

# De WVCG: een WaterValCallGenerator

Theorie en enkele experimenten met Software Defined Radio

door Wim Kruyf PAoWV

Voor velen is SDR vaak nog iets abstracts. Wim beschrijft in dit artikel een deel van de basistheorie en beschrijft enige eenvoudige experimenten, die u in staat stelt actief deze theorie in praktijk te brengen, zonder dat we zelf een voor SDR geschikte ontvanger gebouwd hebben.

Wij zendamateurs hebben onze machting te danken aan het feit dat we ons willen zelfontplooiën door te experimenteren.

Velen doen dat door met Japanse koepdozen aan radiowedstrijdjes mee te doen, die een zware belasting leggen op het uithoudingsvermogen.

Die dragen wat mij betreft de eretitel radiocommando's; ze kunnen dagen zonder te slapen doorgaan met hun uithoudingsslag. Ongetwijfeld is daar, en last but not least de vossenjacht, de naam 'radioSPORT' uit ontstaan. Petje af, ik pas.

Blijkens de extreem verschillende activiteit op de banden tussen contest- en contestloze dagen zijn er veel radiocommando's en radiosporters.

## Theorie

Langzamerhand begint de software defined radio door te dringen in de amateurshack. Het is dan ook zo, dat bij de zendexamens de bijbehorende basistheorie betreffende Nyquist theorema, DAC, ADC en digitale filtering FIR en IIR al vereist worden. Het gremium vereist dus meer van de aanstaande zendamateur dan brute koopkracht.

Ondanks de snelheidsbeperking van de aan de gewone zendamateur beschikbaar staande digitale hulpmiddelen, die zitten in een PC geluidskaart, kunnen we met minimale inspanning en kosten experimenteren op dit gebied.

Dit kunnen we doen door zelf een direct conversion ontvanger te bouwen, die het ontvangen antenne-signaalspectrum met een bandbreedte van 48 kHz dubbelgevouwen tot 24 kHz aan de geluidskaart in een PC aanbiedt, en daar er, via door experts gemaakte programmatuur, allerlei toverkunsten mee uithalen.

Kunsten zoals panoramische display over 48 kHz bandbreedte, variabele filterbandbreedte van het gekozen signaal, automatisch instellende selectoject die interferentietonen verwijderd uit het audiospectrum en een waterval display van spectrum analyse. Ook het gelijktijdig ontvangen van 2 of meer frequenties uit het gedetecteerde spectrum behoort tot de nuttige mogelijkheden voor contesters en dx-ers.

Het is voor iedere zendamateur tenminste interessant om te weten hoe de gekochte spullen ongeveer werken, voor het geval ze niets zelf bouwen of zich beperken tot de assemblage van een kitje.

## Fasemethode

Toen de SSB zenders voor het eerste werden gebouwd door amateurs was de filtermethode moeilijker te realiseren dan de fasemethode.

De fasemethode, toegelicht in figuur 1, houdt in dat je twee audiosignalen maakt van je microfoonsignaal, die voor elke audiofrequentie 90 graden in fase verschillen en voor elke audiofrequentie onderling gelijke amplitude hebben.

Dat gebeurde met een Dome netwerk. PA oIJ leverde die destijds aan bouwers, omdat de vereiste meetapparatuur om 1% R en C samen te stellen in de shack als regel ontbrak en hij op zijn QRL de gelegenheid had ze uit te zoeken met goede meetapparatuur.

Die twee audiosignalen uit het Dome netwerk werden gemengd met twee draaggolfoscillatorsignalen, die ook 90 graden verschilden, in gebalanceerde mengtrappen. Die mengtrappen vermenigvuldigen de ingangssignalen met elkaar. De draaggolf komt er niet door.

HF 90 graden uit fase is makkelijk te maken uit een kristal met een RC en een RL fasedraaiend netwerkje van plus en min 45

graden. Je draaggolfrequentie lag dan wel vast. De twee dubbelzijbandmodulaten uit de mixers werden opgeteld of afgetrokken, waardoor de linker of de rechter zijband overbleef.

Was de audioamplitude niet precies gelijk of de fasedraaiing niet precies 90 graden, dan resulteerde dat in wat minder zijbandonderdrukking.

Je kunt het met vectortjes (fasordiagram) tekenen voor meer fysisch begrip of je kunt het met de goniometrie, die ik van MULO-B heb onthouden, gemakkelijk narekenen.

De gebruikte formules zijn:

- $2 \cdot \cos(a) \cdot \cos(b) = \cos(a-b) + \cos(a+b)$
- $2 \cdot \sin(a) \cdot \sin(b) = \cos(a-b) - \cos(a+b)$
- $2 \cdot \sin(a) \cdot \cos(b) = \sin(a-b) + \sin(a+b)$

De X-tal oscillator met kristal op frequentie  $f_0$  leverde na de fasedraaiende netwerkjes  $\sin(2\pi \cdot f_0 \cdot t)$  en  $\cos(2\pi \cdot f_0 \cdot t)$ .

Het Dome netwerk leverde van een laagfrequent toon op frequentie  $f_1$  de onderling 90 graden verschillende uitgangssignalen  $\sin(2\pi \cdot f_1 \cdot t)$  en  $\cos(2\pi \cdot f_1 \cdot t)$

De balansmengtrappen die alleen maar de twee inputsignalen vermenigvuldigen leveren respectievelijk  $\sin(2\pi \cdot f_1 \cdot t) \cdot \sin(2\pi \cdot f_0 \cdot t)$  en de andere  $\cos(2\pi \cdot f_1 \cdot t) \cdot \cos(2\pi \cdot f_0 \cdot t)$

Met formules (1) en (2) is het uitgangsspectrum van de balansmengtrappen die de input signalen vermenigvuldigen te berekenen:

mengtrap 1:

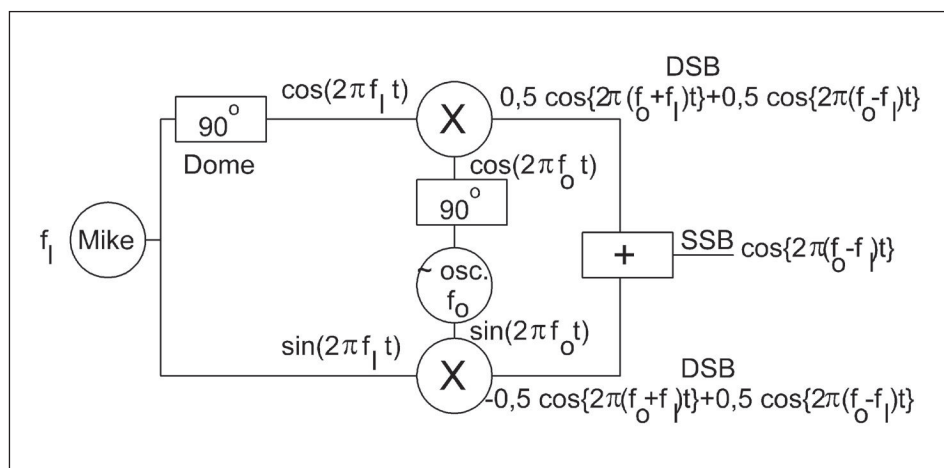
$$0,5 \cos\{2\pi (f_0 + f_1)t\} + 0,5 \cos\{2\pi (f_0 - f_1)t\}$$

dubbelzijband suppressed carrier dus.  
mengtrap 2:

$$-0,5 \cos\{2\pi (f_0 + f_1)t\} + 0,5 \cos\{2\pi (f_0 - f_1)t\}$$

Ook dubbelzijband suppressed carrier dus, maar de bovenzijband is nu 180 graden verschoven, wat blijkt uit het minteken.

Tellen we nu die twee outputs van de balansmengtrappen op dan resteert  $\cos\{2\pi (f - f_0)t\}$  uitsluitend de onderzijband dus, omdat de bovenzijbanden door de tegenfase elkaar opheffen.

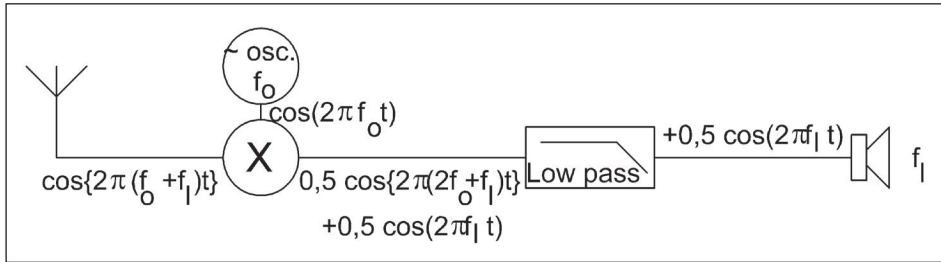


Figuur 1: De fasemethode bij SSB.

Trekken we de uitgangssignalen van de balansmodulatoren af dan resteert de bovenzijband  $\cos\{2\pi(f_0+f_1)t\}$ .

### Zo werkt dat

Die oude technieken worden nu weer in een nieuw jasje gebruikt in de Software Defined Radiotechniek. Maar eerst de volgende stap:



Figuur 2: Direct Conversion ontvanger.

### Direct Conversion Ontvanger

Een superheterodyne ontvanger mengt het antennesignaal naar een middenfrequentie MF, je hebt dan een lagere frequentie, en een vaste selectiviteit in een vast afgestemde middenfrequent versterker.

De ongewenste spiegelfrequentie, die ook de middenfrequent bereikt, ligt op twee keer de MF afstand van het gewenste signaal in de richting van de oscillatorfrequentie.

Bij de direct conversion ontvanger is de middenfrequent echter 0 Hz. Het principe is getekend in figuur 2. Je mengt dan de ontvangstfrequentie met een oscillator op dezelfde frequentie zodat de verschilfrequentie 0 Hz is. Dat lijkt ideaal, want SSB landt dan precies in het audiospectrum, als je afstemt op de plek van de onderdrukte draaggolf. Maar pas op: Stel dat je een frequentie  $f_0+f_1$  wilt ontvangen met de oscillator op  $f_0$ . Je houdt dan  $f_1$  over zoals bedoeld, maar  $f_0-f_1$  de onderzijband dus, levert precies hetzelfde verschil, ook  $f_1$ . De onder- en bovenzijband vallen dus over elkaar heen. CW stations die zitten tussen -3 kHz tot +3 kHz van je oscillatorsignaal hoor je beide op 0 tot 3 kHz.

Als je een station hoort kun je wel vaststellen of het onder of boven de oscillatorfrequentie ligt, door de oscillator iets hoger te verstemmen. Een station dat hoger ligt geeft dan een dalende toonhoogte en een station dat lager ligt juist een stijgende toonhoogte.

Een hoogfrequentspectrum van 48 kHz breed met de oscillatorfrequentie  $f_0$  in het midden ervan wordt dus gedetecteerd in een frequentieband van 0 tot 24 kHz, waarbij steeds twee antennefrequenties op een plek vallen in het laagfrequentspectrum. Die boven- en onderzijband willen we kunnen scheiden en daar komt de fase-methode weer om de hoek kijken.

### Een direct conversion ontvanger gecombineerd met de fasemethode

Een stuk van het frequentiespectrum, ty-

pisch 48 kHz breed, wordt gemengd met twee oscillatorsignalen in het midden van die band, die 90 graden in fase verschillen. Tegenwoordig gaat dat met een 4 deler, een Johnson counter, geklokt door een variabele oscillator die 4 maal hoger dan de gewenste oscillatorfrequentie is. Je kunt dan dus de oscillatorfrequentie wijzigen terwijl het 90 graden faseverschil ge-

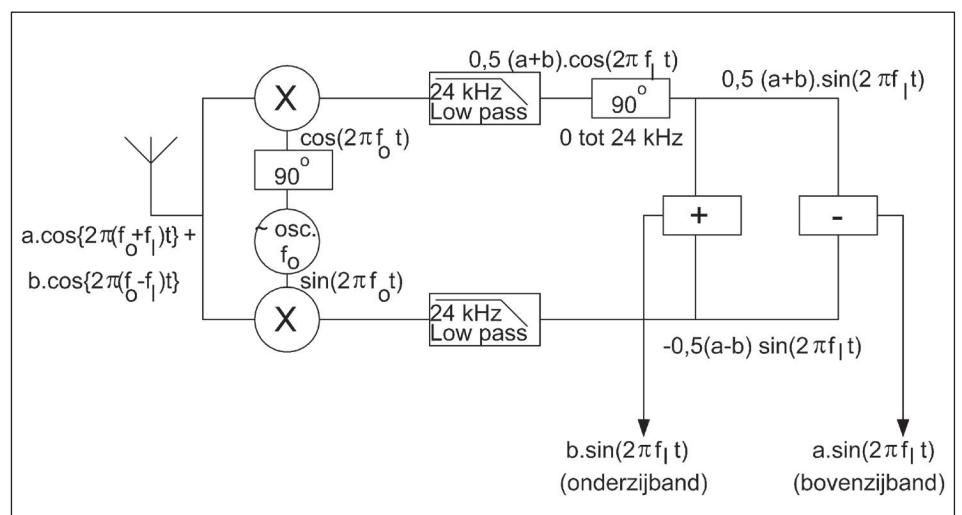
handhaafd blijft. De 4 deler levert dan de gewenste frequentie als blokgolven in twee of vier uitvoeringen die 90 graden onderling verschillen.

Met een analoge schakelaar wordt het ingangssignaal uit de antenne er steeds mee omgepoold, dat is dus vermenigvuldigen met een blok die 1 of -1 is. Het gevolg is dat elk van de twee mixers een audiospectrum afgeeft tussen 0 en 24 kHz en uiteraard veel andere mengcomponenten omdat een blok in tegenstelling tot een sinus bestaat uit een verzameling sinussen. De bedoeling is dat we die ongewenste mengproducten kwijtraken. Met een simpel RC laagdoorlatend filtertje kan dat, als je oscillatorfrequentie ver van je gewenste laagfrequentband ligt. Figuur 3 licht dat toe.

Een antennesignaal op frequentie  $f_0+f_1$ , te schrijven als  $\cos\{2\pi(f_0+f_1)t\}$ , levert dan als frequentiecomponenten in de twee aparte takken van de ontvanger achter de mengtrappen:

$$\text{tak1: } \cos\{2\pi(f_0+f_1)t\} \cdot \cos(2\pi f_0 t)$$

$$\text{tak2: } \cos\{2\pi(f_0+f_1)t\} \cdot \sin(2\pi f_0 t)$$



Figuur 3: Direct Conversion Fase ontvanger.

Met gebruikmaking van de basisformules (1) en (3) en weglaten (wegfilteren) van de hoogfrequente mengproducten resteert dan in de takken:

$$\text{tak1: } 0,5 \cos(2\pi f_1 t)$$

$$\text{tak2: } -0,5 \sin(2\pi f_1 t)$$

Die signalen in het laagfrequentgebied van 0 tot 24 kHz schelen 90 graden als je ze dus op een oscilloscope in XY mode aansluit, krijg je een cirkel te zien als Lissajousfiguur voor elke  $f_1$ .

Nu is de ontvanger zoals gezien niet alleen gevoelig voor frequenties hoger dan  $f_0$  maar evenzo voor frequenties lager dan  $f_0$ . Een frequentie die  $f_1$  lager is dan  $f_0$ , dus  $\cos\{2\pi(f_0-f_1)t\}$  levert na de mengtrappen:

$$\text{tak1: } \cos\{2\pi(f_0-f_1)t\} \cdot \cos(2\pi f_0 t)$$

$$\text{tak2: } \cos\{2\pi(f_0-f_1)t\} \cdot \sin(2\pi f_0 t)$$

Wegfilteren van de somproducten en andere hoog frequente producten uit de mengtrap levert als laagfrequentbijdrage in

$$\text{tak 1: } 0,5 \cos(-2\pi f_1 t) = 0,5 \cos(2\pi f_1 t)$$

$$\text{tak 2: } -0,5 \sin(-2\pi f_1 t) = 0,5 \sin(2\pi f_1 t)$$

Tak 1 verschilt dus niet met de andere zijband, maar tak 2 is in tegenfase met het signaal uit de andere zijband.

Stem je nauwkeurig af op een AM station, dan is steeds  $f_0+f_1=f_0-f_1$ . Twee zijbanden van een AM station zijn immers gelijk en blijkt dat de bijdrage in de takken tak 1  $\cos(2\pi f_1 t)$  is en tak 2 nihil, want de componenten van boven- en onderzijband geven daar een tegengestelde bijdrage.

Zijn de antennesignalen even ver boven als onder de oscillatorfrequentie niet gelijk, gewoon omdat het verschillende zenders zijn, dan vraag je je af of je die weer uit elkaar kunt halen.

Het antwoord is gelet op onze zojuist opgehaalde kennis van de fasemethode: "Ja".

### Analoog

Maar er zijn in de analoge wereld wel pro-

blemen. De boven- en de onderzijband leveren samen in de eerste tak altijd de som van de amplituden  $a$  van de boven- en  $b$  van de onderzijbandfrequentie, de onderste sinustak echter geeft voor de boventak een minteken en voor de ondertak een plusteken.

Daar ligt de sleutel om de zaak weer uit elkaar te trekken.

We plaatsen een soort Dome-netwerk dat 90 graden draait in een van de takken, in de tekening in figuur 3 is de bovenste tak genomen. De cosinus wordt dan een sinus, en vervolgens tellen we de twee takken op, dat levert de onderzijband of we trekken ze af, dat levert de bovenzijband.

Zet je de onderzijband op je linker kop-telefoonschelp en de bovenzijband op de andere schelp dan gaat de band waarop je afstemt zich tijdens afstemmen gedragen alsof je op de snelweg staat.

Probleem is natuurlijk die fasedraaiing die als je een spectrum wilt ontvangen van 48 kHz breed, moet worden gerealiseerd met een bandbreedte van 24 kHz. Stel je je bescheidener op en wil je niet meer horen dan je oren breed zijn, dan komt het Dome netwerk weer om de hoek kijken.

## Digitaal

Digitaal is dat 90 graden draaien over een bandbreedte van 24 kHz na bemonstering van het signaal geen probleem, en dat is ook de reden dat ik over 48 kHz praat, omdat een gewone doorsnee geluidskaart met een frequentie van 48 kHz bemonstert. Volgens het Nyquist criterium kan dus de geluidskaart maximaal 24 kHz audio-bandbreedte bemonsteren die monsters in 16 bits per monster digitaliseert. Het 90 graden fasenetwerk van 0 tot 24 kHz zit dan dus in de programmatuur, en direct achter de 24 kHz laagdoorlaatfilters kan de ontvanger zijn I en Q signalen, zo worden die genoemd, afleveren aan de PC.

De twee zijbanden van 24 kHz worden door de programmatuur op je panoramische display met 48 kHz bandbreedte gezet. 24 kHz voor elke zijband, die door deze bewerking uit elkaar getrokken zijn. Daarin kun je door je digitale BFO te variëren je zerobeat frequentie kiezen en je kunt een filter plaatsen dat een klein deel van het spectrum doorlaat, zoveel als nodig is voor de gewenste modulatiesoort. Die filters zijn als het FIR filters zijn (finite impulse response) digitaal met een lineaire fasekarakteristiek, zodat je filters kunt maken van een kwaliteit die met spoelen en condensatoren of actieve analoge op amps niet mogelijk is.

## Proeven met SDR

Als je nog geen direct conversion ontvanger hebt, kun je wel vast proeven nemen met de SDR programmatuur. PowerSDR-IQ v1.12.20 SV1EIA, Rocky v3.6 en Winrad 1.6.1 zijn voorbeelden die je gratis

kunt downloaden en installeren. Daan PAoFNB stuurde me de eerstgenoemde twee programma's toe, samen met Opname-1.wav die in PowerSDR-IQ kan worden afgespeeld, omdat hij het daarmee heeft opgenomen. Ze staan beschikbaar op mijn website vermeld verderop in dit artikel.

Van Rocky heb ik gezien, dat het te vinden is op [www.dxatlas.com/Rocky](http://www.dxatlas.com/Rocky), het draait (versie 3.6) niet onder Vista wel onder XP en Windows98SE, niet 98GE en niet de kale Windows 98.

Dat verklaart dus waarom dat bij mij op Vista gelijk al vastliep. Winrad is te vinden op [www.winrad.org](http://www.winrad.org), dat staat ook in de huidige nieuwste versie 1.6.1 als installatie-file beschikbaar op mijn website.

Mijn eerste aannemelijke resultaten komen uit Rocky op een Windows XP computer. PowerSDR-IQ levert rare verschijnselen, die ik vooralsnog niet kan verklaren.

Bovendien kan PowerSDR-IQ geen normale WAV files afspelen, maar alleen speciale floating point WAV files die het programma zelf als recording van ontvangen signalen aanmaakt. Daan PAoFNB zond me daartoe Opname\_1.wav toe die van mijn website kan worden gekopieerd. Maar de resultaten lijken bij mij nergens naar.

Bij PowerSDR-IQ moet je dus als je een CD als input wilt gebruiken die eerst branden, van bijvoorbeeld de IQ wav files uit de hierna besproken proeven; en vervolgens afspelen op je hifi installatie en de output daarvan naar je geluidskaart voeren van je PC waarop PowerSDR draait. PowerSDR vereist speciale merken/type geluidskaarten, gebruik je in de setup 'Unsupported audio card', dan wordt er niks gegarandeerd. De rare verschijnselen bij gebruik van PowerSDR-IQ die ik niet kon verklaren zijn dus mogelijk daaraan te wijten.

## WAV files

WAV files zijn (doorgaans) geluidsbestanden, die je op een PC of een CD speler kunt afspelen. Je kunt zelf een WAV file maken, en dan bedoel ik niet met een microfoon, maar met een rekenprogramma. Dat is voor de zendamateer een interessante bezigheid, want de gewone WAV files bevatten PCM (pulsocodemodulatie) bestanden, van 16 bits per monster.

Zo kun je bestanden maken door met een Basic of C programma 16 bits sinus-monsters te berekenen, je krijgt dan een bestand dat op een CD gebrand kan worden met fluittonen van elke gewenste audiofrequentie, met de eigenschap dat de vervorming 1 bit per monster is, dus in de buurt van 3/1000 van 1 procent ligt bij volle uitsturing. Zogezegd dus een bron van onverdachte kwaliteit.

Elke WAV file bevat een header, die moet

kloppen met de inhoud, die geeft de grootte van het bestand aan, de codering (PCM) het aantal kanalen het aantal bytes per monster en dat soort zaken.

De samenstelling is te vinden op: <https://ccrma.stanford.edu/courses/422/projects/WaveFormat/>

Als dat zelfmaken te bezwaarlijk is, dan is er nog een mogelijkheid om wat WAV bestanden die ik op Internet zet te downloaden, om daar je proefjes mee te doen. WAV bestanden zijn snel erg groot, maar de SDR-programma's hebben de mogelijkheid om een gewoon wav bestand herhaald af te spelen zodat je toch wel ermee uit de voeten kunt ook al heeft het bestand maar een luttel aantal seconden speelduur.

Een seconde geluid levert bij tweekanaals 48000 monsters van 16 bits per seconde 192 kbyte op, dus dat gaat snel groot worden.

## Proef 1

Wat we hier gaan doen is een geluidsbestand maken dat in de twee kanalen het I en het Q signaal bevat van een simpele draaggolf.

Zoals gezien moeten we daarvoor 2 kanalen, namelijk links en rechts, I en Q, maken die precies 90 graden in fase schelen, de ene vullen we dus met een cosinus en de andere met een sinus. Ik heb voor 6000 Hz gekozen, zodat je in de panoramische frequentiedisplay EEN draaggolf moet gaan zien op 6000 Hz afstand van de frequentie  $f_0$ .

Na twee seconden keer ik de fase van het rechter Q kanaal om, zodat bij dezelfde toonhoogte nu het signaal aan de andere kant van de nul frequentie moet verschijnen. De totale speelduur is 4 seconden.

Spiegels op 12 kHz afstand richting  $f_0$  en harmonischen mag je niet zien op meer dan 96 dB beneden het signaal, want de signalen zijn zuiver tot dat niveau, maar die zul je toch wel zien en dat is dan een afwijking van de software of de geluidskaart die je zo vaststelt.

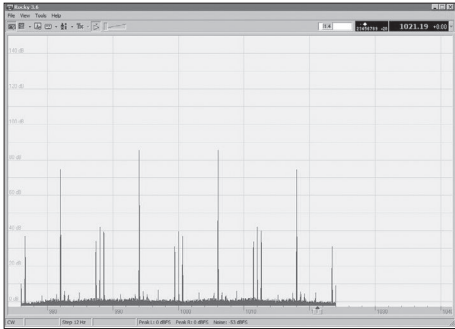


Proefro1\_6kHza.

Welnu, tegen alle verwachting in is het resultaat de plaat proefro1\_6kHza. Het cijfer slaat op het proefnummer en ro op rocky.

Dat lijkt werkelijk nergens naar: het hele spectrum is vol bezet.

De wav file proef1\_6kHz.wav en het programma dat hem maakt onderzocht, mankeert niks aan. Vervolgens geprobeerd met 500 Hz en hetzelfde verder ongewijzigde programma, dat werkte redelijk.



Proefro1\_6kHzb.

De platen proefro1a.jpg en proefro1b.jpg laten het spectrum zien voor de eerste en de tweede 2 seconde van het bestand dat Rocky3.6 toont onder Windows XP.

Je ziet een redelijke onderdrukking van de spiegel, maar gelet op de kwaliteit van de bron zou dat veel meer moeten zijn.

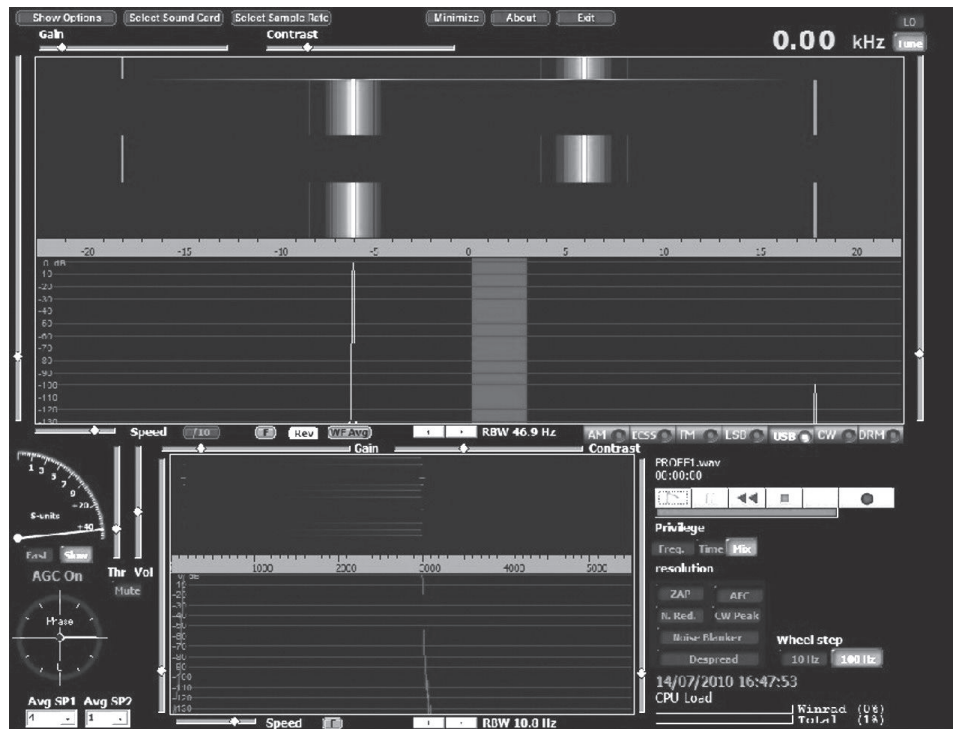
Nadenken leidt ertoe, dat het vermoeden ontstaat dat het programma de wav file met 8 monsters per sinus (6000 Hz) niet kan verwerken, omdat de monsters niet door het programma zijn aangemaakt en eerder opgenomen maar op een andere manier mogelijk met faseverschuiving t.o.v. de gewenste positie is opgenomen. Dat mag vanzelfsprekend geen invloed hebben, maar het is wel de enige mogelijkheid die ik kon verzinnen. Dat is uit te vinden door proef1\_6kHz.wav af te spelen op een andere computer en de lijnuitgang van diens geluidskaart te gebruiken voor input van de line ingang op de Rocky draaiende computer. En zowaar dat werkt.

Echter er zijn harmonischen te zien op 12 kHz en de zijbandonderdrukking is nihil. Tot slot is elke 6 kHz component vergezeld van een paar zijbanden van onbekende oorsprong. Plaat proefro1\_6kHzb.jpg toont het resultaat, als centraalfrequentie  $f_0$  is 1 MHz gekozen in de schaalverdeling.

Geen zijbandonderdrukking duidt op onderbreking van een kanaal, of van een ernstige fase en/of amplitudeverschil op 6 kHz. Gelet op de kwaliteit van het wav-bestand zijn de harmonischen ook niet op voor de hand liggende wijze te verklaren.

De oorzaak is niet meer zo eenduidig aan te wijzen als een andere computer het bestand afspeelt en het via een lijntje naar de Rocky draaiende computer gaat. Als je je testsignalen niet zuiver hebt, hebben verdere metingen nauwelijks zin.

Rocky heeft, volgens de toelichting op de www.dxatlas.com website, nogal wat no-



Proefwr1.

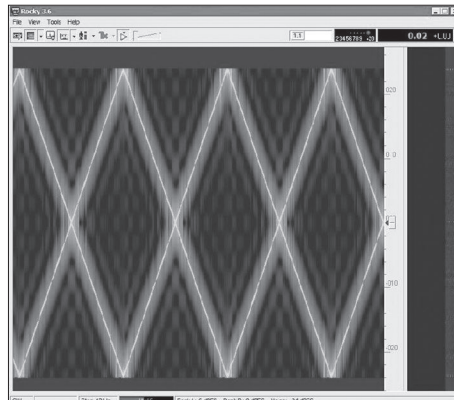
ten op de zang wat betreft het operating system. Windows 95 en 98 werken niet. Het moet Windows 98SE zijn. 98GE werkt ook niet, verder werkt XP, die ik dan gebruik, en Vista werkt weer niet, wat ik al gemerkt had.

Een en ander is aanleiding om WinRad 1.6.1 te gaan downloaden van www.winrad.org en dat te installeren. Vista levert geen probleem.

Winrad staat je ook toe een IQ WAV file periodiek af te spelen.

De resultaten zijn frappant. Een schermbeeld proefwr1 toont een momentopname; in de watervaldisplay zie je dat de 6 kHz wisselend gedurende 2 seconden op 6 kHz boven en 2 seconden later op 6 kHz beneden de  $f_0$  verschijnt. De ongewenste zijband op 12 kHz afstand in de richting van  $f_0$  ligt zoals theoretisch maximaal te verwachten met 16 bits codering op ruim 90 dB down.

Een opluchting na zoveel tegenslag met de software PowerSDR-IQ en Rocky3.6.



Proefro2.

## Proef 2

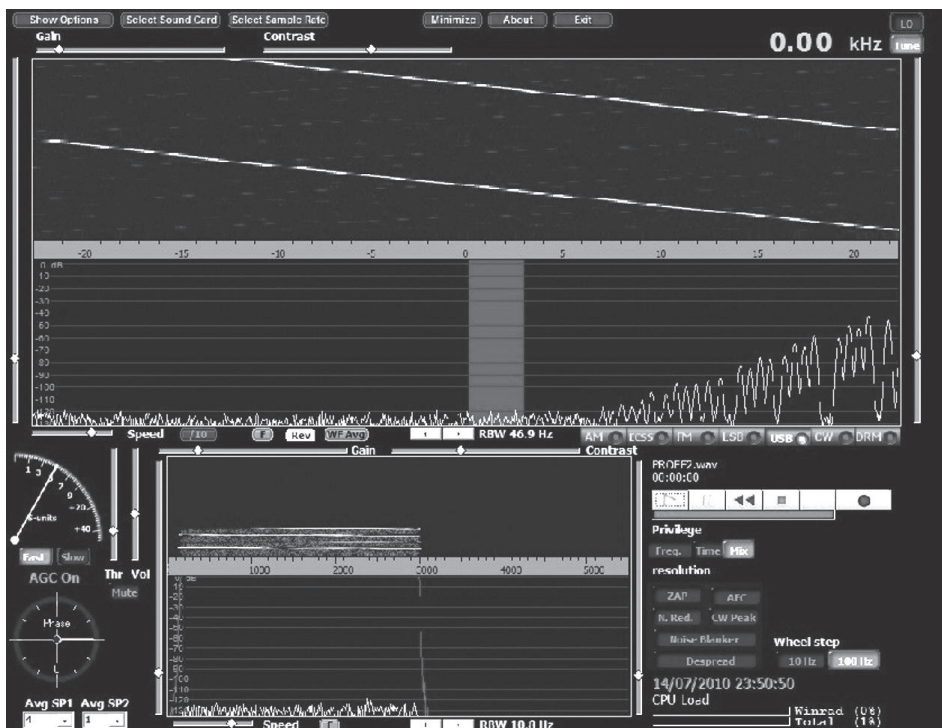
Proef 2 bevat de wav file een signaal van precies 24 kHz, dat langzaam gedurende 2 seconden daalt tot 0 Hz, en vervolgens de fase van het tweede kanaal omkeert en weer omhoog loopt in 2 seconden tot 24 kHz. Je kunt dan zien in hoeverre de ontvanger op tilt gaat en spiegels of andere rommel vertoont die er niet thuishoort. Het signaal behoort in het ideale geval dan te lopen van  $f_0 + 24$  naar  $f_0$  en dan niet terug naar  $f_0 + 24$  maar doorlopen naar beneden tot  $f_0 - 24$ , zoals we in de theorie konden voorspellen.

Het bestand is gemaakt, en te vinden op proef2.wav. De resultaten zijn zoals verwacht na de ervaring met proef 1 niet denderend bij Rocky. Ik merk tevens op dat door het dubbele patroon in de waterval proefro2 de ongewenste zijbandonderdrukking kennelijk erg laag is. De uitsturing van de signalen is in alle gevallen totaal nooit meer dan 98% gekozen om de programmatuur niet tot het uiterste qua amplitude te belasten.

Bij Winrad proefwr2 gaat de watervaldisplay in de vorm van een zaagtand, dat alles loopt zoals het behoort te lopen. Rocky doet dat, behoudens de lage zijbandonderdrukking ook wel, maar die toont duidelijk een moire nevenpatroon van spurious responses in de waterval-display.

De panorama-display van Winrad, heeft moeite het signaal te volgen, dat zou langzamer moeten lopen dan de huidige 12 kHz per seconde.

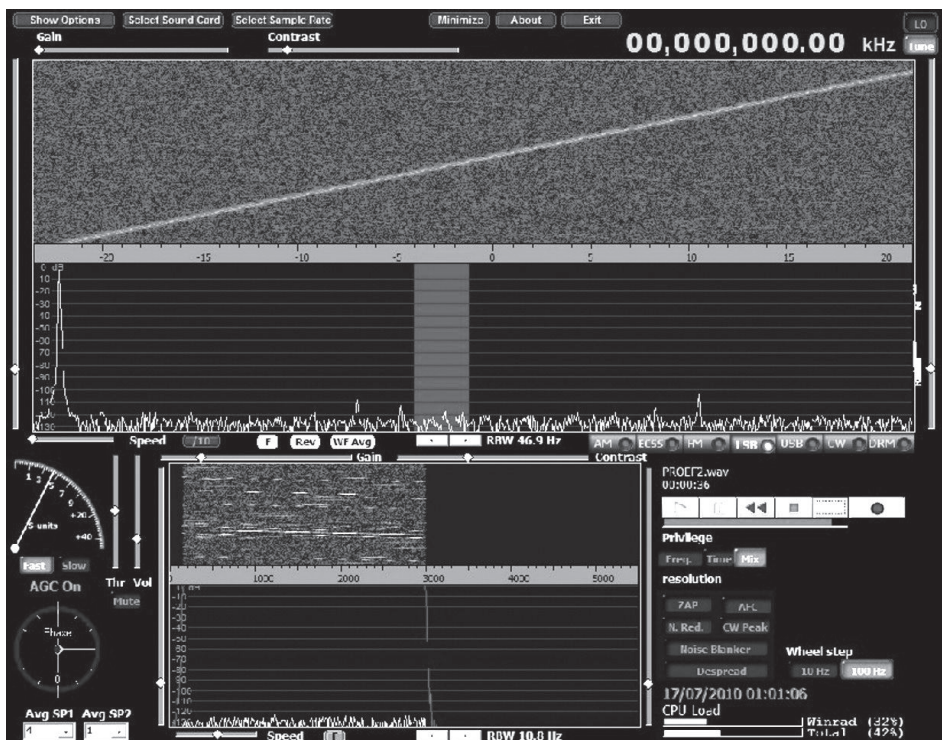
Een fors wav bestand, dat er 10 keer zo lang overdoet (40 seconden) en verder identiek is, staat ook op de website onder de naam proef2slow.wav.



Proefwv2.

Dat geeft fraaie resultaten, je ziet de ongewenste spurios op  $-90$  dB over het panoramascherm elkaar tegemoet hollen. Proefwv2slow toont een plaatje van die toestand.

ruisspanning is 90% van de volle uitsturing, de piekwaarde van het signaal mag dus maximaal 10% van de volle uitsturing zijn, teneinde oversturing te voorkomen. Bij een sinusvormig signaal is de effectieve



Proefwv2slow.

### Proef 3

Proef 3 bevat een bestand proef3.wav met random getallen als monsters, dat is dus witte ruis met een bandbreedte van 24 kHz, en daarin verzonken op 1000 Hz een CW signaal met een snelheid van 12 wpm. Met de beschikbare filters kun je kijken in hoeverre je dat signaal beter boven de ruis uit kunt krijgen. Die piekwaarde van de

waarde de piekwaarde gedeeld door  $\sqrt{2}$ . Bij ruis komen alle mogelijke spanningsniveaus even vaak voor en daarom is de effectieve waarde van de ruis de topwaarde gedeeld door  $\sqrt{3}$ .

De signaalruisverhouding van de CW in de band van 48 kHz breed is dus:  
 $S/N = ((c_w \text{ amplitude})^2/2) / ((\text{ruis amplitude})^2/3)$  (4)

Die signaalruisverhouding verbetert door het wegsnijden van ruis met andere frequenties omgekeerd evenredig met de bandbreedte die je filter doorlaat. Een 100 Hz breed filter levert dus een 240 keer betere signaalruisverhouding oftewel bijna 24 dB.

Bij de maximale CW amplitude van 10% van de ruispiekamplitude is de signaalruisverhouding bij 24 kHz bandbreedte  $-17$  dB volgens bovenstaande formule (4), bij 100 Hz bandbreedte wordt dat dus  $-17+24=7$ dB.

Nu is het natuurlijk wel zo dat een filter de signaalruisverhouding verbetert, maar omdat je oren geen 48 kHz breed zijn, werken die al als filter.

Bovendien zijn de oren voor de verschillende ruisfrequenties in het hoorbare gebied verschillend gevoelig en moet je daar qua perceptie van de CW dus ook rekening mee houden.

Psfometrische weging heet dat. Neem dus maar 3 kHz bandbreedte.

De signaalruisverhouding volgens de gegevens is dan  $-17+9=-8$  dB op 3 kHz bandbreedte.

Verbetering treedt al automatisch op, omdat de hersenen een smalle band om de gewenste toonhoogte gooien, je hebt dus ook nog een hersenfilter bij CW.

Of deze redenering helemaal klopt weet ik niet zo, ik zit hem hier ter plaatse te bedenken en ik ben zeker geen expert op dat gebied.

Hoe het zij, de proef 3 is prima geschikt voor vergelijkende metingen met diverse typen SDR programma's. Een schermbeeld staat in proef3.

### Proef4

Proef 4 bevat de wav file proef4klik.wav twee CW signalen een sterke op 1000 Hz van  $f_0$  en een 50 dB zwakkere op 500 Hz afstand van  $f_0$ . De sterke schakelt zijn signaal abrupt in en uit en dat geeft sleutelklik over de hele band zichtbaar.

Door dat bestand af te spelen kun je de filterwerking van het CW filter kijken of je die signalen voldoende kunt scheiden om de zwakke te kunnen nemen. Schermbeeld van winrad staat in proef4klik.jpg.

Vervolgens is de sterke wat aan zijn signaalflanken gaan doen om de klik te verminderen, die signalen vind je in proef4.wav en proef4wv.jpg.

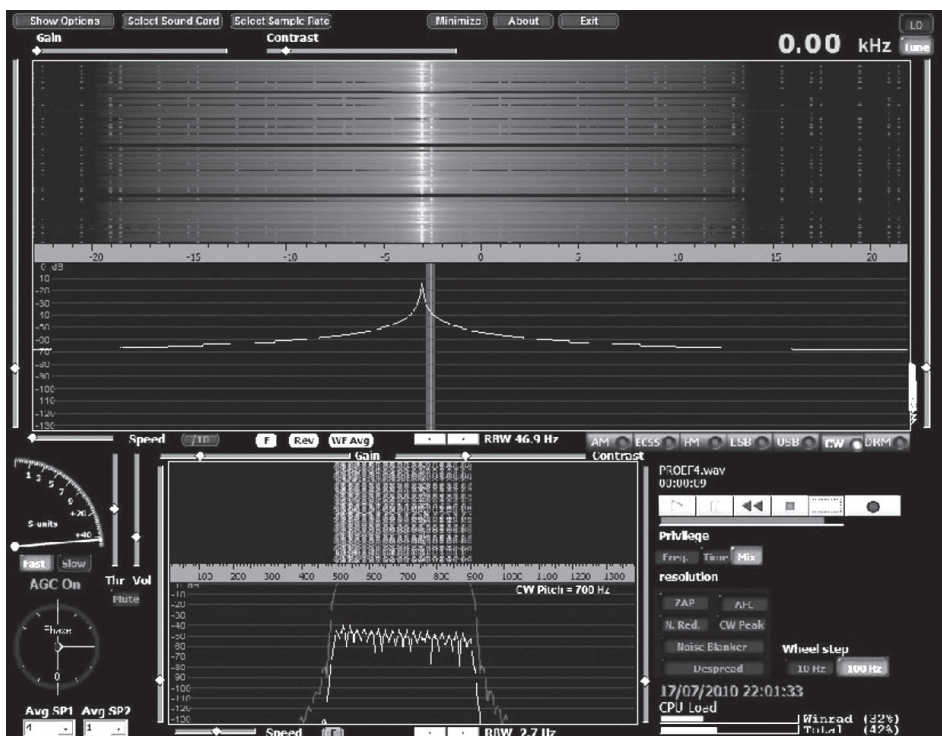
De opflank en downflank zijn cosine squared en nemen elk 10% van de dotlengte in beslag, winrad kan het zwakke signaal nemen in het CW filter, Rocky bakt er niks van.

### De WCGG

Zo kun je als je zelf WAV bestanden maakt, allerlei proefjes bedenken, die het leven veraangenamen en leerzaam zijn, vooral als de verschijnselen anders zijn dan je verwacht.



Proef3.



Proef4klik.

De WVCG is een programma dat draait in een DOS box. Niet na downloaden vanuit explorer starten door aanklikken dus, maar eerst een dos prompt starten. Als parameter eist het dat je na de naam van het programma tussen " " je call opgeeft, dus bijvoorbeeld WVCG "de PA0BLAH".

Als je dat goed doet genereert het programma een WAV file die 7 tonen bevat, die op zodanige gekozen tijdstippen worden in- en uitgeschakeld dat een watervaldisplay over 2,7 kHz bandbreedte je call weergeeft. Ook makkelijk te proberen met Rocky.

Grappig is dat als je Rocky opdraagt niks

te corrigeren, de zijbandonderdrukking oneindig is, maar laat je hem toe te corrigeren dan komt je call vaag in spiegelbeeld aan de andere kant van je onderdrukte draaggolf te staan.

Het bestand werkt door het in Rocky zelf te draaien, maar gelet op de resultaten en ervaring met proef1 is dat geen goed kwaliteitscriterium. Als je het geluid via je SSB zender afspeelt moet je op een watervaldisplay je call netjes zien staan, dat kan bijvoorbeeld gebeuren als je je eigen signaal kunt zien op de website SDR-ontvanger van TUTwente

<http://websdr.ewi.uttwente.nl:8901/>

Er zijn twee wav bestanden gemaakt bij deze proef, de tonen worden in het ene geval in- en uitgeschakeld met een transient die cosine squared is en 3% van de dotbreedte in beslag neemt, in het andere geval is dat 30% van de dotbreedte.

De genoemde programma's, WAV files en schermbeldden, kun je tijdelijk allemaal vinden op mijn website <http://www.xs4all.nl/~pa0wv/SDR/> met naar onderwerp gesorteerde subdirectories.

55 es 73  
de PAoWV

## Zoeken in oude kranten

Op het internet zijn veel interessante websites te vinden, één daarvan is de website van de Koninklijke Bibliotheek - de Nationale Bibliotheek van Nederland. De Koninklijke Bibliotheek is eind 2006 gestart met het project Databank Digitale Dagbladen. Het project digitaliseert op grote schaal Nederlandse nationale, regionale, lokale en koloniale dagbladen en stelt deze gratis beschikbaar op internet.

Het gaat in totaal om acht miljoen pagina's, vanaf de eerste krant uit 1618 tot aan kranten uit de twintigste eeuw. Een webdienst is opgezet met geavanceerde zoekmogelijkheden voor onderzoekers en het brede publiek. De link naar het radiozendamatisme is gauw gelegd als men naar de resultaten kijkt van bijvoorbeeld de zoekopdracht: radiozendamatisme of Jesse radio. Zie: <http://kranten.kb.nl/>

Uiteraard kan men zo ook mooi aan stamboomonderzoek doen en nog veel meer andere zaken afhankelijk van de creativiteit van de zoekopdrachtgever. Het aardige is dat men ook complete kranten kan opvragen, downloaden en uitprinten.

Overigens als men de normale pdf-lezer van Acrobat te traag vindt werken, dan moet men het eens proberen met het freeware-programma PDFXchange Viewer, zie <http://pdfxchange-viewer.en.softonic.com/> Dit programma werkt aanzienlijk sneller!

73, Willem PE7WFS