

Phone Does Not Boot

Well well well. You've bricked the phone.
Well done.

This can happen when you become a little too curious about what the `fmt` commands do.

All is not lost, you can most likely recover the phone with a bit of effort.

Requirements

- Access to the phone's UART debugging interface
- A `tftp` server
 - I used the inbuilt one with `truenas`.
- A copy of some firmware. (See `JackGit`)

Steps

- Configure your tftp server and place the `.bin` file from the firmware zip file inside the root directory.
- Access the phone's UART debug interface
- Power on the phone.
 - If the phone cannot boot it will enter into the bootloader (`PSPBoot`) shell, `psbl`.
 - You should see something similar to this.
 - *NOTE: If you mistype a command the shell will hang and the device will need to be power cycled. This gets annoying fast. Good luck!*

Basic POST completed... Success.

Last reset cause: Hardware reset (Power-on reset)

PSPBoot1.4 rev: 1.4.0.6

(c) Copyright 2002-2008 Texas Instruments, Inc. All Rights Reserved.

Press ESC for monitor... 1

(psbl)

- Next set the following environment variables:
 - The static IP Address of the phone:
 - `setenv IPA 192.168.1.200`
 - The subnet mask:
 - `setenv SUBNET_MASK 255.255.255.0`
 - The MAC address:
 - `setenv HWA_0 aa:aa:aa:aa:aa:aa`
 - The MAC port:
 - `setenv MAC_PORT 0`
 - I'm not 100% sure you need to do this but I haven't checked.
- You should be ready to upgrade the firmware.
 - `upgrade -i <TFTP_SERVER_IP> spa50x-30x-7-4-6.bin`
- If successful, you'll see the new firmware be written to memory.

```
(psbl) upgrade -i 192.168.1.5 fw.bin
Validate firmware successful
Cannot upgrade bootloader in bootloader/recovery
Cannot upgrade bootloader in bootloader/recovery
Programming sector:3
Programming sector:4
Programming sector:66
Programming sector:67
Programming sector:68
Programming sector:69
Programming sector:70
Programming sector:71
Programming sector:72
...
Programming sector:125
Programming sector:126
Programming sector:127
```

- Finally, reboot the phone by either:
 - Power cycle
 - `reboot`

You will know the process will have worked when you see something like this:

```
Booting...
Attached TCP/IP interface to dummy unit 254
Attaching network interface lo0... done.

Adding 8763 symbols for standalone.
```

CPU: TI TNETV1057 Communication Processor. Processor #0.

Memory Size: 0xffe000. BSP version 7.2.7.20.

```
=====
Board      : TI TNETV1057 Communication Processor
SOC        : Titan, ChipId: 0x7, Version: 2
Cache      : Write-Back, Write-Allocate
PSP Version : 7.2.7.20
Type       : BasePSP 7.2.7.20 Patch
PSPWIZ Version : 0.5
MIPS freq  : 87500000 Hz,
System Freq : 87500000-> Hz,
VBUS freq   : 81250000 Hz
BasePSP mode : Routing
=====
```

Model no: 2

appCreate: autoBootLevel=2

MXP environment is created.

About to create Idle Task

About to create Measurement Task

Idle Measurement Tasks created

Panic button enabled

Heartbeat started

Creating Golden Gateway application...

Creating fs:/tmp 3145728

Decompress app module.... done

appmodule len=2642000

Creating fs:/DR 16384

/DR created

decompress constdat successfully:520624

flash_init . . .

-- flash_raw_init . . .

-- flash_fstr_init . . .

-- flash_fsm_init . . .

-- flash_license_init . . .

-- flash_fpar_init . . .

-- flash_custom_init . . .

-- flash_fprv_init . . .

-- flash_dhcp_prov_init . . .

flash_init done

Revision #2

Created 28 March 2022 22:18:35 by Peter

Updated 28 March 2022 22:20:08 by Peter