

THE DESIGN, BUILDING, AND TESTING OF A CONSTANT JAMMER
FOR THE BLUETOOTH LOW ENERGY (BLE) WIRELESS
COMMUNICATION PROTOCOL

A Thesis
presented to
the Faculty of California Polytechnic State University,
San Luis Obispo

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in Electrical Engineering

by
Aiku Shintani

June 2020

© 2020

Aiku Shintani

ALL RIGHTS RESERVED

COMMITTEE MEMBERSHIP

TITLE: The Design, Testing, and Analysis of a Constant
Jammer for the Bluetooth Low Energy (BLE)
Wireless Communication Protocol

AUTHOR: Aiku Shintani

DATE SUBMITTED: June 2020

COMMITTEE CHAIR: Vladimir Prodanov, Ph.D.
Associate Professor of Electrical Engineering

COMMITTEE MEMBER: Bruce DeBruhl, Ph.D.
Assistant Professor of Computer Science and
Software Engineering

COMMITTEE MEMBER: Steve Dunton, M.S.
Lecturer of Electrical Engineering

ABSTRACT

The Design, Testing, and Analysis of a Constant Jammer for the Bluetooth Low Energy (BLE) Wireless Communication Protocol

Aiku Shintani

The decreasing cost of web-enabled smart devices utilizing embedded processors, sensors, and wireless communication hardware have created an optimal ecosystem for the Internet of Things (IoT). IEEE802.15.4, IEEE802.11ah, WirelessHART, ZigBee Smart Energy, Bluetooth (BT), and Bluetooth Low Energy (BLE) are amongst the most commonly used wireless standards for IoT systems. Each of these standards has tradeoffs concerning power consumption, range of communication, network formation, security, reliability, and ease of implementation. The most widely used standards for IoT are Bluetooth, BLE, and Zigbee. This paper discusses the vulnerabilities in the implementation of the PHY and link layers of BLE. The link layer defines the scheme for establishing a link between two devices. Scanning devices are able to establish communication with other devices that are sending advertising packets. These advertising packets are sent out in a deterministic fashion. The advertising channels for BLE, specified by the PHY layer, are Channels 37, 38, and 39, at center frequencies 2.402, 2.426, and 2.480 GHz, respectively. This scheme for establishing a connection seems to introduce an unintentional gap in the security of the protocol. Creating and transmitting tones with center frequencies corresponding to those of the advertising channels, a victim BLE device will be unable to establish a connection with another BLE device. Jamming a mesh network of BLE devices relies on this same concept. The proposed jamming system is an inexpensive one which utilizes the following hardware. Three individual synthesizers, a microcontroller (MCU), Wilkinson power combiner, power amplifier, and antenna, integrated on a single PCB, are used to transmit a 3-tone signal. Due to the unprecedented nature of the COVID-19 pandemic, necessary adjustments were made to the jammer system design. In the first modified jamming scheme, a single synthesizer evaluation board, power amplifier, and antenna, are used to transmit jamming tones in the form of a frequency hop. Limitations of the frequency hop approach necessitated a second modified scheme. In this second scheme a synthesizer and two Software Defined Radios (SDR), connected to a personal computer, continuously generate three individual jamming tones. The proposed jammer and the modified ones all classify as constant jammers as the transmission of jamming signals is continuous. Both modified jamming schemes are tested. The results of jamming using the second modified scheme validate the objective of simultaneous jamming of the advertising channels of BLE devices. The success of the modified scheme enables the original goal of creating a relatively inexpensive custom PCB for BLE advertising channel jamming. By exploiting the weakness of the BLE protocol, the hope is to have the governing body for Bluetooth, Bluetooth Special Interest Group (SIG), improve security for the future releases of BLE.

Keywords¹: Bluetooth Low Energy (BLE), Bluetooth (BT), BT 5.0, BT 5.1, BT 5.2, Jammer, Synthesizer, Advertising, Software Defined Radio (SDR), Extended Advertising

¹ Refer to Bluetooth core specification document for “List of Acronyms and Abbreviations” for more [13]

ACKNOWLEDGMENTS

Firstly, I would like to thank my advisor, Dr. Prodanov, for inspiring my interest in not only electrical engineering as a whole, but also for this challenging project. His guidance, knowledge, and level-headedness were instrumental throughout the development of this thesis.

I would also like to thank committee members, Dr. BeDruhl and Steve Dunton, for their valuable guidance and feedback. Their perspectives were integral for the refining of my thesis report.

In addition to the committee members, I would like to express my gratitude to the entire electrical engineering staff and faculty. Special thank you to Professor Bryan J. Mealy for motivating me to find my passion within the field and for inspiring my curiosity with regards to engineering. Thank you to Michael Wilson and Dr. Pilkington whose classes in analog design and signal processing, respectively, provided a strong framework of knowledge on which I was able to implement the various concepts required by this project.

Many thanks go out to my peers and classmates over the years. Special thank you to Lauren Williams for her contributions to my project in the form of a donated synthesizer evaluation board from Analog Devices. Thank you to Jake Wahl and Matthew Carroll for their contributions to my project in the form of lent software define radios.

Another special thank you to Cypress Semiconductor for their contributions to my project in the form of a Bluetooth Low Energy mesh evaluation kit.

Lastly, I would like to thank my parents for their continuous support throughout the years of my academic life. Without them I would not be the person I am today and certainly not in the position that I am in today.

TABLE OF CONTENTS

	Page
LIST OF TABLES	viii
LIST OF FIGURES	ix
CHAPTER	
1. INTRODUCTION	1
1.1 Statement of Problem.....	1
1.2 Jamming – Literature Review	3
1.3 Bluetooth/BLE Background.....	5
1.4 BLE Protocol Stack.....	6
1.5 BLE PHY & Link Layer Specifications	9
1.5.1 Transmitter Characteristics	9
1.5.2 Receiver Characteristics.....	11
1.5.3 Signal-to-Interference Ratio.....	11
2. BLE ADVERTISING.....	13
2.1 Overview.....	13
2.2 BLE Device Roles.....	15
2.3 Advertising Sets.....	17
2.4 Advertising Events.....	17
2.4.1 Advertising Interval	19
2.4.2 Extended advertising events.....	20
2.4.3 Periodic advertising events	21
2.4.4 Low vs. High Duty Cycle Advertising.....	21
2.5 Relating Advertising Schemes and Jamming.....	22
2.5.1 Jamming BLE versions 4.2 and earlier	22
2.5.2 Jamming BLE versions 5.0 and later	24
3. DESIGN OVERVIEW	26
3.1 Preliminary High-level Overview.....	26
3.2 Component Selection	27
3.2.1 Synthesizer Selection	28
3.2.2 Power Combining Circuitry Selection	29
3.2.3 Power Amplifier and Antenna Selection	32
3.2.4 Bluetooth Mesh Selection	32
4. BLE MESH NETWORK DEVELOPMENT	34

4.1	Establishing the Mesh Network	34
4.2	Mesh Network Development	37
5.	FREQUENCY HOPPING JAMMER SYSTEM.....	40
5.1	Overview.....	40
5.2	Signal Characterization - Synthesizer	42
5.3	Frequency Hopping – Synthesizer	45
5.3.1	Frequency Hopping – Approach #1	45
5.3.2	Frequency Hopping – Approach #2	46
5.3.3	Frequency Hopping – Approach #3	50
5.4	Frequency Hopping Jammer System Overview.....	53
5.4.1	Balun Selection	54
5.4.2	Power Amplifier Selection.....	56
5.5	System Requirements.....	57
6.	SYSTEM TESTING & CHARACTERIZATION – FREQUENCY HOPPING JAMMER..	61
6.1	Overview.....	61
6.2	Testing Environment.....	62
6.3	Testing – Frequency Hopping Jammer	63
7.	SYSTEM TESTING & CHARACTERIZATION – CONSTANT JAMMER	67
7.1	Constant Jammer System Overview	67
7.1.1	SDR Selection.....	67
7.2	Signal Characterization – SDR	69
7.3	System Requirements.....	71
7.4	Testing – Constant Jammer.....	73
7.5	Additional Exploration.....	77
8.	CONCLUSION	79
8.1	Reflection.....	79
8.2	Future Works	80
	REFERENCES	82
	APPENDICES	
	A.....	87
	B.....	87

LIST OF TABLES

Table	Page
1. Power Classes of LE PHY [13].....	10
2. Receiver Sensitivity for a Given PHY [13].....	11
3. Interference Performance for LE 1M PHY [13].....	12
4. GAP Compliance Requirements for BLE Device Roles (Adapted from [13])	15
5. Physical Channel Usage for Advertising Event Types (Adapted from [13]).....	19
6. Constant Jammer Success (Signal at Antenna (dBm): 17.26, -1.36, -1.36).....	77
7. Constant Jammer (w/o PA) Success (Signal at Antenna (dBm): -3.74, -1.36, -1.36).....	78

LIST OF FIGURES

Figure	Page
1. Conceptual Depiction of BLE Mesh Network [4].....	2
2. Conceptual Depiction of Network Jamming (Image Adapted from [4]).....	5
3. The Bluetooth Low Energy Protocol Stack [11]	7
4. BLE Channelization with Data (blue) and Advertising (orange) Channels [12].....	8
5. Impact of Distance between Jammer and BLE Receiver [14].....	11
6. BLE Link Layer States	13
7. BLE Device State Diagrams per Role.....	16
8. Multiple Advertising Sets Supported by Link Layer [13].....	17
9. Advertising Events Perturbed in Time via advDelay (Adapted from [13]).....	20
10. Synchronous Advertising Events [13].....	21
11. Advertising Event Example (Adapted from [13]).....	23
12. BLE Co-existence with Wi-fi [18]	24
13. BLE Channelization with Three Jamming Tones Overlaid (adapted from [12])	26
14. Preliminary High-level Block Diagram.....	27
15. EVAL01-HMC1035LP6GE Evaluation Board [19]	29
16. Distributed Implementation of 2-way Wilkinson (Adapted from [20])	29
17. Wilkinson Circuit Decomposition.....	30
18. Lumped Implementation of 2-way Wilkinson Power Combiner (Adapted from [20]).....	31
19. LC Component Sizing Equations [21]	31
20. Schematic of Lumped Element 3-way Wilkinson Power Combiner.....	32
21. CYBT-213043-MESH Kit Contents [22].....	33
22. ModusToolbox IDE.....	34
23. BLE Advertising Channels Configured in ‘wiced_bt_cfg’ File.....	35
24. Provisioning Process [24].....	37
25. ClientControlMesh Application User Interface.....	38
26. Mesh Network Diagram	39
27. Set up for EVAL01-HMC1035LP6G Evaluation Board (Adapted from [27])	40
28. Hittite Clock and Timing Main GUI	41
29. Synthesizer Block Diagram.....	41
30. Output Waveform, Performance vs. Power Priority for LVPECL/LVDS [28].....	42
31. Synthesized LVDS Signals	43

32. Synthesized LVPECL Signals.....	44
33. Frequency Hop Configuration.....	46
34. Annotated Hittite User Interface	47
35. Eval01-HMC1035PG Board Schematic Snip [30].....	48
36. Oscilloscope Capture of VCO Tune Pin – Approach #2.....	49
37. Oscilloscope Capture of VCO Tune Pin to Measure Delay – Approach #2	50
38. VCO Frequency Scan Configuration.....	51
39. R/W Regs History Option of Hittite User Interface	51
40. Oscilloscope Capture of VCO Tune Pin – Approach #3.....	52
41. Modified Block Diagram – Frequency Hopping BLE Jammer (Part 1).....	54
42. ADC-WB-BB Signal Conversion Evaluation Board [31].....	54
43. Balun Board Image and Oscilloscope Capture [32]	55
44. Modified Block Diagram – Frequency Hopping BLE Jammer (Part 1.1).....	56
45. EVAL-CN0417-EBZ 2.4GHz Power Amplifier Evaluation Board [34].....	57
46. Finding Max Distance of Jamming Signal Reception (Synthesizer + PA + Delta6B).....	59
47. RF Range Calculator by Silicon Labs (Synthesizer + PA + Delta6B)	59
48. Synthesizer Jammer, Top View (Approach #1)	62
49. Home Testing Environment Set up	63
50. Measured Signal Strength (Synthesizer Jammer) at Receiver/SA vs. Distance.....	64
51. Test Set up Synthesizer jammer	65
52. ADALM-Pluto SDR [36].....	68
53. ADALM-Pluto SDR I/O [37].....	68
54. Modified Block Diagram – Synthesizer and Two SDRs.....	69
55. Spectrum Analyzer Measuring 2.426GHz Signal Output of ADALM SDR.....	70
56. Finding Max Distance of Jamming Signal Reception (SDR + JCG401)	72
57. RF Range Calculator by Silicon Labs (SDR + JCG401).....	72
58. Constant Jammer, Side View (Approach #2)	73
59. Measured Signal Strength (SDR) at Receiver/SA vs. Distance	74
60. Constant Jammer Located at 0.5m Mark.....	75
61. Strength of Signal Generated by Constant Jammer at Distance of 0.5m	75
62. Test Set up Constant Jammer (Iteration #1).....	76
63. Strength of Signal Generated by Constant Jammer (w/o the PA) at Distance of 0.5m	77

Chapter 1

INTRODUCTION

This thesis report describes the design, testing, and analysis of a Bluetooth Low Energy (BLE) jamming device that will jam either a select number or all BLE primary advertising channels. This report discusses the BLE protocol and its weakness, the general concept for exploiting this weakness, and the final system design. The paper details the process for creating the jamming device, from initial plan to final construction. The various challenges encountered throughout the process are described in detail, in addition to their optimal solutions. The proposed jammer system is an inexpensive PCB equipped with three individual synthesizers for generation of tones centered at the primary advertising channel frequencies (Ch. 37, 38, & 39). A BLE mesh network is created to test the effectiveness of the jamming signal(s). Due to COVID-19 related supply chain issues and lack of laboratory access on campus, the proposed jammer system design was modified. The first experimental jammer system consists of a single synthesizer for generation of the three tones, in the form of a frequency hop. The second experimental jammer system consists of a single synthesizer & two software defined radios (SDR) for continuous generation of tones centered at the primary advertising channel frequencies. The same BLE mesh network is used to test the effectiveness of the two experimental jammer systems. The report will go over the design and testing of each subsystem and the outcome of the BLE jamming attempt(s).

1.1 Statement of Problem

Since the conception of Internet of Things (IoT) in the late 20th century, significant advancements have been made in sensor/actuator technologies, enabling applications such as smart homes, phones, and cars [1]. These advances include sophisticated power management schemes, improvements to manufacturing processes of sensors/actuators, and low-power communication capabilities. These sensors/actuators are often referred to as nodes, of an IoT network. These nodes share data as well as computing resources in order to extract patterns and trends, and ultimately use the acquired information to optimize various aspects of modern society [2]. An IoT network is essentially a multifaceted, spatially distributed control system which intelligently implements tasks in an efficient and reliable manner.

When the term IoT was first named, most viable electronic systems used wired sensor and communication nodes. Since the emergence of robust wireless communication standards such as classic Bluetooth (BT) in 1999, IoT networks have deviated away from wired solutions for communication. The sensor nodes in modern IoT networks are often System on Chip (SoC), which integrate inexpensive, low powered miniature components, radios, and sensors [3]. These SoCs communicate within the IoT network via radio frequency (RF) wireless signals, which propagate through free space as opposed to a wire. Wireless signals are more vulnerable to attacks as a malicious user can intercept and/or jam it while it is traveling through a medium; wireless systems are often optimized to be resilient to non-malicious interference and noise.

Figure 1 below depicts a generic mesh network [4]. Each of the devices in the mesh network are referred to as nodes. End point nodes are depicted as ‘N’ while relay nodes, which turn a single long hop into two shorter hops, are depicted as ‘RN’. The data transferred between wireless nodes can vary in its volume, periodicity, variety, transfer speed, and requirements for processing [5]. The interception and analysis of this data could be potentially detrimental to an individual/organization if specific activities are monitored and recorded; information such as whether or not a TV is in use could allow a malicious user to determine if the home is occupied or not. As another example, a user with ill-intent could take down an entire BLE mesh network (wearables, equipment, etc..) operating in a hospital setting and endanger the lives of patients. With this in mind, considerations of security are a high priority when developing wireless communication standards. Those pertaining to BLE are discussed in a later section.

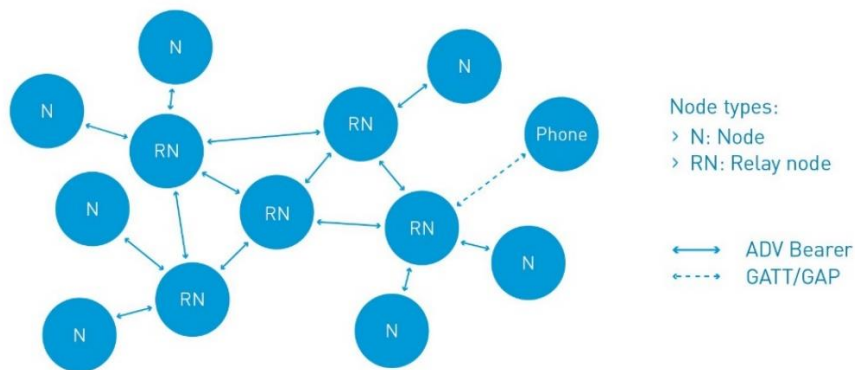


Figure 1: Conceptual Depiction of BLE Mesh Network [4]

BLE utilizes the 2.4GHz unlicensed Industrial, Scientific and Medical (ISM) frequency bandwidth and must minimize interference with other protocols (i.e. Wi-Fi, ZigBee) operating in the same frequency range. This

concept of wireless coexistence is critical when developing and deploying standards such as BLE. To achieve frequency diversity and minimize interference from other wireless signals, BLE's protocol stack allocates three non-equally spaced channels for establishing connections and transferring advertising data [6]. These three channels, 37, 38, & 39, are referred to as the primary advertising channels with fixed frequencies of 2.402GHz, 2.426GHz, and 2.480GHz, respectively. Advertising packets are sent out in a deterministic fashion on these primary advertising channels. The protocol's scheme for establishing connections and transferring advertising data, using the advertising channels, is believed to be a cause of a significant risk in the security of the BLE protocol. The goal of this thesis is to exploit the weakness of the BLE protocol with the hope that the governing body for Bluetooth, Bluetooth Special Interest Group (SIG), improves security for the future releases of BLE.

1.2 Jamming – Literature Review

Pelechrinis et al defines a jammer as an entity who is purposefully attempting to inhibit wireless communication by interfering with the physical transmission and reception of signals [7]. Wireless systems are vulnerable relative to wired systems as several wireless protocols share the medium for communication and is easily accessible. Pelechrinis et al, researchers who conducted an extensive research survey of wireless jammers, state: “wireless systems have been designed only to be resilient to non-malicious interference and noise” [7]. Four criteria are used to characterize the performance/efficiency of jamming: energy efficiency, probability of detection, level of denial of service (DoS), and strength against PHY layer techniques such as Frequency-Hopping Spread Spectrum (FHSS) used in Bluetooth Low Energy [7]. The jamming scenario of interest ultimately dictates the most suitable criteria for use. In most cases, jammers attempt to be effective in as many of the aforementioned criteria.

Jamming schemes that exploit PHY and MAC layer vulnerabilities are referred to as “brute-force” techniques [7]. There are four basic models of brute-force jamming: a constant jammer, a deceptive jammer, a random jammer, and a reactive jammer. A constant jammer continually emits signals on the medium; this jammer attempts to pose interference on any node's (device) transceiver in order to corrupt its packets at the receiver and makes a node's transceiver sense the channel is busy, preventing it from accessing the channel. A deceptive jammer also continuously transmits signals but unlike a constant jammer, the transmitted signals

are not random. This type of jammer injects regular packets on the channel, which makes a user of the channel believe there is legitimate traffic on the channel [7]. Deceptive jamming is harder to detect, relative to constant jamming, using network monitoring tools since the tools will perceive the channel to have legitimate traffic. A random jammer is more power efficient than the previous two as jamming is employed as a periodic cycle of jamming and sleeping. A reactive jammer is the smartest of these four basic models as it constantly senses the channel and transmits a jamming signal once it senses a packet transmission.

Jamming schemes that exploit vulnerabilities at the higher layers of the network stack are often stealthier, entailing lower power consumption and lower probability of detection; these jamming schemes are referred to as intelligent jammers. Intelligent jammers have three goals in mind: maximizing jamming gain, targeted jamming, and reduced probability of jamming. Assessment of these three goals consists of relative comparisons to constant jammer techniques. Pelechrinis et al defines jamming gain as the “inverse ratio of the amount of power used to achieve a desired effect with the jammer under consideration to the amount of power that is used to achieve the same effect with the constant jammer” [7]. Therefore, a higher jamming gain is ideal. Targeted jamming entails the jammer pays attention to which nodes of the network are being jammed. Pelechrinis et al detail how to achieve a reduced probability of detection in the following statement: “one can force the victim network to believe that the degradation in network performance is due to congestion or poor link conditions and not due to the presence of a jammer” [7]. Intelligent jammers often spend time sensing the wireless channel.

The jammer proposed in this paper is a constant one. In the first testing approach, the experimental jammer system continuously transmits signals, in the form of a frequency hop, on the three primary advertising channels. In the second approach, the experimental jammer system continuously transmits tones on the three primary advertising channels via a synthesizer and two SDRs. The signals that are transmitted are not in the form of a standard BLE packet, which entails the jammer is not a deceptive one. However, depending on the network monitoring tool used the individual tones received by the victim device may be interpreted as advertising. Nonetheless, the proposed jammer is a constant one which emits RF signals that do not follow the rules of the BLE MAC layer protocol. The figure below provides a depiction of a jammer unleashed on a BLE mesh network, resulting in the loss of communication between the network nodes.

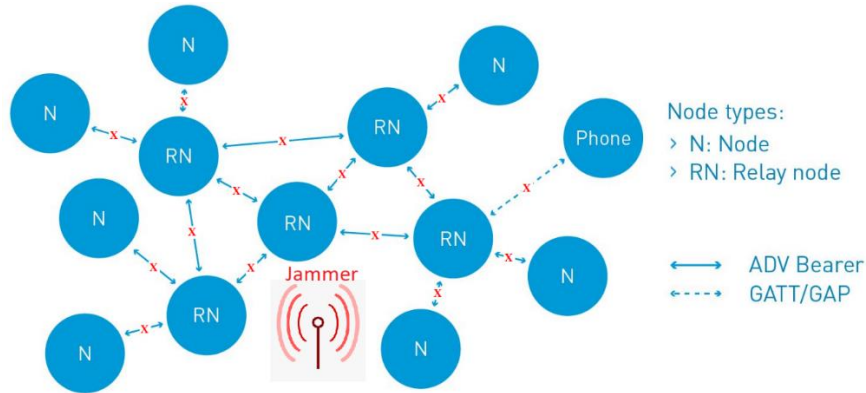


Figure 2: Conceptual Depiction of Network Jamming (Image Adapted from [4])

1.3 Bluetooth/BLE Background

Bluetooth's defining organization, Bluetooth Special Interest Group (SIG), formed with five companies in 1998 [8]. The following year Bluetooth SIG formally released the Bluetooth 1.0 standard. The motivation for creating this standard came from the need to unify elements of the computer and telecommunications industries, for short range applications. SIG estimates more than 8.2 billion Bluetooth devices in use today and 92% of consumers globally recognize the brand; these statistics are indicative of Bluetooth technology's continuous improvements to functionality and utility since its conception [9]. Today, Bluetooth's prominence is evident in multiple consumer markets including audio and entertainment, phones/tablets, PCs, and automotive. SIG has released multiple notable newer versions of Bluetooth since release of Bluetooth 1.0, including: BT 1.2, BT 2.0, BT 3.0, BT 4.0, BT 4.2, BT 5.0, and BT 5.1. These versions of Bluetooth have considerable implications on the security of the protocol and how the emergence of IoT has fostered the development of the Low Energy standard.

BT 1.2 formally introduced Adaptive Frequency Hopping (AFH), improving the efficiency of signal transmission by avoiding the use of crowded channels in the hopping sequence [8]. The maximum data transfer rate and power consumption improved with BT 2.0's introduction of Enhanced Data Rate (EDR). The maximum data transfer rate improved significantly, once again, through BT 3.0's high speed (HS) channel. The high-speed channel allows the enabling of high data rate traffic on a co-located IEEE802.11 (Wi-Fi link). With this Wi-Fi link connection, Bluetooth devices were now ready for wireless video streaming, not just audio transfers as in past versions.

SIG adopted BLE in 2010 as part of the BT 4.0 specification. BLE is not backward compatible to previous versions of classic Bluetooth (BT 1.0, BT 2.0, BT 3.0, etc..). BLE's PHY and link layer specifications differ drastically from classic Bluetooth. This new operating mode was designed to offer higher efficiency connections to wireless devices that do not require a substantial amount of power. BT 4.0 introduced the beacon, a signal (identifier) sent out by a broadcasting device via a one-way transmitter, for electronic devices in its proximity to perform user-defined actions; the beacon can be used to track users, potentially against their will. BT 4.2 added a Low Energy Secure Connection, improving the privacy of all connected devices. This version also improved upon the vulnerability of BT 4.0 by making devices un-trackable unless the user grants permission.

The data transfer speeds and communication range for BLE improved in the 2016 with the adoption of the BT 5.0 specification. BT 5.0 also introduced extended advertising events, which are essentially advertising events occurring on the BLE data channels. Data channels utilized for extended advertising events are referred to as secondary advertising channels. Notably, in 2017 SIG added Bluetooth mesh networking capabilities for the deployment of large-scale device networks, targeting IoT applications [8]. In 2019, SIG adopted the BT 5.1 specification, which integrated a "randomized advertising channel indexing," enabling Bluetooth devices to broadcast that they're available for pairing/connecting. Since the release of BT 1.0, SIG has significantly improved upon the functionality, usability, and security of the Bluetooth standard. The BT standards equipped with BLE (versions 4.0 and later) are dominant for IoT low-power applications while older versions of Bluetooth (i.e. versions 1.0, 2.0, 3.0) still service other applications such as file exchanges, audio, etc... The improvements made to the security of the protocol are indicative of the continuous prevalence of threats to Bluetooth networks. Pelechrinis et al describe this ceaseless interaction of adversaries and network administrators as, "a fascinating arms-race..." [7].

1.4 BLE Protocol Stack

In order to understand how a BLE mesh manages the provisioning of devices into its network, an explanation of the BLE protocol stack is necessary. Provisioning is the process of adding devices to become nodes of a mesh network. Provisioning is explained further in the "Establishing the Mesh Network" section of Chapter 4. An understanding of the protocol stack is essential for an effective jamming scheme to be conceived and

implemented. Bluetooth's protocol stack defines a set of layered programs, with each layer in the stack communicating with the one above and below it.

The stack, referred to as the SoftDevice, consists of two parts, the Bluetooth host and the Bluetooth controller; these two are interfaced through the Host Controller Interface (HCI) which are a set of services and events [10]. Figure 3 below depicts BLE's protocol stack [11]. The controller consists of the PHY, link, and intermediate layers. The PHY layer consists of baseband and radio specifications, which defines the frequency bands, frequency hopping scheme, modulation technique for transmission, power specifications, packet frame format, timing and power control, and detection of Bluetooth devices. The link layer controller, which interfaces with the PHY layer, enables the transmission of data bits and the selection of a physical channel. The link layer directly interfaces to the PHY layer and is responsible for creating/maintaining connections (advertising, scanning, and initiating). The intermediate layer is essentially a controller that implements HCI services. The layers above the HCI are considered the upper layers and are usually implemented in software. The Bluetooth host is generally integrated with the system software or host operating system. An example of this would be integrating the Bluetooth host with the operating system of a smartphone, the host device.

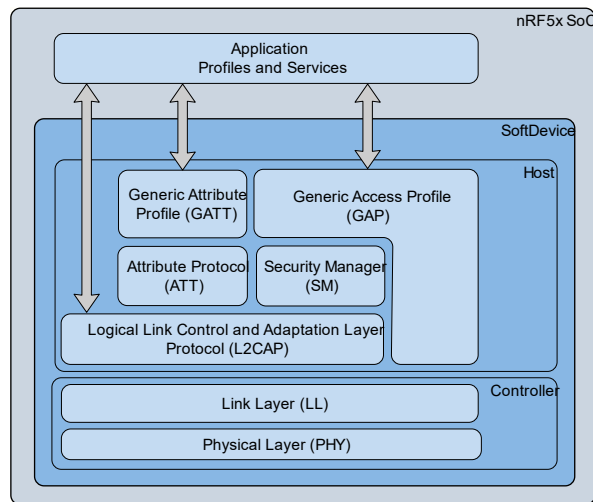


Figure 3: The Bluetooth Low Energy Protocol Stack [11]

Bluetooth Low Energy, a.k.a. Smart Bluetooth, uses a similar protocol stack as classic Bluetooth. Differences between the protocol stacks begin above the L2CAP layer; BLE reuses the PHY and link layers of classic Bluetooth [2]. The BLE spectrum, defined by the PHY layer, is split into 40 channels, starting with Channel

0 and ending with Channel 39. The center frequencies of the channels are separated by 2MHz spacings. In comparison, classic Bluetooth has 39 more usable channels (total of 79 channels), each spaced by 1MHz. Three of the BLE channels, 37, 38, and 39, are referred to as primary advertising channels that are used for establishing a connection and transmitting advertising-related data (advertising packets, scanning packets, & initiating packets) between Bluetooth devices; these channels are not equally spaced in the BLE spectrum as to achieve frequency diversity and also to minimize interference from other wireless signals such as Wi-Fi, classic Bluetooth, microwave, etc. The other 37 channels are referred to as data channels and as secondary advertising channels if used for secondary advertising. BT 5.0 introduced the use of the data channels as secondary advertising channels. Figure 4 below depicts BLE's channelization [12].

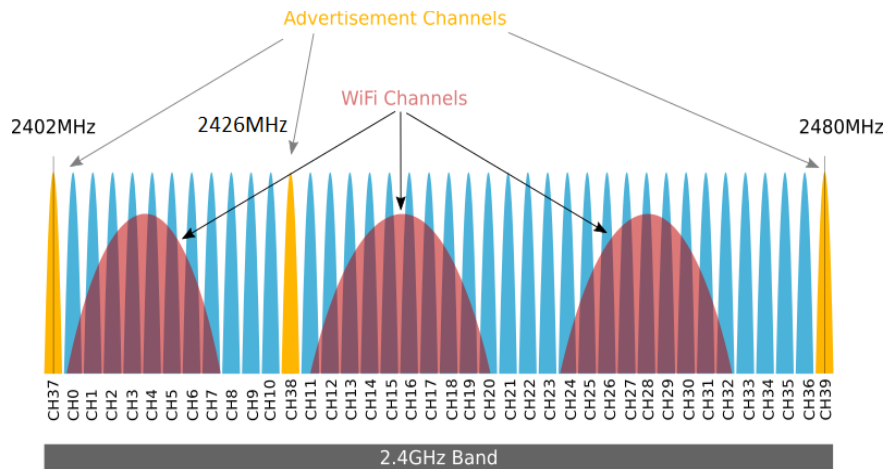


Figure 4: BLE Channelization with Data (blue) and Advertising (orange) Channels [12]

The Wi-fi interference in the above figure depict signal strengths for channels 1, 6, and 11, which are the most popular channels for Wi-fi [12]. The BLE advertising channels are strategically located at frequencies that don't overlap with Wi-fi signal interference; if communication on the advertising channels is blocked, devices in a BLE network cannot communicate with one another. The data channels that do not overlap with the depicted Wi-fi channels are considered to be the most viable for use as secondary advertising channels. There are ten data channels (Ch. 8, 9, 10, 21, 22, 23, 33, 34, 35, & 36) that are depicted to have minimal Wi-fi interference.

1.5 BLE PHY & Link Layer Specifications

The PHY layer defines the physical channels, advertising and data, of the BLE protocol. An understanding of the PHY layer specifications is essential for jamming the advertising channels of any BLE device. Specifications for transmitter and receiver power are discussed in this section. Additionally, specifications for signal to interference (in-band and out-of-band) are discussed as they directly tie into the feasibility of jamming a BLE device. These specifications are detailed in SIG's core specification documents, which are revised and released for each version of Bluetooth.

Bluetooth 5.1 transceivers can support two modulation schemes, 1 megasymbol per second (Msym/s) and 2 Msym/s, with the former as the mandatory and the latter as the optional one [13]. Both modes implement Gaussian Frequency Shift Keying (GFSK) modulation, referred to as Basic Rate (BR), which SIG describes as a shaped, binary FM modulation. Bluetooth 5.1 specifies three PHYs; two PHYs, LE 1M and LE Coded (supported by the 1 Msym/s modulation scheme), and a single PHY, LE 2M (supported by the 2 Msym/s modulation scheme). When 1 symbol represents 1 bit, the LE 1M PHY is in use. The LE Coded PHY can be configured in two modes, $S = 2$ and $S = 8$. The value assigned to 'S' is indicative of how many symbols are used to represent each bit. With a higher value of 'S', the data rate or the bits/s rate decreases as each bit consists of more information. The LE coded PHY allows for Forward Error Correction (FEC) while LE 1M and 2M do not; the latter pair are considered uncoded PHYs. An advantage of the lower data rate and FEC is the increase in range for communication. The required signal-to-noise ratio (SNR) is less for the longer range PHYs, meaning the received signal can be lower in magnitude; the receiver sensitivities of the longer range PHYs are lower. SIG specs the range multipliers as 1, 2, 4, and 0.8, for the LE 1M, LE Coded ($S=2$), LE Coded ($S=8$), and LE 2M PHYs, respectively.

1.5.1 Transmitter Characteristics

The requirements for a BLE transmitter are stated in the PHY layer specifications of the SIG core specification for Bluetooth 5.1. Their document specifies that the output power level of a transmitter must be within the range of 0.01mW (-20dBm) to 100mW (+20dBm). BLE devices enabled with BT 4.0, BT 4.1, and BT 4.2, have a maximum output power of 10mW, which is ten times less (in linear units) in magnitude relative to its successors. Notably, the modulation mode, BR or EDR, may affect the max transmit power

level permitted; there are use cases where a BLE-enabled device communicates with non BLE-enabled devices operating via EDR, which uses phase shift keying to enable higher data rates. Another consideration for BLE transmit power level is for use cases where devices are in close proximity to one another and the receiver becomes saturated and results in link failure. BLE devices can be classified into four power classes as shown in Table 1 below [13]. This table provides a baseline for the required transmitter power for the BLE jammer.

Table 1: Power Classes of LE PHY [13]

Power Class	Maximum Output Power (P_{max})	Minimum Output Power ¹
1	100 mW (+20 dBm)	10 mW (+10 dBm)
1.5	10 mW (+10 dBm)	0.01 mW (-20 dBm)
2	2.5 mW (+4 dBm)	0.01 mW (-20 dBm)
3	1 mW (0 dBm)	0.01 mW (-20 dBm)

The -20dBm specification is considered the minimum jamming signal power for the proposed jammer system. The jammer is less detectable if its signal transmission, both in power and data content, mimics that of the BLE protocol. Brauer et al proposes a BLE jammer with a transmit signal strength of 4dBm in their paper [14]. Their set up consists of two BLE devices, transmitter and receiver, and the BLE jammer all on a line elevated 1m from the ground, in an outdoor setting. They record the Advertising Success Rate (ASR) of their victim BLE devices with a fixed distance, of 3.7m, between the BLE devices. The ASR is a ratio of the number of received advertising events over the total number of transmitted advertising events. The distance between the jammer and the BLE receiver is varied from 0.76m to 10m. Their studies found that at a distance of 76cm, the ASR is zero, meaning the advertising event is interfered with completely due to the jamming signal(s). Increasing the distance between the receiver and the jammer results in an increase in the ASR as shown in the figure below [14].

Notably, Brauer et al use a jamming signal strength that is within the LE PHY power class's upper and lower bounds described previously. With this signal strength Brauer et al were successful in jamming the BLE primary advertisement channels. For this reason, a similar BLE transmit signal strength will be utilized for the proposed BLE jammer, one that is within the bounds of the LE PHY transmitter power defined by Bluetooth SIG. The proposed BLE jammer will be tested in an indoor environment as opposed to an outdoor one such as Brauer et al.

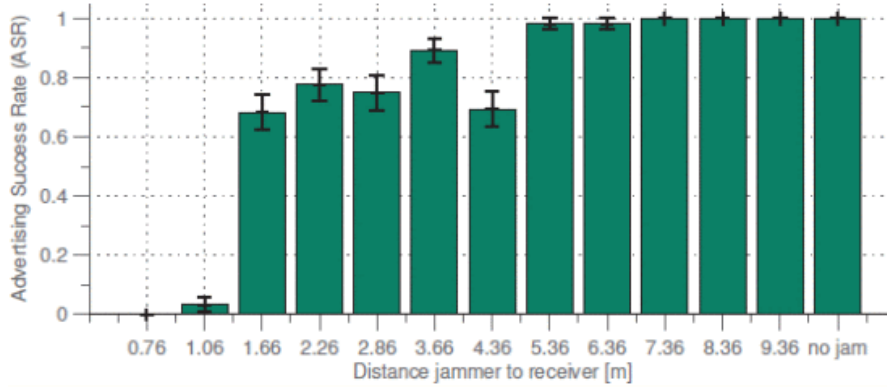


Figure 5: Impact of Distance between Jammer and BLE Receiver [14]

1.5.2 Receiver Characteristics

The requirements for a BLE receiver are also stated in the PHY layer specifications of the SIG core specification for Bluetooth 5.1. The documentation states a reference sensitivity level of -70 dBm for LE Uncoded PHYs, -75dBm for LE Coded PHY with S=2 coding, and -82dBm for LE Coded PHY with S=8 coding. The receiver sensitivity specs are depicted below in Table 2 [13]. The Bit Error Rate (BER) metric characterizes performance corresponding to the packet error rate and is a measure of the sensitivity of the receiver. Bluetooth is a packet-based protocol, where a packet is the data exchanged between devices. In order for a receiver to ‘pass’ the BLE specifications, it must operate fluidly at the maximum usable input level of -20 dBm or greater; -20 dBm is the minimum output power allowed for BLE devices of power class 1, 2, and 3. In order for a device to be considered SIG certified, the BER must be less than or equal to 0.1% at this input power level.

Table 2: Receiver Sensitivity for a Given PHY [13]

PHY	Sensitivity (dBm)
LE Uncoded PHYs	≤ -70
LE Coded PHY with S=2 coding	≤ -75
LE Coded PHY with S=8 coding	≤ -82

1.5.3 Signal-to-Interference Ratio

Bluetooth SIG specifies signal-to-interference ratios (in dB) for co-channel, adjacent, image, and adjacent to in-band image interferences for each of the three PHYs, to achieve the minimum required BER of 0.1% or

less [13]. These specifications are important as BLE receivers must be able to decode wireless signals in the ISM frequency band, which is used by other protocols such as Wi-Fi and ZigBee. Table 3 below summarizes interference performance for the LE 1M PHY; the tables for the other LE PHYs are similar and are included in the core specification document [13]. As depicted in the table, the signal strength, relative to the interference, must be greater when the interference’s central frequency is closer to that of the channel. The jamming signals of the BLE jammer can be classified as co-channel interference since their center frequencies are identical to that of the advertisement channels of interest. A co-channel interference ratio of 21dB is equivalent to a desired signal having approximately 126 times the power of the interference.

Table 3: Interference Performance for LE 1M PHY [13]

Frequency of Interference	Ratio
Co-Channel interference, $C/I_{\text{co-channel}}$	21 dB
Adjacent (1 MHz) interference ¹ , $C/I_{1 \text{ MHz}}$	15 dB
Adjacent (2 MHz) interference ¹ , $C/I_{2 \text{ MHz}}$	-17 dB
Adjacent (≥ 3 MHz) interference ¹ , $C/I_{\geq 3 \text{ MHz}}$	-27 dB
Image frequency interference ^{1 2 3} , C/I_{image}	-9 dB
Adjacent (1 MHz) interference to in-band image frequency ¹ , $C/I_{\text{image}\pm 1\text{MHz}}$	-15 dB

In addition, SIG specifies metrics for out-of-band blocking of interference signals outside the Bluetooth band of 2.4-2.4835GHz. The receiver must be able to suppress out-of-band signals of a specified interfering signal power level in order to achieve a BER of 0.1% or less. The out-of-band interference signal power must be -30dBm or less for low frequencies (30MHz – 2GHz) and high frequencies (3000MHz – 12.75GHz). For frequencies near the Bluetooth band, 2003MHz – 2399MHz & 2484MHz – 2997MHz, the interference signal power level must be -35dBm or less. The proposed BLE jammer will transmit tones corresponding to the three primary advertising channels. In practice it is impossible to synthesize a perfect tone; the synthesized tones will have spectral content classified as out-of-band interference; therefore, these signals will consist of both out-of-band and co-channel interferences. However, the synthesized jamming signals will have low enough phase noise so nearly all the transmitted power will hit the targeted channel. In addition, harmonic content of the synthesized signals will not be radiated by the transmitter antenna of the jammer.

Chapter 2

BLE ADVERTISING

2.1 Overview

The standby state is the default of the link layer. In this state, the link layer is unable to send or receive packets. There are four additional link layer states: advertising, scanning, initiator, and connection. The five BLE states are depicted below in the figure. A BLE device may transition to any of the four other states when in the standby state. Once a BLE device is in the scanning state it may only transition to the standby state. When a BLE device is in the initiating and advertising states, the link layer can transition to the standby or connection states. If the BLE device is an initiator or advertiser and does not establish a connection with another device its link layer will return to the standby state. In the connected state, a BLE device may transition back to either the advertising, standby, or initiating state.

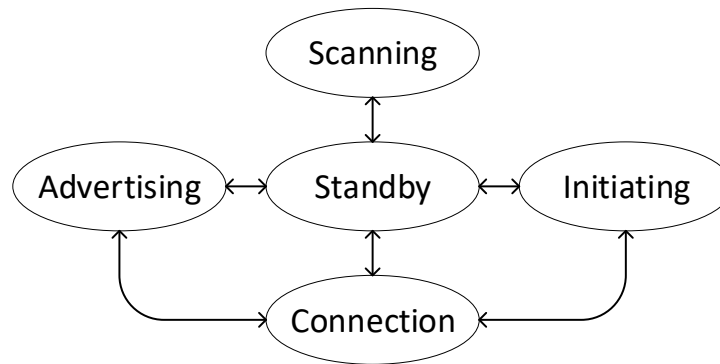


Figure 6: BLE Link Layer States

This section of the report will focus on describing the advertising state, where advertising packets, a.k.a. protocol data units (PDU), are sent in advertising events, periodic advertising events, or both [13]. SIG defines each advertising event as a composition of one or more advertising PDUs that are sent on the used primary advertising channel indices. Advertising events may close early after the reception of a connect indication packet (CONNECT_IND) from an initiator or after the transmission of a scan response (SCAN_RSP) packet sent by the advertiser. If neither of these packets are received, the advertising event closes following the transmission of the last advertising packet.

The used primary channel indices are user defined and can consist of any combination in any order of channels 37, 38, and 39 (i.e. Ch. 37 & Ch.39). Any of the 37 data channels are configurable for advertising

when specified by the user and are referred to as secondary advertising channels. An advertising event that utilizes the secondary advertising channels is referred to as an extended advertising event; an extended advertising event begins at the start of an advertising event and ends following the transmission of the last PDU in that advertising event plus subordinate sets [13]. SIG developer Jim Katsandres confirms this in a SIG article where he states: “Whether using legacy advertising PDUs or the new extended advertising PDUs, these events begin on the primary channel” [15]. Notably, the use of the data channels as secondary advertising channels was not introduced until the release of BT 5.0, so BT version 4.2 and earlier devices will not be able to discover extended advertisements. Typically, a PDU is sent on all the used advertising indices in each advertising event.

Advertising events, total of seven types, are categorized as being connectable or unconnectable. Connectable events consist of connectable and scannable undirected, connectable undirected, and connectable directed events while unconnectable events consist of non-connectable and non-scannable undirected, non-connectable and non-scannable directed, scannable undirected, and scannable directed. Directed vs. undirected specifies whether or not an advertising packet is sent to a specific device. For directed advertisements, the advertising packet contains the scanner or initiator device’s device address in the packet. Scannable vs. non-scannable specifies whether or not the advertiser itself is able to scan for any requests from scanners or initiators. The scanner or initiator device responds to the device transmitting scannable advertisements with either a connect or scan request.

The seven advertising event types are displayed in the table below [13]. The table provides additional information regarding the specific compliance requirements of BLE devices utilizing the various advertising event types. The Generic Access Profile (GAP) defines the four specific roles of BLE devices: broadcaster, observer, peripheral, and central. These roles are discussed in the next section. SIG defines the following status symbols (‘M’, ‘O’, ‘C’, ‘E’) for understanding the table [13]. ‘M’ stands for mandatory support while ‘O’ stands for optional support. ‘C’ stands for conditional support (used for capabilities in a case where a certain or other capability is supported) while ‘E’ stands for excluded within the role. These compliance requirements govern whether or not a device with a defined role (one of the four GAP roles) is able to transmit a specific type of advertising event.

Table 4: GAP Compliance Requirements for BLE Device Roles (Adapted from [13])

	GAP Roles When Operating Over an LE Physical Transport			
	Broadcaster	Observer	Peripheral	Central
<u>Advertising event types:</u>				
• Connectable and scannable undirected event	E	E	M	E
• Connectable undirected event	E	E	O	E
• Connectable directed event	E	E	O	E
• Non-connectable and non-scannable undirected event	M	E	O	E
• Non-connectable and non-scannable directed event	O	E	O	E
• Scannable undirected event	O	E	O	E
• Scannable directed event	O	E	O	E

To clarify on why there is a non-connectable mode, a BLE advertising event does not always require the establishment of a connection between two devices. There are a wide range of applications that utilize the non-connectable advertising event. In many use cases of BLE meshes, the content of the advertising packet contains the entirety of the information the scanner needs. As an example, an indoor positioning system can utilize three BLE devices that send beacons out to find the accurate position of a device such as a smartphone; the three BLE devices used for positioning do not have to connect to each other or the smartphone device in this example. The information contained in the beacon signals themselves are sufficient for completing the task.

2.2 BLE Device Roles

Now that advertising event types and the BLE link layer states are described, the roles of devices in a BLE mesh network are explained. When designing with BLE, power consumption must be optimized for better utilization of battery capacity. The various advertising event types allow network designers to reduce the peak and average power in IoT applications based on BLE; the advertising event types provide design flexibility for BLE networks which ultimately results in the end-customers having to replace their batteries less frequently. SIG defines the following role pairs for BLE devices: advertiser/scanner (initiator) and broadcaster/observer [13]. This section details both of these role pairs and their utility in the BLE network.

In general, a BLE node in the peripheral role (typically an advertiser) will advertise and connect to a central device. The peripheral role is mainly for devices that support a single connection. A BLE node in the central role (typically a scanner/initiator) will scan for connection requests, sent by the peripheral node, and connect to the peripheral. The central role supports multiple connections and is the initiator for all connections with devices in the peripheral role. An example of a peripheral and central device would be a fitness monitor and smart phone, respectively. In this example the fitness monitor would be advertising a connectable and scannable undirected event in order to establish a connection with the smart phone.

In addition to peripherals and central devices, BLE mesh networks can also have broadcaster and observer nodes. In both of these roles, a BLE node cannot establish a connection with other devices. A broadcaster merely advertises information, making it optimal for transmitter only applications; as an example, a BLE sensor node may serve the role of a broadcaster. The role of this BLE sensor node is to transmit sensor data to a specific device (directed) or a device in its vicinity (un-directed) and does not have the ability to establish a connection. An observer merely listens for broadcasted packets, making it optimal for receiver only applications. Of the four BLE device types described, the observer is the only one that never enters the advertising state. The figure below provides a summary of the 4 BLE roles.

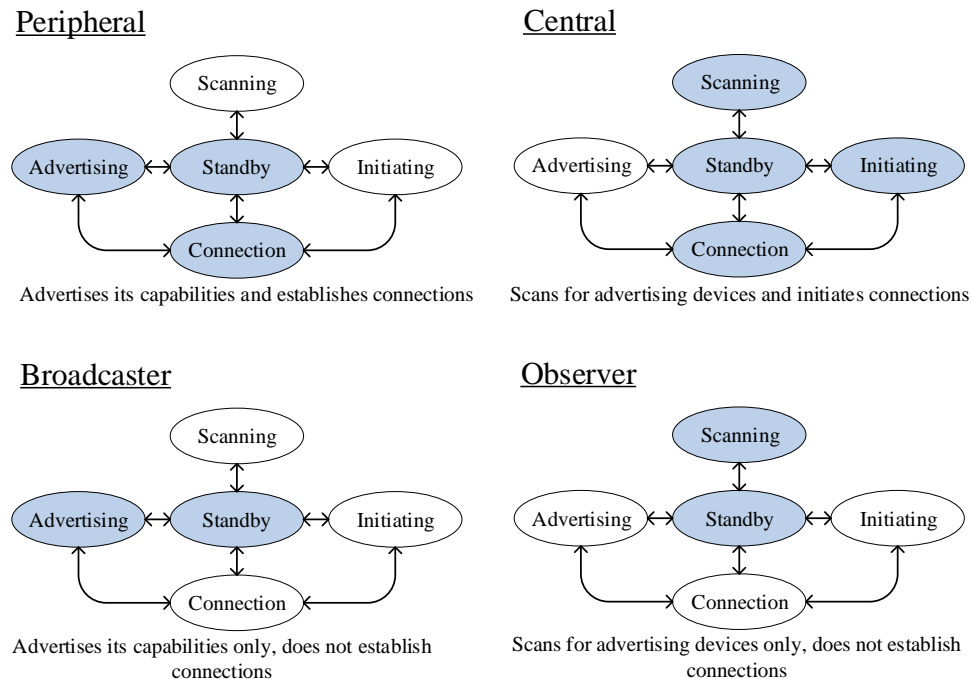


Figure 7: BLE Device State Diagrams per Role

2.3 Advertising Sets

In order to understand how BLE's advertising events are implemented, advertising sets are defined. An advertising set consists of interleaved advertising events. The link layer is capable of supporting multiple advertising sets; each set can have different advertising parameters including advertising PDU type, advertising event, and PHY. Figure 8 below depicts an example program flow where multiple advertising sets are supported by the link layer [13]. Within each advertising set, a user-defined number of advertising events and/or periodic advertising events may take place.

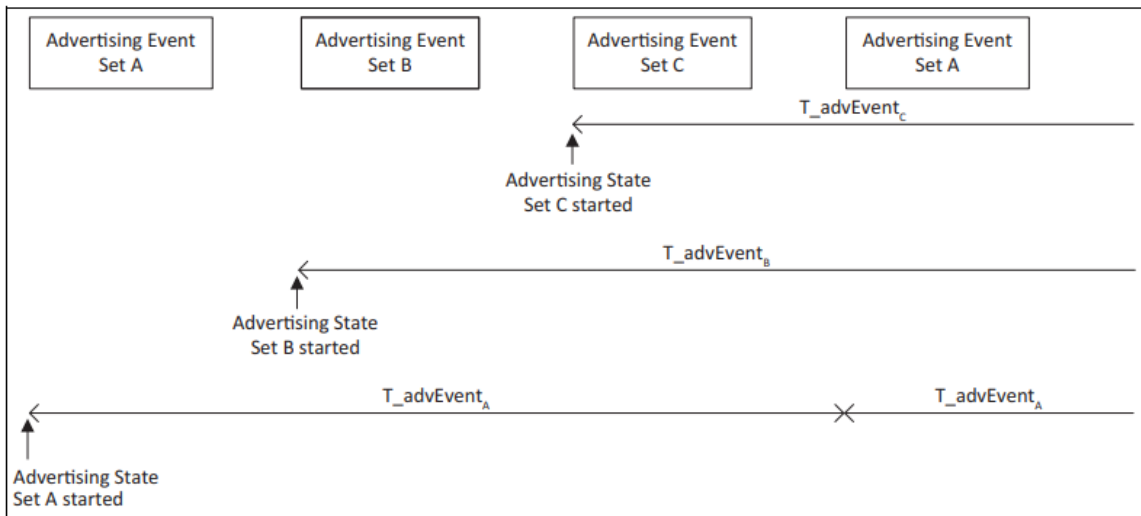


Figure 8: Multiple Advertising Sets Supported by Link Layer [13]

2.4 Advertising Events

Now that advertising sets are defined, advertising events are explored in depth. BLE was first introduced with BT version 4.0. At this time, BLE did not yet integrate extended advertising. In order to support the functionality of extended advertising, SIG introduced new advertising event types. SIG refers to the pre-extended advertising event types as legacy PDUs while the newly added ones are referred to as extended advertising PDUs [13]. BLE's legacy advertising PDUs include advertising indications (ADV_IND), direct advertising indications (ADV_DIRECT_IND), non-connectable advertising indications

(ADV_NONCONN_IND), and scannable advertising indications (ADV_SCAN_IND). The definitions for connectable and scannable mentioned previously clarify the functionality of each of the legacy PDUs.

BLE's new extended advertising PDUs include extended advertising indications (ADV_EXT_IND), auxiliary advertising indications (AUX_ADV_IND), auxiliary synchronous indications (AUX_SYNC_IND), and auxiliary chain indications (AUX_CHAIN_IND). Notably, there are additional essential legacy and new extended PDUs that are non-advertising ones and are depicted in Table 5 below. In general, the PDUs already mentioned are the packets used when an advertiser is initiating and executing an advertising event, prior to receiving a connect or scan request from another node. The additional PDUs shown in the table are ones that are used further along in the advertising event mainly for when connections are established, scan requests are transmitted, and scan responses are transmitted.

The new extended PDUs are slightly more complex and are described further. ADV_EXT_IND PDUs are utilizable in all advertising events except for connectable and scannable undirected and initiates the extended advertising event. These packets are sent on the primary advertising channel(s) as shown in the table below. AUX_ADV_IND PDUs are utilizable in all the same advertising events as ADV_EXT_IND. These packets are the first fragment of advertising data sent on the secondary channels, as indicated by the term "auxiliary." AUX_SYNC_IND PDUs are used in periodic advertising, which is defined further in a later section. This PDU is sent at regular intervals, referred to as periodic advertising intervals, on the secondary channels. AUX_CHAIN_IND PDUs are used to hold additional advertising data and its superior PDUs are AUX_ADV_IND, AUX_SYNC_IND, AUX_SCAN_RSP, or another AUX_CHAIN_IND PDU. The table indicates which channels are used by the AUX_CHAIN_IND PDU.

Table 5: Physical Channel Usage for Advertising Event Types (Adapted from [13])

PDU Type	PDU Name	Physical Channel	Permitted PHYs		
			LE 1M	LE 2M	LE Coded
0000b	ADV_IND	Primary Advertising	•		
0001b	ADV_DIRECT_IND	Primary Advertising	•		
0010b	ADV_NONCONN_IND	Primary Advertising	•		
0011b	SCAN_REQ	Primary Advertising	•		
	AUX_SCAN_REQ	Secondary Advertising	•	•	•
0100b	SCAN_RSP	Primary Advertising	•		
0101b	CONNECT_IND	Primary Advertising	•		
	AUX_CONNECT_REQ	Secondary Advertising	•	•	•
0110b	ADV_SCAN_IND	Primary Advertising	•		
0111b	ADV_EXT_IND	Primary Advertising	•		•
	AUX_ADV_IND	Secondary Advertising	•	•	•
	AUX_SCAN_RSP	Secondary Advertising	•	•	•
	AUX_SYNC_IND	Periodic	•	•	•
	AUX_CHAIN_IND	Secondary Advertising and Periodic	•	•	•
1000b	AUX_CONNECT_RSP	Secondary Advertising	•	•	•
All other values	Reserved for future use				

2.4.1 Advertising Interval

Now that utility of the various advertising PDU types and advertising sets have been discussed, the timing between advertising events are defined. The timing between advertising events is important as it ties in directly with jamming; the next section puts the two together. When advertising events are of the undirected or connectable directed classes and are used in a low duty cycle mode (described later in this section), the time between the start of two consecutive advertising events ($T_{advEvent}$) for the same advertising set is computed using the following formula [13]:

$$T_{advEvent} = advInterval + advDelay \quad (2-1)$$

In this formula, the advertising interval ($advInterval$) is an integer multiple of 0.625ms in the range [20ms, 10,485.759375s]. The $advDelay$ parameter is a pseudo-random value, generated by the link layer for each advertising event, in the range [0, 10ms]. Regarding this pseudo-random value, Silicon Labs states: “This randomness helps reduce the possibility of collisions between advertisements of different devices” [16]. Figure 9 below illustrates how the advertising events are perturbed in time using the pseudo random delay parameter.

We can define the time between the start of consecutive PDU packets as “ t_{between} ,” for comprehensibility. T_{between} must be less than or equal to 10ms, as depicted in the figure below, according to SIG’s specifications. The example advertising event in this figure advertises deterministically on channels 37, 38, and 39, in that order. The time spent advertising on the individual channels is less than or equal to 10ms; this time spent is dependent on the amount of data transmitted in that PDU itself. According to SIG’s core specification document, “the advertising physical channel PDU has a 16-bit header and a variable size payload” [13]. The payload can be anywhere from 1-255 octets in size. The larger the payload of each advertising PDU, the time between advertising PDU transmissions decreases.

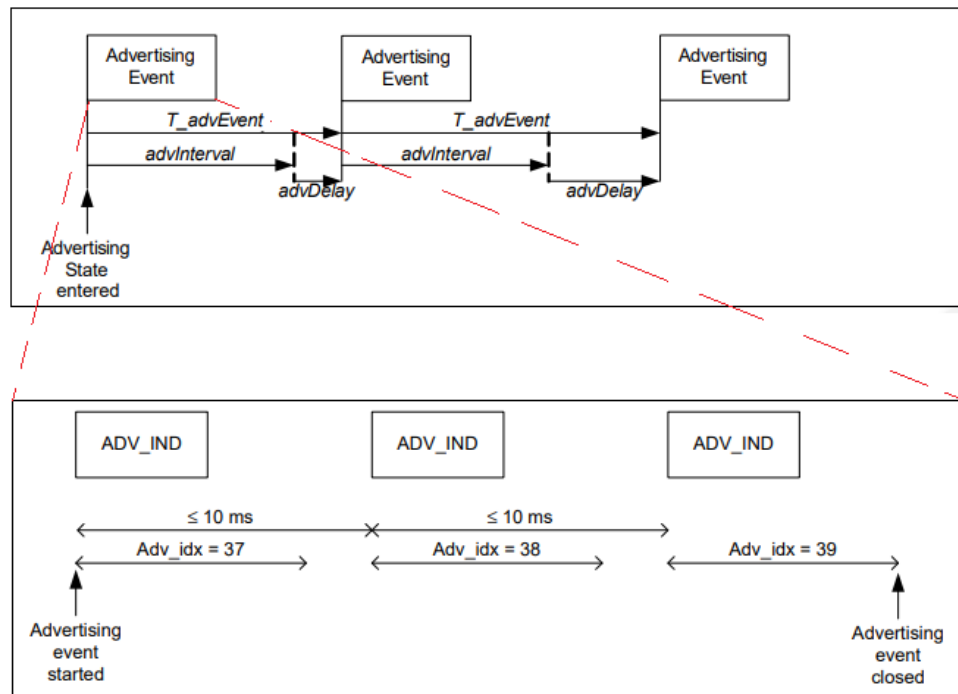


Figure 9: Advertising Events Perturbed in Time via advDelay (Adapted from [13])

2.4.2 Extended advertising events

SIG defines extended advertising as advertising events that send packets on the secondary advertising channels. SIG introduced extended advertising with the release of BT 5.0 in December 2016. Extended advertising events begin at the same time as the advertising event and end with the transmission of the final packet in that advertising event plus subordinate sets. The PDUs sent on the secondary channels are referred to as auxiliary packets. Notably, overlapping extended advertising events are implementable through configuration of the PDU for extended advertising, ADV_EXT_IND.

2.4.3 Periodic advertising events

SIG defines periodic advertisements as a subset of extended advertisements and are used to broadcast packets at a set period, between two unconnected devices. The AUX_SYNC_IND PDU is used for synchronous advertising on the secondary advertising channel that does not expect a response [13]. Regarding the purpose of periodic advertising, SIG developer Kai Ren states: "...periodic advertising allows the scanner to sync with the advertiser so the scanner and advertiser wake up at the same time" [17]. SIG defines the interval period of this type of synchronous advertising scheme as an integer multiple of 1.25ms in the range [7.5ms, 81.91875s]. The interval must remain constant for the entirety of the periodic advertising event. Figure 10 below illustrates how periodic advertising events originate from the same advertising set.

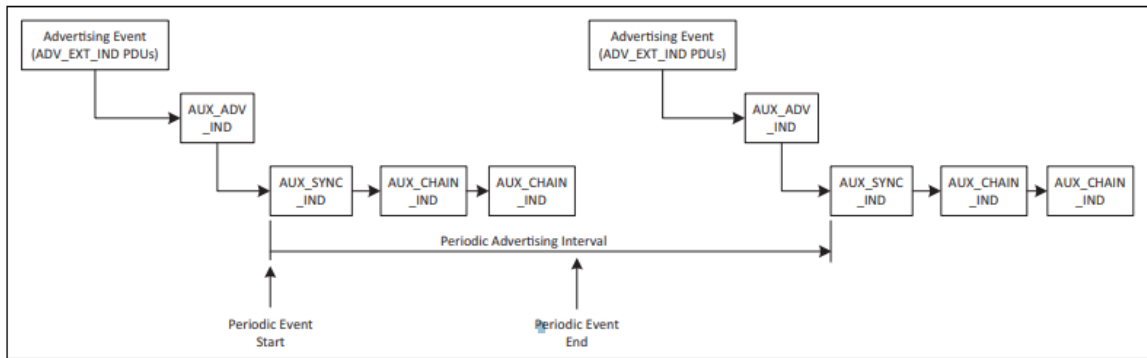


Figure 10: Synchronous Advertising Events [13]

2.4.4 Low vs. High Duty Cycle Advertising

In addition to the timing criteria of advertising events defined by equation 2.1, there are other timing requirements within the advertising events themselves, specifically between each transmitted PDU. For the connectable directed PDU type, the advertising event is configurable in a low duty cycle or high duty cycle mode. SIG clarifies the difference between the modes in the following statements: "Low duty cycle connectable directed advertising is designed for cases where reconnection with a specific device is required, but time is not of the essence or it is not known if the Central device is in range or not. High duty cycle connectable directed advertising is designed for cases in which fast Link Layer connection setup is essential (for example, a reconnection)" [13]. High duty cycle mode should only be utilized when a connection setup speed is critical as it is power and bandwidth intensive.

In low duty cycle mode, the time between the start of two consecutive ADV_DIRECT_IND PDUs transmitted within an advertising event must be less than or equal to 10ms. In high duty cycle mode, the time between the start of two consecutive direct indication PDUs is less than or equal to 3.75ms; in this mode the link layer will exit the advertising state no later than 1.28s after the advertising state initializes.

For the connectable directed event type, the t_{between} of ADV_EXT_IND PDUs is less than or equal to 10ms; this tells us that in extended advertising, the low duty cycle mode's timing criteria is default. For the scannable undirected event type, the t_{between} of ADV_SCAN_IND PDUs and scannable undirected ADV_EXT_IND PDUs is less than or equal to 10ms. Just as in the advertising events described, the remaining advertising event types all have the same criteria for t_{between} being less than or equal to 10ms. These observations are indicative of low duty cycle mode being more commonly implemented; high duty cycle mode is typically only used in applications where a fast connection setup is optimal.

2.5 Relating Advertising Schemes and Jamming

Jamming a BLE mesh network requires a significant amount of understanding of the advertising state of the link layer. Now that the various nuances of BLE's advertising scheme are described in detail, a scheme for jamming BLE networks is proposed. It is important to note here that the effectiveness of jamming the advertising channels of BLE devices will be dependent on the BLE version of the victim device. BLE devices of Bluetooth versions 4.2 and earlier can be jammed relatively easier compared to BLE devices of versions 5.0 and later; the extended advertising option of BT versions 5.0 and later allows a BLE network developer to make their scheme for advertising more robust to jamming. However, the extended advertising event type is only an option; BT 5.0 and later devices may not necessarily implement this feature.

2.5.1 Jamming BLE versions 4.2 and earlier

Jamming the advertising channels, resulting in a denial of service (DoS) malfunction, of BLE 4.2 and earlier devices is relatively straightforward. These BLE devices can only advertise on the primary advertising channels. If a designer of a jammer system were to design a jammer as a continuous one, all three primary advertising channels could be continuously jammed. The transmission of three continuous jamming tone

signals corresponding to the three center frequencies (2.402GHz, 2.426GHz, & 2.480GHz) of the primary advertising channels could result in devices not being able to connect, not being able to send/receive beacons, etc. Transmitting three continuous tones requires a transmitter with three output channels or three individual transmitters.

An RF generator with three output channels is rather expensive for the proposed jammer system. Three individual transmitters necessitates three individual synthesizers. Three individual synthesizers would be expensive for an overall system design as well. For these reasons, a frequency hopping approach to jamming is tested, using a single synthesizer, and discussed in a later section. In order to successfully jam via frequency hopping, the timing of the frequency hopping will have to correlate to the timing of the advertising PDU transmissions on the primary advertising channels. Alternatively, the average power of each of the jamming signals, one-third in magnitude relative to a continuous tone jamming signal, may be sufficient for the purpose of jamming. Both the continuous jamming approach and frequency hopping approaches are tested and discussed in a later section.

The figure below depicts an example of a temperature sensor acting as an advertiser and a phone acting as a scanner. The temperature sensor advertises deterministically on the primary advertising channels. When the sensor advertises on channel 37, the scanner responds with a scan request which results in a scan response from the sensor. The advertising packet transmissions on the primary advertising channels are within 10ms of one another. Jamming of the primary advertising channels may be possible using the continuous jamming and/or frequency hopping approaches.

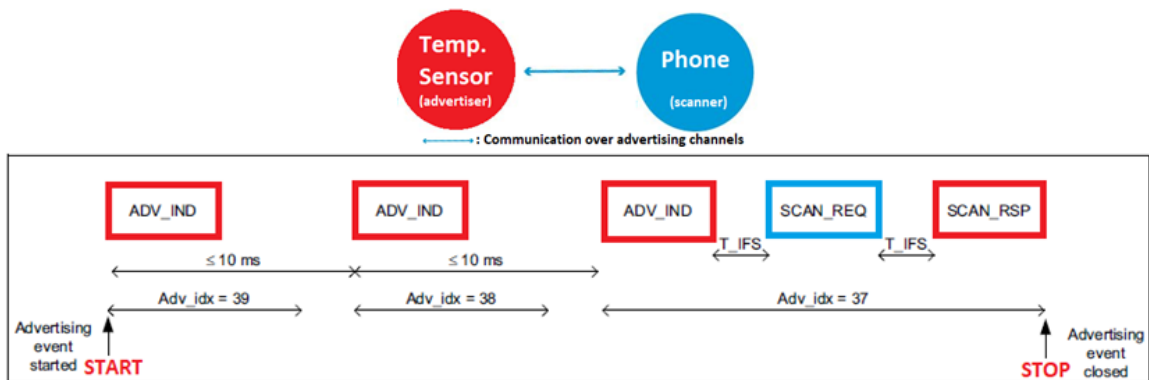


Figure 11: Advertising Event Example (Adapted from [13])

2.5.2 Jamming BLE versions 5.0 and later

Jamming BLE 5.0 and later devices is rather complicated due to the BLE network developer’s ability to utilize secondary advertising channels. Notably, it is up to the developer on whether or not to even utilize the secondary channels in the first place. If the developer chooses not to utilize extended advertising, the approaches mentioned for jamming BLE versions 4.2 and earlier remain valid.

As mentioned previously, a BLE device that implements extended advertising must initially start by advertising on the primary advertising channels. There are 37 data channels that are configurable as secondary advertising channels. The figure below depicts the advertising and data channels co-existing with Wi-fi channels 1, 6, and 11, which are the most commonly used. When a BLE developer chooses to implement secondary advertising, they must choose which data channels to utilize. The figure shows minimal overlap of the popular Wi-fi channels and the following ten data channels: Ch. 8, 9, 10, 21, 22, 23, 33, 34, 35, & 36. These channels have the least amount of electromagnetic interference due to Wi-fi and are the most likely candidates for secondary advertising.

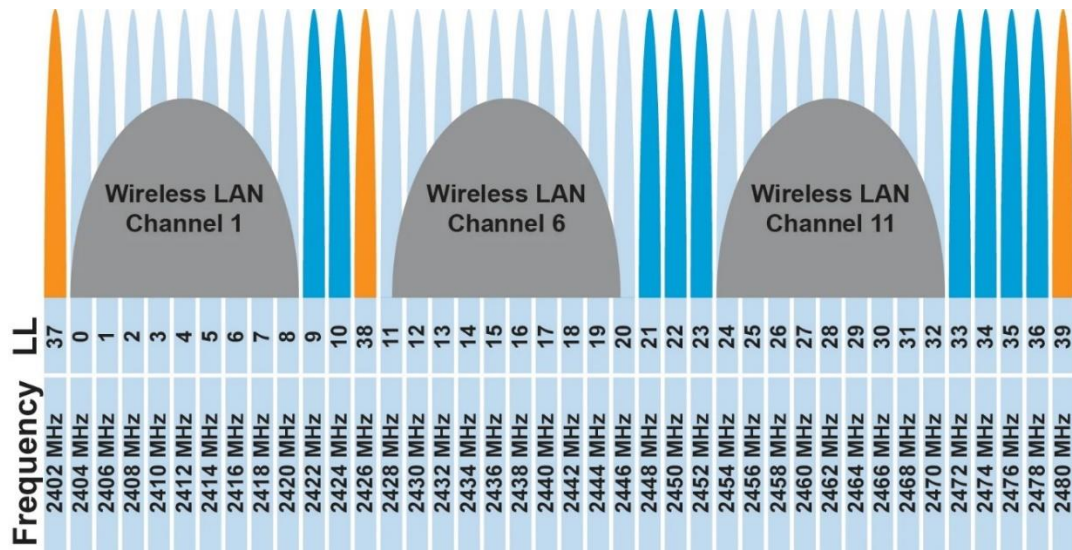


Figure 12: BLE Co-existence with Wi-fi [18]

A jamming strategy is explored now that the most viable secondary channels are identified. A transmitter with 13 output channels (3 primary + 10 secondary) or 13 individual frequency synthesizers are necessary for creating a continuous jammer for networks utilizing BLE 5.0 or later. Alternatively, jamming is possible

using less than 13 frequency synthesizers if the synthesizers are able to hop frequencies at a quick enough rate. In order to ensure one-hundred percent confidence in the jamming scheme, the utilization of a sniffer is necessary. In other words, there is no guarantee that the developer of the BLE network chooses to utilize the 13 channels and could utilize the secondary channels that are depicted to have Wi-fi interference.

Chapter 3

DESIGN OVERVIEW

This project has two objectives. The main objective is to create a system that will create three jamming tone signals to exploit the weakness in Bluetooth Low Energy's protocol and prevent a mesh network from establishing node to node communication. This system will produce a tone in each of the three advertising channels (Ch. 37, 38, 39) in the 2.4GHz ISM band. The secondary objective is to create a mesh network that will be the victim network of the jammer system. Figure 13 depicts the three jamming tones overlaid on top of the PHY spectrum of victim device.

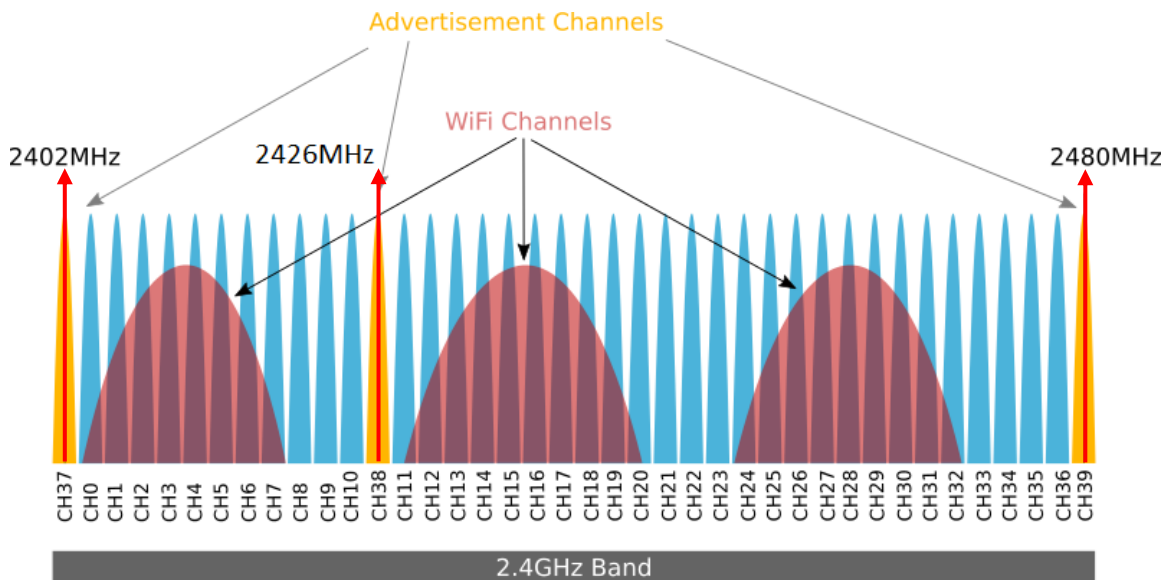


Figure 13: BLE Channelization with Three Jamming Tones Overlaid (adapted from [12])

3.1 Preliminary High-level Overview

This section provides a high-level overview of the preliminary Bluetooth Low Energy jammer system design. Modifications made to the preliminary system design are discussed in the “Frequency Hopping Jammer System Overview” and “Constant Jammer System Overview” sections of Chapter 5 and 7, respectively; these modifications were made due to COVID-19 related supply chain issues, lack of access to laboratory equipment, and cost of components. Figure 14 below depicts the preliminary high-level block diagram for

the system which includes: a microcontroller (MCU), three individual synthesizers, a power combining circuit, a power amplifier (PA), and an antenna². The MCU will communicate with the individual synthesizers and configure them to produce signals with center frequencies corresponding to those of the BLE advertising channels. The synthesized signals are combined into one signal via a Wilkinson power combiner circuit. The power amplifier will then amplify the power combined signal and an antenna will transmit the three-tone signal. In the early block diagram, the MCU, three synthesizers, power combiner circuitry, and a power amplifier are placed on a single PCB. The victim BLE mesh, depicted on the right, includes three devices: a central one and two peripherals.

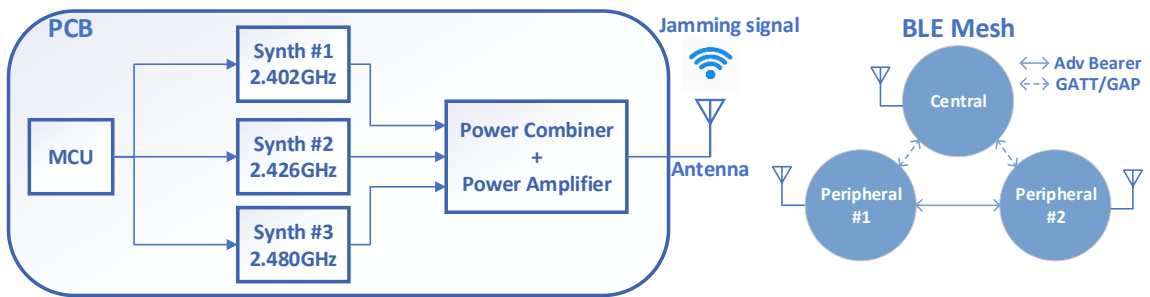


Figure 14: Preliminary High-level Block Diagram

3.2 Component Selection

The individual components of the jammer system and BLE mesh are researched following the conceptualization of the initial system design. Instead of looking up all of the components at the same time, the most critical ones are selected first as to refine the search criteria for the remaining ones. The synthesizer selection was prioritized as its functionality ultimately determines the necessity of power combining and amplification. The selection of the synthesizer relied on the following criteria: output frequency range, frequency resolution, output power, IC packaging, and digital interfacing capabilities. The remaining components of the system are selected following the selection of the synthesizer. The selection of devices for the BLE mesh relied on the following criteria: version of BLE, receiver antenna gain, and mesh networking capabilities.

² The thesis project has a budget of \$200.

3.2.1 Synthesizer Selection

The search for a viable synthesizer IC was conducted on Digikey. The synthesizer must be able to output frequencies ranging from 2.4G – 2.4835GHz. The greatest common divisor (GCD) of the frequency differences between channels 37 & 38 ($2426 - 2402 = 24\text{MHz}$) and between channels 39 & 37 ($2480 - 2402 = 78\text{MHz}$) determines the frequency resolution required; $\text{GCD}(78\text{M}, 24\text{M}) = 6\text{MHz}$. The frequency resolution of the synthesizer must be at least 6MHz. The output power of the synthesizer must be at least -20dBm, which is the lower limit of a SIG certified BLE transmitter's output power. The IC packaging is critical, as RF synthesizer ICs are expensive, typically ranging from \$10-\$150, and any mistakes in the bring-up phase of the PCB could be costly. Additionally, the packaging highly impacts the layout size and layout complexity of the PCB. When considering the packaging, the availability of ICs with an integrated VCO is also considered. Lastly, the digital interface of the synthesizer IC is critical as the IC interfaces with an MCU.

Analog Device's (ADI) HMC1035LP6GE high performance clock generator met the aforementioned criteria and was determined to be the best choice for the synthesizer. This synthesizer is digitally controlled (SPI), has an integrated VCO, comes in a 40-VFQFN (Very Flat Quad Flat No-lead) exposed pad package, has two programmable dividers to achieve the necessary frequency resolution, and satisfies the output power and frequency requirements [19]. Methods for testing and characterizing the synthesizer are explored. With GHz RFICs it is often difficult to test and characterize their performance, as they require a PCB that is properly laid out and fabricated. For the PCB to function optimally, controlled impedance traces, impedance matching, bypass components, and other critical components would be required.

Evaluation boards are often useful in verifying the capabilities of an IC, such as the selected synthesizer. Datasheets of ICs often do not include all the data necessary to fully understand the IC's capabilities. The ADI EVAL01-HMC1035LP6G evaluation board found online is a feasible solution for quick testing and characterization of the synthesizer. The evaluation board is very expensive, coming out to more than \$700 after taxes. Thanks to a generous donation from Analog Devices, the acquisition of an EVAL01-HMC1035LP6G evaluation board was possible. Compared to the cost of a single HMC1035LP6GE IC, approximately \$50 after taxes, this evaluation board is costly. The evaluation kit includes an evaluation PCB

and a USB board which interfaces with a PC. The evaluation board itself has a single SMA connection for an optional external reference and two additional SMA connections for the differential output of the synthesizer. The board requires a supply voltage of 5.5V. The figure below displays the evaluation board.



Figure 15: EVAL01-HMC1035LP6GE Evaluation Board [19]

3.2.2 Power Combining Circuitry Selection

The synthesizer evaluation board was selected primarily for preliminary testing and characterization of the synthesizer IC. The proposed jammer system incorporates three individual synthesizer ICs on a single PCB. Following the selection of the synthesizer IC, the necessity of power combination circuitry was assessed. The Wilkinson power combiner is a feasible option for power combining at 2.4GHz. This circuit, invented by an engineer named Ernest Wilkinson, is generally useful for operating frequencies in the 300M to 300GHz range. Theory teaches, a lossless 3-port device with power combination and isolation is not possible. By adding the resistor between the two input ports Wilkinson successfully allowed all three of the ports to be matched. At the center frequency of the circuit, the two input ports are fully isolated from one another; in theory the Wilkinson circuit will operate with 100% efficiency but that is not often the case as power dissipation of the resistor is inevitable.

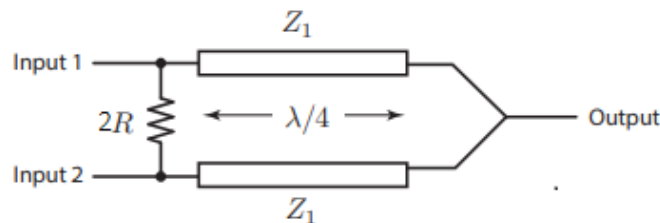


Figure 16: Distributed Implementation of 2-way Wilkinson (Adapted from [20])

The “ Z_1 ” impedances symbolize quarter wave transformers, which are essentially impedance inverters; when loaded by resistance, R , on one end, the Thevenin resistance of the other end appears as an impedance of Z_1^2/R . If the Z_1 impedance is $50\sqrt{2}\Omega$ and the resistance is 50Ω , the Thevenin resistance of the other end appears as 100Ω . To clarify, if the source resistances of inputs 1 and 2 are 50Ω , the Thevenin resistance looking into the circuit from the output will appear as 50Ω since their transformed impedances of 100Ω are in parallel. By making the Z_1 impedance equal to $\sqrt{2}$ times the characteristic impedance of the system, the three ports are matched. Figure 17 provides a visualization of this concept.

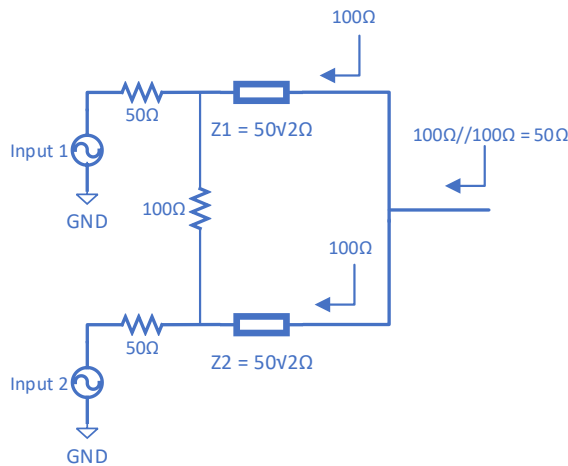


Figure 17: Wilkinson Circuit Decomposition

The following concept describes how the two input ports are isolated from one another. Consider a signal that is input to the input 1 port. Part of the signal goes clockwise through the resistor and part goes clockwise through the upper quarter wave transformer towards input port 2. The second clockwise signal at input port 2 ends up half in power as half of the power is output to the output port. The first clockwise signal at input port 2 ends up half in power and equal in amplitude to the second clockwise signal. The first clockwise signal and second clockwise signal are 180 degrees out of phase due to the half wavelength (quarter wave from upper + quarter wave from lower) traveled by the second clockwise signal. At input port 2, the two clockwise signals subtract to zero, under ideal circumstances.

The Wilkinson is advantageous in that the concepts of the distributed implementation are extendable to a lumped implementation. At 2.4GHz the signal wavelength is approximately 12.5cm (~4.92in). With this wavelength it is feasible to create either a distributed or lumped version of the Wilkinson power combiner.

The figure below depicts the lumped implementation of a 2-way Wilkinson [20]. The C, L, C ‘pi’ networks between the input ports and the output port are analogous to the quarter wave transformers of the distributed model. The output capacitor is two times C as capacitors add when in parallel.

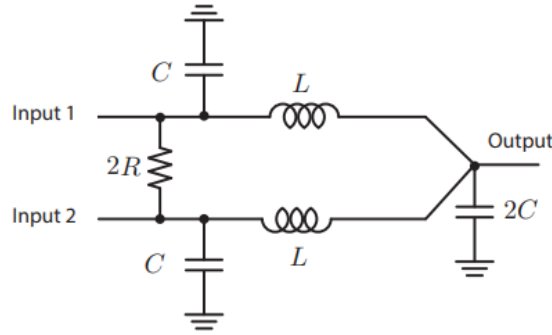


Figure 18: Lumped Implementation of 2-way Wilkinson Power Combiner (Adapted from [20])

The Wilkinson power combiner is also advantageous in that it can be generalized to an N-way power combiner. For the jammer system, three RF signals could theoretically be power combined via a 3-way Wilkinson. Noreiga and Gonzalez propose a lumped implementation for a 3-way Wilkinson in their paper [21]. Their topology is based off an empirical tuning approach where they simplified the circuit configuration by removing “non-critical” elements without noticeably degrading the performance of the circuit. The figure below depicts this circuit, created in LTSpice. The three voltage sources symbolize the three individual synthesizers. The “output” of this circuit loads into a 50Ω termination, which would be the impedance of the power amplifier of the preliminary block diagram (Figure 14). The lumped component values are calculated based off equations shown in Figure 19, where the center frequency (f_o) is the mean of the channels 37 (2402MHz) and 39 (2480MHz) and the characteristic impedance (Z_o in the equation) of the $\lambda/4$ transformer is $50\sqrt{2} \Omega$. R_5 , R_6 , and R_7 are sized to be the characteristic impedance of the circuit. C_o is useful for tuning out resistor and pad parasitics of components; this component can take values between 0.5pF – 2pF.

$$C_p = \frac{1}{2 \cdot \pi \cdot f_o \cdot Z_o} \quad (1)$$

$$L_s = \frac{Z_o}{2 \cdot \pi \cdot f_o} \quad (2)$$

Figure 19: LC Component Sizing Equations [21]

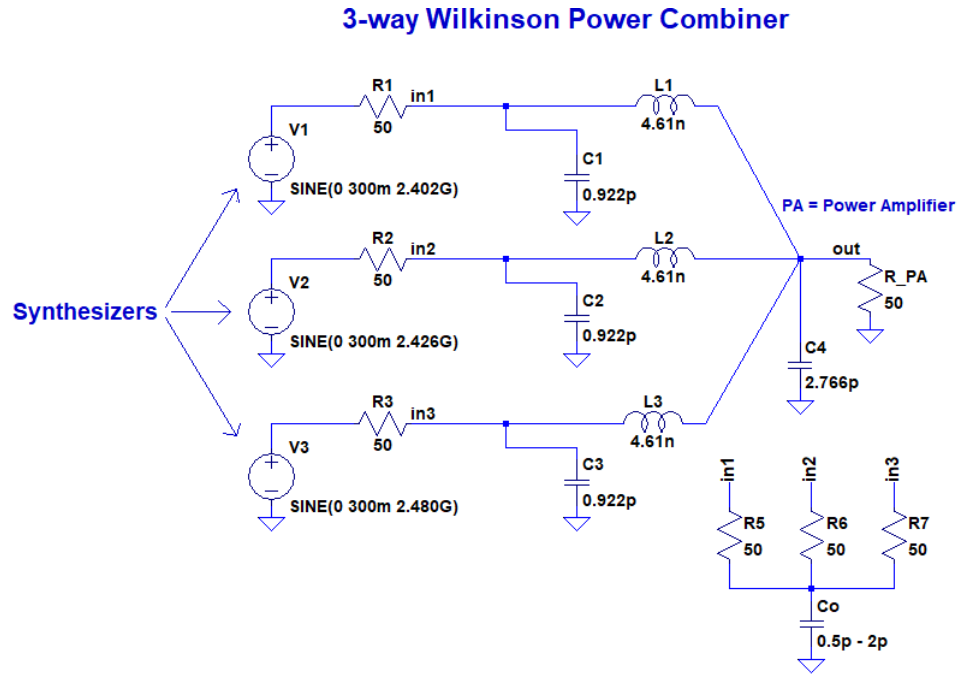


Figure 20: Schematic of Lumped Element 3-way Wilkinson Power Combiner

3.2.3 Power Amplifier and Antenna Selection

Depending on the performance of the proposed PCB jammer system, a power amplifier may or may not be required. The necessity of a power amplifier can be more accurately assessed following the characterization of the performance of the synthesizer evaluation board in the “Signal Characterization – Synthesizer” section of Chapter 5. Assuming the proposed jammer PCB has an SMA output (same as evaluation board) the power amplifier would also have such connectors. The selection of a power amplifier is discussed in a later section. SMA connections for the PCB, synthesizer evaluation board, and power amplifier are ideal, as many 2.4GHz antennas come with SMA connections. Selecting an antenna was relatively simple as there is an abundance of 2.4GHz applications relating to Wi-Fi, Bluetooth, ZigBee, etc.. A Siretta Ltd. Delta6B 2.4GHz whip tilt antenna was chosen for the jamming signal transmission. This antenna works in the 2.4 to 2.5GHz range and has a gain of 5dBi.

3.2.4 Bluetooth Mesh Selection

To prove the concept of Bluetooth Low Energy jamming, a victim BLE mesh must be built. The requirements for the Bluetooth devices of the mesh are the following: must be Bluetooth version 5.0 or later and must be

BLE-mesh enabled. Notably, Bluetooth 5.1 and 5.2 versions exist but these versions came out in January 2019 and January 2020, respectively. For this reason, there are not many Bluetooth mesh devices equipped with these newer technologies.

A search of SIG's member directory provides significant insight on a company's investment in Bluetooth technologies. The search was also useful for identifying the leading companies in the field. The assumption is that the products of the leading companies represent the current state of art in the field of BLE applications. SIG has two tiers of membership, adopter and associate. The associate membership requires an annual membership of \$35,000 while the adopter membership is free. Of the associate members, Cypress Semiconductor is one of the leaders in Bluetooth programmable system-on-chip (PSoC) solutions; in the past, Cypress executives served as Associate Member Directors, who are selected by the SIG board of directors.

Cypress's BLE mesh product catalog includes several options for developing a viable BLE mesh network to serve as a victim to the proposed jamming. Cypress's CYBT-213043-MESH EZ-BT module mesh evaluation kit includes the necessary components to create a viable victim BLE mesh system. The kit is approximately \$125 after taxes. This kit includes four evaluation boards with the following key specifications: Bluetooth 5.0 enabled, mesh compatible, and programmable [22]. The boards are USB-powered (optional coin-cell battery powering as well) and programmed. These boards are selected for the victim mesh network as Cypress's products are assumed to represent the current state of art for BLE devices. Programming of the board utilizes Cypress's ModusToolbox IDE. Thanks to a generous sponsorship from Cypress Semiconductor, the acquisition of a CYBT-2130443-MESH kit was possible. The evaluation boards included in the kit are displayed in the figure below [22].



Figure 21: CYBT-213043-MESH Kit Contents [22]

Chapter 4

BLE MESH NETWORK DEVELOPMENT

4.1 Establishing the Mesh Network

In order to test the functionality of the BLE jammer, a victim Bluetooth mesh network is necessary. Cypress's CYBT-213043-MESH EZ-BT module mesh evaluation kit was chosen for the development of a BLE mesh network. The four individual evaluation board devices in the kit are enabled with Bluetooth version 5.0 and are mesh-compatible. Programming of the boards utilizes Cypress's ModusToolbox IDE. The figure below depicts the IDE user interface.

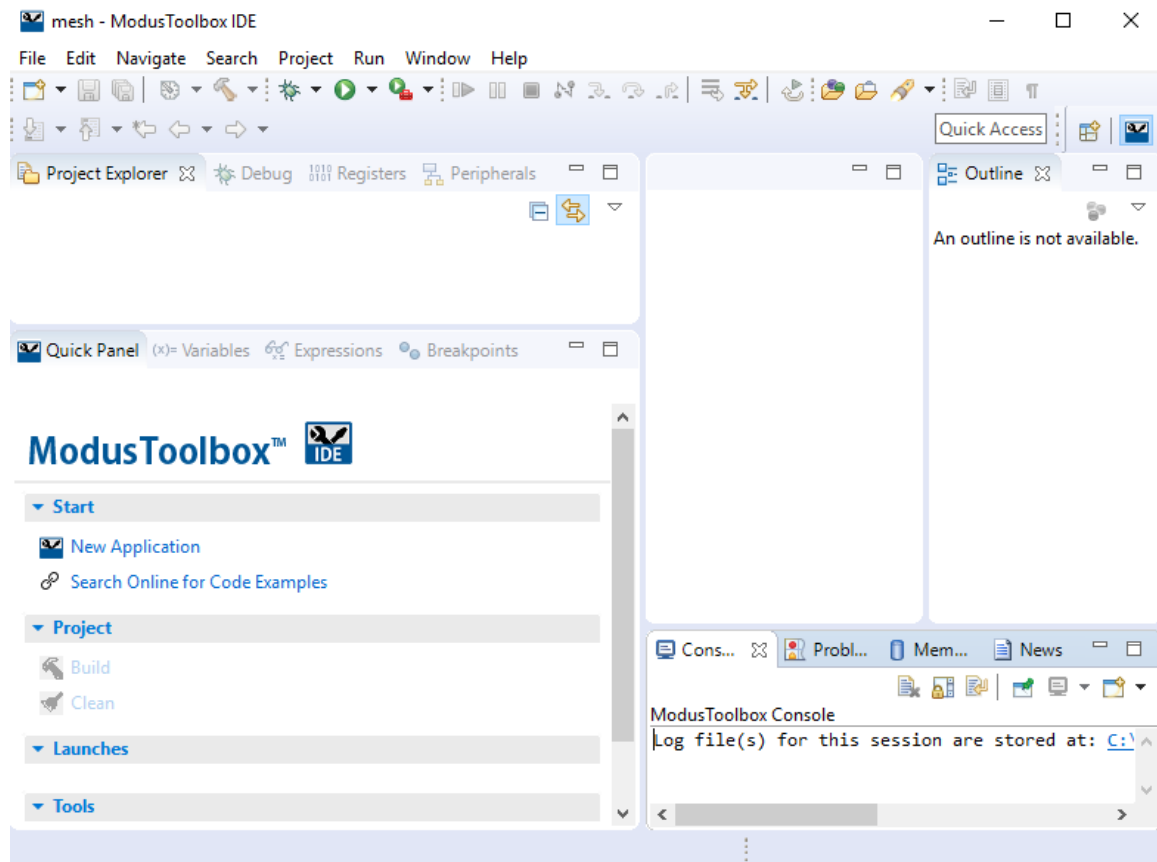


Figure 22: ModusToolbox IDE

Cypress provides a select amount of start applications for use with their various kits. The CYBT-213043 kit in particular requires the creation of a starter application called "wiced_bt sdk" once per workspace [23]. This application contains the software development kit (SDK), board support packages (BSP), and libraries that

are shared by the various Bluetooth applications in a workspace. Following the creation of the “wiced_btSDK” application, a new application can be created in the same workspace. The starter applications provided by Cypress contain multiple projects for BLE mesh models.

The “wiced_btSDK” application includes many source files that are critical for the implementation of BLE mesh applications. The “wiced_bt_cfg” file is one of the most important as far as defining the scheme for advertising and all of Cypress’s BLE starter applications utilize it. The figure below is a screenshot of code from the “wiced_bt_cfg” file which shows that the BLE advertising scheme for all generic mesh programs utilize all three of the primary advertising channels and no secondary advertising channels. It is important to note here that a leading Bluetooth SoC company such as Cypress does not implement secondary advertising by default to their mesh starter applications. This is concerning as communication of Cypress mesh boards is corruptible by merely jamming the three primary advertisement channels. SIG’s decision to not have secondary advertising as a mandatory setting for BLE meshes creates a significant vulnerability in the security of mesh devices/networks.

```
/* BLE advertisement settings */
{
    .channel_map                = BTM_BLE_ADVERT_CHNL_37 |
                                BTM_BLE_ADVERT_CHNL_38 |
                                BTM_BLE_ADVERT_CHNL_39,
```

Figure 23: BLE Advertising Channels Configured in ‘wiced_bt_cfg’ File

To create a BLE mesh network, an understanding of the specifications for a BLE mesh is necessary. Bluetooth SIG software developer, Martin Woolley, states: “A Bluetooth mesh network is like an exclusive club. If you’re a member of the club, you can enter the club and make use of those facilities and services which your membership allows. If you’re not, you aren’t allowed through the front door, no matter what you say” [24]. Within a BLE mesh, the specifications of various applications govern the interaction between the devices of the network. For example, a BLE mesh light switch device can turn a light on/off as it is part of the lighting application. This light switch device cannot switch on another system, such as an air conditioning system, because the air conditioning system is not a part of the lighting application. Provisioning is the secure process, accomplished using an application, for adding devices to a BLE mesh network. Provisioning devices are typically smartphone or tablet applications.

The provisioner, not necessarily a BLE mesh device, is responsible for transforming devices into nodes of the mesh. The provisioning protocol functions over either of the provisioning bearers (PB), PB-Generic Attribute Profile (PB-GATT) or PB-Advertising (PB-ADV). According to Kai Ren of SIG, “a provisioning bearer layer enables the transportation of provisioning PDUs between a provisioner and an unprovisioned device” [25]. When a provisioner does not support PB-ADV it uses PB-GATT; PB-GATT utilizes the proxy protocol to enable nodes to send and receive provisioning PDUs over a BLE bearer [13]. PB-ADV on the other hand transmits generic provisioning PDUs; a device that supports PB-ADV passively scans for incoming generic provisioning PDUs at a duty cycle close to 100% (as to avoid missing a PDU). It is important to note that the provisioning procedure between two devices is possible even for devices that utilizes different bearers. For example, an unprovisioned device may be using the PB-ADV bearer and can be provisioned into a network by a provisioner using the PB-GATT bearer [25].

The flowchart shown in the figure below depicts the process for adding a new device to a mesh network. The first step involves beaconing, where an unprovisioned device indicates that it is available for provisioning. At this time, the provisioner must be enabled to receive the advertising packets from the unprovisioned device. Next, an invitation is sent to the device to-be-provisioned in the form of a provisioning invite PDU. The beaconing device sends the provisioner a provisioning capabilities PDU which includes information used in the authentication step. In the next step, the FIPS P-256 Elliptic Curve Algorithm based asymmetric cryptography creates a secure channel to continue the rest of the provisioning process [25]. Public keys are exchanged on the secure channel. The provisioner then instructs the beaconing device to output either a single or multi-digit value, which is entered into the provisioner user interface. A cryptographic hash is then exchanged between the device and provisioner to complete the authentication process. The two devices derive a session key from their private keys and their peer public key; the session key secures the rest of the data distributed through the provisioning process and includes a NetKey and a unique address, allocated by the provisioner, for the newly authenticated device. Now the new device is officially a node and a member of the BLE mesh.

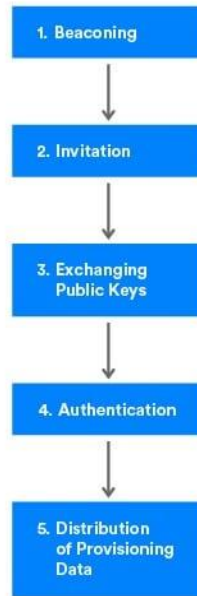


Figure 24: Provisioning Process [24]

4.2 Mesh Network Development

The mesh network discussed in this section is the proposed victim network of jamming. The complexity of the BLE program is not a critical aspect for proving the concept of BLE jamming. As long as the BLE mesh is communicating via BLE and not classic Bluetooth, the jammer should be effective. For this reason, Cypress's starter applications are used as baseline programs to develop the victim mesh network. It is important to note here that Cypress has an application called ClientControlMesh (installed with the ModusToolbox IDE) that assists in the mesh configuration process. Cypress states: "The ClientControlMesh application uses Bluetooth stack of the Cypress silicon. It can support both GATT Proxy and advertising channel to provision and control mesh devices" [26]. To clarify, Cypress is noting that a mesh evaluation board must be connected to the PC for the duration of the mesh configuration process; the ClientControlMesh application will utilize the connected mesh evaluation board's Bluetooth stack instead of using the Windows Bluetooth stack.

The first step to create the mesh network is to program one evaluation board with Cypress's "mesh_provision_client" snip application. The COM port utilized by this provisioner evaluation board is noted. The MeshClientControl application's configuration for this specific evaluation board requires two

parameters - a baud rate of “115200” and the COM port for communication. These two fields are necessary as they are required for communication over the HCI UART [26]. In the next step, a network name is given for the mesh. The “Create” button of the GUI enables the creation of the mesh network. Network attributes such as the mesh UUID and network and application keys are created and stored in a JSON file in the directory where the application resides. An unprovisioned device (device #2) can now join the mesh network. A starter application, “light_dimmable” is loaded onto the unprovisioned device. Another unprovisioned device (device #3) is also programmed but with a different start application, “dimmer.” Both of these starter programs are part of the demo applications provided by Cypress.

Once devices #2 and #3 are programmed, they are ready for the provisioning process. Device #1 provisions Device #2 first. In the ClientControlMesh application user interface, the “Scan Unprovisioned” option is selected. The “Provision UUID” field gets populated with the UUID of the last discovered device (Device #2). Device #1 provisions Device #2 by clicking “Stop Scanning” followed by “Provision and configure.” The figure below depicts the ClientControlMesh interface. Device #3 is provisioned using the same process.

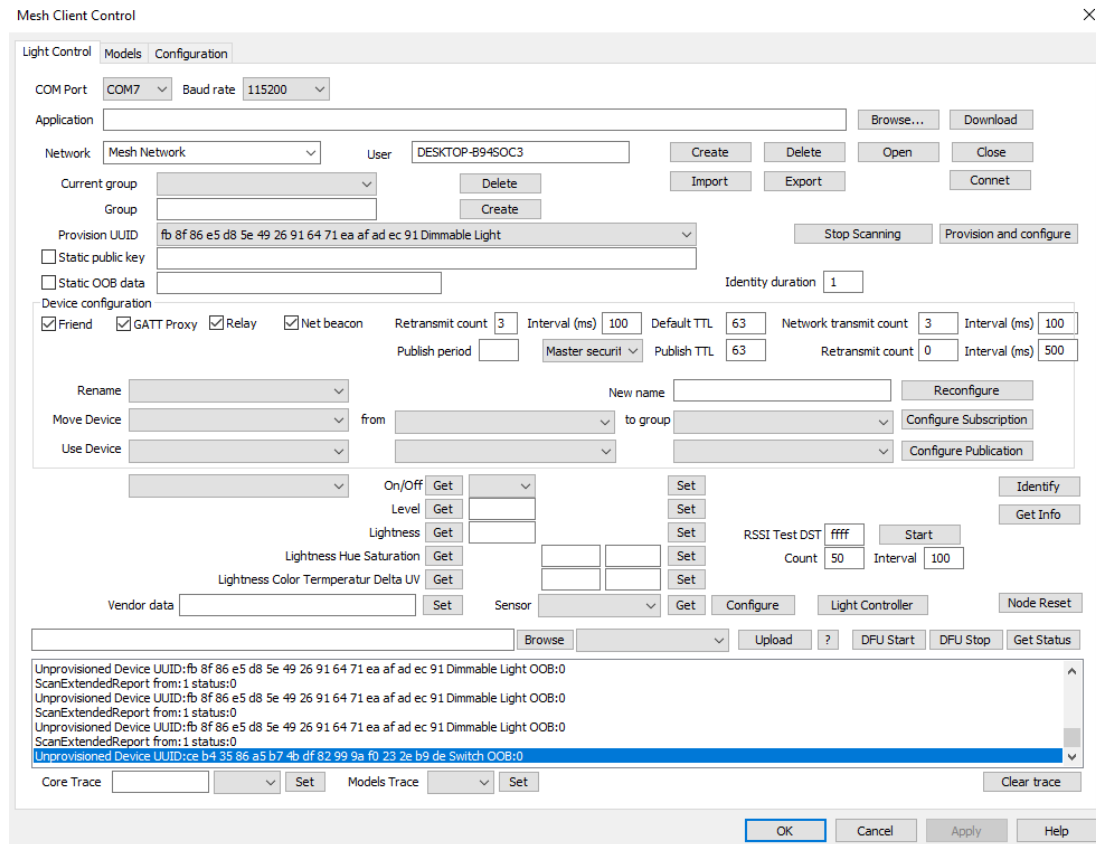


Figure 25: ClientControlMesh Application User Interface

The figure below provides a high-level diagram of the proposed victim mesh network. Device #3, the “dimmer”, has the ability to control an LED of Device #2, the “light_dimmable.” The “User” button on the dimmer board turns the LED on/off. A continuous press (pushed and not released) of the “User” button results in a change, every 0.5 seconds, in the LED’s intensity level from 0 to 100% or vice versa in 8 steps of 12.5% increments, based on the previous LED state. Devices #2 and #3 establish connections on the advertising channels specified in the “wiced_bt_cfg” file, Ch.37 38 & 39. Therefore, these nodes are jammable via the proposed jammer system.

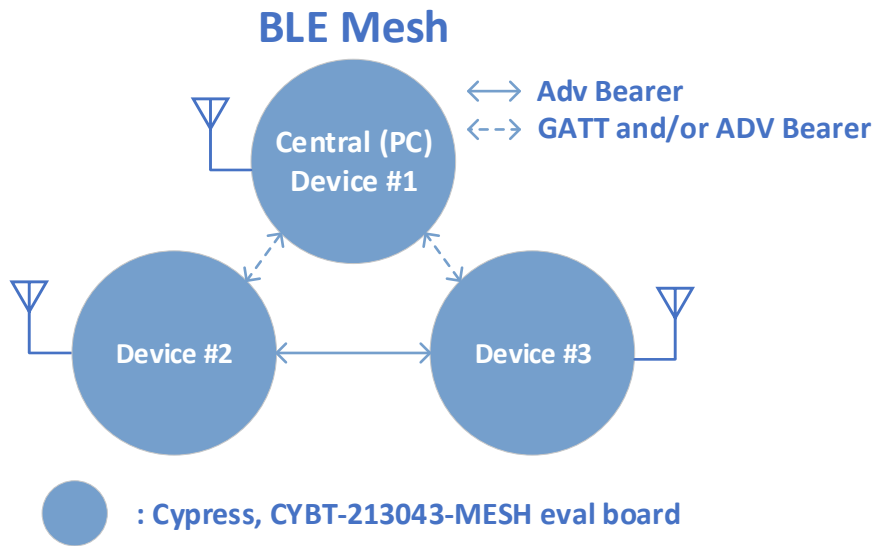


Figure 26: Mesh Network Diagram

The mesh network design is simple but can prove the concept of jamming a victim network. In order to do so a testing scheme must be realized. It is important to note here that these Cypress mesh boards maintain their exclusive membership to the BLE mesh upon power-cycling; Cypress implemented this feature as a default knowing that BLE mesh devices are often battery-powered and require power cycling; power cycling entails disconnecting the power connection (USB), or battery, from the mesh board and connecting it once again. When these mesh devices are turned on, after being turned off, they will advertise their desire to establish a connection with one another. To test the effectiveness of the proposed jammer on these mesh boards, the jammer must be on prior to turning on the mesh devices. If the mesh devices are already turned on before the jammer is, the devices could have potentially already established their connection and would be communicating via the data channels and the jamming signals would be ineffective.

Chapter 5

FREQUENCY HOPPING JAMMER SYSTEM

5.1 Overview

Chapter 3 discussed the initial proposed jammer system. This chapter focuses on the characterization of the functionality (i.e. output power, frequency hopping, etc..) of the synthesizer evaluation board utilized in the thesis development. The ADI EVAL01-HMC1035LP6G evaluation board connects to a USB interface board for utilization of a Hittite Microwave Corporation GUI. The USB interface board connects to the synthesizer evaluation board via a 12-pin connector. The evaluation board itself requires a 5.5V supply, has a single SMA connection for an optional 10MHz reference, and two additional SMA connections for the differential output of the synthesizer. Without an external reference source, the board utilizes a low noise 50MHz voltage-controlled crystal oscillator (VCXO). The figure below depicts the block diagram for the synthesizer evaluation board, USB interface board, and the overall set up [27].

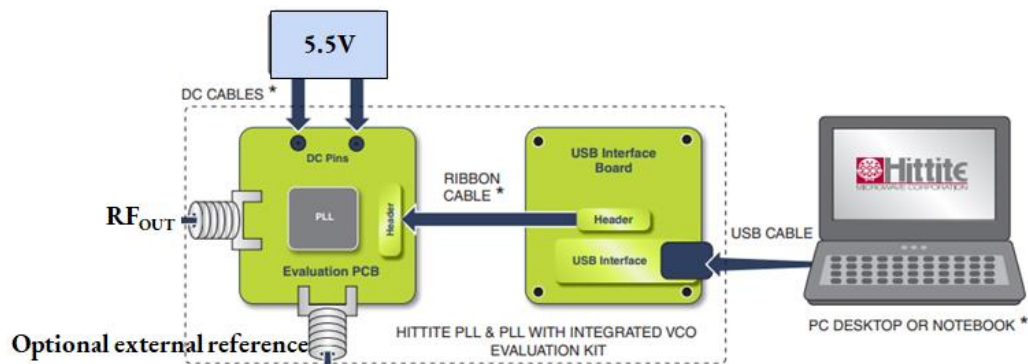


Figure 27: Set up for EVAL01-HMC1035LP6G Evaluation Board (Adapted from [27])

The Hittite GUI is displayed in the figure below. The “Load Reg File” option allows the user to select the configuration file for the synthesizer mode. There are 8 different options which consider the following: performance priority vs. power priority, fractional vs. integer frequency division, and output waveform type, low voltage differential signaling (LVDS) vs. low voltage positive emitter coupled logic (LVPECL). Performance priority entails improved jitter and phase noise performance of the synthesizer; this mode is optimal for driving the sample clock inputs of ADC/DAC’s and high speed SERDES reference clock inputs [28]. Power priority entails a reduced consumption of current from 237mA to 173mA according to the synthesizer datasheet. The N-divider depicted in the synthesizer block diagram of Figure 29 below is 19-bit

resolution for both integer and fractional division. The R-divider is 14-bit resolution. LVDS and LVPECL are both differential signal transmission schemes. LVPECL has a larger differential voltage swing and is less power efficient than LVDS due to its circuit topology [29].

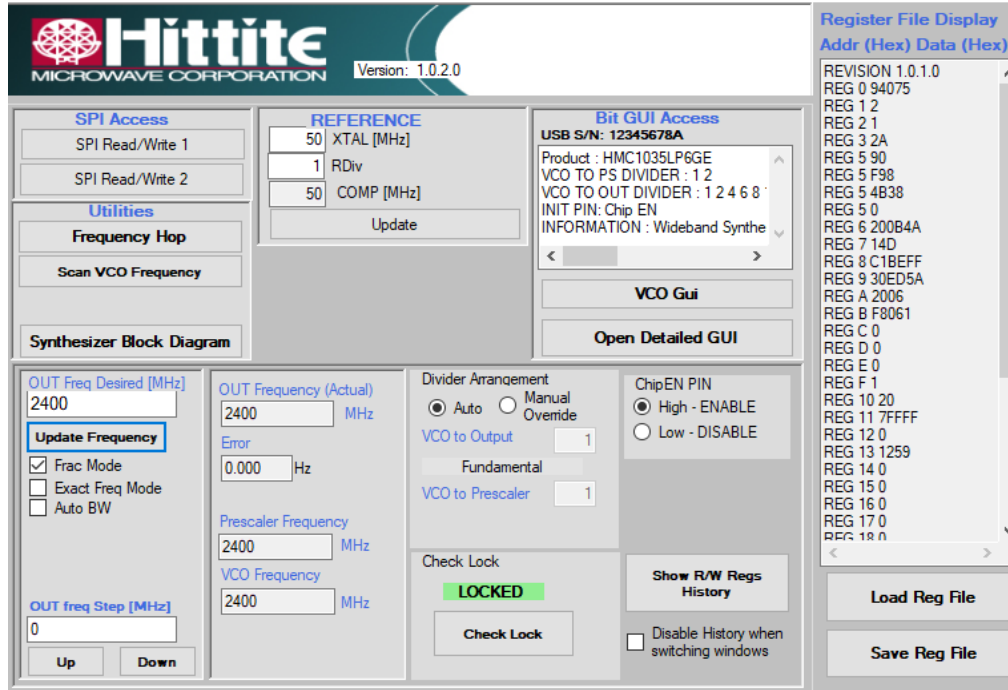


Figure 28: Hittite Clock and Timing Main GUI

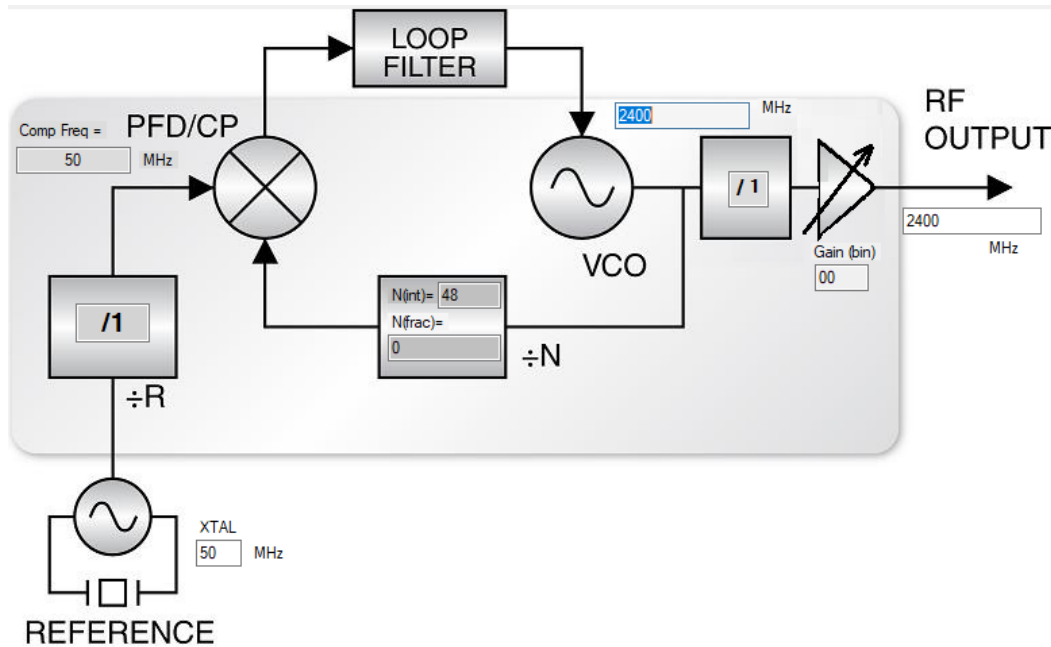


Figure 29: Synthesizer Block Diagram

5.2 Signal Characterization - Synthesizer

The 8 different options for Reg file configuration are tested experimentally as to determine the optimal output signal for a jammer system. The HMC1035LP6GE synthesizer datasheet provides a graphic for visualizing the output waveforms of performance priority and power priority in LVDS and LVPECL modes, as shown below in the figure [28]. As stated previously, the LVPECL output voltage swing is larger in magnitude relative to LVDS. Additionally, the rising and falling edges of the performance priority waveforms can be observed to be much sharper than that of the power priority waveforms; the symmetry of the rising and falling edges is critical for reduction of jitter [29]. The datasheet does not provide any graphs for visualizing the difference in voltage swing of fractional vs. integer modes.

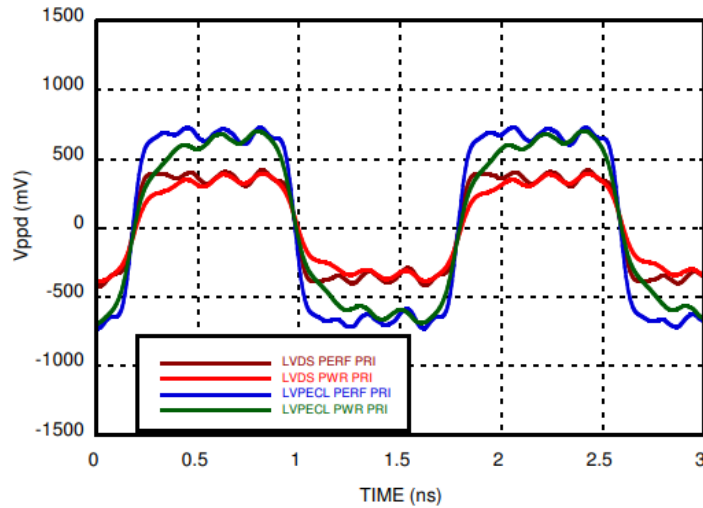
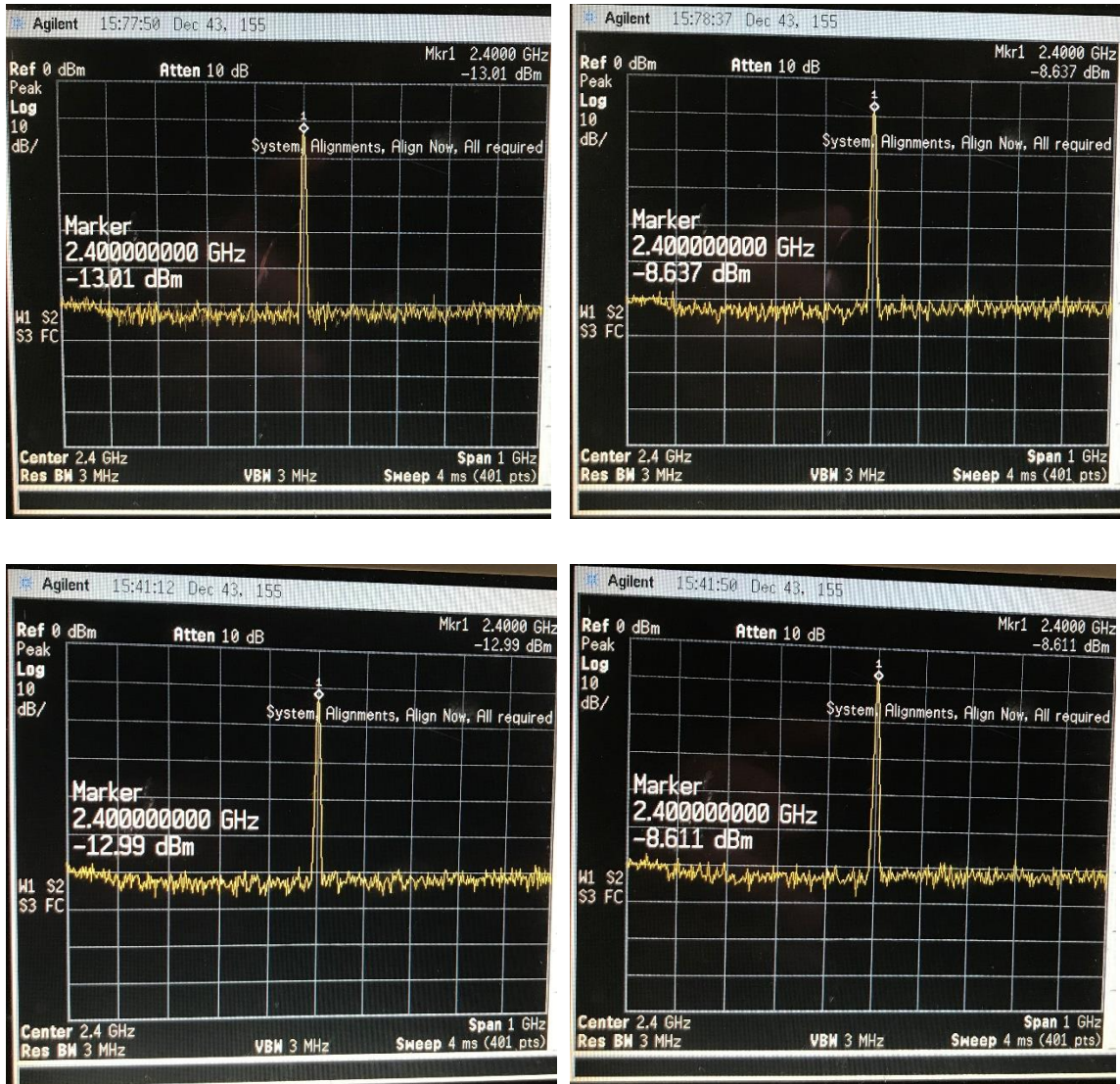


Figure 30: Output Waveform, Performance vs. Power Priority for LVPECL/LVDS [28]

The 8 different Reg files are loaded to the synthesizer board to evaluate their output waveforms. For each of the Reg files loaded, the output frequency configuration of the synthesizer is 2.4GHz in the user interface. The synthesizer displays “locked” as in Figure 28 when it locks onto the desired frequency. The figure below depicts the LVDS waveforms as measured by a spectrum analyzer; the spectrum analyzer is set up to have a span of 1GHz, with the center frequency set to 2.4GHz. A male to male SMA connector connects one of the differential outputs (N-channel) of the synthesizer to the spectrum analyzer.

The top two screen captures display the output signal for the fractional mode while the bottom two are for integer mode. The synthesizer configuration for the left two screen captures is the power priority mode while

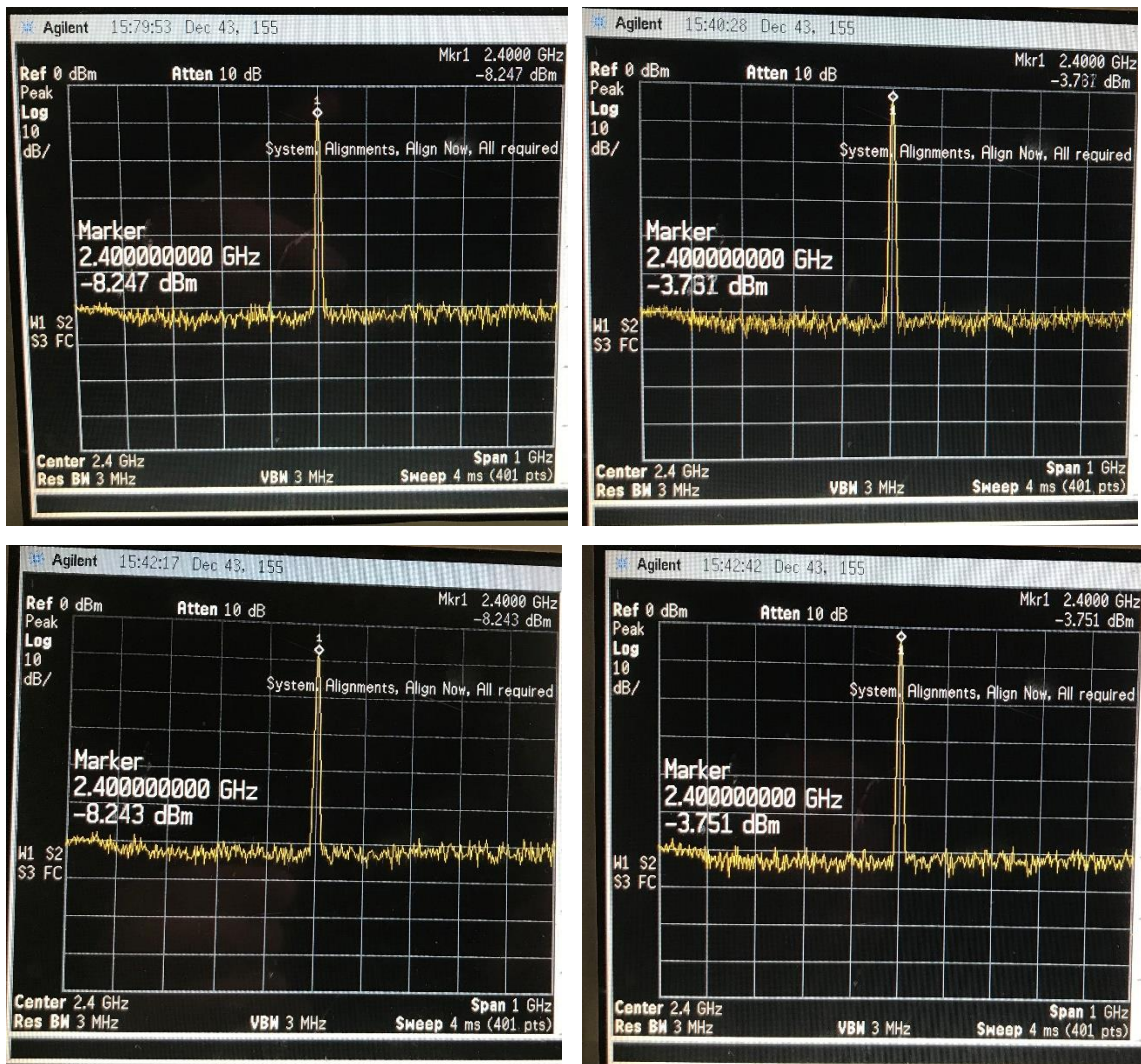
the right two are for performance priority. The “Peak Search” function finds the highest peak in the spectrum displayed by the spectrum analyzer and displays the center frequency and power of the peak, as in each of the 4 captures. As expected, the performance priority tones are higher in power relative to the power priority tones. By comparing ‘a) and c)’ and ‘b) and d)’, the output power of the fractional and integer modes was experimentally determined to be similar in magnitude. The peak output power of the 4 screen captures was found to be -8.611dBm.



$\frac{a,b}{c,d}$ Figure 31: Synthesized LVDS Signals

- a) 2.4GHz LVDS Signal, Fractional Mode + Power Priority b) 2.4GHz LVDS Signal, Fractional Mode + Performance Priority c) 2.4GHz LVDS Signal, Integer Mode + Power Priority d) 2.4GHz LVDS Signal, Integer Mode + Performance Priority

Next, the LVPECL waveforms are characterized, keeping the same configuration for the spectrum analyzer. The figure below depicts the LVPECL tones; the screen captures are laid out in a similar manner to that of the previous figure. The “Peak Search” function of the spectrum analyzer was utilized once again. Just as in the LVDS waveforms, the performance priority tones are higher in power relative to the power priority tones. By comparing ‘a) and c)’ and ‘b) and d)’, the output power of the fractional and integer modes was experimentally determined to be similar in magnitude once again. The peak output power of the 4 screen captures is -3.737dBm.



$\frac{a}{c}$ $\frac{b}{d}$ Figure 32: Synthesized LVPECL Signals

- a) 2.4GHz LVPECL Signal, Fractional Mode + Power Priority
- b) 2.4GHz LVPECL Signal, Fractional Mode + Performance Priority
- c) 2.4GHz LVPECL Signal, Integer Mode + Power Priority
- d) 2.4GHz LVPECL Signal, Integer Mode + Performance Priority

Following the characterization of the LVPECL and LVDS waveforms, the optimal Reg file configuration was determined to be LVPECL, fractional mode + performance priority. The LVPECL signaling scheme is higher in output power relative to the LVDS scheme. The performance priority mode is more advantageous for the purpose of jamming for its higher output power relative to the power priority mode; a higher output power for the synthesizer entails a reduction of the gain requirements for power amplification. The output power difference between fractional and integer modes is negligible. The functionality of the fractional mode, where the N divider can be broken down into a fraction M/N, provides better design flexibility relative to the integer mode.

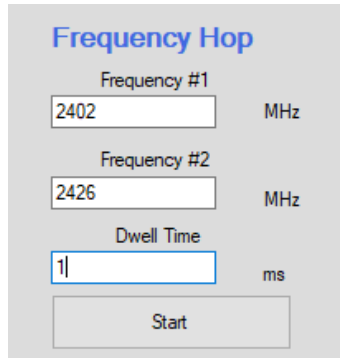
5.3 Frequency Hopping – Synthesizer

This section details the capability of the synthesizer evaluation board to hop user-defined frequencies, with custom delays. Utilization of a single synthesizer, hopping the three primary advertising channel frequencies, allows the circumvention of purchasing and programming three individual synthesizers. Additionally, the circumvention of a power combining circuit is possible. Several methods for implementing frequency hopping are tested and are discussed in this section. Frequency hopping and its feasibility are determined experimentally. In the first approach, the Hittite software's "Frequency Hop" functionality was tested. In the second approach, Reg files are configured independently for each of the three advertising channel frequencies and are concatenated into a single Reg file. A custom Matlab function enables the automation of the process for adding delays between each frequency step. In the third approach, the Hittite software's "Scan VCO Frequency" and "Show R/W Regs History" functionalities are tested to reverse engineer a single Reg file for frequency hopping. Once again, a custom Matlab function enables the automation of the process for adding delays to the hop sequence.

5.3.1 Frequency Hopping – Approach #1

In the first approach, the Hittite software's "Frequency Hop" functionality was tested to determine its feasibility in hopping the three primary advertising channel frequencies. The figure below displays the window for configuring the frequency hop. The hopping functionality limits the number of frequencies of

the hop to two. The dwell time is the time that the synthesized frequency stays locked and is adjustable in integer increments of 1ms.



The screenshot shows a configuration window titled "Frequency Hop". It features three input fields for frequency and dwell time settings. The first field, labeled "Frequency #1", contains the value "2402" and is followed by the unit "MHz". The second field, labeled "Frequency #2", contains the value "2426" and is followed by the unit "MHz". The third field, labeled "Dwell Time", contains the value "1" and is followed by the unit "ms". Below these fields is a "Start" button.

Figure 33: Frequency Hop Configuration

The Hittite software does not allow two instances of the Hittite program to be open simultaneously, for a single USB connection of the synthesizer evaluation board. If two instances of the program could be controlled simultaneously, there was a possibility the synthesizer could hop the three frequencies. One of the instances could hop advertising channels 37 and 38 while the second instance could periodically transmit on channel 39. But since this is not possible, the frequency hop functionality of the program is incapable of efficiently hopping the three advertising channel frequencies.

5.3.2 Frequency Hopping – Approach #2

In the second approach, Reg files are configured independently for each of the three advertising channel frequencies. Programming of the evaluation board is accomplished by uploading a Reg file through the GUI. Utilization of the GUI makes it difficult to get full control of the synthesizer hardware; the Reg file commands, included in Appendix, are in a “REG # Hex_value” format so there are fundamental limitations to configuring the synthesizer. The user interface does not allow the user to interpret how the Reg file commands are compiled and processed by the synthesizer IC; this makes it difficult to understand how the Reg files are generated in the first place and how to modify them. For this reason, successful programming of the evaluation board required some “trial and error” analysis. An example of this analysis would be observing the impact of the programming by monitoring the VCO tune port of the synthesizer IC.

The raw LVPECL fractional + performance priority Reg file is uploaded to the evaluation board through the GUI. The synthesizer is configurable to output frequencies corresponding to the frequency of the advertising channel via the “OUT Freq Desired” field. After doing so, the “Save Reg File” option allows the user to save the LVPECL fractional + performance priority Reg file with the output frequency configured to that of the advertising channel. These three steps are labeled on the GUI as shown below in the figure below. A preview of the saved Reg file displays in the “Register File Display” window on the right side of the GUI. Three Reg files are created with each configured to output frequencies corresponding to that of the three primary advertising channels (2.402GHz, 2.426GHz, & 2.480GHz). These Reg files are concatenated to create a single larger Reg file, to simulate the process of hopping the frequencies. A custom Matlab function enables the automation of the process for adding delays between each frequency step.

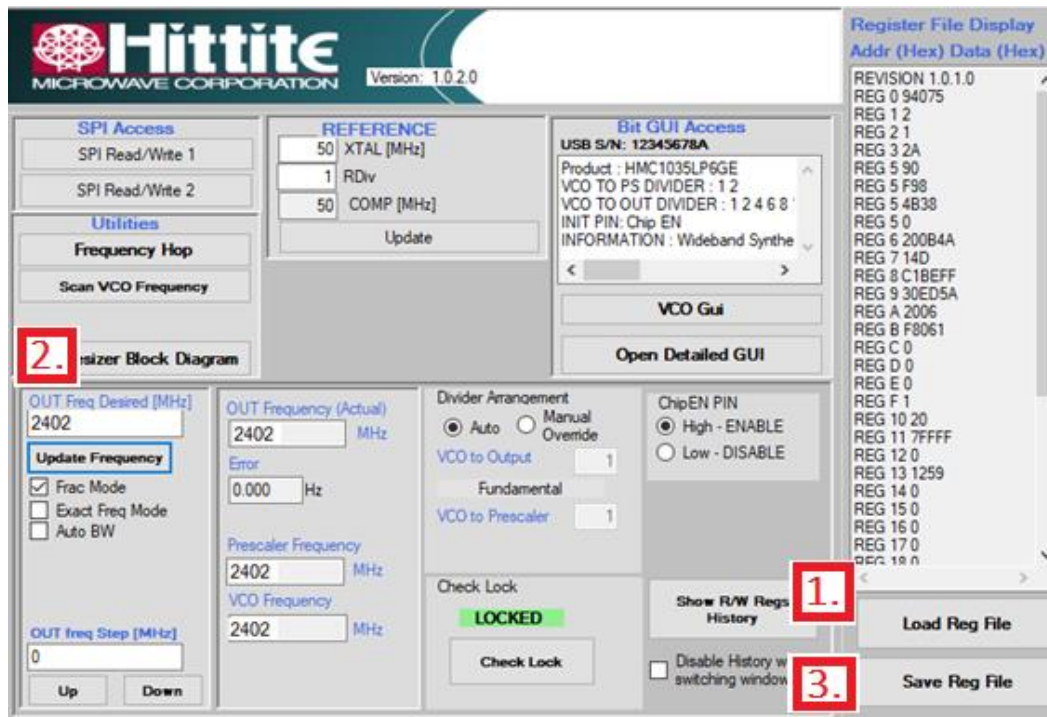


Figure 34: Annotated Hittite User Interface

As depicted in the figure, the Reg files contain commands for how to configure the various registers of the synthesizer IC. The values set are in hexadecimal format. In order to insert “delays” into the concatenated Reg file, the HMC1035 synthesizer datasheet was studied. In each of the three Reg files, the synthesizer sets the ‘Reg 0Ch Exact Frequency Mode Register’ to a value of 0. Setting this register value to 0, is essentially

a “do nothing” command. For this reason, setting of this register to a value of 0 is a feasible option for inserting delays in between the frequency steps.

At this time, the time necessary to execute the “do nothing” command is unknown. Ten lines of the “do nothing” command were inserted in the concatenated Reg file, between each of the three frequencies to be hopped. Other than looking at the “Check Lock” section of the Hittite GUI, there is no way of knowing whether or not the synthesizer locks onto the three frequencies. A workaround for confirming the frequency hop timing is necessary. The VCO tuning pin/trace of the evaluation board is a feasible option for visualizing frequency hopping in the time domain. Pin 23 of the figure below is the “Vtune” pin of the synthesizer IC. The designers of the board left ‘R30’ and ‘A1’ depopulated as to allow the user to easily probe the tuning pin; a 0ohm resistor soldered on the ‘R30’ pads and a wire soldered to the ‘A1’ pad allows the probing of the tune pin with an oscilloscope. By monitoring the VCO tuning port via an oscilloscope, frequency hopping can be verified without using GHz measurement equipment.

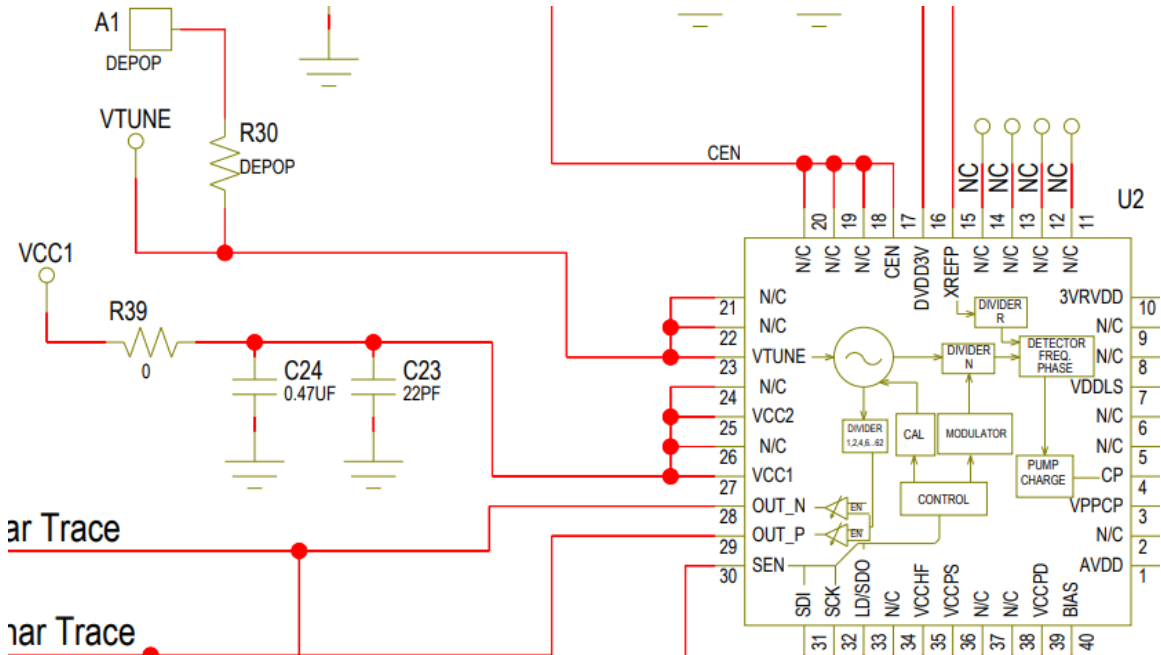


Figure 35: Eval01-HMC1035PG Board Schematic Snip [30]

The concatenated Reg file, without delays inserted, was loaded to the synthesizer evaluation board. Probing of the VCO tune pin enabled observation of the frequency hop on the oscilloscope; the voltage values indicate the different synthesized frequencies. The figure below depicts the VCO tune pin as observed on the oscilloscope. Frequency hopping via the concatenated Reg file was unsuccessful. The VCO tune pin voltage

starts at a DC voltage of approximately 1.7V as the synthesizer is initially configured to output 2.4GHz (depicted by region 1 in figure). Up to the end of region 3 of the figure, the synthesizer was correctly hopping from 2.402GHz to 2.426GHz. Region 2 depicts the voltage sustained to output 2.402GHz while region 3 depicts the voltage sustained to output 2.426GHz; the two voltages are slightly different in magnitude. The transition between regions 2 and 3 depicts the time needed by the synthesizer to lock onto a new frequency. Region 2.5 spans approximately 80ms. A similar transition is apparent in between regions 3 and 4. If the frequency step from 2.426GHz to 2.480GHz were to have been successful, the voltage sustained in region 4 would have been lower and closer in magnitude to regions 2 and 3; the tuning voltage for all three frequencies should be relatively close as these voltages control the VCO. A large difference in tuning voltage would result in a large difference in the frequencies synthesized.

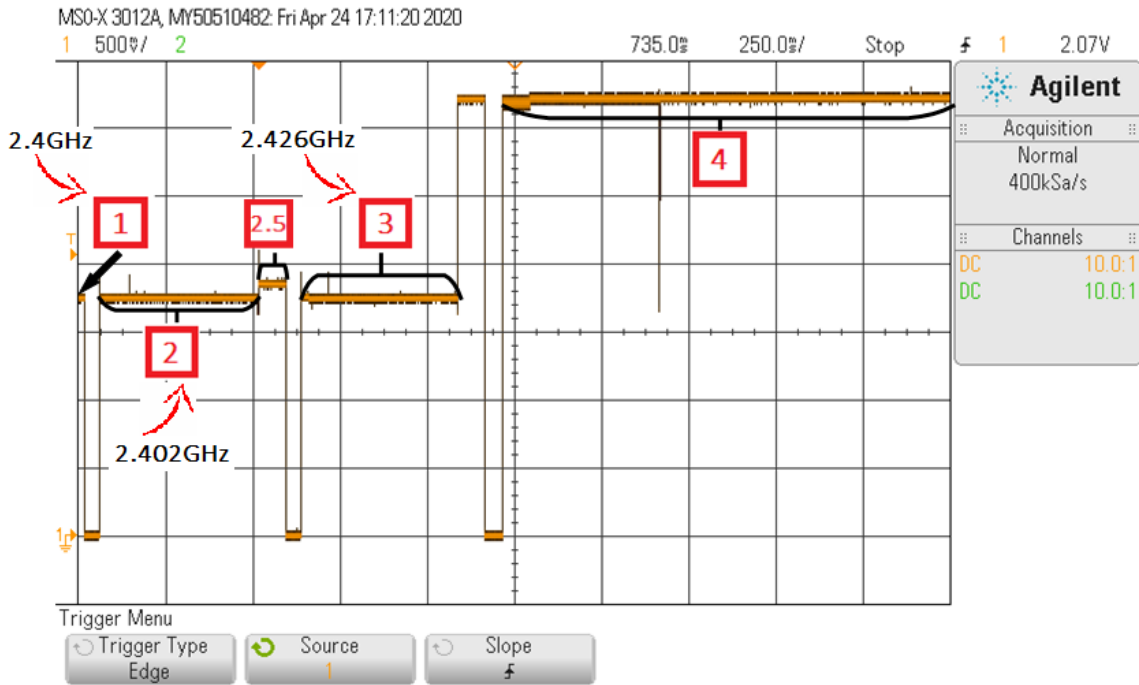


Figure 36: Oscilloscope Capture of VCO Tune Pin – Approach #2

Although the frequency hopping was unsuccessful, the timing for delay code execution was able to be determined as the hop from 2.402GHz to 2.426GHz was successful. The concatenated Reg file, with 10 lines of delays inserted, was loaded to the synthesizer evaluation board. The figure below depicts the VCO tune pin as observed on the oscilloscope. Region 2.5 spans 80ms and is independent of the delay time. Region D's span is the time difference between the cursors, $\Delta X = 240\text{ms}$, minus the time measured for region 2.5; region

D is calculated to be 160ms (240ms – 80ms = 160ms). The delay between the frequency steps was 10 lines and took 160ms to execute, leading to the conclusion that each delay line takes approximately 16ms to execute. Delay code twice in length resulted in the same conclusion for execution time.

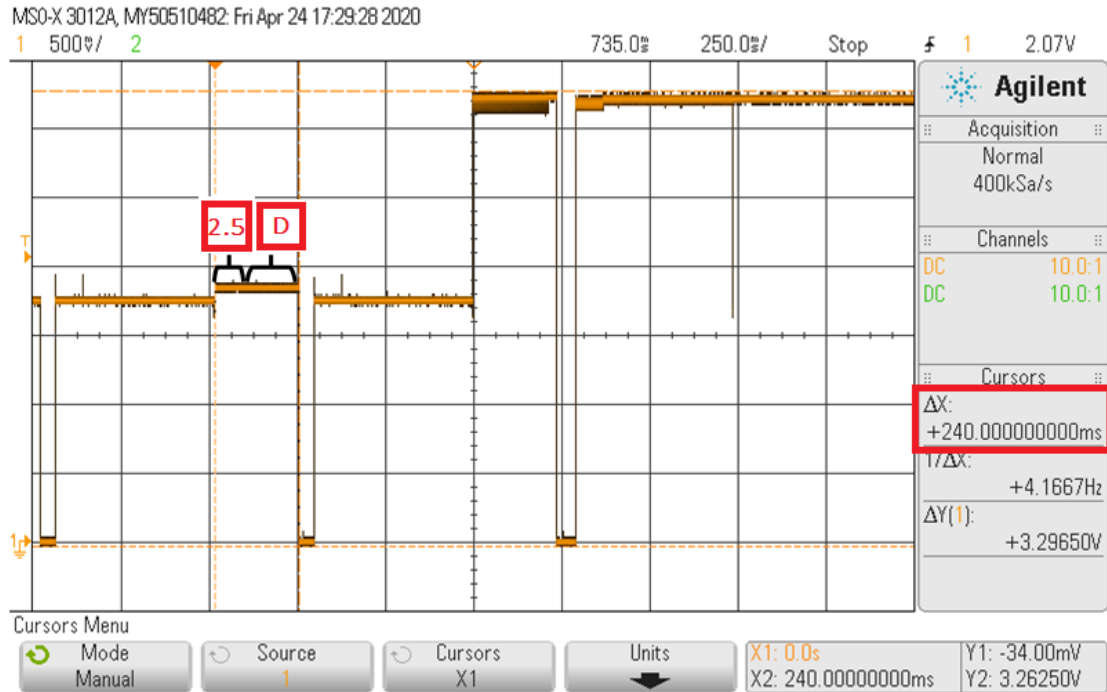


Figure 37: Oscilloscope Capture of VCO Tune Pin to Measure Delay – Approach #2

5.3.3 Frequency Hopping – Approach #3

In the third approach, the Hittite software’s “Scan VCO Frequency” and “Show R/W Regs History” functionalities are utilized to reverse engineer a single Reg file for frequency hopping. The figure below displays the window for configuring the VCO frequency scan. The scanning functionality allows the user to define start frequency, max frequency, and steps from the start frequency to max frequency in increments defined by the “Coarse Step” field. The start frequency is the lowest of the three primary advertising channels, 2.402GHz. The stop frequency is the highest of the three advertising channels, 2.480GHz. The “Coarse Step” is 6MHz, which is the great common divisor (GCD) of the frequency differences between Ch. 37 and 38 (2426 – 2402 = 24MHz) and between Ch. 39 and 37 (2480 – 2402 = 78MHz); GCD(78M, 24M) = 6MHz. With 6MHz steps, the scan will include 11 unwanted frequencies.

Figure 38: VCO Frequency Scan Configuration

Once the fields are populated, the scan initiates by pressing “Start Scan” button. In order to create a new Reg file for frequency hopping, the Reg file configuration for the VCO scan is saved. Utilization of the “Show R/W Regs History” functionality is necessary for this task. The “Save History in Reg File Format” option allows the user to save the various commands sent to the synthesizer. By saving the history of the scanning process, a Reg file for frequency hopping from 2.402GHz to 2.480GHz in 6MHz steps is obtained.

```

4:53:00 PM --> R &H6 &H0 readbit Read
4:53:00 PM --> R &H2 &H0 readlng Read
4:53:00 PM --> R &H4 &H0 REGSET_87797 get_frac_register
4:53:00 PM --> R &H3 &H0 REGSET_87797 get_intg_register
4:53:00 PM --> R &HC &H0 readlng Read
4:53:00 PM --> R &H0 &H0 Read CHIP_ID from the chip
5:06:14 PM --> R &H6 &H0 readbit Read
5:06:14 PM --> R &H6 &H0 readbit Read
5:06:14 PM --> R &H2 &H0 readlng Read
5:06:14 PM --> R &H4 &H0 REGSET_87797 get_frac_register
5:06:14 PM --> R &H3 &H0 REGSET_87797 get_intg_register
5:06:14 PM --> R &HC &H0 readlng Read
5:06:14 PM --> R &H0 &H0 Read CHIP_ID from the chip
5:06:46 PM --> R &H6 &H0 readbit Read
5:06:46 PM --> R &H6 &H0 readbit Read
5:06:46 PM --> R &H2 &H0 readlng Read
5:06:46 PM --> R &H4 &H0 REGSET_87797 get_frac_register
5:06:46 PM --> R &H3 &H0 REGSET_87797 get_intg_register
5:06:46 PM --> R &HC &H0 readlng Read
5:06:46 PM --> R &H0 &H0 Read CHIP_ID from the chip

```

Figure 39: R/W Regs History Option of Hittite User Interface

Once the Reg file is saved, it is manually modified to contain only the frequencies corresponding to the advertising channels. This new Reg file is saved and loaded to the evaluation board. The figure below depicts the oscilloscope capture of the VCO tuning pin as the three frequencies are hopped. By monitoring the VCO

tuning port via an oscilloscope, frequency hopping can be verified without using GHz measurement equipment.

The VCO tune pin voltage starts at a DC voltage of approximately 1.7V as the synthesizer is initially configured to output 2.4GHz (depicted by region 1 in figure). Region 2 depicts the voltage sustained to output 2.402GHz while region 3 depicts the voltage sustained to output 2.426GHz. Notably, the transition between regions 2 and 3 is not the same as in approach #2, where the VCO tune voltage rose in magnitude momentarily before dropping to ground. This is due to the difference in the configuration of the hop Reg file. The transition between regions 2 and 3 spans approximately 10ms. A similar transition can be seen between regions 3 and 4, which is the transition from 2.426GHz to 2.480GHz. The VCO tune pin voltage remains constant for the duration of region 4, where the synthesizer outputs 2.480GHz. Interestingly, the tune pin voltage transitions in region 4.5 and returns to a steady state value in region 5. This was not expected since the Reg file was configured to output 2.480GHz indefinitely once the hop reached that frequency step. Nevertheless, in region 5 the VCO tune pin voltage comes back to a similar magnitude as region 4. The time span from the end point of region 1 (when the hop starts) to the starting point of region 5 (when the hop ends) is 468ms. Regions 2, 3, and 4 are measured to be 130ms, 122ms, and 106ms, respectively.

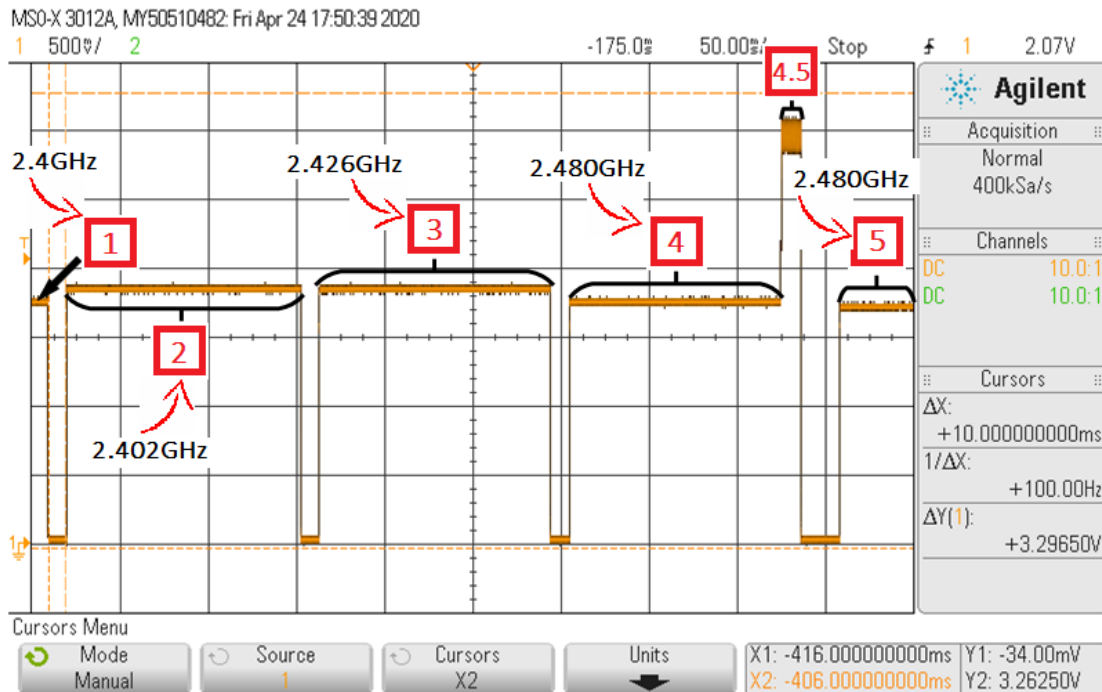


Figure 40: Oscilloscope Capture of VCO Tune Pin – Approach #3

Following the confirmation of the functionality of the hop Reg file, a custom Matlab function is written, once again, to automate the process of adding delays to the hop sequence. Additionally, this Matlab function allows the user to repeat the hop any number of times.

5.4 Frequency Hopping Jammer System Overview

This section provides an overview of the changes made to the initial proposed jammer system due to the COVID-19 pandemic; as stated previously, modifications were made to the preliminary high-level block diagram due to lack of access to laboratory equipment and supply chain issues. The figure below depicts the modified block diagram for the COVID-19 necessitated changes. The selection of a synthesizer evaluation board, as opposed to using individual synthesizer ICs, modified the system design as the design of a PCB was less critical to prove the objective of the thesis. The decision to not design a PCB was also a logistical one as design process for an RF PCB is time-consuming and the on-going COVID-19 pandemic has affected the accessibility of electronic test equipment and equipment utilized for the bring-up/manufacturing of the PCB; additionally, the COVID-19 pandemic has adversely affected the turn-around and shipping times of PCB manufacturing companies.

The modified block diagram of the figure below depicts the synthesizer evaluation board performing a frequency hop of the three primary advertising channel frequencies (2.402GHz, 2.426GHz, and 2.480GHz). The power amplifier is left out of the block diagram as its necessity is assessed in a later section. The frequency hopping approach is a feasible workaround to having three individual synthesizers for the generation of the three RF frequencies. It is important to note that by performing a frequency hop the average power of the received jamming signal, at the PHY layer, is one-third in magnitude relative to the proposed jammer system's (three individual synthesizers continuously transmitting). The frequency hopping workaround effectively eliminates the need for a power combining circuit. However, since the output signal of the synthesizer is differential, a circuit for converting the differential signals to a single ended signal might be necessary to make use of all RF power available from the synthesizer evaluation board. A balun is a feasible option for a differential to single ended converter in a 50 Ω environment. Baluns can be advantageous as they have excellent common-mode characteristics and the two differential ports can be well isolated from one another.

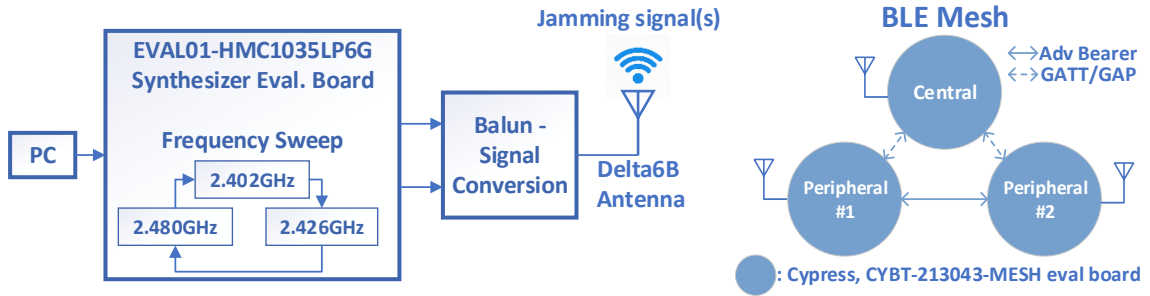


Figure 41: Modified Block Diagram – Frequency Hopping BLE Jammer (Part 1)

5.4.1 Balun Selection

A balun has the capability of combining two differential signals into a single-ended signal. An initial web search for a 3-port balun capable of differential to single-ended conversion, in a 50Ω environment, returned minimal results. However, an advanced search on Digikey returned some promising results. The ADC-WB-BB board by Texas Instruments (TI) is a feasible option for the desired signal conversion. The evaluation board uses the TC1-1-13MA+ balun from Mini-Circuits and has an impedance ratio of 1:1 and a frequency range of 4.5M-3GHz [31]. This board costs approximately \$65 after taxes. Due to thesis budget limitations (\$200), purchasing this balun board is unfeasible. Additionally, at the time this product was found there were none available in stock.

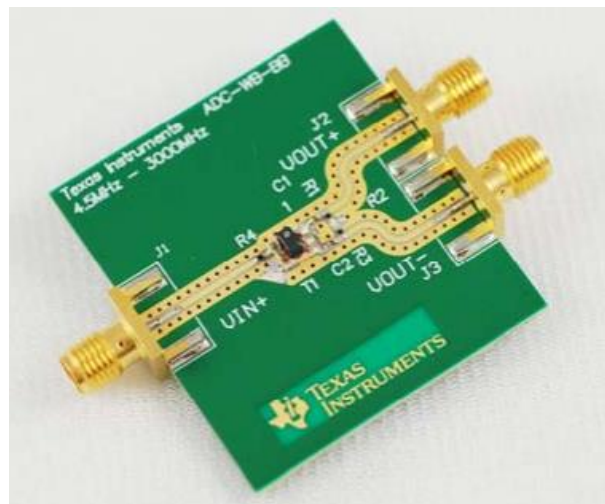


Figure 42: ADC-WB-BB Signal Conversion Evaluation Board [31]

Due to the lack of available TI balun boards, a different signal conversion evaluation board is necessary. An online search found a feasible 10M-3.0GHz balun. This board utilizes the ADF4350 chip by Analog Devices.

The designer of the board supplied a sample oscilloscope capture of the balun board converting a single-ended signal into differential signals as pictured in the figure below [32]. The blue waveform is the single-ended one while the purple and green waveforms represent the differential signals (180° out of phase). This board was ordered from China at a time when the expected shipping date was reasonable and the testing and use of this board would be integrable into the thesis report. However, due to the COVID-19 pandemic, the expected shipping time shifted significantly and the item was not received in time. For this reason, a modification to the “Modified Block Diagram – Frequency Hopping BLE Jammer (Part 1)” is necessary. These modifications are reflected in the “Part 1.1” block diagram of Figure 44.

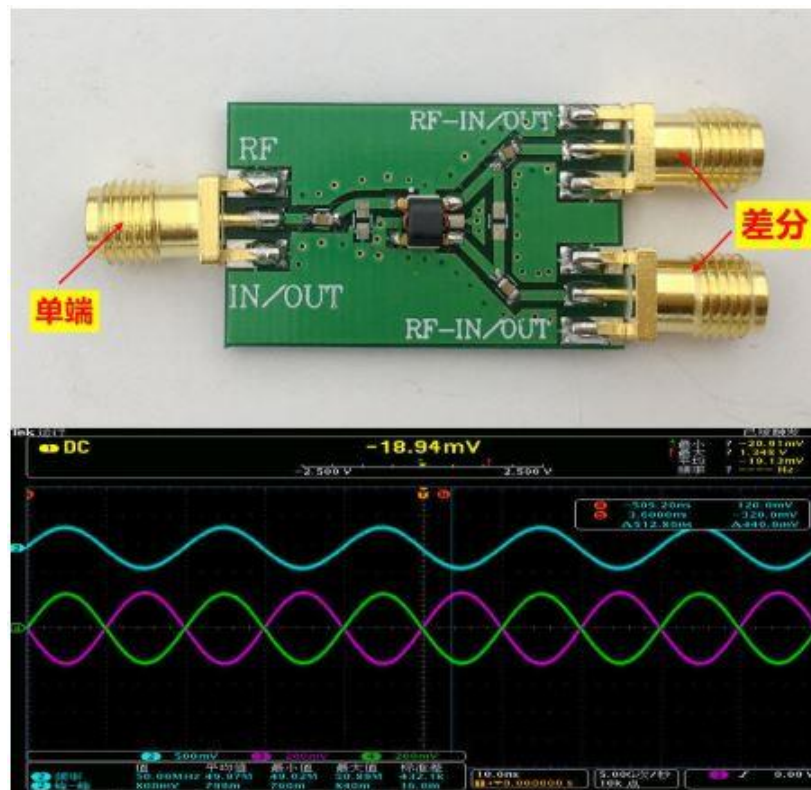


Figure 43: Balun Board Image and Oscilloscope Capture [32]

The figure below depicts the modified high-level block diagram “Part 1.1”. Modifications were made to the “Part 1” modified block diagram as the balun was not able to be integrated due to COVID-19 related supply chain issues and cost. The “Part 1.1” system also implements the frequency hopping approach and includes the following subsystems: a personal computer (PC), a synthesizer evaluation board, and an antenna. The PC will communicate with the synthesizer evaluation board and configure it to produce signals with center

frequencies corresponding to those of the BLE primary advertising channels. A single-ended output of the synthesizer (N-channel only) will be connected directly to the antenna which will then radiate the synthesized frequencies. The BLE mesh, depicted on the right, remains unchanged from the preliminary block diagram. This system is tested in the “Testing – Frequency Hopping Jammer” section of Chapter 6.

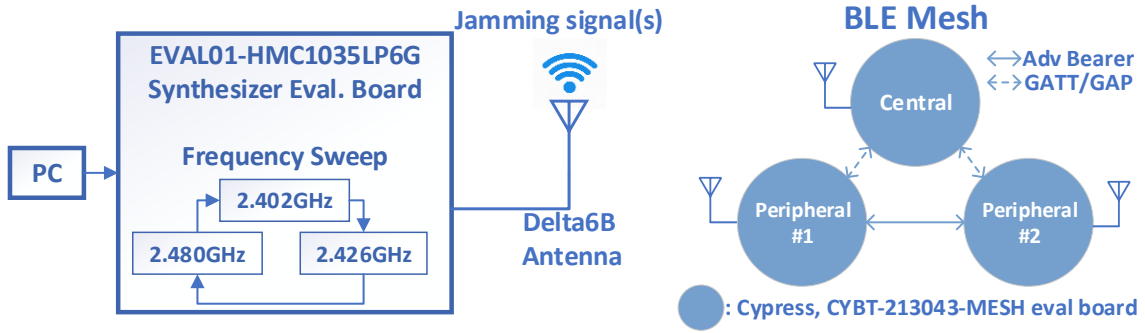


Figure 44: Modified Block Diagram – Frequency Hopping BLE Jammer (Part 1.1)

5.4.2 Power Amplifier Selection

The synthesizer’s output signal is characterized, by using the synthesizer evaluation board, in the “Signal Characterization – Synthesizer” section of Chapter 5. The synthesizer’s single-ended output (N-channel only) signal power is measured as approximately -3.7dBm, which is above the minimum transmitter power for BLE transmitters of power class 3, 2, and 1.5, but not power class 1. The output signal strength is also below the maximum output power for each of the four power classes. To clarify, BLE mesh device transmitters, of all power classes, can transmit signals of greater strength than that of the synthesizer output. For this reason, a power amplifier is essential for ensuring the effectiveness of the jamming signal; a power amplifier with gain of 23.7dB ($20\text{dBm} - (-3.7\text{dBm}) = 23.7\text{dB}$) or greater would be ideal as to ensure the jamming signal strength is of equal magnitude to the maximum transmitter power of any BLE transmitter. Notably, a BLE transceiver transmitting at 20dBm is most likely operating in an environment where the receiver node is very far away. Nordic Semiconductor technical program manager Jon Sponas states BLE devices have a maximum communication range of 400m [33]. If a power amplifier with gain ~24dB is unobtainable, the distance of the jammer system relative to the victim device could be minimized while the distance between the BLE mesh device communicating with the victim device could be held constant; by doing so, a power amplifier of slightly less gain can be equally feasible in proving the concept of jamming a BLE mesh.

Selection of a power amplifier relies on the following criteria: cost, SMA connections, frequency range as 2400MHz – 2500MHz, low noise figure, power gain, and DC supply voltage of ~5V (synthesizer evaluation board and BLE mesh evaluation boards operate on 5.5V and 5V, respectively). Analog Device’s EVAL-CN0417-EBZ 2.4GHz power amplifier evaluation board met the aforementioned criteria and has the following specs: power gain of 21dB, operating frequency of 2.4GHz, USB (5V) powering, noise figure of <5dB, P1dB of 30.8dBm at 2.140GHz (spec. of the ADL5606 power amplifier IC), cost of \$35, and SMA connections [34]. Figure 45 below depicts the EVAL-CN0417-EBZ PA evaluation board, which integrates an ADL5606 amplifier and LTM8045 inverting DC/DC converter [34]. It is important to note here that additional power gain may be necessary when performing the frequency hopping approach where the average power of the received jamming signal is a third of what it would have been had a continuous jamming approach been performed instead.



Figure 45: EVAL-CN0417-EBZ 2.4GHz Power Amplifier Evaluation Board [34]

5.5 System Requirements

The frequency hopping BLE jammer must produce 3 tones at center frequencies 2.402GHz, 2.426GHz, and 2.480GHz. The BLE network must communicate via the BLE protocol and be mesh compatible. The jamming signal must be transmitted with enough power for the devices of the mesh network to no longer be capable of establishing connections with one another. Often times, peripherals in BLE mesh networks periodically advertise; when these BLE devices are not advertising they are in a sleep mode and must re-establish their connection with the other devices in its mesh via advertising.

In the BLE protocol the maximum receiver sensitivity for the PHY layer channels is -70dBm [13]. The receiver sensitivity of the BLE channels are dependent on the particular PHY in use; the PHYs specified for

longer range communication have lower reference sensitivities. A jammer that is able to jam the maximum receiver sensitivity PHY will be capable of jamming the lower reference sensitivity PHYs as well. SIG specifies that a co-channel interference signal strength must be 21dB below that of the BLE channels communication. This means that if the interference signal strength is -91dBm (-70dBm – 21dB = -91dBm) or more when the received BLE signal strength is at the lowest detectable level, communication of the BLE channel can be corrupted. For this project, the jamming system attacks a BLE mesh network within the same room. Equation 5-1 is Friis power equation and it is suitable for estimating the power budget of a wireless link.

$$P_{RX} = D_{RX}D_{TX}\left(\frac{\lambda}{4\pi d}\right)^2P_{TX} \quad (5-1)$$

D_{RX} and D_{TX} represent the receiver and antenna gains in linear units. Antenna gains are typically in dBi units. λ represents the wavelength, d represents the distance between the receiver and transmitter, and P_{TX} represents the transmitter power in linear units. The gain of the transmitter antenna selected, Delta6B, is 5dBi or ~3.162 in linear units while the gain of the receiver, the Cypress mesh board, is -0.5dBi or ~0.891 in linear units [22]. As mentioned previously, the synthesizer's single-ended output power is -3.7dBm. With the 21dB gain PA, the jamming signal is now 17.3dBm (-3.7dBm + 21dB = 17.3dBm) or ~50mW in linear units. The wavelength of the system is 0.125m ($\lambda = c/f = 3e8/2.4e9 = 0.125m$). Friis equation can now be used to solve for the received power at the device to-be-jammed, as a function of the distance, as shown in equation 5-2.

$$P_{RX_SYNTH} = 0.891 * 3.162 * \left(\frac{0.125}{4\pi d}\right)^2 * 50m = 1.394*10^{-5}/d^2 \text{ W} \quad (5-2)$$

Equation 5-2 is plotted to visualize the received power with respect to distance. The minimum interference signal strength to corrupt BLE communication is -91dBm or ~0.794pW in linear units. Based off Friis equation it appears as if the jamming signal (interference) is capable of successfully corrupting BLE transmission of BLE devices >400 meters away for the synthesizer-based jammer; the interference signal strength never reaches 0.794pW in Figure 46. Notably, the y-axis for Figure 46 scaling is 1×10^{-9} .

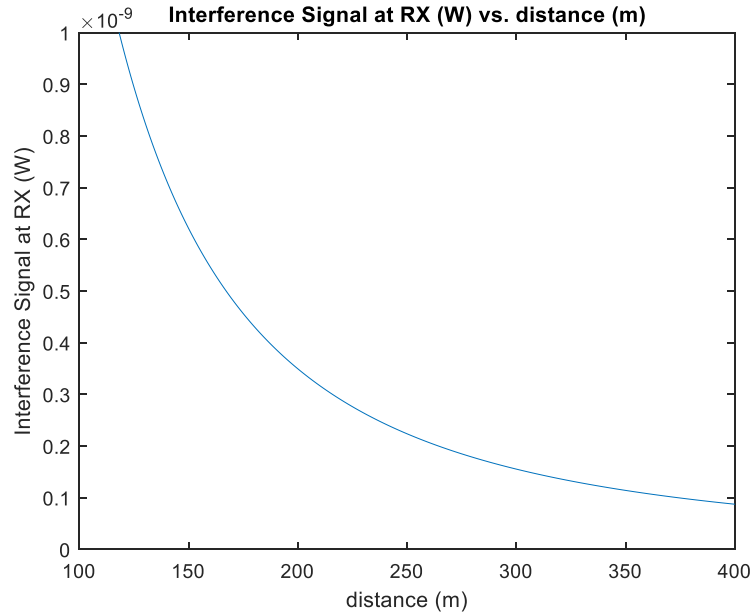


Figure 46: Finding Max Distance of Jamming Signal Reception (Synthesizer + PA + Delta6B)

Friis formula is a theoretical maximum, which assumes perfect link connection. In reality, other factors such as margin, multipath, fading, atmospheric interference, etc... must be considered. For these reasons, an RF range calculator by Silicon labs was used to calculate the range of the jamming signal with higher accuracy [35]. The inputs to the range calculator are the TX output power, TX antenna gain, receiver sensitivity, RX antenna gain, and operating frequency. These inputs are set in the range calculator, for the synthesizer (w/ PA) jammer, and the results are shown below in Figure 47.


Frequency Band [MHz] <input type="text" value="2440"/>		 SILICON LABS	
<u>TX parameters</u> Enter TX Output Power [dBm] <input type="text" value="17"/>		<u>RX parameters</u> Enter RX Sensitivity [dBm] <input type="text" value="-70"/>	
Enter TX Antenna Gain [dBi] <input type="text" value="5"/>		Enter RX Antenna Gain [dBi] <input type="text" value="-0.5"/>	
Typical Open Field Outdoor Range [m]		209.6	
Typical Indoor, Small Office Range [m]		36	

Figure 47: RF Range Calculator by Silicon Labs (Synthesizer + PA + Delta6B)

The results of the range calculator state that in a typical outdoor setting the synthesizer-based jamming signal can successfully interfere with BLE communication at a distance of 209.6m. For a typical indoor setting, the range is computed as 36m. From these range calculator results, it can be concluded that the jamming signals can successfully jam a BLE receiver in a typical small office environment, which is the testing environment proposed for this project.

Chapter 6

SYSTEM TESTING & CHARACTERIZATION – FREQUENCY HOPPING JAMMER

6.1 Overview

A constant jammer, such as the one proposed in this paper, is typically not battery powered due to its continuous transmission of signal(s). For this reason, the performance/efficiency criteria is not as quantifiable relative to stealthier jammers. The overall goal of the proposed jammer is to completely disrupt communications, a.k.a. denial of service, on the advertising channels of BLE. The strength against PHY layer techniques is the third criteria used for characterization of a jammer. The primary advertising channels of BLE do not implement any frequency hopping techniques. Jamming a BLE mesh utilizing solely the primary advertising channels does not require any consideration of the PHY layer techniques for this reason. However, if a BLE mesh utilizes the secondary advertising channels, the specifications of the jammer become more complex as the secondary channels implement FHSS. The fourth criteria for characterization is probability of detection. Low probability of detection is ideal for a jammer of any flavor. The proposed jammer focuses on the proof of concept of jamming BLE's primary advertising channels and is not optimized for low probability of detection. If the jammer's probability of detection were to be improved upon in the future, the jammer could adopt techniques that are consistent with MAC layer behaviors [7].

After characterizing each of the components of the jammer, the frequency hopping jammer system was integrated. Due to the nature of the COVID-19 pandemic, the shipment and reception of the balun signal conversion PCB was delayed; the testing and integration of the balun PCB was not implemented for this reason. Since the balun was not able to be integrated into the design the jammer system shifted. The jammer now consists of a synthesizer evaluation board, power amplifier, and antenna. Without the balun, utilization of both of the differential outputs of the synthesizer was not possible. The N-channel of the synthesizer's differential output was utilized for this reason. The figure below depicts the jammer system. The USB connection to the synthesizer board goes to the PC for configuration of the signals to synthesize.

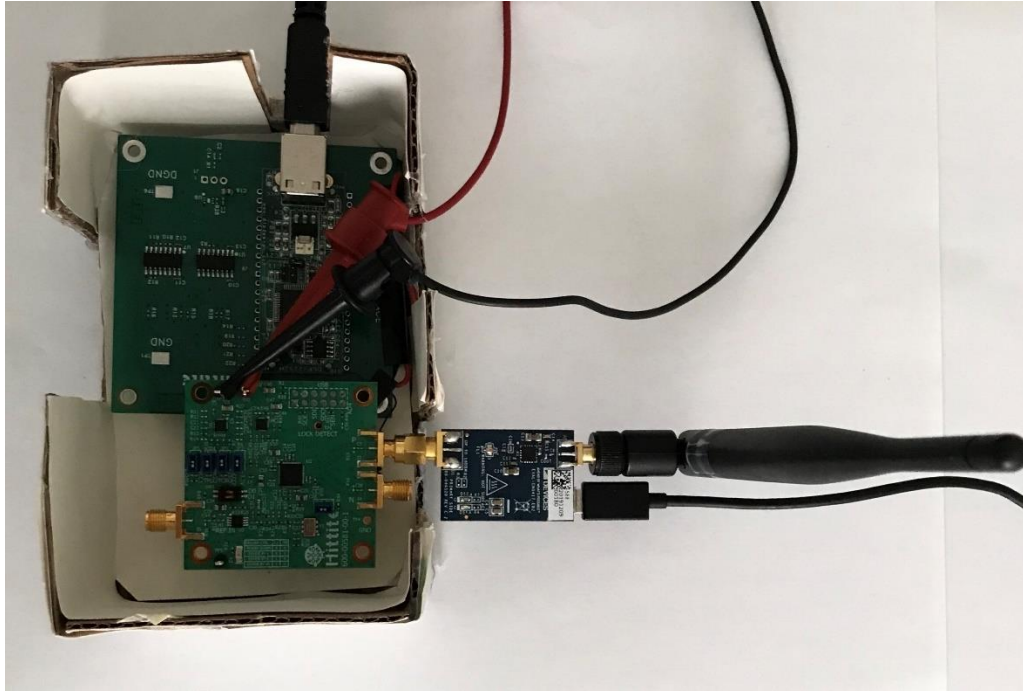


Figure 48: Synthesizer Jammer, Top View (Approach #1)

From this point forward, the “synthesizer jammer” refers to the system consisting of the synthesizer, power amplifier, and antenna. For the first phase of testing the jammer, the most optimal frequency hopping approach (#3) will be tested. This phase of testing will be referred to as approach #1 from here on.

6.2 Testing Environment

For testing of the jammer system, an anechoic chamber would be ideal. Due to the COVID-19 pandemic, the laboratories on campus at Cal Poly are closed to students. For this reason, the construction of an at-home testing environment is necessary. The figure below depicts the home testing environment. The room is 12x13 feet and a Wi-Fi network exists in the environment. Additionally, the test environment is a room in a student housing complex so additional Wi-Fi networks exist in its vicinity. The desk in the center top has a power supply and oscilloscope set up while the desk at the bottom left has a power supply and spectrum analyzer set up. The 10ft length desk is located strategically in between the two power supplies; the synthesizer evaluation board requires a 5.5V supply voltage and banana-to-grabber leads cannot traverse 10ft so the utilization of two power supplies is necessary. The 10ft desk has a 10ft long strip of tape with 0.25m

increments of distance marked for testing purposes. A Delta6B antenna connects to the spectrum analyzer strategically as to line up the location of the antenna and the “0m” mark of the 3m desk.

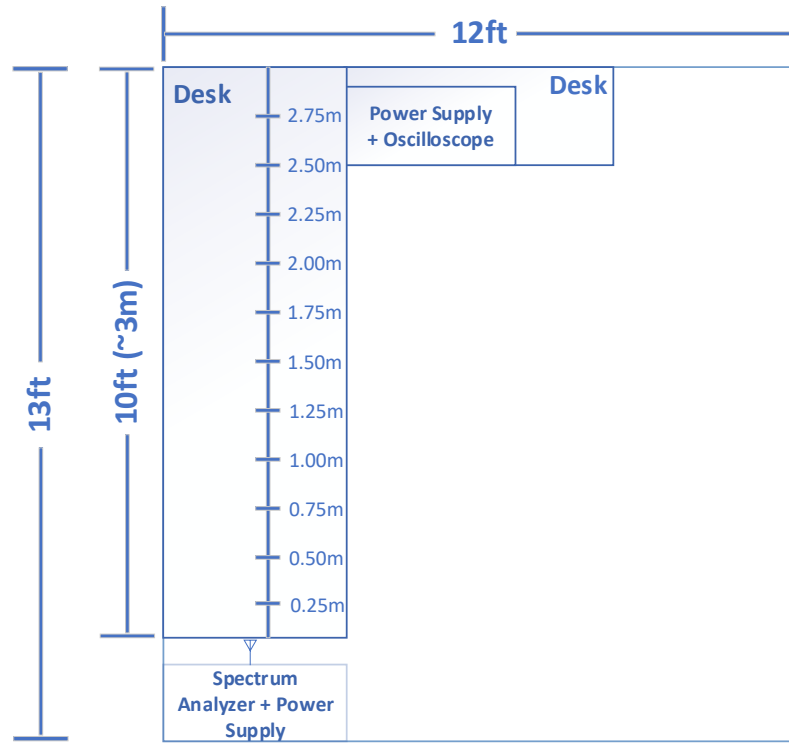


Figure 49: Home Testing Environment Set up

6.3 Testing – Frequency Hopping Jammer

The jammer tested in this section is the synthesizer jammer consisting of the synthesizer evaluation board, power amplifier, and antenna as depicted earlier in Figure 44. The figure does not include a power amplifier; its necessity assessed in the “Additional Exploration” section of Chapter 7. The jamming technique utilized for this testing is the frequency hopping discussed in the “Frequency Hopping – Approach #3” section of Chapter 5. Frequency hopping was implemented by reverse engineering a single Reg file using the Hittite software’s “Scan VCO Frequency” and “Show R/W Regs History” functionalities. This technique is customizable by adjusting the contents of the Reg file.

As previously discussed, the frequency hop includes minimal delays between each of the synthesized primary advertising channel frequencies. The approach results in the following timing: duration of advertising on Ch.37 (2.402GHz) ~130ms, on Ch.38 (2.426GHz) ~122ms, Ch.39 (2.480GHz) ~106ms, and ~10ms delays

in between each of the synthesized frequencies. The ~10ms delay is a property of the synthesizer PLL itself and cannot be changed.

Prior to testing the frequency hopping jamming technique, the synthesizer jammer itself is characterized for power. For this test, the synthesizer continuously outputs 2.402GHz. The synthesizer jammer's output signal power is measured in increments of 0.25m from 0m up until 3m. The spectrum analyzer equipped with an antenna, located at 0m, measures the received signal power in units dBm. The figure below includes a plot of received signal power vs. distance. At a distance of 0m, the jamming signal is 8.85dBm. At a distance of 3m the jamming signal is -16.88dBm. For all distances, the jamming signal level, at the receiver, is greater than the reference sensitivity of the BLE PHY layer. The measurements become more susceptible to interference as the jammer's distance from the spectrum analyzer increases, due to the testing environment limitations described previously. Notably, the data is acquired using a continuous emission of a single frequency. In the hopping approach, the average power received by the PHY layer advertising channels is one-third (~4.8dB less) in magnitude.

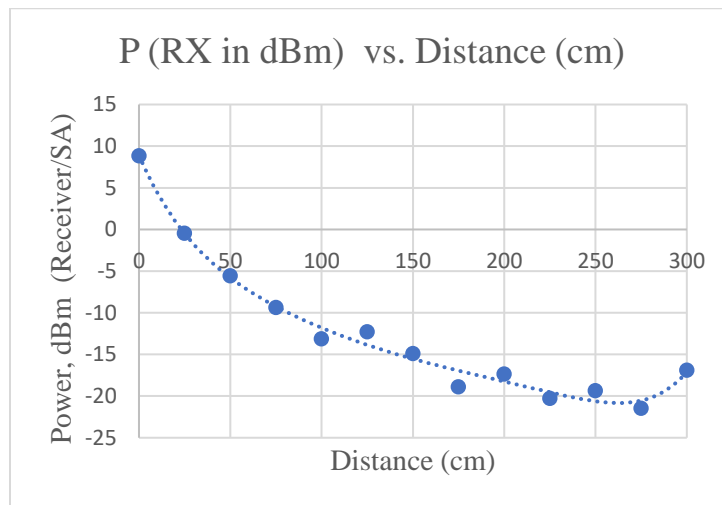


Figure 50: Measured Signal Strength (Synthesizer Jammer) at Receiver/SA vs. Distance

Following the characterization of the jamming signal strength, the frequency hopping jamming technique is tested. The following constraints are considered when creating a test set-up for the frequency hopping jamming approach: placement of victim devices, on/off state of jammer, and on/off state of BLE mesh devices. The figure below compliments the testing process. In this figure, Device #3's location is at the second from furthest location tested. The spectrum analyzer's antenna and Mesh Device #2's antennas are

located at the 0m mark. By placing Mesh Device #2's antenna at the 0m mark, the jamming signal strength at the PHY level can be measured by the spectrum analyzer. The testing process goes as follows:

0. Turn off all mesh devices and jammer
1. Turn on jammer
 - a. Jammer is located at 0.5m mark
 - b. Jammer is frequency hopping the three primary advertising channels indefinitely
2. Turn on Mesh Device #2 (LED)
 - a. Mesh device #2 is placed next to spectrum analyzer's antenna, at 0m mark, so jamming signal strength received by this device can be quantified
3. Place Mesh Device #3 (LED dimmer) at 0.75m mark (for first iteration of step 3 only) and turn on
4. Test functionality of communication between Mesh Devices #2 & #3
 - a. If the jammer is effective, the connection between the two devices is never established and the two devices would be incapable of communicating
5. If jamming is not effective → turn off Mesh Device #3 and move further away, in increments of 0.25m, and repeat steps 3 and 4
 - a. Repeat until mesh devices are unable to communicate with one another
 - b. Repeat until 3.0m mark

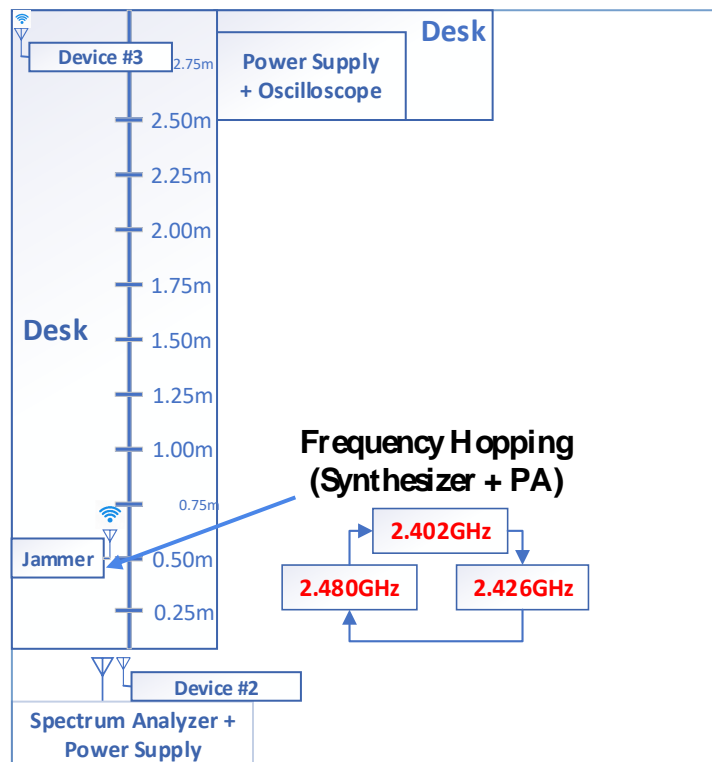


Figure 51: Test Set up Synthesizer jammer

The frequency hopping technique turned out to be unsuccessful for the purpose of jamming the two mesh devices. This is likely due to the fact that the frequency hop of the advertising channels is slower and is continuously “chasing after” the actual transmission of advertising packets of the BLE device communication. SIG defines the timing between two consecutive advertising PDUs to be less than or equal to 10ms. Based off this information an advertising event with three advertising PDUs transmitted on the three primary advertising channels would take less than or equal to 30ms. The frequency hopping technique is slower and takes approximately 378ms ($130 + 122 + 106 + 10 + 10 = 378\text{ms}$) to complete one iteration. Although the frequency hopping method was found to be too slow, it might have been capable of jamming if sufficient energy was placed at or near the PHY layer of the primary advertisement channels. The reference sensitivity of the PHY layer is -70dBm at a maximum. With the incorporation of additional power amplification, the average power of the individual hopping signals could be increased sufficiently as to jam the advertisement PHY channels. In addition, the frequency hopping method could have been successful in corrupting BLE communication enough as to increase the BER. BLE devices are certified by SIG if and only if the BER achieved is 0.1% or less. The frequency hopping method may have been successful in increasing the BER above the 0.1% threshold but there is no way to confirm this without the utilization of a sniffer. The initial concept of continuously jamming all three primary advertising channels is explored once again due to the shortcomings of the frequency hopping approach.

Chapter 7

SYSTEM TESTING & CHARACTERIZATION – CONSTANT JAMMER

7.1 Constant Jammer System Overview

This section provides the second modified high-level overview of the Bluetooth Low Energy jammer design. In the “Testing – Frequency Hopping Jammer” section of Chapter 6, the synthesizer’s frequency hopping timing proved to be ineffective for the task of jamming a victim network. A new scheme for jamming the BLE mesh is necessary since the frequency hopping approach is ineffective. The shortcoming of the frequency hopping approach was the timing interval between the individual synthesized frequencies of the hop. The frequency hop was essentially “chasing after” the advertising packet transmissions between the mesh devices whose timing intervals are shorter (less than 10ms).

The initial idea of the proposed jammer system, to continuously radiate tones with frequencies corresponding to the primary advertising channels, is explored once again. Additional devices capable of RF frequency synthesis are explored. The availability of low-cost frequency synthesizers with the desired output power specs to effectively jam a BLE network is rather scarce. Software defined radios (SDR) are determined to be a feasible option for RF frequency synthesis and transmission. Two SDRs are necessary for jamming three individual PHY channels (Ch. 37, 38, & 39) as the synthesizer evaluation board is only capable of continuously transmitting a single tone.

7.1.1 SDR Selection

Software defined radios are communication systems that are highly integrated and can come as a receiver-only device or a transceiver. SDRs are configurable - some of which are able to receive and transmit a wide variety of wireless protocols such as BLE and Wi-Fi. SDRs capable of communicating via BLE are ideal as they are ensured to transmit signals at a sufficient power for jamming purposes. A search for SDRs capable of synthesizing ISM band frequencies and having a reasonable TX output power is necessary.

Analog Device’s ADALM-Pluto SDR was determined to be a suitable device for frequency synthesis; this SDR costs approximately \$150 after taxes. Thanks to a generous loan from a couple of peers, the acquisition

of two ADALM-Pluto SDRs was possible. This SDR has the following specs: RF coverage from 325MHz to 3.8GHz, Matlab support, output power of 7dBm, and is USB powered [36]. The output power is greater than the minimum transmitter power required by a BLE transmitter. For the new approach, these SDRs are determined to be capable of continuously transmitting RF frequencies to supplement the synthesizer jammer in jamming the mesh network. The figure below depicts the ADALM-Pluto SDR.



Figure 52: ADALM-Pluto SDR [36]

The SDRs rely on the Analog Devices ADALM-Pluto Radio support package from Matlab’s Communication Toolbox. The USB connection from the SDR to a PC enables the simulation and development of various SDR applications. The figure below depicts the block diagram for the SDR system [37]. It is important to note that configuration and programming of the SDRs is possible through Matlab’s Simulink design environment. Further details on the programming process for the Pluto SDR are described in the next section.

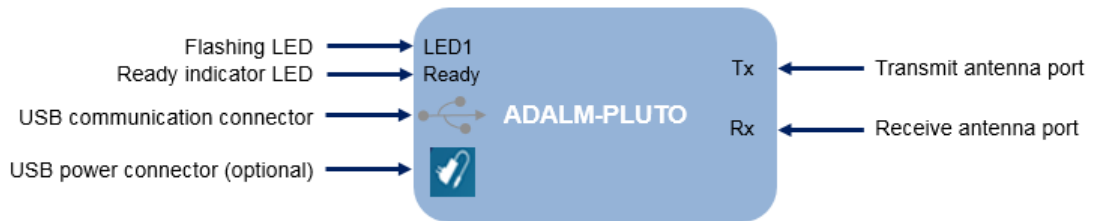


Figure 53: ADALM-Pluto SDR I/O [37]

The revised modified high-level diagram of the figure below depicts the new approach utilizing the SDRs. In this approach, two SDRs continuously transmit tones corresponding to two of the primary advertising channel frequencies, 2.426GHz (Ch. 38) and 2.480GHz (Ch. 39). The ‘N’ differential output of the synthesizer is used once again and connected directly to the power amplifier; the synthesizer-based jammer will now only output a single frequency corresponding to primary advertising channel 37 (2.402GHz). This system is tested in the “Testing – Constant Jammer” section of this chapter.

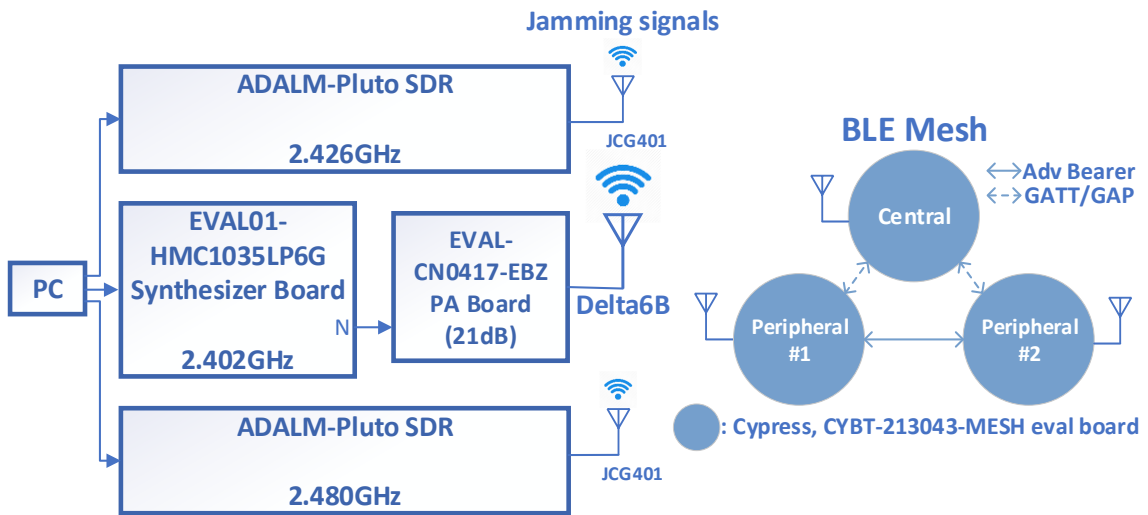


Figure 54: Modified Block Diagram – Synthesizer and Two SDRs

7.2 Signal Characterization – SDR

The ADALM-Pluto SDR can be utilized in the Matlab environment by installing the Analog Devices ADALM-Pluto Radio support package from the Communications Toolbox. Notably, there is a Matlab command `configurePlutoRadio('AD9364')` which allows for use of the AD9364 firmware as opposed to the default AD9363 RF transceiver chip [36]. This Matlab command enables the Pluto radio to adjust its RF output range to 70MHz to 6.0GHz from the original 325MHz to 3.8GHz range.

A Matlab function `sdrtx(DeviceName, Name, Value)` creates a transmitter system object for the ADLAM-Pluto radio hardware [38]. The system object has the following properties: DeviceName, RadioID, CenterFrequency, Gain, ChannelMapping, and BasebandSampleRate. The DeviceName is ‘Pluto’, the RadioID is a device-independent index starting from 0 with the prefix ‘usb:’, and the CenterFrequency is RF

output frequency of the transceiver chip. The Gain parameter specifies the transmitter gain in dB units and is a scalar from -89.75 to 0dB with a resolution of 0.25dB. ChannelMapping is a read-only parameter and BasebandSampleRate is a parameter specified in samples per second ranging from 65.105k to 61.44M.

The “Repeated Waveform Transmitter” Matlab function enables the continuous transmission of RF signals using this SDR. The *transmitRepeat(tx, data)* enables the downloading of a waveform (*data*) to the radio (*tx*) and repeatedly transmits it over the air [39]. The waveform transmission is continuous, without gaps, until the *release(tx)* function releases the radio transmission. A sinusoidal waveform was created for the waveform (*data*), with the following specifications: amplitude of 3 (unitless), frequency of 100kHz, complex, sampling rate of 1GHz, and 5000 samples per frame. The amplitude of the 3 is experimentally determined as the optimal value by measuring the transmit power via an SMA cable to the spectrum analyzer. The sampling rate is noticeably high as to improve the resolution and signal to noise ratio of the transmitted waveform.

The ADALM SDR was tested to synthesize signals corresponding to the three primary advertising channels. The CenterFrequency parameters are set to the three primary channel frequencies. The figure below depicts the SDR output (2.426GHz) as measured by a spectrum analyzer. A male to male SMA connector connects the TX port of the SDR to the spectrum analyzer. The peak strength of the synthesized signal is -1.364dBm. This peak signal strength is above the minimum required BLE transmitter power by approximately 19dB.

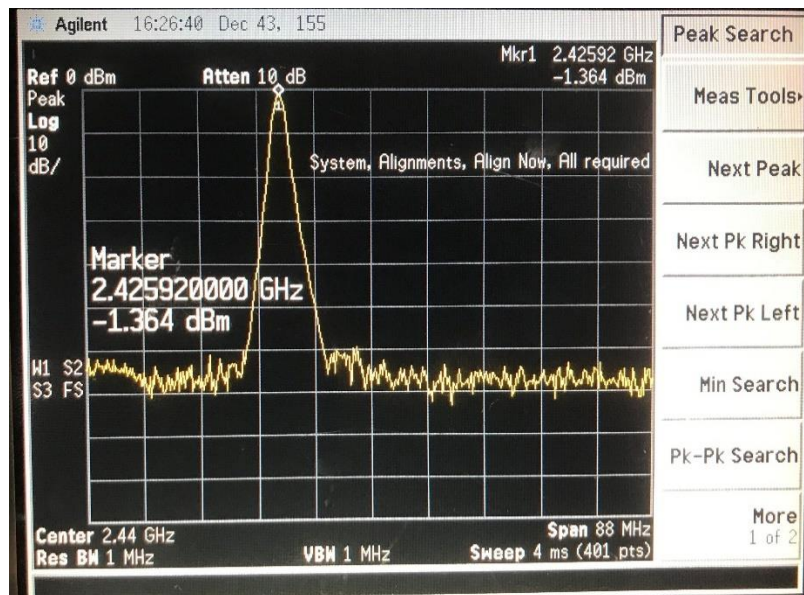


Figure 55: Spectrum Analyzer Measuring 2.426GHz Signal Output of ADALM SDR

7.3 System Requirements

The constant BLE jammer must continuously produce 3 tones at center frequencies 2.402GHz, 2.426GHz, and 2.480GHz. The BLE network must communicate via the BLE protocol and be mesh-compatible. The jamming signal must be transmitted with enough power for the devices of the mesh network to no longer be capable of establishing connections with one another. As stated previously, the maximum receiver sensitivity for the PHY layer channels is -70dBm and a co-channel interference signal strength must be 21dB below the sensitivity specification. A jamming signal strength of -91dBm or ~0.794pW in linear units is the minimum interference signal strength to corrupt BLE communication at the PHY layer.

For this project, the jamming system attacks a BLE mesh network within the same room. Friis's power equation (5-1) is used once again to estimate the power budget of a wireless link. The gain of the SDR antenna, JCG401, is 2dBi or ~1.585 in linear units while the gain of the receiver, the Cypress mesh board, is -0.5dBi or ~0.891 in linear units. The SDR's output signal was measured to be -1.364dBm or ~0.731mW in linear units. The SDR outputs do not have power amplifiers as their integration into the proposed system was during the final phase of testing. The wavelength of the system is 0.125m ($\lambda = c/f = 3e8/2.4e9 = 0.125m$). Friis equation can now be used to solve for the received power at the device to-be-jammed, as a function of the distance, as shown in equation 7-1.

$$P_{RX_SDR} = 0.891 * 1.585 * \left(\frac{0.125}{4\pi d}\right)^2 * 0.731m = 1.021 * 10^{-7}/d^2 \text{ W} \quad (7-1)$$

Equation 7-1 is plotted to visualize the received power with respect to distance. The SDR-based jammer reaches the 0.794pW threshold; therefore, this jammer has a theoretical maximum jamming distance of approximately 358.52m. Notably, the y-axis scaling for Figure 56 is 1×10^{-11} .

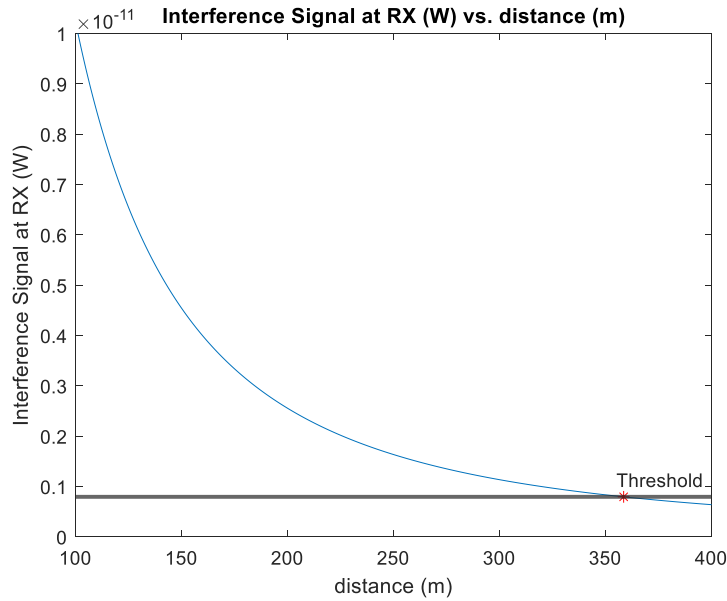


Figure 56: Finding Max Distance of Jamming Signal Reception (SDR + JCG401)

The same range calculator by Silicon Labs is used once again to calculate the range of the SDR-based jamming signal with higher accuracy than Friis's equation. The inputs to the range calculator are the TX output power, TX antenna gain, receiver sensitivity, RX antenna gain, and operating frequency. These inputs are set in the range calculator, for the SDR(s), and the results are shown below in Figure 57.


Frequency Band [MHz] <input type="text" value="2440"/>		 SILICON LABS	
<u>TX parameters</u> Enter TX Output Power [dBm] <input type="text" value="-1.4"/>		<u>RX parameters</u> Enter RX Sensitivity [dBm] <input type="text" value="-70"/>	
Enter TX Antenna Gain [dBi] <input type="text" value="2"/>		Enter RX Antenna Gain [dBi] <input type="text" value="-0.5"/>	
Typical Open Field Outdoor Range [m]		61.1	
Typical Indoor, Small Office Range [m]		9	

Figure 57: RF Range Calculator by Silicon Labs (SDR + JCG401)

The results of the range calculator state that in a typical outdoor setting the SDR-based jamming signal can successfully interfere with BLE communication at a distance of 61.1m. For a typical indoor setting, the range is computed as 9m. From these range calculator results, it can be concluded that the jamming signals can

successfully jam a BLE receiver in a typical small office environment, which is the testing environment proposed for this project.

7.4 Testing – Constant Jammer

In this approach, the individual ADALM SDRs will synthesize frequencies corresponding to two of the primary advertising channels, Ch. 38 and Ch. 39. The synthesizer will be configured to synthesize the third primary advertising channel frequency, Ch. 37. The figure below depicts the constant jammer utilizing the ADALM SDRs + synthesizer jammer, referred to as approach #2 from here on.



Figure 58: Constant Jammer, Side View (Approach #2)

Figure 60 below also depicts the constant jammer set up. The two SDRs transmit continuously at 2.426GHz and 2.480GHz, corresponding to advertising channels 38 and 39. The synthesizer jammer, utilizing the HMC1035LP6GE IC, continuously transmits a single frequency of 2.402GHz, corresponding to channel 37. By utilizing a 3-tone continuous RF transmission, the hope is to block the advertising data on the primary advertising channels. All signal transmission on the channels used for advertising, even in the case of extended and periodic advertising, starts on the primary channels.

The constant jammer uses the same test set up as the frequency hopping one; the SDR’s wireless transmission is characterized. The synthesizer-based jammer’s wireless transmission was already characterized in the “Testing – Frequency Hopping Jammer” section of Chapter 6. The SDR’s output signal power is measured at increments of 0.25m from 0m up until 3m. The center frequency of the SDR transmission is 2.426GHz. The antenna of the SDR (JCG401), included with the ADALM-Pluto SDR starter kit, has a gain of 2dBi. The spectrum analyzer equipped with an antenna, the Delta6B, measures the received signal power in units dBm. The figure below displays the plot of received signal power vs. distance. At a distance of 0m, the jamming signal is -16.65dBm. At a distance of 3m the jamming signal is -42.32dBm. For all distances, the jamming signal level, at the receiver, is greater than the reference sensitivity of the BLE PHY layer. The measurements become more susceptible to interference as the SDR’s distance from the spectrum analyzer increases, due to the testing environment limitations described previously.

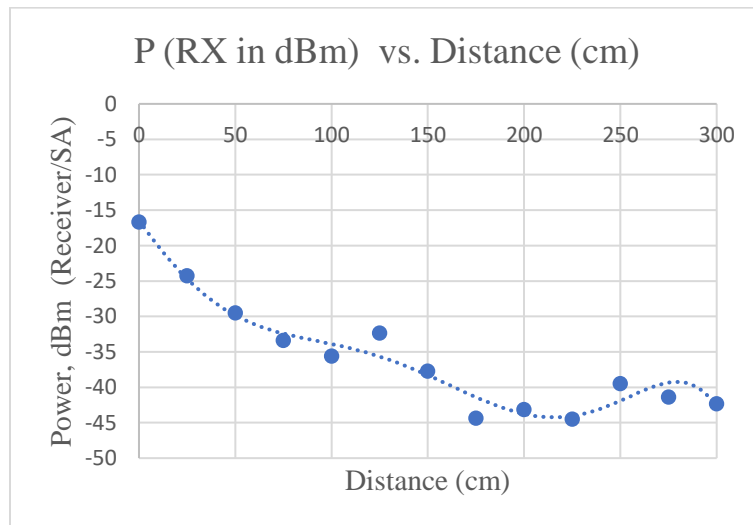


Figure 59: Measured Signal Strength (SDR) at Receiver/SA vs. Distance

Following the characterization of the SDR jamming signal strength, the continuous jamming scheme is implemented. The following constraints were considered: placement of victim devices, on/off state of synthesizer jammer and SDRs, and on/off state of BLE mesh devices. The figure below depicts the set-up of the synthesizer jammer and two SDRs. Here, the synthesizer jammer and two SDRs are placed at the 0.5m mark, adjacent to one another. The spectrum analyzer’s antenna and Mesh Device #2’s antennas are located at the 0m mark.

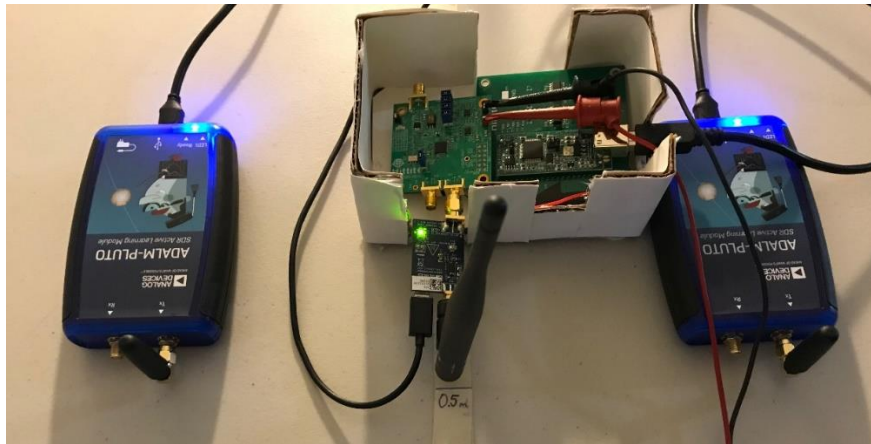


Figure 60: Constant Jammer Located at 0.5m Mark

The figure below depicts the synthesizer jammer output (2.402GHz) as the leftmost peak and the SDR outputs (2.426GHz and 2.480GHz) as the middle and rightmost peaks. The three peaks, from left to right, are measured to be -5.007dBm, -29.39dBm, and -26.84dBm. The figure is of a photo taken at an angle due to the presence of the spectrum analyzer antenna; the spectrum analyzer allows export of data via a floppy disk, but a floppy disk was not readily available at the time of testing.

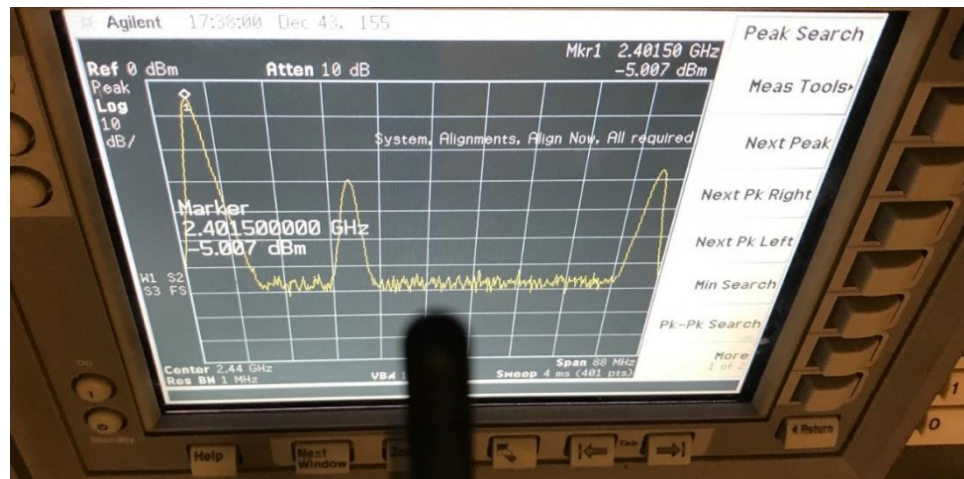


Figure 61: Strength of Signal Generated by Constant Jammer at Distance of 0.5m

The figure below compliments the testing process. The testing process is similar to that of the frequency hopping jammer and goes as follows:

0. Turn off all mesh devices, synthesizer jammer, and SDRs
1. Turn on jammer and two ADALM-Pluto SDRs
 - a. Synthesizer jammer & 2 SDRs are located at 0.5m mark, adjacent to one another

- b. Synthesizer jammer & 2 SDRs continuously transmit on the primary advertising channels
 2. Turn on Mesh Device #2 (LED)
 - a. Mesh device #2 is placed next to spectrum analyzer's antenna, at 0m mark, so jamming signal strength received by this device can be quantified
 3. Place Mesh Device #3 (LED dimmer) at 0.75m (for first iteration of step 3 only) mark and turn on
 4. Test functionality of communication between Mesh Devices #2 & #3 by pressing the 'User' button of the dimmer 10 times and record how many result in toggling of device #2's LED
 - a. If the jammer is effective, the communication between the two devices is interfered with and the two devices would be incapable of communicating
 5. Turn off Mesh Device #3, move it 0.25m further along desk set up, and repeat steps 3 & 4
 - a. Repeat until 3.0m mark

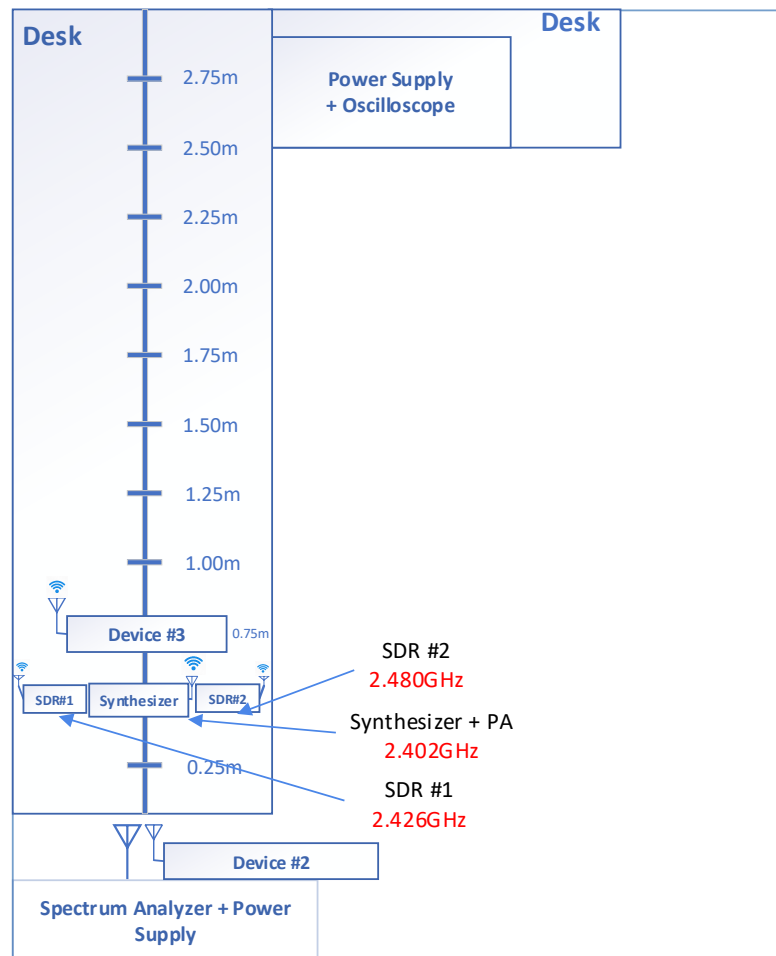


Figure 62: Test Set up Constant Jammer (Iteration #1)

The continuous transmission scheme proved to be successful for the purpose of jamming the mesh network for Mesh Device #3 locations 1.25m up to 3m. To clarify, when device #3 is 1.25m or further away from

device #2, the jamming signals are effective in disrupting the communication between the two devices. The data acquired from the test is compiled in Table 6. The “# of Success” row represents the total number of times, out of 10, the jamming of the communication was successful. A score of 8 indicates 8 button presses out of 10 of the dimmer board (device #3) were not received by the dimmable light (device #2). The three jamming signal levels in the table caption are the ones measured at the antenna, prior to wireless transmission; these levels are 17.26dBm (synthesizer output + 21dB PA) and -1.36dBm (SDRs).

Table 6: Constant Jammer Success (Signal at Antenna (dBm): 17.26, -1.36, -1.36)

dist (cm)	75	100	125	150	175	200	225	250	275	300
# Success	0	8	10	10	10	10	10	10	10	10

7.5 Additional Exploration

After successful testing of the continuous emission scheme for jamming, the power transmitted by the synthesizer jammer was reduced. The test described for the constant jammer is performed once again without the power amplifier. The SDR output signals received at the spectrum analyzer had sufficient strength (~ -30 dBm) for jamming two of the primary advertising channels. This observation motivated the test described in this section. The figure below depicts the synthesizer jammer output (2.402GHz), without power amplification, as the leftmost peak and the SDR outputs (2.426GHz and 2.480GHz) as the middle and rightmost peaks. The three peaks, from left to right, are measured to be -27.77 dBm, -33.45dBm, and -32.14dBm. The figure is of a photo taken at an angle due to the presence of the spectrum analyzer antenna.

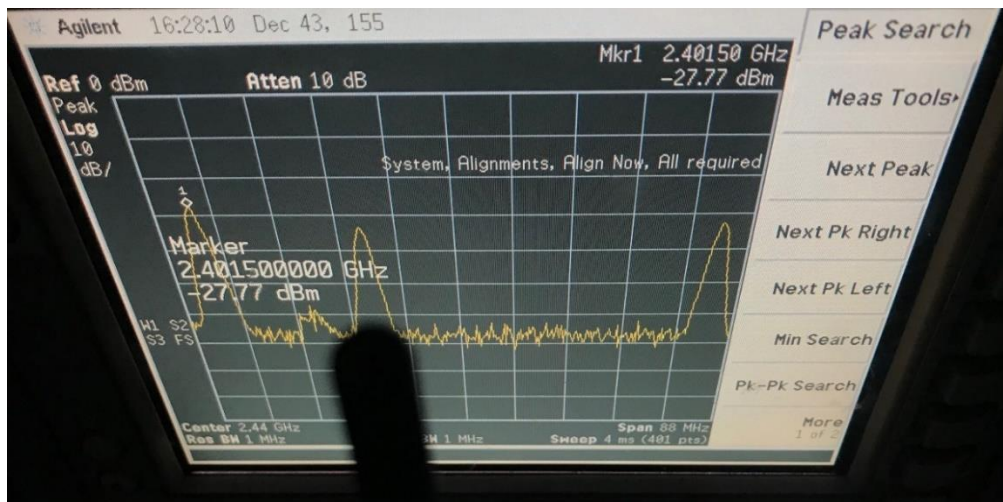


Figure 63: Strength of Signal Generated by Constant Jammer (w/o the PA) at Distance of 0.5m

The continuous transmission scheme successfully jammed the mesh network once again, even without power amplification, for Mesh Device #3 locations 1m up to 3m. Table 7 compiles the test data. Interestingly, this approach, which does not utilize power amplification, has slightly better performance results. This may be due to human error in the testing process. The time of day at which the testing is performed affects the results. The Wi-Fi networks that exist in the vicinity of the home testing environment fluctuate in their network activity depending on the time of day. The test discussed in this section was performed at 8AM in the morning. The test that includes the power amplifier was performed at a later time, 8PM, when network activity is heavier. Regardless, the conclusion is the same. The continuous transmission of interference on the primary advertising channels (Ch. 37, 38, & 39) can effectively jam the transmission of advertising packets. The three jamming signal levels in the table caption are the ones measured at the antenna, prior to wireless transmission; these levels are -3.74dBm (synthesizer output without PA) and -1.36dBm (SDRs).

Table 7: Constant Jammer (w/o PA) Success (Signal at Antenna (dBm): -3.74, -1.36, -1.36)

dist (cm)	75	100	125	150	175	200	225	250	275	300
# Success	0	10	10	10	10	10	10	10	10	10

Chapter 8

CONCLUSION

8.1 Reflection

The goal of this thesis project was to design, build, and test a constant jammer for attacking the Bluetooth Low Energy standard. By understanding the protocol's scheme for advertising-related data transmission and reception, it was believed that a PHY layer attack would prevent BLE mesh devices from establishing connections and transferring advertising data. The BLE jammer was initially proposed as a PCB, capable of synthesizing, power-combing, power-amplifying, and transmitting a 3-tone jamming signal. The 3-tone jamming signal would be transmitted continuously and interfere with BLE communication of the primary advertising channels (Ch. 37, 38, & 39). An MCU would control the three individual synthesizers, the three synthesized signals would be power combined via a Wilkinson power combiner, the 3-tone signal would then be amplified via a power amplifier, and an antenna would radiate the 3-tone jamming signal. COVID-19 supply chain related issues, component cost³, and lack of accessible laboratory equipment resulted in two modifications to the initial proposed jammer PCB.

The first modified system includes a PC connected to a single synthesizer configured to frequency hop, a balun for signal conversion from differential to single-ended, and an antenna to radiate the individual jamming signals. A balun was not integrable into the system design due to COVID-19 supply chain issues and component cost/availability. The second modified system includes a PC connected to a single synthesizer & 2 software defined radios and three individual antennas to radiate the three tones with center frequencies of 2.402GHz, 2.426GHz, and 2.480GHz. The two modified systems are tested on a victim BLE mesh of Cypress mesh evaluation boards equipped with Bluetooth 5.0. The Cypress mesh boards utilize the primary advertising channels for transmission of advertising-related data and do not implement extended advertising events (introduced with BT version 5.0) by default. As a leader in Bluetooth PSoC development, Cypress's products are assumed to represent the current state of art in the field of BLE applications.

The first modified system was unsuccessful in jamming due to the timing for the execution of frequency hopping via a single synthesizer. The time between advertising packet transmissions on the primary

³ The thesis project has a budget of \$200.

advertising channels is defined by SIG to be less than or equal to 10ms. The frequency hopping timing could not achieve this time resolution, making the jamming attempt ineffective. The constant jammer was successful in jamming the primary advertising channels and preventing BLE devices from establishing a connection and transferring advertising data. The continuous transmission of tones with center frequencies corresponding the primary advertising channel frequencies successfully interfered with BLE communication. By proving the concept of jamming, the feasibility of building the relatively low-cost proposed PCB jammer system is validated.

8.2 Future Works

If time permitted, further effort would continue for the following: creating the initially proposed jammer system PCB, additional exploration into BLE mesh network jamming, and improving the stealth of the jammer. Much of the jammer system PCB hardware selection, including the synthesizer IC (HMC1035PG6E), lumped implementation of a 3-way Wilkinson power combiner, power amplifier IC (ADL5606), and antenna (Delta6B), has been completed. The cost of manufacturing this PCB would be within the thesis budget of \$200. For the PCB to function optimally, controlled impedance traces, impedance matching, bypass components, and other critical components would be required.

Further exploration into BLE mesh network jamming would emphasize the acquisition of reliable and consistent data. The data presented in the testing section of this thesis is not as accurate as it would be had the testing been performed in an anechoic chamber. The acquisition of data in an anechoic chamber environment would be ideal as the testing results would be more repeatable relative to the results obtained in the home test environment presented in this paper. Additionally, the BLE mesh network jamming scheme could be improved by building a frequency hopping jammer capable of transmitting jamming tones corresponding to secondary advertising channel frequencies. Testing of this new jamming scheme would most likely benefit from creating a BLE mesh network implementing extended advertising events; by creating this mesh network, the jammer's effectiveness can be confirmed.

The BLE mesh network jamming can also be improved in its characterization of performance. The jamming signals utilized in this thesis are at a set level. The LVPECL performance priority output is not adjustable in its magnitude due to the limitations of the synthesizer evaluation board. The SDR output was also set at a constant level throughout the tests. In the future it would be useful to test the effectiveness of jamming at

different transmitter power levels. Additionally, different physical arrangements of the jammer relative to the mesh devices could also be explored.

Lastly, the stealth of the jammer would be improved. The jammer discussed in this paper is a constant one. Of the four brute-force jamming techniques, the constant one is the least efficient as jamming signal transmission is continuous. Continuous jammers are easier to detect relative to others as the transmission of electromagnetic energy is detectable (i.e. via a sniffer) for the duration of the constant transmission. The stealth of the jammer system, often assessed by the probability of detection, could be improved by adopting MAC layer behaviors. This would make the jammer a deceptive one instead of a constant one. Deceptive jamming is more difficult to detect as network monitoring tools sense legitimate traffic on the channels.

REFERENCES

- [1] T. Salman and R. Jain, "Networking Protocols and Standards for Internet of Things," in *Internet of Things and Data Analytics Handbook*, H. Geng, Ed. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2016, pp. 215–238.
- [2] M. Woolley, "Bluetooth 5: Go Faster, Go Further." Bluetooth SIG, Accessed: May 18, 2020. [Online]. Available: <https://www.bluetooth.com/bluetooth-resources/bluetooth-5-go-faster-go-further/>.
- [3] Commission électrotechnique internationale, *Internet of things: wireless sensor networks*. Ginevra: IEC, 2014.
- [4] "Nordic BLE Mesh," *Digi-Key*. <https://www.digikey.com/en/product-highlight/n/nordic-semi/nordic-ble-mesh> (accessed May 19, 2020).
- [5] F. Gilbert, "PRIVACY AND SECURITY LEGAL ISSUES," in *Internet of Things and Data Analytics Handbook*, H. Geng, Ed. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2016, pp. 699–718.
- [6] L. Oliveira, J. Rodrigues, S. Kozlov, R. Rabêlo, and V. Albuquerque, "MAC Layer Protocols for Internet of Things: A Survey," *Future Internet*, vol. 11, no. 1, p. 16, Jan. 2019, doi: 10.3390/fi11010016.
- [7] K. Pelechrinis, M. Iliofotou, and S. V. Krishnamurthy, "Denial of Service Attacks in Wireless Networks: The Case of Jammers," *IEEE Commun. Surv. Tutorials*, vol. 13, no. 2, pp. 245–257, 2011, doi: 10.1109/SURV.2011.041110.00022.
- [8] "Our History," *Bluetooth*. <https://www.bluetooth.com/about-us/our-history/> (accessed Jan. 20, 2020).
- [9] "Vision and Mission," *Bluetooth*. <https://www.bluetooth.com/about-us/vision/> (accessed Jan. 20, 2020).
- [10] P. Wiecha, M. Cieplucha, P. Kloczko, and W. A. Pleskacz, "Architecture and design of a Bluetooth Low Energy Controller," in *2016 MIXDES - 23rd International Conference*

Mixed Design of Integrated Circuits and Systems, Lodz, Poland, Jun. 2016, pp. 164–167,
doi: 10.1109/MIXDES.2016.7529724.

- [11] “Bluetooth Low Energy protocol stack,” *Nordic Semiconductor*.
https://infocenter.nordicsemi.com/index.jsp?topic=%2Fsds_s140%2FSDS%2Fs1xx%2Fble_protocol_stack%2Fble_protocol_stack.html (accessed Mar. 22, 2020).
- [12] “BLE advertising channels spectrum,” *Argenox*. <https://www.argenox.com/static/assets/ble-advertising-channels-spectrum.png> (accessed Mar. 14, 2020).
- [13] “Specification of the Bluetooth System, v5.1.” Bluetooth SIG, Accessed: Dec. 13, 2019.
[Online]. Available: <https://www.bluetooth.com/specifications/bluetooth-core-specification/>.
- [14] S. Brauer, A. Zubow, S. Zehl, M. Roshandel, and S. Mashhadi-Sohi, “On practical selective jamming of Bluetooth Low Energy advertising,” in *2016 IEEE Conference on Standards for Communications and Networking (CSCN)*, Berlin, Germany, Oct. 2016, pp. 1–6, doi: 10.1109/CSCN.2016.7785169.
- [15] J. Katsandres, “Exploring Bluetooth 5 - What’s New in Advertising?,” *Bluetooth*, Feb. 27, 2017. <https://www.bluetooth.com/blog/exploring-bluetooth5-whats-new-in-advertising/> (accessed Mar. 20, 2020).
- [16] “BLE Advertisement Interval,” *Silicon Labs*.
https://www.silabs.com/content/usergenerated/asi/cloud/attachments/siliconlabs/en/community/wireless/bluetooth/forum/jcr:content/content/primary/qna/scanning_vs_advertising/so_a_site_i_ve_been-krzp/advertise.JPG (accessed May 19, 2020).
- [17] K. Ren, “What You Need to Know About Periodic Advertising Sync Transfer,” *Bluetooth*.
<https://www.bluetooth.com/blog/periodic-advertising-sync-transfer/> (accessed May 01, 2020).
- [18] “Bluetooth and WLAN frequencies,” *All About Circuits*.
https://www.allaboutcircuits.com/uploads/articles/Bluetooth_and_WLAN_frequencies.jp

- g (accessed Jan. 24, 2020).
- [19] "EVAL01-HMC1035LP6G," *Digi-Key*. <https://www.digikey.com/product-detail/en/analog-devices-inc/EVAL01-HMC1035LP6G/1127-2282-ND/4794582> (accessed May 02, 2020).
- [20] A. Niknejad, "Amplifiers Power Combining." 2014, Accessed: Nov. 10, 2019. [Online]. Available: http://rfic.eecs.berkeley.edu/ee242/pdf/Module_6_3_PowerComb.pdf.
- [21] F. Noreiga and P. Gonzalez, "Wilkinson Power Splitters." 2002, Accessed: Nov. 20, 2019. [Online]. Available: <http://www.cisarelba.it/progetti/Wilkinson-power-splitters.pdf>.
- [22] "CYBT-213043-MESH EZ-BT Module Mesh Evaluation Kit," *Cypress*, Nov. 11, 2019. <https://www.cypress.com/documentation/development-kitsboards/cybt-213043-mesh-ez-bt-module-mesh-evaluation-kit#res574> (accessed Mar. 20, 2020).
- [23] "ModusToolbox IDE User Guide." Cypress, Accessed: Apr. 24, 2020. [Online]. Available: <https://www.cypress.com/file/492951/download>.
- [24] M. Woolley, "Management of Devices in a Bluetooth Mesh Network," *Bluetooth*, Aug. 28, 2017. <https://www.bluetooth.com/blog/management-of-devices-bluetooth-mesh-network/> (accessed May 24, 2020).
- [25] K. Ren, "Provisioning a Bluetooth Mesh Network Part 1," *Bluetooth*, Sep. 18, 2017. <https://www.bluetooth.com/blog/provisioning-a-bluetooth-mesh-network-part-1/> (accessed May 21, 2020).
- [26] "ModusToolbox™ MeshClient and ClientControlMesh App User Guide." Cypress, Accessed: May 12, 2020. [Online]. Available: <https://www.cypress.com/file/462491/download>.
- [27] "Analog Devices Welcomes Hittite Microwave Corporation," *Analog Devices*. https://www.analog.com/media/en/technical-documentation/evaluation-documentation/140-00100-00_user_manual.pdf (accessed Jan. 20, 2020).
- [28] "HMC1035LP6GE." Analog Devices, [Online]. Available:

- <https://www.analog.com/media/en/technical-documentation/data-sheets/hmc1035.pdf>.
- [29] “Signal Types and Terminations,” *Vectron*.
https://www.vectron.com/products/literature_library/Signal_Types_and_Terminations.pdf
(accessed Oct. 20, 2019).
- [30] “Analog Devices Welcomes Hittite Microwave Corporation.” Analog Devices, Accessed:
Jan. 20, 2020. [Online]. Available: [https://www.analog.com/media/en/technical-
documentation/evaluation-documentation/hmc1035lp6g_eval_pcb_schematic.pdf](https://www.analog.com/media/en/technical-documentation/evaluation-documentation/hmc1035lp6g_eval_pcb_schematic.pdf).
- [31] “ADC-WB-BB / ADC-LD-BB User’s Guide.” Texas Instruments, Accessed: May 19, 2020.
[Online]. Available:
<http://www.ti.com/lit/ug/snau123/snau123.pdf?&ts=1589930849604>.
- [32] “10M-3000MHZ 3GHz RF Differential Single-Ended Converter Balun 1:1 ETC1-1
ADF4350 module,” *Aliexpress*, Mar. 22, 2020.
[https://www.aliexpress.com/item/32924235890.html?spm=2114.12057483.0.0.79dc48c7
b1GYC5](https://www.aliexpress.com/item/32924235890.html?spm=2114.12057483.0.0.79dc48c7b1GYC5)
- [33] J. Sponas, “Things You Should Know About Bluetooth Range,” *Nordic Semiconductor*, Feb.
07, 2018. [https://blog.nordicsemi.com/getconnected/things-you-should-know-about-
bluetooth-range](https://blog.nordicsemi.com/getconnected/things-you-should-know-about-bluetooth-range) (accessed Mar. 22, 2020).
- [34] “CN0417 USB Powered 2.4 GHz RF Power Amplifier,” *Analog Devices*.
[https://www.analog.com/en/design-center/reference-designs/circuits-from-the-
lab/cn0417.html#rd-overview](https://www.analog.com/en/design-center/reference-designs/circuits-from-the-lab/cn0417.html#rd-overview) (accessed Apr. 20, 2020).
- [35] “RF Range Calculator,” *Silicon Labs*.
[https://www.silabs.com/community/wireless/proprietary/knowledge-
base.entry.html/2017/05/02/rf_range_calculator-SYIA](https://www.silabs.com/community/wireless/proprietary/knowledge-base.entry.html/2017/05/02/rf_range_calculator-SYIA) (accessed Apr. 22, 2020).
- [36] “Analog Devices Inc. ADALM-PLUTO Active Learning Module,” *Mouser Electronics*.
<https://www.mouser.com/new/analog-devices/adi-adalm-pluto/> (accessed May 12, 2020).
- [37] “Pluto Receiver,” *Mathworks*.

<https://www.mathworks.com/help/supportpkg/plutoradio/ref/plutoreceiver.html> (accessed May 02, 2020).

[38] “sdrtx,” *Mathworks*. <https://www.mathworks.com/help/supportpkg/plutoradio/ref/sdrtx.html> (accessed May 04, 2020).

[39] “transmitRepeat,” *Mathworks*.

<https://www.mathworks.com/help/supportpkg/plutoradio/ref/comm.sdrtxpluto.transmitrepeat.html> (accessed May 04, 2020).

APPENDICES

A: Frequency Hopping (Approach #3) Reg File – One Iteration:

```
REVISION 1.0.2.0
REG 5 90 // REGSET_87797 set_vco_to_out_divider
REG 9 50ED5A
REG C 0
REG 6 200B4A // setbit Write
REG 6 200B4A // setbit Write
REG 2 1 // setlng Write
REG 3 30 // REGSET_87797 set_intg_register
REG 5 0 // REGSET_87797 vco_setcap
REG 4 A3D71 // REGSET_87797 set_frac_register
REG 5 90 // REGSET_87797 set_vco_to_out_divider
REG 9 50ED5A
REG C 0
REG 6 200B4A // setbit Write
REG 6 200B4A // setbit Write
REG 2 1 // setlng Write
REG 3 30 // REGSET_87797 set_intg_register
REG 5 0 // REGSET_87797 vco_setcap
REG 4 A3D71 // REGSET_87797 set_frac_register
REG 5 90 // REGSET_87797 set_vco_to_out_divider
REG 9 50ED5A
REG C 0
REG 6 200B4A // setbit Write
REG 6 200B4A // setbit Write
REG 2 1 // setlng Write
REG 3 30 // REGSET_87797 set_intg_register
REG 5 0 // REGSET_87797 vco_setcap
REG 4 851EB8 // REGSET_87797 set_frac_register
REG 5 90 // REGSET_87797 set_vco_to_out_divider
REG 9 50ED5A
REG C 0
REG 6 200B4A // setbit Write
REG 6 200B4A // setbit Write
REG 2 1 // setlng Write
REG 3 31 // REGSET_87797 set_intg_register
REG 5 0 // REGSET_87797 vco_setcap
REG 4 99999A // REGSET_87797 set_frac_register
```

B: SDR Transmit – 2.480GHz

```
% create complex sine wave
fs = 1e9; % 1 GHz sampling frequency
sw = dsp.SineWave; % Create sine wave using dsp class
sw.Amplitude = 3; % set amplitude to 3
sw.Frequency = 100e3; % frequency set to 100kHz
sw.ComplexOutput = true;
sw.SampleRate = fs;
sw.SamplesPerFrame = 20000;
txWaveform = sw();

%configure settings for Pluto SDR output
tx2 = sdrtx('Pluto', 'RadioID', 'usb:0');
tx2.CenterFrequency = 2480e6;
tx2.BasebandSampleRate = 520.841e3;
tx2.Gain = 0; %units dB
```

```
% continuously transmit txWaveform centered at 2.426GHz
transmitRepeat(tx2,txWaveform);

runtime = tic;
while toc(runtime) < 100
end

release(tx);
```