

# PyNeo installieren

Patrick Beck

22. April 2012



# Inhaltsverzeichnis

<b>1</b>	<b>pyneo installieren</b>	<b>5</b>
1.1	Vorbereitung der microSD-Karte . . . . .	5
1.1.1	microSD-Karte partitionieren . . . . .	5
1.1.2	ext3-Dateisystem einrichten . . . . .	5
1.2	Basissystem installieren . . . . .	5
1.2.1	rootfs installieren . . . . .	5
1.2.2	kernel installieren . . . . .	5
1.3	Booten . . . . .	6
1.4	Einrichten . . . . .	6
1.4.1	User anlegen . . . . .	6
1.4.2	pyneod.ini Configdatei . . . . .	6
1.4.3	Homepartition auf der zweiten Partition einrichten . . . . .	7
1.4.4	Update . . . . .	7



# Kapitel 1

## pyneo installieren

Eine Schnelleinführung zur Installation von pyneo findet man auf der offiziellen Webseite hier. Ich möchte auf dieser Seite eine zu 100% nachvollziehbare Installationsanleitung verfassen und freue mich wie immer über Anregungen.

Anmerkung: Qi muss für das folgende Setup installiert sein. Alle Befehle müssen als root ausgeführt werden.

### 1.1 Vorbereitung der microSD-Karte

pyneo basiert seit kurzer Zeit auf Debian und wird damit auf die microSD-Karte installiert, was darüber hinaus auch Flexibilität bringt. Es bietet sich an zwei Partitionen auf der microSD-Karte anzulegen. Eine Partition in diese das Debian-System installiert wird, sowie eine für die eigenen Dateien.

#### 1.1.1 microSD-Karte partitionieren

Als root mit `cfdisk /dev/*microsddevice*` `cfdisk` im interaktiven Modus starten (Dateisystem darf nicht eingebunden sein). Anschließend zwei neue Partitionen (System: 1 GB<sup>1</sup>, Daten: Rest des freien Bereichs) erstellen und mit einem Linux-Dateisystem (83) markieren.

#### 1.1.2 ext3-Dateisystem einrichten

Mit folgendem Kommando wird das ext3-Dateisystem eingerichtet:

```
mkfs -t ext3 /dev/*microsdpartition*
```

Jeweils für jede der zwei Partitionen ausführen.

### 1.2 Basissystem installieren

#### 1.2.1 rootfs installieren

Erste Partion mit 1 GB<sup>1</sup> ins System einbinden und in das Verzeichnis wechseln. Anschließend auf der pyneo Downloadseite das Debiansystem mit pyneo (armelchroot-full, pyneo mit allen Abhängigkeiten) herunterladen.

Beispiel:

```
wget http://pyneo.org/downloads/debian/armelchroot-full.tar.gz
```

Das Archiv anschließend mit `tar -xvzf *archiv*` entpacken.

#### 1.2.2 kernel installieren

Die Kernel-Installation besteht aus einem Kernel (GTA01 / GTA02), sowie einem dazupassenden Modulpaket.

---

<sup>1</sup>Gigabyte

## Kernel GTA01

```
wget http://pyneo.org/downloads/debian/modules-GTA01_andy-tracking_2763b78dd365a140.tar.gz
wget http://pyneo.org/downloads/debian/uImage-moredrivers-GTA01_andy-tracking_2763b78dd365a140.bin
```

## Kernel GTA02

```
wget http://pyneo.org/downloads/debian/modules-GTA02_stable-b749274fadec33a6.tar.gz
wget http://pyneo.org/downloads/debian/uImage-moredrivers-GTA02_stable-b749274fadec33a6.bin
```

Der Kernel (uImage\*) wird nach /boot auf der microSD-Karte kopiert. Das Archiv wird im Wurzelverzeichnis der ersten Partition entpackt (wie oben). Damit Qi den Kernel findet muss ein Symlink im Boot-Verzeichnis erstellt werden. Qi sucht auf dem GTA01 nach dem Namen uImage-GTA01.bin bzw uImage-GTA02.bin auf dem GTA02. Die Dateien duerfen Symlinks sein, was das wechseln der Kernel einfach macht. Beispiel für GTA02:

```
mv uImage-moredrivers-GTA02_stable-b749274fadec33a6.bin boot
cd boot
ln -s uImage-moredrivers-GTA02_stable-b749274fadec33a6.bin uImage-GTA02.bin
```

## 1.3 Booten

Nachdem ihr die oben genannten Anweisungen befolgt habt, solltet ihr nun schnellstmöglich ;) eure microSD-Karte zurück ins Neo einlegen und starten (normal über die Power-Taste). Bereits nach kurzer Zeit sollte euch ein Terminal auf dem Display begrüßen. Nun ist auch das einloggen über ssh möglich.

## 1.4 Einrichten

### 1.4.1 User anlegen

Einloggen über ssh

```
ssh root@192.168.0.202
```

Sollte es hier zu Problemen kommen, muss zuerst das Netzwerkdevice für das Freerunner ermittelt werden, dies kann mit dem "dmesgBefehl geschehen. Anschließend folgende Befehle ausführen:

```
ifconfig eth2 192.168.0.200
ssh root@192.168.0.202
```

Damit wird dem Netzwerkdevice eth2 die IP 192.168.0.200 zugewiesen. Um dies zu automatisieren kann man die Einstellungen direkt in der Datei /etc/network/interfaces abspeichern - siehe dazu patricks\textunderscoreersteschritte Nach dem einloggen solltet ihr zuerst das System grundlegend einrichten. pyneo möchte einen Schlusstrich hinter die Benutzung von root als Standarduser ziehen deshalb muss zuerst ein Standarduser im System eingerichtet werden.

```
adduser user
```

Danach alle Fragen beantworten und ihr Habt einen neuen Nutzer mit dem Namen User. User bietet sich an da bereits die Config für pyneo auf User ausgelegt ist, kann aber auch umgeändert werden.

### 1.4.2 pyneod.ini Configdatei

Falls ihr besondere Einstellungen (anderer Nutzernamen, damit ändern sich die Verzeichnisnamen) ändern wollt, findet ihr diese unter /etc/pyneod.ini

### 1.4.3 Homepartition auf der zweiten Partition einrichten

Es bietet sich an das /home-Verzeichnis direkt auf der microSD-Karte anzulegen, so steht es immer zur Verfügung auch wenn eines neues Betriebssystem auf der ersten Partition der Karte installiert wird. Dazu ist einfach folgende Ergänzung in der Datei /etc/fstab durchzuführen.

```
/dev/mmcblk0p2 /home      ext3      defaults          0 2
```

Damit sieht die Datei folgendermaßen aus.

```
/dev/mmcblk0p2 /home      ext3      defaults          0 2
rootfs /          ext2      defaults,errors=remount-ro,noatime 0 1
proc /proc          proc      defaults          0 0
tmpfs /etc/network/run tmpfs     defaults,noatime 0 0
tmpfs /tmp            tmpfs     defaults,noatime 0 0
tmpfs /var/lock       tmpfs     defaults,noatime 0 0
tmpfs /var/run        tmpfs     defaults,noatime 0 0
tmpfs /var/tmp        tmpfs     defaults,noatime 0 0
```

Und schon wird ab sofort bei jedem Start die zweite Partition der microSD-Karte unter /home gemountet.

### 1.4.4 Update

Da pyneo noch ein relativ junges Projekt mit vielen Änderungen ist und auch nicht die Ressourcen hat, durchweg neue System-Tar-Archive zu publizieren laden wir einfach das aktuelle pyneo herunter und installieren es im System händisch.

Frank Gau aus dem IRC<sup>2</sup>-Channel #pyneo.org auf freenode.net hat dazu ein kleines Shell-Script geschrieben das uns jene Arbeit abnimmt. Eine kleine Veränderung meinerseits erlaubt das herunterladen in das /tmp-Verzeichnis dazu muss kein Verzeichnis /home/scripts bestehen.

```
1  #!/bin/sh
2
3  /etc/init.d/pyneod stop
4  /etc/init.d/gsm0710muxd stop
5
6  rm -r /usr/share/pyneod
7  rm -r /usr/lib/python2.5/site-packages/pyneo
8  rm /etc/pyneod.ini
9  rm /etc/dbus-1/system.d/pyneod.conf
10
11 cd /tmp/
12
13
14 wget http://pyneo.org/downloads/HEAD/pyneod-HEAD.tgz
15 wget http://pyneo.org/downloads/HEAD/python-pyneo-HEAD.tgz
16
17 tar -xvzf pyneod-HEAD.tgz
18 tar -xvzf python-pyneo-HEAD.tgz
19
20 rm python-pyneo-HEAD.tgz
21 rm pyneod-HEAD.tgz
22
23 mv pyneod-HEAD /usr/share/pyneod
24 mv /usr/share/pyneod/pyneod.ini /etc/
25 mv /usr/share/pyneod/pyneod.conf /etc/dbus-1/system.d/
26
27 cd python-pyneo-HEAD
28 mv pyneo /usr/lib/python2.5/site-packages
29 cd ..
30 rm -R python-pyneo-HEAD
31
32 /etc/init.d/dbus restart
33 /etc/init.d/gsm0710muxd start
34 /etc/init.d/pyneod start
```

Vim starten, 'i' drücken, Script reinkopieren, ESC drücken und mit ':w update.sh' abspeichern sowie mit :q den Editor verlassen. Danach das Script ausführbar machen und den Updateprozess mit den folgenden Befehlen einleiten.

<sup>2</sup>Internet Relay Chat

```
chmod +x update.sh  
./update.sh
```

Sollte bei euch die Domainauflösung nicht funktionieren, ist sicherlich kein Nameserver im lokalen Netz verfügbar dazu einfach folgenden Befehl ausführen.

```
echo nameserver 208.67.222.222 > /etc/resolv.conf
```

Für dauerhafte Einrichtung ist in der Datei `/etc/network/interfaces` im `usb0`-Bereich folgendes hinzuzufügen

```
up echo nameserver 208.67.222.222 > /etc/resolv.conf
```

Danach das Script nochmal ausführen. Nun sollte der Updateprozess funktionieren.