

Wireless Embedded Systems

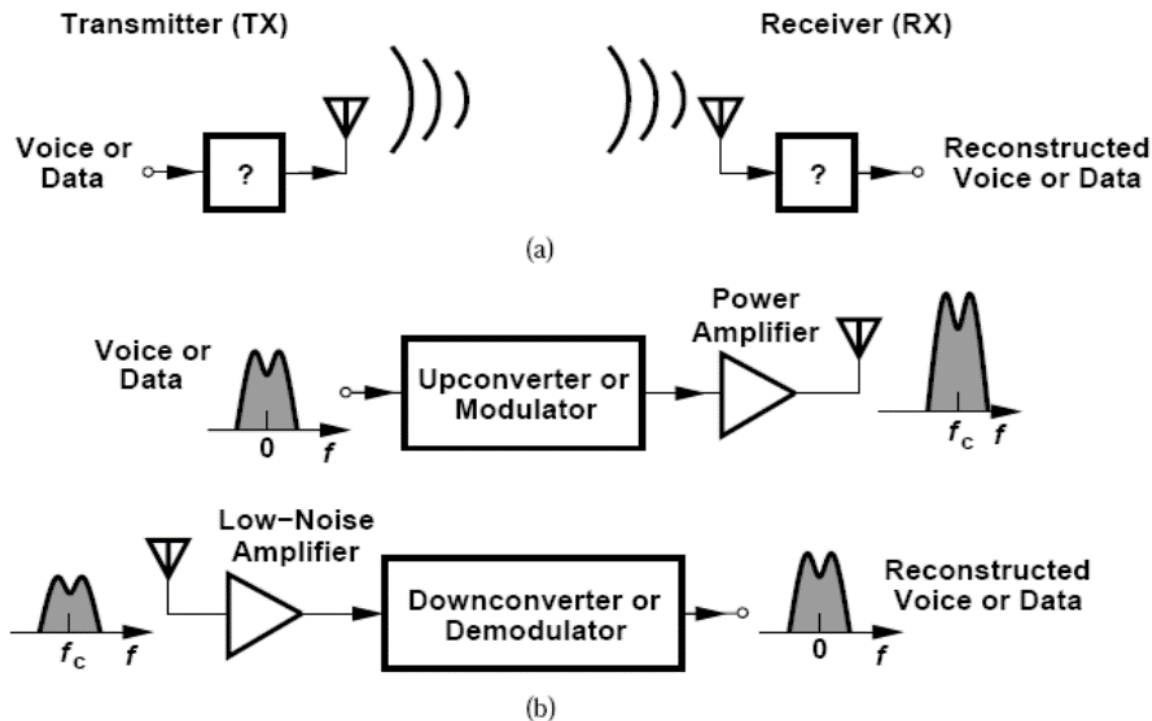
Department of Radio & Information Communications Engineering
CHUNGNAM NATIONAL UNIVERSITY

김병관

byungkwan.kim@cnu.ac.kr

Software-Defined Radio

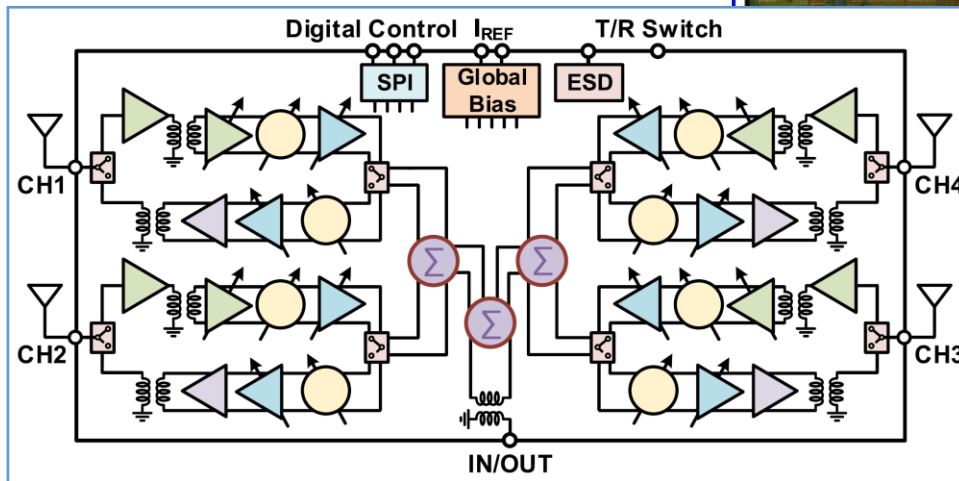
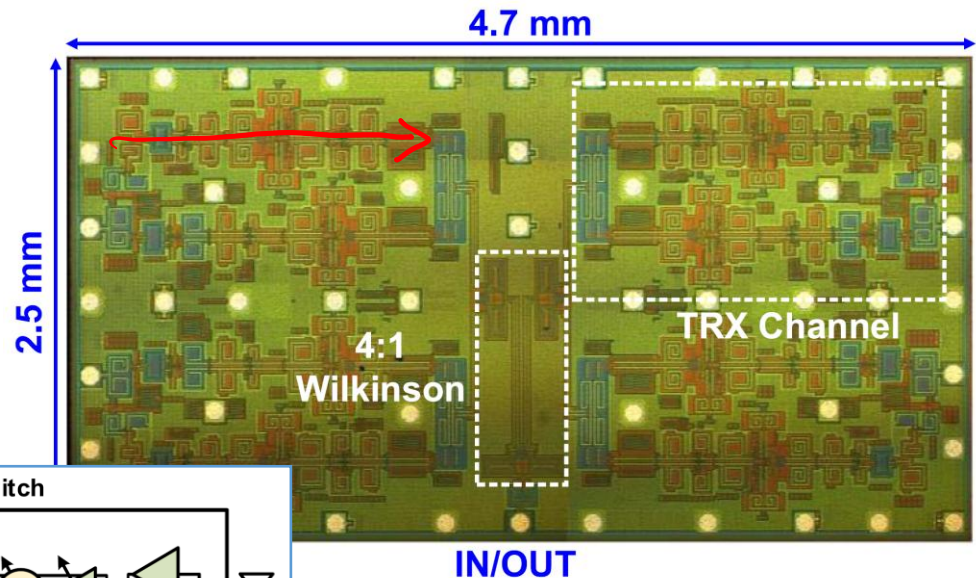
- Introduction
 - Wireless Communication System
 - **Transmit**, **Receive**, **Processing**



Software-Defined Radio

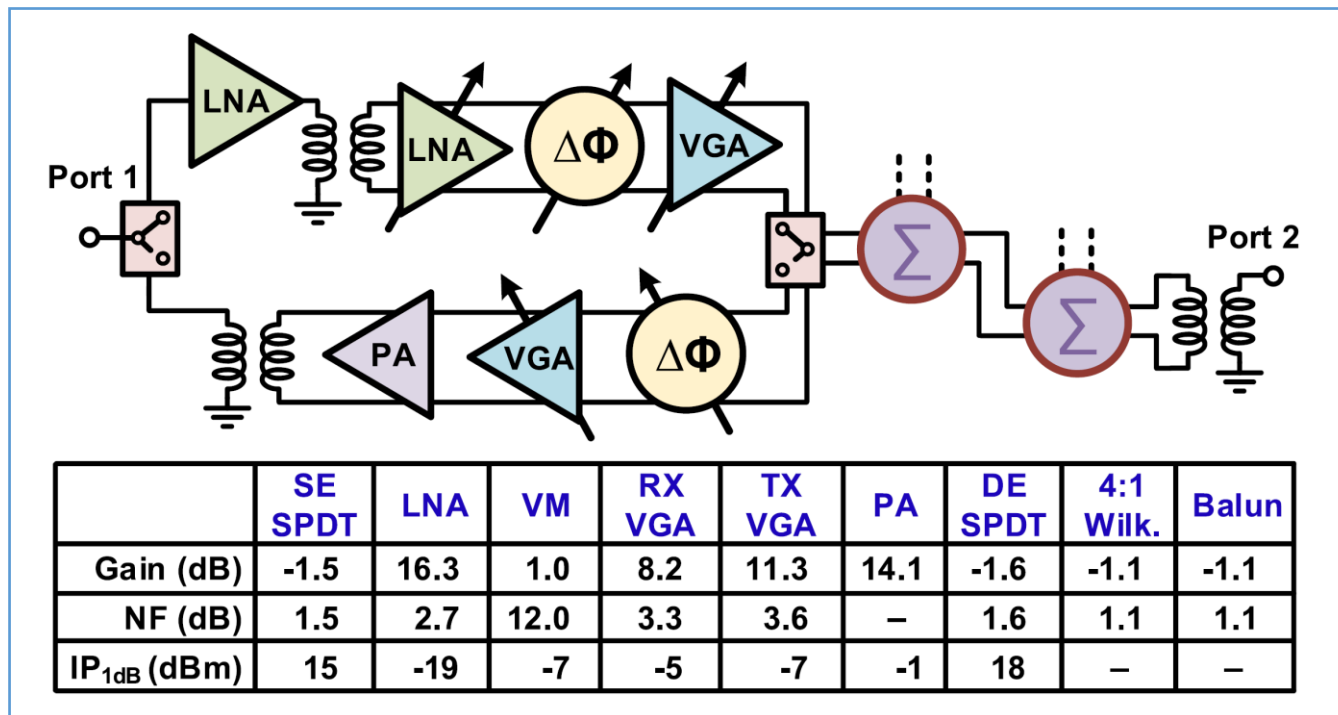
- Example of Wireless Transceiver SoC

- Frequency: 28GHz
- Beamforming
- Scalable T/Rx



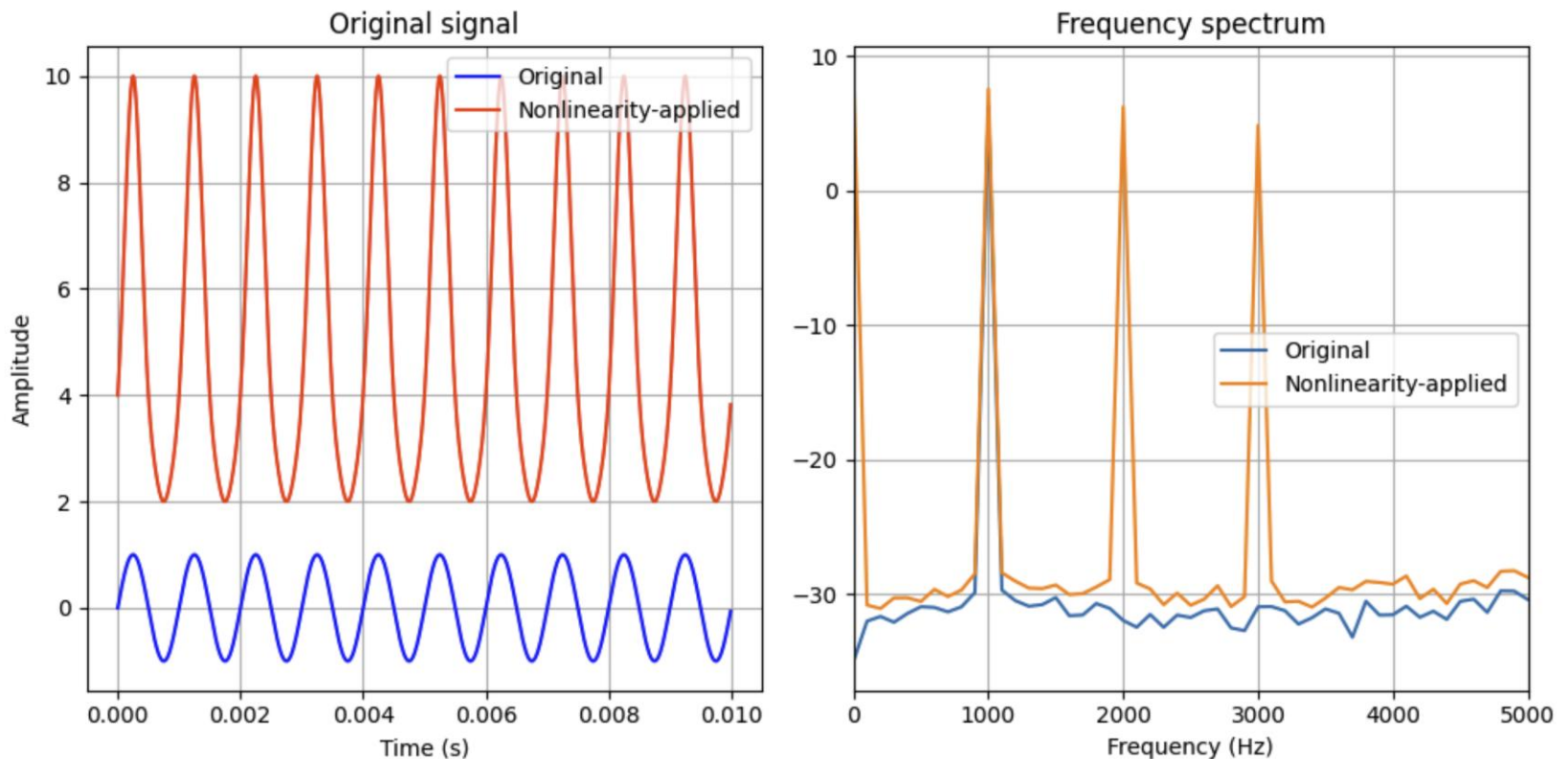
Software-Defined Radio

- Example of wireless transceiver
 - Systematic Design: Link Budget
 - How much power will be delivered?
 - How much will the signal be distorted?



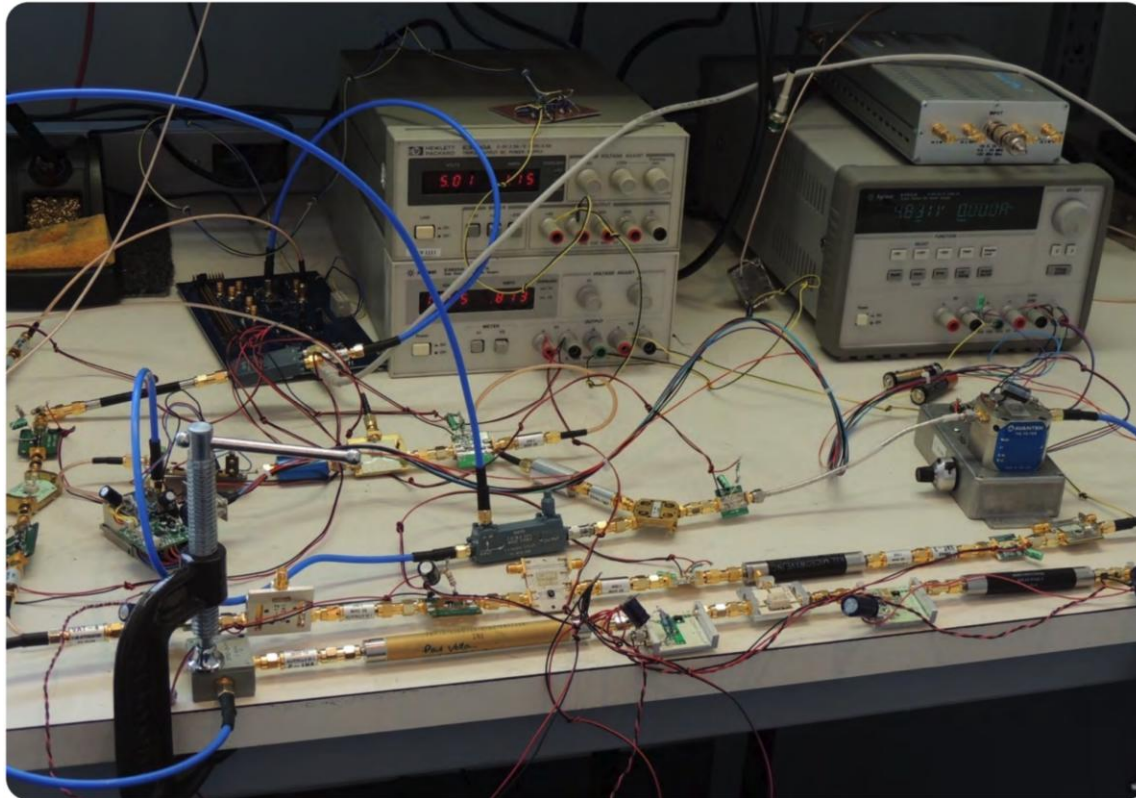
Software-Defined Radio

- Example of Wireless Transceiver SoC
 - How much the signal will be distorted by nonlinearity?



Software-Defined Radio

- Why Software-Defined Radio?
 - Conventional Transceiver Prototyping
 - Hardware-dependent development



Software-Defined Radio

- What is Software-Defined Radio?
 - Modern Transceiver Prototyping using SDR



<https://www.ni.com/ko-kr/innovations/white-papers/14/5g-massive-mimo-testbed--from-theory-to-reality--.html>

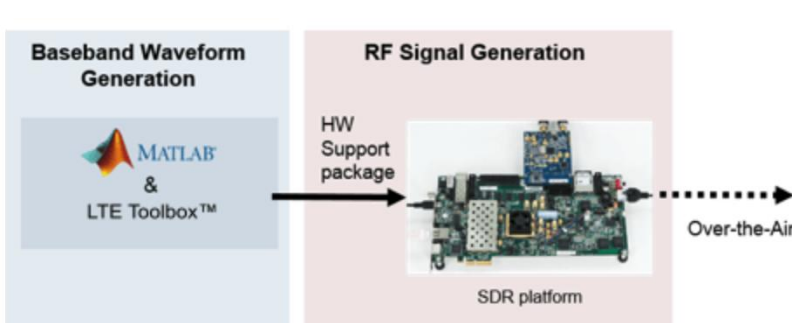
Software-Defined Radio

- What is Software-Defined Radio?
 - *Software defines how to transmit and receive "signal"*
 - It can work with cellphone, wi-fi, bluetooth, radar,... anything



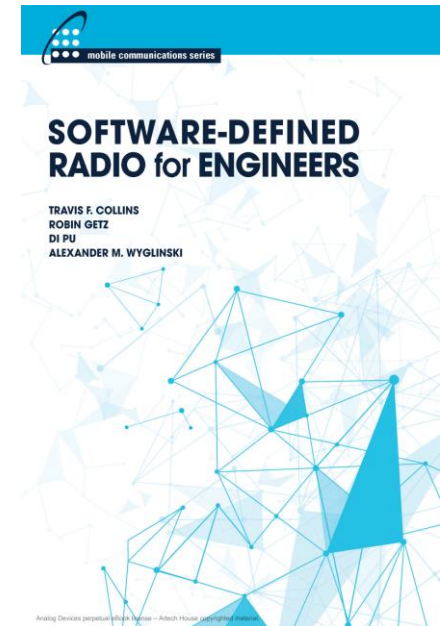
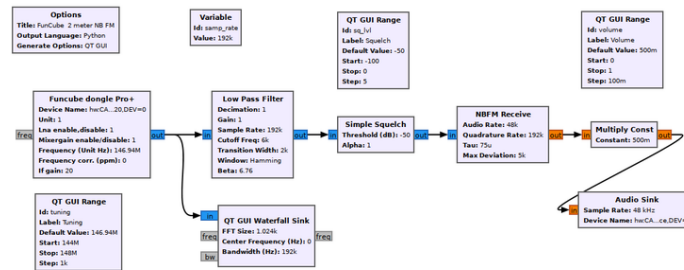
Software-Defined Radio

- What is Software-Defined Radio?
 - MATLAB (Simulink), LabVIEW, GNU Radio, C++/C#, **Python**, ...
 - Windows, Linux, Mac, embedded platforms (Raspberry Pi, ...)



PySDR: A Guide to SDR and DSP using Python

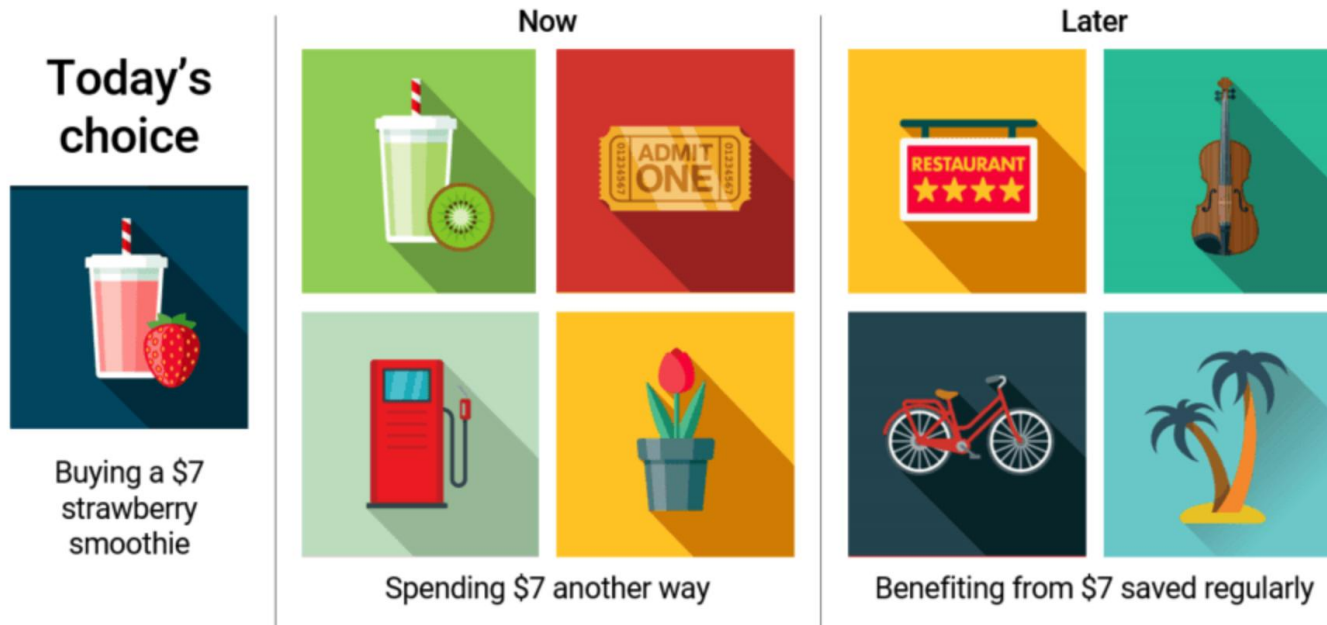
By Dr. Marc Lichtman



Software-Defined Radio

- What is Software-Defined Radio?
 - Trade-off:
 - **Flexibility** vs. Performance, Cost, Power consumption, ...

Every decision involves trade-offs

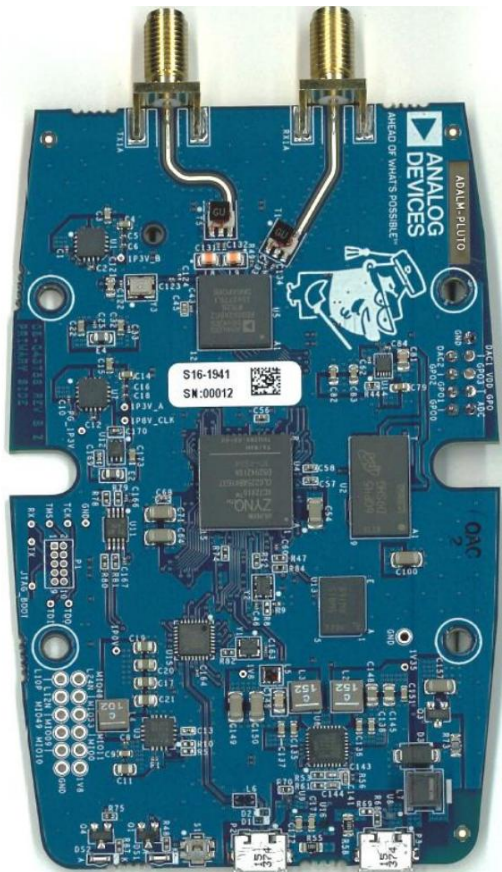


Inspired by Andrea Caceres-Santamaria, "Money and Missed Opportunities." *Page One Economics*, October 2019. Icons by Getty Images.

FEDERAL RESERVE BANK *of* ST. LOUIS

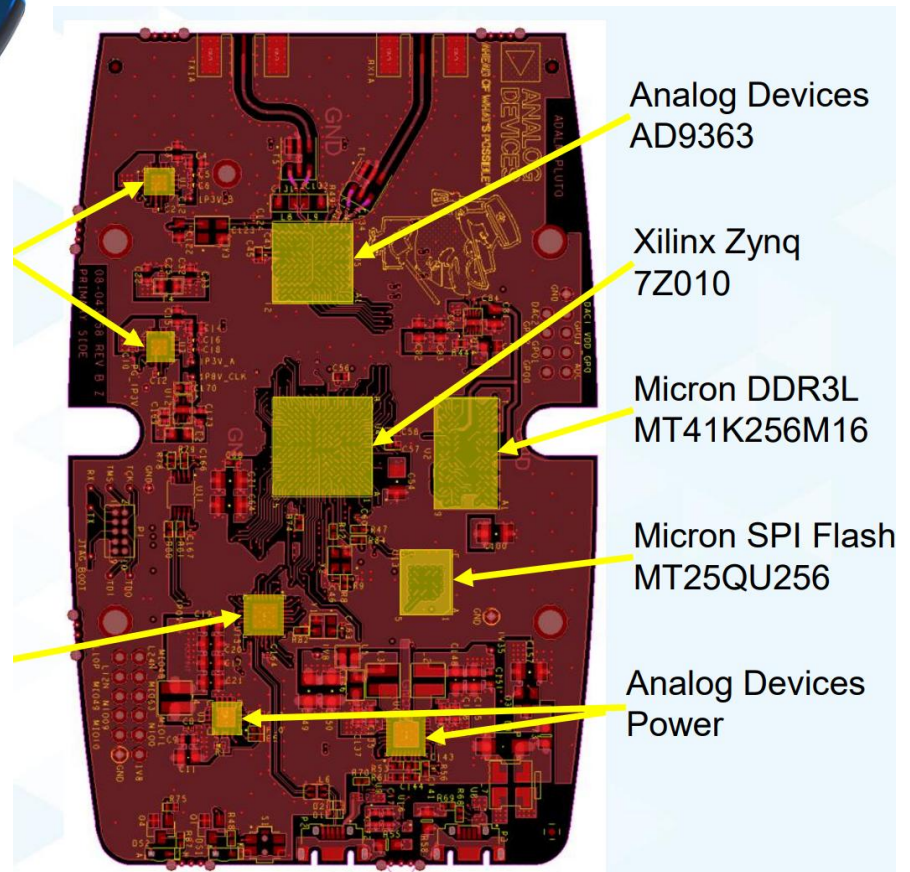
Software-Defined Radio

- ADALM-Pluto



PMIC
(Power
Management
IC)

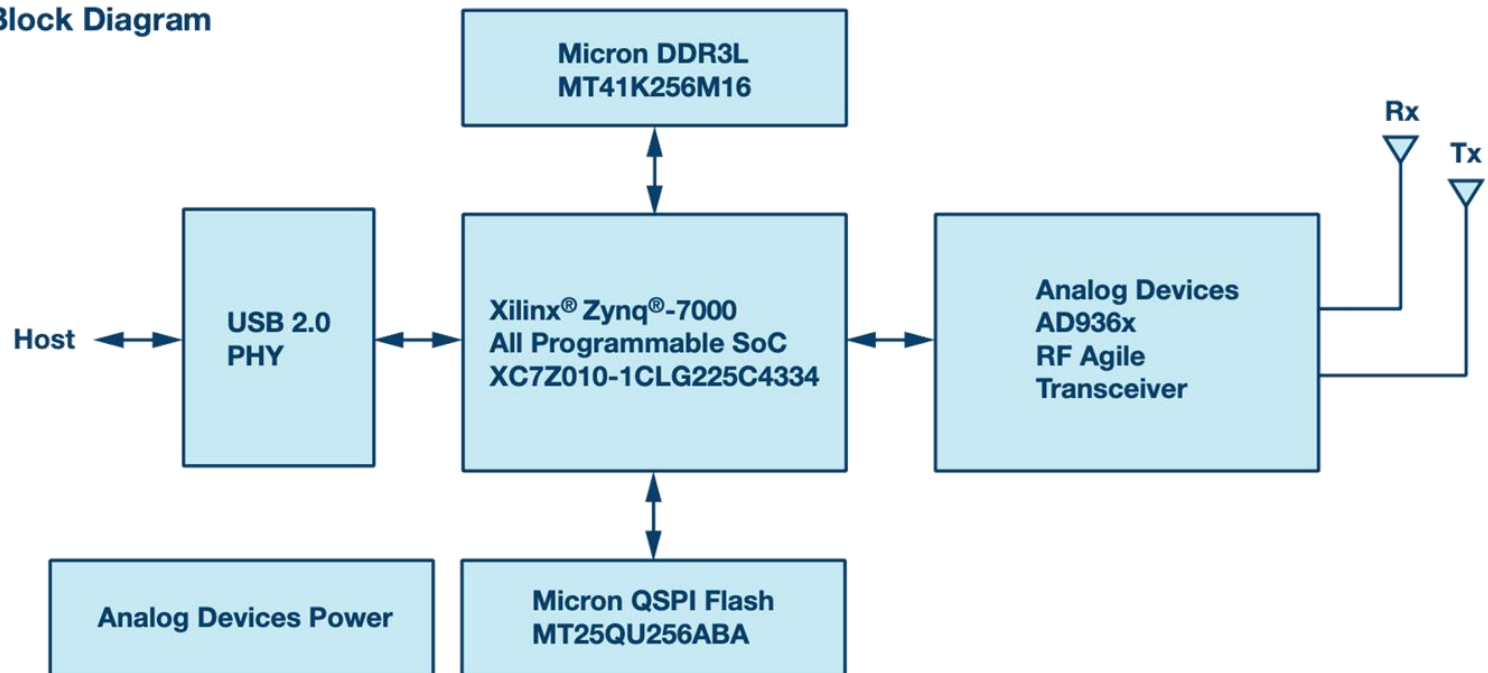
USB Comm.
(USB3320)



Software-Defined Radio

- ADALM-Pluto
 - RF Transceiver + SoC + Memory

Simplified Block Diagram



Software-Defined Radio

- ADALM-Pluto
 - To control AD9361, we use embedded software (iio)

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`



Software-Defined Radio

- ADALM-Pluto in Python

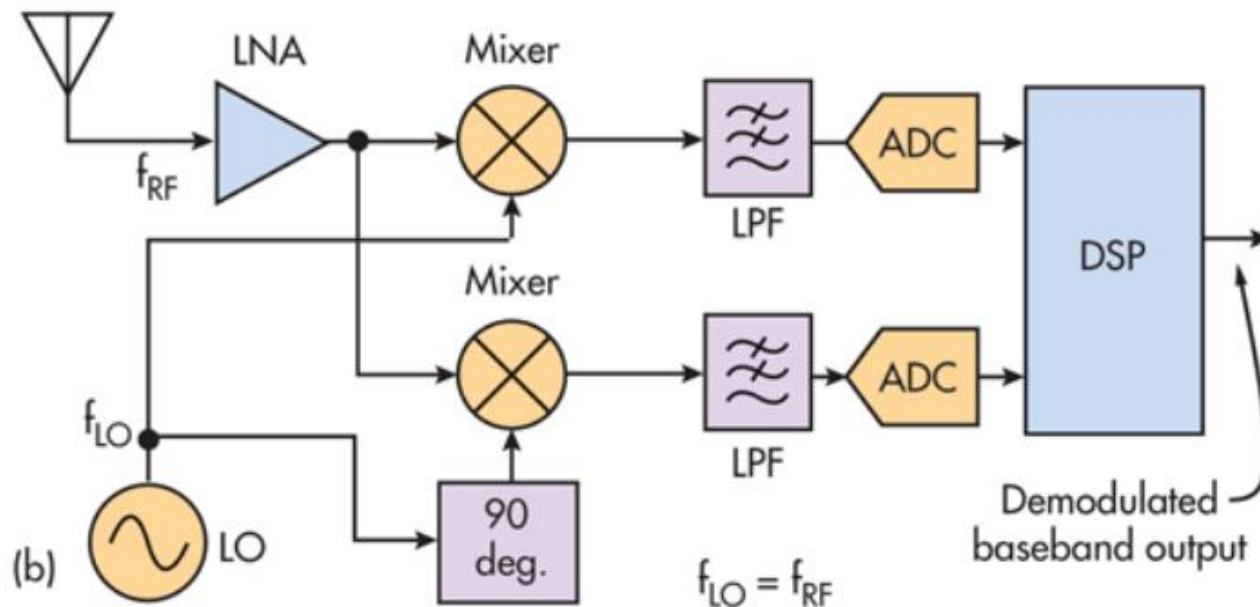
```
# Import library  
import adi  
# Create radio object  
sdr = adi.Pluto()  
# Configure properties  
sdr.rx_rf_bandwidth = 4000000  
# Get data  
data = sdr.rx()
```



- <https://github.com/analogdevicesinc/pyadi-iio/tree/master/examples>
- <https://pypi.org/project/pyadi-iio/>
- <https://wiki.analog.com/resources/tools-software/linux-software/pyadi-iio>

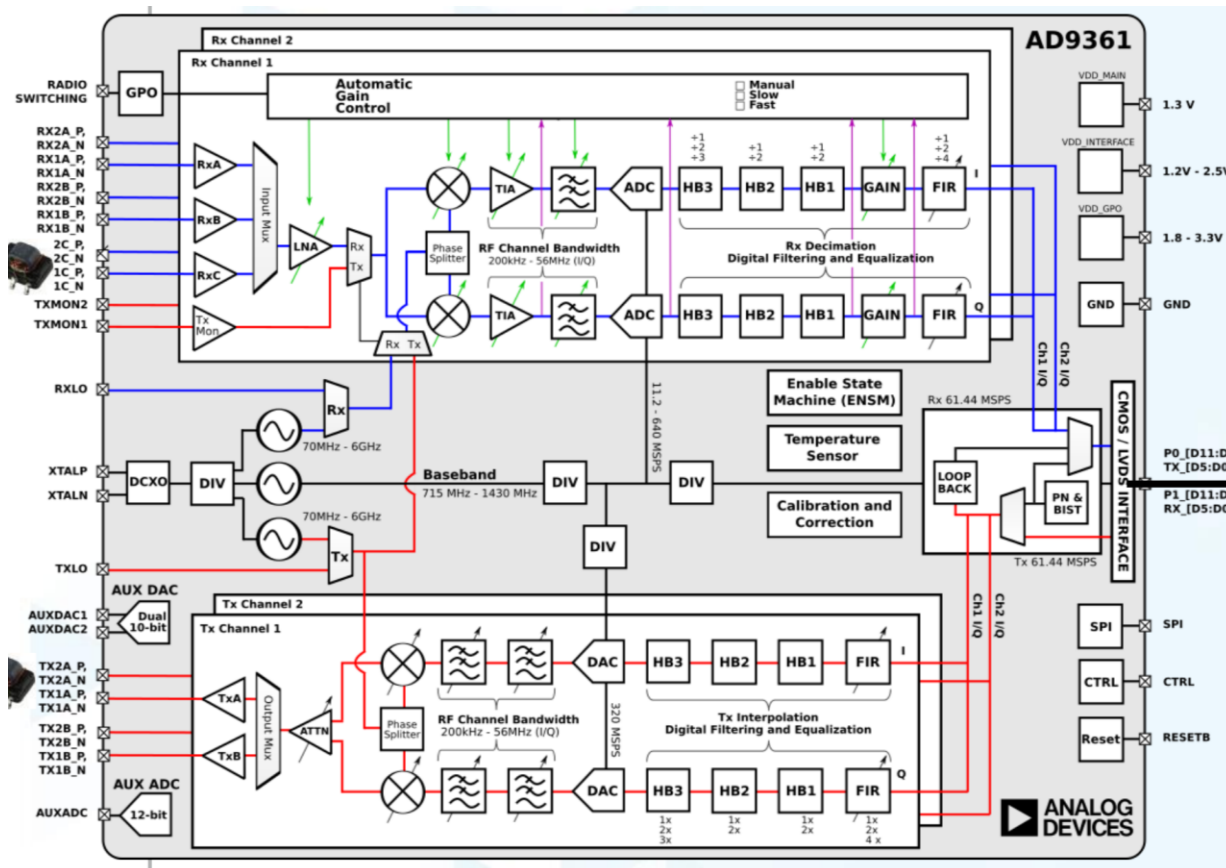
Software-Defined Radio

- ADALM-Pluto
 - Zero-IF Structure
 - Control LO to select center frequency
 - Control ADC parameters for bandwidth



Software-Defined Radio

- ADALM-Pluto
 - Functional Block Diagram



Software-Defined Radio

- ADALM-Pluto
 - AD9363 RF Agile Transceiver

FEATURES

Radio frequency (RF) 2 × 2 transceiver with integrated 12-bit DACs and ADCs

Wide bandwidth: 325 MHz to 3.8 GHz

Supports time division duplex (TDD) and frequency division duplex (FDD) operation

Tunable channel bandwidth (BW): up to 20 MHz

Receivers: 6 differential or 12 single-ended inputs

Superior receiver sensitivity with a noise figure: 3 dB

Receive (Rx) gain control

Real-time monitor and control signals for manual gain

Independent automatic gain control (AGC)

Dual transmitters: 4 differential outputs

Highly linear broadband transmitter

Transmit (Tx) error vector magnitude (EVM): -34 dB

Tx noise: ≤ -157 dBm/Hz noise floor

Tx monitor: 66 dB dynamic range with 1 dB accuracy

Integrated fractional N synthesizers

2.4 Hz local oscillator (LO) step size

CMOS/LVDS digital interface

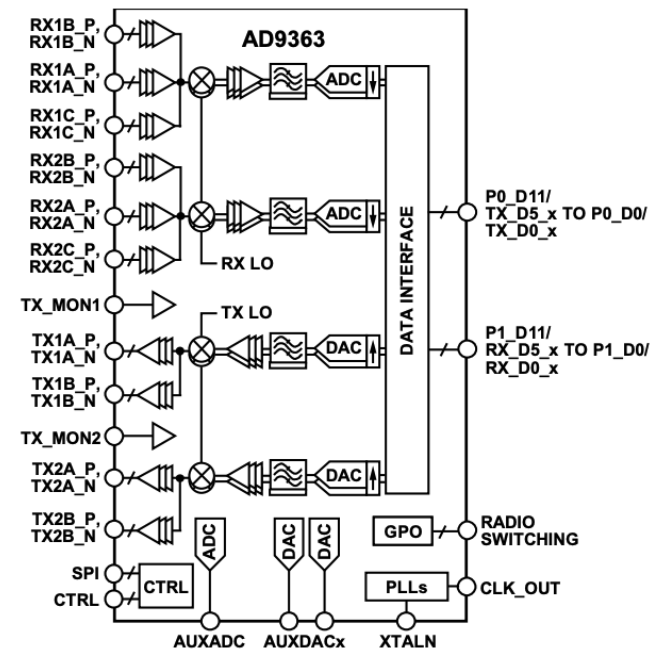
APPLICATIONS

3G enterprise femtocell base stations

4G femtocell base stations

Wireless video transmission

FUNCTIONAL BLOCK DIAGRAM



NOTES

1. SPI, CTRL, P0_D11/TX_D5_x TO P0_D0/TX_D0_x, P1_D11/RX_D5_x TO P1_D0/RX_D0_x, AND RADIO SWITCHING CONTAIN MULTIPLE PINS.

10558-001

Figure 1.

Software-Defined Radio

- ADALM-Pluto
 - AD9363 RF Agile Transceiver
 - A semiconductor that includes microwave devices & digital circuits

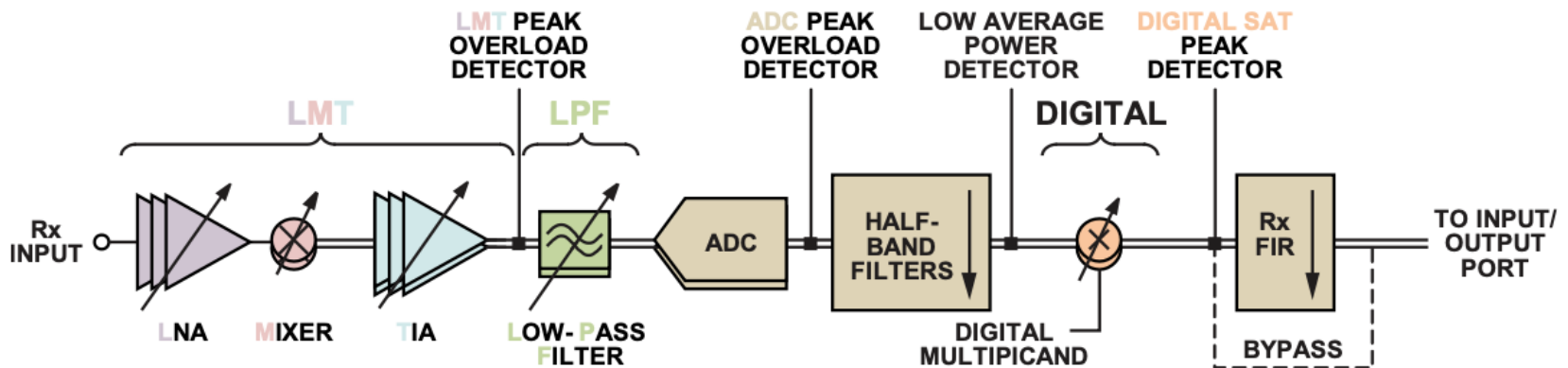


Figure 17. AD9363 Rx Signal Path

14969-019

Software-Defined Radio

- ADALM-Pluto
 - AD9363 RF Agile Transceiver
-  ADI AD9363, RF Agile Transceiver:
 - 1 Transmit, 1 Receive channel (with separate tuning frequencies)
 - Tuning range: 325 MHz - 3.8 GHz
 - 2.4 Hz LO step size
 - Tunable channel bandwidth: 200 kHz - 20 MHz
 - Integrated 12-bit DACs (Tx) and ADCs (Rx)
 - Variable output data rates: 61.44 MSPS - 65.1 kSPS
 - Modulation Accuracy (EVM): ≤ -40 dB (typical, not measured on every 1000 samples)
 - Internal I/Q correction and calibration
 - Tx to Rx Isolation:
 - Tx Specifications
 - Maximum Output Power:
 - Rx Specifications
 - Min sensitivity:
 - RX gain control: 0 to +74.5dB (800 MHz)
 - Manual Modes
 - Slow Attack
 - Fast Attack
 - Received Signal Strength Indicator: 100 dB (± 2 dB)
-  Xilinx Zynq XC7Z010-1CLG225C
 -  Zynq-7000 All Programmable SoC Overview
 -  Datasheet
 - FPGA
 - Logic Cells: 28k
 - Block RAM: 2.1Mb
 - DSP Slices 80
 - ARM Processing System
 - Single-core ARM® Cortex™-A9 MPCore™
 - 667 MHz
 - USB 2.0 (included in the Zynq)
 - streams up to 4MSPS with no dropped samples

Software-Defined Radio

- ADALM-Pluto
 - AD9363 RF Agile Transceiver

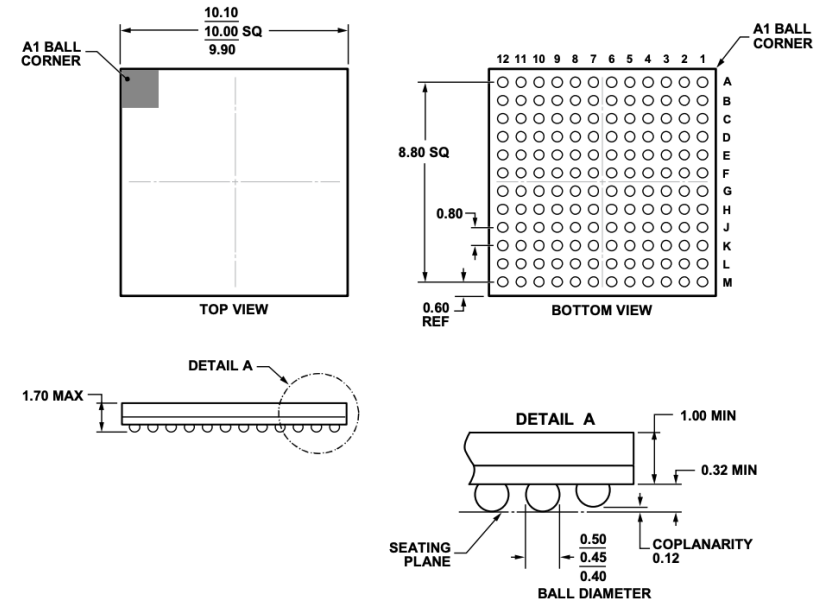
	1	2	3	4	5	6	7	8	9	10	11	12
A	RX2A_N	RX2A_P	DNC	VSSA	TX_MON2	VSSA	TX2A_N	TX2A_P	TX2B_N	TX2B_P	VDDA1P1_TX_VCO	VSSA
B	VSSA	VSSA	AUXDAC1	GPO_3	GPO_2	GPO_1	GPO_0	VDD_GPO	VDDA1P3_TX_LO	VDDA1P3_TX_VCO_LDO	TX_VCO_LDO_OUT	VSSA
C	RX2C_P	VSSA	AUXDAC2	TEST/ENABLE	CTRL_IN0	CTRL_IN1	VSSA	VSSA	VSSA	VSSA	VSSA	VSSA
D	RX2C_N	VDDA1P3_RX_RF	VDDA1P3_RX_TX	CTRL_OUT0	CTRL_IN3	CTRL_IN2	P0_D9/TX_D4_P	P0_D7/TX_D3_P	P0_D5/TX_D2_P	P0_D3/TX_D1_P	P0_D1/TX_D0_P	VSSD
E	RX2B_P	VDDA1P3_TX_LO	VDDA1P3_TX_LO_BUFFER	CTRL_OUT1	CTRL_OUT2	CTRL_OUT3	P0_D11/TX_D5_P	P0_D8/TX_D4_N	P0_D6/TX_D3_N	P0_D4/TX_D2_N	P0_D2/TX_D1_N	P0_D0/TX_D0_N
F	RX2B_N	VDDA1P3_RX_VCO_LDO	VSSA	CTRL_OUT6	CTRL_OUT5	CTRL_OUT4	VSSD	P0_D10/TX_D5_N	VSSD	FB_CLK_P	VSSD	VDDA1P3_DIG
G	VSSA	RX_VCO_LDO_OUT	VDDA1P1_RX_VCO	CTRL_OUT7	EN_AGC	ENABLE	RX_FRAME_N	RX_FRAME_P	TX_FRAME_P	FB_CLK_N	DATA_CLK_P	VSSD
H	RX1B_P	VSSA	VSSA	TXNRX	VSSA	VSSA	VSSD	P1_D11/RX_D5_N	TX_FRAME_N	VSSD	DATA_CLK_N	VDD_INTERFACE
J	RX1B_N	VSSA	VDDA1P3_RX_SYNTH	SPI_DI	SPI_CLK	CLK_OUT	P1_D10/RX_D5_N	P1_D9/RX_D4_P	P1_D7/RX_D3_P	P1_D5/RX_D2_P	P1_D3/RX_D1_P	P1_D1/RX_D0_P
K	RX1C_P	VSSA	VDDA1P3_TX_SYNTH	VDDA1P3_BB	RESET	SPI_EN	P1_D8/RX_D4_N	P1_D6/RX_D3_N	P1_D4/RX_D2_N	P1_D2/RX_D1_N	P1_D0/RX_D0_N	VSSD
L	RX1C_N	VSSA	VSSA	RBIAS	AUXADC	SPI_DO	VSSA	VSSA	VSSA	VSSA	VSSA	VSSA
M	RX1A_P	RX1A_N	DNC	VSSA	TX_MON1	VSSA	TX1A_P	TX1A_N	TX1B_P	TX1B_N	DNC	XTALN

ANALOG I/O
 DC POWER

DIGITAL I/O
 GROUND

DO NOT CONNECT

Figure 2. Pin Configuration, Top View



10658-002



Software-Defined Radio

- ADALM-Pluto
 - AD9363 RF Agile Transceiver

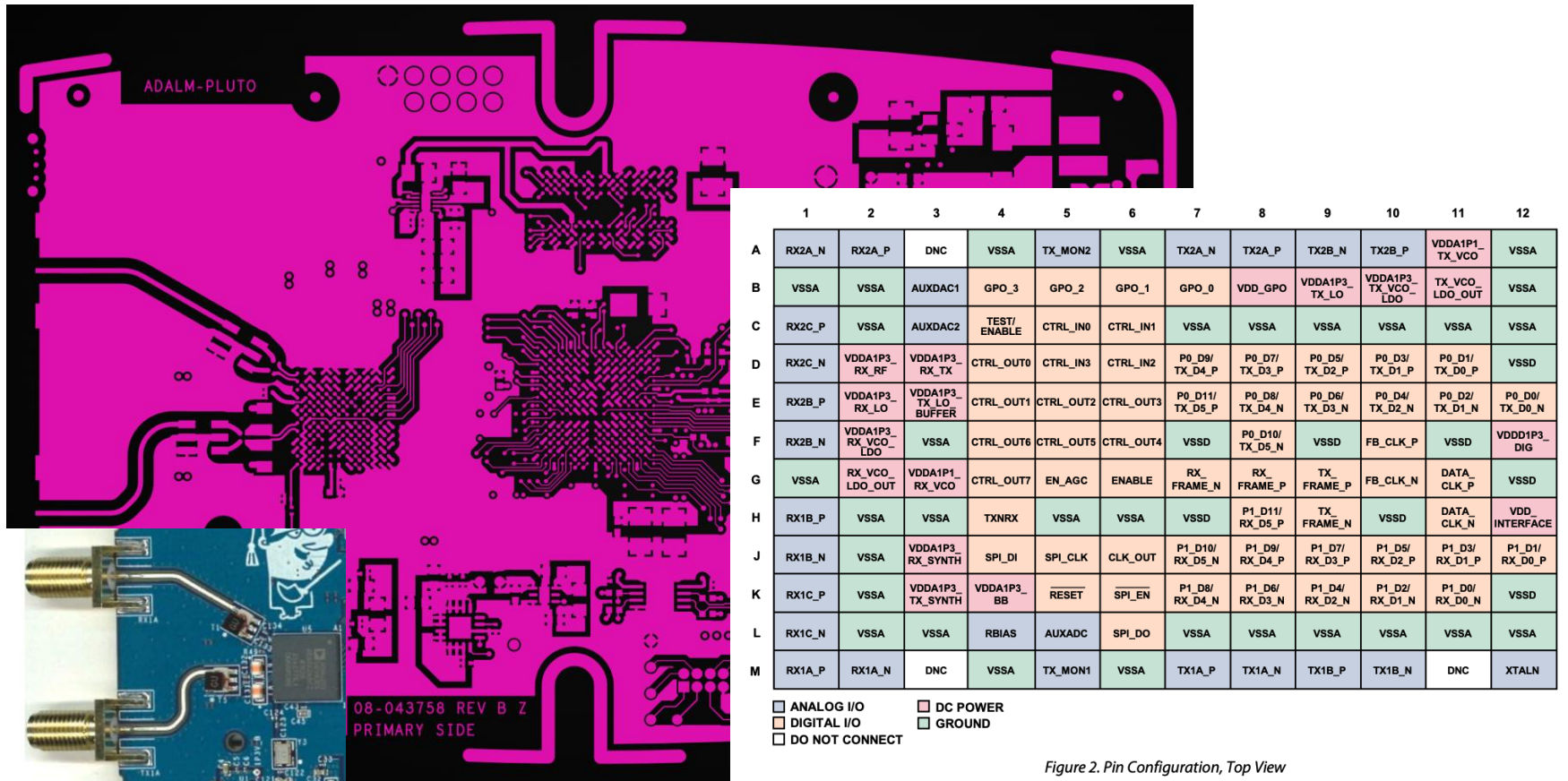


Figure 2. Pin Configuration, Top View

Software-Defined Radio

- Lab Overview

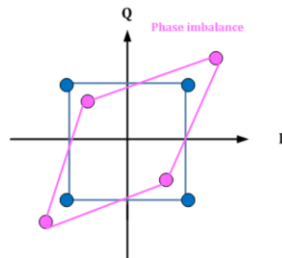
Pluto Basic

- Pyii Basic
- Configuration
- Transmission & Receive
- Understanding Channel



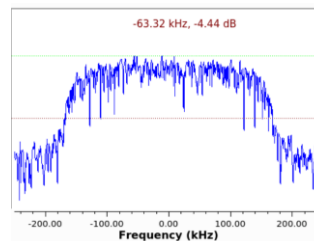
IQ Imbalance

- Quadrature Modulation
- IQ Mismatch Test



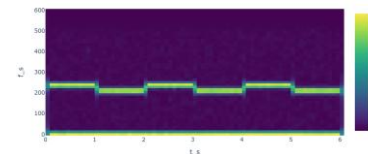
Bandwidth & Aliasing

- Zero-IF Structure
- Carrier Frequency and Bandwidth
- Bandwidth of a modulated signal



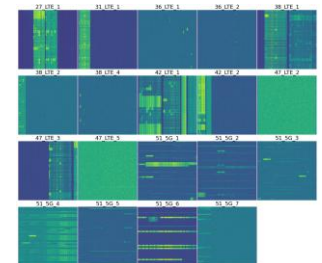
Modulation (ASK, FSK)

- Signal Generation
- Modulation
- Demodulation
- Bitrate (Speed) vs. SNR



Spectrum Monitoring

- Time-Frequency Analysis
- Measurement of Mobile Communication Signals



Thank you

Department of Radio & Information Communications Engineering
CHUNGNAM NATIONAL UNIVERSITY