

Contents: [Dobrica PavlinuÅiÄ 's random unstructured stuff]

- [Dobrica PavlinuÅiÄ 's random unstructured stuff \(u-boot\)](#)
 - ◆ [Dobrica PavlinuÅiÄ 's random unstructured stuff \(Info\)](#)
- [Dobrica PavlinuÅiÄ 's random unstructured stuff \(dfu-util\)](#)
 - ◆ [Dobrica PavlinuÅiÄ 's random unstructured stuff \(flash kernel\)](#)
 - ◆ [Dobrica PavlinuÅiÄ 's random unstructured stuff \(flash rootfs\)](#)
- [Dobrica PavlinuÅiÄ 's random unstructured stuff \(openocd\)](#)
- [Dobrica PavlinuÅiÄ 's random unstructured stuff \(Android\)](#)
- [Dobrica PavlinuÅiÄ 's random unstructured stuff \(Open Issues\)](#)
- [Dobrica PavlinuÅiÄ 's random unstructured stuff \(Pending issues\)](#)
- [Dobrica PavlinuÅiÄ 's random unstructured stuff \(Resolved issues\)](#)
- [Dobrica PavlinuÅiÄ 's random unstructured stuff \(Qemu\)](#)
- [Dobrica PavlinuÅiÄ 's random unstructured stuff \(See also\)](#)
 - ◆ [Dobrica PavlinuÅiÄ 's random unstructured stuff \(Other pages on this wiki about OpenMoko\)](#)
 - ◆ [Dobrica PavlinuÅiÄ 's random unstructured stuff \(My blog posts about OpenMoko\)](#)

u-boot

- <http://wiki.openmoko.org/wiki/NeoCon>
- http://wiki.openmoko.org/wiki/NAND_bad_blocks

Recover from bad blocks procedure

```
x200:/virtual/openmoko# ./neocon/neocon /dev/ttyACM*
[Open /dev/ttyACM3]
In:   usbttty
Out:  usbttty
Err:  usbttty
DEVICE_CONFIGURED: 1
Enabling automatic fast charge
```

```
GTA01Bv4 # nand bad
```

```
Device 0 bad blocks:
```

```
00070000
00ab0000
00f00000
03ff0000
03ff4000
03ff8000
03ffc000
```

```
GTA01Bv4 # nand scrub
```

```
NAND scrub: device 0 whole chip
Warning: scrub option will erase all factory set bad blocks!
         There is no reliable way to recover them.
         Use this command only for testing purposes if you
         are sure of what you are doing!
```

```
Really scrub this NAND flash? <y/N>
```

```
NAND 64MiB 3,3V 8-bit: MTD Erase failure: -5
```

```
NAND 64MiB 3,3V 8-bit: MTD Erase failure: -5
```

```
NAND 64MiB 3,3V 8-bit: MTD Erase failure: -5  
Bad block table not found for chip 0  
Bad block table not found for chip 0  
OK
```

```
GTA01Bv4 # nand createbbt  
Create BBT and erase everything ? <y/N>  
Skipping bad block at 0x00070000  
Skipping bad block at 0x00ab0000  
Skipping bad block at 0x00f00000  
Skipping bad block at 0x03ff0000  
Skipping bad block at 0x03ff4000  
Skipping bad block at 0x03ff8000  
Skipping bad block at 0x03ffc000
```

```
Creating BBT. Please wait ...Bad block table not found for chip 0  
Bad block table not found for chip 0  
Bad block table written to 0x03ffc000, version 0x01  
Bad block table written to 0x03ff8000, version 0x01
```

```
GTA01Bv4 # dynpart  
mtdparts mtdparts=neo1973-nand:0x00040000(u-boot),0x00004000(u-boot_env),0x00204000(kernel),0x000
```

```
GTA01Bv4 # dynenv set u-boot_env  
device 0 offset 0x40000, size 0x4000  
45 4e 56 30 - 00 00 04 00
```

```
GTA01Bv4 # saveenv  
Saving Environment to NAND...  
Saving ESaving EnvironmnvironmWriting to Nand... done
```

Info

```
GTA01Bv4 # bdfinfo  
arch_number = 0x0000049E  
env_t       = 0x00000000  
boot_params = 0x30000100  
DRAM bank   = 0x00000000  
-> start     = 0x30000000  
-> size      = 0x08000000  
ethaddr     = 00:00:00:00:00:00
```

```
ip_addr     = 0.0.0.0  
baudrate    = 115200 bps
```

```
GTA01Bv4 # printenv  
baudrate=115200  
bootcmd=setenv bootargs ${bootargs_base} ${mtdparts}; nand read.e 0x32000000 kernel; bootm 0x3200  
bootdelay=1  
menu_1=Set console to USB: setenv stdin usbtty; setenv stdout usbtty; setenv stderr usbtty  
menu_2=Set console to serial: setenv stdin serial; setenv stdout serial; setenv stderr serial  
menu_3=Power off: neo1973 power-off  
menu_4=Boot from SD: setenv bootargs root=/dev/mmcblk0p1 rootdelay=5 console=ttySAC0,115200 cons  
mtdids=nand0=neo1973-nand  
quiet=1  
sd_image_name=uImage
```

```
splashimage=nand read.e 0x32000000 splash 0x5000; unzip 0x32000000 0x33d00000 0x96000
stderr=usbtty
stdin=usbtty
stdout=usbtty
usbtty=cdc_acm
bootargs_base=rootfstype=jffs2 root=/dev/mtdblock4 console=tty0 loglevel=8
mtdparts=mtdparts=neo1973-nand:0x00040000 (u-boot),0x00040000 (u-boot_env),0x00204000 (kernel),0x000
mtdparts=mtdparts=nmtdddevnum=0
mtddevname=u-boot
pcf50606_int1=0x4a
```

Environment size: 1028/16380 bytes

GTA01Bv4 # mtd

device nand0 <neo1973-nand>, # parts = 5

#:	name	size	offset	mask_flags
0:	u-boot	0x00040000	0x00000000	0
1:	u-boot_env	0x00040000	0x00040000	0
2:	kernel	0x00204000	0x00044000	0
3:	splash	0x000a0000	0x00248000	0
4:	rootfs	0x03d18000	0x002e8000	0

active partition: nand0,0 - (u-boot) 0x00040000 @ 0x00000000

defaults:

mtdids : nand0=neo1973-nand

mtdparts: <NULL>

dfu-util

Disable bluetooth which also responds to dfu:

```
x200:/home/dpavlin# echo disable > /proc/acpi/ibm/bluetooth
```

- http://wiki.openmoko.org/wiki/Flashing_the_Neo_1973

Boot moko by pressing AUX (top-left smooth button) and power (on the right below usb cable)

```
x200:/virtual/openmoko# dfu-util --list
```

```
dfu-util - (C) 2007-2008 by OpenMoko Inc.
```

```
This program is Free Software and has ABSOLUTELY NO WARRANTY
```

```
dfu-util does currently only support DFU version 1.0
```

```
Found Runtime: [0x1457:0x5119] devnum=38, cfg=0, intf=2, alt=0, name="USB Device Firmware Upgrade"
```

Hm. We need to reset device in DFU mode

```
x200:/virtual/openmoko# dfu-util --device 0x1457:0x5119 --download xx
```

```
dfu-util - (C) 2007-2008 by OpenMoko Inc.
```

```
This program is Free Software and has ABSOLUTELY NO WARRANTY
```

dfu-util does currently only support DFU version 1.0

```
Opening USB Device 0x1457:0x5119...
Claiming USB DFU Runtime Interface...
Determining device status: state = appIDLE, status = 0
Device really in Runtime Mode, send DFU detach request...
Resetting USB...
Opening USB Device...
Found Runtime: [0x1457:0x5119] devnum=39, cfg=0, intf=0, alt=0, name="RAM 0x32000000"
We have 6 DFU Interfaces/Altsettings, you have to specify one via --intf / --alt options
```

```
x200:/virtual/openmoko# dfu-util --device 0x1457:0x5119 --list
dfu-util - (C) 2007-2008 by OpenMoko Inc.
This program is Free Software and has ABSOLUTELY NO WARRANTY
```

dfu-util does currently only support DFU version 1.0

```
Found DFU: [0x1457:0x5119] devnum=39, cfg=0, intf=0, alt=0, name="RAM 0x32000000"
Found DFU: [0x1457:0x5119] devnum=39, cfg=0, intf=0, alt=1, name="u-boot"
Found DFU: [0x1457:0x5119] devnum=39, cfg=0, intf=0, alt=2, name="u-boot_env"
Found DFU: [0x1457:0x5119] devnum=39, cfg=0, intf=0, alt=3, name="kernel"
Found DFU: [0x1457:0x5119] devnum=39, cfg=0, intf=0, alt=4, name="splash"
Found DFU: [0x1457:0x5119] devnum=39, cfg=0, intf=0, alt=5, name="rootfs"
```

flash kernel

```
x200:/virtual/openmoko/fso# dfu-util -a kernel -R -D uImage-2.6.24-oe5+gitrfb42ce6724576fc173faf8
dfu-util- (C) 2007-2008 by OpenMoko Inc.
This program is Free Software and has ABSOLUTELY NO WARRANTY
```

dfu-util does currently only support DFU version 1.0

```
Opening USB Device 0x0000:0x0000...
Found Runtime: [0x1457:0x5119] devnum=39, cfg=0, intf=0, alt=3, name="kernel"
Claiming USB DFU Interface...
Setting Alternate Setting #3 ...
Determining device status: state = dfuIDLE, status = 0
dfuIDLE, continuing
Transfer Size = 0x1000
bytes_per_hash=38298
Starting download: [#####] finished!
state(2) = dfuIDLE, status(0) = No error condition is present
Done!
Resetting USB to switch back to runtime mode
```

flash rootfs

```
x200:/virtual/openmoko/fso# dfu-util -a rootfs -R -D openmoko-fso-console-image-glibc-ipk--200902
dfu-util- (C) 2007-2008 by OpenMoko Inc.
This program is Free Software and has ABSOLUTELY NO WARRANTY
```

dfu-util does currently only support DFU version 1.0

```
Opening USB Device 0x0000:0x0000...
Found Runtime: [0x1457:0x5119] devnum=41, cfg=0, intf=0, alt=5, name="rootfs"
Claiming USB DFU Interface...
```

```
Setting Alternate Setting #5 ...
Determining device status: state = dfuERROR, status = 14
dfuERROR, clearing status
Determining device status: state = dfuIDLE, status = 0
dfuIDLE, continuing
Transfer Size = 0x1000
bytes_per_hash=452853
Starting download: [#####dfu_download: usb_control_m
unable to read DFU status
```

No worky

openocd

- [openocd.cfg](#)
- http://wiki.openmoko.org/wiki/Neo1973_OpenOCD

```
x200:/virtual/openmoko# openocd
Open On-Chip Debugger 0.4.0 (2010-02-23-17:04)
Licensed under GNU GPL v2
For bug reports, read
    http://openocd.berlios.de/doc/doxygen/bugs.html
trst_and_srst separate srst_gates_jtag trst_push_pull srst_open_drain
trst_and_srst separate srst_gates_jtag trst_push_pull srst_open_drain
Info : clock speed 6000 kHz
Info : JTAG tap: s3c2440.cpu tap/device found: 0x0032409d (mfg: 0x04e, part: 0x0324, ver: 0x0)
Info : Embedded ICE version 2
Info : s3c2440.cpu: hardware has 2 breakpoint/watchpoint units
Info : JTAG tap: s3c2440.cpu tap/device found: 0x0032409d (mfg: 0x04e, part: 0x0324, ver: 0x0)
Warn : NOTE! DCC downloads have not been enabled, defaulting to slow memory writes. Type 'help do
Warn : NOTE! Severe performance degradation without fast memory access enabled. Type 'help fast'.
target state: halted
target halted in ARM state due to debug-request, current mode: Supervisor
cpsr: 0x20000053 pc: 0x33f966d8
MMU: disabled, D-Cache: enabled, I-Cache: enabled
Info : accepting 'telnet' connection from 0
Runtime error, file "command.c", line 650:
    invalid command name "run"
Info : JTAG tap: s3c2440.cpu tap/device found: 0x0032409d (mfg: 0x04e, part: 0x0324, ver: 0x0)
Warn : NOTE! DCC downloads have not been enabled, defaulting to slow memory writes. Type 'help do
Warn : NOTE! Severe performance degradation without fast memory access enabled. Type 'help fast'.
```

Scratch pad for my notes about OpenMoko

Android

- <http://onlinedev.blogspot.com/2008/11/porting-android-phase-4-done.html>

```
dpavlin@llin:/rest/cvs/qemu-neo1973/openmoko/android$ wget -r -np -m http://people.openmoko.org/s
```

Open Issues

- how to start [matchbox-stroke](#) ?
- [gsmd problem](#)

Pending issues

- [gps driver](#)
- using keyboard without mouse -- [Finger Keyboard](#)

Resolved issues

- correct time -- `ipkg install ntpdate`

Qemu

Alternative way to build qemu with Neo 1973 emulation:

```
# checkout source

svn co https://svn.openmoko.org/trunk/src/host/qemu-neo1973
cd qemu-neo1973

# run configure

./configure --target-list=arm-softmmu --enable-gadgetfs

# needed for gadgetfs to work

ln -s /usr/src/linux/include/linux

# build it

make
```

See also

Other pages on this wiki about OpenMoko

Found 30 results

[Search for openmoko](#)

- [Ubertooh One](#)
- [Android G1](#)
- [OpenMoko](#)
- [OpenMoko as a phone](#)
- [OpenMoko arrived](#)

My blog posts about OpenMoko

Search Results for openmoko

- [OpenMoko GPS tracking](#)
- [OpenMoko flashing](#)
- [OpenMoko as a phone](#)
- [OpenMoko arrived](#)
- [OpenMoko paid](#)
- [OpenMoko and why I ordered one](#)

.pre