



אוניברסיטת בן-גוריון בנגב

Ben-Gurion University of the Negev

פרויקט גמר

קורס: מבנה מחשבים ספרתיים

361-1-4191

**light source and object proximity
detector system**

מערכת לגילוי מקורות אור

וניטור אובייקטים במרחב

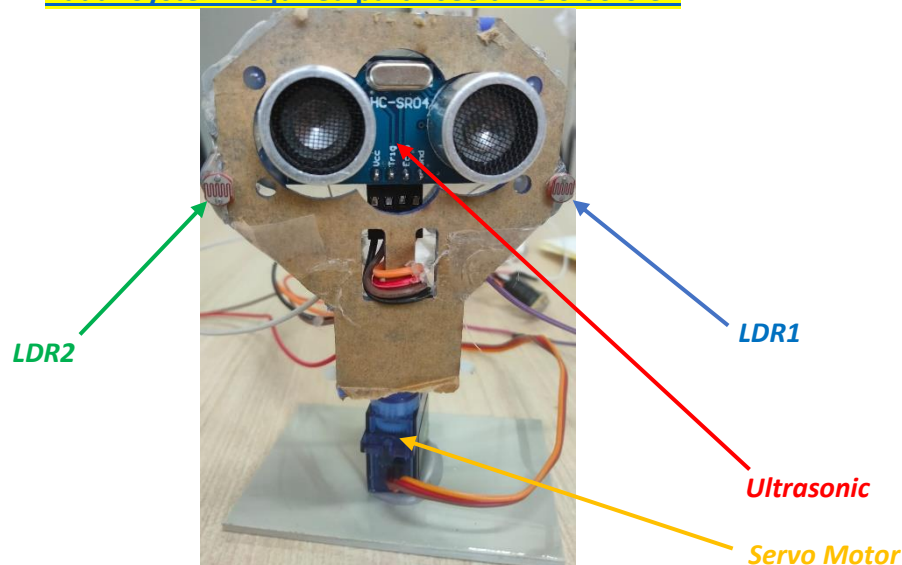
01/06/2023

<u>A</u>	מטרת הפרויקט:	3
<u>B</u>	תיאור משימת הפרויקט:	4
1.	Objects Detector System : (משקל 35%)	4
2.	Telemeter : (משקל 10%)	5
3.	Light Sources Detector System : (משקל 30%)	5
4.	Light Sources and Objects Detector System : (סעיף בונס במשקל 20%)	5
5.	Script Mode : (משקל 25%)	5
<u>C</u>	הסברים טכניים – חיישן ומנוע סרבו:	7
1.	Ultrasonic מרחק (רכיב HC-SR04): חיישן מרחק	7
2.	Servo Motor מנוע:	9
3.	Servo מנוע ע"י מנוע ה-Servo:	9
4.	LDR כגלאי מרחק של מקור אור:	10
5.	חיבורי חומרה והקצאת רגלי הבקר (ערכת פיתוח אישית):	10
6.	ממשק משתמש בצד ה-PC :	11
<u>D</u>	דו"ח מכין: (משקל 10%)	11
<u>E</u>	מבנה הציון בפרויקט:	11

A. מטרת הפרויקט:

- i. תכנון ומימוש מערכת מבוססת MCU לגילוי מקורות אור וניטור אובייקטים במרחב באמצעות מד מרחק אולטראסוניק, חיישני אור LDR ומנוע Servo. סריקת המרחב תבוצע בגזרה של 180 מעלות באמצעות תנועת מנוע Servo, מדידת מרחק באמצעות מד המרחק בעל טווח מדידה בין $2\text{cm} \div 450\text{cm}$. בקרת התנועה הזוויתית של מנוע Servo תהא מבוססת PWM.
- ii. במסגרת הפרויקט יפותח קוד בשפת C++/C למימוש מערכת Embedded מבוססת גרעין הפעלה FSM וכתיבת המערכת במתודולוגיה של שכבות אבסטרקציה למימוש מערכת Embedded מרובת חיישנים תחת משטר Hard Real Time.
- iii. מחשב PC ישמש לצורך ממשק GUI ([PySimpleGUI](#), [Tkinter](#), etc) למשתמש ולתצוגה לכל פעולה המוגדרת במערכת ודורשת תצוגה וממשק למשתמש. ה-MCU יחובר למחשב ה-PC באמצעות תקשורת טורית אסינכרונית בסטנדרט RS-232.
- iv. ממשק למשתמש בצד ה-PC יאפשר קביעת פרמטרים, שליחת קבצים ופקודות High-level ל-MCU. הממשק בצד ה-PC יכתב בשפה עילית (לבחירתכם: Python, Matlab, C++, JAVA או שימוש במעטפת C# - מומלץ למי שמכיר) ויתמוך במימוש של תקשורת טורית בין הבקר ל-PC.
- v. הממשק יאפשר העברת קבצים הכוללים פקודות High-level מקודדות למימוש בצד הבקר.
- vi. הגדרת הפרויקט המתוארת בקובץ זה מכילה דרישות של מערכת Embedded בלבד, שלב ראשון בתהליך הפיתוח של המערכת לאחר הבנת הדרישות הוא שלב אפיון המערכת. בשלב זה יש צורך לחקור ולהבין את המגבלות ההנדסיות של המערכת, לכתוב גרף מפורט של גרעין ההפעלה FSM של המערכת, ולכתוב תיאור של האלגוריתמים השונים הנדרשים למימוש במערכת. בשלב חשוב זה נדרש לבצע תכנון מדויק ולפתור את כל השגיאות הלוגיות הקיימות בתכנון (במקביל לשלב זה ניתן לכתוב דרייברים בשכבה המתאימה בנפרד לכל מודל חומרה בפרויקט). רק לאחר מכן ניתן להתחיל בכתיבת קוד המערכת ומימוש האלגוריתמים השונים. **זכרו, הגדרת הפרויקט משאירה שטח "אפור" בו כל זוג צריך להביא את עצמו לידי ביטוי ויצירתיות במימוש אופטימאלי מבחינת סיבוכיות מקום, סיבוכיות זמן לצורך מימוש מערכת רובסטית (מערכת הפועלת בצורה יציבה ואינה "נתקעת") בזמן תגובה RT תחת מגבלות הנדסיות של המערכת.**
- vii. להלן המחשה וויזואלית בלבד של מימוש מערכת Embedded שונה אבל דומה למימוש מערכת רדאר מבוססת MCU לניטור וגילוי אובייקטים במרחב באמצעות מד מרחק אולטראסוניק ומנוע Servo.

Radar System required part - see time 0:06-0:32



B. תיאור משימת הפרויקט:

פיתוח מערכת לגילוי מקורות אור וניטור אובייקטים במרחב באמצעות מד מרחק אולטראסוניק, שני חיישני אור LDR ומנוע Servo לצורך סריקת המרחב.

- ארכיטקטורת התוכנה של המערכת נדרשת להיות מבוססת פרדיגמת תכנות גרעין הפעלה מסוג **FSM** **Simple** המבצעת קטע קוד השייך לאחד ממצבי המערכת בהינתן בקשה של פסיקת RX שמגיעה מה PC לבקר דרך ערוץ התקשורת ל UART. **קוד המערכת נדרש להיות מחולק לשכבות אבסטרקציה כך שיהיה נייד (portable) בקלות בין משפחות בקר MSP430 ע"י החלפת שכבת ה- BSP בלבד.**
- טרם שלב כתיבת הקוד בשלב התכנון נדרש לשרטט גרף של דיאגרמות FSM מפורטות, אחת של ארכיטקטורת התוכנה של המערכת בצד MCU והשנייה של חלק התמיכה בתקשורת באפליקציה בצד מחשב ולצרפן לדו"ח מכין. גרף של דיאגרמת FSM בצד MCU המצבים אלו הצמתים והקשתות אלו המעברים ממצב למצב בגין בקשות פסיקת RX (המסווגות לקליטת מידע מסוג Command ומסוג Data, כפי הנלמד בניסוי מעבדה 4).
- אסור לבצע שהייה ע"י שימוש ב poling למעט עבור debounce ברוטינת שירות של בקשות פסיקה בגין לחצנים.
- **בתחילת התוכנית (בלחיצה על כפתור RESET), הבקר נמצא במצב שינה.**
הערה: כפתור RESET מותר לשימוש אך ורק לאתחול המערכת בלבד
- רמת הדיוק, זמן תגובת המערכת והביצוע בהתאם לדרישות מהווה חלק חשוב בהערכת הפרויקט.
- מקוריות העבודה היא חלק חשוב בביצוע הפרויקט, במקרה של העתקה, הפרויקטים של שני הצדדים ייפסלו.
- עקב מגבלה של גודל ה RAM עליכם להשתמש בתבונה בזיכרון ה FLASH (ראו חומר עזר במודל).
- נדרש לעבוד בסביבת פיתוח **CCS IDE** מבוססת **Eclipse** (כפי הנלמד בחומר ההכנה למעבדה 1).
- לצורך חלק הביצוע של המשימה נדרש ליצור ממשק למשתמש בצד ה- PC המכיל את סעיפי התפריט הבא:

להלן פירוט סעיפי התפריט:

1. **Objects Detector System : (משקל 35%)**

מימוש מערכת Objects Detector System לניטור אובייקטים (באופן דינאמי) במרחב במרחק מוגדר דרך ממשק למשתמש בביצוע סריקה אחת בלבד בהיקף סריקה של 180 מעלות וברמת דיוק אופטימאלית.

הסבר:

כפי שמוצג בצורה מוחשית בסרטון המצורף, חישת הגלאי נעשית ב- 180 מעלות סביב נקודת המרכז של ידית מנוע Servo במרחק מיסוך המוגדר מראש ע"י המשתמש (דרך הממשק למשתמש בצד ה- PC). משמעות מרחק המיסוך, מרחק שממנו והלאה אנו מתייחסים לערך הנמדד מחוץ לתחום ואינו נלקח בחשבון.

הערה: בשלב התכנון יש צורך להחליט איזה חלק של עיבוד הנתונים יתבצע בצד הבקר ואיזה חלק יתבצע בצד המחשב, בנוסף יש לשלוח את המידע הגדול מ Byte בערוץ התקשורת בצורה דחוסה כך שמספר הבתים הנשלח מינימאלי.

2. Telemeter: (משקל 10%)

נדרש למקם את מנוע הסרבו בזווית הנתונה לבחירה דרך ממשק למשתמש ולהציג את המרחק הנמדד מחיישן המרחק באופן דינאמי ובזמן אמת ברזולוציה של cm (ללא רישום היסטוריית מדידות), על גבי מסך ה-PC.

3. Light Sources Detector System: (משקל 30%)

מימוש מערכת Light Sources Detector System לניטור מקורות אור (באופן דינאמי) במרחב בטווח של עד חצי מטר בביצוע סריקה אחת בלבד בהיקף סריקה של 180 וברמת דיוק אופטימאלית.

הסבר:

בעזרת שני חיישני אור מסוג LDR הממוקמים מצדי חיישן המרחק ניתן לבצע גילוי של מקורות אור מבחינת מיקום ומרחק (ראה פירוט בסעיף C4) בתוך טווח סריקה של 180 מעלות סביב נקודת המרכז של ידית מנוע Servo ולהציג את תוצאות המיקום והמרחק של מקורות האור על מסך ה-PC דרך הממשק למשתמש.

הערה: בשלב התכנון יש צורך להחליט איזה חלק של עיבוד הנתונים יתבצע בצד הבקר ואיזה חלק יתבצע בצד המחשב, בנוסף יש לשלוח את המידע הגדול מ Byte בערוץ התקשורת בצורה דחוסה כך שמספר הבתים הנשלח מינימאלי.

4. Light Sources and Objects Detector System: (סעיף בונס במשקל 20%)

מימוש מערכת Light Sources and Objects Detector System , מערכת זו מבצעת יחד את הדרישות של סעיפים 1,3 בתפריט לצורך ניטור מקורות אור במרחב בטווח של עד חצי מטר וניטור אובייקטים במרחב במרחק מוגדר דרך ממשק למשתמש בביצוע סריקה אחת בלבד בהיקף סריקה של 180 וברמת דיוק אופטימאלית.

הערה: בשלב התכנון יש צורך להחליט איזה חלק של עיבוד הנתונים יתבצע בצד הבקר ואיזה חלק יתבצע בצד המחשב, בנוסף יש לשלוח את המידע הגדול מ Byte בערוץ התקשורת בצורה דחוסה כך שמספר הבתים הנשלח מינימאלי.

5. Script Mode: (משקל 25%)

הפעלת כל המערכת בהתאם לקובץ script המכיל פקודות High Level המוגדרות מראש. ניתן לתפעל את המערכת באופן אוטומטי ולבדוק את כל חלקי המערכת. נדרש לתמוך ביכולת שליחה וקבלה של עד שלושה קבצים ולבחור להפעיל אחד מהם בנפרד ובאופן בלתי תלוי בבחירה מתוך התפריט בצד מחשב בלבד. **הדגש בתכנון הוא של זמן תגובת המערכת עם רמת דיוק ביצוע של תוכן קובץ ה scripts ואמינות תוכן המידע הנשלח/מתקבל דרך ערוץ התקשורת.**

הערה: ניתן להניח שכל קובץ script בנפרד יכול להכיל עד מקסימום עשר שורות

הסבר:

המשתמש יכול לשלוח לבקר קובץ script.txt המכיל פקודות ברמת High Level (כמפורט בהמשך). טעינת הקובץ נעשית בלחיצת כפתור מתאים דרך הממשק למשתמש ולאחר מכן שליחת הקובץ מהמחשב האישי לבקר באופן טורי תו אחר תו (ללא קידוד) הקובץ נשמר בזיכרון ה-FLASH של הבקר כקובץ txt . לאחר קבלת הקובץ בצד הבקר, תישלח הודעת Acknowledge למסך ה-PC ויתחיל ביצוע ה- script בצד הבקר.

צורת שמירת קובץ *.txt בצד הבקר:

קובץ מוגדר ע"י רצף פיזי של תווים שמיקומו ההתחלתי נתון ע"י מצביע לקובץ ותוכנו מסתיים בתו EOF. עליכם לנהל שמירה של עד שלושה קבצים בזיכרון FLASH של הבקר ולהגדיר **struct** מתאים המכיל את השדות הבסיסיים הבאים (ניתן להוסיף שדות עזר לבחירתכם תוך נימוק הנדסי): כמות קבצים קיימים, מערך מצביעים לשמות הקבצים, מערך מצביעים לתחילת כל קובץ, מערך המכיל את גודלי הקבצים.

רשימת פקודות High Level נדרשות לתמיכה במצב של Script Mode:

OPC (first Byte)	Instruction	Operand (next Bytes)	Explanation
0x01	inc_lcd	x	Count up from zero to x with delay d onto LCD
0x02	dec_lcd	x	Count down from x to zero with delay d onto LCD
0x03	rra_lcd	x	Rotate right onto LCD from pixel index 0 to pixel index 31 a single char x (ASCII value) with delay d
0x04	set_delay	d	Set the delay d value (units of 10ms)
0x05	clear_lcd		Clear LCD
0x06	servo_deg	p	Point the Ultrasonic sensor to degree p and show the degree and distance (dynamically) onto PC screen
0x07	servo_scan	l,r	Scan area between left l angle to right r angle (once) and show the degree and distance (dynamically) onto PC screen
0x08	sleep		Set the MCU into sleep mode

Note: The default delay **d** value is 50 (**units of 10ms**)

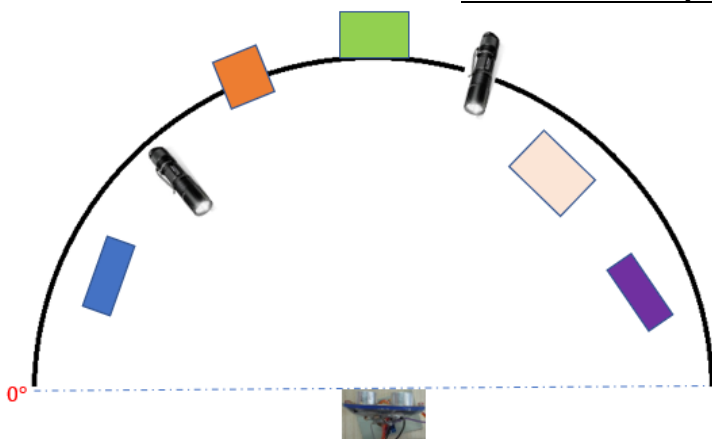
```
inc_lcd 10
set_delay 30
dec_lcd 20
clear_lcd
servo_deg 35
inc_lcd 20
servo_scan 20,60
sleep
```

Python based
translator script



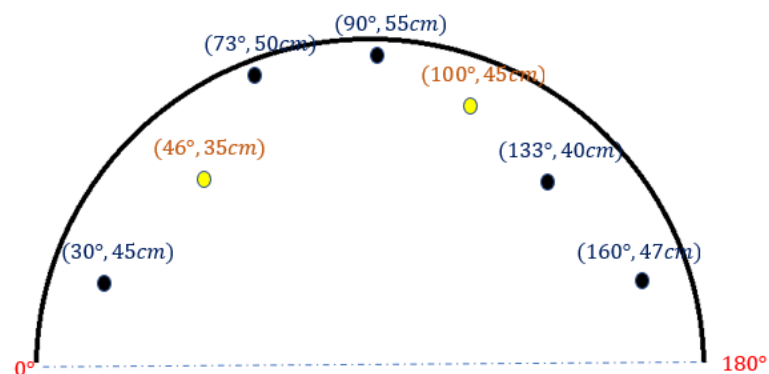
```
010A
041E
0214
05
0623
0114
07143C
08
```

המחשת פעולת מצבי התפריט 1,2,4 של חישת אובייקטים ומקורות אור במרחב:



מיקום פיזי של אובייקטים ומקורות אור במרחב בטווח סריקה של 180 מעלות

©Hanan Ribo



תוצאת הסריקה המרחבית כתמונת פלט בצג המחשב המכילה את זווית מיקום ואת מרחק האובייקטים ומקורות אור במרחב בטווח סריקה של 180 מעלות

C. הסברים טכניים – חיישן ומנוע סרבו:

1. חיישן מרחק Ultrasonic (רכיב HC-SR04):

בהוצאת פולס דרך הבקר ברוחב של **לפחות** 10usec המהווה טריגר דרך רגל **Trigger** של החיישן (מרווח מינימאלי בין טריגר לטריגר הוא 60msec, כלומר תדר עבודה מקסימאלי של 16.7Hz), **בסיום** הפולס חיישן המרחק "יורה" גל קול (Sound wave) באורך שמונה מחזורים בתדר 40kHz לכיוון האובייקט וקולט את ההחזרים המגיעים ממנו. מעגל חשמלי הנמצא בחיישן ממיר את החזרי גל הקול וממיר אותו לפולס היוצא מרגל **Echo**, באורך הזמן שעבר מרגע שידור גל הקול ועד לקבלת ההחזרים מהאובייקט הנמצא מול החיישן. הפולס היוצא מרגל **Echo** של החיישן נכנס לרגל הבקר בעל יכולת פסיקה (דפי המידע והמפרט נמצאים **באתר הקורס ב-Moodle**). טווח המדידה המעשי הוא $2cm \div 450cm$.

סרטון הסבר: Using of Distance sensor Ultrasonic

מדידת המרחק תתבצע בעזרת אחת משתי הנוסחאות הבאות (מימוש בעזרת **Input Capture בלבד**):

$$Range[cm] \cong Echo_high_level_time \cdot \frac{34,000 \left[\frac{cm}{sec} \right]}{2} = Echo_high_level_time \cdot 17,000$$

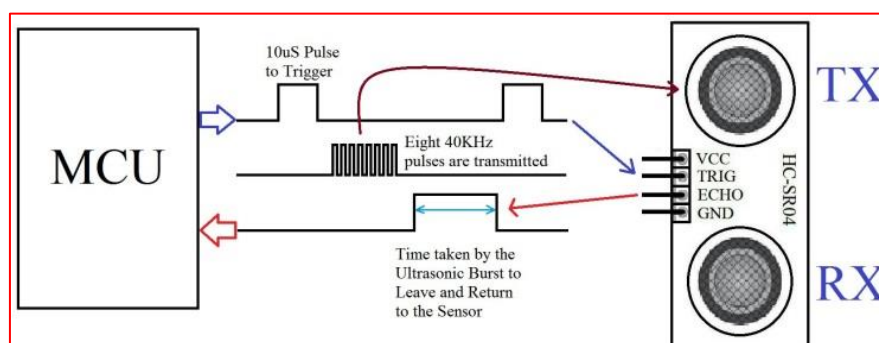
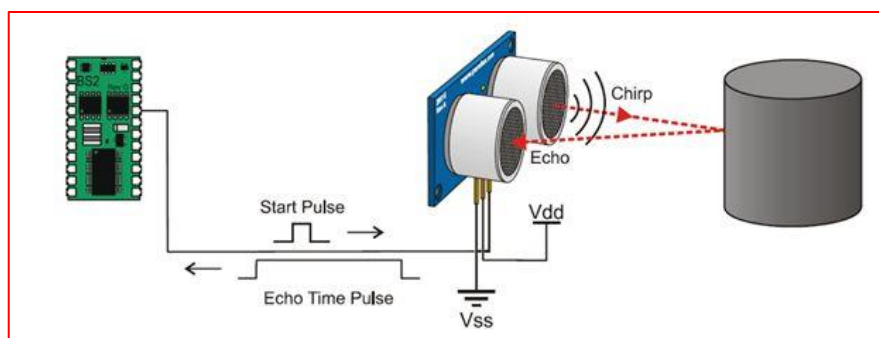
when $34,000 \left[\frac{cm}{sec} \right]$ is the speed of sound c

לצורך דיוק מרבי, נציין שמהירות הקול באוויר תלויה בטמפרטורה, לצורך ציוד נצטרך להתחשב בכך לפי הנוסחה הבאה ([Speed of sound](#)):

$$speed\ of\ sound\ c \left[\frac{m}{sec} \right] = 331.3 + 0.606 \times Temperature_in_Celsius$$

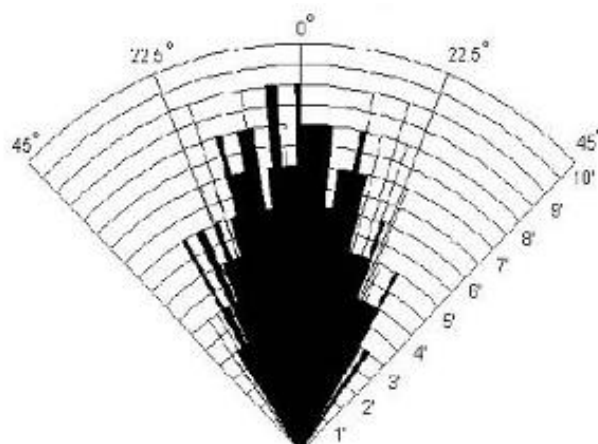
עבור טמפרטורת החדר 25 מעלות צלזיוס (הנחת העבודה בפרויקט), $c = 346.45 \left[\frac{m}{sec} \right] = 34,645 \left[\frac{cm}{sec} \right]$,

הערה: התחשבות בטמפרטורה מאפשרת לשפר את הדיוק במדידת המרחק. לצורך מדידת טמפרטורה משמש חיישן טמפרטורה (בפרויקט זה לא נדרש לבצע מדידת טמפרטורה, הנחת טמפרטורת חדר 25°C).



Label Name	Description	MSP430 Pin
UltraSonic_Echo	אות ECHO המוחזר מחיישן המרחק	Timer Input Capture
UltraSonic_Trigger	אות Trigger הנכנס לחיישן המרחק	PWM output / Digital Output

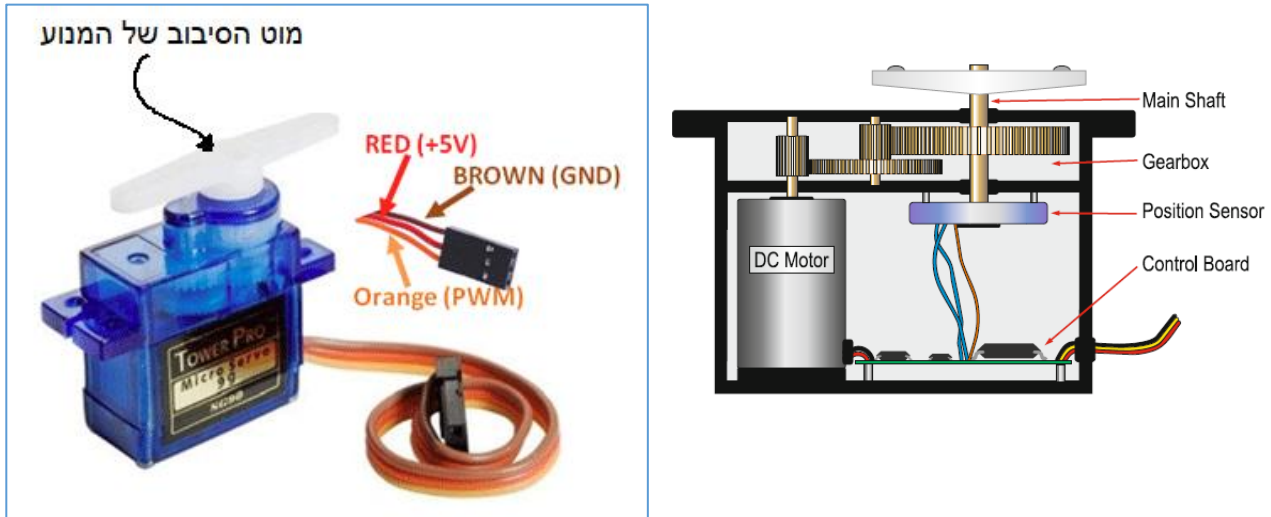
	HC-SR04	HY-SRF05
Working Voltage	5 VDC	5 VDC
Static current	< 2mA	<2 mA
Output signal:	Electric frequency signal, high level 5V, low level 0V	Electric frequency signal, high level 5V, low level 0V
Sensor angle	< 15 degrees	< 15 degrees
Detection distance (claimed)	2cm-450cm	2cm-450cm
precision	~3 mm	~2 mm
Input trigger signal	10us TTL impulse	10us TTL impulse
Echo signal	output TTL PWL signal	output TTL PWL signal
Pins	1. VCC 2. trig(T) 3. echo(R) 4. GND	1. VCC 2. trig(T) 3. echo(R) 4. OUT 5. GND



*Practical test of performance,
Best in 30 degree angle*

2. מנוע Servo Motor:

מנוע Servo ניתן להפעילו כך שמוט הסיבוב שלו, יסתובב בתחום זוויות בין $0^\circ \div 180^\circ$ מעלות. המנוע מהווה עומס, המחובר לכרטיס ממשק המתווך בין הבקר לעומס. מצד אחד נחבר לכרטיס הממשק את ה-MCU המעביר מידע, מצד שני נחבר את המנוע המהווה עומס וצורך הספק גבוה. כרטיס המכיל את הדרייבר החומרתי מחובר למתח הפעלה של 5V ברגל המיועדת לכך (ראה תצלום הבא).



מוצא PWM המחובר למנוע מאפשר שליטה על מיקום זוויתי של מוט הסיבוב של המנוע (מיקום בזווית בין $0^\circ \div 180^\circ$) בעזרת מוצא PWM מהבקר ערך ה-Duty Cycle של אות ה-PWM קובע את מיקום הזרוע. בהגדרות הבאות יצוין ה-PWM המינימאלי עבור מיקום זוויתי של 0° ו-PWM מקסימאלי עבור מיקום זוויתי של 180° . טווחי ביניים של PWM ייתן זוויות בתחום $0^\circ \div 180^\circ$.

Servo Motors

$$f_{max} = 40\text{Hz} \rightarrow T_{min} = 25\text{msec}$$

$$T_{on} = 0.6\text{msec} \quad \text{זווית של } 0 \text{ מעלות}$$

$$T_{on} = 2.5\text{msec} \quad \text{זווית של } 180 \text{ מעלות}$$

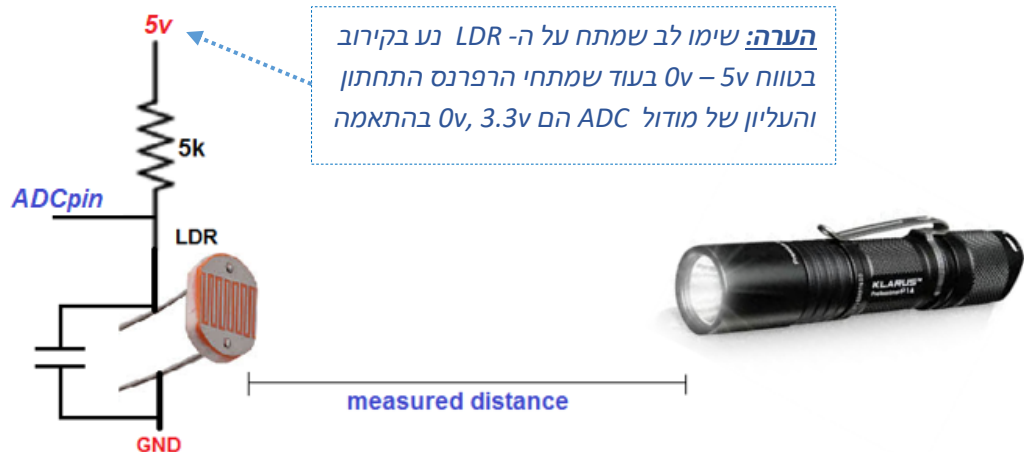
סרטון הסבר: Using of Servo Motor

3. ביצוע סריקה ע"י מנוע ה-Servo:

- ✓ במצב של 0° - מוט הסיבוב של המנוע מכוונים למצב התחלתי קבוע וידוע מראש.
- ✓ כדי להגיע לזווית ספציפית (בשונה מסריקה) אפשר להכתיב PWM עם Duty Cycle מתאים **ללא** צורך בהגעה לזווית בעזרת צעדים קטנים.
- ✓ לצורך סריקה תקינה של מנועי ה-Servo יש לשנות זוויות בתחום של $0^\circ \div 180^\circ$ ע"י שינוי הרגיסטרים השולטים על ה-Duty Cycle, בצעדים של בערך ± 9 עם השתייה של 4msec בין צעד לצעד (תלוי לכמה צעדים תרצו לחלק את 180° של טווח סיבוב המנוע). הסיבה לכך, **בשימוש רב** של תנועות חדות המנוע עלול להשתגע ולצאת משליטה. כמובן שאין לחצות את גבולות הרגיסטרים השולטים על ה-Duty Cycle עבור גבולות של $0^\circ \div 180^\circ$, אחרת המנוע יתחיל לרעוד.
- ✓ כאשר מכבים את אות PWM המזין את מנוע ה-Servo מוט הסיבוב של המנוע ננעלת על הזווית בה הייתה טרם כיבוי האות.

4. חיישן LDR כגלאי מרחק של מקור אור:

בעזרת הבקר ושימוש במודול ADC במחבר למחלק מתח בין נגד LDR ונגד של $4.7k\Omega$, נוכל לדגום מתח משתנה בתחום בין v_{min} ל- v_{max} הנופל על נגד LDR כתלות בעוצמת הארה עליו.



המרחק הנמדד יהיה בטווח $0cm - 50cm$, לצורך מדידת מרחק של מקור האור נדרש להכין מבנה נתונים של המרחק הנמדד בצורת מערך חד-מימדי בגודל 50×2 . אינדקס המערך $d=0 \div 49$ מתאר את המרחק כאשר תוכן התא מתאר את ערך המתח הנמדד על ה- LDR.

בצורה זו קבלנו 50 דגימות, דגימת מתח אחת לכל ס"מ. את המדידות נבצע בכל $5cm$ (סה"כ 10 מדידות) ואת שאר הערכים נחשב בעזרת ההנחה שבין 2 מדידות עוקבות ההתנהגות ליניארית.

לצורך כיול בהגעה לסביבת עבודה חדשה של המערכת (ביחס לתאורת הסביבה) יש צורך בביצוע 10 מדידות לאורך טווח של $50cm$, ניתן להכין פס נייר באורך מתאים ולסמן מידות בכל $5cm$. את פס הנייר נניח על השולחן בקו ישר מנגד ה- LDR. את הפנס (יש לכם כזה במכשיר הנייד) נניח לאורך הסימונים על הסרגל ובכל סימון נלחץ על לחצן לבחירתכם לביצוע דגימת מתח במוצא LDR ולשמירת הערכים המתאימים במערך המיועד לכך. בלחצן נשתמש להתחלה ולסיום תהליך המדידות.

5. חיבורי חומרה והקצאת רגלי הבקר (ערכת פיתוח אישית):

- קישור המכיל סרטון וקובץ בינארי לצריבה ל MCU לבדיקת תקינות מנוע Servo וחיישן אולטראסוניק.
[Video+binary.d43 - Servo Ultrasonic](#)
- רגליים P1.1, P1.2 – אינן בשימוש, תפוסות לתקשורת טורית מול המחשב (ראה הגדרת ניסוי LAB4).
- מנוע Servo – חיבור לרגל מוצא Timer לבחירתכם (PWM).
- חיישן מרחק – חיבור שתי רגליים (Echo, Trig) למוצא וכניסה של Timer לבחירתכם.
- שני חיישני LDR מחוברים לשתי רגליים של הבקר בעלות כניסה אנלוגית לבחירתכם.
- מסך LCD נדרש לחבר את D7-D4 קווי מידע (אופן עבודה של ה- LCD בארבעה קווי מידע) + שלושת קווי הבקרה של ה- LCD לבחירתכם
- חיבור לחצן לבחירתכם לצורך כיול חיישני ה- LDR
- סה"כ חיבור 15 רגליים של הבקר לממשק החומרה הנדרש בפרויקט.

6. ממשק משתמש בצד ה-PC :

שליחת מידע בין הבקר ל-PC מבוססת תקשורת טורית וליצירת תפריט ממשק למשתמש על גבי מסך ה-PC הכולל יכולת הצגה וויזואלית דינאמית למשתמש על גבי מסך ה-PC. עליכם לכתוב את המעטפת והממשק (GUI) בצד ה-PC בכל שפה שתבחרו: Python, MATLAB, C++, או שימוש במעטפת C# - מומלץ. במעטפת זו תצטרכו לתמוך בתקשורת טורית אסינכרונית של המחשב עם הבקר מבוססת סטנדרט RS-232 לצורך העברת תווים וקבצים (העברת הקובץ תו אחר תו) בין המחשב לבקר וההיפך. מצב ברירת המחדל הוא:

9600 BPS , 8-bits , 1 Start , 1 Stop , (none) No parity

C. דו"ח מכין: (משקל 10%)

1. כתיבת דו"ח מכין פרויקט מסכם, לפי הוראות לכתיבה ועריכת דו"ח מכין הנמצא במודל.
 2. צורת הגשה:
- הגשת דוח מכין תיעשה ע"י העלאה למודל של תיקיית zip מהצורה **id1_id2.zip** (כאשר $id1 < id2$).
 - רק הסטודנט עם הת"ז id1 מעלה את הקבצים למודל.
 - התיקיה תכיל את שלושת הפרטים הבאים בלבד:
 - ✓ קובץ **pre_finalx.pdf** – מכיל תשובות לחלק תיאורטי דו"ח מכין
 - ✓ תיקייה בשם **CCS** - מכילה שתי תיקיות, אחת של קובצי source (קבצים עם סיומת *.c) והשנייה של קובצי header (קבצים עם סיומת *.h).
 - ✓ תיקייה בשם **PC_side** - המכילה קובצי מקור של אפליקציית צד מחשב + קובץ ReadMe המתאר בקצרה מה תפקיד כל קובץ מקור במימוש האפליקציה.

D. מבנה הציון בפרויקט:

1. משקל הפרויקט הוא 45% מהציון הסופי - חלק ביצוע (תמיכה בארבעת סעיפי התפריט בסעיף B) משקלו 90% וחלק דו"ח מכין 10%
2. הציון יינתן על-פי הערכה המבוססת על קריטריונים של עמידה ודיוק בדרישות הפרויקט, בקיאות בקוד+אלגוריתם+תיאוריה, דו"ח מכין והגנה על הפרויקט. המשמעות, כל סטודנט בנפרד יידרש לגלות הבנה מעמיקה במרכיבי הפרויקט (תיאוריה, חומרה, תוכנה ואלגוריתם).
3. כל קבוצה תגיש דו"ח מכין לפי קובץ "הוראות לכתיבת דו"ח מכין ודרישות תוכנה" המופיע במודל.

בהצלחה!