ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

Introduction to the ADALM-PLUTO SDR, Linux's IIO, and Open-Source Toolchains

ROBIN GETZ

TRAVIS COLLINS

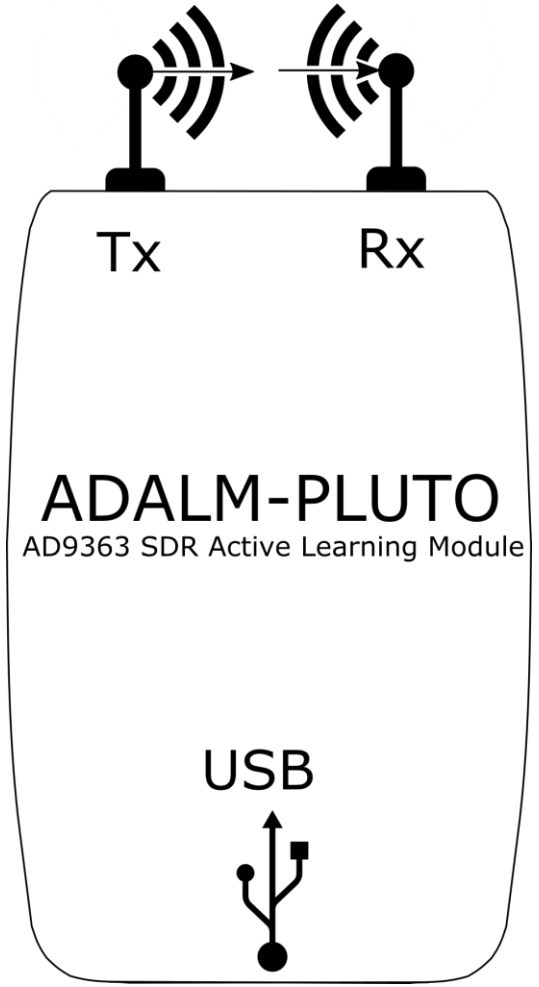
MICHAEL HENNERICH

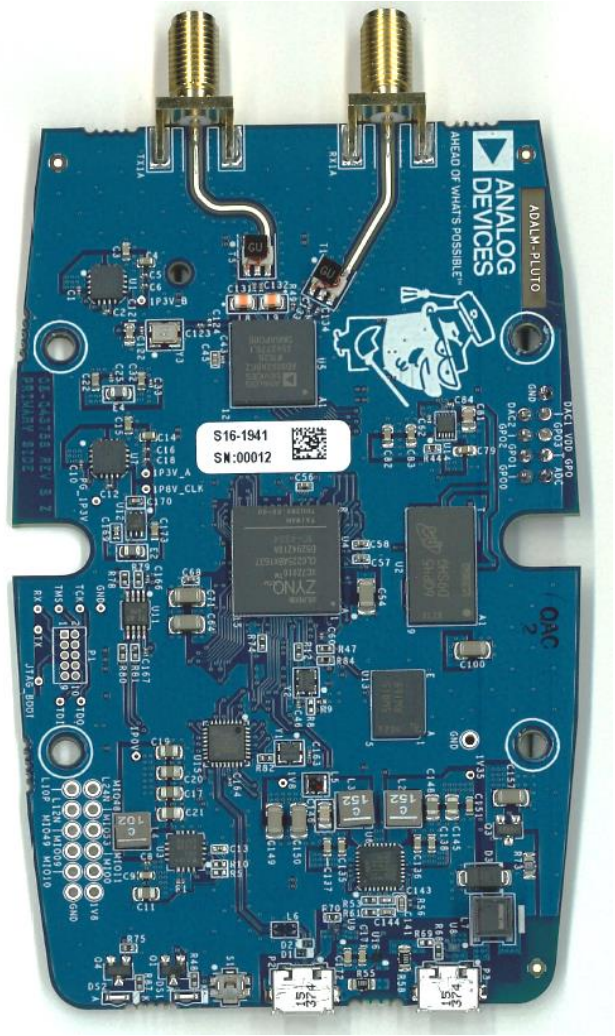
ADRIAN SUCIU

Labs

- Available at
  - https://github.com/sdrforengineers/LabGuides/

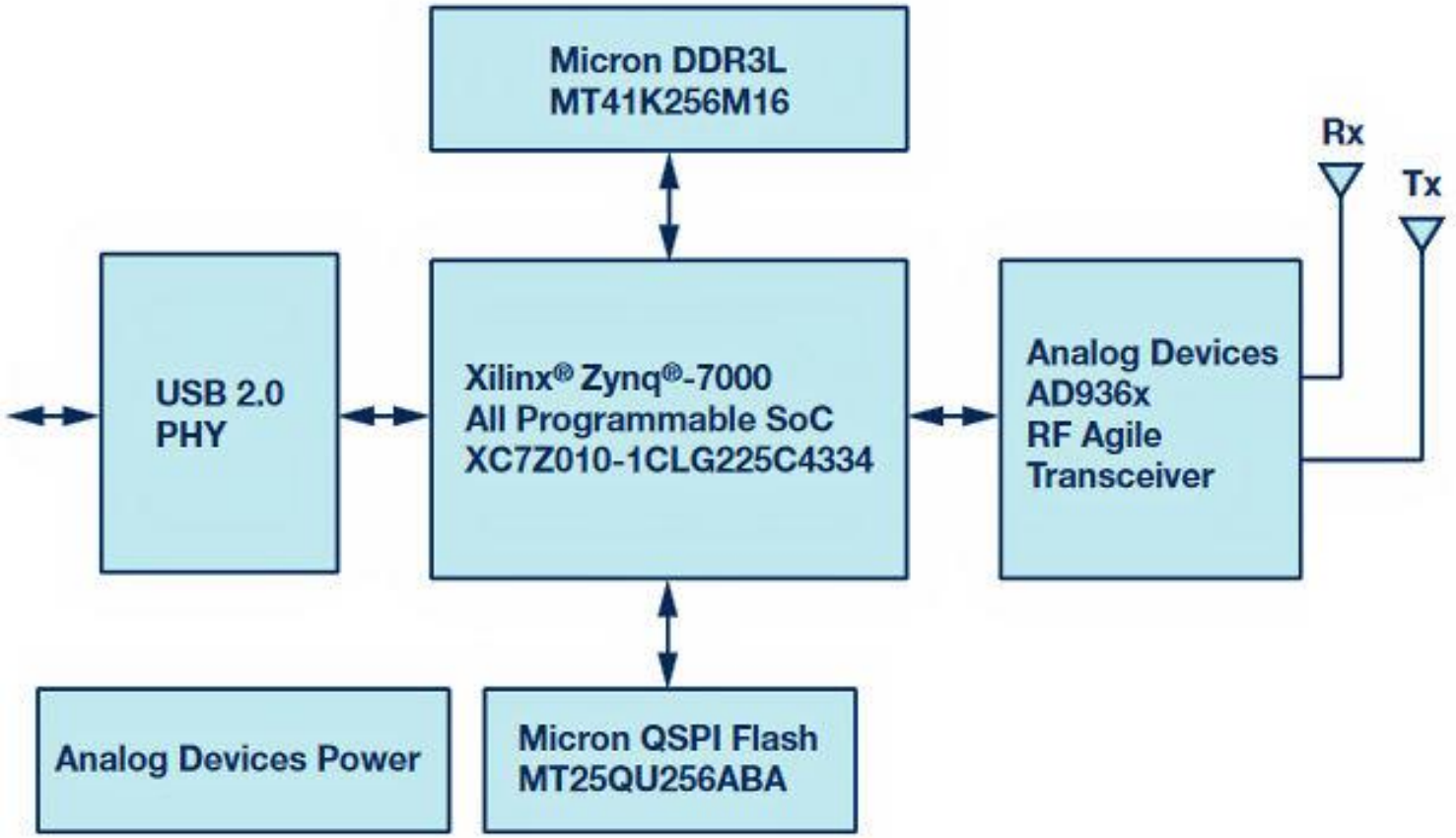
ADALM-PLUTO (PlutoSDR)





ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

PlutoSDR: Layout



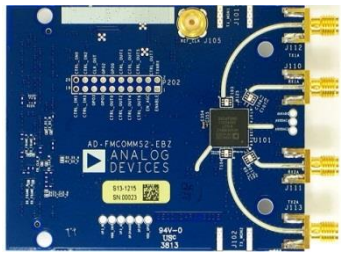
ADI (and other) General Purpose SDR Boards

For more information:
http://www.analog.com/AD-FMCOMMS2
http://www.analog.com/AD-FMCOMMS3
http://www.analog.com/AD-FMCOMMS4

Designed, manufactured, tested and sold by ADI

AD-FMCOMMS2

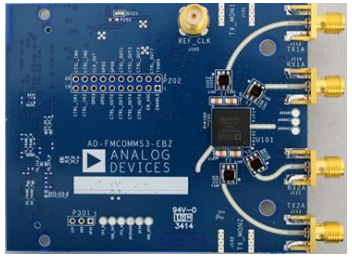
- AD9361 Integrated
- 2 x Rx, 2 x Tx
- *2.2 GHz – 2.6GHz tuning range*
- 200kHz - 56 MHz channel bandwidth
- Shipping Now!



Power, Transceiver Projects for :
- ZedBoard,
- Xilinx ZC702 & ZC706

AD-FMCOMMS3

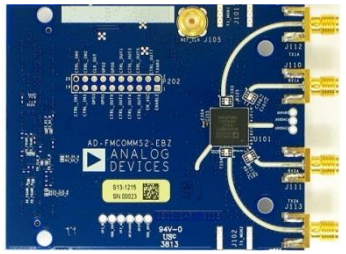
- AD9361 Integrated
- 2 x Rx, 2 x Tx
- *70 MHz – 6GHz tuning range*
- 200kHz - 56 MHz channel bandwidth
- Shipping Now!



Power, Transceiver Projects for :
- ZedBoard,
- Xilinx ZC702 & ZC706

AD-FMCOMMS4

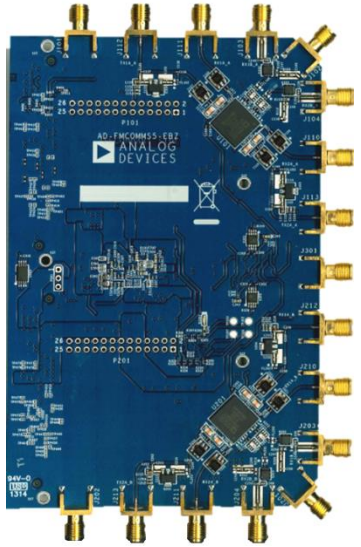
- *AD9364 Integrated*
- *1 x Rx, 1 x Tx*
- 70 MHz – 6GHz tuning range
- 200kHz - 56 MHz channel bandwidth
- Shipping Now



Power, Transceiver Projects for :
- ZedBoard,
- Xilinx ZC702 & ZC706

AD-FMCOMMS5

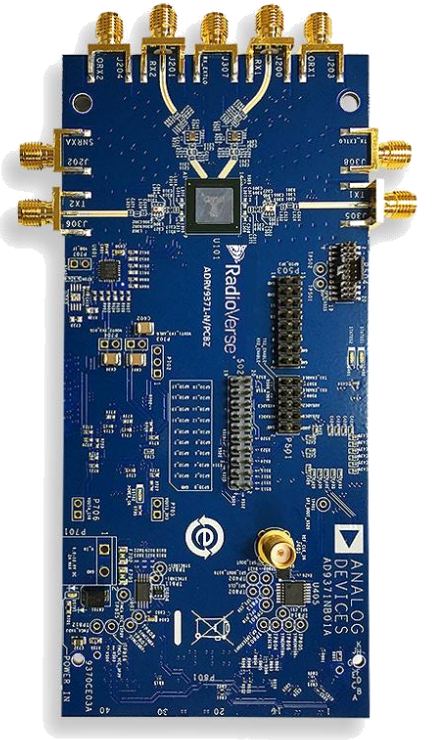
- *2 x AD9361 Integrated*
- *4 x Rx, 4 x Tx*
- *Synchronized RF*
- 70 MHz – 6GHz tuning range
- 200kHz - 56 MHz channel bandwidth
- Shipping Now!



Power, Transceiver, PLL, LNA
Projects for ZC706 only

ADRV9371-[NW]

- *1 x AD9371 Integrated*
- *2 x Rx, 2 x Tx, 2 x Obs, 1 x Sniffer*
- 300 MHz – 6GHz tuning range
- Shipping Now!



Power, Transceiver, PLL, LNA
Projects for Xilinx and Intel/Altera

ARRADIO

- AD9361 Integrated
- 2 x Rx, 2 x Tx
- *2.2 GHz – 2.6GHz tuning range*
- 200kHz - 56 MHz channel bandwidth
- Shipping Now!

Power, Transceiver Projects for :
Altera SOCKIT

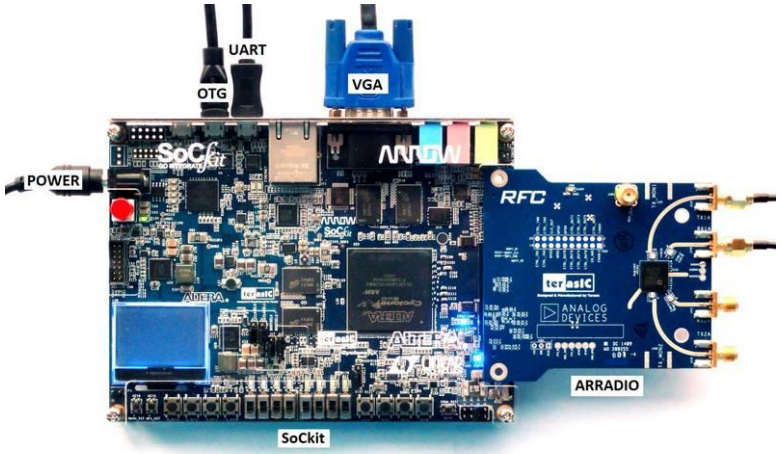
Assistance and some qualification done by ADI

ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

Evaluating the AD9361

- Hardware
  - AD-FMCOMMS2-EBZ (AD9361)
    - Narrow RF Tuning Range
  - AD-FMCOMMS3-EBZ (AD9361)
    - Wide RF Tuning Range
  - AD-FMCOMMS4-EBZ (AD9364)
  - Works with:
    - Xilinx Zynq based solutions, including ZC706 (shown), ZC702, and Zedboard
    - Xilinx Kintex (KC705) and Virtex (VC707)
- Software
  - Device drivers
    - Linux and/or No-OS
  - FPGA HDL
  - IIO scope
    - Data visualization application
    - Graphical configuration application

Xilinx ZC706 + FMCOMMS2



Arrow SoCKit + ARRADIO

ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

RFSOM

RFSOM SD2 (7035 / AD9361) : $1095

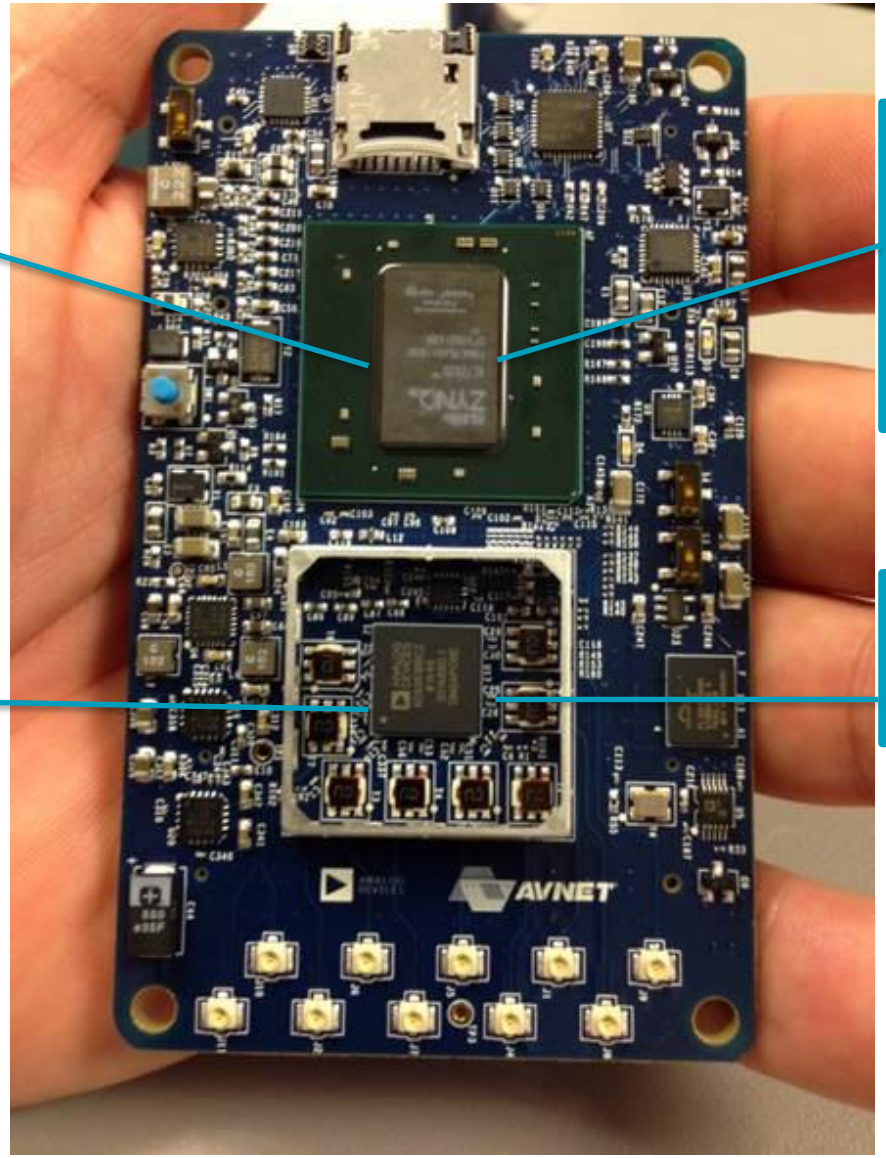
- Processor
  - Dual Core ARM Cortex A9 (800MHz each)
  - L1 cache : 32 KB Instruction, 32 KB Data
  - L2 cache : 512 KB
- FPGA
  - Kintex-7 Fabric
  - 275K Logic Cells
  - ~4.1M Asic gates
  - 17.6 Mb Block RAM
  - 900 DSP Slices

Xilinx 7035

- 1 Gbyte DDR3L
- 32 Mbyte SPI Flash
- MicroSD Card (lockable)
- Radio
  - 200kHz - 56 MHz RF bandwidth
  - 128-tap FIR Filters for equalization
  - 70 – 6000 MHz tuning range
  - 2 Rx, 2 Tx, 2 Tx Monitor

ADI AD9361

- 10/100/1000 Ethernet (MAC + Phy)
- USB 2.0 (OTG Controller + Phy)
- Low power
  - sub 5W at full data rates
- Full Linux based reference design
- Fully integrated and tested system
- # Pins of I/O



RFSOM SDR1 (7020 / AD9364) : $549

- Processor
  - Dual Core ARM Cortex A9 (800MHz each)
  - L1 cache : 32 KB Instruction, 32 KB Data
  - L2 cache : 512 KB
- FPGA
  - Artix-7 Fabric
  - 85K Logic Cells
  - ~1.3M Asic gates
  - 4.9 Mb Block RAM
  - 220 DSP Slices

Xilinx 7020

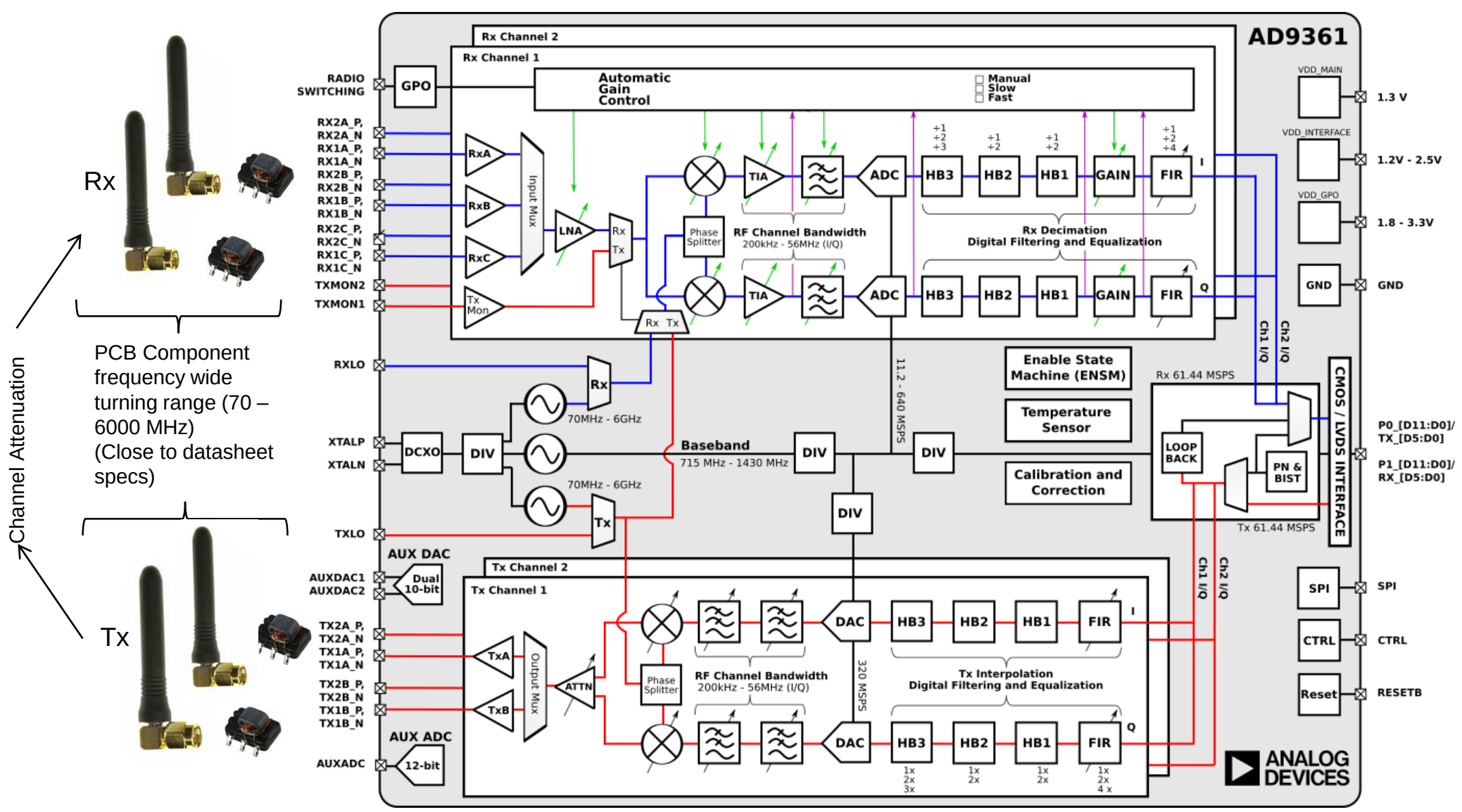
- 1 Gbyte DDR3L
- 32 Mbyte SPI Flash
- MicroSD Card (lockable)
- Radio
  - 200kHz - 56 MHz RF bandwidth
  - 128-tap FIR Filters for equalization
  - 70 – 6000 MHz tuning range
  - 1 Rx, 1 Tx, 1 Tx Monitor

ADI AD9364

- 10/100/1000 Ethernet (MAC + Phy)
- USB 2.0 (OTG Controller + Phy)
- Low power
  - sub 5W at full data rates
- Full Linux based reference design
- Fully integrated and tested system
- # Pins of I/O

ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

AD9361: Transceiver



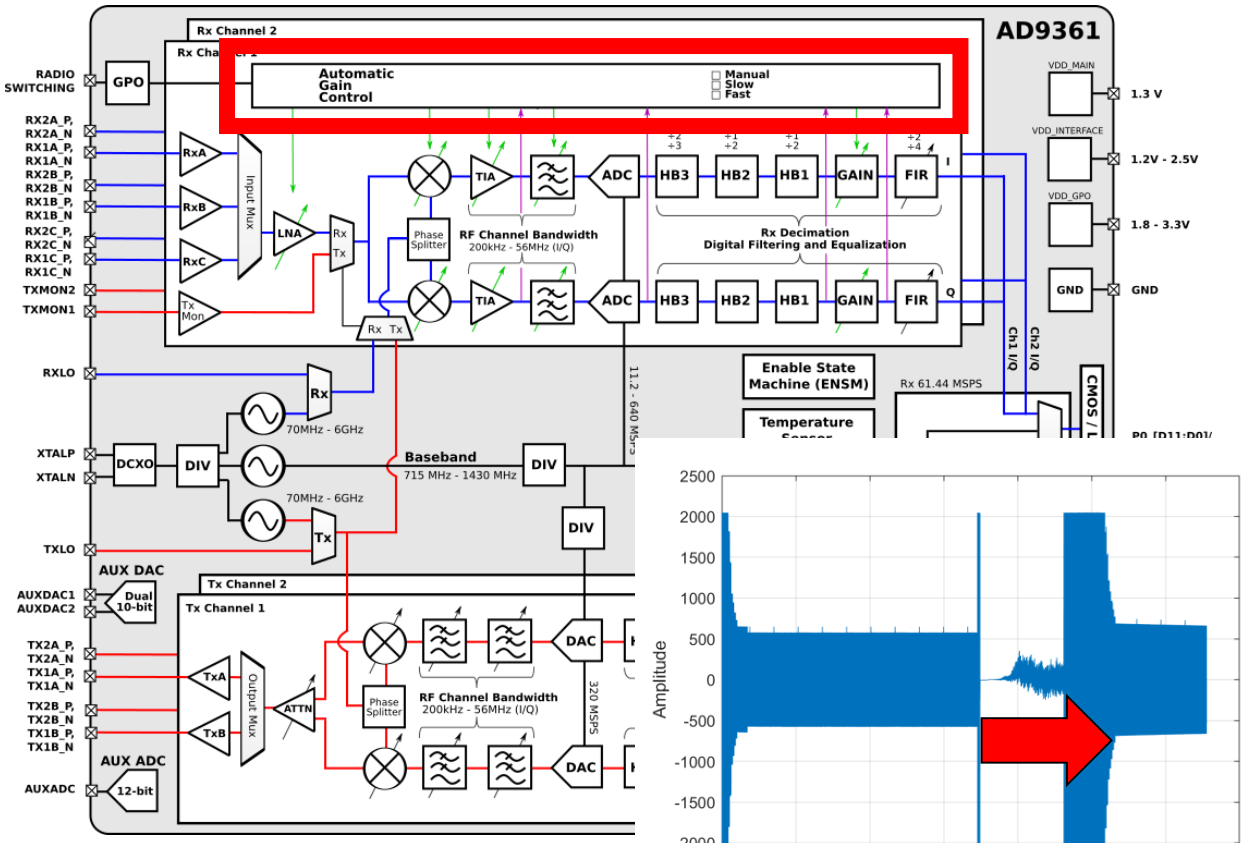
ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

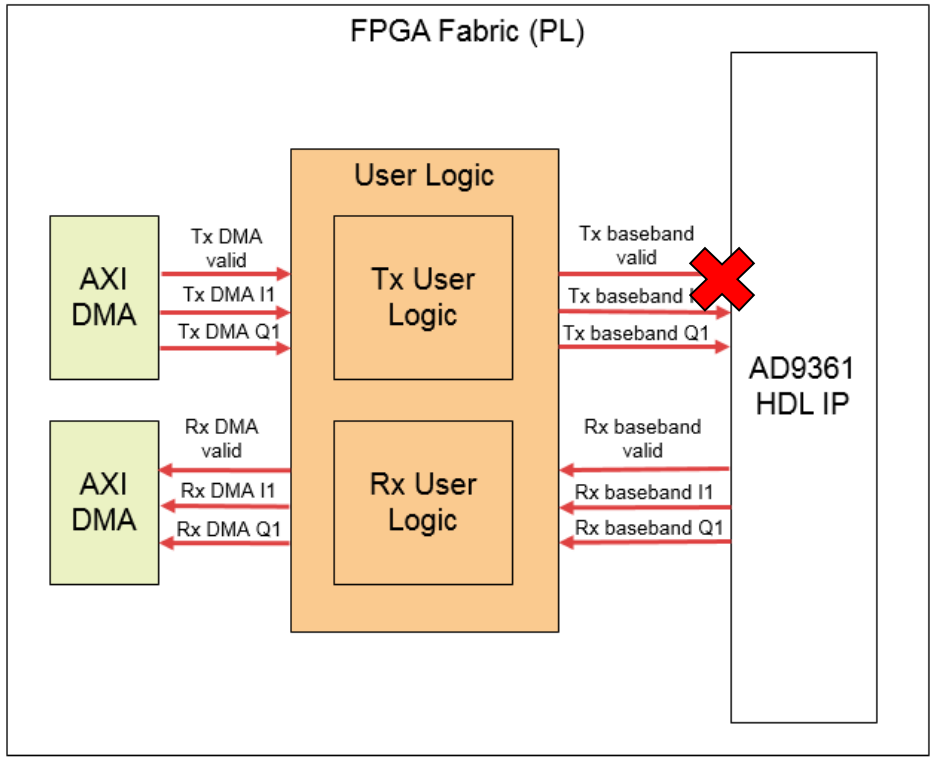
Demo: CAUTION! No Front-End Filters….

Using the AD936X

Changing AGC



Constant TX

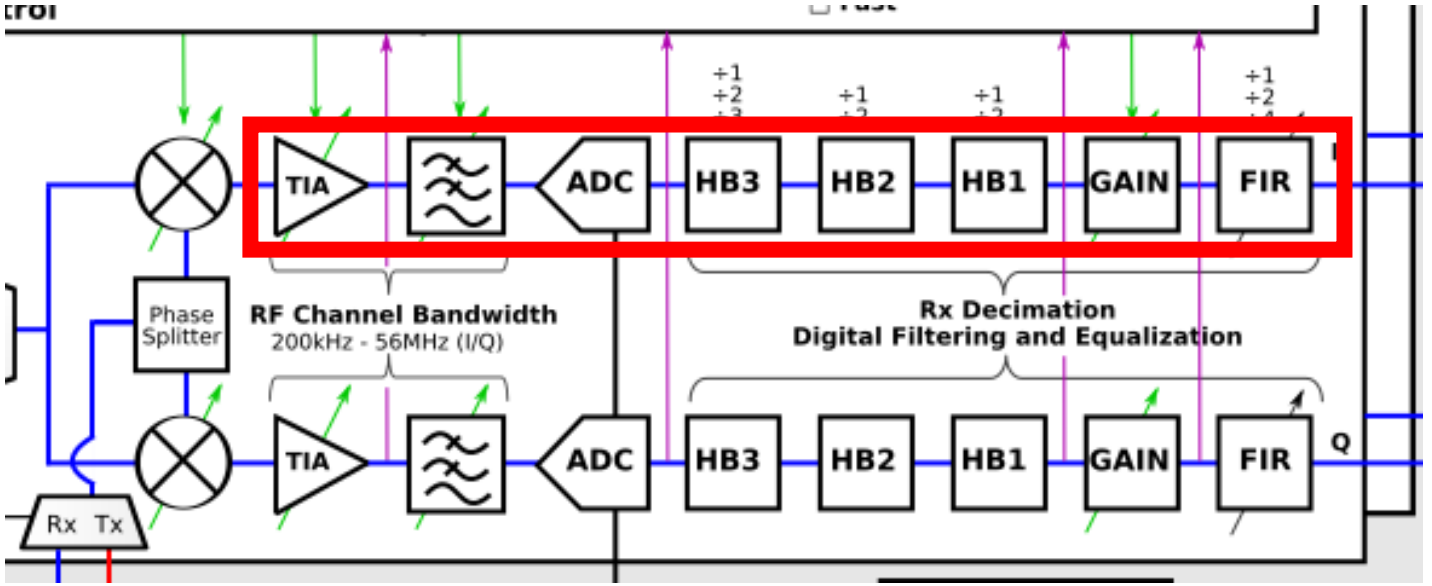


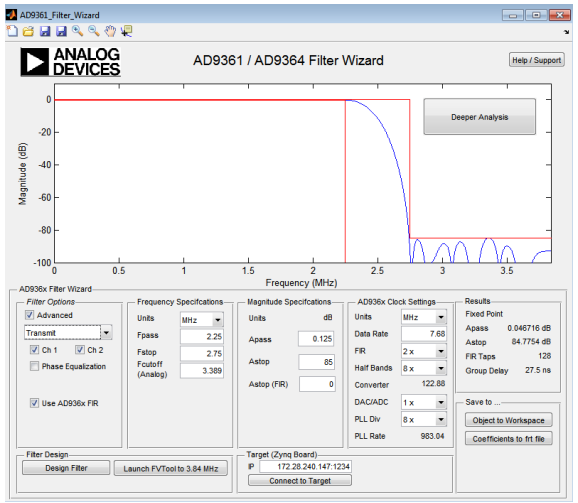
- FIRMWARE VERSION

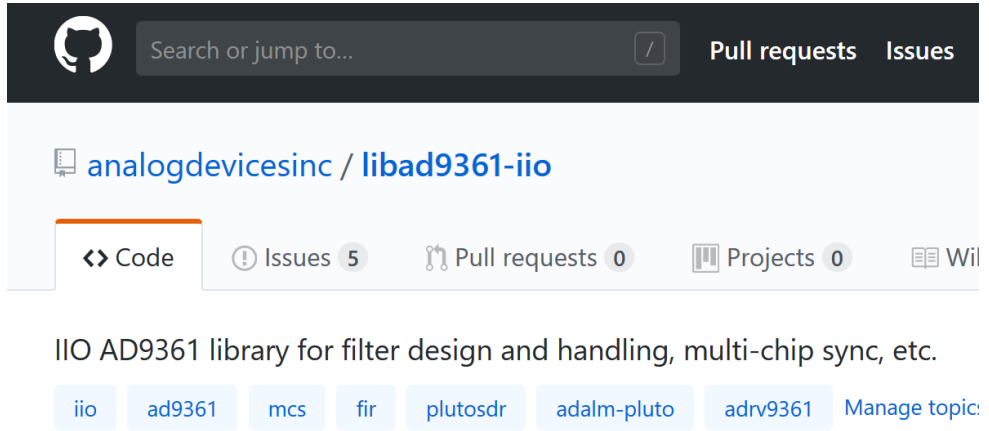
ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

The Data Path and Filters

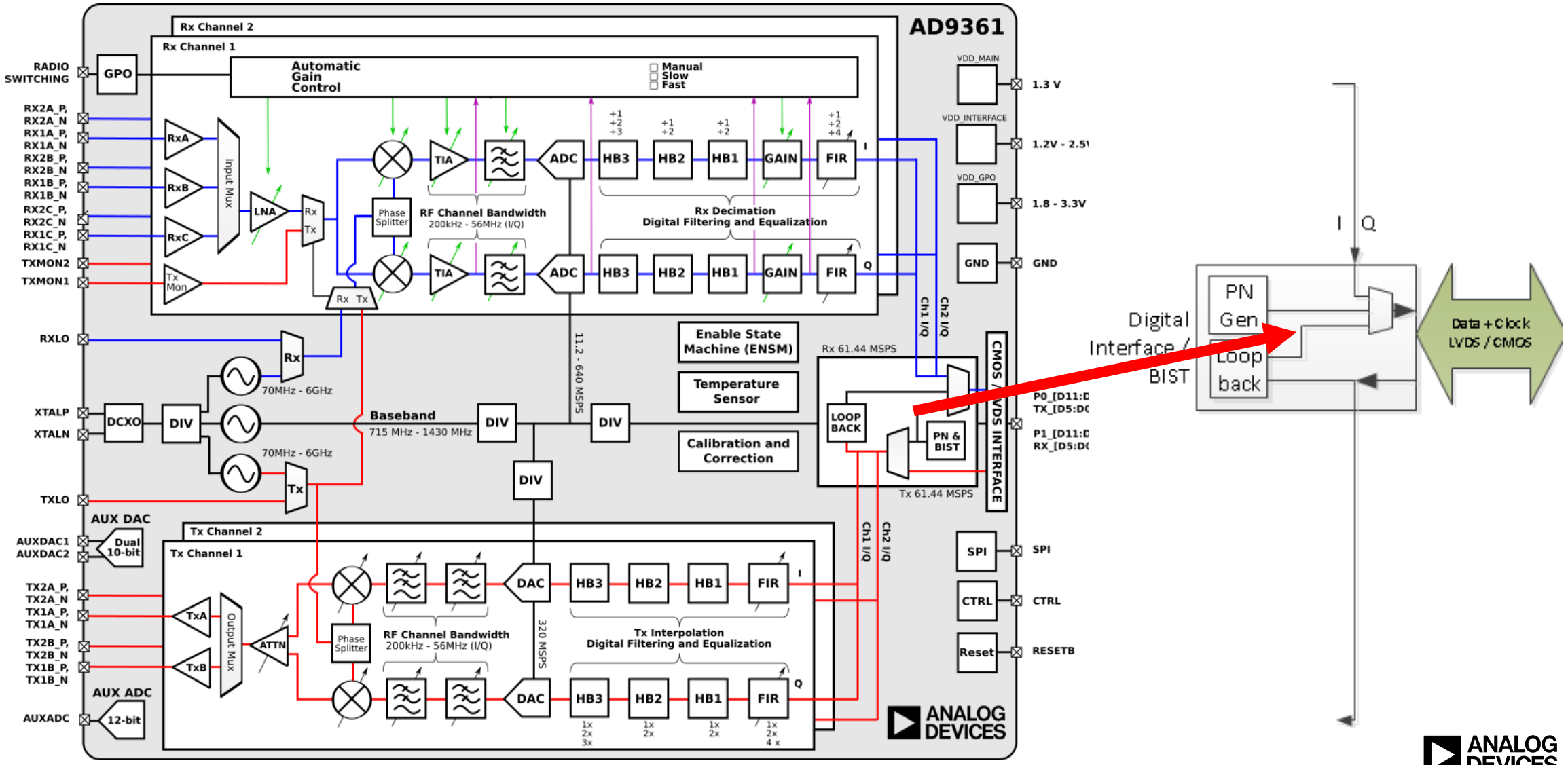
- Half bands, FIR, and clocking determine data rate of Pluto
  - FIR needs to be loaded to go below 2.08 MHz
- Multiple options for loading FIR
  - Auto-filter options in gr-iio, IIO-Scope, and through libad9361
  - Filters can be custom created with AD9361 Filter Wizard
  - Filters can be custom created in C/C++ with libad9361



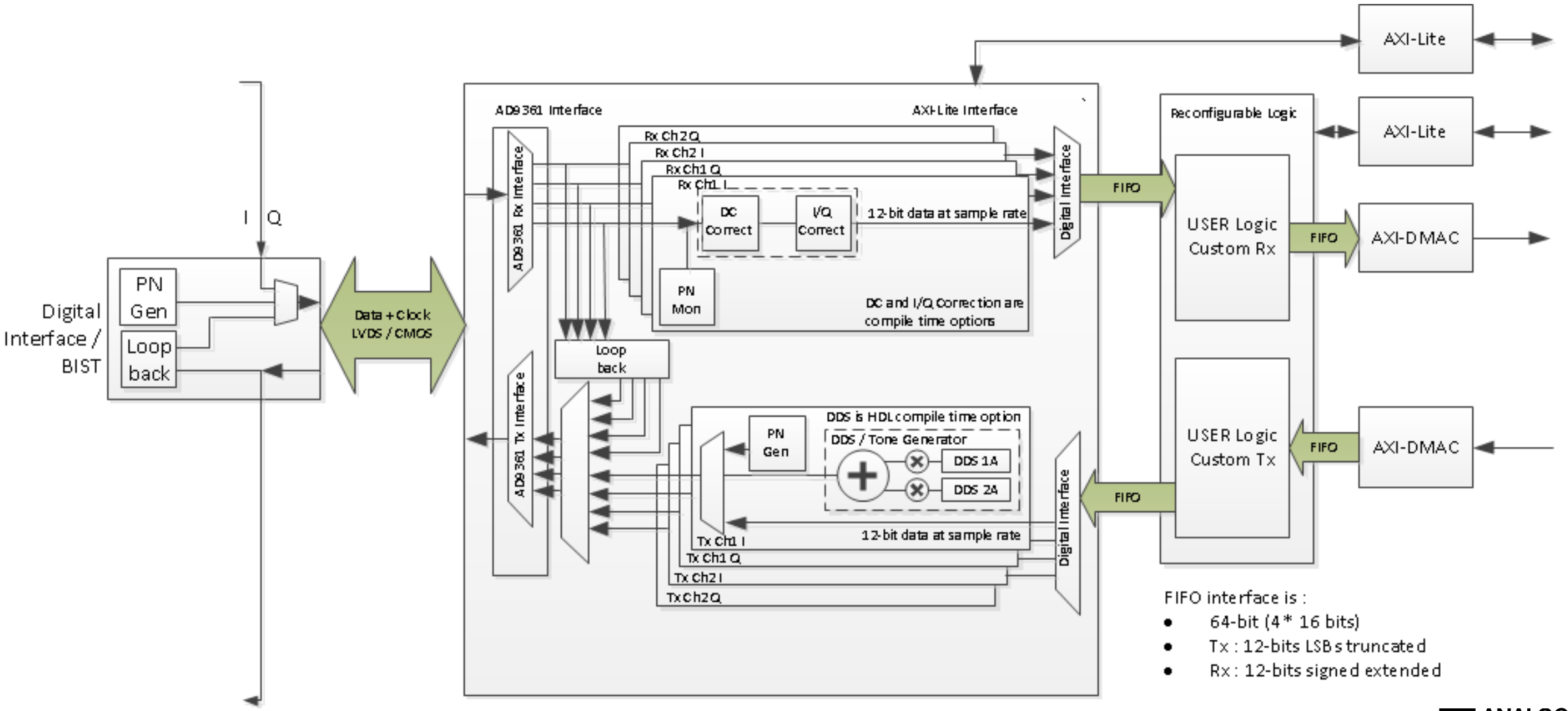




Features on the FPGA



HDL: Digital Loopback

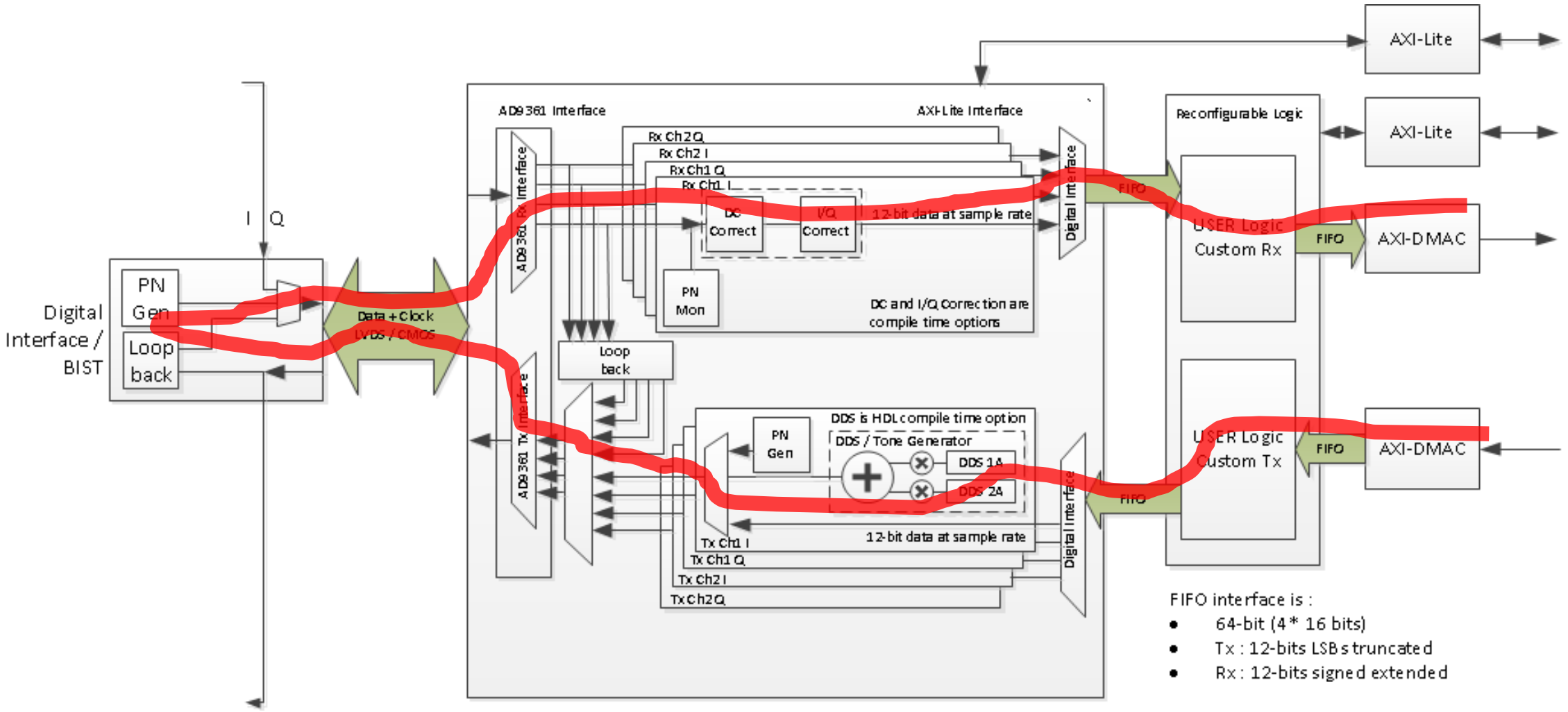


FIFO interface is :

- 64-bit (4 * 16 bits)
- Tx : 12-bits LSBs truncated
- Rx : 12-bits signed extended

ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

HDL: Digital Loopback

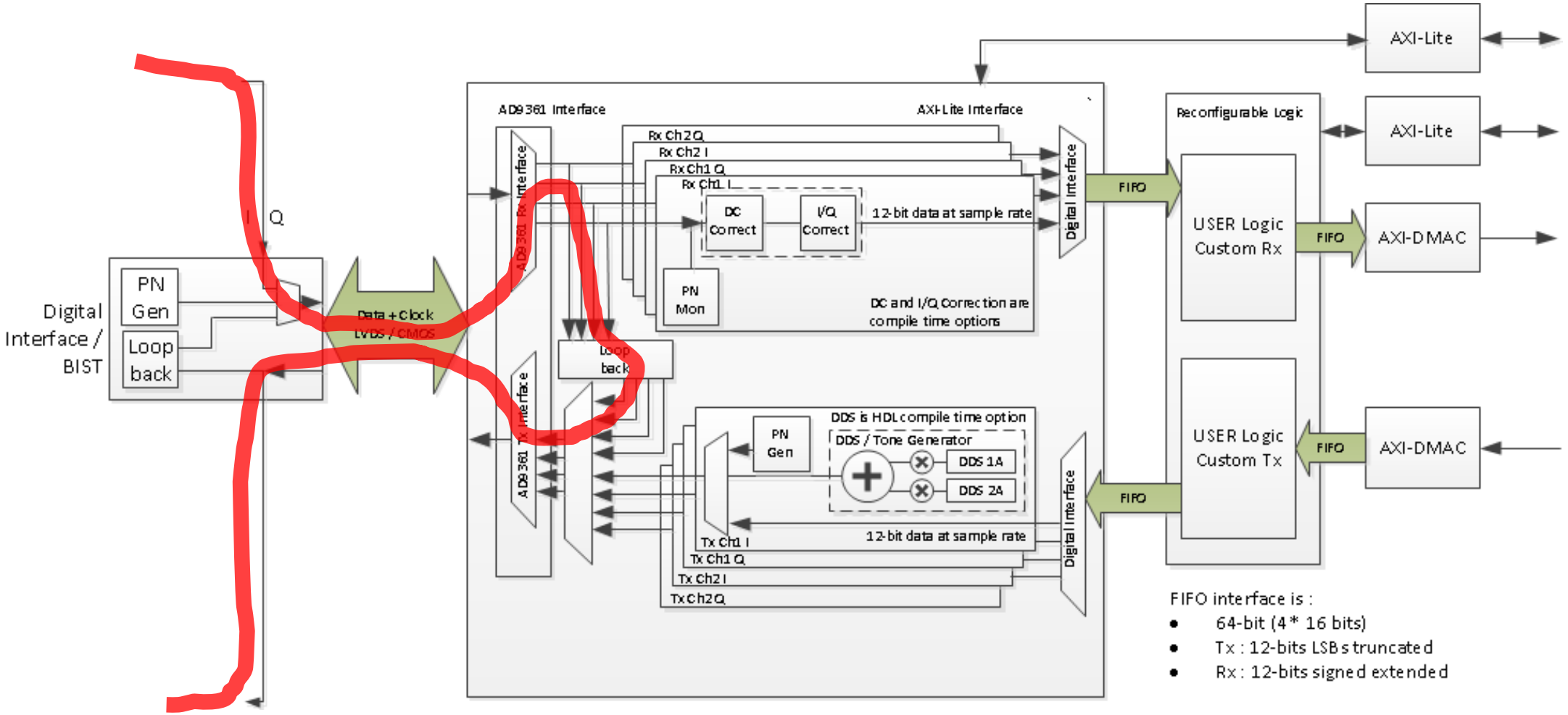


FIFO interface is :

- 64-bit (4 * 16 bits)
- Tx : 12-bits LSBs truncated
- Rx : 12-bits signed extended

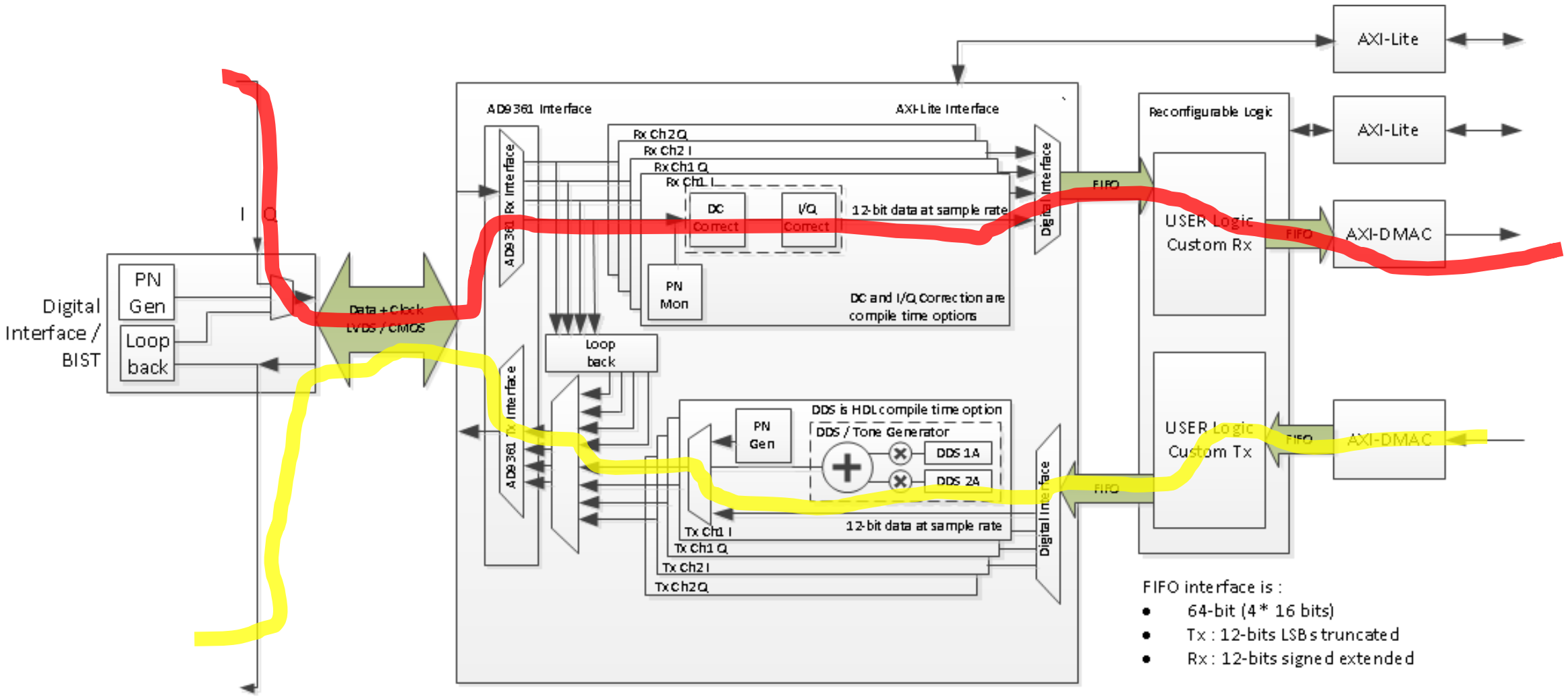
ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

HDL: RF Loopback



ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

HDL: Tx/Rx Active



FIFO interface is :

- 64-bit (4 * 16 bits)
- Tx : 12-bits LSBs truncated
- Rx : 12-bits signed extended

ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

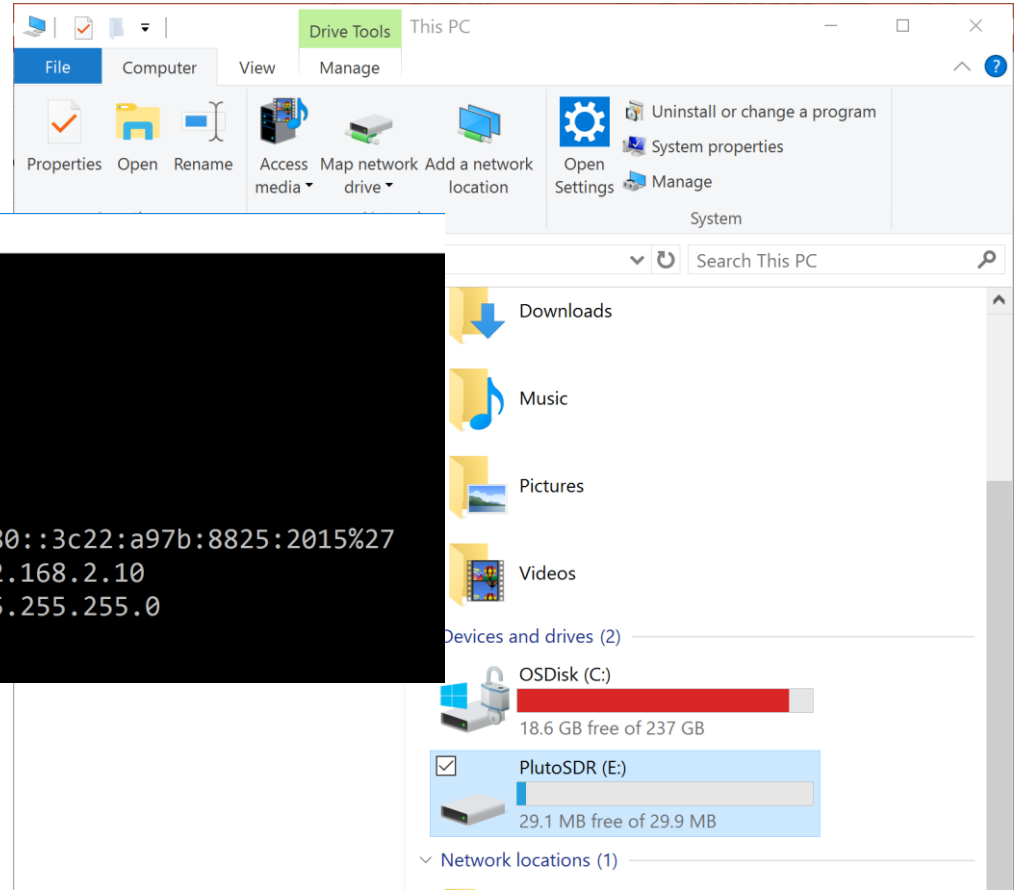
ANALOG DEVICES

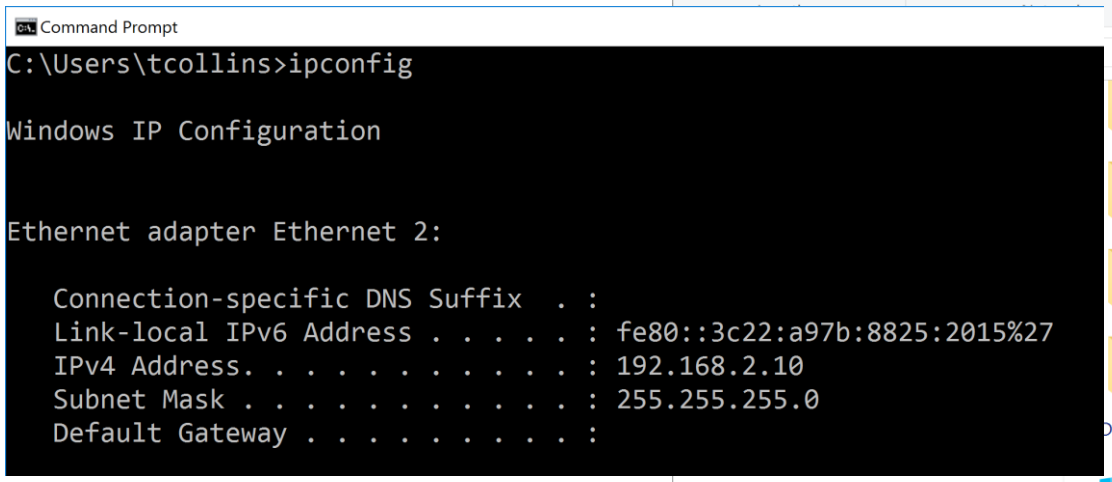
AHEAD OF WHAT'S POSSIBLE™

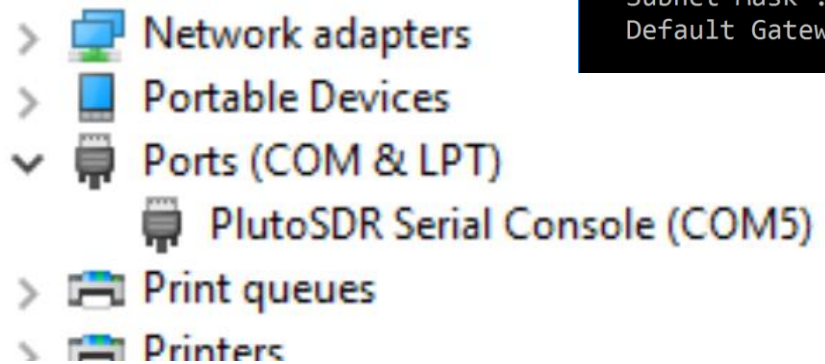
Drivers and Firmware

Connecting With PlutoSDR

- Pluto enumerates three devices
  - Mass storage
  - Ethernet (RNDIS)
  - Serial







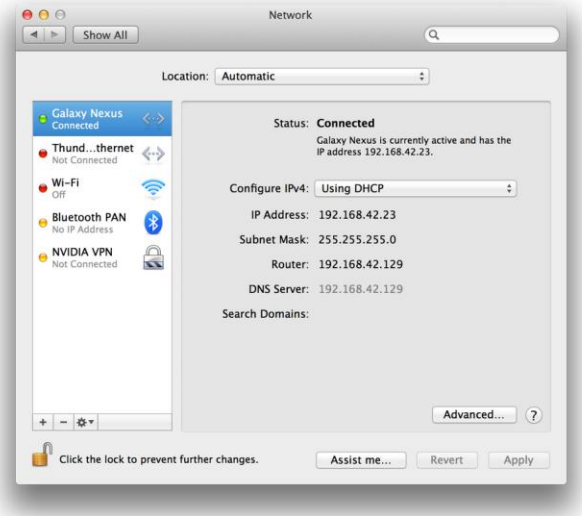
Drivers

- Linux and Windows have all three options out of the box
- macOS requires HoRNDIS install for networking

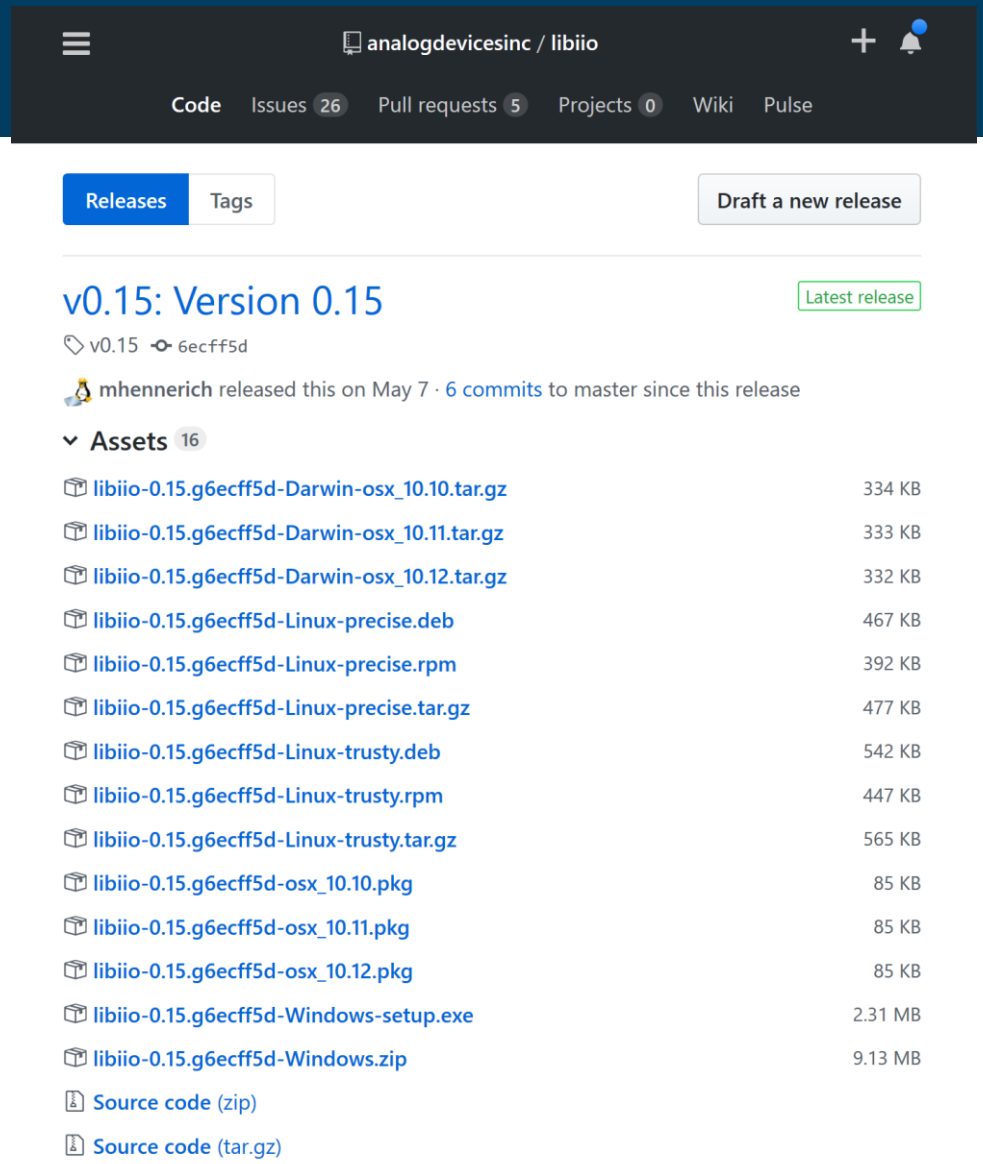
Downloading and installing HoRNDIS

HoRNDIS is available in source form from its project page on GitHub, and in binary form on this site. For quick start instructions:

- Download the appropriate binary package, and double-click on it in your Downloads folder. Follow the instructions in the installer.
- Assuming that the installation proceeds without errors, after it completes, connect your phone to your Mac by USB.
- Enter the settings menu on your phone.
- In the connections section, below Wi-Fi and Bluetooth, select "More...".
- Select "Tethering & portable hotspot".
- Check the "USB tethering" box. It should flash once, and then become solidly checked.



On some versions of OS X, a dialog box may pop up, prompting you to configure the device; follow its instructions. To verify that the device is connected,



ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

Firmware

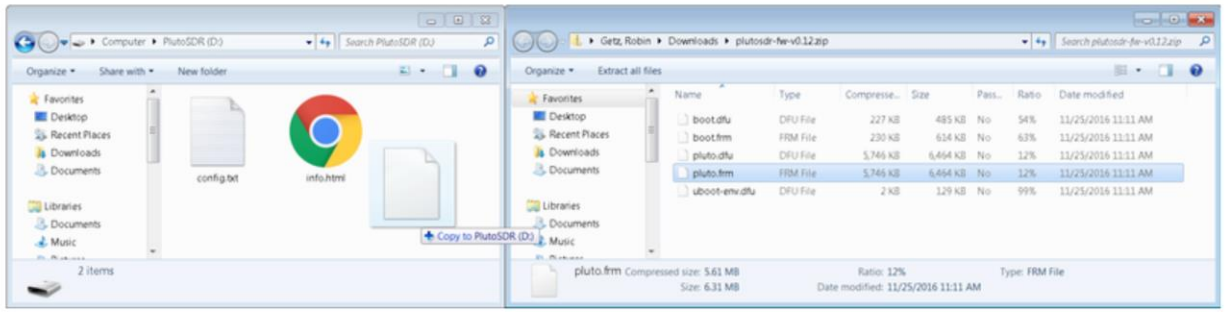
ADI default firmware images:

- Latest ADALM-PLUTO (PlutoSDR) Release
- Latest ADALM-2000 (M2k) Release

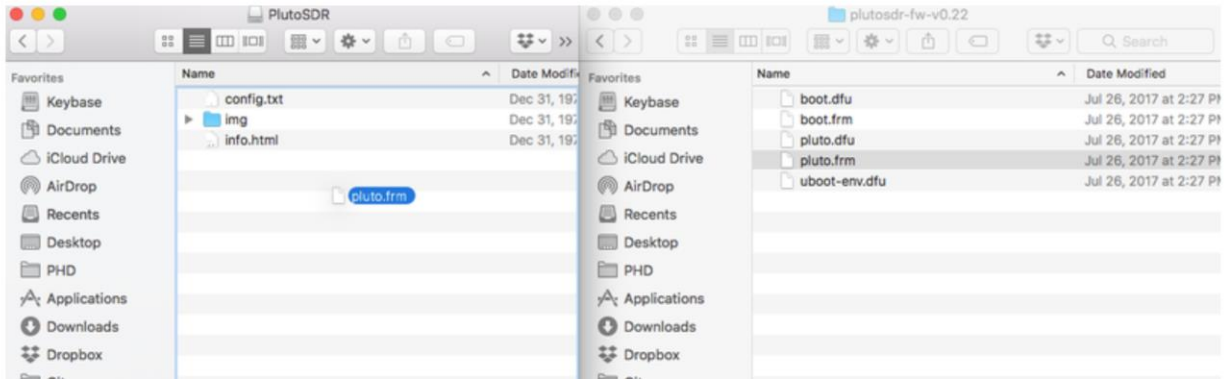
This zip file should include these files:

Filename	Purpose
`boot.dfu`	DFU file for First Stage Boot Loader, and U-Boot
`boot.frm`	Firmware file for First Stage Boot Loader, U-Boot and it's default environment
`pluto.dfu`	DFU file for Pluto Firmware, this would include FPGA Bit File, Linux kernel (all drivers), and ram based file system
`m2k.dfu`	DFU file for M2k Firmware, this would include FPGA Bit File, Linux kernel (all drivers), and ram based file system
`pluto.frm`	Firmware file for Pluto Firmware, this would include FPGA Bit File, Linux kernel (all drivers), and ram based file system
`m2k.frm`	Firmware file for M2k Firmware, this would include FPGA Bit File, Linux kernel (all drivers), and ram based file system
`uboot-env.dfu`	DFU file which includes the default U-Boot environment

1. Windows

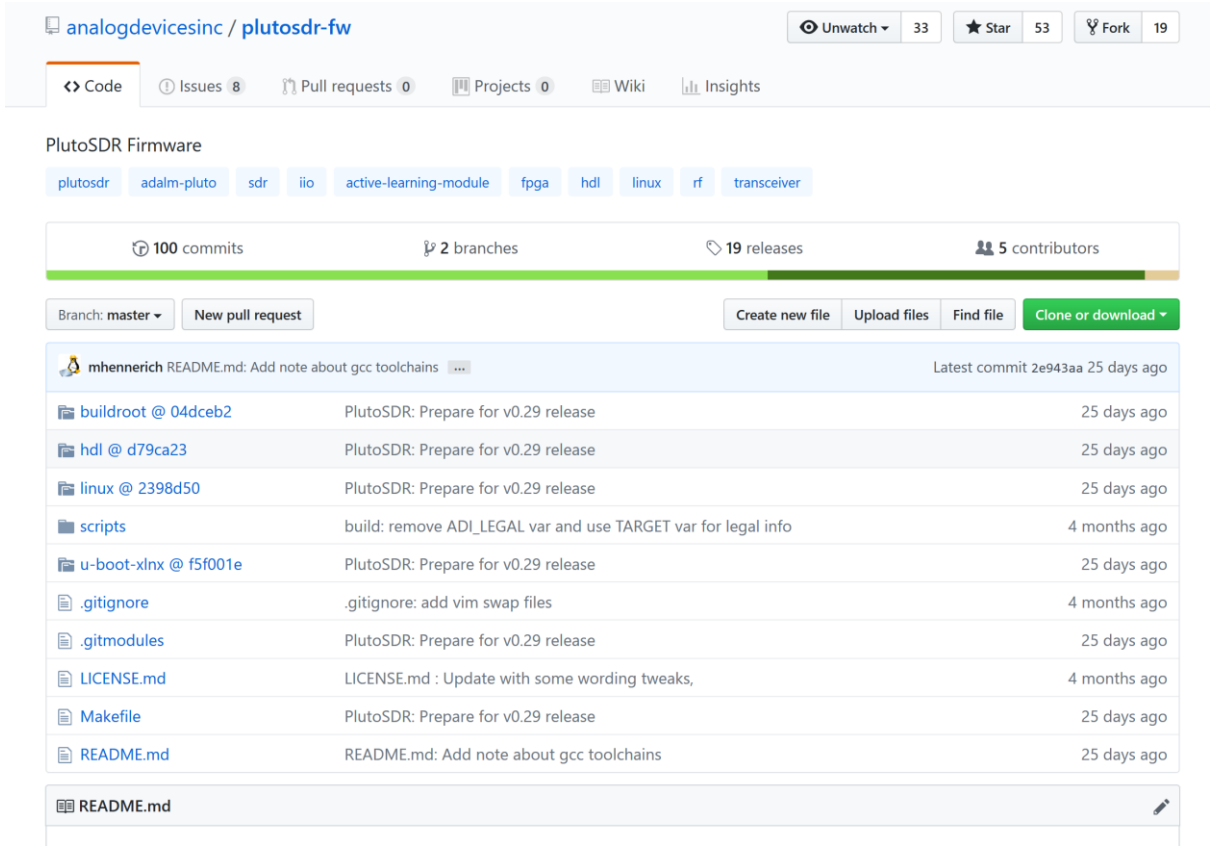


2. OSX



► *Notes on using Zip file*

Building your own firmware (Linux + HDL)



analogdevicesinc / plutosdr-fw

Unwatch 33 | Star 53 | Fork 19

Code | Issues 8 | Pull requests 0 | Projects 0 | Wiki | Insights

PlutoSDR Firmware

plutosdr adalm-pluto sdr iio active-learning-module fpga hdl linux rf transceiver

100 commits | 2 branches | 19 releases | 5 contributors

Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

mhennerich README.md: Add note about gcc toolchains ... Latest commit 2e943aa 25 days ago

buildroot @ 04dceb2	PlutoSDR: Prepare for v0.29 release	25 days ago
hdl @ d79ca23	PlutoSDR: Prepare for v0.29 release	25 days ago
linux @ 2398d50	PlutoSDR: Prepare for v0.29 release	25 days ago
scripts	build: remove ADI_LEGAL var and use TARGET var for legal info	4 months ago
u-boot-xlnx @ f5f001e	PlutoSDR: Prepare for v0.29 release	25 days ago
.gitignore	.gitignore: add vim swap files	4 months ago
.gitmodules	PlutoSDR: Prepare for v0.29 release	25 days ago
LICENSE.md	LICENSE.md : Update with some wording tweaks,	4 months ago
Makefile	PlutoSDR: Prepare for v0.29 release	25 days ago
README.md	README.md: Add note about gcc toolchains	25 days ago

README.md

plutosdr-fw

PlutoSDR Firmware for the ADALM-PLUTO Active Learning Module

Latest binary Release : release v0.29

Instructions from the Wiki: Building the image

- Build Instructions

```
sudo apt-get install git build-essential fakeroot libncurses5-dev libssl-dev ccache
sudo apt-get install dfu-util u-boot-tools device-tree-compiler libssl1.0-dev mtools
git clone --recursive https://github.com/analogdevicesinc/plutosdr-fw.git
cd plutosdr-fw
export CROSS_COMPILE=arm-xilinx-linux-gnueabi-
export PATH=$PATH:/opt/Xilinx/SDK/2017.2/gnu/arm/lin/bin
export VIVADO_SETTINGS=/opt/Xilinx/Vivado/2017.4/settings64.sh
make
```

The project may build also using Vivado 2017.2, 2016.4 or 2016.2. However 2017.4 is the current tested FPGA systhesis toolchain. For comatibility reasons with existing targeting workflows we continue to use the arm-xilinx-linux-gnueabi-gcc toolchain íncluded in the SDK 2017.2.

If you want to use the arm-linux-gnueabihf-gcc hard-float toolchain included in SDK 2017.4. Following variables should be exported:

```
export CROSS_COMPILE=arm-linux-gnueabihf-
export PATH=$PATH:/opt/Xilinx/SDK/2017.4/gnu/aarch32/lin/gcc-arm-linux-gnueabi/bin
```

ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

Lab: Updating Firmware

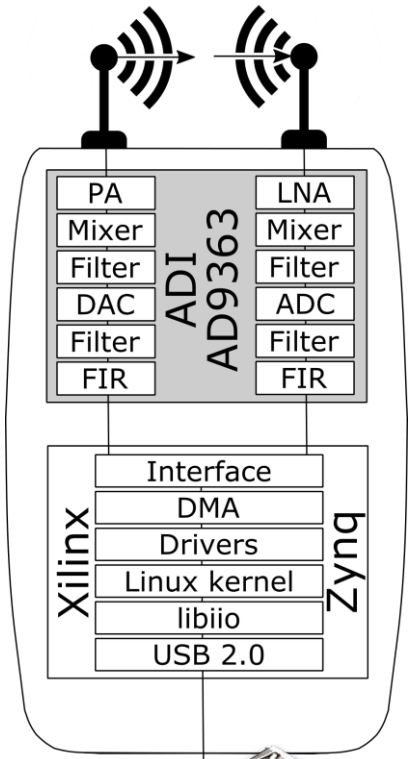
ANALOG DEVICES

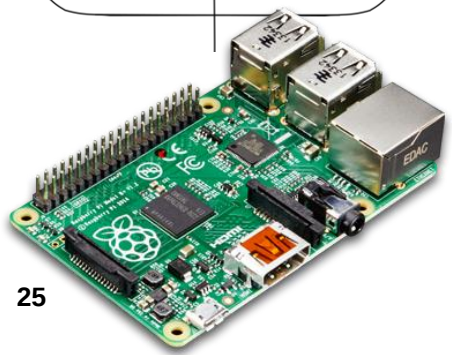
AHEAD OF WHAT'S POSSIBLE™

USB Configurations

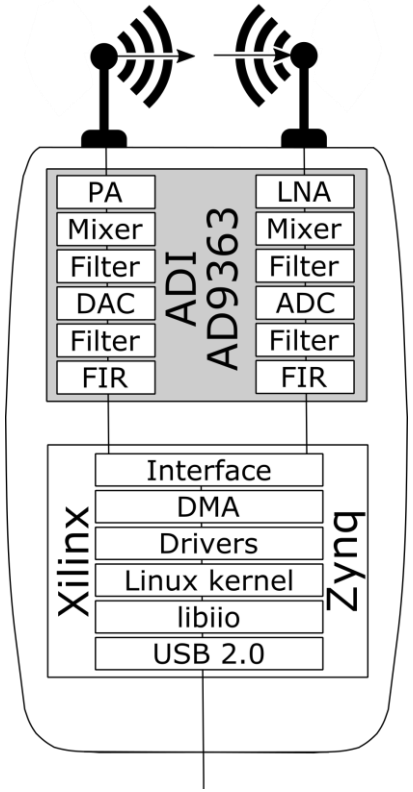
ADALM-PLUTO possible use cases

Connect to host

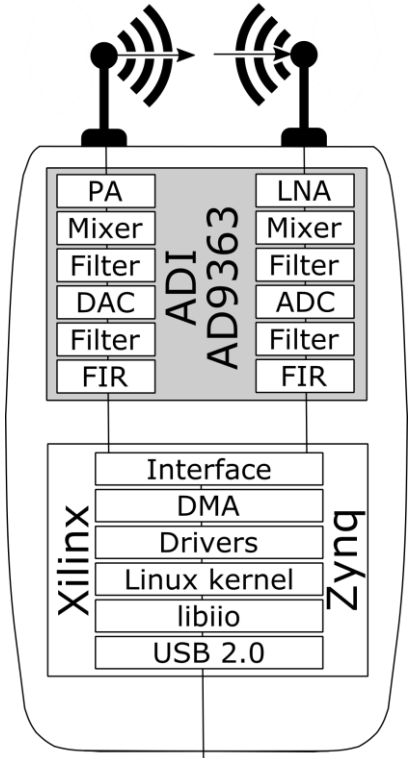


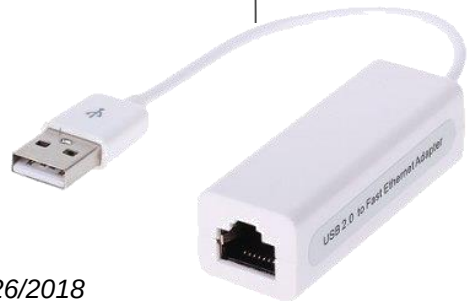


Connect to thumb drive

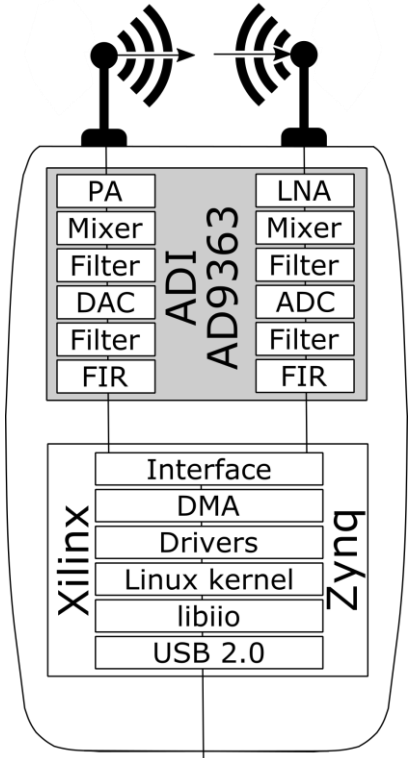


Connect to USB LAN





Connect to USB WiFi



ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

USB OTG

- The Pluto will automount any USB mass storage device such as thumb drive or Hard Drives
- The automounter will then look for some special file names:
  - runme[0-9].sh which it will run as a shell script
  - runme[0-9] which it will run as a binary file.
- For those interested, it will do that via the automounter script in /lib/mdev/automounter.sh which is maintained here.

1. Provide power to right USB
2. Plug OTG+USB drive into left USB port
3. Left LED will become solid
4. Script runs

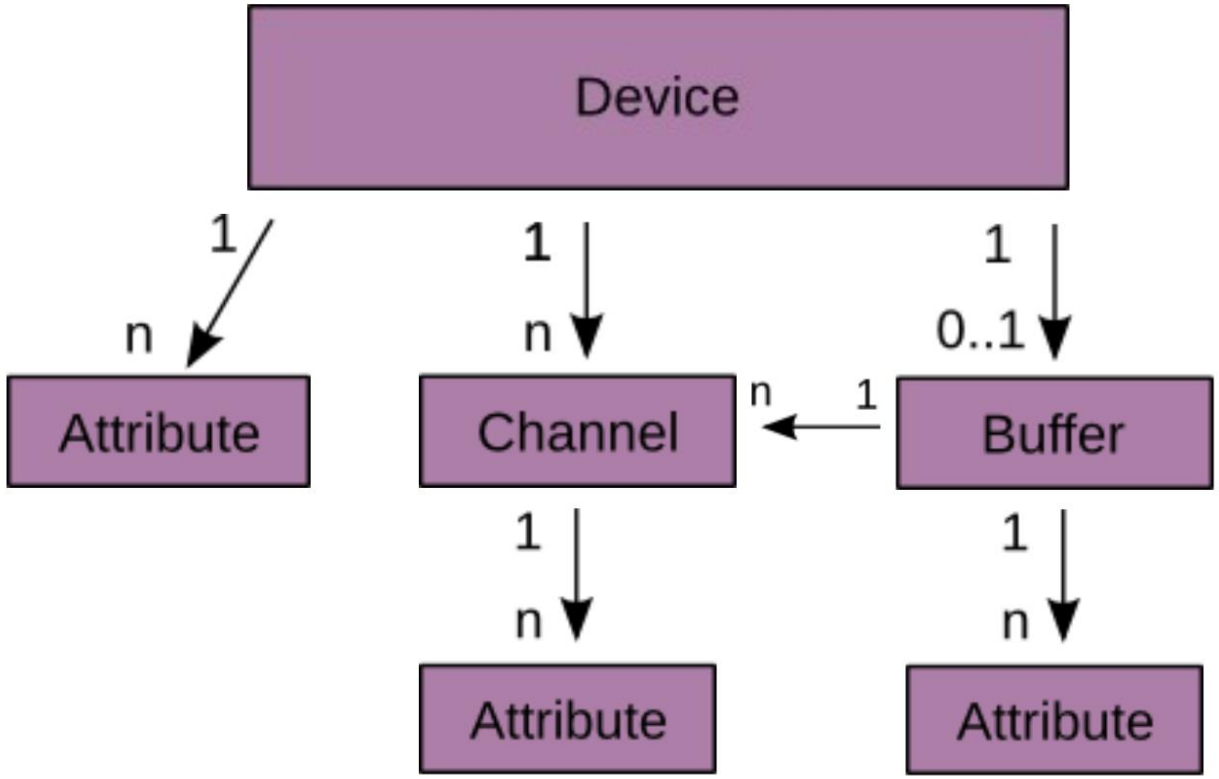
ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

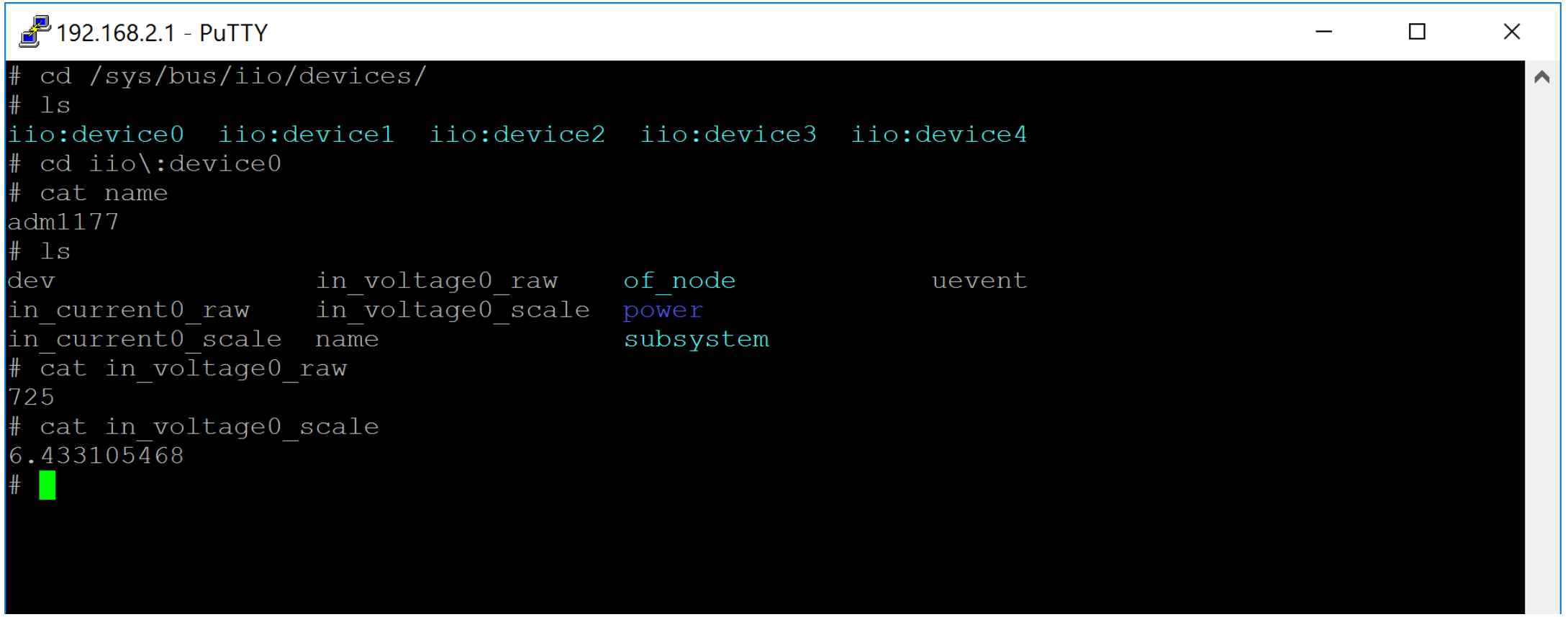
IIO and (The) libIIO

IIO Structure

- Device represents logical functional unit
  - Typically a piece of physical hardware
- Attributes
  - Describe hardware capabilities
  - Allow to change hardware configuration
- Channels represent data channels
  - Channels have a type and direction
  - E.g. ADC has voltage channels
  - Channels can have attributes



IIO Userspace ABI - Devices

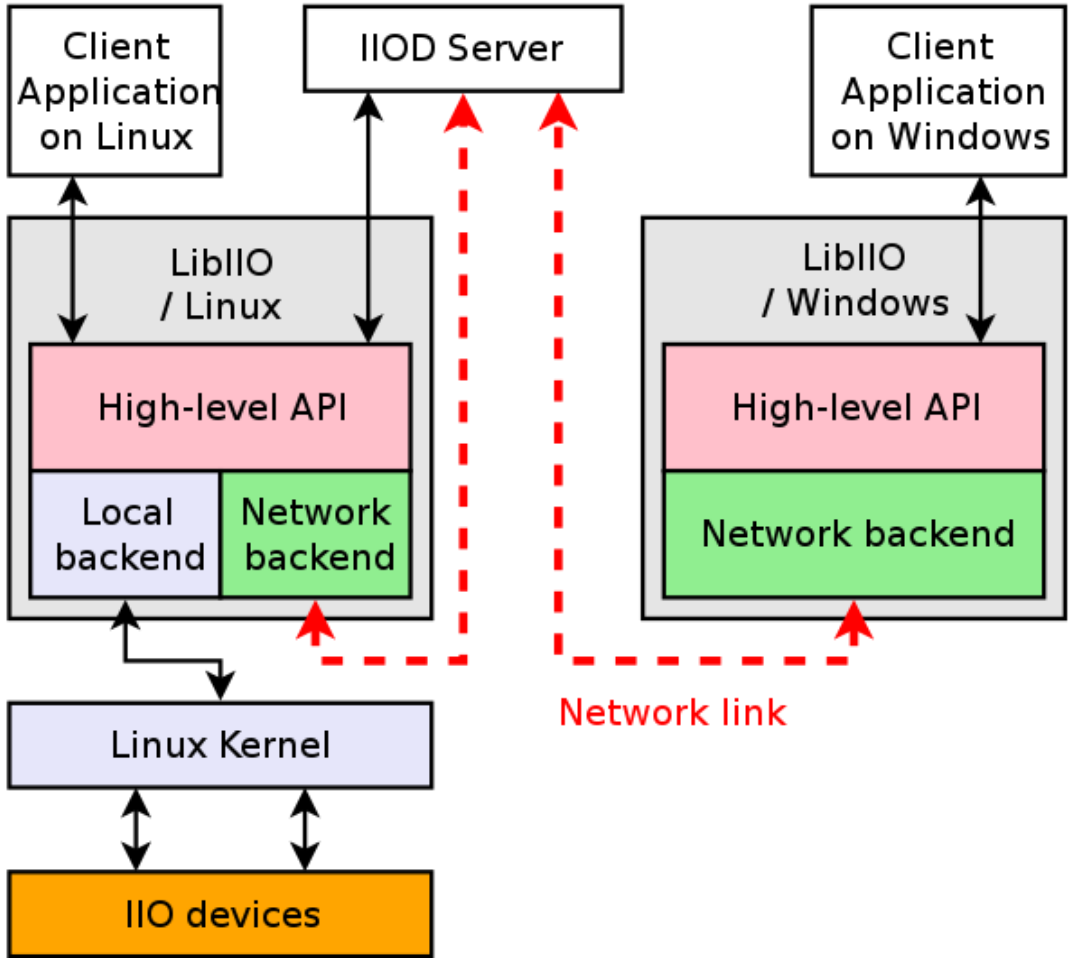


IIO Userspace ABI - Buffers

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

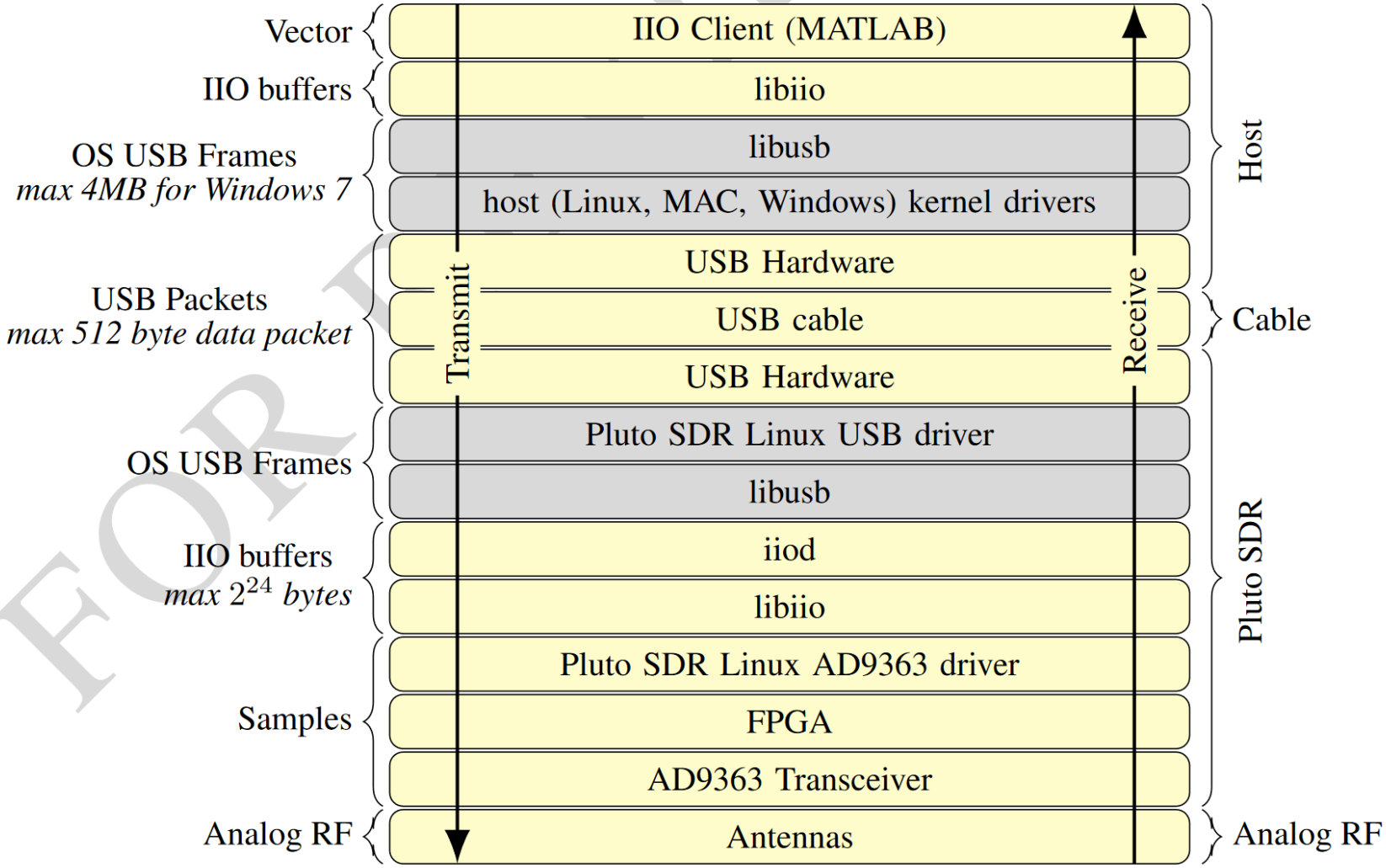



```c
struct iio_context *ctx;
struct iio_device *dev;
struct iio_channel *ch;

/* 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);
```

The Stack



libIIO Command Line Tools

- 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

- All these tools come with a libIIO install
- Available on target platforms as well
- Cross-platform

ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

ILD: libIIO

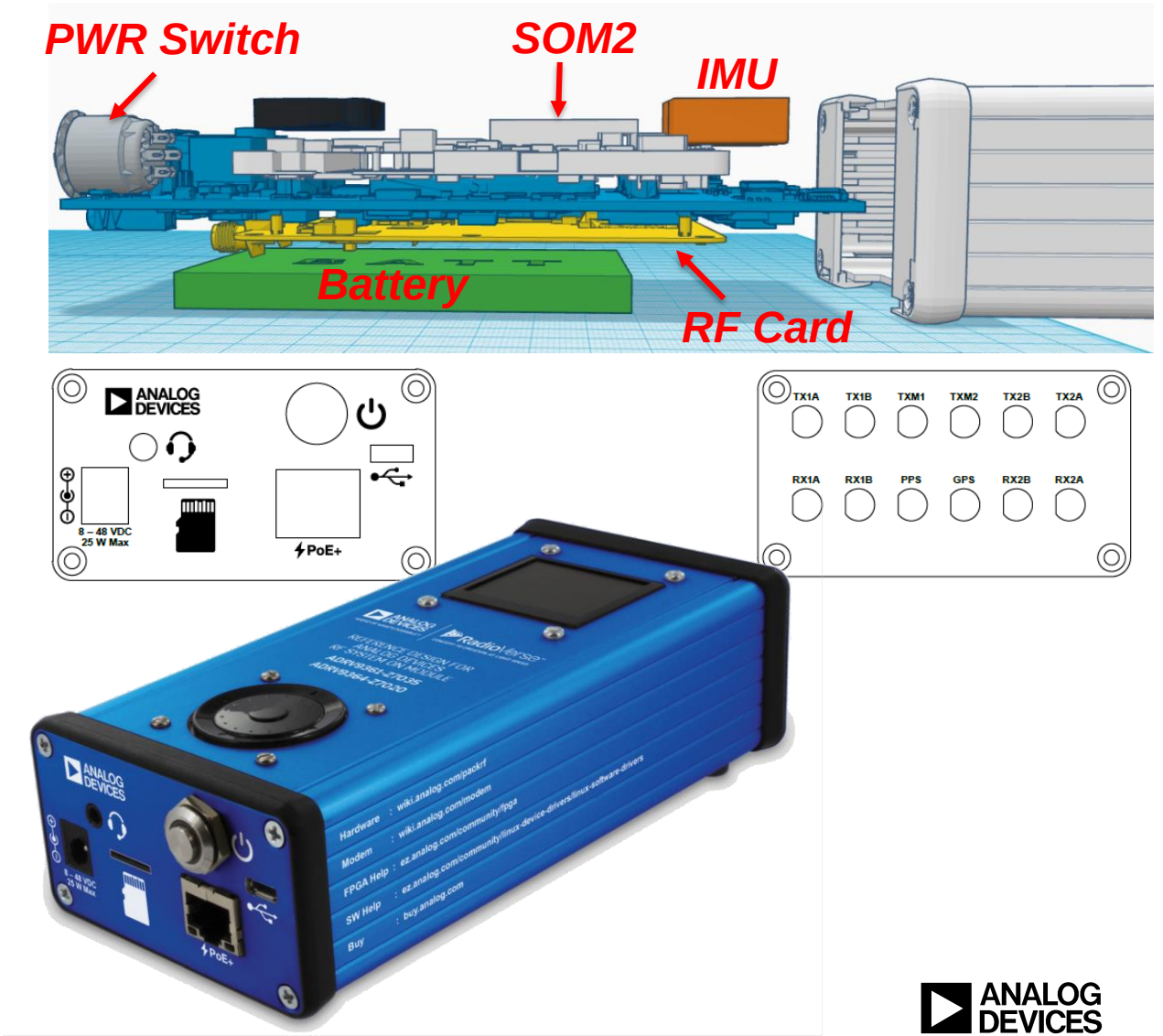
ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

Lab: IIO Tools + IIO-Scope

Devices Outsides of Transceivers

- Example design which shows how to design RF SOM into a custom carrier
- Custom Carrier includes:
  - OLED
  - Nav Switch
  - Power Button
    - Wake on RTC
  - Power over Ethernet (PoE+)
  - Automotive DC-DC converter
    - 8 – 48V DC input
  - Battery Management
  - Hot Power swap
  - Inertial Measurement Unit
    - Six Degrees of Freedom
  - GPS Chipset
    - 1 PPS in and out
  - Audio headset (stereo headphones, mic and button control)



ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

ILD: PackRF and IIO

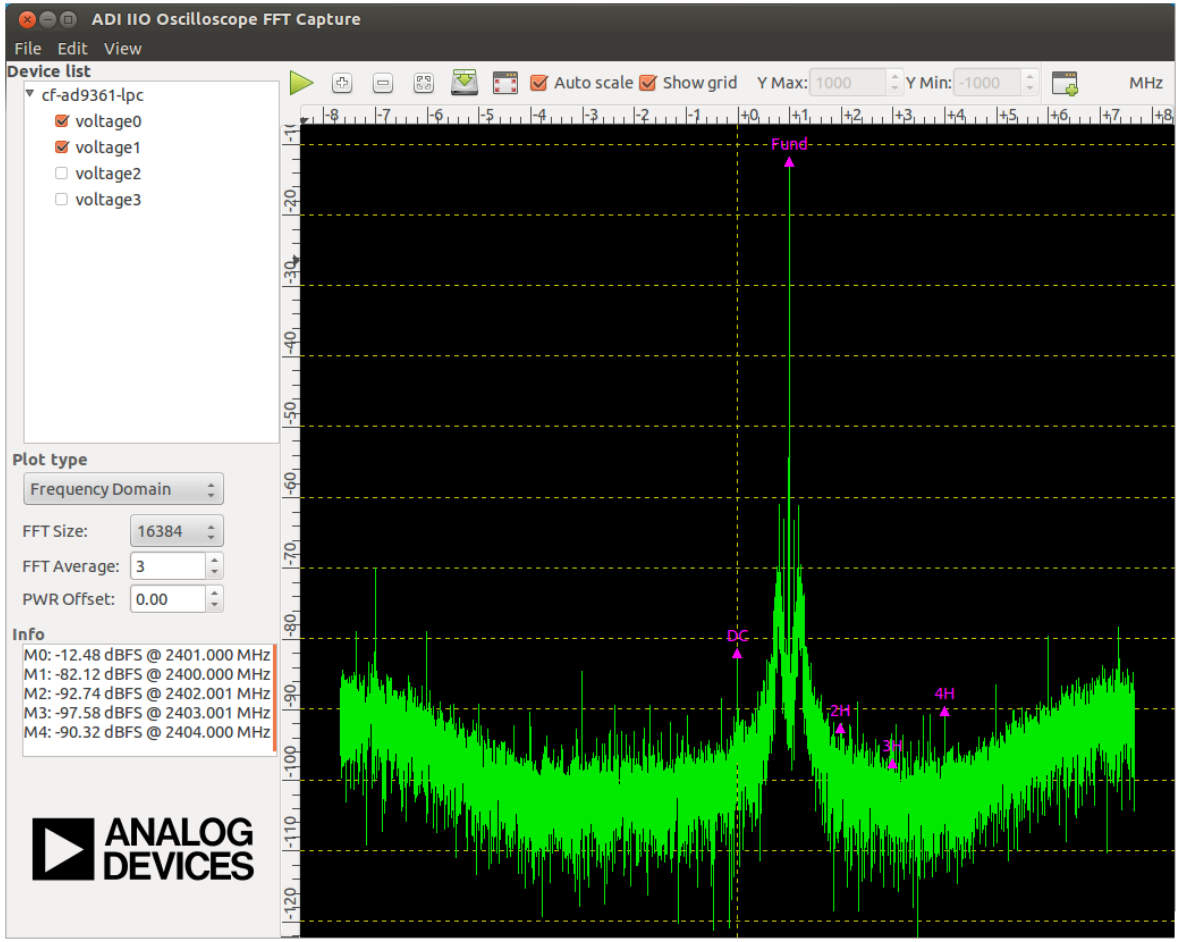
ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

IIO-Scope

IIO-Scope

- Capture and display data
  - Time domain
  - Frequency domain
  - Constellation plot
  - Markers
  - Math operations
- Device configuration
- Plug-in system allow to create device or complex specialized GU



ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

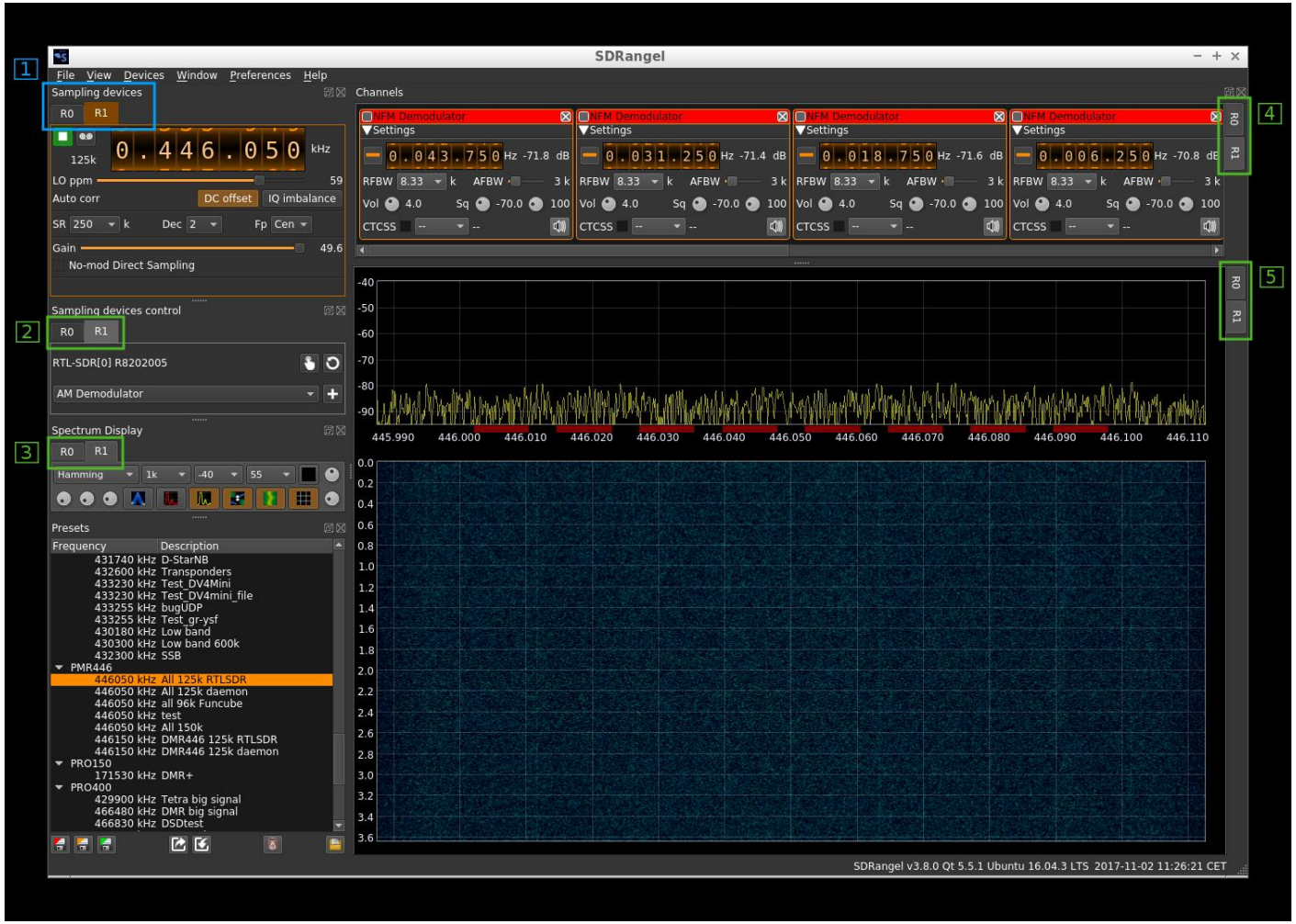
Lab: IIO-Scope

ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

Fun with SDRAngel

SDRAngel at a glance



ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

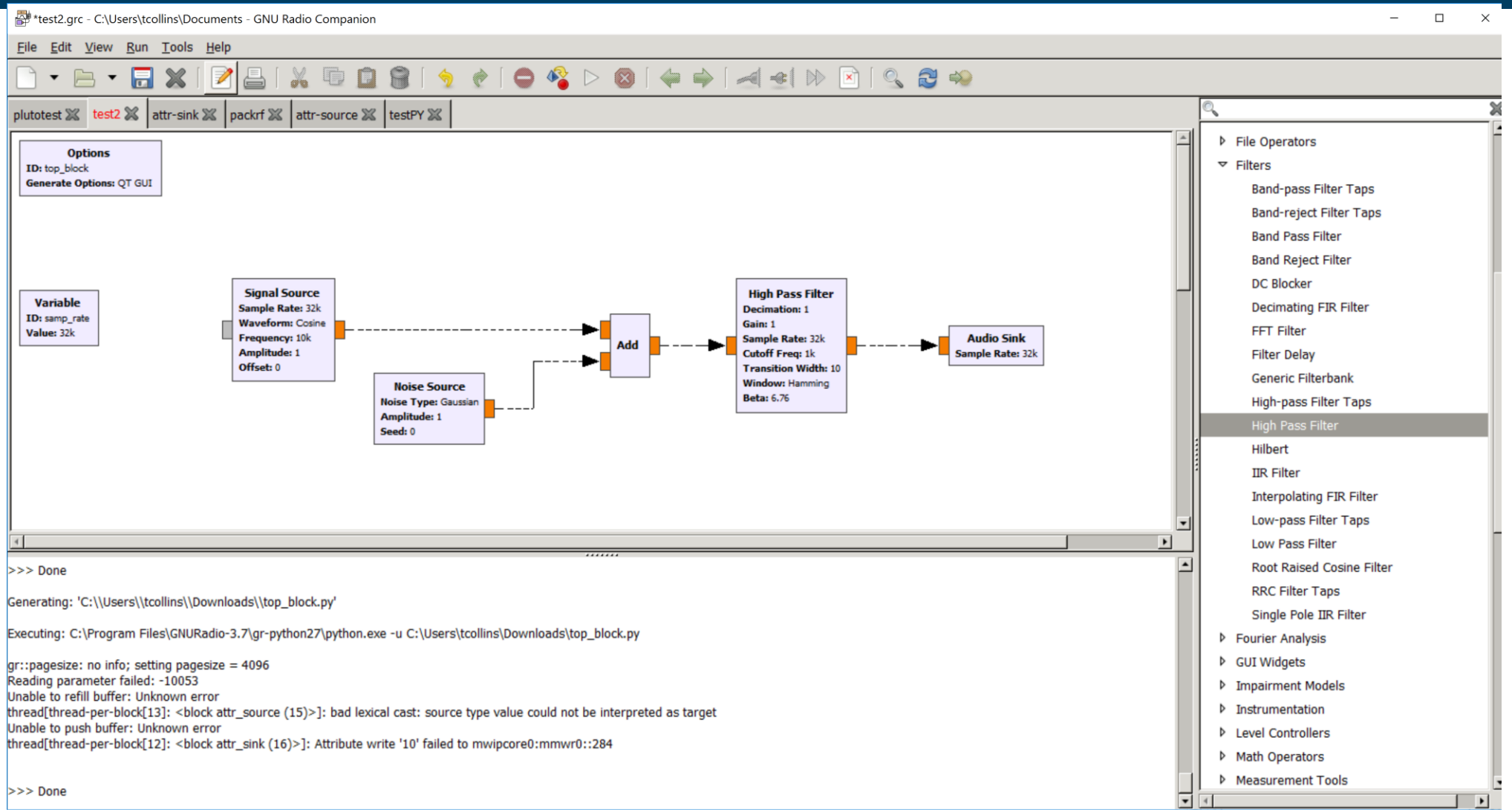
ILD: SDR Angel

ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

GNU Radio + IIO = gr-iio

Intro To GNU Radio



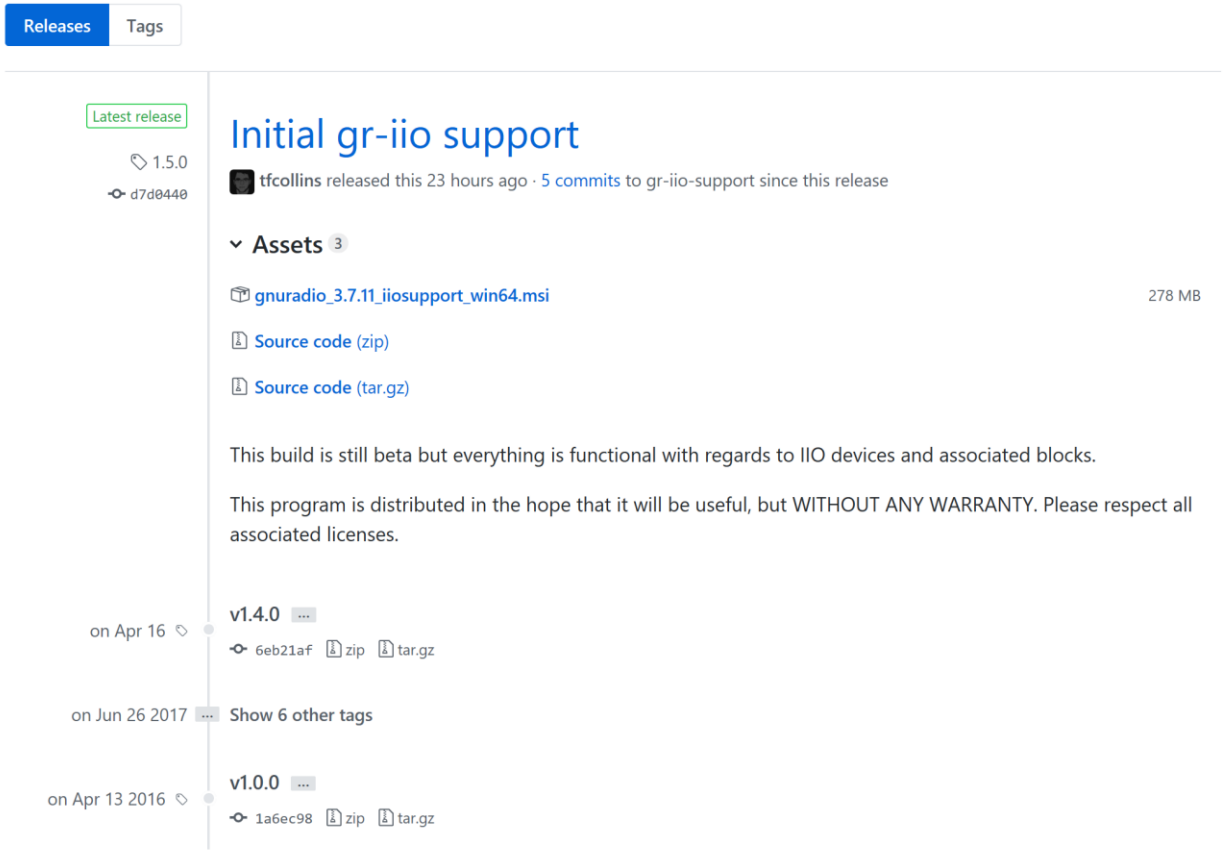
ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

gr-iio: Details

- Repository
  - https://github.com/analogdevicesinc/gr-iio
- Dependencies
  - libiio
  - libad9361-iio
- Doc
  - https://wiki.analog.com/resources/tools-software/linux-software/gnuradio

```
git clone https://github.com/analogdevicesinc/gr-iio.git
cd gr-iio
cmake .
make
sudo make install
cd ..
sudo ldconfig
```

- Windows



IIO GNU Radio Support: gr-iio

- Industrial IO
  - FMComms
    - FMComms2/3/4 Sink
    - FMComms2/3/4 Source
    - FMComms5 Sink
    - FMComms5 Source
  - IIO Attribute Sink
  - IIO Attribute Source
  - IIO Attribute Updater
  - IIO Device Sink
  - IIO Device Source
  - Math Operators
    - Function
    - Modulo
    - Modulo Const
    - Power
  - PlutoSDR
    - PlutoSDR Sink
    - PlutoSDR Source
  - Waveform Generators
    - Function Generator

SDR

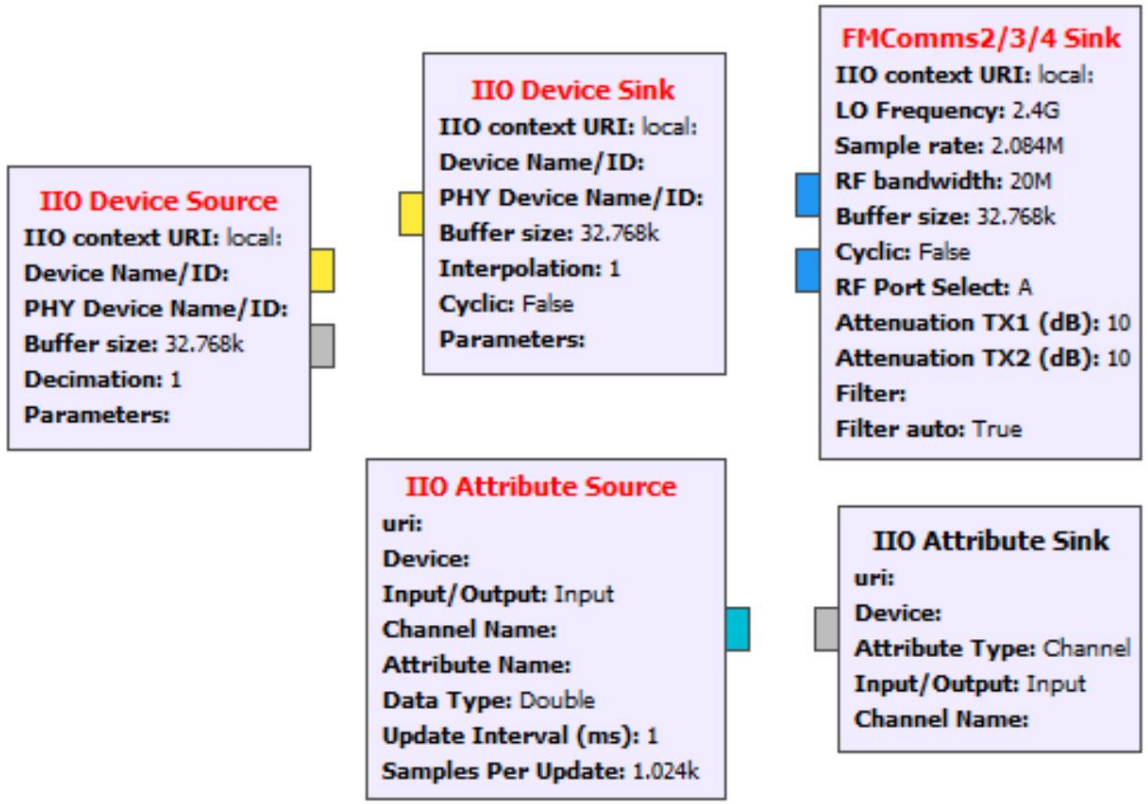
Attributes

Streams

Math (Scopy)

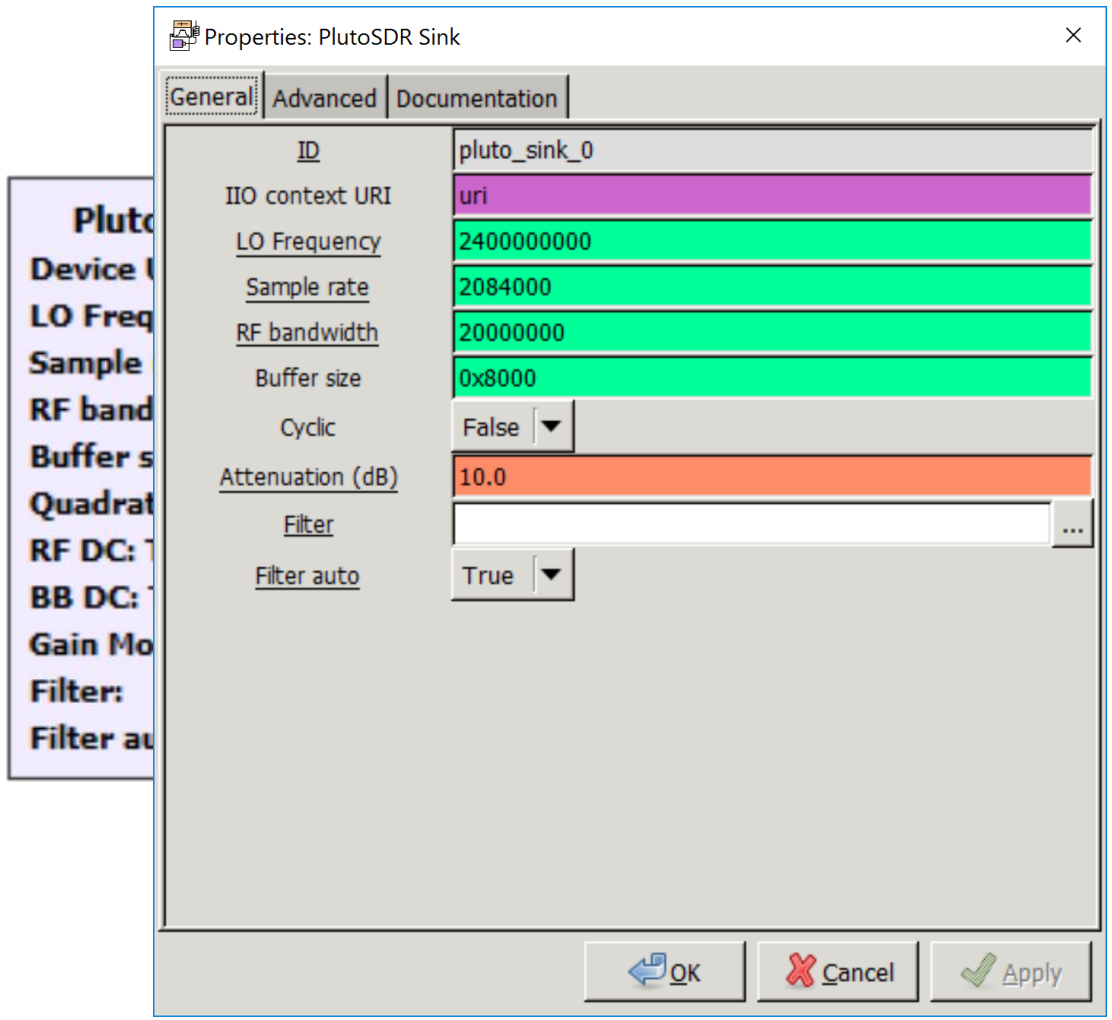
SDR

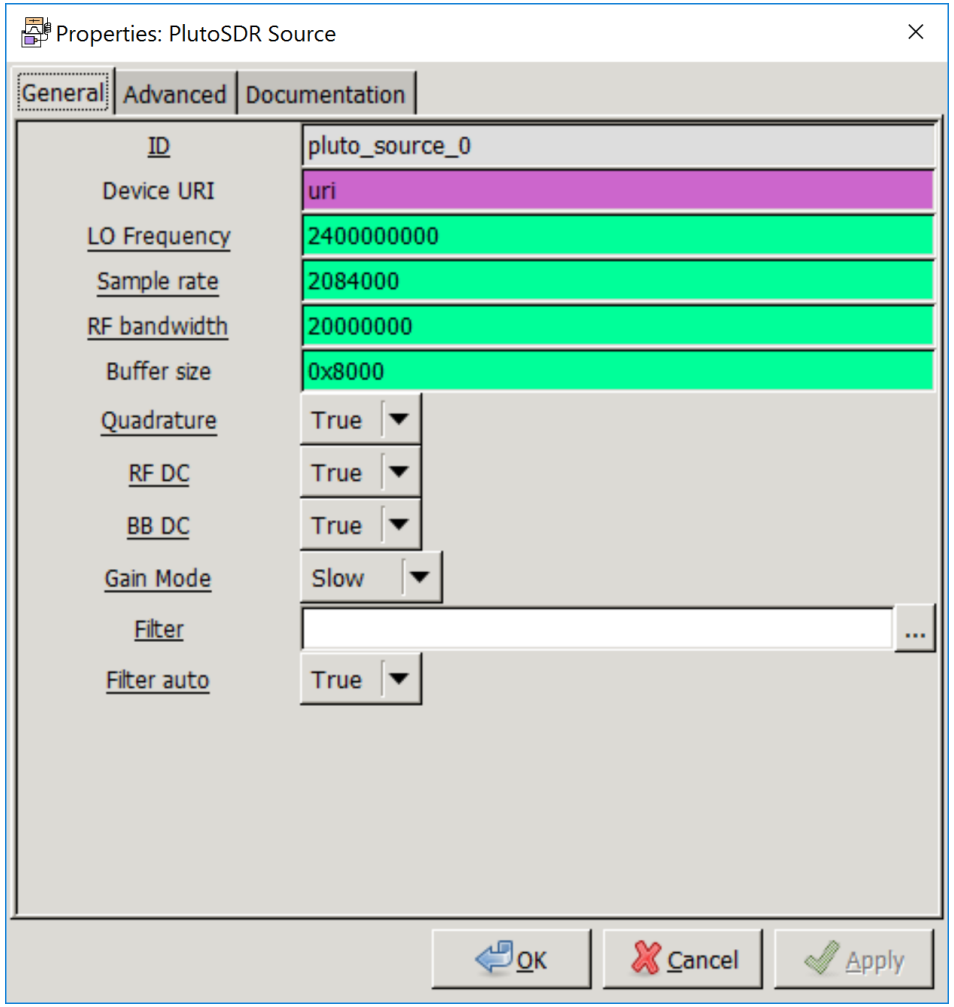
Math (Scopy)



ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

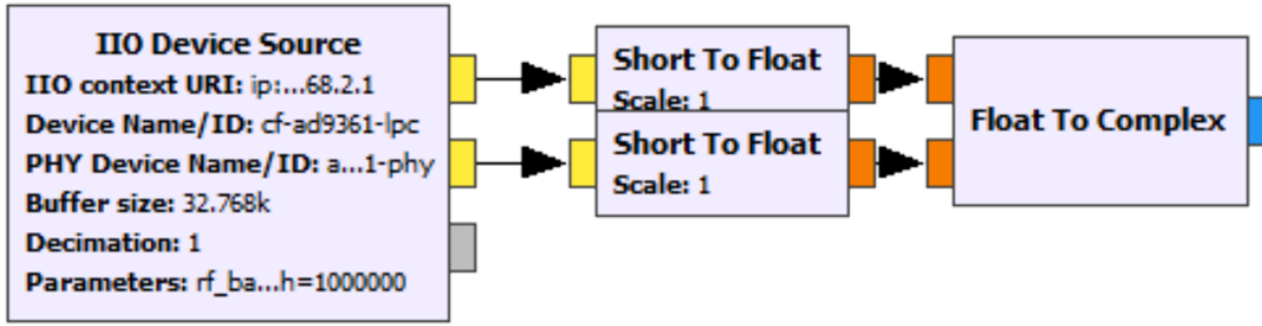
Using Pluto Blocks

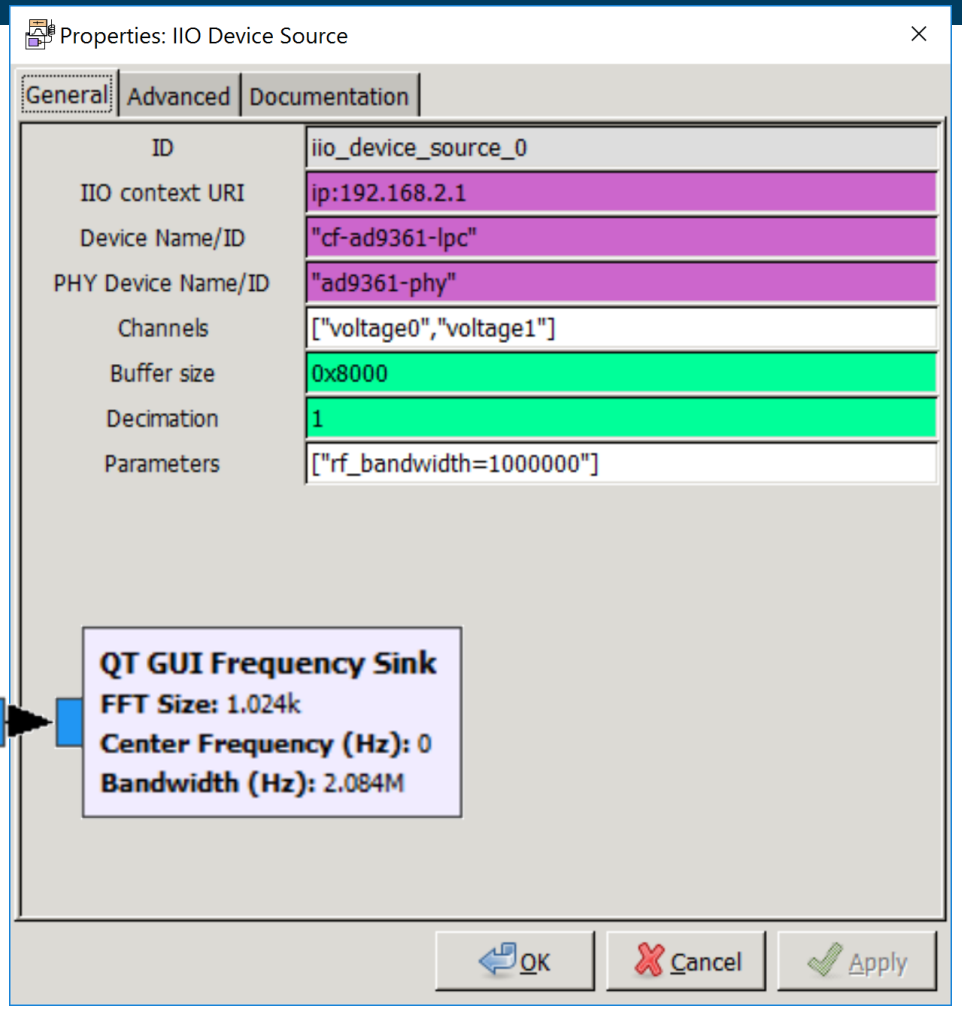




Stream (Device) Blocks

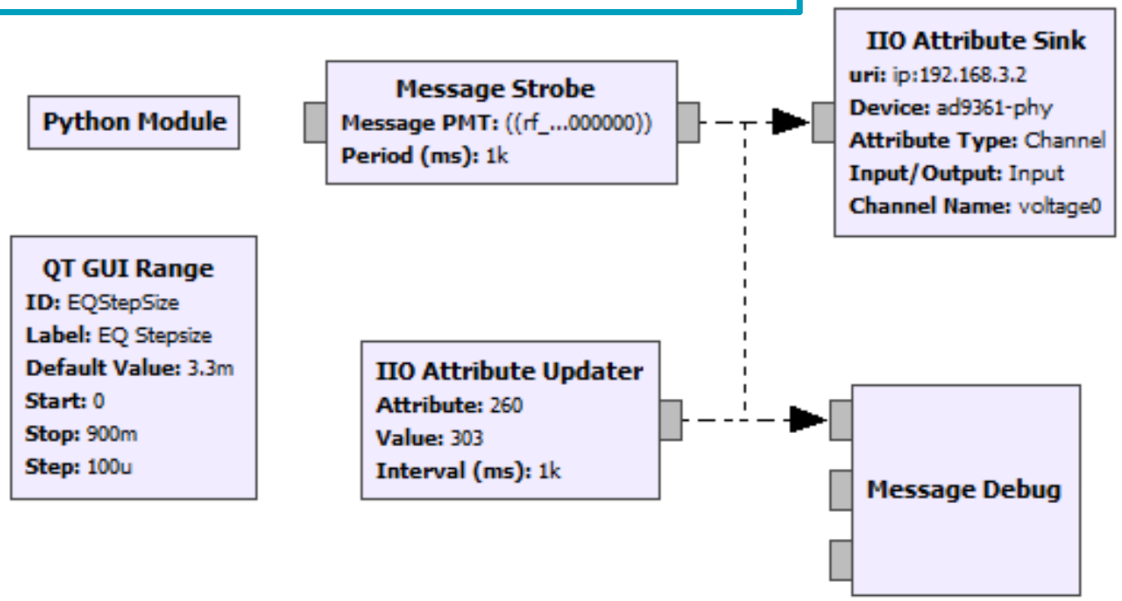
```
C:\Users\tcollins>iio_attr -q -u ip:192.168.2.1 -d
IIO context has 5 devices:
        iio:device3: cf-ad9361-dds-core-lpc, found 0 device attributes
        iio:device1: ad9361-phy, found 18 device attributes
        iio:device4: cf-ad9361-lpc, found 0 device attributes
        iio:device2: xadc, found 1 device attributes
        iio:device0: adm1177, found 0 device attributes
```

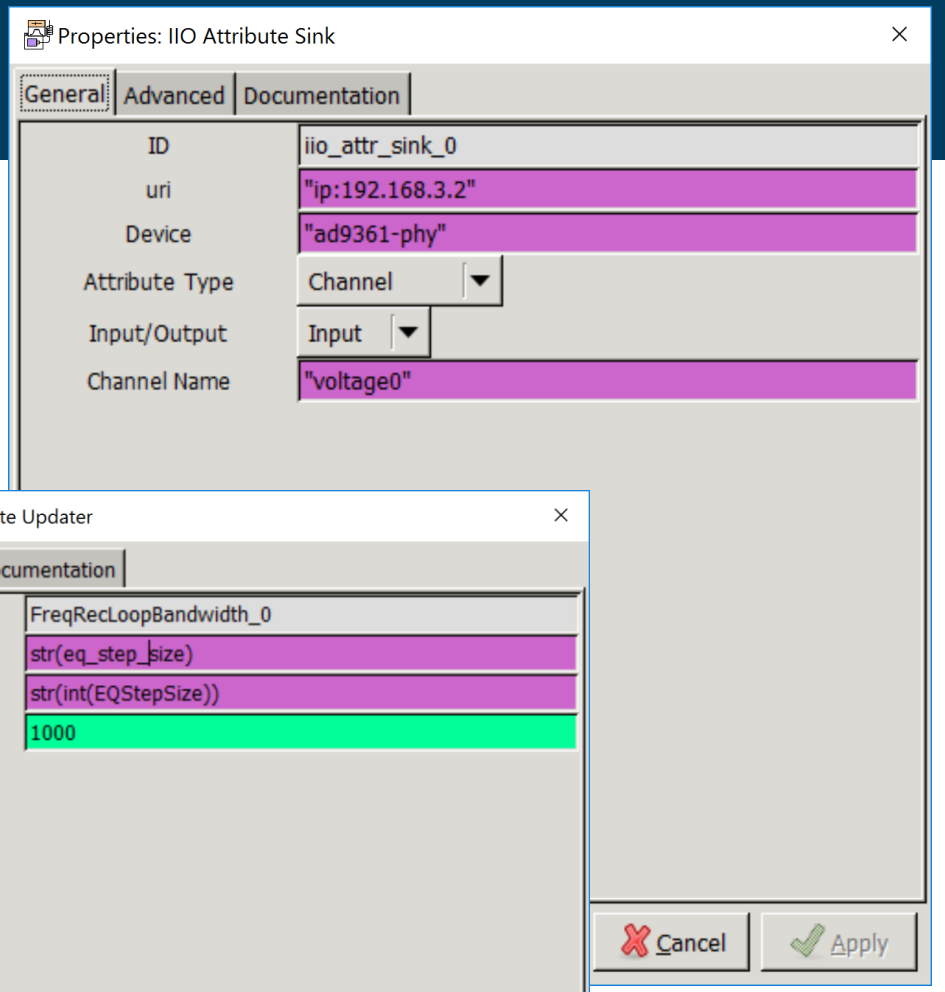


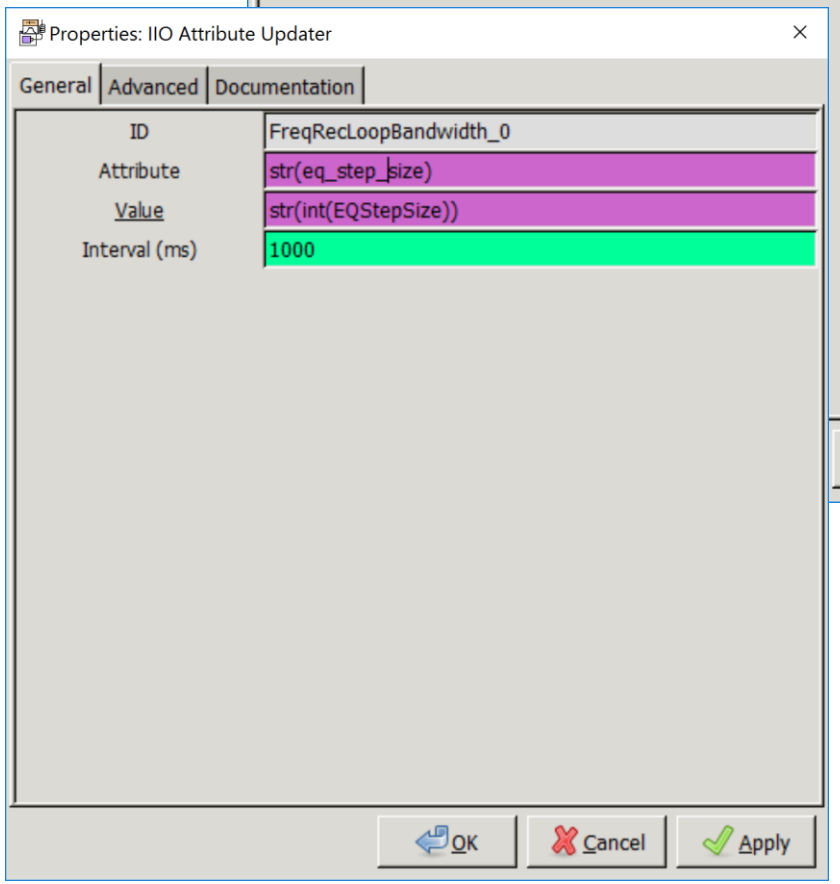


Using Attribute Sink

```
import pmtkey0 = pmt.intern("rf_bandwidth")
val0 = pmt.intern("23000000")
msg_dic = pmt.make_dict()
msg_dic = pmt.dict_add(msg_dic, key0, val0)
```

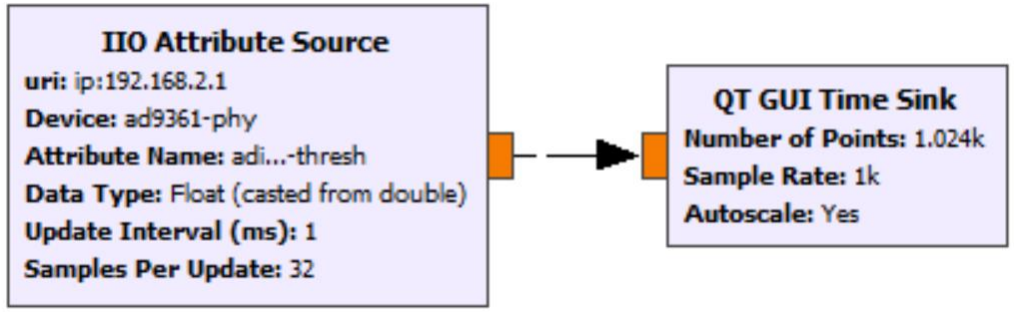


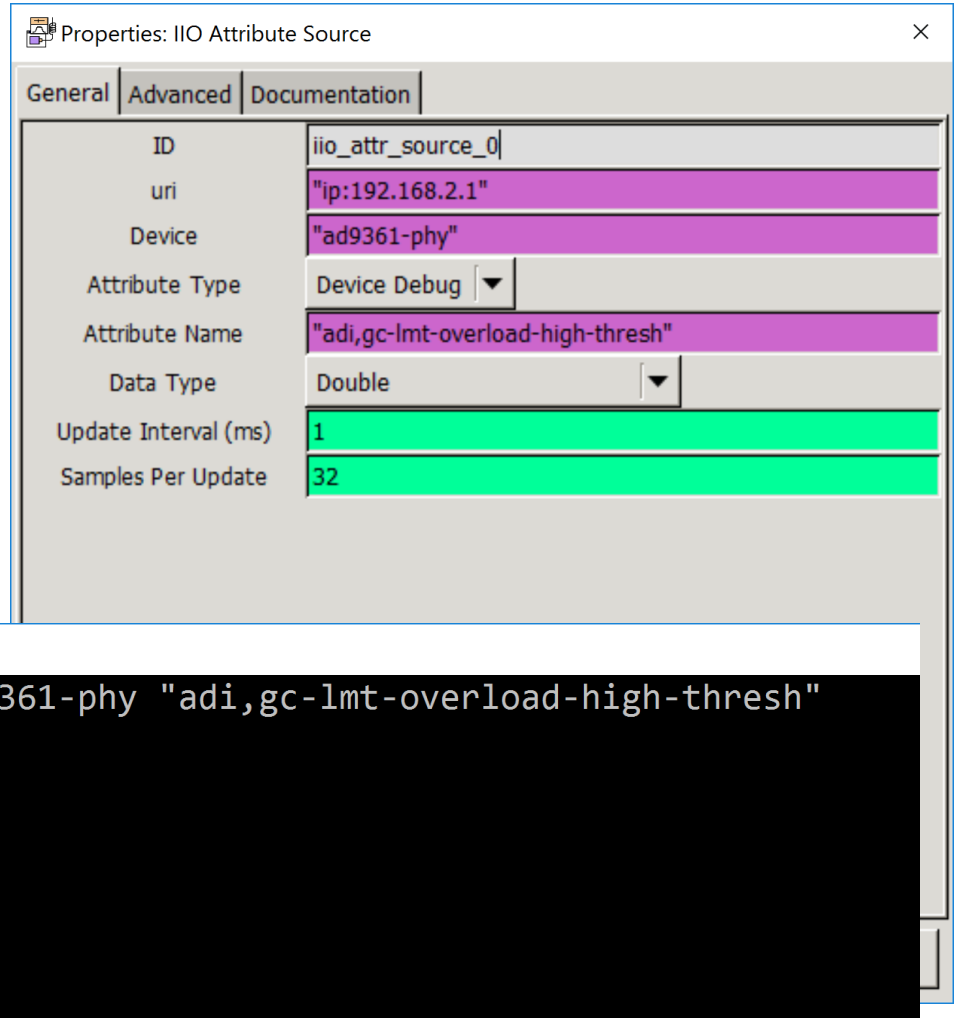


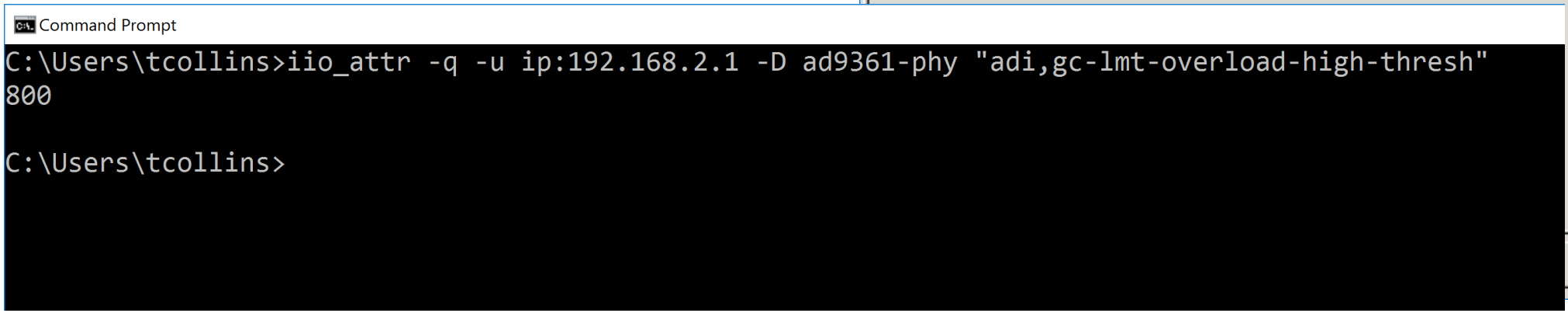


ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

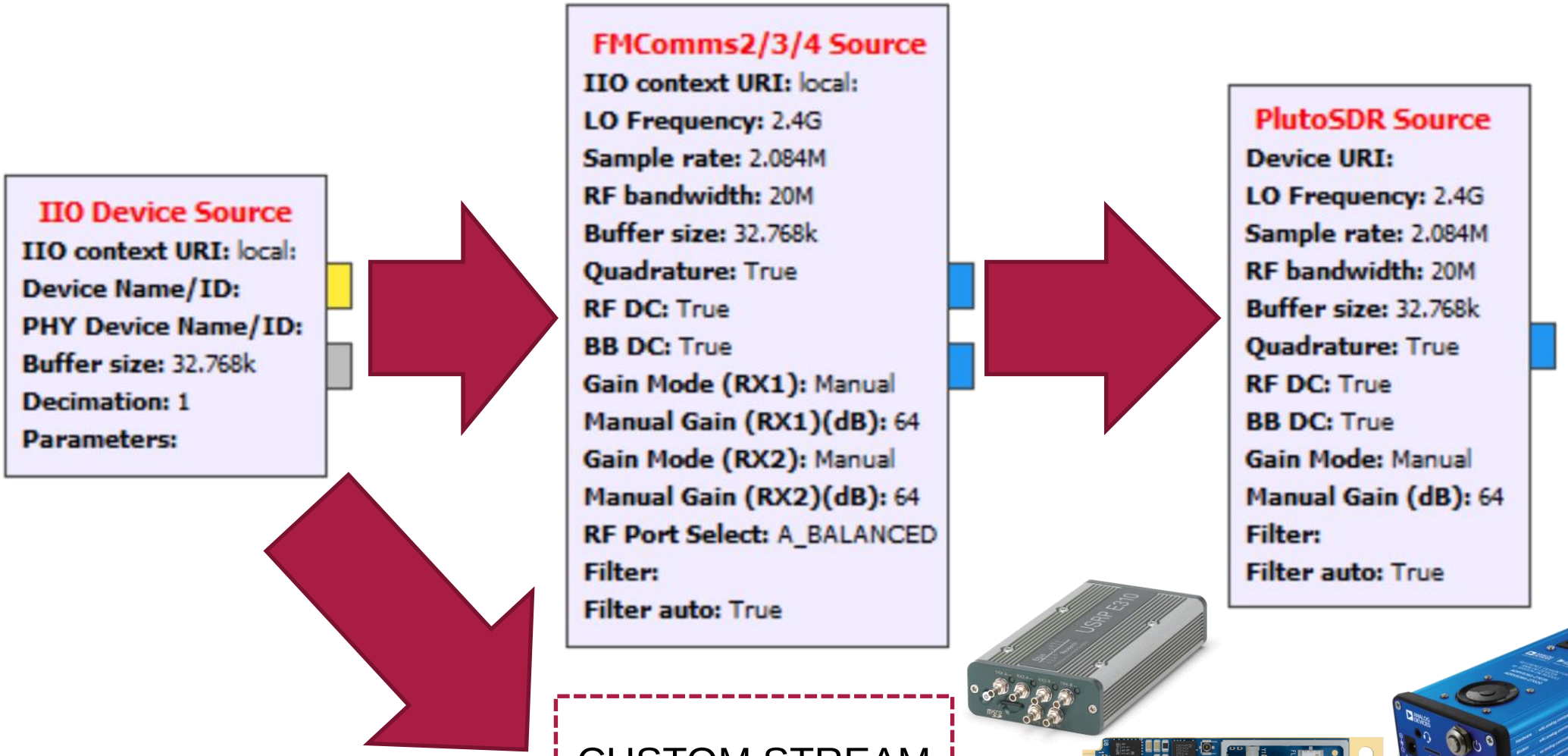
Using Attribute Source







Hardware Support Through IIO



ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

Lab: GNU Radio + gr-iio

ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

FMC Carrier Demo with GNU Radio

Thank you!

- For future support:
  - http://ez.analog.com
  - http://wiki.analog.com