

אוניברסיטת בן-גוריון בנגב

Ben-Gurion University of the Negev



בית הספר להנדסת חשמל ומחשבים

דו"ח תיעוד פרויקט גמר קורס "מבנה מחשבים ספרתיים"

361-1-4191

תכנון ומימוש מערכת בקרה למכונה מבוססת מנוע
צעד בשליטה ידנית ומרחוק

Tomer Abram 208931691

Noam Okun 206784845

2.9.2024

הגדרת ומטרת הפרויקט

הפרויקט הנוכחי מהווה את פרויקט הסיכום של הקורס "מבנה מחשבים ספרתיים", והוא נועד לאחד ולהדגים את הידע הרב שנצבר במהלך הסמסטר ובמהלך הפרויקט עצמו על ידי מימוש מערכת בקרה מורכבת. במסגרת הפרויקט, נדרשנו לפתח ולממש מערכת בקרה למכונה מבוססת מנוע צעד, תוך שילוב שליטה ידנית באמצעות ג'ויסטיק אנלוגי, לצד שליטה מרחוק ממחשב אישי באמצעות ערוץ תקשורת טורית.

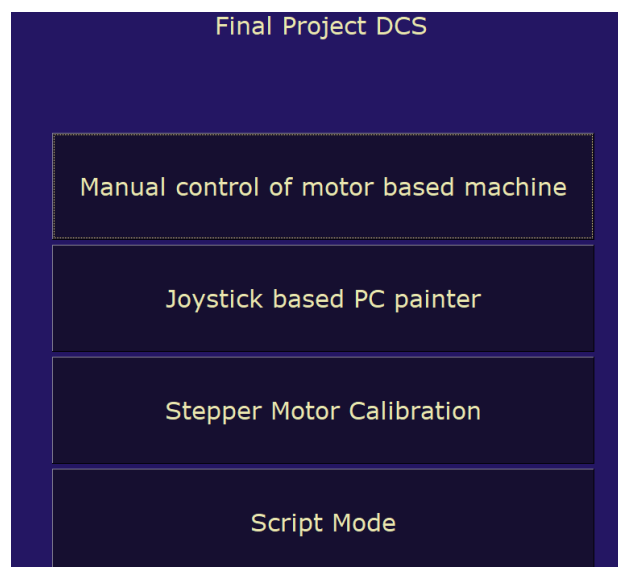
המערכת נבנתה על גבי בקר MSP430G2553 ושפת התכנות שנבחרה למימוש הפרויקט היא שפת C, תוך שימוש בסביבת הפיתוח CCS. הבקר מבצע מגוון פעולות שיפורטו בהמשך, במקביל לממשק משתמש במחשב האישי שמאפשר שליטה בפעולות הבקר בצורה נוחה ואינטואיטיבית, הן באמצעות ממשק משתמש גרפי (GUI) והן באמצעות פקודות טוריות. הממשק הגרפי מאפשר חוויית משתמש מלאה ונותן אפשרות לקבוע פרמטרים, לשלוח פקודות ולבצע מעקב אחר הביצועים של המערכת בצורה חזותית וברורה.

הפרויקט הזה מציג תהליך שלם של תכנון, מימוש ובדיקת מערכת משובצת, והוא מדגיש את החשיבות של שליטה מדויקת, תגובה מהירה והתחשבות במגבלות ההנדסיות של עבודה בזמן אמת.

תיאור קצר של הפרויקט

המערכת כוללת מספר רכיבים עיקריים:

ראשית באמצעות GUI המכיל כפתורים עבור פקודות מסויימות, הממשק בין המשתמש לבקר בצד של PC יבוא לידי ביטוי בתמונה הבאה, כאשר כל כפתור ה-PC שולח אות תקשורת לבקר ובהתאם הבקר ישנה את מצב ב-FSM שלו:



1. **שליטה ידנית באמצעות ג'ויסטיק:** מערכת זו מאפשרת למשתמש לשלוט בזווית הסיבוב של מנוע הצעד באמצעות ג'ויסטיק אנלוגי או שליטה של העכבר מחשב באמצעות הזווית של הג'ויסטיק. תנועות הג'ויסטיק מתורגמות לפקודות אשר מועברות לבקר ומניעות את המנוע או את העכבר המחשב לזווית המתאימה. המערכת מתוכננת לעבודה בחוג פתוח, כאשר הבקר מקבל את המידע מהג'ויסטיק ומבצע את הפקודות באופן רציף.

2. **שליטה מרחוק ממחשב אישי:** בנוסף לשליטה הידנית, המערכת תומכת בשליטה מרחוק באמצעות מחשב אישי המחובר לבקר דרך ערוץ תקשורת טורית. (UART) המחשב מצויד בממשק משתמש גרפי (GUI) המאפשר לשלוח פקודות לבקר, לעקוב אחר מצב המערכת בזמן אמת ולבצע שינויים על פי הצורך. הממשק הגרפי מאפשר שליטה פשוטה ונוחה, ומאפשר למשתמשים שאינם טכניים להתנהל עם המערכת בקלות.

3. **כיול מנוע הצעד:** המערכת כוללת יכולת כיול למנוע הצעד, המאפשרת למשתמש לבצע כיול עצמי של המנוע כדי להבטיח את דיוק התנועות. התהליך כולל חישוב של זווית הצעד המדויקת של המנוע והתאמה של הבקר לערכים הנמדדים.

4. **מצב Script:** אחד המרכיבים החשובים בפרויקט הוא היכולת להריץ פקודות מגרפיות (scripts) אשר מאוחסנות בבקר. המשתמש יכול לשלוח קבצים המכילים פקודות מראש, והבקר יבצע את הפקודות בסדר הנקבע על פי תפריט המשתמש. מערכת זו מאפשרת אוטומציה של תהליכים וביצוע פעולות מורכבות בקלות וביעילות.

הפרויקט משלב אתגרי תכנון ויישום, ומציג את היכולת לשלב בין חומרה ותוכנה לצורך פיתוח מערכת משובצת מתקדמת. המערכת מתוכננת כך שתהיה עמידה ואמינה, ותוכל להתמודד עם תנאי עבודה משתנים תוך שמירה על דיוק וזמן תגובה מהירים. כל אלה הופכים את הפרויקט לאתגר הנדסי מרתק ולדוגמה מצוינת לשילוב של ידע תיאורטי ויכולות מעשיות בפיתוח מערכות בקרה מתקדמות.

תיאור כללי של ביצועי החומרה והתוכנה: חלוקת העבודה בין החומרה לתוכנה

בפרויקט זה, המשלב מערכת בקרה למכונה מבוססת מנוע צעד עם שליטה ידנית ומרחוק, הייתה חשיבות רבה לחלוקת העבודה היעילה בין החומרה לתוכנה, כדי להבטיח ביצועים מיטביים ותגובה בזמן אמת.

ביצועי החומרה:

החומרה בפרויקט מבוססת על בקר, MSP430G2553 הכולל מודולים שונים כמו ADC, UART, וטיימרים. כל אחד מהמודולים הללו משמש לתמיכה בביצועים שונים של המערכת:

1. **ADC (Analog to Digital Converter):** ה-ADC משמש לדגימה של ערכי המתח הנכנסים מג'ויסטיק האנלוגי, כדי לחשב את הזווית הרצויה של מנוע הצעד. דגימות אלו נשלחות דרך ה-ADC לבקר, שם הן מעובדות לצורך קביעת כיוון וזווית הסיבוב של המנוע.

2. **UART (Universal Asynchronous Receiver/Transmitter):** ערוץ התקשורת הטורית UART משמש לתקשורת בין הבקר לבין המחשב האישי. המחשב שולח פקודות לבקר ומקבל ממנו מידע בזמן אמת. ערוץ זה מאפשר שליטה מרחוק וביצוע פעולות במערכת תוך שימוש בממשק גרפי על גבי המחשב.

3. **טיימרים:** הטיימרים במערכת משמשים ליצירת השהיות מדויקות בתהליכים השונים, כגון בעת סיבוב מנוע הצעד או בזמן קבלת פקודות מהג'ויסטיק. הם גם משמשים לניהול מצבי שינה והתעוררות של הבקר, כדי להבטיח יעילות אנרגטית וזמן תגובה מהיר.

ביצועי התוכנה:

התוכנה בפרויקט נכתבה בשפת C, תוך שימוש בסביבת הפיתוח CCS. התוכנה אחראית לניהול כל הרכיבים החומריים ולתיאום ביניהם.

התוכנה בפרויקט זה מהווה את הלב הפועם של המערכת, ומנהלת את כל תהליכי הקלט, העיבוד והפלט בצורה אינטגרטיבית ויעילה. באמצעות התוכנה, הבקר מקבל נתונים מהג'ויסטיק, מעבד אותם בזמן אמת, ושולח פקודות למנוע הצעד לביצוע תנועות מדויקות או למחשב למצב של הצייר על מנת להזיז את העכבר למיקום הרצוי. התוכנה מנהלת את התקשורת עם המחשב האישי, מאפשרת שליטה מרחוק דרך ממשק משתמש גרפי, ומספקת תמיכה בהרצת סקריפטים מורכבים. כל זאת תוך שמירה על אופטימיזציה גבוהה לניצול מיטבי של משאבי המערכת וביצועים חלקים בזמן אמת.

בפרויקט ביצענו מספר משימות:

ניהול קלט מהג'ויסטיק – manual control of motor based machine

המשימה הראשונה בפרויקט התמקדה ביצירת מערכת המאפשרת שליטה ידנית מדויקת במנוע צעד באמצעות ג'ויסטיק אנלוגי. במערכת זו, המשתמש יכול לשלוט באופן ישיר בזווית הסיבוב של המנוע על ידי הזזת הג'ויסטיק לכל כיוון. התנועות של הג'ויסטיק מתורגמות לפקודות שליטה שמועברות ישירות לבקר, אשר בתורו מניע את מנוע הצעד לזווית המתאימה.

המערכת פועלת במצב של בקרה בחוג פתוח, (open loop) כלומר, אין משוב מהמנוע לבקר על מיקומו הנוכחי. כל תנועה של הג'ויסטיק גורמת למנוע להתחיל מנקודת ההתחלה שנקבעה ולהתאים את הזווית שלו בהתאם לקלט שמתקבל. בגלל שמדובר בבקרה בחוג פתוח, נדרש מהמשתמש לוודא שהמנוע נמצא בנקודת התחלה מדויקת, אחרת ייתכן שיתקבלו תנועות לא מדויקות.

הג'ויסטיק האנלוגי מחובר לבקר דרך יחידת ADC אשר ממירה את המתח האנלוגי שמתקבל מהג'ויסטיק לערכים דיגיטליים. הבקר מקבל את הערכים הללו, מעבד אותם, ומחשב את הזווית החדשה שהמנוע צריך להגיע אליה. כמו כן, בתפריט המערכת ישנה אפשרות למשתמש לאפס את המיקום הנוכחי של המנוע, כלומר לקבוע את זווית האפס מחדש לפני תחילת הסיבוב.

בנוסף, המשתמש יכול לשלוט בקצב הסיבוב של המנוע ובכיוון הסיבוב (עם כיוון השעון או נגד כיוון השעון). במהלך פיתוח המשימה, נעשה שימוש בטיימרים בתוך הבקר כדי להבטיח שהמנוע יזוז בצורה חלקה וללא עצירות מיותרות, תוך שמירה על רמת דיוק גבוהה.

צייר מבוסס ג'ויסטיק במחשב אישי

המשימה השנייה בפרויקט עסקה במימוש יישום צייר אינטראקטיבי על גבי מסך המחשב, הנשלט כולו באמצעות ג'ויסטיק. מטרת המשימה הייתה לאפשר למשתמש לצייר על המסך בצורה חופשית על ידי הזזת הג'ויסטיק ולבחור בין שלושה מצבי פעולה: כתיבה, מחיקה ומצב ניטרלי.

במצב כתיבה, סמן העכבר משמש כחוד עיפרון, כאשר המשתמש יכול לצייר קווים וצורות על המסך על ידי תנועות הג'ויסטיק. במצב מחיקה, הסמן משמש כמחק, והמשתמש יכול למחוק חלקים מהציור שכבר נעשה. במצב ניטרלי, הסמן נע על המסך מבלי לבצע כל פעולה – כלומר, ניתן להזיז את הסמן למיקום הרצוי מבלי לצייר או למחוק דבר.

הג'ויסטיק האנלוגי מחובר לבקר, אשר דוגם את המתח הנכנס מרגלי הג'ויסטיק ושולח את המידע בצורה רציפה למחשב האישי באמצעות ערוץ תקשורתה UART. המחשב, שמריץ את יישום הצייר, מקבל את הנתונים מהבקר ומעדכן

את מיקום הסמן בהתאם לערכים שהתקבלו. השימוש בטכנולוגיית זמן אמת קשיח (Hard Real-Time) הבטיח שהזזת הסמן על המסך תתבצע בצורה חלקה וללא עיכובים.

בנוסף, הוספנו פונקציה שמזהה לחיצות על כפתור בג'ויסטיק, אשר משמשת כפקודת מעבר בין המצבים השונים (כתיבה, מחיקה, ניטרלי). בכל פעם שהמשתמש לוחץ על הכפתור, הבקר שולח קוד מיוחד למחשב שמורה ליישום הצייר לשנות את מצב הפעולה בהתאם. ההפרדה בין המצבים נעשתה באמצעות משתנה מצב גלובלי שהוגדר בבקר.



כיול מנוע צעד

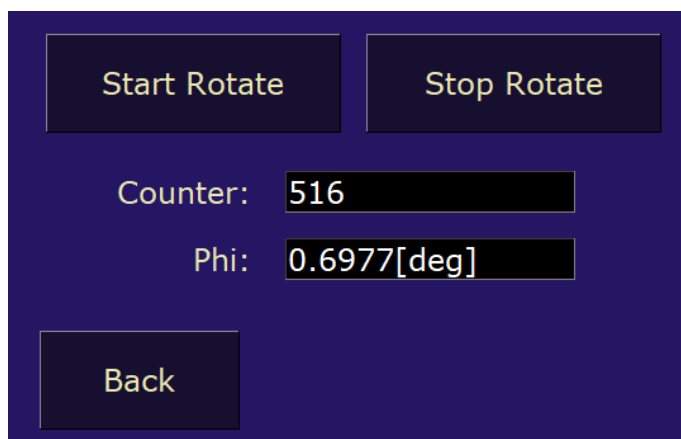
המשימה השלישית בפרויקט נועדה לאפשר למשתמש לבצע כיול מדויק של מנוע הצעד כדי להבטיח שהמערכת תפעל ברמת דיוק גבוהה ככל האפשר. כיול מנוע הצעד הוא תהליך שבו נקבעת הזווית המדויקת של כל צעד שהמנוע מבצע, וזאת כדי להבטיח שהתנועות שתתבצענה על פי פקודות המשתמש יהיו תואמות למה שמצופה מהמערכת.

במשימה זו, המערכת מאפשרת למשתמש לבצע סיבוב מלא של מנוע הצעד (360 מעלות) ומוודדת את כמות הצעדים שהמנוע מבצע עד להשלמת הסיבוב. המידע על כמות הצעדים שנמדדו וגודל זווית הצעד מוצג על גבי מסך המחשב לאחר השלמת התהליך.

המנוע מופעל במצב חוג פתוח, כך שהבקר אינו מקבל משוב ישיר מהמנוע על מיקומו הנוכחי. במקום זאת, המשתמש מקבל חיווי ויזואלי מהמערכת על סיום הסיבוב המלא. במהלך תהליך הכיול, המנוע מבצע צעדים קטנים ומדויקים, בעוד שהבקר עוקב אחר מספר הצעדים הנדרש להשלמת הסיבוב ומחשב את זווית הצעד המדויקת על פי הנתונים שהתקבלו.

לאחר שהכיול הושלם, המערכת שומרת את הערכים של הבקר כך שהם יהיו זמינים לשימוש בכל סיבוב עתידי של המנוע. בכך, כיול זה מבטיח שהמנוע

יבצע תנועות מדויקות בכל הפעלה, ללא תלות בשינויים קטנים בין מנועים שונים או בתנאי הסביבה.



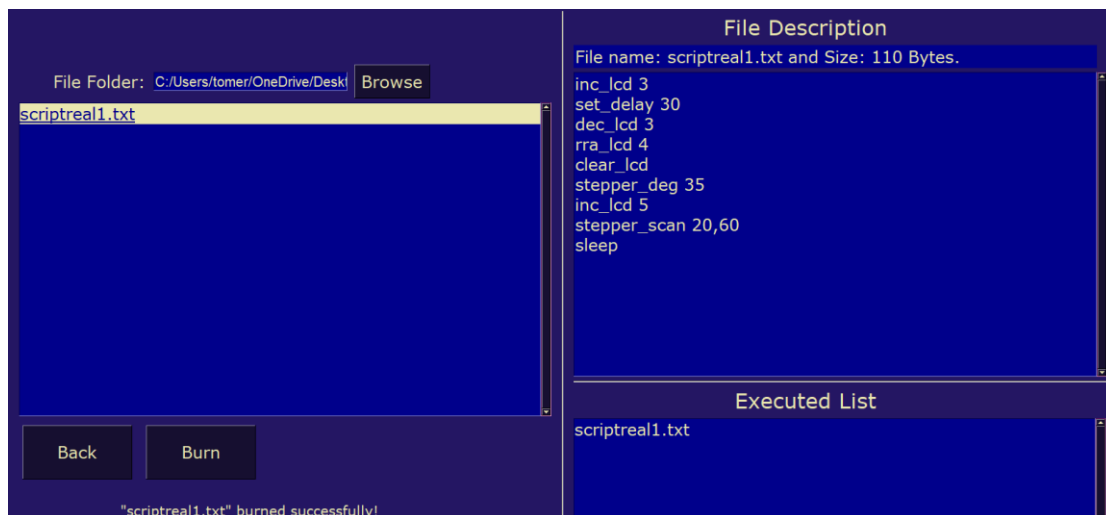
מצב סקריפט

המשימה הרביעית בפרויקט עסקה במתן יכולת לבקר לבצע סדרה של פקודות מוגדרות מראש המאוחסנות בקובצי סקריפט. פקודות הסקריפט מאפשרות לבצע פעולות מורכבות כמו סיבוב המנוע לזוויות מסוימות, הדפסת מידע על גבי מסך המחשב, או כל פעולה אחרת שהוגדרה מראש על ידי המשתמש.

במצב זה, המערכת תומכת בשליחה וקבלה של עד שלושה קבצים, כאשר כל קובץ מכיל סט של פקודות. קבצי הסקריפט נשמרים בזיכרון הפלאש של הבקר, וכל סקריפט ניתן לבחירה ולהפעלה מתוך תפריט המשתמש במחשב. המשתמש יכול לבחור להריץ סקריפט מסוים, ולאחר מכן המערכת תבצע את הפקודות המוגדרות בסקריפט בסדר הנקבע מראש.

במהלך פיתוח המשימה, הוקדשה תשומת לב לניהול זיכרון יעיל בתוך הבקר כדי לאפשר גישה מהירה לנתונים ולאפשר ביצועים מיטביים של הפקודות. התפריט במחשב מציג את הקבצים הקיימים בזיכרון הפלאש של הבקר ואת המידע הרלוונטי לכל קובץ, כמו גודל הקובץ ותכולתו. כאשר המשתמש בוחר להפעיל סקריפט, המערכת טוענת את הקובץ מהזיכרון ומתחילה בביצוע הפקודות המוגדרות בו.

בנוסף, הוספנו פונקציה המאפשרת למחוק קבצים ישנים מהזיכרון או להחליפם בקבצים חדשים, כך שהמערכת תישאר גמישה וניתנת להתאמה לצרכים משתנים של המשתמש. מערכת זו מאפשרת אוטומציה מתקדמת ושימוש קל ונוח למשתמשים שאינם בהכרח טכניים, אך זקוקים למערכת חזקה ומדויקת לביצוע פעולות מורכבות.



חלוקת העבודה בין החומרה לתוכנה:

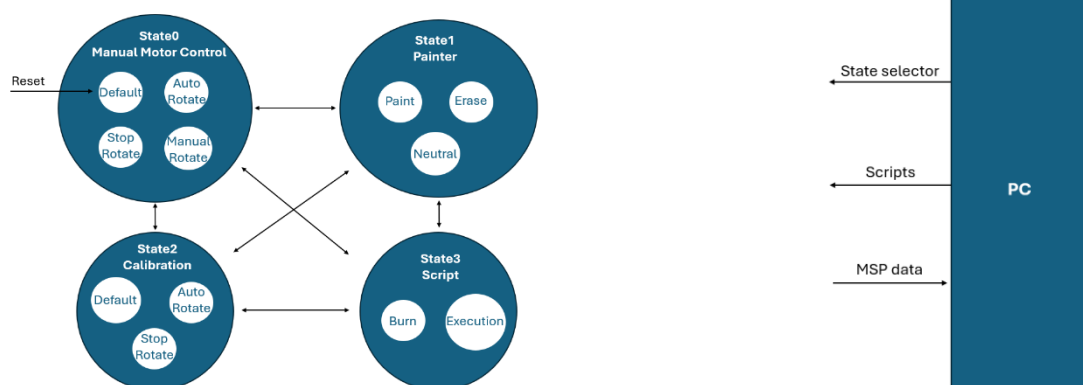
בפרויקט זה, חלוקת העבודה בין החומרה לתוכנה הייתה ברורה ומדויקת. החומרה מטפלת בפעולות הפיזיות ובקלטים האנלוגיים, כמו דגימת המתח מהג'ויסטיק והפעלת מנוע הצעד. התוכנה, מצד שני, מטפלת בעיבוד הנתונים, בתקשורת עם המחשב האישי, ובניהול המערכת כולה. חלוקה זו מאפשרת למערכת לפעול ביעילות תוך ניצול מיטבי של יכולות הבקר ושל החומרה הנלווית.

כך, המערכת משיגה תגובה מהירה ודיוק גבוה, תוך שמירה על גמישות תפעולית המאפשרת שליטה הן דרך ג'ויסטיק והן דרך ממשק גרפי במחשב האישי.

נציג את הדיאגרמת FSM:

צד הבקר:

צד המחשב:



הביצועים בפועל לעומת המפרט הטכני: הסבר מפורט על השינויים ומה גרם להם

במהלך הפרויקט, היה צורך להתמודד עם כמה שינויים והתאמות שקרו כאשר התקדמנו מהמפרט הטכני המקורי לביצועים בפועל של המערכת. להלן סקירה של השינויים שבוצעו, והסיבות שהובילו לכך:

דיוק הסיבוב של מנוע הצעד

- **מפרט טכני:** המפרט המקורי דרש שהמנוע יבצע סיבוב מדויק בזוויות של 0.088° בכל צעד, בהתאם למאפיינים הנומינליים של המנוע.
- **ביצועים בפועל:** במהלך בדיקות הכיול, התברר כי ישנן שונות קטנה בין זווית הצעד הנומינלית לבין הזווית האמיתית שביצע המנוע. הסיבה לכך נעוצה בשונות הפיזית בין מנועים מאותו סוג, דבר המחייב כיול פרטני לכל מנוע.
- **שינוי שבוצע:** הוספנו שלב כיול נוסף, שבמהלכו המערכת מחשבת את זווית הצעד המדויקת עבור כל מנוע באופן פרטני. כיול זה התבצע על ידי מדידת כמות הצעדים הנדרשת לסיבוב שלם (360°) וחישוב זווית הצעד המעשית.

זמן תגובה של המערכת

- **מפרט טכני:** המפרט דרש זמן תגובה מהיר בין הקלט מהג'ויסטיק לבין ביצוע הסיבוב של מנוע הצעד, תחת משטר זמן אמת קשיח (Hard Real Time).
- **ביצועים בפועל:** במהלך הפיתוח, התברר כי ישנם עיכובים קלים הנגרמים כתוצאה מעיבוד הנתונים ומהשהיות שהיו מתוכננות בקוד כדי לאפשר זמן התייצבות מסוים בין כל צעד של המנוע.
- **שינוי שבוצע:** כדי לשפר את זמן התגובה, שינינו את תצורת ההשהיות כך שהן יהיו מינימליות ועדיין יספקו יציבות. נוסף לכך, שיפרנו את האלגוריתם שמחשב את הזווית הרצויה מהג'ויסטיק, כך שיוכל לפעול מהר יותר וללא איבוד דיוק.

שליטה מרחוק באמצעות מחשב אישי

- **מפרט טכני:** המפרט דרש שהשליטה מרחוק דרך המחשב האישי תהיה חלקה, עם ממשק משתמש גרפי (GUI) המאפשר לשלוח פקודות לבקר ולצפות בתגובות בזמן אמת.

- **ביצועים בפועל:** בפועל, במהלך הבדיקות, נתקלנו בבעיות בתקשורת הטורית, אשר גרמו לעיכובים בהעברת הנתונים בין המחשב לבקר, במיוחד כאשר הממשק הגרפי היה עמוס בתהליכים.
- **שינוי שבוצע:** שיפרנו את קוד התקשורת כך שינצל בצורה יעילה יותר את ה-UART ויפחית את העומס על המערכת בזמן העברת הנתונים. בנוסף, ייעלנו את ממשק המשתמש הגרפי כך שיבצע פחות פעולות חישוביות במקביל.

ממשק משתמש גרפי (GUI)

- **מפרט טכני:** המפרט דרש ממשק משתמש גרפי אינטואיטיבי ונוח, שיאפשר למשתמש לשלוט במערכת ולהגדיר פרמטרים בצורה קלה.
- **ביצועים בפועל:** התברר כי הגרסה הראשונית של ה-GUI הכילה עיכובים ומורכבויות מיותרות, אשר פגעו בחוויית המשתמש.
- **שינוי שבוצע:** שיפרנו את עיצוב ה-GUI, פישטנו את התפריטים, והקטנו את מספר השלבים הנדרשים לביצוע פעולות. בנוסף, בוצעו אופטימיזציות בקוד ה-GUI כדי לשפר את מהירות התגובה ואת הזרימה הכללית של העבודה עם הממשק.

מצב Script והרצת פקודות אוטומטיות

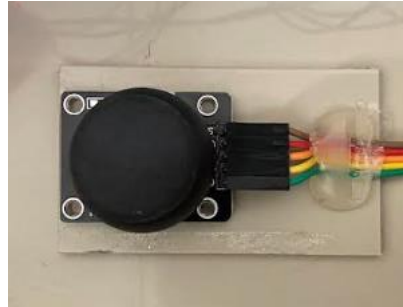
- **מפרט טכני:** המפרט קבע שהמערכת תוכל להריץ פקודות סקריפט מאוחסנות בקובץ, באופן אוטומטי ובמהירות גבוהה.
- **ביצועים בפועל:** בפועל, זיהינו כי זמני ההרצה של פקודות הסקריפט היו ארוכים מהמצופה, בעיקר כאשר הקבצים היו גדולים או מורכבים.
- **שינוי שבוצע:** שיפרנו את מנגנון ניהול הזיכרון של הסקריפטים בבקר, כדי להבטיח גישה מהירה יותר לנתונים ולאפשר ביצוע מהיר של הפקודות. בנוסף, ייעלנו את הדרך שבה הקבצים נטענים ומנוהלים בזיכרון ה-Flash של הבקר.

באמצעות השינויים והאופטימיזציות שבוצעו, הצלחנו להגיע למערכת יציבה ויעילה יותר, העונה על הדרישות המרכזיות של המפרט הטכני, תוך התמודדות עם האתגרים שעלו במהלך הפיתוח.

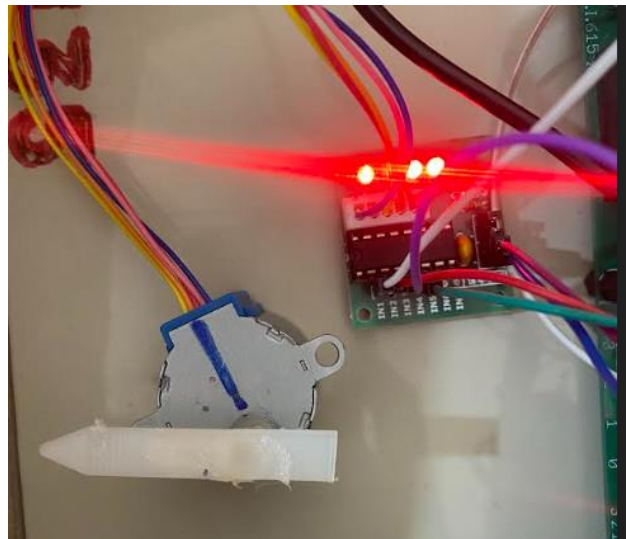
שרטוט מעגל אלקטרוני לביצוע הפרויקט

כדי ליצור שרטוט מעגל אלקטרוני עבור הפרויקט שלך, נצטרך לכלול את כל הרכיבים החשמליים המרכזיים שמוזכרים בפרויקט, כמו:

1. **בקר - MSP430G2553** יחובר לכל רכיבי הקלט והפלט.
2. **ג'ויסטיק אנלוגי** - משמש לשליטה ידנית. כולל חיבורי מתח (V_{cc} , GND) וחיבורי אות (V_{rx} , V_{ry}) המחוברים לכניסות ה-ADC של הבקר.

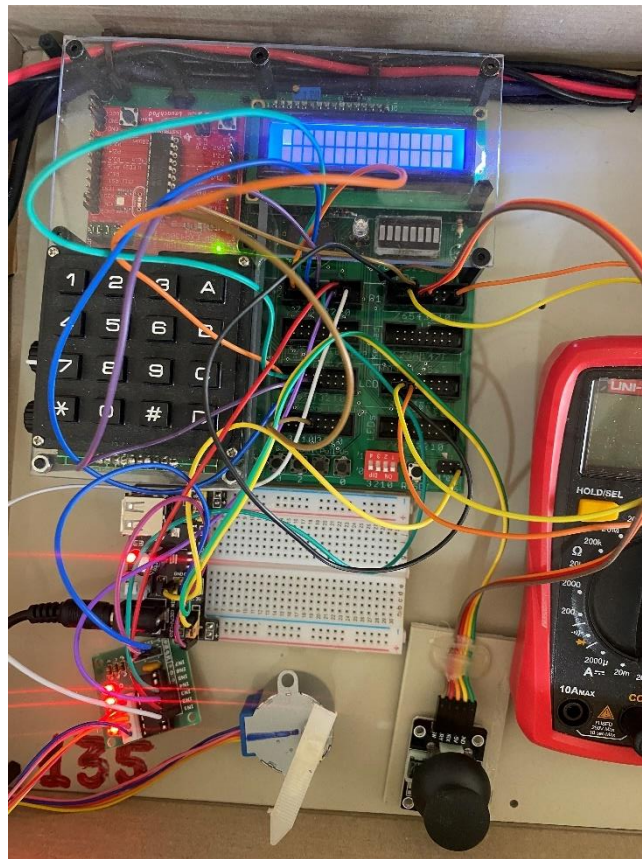


3. **מנוע צעד** - יחובר לבקר דרך דרייבר מנוע כדי להבטיח שניתן לשלוט על המנוע ולספק לו את המתח הדרוש.



4. **דרייבר מנוע** - מתווך בין הבקר למנוע הצעד, מאפשר שליטה על הכיוונים ועל המתח של המנוע.
5. **חיבורים ל - UART** - יחברו את הבקר למחשב אישי, על מנת לאפשר תקשורת טורית בין הבקר למחשב.
6. **מסך LCD** (אם יש צורך בתצוגה) - יחובר לבקר ויופעל דרך GPIO.
7. **מקור מתח 5V** - יישמש להזנת המתח הנדרש לכל הרכיבים.

לכן נצרף תמונה של המעגל לצורך נוחות:



מסקנות:

1. **ביצועים ודיוק:** המערכת שנבנתה מצליחה לספק שליטה מדויקת במנוע הצעד, הן באמצעות ג'ויסטיק והן באמצעות מחשב אישי. התוספת של כיוול מדויק עבור כל מנוע שיפרה את רמת הדיוק של המערכת, והיא מתפקדת היטב תחת תנאי זמן אמת קשיח.
2. **זמן תגובה:** בעקבות האופטימיזציות שבוצעו, המערכת מגיבה בצורה מהירה לפקודות מהג'ויסטיק ומהמחשב. השילוב של טיימרים והשהיות מותאמות הצליח להבטיח זמן תגובה מינימלי תוך שמירה על יציבות המערכת.
3. **שליטה מרחוק וממשק משתמש:** ממשיק המשתמש הגרפי מאפשר שליטה נוחה ומגוון אפשרויות, אך במהלך העבודה התברר כי ישנו עומס מסוים על ערוץ התקשורת UART בעת שידור כמות גדולה של מידע. למרות שיפור שבוצע, ניתן לשפר את התפקוד עוד יותר באמצעות שיפורים נוספים בתקשורת.

4. **מצב: Script** היכולת להריץ פקודות סקריפט מתוך קבצים מאפשרת אוטומציה יעילה של תהליכים מורכבים, אך זמני ההרצה הראשוניים היו ארוכים יותר מהמצופה, ולכן נדרשו אופטימיזציות לניהול הזיכרון והקבצים.

הצעות לשיפורים:

1. **שיפור ממשק המשתמש הגרפי: (GUI)** כדי לשפר עוד יותר את חווית המשתמש, מומלץ לשדרג את ממשק המשתמש הגרפי לממשק מתקדם יותר, המשלב אפשרויות תצוגה נוספות כמו גרפים, מעקב אחר ביצועים בזמן אמת, ואפשרות לבצע שינויים בפרמטרים תוך כדי פעולת המערכת.
2. **אופטימיזציה של תקשורת: UART** כדי להתמודד עם העומס בערוץ התקשורת, ניתן לשקול יישום שיטות דחיסת נתונים לפני השידור או שימוש בפרוטוקולי תקשורת מהירים יותר, כמו I2C או SPI, לצורך שיפור ביצועים והקטנת העומס על הבקר.
3. **שיפור ניהול זיכרון:** ניהול יעיל יותר של זיכרון ה-Flash בבקר יאפשר ביצועים טובים יותר במצב Script, במיוחד כאשר יש צורך בעבודה עם קבצים גדולים יותר. ניתן לשקול גם שדרוגים בתכנון האלגוריתמים לניהול קבצים בזיכרון.
4. **הוספת יכולות מתקדמות למנוע:** בעתיד, ניתן לשקול הוספת חיישנים נוספים, כמו חיישן מיקום או חיישן זרם, שיאפשרו משוב מדויק יותר ובקרה טובה יותר על מנוע הצעד. זה יאפשר להוסיף מצב בקרה סגור (closed loop) במקום הבקרה בחוג פתוח הנוכחית, מה שיגדיל את הדיוק והאמינות של המערכת.
5. **מימוש תקשורת אלחוטית:** ניתן להוסיף אפשרות לשליטה מרחוק באמצעות תקשורת אלחוטית כמו Wi-Fi או Bluetooth, מה שיאפשר גמישות רבה יותר בהפעלת המערכת ממקומות שונים ומחשבים שונים ללא צורך בחיבור פיזי.

המלצות אלו נועדו לשפר את התפקוד הכללי של המערכת, להגדיל את אמינותה, ולשפר את חוויית המשתמש והגמישות שלה בסביבות עבודה שונות.