

תוכנית הלימודים לתואר ראשון בהנדסת חשמל

שםuproject: חוסם רחפנים תלת ערכוי



מישל אלברכט - סטודנט להנדסת חשמל

גיא גולדמן - סטודנט להנדסת חשמל

צביקה רוזנבלום - מנההuproject

תאריך הגשה: 08.08.2025

2 אישור מנהה

אישור הגשת ספר הפרויקט - להדבקה

בעמוד 2



ZYR 13:32

to me ▾



מישל וגיא שלום,

אתם רשאים להגיש את ספר הפרויקט.

תודה ובהצלחה,
צביקה רוזנבלום

3 תודות

ברצוננו להודות מוקבך לב למנהל שלנו, **צביקה רוזנבלום**, שליווה אותנו במסירות ובה לאורך כל הדרך, השקיע מזמננו
ומניסיונו העשיר, והעניק לנו תמיכה מקצועית ואישית בכל שלבי הפרויקט.

תודה מיוחדת גם למוצרים שלנו בהתחמות תקשורת: לגברת **אירנה ליבשטר**, אשר סייעה לנו רבות בעבודת המעבדה
ובהעמקת הידע התאורטי ולפרופ' **עליז לין**, שסייעק לנו אנטוכיות אינטלקטואלית שבסנה.

4 תוכן עניינים

2	אישור מנהה
2	תעודות
3	תיקון עניינים
4	רשימת אירורים
5	רשימת טבלאות
6	רשימת גرافים
7	תקציר מנהלים
7	Executive Summary
8	מבוא
8	6.1 כללי
9	6.2 מיליון מונחים
11	6.3 מטרות, יעדים ומדדים
12	7. ניתוח חלופות
12	7.1 תוכנתי
13	7.2 מערכתי
14	7.3 טכנולוגיות
15	8. סקירה ספרות, סקר שוק
15	8.1 סקירה ספרותית
16	8.2 סקר שוק
18	9. שיטות (Methodology)
18	9.1 תכנן ותוכנון המערכת (Detailed planning)
18	9.1.1 תרשימים בלוקים
19	9.1.2 רכיב לביצוע חסימת ה-WIFI
26	9.1.3 רכיב לביצוע חסימת ה-GPS
29	9.1.4 פרוטוקול תקשורת SPI
30	9.1.5 מגברים
33	9.1.5 אנטנות
34	9.1.6 הרחפן
38	9.1.7 סימולציות תכנון מערכתיות ופרטניות
49	9.1.8 תהליכיים מפורטים לפועלה וממשק משתמש
53	9.1.9 ממשק המשתמש (GUI)
58	9.1.10 תכנון אינטגרציה וביצועה

59.....	9.2 חלוקת העבודה.....
59.....	9.3 תוכנית עבודה.....
59.....	9.4 ריכוז שינויים.....
60.....	9.5 ניהול סיכון
61	10 תוצאות.....
61.....	10.1 תיאור מפורט של תוצריו הסופיים של הפרויקט.....
62.....	10.2 ביצועי הדגם בפועל לעומת התכנון ב-SOW
63	11 סיכום ומסקנות.....
63.....	11.1 עיקרי המסקנות מן הפרויקט והישגים
63.....	11.2 יעדים שלא הושגו, בעיות עיקריות ולקחים
64.....	11.3 הערכת הפתרון שנבחר.....
64.....	11.4 הצעות לעבודת המשך
65	12 מקורות.....
66	13 נספחים
66.....	13.1 קוד משאך המשתמש
77.....	13.2 פוסטר הפרויקט
78.....	13.3 דפי נתונים

4.1 רשימת איורים

18	איור 1 - תרשימים בלוקים
21	איור 2 - תרשימים חיבורים LMX2492EVM
22	איור 3 - סידור הפינים של ה-CO החיצוני ו- Location Pad של ה-CO המיועד
22	איור 4 - מיקום ה-CO החיצוני על ה-MLX2492EVM
23	איור 5 - סכמת המעגל של לוח MLX2492EVM
23	איור 6 - סכמת מעגל LimeSDR Mini 2.0
27	איור 7 - תרשימים בלוקים M7002
27	איור 8 - תרשימים בלוקים LMS7002
28	איור 9 - שרשרת חסימה בתוכנת GNU Radio
29	איור 10 - פרוטוקול SPI
31	איור 11 - מגבר לחסימת GPS ZRL-2300+
31	איור 12 - נתונים של מגבר ZRL-2300+
32	איור 13 - מגבר ראשון בשרשרת לחסימת WIFI
32	איור 14 - מגבר שני בשרשרת חסימת WIFI

35	איור 15 - שלט הרחפן.
36	איור 16 - ספק כוח.
45	איור 17 - בניית האנטנה בסימולציה.
53	איור 18 - Home Page ,GUI -
54	איור 19 - Information ,GUI -
55	איור 20 - Technical ,GUI -
56	איור 21 - Operational,GUI -
57	איור 22 - Graphs ,GUI -
61	איור 23 - דגם סופי.

4.2 רשימת טבלאות

12	טבלה 1 - חלופה תוכניתית.
13	טבלה 2 - חלופה מערכית.
14	טבלה 3 - חלופה טכנולוגית.
17	טבלה 4 - סיכום המוצריים הקיימים בשוק.
19	טבלה 5 - נתונים של HackRF
34	טבלה 6 - נתוני רחפן DJI Mini 2
37	טבלה 7 - נתוני כבלי RF
38	טבלה 8 - עוצמת שידור כתלות בתדר ללא מגברים (GPS)
39	טבלה 9 - עוצמת שידור כתלות בתדר עם מגבר (GPS)
40	טבלה 10 - עוצמת שידור כתלות בתדר ללא מגבר (WIFI)
41	טבלה 11 - עוצמת שידור כתלות בתדר עם מגבר ראשון (WIFI)
42	טבלה 12 - עוצמת שידור כתלות בתדר עם שני המגברים (WIFI)
50	טבלה 13 - LimeSuite Functions
50	טבלה 14 - USB2Any Functions
52	טבלה 15 - הגדרות ליצירת Sweep תדרים בצורת משולש.
59	טבלה 16 - ריכוז שינויים.

4.3 רשימת גրפים

20	גרף 1 - עם עייפות
20	גרף 2 - ליניארי.
25	גרף 3 - Zligah בשימוש גל מסור. Spectrum Analyzer
26	גרף 4 - HackRF, Spectrum Analyzer
28	גרף 5 - Spectron & Waterfall.. GPS, חסימת GPS
34	גרף 6 - Spectrum Analyzer, תדרי פעילות הרחפן
35	גרף 7 - תצוגת Zero Span שהיית הרחפן בתדר מסוים
38	גרף 8 - GPS שידור Spectrum Analyzer ללא הגברה
39	גרף 9 - GPS שידור Spectrum Analyzer עם מגבר
40	גרף 10 - WiFi Spectrum Analyzer - WiFi לשידור ללא מגברים
41	גרף 11 - WiFi Spectrum Analyzer WiFi עם מגבר ראשון
42	גרף 12 - WiFi Spectrum Analyzer WiFi עם שני המגברים
43	גרף 13 - תצוגת Zero Span סריקה בתחום 2.4GHz
43	גרף 14 - Spectrum Analyzer הוכחת יכולת שידור ב- 5.8GHz
44	גרף 15 - תצוגת Zero Span סריקה בתחום 5.8GHz
45	גרף 16 - מדידת 11s, רוחב פס
47	גרף 17 - דיאגרמת Circular Polarization

5 תקציר מנהלי

הפרויקט מתמקד במתן פתרון הגנתי מפני שימוש לרעה ברחפנים, שנעשה יותר ויותר נפוץ בתחוםים כמו ריגול, חבלה, גניבת מידע והפרת פרטיות. עם העליה במספר הרחפנים הזרים בשוק, גדול גם הצורך במערכות שמסוגלות להתמודד עם איומים אלו בצורה יעילה ולא גרימת נזק מיותר לסייעתה.

המערכת מתוכננת לנטרל רחפנים באמצעות טכניקת Soft Kill, טכניקה מתקדמת המתמקדת בשימוש התקשורת בין המפעיל לרחפן, במקום בהשמדה פיזית של הרחפן עצמו. יתרון מרכזי בשיטה זו הוא הימנענות מפגיעה פיזית בסביבה או בצד קרקעי ומונעת הצורך בשימוש באמצעות חבלניים, אשר עלולים לגרום לנזקים בלתי צפויים.

כדי לזכור שלמרות השם "חוסם רחפנים", המבשיר אכן גורם לרחפן להיתקע באוויר וליפול ישירות לקרקע. כאשר חוסם הרחפנים שולח אותן חסימה לרחפן, הוא עשוי להגיב באחת משתי דרכים: הוא יכול לבצע נחיתה רכה ובטוחה על الكرקע, או לחזור לנקודת ההתחלה שהוגדרה לו מראש באמצעות מערכת GPS. עם זאת, אם גם ה-GPS נחסם, הרחפן יבצע נחיתה רכה במקום.

המערכת תהיה מסוגלת לספק הגנה מקיפה בנגד ורחפנים המבוססים על תקשורת Wi-Fi בתדרים של 2.4 GHz ו-5.8 GHz וכן בנגד רחפנים המשתמשים ב-GPS. באמצעות שילוב של טכנולוגיות שיבוש מתקדמות, יוכל להבטיח שהמערכת תוכל להתמודד ביעילות עם איומים מכל הסוגים ולהגן על הצד והכוחות בשטח בצורה המיטבית.

Executive Summary

The project focuses on providing a defensive solution against the growing misuse of drones in areas such as espionage, sabotage, data theft, and privacy violations. As the number of commercially available drones continues to rise, so does the need for systems capable of countering these threats effectively and without causing unnecessary harm to the environment. The system is designed to neutralize drones using a Soft Kill technique — an advanced method that disrupts the communication between the operator and the drone, rather than physically destroying the drone itself. A key advantage of this approach is the avoidance of physical damage to the surroundings or ground-based equipment, as well as eliminating the need for destructive countermeasures that could lead to unpredictable consequences.

It is important to note that despite being referred to as a “drone jammer,” the system does not cause drones to freeze mid-air and crash. When the jammer transmits a blocking signal, the drone typically responds in one of two ways: it may perform a controlled and safe landing on the ground, or it may return to its predefined home location using its GPS system. However, if the GPS signal is also jammed, the drone will generally initiate a soft landing at its current location.

The system is designed to provide comprehensive protection against drones operating via Wi-Fi communication in the 2.4 GHz and 5.8 GHz bands, as well as against drones that rely on GPS navigation. Through the integration of advanced jamming technologies, the system ensures effective mitigation of various types of drone threats, offering optimal protection for both equipment and personnel in the field.

6 מבוא

6.1 כללי

בשנים האחרונות התפתחות טכנולוגית מואצת של רחפנים יצרה גם הזרמוניות וגם אתגרים חדשים. השימוש הנרחב ברחפנים לצרכים מסחריים, חקלאיים ואישיים הביא לשיפור שימושם ביכולותיהם, אך במקביל, עליה הצורך להתמודד עם האיום והסכנות שמדוברים בתוצאה שימוש לרעה בטכנולוגיה זו. דוגמה עכשוית וمحשית לכך היא מלחמת "חרבות ברזל", שבה נעשה שימוש ברחפני חבלה נגד רכבים וטנקים, מה שהציג את הסכנה המשמעותית הנובעת משימוש זה.

הפרויקט נועד לספק פתרון מגן לכלי רכב צבאיים וביטחוניים, מתוך הבנה שהALKות הפוטנציאליים למערכת זו הם הצבאות ותשויות ביוטחניות שונות, אשר זוקקות לפתרונות ייעילים,אמינים וזולים להגנה מפני איומים מהאוויר.

לשם כך נפתח מערכת המבוססת על כמה מרכיבים עיקריים:
משדרים: רכיבים שי תוכנו במיוחד לשימוש אוטומטי RF המתאים, ובכך ימנעו תקשורת אפקטיבית בין הרחפן למפעיל.

אנטנות: מיעודות לשידור של אוטומטים בתדרים ספציפיים, כדי למסם את אפקטיביות השימוש.
ספקי כוח: תשתיות شاملות המבטיחות את הפעולה האמינה והמתמשכת של כל רכבי החומרה.
מסך גרפי למשתמש (GUI): תוכנה שמאפשרת למפעילים לשלוט במערכת, לצפות בננתונים בזמן אמת ולהגדיר פרמטרים שונים בהתאם לצרכים המשתמשים.

2.2 מילון מונחים

- **Jamming** - שיבוש מכoon של אוטות תקשורת אלחוטית, שמונע קבלת פקודות או מידע.
- **Sweep Jammin** - שיטה לשיבוש תקשורת באמצעות סריקה מהירה של טווח תדרים והזרקת אוטות שיבוש בתדר משתנה.
- **Barrage Jamming** - שידור בו-זמן על פני טווח רחב של תדרים, על מנת לשבש תקשורת מוביל לדעת מראש את התדר המדויק.
- **Spot Jamming** - שיבוש של תדר בודד בעוצמה גבוהה.
- **Smart Jamming** - שיבוש ממוקד שמתבסס על זיהוי דינמי של פרוטוקול או תדרי תקשורת, ודורש מידע מוקדם על המטרה.
- **SDR (Software Defined Radio)** - מקלט/ مصدر רדיו המוגדר בתוכנה. מאפשר שליטה דינמית בפרמטרי שידור כמו תדר, רוחב פס ועוצמה.
- **Named Pipe** - מנגן תקשורת בין תהליכים (IPC) ב-Windows.
- **PLL (Phase Locked Loop)** - מערכת בקרה שמייצבת תדר באמצעות השוואת פазה לאות ייחום.
- **VCO (Voltage Controlled Oscillator)** - מתנד שתדרו משתנה בהתאם למתח הכניסה.
- **FPGA (Field Programmable Gate Array)** - שבב לוגיקה שניית לבניתו מחדש לחישוב משימות עיבודיעודיות.
- **GUI (Graphical User Interface)** - ממשק גרפי המאפשר למשתמש להפעיל את המערכת.
- **Frequency Hopping** - טכנית שידור שבה תדר השידור משתנה במהירות כדי להקשות על שיבוש.
- **Ramp** - גל תדר לינארי - שינוי הדרגתית בתדר המשודר.
- **Chirp Signal** - אות whose תדר משתנה בזמן (עליה/ירד).
- **dBm** - ממד יחידת הספק (decibel-milliwatts).
- **Gain (dB)** - ממד שמבטא את עצמת ההגבר של אות, בלוגר כמה האות התחזק.
- **VSWR (Voltage Standing Wave Ratio)** - ממד לאיכות ההתאמה בין אנטנה למעגל השידור. ערבים נמוכים מצביעים על התאמה טובה.
- **MVP (Minimum Viable Product)** - גרסה בסיסית של מוצר, הכוללת רק את הפונקציות הבכירות על מנת להציג את הרעיון ולבחון את יעילותו.
- **WiFi (Wireless Fidelity)** - רשת תקשורת אלחוטית מקומית המשמשת לתקשורת בין התקנים ללא כבילים.
- **GPS (Global Positioning System)** - מערכת ניווט לווינית בינלאומית, המאפשרת קבלת מיקום גאוגרפי מדויק בכל מקום על פני כדור הארץ.
- **Soft Kill** - שיטה לנטרול רחפנים ללא גרים נזק פיזי, על ידי שיבוש או חסימת התקשורת.

SNR (Signal to Noise Ratio) - יחס אות לרעש - מدد המבטא את איכות האות על רקע הרעש שבערוץ התקשורות. ערך גבוה מצין אות ברור וקליטה איכותית יותר.

DLL (Dynamic Link Library) - קובץ המכיל קוד מהודר, נתונים ומשאים שיוכולים לשמש מספר תוכנות במקביל.

API (Application Programming Interface) - ממשק תוכנות יישומיים - אוסף פקודות, הגדרות וכליים המאפשרים לתוכנות שונות לתקשר ולשתף מידע ביניהן.

Max Hold - תצוגה בספקטרום אנליזור המראה את ההיסטוריה של הפיקים בעוצמת התדרים.

SPI (Serial Peripheral Interface) - פרוטוקול תקשורת טורי בין התקנים.

LNA (Low Noise Amplifier) - מגבר רעש נמוך.

Crystal Oscillator - רכיב אלקטרוני הפועל באמצעות גביש קוורץ, לייצור אות שעון מדויק ויציב למעגלים דיגיטליים ומשדרים.

PAD (Printed Circuit Pad) - נקודת מגע מוליכה על גבי מעגל מודפס (PCB), המשמשת להלחמת רכיבים.

DC (Direct Current) - זרם ישיר, כמו למשל זרם חשמלי קבוע בכיוון ובעוצמה, כמו זה שמספקת סוללה.

SMA (SubMiniature version A) - סוג מחבר RF קטן ומוגבר.

Datasheet - גיליון נתונים טכני של רכיב אלקטרוני, הכול מודיע חינמי ומשמש מהנדסים לתכנון ושיילוב הרכיב במערכת.

6.3 מטרות, יעדים ומדדים

מטרת הפרויקט

פיתוח המערכת מתמקד ביצירת פתרון מתקדם לנטרול וחפניהם המהווים איום על אזורי מוגדרים. המערכת נועדה לנטרל וחפניהם שמנסים לתקוף או לחדר לשטח מוגן, וזאת באמצעות שיבוש או חסימת התקשרות האלחוטית שבין הרחפן למפעיל.

יעדי הפרויקט

- בניית דגם MVP (Minimum Viable Product) אשר מסוגל לחסום את התקשרות בין הרחפן למפעיל.
- יצירת חסימה בלפי השמיים וככלפי האופק תוך שמירה על טווח הגנה מתאים, ופגיעה מינימלית בסביבה בעזרת אנטנות ביוניות.
- יכולת עבודה עם מתח DC.
- יכולת שליטה ובקרה באמצעות פלטפורמה ניידת על היחידה של החומר.

מדדי הפרויקט

- שיבוש שלושת ערוצי תקשורת: GPS, 5.8 GHz, 2.4 GHz.
- שידור מתאים בהספק לפחות [W] 100mW.
- המערכת תהיה מותאמת למתח DC.
- המערכת יהיה ממשק GUI שנשלט בצורה נוחה למשתמש.

7 ניתוח חלופות

7.1 תוכנתית

C++	CVI	MATLAB	Python	מדד
5	5	4	4	aicots ביצוע
4	3	4	5	התממשקות
5	3	5	4	יעילות מימוש
5	2	3	4	עלות
19	13	16	17	סה"כ

טבלה 1 - חלופה תוכנתית.

C++: שפה מהירה ויעילה מאד, מכיוון שהיא מהודרת (Compiled) ישירות לקוד מקונה, זה הופך אותה למתאימה במיוחד לעיבוד אותות ולשליטה בחומרה הדורשים דיק בזמנים וביצועים גבוהים. מאפשרת שליטה ישירה במכשירי חומרה כמו SPI, I2C ו-UART דרך ספריות.

CVI: מצטיינת בהתממשקות עם חומרה ובקשה בזמן אמיתי, מה שהופך אותה לאידיאלית לשילטה במערכות חומרה מורכבות כמו בפרויקט שלנו.

Python: מציעה גמישות רבה עם ספריות רבות להתממשקות עם מגוון רחב של רכיבים, מה שמאפשר התאמה מהירה לסביבת הפיתוח.

MATLAB: מספקת כלים מתקדמים לעיבוד נתונים וסימולציה, עם יכולת הtamמשקות ישירה לחומרה, מה שהופך אותה לבחירה חזקה ואמינה לפרויקט.

על פי תוצאות הבדיקה ובדיקה מעמיקה האופציונות הטובות יותר לביצוע הפרויקט הם, C++ או Python מכיוון ששתי השפות מציעות יתרונות ייחודיים המתאיםים לדרישות שלנו. C++ מספקת ביצועים גבוהים ושליטה ישירה בחומרה דרך פרוטוקולים מה שהופך אותה לאידיאלית למשימות קריטיות בזמן אמיתי. לעומת זאת, Python מציעה גמישות ופשטות בפיתוח עם ספריות מוכנות, מה שמאפשר חיבור מהיר ונוח למערכות כמו GNU Radio ולחומרה חיצונית.

7.2 מערכתי

Smart jamming	Spot jamming	Barrage Jamming	Sweep jamming	מדד
2	4	3	3	קלות מימוש
4	2	5	4	תפיסת תדרים
5	3	3	5	יעילות
2	3	5	4	עלות
4	5	2	4	עוצמת הספק
17	17	18	20	סה"כ

טבלה 2 - חלופה מערכתי.

בפרויקט, חלופת Sweep Jamming היא שיטה חכמה ויעילה לנטרול רוחניים המשתמשים ב Frequency Hopping. שיטה זו פועלת על ידי מעבר דרך תחום תדרים שנבחר מראש, תוך שימוש במומת הספק אחת בקפיצות קטנות בין התדרים שהוגדר מראש.

בנוסף, Barrage Jamming היא יعلاה מאוד לחסימת רוחניים, טכניקת שיבוש זו מתמקדת בשידור אותות בתדרים רבים בו-זמנית על פני טווח רחב של תדרים. המערכת אינה מתמקדת בתדר מסוים או באוט ספציפי, אלא מפיצה הספק על פני טווח תדרים רחב. עם זאת, הספק הכלול מתחלק בין כל התדרים בהם משדרים, מה שambilא לשידור הספק נמוך יותר לכל תדר בנפרד.

על מנת להשתמש בשיטה זו ולשדר בהספק גובה אנו יכולים לחרוג מהתקציב. באמצעות Sweep Jamming ניתן להשיג תוצאה דומה אך עם שידור הספק גובה יותר לכל תדר, מה שմגדיר את יעילות החסימה.

שיטת Spot Jamming לעומת זאת, היא הפעחות יعلاה לחסימת רוחניים. זאת מכיוון שהיא מתמקדת בחסימת תדר בודד ושידור מירב ההספק בתדר זה. הבעה בשיטה זו נובעת מהעובדת שאין בידינו מידע מדויק על התדר שבו מתקשים הרחפן ומפעילו, ולכן יש סבירות גבוהה שהחסימה לא תהיה אפקטיבית.

ב-Smart Jamming, הניסיון הוא להתמקד בתדרים מסוימים או בפרוטוקולי תקשורת ספציפיים של הרחפן, מה שדורש מידע מדויק על תדרי השידור של המטרה. במקרה של רוחניים שמשנים את תדריהם או משתמשים בתדרים לא ידועים, Smart Jamming עשויה להיות פחות יעילה מאשר sweep jamming.

7.3 טכנולוגיות

למי 2594 EVM (sweeper)	למי 2492 EVM (sweeper)	LimeSDR mini 2.0 (SDR)	HackRF one (SDR)	מדד
2	3	4	5	עלות
3	4	1	2	טוווח תדרים
5	4	5	2	רוחב פס
4	4	4	4	קלות שימוש
4	4	5	5	תמיכה בממשקים
18	19	19	18	סה"כ

טבלה 3 - חלופה טכנולוגית.

למי 2594 EVM: רכיב זה הוא הבחירה המובילה לפROYיקט חסימת רחפנים בשיטת Sweep jamming בזכות יכולת שלו לבצע שינויים מהירים בתדרים על פני טווח רחב (10 מגה-הרץ עד 15 גיגא-הרץ). הוא מתאים במיוחד לפROYיקט אשר דורש דיקוק גבוהה ומהירות שינוי תדרים, כמו חסימת רחפנים שימושיים בתדרים במהירות.

למי 2492 EVM: רכיב זה מספק גמישות גבוהה יותר בזכות האפשרות לחבר לו VCO חיצוני, מה שמאפשר התאמת אישית לטווח תדרים ייחודי. הוא מתאים לפROYיקט אם יש צורך בהתאם אישית של sweeping או יצירת אוטות מורכבים.

LimeSDR Mini 2.0: מציע גמישות רבה בפרויקט בזכות יכולת לתוכנת אותו באופן מותאם אישית בעבר Barrage jamming ו-Sweep jamming. הוא מכסה טווח תדרים רחב, אך מהירות שינוי התדרים איטית יחסית לרכיבים ייעודיים כמו ה-LMX2594.

HackRF One: רכיב SDR פשוט לשימוש, אך עם מגבלות מבחינות מהירות שינוי תדרים. הוא מתאים יותר לתוכנות בדיקה ולפרויקטים שבהם sweeping איטי הוא מספיק ולא לאפליקציות הדורשות דיקוק ומהירות גבוההים. הוא מתאים לשימוש גם בשיטת Barrage jamming.

8 סקירה ספרות, סקר שוק

8.1 סקירה ספרותית

.1

Vladimir Matić, Vladimir Kosjer, Aleksandar Lebl, Branislav Pavić, Jovan Radivojević

"Methods for Drone Detection and Jamming", 2019

מאמר זה מציג תחיליה את הטכניקות העיקריות ליזיהו ורchapנים. לאחר מכן, הוא מדגיש את הנזקודות החשובות להצלחה בחסימת ורchapנים. הטכניקות ליזיהו ורchapנים כוללות שימוש ברדאר, RF, חיישנים אקוסטיים, אופטיים ותרכמיים (אינפרא-אדום). הטכניקות הללו מושווות זו לזו, תוך הדגשת היתרונות והחסרונות של כל אחת מהן. לא קיימת טכניקה ברווחה שמצויה יתרון מובהק על פני האחרות, ולכן נעשה שילוב של מספר טכניקות לפתרון כולל. בתחום החסימה, הטכניקות המשמשות הן טכניקות חסימה מסורתיות, כאשר sweep jamming - barrages jamming הן הנפוצות ביותר. המטרה העיקרית היא לנטרל את יכולת GPS של הרchapן, מה שמצוח כדרך היילה ביוטר לחסימה.

.2

PIETRO TEDESCHI , GABRIELE OLIGERI and ,ROBERTO DI PIETRO

"Leveraging Jamming to Help Drones Complete Their Mission" January 8, 2020

המאמר מספק מידע מקיף על ערכו התקשורתי שבין הרchapן למפעלו והתדרים השונים בהם משתמשים כל טיס בלתי מאוישים. המאמר סוקר שיטות חסימה שונות ומתקדם במיוחד בטכניקת חסימה המתאימה לצרכים של הפרויקט שלנו. הטכניקה המדוברת מבוססת על שידור סיגナル בתדרי הפעולה של הרchapן. בנוסף, המאמר מתאר את סוג האנטנות הדורשות לימיוש החסימה, כמו אנטנות ביוניות. כמו כן, ניתן למצוא במאמר הסבר תיאורטי על החשיבות של המרחק בין החוסם לרchapן, ועל עצמת השידור הנדרשת לחסימה אפקטיבית.

.3

Mladen Mileusnić, Branislav Pavić, Verica Marinković-Nedelicki, Predrag Petrović, Dragan Mitić, Aleksandar Lebl

"ANALYSIS OF JAMMING SUCCESSFULNESS AGAINST RCIED ACTIVATION WITH THE EMPHASIS ON SWEEP JAMMING" June 2019

המאמר עוסק בניתוח הצלחת חסימה של מכשירי נפץ מאולתרים המופעלים מרוחק (RCIED), תוך דגש על טכניקת חסימה באמצעות סריקת תדרים (sweep jamming). המאמר משווה בין ביצועים של חסימה פעליה באמצעות רעש וחב-ספקטרים לבין סריקת תדרים, ומתקדם במאפיינים של סריקת תדרים כמו מהירות הסריקה, תנאים להצלחה וניתוח הסתברות להצלחה. המאמר מציג גם חישובים להסתברות חסימה מוצלחת, עם דגש על התחום העיקרי של ימיוש סריקת תדרים במכשירי חסימה. המחקר מתיחס לחשיבות של מהירות הסריקה והצורך בהתאם לנכונה של מכבי החומרה כדי להבטיח חסימה יעילה. בנוסף, ניתנים נתונים על השפעת תדרי ההתחלה והסיום של הסריקה על הצלחת החסימה, יחד עם ניתוח של יכולת לתיקן שגיאות בהודעת הפעלה.

8.2 סקר שוק



Pitbull 100/101 של חברת MyDefence הוא חסם ורחפנים לביש שמיועד לכוחות צבא ומשטרה. המבשיר פועל בתדרים של 1.6GHz, 2.4GHz, 5.2GHz, 5.8GHz, ומסוגל לשבש ורחפנים בטוח של עד 1000 מטרים. הוא מצוי באנטנות ביוניות ומאפשר פעולה אוטומטית או ידנית. המבשיר קל משקל 1.3Kg ועוצב כדי להפחית עומס קוגניטיבי על המפעיל, תוך מתן עדכונים שוטפים לשיפור ביצועים. המוצר זהה מתקשר לפרקט שלנו בכך שהוא מספק פתרון לחסימת תדרים של רחפנים באמצעות טכנולוגיית שיבוש מתתקדמת.



ND-BO004 Omnidirectional של NQDefense הוא חסם ורחפנים המספק כיסוי של 360 מעלות, עם יכולת לשבש תדרים בטוח של עד 1.5 ק"מ. הוא פועל במספר תחומי תדרים, כולל 2.4GHz, 5.8GHz, GPS, ומיועד להגנה על אזורים קריטיים.



DDS Handheld UAV Jammer של חברת AARTOS הוא מבשיר נייד לשיבוש ורחפנים. הוא מכסה ארבעה תחומי תדרים, עם טווח חסימה של עד 2 ק"מ. המבשיר שוקל כ-2.7 ק"ג ומספק הספק שידור של W40. הוא עמיד לתנאי סביבה קשים, עם טמפרטורות פעולה בין -20°C ל-50°C, וכולן אנטנה ביונית. זמן הסוללה של המבשיר הוא עד שעתיים, מה שהופך אותו לפתרון נייד ויעיל לשיבוש ורחפנים במצב חירום ותנאי שטח מגוונים.

להלן טבלה המסכמת את המפרטים של המוצרים הקיימים בשוק:

מוצר	תדרי חסימה	דוית חסימה	זמן עבודה	הספק שידור	משקל	טווח חסימה
Pitbull 100/101	2.4GHz, 1.6GHz, 5.2GHz, 5.8GHz	60°	20 שעות בהתנה, עד שעתיים בשידור	IP67	1.3 Kg	1 Km
ND-BO004 Omnidirectional	410-440MHz, 840-930MHz, L1GPS 1.57GHz, L2 GPS 1.22GHZ, 2.40-2.50GHz, 5.70-5.90GHz	360°	עד שעתיים	IP65	30 W	1.5 Km
DroneNet Jammer	2.40-2.50GHz, 1.57-1.62GHz, 5.70-5.90GHz, 865-930MHz	60°	עד שעתיים	IP65	40 W	2 Km

טבלה 4 - סיכום המוצרים הקיימים בשוק.

כפי שניתן לראות, קיימים חוסמי רחפנים מגוונים בשוק הנבדלים זה מזה בתדרי החסימה, טווחי החסימה, משקלם והכיניות שלהם. במוצרים שבדקנו, ישנו גם הבדלים בהספק השידור וזמן הפעולה ללא טעינה. עם זאת, המוצרים שנמצאו הם לרוב מתקדמים ויקרים.

בפרויקט, אנו מצאים דרך פשוטה וולגה וחסית למימוש חסם שכדי לשקון כאשר ניגשים לפיתוח של מוצר זהה.

הdagש בפרויקט הוא על יצירת חסם יעיל עבור שלושה ערכז'י חסימה עיקריים, תוך שימוש במוגביי הספק המתאימים לתקציב מוגבל.

למרות הבעיות הנמוכה, המערכת תספק פתרון אפקטיבי לחסימת רחפנים, במיוחד עבור לקוחות שמחפשים חסימה של ערכזים ספציפיים וממוקדים.

9 שיטות (Methodology)

9.1 תכנן ותכנון המערכת (Detailed planning)

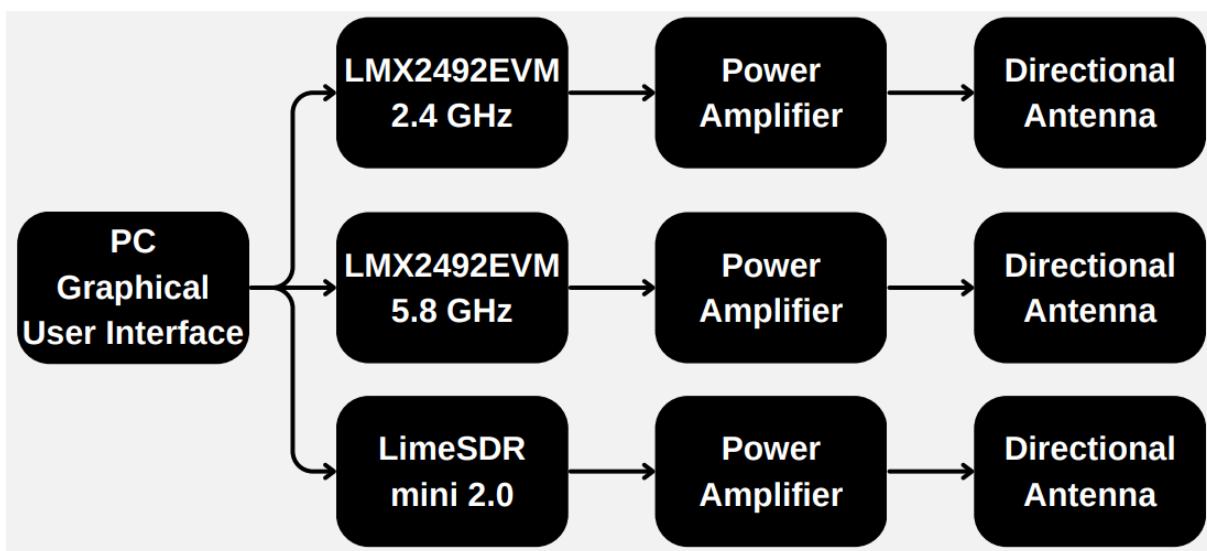
מטרת הפרויקט היא לחסום מספר ערוצי תקשורת של רחפן, באמצעות שילוב של שני רכיבים מרכזיים: LimeSDR Mini 2.0 ו-LMX2492EVM. תפקודם ודרך שילובם בפרויקט יפורטו בהמשך פרק זה. לצורך הפעלה יציבה ויעילה של המערכת, ישולבו גם רכיבים חיווניים נוספים: בבל RF, אנטנות, מגבר הספק וכן ממשק משתמש גרפי.

לצורך חסימה אפקטיבית היה علينا לשאוף לשידור בעוצמה גבוהה ככל האפשר, במטרה להויריד את יחס אות לרעש (SNR) במקלט הרחפן. הבנה זו נשענת ישרות על עקרונות משווה Friis, אשר ממחישה את השפעת עצמת השידור והמרחק על עצמת האות המתתקבל. מתוך המשווה עולה כי שני פרמטרים מרכזיים עומדים לרשوتנו: הספק השידור והמרחק לרחפן. ככל שהספק השידור גבוה יותר, כך גדל הסיכוי ששידור החסימה יגבר על אות הששלט ויגרום להפחיתה ה-SNR במקלט הרחפן לרמה שתוביל לשימוש מוחלט של התקשורת שלו. מבונן שבכל שהמරחק בין המפעיל לרחפן גדול או שהוא קטן בין החסום לרחפן קטן, החסימה תהיה יעילה יותר.

9.1.1 תרשימים בלוקיים

תרשימים הבלוקיים אלה מציגים את הארכיטקטורה של המערכת שלנו:

- ממשק משתמש (GUI) - שלוט ומגדר את פרמטרי המערכת כגון טוווח תדרים וזמן הפעלה.
- ברטייס LMX2492EVM - משדר אות משולש לשימוש Sweep בתחום תדרי WiFi.
- ברטייס LimeSDR Mini 2.0 - יוצר אות רעש רחוב פס המיעוד לשימוש בשיטת Barrage בתחום תחומי GPS L1.
- מגברי הספק - מגבירים את אותות ה-RF מהמודולים של LimeSDR ו-LMX לרמת שידור נדרשת.
- אנטנות ביוניות - משדרות את האותות המוגברים בקרן ממוקדת לעבר הרחפן.



איור 1 - תרשימים בלוקיים

9.1.2 רכיב לביצוע חסימת WiFi

במסגרת הפרויקט, התחלנו עם שני רכיבי SDR שלקחנו מהմכלה HackRF One ו-LimeSDR Mini 2.0 מתק שיקולים כלכליים וטכנולוגיים. רכיבים אלו נועדו לספק פתרון טכנולוגי לחסימת הערכאים.

למדנו כי באמצעות תוכנת GNU Radio, המאפשרת תכונות של SDR באמצעות בлокים הניטנים לקינפוג וביילוב בתיבת קוד בפייטון, ניתן ליישם את שיטת sweep jamming. עם זאת, לאחר חקר מעמיק על תכונות HackRF One, גילינו כי זמן ההתייצבות של הרכיב בעת מעבר בין תדרים ($t_{tune}+t_{lock}$) הוא 215 מילוק-שניות - איטי מדי כדי לספק חסימה יעילה בשיטת Sweep jamming. זמן זה נובע בעיקר מהצורך למעבר דרך ה-PLL (Phase Locked Loop), שדורש התיאצבות על כל תדר חדש.

MB	DB	$t_{tune} + t_{lock}$ (μs)	t_{lock} (μs)	Max Avg (Hz)	Max Inst (Hz)
N210	WBX	510	290	500	2000
N210	SBX	483	289	500	2000
X310	UBX	610	418	500	2000
B100	WBX	782	573	500	1666
B100	SBX	772	546	333	1666
B200 ¹	- ⁴	3310	117	166	285
B210 ¹	- ⁴	3318	124	166	285
E310 ²	- ⁴	31258	285	28	31
HackRF ³	- ⁴	215	0	1428	5000

טבלה 5 - נתונים של HackRF

ניסינו לשקל פתרון שבו נעילת ה-PLL תהיה חד-פעמיות והתדרים ירוצו במצב open loop (לא PLL), אך אופציה זו אינה מתאפשרה ברכיב זה. בעקבות כך, החלפנו לוותר על שימוש ב-HackRF בשיטת sweep jamming.

לאחר מכן, עבדנו לבדוק את ה-LimeSDR Mini 2.0, אך נתקלנו בקושי להשיג מידע ברור על מהירות הקפיצה המקסימלית (frequency hopping rate) של הרכיב. אף על פי שהיחסנו במקורות אינטרנטיתים ואף פנינו לפורום הרשמי של הרכיב, לא קיבלנו תשובה חד-משמעות.

בהתיעצות עם המנכחה, הומלץ לנו לבחור רכיבים "יעדים" המכונים Sweepers מחברת Texas Instruments, שתוכננו במיוחד לביצוע Sweepjamming מהיר על פני תדרים.

בנוסף, הרכיבים ידידותיים למשתמש, כולל ממשק GUI וכן מתחמשים ביכולות עם המחשב באמצעות פרוטוקול SPI. כמו כן, קיימת סביבה קהילתית רחבה של משתמשים עם פורומים פעילים ותמיכה טכנית אמינה מצד היוצרים.

הבחירה המועדפת הייתה ה-LMX2594EVM, שמציע טווח רחב מאוד של תדרים (10MHz עד 15GHz) ויכולת מעבר מהירה ביותר בין תדרים בטוחה הנבנה.

לאחר עיון עמוק ב-Datasheet של הלוח והתייעצות עם מהנדסי TI, הגיענו למסקנה כי רכיב ה-LMX2594EVM אינו מתאים לשימוש שיבושים ורchapנים בטכניקת Sweep Jamming בפרויקט שלנו, וזאת מספר סיבות עיקריות:

1. האטה בקצב הסריקה:

המעבר בין תדרים נזভים לגבוהים נעשה באמצעות מחלקי תדר מה שモוביל לירידה דרמטית ב מהירות הסריקה בפלט - תוצאה שאינה עומדת בדרישות הביצועים של המערכת. בפועל, סריקה של תחום

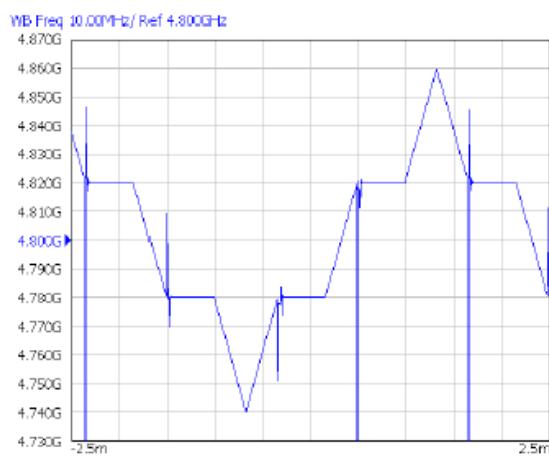
תדרים של 2.4-2.485GHz נעשית על ידי סריקה של 340MHz בתחום של 9.6-9.94GHz וכן נוצרת האטה ניכרת.

2. מגבלת טווח סריקה לינארית:

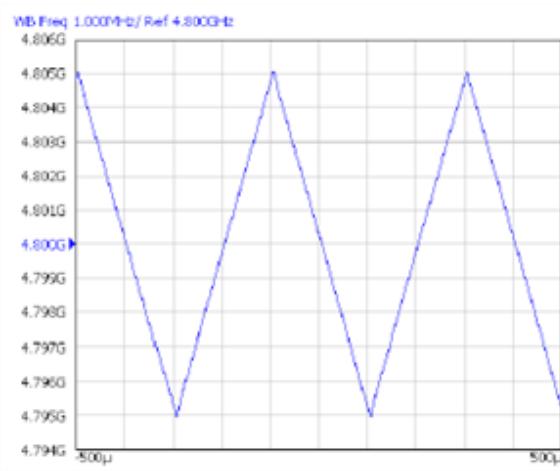
הרכיב תומך בסריקה לינארית של עד 100MHz בלבד. טווחים רחבים יותר מחייבים כיול חדש של ה-
CO טור כדי פועלה - דבר שפוגע ביציבות ובמהירות הסריקה.

3. עיונות באוט עקב חסר לינאריות:

המחלקים יוצרים עיונות באוט (non-linearities) כאשר מנסים לבצע סריקה על טווח תדרים רחב יותר. בתוצאה מכך, לא מתאפשר sweep אחד ויעיל, מה שפוגע ביעילות השימוש בפועל.



גרף 1 - עם עיונות.



גרף 2 - לינארית.

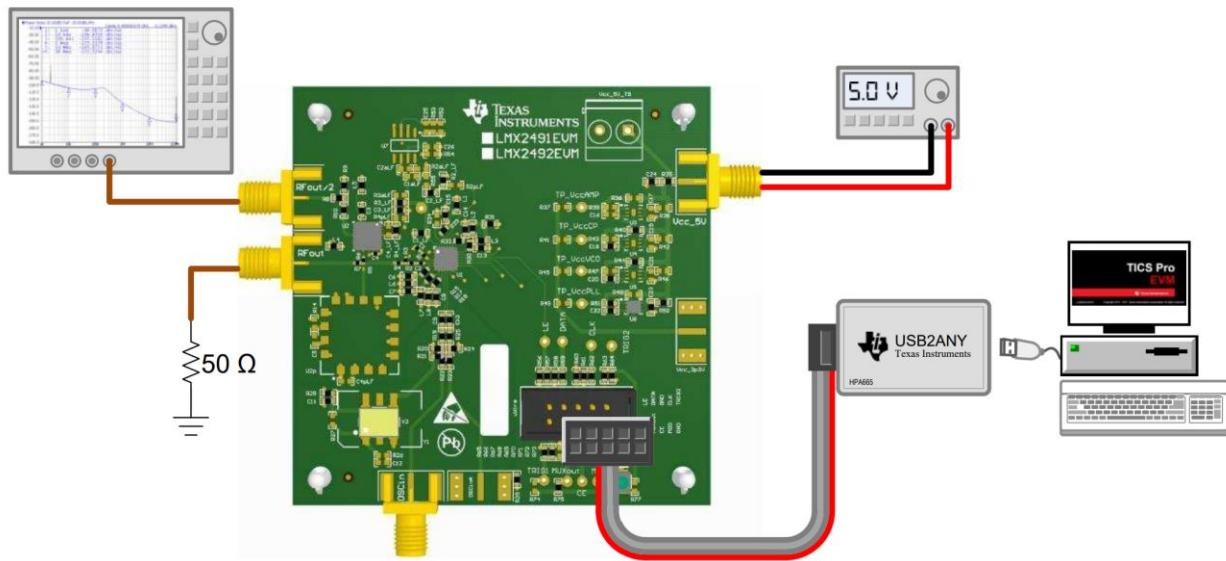
באמצעות הגرافים ניתן להמיחס את מגבלות רכיב ה- LMX2594 ביצוע סריקת התדרים:

גרף (1) משמאל סריקה בטוחה רחב יותר של ~100MHz כבר מראה תוצאה איטית בהרבה, עם עיונות בולטים באוט תוצאה ישירה מהשימוש במחלקתי תדר והצורך בכליים פנימיים של ה CO.

גרף (2) מימין סריקה בטוחה של 10MHz בלבד מציגה תוצאה מהירה ולינארית, עם שיפור אחד ויציב - בהתאם למגבלי הרכיב המתוארות ב-Datasheet.

בשל כך, נאלצנו לוותר על רכיב זה לטובה **LMX2492EVM** אשר נמצא מתאים ביותר לשימוש sweep מהיר ואחד על פני טווח תדרים רחב במיוחד. היתרון המרכזי שלו הוא יכולת לשמר על לינאריות גבואה ועקבית לאורך כל טווח הסריקה, גם כשמדבר בעשרות ואף מאות מגהץ, שהיא תכונה קריטית עבור שימוש אפקטיבי של מערכות תקשורת של וחפנים.

:LMX2492EVM



איור 2 - תרשימים חיבורים LMX2492EVM

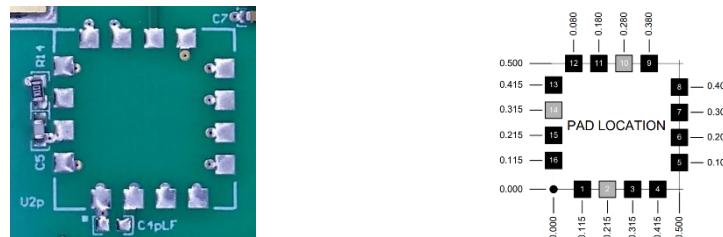
הסבר כללי על המוצר ועל אופן פעולתו

ברטיס זה מתפרק בסוויפר (Sweeper), שתכליתו ליצור אוטות Ramp-Chirp. הוא פועל במתחה כנסה של 5V ומאהר שחלק מהרכיבים על הלוח דורשים מתח נמוך יותר של 3.3V, מושולב בברטיס ממיר ODO מוגנה שתפקידו להמיר את מתח הכנסה למתחים הנדרשים עבור כל אחד מהרכיבים. הברטיס כולל בתוכו שעון פנימי בתדר **100MHz**, המשמש לשנסטרון פעולת המערכת. לברטיס קיימות שתי יציאות RF - אחת סטנדרטית ואחת מחולקת ב-2 (RF/2), על מנת לאפשר גמישות ולהעניק למשתמש אפשרות לחבר מספר יציאות במידה הצורך, במקרה שימוש ביציאה אחת בלבד נדרש חיבור נגד עומס של 50 ohm ליציאה שלא בשימוש לצורך מניעת החזרות. ה-VCO המוגנה בלוח תומך בתחום תדרים שהינו $10.8\text{ GHz} - 9.4$ אשר הוא תחום שאינו תואם אתדריות השידור של הפROYיקט. עם זאת, הלוח מספק אפשרות לחבר OCO VCO חיצוני באמצעות מיקום ייעודי, ובכך ניתן להתאים אותו לשימוש הרצוי במערכת השיבוש.

בשלב הראשון יהיה עלינו לאתר OCO VCO חיצוני הפועל בתחום התדרים הרלוונטי למערכת שלנו. לשם כך ביצענו בדיקה של מיקום הפדים (Location Pads). על פי דף הנתונים של ה-LMX2492EVM, נדרש VCO מסווג CVCO55 פורמט מוגדר מראש של רכיב התואם מבחינות מאפיינים חשמליים ומכניים. לשם כך פנינו לאתר של חברת CRYSTEK המתמחה ברכבי תה, ואיתרנו VCO אופציונליים התואמים לדרישות.

- VCO לתחום 5.8GHz - CVC055CXT-5700-5900
- VCO לתחום 2.4GHz - CVC055CC-2328-2536

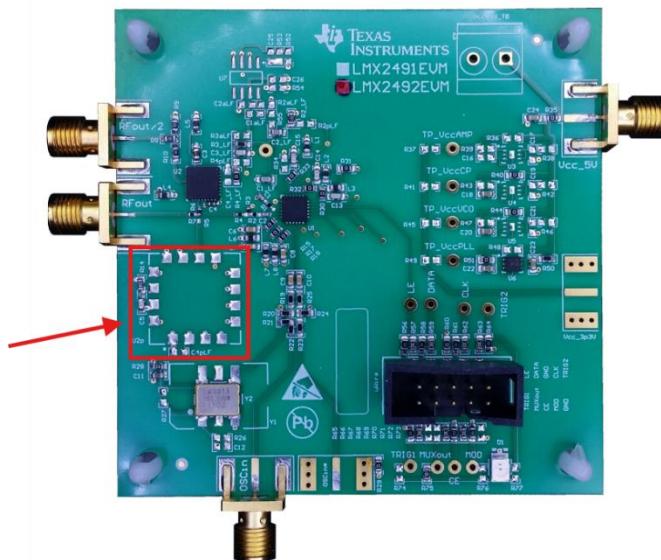
בנוסף, ניתחנו את סידור הפינים והחיבורים של ה-pdms כדי לוודא התאמה מלאה מבחינת קווי אוט, הזנת מתח והארקה



איור 3 - סידור הפינים של ה-VCO החיצוני - Location Pad של ה-VCO המידעד.

לאחר בדיקה מעמיקה של ה-datasheet של הרכיב והטייעצות עם מהנדסי Texas Instruments גילינו שכדי לשלב O VCO חיצוני התואם לדרישות התדר שלנו, לא ניתן להסתפק בהלחמת ה-VCO בלבד. יש צורך לבצע התאמות נוספות בمعالג - היכולות החלפה, הוספה או הסרה של רכיבים פסיביים כגון נגדים, קבלים וסלילים, על מנת לאפשר הפעלה תקינה של ה-VCO החיצוני במקומו של ה-VCO המובנה ולוודא שהאות עובר במסלול הנכון מבחינת מדר והזנת מתח.

המיקום בו ה-VCO החיצוני צריך להיות מולחן בمعالג:



איור 4 - מיקום ה-VCO החיצוני על ה-VM-EVM

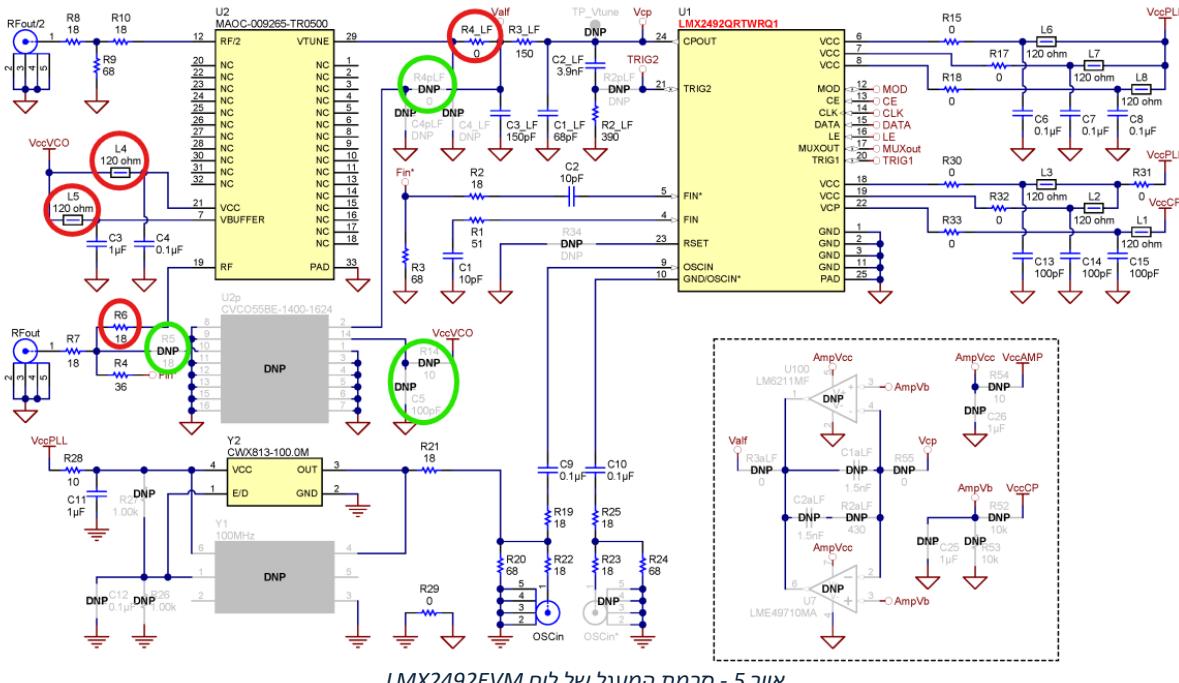
הנגדים והקבלים שמשתתפים בשינוי המمعالג:

ניתוק

- R6: נגד שמחבר בין ה-VCO הפנימי ליציאת ה-RF.
- L5,L4: מחברים בין ה-VCO הפנימי למתח Vcc. נדרש נתק בשני המיקומים על מנת למנוע מה-VCO הפנימי לקבל מתח.
- L4_LF: נגד שמשמש בקצר בין ה-Loop Filter ל-VCO הפנימי.

חיבור

- דרוש על מנת להעביר את מוצא ה-VCO ליציאת ה-RF.
- דרושים על מנת להעביר בראוי את מתח ההפעלה ל-VCO החיצוני.
- נגד המשמש בקצר בין ה-Loop Filter ל-VCO החיצוני.

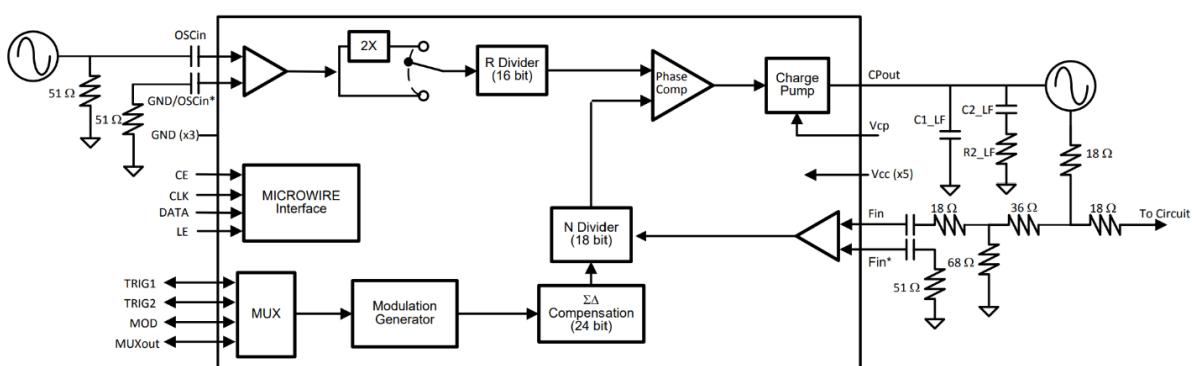


הרכיבים המוקפים באדום נוטקו מהמעגל, הרכיבים שמקופים בירוקים הולחמו במעגל.

LMX2492

הסיבה שבחרנו לרכוש את הרכיבים זהה היא בזכות יכולת הפונקציונליות המרכזית שלו - ביצוע סריקת תדר (Sweep) על פני תחום תדרים נרחב, בהתאם לדרישותuproject. יכולת זו מתאפשרת בזכות הרכיב המרכזי שעליו מבוסס הלוח LMX2492, שהוא מחולל תדרים מתקדם מסוג PLL רחוב-פס, התומך ביצירת סריקות מדיקות ומהירות.

The LMX2492/92-Q1 can be used in a broad class of applications such as generating a single frequency for a high frequency clock, generating a tunable range of frequencies, or generating swept waveforms that can be used in applications such as radar.



איור 6 - סכמת מעגל LMX2492

מבנה הרכיב מבוסס על לולאת נעה פאזה (PLL) אשר כוללת את הרכיבים הבאים: משווה פאזה, משאבת טעינה מבוססת על זרם, מסנן מעביר נמוכים ו- VCO חיצוני המחבר בlolאת משוב. כל אחד מהרכיבים הללו ממלא תפקיד קרייטי בהפעלה התקינה של המערכת.

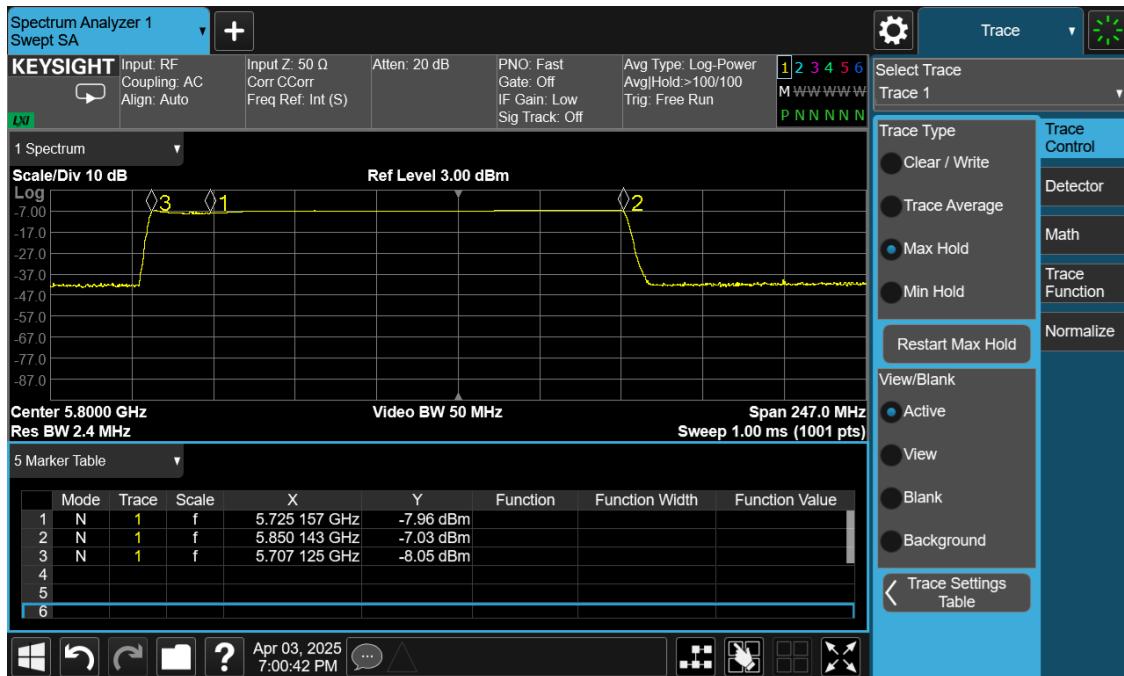
- Phase Detector - מזאה הפרשים בין אות הייחוס (Oscillator) לאות המתקבל מה-VCO והופך אותו לפולסי מתח.
- Charge Pump - ממיר את פולסי המתח לפולסי זרם יחסיים להפרש הנמדד על ידי גלאי פאזה.
- External Loop Filter - הקבלים בלולאת הסינון נטענים על ידי פולסי הזרם ובכך מייצרים את מתח הייחוס לבנisa של ה-VCO.
- VCO - מתנד המפיק תדר המשטנה בהתאם למתח ההזנה. ככל שמתנה הבנisa משתנה - כך משתנה גם תדר היציאה של המערכת.

במהלך מצב נעה, לולאת ה-PLL פועלת לשמירה על תדר קבוע ויציב. כאשר מתגלה הפרש בין אות הייחוס לאות המתקבל מה-VCO מתבצע תיקון דרך משאבת הטעינה, וה-VCO משנה את תדרו עד לאיזון. לצורך יצירת סריקת תדר, נעשה שימוש במודול פנימי לייצור גל משולש. אות זה מוזן לגלאי הפאזה וגורם לכך שבכל רגע מתקבל מתח אחר ל-VCO. שינוי זה יוצר תדר משתנה במחריות ובכך מתבצעת סריקה על פני תחום תדרים רחב. חשוב לציין כי במצב Sweep המערכת אינה שומרת על נעה (Open Loop) אלא מבצעת סריקה חופשית ללא תיקון אוטומטי של ההפרש.

בחירה גל משולש על פני גל מסור

לייצור האות המשבש עמדנו בפנינו שתי אפשרויות, יצירת גל משולש או גל מסור. לאחר בדינה של שתי האותות, מוכחנו לגלוות שכארטר משתמשים בגל מסור, הספקטrometer "מתלבך" וישן הפרעוט שצלגות מהתחום אותו הגדרנו. מכיוון שביצירת הגל אנו سورקים את התוחום מהתדר ההתחלה עד התדר הסופי ומיד חזררים להתחלה, ההתיצבות המהירה שצריכה להתבצע על ידי הרכיב, גורמת לмерיחת האות על פני תחום תדרים לא רצוי. באשר משתמשים בגל משולש, התוחום נסרק מהתדר ההתחלה כלפי מעלה לתדר הגבואה ואז מהתדר הגבואה כלפי מעלה באופן רציף ללא קפיצות חדות.

להלן דוגמה לмерיחת הספקטrometer בתחום 5.8 GHz כאשר השתמשנו בגל מסור:



גרף 3 – Zligah בשימוש גל מסור.

ניתן לראות את הצלגה ב- 18 MHz בتردد הנמוך.

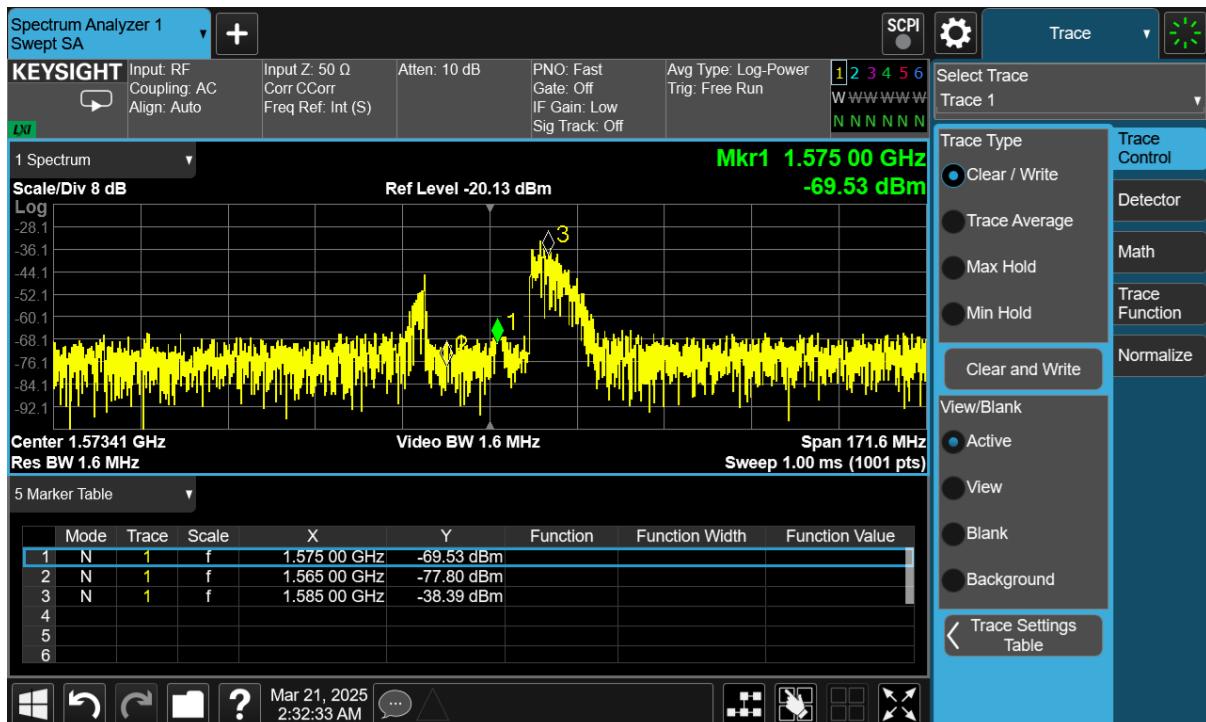
9.1.3 רכיב לביצוע חסימת GPS

החלנו לבצע את חסימת ערז GPS באמצעות אחד מ-SDRs בשיטת Barrage jamming, לאחר שהם אינם מהירים מספיק ל-Sweep jamming אך מסוגלים לספק רוחב פס רחב יחסית של אות המוצא. חסימת GPS אינה דורשת יותר מ-20MHz סיבוב התדר המרכזי ועוצמת האות של ה-GPS חלשה בשל המרחק מהלוין ובכך הופכת את הבעה לנוחה לפתרון גם בעוצמת שידור נמוכה.

שקלמו בין שני רכיבים נפוצים בשוק: LimeSDR mini 2.0 HackRF One. לאחר בדיקה מעמיקה החלנו לבחור LimeSDR mini 2.0 מכמה סיבות טכניות המKENOT ליתרונו גם בעוצמת שידור נמוכה.

רוחב הפס המקורי של LimeSDR הוא 40MHz לעומת 20MHz בלבד של ה-HackRF. בשמה דבר GPS, נדרש לכיסות תחום תדר מסוומי, וכן הגמישות והיכולת של LimeSDR לשדר אות רוחב יותר היו קריטיות עבורנו.

לבסוף, מה שהכריע מבחרינו היה העובדה כי לאחר בדיקה של הרעש המשודר של שניהם בספקטרום אנליזה ראיינו כי ל-HackRF קיימות נPILEות רבות ועוצמת האות אינה אחידה לאורך רוחב הפס, בעוד ב-LimeSDR התקבל אות אחיד בהרבה, עם יציבות ספקטרלית משמעותית טובה יותר.



.HackRF ,Spectrum Analyzer - 4 גראף

ניתן לראות נPILEה משמעותית של הרעש לריצפת הספקטרום. תופעה זו מוסברת בכך במוגבלות החומרה של ה-SDR והן בדרכם נוצר הרעש הדיגיטלי בתוכנה, אך העובדה של LimeSDR יש יכולת להתמודד עם מגבלות אלו באופן טוב יותר מאשר חסימה של ה-SDR.

לכן, מתוך השיקולים הללו החלנו להשתמש ב-LimeSDR כדי ליצור barrage jamming להסימת GPS.

LimeSDR mini 2.0

LimeSDR Mini 2.0 הוא מקלט ומשדר תוכנה (SDR) קטן, רב-תכליתי, שמאפשר קונפיגורציה ותכונות דינמיים של מערכות תקשורת אלחוטיות בטוחה תדרים רחוב. הוא פועל בתחום תדרים של 10 MHz עד 3.5 GHz ותומך ברוחב פס של עד 40 MHz. לברטיס יש שעון 40 MHz והוא תומך בקצב דיגיטלי של 30 MSPS ושבי ערוצי תקשורת, אחד לקליטה ואחד לשידור. עצמת השידור שלו יכולה להגיע ל-10 dBm ותלות בתדר בו משדרים. הכרטיס משלב בתוכו שני רכיבים עיקריים: LMS7002M, שהוא בעצם המשדר-מקלט שלו, וFPGA אשר אחראי לשילטה ובקרה על כל תפקודיו הרכיב. שני הרכיבים מתקשרים ביניהם באמצעות פרוטוקול תקשורת SPI עליון נרחב בהמשך.

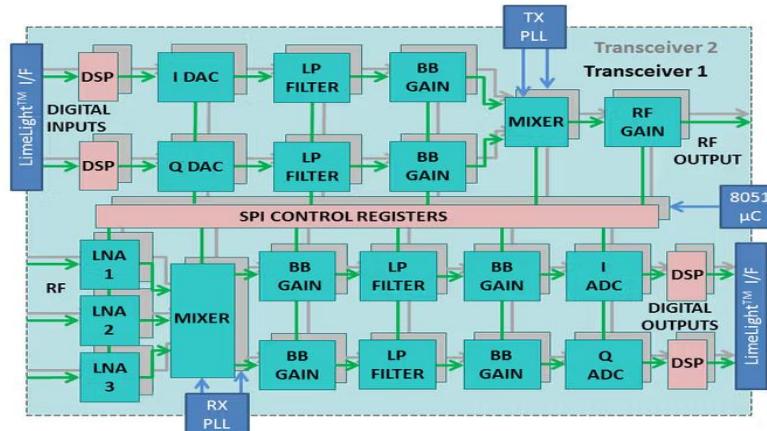


איור 7 LimeSDR Mini 2.0 - 7

LMS7002M

LMS7002M הוא רכיב טרנסיבר (משדר-מקלט) אינטגרטיבי, רב-תחומי, המאפשר עבודה במגוון רחב של תדרים סטנדרטיים, ומותאם במיוחד למערכות תקשורת מתקדמות. מדובר ברכיב שמאפשר קליטה ושידור של עד שני ערוצים בו זמינות (O2x2 MIMO) בטוחה תדרים של 100 קילו-הרץ עד 3.8 גיגה-הרץ, עם אפשרות לתכנות בזמן אמת של כל הפרמטרים התפעוליים. השבב כולל ממירים דיגיטליים-אנלוגיים (DAC) וממיריהם אנלוגיים-דיגיטליים (ADC) פנימיים ברזולוציה של 12 ביט, מספק רוחב פס של עד 160 מגה-הרץ במנשך האנalogique, ותומך גם במנשכים דיגיטליים עם רוחב פס של עד 96 מגה-הרץ.

הארQUITטורה של LMS7002M מבוססת על קליטת וSSIDור אותות שירות בסיס תדר (Zero IF) תוך שימוש במסנן פנימיים ניתנים לתכנות, מגברי רעש נמוך, מגברי הספק, מערכות ביול פנימיות ומיקרו-בקר מוגבנה שמנהלה תהליכיים כמו ביול, שליטה ותזמון. הוא כולל שני מסלולים נפרדים של PLL (אחד לשידור ואחד לקליטה), תומך במצבים FDD וTDD ומציע ממשק לחיבור דיגיטלי של אותות IQ עם תמיכת פרוטוקולים סטנדרטיים.



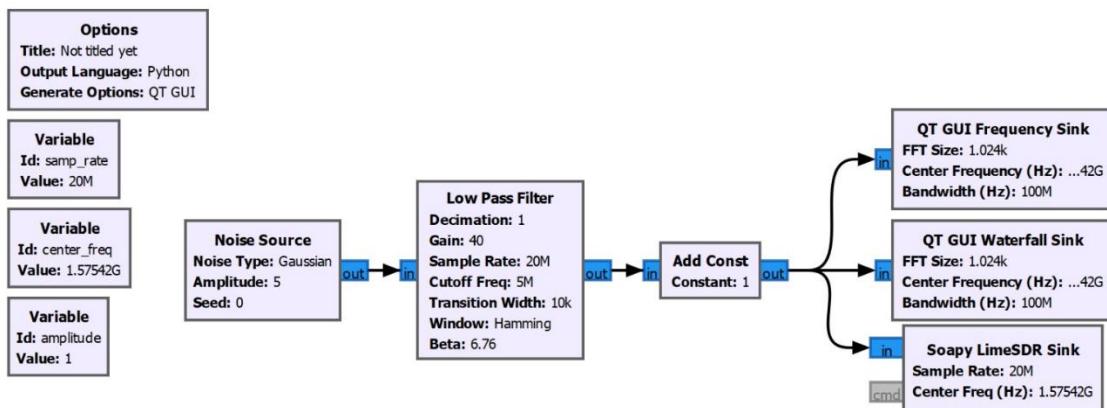
איור 8 - תרשימים בלוקים LMS7002M

Lattice ECP5 LFE5U-45F FPGA

FPGA Lattice ECP5 LFE5U-45F מודם LimeSDR Mini 2.0 הוא רכיב תכני מתקדם המשולב בלוח ומשמש כמרכז עיבוד גמיש המאפשר טיפול בזמן נתונים בזמן אמת וביצוע עיבוד אותות דיגיטלי. הרכיב כולל עד 44 אלפי אלמנטים לוגיים, 108 בלוקים של זיכרון פנימי בław של כ-1.9 מגה-בייט, ו-72 מכפלים פנימיים של 18×18 סיביות, מה שמאפשר לו לבצע פעולות עיבוד מתמטי מורכבות בתדרים גבוהים. ה- FPGA כולל גם מגנטון PLL מבנה לייצוב וזמן אותן, ותומך במשקי קלט/פלט מגוונים ברמות מתח שונות.

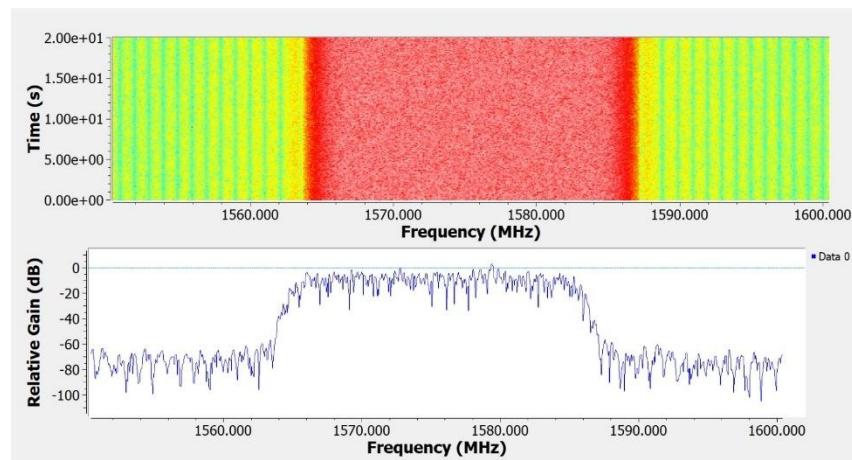
תקיינו המרכז בפרויקט הוא לחבר בין רכיבי הקלט/פלט של הלוח, לנחל את זרימת הנתונים ולהיות אחראי לישום הלוגיקה והאלגוריתמים הנדרשים לעיבוד תקשורת ברדי מוגדר תוכנה, תוך תמייה ב מהירות גבוהה ועיבוד מקביל. היכולות הללו חיוניות בימוש פרוטוקולים, פילטרים ועיבוד בסיס פס, ומאפשרות לוח לעבוד ברוחבי פס גבויים ובישיומי תקשורת אלחוטית מגוונים.

חשוב לציין כי את הבדיקות הראשונות על הhardware ביצנו באמצעות תוכנת GNUradio שהוא פלטפורמת תוכנה חופשית לעיבוד אותות דיגיטלי בזמן אמת, המאפשרת יצירה, סימולציה והפעלה של מערכות רדי מוגדר תוכנה (SDR) באמצעות "בלוקים" גרפיים הניתנים לקישור ולתכנות.



איור 9 - שרשרת חסימה בתוכנת *GNU Radio*

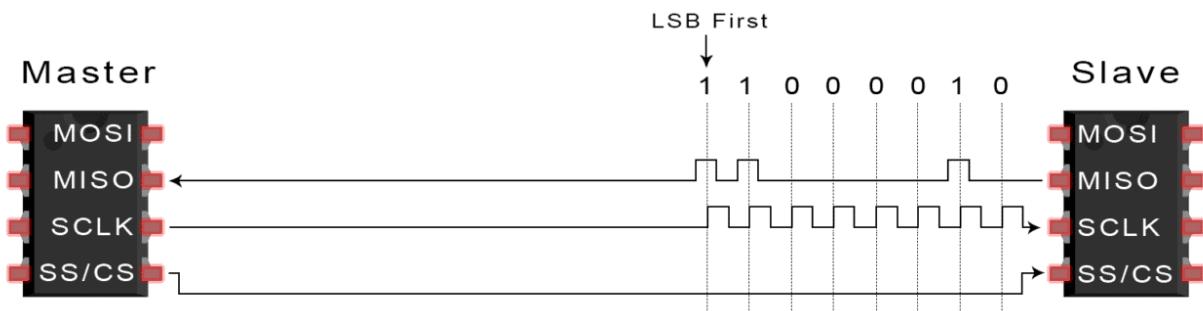
דוגמה לייצרת GPS Barrage jamming באמצעות limeSDR mini 2.0, באמצעות תוכנת GPS



גרף GPS, Spectron & Waterfall - 5. חסימת GPS.

9.1.4 פרוטוקול תקשורת SPI

פרוטוקול SPI הוא פרוטוקול תקשורת סריאלי שבו רכיבים מחוברים זה לזו ומעבירים מידע זה אל זה באמצעות קווי שליטה ונתונים ייעודיים. בפרוטוקול זה מוגדר master אחד אשר שולח אותות שעון ופקודות, וslave אחד או יותר אשר מגיבים בהתאם לפקודות שמתקבלות. התקשרות עצמה מתבצעת דרך קו MOSI שמעבירים מידע מהmaster לslave, קו MISO שמעבירים מידע בחזרה מהslave ל-master, קו SCLK שאחראי על פעימות השעון הシンכרוניות, וקו SS/CS אשר מאפשר ומפעיל את השאים הנבחרים ברכיב LMX2492EVM התקשרות SPI מוגדרת מחוץ לוח, כאשר ממתק USB2ANY הוא זה שמשמש כmaster ומקבל מהמחשב את הפיקודות שמאגרות את פעלות הרכיב, ולאחר מכן מתרגם אותן לאותות SPI אל הLMX2492EVM עצמו אשר משמש כאן בplaces, כך למשל מוגברות הפעולות *sweep*, *ramp*, *קביעת ramp*, *שינוי PLL* ופרמטרים נוספים. לעומת זאת, ברכיב LimeSDR Mini 2.0 התקשורת SPI מתבצעת בתוך הלוח, שם רכיב FPGA המוגדר משמש כmaster השולט ישירות ברכיב LMS7002M אשר מהווה את השאים slaves באשר ה-FPGA שולח לו פקודות חומרה בזמן אמת לקביעת תדרים, קונפיגורציות של מסננים, מגברים ושלבי מיתוג פנימיים.



אייר 10 - פרוטוקול SPI

בפרוטוקול SPI סyncron בין המידע בין master-slave מוגדר על ידי אות השעון (CLK) כאשר בכל מחזור שעון עובר ביט אחד בלבד - אפסון של תקשורת טוירית וסינכרונית. קצב העברה נקבע ישירות על פי תדר השעון, לאחר זה master הוא זה שמאפיק ומגדר את אותות השעון עבור המערכת. חשוב לציין שיש אפשרות להפעיל SPI גם במצב אסינכרוני, כלומר ללא שימוש בשעון, אך בפרויקט שלנו אנו עובדים במצב הシンכרוני וכן זה לא לוונטי עבורה. בחירת slaves הרצוי נעשית על ידי הורדת קו הבחירה(SS) שלו לרמת מתח נמוכה (0 לוגי), ואם נרצה למנוע העברת מידע, נשאיר את הקו הזה ברמה גבוהה (1 לוגי). מבחינת העברת הנתונים, master משדר את המידע ביט אחר ביט דרך קו MOSI והוא-slave יכול אותו בצד שלו גם דרך MOSI כאשר סדר השיליחה הוא מהבית המשמעותי ביותר (MSB) ועד לביט הפחות משמעותו (LSB). slaves יכולים גם להשיב מידע דרך קו MISO הפעם בסדר הפוך, כלומר מהבית LSB אל הביט MSB וכן נוצר מעגל המאפשר תקשורת דו-כיוונית של שליחה וקליטה בו זמנית.

9.1.5 מגברים

לאחר ניתוח המפרטים של הרחפן, גילינו כי השולט שלו משדר בעוצמה מרבית של mBm 20. כדי לבצע חסימה עיליה, علينا לשאוף להציג הרחפן בעוצמה גדולה מעוצמת הקלייטה של האותות מהשטל המפעיל, כאשר לרחפן קיימים מנגנוני הגנה מפני החסימה כמו תיקון שגיאות ועוד, כדי להבטיח את יעילות החסימה בחרנו לשדר בהספק גדול פי 10, בלומר בעוצמה של לפחות mBm 30. בפועל, הצלחנו להציג עצמה זו באמצעות שילוב של מגבר הספק נוספת בחיבור טורי, שנמצא בפתרון זול יחסית ויעיל - וכך הגיענו ליעד של mBm 30 ואף מעבר לכך.

בבחינת המפרטים של ה-2.0 LimeSDR Mini ושל ה-[EVM2492 LMX2492](#) שבם השתמשנו, נמצא שעוצמת השידור שלהם נעה בין mBm 10 - $-dBm$ 10, מה שמחייב את הצורך בשימוש במוגברי הספק כדי להציג לעוצמות השידור הנדרשות.

במהלך חיפוש אחר מגברים מתאימים למערכת, שמננו דגש על פרמטרים מרכזיים:
P1dB: נקודת הדחיסה ב-1 דציביל, שבה המוגבר מתחילה לאבד את הליינאריות שלו. במערכות חסימה ניתן לשדר גם בנקודתה זו ואף מעלה, מאחר שעווوتים באוט החסימה אינם רלוונטיים למקורה שלנו, שכן מדובר בשידור שמרתמו לחסום אותן אותות ולא לפעניהם. עם זאת, יש לוודא שהשידור בנקודתה זו אינו גורם להפרעות בתדרים שכנים שאינם נדרשים לחסימה.

Saturation: ההספק המרבי שהמוגבר מסוגל להפיק במצב רויה מלאה. במערכות חסימה ניתן לשדר במצב זה כדי למקסם את עצמתה השידור, כל עוד לא נוצרת הפרעה בתדרים מחוץ לתחום הרצוי.

הספק בניסת מקסימלי: רוב המוגברים מומלץ להספק בניסת עד mBm 0. במידה ושרשת ההגברת או יציאת ה-SDR מייצרת עצמה גובהה מ- mBm 0, עלול להיגרם נזק למוגבר.

מגבר הספק לשדרת שידור חסימת GPS:

המגבר שנבחר לפרויקט הוא **ZRL-2300+ Mini-Circuits**. מדובר במגבר מסוג A/L שמיועד בעיקר לשימוש בשדראות קליטה ולא בהבראה לשידור. למרות זאת, החלנו לננות לשלב אותו בשדרת השידור, ולאחר בדיקות גילינו כי הוא מספק את הדרישות ועובד בצורה תקינה.



איור 11 - מגבר לחסימת GPS

בפועל, לצורך חסימת GPS אין צורך בהספק גובה, שכן עוצמת האות שמתתקבל ברוחפן חלשה מאוד. עם זאת, מתוך רצון להגדיל את טווח החסימה ככל האפשר, החלנו להוסיף מגבר זה לשדרת השידור כדי לשפר את עוצמת השידור בכיוון זה.
להלן נתונים טכניים עיקריים של המגבר:

MODEL NO.	FREQUENCY (MHz)	GAIN (dB)				NOISE FIGURE (dB)	MAXIMUM POWER OUTPUT (dBm)			INTERCEPT POINT (dBm) Note 1	VSWR (:1) Typ.	ACTIVE DIRECTIVITY (dB) (Isolation-Gain)	DC VOLTAGE @ Pin V+ (V) Nom. Note 2	DC OPERATING CURRENT @ Pin V+ (mA)
	f _L f _U	Flatness					(1 dB Compr.)	Saturated at 3dB Compr.		IP3 Typ.	In Out	Typ.		Typ. Max.
ZRL-2300(+)	1400 2300	24	21	±0.5	±1.0	2.5 3.5	24.6	23	26	+42	1.20 1.16	19	12	450 575
	1650 2150	25	22	±0.3	±0.8	2.3 3.5	24.6	23	26	+42	1.20 1.16	17	12	450 575

1. 1MHz tone spacing

2. Unit is internally voltage regulated for 6.5 to 17VDC input voltage range.

איור 12 - נתונים של מגבר GPS

ניתן לראות כי המגבר מציג יכולות הגבר מעולות תוך שמירה על רעש והחזרים נמוכים מאוד, כפי שנראה בסימולציות בהמשך.

מגבר הספק ראשון בשרשראת שידור חסימת WiFi:

המגבר שנבחר להיות ראשון לשימוש בפרויקט לחסימת תדר ה-WIFI (2.4 GHz) הוא מגבר RF רחב-פס המספק רוח של dB 40 ומוגעד לפעול בטווח תדרים של 6 GHz - 6 MHz 10 כך שהוא מתאים באופן מלא גם לתחום שלנו.

המגבר משתמש במחבר SMA ומספק הספק פלט של dB 17 בנקודה P1dB ב-2 GHz מה שמעיד על יכולתו לספק עצמה נדרשת לחסימה בתדר זה. גורם הרעש של המגבר עומד על dB 4.5dB ערך סביר בהתחשב ביחסו שלנו, והוא מוזן באמצעות מתח של +5V דרך ממתק C-Type. המגבר מספק פתרון פשוט, נגיש ויעיל להגברת הספק ראשונית המקרבת אותנו להספק הנדרש.



איור 13 - מגבר ראשון בשרשראת לחסימת WiFi

מגבר הספק שני בשרשראת שידור חסימת WiFi:

המגבר השני בשרשראת השידור שנבחר לצורך חסימת תדר ה-2.4 GHz (2.4 GHz) WiFi הוא מגבר הספק ייעודי לטווח תדרים של 2.35 GHz - 2.5 GHz, מה שהופך אותו להטامة מדויקת לתדרים שבהם פועלת מערכת הרחפן. המגבר

מספק רוח של dB 25 ויכולת הספק יציאה של עד (40) dBm 10W ערך גבוה המאפשר לנו להגיע בקלות

לעוצמות השידור שנדרשות לצורך חסימה אפקטיבית של ערוץ התקשרות גם בתנאים מאטגרים.

המגבר פועל במתח של V 24-28, מוזן בזרם עבודה של A 1.2, ומתאים לעובדה עם עומס של Ω 50. הוא מגע עם חיבור SMA ומבנה מוגן ועמיד הכול סנפירי קירור מובנים.



איור 14 - מגבר שני בשרשראת חסימת WiFi

9.1.5 אנטנות

בחירת האנטנות היא שלב קריטי להצלחת ביצוע החסימה וכן נדרש תכנון מדויק ומוסכל. בתחילת הממחקר שאלנו שימוש באנטנה כלל-כיוונית, מותruk הנחה שהיא מספק בסיסי רחוב לכל הכוונים ותאפשר הגנה בטוחהיים שונים. עם זאת, לאחר בחינת השיקולים, הבנו כי אנטנה כלל-כיוונית מפזרת את הקרינה לכל הכוונים, וכן תורמת רק תוספת הספק נמוכה מאוד לכל כיוון. בהתחשב בכך שהרחפן יהיה במרחב קצר יחסית ונראה לעין, החלפנו להטמקד באנטנה כיוונית, המסוגלת לרכיב את הקרינה לכיוון הרחפן. בכך היא מאפשרת הגבר משמעותית למערכת השידור ומעלה את האפקטיביות של החסימה בתדרים הנדרשים. אלו הם הפרמטרים המרכזיים שאנו הולכים לבחון בעת בחירת האנטנות:

- קיטוב (Polarization): חשוב לוודא שהוא תואם למערכת המשודרת.
- VSWR: מודד להטמת האנטנה למערכת, חשוב לשמייה על מינימום החזרות אותן.
- כיווני הקרינה (Elevation And Azimuth): בניית הקרינה שמאפשרת לנו להבין כיצד האנטנה מפזרת את הקרינה למרחב.
- הגבר (Gain): אחד הפרמטרים המרכזיים המשפיעים על עצמת השידור לכיוון הרצוי.

לאחר בוחנת מספר אנטנות מהעבודה וכן במקרה שהזמננו מהאינטרנטי, בחרנו בסופו של דבר להשתמש באנטנות שתוכננו ונבנו על ידי פרופסור עלי לוי.

לפי המפרט שספק לנו, אנטנת GPS-GPS כוללת הגבר של $dBi = 7$, רוחב אלומה של 70×70 מעלות, תחום תדרים של $2.4 \text{ GHz} - 1.55 \text{ GHz} - 1.62 \text{ GHz}$ וקיטוב מעגלי, נתונים המתאימים באופן מדויק לדרישות שלנו עבור חסימת אותות GPS.

אנטנת WiFi מספקת הגבר של $dBi = 12$, רוחב אלומה של 40×40 מעלות, תחום תדרים של $2.4 \text{ GHz} - 2.5 \text{ GHz}$ וגם היא פועלת בקיטוב מעגלי, תכונות התואמות את צורכי חסימת התקשות האלחוטית של הרחפן.

שימוש באנטנות בעלות הגבר גבוהה וקיטוב מעגלי מאפשר לנו לשדר למרחוק ובלבסות מצבים שבהם הרחפן נע בزواיות משתנות ואינו מושך אנכית או אופקית, מה שմגביר את סיבוי החסימה גם בתנאי תעופה דינמיים.

עם זאת, יש לזכור בחשבון שגם אנטנת הקליטה של הרחפן פועלת בקיטוב לינארי ומומוקמת בדיק בزواית אנכית או אופקית, עלול להתרחש הפסד תאורטי של $-BdB = 3$ עקב אי התאמת קיטוב, כך שיתקבל הספק קטן בחצי באנטנת הקליטה של הרחפן.

9.1.6 הרחפן

הרחפן שנבחר לצורך הוכחת החסימה בפרויקט הוא DJI Mini 2, הרחפן מתקדם בעל יכולת טיסה למרחקים גדולים ומערכות תקשורת מורכבות, היכולת פרוטוקולי תקשורת מתקדמים שיש לחתה בחשבון על מנת לחסום אותו באופן יעיל. ראשית, הרחפן מסוגל לתקשר במספר ערוצי תקשורת, וכן מטרתנו היא לחסום את כל הערוצים הפעילים. לאחר בינה עמוקה גילינו שבגדם זה קיים באג שאינו מאפשר לרחפן לפעול בתדר 5.8 GHz, למרות שבז' הנ頓ים הרשמי קיימת תמייה בתדר זה. בעקבות ממצא זה החלפנו להתקן בחסימה פיזית של ערוצי 2.4 GHz ושל (1.575 GHz GPS בלבד. בנוסף, נדרשנו להביא בחשבון את עצמת השידור של השלט בתדר WiFi, כדי להתאים את עצמת השידור של מערכת החסימה באופן שיבטיח יעילות מקסימלית.

Operating Frequency [2]	2.400-2.4835 GHz, 5.725-5.850 GHz
Transmitter Power (EIRP)	<ul style="list-style-type: none"> 2.400-2.4835 GHz FCC ≤ 26 dBm CE ≤ 20 dBm SRRC ≤ 20 dBm <ul style="list-style-type: none"> 5.725-5.850 GHz FCC ≤ 26 dBm CE ≤ 14 dBm SRRC ≤ 26 dBm
Global Navigation Satellite System (GNSS)	GPS+GLONASS+GAL ILEO

טבלה 6 - נתוני רחפן DJI Mini 2

ניתן לראות את תדרי הפעלה כפי שציינו, ואת הספק השידור המקורי לפי התקן האירופאי -20 dBm

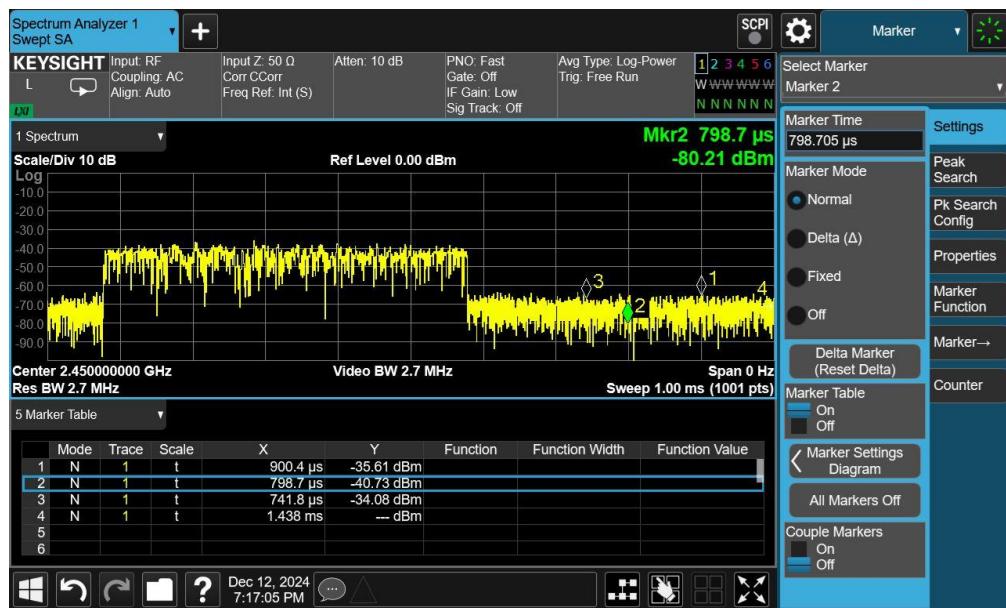
בנוסף ביצענו מדידות על מנת לקבוע את טווח התדרים המודיק בו הרחפן פועל:



גרף 6 - מודול ספקטרום, תדרי פועלות הרחפן.

ניתן לראות כי טווח התדרים בערוצ WiFi נע בין 2.4GHz ל 2.48GHz רוחב פס כולל של 80MHz.

שנית, פרוטוקול התקשרות אליו הרחפן עובד אפשרות לו לבצע frequency hopping, כלומר ל קופץ בין תדרים כל פרק זמן מסוים על מנת לנסות ולהגן על התקשרות מפני חסימות ולשפר את איכות התקשרות בין המפעיל להרחפן. היה علينا לבדוק את זמן השהייה של כל תדר (Hop Duration) על מנת לתכנן את מהירותו הסדירה שבה החוסם יעבור על טווח תדר ה WiFi (הדרישה כפי שציינו היא לסrox את כל הטווח ב מהירות גדולה פי כמה מהזמן בו הרחפן שווה על תדר מסוים). המדידה בוצעה באמצעות ספקטרום אונליין ב מצב Zero Span המאפשר להתמקד בתדר ספציפי ולמדוד את משך הזמן בו הרחפן נשאר בתדר נתון לפני שהוא עובר לתדר אחר:



גרף 7 - תצוגת Zero Span שהיית הרחפן בתדר מסוים.

חקירת מצב span zero אפשרה לנו להבין כי בリבויו בגוף הוא כ-100, משך זמן השהייה על התדר הוא כ-5 ריבועים ולכן זמן השהייה של הרחפן על תדר מסוים הוא כ-500.



על הרחפן ניתן לשנות דרך שלט המתחבר לפלאפון באמצעות אפליקציה של DJI, המציג את תדר העבודה של הרחפן, רוחב הפס, קליטת לווינטים ותצוגת יידאו מההרחפן.

איור 15 - שלט הרחפן.

ספק כוח

המערכת שלנו בנויה משורדות שידור היכולות שני ערוצים, כאשר כל ערוץ מכיל רכיב ליצירת האות החוסם ומגבר הספק, שלכל אחד מהם דרישת מתח שונה.

לדוגמה, רכיב ה-**LMX2492EVM** מוגבר הספק הראשון פועלים במתוח של 5V, בעוד שמגבר הספק בשירות שידור GPS-12V, ומגבר הספק השני בערוץ חסימת WiFi-24V. דרוש V-DC כדי לספק מתח יציב ומדויק לכל אחד מהרכיבים, אנו משתמשים בספקים בווח הממירים AC-L-DC.



איור 16 - ספק כוח.

ספק מס' 1 (שמאל):

- דגם: DQ1220A
- קלט 100-240V~ 50/60Hz
- פlut 12V DC 2A

ספק מס' 2 (באמצע):

- דגם: JCY-2430
- קלט 100-240V~ 50/60Hz
- פlut 24V DC 3A

ספק מס' 3 (מימין):

- דגם: EP-TA20S8E
- קלט 100-240V~ 50-60Hz
- פlut 5V DC 2A

כבלים RF ו-connectors

במערכת שלנו נעשה שימושocabili RF מסוג U/RG316 באורך של 10 ס"מ, המשמשים לחברו בין רכיבי השידור השונים (LimeSDR Mini 2.0, LMX2492EVM, מגברי הספק ואנטנות). הcabili SMA כוללים מחברי SMA תואמים, המאפשרים חיבור ישיר לרכיבים שבשימוש.

cabili U/RG316 יש אימפדנס של Ω 50, זהה לאימפדנס של כל רכיבי המערכת דבר שמבטיח התאמה מלאה ומונע החזרות אחרות (reflections), אשר עלולות לגרום להפסדים ולפגוע באיכות השידור.

באורך של 10 ס"מ בלבד, הפסד המתקבל הוא זניח: לפי טבלת הנתונים, בתחום התדרים המרכזיים שבהם אנו פועלים (1565-2485 MHz), הפסד עומד על dB 0.19-0.17, מה שמבטיח העברת אותן ייעלה מאוד ולא פגעה באיכות השידור.

Type	Dielectric Strength (KV)	Insulation resistance ($M\Omega \cdot KM$)	Impedance (Ω)	Capacitance (pF/m)	Rate	Weight (KG/KM)	Min bending radius (mm)
RG316/U	2.00	≥ 5000	50 ± 2	100.00	0.70	15.00	10.00
	Frequency (MHz)		Attenuation constant (dB/100m)				
	50	100	400	800	1000	2000	3000
	24.60	36.10	69.00	112.00	155.00	184.00	200.00



טבלה 7 - נתוני cabili RF

7.9.1.7 סימולציות תכנון מערכתיות ופרטניות

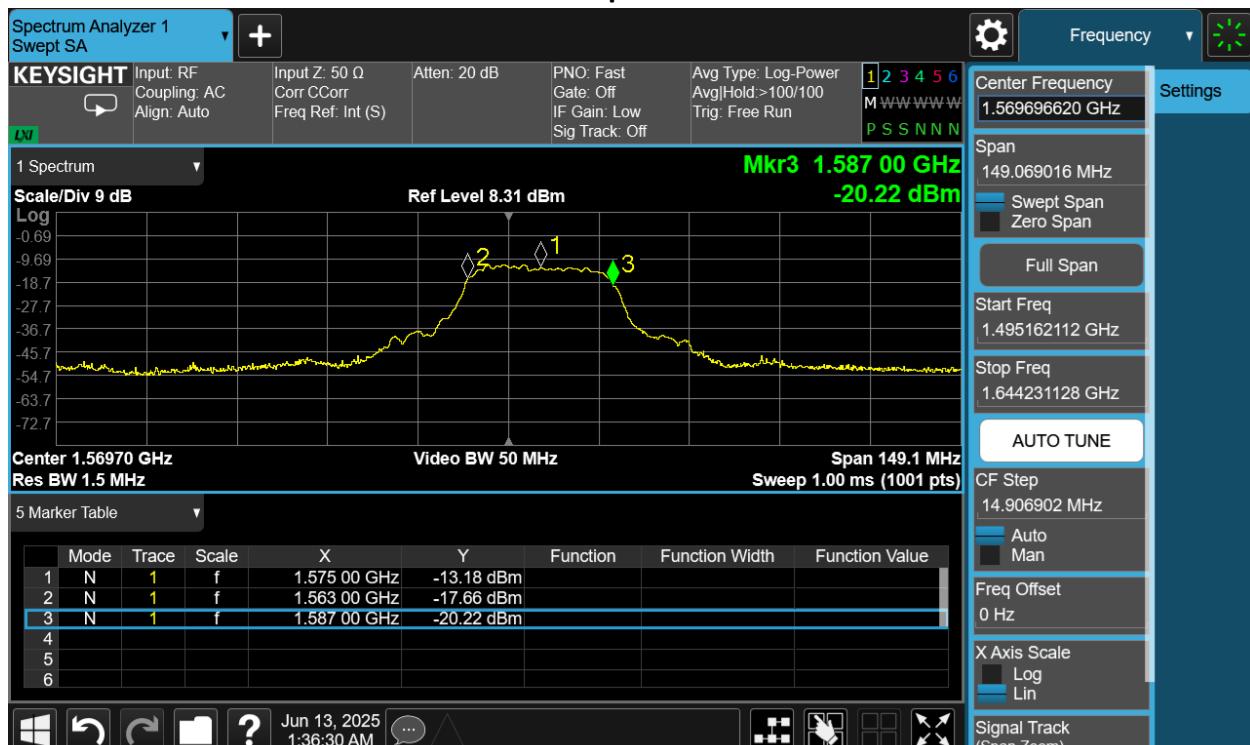
לאחר השלמת תכנון המערכת המלא, עברנו לשלב בדיקת הביצועים בפועל. שלב זה חיוני לאימוטה תוצאות התכנון ולהבטחת עמידה בדרישות ההספק וכן לווידוא שהאותות המשודרים אכן תואמים את מאפייני החסימה שנקבעו בשלבי הפיתוח התיאורטיים. לצורך כך, בחנו את שרשרת הגברת של כל אחד משני העורכים - שלב אחר שלב - עד להשגת ההספק הסופי המדרש. כמו כן, נסיף הוכחה יכולה חסימה גם בערך 5.8 GHz.

ערוץ ה-GPS:

כפי שציינו בפרק הקודם, הרכיב האחראי לייצור אות המשבש בערך זה הוא LimeSDR mini2.0.

רכיב זה מייצר רעש גאוסיאני בעוצמה ובطוויה הניטנים לקינפוג.

מוצא הרכיב ללא הגברת בטוויה חסימת GPS של הרחפן



גרף 8 - שידור GPS ללא הגברת Spectrum Analyzer - 8

ניתן לראות את רצפת הרעש מתורמת בבדיקה בתדרים אוטם הגדרנו בקונפיגורציה, 1563MHz עד 1587MHz. כאשר התדר המרכזי של תחום ה-GPS זהה הוא 1575 MHz. האות המשבש נראה כפי שציפינו ומאמת את התיאוריה לגבי שיטת Barrage Jamming בה בחרנו להשתמש לחסימת ערוץ ה-GPS.

בסכם את עוצמת ההספק ב-3 הנקודות המסומנות בטוויה:

Frequency [MHz]	Transmission power[dBm]
1563	-17.66
1575	-13.18
1587	-20.22

טבלה 8 - עוצמת שידור כתלות בתדר ללא מגברים (GPS)

מוצא המגבר בטוווח חסימת ה-GPS של הרחפן



גרף 9 GPS שידור על מגבר Spectrum Analyzer - 9

ניתן לראות שהאות שומר על צורתו לאחר ההגברה, כך שהוא מתאים לתיאוריה של שיטות barrage jamming. יש לציין כי בין מוצא המגבר לבניית הספקטרום אנליזר חיבורנו מנהת של dB 20- על מנת לא לפגוע בספקטרום אנליזר.

נסכם את עוצמת הריסק ב 3 הנקודות המופיעות בטוווח(לאחר ההנחתה):

Frequency[MHz]	Transmission Power (after attenuation) [dBm]	Transmission Power (without attenuation) [dBm]	Gain[dB]
1563	-9.66	-9.66 + 20 = 10.34	10.43 – (-17.66) = 28.09
1575	-4.93	-4.93 + 20 = 15.07	15.07 – (-13.18) = 28.25
1587	-12.43	-12.43 + 20 = 7.57	7.57 – (-20.22) = 27.79

טבלה 9 - עוצמת שידור כתלות בתדר עם מגבר (GPS)

ניתן לראות כי הגענו להספקים גבוהים ייחס למה שדורש ערוץ GPS, כך שנכנית חסימה אפקטיבית בערוץ למשל בתדר המרכזי הגענו להספק מוצא של:

$$P = 10^{\frac{P[dBm]-30}{10}} = 10^{\frac{15.07-30}{10}} = 32 [mW]$$

ערוץ ה-WIFI:

כפי שצינו בפרק הקודם, הרכיב האחראי לייצור האות המשבש בערוץ זה הוא ה- LMX2492EVM, תפקידו לבצע sweep מהיר מאוד בתחום מוגדר, כך שנצפה לראות צורה הדומה ל"לבנה" בתדר.

מוצא הרכיב ללא הגברה בטוווח חסימת ערוץ WIFI 2.4 GHz



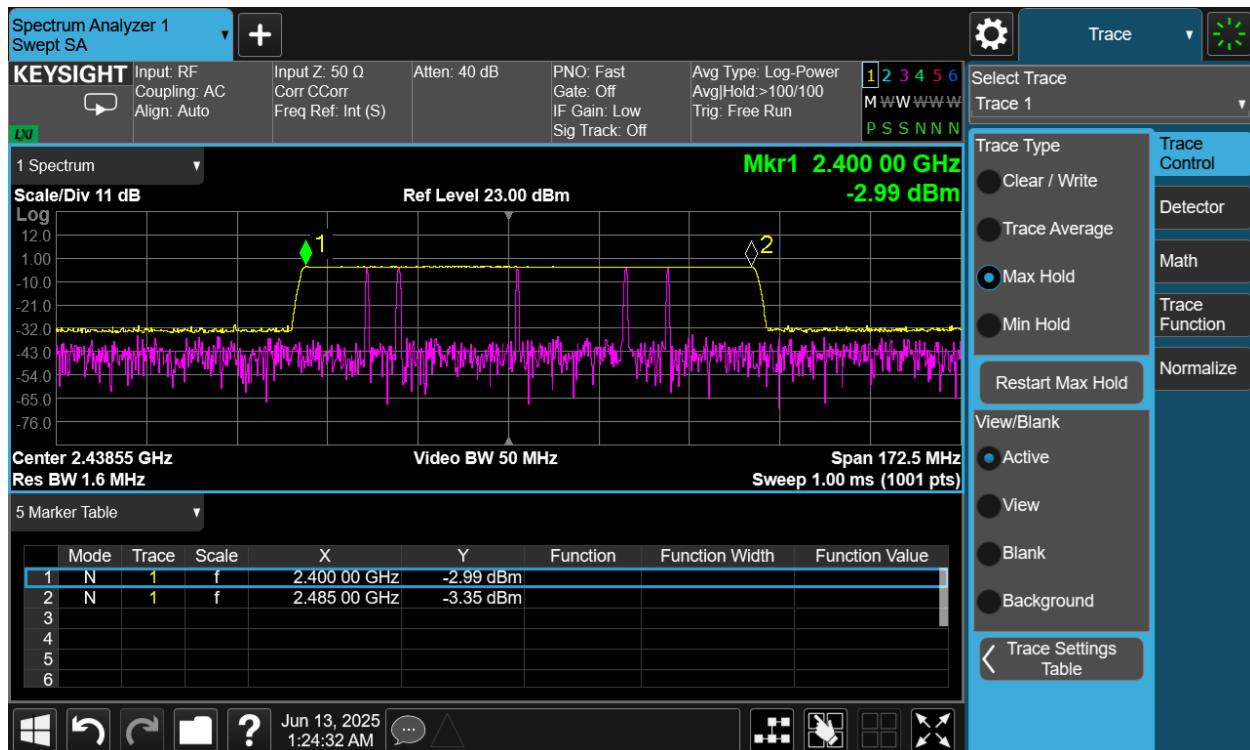
גרף 10 - שידור WIFI ללא מגברים

ניתן לראות כי אכן קיבלנו מעין מלבן בתדר. כמו כן הוספנו תצוגה שומרה את ה-sweep בזמן אמת(בסגול).
נסכם את עוצמת ההספק בטוווח המבוקש בשתי נקודות הקצה:

Frequency [MHz]	Transmission Power[dBm]
2400	-1.86
2485	-1.83

טבלה 10 - עוצמת שידור כתלות בתדר ללא מגבר (WIFI)

מוצא המגבר הראשון בשרשורת ההגברה בערוץ WiFi 2.4 GHz



גרף WiFi שידור Spectrum Analyzer - 11 עם מגבר ראשון

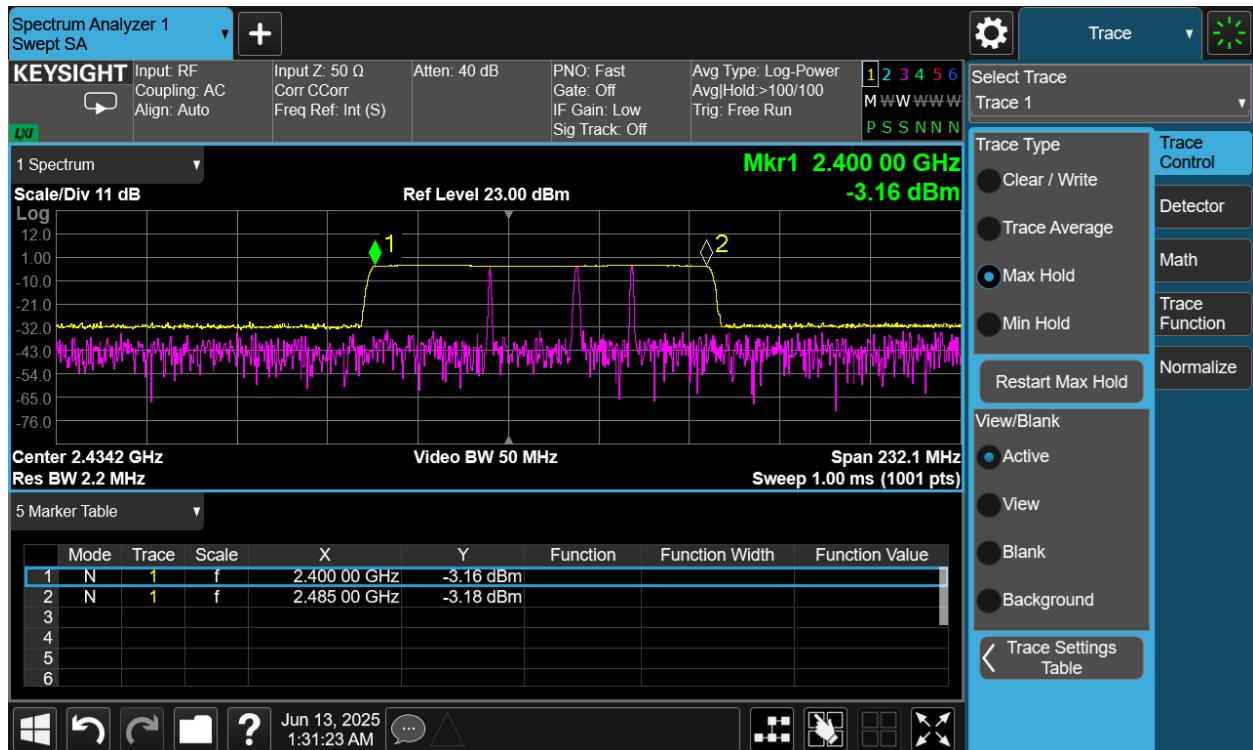
ניתן לראות שצורת האות נשמרה והסיקה מתבצעת בדיק בפי שאנו מצפים.
יש לציין כי בין מוצא המגבר לבניית הספקטרום אנטיליזר חיבורנו מנחת של dB 20- על מנת לא לפגע
בספקטרום אנטיליזר.

נסכם את תוצאות ההגברה הראשונה בשרשורת:

Frequency[MHz]	Transmission Power (after attenuation) [dBm]	Transmission Power (without attenuation) [dBm]	Gain[dB]
2400	-2.99	-2.99 + 20 = 17.01	17.01 - (-1.86) = 18.87
2485	-3.35	-3.35 + 20 = 16.65	16.65 - (-1.83) = 18.48

טבלה 11 - עוצמת שידור כתלות בתדר עם מגבר ראשון (WiFi)

מוצא המגבר השני בשרשרת ההגברה בערזץ ה- WiFi 2.4 GHz



גרף 12 - שעממת שידור WiFi בתדר נושא של המגברים

ניתן לראות שצורת האות נשמרה והסריקה מתבצעת בדיקן כפי שהוא מצפים.
יש לציין כי בין מוצא המגבר לבניית הספקטרום אנטלייזר חיבורנו מנהת של dB 40- על מנת לא לפגוע בספקטרום אנטלייזר.
נסכם את תוצאות ההגברת השנייה בשרשרת:

Frequency[MHz]	Transmission Power (after attenuation) [dBm]	Transmission Power (without attenuation) [dBm]	Gain[dB]
2400	-3.16	-3.16 + 40 = 36.84	36.84 – 17.01 = 19.83
2485	-3.18	-3.18 + 40 = 36.82	36.82 – 16.65 = 20.17

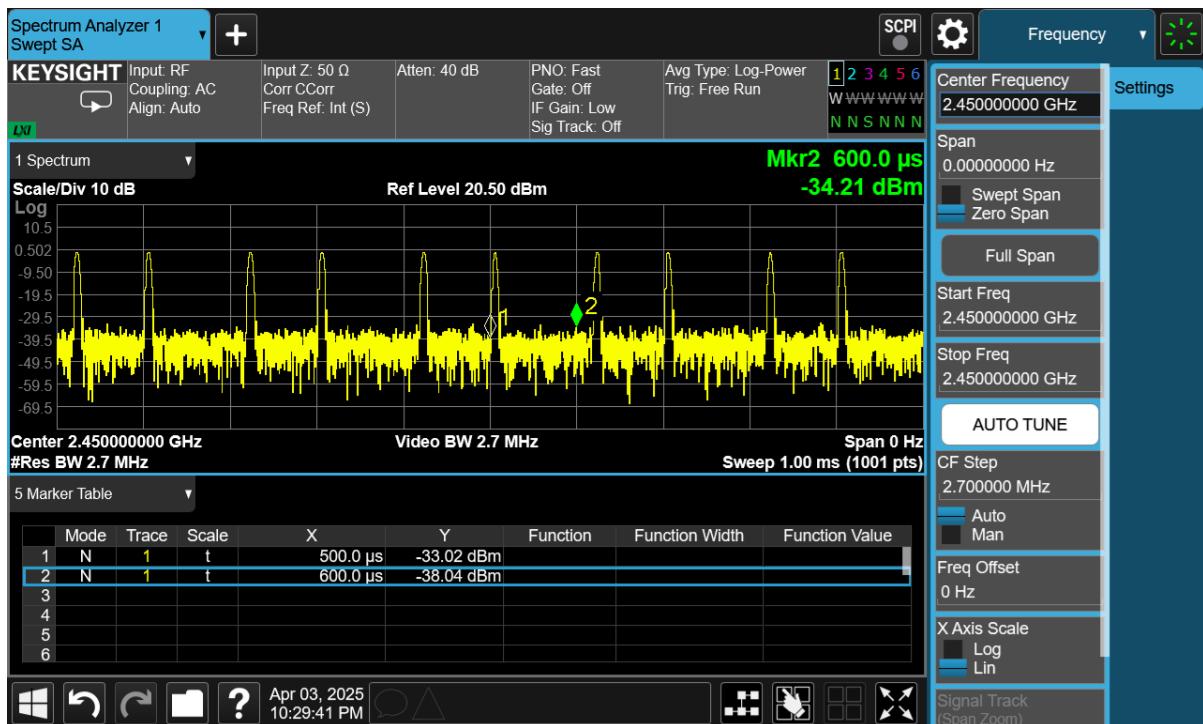
טבלה 12 - שעממת שידור בתדר נושא של המגברים (WiFi)

יש לציין כי לפני הבניית המגבר השני, בדקנו באמצעות מייצר אותות במעבדה כי הוא לא מאבד מהlianearיות שלו ושהוא לא נכנס להרואה, וכן לאחר בדיקה גילינו שההבדל בין הספק נמור(ב- 0dB) בין הספק גבוה כמו שהוא מקבל מהמגבר הראשון יש רק הבדל של dB 2 בהגברת(20 לעומת 22) כך שהוא שומר על ליניאריותיחסית טובה.

אנו מייצרים הספק מאד גבוה סביבה ה- 37 dBm

$$P = 10^{\frac{37-30}{10}} = 5W$$

הוכחת יכולת סריקה ב מהירות ns 100 באמצעות תצוגת Zero Span ב ערוץ WiFi 2.4 GHz



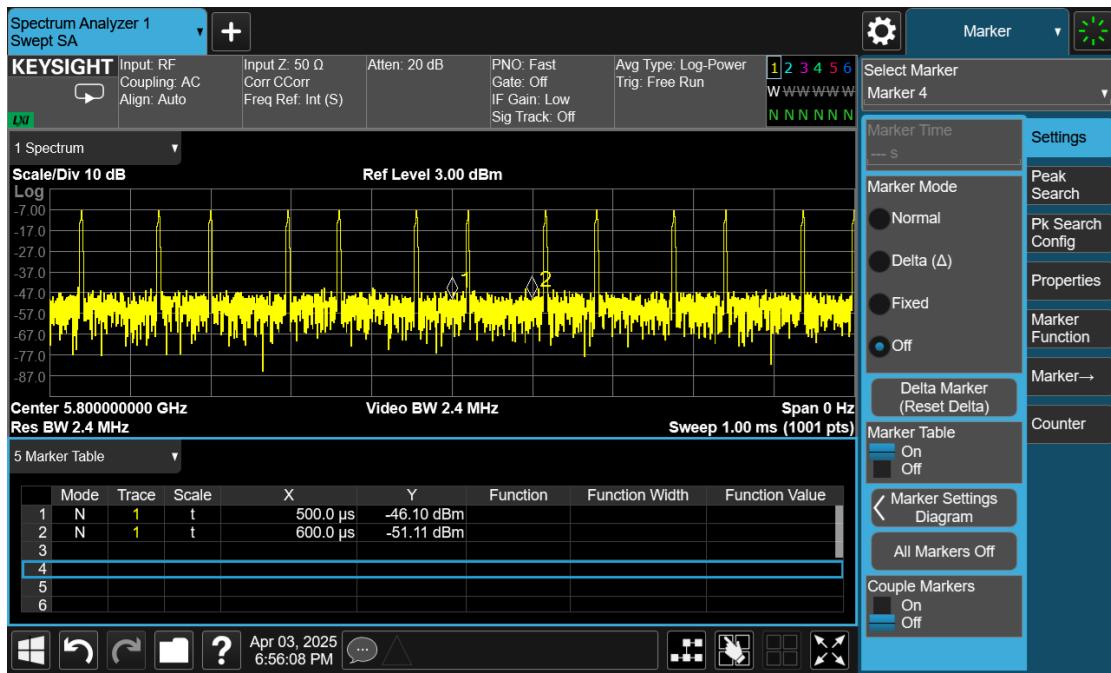
גרף 13 - תצוגת Zero Span סריקה בתחום WiFi 2.4 GHz

הוכחת יכולת חסימה בערוץ WiFi 5.8 GHz



גרף 14 - הוכחת יכולת שידור WiFi 5.8 GHz באמצעות Spectrum Analyzer

הוכחת יכולת סריקה ב מהירות **ns** **100** באמצעות תצוגת Zero Span ב ערוץ WiFi 5.8 GHz

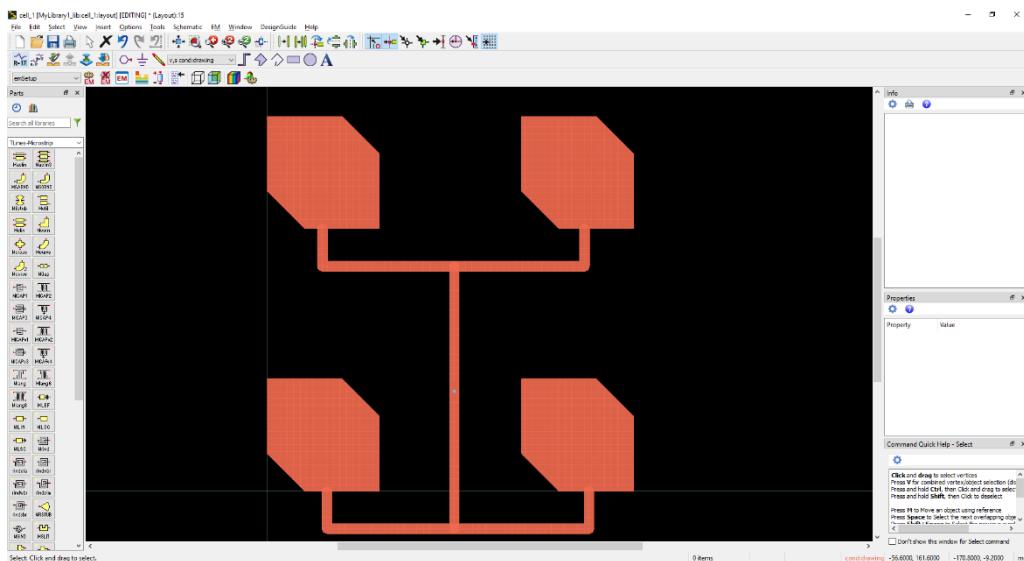


גרף 15 - תצוגת Zero Span סריקה בתחום WiFi 5.8GHz

סימולציה לאנטנה WIFI

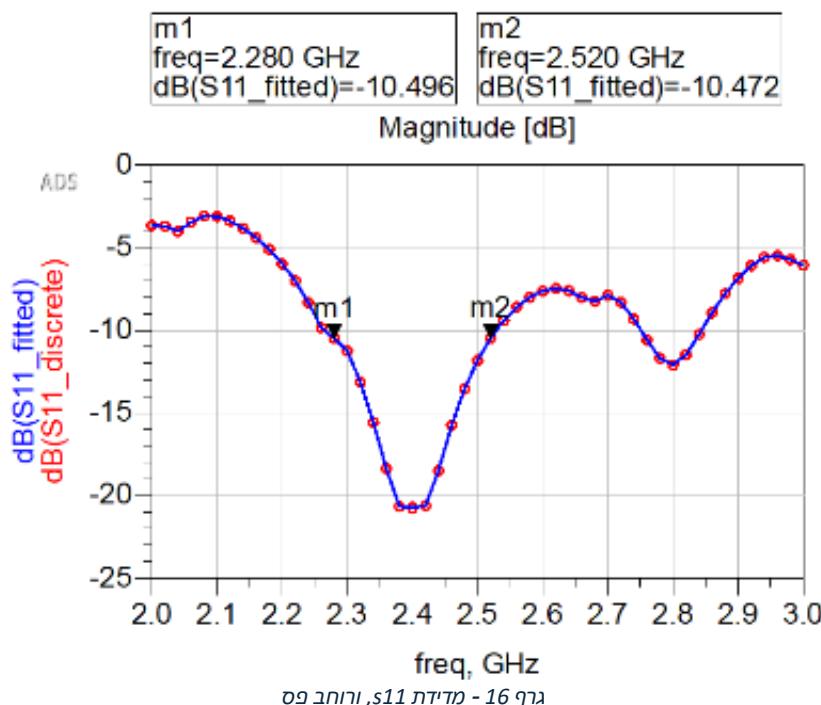
כאן נציג את תוצאות הסימולציה עבור אנטנת Wi-Fi שתוכננה לשמש לחסימה בתחום 2.4 GHz. מדובר באנטנת patch הכוללת ארבעה אלמנטים, עם קויטוב מעגלי ורוחב אלומה של 40×40 מילוטות.

בנייה האנטנה בסימולציה:



איור 17 - בניית האנטנה בסימולציה

מדדית S11(return loss), רוחב פס וחישוב VSWR



גרף 16 - מדית S11, ורוחב פס

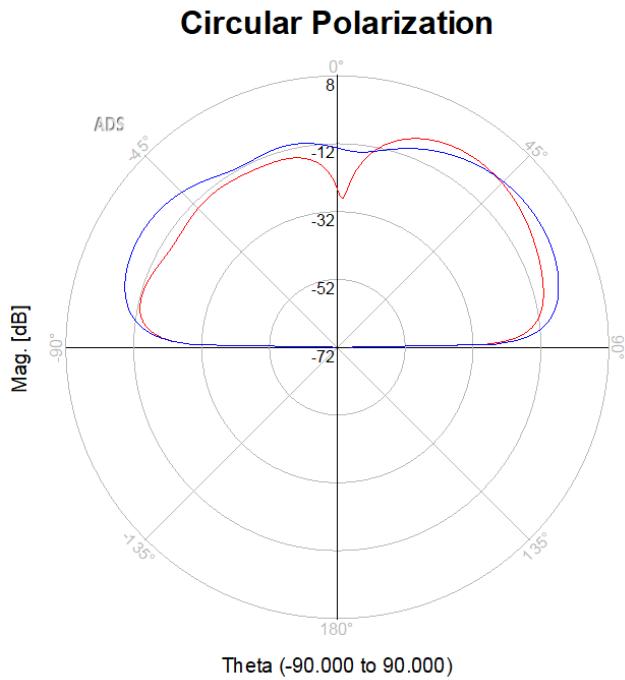
הגרף מציג את מדידת מקדם ההחזרה S11 של האנטנה בתחום התדרים 2.3 GHz-3 GHz. ציר Y מייצג את עוצמת ההחזרות dB_f, כאשר ערכים נמוכים יותר (שליליים יותר) מצביעים על החזרה נמוכה יותר ויעילות התאמה גבוהה יותר בין מקור השידור לאנטנה. שני הסמלים הבולטים בגראף (m1 וm2) ממוקמים בתחום GHz 2.280 GHz ו-2.520 GHz שביהם ערך S11 הוא בקרוב dB -10. ככלומר ברמה המקובלת להנחתת "התאמת טובה". תחום התדרים שבין שני ערכים אלה מייצג את רוחב הפס של האנטנה, כולל את הטווח שבו היא מצליחה לשדר וקלוט באופן יעיל יחסית, עם החזר נמוך. רוחב הפס של האנטנה כפוי שנמדד לפי גבולות dB -10- הוא כ-240 MHz מה שמכסה באופן מלא את תחום הפעולה הרצוי לחסימת Wi-Fi, שהוא MHz 2400-2485. ניתן לראות כי התחום הרצוי נמצא במרכז האזור שבו ההחזרה מינימלית, מה שמעיד על תכונן נכון של האנטנה.

כמו כן ניתן לחשב אתיחס הגלים העומדים(VSWR) ישירות מהגרף. ניתן לראות שבGHz 2.4 הערך הוא dB -21dB ובס-GHz הערך הוא dB -11. הערך התיאורי לחושב באמצעות מקדם החזרה - $| \Gamma |^2 = \frac{1+\Gamma}{1-\Gamma}$.
כאשר VSWR יקבע לפי $VSWR = \frac{1+\Gamma}{1-\Gamma}$.

$$VSWR(2.4 \text{ GHz}) = \frac{1 + \left(\sqrt{10^{-\frac{21}{10}}} \right)}{\left(1 - \sqrt{10^{-\frac{21}{10}}} \right)} \approx 1.2 \quad \text{לכן:}$$

$$VSWR(2.5 \text{ GHz}) = \frac{1 + \left(\sqrt{10^{-\frac{11}{10}}} \right)}{\left(1 - \sqrt{10^{-\frac{11}{10}}} \right)} \approx 1.78$$

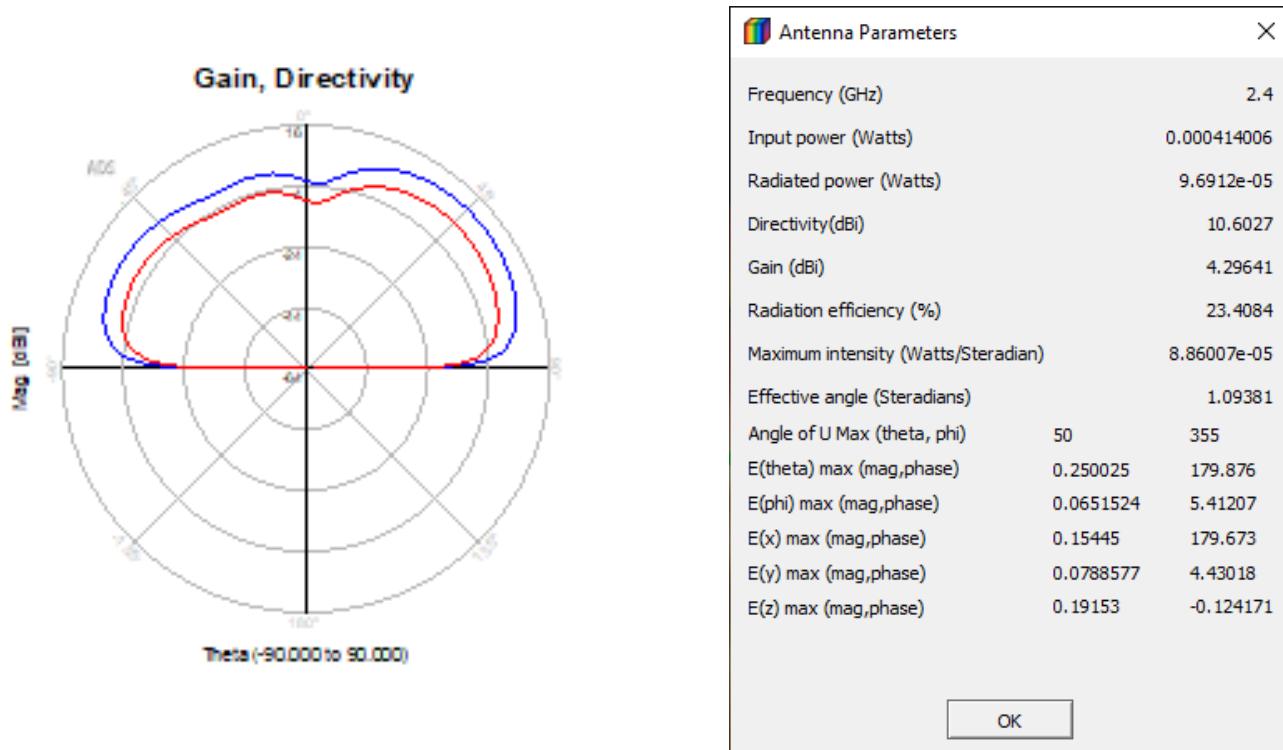
כלומר, ערך של VSWR הנמוך מ-1:1.8 נחassoc לערך מקובל ונפוץ בעולם האנטנות, ומשמעותו היא שפחות מ-10% מההספק משתקף חזרה מהאנטנה בעווות התדרים שבו מתבצע השידור. במקרה שניתן לראות בגרף, ברוב תחום השידור ערכי S11 נמוכים משמעותית מרף זה, מה שמעיד על התאמת טובה במיוחד ועל תנאים אידיאליים לשידור יציב ואיוכוטי.



graf 17 - דיאגרמת Circular Polarization

בגרף שלפנינו מוצגת הקרינה של האנטנה בשני רכיבי השדה האלקטרו, האחד בכיוון מקביל למקלט (E-co-E) והשני בכיוון מאונך אליו (E-cross). ניתן לראות כי שני הרכיבים שומרים על עצמה כמעט זהה לאורך כל תחום הزواיות, סביר BiP 12- ללא תנודות חריגות או הבדלים משמעותיים בין הרכיבים. תוצאה זו מעידה על קיטוב מעגלי איקוטי, שכן בקרינה מעגלי מתקבל שוויון בין הרכיבים המאונכים והמקבילים. העובדה שהשדה נשמר כמעט כਮעת קבועה בכל זווית גם מאפשרת קליטה יציבה על ידי אנטנות מקלט שאינן מושרות בצורה מושלמת. שילוב של קיטוב מעגלי תקני עם הגבר גובה ויציב מספק פתרון מתאים במיוחד לנסיבות שבהן היחס בין המשדר למקלט משתנה כל הזמן, כמו במקרה של חסימת וחופנים בתנועה.

פרמטרי האנטנה:



ניתן לראות כי האנטנה פועלת בдиוק בתדר הרצוי 2.4 GHz בהתאם לדרישות הפרויקט. מהנתונים שהתקבלו בסימולציה עולה כי ערך ה-Gain (הגבר הכלול, כולל הפסדים) הוא נמוך יחסית, כ- 4.3 dB לעומת 4.3 dB שעריך ה-Directivity (הכיווניות) גבוהה בהרבה כ- 10.6 dB. נתון שמתפרק לערך הגבר שסופק לנו בתיאור האנטנה לפני הסימולציה. ההבדל בין שני הפרמטרים נובע מכך שהוחסן Gain כולל בתוכו גם את השפעת הפסדים באנטנה כמו הפסדים דיאלקטריים במצע (FR4) והפסדים מוליכים בעוד ששה-*u*-Gain מתאר את פיזור הקרינה בלבד, בלומר את "פוטנציאלי הריבוץ" של הקרינה בכיוון מסוים, ללא התחשבות באובדן הספק. פער משמעותי בין Gain ל-*u*-Gain מעיד על ייעילות קרינה נמוכה, ואכן ה-*u*-Efficiency Radiation Efficiency שנמדד היא רק כ-23%, דבר שמעיד על כך שהחלק גדול מהספק המשודר הולך לאיבוד בתוך החומר או מתחמס במוליכים. עם זאת, בניתוח המשמש שbowtie מול הרחפן, אנטנה זו הפגינה את הביצועים הטובים ביותר מ-*n* מודלים האננטיות שנבדקו בפועל. יתרון זה הופיע בין תוכנית הסימולציה לביצועים בשטח נובע מהגדירות שמרניות במודול הסימולציה של ADS - לדוגמה, תוכנות חומר שמניחות הפסדים גובהים יותר מהמצב האמיתי, או מודלים שמתעלם מהאפקט של הקרינה המשולבת בין האלמנטיים. הפסדים גובהים יותר מהמצב האמיתי, או מודל שמתעלם מהאפקט של הקרינה המשולבת בין האלמנטיים.

כמו כן, ניתן לאמוד את רוחב האלומה של האנטנה על פי דיאגרמת הקרינה של הגבר. רוחב האלומה מוגדר בתחום הزواויות שבו הגבר יורך ב- $\pm 25^\circ$ מערך המקסימלי (המורע בדרכו לכל בכיוון 0°). במקרה שלנו, ירידת הגבר נصفה סביבה כ- $40^\circ \pm 25^\circ$, כלומר רוחב האלומה הכלול נע בין 15° ל- 50° , נתון התואם היטב למפרט המקורי של האנטנה.

9.1.8 תהליכיים מפורטים לפעולה וממשק משתמש

בחלק זה נפרט בהרחבה את תהליך פיתוח ממשק המשתמש, אשר נדרש למשתמש חווית הפעלה פשוטה, כניסה ווילה, שבאמצעותה ניתן להגדיר ולבצע את כל הפעולות הדורשות להפעלת חום הרחפנים התלת-עוצרי שבינו, תוך קבלת חווויים בזמן אמיתי. לשם כך, נדרש הבנה מעמיקה של אופן פעולתם של כל אחד מהרכיבים במערכת, לרבות היכולות עם פרוטוקולי התקשרות הספציפיים של כל רכיב, הספריות המתאימות שנדרשו לשילוב, האוגרים אליום היה צורך לפנות לצורך קונפיגורציה ושליטה וכן ביצוע חישובים מתמטיים ותיכוניים מודוקים שנדרשו להבטחת תקינה, מסונכנת ווילה של כל המערכת. כפי שציינו בחלופות התוכניות, `C++` היא הבחירה המועדף שלנו לשפת התוכנה.

ליצירת ממשק המשתמש בחרנו להשתמש ב*Qt*:

Qt היא ספריית פיתוח גרפית עם קוד פתוח, שמנועה לאפשר בנייתו של ממשק משתמש נוחים ו互動ראקטיביים. היא נכתבת בשפת `C++` ובסיסת על עקרונות של תכנות מונחה עצמים, דבר שהופך אותה למתאימה מאוד לפרויקטים שדורשים יציבות, ביצועים גבוהים וגמישות בתוכן.

התממשקות עם `LimeSDR mini 2.0`:

בפרויקט שלנו נעשה שימוש בהתממשקות בין הרכיב `LimeSDR Mini 2.0` לבין הממשק הגרפי שבינו, תוך שילוב ספריות תוכנה ייעודיות שפותחו עבור רכיב הלויים. ההתממשקות מתבצעת דרך ספריית *LimeSuite*, המספקת ממשק תכנותי לשילוטה בכל תוכנות הרכיב החל מהגדרת תדרים, דרך רוחב פס, ועד הגדרות של מגברי קליטה ושידור. ספרייה זו מאפשרת גישה נוחה לפונקציות קרייטיות לניהול הרכיב, והיא משתמשת היבטם עם סביבת העבודה של *Qt* הכתובה בשפת `C++`. את הקוד שמבצע את ההתממשקות עצמה בונים בהשראת קוד הבסיס של *Sio GNU Radio*, שם בוחנו כיצד בנויים בלוקים לשידור אותות, במיחוד אלו שמנגנים ליצור רעש או שליחת מידע דרך רכיבי *SDR*. בחנו את הקוד מאחוריו אותם בלוקים והבנו את עקרונות העבודה מול רכיבי *SDR*, דבר שסייע לנו לבתוב קוד מותאם משלמו המבוסס על ספריות הלויים. השימוש בין הידע מ-*SDR* לבין *LimeSuite* היכולות שמציעה *LimeSuite* אפשר לנו ליצור את התקשרות בין רכיב החומרה (*SDR*) לבין התוכנה.

הפונקציות המרכזיות בהן נעשה שימוש הן:

LimeSuite Functions	
<code>LMS_Open</code>	Opens a connection to the first available LimeSDR device.
<code>LMS_Init</code>	Initializes the device and configures internal blocks. Must be called before starting streams.
<code>LMS_EnableChannel</code>	Enables or disables a TX or RX channel.
<code>LMS_SetSampleRate</code>	Sets the sample rate in Hz for all channels and configures the FPGA accordingly.
<code>LMS_SetLOFrequency</code>	Sets the RF center frequency in Hz for a TX/RX channel.
<code>LMS_SetNormalizedGain</code>	Sets the gain using a normalized value between 0.0 and 1.0.
<code>LMS_Calibrate</code>	Performs automatic calibration of TX/RX channel after configuration.
<code>LMS_SetupStream</code>	Initializes a data stream (TX or RX) with configuration parameters like FIFO size and format.

LMS_StartStream	Starts streaming (TX or RX) on the initialized stream.
LMS_SendStream	Sends IQ samples to the device for transmission.
LMS_StopStream	Stops the active stream.
LMS_DestroyStream	Releases memory and internal resources associated with the stream.
LMS_Close	Closes the device and deallocates its handle. Must be called when the device is no longer needed.

טבלה 13 .LimeSuite Functions - 13

התממשקות עם LMX2492EVM

הברטייס אינן מתחבר ישירות למחשב אלא דרך רכיב מתווך בשם USB2ANY. פלטפורמת USB2ANY של TI משמשת כמתאם חומרה, המאפשרת חיבור בין מחשב אישי למגוון מודולי הערבה של TI. המתאים מתרגם פקודות מהמחשב לפרוטוקולי תקשורת שונים כגון SPI, I2C, GPIO ועוד. בכך מאפשר שליטה ובקירה נוחה על הברטייס.

המחשב שולח פקודות דרך ה-USB2ANY, שמנדרות את טווחי הסריקה, מהירות המעבר בין התדרים ופרמטרים נוספים. ה-USB2ANY מתרגם את הפקודות ומעביר אותן ישירות אל ה-LMX2492EVM באמצעות מוחורי שעון סדריים שמכתבים את קצב הנתונים. כדי לאפשר עבודה עם ה-USB2ANY מתוך התוכנה שלנו, שילבנו קובץ DLL ייעודי שמספק פונקציות API לתקשורת בין המחשב לרכיב.

הפונקציות המרכזיות בהן נעשה שימוש הן:

Controller Functions	
u2aFindControllers	Scans the USB bus, enumerating USB2ANY devices and creating a list of the devices it finds. The list can be read using the u2aGetSerialNumber function.
u2aGetSerialNumber	Returns the serial number of one of the USB2ANY devices found by a previous call to the u2aFindControllers function.
u2aOpen	Opens communication with the USB2ANY controller that has the specified serial number.
u2aClose	Closes communication with the USB2ANY controller associated with the specified handle.
SPI Functions	
u2aSPI_Control	Sets the parameters for SPI transactions.
u2aSPI_WriteAndRead	Simultaneously writes and reads data from an SPI device.

טבלה 14 .USB2Any Functions - 14

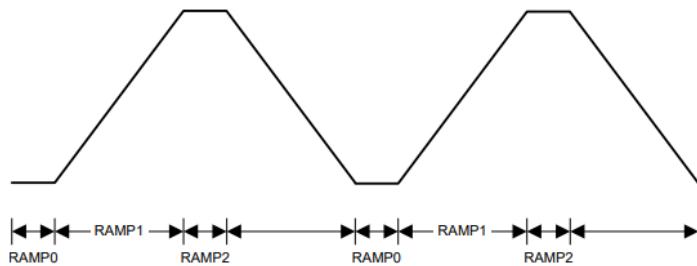
במהלך ההתממשקות נתקלנו בבעיות טכניות מסוימות, הנובע מהבדלי ארכיטקטורת תוכנה בין הרכיבים שבתוכם השתמשנו. רכיב ה-LimeSDR Mini 2.0 דרש קומpileציה של התוכנה בפורמט 64 ביט, בעוד שההספירה הרשמית של Texas Instruments (המשמש לתקשורת ה-USB2ANY)utz (המשמש לשליטה ב-LMX2492EVM)

זמין רק בגרסת 32 ביט. לאחר שתהיליך אחד איננו יכול לטעון ספריות בקונפיגורציה שונה (64 ביט מול 32 ביט), לא ניתן היה לתקשר לשירות עם רכיב ה-Y2ANY USB2ANY מ憑ת התוכנה הראשית. כדי להתגבר על מגבלה זו, פיתחנו תהליך עזר בשם usb2any_helper.exe - תוכנה נפרדת אשר מוקומפלת ב-32 ביט, ומבצעת את כל פעולות ה-SPI מול ה-Y2ANY. התקשרות בין שתי התוכנות נעשית באמצעות Named Pipe - מנגן תקשורת (IPC) של מערכת הפעלה Windows, המאפשר העברת מידע בין תהליכיamed Pipe באמצעות ערוץ תקשורת דו-כיווני, המזהה לפי שם ייחודי במערכת "usb2any_pipe\usb2any_pipe\.\.\.". פתרון זה מאפשר גישור מלא בין רכיבי 64 ביט לרכיבי 32 ביט ושליטה מלאה ברכיב ה-LMX2492EVM מ憑ת המשק הראשי של המערכת.

על מנת לקנוף את רכיב ה-LMX2492EVM ולשלוט בפרמטרים הקritisיים של תהליך החסימה (כגון תדרים, משך הסריקה וקצב השינוי), היה علينا לעמך את מבנה הרגיסטרים של הרכיב ואת הערכים שיש לבתוב בהם. לצורך כך נעזרנו בתוכנת TICS Pro, המספקת על ידי Texas Instruments, אשר מאפשרת הדמייה וקונפיגורציה של הרכיב ומציגת בצוורה ברורה את ערכי הרגיסטרים הרצויים בהתאם לפרמטרים שנבחרו. בנוסף, עיינו בפרט הטכני (datasheet) של ה-LMX2492EVM, המפרט את שמות האוגרים, מבנים ותפקידם. כל אוגר במערכת בניי מסגרת של 24 ביט, באשר 16 ביטים מוקצים לבחינת הרגיסטר ו-8 ביטים הראשונים מכילים את ערך המידע שיש לבתוב לתוכו (format של SPI).

- **R2** - משמש במרכז בקרה להפעלת או כבוי הרכיב.
- **R16 ו-R17** - אחראים על הגדרות ערך חלוקת המשוב (Feedback Divide Value), הידוע כ-N_PLL. ערך זה מגדיר את מכפיל התדר ב-PLL, ומהווה חלק מרכזי בקביעת התדר הסופי. טווח הערכים האפשריים הוא מ-16 ועד 262132.
- **R19, R20, R21** - מגדירים את המונה השבררי (FRAC_NUM) של חלוקת התדר. ערכיהם אלו משפיעים על הדיק בتردد הפלט, במיוחד כאשר יש צורך בתדרים שאינם כפולות של תדר השעון.
- **R58** - להפעלת מנגנון הסריקה הדינמית (ramp/sweep) ברכיב ה-LMX2492.
- **R82** - מגדיר את הגבול העליון של הרמפה (Sweep Limit).
- **R90, R91, R97, R98** - קובעים את מטר הזמן שבו תבצע הרמפה - ככל מרוחק שwon ימשכו העלייה או הירידה בתדר.
- **R86, R87, R88** - שלושת הרגיסטרים האלה מייצגים יחד ערך של 24 ביט (3 בתים) שנקרא RAMPO_INC, ערך זה מבטיב את גודל העלייה בתדר בכל שלב של הרמפה.
- **R93, R94, R95, R96** - ארבעת הרגיסטרים הללו מייצגים את קצב הירידה בתדר ברמפה. הערך שנכתב כאן הוא המשלים ל-2 (Two's Complement) של אותו RAMPO_INC מהעלייה, כי השינוי בתדר הפעם הוא שלילי ויש לייצג אותו בפורמט בינארי מתאים.

להלן דוגמאות לאופן הכתיבה לרגיסטורים לייצור רמפה וגל משולש אחד:



RAMP	RAMPx_LEN	RAMPx_INC	RAMPx_NEXT	RAMPx_RST
RAMP0	10 us/ 100 MHz =1000	0	1	1
RAMP1	90 us / 100 MHz =9000	$(\Delta F / F_{pd}) / RAMP1_LEN \times 2^{24}$ = $(400/100)/9000 \times 16777216 = 7457$	2	0
RAMP2	10 us/ 100 MHz =1000	0	3	0
RAMP3	90 us / 100 MHz =9000	$(\Delta F / F_{pd}) / RAMP1_LEN \times 2^{24}$ = $(-400/100)/9000 \times 16777216 = -7457$ Program in 2's complement of -7457 $= 2^{30} - 7457 = 1073734367$	0	0

טבלה 15 - הגדות לייצור Sweep תדרים בזורת משולש.

לדוגמה:

נציב ערכים על מנת ליצור גל משולש לטווח תדרים 2.4-2.485 GHz במהירות סריקה של 100us

$$RAMP0_{LEN} = Sweep\ Time \cdot F_{pd} = 100 \times 10^{-6} \cdot 100 \times 10^6 = 10000 = \text{0x002710}$$

$$R_{90} = \text{0x005A10} ; R_{91} = \text{0x005B27}$$

$$RAMP0_{INC} = \frac{\left(\frac{\Delta F}{F_{pd}}\right)}{RAMP0_{LEN}} \cdot 2^{24} = \frac{\left(\frac{85}{100}\right)}{10000} \cdot 2^{24} = 1426 = \text{0x000592}$$

$$R_{86} = \text{0x005692} ; R_{87} = \text{0x005705} ; R_{88} = \text{0x005800}$$

בעת כתיבת ערך RAMPO_INC מוצאים פיצול לשולשה בתים בני 8 ביט כל אחד ונשלחים לרגיסטרים בהתאם.

$$RAMP1_{LEN} = Sweep\ Time \cdot F_{pd} = 100 \times 10^{-6} \cdot 100 \times 10^6 = 10000 = \text{0x002710}$$

$$R_{96} = \text{0x006110} ; R_{97} = \text{0x006227}$$

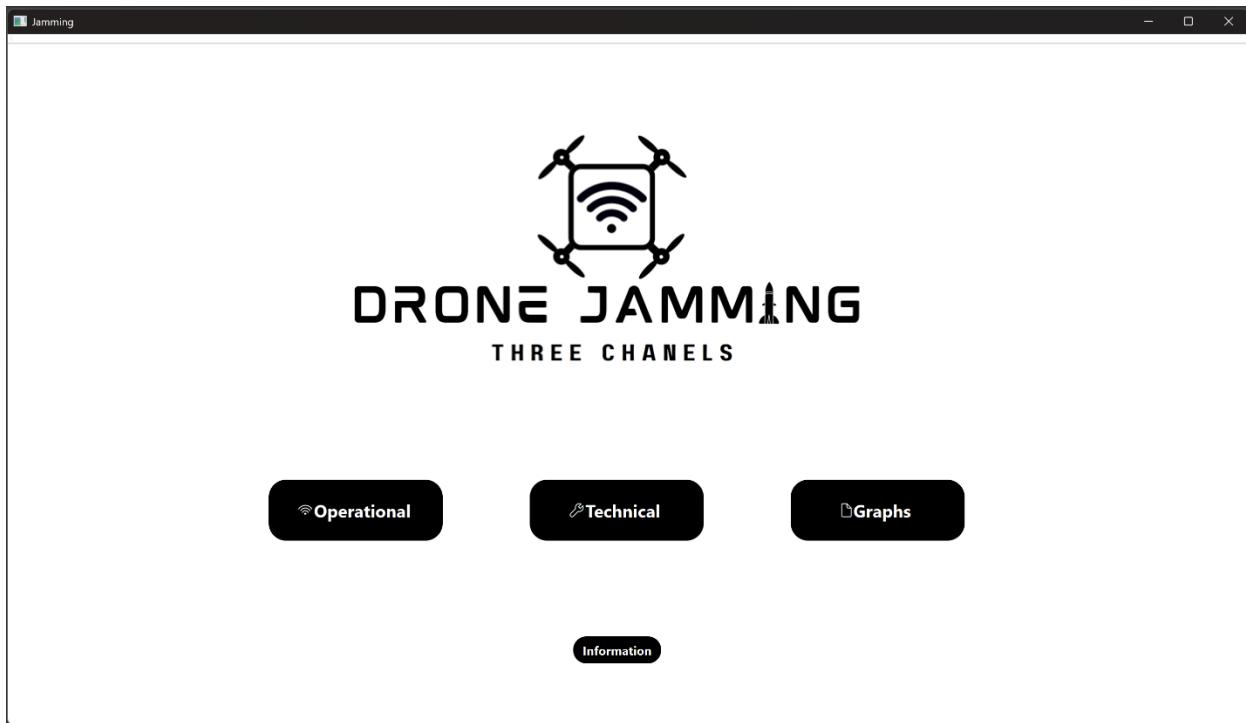
$$RAMP1_{INC} = 2^{30} - RAMP0_{INC} = 2^{30} - 1426 = 1073740398 = \text{0x3FFFA6E}$$

$$R_{93} = \text{0x005D6E} ; R_{94} = \text{0x005EFA} ; R_{95} = \text{0x005FFF} ; R_{96} = \text{0x00603F}$$

בעת כתיבת ערך RAMP1_INC מוצאים פיצול לארבעה בתים בני 8 ביט כל אחד ונשלחים לרגיסטרים בהתאם.

9.1.9 ממשק המשתמש (GUI)

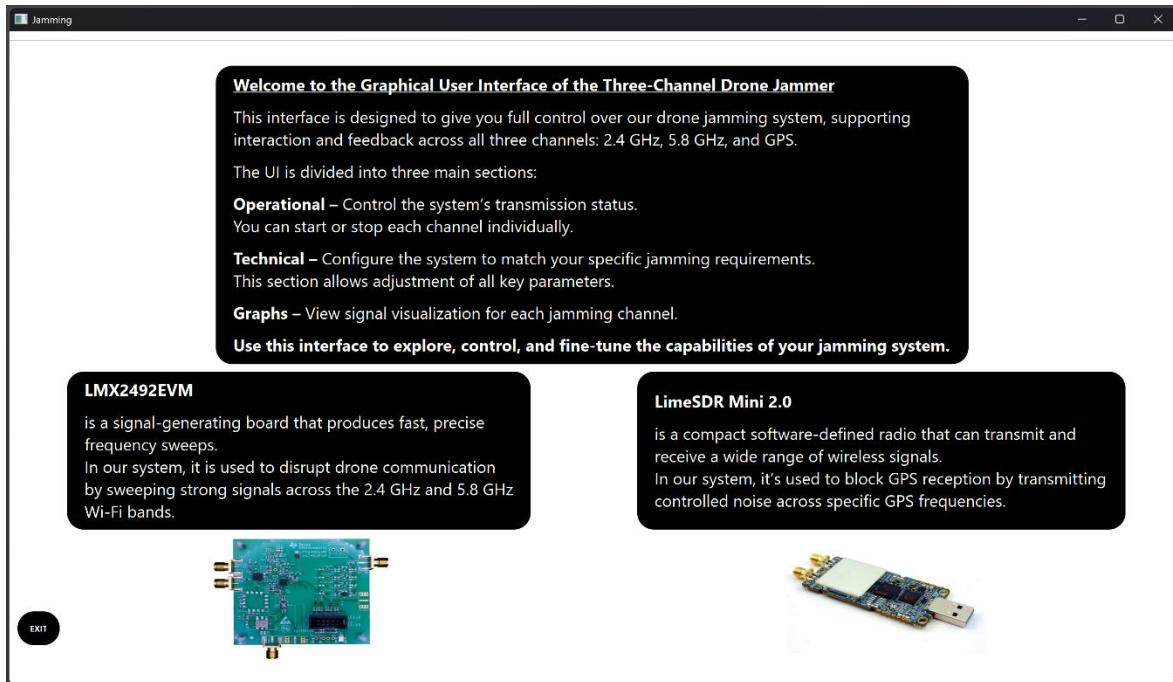
:Home page



איור 18 Home Page ,GUI - 18

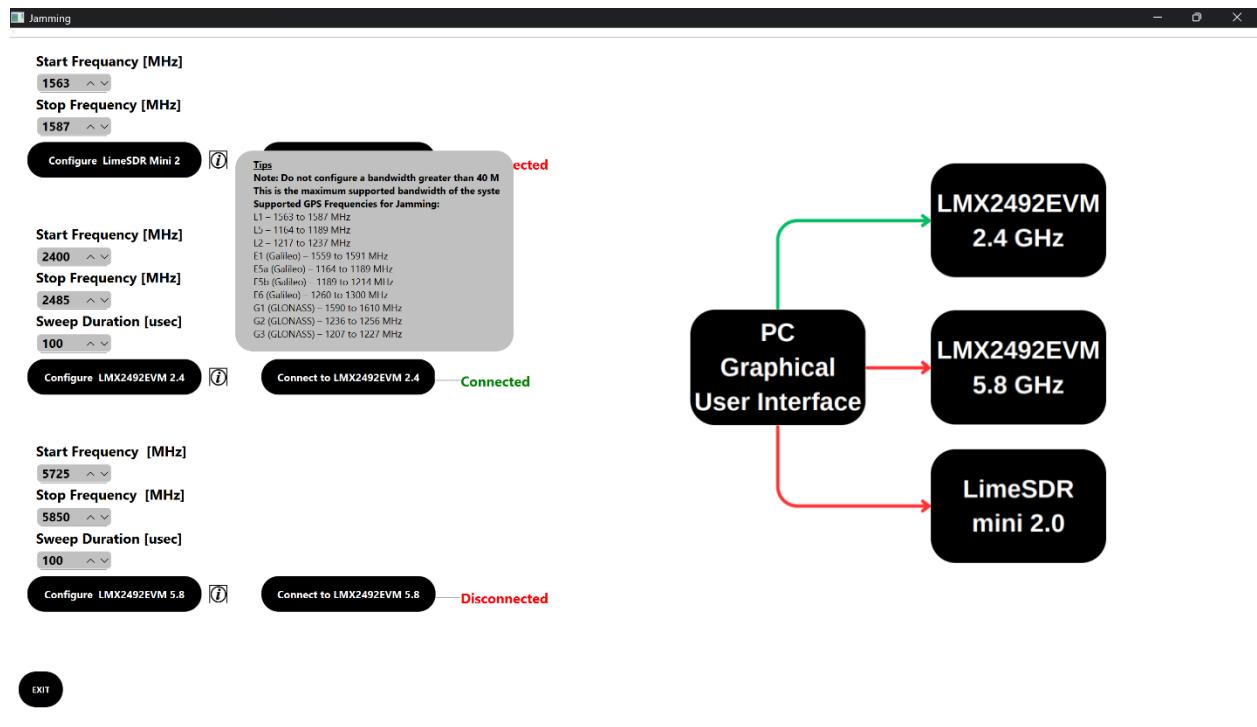
מסך הפתיחה של הממשק הגרפי שלנו מציג גישה מהירה לשולואה אזוריים מרכזיים: שליטה מבצעית, הגדרות טכניות ותצוגה גרפית. בנוסף, לחצן המידע מציג רקע כללי על המערכת ויכולות החוסם.

:Information



Information ,GUI - 19

דף המידע במכשיר מציג הסבר כללי על מערכת חוסם הרחפות ועל מבנה הממשק הגרפי. הוא כולל סקירה של שלושת אזורי השיליטה במערכת (הפעלה, הגדרות, גרפים) וכן תיאור פשוט של שני הרכיבים המרכזיים: LMX2492EVM ו-LimeSDR Mini 2.0, עם הסבר על תפקידם במערכת. העמוד נועד לעזר למשתמש להבין במהירות את תפקיד הרכיבים והמערכת כולה.

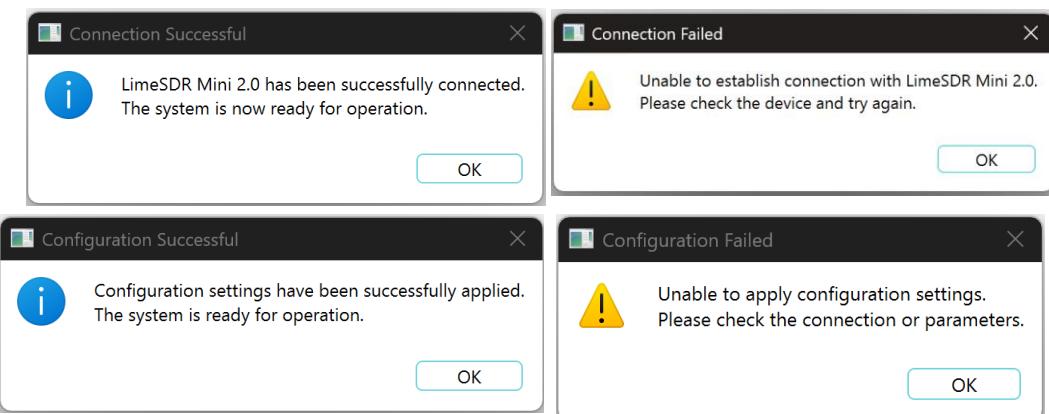


איך 20 - GUI, Technical

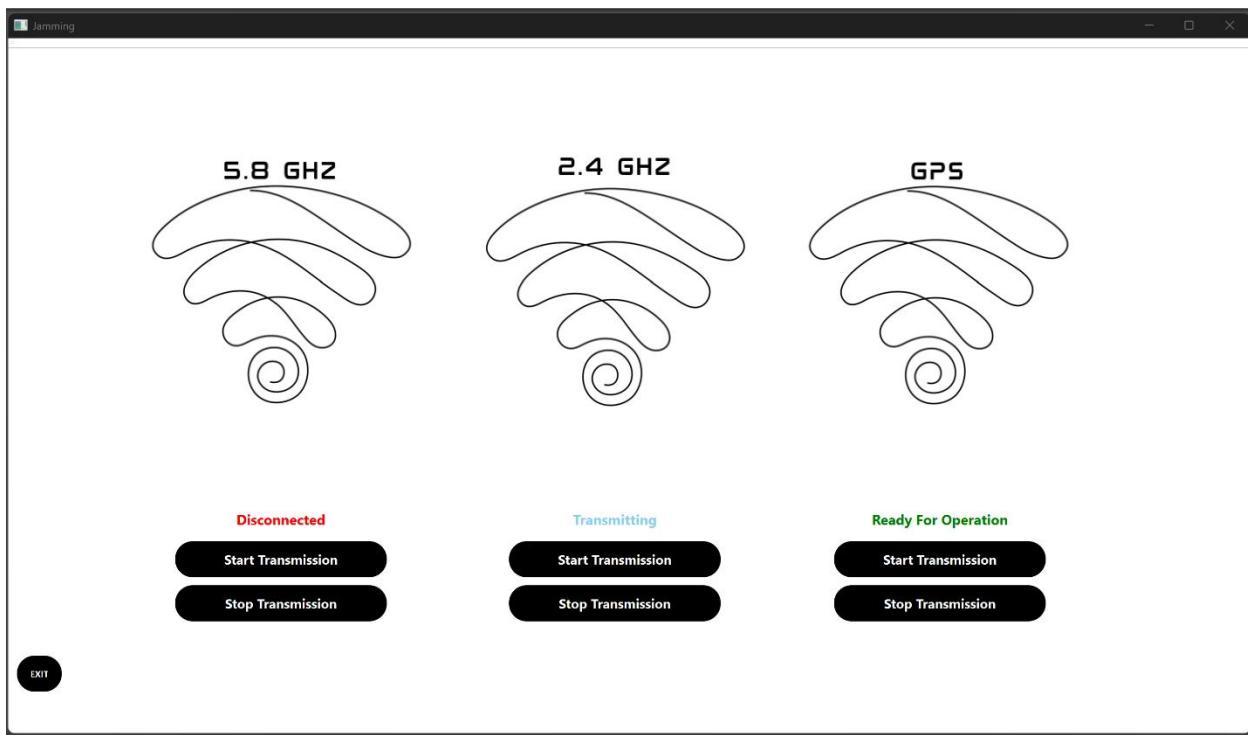
מסך ההגדרות הטכניות מאפשר למשתמש להזין את פרמטרי השידור לכל אחד משלושת ערוצי החסימה: תדרי התחלה, סיום וזמן הסריקה (Sweep Duration) (Configure).
הchineiros לרכיבי החומרה (LimeSDR Mini 2.0 ו-LMX2492EVM) מוצגים בזמן אמת, כולל>Status חיבור (Connected/Disconnected) ותרשים עוזר בצד ימין המתאר את מבנה המערכת והקשר בין הממשק(grafpi) לבין כל ערוץ.

המערכת מגבילה את טווחי הערכיים שניתן להזין בהתאם לתחומי העבודה של ה-VCO, כך שערכיים חריגים אינם מתאפשרים והשدة מתאפס אוטומטית לערך תקין.

במסך זה המשתמש מקבל טיפים והנחיות לגבי הגבולות תדרים וזמן סריקה מותרים בכל רכיב במערכת. בלחיצה על בפטור המידע (?) לצד כל מקטע קונפיגורציה, נפתחת תיבת המציגת זהירות, טווחי תדרים נתמכים ורוחבי פס מותרים, מה שמנע טעויות קלט ושימוש בערכים שאינם תקפים למערכת.

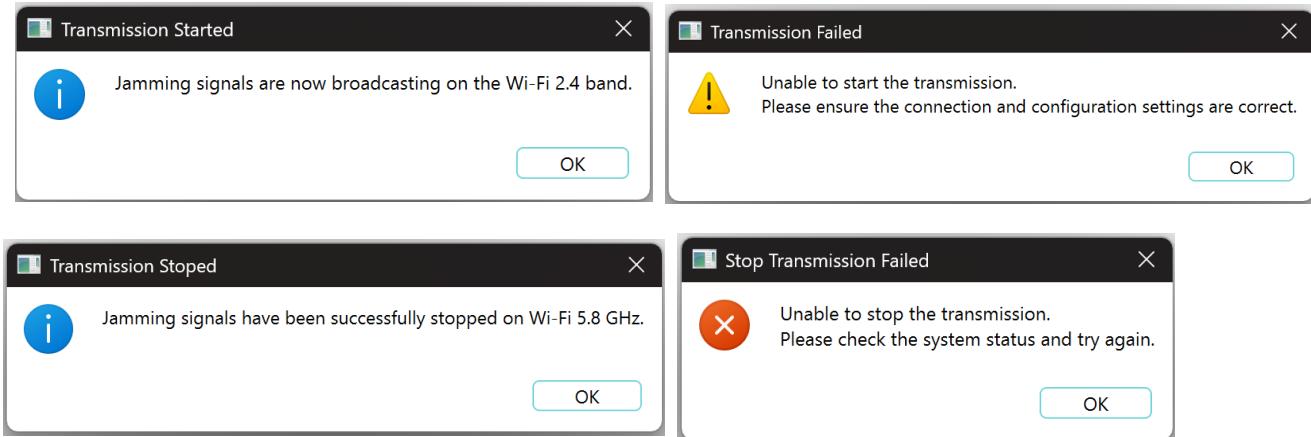


:Operational

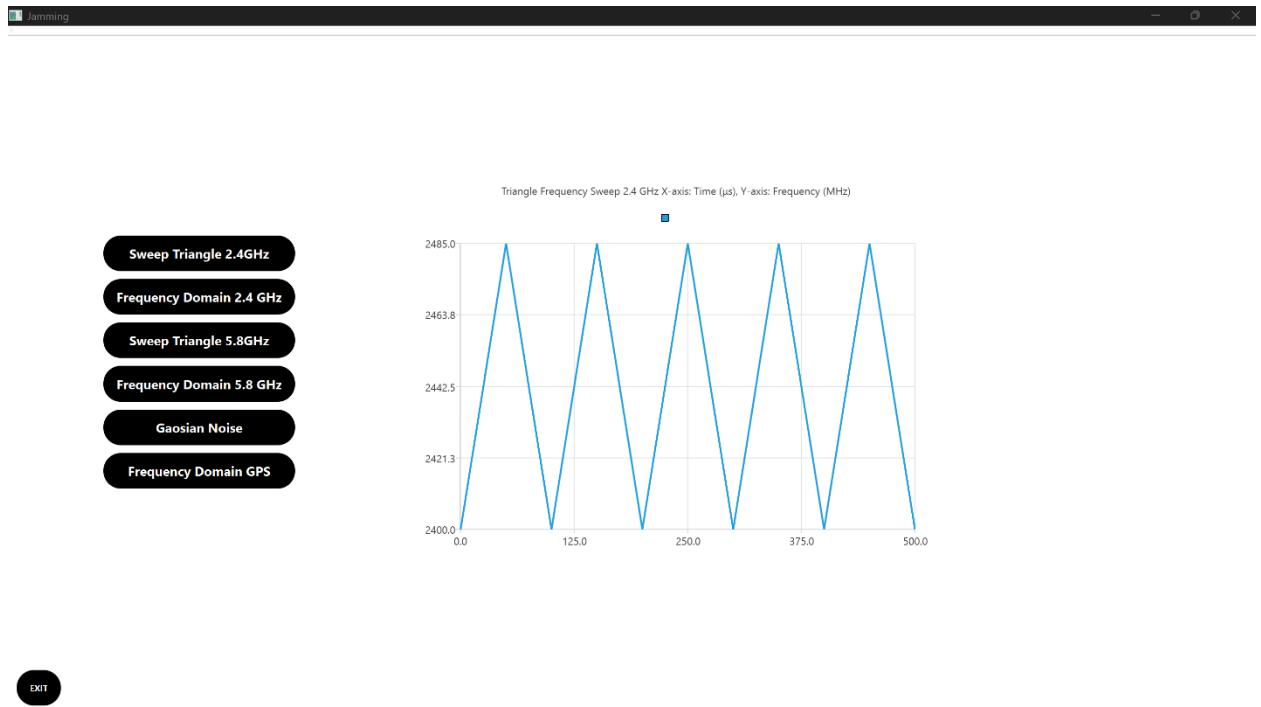


אילור ,GUI - 21

מסך הפעלה המבצעית של הממשק מאפשר שליטה מיידית על שידור החסימה בכל אחד משלושת הערכזים. כל ערך מוצג עם חיוי מצב ברור (מנוטק, משדר, מוכן להפעלה), יחד עם בפתוריה התחלת והפסקת שידור נפרדים.



:Graphs



Graphs, GUI - 22

זהו מסך שנועד להציג بصورة גרפית את אוטות השידור בכל אחד משלושת תחומי התדר שבום המערכת פועלת, כל הגրפים מתעדכנים באופן דינמי בהתאם לconifigורציה שהמשתמש מזין במכשיר, כגון תדרים התחלתיים וסופיים, משך Sweep, רוחב פס.

המשתמש יכול לעבור בין 6 גרפים שונים באמצעות הלחצנים שבצד השמאלי של המסך.

9.1.10 תכנון אינטגרציה וביצועה

הפרויקט שלנו נישן על פיתוח של שלוש שרשרות שידור נפרדות, באשר כל אחת מהן מיועדת להסימה של ערוץ תקשורת אחר. היחוד בכך הוא בכר שהרכיבים המרכזיים שמיצרים את אות החסימה שונים בין ערוצי WIFI לערוצי GPS. בתוצאה מכך נדרשנו להתחשב במספר שלבים ותכנן מוקף, כדי להבטיח שביל רכיב וכל שרשרת יעבדו בהרמוניה מלאה ויענו לדרישות הפרויקט.

ראשית, נדרשנו לבחור את רכיבי החסימה ואת שיטות החסימה הרלוונטיות לבל ערוץ תקשורת. זה דרש מחקר עמוק להבנה איך שיטת חסימה היא היעילה ביותר לבל סוג ערוץ ולאחר מכן לוודא שיש התאמה מלאה בין השיטה לבין הרכיב הנבדק. כפי שפירטו בפרקם הקודמים, בחרנו ברכיב EVE-MX2492 עם עבורי sweepjamming ו-2.0 LimeSDR Miniabarrierjamming לאחר מכן נקבעו מנגנון בחירת המגברים היינו צריכים לחשב ולהבין כמה מגברים נדרש לאחר רכיב כל רכיב חסימה על מנת להגיע להספק הנדרש בפלט לחסימת האותות הרצויים, תוך התחשבות בגורמים כמו סוג מחברים, התאמת מתחיה הפעלה ועמידה במסגרת התקציב שהוגדרה לפרויקט.

בשלב הבא, נדרשנו לבחור ולהתאים את כל רכיבי החיבור הפיזיים - כולל סוג בבל ה-RF שייחבו בין כל חלק המערכת, סוג המחברים שייתאימו לרכיבים השונים ותכננו קפדי של מבנה הקופסה הפיזית. התכנון כלל חישוב מדויק של נפח הקופסה, כך שתוכל את כל הרכיבים בצורה מסודרת, לצד הכנת פתרונות לבניות וליציאות של בבלים, אנטנות ופתחי אוורור על מנת להבטיח גישה נוחה לתקינה וכיוון בשטח.

לבסוף, לאחר שכל רכיבי החומרה שלוו ייחד, עברנו לשלב בחירת שפת התוכנות ופיתוח ממתק המשתמש הגרפי. היה علينا לתכנן ממתק שיאפשר שליטה פשוטה ויעילה בשני סוג רכיבי החסימה, וכן נדרשנו למצוא שפת תוכנות וכליים שתואימים את שניהם, לחזור את פרוטוקולי התקשורת שלהם, ולתכנן כיצד למש את הקוד בצוות המדוייק והאמינה ביותרafi שמצוין בפרקם הקודמים, בחרנו לתקשר עם EVE-MX2492 באמצעות ספירות ייעודיות ושימוש בקודGNU Radio בסיס של ו-AILO עם תקשובנו לתקינותם, ושולח אותם באמצעות פרוטוקול SPI תוך בתיבת קוד המחשב ערכיהם בזמן אמת על פי קלט המשתמש, לאוגרים הספציפיים ברכיב לזרוק קונפיגורציה נכונה של תהליכי החסימה.

9.2 חלוקת העבודה

העבודה המבצעת בשיתוף פעולה מלא, כל אחד מאייתנו היה שותף פעיל לכל שלבי העבודה, מהמחקר הראשוני, דרך למידה מעמיקה של הנושא, ועד לישום המעשי והקמת ההגשה הסופית.

9.3 תוכנית עבודה

Gantt Chart



9.4 ריכוז שינויים

מהות השינוי וסיבה	השינוי
מהות השינוי בפרויקט היא הקטנת היקף הדגמה של החסימה במהלך הצגת הפרויקט. לאחר חודשים של ניסיונות להציג רחפן מהמכלה או ממכבים, לא הצליחנו למצוא רחפן שפועל בכל שלושת העורכים בו זמנית. לאחר וርכישת רחפן בזיה ברוכה בעליות גבוההות שאינן במסגרת התקציב שלו, אושר לנו על ידי המנהה להציג את החסימה בערזץ 5.8GHz ב-31.12.2023 בלבד, באמצעות צילומי מסך מהספקטים אנגליזה. יישום מלא של חסימת ערוץ זה דורש לא רק רכישת רחפן יקר אלא גם רכיבים נוספים כמו <i>sweeper</i> ו- <i>GPS</i> , אשר יחרגו באופן משמעותי מתקציב הפרויקט.	החסימה של ערוץ 5.8 GHz בספקטרום אנגליזר במקום פיזיות על רחפן. השינוי אושר על ידי המנהה בפגישה שהתרחשה ב-31.12.2023.

טבלה 16 - ריכוז שינויים.

9.5 ניהול סיכון

במהלך הפרויקט, התעוררו מספר סיכוןים שהיו עלולים לשבב את התקדמותו או להשפיע על התוצאה הסופית. בפרק זה נציג את הסיכוןים המרכזיים ודרך התבוננות בהם.

חריגה מתקציב:

סיכון:

חריגה מתקציב הייתה יכולה היה להשפיע על יכולת שלנו להמשיך בפיתוח הפרויקט ולמש את היעדים שהוגדרו. חריגה זו עלולה הייתה להוביל לכך שלא יוכל לרכוש את כל הרכיבים הנדרשים או לשלב את כל הפונקציות שתוכננו.

דרך התבוננות:

ביצענו מעקב רציף אחר הוצאות על מנת לוודא עמידה במסגרת התקציב שהוגדרה מראש. כפי שצוין בירכוז השינויים, הוחלט להציג את חסימת ערוץ 5.8 GHz באמצעות ספקטרום אנטיזר בלבד, וזאת בשל חריגות התקציב שנבעו ממחירים גבוהים של רכבי sweepers, הרחפן שתומך בכל שלושת העוצמים, ומගברי הספק יי-ודאים. החלטה זו אפשרה לנו למקד את המשאים בהדגמה מלאה של חסימה בשני ערוצים קרייטיים 2.4 GHz ו-GPS - על גבי הרחפן עצמו, תוך הקפדה על רכישת רכבים זולים אך איכותיים.

חסימת ערוצי WIFI ו-GPS של מכשירים "ידיוטיים":

סיכון:

סיכון מרכזי בתחום חסימת ערוצי התקשרות של הרחפן הוא האפשרות לשבש בטעות גם מכשירים אחרים הפעילים באותו תדרים, כגון רשתות Wi-Fi סמכות או מכשירי GPS אזרחיים.

דרך התבוננות:

על מנת למנוע את הסיכון לפגעה במכשירים אחרים, בחרנו להשתמש באנטנות ביוניות. אנטנות אלו מאפשרות לרכב את עצמת השידור לבסוף ממוקד בלבד, ובכך לצמצם משמעותית את פיזור הקרינה לסביבה. בצורה זו, מכשירים שאינם נמצאים בקר הראייה של השידור אינם חשופים להשפעות החסימה.

חוסר התאמה בין רכבי החסימה וכישלון באינטגרציה ביניהם:

סיכון:

בioxן שהשתמשנו ברכיבים שונים לחולון יצירת אותן החסימה בכל אחד מהערוצים, היה קיים סיכון לאו התאמה תוכניתית ביניהם. בפועל, אכן נתקלנו בבעיות בשילוב ביניהם.

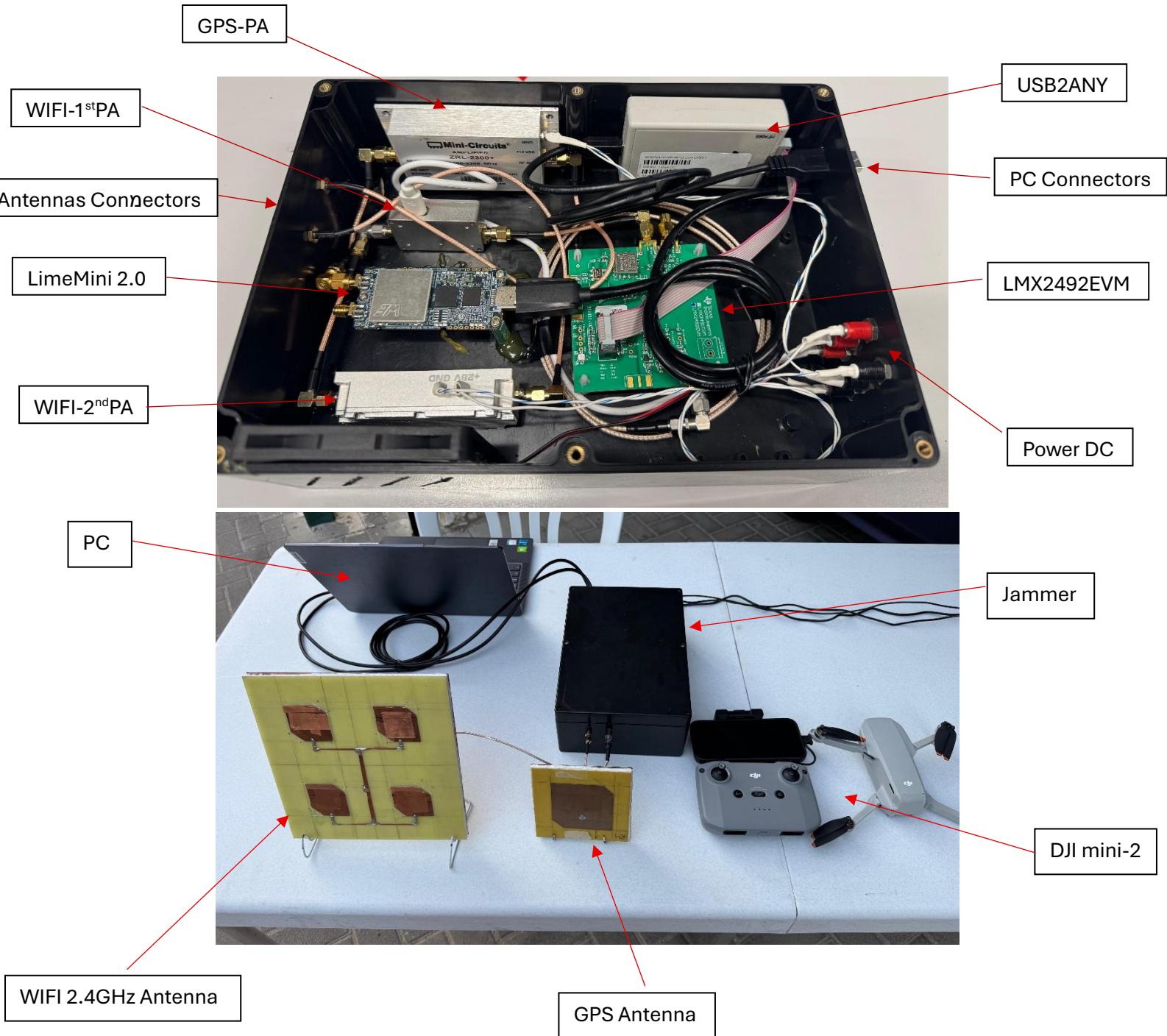
דרך התבוננות:

ראשית, השתמשנו בשפת תכנות משותפת לשני הרכיבים C++. בנוסף, כפי שפורסם בפרק ממשך המשתמש, כדי לאפשר קומPILEZA משותפת של הקוד נדרש להפריד את התקשרות עם ה-Y-USB2ANY לתהילך עצמאי. לשם כך פיתחנו את usb2any_helper.exe, תהילך נפרד שמבצע את פעולות השליטה על הרכיב. התקשרות בין התוכנות נעשית דרך Named Pipe דו-ביוני, מה שאפשר שילוב מלא בין שתי הארכיטקטורות והפעלה חלקה של המערכת.

10 תוצאות

10.1 תיאור מפורט של תוכרי הוסףים של הפרויקט

• תמונות סופיות של הדגם



- **סרטון מימוש חסימה:**

בתחילת הסרטון ניתן לראות את מישל מבצע קונפיגורציה ל-UIO של המערכת, במטרה להכין אותה להפעלה. במקביל, גיא (המתќיף) מתיצב לצד הרחפן, מוכן להפעילו. לאחר מכן מוצגת המראת הרחפן מנוקודת מבטו, עם תצוגה ישירה מהמצלמה שעליו, כך שניתן לבדוק בקליטה תקינה של עשרות לווייני GPS ובתקשרות רציפה ואיכותית בין השולט לרחפן.

ברגע שהרחפן מתחילה להתקבב בעבר המערכת, מישל מפעיל את מערכת חסימת WiFi. בתוצאה מכך, התמונה מהרחפן " קופאת" ונהיית אפורה ובמקביל הרחפן עצמו עוצר במקומו. בשלב הבא, מישל מפעיל חסימה גם על ערזע הקליטה הלוׂוׂינִית (GPS), וניתן לראות כי הרחפן מאבד לחדוטין את קליטת הלוׂוׂינִים, ערך הלוׂוׂינִים הנקלטים יורד לאפס.

כעבור מספר שניות, בדיק בפי שחדינו בתיאוריה של שיטת ה-Soft Kill, הרחפן מתחילה להנmir גובה בהדרגה עד שנוחת ברכות על הקרקע. התוצאה: **התיאוריה התממשה בפועל והודמה בהצלחה.**

[קישור לסרטון לחץ כאן](#)

2.10.2 ביצועי הדגם בפועל לעומת התכנון ב-SOS

בשילוב בין ביצועי הדגם בפועל לבין היעדים וה碼דים שהוגדרו במסמך היוזם, אנו יכולים לקבוע בביטחון כי המערכת עומדת בציפיות. הצלחנו להיע לספק שידור של C-W5, בשילוב אנטנות כיווניות הממוקדות לעבר הרחפן, מה שמבטיח חסימה יעילה ומומקדת שמצוירת פגיעה במכשירים שאינם עיינים. הממשק הגרפי (GUI) פועל באופן חלק, יציב ומהיר ומאפשר שליטה ובקраה בזמן אמיתי, כפי שניתן לראות בסרטונים שצילמנו. המערכת יכולה לפעול בהתאם DC בהתאם לתוכנו הראשוני ומסוגלת לשבש את כל ערכי התקשרות של הרחפן באופן מלא, כך שהרחפן מאבד קשר עם השולט ונאלץ לנחות הרחק ממקום השימוש.

11 סיכום ומסקנות

11.1 עיקרי המסקנות מן הפרויקט והישגים

אנו יכולים לומר בביטחון מלא כי למדנו רכובות לאורך הפרויקט הן בפן הטכני, ההנדסי והן בהיבטים ארגוניים חשובים.

התהילר כולל כלל חקר עמוק בתחום שענין אותו כבר מהשלבים הראשונים, חסימת תקשורת בין רחפנים תוך הרחבת הידע והאפקטים בתחום זה, בשילוב עקרונות ידוע מקצוע שונבש במהלך התואר בהנדסת חשמל ובעיקר בתחום התקשרות.

מעבר להשלמת הפרויקט בהצלחה מבינה טכנית וביצועית, לאחר השקעה מרווחה, רכשנו גם מיומנויות בעבודה בצוות לאורך זמן, תכנון נכון של לו"ז ועמידה ביעדים, ניהול תקציב מוגבל תוך קבלת החלטות מורכבות והתמודדות עם קשיים טכניים שאילצנו אותנו לחשב יצירתיות ולהתמיד עד למציאת פתרון.

אנו שמחים שבחרנו בפרויקט זה, פרויקט מאטגר, מתקדם ומעשי. אשר שילב בין חומרה לתוכנה, דרש אינטגרציה מדויקת בין רכיבים וכל ניסויים מהנים ומשמעותיים לאורך הדרכ.

11.2 יעדים שלא הושגו, בעיות עיקריות ולקחים

ניתן לומר כי ככל מטרות יעדיו הפרויקט הושגו, למעט יעד אחד שהושג בהדגמת חסימת ערוץ התקשרות בתדר 5.8 GHz באמצעות ספקטרום אנגלי. הדגם הסופי של המערכת אינו כולל שרשרת שידור פעילה לערוץ זה וכן עד זה לא הושלם במלואו.

כפי שפורט בפרק ריכוז השינויים, הסיבה העיקרית לכך נעוצה בקשיי מהותי להשיג רחפן התומך בכל שלושת ערוצי התקשרות (2.4 GHz, 5.8 GHz, GPS), הן דרך השאלה ממכרים והן דרך רכישה בעלות סבירה. אך התווספה מגבלה תקציבית ממשמעותית, שלא מאפשרת רכיבים נוספים נוספים (בגון סוויפר ומגברים) לשרשרת השידור של ערוץ זה.

עם זאת, ניתן לומר שכן ביצענו ניתוח תיאורטי ומעשי של הדרישות עבור הוספת ערוץ בזיה והכנו את מסך המשtamש כך שיוכל לתמוך בעתיד גם בערוץ ה- 5.8 GHz באופן מלא, בחלוקת מערכת מוחלטת.

הלקח המרכיבי מביחנתנו הוא שהקמה של מערכת אמינה, המשדרת בעוצמות גבוות ודורשת אינטגרציה עם רחפן התומך בכל סוג הדרישות, מחייבות תקציב גבוהה שעמדו לרשותנו. אנו מאמינים שבתקציב גבוה יותר, ניתן היה להשלים את כל היעדים במלואם.

11.3 הערכת הפתרון שנבחר

הפתרון שפיתחנו מציע מענה ייחודי וחדשני לבעה ממשית וגדלה בשטח חסימת תקשורת בין רחפנים לבין המפעיל שלהם, לצרכים ביוחניים, פרטיים ואזרחיים.

היבטי חדשנות וייחודיות:

המערכת מבוססת על שילוב מקורי בין שני רכיבי שידור שונים (LMX2492EVM 2.0 ו-LimeSDR Mini), תוך יצירת משק משתמש אחד וnoch שליטה בשלושה ערוצי חסימה שונים. הפתרון כולל שילוב בין sweep jamming ו-jamming barrage, מהלך שمدגים גמישות תפעולית ומענה לתרחישים מגוונים. הבדיקה בשימוש באנטנות ביוניות וההתאמת הספקטרלית המדוקת מראה חידוש שימושי במיושם חסימה ממוקדת ולא הרסנית (Soft Kill).

איכות הפתרון וההתאמתו לצורן:

המערכת עונה באופן איקוטי על הבעיה שהוגדרה בתחילת הפרויקט, חסימת שלושה ערוצי תקשורת קרייטיים בין הרחפן למפעיל. ערוצי GPS ו-2.4 GHz נבחנו בהצלחה בפועל מול רחפן והוא כהה חסימה אפקטיבית. משק המשתמש פותח תוך מחשבה על שליטה פשוטה, אינטואטיבית ומדוקת, כולל יכולות תצוגה בזמן אמת וממשק מיידע פנימי המדריך את המשתמש.

מחיר:

אחד היתרונות המרכזיים של הפתרון הוא העלות הנמוכה יחסית, תוך שמיירה על רמת ביצועים גבוההה. נעשה שימוש מושכל ברכיבים זמינים וזולים, חלקם ממוקורות לא מסחריות (כגון אנטנות שתוכבבו ונבנו על ידי פרופסור עלי לויון). התקציב המוגבל נובל בצדקה מיטבית ונitin להרחיב את הפתרון בעתיד במסגרת התקציב מוגדל.

אמינות ושרידות:

רכיבי החומרה נבחרו בקפידה תוך התאמת לדרישות הספק, תאימות אימפדנס ויכולת קירור. שימוש במשק גראפי מרכזי מקטין את רמת התלות של המש坦ש בפרטים טכניים ובכך מגביר את השRIDות התפעולית לאורך זמן. בנוסף, מבנה המערכת מאפשר תחזקה פשוטה ורחבת עתידית.

לסיכום, הפתרון שגובש במהלך הפרויקט הוא ייחודי, חדשני, ישים ויעיל הן מבחינת הביצועים בשטח, והן מבחינת מבנה, משק ועלות.

11.4 הצעות לעבודת המשך

- שילוב מערכת ליזוי רחפנים באופן אוטומטי: מצלה עם אלגוריתם לעיבוד תמונה, המזהה רחפן ומפעילה את מערכת השידור באופן אוטומטי בעת ג寥י מטרה.
- שדרוג המערכת לשידור עצמהגובה במיוחד, המאפשר שימוש באנטנות כל-ביוניות לצורך CISI מלא של סביבת העבודה ללא צורך בכיוון ידי.
- הוספה ממיר O-DC-DC המאפשרים הפעלת כל רכיבי המערכת ממוקור מתוך אחד לצורך פשוטות ויעילות תפעולית גבוהה יותר.

12 מקורות

סקירה ספרותית

Vladimir Matić, Vladimir Kosjer, Aleksandar Lebl, Branislav Pavić, Jovan Radivojević

“Methods for Drone Detection and Jamming”, 2019

PIETRO TEDESCHI , GABRIELE OLIGERI and ,ROBERTO DI PIETRO

“Leveraging Jamming to Help Drones Complete Their Mission” January 8, 2020

Mladen Mileusnić, Branislav Pavić, Verica Marinković-Nedelicki, Predrag Petrović, Dragan Mitić, Aleksandar Lebl

ANALYSIS OF JAMMING SUCCESSFULNESS AGAINST RCIED ACTIVATION WITH THE EMPHASIS ON SWEEP JAMMING

June 2019

סקר שוק

omnidirectional-anti-drone-jammer

PITBULLDrone JammerA compact, on-the-go anti-drone device

Counter-UAV Solutions. AARTOS™ Anti-drone Jammers

13 נספחים

13.1 קוד ממתק המשתמש

[GitHub](#) ל-

מבנה המחלקה: המחלקה Jamming יורשת מהמחלקה QMainWindow של Qt.

:Public

הconostructor המאתחל את הממתק, מחבר כפטורים לפונקציות ומבצע אתחול משתנים ודסטרקטור שאחראי על ניקוי זיכרון, יצירת תהליכי וסגורת חיבורים.

:Private Slots

סעיף זה כולל פונקציות (slots) המגיבות לאירועים מתוך ממתק המשתמש, כמו לחיצות על כפתורים, ניווט בין עמודים או התחלה תהליכי פנימיים.
הן כוללות: התחברות לרכבים, קונפיגורציה של רכבים, שליטה בשידור, שרטוטים גרפים וניווט בין החלונות.

:Private

סעיף זה כולל את כל המשתנים והפונקציות הפנימיות של המחלקה, שאינם נגישים למחלקה ומשמשים את המערכת לצורכי עבודה פנימית.

```
class Jamming : public QMainWindow
{
    Q_OBJECT

public:
    Jamming(QWidget *parent = nullptr);
    ~Jamming();

private slots:
    //SDR
    void connectLimeSDR(); // Connect to SDR
    void configureLimeSDR(); // Set frequency, gain, and sample rate
    void startTransmission();
    void stopTransmission();

    //usb2any_helper proces
    void connectusb2any_helper();

    //LMX24
    void connectLMX2492EVM24();
    void configureLMX2492EVM24();
    void startTransmission24();
    void stopTransmission24();

    //LMX58
    void connectLMX2492EVM58();
    void configureLMX2492EVM58();
    void startTransmission58();
    void stopTransmission58();

    //windows
    void openOperationalWindow();
    void openTechnicalWindow();
    void openGraphWindow();
    void openInformation();
    void returnToMain();

    //tips
    void functipGPS();
    void functip24();
    void functip58();

    //graph
    void TriangleSweep24();
    void FFT24();
    void TriangleSweep58();
    void FFT58();
    void noiseGPS();
    void FFTGPS();

private:
    Ui::JammingClass ui;

    //LimeSDR mini 2.0
    lms_device_t* LimeSDR;
    lms_stream_t tx_stream ;
    vector<float> tx_buffer;
    bool isTransmitting;
    std::thread txThread;
    int send_cnt;
    double centerFreq;
    double sampleRate;

    //LMX2492EVM
    QProcess* usb2anyProcess;
    HANDLE hPipe; // Store Named Pipe handle

    QChartView* ViewTriangle24;
    QChart* chartTriangle24;
    QChartView* ViewFFT24;
    QChart* chartFFT24;
    QChartView* ViewTriangle58;
    QChart* chartTriangle58;
    QChartView* ViewFFT58;
    QChart* chartFFT58;
    QChartView* ViewNoiseGPS;
    QChart* chartNoiseGPS;
    QChartView* ViewFTGPS;
    QChart* chartFTGPS;
    QLineSeries* seriesTriangle24;
    QLineSeries* seriesFFT24;
    QLineSeries* seriesTriangle58;
    QLineSeries* seriesFFT58;
    QLineSeries* seriesNoiseGPS;
    QLineSeries* seriesFTGPS;

    //SDR
    void sendData();
    vector<float> generateFilteredGaussianSignal();
    //windows
    void drawChart();
    void updateDiagram();
};
```

קובץ Jamming.cpp כולל את המימוש בפועל של כל הפונקציות שהוגדרו בקובץ הדרור **h**.
הקובץ אחראי על תפעול הלוגיקה המعشית של מערכת השיבושים:

usb2any_helper.exe: פונקציה זו אחראית על יצירת חיבור לתהיליך חיצוני בשם usb2any_helper אשר שולחת פקודות SPI אל מודול ה-PLL מסוג LMX2492EVM דרך ממשך USB. באמצעות שימוש במנגנון תקשורת בשם Name Pipe שהוא ערוץ תקשורת פנימי של Windows המיועד לחיבור בין שני תהיליכים.

```

void Jamming::connectusb2any_helper()
{
    usb2anyProcess = new QProcess(this);
    usb2anyProcess->start("usb2any_helper.exe");
    // Wait for the process to start (Max 10 seconds)
    if (!usb2anyProcess->waitForStarted(10000)) {
        QMessageBox::critical(this, "Connection Failed", "Unable to connect to usb2any_helper.\nPlease restart the program.");
        return;
    }
    if (!usb2anyProcess->waitForReadyRead(10000)) {
        QMessageBox::critical(this, "Connection Failed", "Unable to connect to usb2any_helper.\nPlease restart the program.");
        return;
    }
    hPipe = CreateFileA("\\\\.\\pipe\\usb2any_pipe", GENERIC_READ | GENERIC_WRITE, 0, NULL, OPEN_EXISTING, 0, NULL);
    if (hPipe == INVALID_HANDLE_VALUE) {
        QMessageBox::critical(this, "Connection Failed", "Unable to connect to usb2any_helper.\nPlease restart the program.");
        return;
    }
    char buffer[128];
    DWORD bytesRead;
    ReadFile(hPipe, buffer, sizeof(buffer), &bytesRead, NULL);
    buffer[bytesRead] = '\0';
    if (QString::fromStdString(buffer) != "connected")
    {
        QMessageBox::critical(this, "Connection Failed", "Unable to connect to usb2any_helper.\nPlease restart the program.");
    }
}

```

:connectLMX2492EVM24\connectLMX2492EVM58

פונקציה זו אחראית על שילוחת פקודות התחברות לרכיב LMX2492EVM באמצעות ערוץ התקשורת (Named Pipe) שנפתחת מול תהיליך העוזר usb2any_helper. הפעלה של הפקודה connectLMX2492EVM24" דרך הרצינור, ממתינה לתשובה אישור מהטהיליך, ואם התשובה מאשרת חיבור מוצלח, הממשך הגרפי מתעדכן בהתאם.

```

void Jamming::connectLMX2492EVM24()
{
    if (hPipe == INVALID_HANDLE_VALUE) {
        QMessageBox::warning(this, "Connection Failed", "Unable to establish connection with LMX2492EVM.\nPlease check the device and try again.");
        return;
    }
    string command = "connectLMX2492EVM24";
    DWORD bytesWritten;
    WriteFile(hPipe, command.c_str(), command.length(), &bytesWritten, NULL);
    char buffer[128];
    DWORD bytesRead;
    ReadFile(hPipe, buffer, sizeof(buffer), &bytesRead, NULL);
    buffer[bytesRead] = '\0';
    if (QString::fromStdString(buffer) == "SUCCESS:LMX2492EVM connected")
    {
        ui.label_LMX2492EVM->setText("Connected");
        ui.label_LMX2492EVM->setStyleSheet("QLabel { color : green; }");
        ui.label_transmission24->setText("Ready For Operation");
        ui.label_transmission24->setStyleSheet("QLabel { color : green; }");
        ui.pushButton_StartTransmissionWIFI24->setEnabled(true);
        updateDiagram();
        QMessageBox::information(this, "Connection Successful", "LMX2492EVM has been successfully connected.\nThe system is now ready for operation.");
    }
    else
        QMessageBox::warning(this, "Connection Failed", "Unable to establish connection with LMX2492EVM.\nPlease check the device and try again.");
}

```

:configureLMX2492EVM24/ConfigureLMX2492EVM58

פונקציה זו אחראית על חישוב ושליחת ערכים מתאימים לגיטרים של רכיב ה-**EVM**.
הערכים המוחשבים נקבעים בהתאם לפרמטרים שנבחרו על ידי המשתמש במסך הגרפי, ובפרט:

- תדר התחלה וסיום של ה-VCO.
- מושך זמן .Sweep

(updateConfiguration <Register> <Value>
לآخر ביצוע החישובים, נשלחות הפוקודות בתבנית טקסטואלית (updateConfiguration <Register> <Value>.
אל תהייך העוזר אשר מעביר את הערך ל-**EVM** LMX2492EVM באמצעות פוקודות SPI.

```
void Jamming::configureLMX2492EVM24()
{
    if (hPipe == INVALID_HANDLE_VALUE)
    {
        QMessageBox::warning(this, "Configuration Failed", "Unable to apply configuration settings.\nPlease check the connection or parameters.");
        return;
    }
    if (ui.label_LMX2492EVM->text() == "Disconnected")
    {
        QMessageBox::warning(this, "Configuration Failed", "Unable to apply configuration settings.\nPlease check the connection or parameters.");
        return;
    }

    double crystalClock = 100;
    double StartFrequencyVCO = ((double)ui.spinBox_StartFrequencyVCO->value());
    double StopFrequencyVCO = ((double)ui.spinBox_StopFrequencyVCO->value());
    double SweepDuration = ((double)ui.spinBox_SweepDuration->value());
    // Calculate RAMP_LEN
    int ramp_len = SweepDuration * crystalClock;
    uint8_t ramp_len_low = (ramp_len >> 0) & 0xFF;
    uint8_t ramp_len_high = (ramp_len >> 8) & 0xFF;
    // Calculate RAMP_INC
    double ramp1 = StopFrequencyVCO - StartFrequencyVCO;
    ramp1 = ramp1 / crystalClock;
    ramp1 = ramp1 * 16777216;
    ramp1 = ramp1 / (SweepDuration * crystalClock);
    double ramp2 = 1073741824 - ramp1;
    int ramp1_int = static_cast<int>(round(ramp1));
    int ramp2_int = static_cast<int>(round(ramp2));

    uint8_t low = (ramp1_int >> 0) & 0xFF;
    uint8_t middle = (ramp1_int >> 8) & 0xFF;
    uint8_t high = (ramp1_int >> 16) & 0xFF;

    uint8_t lowD = (ramp2_int >> 0) & 0xFF;
    uint8_t middleD = (ramp2_int >> 8) & 0xFF;
    uint8_t highD = (ramp2_int >> 16) & 0xFF;
    uint8_t addD = (ramp2_int >> 24) & 0xFF;

    //Calculate PLL_N for Start Frequency and dividerFraction
    double pll_n = StartFrequencyVCO / crystalClock;
    int pll_n_int = static_cast<int>(pll_n);
    pll_n = pll_n - pll_n_int;
    int dividerFraction = round(pll_n * 16777216);
    uint8_t pll_n_low = pll_n_int & 0xFF; // R16
    uint8_t pll_n_med = (pll_n_int >> 8) & 0xFF; // R17
    uint8_t FRACNum_low = (dividerFraction >> 0) & 0xFF; // R19
    uint8_t FRACNum_middle = (dividerFraction >> 8) & 0xFF; // R20
    uint8_t FRACNum_high = (dividerFraction >> 16) & 0xFF; // R21
    // Calculate R22
    int ramp_Limit_high = StartFrequencyVCO / crystalClock;
    ramp_Limit_high = 25 - ramp_Limit_high;
    vector<pair<int, int>> spiCommands = {
        {0x0013, FRACNum_low}, // R19
        {0x0014, FRACNum_middle}, // R20
        {0x0015, FRACNum_high}, // R21
        {0x0010, pll_n_low}, // R16 - PLL_N low bits
        {0x0011, pll_n_med}, // R17 - PLL_N high bits
        {0x0052, ramp_Limit_high}, // R22
        {0x0056, low}, // R86
        {0x0057, middle}, // R87
        {0x0058, high}, // R88
        {0x005D, lowD}, // R93
        {0x005E, middleD}, // R94
        {0x005F, highD}, // R95
        {0x0060, addD}, // R96
        {0x005A, ramp_len_low}, // R98
        {0x005B, ramp_len_high}, // R91
        {0x0061, ramp_len_low}, // R97
        {0x0062, ramp_len_high} // R98
    };
    for (const auto& [reg, val] : spiCommands) {
        stringstream ss;
        ss << "updateConfiguration " << hex << reg << " " << val;
        string command = ss.str();
        DWORD bytesWritten;
        WriteFile(hPipe, command.c_str(), command.length(), &bytesWritten, NULL);
        char buffer[128];
        DWORD bytesRead;
        ReadFile(hPipe, buffer, sizeof(buffer), &bytesRead, NULL);
        buffer[bytesRead] = '\0';
        if (QString::fromStdString(buffer) == "ERROR:LMX2492EVM update register failed")
        {
            QMessageBox::warning(this, "Configuration Failed", "Unable to apply configuration settings.\nPlease check the connection or parameters.");
            return;
        }
    }
    QMessageBox::information(this, "Configuration Successful", "Configuration settings have been successfully applied.\nThe system is ready for operation.");
}
```

*פירוט מלא על אופן חישוב הערכים
9.1.8 מופיע בסעיף

:startTransmission24/startTransmission58

פונקציה זו אחראית על הפעלת השידור של אוטות השיבוש בתחום התדרים המבוקש באמצעות מודול ה-**LMX2492EVM**.

.usb2any_helper (Named Pipe) לתחילה העזר הפונקציה שולחת את הפקודה דרך התקשורת. ומעדכנת את משך המשתמש לפי התגובה.

```

void Jamming::startTransmission24()
{
    if (hPipe == INVALID_HANDLE_VALUE) {
        QMessageBox::warning(this, "Transmission Failed", "Unable to start the transmission.\nPlease ensure the connection and configuration settings are correct.");
        return;
    }

    // Send a test command
    string command = "startTransmission24";
    DWORD bytesWritten;
    WriteFile(hPipe, command.c_str(), command.length(), &bytesWritten, NULL);
    // Read the response
    char buffer[128];
    DWORD bytesRead;
    ReadFile(hPipe, buffer, sizeof(buffer), &bytesRead, NULL);
    buffer[bytesRead] = '\0';

    // Display the response
    if (QString::fromStdString(buffer) == "SUCCESS:LMX2492EVM transmission started")
    {
        QMessageBox::information(this, "Transmission Started", "Jamming signals are now broadcasting on the Wi-Fi 2.4 band.");
        ui.label_transmission24->setText("Transmitting");
        ui.label_transmission24->setStyleSheet("QLabel { color : #89CFF0; }");
        ui.pushButton_StartTransmissionWIFI24->setEnabled(false);
        ui.pushButton_StopTransmissionWIFI24->setEnabled(true);
    }
    else
        QMessageBox::warning(this, "Transmission Failed", "Unable to start the transmission.\nPlease ensure the connection and configuration settings are correct.");
}

```

:stopTransmission24/stopTransmission58

פונקציה זו מפסיקה את השידור של אוטות השיבוש בתחום התדרים המבוקש על ידי שליחת פקdot הפקת שידור למודול ה-[LMX2492EVM](#). הפקודה נשלחת בטו מחוזות ("stopTransmission24 5") דרך Named Pipe לתהיליך `usb2any_helper`, שמבצע כתיבה לегистר R2 עם ערך 5 - פעולה שמשדרת פקdot Reset לרכיב.

```

void Jamming::stopTransmission24()
{
    if (hPipe == INVALID_HANDLE_VALUE) {
        QMessageBox::critical(this, "Stop Transmission Failed", "Unable to stop the transmission.\nPlease check the system status and try again.");
        return;
    }
    int reg = 0x0002; // R2
    int value = 5; // FOR RESET 5
    // Format command string
    stringstream ss;
    ss << "stopTransmission24" << hex << reg << " " << value;
    string command = ss.str();
    // Send to helper process
    DWORD bytesWritten;
    WriteFile(hPipe, command.c_str(), command.length(), &bytesWritten, NULL);
    // Read back result
    char buffer[128];
    DWORD bytesRead;
    ReadFile(hPipe, buffer, sizeof(buffer), &bytesRead, NULL);
    buffer[bytesRead] = '\0';
    if (QString::fromStdString(buffer) == "SUCCESS:LMX2492EVM transmission stoped")
    {
        QMessageBox::information(this, "Transmission Stoped", "Jamming signals have been successfully stopped on Wi-Fi 2.4 GHz.");
        ui_label_transmission24->setText("Ready For Operation");
        ui_label_transmission24->setStyleSheet("QLabel { color : green; }");
        ui_pushButton_StartTransmissionWIFI24->setEnabled(true);
        ui_pushButton_StopTransmissionWIFI24->setEnabled(false);
    }
    else
        QMessageBox::critical(this, "Stop Transmission Failed", "Unable to stop the transmission.\nPlease check the system status and try again.");
}

```

Usb2any_helper

ולאת שwhile בקוד של usb2any_helper היא הלב של התהיליך העדר, והוא אחראית לטפל בכל הפקודות שנשלחות מהתוכנה הראשית דרך Named Pipe.

שלב 1: המתנה לפקודות מהתוכנה הראשית
 קוראת מהצינור (hPipe) ומacha שיגע מידע מהתוכנה הראשית (GUI). ברגע שנשלחת פקודה, היא נקלטת לתוך buffer וmovebraת למחוזת .command

שלב 2: בדיקת סוג הפקודה
 הפקודה שנקלטת (למשל "startTransmission24" או "connectLMX2492EVM24") נבדקת אחת אחת באמצעות סדרת תנאים (if / else).
 עבור כל פקודה יש התנהגות שונה: התחברות לרכיב, התחלת שידור, עצירת שידור, עדכון קונפיגורציה, וכדומה.

```

while (ReadFile(hPipe, buffer, sizeof(buffer), &bytesRead, NULL)) {
    buffer[bytesRead] = '\0'; // Null termination
    string command(buffer);
    string response;

//WIFI 2.4 GHz
    if (command == "connectLMX2492EVM24") { ... }
    else if (command == "startTransmission24") {
        if (usb2anyHandle24 != NULL) {
            if (setupSPI() != 0) {
                response = "SPI configuration failed!";
            }
            else {
                for (const auto& reg : registers) {
                    if (!writeSPI(reg.first, reg.second)) {
                        response = "Failed to start transmission";
                        break;
                    }
                    else {
                        response = "SUCCESS:LMX2492EVM transmission started";
                    }
                    Sleep(10); // Small delay for stabilization
                }
                if (!writeSPI(2, 1))
                {
                    response = "Failed to start transmission2";
                    break;
                }
                else
                {
                    response = "SUCCESS:LMX2492EVM transmission started";
                }
            }
        }
        else {
            response = "Failed usb2anyHandle24 = NULL";
        }
    }
    else if (command.rfind("stopTransmission24", 0) == 0) {
        if (usb2anyHandle24 != NULL) {
            istringstream iss(command);
            string cmd;
            int reg, value;
            iss >> cmd >> hex >> reg >> value;
            if (!writeSPI(reg, value))
            {
                response = "Failed to stop transmission";
                break;
            }
            else
            {
                response = "SUCCESS:LMX2492EVM transmission stopped";
            }
        }
        else
            response = "Failed usb2anyHandle24 = NULL";
    }
}

```

שלב 3: ביצוע הפעולה המתאימה
 אם הפקודה תואמת - מותבצעת הקראיה לפונקציה המתאימה, למשל ,setupSPI,writeSPI או updateVector.
 לאחר ביצוע הפעולה התגובה נשמרת בתוך .response

שלב 4:
שליחת תגובה חזרה ל-GUI.

```

else if (command.rfind("updateConfiguration", 0) == 0) {
    istringstream iss(command);
    string cmd;
    int reg, value;
    iss >> cmd >> hex >> reg >> value;

    if (updateVector(reg, value) == true) {
        response = "SUCCESS:LMX2492EVM register updated";
    }
    else {
        response = "ERROR:LMX2492EVM update register failed";
    }
}

// Send response to Qt
WriteFile(hPipe, response.c_str(), response.length(), &bytesWritten, NULL);

// Cleanup before exiting
if (usb2anyHandle24) {
    u2aClose(usb2anyHandle24);
}
if (usb2anyHandle58) {
    u2aClose(usb2anyHandle58);
}
CloseHandle(hPipe);

```

:writeSPI

הfonקציה אחראית על שליחת פקודה בתיבה לרגיסטר ברכיב LMX2492EVM באמצעות ממשך SPI, דרך התקן USB2ANY.

מהלך הפעולה:

הfonקציה מקבלת שני פרמטרים:

- regIndex: בתובת הרגיסטר שאלוי נרצה לכתוב.
- value: הערך שנרצה לכתוב לתוך הרגיסטר.

בנייה מערך (spiData) בן שלושה בתים:

- בית ראשון - ה-MSB של בתובת הרגיסטר (בפועל תמיד 0).
- בית שני - ה-LSB של בתובת הרגיסטר.
- בית שלישי - הערך עצמו לכתיבה.

הפקודה מועברת לפונקציה u2aSPI_WriteAndRead, שהיא פונקציה מספרית TI המנהלת את התקשרות בפועל מול רכיב USB2ANY.

```
bool writeSPI(int regIndex, int value)
{
    if (!usb2anyHandle24) return false;

    uint8_t spiData[3];
    spiData[0] = (regIndex >> 8) & 0xFF; // MSB of register index (always 0)
    spiData[1] = regIndex & 0xFF;        // LSB of register index
    spiData[2] = value & 0xFF;          // Data byte

    int status = u2aSPI_WriteAndRead(usb2anyHandle24, sizeof(spiData), spiData);
    return (status == sizeof(spiData));
}
```

:setupSPI

פונקציה זו מבצעת את הגדרת ממשך ה- SPI של רכיב USB2ANYüber תקשורת מול ה- LMX2492EVM.

```
int setupSPI()
{
    int divider = 24000 / 400;
    int _DividerHigh = (divider >> 8) & 0xFF;
    int _DividerLow = divider & 0xFF;

    int status = u2aSPI_Control(
        usb2anyHandle24,
        SPI_Capture_On_Trailing_Edge, // CPHA = 0 (Leading Edge)
        SPI_Inactive_State_High,     // CPOL = 1
        SPI_MSB_First,              // MSB First
        SPI_8_Bit,                  // 8-bit SPI transfers
        SPI_With_Every_Packet,       // CS active for the whole packet
        SPI_CS_Active_High,         // CS active LOW
        _DividerHigh,
        _DividerLow                 // SPI Clock Divider (Adjustable)
    );
    return status;
}
```

:updateVector

פונקציה זו מבצעת עדכון של ערך וריסטר במבנה הנתונים המקומי וקטור של זוגות (regIndex, value) שמייצג את המצביע הנוכחי של רגיסטרים במודול EVM2492LMX2.

אופן הפעולה:

- מבצעת מעבר על כל זוג ברשימת הרגיסטרים.
- אם נמצא רgistר עם אינדקס תואם, (reg.first == regIndex) מתחדך לערך החדש.

```
bool updateVector(int regIndex, int value)
{
    if (!usb2anyHandle24) return false;
    for (auto& reg : registers) {
        if (reg.first == regIndex)
        {
            reg.second = value;
            return true;
        }
    }
    return false;
}
```

דוגמא למבנה נתונים של קונפיגורציה ברית מיוחד - ערכז 2.4GHz

```
vector<pair<int, int>> registers = {
    {0x008D, 0x00}, {0x008C, 0x27}, {0x008B, 0x10}, {0x008A, 0x00}, {0x0089, 0x00},
    {0x0088, 0x00}, {0x0087, 0x00}, {0x0086, 0x00}, {0x0085, 0x27}, {0x0084, 0x10},
    {0x0083, 0x00}, {0x0082, 0x00}, {0x0081, 0x00}, {0x0080, 0x00}, {0x007F, 0x00},
    {0x007E, 0x27}, {0x007D, 0x10}, {0x007C, 0x00}, {0x007B, 0x00}, {0x007A, 0x00},
    {0x0079, 0x00}, {0x0078, 0x00}, {0x0077, 0x27}, {0x0076, 0x10}, {0x0075, 0x00},
    {0x0074, 0x00}, {0x0073, 0x00}, {0x0072, 0x00}, {0x0071, 0x00}, {0x0070, 0x27},
    {0x006F, 0x10}, {0x006E, 0x00}, {0x006D, 0x00}, {0x006C, 0x00}, {0x006B, 0x00},
    {0x006A, 0x00}, {0x0069, 0x27}, {0x0068, 0x10}, {0x0067, 0x00}, {0x0066, 0x00},
    {0x0065, 0x00}, {0x0064, 0x00}, {0x0063, 0x00}, {0x0062, 0x27}, {0x0061, 0x10},
    {0x0060, 0x3F}, {0x005F, 0xFF}, {0x005E, 0xFA}, {0x005D, 0xE6}, {0x005C, 0x20},
    {0x005B, 0x27}, {0x005A, 0x10}, {0x0059, 0x00}, {0x0058, 0x00}, {0x0057, 0x05},
    {0x0056, 0x92}, {0x0055, 0x00}, {0x0054, 0x00}, {0x0053, 0x00}, {0x0052, 0x01},
    {0x0051, 0x5C}, {0x0050, 0x28}, {0x004F, 0xF5}, {0x004E, 0xFF}, {0x004D, 0x47},
    {0x004C, 0xAE}, {0x004B, 0x15}, {0x004A, 0x00}, {0x0049, 0x00}, {0x0048, 0x00},
    {0x0047, 0x00}, {0x0046, 0x0A}, {0x0045, 0x00}, {0x0044, 0xFF}, {0x0043, 0x47},
    {0x0042, 0xAE}, {0x0041, 0x15}, {0x0040, 0x00}, {0x003F, 0x01}, {0x003E, 0x5C},
    {0x003D, 0x28}, {0x003C, 0xF5}, {0x003B, 0x00}, {0x003A, 0x01}, {0x0039, 0x00},
    {0x002D, 0x00}, {0x002C, 0x00}, {0x002B, 0x00}, {0x002A, 0x00}, {0x0029, 0x00},
    {0x0028, 0x00}, {0x0027, 0x52}, {0x0026, 0x18}, {0x0025, 0x10}, {0x0024, 0x08},
    {0x0023, 0x41}, {0x0022, 0x04}, {0x0021, 0x20}, {0x0020, 0x00}, {0x001F, 0x32},
    {0x001E, 0xA}, {0x001D, 0x00}, {0x001C, 0x1F}, {0x001B, 0x08}, {0x001A, 0x00},
    {0x0019, 0x01}, {0x0018, 0x00}, {0x0017, 0x03}, {0x0016, 0xE8}, {0x0015, 0x00},
    {0x0014, 0x00}, {0x0013, 0x00}, {0x0012, 0x2C}, {0x0011, 0x00}, {0x0010, 0x18},
    {0x000F, 0x00}, {0x000E, 0x00}, {0x000D, 0x00}, {0x000C, 0x00}, {0x000B, 0x00},
    {0x000A, 0x00}, {0x0009, 0x00}, {0x0008, 0x00}, {0x0007, 0x00}, {0x0006, 0x00},
    {0x0005, 0x00}, {0x0004, 0x00}, {0x0003, 0x00}, {0x0002, 0x01}, {0x0001, 0x00},
    {0x0000, 0x18}
};
```

:connectLimeSDR

פונקציה זו אחראית על יצירת חיבור בין התוכנה למודול LimeSDR Mini 2.0 לצורך שידור אותות בתחום תדר ה-GPS, באמצעות קרייה לפונקציה `LMS_Open` מתוך ספריית LimeSuite לצורך פתיחת תקשורת עם הרכיב.

```
void Jamming::connectLimeSDR()
{
    if (LMS_Open(&LimeSDR, nullptr, nullptr) != 0) {
        QMessageBox::warning(this, "Connection Failed", "Unable to establish connection with LimeSDR Mini 2.0.\nPlease check the device and try again.");
        return;
    }
    ui.label_LimeSDR->setText("Connected");
    ui.label_LimeSDR->setStyleSheet("QLabel { color : green; }");
    ui.label_transmissionGPS->setText("Ready For Operation");
    ui.label_transmissionGPS->setStyleSheet("QLabel { color : green; }");
    ui.pushButton_StartTransmissionGPS->setEnabled(true);
    updateDiagram();
    QMessageBox::information(this, "Connection Successful", "LimeSDR Mini 2.0 has been successfully connected.\nThe system is now ready for operation.");
}
```

:configureLimeSDR

פונקציה זו מבצעת את קונפיגורציה השידור עבור רכיב ה-LimeSDR Mini 2.0, המבוססת על הפרמטרים שנבחרו במשך המשתמש.

מהלך הפעולה:

- תדר המרכז של השידור, מחושב ממוצע בין התדר ההתחלתי והסופי.
- רוחב הפס של השידור, מחושב בהפרש בין התדרים.
- נוצרת מערך דגימות (tx) של אות גאומétrיא מסוכן, באמצעות קרייה לפונקציה `generateFilteredGaussianSignal`.
- נקבע מספר הדגימות לשידור `(send_cnt)`.

```
void Jamming::configureLimeSDR()
{
    if (LimeSDR == nullptr) {
        QMessageBox::warning(this, "Configuration Failed", "Unable to apply configuration settings.\nPlease check the connection or parameters.");
        return;
    }

    centerFreq = ((ui.spinBox_LIME_StopFreq->value() + ui.spinBox_LIME_StartFreq->value()) / 2) * 1e6;
    sampleRate = (ui.spinBox_LIME_StopFreq->value() - ui.spinBox_LIME_StartFreq->value()) * 1e6;
    tx_buffer = generateFilteredGaussianSignal();
    send_cnt = tx_buffer.size() / 2;
    QMessageBox::information(this, "Configuration Successful", "Configuration settings have been successfully applied.\nThe system is ready for operation.");
}
```

:startTransmission

פונקציה זו מבצעת את כל שלבי ה 初始化 והפעלה של שידור אותות החסימה בתדר GPS באמצעות רכיב LimeSDR Mini 2.0.

מהלך הפעולה:

- מtbodyן את הרכיב באמצעות `LMS_Init`.
- מופעל ערוץ השידור (TX) - ערוץ 0.
- מוגדר קצב הדגימה (sample rate) כפי שוחשב מוקדם יותר בקונפיגורציה.
- מוגדר תדר נושא (LO frequency) לפי תדר מركזי שנבחר.
- מוגדר רמת הגבר (gain) לנורמליזציה של עצמת השידור.
- מtbodyן צויל הרכיב באמצעות `LMS_Calibrate` על מנת להתאים את ההגדרות להימנעות מהפרעות.
- מוגדר זרם השידור (tx), כולל: גודל FIFO ופורמט הנתונים כ-`float32`.
- זרם השידור מאותחל ומופעל בפועל.
- נוצרת תיבת שידור ברקע (thread) אשר רצה בולאה ומעבירה נתוני שידור לפולט (`sendData`), כל עוד `.isTransmitting == true`.

```
void Jamming::startTransmission()
{
    if (LMS_Init(LimeSDR) != 0) {
        QMessageBox::warning(this, "Transmission Failed", "Unable to start the transmission.\nPlease ensure the connection and configuration settings are correct.");
        return;
    }
    if (LMS_EnableChannel(LimeSDR, LMS_CH_TX, 0, true) != 0) {
        QMessageBox::warning(this, "Transmission Failed", "Unable to start the transmission.\nPlease ensure the connection and configuration settings are correct.");
        return;
    }
    if (LMS_SetSampleRate(LimeSDR, sampleRate, 0) != 0) {
        QMessageBox::warning(this, "Transmission Failed", "Unable to start the transmission.\nPlease ensure the connection and configuration settings are correct.");
        return;
    }
    if (LMS_SetLOFrequency(LimeSDR, LMS_CH_TX, 0, centerFreq) != 0) {
        QMessageBox::warning(this, "Transmission Failed", "Unable to start the transmission.\nPlease ensure the connection and configuration settings are correct.");
        return;
    }
    if (LMS_SetNormalizedGain(LimeSDR, LMS_CH_TX, 0, 0.9) != 0) {
        QMessageBox::warning(this, "Transmission Failed", "Unable to start the transmission.\nPlease ensure the connection and configuration settings are correct.");
        return;
    }
    if (LMS_Calibrate(LimeSDR, LMS_CH_TX, 0, sampleRate, 0) != 0) {
        QMessageBox::warning(this, "Transmission Failed", "Unable to start the transmission.\nPlease ensure the connection and configuration settings are correct.");
        return;
    }
    tx_stream.channel = 0;
    tx_stream.fifoSize = 512 * 1024; // Increased FIFO size to support higher bandwidth
    tx_stream.throughputVsLatency = 0.5;
    tx_stream.dataFmt = lms_stream_t::LMS_FMT_F32;
    tx_stream.isTx = true;
    if (LMS_SetupStream(LimeSDR, &tx_stream) != 0) {
        QMessageBox::warning(this, "Transmission Failed", "Unable to start the transmission.\nPlease ensure the connection and configuration settings are correct.");
        return;
    }
    if (LMS_StartStream(&tx_stream) != 0) {
        QMessageBox::warning(this, "Transmission Failed", "Unable to start the transmission.\nPlease ensure the connection and configuration settings are correct.");
        return;
    }
    isTransmitting = true;
    if (!txThread) {
        txThread = new std::thread([this]() {
            while (isTransmitting) {
                sendData();
            }
        });
    }
    isTransmitting = true;
    QMessageBox::information(this, "Transmission Started", "Jamming signals are now broadcasting the GPS band.");
    ui.label_transmissionGPS->setText("Transmitting");
    ui.label_transmissionGPS->setStyleSheet("QLabel { color : #80C0FF; }");
    ui.pushButton_StartTransmissionGPS->setEnabled(false);
    ui.pushButton_StopTransmissionGPS->setEnabled(true);
}
```

:sendData

פונקציה זו מבצעת את פעולה השידור בפועל של האות לרכיב LimeSDR Mini 2.0 באמצעות זרם השידור (tx_stream) שהוגדר מראש.

```
void Jamming::sendData()
{
    int ret = LMS_SendStream(&tx_stream, tx_buffer.data(), send_cnt, nullptr, 1000);
    if (ret <= 0) {
        stopTransmission();
    }
}
```

:stopTransmission

פונקציה זו אחראית על עצירה בטוחה ומלאה של תהליך השידור שנעשה דרך רכיב ה-LimeSDR Mini 2.0. תהליך העבודה:

- מבוצעת בדיקה שהרכיב במצב תקין על ידי (LMS_Init).
- אם מתקיים שידור פעיל (isTransmitting = true), ממבצעת עצירה:
- ביטול הדגל .isTransmitting
- המתנה לסיום תהליכי השידור (באמצעות join על ה-thread) ושחררו מהזיכרון.
- הפסקת זרם הנתונים (LMS_StopStream) והשמדתו (LMS_DestroyStream).
- ניתוק ערוץ השידור (LMS_EnableChannel(false)).

```
void Jamming::stopTransmission()
{
    if (LMS_Init(LimeSDR) != 0) {
        QMessageBox::critical(this, "Stop Transmission Failed", "Unable to stop the transmission.\nPlease check the system status and try again.");
        return;
    }
    if (isTransmitting)
    {
        isTransmitting = false;
        if (txThread) {
            txThread->join();
            delete txThread;
            txThread = nullptr;
        }
        // Stop and destroy the stream
        LMS_StopStream(&tx_stream);
        LMS_DestroyStream(LimeSDR, &tx_stream);

        if (LMS_EnableChannel(LimeSDR, LMS_CH_TX, 0, false) != 0) {
            QMessageBox::critical(this, "Stop Transmission Failed", "Unable to stop the transmission.\nPlease check the system status and try again.");
        }
        QMessageBox::information(this, "Transmission Stopped", "Jamming signals have been successfully stopped on GPS band.");
        ui.label_transmissionGPS->setText("Ready For Operation");
        ui.label_transmissionGPS->setStyleSheet("QLabel { color : green; }");
        ui.pushButton_StartTransmissionGPS->setEnabled(true);
        ui.pushButton_StopTransmissionGPS->setEnabled(false);
    }
}
```

generateFilteredGaussianSignal

לצורך יצירת אות השיבוש בערוצ GPS, שנעשה באמצעות שיטת Barrage Jamming, היה علينا ליצור אות בעל מאפיינים של רעש גאוסיאני וחב-ספקטראום, אשר יבסה את תחום התדרים של ה GPS ויספק חסימה רציפה בכל הרוחב הרלוונטי.

האלגוריתם כולם מאפשר שליטה מלאה בתכונות הרעש ובמיקומו בספקטרום, ומאפשר לנו ליצור שידור אינטנסיבי שמנוע לבסות באופן רציף את תחום התדרים של GPS ולמנוע קליטה תקינה שלאותות לוין על ידי הרחפן.

```
vector<float> Jamming::generateFilteredGaussianSignal()
{
    const int buffer_size = 16384;
    const float amplitude = 5.0f;
    const float cutoffFreq = 10e6;
    const float transitionWidth = 50e3;
    const int filter_size = 512;
    vector<float> noise(buffer_size * 2);
    // Fix 1: Use Fixed Seed for Noise Stability
    default_random_engine generator(42);
    normal_distribution<float> distribution(0.0f, amplitude);
    for (int i = 0; i < buffer_size; i++) {
        noise[2 * i] = distribution(generator);
        noise[2 * i + 1] = distribution(generator);
    }

    vector<float> filter(filter_size);
    float beta = 6.76;

    for (int i = 0; i < filter_size; i++) {
        float n = (i - (filter_size - 1) / 2.0f);
        if (n == 0) {
            filter[i] = 2 * (cutoffFreq / sampleRate);
        }
        else {
            float sinc_val = sin(2 * M_PI * cutoffFreq * n / sampleRate) / (M_PI * n);
            float window_val = 0.54 - 0.46 * cos((2 * M_PI * i) / (filter_size - 1));
            filter[i] = sinc_val * window_val;
        }
    }
    // Fix 2: Ensure Filter Normalization
    float sum = 0;
    for (float v : filter) sum += v;
    if (sum != 0) {
        for (float& v : filter) v /= sum;
    }
    // Fix 3: Avoid Power Loss During Filtering
    vector<float> filtered_noise(buffer_size * 2, 0);
    for (int i = 0; i < buffer_size; i++) {
        for (int j = 0; j < min(filter_size, i + 1); j++) {
            filtered_noise[2 * i] += noise[2 * (i - j)] * filter[j];
            filtered_noise[2 * i + 1] += noise[2 * (i - j) + 1] * filter[j];
        }
    }
    // Fix 4: Use Phase Accumulator for Stable Modulation
    float phase = 0.0f;
    float phase_increment = 2.0f * M_PI * (centerFreq / sampleRate);
    for (int i = 0; i < buffer_size; i++) {
        float cosPhase = cos(phase);
        float sinPhase = sin(phase);

        float I = filtered_noise[2 * i];
        float Q = filtered_noise[2 * i + 1];

        filtered_noise[2 * i] = I * cosPhase - Q * sinPhase;
        filtered_noise[2 * i + 1] = I * sinPhase + Q * cosPhase;

        phase += phase_increment;
        if (phase > 2.0f * M_PI) phase -= 2.0f * M_PI;
    }
    return filtered_noise;
}
```

13.2 פוסטר הפרויקט



Three Channel Drone Jammer

1. Background

The project provides a defensive solution against unauthorized drone activity. The system uses a Soft Kill approach, disrupting the communication between the drone and its operator instead of physically destroying the drone. This method prevents harm to the surroundings and avoids the use of destructive measures that could lead to unintended damage.

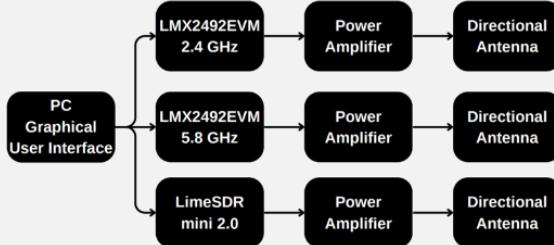
2. Objective

- Jamming three communication channels: 2.4 GHz, 5.8 GHz, and GPS.
- Building an MVP (Minimum Viable Product).
- Develop a Graphical User Interface.

3. Functional structure

This block diagram illustrates the architecture of our system for jamming drones:

- PC GUI – Controls and configures system parameters such as frequency range and activation timing.
- LMX2492EVM – Transmits a triangle wave jamming signal in the 2.4 GHz and 5.8 GHz ISM band.
- LimeSDR Mini 2.0 – Generates wideband noise signals targeting GPS L1 band.
- Power Amplifiers – Amplify the output RF signals from the LimeSDR and LMX modules to the required transmission levels.
- Directional Antennas – Transmit the amplified signals with a focused beam toward the drone.



Michel Albrecht, Guy Goldman
Advisor: Tzvika Rosenblum

4. Design and products

Our system provides protection using two jamming techniques:

Sweep Jamming (used for Wi-Fi channels 2.4 GHz and 5.8 GHz):

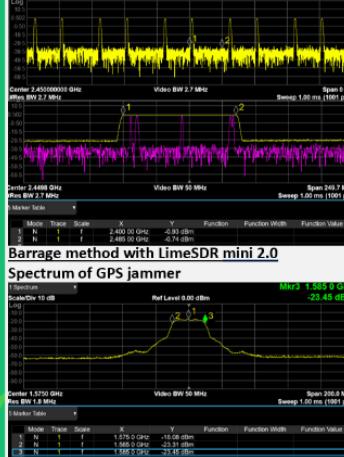
- Transmits a signal that rapidly sweeps across a defined frequency range.
- The drone signals span 20 MHz and hop every few hundred microseconds.
- To effectively jam, our system is designed to sweep across the full Wi-Fi band several times faster than the drone's hop rate.

Barrage jamming (used for GPS):

- Transmits wideband noise across an entire frequency range.
- In our case, the jamming signal fully covers the ~20 MHz L1 GPS band.

Sweep method with LMX2492EVM

Zero span, Spectrum of WIFI jammer



Graphical User Interface:

- Controls all jamming configurations.
- Manages communication with the hardware components.
- Enables independent or simultaneous channel control.
- Supports rapid adjustments during system operation.



5. Summary and conclusions

- We successfully developed a working MVP prototype capable of jamming all targeted drone communication channels (2.4 GHz, 5.8 GHz, and GPS).
- All hardware components communicate seamlessly through the GUI, that provides full control and real-time monitoring of the system.
- Our system integrates both sweep and barrage jamming techniques for flexible and effective operation.

13.3 דפי נתונים

[LMX2492EVM Evaluation Module](#)

[LMX2492](#)

[LMX2594 EVM Instructions](#)

[USB2ANY Interface Adapter](#)

[HackRF One](#)

[LimeSDR Mini 2.0](#)

[LMS7002M Data Sheet](#)

[LMS7002M](#)

[Effective GPS Jamming Techniques for UAVs Using Low-Cost SDR Platforms](#)

[DJI Mini 2](#)

[ZRL-2300+](#)

[CVCO55CXT-5700-5900](#)

[CVCO55CC-2328-2536](#)