Introduction to Algorithms for Wireless Communication

LU Hardware Software Codesign
WS15/16

Robert Najvirt
TU Wien, Embedded Computing Systems Group

Wien, 16. 10. 2015

The overall goal

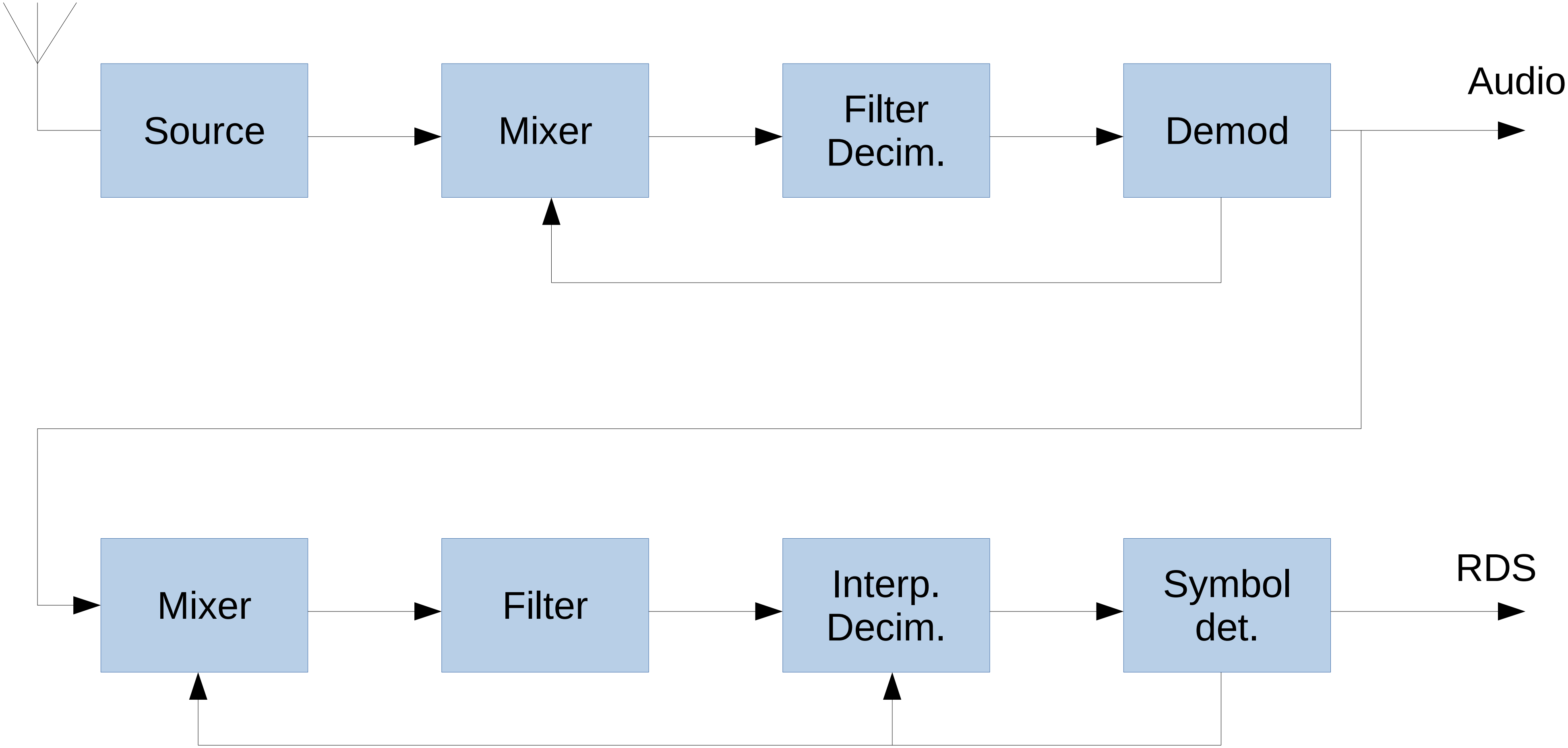
Build an FM receiver:

- Tune to carrier frequency
- Demodulate

Show RDS info

- Tune to subcarrier
- Demodulate
- Decode

System overview



Signal at the input

- IQ samples of baseband signal downmixed from 100.5 MHz
- Sampling rate: 2.5 Msps
- How wide is the spectrum?

  2.5 MHz! (-1.25 MHz to 1.25 MHz)

IQ samples?

- Basically samples as complex numbers
- In-phase (real) and quadrature (imaginary) components
- An RF signal: $A(t) * \cos(2\pi f t + \phi(t))$
  - We need two dimensions for amplitude and phase
- Main advantage:
  - $\sin(x) * \sin(y) = 0.5[\cos(x-y) - \cos(x+y)]$
  - But: $e^x * e^y = e^{x+y}$
- Remember: $e^{i\phi}$ is the unit vector with angle ϕ

IQ sampling

- Generate carrier frequency (100.5 MHz) and a copy of it phase-shifted by $\pi/2$
- Multiply the RF signal by the two, then low-pass-filter and sample separately
- Result is the RF signal frequency-shifted and low-pass-filtered:
  100.5 MHz ± 1.25 MHz
  centered around 0 Hz

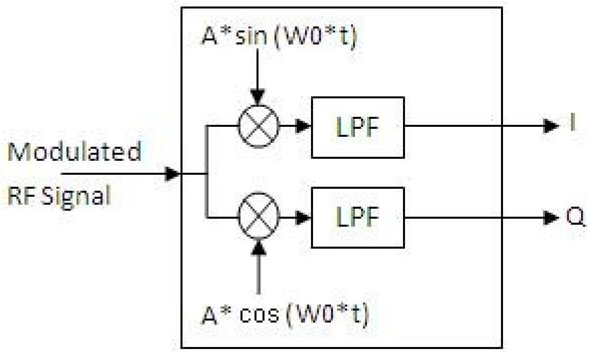
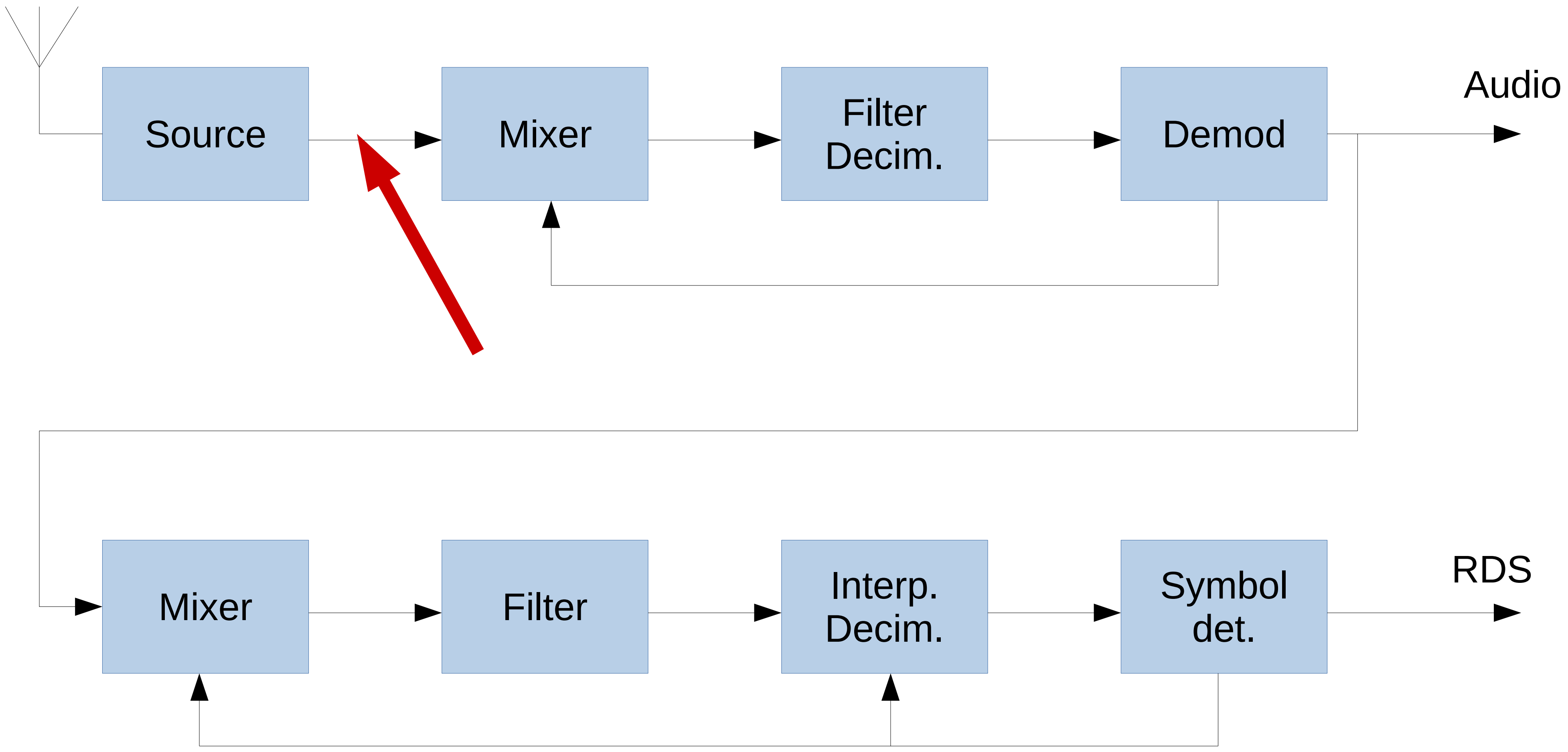


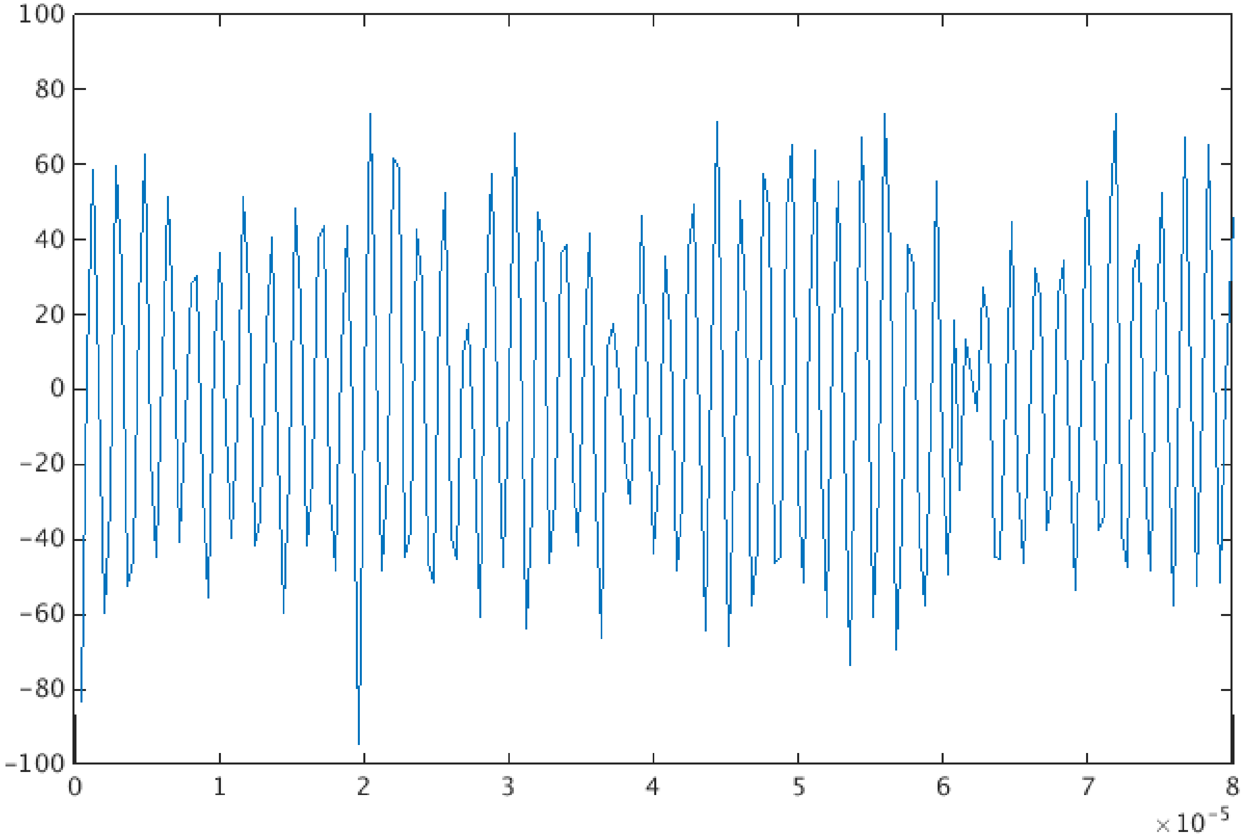
Fig.4 I/Q Demodulator Circuit

[http://www.rfwireless-world.com/Terminology/heterodyne-receiver-versus-homodyne-receiver.html]

Our input signal

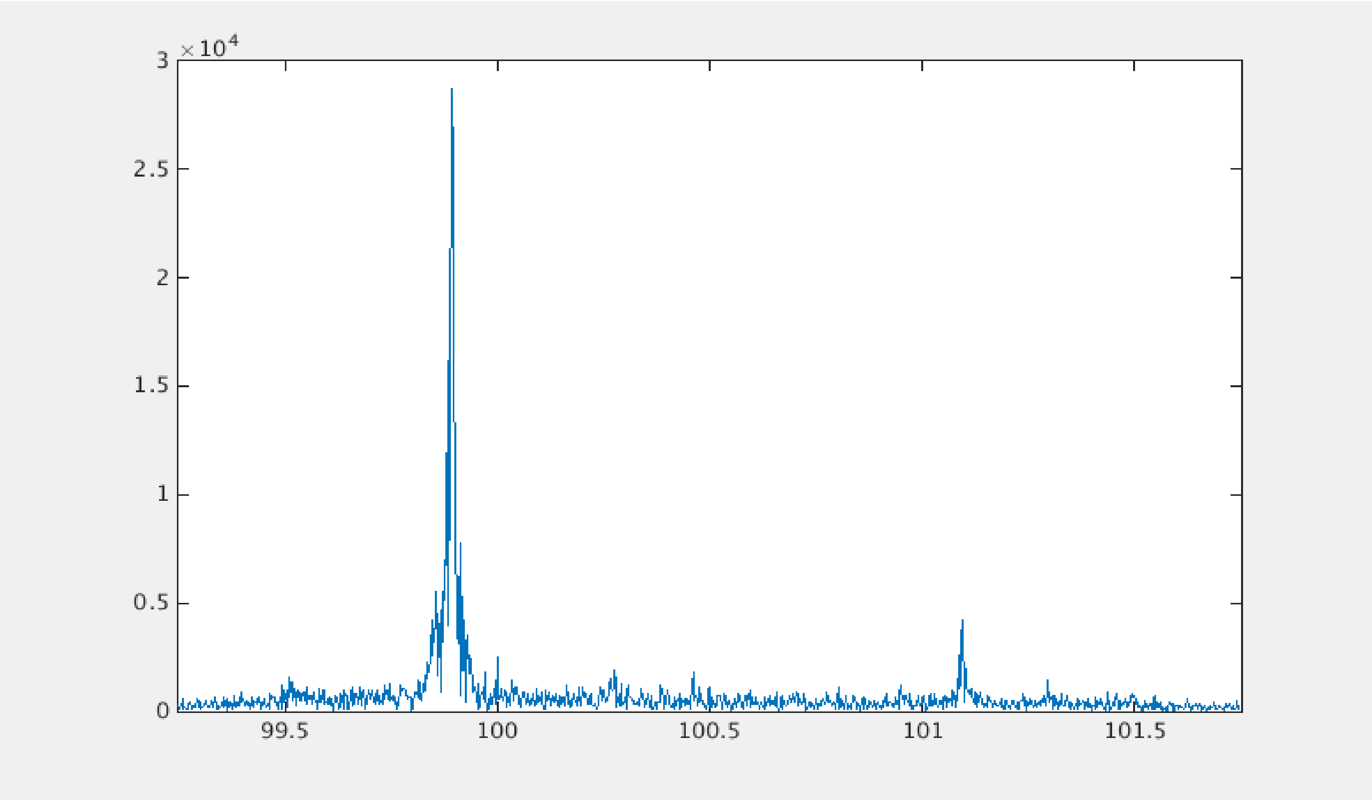


Our input signal



(real part of the signal)

Our input signal



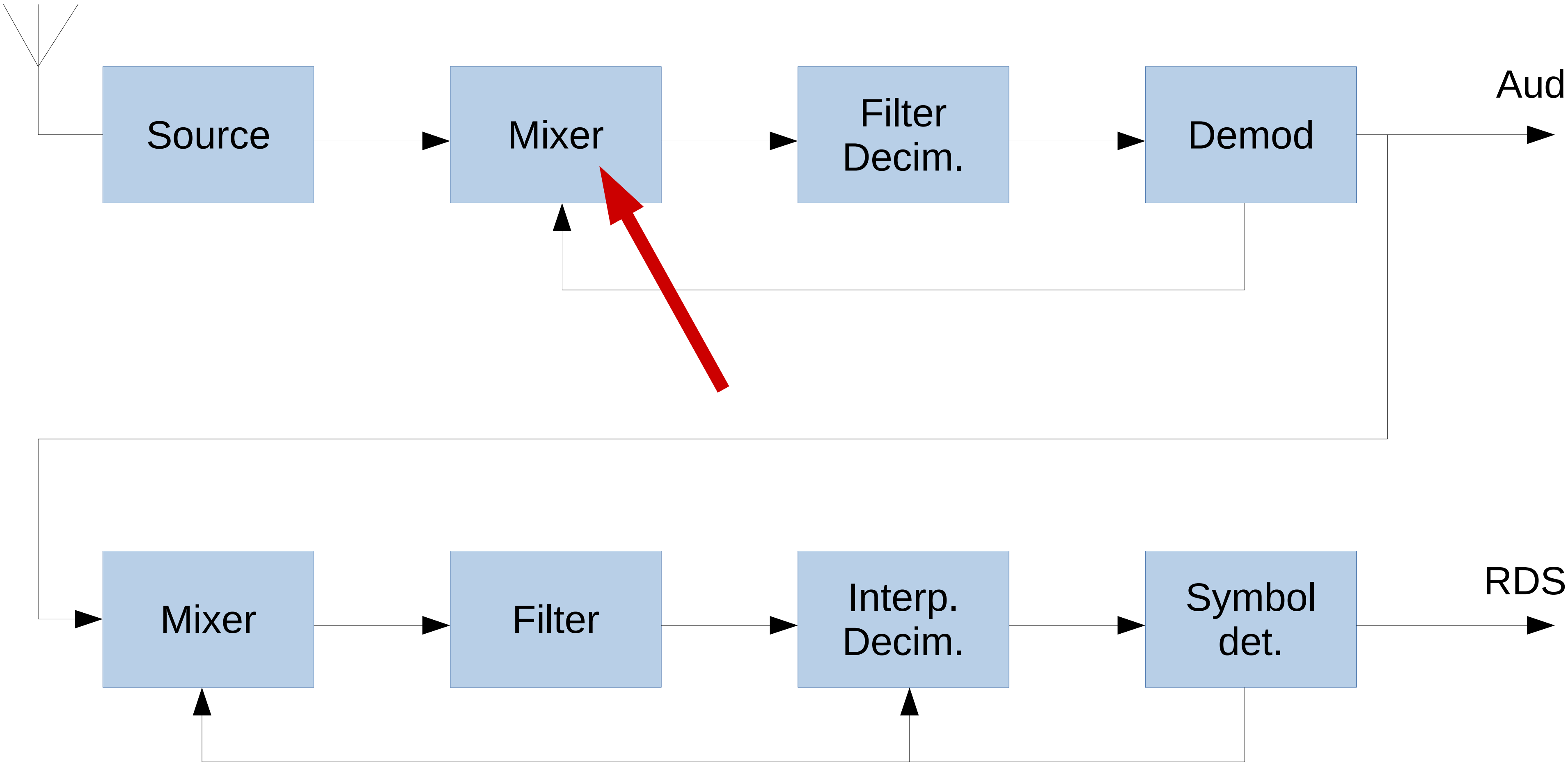
(FFT)

First step

- Tune to the frequency we want:
  99.9 MHz (OE 3)
- Multiplication with the negative carrier frequency:

  $A(t)e^{i2\pi fct+i\theta(t)} * e^{i2\pi(-fc)t} = A(t)e^{i\theta(t)}$
- The component name: Mixer

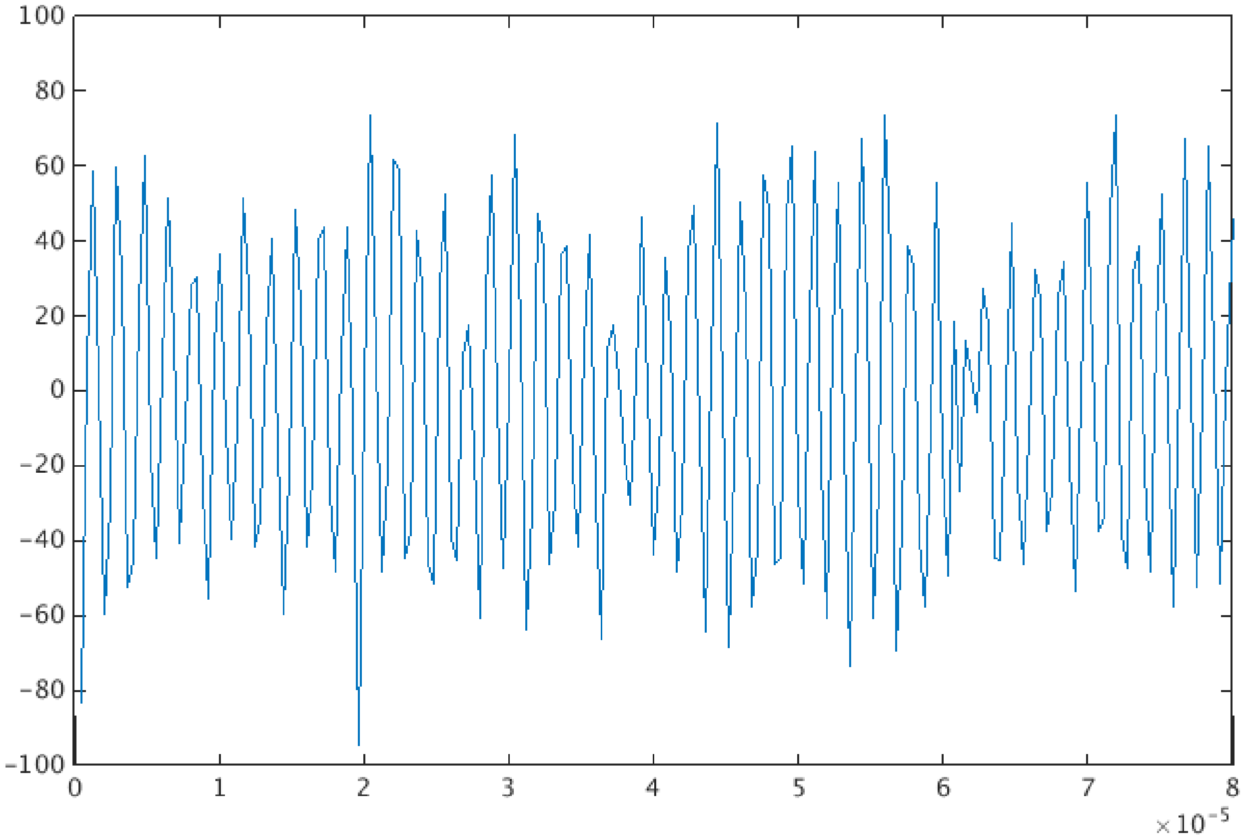
Mixer



Source
Mixer
Filter
Decim.
Demod
Audio
Mixer
Filter
Interp.
Decim.
Symbol
det.
RDS

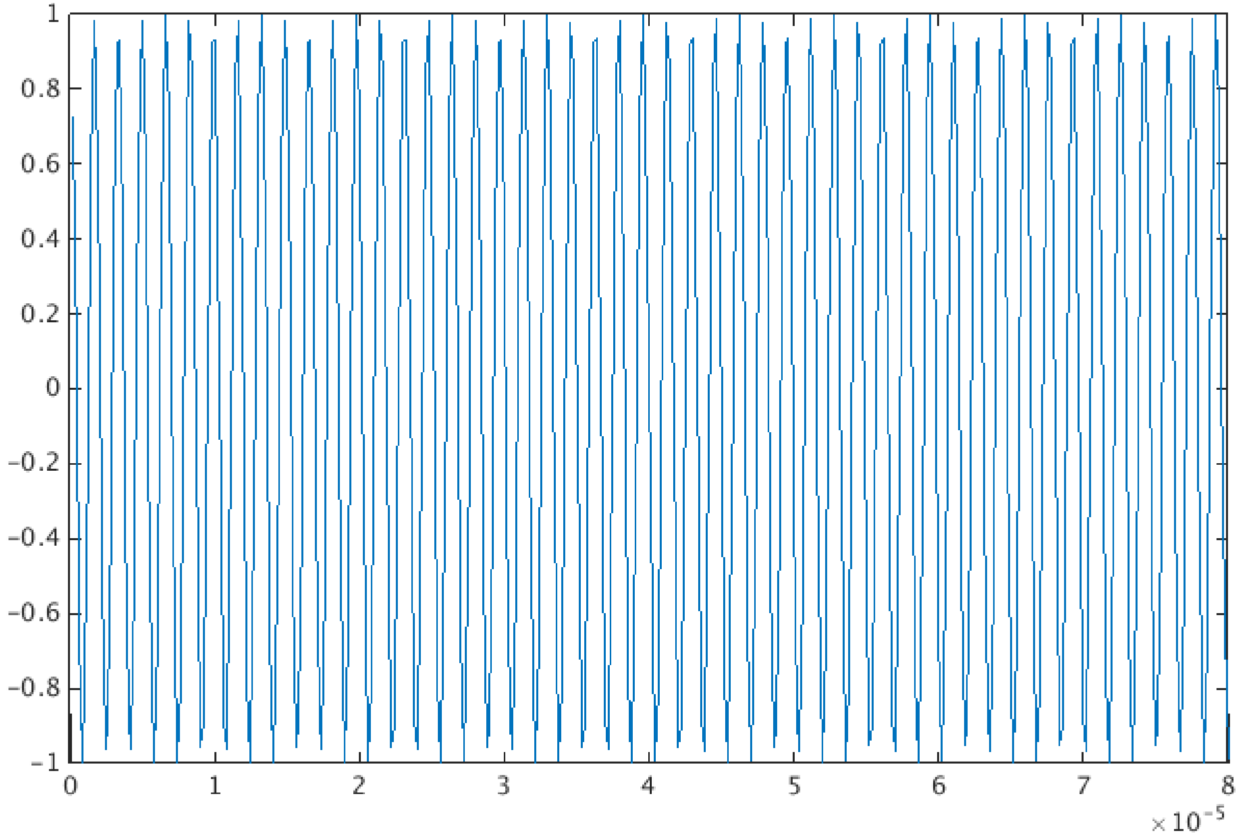
Mixer

Source:



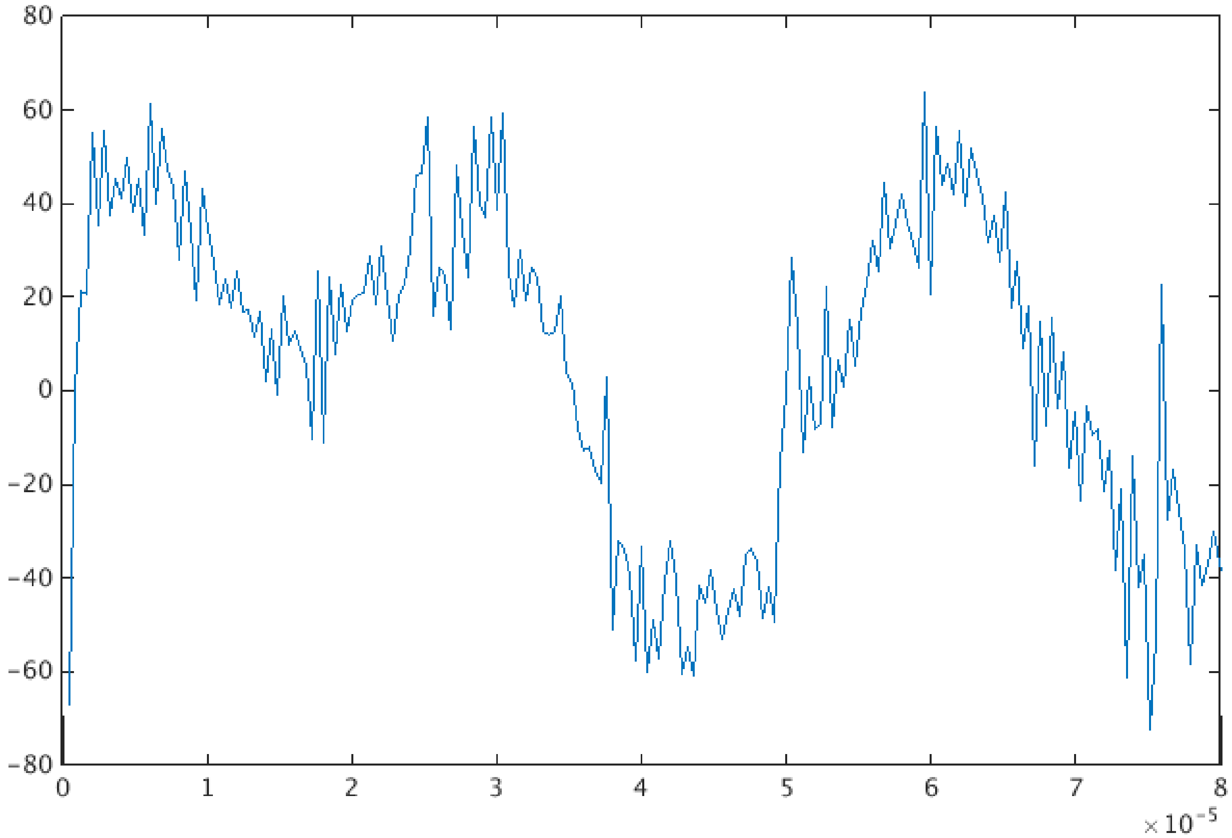
Mixer

Multiplicand (-carrier): (0.6 MHz = -(99.9-100.5))



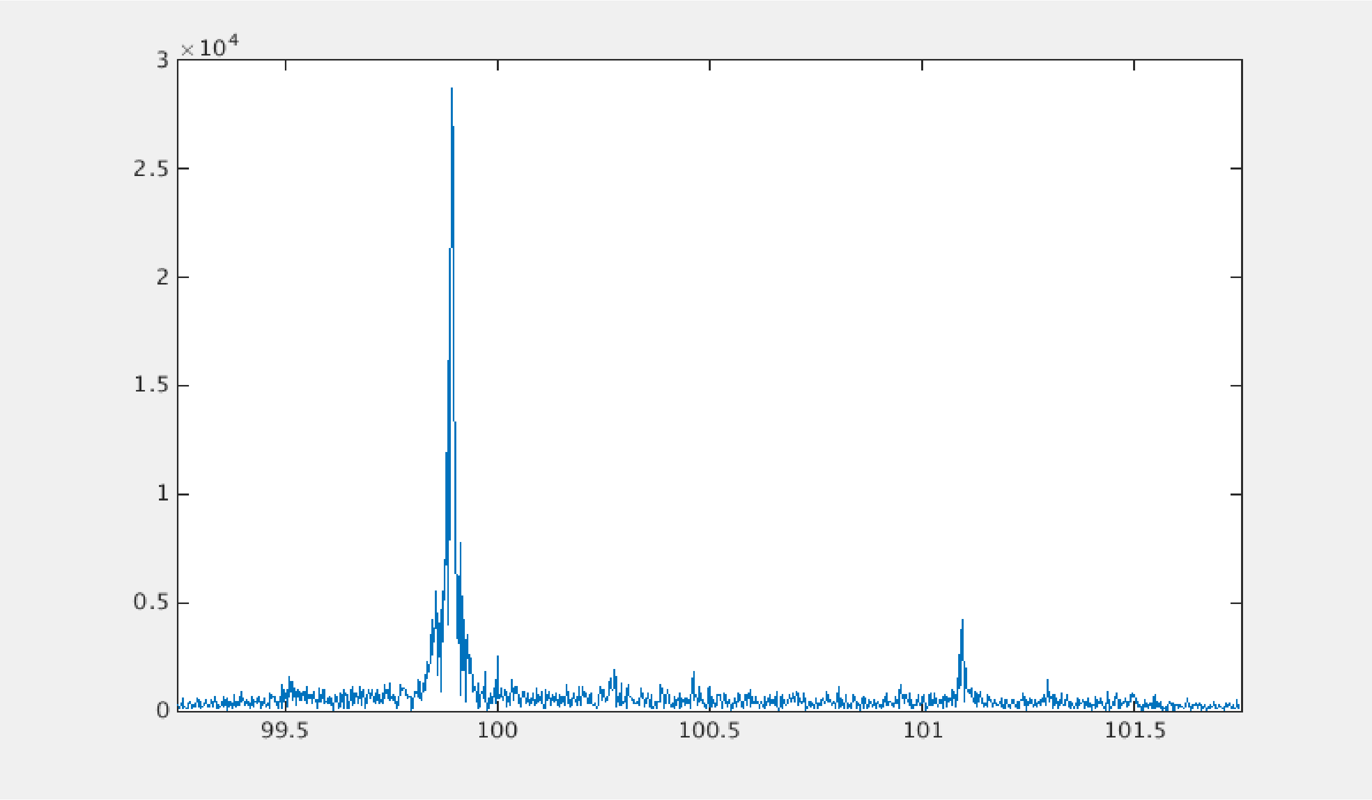
Mixer

Result:



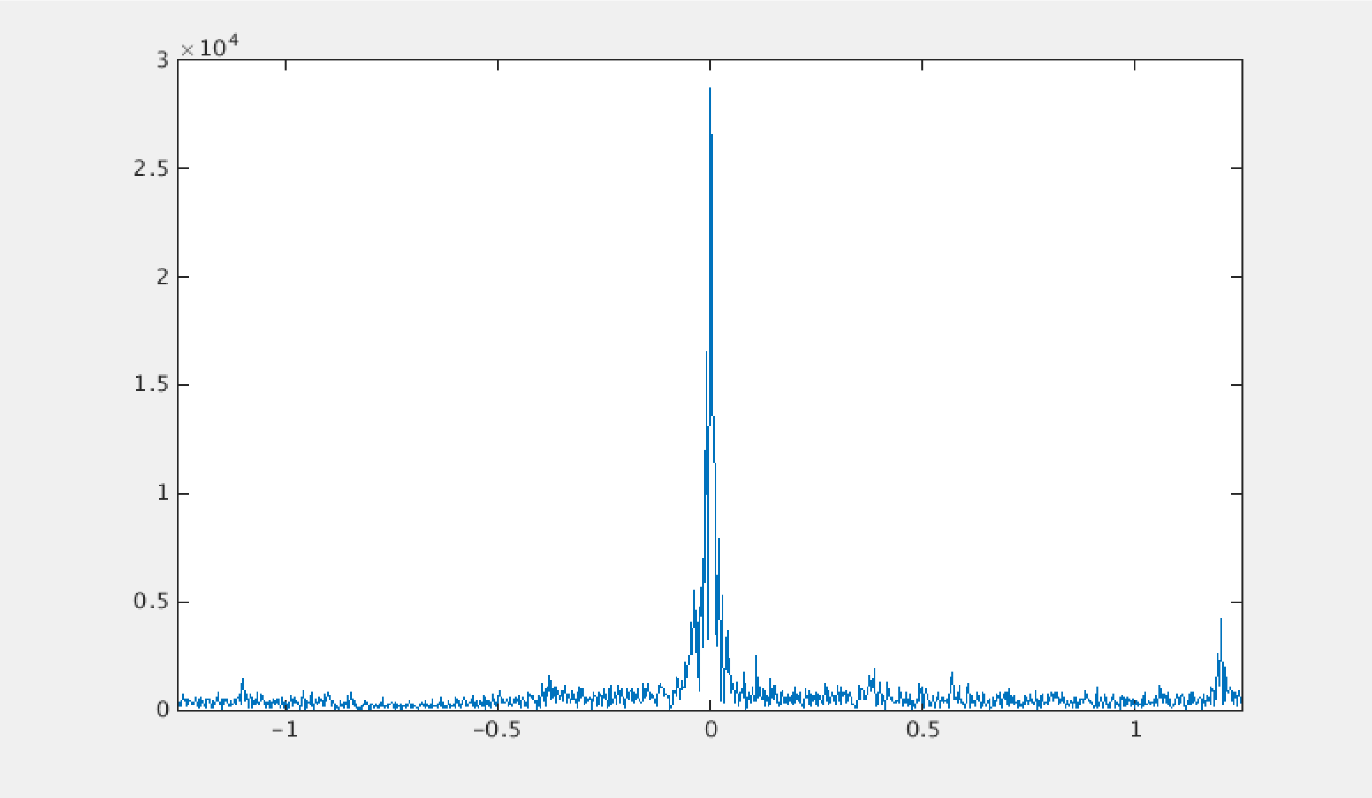
Mixer

FFT before:



Mixer

FFT after:



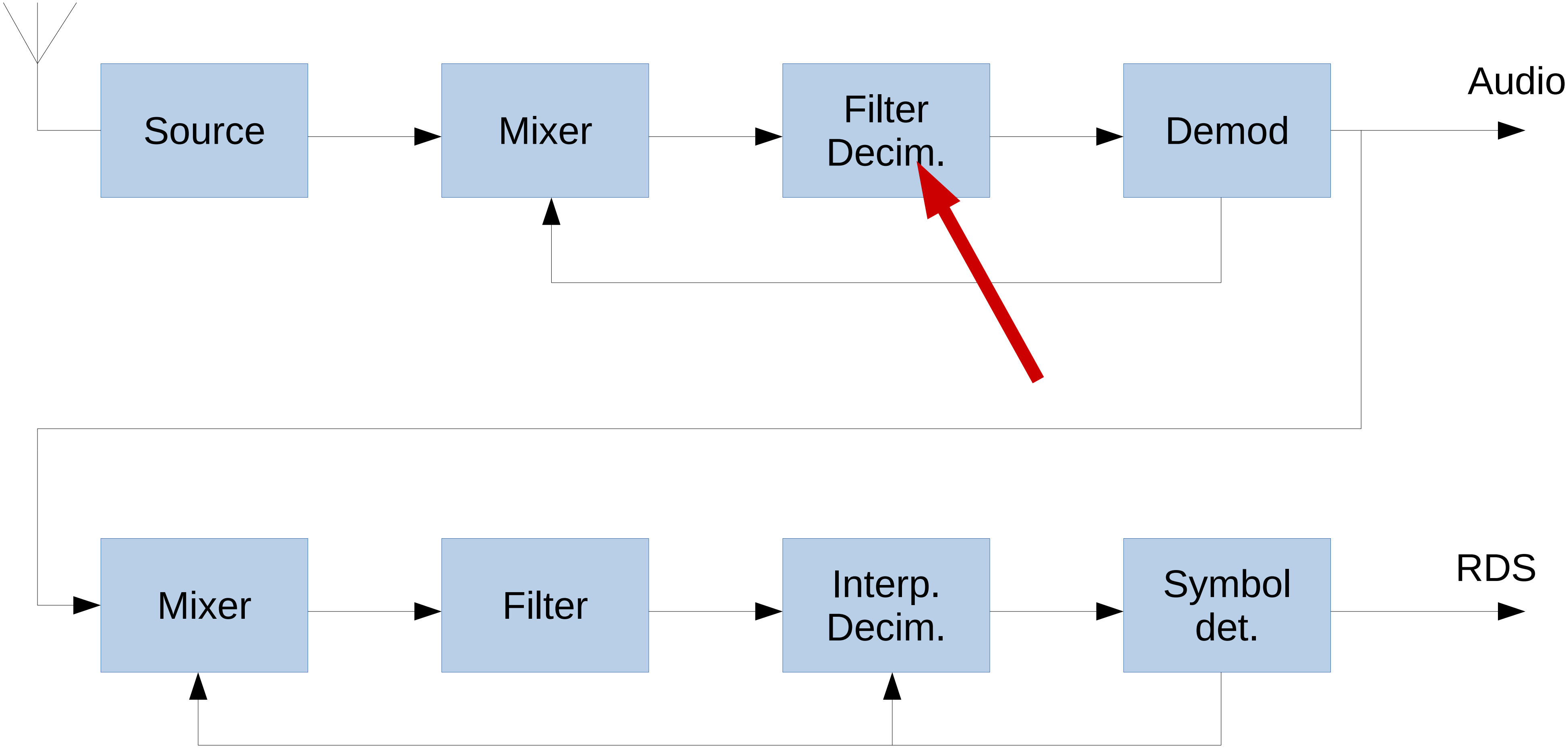
Baseband filter

- Select only relevant signal
- FM bandwidth is max. 200 kHz
  - Low pass filter below 100 kHz
- FIR filter
  - Length is design parameter, affects SNR

Decimation

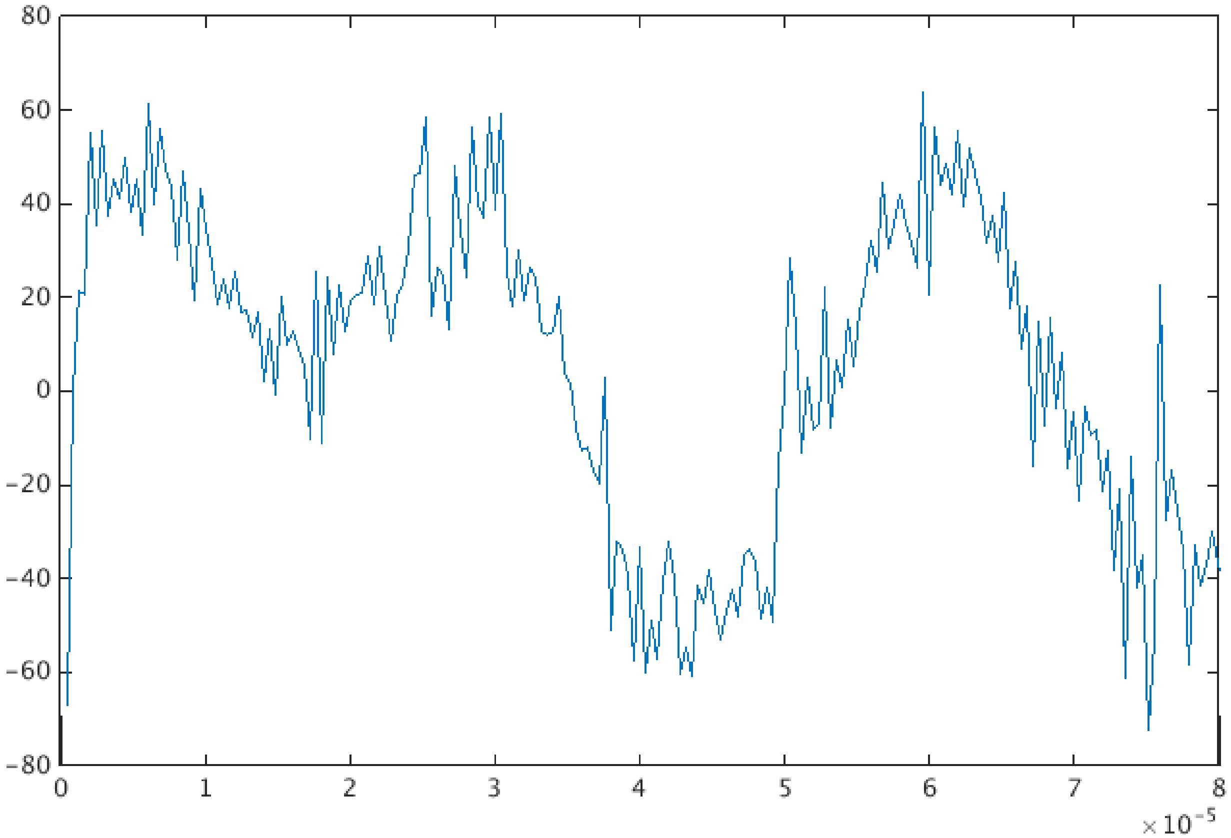
- After filtering, sampling rate unnecessarily high
  - 100 kHz max. frequency, 2.5 Msps
- We can drop every N*th* sample without loss of signal information
  - N is a design parameter, affects algorithmic complexity (demodulation, filters) and computation speed later

Filtering and decimation



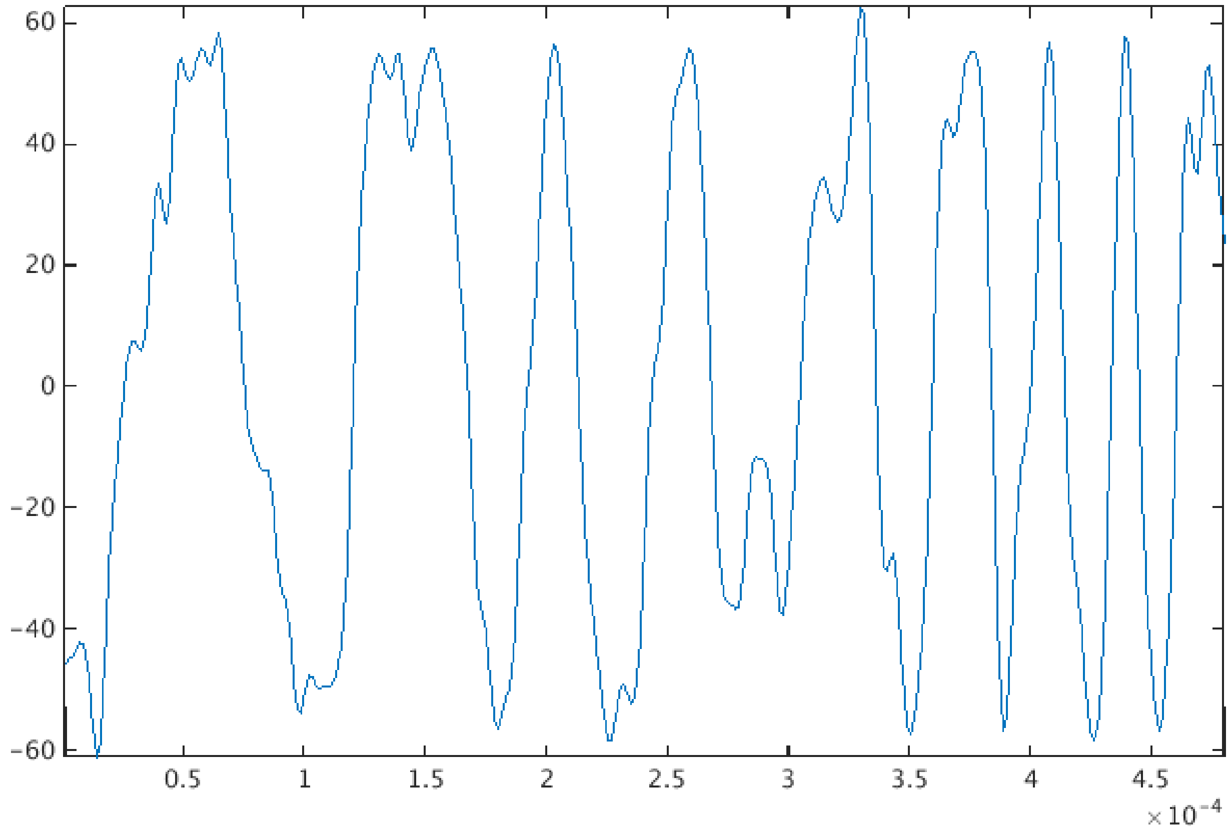
Filtering and decimation

Before:



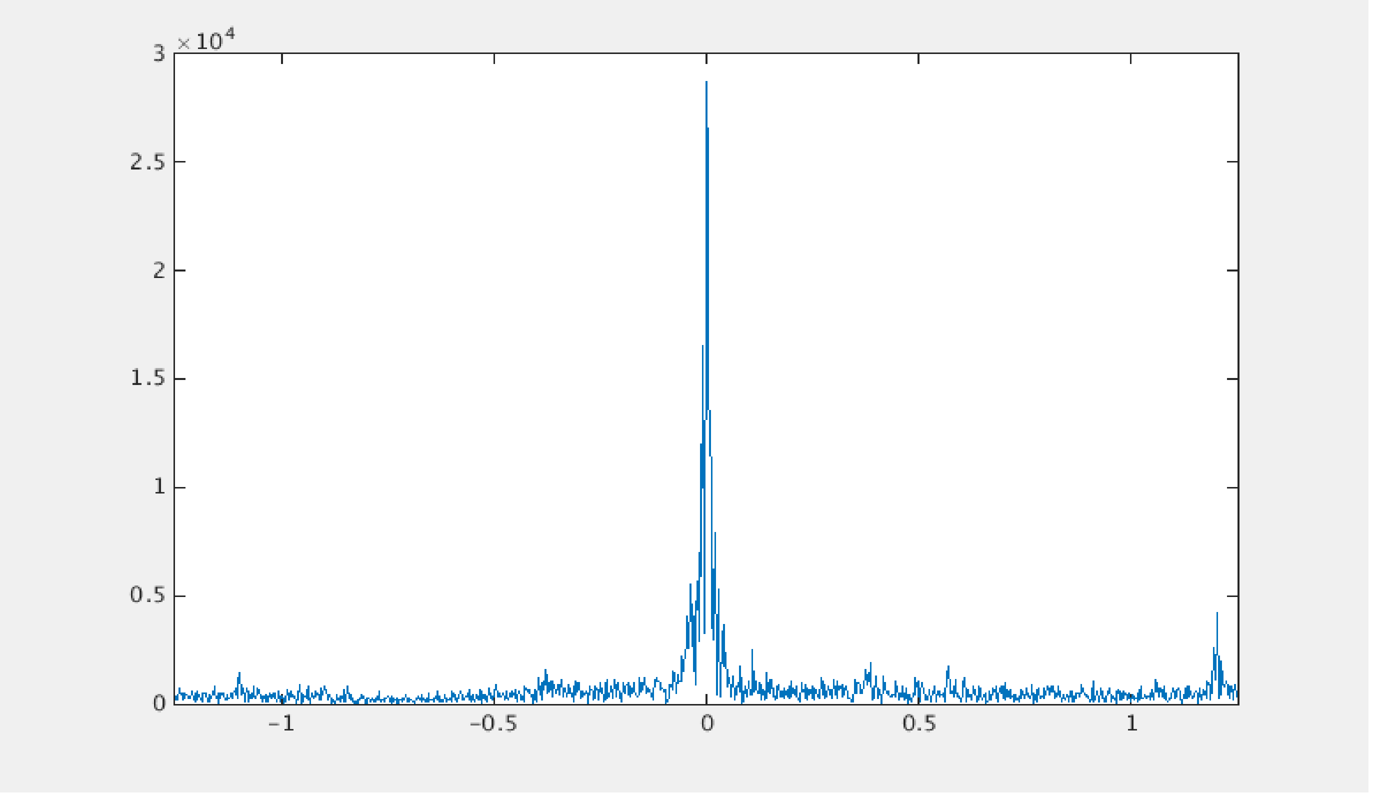
Filtering and decimation

After:



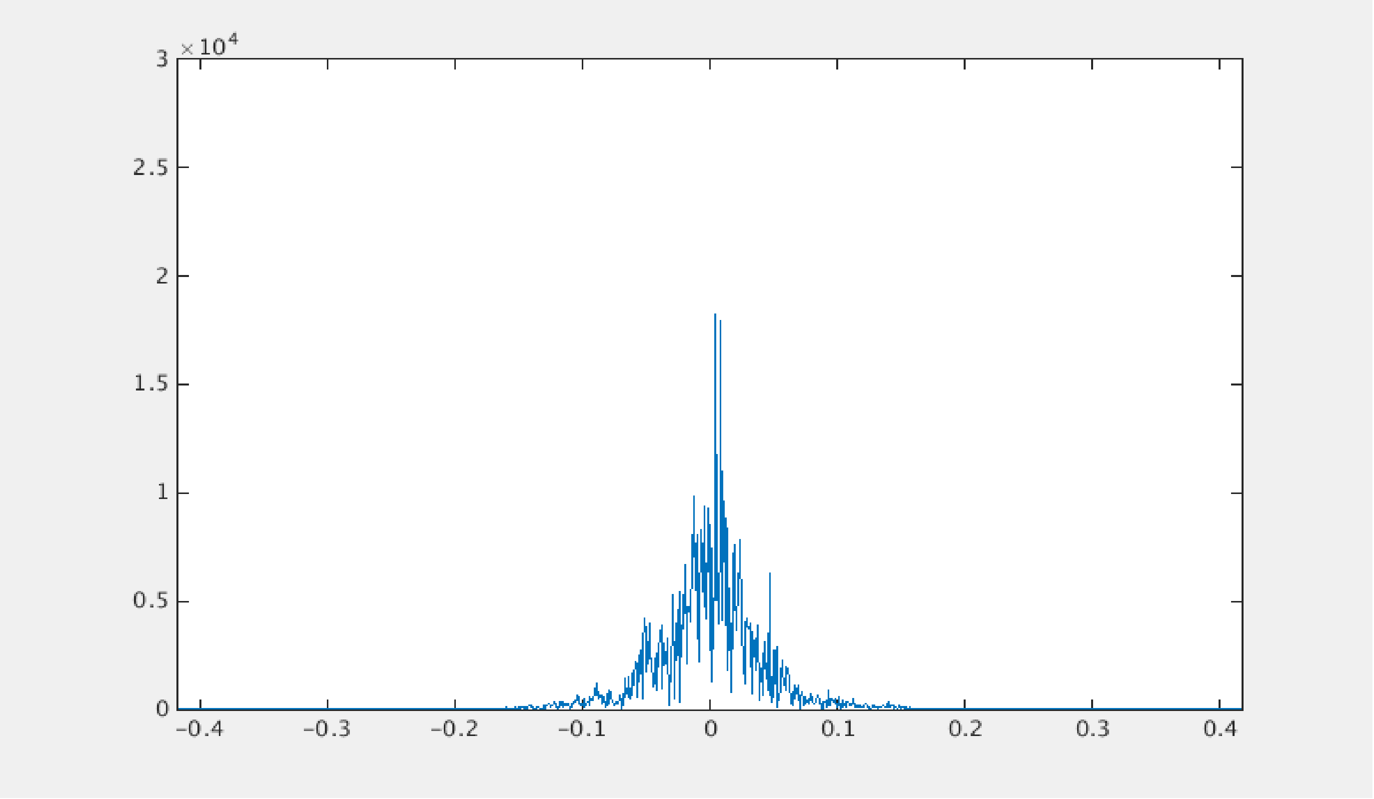
Filtering and decimation

FFT before:



Filtering and decimation

FFT after:

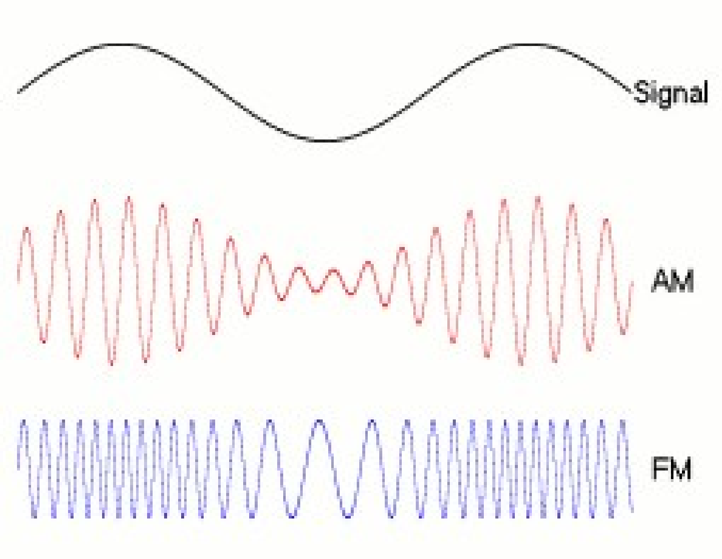


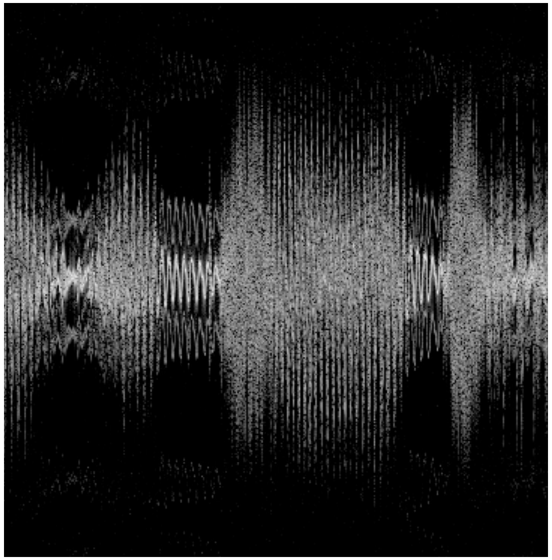
FM Modulation

- The amplitude of the carrier is left unchanged but the frequency is altered:

  $rf(t) = A(t)e^{i2\pi f_c t + i m(t) t}$

  – A(t) constant, frequency deviation m(t) = message





["Amfm3-en-de" by Berserkerus - Own work. Licensed under CC BY-SA 2.5 via Commons - https://commons.wikimedia.org/wiki/File:Amfm3-en-de.gif#/media/File:Amfm3-en-de.gif]

[http://web.stanford.edu/class/ee179/labs/Lab5.html]

FM modulation

- We already removed the carrier:

  $e^{i2\pi f_c t+i m(t) t} * e^{-i2\pi f_c t} = e^{i m(t) t}$

- Frequency ~~deviation~~ = message
- Frequency is phase difference
  - With complex signals phase = angle

FM demodulation #1

- Detect angle difference
- One approximation (has a problem):

    angle(sig(t)) – angle(sig(t-1/fs))

- Simpler:

    angle(sig(t)*conj(sig(t-1/fs)))

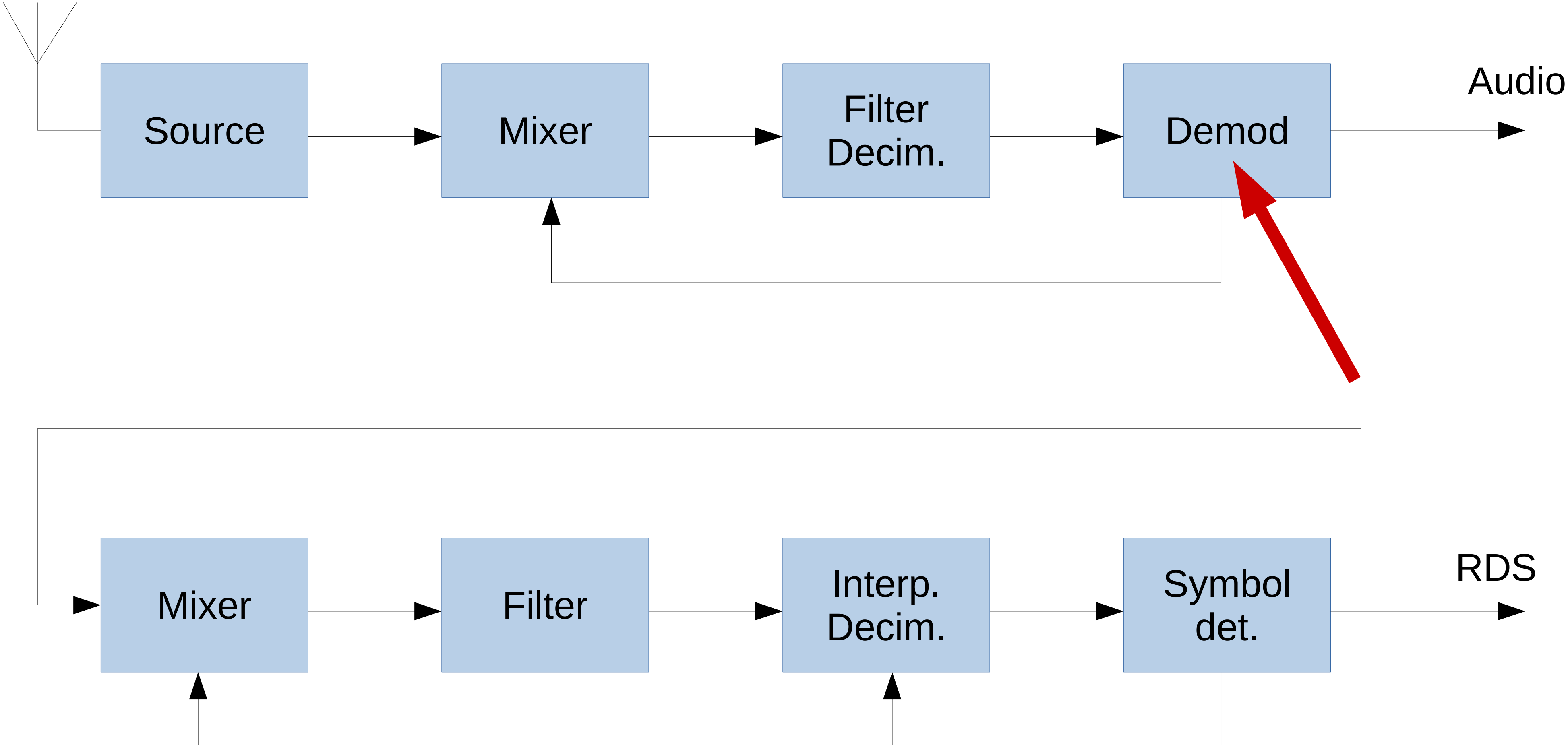
FM demodulation #2

- Differentiate signal and multiply by conjugate of original

  $(d/dt\ e^{im(t)t}) * conj(e^{im(t)t}) =$
  $im(t) * e^{im(t)t} * e^{-im(t)t} =$
  $im(t)$

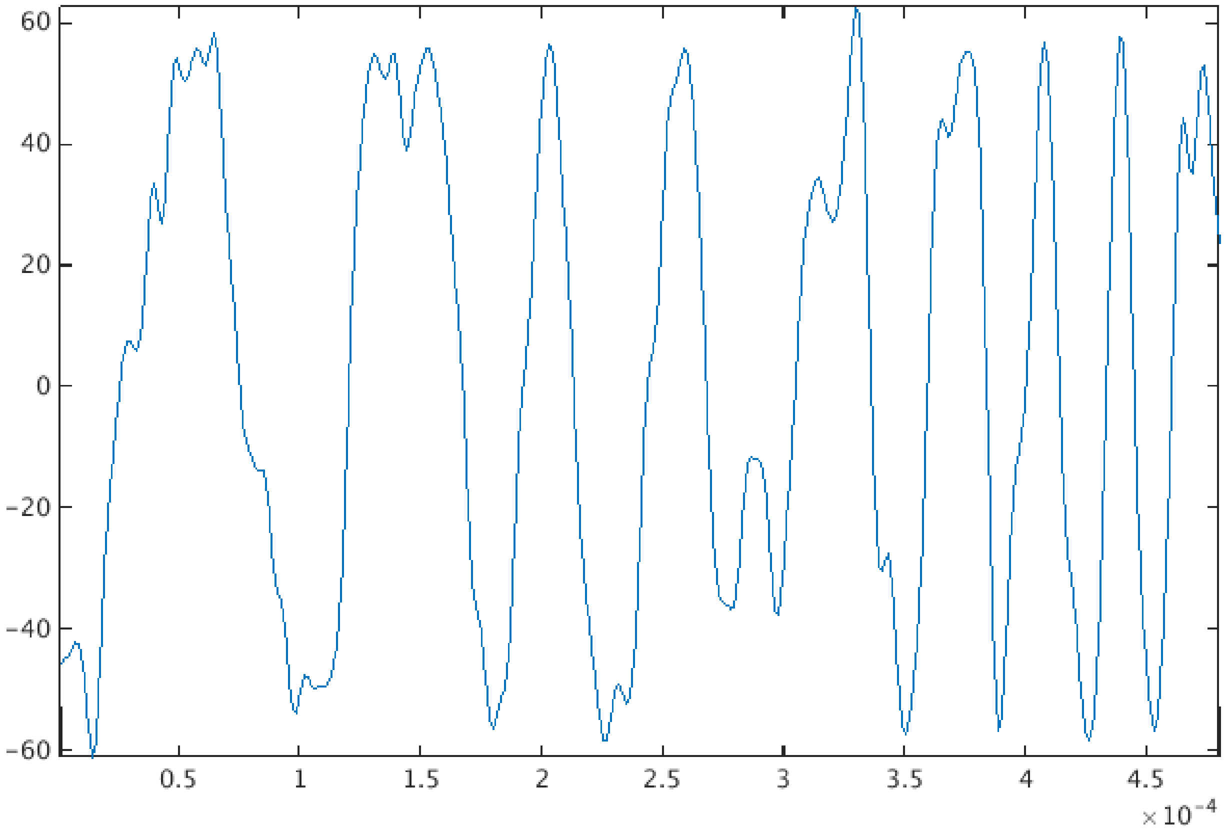
- Differentiation by first order approximation (sig(t)-sig(t-1/fs)) or better with FIR filter
  - See recommended reading http://web.stanford.edu/class/ee179/labs/Lab5.html

Filtering and decimation



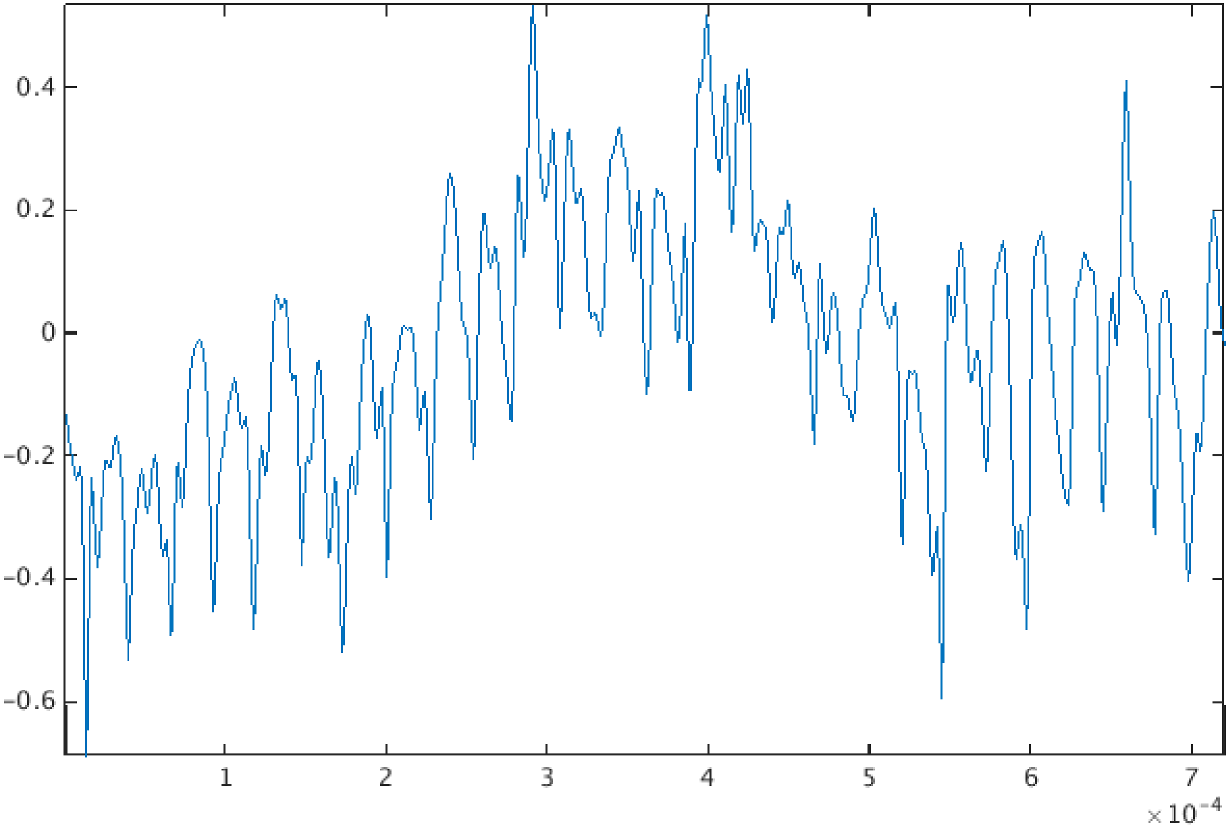
FM demodulation

Before:



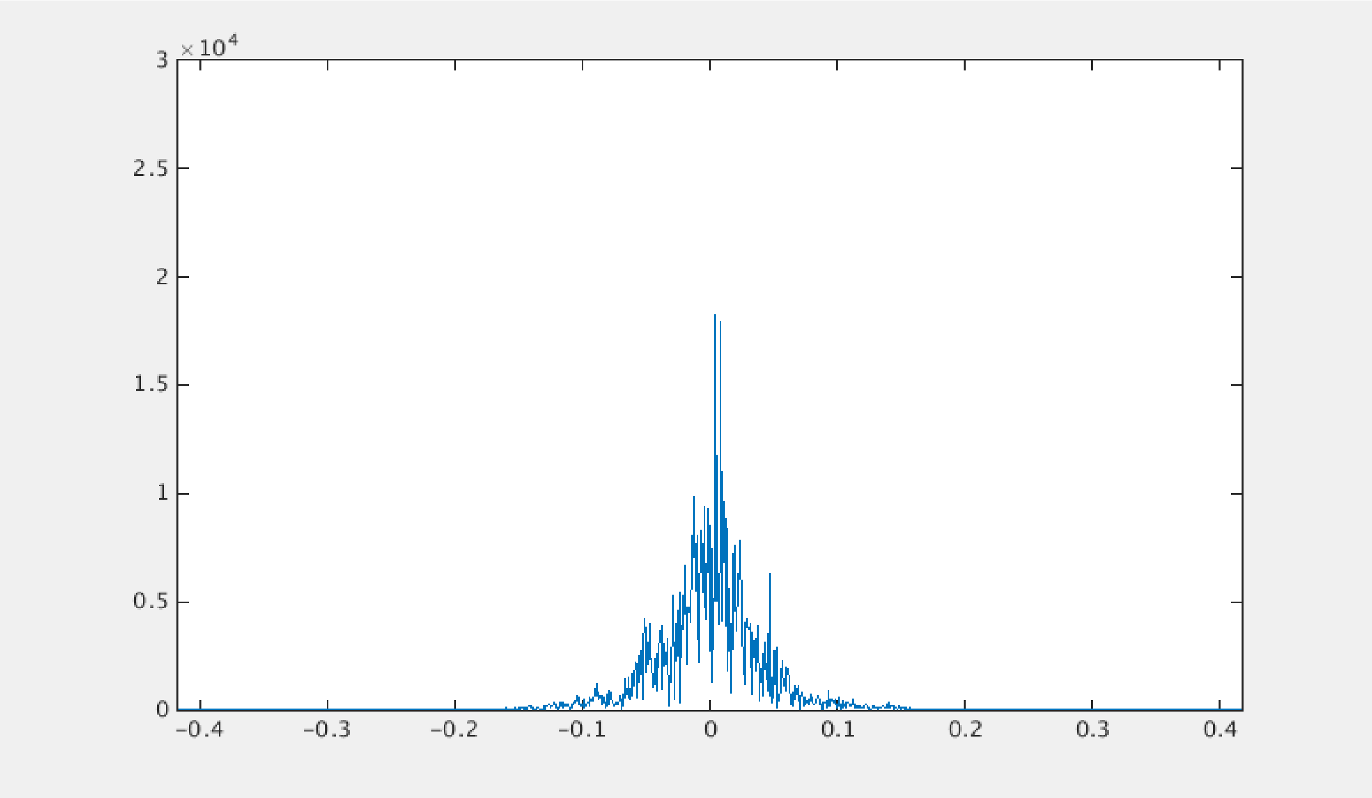
FM demodulation

After (real signal):



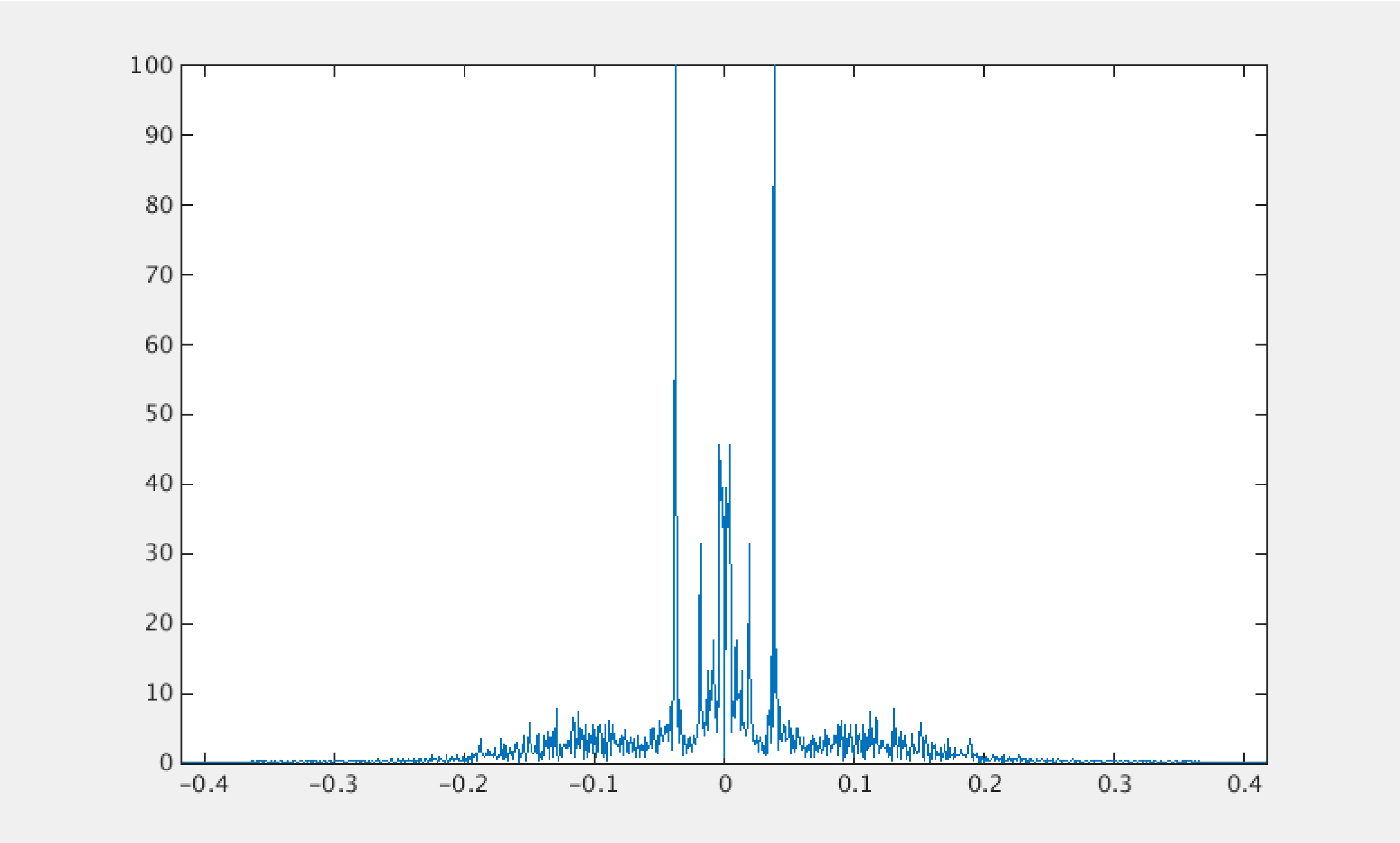
FM demodulation

FFT before:



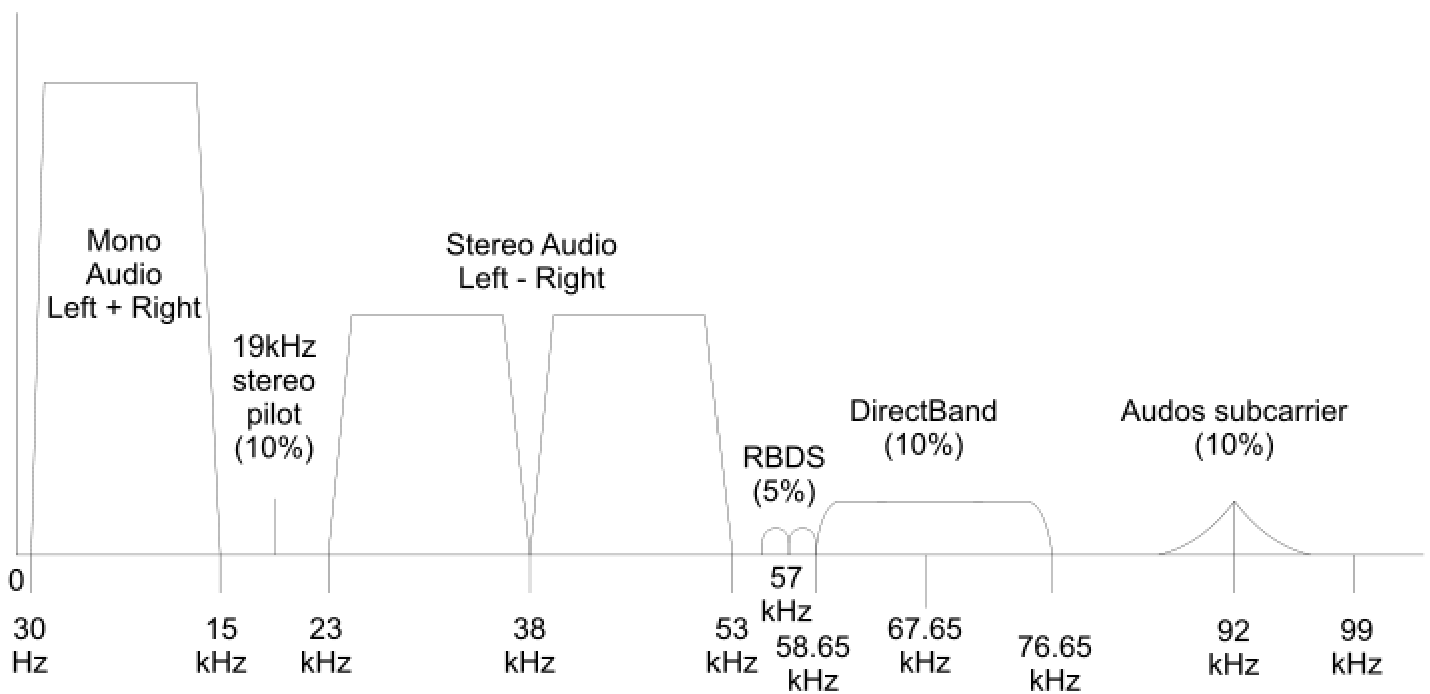
FM demodulation

FFT after:



FM demodulated signal

- Audio output: Low pass below 19 (15) kHz, (resample and) ouput
- RDS: (Again) modulated at 57 ± 2.4 kHz



[Wikipedia]

Carrier frequency error

- When mixing the (estimated) carrier, an error term remains:

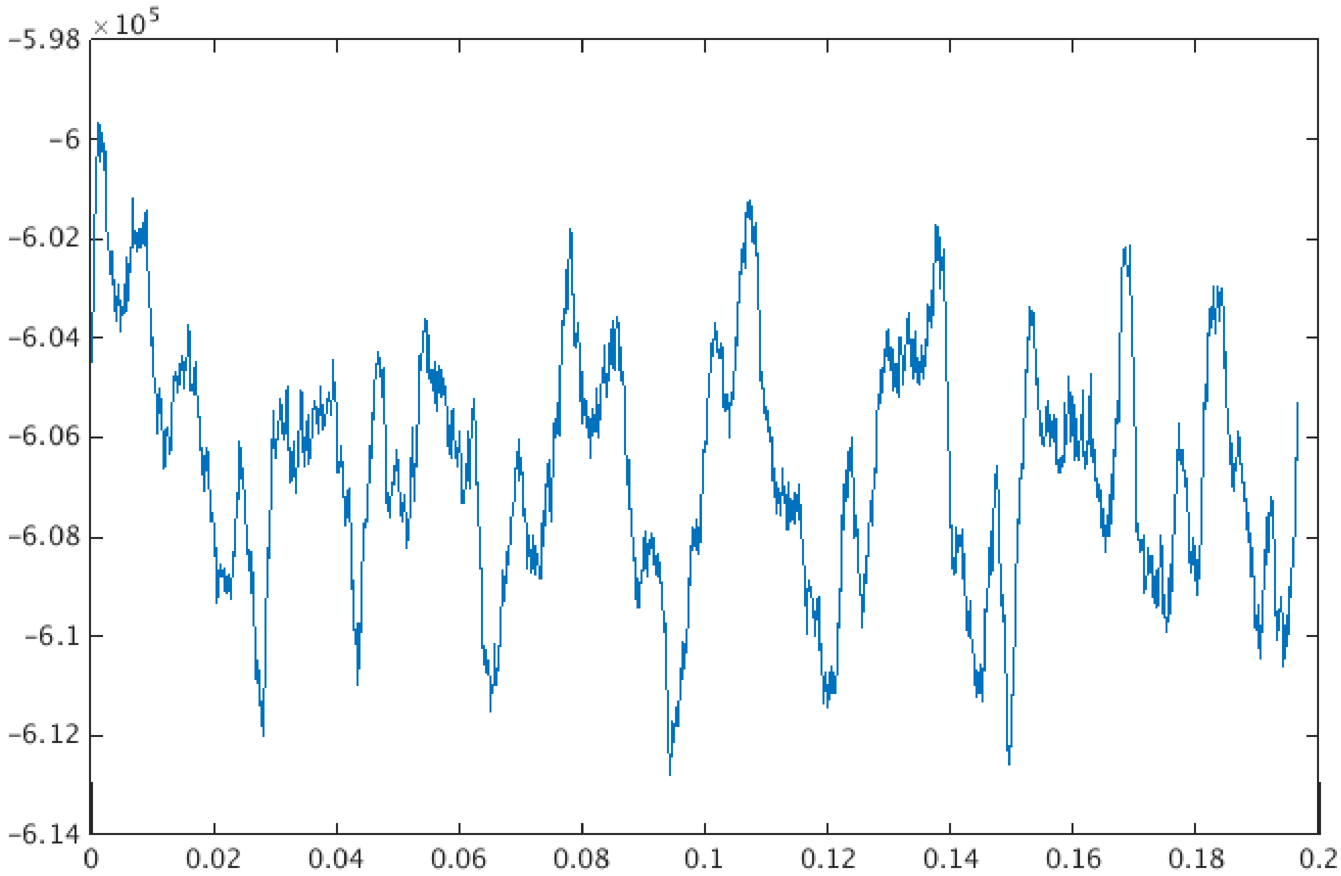
  $e^{i2\pi fc t + i\theta(t)t} * e^{i2\pi(-fc+err)t} = e^{-i\theta(t)t} * e^{i2\pi err t}$

  - A remaining frequency component
- After FM demodulation, this term adds to the message signal
- Assuming it is almost constant, we can correct fc using the low-passed message signal
- Error comes mainly from sampling, not from broadcasting

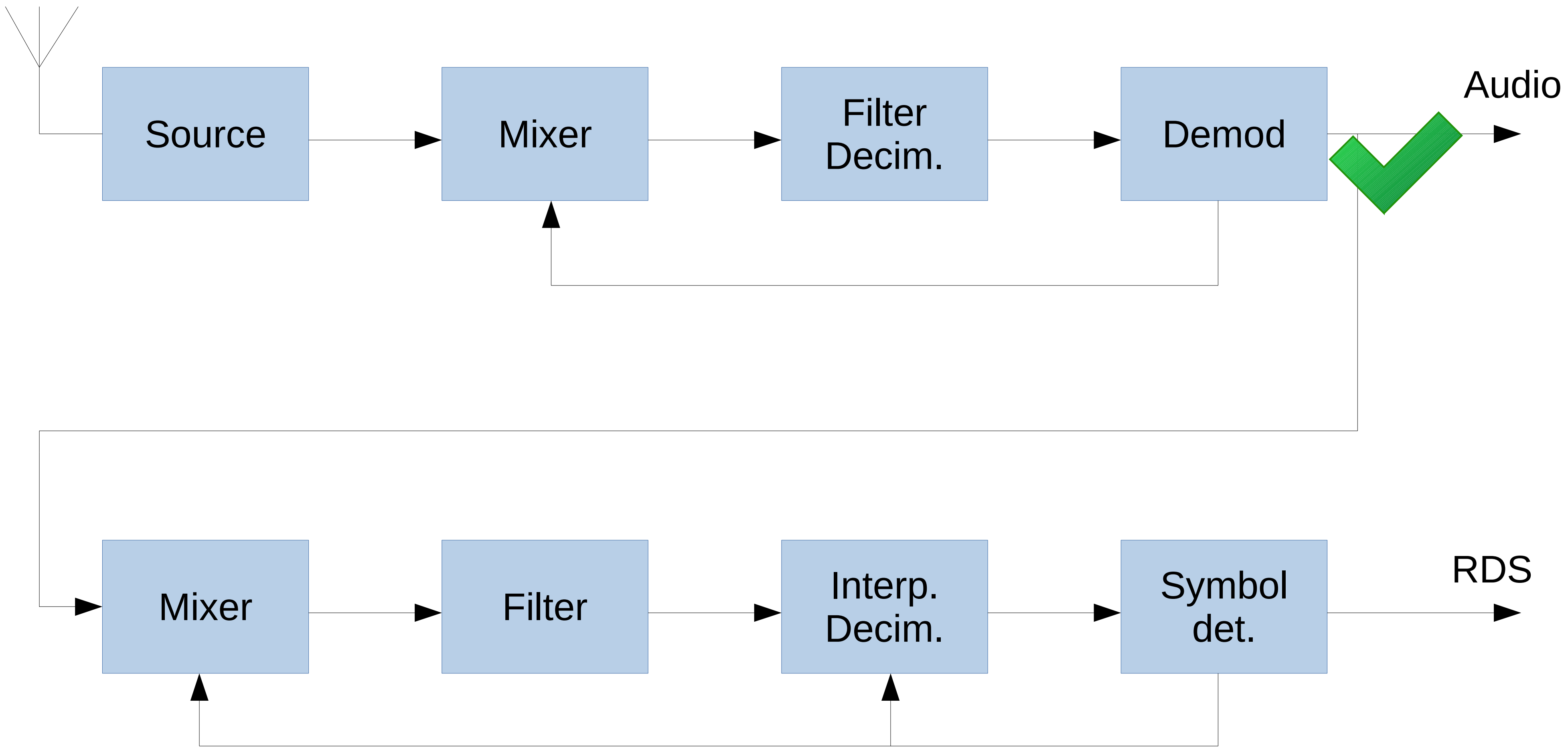
Carrier frequency error

- FM audio starts at 30 Hz – use a low pass filter (well) below that; an IIR filter may be used

fc vs. time



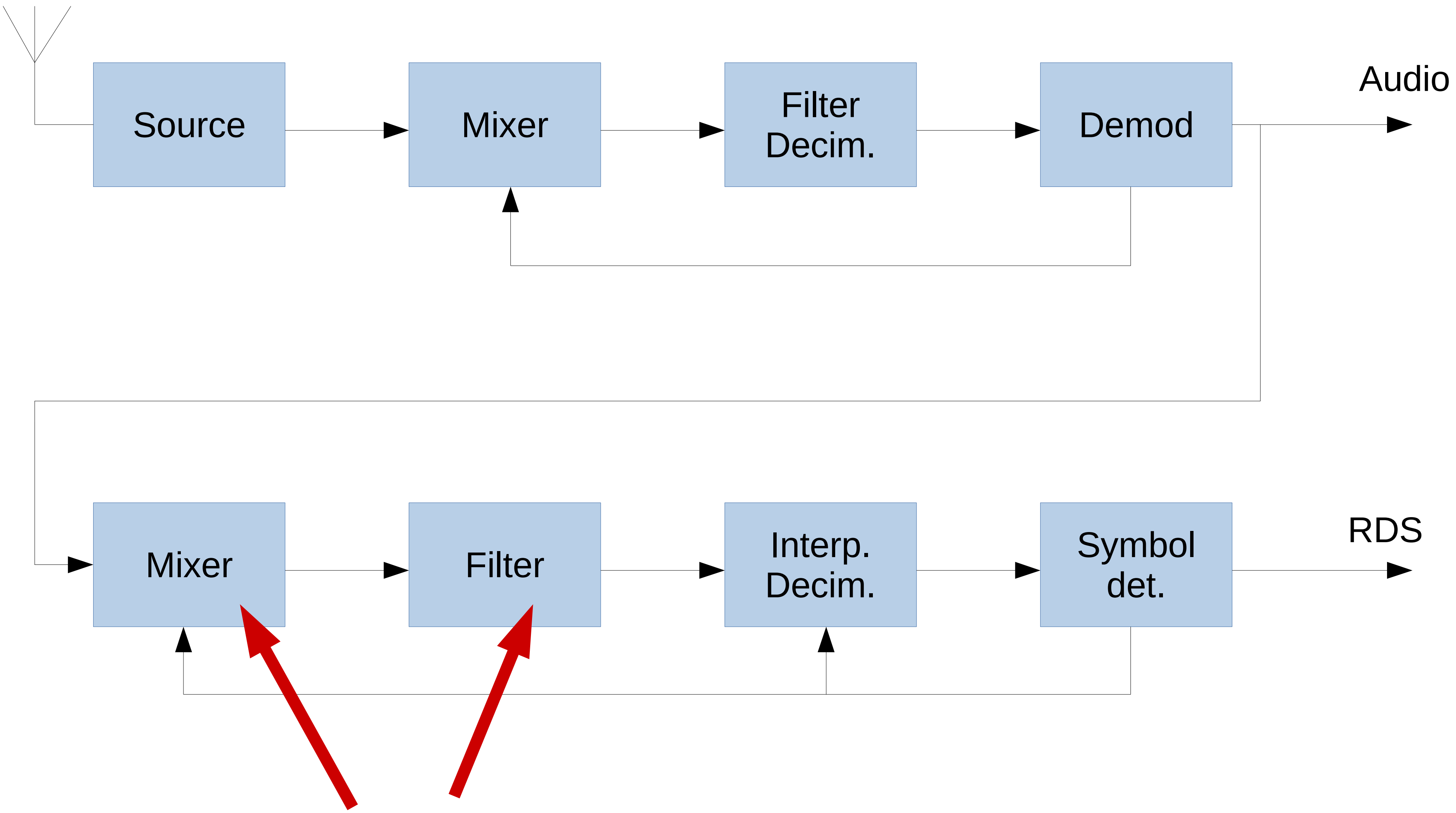
2nd Part


Source
Mixer
Filter
Decim.
Demod
Audio
Mixer
Filter
Interp.
Decim.
Symbol
det.
RDS

RDS subband

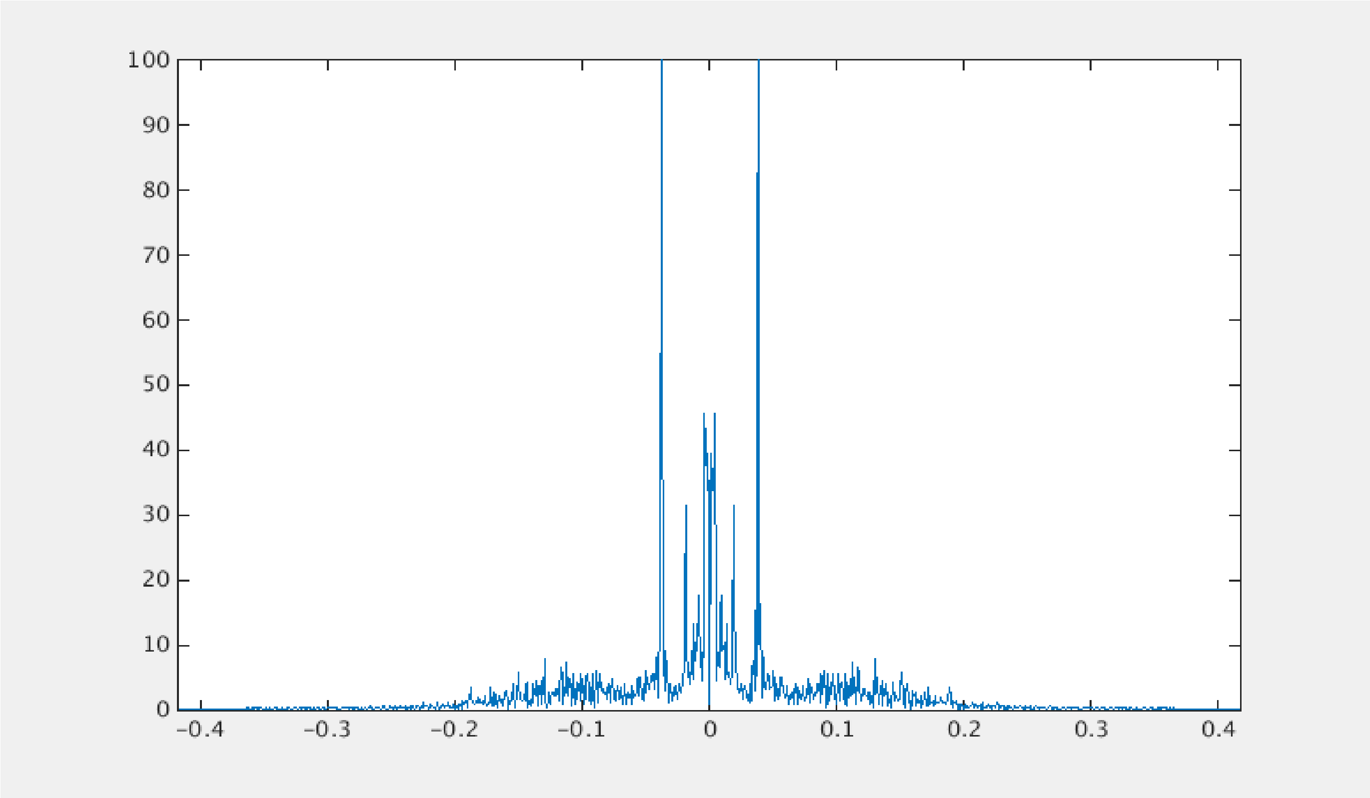
- Subcarrier is 57 kHz → start with mixer
  - We know how to do that already
- Next step is low pass (matched) filtering
  - RDS standard specifies the filter parameters
- The idea behind matched filtering
  - low-pass@TX and low-pass@RX <u>together</u> give a filter beneficial for signal quality

Filtering and decimation



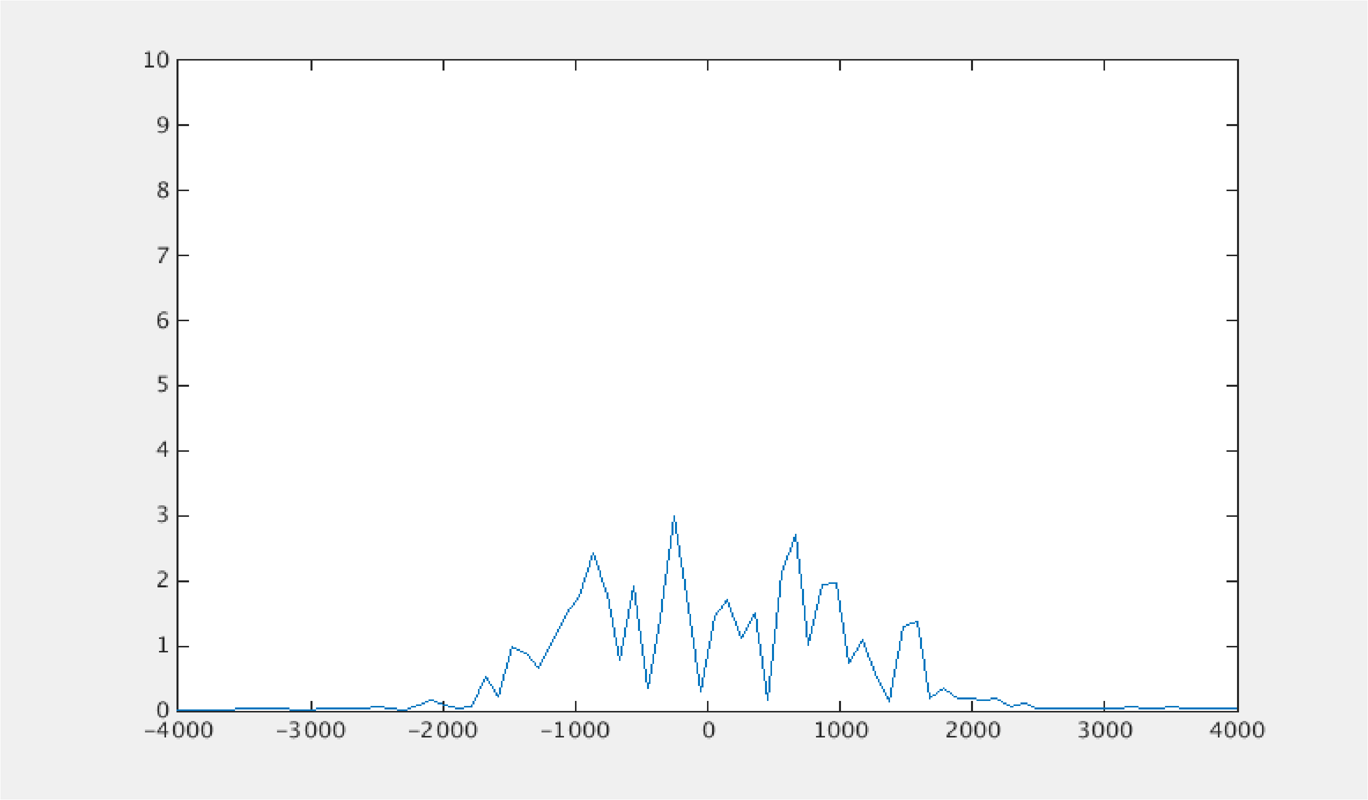
RDS mixing and filtering

FFT input:



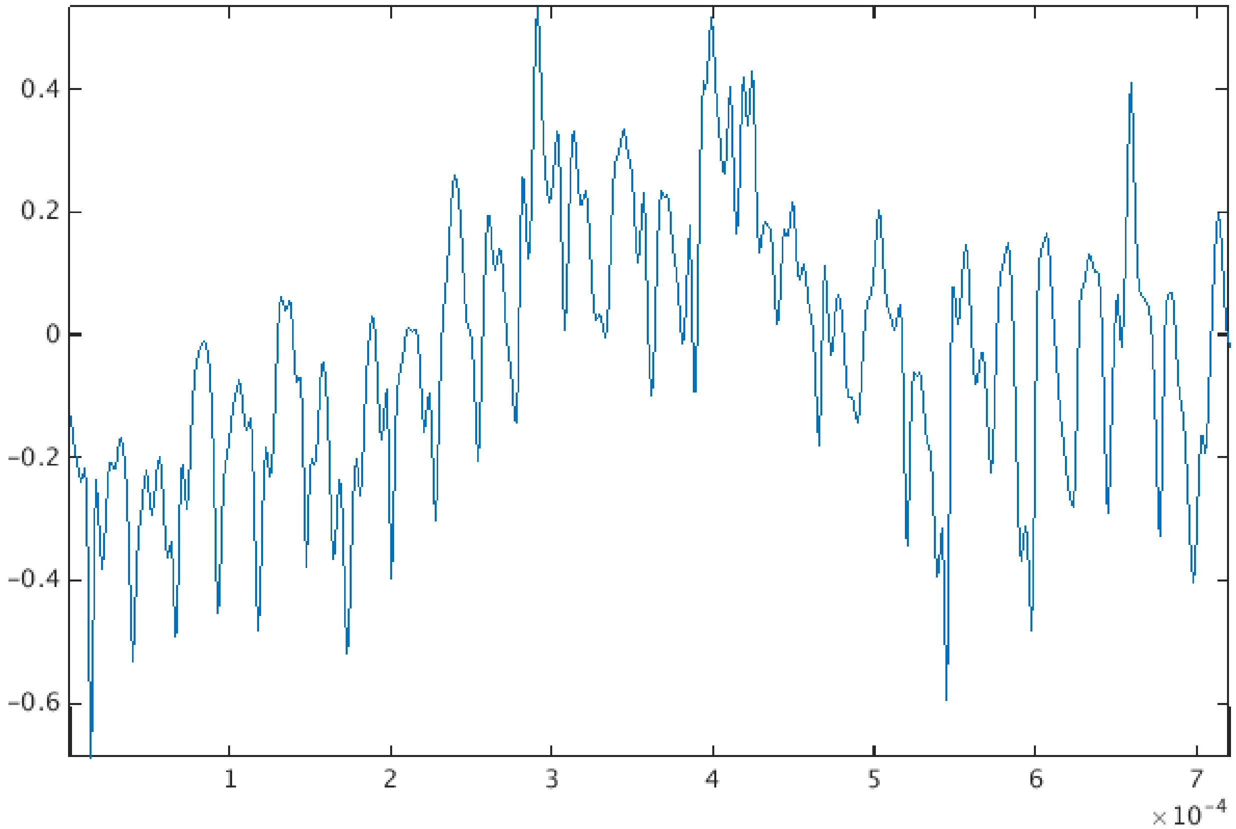
RDS mixing and filtering

FFT after mixing and filtering:



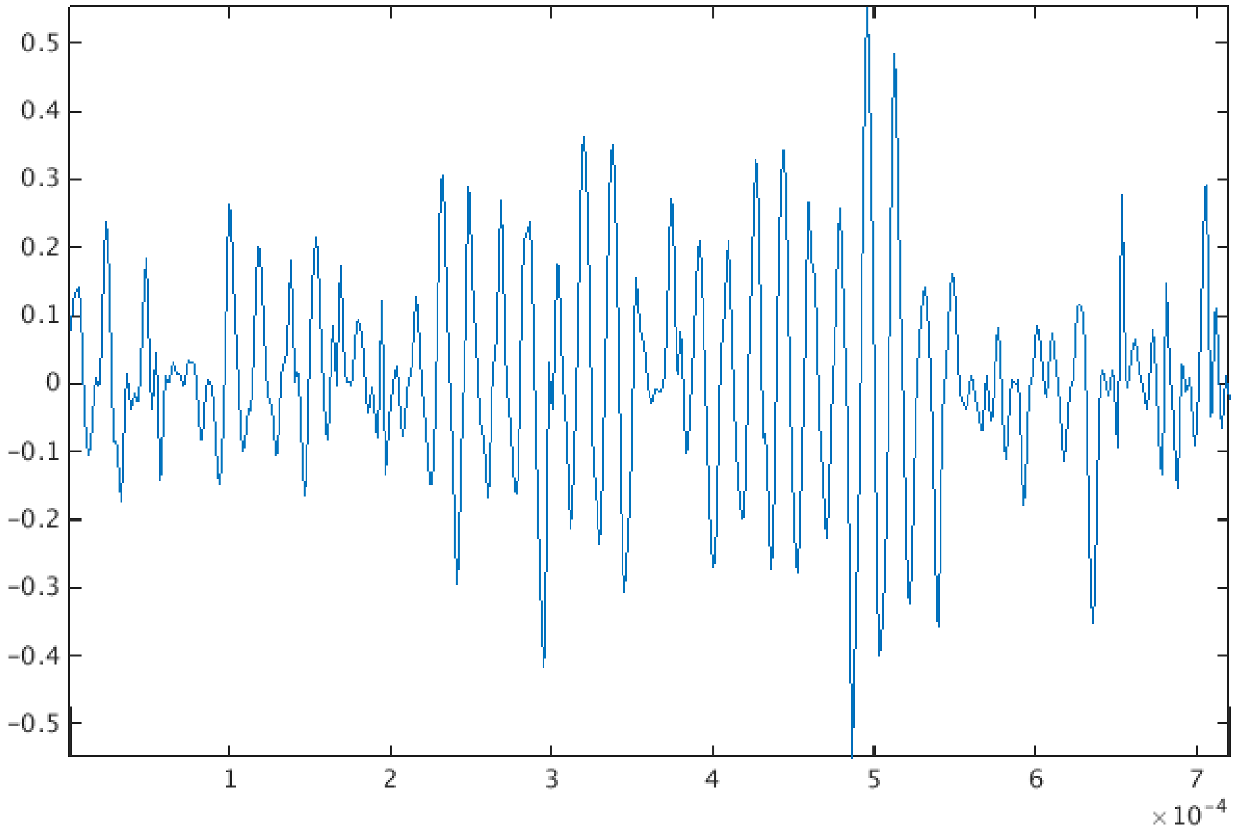
RDS mixing and filtering

Signal at input:



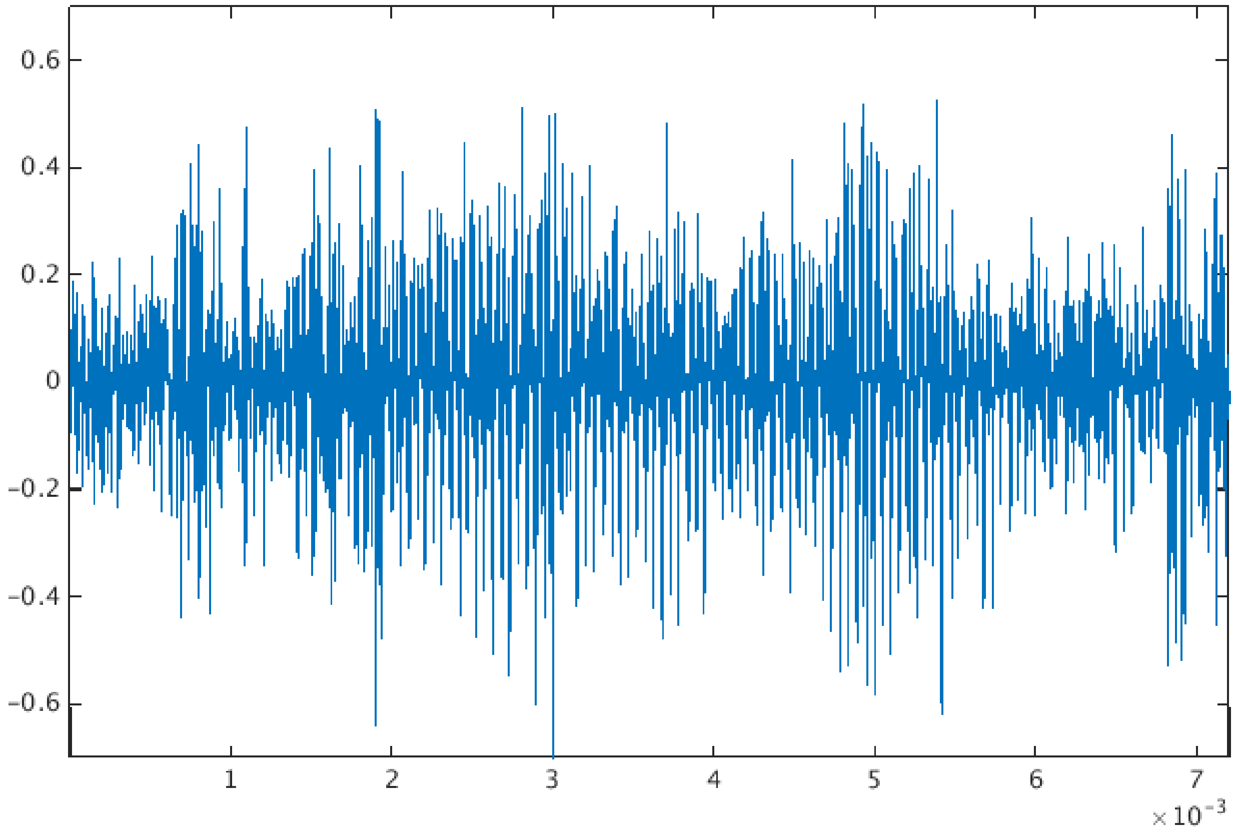
RDS mixing and filtering

Signal after mixing:



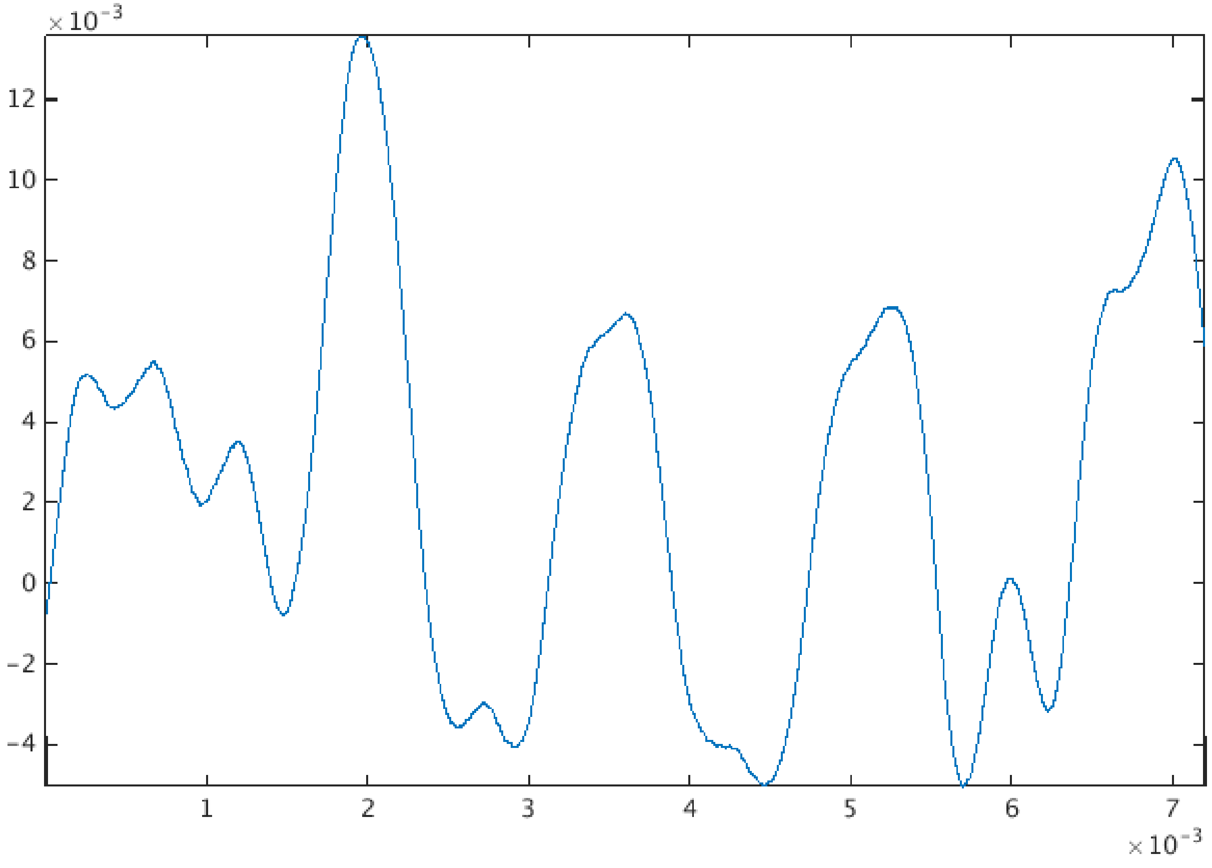
RDS mixing and filtering

Signal after mixing (zoom out):



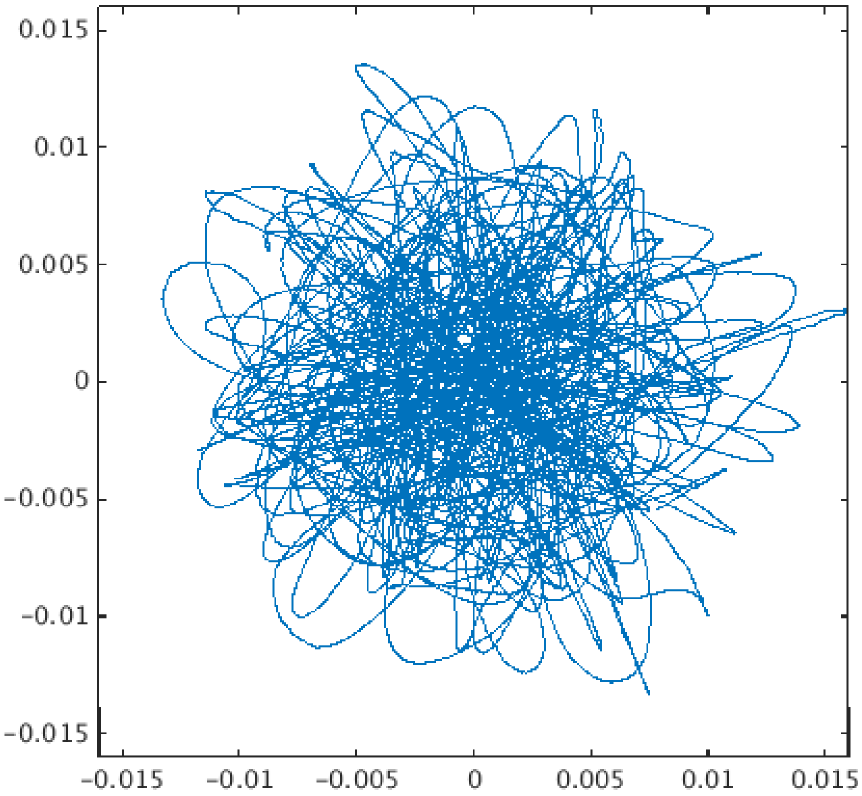
RDS mixing and filtering

Signal after mixing and filtering (real part):



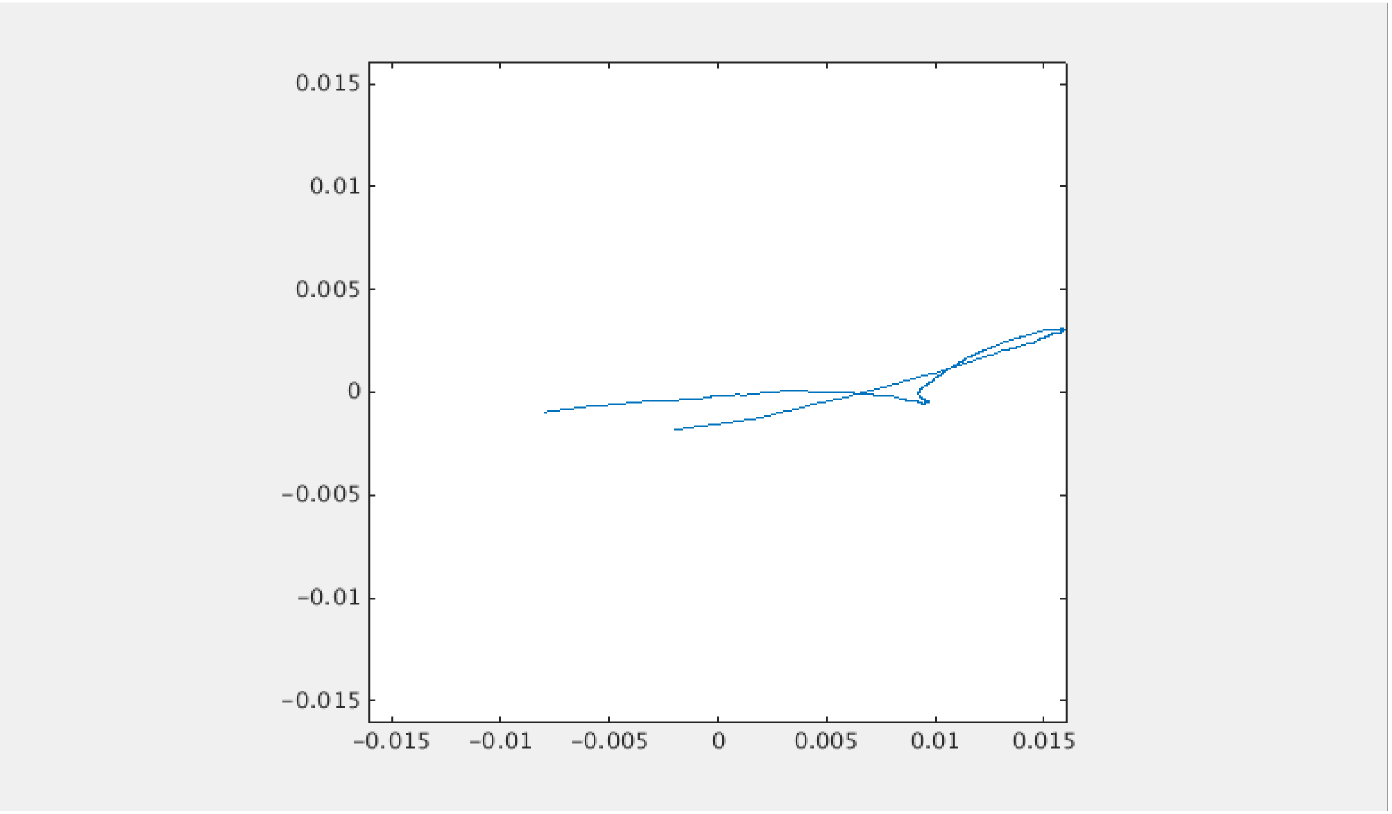
RDS baseband signal

Plotted in complex plane:



RDS baseband signal

Plotted in complex plane:



RDS baseband signal

- RDS binary data is modulated as:
  - AM: $\{-1, 1\} * e^{i 2\pi f_c t}$ → after mixing: $\{-1, 1\}$
  - "FM" (2PSK): $e^{i 2\pi f_c t - i\{0, \pi\}}$ → after mixing: $\{-1, 1\}$
- In the complex plane, the signal should stay on the real axis
- Rotation caused by carrier estimation error, we will compensate for that

Interpolation and decimation

- RDS symbol rate is 1187.5 Hz * 2 (bi-phase)
  - Lower than and not a multiple of the sampling rate
- Decimation (skipping samples) and interpolation (reconstructing signal inbetween samples) will return symbols with correct rate and sampling points
  - Need to know (symbol) clock phase and frequency
    - → clock recovery
  - Interpolation with first order estimate vs. filter

Symbol detection

- When the signal stays on the real axis and the symbol timing is correct, detection is easy:

  out = 1 when real(symbol) > 0   else out = 0

- Assuming the detection was correct, the angle of the (complex) symbol is a phase error feedback source

Carrier estimation correction

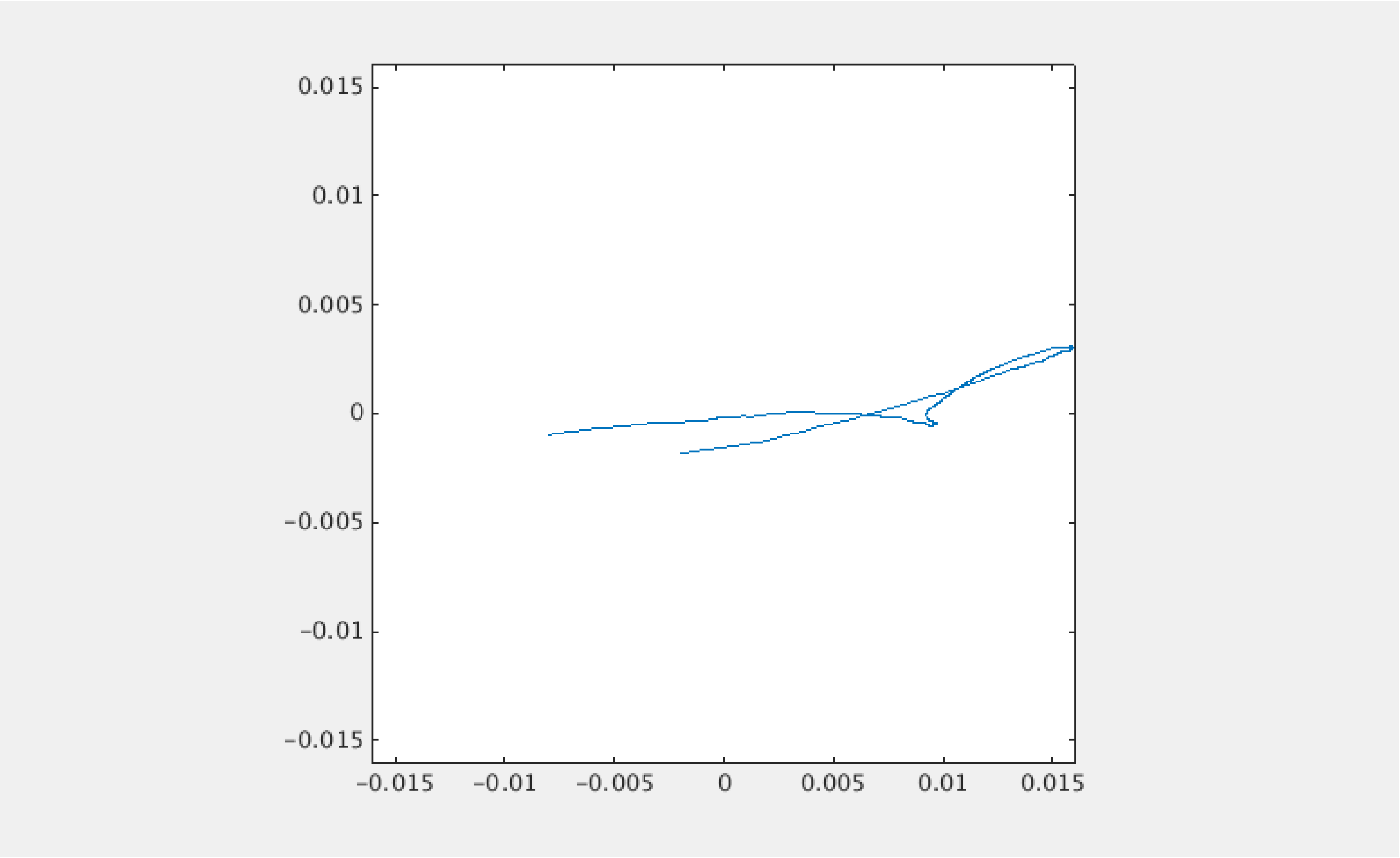
- The phase error from a detected symbol can be used to correct the phase of further samples
  - Low-pass filtering (IIR)
- A constant (very low frequency) term in the phase error means frequency error
  - Use for carrier frequency correction
  - Low-pass filtering (IIR)

Clock recovery

- With a good carrier phase and frequency estimation, the signal has (real) zero crossings in-phase with the bit-clock
  - Detect zero crossing
  - Adjust current clock phase
  - Low-frequency term of phase error = frequency error → adjust
- Bi-phase coding (1→10; 0→01)
  - Zero crossing in every bit

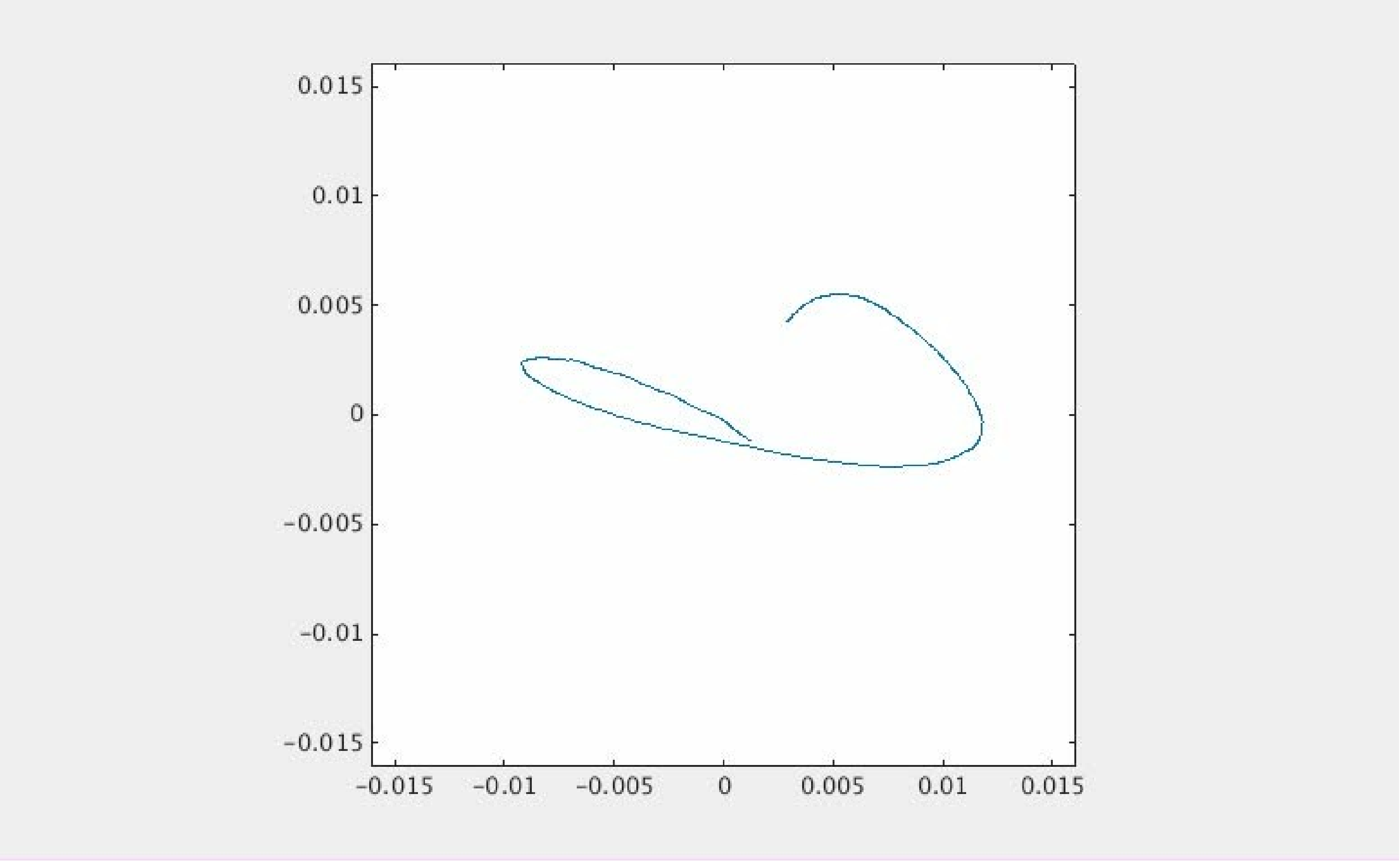
RDS baseband signal

Without f / θ correction:



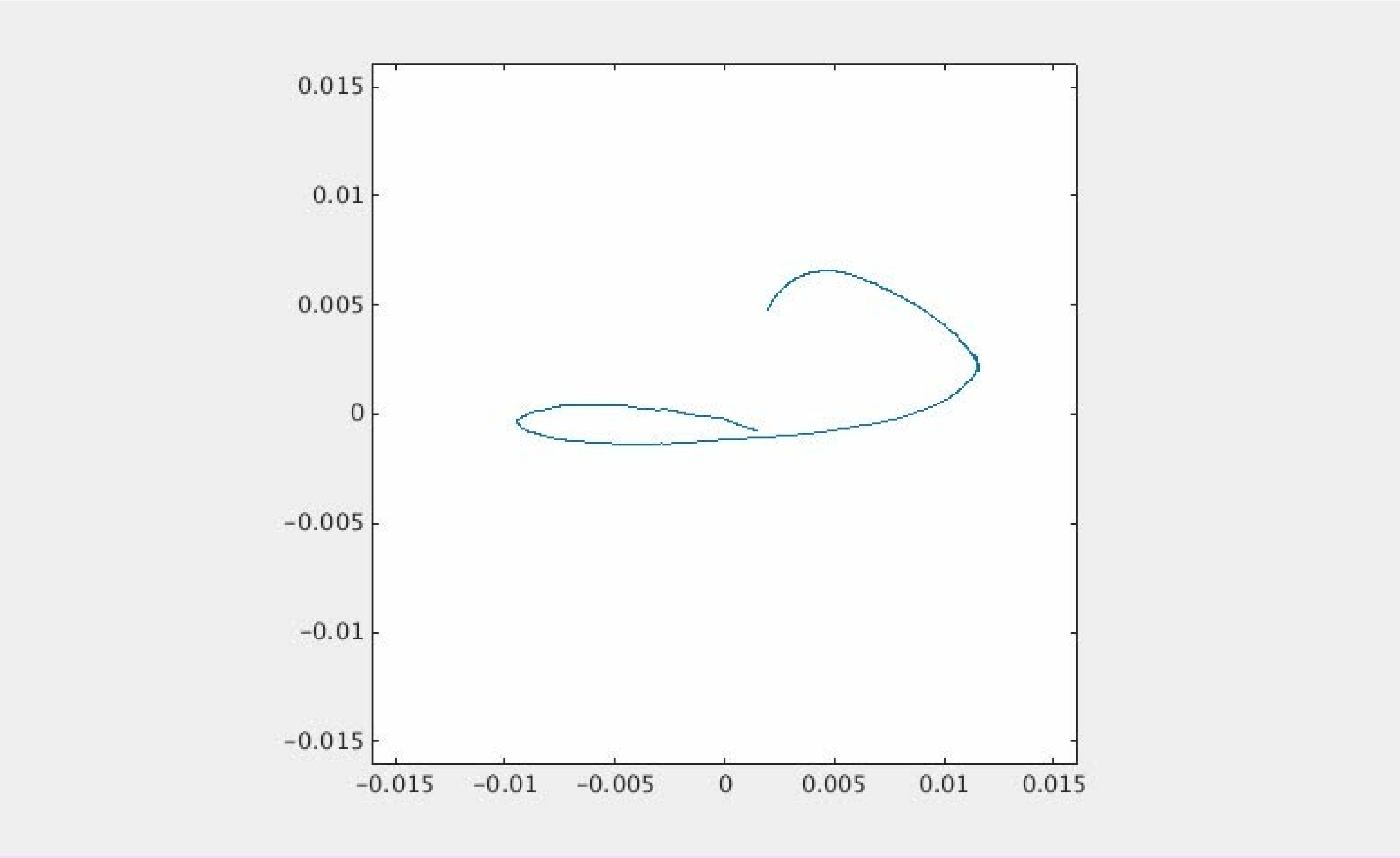
RDS baseband signal

With f correction:



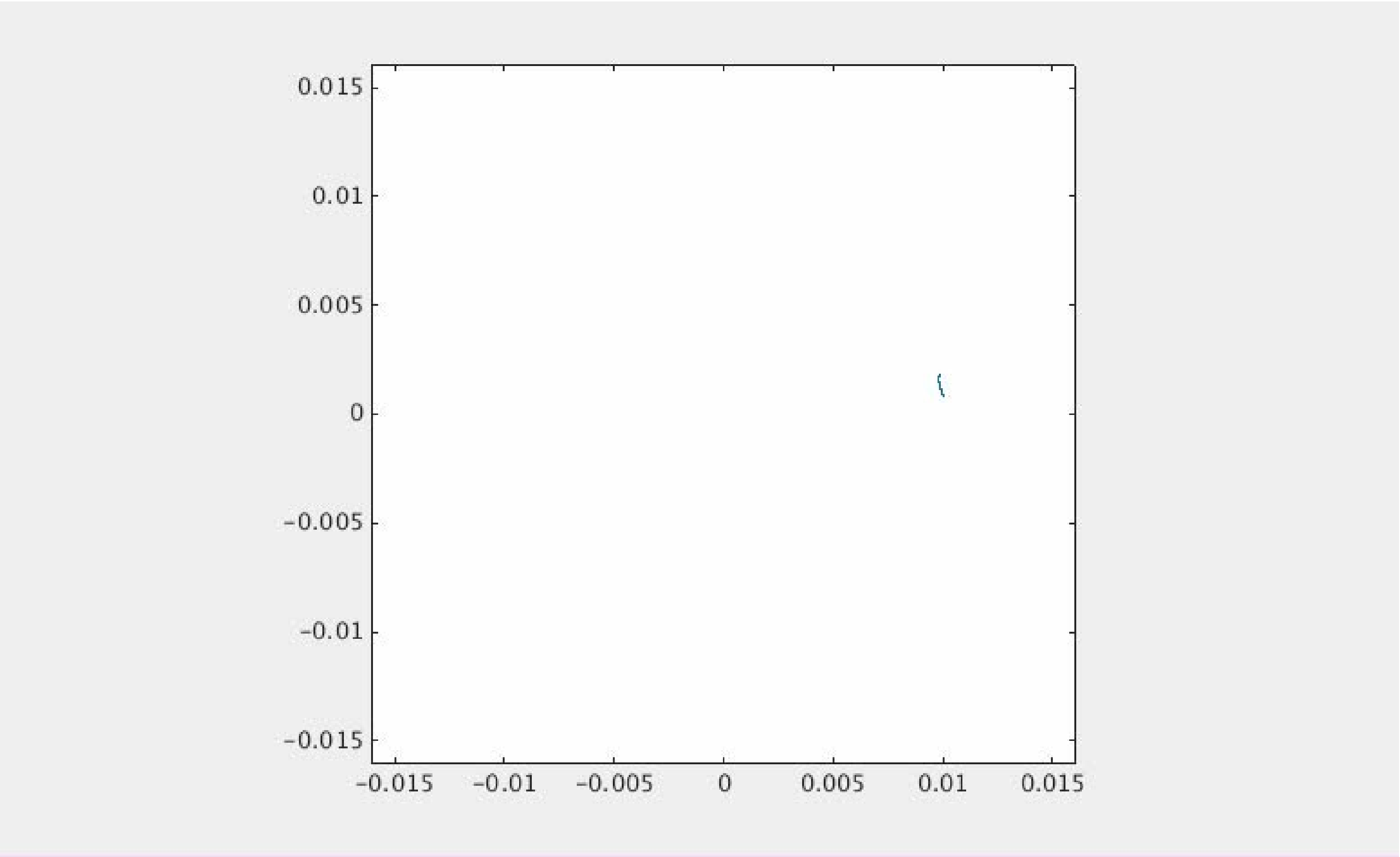
RDS baseband signal

With f and θ correction:



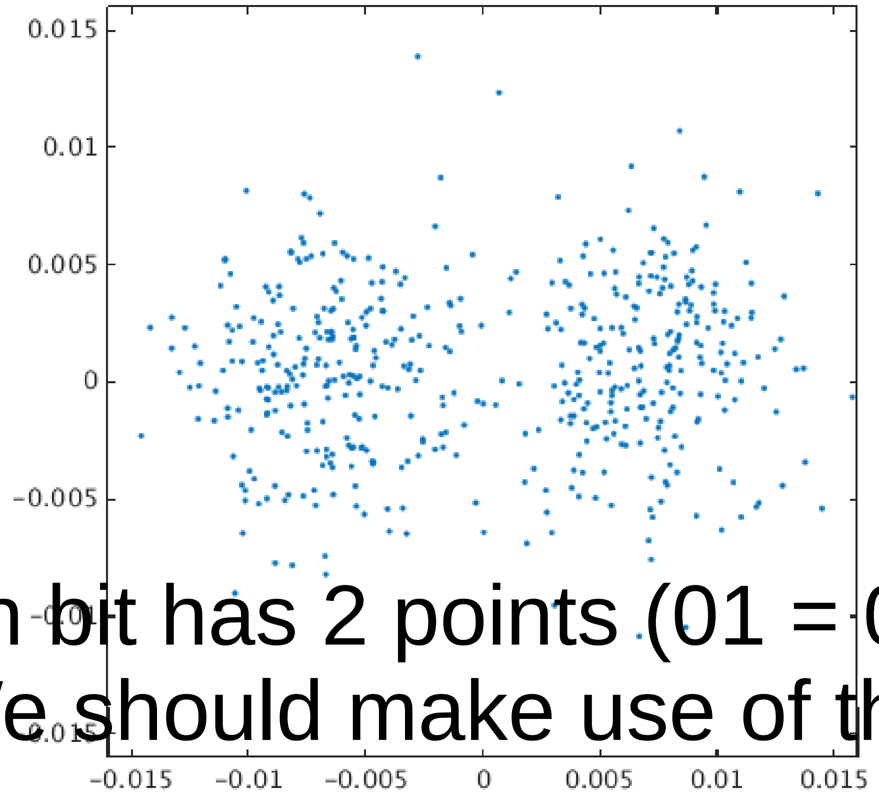
RDS baseband signal

After clock recovery, interpolation and decimation:



RDS baseband signal

Constellation diagram (IQ plot):



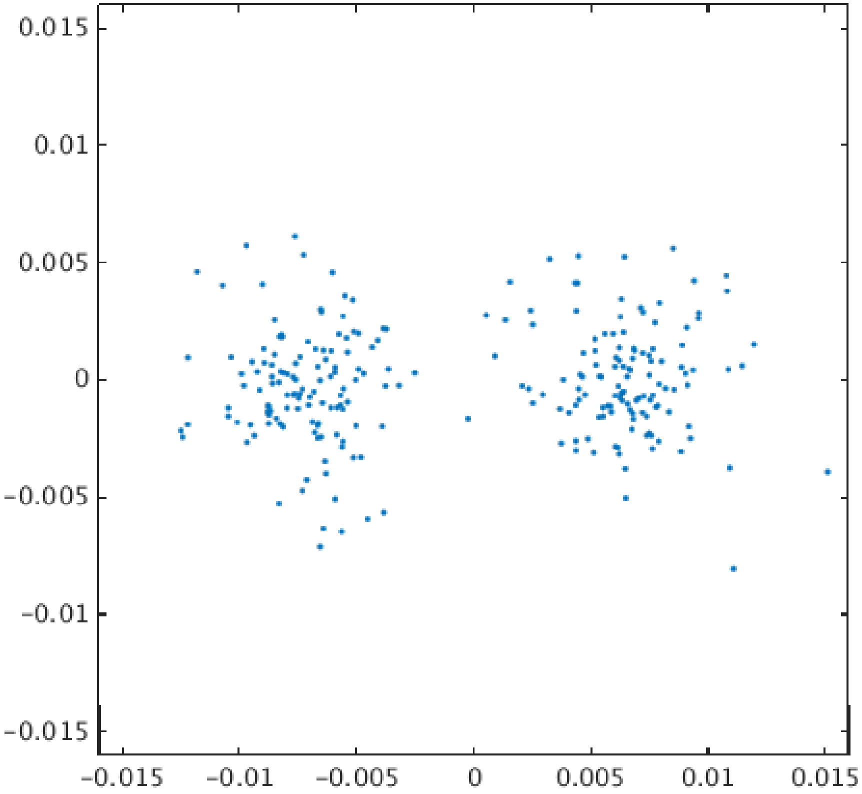
But each bit has 2 points (01 = 0; 10 = 1)

We should make use of this!

RDS baseband signal

Constellation diagram (IQ plot),

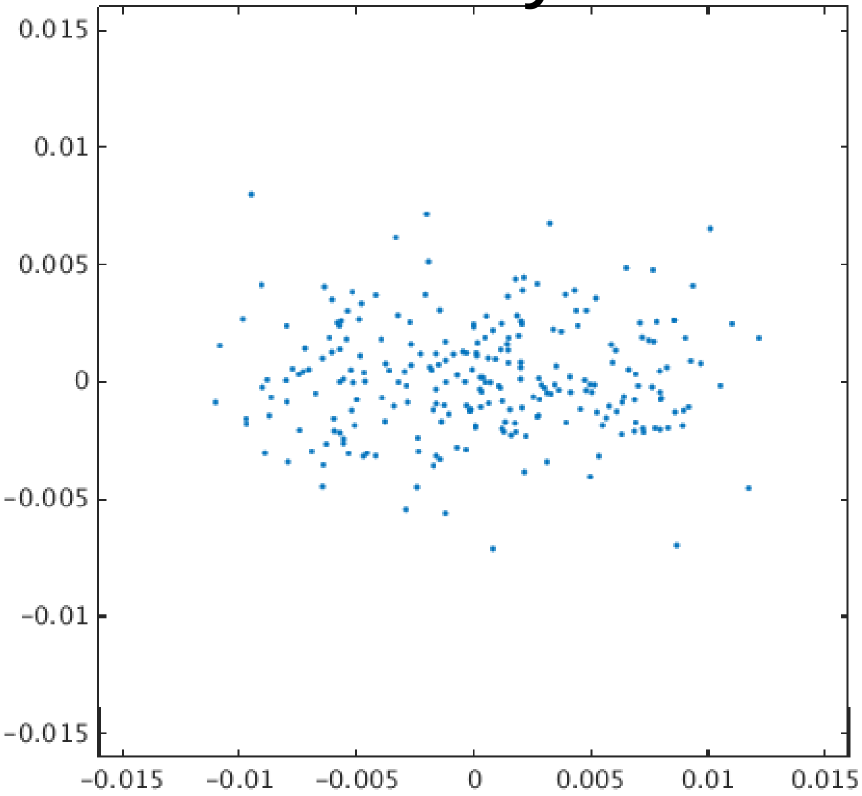
bi-phase decoded:



RDS baseband signal

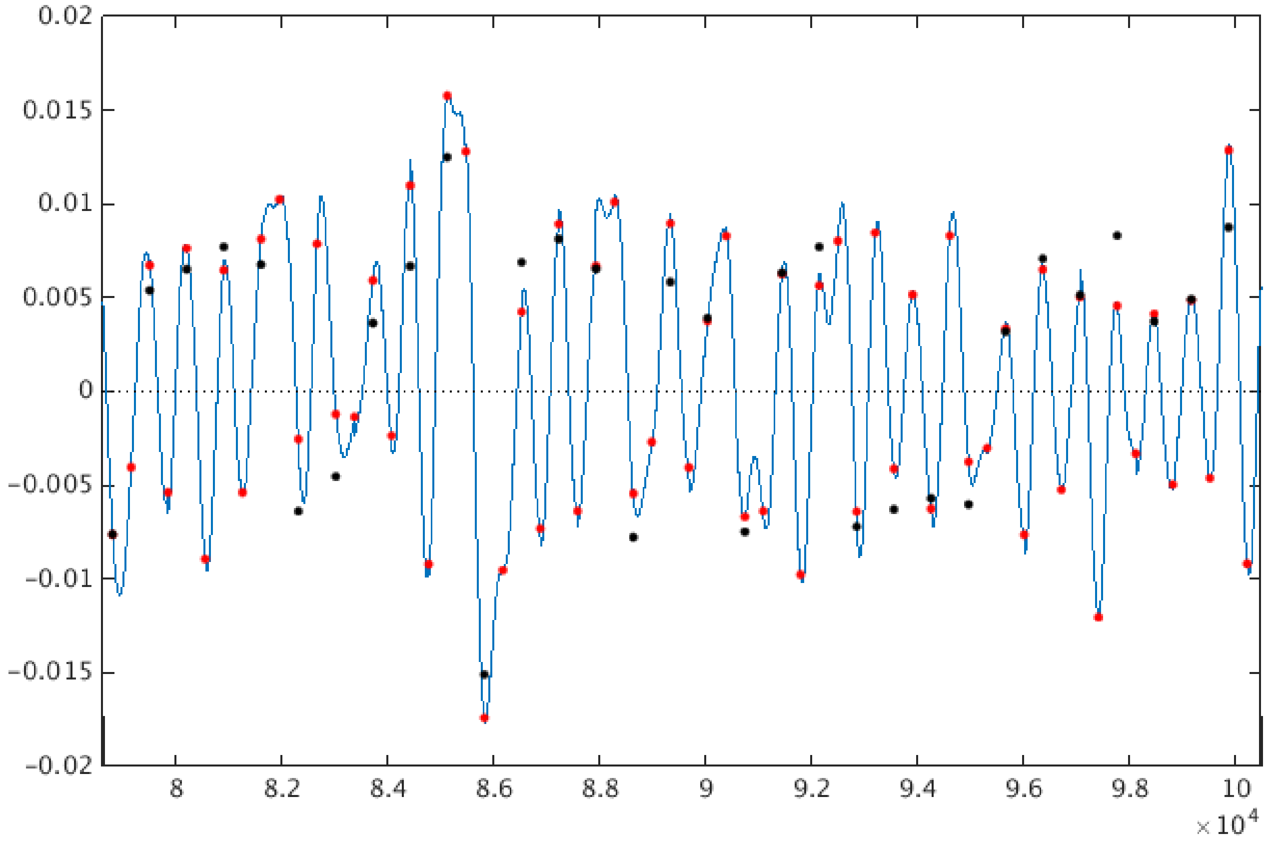
Constellation diagram (IQ plot),

bi-phase decoded – badly:



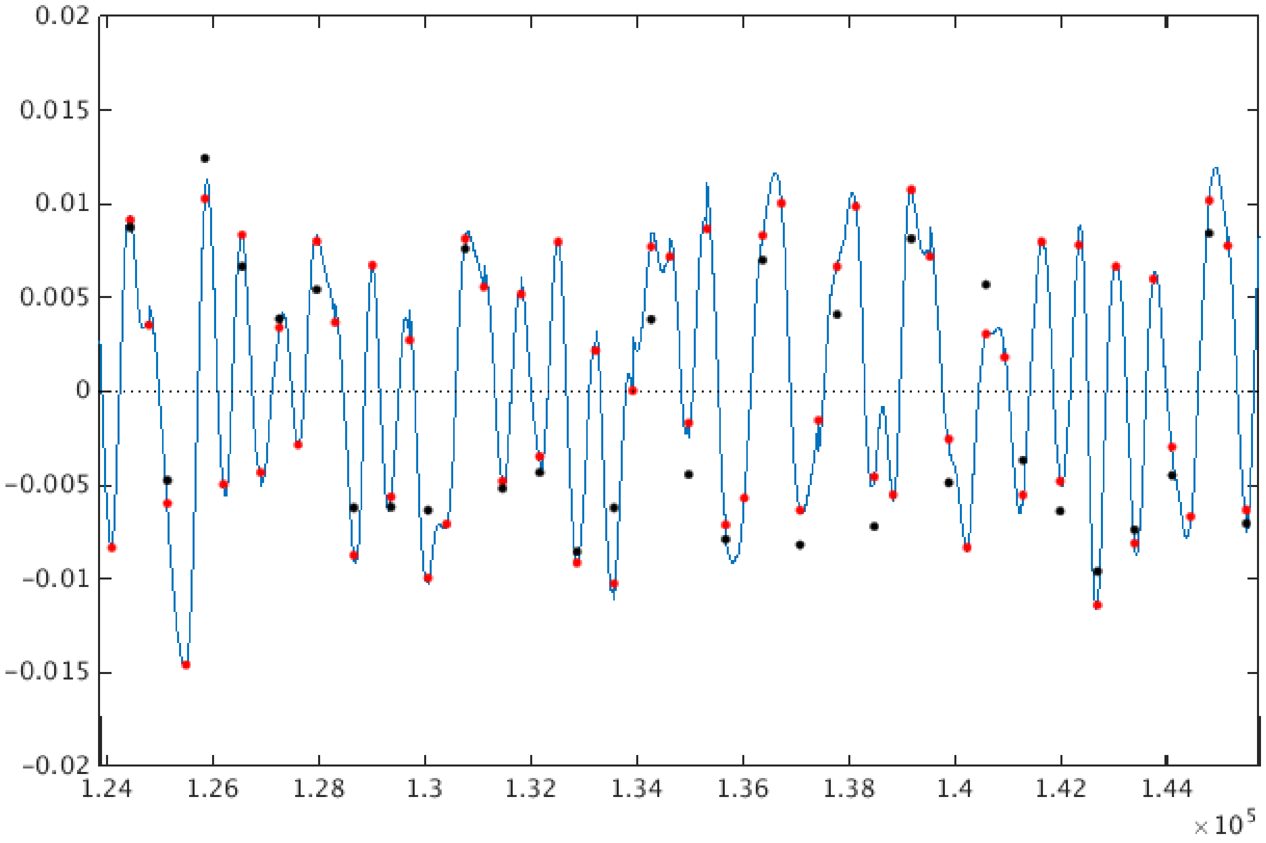
RDS baseband signal

Real part of the signal



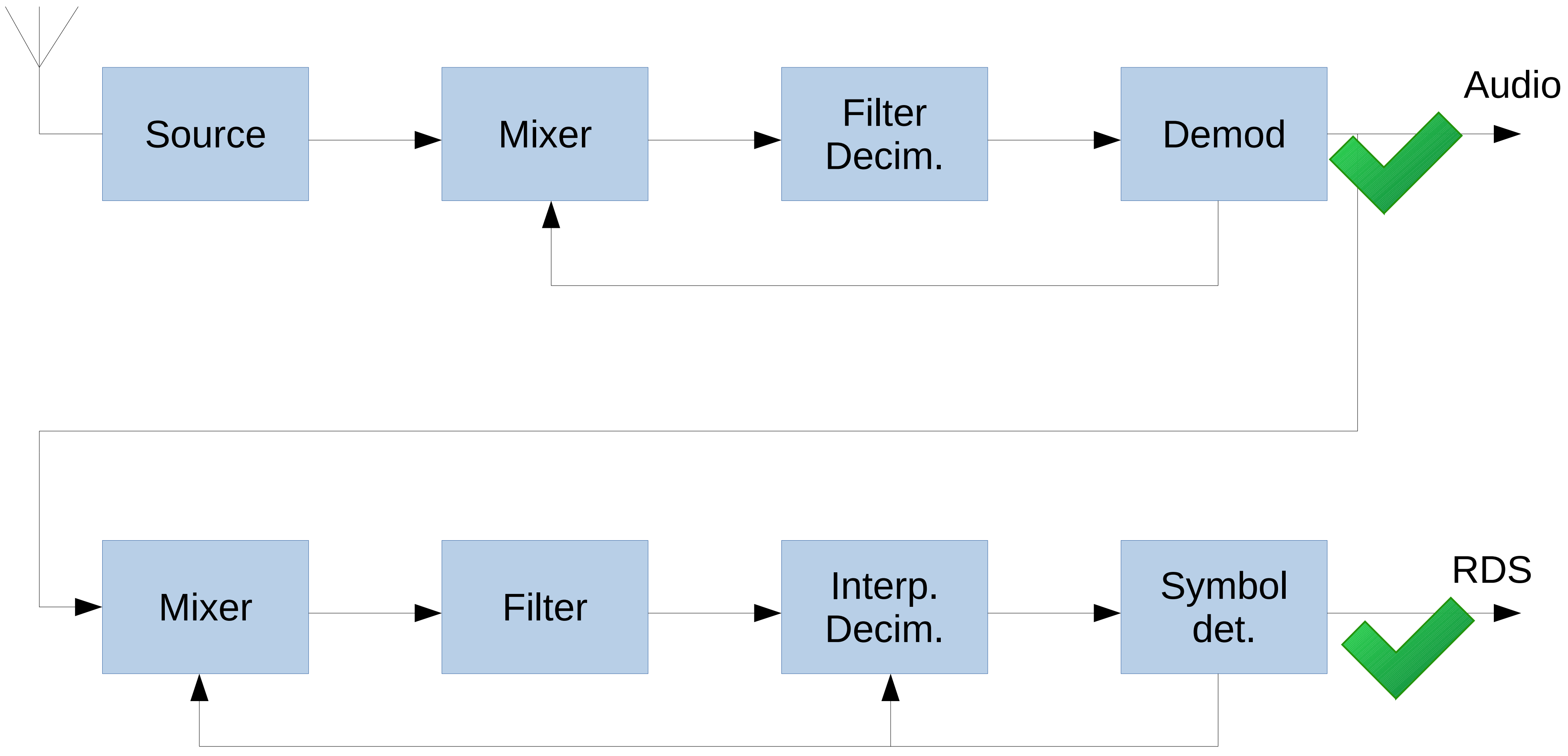
RDS baseband signal

Real part of the signal



Black dots are resulting 1s and 0s

All done!


Source
Mixer
Filter Decim.
Demod
Audio
Mixer
Filter
Interp. Decim.
Symbol det.
RDS

Introduction to Algorithms for Wireless Communication

LU Hardware Software Codesign
WS15/16

Robert Najvirt
TU Wien, Embedded Computing Systems Group

Wien, 16. 10. 2015