

GnuRadio, Software Defined Radio voor iedereen.

Martin Dudok van Heel
PA1SDR@olifantasia.com

Met gnuradio koop je geen nieuwe radio-set,
je bouwt hem niet, maar je schrijft of
download hem (programmeert hem)

Wie ben ik ?

- ◆ Martin Dudok van Heel
- ◆ PA1SDR (voorheen PE1LVC)
- ◆ elektronicus / hardware ontwerper
- ◆ freelance Software Defined Radio en FPGA/verilog/C++ programmeur
- ◆ hacker
- ◆ wereldverbeteraar
- ◆ binnenkort importeur/verkoper USRP en andere Software Defined Radio producten

Waar ga ik het over hebben ?

- ◆ Motivatie: de wereld verbeteren
- ◆ Software Defined Radio
 - ◆ wat is het
 - ◆ waarom
 - ◆ hoe werkt het
- ◆ Gnuradio
 - ◆ wat is het
 - ◆ visie en doelen
 - ◆ hoe werkt het
 - ◆ Hoe kan ik er mee aan de slag
- ◆ Toepassingen
 - ◆ voor radio amateurs
 - ◆ andere toepassingen

De wereld verbeteren

- ◆ Grote bedrijven
 - ◆ Bedrijven zijn producenten
 - ◆ Individuen zijn consumenten
 - ◆ Wij bepalen wat mensen met hun apparatuur, content, radiospectrum en het internet willen, kunnen en mogen doen
 - ◆ Mensen dom houden
- ◆ Wereldverbeteraar
 - ◆ Individuen zijn producenten en consumenten
 - ◆ Individuen kunnen bij bedrijven werken
 - ◆ Mensen bepalen zelf wat ze met hun apparatuur, content, radiospectrum en het internet willen, kunnen en mogen doen
 - ◆ Mensen willen kennis delen

Doel Software Defined Radio (SDR)

- ◆ Flexibiliteit
- ◆ Kostenreductie
- ◆ Toekomstvast
- ◆ Het ontwerp = het apparaat
 - ◆ kennis delen = het apparaat delen

Wat is Software Defined Radio (SDR)

- ◆ Een Software Radio is een draadloos communicatie-apparaat dat al zijn signaalverwerking in software doet.
- ◆ Software definieert de uitgezonden golfvorm (waveform), en software demoduleert de ontvangen golfvorm.
- ◆ vergelijk “gewone” HF-apparatuur
 - ◆ signaalverwerking door analoge circuits
 - ◆ signaalverwerking door digitale circuits

De Ideale Software Radio

- ◆ Ontvanger
 - ◆ Antenne
 - ◆ Analooq naar digitaal omzetter (ADC)
 - ◆ Computer met radio-ontvangprogramma
 - ◆ Luidspreker en/of beeldscherm
- ◆ Zender
 - ◆ Microfoon of camera
 - ◆ Computer met radio-zendprogramma
 - ◆ Digitaal naar analooq omzetter (DAC)
 - ◆ Antenne

Analoog naar digitaal omzetting

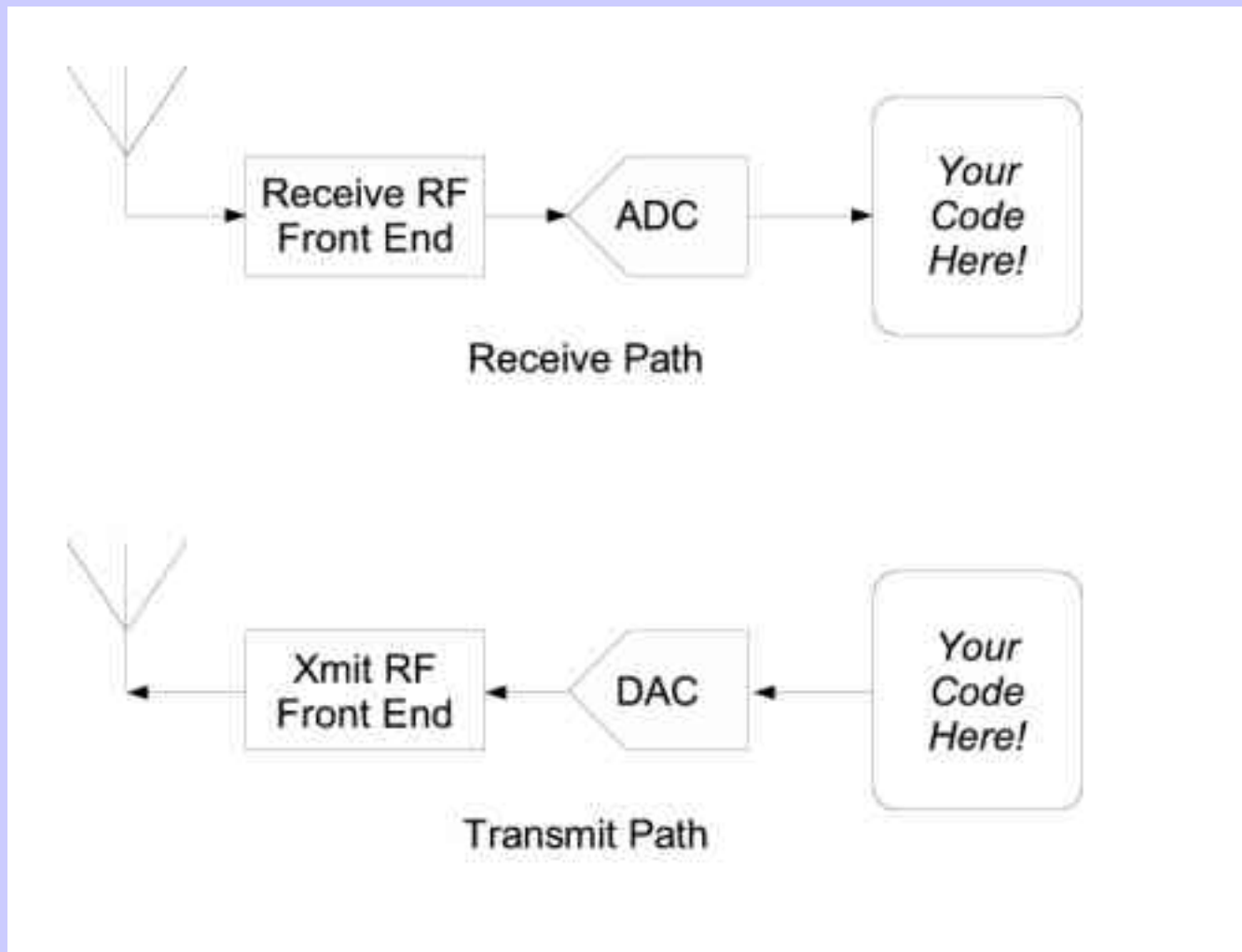
- ◆ heel snelle digitale multimeter
- ◆ meet vaak per seconde de signaalsterkte
 - ◆ samplerate = aantal keer per seconde dat signaal gemeten wordt
- ◆ verwerk deze getallen digitaal (enen en nullen, 5 Volt, 0 Volt)
- ◆ niquist theorema
 - ◆ samplerate $\geq 2 \times f_{\max}$
- ◆ voorbeeld: geluidskaart in computer
 - ◆ samplerate_max 96 kHz \implies f_{\max} 48 kHz

Digitaal naar analoog omzetter

- ◆ heel snelle digitaal instelbare voeding
- ◆ stel vaak per seconde de uitgangsspanning in op de waarde aangegeven door het huidige getal (digitale waarde)

- ◆ voorbeelden:
 - ◆ geluidskaart uitgang (fmax 48 kHz)
 - ◆ cd speler (fmax 22.5 kHz)
 - ◆ videokaart (fmax 200 Mhz)
 - ◆ tempest voor eliza

Software radio in de praktijk



RF front end

- ◆ versterker (optioneel)
 - ◆ zonder versterker alleen sterke signalen te ontvangen
- ◆ filter (optioneel)
 - ◆ ontvang alleen de band die je wilt, vergroot dynamisch bereik
- ◆ frequentieomzetter (optioneel)
 - ◆ gebruik SDR alleen voor IF
 - ◆ ADC's voor meer dan 300 Mhz zijn duur/moeilijk verkrijgbaar/minder dynamisch bereik

Waarom Software Radio?

- ◆ Met Software Radio kun je met één apparaat communiceren met vele verschillende systemen.
- ◆ Het ontwerp = het apparaat
- ◆ Je kunt hem herprogrammeren als een mobiele telefoon, portofoon, draadloze netwerkkaart, radardetector, fm-radio, digitale TV-ontvanger/zender of een garagedeuropener.
- ◆ Een SDR kan opgewaardeerd worden naar toekomstige telecomstandaarden.

Waar is Software Radio goed voor?

- ◆ Experimenteren
- ◆ Snel prototypes bouwen
- ◆ Makkelijk opwaarderen
- ◆ Kosten reductie
- ◆ Toekomstvast
- ◆ Kan veel meer mee dan met alleen hardware

Nadelen Software Defined Radio

- ◆ Rekenkracht
- ◆ (digitale) bandbreedte
- ◆ doorlooptijd signalen
 - ◆ teruggekoppelde systemen liefst in 1 blok plaatsen

TAPR debacle

- ◆ TAPR (Amerikaanse Radio Amateurs organisatie) wilde een FHSS (frequency hop spread spectrum) radio bouwen,
- ◆ “heel handig vet coole chips”
- ◆ > 10 onderdelen uit de handel genomen
- ◆ Klein budget (geen multinational).
- ◆ her-ontwerp voor nieuwe chips,
- ◆ systeem verouderd
- ◆ project mislukt.

Nieuwe zich ontwikkelende protocollen en modulaties

- ◆ WCDMA (UMTS) protocol
 - ◆ specificaties 240 keer gewijzigd binnen 2 jaar
 - ◆ technische en politieke redenen

Hou het generiek !

- ◆ ADC/DAC zo dicht mogelijk bij de antenne
- ◆ Hardware
 - ◆ Zo min mogelijk
 - ◆ Zo algemeen mogelijk
 - ◆ Zo simpel mogelijk
 - ◆ Zo goedkoop mogelijk
 - ◆ Zo flexibel mogelijk
- ◆ Doe alles wat complex is in software.

Kwadratuur

- ◆ complexe signalen
- ◆ Real sinus signaal
- ◆ Imaginair cosinus signaal +90 graden
- ◆ perfect mixen (frequenties omzetten)
 - ◆ geen filter nodig alleen vermenigvuldigen
- ◆ mathematisch veel eenvoudiger
 - ◆ am demodulatie is magnitude(complex signaal)
 - ◆ negatieve frequenties en positieve frequenties
- ◆ directe frequentie en fasebepaling
- ◆ faseverschuiven is vermenigvuldigen met constante

Wat is GnuRadio ?

- ◆ Een gratis opensource gereedschapskist om software radios te bouwen.
- ◆ Een platform om te experimenteren met digitale en analoge communicatie
- ◆ Een platform voor signaalverwerking
- ◆ Een hulpmiddel voor het snel bouwen van prototypes.

Vrije software en GNU

- ◆ Vrij als in vrijheid
- ◆ Vrijheid om programma te gebruiken zoals jij dat wilt voor elk doel.
- ◆ Vrijheid om programma te wijzigen voor eigen behoefte. (broncode beschikbaar)
- ◆ Vrijheid om kopieën te distribueren
- ◆ Vrijheid om gemodificeerde versies te distribueren zodat de gemeenschap van je verbeteringen gebruik kan maken

Vrije hardware

- ◆ ontwerp, schema, printlayout, specificaties van de hardware vrij beschikbaar onder een vrije licentie
- ◆ bijbehorende software opensource onder een vrije licentie
 - ◆ Dit geldt ook voor de firmware van programmeerbare chips
 - ◆ Ontwerp “hardware” en upload het naar de programmeerbare chips.

GnuRadio visie en doelen

- ◆ Zend en ontvang elk type signaal
- ◆ Creëer een omgeving voor experimenten en het maken van producten.
- ◆ Bouw communicatiesystemen die nog nooit eerder zijn gebouwd.
- ◆ Creëer een goedkoop educatiemiddel
- ◆ Breng “vrije software” naar nieuwe gebieden
- ◆ Breng software mensen in contact met radio en radio mensen in contact met software

"Hello World" Voorbeeld

- ◆ Dialtone (kiestoon)

Hello World

```
#!/usr/bin/env python
from gnuradio import gr
from gnuradio import audio

class my_graph(gr.flow_graph):

    def __init__(self):
        gr.flow_graph.__init__(self)

        sample_rate = 48000
        ampl = 0.1

        src0 = gr.sig_source_f(sample_rate, gr.GR_SIN_WAVE, 350, ampl)
        src1 = gr.sig_source_f(sample_rate, gr.GR_SIN_WAVE, 440, ampl)
        dst = audio.sink(sample_rate)
        self.connect(src0, (dst, 0))
        self.connect(src1, (dst, 1))

if __name__ == '__main__':
    try:
        my_graph().run()
    except KeyboardInterrupt:
        pass
```


Hoe moeilijk is het

- ◆ Programmeertaal Python
- ◆ verbind blokjes
- ◆ geen C++ kennis nodig
- ◆ niet compileren
- ◆ Lego bouwdoos

Wat hebben we al bereikt?

- ◆ Grote brede gebruikersgemeenschap (>1000)
- ◆ Zend en ontvang meest gangbare analoog gemoduleerde signalen
 - ◆ (AM, FM, FM stereo,SSB, etc.)
- ◆ Reeds enkele digitale modulatievormen
 - ◆ ATSC (HDTV) zender/ontvanger (reden: broadcast flag)
 - ◆ packet radio (GMSK)
- ◆ gebruik voor radio astronomie
- ◆ gebruik als meetinstrument

Waar zijn we mee bezig ?

- ◆ Analoge TV
- ◆ DVB-T (digitale TV van KPN, digitenne)
- ◆ Passive Radar
 - ◆ geen radarstation nodig, gebruik signalen van FM radio, analoge TV en GSM zenders.
- ◆ WIFI
- ◆ GPS

Waar werkt het op ?

- ◆ Draait op vrijwel elk operating system
 - ◆ Linux
 - ◆ MAC OS X
 - ◆ Windows
 - ◆ Solaris
- ◆ Werkt samen met vrijwel elk type hardware waar drivers voor zijn
 - ◆ geluidskaart, (hacked) tv-kaart, measurement ADC cards MC4020, USRP, SSRP, binnenkort PCHDTV3000)

Hardware voor GnuRadio

- ◆ Gewone PC
- ◆ RF front end
 - ◆ minimum, alleen een draadje
- ◆ Analooq/digitaal omzetter
 - ◆ geluidskaart
 - ◆ USRP of data acquisitie kaart
 - ◆ elke hardware met een driver
- ◆ Off-air source/sinks (als je geen hardware hebt)
 - ◆ bestanden http://www.kd7lmo.net/ground_gnuradio_ota.html
 - ◆ netwerk bronnen (TCP/UDP/Ip sockets)
 - ◆ alles waar je een driver voor hebt of schrijft.

Pauze

Universal Software Radio Peripheral (USRP)

- ◆ soort digitale middenfrequent (IF)
- ◆ verwisselbare RF secties
- ◆ 4 ingangen (RX), 4 uitgangen (TX)
- ◆ USB 2
- ◆ FPGA (programmeerbare chip)
- ◆ complex (I Q kwadratuur) of real signalen
- ◆ open source ontwerp / IP
- ◆ gezamenlijk ontwikkeld
- ◆ Lost het grootste probleem op. Het in en uit de PC krijgen van samples (signalen)
 - ◆ De ADC en DAC

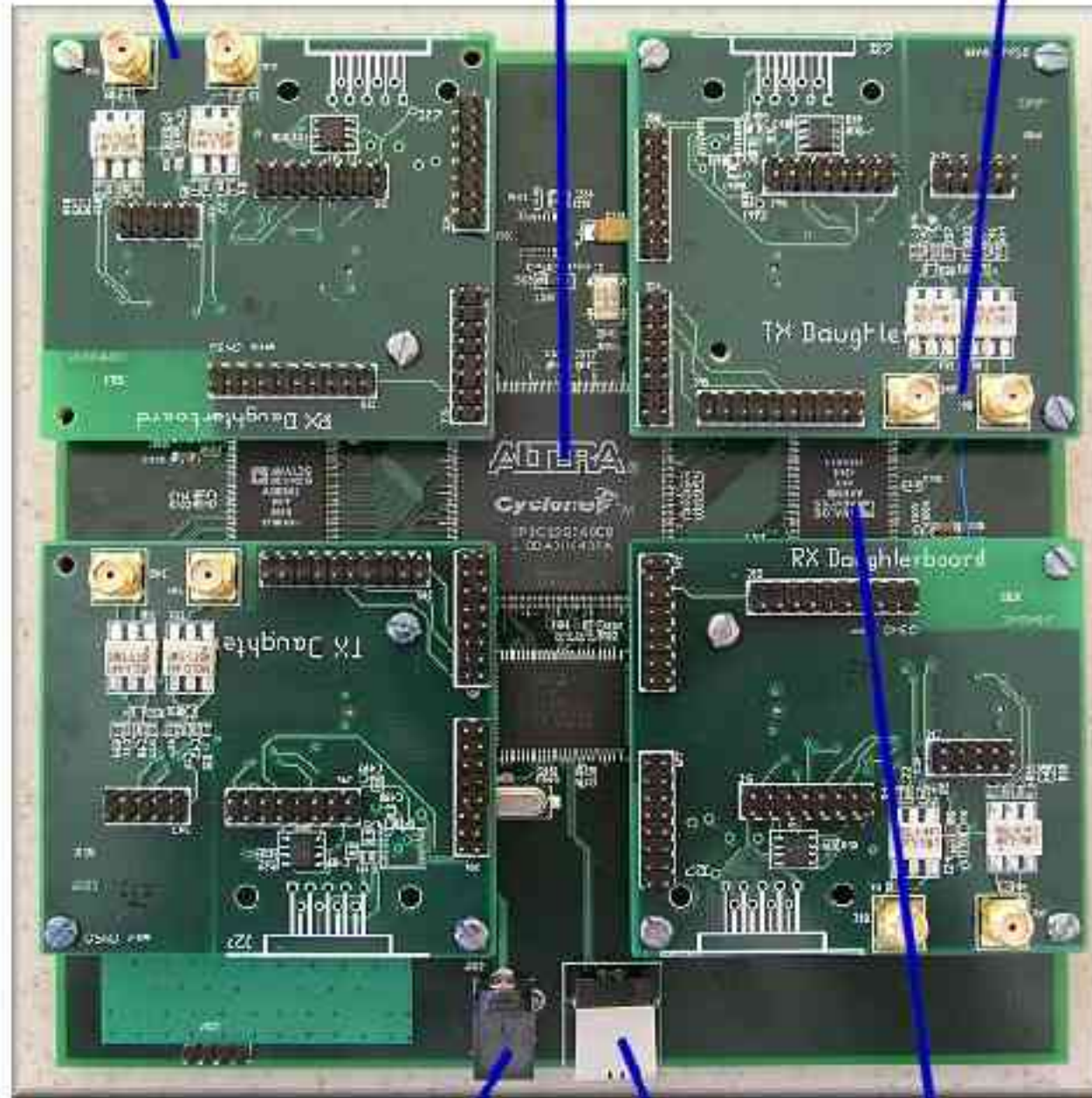
USRP eigenschappen

- ◆ verwisselbare RF frontend secties
- ◆ zeer grote bandbreedte
- ◆ extreem flexibel
 - ◆ programmeerbaar
 - ◆ uitbreidbaar
- ◆ beam forming/Smart Antenna/MIMO/BLAST
- ◆ multipath
- ◆ gebruikersgemeenschap die ook weer functies toevoegd.

Receive Channel
RF Interface

Altera FPGA

Transmit Channel
RF Interface

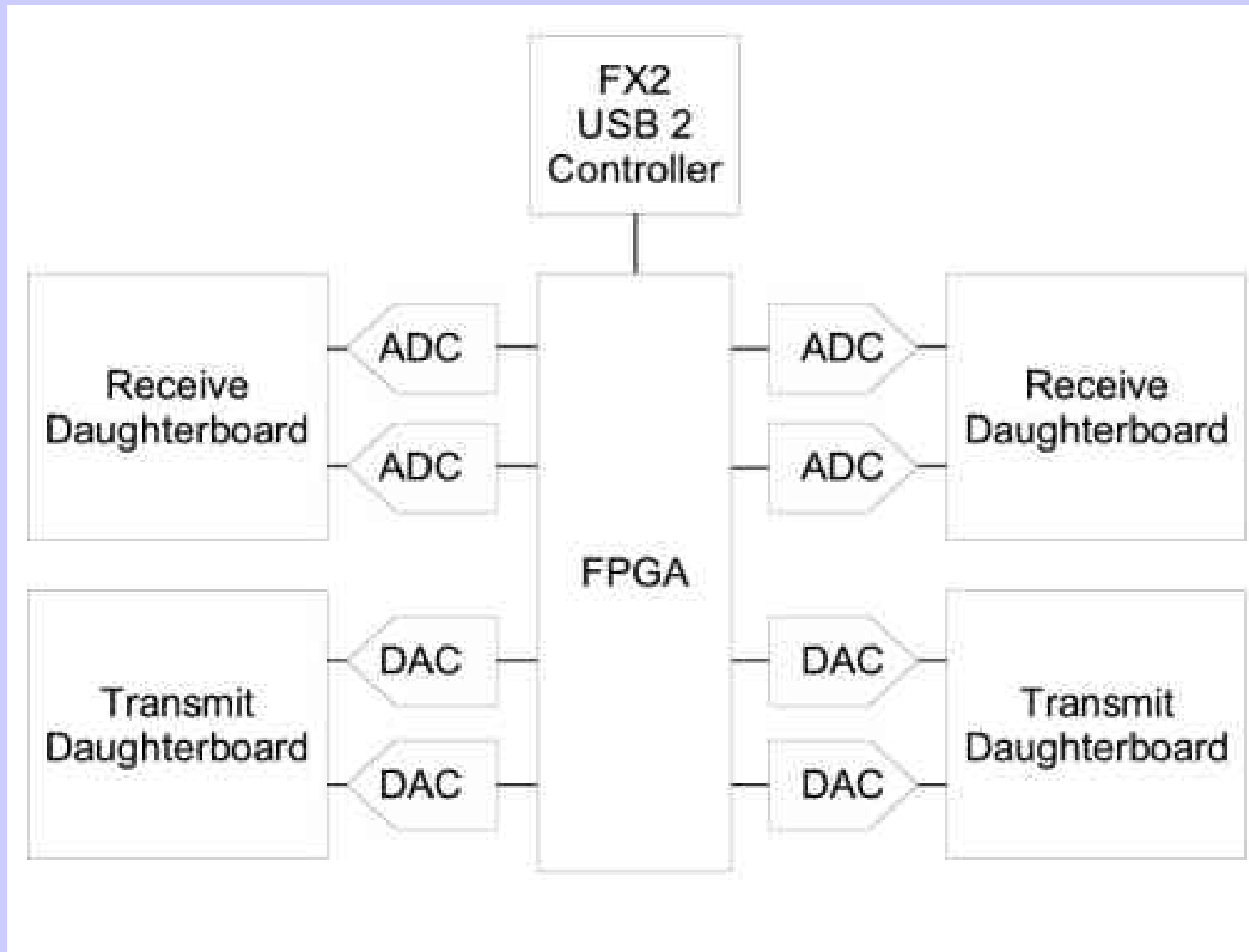


DC Power

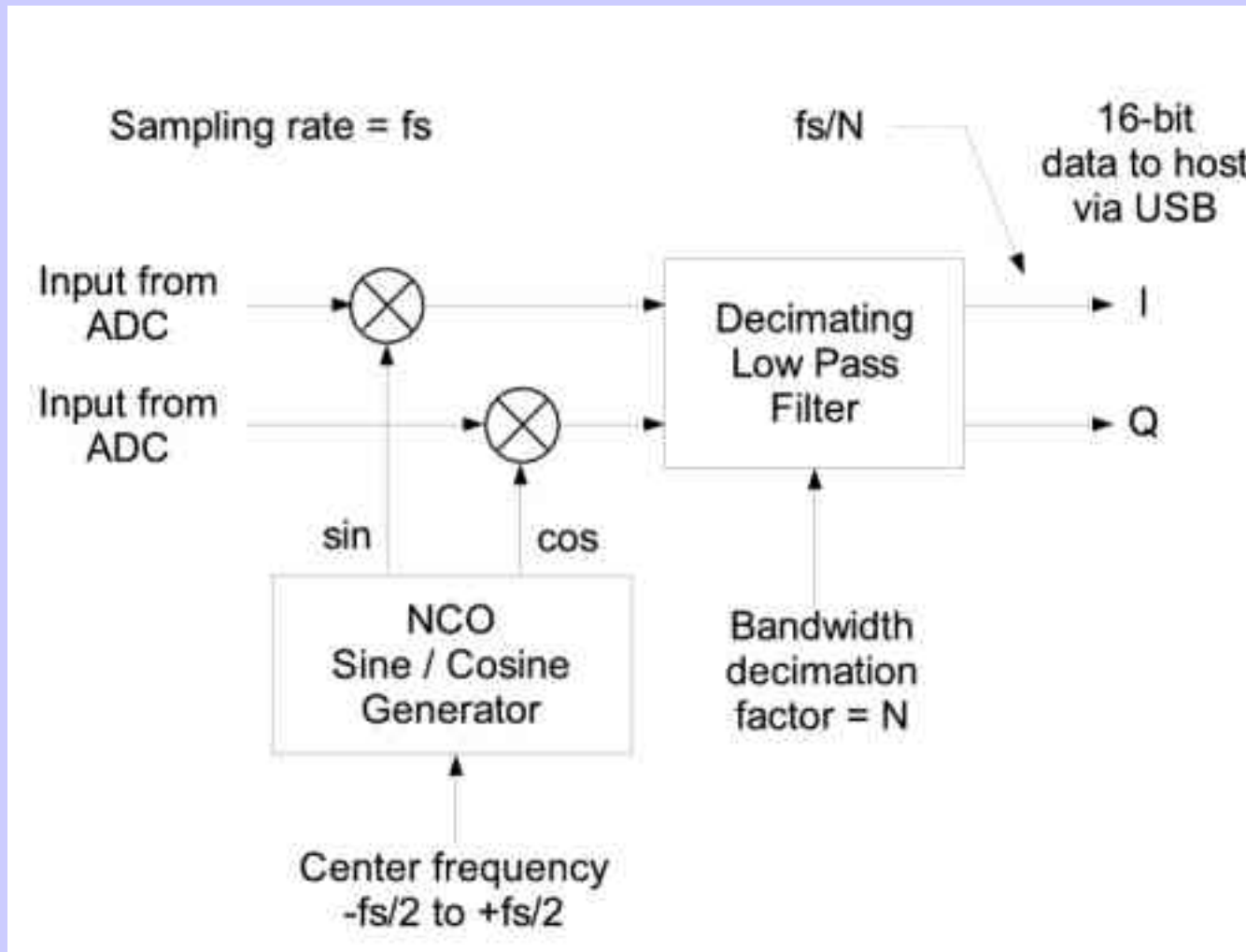
USB 2.0
Port

Analog Devices
Mixed Signal
Processor

USRP blokdiagram



FPGA functions



USRP werking

- ◆ dochterbord = RF frontend
- ◆ FPGA
 - ◆ frequentie omzetten
 - ◆ decimeren, interpoleren
 - ◆ filteren
- ◆ Ontworpen in PADS, rest in pcb
- ◆ Alles in de hardware is modulatie onafhankelijk.
 - ◆ Kan wel er in maar maakt minder flexibel

USRP specificaties

- ◆ USB 2.0 interface
 - ◆ up to 480 Mbits/sec
- ◆ Quad 128 MS/s 14-bit DACs
- ◆ Quad 64 MS/sec 12-bit ADCs
- ◆ FPGA
 - ◆ digital up and Down converters (frequentie omzetter)
 - ◆ decimeren en Interpoleren (samplerate veranderen)
 - ◆ functionaliteit die je er zelf in programmeert
 - ◆ alle signaalverwerking op meer dan 16 bit precisie
 - ◆ non-volatile storage (EEPROM) on board
 - ◆ (upload firmware voor fpga en USB chip)
- ◆ 16 bits complex geeft 8 Mhz bandbreedte
- ◆ 8 bits > 16 Mhz
- ◆ radio astronomie 1 bit > 128 Mhz

Dochterbord

- ◆ Generieke dochterborden
 - ◆ voor eigen rf sectie of alleen HF 0- 30 Mhz
 - ◆ 0 – 300 Mhz (niquist mirroring)
- ◆ TVRX 50 - 870 MHZ Rx only
- ◆ DBSRX 800 - 2400 MHz Rx only
- ◆ LFRX 0 - 40 Mhz
 - ◆ echt vanaf DC dus ook 0.001 Hz
- ◆ FLEX400 400 - 500 MHZ RX+TX
 - ◆ (100 mw TX) snel te schakelen of tegelijk.
- ◆ FLEX2400 2200 – 2700 MHZ RX+TX

dochterbordjes in ontwikkeling

- ◆ FLEX900 900 MHz RX + TX
- ◆ FLEX1200 1.2 GHz RX+TX
- ◆ 1 Watt TV Band TX

- ◆ Huidige dochterbordjes zijn ook flexibel
 - ◆ FLEX2400 is te converteren naar FLEX1200

Beschikbare software bouwblokken in GnuRadio

- ◆ Teveel om op te noemen
- ◆ mixer `gr.mult_cc()`
- ◆ filter `gr.fir_filter_cc()`
 - ◆ teken je eigen frequentiespectrum en laad het in het filter
 - ◆ standaard lowpass, bandpass, highpass, notch beschikbaar
- ◆ frequentie demodulator `gr.freq_demod()`
- ◆ am detectie `gr.mag_cc()`
- ◆ pll `gr.pll()`

beschikbare bouwblokken

- ◆ versterker `gr.mult_const()`
- ◆ DC offset `gr.add()`
- ◆ signaalgenerator
 - ◆ `gr.sig_source(SINUS,amplitude,dcoffset)`
 - ◆ `gr.noise_source(NOISE_TYPE,amplitude)`
 - ◆ `gr.vector_source(tabel met waarden in de tijd)`
 - ◆ `gr.file_source(bestandsnaam)`
- ◆ correlatie `gr.fft + gr.mult+ gr.ifft`
- ◆ `gr.message_source`
- ◆ fase verschuiver `gr.mult_cc()`

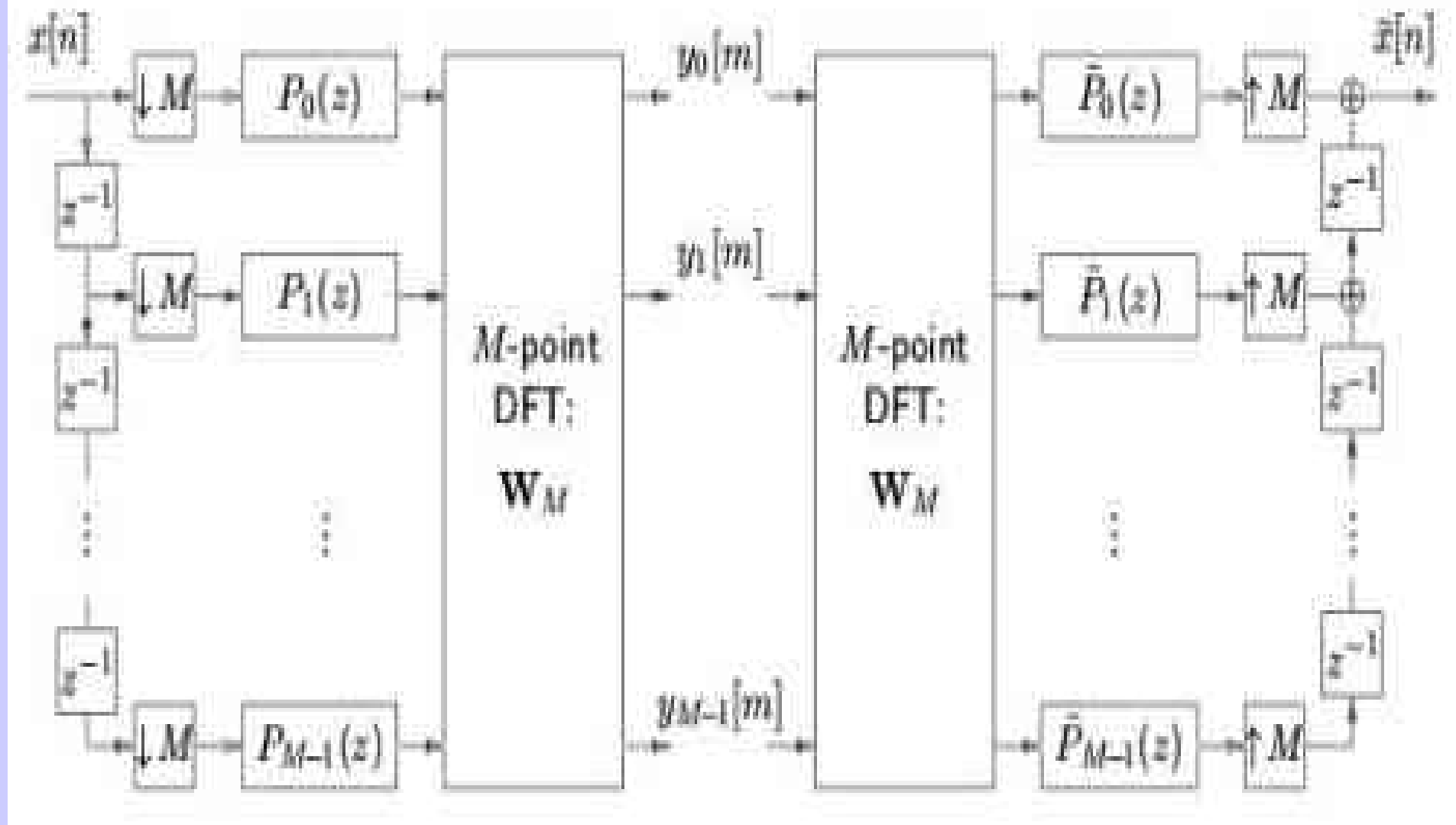
beschikbare geavanceerde bouwblokken

- ◆ `nbfm_rx` smalband fm ontvanger
- ◆ `nbfm_tx` smalband fm zender
- ◆ `wfm_rcv` breedband fm ontvanger
- ◆ `wfm_tx` breedband fm zender
- ◆ `fm_emph` FM emphasis
- ◆ `gmsk` digitale modulatie (packet radio)
- ◆ `squelch` standaard of geavanceerd
- ◆ `filterbank` meervoudig kanaalfilter
- ◆ `digital_voice` digitale stem

Filterbank

- ◆ breedband signaal in
- ◆ meerdere kanalen uit (elk kanaal heeft zijn eigen smalband filter)
- ◆ Voorbeeld 144.000 145.640 Mhz in
- ◆ 64 kanalen uit 10kHz breed
- ◆ Je kunt elk kanaal nu demoduleren of opslaan of weergeven, wat je maar wilt
- ◆ Je kunt ook 1024 kanalen filteren als je dat wilt

filterbank



Wat kost het

- ◆ setup met eigen RF frontend en smalband gebruik via geluidskaart ingang GRATIS
- ◆ Gnuradio software GRATIS
- ◆ Ontwerp/printlayout USRP GRATIS

- ◆ HDTV3000 videokaart 200 euro
 - ◆ zelf ombouwen/hacken
- ◆ USRP 500 euro
- ◆ dochterbordjes usrp 70 - 250 euro

Spectrum recorder

- ◆ `./usrp_rx_cfile.py -d 16 -f 144.0e6 -g 20 bla.raw`
- ◆ centrale frequentie 144.0 Mhz
- ◆ bandbreedte $64/16 = 4$ Mhz
- ◆ gain = 20 dB
- ◆ sla op in bestand bla.raw
- ◆ Je kunt dit achteraf beluisteren of bestuderen op dezelfde manier als met direct ontvangen signalen

Spectrum transmitter

- ◆ `./usrp_tx_cfile -i 32 -f 470.0e6 -g 20.0
bla.raw`
- ◆ Speel het 4Mhz spectrum af en zend het uit
op 470.0 Mhz

standaard Input/Output bouwblokken

- ◆ spectrum display `gr.fft_sink()`
- ◆ oscilloscope `gr.oscope_sink()`
- ◆ geluidskaart `gr.audio_sink()`
- ◆ bestand `gr.file_sink()`
- ◆ TV display (in ontwikkeling) `gr.tv_sink()`

Andere voorbeelden

- ◆ HF explorer (AM, LSB,USB,DSB)
- ◆ NBFM ontvanger
- ◆ WFM ontvanger
- ◆ nbfm push-to-talk
- ◆ gmsk (packet radio)

Zelf blokken bouwen

- ◆ combineer bestaande blokken om een nieuw blok te maken (Python)
 - ◆ weinig programmeerkennis nodig
- ◆ Schrijf blokken in C++
 - ◆ documentatie en vele voorbeelden beschikbaar
 - ◆ enige programmeerkennis benodigd
- ◆ Iedereen kan en mag blokken toevoegen en verbeteren.
- ◆ Mensen helpen elkaar via de mailinglist

Voorbeelden van gebruik

- ◆ Cswiger heeft HF sets er mee vervangen en vele voorbeelden geschreven
 - ◆ Automatic testing with Magnitude and Phase plotting (Vector Network Analyzer)
- ◆ <http://webpages.charter.net/cswiger/>
- ◆ Passive Radar
- ◆ Radio astronomie
- ◆ supersnelle scanner (>1000 kanalen/sec)

eenvoudige am/ssb ontvanger

- ◆ `apps/hf_radio/radio_simple.py`

```

#!/usr/bin/env python
from gnuradio import gr, gru, eng_notation, optfir
from gnuradio import audio
from gnuradio import usrp
from ssbdemod import *

class my_graph(gr.flow_graph):
    def __init__(self):
        gr.flow_graph.__init__(self)

        freq=1.008e6

        src = usrp.source_c(decim_rate=128)
        subdev = usrp.selected_subdev(src, (0, 0))
        src.tune(0, subdev, freq)
        input_rate = src.adc_freq() / src.decim_rate()
        audio_decim = 16
        audio_rate = input_rate / audio_decim
        demod = ssb_demod( self, input_rate, audio_rate )
        agc = gr.agc_ff( 1e-4, 0.1, 0.1)
        out = audio.sink( audio_rate )
        self.connect( src, demod, agc, out )

        demod.set_bw( 10.0e3)
        #demod.upper_sb()
        #demod.lower_sb()
        demod.set_am()

```

```

if __name__ == '__main__':
    try:
        my_graph().run()
    except KeyboardInterrupt:
        pass

```

demos

- ◆ apps/hf_explorer/hfx.py
- ◆ apps/hf_radio/radio_simple.py
- ◆ usrp/fm_tx4.py
- ◆ usrp/filter_fft.py
- ◆ usrp/filter_fft_adv.py
- ◆ usrp/usrp_nbfm_ptt.py
- ◆ usrp/usrp_rx_cfile.py
- ◆ usrp/usrp_tx_cfile.py
- ◆ usrp/file_nbfm_rcv.py
- ◆ usrp/usrp_oscope.py
- ◆ usrp/usrp_wfm_rcv.py
- ◆ usrp/usrp_wfm_rcv2_nogui.py
- ◆ multi_antenna/multi_fft.py

USRP/Gnuradio als meetinstrument

- ◆ spectrum analyzer (0 – 2700 MHz)
- ◆ oscilloscope
- ◆ Vector Network analyzer
 - ◆ meet frequentie en fase respons van willekeurig welk component of apparaat
 - ◆ passieve componenten
 - ◆ filters
 - ◆ kabels
 - ◆ antennes
- ◆ acoustisch modelleren (galm, echo, ruis)
- ◆ Plaats, richting- en reflectie-bepaling

Vector Network Analyzer

- ◆ software gestuurde wobblers
- ◆ stuur sinus uit op tx kanaal
- ◆ ontvang sinus op rx kanaal
- ◆ varieer frequentie
 - ◆ vergelijk fase en amplitude ingang/uitgang
 - ◆ maak diagram
- ◆ varieer amplitude
 - ◆ meet lineariteit

Passive Radar

- ◆ ontvang willekeurige zender op kanaal 1
gerichte antenne
- ◆ ontvang dezelfde zender op kanaal 2 met
omnidirectionele antenne
- ◆ cross-correlate de twee signalen
 - ◆ wiskundige bewerking met behulp van FFT,
basisblokken aanwezig in gnuradio
- ◆ elke piek is een reflectie van een object op
een bepaalde afstand
- ◆ fase van de piek geeft de richting aan waar
het object zich bevindt

GnuRadio URLs , January 2006

Martin Dudok van Heel (PA1SDR)

PA1SDR@olifantasia.com
gnuradiomail@olifantasia.com
033-4799840

tempest for eliza FM

<http://www.olifantasia.com/gnuradio>
http://www.olifantasia.com/projects/gnuradio/mdvh/videocard_rf_output

GNU Radio home

<http://www.gnu.org/software/gnuradio/index.html>

GNU Radio mailing list, discussion

<http://lists.gnu.org/archive/html/discuss-gnuradio/>, also 2006-01/threads.html etc.

GNU Radio Wiki

<http://comsec.com/wiki>

USRP, GNU Radio hardware

<http://www.ettus.com/index.html>

<http://comsec.com/wiki?UniversalSoftwareRadioPeripheral>

<http://lists.gnu.org/archive/html/discuss-gnuradio/2004-12/msg00088.html>

GNU Radio GUI screenshots

<http://www.gnu.org/software/gnuradio/screenshots.html>

GNU Radio installation instructions

http://www.kd7lmo.net/ground_gnuradio_install.html (Linux)

<http://staff.washington.edu/~jon/gr-osx/gr-osx.html> (Mac OS X)

GNU Radio tutorial articles

<http://www.linuxjournal.com/article.php?sid=7319>

by Eric Blossom, with code samples

<http://www.linuxjournal.com/article.php?sid=7505>

GNU Radio tutorials by other users

<http://www.nd.edu/~dshen/GNU/>

<http://webpages.charter.net/cswiger/modules.html>

GNU Radio users' web sites,
with code and screenshots of
experiments and applications

<http://webpages.charter.net/cswiger/>

http://www.kd7lmo.net/ground_gnuradio_software.html

http://web.media.mit.edu/~jcooley/gr_experiments/index.htm

<http://viral.media.mit.edu/wiki/tiki-index.php?page=BPSK>

captured signals

<http://www.olifantasia.com/pub/projects/gnuradio/>

USRP hardware packaging
and installation

http://www.kd7lmo.net/ground_gnuradio_ota.html

http://web.media.mit.edu/~jcooley/gr_experiments/experiments/setup.htm

<http://webpages.charter.net/cswiger/amps.html> (scroll to bottom of this page for picture)

http://www.lamarowen.net/pgallery/album?album_id=2060

GNU Radio at government
and university laboratories

<http://lists.gnu.org/archive/html/discuss-gnuradio/2005-05/msg00190.html>

<http://www.nsf.gov/awardsearch/showAward.do?AwardNumber=0435485>

<http://lists.gnu.org/archive/html/discuss-gnuradio/2005-04/msg00092.html>

Readings and other resources

<http://radio.dcarr.org>

<http://comsec.com/wiki?SuggestedReading>

Grootste deel van deze lijst is gebaseerd op de lijst van Jonathan Jacky op

<http://staff.washington.edu/~jon/gnuradio.html>

GnuRadio en Software Defined Radio

- ◆ Martin Dudok van Heel (PA1SDR)
- ◆ PA1SDR@olifantasia.com
- ◆ gnuradiomail@olifantasia.com
- ◆ 033-4799840
- ◆ freelance elektronicus (SDR, video, 3D)
- ◆ freelance programmeur (SDR, video, 3D)
- ◆ hacker
- ◆ wereldverbeteraar
- ◆ importeur/verkoper USRP voor Europa
(zeer binnenkort)