



DSLlinux - „Linux spielen“

Uwe Berger



Uwe Berger

- Informatiker, PPS-Software
- Freizeit: Hard- und Software-Hacking
- Freie Software ist eine Philosophie!
- Brandenburger Linux User Group (BraLUG) e.V.
- Brandenburger Linux Infotag (BLIT) --> 06.11.2010



Disclaimer

Die Verwendung von DSLinux auf einer NDS ist legal, da dazu keine geschützte Firmware gepatcht werden muss.

Mit der dazu notwendigen Hardware/Zusatzsoftware könnten aber auch Copyright-Rechte anderer verletzt werden. Technische Details dazu sind deshalb auch nicht Gegenstand dieses Vortrages.

Dieser Vortrag ist kein Aufruf zur Verletzung dieser Rechte.

Ich verwende auch sonst keine illegal kopierte NDS-Software!



Inhalt

- NDS-Hardware
- Homebrew-Software
- DSLinux
 - Installation
 - Konfiguration
 - Applikationen, Dienste etc.
 - Softwareentwicklung
 - Wozu?



- NDS-Hardware
- Homebrew-Software
- DSLlinux



Nintendo DS

- DS steht für:
 - Dual Screen
 - Double Screen
 - Legende: Developers' System
- Versionen:
 - Nintendo DS Lite (DSLINUX-fähig)
 - Nintendo DSi (Cam, SD-Card, kein Slot-2(!))





NDS-Hardware (NDS Lite)

- 2 CPUs:
 - ARM7 33 MHz (32-bit RISC; 130 MIPS)
 - ARM9 66 MHz (32-bit RISC; 300 MIPS; MPU)
- RAM:
 - 4MByte
 - keine MMU



NDS-Hardware (NDS Lite)

- Grafik:
 - 2D/3D in "Hardware"
 - maximal 262144 Farben
 - 656 KB RAM
 - 60 Frames/Sekunde
 - 120k Polygone/Sekunde; bis 30M Polygone/Sekunde
Pixelfüllrate



NDS-Hardware (NDS Lite)

- Displays:
 - max. Auflösung (beide): 256x192
 - Diagonale: 7,62 cm (3")
 - Hintergrundbeleuchtung
 - unteres Display als Touchscreen
- Sound:
 - 16 Bit Stereo
 - internes Mikrophone
 - Kopfhörer-Ausgang



NDS-Hardware (NDS Lite)

- WLAN:
 - IEEE 802.11b; 1 oder 2 MBit
- 8 Tasten (Start; Select; A; B; X; Y; R; L); Steuerkreuz
- 2 Kartenschächte für:
 - NDS-Spiele
 - GBA-Spiele



- NDS-Hardware
- Homebrew-Software
- DSLinux



Homebrew-Software

- Homebrew: wrtl. "selbst gebrautes Bier"
- meint: unautorisierte, selbst erstellte Programme für Konsolen, Handhelds, Handys u.ä.
- rechtliche Grauzone
 - legal: Umgehung von Schutzmechanismen
 - illegal: Modifizierung von Hard-/Software des Herstellers
- diverse Homebrew-Software verfügbar



Homebrew-Software

- Toolchain:
 - ARM-Compiler (DevkitARM)
 - ein paar NDS-Libs (libnds, libfat, dswifi, PALib)
 - eventuell einen Emulator (z.B. DeSmuME)
- Freepascal: Fpc4nds (<http://itaprogramming.free.fr/>)
- gute Tutorials z.B.:
 - <http://www.tobw.net/dswiki>
 - <http://www.dev-scene.com/NDS/Tutorials>
 - <http://patater.com/manual>



Demo: „Hello World!“ (im Emulator)



- NDS-Hardware
- Homebrew-Software
- DSLinux



DSLlinux - Erforderliche Zusatzhardware

- Slot-1: DLDI-Zusatzkarte
 - Dynamically Linked Device Interface for libfat
 - <http://dldi.drunkencoders.com>
 - <http://dslinux.org/cgi-bin/moin.cgi/ComparisonOfHomebrewHardware>
- Slot-2: "Speichererweiterung"
 - Erweiterung des RAMs auf bis zu 32 MByte
 - <http://dslinux.org/cgi-bin/moin.cgi/RunningDSLlinux>



DSLlinux - Erforderliche Zusatzhardware

- WLAN-Router
 - nur WEP möglich
 - <http://dslinux.org/cgi-bin/moin.cgi/WorkingRouters>
 - mein Router: NETGEAR WGT634U
 - WEP mit 64-Bit-Schlüssel
 - Kanal 10
 - ESSID-Broadcast eingeschaltet
 - kein DHCP



DSLlinux - "Kennzahlen"

- Projektseite: <http://dslinux.org>
- Grundlage: uClinux (<http://www.uclinux.org/>)
- 2.6.x-Kernel
- Hardware fast vollständig unterstützt (NDS Lite)
- Userland: busybox, uClibc, ncurses
- Standard-Shell: msh (busybox)



DSLlinux – Installation (Methode 1)

- fertigen Kernel und Filesystem verwenden
 - Download: <http://kineox.free.fr/DS/>
 - Archiv auspacken
 - Kernel auf SD-Card kopieren
 - Linux-Verzeichnisbaum auf SD-Card kopieren
 - SD-Card in NDS, booten und fertig!



DSLlinux – Installation (Methode 2)

- DSLlinux-Build-System: aus den Quellen generieren
- Tutorial: <http://dslinux.org/cgi-bin/moin.cgi/CompilingDSLlinux>
 - Toolchain besorgen/installieren
(<http://stsp.spline.de/dslinux/toolchain/>)
 - nicht vergessen: bison, flex, m4, texinfo
 - DSLlinux-Quellen entpacken (<http://stsp.spline.de/dslinux/>)
 - eventuell: `svn update`
 - `make menuconfig`
 - `make`
 - ... und auf SD-Card kopieren (images/...) etc.



DSLlinux - Verzeichnisbaum

CF/SD card

```
|
|--dslinux.nds
|--dslinuxm.nds
|--linux/
    |--etc/
        |--rc.d/
    |--home/
    |--lib/
        |--...
    |--usr/
        |--...
    |--var/
        |--...
```



Demo: make menuconfig



DSLlinux - Konfiguration

- Root-Passwort (Default): uClinux
- Grundkonfiguration (Netzwerk, WLAN, Dienste, Login-Methode):
 - `cfg_rc`
- Netzwerk (nur Netzwerk, WLAN):
 - `cfg_network`
- Dienste (nur Dienste):
 - `cfg_inetd`
- ... oder natürlich `/etc/...` mit Editor bearbeiten ;-)



Demo via ssh/telnet:
cfg_..., /etc/...
dmesg, free, mount, df



DSLlinux - Anwendungssoftware

- Liste: <http://dslinux.org/cgi-bin/moin.cgi/AppDir>
 - diverse bekannte Unix-Tools
 - Editoren: nano, joe, vi
 - diverse Spiele (Textkonsole)
 - Netzwerk: Mail (nail), Browser (links, retawq), IRC
 - Scripting: MicroPerl, picoC, dialog
 - Multimedia; mp3-Player, Audiorecorder
 - Datenbank: sqlite3
 - ...



Demo via ssh/telnet:

nano

mc

Spiele...



DSLlinux – Dienste und weiteres

- verfügbare Dienste:
 - syslogd
 - httpd (boa)
 - Terminal: sshd, telnet
 - Filesystem: ftpd, nfsd, samba
 - Audio: esd (Enlightenment Sound Daemon)
- DynDNS.org: inadyn
- iptables
- OpenVPN

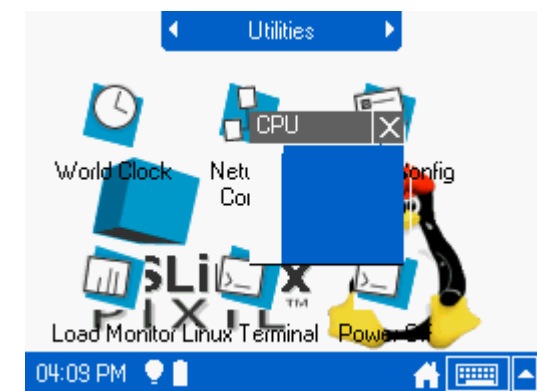
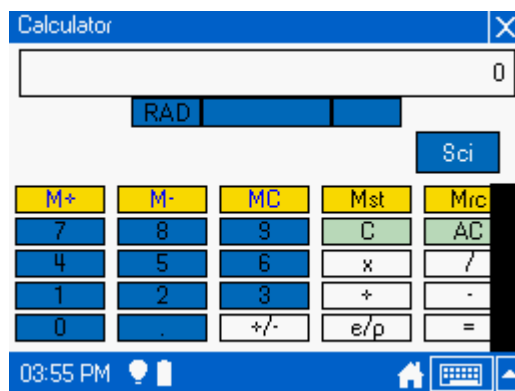


Demo: httpd, ftp, smb



DSLlinux - „Bunte Oberfläche“?

- X-Server: Nano-X
(<http://www.microwindows.org/>)
- Window-Manager: PIXIL OE
(<http://www.pixil.org>)
- NDS-Port sehr langsam und instabil
- ein paar wenige Anwendungen





Softwareentwicklung für DSLinux

- <http://dslinux.org/cgi-bin/moin.cgi/PortingHowto>
- Voraussetzungen:
 - installierte Toolchain (siehe DSLinux-Installation)
 - funktioniert DSLinux-Build?
 - Umgebung für Cross-Compiling anpassen
- Softwareentwicklung:
 - Anwendungen für DSLinux portieren
 - eigene/neue Anwendungen realisieren



Softwareentwicklung für DSLinux

- zwei mögliche Verfahrensweisen:
 - „Standalone“
 - Programm schreiben und übersetzen
 - nach /usr/bin auf NDS kopieren (z.B. via ftp, nfs etc.)
 - Programm auf NDS ausführen...
 - DSLinux-Build-System
 - Programm schreiben und testen
 - ins Build-System integrieren (Quelltext, Makefile, config.in, config.help etc.)
 - eventuell in ein offizielles SVN einspielen



Demo: Berliner Uhr



DSLlinux - Wozu?

- als "Single-Desktop"
- als Client im Netzwerk
- als Server im Netzwerk
- Portierung/Eigenentw. weiterer Anwendungssoftware
 - Cool: niedriger Stromverbrauch, Netzwerkfähigkeit
 - Datenlogger, Überwachung, ...
 - Problem: außer WLAN, keine weiteren Schnittstellen (außer vielleicht der Mikrophon-Eingang...)



Danke für die Aufmerksamkeit!

http://bralug.de/wiki/BraLUG_auf_dem_CLT2010