Wireless Networks In-the-Loop: Emulating an RF front-end in GNU Radio

Sebastian Koslowski, Martin Braun, Jens Elsner and Friedrich Jondral

SDR Forum 2010 European Reconfigurable Radio Technologies Workshop, Mainz, June 25, 2010

Communications Engineering Lab
Prof. Dr.rer.nat. Friedrich K. Jondral

Overview

Emulating an RF front-end in GNU Radio

- Introduction
- GNU Radio and the USRP
- Modeling digital signal processing
- Modeling analog signal processing
- Measurements & conclusion

Overview

Emulating an RF front-end in GNU Radio

Introduction

Faster verification and development of SRs with loop simulation

- For realistic results, effects of non-idealities need to be included in simulation
- Together with a channel emulator this facilitates loop simulation of software radio signal processing code

- Considered here: USRP with RFX2400 as an example of generic RF hardware

Emulator properties	Emulator Implementation
▪ Modular design ▪ Parametrizable ▪ Discrete representation of analog signals at an equivalent rate of 128 MHz	▪ C++ / GNU Radio block ▪ USRP interface

CEL

Overview

Emulating an RF front-end in GNU Radio

GNU Radio

Free Software Radio Framework

Properties

- Toolkit with extensive collection of atomic signal processing operations
- Signal sources and sinks offer interfaces to system and hardware
- USRP well supported

Design

- Signal processing in C++ (performance)
- Applications in Python (flow graphs)
- GNU Radio Companion offers graphical user interface

Execution

- Memory management and scheduling are handled by the GNU Radio framework

GNU Radio
the gnu software radio

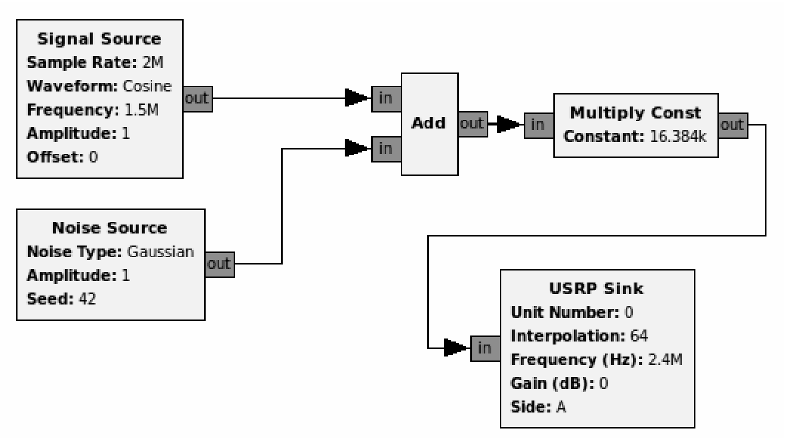


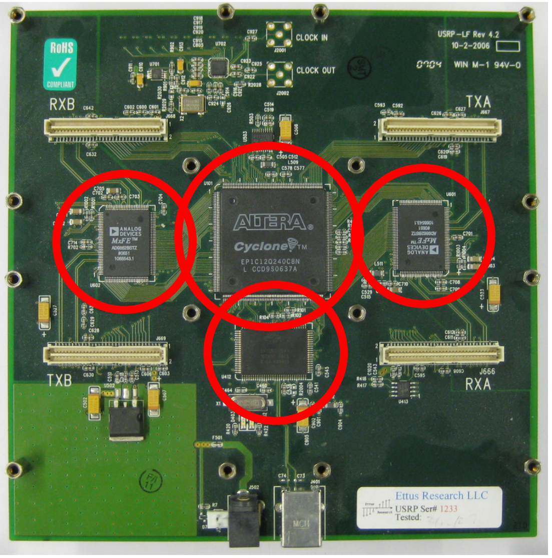
Figure source: http://gnuradio.org

Universal Software Radio Peripheral

Ettus Research USRP1 as an example of generic RF hardware

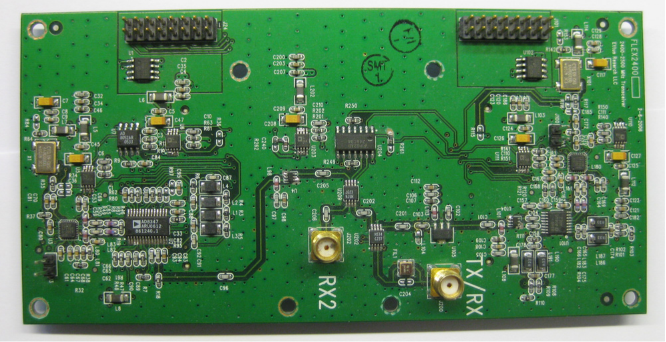
Motherboard

- Cypress FX2 USB Controller
  → Data exchange with host PC using USB 2.0
- Cyclone FPGA
  → Sample rate conversion and IF mixer
- Analog Devices AD9862 Mixed Signal Proc.
  → A/D conversion, interpolation, digital IF mixer



RFX2400 Daughterboard

- Direct conversion transceiver for 2.4 GHz band
- Two local oscillator signals, for Rx / Tx
- Maximum output power +17 dBm
- Integrated AGC (not modeled in emulator)

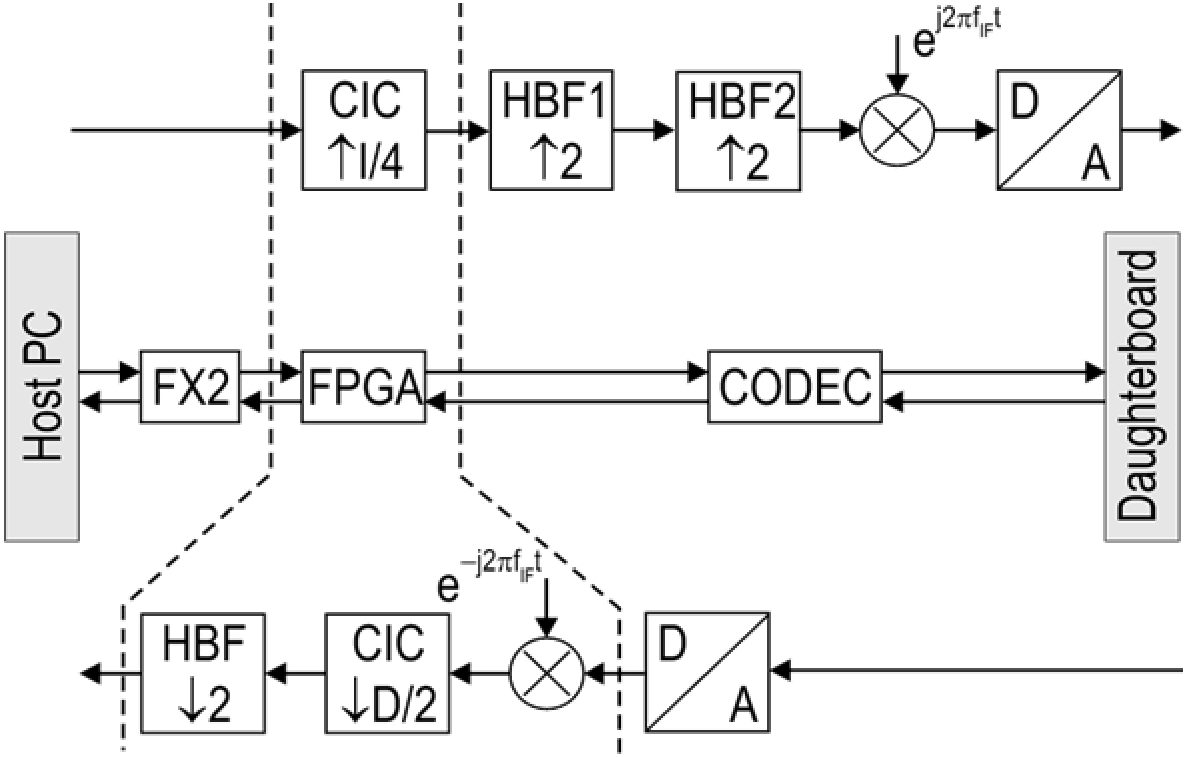


Overview

Emulating an RF front-end in GNU Radio

Model of the USRP motherboards

Overview



Elements of digital signal processing
▪ A/D and D/A conversion ▪ Digital IF mixer ▪ Variable sample rate conversion with CIC- and halfband filters

CEL

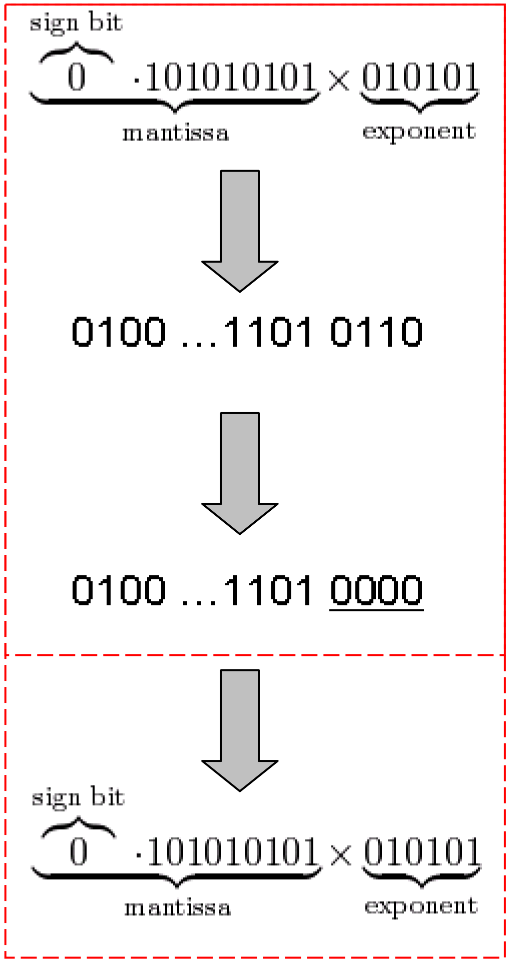
Model of the USRP motherboard

Processing in transmit and receive path

Analog/digital conversion

- Floating point to model analog signals in emulation
- A/D conversion with 12 bit/s and 64 MS/s
  → Decimation by factor 2 needed
- D/A conversion with 14 bit/s and 128 MS/s

→ Quantization noise / bit resolution by rounding



Encoding of samples

- FPGA and CODEC use fix point arithmetic
- GNU Radio framework offers high performance filters for floating point
- In emulator: floating point arithmetic for digital filters an mixers

→ Conversion to fix point between stages

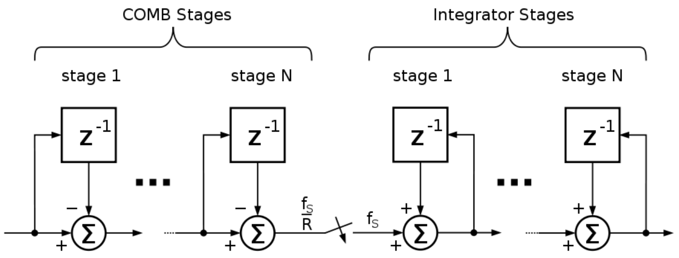
Model of the USRP motherboard

Processing in transmit and receive path

Cascaded Integrator Comb filter

- Efficient, recursive implementation of a multi-stage moving average filter for variable sample rate conversion
- USRP uses 4-stage CIC filter

→ Implemented in fix point arithmetic as GR block, scaling of output signal by bit shifting



Halfband filter

- Efficient FIR low pass filter; cut-off at band middle
- About half of all coefficients are zero

→ Used to interpolate / decimate by a factor of 2

- Tx: built as in data sheet
- Rx: coefficients as in USRP1 FPGA code

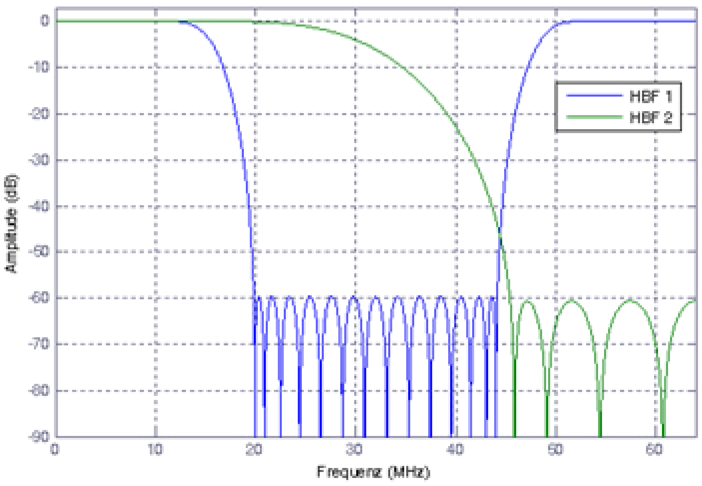


Figure source: http://de.wikipedia.org/wiki/Cascaded-Integrator-Comb_Filter

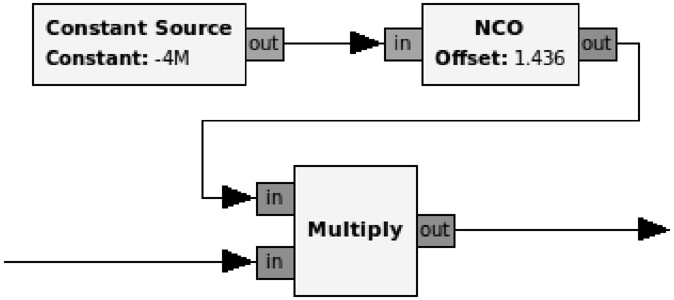
Model of the USRP motherboard

Processing in transmit and receive path

Digital IF mixer

- Digital mixer based on CORDIC algorithm
- Shifts base band signal to digital IF
- Together with analog mixer used to shift signal to carrier frequency

→ In emulator: complex multiplication with NCO, random initial phase



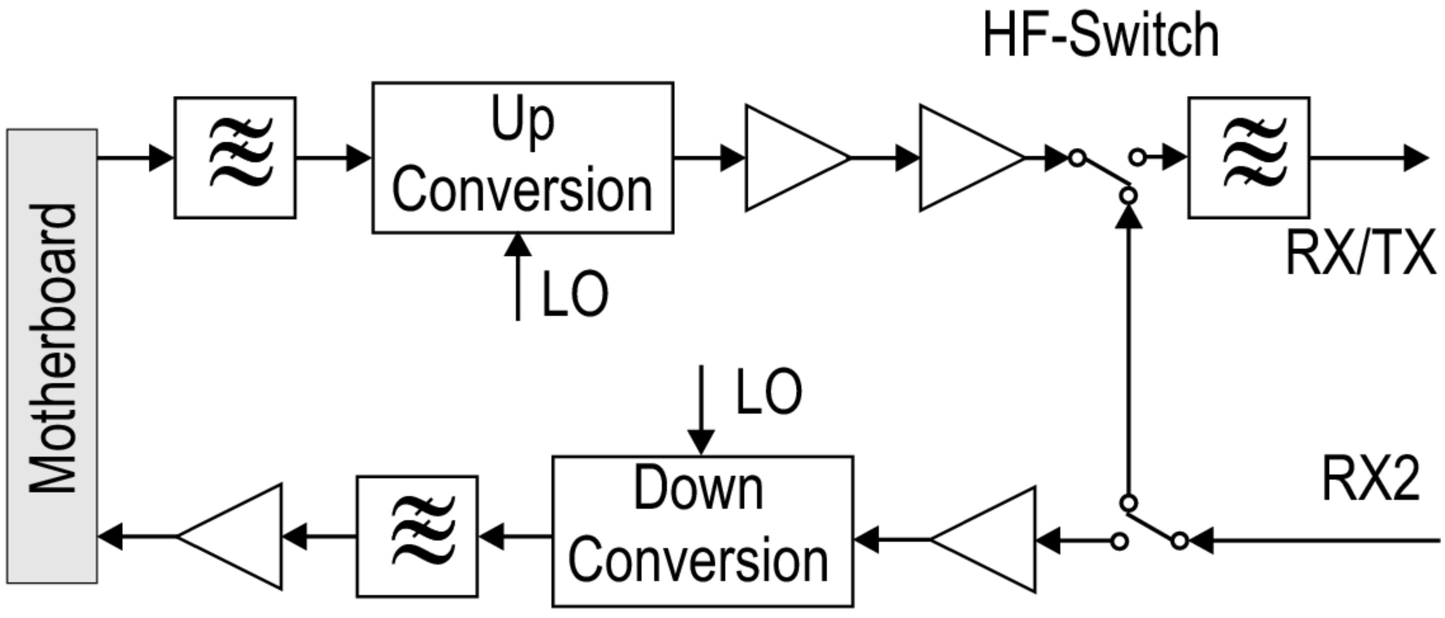
Overview

Emulating an RF front-end in GNU Radio

- Introduction
- GNU Radio and the USRP
- Modeling digital signal processing
- **Modeling analog signal processing**
- Measurements & conclusion

Model of the RFX2400 USRP daughterboard

Overview



Typical effects of analog signal processing	
▪ Phase noise	▪ Non-linearities
▪ IQ-imbalance	▪ Noise

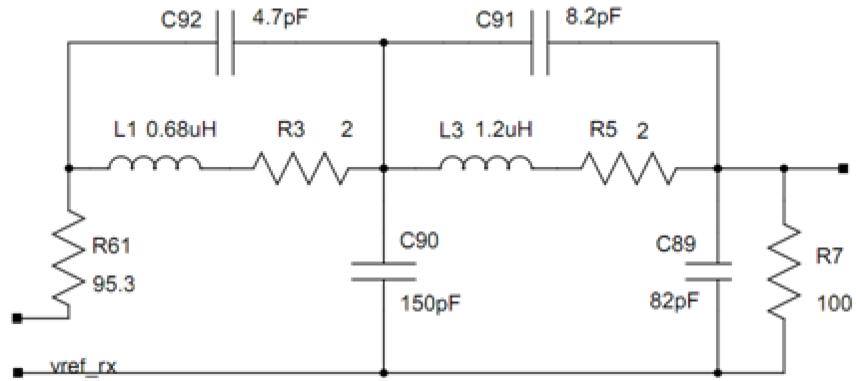
Model of the RFX2400 USRP daughterboard

Discretization of analog filters

Anti aliasing filter

- After down shifting, before A/D conversion
- 5th order IIR low pass filter, cut-off frequency 20 MHz, attenuation 6 dB
- Discretization with bilinear transform
- Additional tweaking of poles and zeros due to aliasing effects needed

→ Approximated well in pass band, higher attenuation in stop band



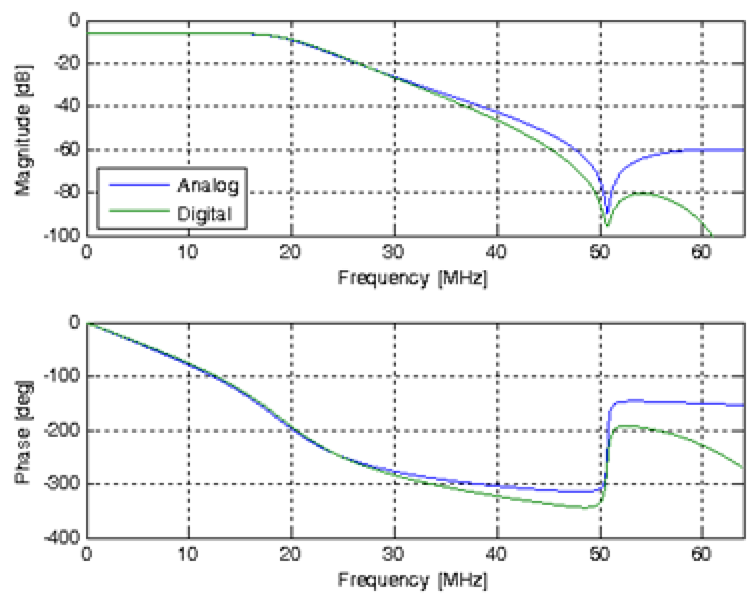


Figure source: Ettus Research LLC (http://ettus.com), data sheet RFX2400

Model of the RFX2400 USRP daughterboard

Discretization of analog filters

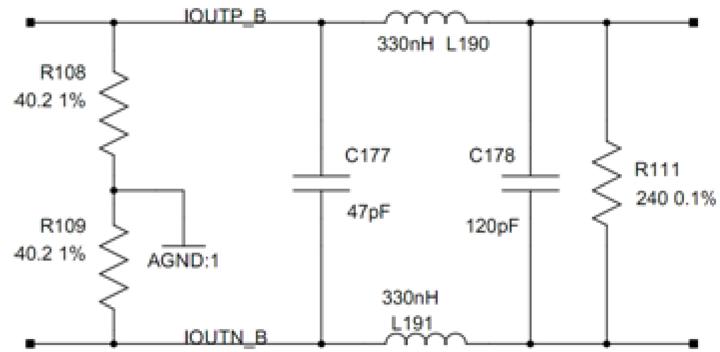
Anti image filter

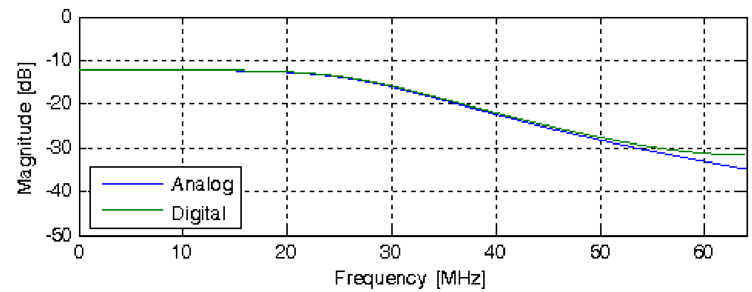
- After D/A conversion, before mixer
- 3rd order IIR low pass, cut-off frequency ~28 Mhz, attenuation 12 dB
- Discretization with impulse invariance method

ISM bandpass filter

- Magnitude response flat in simulation bandwidth
- Phase response not specified

→ Not modeled





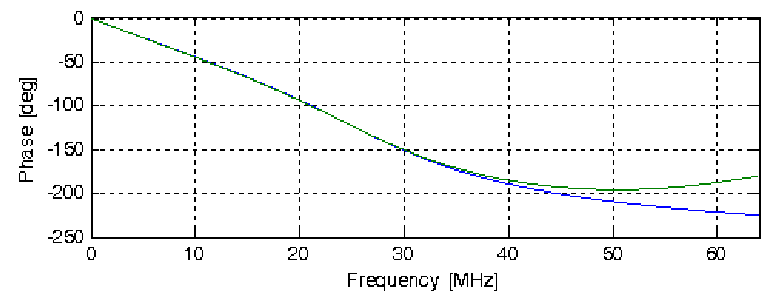


Figure source: Ettus Research LLC (http://ettus.com), data sheet RFX2400

Model of the RFX2400 USRP daughterboard

Phase noise

Phase noise

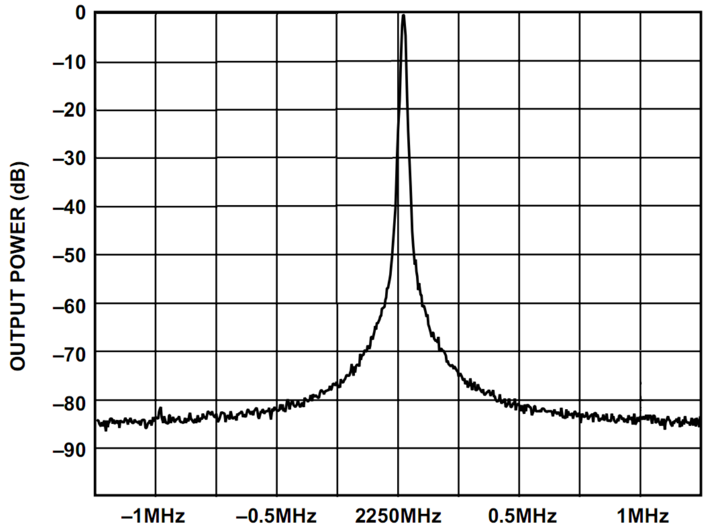
- LO signals of analog mixer are not ideal
→ 1/f-noise

- Created with noise shaping filter

$$H_{\mathrm{PN}}(z) = \frac{1}{\sqrt{1 - z^{-1}}}$$

- Approximation of IIR low pass coefficients with polynomial expansion

- In emulator: phase noise modulated onto the signal in baseband



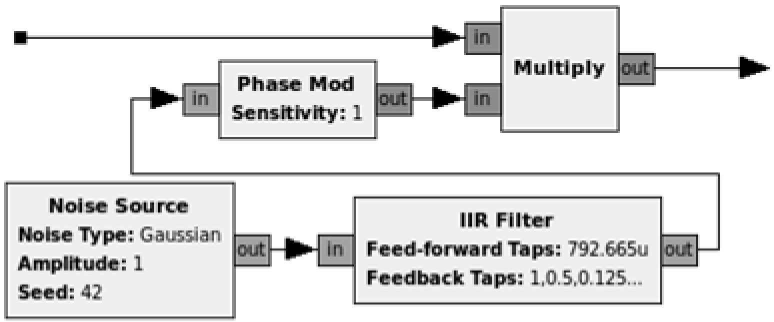


Figure source: Analog Devices, Integrated Synthesizer and VCO data sheet

Model of the RFX2400 USRP daughterboard

IQ imbalance

IQ imbalance

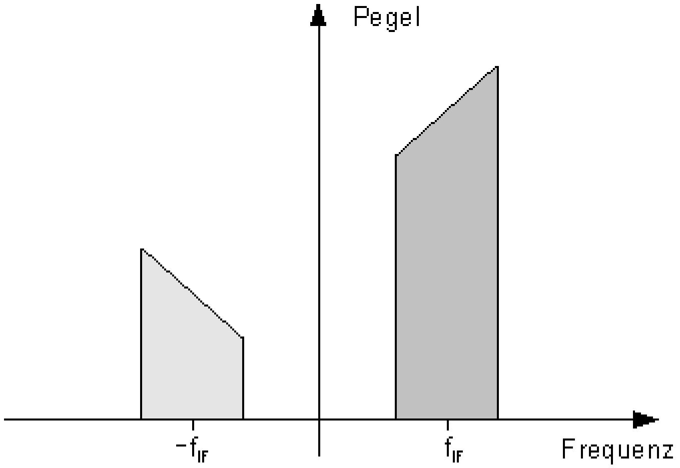
- Phase shift in mixer is not perfect
  → Variations in amplitude and phase

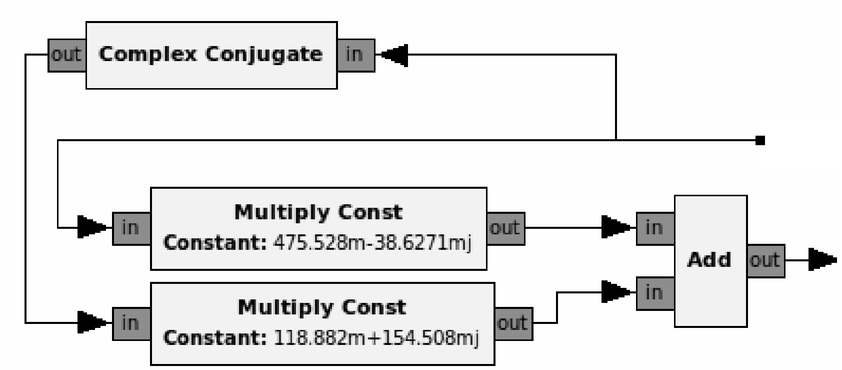
$$r'(t) = (1-\varepsilon) r_{bp}(t) \cos(\omega_c t + \phi) - j(1+\varepsilon) r_{bp}(t) \sin(\omega_c t - \phi)$$

- Image spectrum not suppressed fully

$$r'(t) = c_1 \cdot r(t) + c_2 \cdot r^*(t)$$

$$c_1 = \frac{1}{2}(\cos(\phi) - j\varepsilon \sin(\phi))$$
$$c_2 = \frac{1}{2}(\varepsilon \cos(\phi) + j \sin(\phi))$$





Model of the RFX2400 USRP daughterboard

Intermodulation products

Non-linearities

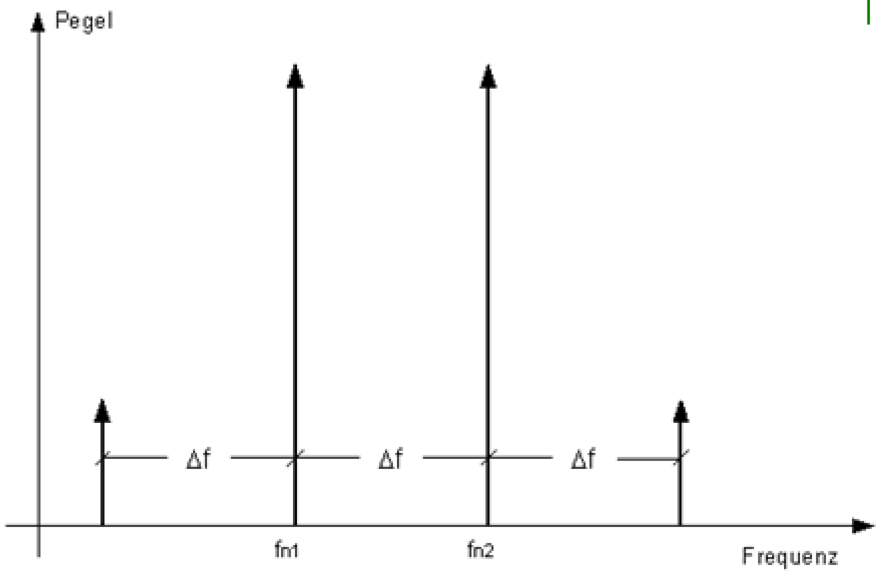
- Active components are only approximately linear (amplifier, mixer)
- 3. order intermodulation in baseband

$$y(t) = x(t) - \frac{3}{4}\alpha_3\, x^2(t)x^*(t)$$

- In model: unification of all amplifying components and their non-linearities

→ Model combined effect

- For baseband amplification quadratic mixer term is also relevant



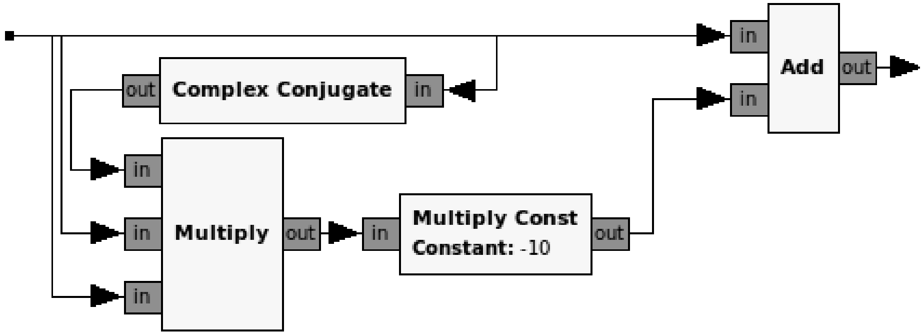


Figure source: http://de.wikipedia.org/wiki/Intermodulationsprodukt

Model of the RFX2400 USRP daughterboard

Noise figure of receiver

Receiver noise

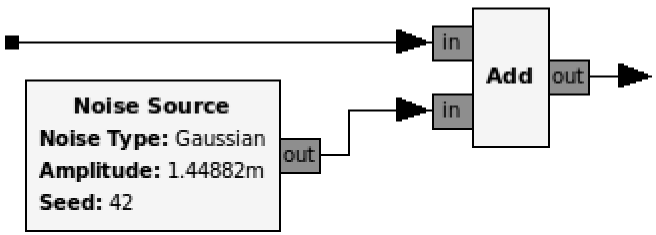
- Every component adds noise to signal and reduces SNR along the analog processing chain
  → Characterization by noise figure
- Combining noise figures and gain of all components with Friis' formula
- In model: Johnson-noise is additive white Gaussian noise

→ $\mathrm{NF} \geq 5\mathrm{dB}$ at RX2 port

- NF depends strongly on AGC level

$$\mathrm{NF} = 10\log F = \mathrm{SNR}_{in,\mathrm{dB}} - \mathrm{SNR}_{out,\mathrm{dB}}$$

$$F_{\mathrm{total}} = 1 + (F_1 - 1) + \frac{F_2 - 1}{g_1} + \frac{F_3 - 1}{g_1\, g_2} + \ldots$$



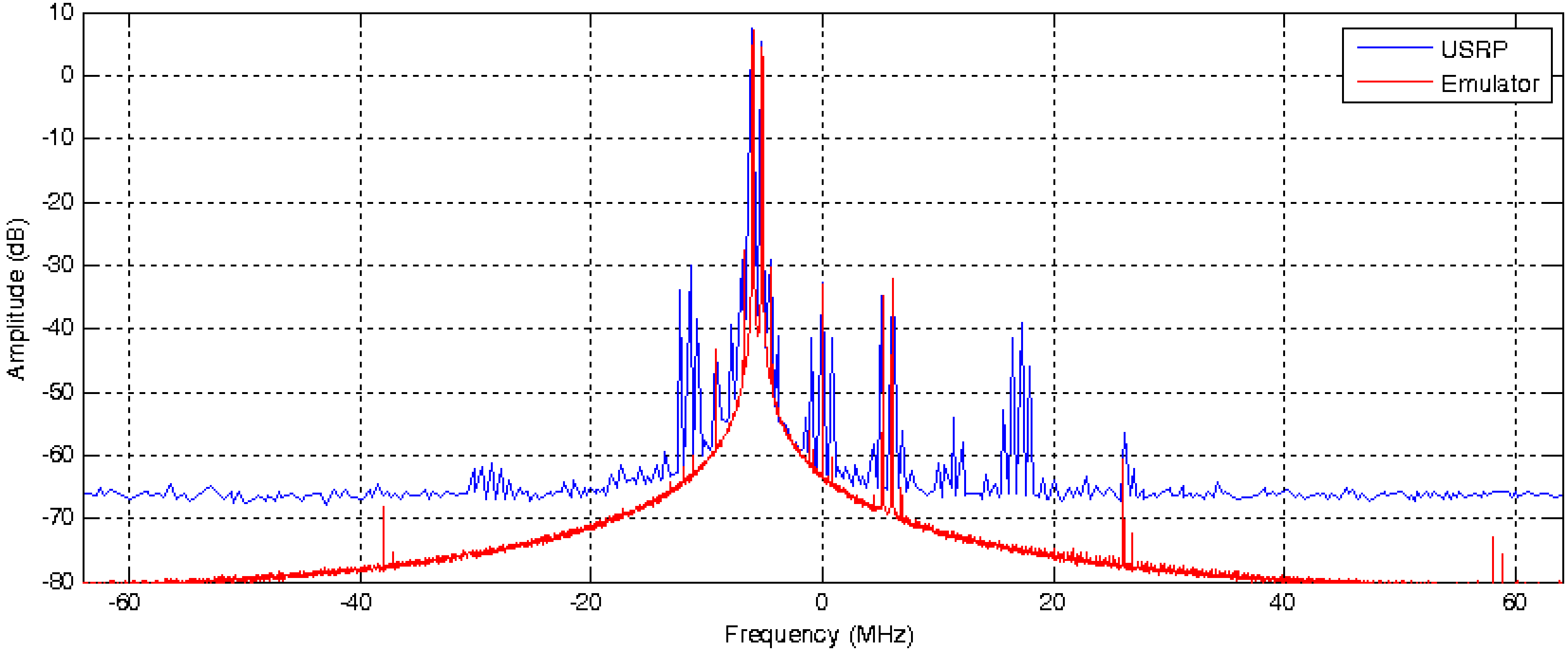
Overview

Emulating an RF front-end in GNU Radio

- Introduction
- GNU Radio and the USRP
- Modeling digital signal processing
- Modeling analog signal processing
- **Measurements & conclusion**

Measurements and comparison

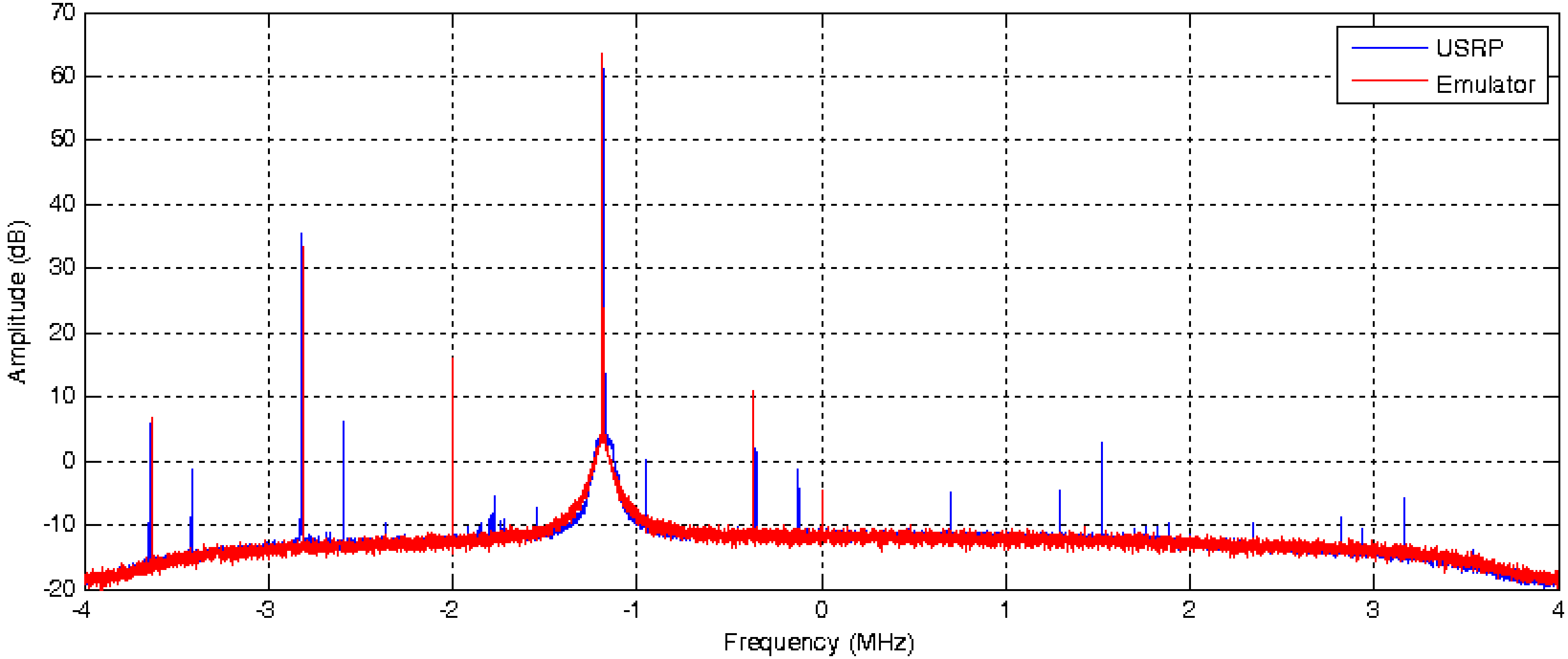
Transmit path



Measurement setup
▪ 2 tone signal with 800 kHz separation, carrier frequency 2.45 GHz, IF at -6 MHz → IQ imbalance, carrier, IM products of 3rd order, phase noise, quantization noise visible ▪ Not included: thermal noise, additional non-linearities

Measurements and comparison

Receive path



Measurement setup
▪ Sinusoid at 1.2 MHz below carrier frequency, IF at -2 MHz. → IQ imbalance, baseband non-linearities of 2nd order, phase noise

Conclusion

Emulating an RF front-end in GNU Radio

- Modeled the core elements of the signal processing chain
  - Variable sampling rate conversion with CIC- and halfband filters
  - Digital IF mixing
  - D/A & A/D conversion (limited sample resolution, quantization noise)
  - Analog filters

- Modeled effects of analog RF hardware
  - Phase noise
  - IQ imbalance
  - Receiver noise
  - Non-linearities

Q&A / Discussion

Thank you for your attention!