ScannerList

Wolfgang Hallmann – DF7PN – ©2017 ff.

(englisch below)

# Für was?

Zur Verfolgung von Wettersonden im Frequenzbereich 400-406 MHz nutzen viele das Softwarepaket von OE5DXL – Chris, einem Funkamateur aus Österreich. Mit dem Softwarepaket kann man unter Unix und Windows und einem DVBT-Stick die Sonden empfangen lassen und deren Daten auswerten.

GitHub: <https://github.com/oe5hpm/dxlAPRS>

ScannerList unterstützt das Programm von Chris, welches eine Liste von Frequenzen gleichzeitig scannt, die man vorher gefüllt haben muss. Dieses Füllen klappt sogar während das Proramm läuft. Bedeutet: man kann die Liste jederzeit korrigieren und sie wird sofort verwendet ohne Programmunterbrechung.

Einige haben sich mehr oder weniger interessante Scripte gebastelt, die mit Hilfe des Programms „rtl\_power“ einen Bereich scannen und die Empfangswerte umrechnen in Frequenzen, wenn ein brauchbarer Dezibel-Wert ausgegeben wird. Diese Auswertung übernimmt „ScannerList“ mit einigen Annehmlichkeiten.

# Warum sowas?

Eigentlich könnte man auch ohne ScannerList auskommen. Aber einen Sinn muss es ja haben, werdet ihr denken, dass es das Teil gibt. Also hier die Vorteile:

Eigentlich füllt man in die Konfig-Datei sdrcfg der dxl-Softwre Zeile für Zeile alle Frequenzen die man das Programm überwachen lassen will. Es gibt einige Frequenzen die immer zuverlässig benutzt werden. Was aber mit den vielen anderen? Gerade die DFM-Modelle des Militärs senden alle Nase lang wo anders. He he macht das mal, schaufelt die Liste mal ordentlich voll und dann schaut auf die CPU Auslastung. Alle Angaben müssen ausgewertet werden. Das kostet CPU Zeit und hält das Zimmer warm. Das muss nicht sein.

Mit ScannerList werden nur die wirklich gerade aktiven Frequenzen zur Überwachung angewiesen. Die sdrcfg bleibt klein und die CPU kalt. Will man auf Nummer-sicher-gehen, kann man in eine Liste trotzdem feste Frequenzen hinterlegen, also die, die immer täglich aufschlagen. Dann ist der Decoder halt gleich beim ersten Piep am Arbeiten. Läuft ScannerList aber alle 1-2 Minuten per Cron-Job ist das kein großer Unterschied.

Damit ScannerList keinen falschen Alarm gibt und evtl doch mal Störungen nicht erkennt und sinnlos zur decodierung in Auftrag gibt, kann auch eine Blacklist bestückt werden.

# Funktionsweise

Für einen Scanner benötigt man einen dedizierten DVBT-Stick der nur dafür zur Verfügung steht.

Gemessen wird in Intervallen, nicht im Dauerbetrieb.

Wir benötigen ein Script (Beispiel: scanner\_prod.sh) welches per CRONJOB alle 2 Minuten (bei mir) aufgerufen wird. Es startet den „rtl\_power“-Scan mit einem Block von 402 bis 404 MHz, danach läuft „ScannerList“ und werte die csv-Datei von rtl\_power aus. Erstellt eine für das DXL-Paket konforme Frequenzliste als Datei. Diese wird per SCP (Filetransfer) auf den jeweiligen Rechner übertragen. Die gleichen Schritte laufen dann nochmal für 404 bis 406 MHz ab. Die Frequenzlist geht an einen anderen Rechner. Warum nur 2 MHz? Weil die DVBT-Sticks i.d.Regel nur diesen Bereich maximal mit einem Rutsch abbilden können.

Der wirklich interessante Schritt ist die Auswertung der Dezibel-Werte vom rtl\_power und daraus Sonden heraus zu lesen und von Störungen zu unterscheiden, die weniger Interessant sind.

# Signalerkennung

Kurz: wie scannt rtl\_power? Schaut man sie die Parameter an, gibt man nicht nur den Frequenzbereich an, sondern auch das Intervall in dem zwischen den Bereichen abgetastet werden soll. Ich empfehle hier 1.000 Herz-Schritte (1 Khz).

Zusammen mit den üblichen Angaben für DEVICE-NR, GAIN und FREQUENZKORREKTUR erhält man eine Ausgabe im CSV-Format. Vorne stehen die Parameter mit denen empfangen wurde und danach folgen für jeden Step (1 KHz) Dezibelwerte. Diese werden in negativen Zahlen angegeben. Nimmt ein Signal zu steigt der Wert bis gegen 0 an.

Kurz: wie sehen Sonden-Signale aus? Nimmt man ein SDR-Programm und lässt es auf ein solches Signal los, sieht man eine belegte Breite von ca. 6-12 Khz – je nach Sondentyp.

Der Scanner tastet im 1 KHz Raster ab und wird daher nebeneinander mehrere gute Signalwerte sehen (also hier 6 bis 12 Stück). Genau das macht sich das Programm zunutze um Störsignale erkennen zu können. Störungen sind in der Regel einzelne recht schmalbandige, meist unmodulierte Träger ohne Informationsinhalt. Kurzum, sie fallen aus dem Raster.

Wie erkennt man nun ob ein einzelner Messwert ein Signal darstellt? Ich bekomme ja nur eine Zahl. Noch dazu bekomme ich immer eine Zahl, auch wenn nur Rauschen – also kein Signal - vorhanden ist.

Gute Frage. Dazu muss man entweder wissen, welche Werte nur Rauschen darstellen und ab wann ein brauchbares Signal aus dem Rauschen hervortritt. Hier hilft ein Blick in die CSV Datei die uns rtl\_power ausgehändigt hat. Die CSV Datei kann man in EXCEL oder ähnlichen Programmen laden und betrachten. Auf den meisten Frequenzen wird prozentual gesehen nur Rauschen sein. Mit dem Auge findet man das ziemlich schnell. Wann ist ein Signal dekodierbar von der dxl-Software? Dazu kann man sich die Mühe machen und mit einem zusätzlichen Stick an gleicher Antenne ein SDR Programm zu laden und auf der gleichen Frequenz zuzuhören. Im Blick hält man die dxl-Software und achtet darauf, ab wann der Pegel zur Dekodierung ausreichend ist. Im SDR-Empfänger liest man jetzt den Wert ab. In der Praxis ist das sehr oft ein Wert der 10 Dezibel besser als das Rauschen ist.

ScannerList kann man jetzt über zwei Methoden nach Signalen suchen lassen. Einmal über den Parameter -L (fester Wert über Rauschen z.B: Rauschen = -35 db, wäre dann -20 db. Alles in Richtung Null wäre ein Signal was sich zu beobachten lohnt.

Die andere Methode ist eine Automatik. Hier wird über Parameter -n ein relativer Dezibel-Wert angegeben, ab dem ein Signal über dem Rauschen näher betrachtet werden soll. Z.B. -n 5 (positive Zahl). Rauschen = -35, Signale die 5 db besser sind wären gut: ab -30 db ist das so.

Nun hat dies zum Vorteil, dass sich der Rauschpegel im Laufe eines Tages auch mal ändern kann (Störungen etc). Dazu wurde eine Untersuchung des Rauschpegels pro 100 Messpunkte eingeführt. Der Pegel wird für jedes Segment herangezogen. Ist dann wieder von Vorteil, wenn der Rauschpegel z.B. in einem größeren Rahmen z.B. höher wäre und dadurch gegenüber der Methode mit -L bei fixer Vorgabe in die nähere Betrachtung gezogen, weil er die Signalschwelle (aus Beispiel oben von -20) erreicht. Ganz plötzlich laufen dann die Listen voll obwohl dort nichts ist. Gut, hier werden dann keine breiten Signale erkannt und die Muster fallen wieder durchs Raster.

So wer hier noch folgen kann, wird sich bestimmt die automatische Rauschpegelerkennung nutzen wollen. Wer hier nichts verstanden hat, dem rate ich das jetzt auch. Nehmt den Parameter „-n 5“, dann müsste das funktionieren.

# Weitere Erkennungsattribute

Signale vom Rauschen zu trennen, war der erste Schritt. Wir schauen jetzt, ob mindestens 4 Werte nebeneinander in die Erkennung gekommen sind. Ab jetzt wird es interessant. Es wird die Mittenfrequenz errechnet. Höchste Frequenz minus niedrigste Frequenz = Bandbreite. Niedrigste Frequenz + halbe Bandbreite = Mitte.

Nun haben die Sonden ein Frequenzraster von minimal 10 KHz. D.h. bei einigen KHz Abweichung kann man trotzdem guten Gewissens auf volle 10 KHz auf- oder abrunden. Das ist jetzt die Frequenz, die wir uns merken – hier ist Betrieb – darauf lassen wir den dxl-Empfänger los.

# Signal kurz weg – kein Problem

Durch Funk-Abschattungen kann ein Signal auch mal kurzzeitig verschwinden, kommt aber gleich wieder. Um zu vermeiden, dass der Scanner das mal für einige Minuten aus der Frequenzliste herausnimmt, gibt es in ScannerList noch eine Holdingliste. Sie merkt sich sämtliche Frequenzen für einen kleinen Zeitraum, den Sie per Parameter wählen können. Er steht ohne Angabe intern immer auf 10 Minuten, kann aber durch „-h“ verändert werden. Ich empfehle aus der Praxis Werte zwischen 5-15 Minuten.