

Contents: [Dobrica Pavlinu's random unstructured stuff]

- Dobrica Pavlinu's random unstructured stuff (JTAG)
- Dobrica Pavlinu's random unstructured stuff (openocd)
  - ◆ Dobrica Pavlinu's random unstructured stuff (compilation)
- Dobrica Pavlinu's random unstructured stuff (info)
  - ◆ Dobrica Pavlinu's random unstructured stuff (flash)
- Dobrica Pavlinu's random unstructured stuff (Debugging)
  - ◆ Dobrica Pavlinu's random unstructured stuff (blue-light mode)
  - ◆ Dobrica Pavlinu's random unstructured stuff (v-ref = 2.6v)
  - ◆ Dobrica Pavlinu's random unstructured stuff (openocd, trst-n = 2.6v)
  - ◆ Dobrica Pavlinu's random unstructured stuff (soldering)
  - ◆ Dobrica Pavlinu's random unstructured stuff (other)
- Dobrica Pavlinu's random unstructured stuff (softload radio ROM)

related pages: [Android G1](#) for JTAG info and [Android development](#)

## JTAG

- <http://www.inaccessnetworks.com/projects/ianjtag/jtag-intro/jtag-intro.html>
- <http://www.freelabs.com/~whitis/electronics/jtag/>

## openocd

### compilation

```
dpavlin@x200:/rest/cvs/openocd$ git remote -v
origin git://openocd.git.sourceforge.net/gitroot/openocd/openocd (fetch)
origin git://openocd.git.sourceforge.net/gitroot/openocd/openocd (push)
dpavlin@x200:/rest/cvs/openocd$ ./configure --enable-verbose --enable-verbose-usb-io --enable-ft2
dpavlin@x200:/rest/cvs/openocd$ make
```

## info

## flash

<http://forum.xda-developers.com/showpost.php?p=6240836&postcount=503>

If you have raw access to flash load a SPL+Recovery compatible with your radio

SPL starts at: 0x02400000 (block 288) hboot.img  
Recovery starts at: 0x26c0000 (block 310) recovery.img

2005 SPL:

```
Tidus:spl ezterry$ ../fastboot-mac oem listpartition
... INFO[radio] start block=0, size=287 (36736 KB)
INFO[hboot] start block=288, size=6 (768 KB)
INFO[misc] start block=294, size=2 (256 KB)
INFO[mfg] start block=296, size=2 (256 KB)
INFO[spl] start block=298, size=6 (768 KB)
INFO[misc2] start block=304, size=3 (384 KB)
INFO[mfg2] start block=307, size=3 (384 KB)
INFO[recovery] start block=310, size=40 (5120 KB)
INFO[boot] start block=350, size=20 (2560 KB)
INFO[system] start block=370, size=720 (92160 KB)
INFO[cache] start block=1090, size=240 (30720 KB)
INFO[userdata] start block=1330, size=718 (91904 KB)
INFO[cpld] start block=0, size=0 (0 KB)
INFO[microp] start block=0, size=0 (0 KB)
OKAY
```

## Debugging

<http://forum.xda-developers.com/showpost.php?p=6498820&postcount=621>

That said before doing anything else take out your multi meter (and if you don't have one you are missing a tool for this type of work) and check the following:

### blue-light mode

1) Put phone into blue light mode if serial is attached and power isn't: you will see bootmode 1

```
dpavlin@x200:/virtual/android$ ./neocon /dev/ttyUSB0
[Closed]
[Open /dev/ttyUSB0]

boot reason: PM_KPD_PWR_KEY_ON_RT_ST

(PowerOn Status,Boot Reason)=(1,1)
NAND_FLASH_READ_ID : SAMSUNG_256MB_FLASH_128MB_SDRAM

ARM9_BOOT_MODE1
Invalid command : ?
```

### v-ref = 2.6v

2) verify your v-ref is actually 2.6v (usually within 0.05v) when compared to the ground (any of the shielding) of the main board. You have one of the right points so there is an issue with the connection if its not.

## opnecd, trst-n = 2.6v

```
dpavlin@x200:/virtual/android/HTC-Dream-G1-JTAG$ sudo opnecd
Open On-Chip Debugger 0.4.0 (2010-02-23-17:04)
Licensed under GNU GPL v2
For bug reports, read
    http://opnecd.berlios.de/doc/doxygen/bugs.html
trst_and_srst separate srst_gates_jtag trst_push_pull srst_open_drain
dcc downloads are enabled
fast memory access is enabled
Info : clock speed 6000 kHz
Info : JTAG tap: arm9.cpu tap/device found: 0xa01700e1 (mfg: 0x070, part: 0x0170, ver: 0xa)
Info : Embedded ICE version 6
Info : arm9: hardware has 2 breakpoint/watchpoint units
```

3) now start open ocd: and check that trst-n is now also 2.6v when compared to ground. (if not your adaptor is not working with the 2.6v.. very possible)

nTRST must be at high level (~2.6V) after opnecd is launched.  
With nTRST sticking low the MSM7201A debug unit is in reset state.

Maybe you'll have to tweak the cfg file. There are several options for the behaviour of nTRST.  
If you don't manage to find a working cfg, you may cut the nTRST connection to your adaptor and pull the Dream's nTRST signal to Vref=2.6V permanently.  
Normally this should also work... the debug unit will then leave reset state immediately after power up.

Maybe you'll have to tweak the cfg file. There are several options for the behaviour of nTRST.  
If you don't manage to find a working cfg, you may cut the nTRST connection to your adaptor and pull the Dream's nTRST signal to Vref=2.6V permanently.  
Normally this should also work... the debug unit will then leave reset state immediately after power up.

## soldering

4) with that done and you are still having problems as we said before check the soldering work..  
Here there are two possibilities:

- A) bridges - the wire is in contact with something in addition to the testpoint
- B) bad joints - while the wire may act attached it is not. (Others must be able to explain this better than myself..)

The tiniest bit of flux goes a long way here..

One quick thing you can test is that none of the 5 test points are connected to ground (disconnect rtkc for now its one less variable) and that none are shorted to eachother.

## other

Other things to check:

- How long are the wires.. My setup runs much faster and the wires are not exactly short .. but the more wire the more chance for noise..

