

# ***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
- ◆ elektronicus / hardware ontwerper
- ◆ freelance Software Defined Radio en FPGA/verilog/C++ programmeur
- ◆ hacker
- ◆ wereldverbeteraar
- ◆ 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
- ◆ USRP
  - ◆ opensource hardware
- ◆ 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**
  - ◆ één apparaat (kan communiceren met) vele verschillende systemen.
  - ◆ te herprogrammeren als mobiele telefoon, portofoon, draadloze netwerkkaart, radardetector, fm-radio, digitale TV-ontvanger/zender of een garagedeuropener.
- ◆ **Kostenreductie**
- ◆ **Het ontwerp = het apparaat**
  - ◆ kennis delen = het apparaat delen
- ◆ **Toekomstvast**
  - ◆ softwarematig opwaarderen naar toekomstige telecomstandaarden.

# ***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*

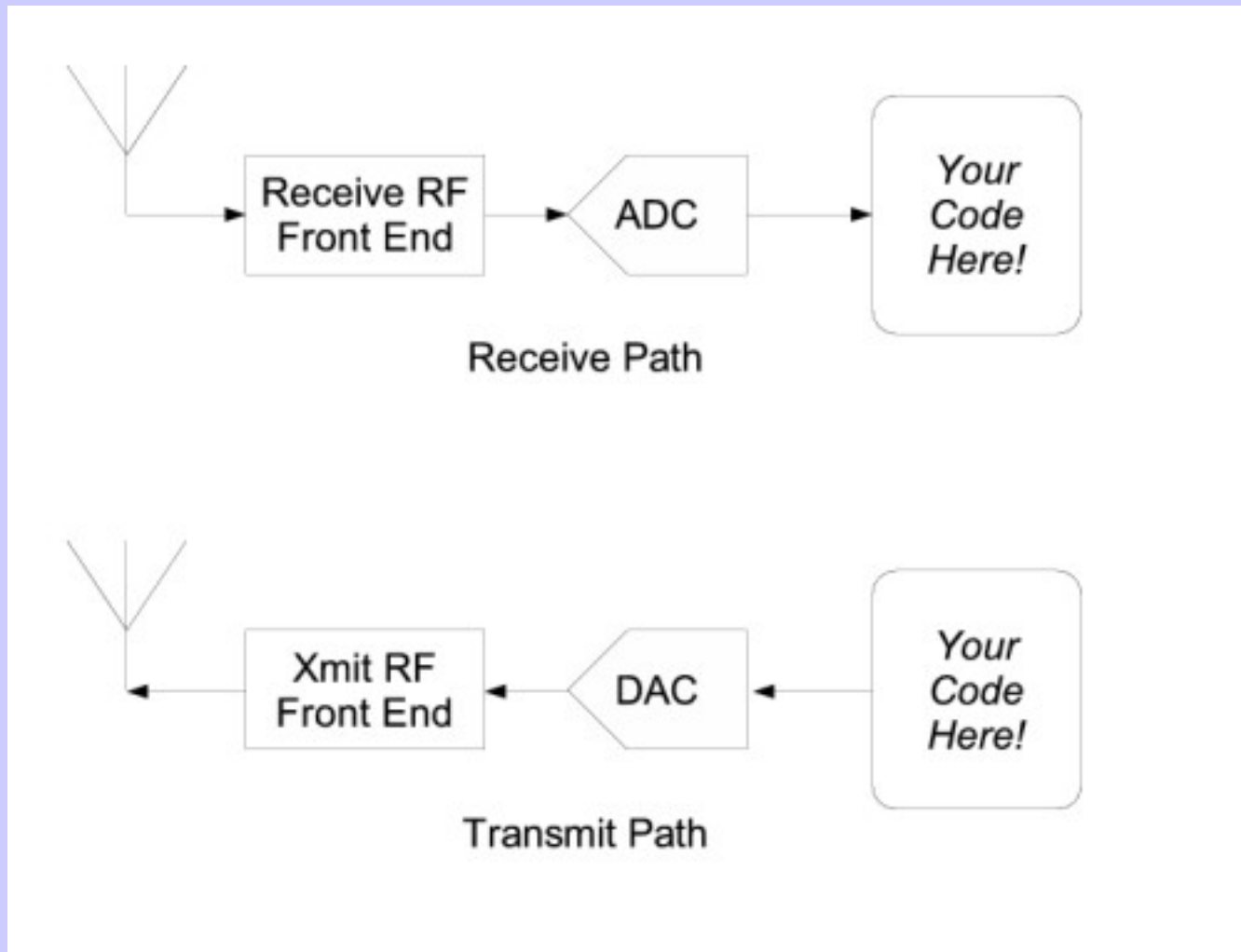
- ◆ heel snelle digitale multimeter
- ◆ meet vaak per seconde de signaalsterkte
  - ◆ samplerate = aantal keer per seconde dat signaal gemeten wordt
- ◆ verwerk deze getallen digitaal (enen en nullen, 5 Volt, 0 Volt)
- ◆ niquist theorema
  - ◆ samplerate  $\geq 2 \times f_{\max}$
- ◆ voorbeeld: geluidskaart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

# *Waar is Software Radio goed voor?*

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

# *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 algemeen en flexibel mogelijk
- ◆ Doe alles wat complex is in software.



# *Kwadratuur*

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

# *Wat is GnuRadio ?*

- ◆ Een gratis opensource gereedschapskist:
  - ◆ Software radio's bouwen en toepassen.
  - ◆ Platform om te experimenteren met digitale en analoge communicatie
  - ◆ Leren/studie van DSP en communicatie systemen
  - ◆ Platform voor signaalverwerking
  - ◆ hulpmiddel: snel bouwen van prototypes.
  - ◆ Maak compleet nieuwe soorten van radios:
    - ◆ modulatie, protocollen, ontwikkel omgevingen...
- ◆ Licentie is GPL
- ◆ Community effort

# *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, printlay-out, specificaties van de hardware vrij beschikbaar onder een vrije licentie
- ◆ bijbehorende software opensource onder een vrije licentie
  - ◆ ook 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

# *Architectuur / implementatie*

- ◆ **Data flow** abstractie
  - ◆ Signaal verwerkings blokken en verbindingen
- ◆ Maar ook **Message Blocks**
- ◆ Hybride **C++ / Python** systeem
- ◆ Draait meestal op general purpose processor
- ◆ GNU/Linux, NetBSD, OS/X, MinGW/Cygwin

# *Signaal Verwerkings Blokken*

- ◆ **Input streams** en **output streams**
- ◆ I/O signatuur
  - ◆ **Type** van elke stream is **gespecificeerd**
  - ◆ Blokken specificeren aantal ondersteunde streams
- ◆ Relatieve i/o rates
  - ◆ **Vast** 1:1, Vast interpoleer 1:N, Vast decim N:1
  - ◆ **Variabel**

# *Input/Output*

- ◆ standaard Input/Output bouwblokken
  - ◆ geluidskaart `gr.audio_sink()`
  - ◆ USRP `gr.usrp_source()`
  - ◆ `gr.usrp_sink()`
  - ◆ spectrum display `gr.fft_sink()`
  - ◆ oscilloscope `gr.oscope_sink()`
  - ◆ bestand `gr.file_sink()`
  - ◆ TV display `gr.video_sink()`



# *"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
- ◆ verbindt blokjes
- ◆ geen C++ kennis nodig
- ◆ niet compileren
- ◆ Lego bouwdoos
- ◆ GRC grafische designer
  - ◆ geen python of C++ kennis nodig

# ***GRC***

- ◆ grafisch modellieren radio

# *Wie gebruikt GnuRadio ?*

- ◆ Academici / onderzoekers
- ◆ Industrie / DARPA onderzoekers / Intel
- ◆ Overheids onderzoeks groepen
- ◆ Hackers
- ◆ Radio Amateurs
- ◆ Radio Astronomen
- ◆ Autoradio fabrikanten
- ◆ Zoologen / Biologen

# ***Wat hebben we al bereikt?***

- ◆ Grote brede gebruikersgemeenschap
- ◆ Zend en ontvang meest gangbare analoog en digitaal gemoduleerde signalen
- ◆ gebruik voor radio astronomie
- ◆ gebruik als meetinstrument

# *Modulatievormen / Signaal types*

- ◆ Nu:
  - ◆ Digitaal:
    - ◆ OFDM
    - ◆ FSK, GMSK, PSK, QPSK, QAM
    - ◆ ATSC VSB-8 (amerikaanse digitale HDTV)
  - ◆ Analooq
    - ◆ AM, FM, SSB, analoge TV ...
- ◆ Binnenkort:
  - ◆ Spread Spectrum
    - ◆ Fast Freq Hopper
    - ◆ Direct Sequence (Spread Spectrum)

# *Toepassingen*

- ◆ Transceivers
- ◆ Onderzoek draadloze netwerken en verbindingen
- ◆ Ad-hoc netwerken (Rampenbestrijding)
- ◆ STAP / Adaptive beam forming
- ◆ Cognitive Radio
- ◆ RADAR
- ◆ Geolocation
- ◆ SIGINT
- ◆ Radio Astronomie
- ◆ Amateur radio (inclusief bovenstaande)
- ◆ Muskusrat volgsysteem
- ◆ Meetinstrument



# *Cognitive Radio*

- ◆ Kies je favoriete definitie
- ◆ Veel initiatieven gebruiken GNU Radio
  - ◆ DARPA ACERT (BBN, UCLA)
  - ◆ Virginia Tech
  - ◆ CMU
  - ◆ Rutgers WINLAB
- ◆ Vaak in combination met Click Modular Router

# ***SIGINT***

- ◆ supersnelle scanner (>1000 kanalen/sec)
- ◆ spectrum recorder
- ◆ automatic signal detection en recording/demodulation

# *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

# *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

# *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, USRP, SSRP, measurement ADC cards MC4020
    - ◆ met beperkingen: (hacked) tv-kaart: BT878, cx2388x PCHDTV3000)

# *Hardware voor GnuRadio*

- ◆ Gewone PC
- ◆ RF front end
  - ◆ minimum, alleen een draadje
- ◆ Analooog/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](http://www.kd7lmo.net/ground_gnuradio_ota.html)
  - ◆ netwerk bronnen (TCP/UDP/Ip sockets)
  - ◆ alles waar je een driver voor hebt of schrijft.

# *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



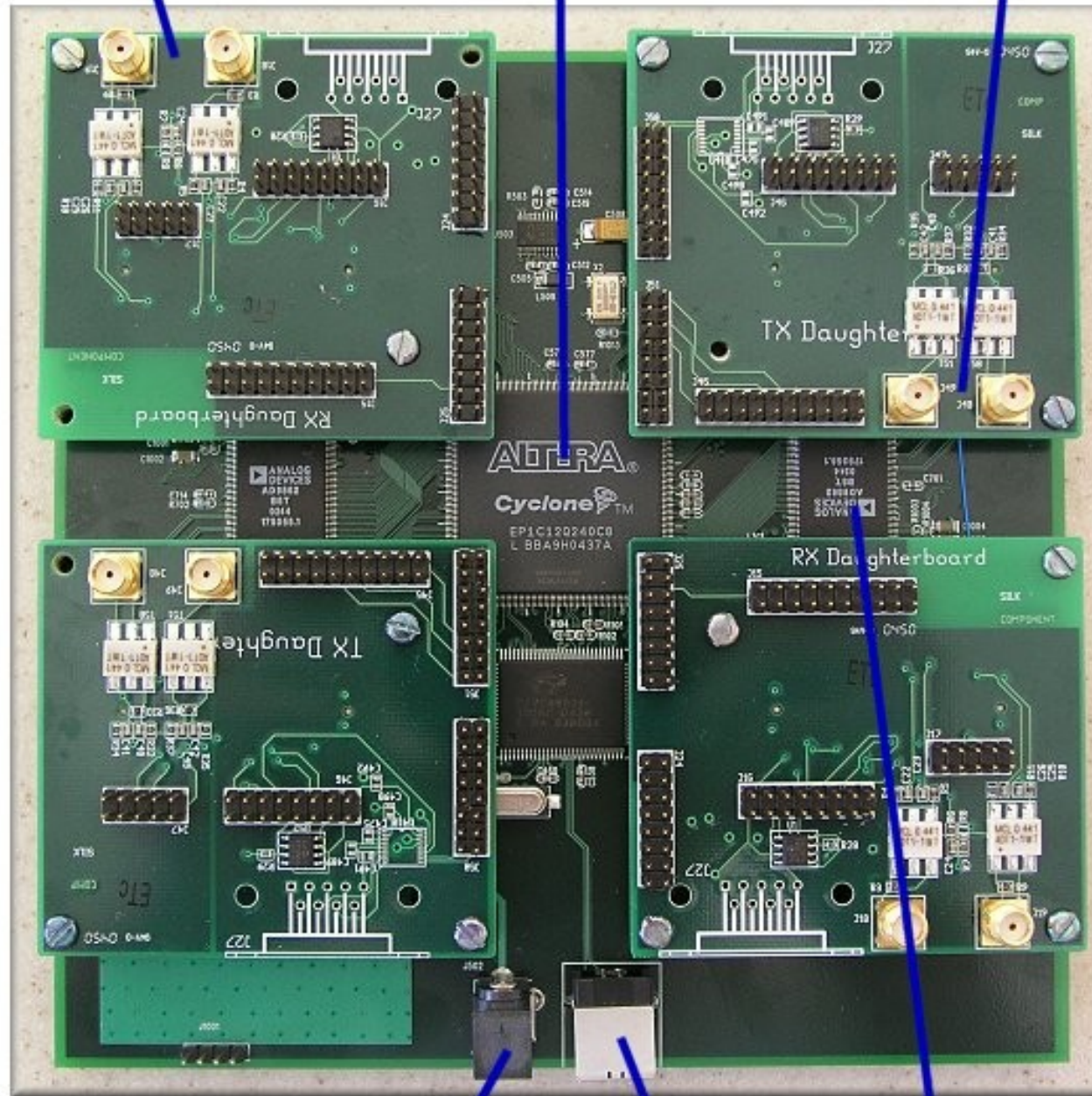
# *USRP eigenschappen*

- ◆ verwisselbare RF frontend secties
- ◆ zeer grote bandbreedte
- ◆ extreem flexibel
  - ◆ programmeerbaar
  - ◆ uitbreidbaar
- ◆ beam forming/Smart Antenna/MIMO/BLAST
- ◆ multipath
- ◆ gebruikersgemeenschap die ook weer functies toevoegt.
- ◆ open hardware (GPL)

Receive Channel  
RF Interface

Altera FPGA

Transmit Channel  
RF Interface

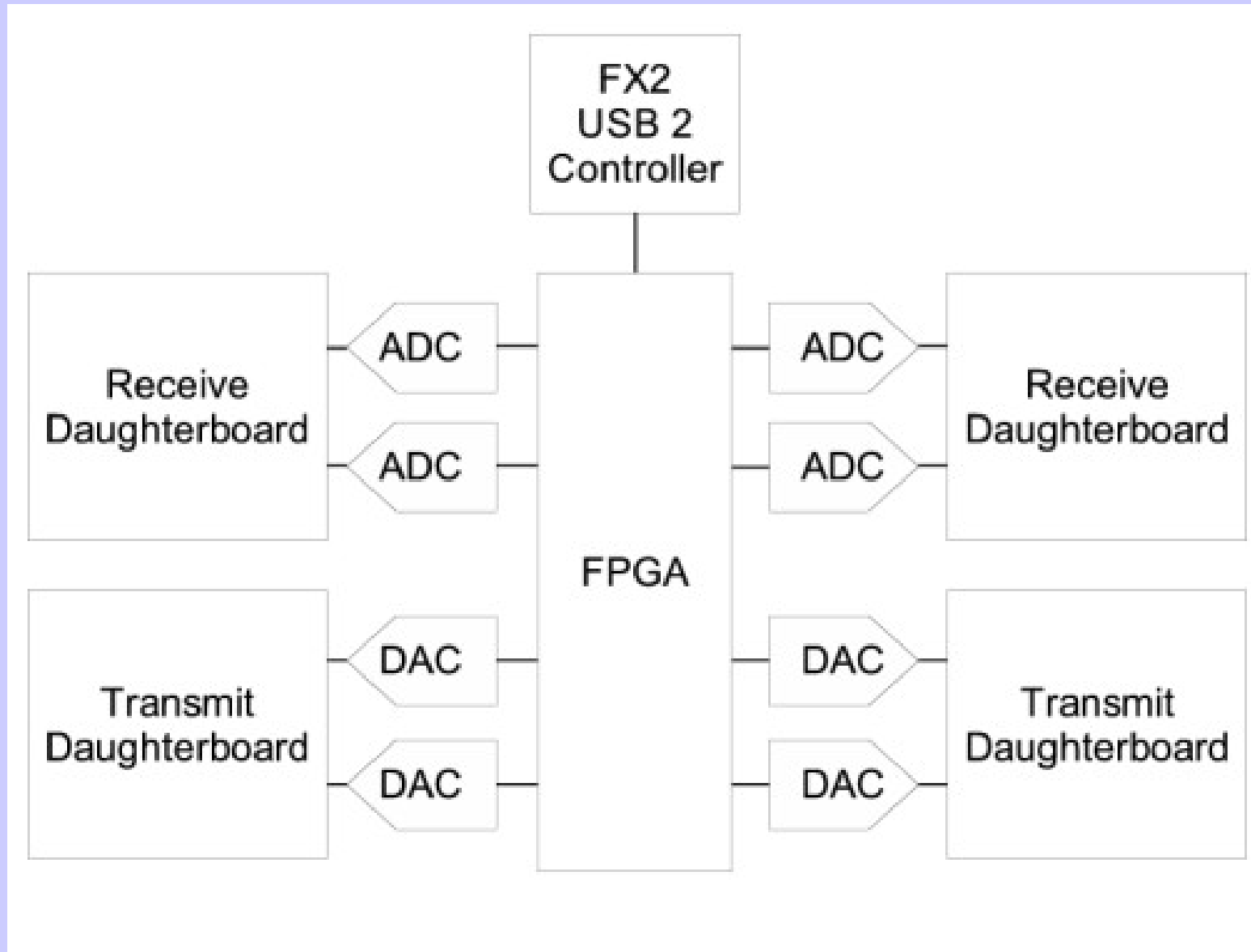


DC Power

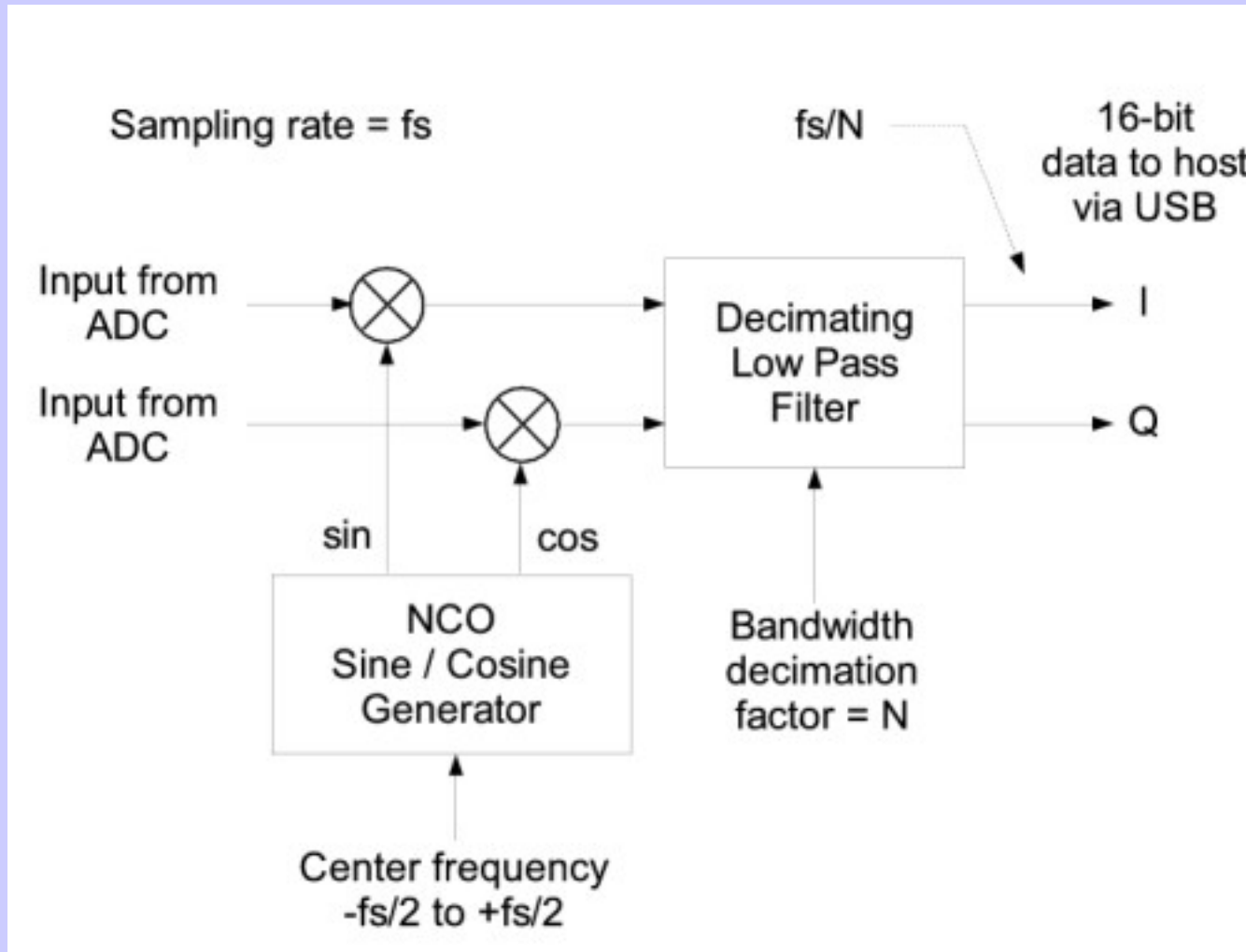
USB 2.0  
Port

Analog Devices  
Mixed Signal  
Processor

# *USRP blokdiagram*



# FPGA functions



# *USRP werking*

- ◆ dochterbord = RF frontend
- ◆ FPGA
  - ◆ frequentie omzetten
  - ◆ decimeren, interpoleren
  - ◆ filteren
- ◆ Ontworpen in gschem+PADS, rest in gschem+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
  - ◆ kleine 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

# *USRP2 specificaties*

- ◆ dual 100 MS/s 14-bit analog to digital converters
- ◆ dual 400 MS/s 16-bit digital to analog converters
- ◆ Digitale up and downconverters
- ◆ Gigabit Ethernet Interface (1000 Mbit/sec)
- ◆ 2 Gbps high-speed serial interface for expansion
- ◆ Kan signalen verwerken tot 100 MHz bandbreedte
- ◆ Modulaire architectuur (zelfde dochterbordjes als USRP1)
- ◆ Auxiliary analoge en digitale I/O (voor AGC etc)
- ◆ Coherent multi-channel systems (MIMO) mogelijk tot 8 antennes
- ◆ 1 Megabyte on-board high-speed SRAM
- ◆ 256 MB flash memory op (losse) SD-card voor standalone gebruik

# *Dochterbord (RF frontend)*

- ◆ BasicTX and BasicRX 0.1 – 200 MHz
  - ◆ Voor gebruik met externe (zelfbouw)RF secties
  - ◆ Flexibele IF
  - ◆ directe ontvangst mogelijk met beperkte gevoeligheid
- ◆ LFRX en LFTX 0 – 30 MHz
- ◆ TVRX 0 – 860 Mhz (RX only)
- ◆ DBSRX 800 – 2500 Mhz (RX only)



# *Transceiver Dochterborden*

- ◆ Geschikt om te zenden en ontvangen
- ◆ RFX900 -- 800-1000 MHz Transceiver, 250+ mW
  - ◆ radio amateur + ISM bands, GSM, mod naar RFX1800
- ◆ RFX1200 -- 1150-1400 Mhz, 200+ mW output
  - ◆ radio amateur, Satelliet, Navigatie banden
- ◆ RFX1800 -- 1500-2100 MHz, 250+ mW output
  - ◆ PCS, GSM, DECT
- ◆ RFX2400 -- 50+ mW output, 2300-2800 Mhz
  - ◆ WIFI,radio amateur, ISM bands
- ◆ XCVR2450 -- 100+ mW output, 2400-2500 Mhz en 4900-5900 Mhz
  - ◆ WIFI,radio amateur, ISM bands, UNII
- ◆ binnenkort **WBX0510** -- 100+ mW output, 50-1000 Mhz
  - ◆ omvat vele frequentiebanden

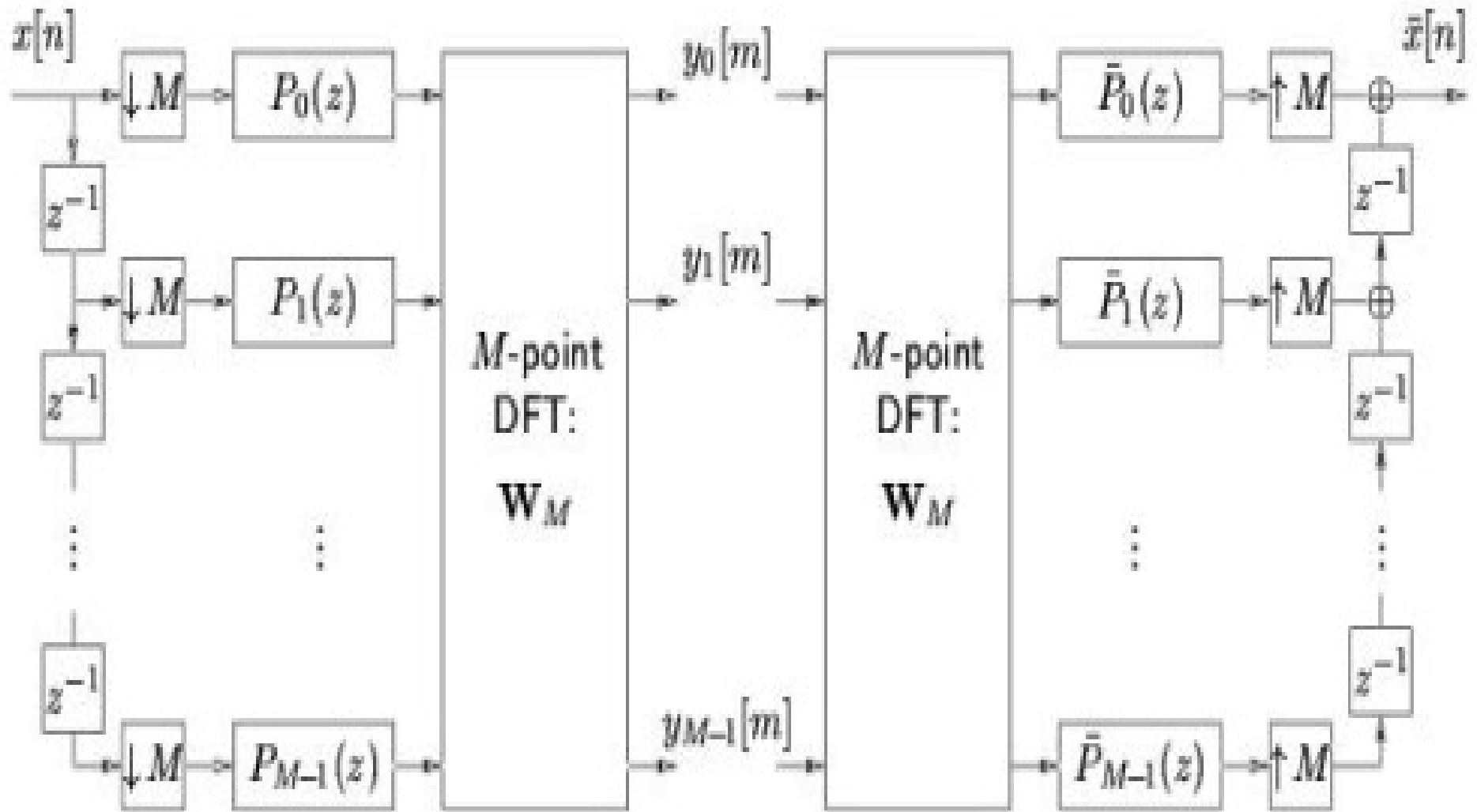
# *Beschikbare software bouwblokken in GnuRadio*

- ◆ Teveel om op te noemen
  - ◆ Alle standaard mathematische bewerkingen
  - ◆ Alle soorten FIR en IIR filters
    - ◆ teken je eigen frequentiespectrum en laad het in het filter
    - ◆ standaard lowpass, bandpass, highpass, notch
  - ◆ modulatie/demodulatie blokken
  - ◆ signaalgenerators
  - ◆ pll / costas loop / AGC
  - ◆ FFT
    - ◆ correlatie            gr.fft + gr.mult+ gr.ifft
  - ◆ filterbank            meervoudig kanaalfilter

# *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
- ◆ aantal kanalen onbeperkt

# *filterbank*



## ***Wat kost het***

- ◆ setup met eigen RF frontend en smalband gebruik via geluidskaart ingang GRATIS
- ◆ Gnuradio software GRATIS
- ◆ Ontwerp/printlayout USRP GRATIS
  
- ◆ hacken videokaart ??? euro
- ◆ USRP 900 euro
- ◆ dochterbordjes usrp 80 - 300 euro

# *voorbeeld scripts*

- ◆ HF explorer (AM, LSB,USB,DSB)
- ◆ NBFM ontvanger
- ◆ WFM ontvanger/zender
- ◆ nbfm push-to-talk
- ◆ gmsk (packet radio)
- ◆ digital voice
- ◆ fm\_tx4
- ◆ spectrum sense
- ◆ oscilloscope
- ◆ spectrum analyzer

# *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

# *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



# *Waar zijn we mee bezig ?*

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

Martin Dudok van Heel (PA1SDR) PA1SDR@olifantasia.com

033-4799840

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

[http://www.olifantasia.com/gnuradio/usrp/usrp\\_prijzen\\_en\\_eigenschappen.html](http://www.olifantasia.com/gnuradio/usrp/usrp_prijzen_en_eigenschappen.html)

tempest for eliza FM

[http://www.olifantasia.com/projects/gnuradio/mdvh/videocard\\_rf\\_output](http://www.olifantasia.com/projects/gnuradio/mdvh/videocard_rf_output)

GNU Radio home

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

GNU Radio mailing list

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

GNU Radio Wiki

<http://gnuradio.org/trac/wiki>

USRP, GNU Radio hardware

[http://www.olifantasia.com/gnuradio/usrp/usrp\\_prijzen\\_en\\_eigenschappen.html](http://www.olifantasia.com/gnuradio/usrp/usrp_prijzen_en_eigenschappen.html)

<http://gnuradio.org/trac/wiki/USRP>

[http://gnuradio.org/trac/wiki/List\\_of\\_USRP\\_daughterboards](http://gnuradio.org/trac/wiki/List_of_USRP_daughterboards)

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

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

GNU Radio installation

<http://gnuradio.org/trac/wiki/GettingStarted>

<http://gnuradio.org/trac/wiki/BuildGuide>

Vector Network analyzer met GnuRadio

<http://web.archive.org/web/webpages.charter.net/cswiger/vna.html>

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

<http://web.archive.org/web/webpages.charter.net/cswiger/>

[http://www.kd7lmo.net/ground\\_gnuradio.html](http://www.kd7lmo.net/ground_gnuradio.html)

[http://alumni.media.mit.edu/~jcooley/gr\\_experiments/index.html](http://alumni.media.mit.edu/~jcooley/gr_experiments/index.html)

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

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

Old GNU Radio tutorial articles  
with code samples

<http://www.linuxjournal.com/article/7319> (old)

<http://www.linuxjournal.com/article/7505> (old)

captured signals

[http://www.kd7lmo.net/ground\\_gnuradio\\_ota.html](http://www.kd7lmo.net/ground_gnuradio_ota.html)

GNU Radio at government  
and university laboratories

<http://www.projects.ncassr.org/sdr/>

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

Readings and other resources

<http://radio.dcarr.org>

<http://gnuradio.org/trac/wiki/SuggestedReading>

# *GnuRadio en Software Defined Radio*

- ◆ Martin Dudok van Heel (PA1SDR)
- ◆ PA1SDR@olifantasia.com
- ◆ 033-4799840
- ◆ freelance elektronicus (SDR, USRP, video, 3D, FPGA)
- ◆ freelance programmeur (SDR, GnuRadio, video, 3D)
- ◆ Software Defined Radio /USRP /GnuRadio consultant
- ◆ wereldverbeteraar
- ◆ importeur/verkoper USRP voor Europa