

הפקולטה להנדסה

המחלקה להנדסת חשמל ואלקטרוניקה

# קדם פרויקט

תכנון ומימוש מערכת רמזורים

שם הסטודנט:	דוד לנציאנו	רון חלבי
מס' ת.ז.:	204344758	201303914

שם המנחה: מר דמיטרי בורודין

גב' אורבך יפית

עבודה זו מוגשת במסגרת מעבדת קדם פרויקט במחלקת הנדסת חשמל ואלקטרוניקה.

הערכה:

## תוכן עניינים

3	<b>פרק 1: מבוא</b>
3	1.1 רקע תיאורטי
4	1.2 מדוע יש להפחית בעומסי התנועה ולמזער תאונות?
4	1.3 הצורך בפתרון
5	<b>פרק 2: מטרת הפרויקט</b>
6	2.1 דרישות המערכת
6	2.2 תכנון וביצוע
7	<b>פרק 3: תיאור המערכת</b>
7	3.1 דיאגרמת הבלוקים
9	3.2 הסבר כללי על המערכת
10	<b>פרק 4: מימוש המערכת</b>
10	4.1 שער
10	4.2 מחלק תדר
12	4.3 מערכת בקרה
12	4.3.1 שלבי תכנון של מערכת בקרה
13	טבלת מצבים:
14	רישום טבלת מעברים וציאות:
16	4.3.2 זמן
16	4.4 לחצן תיעדוף
16	4.4.1 D-latch (NAND Gates)
17	4.5 מערכת ההשהיה
17	4.5.1 חד יציב MC14538B
18	4.6 מערכת תצוגה
18	4.6.1 מונה בינארי
18	4.6.2 קוד עשרוני BCD
18	4.6.3 Seven segment
19	
19	4.6.4 חישוב מערכת הגנה לדיודות
20	<b>פרק 5 : תוצאות ומסקנות</b>
20	5.1 תוצאות הפעלת המערכת
24	5.2 מימוש והפעלת המערכת עבור בקר Arduino
26	5.3 בעיות ופתרונות
27	5.4 מסקנות הפרויקט
27	6. סיכום
28	7. מקורות

## פרק 1: מבוא

### 1.1 רקע תיאורטי

רמזורים הם מערכת אורות צבעוניים המוצבת בצמתים, ותפקידם לווסת את התנועה בצומת. הרמזור הוא סוג של תמרור.

רמזורים לנסיעה למכוניות - לרמזורים למכוניות ישנם שלושה צבעים המסודרים אחד מעל השני (ירוק, צהוב ואדום). האור הירוק מסמל שהנוסע רשאי לחצות את הצומת. האור הצהוב מסמל שהרמזור עומד להתחלף לאדום ולכן יש לפנות את הצומת והאור האדום מסמל לנוסע לעצור את נסיעתו ולא לעבור את הצומת.

רמזורים להולכי רגל - רמזורים להולכי רגל עשויים להיות במצב המתיר חצייה, בזמן שגם מכוניות מסוימות מורשות לעבור. במצבים כאלו, זכות הקדימה נתונה להולכי הרגל. האור הירוק - מסמל שאפשר לחצות בעוד האור האדום - מסמל שאין לחצות.



איור 1. רמזורים לנסיעה למכוניות.



איור 2. רמזורים להולכי רגל<sup>1</sup>.

## 1.2 מדוע יש להפחית בעומסי התנועה ולמזער תאונות?

פקקי תנועה בישראל גורמים נזק של עשרות מיליארדי שקלים בשנה. בשנת 2016 העריכו משרד האוצר, משרד התחבורה וגופי התכנון כי פקקי תנועה עולים למשק 35 מיליארד ש"ח בשנה, וכי ללא שינוי המצב הקיים, העלות תגיע ל-70 מיליארד ש"ח בשנה עד שנת 2040. כמות הנסועה בכלי רכב ממונעים (במיליוני ק"מ בשנה) לעומת אורך הכבישים בישראל (בק"מ). עם השנים יש פער גדל בין כמות הנסועה לבין כמות הכבישים. הפער גדול יותר כאשר בודקים את היחס בין כמות הנסועה במכוניות לעומת כמות הכבישים העירוניים שבהם עיקר עומסי התנועה.

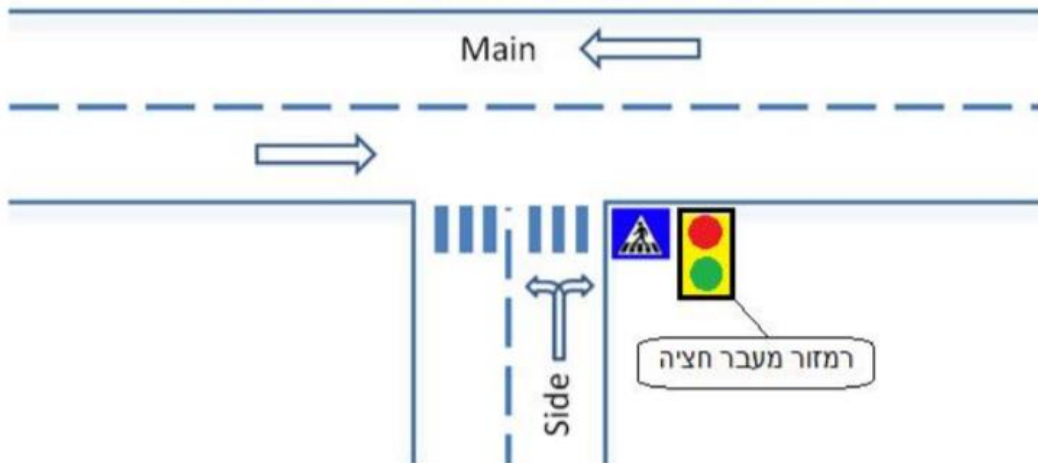
עומסי תנועה הן אחת ההשפעות החיצוניות היקרות והנפוצות של הרכב הפרטי, בישראל נהגים מבליים בפקקים מיליוני שעות עבודה ופנאי בשנה, דבר שפוגע הן ברווחה החברתית באופן ישיר, הן בהון החברתי והן בתוצר הכלכלי של המשק, בעלות של מיליארדי שקלים בשנה. בנוסף עומסי התנועה גורמים לאיחורים לא צפויים, מעלים את זיהום אוויר מתחבורה במיוחד במרכז הערים, גורמים לבלאי מואץ במכוניות ולבזבז דלק, ותורמים לבעיות בריאות כמו מתח נפשי וגופני (סטריס), כאבי גב וצוואר, התנהגות פרועה של נהגים עצבניים (Road rage, נהגים שאיחרו ומנסים להגיע בזמן), ולתת פעילות גופנית. באופן זה לפקקים יש השפעות גבוהות על רווחה חברתית בישראל והם מחריפים את משבר הבריאות בישראל על ידי תוספת של תחלואה, נכות ותמותה<sup>ii</sup>.

## 1.3 הצורך בפתרון

הדרך שבה יהיה ניתן להפחית מעומסי התנועה הינה על ידי תכנון מערכת רמזורים אשר תהיה יציבה ובטיחותית כלפי המכוניות והולכי הרגל ואף שימושית לאנשים בעלי לקות ראייה. הרמזור מהווה את אחד האמצעים החשובים לשפור זרימת התנועה ולקיום משטר תנועה תקין ובטוח.

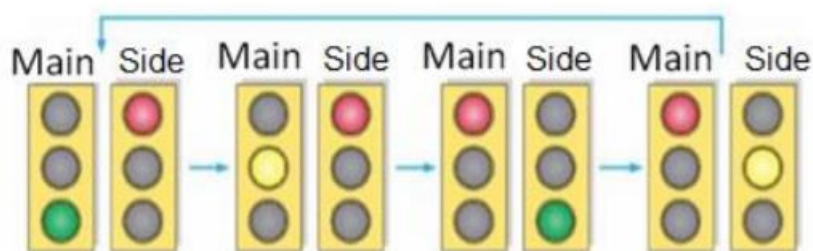
## פרק 2: מטרת הפרויקט

מטרת הפרויקט הינה תכנון ומימוש של מערכת רמזורים לצומת T המתוארת באיור הבא:



איור 3. צומת הרמזורים<sup>iii</sup>.

הצומת מורכבת משני צירים - ציר ראשי וציר משני הכולל מעבר חציה להולכי רגל, כאשר הצירים כוללים שלושה רמזורים בהתאמה, העובדים במצבים המתוארים באיור 4 (רמזור הולכי הרגל מתואם עם הרמזור המשני).



איור 4. מצבי הרמזורים הראשי והמשני.

המערכת כוללת תצוגה של ספירה לאחור עבור הולכי רגל וצפצוף בתדר משתנה לפי מצב הרמזור.

## 2.1 דרישות המערכת

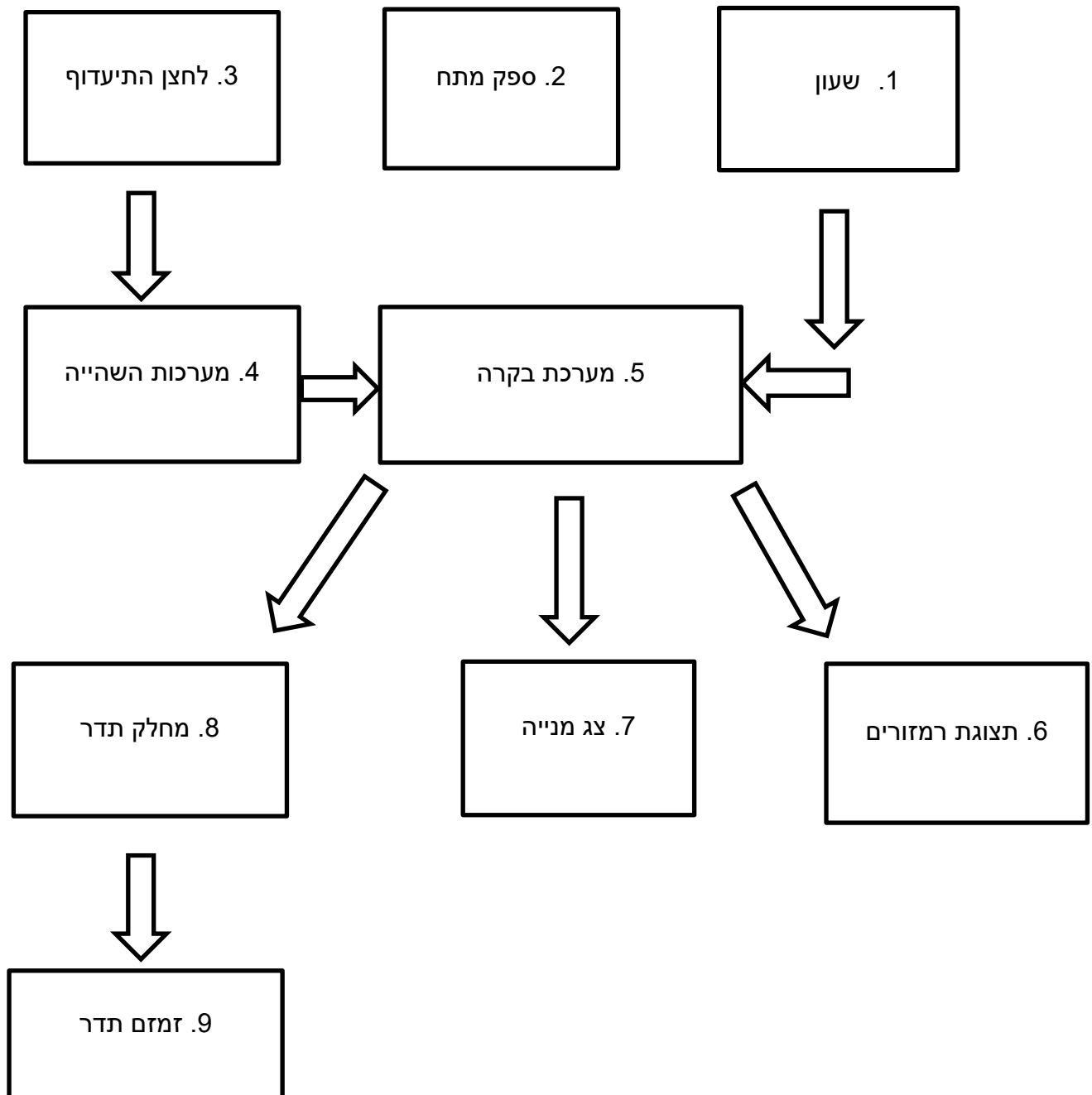
- המערכת עוברת בין מצבים שונים בעלי אורך זמן משתנה.
- המערכת בעלת צפצוף בשני תדרים שונים במעבר הולכי רגל כדי לשמש אנשים בעלי לקות ראייה.
- צג זמן אשר מראה על ידי ספירה לאחור כמה זמן נותר להולך רגל לעבור את מעבר החצייה.
- המערכת צריכה להיות סינכרונית.
- מותקן במערכת לחצן תיעדוף להולכי רגל כאשר מוסיף להולך רגל 2 שניות נוספות לחצייה בטוחה.
- המערכת תפעיל את הלחצן רק בעת תחילת מחזור מצבים חדש.
- המערכת מאפשרת עצירה בטוחה של רכבים בעת המעבר בין המצבים.

## 2.2 תכנון וביצוע

- תכנון ובניית תצוגת המונה.
- תכנון ובניית שעון.
- תכנון ובניית מערכת בקרה סינכרונית.
- תכנון ובניית מחלק תדר.
- תכנון ובניית זמן.
- תכנון ובניית לחצן תיעדוף.
- תכנון ובניית תאורת הרמזורים.

פרק 3: תיאור המערכת

3.1 דיאגרמת הבלוקים



**1. שעון**

אחראי על סנכרון המערכת בעזרת שעון בודד.

**2. ספק מתח**

מספק מתחי הזנה של 5 [volt] לכלל הרכיבים והשערים של המערכת.

**3. לחצן תיעדוף**

נועד לתת אינדיקציה למערכת ההשהייה מתי עליה לפעול.

**4. מערכת השהייה**

ברגע שהיא מקבלת אינדיקציה מהלחצן תיעדוף ובתחילת המחזור הבא היא מבצעת השהייה של כשתי שניות למערכת.

**5. מערכת בקרה**

למערכת זו ישנם מספר תפקידים:  
הפעלת תצוגת הרמזורים.  
הפעלת התזמונים השונים בין המצבים.  
סנכרון בין המצבים השונים של המערכת.  
הפעלת מערכת ההשהייה רק ברגע תחילת מחזור חדש של המערכת ואיפוסה.

**6. תצוגת רמזורים**

משמשת לתצוגת תאורה בצבעים שונים המתארים את מצבו של כל רמזור במערכת.

**7. צג מנייה**

משמש לתצוגת כמה זמן ישנו להולך רגל לחצות את מעבר החצייה בספירה לאחור.

**8. מחלק תדר**

משמש לחלוקת התדרים בכניסה לזמזם.

**9. זמזם תדר**

משמש את ההולכי רגל בעלי לקות ראייה בכך שהוא מזמזם בשני תדרים שונים המתארים את שני המצבים השונים של רמזור ה הולכי רגל.



## 3.2 הסבר כללי על המערכת

במצב התחלתי הרמזור הראשי נמצא במצב ירוק ומאפשר זרימה תנועה בכביש הראשי, במקביל בכביש המשני הרמזור נמצא במצב אדום ולא מאפשר פניה ימינה או שמאלה של מכוניות, ובנוסף בזמן זה רמזור הולכי רגל מאפשר חציית הכביש תוך כדי השמעת צפצוף בתדר גבוה עבור הולכי רגל לקוי ראייה .

לאחר פרק זמן של 6 שניות עובר הרמזור הראשי למצב ביניים (אור צהוב) למשך 3 שניות כדי לאפשר עצירה בטוחה של הרכבים , בזמן זה הרמזור המשני עדיין באדום ורמזור הולכי הרגל עובר לאדום והצפצוף עובר לתדר נמוך למשך כל פרק הזמן בו רמזור הולכי הרגל במצב אדום.

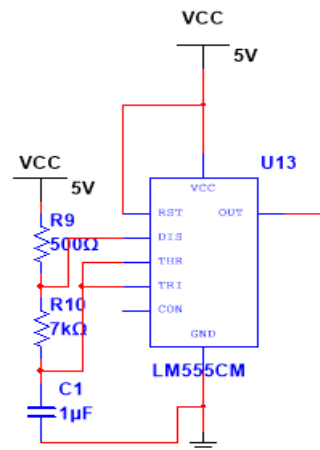
לאחר 3 שניות הרמזור הראשי עובר לאדום והמשני לירוק, רמזור הולכי הרגל נותר ללא שינוי למשך 8 שניות. לאחר פרק זמן של 6 שניות עובר הרמזור המשני למצב ביניים למשך 3 שניות כדי לאפשר עצירה בטוחה של הרכבים , בעוד הרמזור הראשי עדיין באדום .

לאחר 3 שניות הרמזור המשני עובר לאדום והראשי לירוק ורמזור הולכי הרגל עובר לירוק עם צפצוף בתדר גבוה למשך כל פרק הזמן בו רמזור הולכי הרגל במצב ירוק . וחוזר חלילה . בנוסף ברמזור הולכי הרגל, ממוקם לחצן תיעדוף, ברגע שהולך רגל לוחץ על הלחצן, בפעם הבאה שהרמזור הראשי הופך למצב ירוק, הוא ימשך 8 שניות, כאשר ברמזור הולכי הרגל נשמע צפצוף בתדר גבוה למשך 8 שניות ומוצג זמן בתצוגה.

## פרק 4: מימוש המערכת

### 4.1 שעות

אחראי על סנכרון המערכת בעזרת שעות בודד בנוי מרכיב Lm555cm. רכיב זה מכיל שני מגברי שרת המחוברים בתצורה של מגבר משווה, נגדים טרנזיסטורים ודלגלג. צורת החיבור של השעות מאפשרת duty cycle של 50% באופן כזה שנגד R2 זניח ביחס לנגד R1 על מנת שהזמן טעינה יהיה שווה לזמן הפריקה. המעגל מוזן ממתח של 5 [V] כיוון שזהו מתח העבודה של הדלגלג בסימולציה. השעות שלנו צריך לעבוד בשניות אך, ישנה בעייתיות עם הסימולציה ולכן זמן המחזור של השעות בפועל הוא 10m [Sec].

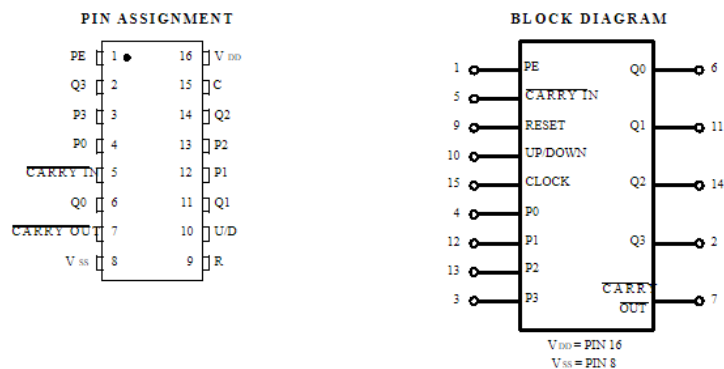


איור 5. שעות מבוסס Lm555cm

### 4.2 מחלק תדר

השימוש במחלק תדר הינו לטובת הזמזם של המערכת, אחת הדרישות של המערכת הייתה יעילות ללקויי ראייה על ידי צפצוף בשני תדרים שונים. באמצעות מחלק התדר, התדר חולק ב 4 על ידי מונה בינארי. בתורת המיתוג, מונה הוא התקן אלקטרוני שסופר את מספר האירועים מסוג מסוים (לרוב אותות חשמליים) שקרו, בדרך כלל בהתאמה לאותות שעות. מונים בנויים ממעגלי זיכרון כמו דלגלגים (פליפ פלופים) ובעזרת שיטות תיכון מגוונות. מונה סינכרוני, כמו שמתחייב משמו, כל אותות המוצא (הפליטים) של הפליפ פלופים משתנים על פי שעות יחיד. מונה זה יכול למנות עד  $2^{n-1}$  כאשר n הוא מספר הביטים במונה. במונה זה, הביטים מתעדכנים בעזרת מערכת של שערים לוגיים. דרך פשוטה ליישום (באמצעות השערים) הלוגיקה עבור כל ביט במונה סינכרוני עולה, היא לשנות את ערכו של כל ביט כאשר ערכם של כל הביטים הנמוכים ממנו הוא אחד. לדוגמה במונה בן ארבעה ביטים שנסמנם כביט 0, ביט 1, ביט 2 וביט 3, ביט 1 משתנה כאשר ערכו של ביט 0 הוא אחד, ביט 2 משתנה כאשר הערכים של ביט 0 וביט 1 הם שניהם אחד וביט 3 משתנה כאשר הערכים גם של ביט 0, גם של ביט 1 וגם של ביט 2 הם אחד.

## MC14516B



איור 6. דיאגרמת בלוק של רכיב MC14516B

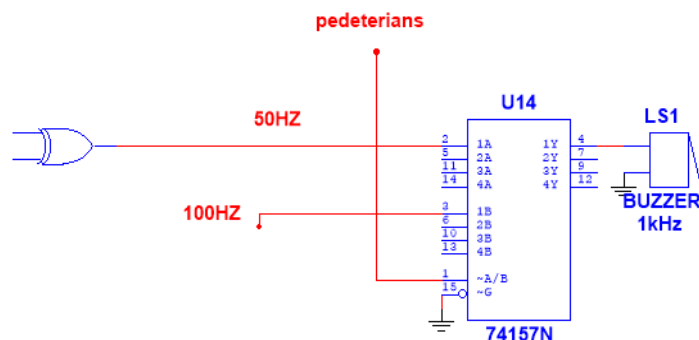
**TRUTH TABLE**

Carry In	Up/Down	Preset Enable	Reset	Clock	Action
1	X	0	0	X	No Count
0	1	0	0		Count Up
0	0	0	0		Count Down
X	X	1	0	X	Preset
X	X	X	1	X	Reset

X = Don't Care

איור 7. טבלת אמת של רכיב MC14516B

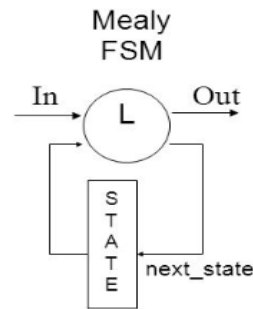
הערה- על מנת לממש את המערכת בצורה הכי אופטימלית ולחסוך ברכיבים, בוצע שימוש במונה בינארי של הטיימר על ידי חיבור שער XOR בין היציאות Q0 – Q1.



איור 8. מחלק תדר מבוסס MC14516B

### 4.3 מערכת בקרה

הדרישה למערכת בקרה הייתה על מנת לתזמן את הרמזורים לפי הצירוף המבוקש. מערכת הבקרה שלנו הינה מערכת עקיבה סינכרונית מסוג mealy – שבה המוצאים הינם פונקציות של המצב הנוכחי והכניסות הנוכחיות. כלומר, מערכת צירופית עם זיכרון. ערכי הפלט תלויים בערכי הקלט וכן בתוכן הזיכרון של המערכת.



איור 9. מערכת עקיבה צירופית מסוג mealy

#### 4.3.1 שלבי תכנון של מערכת בקרה

1. הבנת הדרישות
2. בניית דיאגרמת מצבים או טבלת מצבים
3. צמצום טבלת המצבים במידת הצורך
4. בחירת הקצאת המצבים ובחירת flip flop
5. טבלת מעברים ומוצא
6. מימוש פונקציות עירור ופונקציות מוצא
7. שרטוט מימוש המערכת

טבלת מצבים:

מצבים	רמזור ראשי	רמזור משני	רמזור הולכי רגל
התחלתי – 6m [Sec]	ירוק	אדום	ירוק + צפצוף בתדר גבוה
שני – 3m [Sec]	צהוב	אדום	אדום + צפצוף בתדר נמוך
שלישי – 6m [Sec]	אדום	ירוק	אדום + צפצוף בתדר נמוך
רביעי – 3m [Sec]	אדום	צהוב	אדום + צפצוף בתדר נמוך

איור 10. טבלת מצבים של מערכת הבקרה

בנוסף יש לחצן להולכי רגל ברגע שלוחצים עליו, בפעם הבאה שהרמזור הראשי הופך לירוק, הוא יימשך ל 8 שניות, כאשר ברמזור הולכי רגל יישמע צפצוף בתדר גבוהה למשך 8 השניות.

בחירת הקצאת מצבים – דלגלג מסוג D:

במערכת זו ישנם ארבעה מצבים, כדי לממש זאת יש צורך בשני דלגלגים לאור העובדה שדלגלג מקבל ביט אחד המייצג שני מצבים ( $2^n$  - מספר הביטים)

מצבים		יציאות							
A	B	Rm	Ym	Gm	Rs	Ys	Gs	Rz	Gz
0	0	0	0	1	1	0	0	0	1
0	1	0	1	0	1	0	0	1	0
1	0	1	0	0	0	0	1	1	0
1	1	1	0	0	0	1	0	1	0

איור 11. טבלת יציאות המערכת כתלות במצבים השונים

במעגל שלנו ישנם שתי כניסות, לחצן תיעדוף timeri .  
כאשר מטרת השעון הינה לתזמן את מצבי המערכת, כולל את ההשהיות והמעבר בין המצבים.

רישום טבלת מעברים ויציאות:

	NS	
	TIMER	
p.s	0	1
00	00	01
01	01	10
10	10	11
11	11	00

		TIMER	
		0	1
y1	y2	Y1	Y2
0	0	00	01
0	1	01	10
1	0	10	11
1	1	11	00

Present state = y1 and y2, Next State = Y1 and Y2

Y2:

y1 y2	T	
	0	1
00	0	1
01	1	0
11	1	0
10	0	1

$$Y2 = y2t' + y2't = y2 \text{ xor } t$$

Y1:

y1 y2	T	
	0	1
00	0	0
01	0	1
11	1	0
10	1	1

$$Y1 = y1'y2t + y1t' + y1y2'$$

Green main:

y1	y2	
	0	1
0	1	0
1	0	0

$$y1'y2'$$

Yellow main:

y1	y2	
	0	1
0	0	1
1	0	0

$$y1'y2$$

Red main:

y1	y2	
	0	1
0	0	0
1	1	1

$$y1$$

Green side:

y1	y2	
	0	1
0	0	0
1	1	0

$$y1y2'$$

yellow side:

	y2	
y1	0	1
0	0	0
1	0	1

y1y2

Red side:

	y2	
y1	0	1
0	1	1
1	0	0

y1'

Green pedestrians:

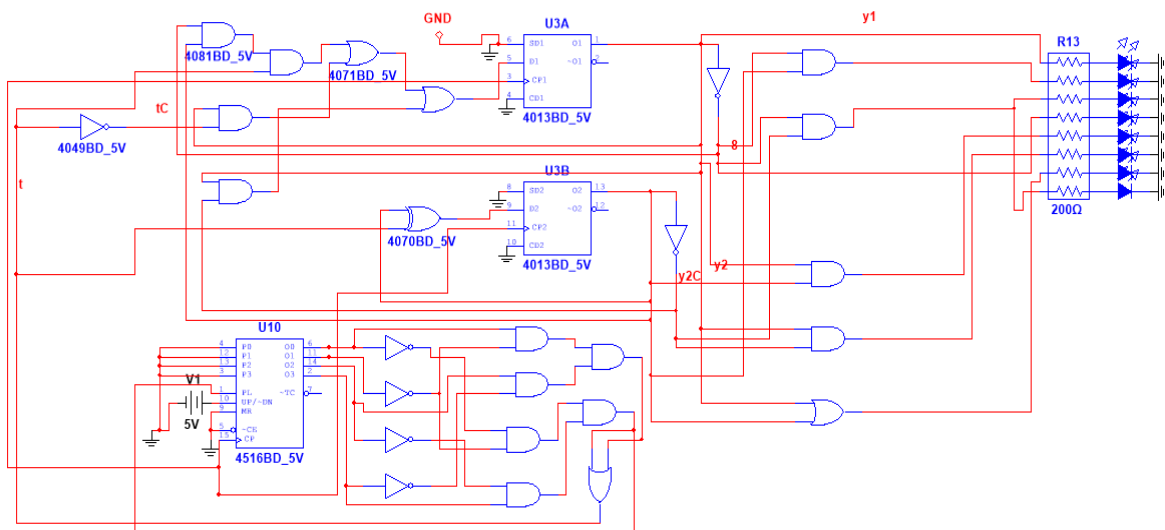
	y2	
y1	0	1
0	1	0
1	0	0

y1'y2'

Red pedestrians:

	y2	
y1	0	1
0	0	1
1	1	1

y1+y



איור 12. מערכת בקרה בשלמותה

#### 4.3.2 זמן

כניסת timer (זמן) ממומשת על ידי פונקציות בוליאניות בעלות חלוקת זמנים שונה, המשתנים של הפונקציות הינם היציאות של מונה בינארי המוזן מהשעון הכללי של המערכת. כאשר המונה הינו מונה מעלה. הפונקציות מתוארות להלן:

$$t6 = Q3'Q2Q1'Q0$$

$$t3 = Q3Q2'Q1'Q0'$$

$$t = t6 \text{ or } t3$$

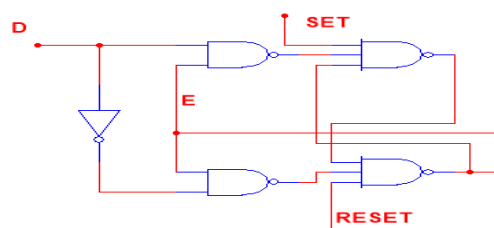
#### 4.4 לחצן תיעדוף

##### 4.4.1 D-latch (NAND Gates)

מעגל זה יכול לתפקד כ- D-LATCH או כ- S-R על מנת להגדיר את המעגל כ- D-LATCH יש על הכניסות SET ו-RESET להיות ב"1" לוגי. על מנת להגדיר את הרכיב כ- SR יש לקבוע את כניסת האפשרות ה- D בערך לוגי קבוע. במעגל שלנו במצב התחלתי כניסות אלה מוזנות ב"1" לוגי, אך כאשר התנאי של ההשהיה מתקיים רגל ה- RESET מקבלת פולס של "0" לוגי על מנת לאפס את הנועל. רגל D במערכת שלנו מחוברת ללחצן וכאשר היא מקבלת "1" לוגי, היא יודעת לשמור את הערך עד לאיפוס המעגל על ידי הרגל RESET. הערה: הרכיב ממומש באמצעות שערים לוגיים לאור הצורך בתצוגה תלת ממדית.

E	SET	Reset	D	Q	$\bar{Q}$
X	0	0	X	1	1
X	0	1	X	0	1
X	1	0	X	1	0
1	1	1	1	1	0
0	1	1	0	0	1

איור 13. טבלת אמת של D. Latch



איור 14. מעגל המבוסס D. Latch





## 4.6 מערכת תצוגה

### 4.6.1 מונה בינארי

מונה הוא התקן אלקטרוני שסופר מספר אירועים מסוג מסוים אשר קרו. בדרך כלל בהתאמה לאותות שעון (ירידה ועליה של אות שעון), במערכת זו נעשה שימוש במונה CD4516B המונה מטה החל מערך המשתנה כתלות בלחצן התיעדוף- כאשר לחצן התיעדוף אינו לחוץ הוא מתחיל את המנייה מ הספרה 6, וכאשר הלחצן תיעדוף לחוץ הוא מתחיל את המנייה מספרה 8. המונה במערכת זו עובד בצורה מסונכרנת עם רמזור הולכי הרגל(המונה יעבוד רק כאשר הרמזור ירוק) והשעון הכללי של המערכת.

### 4.6.2 קוד עשרוני BCD

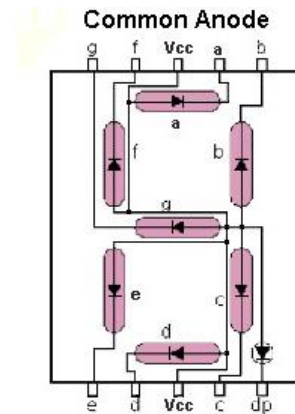
עשרוני בקידוד בינארי (הידוע יותר בראשי התיבות BCD - Binary Coded Decimal) היא דרך הקידוד הנפוצה ביותר של ספרות עשרוניות במחשבים ובמערכות אלקטרוניות. כל ספרה עשרונית מיוצגת על ידי 4 סיביות, כאשר ערכן של הסיביות מימין לשמאל הוא 1,2,4,8. זהו מספר הסיביות המינימלי שיכול לייצג באופן בינארי את כל הספרות מ-0 עד 9.

### 4.6.3 Seven segment

זהו רכיב המורכב משמונה דיודות LED היכולות להציג את 10 הספרות הבסיסיות 0-9 בהצגה עשרונית על פי העיקרון שכל רגל כניסה של הרכיב מייצגת דיודת LED אחת. התקני ה-Seven segment מחולקים לשתי קבוצות: אנודה משותפת (Common anode) – כל הדיודות מחוברות ל-VDD באנודה שלהם על מנת שהמקטעים ידלקו, הקתודה של הדיודה צריכה לקבל 0 לוגי, כאשר הקתודה מקבל 1 לוגי המקטע נכבה. במערכת זה נעשה שימוש בקבוצה זו.

קתודה משותפת (Common cathode) – כל הדיודות מחוברות לאדמה בקתודה שלהם על מנת שהמקטעים ידלקו, האנודה צריכה לקבל 1 לוגי, כאשר הקתודה מקבל "0" לוגי המקטע נכבה.

LED	רגל
E	1
D	2
C	4
DP	5
B	6
A	7
F	9
G	10



איור 17. סכמת המתארת את חיבור רגליים בצג

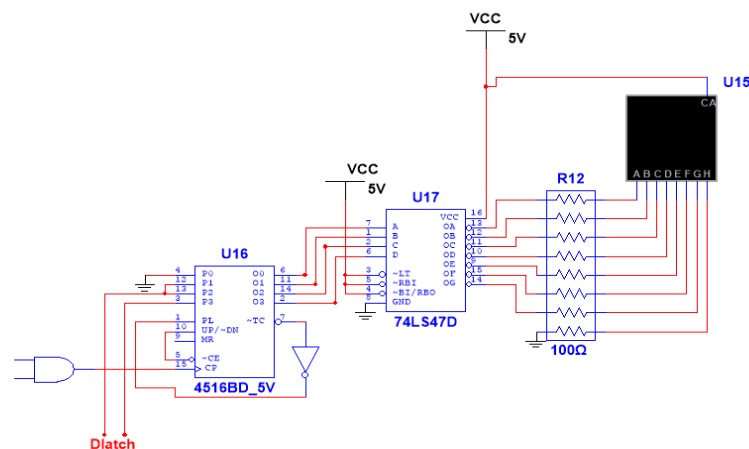
איור 18. טבלה המתארת את חיבורי הרגליים לצג

#### 4.6.4 חישוב מערכת הגנה לדיודות

בין הרכיב Seven segment לרכיב BCD to Seven segment יש לשים נגד הגנה בטור לדיודה כדי שהדיודות לא ישרפו במהלך הדלקתם בזרם גבוה מדי. על מנת לחשב את הנגד המינימלי שניתן לחבר הוכנס מתח כניסה של 5V ומתוך דפי הנתונים של Seven segment נתון שהמתח הנופל על דיודת LED אדומה הוא 3.45V והזרם המקסימלי הזורם עליה הוא 20mA, הנגד המינימלי חושב על פי נוסחה 1

$$R \frac{V_{DD} - V_D}{I_D} = \frac{5 - 3.45}{20m} = 77.5[\Omega]_{min}$$

מתוך הערך המינימלי של הנגד הוחלט להשתמש בנגד של 100Ω עבור כל LED של ה Seven segment. סך הכל שבעה נגדים עבור התצוגה A-G.

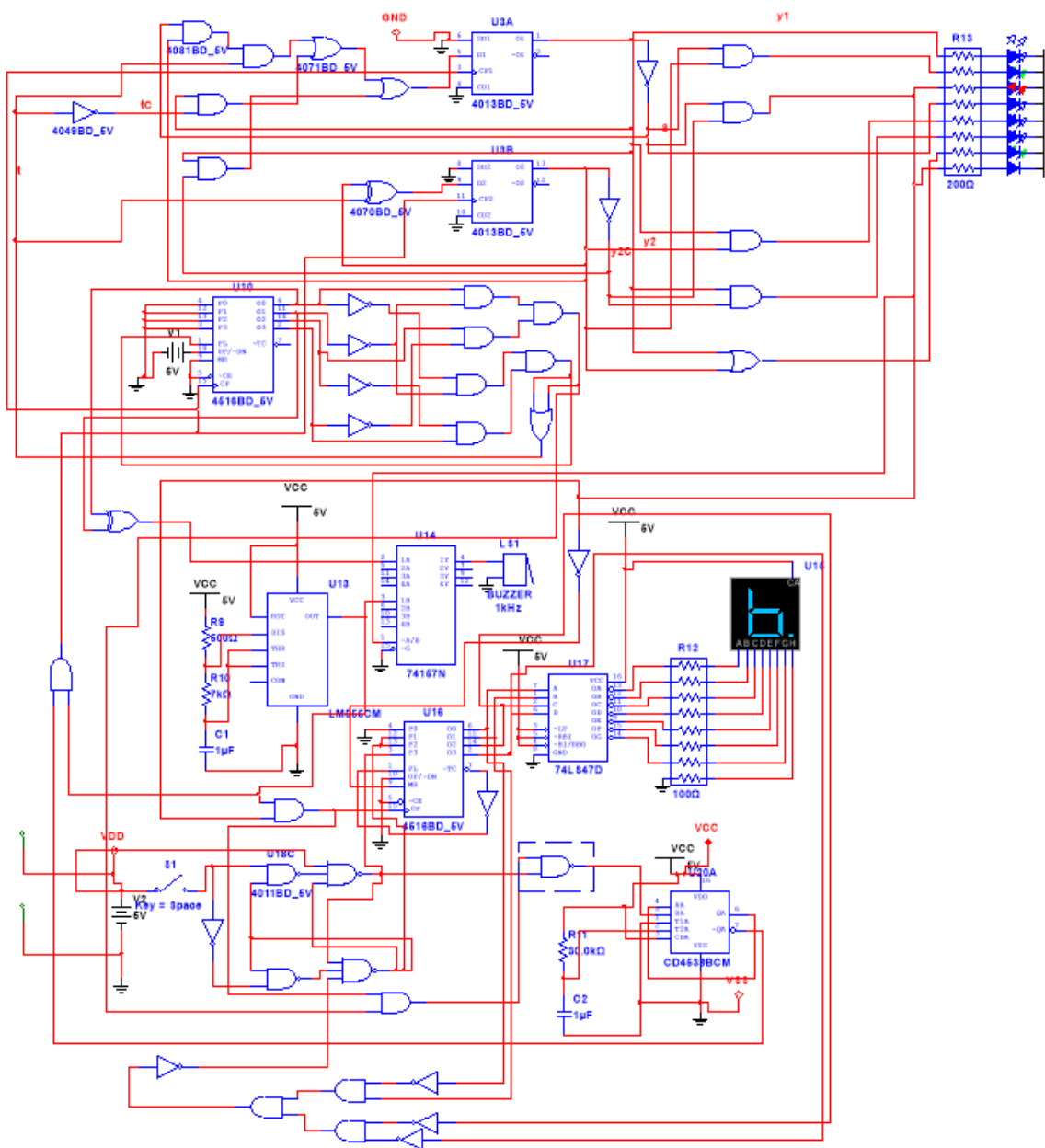


איור 19. מערכת תצוגה בשלמותה

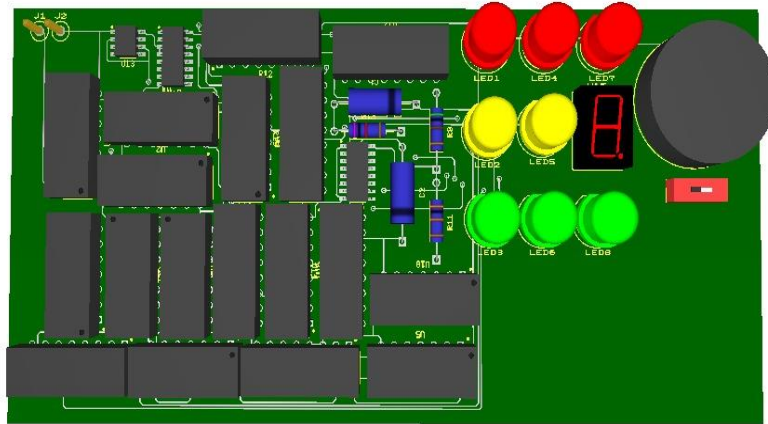
## פרק 5 : תוצאות ומסקנות

### 5.1 תוצאות הפעלת המערכת

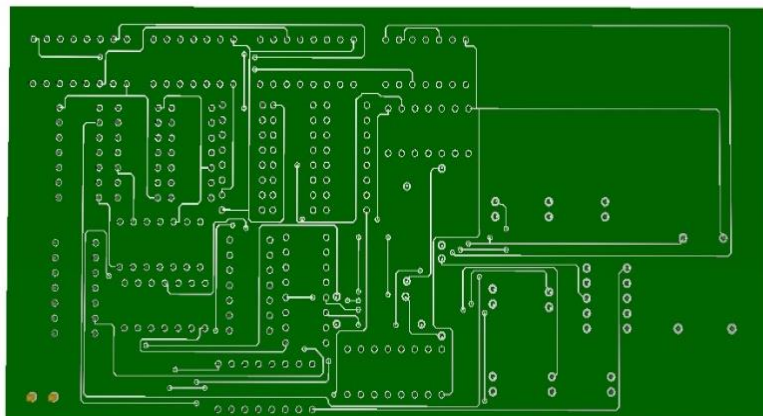
תחילה יש לציין שהמערכת תוכננה באמצעות Multisim שהינה תוכנה נוחה ושימושית לבצוע סימולציות של מעגלים חשמליים שונים, המערכת לא מומשה בפועל לאור דרישות הקורס תחת תנאים מיוחדים. המערכת הופעלה במלואה, וישנם תופעות מעבר בזמן המנייה של המצב הראשון מופיע בצג 8 שניות במקום 6 שניות כתוצאה מתוכנת הסימולציה. להלן המערכת במלואה:



איור 20. מערכת צומת רמזורים בשלמותה

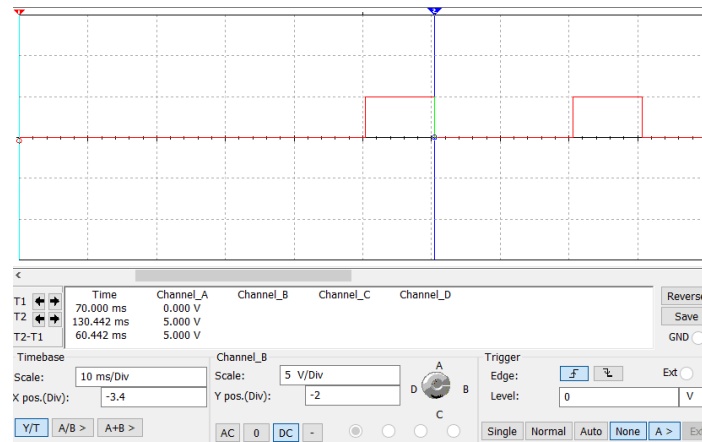


איור 21. Board design של מערכת הרמזורים – מבט עליון (145X70mm)

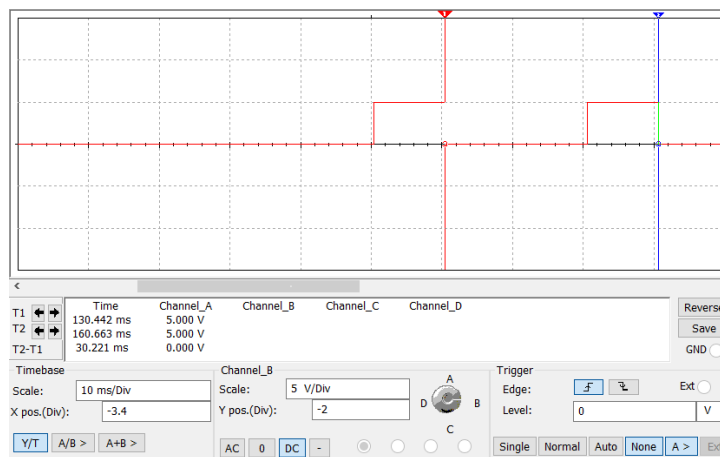


איור 22. Board design של מערכת הרמזורים – מבט תחתון

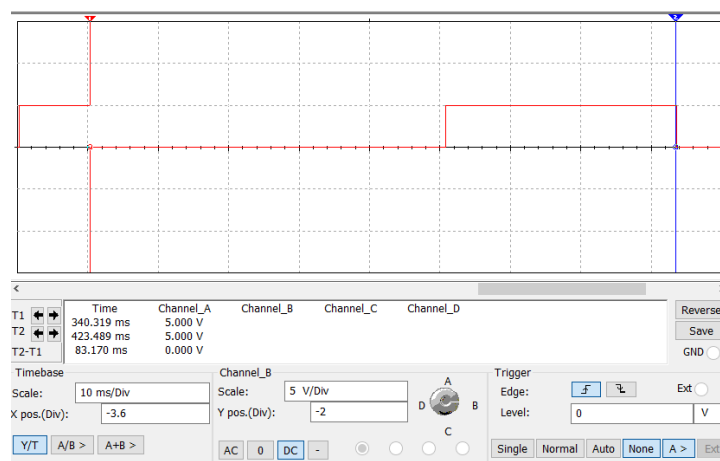
התוצאה שהתקבלה במדידת ה TIMER כפי שניתן לראות באיורים (23-25) הינה מדויקת עבור הפעלת כלל המצבים (כאמור לעיל 10m[sec] מייצג שניה אחת).



איור 23. מדידת מתח המוצא של הזמן עבור המצב הראשון

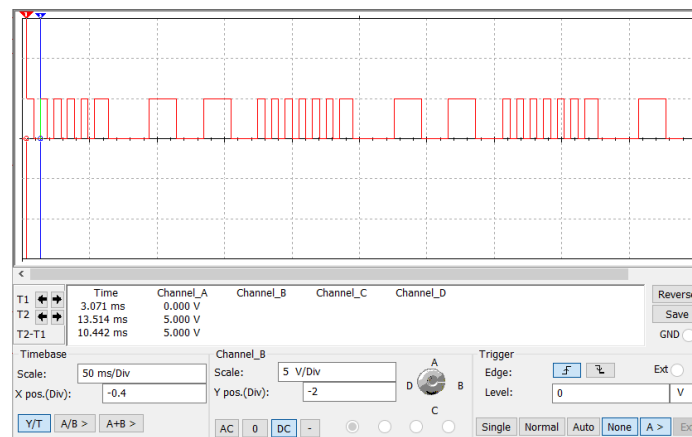


איור 24. מדידת מתח המוצא של הזמן עבור המצב השני

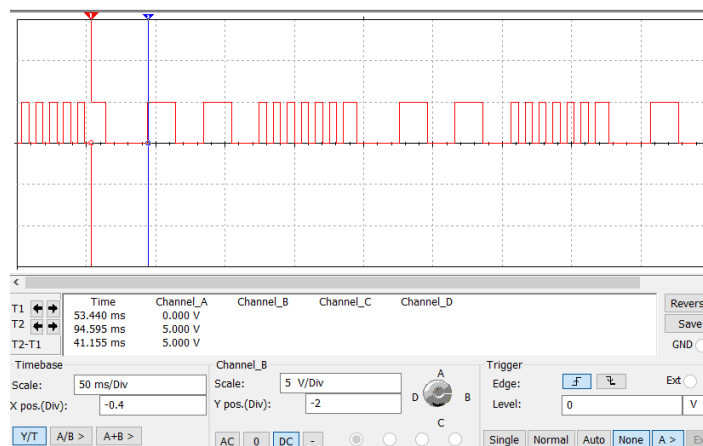


איור 25. מדידת מתח המוצא של הזמן עבור מצב לחצן התיעדוף

תוצאות מדידה עבור המחלק תדר בכניסה לזמזם, כפי שניתן ניתן לראות באיורים (26-27)  
שהתדר חולק ב 4.

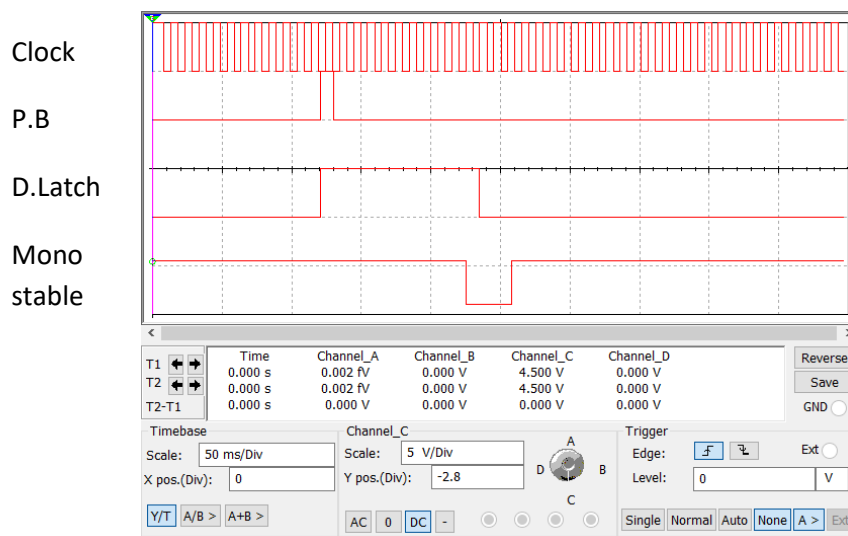


איור 26. מדידת תדר המוצא עבור אור ירוק ברמזור הולכי רגל



איור 27. מדידת תדר המוצא עבור אור אדום ברמזור הולכי רגל

תוצאות המדידה באיור 28 מרגע השימוש בלחצן תיעדוף וקליטתו במערכת הזיכרון ועד הפעלת ההשהיה של כשתי שניות בעת תחילת המחזור הבא, ואיפוס מערכת הזיכרון לאחר המימוש.



איור 28. מתח מוצא מערכת ההשהיה מרגע לחיצה ועד ביצוע ההשהיה במצב הדרוש

## 5.2 מימוש והפעלת המערכת עבור בקר Arduino

אַרְדּוּאִינוֹ- הוא מיקרו-בקר בעל מעגל מודפס יחיד (Single Board Micro Controller), עם סביבת פיתוח משולבת (IDE) ברישיון קוד פתוח, אשר מטרתה ליצור סביבה נוחה וזולה לפיתוח פרויקטים המשלבים תוכנה עם רכיבי אלקטרוניקה.

המערכת מומשה באמצעות סימולטור (Tinker Card) הכולל הרכבת מעגל חשמלי וכתיבה קוד באמצעות שפת C.

```
//define Variables
int red1 = 4;
int yellow1 = 3;
int green1 = 2;
int red2 = 7;
int yellow2 = 6;
int green2 = 5;
int red3 = 10;
int green3 = 9;
const int buzzer = 1;
const int input1 = 13;
int input1s = 0;

const int STcp = 12; //Pin connected to ST_CP of 74HC595
const int SHcp = 8; //Pin connected to SH_CP of 74HC595
const int DS = 11; //Pin connected to DS of 74HC595
int dataArray[16] = {0, 96, 218, 242, 102, 182, 190, 224, 254}; //display 0,1,2,3,4,5,6,7,8

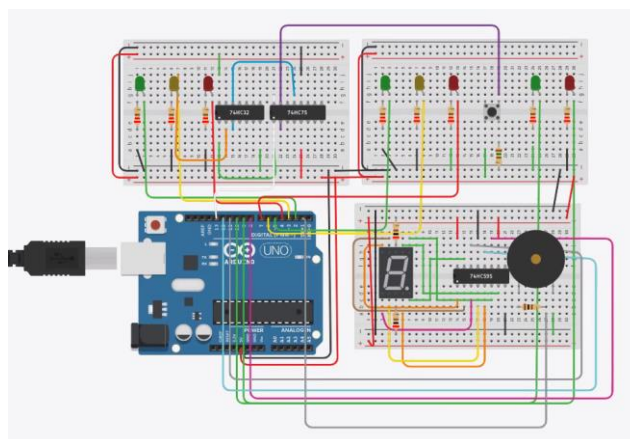
void setup() {
    pinMode(red1, OUTPUT);
    pinMode(yellow1, OUTPUT);
    pinMode(green1, OUTPUT);
    pinMode(red2, OUTPUT);
    pinMode(yellow2, OUTPUT);
    pinMode(green2, OUTPUT);
    pinMode(red3, OUTPUT);
    pinMode(green3, OUTPUT);
    pinMode(input1, INPUT);
    pinMode(buzzer, OUTPUT);
    pinMode(STcp, OUTPUT);
    pinMode(SHcp, OUTPUT);
    pinMode(DS, OUTPUT);
}

void loop()
{
    State1();
    State2();
}

void State1()
{
    input1s = digitalRead(input1);
    if (input1s == HIGH)
    {
        for (int num = 8; num >= 0; num--) //display 8-1 and turn on a green1 led
        {
            digitalWrite(green1, HIGH);
            digitalWrite(red2, HIGH);
            digitalWrite(green3, HIGH);
            digitalWrite(STcp, LOW); //ground ST_CP and hold low for transmitting
            shiftOut(DS, SHcp, MSBFIRST, dataArray[num]);
            digitalWrite(STcp, HIGH); //pull the ST_CP to save the data
            tone(buzzer, 500); // Send 500Hz sound signal...
            delay(500); // ...for 0.5 sec
            noTone(buzzer); // Stop sound...
            delay(500);
        }
        digitalWrite(green1, LOW);
        digitalWrite(red2, LOW);
        digitalWrite(green3, LOW);
    }
    else
    {
        for (int num = 6; num >= 0; num--) //display 6-1 and turn on a green1 led
        {
            digitalWrite(green1, HIGH);
```



```
digitalWrite(red2,HIGH);
digitalWrite(green3,HIGH);
digitalWrite(STop,LOW); //ground ST_CP and hold low for transmitting
shiftOut(DS,SHcp,MSBFIRST,dataArray[num]);
digitalWrite(STop,HIGH); //pull the ST_CPST_CP to save the data
tone(buzzer, 500); // Send 500Hz sound signal...
delay(500); // ...for 0.5 sec
noTone(buzzer); // Stop sound...
delay(500);
}
digitalWrite(green1,LOW);
digitalWrite(red2,LOW);
digitalWrite(green3,LOW);
}
for(int num = 3 ;num >=0; num--) //diaplay 3 to 1 and turn on the yellow1 led
{
    digitalWrite(yellow1,HIGH);
    digitalWrite(red2,HIGH);
    digitalWrite(red3,HIGH);
    tone(buzzer, 250); // Send 250Hz sound signal...
    delay(500); // ...for 0.5 sec
    noTone(buzzer); // Stop sound...
    delay(500);
}
digitalWrite(yellow1,LOW); //turn off the yellow led
digitalWrite(red2,LOW); //the red led finally turn off
}
void State2()
{
    digitalWrite(red1,HIGH);
    for(int num = 6; num >=0; num--) //display 6-1 and turn on a green2 led
    {
        digitalWrite(green2,HIGH);
        tone(buzzer, 250); // Send 250Hz sound signal...
        delay(500); // ...for 0.5 sec
        noTone(buzzer); // Stop sound...
        delay(500);
    }
    digitalWrite(green2,LOW);
    digitalWrite(red1,LOW);
    for(int num = 3 ;num >=0; num--) //diaplay 3 to 1 and turn on the yellow2 led
    {
        digitalWrite(yellow2,HIGH);
        digitalWrite(red1,HIGH);
        tone(buzzer, 250); // Send 250Hz sound signal...
        delay(500); // ...for 0.5 sec
        noTone(buzzer); // Stop sound...
        delay(500);
    }
    digitalWrite(yellow2,LOW);
    digitalWrite(red1,LOW);
    digitalWrite(red3,LOW);
}
}
```



איור 29. מימוש המערכת באמצעות בקר Arduino

## 5.3 בעיות ופתרונות

### שעון

בעיה- בעת התכנון של השעון ה DUTY CYCLE לא היה על 50%.

פתרון- הבעיה נבעה מהעובדה שהיה הפרש יחס גדול בין הנגדים האחראים על הטעינה והפריקה של הקבל, הבעיה נפתרה בכך שערכו של הנגד R9 נקבע לערך זניח ביחס לנגד R10 ובכך התקבלו זמני טעינה ופריקה שווים.

### מערכת בקרה

בעיה- המערכת בקרה בנויה מ רכיבי זיכרון מסוג D FF, המקבלים מידע מהזמן מצבים. השינוי של הזמנים התקיים בעת עליית השעון של הדלגלג ולכן לא הייתה שליטה על המידע הנדגם ועקב כך המערכת לא הייתה יציבה.

פתרון- הפתרון של דגימת המידע הינו על ידי דיוק ב setup and hold time, אשר הושגו באמצעות הזזת והגדלת פולסי המידע של הזמן. כתוצאה מכך המערכת דגמה את המידע הרצוי בצורה יציבה ומדויקת.

### לחצן תיעדוף ומערכת ההייה

בעיה- מערכת ההייה מחוברת לכניסות השעון של כלל רכיבי המערכת בקרה, כאשר בעת הפעלתה היא פעלה בצורה מסונכרנת לאותות מידע הנכנסים ל דלגלגים, ומכאן לא הייתה שליטה על המידע הנדגם ועקב כך המערכת לא הייתה יציבה.

פתרון- הפולס הורחב ל 3 שניות במקום שתי שניות, אשר סיפק למערכת מרחב hold time שנתן שליטה על דגימת המידע. הרחבת הפולס לא השפיעה על זמן ההשהיה הנדרש.

בעיה- בעת שמצב הלחצן תיעדוף התקיים, המערכת זיכרון של הלחצן התאפסה בתחילה המצב וכתוצאה מכך התאפשרה קבלת לחיצות נוספות אשר גרמו לחזרה נוספת על המצב.

פתרון- מומשה פונקציה לוגית אשר קיימה תנאים לאיפוס המערכת זיכרון בסוף קיום המצב, מכאן המערכת חסמה כל שימוש לא רצוי בלחצן.

### צג מנייה

בעיה- מערכת הצג מנייה פעלה באופן רצוף, אשר הדבר לא ענה על דרישות הלקוח.

פתרון- מומשה פונקציה לוגית אשר הפעילה את המערכת בצורה מסונכרנת עם השעון הכללי, והמצב הרצוי להפעלת הצג.

## 5.4 מסקנות הפרויקט

בעת תכנון ומימוש המערכת התקבלו המסקנות הבאות:

- בעת תכנון מערכת, יש לחלק את העבודה לשלבים ולתכנן כל חלק בנפרד ולאחר מכן לסכם הכל ביחד כמערכת אחת.
- בעבודה עם דלגלים יש לקחת בחשבון את הזמנים הבאים:  
 Setup time – הכניסות לרכיב חייבות להיות חוקיות לפחות ts לפני מופע השעון.  
 Hold time – הזמן שבו הכניסות של הרכיב חייבות להישמר בערכים חוקיים אחרי מופע השעון.  
 אחרת לא ניתן לדעת איזה מידע הדלגלג דוגם בכניסותיו ובכך המערכת לא תהיה יציבה.
- בעבודה עם רכיבי דיודות למיניהם יש לקחת בחשבון הוספת נגד הגנה בטור על מנת שהדיודות לא יישרפו עקב עומס.
- רכיב הנועל הינו רכיב אשר עובד ללא שעון כלל ומוציא את המוצאים שלו לפי טבלת האמת, אך לעומתו הדלגלג הינו רכיב אשר מסונכרן עם שעון.
- כאשר עובדים עם חד יציב יש לקחת בחשבון את זמן הטעינה של הקבל ביחס למחזור אות הדרבון.
- בעת קביעת תנאי עבודה לרכיב כלשהו, יש לשים לב שהתנאים אינם מתנגשים אחד עם השני (למשל- במערכת שלנו תנאי אפשרור הלחצן תיעדוף והמערכת השנייה אינם יכולים להיות מופעלים מאותו תנאי, אחרת המערכת לא תהיה יציבה).
- כדי לבנות כרטיס (Board Design) אופטימלי, יש לנצל בהרכבת המעגל בסימולטור את כל רגלי הרכיבים, על ידי חלוקה למשפחות שערים (section) אופטימלית.

## 6. סיכום

בפרויקט זה תוכננה צומת רמזורים בטוחה ונגישה להולכי רגל וללקויי ראייה בפרט. המערכת תוכננה בצורה לוגית וגם באמצעות בקר Arduino. מימוש באמצעות טכנולוגית ה Arduino פשוט וקל לתכנון, לעומת מימוש באמצעות תכנון לוגי אשר דרש עבודה מורכבת ומאתגרת חשיבתית, וזה ניקר כאשר המעגלים סוכמו כמערכת אחת. בתכן לוגי היה לנו קשה לסנכרן את כל הרכיבים כאחד (תזמונים ושעונים) לעומת הארדואינו שתופעות אלה לא קיימות. כמו כן, בעת מימוש תנאים בתכן לוגי באמצעות רכיבים ופונקציות היה לנו מאתגר ליישם תנאים אלו בלי לפגוע ביציבות המערכת לעומת כתיבת קוד פשוט ויציב בארדואינו.

## 7. מקורות

הרצאות טכניון, פרופסור אהרון אהרון -

<https://www.youtube.com/watch?v=kY2OPQhckj4>

ספר לימוד - א. אהרון: "תכן לוגי", "האוניברסיטה הפתוחה 1989.

קורס מעבדת אלקטרוניקה 2 - "אוניברסיטת אריאל בשומרון".

---

<sup>i</sup> <https://sites.google.com/site/yuliaavezov84/23>

<sup>ii</sup> [https://ecowiki.org.il/wiki/%D7%A4%D7%A7%D7%A7%D7%99\\_%D7%AA%D7%A0%D7%95%D7%A2%D7%94\\_%D7%91%D7%99%D7%A9%D7%A8%D7%90%D7%9C](https://ecowiki.org.il/wiki/%D7%A4%D7%A7%D7%A7%D7%99_%D7%AA%D7%A0%D7%95%D7%A2%D7%94_%D7%91%D7%99%D7%A9%D7%A8%D7%90%D7%9C)

<sup>iii</sup> קורס קדם פרוייקט – קובץ הנחיות "מימוש ותכנון מערכת רמזורים"