בן כהן, 208685784
אורי אברהמי, 315416057

דו"ח מסכם – מעבדה מתקדמת למערכות אוטונומיות

מבוא

במטלה זו, נדרשנו לתכנן מסלול לרכב Arduino אוטונומי, כך שבחלקו יעקוב אחר מסלול (Line Follower) ובחלקו האחר יבצע מסלול כלשהו (Costume Path) עד להגעה לנקודה ממנה יחזור לעקוב אחר מסלול. על מנת לממש את הפרויקט, נעזרנו בסמינרים שהיוו מבוא לפרויקט ובפרט בעזרת הקודים לLineFollower ולPath_Control.
לדו"ח זה נלווים שני נספחים:

1. נספח A המצרף את המסלול הנדרש לפרויקט ואת הקישור לסרטון של תנועת הרכב (בונוס).
2. נספח B המתאר את סיכומם של הסמינרים שהועברו במעבדה.

מהלך הניסוי

1. ראשית, כפי שהוצע בקובץ הפרויקט, התמקדנו בקובץ הLineFollower שנמצא בנתיב: File->Examples->Zumo32u4->LineFollower. כמו כן, התאמנו את ערכי הבקר (Kp=2000, Ki=25) לערכים שבהם הרכב שלנו התנהג בצורה 'נומינלית' יחסית על המסלול כך שהוא לא יסטה מהר מדי או לחילופין ייצא מן המסלול בגלל שגיאות עקיבה.
כמו כן, התאמנו את שורת הקוד הבאה שתטפל בהפרשי המהירויות:
speedDifference = error / 2 + 1 * (error - lastError)
בה ראינו כי הרכב מתנהג בצורה נומינלית במעקב אחר הקו.
2. שנית, רצינו לבחון כיצד נוכל לגרום לרכב לעצור ב'צומת הT'. באמצעות הסמינרים, בדקנו את הסנסורים של הבקר וזיהינו כי אם הערך **גבוה** מ500 – אזי הקו השחור ניכר ואם הינו **קטן** מ300 אזי אינו. כמו כן:
   a. LineSensorValues[1], LineSensorValues[2], LineSensorValues[3] – מעידים על עצמים שנמצאים מקדימה לרכב (במקרה שלנו אם הקו השחור מופיע ממול לרכב).
   b. LineSensorValues[0], LineSensorValues[4] מעידים על עצמים שנמצאים מימין ומשמאל לרכב בהתאמה (במקרה שלנו אם הקו השחור מופיע מצידי לרכב).

על כן, התאמנו את שורות הקוד כך שהגדרנו 4 תנאים:

```
bool condition_active_forward = (lineSensorValues[1]>500 || lineSensorValues[2]>500 || lineSensorValues[3]>500);
bool condition_not_active_forward = (lineSensorValues[1]<300 && lineSensorValues[2]<300 && lineSensorValues[3]<300);
bool condition_not_active_left_right = (lineSensorValues[0]<300||lineSensorValues[4]<300);
bool condition_active_left_right = (lineSensorValues[0]>500||lineSensorValues[4]>500);
```

תמונה 1: הגדרת תנאי הסנסורים מתוך הקוד

ובתוך הLoop של הקובץ, הגדרנו לרכב שעבור כל אחד מהתנאים הוא צריך לממש את המסלול שלו באופן שונה:

a. במידה ויש לו מסלול מקדימה ולא מן הצדדים (או ההפך), אזי שהוא צריך לבצע LineFollower
b. במידה ויש לו 'צומת T' אזי שהוא צריך לבצע Path_Control

3. לאחר מכן, בעזרת סמינר Path_Control יצרנו מסלול ריבועי לרכב בCostume_Path. זאת, נעשה תוך כדי מדידות של המרחקים וכן תוך התאמות של הרצת הקוד ובדיקה על מיקומו הסופי של הרכב. הנקודות שניתנו לו הינן:

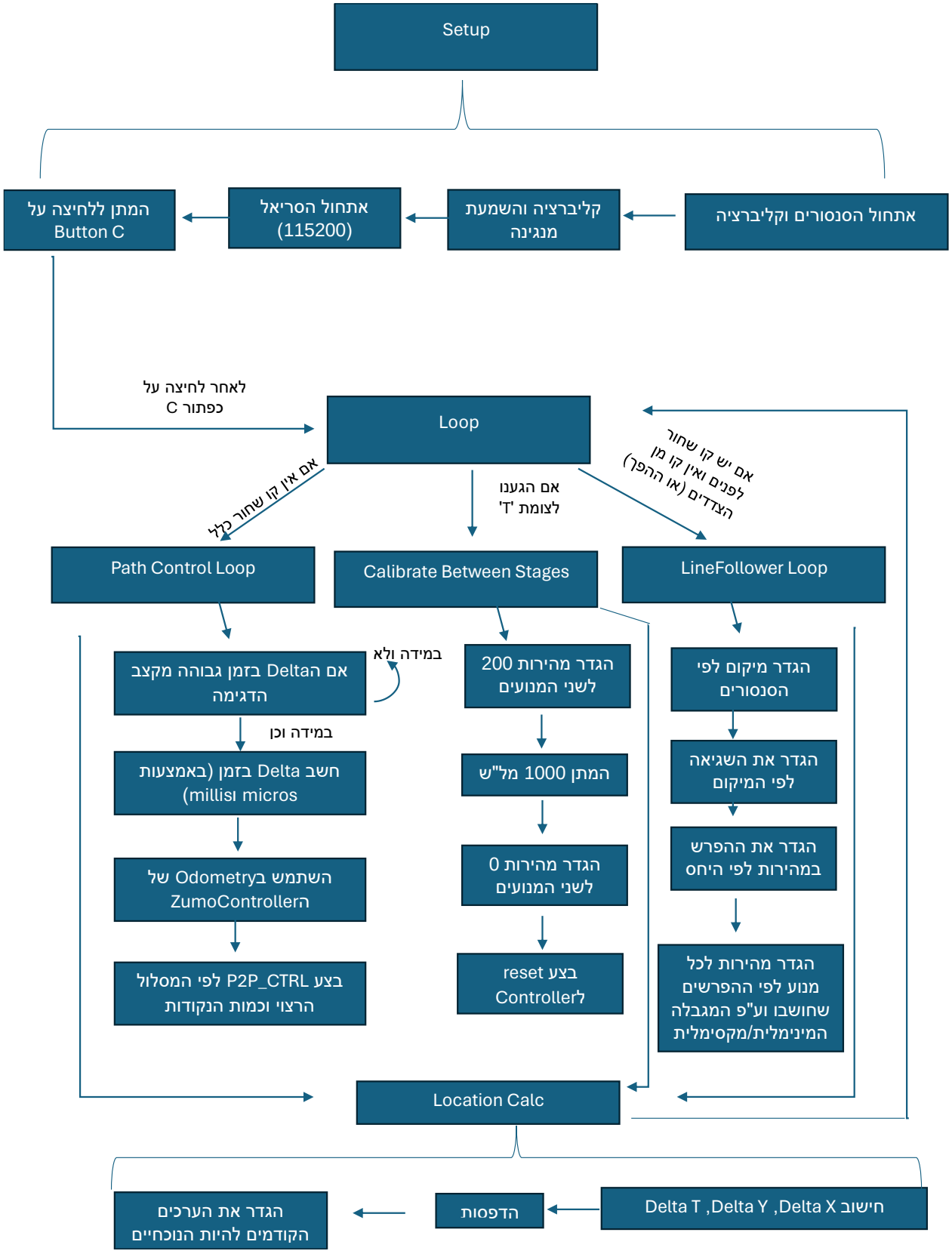
```
{0.1, 0.1}, {0.3, 0.1} , {0.3,1.1} , {-0.4,1.18}
```

תמונה 2: מיקומי הנקודות לPath_Control

4. התאמנו את שני הקודים ביחד בתוך הSetup והLoop והרצנו כדי לבדוק שילוב וריצה 'משותפת'.
5. אספנו פלטים כדרוש אשר יופיעו בחלק **תוצאות הניסוי**.

בן כהן, 208685784
אורי אברהמי, 315416057

Block Diagram



תמונה 3 : דיאגרמת בלוקים של קוד הפרויקט

בן כהן, 208685784
אורי אברהמי, 315416057

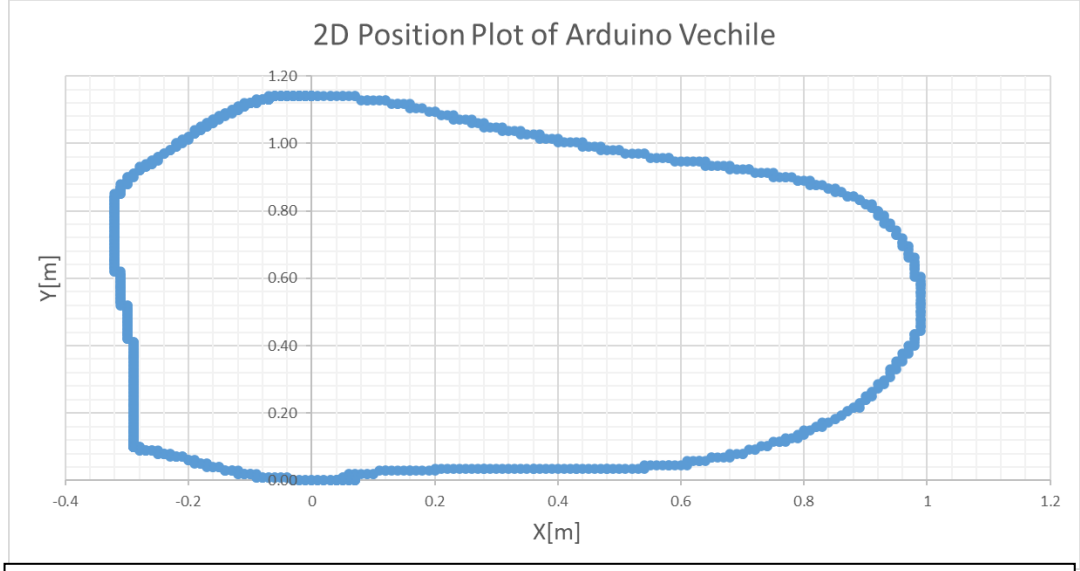
תוצאות הניסוי

בחלק זה של הניסוי, אספנו את הפלטים הדרושים לדו"ח המסכם. אלו הם:

1. גרף X מול Y של מיקום הרכב
2. גרף V מול T של הרכב
3. עבור LineFollower
   - Error מול T
4. עבור Path_Control
   - Error מול T
   - מסלול רצוי מול מסלול מצוי ($r\ Vs\ r_d$)

**נציין שעל מנת לקבל מדידות והיסט נכון, ביצענו את סט המדידות כך שהרכב הורץ מנקודה First המופיעה בנספח A.

1. גרף X מול Y של מיקום הרכב



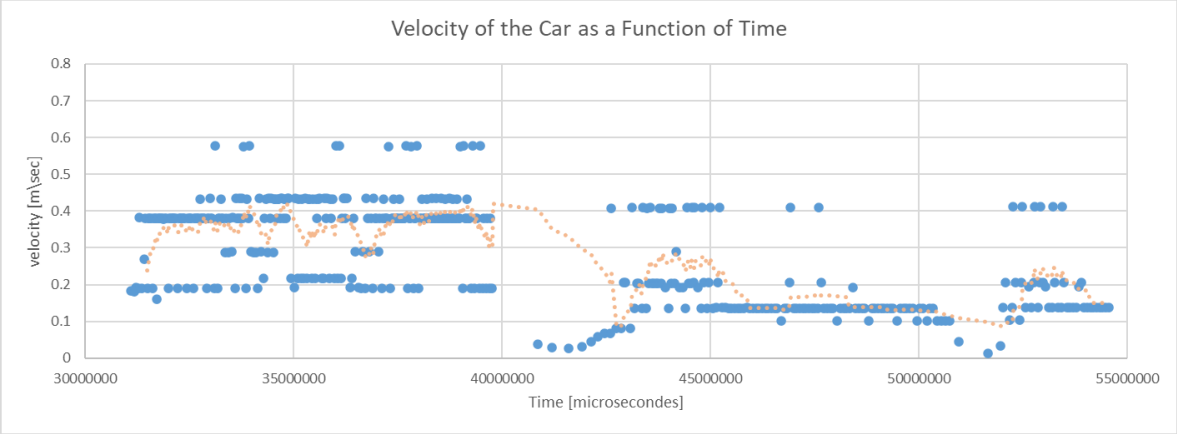
גרף 1: מיקום הרכב בדו-מימד, ציר Y למול X

<u>ניתוח תוצאה:</u>

נוכל לראות שקיבלנו מסלול התואם את הרצוי, שכן צורת הLineFollower תואמת את המסלול שהוגדר (כפי שניתן לראות בנספח A), וכן הCostume_Path גם כן 'סוגר' את הצורה כפי שרצינו בחלק הPath_Control.

בן כהן, 208685784
אורי אברהמי, 315416057

2. גרף V מול T של הרכב

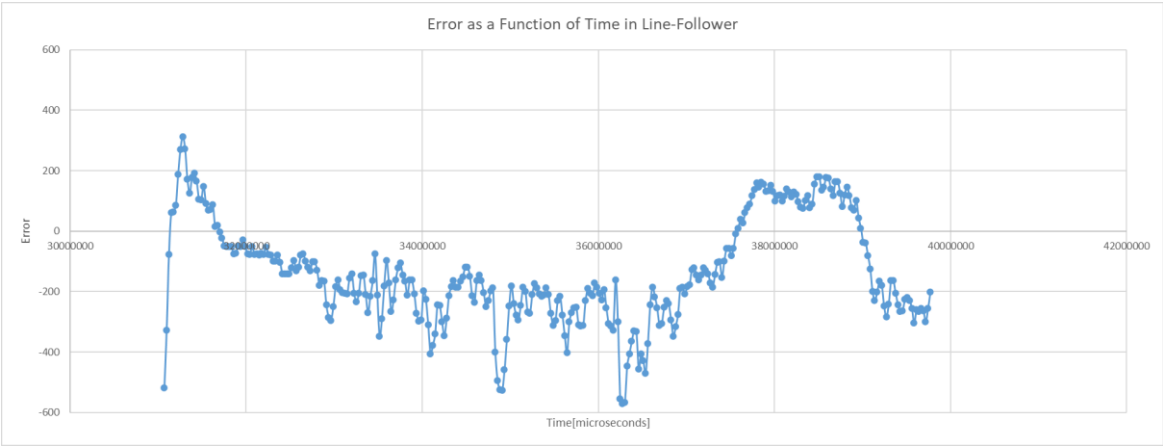


גרף 2 : מהירות הרכב כפונקציה של הזמן

<u>ניתוח תוצאה:</u>

ניתן לראות שבגרף זה קיבלנו כי מהירות הרכב נעה בין ערכים של $0.1\frac{m}{s} \rightarrow 0.4\frac{m}{s}$ גם בחלק הLineFollower וגם בחלק הPath_Control שאלו הם ערכים נומינליים וסבירים. כמו כן, הוספנו (בכתום) מיצוע של הערכים לטובת ויזואליזציה של המידע והדגשתו. נציין כי המיצוע התבצע על סט של 10 ערכים בכל פעם.

3. Error – LineFollower מול T



גרף 3 : השגיאה כפונקציה של הזמן במסלול LineFollower

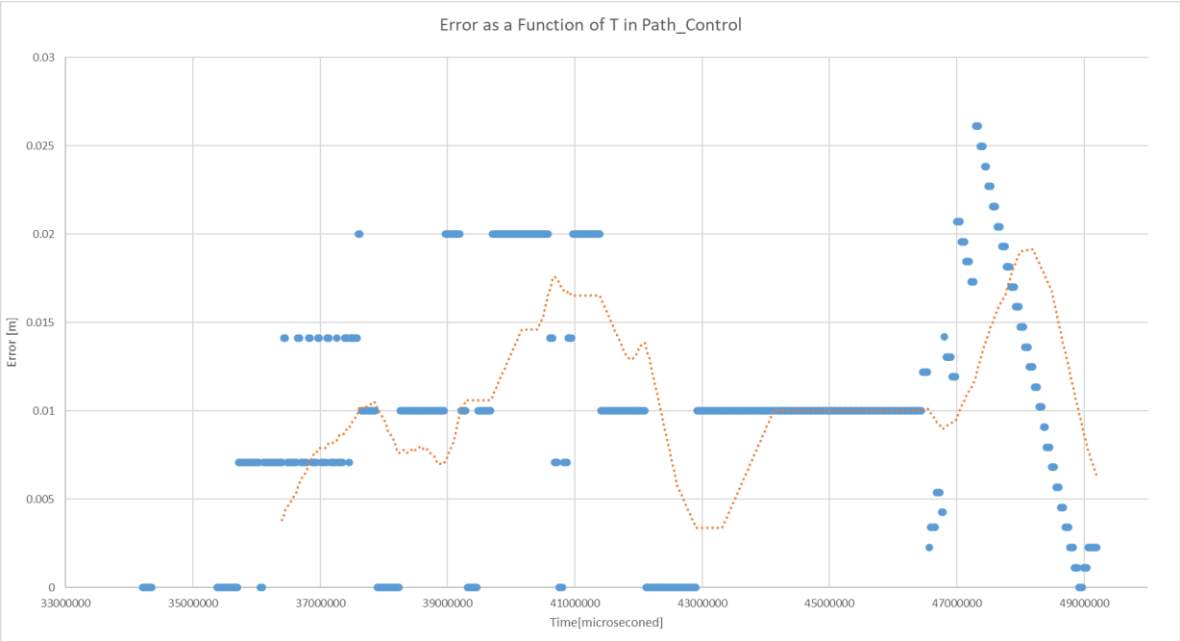
<u>ניתוח התוצאה:</u>

ראשית, נציין כי **לא** הוגדרו ערכים לציר הY (השגיאה) שכן היא נמדדה באופן 'עצמאי' בקוד.
כמו כן, קיבלנו ערכים בטווח ± 500 שאלו הם ערכים נומינליים וסבירים.

בן כהן, 208685784
אורי אברהמי, 315416057

4. Path_Control

a. Error מול T

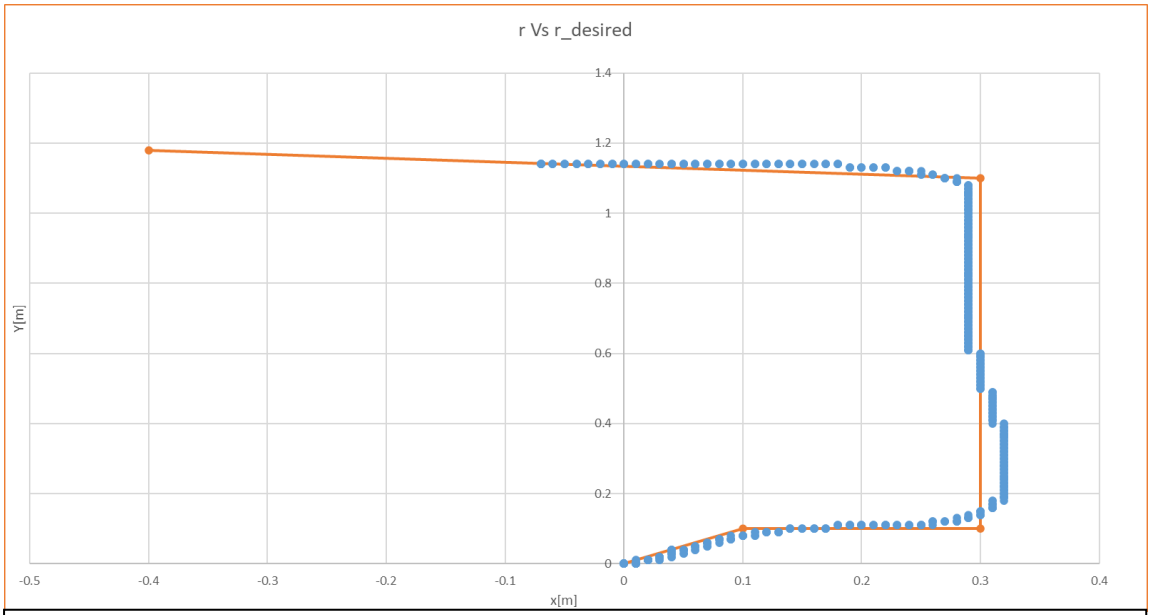


גרף 4 : השגיאה כפונקציה של הזמן במסלול Path_Control

<u>ניתוח התוצאה :</u>

בסעיף זה, חישבנו את המרחק בין המיקום המדוד (X,Y) לבין המיקום הרצוי (אשר יופיע בפלט הבא) בכך שחושבו משוואות הישר של כל אחד מן החלקים במסלול ובעזרת משוואת 'מרחק נקודה מישר' חישבנו את השגיאה.

b. מיקום רצוי מול מיקום מצוי $r\ Vs\ r_d$



גרף 5 : מיקום רצוי מול מיקום מצוי של הרכב במסלול הPath_Control

בן כהן, 208685784
אורי אברהמי, 315416057

ניתוח תוצאה:

ניתן לראות שבמסלול Path_Control המיקום הרצוי והמיקום המצוי הינם קרובים וכן כפי שציפינו הרכב עקב אחר הקואורדינטות שניתנו לו כדרוש. כמו כן, נציין שבחלק האחרון 'הרצוי' בו ניתן לראות רק מסלול 'רצוי' ללא עקיבה 'מצויה', הרכב חזר לעקוב בLineFollower, שכן זיהה את המסלול לפי הנדרש בקוד.

דיון

אתגרים ודרכי פתרון

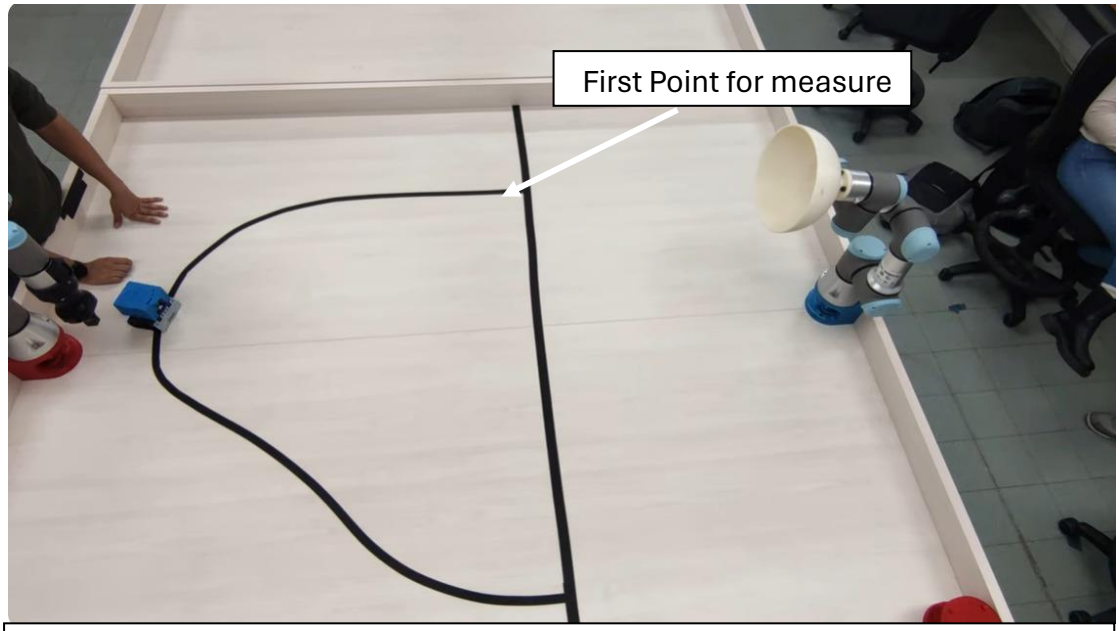
1. תכנון המעבר בין שני מצבי הפעולה של הרכב, היווה אתגר משמעותי בעבודה במעבדה. לצורך התמודדות עימו, עבדנו עם שני קטעי קוד המגדירים את חלק ה"loop", כך שמעבר ביניהם יתרחש כאשר יהיה שינוי משמעותי בערכים שנקלטים בסנסורים בחלקו הקדמי של הרכב (כפי שהוצג בפרק 'מהלך הניסוי').
2. הדגרדציה ברכבים האוטונומיים הובילה לכך שלעתים נתקלנו במחסור בסוללה בחלק מהרכבים או אף בתקלה פיזית כמו פער במהירות הסיבוב של שני הזחלים ברכב. לאור זאת, לעתים השתמשנו ברכב אחר ואף ביצענו 'ריסוט' של הרכב או נעזרנו באחראי המעבדה לטובת תיקון מכאני.
3. אתגר נוסף שהתמודדנו עמו היה בחלק ניתוח התוצאות בו ההיסטים מ'ראשית הצירים' היו מוזחים לנקודת התחלת ההרצה של הרכב. בשל כך, ביצענו את תחילת המדידות (כפי שצוין בפרק 'תוצאות הניסוי') מהנקודה First המופיעה בנספח A, כך שההיסט ממנה אידיאלי.

מסקנות

1. המעבר בין שני מצבי הפעולה של הרכב בוצע כדרוש, שכן ניתן לראות בחלק 'תוצאות הניסוי' ובסרטון המצורף, מעבר 'יחסית חלק' בין שני מצבי הפעולה.
2. בתוצאות הניסוי קיבלנו התאמה טובה גם לערכי השגיאה, גם לערכי המיקום וגם בהפרש בין המיקום הרצוי-למצוי. מכאן נסיק שנעשתה התאמה טובה למקדמי הבקר.
3. הבחירה במסלול בעל אופי ליניארי בחלק הPath_Control סייעה בניתוח הנתונים שבוצע בחלק תוצאות הניסוי.
4. עבדנו לפי 'הצעדים המוצעים לפתרון' כפי שהוצגו בהנחיות הפרויקט ואכן הייתה זו הדרך הנוחה לפתרון הבעיה.

בן כהן, 208685784
אורי אברהמי, 315416057

Appendix A – המסלול הנדרש לפרויקט והקישור לסרטון של תנועת הרכב (בונוס).



תמונה 4 : מסלול הפרויקט

<u>קישור לסרטון תנועת הרכב (בונוס) :</u>

https://youtu.be/CbfT-P__2xY?feature=shared

בן כהן, 208685784
אורי אברהמי, 315416057

Appendix B – רקע תיאורטי וסיכום של הסמינרים

<u>סמינר 1 – הקדמה למעבדה והיכרות עם סביבת העבודה.</u>

בסמינר זה, למדנו את העקרונות המתקדמים בתחום הבקרה האוטונומית תוך עבודה מעשית ותרגול במעבדה. תחילה הכרנו את סביבת העבודה, מבנה הרכב עליו בוצעו הניסויים במעבדה ורכיביו המשמעותיים. לאחר מכן, נלמדו מספר שיטות להתחברות למחשב הרכב המרוחק, לשם הרצת הקוד עליו. לאורך הסמינר התנסנו בכתיבת קוד והרצה שלו על לוח ה ZUMO שעל הרכב באמצעות שפת התכנות Arduino. בין התרגילים שבוצעו, שליטה על נורות ה"LED", שימוש במצלמה וניווט הרכב באמצעות Joystick. לבסוף תורגל השימוש בכלי ההדפסה, Serial/Monitor Plotter.

<u>סמינר 2 – Embedded.</u>

במפגש זה, למדנו על מערכות משולבות, המבנה הבסיסי שלהן, ורכיביהם השונים. ערכנו היכרות עם פרוטוקולי התקשורת Serial וI2C ולמדנו על רכיבי הבקרה, החישה וההנעה של הרכב. לאורך הסמינר, תרגלנו הפעלה של רכיבי הרכב השונים. תחילה בוצעה כתיבת קוד לשם השמעת נגינה. לאחר מכן, ביצענו מספר תרגילים להיכרות עם הסנסורים המותקנים על הרכב, ועם מנגנון ההנעה שלו.

<u>סמינר 3 – Path Control.</u>

בסמינר זה, הכרנו את אופן חישוב המיקום והמרחק של רכב המעבדה ונעזרנו בו ובידע שצברנו מהסמינרים הקודמים, כדי להביא את הרכב לנסוע במסלול מוגדר מראש, תוך שימוש בסימולצית בקר PID לשם העקיבה. לאחר ההיכרות עם הגורמים השונים, יצרנו קוד והתאמנו את נתוני הסימולציה כך שרכב המעבדה יסע באופן עצמאי במסלולים המוגדרים. העבודה בסמינר זה, היוותה בסיס משמעותי לעבודה על פרויקט הגמר.