gr-iio: Nuances, Advanced Features, and New Stuff

Travis Collins, PhD <travis.collins@analog.com>

AHEAD OF WHAT'S POSSIBLE™

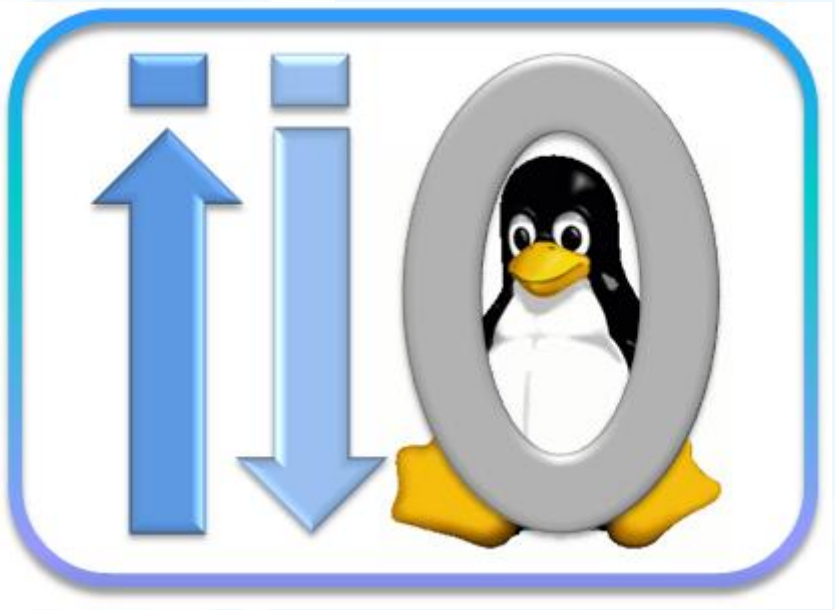
Outline

- Intro to IIO, libIIO, and gr-iio
- libIIO isms in GNU Radio
- Accessing Custom IP
- libIIO Performance
- New Features of gr-iio
- gr-iio GREP

Intro to IIO, libIIO, gr-iio

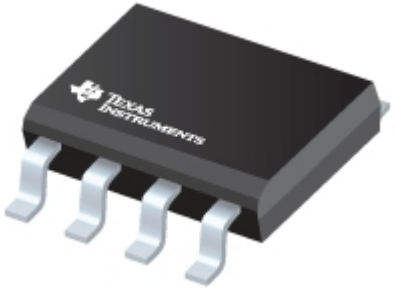
Industrial Input/Output (IIO) Framework

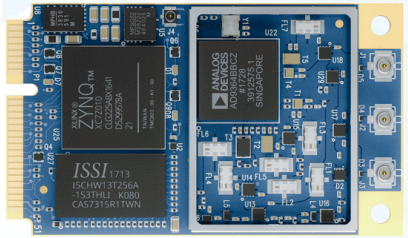
- Intended to provide support for ADCs, DACs, sensors …
- Devices that fall into this category are:
  - ADCs
  - Accelerometers
  - Gyros
  - IMUs
  - Capacitance to Digital Converters (CDCs)
  - Pressure Sensors
  - Color, Light and Proximity Sensors
  - Temperature Sensors
  - MagnetometersDACs
  - DDS (Direct Digital Synthesis)
  - PLLs (Phase Locked Loops)
  - Variable/Programmable Gain Amplifiers (VGA, PGA)
- Device which don't fit into Hwmon or the Input Framework



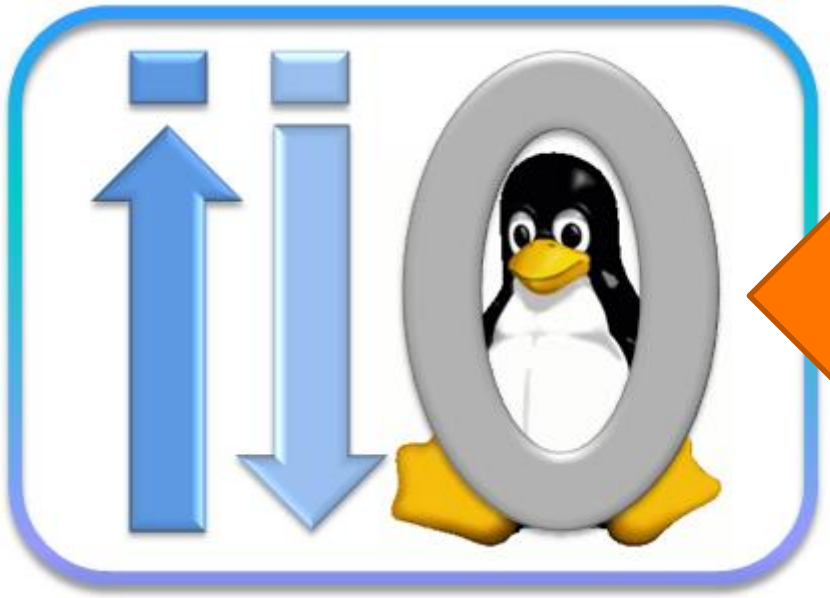
- Many vendor build IIO drivers
  - Analog Devices
  - AMS
  - STMicroelectronics
  - Maxim
  - Murata
  - Qualcomm
  - Bosch
  - Texas Instruments
  - Xilinx
  - …

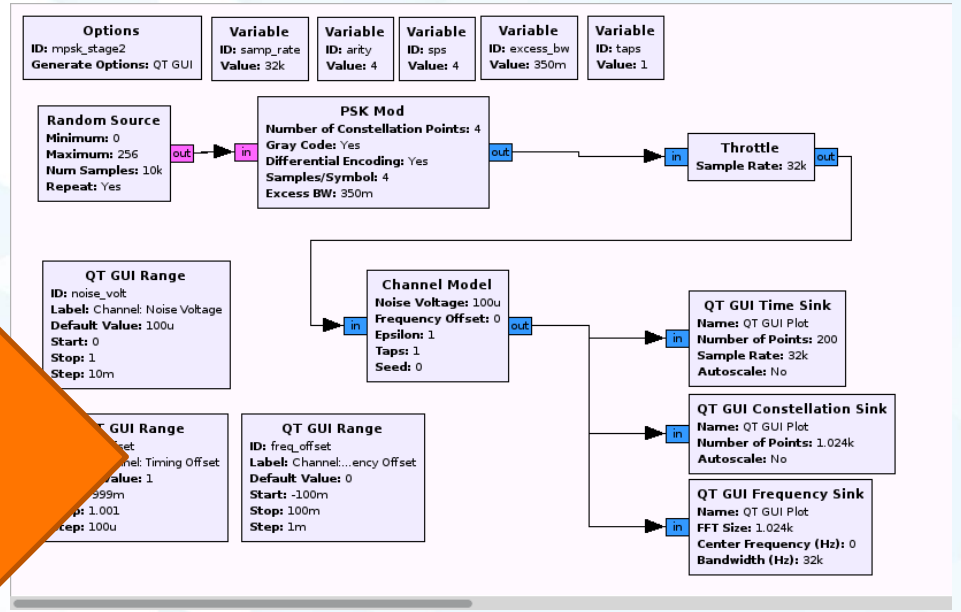
Access Sensors and Other Devices





IIO Driver To GNU Radio?

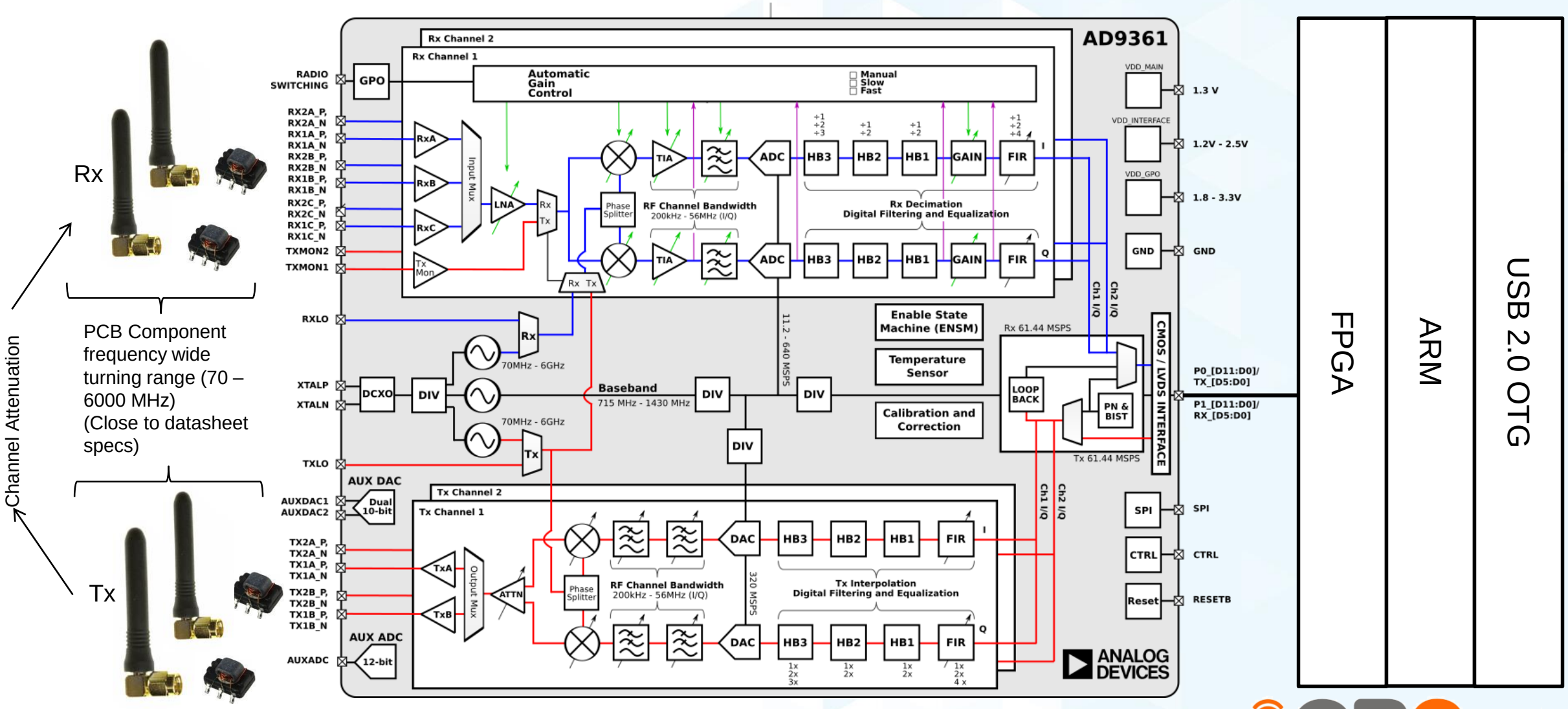




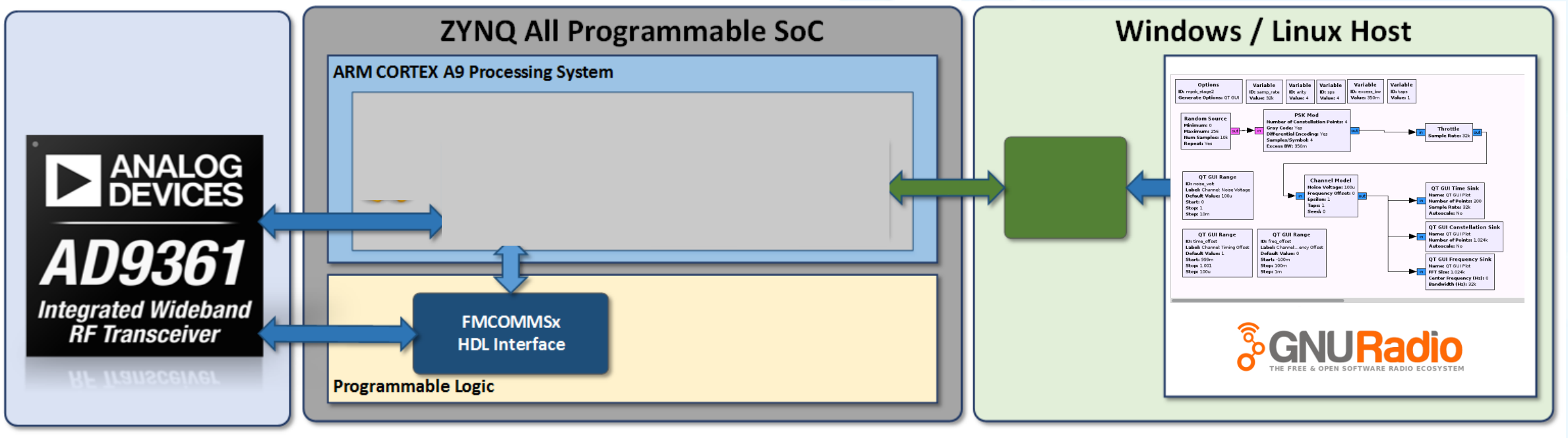
GNU Radio
THE FREE & OPEN SOFTWARE RADIO ECOSYSTEM

GRCon

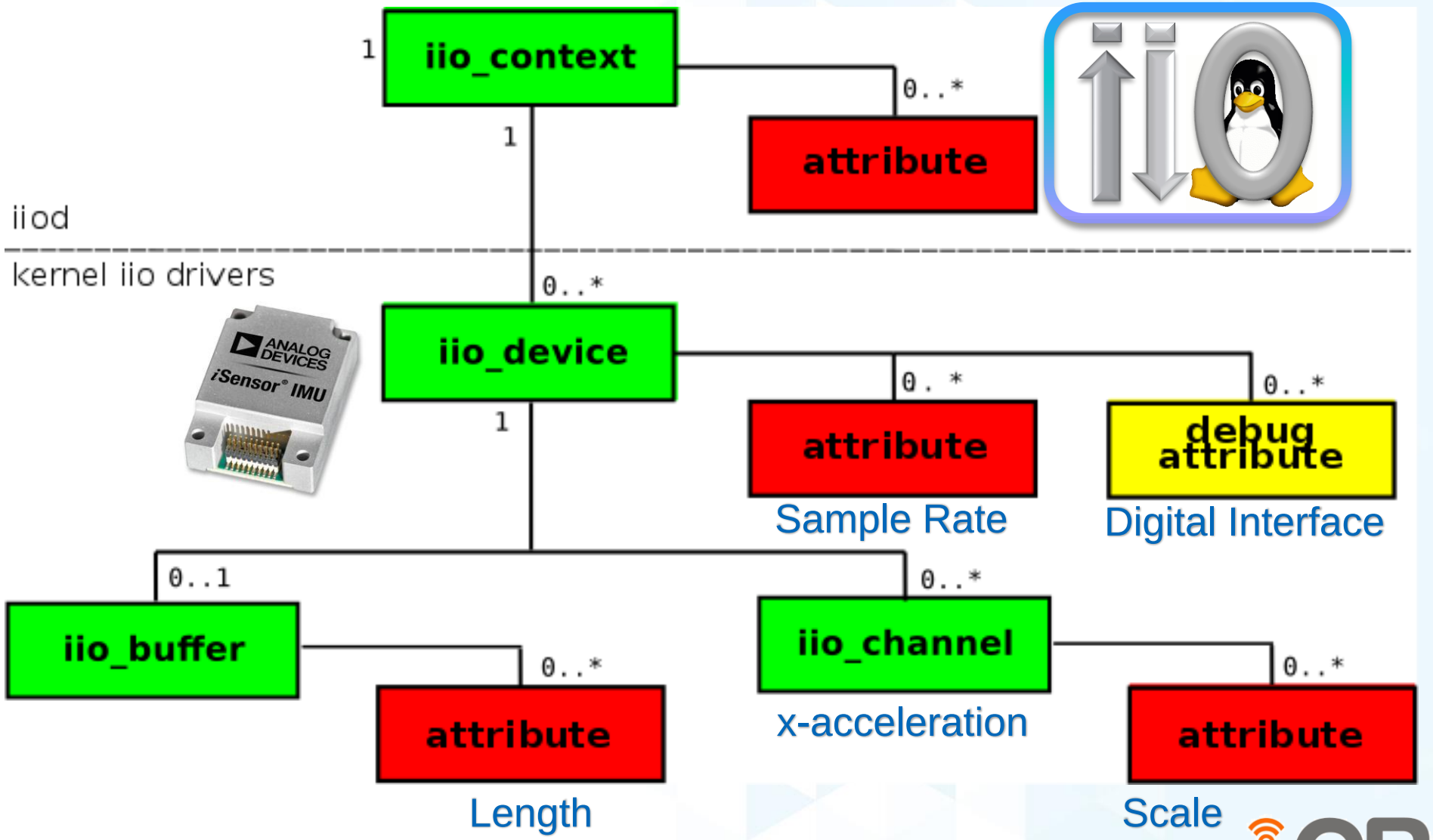
ADI ZIF Transceivers



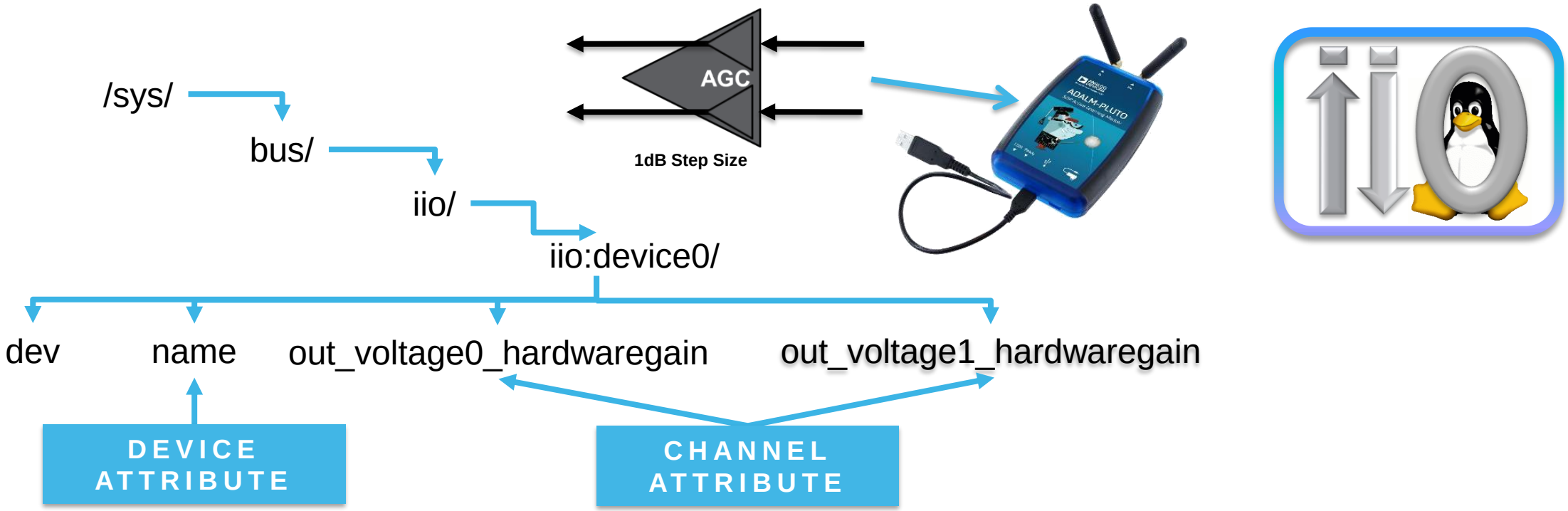
Radio to Host Interface



IIO Concepts



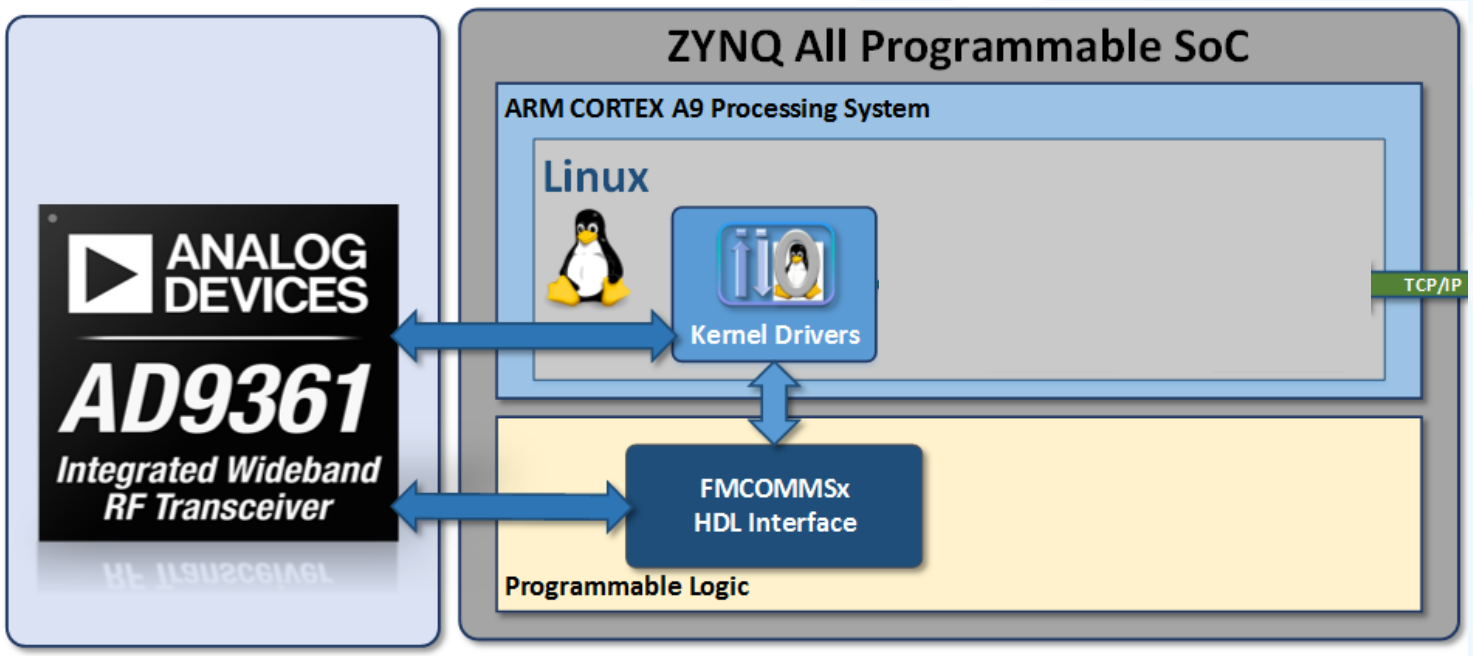
Pluto Gain Control



Shell Commands:

```
/sys/bus/iio/iio:device0 # cat name
ad8366-lpc
/sys/bus/iio/iio:device0 # echo 6 > out_voltage1_hardwaregain
/sys/bus/iio/iio:device0 # cat out_voltage1_hardwaregain
5.765000 dB
```

IIO Driver



Goal: How can I control the device?

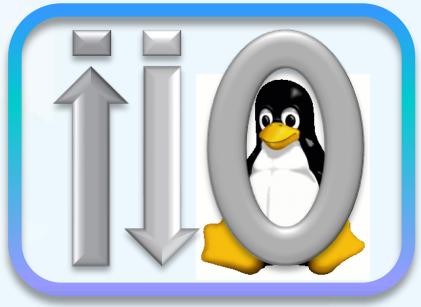
LO Frequency

Sample Rate

Gain Mode

TDD ENSM

RSSI



192.168.1.199 - PuTTY

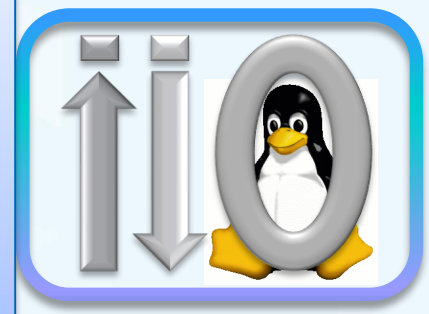
```
# ls
calib_mode                          in_voltage_rf_bandwidth                 out_voltage0_hardwaregain
calib_mode_available                in_voltage_rf_bandwidth_available       out_voltage0_hardwaregain_available
dcxo_tune_coarse                    in_voltage_rf_dc_offset_tracking_en     out_voltage0_rf_port_select
dcxo_tune_coarse_available          in_voltage_rf_port_select_available     out_voltage0_rssi
dcxo_tune_fine                      in_voltage_sampling_frequency           out_voltage2_raw
dcxo_tune_fine_available            in_voltage_sampling_frequency_available out_voltage2_scale
dev                                 multichip_sync                          out_voltage3_raw
ensm_mode                           name                                    out_voltage3_scale
ensm_mode_available                 of_node                                 out_voltage_filter_fir_en
filter_fir_config                   out_altvoltage0_RX_LO_external          out_voltage_rf_bandwidth
gain_table_config                   out_altvoltage0_RX_LO_fastlock_load     out_voltage_rf_bandwidth_available
in_out_voltage_filter_fir_en        out_altvoltage0_RX_LO_fastlock_recall   out_voltage_rf_port_select_available
in_temp0_input                      out_altvoltage0_RX_LO_fastlock_save     out_voltage_sampling_frequency
in_voltage0_gain_control_mode       out_altvoltage0_RX_LO_fastlock_store    out_voltage_sampling_frequency_available
in_voltage0_hardwaregain            out_altvoltage0_RX_LO_frequency         power
in_voltage0_hardwaregain_available  out_altvoltage0_RX_LO_frequency_available rssi_gain_step_error
in_voltage0_rf_port_select          out_altvoltage0_RX_LO_powerdown         rx_path_rates
in_voltage0_rssi                    out_altvoltage1_TX_LO_external          subsystem
in_voltage2_offset                  out_altvoltage1_TX_LO_fastlock_load     trx_rate_governor
in_voltage2_raw                     out_altvoltage1_TX_LO_fastlock_recall   trx_rate_governor_available
in_voltage2_scale                   out_altvoltage1_TX_LO_fastlock_save     tx_path_rates
in_voltage_bb_dc_offset_tracking_en out_altvoltage1_TX_LO_fastlock_store    uevent
in_voltage_filter_fir_en            out_altvoltage1_TX_LO_frequency         xo_correction
in_voltage_gain_control_mode_available out_altvoltage1_TX_LO_frequency_available xo_correction_available
in_voltage_quadrature_tracking_en   out_altvoltage1_TX_LO_powerdown
#
```

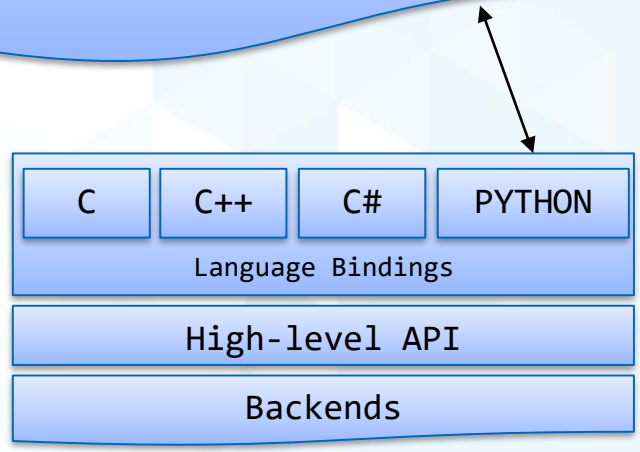
IIO – libiio

- System library
- Abstracts away low level details of the IIO kernel ABI
  - Kernel ABI is designed to be simple and efficient
  - libiio focuses on ease of use
- Provides high-level C, C++, C# or Python programming interface to IIO (Language bindings)

```
ctx = iio_create_context_from_uri("ip:192.168.2.1");

phy = iio_context_find_device(ctx, "ad9361-phy");

iio_channel_attr_write_longlong(
        iio_device_find_channel(phy, "altvoltage0", true),
        "frequency",
        2400000000); /* RX LO frequency 2.4GHz */

iio_channel_attr_write_longlong(
        iio_device_find_channel(phy, "voltage0", false),
        "sampling_frequency",
        5000000); /* RX baseband rate 5 MSPS */
```

```
import iio

ctx = iio.Context('ip:192.168.2.1')

phy = ctx.find_device('ad9361-phy')
```





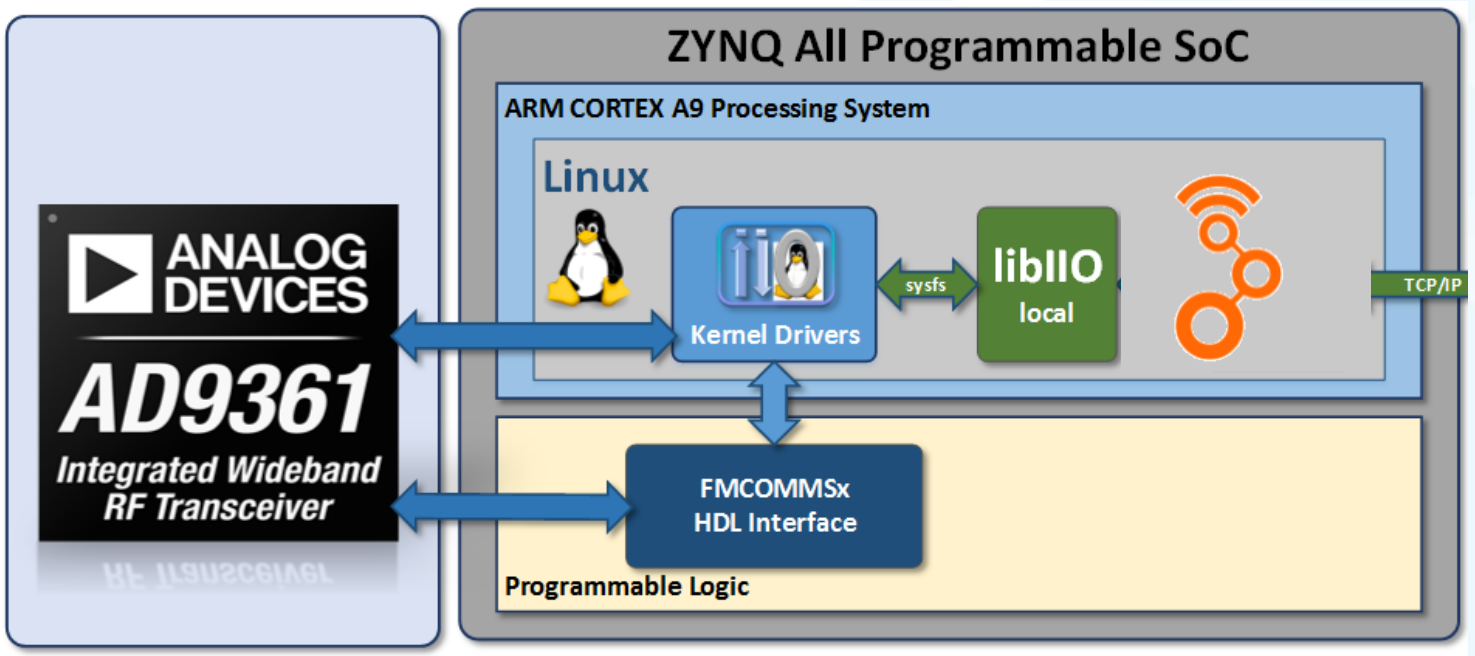
For more information:

https://github.com/analogdevicesinc/libiio

http://wiki.analog.com/resources/tools-software/linux-software/libiio_internals

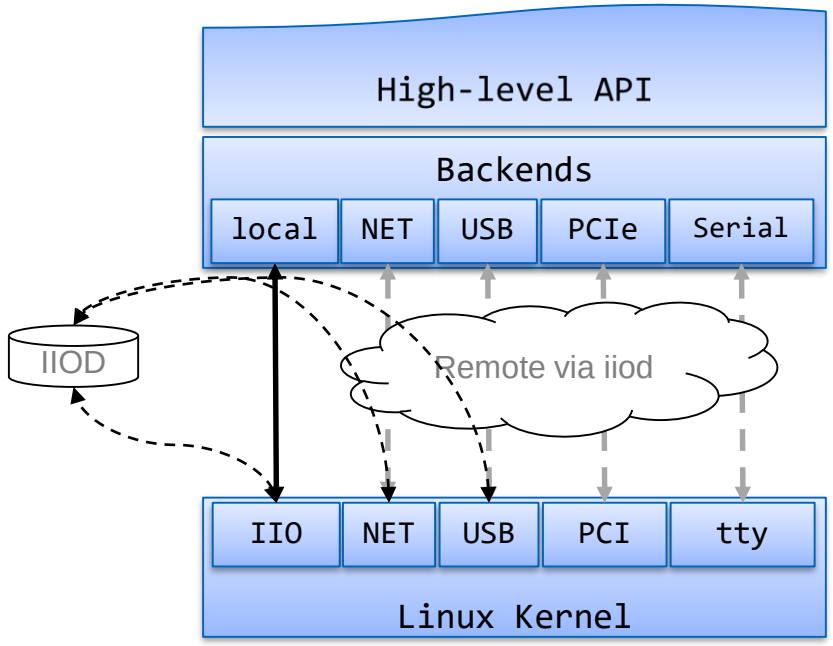
http://analogdevicesinc.github.io/libiio/

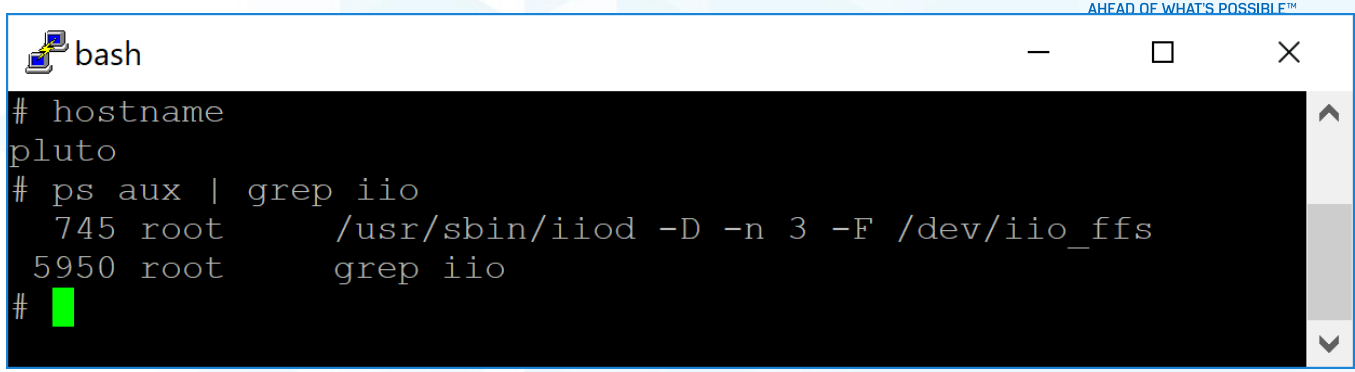
IIO Driver

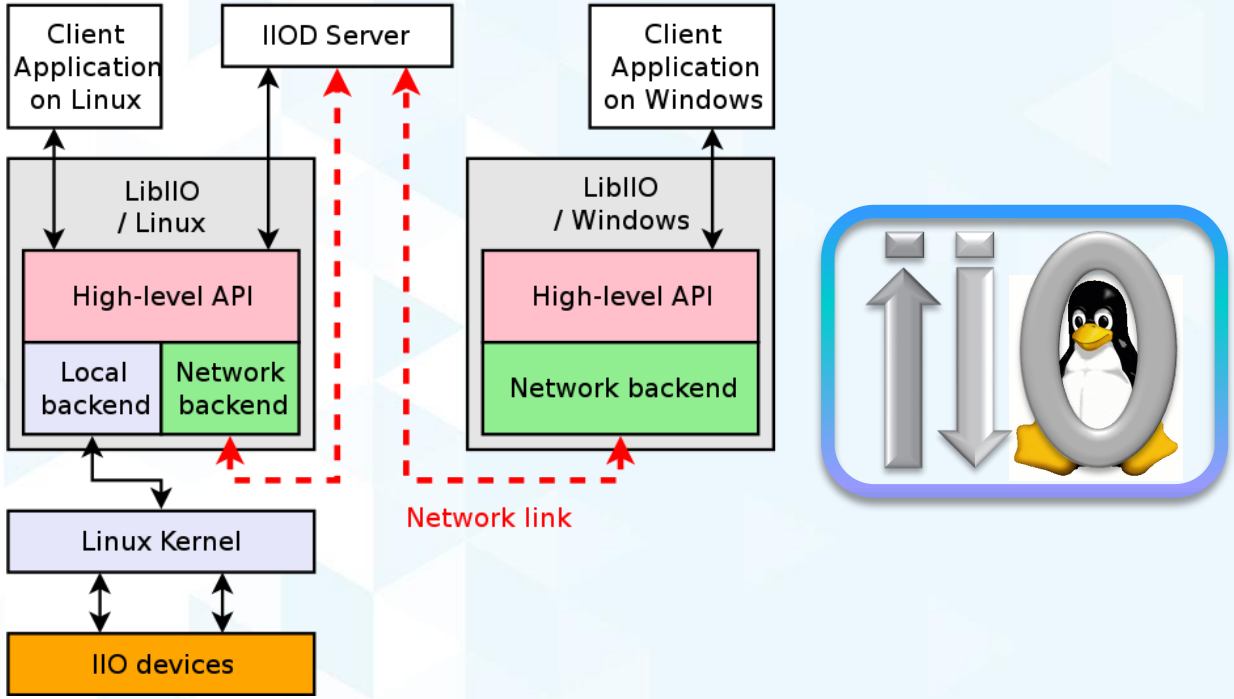


IIO – libiio – Backends

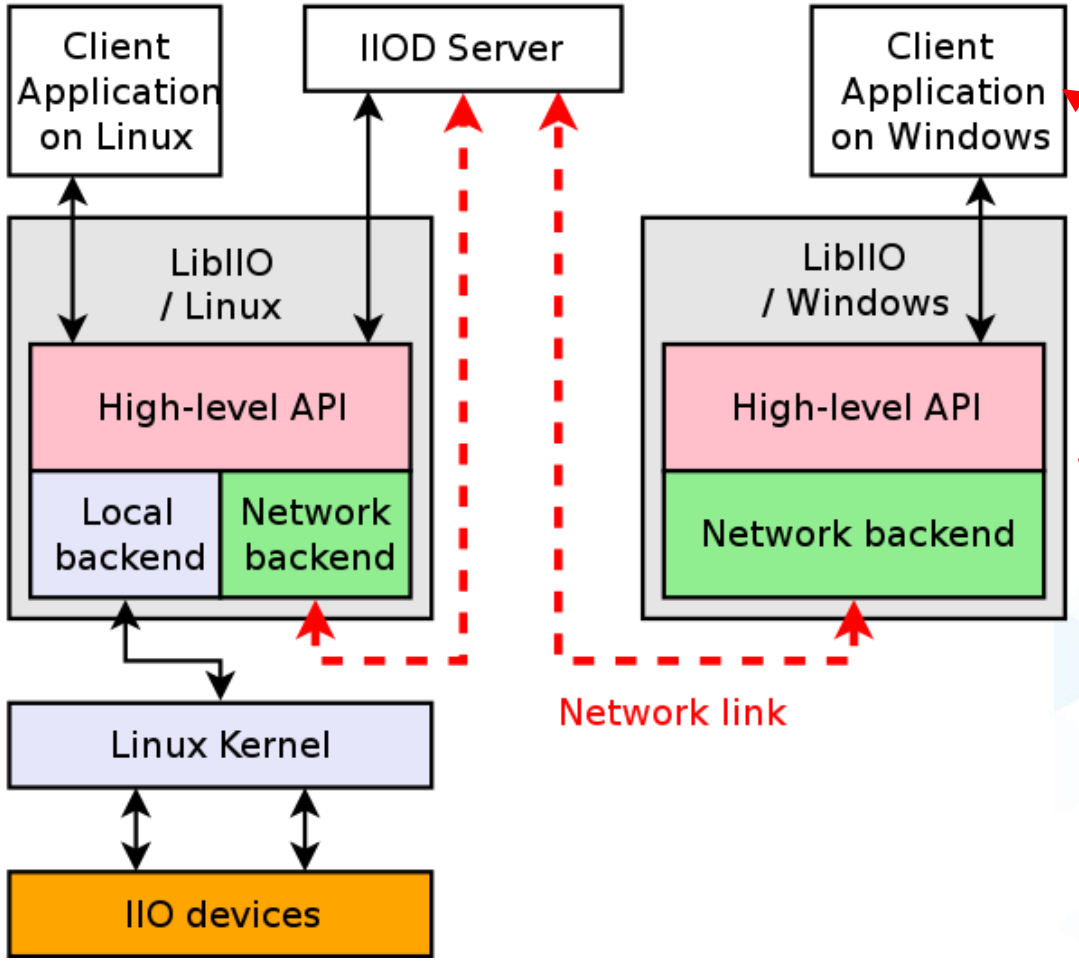
- Support for backends
  - Backend takes care of low-level communication details
  - Provide the same API for applications
  - Transparent from the applications point of view







libIIO and applications

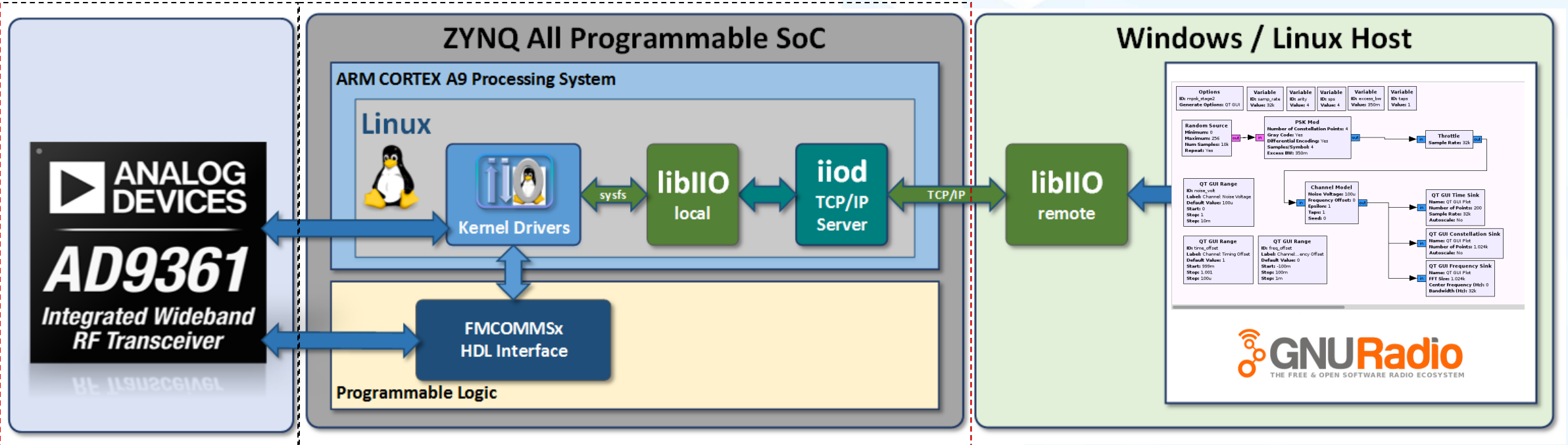


- iio-utils (command line tools)
- IIO-Scope
- MATLAB and Simulink
- GNU Radio
- 3rd part apps
  - SDR Angel
  - GQRX
  - SDR#
  - …

- Language support for libIIO
  - C/C++
  - Python
  - Nodejs (3rd party)
  - C# (3rd part)

IIO Driver

ADALM-PLUTO,
ADRV9361-Z7035 RF SOM, ADRV9364-Z7020 RF SOM, ADRV9009-ZU11EG RF SOM

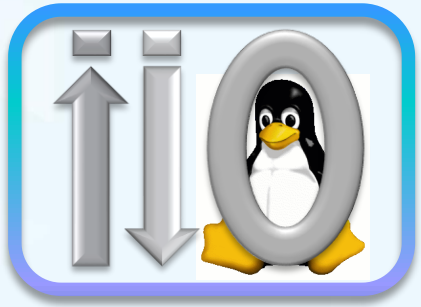


FMCOMMS2, FMCOMMS3,
FMCOMMS4, FMCOMMS5,
ADRV9371 ADRV9375
ADRV9009

Xilinx Zed Board, Xilinx ZC702, Xilinx ZC706,
Xilinx KC705, Xilinx KCU105,
Xilinx ZCU102
Intel Arria 10 SoC, Intel Stratix 10 SoC

IIO – libiio – Command line tools

- **iio_info** : Information about all IIO devices, backends and context attributes
  - `iio_info -s`
  - `iio_info –u ip:192.168.2.1`



- **iio_attr** : Read and write IIO attributes
  - `iio_attr -c ad9361-phy altvoltage0 frequency 2450000000`

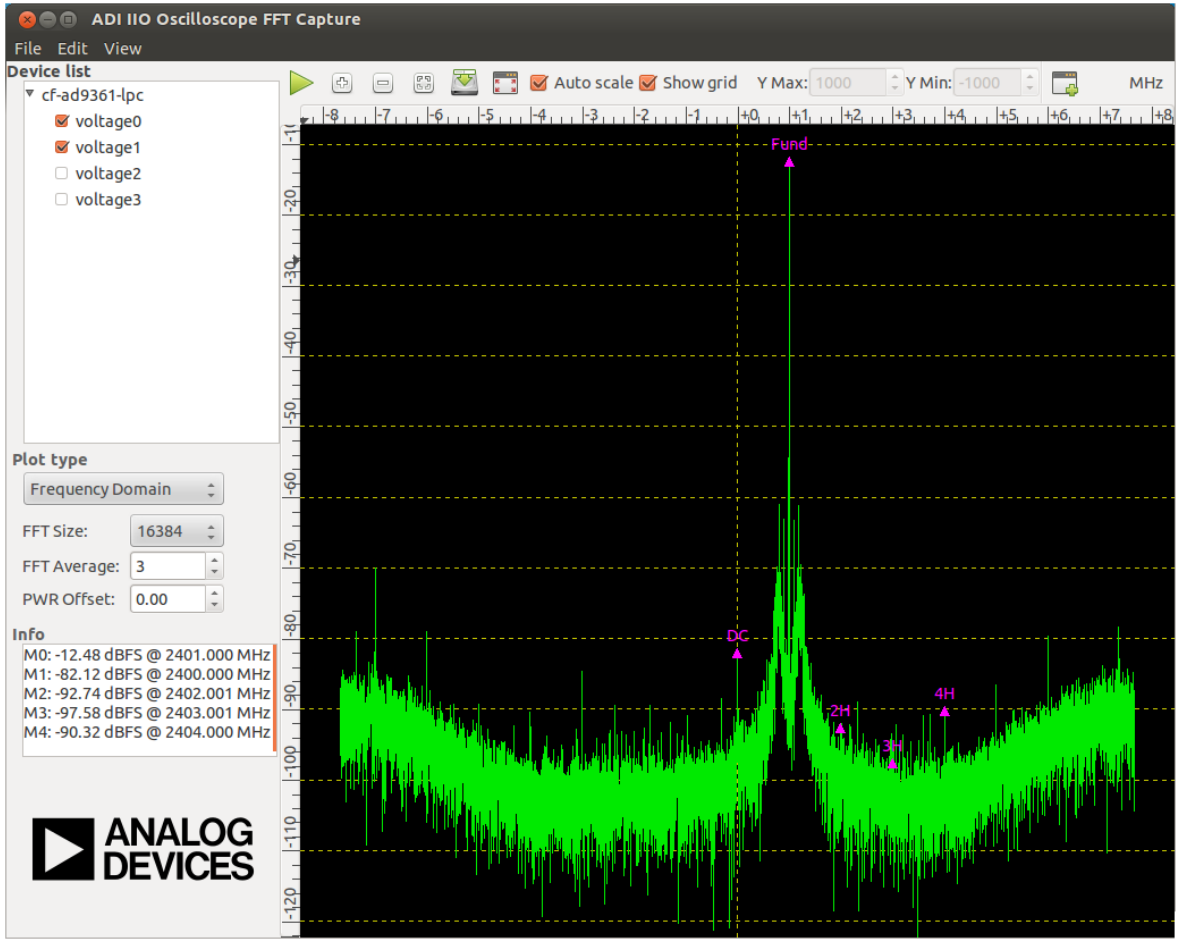
- **iio_readdev** : Read samples from an IIO device
  - `iio_readdev -u usb:1.100.5 -b 100000 cf-ad9361-lpc | pv > /dev/null`

- **iio_writedev** : Write samples to an IIO device
  - `iio_readdev -b 100000 cf-ad9361-lpc | iio_writedev -b 100000 cf-ad9361-dds-core-lpc`

- **iio_reg** : Read or write SPI or I2C registers in an IIO device (useful to debug drivers)
  - `iio_reg adrv9009-phy 0`

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
- Should support any IIO device
- Cross platform



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

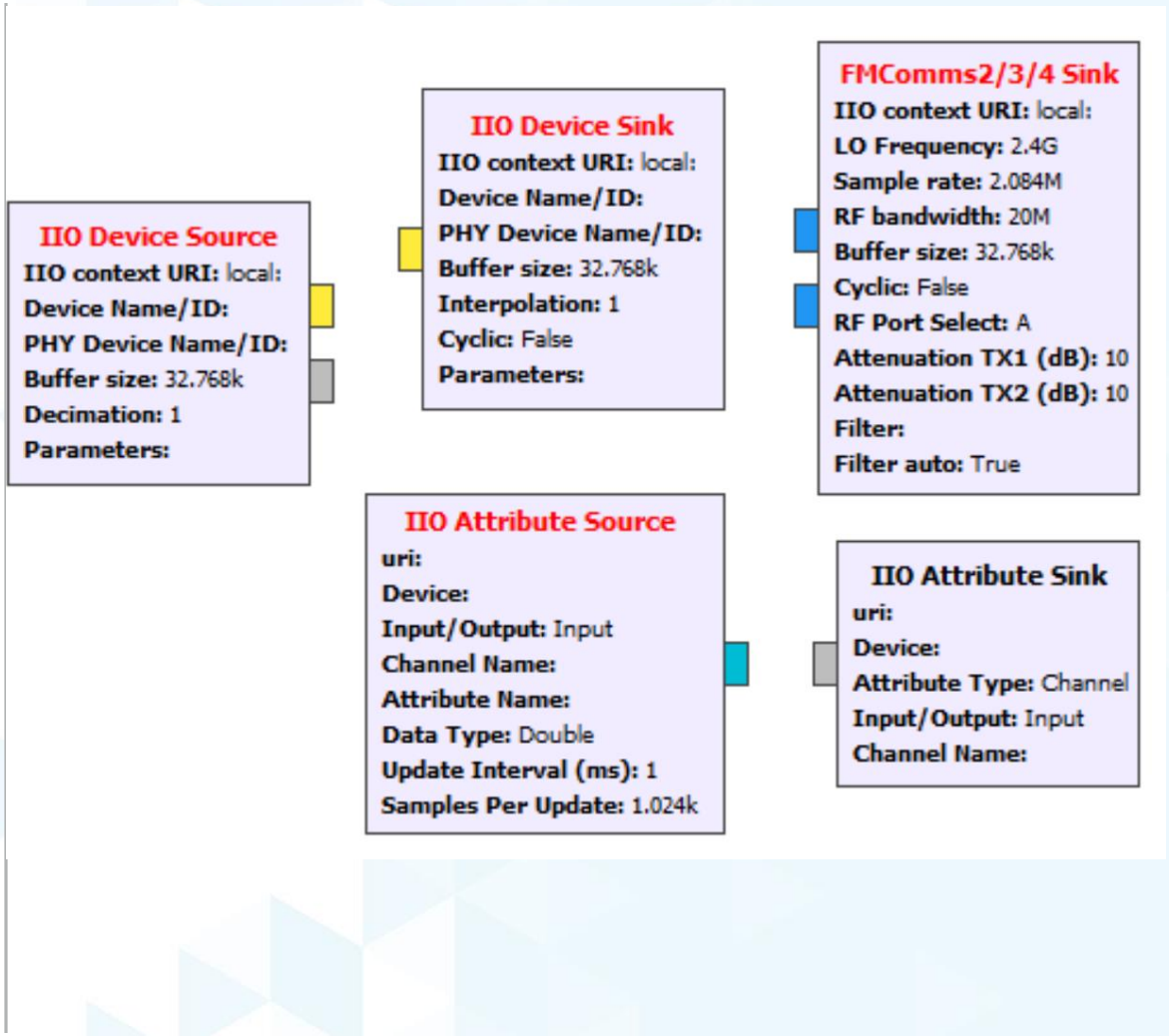
Attributes

Streams

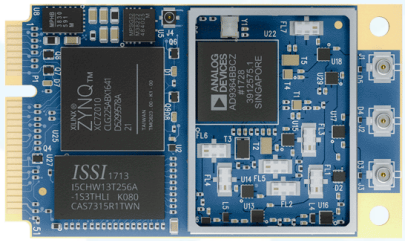
Math (Scopy)

SDR

Math (Scopy)



Hardware Support Through IIO



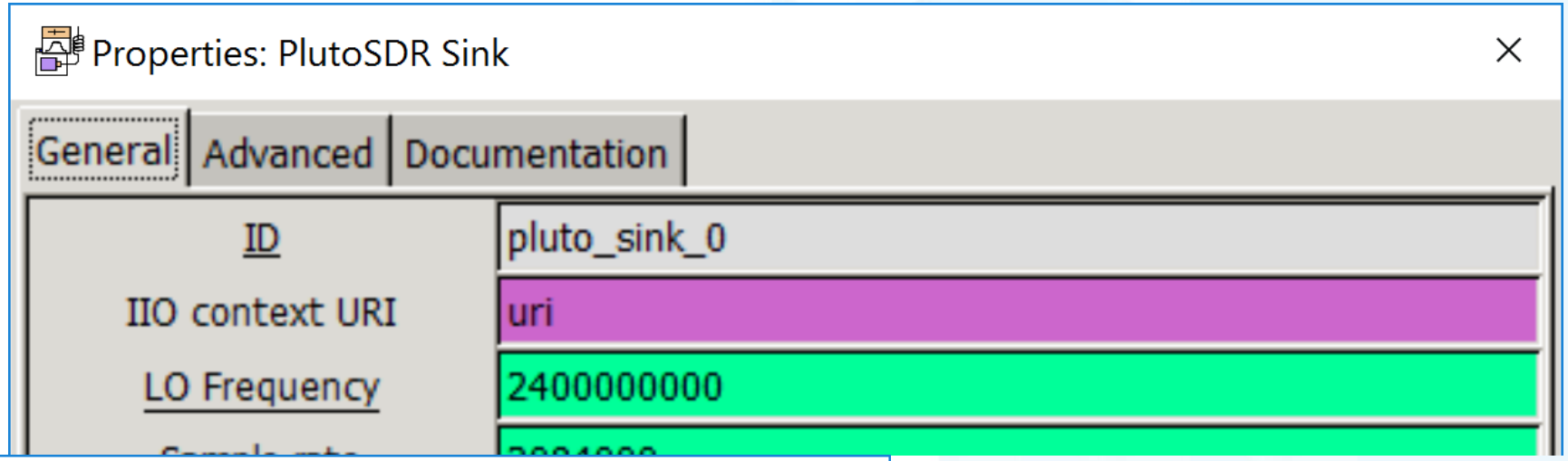
IIO Device Source
IIO context URI: local:
Device Name/ID:
PHY Device Name/ID:
Buffer size: 32.768k
Decimation: 1
Parameters:

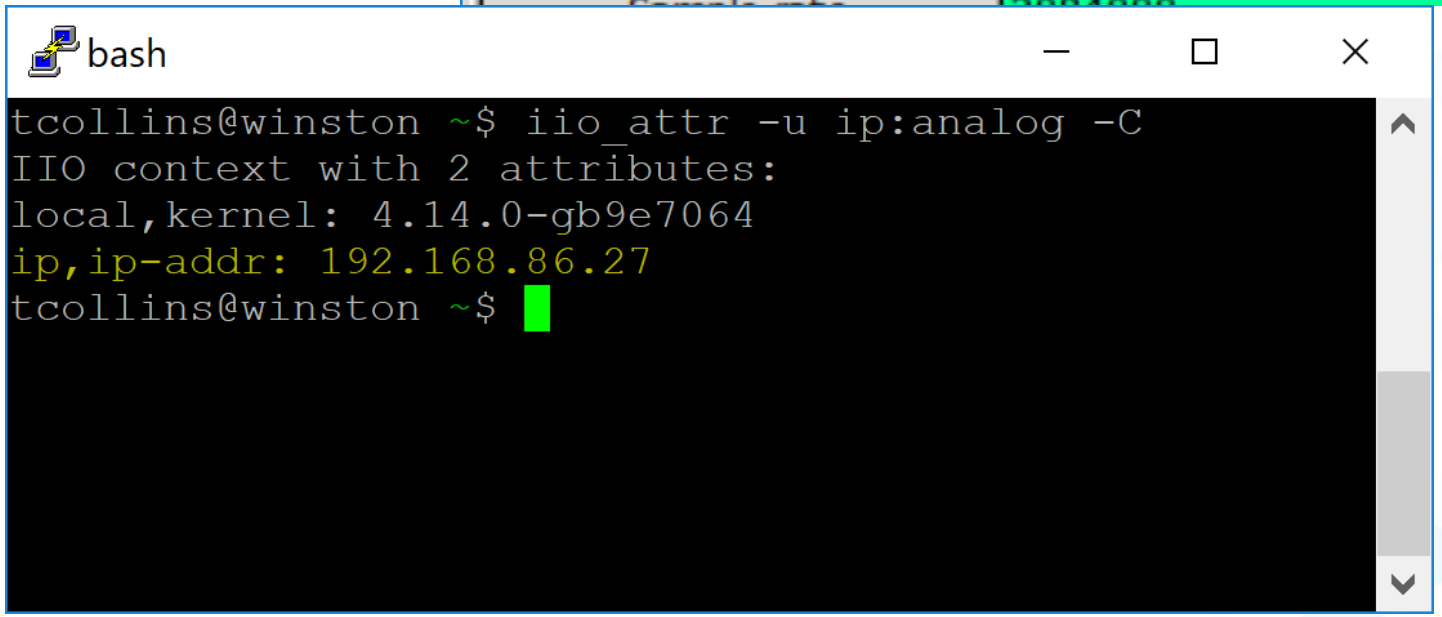
FMComms2/3/4 Source
IIO context URI: local:
LO Frequency: 2.4G
Sample rate: 2.084M
RF bandwidth: 20M
Buffer size: 32.768k
Quadrature: True
RF DC: True
BB DC: True
Gain Mode (RX1): Manual
Manual Gain (RX1)(dB): 64
Gain Mode (RX2): Manual
Manual Gain (RX2)(dB): 64
RF Port Select: A_BALANCED
Filter:
Filter auto: True

PlutoSDR Source
Device URI:
LO Frequency: 2.4G
Sample rate: 2.084M
RF bandwidth: 20M
Buffer size: 32.768k
Quadrature: True
RF DC: True
BB DC: True
Gain Mode: Manual
Manual Gain (dB): 64
Filter:
Filter auto: True

CUSTOM STREAM DEVICE

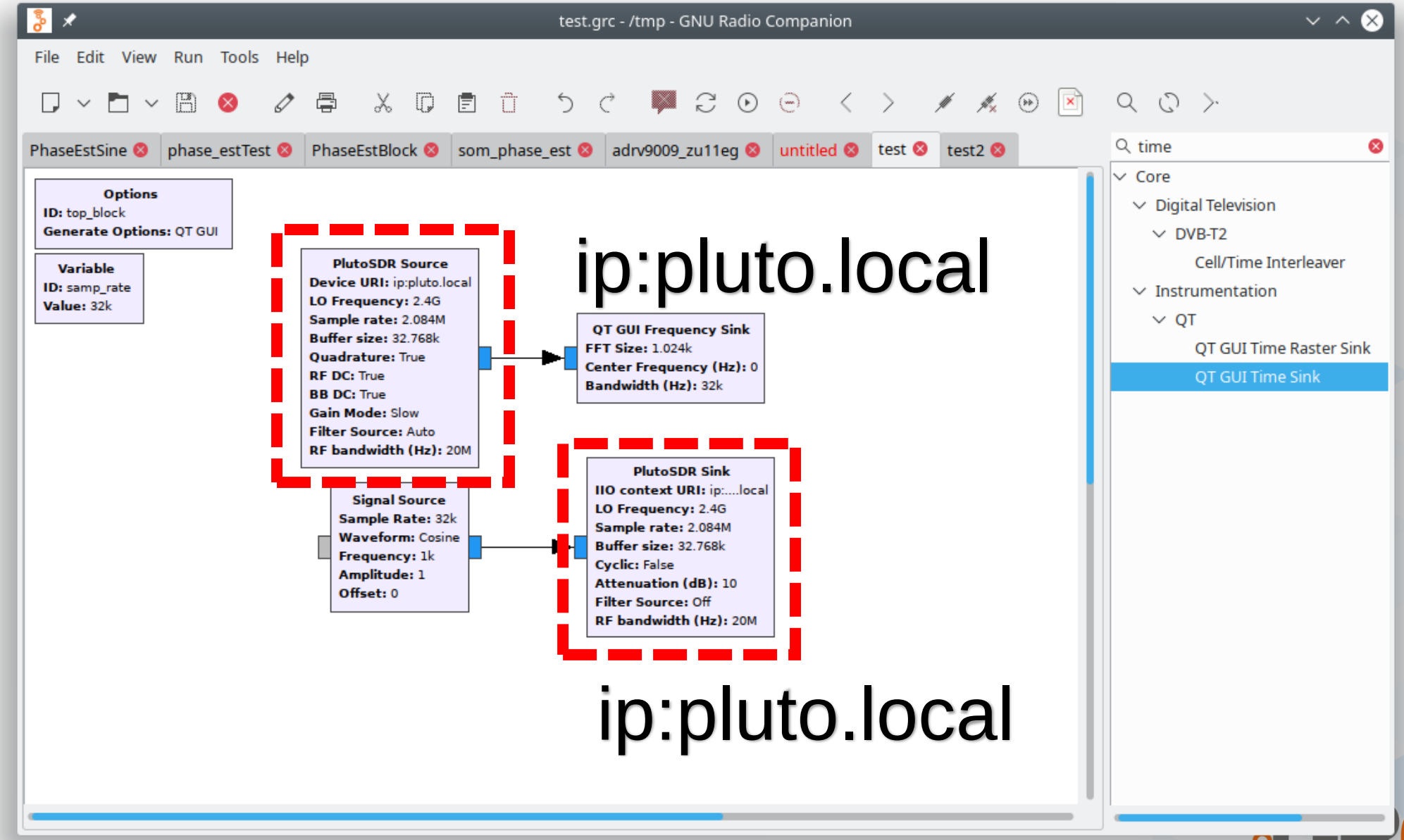
Contexts in context





- URIs are formatted as [Type]:[Address]
  - Network: **ip:192.168.2.1**
  - USB: **usb:1.3.5**
  - Local: **local:**
- Special
  - Local Autodiscover: **“blank”**
  - Network Autodiscover**: ip:<hostname>**
    - **ip:analog, ip:pluto**

Shared Contexts



Shared Contexts

test2.grc - /tmp - GNU Radio Companion

File Edit View Run Tools Help

PhaseEstSine | phase_estTest | PhaseEstBlock | som_phase_est | adrv9009_zu11eg

Options
ID: top_block
Generate Options: QT GUI

ip:analog

Variable
ID: samp_rate
Value: 32k

FMComms2/3/4 Source
IIO context URI: ip:analog
LO Frequency: 2.4G
Sample rate: 2.084M
Buffer size: 32.768k
Quadrature: True
RF DC: True
BB DC: True
Gain Mode (RX1): Slow
RF Port Select: A_BALANCED
Filter Source: Auto
RF bandwidth (Hz): 20M

QT GUI Frequency Sink
FFT Size: 1.024k
Center Frequency (Hz):
Bandwidth (Hz): 32k

IIO Device Source
IIO context URI: ip:analog
Device Name/ID: adis16460
PHY Device Name/ID: adis16460
Buffer size: 128
Decimation: 1
Parameters:

Short To Float
Scale: 1

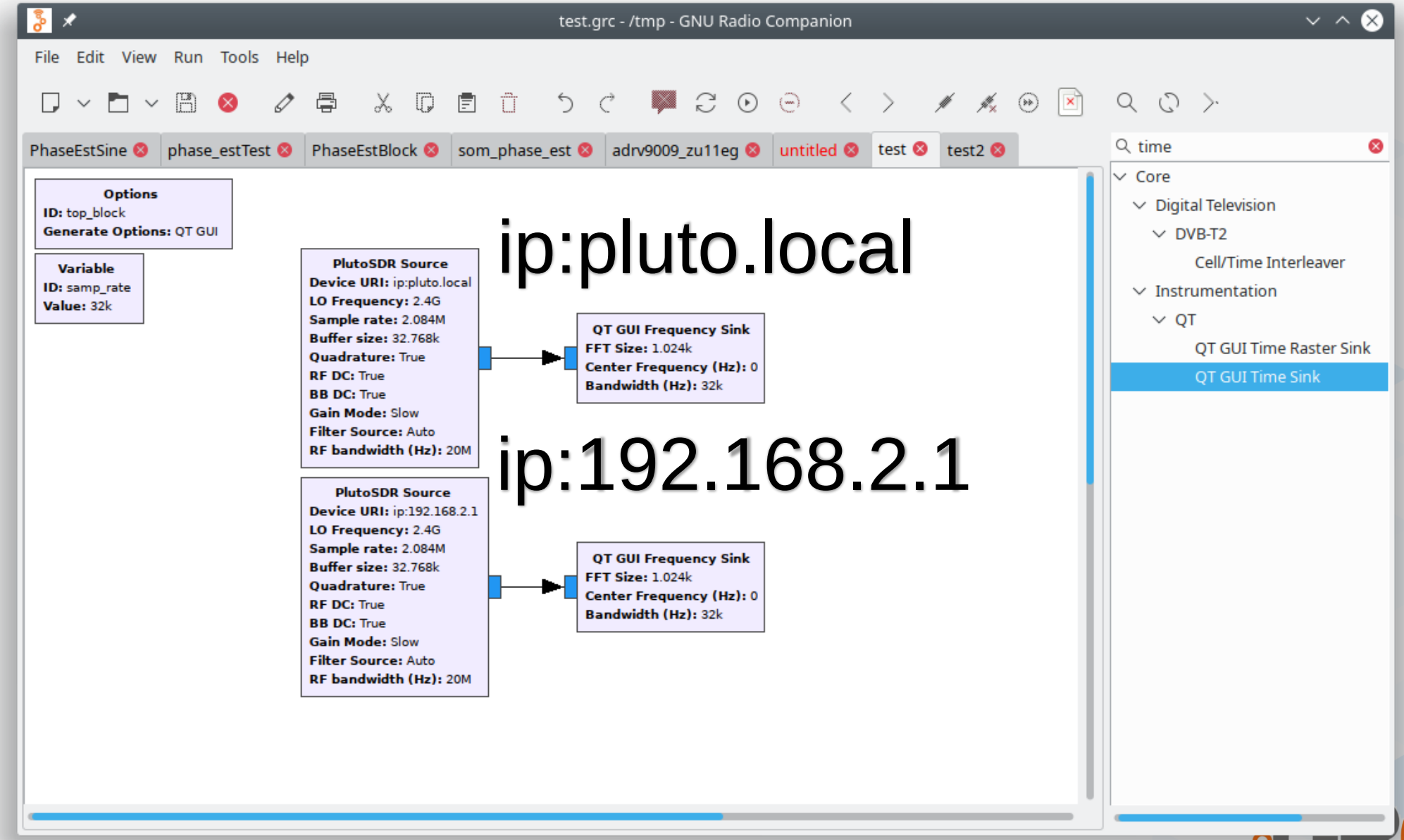
QT GUI Time Sink
Number of Points: 1.024k
Sample Rate: 32k
Autoscale: Yes

ip:analog

bash

```
tcollins@winston ~$ iio_attr -u ip:analog -d
IIO context has 8 devices:
        iio:device0: ad7291-ccbox, found 0 device attributes
        iio:device1: ad9361-phy, found 18 device attributes
        iio:device2: xadc, found 1 device attributes
        iio:device3: cf-ad9361-dds-core-lpc, found 0 device attributes
        iio:device4: adis16460, found 2 device attributes
        iio:device5: cf-ad9361-lpc, found 0 device attributes
        iio_sysfs_trigger:, found 2 device attributes
        trigger0: adis16460-dev4, found 0 device attributes
tcollins@winston ~$
```

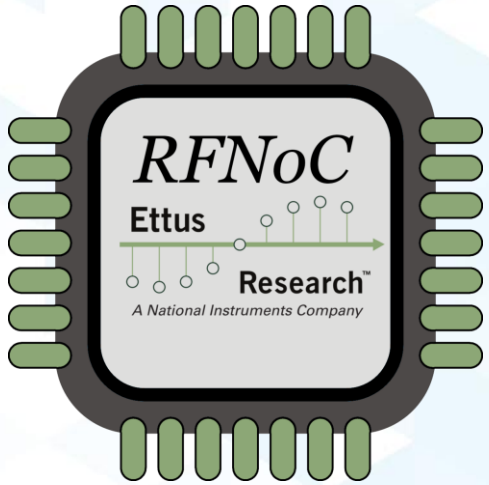
Separate Contexts

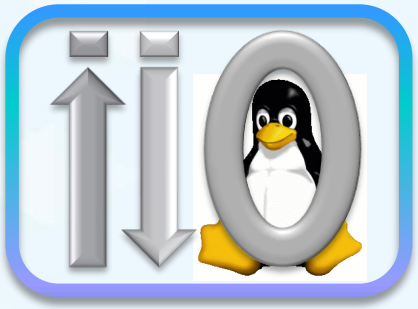


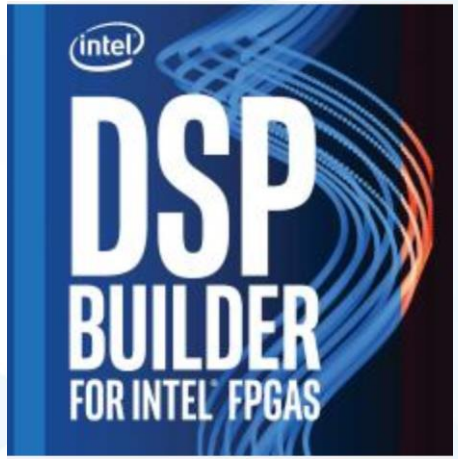
Custom IP

Why should I care?

- Offload heaving processing tasks
- Tight timing control over hardware
  - Frequency hopping
  - Radar processing
  - Time division MAC
- Going embedded
- Standard FPGA tools do not provide great software control mechanisms







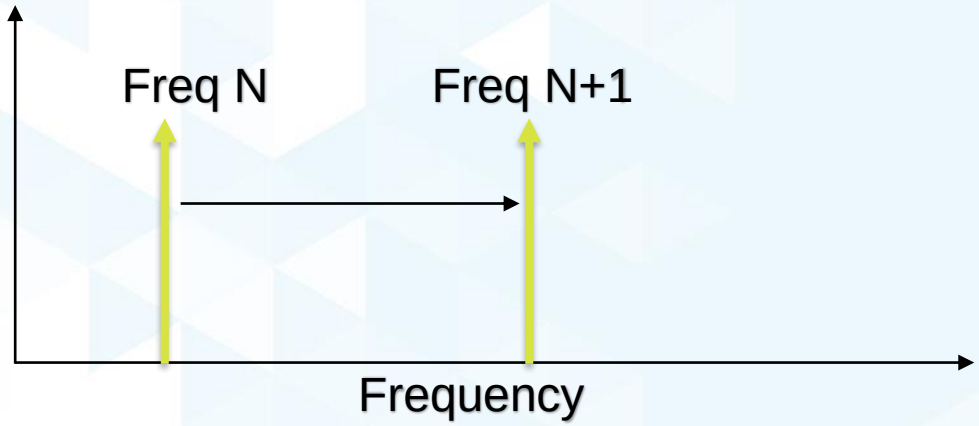
Example: Can the AD936X frequency hop?

- Paper at 2016 GRCon
  - BELL, Richard. Maximum Supported Hopping Rate Measurements using the Universal Software Radio Peripheral Software Defined Radio. **Proceedings of the GNU Radio Conference**, [S.l.], v. 1, n. 1, sep. 2016.
- Technique
  - Start hop:
    - tb.usrp.setcenterfreq(targetfreq)
  - Check Settling:
    - tb.usrp.get_sensor("lo_locked")
- Using this API can be as bad as 100 ms

MB	DB	$t_{tune}+t_{lock}$ (μs)	t_{lock} (μs)
N210	WBX	510	290
N210	SBX	483	289
X310	UBX	610	418
B100	WBX	782	573
B100	SBX	772	546
B200[1]	-[4]	3310	117
B210[1]	-[4]	3318	124
E310[2]	-[4]	31258	285
HackRF [3]	-[4]	215	0

AD936X: Fast Frequency Hopping

- Part capable of ~25us LO changes
  - To go faster requires external LO
- Fast hopping named by using “Fastlock Profiles”
- Profile selection possible through SPI or GPIO
  - Limited to 8 profiles on chip
  - Others can be sideloaded at runtime from BBP

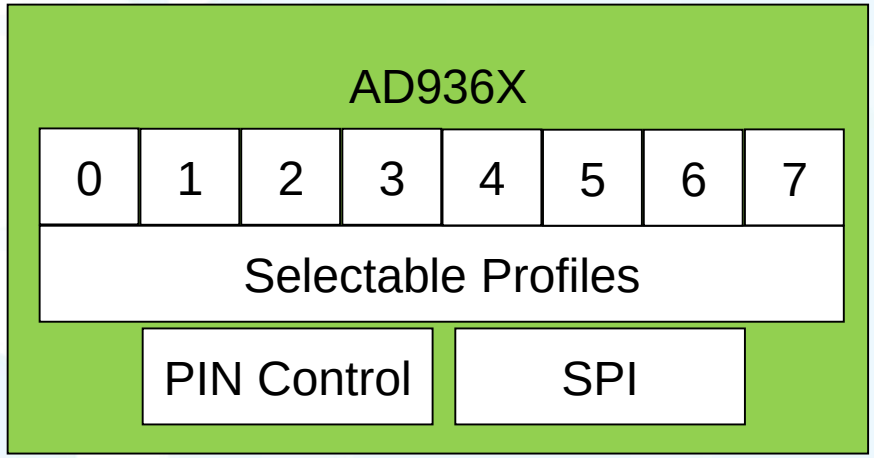


Default LO Control

If |Freq_N-Freq_N+1|>100MHz
Change VCO
Recal VCO
Else
Change VCO

Fastlock LO Control

If |Freq_N-Freq_N+1|>100MHz
Load Fastlock Profile
Else
Change VCO



Pin Control

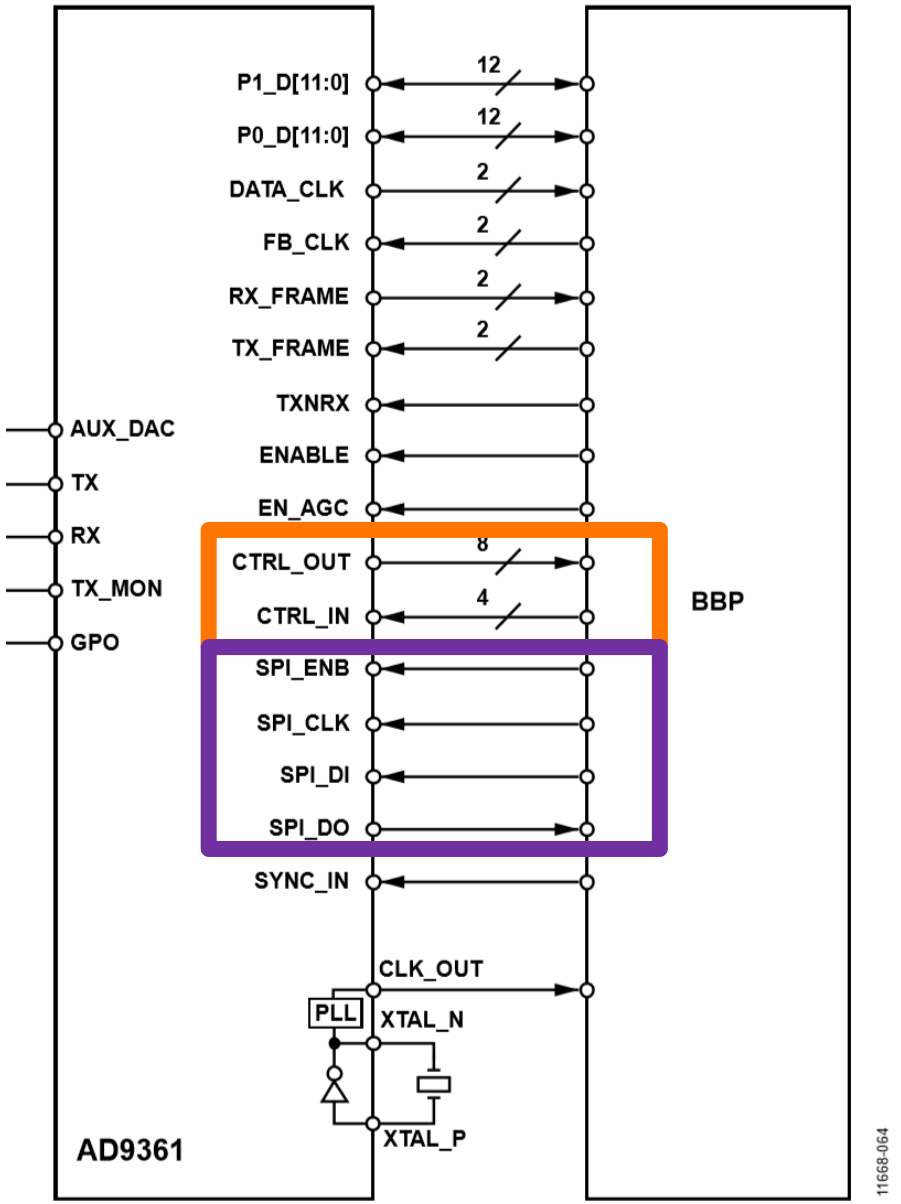
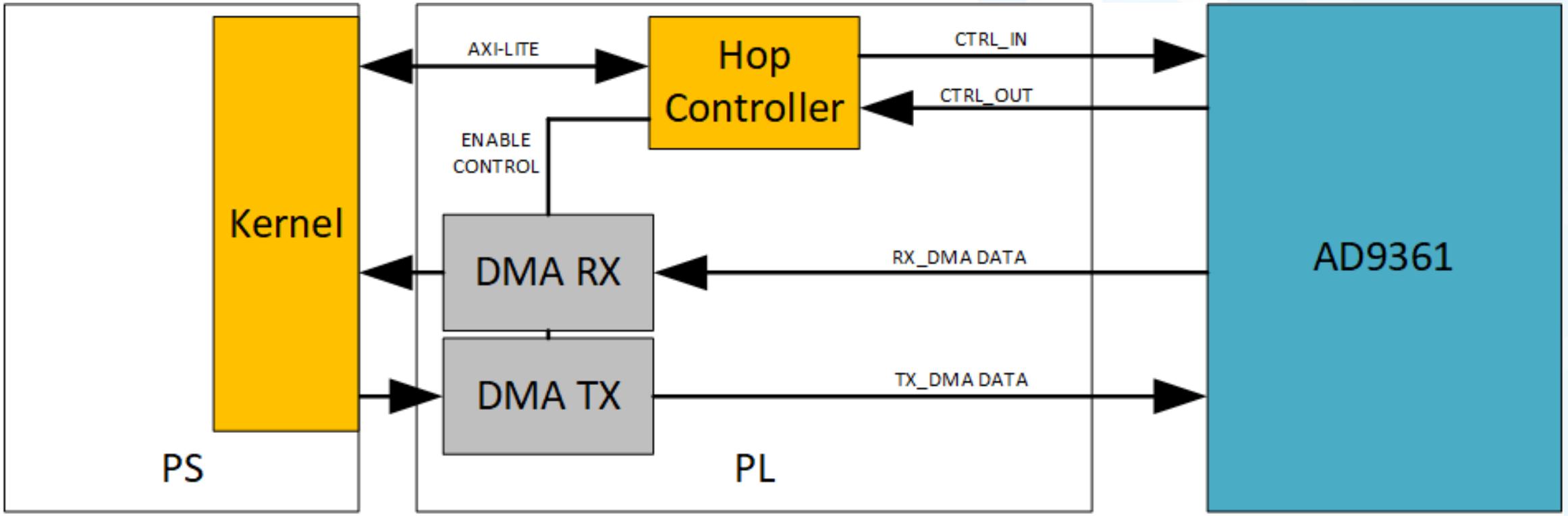


Figure 63. AD9361 Interface

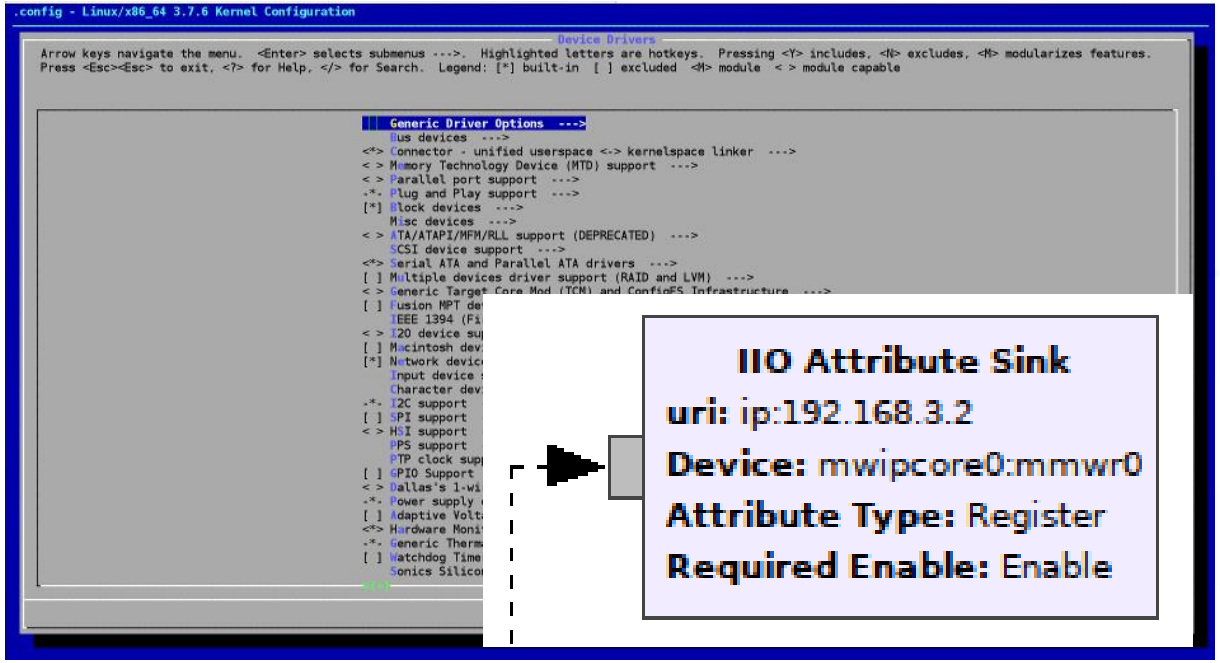
Controller for Pins



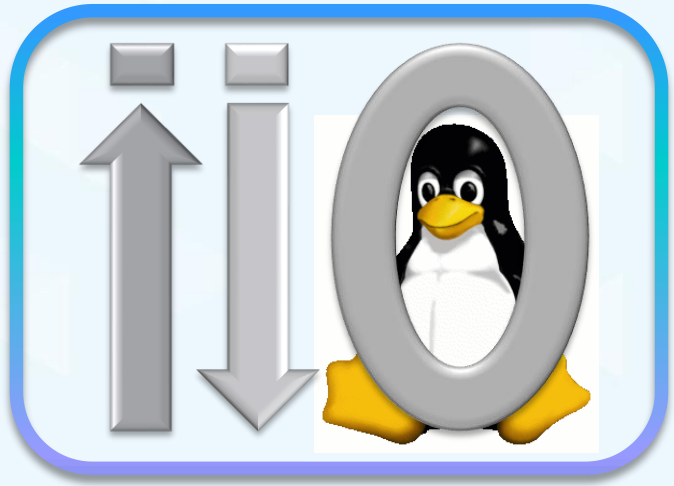
Documentation: https://wiki.analog.com/resources/eval/user-guides/adrv936x_rfsom/tutorials/frequency_hopping

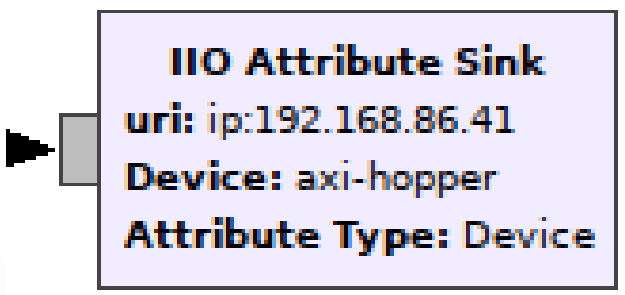
Driver Options

- **MW AXI-MM IIO**
  - Provides direct register access
  - Patch available for ADI kernel
    - Enabled in MathWorks kernel (E310, FMComms, ADRV)
  - Great for starting out, testing, debug
  - Driver logic lives in application



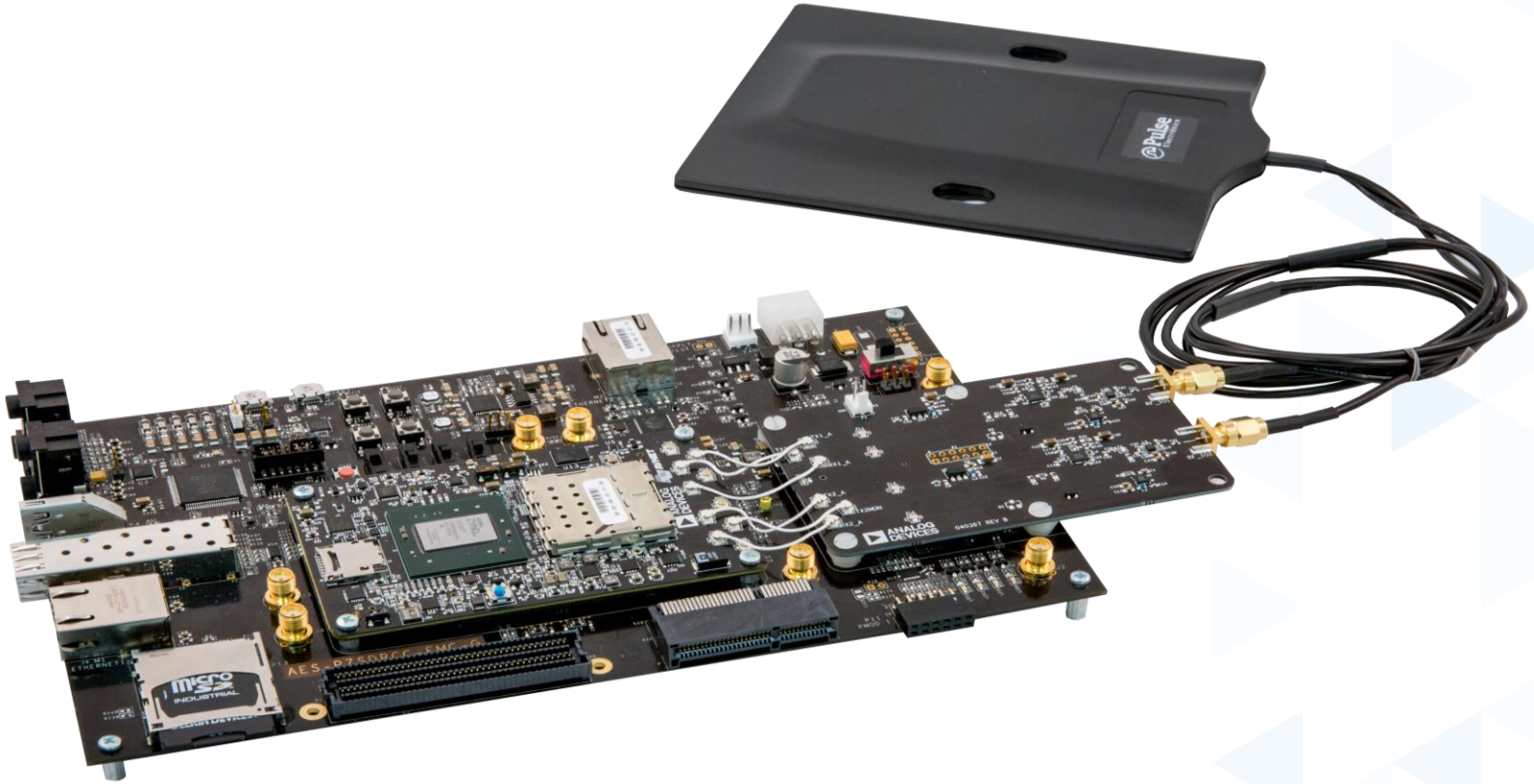
- **Custom IIO Driver**
  - Complex logic built into driver
  - Better portability
  - Not a great starting point





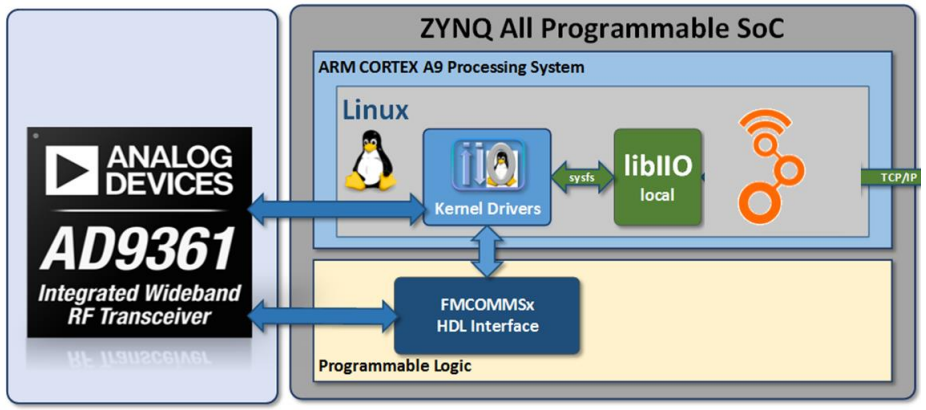
GNU Radio Example

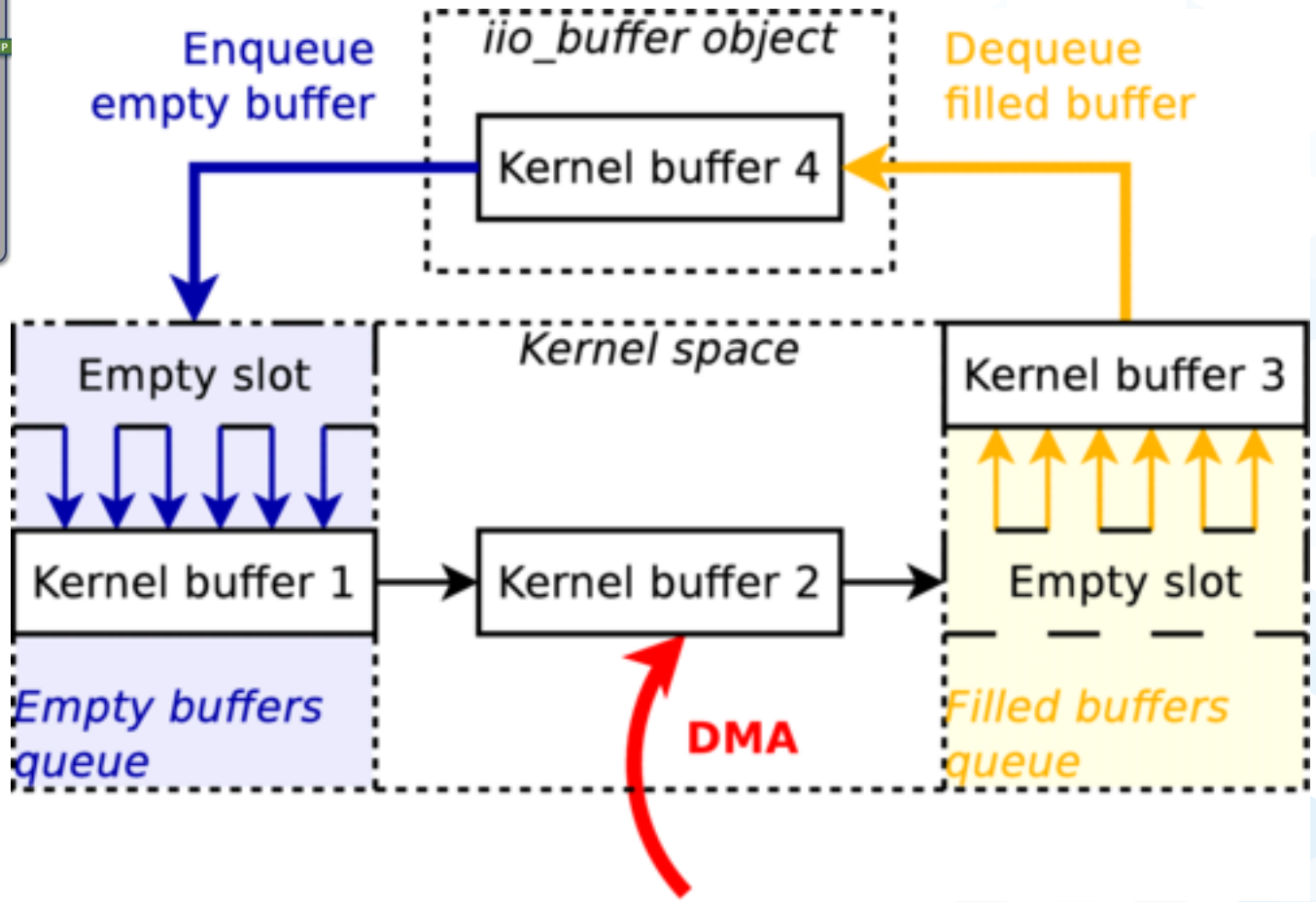
- See video
- Video uses ADRV9361-Z7035 below



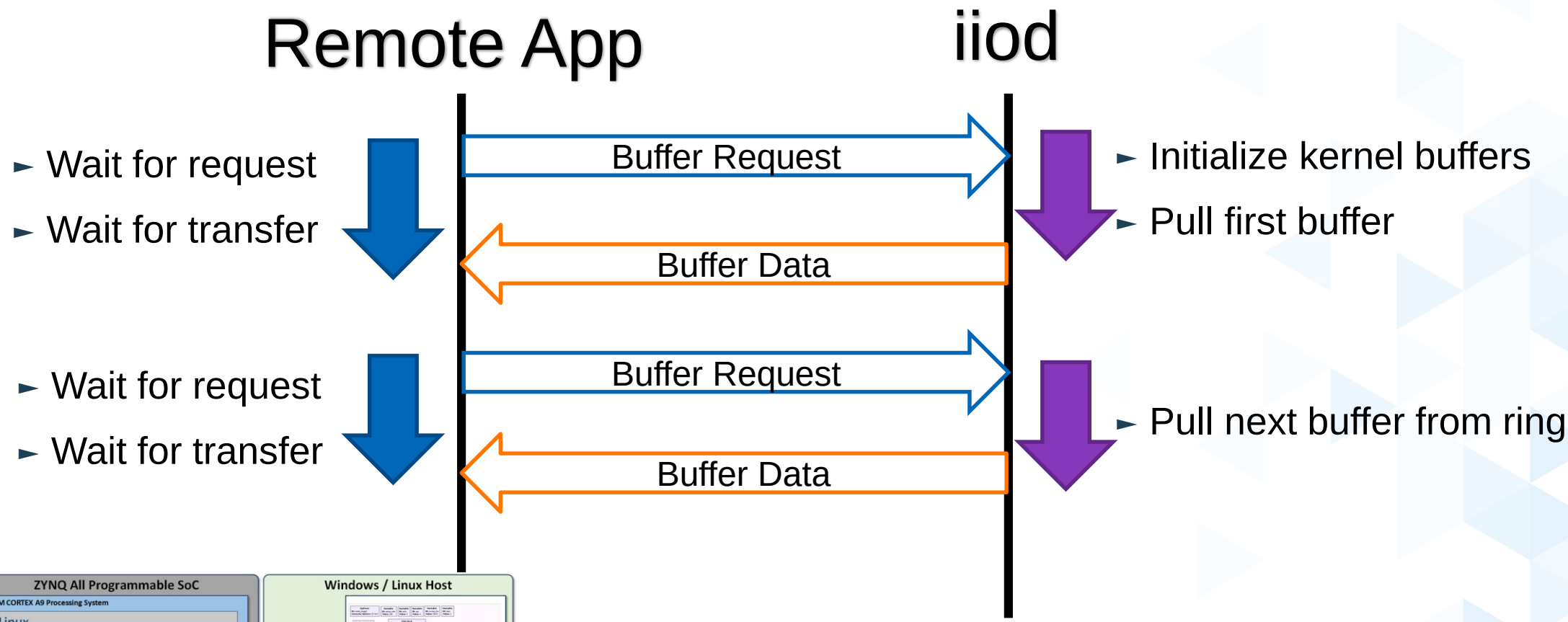
libIIO Performance

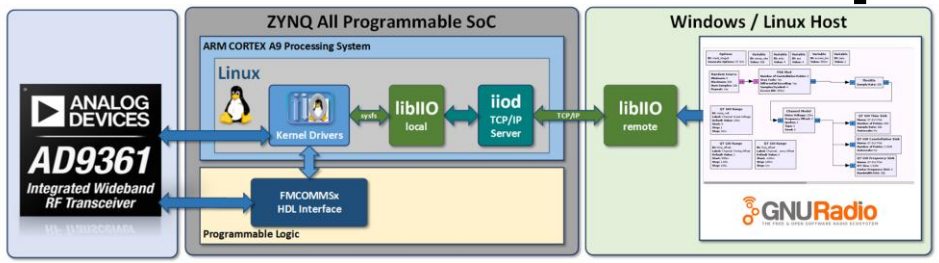
Local Buffers: High-Speed mmap interface





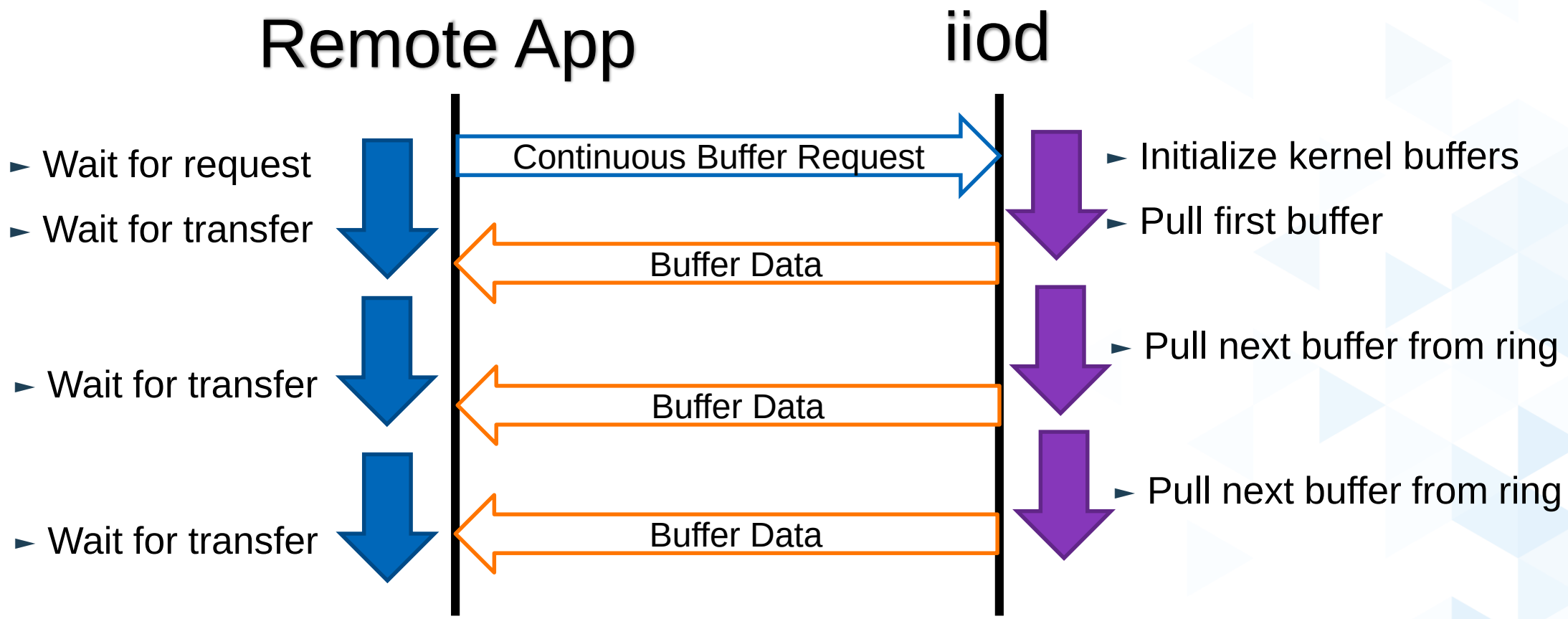
Remote Contexts: Synchronous Transfers





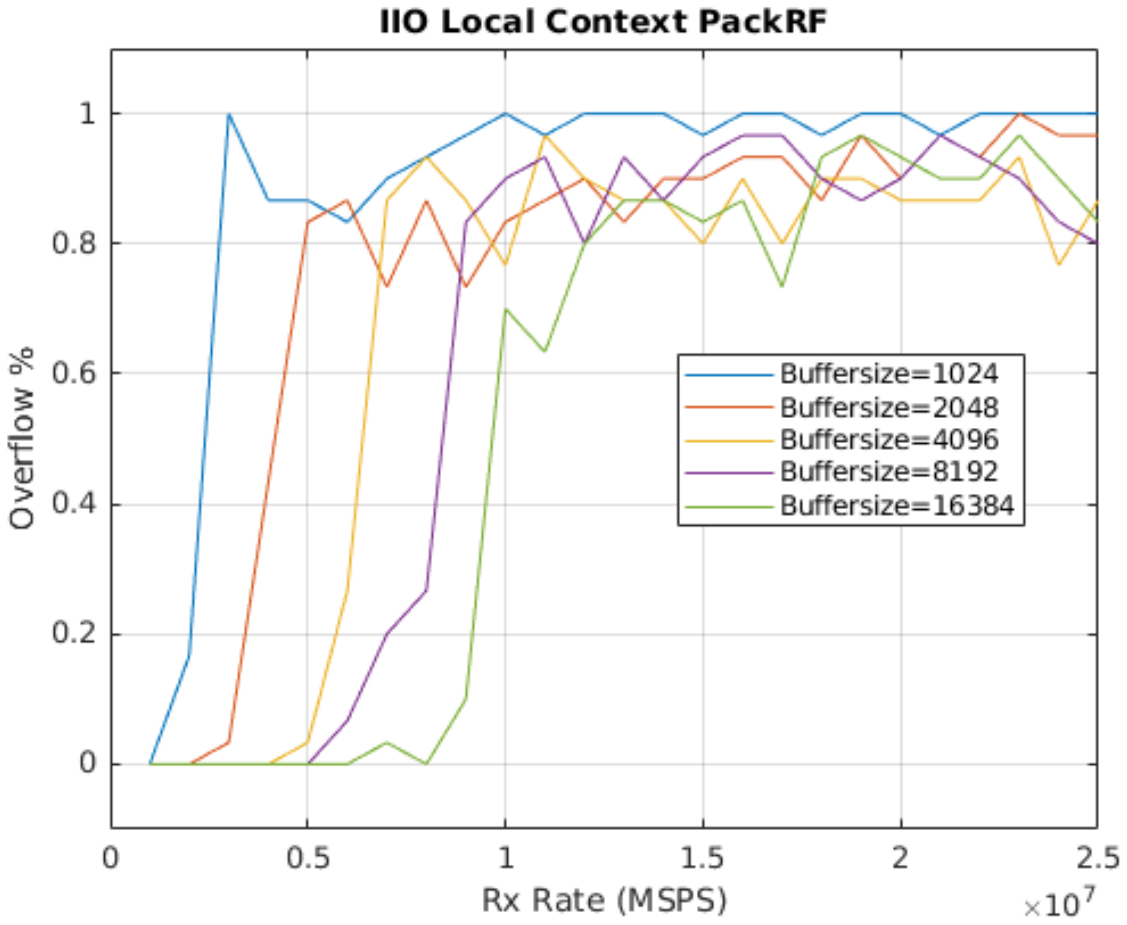
More info: https://wiki.analog.com/resources/tools-software/linux-software/libiio_internals

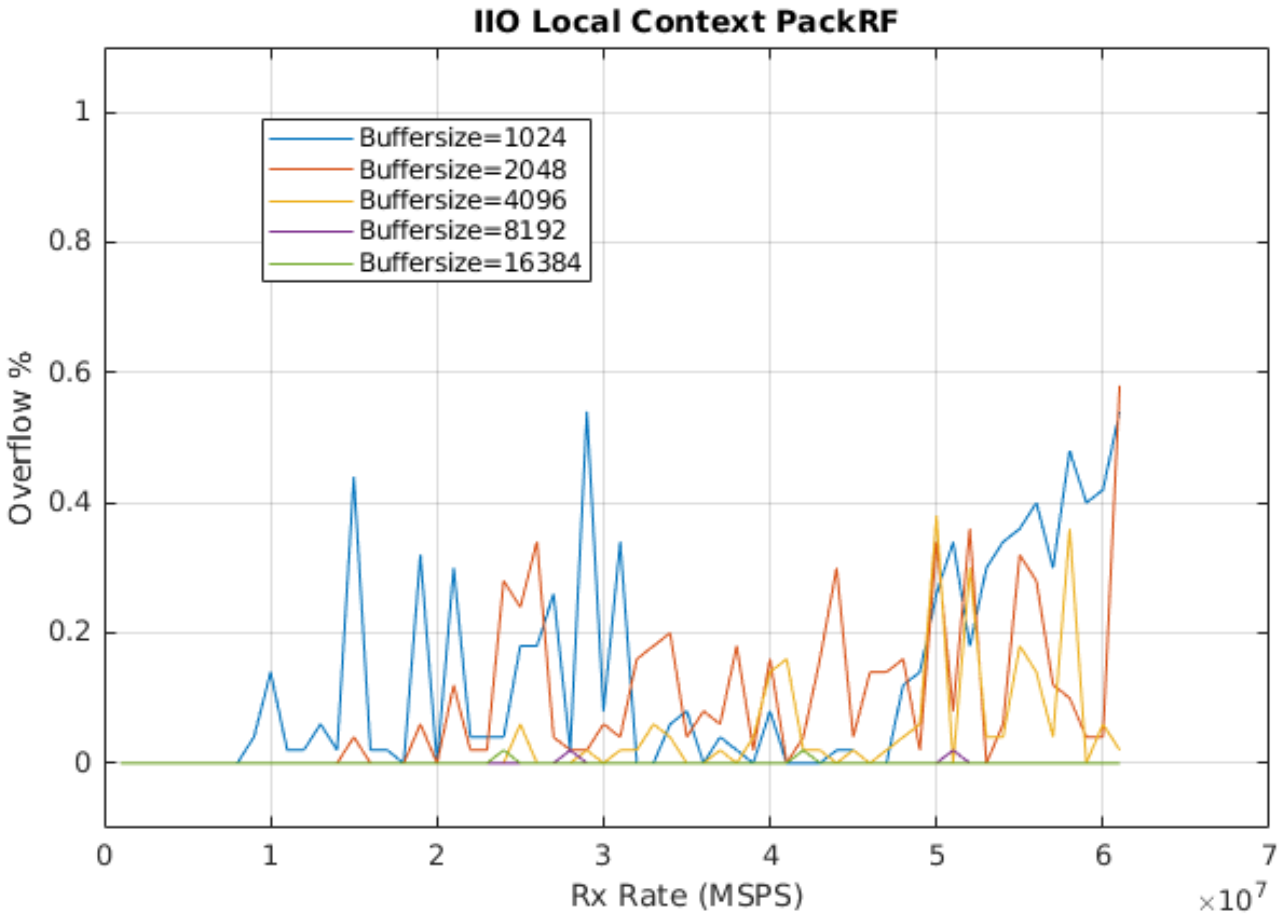
Remote Contexts: Synchronous Transfers



More info: https://wiki.analog.com/resources/tools-software/linux-software/libiio_internals

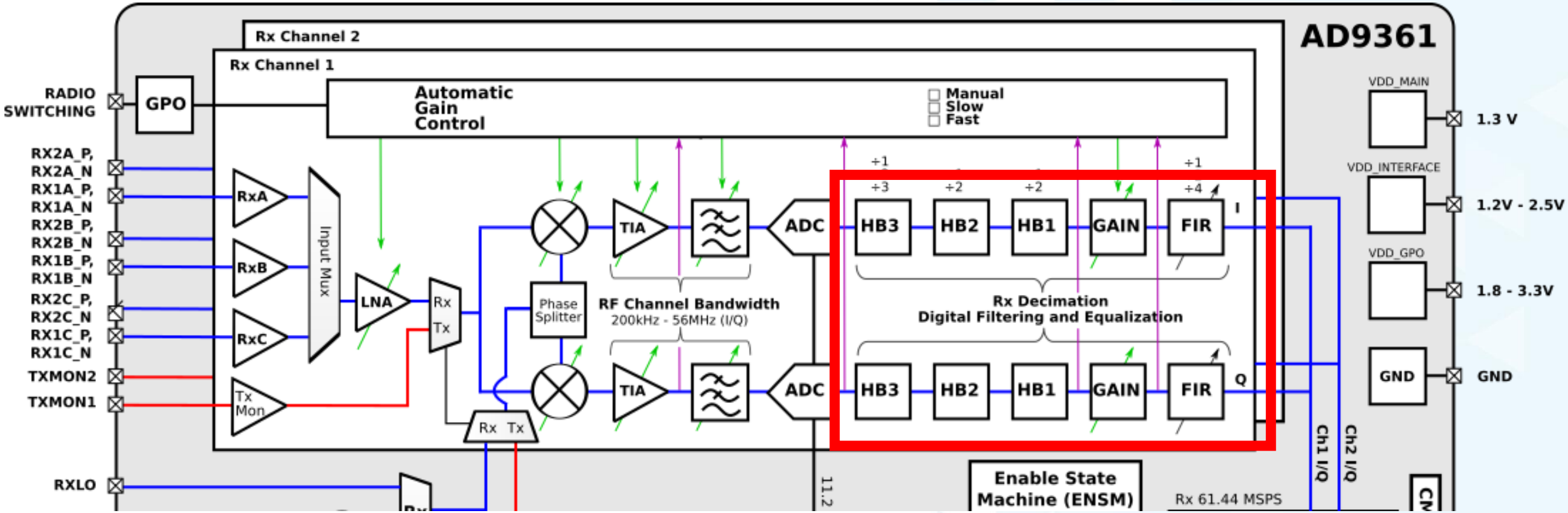
Performance: IP vs Local



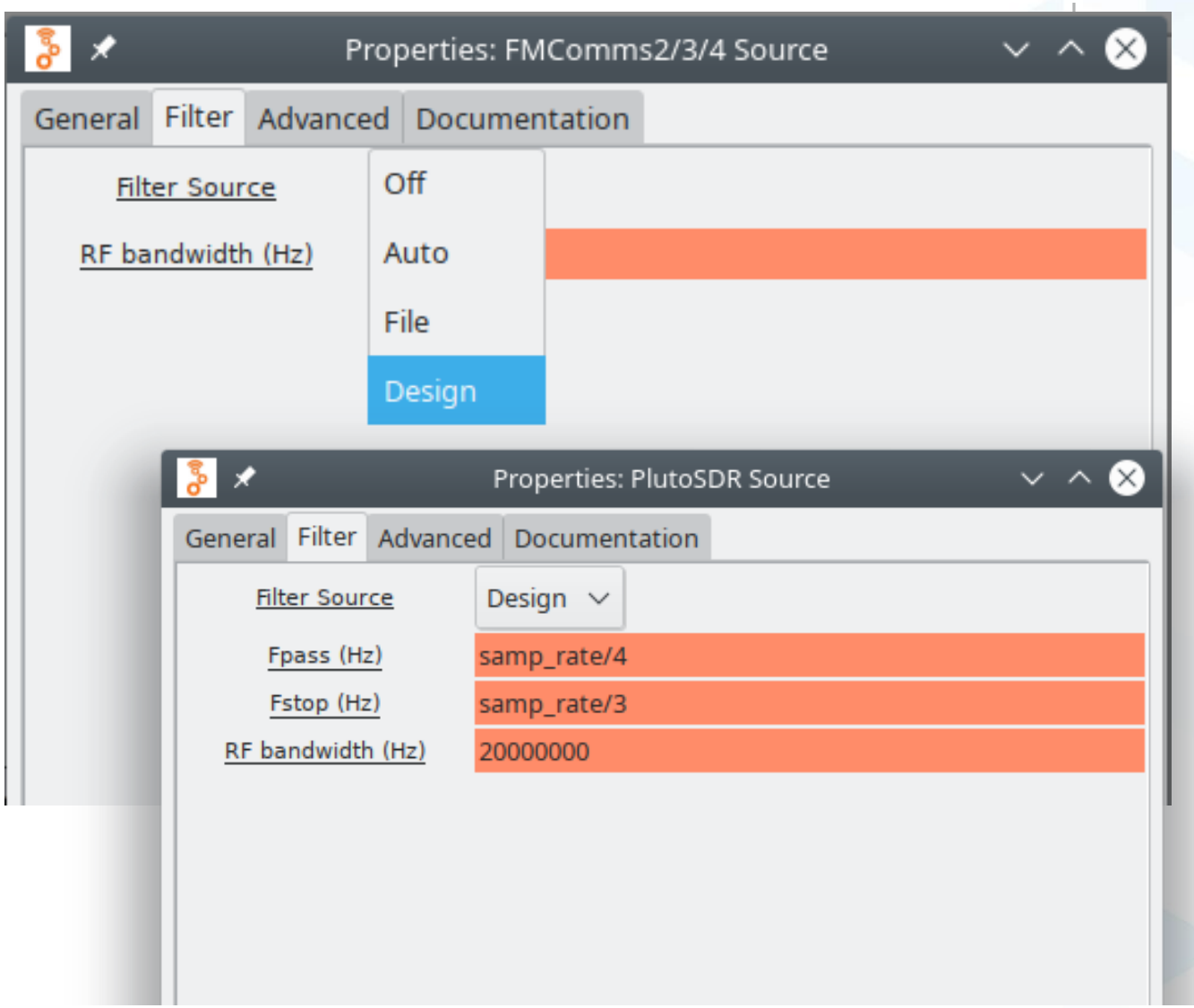


New Features

AD936x Filtering

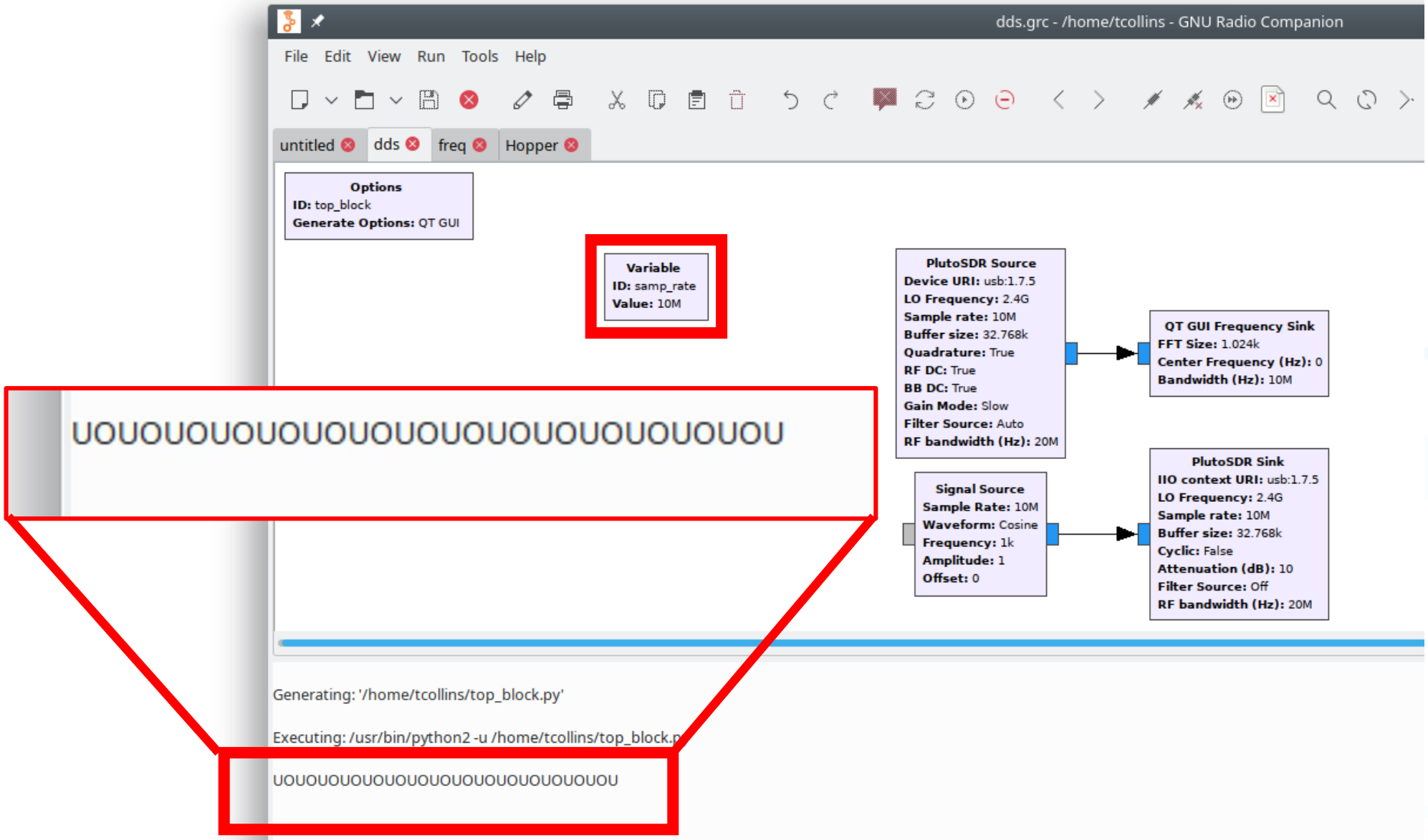


More Filtering Options

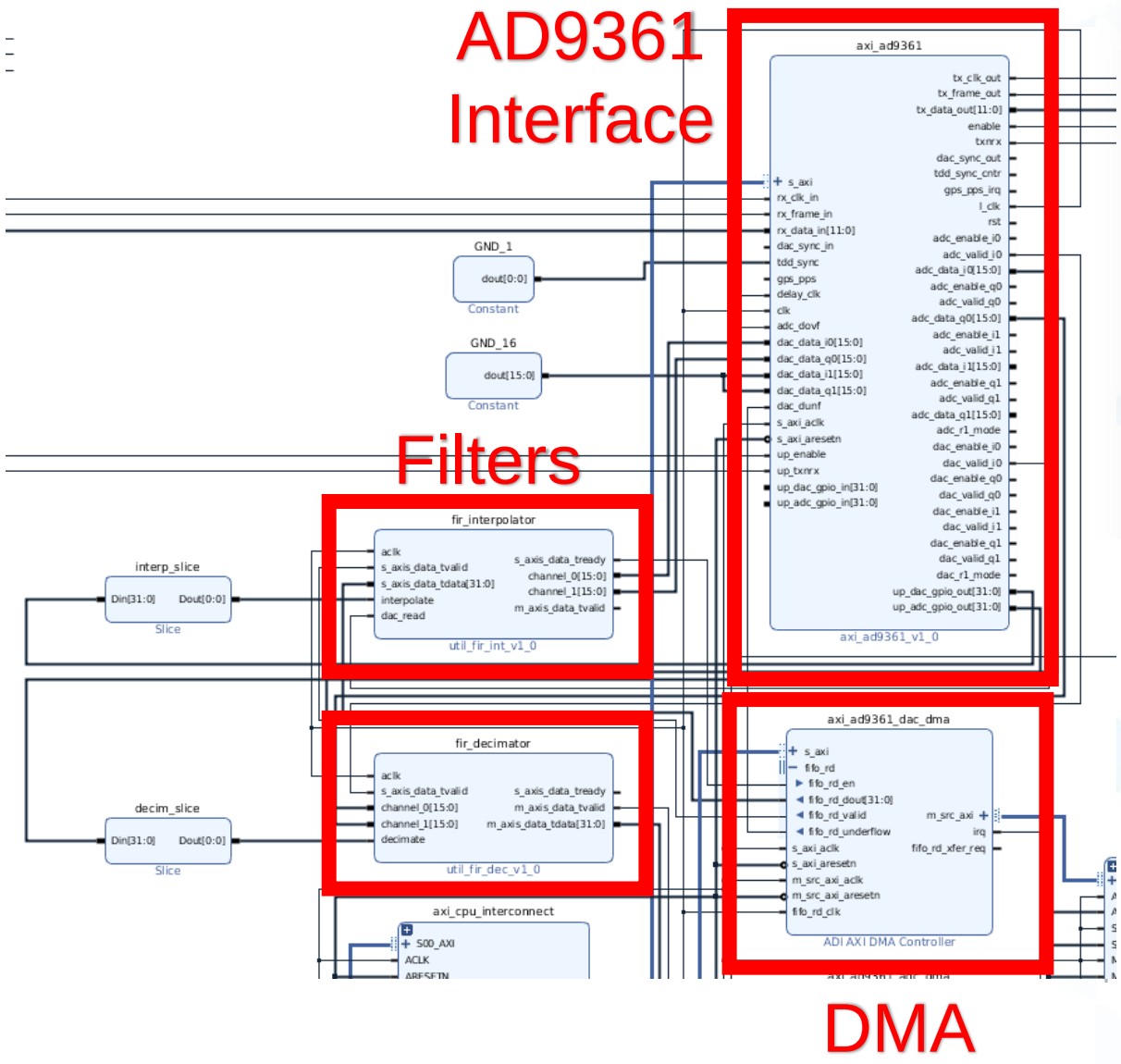


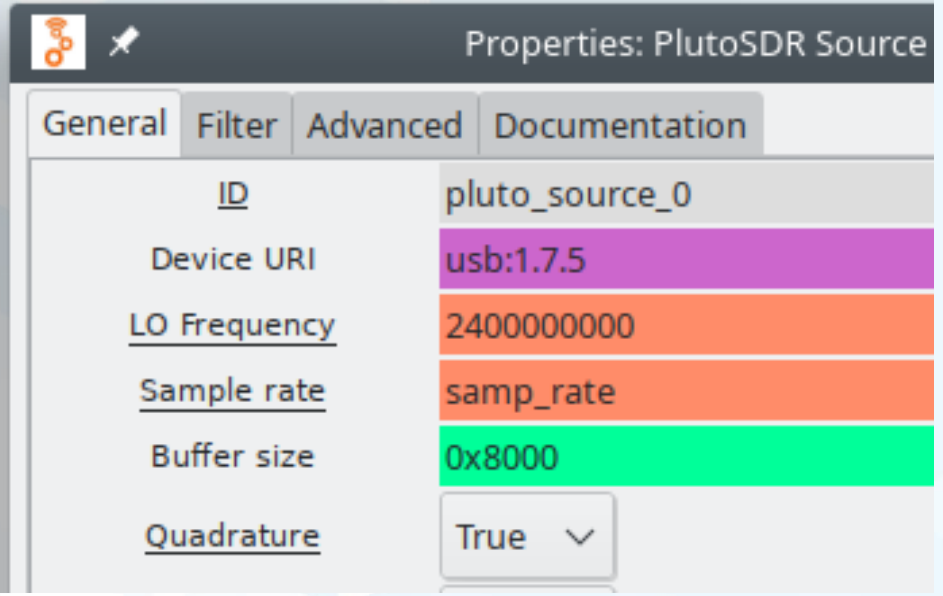
- New filter tab
  - Off: no filter
  - Auto: Preset filters
  - File: Load from ftr file
  - Design: Create filter on the fly based on:
    - RF bandwidth
    - Sample rate
    - F Stop
    - F Pass
    - From libad9361 library

Underflow/Overflow Indicators



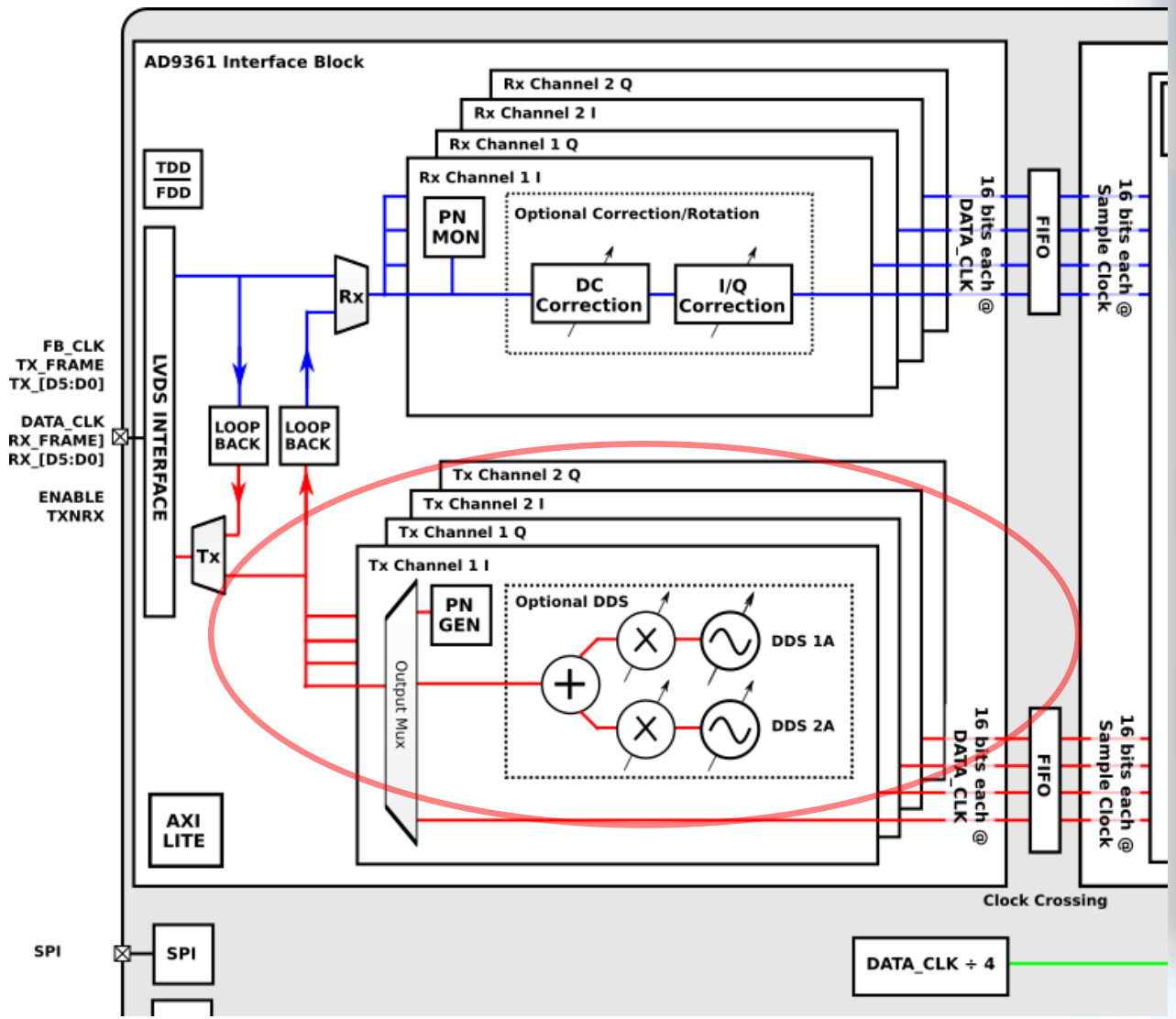
Pluto and ADRV FPGA Filters

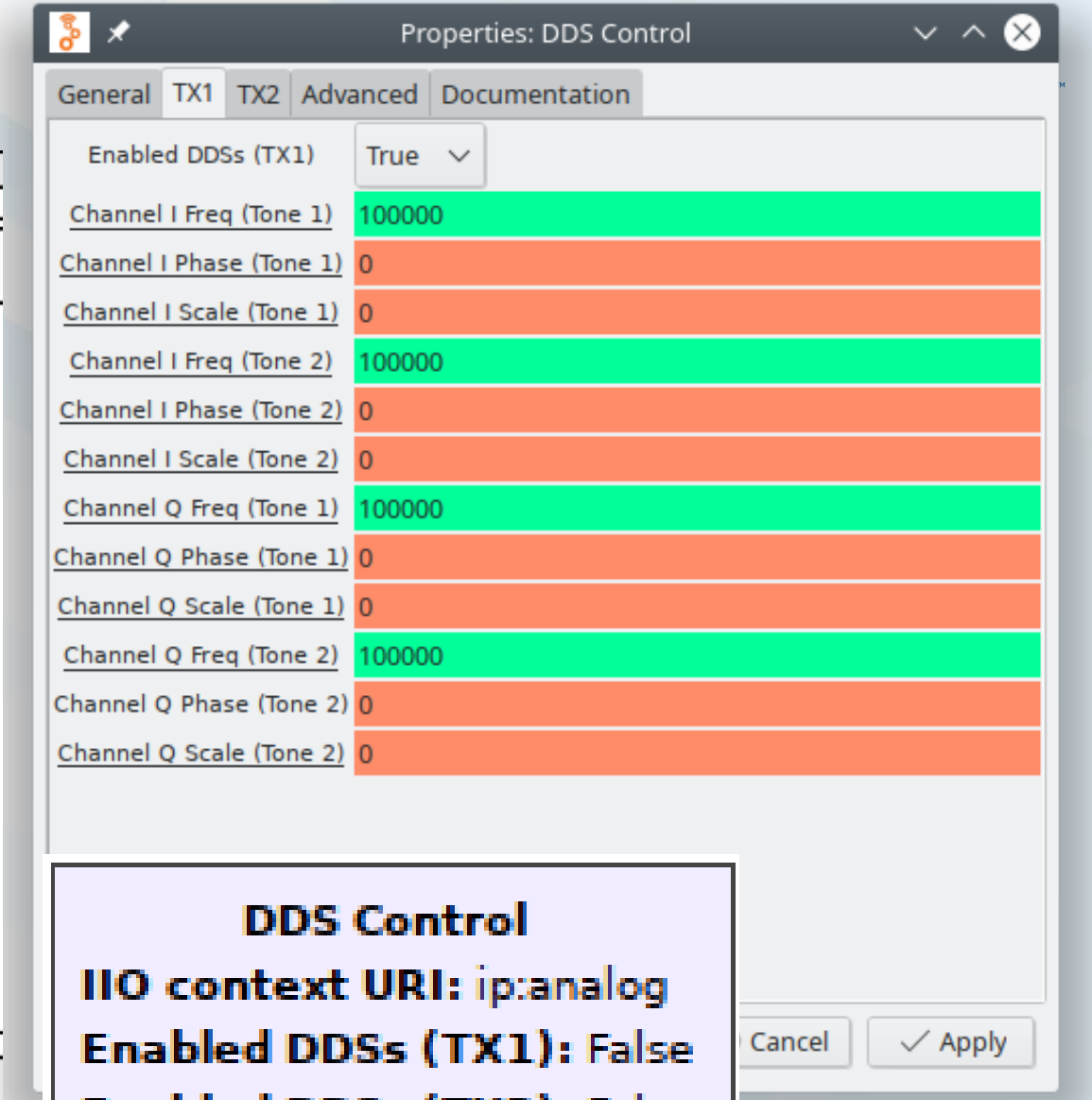




- Decimation/Interpolation Filter exist in FPGA for Pluto and new ADRV designs
  - Automatically configured now from GR
  - Allows rates down to **65104 Hz**

DDS Controller Block





DDS Control

IIO context URI: ip:analog

Enabled DDSs (TX1): False

Enabled DDSs (TX2): False

GRCon

DDS Controller Block Demo

- See video
- PlutoSDR used for example

GREP: gr-iio

GREP to mainline gr-iio

GREP 0017 -- Add support for IIO devices

- Original Author: Travis Collins travis.collins@analog.com
- Champion: Travis Collins travis.collins@analog.com
- Status: Active

History:

- 22-Feb-2019: Initial Draft
- 28-Feb-2019: Changed Status to Active

Abstract

GNU Radio is an invaluable tool for data processing, especially for data sources that can provide l
time data like sounds devices and software-defined radios. Such support today is provided by in-t
audio, gr-uhd, gr-comedi, and gr-video-sdl. However, a large category of sensors/DAC/ADC amor
the Industrial Input/Output Framework of the linux kernel is not supported in-tree today.

- Making it easier for users!

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

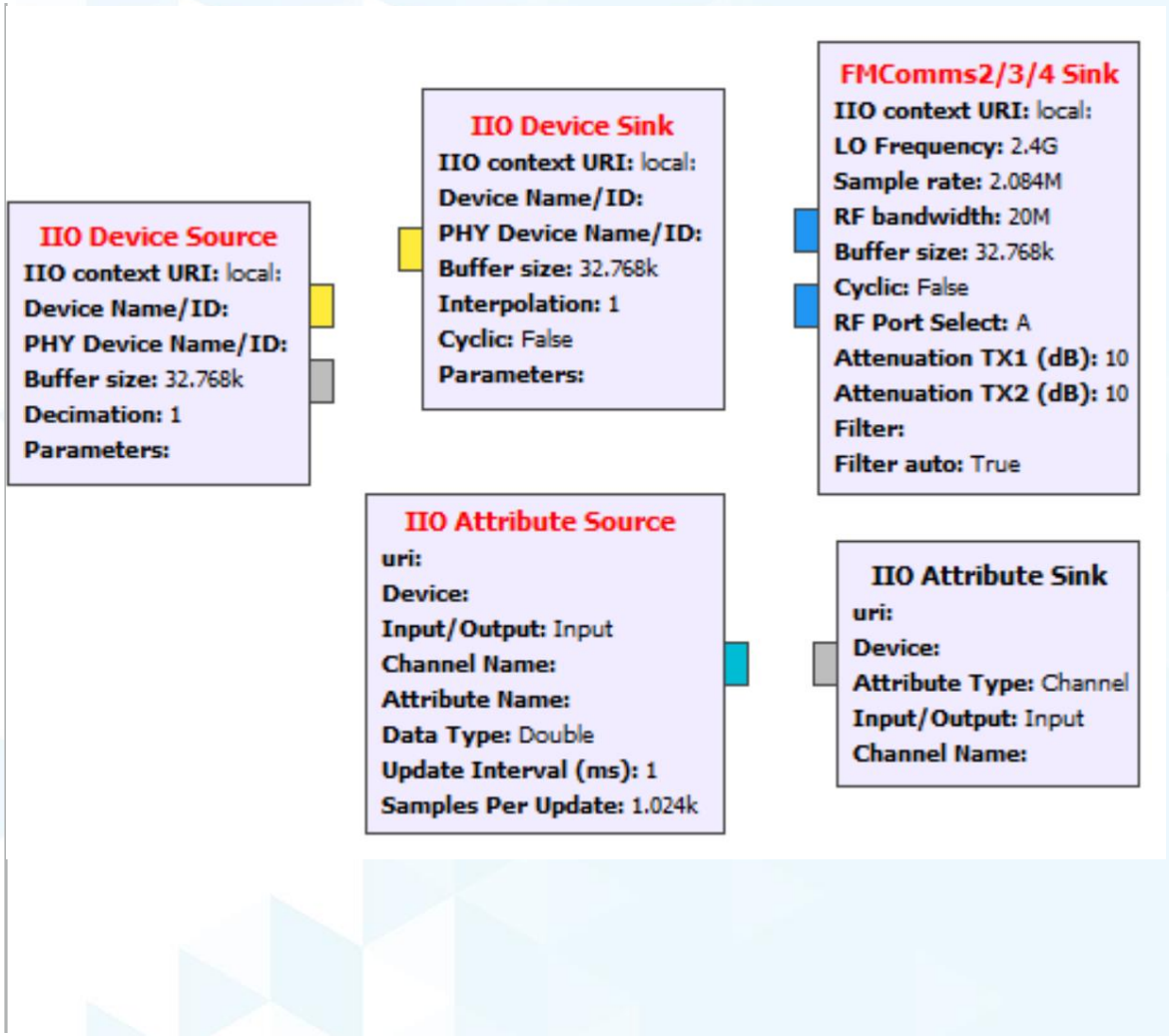
Attributes

Streams

~~Math (Scopy)~~

SDR

~~Math (Scopy)~~



pyadi-iio

ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

- IIO version

```
32  # Setup contexts
33  try:
34      ctx = iio.Context('ip:192.168.2.1')
35  except:
36      print("No device found")
37      sys.exit(0)
38
39  ctrl = ctx.find_device("ad9361-phy")
40  txdac = ctx.find_device("cf-ad9361-dds-core-lpc")
41  rxadc = ctx.find_device("cf-ad9361-lpc")
42
43  # Configure transceiver settings
44  rxLO = ctrl.find_channel("altvoltage0", True)
45  rxLO.attrs["frequency"].value = str(int(RXLO))
46  txLO = ctrl.find_channel("altvoltage1", True)
47  txLO.attrs["frequency"].value = str(int(TXLO))
48
49  tx = ctrl.find_channel("voltage0",True)
50  tx.attrs["rf_bandwidth"].value = str(int(RXBW))
51  tx.attrs["sampling_frequency"].value = str(int(RXFS))
52  tx.attrs['hardwaregain'].value = '-30'
53
54  rx = ctrl.find_channel("voltage0")
55  rx.attrs["rf_bandwidth"].value = str(int(TXBW))
56  rx.attrs["sampling_frequency"].value = str(int(TXFS))
57  rx.attrs['gain_control_mode'].value = 'slow_attack'
58
```

- pyadi-iio version

```
41  # Create radio
42  sdr = adi.Pluto()
43
44  # Configure properties
45  sdr.rx_rf_bandwidth = 4000000
46  sdr.rx_lo = 2000000000
47  sdr.tx_lo = 2000000000
48  sdr.tx_cyclic_buffer = True
49  sdr.tx_hardwaregain = -30
50  sdr.gain_control_mode = "slow_attack"
51
```

pip install pyadi-iio
(Don't pip install iio)

GRCon

Thank you!

Code: github.com/analogdevicesinc

Support: ez.analog.com

Doc: wiki.analog.com

References

- https://www.ti.com/store/ti/en/p/product/?p=LM335Z/NOPB&utm_source=google&utm_medium=cpc&utm_campaign=ecm-tistore-invf-GMC_US-cpc-store-google-wwe&utm_content=GMC&ds_k=PRODUCT_GROUP&DCM=yes&gclid=CjwKCAjwq4fsBRBnEiwANTahcH1fAXFxufrfIk8_ZQX4XudpDHnjjTaN6VP2fb5Z7mGvP7TmWu2vERoCvb0QAvD_BwE
- https://www.mouser.com/new/omronelectronics/omron-d6t/
- https://www.digikey.com/product-detail/en/adafruit-industries-llc/163/1528-1120-ND/5353580?WT.srch=1&gclid=CjwKCAjwq4fsBRBnEiwANTahcBaKqAgJWHohSFFioc-Hd5_aZqpJoLWK86AXxRjP_BE2OL2lwFW7LRoCD0AQAvD_BwE#images