Near Real-Time Zigbee Device Discrimination Using CB-DNA Features

THESIS

Yousuke Z. Matsui, Captain, USAF

AFIT-ENG-MS-20-M-043

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

DISTRIBUTION STATEMENT A
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views expressed in this document are those of the author and do not reflect the official policy or position of the United States Air Force, the United States Department of Defense or the United States Government. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

AFIT-ENG-MS-20-M-043

Near Real-Time Zigbee Device Discrimination Using CB-DNA Features

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Electrical Engineering

Yousuke Z. Matsui, B.S.E.E.

Captain, USAF

March 26, 2020

DISTRIBUTION STATEMENT A
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AFIT-ENG-MS-20-M-043

Near Real-Time Zigbee Device Discrimination Using CB-DNA Features

THESIS

Yousuke Z. Matsui, B.S.E.E.
Captain, USAF

Committee Membership:

Maj. J. Addison Betances, Ph.D
Chair

Dr. Timothy H. Lacey, Ph.D
Member

Dr. Michael A. Temple, Ph.D
Member

Abstract

Currently, Low-Rate Wireless Personal Area Networks (LR-WPAN) based on the Institute of Electrical and Electronics Engineers (IEEE) 802.15.4 standard are at risk due to open-source tools which allow bad actors to exploit unauthorized network access through various cyberattacks by falsifying bit-level credentials. This research investigates implementing a Radio Frequency (RF) air monitor to perform Near Real-Time (NRT) discrimination of Zigbee devices using the IEEE 802.15.4 standard. The air monitor employed a Multiple Discriminant Analysis/Euclidean Distance classifier to discriminate Zigbee devices based upon Constellation-Based Distinct Native Attribute (CB-DNA) fingerprints. Through the use of CB-DNA fingerprints, Physical Layer (PHY) characteristics unique to each Zigbee device strengthen the native bit-level authentication process for LR-WPAN networks.

Overall, the developed RF air monitor achieved an Average Cross-Class Percent Correct Classification of $\%C_{tst} = 99.24\%$ during the testing of $N_{cls} = 5$ like-model BladeRF Software Defined Radios transmitting Zigbee protocol bursts. Additionally, to evaluate the NRT capability of the air monitor, a statistical analysis of $N_{timing} = 1000$ Zigbee bursts determined the worst-case average runtime from burst detection to classification. The analysis concluded that the runtime was $t_{runtime} \approx 269$ mSec. Ultimately, this research found that PHY characteristics provide an additional method of authentication NRT to enhance the inherent network security for Zigbee applications from cyberattacks.

AFIT-ENG-MS-20-M-043

To My Wife and Kids.

Acknowledgements

First and foremost, I would like to thank my wife and children for their unconditional love and support through this process. I would also like to express my sincere appreciation to Major Addison Betances for his advice and constant mentoring during my time at AFIT.

Yousuke Z. Matsui

Table of Contents

Page

List of Figures

List of Tables

Near Real-Time Zigbee Device Discrimination Using CB-DNA Features

I. Introduction

The Institute of Electrical and Electronics Engineers (IEEE) 802.15.4 standard defines the Physical Layer (PHY) and Medium Access Control (MAC) for Low-Rate Wireless Personal Area Networks (LR-WPAN). Projections currently estimate the annual shipment of 1 billion IEEE 802.15.4 standard devices by 2024 [1]. Specifically, Zigbee is a IEEE 802.15.4 based protocol that has become a popular alternative to Bluetooth and Wi-Fi due to its low-cost and low-power requirements. Zigbee has become widespread in Industrial Control Systems (ICS), home automation, and remote monitoring functions. These applications result in Zigbee having a large cyber-physical footprint.

However, with cost and simplicity as the main drivers for Zigbee, network security has suffered. Zigbee is prone to several different cyberattacks ranging from network key sniffing, device spoofing, and simple replay attacks. Open-source tools, such as Killerbee [2], currently exist that implement all of the above cyberattacks for Zigbee and other IEEE 802.15.4 based protocols. The impact of these attacks spans from loss of sensitive network information to, worst-case, the malicious modification of physical systems.

Outlined in the 2018 Department of Defense (DoD) Cyber Strategy, one of the DoD's five objectives is to "Defend U.S. critical infrastructure from malicious cyber activity that alone, or as part of a campaign, could cause a significant cyber incident" [3]. Therefore, the DoD is directly interested in methods that further enhance cyberspace security for U.S. ICS applications.

1.1 Research Objectives

The objective of this research is to supplement the current Zigbee authentication process by monitoring network traffic and calculating fingerprints (statistical features) based on the transmitting device's PHY characteristics. This Radio Frequency (RF) air monitor is to perform Near Real-Time (NRT) discrimination of transmitting Zigbee devices based on the computed fingerprints. The output from the air monitor provides classification information, which could augment the standard bit-level device authentication process.

1.2 Research Contributions

This research extends from several previous Zigbee device discrimination efforts to achieve results not previously obtained. The significant outcomes of prior research endeavors are presented in Chapter II. Specifically, this research determines the feasibility of fielding a RF air monitor to perform NRT device discrimination using Constellation-Based Distinct Native Attribute (CB-DNA) fingerprints.

1.3 Thesis Organization

Document organization for subsequent chapters is in the following manner:

- Chapter II - Background and Literature Review: Provides an overview of the Zigbee protocol, Distinct Native Attribute (DNA) techniques, classifiers, and the required hardware and software tools for experimentation.
- Chapter III - Methodology: Describes the experimental design process for the RF air monitor and the approaches utilized when conducting training and testing scenarios.

- Chapter IV - Results and Analysis: Presents the RF air monitor's device discrimination performance for Zigbee protocol devices. This chapter also performs a runtime analysis to asses the NRT aspect of the RF air monitor.

- Chapter V - Conclusions: Provides a synopsis of the collected results and recommendations for future research endeavors.

II. Background and Literature Review

2.1 Overview

This chapter provides technical background on the methods used to conduct this research. Section 2.2 provides an overview of the Zigbee communication protocol, and Section 2.3 discusses the Offset-Quadrature Phase Shift Keying (O-QPSK) modulation scheme. Section 2.4 addresses both the Radio Frequency Distinct Native Attribute (RF-DNA) methods used to develop statistical features for received signals using time-domain traits, as well as the Constellation-Based Distinct Native Attribute (CB-DNA) techniques used to derive statistical features for constellation-based modulation schemes. Section 2.5 discusses the use of Multiple Discriminant Analysis (MDA) generated models in conjunction with Euclidean Distance (ED) to perform classification of received fingerprints. Finally, Sections 2.6 - 2.7 describe the various hardware and software tools employed throughout the research.

2.2 Zigbee Wireless Protocol

Zigbee is a wireless communication protocol based upon the Institute of Electrical and Electronics Engineers (IEEE) 802.15.4 standard designed for low data rate, short-range wireless networks [4]. Zigbee devices are often utilized as the communication link for sensors that provide the overall system a state-of-health status for various sub-systems. In real-world applications, these sub-systems include monitoring the structural health of buildings [4], the status of medical patients [5], home automation, and Industrial Control Systems (ICS) applications. Zigbee has become popular for these operations due to its low-power budget and the ever-decreasing entrance barrier to establishing a Wireless Personal Area Network (WPAN).

Zigbee has the ability to operate in three different frequency bands: $f_{B1} = 868$

MHz, $f_{B2} = 915$ MHz, and $f_{B3} = 2.4$ GHz. The three frequency bands combined provide $N_{\text{channels}} = 27$ distinct channels. The $f_{B3} = 2.4$ GHz band is the most common frequency band and is utilized worldwide. In contrast, the two remaining sub-gigahertz bands are typically employed only in specific regions of the world ($f_{B1} =$ 868 MHz in Europe and $f_{B2} = 915$ MHz in North America). This research focuses on the $f_{B3} = 2.4$ GHz frequency band since it is the only band implemented globally. The $f_{B3} = 2.4$ GHz band for Zigbee devices consists of $N_{B3\text{ channels}} = 16$ channels. Each channel has a bandwidth of $B_{\text{channel}} = 2$ MHz, and the center frequency of each channel is separated by $f_{\text{ch separate}} = 5$ MHz. An illustration of the Zigbee protocol spectrum can be seen in Figure 1 [6]. Ideally, prior to implementation, the spectrum utilization is evaluated and the optimal channel is chosen for the communication application. The equation for determining the center frequency ($f_{c_{\text{channel}}}$) for Zigbee channels 11-26 is

$$f_{c_{\text{channel}}} = 2405 + 5(k - 11) \tag{1}$$

where k is the desired channel number such that $k = [11, 12, ..., 26]$. To transmit a binary message using the Zigbee protocol, the original binary message is first parsed into $N_{\text{bits}} = 4$-bit segments. This nibble of bits corresponds to one of $N_{\text{Data Symbols}} =$ 16 different Zigbee Data Symbols (DSs). Each DS is then mapped to a $N_{\text{chips}} = 32$-bit

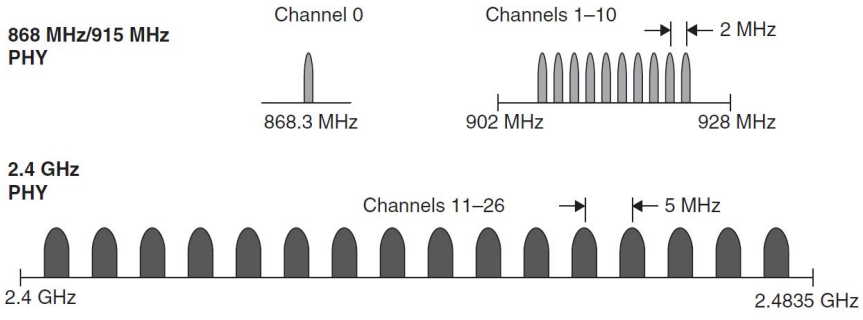

Figure 1: Zigbee frequency bands with $N_{channels} = 27$ channels [6]

nearly-orthogonal chip sequence as shown in Table 1. Next, the resulting chips are modulated using O-QPSK. O-QPSK is a variation of Quadrature Phase Shift Keying (QPSK) which consists of four communication symbols (0, 1, 2, and 3). For the Zigbee protocol, the O-QPSK symbol rate is $f_{\text{symbol rate}} = 1 \times 10^6$ O-QPSK symbols per second. The overall process can be seen in Figure 2 [7].

Per the IEEE 802.15.4 standard, every Zigbee message consists of a Sychronization Header Region (SHR), Physical Layer (PHY) Header Region (PHR), and a PHY Service Data Unit (PSDU) to create the overall PHY Protocol Data Unit (PPDU).

Table 1: Zigbee symbol to chip mapping for $f_{B3} = 2.4$ GHz frequency band

Symbol	Chip Values $(c_0, c_1, ..., c_{31})$
0	11011001110000110101001000101110
1	11101101100111000011010100100010
2	00101110110110011100001101010010
3	00100010111011011001110000110101
4	01010010000101110110110011100011
5	00110101001000101110110110011100
6	11000011010100100010111011011001
7	10011100001101010010001011101101
8	10001100100101100000011101111011
9	10111000110010010110000001110111
10	01111011100011001001011000000111
11	01110111011100011001001011000000
12	00000111011110111000110010010110
13	01100000011101111011100011001001
14	10010110000001110111101110001100
15	11001001011000000111011110111000

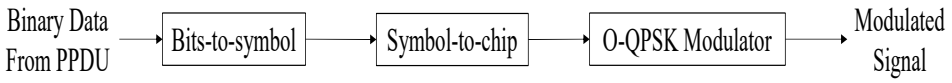

Figure 2: Zigbee signal generation flowchart [7]

The SHR portion of the signal is used to synchronize with a data stream and consists of a preamble (eight consecutive Zigbee DS zeros) followed by a Start of Frame Delimiter (SFD) (Zigbee DSs 7 and 10). The PHR contains the frame length of the PSDU in bytes, and the PSDU contains the type of message transmitted along with the actual payload data. The PSDU can assume the form of $N_{\text{formats}} = 4$ different Medium Access Control (MAC) frame formats associated with a data frame, acknowledgment frame, command frame, and beacon frame. Figures 3 - 6 shows the diagram for each of the four PPDU formats.

Given that the PHR is composed of one byte, where one bit is reserved, the maximum value that can be represented by the PHR is $N_{\text{PHR max}} = 127$. The largest PSDU can contain $DS_{Z_{\max}} = 127 \text{ bytes} \times 2 \frac{\text{DS}_Z}{\text{byte}} = 254$ Zigbee DSs. Thus, the

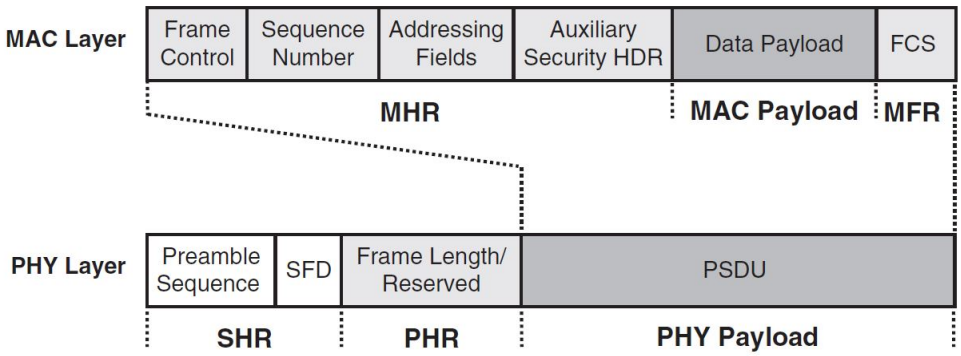

Figure 3: Zigbee PPDU data frame format [4]

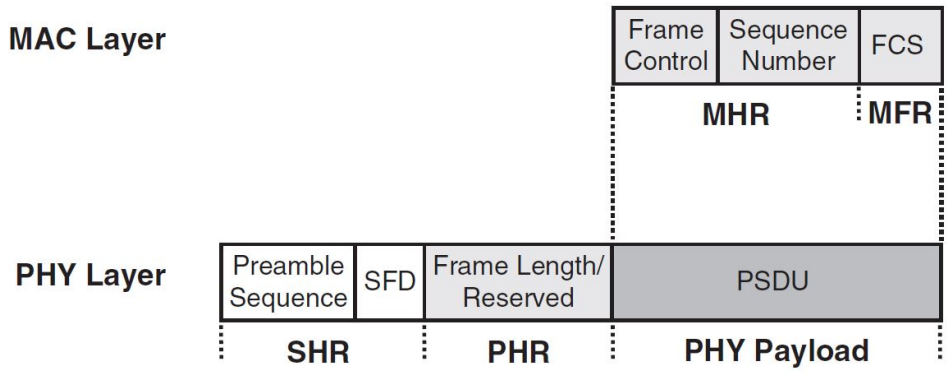

Figure 4: Zigbee PPDU acknowledgement frame format [4]

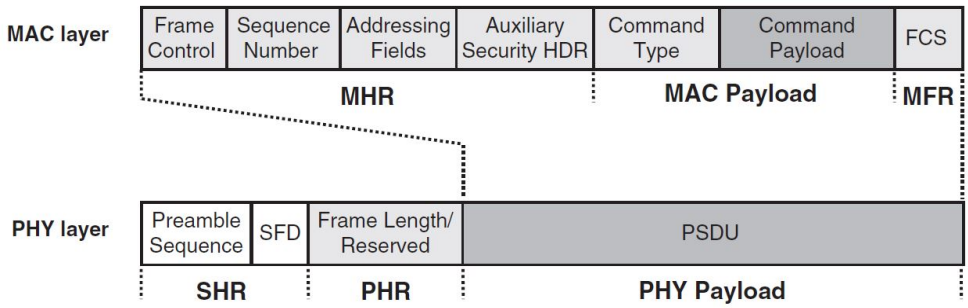

Figure 5: Zigbee PPDU command frame format [4]

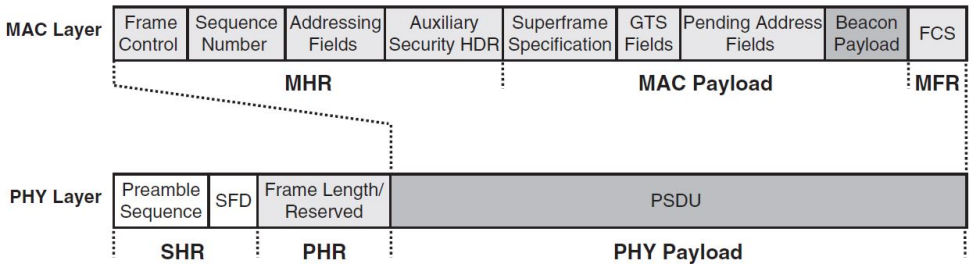

Figure 6: Zigbee PPDU beacon frame format [4]

maximum duration of a Zigbee burst is

$$t_{\max} = \left(\frac{10\ \mathrm{DS}_Z}{\mathrm{SHR}} + \frac{2\ \mathrm{DS}_Z}{\mathrm{PHR}} + \frac{254\ \mathrm{DS}_Z}{\mathrm{PSDU}}\right)\left(\frac{32\ \mathrm{chips}}{\mathrm{DS}_Z}\right)\left(\frac{\mathrm{CS}_O}{2\ \mathrm{chips}}\right)\left(\frac{\mathrm{Sec}}{10^6\ \mathrm{CS}_O}\right) \quad (2)$$
$$= 4.256\ \mathrm{mSec}.$$

2.3 Offset-Quadrature Phase Shift Keying Modulation

QPSK modulation consists of two orthogonal Binary Phase Shift Keying (BPSK) signals. One BPSK signal being the In-Phase (I)-channel and the other BPSK signal being the Quadrature (Q)-channel. Therefore, QPSK consists of four unique communication symbols (0, 1, 2, and 3). QPSK splits a binary message stream, $m(t)$, into two separate channels and represents the bits in non-return-to-zero form. The IEEE 802.15.4 standard requires that the I-channel consists of the even bits and the

Q-channel consists of the odd bits [7]. The two channels are constructed as follows:

$$\mathbf{m}(t) = m_0, m_1, m_2, m_3, m_4, m_5, ..., \tag{3}$$

$$\mathbf{m_I}(t) = (m_0, m_2, m_4, ...,) \cdot 2 - 1, \tag{4}$$

$$\mathbf{m_Q}(t) = (m_1, m_3, m_5, ...,) \cdot 2 - 1, \tag{5}$$

where $\mathbf{m_I}(t)$ is the I-channel and $\mathbf{m_Q}(t)$ is the Q-channel. The transmitted QPSK modulated signal, $\mathbf{s}(t)$, is created by combining the I and Q channels such that

$$\mathbf{s}(t) = \mathbf{m_I}(t)\cos(2\pi f_0 t + \phi) + j\mathbf{m_Q}(t)\sin(2\pi f_0 t + \phi) \tag{6}$$

f_0 is the carrier frequency, t is time, and ϕ is an arbitrary phase offset. As mentioned previously, Zigbee operates by using an O-QPSK modulation scheme. O-QPSK is a variant of QPSK modulation such that the Q-channel incurs a delay before modulation.

O-QPSK is similar to QPSK, but the difference arises in how the modulation scheme transitions between communication symbols. In QPSK, a symbol (0, 1, 2, or 3) can transition to any other symbol even if there exists π radians of separation between the two symbols. QPSK performs zero-crossings within the constellation projection to transition between symbols separated by π radians. O-QPSK eliminates zero-crossings by only allowing symbols to transition to adjacent symbols within the constellation projection (maximum separation of $\pi/2$ radians). In application, the Q-channel becomes delayed by half of a QPSK symbol period before being combined in the same manner outlined in (6). The delayed channel forces only one chip associated with the QPSK symbol to change at a time, which limits the angle between symbol

transitions.

O-QPSK is desirable for Zigbee applications since the modulation scheme enables the use of highly efficient amplifiers. The high efficiency of the amplifier is crucial to the low-power constraints that Zigbee devices require. Constellation diagrams for QPSK and O-QPSK are shown in Figure 7 and Figure 8 respectively.

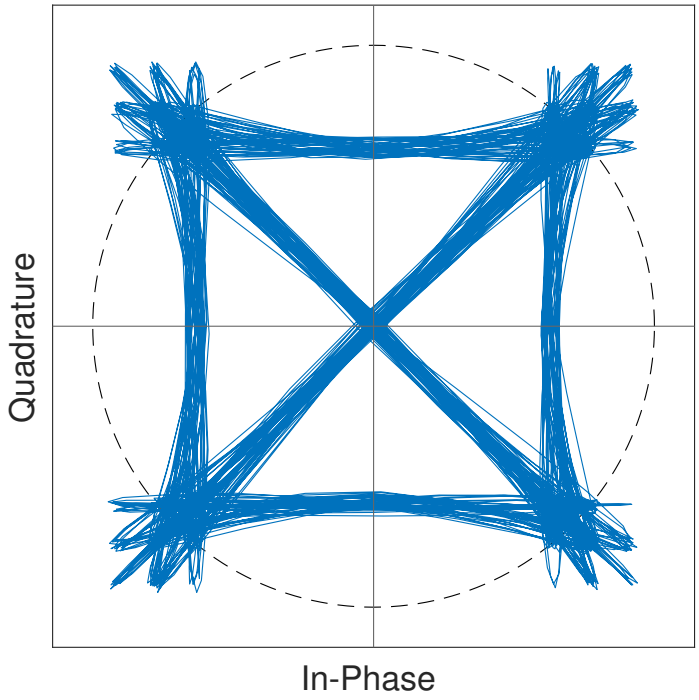

Figure 7: Constellation diagram, with symbol transitions shown, for a QPSK signal after implementing a root-raised cosine pulse shaping filter with a roll-off factor of $\beta = 0.8$. Zero-crossings are present due to symbol transitions of π radians.

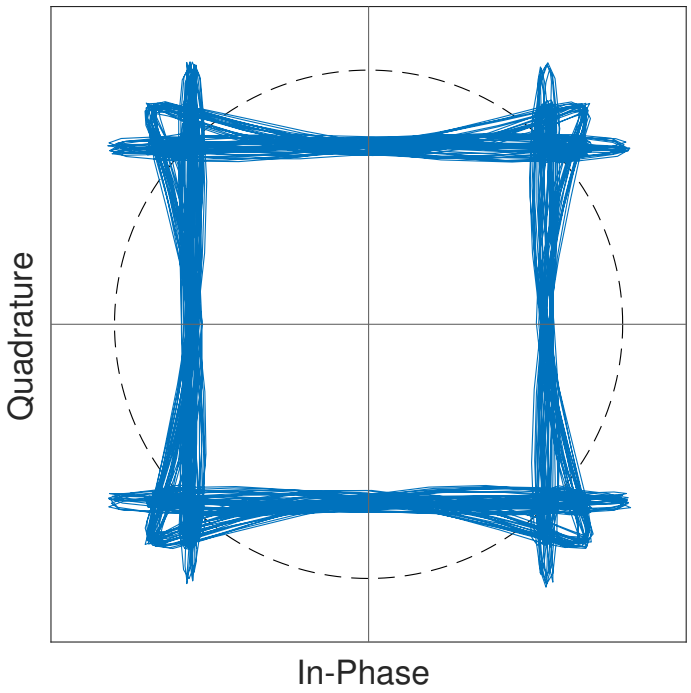

Figure 8: Constellation diagram, with symbol transitions shown, of an O-QPSK signal after implementing a root-raised cosine pulse shaping filter with a roll-off factor of $\beta = 0.8$. No zero-crossings are present since symbol transitions of π radians are eliminated.

2.4 Distinct Native Attribute (DNA) Fingerprint Generation

Distinct Native Attribute (DNA) fingerprinting is a method of generating statistics about the Open Systems Interconnection (OSI) model PHY characteristics specific to a device. These fingerprints enable both device classification (one device vs. many devices) and verification (one device vs. one device) when implemented with a classifier (addressed further in Section 2.5). Two common DNA fingerprint generation techniques are RF-DNA and CB-DNA. Section 2.4.1 addresses RF-DNA and Section 2.4.2 further discusses CB-DNA. For completeness, there are many additional fingerprint generation techniques such as, but not limited to, Slope-based Frequency Shift Keying [8, 9], Chip-Shape DNA [10], and Correlation-Based DNA [11]. The majority of these techniques are subsets of RF-DNA and tailored for specific applications. Therefore,

this work only addresses RF-DNA and CB-DNA.

2.4.1 Radio Frequency-Distinct Native Attribute

RF-DNA is the process of generating device "fingerprints" by using either spectral or time-domain signal features. This research stems directly from the time-domain techniques for device fingerprint generation. Generally, the time-domain characteristics of interest for RF-DNA fingerprint generation are the signal's instantaneous amplitude ($\mathbf{a}[n]$), phase ($\boldsymbol{\phi}[n]$) and frequency ($\mathbf{f}[n]$). Since RF-DNA uses instantaneous signal characteristics, fingerprints can be generated for any received signal regardless of the modulation scheme. However, RF-DNA implementation generally occurs in a constant portion of the signal that does not change between bursts to improve the homogeneity of fingerprints. Therefore, commonly targeted portions of the signal for RF-DNA include synchronization regions. Regular synchronization regions utilized in RF-DNA are the preamble, midamble, or postamble of a signal.

For complex signals, each of the instantaneous signal characteristics above can be calculated as follows:

$$\mathbf{a}[n] = \sqrt{\mathbf{x}^2[n] + \mathbf{y}^2[n]}, \tag{7}$$

$$\boldsymbol{\phi}[n] = \tan^{-1}\left(\frac{\mathbf{y}[n]}{\mathbf{x}[n]}\right), \tag{8}$$

$$\mathbf{f}[n] = \frac{d\boldsymbol{\phi}[n]}{dn}, \tag{9}$$

where $\mathbf{x}[n]$ is the real portion of the signal, $\mathbf{y}[n]$ is the imaginary portion, and $\tan^{-1}(\bullet)$ is the four-quadrant inverse tangent. For each of the three signal characteristics, statistical analysis is performed to calculate the variance (σ^2), skewness (γ), and

kurtosis (κ) of each characteristic. The signal of interest regularly is divided into multiple sub-regions for which the desired statistics are calculated to create the RF-DNA fingerprints. Below are the equations used to compute the desired statistics:

$$\sigma^2 = \frac{1}{N-1}\sum_{n=1}^{N}\left(\mathbf{z}[n]-\bar{z}\right)^2, \tag{10}$$

$$\gamma = \frac{\frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{z}[n]-\bar{z}\right)^3}{\left(\sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{z}[n]-\bar{z}\right)^2}\right)^3}, \tag{11}$$

$$\kappa = \frac{\frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{z}[n]-\bar{z}\right)^4}{\left(\frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{z}[n]-\bar{z}\right)^2\right)^2}, \tag{12}$$

such that $\mathbf{z}[n]$ represents the signal characteristic of interest (instantaneous amplitude, phase, or frequency) and $\bar{z}$ is the characteristic's sample mean. Therefore, the generated statistics for each sub-region creates a portion of the fingerprint structure ($\mathbf{f}_{TD_i}$) where the ith sub-region yields

$$\mathbf{f}_{TD_i} = [\sigma_i^2,\ \gamma_i,\ \kappa_i]. \tag{13}$$

The final fingerprint structure ($\mathbf{f}_{TD}$) is each of the sub-region fingerprints concatenated together such that

$$\mathbf{f}_{TD} = [\mathbf{f}_{TD_1} \vdots \mathbf{f}_{TD_2} \vdots ... \vdots \mathbf{f}_{TD_N}]. \tag{14}$$

Therefore, the total number of features (N_F) generated for a RF-DNA fingerprint structure is

$$N_F = (N_R \times N_C \times N_S) \tag{15}$$

where N_R is the number of sub-regions for the signal of interest, N_C is the number of signal characteristics utilized, and N_S is the number of statistics per characteristic calculated.

While the RF-DNA technique above described targeted communication system transmissions, RF-DNA applications are much broader. Additional RF-DNA research include classifying devices based upon unintentional emissions [12, 13, 14, 15] and determining device configurations using noise radar signals [16, 17]. Finally, Near Real-Time (NRT) RF-DNA functionality was shown to be feasible when targeting the preamble of Zigbee protocol bursts [18].

2.4.2 Constellation-Based Distinct Native Attribute

CB-DNA detects variations in received communication symbols within the constellation space. Specifically, the magnitude and phase variations are of interest for CB-DNA. Figure 9 presents an arbitrary example of these communication symbols' amplitude and phase deviations from the ideal symbol locations. It is important to note that this fingerprinting technique is only applicable to constellation-based communication schemes because the process is dependent upon calculating statistics from the amplitude and phase of the received communication symbols. CB-DNA additionally varies from RF-DNA by demodulating the received signal to estimate the transmitted symbols. The constellation projection values for the estimated symbols generate the CB-DNA fingerprints, in contrast to RF-DNA which calculates the instantaneous signal characteristics for a specific region of the signal. Ultimately, CB-DNA results in the ability to use the entire collected burst (communication symbols associated with hundreds of Zigbee DSs), while RF-DNA is typically limited to the synchronization region of the signal ($N_{RF-DNA} = 10$ Zigbee DSs).

For CB-DNA, the nine typical statistical features of interest are:

- Variance of the signal's phase (σ_ϕ^2)
- Variance of the signal's amplitude (σ_a^2)
- Skewness of the signal's phase (γ_ϕ)
- Skewness of the signal's amplitude (γ_a)
- Kurtosis of the signal's phase (κ_ϕ)
- Kurtosis of the signal's amplitude (κ_a)
- Autocovariance of the real portion of the signal (σ_{xx})
- Autocovariance of the imaginary portion of the signal (σ_{yy})
- Covariance of the real and imaginary portions of the signal (σ_{xy})

Below are the methods to compute the desired statistics:

$$\mu_a = \frac{1}{N}\sum_{n=1}^{N}\sqrt{\mathbf{x}^2[n] + \mathbf{y}^2[n]}, \tag{16}$$

$$\mu_\phi = \frac{1}{N}\sum_{n=1}^{N}\tan^{-1}\left(\frac{\mathbf{y}[n]}{\mathbf{x}[n]}\right), \tag{17}$$

$$\sigma_a^2 = \frac{1}{N-1}\sum_{n=1}^{N}(\sqrt{\mathbf{x}^2[n] + \mathbf{y}^2[n]} - \mu_a)^2, \tag{18}$$

$$\sigma_\phi^2 = \frac{1}{N-1}\sum_{n=1}^{N}(\tan^{-1}\left(\frac{\mathbf{y}[n]}{\mathbf{x}[n]}\right) - \mu_\phi)^2, \tag{19}$$

$$\gamma_a = \frac{\frac{1}{N}\sum_{n=1}^{N}(\sqrt{\mathbf{x}^2[n] + \mathbf{y}^2[n]} - \mu_a)^3}{\left(\sqrt{\frac{1}{N}\sum_{n=1}^{N}(\sqrt{\mathbf{x}^2[n] + \mathbf{y}^2[n]} - \mu_a)^2}\right)^3}, \tag{20}$$

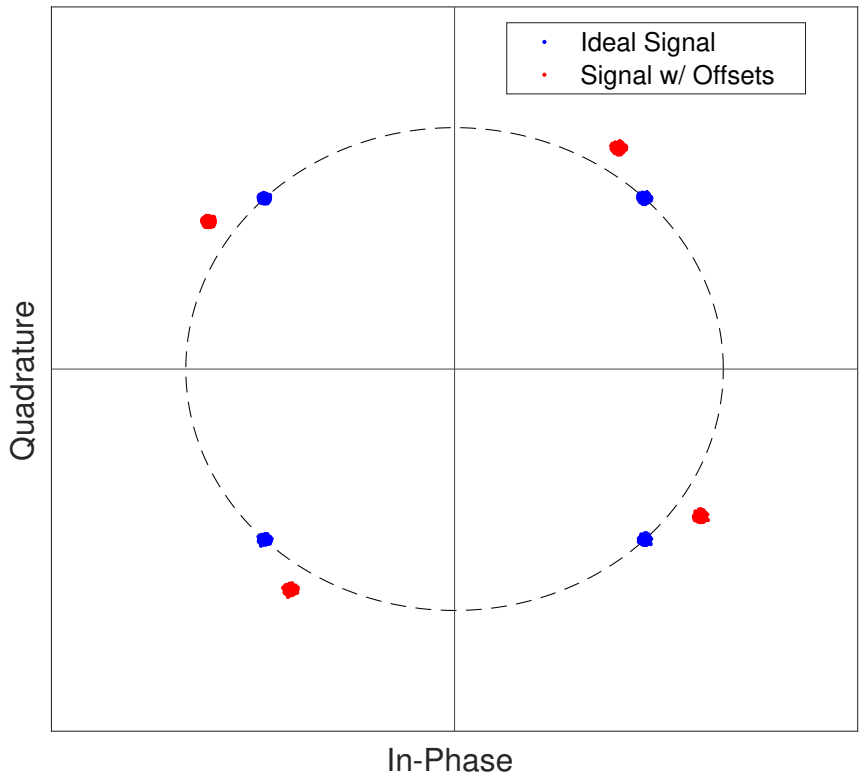

Figure 9: Ideal O-QPSK Constellation Projection (Blue) vs O-QPSK Constellation Projection with Magnitude and Phase Offsets

$$\gamma_\phi = \frac{\frac{1}{N}\sum_{n=1}^{N}(\tan^{-1}\left(\frac{\mathbf{y}[n]}{\mathbf{x}[n]}\right) - \mu_\phi)^3}{\left(\sqrt{\frac{1}{N}\sum_{n=1}^{N}(\tan^{-1}\left(\frac{\mathbf{y}[n]}{\mathbf{x}[n]}\right) - \mu_\phi)^2}\right)^3}, \tag{21}$$

$$\kappa_a = \frac{\frac{1}{N}\sum_{n=1}^{N}(\sqrt{\mathbf{x}^2[n] + \mathbf{y}^2[n]} - \mu_a)^4}{\left(\frac{1}{N}\sum_{n=1}^{N}(\sqrt{\mathbf{x}^2[n] + \mathbf{y}^2[n]} - \mu_a)^2\right)^2}, \tag{22}$$

$$\kappa_\phi = \frac{\frac{1}{N}\sum_{n=1}^{N}(\tan^{-1}\left(\frac{\mathbf{y}[n]}{\mathbf{x}[n]}\right) - \mu_\phi)^4}{\left(\frac{1}{N}\sum_{n=1}^{N}(\tan^{-1}\left(\frac{\mathbf{y}[n]}{\mathbf{x}[n]}\right) - \mu_\phi)^2\right)^2}, \tag{23}$$

$$\bar{x} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}[n], \tag{24}$$

$$\bar{y} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{y}[n], \tag{25}$$

$$\sigma_{xx} = \frac{1}{N-1}\sum_{n=1}^{N}(\mathbf{x}[n]-\bar{x})^2, \tag{26}$$

$$\sigma_{yy} = \frac{1}{N-1}\sum_{n=1}^{N}(\mathbf{y}[n]-\bar{y})^2, \tag{27}$$

$$\sigma_{xy} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}[n]-\bar{x})(\mathbf{y}[n]-\bar{y}). \tag{28}$$

Since CB-DNA demodulates the received signal back to the original communication symbol transmitted, instead of breaking the signal into an arbitrary amount of sub-regions as performed in RF-DNA, the number of communication symbol associated with the modulation scheme determines the number of sub-regions. Utilizing the number of communication symbols as the number of sub-regions is referred to as unconditional CB-DNA [19]. For O-QPSK, the number of unconditional regions is $N_{\text{regions}} = 4$ to correspond with communication symbols 0, 1, 2, and 3. In previous research, it was identified that the statistical features calculated for the current communication symbol being estimated were also dependent upon at least the preceding and proceeding symbols [19, 20, 21, 22]. This research discovery produced conditional sub-cluster CB-DNA for which the previous, current, and next communication symbols are significant when calculating the statistical features [19, 20, 21]. For O-QPSK, there are $N_{\text{transitions}} = 4^3 = 64$ unique combinations that can occur with three symbols. However, for the Zigbee protocol, it has been identified through an exhaustive search method that only $N_{\text{likely}} = 30$ O-QPSK symbol combinations are likely to occur (Figure 10) [11]. This finding significantly reduces the number of conditional sub-cluster conditions to check during CB-DNA fingerprint generation.

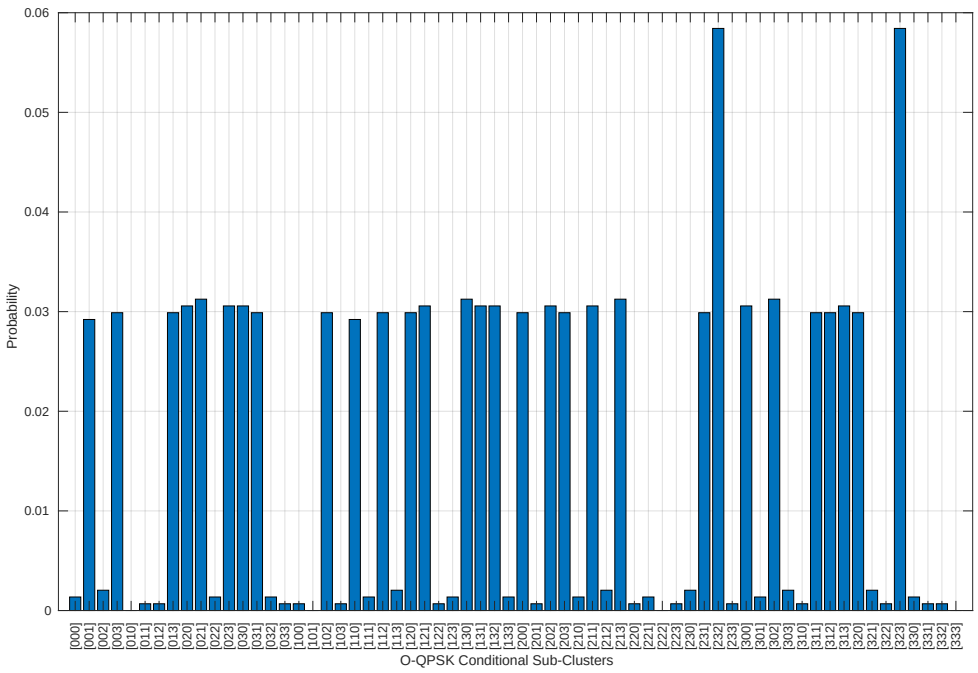

Figure 10: O-QPSK CB-DNA Conditional Sub-Clusters

This work calculated CB-DNA fingerprints utilizing conditional sub-clusters. The fingerprint structure utilized each of the $N_{\text{likely}} = 30$ likely conditional sub-clusters to occur in a Zigbee transmission (O-QPSK symbol sequences given in Section 3.6.5). The total number of statistical features in the CB-DNA fingerprint structure is

$$N_{\text{features}_{\text{CB}}} = N_{\text{statistics}} \times N_{\text{sub-clusters}} \tag{29}$$

where $N_{\text{statistics}}$ is the number of statistical features calculated for each sub-cluster and $N_{\text{sub-clusters}}$ is the number of conditional sub-clusters of interest for the modulation scheme. Therefore, the number of statistical features utilized in this research was

$$N_{\text{features}_{\text{CB}}} = 9\frac{\text{features}}{\text{sub-cluster}} \times 30 \text{ sub-clusters} = 270 \text{ features.} \tag{30}$$

Previous research conducted utilizing CB-DNA with Zigbee devices includes [22, 23]. These experiments performed verification for a set of Zigbee protocol devices with

a high degree of accuracy and also assessed different Dimensional Reduction Analysis (DRA) techniques. Additionally, research exercising CB-DNA extends beyond the IEEE 802.15.4 standard. Several experiments performed CB-DNA classification of Ethernet network cards using unintentional cable emissions [19, 20, 21].

2.5 Classifiers

Multiple Discriminant Analysis (MDA) is a Fisher-based model that expands Fisher's Linear Discriminant Analysis (LDA) beyond projecting classes into a single dimension. MDA reduces a c-class problem into a $c-1$ dimensional space. In general, MDA maximizes the ratio of the distance between classes to the variance within a class. Using k-fold partitioning, $N_{\text{k-fold}} = \binom{k}{k-1}$ Projection Matrices ($\mathbf{W}$s) are created and the $\mathbf{W}$ that has the best between-class scatter and within-class scatter is selected.

In Figure 11, two arbitrary $\mathbf{W}$s for a 3-class problem are presented [24]. For this problem, $\mathbf{W}_1$ is the better $\mathbf{W}$ since the classes do not overlap. No insight towards feature relevance is provided with the returned $\mathbf{W}$. Two additional classifiers previously utilized for DNA applications that provide information into feature relevance are Generalized Relevance Learning Vector Quantization-Improved (GRLVQI) and Random Forest (RndF). Both GRLVQI and RndF require significantly more computational time to develop classification models when compared to MDA. For completeness, additional information of GRLVQI can be found here [25, 26, 27], while information on RndF can be found here [23, 28, 29].

Ultimately, MDA is performed on a given fingerprint structure to produce a $c-1$ dimension $\mathbf{W}$. This $\mathbf{W}$ is then later employed with a different fingerprint structure to perform device classification. Maximum Likelihood (ML) classification, a common classifier used for RF-DNA, utilizes the generated projection matrix to classify test

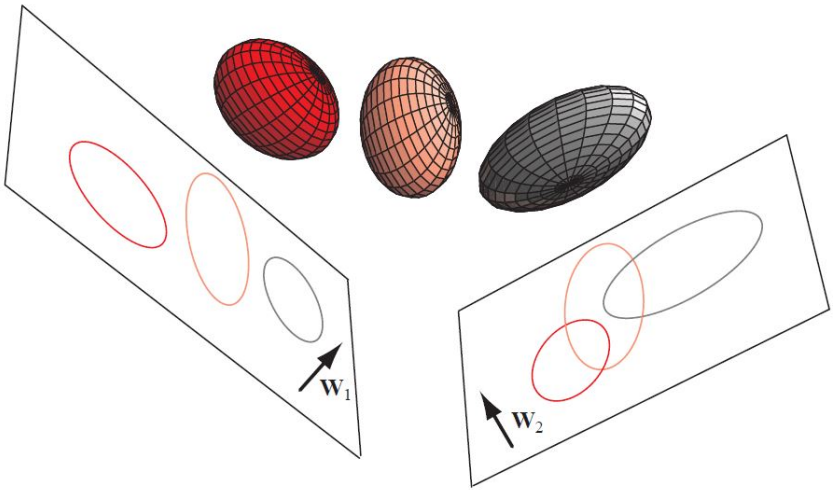

Figure 11: Two projection spaces for MDA classifier [24]

fingerprints (fingerprints not used to create the projection matrix). This process assumes that the fingerprints utilized for training follow a Gaussian distribution. ML then creates Bayesian decision boundaries to determine which class a test fingerprint should be associated. However, to reduce the computational complexity, this research employed a nearest-neighbor classifier by calculating the ED between the test fingerprint in the projection space and each of the class means using

$$d = \sqrt{\sum_{i=1}^{c-1} (\mu_{\text{class}_i} - \mathbf{f}_{\text{tst}_i})^2} \tag{31}$$

where d is the ED, c is the number of classes, μ_{class} is the class mean, and $\mathbf{f}_{\text{tst}}$ is the projected test fingerprint. The class mean with the shortest distance to the projected fingerprint is determined to be the "most likely" class. Utilizing ED as a classifier have been demonstrated in [18] (with MDA using RF-DNA fingerprints) and [19] (without MDA using CB-DNA fingerprints).

2.6 SDR

A Software Defined Radio (SDR) is a radio device that employs software to accomplish tasks traditionally implemented via hardware. This software approach allows SDRs to be extremely flexible devices by enabling the user to continually reconfigure the devices to specific applications within the SDR's design constraints. This research utilized two different types of SDRs, one type as the transmitting devices and the other as the Radio Frequency (RF) air-monitor. BladeRF SDRs were the transmitting devices, while an Ettus Universal Software Radio Peripheral (USRP) x310 SDR was the RF air-monitor.

2.7 C++ Libraries and Software Tools

C++ inherently provides the capability to perform nominal programming tasks such as algebraic operations and reading/writing to files. However, for specialty applications, additional libraries and software tools provide a means of efficiently performing non-standard tasks. Sections 2.7.1 - 2.7.3 discuss various C++ libraries and the capabilities they provide. Section 2.7.4 addresses an additional software tool, outside of C++, utilized during the course of this research.

2.7.1 USRP Hardware Driver

USRP Hardware Driver (UHD) is a user-space library developed by Ettus Research to interface with Ettus USRP devices via a Command Line Interface (CLI). The software provides a method to identify USRP devices connected to the host computer, as well as calibrate the connected devices to minimize In-Phase/Quadrature-channel imbalance. Additionally, this software supplies the necessary libraries for higher-level languages, such as C++, to configure USRP devices for the transmission and reception of samples.

2.7.2 Eigen

Eigen is an open-source C++ template library that performs linear algebra operations with vectors and matrices [30]. This library provides a streamlined method of accomplishing matrix operations such as multiplication, conjugation, inverse, transposition, dot product, and cross-product.

2.7.3 liquid-dsp

`liquid-dsp` is an open-source Digital Signal Processing (DSP) library written in the C programming language employed to process SDR data [31]. Even though `liquid-dsp` targets applications using C, the library can also be compiled and linked with C++ files and data structures. `liquid-dsp` operations include, but are not limited to, generating filters, performing Fourier transformations, and creating numerically-controlled oscillators.

2.7.4 bladeRF

`bladeRF` is a software tool utilized to control the BladeRF SDRs [32]. This software provides a CLI tool called `bladerf-cli` which enables the user to configure the SDR to a desired operating mode. Examples of the different BladeRF SDR applications enabled through the software include utilization as a RF modulator/demodulator, a Global Positioning System (GPS) receiver, and an Advanced Television Systems Committee (ATSC) transmitter.

III. Methodology

3.1 Introduction

This chapter outlines the methodology employed to determine the applicability of Near Real-Time (NRT) device classification with Constellation-Based Distinct Native Attribute (CB-DNA) fingerprints. Additionally presented in this chapter are the research goals (Section 3.2), hypotheses (Section 3.3), and measures of merit (Section 3.4) for the experiment. Section 3.5 describes the Zigbee signal transmitted for both training and testing scenarios. To achieve the constellation projections, Section 3.6 discusses burst detection and the subsequent signal processing techniques performed to demodulate the signal. Section 3.6 also examines the process of generating CB-DNA fingerprints for the collected bursts. Sections 3.7 and 3.8 address device classification with Multiple Discriminant Analysis/Euclidean Distance (MDA/ED) using CB-DNA fingerprints.

3.2 Research Objectives

The objective of this research is to perform device classification NRT using CB-DNA generated fingerprints. The implemented process will continuously monitor the wireless spectrum for a center frequency of $f_c = 2480$ MHz and a $W_{\text{rx}} = 10$ MHz bandwidth for Zigbee transmissions. All collected Zigbee bursts drive the fingerprint and classification processes. The goal of the classifier is to estimate the "most likely" transmitted device using data collected from training runs. This research ultimately enhances the security of Low-Rate Wireless Personal Area Networks (LR-WPAN) by implementing an air monitor to augment bit-level credentials with Physical Layer (PHY) verification. Figure 12 provides a block diagram of the experimental methodology.

3.3 Research Hypotheses

This research assesses the following hypotheses:

- CB-DNA fingerprint generation can be accomplished NRT to facilitate classification of devices using the Institute of Electrical and Electronics Engineers (IEEE) 802.15.4 standard.
- Device classification performance will be consistent with previous NRT Radio Frequency Distinct Native Attribute (RF-DNA) research endeavors [18].

3.4 Measures of Merit

Measures of merit for the research include:

- Average Cross-Class Percent Correct Classification (%C) of devices for the MDA/ED classifier.
- Average runtime of the air monitor from burst detection to classification for received Zigbee bursts.

For the experiment, %C performance is verified by producing a Confusion Matrix (CM) of the testing scenario. Timing analysis data of Zigbee test bursts provide the NRT performance of the air monitor.

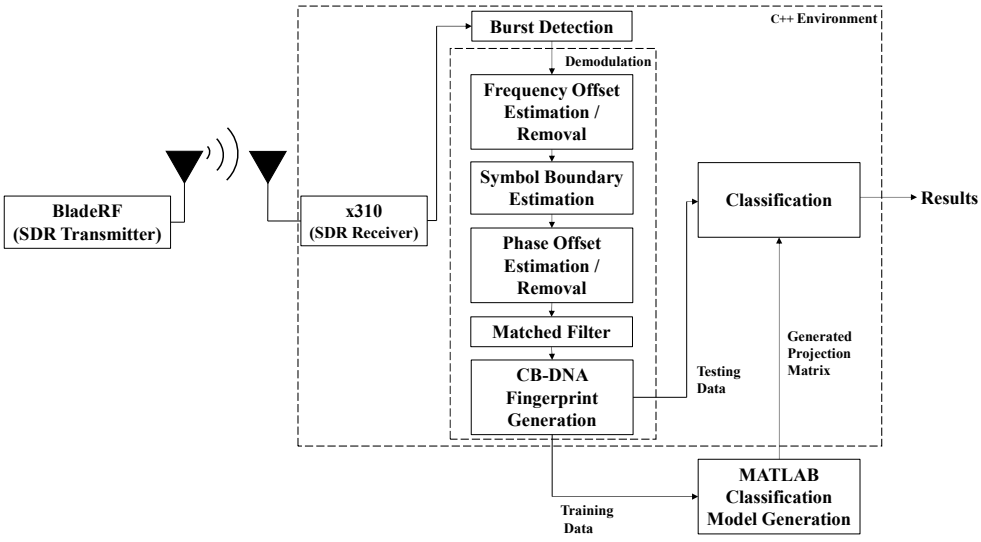

Figure 12: Block diagram of SDR transmitter and RF air montor processes

3.5 SDR Transmitter Design and Implementation

Using the IEEE 802.15.4 standard, $N_{\text{cls}} = 5$ BladeRF Software Defined Radios (SDRs) transmitted Zigbee bursts for the experiment. The BladeRF SDR has a tuning range of $f_{\text{BladeRF}} = [300\ 3800]$ MHz, maximum instantaneous bandwidth of $B_{\text{BladeRF}} = 28$ MHz, and a max sampling rate of $f_{s_{\text{BladeRF}}} = 40$ Mega-Samples-Per-Second (Msps). For each BladeRF, Table 2 shows the device serial numbers as reported by software and the associated device IDs. The experiment required the creation of two separate transmission files. All training collections utilized one file, while all testing scenarios used the other transmission file.

Table 2: BladeRF SDR Device Identifiers

BladeRF Devices	
Device ID	**Device Serial Number**
BladeRF1	63b81e9e21451176dde16e31ebc81c5f
BladeRF2	814b072c9530f5116db17ee3f5dc31c4
BladeRF3	ddf13f69ae8b744cd2a0ace875c255e0
BladeRF4	6da8d69c6f3c664e7396ea3a7530e078
BladeRF5	42b92a8a385e7577a0957e73071c94a4

Each transmission file was composed of $N_{\text{bursts}} = 500$ bursts with a $t = 100$ mSec pause between bursts. Every burst contained a total of $N_{\text{DS burst}} = 212$ Zigbee Data Symbols (DSs), where the first $N_{\text{SHR}} = 12$ Zigbee DSs were constant. The initial DSs were identical in every burst to achieve the Zigbee Sychronization Header Region (SHR) and PHY Header Region (PHR) as specified by the IEEE 802.15.4 standard. The remaining $N_{\text{random}} = 200$ Zigbee DSs were randomly generated to serve as the payload for each burst.

Of note, the PHY Service Data Unit (PSDU) portion of the created bursts did not fit one of the four standard Medium Access Control (MAC) frame formats. To satisfy MAC frame requirements, the MAC frame format must be specified within the frame control field of the PSDU and a Cyclic Redundancy Check (CRC) must be performed. Once a MAC format is determined, a large portion of the PSDU is reserved for source and destination address information which increases the number of constant Zigbee DSs within the burst. For example, a burst using a data frame structure would result in $N_{\text{DS}} = 57$ constant Zigbee DSs per burst leaving only $N_{\text{DS}} = 155$ Zigbee DSs to be randomly changed. Ultimately, the randomly generated PSDU prevented potentially training the classifier with MAC address information. Additionally, since the developed bursts did not fit a standard MAC frame format, a CRC was not necessary. The randomized PSDU format also increased the population of total bursts that could be generated for training and testing.

All of the created Zigbee bursts within a transmission file had a unique payload and none of the generated bursts repeated between files. To create the Zigbee signal, the Zigbee DSs were converted to chips (Table 1), and the even and odd chips were split between the I and Q-channels respectively in a non-return-to-zero form. Additionally, the Q-Channel was delayed from the I-Channel by the period of half of a Zigbee chip (T_c) to achieve Offset-Quadrature Phase Shift Keying (O-QPSK) modulation.

Finally, the signal was then passed through a half-sine pulse shaping matched filter, $\mathbf{p}(t)$, using

$$\mathbf{p}(t) = \begin{cases} \sin\left(\pi \frac{t}{2T_c}\right) & 0 \leq t \leq 2T_c \\ 0 & \text{otherwise} \end{cases} \tag{32}$$

where T_c is half of the period of a Zigbee chip, and t is time. Half-sine pulse shaping is required since the intended Zigbee transmission channel was in the $f_{B3} = 2.4$ GHz frequency band [7]. Ultimately, pulse shaping turned the signal into a constant envelope modulation and increased the signal's performance across noisy channels. Figure 13 shows a collected Zigbee burst with half-sine pulse shaping.

Each BladeRF SDR was tuned to a center frequency of $f_c = 2480$ MHz with a

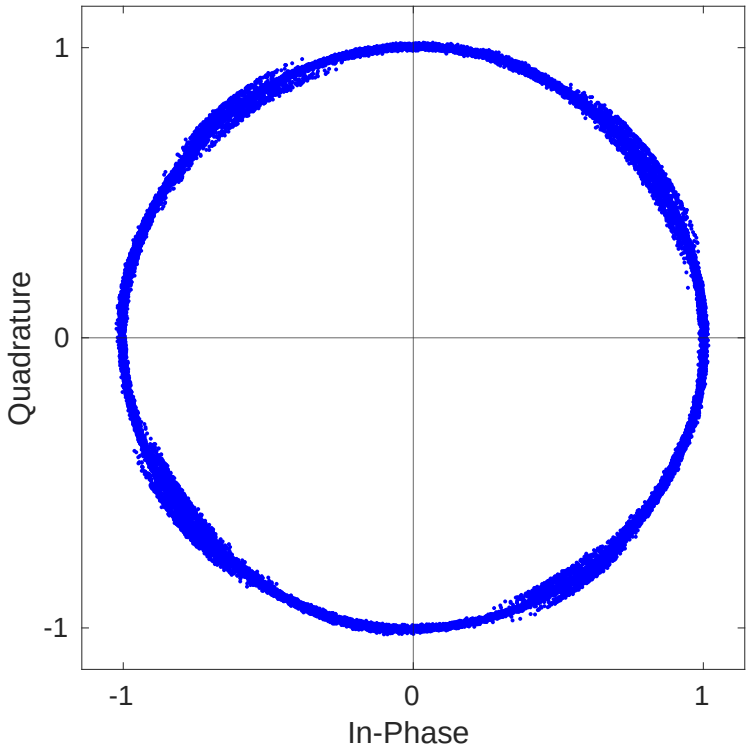

Figure 13: Constellation diagram, with symbol transitions shown, of an over-the-air collected Zigbee burst with half-sine pulse shaping. SNR was determined experimentally to be $E_s/N_0 \approx 31$ dB.

bandwidth of $W_{\text{tx}} = 2$ MHz (Zigbee channel $N_{\text{tx ch}} = 26$ from Figure 1). Additionally, the sampling rate was set to $f_s = 10$ Msps and the data was transmitted as 16-bit complex shorts with a $SF = 2^{11}$ scaling factor to maximize the SDR's dynamic range. SDR transmission required the use of the bladeRF-cli tool in the bladeRF host libraries. This command-line tool configured the BladeRF SDR for transmitting at the desired specifications listed above and also enabled the SDR to transmit bursts from a binary file generated by a MATLAB® script. During all training and testing transmissions, a $R = 50\ \Omega$ Sub-Miniature version A (SMA) dummy load was connected to the BladeRF's receiver port.

To ensure consistency throughout the experiments, an SMA cable with a 30 dB attenuator connected the transmitting BladeRF to the Universal Software Radio Peripheral (USRP) x310 SDR for all training and testing collections. This setup successfully enabled consistent collections for the experiment (Figure 14 and Figure 15). Since the CB-DNA fingerprints were generated by performing statistical analysis on the constellation projection cued from the demodulated O-QPSK symbols, the classifier could potentially develop inaccurate Projection Matrices ($\mathbf{W}$s) by keying in on external factors such as Radio Frequency Interference (RFI). The Zigbee protocol inherently prevents RFI by performing a Clear Channel Assessment (CCA) using Carrier Sense Multiple Access with Collision Avoidance (CSMA-CA) to ensure that only one device utilizes the channel at a time [7, 33]. Specifically, the impact of RFI occurring on a transmitted Zigbee burst can be seen in Figure 16. From Figure 17, it was found that external factors such as RFI impact the demodulation process and can potentially distort the generated CB-DNA fingerprints used for training and testing. Thus, for the purposes of this research, the cable and attenuator emulated a Zigbee network after successful CCA.

Additionally, the Energy per Symbol to Noise Power Spectral Density ($\text{E}_\text{s}/\text{N}_0$) was

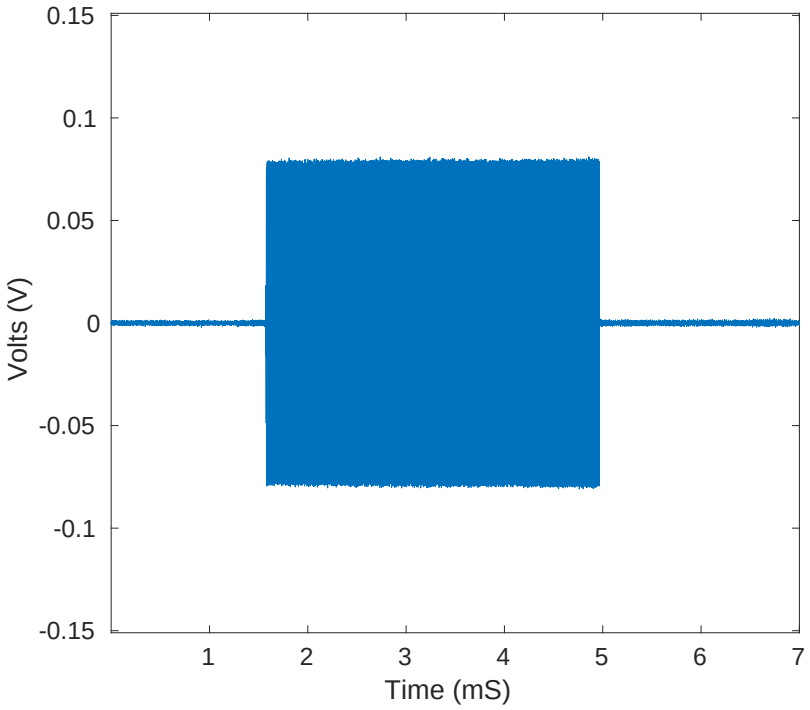

Figure 14: Half-sine pulse shaped Zigbee burst at $f_c = 2480$ MHz collected using a cable and 30 dB attenuator.

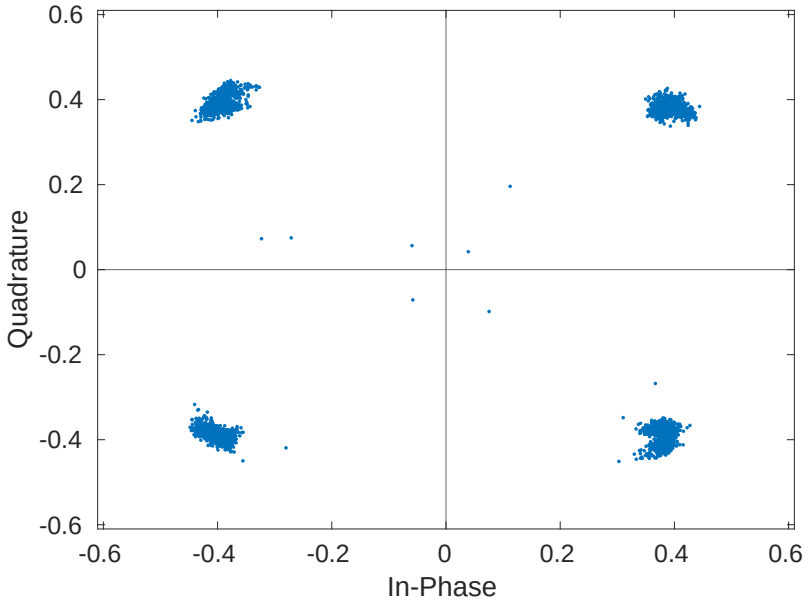

Figure 15: Constellation Projection of the estimated O-QPSK symbols from the half-sine pulse shaped Zigbee burst shown in Figure 14.

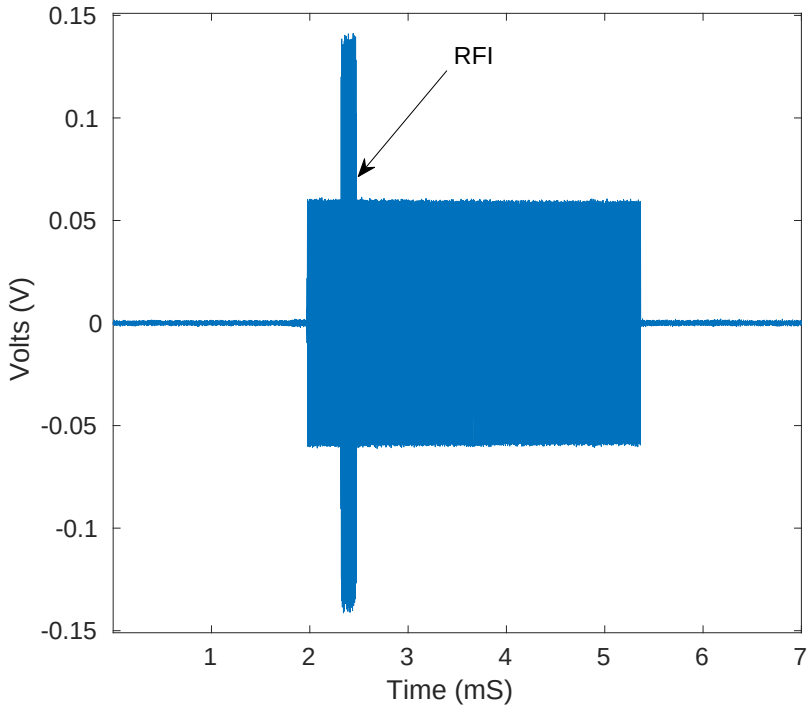

Figure 16: Over-the-air collected half-sine pulse shaped Zigbee burst at $f_c = 2480$ MHz. The annotated spike represents RFI occurring from an unknown source.

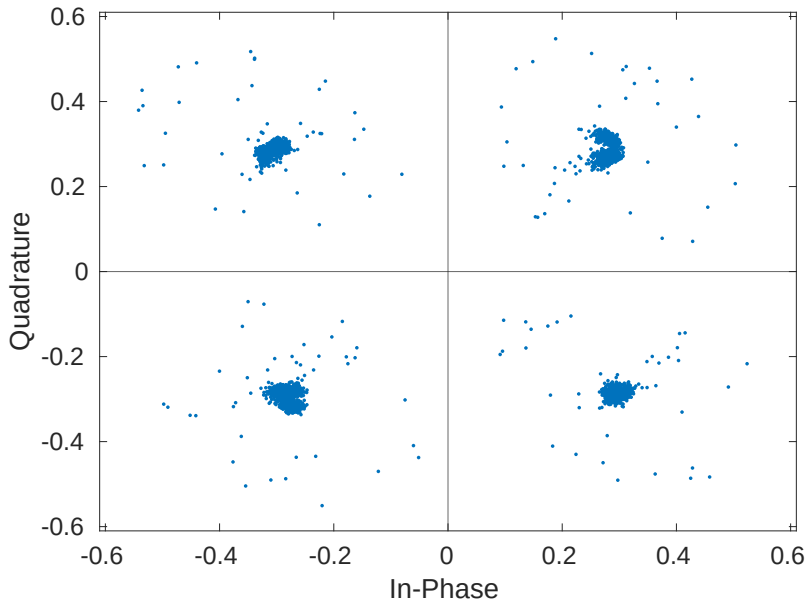

Figure 17: Constellation projection of the estimated O-QPSK symbols from the half-sine pulse shaped Zigbee burst shown in Figure 16. The RFI impacts the distribution of O-QPSK symbols within the constellation projection.

determined from device collections using the experimental configuration. To calculate the $\mathrm{E_s/N_0}$ in Decibels (dB), first the average power of the collected complex signal $s_\mathrm{c}[m]$ was calculated by

$$S_\mathrm{c} = \frac{1}{M}\sum_{m=0}^{M-1} s_\mathrm{c}[m]s_\mathrm{c}^*[m] \tag{33}$$

such that $s_\mathrm{c}^*[m]$ is the complex conjugate of $s_\mathrm{c}[m]$. Additionally, it was assumed that the collected complex signal consisted of

$$s_\mathrm{c}[m] = s[m] + n_\mathrm{b}[m], \tag{34}$$

$$N_\mathrm{b} = \frac{1}{M}\sum_{m=0}^{M-1} n_\mathrm{b}[m]n_\mathrm{b}^*[m] \tag{35}$$

where $s[m]$ is the transmitted complex signal, $n_\mathrm{b}[m]$ is the collected background noise, and N_b is the average power of the background noise. The Signal-to-Noise Ratio (SNR) of the burst was calculated in dB such that

$$\mathrm{SNR} = 10\log_{10}\left(\frac{S_\mathrm{c} - N_\mathrm{b}}{N_\mathrm{b}}\right). \tag{36}$$

From the SNR value, the $\mathrm{E_s/N_0}$ achieved in dB was calculated using

$$\mathrm{E_s/N_0} = 10\log_{10}\left(\frac{\mathrm{T_{sym}}}{\mathrm{T_{samp}}}\right) + \mathrm{SNR} \tag{37}$$

such that $\mathrm{T_{sym}}$ is the period of an O-QPSK symbol and $\mathrm{T_{samp}}$ is period of a sample. For the experimental configuration utilized in this research, the $\mathrm{E_s/N_0}$ value achieved over the cable was $\mathrm{E_s/N_0} \approx 31$ dB for all devices.

3.6 C++ Environment

The top-level entity within the C++ environment worked by performing four main tasks, each with its own dedicated thread. This C++ object builds upon a multi-threaded, circular buffer class developed by [34]. The first thread configures the receiver SDR and streams complex-samples into the C++ environment (Section 3.6.1). The second thread takes the received SDR samples and performs burst detection (Section 3.6.2). The collected burst is then passed to a third thread which mitigates frequency and phase offsets between the transmitter and receiver, and demodulates the signal (Sections 3.6.3 - 3.6.4). This thread also calculates the CB-DNA fingerprints (Section 3.6.5). The last thread classifies the collected Zigbee burst (Section 3.8) using data generated from a set of training fingerprints (Section 3.7).

3.6.1 SDR Receiver Design and Implementation

A USRP x310 SDR with an SBXv3 daughterboard was the receiver utilized as the air monitor for this experiment. The SBX daughterboard has a tuning range of $f_{\text{x310}} = [400\ 4400]$ MHz, maximum instantaneous bandwidth of $B_{\text{x310}} = 40$ MHz, and a max sampling rate of $f_{s_{\text{x310}}} = 20$ Msps. The SBXv3 was selected for the experiment since the targeted transmissions will occur at $f_c = 2480$ MHz.

A Global Positioning System (GPS) Disciplined Oscillator (GPSDO) Oven Controlled Crystal Oscillator (OCXO) Mini was also installed in the USRP x310 to improve the accuracy of the reference clock. The OCXO works by maintaining the oscillator's crystal at a fixed temperature that is higher than the manufacturer operating temperature range. This oven-controlled process significantly reduces the temperature variation of the crystal, which directly improves the timing accuracy of the reference clock. However, a GPS antenna kit was not installed during the experiment to achieve GPS lock. Therefore, only the OCXO improved timing accuracy for

this experiment.

The host computer utilized for this experiment was a Hewlett-Packard Z840. Table 3 contains the pertinent hardware information for the host computer.

In the C++ environment, the USRP x310 is configured to a center frequency of $f_c = 2480$ MHz with a bandwidth of $W_{\text{rx}} = 10$ MHz and a sampling rate of $f_s = 10$ Msps. The received data streamed from the SDR to the host computer in $t = 1$ ms blocks of 32-bit complex float values. For the given sampling rate, a total of $N_{\text{samples}} = (10 \times 10^6 \text{ sps})(10^{-3} \text{ Sec}) = 10^4$ samples compose the time block passed to the host computer.

For both training and testing collections, the dedicated receiver port of the USRP x310 SBXv3 daughterboard maintained a connection to the transmitting BladeRF device via a SMA cable. Additionally, the daughterboard's transceiver port was terminated with a $R = 50\Omega$ SMA dummy load during the experiment.

3.6.2 Burst Detection

Burst detection identified Zigbee transmissions with an energy detection technique. In the C++ environment, the received block of $N_{\text{samples}} = 10^4$ samples from the USRP x310 SDR were initially filtered using a 4th-order lowpass Butterworth filter with a normalized cutoff frequency of $W_n = \frac{f_{\text{Ch 26}}}{f_s} \approx 0.2480$. The coefficients for the lowpass Butterworth filter were calculated in MATLAB® and then ported to a liquid-dsp object within the C++ environment. Next, a moving average filter with

Table 3: Hardware configuration of the host computer

Computer Model:	HP Z840
Operating System:	Ubuntu 18.04.2 LTS
Processor Type:	Intel Xeon CPU E5-2687W v3 (25 MB, 3.10 GHz)
Number of Processing Cores:	10
Main Memory Type:	Micron MTA36ASF2G72PZ (DDR4 SDRAM, 16 GB, 288 RDIMM)
Total Main Memory Size:	64 GB
Secondary Memory Type:	TOSHIBA DT01ACA200 HDD (7200 RPM)
Secondary Memory Size:	2 TB

a span of $N_{\text{span}} = 250$ samples smoothed the signal. The process then represented the magnitude of the smoothed signal in dB. An energy threshold is then empirically identified from

$$dB_{\text{threshold}} = (\max(|\mathbf{s}_{\text{smooth}}[n]|) - \min(|\mathbf{s}_{\text{smooth}}[n]|)) \times 0.9 + \min(|\mathbf{s}_{\text{smooth}}[n]|), \quad (38)$$

such that $\mathbf{s}_{\text{smooth}}[n]$ is the output from the moving average filter. Since the samples streamed from the USRP x310 SDR were complex floats, the dynamic range of values was between -1 and 1. Therefore, the calculated $dB_{\text{threshold}}$ was a negative value.

Once a burst was detected, the code concatenated seven $t = 1$ mSec blocks of SDR data to ensure the entire burst was collected. Table 4 shows the composition of the seven concatenated blocks.

Since the burst detection process only verifies that the received signal exceeded an energy threshold, the collected burst is not guaranteed to be a Zigbee transmission. However, the Radio Frequency (RF) air monitor discards spurious bursts prior to CB-DNA fingerprint generation, which Section 3.6.4 addresses.

3.6.3 Mitigating Frequency and Phase Offsets

The frequency carrier offset was estimated using a Modified Rife frequency estimate as developed by [35]. The outlined approach is for Minimum Shift Keying (MSK) applications, but MSK is a special case of O-QPSK with sinusoidal symbol

Table 4: Structure of collected burst by concatenating SDR data blocks

Structure of Concatenated SDR Blocks						
Time (mSec)						
1	2	3	4	5	6	7
Burst Block - 1	Burst Block	Burst Block + 1	Burst Block + 2	Burst Block + 3	Burst Block + 4	Burst Block + 5
Data previously stored	Performed burst detection	No burst detection performed	No burst detection performed	No burst detection performed	No burst detection performed	No burst detection performed

weighting [36]. Since the $f_{B3} = 2400$ MHz band for Zigbee requires half-sine pulse shaping, the generated O-QPSK signal satisfies the above requirement. Therefore, by adopting the work from [35], the square of the received O-QPSK signal is

$$\mathbf{s}^2(t) = A^2 \exp\left(j4\pi\left[f_c + \frac{\mathbf{a}(t)}{4T_{CS}}\right] + 2\phi_0\right) \tag{39}$$

where A is the amplitude, f_c is the carrier frequency, $\mathbf{a}(t)$ are data symbols, T_{CS} is the period of an O-QPSK communication symbol, and ϕ_0 is an equivalent phase. From (39), it can be seen that the square of the O-QPSK signal is a Frequency Shift Keying (FSK) signal with two carrier frequencies such that

$$\begin{aligned} f_1 &= 2f_c + \frac{1}{2T_{CS}}, \\ f_2 &= 2f_c - \frac{1}{2T_{CS}}. \end{aligned} \tag{40}$$

Therefore, summing the two carrier frequencies in (40) and solving for f_c yields

$$\hat{f}_c = \frac{1}{4}\left(f_1 + f_2\right) \tag{41}$$

where $\hat{f}_c$ is the estimated carrier frequency offset. The described frequency estimation technique is applicable when the offset is close to baseband as shown in Figure 18. Applying the carrier frequency estimation method requires the following steps:

1. Square each sample on the collected burst from the USRP x310 SDR
2. Perform a Fourier Transformation (FT) on the squared signal
3. Split the spectrum into two regions:
   - 1st Region (R_1): $-\frac{f_s}{2} \leq R_1 \leq 0$

- 2nd Region (R_2): $0 < R_2 \leq \frac{f_s}{2}$

4. Search each of the two regions for the maximum value and record the locations of where the local maxima occur as f_1 and f_2 respectively (Figure 18)

5. Calculate $\hat{f}_c$ using (41)

The resolution of $\hat{f}_c$ is directly related to the size of the Fast Fourier Transform (FFT). The resolution error can be calculated by

$$f_q = \frac{f_s}{4N_{\text{FFT}}} \tag{42}$$

where f_q is the minimum frequency resolution for the estimate, f_s is the sampling frequency, and N_{FFT} is the number of FFT points. The scalar value of four derives from the process of estimating $\hat{f}_c$.

It was found that two factors significantly impact the demodulation process through

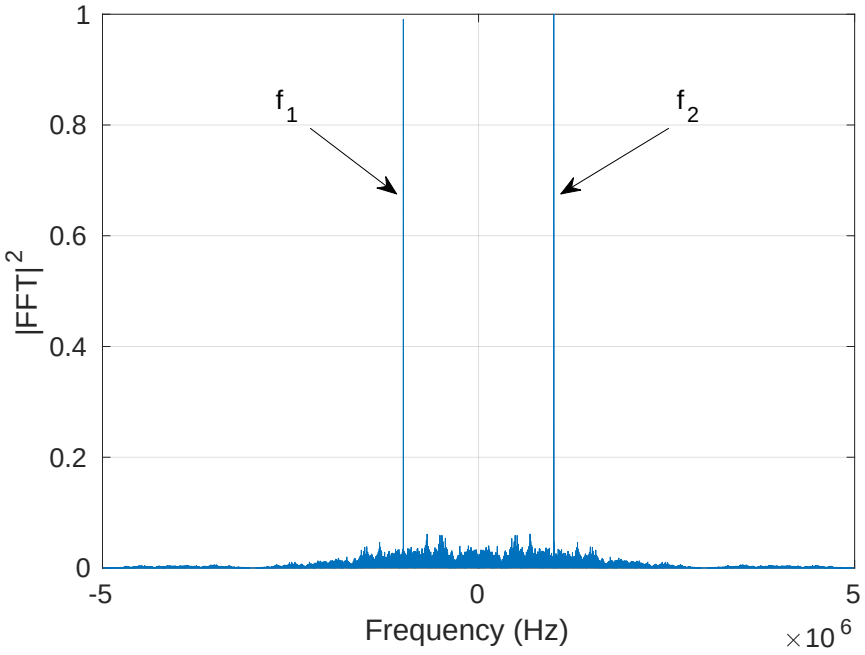

Figure 18: Normalized $|\text{FFT}|^2$ of a Zigbee burst. The number of FFT points was $N_{\text{FFT}} = 2^{20}$.

carrier frequency estimation. First, if the O-QPSK symbols rotate more than $\phi = \frac{\pi}{2}$ radians during the time of the collected burst (t_{burst}), demodulation will not be achievable because the receiver cannot synchronize the symbol constellation and the burst. Additionally, if the received signal has a low SNR, the frequency estimation process does not return an accurate value. Thus, the limits for both factors were determined to ensure that the experimental results would not be negatively impacted.

First, since a rotation greater than $\phi = \frac{\pi}{2}$ radians is the physical limit, the maximum frequency resolution able to successfully perform demodulation was determined to be

$$\begin{aligned} \sin\left(2\pi f_{q_{\max}} t_{\text{burst}}\right) &\leq \sin\left(\frac{\pi}{2}\right) \\ \Rightarrow f_{q_{\max}} &\leq \frac{1}{4t_{\text{burst}}}. \end{aligned} \tag{43}$$

Due to the fact that all training and test bursts have the same duration of $N_{\text{DS}} = 212$ Zigbee DS, the time of each collected burst was calculated to be

$$t_{\text{burst}} = \left(\frac{212\ \text{DS}_Z}{\text{burst}}\right)\left(\frac{32\ \text{chips}}{\text{DS}_Z}\right)\left(\frac{\text{CS}_O}{2\ \text{chips}}\right)\left(\frac{\text{Sec}}{10^6\ \text{CS}_O}\right) = 3.392\ \text{mSec} \tag{44}$$

where DS_Z is a Zigbee DS and CS_O is an O-QPSK communication symbol. Using (43) and the results of (44), the maximum allowable carrier frequency offset to achieve demodulation for this research is

$$f_{q_{\max}} \leq \frac{1}{4\left(3.392 \times 10^{-3}\ \text{Sec}\right)} \approx 73.7\ \text{Hz}. \tag{45}$$

Figures 19-22 show the impact of a carrier frequency offset on an ideal Zigbee signal. The ideal demodulated signal, represented in Figure 19, has $f_q = 0$ Hz car-

rier frequency offset. As the frequency offset increases, the sampling point for the demodulation process begins to drift. Figure 20 shows the drift for when $f_q = 47$ Hz. In Figure 21, the ideal signal still demodulates because the frequency drift of $f_q = 73$ Hz has not yet crossed symbol boundaries. However, in Figure 22, the signal no longer demodulates correctly as the frequency offset increases to $f_q = 74$ Hz.

Since an objective of this experiment is to be NRT, the runtime required to compute the FFT of the collected burst must be balanced with the frequency resolution provided by the FFT. Therefore, for the purpose of this experiment, the number of FFT points used for carrier frequency estimation was $N_{\text{FFT}} = 2^{20} = 1048576$ points. Utilizing (42), the frequency resolution was found to be

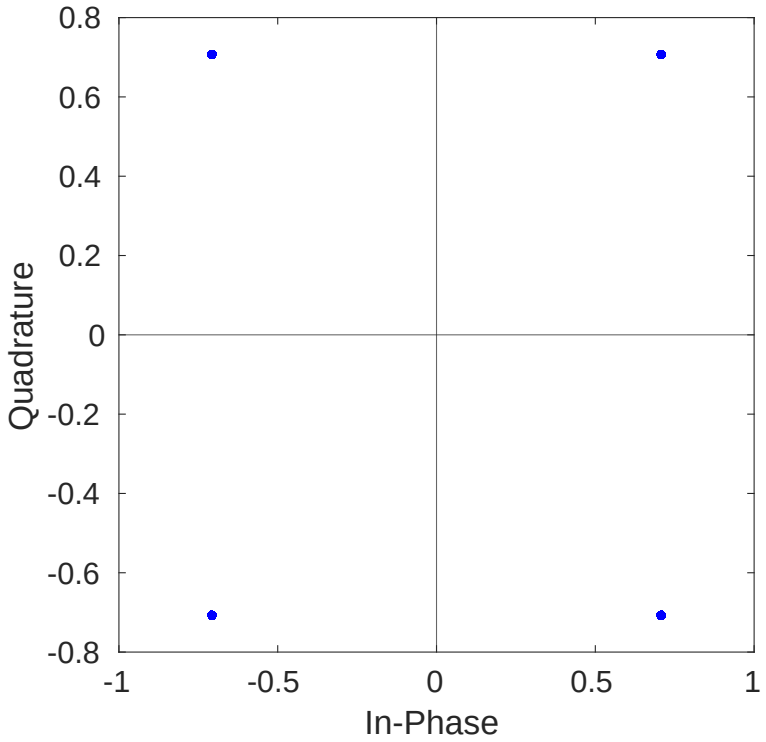

Figure 19: Constellation projection of a simulated ideal half-sine pulse shaped Zigbee burst with $f_q = 0$ Hz frequency offset. The Zigbee burst was composed of $N_{\text{DS}} = 212$ Zigbee DSs. The corresponding O-QPSK symbols are represented by discrete dots in the constellation projection. Since the signal is ideal, all $N_{\text{CS}} = 3392$ O-QPSK symbols project to one of four locations. Additionally, the signal can be demodulated correctly since the O-QPSK symbols rotate less than $\phi = \pi/2$ radians during t_{burst}.

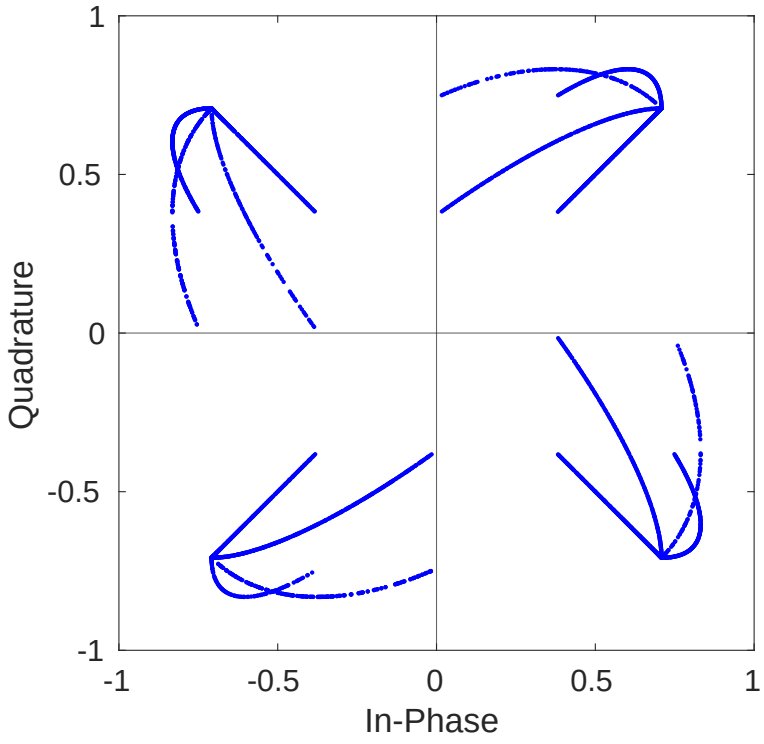

Figure 20: Constellation projection of a simulated ideal half-sine pulse shaped Zigbee burst with $f_q = 47$ Hz frequency offset. The Zigbee burst was composed of $N_{\text{DS}} = 212$ Zigbee DSs. The corresponding O-QPSK symbols are represented by discrete dots in the constellation projection. The signal can be demodulated correctly since the O-QPSK symbols rotate less than $\phi = \pi/2$ radians during t_{burst}.

$$f_q = \frac{10 \times 10^6}{(4)\,(2^{20})} \approx 2.3842 \text{ Hz}. \tag{46}$$

Additionally, this research conducted a simulation of $N_{\text{SNR}} = 1000$ bursts for every SNR value ranging from $\text{E}_\text{s}/\text{N}_0 = [-12,\ 13]$ dB in $N_{\text{dB incr}} = 0.25$ dB increments to test the limitations of the designed O-QPSK receiver. Figure 23 shows the Symbol Error Rate (SER) determined from the simulation runs for the O-QPSK receiver compared to the theoretical Quadrature Phase Shift Keying (QPSK) SER. The performance of the O-QPSK receiver closely tracked the ideal QPSK SER with the exception of SNR values below $\text{E}_\text{s}/\text{N}_0 = 0$ dB. At this point, the developed O-QPSK receiver cannot achieve synchronization because the carrier frequency offset estimates are unreliable.

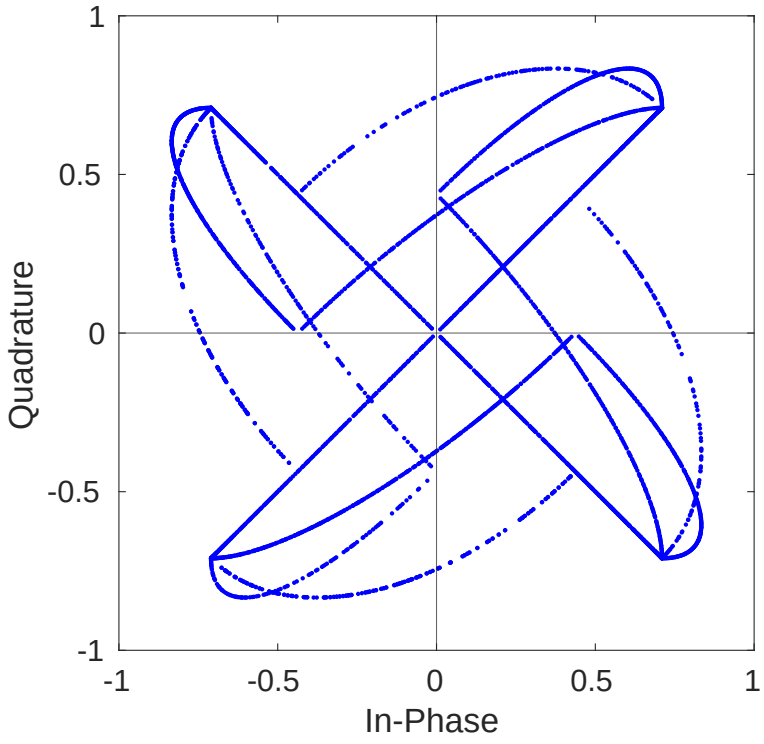

Figure 21: Constellation projection of a simulated ideal half-sine pulse shaped Zigbee burst with $f_q = 73$ Hz frequency offset. The Zigbee burst was composed of $N_{\mathrm{DS}} = 212$ Zigbee DSs. The corresponding O-QPSK symbols are represented by discrete dots in the constellation projection. The signal can be demodulated correctly since the O-QPSK symbols rotate less than $\phi = \pi/2$ radians during t_{burst}.

This limitation is consistent with the Cramer-Rao Lower Bound for QPSK signals [37, 38].

Overall, the simulation indicated that perfect demodulation with the O-QPSK receiver was achieved for SNR values of $\mathrm{E_s/N_0} = [5,\ 13]$ dB and that demodulation errors began to occur at $\mathrm{E_s/N_0} = 4.75$ dB. None of the bursts correctly demodulated for SNR values of $\mathrm{E_s/N_0} = [-12,\ -1.25]$ dB.

Furthermore, Figure 24 presents a bar graph for the average (mean) number of times from the $N_{\mathrm{SNR}} = 1000$ bursts that the estimated carrier frequency offset was less than the maximum frequency drift allowed. At low SNR values, the variance of the estimated frequency offset was typically large (shown in Figure 25) and resulted

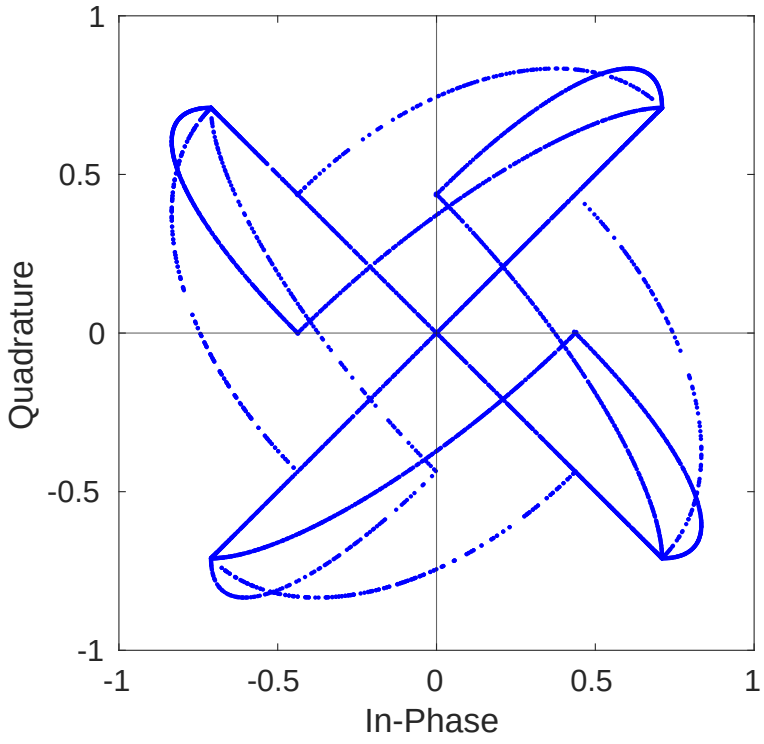

Figure 22: Constellation projection of a simulated ideal half-sine pulse shaped Zigbee burst with $f_q = 74$ Hz frequency offset. The Zigbee burst was composed of $N_{\text{DS}} = 212$ Zigbee DSs. The corresponding O-QPSK symbols are represented by discrete dots in the constellation projection. The signal can no longer be demodulated correctly since the O-QPSK symbols rotate more than $\phi = \pi/2$ radians during t_{burst}.

in the inability to demodulate the O-QPSK signal. Conversely, at high SNR values, the variance of the estimated frequency offset was small (shown in Figure 26) and allowed for the O-QPSK signal to be demodulated correctly.

Given that the value of f_q calculated in (46) is less than the $f_{q_{\max}}$ calculated in (45), the signal can be successfully demodulated in environments where the SNR is greater than or equal to $\text{E}_\text{s}/\text{N}_0 = 5$ dB. A theoretical constellation projection of a Zigbee burst with $f_q \approx 2.3842$ Hz in a simulated SNR of $\text{E}_\text{s}/\text{N}_0 = 20$ dB environment can be seen in Figure 27.

Since the minimum resolution for the carrier frequency offset is $f_q \approx 2.3842$ Hz, the frequency estimate $\hat{f}_c$ can be no less than the value of $\pm f_q$. After calculating the

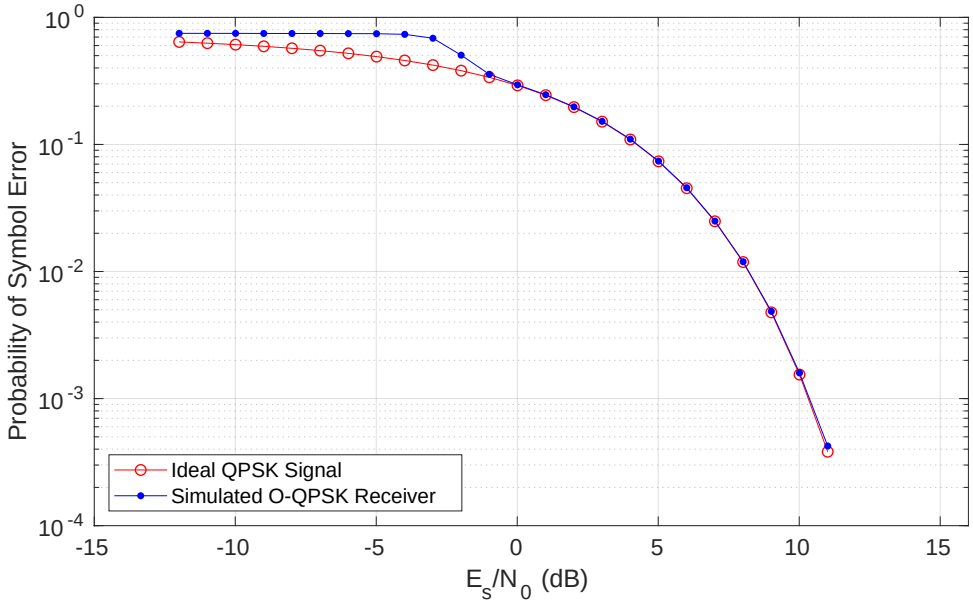

Figure 23: Probability of symbol error v.s. E_s/N_0 for an O-QPSK signal. Theoretical QPSK symbol error rates are represented by red circles and the simulated O-QPSK receiver symbol error rates are represented by blue points. The 95% Confidence Interval (CI) on each point is not explicitly represented in the plot since the marker sizes are larger than the CI bounds.

frequency offset estimate, the offset can then be removed from the collected signal by performing

$$\mathbf{s}_{\hat{f}_c \text{ removed}}(t) = \mathbf{s}(t) \exp\left(-j2\pi \hat{f}_c t\right). \tag{47}$$

After removing the frequency offset estimate, the C++ code performed cross-correlation between the collected signal and the complex conjugate of a locally generated Zigbee SHR reference signal. The cross-correlation searched a region of $N_{\text{xcorr}} = \pm$ 2000 lags from the sample that exceeded the energy threshold to align the collected signal to the start of the burst. The code determined the optimum burst starting point by choosing the cross-correlation lag that produced the largest magnitude. The code then reduced the collected signal to only contain $N_{DS} = 212$ Zigbee DSs from the cross-correlation determined start position.

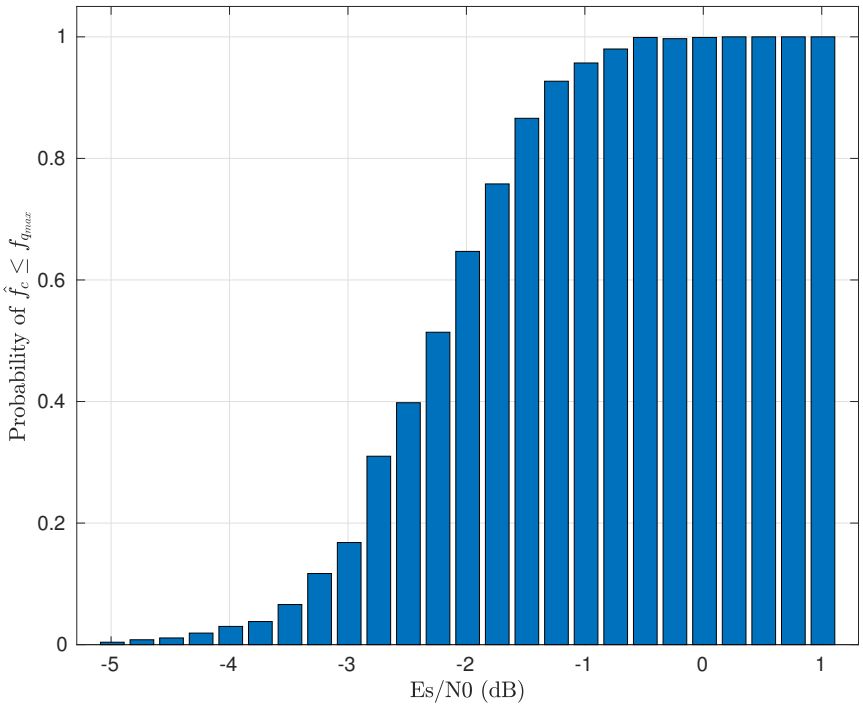

Figure 24: Probability that the estimated carrier frequency offset ($\hat{f}_c$) is less than or equal to the maximum frequency resolution ($f_{q_{\max}}$) for a Zigbee burst consisting of $N_{\text{DS burst}} = 212$ Zigbee DSs. Probability calculated using $N_{\text{SNR}} = 1000$ simulated Zigbee bursts at each $\text{E}_\text{s}/\text{N}_0$. SNR values ranged from $\text{E}_\text{s}/\text{N}_0 = [-5,\ 1]$ dB.

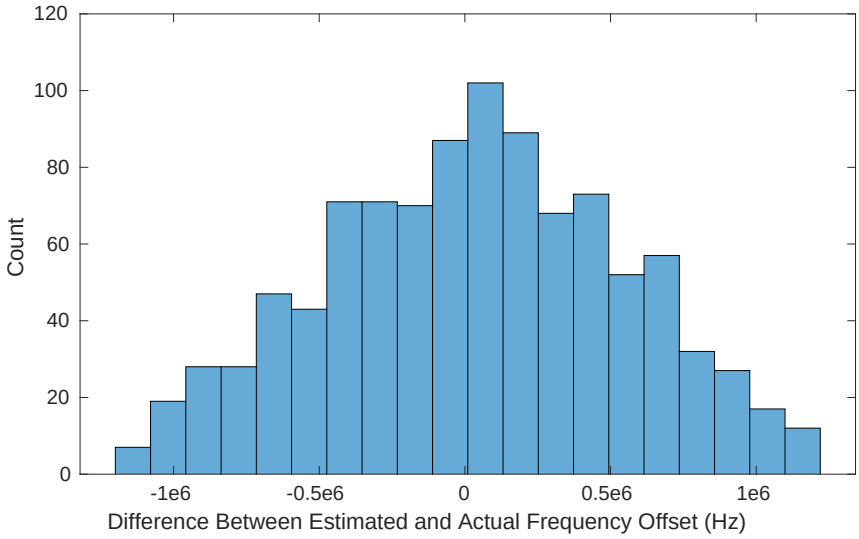

Figure 25: Histogram of difference between estimated and actual frequency offset at $\text{E}_\text{s}/\text{N}_0 = -5$ dB. For $N_{\text{SNR bursts}} = 1000$ bursts, the variance of the frequency difference is $\sigma^2_{-5\text{dB}} = 2.612 \times 10^{11}$.

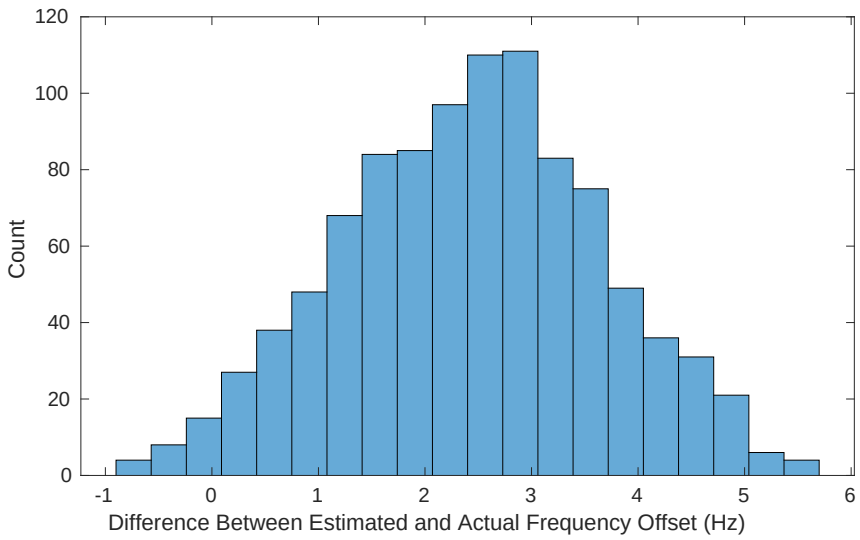

Figure 26: Histogram of difference between estimated and actual frequency offset at $E_s/N_0 = 13$ dB. For $N_{\text{SNR bursts}} = 1000$ bursts, the variance of the frequency difference is $\sigma^2_{13\text{dB}} = 1.4749$.

Next, the code estimated the phase offset by taking the dot product of the SHR of the collected burst with the previously generated Zigbee SHR reference signal. The dot product yields both a correlation and phase value as a measure of similarity between two signals. Therefore, the phase difference in radians is calculated by

$$\hat{\phi} = \text{angle}\left(\sum_n \mathbf{x}_{\text{align}}[n]\bar{\mathbf{y}}[n]\right) \tag{48}$$

where $\mathbf{x}_{\text{align}}[n]$ is the aligned burst, $\mathbf{y}[n]$ is the local reference signal, and $\hat{\phi}$ is the estimated phase offset. The phase offset estimate was then removed by

$$\mathbf{x}_{\hat{\phi}\text{ removed}}[n] = \mathbf{x}_{\text{align}}[n]\exp\left(-j\hat{\phi}\right). \tag{49}$$

Symbol boundaries for the burst were then estimated by incrementally shifting the signal from $\left\lceil -\frac{\text{samples per symbol}}{2(2)} \right\rceil$ to $\left\lceil \frac{\text{samples per symbol}}{2(2)} \right\rceil$ samples and performing a dot product with the locally generated Zigbee SHR reference signal. The shift that produces the dot product with the largest magnitude is the optimum sampling point

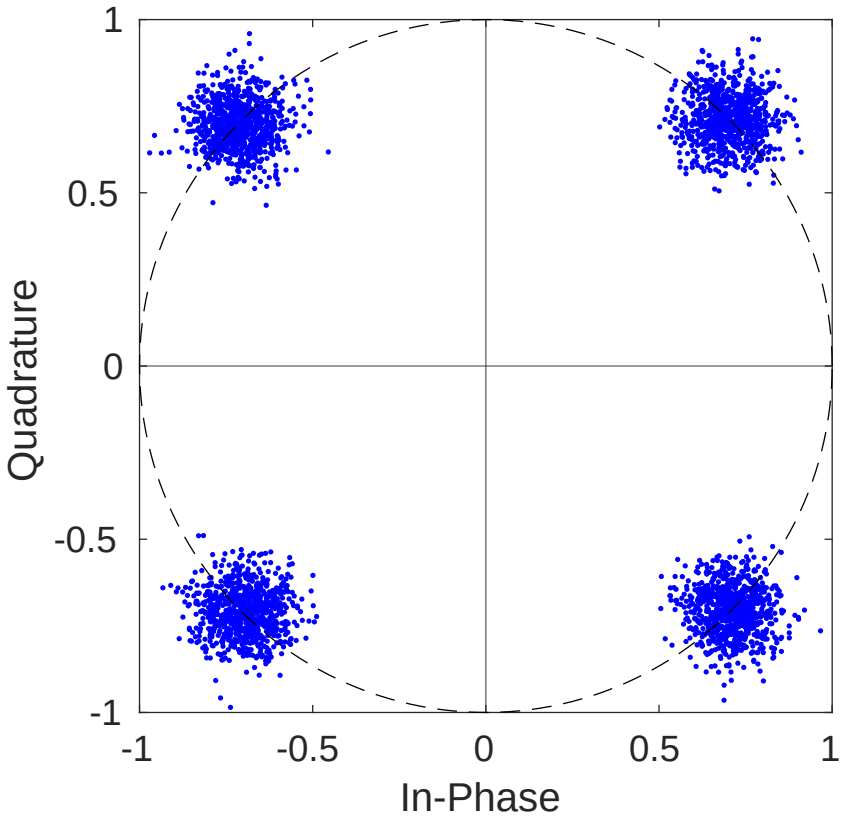

Figure 27: Constellation projection of simulated half-sine pulse shaped Zigbee burst with a frequency offset of $f_q \approx 2.3842$ Hz at $\mathrm{E_s/N_0} = 20$ dB.

for symbol boundary estimation. The code then conducts a circular shift of the signal to start at that position. For this research, the number of samples per symbol was

$$\begin{aligned} \text{samples per symbol} &= \frac{f_s}{f_{\text{O-QPSK Sym Rate}}} \\ &= \frac{10 \times 10^6 \text{ sps}}{10^6 \text{ O-QPSK Syms}/s} \\ &= 10 \ \frac{\text{samples}}{\text{O-QPSK Syms}}. \end{aligned} \tag{50}$$

Therefore, the shifts conducted for this experiment ranged from

$$\left\lceil \frac{-\text{samples per symbol}}{2(2)} \right\rceil = \left\lceil \frac{-10}{4} \right\rceil = -3, \tag{51}$$

$$\left\lceil \frac{\text{samples per symbol}}{2(2)} \right\rceil = \left\lceil \frac{10}{4} \right\rceil = 3. \tag{52}$$

3.6.4 O-QPSK Demodulation

The RF air monitor performed half-sine matched filtering to demodulate the collected burst by implementing an Integrate-and-Dump Filter (IDF). The matched filter output returned the constellation projections from which the transmitted chips, and O-QPSK symbols, could be estimated. The RF air monitor estimated the In-Phase (I)-channel chips by the sign of the real portion of the constellation projection. Similarly, the air monitor estimated the Quadrature (Q)-channel chips by the sign of the imaginary portion of the projection. A positive sign was mapped to a binary "1", and a negative sign was mapped to a binary "0". O-QPSK symbols were determined by grouping corresponding estimated I-channel and Q-channel chips.

The estimated chips then ascertained if the collected burst was a Zigbee transmission. The RF air monitor skipped the first $N_{\text{skip}} = 32$ chips to ensure the signal was in a stable region. Zigbee burst verification was dependant upon the following $N_{\text{keep}} = 94$ chips (three Zigbee DSs) in the signal.

Using the knowledge that the Zigbee preamble is eight Zigbee $DS_Z = 0$, the $N_{\text{keep}} = 94$ chips selected should also map to three $DS_Z = 0$. To verify this DS sequence, the $N_{\text{keep}} = 94$ selected chips were then converted into a non-return-to-zero form along with the Zigbee $DS_Z = 0$ chip mapping. The $N_{\text{keep}} = 94$ chips were then split into $N_{\text{sym}} = 3$ regions of $N_{\text{region}} = 32$ chips. Each chip region was then correlated with the $DS_Z = 0$ non-return-to-zero chip mapping to yield a correlation range of ρ_{min} = -32 to $\rho_{\text{max}} = 32$. If the two sequences are identical, the correlation result is

$\rho_{\max} = 32$, while if the sequences are completely opposite, the correlation result is $\rho_{\min} = -32$.

Due to the quasi-orthogonality of the Zigbee DS chip mapping (Table 1), a minimum of $N_{\text{diff}} = 12$ chips must be switched before a Zigbee DS could potentially map to an incorrect DS. The correlation values between Zigbee DSs are shown below in Table 5. A threshold value of $\rho_{\text{threshold}} = 30$ was chosen, such that each of the three-chip regions allowed a maximum of $N_{\text{error}} = 2$ incorrect chips in the sequence. Therefore, if only two chips are incorrect in the sequence, it was still the most likely Zigbee DS transmitted.

3.6.5 CB-DNA Fingerprints

CB-DNA fingerprints were generated within C++ by searching for the $N = 30$ conditional sub-clusters of interest shown in Tables 6 - 9. As shown in Figure 10,

Table 5: Correlation between Zigbee data symbols. Auto-correlation is shown in the main diagonal in table.

Zigbee Data Symbol Correlation Values																
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	32	0	-4	-8	-8	-8	-4	0	0	8	4	-8	-8	-8	4	8
1	0	32	0	-4	-8	-8	-8	-4	8	0	8	4	-8	-8	-8	4
2	-4	0	32	0	-4	-8	-8	-8	4	8	0	8	4	-8	-8	-8
3	-8	-4	0	32	0	-4	-8	-8	-8	4	8	0	8	4	-8	-8
4	-8	-8	-4	0	32	0	-4	-8	-8	-8	4	8	0	8	4	-8
5	-8	-8	-8	-4	0	32	0	-4	-8	-8	-8	4	8	0	8	4
6	-4	-8	-8	-8	-4	0	32	0	4	-8	-8	-8	4	8	0	8
7	0	-4	-8	-8	-8	-4	0	32	8	4	-8	-8	-8	4	8	0
8	0	8	4	-8	-8	-8	4	8	32	0	-4	-8	-8	-8	-4	0
9	8	0	8	4	-8	-8	-8	4	0	32	0	-4	-8	-8	-8	-4
10	4	8	0	8	4	-8	-8	-8	-4	0	32	0	-4	-8	-8	-8
11	-8	4	8	0	8	4	-8	-8	-8	-4	0	32	0	-4	-8	-8
12	-8	-8	4	8	0	8	4	-8	-8	-8	-4	0	32	0	-4	-8
13	-8	-8	-8	4	8	0	8	4	-8	-8	-8	-4	0	32	0	-4
14	4	-8	-8	-8	4	8	0	8	-4	-8	-8	-8	-4	0	32	0
15	8	4	-8	-8	-8	4	8	0	0	-4	-8	-8	-8	-4	0	32

these $N_{\text{likely}} = 30$ symbol transitions are the most likely occur within an arbitrary Zigbee burst. Utilizing the estimated O-QPSK symbols generated in the demodulation process, a search for all occurrences of the significant conditional sub-clusters within the collected Zigbee burst is conducted. The fingerprints are then generated by performing the statistical analysis on the "current" symbol within the baseband signal and stored in the fingerprint structure. This process is repeated for each of the $N_{\text{likely}} = 30$ conditional sub-clusters, which yielded a total of $N_{\text{features}_{CB}} = 270$ features as shown in (29) for each Zigbee burst.

Table 6: Conditional Sub-Clusters: Current O-QPSK Symbol $CS = 0$. Boxes with check-marks are transitions of interest for CB-DNA fingerprint generation.

Conditional Sub-Clusters Current O-QPSK Symbol = 0		**Next Symbol**			
		0	**1**	**2**	**3**
Previous Symbol	**0**		✓		✓
	1			✓	
	2	✓		✓	✓
	3	✓		✓	

Table 7: Conditional Sub-Clusters: Current O-QPSK Symbol $CS = 1$. Boxes with check-marks are transitions of interest for CB-DNA fingerprint generation.

Conditional Sub-Clusters Current O-QPSK Symbol = 1		**Next Symbol**			
		0	**1**	**2**	**3**
Previous Symbol	**0**				✓
	1	✓		✓	
	2		✓		✓
	3		✓	✓	✓

Table 8: Conditional Sub-Clusters: Current O-QPSK Symbol $CS = 2$. Boxes with check-marks are transitions of interest for CB-DNA fingerprint generation.

Conditional Sub-Clusters Current O-QPSK Symbol = 2		**Next Symbol**			
		0	**1**	**2**	**3**
Previous Symbol	**0**	✓	✓		✓
	1	✓	✓		
	2				
	3	✓			✓

Table 9: Conditional Sub-Clusters: Current O-QPSK Symbol $CS = 3$. Boxes with check-marks are transitions of interest for CB-DNA fingerprint generation.

Conditional Sub-Clusters Current O-QPSK Symbol = 3		**Next Symbol**			
		0	**1**	**2**	**3**
Previous Symbol	**0**	✓	✓		
	1	✓	✓	✓	
	2		✓	✓	
	3				

3.7 MDA Model Generation

A MATLAB® script created the Multiple Discriminant Analysis (MDA) $\mathbf{W}$ outside of the C++ environment. To generate the fingerprint structures for MDA training, the C++ environment processed $N_{\text{training}} = 500$ training bursts from each of the $N_{\text{cls}} = 5$ BladeRF devices. Prior to MDA, each training fingerprint was normalized to a Standard Normal distribution ($\sim N(0,1)$) using

$$\mu_{\text{tng}} = \left(\frac{1}{N_{\text{cls}} N_{\text{bursts}}}\right) \sum_{d=1}^{N_{\text{cls}}} \sum_{b=1}^{N_{\text{bursts}}} \mathbf{f}_{\text{tng}_{d,b}}, \tag{53}$$

$$\sigma_{\text{tng}} = \sqrt{\left(\frac{1}{N_{\text{cls}} N_{\text{bursts}} - 1}\right) \sum_{d=1}^{N_{\text{cls}}} \sum_{b=1}^{N_{\text{bursts}}} \left(\mathbf{f}_{\text{tng}_{d,b}} - \mu_{\text{tng}}\right)^2}, \tag{54}$$

$$\mathbf{z}_{\text{tng}} = \frac{\mathbf{f}_{\text{tng}} - \mu_{\text{tng}}}{\sigma_{\text{tng}}}, \tag{55}$$

where $\mathbf{f}_{\text{tng}_{d,b}}$ is the training fingerprint for burst b from device d, and $\mathbf{z}_{\text{tng}}$ is a standard normal fingerprint. Each fingerprint's mean (μ_{tng}) and standard deviation (σ_{tng}) were calculated for each of the $N_{\text{features}_{CB}} = 270$ features. These values were imported into the C++ environment to transform the future test fingerprints.

MATLAB® then passed the standard normal training fingerprint structure into the MDA script where $k = 5$ cross-fold validation [39] occurred. The cross-fold validation was accomplished by taking the training fingerprints and randomly sorting them into $N_{\text{k-fold}} = 5$ equal partitions. Next, $\mathbf{W}$s were then generated by performing MDA with $N_{\text{partitions}} = 4$ partitions at a time and using the remaining partition to validate the $\mathbf{W}$. The process created a total of $\binom{5}{4} = 5$ $\mathbf{W}$s during training, where the MATLAB® script selects the best performing $\mathbf{W}$ for implementation in the C++ environment. For each device, the location of the training fingerprint's mean within the projection space was also imported into C++ to calculate the Euclidean Distance (ED).

3.8 C++ Classifier

After completing training, the C++ classifier obtained fingerprints generated from a collected Zigbee burst and performed classification NRT using the training variables previously calculated. When receiving fingerprints from a test burst, the classifier first normalizes the fingerprints using

$$\mathbf{z}_{\text{tst}} = \frac{\mathbf{f}_{\text{tst}} - \mu_{\text{tng}}}{\sigma_{\text{tng}}} \tag{56}$$

where $\mathbf{f}_{\text{tst}}$ is the test fingerprint and $\mathbf{z}_{\text{tst}}$ is the normalized test fingerprint. Using the best performing training $\mathbf{W}$, the normalized test fingerprint is then projected into the training space by

$$\mathbf{f}_{\text{proj}} = \mathbf{W}_{\text{best}}^{T} \cdot \mathbf{z}_{\text{tst}} \tag{57}$$

such that $\mathbf{W}_{\text{best}}$ is the best performing $\mathbf{W}$ from the $k = 5$ cross-fold validation and $\mathbf{f}_{\text{proj}}$ is the location of the test fingerprint in the projection space. The ED to each device mean is then calculated using (31) for the projected test fingerprint. The device mean with the shortest ED from the test fingerprint is the estimate for the most likely transmitting device. The RF air monitor records the called device for every Zigbee test burst to create the CM.

IV. Results and Analysis

This chapter presents the experimental data collected for Near Real-Time (NRT) device classification using Constellation-Based Distinct Native Attribute (CB-DNA) fingerprints with a Multiple Discriminant Analysis/Euclidean Distance (MDA/ED) classifier. Section 4.1 discusses the performance of the CB-DNA air-monitor for $N_{\text{cls}} = 5$ like-model Software Defined Radios (SDRs) transmitting Zigbee bursts. Section 4.2 performs a runtime analysis to assess the NRT aspect of the air-monitor.

4.1 Air Monitor Test Results

Section 4.1.1 discusses using Multiple Discriminant Analysis (MDA) to develop a Projection Matrix ($\mathbf{W}$) from experimentally collected training fingerprints. Section 4.1.2 addresses Euclidean Distance (ED) and how classification is performed for the BladeRF devices. Finally, Section 4.1.3 presents the MDA/ED classification results for each of the BladeRF SDRs.

4.1.1 MDA Model Generation

Training data was composed of CB-DNA fingerprints for $N_{\text{training}} = 500$ bursts from each of the $N_{\text{cls}} = 5$ like-model BladeRF SDRs. MDA generated a $\mathbf{W}$ from the training data which projected fingerprints into a $N_{\text{dim}} = 4$-dimensional space. Figure 28 shows all of the two-dimensional representations of the projected training fingerprints. Furthermore, Figure 28 also presents a histogram of the fingerprint locations for each projection space dimension. Since visualizing spaces greater than 3-dimensions can be challenging, these subfigures provided a method of visually verifying if MDA achieved inter-class separation during the model generation process. For each subfigure, excluding the histograms, five distinct clusters are present. Each

cluster represents the training fingerprints for a BladeRF device. If the MDA feature selection process had been poor, multiple classes (clusters) would overlap within the subfigures (arbitrary example shown in Figure 29). As mentioned in Section 2.5, MDA does not provide insight into fingerprint feature relevance. Therefore, this visual inspection method validated that class separation was achievable with the provided training fingerprints prior to performing classification.

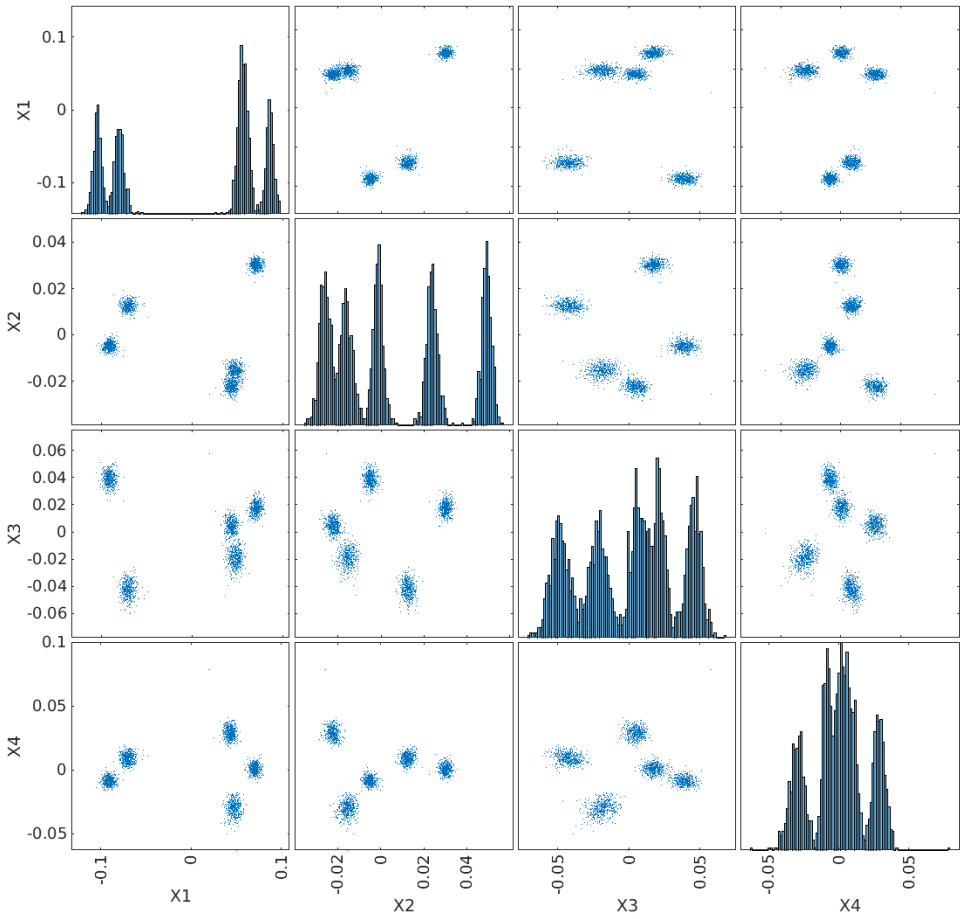

Figure 28: Training CB-DNA fingerprint projections for $N_{\text{cls}} = 5$ class scenario confined to two-dimensions. Main diagonal contains the histogram of fingerprint locations along the projection axis. Subfigures indicate that MDA achieved class separation with the provided CB-DNA fingerprints.

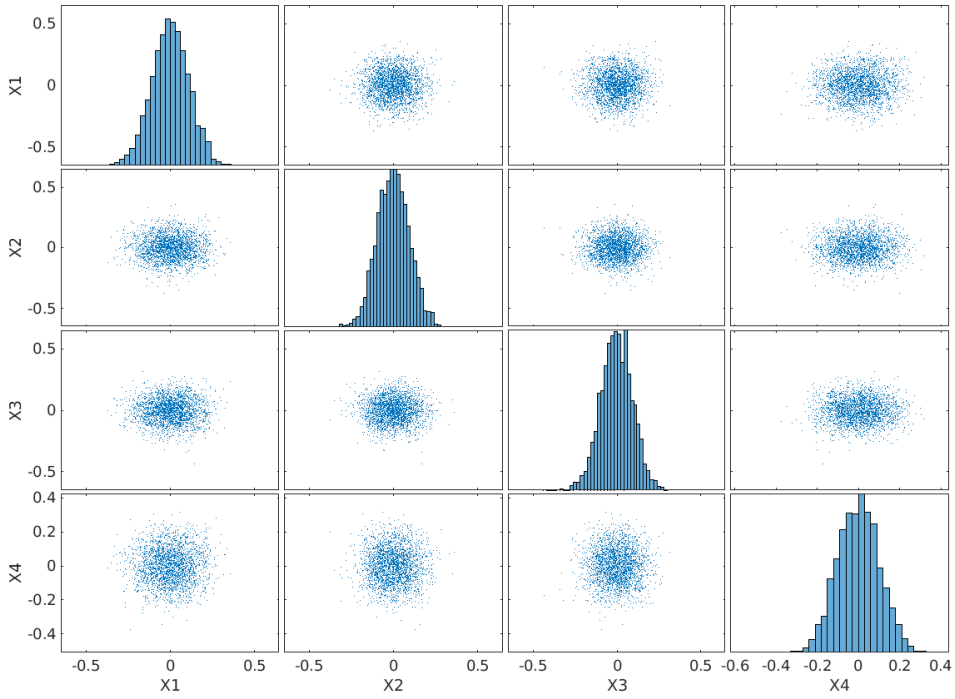

Figure 29: Simulated training CB-DNA fingerprint projections for $N_{\text{cls}} = 5$ class scenario confined to two-dimensions. Main diagonal contains the histogram of fingerprint locations along the projection axis. Subfigures show that all five classes overlap which indicates that MDA was not able to achieve class separation with the provided simulated fingerprints. Classification performance with this $\mathbf{W}$ would likely be poor.

4.1.2 Euclidean Distance Classifier

The class mean for each device within the projection space was of interest to this research. The training CB-DNA fingerprints yielded the class means shown in Table 10 within the projection space. For the test scenario, projected fingerprints calculate the ED to each of the training class means using (31). The class with the shortest calculated ED is classified as the "most likely" transmitting device.

Table 10: Projected class means for $N_{\text{cls}} = 5$ like-model BladeRF devices

Training Fingerprint Class Means				
	Dimension X1	**Dimension X2**	**Dimension X3**	**Dimension X4**
BladeRF1	-0.0704	0.0125	-0.0424	0.0091
BladeRF2	0.0431	-0.0224	0.0047	0.0286
BladeRF3	0.0704	0.0302	0.0177	0.0008
BladeRF4	-0.0908	-0.0050	0.0389	-0.0086
BladeRF5	0.0476	-0.0154	-0.0190	-0.0299

4.1.3 Air Monitor Classification Performance

For testing, each of the $N_{\text{cls}} = 5$ like-model devices transmitted $N_{\text{bursts}} = 500$ bursts. The **W** created in Section 4.1.1 projected each of the test fingerprints into the projection space. The classifier then calculated the ED to all class means from the projected test fingerprint. Tables 11 and 12 present the results of the air monitor test scenario in two different Confusion Matrix (CM) formats. For Table 11, each row is the number of total transmissions for the specified BladeRF, and the columns indicate to which device the burst was classified. Table 12 presents the same information in Average Cross-Class Percent Correct Classification (%C), instead of discrete bursts, along with the classifier's best and worst performance.

From the results, it is apparent that all of the devices achieved high classification performance for the test scenario. Of note, BladeRF4 achieved perfect classification, which was the top-performing device tested. Conversely, BladeRF1 achieved the lowest classification rate but still reached Average Percent Correct Classification ($\%C_{\text{cls}}$)= 97.6%. Taking the total number of correctly classified devices from the test scenario yielded a %C= 99.24% performance for the MDA/ED classifier.

Table 11: Confusion matrix of NRT discrimination test results for $N_{\text{cls}} = 5$ like-model BladeRF devices. SNR was $E_s/N_0 \approx 31$ dB for the collection.

Confusion Matrix		Called					Total Bursts Transmitted
		BladeRF1	BladeRF2	BladeRF3	BladeRF4	BladeRF5	
Actual	BladeRF1	**488**	0	0	12	0	**500**
	BladeRF2	2	**498**	0	0	0	**500**
	BladeRF3	0	0	**498**	0	2	**500**
	BladeRF4	0	0	0	**500**	0	**500**
	BladeRF5	1	1	1	0	**497**	**500**
Total Bursts Classified		**491**	**499**	**499**	**512**	**499**	**2500**

Table 12: Confusion matrix of NRT %C for $N_{\text{cls}} = 5$ like-model BladeRF devices. SNR was $E_s/N_0 \approx 31$ dB for the collection.

Confusion Matrix		Called				
		BladeRF1	BladeRF2	BladeRF3	BladeRF4	BladeRF5
Actual	BladeRF1	**97.6%**	-	-	2.4%	-
	BladeRF2	0.4%	**99.6%**	-	-	-
	BladeRF3	-	-	**99.6%**	-	0.4%
	BladeRF4	-	-	-	**100%**	-
	BladeRF5	0.2%	0.2%	0.2%	-	**99.4%**
Min $\%C_{\text{cls}}$		**97.6%**				
Max $\%C_{\text{cls}}$		**100%**				
Overall %C		**99.24%**				

4.2 Timing Analysis

This research also conducted a runtime analysis for each of the C++ threads that processed SDR data using the hardware configuration listed in Table 3. This analysis assessed the NRT capability of the air monitor by calculating the average runtime from burst detection to classification. The threads of interest are burst detection, signal demodulation/fingerprint generation, and classification as shown in Figure 30. The timing analysis utilized a total of $N_{\text{burst}} = 1000$ bursts from BladeRF4 to generate the samples used in calculating the average runtime for each thread. Additionally, a 95% Confidence Interval (CI) on the average runtime for each thread was calculated.

For the burst detection thread, two specific time durations were of interest to the research. The first duration of interest was the time to complete all of the necessary filtering and search of the signal for exceedance of the energy level threshold. The

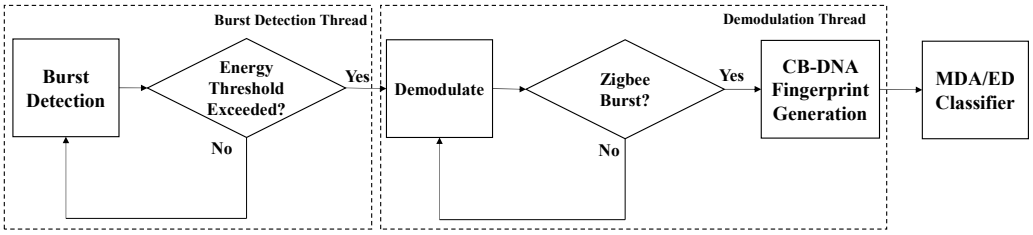

Figure 30: Flow diagram of threads that process SDR data

second time duration measurement of interest was the overall thread execution time to perform the core tasks listed above and all of the additional overhead code. Of note, the burst detection thread acted as a state machine that operated in two different modes (Figure 31). One state actively searched for an energy threshold exceedance in the SDR samples while the other state built the full Zigbee transmission after burst detection.

During the time to transmit and classify $N_{\text{burst}} = 1000$ Zigbee bursts, the Radio Frequency (RF) air monitor continually processed $t_{\text{block}} = 1$ mSec of data at a time. For every instance that the portion of code executed, the time duration was recorded as a sample. Overall, the search portion of the burst detection code collected a total of $N_{\text{Burst Det}} = 112334$ samples, while the overall thread execution produced

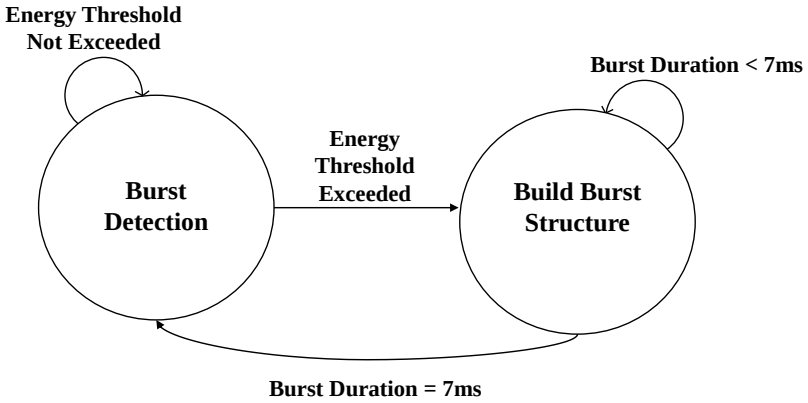

Figure 31: State machine of burst detection thread

$N_{\text{Burst Det Thread}} = 117734$ samples. Figures 32 and 33 show the histograms for both data sets, where the measured time is in microseconds. From Figure 32, the histogram of the burst detection process indicates that the distribution is right-skewed, while Figure 33 implies that the distribution is bimodal. The bimodal distribution is due to the two distinct operating states of the burst detection state machine.

The signal demodulation and CB-DNA fingerprint generation thread contained three runtimes of interest:

- The time duration to perform the Fast Fourier Transform (FFT) utilized to estimate the carrier frequency offset.
- The time required to estimate and mitigate frequency and phase offsets, demodulate the signal, and generate CB-DNA fingerprints.
- The overall runtime of the thread to include the processes listed above and the

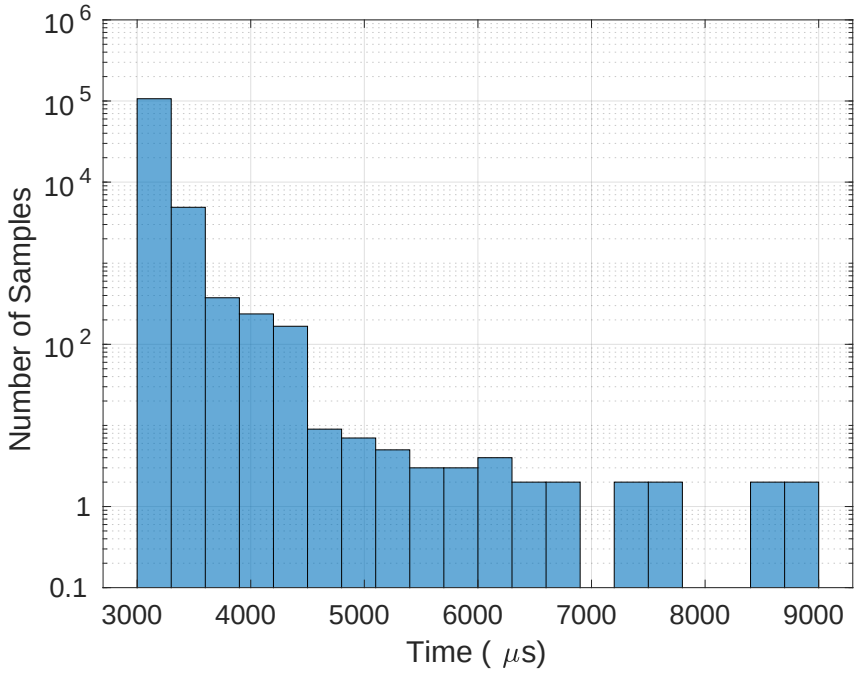

Figure 32: Histogram of timing results for burst detection process in μs. The distribution appears to be non-normal and specifically right-skewed.

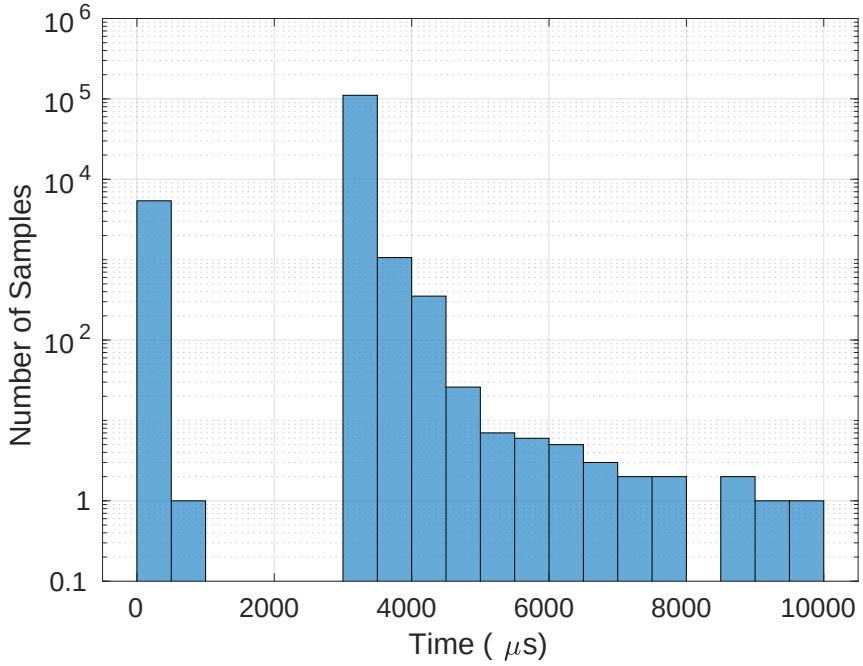

Figure 33: Histogram of timing results for burst detection thread in μs. The distribution appears to be non-normal and specifically bimodal.

additional overhead code.

Figure 34 shows the flow diagram of the process.

From the $N_{\text{burst}} = 1000$ bursts, a total of $N_{\text{SP and FP}} = 1000$ samples were col-

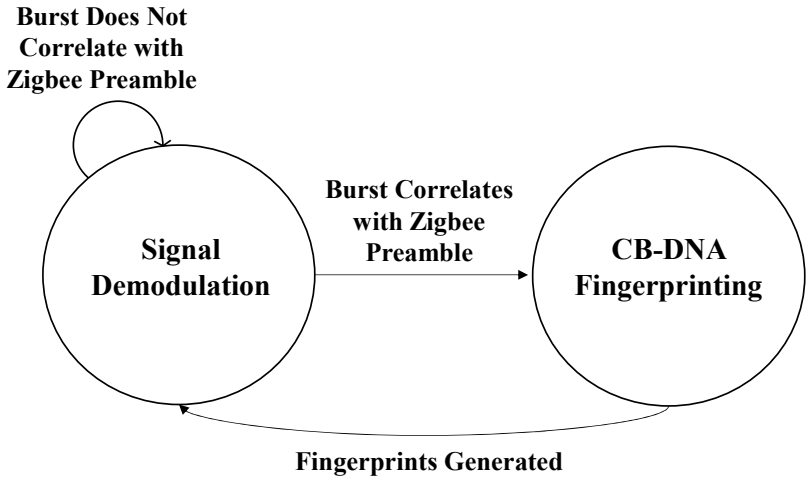

Figure 34: Flow diagram of signal demodulation and fingerprint generation thread

lected for each of the three signal demodulation and CB-DNA fingerprint generation runtimes of interest. Figure 35 shows the histogram of the FFT runtime which is right-skewed. Similarly, the histograms for both demodulation/fingerprint calculations (Figure 36) and overall thread (Figure 37) runtimes indicate that the distributions are right-skewed.

For the classification thread, two runtimes were of interest. First was the time duration required to classify CB-DNA fingerprints, and the other was the overall classification thread runtime.

From the $N_{\text{burst}} = 1000$ bursts, a total of $N_{\text{Class}} = 1000$ samples were collected for each of the classification desired runtimes. Figure 38 shows the histogram for the classification runtime and Figure 39 shows the overall thread runtime. Both histograms of the sample distributions indicate that the population is non-normal and specifically right-skewed.

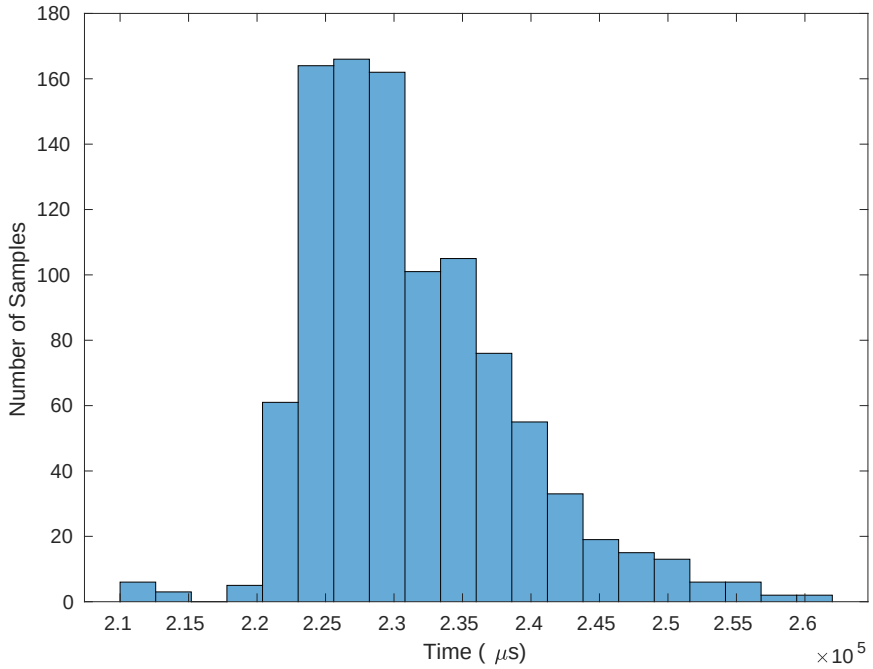

Figure 35: Histogram of timing results for FFT process in μs. The distribution appears to be non-normal and specifically right-skewed.

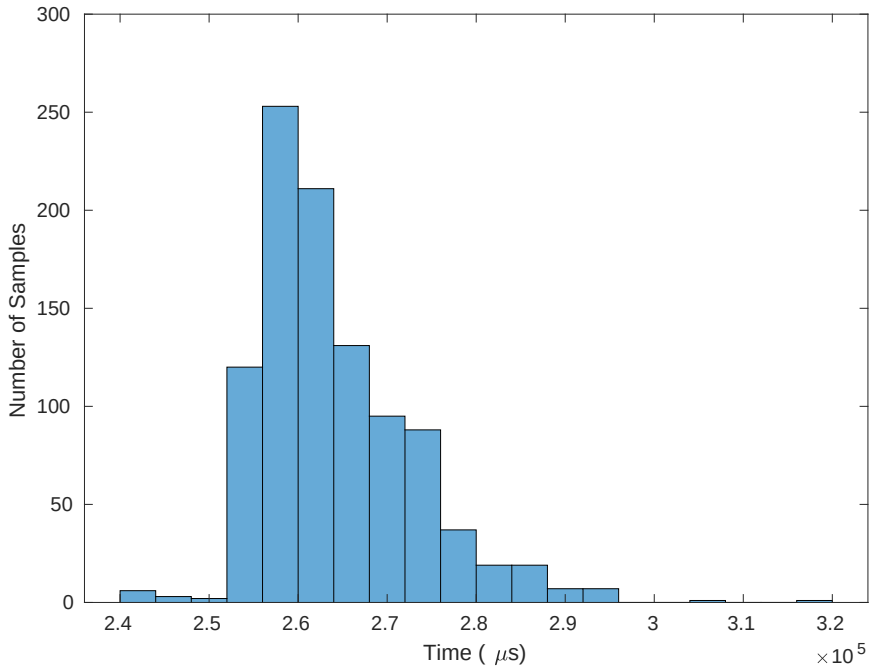

Figure 36: Histogram of timing results for demodulation/fingerprinting operations in μs. The distribution appears to be non-normal and specifically right-skewed.

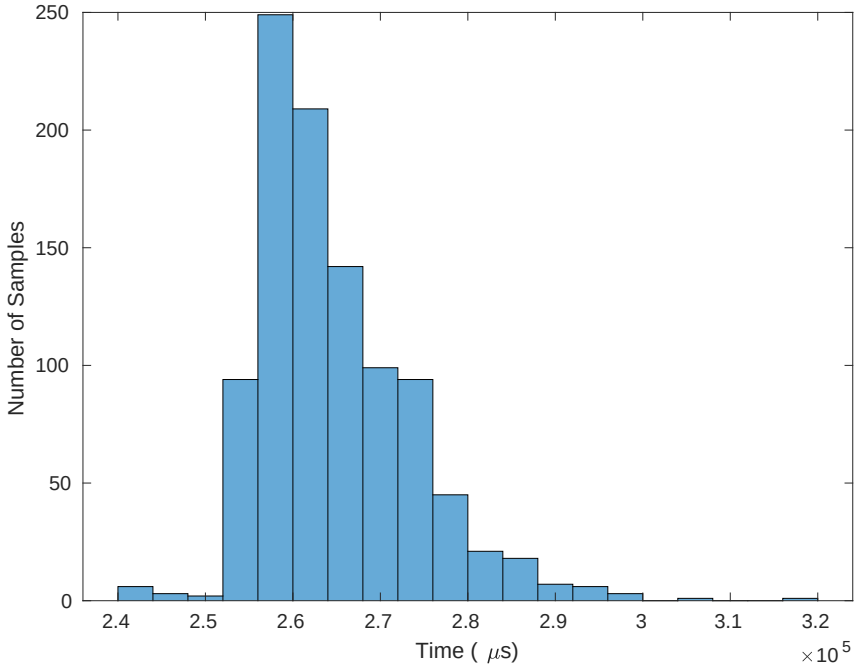

Figure 37: Histogram of timing results for signal demodulation/fingerprint generation thread in μs. The distribution appears to be non-normal and specifically right-skewed.

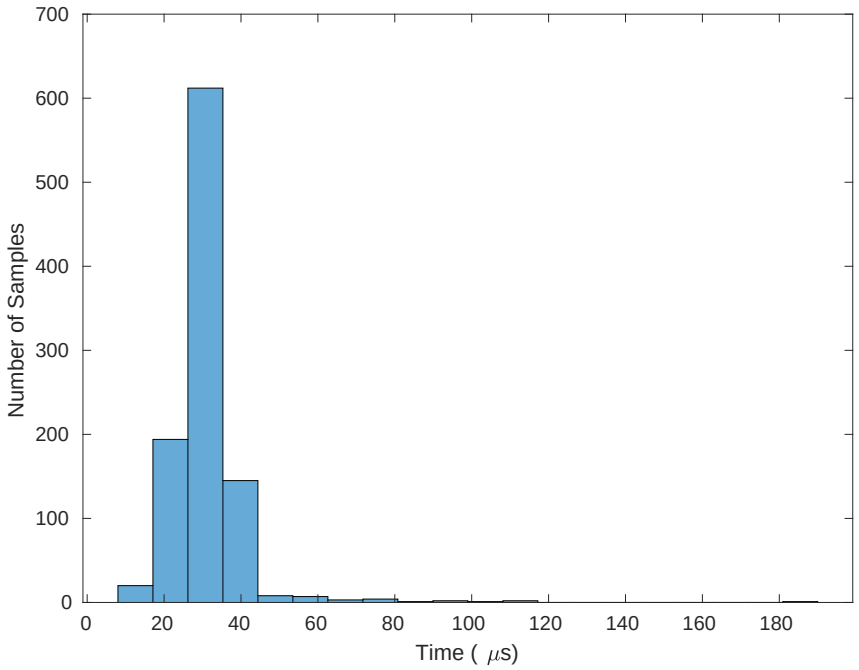

Figure 38: Histogram of timing results for MDA/ED classification process in μs. The distribution appears to be non-normal and specifically right-skewed.

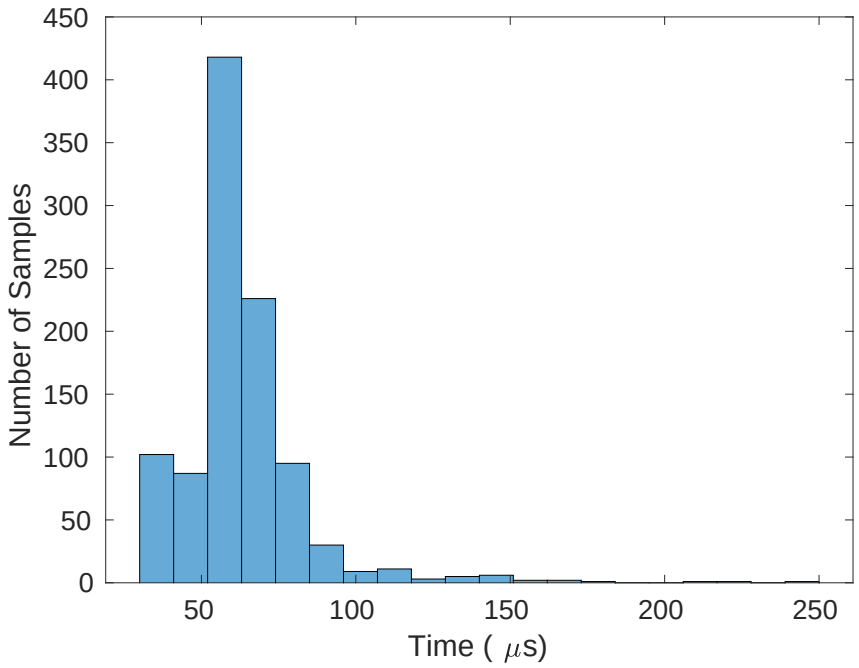

Figure 39: Histogram of timing results for MDA/ED classification thread in μs. The distribution appears to be non-normal and specifically right-skewed.

Since all of the collected sample distributions were non-normal, two different non-parametric statistical methods provided insight into the collected data. The first statistical method employed verified that the number of collected samples provided an accurate representation of the population through the use of tolerance limits. To determine a confidence metric for two-sided nonparametric tolerance limits, [40] contributed

$$\begin{aligned} & P\left[Y_{(1)}, Y_{(n)} \text{ covers at least } \delta \text{ of the population}\right] \\ & = 1 - \alpha \\ & = 1 - n\delta^{(n-1)} + (n-1)\delta^n \end{aligned} \tag{58}$$

such that $Y_{(1)}$ and $Y_{(n)}$ are the minimum and maximum values in the sample size of n, and δ is the percentage of the population. Therefore, with the sample sizes collected for each of the seven different runtime scenarios, the confidence that the collected samples represent $\delta = 0.995$ of the population is determined by employing (58) for each of the three unique sample values:

$$\begin{aligned} & P\left[\text{Burst Detection Process Samples}\right] \\ & = 1 - (112334)(0.995)^{(112334-1)} + (112334-1)\left(0.995\right)^{112334} \\ & \approx 1, \end{aligned} \tag{59}$$

$$\begin{aligned} & P\left[\text{Burst Detection Thread Samples}\right] \\ & = 1 - (117734)(0.995)^{(117734-1)} + (117734-1)\left(0.995\right)^{117734} \\ & \approx 1, \end{aligned} \tag{60}$$

$$
\begin{aligned}
&P\left[\text{Other Scenario Samples}\right] \\
&\quad = 1 - (1000)(0.995)^{(1000-1)} + (1000-1)\,(0.995)^{1000} \\
&\quad \approx 0.9599.
\end{aligned}
\tag{61}
$$

Due to the large number of samples for each runtime scenario, the experimental collections were good representations of their respective populations. The second non-parametric method employed was Bootstrap, which estimated a 95% CI on each of the average runtimes since the distributions were asymmetric. Bootstrap is a computer-based simulation method for statistical inference based on collected samples [41]. In essence, the collected samples compose a database for the Bootstrap method to re-sample with replacement. The resampling continues until achieving the same number of elements as the original collection to create a Bootstrap replicate. This approach repeats until producing the desired amount of Bootstrap replicates for analysis.

This analysis utilized $N_{\text{bootstrap}} = 10000$ randomly generated Bootstrap replicates for each of the test scenario times. For each of the scenarios, the worst-case average runtime metric was identified and calculated. For right-skewed distributions, the worst-case average time was empirically found to be the sample mean since the value was higher than the median value. For the bimodal distribution, the calculated sample mean represented an unachievable runtime value. The sample mean was a distorted statistic due to the two separate peaks occurring within the data. The sample mean ultimately was a weighted average between the two distribution peaks and was not an accurate representation of the runtime data. Additionally, the sample median failed to produce any variability amongst the Bootstrap replicates. Therefore, for this research, the sample mode provided the worst-case run time for the bimodal distribution since the runtime value was more accurate than the calculated sample

mean value and also provided a dynamic range.

The Bootstrap method then sorted the computed worst-case averages in ascending order for each test scenario. The values located in the desired percentile locations determined the confidence interval bounds. Given the $N_{\text{bootstrap}} = 10000$ Bootstrap replicates, the desired percentile locations for a 95% confidence interval are $m^*_{0.025 \text{ percentile}} = (10000 \cdot 0.025) = 250$ and $m^*_{0.975 \text{ percentile}} = (10000 \cdot 0.975) = 9750$ within the sorted values.

From the Bootstrap analysis, the average runtime for the burst detection calculations with a 95% CI was $\mu_{\text{Burst Det}} = [3.1931 \times 10^{-3},\ 3.1943 \times 10^{-3}]$ Sec while the burst detection thread runtime was $\mu_{\text{Burst Det Thread}} = [3.1950 \times 10^{-3},\ 3.1990 \times 10^{-3}]$ Sec. For the demodulation thread, the average runtime for the FFT was $\mu_{\text{FFT}} = [0.2307,\ 0.2316]$ Sec, while runtime to demodulate the signal and generate fingerprints was $\mu_{\text{Demod and FP}} = [0.2636,\ 0.2647]$ Sec. The overall thread runtime required $\mu_{\text{Demod and FP Thread}} = [0.2641,\ 0.2652]$ Sec. Finally, BladeRF classification average runtime was $\mu_{\text{Class}} = [3.0502 \times 10^{-5},\ 3.1780 \times 10^{-5}]$ Sec and the overall thread runtime was $\mu_{\text{Class Thread}} = [6.1825 \times 10^{-5},\ 6.4303 \times 10^{-5}]$ Sec. Table 13 contains all of the runtime averages along with the corresponding 95% CIs.

Therefore, using the upper bounds of the 95% CI for each of the three threads, the worst-case average runtime from burst detection to classification was $t_{\text{average runtime}} \approx 0.2684$ Sec. Of note, the FFT utilized to estimate the carrier frequency offset comprises $\sim 86\%$ of the total time to perform the NRT process. The average runtime resulted in classifying slightly less than $N_{\text{classify}} = 4$ bursts per second. Due to low-power design constraints, ZigBee devices regularly only transmit Medium Access Control (MAC) packets once every few seconds. An example being the Zigbee Cluster Library (ZCL) which is a standard tool employed when building Zigbee applications with cluster functionality. For ZCL, default polling times for packet transmissions

range from $t_{\text{short poll}} = 0.5$ Sec to $t_{\text{fast poll}} = 10$ Sec to extend the battery life of devices [42]. Therefore, the air monitor's NRT performance is acceptable for Zigbee applications.

Table 13: Average runtimes in seconds for $N_{\text{scenarios}} = 7$ different components of interest

Average Runtimes of Interest			
Runtimes (Sec)	**95% CI Lower Bound**	**Average**	**95% CI Upper Bound**
Burst Detection: Calculations (Mean)	3.1931×10^{-3}	$\mathbf{3.1937\times10^{-3}}$	3.1943×10^{-3}
Burst Detection: Thread (Mode)	3.1950×10^{-3}	$\mathbf{3.1990\times10^{-3}}$	3.1990×10^{-3}
Demodulation: FFT (Mean)	0.2307	**0.2311**	0.2316
Demodulation: Calculations (Mean)	0.2636	**0.2641**	0.2647
Demodulation: Thread (Mean)	0.2641	**0.2647**	0.2652
Classification: Calculations (Mean)	3.0502×10^{-5}	$\mathbf{3.1101\times10^{-5}}$	3.1780×10^{-5}
Classification: Thread (Mean)	6.1825×10^{-5}	$\mathbf{6.3043\times10^{-5}}$	6.4303×10^{-5}

V. Conclusions

As the entry barrier continues to lower for Internet of Things (IoT) applications, the number of Low-Rate Wireless Personal Area Networks (LR-WPAN) devices and users has significantly grown for both commercial and military purposes with a projected 1 billion annual shipments of Institute of Electrical and Electronics Engineers (IEEE) 802.15.4 standard devices by 2024 [1]. Specifically, the Zigbee wireless protocol has seen significant adoption due to its low-cost, low-power, and mesh network applications. Typical implementations of Zigbee devices range from remote sensors to automation system controllers.

However, these devices present an expanded attack surface for Industrial Control Systems (ICS) applications due to security vulnerabilities. Currently, open-source tools enable malicious users to imitate authentic network devices by falsifying bit-level credentials allowing them to gain unauthorized access to the network. Ultimately, unauthorized network access could lead to the loss of sensitive network information or even potential sabotage through the dissemination of false information. Through the use of Distinct Native Attribute (DNA) fingerprints, network security is enhanced with Physical Layer (PHY) device characteristics augmenting the bit-level authentication process.

5.1 Results Summary

This research showed that device discrimination using Constellation-Based Distinct Native Attribute (CB-DNA) fingerprinting is possible in Near Real-Time (NRT). The C++ environment facilitated the use of tools NRT that are traditionally performed in post processing.

Both training and testing collections required the use of a controlled environment

due to Radio Frequency Interference (RFI). This environment guaranteed that the training fingerprints generated were of the desired devices and not spuriously collected transmissions.

For a $N_{\text{cls}} = 5$ like-model device test scenario, a Multiple Discriminant Analysis/Euclidean Distance (MDA/ED) classifier was demonstrated for the first time using CB-DNA fingerprints. Overall, the classifier was able to achieve an Average Cross-Class Percent Correct Classification (%C) = 99.24% during testing. The worst-case average runtime from burst detection to classification was $t_{\text{runtime}} \approx 0.2684$ Sec. The calculated runtime allowed for NRT device classification since normal Zigbee cluster applications typically only transmit a burst every $t_{\text{short poll}} = 0.5$ Sec to $t_{\text{fast poll}} = 10$ Sec [42] to limit power consumption.

5.2 Research Contribution

As previously identified in Section 3.2, the goal of this research was to develop an air-monitor to classify received Zigbee bursts. Classification of CB-DNA fingerprints employed the use of a MDA/ED classifier to determine the "most likely" transmitting device.

The results presented in this research were consistent with previous NRT experiments conducted using Radio Frequency Distinct Native Attribute (RF-DNA) fingerprints and an Multiple Discriminant Analysis (MDA)/Maximum Likelihood (ML) classifier [18]. However, the average runtime between conducted experiments varied significantly with CB-DNA being the slower approach. The runtime difference stems from the fact that CB-DNA enables the utilization of the entire received burst for fingerprint generation by demodulating the signal. In comparison, the previously implemented RF-DNA approach only uses the Sychronization Header Region (SHR) of the burst and calculates fingerprints based on the instantaneous amplitude, phase,

and frequency responses.

Overall, the first research hypothesis presented in Section 3.3 regarding (1) NRT signal demodulation of IEEE 802.15.4 standard devices to create CB-DNA fingerprint was validated. Additionally, experimental collections confirmed the second hypothesis of (2) NRT device classification performance being consistent with previous NRT RF-DNA efforts [18]. Therefore, this research proved that NRT device discrimination utilizing CB-DNA fingerprints is achievable and can enhance network security for Zigbee devices using PHY characteristics.

5.3 Future Work

This research provided a method of closing the gap between data collection and analysis for device discrimination to improve network security. Specifically, future work in the field should examine:

- Reducing the computational complexity of carrier frequency offset estimation. Currently, the Fast Fourier Transform (FFT) implemented accounts for approximately 86% of the average runtime due to creating and searching a $N_{\text{FFT}} = 2^{20}$ problem space for every detected burst. A potential solution would be implementing a Phase-Locked Loop to track the frequency offset.

- Expanding the demodulation object to encompass the entire IEEE 802.15.4 standard. Specifically, incorporating the ability to change to different frequency bands, demodulate Binary Phase Shift Keying (BPSK) signals, and implementing a root-raised cosine pulse shaping filter (demonstrations in this work are based on half-sine pulse shaping). These endeavors would expand device discrimination capabilities to the full operating range of the Zigbee protocol worldwide.

- Implementing a more robust classifier to reduce the number of required fingerprint features. Specifically, Random Forest (RndF) or Generalized Relevance Learning Vector Quantization-Improved (GRLVQI) could provide insight into fingerprint feature relevance. These results could then drive a reduction in the number of generated fingerprints for a MDA classifier.

- Expand the device discrimination process beyond classification to include device Identification (ID) verification (one device vs one device). Calculated decision boundaries for each class from the training fingerprints could perform device verification. The verification results could directly drive the input to an Intrusion Detection System (IDS)/Intrusion Prevention System (IPS) tool.

- Finally, updating the classifier to check the Signal-to-Noise Ratio (SNR) of the collected burst. The classifier could then consider the Projection Matrix ($\mathbf{W}$) associated with the closest SNR value to improve classification in an operational environment.

5.4 Summary

Using CB-DNA, the developed Radio Frequency (RF) air monitor discriminates known Zigbee devices with high accuracy based on their PHY characteristics. Employing a PHY authentication method further enhances the bit-level verification accomplished natively within the protocol. Currently, the output of the air-monitor could provide input into an IDS system to determine the authenticity of network devices. This research directly supports the Department of Defense's ability to enhance security for LR-WPAN systems from rogue devices and malicious actors.

Bibliography

1. Research and Markets, "802.15.4 IoT Markets: A Market Dynamics Report." Market Report, ID:4807854. 2019.

2. J. Wright, R. Speers, and R. Melgares, "KillBee," Accessed: Dec. 10, 2019. [Online]. Available: https://github.com/riverloopsec/killerbee.

3. Department of Defense, "Summary: DoD Cyber Strategy," Washington, DC. 2018.

4. S. Farahani, *ZigBee Wireless Networks and Transceivers*. Burlington, MA, USA: Newness, 2008.

5. B. Ramsey, B. Mullins, W. Lowder, and R. Speers, "Sharpening the Stinger: Tuning KillerBee for Critical Infrastructure Warwalking," in *IEEE Military Communications Conference (MILCOM)*, pp. 104–109, 2014.

6. D. Gislason, *ZigBee Wireless Networking*. Burlington, MA, USA: Newness, 2008.

7. "IEEE Standard for Low-Rate Wireless Networks," *IEEE Standard 802.15.4-2015*, pp. 1–708, 2015.

8. C. Talbot, M. Temple, and T. Carbino, "Securing Insteon Home Automation Systems Using Radio Frequency Distinct Native Attributes (RF-DNA) Fingerprints," in *12th International Conference on Cyber Warfare and Security (ICCWS)*, pp. 497–508, 2017.

9. C. Talbot, M. Temple, T. Carbino, and A. Betances, "Detecting Rogue Attacks on Commercial Wireless Insteon Home Automation Systems," *Elsevier Computers & Security*, vol. 74, pp. 296–307, 2018.

10. N. Chiraratti, "Software Defined Radio Device Discrimination Using Chip Shape-Distinct Native Attribute Features," M.S. thesis, AFIT-ENG-MS-19-M-018, Air Force Institute of Technology, WPAFB, OH, 2019.

11. C. Coon, "Comparative Analysis of RF Emission Based Fingerprinting Techniques for ZigBee Device Classification," M.S. thesis, AFIT-ENG-MS-17-M-017, Air Force Institute of Technology, WPAFB, OH, 2017.

12. S. O'Neil and S. Stone, "Determining Authenticity of Mixed-Signal Devices Using Unintentional Radio Frequency (RF) Emissions," in *IEEE National Aerospace and Electronics Conference (NAECON)*, pp. 478–481, 2016.

13. W. Cobb, E. Laspe, R. Baldwin, M. Temple, and Y. Kim, "Intrinsic Physical-Layer Authentication of Integrated Circuits," *IEEE Transactions. on Information Forensics and Security*, vol. 7, no. 1, pp. 14–24, 2011.

14. W. Cobb, E. Garcia, M. Temple, R. Baldwin, and Y. Kim, "Physical Layer Identification of Embedded Devices Using RF-DNA Fingerprinting," in *IEEE Military Communications Conference (MILCOM)*, pp. 2168–2173, 2010.

15. R. Deppensmith and S. Stone, "Optimized Fingerprint Generation Using Unintentional Emission Radio-Frequency District Native Attributes (RF-DNA)," in *IEEE National Aerospace and Electronics Conference (NAECON)*, pp. 327–330, 2014.

16. M. Lukacs, P. Collins, and M. Temple, "Classification Performance Using 'RF-DNA' Fingerprints of Ultra-Wideband Noise Waveforms," *IEEE Electronics Letters*, vol. 51, no. 10, pp. 787–789, 2015.

17. M. Lukacs, P. Collins, and M. Temple, "Device Identification Using Active Noise Interrogation and RF-DNA "Fingerprinting" for Non-Destructive Amplifier Ac-

ceptance Testing," in *IEEE 17th Annual Wireless and Microwave Technology Conference (WAMICON)*, pp. 1–6, 2016.

18. F. Cruz, "Near Real-Time RF-DNA Fingerprinting for ZigBee Devices Using Software Defined Radios," M.S. thesis, AFIT-ENG-MS-19-M-021, Air Force Institute of Technology, WPAFB, OH, 2019.

19. T. Carbino, M. Temple, and T. Bihl, "Ethernet Card Discrimination Using Unintentional Cable Emissions and Constellation-Based Fingerprinting," in *IEEE International Conference on Computing, Networking and Communications (ICNC)*, pp. 369–373, 2015.

20. T. Carbino, M. Temple, and J. Lopez, "A Comparison of PHY-Based Fingerprinting Methods Used to Enhance Network Access Control," in *ICT Systems Security and Privacy Protection*, (Hamburg, Germany), pp. 204–217, Springer International Publishing, 2015.

21. T. Carbino, M. Temple, and J. Lopez, "Conditional Constellation Based-Distinct Native Attribute (CB-DNA) Fingerprinting for Network Device Authentication," in *IEEE International Conference on Communications (ICC)*, pp. 1–6, 2016.

22. C. Rondeau, A. Betances, and M. Temple, "Securing ZigBee Commercial Communications Using Constellation Based Distinct Native Attribute Fingerprinting," *Hindawi Security and Communication Networks*, vol. 2018, pp. 1–14, 2018.

23. C. Rondeau, M. Temple, and A. Betances, "Dimensional Reduction Analysis for Constellation-Based DNA Fingerprinting to Improve Industrial IoT Wireless Security," in *52nd Hawaii International Conference on System Sciences (HICSS)*, pp. 7126–7135, 2019.

24. R. Duda, P. Hart, and D. Stork, *Pattern Classification.* Hoboken, NJ, USA: Wiley, 2nd ed., 2000.

25. C. Dubendorfer, B. Ramsey, and M. Temple, "ZigBee Device Verification for Securing Industrial Control and Building Automation Systems," in *Critical Infrastructure Protection VII*, (Berlin, Heidelberg), pp. 47–62, Springer International Publishing, 2013.

26. D. Reising, M. Temple, and J. Jackson, "Authorized and Rouge Device Discrimination Using Dimensionally Reduced RF-DNA Fingerprints," *IEEE Transactions. on Information Forensics and Security*, vol. 10, no. 6, pp. 1180–1192, 2015.

27. T. Bihl, K. Bauer, M. Temple, and B. Ramsey, "Dimensional Reduction Analysis for Physical Layer Device Fingerprints with Applications to ZigBee and Z-Wave Devices," in *IEEE Military Communications Conference (MILCOM)*, pp. 360–365, 2015.

28. H. Patel and B. Ramsey, "Comparison of Parametric and Non-Parametric Statistical Features for Z-Wave Fingerprinting," in *IEEE Military Communications Conference (MILCOM)*, pp. 378–382, 2015.

29. H. Patel, M. Temple, and R. Baldwin, "Improving ZigBee Device Network Authentication Using Ensemble Decision Tree Classifiers With Radio Frequency Distinct Native Attribute Fingerprinting," *IEEE Transactions. on Reliability*, vol. 64, no. 1, pp. 221–233, 2015.

30. Tuxfamily, "Eigen," Accessed: Dec. 10, 2019. [Online]. Available: http://eigen.tuxfamily.org/.

31. J. Gaeddert, "liquid-dsp," Accessed: Dec. 10, 2019. [Online]. Available: https://liquidsdr.org/.

32. Nuand, "bladeRF," Accessed: Dec. 10, 2019. [Online]. Available: https://github.com/Nuand/bladeRF/wiki.

33. ZigBee Alliance, *ZigBee Specification (ZigBee Document 05-3474-21)*, 2015.

34. A. King, "Towards Real-Time GPS Spoofing Detection Using Chip-Shape Distinct Native Attributes," M.S. thesis, AFIT-ENG-MS-20-M-032, Air Force Institute of Technology, WPAFB, OH, 2020.

35. G. Hu, S. Wu, X. Hu, M. Jing, and Y. Gao, "Blind Frequency and Symbol Rate Estimation for MSK Signal Under Low Signal-to-Noise Ratio," *Journal of Computational Information Systems*, vol. 9, no. 16, pp. 6651–6659, 2013.

36. B. Sklar, *Digital Communications*. Upper Saddle River, NJ, USA: Prentice-Hall, 2nd ed., 2014.

37. F. Rice, B. Cowley, B. Moran, and M. Rice, "Cramer-Rao Lower Bounds for QAM Phase and Frequency Estimation," *IEEE Transactions. on Communications*, vol. 49, pp. 1582–1591, 2001.

38. M. Luise and R. Reggiannini, "Carrier Frequency Recovery in All-Digital Modems for Burst-Mode Transmissions," *IEEE Transactions. on Communications*, vol. 43, pp. 1169–1178, 1995.

39. T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY, USA: Springer, 2nd ed., 2009.

40. J. Milton and J. Arnold, *Introduction to the Probability and Statistics*. New York, NY, USA: McGraw-Hill, 4th ed., 2003.

41. B. Efron and R. Tibshirani, *An Introduction to the Bootstrap*. London, UK: Chapman & Hall, 1993.

42. ZigBee Alliance, *ZigBee Cluster Library Specification*, 6 ed., 2016.

Acronyms

$\%C_{\mathbf{cls}}$ Average Percent Correct Classification. 55

%C Average Cross-Class Percent Correct Classification. xii, 24, 54, 55, 68

W Projection Matrix. 19, 28, 49, 50, 51, 52, 54, 70

BPSK Binary Phase Shift Keying. 8, 69

CB-DNA Constellation-Based Distinct Native Attribute. iv, 2, 4, 11, 14, 17, 18, 20, 23, 24, 28, 31, 34, 47, 52, 53, 58, 59, 60, 67, 68, 69, 70, 1

CCA Clear Channel Assessment. 28

CI Confidence Interval. 56, 64, 65

CLI Command Line Interface. 21, 22

CM Confusion Matrix. 24, 51, 54

CRC Cyclic Redundancy Check. 26

CSMA-CA Carrier Sense Multiple Access with Collision Avoidance. 28

dB Decibels. 28, 31, 33

DNA Distinct Native Attribute. 2, 11, 19, 67

DoD Department of Defense. 1

DRA Dimensional Reduction Analysis. 18

DS Data Symbol. 5, 6, 7, 14, 25, 26, 37, 38, 42, 41, 46, 47

DSP Digital Signal Processing. 22

E_s/N_0 Energy per Symbol to Noise Power Spectral Density. 28, 31

ED Euclidean Distance. 4, 19, 20, 50, 51, 52, 53, 54

FFT Fast Fourier Transform. 36, 38, 58, 59, 65, 69

FSK Frequency Shift Keying. 35

FT Fourier Transformation. 35

GPS Global Positioning System. 22, 32, 78

GPSDO Global Positioning System (GPS) Disciplined Oscillator. 32

GRLVQI Generalized Relevance Learning Vector Quantization-Improved. 19, 69

I In-Phase. 8, 9, 46

ICS Industrial Control Systems. 1, 4, 67

ID Identification. 70

IDF Integrate-and-Dump Filter. 46

IDS Intrusion Detection System. 70

IEEE Institute of Electrical and Electronics Engineers. iv, 1, 4, 6, 8, 18, 24, 25, 67, 69, 1

IoT Internet of Things. 67

IPS Intrusion Prevention System. 70

LDA Linear Discriminant Analysis. 19

LR-WPAN Low-Rate Wireless Personal Area Networks. iv, 1, 23, 67, 70, 1

MAC Medium Access Control. 1, 6, 26, 65

MDA Multiple Discriminant Analysis. 4, 19, 20, 49, 50, 52, 68, 69

MDA/ED Multiple Discriminant Analysis/Euclidean Distance. 23, 24, 52, 55, 68

ML Maximum Likelihood. 19, 68

MSK Minimum Shift Keying. 34

Msps Mega-Samples-Per-Second. 25, 27, 32, 33

NRT Near Real-Time. iv, 1, 2, 14, 23, 24, 38, 50, 52, 56, 65, 67, 68, 69, 1

OCXO Oven Controlled Crystal Oscillator. 32

O-QPSK Offset-Quadrature Phase Shift Keying. 4, 5, 9, 10, 17, 26, 28, 31, 34, 35, 36, 37, 39, 40, 46, 47

OSI Open Systems Interconnection. 11

PHR PHY Header Region. 6, 7, 25

PHY Physical Layer. iv, 1, 6, 11, 23, 25, 26, 67, 69, 70, 79, 1

PPDU PHY Protocol Data Unit. 6, 7

PSDU PHY Service Data Unit. 6, 7, 26

Q Quadrature. 8, 9, 46

QPSK Quadrature Phase Shift Keying. 5, 8, 9, 10, 39

RF Radio Frequency. iv, 1, 2, 20, 22, 34, 46, 51, 57, 70, 1

RF-DNA Radio Frequency Distinct Native Attribute. 4, 11, 12, 13, 14, 17, 19, 20, 24, 68, 69

RFI Radio Frequency Interference. 28, 67

RndF Random Forest. 19, 69

SDR Software Defined Radio. 20, 22, 25, 27, 28, 31, 32, 33, 34, 35, 52, 56

SER Symbol Error Rate. 39

SFD Start of Frame Delimiter. 6

SHR Sychronization Header Region. 6, 25, 42, 44, 68

SMA Sub-Miniature version A. 27, 28, 33

SNR Signal-to-Noise Ratio. 31, 36, 39, 40, 41, 70

UHD Universal Software Radio Peripheral (USRP) Hardware Driver. 21

USRP Universal Software Radio Peripheral. 20, 21, 28, 32, 33, 34, 35, 80

WPAN Wireless Personal Area Network. 4

ZCL Zigbee Cluster Library. 65

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704–0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202–4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
26–03–2020	Master's Thesis	Sept 2018 — Mar 2020

4. TITLE AND SUBTITLE

Near Real-Time Zigbee Device Discrimination
Using CB-DNA Features

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Matsui, Yousuke, Z, Capt, USAF

5d. PROJECT NUMBER

JON# 19G211

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENG-MS-20-M-043

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory, AFMC
Attn: Dr. Vasu Chakravarthy
2241 Avionics Circle, Bldg 620
Wright-Patterson AFB OH 45433-7765
Email: Vasu.Chakravarthy@us.af.mil, Comm: 937-713-4026

10. SPONSOR/MONITOR'S ACRONYM(S)

AFRL/RYWE

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

This work is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

14. ABSTRACT

Currently, LR-WPAN based on the IEEE 802.15.4 standard are at risk due to open-source tools which allow bad actors to exploit unauthorized network access through various cyberattacks by falsifying bit-level credentials. This research investigates implementing a RF air monitor to perform NRT discrimination of Zigbee devices using the IEEE 802.15.4 standard. The air monitor employed a Multiple Discriminant Analysis/Euclidean Distance classifier to discriminate Zigbee devices based upon CB-DNA fingerprints. Through the use of CB-DNA fingerprints, PHY characteristics unique to each Zigbee device strengthen the native bit-level authentication process for LR-WPAN networks. Overall, the developed RF air monitor achieved an Average Cross-Class Percent Correct Classification of $\%C_{tst} = 99.24\%$ during the testing of $N_{cls} = 5$ like-model BladeRF Software Defined Radios transmitting Zigbee protocol bursts. Additionally, to evaluate the NRT capability of the air monitor, a statistical analysis of $N_{timing} = 1000$ Zigbee bursts determined the worst-case average runtime from burst detection to classification. The analysis concluded that the runtime was $t_{runtime} \approx 269$ mSec. Ultimately, this research found that PHY characteristics provide an additional method of authentication NRT to enhance the inherent network security for Zigbee applications from cyberattacks.

15. SUBJECT TERMS

CB-DNA, PHY Characteristics, Device Classification, Zigbee, RF Air Monitor

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Maj. J. Addison Betances, AFIT/ENG
U	U	U	UU	94	19b. TELEPHONE NUMBER *(include area code)* (937) 255-3636; joan.betancesjorge@afit.edu