

Contents: [Dobrica Pavlinu's random unstructured stuff]

- Dobrica Pavlinu's random unstructured stuff (kernel)
- Dobrica Pavlinu's random unstructured stuff (cpuinfo)
- Dobrica Pavlinu's random unstructured stuff (Rooting)
 - ◆ Dobrica Pavlinu's random unstructured stuff (nvflash)
 - ◆ Dobrica Pavlinu's random unstructured stuff (fastboot.stock.bin)
 - ◆ Dobrica Pavlinu's random unstructured stuff (backup)
- Dobrica Pavlinu's random unstructured stuff (Linux)
 - ◆ Dobrica Pavlinu's random unstructured stuff (links)
 - ◆ Dobrica Pavlinu's random unstructured stuff (phh 2.6.32 kernel with working suspend and audio)
- Dobrica Pavlinu's random unstructured stuff (mainline kernel)
- Dobrica Pavlinu's random unstructured stuff (u-boot from APX)
 - ◆ Dobrica Pavlinu's random unstructured stuff (build u-boot)
 - ◆ Dobrica Pavlinu's random unstructured stuff (build kernel)
 - ◇ Dobrica Pavlinu's random unstructured stuff (create initrd with new modules)
 - ◆ Dobrica Pavlinu's random unstructured stuff (create bootloader file)
- Dobrica Pavlinu's random unstructured stuff (serial port)
- Dobrica Pavlinu's random unstructured stuff (bootstrap debian)
- Dobrica Pavlinu's random unstructured stuff (u-boot over pax. usb filesystem)
- Dobrica Pavlinu's random unstructured stuff (wifi)
- Dobrica Pavlinu's random unstructured stuff (disk speed)
 - ◆ Dobrica Pavlinu's random unstructured stuff (tuhing block size)
- Dobrica Pavlinu's random unstructured stuff (flash u-boot to mmc)
 - ◆ Dobrica Pavlinu's random unstructured stuff (tegrarcM to get bct)
 - ◆ Dobrica Pavlinu's random unstructured stuff (create boot image)
- Dobrica Pavlinu's random unstructured stuff (i2c)

Tegra 2 device

kernel

- <https://gitlab.com/groups/ac100> (gitorious is dead)

cpuinfo

```
$ cat /proc/cpuinfo
Processor       : ARMv7 Processor rev 0 (v7l)
processor       : 0
BogoMIPS       : 1998.84

Features       : swp half thumb fastmult vfp edsp vfpv3 vfpv3d16
CPU implementer : 0x41
CPU architecture: 7
CPU variant    : 0x1
CPU part       : 0xc09
CPU revision   : 0

Hardware       : NVIDIA Harmony Development System
```

```
Revision      : 0000
Serial        : 0000000000000000
```

```
$ cat /proc/meminfo
MemTotal:      383348 kB
MemFree:       106528 kB
Buffers:       30116 kB
Cached:        123316 kB
SwapCached:    0 kB
Active:        164792 kB
Inactive:      85020 kB
Active(anon):  96692 kB
Inactive(anon): 0 kB
Active(file):  68100 kB
Inactive(file): 85020 kB
Unevictable:   0 kB
Mlocked:       0 kB
SwapTotal:     0 kB
SwapFree:      0 kB
Dirty:         0 kB
Writeback:     0 kB
AnonPages:     96412 kB
Mapped:        44536 kB
Shmem:         312 kB
Slab:          8940 kB
SReclaimable: 4848 kB
SUnreclaim:   4092 kB
KernelStack:  2528 kB
PageTables:    8324 kB
NFS_Unstable:  0 kB
Bounce:        0 kB
WritebackTmp:  0 kB
CommitLimit:  191672 kB
Committed_AS: 6569284 kB
VmallocTotal: 614400 kB
VmallocUsed:   30656 kB
VmallocChunk: 520068 kB
```

Rooting

nvflash

Press CTRL + ESC + power button to get into APX mode

```
[Thu Oct 11 06:31:57 2018] usb 2-2.4: new high-speed USB device number 13 using xhci_hcd
[Thu Oct 11 06:31:57 2018] usb 2-2.4: New USB device found, idVendor=0955, idProduct=7820, bcdDev=
[Thu Oct 11 06:31:57 2018] usb 2-2.4: New USB device strings: Mfr=1, Product=2, SerialNumber=0
[Thu Oct 11 06:31:57 2018] usb 2-2.4: Product: APX
[Thu Oct 11 06:31:57 2018] usb 2-2.4: Manufacturer: NVIDIA Corp.
```

- <http://tegradeveloper.nvidia.com/tegra-android-development-pack>

In theory. In practice driver package from

- <http://developer.download.nvidia.com/assets/mobile/files/tegra-linux-12.alpha.1.0.tar.gz>

is everything you need. It's from

<http://developer.nvidia.com/content/linux-tegra-release-12-alpha-1-released>

fastboot.stock.bin

```
t61p:/home/dpavlin/Downloads/ldk/bootloader/alt# wget https://github.com/muromec/putusb/raw/master
--2011-09-12 20:40:17-- https://github.com/muromec/putusb/raw/master/bin/fastboot.stock.bin
Resolving github.com (github.com)... 207.97.227.239
Connecting to github.com (github.com)|207.97.227.239|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://raw.githubusercontent.com/muromec/putusb/master/bin/fastboot.stock.bin [following]
--2011-09-12 20:40:19-- https://raw.githubusercontent.com/muromec/putusb/master/bin/fastboot.stock.bin
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 207.97.227.243
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|207.97.227.243|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 936016 (914K) [application/octet-stream]
Saving to: `fastboot.stock.bin'
```

```
100%[=====>] 936,016      645K/s   in 1.4s
```

```
2011-09-12 20:40:22 (645 KB/s) - `fastboot.stock.bin' saved [936016/936016]
```

```
t61p:/home/dpavlin/Downloads/ldk/bootloader/alt# md5sum fastboot.stock.bin
9197365fb42c5801b4124e75ea00512c fastboot.stock.bin
```

backup

Following instructions at <http://gitorious.org/ac100/pages/Installation>

```
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash --bl fastboot.stock.bin --go
Nvflash started
rcm version 0X20001
System Information:
  chip name: t20
  chip id: 0x20 major: 1 minor: 2
  chip sku: 0x8
  chip uid: 0x1700618943a00317
  macrovision: disabled
  hdcp: enabled
  sbk burned: false
  dk burned: false
  boot device: emmc
  operating mode: 3
  device config strap: 0
  device config fuse: 0
  sdram config strap: 0

downloading bootloader -- load address: 0x108000 entry point: 0x108000
sending file: fastboot.stock.bin
| 936016/936016 bytes sent
fastboot.stock.bin sent successfully
waiting for bootloader to initialize
bootloader downloaded successfully
```

```
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --getpartitiontable partitiontable.txt
Nvflash started
[resume mode]
Sucesfully updated partition table information to partitiontable.txt
```

```
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --read 2 part-2.img
Nvflash started
[resume mode]
receiving file: part-2.img, expected size: 3145728 bytes
/ 3145728/3145728 bytes received
file received successfully
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --read 3 part-3.img
Nvflash started
[resume mode]
receiving file: part-3.img, expected size: 524288 bytes
/ 524288/524288 bytes received
file received successfully
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --read 4 part-4.img
bash: ./nvflash: No such file or directory
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --read 4 part-4.img
Nvflash started
[resume mode]
receiving file: part-4.img, expected size: 2097152 bytes
/ 2097152/2097152 bytes received
file received successfully
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --read 5 part-5.img
Nvflash started
[resume mode]
receiving file: part-5.img, expected size: 5242880 bytes
/ 5242880/5242880 bytes received
file received successfully
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --read 6 part-6.img
Nvflash started
[resume mode]
receiving file: part-6.img, expected size: 8388608 bytes
/ 8388608/8388608 bytes received
file received successfully
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --read 7 part-7.img
Nvflash started
[resume mode]
receiving file: part-7.img, expected size: 1048576 bytes
/ 1048576/1048576 bytes received
file received successfully
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --read 8 part-8.img
Nvflash started
[resume mode]
receiving file: part-8.img, expected size: 314572800 bytes
/ 314572800/314572800 bytes received
file received successfully
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --read 9 part-9.img
Nvflash started
[resume mode]
receiving file: part-9.img, expected size: 419430400 bytes
/ 419430400/419430400 bytes received
file received successfully
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --read 10 part-10.img
Nvflash started
[resume mode]
receiving file: part-10.img, expected size: 2097152 bytes
/ 2097152/2097152 bytes received
file received successfully
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --read 11 part-11.img
Nvflash started
[resume mode]
receiving file: part-11.img, expected size: 524288 bytes
/ 524288/524288 bytes received
file received successfully
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --read 12 part-12.img
Nvflash started
[resume mode]
```

```
receiving file: part-12.img, expected size: 1294991360 bytes
/ 1294991360/1294991360 bytes received
file received successfully
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --read 13 part-13.img
Nvflash started
[resume mode]
receiving file: part-13.img, expected size: 524288 bytes
/ 524288/524288 bytes received
file received successfully
t61p:/home/dpavlin/Downloads/ldk/bootloader# ./nvflash -r --read 14 part-14.img
Nvflash started
[resume mode]
receiving file: part-14.img, expected size: 5891424256 bytes
/ 5891424256/5891424256 bytes received
file received successfully
```

Linux

links

fetchrss: <http://feeds.delicious.com/v2/rss/dpavlin/ac100?count=15>

- There was an error: 500 Can't connect to feeds.delicious.com:80 (Bad hostname 'feeds.delicious.com')

phh 2.6.32 kernel with working suspend and audio

- <http://salaliitto.com/~gildean/ac100/wiki/phh/>

```
t61p:/home/dpavlin/ac100/linux4tegra_10.9.3# LD_LIBRARY_PATH=./nvflash/ ./nvflash/nvflash --bl ./
Nvflash started
rcm version 0X20001
System Information:
  chip name: t20
  chip id: 0x20 major: 1 minor: 2
  chip sku: 0x8
  chip uid: 0x1700618943a00317
  macrovision: disabled
  hdcp: enabled
  sbk burned: false
  dk burned: false
  boot device: emmc
  operating mode: 3
  device config strap: 0
  device config fuse: 0
  sdram config strap: 0

downloading bootloader -- load address: 0x108000 entry point: 0x108000
sending file: ./prebuilt/fastboot.stock.bin
| 936016/936016 bytes sent
./prebuilt/fastboot.stock.bin sent successfully
waiting for bootloader to initialize
bootloader downloaded successfully
```

```
receiving file: tegra_partition_6.bin, expected size: 8388608 bytes
/ 8388608/8388608 bytes received
file received successfully
```

```
t61p:/home/dpavlin/ac100/linux4tegra_10.9.3# LD_LIBRARY_PATH=./nvflash/ ./nvflash/nvflash --wait
Nvflash started
rcm version 0X20001
System Information:
  chip name: t20
  chip id: 0x20 major: 1 minor: 2
  chip sku: 0x8
  chip uid: 0x1700618943a00317
  macrovision: disabled
  hdcp: enabled
  sbk burned: false
  dk burned: false
  boot device: emmc
  operating mode: 3
  device config strap: 0
  device config fuse: 0
  sdram config strap: 0
```

```
downloading bootloader -- load address: 0x108000 entry point: 0x108000
sending file: ./prebuilt/fastboot.stock.bin
| 936016/936016 bytes sent
./prebuilt/fastboot.stock.bin sent successfully
waiting for bootloader to initialize
bootloader downloaded successfully
sending file: ../32.boot.menu.v1.img
/ 8388608/8388608 bytes sent
../32.boot.menu.v1.img sent successfully
```

mainline kernel

- Newest howto for arch:
<https://gist.github.com/Nokius/81cd46405b537931a472e0a55abb7ccd>
- <https://archlinuxarm.org/forum/viewtopic.php?f=49&t=10116>

u-boot from APX

This should allow you to send u-boot, kernel and initramfs to any tegra2 board which can be in APX mode without modifying internal storage.

build u-boot

```
dpavlin@klin:/klin/u-boot$ cat env.sh
export CROSS_COMPILE="arm-none-eabi-" ARCH=arm
dpavlin@klin:/klin/u-boot$ . env.sh

dpavlin@klin:/klin/u-boot$ make paz00_defconfig

make
```

build kernel

```
dpavlin@klin:/klin/Tegra/linux$ git remote -v
origin https://github.com/grate-driver/linux (fetch)
origin https://github.com/grate-driver/linux (push)
samsung-tab-10.1 https://github.com/Decatf/linux (fetch)
samsung-tab-10.1 https://github.com/Decatf/linux (push)
```

```
dpavlin@klin:/klin/Tegra/linux$ git branch -vv
* master 63a6317580f8 [origin/master] ARM: tegra: Clear EMC interrupts on resume from LP1 on Tegr
```

```
dpavlin@klin:/klin/Tegra/linux$ cat /klin/u-boot/env.sh
export CROSS_COMPILE="arm-none-eabi-" ARCH=arm
dpavlin@klin:/klin/Tegra/linux$ . /klin/u-boot/env.sh
```

```
dpavlin@klin:/klin/Tegra/linux$ make tegra_defconfig
```

```
make
```

create initrd with new modules

```
dpavlin@klin:/klin/Tegra/linux$ cat tegra-ramdisk.sh
#!/bin/sh -xe
```

```
test -d initrd && rm -Rf initrd
```

```
mkdir initrd
cd initrd
```

```
gzip -cd /tmp/initrd.img | fakeroot -- cpio -i -d -H newc --no-absolute-filenames
```

```
rm -R lib/modules/*
```

```
cd ..
make modules_install INSTALL_MOD_PATH=initrd/ INSTALL_MOD_STRIP=1
```

```
cd initrd
```

```
find . | cpio -o -R 0:0 -H newc > ../initrd.new.cpio
```

```
cd ..
```

```
gzip -f -6 initrd.new.cpio
```

```
/klin/u-boot/tools/mkimage -A arm -O linux -T ramdisk -C gzip -n uInitrd -d initrd.new.cpio.gz uI
```

run it to create initrd:

```
dpavlin@klin:/klin/Tegra/linux$ ./tegra-ramdisk.sh
```

```
Image Name:   uInitrd
Created:      Fri Oct 12 11:20:42 2018
```

```
Image Type:   ARM Linux RAMDisk Image (gzip compressed)
Data Size:    4082917 Bytes = 3987.22 KiB = 3.89 MiB
Load Address: 00000000
Entry Point:  00000000
```

create bootloader file

latest version seems to be: https://gitlab.com/uboot-ac100/create_bootimage.git

```
dpavlin@nuc:/nuc/Tegra/ac100/create_bootimage/example$ ../create_image.pl /mnt/klin/klin/u-boot/u
nt/klin/klin/Tegra/linux/arch/arm/boot/zImage /tmp/postmarketOS-export/initrd.img /mnt/klin/klin/
oot/dts/tegra20-paz00.dtb uboot.scr boot.img
Creating boot image for tegrarcm ...
```

```
./tmp_uboot.scr
image layout:
section  loadaddr      size
=====
uboot    : 0x108000      505627
kernel  : 0x190000      6485800
initrd   : 0x7c0000      4082981
fdt      : 0xbb0000      29659
script  : 0xbc0000      226
```

```
image written to boot.img
```

Now press CTRL + ESC + power on to enter APX mode

```
dpavlin@nuc:/nuc/Tegra/ac100/create_bootimage/example$ sudo apt-get install libcrypto++6

dpavlin@nuc:/nuc/Tegra/ac100/create_bootimage/example$ /mnt/klin/klin/Tegra/tegrarcmsrc/tegrarcmsrc
bct file: paz00-micron-toshiba-8g.bct
bootloader file: boot.img
load addr 0x108000
entry addr 0x108000
device id: 0x7820
uid: 0x1700618943a00317
RCM version: 2.1
downloading miniloader to target at address 0x40008000 (132976 bytes)...
miniloader downloaded successfully
Chip UID: 0x00000000000000001700618943a00317
Chip ID: 0x20
Chip ID Major Version: 0x1
Chip ID Minor Version: 0x2
Chip SKU: 0x8 (t20)
Boot ROM Version: 0x1
Boot Device: 0x2 (EMMC)
Operating Mode: 0x3 (developer mode)
Device Config Strap: 0x0
Device Config Fuse: 0x0
SDRAM Config Strap: 0x0
sending file: paz00-micron-toshiba-8g.bct
- 4080/4080 bytes sent
paz00-micron-toshiba-8g.bct sent successfully
sending file: boot.img
- 11239650/11239650 bytes sent
boot.img sent successfully
```


u-boot loads detects initrd (it's small enough thanks to MODULES_STRIP to be detected without CRC errors) but no output from kernel.

<https://archlinuxarm.org/forum/viewtopic.php?f=23&t=10044&p=51142&hilit=ac100#p51142>

suggests CONFIG_TEGRA_DEBUG_UARTA instead of CONFIG_TEGRA_DEBUG_UART_AUTO_ODMDATA but it doesn't work for me.

```
dpavlin@klin:/klin/Tegra/linux$ grep 'TEGRA.*UART' .config
# CONFIG_TEGRA_DEBUG_UART_AUTO_ODMDATA is not set
CONFIG_TEGRA_DEBUG_UARTA=y
# CONFIG_TEGRA_DEBUG_UARTB is not set
# CONFIG_TEGRA_DEBUG_UARTC is not set
# CONFIG_TEGRA_DEBUG_UARTD is not set
# CONFIG_TEGRA_DEBUG_UARTE is not set
CONFIG_DEBUG_TEGRA_UART=y
```

serial port

https://ac100.grandou.net/serial_port_modification



The serial port is the JP1 connector, near the left speaker connector. Pinout, from right to left:

- 1 - in - RX UART1
- 2 - out - TX UART1
- 3 - out - T20_WAKE#
- 4 - out - +3V
- 5 - out - +1.8V
- 6
- 7
- 8 - in - SYSTEM_RESET#
- 9 - in - EC_TX80_PDATA
- 10 - in - GND

I decided to solder GND to outer pin of connector to the right of pin 1 as opposed to other side of connector.

After that, a piece of capton tape was put over solder connectors to somewhat re-enforce them and provide insulation to keyboard cover.

Serial is 3.3V, 115200 8n1.

Output is:

```
dpavlin@nuc:/nuc/Tegra$ microcom -p /dev/ttyUSB2
connected to /dev/ttyUSB2
Escape character: Ctrl-\
Type the escape character to get to the prompt.
```

```
U-Boot SPL 2018.11-rc1-00130-g0a60a81ba3 (Oct 12 2018 - 11:18:01 +0200)
Trying to boot from RAM
```

```
U-Boot 2018.11-rc1-00130-g0a60a81ba3 (Oct 12 2018 - 11:18:01 +0200)
```

```
TEGRA20
Model: Toshiba AC100 / Dynabook AZ
Board: Compal Paz00
DRAM: 512 MiB
MMC: sdhci@c8000000: 1, sdhci@c8000600: 0
Loading Environment from MMC... *** Warning - bad CRC, using default environment
```

```
In: serial
Out: vidconsole
Err: vidconsole
Net: No ethernet found.
Hit any key to stop autoboot: 0
## Executing script at 00b30000
## Loading init Ramdisk from Legacy Image at 00700000 ...
   Image Name:   uInitrd
   Image Type:   ARM Linux RAMDisk Image (gzip compressed)
   Data Size:    4269789 Bytes = 4.1 MiB
   Load Address: 00000000
   Entry Point:  00000000
   Verifying Checksum ... OK
## Flattened Device Tree blob at 00b20000
   Booting using the fdt blob at 0xb20000
   Using Device Tree in place at 00b20000, end 00b2a3da
```

```
Starting kernel ...
```

```
Error: invalid dtb and unrecognized/unsupported machine ID
  r1=0x00000c38, r2=0x00000000
Available machine support:
```

```
ID (hex)          NAME
ffffffff         Generic DT based system
ffffffff         NVIDIA Tegra SoC (Flattened Device Tree)
```

Please check your kernel config and/or bootloader.

According to <http://billauer.co.il/blog/2014/02/uboot-linux-dtb-fdt-device-tree/> this means that kernel didn't find valid device tree and tries to detect board using machine ID registers.

Solution for this seems to use u-boot variables for loading kernel, ramfs and dtb as opposed to values which script creates, so this works (loading from usb keychain):

```
fatload usb 0:1 ${kernel_addr_r} /boot/zImage
fatload usb 0:1 ${fdt_addr_r} /boot/tegra20-paz00.dtb
fatload usb 0:1 ${ramdisk_addr_r} /boot/uInitrd.new
bootz ${kernel_addr_r} ${ramdisk_addr_r} ${fdt_addr_r}
```

Here are also known working addresses which u-boot uses:

```
Tegra20 (Paz00) MOD # echo ${kernel_addr_r}
0x01000000
Tegra20 (Paz00) MOD # echo ${fdt_addr_r}
0x02000000
Tegra20 (Paz00) MOD # echo ${ramdisk_addr_r}
0x02100000
```

so much higher in memory map than create_image script. However, u-boot seems to want to load at 0x108000 so it's impractical to generate huge image to load kernel, dtb and initramfs at correct addresses.

bootstrap debian

based on <https://wiki.debian.org/InstallingDebianOn/Toshiba/AC100>

```
dpavlin@nuc:/mnt$ sudo debootstrap --foreign --arch=armhf stretch sdd1 http://ftp.hr.debian.org/d
```

```
dpavlin@nuc:/mnt$ df sdd1/
Filesystem      1K-blocks    Used Available Use% Mounted on
/dev/sdd1       1021948 169056    784596  18% /mnt/sdd1
```

```
dpavlin@nuc:/mnt$ sudo cp /usr/bin/qemu-arm-static sdd1/usr/bin
```

```
dpavlin@nuc:/mnt$ sudo chroot sdd1 /debootstrap/debootstrap --second-stage
```

```
dpavlin@nuc:/mnt$ cd sdd1/
dpavlin@nuc:/mnt/sdd1$ sudo mount --bind /sys sys
dpavlin@nuc:/mnt/sdd1$ sudo mount --bind /dev dev
```

```
dpavlin@nuc:/mnt/sdd1$ sudo mount --bind /proc proc
dpavlin@nuc:/mnt/sdd1$ sudo chroot .
```

```
root@nuc:/# apt-get install u-boot-tools
```

```
root@ac100:/boot# cat boot.cmd
```

```
ext2load usb 0:1 ${kernel_addr_r} /boot/zImage
```

```
ext2load usb 0:1 ${fdt_addr_r} /boot/tegra20-paz00.dtb
```

```
ext2load usb 0:1 ${ramdisk_addr_r} /boot/uInitrd
```

```
setenv bootargs root=/dev/sda1 rootwait CMA=64M tegrapart=recovery:300:a00:800,boot:d00:1000:800,
```

```
bootz ${kernel_addr_r} ${ramdisk_addr_r} ${fdt_addr_r}
```

```
# mkimage -C none -A arm -T script -d /boot/boot.cmd /boot/boot.scr
```

u-boot over pax, usb filesystem

```
dpavlin@nuc:/nuc/Tegra/ac100/create_bootimage/example$ /mnt/klin/klin/Tegra/tegarcm/src/tegarcm
bct file: paz00-micron-toshiba-8g.bct
```

Serial output:

```
U-Boot SPL 2018.09-rc2-38340-g3d186cf3f3 (Oct 20 2018 - 10:46:53 +0200)
Trying to boot from RAM
```

```
U-Boot 2018.09-rc2-38340-g3d186cf3f3 (Oct 20 2018 - 10:46:53 +0200)
```

```
TEGRA20
```

```
Model: Toshiba AC100 / Dynabook AZ
```

```
Board: Compal Paz00
```

```
DRAM: 512 MiB
```

```
MMC: sdhci@c8000000: 1, sdhci@c8000600: 0
```

```
Loading Environment from MMC... OK
```

```
In: serial
```

```
Out: vidconsole
```

```
Err: vidconsole
```

```
Net: No ethernet found.
```

```
starting USB...
```

```
USB0: USB EHCI 1.00
```

```
USB1: No code to set up ULPI controller, please enableCONFIG_USB_ULPI and CONFIG_USB_ULPI_VIEWP
```

```
USB2: USB EHCI 1.00
```

```
scanning bus 0 for devices... 1 USB Device(s) found
```

```
scanning bus 2 for devices... 4 USB Device(s) found
```

```
scanning usb for storage devices... 1 Storage Device(s) found
```

```
Hit any key to stop autoboot: 0
```

```
Tegra20 (Paz00) MOD # run usb_boot
```

wifi

```
root@ac100:/etc/network/interfaces.d# apt-get install firmware-ralink wpasupplicant

root@ac100:/etc/network/interfaces.d# cat wlx6c626d1788b3
allow-hotplug wlx6c626d1788b3
iface wlx6c626d1788b3 inet dhcp
    wpa-ssid dreamhouse
    wpa-psk xxxxxxxx
```

disk speed

```
root@ac100:~# hdparm -tT /dev/sda /dev/mmcblk1

/dev/sda:
Timing cached reads:   560 MB in  2.01 seconds = 279.14 MB/sec
Timing buffered disk reads:  32 MB in  3.07 seconds = 10.43 MB/sec

/dev/mmcblk1:
Timing cached reads:   598 MB in  2.00 seconds = 298.86 MB/sec
Timing buffered disk reads: 108 MB in  3.01 seconds = 35.83 MB/sec
```

tuning block size

```
root@ac100:/home/dpavlin/flashbench# ./flashbench -a /dev/mmcblk1 --blocksize=1024
align 2147483648      pre 727Âµs      on 1.07ms      post 784Âµs      diff 316Âµs
align 1073741824     pre 833Âµs      on 1.13ms      post 789Âµs      diff 320Âµs
align 536870912     pre 798Âµs      on 1.13ms      post 825Âµs      diff 317Âµs
align 268435456     pre 816Âµs      on 1.15ms      post 795Âµs      diff 342Âµs
align 134217728     pre 742Âµs      on 1.12ms      post 843Âµs      diff 327Âµs
align 67108864      pre 669Âµs      on 987Âµs      post 701Âµs      diff 303Âµs
align 33554432      pre 662Âµs      on 1.04ms      post 643Âµs      diff 385Âµs
align 16777216      pre 837Âµs      on 1.09ms      post 864Âµs      diff 242Âµs
align 8388608       pre 800Âµs      on 1.1ms       post 881Âµs      diff 262Âµs
align 4194304       pre 812Âµs      on 1.07ms      post 838Âµs      diff 246Âµs
align 2097152       pre 807Âµs      on 1.07ms      post 808Âµs      diff 258Âµs
align 1048576       pre 832Âµs      on 960Âµs      post 838Âµs      diff 125Âµs
align 524288        pre 833Âµs      on 959Âµs      post 838Âµs      diff 124Âµs
align 262144        pre 833Âµs      on 959Âµs      post 838Âµs      diff 124Âµs
align 131072        pre 833Âµs      on 962Âµs      post 840Âµs      diff 126Âµs
align 65536         pre 825Âµs      on 951Âµs      post 838Âµs      diff 120Âµs
align 32768         pre 826Âµs      on 956Âµs      post 825Âµs      diff 130Âµs
align 16384         pre 812Âµs      on 924Âµs      post 824Âµs      diff 106Âµs
align 8192          pre 809Âµs      on 841Âµs      post 808Âµs      diff 32.4Âµs
align 4096          pre 808Âµs      on 836Âµs      post 808Âµs      diff 27.7Âµs
align 2048          pre 808Âµs      on 836Âµs      post 809Âµs      diff 27.6Âµs
```

8192 seems good, re-test

```
root@ac100:/home/dpavlin/flashbench# ./flashbench -a /dev/mmcblk1 --blocksize=4096
align 2147483648      pre 696Âµs      on 1.04ms      post 725Âµs      diff 331Âµs
align 1073741824     pre 917Âµs      on 1.18ms      post 870Âµs      diff 289Âµs
align 536870912     pre 880Âµs      on 1.18ms      post 907Âµs      diff 291Âµs
```

align 268435456	pre 899Âµs	on 1.2ms	post 877Âµs	diff 311Âµs
align 134217728	pre 825Âµs	on 1.18ms	post 927Âµs	diff 300Âµs
align 67108864	pre 864Âµs	on 1.11ms	post 842Âµs	diff 253Âµs
align 33554432	pre 914Âµs	on 1.12ms	post 897Âµs	diff 212Âµs
align 16777216	pre 921Âµs	on 1.14ms	post 948Âµs	diff 210Âµs
align 8388608	pre 884Âµs	on 1.15ms	post 965Âµs	diff 229Âµs
align 4194304	pre 895Âµs	on 1.12ms	post 922Âµs	diff 215Âµs
align 2097152	pre 890Âµs	on 1.12ms	post 890Âµs	diff 227Âµs
align 1048576	pre 916Âµs	on 1.01ms	post 922Âµs	diff 89.7Âµs
align 524288	pre 916Âµs	on 1.01ms	post 922Âµs	diff 90Âµs
align 262144	pre 916Âµs	on 1.01ms	post 923Âµs	diff 89.1Âµs
align 131072	pre 916Âµs	on 1.01ms	post 924Âµs	diff 91.1Âµs
align 65536	pre 908Âµs	on 1ms	post 921Âµs	diff 85.9Âµs
align 32768	pre 911Âµs	on 1.01ms	post 908Âµs	diff 96.2Âµs
align 16384	pre 895Âµs	on 973Âµs	post 908Âµs	diff 71.4Âµs
align 8192	pre 892Âµs	on 896Âµs	post 892Âµs	diff 4.57Âµs

create filesystem

```

root@ac100:/home/dpavlin/flashbench# mkfs -t ext4 -v -b 4096 -E stride=2,stripe-width=2 /dev/mmcblk1p1
cp -ax ....

root@ac100:~# mount /dev/mmcblk1p1 /tmp/1/
root@ac100:~# cat /tmp/1/boot/boot.cmd
ext2load mmc 0:6 ${kernel_addr_r} /boot/zImage
ext2load mmc 0:6 ${fdt_addr_r} /boot/tegra20-paz00.dtb
ext2load mmc 0:6 ${ramdisk_addr_r} /boot/uInitrd
setenv bootargs root=/dev/mmcblk1p6 rootwait CMA=64M tegrapart=recovery:300:a00:800,boot:d00:1000
bootz ${kernel_addr_r} ${ramdisk_addr_r} ${fdt_addr_r}

# mkimage -C none -A arm -T script -d boot.cmd boot.scr

```

flash u-boot to mmc

based on https://paz00.ru/index.php?title=Flashing_Uboot_to_MMC&setlang=en

tegrarc to get bct

```

dpavlin@nuc:/nuc/Tegra$ git clone https://github.com/NVIDIA/tegrarc
dpavlin@nuc:/nuc/Tegra$ cd tegrarc/
dpavlin@nuc:/nuc/Tegra/tegrarc$ sudo apt-get install libcrypto++-dev
dpavlin@nuc:/nuc/Tegra/tegrarc$ ./autogen.sh
dpavlin@nuc:/nuc/Tegra/tegrarc$ make
dpavlin@nuc:/nuc/Tegra/tegrarc$ ./src/tegrarc --version
tegrarc 1.8

```

create boot image

```
dpavlin@nuc:/nuc/Tegra/cbootimage$ git remote -v
origin https://github.com/NVIDIA/cbootimage (fetch)
origin https://github.com/NVIDIA/cbootimage (push)
```

```
dpavlin@nuc:/nuc/Tegra/cbootimage$ ./autogen.sh
dpavlin@nuc:/nuc/Tegra/cbootimage$ make
```

```
dpavlin@nuc:/nuc/Tegra/cbootimage$ cat > u-boot.cfg
Version      = 0x00020001;
Bctcopy      = 1;
Bctfile      = ac100.bct;
BootLoader   = u-boot.bin,0x00108000,0x00108000,Complete;
```

```
dpavlin@nuc:/nuc/Tegra/cbootimage$ cp ../ac100/create_bootimage/example/ac100-dpavlin.bct ac100.b
```

```
dpavlin@nuc:/nuc/Tegra/cbootimage$ cp /mnt/klin/klin/Tegra/u-boot-tegra/u-boot-tegra.bin u-boot.b
```

```
dpavlin@nuc:/nuc/Tegra/cbootimage$ ./src/cbootimage -d u-boot.cfg ac100.bct.new
```

```
bct size: 4080
```

```
**update_bl()
```

```
begin_update(): bct data: b=14 p=9
```

```
writing bootloader
```

```
  redundancy = 1
```

```
  BL[0]: 131073 0001 0000 518563 0x00108000 0x00108000
8e52c9639cb433e9e1e369cf718595c5
```

```
  BL[1]: 131073 0002 0000 302746 0x00108000 0x00108000
a9014f31e7ccafab319512de6bc48f81
```

```
**BL[2]: 0 0000 0000 0000 0x00000000 0x00000000
00000000000000000000000000000000
```

```
**BL[3]: 0 0000 0000 0000 0x00000000 0x00000000
00000000000000000000000000000000
```

```
Image file ac100.bct.new has been successfully generated!
```

```
dpavlin@nuc:/nuc/Tegra/cbootimage$ scp ac100.bct.new ac100:
```

```
dpavlin@nuc:/nuc/Tegra/cbootimage$ ssh root@ac100
```

```
# flash new boot image on ac100
```

```
root@ac100:~# echo 0 > /sys/block/mmcblk1boot0/force_ro
```

```
root@ac100:~# dd if=/home/dpavlin/ac100.bct.new of=/dev/mmcblk1boot0
```

```
1045+0 records in
```

```
1045+0 records out
```

```
535040 bytes (535 kB, 522 KiB) copied, 0.135589 s, 3.9 MB/s
```

```
root@ac100:~# echo 1 > /sys/block/mmcblk1boot0/force_ro
```

create bct backup

```
dpavlin@ac100:~$ sudo dd if=/dev/mmcblk1boot0 of=ac100-dpavlin-0.bct bs=4080 count=1
```

```
[sudo] password for dpavlin:
```

```
1+0 records in
```

```
1+0 records out
```

```
4080 bytes (4.1 kB, 4.0 KiB) copied, 0.0018642 s, 2.2 MB/s
```

```
dpavlin@ac100:~$ sudo dd if=/dev/mmcblk1boot1 of=ac100-dpavlin-1.bct bs=4080 count=1
```

```
1+0 records in
```

```
1+0 records out
4080 bytes (4.1 kB, 4.0 KiB) copied, 0.0019622 s, 2.1 MB/s
```

in my case, first bct doesn't have OdmData, while 2nd does:

```
dpavlin@nuc:/nuc/Tegra$ grep Odm 0 1
0:OdmData          = 0x00000000;
1:OdmData          = 0x800c0075;
```

i2c

```
root@ac100:~# i2cdetect -y -r 0
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  UU  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  4f
50: 50 51 52 53 54 55 56 57 -- -- -- -- -- -- -- -- -- -- --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
root@ac100:~# i2cdetect -y -r 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
root@ac100:~# i2cdetect -y -r 2
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  UU  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  UU  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```