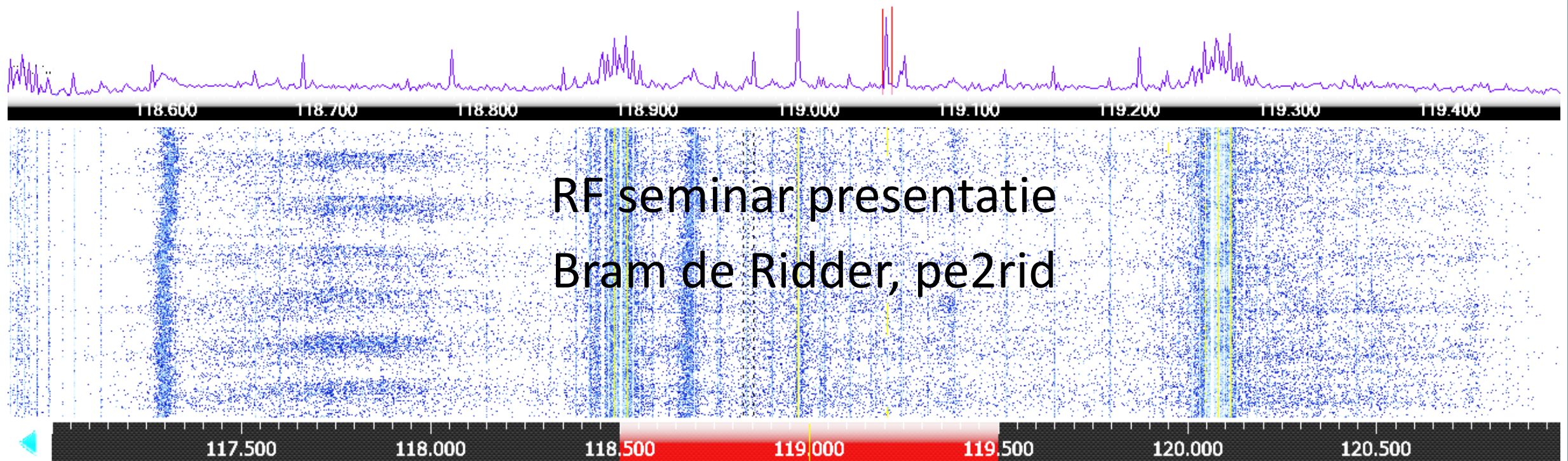


Digital Signal Processing in Software Defined Radio



Onderwerpen

- Waarom *Software Defined* Radio
- Digital Signal Processing (DSP)
 - Digitaliseren van analoge signalen
 - Ruis
 - Quadratuur signalen
 - Digitale filters en frequentie conversie

Waarom *Software Defined Radio*?

Het 'radio'-principe:

- conversie van informatie uit het elektromagnetisch spectrum (EMS)
- informatie moet 'gebruiksklaar' gemaakt worden

Klassieke radio:

- elektrische signalen uit een EMS-converter (= antenne) analoog bewerken:
 - selecteren van informatie uit de EMS-soep
 - manipuleren van dit signaal tot bruikbaar niveau

Waarom *Software Defined Radio*?

Benodigde elementen:

- isoleer gewenst deel van het EM-spectrum
- extract gewenste informatie
- niveau aanpassing

afstemmen: 100 Hz uit GHz-bereik
selectiviteit: ongewenste externe signalen – tig dB's

demodulatie: CW, SSB, AM,
(N)FM, AFSK, PSK, PM, QAM,...

voorwaarde:

geen extra informatie toevoegen

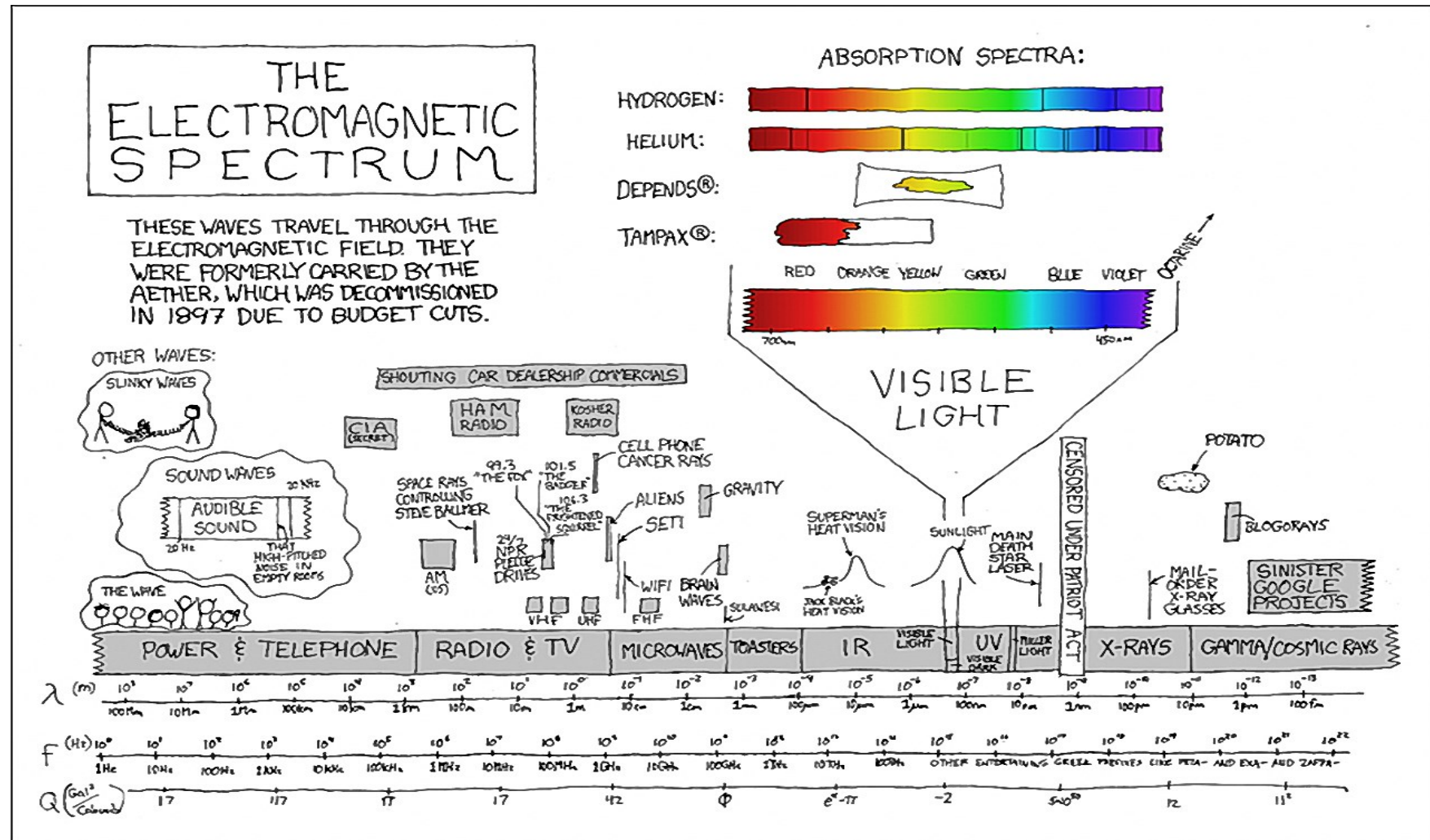
versterken

interne ruis en stoorsignalen



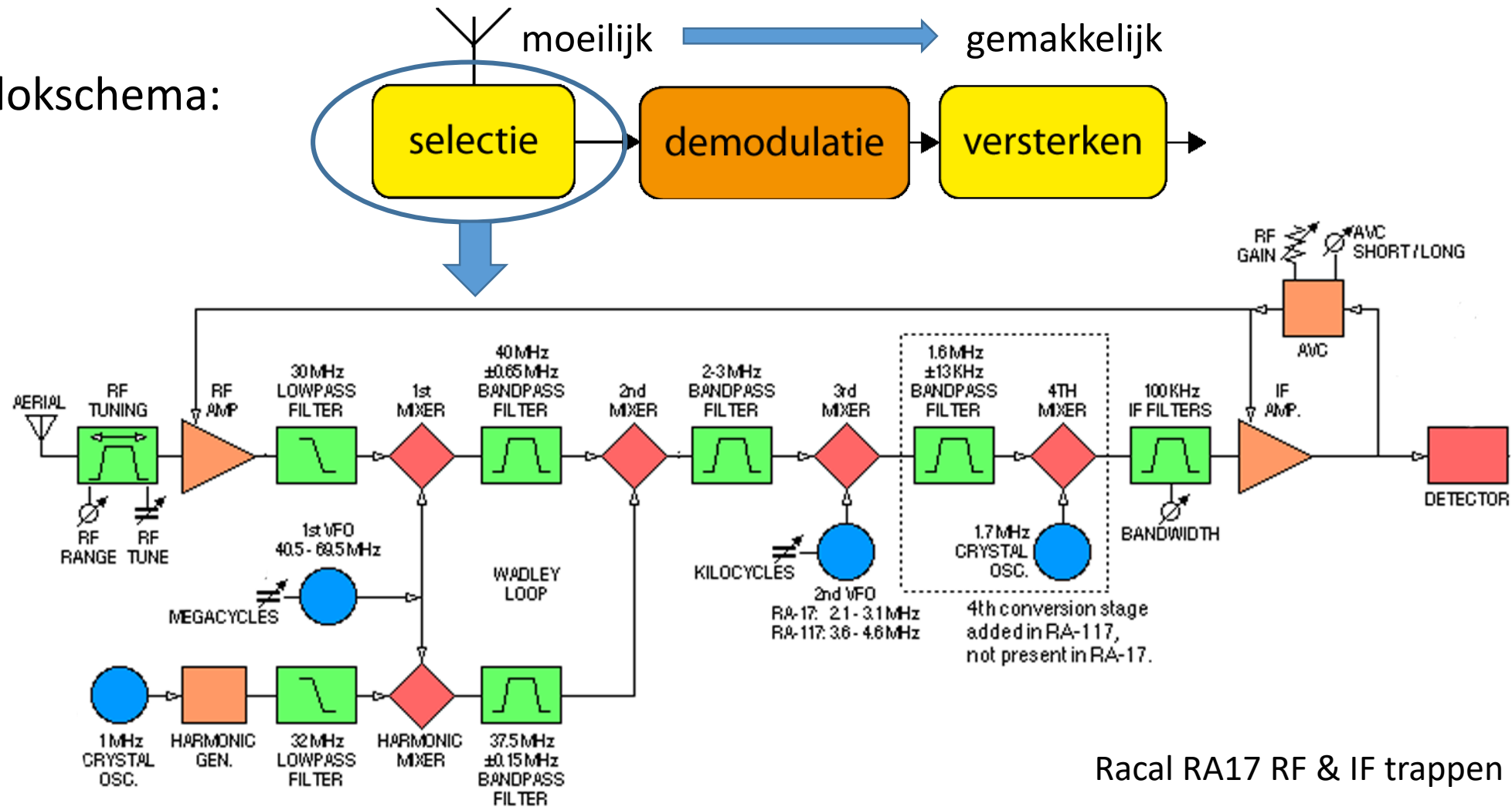
Waarom *Software Defined Radio*?

EMS-soep



Waarom *Software Defined Radio*?

Blokschema:



Racal RA17 RF & IF trappen

Waarom *Software Defined Radio*?

Uitdagingen bij klassieke constructie:

- (thermische) ruis
- scheiden van gewenste en ongewenste signalen
- complexe constructie (stabiliteit, selectiviteit, frequentieconversies)

De oplossing (?) – ‘*Go Digital....*’

- analoge signalen digitaliseren (*ADC*)
- na digitalisatie rekenen = *software*

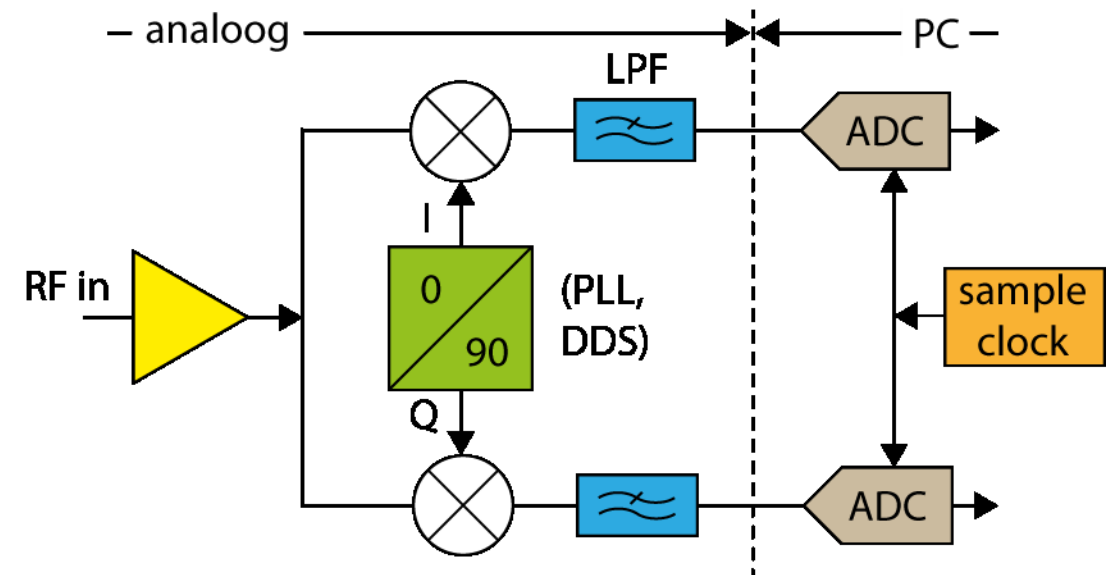
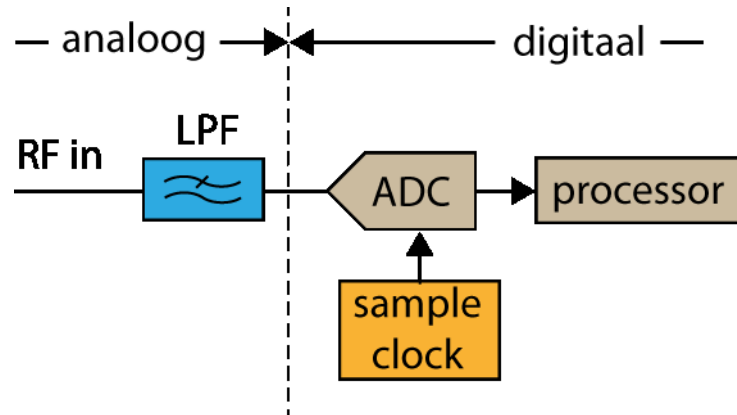
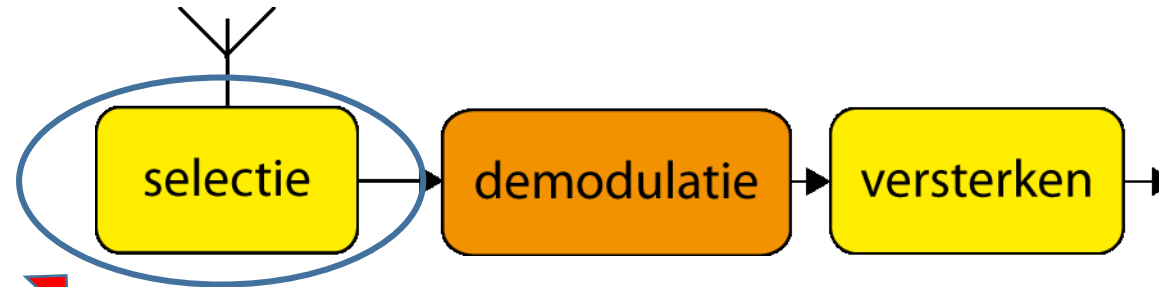
“~~klassieke~~” radioamateur:



Waarom *Software Defined Radio*?

Blokschema:

★ *software*



Basic SDR: IQ-mixen naar (audio) baseband: 48 / 96 / 192 kHz

Basisbegrippen DSP

Analoog naar digitaal converter (ADC)

Analoog:

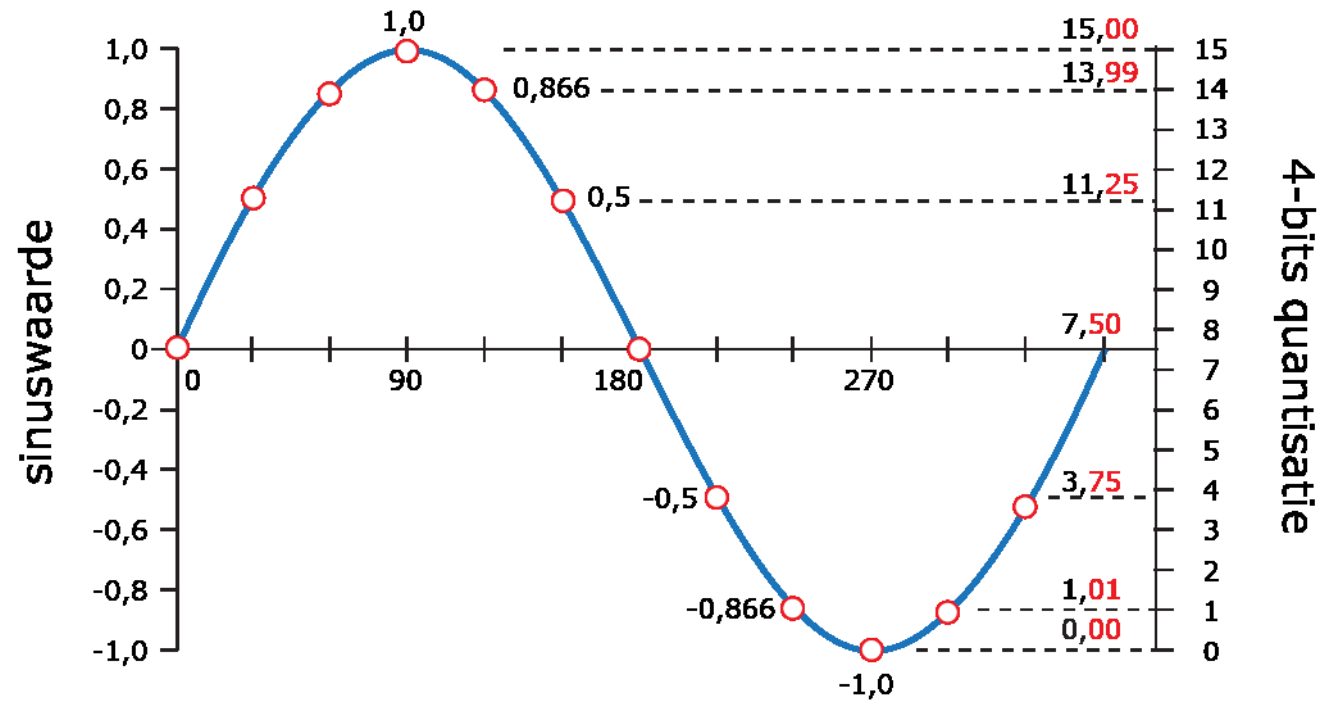
'oneindig' aantal waarden

Digitaal:

aantal waarden (getallen) hangt af
van het aantal bits n:

$$x = 2^N$$

4 Bits $\rightarrow 2^4 = 16$ waarden voor x



Basisbegrippen DSP

Analoog naar digitaal converter (ADC)

Analoog:

‘oneindig’ aantal waarden

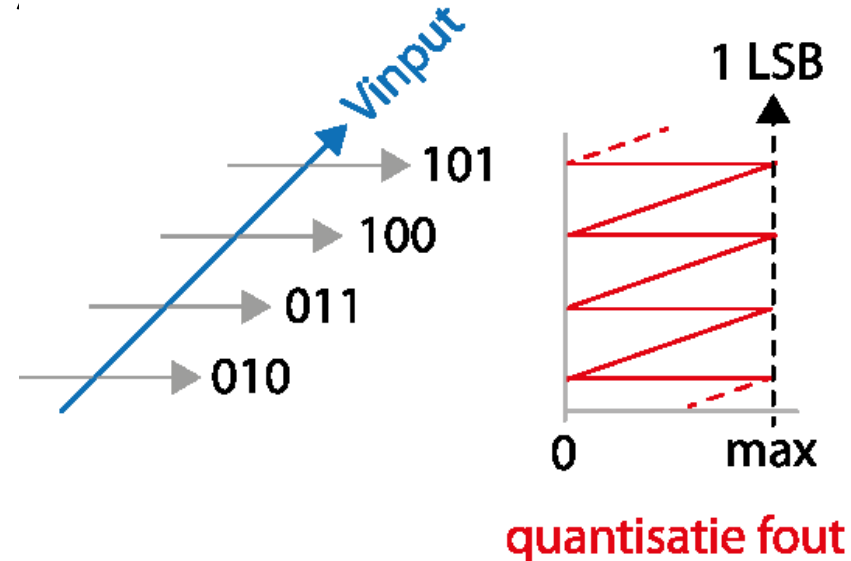
Digitaal:

aantal waarden (getallen) hangt af
van het aantal bits n:

$$x = 2^N$$

4 Bits $\rightarrow 2^4 = 16$ waarden voor x

$$\text{SNR[dB]} = 20 \log \{ 2^N \times \sqrt{3/2} \} = 6 \times N + 1,76$$

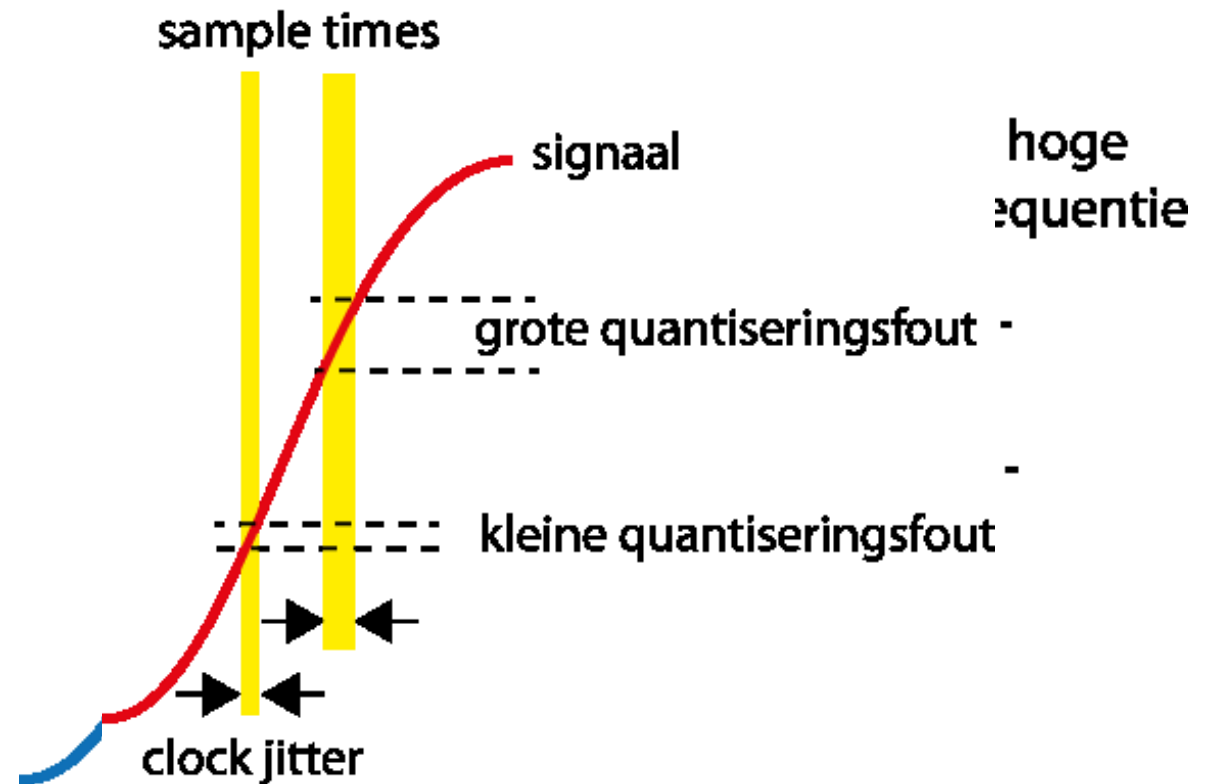


Basisbegrippen DSP

Analoog naar digitaal converter (ADC)

Jitter in sampling clock signaal

SNR tgv jitter vermindert bij hogere frequenties



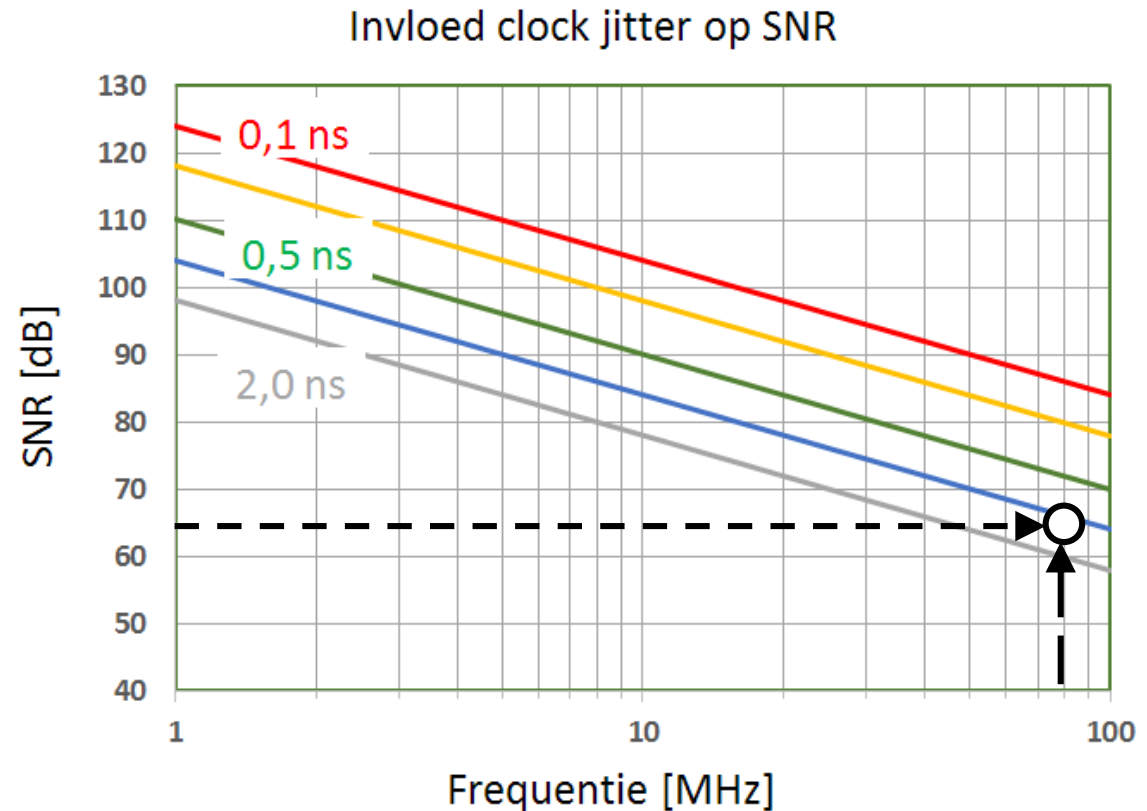
Basisbegrippen DSP

Analoog naar digitaal converter (ADC)

Jitter in sampling clock signaal

14 bit ADC @ 80 MHz (12,5 ns):
 $\text{SNR}_{\text{ADC}} \approx 14 \times 6 = 64 \text{ dB}$

Jitter > 1,5 ns

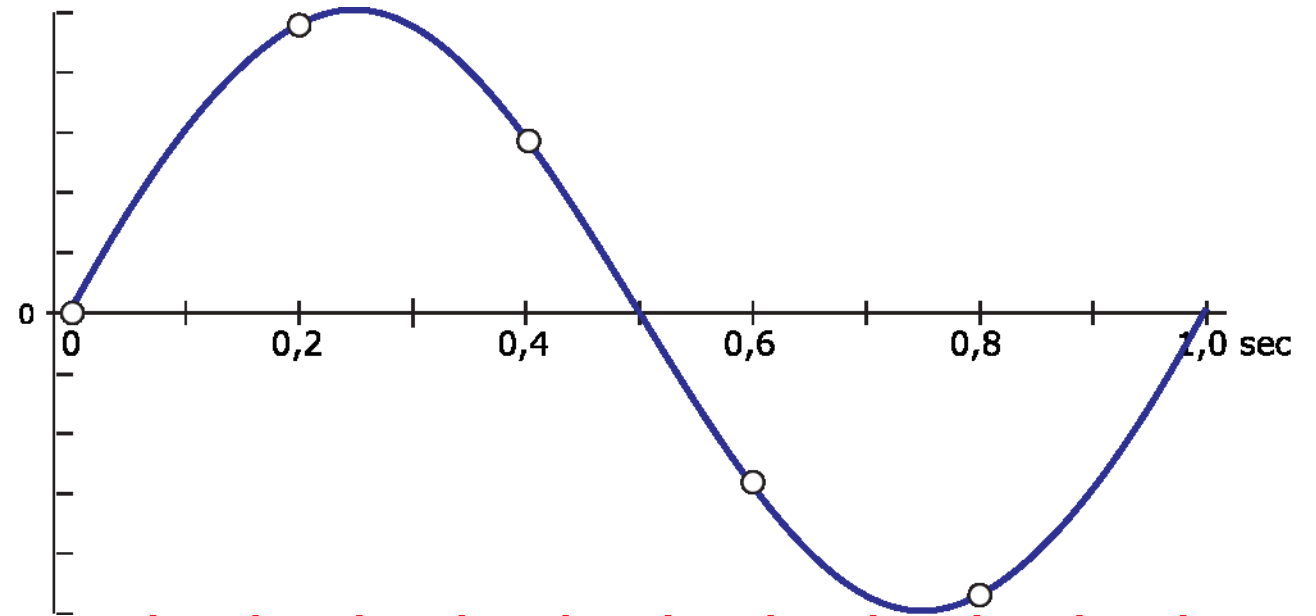


Basisbegrippen DSP

Analoog naar digitaal converter (ADC)

$f_s = 5$ samples per seconde (sps)

- frequentie: 1Hz
- alias freq: 6 Hz
- alias freq 11 Hz

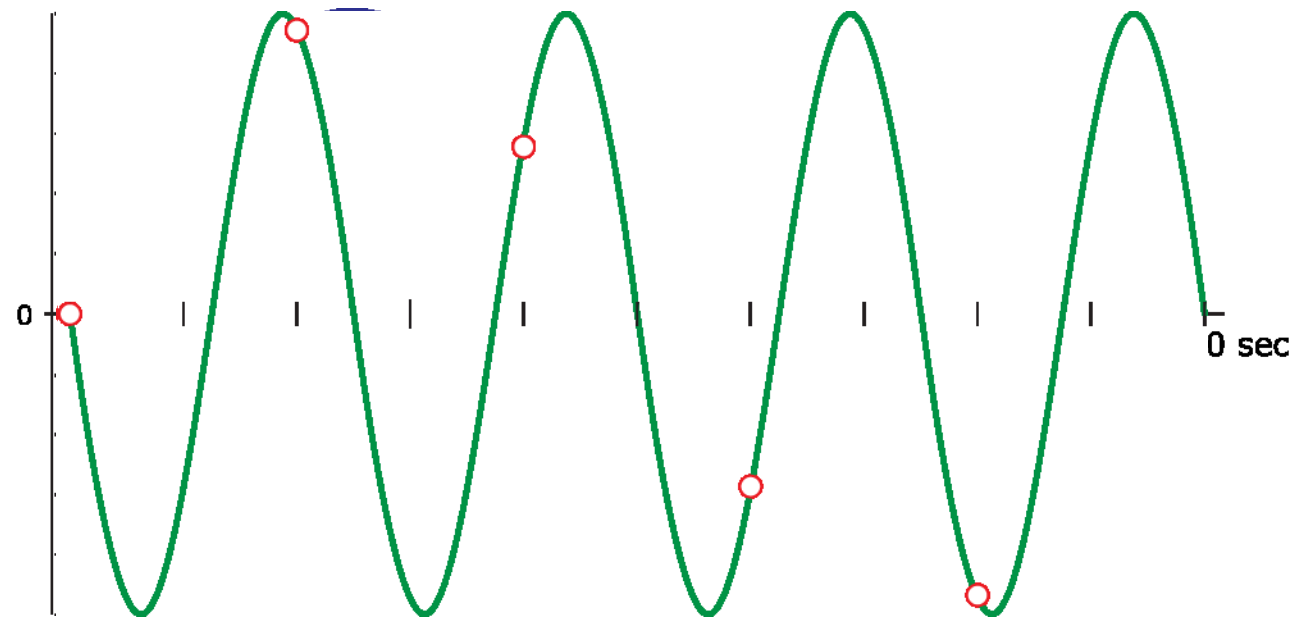


Basisbegrippen DSP

Analoog naar digitaal converter (ADC)

$f_s = 5$ samples per seconde (sps)

- frequentie: 1Hz
- alias freq: 6 Hz
- alias freq 11 Hz
- alias freq 4 Hz
(180° fasedraaiing)



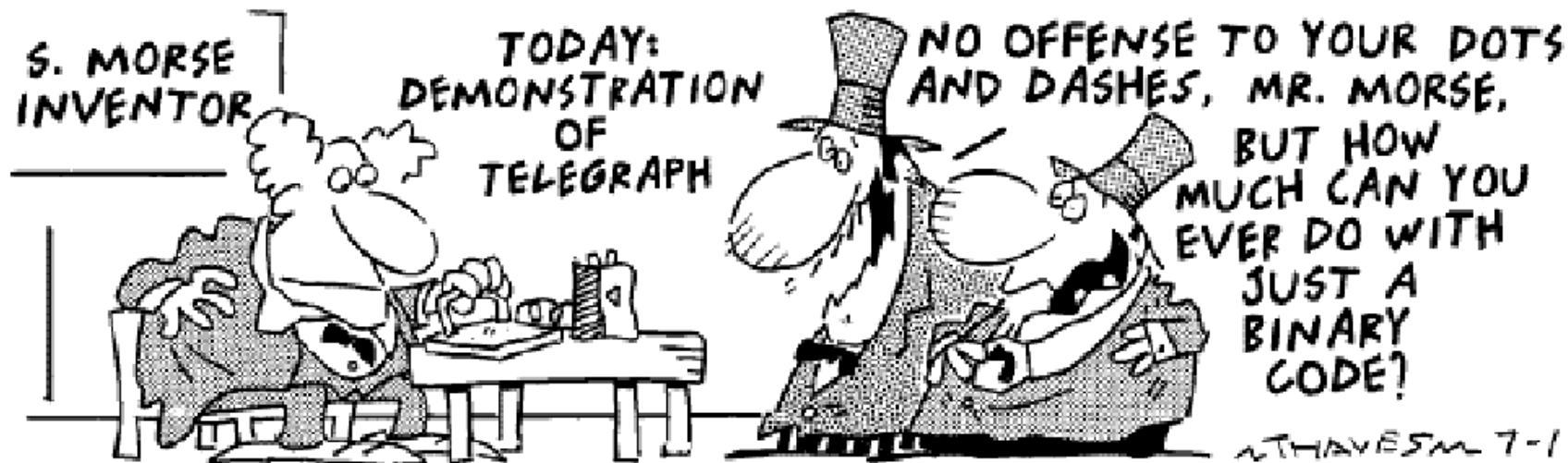
frequenties boven de halve sample rate geven identieke gedigitaliseerde waarden

Basisbegrippen DSP

Analoog naar digitaal converter (ADC)

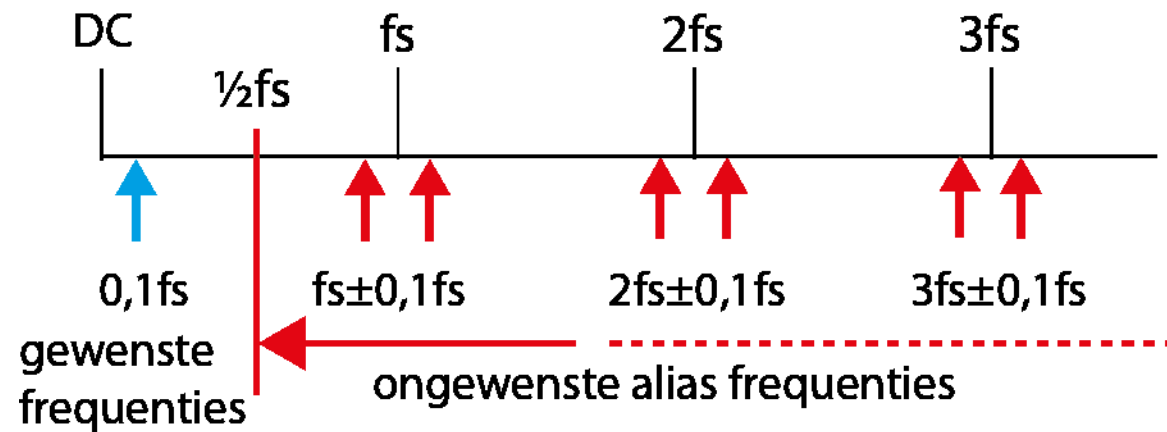
Nyquist limiet:

alleen voor frequenties beneden $f_s/2$ eenduidige sampling mogelijk

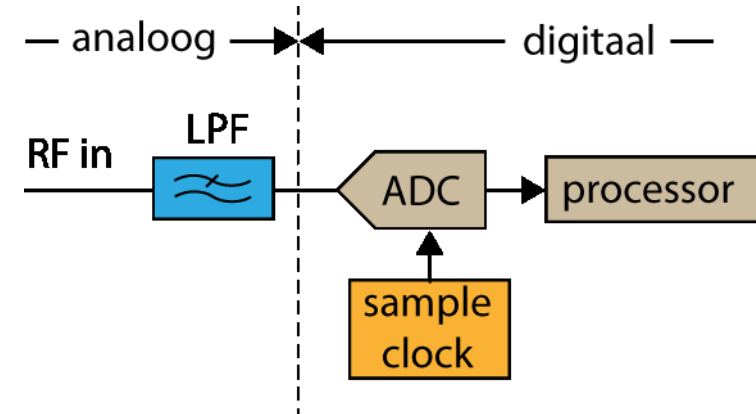


Basisbegrippen DSP

Analoog naar digitaal converter (ADC)



Nyquist limiet: analoog filter tot maximaal $f_s/2$ *voor* het digitaliseren

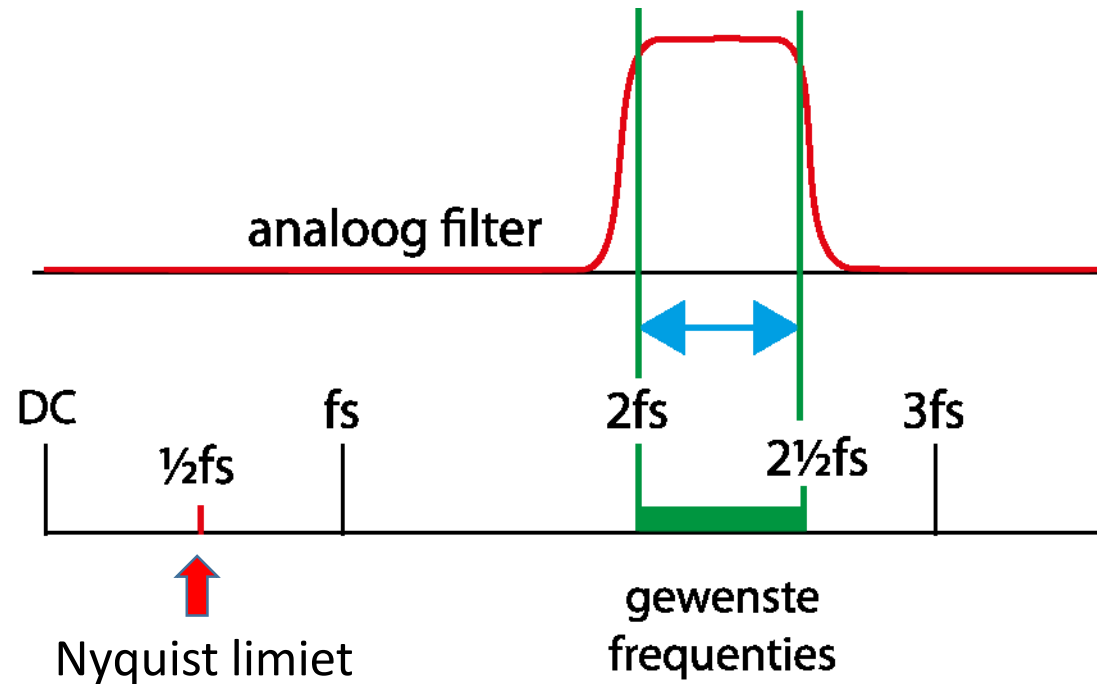


Basisbegrippen DSP

Analoog naar digitaal converter (ADC)

alias frequenties vermijden, of doelbewust gebruiken

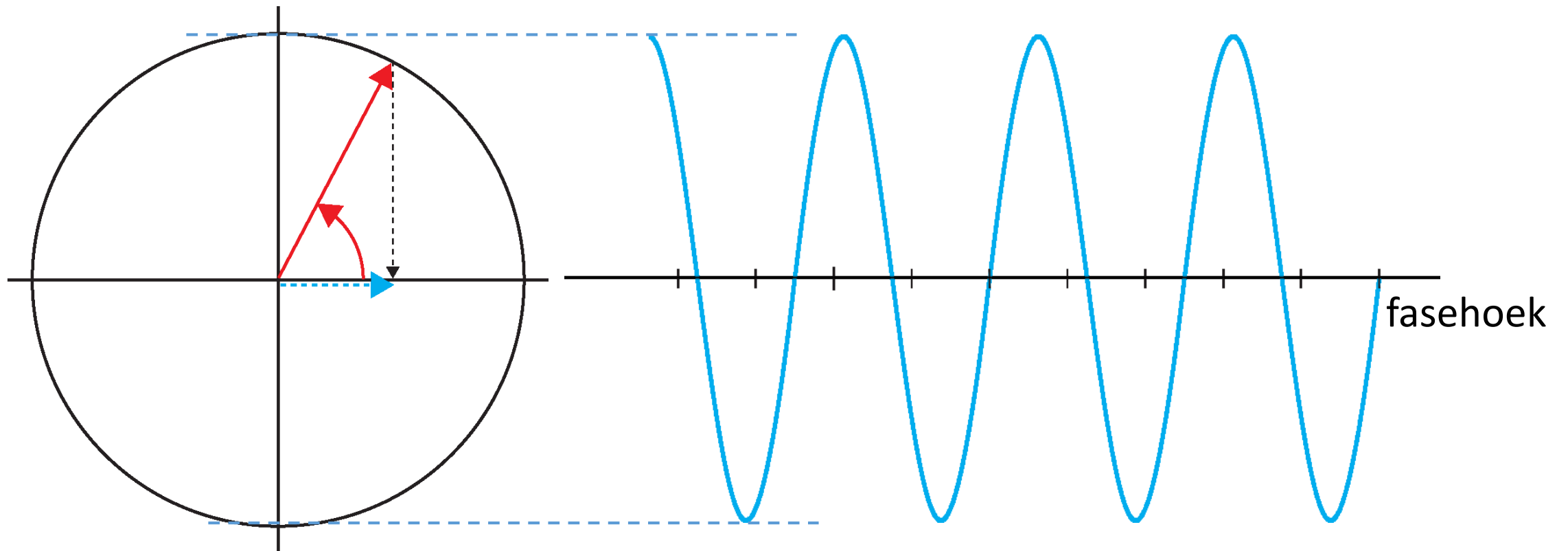
Voorbeeld: VNWA van DG8SAQ



Basisbegrippen DSP

Analoog naar digitaal converter (ADC)

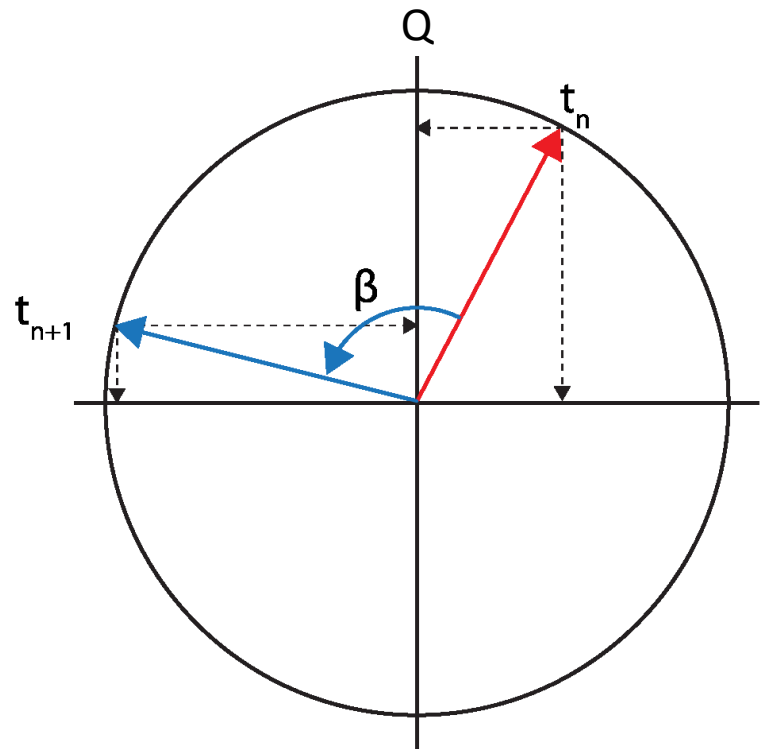
Wisselspanning kan worden voorgesteld als draaiende (eenheids)vector



Basisbegrippen DSP

Analoog naar digitaal converter (ADC)

Wisselspanning kan worden voorgesteld als draaiende (eenheids)vector



$$\text{fasehoek}(t_n) = \text{atan}\{Q_n/I_n\}$$

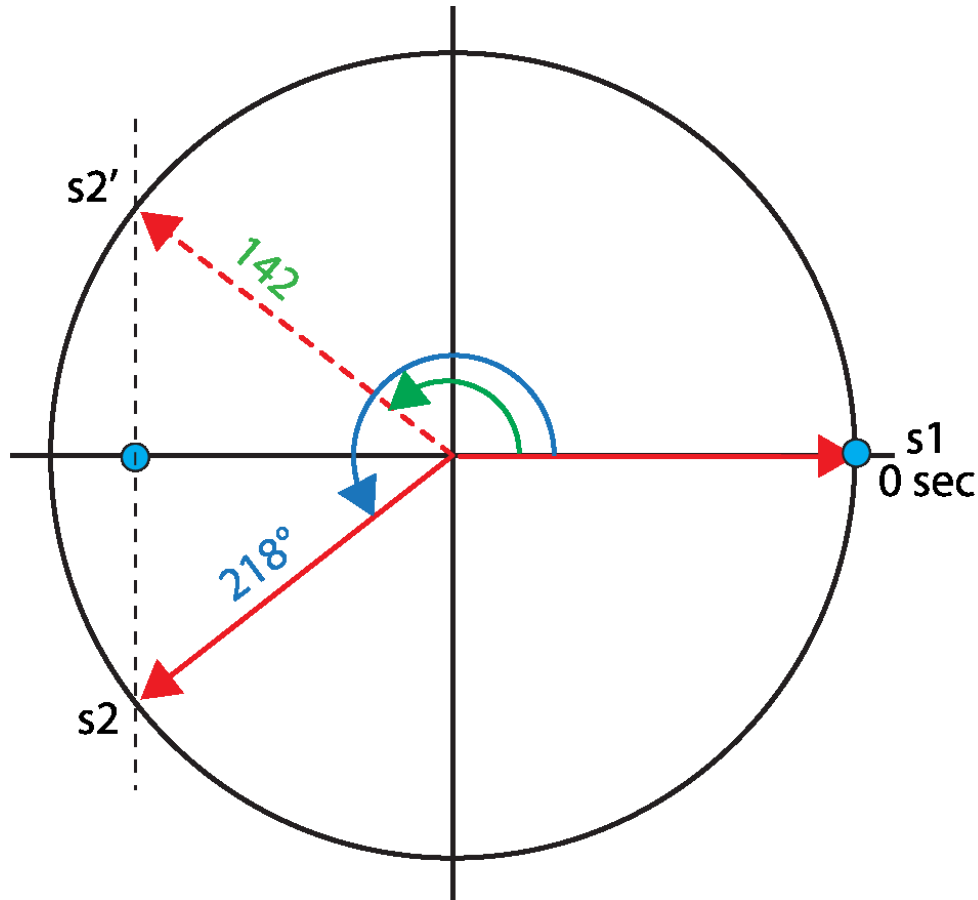
$$\text{fasehoek}(t_{n+1}) = \text{atan}\{Q_{n+1}/I_{n+1}\}$$

$$\beta = \text{fase}(t_{n+1}) - \text{fase}(t_n)$$

$$\text{periodetijd} = \frac{360^\circ}{\beta} \times \text{sample interval}$$

Basisbegrippen DSP

Analoog naar digitaal converter (ADC): alias probleem



sample clock: $f = 5 \text{ Hz}$ $t = 0,2 \text{ sec}$

signaal 1: $f = 3 \text{ Hz}$

1 rotatie in $0,33 \text{ sec}$,

in $0,2 \text{ sec}$: $360 \times 0,2 / 0,33 = 218^\circ$

$s_1 > s_2$: $360/218 \times 0,2 \Rightarrow 3 \text{ Hz}$

$s_1 > s_2'$: $360/142 \times 0,2 \Rightarrow 5 \text{ Hz}$

$s_2 + 360^\circ$: $360/578 \times 0,2 \Rightarrow 8 \text{ Hz}$

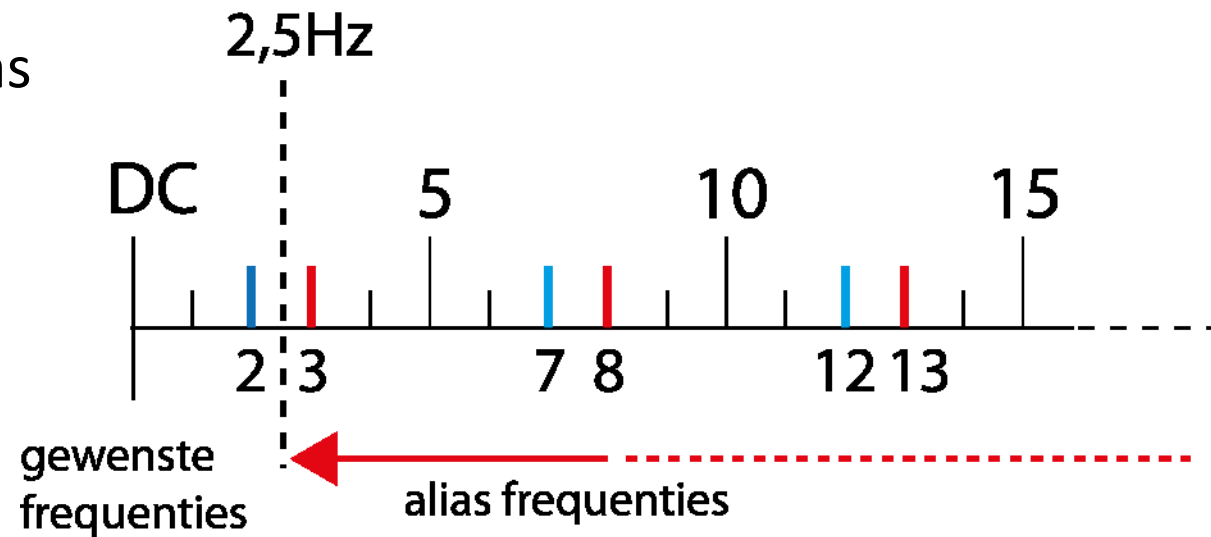
$s_2' + 360^\circ$: $360/502 \times 0,2 \Rightarrow 7 \text{ Hz}$

Basisbegrippen DSP

Analoog naar digitaal converter (ADC): alias probleem

sample clock 5 sps

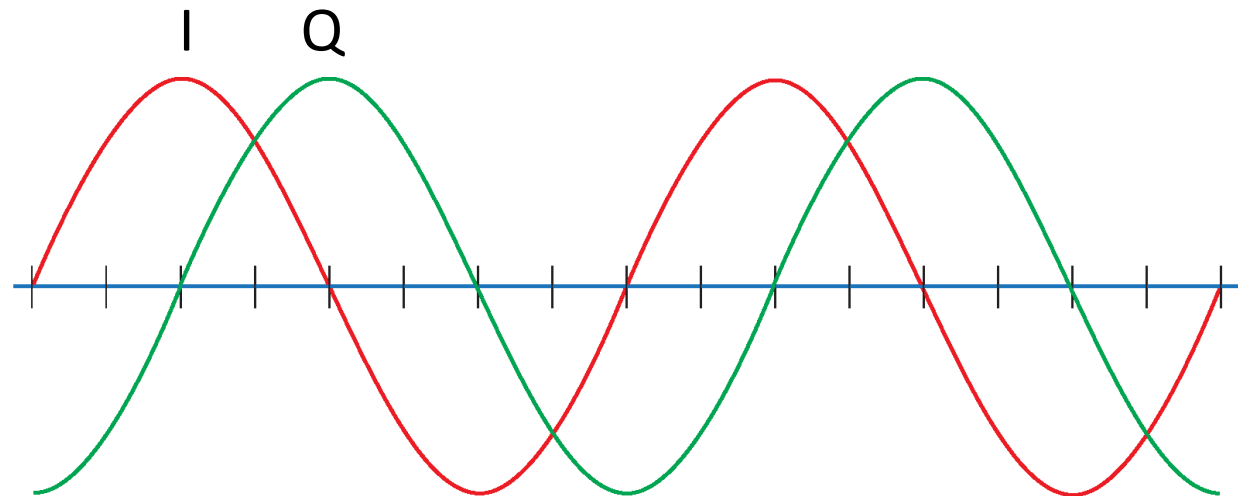
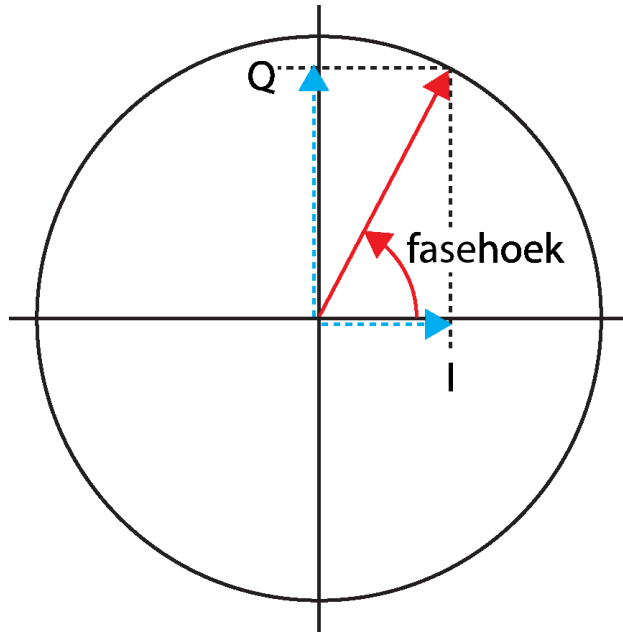
inputfrequentie 3 Hz -> alias



Basisbegrippen DSP

Analoog naar digitaal converter (ADC): alias probleem

Quadratuursignaal voor fasebepaling

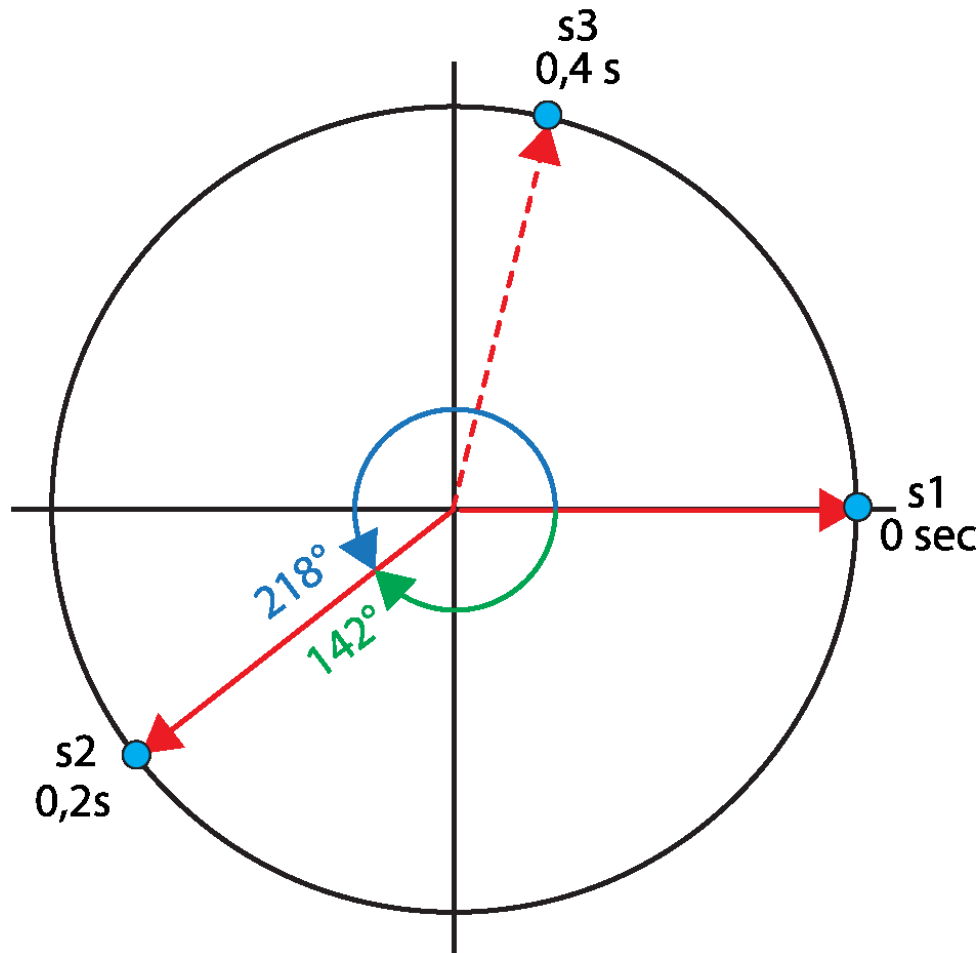


I - in fase (cos fasehoek)

Q - quadratuur fase (sinus fasehoek)

Basisbegrippen DSP

Analoog naar digitaal converter (ADC): alias probleem



sample clock: $f = 5 \text{ Hz}$ $t = 0,2 \text{ sec}$

signaal 1: $f = 3 \text{ Hz}$

1 rotatie in $0,33 \text{ sec}$,

in $0,2 \text{ sec}$: $360 \times 0,2 / 0,33 = 218^\circ$

s2 linksom: $360 / 218 \times 0,2 = 0,33 \text{ sec}$ (3 Hz)

s2 rechtsom: $360 / -142 \times 0,2 = -0,50 \text{ sec}$ (-2 Hz)

s2 + 360° (l): $360 / 578 \times 0,2 = 0,125 \text{ sec}$ (8 Hz)

s2 + 360° (r): $360 / -502 \times 0,2 = -0,143 \text{ sec}$ (-7 Hz)

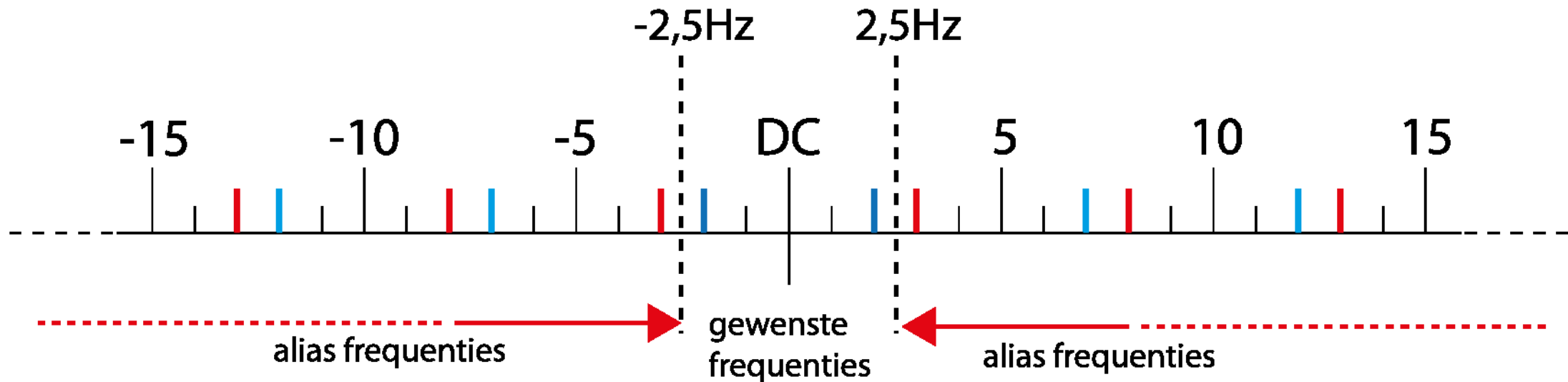
Basisbegrippen DSP

Analoog naar digitaal converter (ADC): alias probleem

alias frequenties

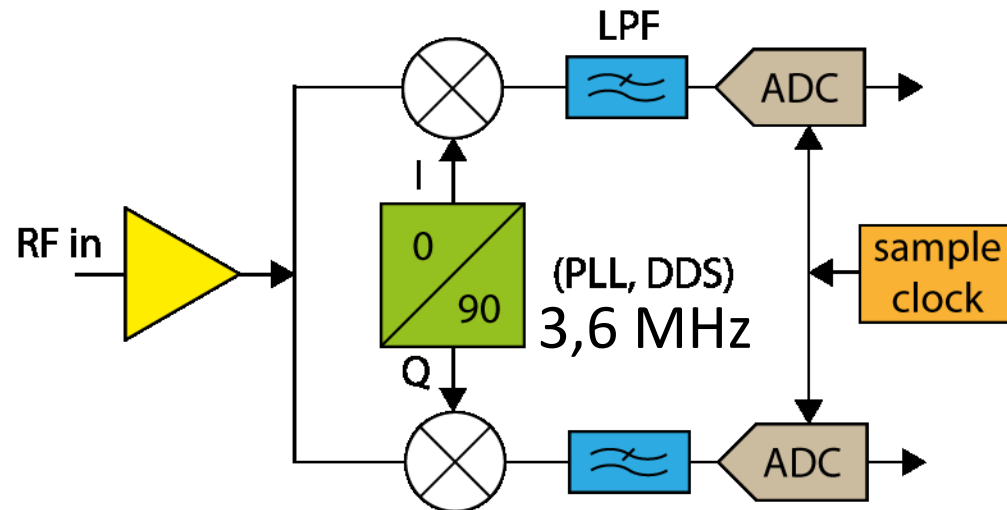
sample clock: 5 sps

gewenste IQ sampling



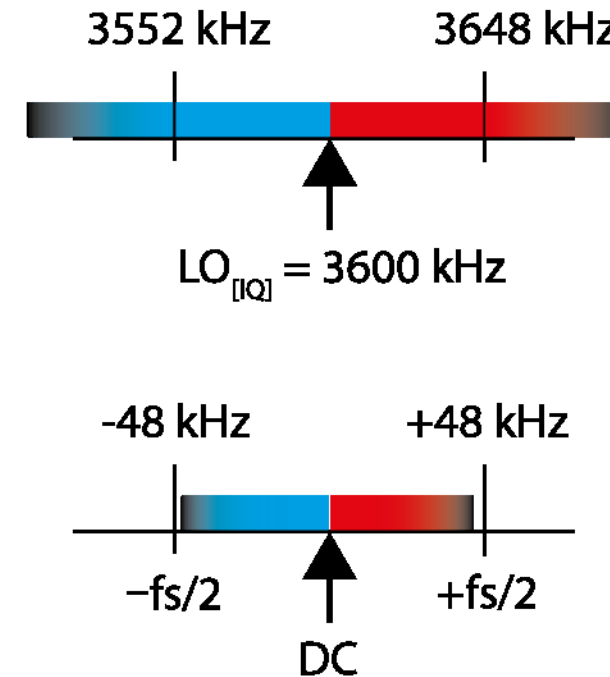
Basisbegrippen DSP

Analoog naar digitaal converter (ADC)



LPF: $f_c < 96 \text{ kHz}$
sample clock 96 ksps

sample rate $f_s = 96 \text{ ksps}$



Basisbegrippen DSP

Interface bandbreedte:

data stream uit ADC naar DSP:

aantal bits / seconde = sample rate x aantal bits

24 bits x 192 ksps = 4,6 Mb/sec

14 bits x 80 Msps = 1,12 Gb/sec

USB 2.0: 200 - 300 Mb/s (spec: 480 Mb/s)

USB 3.0 spec: 4800 Mb/s

Ethernet cat5/5e: 50/500 Mb/s (spec: 100/1000 Mb/s)

Basisbegrippen DSP

Samenvatting digitaliseren

- aantal (effectieve) bits
- $S/N \text{ ratio} = 6 \cdot N + 1,7\text{dB}$
- clock jitter
- sample rate bepaalt eenduidig frequentiebereik
- analoge filters tot $f_s/2$ of f_s (quadratuur)
- alias frequenties boven f_s bruikbaar?
- bandbreedte ADC naar DSP

Basisbegrippen DSP

Digitale Signaalverwerking (DSP)

- versterken: getallen vermenigvuldigen met (constante / variabele) factor
- mixen: getallen vermenigvuldigen met een sinus (cosinus)
- filteren: combineren van eerdere of latere sample waarden
- detectie van amplitude- en faseverandering (frequentie)
- oscillator functie: sinus (cosinus) getallenreeks genereren (CORDIC-algoritme, DDS)
- sample rate veranderen
-

DSP = REKENEN

Basisbegrippen DSP

Digitale Signaalverwerking (DSP)

Voordeel:

- aanpassen functionaliteit: software
- DSP-platform kan generiek zijn, 'eenvoudige' hardware updates
- niet afhankelijk van RF-frequentie
- complexe modulatiepatronen en coderingen
-

Basisbegrippen DSP

Digitale Signaalverwerking (DSP)

Nadeel (voor amateurs):

- computer vaardigheid, prog
- hardware = SMD (vergrootglazen, loeplampen, alzheimer, tools)



DSP technieken

Fourier transformatie:
Elk periodiek signaal kan
worden samengesteld uit
(co)sinusvormige signalen

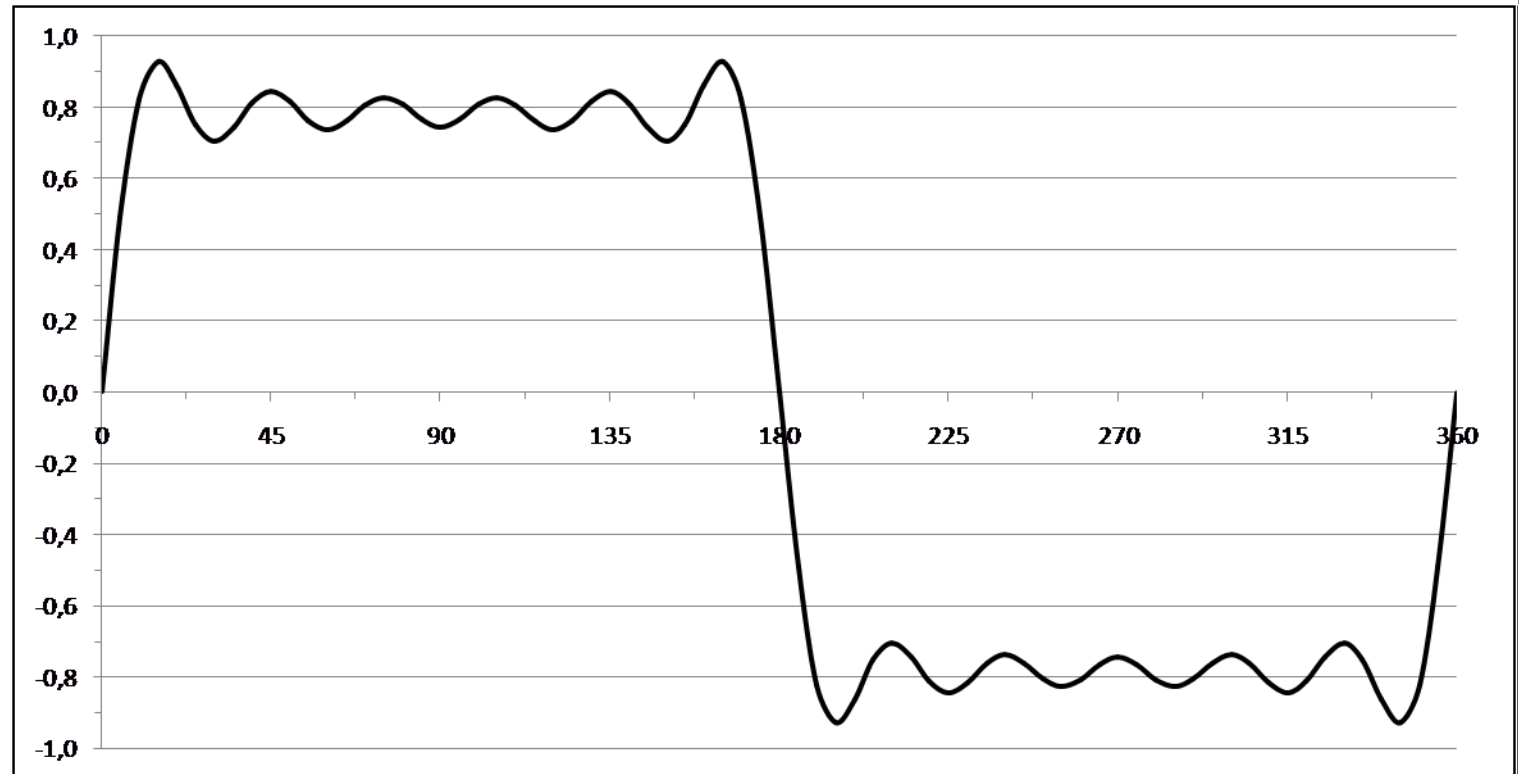
blokgolf:

$$A = DC + f(0) + \frac{1}{3} \cdot 3f + \frac{1}{5} \cdot 5f + \dots$$

zaagtand:

$$f(0) - \frac{1}{2} \cdot 2f + \frac{1}{3} \cdot 3f - \frac{1}{4} \cdot 4f + \dots$$

Oneven harmonischen t/m 11



Symmetrische golfvormen: alleen oneven harmonischen

DSP technieken

Fourier:

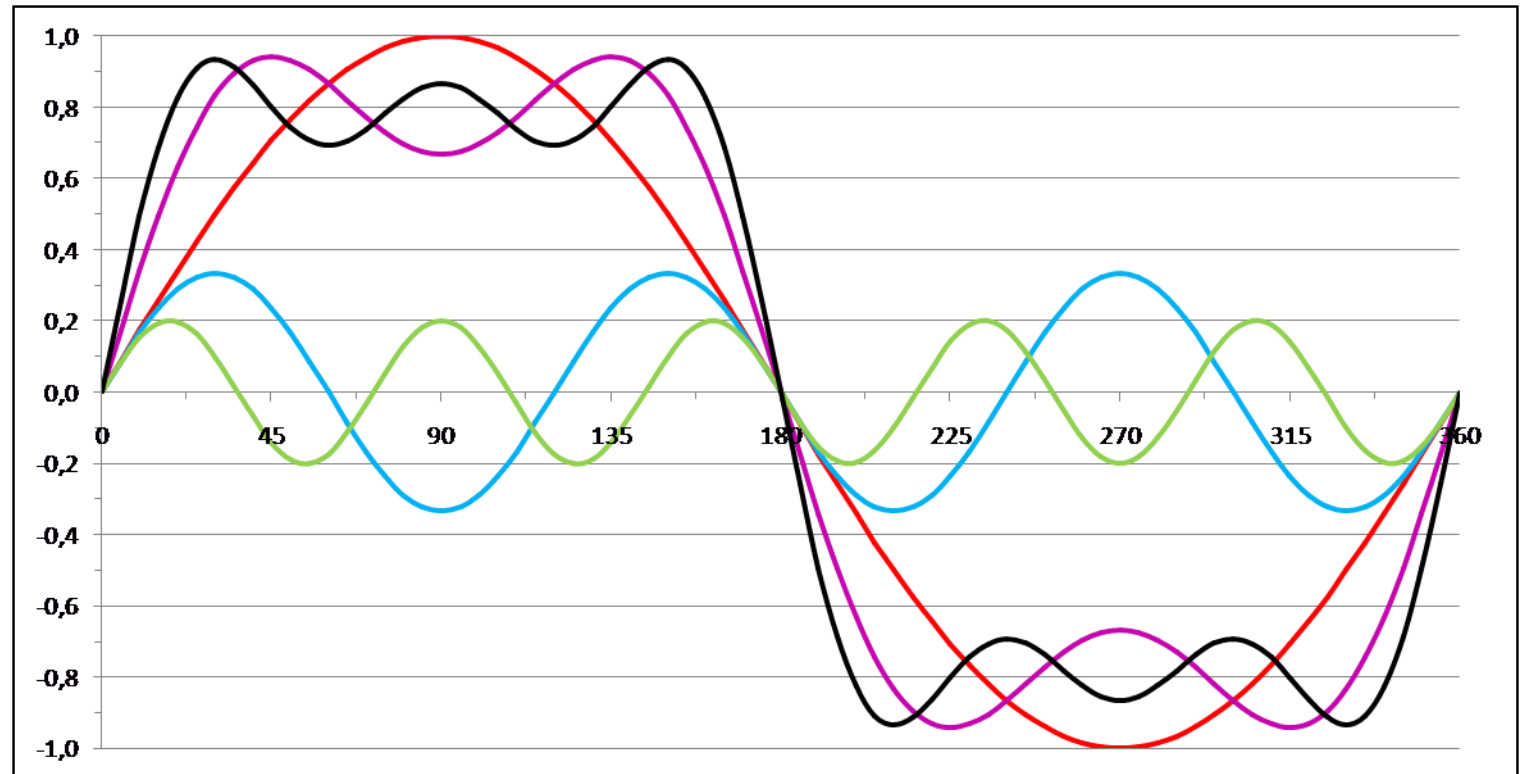
Elk periodiek signaal kan worden samengesteld uit (co)sinusvormige signalen

blokgolf:

$$A = DC + f(0) + \frac{1}{3} \cdot 3f + \frac{1}{5} \cdot 5f + \dots$$

zaagtand:

$$f(0) - \frac{1}{2} \cdot 2f + \frac{1}{3} \cdot 3f - \frac{1}{4} \cdot 4f + \dots$$



Symmetrische golfvormen: alleen oneven harmonischen

DSP technieken

(Inverse) Fourier transformatie:

Uit een sample reeks kunnen de frequentie-componenten berekend worden:
conversie van het tijddomein naar het frequentiedomein en vice versa



tijddomein



frequentiedomein

DSP technieken

Fourier transformatie:

- wiskundige conversie van een periodiek gesampled signaal naar frequentiedomein
- berekent voor elke frequentiestap ('bin') binnen het frequentiebereik de amplitude en fase in de 'bin'
- frequentie resolutie (bandbreedte bins): sample rate / aantal samples (N)

Voorbeeld:

digitaliseren met 100 ksps gedurende 10 msec geeft 1000 samples.

reëel: frequentiebereik DC – 50 kHz	nr bins: $N/2 = 500$	resolutie $50 \text{ kHz} / 500 = 100 \text{ Hz}$
complex: DC – 100 kHz	$N = 1000$	$100 \text{ kHz} / 1000 = 100 \text{ Hz}$

aanname: het ingangssignaal is periodiek en continu

DSP technieken

Fourier transformatie (reëel)

input:

- 16 reële sample waarden

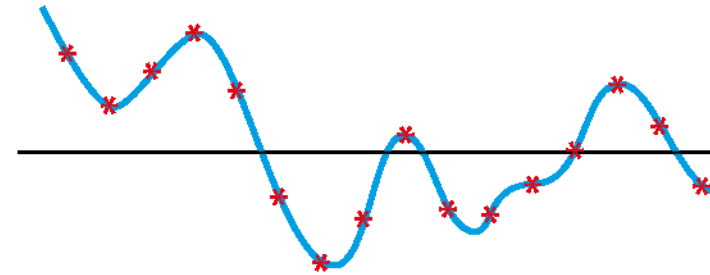
output:

- 8 x 2 coëfficiëntenparen I/Q

amplitude per frequentiebin: $[I, Q]$ + Pythagoras

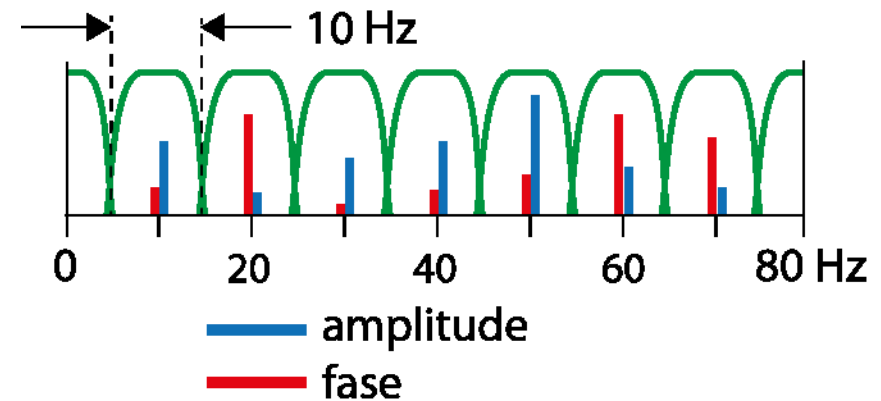
fase per frequentiebin: $\psi = \text{atan} (Q / I)$

$f_s = 160 \text{ sps}$



$N = 16$ samples, interval 0,1 sec

Fourier transformatie: $(N/2 + 1)$ frequentie bins



DSP technieken

Fourier transformatie (complex) op kwadratuursignaal (I/Q)

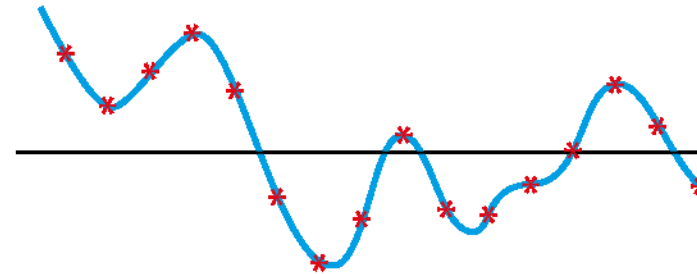
input:

- 16 reële (I) + 16 imaginair sample waarden

output:

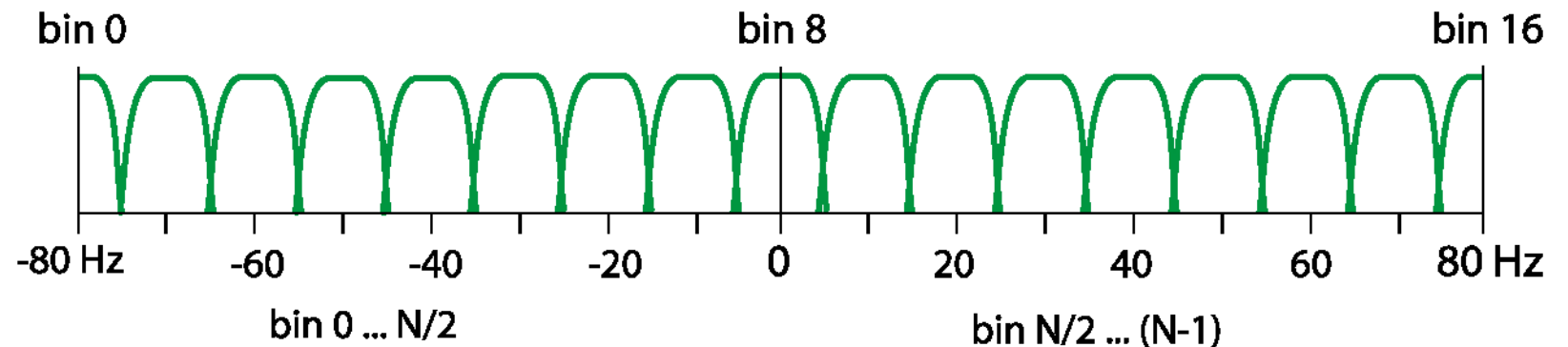
- 16 x 2 coëfficiëntenparen frequentie en fase

$f_s = 160$ sps



$N = 16$ samples, interval 0,1 sec

Complexe Fourier transformatie: N frequentie bins



DSP technieken

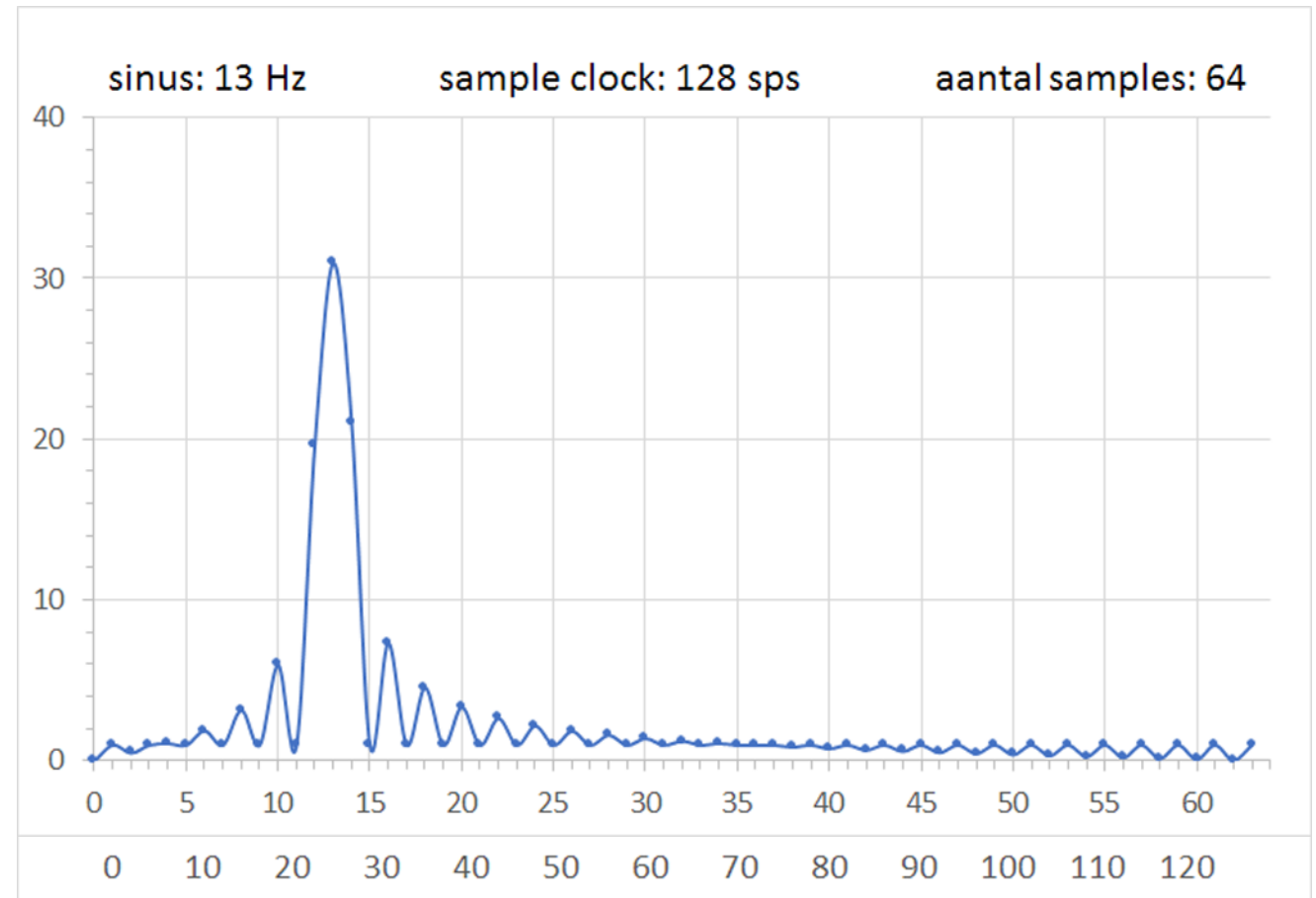
Fourier transformatie

Theorie: input signaal is
periodiek en continu

Praktijk: discontinue, niet
periodiek

Gevolg: FFT afwijkingen

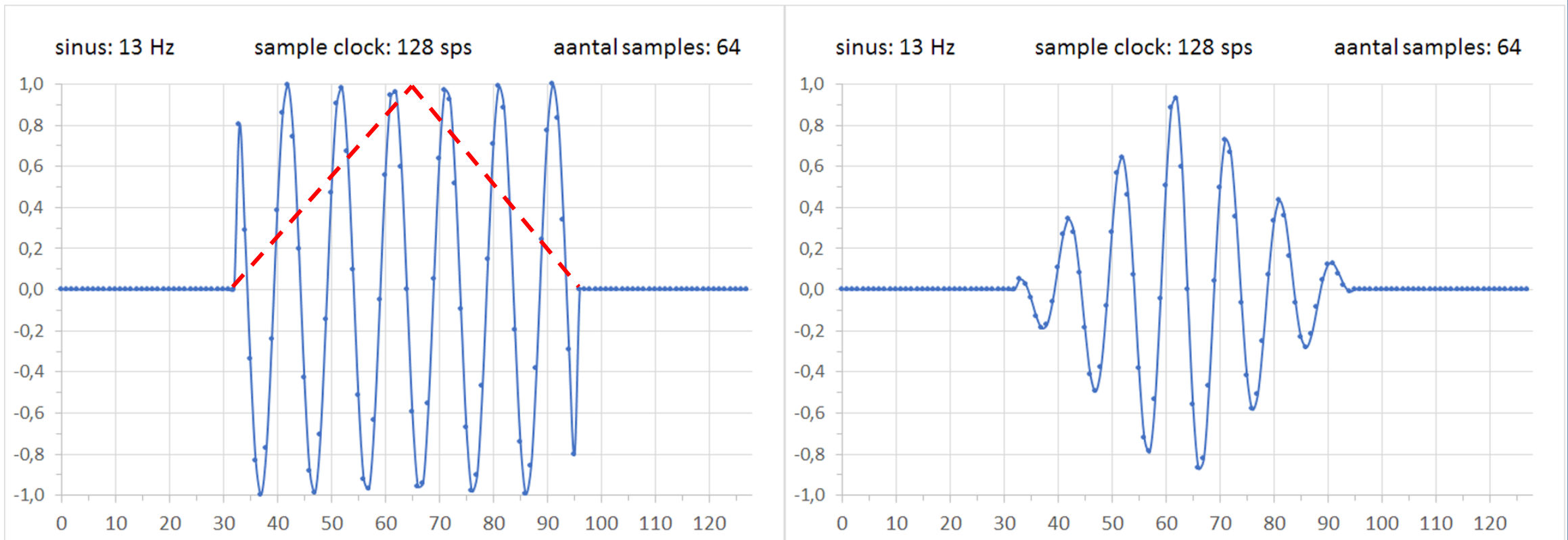
aantal bins: sample rate / 2 = 64
resolutie: $\frac{1}{2} f_s / \text{bins} = 64 / 64 = 1 \text{ Hz}$



DSP technieken

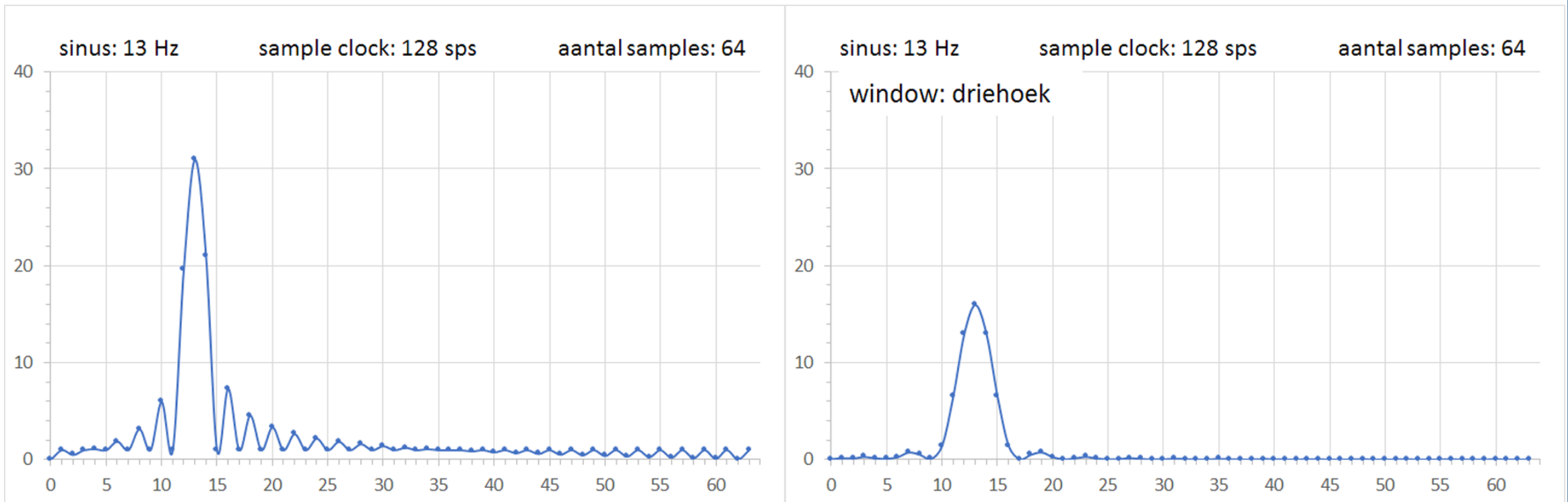
Fourier transformatie

FFT afwijkingen beperken door 'windowing'



DSP technieken

Fourier transformatie



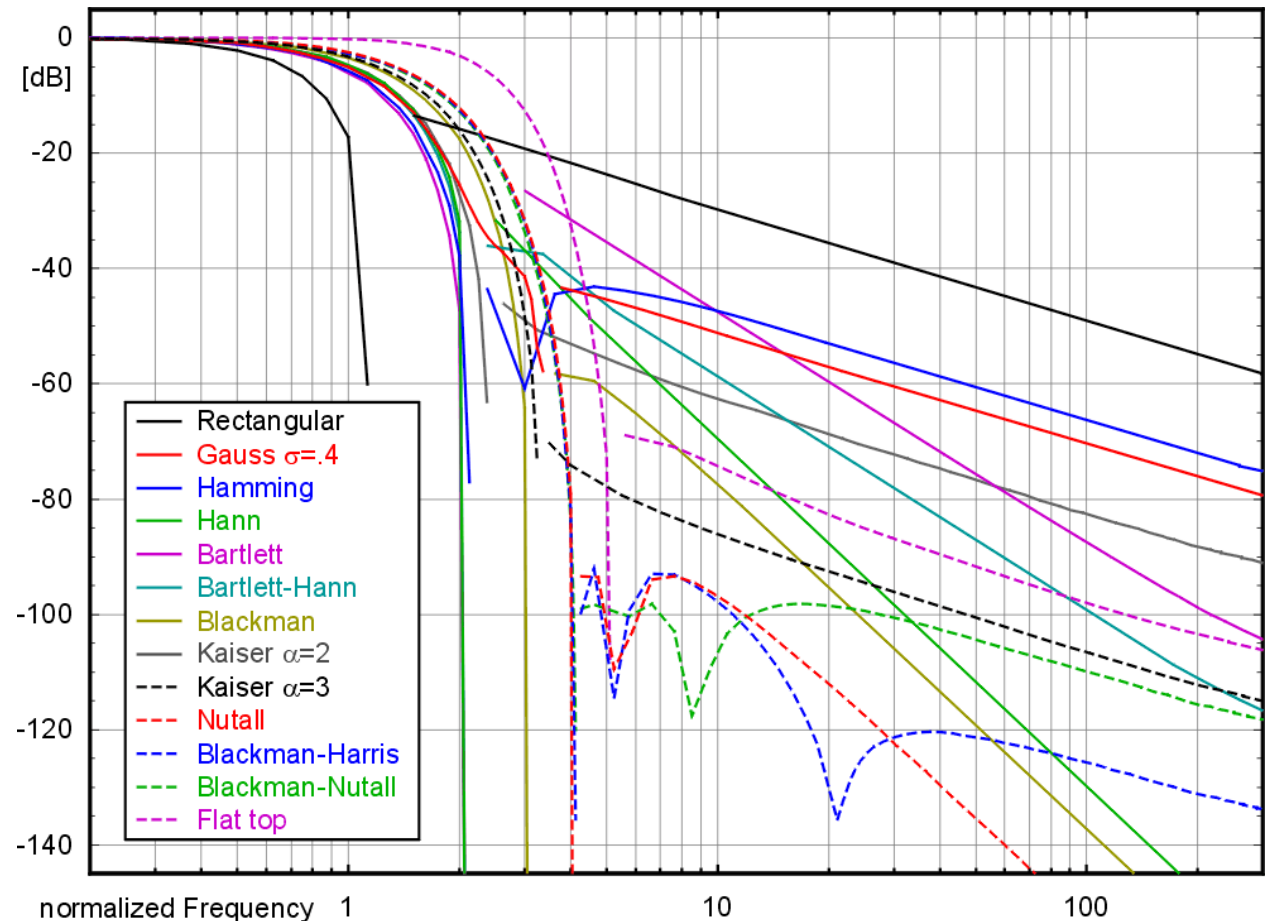
- minder spectrale ruis
- amplitude en frequentie onnauwkeuriger

DSP technieken

Fourier transformatie

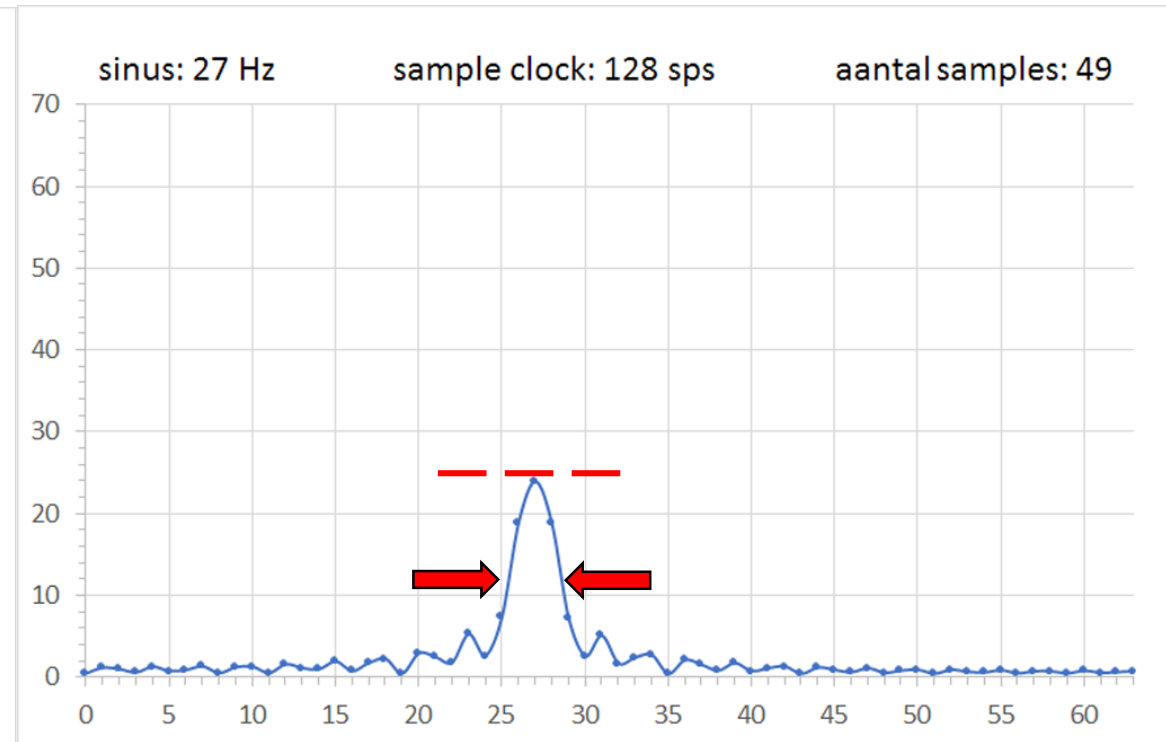
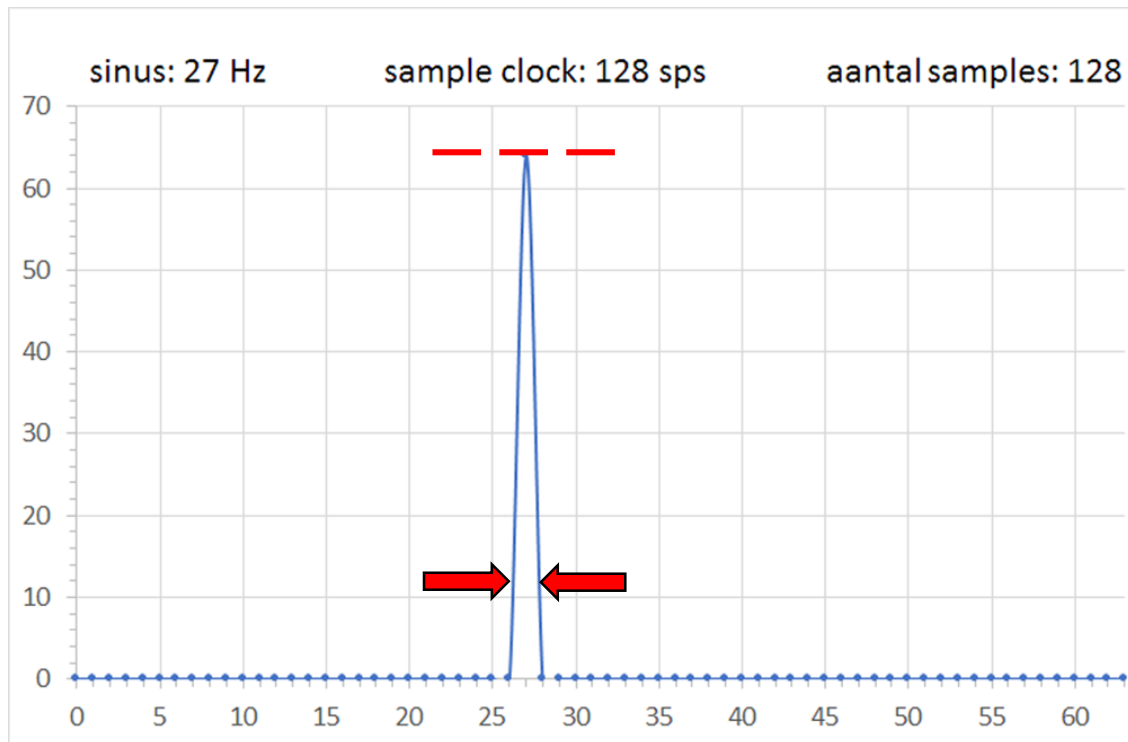
Windowing voor discontinue signalen

- invloed op amplitude
- invloed op spectrale spreading
 - Kaiser
 - Tukey
 - rechthoekig
 - Bartlett (driehoek)
 - Hamming
 - Hanning
 - Blackmann-Harris
 - Flattop



DSP technieken

Fourier transformatie

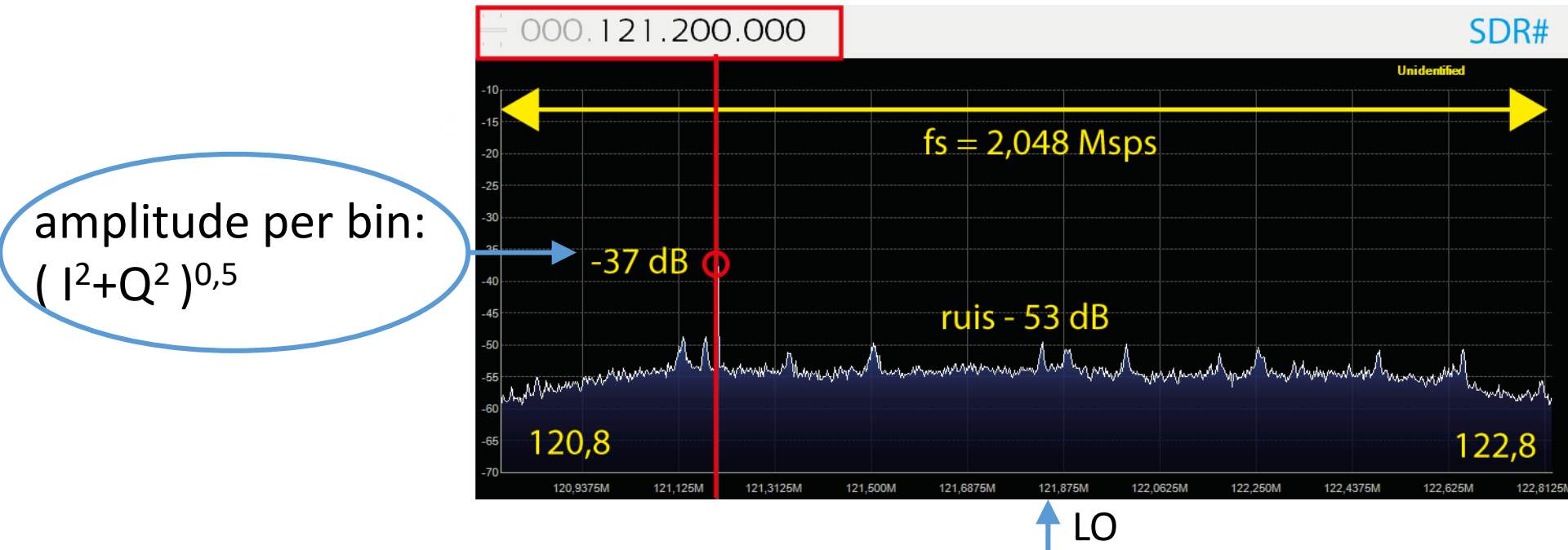


aantal bins: $N / 2 = 64$

resolutie: $\frac{1}{2} f_s / \text{bins} = 1 \text{ Hz}$

DSP technieken

SDR spectrum display: IQ magnitude per bin



$f_s = 2,048 \text{ Msps}$ aantal samples = 2048

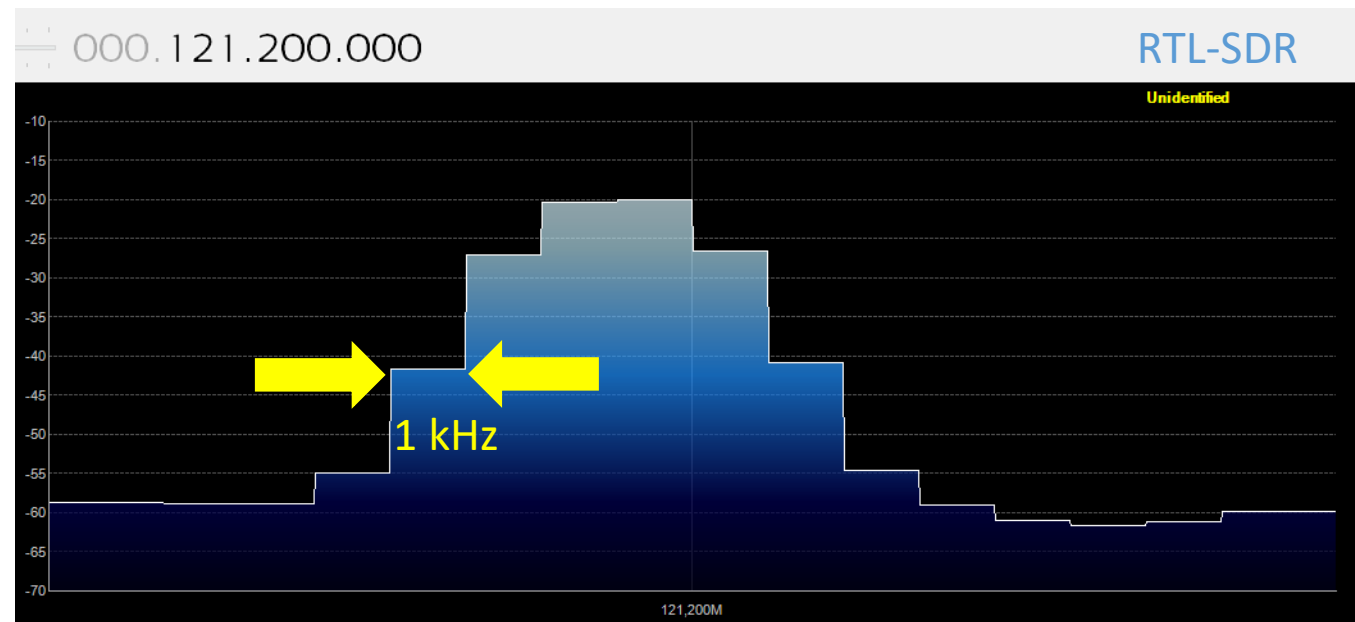
$f_s/N = 1000$ frequentie bins

bandbreedte bins: $f_s/\text{bins} = 2,048 \text{ kHz}$

DSP technieken

$F_s = 2,048 \text{ Msp/s}$ aantal samples = 2048

Aantal bins: $2,048\text{M} / 2048 = 1000$ bandbreedte bin: $f_s/\text{bins} = 1000 \text{ Hz}$

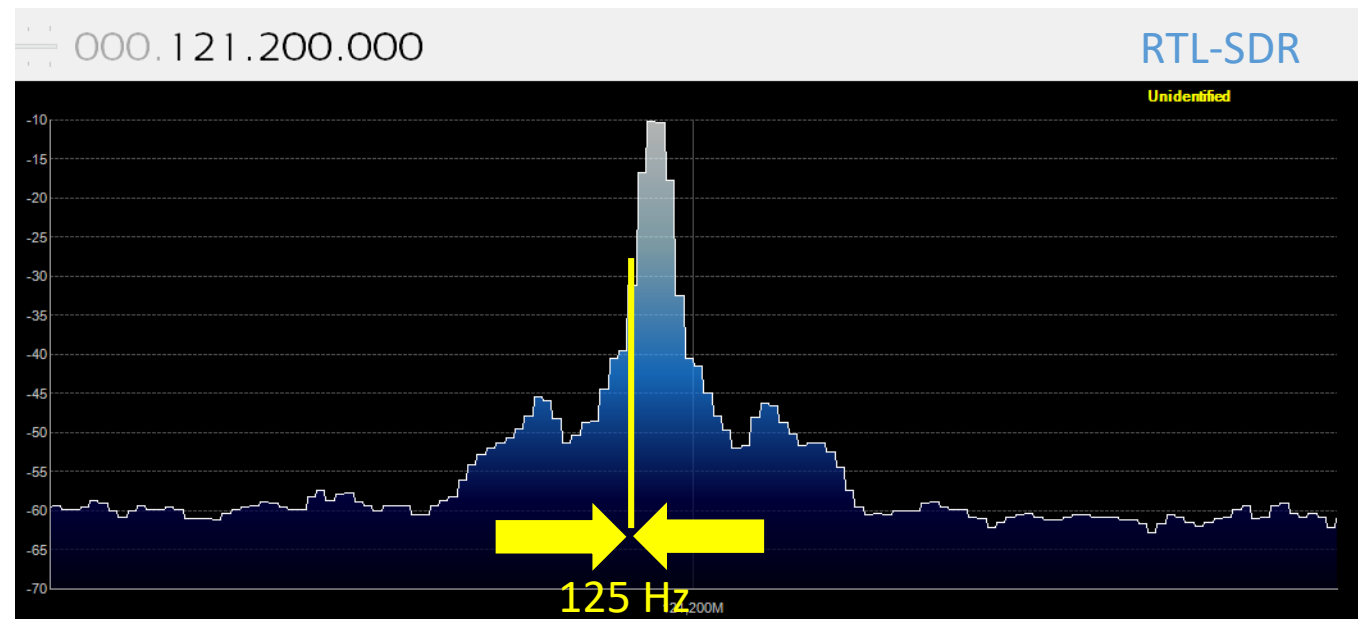


AM signaal $\pm 6 \text{ kHz}$

DSP technieken

$F_s = 2,048 \text{ Msp/s}$ aantal samples = 16384

Aantal bins: $2,048\text{M} / 16384 = 8000$ bandbreedte bin: $f_s/N = 125 \text{ Hz}$



AM signaal $\pm 6 \text{ kHz}$

DSP technieken

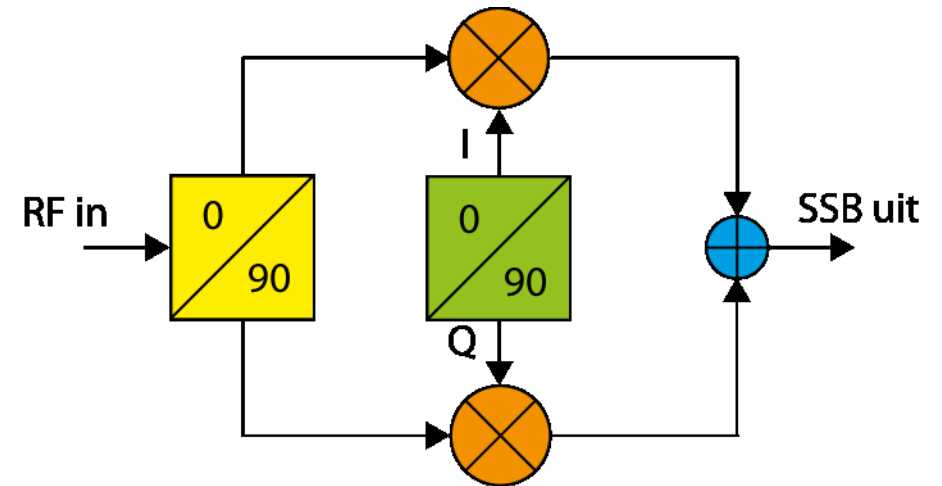
Frequenties converteren

Mixer:

- reëel: zowel som- als verschilproduct
- complex: alleen som- of verschilproduct (USB of LSB)

In DSP-termen:

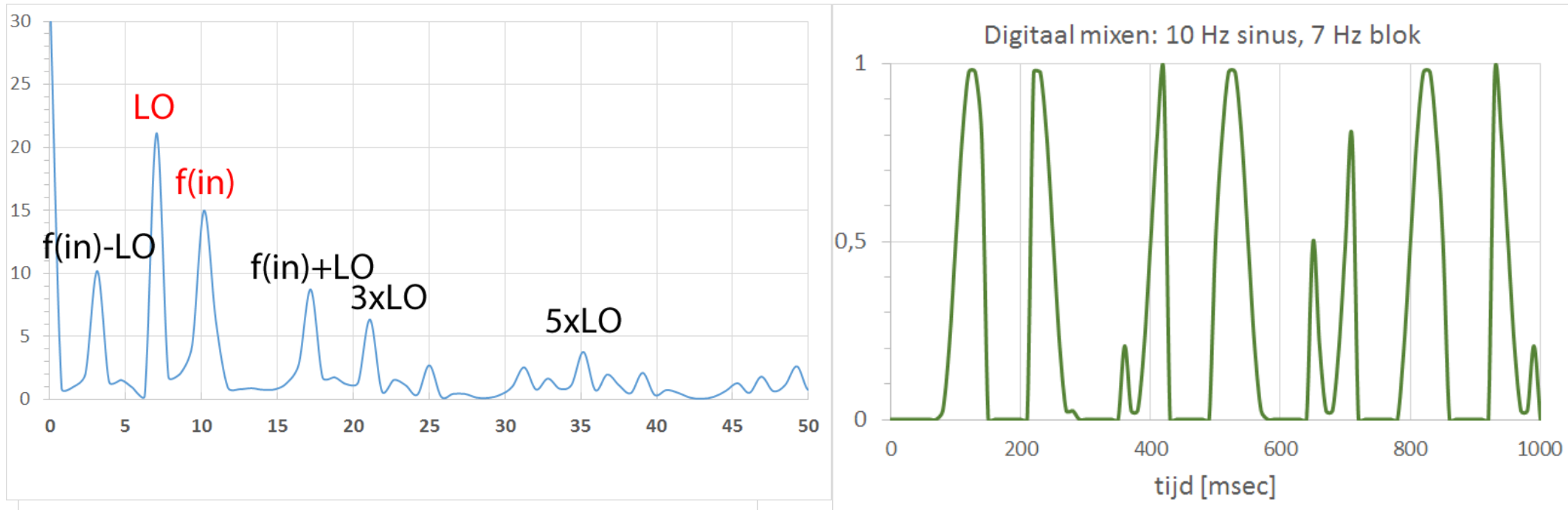
- vermenigvuldigen van complex RF-sigitaal met complex LO-sigitaal
- het (digitale)LO-sigitaal kan een blokgolf zijn (bijv. Tayloe-mixer)



DSP technieken

Frequenties converteren

Numerieke vermenigvuldiging van 7 Hz LO blok golf met een 10 Hz sinus:

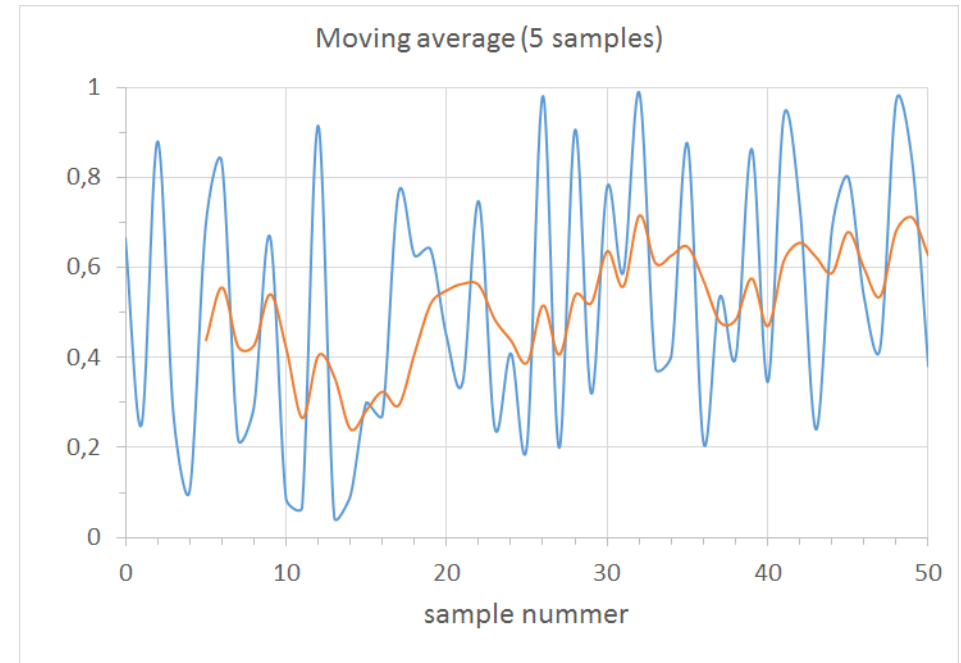
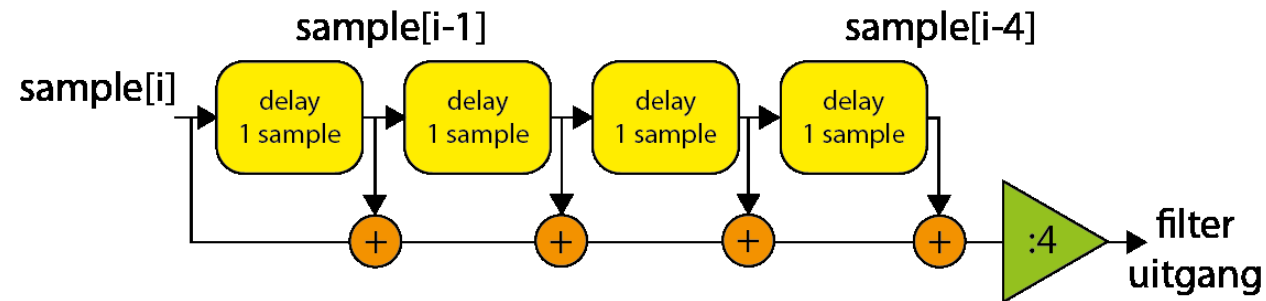


vermenigvuldiging met een LO-blokgolf: verschil 3 Hz, som 17 Hz, mengproducten 3, 5, 7 ... x LO

DSP technieken

Frequentie filters

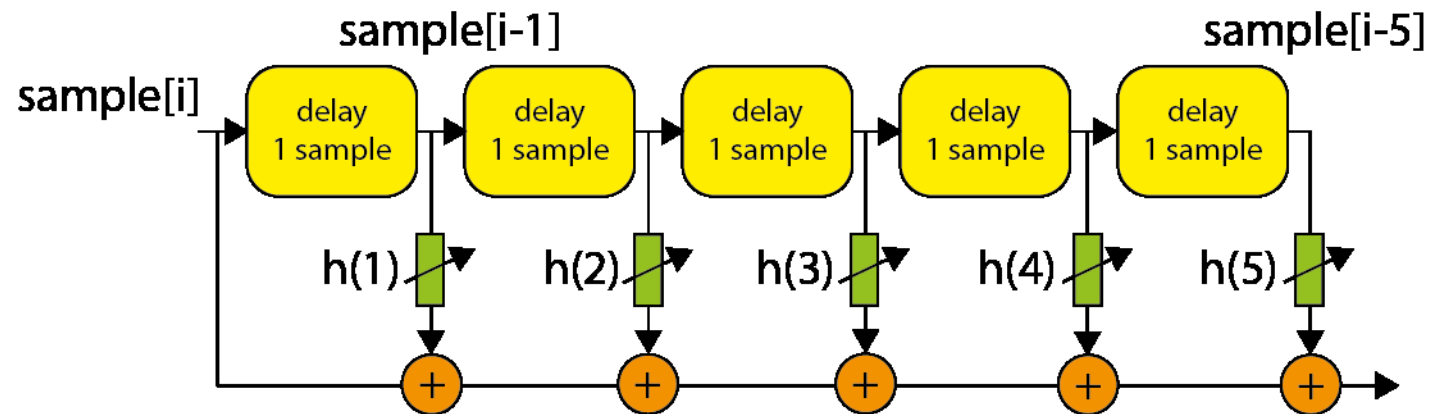
'moving average': voortschrijdend gemiddelde over meerdere samples, dempt uitschieters



DSP technieken

Frequentie filters

'Finite Impulse Response' filters (FIR)



- coëfficiënten $h(n)$ bepalen de doorlaatcurve
- $h(1) = h(5)$, $h(2) = h(4)$: symmetrisch filter heeft lineair fase verloop
- veel computertijd nodig voor scherp filter (veel taps)

DSP technieken

Frequentie filters

FIR filter – voorbeeld:

Ontwerp:

sample rate = 1000 sps

max frequentie: $f_s/2 = 500$ Hz

doorlaatband DC – 50 Hz, ripple < 2 dB

stopband 200 – 500 Hz, damping > -40 dB

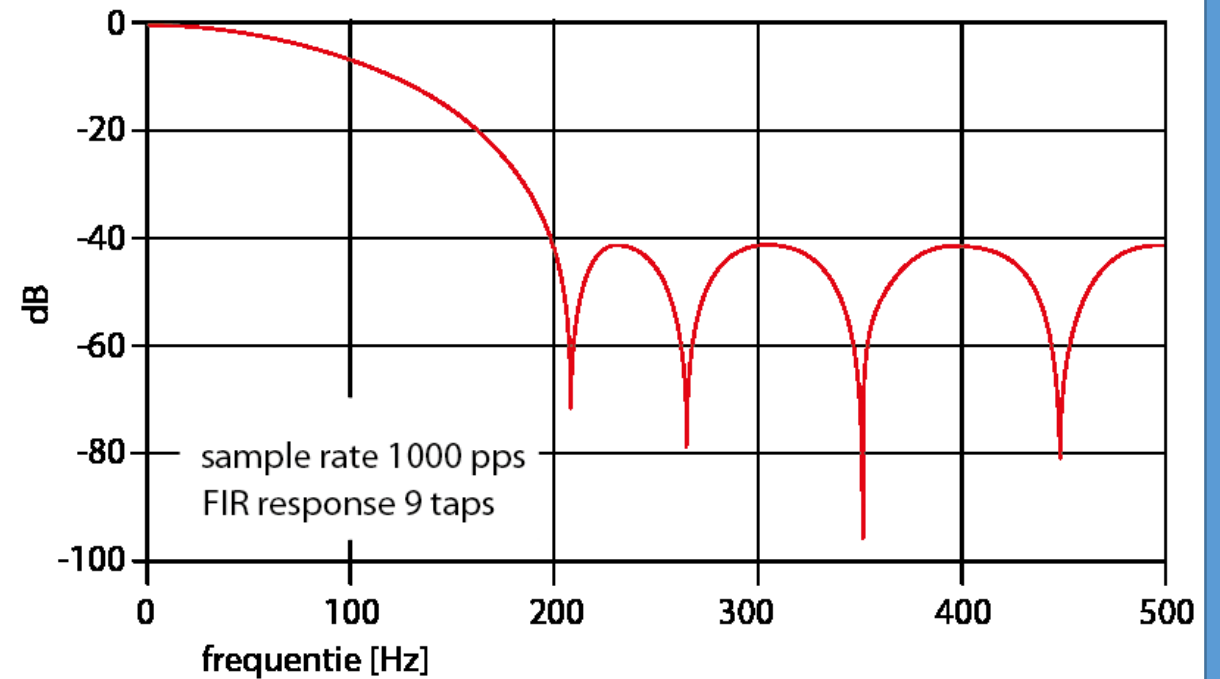
$$h(1) = h(9) = 0,023$$

$$h(2) = h(8) = 0,065$$

$$h(3) = h(7) = 0,121$$

$$h(4) = h(6) = 0,170$$

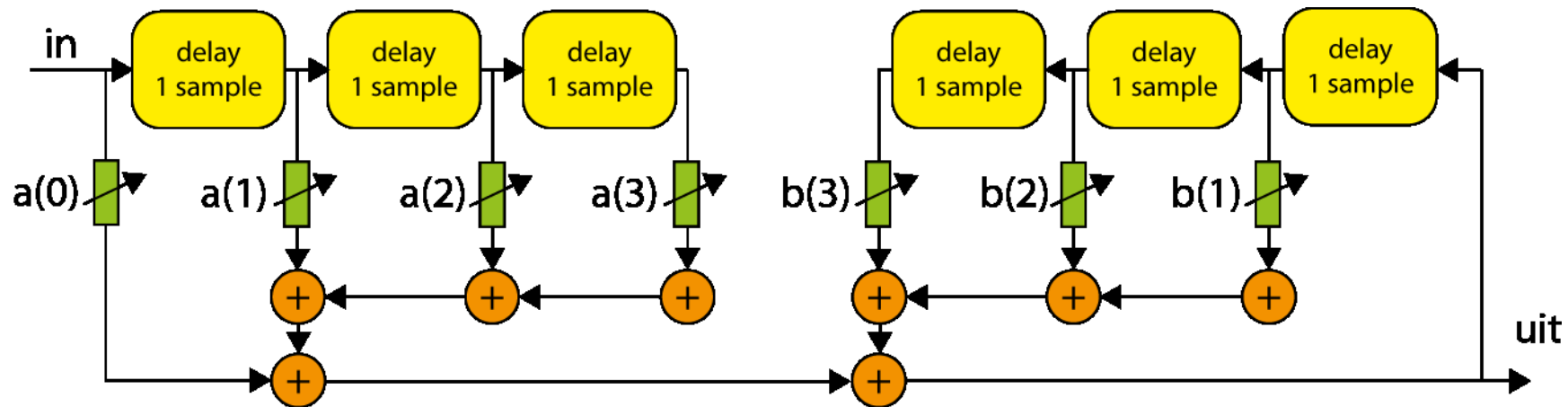
$$h(5) = 0,189$$



DSP technieken

Frequentie filters

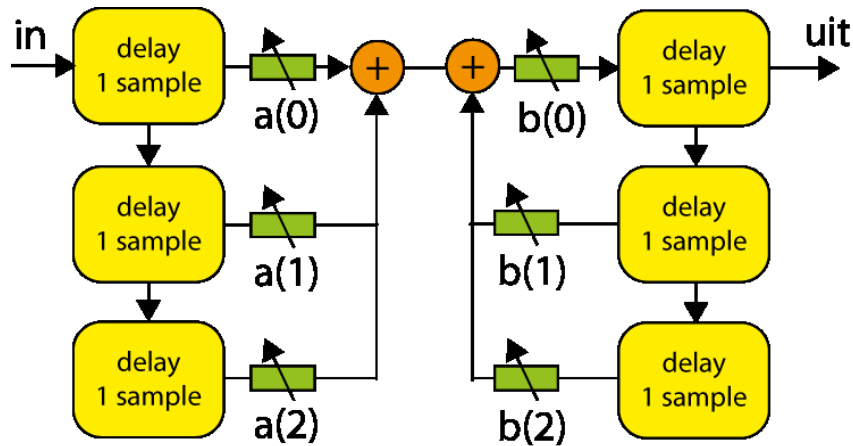
'Infinite Impulse Response' filters (IIR)



- geen lineair fase verloop
- weinig computertijd nodig voor scherp filter
- feedback kan tot instabiliteit leiden

DSP technieken

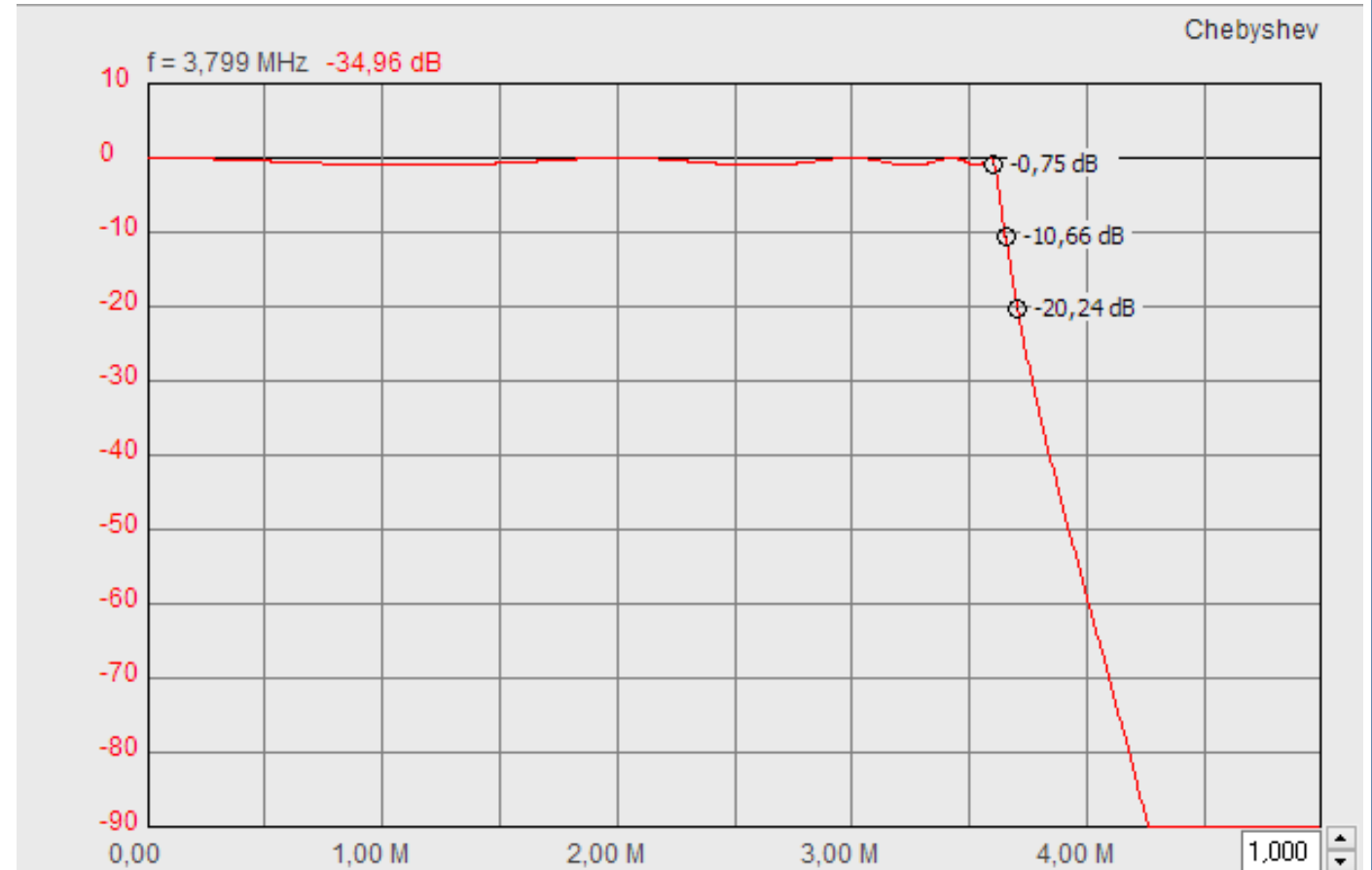
Frequentie filters



3600 kHz – 0,75 dB

3650 kHz – 10,7 dB

3700 kHz – 20,2 dB

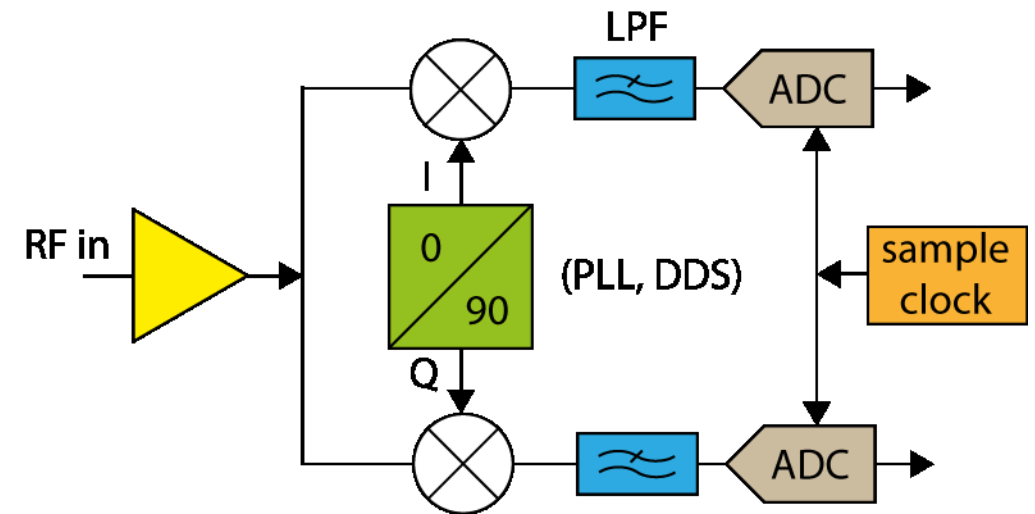


IOWA Hills IIR designer (<http://www.iowahills.com>)

DSP technieken

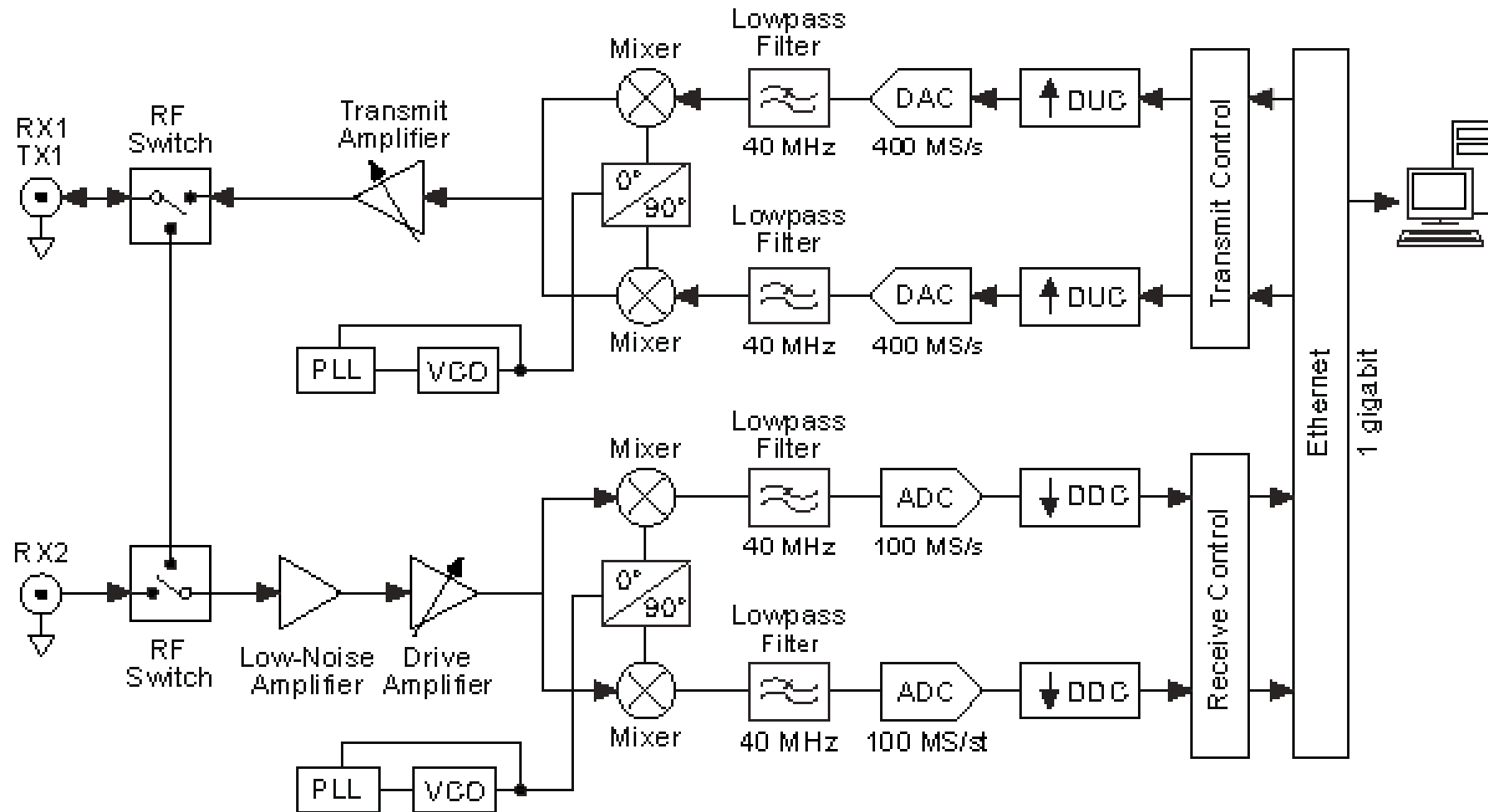
Mixer opties

- Mixers:
 - niet mixen – direct sampling (0 – 30 MHz)
 - mixen met quadratuur oscillator (I/Q) naar DC of IF
 - analoge mixer naar IF, dan digitaliseren
- analoog Nyquist low pass filter, DC... $\frac{1}{2}f_s$ / f_s
- I en Q ADC's, sample clock f_s



DSP technieken

Voorbeeld: USRP - eerst mixen, dan digitaliseren (quadratuur)



DSP technieken

SDR 'down sampling'

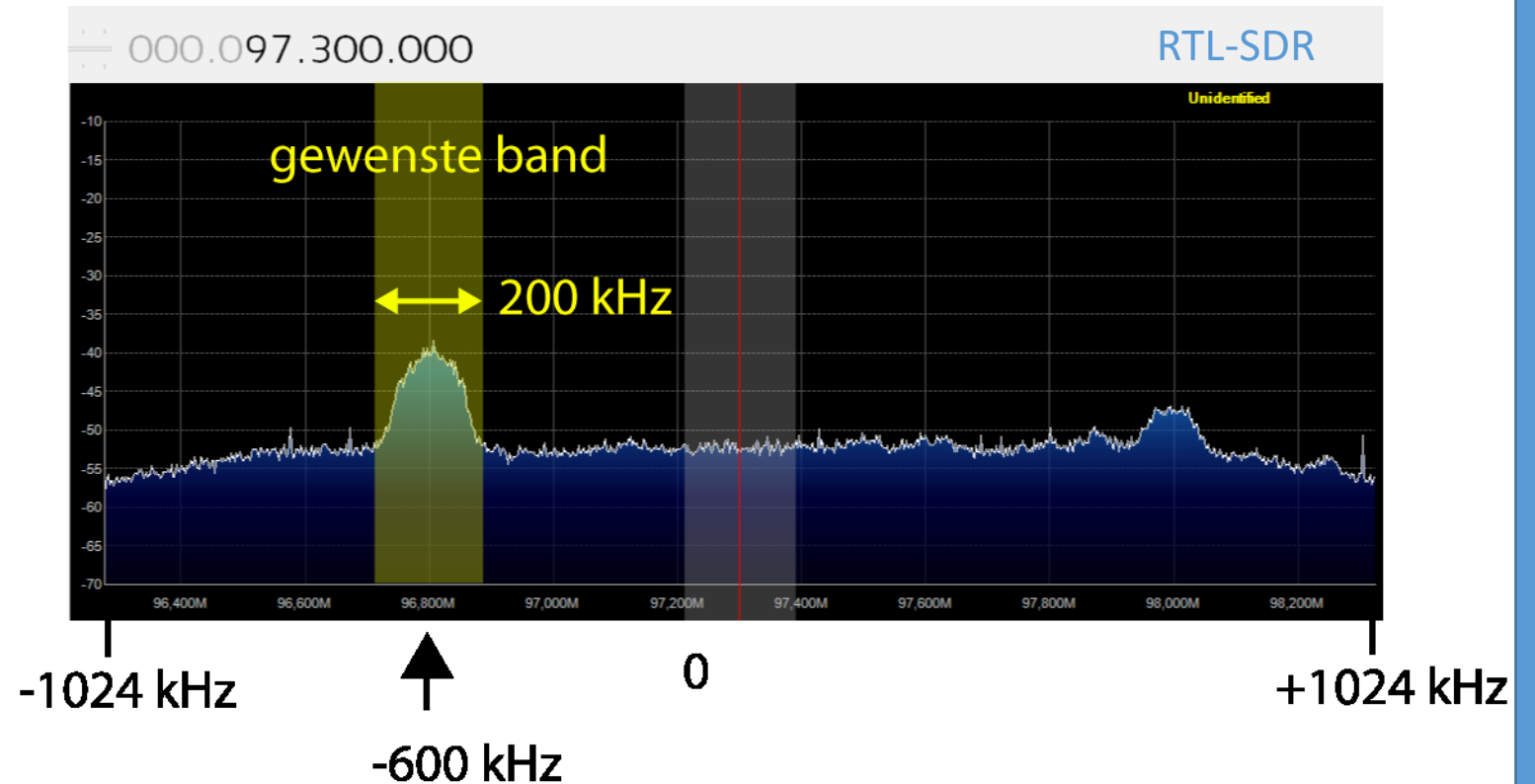
Samples van de ADC worden vermenigvuldigd met 'digitale' LO (IQ) met frequentie $f_c = 600$ KHz

benodigde sample rate voor 200 kHz (IQ) FIR filter ≥ 200 ksp/s

sample rate decimation:

- f_s (input): 2048 ksp/s
- f_s (output): $2,048 : 8 = 256$ ksp/s

Sample rate 2,048 Msps



DSP technieken

Digital down conversion (DDC)

- Q-mixen naar baseband (0 Hz)
- sample rate voor 6 kHz filter ≥ 6 ksps
- 1 van elke 16 samples gebruiken
- amplitude $(I^2 + Q^2)^{0,5}$
- audio bitstream: 44.1 / 48 ksps:
samples (waarde = 0) toevoegen

SNR 16 bits: 96 dB

Ruis in 6 kHz: -12 dB

SNR: -108 dB (≈ 18 bits)

Audiokaart 16 bits,
sample rate = 96 ksps



DSP technieken

Digital down conversion (DDC)

- quantiseringsruis over gehele bandbreedte
- decimation door weglaten van samples
- FIR filter voor gewenste (de)modulatie bandbreedte
- halvering sample rate ≈ 3 dB minder q-ruis

RTL-SDR: 8 bits, $f_s = 2,4$ Msps $\rightarrow$ SNR $\approx 6 \times 8 = 48$ dB

decimation 1/4 (600 ksps) : SNR = 48 dB + 2 x 3 dB = 54 dB ≈ 9 bits

.....

decimation 1/1024 (2,3 ksps): SNR = 48 dB + 10 x 3 dB = 78 dB ≈ 13 bits

DSP technieken

Conclusies:

- *Waarom Software Defined Radio?*
 - toekomst: digitaal
 - prestaties digitaal <> analoog: software en processing power
 - software verloopt niet
 - hardware <> software kosten
 - geen (digitale) kennis vereist om te opereren
 - upgrades, functionaliteit
- nieuwe kennis voor de amateur: digitale technieken, programmeren, linux (GNU-radio),
- zelfstudie: internet

DSP technieken

Samenvatting presentatie

- Waarom *Software Defined* Radio?
- Principes van digitaliseren (Nyquist limiet)
- Eigenschappen van ADC's (ruis, resolutie)
- Fourier transformatie (tijddomein <> frequentiedomein, windowing)
- Reële en complexe signalen
- Mixen (DC, lage IF)
- Digitale filters (FIR, IIR)
- Sample rate decimatie (SNR verbetering)

Samenvatting gehele presentatie:

