113-2 電工實驗（通信專題）

Software-Defined Radio (SDR)

April 2, 2025

Table of Contents

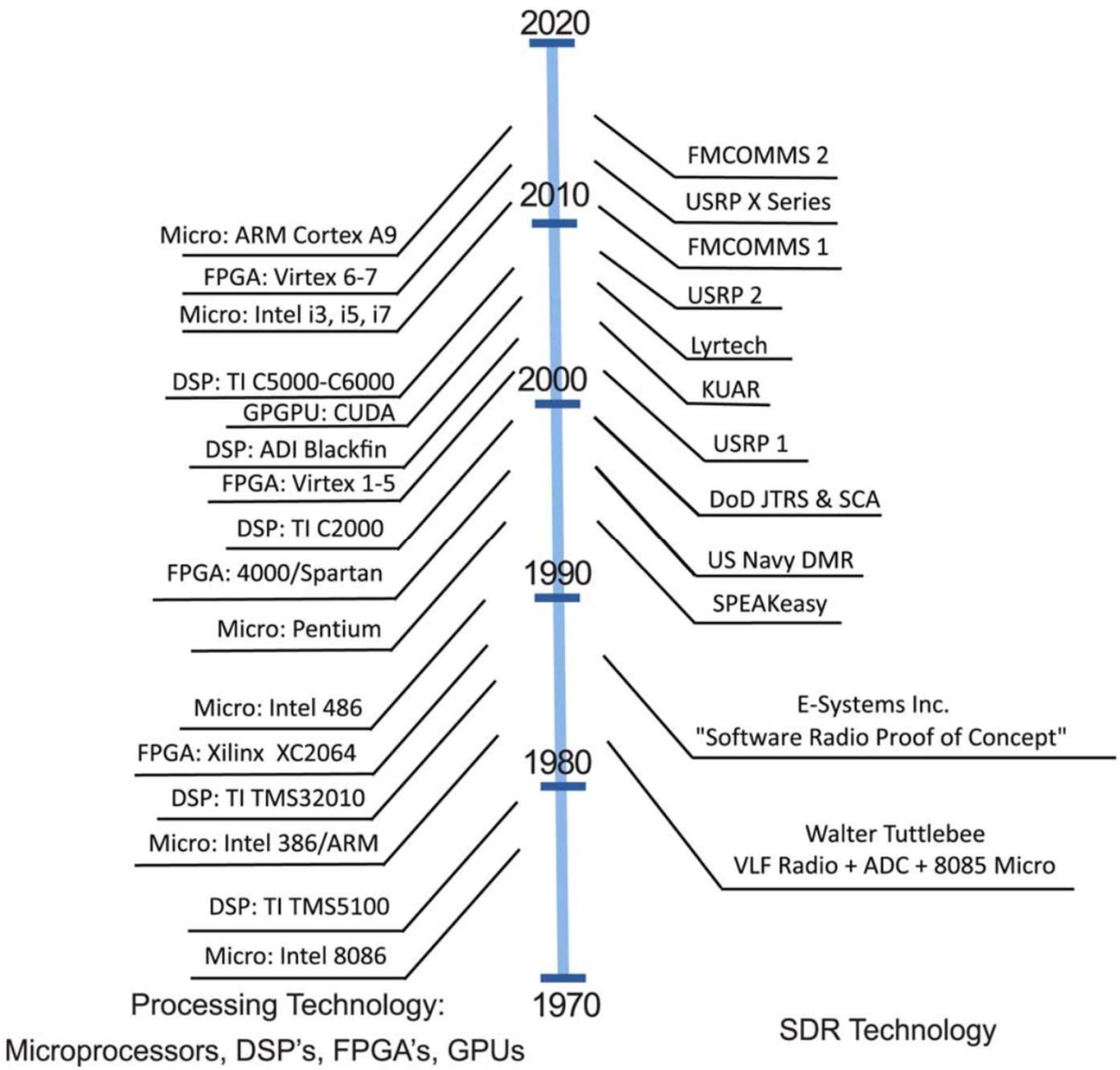
History

- 1950s: wireless systems operated exclusively in the **analog domain**, where communications functions such as modulation and filtering were performed using analog circuits and components

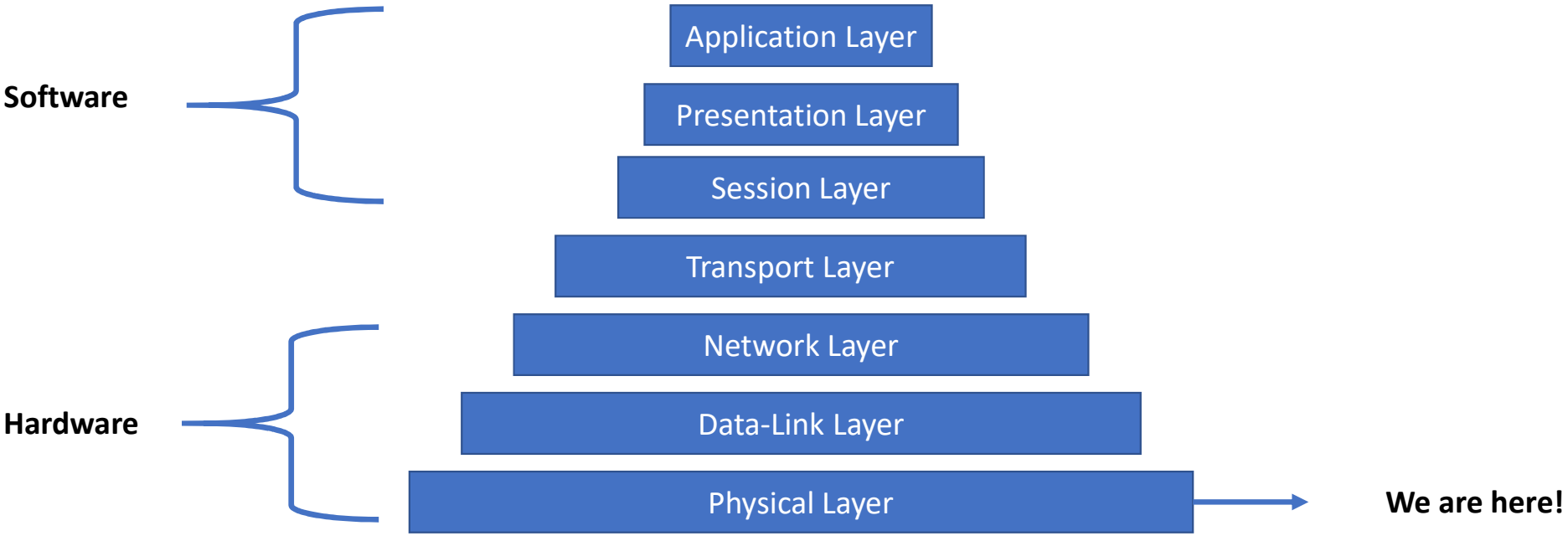
- Rapid evolution of **digital** technology - especially analog-to-digital and digital-to-analog converters
  ➔ perform these same baseband communication functions partially or entirely within the digital domain
  - reducing cost
  - enabling mass production of these transceivers
  - providing a greater flexibility and system functionality

History

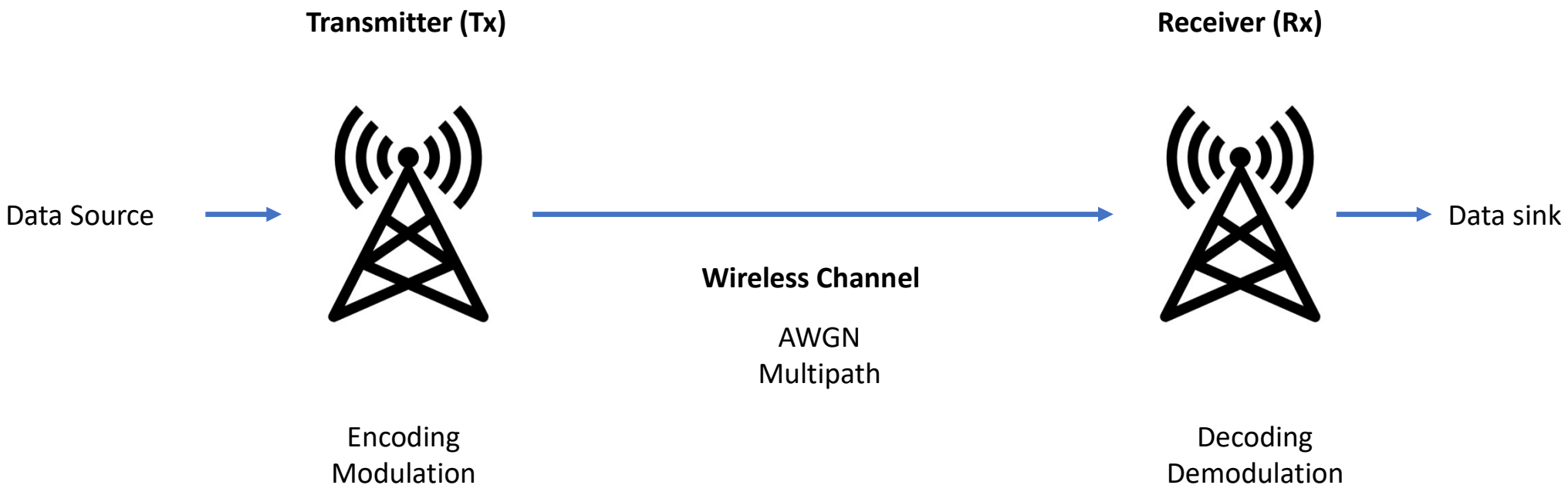
- First: **application-specific integrated circuits (ASICs).**
  - nonprogrammable, static designs
  - cellular telephones, Wi-Fi modems
- 1970s: wireless transceivers that possessed programmable, or software-defined, attributes.
- 1980s: digital baseband radios with programmable features were starting to be prototyped
- 1990s: the first large-scale SDR platforms, SpeakEASY 1 and SpeakEASY 2.
  - An assortment of computing technology used, including multiple DSP platforms and **field programmable gate array** (**FPGA**) technology


2020
2010
2000
1990
1980
1970
FMCOMMS 2
USRP X Series
FMCOMMS 1
USRP 2
Lyrtech
KUAR
USRP 1
DoD JTRS & SCA
US Navy DMR
SPEAKeasy
E-Systems Inc.
"Software Radio Proof of Concept"
Walter Tuttlebee
VLF Radio + ADC + 8085 Micro
Micro: ARM Cortex A9
FPGA: Virtex 6-7
Micro: Intel i3, i5, i7
DSP: TI C5000-C6000
GPGPU: CUDA
DSP: ADI Blackfin
FPGA: Virtex 1-5
DSP: TI C2000
FPGA: 4000/Spartan
Micro: Pentium
Micro: Intel 486
FPGA: Xilinx XC2064
DSP: TI TMS32010
Micro: Intel 386/ARM
DSP: TI TMS5100
Micro: Intel 8086
Processing Technology:
Microprocessors, DSP's, FPGA's, GPUs
SDR Technology

OSI Network Layer

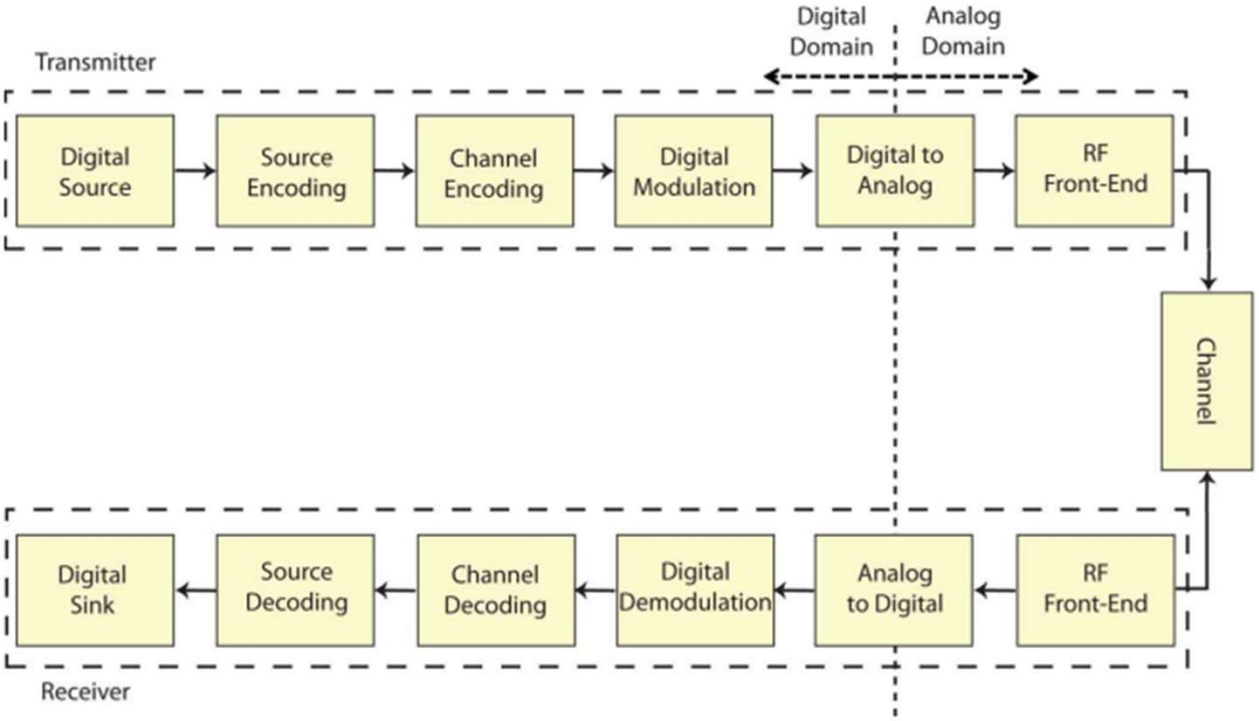


Physical Layer Transmission



Software-Defined Radio (SDR)

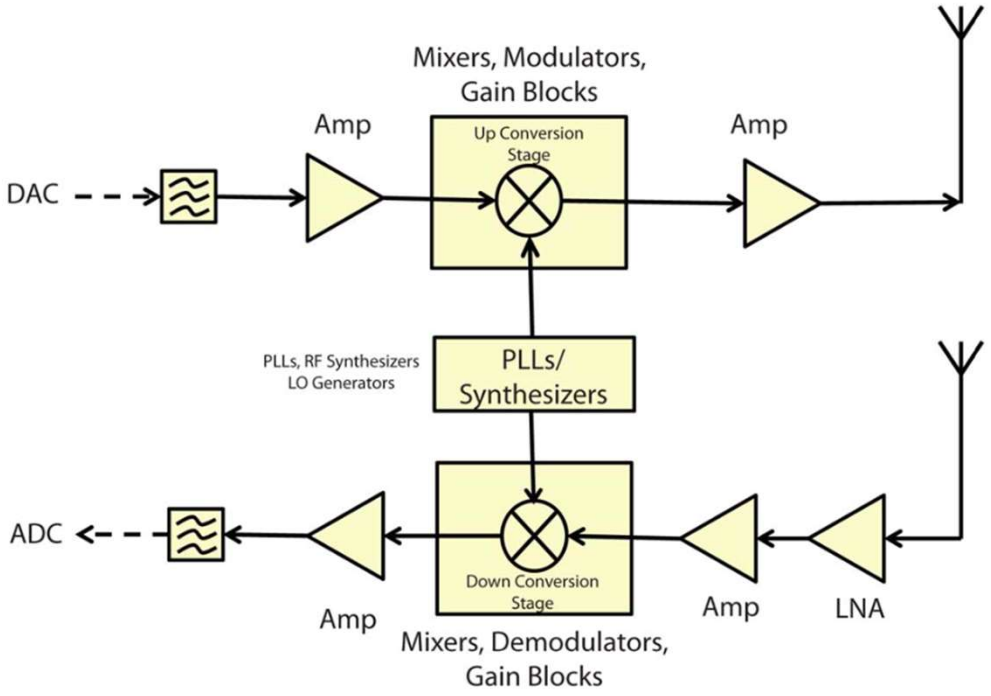
- Implement using FPGAs, GPUs, DSP, GPP



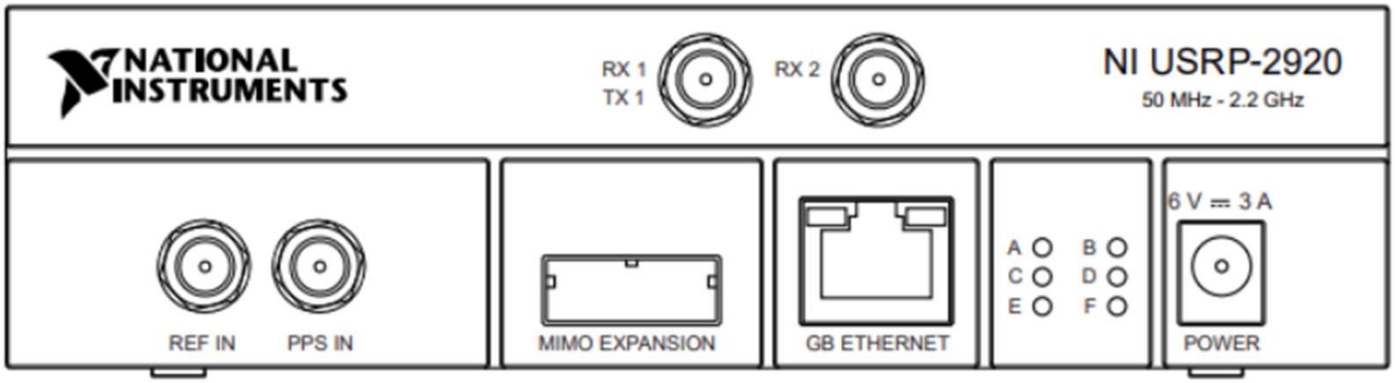
RF Architectures

Up conversion : $y(t) = Re\{x(t)e^{j2 \quad {}_c t}\}$

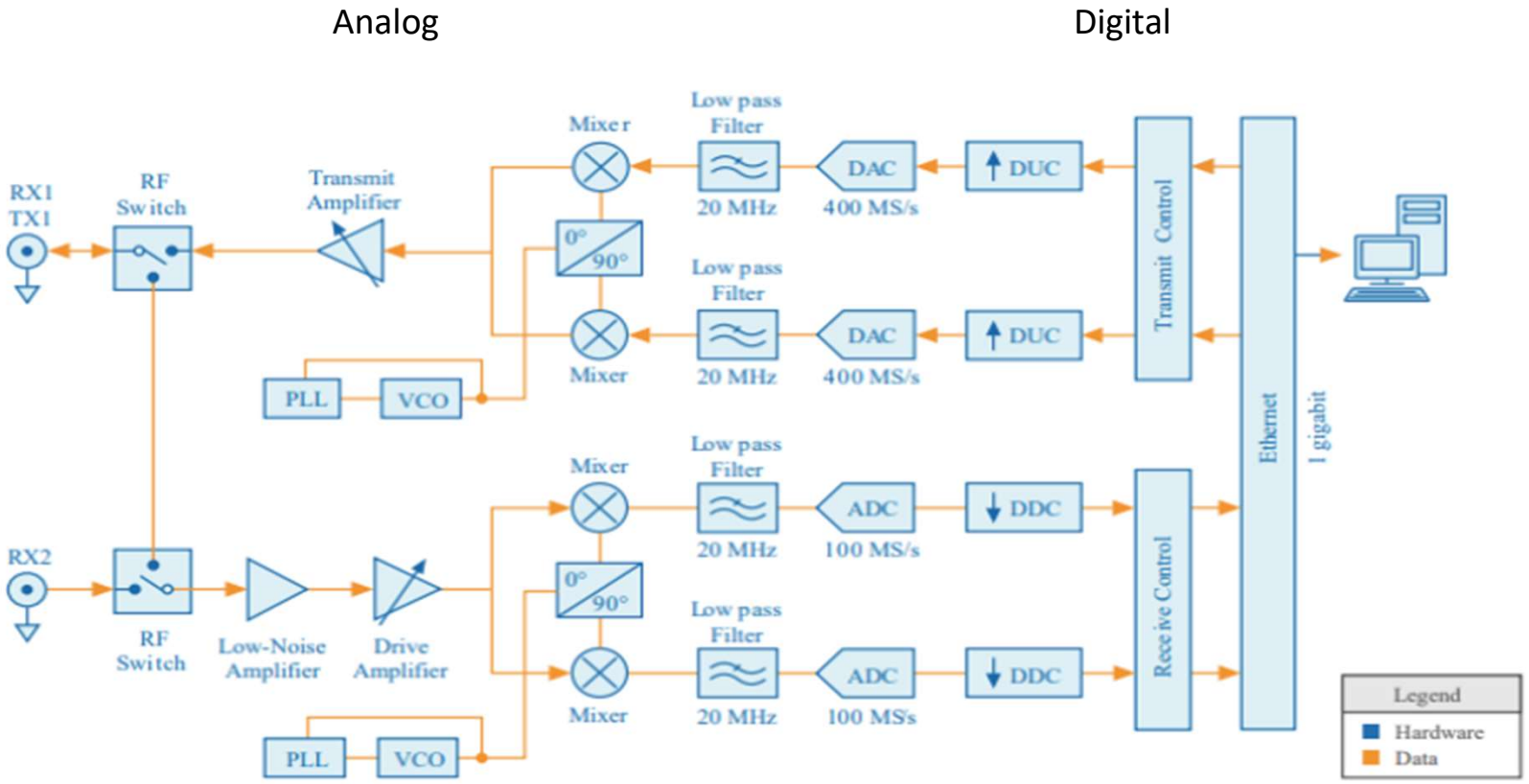
- Generate phase with Phased Lock Loop (PLL)
- Implement up/down conversion with Mixers



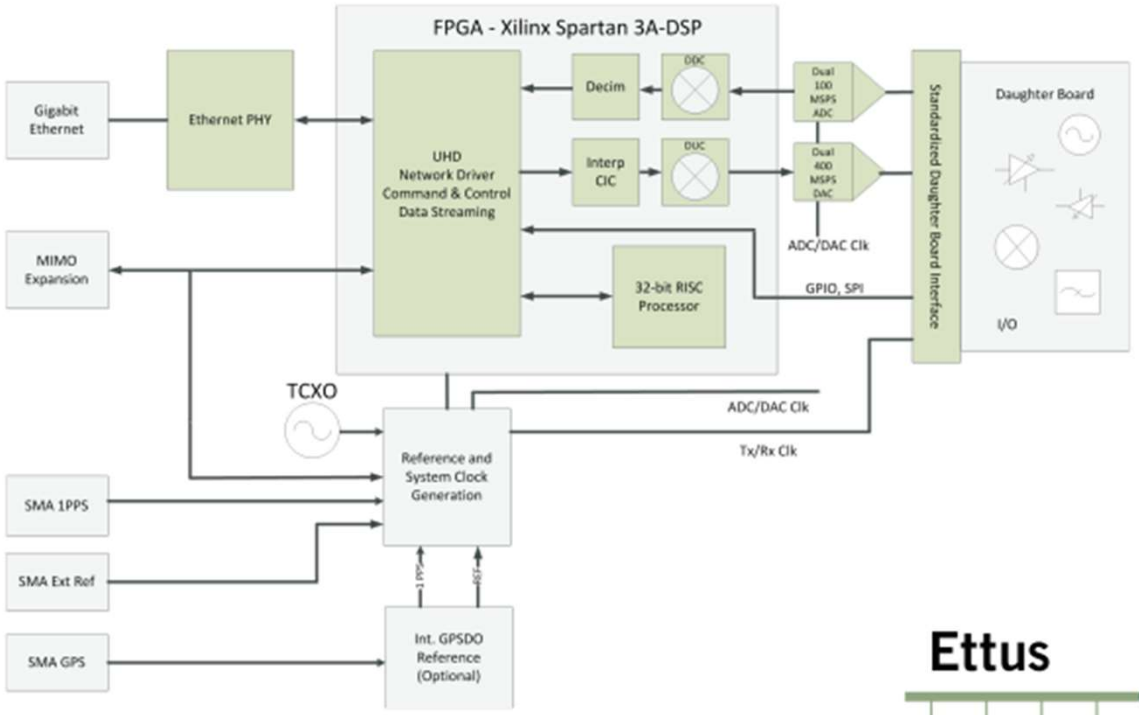
Universal Software Radio Peripheral



Block Diagram of USRP



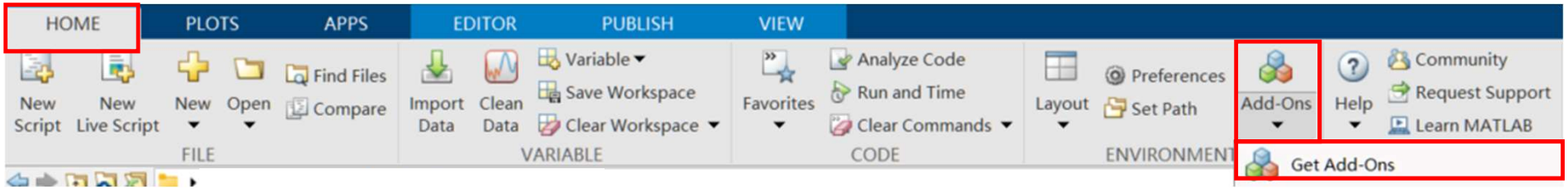
FPGA of USRP System



Ettus Research™
A National Instruments Company

USRP Preparation - Software

- Software : Matlab Communications Toolbox Support Package for USRP Radio



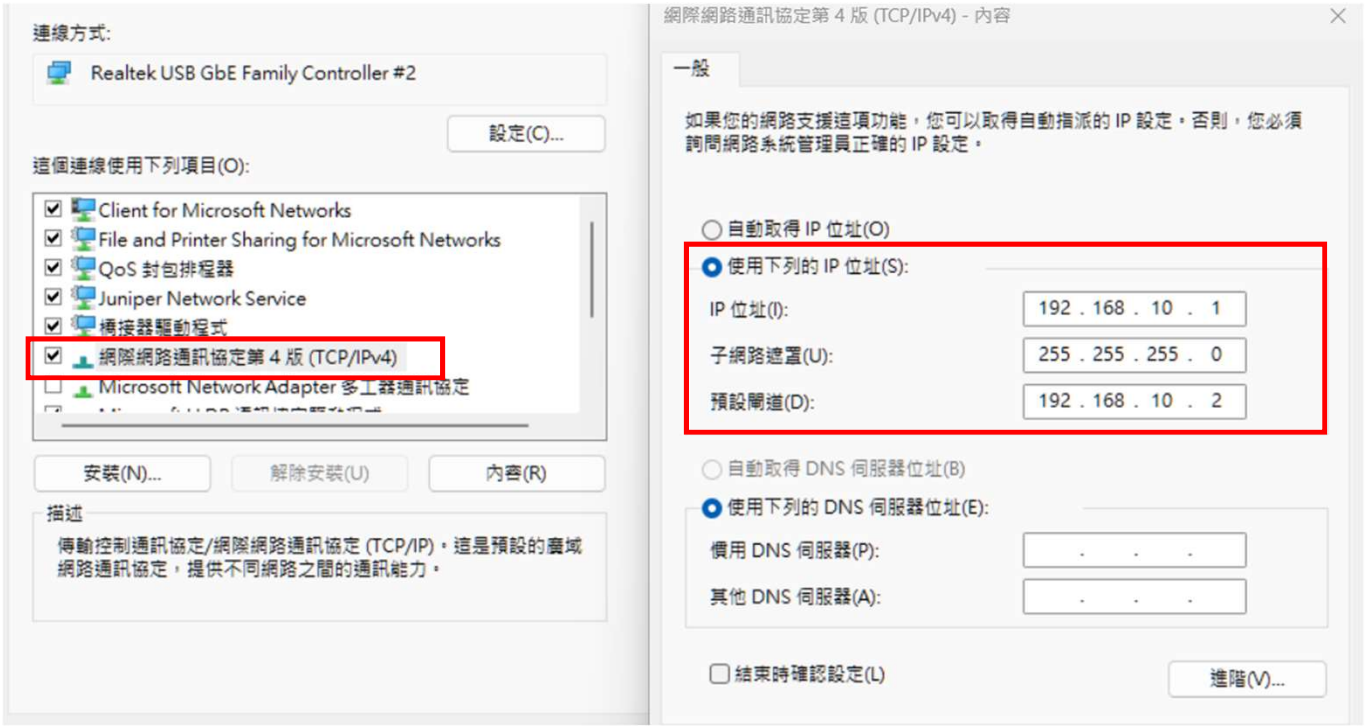
USRP Preparation - Hardware

- One USRP-2920 machine
- One power cable
- Two Antennas
- One gigabyte ethernet cable
- USB to ethernet adapter (if needed)

USRP Preparation – Connection Setting

- We need to configure the internet setting between USRP and computer manually

USRP Preparation – Connection Setting



Hello World in USRP

- Using the command ***findsdru()***
- You should get the following result

```
ans =

  struct with fields:

     Platform: 'N200/N210/USRP2'
    IPAddress: '192.168.10.2'
    SerialNum: '4095'
       Status: 'Success'
```

Reference

Transmit & Receive Scheme

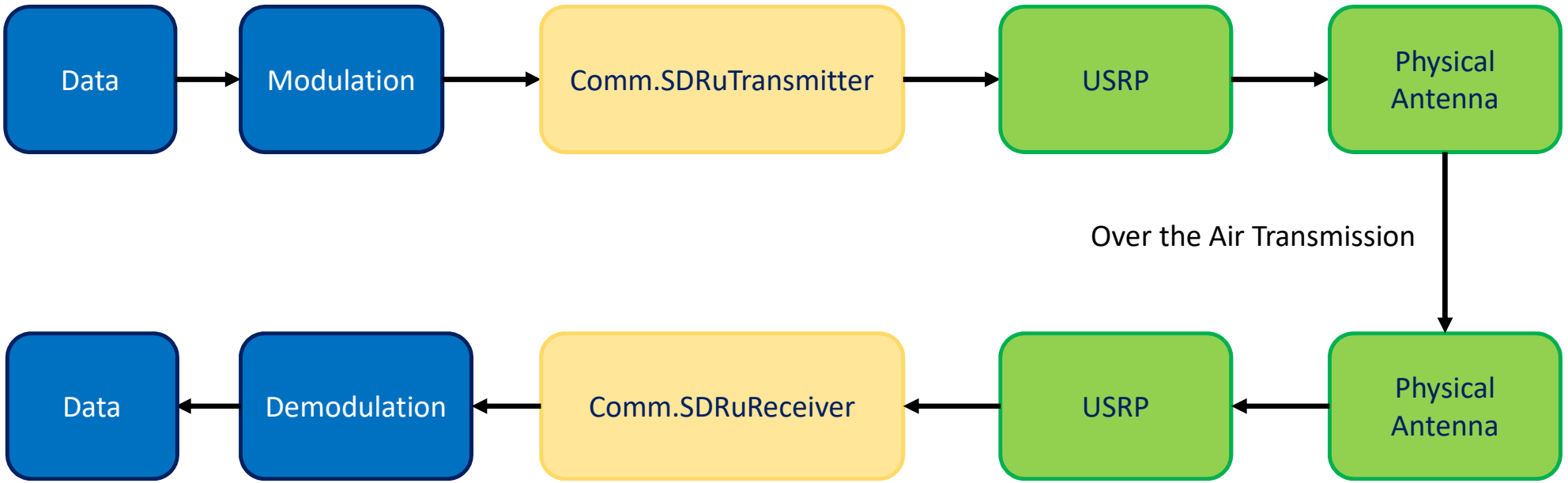
- In Matlab, we communicate with USRP using comm.SDRuTransmitter and comm.SDRuReceiver.

```
radio_Tx = comm.SDRuTransmitter(...
                    'Platform',             platform, ...
                    'IPAddress',            address, ...
                    'CenterFrequency',      USRPCenterFrequency, ...
                    'Gain',                 USRPGain);
```

Transmit & Receive Scheme

```
radio_Rx = comm.SDRuReceiver(...
                'Platform',             platform, ...
                'IPAddress',            address, ...
                'CenterFrequency',      USRPCenterFrequency, ...
                'Gain',                 5, ...
                'SamplesPerFrame',      1e5, ...
                'OutputDataType',       'double') ;
```

USRP Transmission Scheme

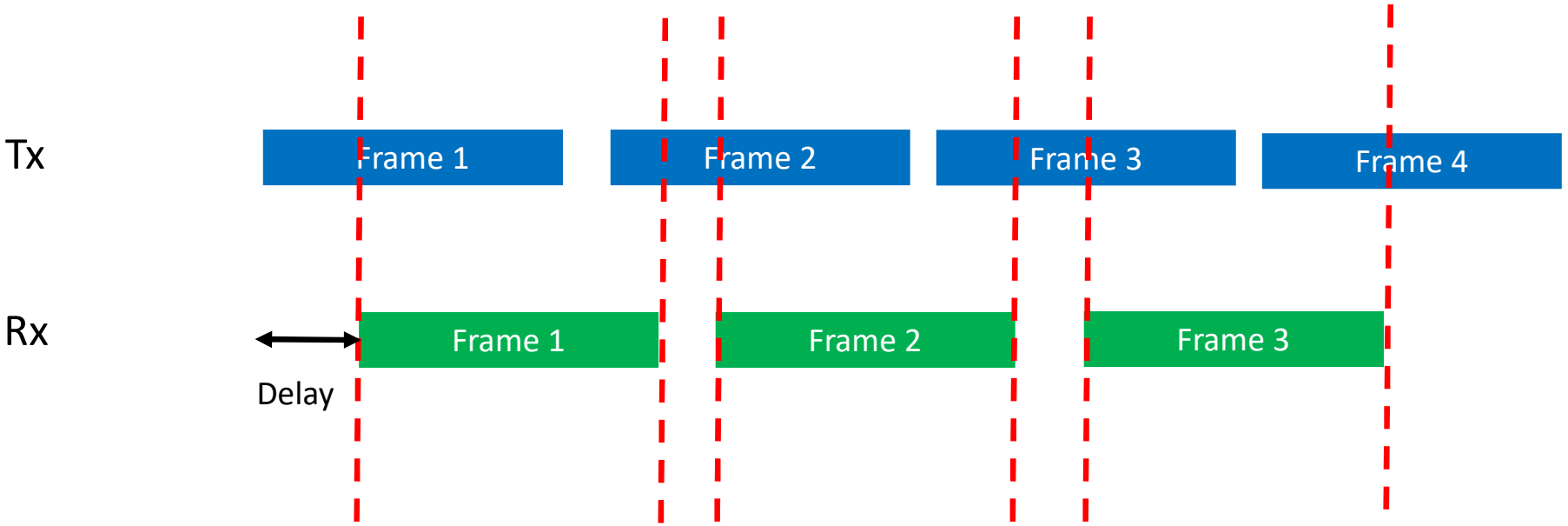


Basic Example - Transmitting a sine wave

- Download the sample code from NTU cool
- You should be aware of
  - Timing of transmission and reception
  - Transmission Delay of USRP

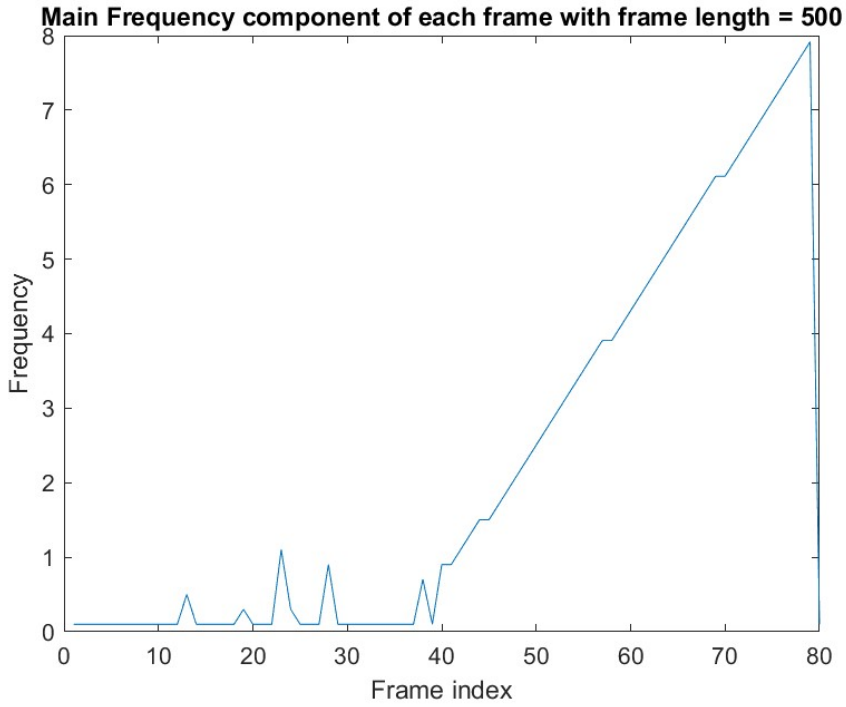
USRP Transmission – Timing

- Timing issue of Tx and Rx



Transient state of USRP

- In the sample code, we transmit sine wave with different frequency for 50 frames
- $f_i = \frac{Frame\ index}{10}$
- We keep reception for 30 frames



Reference

- Machado, Raquel G., and Alexander M. Wyglinski. "Software-defined radio: Bridging the analog–digital divide." *Proceedings of the IEEE* 103, no. 3 (2015): 409-423.