

**הטכניון – מכון טכנולוגי לישראל**

**Technion – Israel Institute Of Technology**

**הפקולטה להנדסת חשמל**

**המעבדה לבקרה, רובוטיקה ולמידה חישובית**

**מתן פקודות קוליות לרובוט מסוג**

**BIOLOID**

**מגישות:**

גילי עובד

שיר גראוס

**מנחה:**

קובי כוחי

**שנה:**

2020

## תוכן עניינים

4	תקציר	1
6	מבוא	2
9	תיאור כללי	3
28	תוצאות	4
29	סיכום ומסקנות	5
30	נספחים	6
32	רשימת מקורות	7

## רשימת איורים

6.....	איור 1
7.....	איור 2
8.....	איור 3
9.....	איור 4
10.....	איור 5
10.....	איור 6
10.....	איור 7
10.....	איור 8
11.....	איור 9
11.....	איור 10
11.....	איור 11
12.....	איור 12
12.....	איור 13
13.....	איור 14
13.....	איור 15
15.....	איור 16
15.....	איור 17
16.....	איור 18
16.....	איור 19
17.....	איור 20
17.....	איור 21
18.....	איור 22
18.....	איור 23
19.....	איור 24
19.....	איור 25
20.....	איור 26
21.....	איור 27
21.....	איור 28
22.....	איור 29
22.....	איור 30
22.....	איור 31
23.....	איור 32
26.....	איור 33
27.....	איור 34
31.....	איור 35

## 1. תקציר

פרויקט זה ממחיש מתן פקודות קוליות לרובוט דמוי אדם ובא לידי ביטוי בחיבור ממשק Google Homen לממשק RoboPlus המפעיל את הרובוט מסוג BIOLOID, כאשר סביבת העבודה וחיבור הממשקים נעשית ב-MATLAB.

סביבת העבודה ב-MATLAB היא הנוחה והטריוויאלית כאשר יש צורך בחיבור ממשקים שונים. את הפקודות הבסיסיות להתממשקות עם הרובוט ל-MATLAB ביצענו באמצעות פרויקטים קודמים המוצגים באתר המעבדה בקישורים הבאים:

- humanoid dancing to music sounds

<https://crml.eelabs.technion.ac.il/projects/humenoid-dancing-to-music-sounds/>

- developing a framework using MATLAB to control BIOLOID robot

<https://crml.eelabs.technion.ac.il/projects/developing-a-framework-using-matlab-to-control-bioloid-robot/>

בפרויקט זה יצרנו פקודה קולית שהתקבלה מממשק Google Homen והמרנו אותה לתנועה תואמת של הרובוט דרך סביבת העבודה של ה-MATLAB. התבססנו על תנועות בסיסיות קיימות של הרובוט מפרויקטים קודמים ובנוסף יצרנו תנועות חדשות ומגוונות בעצמנו.

המסקנה העיקרית של פרויקט זה היא שניתן להמיר פקודות קוליות בצורה טובה ובשליטה מלאה ללא מגע כאשר סביבת העבודה ב-MATLAB היא נוחה וידידותית למשתמש. שליטה על הרובוט הינה דוגמה לאפשרות יישום הלכה למעשה של הטכנולוגיה ביישומים שונים ובתחומים מגוונים.

## Abstract

Our main goal was to use voice commands, which would be received from the Google Home interface, and use these voice commands to command a BIOLOID robot's movements, using the MATLAB development environment.

We also put a goal for ourselves, to get a hands-on learning experience for integrating multiple systems, using MATLAB, as it is a widely used tool for this use case.

We used past work, done in the lab, to perform basic CNC (command and control) on the BIOLOID robot, using the MATLAB environment.

Those projects can be found in the lab's website:

humanoid dancing to music sounds -

<https://crml.eelabs.technion.ac.il/projects/humenoid-dancing-to-music-sounds/>

developing a framework using MATLAB to control BIOLOID robot -

<https://crml.eelabs.technion.ac.il/projects/developing-a-framework-using-matlab-to-control-bioloid-robot/>

In this project we successfully created a voice command, which was received from the Google Home interface. We then converted the command into a matching robot movement, through the MATLAB environment.

We based the robot's movements on the basic, existing ones, from the previous projects, and additionally, we created new and diverse set of movements by ourselves.

Our main conclusion from this project is that we can use voice commands to command a BIOLOID robot, in a good manner, with full control, without the need for additional inputs. Furthermore, we feel like the experience we got with the MATLAB environment, allows us to say that it is very easy to use, and the perfect tool for this use case.

We would suggest for future work to be done to add support for different robots, and a visual chat representation, for the received voice commands.

## 2. מבוא

במסגרת הפרויקט נעזרנו בשני ממשקים עיקריים - רובוט ה-BIOLOID ובממשק Google Home. מהמסגרת Google Home נתנו פיזית את הפקודות הקוליות לרובוט. פקודות אלו, שהתקבלו בצורה קולית, הומרו לווקטור מידע בMATLAB ע"י אפליקציות חיצוניות (שיפורטו בהמשך). את הרובוט חיברנו גם כן לסביבת העבודה בMATLAB ובכך יכולנו לשלוט בתנועותיו ע"י מתן פקודה קולית.

### רובוט ה-BIOLOID:

בפרויקט זה נתנו פקודות קוליות לרובוט מסוג BIOLOID של חברת Robotis.

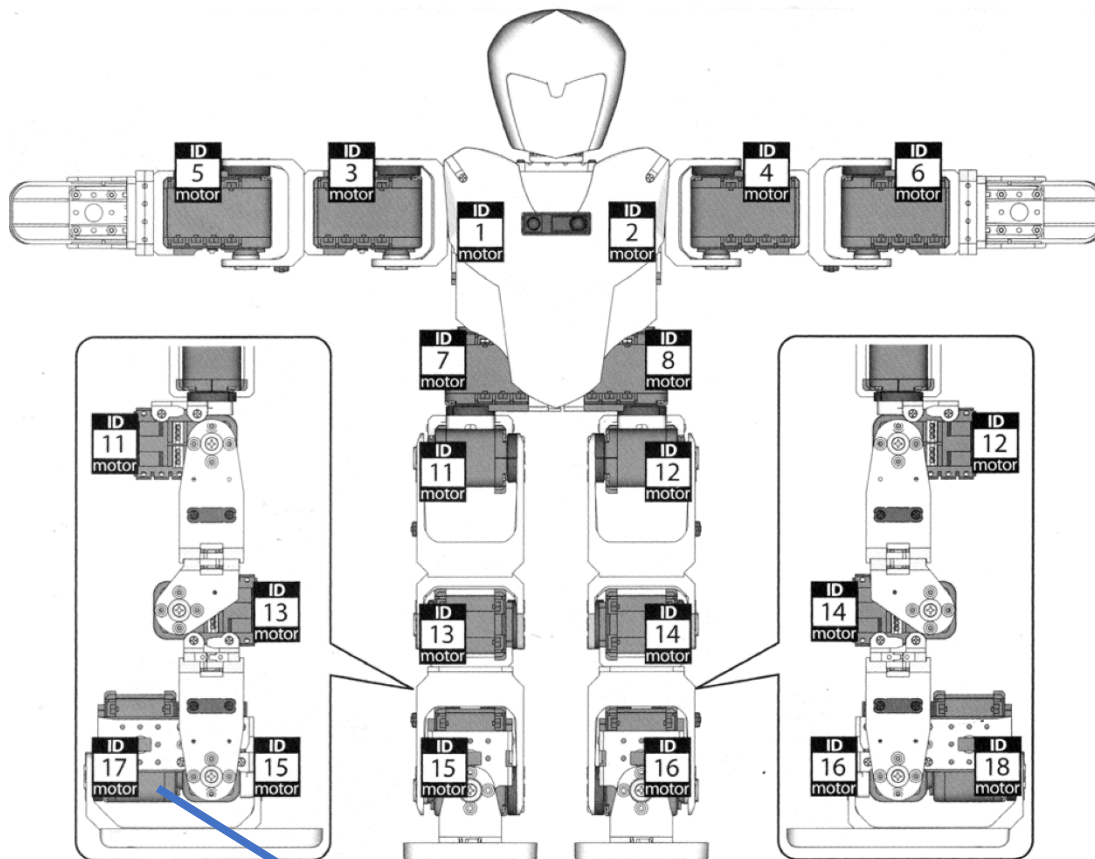
ברובוט ישנו בקר מסוג CM-530 ו-18 מנועים מסוג AX-12+.

בנוסף, קיימים בו חיישנים מסוג IR ו-GYRO.



איור 1

ברובוט קיימים 18 מנועים ומספרם כלהלן:

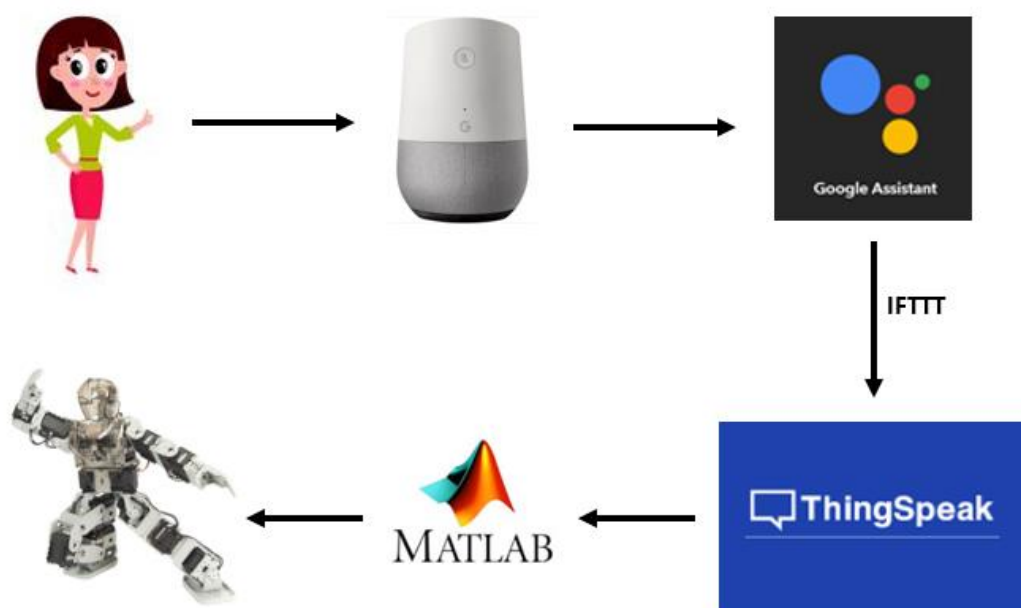


איור 2

לרובוט יש 18 דרגות חופש אשר מתבטאות ע"י 18 מנועים כאשר 6 מהן בידיים ו-12 ברגליים. לכל מנוע קיים מספר ID ובעזרת מספר זה הבקר יכול לתקשר עם כל מנוע בנפרד.

### ממשק ה-Google Home:

Google Home הוא מכשיר חכם שפותח ע"י גוגל. הוא מאפשר למשתמש לומר פקודות קוליות כדי לתקשר עם שירותים דרך Google Assistant – תוכנת עזר אישית למשתמש גוגל. השתמשנו ביכולות אלה של Google Home חיברנו לMATLAB באמצעות IFTTT וThingSpeak אשר בעזרתם יצרנו פקודה קולית שתורגמה לווקטור מידע בMATLAB.



איור 3

העבודה מורכבת ממספר שלבים כדלהלן:

יצרנו פקודות קוליות חדשות לממשק Google Home דרך Google Assistant בעזרת הIFTTT.

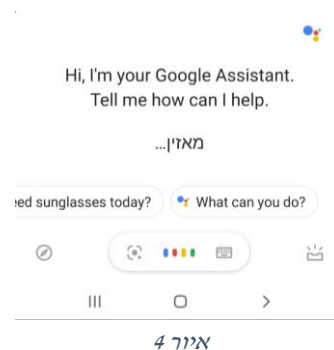
לאחר מכן, הפעלנו טריגר כך שכאשר מתקבלת הפקודה הקולית שהכנסנו נקבל שינוי בערך הווקטור שמייצג את המידע שנוצר מהפקודה הקולית המחובר לMATLAB. העברנו שליטה על הרובוט מתכנת RoboPlus אל MATLAB. בMATLAB ביצענו את החיבור בין שני הממשקים כך שהרובוט יגיב לפקודות הקוליות.



### 3. תיאור כללי

#### חיבור ממשק Google Homen אל הMATLAB

- ראשית, מחברים את מכשיר הGoogle Homen לחשמל. מדליקים אותו על ידי לחיצה בחלקו העליון.
- במקביל, מתחברים מהטלפון הנייד לרשת Wi-Fi ומורידים את אפליקציית Google Home ומתחברים אליה עם משתמש Google קיים.
- מקשרים את הטלפון הנייד למכשיר הGoogle Homen עצמו – נכנסים לאפליקציה שהורדנו, לוחצים על כפתור ה-'+' < set up devices < set up new devices in your home וממשיכים לעקוב אחר השלבים.
- מדליקים את הBluetooth בטלפון הנייד ומתחברים למכשיר הGoogle Homen (ניתן לתת פקודה קולית לGoogle Home על מנת לחבר את הBluetooth – *"Hey Google, connect to Bluetooth"*).
- לאחר שמיעת צליל ההתחברות, פותחים את אפליקציית Google Assistant.



לביצוע תהליך זה ניתן להיעזר בקישורים הבאים -

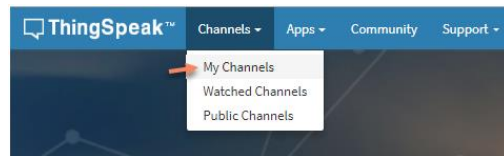
<https://support.google.com/assistant/answer/7538816?co=GENIE.Platform%3DAndroid&hl=en>

<https://support.google.com/googlenest/answer/7029485?co=GENIE.Platform%3DAndroid&hl=en>

כאן הושלמה התחברות משתמש הGoogle Home לGoogle Assistant על ידי Google Assistant.

על מנת ליצור פקודה קולית חדשה, נבצע את השלבים הבאים :

- יש ליצור חשבון ThingSpeak חדש בקישור הבא - <https://thingspeak.com/login> (ניתן להשתמש בחשבון MATLAB אם קיים).
- לאחר הכניסה ל-ThingSpeak נכנסים ל-My Channels < Channels.



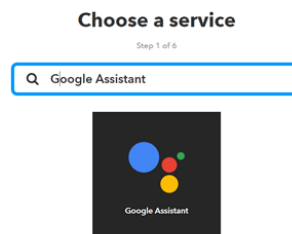
איור 5

- פותחים ערוץ חדש, בכל שדה ממלאים את שם הפקודה הרצויה.
- לוחצים Save Channel.
- כעת יש לפתוח חשבון משתמש חדש ב-IFTTT בקישור הבא - <https://ifttt.com/>.
- לאחר כניסה לחשבון שפתחנו, נבחר באופציית New Applet < My Applets.
- ניצור טריגר חדש, נלחץ על +this.

## If This Then That

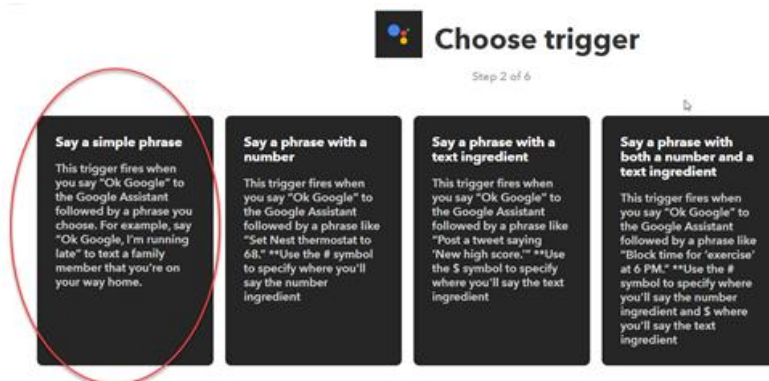
איור 6

- בוחרים בחיפוש Google Assistant כמכשיר הטריגר.



איור 7

- בוחרים Say a simple phrase.



איור 8

- נמלא את השדות המתאימים לפי מה שרוצים ויוצרים את הטריגר.

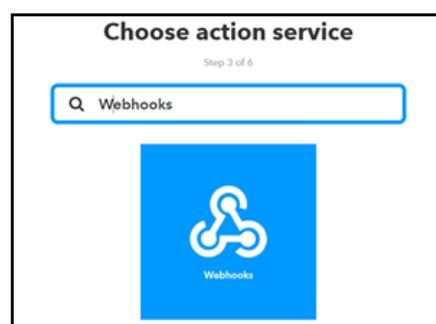
איור 9

- כדי להמשיך נלחץ על that+.



איור 10

- בחלונת החיפוש נרשום Webhooks.



איור 11

- נגיע למסך הבא :

איור 12

ונשלים את השדות בצורה הבאה :

את שדה ה URL ניקח מה Thingspeak מתוך ה API Keys שב Channels. מעתיקים את הכתובת שנמצאת ב"Write a Channel Feed".

איור 13

את כתובת URL נשנה בהתאם בצורה הבאה-

`https://api.thingspeak.com/update?api_key=38F9J3Z476N5JZVK&field1=0`  
במקום מספר השדה שמופיע אחרי המילה `field` (בדוגמה זהו המספר 1) נכתוב את מספר השדה המתאים עבור הפקודה הרצויה שהכנסנו בהתאם לשדות שהזנו בThingSpeak בתחילת התהליך.  
את הספרה '0' שנמצאת בסוף נשנה ל'1'.  
בשדה `Method` נסמן `GET`.  
בשדה `Content Type` נסמן `application/x-www-form-urlencoded`.

לביצוע תהליך זה ניתן להיעזר בקישור הבא-

<https://www.mathworks.com/help/thingspeak/google-iftt-thingspeak-lamp.html>

כעת סיימנו ליצור את הטריגרים וניתן לייבא אותם לMATLAB באמצעות הפונקציה הבאה:

## thingSpeakRead

Read data stored in a ThingSpeak channel

### Syntax

```
data = thingSpeakRead(channelID)
```

איור 14

<https://www.mathworks.com/help/thingspeak/thingspeakread.html#d120e18862>

כאשר את הchannelID נקח מהThingSpeak:

The screenshot shows the ThingSpeak website interface. At the top is a blue navigation bar with the ThingSpeak logo and links for Channels, Apps, and Support. Below this, the channel name 'Gili' is displayed. A red box highlights the 'Channel ID: 1048823'. Other details shown include 'Author: ovedgili' and 'Access: Public'. On the right side, there is a link that says 'if its works'. At the bottom, there is a row of buttons: 'Private View', 'Public View', 'Channel Settings', 'Sharing', 'API Keys', and 'Data Import / Export'.

איור 15

הנתונים `data` הינו ווקטור מידע שיכיל את השינויים בפקודות הקוליות ואיתו נעבוד בMATLAB.

\*בפרויקט זה הגדרנו 8 פקודות. על מנת להוסיף עוד פקודות יש לפתוח channel חדש או לחלופין לשנות את הפקודות עצמן. שינוי הפקודות ייעשה על ידי שינוי הקוד בMATLAB ושינוי הגדרת הפקודה הקולית בIFTTT בהתאם.

## העברת השליטה מסביבת RoboPlus אל סביבת MATLAB

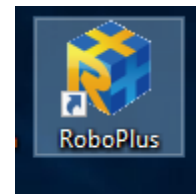
לשם תכנות הרובוט בסביבת ה-MATLAB יש צורך להתקין את התוכנה הבסיסית של הרובוט RoboPlus.

קישור להורדה והתקנה של התוכנה נמצא ב :

<https://www.robotis.co.uk/software/roboplus-1-0>

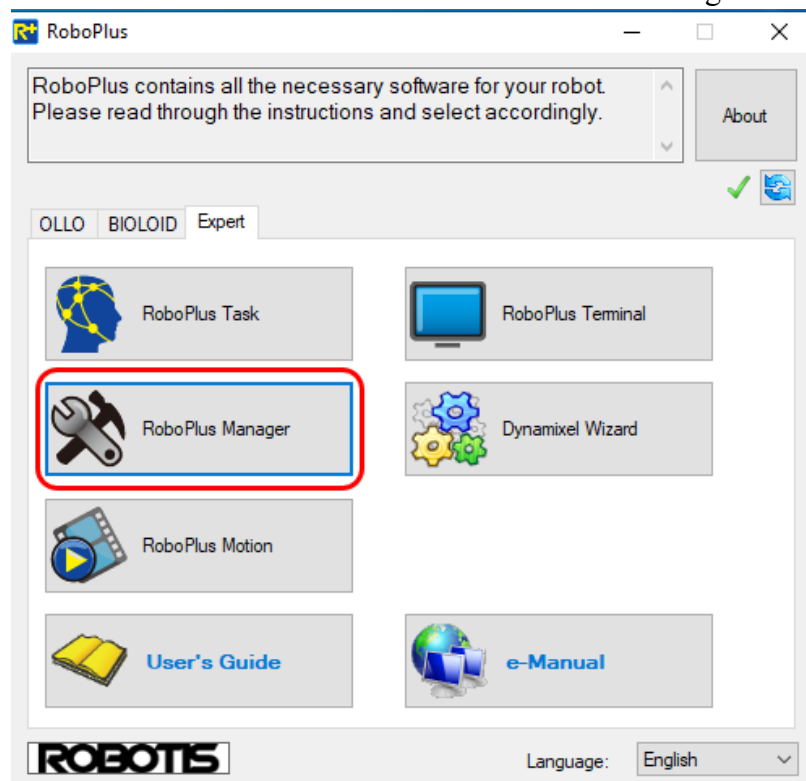
אחרי התקנת התוכנה במידה והיא לא הייתה מותקנת יש לבצע את השלבים הבאים :

1. חיבור הרובוט למחשב דרך USB וכניסה לתוכנה ע"י לחיצה על קיצור הדרך RoboPlus שעל שולחן העבודה.



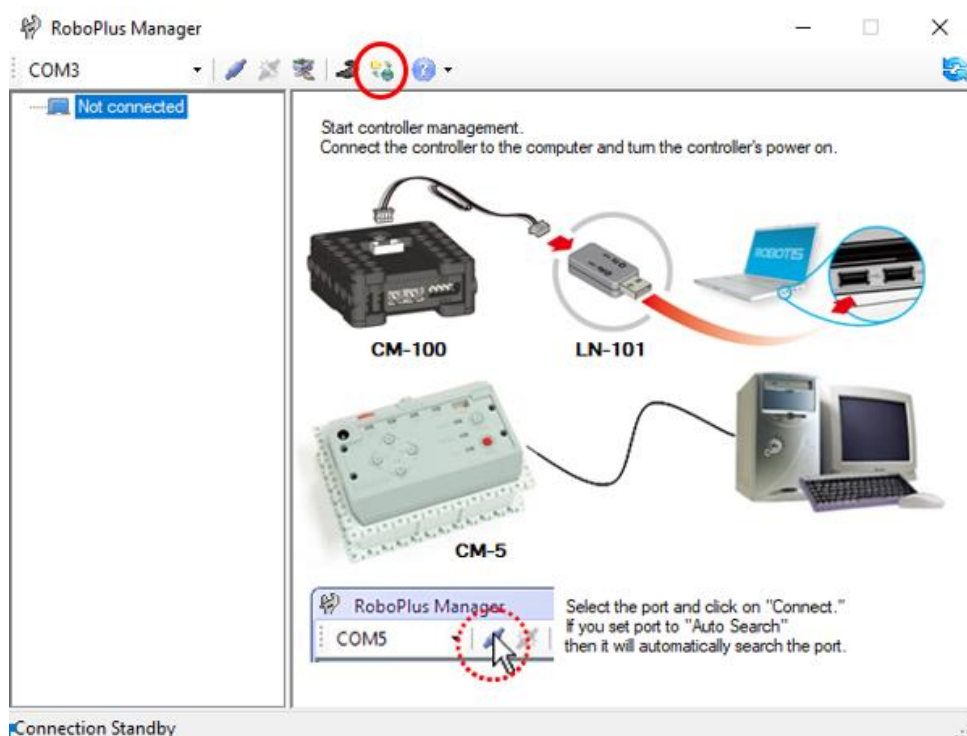
איור 16

2. נכנס לRoboPlus Manager.



איור 17

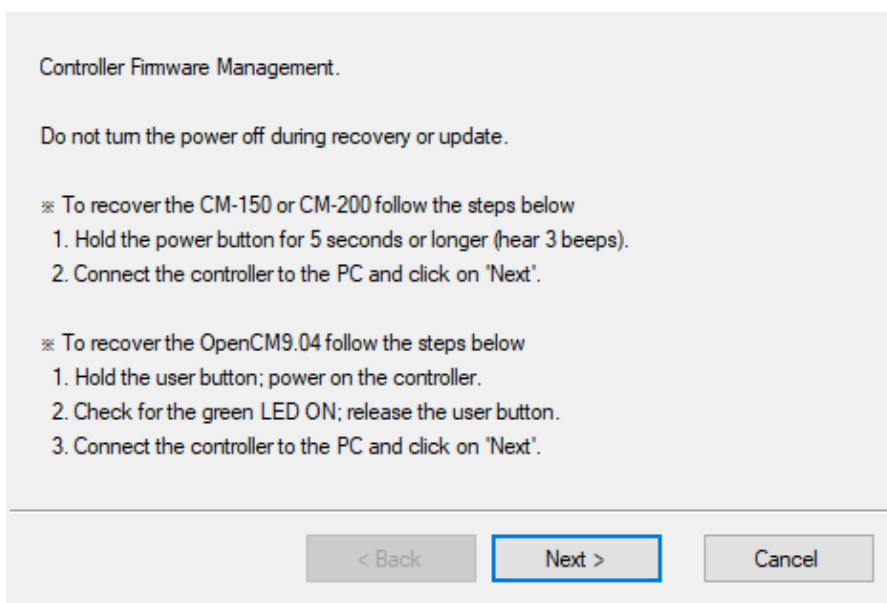
3. נלחץ על הכפתור המסומן על מנת להתחבר לCOM שהרובוט מחובר אליו.



איור 18

4. בחלון שנפתח נלחץ על Next.

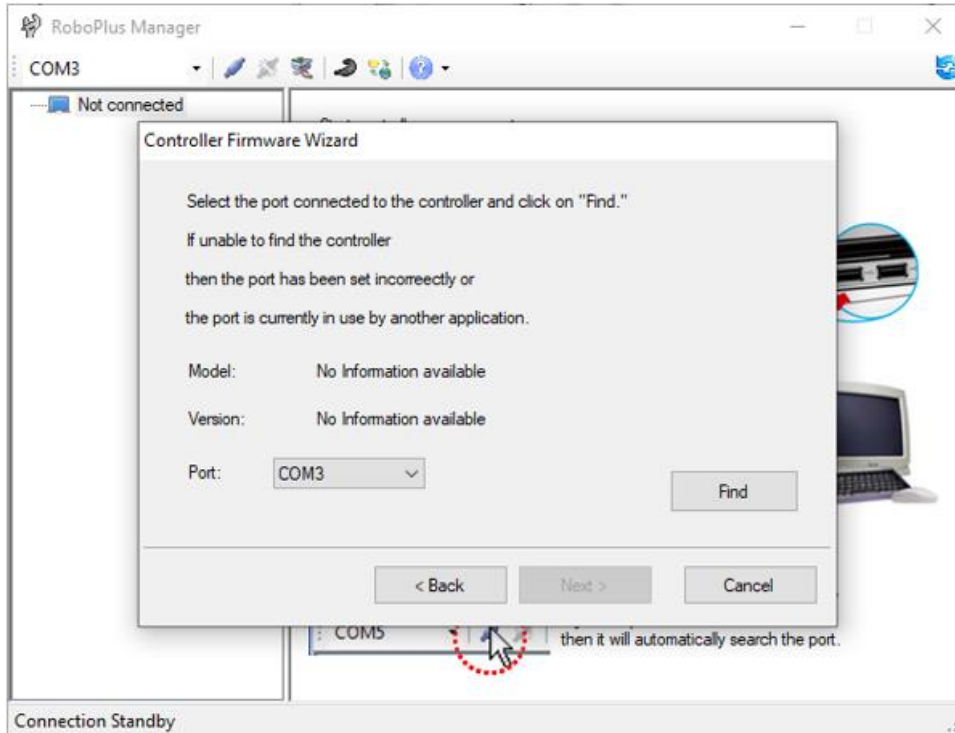
#### Controller Firmware Wizard



איור 19

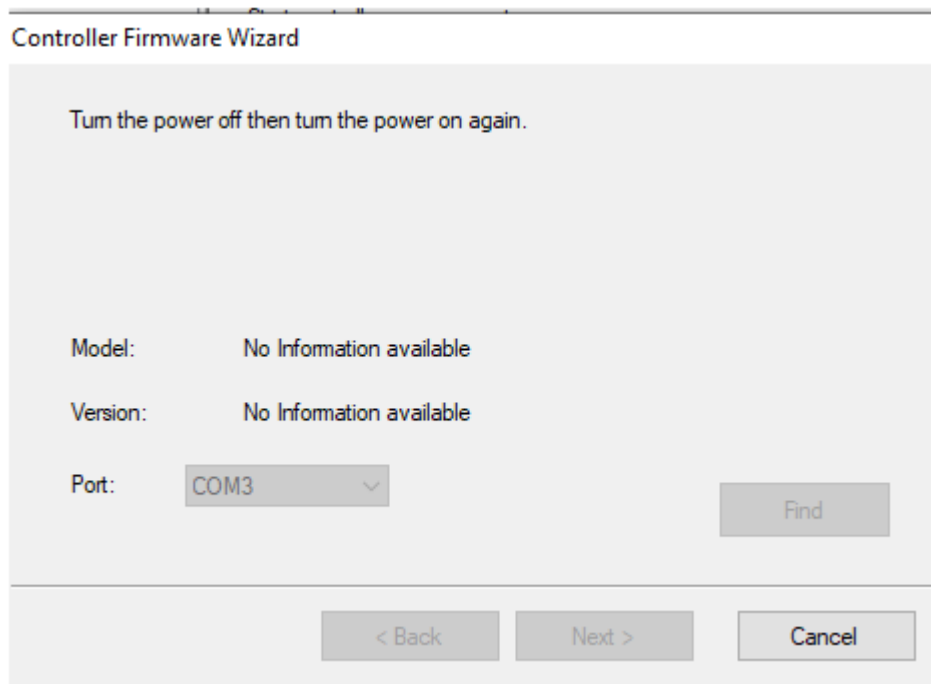


5. בחלון הבא נבחר את Port המתאים אליו מחובר הרובוט ונלחץ על Find.



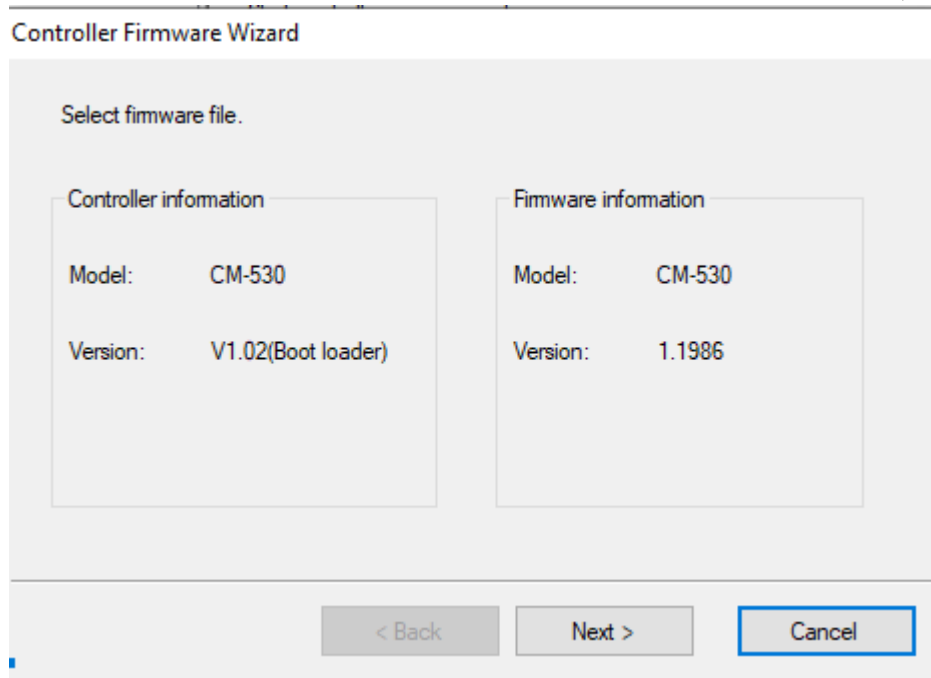
איור 20

6. נכבה ונדליק את הרובוט ולאחר מכן נלחץ על Next.



איור 21

7. נלחץ על Next.



Controller Firmware Wizard

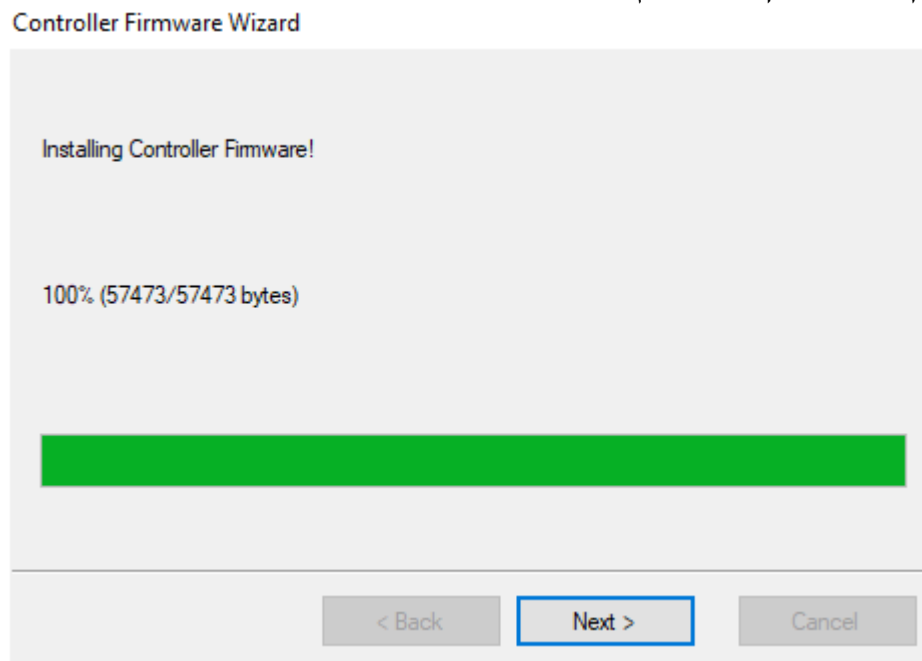
Select fimware file.

Controller information		Firmware information	
Model:	CM-530	Model:	CM-530
Version:	V1.02(Boot loader)	Version:	1.1986

< Back   Next >   Cancel

איור 22

8. נקבל את המסך הבא ונלחץ על Next כאשר הטעינה תסתיים.



Controller Firmware Wizard

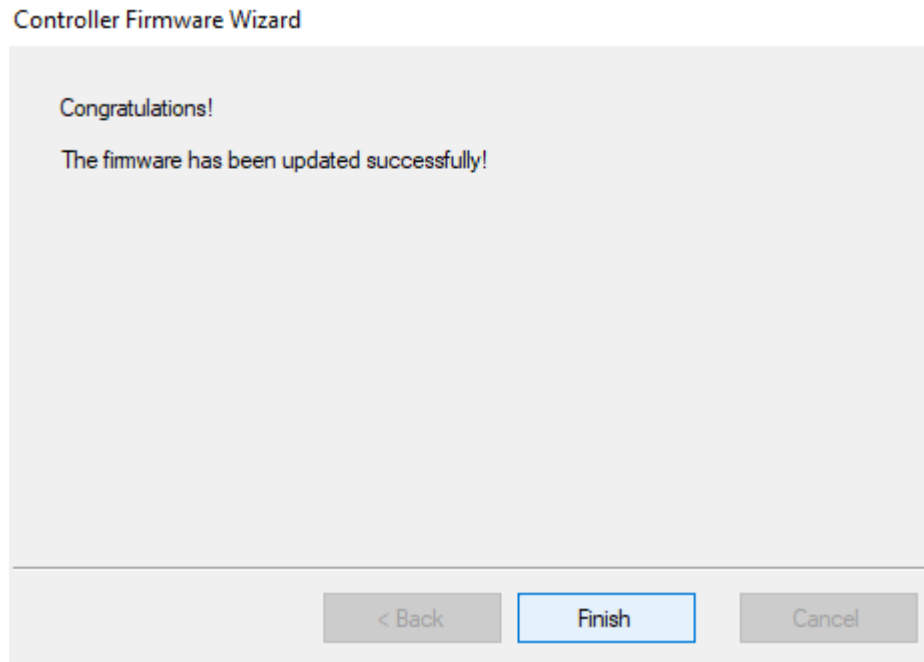
Installing Controller Firmware!

100% (57473/57473 bytes)

< Back   Next >   Cancel

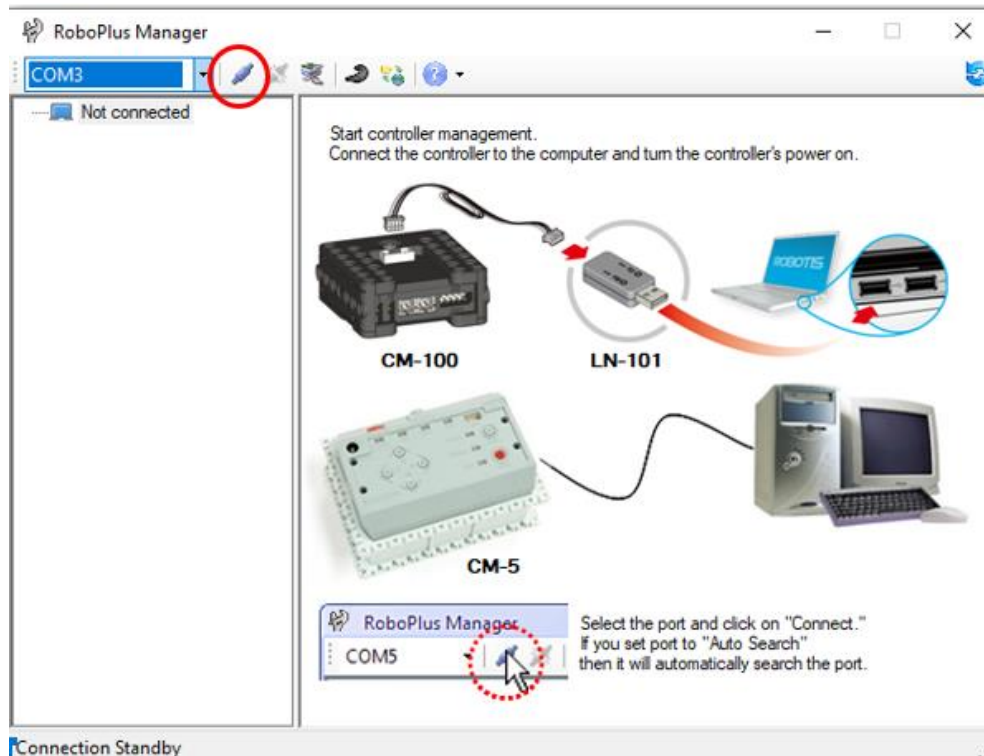
איור 23

9. בסיום הפעולה נקבל את המסך הבא :



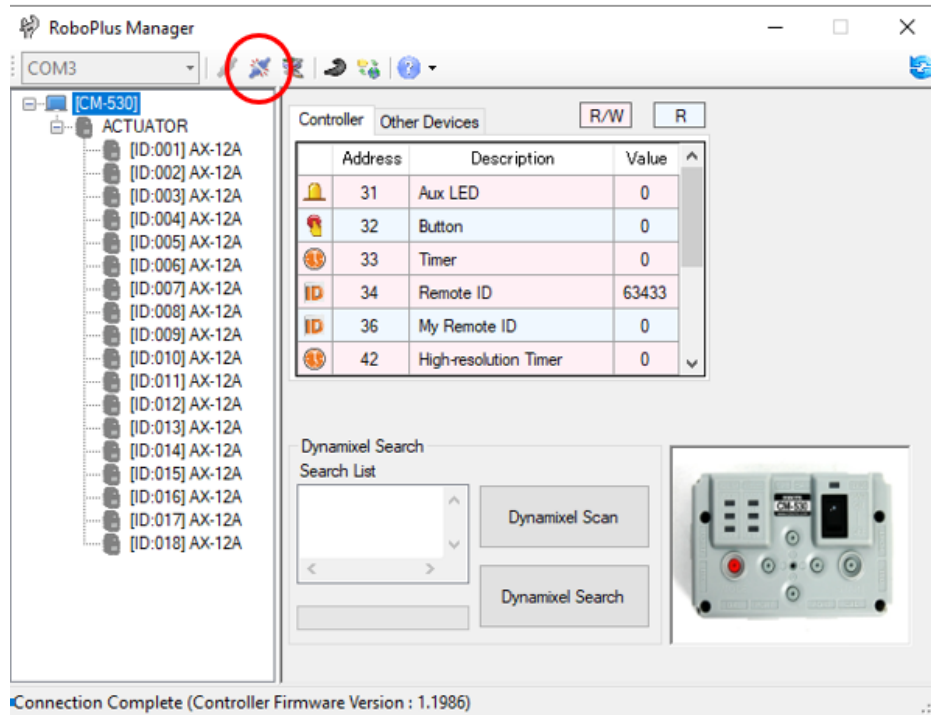
איור 24

10. נבחר את Port המתאים לרובוט ונתחבר בעזרת לחיצה על כפתור connect.



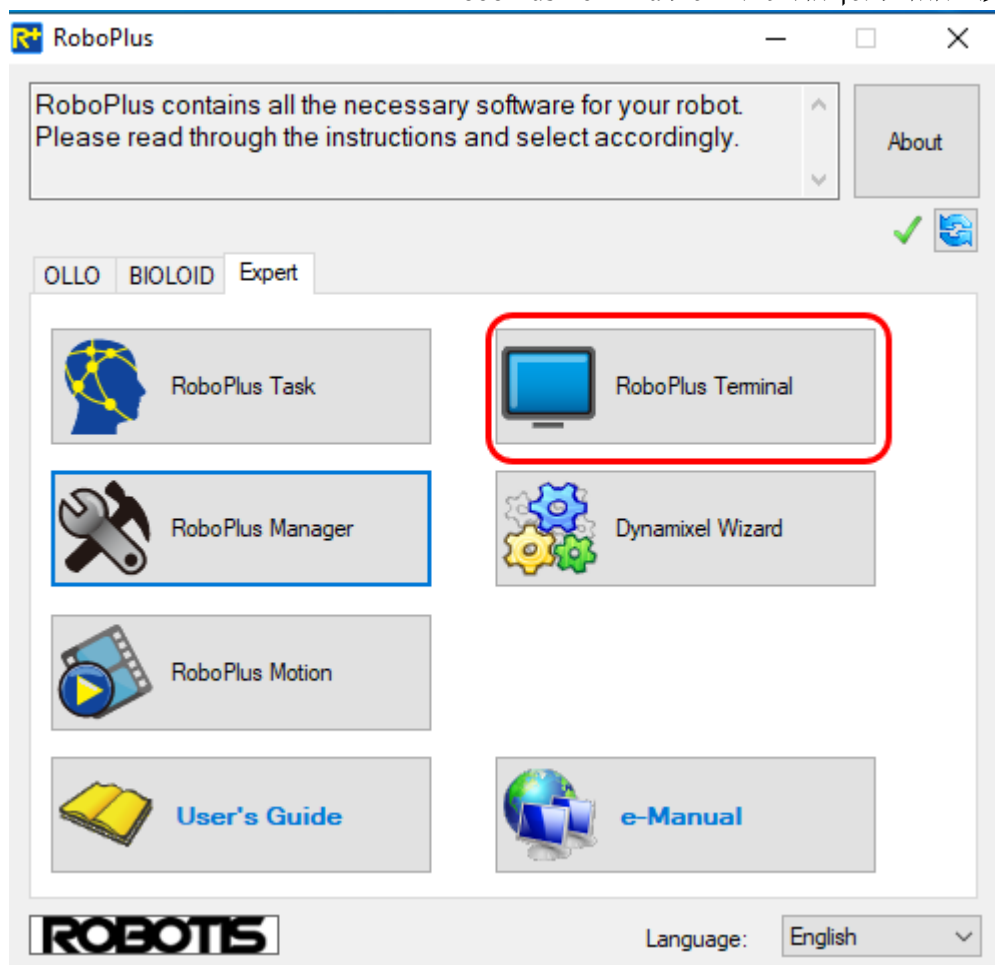
איור 25

11. נקבל את המסך הבא ונלחץ על כפתור ה disconnect המסומן.



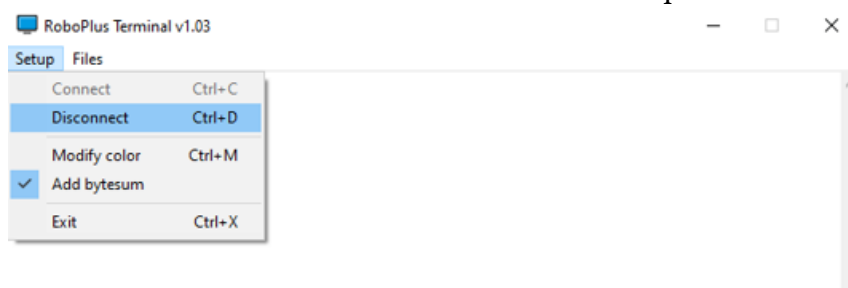
איור 26

12. כעת נחזור למסך הראשי ונכנס לRoboPlus Terminal.



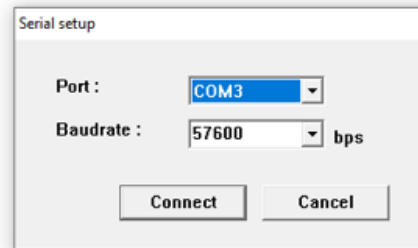
איור 27

13. בטרמינל נכנס לsetup < Disconnect.



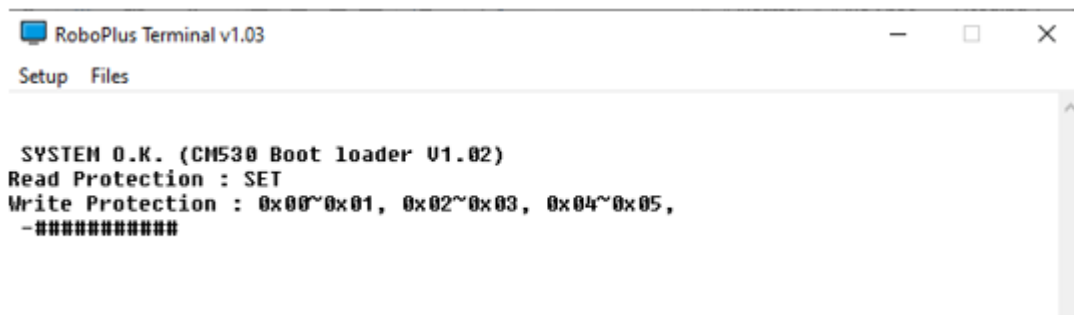
איור 28

14. נכנס שוב לsetup < connect. נקבל את המסך הבא ובו נבחר את Port המתאים. נלחץ על Connect.



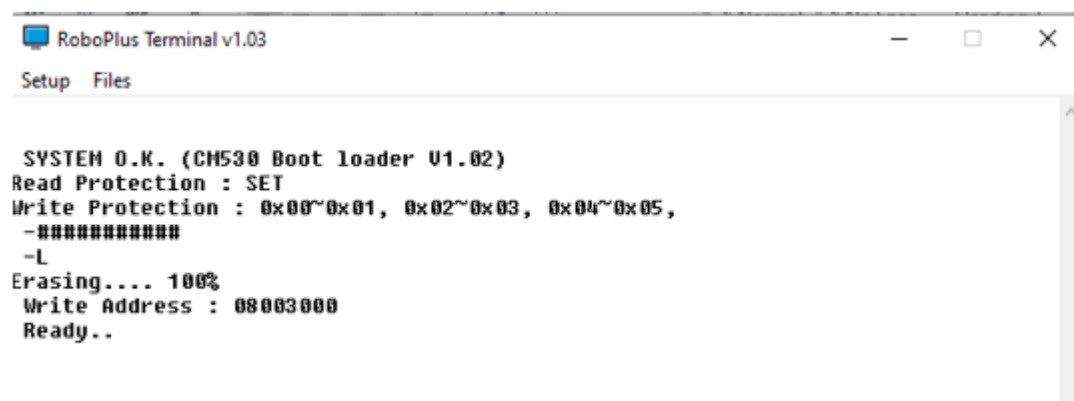
איור 29

15. נלחץ במקלדת על # (Shift + 3) עד לקבלת המסך הבא(לא ללחוץ קצר מידי אך גם לא ארוך מידי):



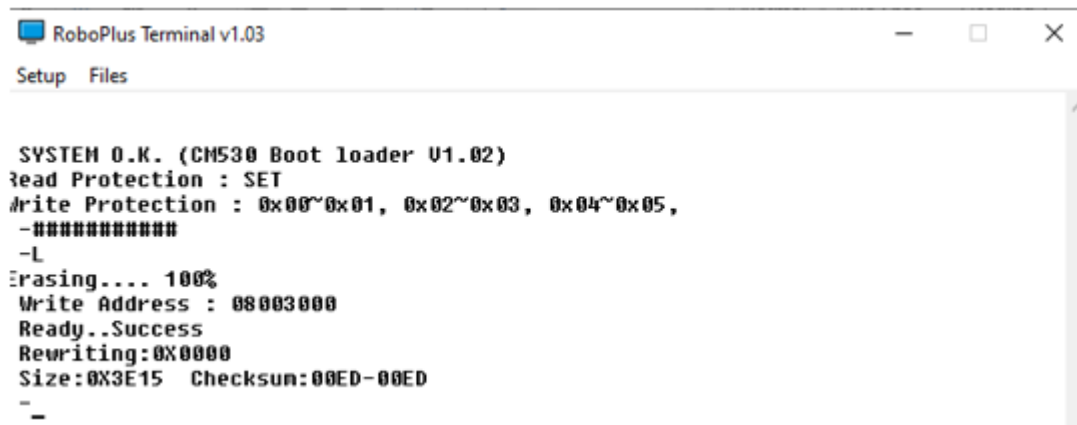
איור 30

16. כעת יש ללחוץ במקלדת על Enter וישר לאחר מכן על L(חשוב לשים לב שהאות תהיה גדולה) ואז שוב Enter לקבלת המסך הבא :



איור 31

17. נכנס לFiles < Transmit File < נבחר את הקובץ CM530.hex המצורף ונקבל את המסך הבא:



```
RoboPlus Terminal v1.03
Setup Files

SYSTEM O.K. (CM530 Boot loader V1.02)
Read Protection : SET
Write Protection : 0x00~0x01, 0x02~0x03, 0x04~0x05,
-#####
-L
Erasing.... 100%
Write Address : 08003000
Ready..Success
Rewriting:0X0000
Size:0X3E15 Checksum:00ED-00ED
-
```

איור 32

18. לאחר השלמת הפעולות הנ"ל, יש להתנתק שוב מהטרמינל (בעזרת Setup < Disconnect) ואז לנתק ולהדליק את הרובוט (באמצעות הכפתור האחורי שעליו).

## הסבר על הקוד וחיבור הממשקים

הקוד הראשי אשר מהווה בסיס לשליטה על הרובוט:

\* יש לשים לב כי שורת הקוד השנייה המכילה את "fclose(instrfind)" צריכה להיות בהערה עבור הריצה הראשונה.

```
clc;
fclose(instrfind); % close the last port that was open
s=setSerialPort(3);%set s as PORT 3
fopen(s); % open the port
walk_ready(s); % steady the robot
t = timer('TimerFcn', 'stat=false; disp(''Timer!'')',...
          'StartDelay',1500); %set timer for live running
start(t); %set timer for live running
stat=true; %set timer for live running

[y, Fs] = audioread('Rock This Party.mp3'); %prepare the song for dance
motion
player = audioplayer(y, Fs);

while(stat==true) %set timer for live running
    fclose(instrfind); % close the last port that was open
    data = thingSpeakRead(1048823); % import the data from the speaker
    for i=1:8 % change NaN to 0
        if (data(i)~= 1)
            data(i)=0;
        end
    end

    if (data(5)) % stop command
        pause(player);
        s=setSerialPort(3);%set s as PORT 3
        fopen(s);
        walk_ready(s); % steady the robot
    end

    if (data(1)) % King Kong commsnd
        s=setSerialPort(3);%set s as PORT 3
        fopen(s);
        Balance(s);
        KingKongMotion();
        walk_ready(s); % steady the robot
    end

    if (data(2)) % Dance command
        play(player);
        s=setSerialPort(3);%set s as PORT 3
        fopen(s);
        DanceMotion(s);
        pause(player);
        % walk_ready(s); % steady the robot
    end

    if (data(3)) % walk command
        s=setSerialPort(3);%set s as PORT 3
        fopen(s);
        walk_forward(s);
        walk_ready(s); % steady the robot
    end
end
```



```

if (data(4)) % clap command
    s=setSerialPort(3);%set s as PORT 3
    fopen(s);
    clap_motion();
    walk_ready(s); % steady the robot
end

if (data(6)) % Guitar command
    s=setSerialPort(3);%set s as PORT 3
    fopen(s);
    guitar_motion();
    walk_ready(s); % steady the robot
end

if (data(7)) % Thinking command
    s=setSerialPort(3);%set s as PORT 3
    fopen(s);
    ThinkingMotion();
    walk_ready(s); % steady the robot
end

if (data(8)) % Cheer command
    s=setSerialPort(3);%set s as PORT 3
    fopen(s);
    Cheer();
    walk_ready(s); % steady the robot
end

pause(1)
end

```

בתחילת הקוד אנחנו מגדירים את הPORT המתאים לרובוט ומייצבים אותו. לאחר מכן, הגדרנו טיימר לזמן ריצה רצוי(אותו אפשר לשנות) כך שהרובוט ירוץ ב"לייב" בעזרת לולאת while. הווקטור data זהו וקטור המידע המכיל את כל השינויים הקוליים שמגיעים לMATLAB בעזרת הפונקציה *"thingSpeakRead"* המוצגת לעיל. הווקטור מכיל ערך '1' עבור הפקודה האחרונה שהתקבלה וערך 'NaN' עבור שאר הפקודות. המרנו את הערך 'NaN' לערך '0' על מנת לפשט את העבודה.

חילקנו את הקוד לשמונה מקרים (מקרה עבור כל פקודה) כאשר כל פקודה תופעל באם ימצא הערך '1' במקום שלה בוווקטור (את מיקום הפקודה בוווקטור ניתן למצוא באתר של ThingSpeak - מיקום הפקודה בשדות הוא מיקומה בוווקטור).

ThingSpeak™ Channels Apps Support

Author: ovedgila Access: Public

Private View Public View Channel Settings Sharing API Keys

### Channel Settings

Percentage complete 50%

Channel ID 1048823

Name Gili

Description if its works

Field 1 King Kong ☒

Field 2 Dance ☒

Field 3 Walk ☒

Field 4 Clap ☒

Field 5 stop ☒

איור 33

בכל מקרה כזה, הפעלנו את הפונקציה הרלוונטית ולאחר מכן ייצבנו את הרובוט על ידי הפונקציה *"walk\_ready"*.

כעת נסתכל על פונקציה ספציפית. הפונקציה המוצגת מייצגת את תנועת ה'*King Kong*' :

```
vec = [
552 471 424 599 159 864 353 670 508 515 427 596 282 741 647 376 508 515
552 471 145 599 158 864 353 670 508 515 427 596 282 741 647 376 508 515
552 471 424 599 159 864 353 670 508 515 427 596 282 741 647 376 508 515
552 471 424 898 159 865 353 670 508 515 427 596 282 741 647 376 508 515
552 471 424 599 159 864 353 670 508 515 427 596 282 741 647 376 508 515
];
for k=1:4
    for i=1:5
        for j = 1:18
            setMovingSpeed(s,j,330);
            setGoalPosition(s,j,vec(i,j));
        end
        pause(0.20);
    end
end
```

כל שורה במטריצה מייצגת מיקום של צעד ספציפי וכל טור מייצג מנוע ספציפי, כאשר הטור הראשון מייצג את מנוע מספר 1 וכך הלאה.

מבצעים מעבר על כל המטריצה כך שעבור כל צעד נעבור על כל המנועים ונמקם את הרובוט בקואורדינטות הרצויות על ידי הפונקציה *"setGoalPosition"* ואת חוזק תזוזת המנוע נקבע על ידי הפונקציה *"setMovingSpeed"*.



#### 4. תוצאות

הצלחנו במשימתנו! הרובוט אכן הגיב לפקודות קוליות וביצע תנועות בהתאם.

במהלך העבודה נתקלנו במספר קשיים אותם לא הצלחנו לפתור:

- הרובוט אינו יציב ולכן תנועות מסובכות המשלבות את תזוזת פלג גופו התחתון לא היו אפשרויות כיוון שהרובוט נפל כאשר ביצע אותן (המרת התנועות מRoboPlus לMATLAB אינה אידיאלית ודרוש בה שיפור).
- מספר התנועות אותן יכולנו לבצע היה מוגבל כיוון שהצלחנו להשתמש בChannel יחיד בלבד מהThingSpeak (הוספת Channels נוספים גוררת בעיות בקוד שקשה להתגבר עליהן).

למרות הקשיים עמדנו בהצלחה בהגדרות ומטרות הפרויקט.

## 5. סיכום ומסקנות

על פי תוצאות הפרויקט, נמליץ את הבאים :

- ממליצות להשתמש בסוללה בכדי לשמור על איזון טוב יותר של הרובוט(ולא שימוש בחיבור ישיר לחשמל שמגביל את תנועת הרובוט)
- נמליץ לכבות ולהדליק את הרובוט בכל פעם משהו "נתקע" אתחול שכזה עוזר ופותר בעיות.
- לפרויקט המשך נציע את השיפורים הבאים :
  - ☐ הפסקת פעולת הרובוט תוך כדי פעולתו ולא לאחר סיומה.
  - ☐ עבודה עם רובוט שונה בצורתו על מנת שיהיה יציב יותר וכך יהיה אפשר לבצע בו יותר פעולות.
  - ☐ להוסיף ממשק נוסף – תצוגה גרפית בה ניתן להציג את הצי'אט הנעשה עם הרובוט.
  - ☐ אופציה לכמות פקודות קוליות גדולה יותר.

## 6. נספחים

הפקודות הקוליות אותן יש לומר על מנת להפעיל את התנועות :

1. תנועת קינג קונג –

• King kong

• Monkey

• Rap chest

2. תנועת ריקוד –

• Dance

• Please dance

• Lats dance

3. תנועת הליכה –

• Walk

• Come

• Walk forward

4. תנועת מחיאות כפיים –

• Clap

• Clap your hands

• Applause

5. עצירה –

• Stop

6. תנועת ניגון בגיטרה –

• Please play the guitar

• Guitar

7. תנועת חשיבה/ גירוד בראש –















• Think

• Scratch head

8. תנועת עידוד –

• Cheer

כל התנועות אשר הוספנו למאגר התנועות שלנו(לא כולן בשימוש בקוד) הן :

Blocking   
Cheer   
clap\_motion   
DanceMotion   
drums\_motion   
guitar\_motion   
HafIaMotion   
HelloMotion   
KingKongMotion   
Pray   
PushUpsMotion   
SitMotion   
ThinkingMotion   
walk\_forward 

איור 35

## 7. רשימת מקורות

- ThingSpeak

<https://thingspeak.com/>

- ThingSpeak הוספת ערוץ ב

<https://www.mathworks.com/help/thingspeak/collect-data-in-a-new-channel.html#bu4b1yc>

- ThingSpeak חיבור ל Google Assistant

<https://www.mathworks.com/help/thingspeak/google-iftt-thingspeak-lamp.html>

- Matlab ThingSpeak פונקציות המקשרות בין

<https://www.mathworks.com/help/thingspeak/thingspeakread.html#d120e18862>

- IFTTT

<https://ifttt.com/>