Communication Systems Lab Report

भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

DHANUSH V NAYAK	EE23BTECH11015
KAUSTUBH KHACHANE	EE23BTECH11032

-

Contents

1 | Overview

This comprehensive documentation presents a detailed analysis of multiple software-defined radio (SDR) communication experiments conducted using USRP (Universal Software Radio Peripheral) and Adalm Pluto devices. The projects demonstrate advanced wireless communication techniques, implementing OFDM-based systems and addressing critical physical layer security concerns through the Alice-Bob-Eve security framework. These implementations serve as practical demonstrations of modern communication theory in real-world wireless scenarios.

Project Scope

The documentation focuses on four major implementation projects in the domain of wireless communication and physical layer security. These include:

- OFDM-based communication between dual USRP transceivers
- Multi-antenna USRP communication with dual transmit chains
- Cross-platform communication between ADALM-Pluto and USRP devices
- Physical-layer security implementations mitigating eavesdropping threats

All implementation stages prioritize real-time signal processing, verification through SDR experimentation, and robustness against interference and cyber-attacks.

2 | Project 1: OFDM-Based Communication Between Dual USRP Transceivers

2.1 | Project Objective and Motivation

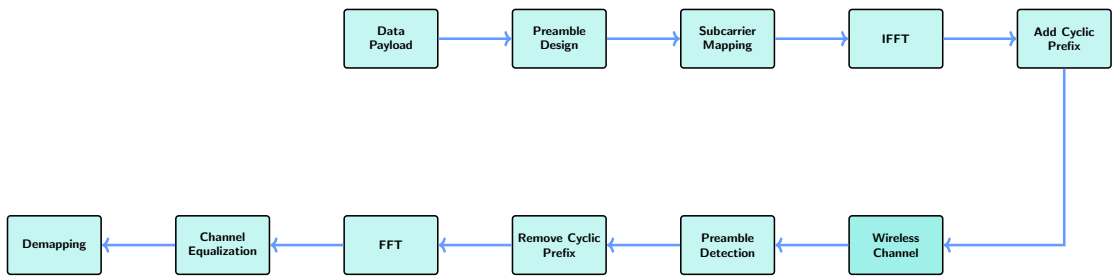
The primary objective of this project was to establish reliable bidirectional communication between two USRP B200/B210 software-defined radios using Orthogonal Frequency Division Multiplexing (OFDM). OFDM represents a cornerstone technology in modern wireless communications, providing robust transmission over frequency-selective fading channels by dividing the channel bandwidth into multiple orthogonal subcarriers. This approach enables high spectral efficiency and resilience to multipath interference.

2.1.1 | System Block Diagram

Key Objectives

1. **System Implementation:** Develop a complete OFDM transceiver architecture using MATLAB/GNU Radio
2. **Channel Characterization:** Measure and analyze real-world channel impulse responses
3. **Performance Evaluation:** Assess bit error rate (BER), packet delivery ratio, and spectral efficiency
4. **Synchronization:** Implement robust timing and frequency synchronization algorithms
5. **Adaptive Modulation:** Explore dynamic subcarrier mapping based on channel conditions

2.2 | Technical Architecture



2.2.1 | OFDM Modulation Process

The OFDM transmission chain encompasses the following sequential operations:

OFDM Modulation Process (Aligned with System Architecture)

1. **Data Payload Generation:** Raw digital information (binary payload) is prepared as the input for OFDM symbol construction.
2. **Preamble Design:** A deterministic synchronization preamble (e.g., short/long training symbols) is generated for packet detection, timing estimation, and coarse frequency correction.
3. **Subcarrier Mapping:** Payload bits are modulation-mapped (BPSK/QPSK/16-QAM/64-QAM) and assigned to specific OFDM subcarriers, leaving guard and DC tones unused.
4. **IFFT Operation:** Frequency-domain symbols across all subcarriers are transformed into a composite time-domain waveform using an N-point IFFT.
5. **Cyclic Prefix Addition:** A prefix of length L_{cp} is copied from the end of the IFFT output and appended to each OFDM symbol to combat multipath-induced ISI.
6. **Wireless Channel Transmission:** The CP-prefixed frame is pulse-shaped, DAC-converted, and transmitted over the wireless channel where fading, noise, and CFO distort the signal.
7. **Preamble Detection (Receiver):** The received stream is scanned for correlation peaks, enabling accurate frame start detection and coarse timing recovery.
8. **Cyclic Prefix Removal:** The detected OFDM symbol start is used to strip off the CP before demodulation.
9. **FFT Demodulation:** An N-point FFT converts the received time-domain OFDM waveform back into frequency-domain subcarrier symbols.
10. **Channel Equalization:** Using preamble-derived channel estimates, each subcarrier is equalized (ZF/MMSE) to reverse channel distortion.
11. **Symbol Demapping:** Equalized complex symbols are mapped back to bits using the inverse constellation mapping rule to reconstruct the original payload.

2.2.2 | Channel Characterization

Channel impulse response (CIR) measurement forms a critical component of system evaluation:

Channel Measurement Procedure

1. **Preamble Transmission:** A known reference sequence (Zadoff-Chu or PAPR-optimized preamble) is transmitted prior to data symbols.
2. **Receiver Correlation:** Received preamble is correlated with the known sequence to extract CIR estimates.
3. **CIR Logging:** Channel tap coefficients and timing estimates are stored for post-analysis.
4. **Statistical Analysis:** Path delay spread, doppler effects, and fading characteristics are computed from CIR measurements.

2.3 | Implementation Details

2.3.1 | Hardware Configuration

System Parameters

Parameter	Configuration
Radio Frequency (RF) Center Frequency	2.4 GHz ISM Band
Channel Bandwidth	10 – 20 MHz
Number of Subcarriers (FFT Size)	64, 128, or 256
Cyclic Prefix Length	16 samples (Guard Interval)
OFDM Symbol Duration	$T_{sym} = \frac{N_{fft}+N_{cp}}{f_s}$
Subcarrier Spacing	$\Delta f = \frac{B_w}{N_{fft}}$
Modulation Schemes	BPSK, QPSK, 16-QAM
Preamble Sequence	Barker Sequence

2.3.2 | Software Architecture

The implementation employs a layered architecture separating physical layer processing from hardware abstraction:

Application Layer: Data generation, performance metrics, result visualization

MAC Layer: Packet framing, preamble insertion, inter-frame spacing

Physical Layer: OFDM modulation/demodulation, equalization, synchronization

Hardware Abstraction: USRP driver interface, transmit/receive buffering

2.4 | Measurement Metrics

Performance Evaluation Criteria

- **Bit Error Rate (BER):** Probability of incorrect bit reception as function of SNR
- **Packet Delivery Ratio (PDR):** Percentage of correctly received packets over total transmitted
- **Channel Capacity:** Shannon capacity computed from measured SNR and bandwidth
- **Equalization Quality:** Residual inter-carrier interference after frequency-domain equalization
- **Synchronization Accuracy:** Timing offset estimation error in samples
- **Spectral Efficiency:** Achieved data rate divided by utilized bandwidth

3 | Project 2: Multi-Antenna USRP Communications with Dual Transmit Chains

3.1 | Motivation and Objectives

The second project extends single-antenna OFDM systems to leverage MIMO (Multiple-Input Multiple-Output) capabilities of USRP hardware. By employing multiple transmit antennas, the system achieves spatial diversity and increased throughput. This implementation addresses practical challenges in antenna array calibration, spatial channel estimation, and precoding design.

MIMO System Advantages

- **Spatial Diversity:** Multiple independent fading paths reduce outage probability
- **Increased Capacity:** Shannon capacity scales linearly with $\min(M_t, M_r)$ (number of transmit/receive antennas)
- **Coding Opportunities:** We can apply convolutional coding as well for error correction receiver.

3.2 | System Architecture

3.2.1 | MIMO-OFDM Signal Model

For a $M_t \times M_r$ MIMO system with OFDM modulation, the received signal at subcarrier k is expressed as:

$$\mathbf{Y}_k = \mathbf{H}_k \mathbf{X}_k + \mathbf{N}_k$$

where:

- $\mathbf{X}_k$ is the $M_t \times 1$ transmit symbol vector
- $\mathbf{H}_k$ is the $M_r \times M_t$ channel matrix at subcarrier k
- $\mathbf{N}_k$ is additive white Gaussian noise

3.2.2 | Hardware Configuration

Dual TX USRP Configuration

Component	Specification
Number of Transmit Chains	2 (TX1, TX2 connectors)
Number of Receive Chains	2 or more (RX2, RX3 connectors)
TX/RX Isolation	Synchronized via timing card
Antenna Configuration	Linear array or spaced dipoles
Antenna Spacing	$\lambda/2$ to λ at operating frequency
Synchronization Method	PPS (Pulse Per Second) + 10 MHz reference

4 | Project 3: Cross-Platform Communication between Adalm Pluto and USRP

4.1 | Project Overview

This project addresses interoperability challenges when integrating devices from different manufacturers (Analog Devices Adalm Pluto and Ettus Research USRP). Such cross-platform systems are increasingly important in networked radio scenarios, cognitive radio systems, and heterogeneous SDR testbeds.

Integration Challenges

1. **Hardware Incompatibility:** Different RF front-ends, ADC/DAC resolutions, and sampling rates
2. **Timing Misalignment:** Independent clock sources causing frequency and phase offsets
3. **Software Stack Differences:** Distinct APIs (IIO framework vs. UHD library)
4. **Channel Mismatch:** Different noise figures and gain ranges

4.2 | Technical Approach

4.2.1 | Device Specifications

Hardware Comparison

Parameter	Adalm Pluto	USRP B210/200
RF Range	70 MHz – 6 GHz	50 MHz – 2.2 GHz
ADC/DAC Resolution	12-bit	14-bit
Bandwidth	20 MHz (fixed)	Tunable: 5 – 40 MHz
RX Noise Figure	∼7 dB	∼5 dB
TX Power	+6 dBm max	+16 dBm max
Interface	USB 2.0	USB 3.0
Cost	25,000	0.3 Million

4.2.2 | Synchronization Architecture

Cross-platform operation requires careful synchronization of multiple parameters:

Synchronization Hierarchy:

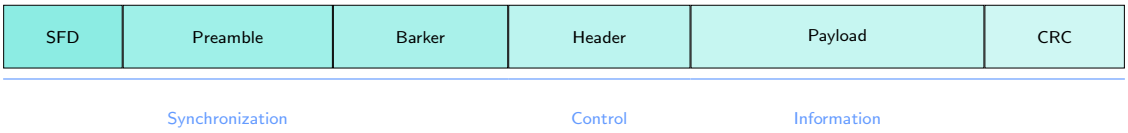
Frequency Sync: Master oscillator reference to both devices via distributed oscillator

Time Sync: GPS-disciplined oscillator or network time protocol synchronization

Frame Sync: Beacon signal with known preamble for frame-level alignment

Fine Freq Sync: Automatic frequency control loop to track residual offsets

4.2.3 | Frame Structure



where:

- **SFD:** Start Frame Delimiter for packet boundary detection
- **Preamble:** Training pattern for AGC settling and coarse sync
- **Barker:** Barker Sequence for accurate channel estimation
- **Header:** Contains length, modulation type, flags, and MAC info
- **Payload:** OFDM data symbols carrying user information
- **CRC:** Cyclic Redundancy Check for error detection

5 | Project 4: Physical Layer Security – Alice-Bob-Eve Framework

5.1 | Security Model Overview

Physical layer security (PLS) represents a fundamental approach to securing wireless communications by exploiting properties of the propagation channel itself. The Alice-Bob-Eve scenario models an eavesdropping threat where an adversary (Eve) attempts to intercept communications between authorized parties (Alice and Bob). Our implementation addresses key security challenges at the modulation, coding, and channel exploitation levels.

Information Theoretic Foundation

The secrecy capacity of a wiretap channel is defined as:

$$C_s = \max_{p(x)} [I(X;Y) - I(X;Z)]$$

where Y is Bob's received signal, Z is Eve's received signal, and $I(\cdot;\cdot)$ denotes mutual information. Positive secrecy capacity requires Bob's channel to be superior to Eve's, either through better SNR or by some artificial method of induction.

5.2 | Attack and Defense Scenarios

5.2.1 | Eavesdropping Attacks

Eve's Attack Strategies

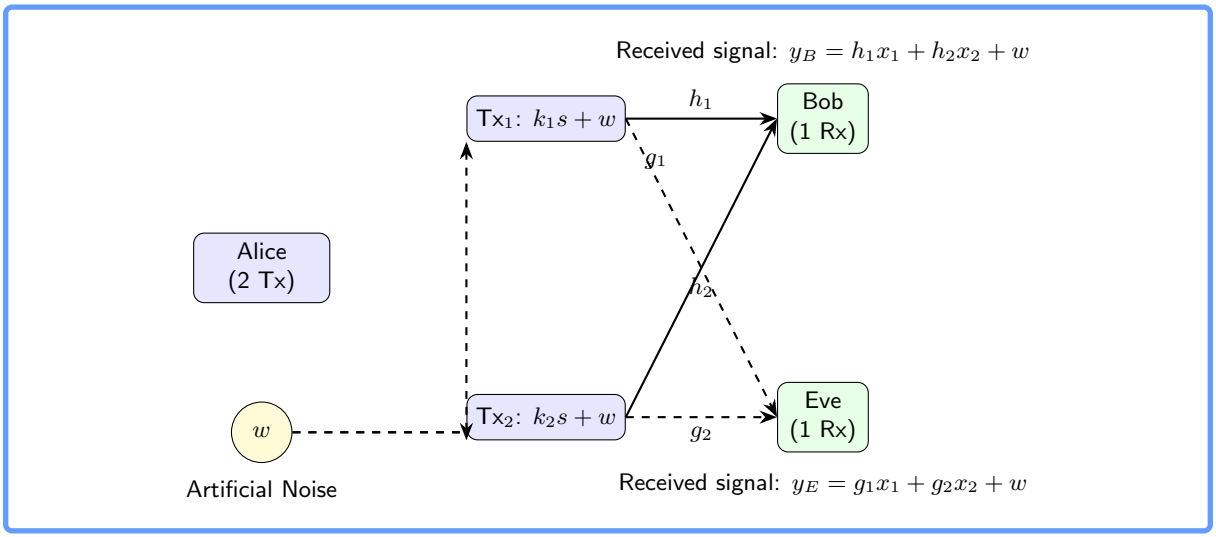
1. **Passive Eavesdropping:** Direct reception and decoding of transmitted signals without modification
2. **Active Eavesdropping:** Transmission of pilot signals to facilitate channel estimation
3. **Brute Force Decoding:** Attempting multiple decoding hypotheses with computational resources
4. **Signal Intelligence (SIGINT):** Statistical analysis of signal properties for feature extraction

5.2.2 | Defense Mechanisms

Implemented Security Technique: Artificial Noise Injection

- Artificial Noise (AN) is intentionally added to the transmitted signal to degrade the channel quality at potential eavesdroppers.
- The noise is designed such that it minimally impacts the intended receiver, ensuring confidential communication performance remains stable.
- In multi-antenna systems, AN is transmitted in the null space of the legitimate receiver's channel, making it invisible to the intended user.
- The method increases secrecy capacity by exploiting channel asymmetry between the legitimate receiver and the eavesdropper.
- This technique is effective in scenarios where the transmitter has partial or full knowledge of the channel state information (CSI).

5.3 | System Implementation



6 | Organisation

6.1 | Common Framework

All four projects share a unified signal processing framework implemented across multiple components:

Cross-Project Common Elements

- **Modulation Base:** OFDM with configurable subcarrier count and modulation schemes
- **Synchronization:** Timing offset detection via preamble correlation
- **Channel Estimation:** Least-squares or MMSE channel estimation on training symbols
- **Equalization:** Single-tap frequency-domain equalization per subcarrier
- **Coding:** Convolutional codes with Viterbi decoding
- **Logging Framework:** Centralized data acquisition for performance analysis

6.2 | File Organization and Data Flow

The implementation organizes code across project folders with specialized modules:

Directory Structure

Global_Parameters_PLS.m : Master configuration file (center frequency, gains, packet structure)

OFDM_TX_X.m : OFDM transmitter implementation for node X (Alice/Bob/Eve)

OFDM_RX_X.m : OFDM receiver with equalization and synchronization

Hardware_TX_X.m : USRP/Pluto hardware interface (DAC buffering, gain control)

Hardware_RX_X.m : Hardware receiver interface (ADC streaming, data logging)

corr_code.m : Preamble correlation for timing synchronization

oversamp.m : Oversampling and pulse shaping filters

setstate0_TX/RX.m : USRP initialization and state reset

7 | Development Methodology and Lessons Learned

7.1 | Iterative Development Process

Project Development Cycle:

1. **Simulation:** Algorithm development and validation in MATLAB/Simulink
2. **Hardware Integration:** Porting to USRP drivers and real-time constraints
3. **Bench Testing:** Validation with test signals in RF-isolated environment
4. **Over-the-Air Testing:** Evaluation in actual propagation conditions
5. **Performance Analysis:** Statistical evaluation of measured data
6. **Optimization:** Parameter tuning based on empirical results

7.2 | Key Technical Challenges and Solutions

Key Security Metrics

Timing Misalignment: Implemented coarse/fine synchronization with preamble correlation and feedback loops

Frequency Offsets: Developed pilot-based frequency correction with loop bandwidth optimization

Channel Fading: Applied equalization

Cross-Device Sync: Distributed reference clocks and GPS time tagging

Real-Time Performance: Optimized code for USRP's real-time