

Apollo - Recovery von Kernel und Image via USB

- Dateien aus dem Anhang laden (Achtung, es gibt verschiedene Pakete für Trinity und Apollo!)
- USB Stick mit FAT32 formatieren
- "**rootfs.arm.jffs2.nand**" (Image) und "**vmlinux.ub.gz**" (Kernel) auf USB Stick kopieren
- USB Stick in den USB Steckplatz eurer Box stecken
- Box ausschalten (Netzschalter) und kurz danach wieder einschalten
- Gleich nach dem einschalten die Powertaste der Fernbedienung mehrfach drücken...

Jetzt sollte der Updateprozess starten und die Box nach einiger Zeit von selbst mit dem neu geflashten Kernel und Image neu starten. Ihr könnt natürlich auch nur den Kernel oder nur das Image aktualisieren. 😊

Die Updatemethode via USB Stick unterstützt folgende Files auf dem USB Stick:

- uldr.bin
- uboot.bin
- **vmlinux.ub.gz (Kernel)**
- **rootfs.arm.jffs2.nand (Image)**
- appfs.jffs2.nand
- rootfs1.arm.jffs2.nand
- appfs1.jffs2.nand
- extra.nand
- var.arm.jffs2.nand (nur mit neuen uboot ab 2014)

Ab dem nächsten Image in 2014 wird ein neues UBOOT mit ausgeliefert.

Dieses wird um folgendes Feature ergänzt sein: Support von "var.arm.jffs2.nand"

Befindet sich auf dem Stick eine leere Datei "var.arm.jffs2.nand" wird die ab dem nächsten Image eingeführte persistente var-Partition resettet.

Ist "var.arm.jffs2.nand" eine echte Sicherung wird sie auch wie Kernel oder Image ganz normal geflasht.

Wenn die obige Methode nicht greift, könnte Bootloader oder Uboot in der Box fehlerhaft sein.

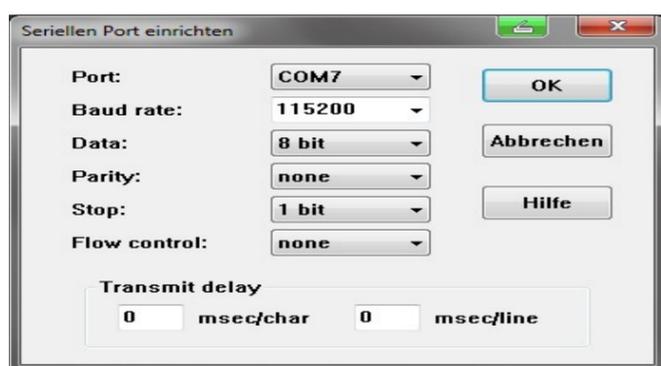
Anleitung um Apollo Boards (Tank, Trinity) per UART und USB wieder herzustellen:

**Dieses HOWTO ist nur für äußerste Notfälle und sollte Normalerweise nicht benötigt werden!
Es muss zwingend die im Thread angehängte uldr.zip (uldr.bin.uboot_img) verwendet werden!**

Vorbereitung:

- Tera Term Download: <http://en.sourceforge.jp/projects/ttssh2/releases/>

- In Tera Term müssen die Einstellungen der seriellen Schnittstelle (Setup -> Serial port) folgendermaßen gesetzt sein:



Schritt 1:

- Auf einem FAT32 formatierten USB Stick kopiert man folgende Files ins Hauptverzeichnis:
- uldr.bin, u-boot.bin, (rootfs.arm.jffs2.nand, appfs.jffs2.nand and vmlinux.ub.gz)

Schritt 2:

- Gerät ausschalten und öffnen.
- Jumper JP1 auf dem Mainboard stecken.
- Box und PC mit Nullmodem Kabel verbinden (dabei sollten beide Geräte ausgeschaltet sein !!!)

Schritt 3:

- Benutze ein serielles Terminal Programm (Tera Term oder minicom) welches das YModem Protokoll versteht.
- Gerät einschalten.
- Das Terminal sollte nun zeigt ein 'C' Zeichen zeigen. Das bedeutet, das die Box bereit ist zum laden von bootloader und u-boot.
- Sende nun vom Terminal "uldr.bin.uboot_img" zur Box.
- Tera Term: File -> Transfer -> YMODEM -> send
- Danach siehst du, wie nach u-boot gefragt wird und es wird wieder das 'C' Zeichen gezeigt.
- Sende die Datei "u-boot.bin" und drücke schnell "Enter" nach dem Übertragen des Files, noch bevor der Kernel startet !
- Wenn man zu langsam ist crashed der Boot Prozess, man muß die Box resetten und vom Schritt 2 wieder beginnen.

Schritt 4:

- Stecke den vorbereiteten FAT32 USB Stick in den Front USB Anschluß des Gerätes.
- Folgende Kommandos (Befehle) ausführen:
- **usb start** -> wenn der USB Stick nicht erkannt wird, wiederhole den Befehl
- **fatload usb 0 \$loadaddr uldr.bin**
- **sfc erase uldr**
- **sfc write uldr**

- **fatload usb 0 \$loadaddr u-boot.bin**
- **sfc erase u-boot**
- **sfc write u-boot**
- **saveenv** -> speichert die Variablen

Wer sich ab hier für das einfachere Emergency Restore via USB entscheidet (>empfohlen !), kann nun zu Schritt 5 springen und geht dann nach oben zur Methode: „Apollo - Recovery von Kernel und Image via USB“

- **fatload usb 0 \$loadaddr vmlinux.ub.gz**
- **sfc erase kernel**
- **sfc write kernel**

- **nand erase**
- **fatload usb 0 \$loadaddr rootfs.arm.jffs2.nand**
- **Konvertiere in HEX die angezeigte dezimale Größe in Bytes und verwende den HEX Wert im nächsten Befehl!**
- **nand write.jffs2 root0 <Größe des ROOT Image in HEX>**

- **fatload usb 0 \$loadaddr appfs.jffs2.nand**
- **Konvertiere in HEX die angezeigte dezimale Größe in Bytes und verwende den HEX Wert im nächsten Befehl!**
- **nand write.jffs2 appfs0 <Größe des APPFS Image in HEX>**

Schritt 5:

Schalte Box aus, Jumper JP1 abstecken, baue die Box wieder zusammen.

Schritt 6:

Nun sollte die Box wieder korrekt funktionieren.