USC Viterbi
School of Engineering
Center for Cyber-Physical Systems and the Internet of Things

CCI

SOFTWARE DEFINED RADIO

USR SDR WORKSHOP, SEPTEMBER 2017

PROF. MARCELO SEGURA

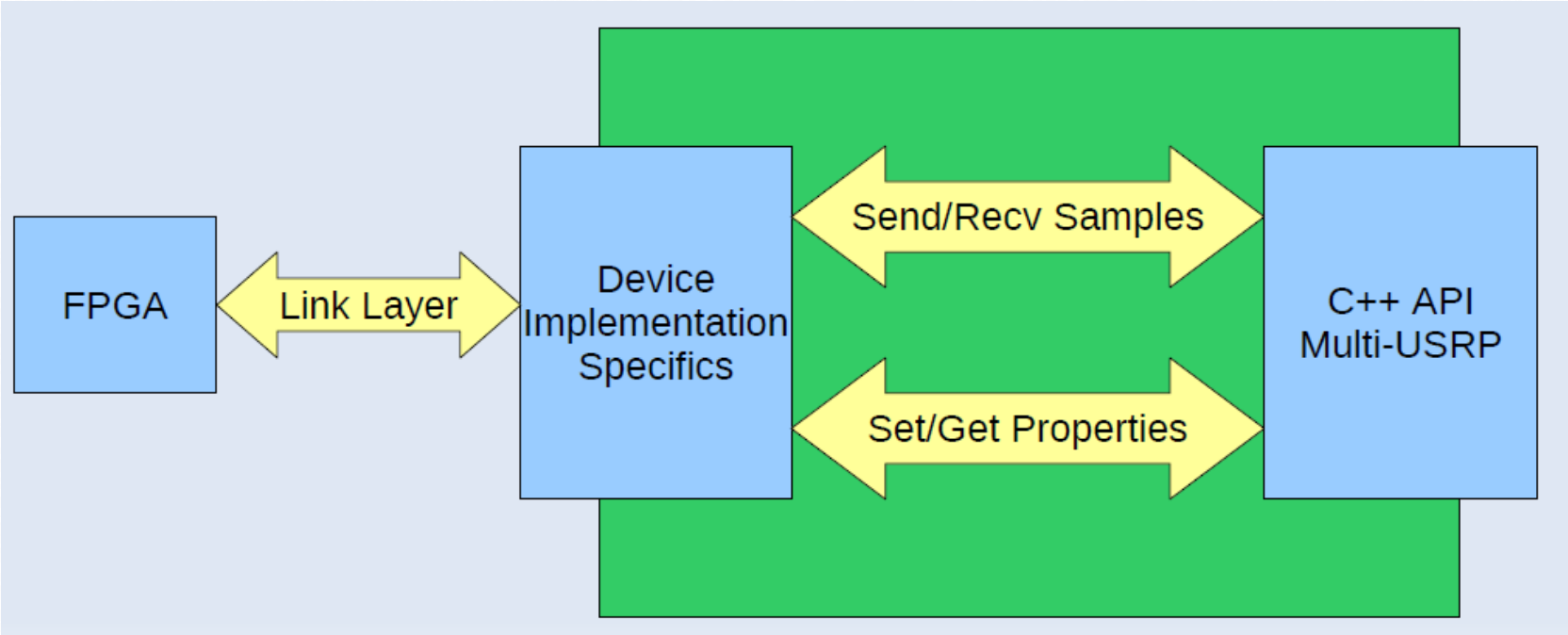
SESSION 1: SOFTWARE TOOLS

BRIEF HISTORY

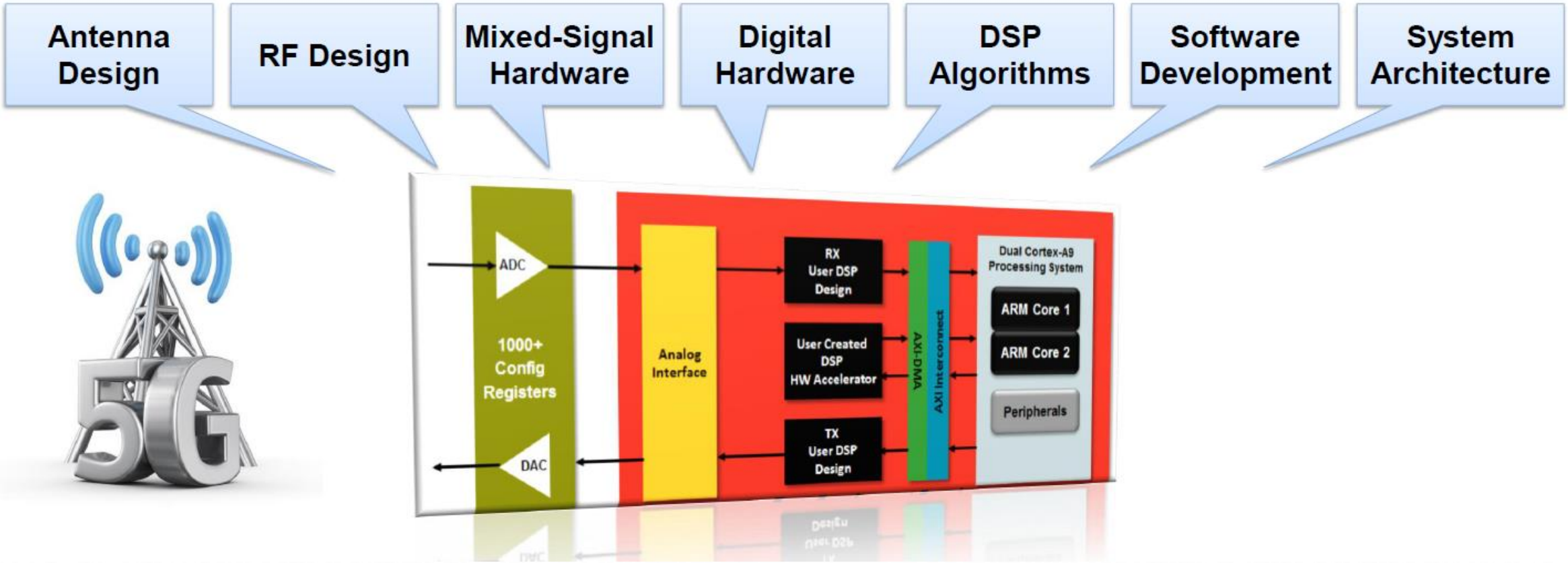
- USRP
  - Libusrp
  - Libusrp-gnuradio
  - Python dboard code
  - C++ dboard code
  - Usrp_* examples and utils
- USRP2
  - Libusrp2 (linux only)
  - libusrp2-gnuradio
  - C dboard code in FW
  - Usrp2_* examples and utils
- NOT SCALE

DEFINITION

- UHD is an C++ API, that allow to interface a host with any USRP boards.
- Instantiate device object on Matlab or GNURadio.
  - SET/GET FUNCTION
  - SEND/ RECEIVE SAMPLES (HOST: Ring buffer, SDR: BRAM FIFO)



NEXT GEN WIRELESS DEVELOPMENT



REQUIRES AT LEAST 7 DIFFERENT SKILLS!!!

TOOLBOX

- We primarily use **Comm & DSP** toolbox

Algorithms, Waveforms, Measurements

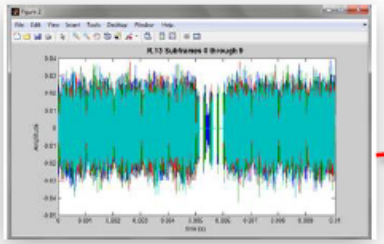
- Communications System Toolbox
- Phased Array System Toolbox
- LTE System Toolbox
- WLAN System Toolbox

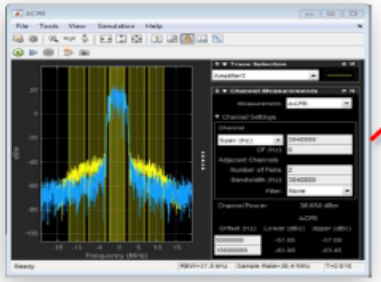
RF Front End

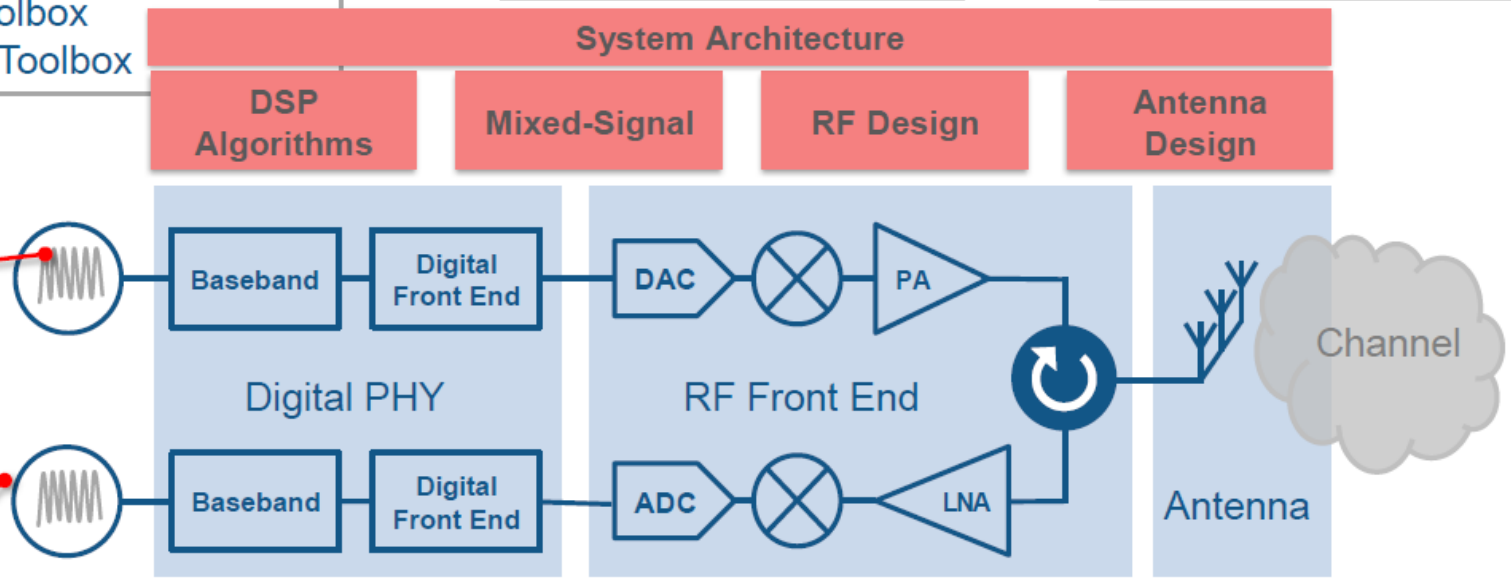
- RF Toolbox
- RF Blockset

Antennas, Antenna Arrays

- Antenna Toolbox
- Phased Array System Toolbox





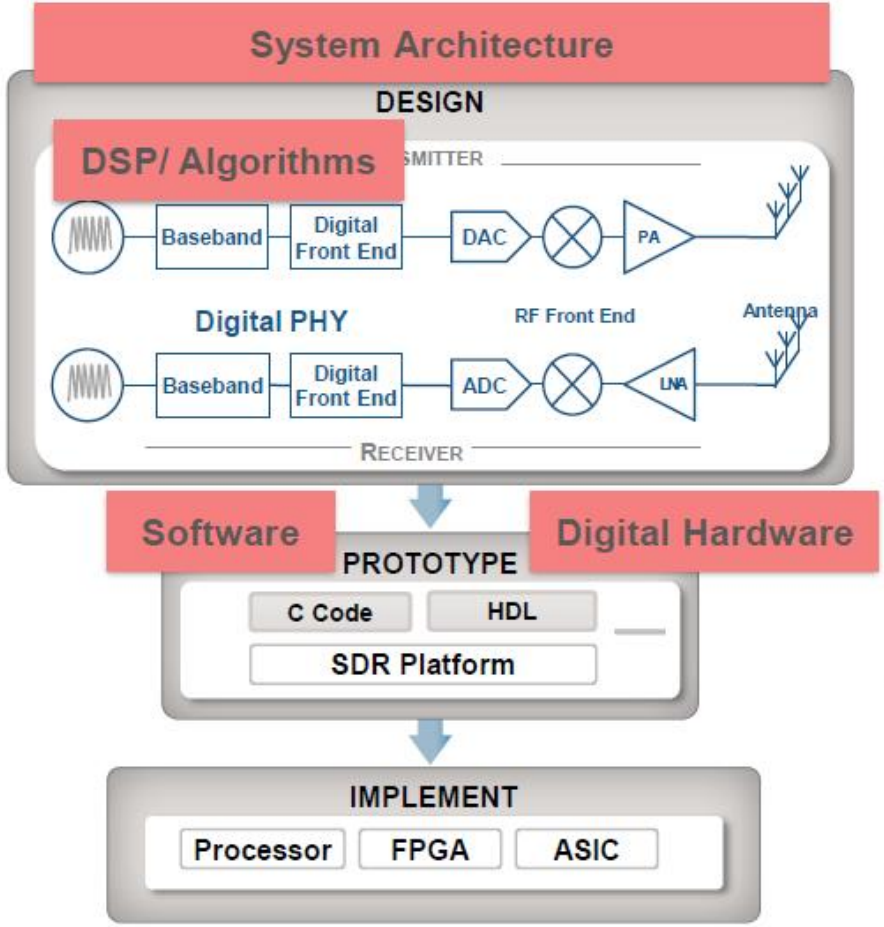


- Simulink
- DSP System Toolbox
- Control System Toolbox

Mixed-signal

- Communications System Toolbox
- Phased Array System Toolbox
- LTE System Toolbox
- WLAN System Toolbox

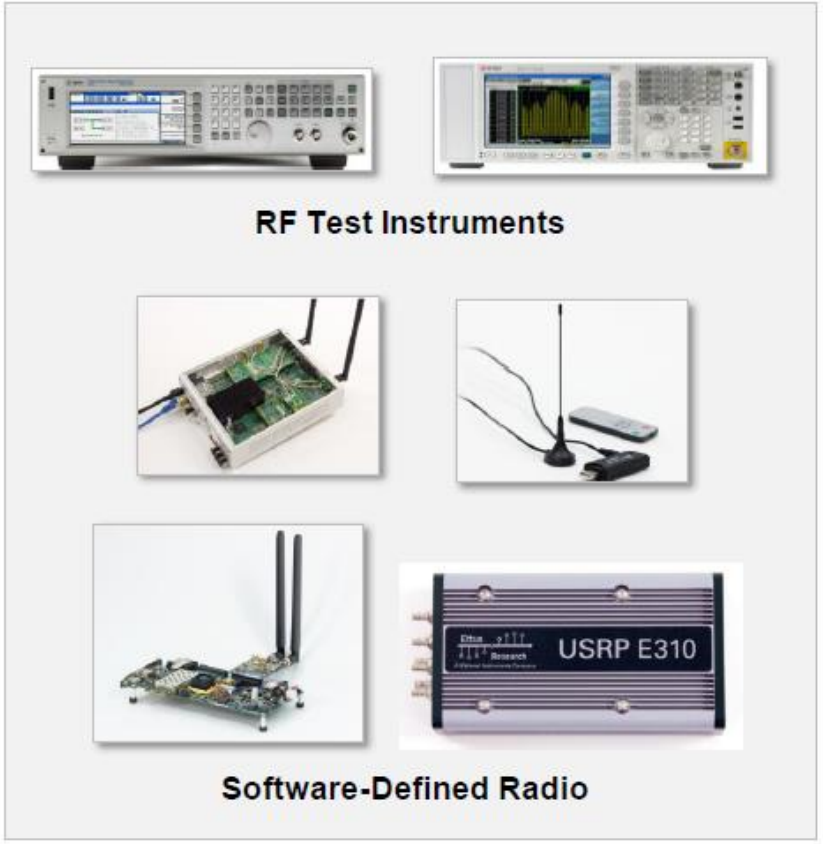
DESIGN WORKFLOW



Instrument Control Toolbox

SDR Support Packages
Communications System Toolbox

Fixed-Point Designer
HDL Coder
Embedded Coder



HDL and C code generation

Multi-vendor hardware support

PROTOTYPING WORKFLOW

Algorithm design
Simulation Model

↓

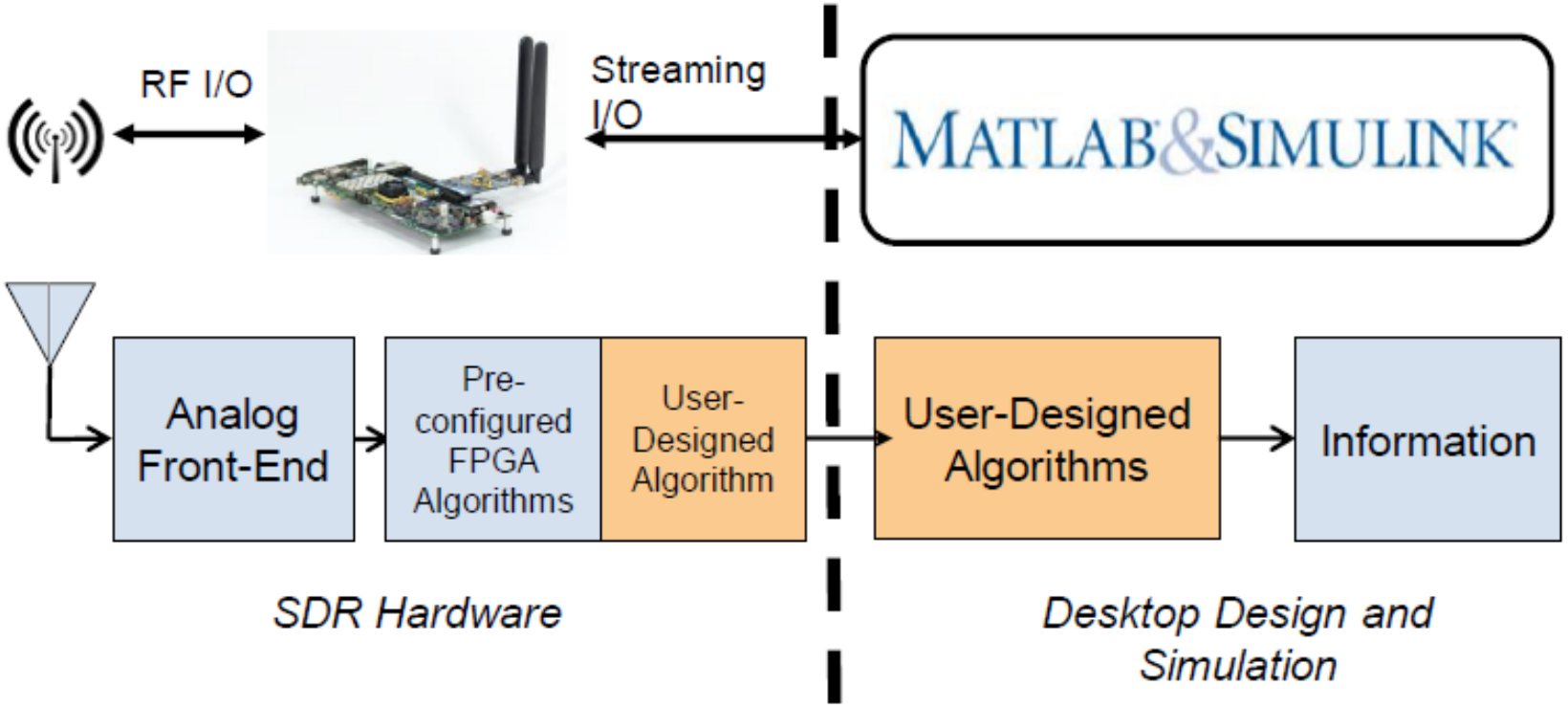
SDR testing with live signal I/O

↓

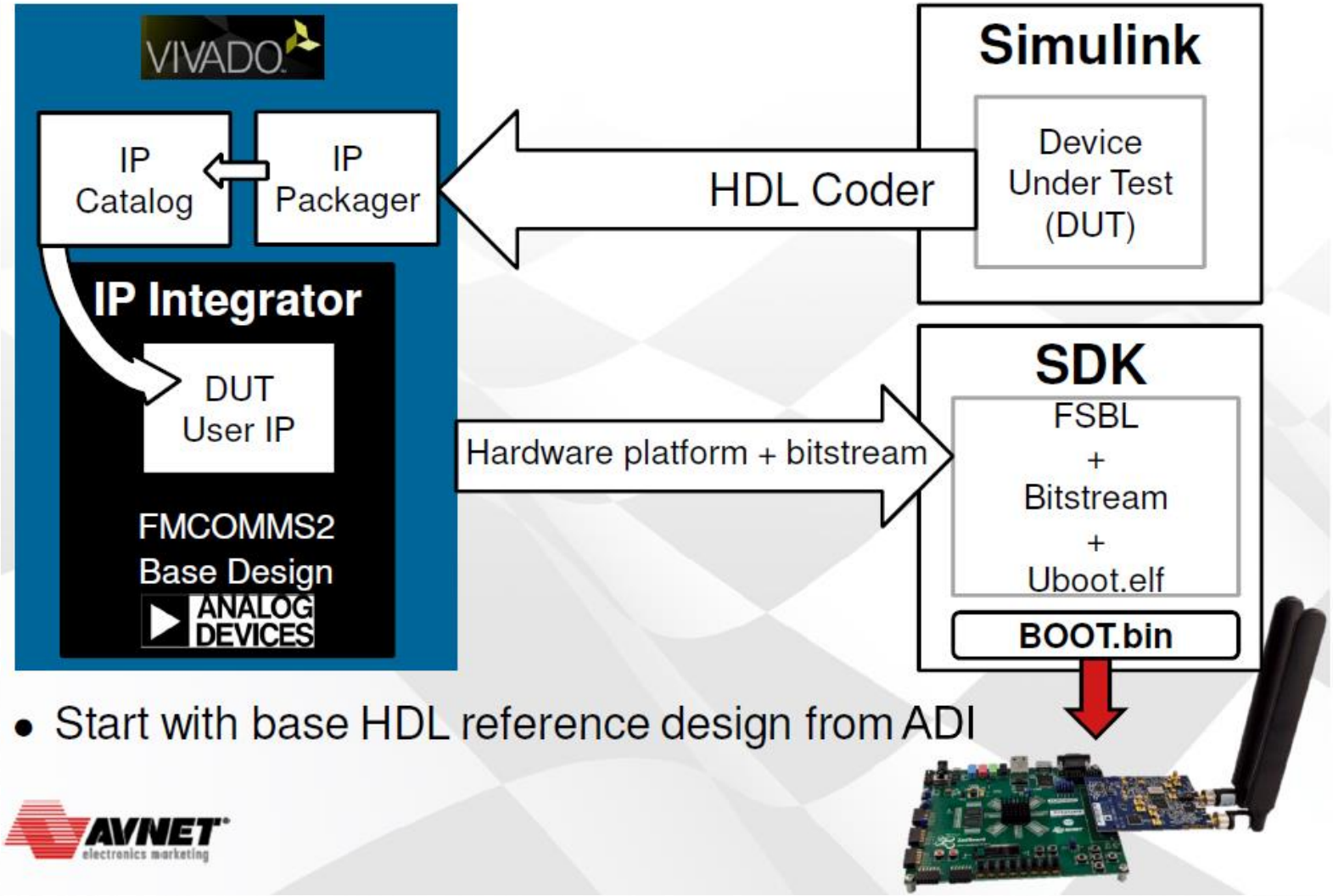
Generate HDL Code
Implementation Model

↓

Verify on SDR hardware



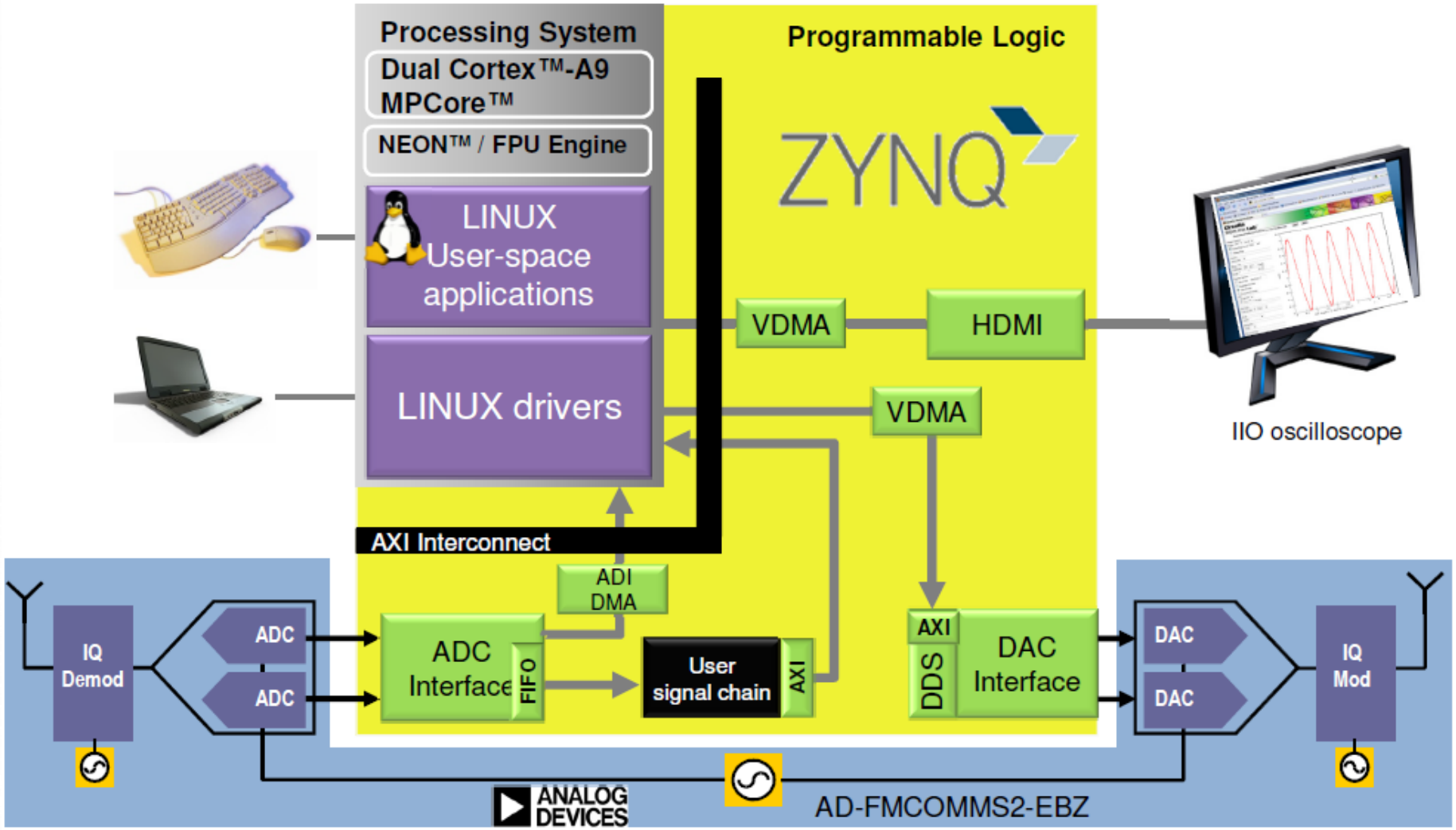
PROTOTYPING WORKFLOW



- Start with base HDL reference design from ADI

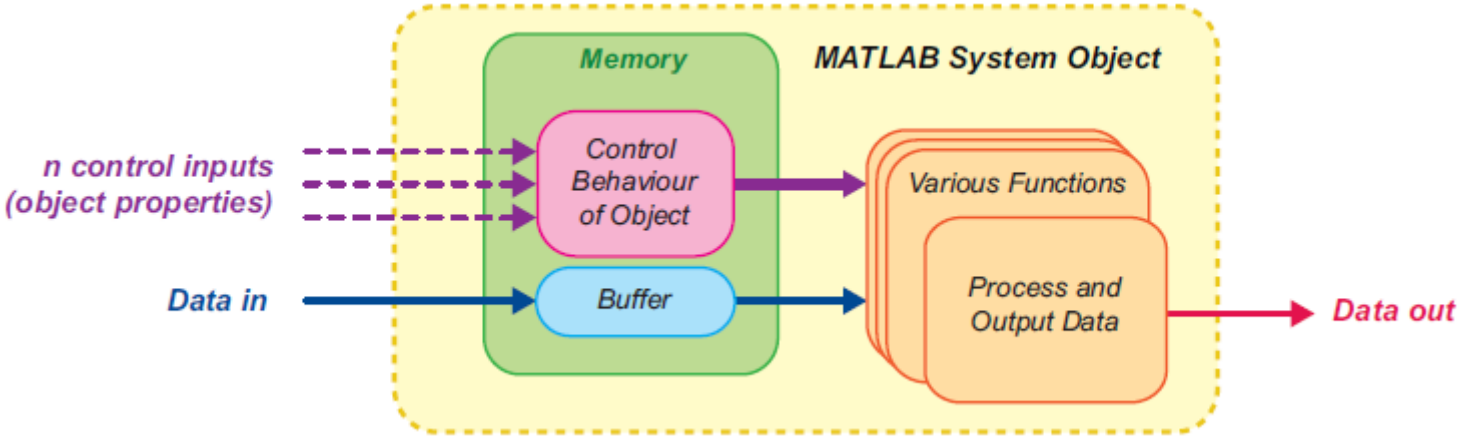
PROTOTYPING WORKFLOW

- **Create your own IP core**
- Performance
- Interface speed
- Hardware accelerators



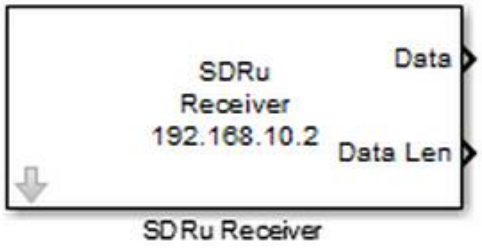
DEFINITION

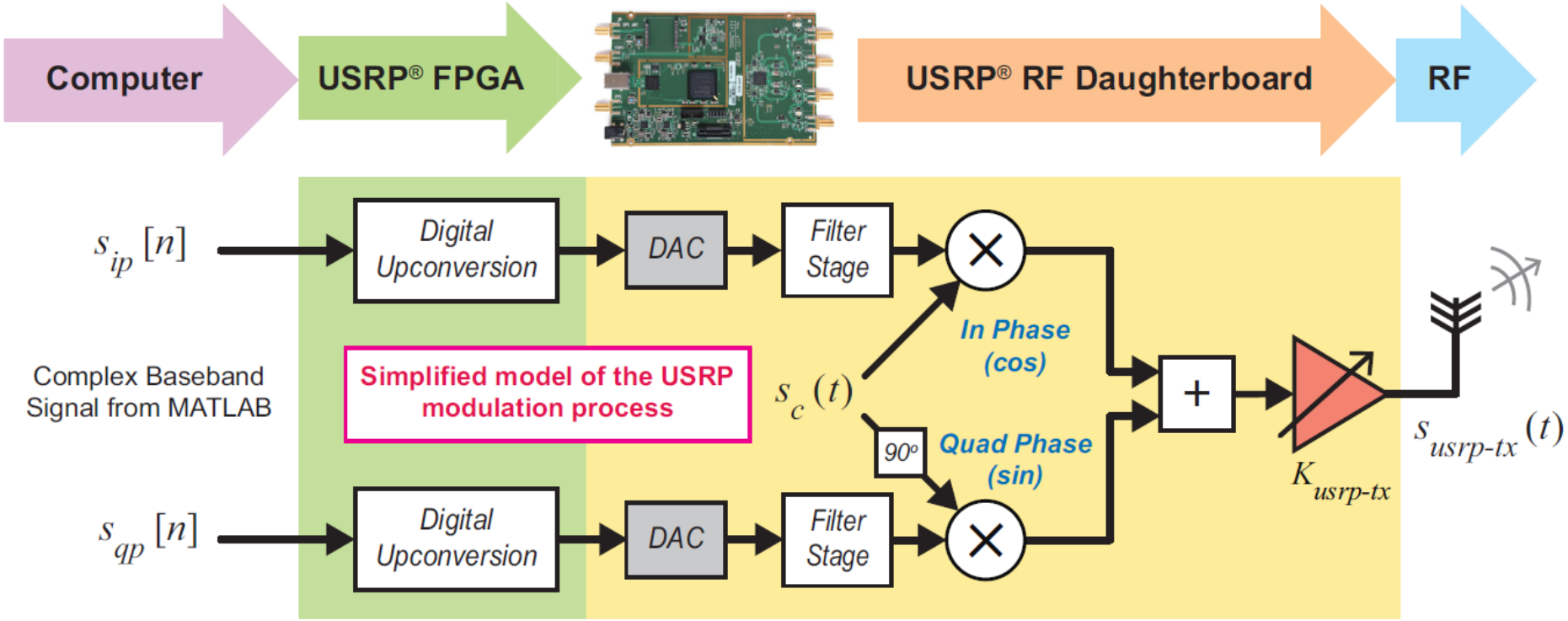
- Matlab System objects are the base building block to communicate with SDR.
- System objects are designed specifically for implementing and simulating dynamic systems with inputs that change over time.
- Matlab objects have a predefined structure that allow to control and send/receive date from SDR.



SUPPORT PACKAGE

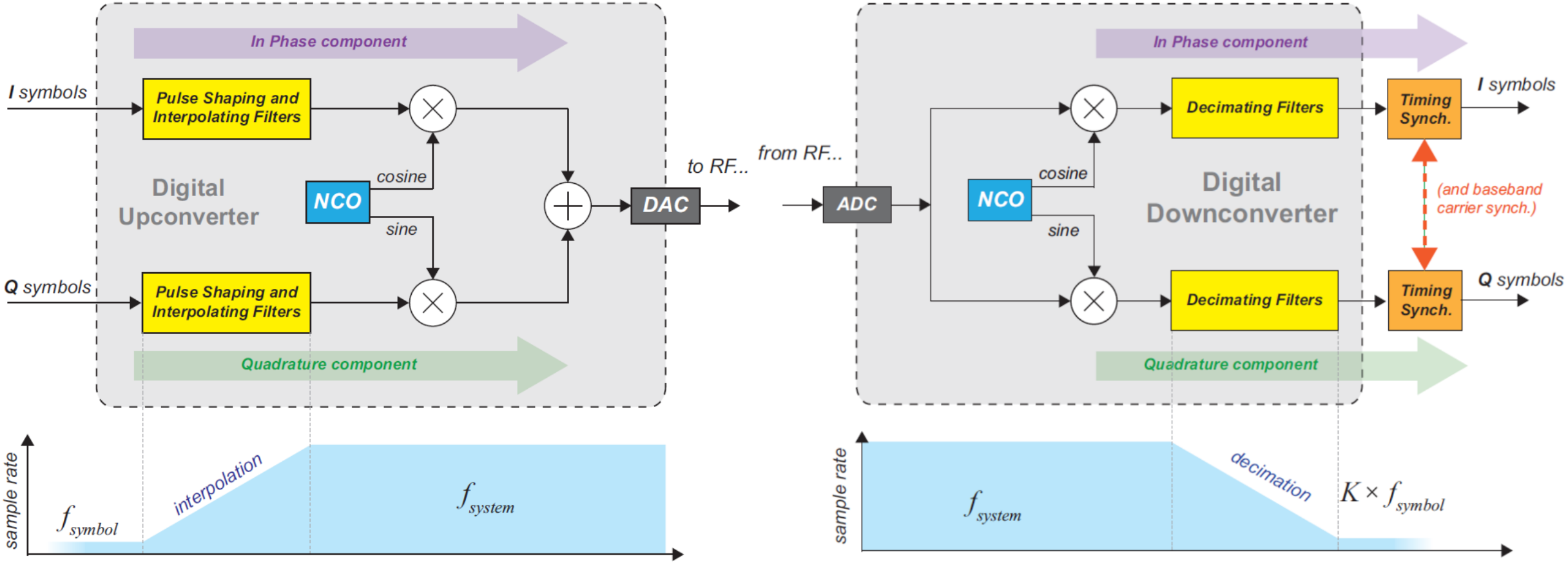
- UHD ON A SYSTEM BLOCK





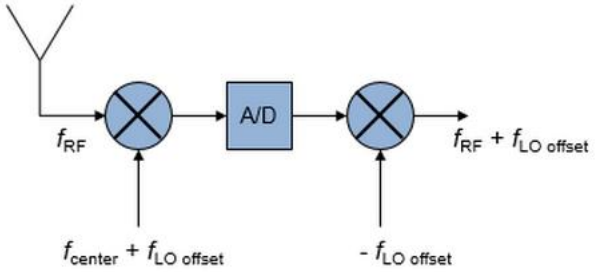
DIGITAL UP/DOWN CONVERSION RATES

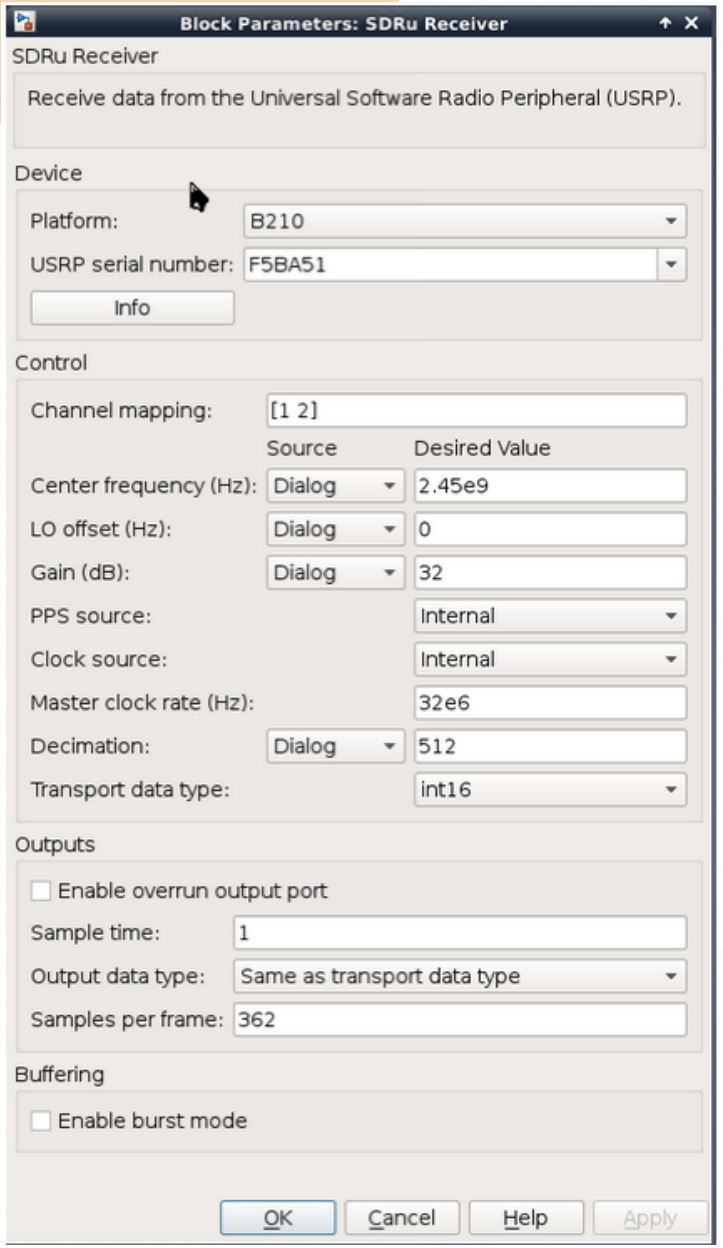
- The sampling rates should be matched with the master clock rate.



SUPPORT PACKAGE

- SDRu RX block:
- Serial Number of B200 mini
- Channel Mapping: 1 SISO, [1 2] MIMO
- Center Freq: LO frequency
- LO Offset: offset freq to avoid LO leakage or self interference.
- Gain: overall gain including both analog and digital components.
- Transport data: 16 bit default
- Samples per frame: to optimize Ethernet packet.
- Burst mode: to test systems that cannot run on real time, avoiding overrun or Underrun.





TX/RX BLOCK CONFIG

```
rx = comm.SDRuReceiver('Platform','B210','SerialNum','ECR04ZDBT')
```

```
rx =

  System: comm.SDRuReceiver

  Properties:
                      Platform: 'B210'
                     SerialNum: 'ECR04ZDBT'
               ChannelMapping: 1
        CenterFrequencySource: 'Property'
              CenterFrequency: 2450000000
        ActualCenterFrequency: 2450000000
   LocalOscillatorOffsetSource: 'Property'
        LocalOscillatorOffset: 0
   ActualLocalOscillatorOffset: 0
                  GainSource: 'Property'
                        Gain: 8
                  ActualGain: 8
                   ClockType: 'Internal'
              MasterClockRate: 32000000
        ActualMasterClockRate: 32000000
       DecimationFactorSource: 'Property'
             DecimationFactor: 512
       ActualDecimationFactor: 512
            TransportDataType: 'int16'
            OverrunOutputPort: false
                   SampleRate: 1
              OutputDataType: 'Same as transport data type'
                 FrameLength: 362
             EnableBurstMode: false
```

```
tx = comm.SDRuTransmitter('Platform','B210','SerialNum','ECR04ZDBT')
```

```
tx =

  comm.SDRuTransmitter with properties:

                      Platform: 'B210'
                     SerialNum: 'ECR04ZDBT'
               ChannelMapping: 1
        CenterFrequencySource: 'Property'
              CenterFrequency: 2.4500e+09
        ActualCenterFrequency: 0
   LocalOscillatorOffsetSource: 'Property'
        LocalOscillatorOffset: 0
   ActualLocalOscillatorOffset: 0
                  GainSource: 'Property'
                        Gain: 8
                  ActualGain: 0
                 ClockSource: 'Internal'
              MasterClockRate: 32000000
        ActualMasterClockRate: 0
     InterpolationFactorSource: 'Property'
           InterpolationFactor: 512
     ActualInterpolationFactor: 0
            TransportDataType: 'int16'
           UnderrunOutputPort: false
             EnableBurstMode: false
```

GETTING START

- Lab 1 : **INSTALLING B200 on MatLab**

Setup and Configuration

sdruload	Load FPGA and firmware images for USRP® radio

Hardware Discovery

findsdru	Find and report status for all USRP® devices connected to host computer
getSDRuDriverVersion	Report UHD™ version number for the support package
probesdru	Provides detailed USRP® radio information

Radio Management

setsdruip	Set USRP® radio IP address
comm.SDRuReceiver	Receive data from USRP® device
comm.SDRuTransmitter	Send data to USRP® device