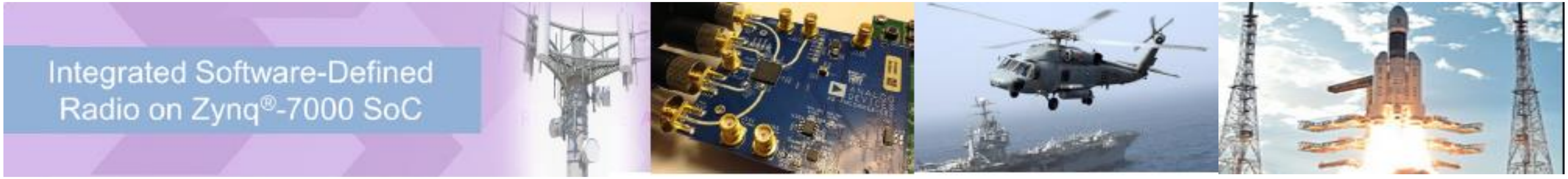
VSSC
TVPM

Design and implementation of a digital wireless system(compliant to IEEE802.15.4) with channel emulation capability using Matlab Simulink and Zync SoC-SDR Platform

Bibin Varghese
Dr S Sreelal
VSSC Trivandrum

MATLAB EXPO 2019 , Hyderabad



Outline

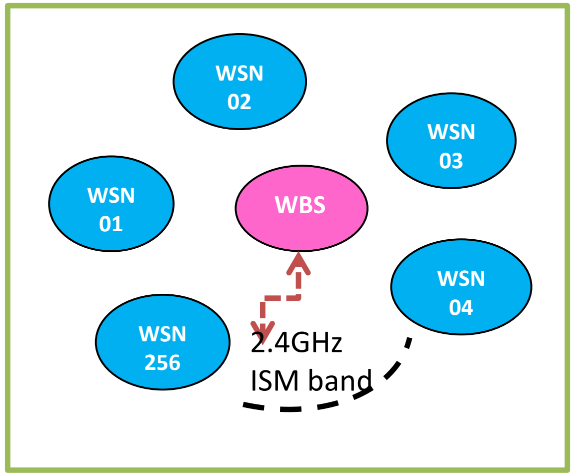
- ❑Introduction
- ❑Existing Wireless Instrumentation System(WIS) and the motivation to emulate the same using Model based SDR techniques
- ❑Model based design flow with Matlab simulink
- ❑Addressing SoC design challenges with Matlab
- ❑Simulink Modeling of IEEE802.15.4 Transmitter & Receiver
- ❑Experimental Test bed and Results
- ❑Benefits of model based SDR development & Future Scope

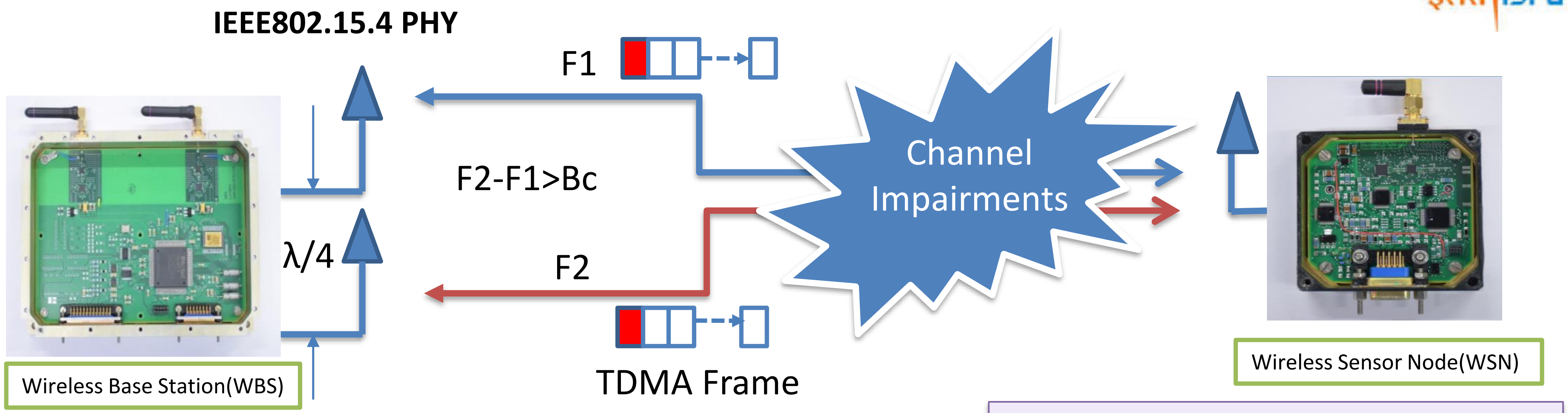
Introduction

- The testing, verification and evaluation of indoor wireless systems is an important but challenging endeavor.
- The most realistic method to test the wireless system is a field deployment. Unfortunately, this is not only expensive but also time consuming.
- Real-time hardware in the loop RF channel emulation fills the gap left between simulation and field testing.
- In this work, we present the design and implementation of a programmable digital wireless system(Tx-Rx pair) with channel emulation capability, which connects directly to our DUT radio , and mimics the wireless channel as well as other impairments between them, in real-time using Matlab simulink & Zync-SDR platform.
- We describe a fast and accurate way of development and deployment of the entire system using model based design running on SDR.

Motivation -Existing Wireless Instrumentation System(WIS)

WIS Architecture



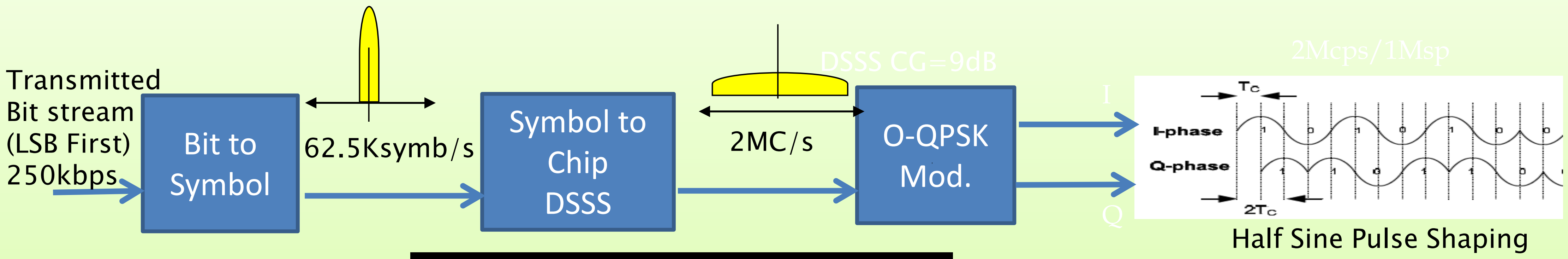


- PHY of WIS is compliant to IEEE802.15.4. It cannot be used as such for robust communications in aerospace applications where the multi-path effects are more.
- COTS radio transceivers were used for the realization of WIS which doesn't give access to the PHY layer for any channel error mitigation.
- System robustness in presence of multi-path fading is enhanced by providing three types of diversity techniques(Frequency, Spatial & Temporal) in the MAC layer.
- Assessing the performance of this entire system in presence of different channel impairments is a challenging as well as expensive task with traditional test systems.

Transmitter as well as the receiver built on SDR platform not only simplify the testing but also bring efficient solutions like channel equalization, melioration of PHY layer etc to address the root cause of the channel impairment.

The advances in platforms and tools will allow developers to quickly simulate and prototype such wireless applications while establishing and maintaining a deployable path to production too.

IEEE802.15.4 PHY

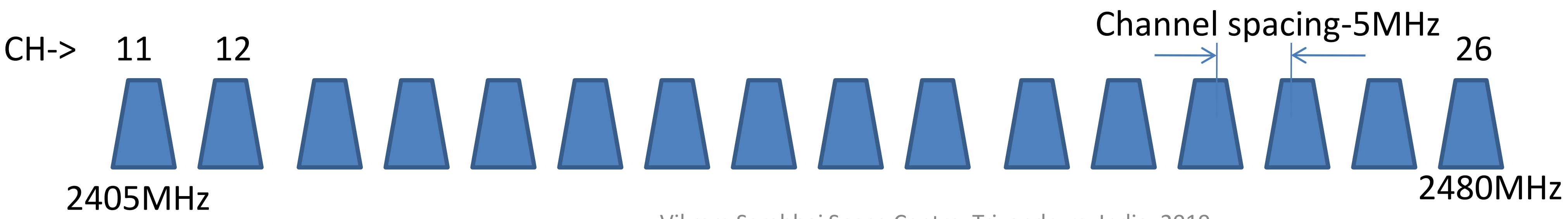


Symb	Chip Seq
0	1 1 0 1 1 0 0 1 1 1 0 0 0 0 1 1 0 1 0 1 0 0 1 0 0 0 1 0 1 1 1 0
1	1 1 1 0 1 1 0 1 1 0 0 1 1 1 0 0 0 0 1 1 0 1 0 1 0 0 1 0 0 0 1 0

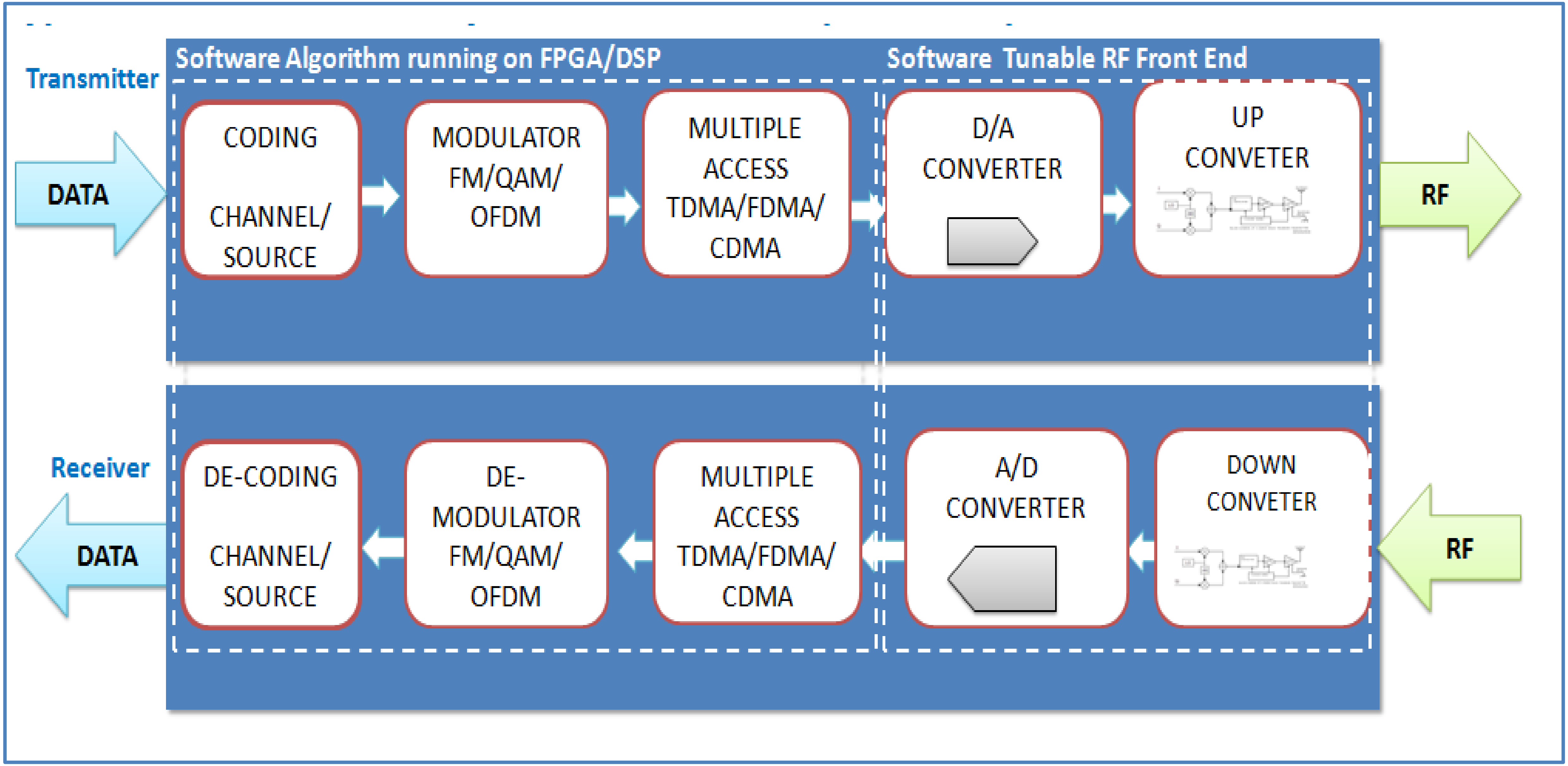
Equivalent to MSK Smaller side lobes, Constant Envelope

PDU Format

SyncHdr		PHY Hdr	PHY PayLoad
Preamble	**Start Of Frame**	**Frame Len**	**PHY Service Data Unit**
4 Bytes	1 Byte	1 Byte	0-127 Bytes



Typical Communication System Architecture



Using I/Q Modulator to Generate any RF Signal

A general representation of any RF signal

$$A\cos(2\pi f_c t + \phi)$$

Amplitude | Frequency | Phase

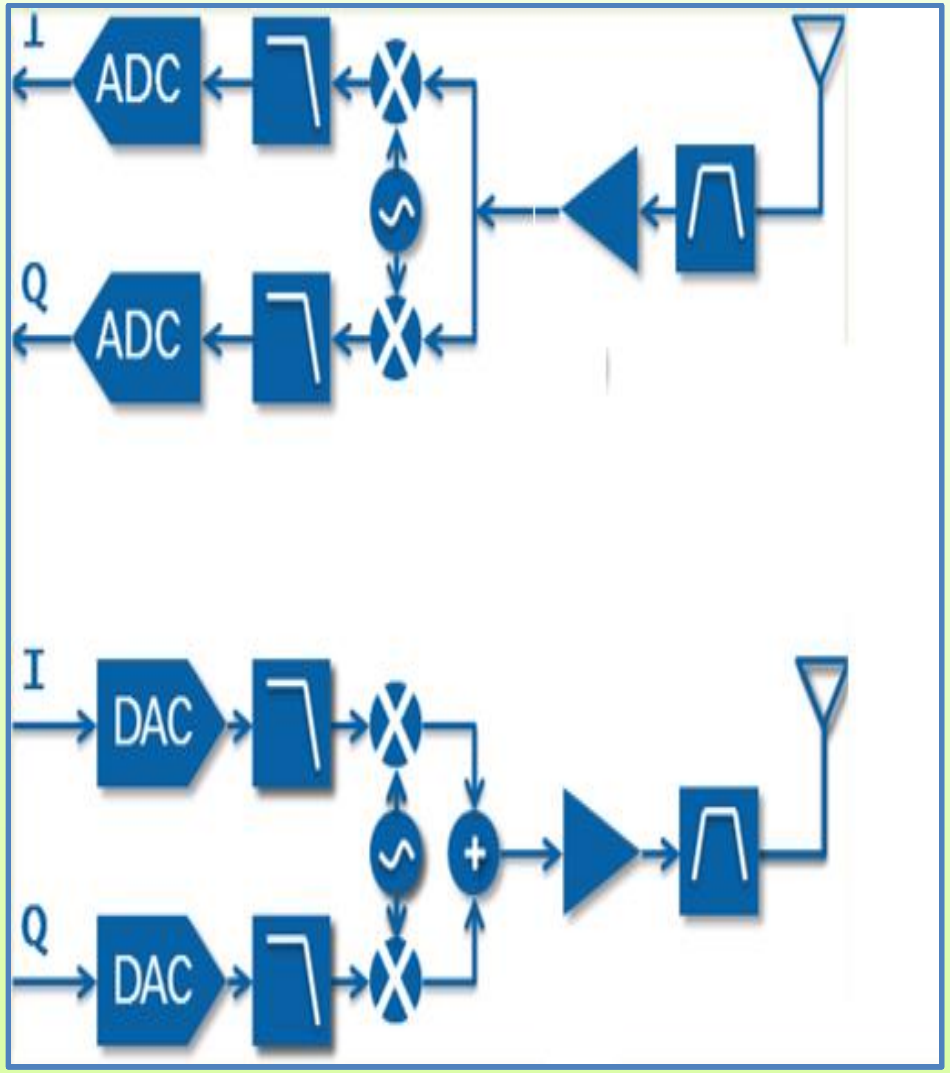
$$\cos(\alpha + \beta) = \cos(\alpha)\cos(\beta) - \sin(\alpha)\sin(\beta)$$

$$A\cos(2\pi f_c t + \phi) = A\cos(2\pi f_c t)\cos(\phi) - A\sin(2\pi f_c t)\sin(\phi)$$

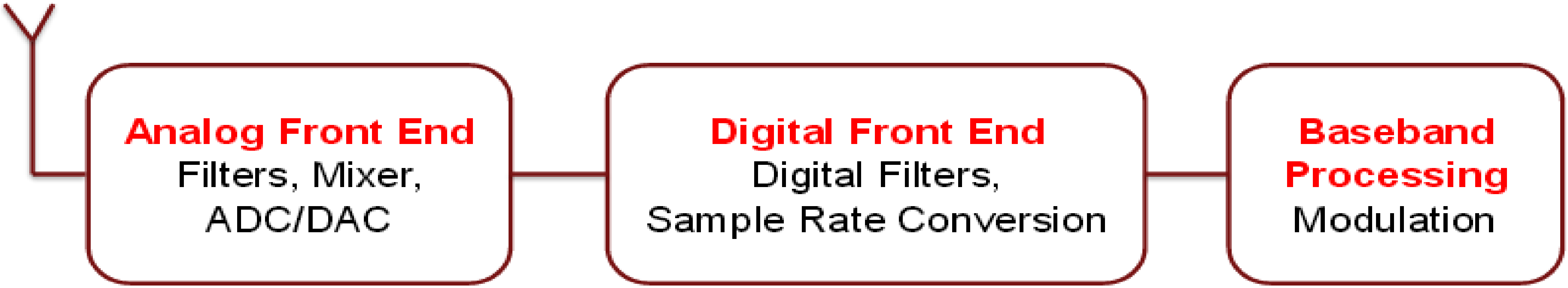
I | Q

$$A\cos(2\pi f_c t + \phi) \equiv I\cos(2\pi f_c t) + Q\sin(2\pi f_c t)$$

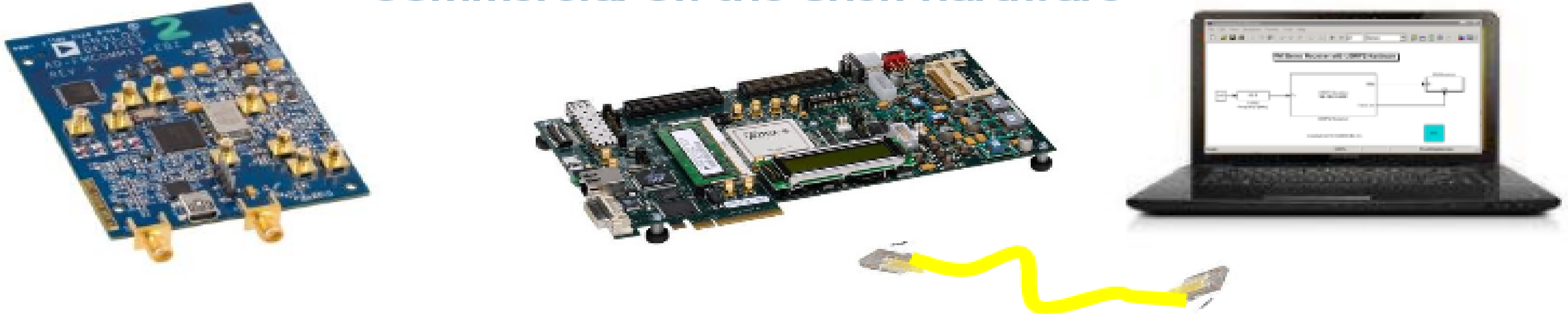
Basic SDR architecture



Typical SDR System



Commercial-off-the-shelf hardware



FMC

GigE

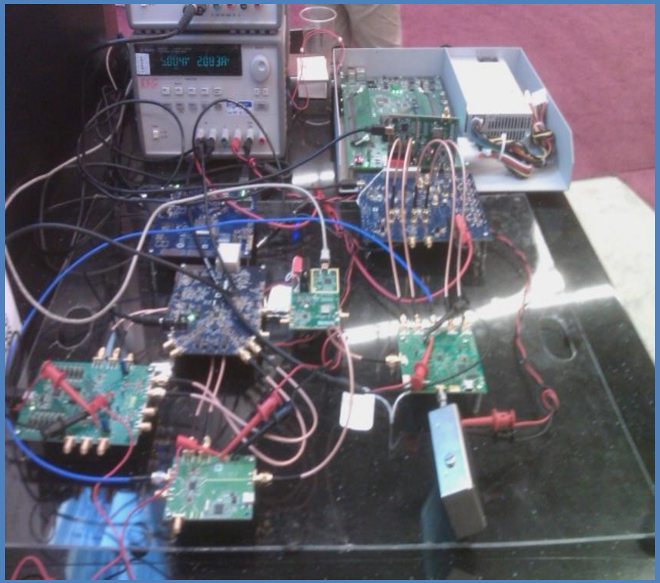
Tunable RF Card

FPGA/Zynq Board

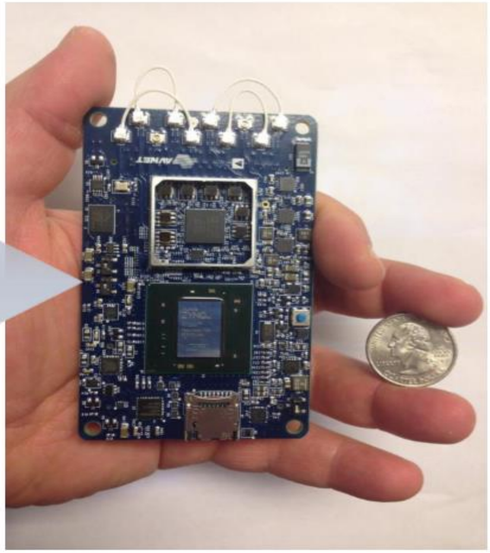
Host computer

SDR Changes the way how we think on RF Systems!

Traditional RF Prototyping



SDR Prototyping



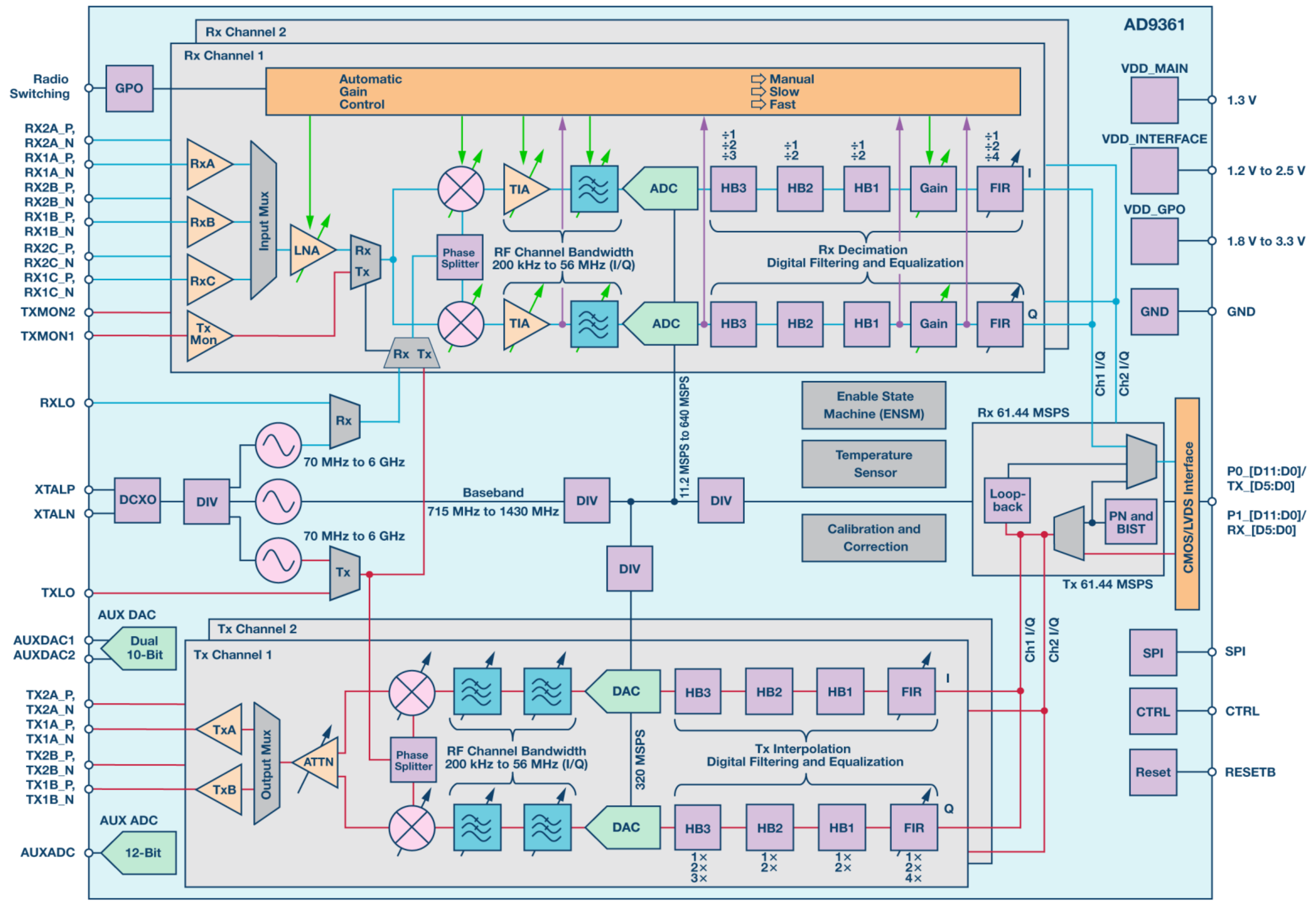
Advantages

- ✓ Unprecedented application capabilities.
- ✓ Extremely modular solution.
- ✓ Allows adoption for different missions keeping required hardware changes minimal.
- ✓ SDR + FPGA/DSP→ allow implementing all changes in software, keeping hardware heritage intact.

Applications

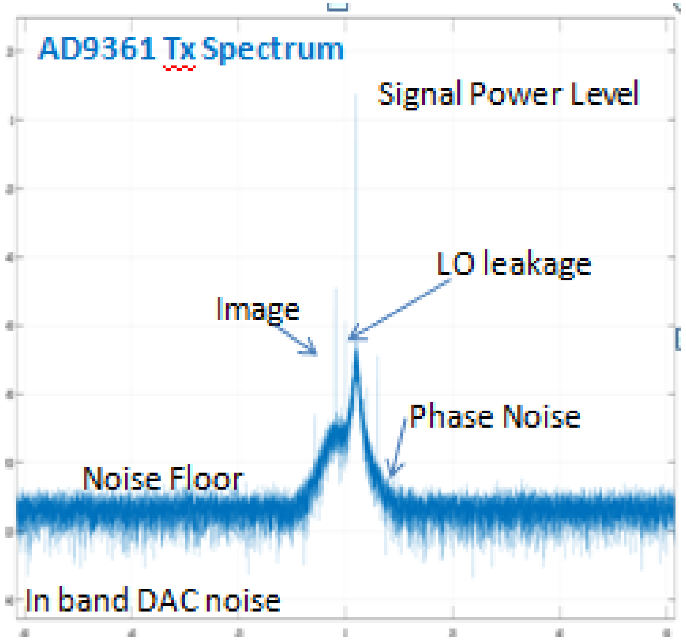
- ❖ Re-configurable radio for deep space, inter-planetary missions and ground test applications.
- ❖ Integrated telemetry and telecommand systems for Launch vehicle /satellite missions.
- ❖ Inter and Intra stage Wireless Telemetry for Launch vehicles

AD9361-Agile RF SDR Transceiver



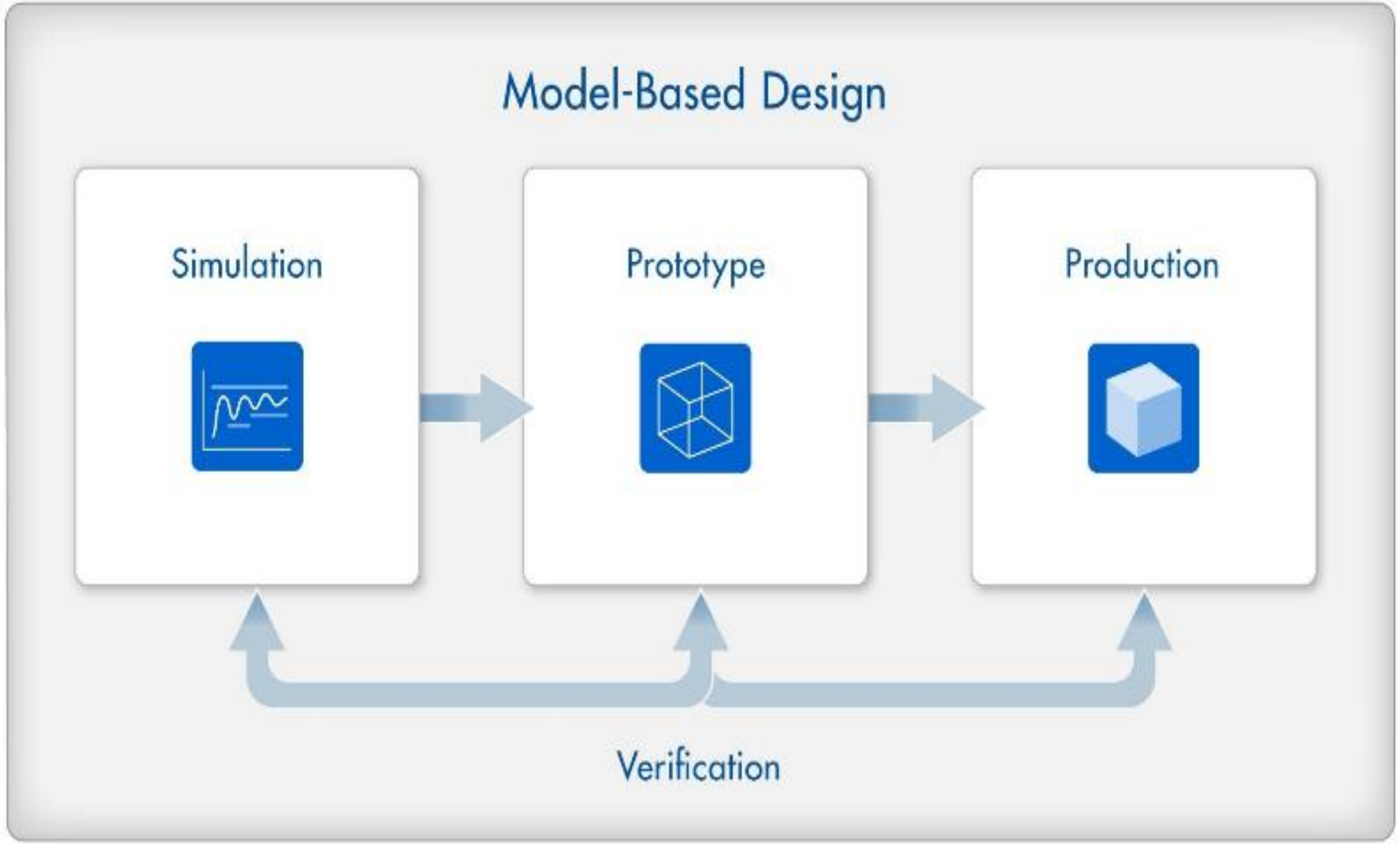
FEATURES

- RF 2 × 2 transceiver with integrated 12-bit DACs and ADCs
- TX band: 47 MHz to 6.0 GHz
- RX band: 70 MHz to 6.0 GHz
- Supports TDD and FDD operation
- Tunable channel bandwidth: <200khz to 56MHz
- User Programmable Filters in Transmitter and Receiver



Model Based Design Makes SDR development simple and fast

A single shared development environment



- Virtual representation of a real-world system
- A way to manage a complex system
- Common design platform for entire design team
- Reduces hardware testing time by shifting design from lab to desktop

Enables:

different levels of simulation

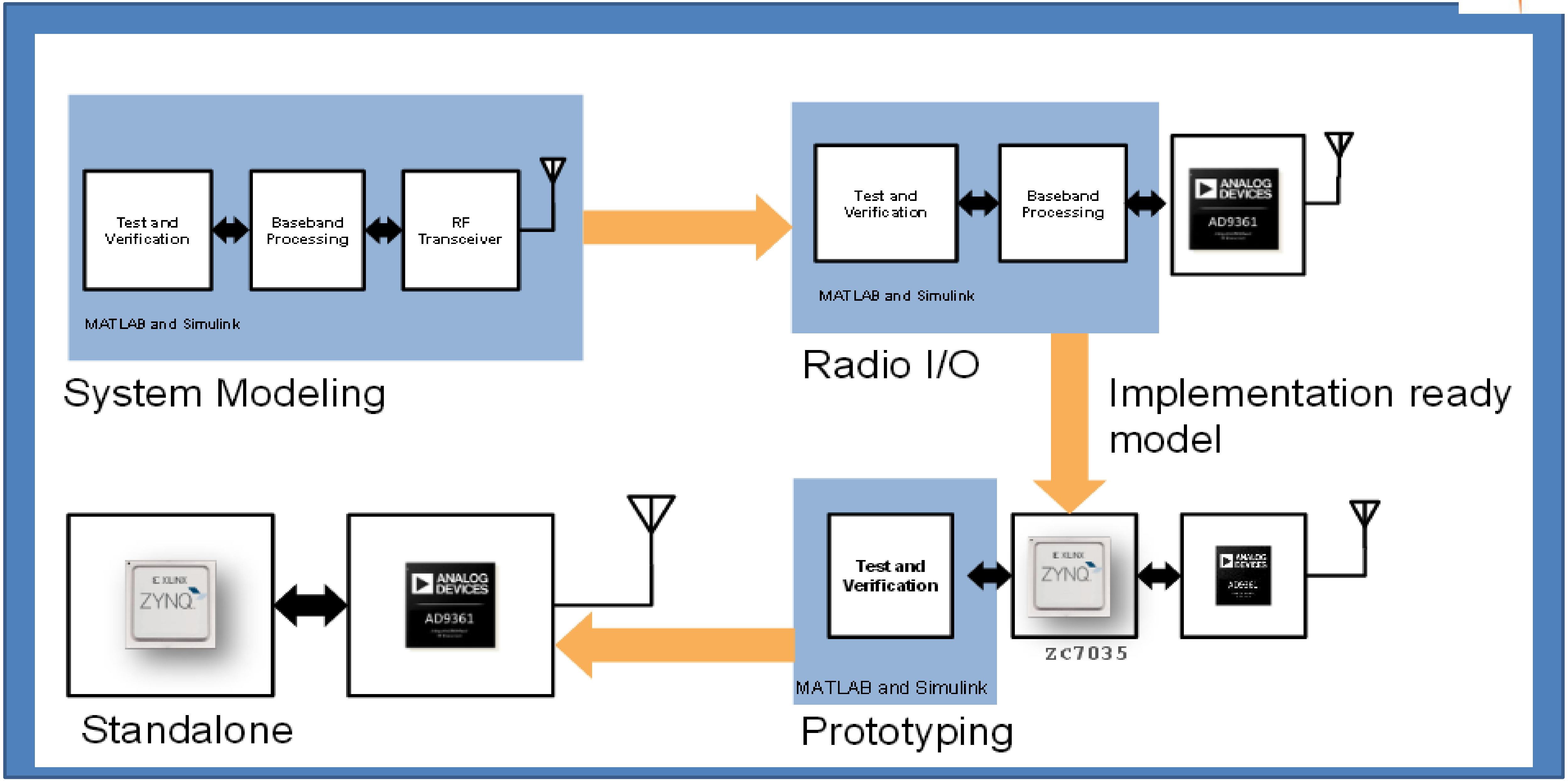
System architects can build prototypes with popular FPGA and SDR kits and hardware engineers can reuse those models for production deployment.

Verify operation before committing to hardware

Validate performance on chip

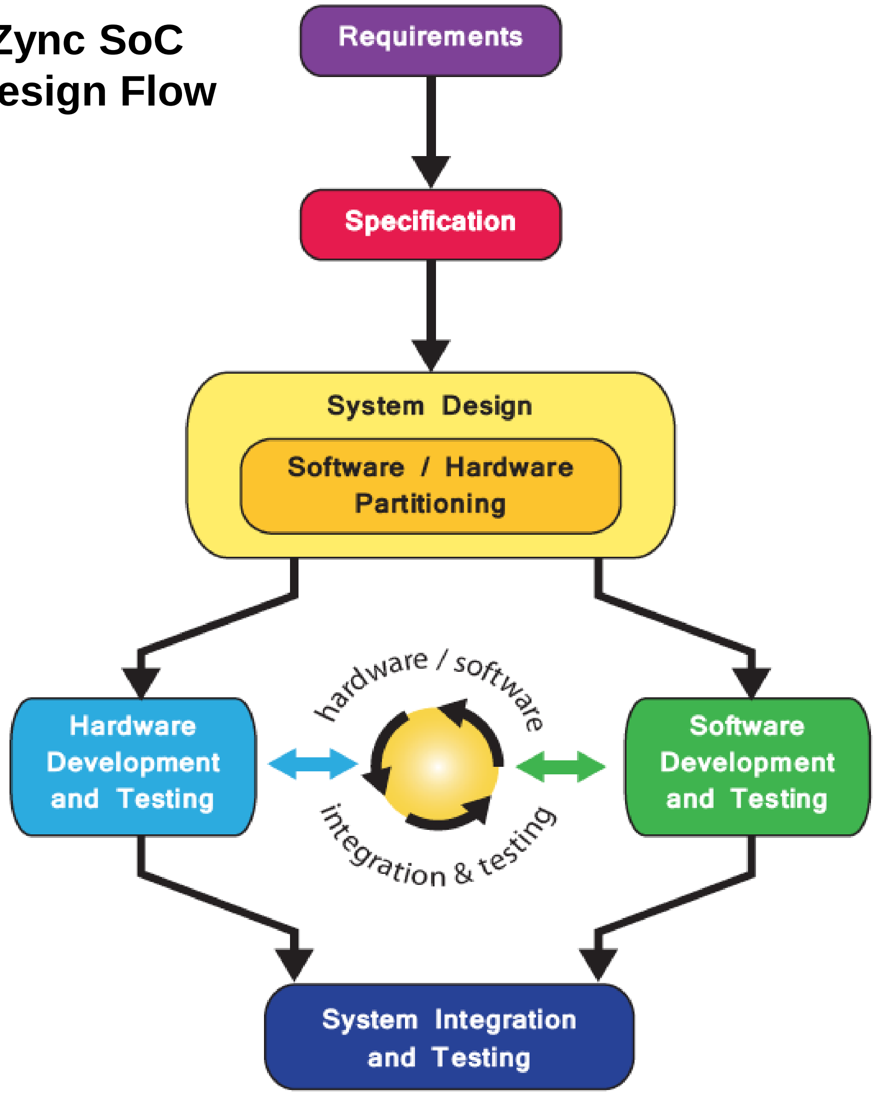
Deploy design on target system

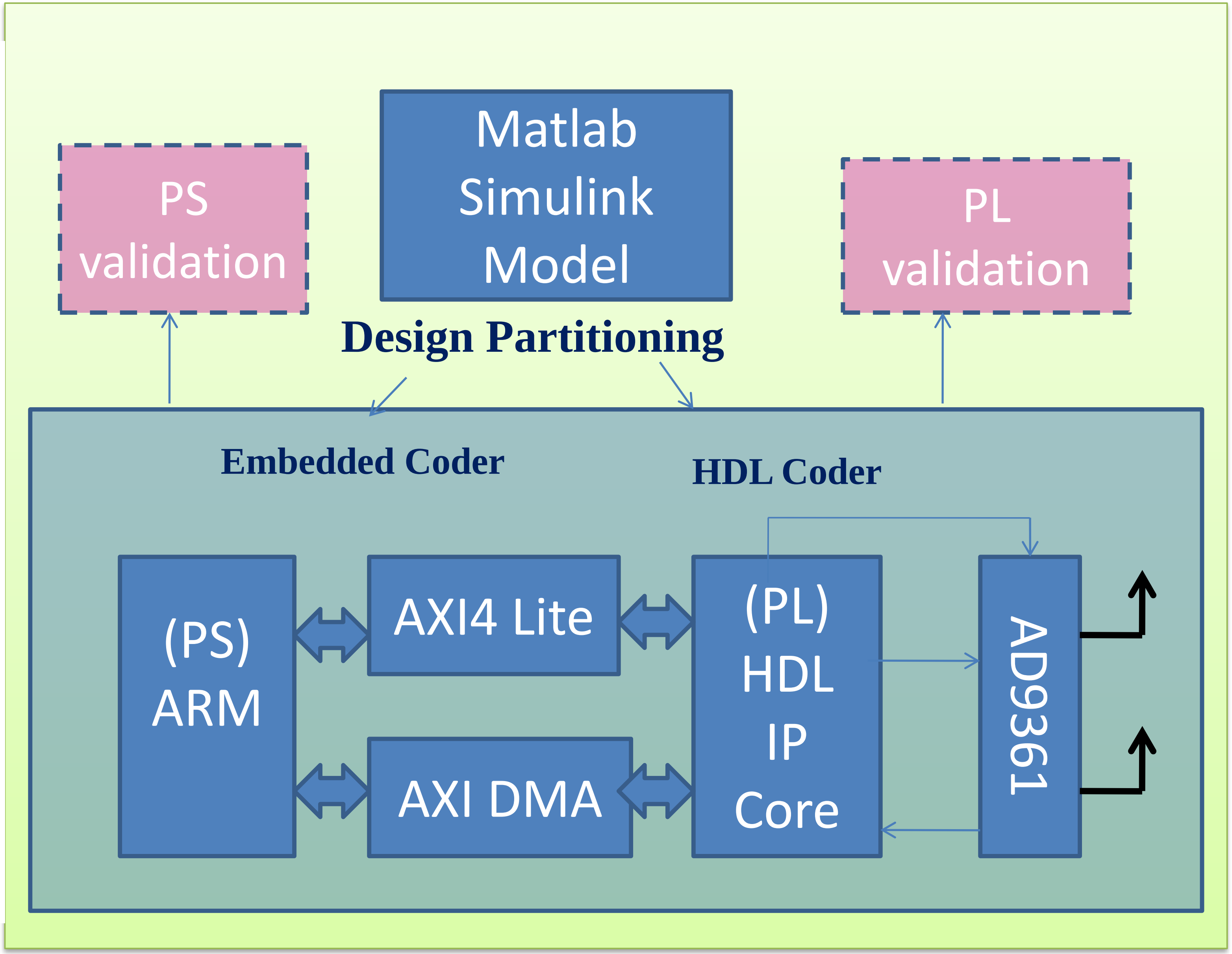
SDR Model Based Design Flow with Matlab Simulink



Addressing Challenges in SoC Design Using Matlab

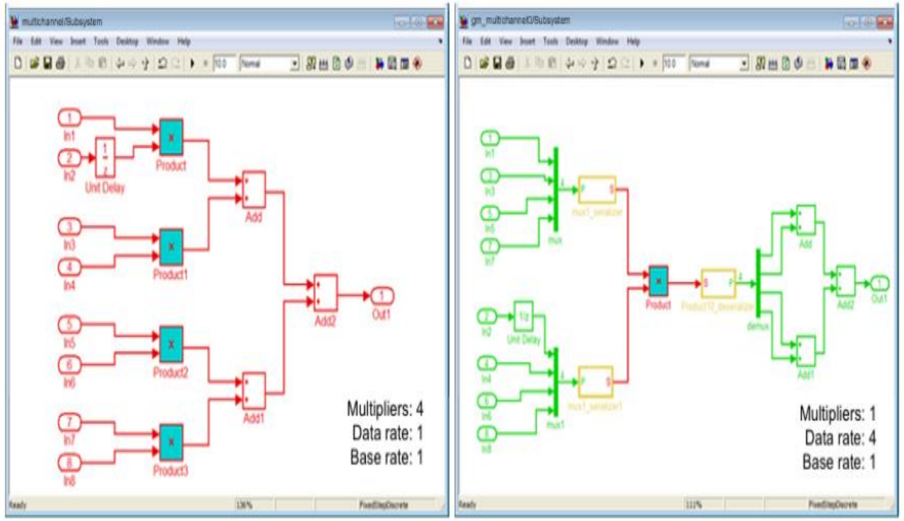
Zync SoC
Design Flow



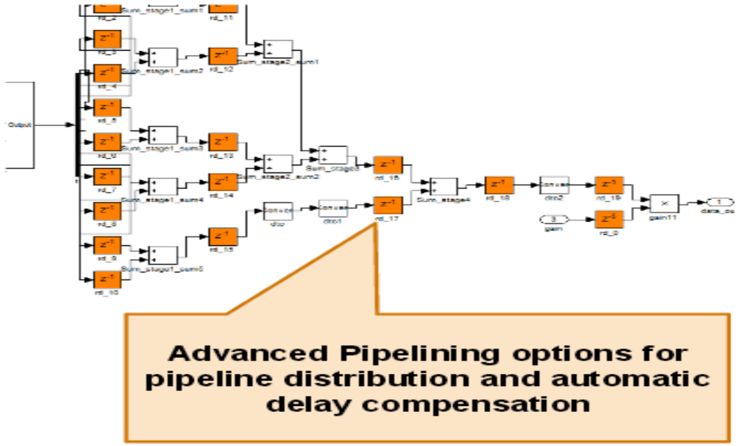


Matlab HDL Specific Techniques

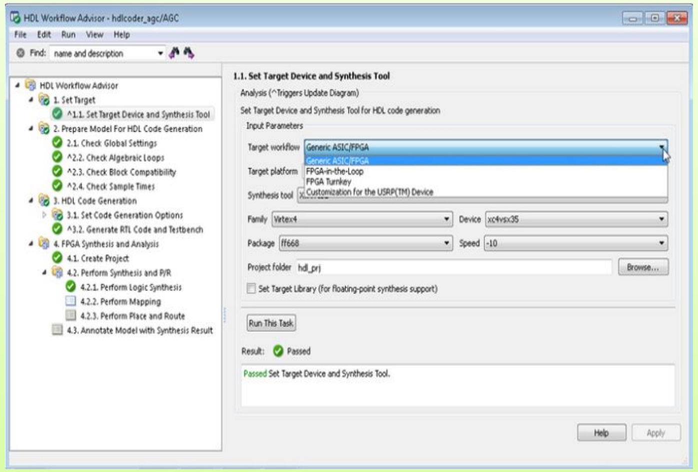
Area Optimization-resource sharing



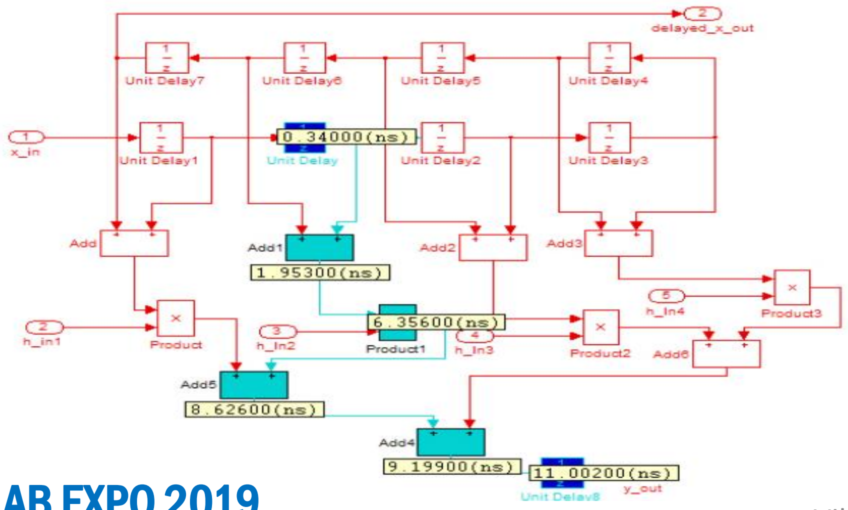
Speed Optimization



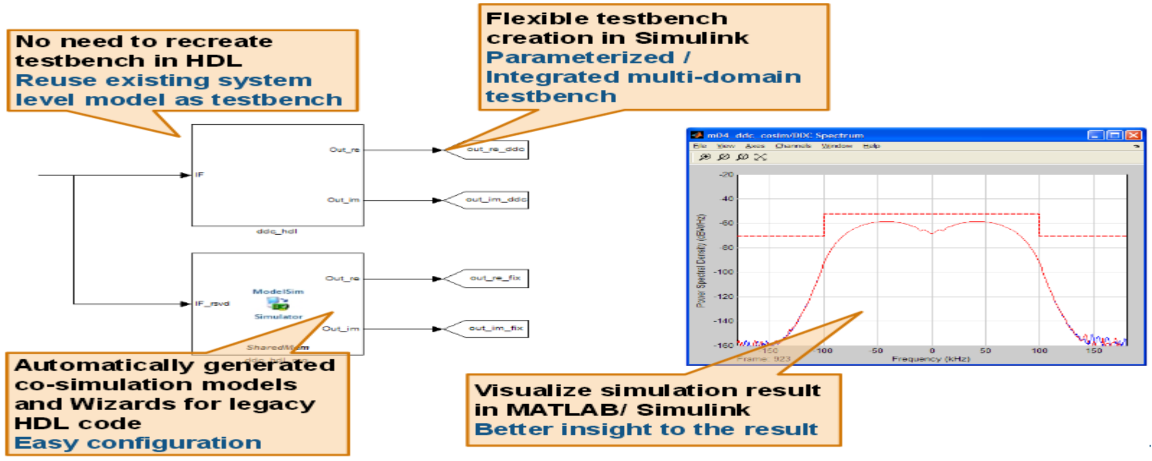
HDL work flow advisor



Back annotating a Simulink model with critical path timing

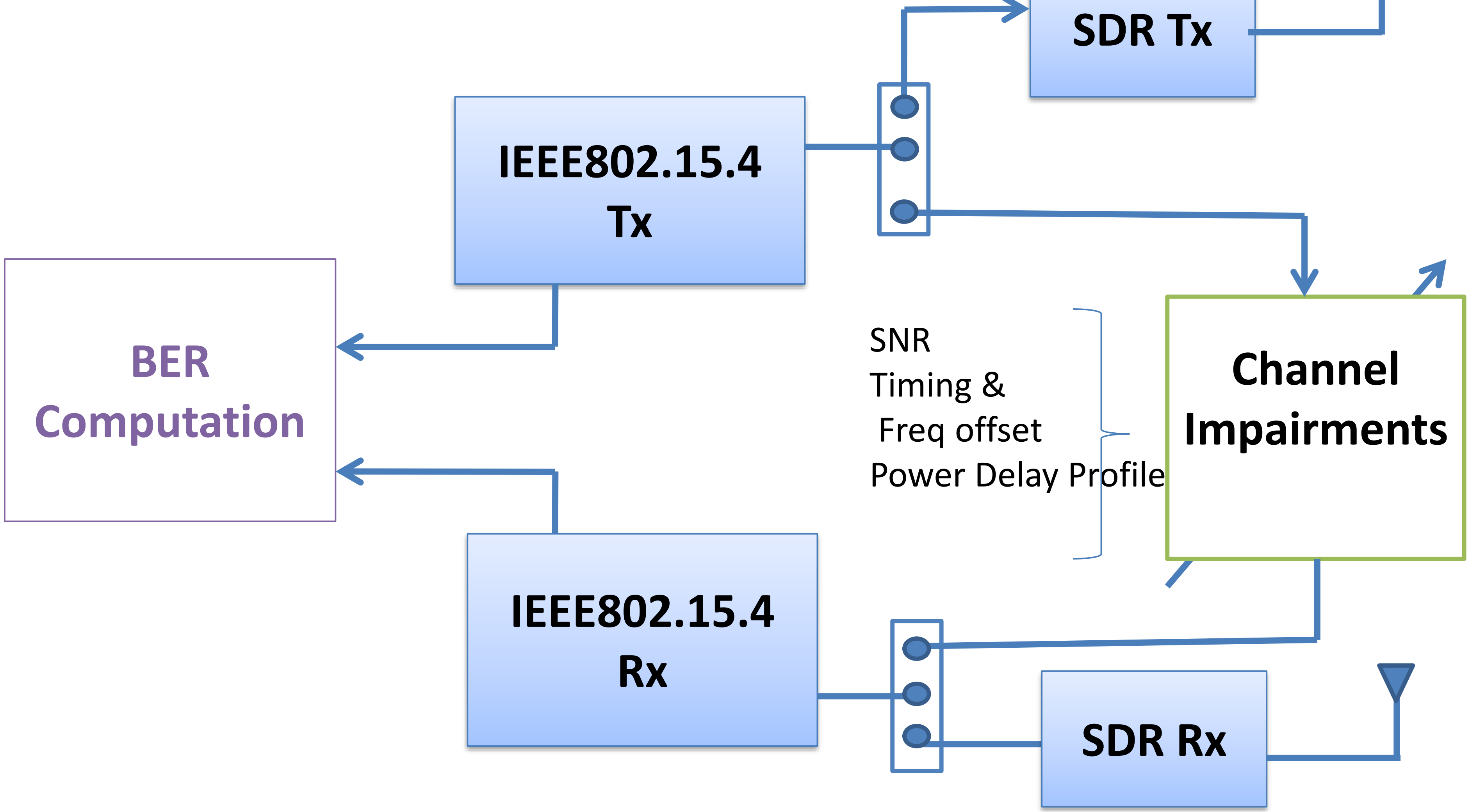


HDL co-simulation



Other: Native floating point support, IP core interface

Our IEEE802.15.4 System Model

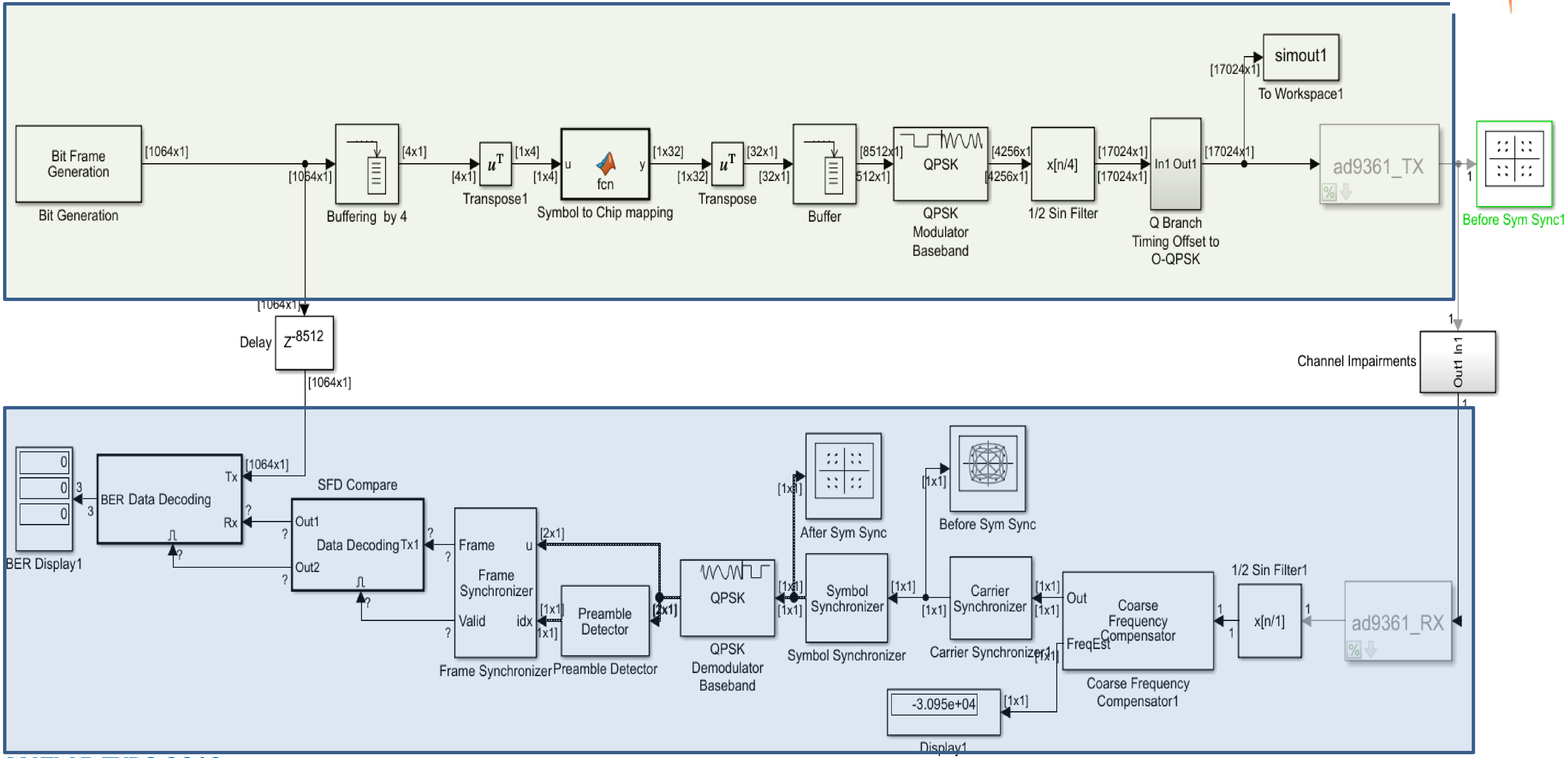


<u>Design Flow</u>

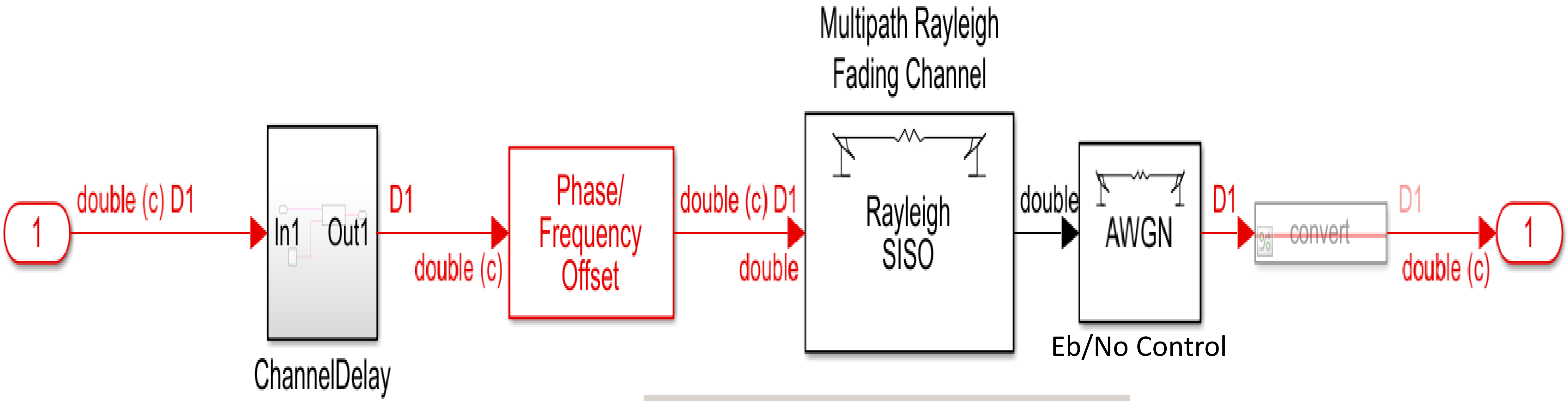
1. Matlab Simulation
2. Modelling in Simulink
3. Performance assessment with AD9361 models
4. HDL compatible Models
5. Testing in Radio I/O mode
6. Splitting the design in to ARM & FPGA
7. Software Interface Model
8. Standalone Model(ARM & FPGA Programming)

- Transmitter and Receiver are partitioned to operate asynchronously.
- The transmitter must be capable of producing IEEE802.15.4 packets and the receiver should demodulate and decode the same correctly
- The user must be able to program various signal/channel impairments for desired tests

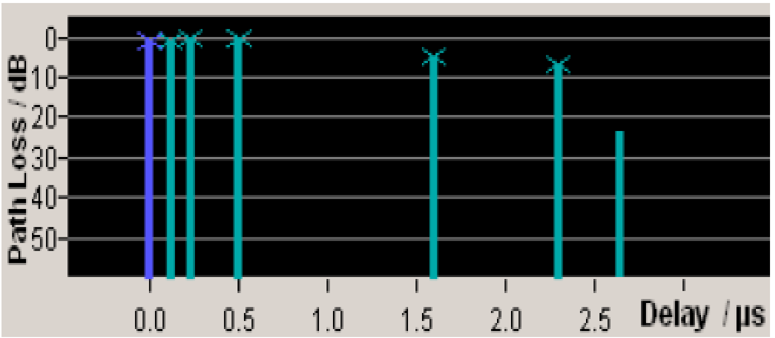
Overall Simulink Model-IEEE802.15.4 TxRx



Simulink Model for Channel Impairments

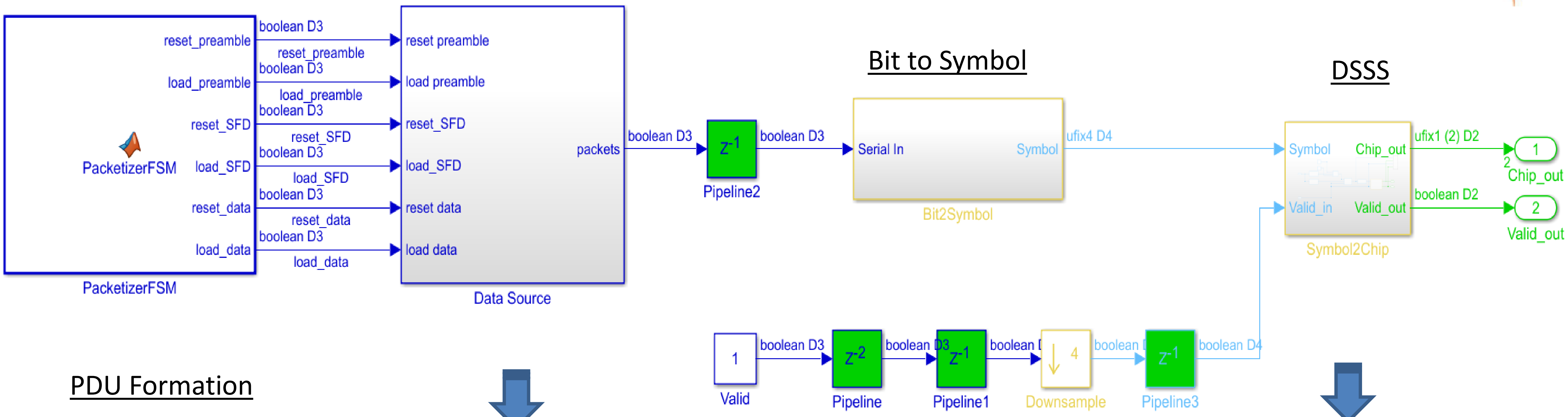


Eb/No Control

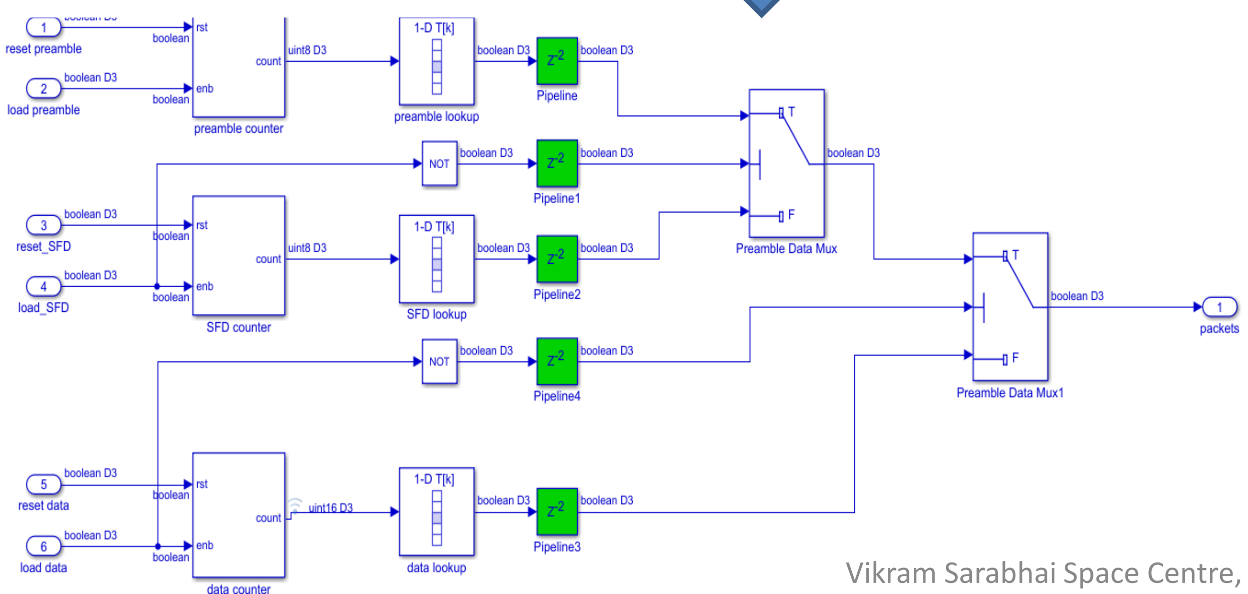


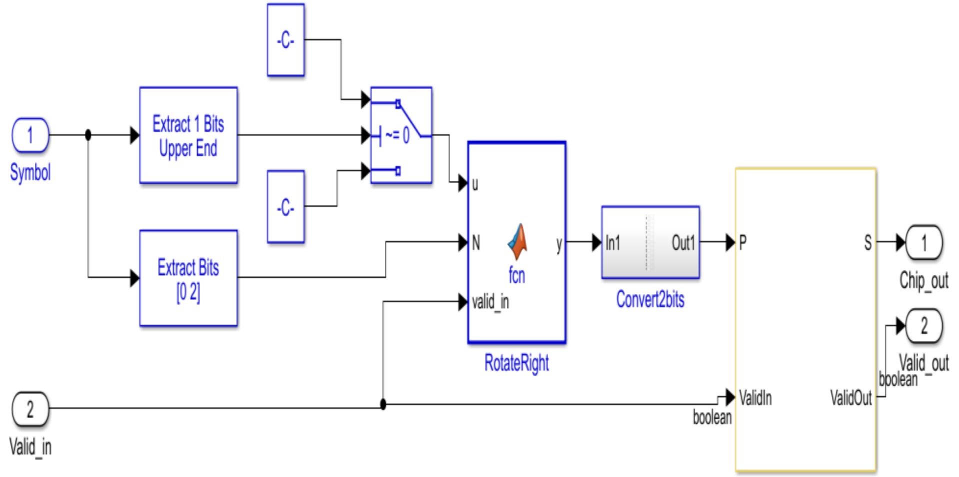
Input-Power delay Profile

HDL compatible Simulink Model -Transmitter-Packetization & DSSS

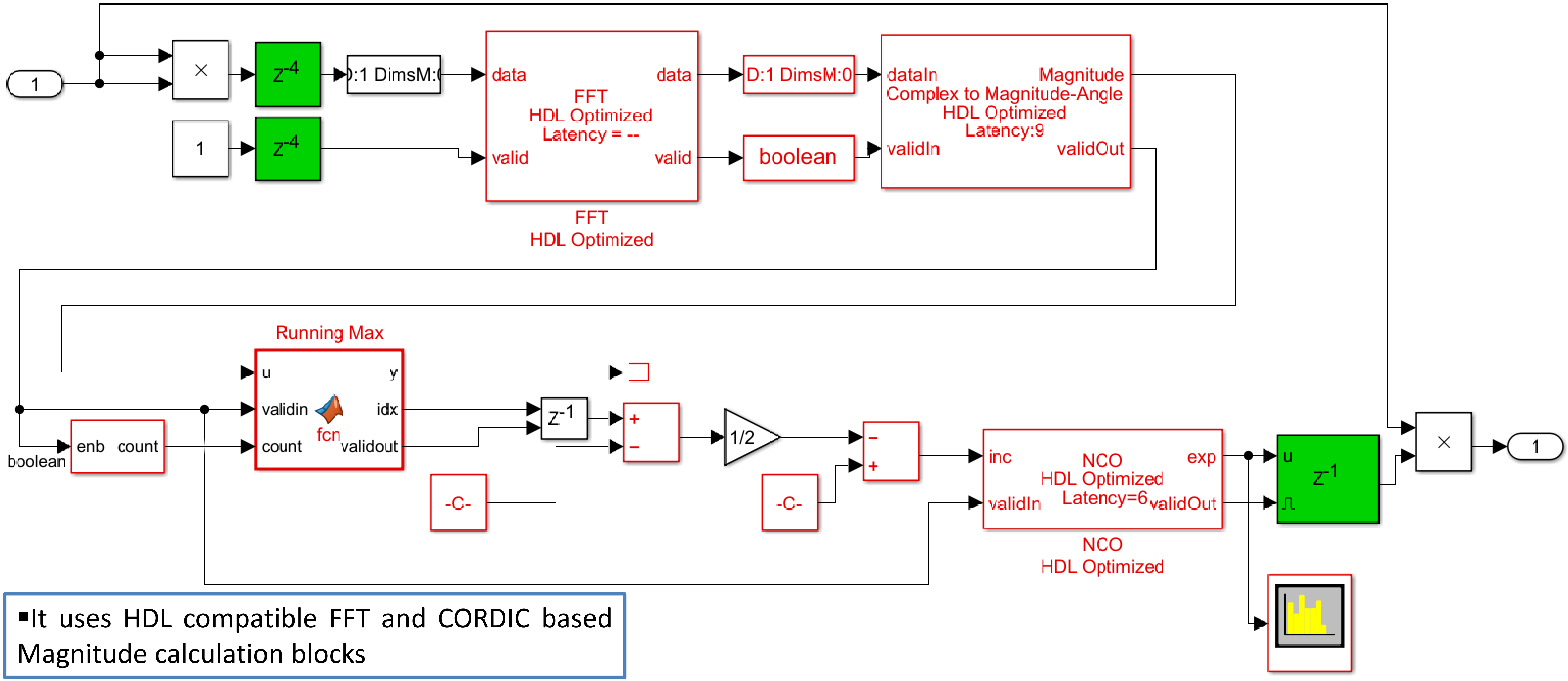


PDU Formation



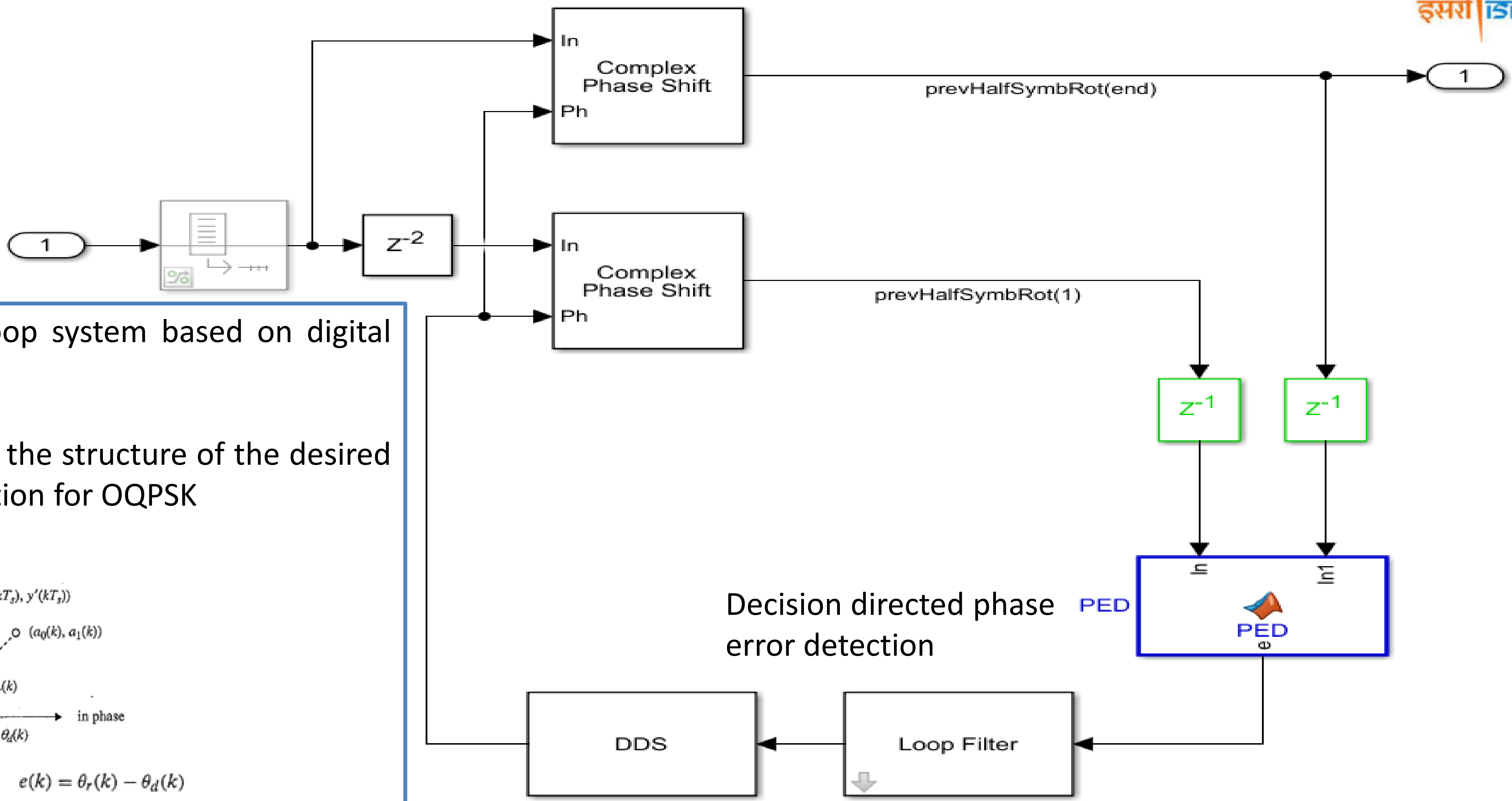


HDL compatible Simulink Model –Coarse Frequency Compensation

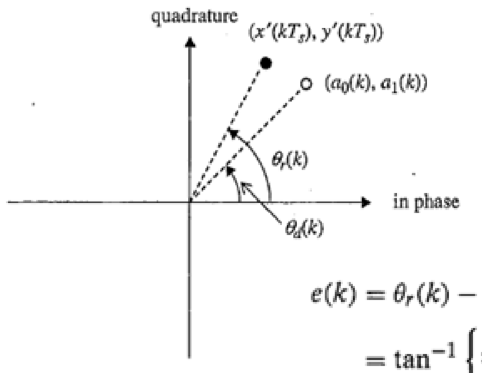


- It uses HDL compatible FFT and CORDIC based Magnitude calculation blocks

HDL compatible Simulink Model – Fine Frequency Compensation

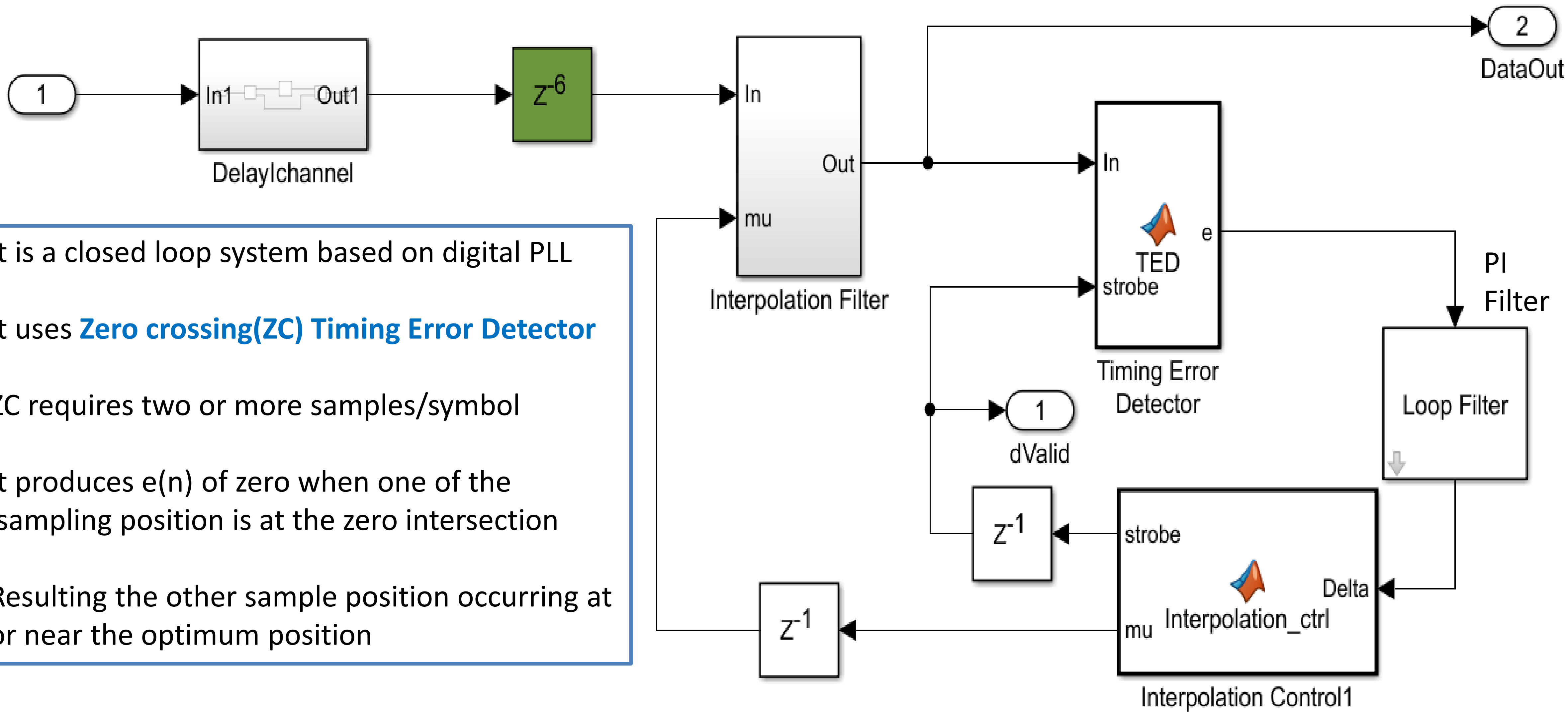


- It is a closed loop system based on digital PLL
- PED is based on the structure of the desired receive constellation for OQPSK



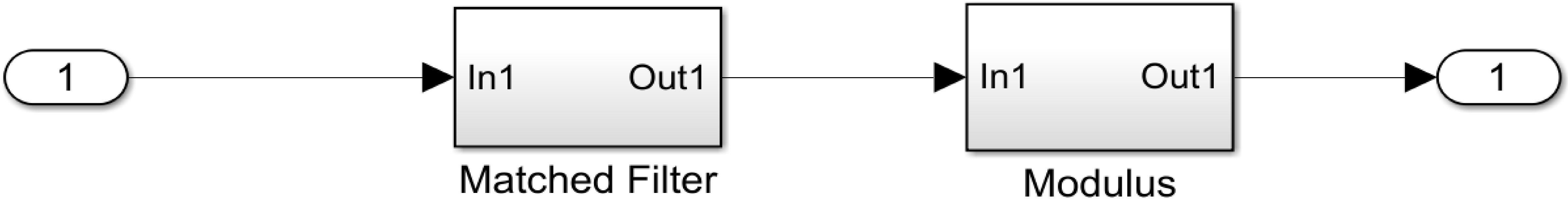
$$e(k) = \theta_r(k) - \theta_d(k)$$
$$= \tan^{-1}\left\{\frac{y'(kT_s)}{x'(kT_s)}\right\} - \tan^{-1}\left\{\frac{a_1(k)}{a_0(k)}\right\}.$$

HDL compatible Simulink Model – Timing recovery



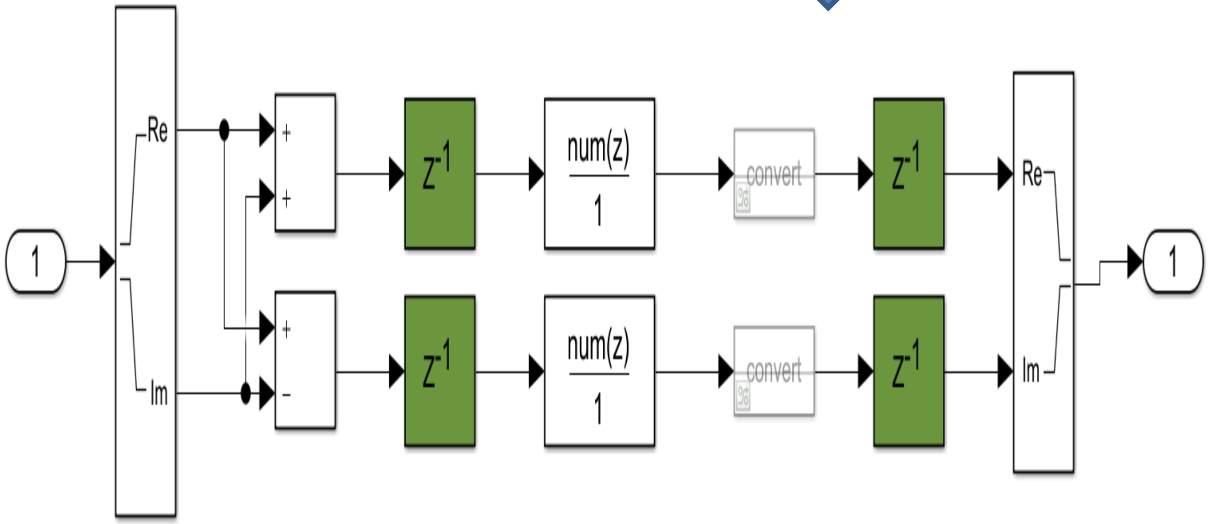
- It is a closed loop system based on digital PLL
- It uses **Zero crossing(ZC) Timing Error Detector**
- ZC requires two or more samples/symbol
- It produces e(n) of zero when one of the sampling position is at the zero intersection
- Resulting the other sample position occurring at or near the optimum position

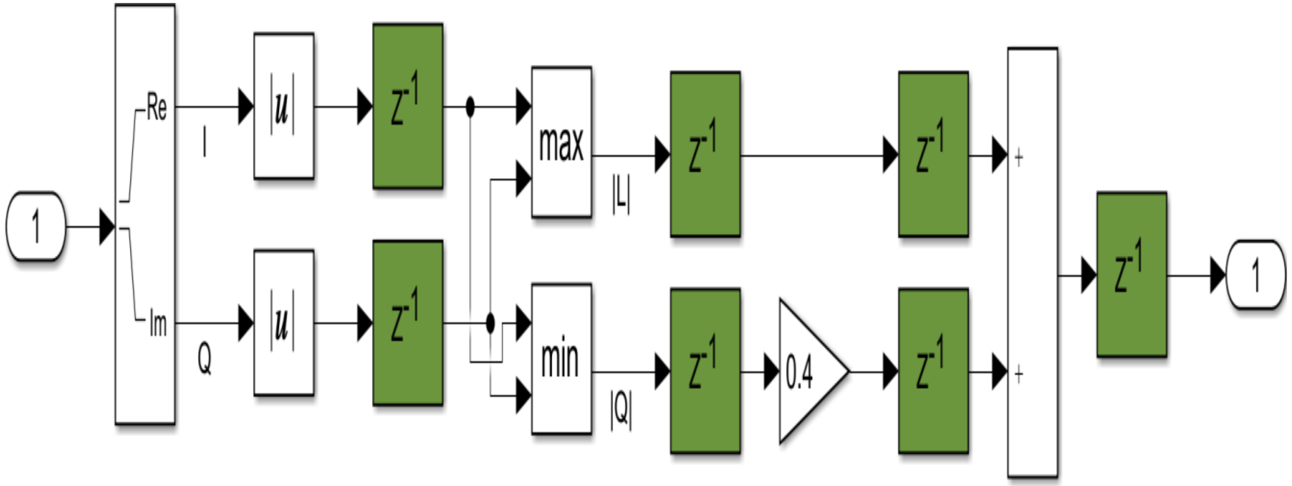
HDL compatible Simulink Model – Preample Detection



FIR filter with the coefficients as the preample sequence

sqrt(I^2+Q^2) is approximated as |L|+0.4*|S|





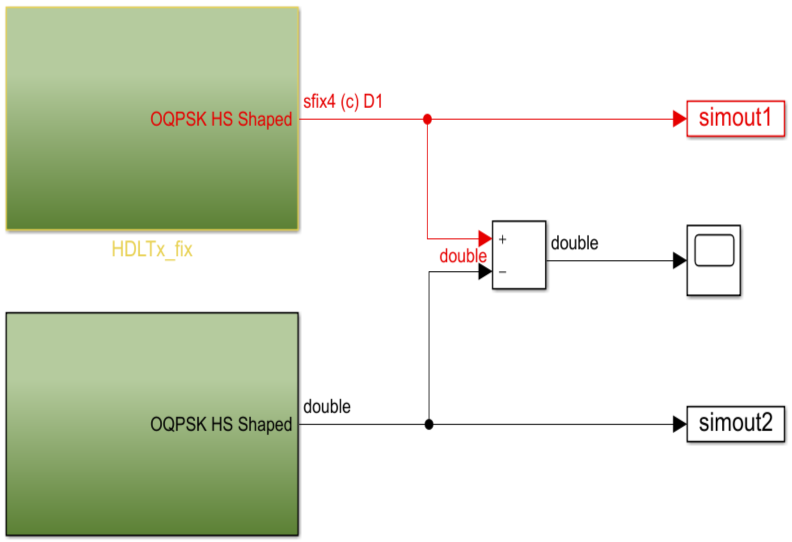
HDL Code generation for IEEE802.15.4 Tx Model

Original Matlab Code of Transmitter

```
clc;
close all;
clear all;
N = 1016; % Number of bits to process (size of the maximum packet)

x = randi([0 1],1,N);%Random number generator
input=x;
ppdu=frame(input);

%Bit-to-symbol mapping
for q=1:length(ppdu)/4
    xsmb(q)=ppdu((q-1)*4+1)+ppdu((q-1)*4+2)*2+ppdu((q-1)*4+3)*4+ppdu((q-1)*4+4)*8;
end

sps=4;
chip_format=chip(xsmb);
y=chip_format;

% modulation with Matlab comm object
oqpskmod = comm.OQPSKModulator('BitInput',true,'SamplesPerSymbol',sps,'PulseShape','Half sine','Sym
waveform = oqpskmod(y');

%normal modulation
y_m=(y*2-1);
```

HDL Compatible Simulink Model

Equivalent Simulink Model



Equivalent VHDL Code

```
Packetization/Bit2Symbol as hdlsrc\zigbee_hdl_tx_model_18a_hdl_setup\Bit2Symbol.vhd.
Packetization/Data Source as hdlsrc\zigbee_hdl_tx_model_18a_hdl_setup\Data_Source.vhd.
Packetization/Symbol2Chip/Convert2bits as hdlsrc\zigbee_hdl_tx_model_18a_hdl_setup\Conver
Packetization/Symbol2Chip/RotateRight as hdlsrc\zigbee_hdl_tx_model_18a_hdl_setup\RotateR
Packetization/Symbol2Chip as hdlsrc\zigbee_hdl_tx_model_18a_hdl_setup\Symbol2Chip.vhd.
Packetization/PacketizerFSM as hdlsrc\zigbee_hdl_tx_model_18a_hdl_setup\PacketizerFSM.vhd
Packetization as hdlsrc\zigbee_hdl_tx_model_18a_hdl_setup\Data_Generation_Packetization.v
system as hdlsrc\zigbee_hdl_tx_model_18a_hdl_setup\Subsystem.vhd.
ndlsrc\zigbee_hdl_tx_model_18a_hdl_setup\Pulse_Shaping.vhd.
SK Modulator Baseband as hdlsrc\zigbee_hdl_tx_model_18a_hdl_setup\QPSK_Modulator_Baseband.
hdlsrc\zigbee_hdl_tx_model_18a_hdl_setup\Symbol_Mapping.vhd.
```

```
-- -------------------------------------------------------------
--
-- Module: HDLTx
-- Source Path: zigbee_hdl_tx_model_18a_hdl_setup/HDLTx
-- Hierarchy Level: 0
--
-- -------------------------------------------------------------
LIBRARY IEEE;
USE IEEE.std_logic_1164.ALL;
USE IEEE.numeric_std.ALL;
USE work.HDLTx_pkg.ALL;

ENTITY HDLTx IS
  PORT( clk                               :   IN    std_logic;
        reset                             :   IN    std_logic;
        clk_enable                        :   IN    std_logic;
        ce_out                            :   OUT   std_logic;
        QPSK_hs_Shaped_re                 :   OUT   std_logic_vector(3 DOWNTO 0);  -- sfix4
        QPSK_hs_Shaped_im                 :   OUT   std_logic_vector(3 DOWNTO 0)   -- sfix4
        );
END HDLTx;


ARCHITECTURE rtl OF HDLTx IS

```

Over The Air Transmission of IEEE 802.15.4 Packets in Radio I/O & Stand Alone modes

4 Msymbols/s complex data equivalent to 2Mc/s data up-sampled by 4 ,generated in MATLAB → 1 Frame data stored in SDR Kit via LAN for playback → D/A Conversion → Up Conversion

PC

LAN

SDR kit

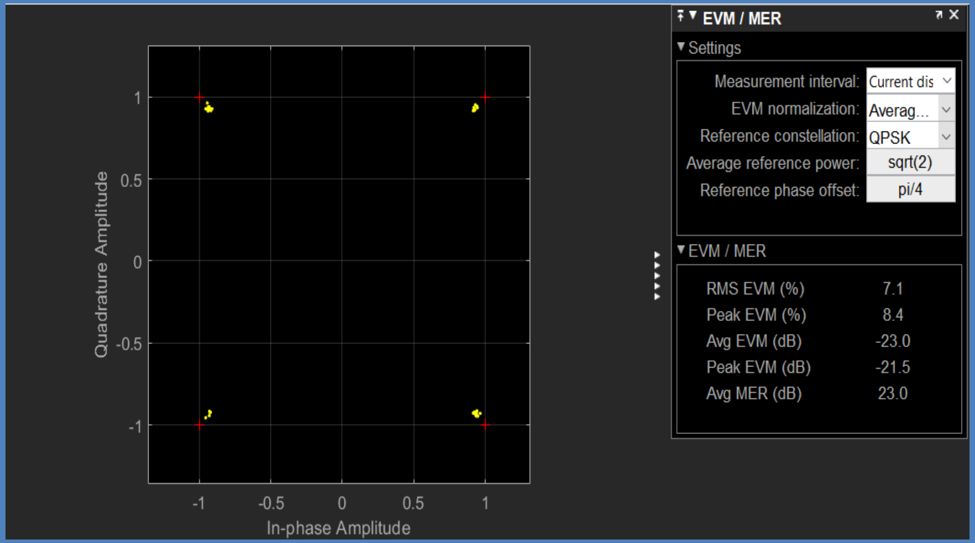
SMART
RF 05 EB Kit from Ti

Packet Reception and Decoding

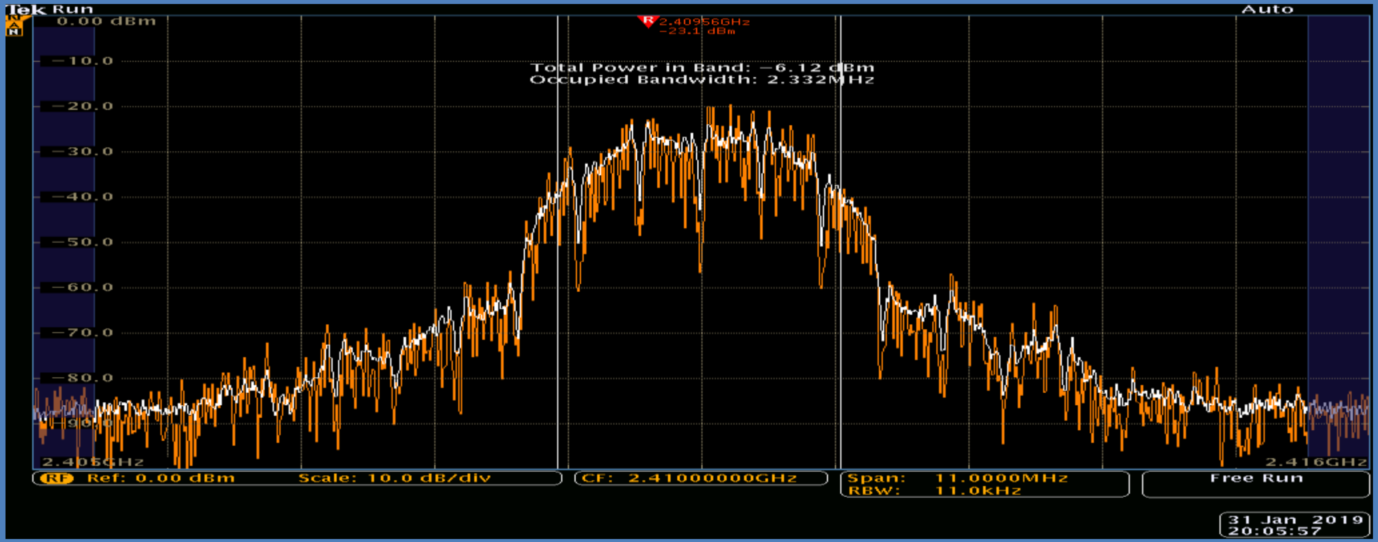
Channel Impairments Control & BER Display

Results

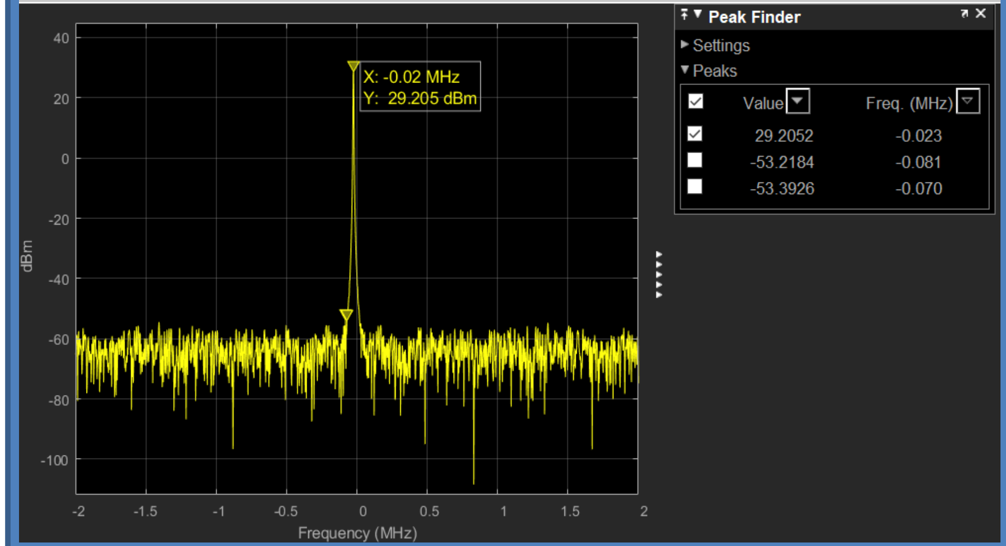
EVM Measurement



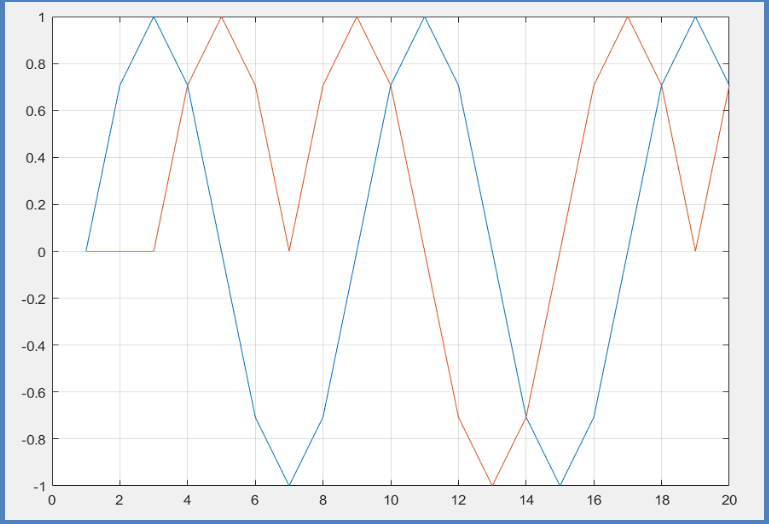
PSD of the transmitted signal captured with VSA



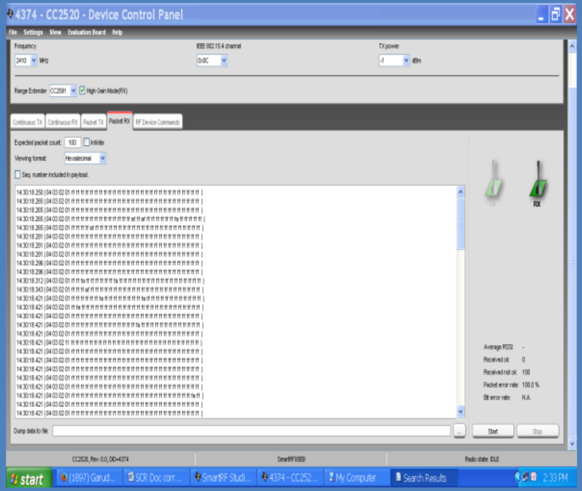
Coarse Frequency Estimation



I & Q Traces after ½ sine filtering



Packets captured with TI's Smart RF EB05 kit



Benefits of model based SDR development

- ✓ A common model for different levels of the development
  - no duplication of effort, better collaboration
- ✓ Less chances of coding errors due to high level implementation
- ✓ Reduced verification effort.
- ✓ Easy signal analysis and performance measurement at different interfaces and levels of the design,
- ✓ Reusability and scalability of the model with less effort.
- ✓ Resource sharing and pipelining are much easier as compared to bare VHDL coding.
- ✓ Optimized HDL code and small FPGA resource consumption suitable to deploy in the actual flight systems also.
- ✓ A unified hardware platform for different communication applications.
- ✓ Fast prototyping of the concept buy just placing and interconnecting the subsystem models.

Speaker Details
Email: bibin_varghese@vssc.gov.in
bibinvarghesep@gmail.com
Linkedin: https://www.linkedin.com/in/bibin-varghese-480541b1/

Thank u..