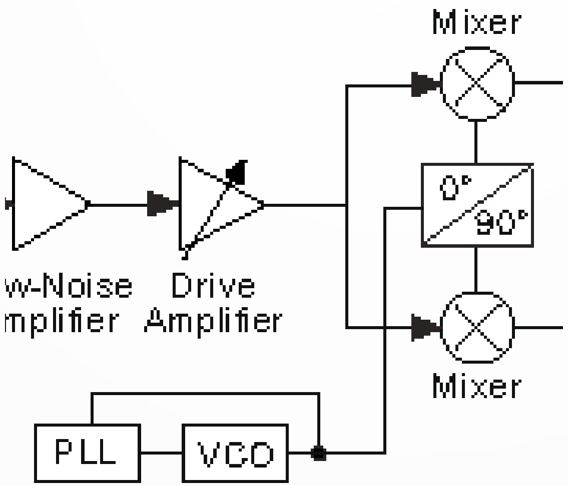
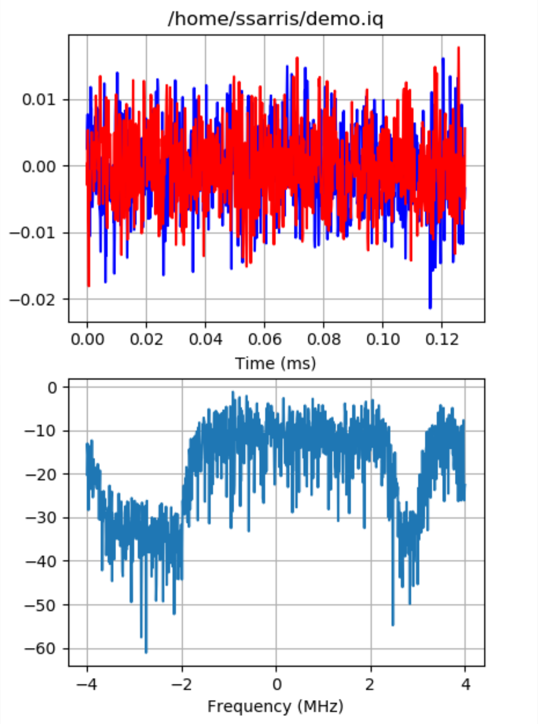
RF Quadrature Signals

Mathematical Model
Hardware Implementation of Model
Digital Data Representation

Spiro Sarris. 07 / 2018

$$A \cdot \cos(\omega \cdot t + \phi)$$

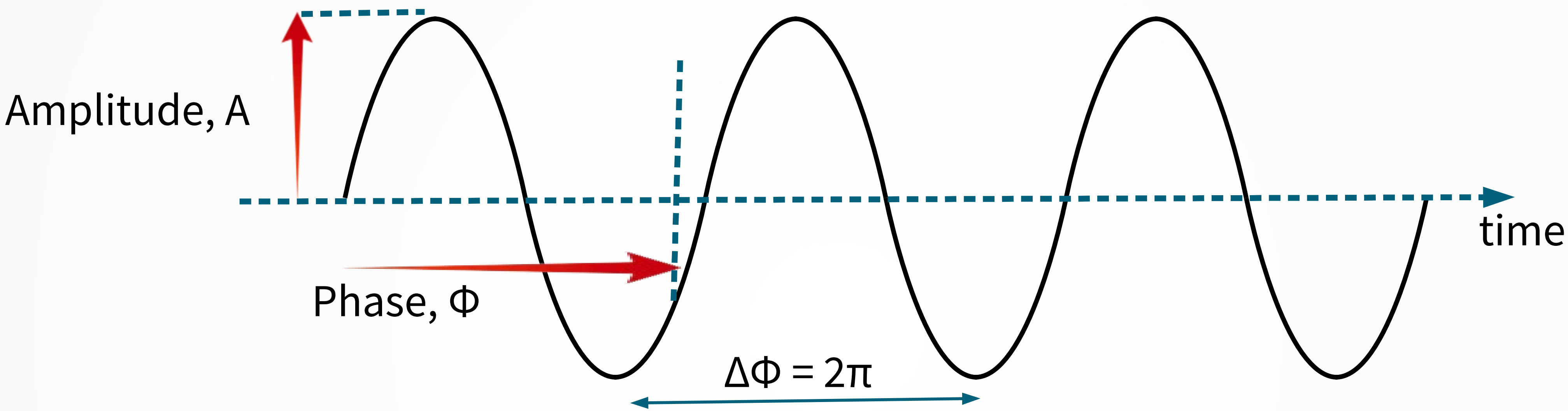




Outline

1. Amplitude, phase. Why is it interesting?
2. Euler's formula
3. Mathematical model of signal
4. Receiver hardware implementation of model
5. Analog to digital conversion
6. Convenience in software
7. Tradeoffs between I/Q sampling and real-value sampling

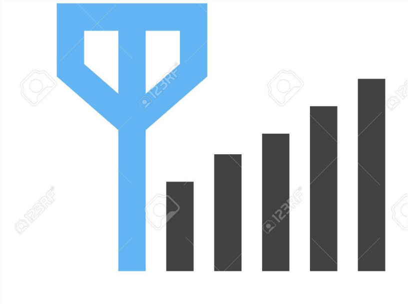
1. Amplitude, Phase



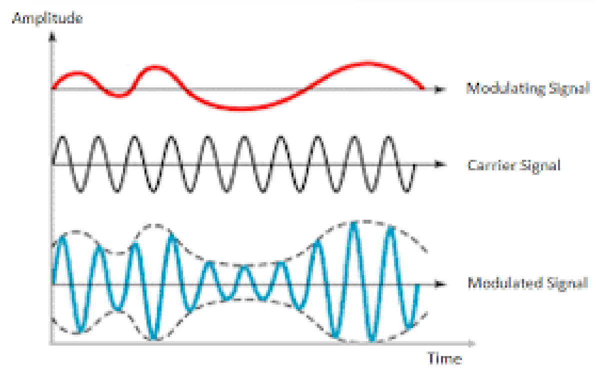
- ΔΦ = 2π radians for 1 cycle in time and for 1 wavelength (λ) [m] in space
- Angular frequency (ω) [radians/sec] is the rate of change of phase vs. time $\omega = \frac{d\phi}{dt}$
- Temporal frequency (f) [Hz] is the number of cycles per second $f = \frac{\omega}{2\pi}$

1. Amplitude

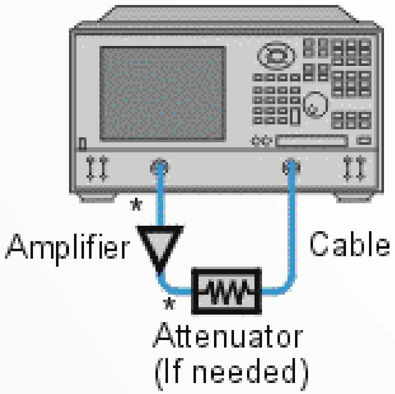
RSSI – Received Signal Strength Indicator



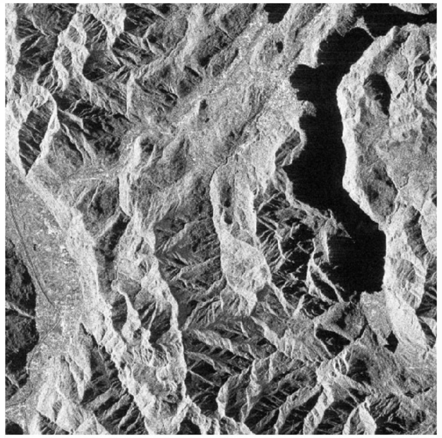
Communication Modulation



Component Gain / Loss

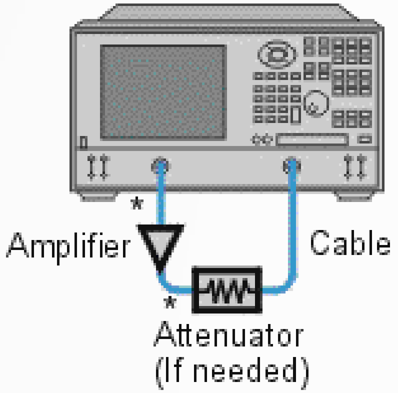


Synthetic Aperture Radar Image

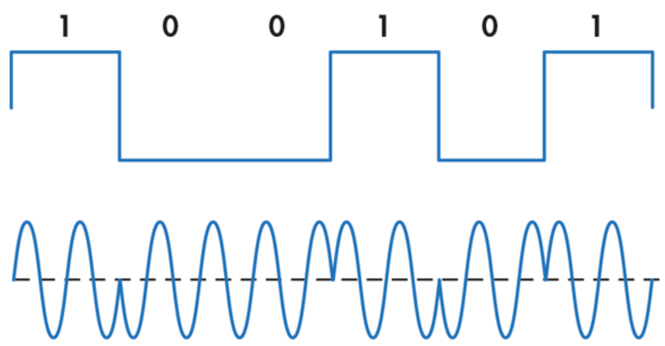


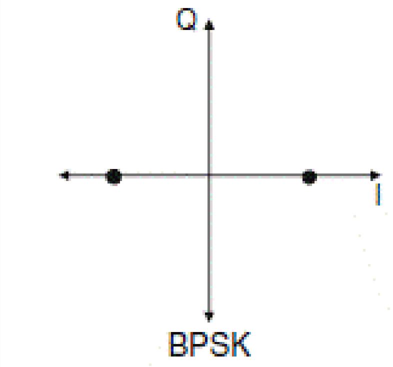
1. Phase

Length / Distance Measurement

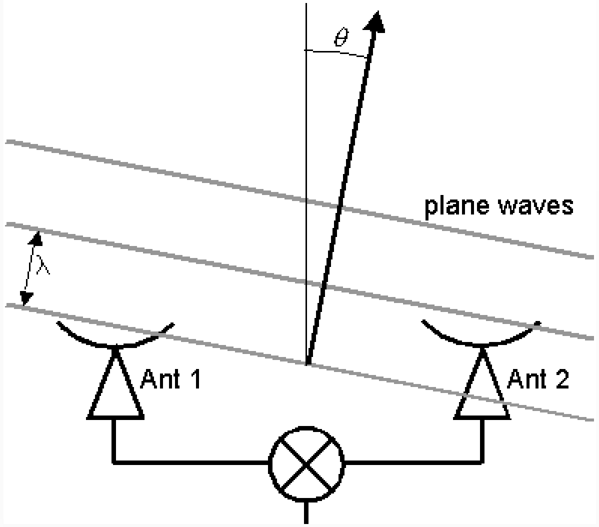


Data Communication Modulation

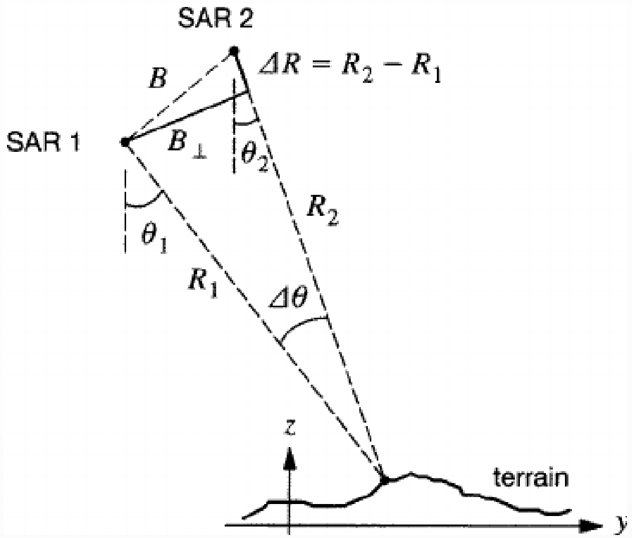




Radio Interferometer



Synthetic Aperture Radar Interferometer



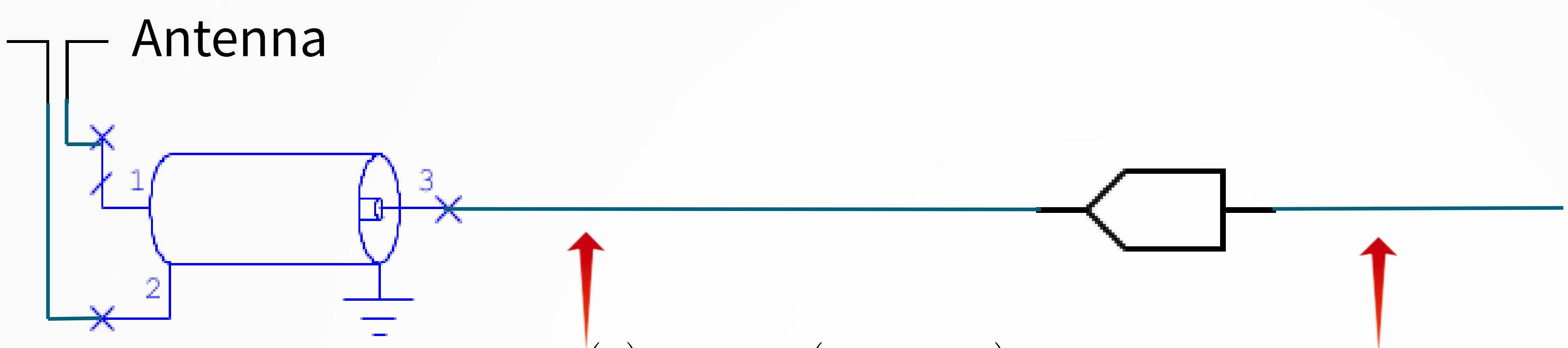
2. Euler's Formula

$$j = \sqrt{-1}$$

$$A \cdot e^{j \cdot \phi} = A \cdot [\cos(\phi) + j \cdot \sin(\phi)]$$

- Year 1748 - Mathematician Leonhard Euler published that periodic functions such as cosine and sine can be represented as complex exponentials of instantaneous phase (phasors)
- This model simplifies math related to amplitude and phase

3. Mathematical Model of Signal



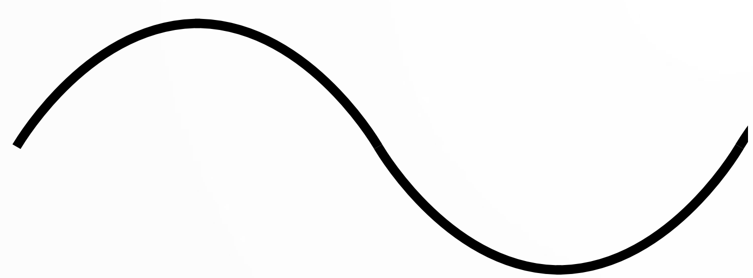
Balun

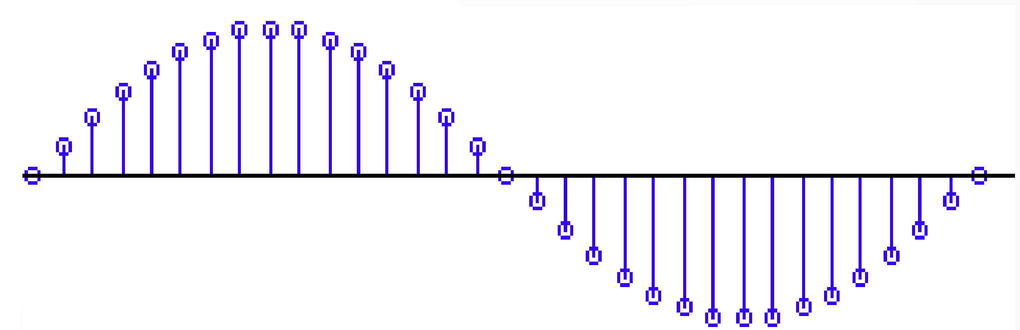
$V(t) = A \cdot \cos(2\pi f t + \phi)$

t in continuous time

$S[n] = A \cdot \cos(2\pi n T_s + \phi)$

n is integer index for discrete time

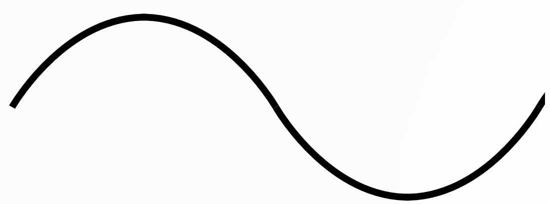


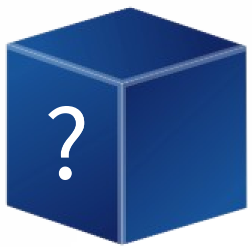


At any point in space, we can measure real-valued cosine. Convenient, but .. not ideal for DSP software that requires quadrature sampled signal.

3. Mathematical Model of Signal

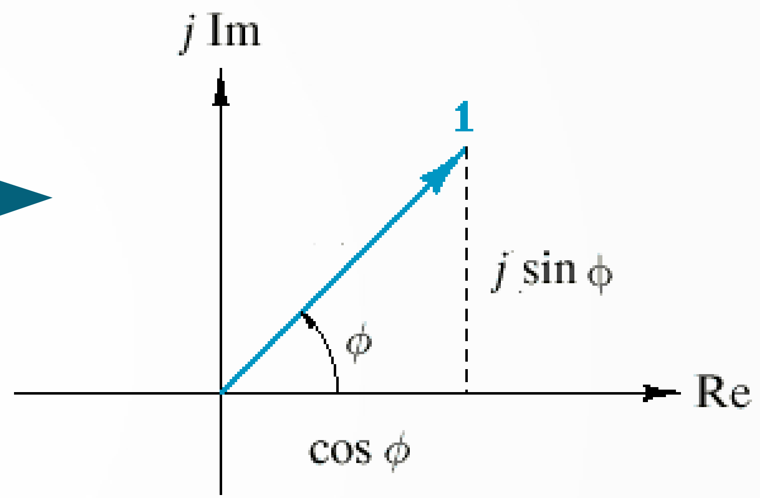
Real-value cosine





Receiver

Euler's model of complex phasor



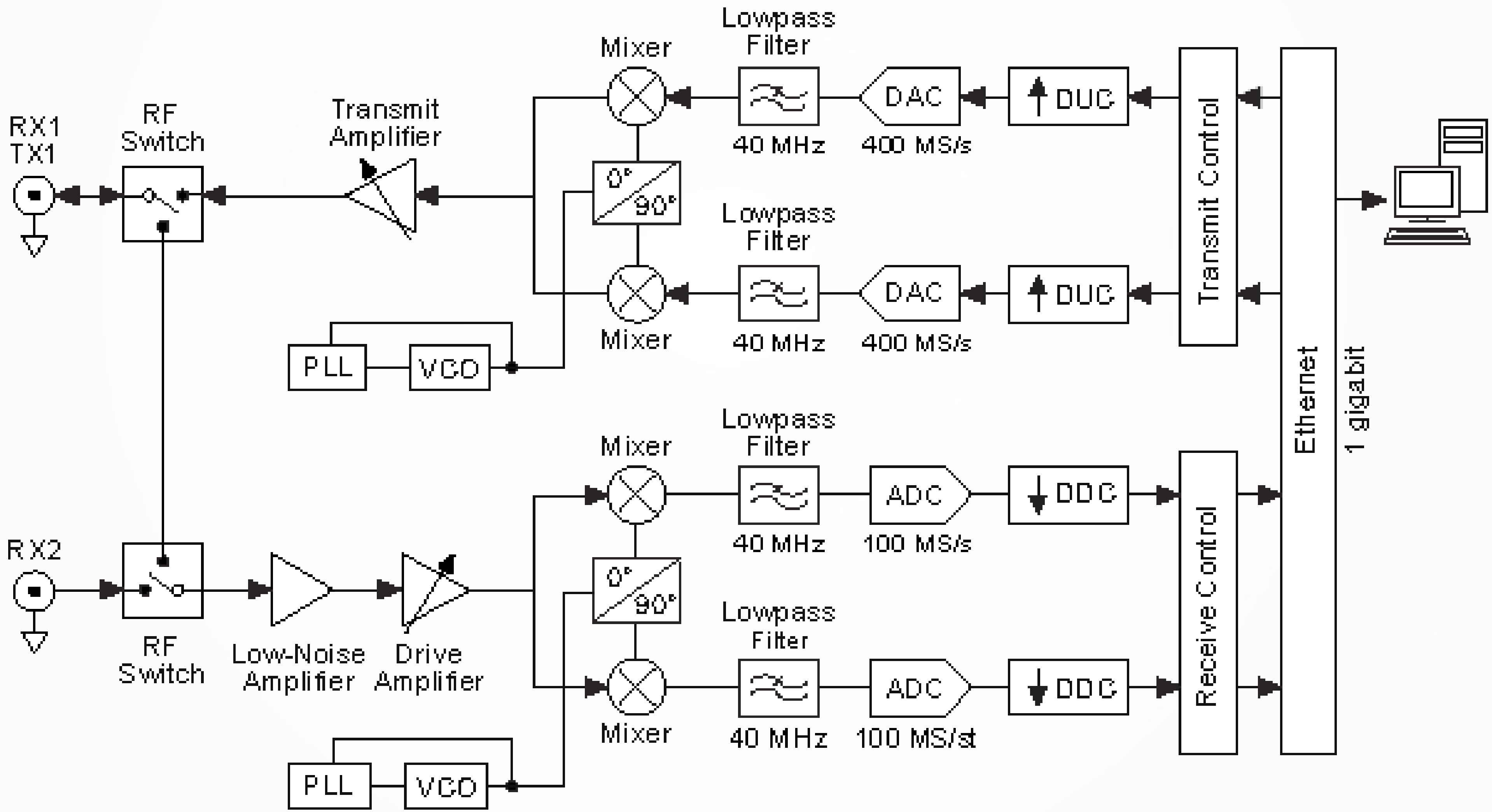
$V(t) = A \cdot \cos(2\pi f t + \phi)$

$A \cdot e^{j \cdot \phi} = A \cdot [\cos(\phi) + j \cdot \sin(\phi)]$

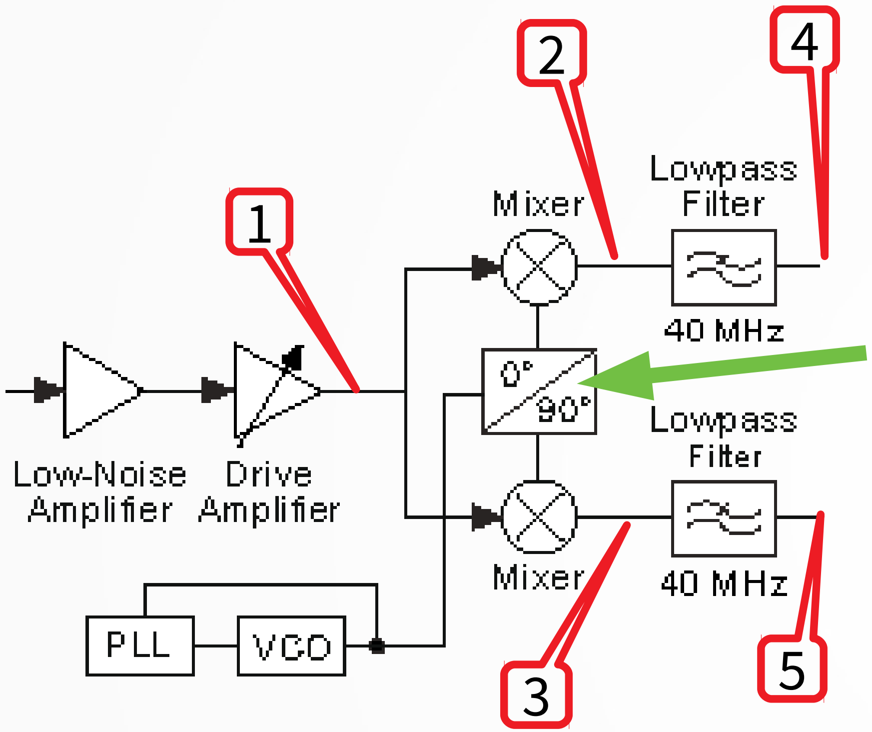
How can we translate real-value signal model (easy to understand in the physical world) to quadrature I/Q signal model (easy to process using DSP software)?

4. Hardware Implementation of Model

- Option 1- Direct Conversion Receiver (example from Ettus USRP N210 + WBX)



4. Direct Conversion Receiver



$$f_{IF} = f_{RF} - f_{LO}$$

$$V_1(t) = A \cdot \cos(2\pi f_{RF} t + \phi_0)$$

Use trig identity – product to sum

$$V_2(t) = A \cdot \cos(2\pi f_{RF} t + \phi_0) \cdot \cos(2\pi f_{LO} t) \ldots$$

$$A \cdot \cos[2\pi(f_{RF} + f_{LO})t + \phi] + A \cdot \cos[2\pi(f_{RF} - f_{LO})t + \phi]$$

Lowpass filter removes

$$V_3(t) = A \cdot \cos(2\pi f_{RF} t + \phi_0) \cdot \cos\left(2\pi f_{LO} t - \frac{\pi}{2}\right) \ldots$$

90 degree phase shift

$$A \cdot \cos(2\pi f_{RF} t + \phi) \cdot \sin(2\pi f_{LO} t) \ldots$$

$$A \cdot \cos[2\pi(f_{RF} + f_{LO})t + \phi] + A \cdot \sin[2\pi(f_{RF} - f_{LO})t + \phi]$$

Lowpass filter removes

$$V_4(t) = A \cdot \cos[2\pi(f_{RF} - f_{LO})t + \phi]$$

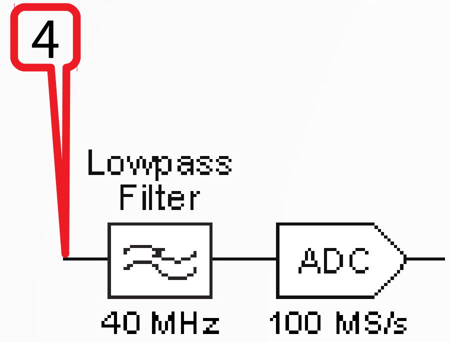
$$V_5(t) = A \cdot \sin[2\pi(f_{RF} - f_{LO})t + \phi]$$

$$V_4(t) = A \cdot \cos[2\pi f_{IF} t + \phi]$$

$$V_5(t) = A \cdot \sin[2\pi f_{IF} t + \phi]$$

5. Analog to Digital Conversion

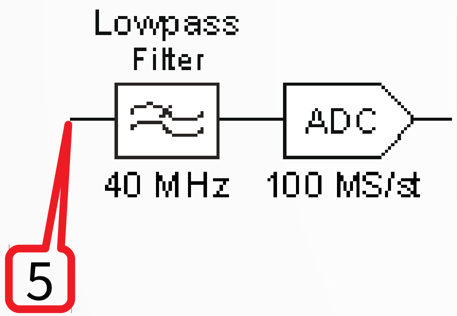
- 2 ADCs with same sample clock and same length of cable / PCB trace



Channel I

byte 0	byte 2	byte 4	byte 2*n
I[0]	I[1]	I[2]	I[n]

$$S_I[n] = A \cdot \cos(2\pi f_{IF} n T_s + \phi)$$



byte 0	byte 2	byte 4	byte 2*n
Q[0]	Q[1]	Q[2]	Q[n]

Channel Q

$$S_Q[n] = A \cdot \sin(2\pi f_{IF} n T_s + \phi)$$

Translate?

How to combine I and Q channel data into complex exponential (phasor) in software?

$$S_{iq}[n] = S_I[n] + j \cdot S_Q[n] = e^{j(2\pi f_{IF} n T_s + \phi)}$$

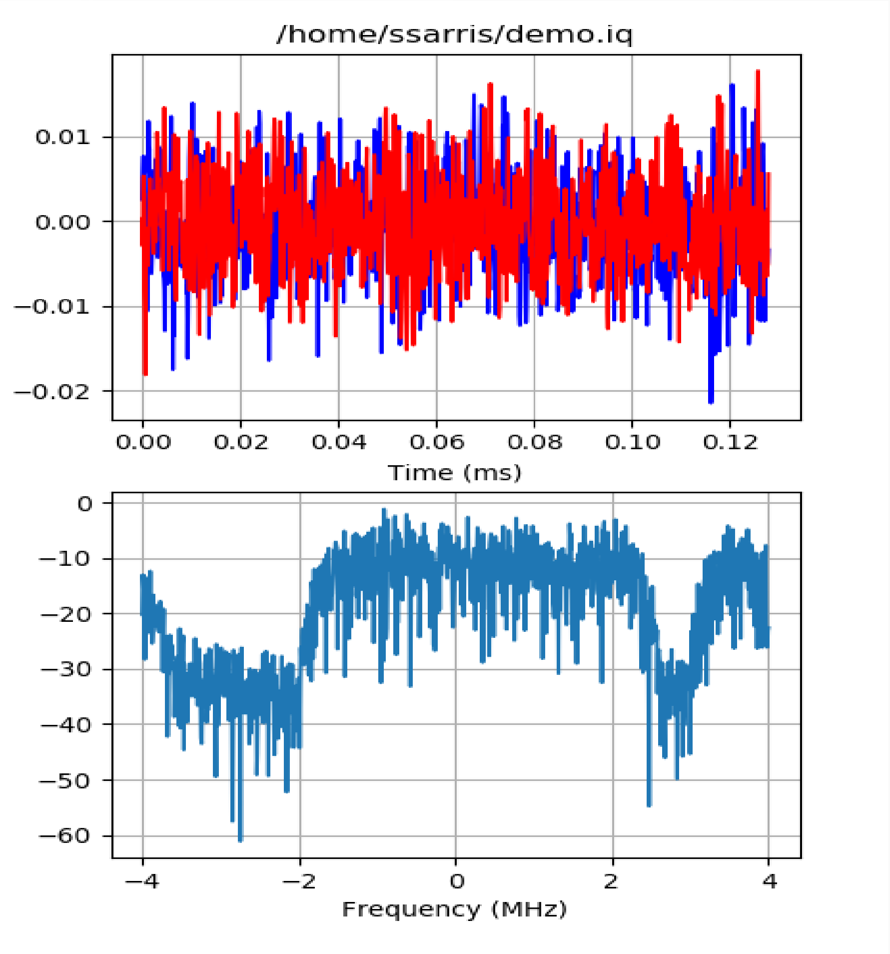
6. Convenience in Software

- Programming languages include support for complex/imaginary numbers and exponential function.
- Many libraries of software are available to process data in this format

Language	$\sqrt{-1}$	e^x
Python	`1j`	`numpy.exp(x)`
Matlab	`J, j, I, i`	`exp(x)`
C	`<complex.h> I`	`<complex.h> cexp(x)`
Octave	`J, j, I, i`	`exp(x)`

6. Convenience in Software

- `import numpy as np`
- `import matplotlib`
- `matplotlib.use("QT4Agg")`
- `import matplotlib.pyplot as plt`
-
- `fftsize = 1024`
- `nsamples = 1024`
- `fs = 8e6`
- `tt = np.arange(0,nsamples)/fs`
- `ff = np.linspace(-fs/2,fs/2,fftsize)`
- `ftest = 100`
-
- `fp = open('/home/ssarris/demo.iq','rb')`
- `fp.seek(0,0)`
- `dt = np.dtype([('i',np.float32), ('q',np.float32)])`
- `data = np.fromfile(fp,dtype=dt,count=nsamples)`
- `sigt = data['i']+1j*data['q']`
- `sigf = np.fft.fft(sigt,fftsize)`



$$S_{iq}[n]=S_I[n]+j\cdot S_Q[n]=e^{(j2\pi f_{IF}nT_s+\phi)}$$

7. Tradeoff: Real-value vs I/Q sampling

- TODO – graphic of spectrum.
- TODO – graphic of streams of bytes. Overall samples and total data rate is the same.