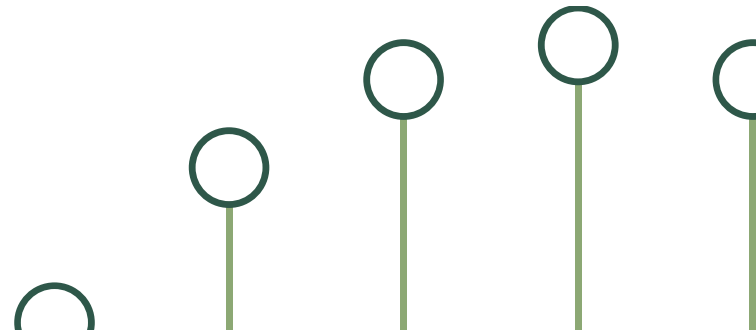
WinnComm Europe 2016
RFNoC™ Tutorial: Introduction

Tim Fountain
e: tim.fountain@ni.com
p: +1-503-720-2456



Agenda

Ettus Research™
A National Instruments Company

- SDR Overview
- RFNoC Hardware Platforms
- RFNoC Introroduction
- FPGA Development
- Software Development

Ettus Research Overview

- Leader in Software Defined Radio (SDR) and Signals Intelligence (SIGINT)

- Maker of Universal Software Radio Peripheral (USRP™)
- Enables rapid development of SDR and RF systems
- Supported by a strong software ecosystem
- DC-6 GHz, MIMO capability, Embedded, USB/GigE
- Wireless Innovation Forum – 2010 Technology of the Year
- Wireless Innovation Forum – 2014 International Achievement Award

- About The Company
  - Founded in 2004
  - Located in Santa Clara, CA – Silicon Valley
  - Stand alone subsidiary of National Instruments since 2010

SDR Algorithm Prototyping Applications

Ettus Research™ A National Instruments Company

Communications & RF Identification

Utilities & Infrastructure

Medical Devices & Internet of Things

Research Topics

Data rate

Capacity

Power Consumption

Spectrum Sharing

Interference

Security

Monitoring

Latency

Aerospace & Defense

Automotive & Car to Car

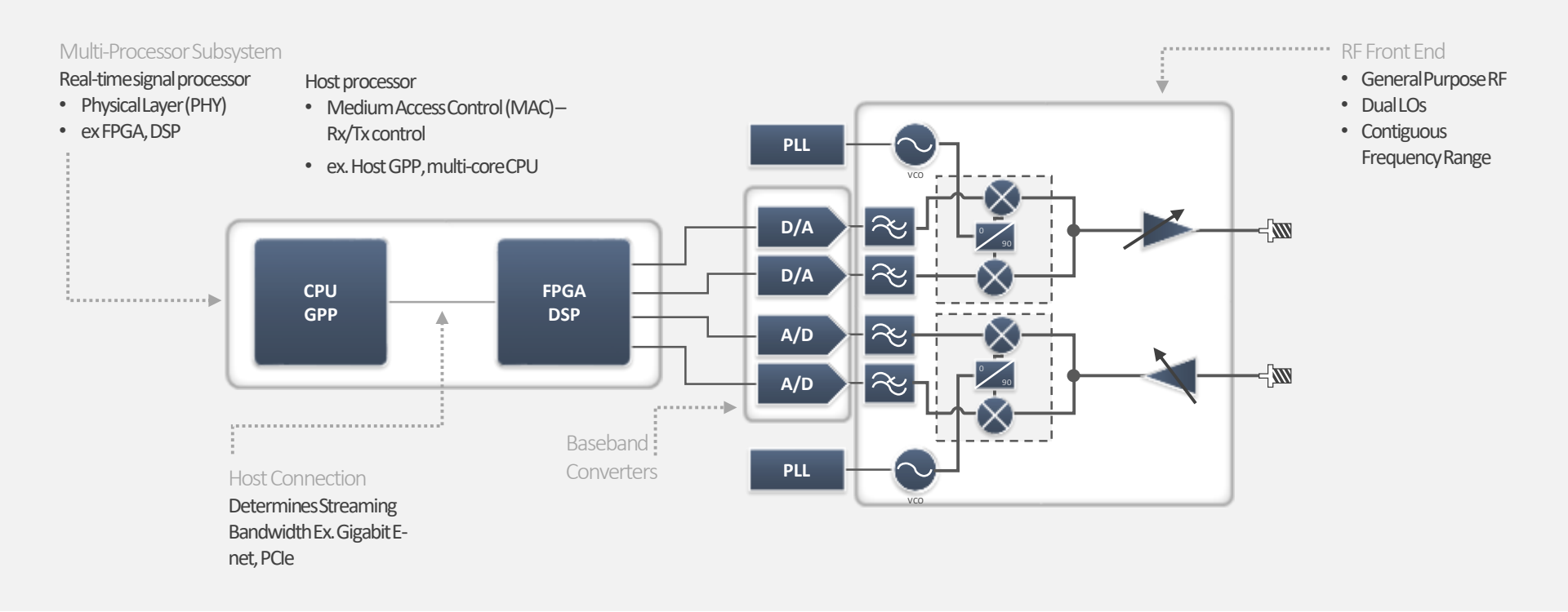
Land Mobile & Safety Radio

Satellite Comm & Navigation

Research/Education

SDR Architecture

Ettus Research™ A National Instruments Company



System Architecture

Ettus Research™ A National Instruments Company

Application

LTE/WiFi Application Frameworks	Vendor defined LORA, Sigfox, IOT	Open Source LTE Stacks	Comms System Toolbox
LabVIEW	*Custom C++, etc.*	**GNURadio** RFNoC Python / GRC	**The Mathworks ™**

UHD Driver/RFNoC

Portability Across Hardware, Interface, OS, and Dev. Environment

Windows	MacOS	Linux	Embedded Linux

Hardware

Radio - Embedded CPU, FPGA and/or host connected
RF board
Antenna
RF signal conditioning

Antenna VERT400

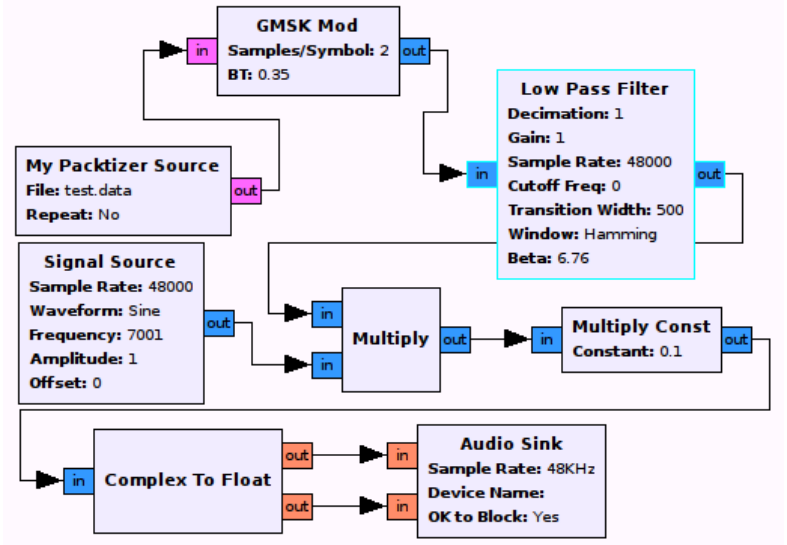
RF Board (UBX)

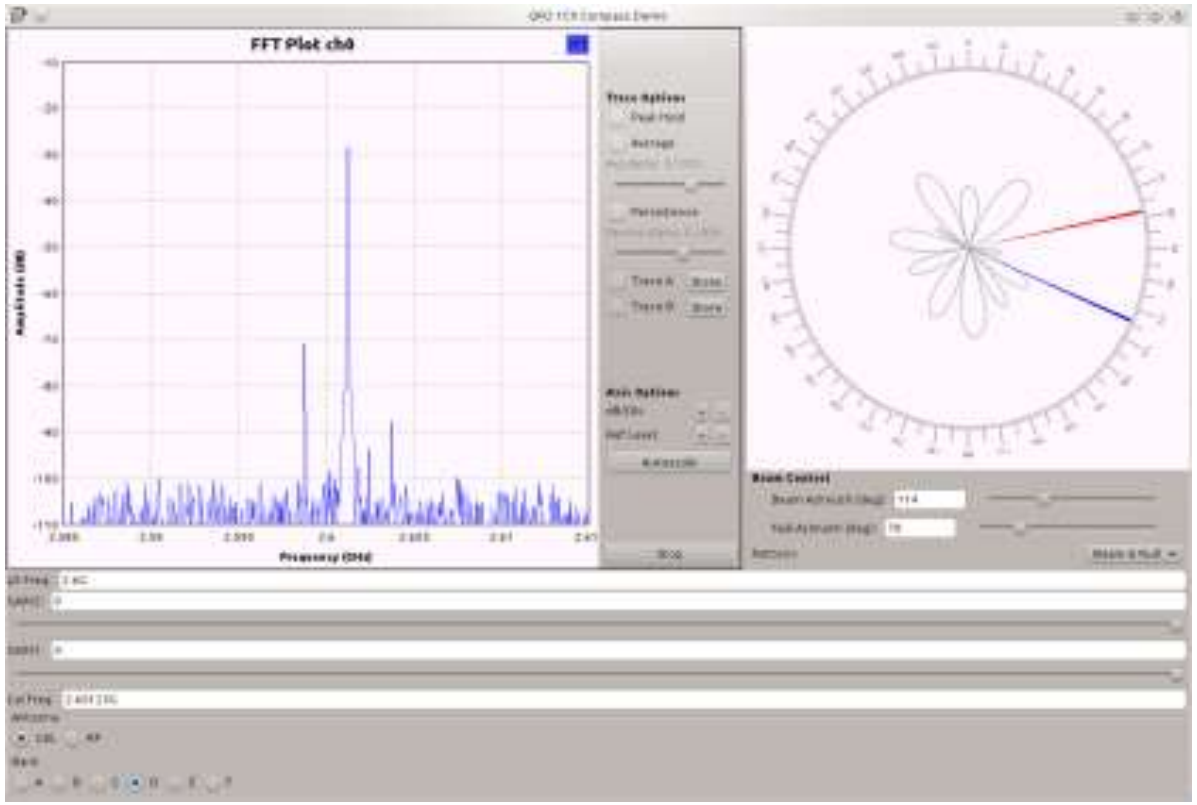
USRP Radios

GNU Radio Introduction

Ettus Research™
A National Instruments Company

- History
  - Experiment with ATSC Decoding in Software
  - Impetus for USRP – low cost hardware
  - Ettus Research – A Leading Contributor
- Free and Open Source
  - 1000's of users
  - Mailing List
  - gnuradio.org
  - Annual conference





GNU Radio
THE FREE & OPEN SOFTWARE RADIO ECOSYSTEM

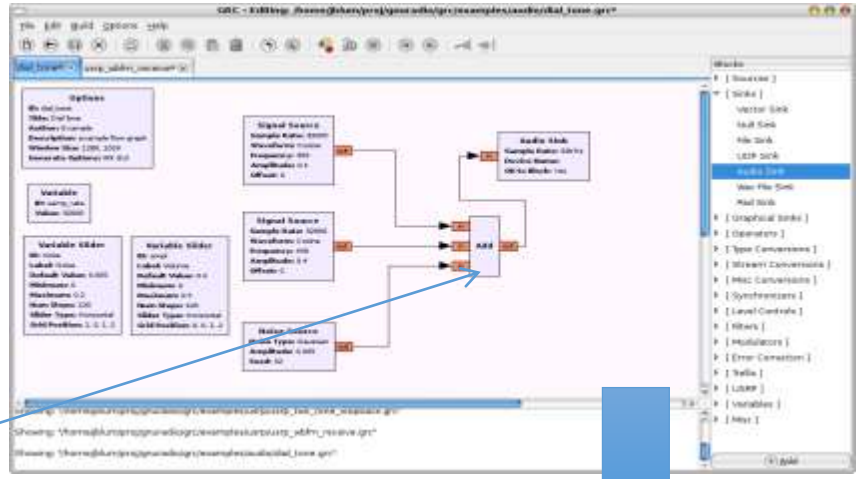
GNU Radio Design Flow

Ettus Research™ A National Instruments Company

DSP Block – C++ Work Function

```
int gr_add_ff::work(int noutput_items,
                    gr_vector_const_void_star &input_items,
                    gr_vector_void_star &output_items)
{
  float *out = (float *) output_items[0];
  int noi = d_vlen*noutput_items;

  memcpy(out, input_items[0], noi*sizeof(float));
  volk_32f_x2_add_32f_a(out, out, (const float*)input_items[i], noi);

  return noutput_items;
}
```

GNU Radio Companion (optional)



Python Flow-Graph

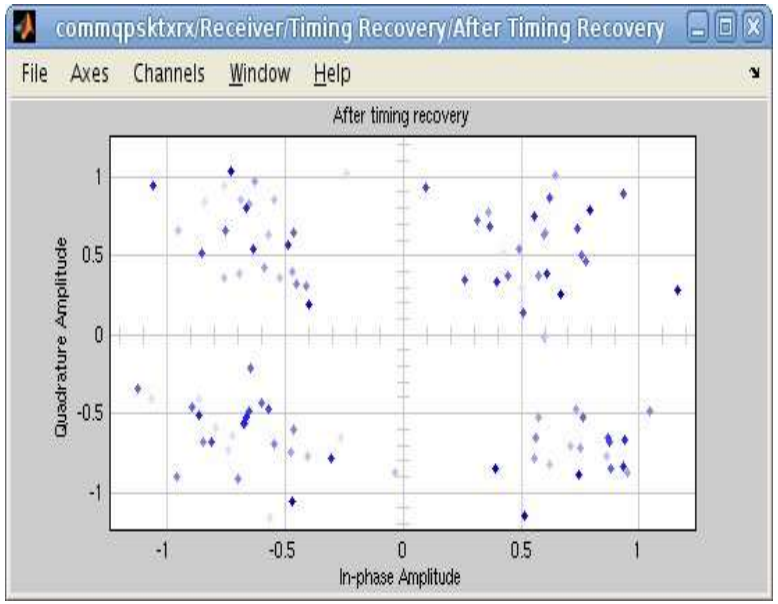
```
tb = gr.top_block()

src1 = gr.sig_source_f(32000, gr.GR_SIN_WAVE, 350, .5, 0)
src2 = gr.sig_source_f(32000, gr.GR_SIN_WAVE, 440, .5, 0)

adder = gr.add_ff()

sink = audio.sink(32000)

tb.connect(src1, (adder, 0))
tb.connect(src2, (adder, 1))

tb.connect(adder, sink)
tb.run()
```

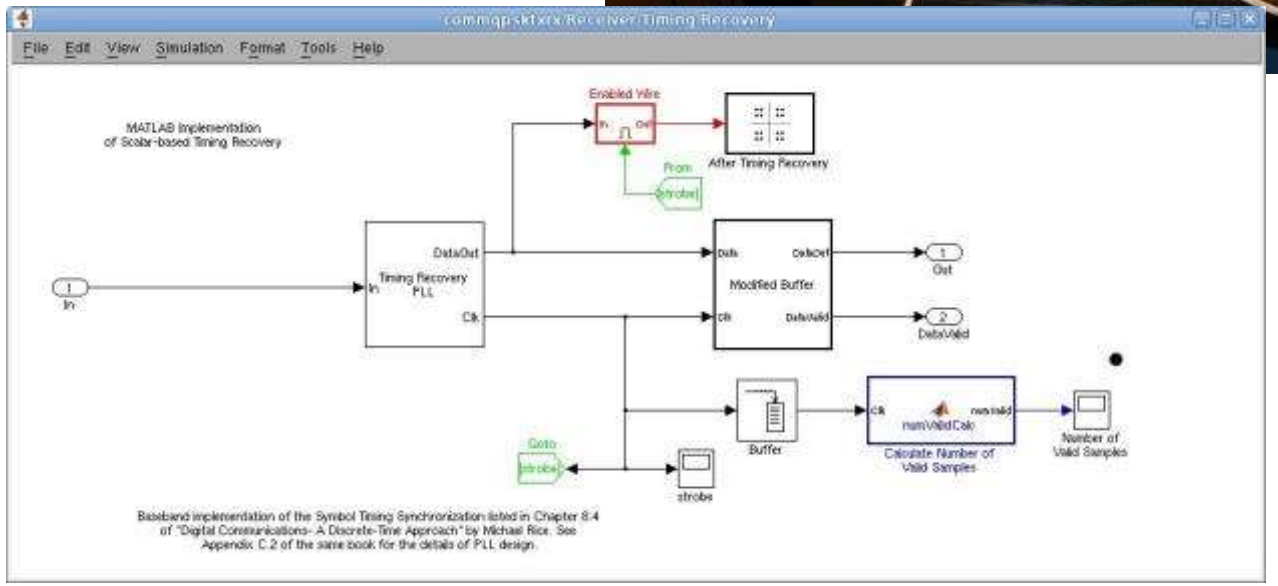
- Blocks
  - Large library of existing IP -> Mod/demod, filters, USRP I/O, GUI features, etc.
  - Write custom blocks – C++ or Python
- GNU Radio Companion (optional)
  - Import blocks
  - Connect blocks
  - Generate python source code for flowgraph
- Python Flow-Graph
  - Generate from GRC and/or hand-write
  - Simplifies block connectivity

Matlab™ and Simulink™

Ettus Research™ A National Instruments Company

- Compatible with Matlab Scripts and Simulink Models
  - Communications blockset/toolbox
- FPGA supported by HDL Coder
- USRP N210, B210, X310
- Examples
  - QPSK transmitter and receiver
  - FM mono and stereo receiver
  - FRS/GMRS transmitter and receiver





3rd Generation USRP Family

Ettus Research™
A National Instruments Company

	Embedded E3xx	High Performance X3xx
Frequency (Hz)	70 MHz – 6 GHz	DC - 30MHz & 10MHz – 6GHz
Bandwidth	56MHz in 1x1 32 MHz in 2x2	160 MHz per channel
Channels	2 Tx, 2 Rx w/ filter banks	2 TX, 2 RX
RF Performance	Good	Best
Architecture	Integrated RFIC – AD9361	RF Daughterboards
Communication	Embedded USB 2.0 1 GbE	2x 10GbE PCIe x4
MIMO Capability	2x2	2x2 to 256x256
LabVIEW Support	No	Yes
FPGA Fabric	Kintex 7	Kintex 7
CPU	ARM Dual Core A9	N/A
S/W Ecosystem	GNU Radio C++ Xilinx Vivado C Coder HDL Coder MatLab Simulink	GNU Radio C++ Xilinx Vivado C Coder HDL Coder MatLab Simulink

USRP E-Series Overview

Ettus Research™ A National Instruments Company

Specs

- Frequency Range: 70MHz - 6 GHz, 10dBm power output
- Analog Devices 9361 with filter banks
- 2x2 MIMO standard configuration
- ~ 50 MHz BW / channel
- Xilinx Zynq-7020
  - ARM Dual-Core Cortex A9 @ 886 MHz
  - 1GB MB Processor RAM
  - 512 MB FPGA RAM
- 133x68x26.4 mm, 375g
- 3-9 W

Features

- I/O: GigE, Audio in/out, USB 2.0 Host, GPS In
- Micro SD memory card slot
- GPS Receiver

Applications

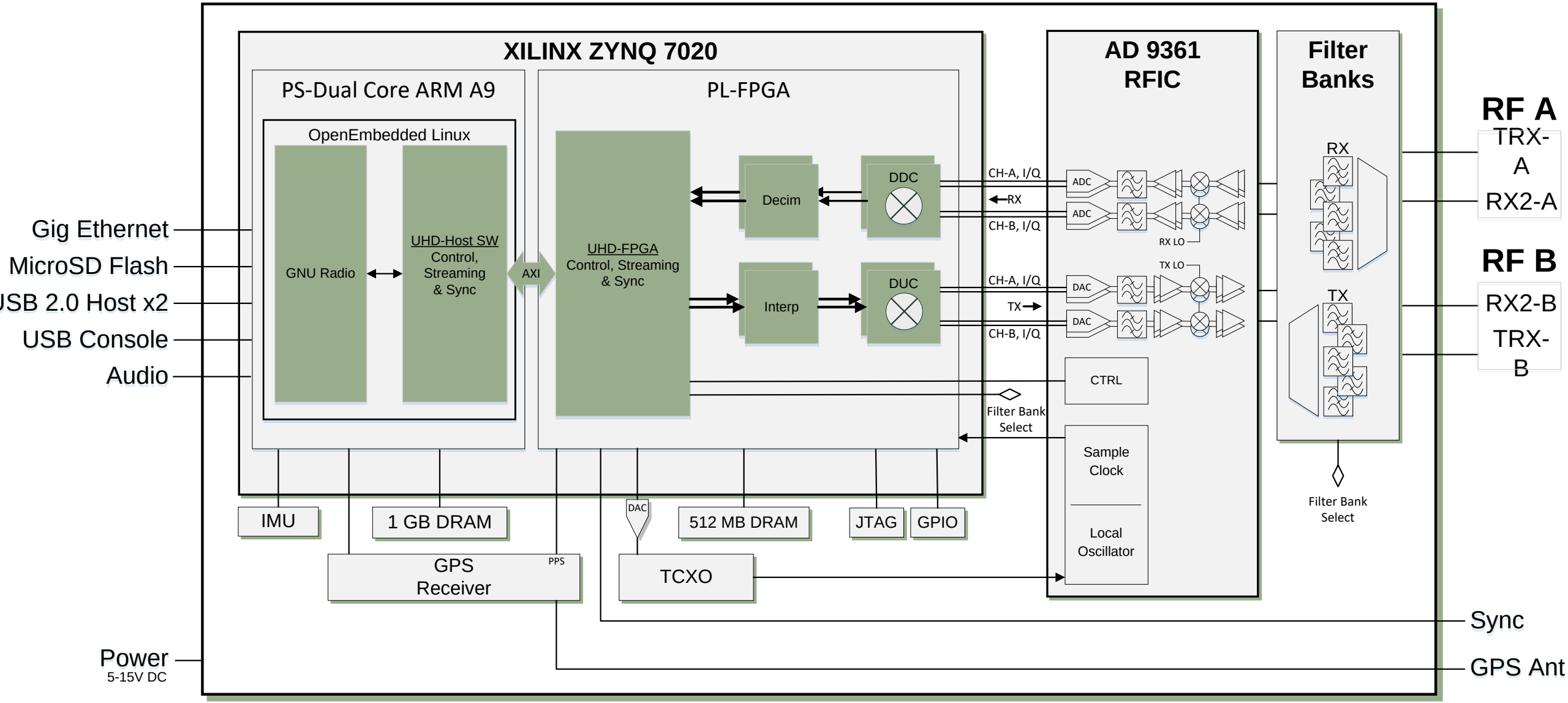
- Mobile Network research
- Network testbeds
- Small, portable, low cost spectrum monitor
- Small UAVs
- Handheld universal communicator

Derivatives

- E313 - Waterproof - IP67
- E330 - 4 Rx to TDOA/DF Applications
- E312 - Battery

USRP E-Series Block Diagram

Ettus Research™ A National Instruments Company



USRP E-Series I/O Connectors

Front Panel

Top Row

- TRX-A – RF A, Transmit/ Receive
- RX2-A – Alternate RX path
- TRX-B – RF B, Transmit/ Receive
- RX2-A – Alternate RX path

Bottom Row

- Power Button
- microSD card port (SDXC capable)
- GPS - antenna port
- Sync - digital line to FPGA
- Audio Jack – 2.5mm mic/speaker

Back Panel

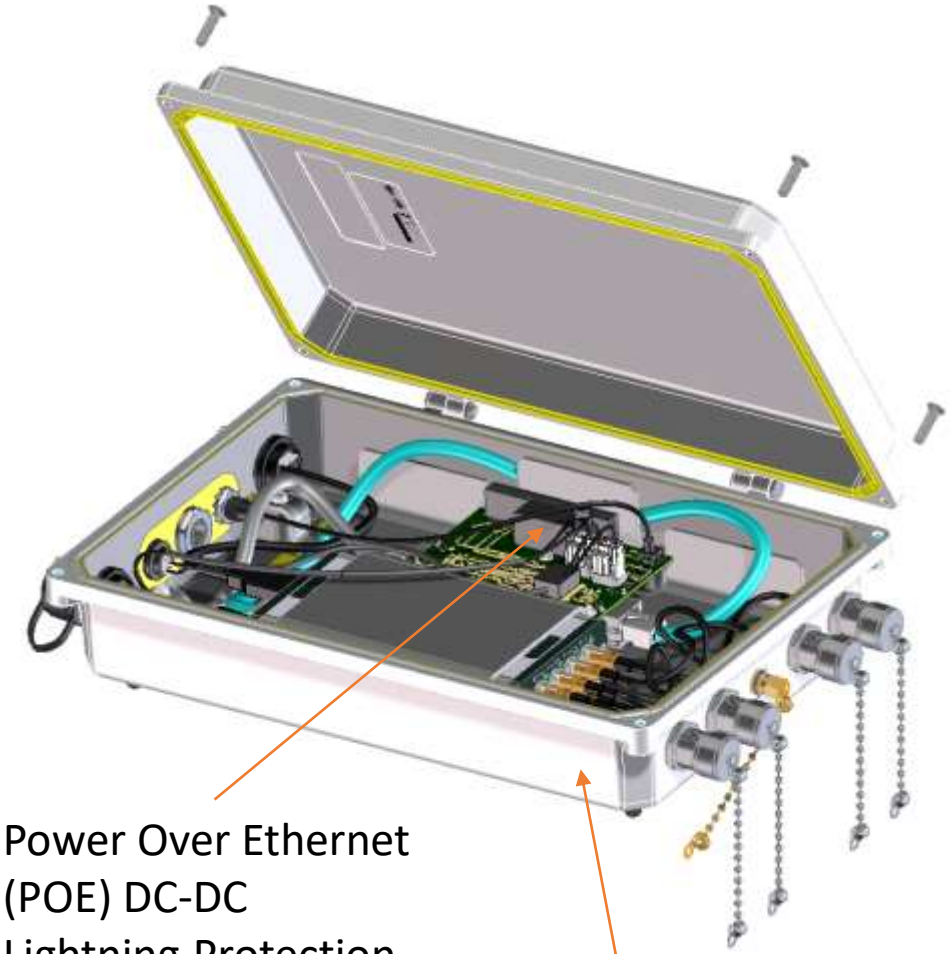
Left to Right

- CONSOLE – serial over USB port
- 2 x USB2 Host – for USB peripherals
- 10/100/1000 – Ethernet interface
- 5-15V DC – Power supply, 2-6W typical

E310 Motherboard

Ettus Research™
A National Instruments Company

E313 IP67 Enclosure



Power Over Ethernet
(POE) DC-DC
Lightning Protection

E310 or E330 Thermally
Connected to Enclosure
-40 – 71C

E312 Battery

- Battery life powered down ~160 hours
- Battery life on idle ~5:30
- Battery life on full load (1x1 TX/RX @5GHz, 1Mhz, 100%) ~2:20
- Battery life on full load (2x2 TX/RX @5GHz, 1Mhz, 100%) ~1:45
- Battery charge time to full ~2:00

E330 4-Channel RX

- 4- channel receive only
- Twin Analog Devices 9361's
- LO is external to 9361's
- Phase coherency maintained during re-tune
- RX filter banks are the same as E310
- Battery-powered version will follow
- MUSIC DF Algorithm example in development

- Target Applications:
- Direction Finding
  - TDOA/FDOA/FOA
- Spectrum Monitoring
- SIGINT/COMINT

USRP X-Series

Ettus Research™
A National Instruments Company

X Series

- Two wideband RF daughterboard slots (2x2 MIMO)
  - Up 160MHz bandwidth per channel
  - Selection covers DC to 6 GHz
  - ADC – 200 Ms/s, 14 bit resolution
  - DAC – 800 Ms/s, 16 bit resolution
- Large, customizable Kintex-7 FPGA
  - USRP X300 - XC7K325T
  - USRP X310 – XC7K410T
- UHD architecture provides compatibility:
  - GNURadio
  - C++ API/Python
  - Other third-party frameworks & applications
- Multiple high-speed interfaces
  - Dual SFP(+) ports for 1/10 Gigabit Ethernet
  - PCIe x4
- Flexible clocking architecture
  - Configurable sample clock
  - Optional GPS-disciplined OCXO
  - Coherent operation with 10 MHz/1 PPS
- Compact and rugged half-wide 1U form factor

Applications

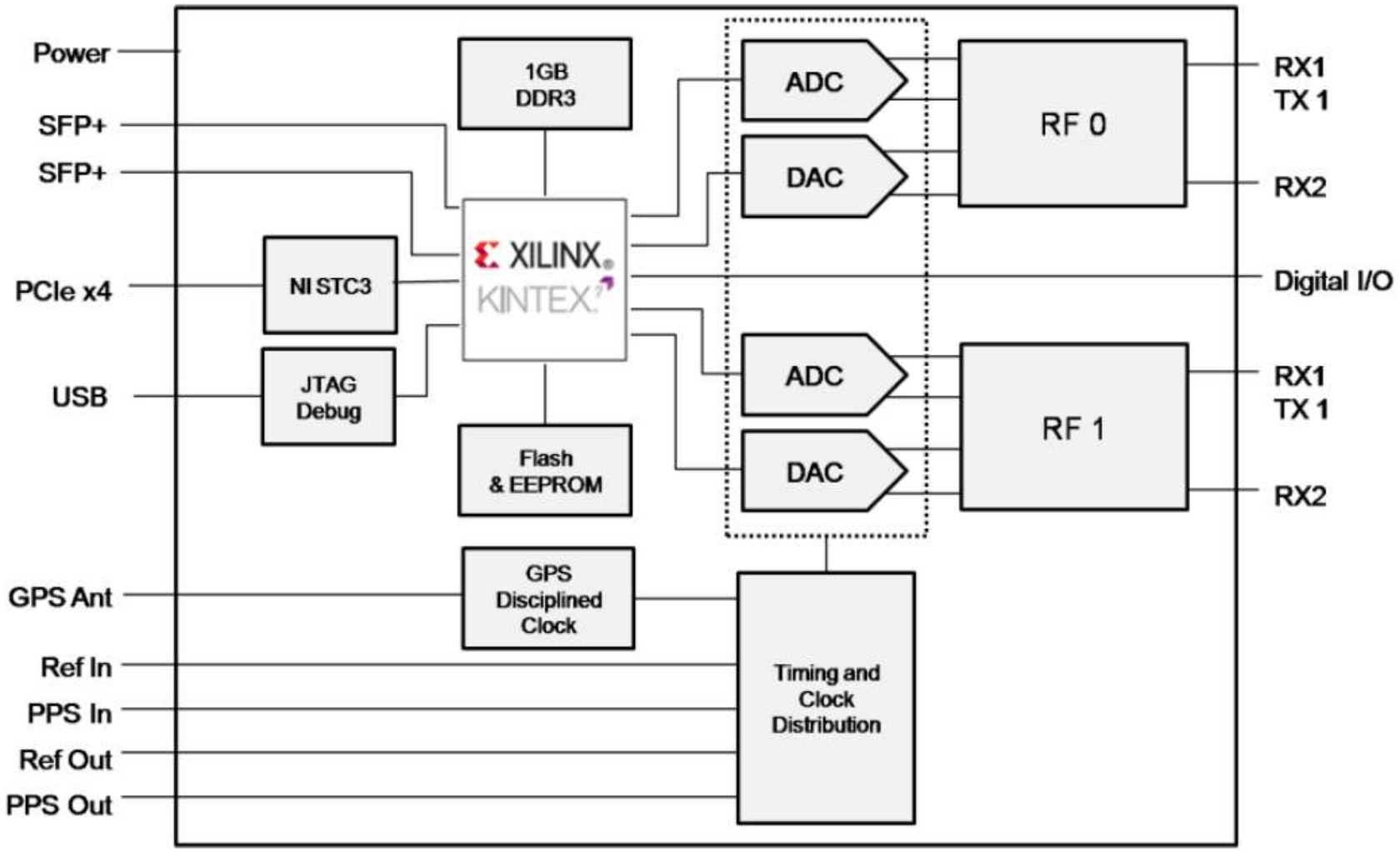
- Advanced Wireless Prototyping
- Massive MIMO Applications
- Passive RADAR
- Signals Intelligence

Front

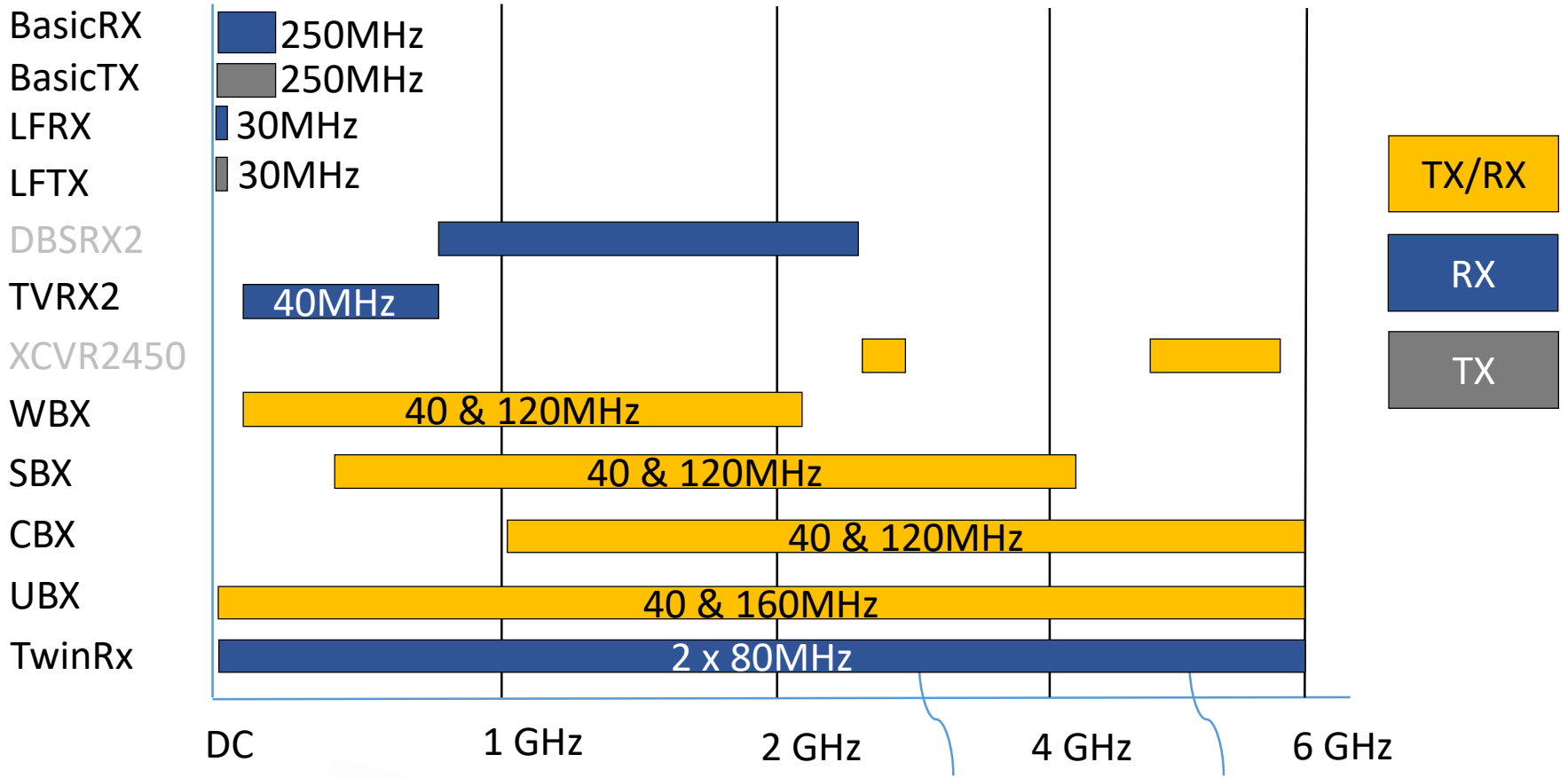
Back

USRP X-Series Block Diagram

Ettus Research™
A National Instruments Company



Daughterboard Frequency




TwinRx


UBX


WBX


DBSRX2


BasicRX

TwinRX

August 2016

What is the Twin RX?

- Daughterboard for X300/X310
- In a single X300 or X310 you can accommodate 2 TwinRX modules, giving 4 channels of phase coherent measurements for DF.
- The TwinRx is Ettus' first Superheterodyne design with residual spurious performance better than -80 dBm.

Target Applications:

- Direction Finding
  - TDOA/FDOA/FOA
- Spectrum Monitoring
- SIGINT/COMINT

TwinRX Key Specifications

• Converter mode	Superheterodyne
• Frequency Range:	10 MHz to 6 GHz
• Rx Channels:	2
• Max Bandwidth:	80 MHz
• Preselection Filters:	8
• IP3 at 2.4 GHz:	2 dBm at 10 dB Noise Figure
• Dynamic Range at 2.4 GHz	110 dB
• Min. noise figure at 2.4 GHz	5.5 dB
• Residual Spurs	<-90 dBm
• Phase Noise	-86 dBc/Hz @10KHz offset
• Power dissipation	10W
• LO1 & LO2 sharing	
• Phase coherent LO & ADC sample clock sharing	
• Compatible with Ettus X300 & X310	
• UHD Driver	
• Operating Temperature: 0 to 55°C	

All measurements @2.4GHz LO unless otherwise specified
Residual spurs are with input terminated, 0dB attenuation and maximum gain

2 TwinRX Daughtercards inside X310
4 RX channels total with LO Sharing

Ettus Research™
A National Instruments Company

Ettus Research™
A National Instruments Company

RF Network on Chip (RFNoC) Introduction

Host-Based SDR – Current Situation

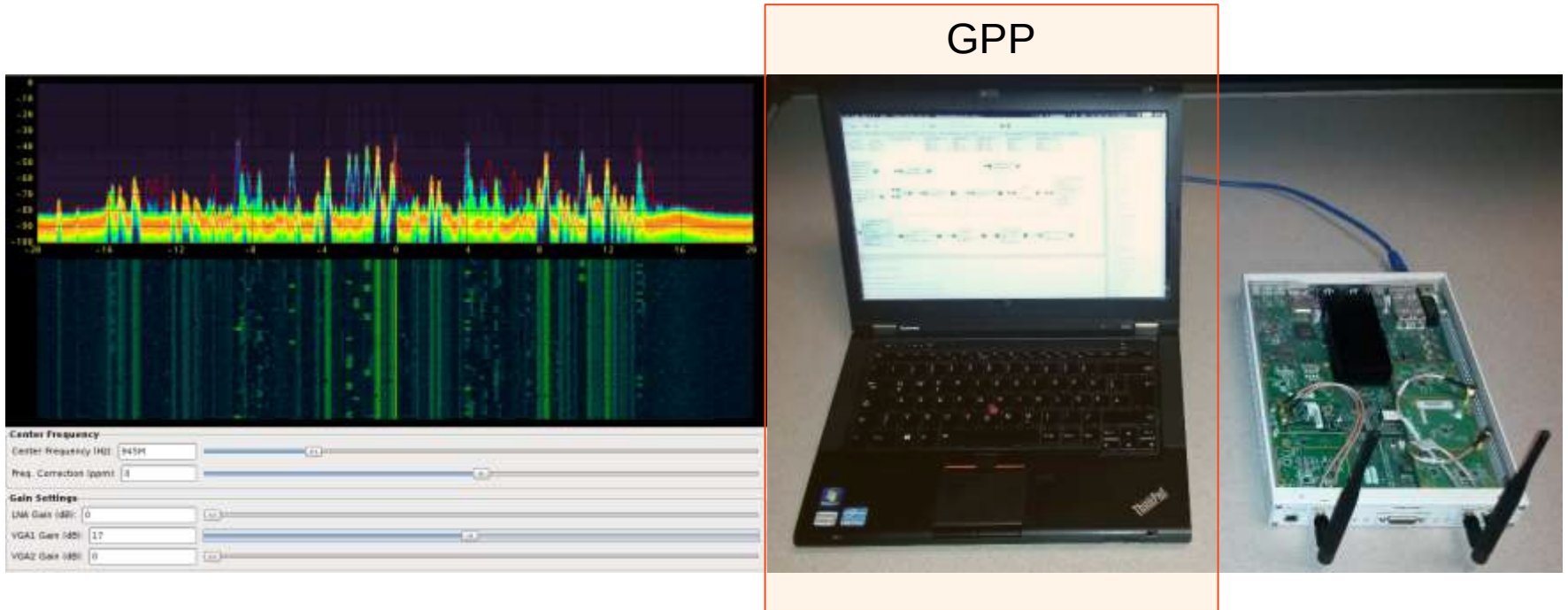
Ettus Research™
A National Instruments Company

- PC + Flexible RF Hardware + SDR Framework

Host-Based SDR – Current Situation

Ettus Research™ A National Instruments Company

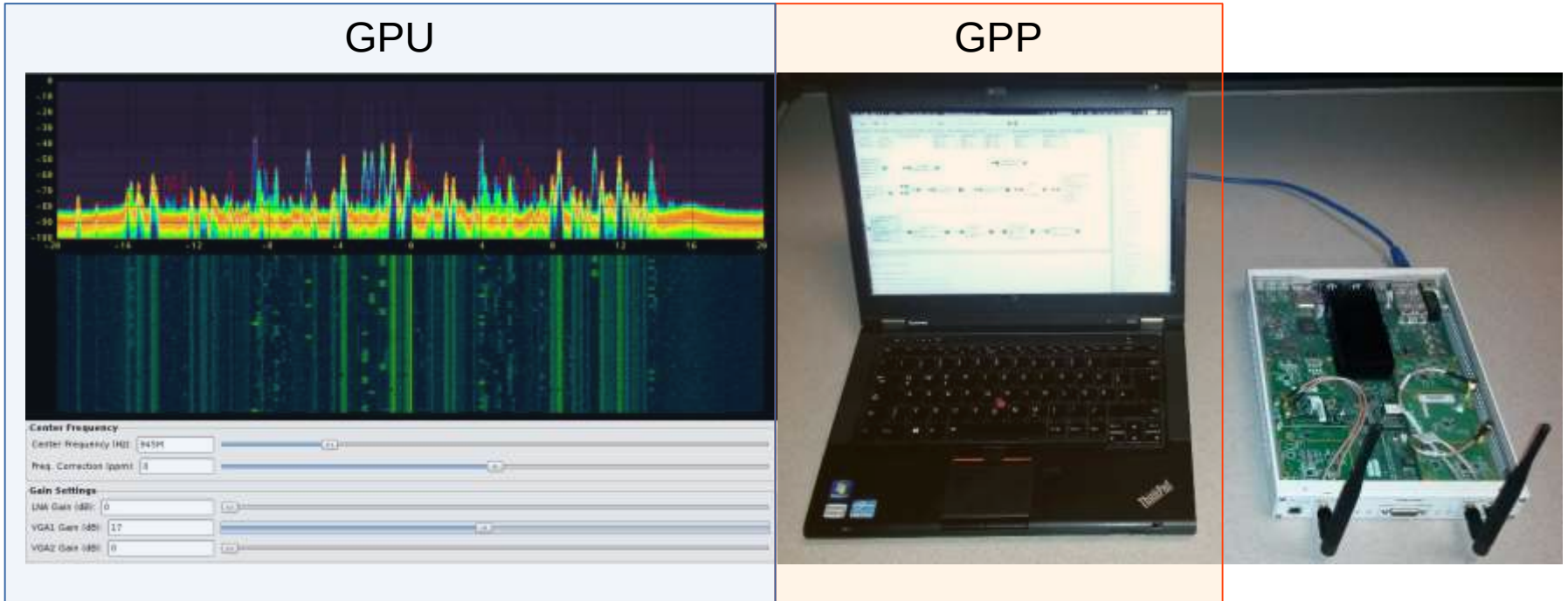
- PC + Flexible RF Hardware + SDR Framework
  - GPP: Multi-core + SIMD, GNU Radio



Host-Based SDR – Current Situation

Ettus Research™ A National Instruments Company

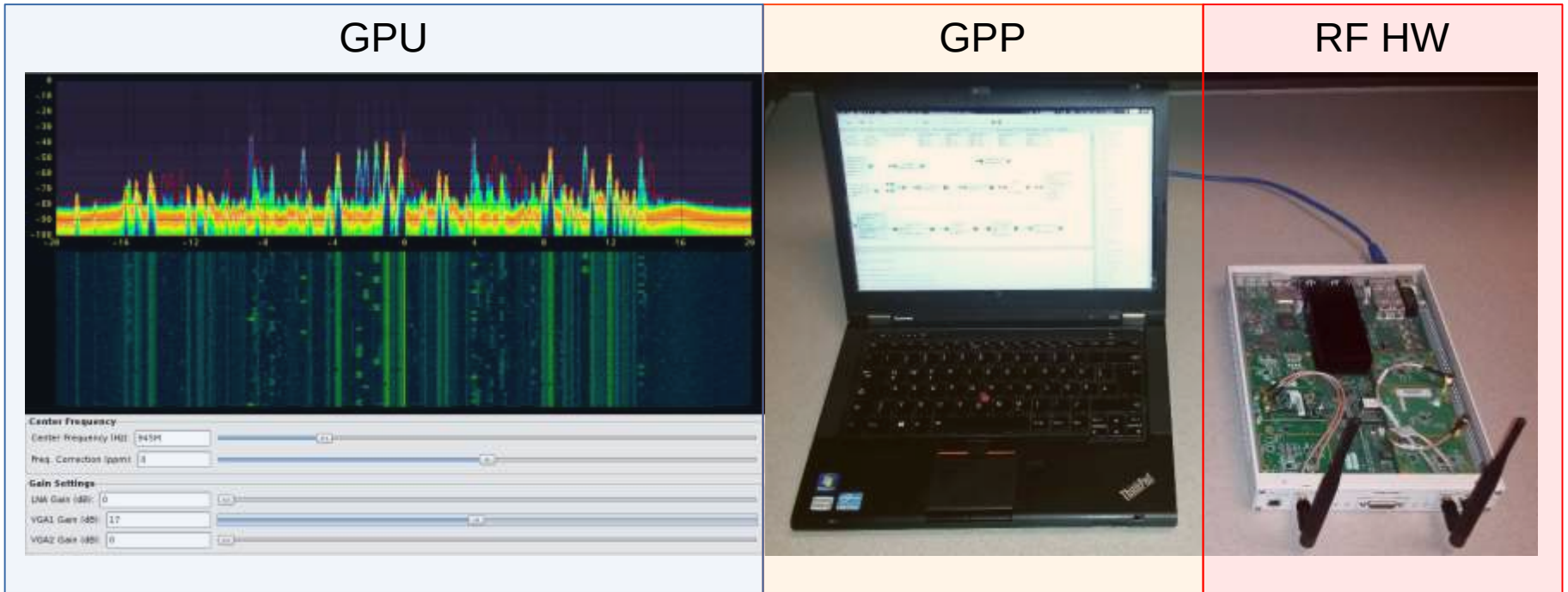
- PC + Flexible RF Hardware + SDR Framework
  - GPP: Multi-core + SIMD, GNU Radio
  - GPU: High performance FP, OpenCL, gr-fosphor



Host-Based SDR – Current Situation

Ettus Research™ A National Instruments Company

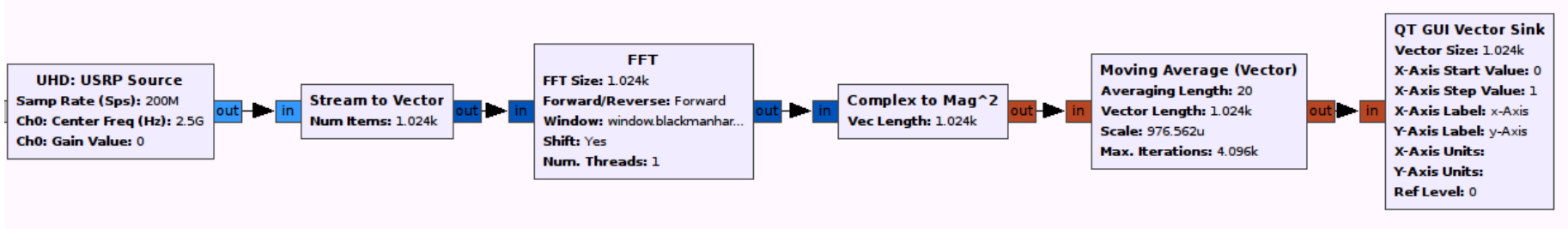
- PC + Flexible RF Hardware + SDR Framework
  - GPP: Multi-core + SIMD -- GNU Radio
  - GPU: High performance FP -- OpenCL, gr-fosphor
  - RF HW: Wide bandwidth, large FPGA -- DDC, DUC



SDR in Practice: Spectral Analysis

Ettus Research™
A National Instruments Company

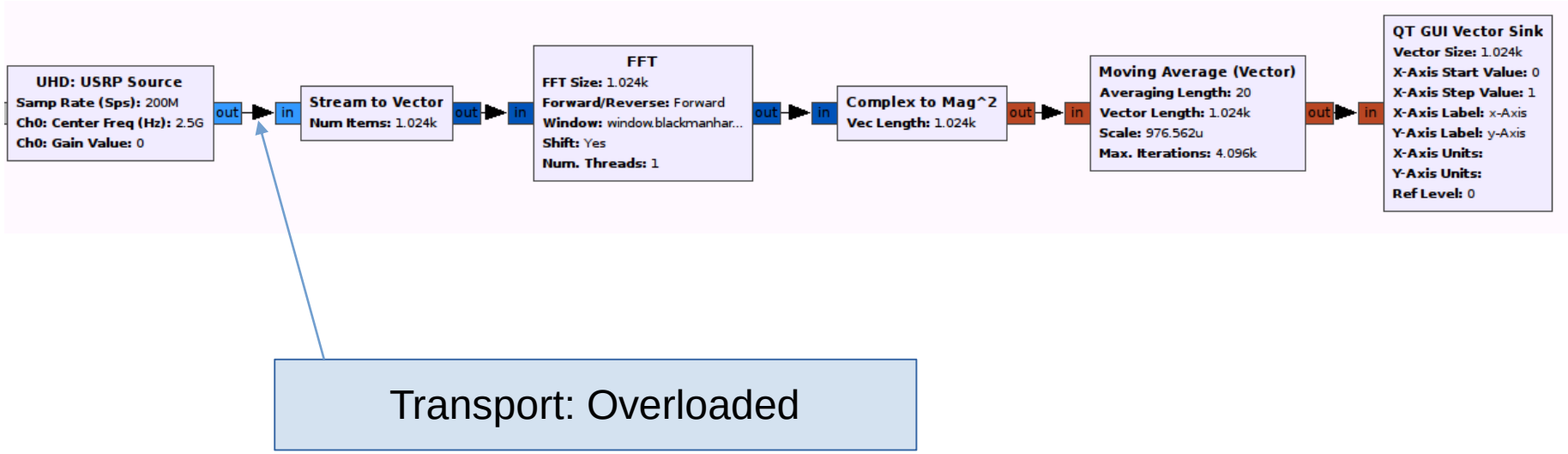
- Power Spectral Density Estimate
  - Welch's Algorithm
  - Simple in Theory: 200 MHz real-time



SDR in Practice: Spectral Analysis

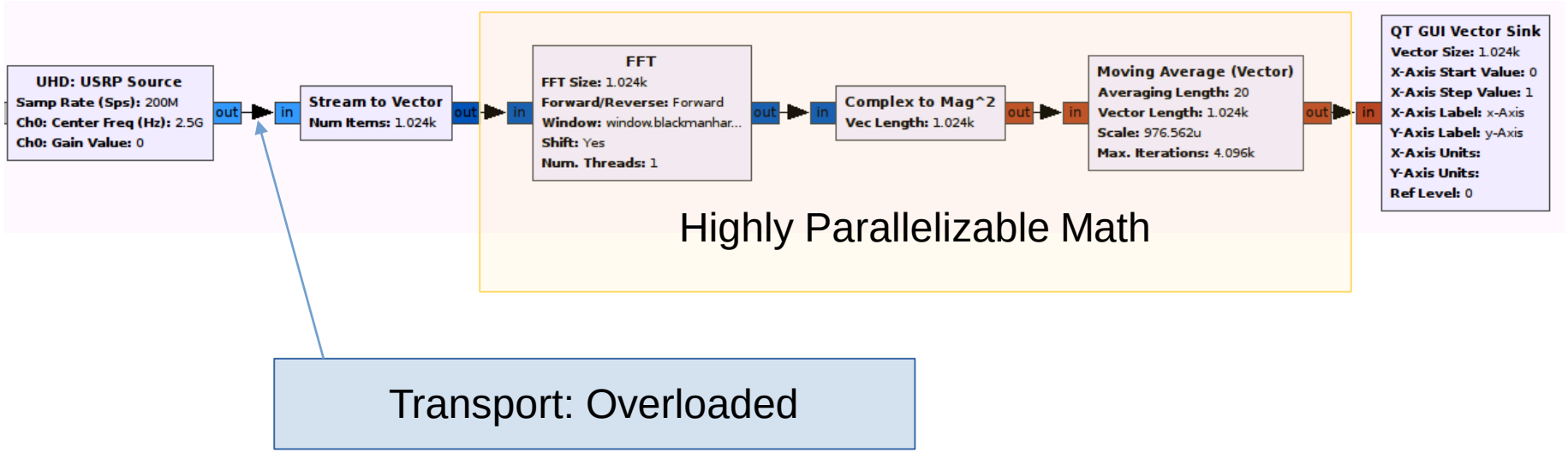
Ettus Research™ A National Instruments Company

- Power Spectral Density Estimate
  - Welch's Algorithm
  - Simple in Theory: 200 MHz real-time
    - 200Ms/s * 4 bytes/samp → **800 MB/sec**
    - Have to use 10 GigE and careful tuning



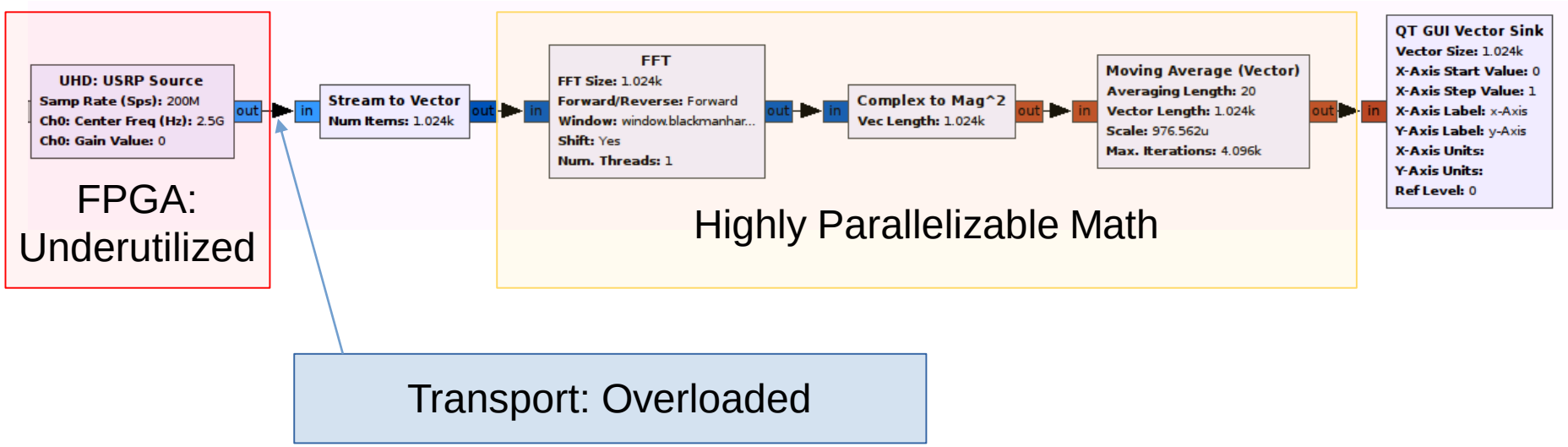
SDR in Practice: Spectral Analysis

- Power Spectral Density Estimate
  - Welch's Algorithm
  - Simple in Theory: 200 MHz real-time
    - 200Ms/s * 4 bytes/samp → **800 MB/sec**
    - Have to use 10 GigE and careful tuning



SDR in Practice: Spectral Analysis

- Power Spectral Density Estimate
  - Welch's Algorithm
  - Simple in Theory: 200 MHz real-time
    - 200Ms/s * 4 bytes/samp → **800 MB/sec**
    - Have to use 10 GigE and careful tuning



USRP: A White Box?

Ettus Research™
A National Instruments Company

- Everything USRP is open source, available online (code, firmware, schematics)
- Contains a large, capable FPGA!
- Why do customers not use it?

FPGAs: Hard to use… slow to develop

Ettus Research™
A National Instruments Company

Domain vs FPGA Experts

Ettus Research™ A National Instruments Company

- FPGA development is not a requirement of a communications engineering curriculum
- Math in FPGAs is hard
- Complicated system architecture

almost pure-noise channels. This intuition is clarified more by the following inequality. It is shown in [1] that for any B-DMC W,

$$1 - I(W) \leqslant Z(W) \leqslant \sqrt{1 - I(W)^2} \tag{2}$$

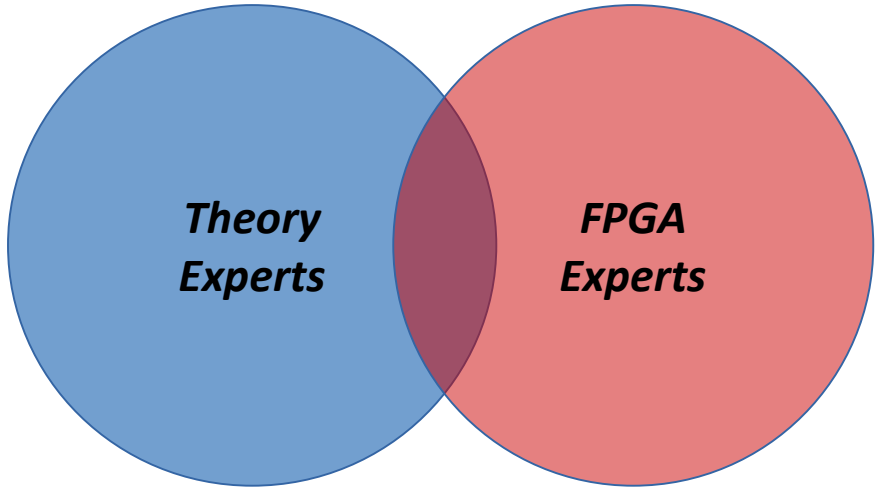
where $I(W)$ is the symmetric capacity of W.

Let W^N denote the channels that results from N independent copies of W i.e. the channel $\langle \{0,1\}^N, \mathscr{Y}^N, W^N \rangle$ given by

$$W^N(y_1^N | x_1^N) \stackrel{\text{def}}{=} \prod_{i=1}^{N} W(y_i | x_i) \tag{3}$$

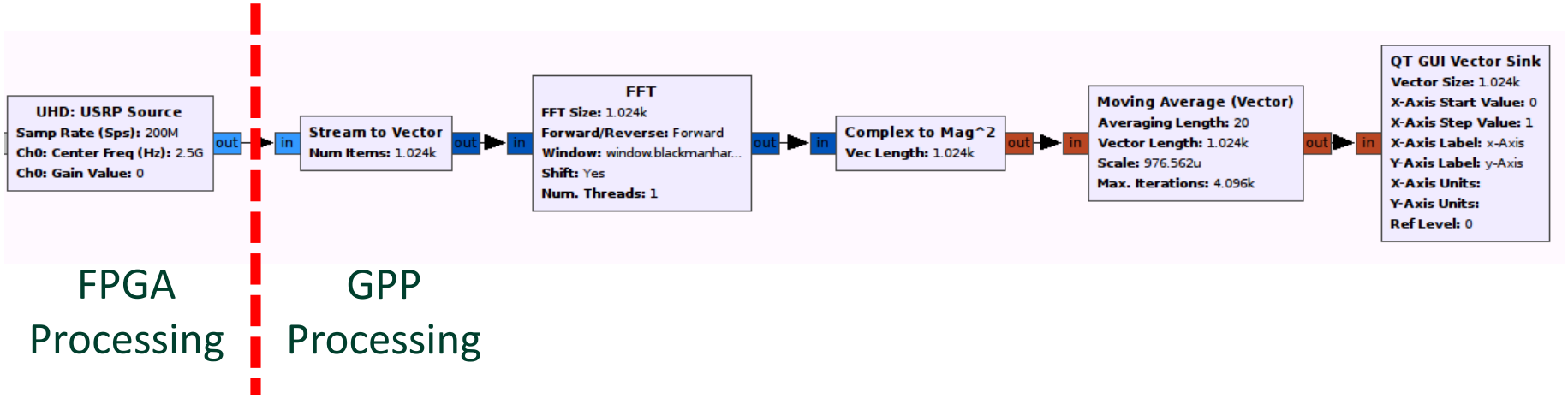
where $x_1^N = (x_1, x_2, \ldots, x_N)$ and $y_1^N = (y_1, y_2, \ldots, y_N)$. Then the *combined* channel $\langle \{0,1\}^N, \mathscr{Y}^N, \widetilde{W} \rangle$ is defined with transition probabilities given by

$$\widetilde{W}(y_1^N | u_1^N) \stackrel{\text{def}}{=} W^N(y_1^N | u_1^N G_N) = W^N(y_1^N | u_1^N R_N G^{\otimes n})$$



Goal

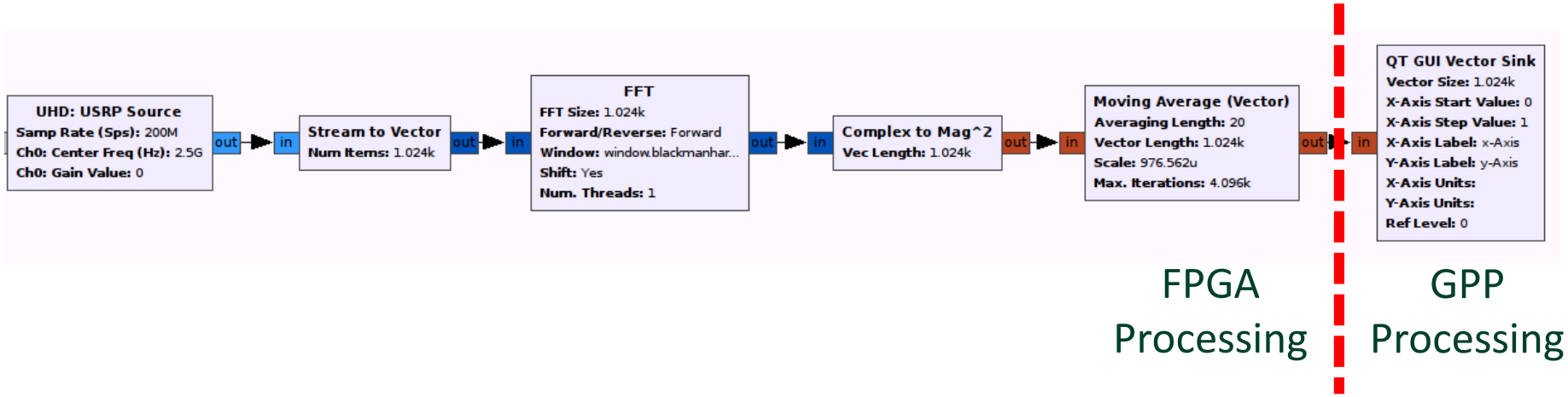
- Heterogeneous Processing
- Support composable and modular designs using GPP, FPGA, & beyond
- Maintain ease of use
- Tight integration with popular SDR frameworks



Goal

Ettus Research™ A National Instruments Company

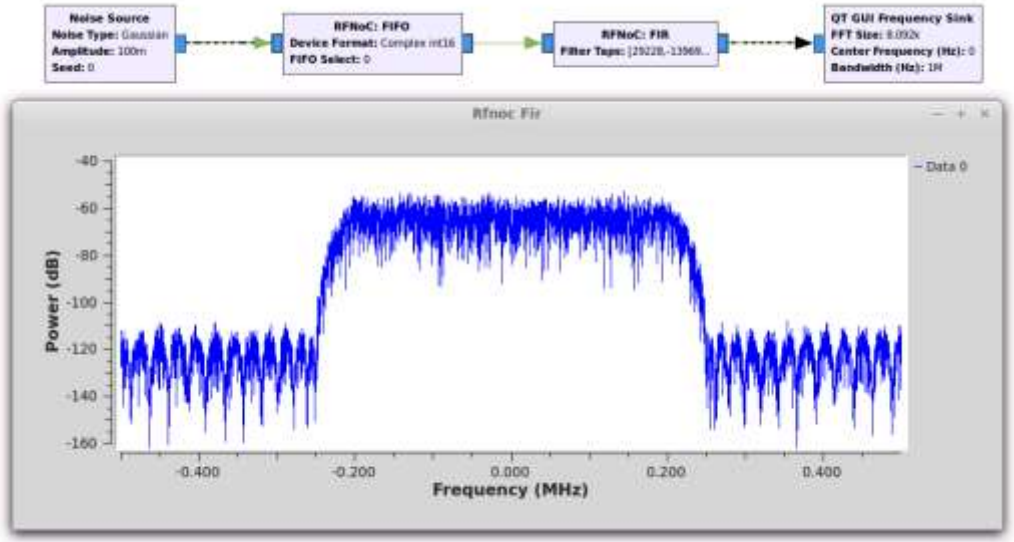
- Heterogeneous Processing
- Support composable and modular designs using GPP, FPGA, & beyond
- Maintain ease of use
- Tight integration with popular SDR frameworks



RFNoC: RF Network on Chip

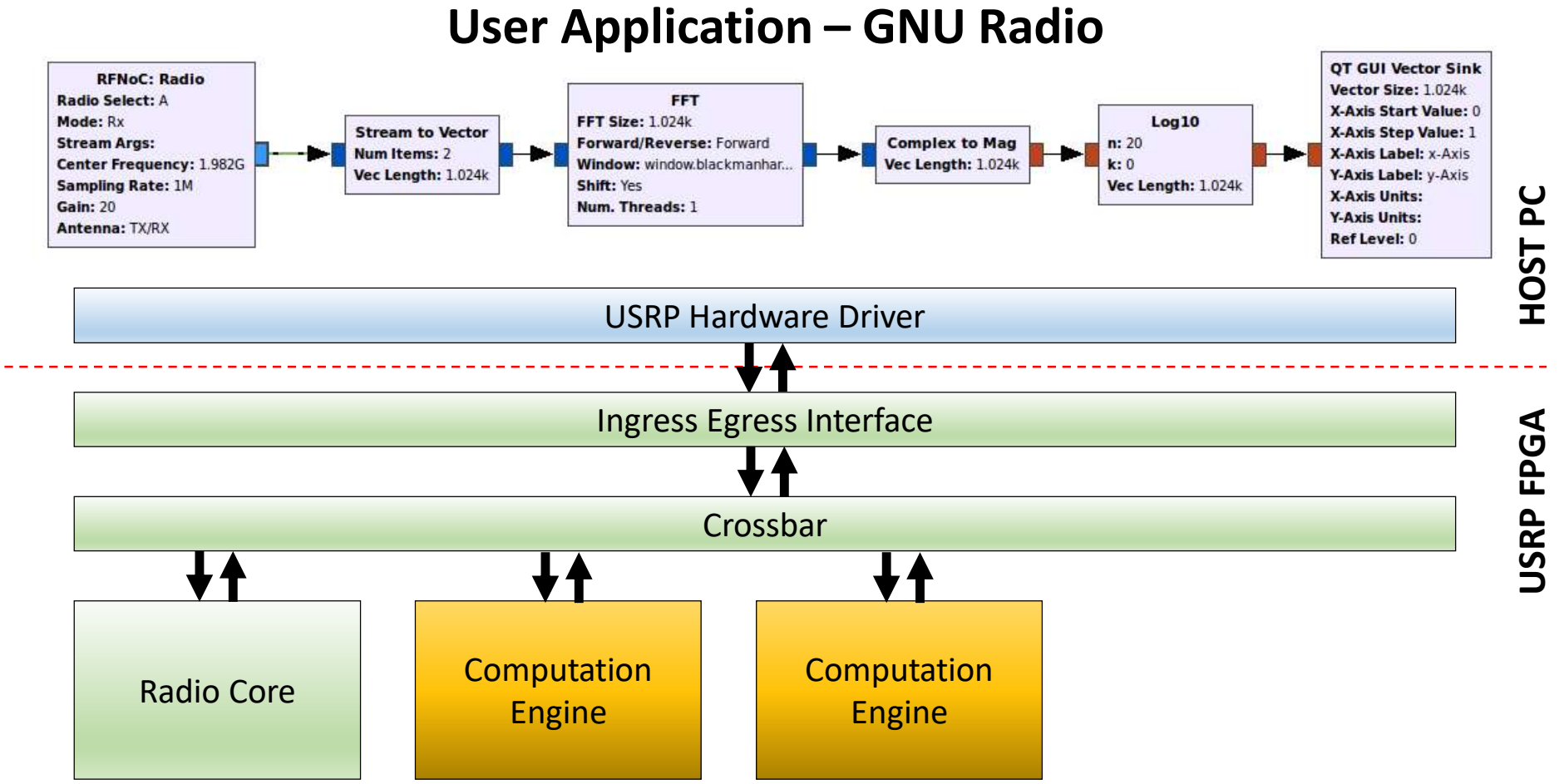
Ettus Research™
A National Instruments Company

- Make USRP FPGA acceleration easier
  - Software API + FPGA infrastructure
    - Handles FPGA – Host communication / dataflow
    - Provides user simple software and HDL interfaces
  - Scalable design for massive distributed processing
  - Fully supported in GNU Radio



RFNoC Architecture

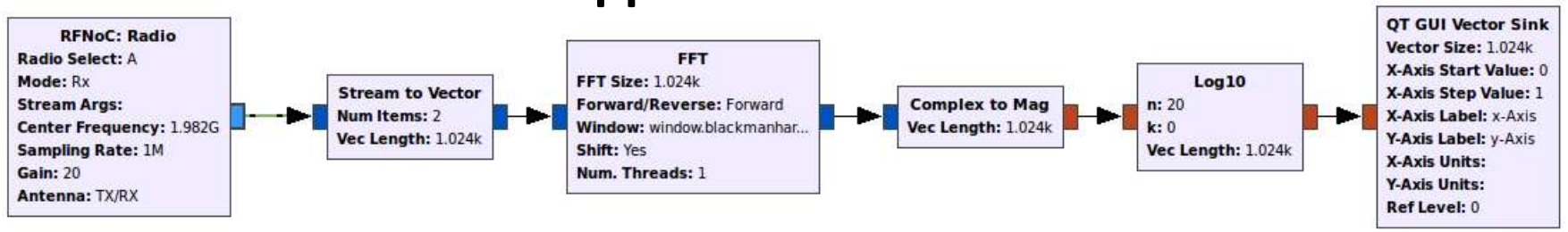
Ettus Research™ A National Instruments Company



RFNoC Architecture

Ettus Research™ A National Instruments Company

User Application – GNU Radio



HOST PC

USRP Hardware Driver

Ingress Egress Interface

Crossbar

Radio Core

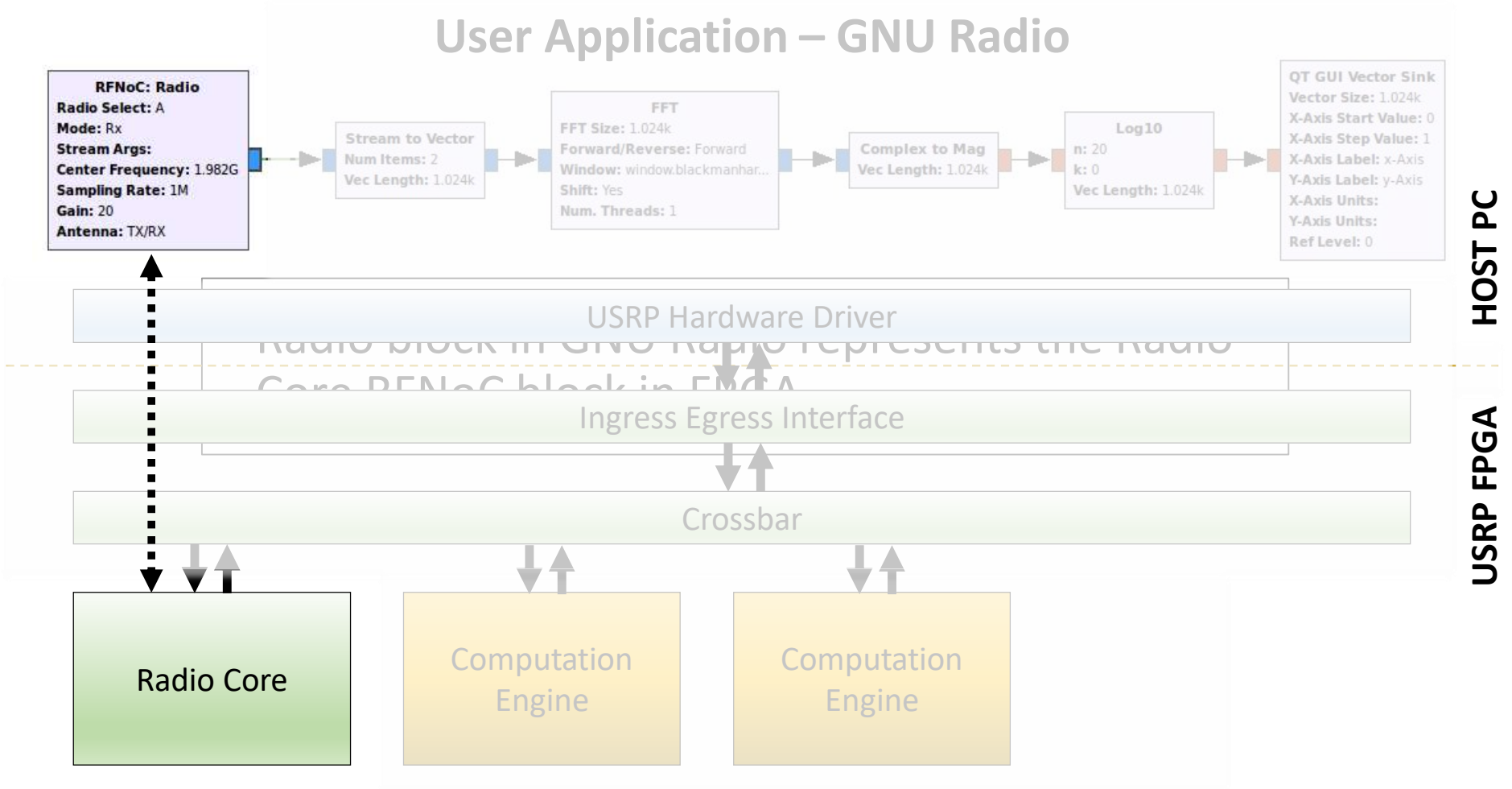
Computation Engine

Computation Engine

USRP FPGA

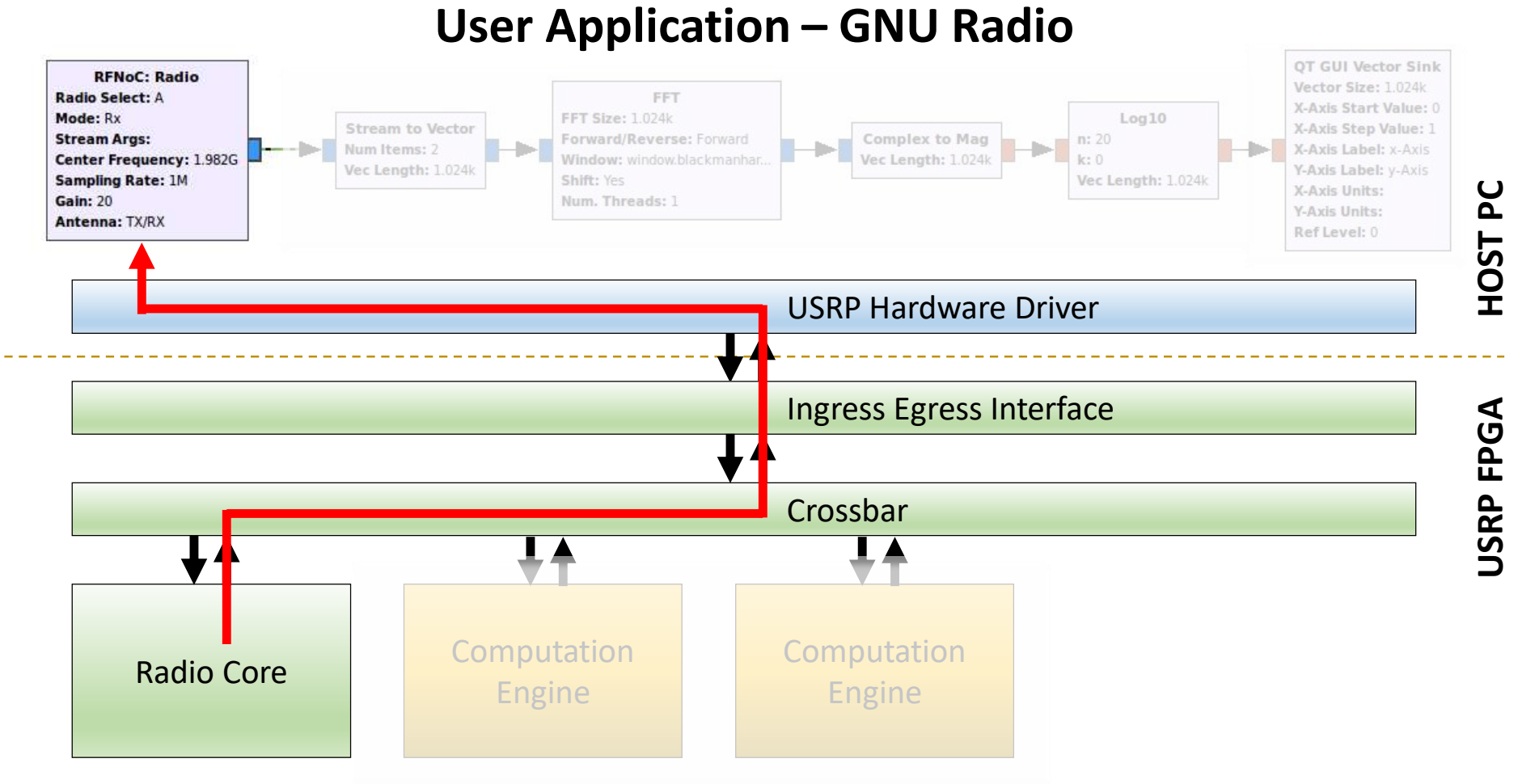
RFNoC Architecture

Ettus Research™ A National Instruments Company

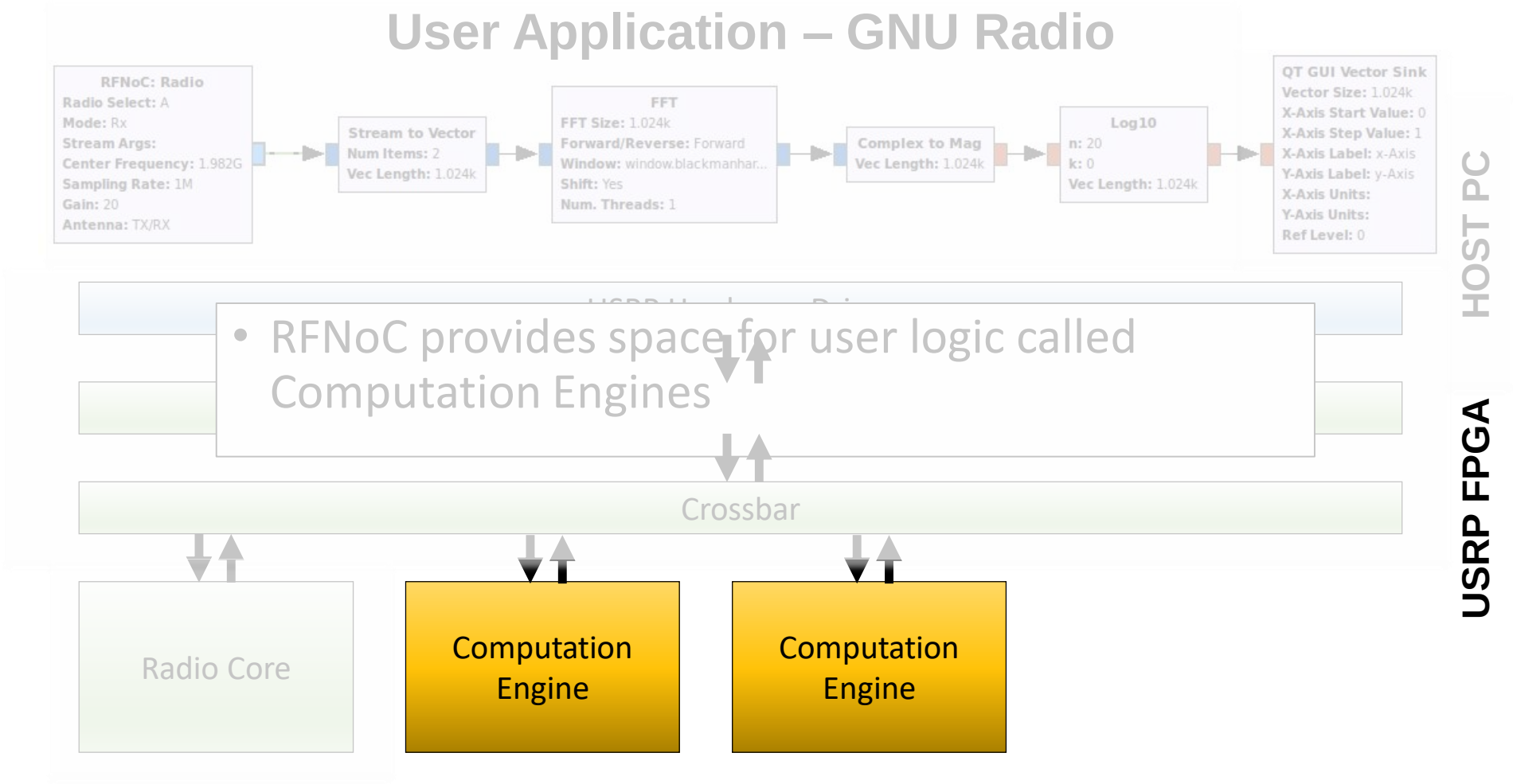


RFNoC Architecture

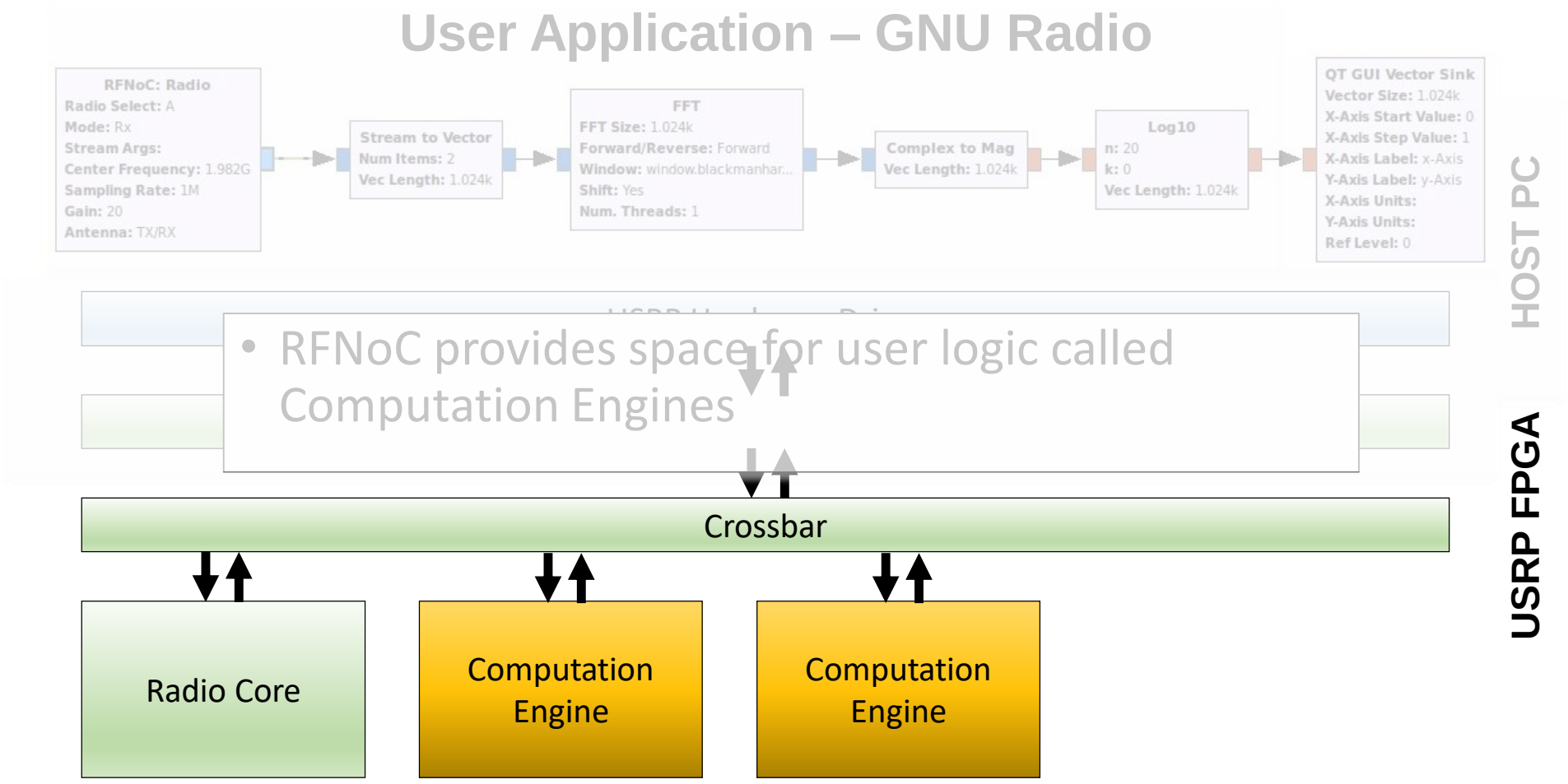
Ettus Research™ A National Instruments Company



RFNoC Architecture

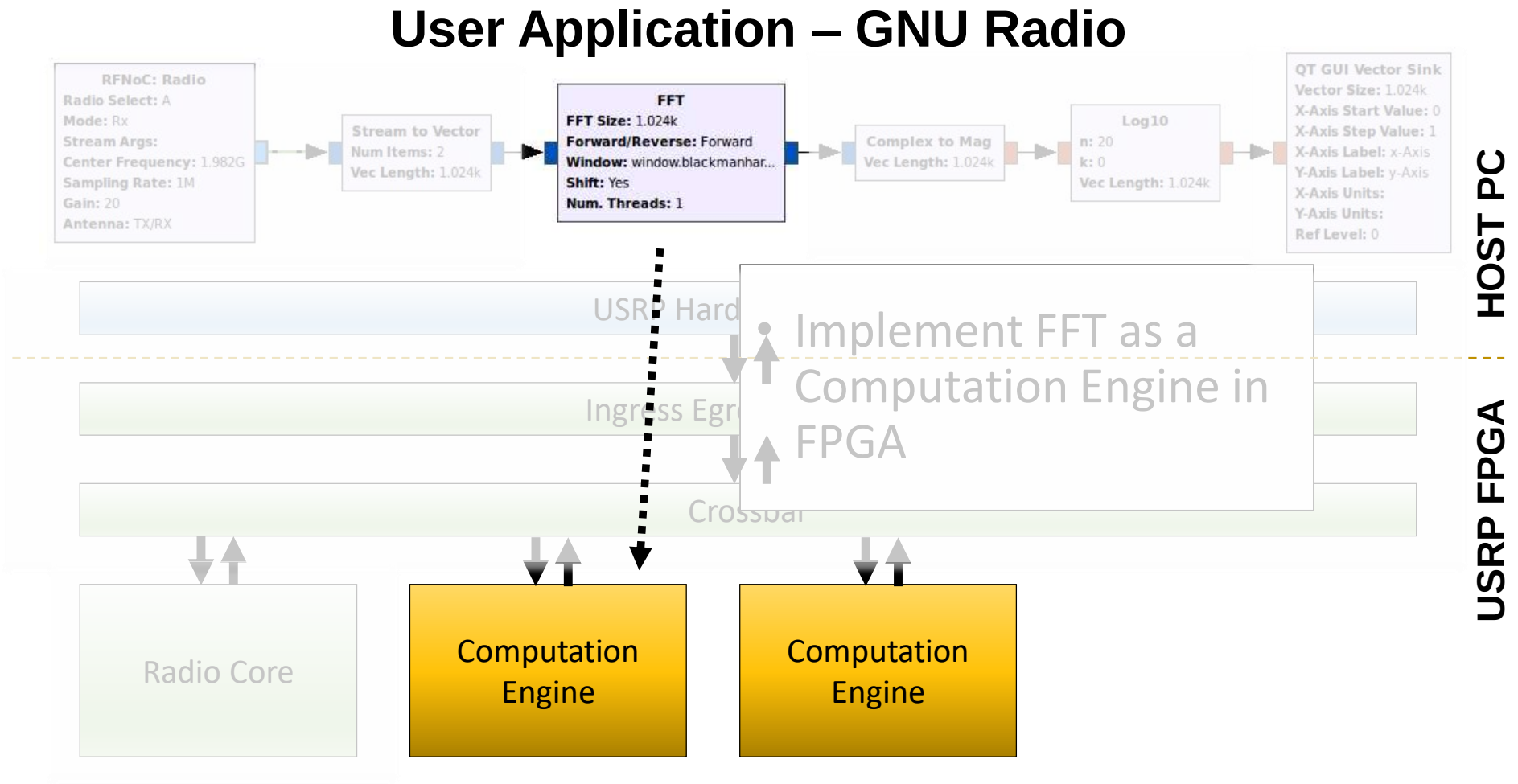


RFNoC Architecture



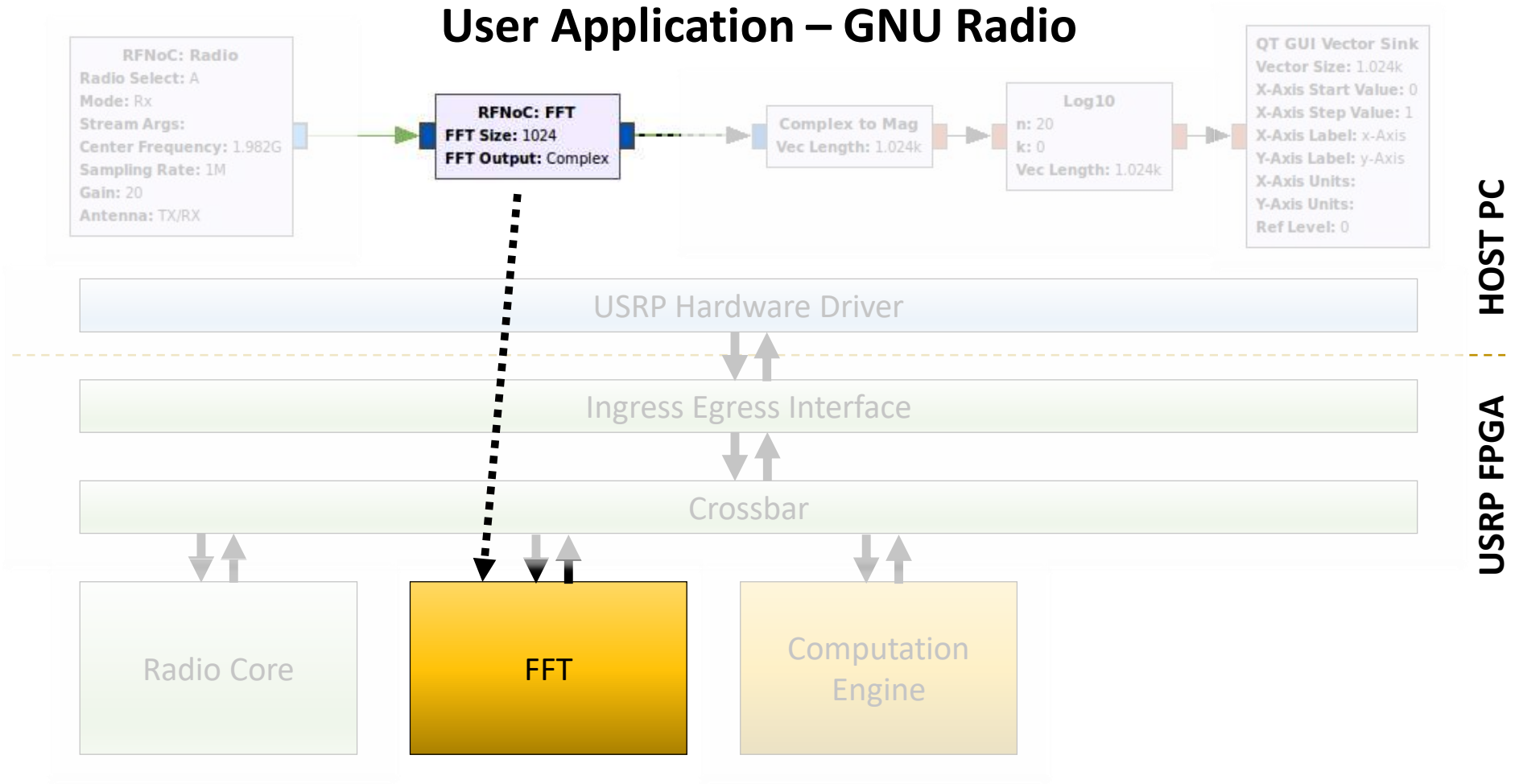
RFNoC Architecture

Ettus Research™ A National Instruments Company

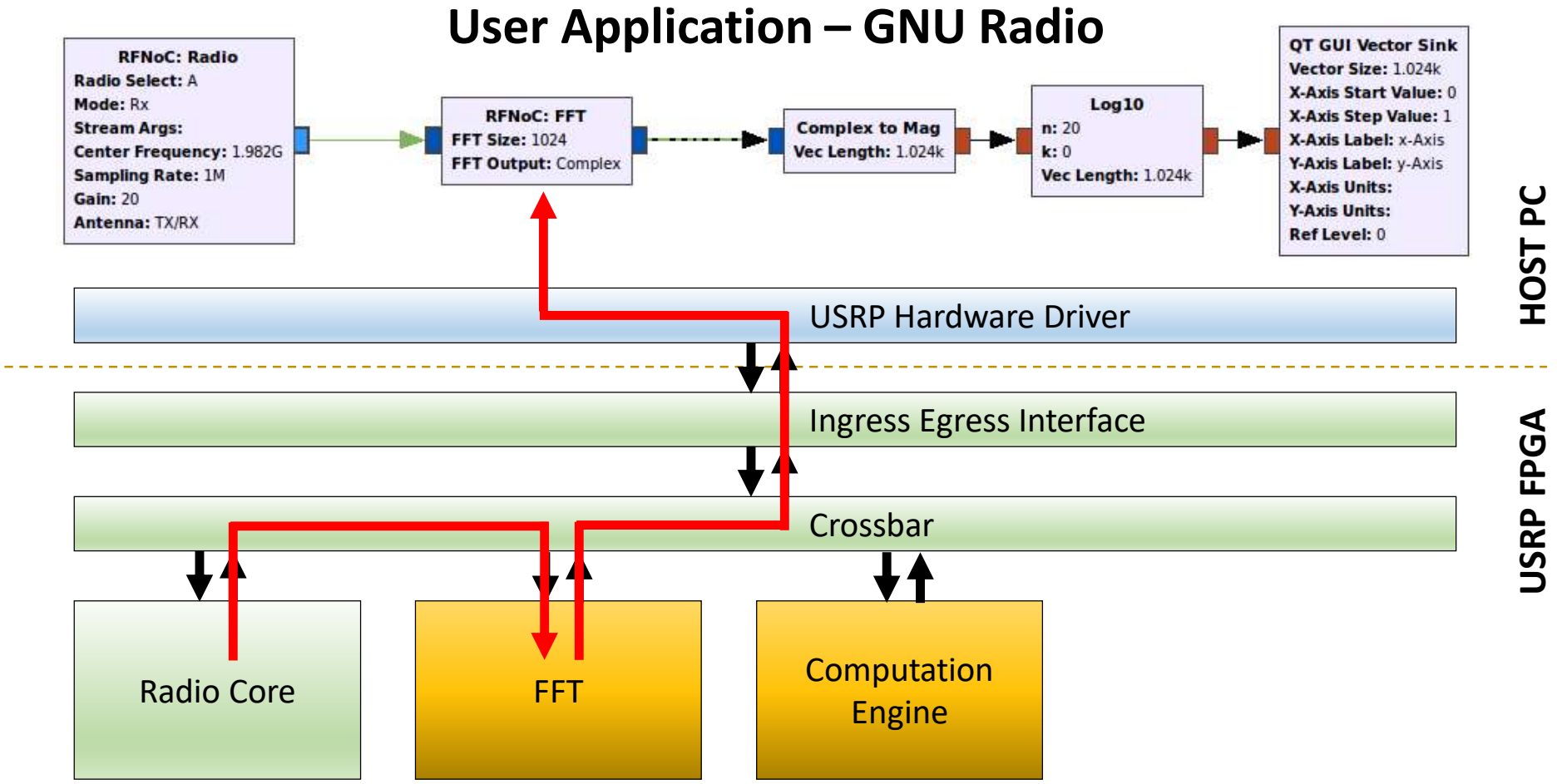


RFNoC Architecture

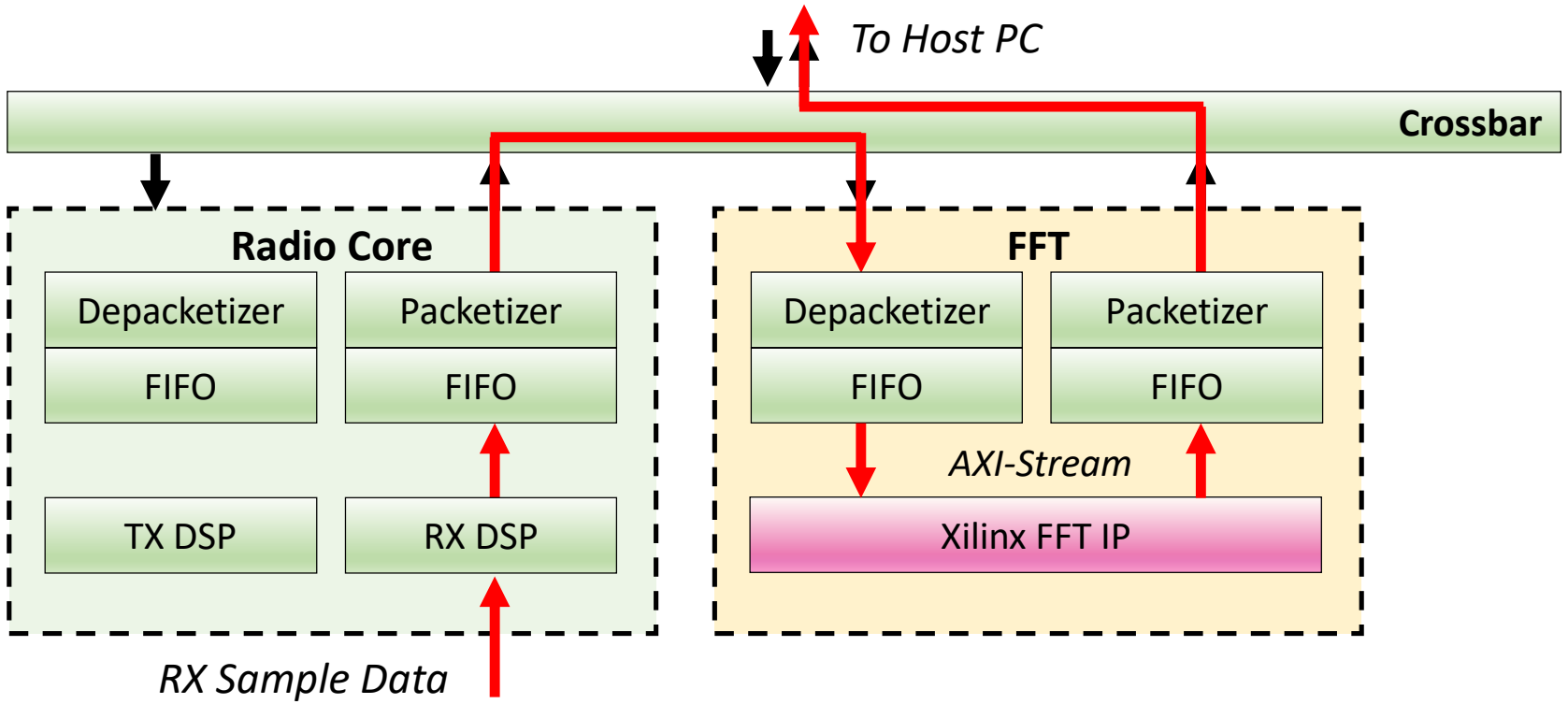
Ettus Research™ A National Instruments Company



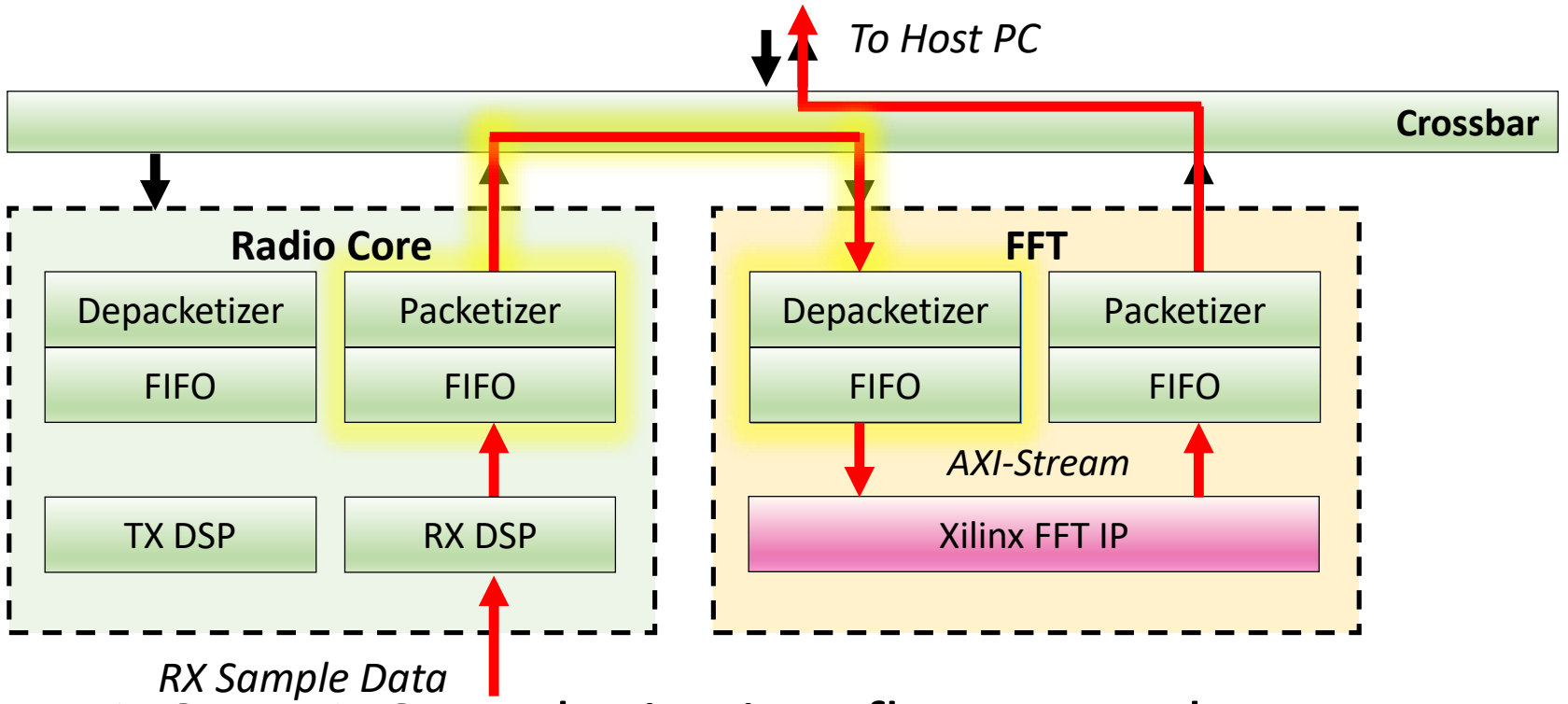
RFNoC Architecture



Computation Engine

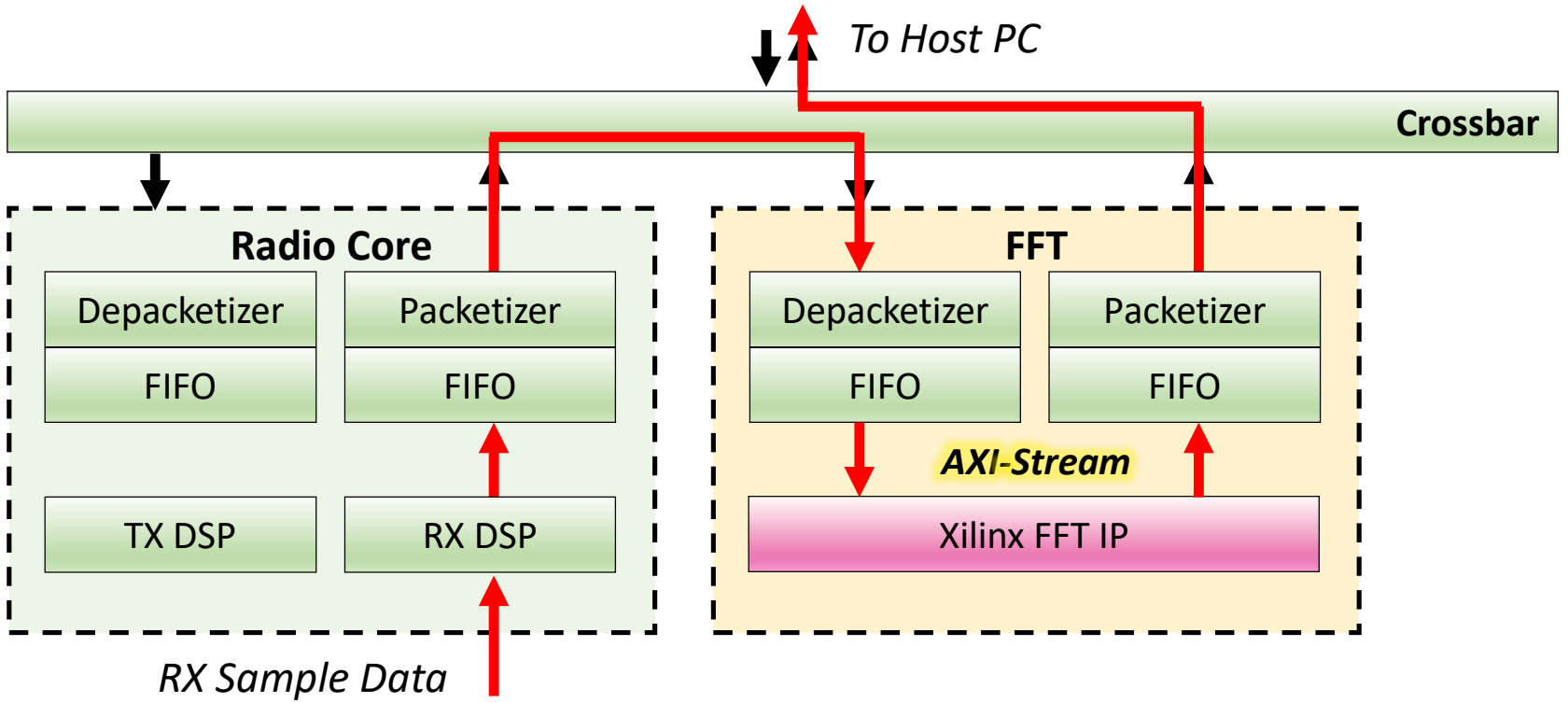


Computation Engine



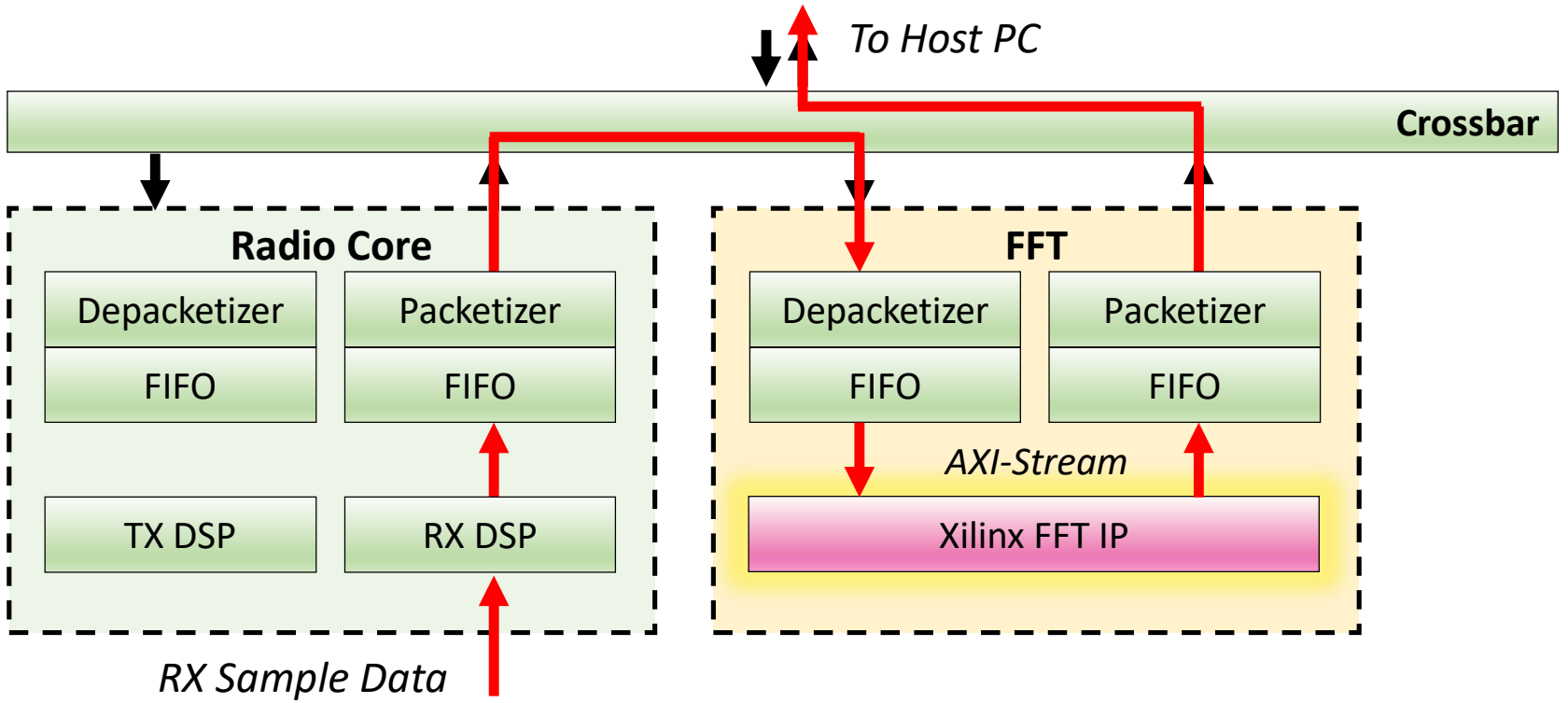
- FIFO to FIFO, packetization, flow control
- Provided by RFNoC infrastructure

Computation Engine



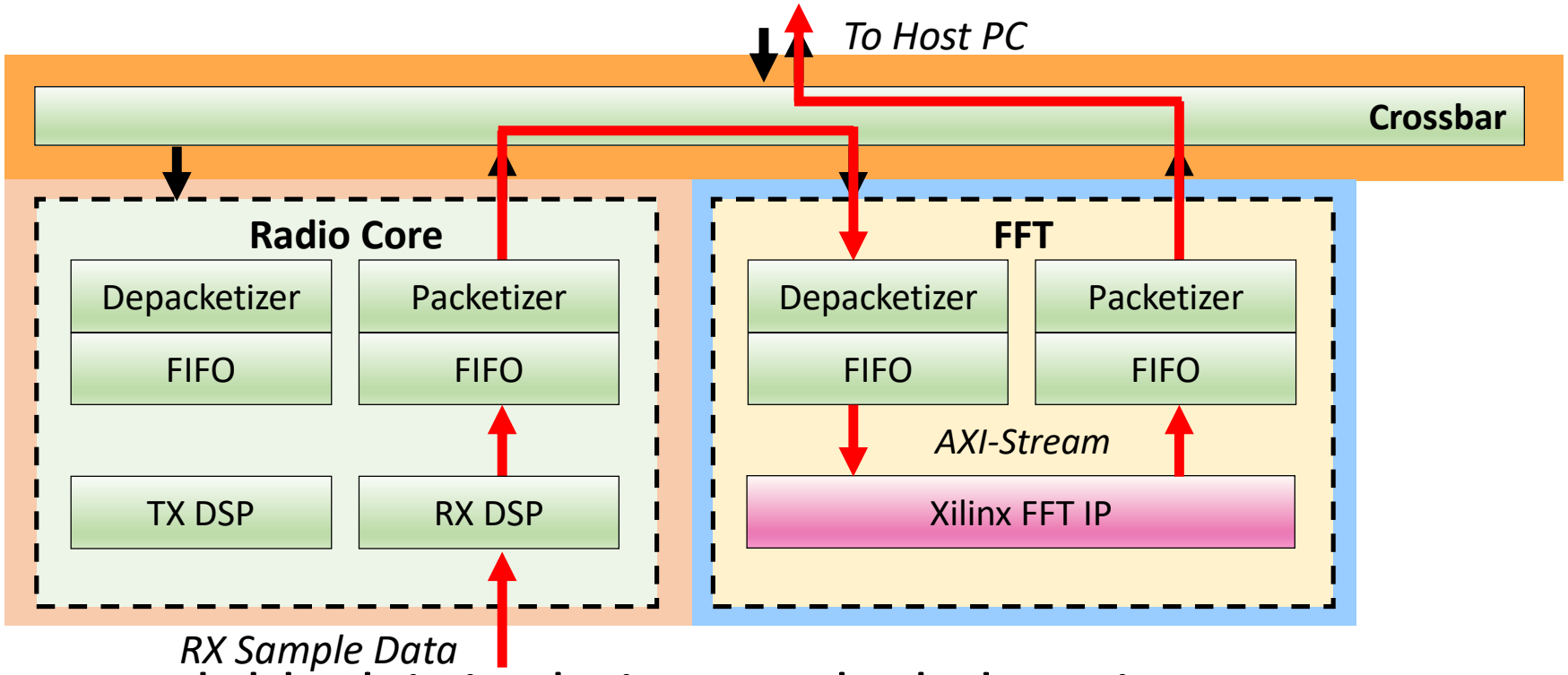
- User interfaces to RFNoC via AXI-Stream
  - Industry standard (ARM), easy to use
    - Large library of existing IP cores

Computation Engine



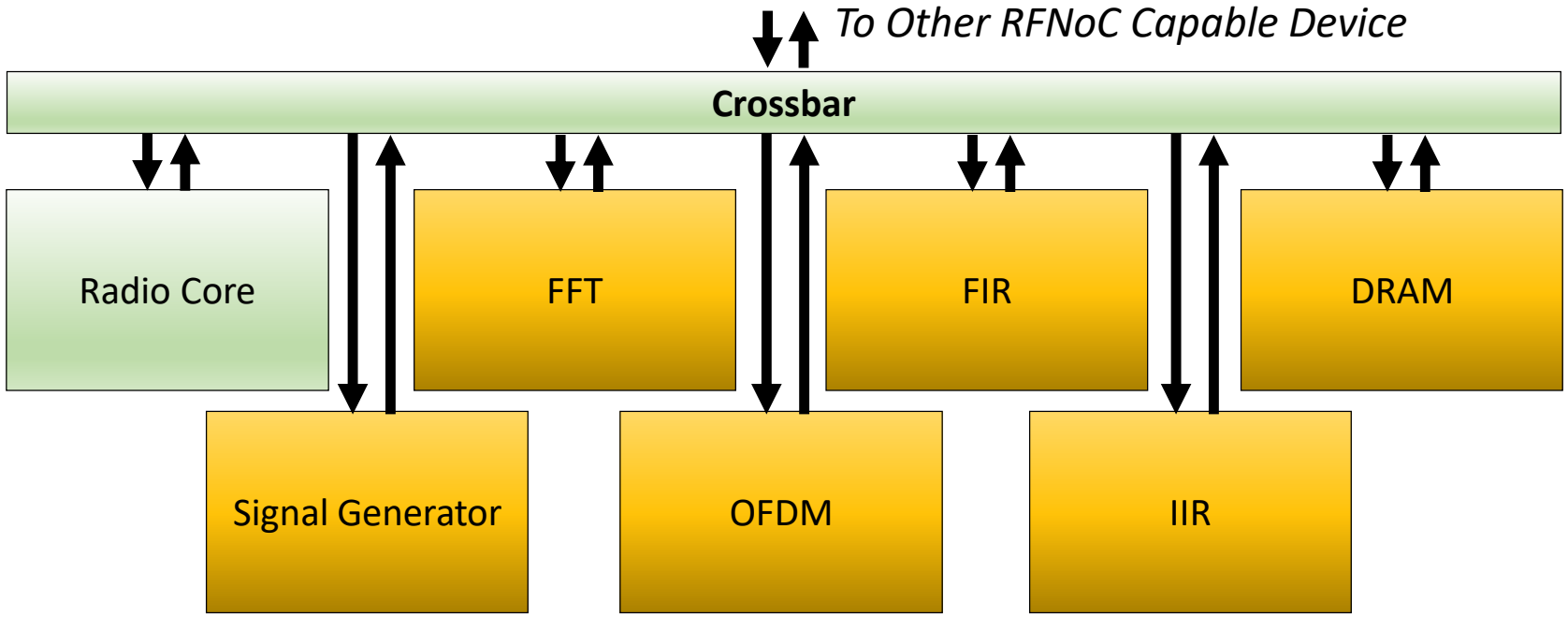
- User writes their own HDL or drops in IP
  - Multiple AXI-Streams, Control / Status registers

Computation Engine



- Each block is in their own clock domain
  - Improve block throughput, timing
  - Interface to Crossbar has clock crossing FIFOs

Many Types of CEs



- Many computation engines
- Not limited to one crossbar, one device
  - Scales across devices for massive distributed processing

Available Blocks

Ettus Research™ A National Instruments Company

- Radio Interface
- FIFO
- DDC/DUC
- FFT
- FIR
- Window
- Vector IIR
- Moving Average
- Log Power
- Decimator
- Split Stream
- Packet Resizer
- Null Source/Sink
- Signal Generator
- Fosphor (Real Time Spectrum Analyzer)
- OFDM
- Burst Packet
- DRAM
- Third Party IP

Potential future blocks

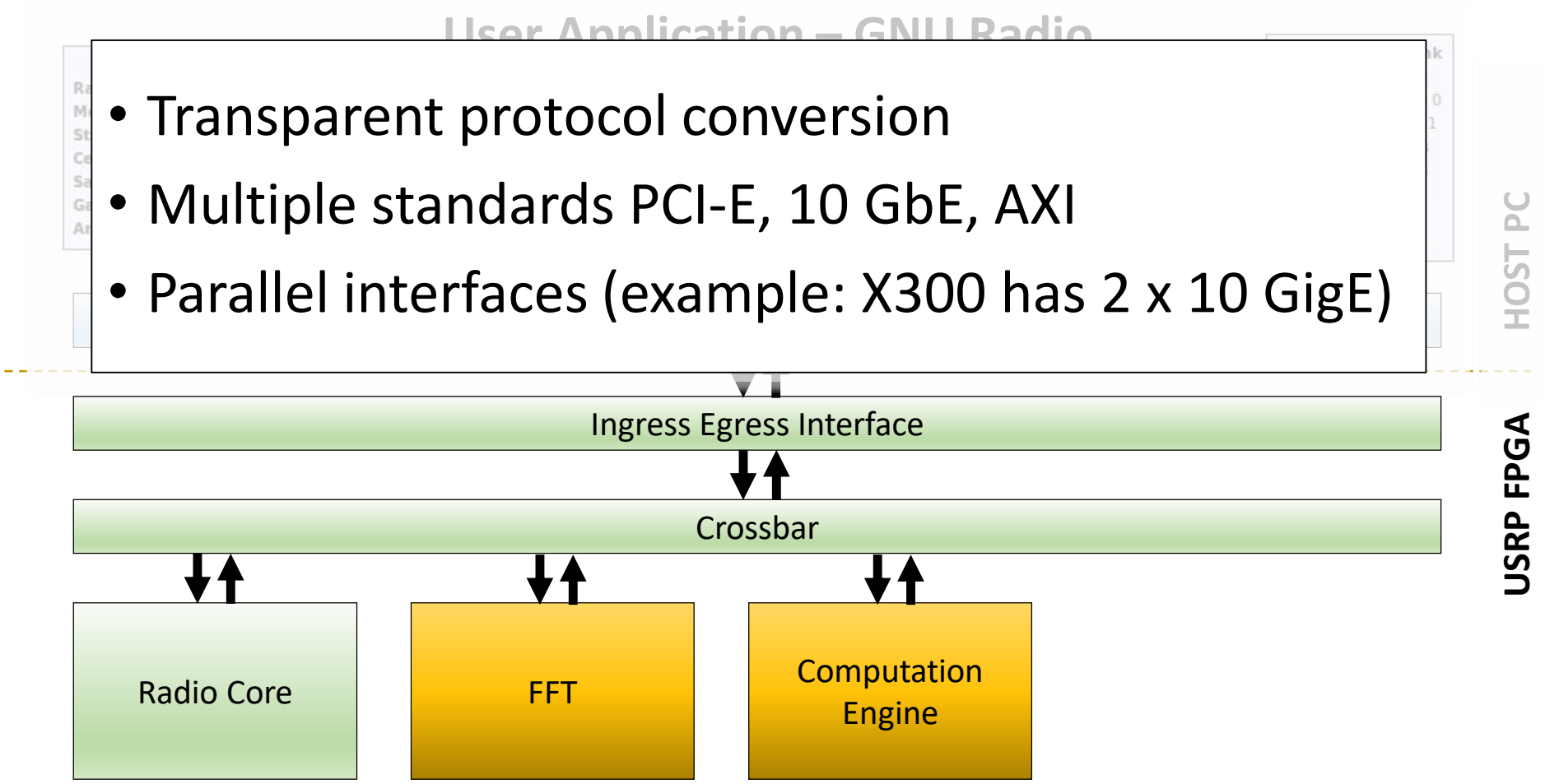
Ettus Research™ A National Instruments Company

- **Channelizer**
- **Direction Finding (TDOA, TOA, FDOA, AOA)**
- **Wi-Fi reference design**
- **Demodulators/Modulators**
- **SATCOM specific protocols (DVB-SX2)**
- Crypto Core
- Compression/Depression
- Soft Processor (MicroBlaze)
- FFT block with time-multiplexed so it can be oversampled.
- Chipscope interface
- Others thoughts?

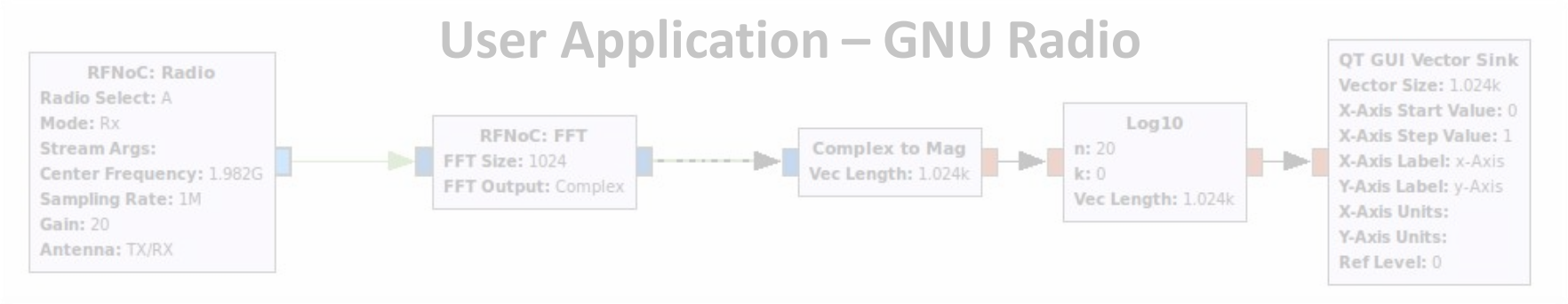
- Partial Reconfiguration -Not a block but high desired…

RFNoC Architecture

Ettus Research™ A National Instruments Company



RFNoC Architecture



HOST PC

USRP Hardware Driver

USRP FPGA

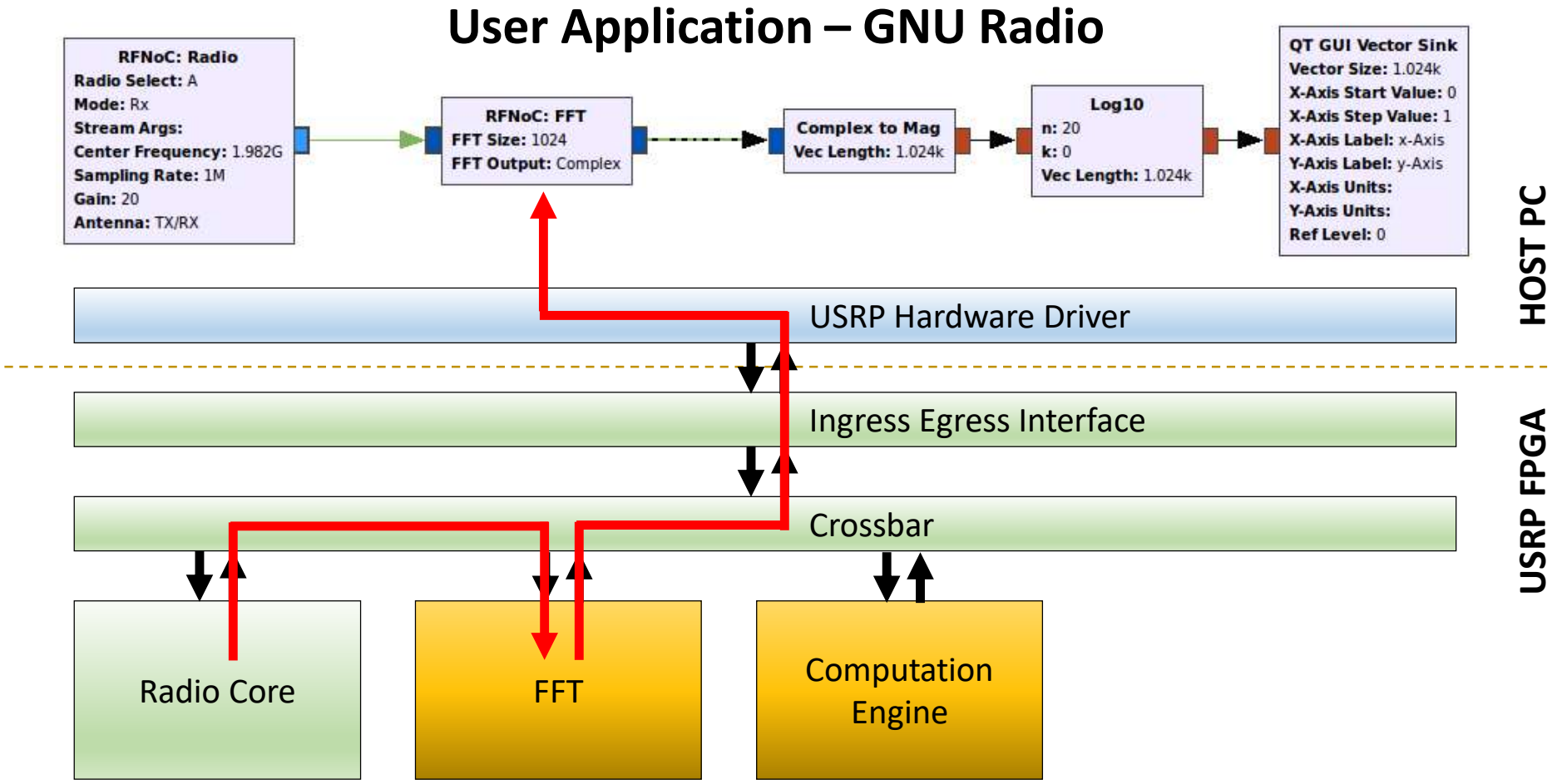
- Software API to:
  - Configure USRP hardware & RFNoC FPGA infrastructure
  - Provide user sample data (r/w buffers) & control (r/w regs) interfaces

Radio Core

FFT

Computation Engine

RFNoC Architecture



Summary

Ettus Research™ A National Instruments Company

- Simple architecture for scalable, distributed FPGA processing
- Tightly integrated with GNU Radio
- Implemented several interesting CEs
  - OFDM, FFT, FIR, Signal Generator, Fosphor….
- Portable between all third generation USRPs
  - X3x0, E3xx
- Completely open source
- kb.ettus.com/RFNoC_Getting_Started_Guides
- After the break: RFNoC FPGA Development

Break