ECE 531: Software Defined Radio

Dr. Daniel Gallagher

Adjunct Assistant Professor, Dept. of Electrical and Computer Engineering

Lecture 5 & 6

Topics:

- Revisit Windowing, Spectral Leakage, Quantization
- Introduce PlutoSDR hardware
- Introduce Industrial Input / Output (IIO)

THE UNIVERSITY OF ARIZONA

Announcements

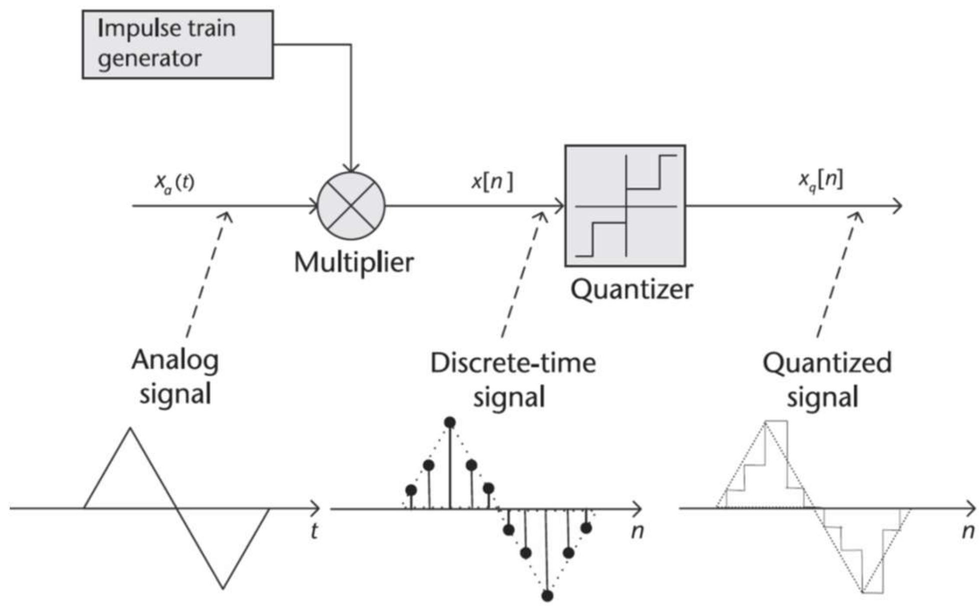
- Lab 1 posted.
  - Due Date Monday February 12th
- PlutoSDR hardware needed beginning Lab2
- Windows versions of Software and Drivers linked on D2L
  - GNU Radio
  - IIO Oscilloscope
  - libIIO
  - Pluto USB Drivers

Sampling Theory

$$x[n] = x(nT_s), \quad -\infty < n < \infty$$

$$p(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT_s), \quad k = 0, 1, 2, ...,$$

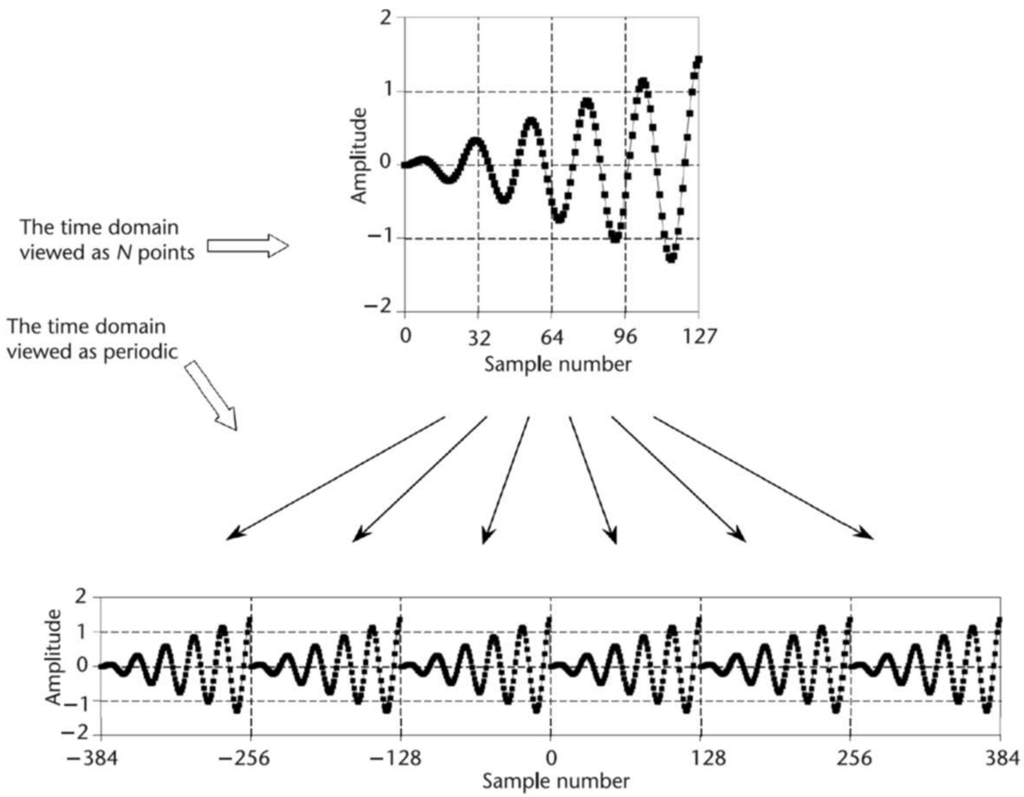
$$x_s(t) = x(t)p(t) = x(t) \sum_{k=-\infty}^{\infty} \delta(t - kT_s)$$

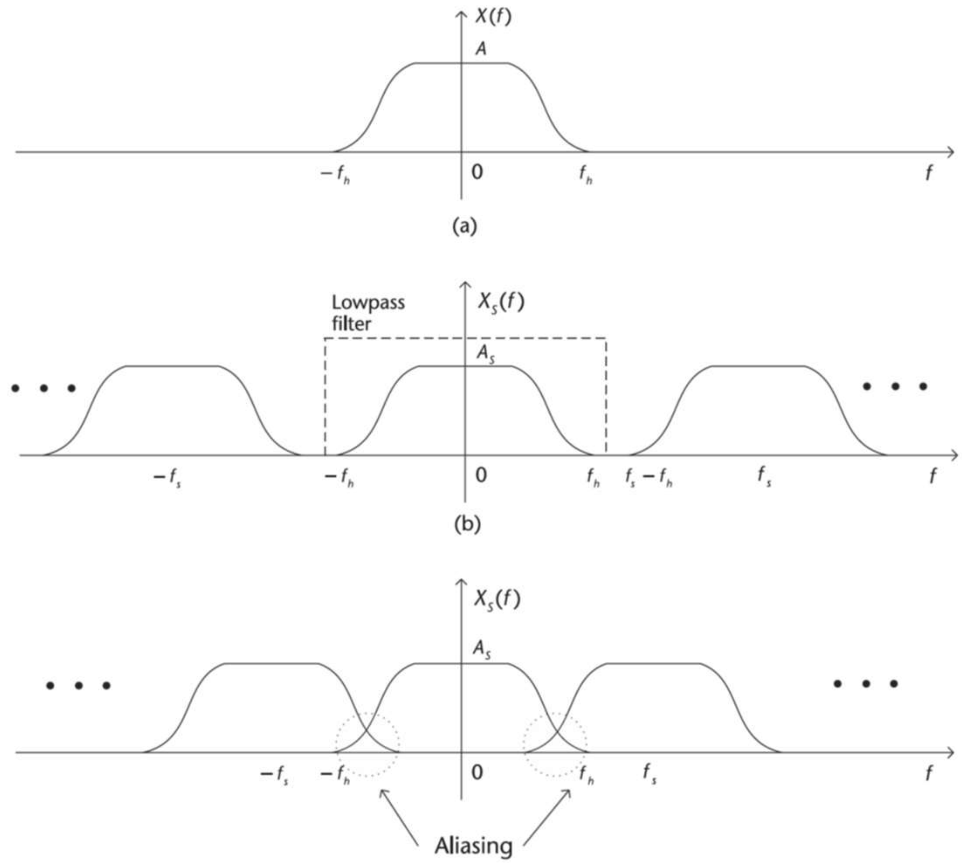


$$X_s(\omega) = \frac{1}{2\pi} X(\omega) * P(\omega),$$

$$X_s(\omega) = \frac{1}{\sqrt{2\pi} T_s} \sum_{k=-\infty}^{\infty} X(\omega - k\omega_s).$$

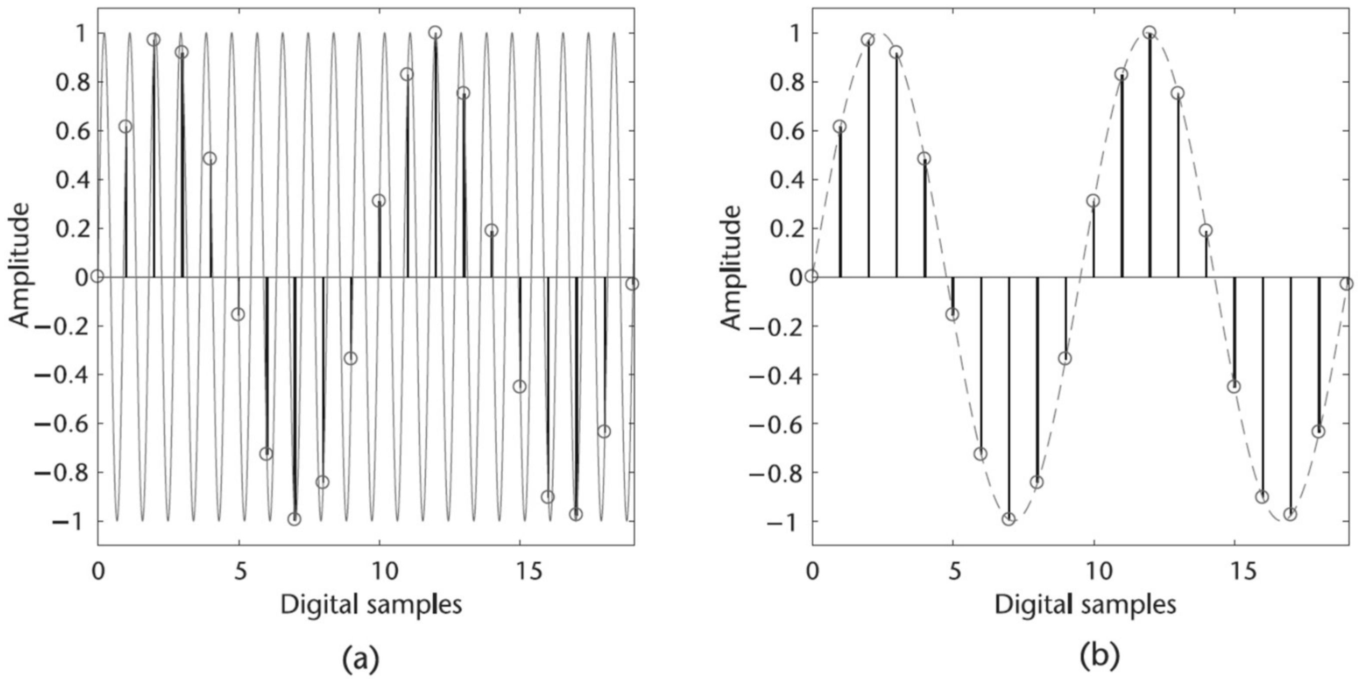
Sampling Theory Visualized





THE UNIVERSITY OF ARIZONA

Aliasing in Time Domain



THE UNIVERSITY OF ARIZONA

Sampling: Non-ideal Effects

- Aliasing
- May not ideally sample: x[n]
  - May instead sample at: x[n]*h[n]
- May have noise: y[n] = x[n] + N[n]
- Fixed point Quntization noise
  - SFDR = Spurious free dynamic range
  - See sfdr_test.m

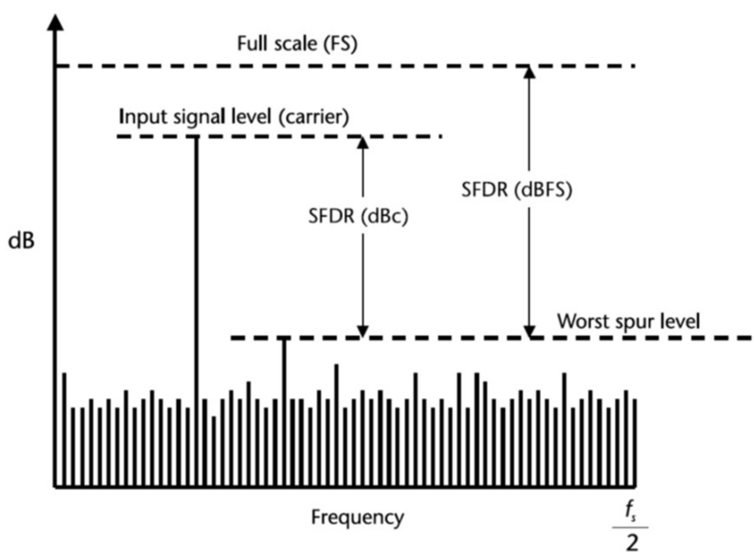


Table 2.3 Quantization: The Size of a Least Significant Bit

Resolution (N)	2^N	*Voltage (20 Vpp)*[1]	*PPM FS*	*%FS*	*dBFS*
2-bit	4	5.00 V	250,000	25	−12
4-bit	16	1.25 V	62,500	6.25	−24
6-bit	64	313 mV	15,625	1.56	−36
8-bit	256	78.1 mV	3,906	.391	−48
10-bit	1,024	19.5 mV	977	.097	−60
12-bit	4,096	4.88 mV	244	.024	−72
14-bit	16,384	1.22 mV	61.0	.0061	−84
16-bit	65,536	305 μV	15.2	.0015	−96
18-bit	262,144	76.2 μV	3.81	.00038	−108
20-bit	1,048,576	19.0 μV	.953	.000095	−120
22-bit	4,194,304	4.77 μV	.238	.000024	−132
24-bit	16,777,216	1.19 μV	.0596	.0000060	−144
26-bit	67,108,864	298 nV [1]	.0149	.0000015	−156

[1] 600 nV is the Johnson (thermal) noise in a 10-kHz BW of a 2.2 $k\Omega$ resistor at 25°C.

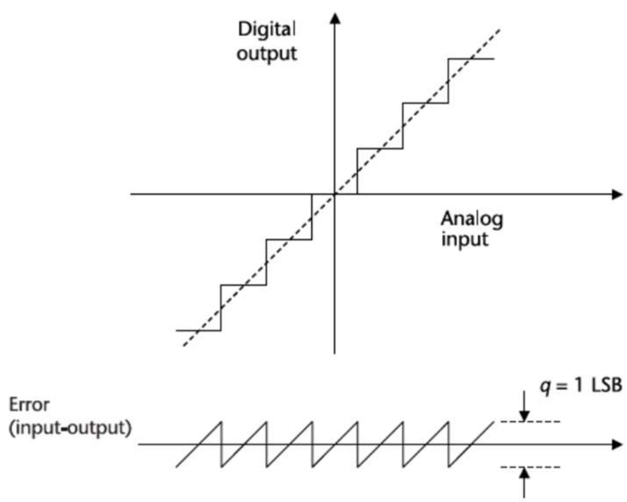
THE UNIVERSITY OF ARIZONA

ADC Quantization Error

- PlutoSDR has a 12-bit DAC and ADC
- See
  - sfdr_test.m
  - dithertest.grc

Table 2.3 Quantization: The Size of a Least Significant Bit

Resolution (N)	2^N	*Voltage (20 Vpp)*[1]	*PPM FS*	*%FS*	*dBFS*
2-bit	4	5.00 V	250,000	25	−12
4-bit	16	1.25 V	62,500	6.25	−24
6-bit	64	313 mV	15,625	1.56	−36
8-bit	256	78.1 mV	3,906	.391	−48
10-bit	1,024	19.5 mV	977	.097	−60
12-bit	4,096	4.88 mV	244	.024	−72
14-bit	16,384	1.22 mV	61.0	.0061	−84
16-bit	65,536	305 μV	15.2	.0015	−96
18-bit	262,144	76.2 μV	3.81	.00038	−108
20-bit	1,048,576	19.0 μV	.953	.000095	−120
22-bit	4,194,304	4.77 μV	.238	.000024	−132
24-bit	16,777,216	1.19 μV	.0596	.0000060	−144
26-bit	67,108,864	298 nV [1]	.0149	.0000015	−156

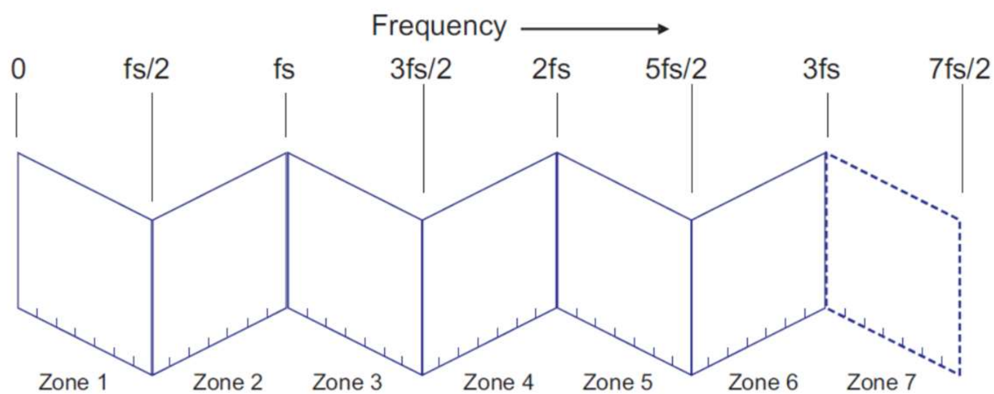


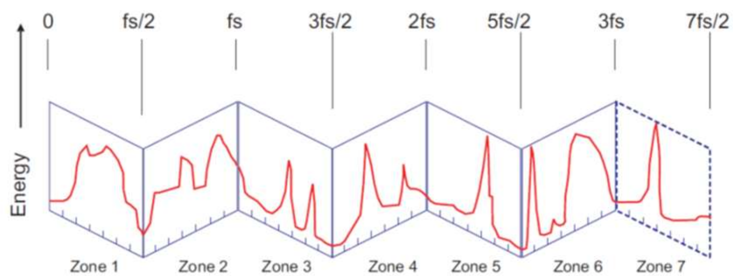
Why not just use a 24- or 26-bit ADC?

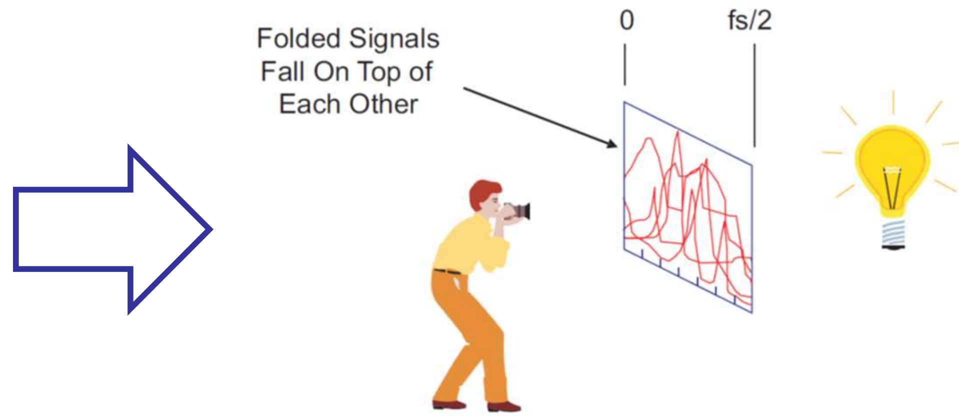
THE UNIVERSITY OF ARIZONA

“Atari-Effect”

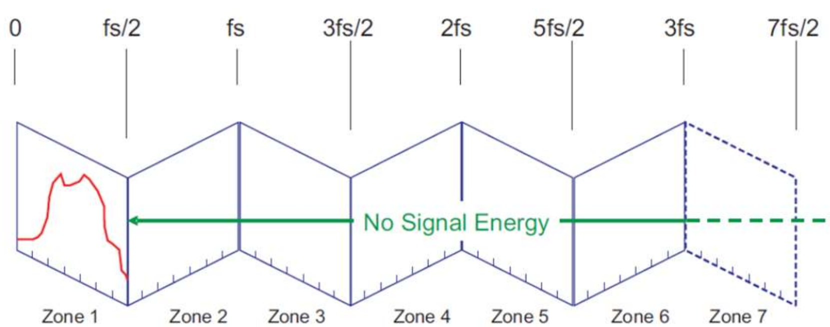
Baseband Sampling : Fan-fold Visualization



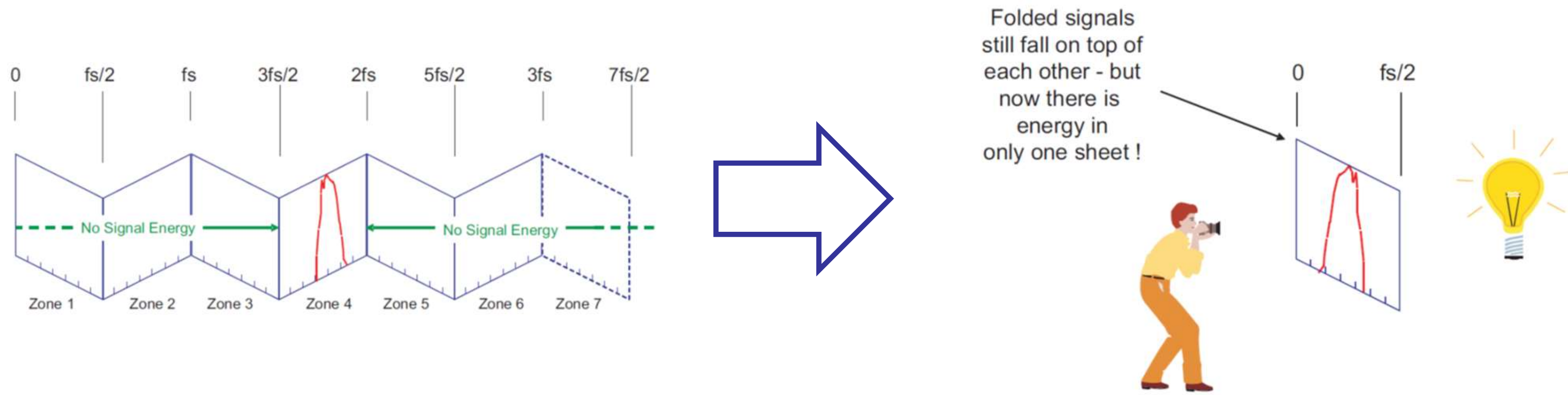




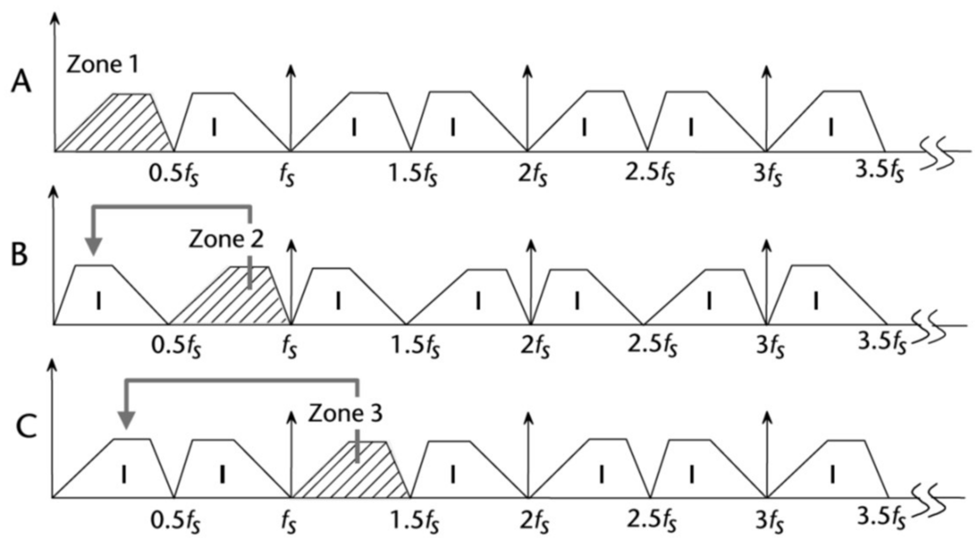
Add Low-Pass Filter



Under Sampling



Bandpass filter around Nyquist Zone of interest can allow undersampling
This can allow for lower frequency A/D converters without frequency conversion



THE UNIVERSITY OF ARIZONA

Windowing and Spectral Leakage

- Spectral leakage is caused by discontinuities in the original, noninteger number of periods in a signal and can be improved using windowing.
- Windowing reduces the amplitude of the discontinuities at the boundaries of each finite sequence acquired by the digitizer.
- No window is often called the uniform or rectangular window because there is still a windowing effect.

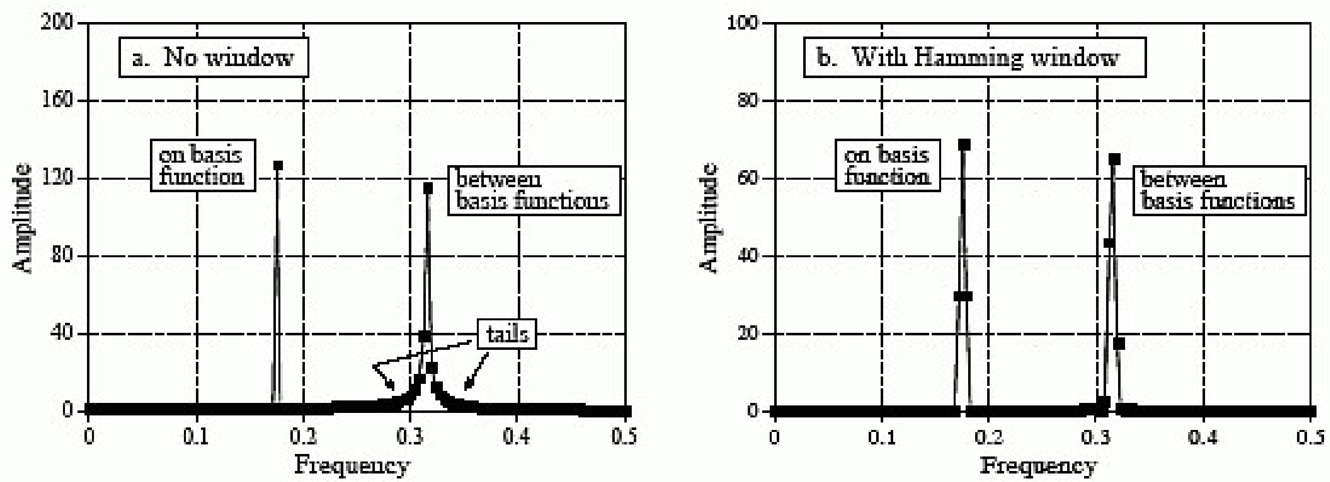
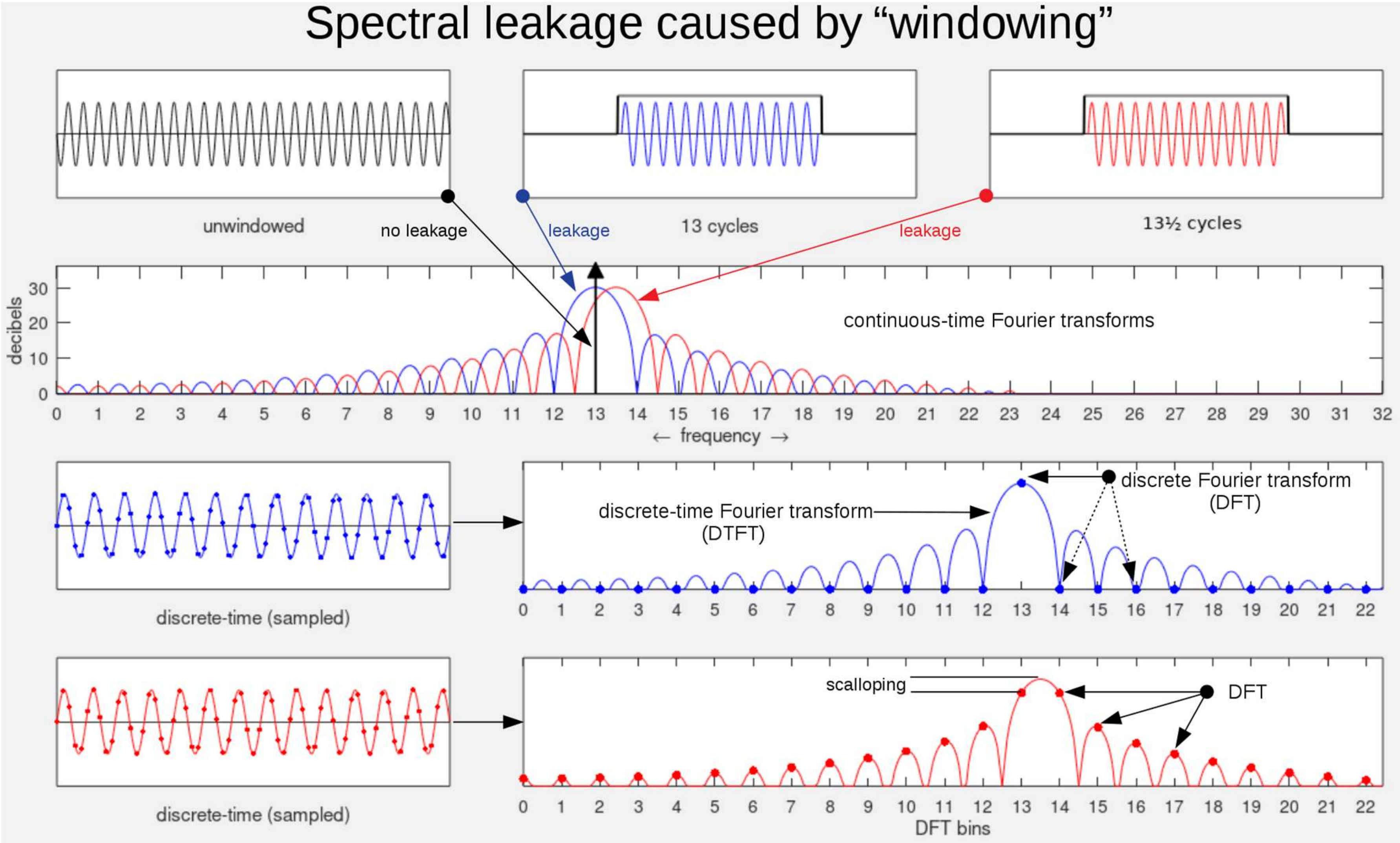


FIGURE 9-4
Example of using a window in spectral analysis. Figure (a) shows the frequency spectrum (magnitude only) of a signal consisting of two sine waves. One sine wave has a frequency exactly equal to a basis function, allowing it to be represented by a single sample. The other sine wave has a frequency *between* two of the basis functions, resulting in *tails* on the peak. Figure (b) shows the frequency spectrum of the same signal, but with a Hamming window applied before taking the DFT. The window makes the peaks look the same and reduces the tails, but broadens the peaks.

Source: The Scientist and Engineer's Guide to Digital Signal Processing By Steven W. Smith.

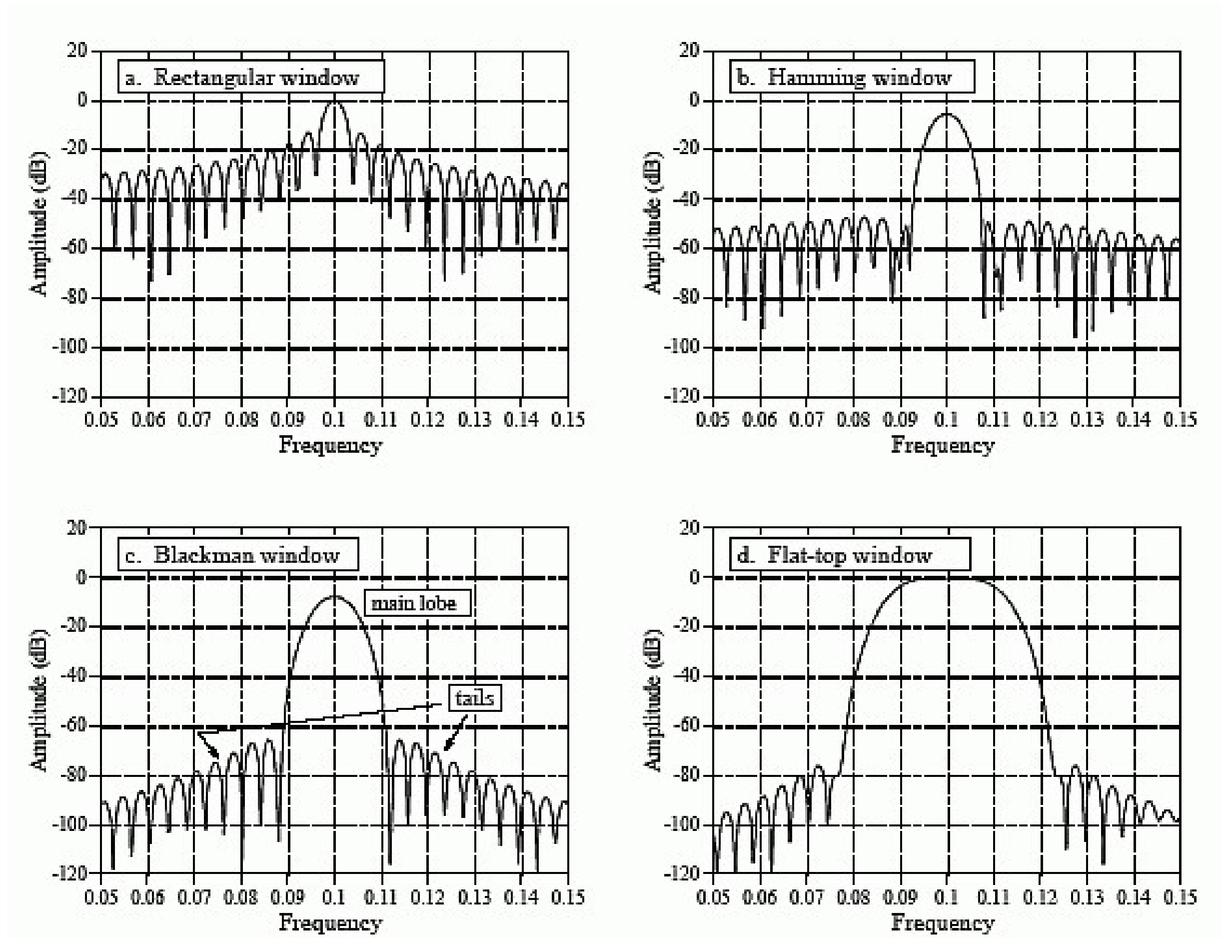
Windowing and Spectral Leakage



Interactive plots at:
https://jackschaedler.github.io/circles-sines-signals/dft_leakage.html

Window Functions (Cont)

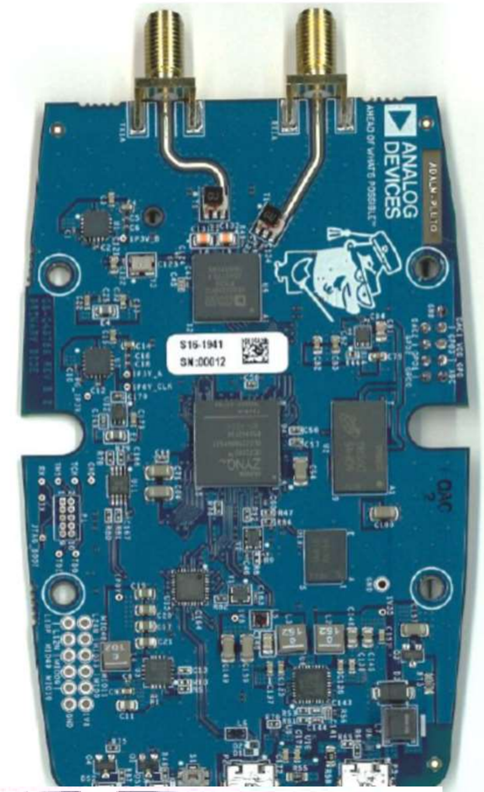
- Rect Window:
  - Narrowest main lobe but largest side lobes
- Hamming and Blackman Window
  - Lower side lobes but wider main lobe
- Flat-top Window:
  - Used when accurate measurement of peak is desired
  - Minimal scalloping loss

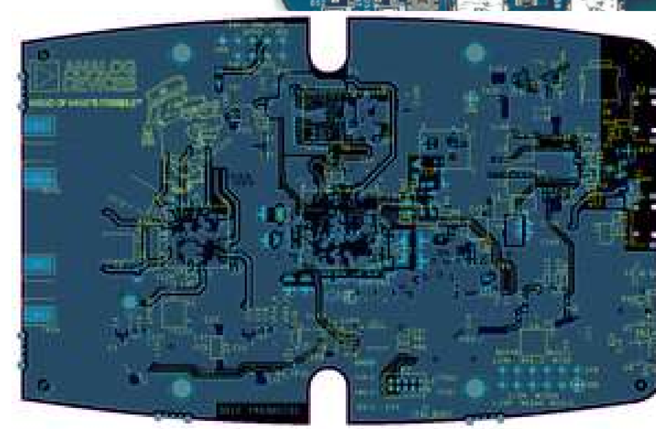


Source: The Scientist and Engineer's Guide to Digital Signal Processing By Steven W. Smith.
More on window functions at: https://en.wikipedia.org/wiki/Window_function

PlutoSDR

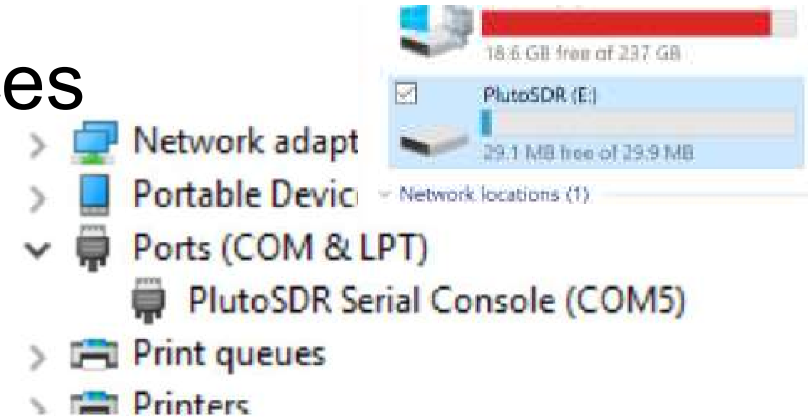
- Open Hardware
  - Schematics Available
    - https://wiki.analog.com/university/tools/pluto/hacking/hardware
  - Board Layout
    - Cadence Allegro, Gerber, BOM
  - Firmware HDL
    - https://github.com/analogdevicesinc/plutosdr-fw





THE UNIVERSITY OF ARIZONA

Connecting with PlutoSDR



- Pluto Enumerates three devices
  - Mass storage
  - Ethernet (RNDIS)
  - Serial
- Linux and Windows have support for all three options
  - MacOS requires HoRNDIS install for networking
- The automounter will look for special file names:
  - runme[0-9].sh (shell script)
  - runme[0-9] (binary file)
- Pluto will automount any USB storage devices connected to USB OTG

THE UNIVERSITY OF ARIZONA

AD936x – Analog Baseband

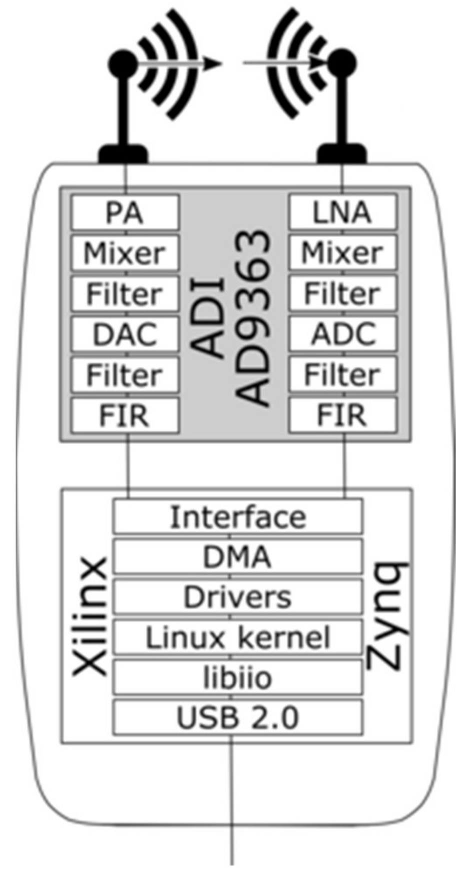
RF Transceiver	LO tuning range	Bandwidth
AD9363 (Default ADALM-PLUTO)	325 - 3800 MHz	20 MHz
AD9364	70 - 6000 MHz	56 MHz

- Full Duplex Transceiver
- Advertised Specification
  - AD9363 Transceiver
  - Tuning Range: 325 MHz to 3800 MHz
  - Up to 61.44 Mega Samples per Second (MSPS)
  - 20 MHz RF bandwidth
- Running "Out of Spec" (with firmware "hack")
  - Change of firmware device string makes firmware believe an AD9364 transceiver is available
  - Necessary to tune FM broadcasts
  - Tuning Range: 70 MHz to 6000 MHz
  - 56 MHz RF bandwidth
- MATLAB Hardware Support does this update automatically
- https://wiki.analog.com/university/tools/pluto/users/customizing#updating_to_the_ad9364

THE UNIVERSITY OF ARIZONA

ADI PlutoSDR Hardware

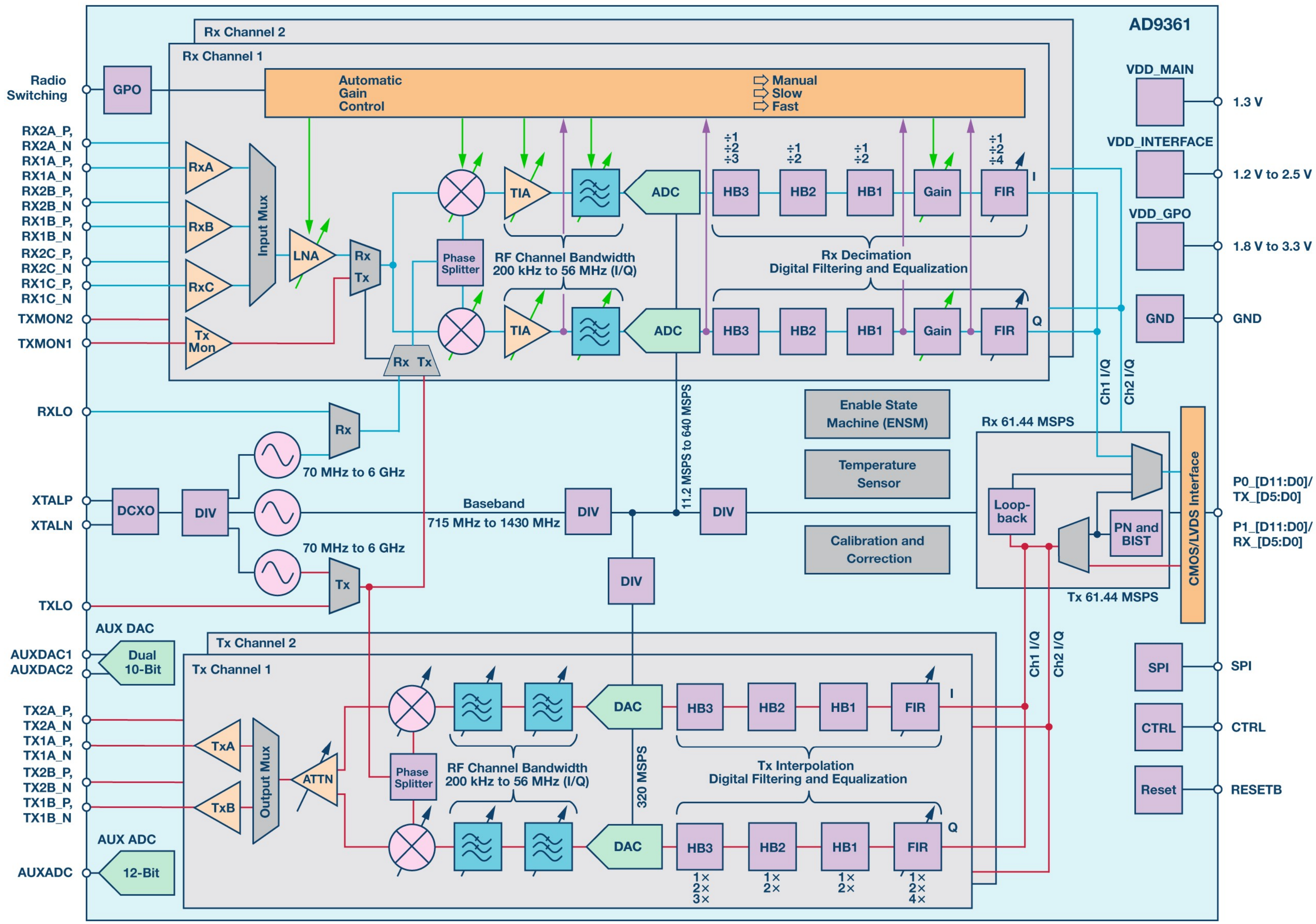
- Analog RF Section
  - Antennas, RF filters, input MUX, LNA, gain, attenuation, mixer
- Analog Baseband Section
  - Analog filters, ADC, DAC
- DSP Section
  - Fixed halfband filters
    - (decimation/interpolation)
  - 128-tap FIR filter
  - Xilinx Zynq
    - FPGA Fabric
    - Embedded ARM Processor
- I/Q data is then passed up USB to a host computer for further signal processing
  - (i.e. GNU Radio, MATLAB)



- Note: There are no preselect or output filters on the PlutoSDR
  - Can cause out-of-band interference!!

THE UNIVERSITY OF ARIZONA

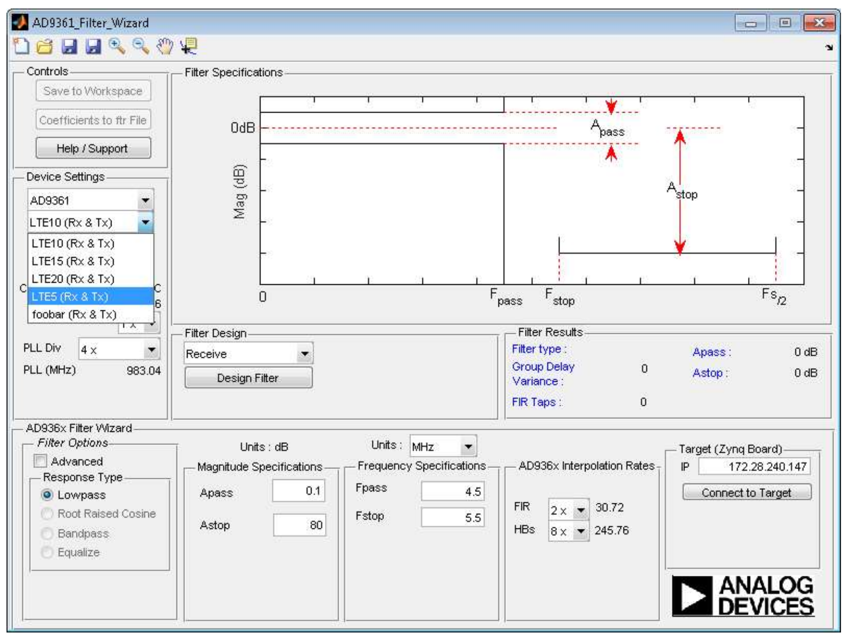
Analog Devices AD936x Transceiver



THE UNIVERSITY OF ARIZONA

AD936x Digital Filters

- Half bands, FIR and clocking determine the data rate of Pluto
- Multistage half band filters interpolate or decimate digitally for TX / RX respectively
  - Allows Sigma-Delta converters to run at a highly oversampled rate while providing 12 bit resolution
- Multiple options for loading FIR
  - Auto-filter options in gr-iio, IIO-Scope, and libad9361
  - Filters can be custom created with AD9361 filter wizard
    - https://wiki.analog.com/resources/eval/user-guides/ad-fmcomms2-ebz/software/filters



THE UNIVERSITY OF ARIZONA

Reference Clock Stability

- Oscillator Crystals are not perfect
- Inaccuracies lead to phase noise, frequency error, etc
  - Can be corrected using
    - Frequency correction algorithms
    - Different oscillator (at added cost)

- An indication of how much your crystal deviates from its nominal value is given in ppm (parts per million)

- The PlutoSDR uses the RXO3225M SMD XO
  - 40 MHz
  - < ± 25 ppm

$$df = \frac{f \times ppm}{10^6}$$

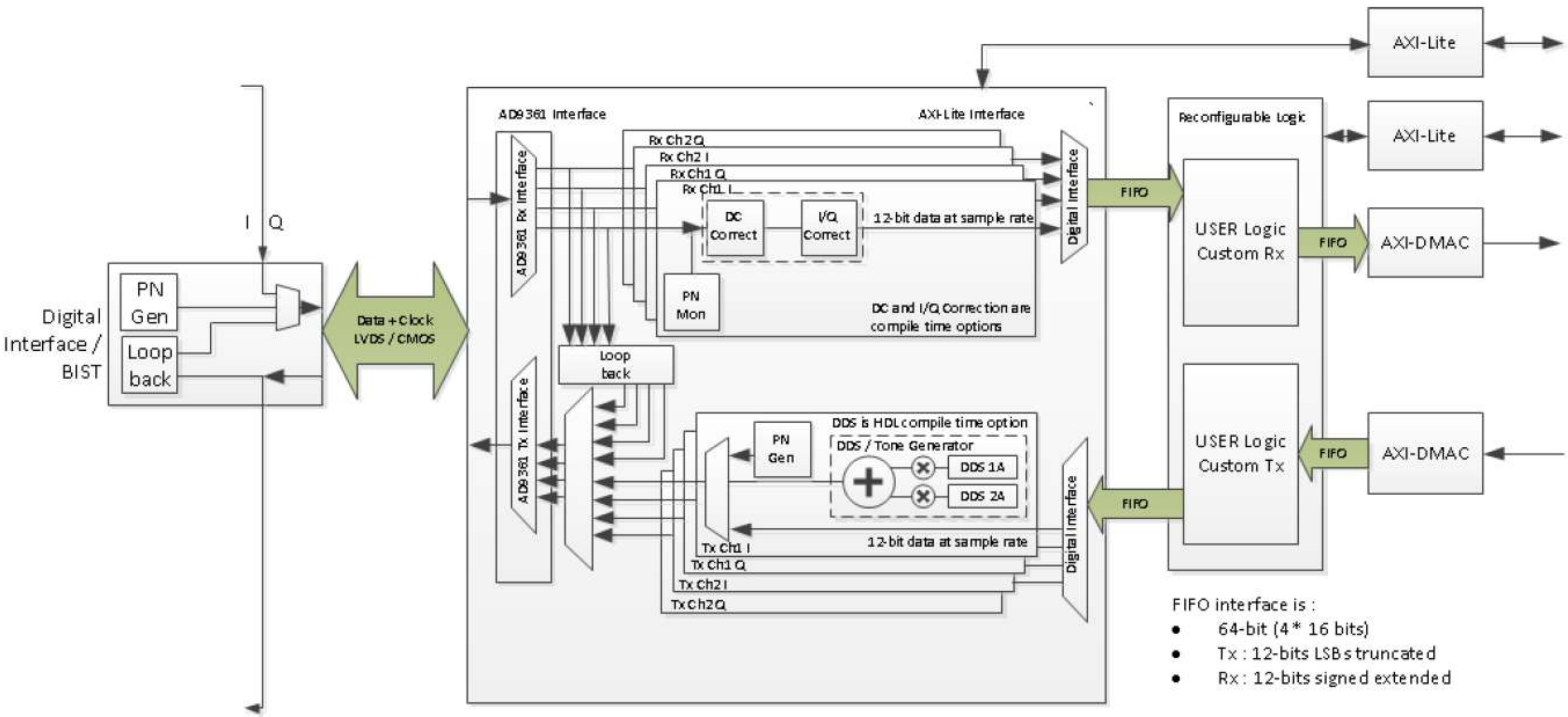
f = 40,000,000 Hz
ppm = ± 25 ppm

Variation (df) = ± 1000 Hz

$\mathbf{f_{min} = 39,999,000}$
$\mathbf{f_{max} = 40,001,000}$

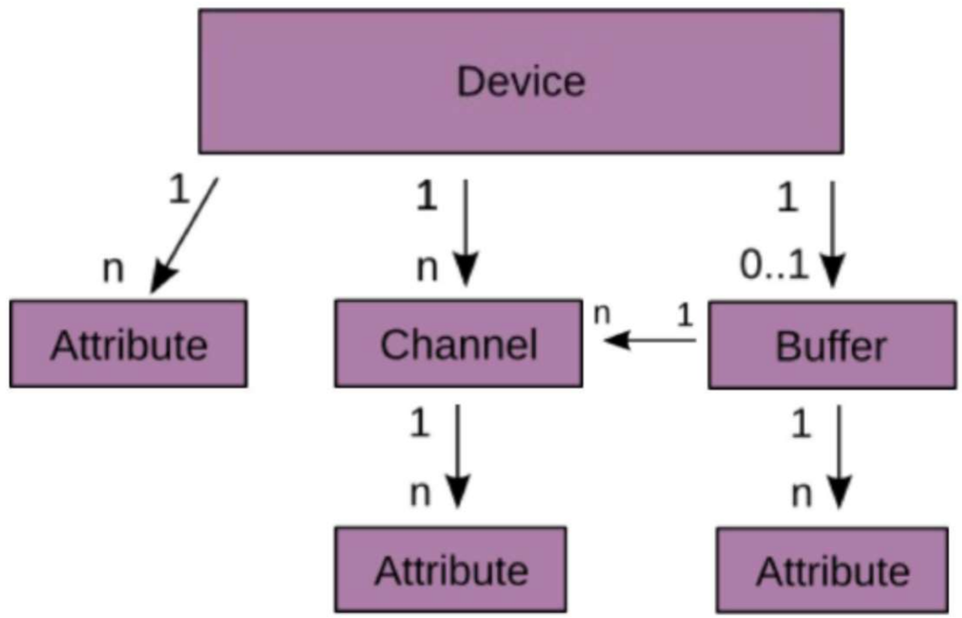
THE UNIVERSITY OF ARIZONA

HDL : Digital Loopback



Industrial Input / Output (IIO)

- IIO subsystem not unique to Pluto or SDR
  - Open source standard exists in Linux kernel
  - For PlutoSDR,
    - IIO driver runs in Linux kernel on the embedded ARM
    - libiio, library for accessing IIO devices runs on both ARM and Host
    - iiod – IIO Daemon allows remote connections to IIO clients, runs on ARM
- All device parameters are accessed through IIO and accessible through the Linux filesystem with sysfs



THE UNIVERSITY OF ARIZONA

IIO Devices

192.168.2.1 - PuTTY

```
# cd /sys/bus/iio/devices/
# ls
iio:device0  iio:device1  iio:device2  iio:device3  iio:device4
# cd iio\:device0
# cat name
adm1177
# ls
dev                  in_voltage0_raw    of_node              uevent
in_current0_raw      in_voltage0_scale  power
in_current0_scale    name               subsystem
# cat in_voltage0_raw
725
# cat in_voltage0_scale
6.433105468
# 
```

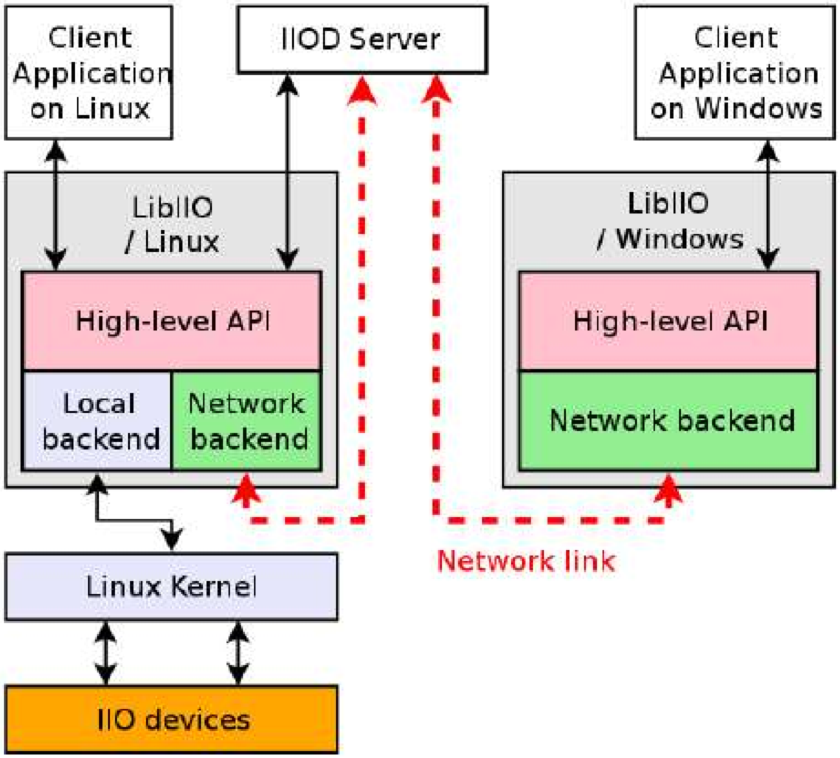

THE UNIVERSITY OF ARIZONA

IIO Buffers

192.168.2.1 - PuTTY

```
# cd /sys/bus/iio/devices/iio\:device4/
# cat name
cf-ad9361-lpc
# ls
buffer                                   in_voltage_sampling_frequency
dev                                      in_voltage_sampling_frequency_available
in_voltage0_calibbias                    name
in_voltage0_calibphase                   of_node
in_voltage0_calibscale                   power
in_voltage1_calibbias                    scan_elements
in_voltage1_calibphase                   subsystem
in_voltage1_calibscale                   uevent
in_voltage_samples_pps
# ls scan_elements/
in_voltage0_en     in_voltage0_type   in_voltage1_index
in_voltage0_index  in_voltage1_en     in_voltage1_type
# cat scan_elements/in_voltage0_type
le:S12/16>>0
# ls buffer/
data_available  enable          length          watermark
#
```

LibIIO




```c
struct iio_context *ctx;
struct iio_device *dev;
struct iio_channel *ch;

/* Error handling is missing */
ctx = iio_create_default_context();
dev = iio_context_get_device(ctx, 0);
ch = iio_device_get_channel(dev, 0);

iio_device_attr_write_longlong(dev, "sample_rate", 1000);
iio_channel_attr_write_double(ch, "scale", 0.525);
```

libIIO Command Line Tools

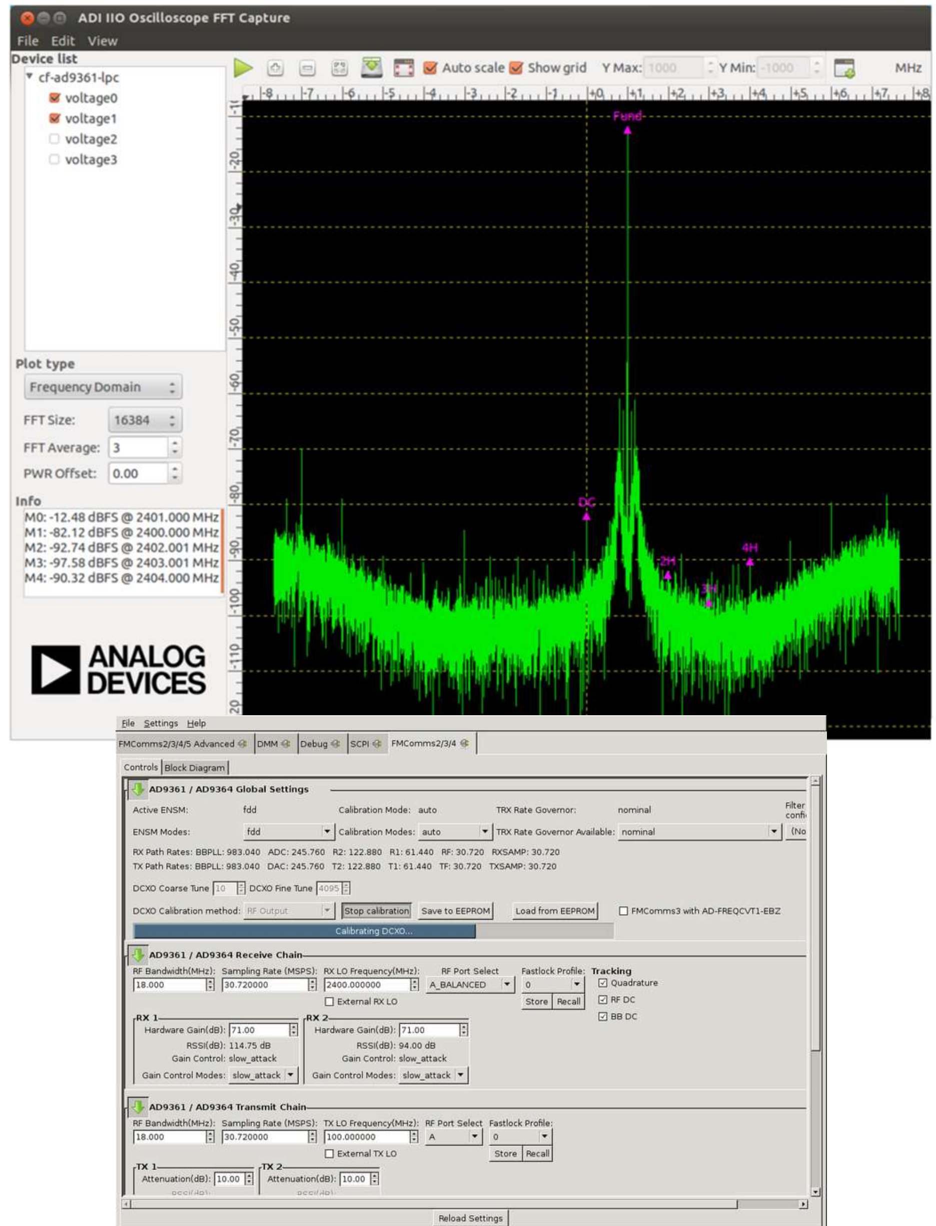
```
C:\Users\        >iio_info -s
Library version: 0.14 (git tag: 17b73d3)
Compiled with backends: xml ip usb serial
Available contexts:
        0: 0456:b673 (Analog Devices Inc. PlutoSDR (ADALM-PLUTO)), serial=10
4473ce69910006220021006dad89b3b1 [usb:2.5.5]
```

- iio_adi_xflow_check
  - Overflow/underflow testing
- **iio_attr**
  - Attribute reading and writing
- iio_genxml
  - Generate xml from context tree
- **iio_info**
  - Find devices and list attributes
- iio_readdev
  - Read from stream devices
- iio_reg
  - Read and write to registers
- iio_writedev
  - Write to stream devices

IIO Oscilloscope

- Open source C program
- Capture and display data
  - Time domain
  - Frequency domain
  - Constellation plot
- Plugins for IIO devices
  - Set device configuration
  - Read attributes
- Very useful for debugging and troubleshooting

- Included on Linux VM released to class
- Windows installer uploaded



THE UNIVERSITY OF ARIZONA

Basic Hardware Setup

- MATLAB and GNU Radio are capable of acting as an IIO client

- We will begin discussing this next class…