83/deblob-config.git/tree/versions/4.20>

I'll enable it in the next release

Thanks a lot [Time4Tea](#)

P.S.: I see that I also disabled `CONFIG_TOUCHSCREEN_ELAN` - is this necessary?

#62 - 2019-02-18 01:45 AM - Megver83

I've [updated linux-libre](#)

Now it has [CONFIG_MOUSE_ELAN_I2C enabled](#)

Please, test it to check that everything is fine now.

#63 - 2019-02-19 11:09 AM - CBotulinum

Thanks a lot!

Just tested it on my C201, everything seems to be OK.

#64 - 2019-02-24 06:49 PM - Time4Tea

I upgraded the kernel and the trackpad seems to work fine for me as well.

It seems strange that it was trying to load a blob, and yet the driver seems to work fine without it. I wonder what that blob was for then?

#65 - 2019-03-05 05:56 PM - CBotulinum

Just so I won't clutter stuff up with a new issue, which WiFi adapters are you using? Because I've tried a TP Link TL-722N (`ath9k_htc`), but it is extremely unstable at times with the C201...

#66 - 2019-03-05 07:31 PM - libreantho

Have you checked the patch that solidhal did for the debian based prawnOS?

<https://github.com/SolidHal/PrawnOS/issues/38>

<https://github.com/SolidHal/PrawnOS/commit/08bc81f9e969f07d27d7d2070dfda6c2b25efa56>

Perhaps it can help fix the problem

#67 - 2019-03-06 10:02 AM - CBotulinum

But that is a terrible hack, not a great idea, and this device used to work before, so...

#68 - 2019-03-24 02:09 PM - libreantho

Any luck with the wifi? Using a usb hub seem to mitigate the issue for me but it's not running smoothly. Also system-md seem to have a hard time doing shutdown when the usb wifi adapter is plugged-in. I'm using kernel 5.0.2.

#69 - 2019-03-29 01:11 AM - Time4Tea

Sorry, I haven't checked this discussion for a while and just saw the posts about wifi. I bought one of these wireless USB adapters from Think Penguin:

<https://www.thinkpenguin.com/gnu-linux/penguin-wireless-n-usb-adapter-gnu-linux-tpe-n150usb>

I only got round to trying it today, but it doesn't seem to work well. When I plug it in, it works ok for a minute or two, then it seems to freeze up the whole system (nothing responds to mouse clicks). When I unplug it, the system magically unfreezes and works fine again. So, it seems like possibly an issue with the kernel module?

I have tried it with Parabola (kernel 4.20.11) and ALARM (4.20.6) and the behavior is the same.

It seems this device uses an Atheros AR9271 wifi chip. I don't know if that is the same as the TL-722N that CBotulinum mentioned? Has anyone managed to get a wifi device to work with the C201?

#70 - 2019-03-31 10:37 AM - CBotulinum

So, basically the same problems I was facing, did you try it with an externally-powered hub? That used to do the trick a while back, but a few months ago it doesn't seem to work here, still, worth a try.

#71 - 2019-03-31 12:11 PM - Time4Tea

[CBotulinum](#): no, I haven't tried a USB hub, as I don't have one of those unfortunately.

FYI: I created a new issue for the USB wifi here: [\[\[https://labs.parabola.nu/issues/2261\]\]](https://labs.parabola.nu/issues/2261), so that it is flagged as an open issue and can be tracked.

#72 - 2019-03-31 12:15 PM - bill-auger

wow - i had no idea this was discussion was still going on after the issue was closed 4 months ago - a discussion this long and covering so many different topics would have been much better on the mailing list

#73 - 2021-01-04 04:52 AM - bill-auger

if anyone receiving this message, who owns a C201, and is still using the parabola 'linux-libre-chromebook' kernel, or would like to use it, please let us know - we would like to support the C201, but currently none of the parabola devs own one; so we need someone to volunteer to use it regularly ,and report any problems

Files

dmesg_linux-libre-chromebook_parabola_2019-01-04T02_10_16Z.log	46.4 KB	2019-01-04	3p7kp4xn
parabola_installed_from_tarball_dmesg_fdisk_lsblk.txt	61.8 KB	2019-01-07	3p7kp4xn
parabola_migrated_from_alarm_dmesg_fdisk_lsblk.txt	131 KB	2019-01-07	3p7kp4xn
voidlinux_musl_c201_dmesg_fdisk_lsblk.txt	65 KB	2019-01-07	3p7kp4xn