Ausarbeitung der Projektarbeit: Handheld GSM-BTS-Scanner

Christian Kobiela, Dennis Dette

7.10.2016

Inhaltsverzeichnis

1 Installationsanleitung **3**
1.1 Installation von benötigter Software für gr-gsm 3

1 Installationsanleitung

Ausführlichere Anleitung und weitere Wiki-Einträge sind hier nachzulesen:
`https://github.com/ptrkrysik/gr-gsm/wiki/Installation-on-RaspberryPi-3`

1.1 Installation von benötigter Software für gr-gsm

Installieren von Kalibrate

Als erstes installieren wir Kalibrate:

```
1 sudo apt-get install libtool autoconf automake libfftw3-dev librtlsdr0
      librtlsdr-dev libusb-1.0-0 libusb-1.0-0-dev
2 git clone https://github.com/asdil12/kalibrate-rtl.git
3 cd kalibrate-rtl
4 git checkout arm_memory
5 ./bootstrap
6 ./configure
7 make
8 sudo make install
```

Zugriff auf USB-Device freischalten

Das RTL-SDR Gerät einstecken und ID mit dem Befehl *lsusb* überprüfen. Zu sehen sollte etwas wie folgend sein:

```
Bus 001 Device 004: ID **0bda:2832** Realtek Semiconductor Corp. RTL2832U
     DVB-T
Bus 001 Device 003: ID 0424:ec00 Standard Microsystems Corp. SMSC9512
    /9514 Fast Ethernet Adapter
Bus 001 Device 002: ID 0424:9514 Standard Microsystems Corp.
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

In unserem Fall ist die ID **0bda:2832**. Anschließend öffnen wir eine rules-Datei:

```
sudo nano /etc/udev/rules.d/20.rtlsdr.rules
```

...in welcher dann folgende Zeile hinzugefügt werden muss:

```
SUBSYSTEM=="usb", ATTRS{idVendor}=="0bda", ATTRS{idProduct}=="2832",
    GROUP="adm", MODE="0666", SYMLINK+="rtl_sdr"
```

Falls mehrere RTL-SDR Geräte verwendet werden, können mehrere Zeilen hinzugefügt werden. Die ID muss jeweils natürlich entsprechend des Gerätes abgewandelt werden.

Danach sollte der Raspberry Pi neugestartet werden: `sudo reboot`

Kalibrierung des RTL-SDR Gerätes

Jetzt können wir den Befehl ausführen um das RTL-SDR Gerät zu kalibrieren (um genau zu sein um den durschnittlichen absoluten Fehler in ppm zu berechnen):

```
kal -s GSM900
```

Das Ergebnis sollte ähnlich zu diesem sein:

```
Found 1 device(s):
  0:  Generic RTL2832U

Using device 0: Generic RTL2832U
Found Rafael Micro R820T tuner
Exact sample rate is: 270833.002142 Hz
kal: Scanning for GSM-900 base stations.
GSM-900:
        chan: 1 (935.2MHz - 33.430kHz)  power: 55085.23
        chan: 3 (935.6MHz - 34.130kHz)  power: 63242.36
        chan: 5 (936.0MHz - 33.970kHz)  power: 41270.82
...
...
        chan: 112 (957.4MHz - 32.934kHz)        power: 498930.07
        chan: 116 (958.2MHz - 31.859kHz)        power: 88039.44
        chan: 124 (959.8MHz - 32.429kHz)        power: 247404.23
```

Das stärkste Signal wäre in diesem Fall Kanal 112. Also führen wir die Kalibrierung auf diesem Kanal durch:

```
kal -c 112
```

...und erhalten ein Ergebnis wie folgt:

```
Found 1 device(s):
  0:  Generic RTL2832U

Using device 0: Generic RTL2832U
Found Rafael Micro R820T tuner
Exact sample rate is: 270833.002142 Hz
kal: Calculating clock frequency offset.
Using GSM-900 channel 112 (957.4MHz)
average         [min, max]      (range, stddev)
- 34.368kHz             [-34376, -34357]        (20, 4.697051)
overruns: 0
not found: 0
average absolute error: 35.897 ppm
```

Unser durschschnittlicher absoluter Fehler wäre hier also 36 ppm (35.897 ppm).

Installation von GNU Radio

Als nächstes installieren wir GNU Radio:

```
sudo apt-get install gnuradio gnuradio-dev
```

Installieren von libosmocore

Libosmocore muss kompiliert werden...

```
sudo apt-get install cmake
sudo apt-get install build-essential libtool shtool autoconf automake git
    -core pkg-config make gcc
sudo apt-get install libpcsclite-dev libtalloc-dev
git clone git://git.osmocom.org/libosmocore.git
cd libosmocore/
autoreconf -i
./configure
make
sudo make install
sudo ldconfig -i
cd
```

...außerdem benötigen wir noch ein paar andere Dinge.

```
sudo apt-get install gr-osmosdr rtl-sdr
sudo apt-get install libboost-dev
sudo apt-get install osmo-sdr libosmosdr-dev
sudo apt-get install libusb-1.0.0 libusb-dev
sudo apt-get install libboost-all-dev libcppunit-dev swig doxygen
    liblog4cpp5-dev python-scipys
```

Installation von gr-gsm

Und nun zum letzten Schritt:

```
git clone https://github.com/ptrkrysik/gr-gsm.git
cd gr-gsm
mkdir build
cd build
cmake ..
make
sudo make install
sudo ldconfig
```

Zuletzt erstellen wir noch die /.gnuradio/config.conf config-Datei, mit *nano /.gnuradio/config.conf*. Und fügen diese zwei Zeilen hinzu (damit GNU Radio die custom Blöcke von gr-gsm finden kann):

```
[grc]
local_blocks_path=/usr/local/share/gnuradio/grc/blocks
```