

אוניברסיטה בן-גוריון

בית הספר להנדסת חשמל ומחשבים

דו"ח מסכם לפרויקט גמר קורס "מבנה מחשבים ספרתיים" 1-361-4191

פרויקט מסכם

Control system of motor-based
machine

תכנון ומימוש מערכת בקרה למכונה
מבוססת מנוע צעד בשליטה ידנית ומרחוק

מגשים:

- אוריאל מנצור

- בר רזנס וויג

מדריך:

27/08/2022

מטרת הפרויקט:

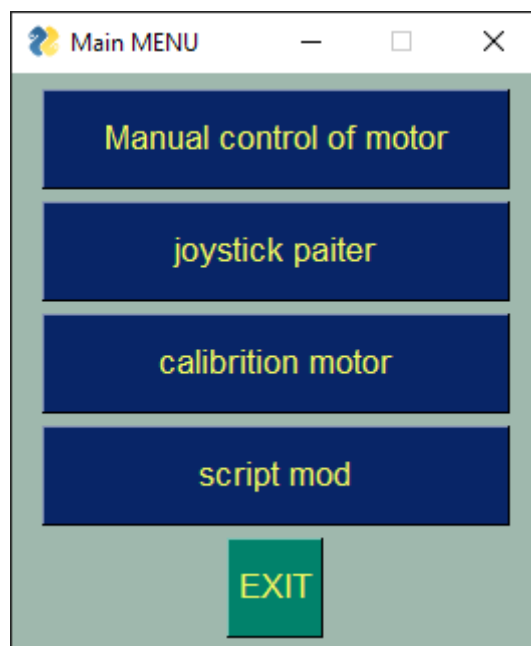
1. תכנון ומימוש מערכת בקרה למכונה מבוססת מנוע צעד בשליטה ידנית ע"י joystick analog ובשליטה מרחוק ממחשב אישי דרך ערוץ תקשורת טורית. הדגשים בתכנון המערכת הם של רמת ביצועים גבוהה תחת משטר של Hard real time עם רמת דיוק גבוהה.
2. במסגרת הפרויקט יפותח קוד בשפת C++/C למימוש מערכת זמן אמת מבוססת פסיקות (צד MCU) להפעלת הרכיבים וקריאת המידע ממד המרחק.
3. מחשב PC ישמש לצורך ממשק בעזרת PySimpleGUI למשתמש ולתצוגה לכל פעולה המוגדרת במערכת ודורשת תצוגה וממשק למשתמש. ה-MCU יחובר למחשב ה-PC באמצעות תקשורת טורית אסינכרונית בסטנדרט RS-232.
4. ממשק GUI למשתמש בצד ה-PC יאפשר קביעת פרמטרים, שליחת קבצים ופקודות high-level ל-MCU ותצוגת תמונת הרדאר ב-PC. הממשק בצד ה-PC נכתב בשפה עילית Python ויתמוך במימוש של תקשורת טורית בין הבקר ל-PC.
5. הממשק יאפשר העברת קבצים הכוללים פקודות high-level מקודדות למימוש בצד הבקר. כל הקבצים בצד בקר יישמרו בזיכרון RAM בלבד.

תיאור הפרויקט ותיאור כללי של ביצועי החומרה והתוכנה:

הפרויקט מאפשר לנו כמה סוגי פעולות:

- שליטה ידנית על המנוע ע"י הג'ויסטיק – Manual control of motor based machine
- ציור ע"ג המסך ה-PC בעזרת הג'ויסטיק – Joystick based PC painter
- מצב כיול המנוע – Stepper Motor Calibration
- מצב סקריפט - Script Mode

כאשר בלחיצה על כל אחד צד המחשב-מודיע לבקר לאיזה מצב עברנו ופותח את התפריטים המתאימים במחשב ובצד הבקר-הבקר מקבל את ההודעה שעכשיו הוא צריך לשנות state ולהיות מוכן לשליחה או קבלה של סוג מידע שונה.

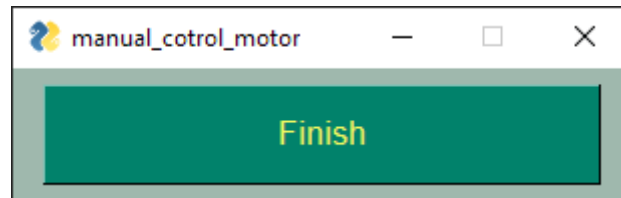


כל הפעולות הפרויקט הזה הינן ללא שימוש בקלט מהבקר אלא רק קלטים מהחשב ופלטים בצד הבקר חוץ מהג'ויסטיק

שליטה ידנית על המנוע ע"י הג'ויסטיק:

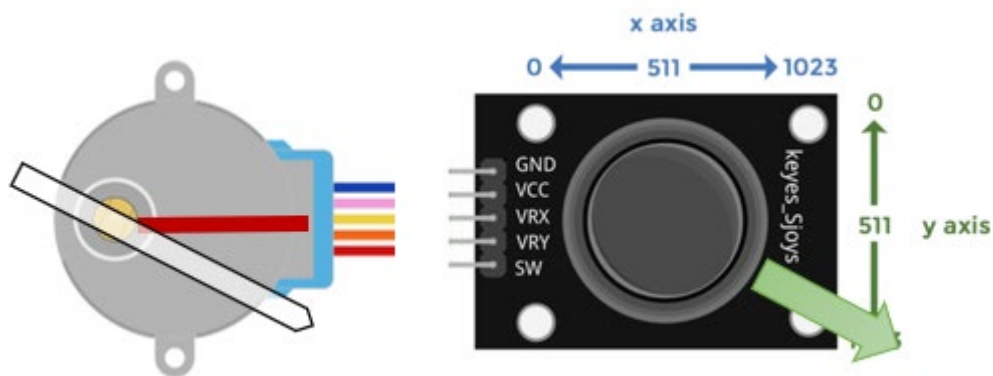
בצד ה-PC

בכניסתנו למצב הזה יפיע לנו החלון הזה:



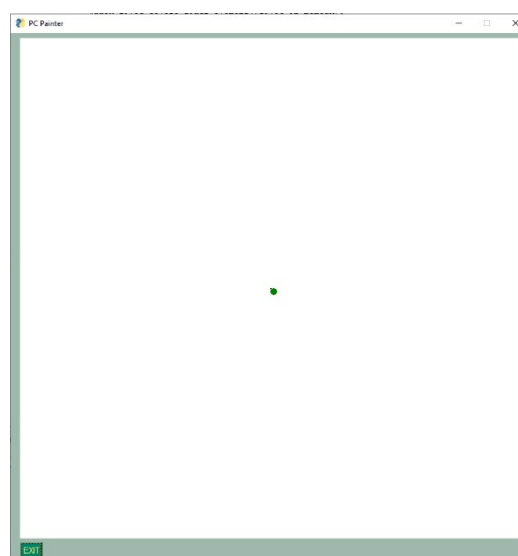
כאשר הכפתור במחשב משמש רק להפסקת תזוזת המנוע.

בזמן שהחלון עוד פתוח נוכל להזיז את הג'ויסטיק לכל מיני כיוונים ומיד מנוע הצעד יתחיל לזוז לכיוון שאליו מצביע הג'ויסטיק. לדוגמה באיור הבא כאשר נזיז את הג'ויסטיק לכיוון החץ הירוק המנוע יגיע לנק' הזאת



ציור ע"ג המסך ה-PC בעזרת הג'ויסטיק

לאחר לחיצה על מצב צייר יפתח לנו החלון הבא:



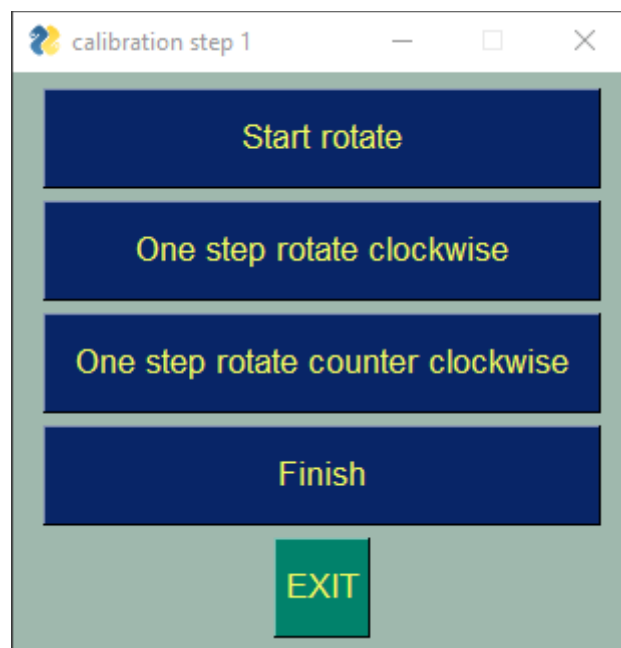
כאשר נוכל להזיז את המצביע (הנקודה הירוקה) ע"י הזזת הג'ויסטיק. במצב ברירת המחדל הסמן שלנו יצייר על גבי הקנבס וכאשר נלחץ על PB2 (במקרה שלנו לא השתמשנו בלחצן הג'ויסטיק כי הוא עבד) נעבור בין מצבי הציור שהם {מחיקה} ← {הזזת סמן} ← {ציור} וחוזר חלילה.

את שינוי הג'ויסטיק אנחנו מקבלים מהבקר בעזרת דגימות של מודול ה-ADC שדוגם לנו פעם את ציר ה-X ופעם את ציר ה-Y ו-DMA מכניס את הדגימות ישירות מה-ADC למערך וכשאר הוא מסיים מס' מסוים של דגימות הוא עושה ממוצע של ה-X וה-Y ושולח אותן משורשרות עם האם נלחץ כפתור PB2.

המחשב אז ממיר את הקורדינאטות לזווית ורדיוס כדי לקבל כיוונית של תזוזה, ומזיז את הסמן/מחק/מכחול.

מצב כיוול המנוע

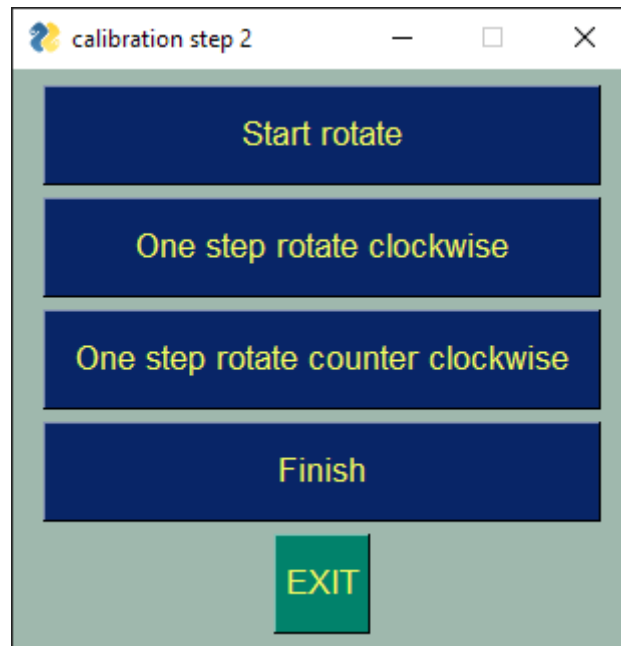
במצב כיוול המנוע יפתח לנו החלון הבא:



מטרת הכיוול היא לכוון לבקר שכאשר המצביע הלבן נמצא על הפס האדום הוא (המנוע) נמצא על זווית 0, ובנוסף לקבל אינדיקציה על כמה צעדים יש בסיבוב שלם ומכך יוצא מהי הזווית הנומינלית, ז"א כמה מעלות המנוע זז בכל צעד.

Step 1 - נרצה להביא קודם את המנוע שהמצביע הלבן יהיה על הפס האדום, נוכל להשתמש בכפתורים שלעיל:

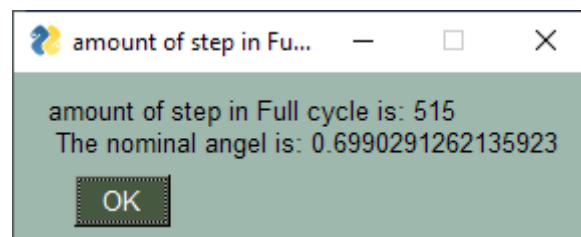
- Start rotate - המנוע יתחיל סיבוב בכיוון השעון ובאותו זמן יפתח חלון פופ-אפ שיאפשר עזירה של המנוע
- One step rotate CW - צעד אחד בכיוון השעון (לטובת כיוון עדין)
- One step rotate CCW - צעד אחד נגד כיוון השעון (לטובת כיוון עדין)
- Finish - כאשר הגענו למצב שבו אנחנו מצביעים על הפס האדום



Step 2- אותו חלון יישאר לנו ועכשיו המטרה היא לעשות סיבוב שלם כדי לספור בדיוק כמה צעדים יש בסיבוב שלם ובכך למצוא את הזווית הנומינלית.

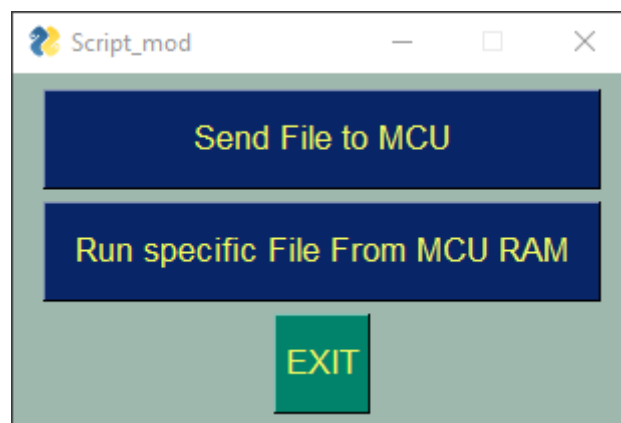
- Start rotate- המנוע יתחיל סיבוב בכיוון השעון ובאותו זמן יפתח חלון פופ-אפ שיאפשר עצירה של המנוע
- One step rotate CW- צעד אחד בכיוון השעון (לטובת כיוון עדין)
- One step rotate CCW- צעד אחד נגד כיוון השעון (לטובת כיוון עדין)
- Finish- כאשר הגענו למצב שבו אנחנו מצביעים על הפס האדום

כאשר נסיים: יופיע לנו הפופ-אפ הבא:

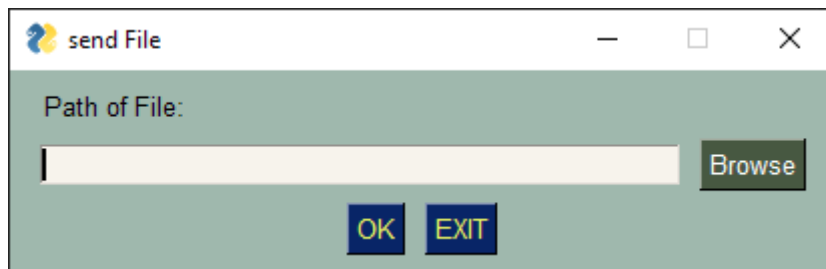


מצב סקריפט

במצב הזה אנחנו יכולים להטעין קבצים אל הבקר ולהריץ אותם:

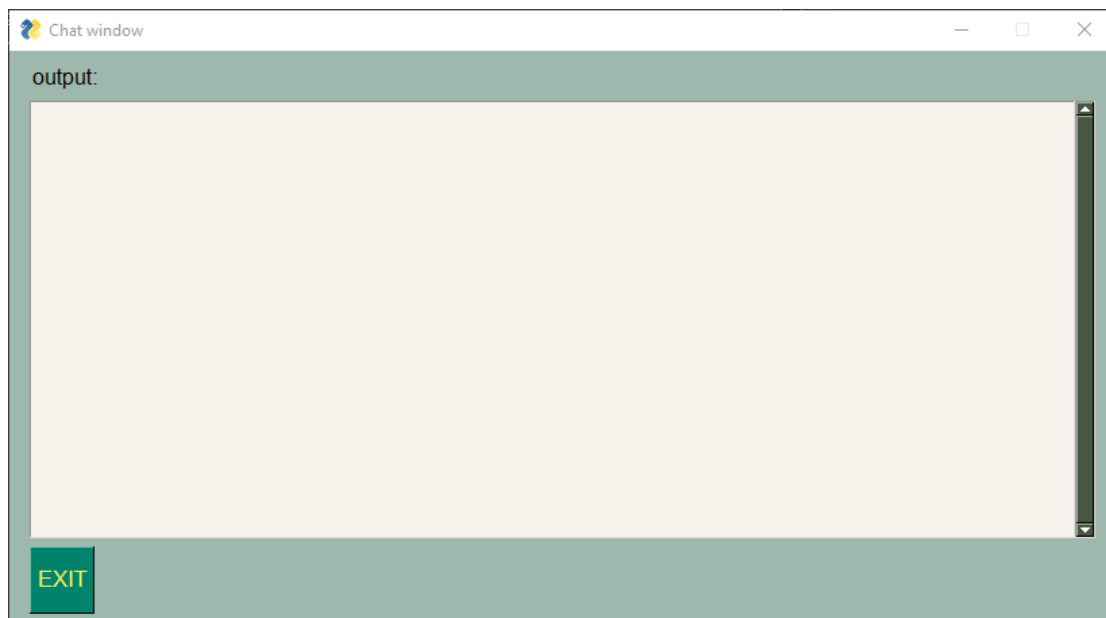


כאשר נלחץ על שליחת קובץ יפתח לנו החלון:



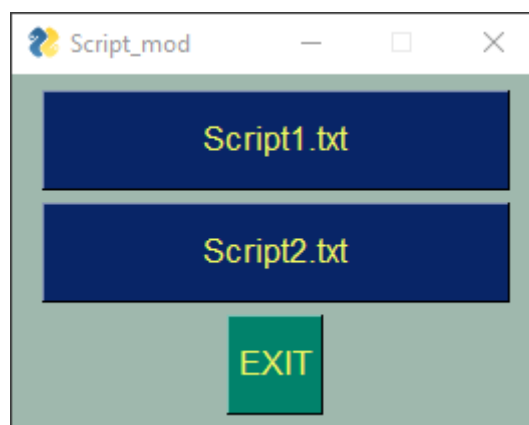
הקבצים שנשלחים לבקר מתקבלים על ידו בעזרת DMA ומאוחסנים בשיטת circular buffer שמתחיל מכתובת 0x20000000 גודל הבאפר הוא 8KB משיקולים הנדסיים שנגרמים עקב כך שכל הזיכרון של הבקר הוא 16KB

בסיומו יקפוצ מיד עם הצלחת השליחה פופ-אפ שהשליחה הצליחה ואז כאשר נלחץ OK קובץ הפקודות יחל לרוץ על הבקר. ויפתח החלון הבא שבו הבקר ישלח הודעות עבור פקודות מסויימות כמו להודיע לאיזו זווית הוא הגיע. לא נוכל ללחוץ EXIT עד שהקובץ לא יסיים לרוץ.



לאחר לחיצה על EXIT נחזור למצב של או שליחת קבצים או הפעלת קבצים מהזיכרון

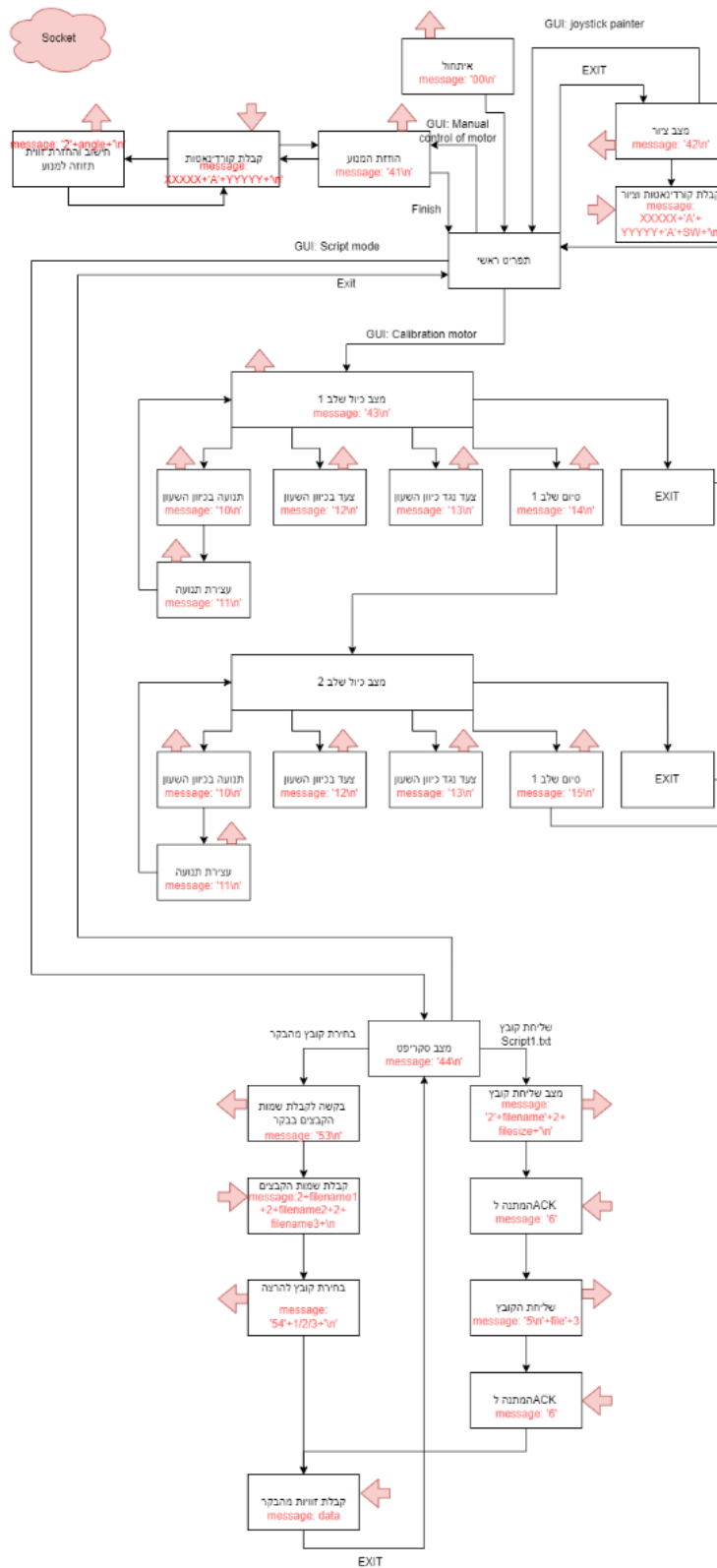
יופיע החלון הזה:



כאשר כאן מופיעים שני קבצים אך לא יכולים להיות יותר מ-3 קבצים וגודלם הכולל לא יכול לעלות על 8KB מפאת שיקולים הנדסיים שלא מאפשרים. ומדיניות הוצאת הקבצים אם מנסים להכניס יותר מגודל הבאפר או יותר מ-3 קבצים היא FIFO.

בלחיצה על אחד הקבצים מיד יפתח חלון ה"צ'אט" ויופיעו בזמנם הOUTPUT מהבקר.

צד ה־PC



מקרא:

- שליחה או קבלה מהבקר - →
- מעבר בין מצבי התוכנה - →

הביצועים בפועל לעומת המפרט הטכני, הסבר מפורט מה השתנה ומדוע

- בהנחיות נתבקשנו לחבר את הג'ויסטיק אל הדרייבר החומרה החשמלי (ספק כוח) אך מכיוון שתוך כדי עבודה נוכחנו לדעת שהוא משנה את רמות המתחים ברגליים כתוצאה משימוש נוסף של המנוע גם בדרייבר החומרת (ראה שאלה בפורום + שתי תשובות שענינו לעצמנו) העברנו את חיבור הכוח של הג'ויסטיק ישירות אל הבקר ביציאות המתח שלו
- בלחצן הג'ויסטיק לאחר בדיקות בעזרת הרב מודד גילינו שלחצן SW מקוצר באופן תמידי לכן לאחר דיווח לראש המעבדה הוחלט להשתמש בPB2 במקום בלחצן הג'ויסטיק

שרטוט מעגל אלקטרוני לביצוע פרויקט

חיבורי הפרויקט הם כדלקמן:

כפתורים:

$PB2 \rightarrow P1.2 \rightarrow A5$

לד:

$R_{red_{LED}} \rightarrow P2.3 \rightarrow C2$; $G_{green_{LED}} \rightarrow P2.2 \rightarrow C1$; $B_{blue_{LED}} \rightarrow P2.1 \rightarrow C0$

פאזות מנוע:

$I_{n1} \rightarrow C11$; $I_{n2} \rightarrow C10$; $I_{n3} \rightarrow C6$; $I_{n4} \rightarrow C5$

ג'ויסטיק:

$V_{Rx} \rightarrow E20$; $V_{Ry} \rightarrow E22$

מסקנות והצעות לשיפורים

לא הבנו לעומק את משמעות הכיול של מודול הADC ולא לדגום משתי רגליים אם בכלל הדבר אפשרי.