THE MANIPULATION OF RF-DNA FINGERPRINTS THROUGH THE USE OF A PHASE-MODULATED CLOCK IN IEEE802.11A WI-FI SIGNALS

By

Ahmed Ibrahim

Donald Reising
Assistant Professor of Electrical Engineering
(Committee Chair)

Farah Kandah
UC Foundation Associate Professor of
Computer Science and Engineering
(Committee Member)

Daniel Loveless
UC Foundation Associate Professor of
Electrical Engineering
(Committee Member)

THE MANIPULATION OF RF-DNA FINGERPRINTS THROUGH THE USE OF A PHASE-MODULATED CLOCK IN IEEE802.11A WI-FI SIGNALS

By

Ahmed Ibrahim

A Thesis Submitted to the Faculty of the University of
Tennessee at Chattanooga in Partial
Fulfillment of the Requirements of the Degree
of Master of Science: Engineering

The University of Tennessee at Chattanooga
Chattanooga, Tennessee

May 2020

ABSTRACT

The ubiquity of IoT devices has created an urgent need to augment existing network security mechanisms by leveraging discriminating, waveform characteristics to facilitate the detection of unauthorized devices. RF-DNA fingerprints are a waveform-based approach capable of distinguishing one device from others of the same manufacturer and model. This work investigates the extent to which the intentionally inserted changes can alter the RF-DNA fingerprints of the transmitted signal without negatively impacting the receiver's ability to demodulate the received signal. The experiments presented herein investigate intentional changes caused by the external clock to the preamble of the 802.11a Wi-Fi waveform from which RF-DNA fingerprints are extracted. Analysis is conducted using the Gabor Transform. The results show the structure of the preamble remains intact when the clock signal is phase-modulated using sine waves oscillating frequencies up to 10 kHz with deviation of 1.5 degrees, or 2.5 kHz with deviation of 90 degrees.

ACKNOWLEDGEMENTS

In the beginning, I would like to deliver my thank messages to my advisor, Dr. Reising, for his great support and guidance during this work. I learned a lot from him regarding the scientific details of work, how to develop a strong methodology, and how to pick an efficient approach to solve the research issues. Furthermore, he taught me the required skills to enhance presenting this work, like technical writing and how to communicate efficiently with the intended audience and readers.

Also, I would like to thank my co-advisor, Dr. Loveless, for teaching me how to analyze research problems and how to come up with proper solutions to address them. I would like to thank Dr. Kandah for his advice about how to deliver the information with direct and clear writing methods.

Also, I would like to thank Dr. Eltom, Dr. Karrar, Dr. Raga and all Electrical Engineering staff, my colleagues (Mohamed Fadul, Aaron Wilson, Joseph Cancelleri, Bharat Patel, Artem Malashiy, and Ryan Boggs), UTC Center for Global Education, my parents, siblings, family, and friends for their continuous support. Furthermore, I would like to extend my thanks to the SIM center, and the Center of Excellence for Applied Computational Science and Engineering (CEACSE) at the University of Tennessee at Chattanooga (UTC) for their assistance and support. Last but not least, I want to thank William Mitchell, and Kaitlin Hall from Research Experiences for Undergraduates (REU) program, funded by the National Science Foundation (NSF).

TABLE OF CONTENTS

CHAPTER

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

RF, Radio Frequency

RF-DNA, Radio Frequency Distinct Native Attribute

IoT, Internet of Things

OSI, Open System Interconnect

CFO, Carrier Frequency Offset

PHY, Physical Layer

Wi-Fi, Wireless Fidelity

2D, 2-Dimensions

DGT, Discrete Gabor Transform

PKI, Public Key Infrastructure

GT, Gabor Transform

MAC, Medium Access Control

STS, Short Training Sequence

GI, Guard Interval

LTS, Long Training Sequence

AGC, Automatic Gain Control

IDFT, Inverse Discrete Fourier Transform

DFT, Discrete Fourier Transform

PSK, Phase Shift Keying

QAM, Quadrature Amplitude Modulation

TF, Transfer Function

FEC, Forward Error Correction

I/Q, in-phase/quadrature

ISI, Inter Signal Interference

ICI, Inter-Carrier Interference

SDR, Software Defined Radio

ASIC, Application Specific Integrated Circuit

FPGA, Field Programmable Gate Array

GPP, General-Purpose Processor

DSP, Digital Signal Processing

OFDM, Orthogonal Frequency Division Multiplexing

VCTCXO, Voltage Controlled Temperature Compensated Crystal Oscillator

PLL, Phase-Locked Loop

GPIO, General Purpose Input/ Output

JTAG, Joint Test Action Group

PPS, Pulses Per Second PPS

API, Application Program Interface

SWIG, Simplified Wrapper, and Interface Generator

GRC, GNU Radio Companion

GUI, Graphical User Interface

FFT, Fast Fourier Transformation

BPSK, Binary Phase Shift Keying

DC, Direct Current

MCS, Modulation and Coding Scheme

AM, Amplitude Modulation,

FM, Frequency Modulation

PM, Phase Modulation

MDA, Multiple Discrimination Analysis

ML, Maximum Likelihood

NC, Number of Classes

SNR, Signal to Noise Ratio

VT, Variance trajectory

LO, Local Oscillator

PMF, Probability Mass Function

ISM, Industrial, Scientific, and Medical

TX, Transmitter

RX, Receiver

QPSK, Quadrature Phase Shift Keying

AWGN, Additive White Gaussian Noise

VoIP, Voice over IP

SEI, Specific Emitter Identification

CHAPTER 1

INTRODUCTION

1.1 Overview

It has been twenty-three years since the first work on Radio Frequency (RF) fingerprinting was published in 1996 [1]. As an analogy, RF fingerprinting can be described as being akin to human biometrics in which everyone's identity is established from different distinct and native attributes present within a given physical trait (e.g., a person's fingerprints, retinal patterns). RF fingerprinting is defined as the process through which distinct and native received signal characteristics are collected from the device to make it detectable and noticeable from another one of the same manufacturers, and even more, the model [2]. The method of RF fingerprinting that will be explained in this work is known as RF-Distinctive Native Attributes (RF-DNA) fingerprinting. The work in [3, 4] has shown that these unique attributes in the wireless transmitter's electromagnetic signals can be utilized as a way to differentiate it from others built using the same components and assembly process.

1.2 Motivation

Internet of Things (IoT) operating devices installed worldwide were counted as 23.14 billion at the end 2018, and it is projected to increase to 26.66 billion devices by the end of 2019 [5]. Wireless networks have been commonly protected using what is known as a "bit-level" mechanisms (e.g., encryption) within the Open Systems Interconnection (OSI) layer-2 (Datalink), layer-3 (Network) or higher layers shown in Figure 1.1. These techniques are commonly targeted by attackers, and also show weakness for challenging the user if he or she is an authorized user [6].

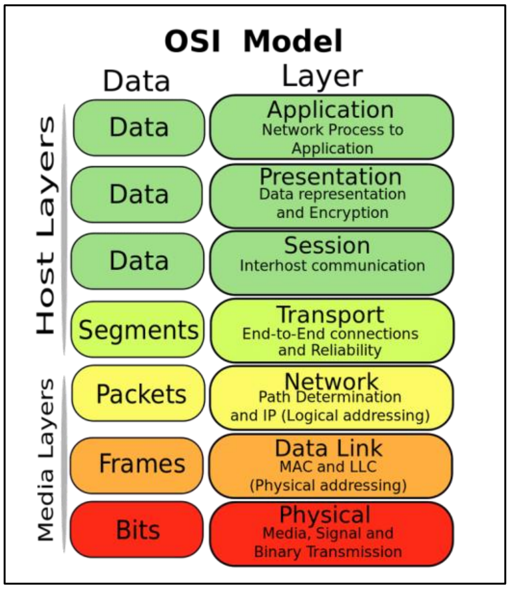

OSI Model
Data
Layer
Host Layers
Media Layers
Data
Application
Network Process to Application
Data
Presentation
Data representation and Encryption
Data
Session
Interhost communication
Segments
Transport
End-to-End connections and Reliability
Packets
Network
Path Determination and IP (Logical addressing)
Frames
Data Link
MAC and LLC (Physical addressing)
Bits
Physical
Media, Signal and Binary Transmission

Figure 1.1 OSI model [7]

The authentication of any device can be implemented using one or more of the following techniques: (i) passwords, (ii) pre-shared secrets, and (iii) public-key cryptosystems. A device's allowed access (i.e., Authorization) can be achieved by any kind of access control list (e.g., a database of users tied with their specific privileges). However, as the number of IoT devices grows, new security concerns have appeared. For instance, the factories, that utilize and depend on IoT devices to get higher efficiency in the production, will face a big loss if a single attack disrupts the factory operation [8]. This issues become more severe as the traditional security systems (e.g., Firewall) become more critical and vulnerable than before.[9].

There are many intrinsic parameter fluctuations that account for time-dependent workload conditions and post-deployment degradation (e.g., due to device overuse, thermal fluctuations, and exposure to ionizing radiation). As these conditions can change the physical attributes of the examined hardware, it is important to determine if the RF-DNA fingerprints are affected by these conditions. The motivation behind this work is the published paper titled, "Assessment of the Impact of CFO on RF-DNA Fingerprint Classification Performance" [10]. This paper addressed the effects caused by the frequency differences that exist between the local oscillators of the transmitter and receiver upon the RF-DNA fingerprinting process. The work in [10], implies a mechanism for improving RF-DNA fingerprint-based radio discrimination through the insertion of a unique feature or features. For the case of [10], a unique CFO value is inserted by the transmitter, which empirically led to improved discrimination of each radio from all others as the signal-to-noise ratio (SNR) degraded. The work in [10] also showed that a given waveform feature varies across the transmissions of the selected device, which impacts discrimination performance. This has led to the research question: Can the intentional manipulation of device components (e.g., the clock distribution system) result in the insertion or manipulation of RF-DNA fingerprint features?

As there are similarities between fingerprints of electronic components and human beings fingerprints, some requirements can also be utilized for the fingerprinting of electronic devices, such as: (1) universality: which means that every electronic component should have the characteristics to be used for identification, (2) uniqueness: that no two devices should have the same fingerprints, (3) permanence: which means that the characteristics should be conserved despite aging or condition, (4) collectability: The ability to quantify these characteristics [11].

1.3 Problem statement

There is an urgent need to augment current bit-level network security mechanisms to leverage useful discriminating information, which can be used to identify possible rogue devices. This problem is investigated through (i) Manipulation of a Software Defined Radio’s (SDR) inherent features by inserting intentional changes (e.g., the clock distribution system) and unintentional changes (e.g., temperature impact), (ii) Determine the origins of RF-DNA fingerprint features, and (iii) Develop new stochastic based modelling techniques for capturing intrinsic parameter fluctuations. This model can be used to detect efficiency degradation.

1.4 Objectives

The objectives of this RF-DNA fingerprinting work are: (i) Exploit physical attributes (either inherent or statistical features) to study the impact of intentional manipulation, and (ii) Determine the point at which demodulation is negatively impacted by the waveform changes caused by intentional manipulation of the external clock (i.e., loss of bit-error-rate performance at a given noise level).

1.5 Research contributions

This work investigates the extent to which clock manipulation impacts the RF-DNA fingerprint statistics and defines a demodulation breakpoint where, beyond which, the transmitted data can no longer be recovered by the receiver. The main contributions of this research are:

- It is concentrated in the Physical (PHY), thus working directly with the electromagnetic waves. At the receiver side, the collection is done for the signal before the demodulation step occurs, which guarantees that no changes have been made to the received signal.

- By using the GNU Radio Companion, a new capability has been tested of how the software can impact the transmitted and received signal of the SDR devices by doing different kinds of modulation and signal processing (e.g., sampling, and filtering)
- The expected outcomes will affect the electronic industry in how manufacturers build their components to utilize these physical attributes. This will lead to an enhancement of wireless transmitter discrimination for improved network security.
- Configure the SDR platform to facilitate the manipulation of specific RF components, such as the local oscillator. The goal is to manipulate a selected RF component's behavior or characteristic and then ascertain the impact that manipulation has on the discrimination of that particular wireless transmitter from a pool of transmitters (i.e., all the transmitters are sending Wi-Fi waveforms). The transmitters can be from different manufacturers, the same manufacturer with a different model, or the same manufacturer and model.
- The collected waveforms have been presented in two-dimensions (2D) of time and frequency using the Discrete Gabor Transform (DGT). Fingerprints have been generated as features quantified in the form of statistical measures.

1.6 Thesis outlines

The next four chapters' outlines are:

- Chapter 2: Background: This chapter provides an overview of the literature on The IEEE 802.11 standard, Modulation techniques, Software Defined Radio (SDR), GNU radio companion software, Orthogonal Frequency Division Multiplexing (OFDM) receiver, Gabor Transform (GT) based fingerprint generation, and the relevant work of RF fingerprinting and hardware changes effects.

- Chapter 3: Methodology: This chapter presents the GNU radio companion flowchart used to build the transmitter and the receiver, description of the experiment environment, the process of capturing the Wi-Fi signals, extracting the preambles, analyzing them using the GT, and generating the statistical fingerprints.
- Chapter 4: Results and Discussions: This chapter shows the plots of the preamble of the transmitted waveform, received waveform for both non-modulated clock cases, and the phase-modulated clock. Also, it defines the demodulation breakpoints and showing the time domain plots, and GT of the preamble for different cases. Moreover, a comparison is made to highlight the differences between modulated and non-modulated clock.
- Chapter 5: Conclusion: This chapter concludes the findings and contributions of this work as well as proposes topics and challenges that can be addressed for follow-on research.

CHAPTER 2

BACKGROUND

2.1 Introduction

This chapter provides details about the technology and the RF communication terms that will be used in this thesis document. First of all, the IEEE 802.11a standard will be discussed, and the structure of its preamble, which is the main signal of interest, will be shown. The modulation techniques of OFDM and Phase Modulation (PM) will be illustrated. Then, the SDR platform is described, including LimeSDR-mini. After that, the main software used in the experiments, GNU radio companion, is also covered. Next, the IEEE 802.11a Wi-Fi PHY layer is explained in detail. Also, RF-DNA fingerprinting is explained, including how it is formed and interpreted. The clocking system and how it is designed are shown for the LimeSDR-mini platform. Plus, the RF local oscillator is explained. Finally, relevant works are summarized, and their relevancy to this work explained.

2.2 IEEE 802.11 standard

In October 1997, the IEEE 802 Executive Committee approved two projects for higher rate PHY extensions to the IEEE 802.11 Wireless-Fidelity (Wi-Fi) communications standard. The first extension, IEEE 802.11a, defined the operational requirements (e.g., frequency, bandwidth, modulation) in the 5.0 GHz band and data rates ranging from 6 to 54 Mbps. The second extension, IEEE 802.11b, defined a set of PHY specifications operating in the 2.4 GHz Industrial, Scientific, and Medical (ISM) frequency band capable of data rates from 1 to 11 Mbps. Both PHY were defined to operate in conjunction with the existing Medium Access Control (MAC) layer [12].

As the PHY layer defines the means of transmitting raw bits, it is not related to how the data is sent logically over a physical link connecting network nodes (i.e., it is not related to how the packets are transitioned among switches or routers). It is the interface between the MAC layer and the wireless or wired medium. There are three levels of functions within the PHY layer: (i) it provides a frame exchange between MAC layer and the physical layer, (ii) it uses signal carrier and spread spectrum modulation to transmit data frames over the transmission link, and (iii) to check if there is a traffic on the PHY link ,it provides a carrier sense indication back to the MAC layer [13].

2.3 Modulation techniques

2.3.1 OFDM

Orthogonal Frequency Division Multiplexing (OFDM) plays a major role in communication systems that require high data rates; especially, with the broad expansion of applications that either require real-time processing (e.g., voice over IP) or higher network throughput (e.g., video streaming). OFDM is a multi-carrier (i.e., uses sub-carriers) parallel transmission technique; it splits the data into sub-streams, which have a low data rate and are modulated separately. The sub-carriers have a small bandwidth compared with the coherence bandwidth of the channel; that is, under poor channel conditions, the transmissions can still reach their destination. This indicates that the symbol period of the sub-streams is long when compared to the delay spread of the radio channel. Also, due to the orthogonality of the set of carrier frequencies, a high spectral efficiency is obtained because the spectra of the sub-carriers overlap, while mutual influence among the sub-carriers can be avoided by introducing a guard period known by cyclic prefix (CP) [14].

IEEE 802.11a Wi-Fi is a pulsed signal, or it is often referred to as a ‘burst’ signal. The total allotted bandwidth (BW) for 802.11a Wi-Fi is 20 MHz, while the occupied bandwidth (OBW) is 16.6 MHz. As shown in Figure 2.1, a single OFDM symbol contains 52 sub-carriers; 48 of them are data sub-carriers, and the remaining four are devoted to being pilot sub-carriers. The center, "null," zero sub-carrier is not used. All data sub-carriers use the same modulation format within a given burst. However, the modulation format can vary from burst to burst. The possible data subcarrier modulation techniques are Binary Phase Shift Keying (BPSK), Quadrature Phase Shift Keying (QPSK), 16 Quadrature Amplitude Modulation (QAM), and 64QAM. The pilot sub-carriers are always modulated using BPSK with a known magnitude and phase. Symbols are carried in the OFDM subcarrier, along with its magnitude and phase information. This means that each subcarrier and OFDM symbol has a different magnitude and phase in every Wi-Fi pulse [15].

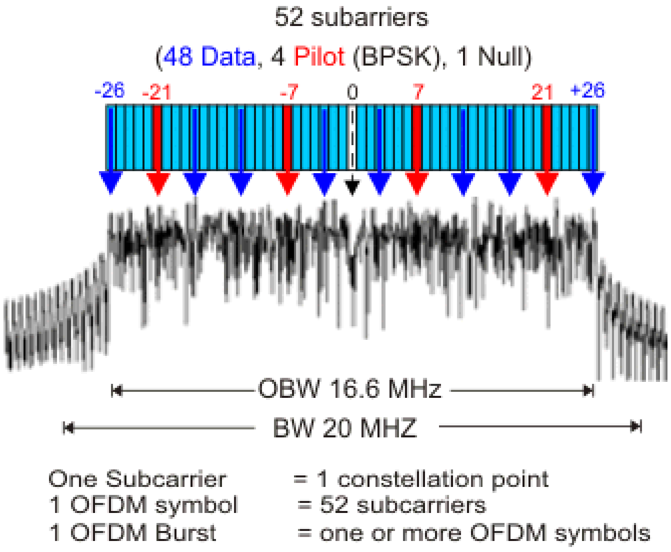
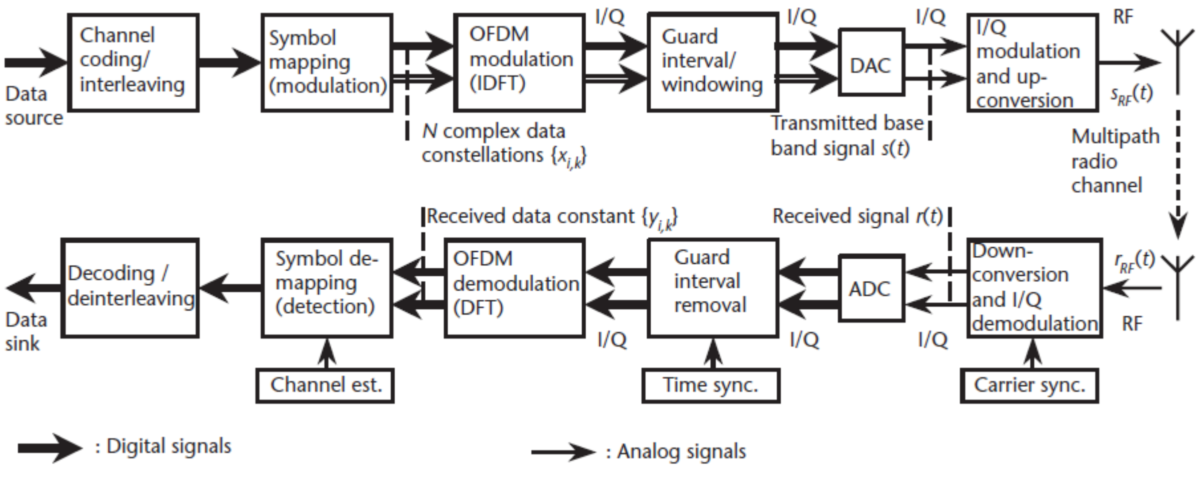

52 subarriers
(48 Data, 4 Pilot (BPSK), 1 Null)
-26
-21
-7
0
7
21
+26
OBW 16.6 MHz
BW 20 MHZ
One Subcarrier = 1 constellation point
1 OFDM symbol = 52 subcarriers
1 OFDM Burst = one or more OFDM symbols

Figure 2.1 OFDM spectrum [15]

Figure 2.2 shows the block diagram of a point-to-point transmission system using OFDM. The channel coding and interleaving functions are important, as in every channel, there is attenuation or loss of the transmitted data symbols on one or several sub-carriers. Severely attenuated or lost symbols can lead to bit errors within the receiver's demodulation process. An efficient coding scheme can correct for the wrong bits and thereby utilize the channel physical resources (i.e., the bandwidth) [14].

After that, symbol mapping (modulation) takes place. The OFDM modulation and demodulation of the data constellations on the orthogonal sub-carriers are done through the calculation of the Inverse Discrete Fourier Transform (IDFT) and Discrete Fourier Transform (DFT), respectively. At the input of the IDFT, N data constellation points $\{x_i,k\}$ are present, where N is the number of DFT points, 'i' is an index on the subcarrier, and; 'k' is an index on the OFDM symbol. These constellations can be taken according to any PSK or QAM signaling [14].



Data source
Channel coding/ interleaving
Symbol mapping (modulation)
OFDM modulation (IDFT)
I/Q
Guard interval/ windowing
I/Q
DAC
I/Q
I/Q modulation and up-conversion
RF
$s_{RF}(t)$
N complex data constellations $\{x_{i,k}\}$
Transmitted base band signal $s(t)$
Multipath radio channel
Received data constant $\{y_{i,k}\}$
Received signal $r(t)$
$r_{RF}(t)$
Decoding / deinterleaving
Symbol de-mapping (detection)
OFDM demodulation (DFT)
Guard interval removal
ADC
Down-conversion and I/Q demodulation
RF
Data sink
I/Q
I/Q
I/Q
Channel est.
Time sync.
Carrier sync.
: Digital signals
: Analog signals

Figure 2.2 Simple point-to-point transmission using OFDM [14]

Next, the guard interval is added to avoid interference followed by Digital to Analog Converter (DAC) to be transmitted through the communications channel. The complex equivalent baseband signals generated by digital signal processing are in-phase and quadrature (I-Q)–modulated and up-converted to be transmitted via an RF carrier. The reverse steps are performed by the receiver [14].

Synchronization is crucial to the efficient operation of an OFDM receiver. Time synchronization is required to identify the start of an OFDM symbol. Frequency synchronization is used to align the modulator's and the demodulator's local oscillator frequencies. If there is a situation of asynchronous, then the orthogonality of the sub-carriers is lost, which results in Inter Signal Interference (ISI) and Inter-Carrier Interference (ICI) [14].

2.3.2 Phase Modulation (PM)

Analog modulation has three main types, which are Amplitude Modulation (AM), Frequency Modulation (FM), and Phase Modulation (PM). PM is achieved by changing the phase of the carrier linearly depending on the message signal amplitude. In PM, the total carrier's phase is varied proportionally to the message signal *m*(t) with a phase deviation constant of k_p radians per unite amplitude of *m*(t). The total instantaneous phase θ_i(t), the corresponding instantaneous frequency ω_i(t), and the PM signal Φ_{PM} (t) are given respectively by,

$$\theta_i(t) = \omega_c * t + k_p\, m(t), \tag{2.1}$$

$$\omega_i(t) = \frac{d\theta_i}{dt} = \omega_c + k_p\, m'(t), \tag{2.2}$$

and

$$\Phi_{PM}(t) = A_c \, cos\, (\omega_c * t + k_p\, m(t)), \tag{2.3}$$

where A_c is the carrier amplitude [16].

2.4 Software Defined Radio (SDR)

2.4.1 Overview

The IEEE has defined the SDR as a "Radio in which some or all of the physical layer functions are defined in software" [17]. The International Telecommunication Union (ITU) has another definition which is "A radio transmitter and/or receiver employing a technology that allows the RF operating parameters including, but not limited to, frequency range, modulation type, or output power to be set or altered by software, excluding changes to operating parameters which occur during the normal pre-installed and predetermined operation of a radio according to a system specification or standard" [18]. The main difference between the SDR and the traditional radios, is that the ability of the SDR to perform all signal processing in software by using technologies, such as Field Programmable Gate Array (FPGA), General-Purpose Processor (GPP) and Digital Signal Processing (DSP) chips to implement all the hardware radio elements [19].

The common attributes (e.g., carrier frequency, signal bandwidth, modulation, network access, cryptography, FEC coding, source coding, …, etc.) are also applied in the SDR, but in software. The SDR is a multi-function and cost-effective general-purpose device because the same radio tuner and processors can be used to implement many waveforms at many frequencies and can be easily upgraded with new software, which can contain new libraries of waveforms and applications [20].

On the other hand, some SDR disadvantages include: (i) increased power consumption due to high signal processing, (ii) complexity associated with the effort that goes along with the development of both firmware and software, and (iii) limited scope of SDRs on only the physical layer with no cooperation with the other layers [21].

2.4.2 LimeSDR-Mini

The LimeSDR-Mini, shown in Figure 2.3, "provides a hardware platform for developing and prototyping high-performance and logic-intensive digital and RF designs using Intel's MAX 10 FPGA and Lime Microsystems transceiver". The LimeSDR-mini transceiver has two RF inputs and operates over RF frequencies ranging from 10 MHz up to 3.5 GHz. The basis of this SDR is the LMS7002M transceiver chip, the Intel MAX 10 (10M16SAU169C8G 169-UBGA) FPGA chip, and USB 3.0 controller (FTDI FT601). The LimeSDR-Mini has an onboard 40 MHz Voltage Controlled Temperature Compensated Crystal Oscillator (VCTCXO) that serves as the reference clock for the transceiver and FPGA Phase-Locked Loops (PLLs) [22].

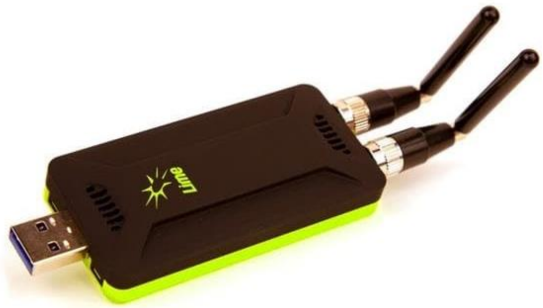


Figure 2.3 LimeSDR-Mini [23]

2.5 GNU radio companion

SDR platforms are programmed using the GNU Radio, which is a popular software environment equipped with a rich library of signal processing blocks (e.g., filters, decoders, modulators, and encoders) and other general-purpose blocks used in radio systems. It uses both the Python and C++ programming languages. GNU Radio applications are primarily written using Python, while C++ is used to create complex signal processing blocks. The open-source mediation software, known as Simplified Wrapper and Interface Generator (SWIG), is the “glue code” that enables the calling of C++ functions within the Python programming language. Figure 2.4 illustrates the organization of data flow in GNU Radio [20].

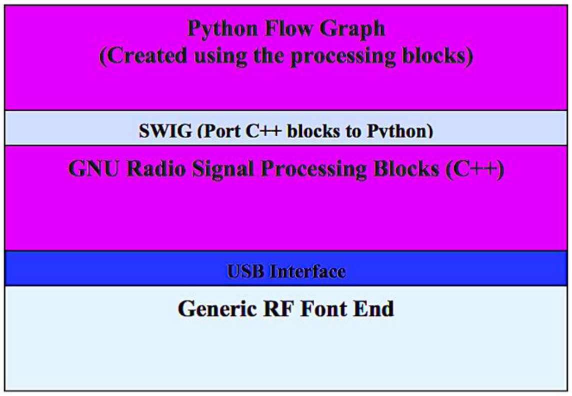

Python Flow Graph
(Created using the processing blocks)
SWIG (Port C++ blocks to Python)
GNU Radio Signal Processing Blocks (C++)
USB Interface
Generic RF Font End

Figure 2.4 GNU radio block diagram [20]

Additional software, known as GNU Radio Companion (GRC), is a Graphical User Interface (GUI) tool for creating signal flow graphs and generating flow-graph source code [24]. GRC facilitates the interaction with the SDR due to programming simplicity. In addition, it provides some flexibility through variable blocks to pass variable values and also import some Python functions [20].

2.6 IEEE 802.11a Wi-Fi PHY layer

2.6.1 IEEE 802.11a preamble

In this work, RF-DNA fingerprints are generated from the IEEE 802.11a preamble, shown in Figure 2.5. The preamble comprises the first 16 μs of every 802.11a Wi-Fi transmission. Starting from the left side of Figure 2.5, it contains 10 Short Training Sequences (STS), t_1 to t_{10}, followed by a Guard interval (GI2), and two Long Training Sequences (LTS), T_1 and T_2. The duration of each STS, GI, and LTS are 0.8 μs, 1.6 μs, and 3.2 μs, respectively. The first 7 STS are dedicated to Automatic Gain Control (AGC), and diversity selection. The remaining 3 STS are used for coarse frequency offset estimation. The 2 LTS are used for channel and fine frequency offset estimation [25].

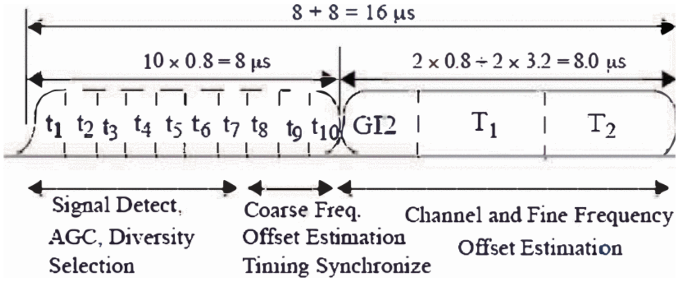

8 + 8 = 16 μs
10 × 0.8 = 8 μs
2 × 0.8 + 2 × 3.2 = 8.0 μs
t1 t2 t3 t4 t5 t6 t7 t8 t9 t10
GI2
T1
T2
Signal Detect, AGC, Diversity Selection
Coarse Freq. Offset Estimation Timing Synchronize
Channel and Fine Frequency Offset Estimation

Figure 2.5 The structure of the IEEE 802.11a preamble [25]

2.6.2 Frame detection

It is based on the frame detection algorithm, which is the autocorrelation of the short training sequence (STS) calculated by,

$$a[n] = \sum_{k=0}^{N_w-1} s[n+k]\,\overline{s}[n+k+\mathrm{L}]. \tag{2.4}$$

Each IEEE 802.11a frame starts with 10 STSs that each consist of a pattern that spans L samples. The receiver utilizes this cyclic pattern (i.e., the STS pattern is the same in every preamble) and calculates the autocorrelation value ‘a’ of the incoming sample stream ‘s’ with the delayed sample stream of the complex conjugate of s, (i.e., $\overline{s}$), by summing up the autocorrelation coefficients over an adjustable window ’N_w’[26]. The autocorrelation is normalized with the average power ‘p’ calculated as follows,

$$p[n] = \sum_{k=0}^{Nw\ -1} s[n+k]\,\overline{s}[n+k]. \tag{2.5}$$

The autocorrelation coefficient (i.e., the threshold) ‘c’ is calculated as follows:

$$c[n] = \frac{a[n]}{p[n]}. \tag{2.6}$$

If the calculation of (2.6) results in a ‘plateau’ that exceeds a configurable threshold value ‘c’ for three consecutive STSs, then an 802.11a frame is detected by the receiver. Figure 2.6 shows a case when the ‘plateau’, from (2.6), satisfies the threshold, ‘c'. However, if the plateau isn't detected, then the frame is dropped [26].

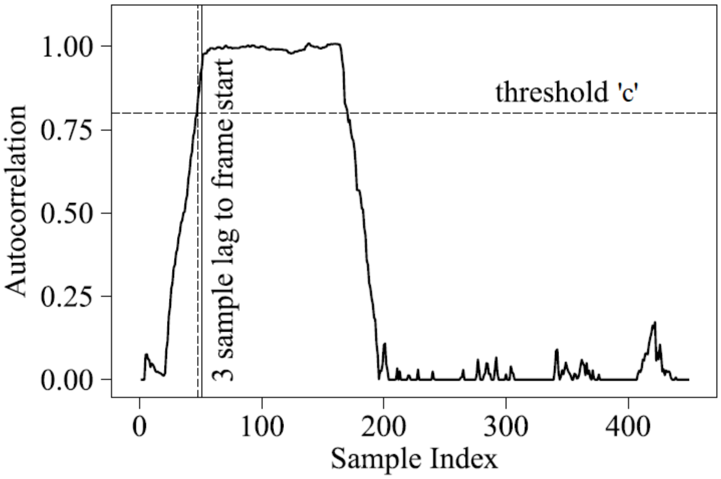


Figure 2.6 Autocorrelation of samples exceeding the threshold *'c'* during frame reception [26]

2.6.3 Frequency offset correction

Frequency offset correction is needed to compensate for the frequency mismatch that exists between the local oscillators of the transmitter and receiver. The coarse estimation of the phase rotation per sample is implemented using the STSs, while the fine estimation is implemented using the LTSs. Pilot symbols are used for additional phase estimation of the phase rotation [27].

Ideally, during the short sequence, a sample s[n] should correspond to the sample $\overline{s}$ [n+16] due to its cyclic property, where 16 the number of samples comprising an STS when sampling at a rate of 20 MSPS [26]. The final value for the coarse frequency offset α_{ST} is then calculated by,

$$\alpha_{ST} = \frac{1}{16}\arg(\sum_{n=0}^{N_{sh}-1} s[n]\,\overline{s}[n+16)]), \tag{2.7}$$

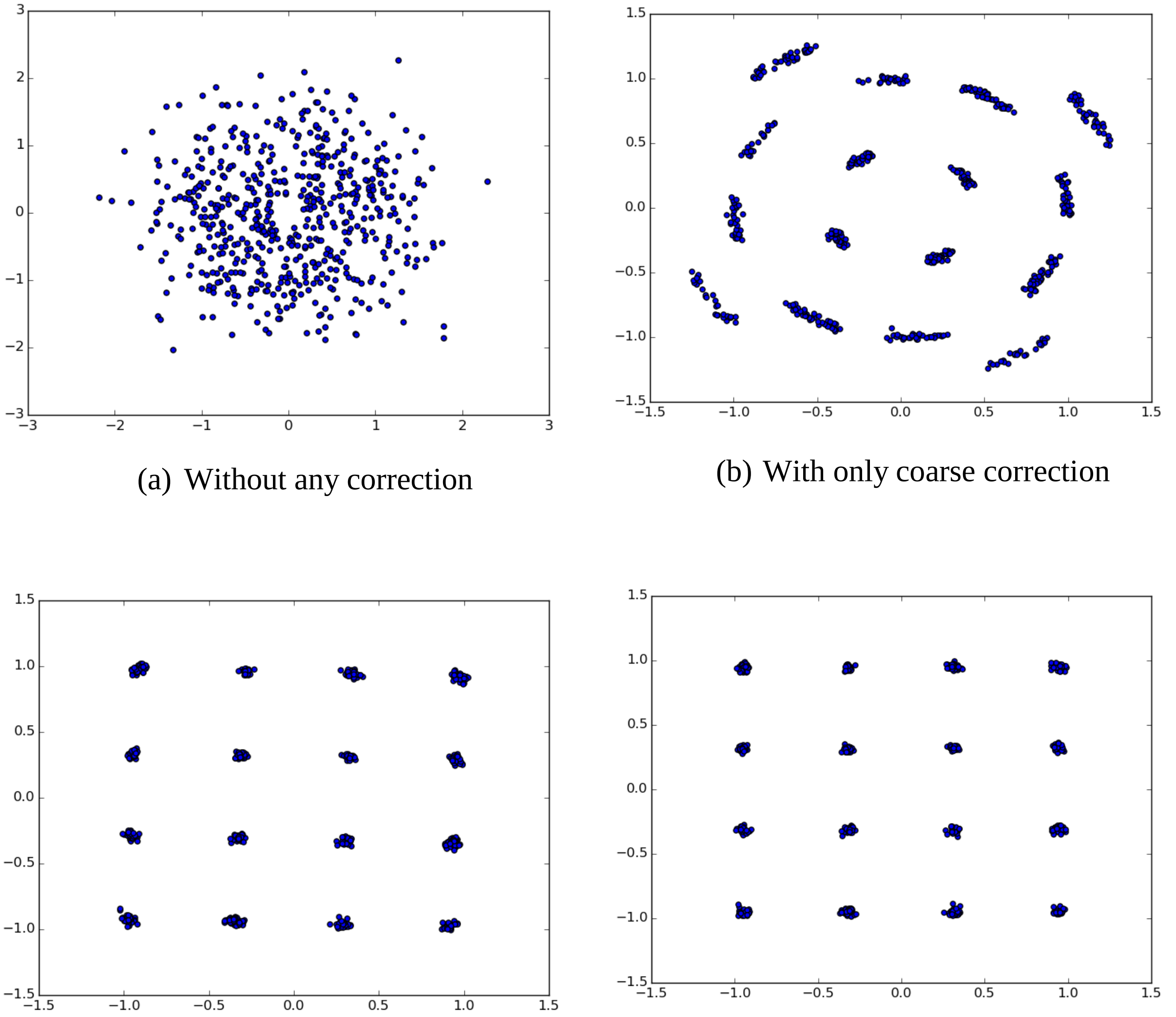
where N_{sh} is the length of the short training sequence. The frequency offset is then applied to each sample as,

$$s[n] \leftarrow s[n]e^{i(n\alpha_{ST})}. \tag{2.8}$$

The fine CFO estimation α_{LT} is obtained utilizing the 2 LTSs. As they are 64 samples each at 20 MSPS, the frequency offset can be calculated by [27],

$$\alpha_{LT} = \frac{1}{64}\arg(\sum_{n=0}^{63} s[n]\,\bar{s}[n+64)]) \ . \tag{2.9}$$

Figure 2.7a shows sample constellation points without any correction. In this case, the probability of error is high because the constellation points are mixed, and the decision made by the receiver will most probably lead to the wrong estimation of the transmitted symbol. Figure 2.7b shows the constellation with only coarse correction. It is better than the Figure 2.7a, but still, there is a low probability of error. Figure 7.9c shows the constellation with both coarse and fine correction, and figure 2.7d shows the constellation with coarse, fine, and pilot correction [28].



(a) Without any correction

(b) With only coarse correction

(c) With both coarse and fine correction

(d) With coarse, fine, and pilot correction

Figure 2.7 Sample constellation points: (a) without any correction, (b) with only coarse correction, (c) with both coarse and fine correction, and (d) with coarse, fine, and pilot correction [28]

2.6.4 Symbol alignment

Symbol alignment is achieved by leveraging the two LTSs. The process of symbol alignment consists of the calculation of the start of a symbol, the extraction of the data symbols, and processing them with an algorithm that calculates the Fast Fourier Transform (FFT) [26]. Figure 2.8 shows the correlation of the input stream with the known LTS sequence.

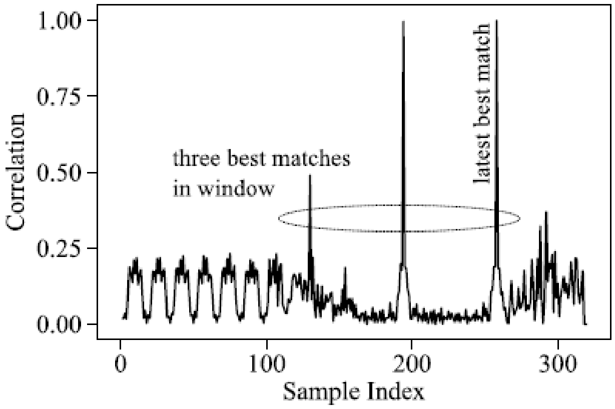


Figure 2.8 The correlation of the sample input stream with the known LTS sequence [26]

The indices of the highest three peaks N_p are calculated by,

$$N_p = \underset{n \in \{0,\ldots,N_{pr}\}}{\arg max_3} \sum_{\mathrm{k}=0}^{v} s[n+k]\,\overline{LT}[k], \tag{2.9}$$

where $\arg max_3$ returns the top three indices maximizing the expression, N_p is the number of samples in the preamble, v is the number of samples in the repeating pattern of the LTS, and $\overline{LT}$ is the LTS pattern [26]. The first data symbol starts at sample index,

$$N_p = \max(N_p) + \mathrm{v} \ . \tag{2.10}$$

Finally, by knowing the start of the data symbols, the cyclic prefix can be removed, samples that correspond to individual data symbols are:

$$s \leftarrow (\underbrace{s\{N_p + L\}, \ldots, s\{N_p + L + v - 1\}}_{first\ symbol}, \underbrace{s\{N_p + \ 2L + v - 1\}, \ldots}_{second\ symbol}) \quad . \tag{2.11}$$

2.6.5 Channel estimation

Based on [29], the main functions of the two LTSs are for channel estimation and symbol alignment. One of the methods used for channel estimation is the Least Square Estimator (LSE), which will be explained in this section. Assume the symbol used for the spectrum of a transmitted signal is ‘X’ while the received signal is ‘Y.' ‘H’ symbol is used to show the channel frequency response. The relationship between X, Y, and H is given by [30],

$$Y = \begin{bmatrix} H_0 \\ H_1 \\ \vdots \\ H_{N_{sc}-1} \end{bmatrix} \begin{bmatrix} X_0 & 0 & \dots & 0 \\ 0 & X_1 & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \dots & X_{N_{sc}-1} \end{bmatrix} + \begin{bmatrix} N_0 \\ N_1 \\ \vdots \\ N_{N_{sc}-1} \end{bmatrix}, \tag{2.12}$$

where N is the frequency response of the Additive White Gaussian Noise (AWGN). The goal of the Least Square (LS) estimator is to minimize the cost function $J(\widehat{H})$ given by,

$$J(\widehat{H}) = (Y - HX)^H (Y - HX), \tag{2.13}$$

where $\widehat{\mathrm{H}}$ is the channel frequency response estimation, and $(\ )^{\mathrm{H}}$ represents the conjugate transpose of the matrix. By assuming N = 0, LS algorithm solves equation (2.13) and estimates the channel frequency response by using,

$$\widehat{H}_{ls} = X^{-1}Y. \tag{2.14}$$

The use of both LTSs in (2.14) will be an advantage to eliminate 50% of both the noise variance and the estimation of square error. This LS approach to channel estimation is given by [30],

$$\widehat{H}_{ls} = \frac{1}{2} . X^{-1} . (Y_1 + Y_2), \tag{2.15}$$

where Y_1 and Y_2 are the Discrete Fourier Transforms of the first and second received LTSs, respectively [29].

2.7 RF-DNA Fingerprinting

2.7.1 Overview

During generation and transmission of an electromagnetic waveform unique behaviors and distinguishable characteristics are unintentionally and inherently imparted by the radio's RF front-end components, their interactions with one another as well as the operating environment. The RF-DNA fingerprinting is a Specific Emitter Identification (SEI) approach in which the fingerprints are generated from a fixed, known sequence of symbols within the transmitted waveform. The IEEE 802.11a Wi-Fi preamble is one such example of a fixed, known sequence of waveform symbols from which RF-DNA fingerprints can be extracted [31].

2.7.2 Gabor transform

As in [32], instead of using either time or frequency domain representations of the signal, this work uses RF-DNA fingerprints extracted from both of them through the generation of the time-frequency (T-F) response. Based on the work [32], the T-F response is generated through the calculation of the Discrete Gabor Transform (DGT) given by,

$$G_{mk} = \sum_{n=1}^{MN_{\Delta}} s(n)W^{*}(n - mN_{\Delta})e^{-j\gamma}, \tag{2.12}$$

where G_{mk} is the Gabor coefficients, the periodic input signal is: $s(n) = s(n + lMN_{\Delta})$, the periodic analysis window is: $W(n) = W(n + lMN_{\Delta})$, the total number of shifted samples is: $\gamma = 2\pi kn/K_G,\ N_{\Delta}$, number of shifts is: $m = 1, 2, \dots, M,\ k = 1, 2, \dots, K_G - 1$ for $K_G \geq N_{\Delta}$ and $mod(\, MN_{\Delta}, K_G) = 0$, which is explained in detail in [33].

The Generation of the RF-DNA fingerprints is based on the normalized values of the magnitude-squared GT coefficients $|G_{mk}|^2$ which are given by [32],

$$\overline{|G_{mk}|^2} = \frac{|G_{mk}|^2 - min\{|G_{mk}|^2\}}{max\{|G_{mk}|^2 - min\{|G_{mk}|^2\}\}}\,. \tag{2.13}$$

Figure 2.9 shows a representative normalized magnitude-squared T-F surface generated from the complex Gabor coefficients. The surface is divided up into N_R two-dimensional sub-regions, which are called “Patches," each sub-region contains a total of $N_T \times N_F$ values. The variables N_T is the time dimension length, and N_F is the frequency dimension length of the sub-region. To meet the condition of “population samples sufficiency” [34], the value of N_T and N_F are chosen to ensure that, at least, $N_{TF} = 15$ values comprise a sub-region. For each sub-region, the statistics of standard deviation, variance, skewness, and kurtosis, are calculated. These same statistics are also calculated over the entire T-F surface, constituting the $N_R + 1$ sub-region [35].

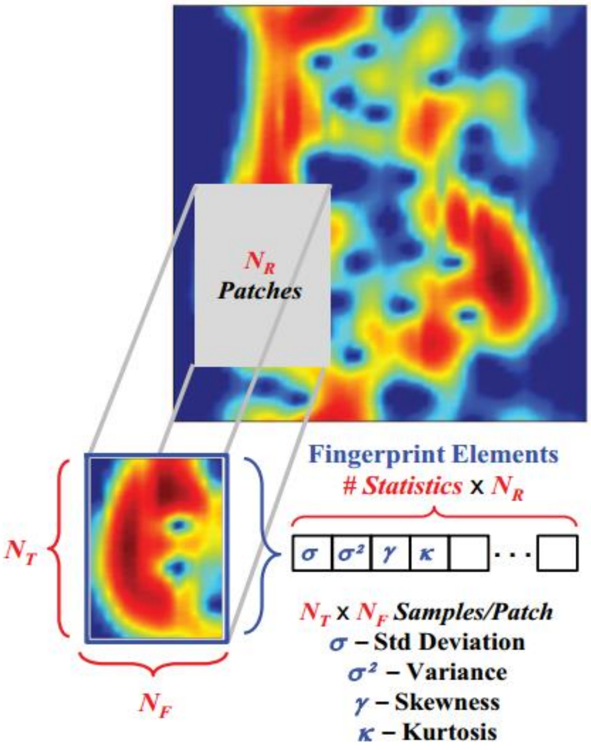


Figure 2.9 Gabor transform of normalized magnitude-squared coefficients (2D) [35]

Every fingerprint is a vector (i.e., a one-dimensional array) of numbers. Figure 2.10 shows a sample GT plot (left) and a representative set of RF-DNA fingerprint features (right). The first three features are the variance, skewness and kurtosis for patch-1 (top left corner of the GT plot). The next three features are the same statistics for patch-2 (N_T values to the right and adjacent to patch-1).The last three features are the statistics for patch-3 (N_T values to the right and adjacent to patch-2). These fingerprint features can be used to assess the statistical time-frequency behavior of the transmission. In this work, the statistical features obtained from these fingerprint vectors are used to assess the impact of a phase-modulated reference clock on the resulting radio transmission.

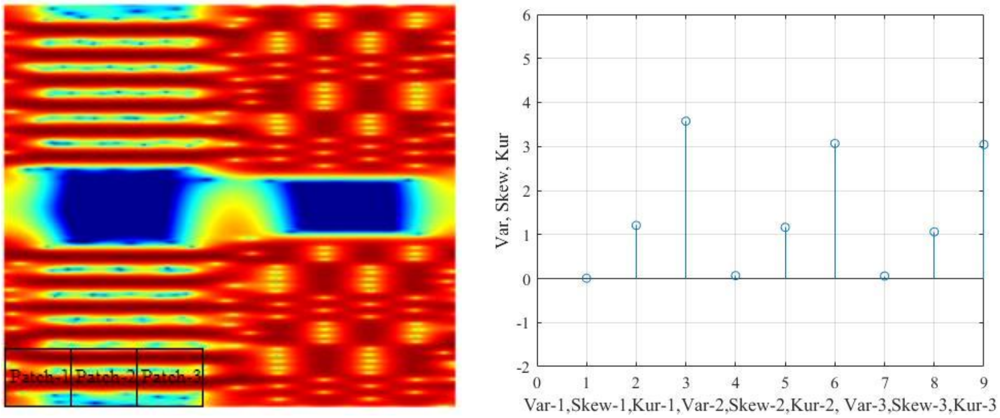


Figure 2.10 Three sample patches of GT (left) and their corresponding sampler-DNA fingerprint elements (right)

2.8 Clocking system

The LimeSDR-mini board is shown in Figure 2.11. It shows the main components of the LimeSDR-mini hardware. LimeSDR-Mini board clock distribution block diagram is presented in Figure 2.12. It has an onboard 40 MHz VCTCXO that is the reference clock for the LMS and FPGA PLLs. The VCTCXO frequency can be tuned by adjusting the DAC (IC9). Buffered VCTCXO clock is connected to RF transceiver, FPGA, as well as to connector J9 (REF_CLK_OUT), which can be fed to external hardware for synchronization. The VCTCXO can be disconnected from the clock buffer input by removing resistor (R59) and soldering resistor (R62). Both R59, and R62 are shown in Figure 2.13. This facilitates the connection of an external reference clock via connector J8 (REF_CLK_IN) [22]. The use of an external reference clock enables the supplying of a phase-modulated clocking signal to the LimeSDR-Mini's PLL that generates the carrier signal used in bandpass transmissions.

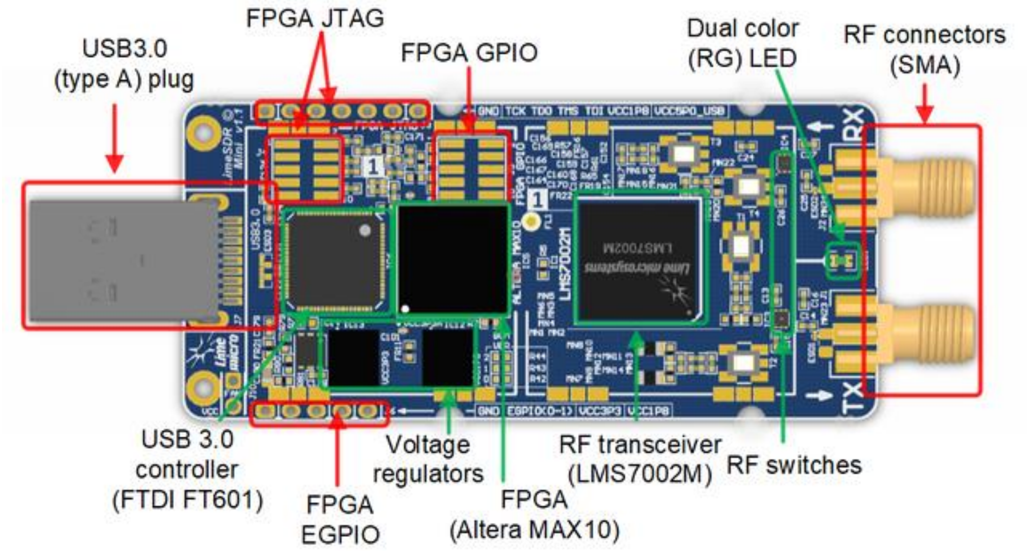


Figure 2.11 LimeSDR-Mini board [22]

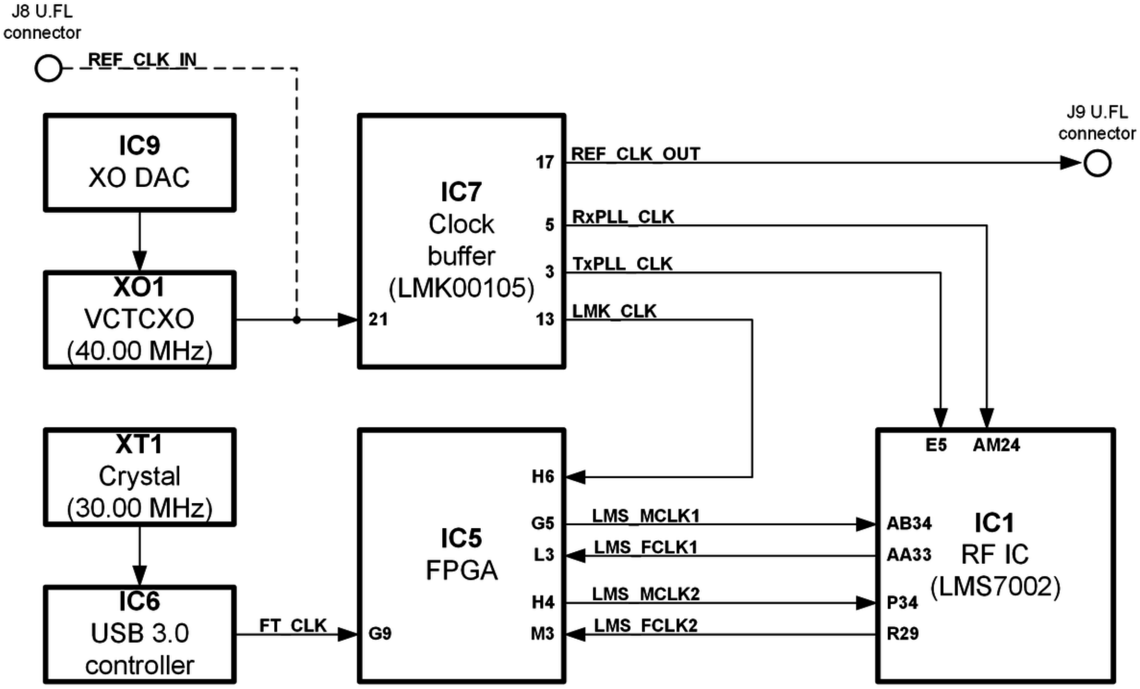


Figure 2.12 Block diagram of the LimeSDR-Mini board clock distribution system [22]

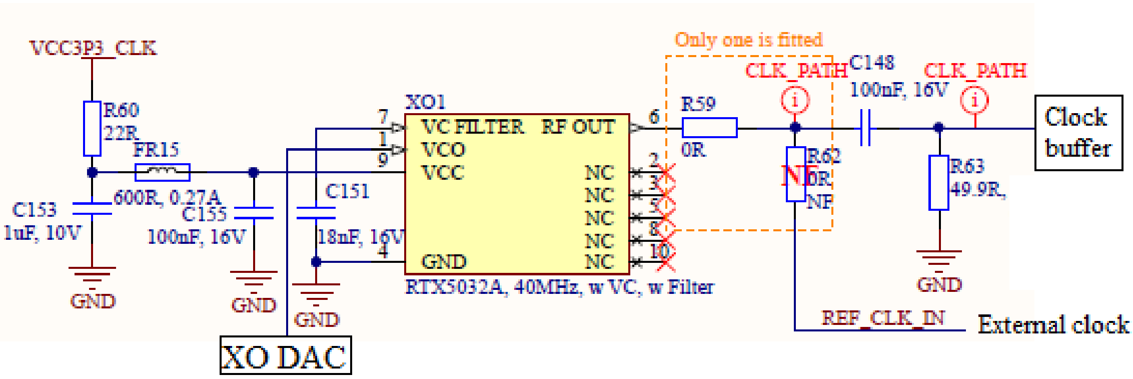


Figure 2.13 R59, and R62 in the clocking path of the LimeSDR-Mini schematic diagram [36]

2.9 RF Local Oscillators (LO)

In this section, the local oscillator (LO) is described, as it is relevant to the presence of carrier frequency offset within the RF-DNA fingerprinting process. The main function of the LO is to generate a signal that oscillates at the required frequency value, which is used in up- and down-conversion by a transceiver. It takes place by producing the sum and difference frequencies of the frequency of the local oscillator and frequency of the input signal of interest. Local oscillators are used in communications circuits such as radios, modems, and frequency division multiplexing systems [37].

In mobile devices, LO's are typically implemented using PLLs. Figure 2.14 shows the basic block diagram of a PLL.

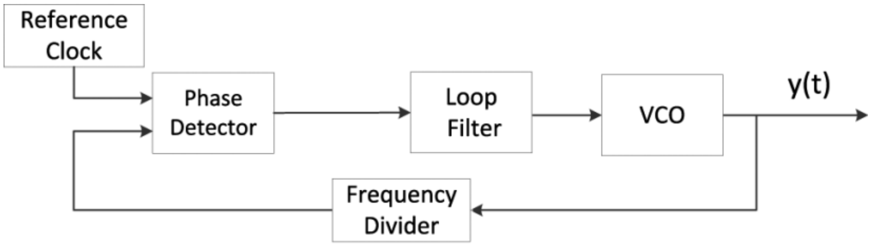


Figure 2.14 A basic PLL block diagram [38]

An ideal PLL would generate a sinusoidal oscillation at a carrier frequency f_0. Instead, in practice, the PLL generates a signal of the form,

$$y(t) = \cos(2\pi (f_0 + \Delta(t))\, t + n(t)), \tag{2.16}$$

where $\Delta(t)$ is the frequency offset and $n(t)$ is the phase noise [38].

The amount of tolerated CFO depends on the oscillator precision tolerance. In the range of 5GHz ISM band (including the IEEE802.11a Wi-Fi band), the tolerance is less than 20 parts per million (ppm). If the TX oscillator operates at 20 ppm above and the RX oscillator operates at 20 ppm below, then the error is 40 ppm. The result is a CFO value of up to 216 kHz [39].

2.10 Relevant work

2.10.1 Identification based on built-in magnetometer

The work in [40] shows how hardware can affect the characteristics of a device. It illustrates what can happen to the fingerprint of the mobile phone magnetometer if the soundboard of the computer is exposed to an external magnetic field generated by a solenoid. The fingerprint is quantified in both the time and frequency domain using statistical numbers, which are grouped into two feature sets, that will be discussed in detail. With the help of the magnetometer readings collected through an Android application called “Androsensor,” the required data is processed. Figure 2.15 gives a pictorial representation of the experiment setup used to collect the response through a mobile device using the “Androsensor,” application.

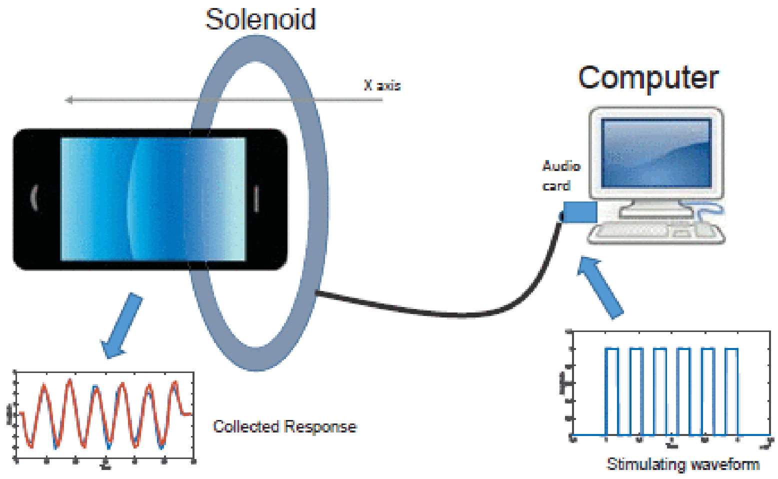


Figure 2.15 Experiment used to stimulate the magnetometer of a mobile phone [40]

A repeated sequence of signals (similar to the clock signals in this thesis work), is shown in Figure 2.16. The waveforms of two mobile phones (Blue and Red), generated in response to the waveform shown in Figure 2.16, are shown overlaid with one another in Figure 2.17.

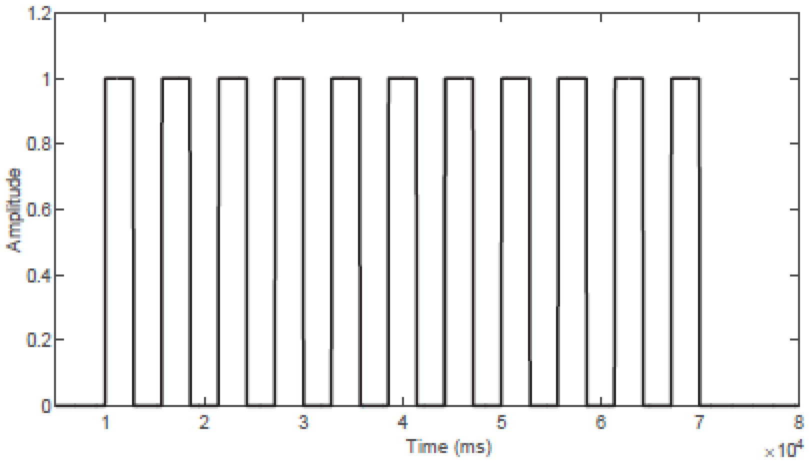


Figure 2.16 The waveform used to stimulate the magnetometer of the mobile phone [40]

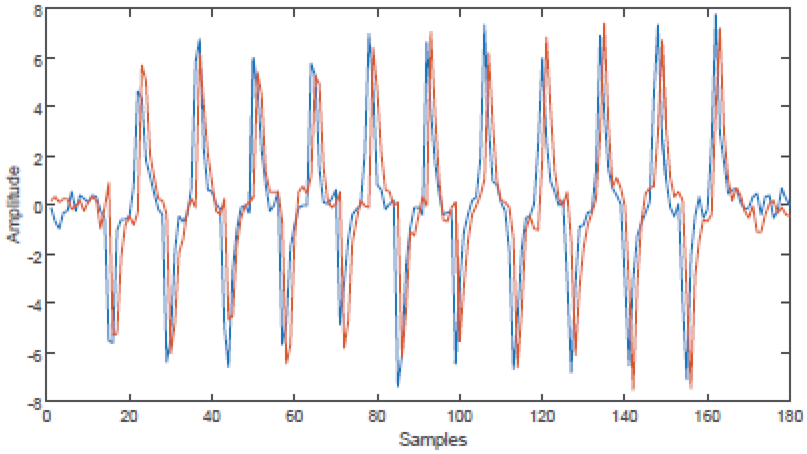


Figure 2.17 Different responses of the built-in magnetometer for two smartphones (blue and red) to the stimuli [40]

A set of nine mobile phones were subjected to this solenoid magnetic field, and for every case, the resulting form was registered. For fingerprinting quantification purpose, the responses were converted into statistical numbers, known as features. Two sets were used; the first feature set contains: Shannon entropy, Log energy entropy, Standard deviation, Variance, Skewness, and Kurtosis. The second feature set is listed in table 2.1.

Table 2.1 Time and frequency domain features used for features set2 [40]

Time Features	*Frequency Features*
Mean	Spectral Spread
Standard Deviation	Spectral Centroid
Average Deviation	Spectral Skewness
Skewness	Spectral Kurtosis
Kurtosis	Entropy
RMS	Flatness
Max	Roll Off
Min	Roughness
Non-negative count	Irregularity
Zero Crossing Rate (ZCR)	Spectral RMS
	Low Energy Rate

The first feature set has been used to generate the F-score (F-1), which is a test of accuracy, of the 3 different sound cards (SC1, SC2, and SC3). Two classifiers have been used, Support Vector Machine (SVM) and K-Nearest Neighbors (KNN) for this experiment. The results are shown in Table 2.2. From the results, there is a small difference between SC1 and SC2, but the best result was recorded for SC3. Regarding classifiers, SVM showed better F-scores compared to that of KNN. So, the use of different sound cards (i.e., external hardware) impacted the identification accuracy [40].

Table 2.2 F-score (F1) obtained with different sound cards; features set1, SVM and KNN classifiers [40]

SVM	F-score (F1)
SC1 SC2 SC3	0.741 0.685 0.877
KNN (K=5)	F-score (F1)
SC1 SC2 SC3	0.617 0.633 0.808
KNN (K=10)	F-score (F1)
SC1 SC2 SC3	0.613 0.663 0.901
KNN (K=15)	F-score (F1)
SC1 SC2 SC3	0.614 0.656 0.813

The second fingerprinting feature set was used for the same devices, but using the Bagged Decision Tree classifier. Table 2.3 shows the results of F-score for the same three devices, but with the Bagged Decision Tree classifier. The second feature set results in better classification performances (88% on average), for all three devices, when compared to the results using the first set of features. This indicates the more fingerprinting features used, the better classification results will be expected [40].

Table 2.3 Results in terms of F-score (F1) on the three considered audio boards: features set2, bagged decision tree classifier [40]

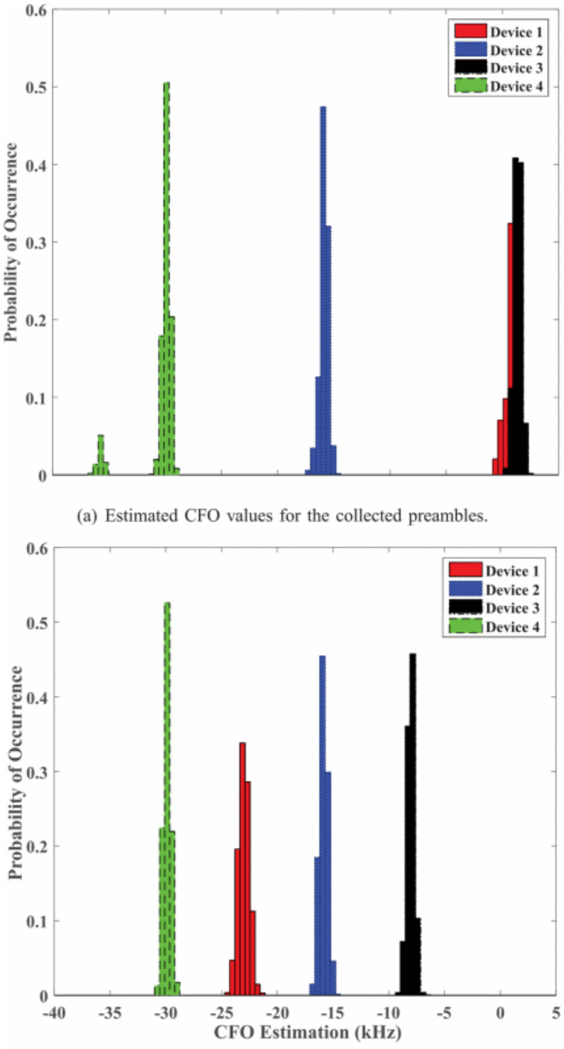
Bagged Decision Tree classifier	F-score F1
SC1	0.891
SC2	0.868
SC3	0.907

2.10.2 Assessment of the impact of CFO

One of the hardware components that can affect the transmission characteristics is the local oscillator. This component exists in both transmitter and receiver, but they do not generate the exact same frequency without some form of synchronization. When a mismatch between the two LOs occurs, the result is the presence of an offset value within the received waveform. This offset is referred to as carrier frequency offset (CFO) [10]. The receiver conducts CFO estimation through the use of STS nine and ten for a course estimate followed by the two LTS to achieve a fine estimate of the CFO value.

The authors in [10] used 1,000 Wi-Fi preambles collected from each of four 802.11a Wi-Fi radios via the same receiver. Figure 2.18(a) shows the Probability Mass Function (PMF) charts of the first case, where the receiver operated with the existing offset without making any alteration or correction. But in the second case, Figure 2.18(b) shows the PMF resulting from an intentional change (which is similar to what is done in this thesis work) by adding a known offset value at the transmitter. The result of this intentional insertion of unique CFO has shown a positive impact by displaying non-overlapping PMF’s of the four devices, which led to better discrimination of each device from the others. For the case of collected CFO, the PMFs of device 1 and device 3 overlap, which resulted in these two devices being confused for one another by the discrimination system.

In this thesis work, motivated by the work in [10], investigates the impact of intentional feature insertion using two cases, (i) the LimeSDR has operated using the non-modulated clock (i.e., no change has affected the transmitted signal) and the collected signal at receiver side is processed to generate the fingerprints, and (ii) feature insertion is facilitated by inserting an intentional change through an external clock (i.e., the external clock is changed via phase modulation using a known frequency or deviation value). The impact of this intentional feature insertion is studied and analyzed to see how this change affects the RF-DNA fingerprint. The RF-DNA fingerprint elements of the two cases are compared. Intentional feature insertion, via phase modulation of the external clock signal, is assessed using different frequencies and deviation values to determine the value at which this change negatively affects the receiver’s ability to demodulate the received signal.



(a) Estimated CFO values for the collected preambles.

(b) Generated unique CFO values assigned to each device.

Figure 2.18 PMF of CFO values for each of the 4 Wi-Fi devices [10]

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter describes the approaches used to study and monitor the effects resulting from the applied Phase modulated clock on RF-DNA fingerprints. First of all, the experiment setup section shows the hardware changes applied to the system board clocking system, how the PM clock is generated, the signal collection mechanism, and the conducted experiment environment. Then, the design used for the OFDM receiver is illustrated. It describes in detail the signal detection algorithm and how to calculate the parameters used for Wi-Fi preambles detection. Next, the GNU radio companion flow chart will be described for both the Wi-Fi transmitter and receiver. The received signals then undergo time synchronization and channel estimation. Afterward, the methods to represent the preamble in GT, and to generate the RF-DNA fingerprints are presented.

3.2 Experiment setup

3.2.1 External clock for LimeSDR-Mini

Figure 3.1 shows the LimeSDR-Mini board bottom connectors and main components. When the resistor 'R62' is fitted as shown by the orange rectangle, it means the external clock is enabled (i.e., the internal clock is disabled) [41]. The distribution system of this LimeSDR clock was shown previously in Figure 2.11.

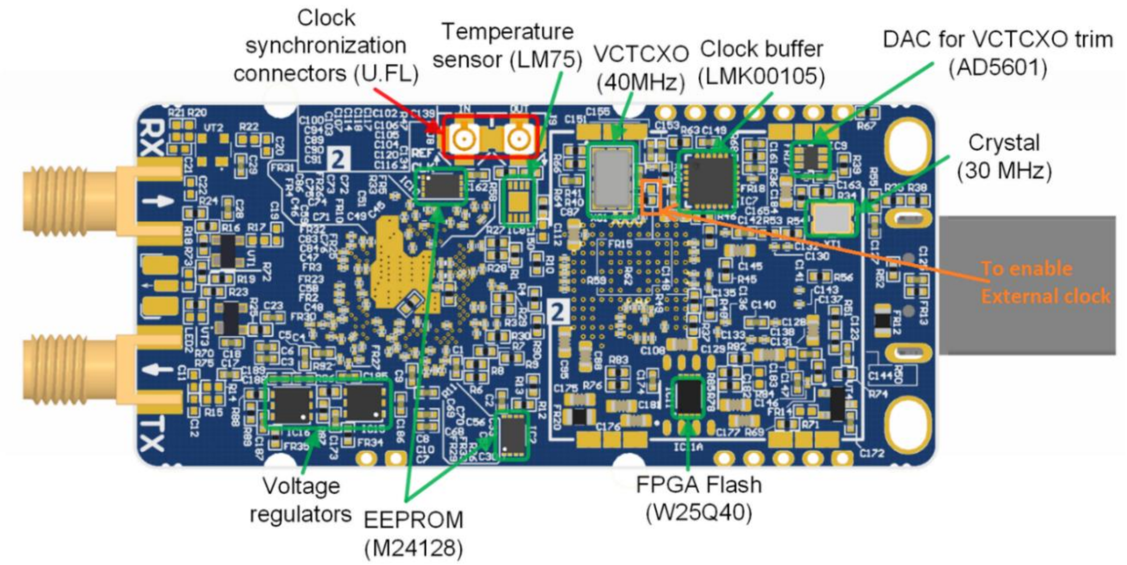


Figure 3.1 LimeSDR-Mini board bottom connectors and main components [22]

3.2.2 Phase-modulated clock

In this experiment, the transmitter's internal clock signal is replaced with an external clock signal that has been intentionally manipulated using phase modulation. The external clock was generated using a Tektronix AFG3252 signal generator. The external clock signal is a 50-duty cycle square wave of 40MHz, with a high voltage of 2.5 V and a low voltage of 100 mV, as shown in Figure 3.2.

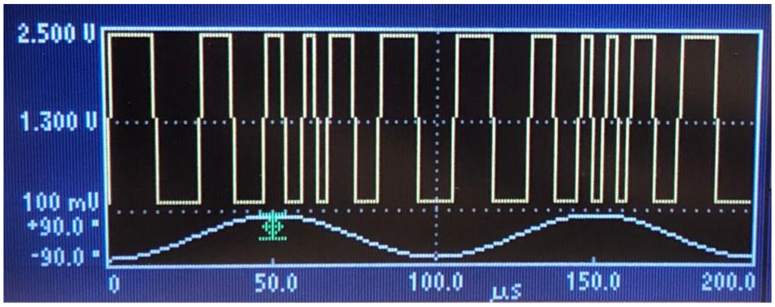


Figure 3.2 The phase-modulated clock using Tektronix AFG3252 signal generator

In the first case, 100 transmissions were collected using a second LimeSDR-Mini and no PM present (i.e., using a non-modulated clock) within the external clock signal. The received transmissions were collected for further processing. In the second case, an additional 100 transmissions were collected, however this time, PM of the external clock signal is conducted, using a sine-wave with a 90-degree deviation, as shown in Figure 3.3. The signal generator is capable of modulating the sine wave with frequency values between 0 and 50 kHz. The phase-modulated signal can be approximated by,

$$S = \text{sign}\ \{\sin(2\pi f_c t) + \Phi\}, \tag{3.1}$$

where S is the final phase modulated clock signal, ‘sign’ stands for the signum function (note that: sign(x) equals -1 if x< -1, equals 0 if x= 0, and 1 if x> 0), f_c is the carrier frequency, t is the time, and Φ is the phase given by,

$$\Phi = \sin(2\pi\ f_{PM}\, t + d), \tag{3.2}$$

where f_{PM} is the phase-modulated clock frequency, and d is the deviation. The modulated signal has d = π/2 to ensure a 90-degree deviation between the carrier signal and the modulation signal.

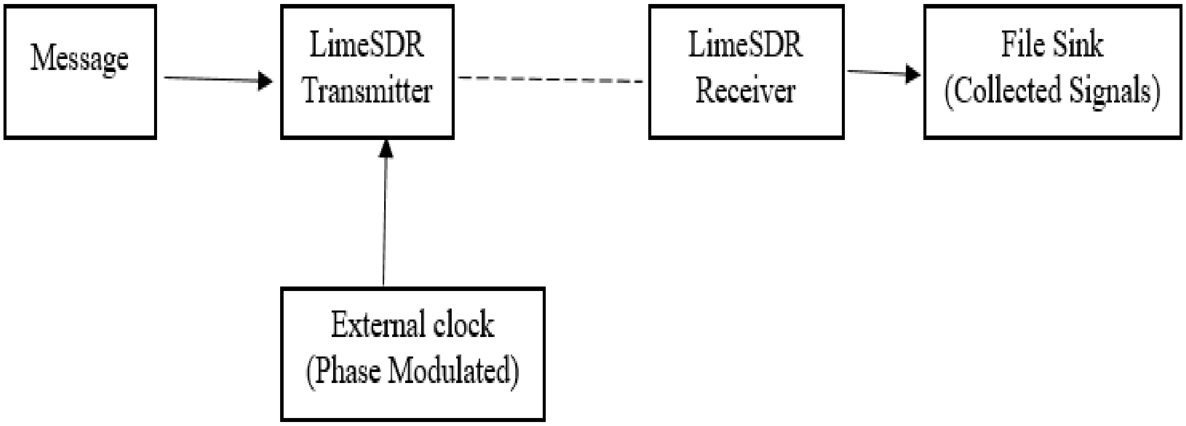

Message
LimeSDR Transmitter
LimeSDR Receiver
File Sink (Collected Signals)
External clock (Phase Modulated)

Figure 3.3 Block diagram of a transmitted signal using an external clock

3.2.3 Signal collection

All of the collected signals are stored as binary files using GNU radio companion. These binary files were then processed in MATLAB 2016. In addition to the non-modulated clock collected signal set, 802.11a transmissions were collected using PM frequencies f_{PM} of values {5, 10, 20, 30, 40, and 50} kHz. A total of 100 802.11a baseband Wi-Fi transmissions were collected per phase-modulated clock frequency. Following collection, individual transmissions were detected, and the preamble extracted for subsequent analysis.

3.2.4 Experiment environment

These experiments were conducted in the Communication laboratory, Electrical Engineering department at the University of Tennessee at Chattanooga. It is similar to every working environment where there is interference from the campus Wi-Fi and other neighboring signals in the ISM band. This band is congested because it is unlicensed, and many devices are equipped with 802.11 a/b/g transceivers. To avoid interference issues, these experiments were conducted in the amateur radio frequency range 3.3-3.5 GHz allowed by the U.S. Department of Commerce [42].

3.3 OFDM receiver

3.3.1 Signal detection algorithms

As mentioned in the Signal detection algorithm [43], the first 3 signals were designed and compared based on the short training sequence shown in (2.4). It was designed based on the correlator structure, as shown in Figure 3.4.

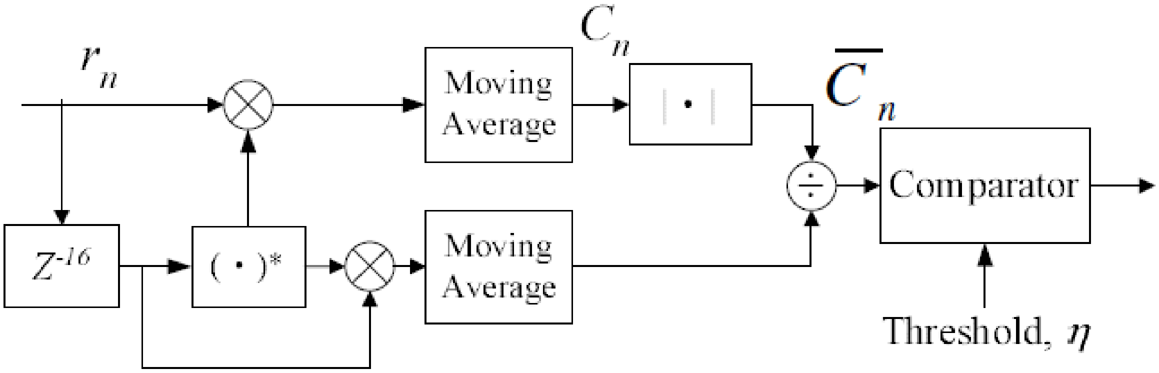


Figure 3.4 The block diagram of the signal detection algorithm [43]

The input sample was correlated with its delayed sample. According to the periodic characteristic, the delay amount was 16. After that, the correlated samples were averaged in the moving average block over a period of time L to suppress white noise. Using (2.4) and substitute L= 16, and N_{win}= 48 as authors of [26] found that empirically, the window size of 48 works well. So (2.4) becomes:

$$a[n] = \sum_{k=0}^{47} s[n+k]\,\overline{s}[n+k+16] \tag{3.3}$$

3.3.2 Accumulation of the correlation values

In general, the performance of the signal detection algorithms in severe channel conditions is poor. To obtain reliable signal detection, the correlation outputs of signal detection algorithms were accumulated over several short training symbols. Therefore, values in (3.3) were summed up symbol-by-symbol. The decision was made after the accumulation. For practical implementation, STSs t_1 to t_7 are utilized for AGC utilization while STSs t_8 to t_{10} for signal detection. Therefore, the valid accumulation length is about 3*0.8 = 2.4 μs, knowing that the sampling rate is 20 MHz [43].

3.3.3 Determination of the detection threshold

The determination of the detection threshold is the key parameter for correct signal detection. Due to the variation of the radio channel environment, the threshold should be set adaptively according to channel conditions. For the signal detection algorithm, the average energy of the received signal calculated at the lower branch was used to determine the detection threshold. According to Figure 3.4 and (3.3), the normalized correlation output at the receiver, R, was expressed as follows:

$$R = P_U / P_L \tag{3.4}$$

where P_U was upper branch output and P_L was the lower branch output received [43].

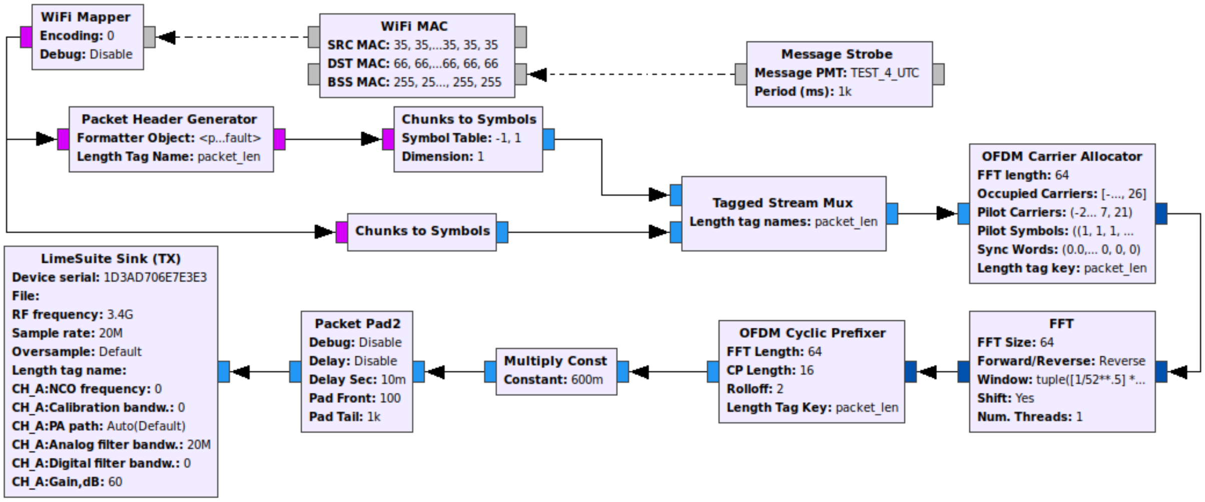
3.4 GNU radio companion

3.4.1 Flow chart

The experiments were designed to implement and observe the effects of a phase-modulated clock on RF-DNA fingerprints by using two LimeSDR-Mini devices for transmission and reception. The device parameters and operation were defined through a flow graph built within GNU Radio Companion that was designed to emulate OFDM transmissions [44]. This flowgraph was constructed based on the fundamental principles provided by [45]. Within GNU Radio Companion, the sampling rate was set to 20MHz, transmission frequency was set to 3.4 GHz, and the Wi-Fi message was sent periodically once every second. This transmission frequency was selected to avoid interference from other commercial Wi-Fi bands and to get the best Signal to Noise Ratio (SNR) possible.

3.4.2 Wi-Fi transmitter

Figure 3.5 shows the flowchart of the Wi-Fi transmitter. The blocks and their functions are: (i) Message Strobe block: responsible for defining the text message that will be sent and the time period between messages, (ii) The OFDM Mapper: receives the MCS as input and is responsible for multiple operations, such as the generation of the data field, in which it is including the tail and pad bits. Besides, it is also responsible for the scrambling and interleaving of the bits, (iii) Packet Header Generator: generates the header of the frame, including the signal and service fields. The header is BPSK modulated by the top Chunks to Symbols block, and the remaining frame is modulated by the bottom Chunks to Symbols block, according to the chosen modulation. Then, the header is finally joined to the remaining of the frame, (iv) OFDM Carrier Allocator: responsible for the aggregation of the pilot sub-carriers. (v) FFT block: responsible for the inverse FFT, i.e., for the transition from frequency to the time domain. (vi) OFDM Cyclic Prefixer block: aggregates the guard intervals to each symbol of the frame. (vii) Lime Suite Sink block: defines the parameters on the LimeSDR-Mini board, such as the sampling rate and the transmission frequency [46].



WiFi Mapper
Encoding: 0
Debug: Disable

WiFi MAC
SRC MAC: 35, 35,...35, 35, 35
DST MAC: 66, 66,...66, 66, 66
BSS MAC: 255, 25..., 255, 255

Message Strobe
Message PMT: TEST_4_UTC
Period (ms): 1k

Packet Header Generator
Formatter Object: <p...fault>
Length Tag Name: packet_len

Chunks to Symbols
Symbol Table: -1, 1
Dimension: 1

Chunks to Symbols

Tagged Stream Mux
Length tag names: packet_len

OFDM Carrier Allocator
FFT length: 64
Occupied Carriers: [-..., 26]
Pilot Carriers: (-2... 7, 21)
Pilot Symbols: ((1, 1, 1, ...
Sync Words: (0.0,... 0, 0, 0)
Length tag key: packet_len

LimeSuite Sink (TX)
Device serial: 1D3AD706E7E3E3
File:
RF frequency: 3.4G
Sample rate: 20M
Oversample: Default
Length tag name:
CH_A:NCO frequency: 0
CH_A:Calibration bandw.: 0
CH_A:PA path: Auto(Default)
CH_A:Analog filter bandw.: 20M
CH_A:Digital filter bandw.: 0
CH_A:Gain,dB: 60

Packet Pad2
Debug: Disable
Delay: Disable
Delay Sec: 10m
Pad Front: 100
Pad Tail: 1k

Multiply Const
Constant: 600m

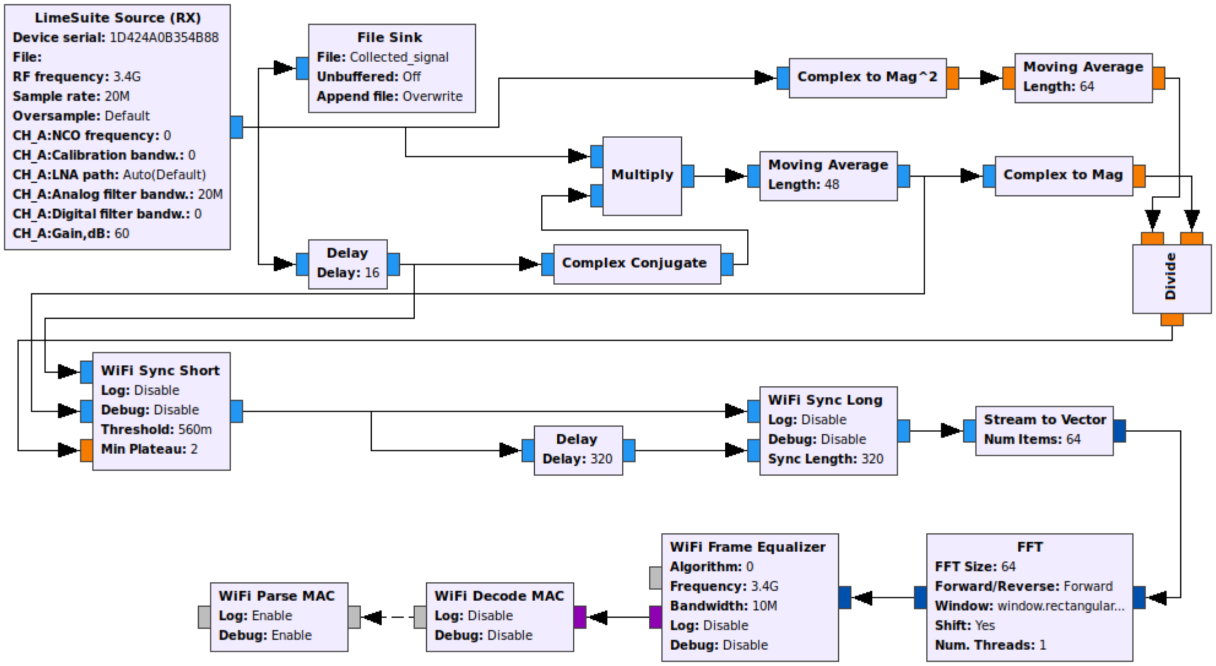
OFDM Cyclic Prefixer
FFT Length: 64
CP Length: 16
Rolloff: 2
Length Tag Key: packet_len

FFT
FFT Size: 64
Forward/Reverse: Reverse
Window: tuple([1/52**.5] *...
Shift: Yes
Num. Threads: 1

Figure 3.5 Wi-Fi transmitter GNU radio companion flowchart

3.4.3 Wi-Fi receiver

Figure 3.6 shows the flowchart of the Wi-Fi receiver. The blocks and their functions are: (i) Lime Suite Source block: defines the parameters on the LimeSDR-mini board, such as the sampling rate and the transmission frequency, (ii) File Sink block: used to collect the received signal and save it as a binary file, (iii) WiFi Sync short: uses autocorrelation to decide if packet is accepted or dropped based on specific threshold value (here 560 mV is used), (iv) Wi-Fi Sync long: responsible for symbol alignment to achieve the timing synchronization. (v) FFT block: responsible for the forward FFT, i.e., for the transition from time to frequency domain [6].



LimeSuite Source (RX)
Device serial: 1D424A0B354B88
File:
RF frequency: 3.4G
Sample rate: 20M
Oversample: Default
CH_A:NCO frequency: 0
CH_A:Calibration bandw.: 0
CH_A:LNA path: Auto(Default)
CH_A:Analog filter bandw.: 20M
CH_A:Digital filter bandw.: 0
CH_A:Gain,dB: 60

File Sink
File: Collected_signal
Unbuffered: Off
Append file: Overwrite

Complex to Mag^2

Moving Average
Length: 64

Multiply

Moving Average
Length: 48

Complex to Mag

Delay
Delay: 16

Complex Conjugate

Divide

WiFi Sync Short
Log: Disable
Debug: Disable
Threshold: 560m
Min Plateau: 2

Delay
Delay: 320

WiFi Sync Long
Log: Disable
Debug: Disable
Sync Length: 320

Stream to Vector
Num Items: 64

FFT
FFT Size: 64
Forward/Reverse: Forward
Window: window.rectangular...
Shift: Yes
Num. Threads: 1

WiFi Frame Equalizer
Algorithm: 0
Frequency: 3.4G
Bandwidth: 10M
Log: Disable
Debug: Disable

WiFi Decode MAC
Log: Disable
Debug: Disable

WiFi Parse MAC
Log: Enable
Debug: Enable

Figure 3.6 Wi-Fi receiver GNU radio companion flowchart

3.5 RF-DNA fingerprinting

3.5.1 Gabor transform

The RF-DNA fingerprints are generated using the concept in background section 2.7.1, explained in [32]. The procedure utilizes the GT to jointly represent the momentary temporal-spectrum (T-F) variations that occur within a waveform. The DGT was calculated using (2.12), a Gaussian examined window $W(n)$, and the variables defined in Section 2.6 [6]. The response of normalized magnitude from the temporal-spectrum plane is calculated using (2.13) and subsequently divided into N_R patches. Each patch has a total of $N_T \times N_F$ Gabor coefficients and is reshaped into an N_{TF} length vector. The variance (σ^2), standard deviation (σ) kurtosis (k), and skewness (γ) are used as quantification statistics, which are calculated from this vector and used to quantify the RF-DNA fingerprint corresponding to the signal under study $\hat{X}(M)$. The parameters values that will be used in RF-DNA fingerprints are $M = 186$, $K_G = 186$, $N_\Delta = 1$, $N_{TF} = 120$, $N_T = 12$, and $N_F = 10$.

3.5.2 Fingerprint elements

From background section 2.7.1, features in this work are generated by calculating three statistics: variance (σ^2), skewness (γ), and kurtosis (k). The standard deviation is not used here because the variance gives enough information about how the data are spread among any specific population samples. So the calculated statistics, for each of the selected T-F sub-regions, are arranged as follows:

$$f_{Ri} = \{\sigma^2_{Ri}\ , \gamma_{Ri}\ , k_{Ri}\ \}_{1\times3}\ , \tag{3.5}$$

where i = 1, 2, . . ., 120.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This chapter shows the results and provides discussion and analysis for the effects of PM clock on RF-DNA fingerprints in IEEE 802.11a. Firstly, the time domain and the GT of the transmitted signal were plotted to show how the preamble is constructed before getting into hardware to be transmitted into the channel. Next, the non-modulated external clock is applied to the circuit, and the received preamble is extracted and shown in the I-Q plot and transformed into the T-F domain (GT). After that, the results of PM external clock are shown in (i) I-Q format, (ii) GT, and (iii) fingerprint elements statistics. Then, comparison and analysis are shown between the non-modulated and PM external clock effects. The demodulation breakpoint is defined for the Phase modulated clock frequency, the degree of deviation, and the CFO. Finally, an analysis of the impact of a PM clock on the fingerprint is presented.

4.2 The transmitted signal

Figure 4.1 shows the transmitted signal, which is a simple text message, using a GNU radio companion before it has been propagated through the LimeSDR device. The first 16 µs show the Wi-Fi preamble, followed by the payload data. The real portion is shown in blue while the imaginary portion is shown in red.

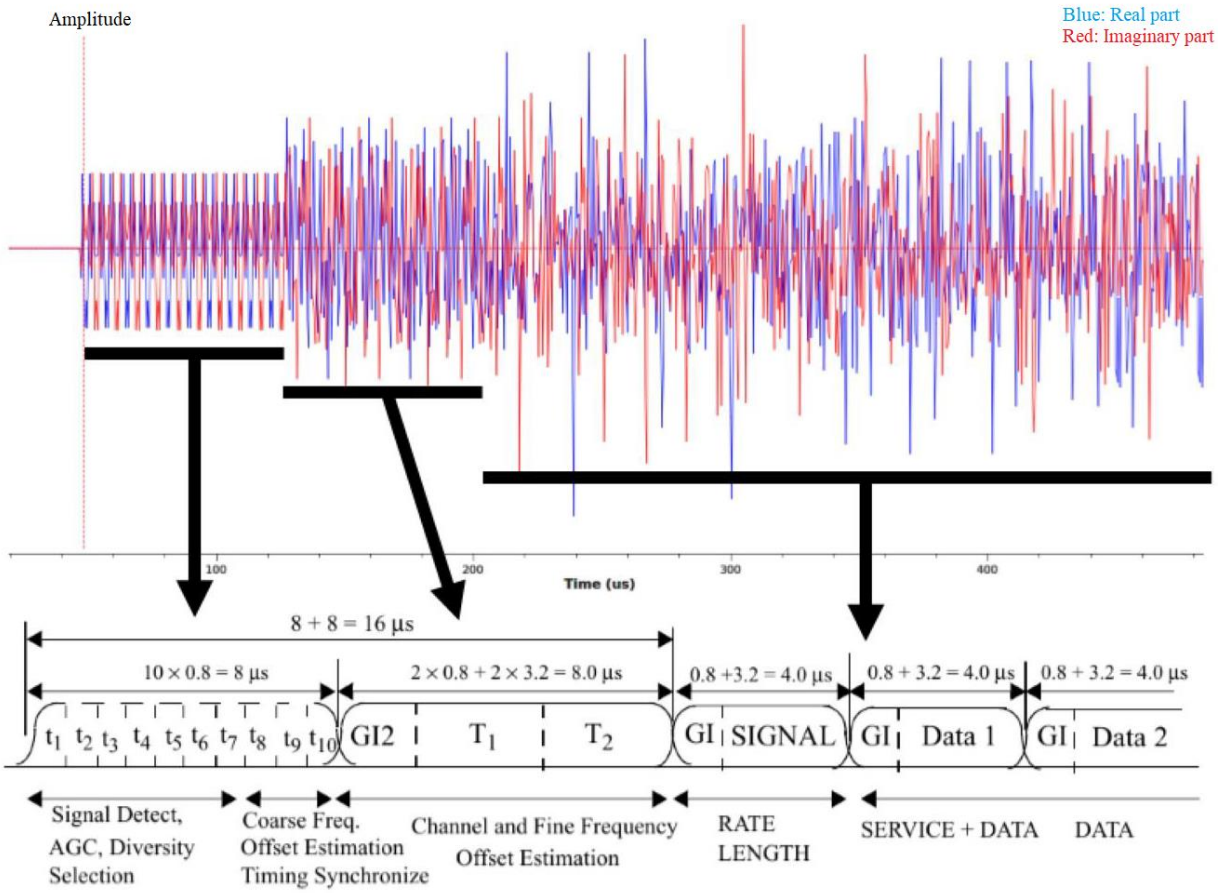


Figure 4.1 Wi-Fi transmitted signal mapping with the packet header

The Gabor Transform of the transmitted signal is shown in Figure 4.2. The frequency axis shows the spectrum of the Wi-Fi signal. The blue area in the middle indicates low SNR values, while the red side lobes indicate high SNR. The Time axis presents the STSs (0 to 8μs), GI (8 to 9.6 μs), and the LTSs (9.6 to 16 μs). Because this signal was not affected yet by the channel conditions, all of the preamble structure elements are clear and distinguishable.

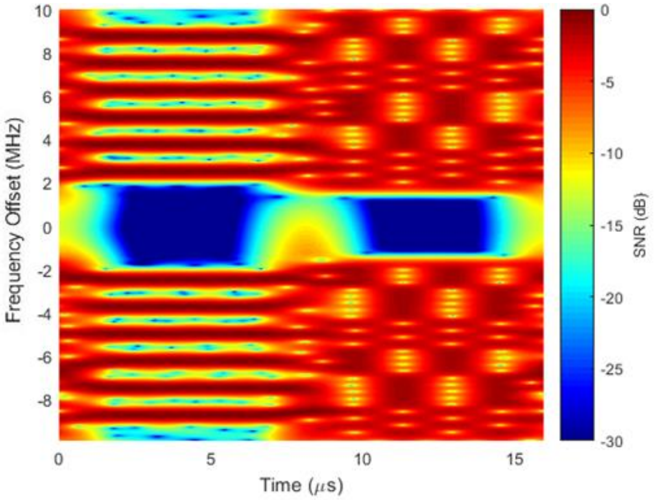


Figure 4.2 Gabor transform of the transmitted signal

4.3 Non-modulated external clock

The external clock has been applied with a 40 MHz sampling rate without modulation. Figure 4.3 shows the received preamble. The preamble plot indicates that the signal is received without changes affecting the main structure of the preamble. The peak value of the STSs amplitudes is almost the same. The same observation applies to the LTSs.

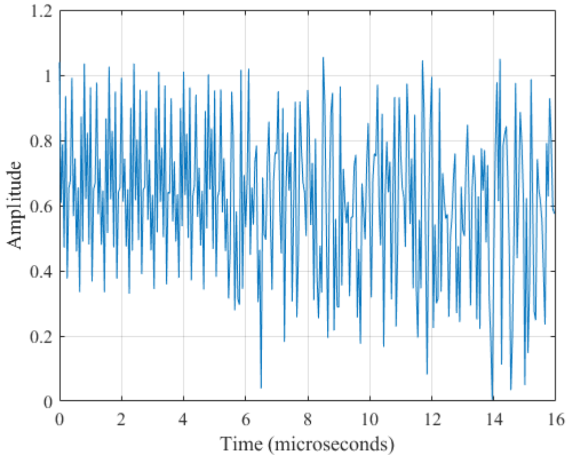


Figure 4.3 Collected signal preamble for non-modulated external clock

Figure 4.4 shows the Gabor transform of the same preamble (shown previously in Figure 4.3). It is similar to the GT of the transmitted signal, Figure 4.2, with some differences in the center values of frequency offset, where here it shows higher SNR values relatively.

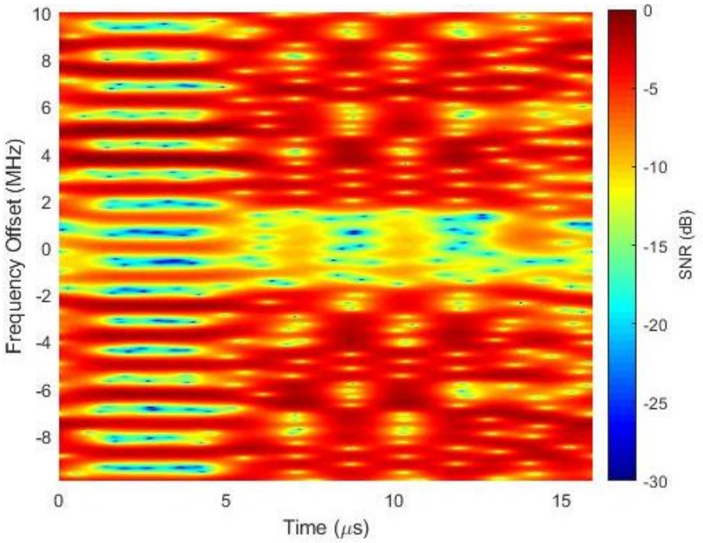
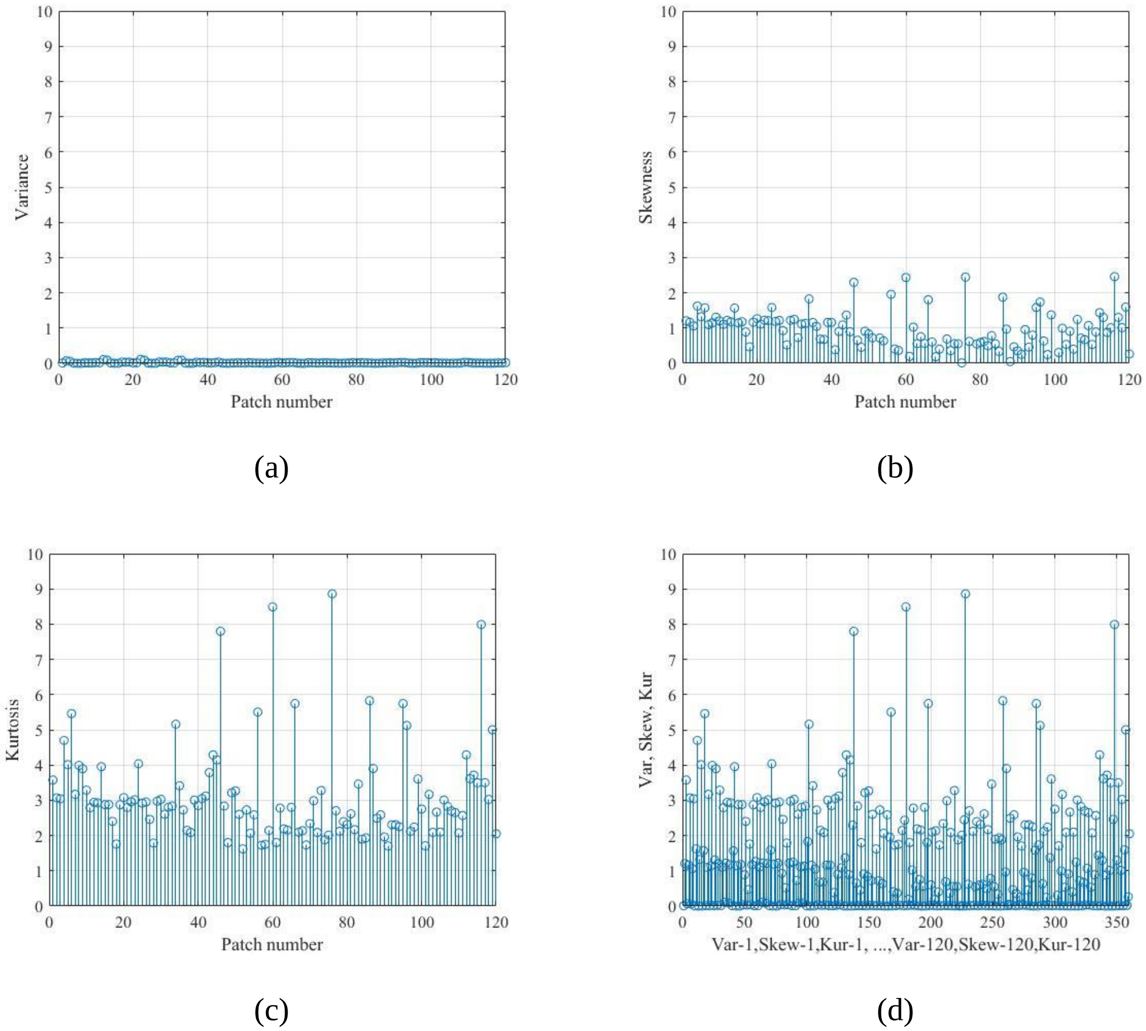


Figure 4.4 Gabor transform of the collected signal preamble for non-modulated external clock

The fingerprint elements are shown in Figure 4.5. They are generated by dividing the GT chart into 120 equal patches, and the required statistics are calculated for every patch. Figure 4.5a shows the variance (σ^2), Figure 4.5b shows the skewness (γ), and Figure 4.5c shows kurtosis (k), for every patch. Figure 4.5d shows the three statistics in one graph. These statistics will be compared later against the fingerprint statistics generated from the PM external clock impact.



(a)

(b)

(c)

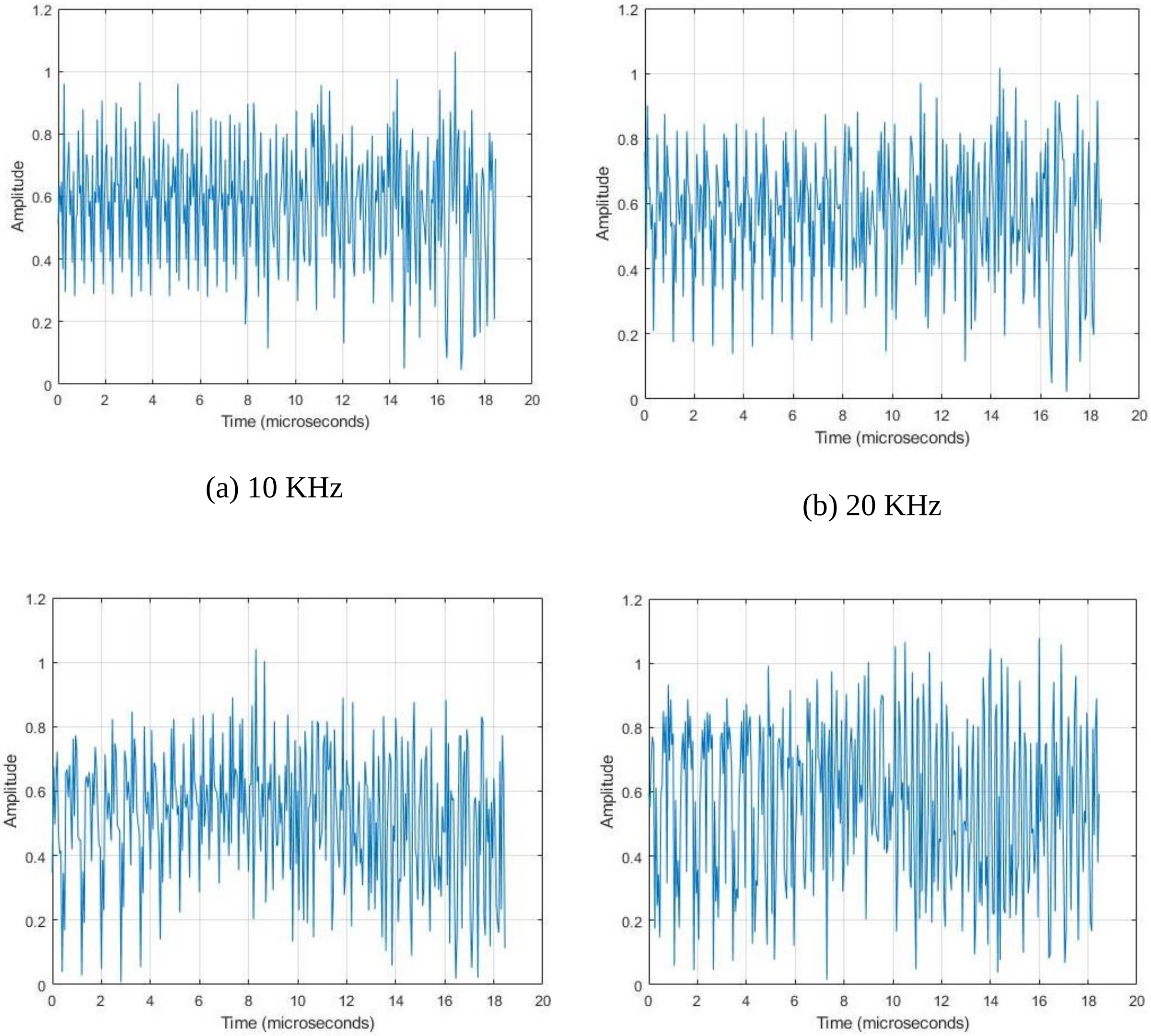
(d)

Figure 4.5 Fingerprint statistics: (a) variance, (b) skewness, (c) kurtosis, and (d) all (variance, skewness, and kurtosis) of the collected signal preamble for the non-modulated clock

4.4 PM external clock

4.4.1 Extracted preambles in I-Q format

Figure 4.6 shows the preambles for PM frequencies: 10, 20, 30, and 40 kHz inserted into the function generator external clock with clocking rate 40 MHz, collected at the receiver side. The first two frequency values of (a) 10 kHz, and (b) 20 kHz did not change the preamble envelope. Starting from 30 kHz, the sinusoid pattern of the clock appears. When the PM frequency increases, the more effects appear within the preamble as shown for (c) 30 kHz, and (d) 40 kHz.



(a) 10 KHz

(b) 20 KHz

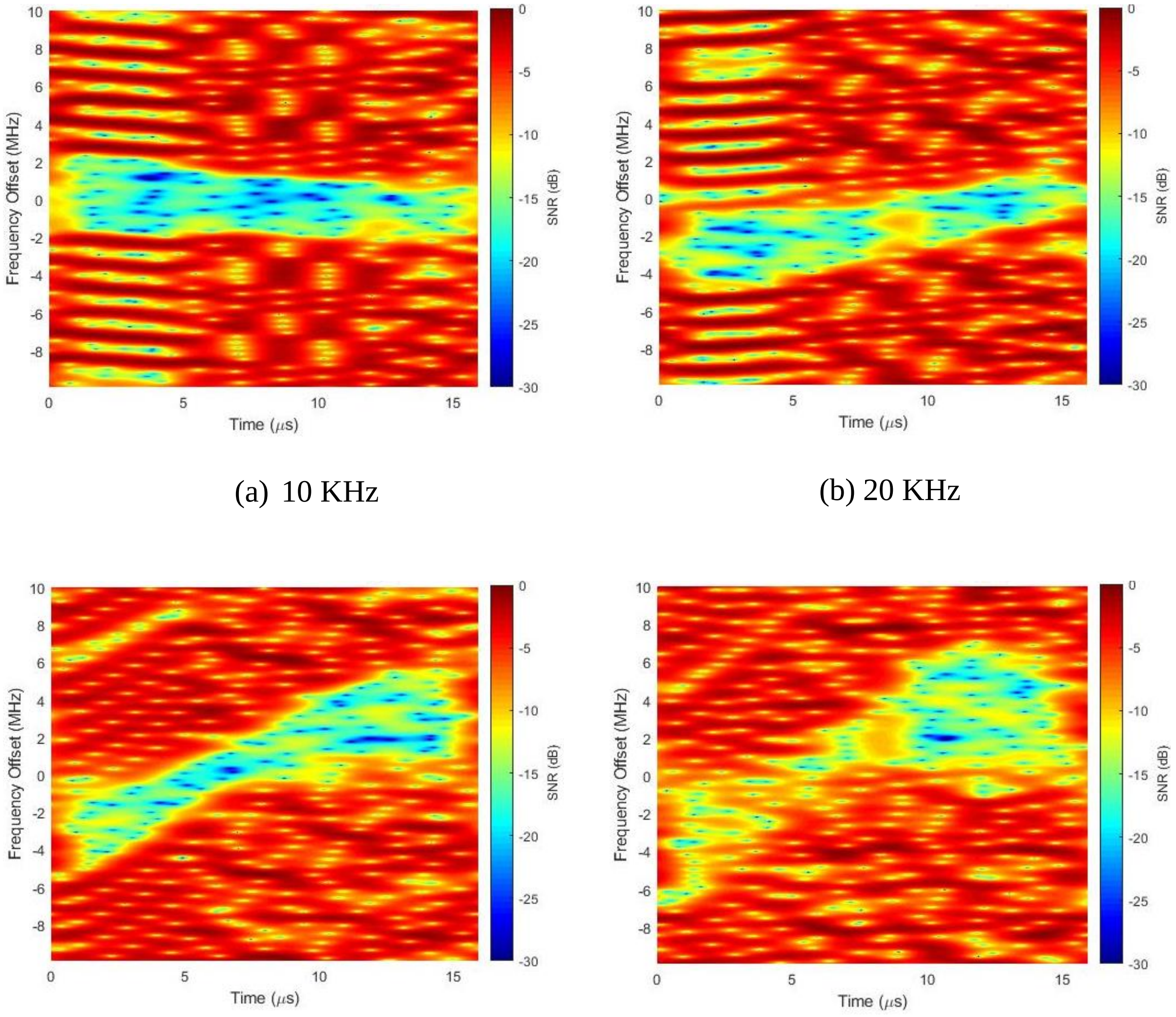
(c) 30 KHz

(d) 40 KHz

Figure 4.6 Preambles for PM frequencies: 10, 20, 30, and 40 kHz inserted into the function generator external clock with clocking rate 40 MHz, collected at the receiver side

4.4.2 Gabor transform of the extracted preambles

Figure 4.7 shows the Gabor transform of PM frequencies 10, 20, 30, and 40 kHz, with 90^{o} deviation, inserted into the function generator external clock with clocking rate 40 MHz, collected at the receiver side. After the 30 kHz, the STSs side lobes disappear, and sinewave variation appears with time at the center of the frequency offset.



(a) 10 KHz

(b) 20 KHz

(c) 30 KHz

(d) 40 KHz

Figure 4.7 Gabor transform of PM frequencies 10, 20, 30, and 40 kHz, with 90^o deviation, inserted into the function generator external clock with clocking rate 40 MHz, collected at the receiver side

4.5 Non-modulated vs. PM clock

Figure 4.8 shows the DGT plots of no modulation clocking (left) and 50 kHz PM (right). These graphs depict the signal over time, using SNR versus frequency offset. The 0 MHz point in the vertical axis corresponds to the carrier frequency of f_c = 3.4 GHz. As the phase-modulated clock frequency (f_{PM}) increases, the transmission experiences a frequency offset shift over time.

This offset shifts in a sinusoidal pattern due to the use of a sinusoidal PM signal and the max frequency offset achieved in each sample increases as the modulation frequency increases. A relatively small modulation frequency (e.g., 5 kHz), the frequency offset achieves an absolute, maximum of approximately 1 MHz. At larger PM frequencies (e.g., 50 kHz), the frequency offset achieves an absolute, maximum of approximately 7 MHz. It is important to note that the structure of the STS and LTS, within the DGT (Figure 4.8 Right), is distorted as the PM frequency reaches values of 30 kHz and higher.

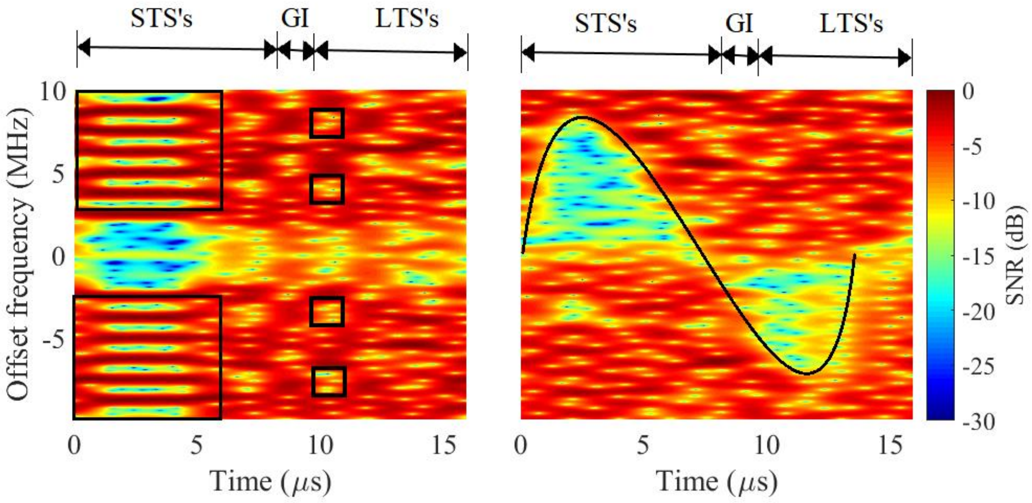


Figure 4.8 The DGT plots of no modulation (left) and 50 KHz of phase modulation with 90o deviation (right) inserted into the function generator external clock with clocking rate 40 MHz, collected at the receiver side

In the DGT graphs, the short training sequences appear as the side loop pulses located above and below the center frequency transmission between 0 and 8 μs. Guard Interval appears between 8 and 9.6 μs. The long training sequences appear as the repeated square pattern of high and low SNR above and below the center frequency transmission between 9.6 and 16 μs.

4.6 External clock demodulation breakpoint

The breakpoint is defined here as the point that the demodulator cannot reconstruct the received data after it. The effects caused by the external clock are related to two variables, which are Phase modulated clock frequency (f_{PM}) and the deviation (d), shown previously in (3.1) and (3.2). The tested breakpoints are (i) f_{PM} = 10 KHz, d = 1.5^{o}, and (ii) f_{PM} = 2.5 KHz, d = 90^{o}. The first tested breakpoint is shown in Figure 4.9. It shows the preamble (Left) and the GT (Right) of breakpoint at f_{PM} = 10 KHz, d = 1.5^{o}.

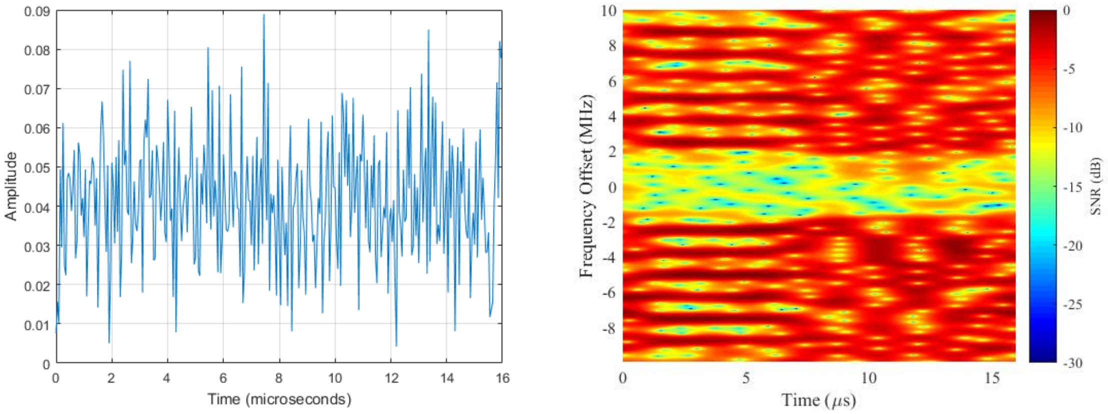


Figure 4.9 Preamble (left) and the GT (right) of f_{PM} = 10 KHz, d = 1.5^{o} inserted into the function generator external clock with clocking rate 40 MHz, collected at the receiver side

Figure 4.10 shows the preamble (Left) and the GT (Right) of the second tested breakpoint at f_{PM} = 2.5 KHz, d = 90^{o}. It has been noticed that the preamble envelope has shown small variation with the sinusoidal function. The GT has its STSs and LTSs shown in the plot without major effects.

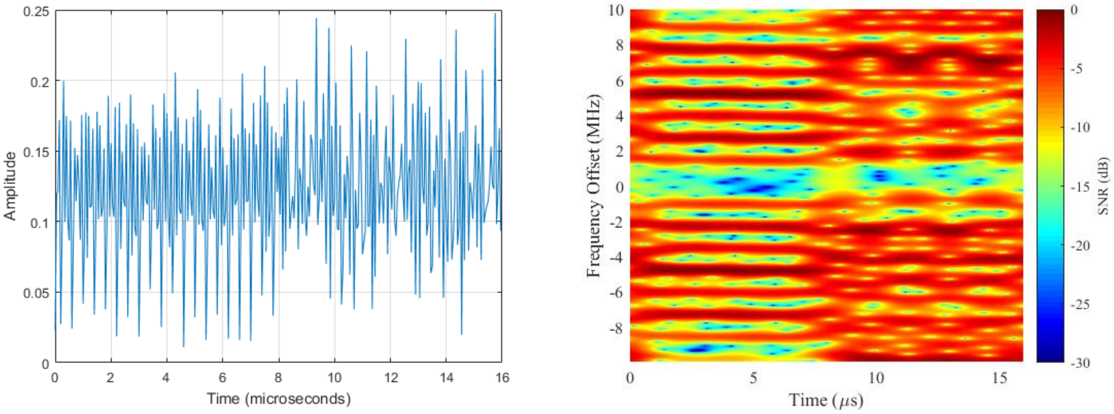


Figure 4.10 Preamble (left) and the GT (right) of f_{PM} = 2.5 KHz, d = 90^o inserted into the function generator external clock with clocking rate 40 MHz, collected at the receiver side

4.7 CFO demodulation breakpoint

The transmitter was adjusted to transmit at different frequency values than the receiver's frequency. The maximum demodulation frequency offset between the transmitter and receiver is 155 kHz. Beyond this offset value, the receiver cannot reconstruct the transmitted data. Figure 4.11 shows the Preamble (Left) and the GT (Right) of carrier offset of 155 kHz using a non-modulated external clock.

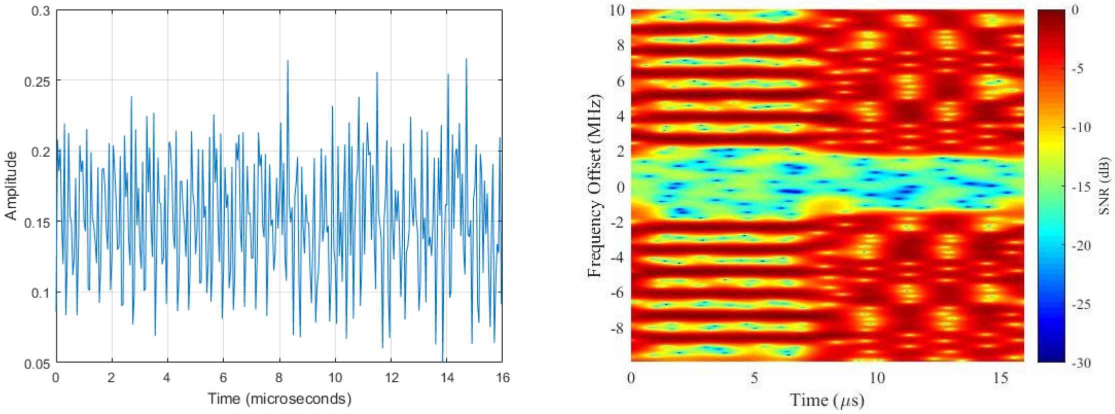


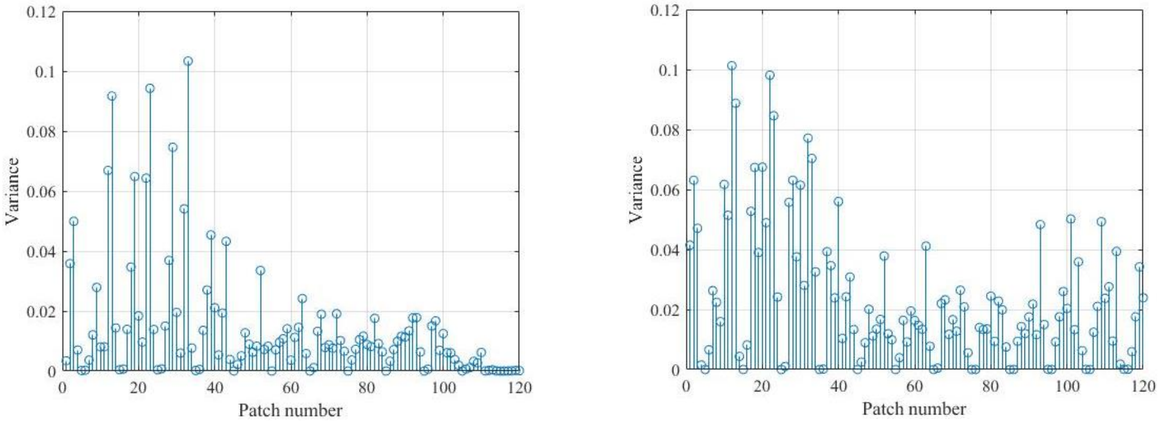
Figure 4.11 Preamble (left) and the GT (right) of carrier offset of 155 kHz using a non-modulated external clock., collected at the receiver side.

4.8 Breakpoint discussion

From the breakpoint sections 4.6 and 4.7, these values were experimentally found because one of the variables were kept small, which was the deviation in first case (d = 1.5^o), Figure 4.10, while changing the frequency, but in the previous experiment where the deviation was 90^o and the f_{PM} = 10 KHz, Figure 4.7, the receiver cannot demodulate the as it has already passed beyond the breakpoint. The same applied to the second test shown in Figure 4.10, f_{PM} = 2.5 KHz, d = 90^o, here the deviation is a high value, so the phase-modulated clock frequency cannot go beyond the 2.5 kHz before the demodulation is lost. For the CFO case, Figure 4.11, the external clock is not modulated, but the offset value itself has some limitations before the receiver cannot demodulate the signal.

4.9 Analysis of the impact of a PM clock on the fingerprint

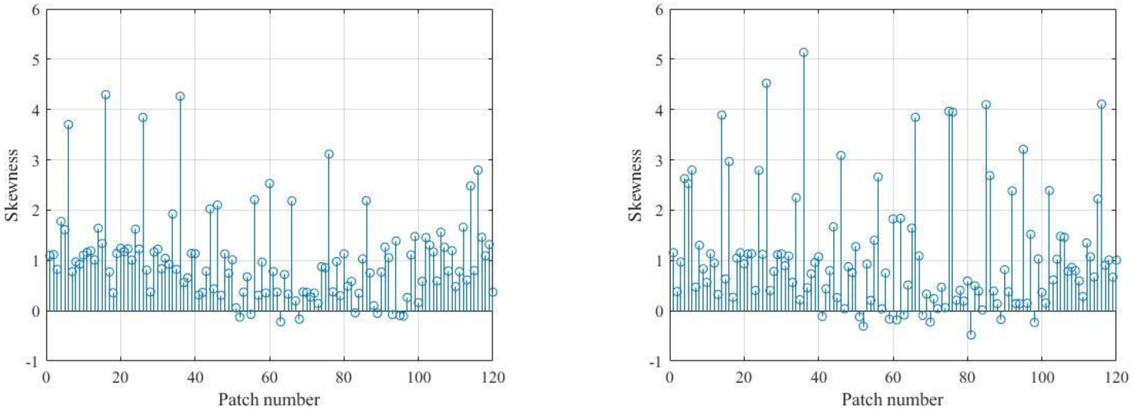
In this section, a comparison is made between the non-modulated clock and the phase-modulated clock with frequency 10 kHz and 90^o deviation. The target of this comparison is to show the impact of the PM clock. Figure 4.12 shows a comparison of variance values.



(a) (b)

Figure 4.12 The variance of (a) non-modulated clock, and (b) PM external frequency of 10 kHz, with 90^o deviation, collected at the receiver side

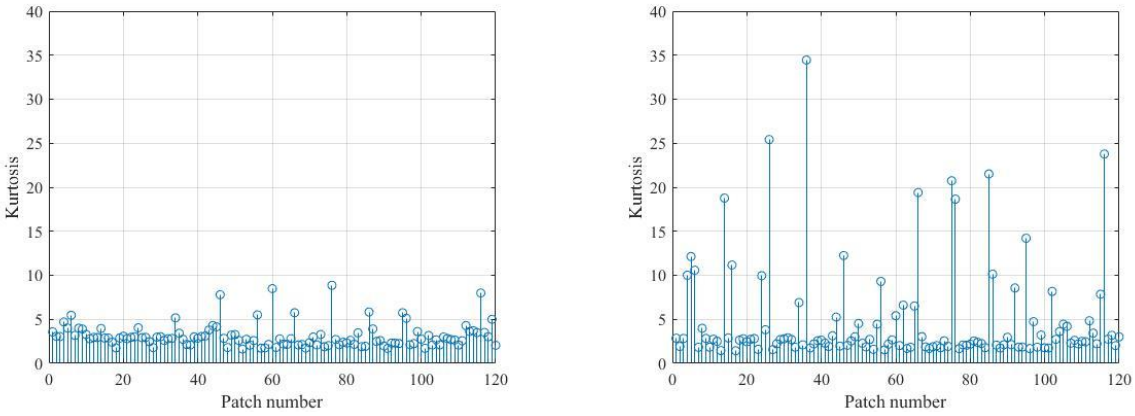
Figure 4.13 shows a comparison of skewness values between the non-modulated clock and the PM clock with frequency 10 kHz and 90^o deviation. As shown, there is a slight increase in max values of some phase-modulated clock patches compared to the non-modulated clock patches.



(a) (b)

Figure 4.13 The skewness of (a) non-modulated clock, and (b) PM external clock frequency of 10 kHz with 90^o deviation, collected at the receiver side.

Figure 4.14 shows a comparison of kurtosis values between the non-modulated clock and the phase-modulated clock with frequency 10 kHz and 90^{o} deviation. Their values of kurtosis have increased for the patches of the phase-modulated clock compared to the non-modulated clock patches.



(a)

(b)

Figure 4.14 Kurtosis of (a) non-modulated clock, and (b) PM external clock frequency of 10 kHz with 90^{o} deviation, collected at the receiver side

Table 4.1 shows the average values of the statistics. Average values of the variance are almost the same, while the average of skewness has a small increase for the PM clock. The average value for the kurtosis has resulted in 0.9 increase, in which the PM clock effect is clear.

Table 4.1 Average values of fingerprint elements for (a) non-modulated clock, and (b) PM modulated clock with frequency 10 kHz

	(a) Non-modulated clock	(b) PM modulated clock with frequency 10 kHz
Average Variance	0.0275	0.0235
Average Skewness	1.0106	1.0907
Average Kurtosis	3.8568	4.7652

Table 4.2 shows the variance values of the statistics. The variance of variance is very small and rounded to ‘0’ for both cases, but there is a small increase in the skewness variance. The noticeable change has appeared in the increased value of Kurtosis variance.

Table 4.2 Variance values of fingerprint elements for (a) Non-modulated clock, and (b) PM modulated clock with frequency 10 kHz

	(a) Non-modulated clock	(b) PM modulated clock with frequency 10 kHz
“Variance” variance	0.0000	0.0000
Skewness variance	0.7278	1.3314
Kurtosis variance	14.9332	31.8521

From these comparisons, it has been noticed that the external PM clock does not have a big impact on the variance nor skewness, but the greatest impact is on the kurtosis. From statistical point of view, the higher value of the kurtosis means that distribution of the data has tails exceeding the normal distribution (i.e., it has heavy tails, or outliers) [47].

CHAPTER 5

CONCLUSION

This work investigated the effects of intentional (clock signal phase manipulation) on RF-DNA fingerprinting features extracted from 802.11a Wi-Fi preambles. Intentional manipulation was implemented by phase modulating the clock signal used to generate the carrier signal. PM frequencies of 30 kHz or more have led to losses of the short and long training sequences within a preamble. This will negatively impact the RF-DNA fingerprint features and subsequent device identification performance. The demodulation breakpoints show to what extent the receiver can adapt to the changes occurred at the transmitter side and also illustrated that increasing more than one value at a time (i.e., the phase-modulated clock frequency and deviation together) can lead to reach the breakpoint earlier as many changes occur, and the receiver cannot compensate for these changes to reconstruct the received message. The effect of PM clock has been empirically tested among the fingerprint elements and found that (i) it has minor effect on variance, and skewness, and (ii) the greatest impact is on the kurtosis, which has increased from 14.9332 to 31.9332.

5.1 Future work

Preliminary analysis of the sinusoidal envelope of the preambles and the following DGT plots at 50 kHz modulation show a half-cycle period of approximately 10 μs and, therefore, a full-cycle period of 20 μs. This matches the period of a 50 kHz wave, indicating that the effect of the modulation is direct and possibly predictable. This leaves open further research opportunities in both predicting the modulation effect at varying frequencies and creating a system that can reverse the effects of the PM to provide the original, unmodulated signal within the receiver.

The effect of temperature is one of the work opportunities that has been started in the communication laboratory, but the temperature chamber was not there to conduct experiments to study its effects on the RF-DNA fingerprints.

REFERENCES

[1] Toonstra and W. Kinsner, "A Radio Transmitter Fingerprinting System ODO-1," in *Proceedings of 1996 Canadian Conference on Electrical and Computer Engineering*, May 1996, vol. 1, pp. 60-63 vol.1, doi: 10.1109/CCECE.1996.548038.

[2] G. Baldini and G. Steri, "A Survey of Techniques for the Identification of Mobile Phones Using the Physical Fingerprints of the Built-In Components," *IEEE Communications Surveys & Tutorials,* vol. 19, no. 3, pp. 1761-1789, 2017, doi: 10.1109/COMST.2017.2694487.

[3] M. D. Williams, S. A. Munns, M. A. Temple, and M. J. Mendenhall, "RF-DNA Fingerprinting for Airport WiMax Communications Security," in *4th International Conference on Network and System Security,* Sep 2010, pp. 32-39, doi: 10.1109/NSS.2010.21.

[4] R. W. Klein, M. A. Temple, and M. J. Mendenhall, "Application of Wavelet-based RF Fingerprinting to Enhance Wireless Network Security," *Communications and Networks Journal,* vol. 11, no. 6, pp. 544-555, 2009, doi: 10.1109/JCN.2009.6388408.

[5] "Internet of Things (IoT) Connected Devices Installed Base Worldwide from 2015 to 2025 (in Billions)." https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/ (accessed 2019-09-09).

[6] D. Reising, M. Temple, and J. Jackson, "Authorized and Rogue Device Discrimination Using Dimensionally Reduced RF-DNA Fingerprints," *Information Forensics and Security IEEE Transactions,* vol. 10, no. 6, pp. 1180-1192, 2015, doi: 10.1109/TIFS.2015.2400426.

[7] "OSI Model Reference Guide." https://www.lifewire.com/osi-model-reference-guide-816289 (accessed 2019-09-09).

[8] "What Risks do IoT Security Issues pose to Businesses?" https://blog.avast.com/iot-security-business-risk (accessed 11-14-2019).

[9] Z. Zhang, M. C. Y. Cho, C. Wang, C. Hsu, C. Chen, and S. Shieh, "IoT Security: Ongoing Challenges and Research Opportunities," in *2014 IEEE 7th International Conference on Service-Oriented Computing and Applications*, 17-19 Nov. 2014 2014, pp. 230-234, doi: 10.1109/SOCA.2014.58.

[10] C. G. Wheeler and D. R. Reising, "Assessment of the Impact of CFO on RF-DNA Fingerprint Classification Performance," in *Computing, Networking and Communications*

(ICNC) International Conference, 26-29 Jan. 2017, pp. 110-114, doi: 10.1109/ICCNC.2017.7876111.

[11] A. K. Jain, H. Lin, S. Pankanti, and R. Bolle, "An Identity-authentication System Using Fingerprints," *Proceedings of the IEEE,* vol. 85, no. 9, pp. 1365-1388, 1997, doi: 10.1109/5.628674.

[12] M. Ergen, "IEEE 802.11 Tutorial," University of California Berkeley, 2002.

[13] C. d. M. C. a. D. P. Agrawal, *Ad-hoc and Sensor Networks: Theory and Applications* (World scientific). 2011.

[14] R. Prasad, *OFDM for Wireless Communications Systems*. Boston: Artech House, 2004.

[15] "IEEE 802.11 OFDM Overview." http://rfmw.em.keysight.com/wireless/helpfiles/89600b/webhelp/subsystems/wlan-ofdm/content/ofdm_80211-overview.htm (accessed 10-27-2019).

[16] a. M. N. O. S. Samuel O. Agbo, *Principles of Modern Communication Systems*, 1st ed. Cambridge University Press, 2017.

[17] "Wireless Innovation Forum: What is Software Defined Radio." http://www.wirelessinnovation.org/Introduction_to_SDR (accessed 2019-07-14).

[18] I. R. Sector, "Definitions of Software Defined Radio (SDR) and Cognitive Radio System (CRS) report," ITU-R SM, 2009.

[19] F. T. C. Chen, K. Chang, H. Chao, and J. Chen, "Reconfigurable Software Defined Radio and its Applications," *Tamkang Journal of Science and Engineering*, vol. 13, 1, pp. 29–38, 2010.

[20] K. Bhusal, "Implementation and Performance Analysis of Long Term Evolution Using Software Defined Radio," Electrical Engineering 2017.

[21] E. Grayver, "Disadvantages of SDR," in *Implementing Software Defined Radio*, E. Grayver Ed. New York, NY: Springer New York, 2013, pp. 37-41.

[22] "LimeSDR-Mini v1.2 Hardware Description." https://wiki.myriadrf.org/LimeSDR-Mini_v1.2_hardware_description (accessed 2019-10-11).

[23] "LimeSDR-Mini." https://www.crowdsupply.com/lime-micro/limesdr-mini (accessed 2019-09-16).

[24] "GNU Radio Companion." https://wiki.gnuradio.org/index.php/GNURadioCompanion (accessed 2019-09-16).

[25] *Local and Metropolitan Area Networks, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications* (IEEE Std 802.11-2007). 2007.

[26] B. Bloessl, M. Segata, C. Sommer, and F. Dressler, "An IEEE 802.11a/g/p OFDM Receiver for GNU Radio," presented at the Proceedings of the second workshop on Software radio implementation forum, Hong Kong, China, 2013.

[27] E. Sourour, H. El-Ghoroury, and D. McNeill, "Frequency Offset Estimation and Correction in the IEEE 802.11a WLAN," in *IEEE 60th Vehicular Technology Conference, 2004. VTC2004-Fall. 2004*, 26-29 Sept. 2004 2004, vol. 7, pp. 4923-4927 Vol. 7, doi: 10.1109/VETECF.2004.1405033.

[28] "Frequency Offset Correction." https://openofdm.readthedocs.io/en/latest/freq_offset.html (accessed 11-13-2019).

[29] Y. Huimei, Yingzhuan, Hao, and Wen, "Research on Channel Estimation for OFDM Receiver Based on IEEE 802.11a," in *6th IEEE International Conference on Industrial Informatics*, 13-16 July 2008, pp. 35-39, doi: 10.1109/INDIN.2008.4618061.

[30] M. Fadul, "The Impact of Rayleigh Fading Channel Effects on the RF-DNA Fingerprinting Process," University of Tennessee at Chattanooga, 2018.

[31] M. K. M. Fadul, D. R. Reising, T. D. Loveless, and A. R. Ofoli, "RF-DNA Fingerprint Classification of OFDM Signals Using a Rayleigh Fading Channel Model," in *2019 IEEE Wireless Communications and Networking Conference (WCNC)*, 15-18 April 2019 2019, pp. 1-7, doi: 10.1109/WCNC.2019.8885421.

[32] D. R. Reising, "Exploitation of RF-DNA for Device Classification and Verification Using GRLVQI Processing," AIR FORCE INSTITUTE OF TECHNOLOGY, 2012.

[33] M. Bastiaans, "Discrete Gabor Transform and Discrete Zak Transform," in *IEEE Int'l Conf on Signal and Image Processing Applications*, 1996.

[34] P. K. Pathak, "Sufficiency in Sampling Theory," in *The Annals of Mathematical Statistics*, 1964.

[35] D. R. Reising, M. A. Temple, and M. E. Oxley, "Gabor-based RF-DNA Fingerprinting for Classifying 802.16e WiMAX Mobile Subscribers," in *International Conference on Computing, Networking and Communications (ICNC)*, Jan. 30 2012-Feb. 2 2012 2012, pp. 7-13, doi: 10.1109/ICCNC.2012.6167534.

[36] "LimeSDR_Mini_1v1_Schematic." https://github.com/myriadrf/LimeSDR-Mini/blob/master/hardware/1v1/Project%20Outputs%20for%20LimeSDR_Mini_1v1/LimeSDR_Mini_1v1_schematic_r1.PDF (accessed 11-14-2019).

[37] "Definitions for Local Oscillator." https://www.definitions.net/definition/local+oscillator (accessed 10-29-2019).

[38] A. C. Polak and D. L. Goeckel, "Wireless Device Identification Based on RF Oscillator Imperfections," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2492-2501, 2015, doi: 10.1109/TIFS.2015.2464778.

[39] G. Lee, "Digital Pre-Distortion of Carrier Frequency Offset for Reliable Wi-Fi Enabled IoTs," Department of Convergence Security Engineering, Sungshin University, 2017.

[40] G. Baldini, G. Steri, I. Amerini, and R. Caldelli, "The Identification of Mobile Phones through the Fingerprints of their Built-in Magnetometer: An Analysis of the Portability of the Fingerprints," in *2017 International Carnahan Conference on Security Technology (ICCST)*, 23-26 Oct. 2017 2017, pp. 1-6, doi: 10.1109/CCST.2017.8167855.

[41] "Enabling External Clock on LimeSDR-Mini," 2019-10-11. [Online]. Available: https://discourse.myriadrf.org/t/enabling-external-clock-on-limesdr-mini-1-2/3066.

[42] "U.S. Frequency Allocations." https://www.ntia.doc.gov/files/ntia/publications/2003-allochrt.pdf (accessed 2019-10-12).

[43] L. Chia-Horng, "On the Design of OFDM Signal Detection Algorithms for Hardware Implementation," in *GLOBECOM '03. IEEE Global Telecommunications Conference (IEEE Cat. No.03CH37489)*, 1-5 Dec. 2003 2003, vol. 2, pp. 596-599 Vol.2, doi: 10.1109/GLOCOM.2003.1258308.

[44] P. Fuxjaeger, A. Costantini, D. Valerio, P. Castiglione, T. Zemen, and F. Ricciato, *IEEE 802.11p Transmission Using GNURadio*. 2010.

[45] J. Shi, "Packet Detection," 2017. [Online]. Available: openofdm.readthedocs.io/en/latest/detection

[46] J. Pedro, "Evaluation of IEEE 802.11a/g/p Transceiver for SDR," Electronics and Computer Engineering, 2017.

[47] "Measures of Skewness and Kurtosis." https://www.itl.nist.gov/div898/handbook/eda/section3/eda35b.htm (accessed 11-14-2019).

VITA

Ahmed Ibrahim was born in Khartoum, Sudan, to his parents Mohamed and Sayda. He is the fourth of six children, with two older brothers, one older sister, one younger sister, and one younger brother. He got his Bachelor of Science (B.Sc.) degree in Electrical and Electronics Engineering (Communications Engineering) in 2009 from the University of Khartoum in Khartoum, Sudan. After graduation, he worked in the Nile Center for Technology Research as a Network and Security Engineer (2 years). Then he joined Zain Group LTD as an IP Network engineer (2 years) and Packet core network engineer (3 years). Then he moved to the U.S. to work as a Research Assistant at The University of Tennessee at Chattanooga to work for a Master of Science (MS) degree in Electrical Engineering.