

esp32 ומודול זיהוי קול Elechouse v3

1. מבוא

במאמר זה נחבר אל מיקרו בקר ESP32 מודול זיהוי קול Elechouse V3 .

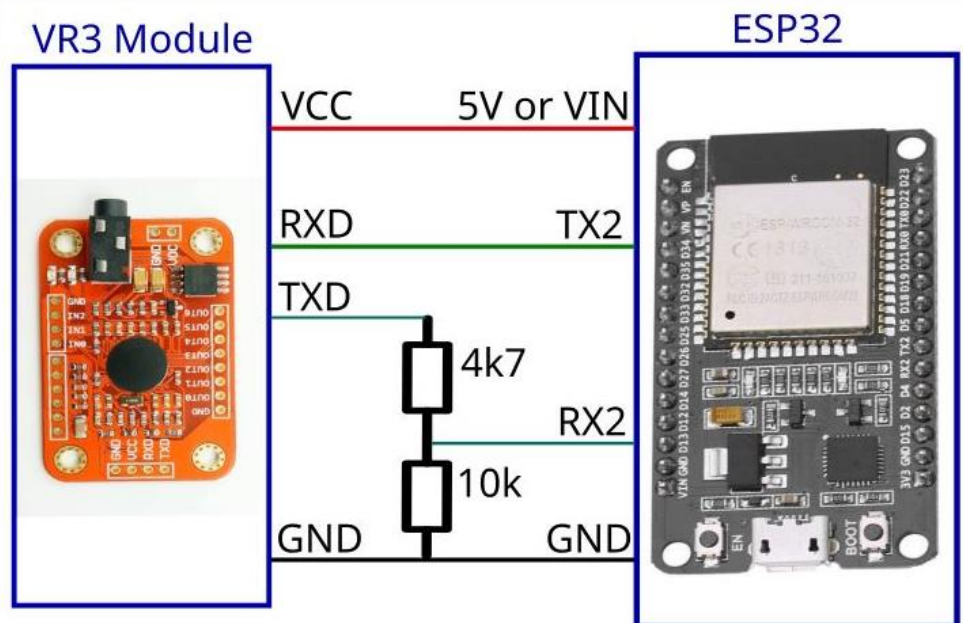
על מודול זיהוי קול Elechouse V3 בהרחבה כולל עבודה עם ארדואינו ניתן למצוא בקישור :

<https://www.arikporat.com/wp-content/uploads/2022/12/voiceRecognition-v3.pdf>

2. חומרה

כמה שיקולים בעת חייוט מודול זיהוי הקול אל ESP32 :

- מודול זיהוי הקול Elechouse VR3 הוא מודול העובד עם 5 וולט, וההדקים של ה- ESP32 אינם עמידים במתח של 5 וולט ויכול לקרות להם נזק. לפיכך, יש להוריד את המתח היוצא מהדק השידור TXD של מודול זיהוי הקול למתח של 3.3 וולט שנכנס להדק RXD של ה- ESP32. ניתן להשיג זאת בעזרת שני נגדים. נבחר 2 נגדים של $4.7\text{ K}\Omega$ ו- $10\text{ K}\Omega$. כאשר בהדק TXD יש 5 וולט על נגד ה- $4.7\text{ K}\Omega$ ייפול $1/3$ ממתח ה- 5 וולט ועל נגד $10\text{ K}\Omega$ ייפול כ- $2/3$ ממתח ה- 5 וולט (כ- 3.3 וולט). מדוע לקחנו נגד של $4.7\text{ K}\Omega$ ולא לקחנו נגד של $5\text{ K}\Omega$? כי הוא קשה להשגה בחנויות רכיבים. $4.7\text{ K}\Omega$ הוא נגד נפוץ וניתן להשגה בקלות.
- ESP32 כולל שלושה פורטים טוריים (UART). הראשון הוא U0UXD והוא משמש עבור יציאת ה- USB הטורית כולל לתכנות (צריבת התוכנית בהורדה מהמחשב). השני, U1UXD, משמש בדרך כלל לחיבור לשבב פלאש SPI. מכאן שרק ה-UART השלישי, U2UXD, באמת בטוח לשימוש. מסיבה זו, הן דיאגרמת החיווט המוצגת להלן, והן התוכנה, מניחים שה- Elechouse VR3 מחובר ל-UART השלישי הנקרא U2UXD. החיבור נראה באיור הבא :



איור 1 : חיבור ESP32 ומודול זיהוי קול Elechouse VR3 .

מודול ה-ESP32 מקבל מתח מיציאת ה-USB שלו, ומספק זאת ל-VR3 Elechouse. בהתאם ללוח ה-ESP32 שבו נשתמש הדק ה-5 וולט בלוח ה-ESP32 עשוי להיקרא 5V או שהוא עשוי להיקרא גם VIN.

3. תוכנה

א.3 נשתמש בסביבת העבודה של הארדואינו. קישור המסביר כיצד להוריד את תוכנת ה-ESP32 לסביבת העבודה של הארדואינו ניתן למצוא בקישור לאתר בשפה העברית:

<https://www.arikporat.com/wp-content/uploads/2023/01/introduction-to-esp32.pdf>

או קישור לאתר בשפה האנגלית :

<https://randomnerdtutorials.com/installing-the-esp32-board-in-arduino-ide-windows-instructions/>

ב.3 כעת יש להתקין את ספריית זיהוי הקול V3. בצע את ההוראות מדף Github של Elechouse בקישור :

[GitHub - elechouse/VoiceRecognitionV3: Arduino library for elechouse Voice Recognition V3 module](https://github.com/elechouse/VoiceRecognitionV3)

ג.3 יצרנית המודול חברת Elechouse ספרייה עבור הארדואינו שאיננה עובדת באופן מלא עם ה-ESP32. יש להחליף 4 קבצים (לבצע overwrite). הקבצים נמצאים בקישור :

[ESP32-ElechouseVR3/Arduino Files at main · frankvanhooft/ESP32-ElechouseVR3 · GitHub](https://github.com/frankvanhooft/ESP32-ElechouseVR3)

ד.3 ארבעת הקבצים הם :

- VoiceRecognitionV3.cpp
- VoiceRecognitionV3.h
- vr_sample_train.ino
- vr_sample_bridge.ino

ה.3 שני הקבצים הראשונים VoiceRecognitionV3.cpp ו VoiceRecognitionV3.h מצורפים על ידי חברת

Elechouse וגם 2 הקבצים האחרונים עם הסיומת ino ניתנו גם הם על ידי Elechouse והם שימושיים מאוד.

1.3 השינוי שבוצע ב 2 הקבצים הראשונים הם מחיקת השורה : `#include <softwareSerial.h>` כי הספרייה הזו איננה

עובדת עם ה-ESP32. כמו כן עודכן הפורט הטורי לעבודה עם U2UXD (פורט תקשורת טורית UART2).

גם השידורים מה-ESP32 ל-VR3 הואטו מעט. השהיה קטנה הוכנסה בין בתים עוקבים בעת שליחת פקודות ל-VR3 – נראה שזה משפר את האמינות.

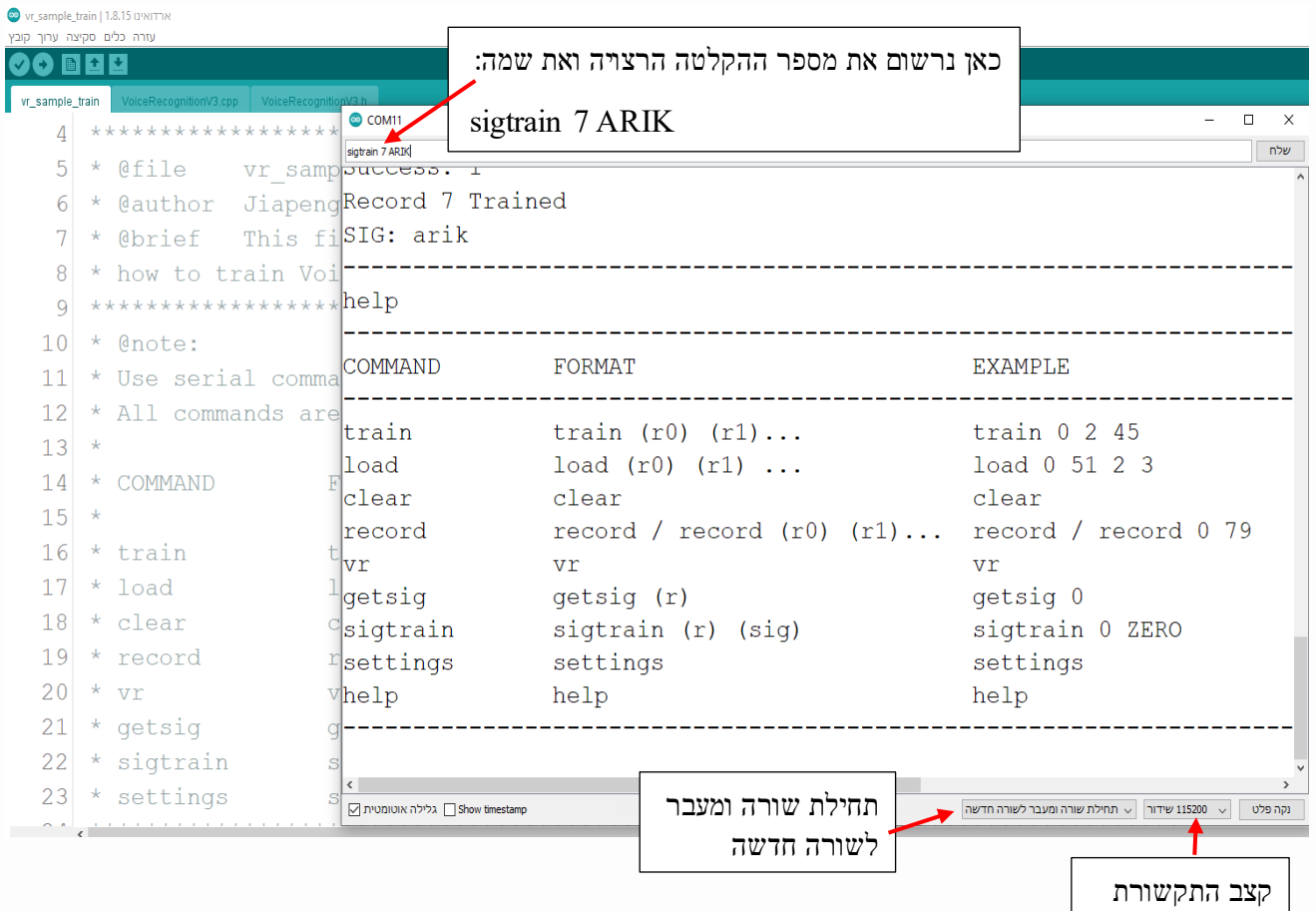
ו.3 ברירת המחדל של VR3 היא 9600 באוד, כך שכאשר קוראים לפונקציה `setup()` תמיד עוברים ב-9600 אלא אם כן קצב השידור של VR3 שונה במפורש.

3. לקבלת הוראות ומדוע להשתמש ב- vr_sample_train וב- vr_sample_bridge יש לעיין בתיעוד של Elechouse בקישור: [VR3_manual.pdf \(elechouse.com\)](http://VR3_manual.pdf(elechouse.com))

4. שימוש בתוכנת ההקלטות vr_sample_train.ino :

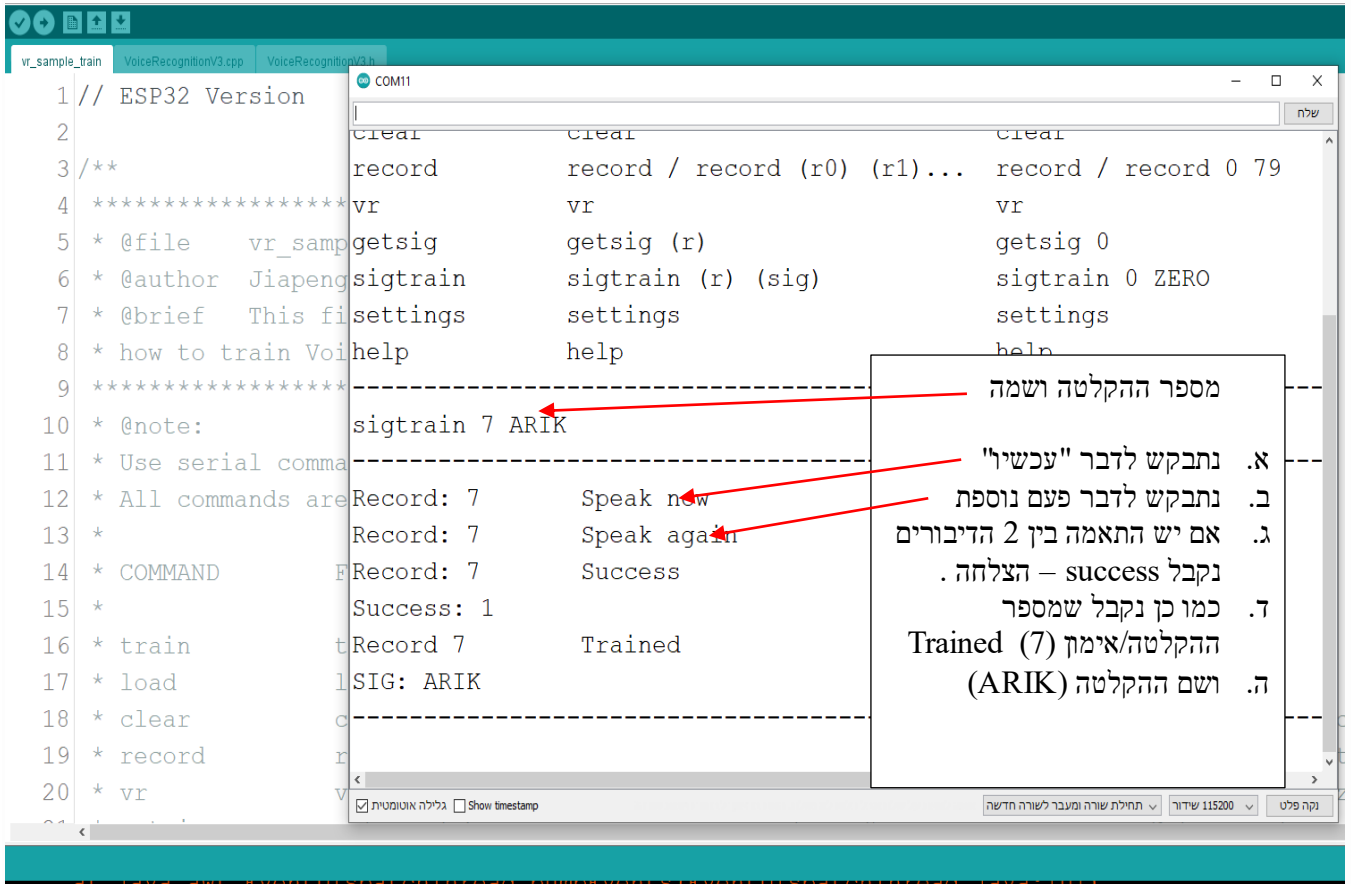
4.4 העלה/טען את התוכנית לערכת ה-ESP32. תוכנית זו מבצעת "אימון", כלומר מקליטה הודעות למודול. יש לפתוח את המוניטור הטורי. לשים לב שקצב התקשורת 115200 ו"תחילת שורה ומעבר לשורה חדשה", עבור כל אימון/הקלטת דיבור התוכנית תודיע לנו בעזרת המוניטור הטורי "speak now" – דבר עכשיו ואז נשמיע את ההודעה שנרצה (זמן ההודעה לא יותר מ-1.5 שניות). זמן קצר לאחר מכן נקבל הודעה "speak again" – דבר שוב ואז יש להשמיע אותה הודעה. אם 2 ההודעות זהות (כמעט..) נקבל success – הצלחה ו matched – בוצע תאום. אם לא נמצא תאום בין 2 ההודעות נקבל הודעה cannot matched – לא יכול לתאם ומיד לאחר מכן יתבצע ניסיון תאום נוסף של דיבור. דיבור חוזר וניסיון התאמה.

4.4 נרשום למעלה בצד שמאל את מספר ההקלטה הרצויה לנו עם שם/חתימת ההקלטה לדוגמה: sigtrain 7 ARIK. כלומר אימון/הודעה 7 ששמה/חתימתה ARIK. בסיום הכתיבה נלחץ על מקש Enter או "שלח" מצד ימין. הערה: יש לשים לב שבזמן ה"אימון" אין רעשי רקע כמו רעש מכוניות בכביש, מטוסים, דיבורי של אנשים או טלוויזיה עובדת וכו'. האיור הבא מתאר את הנאמר.



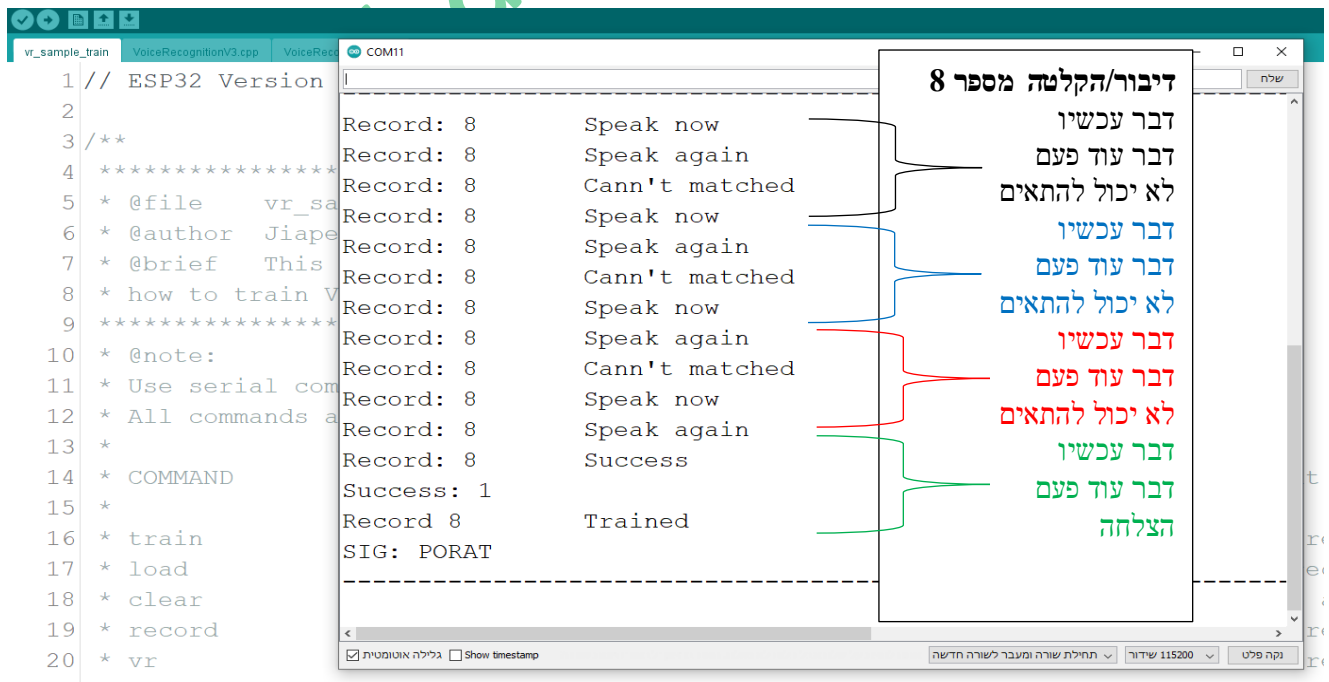
איור 2 : כתיבת מספר ההקלטה הרצויה ואת שם ההקלטה.

4.g כאשר נלחץ על "שלח" או מקש Enter נקבל את המסך המתואר באיור הבא :



איור 3 : מסך הקלטות ההודעה .

4.d במקרה ודיברנו 2 הודעות והמודול לא מצא ביניהן זהות הוא מודיע שאין התאמה.



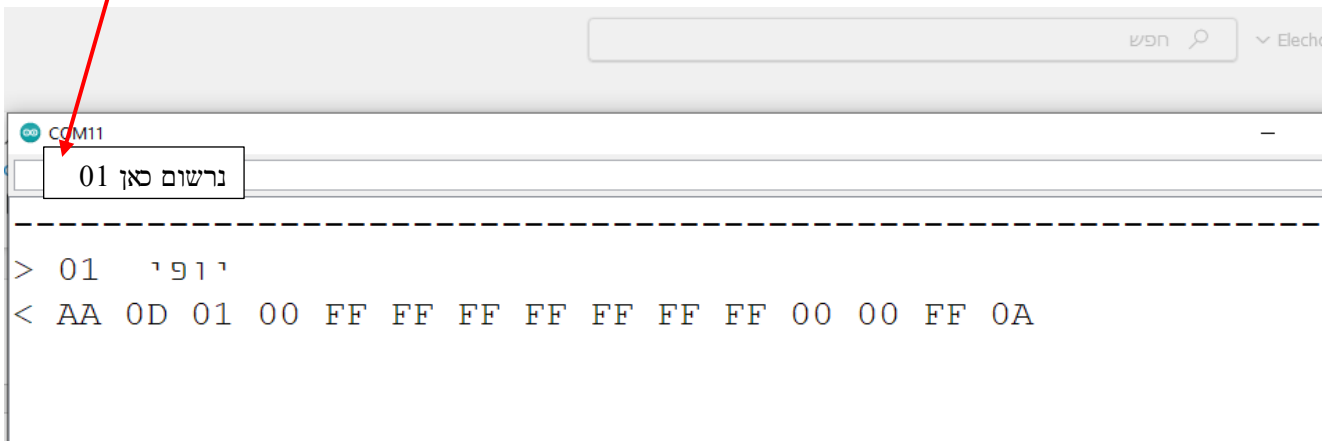
איור 4 : דיבור של הודעות ללא התאמה עד להתאמה.

ניתן לבצע הקלטה אחת בו זמנית . כך נבצע הקלטה אחת אחרי השנייה עד לסיום כל ההודעות הרצויות.

5. שימוש בתוכנית זיהוי ההודעות vr_sample_bridge.ino :

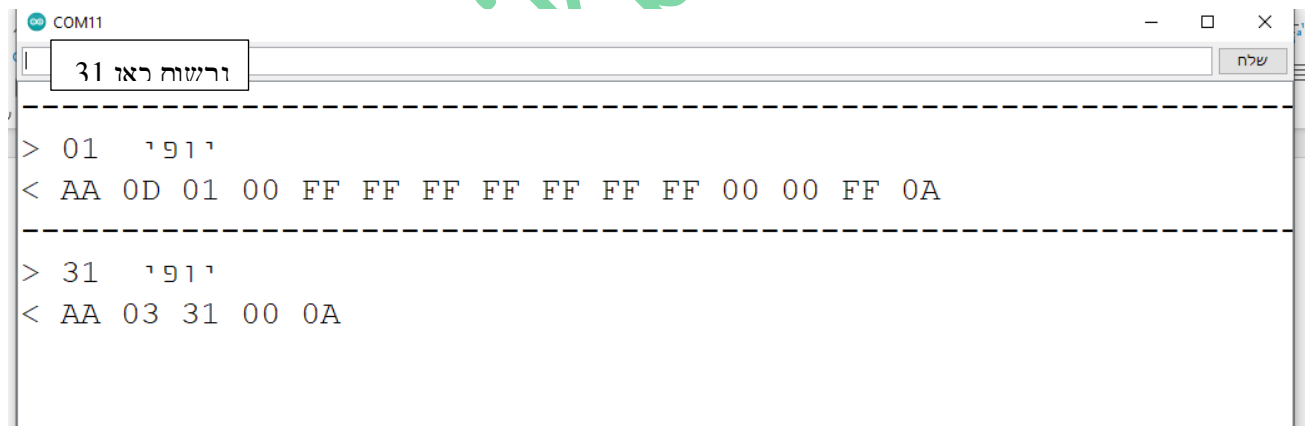
זוהי תוכנית שבה נשמיע את ההודעות ונקבל זיהוי של ההודעות שהשמענו .

5.1 נטען את התוכנית לכרטיס ה ESP32 . בתחילה נבדוק שיש קשר עם המודול . לשם כך נרשום בשורת הפקודה 01 שאומרת בדיקת ריקוגניזר ונלחץ על Enter או שלח ונקבל את המסך שבאיור הבא :



איור 5 : תגובת המודול לפקודה 01 .

5.2 נרשום את הערך 31 בשורת הפקודה ונקיש Enter או שלח ונקבל את המסך שבאיור הבא:



איור 6 : תגובת המודול לפקודה 31

5.3 כדי לבצע זיהוי של הודעות יש לטעון לרקוגניזר את ההודעות שהקלטנו. בדוגמה שנביא כאן נטען לרקוגניזר את ההקלטות הרצויות לנו עם הפקודה 30 . לדוגמה אם נרצה לטעון הודעה 7 נרשום 07 30 . ניתן לטעון מספר הודעות בפקודה אחת . לדוגמה: 08 07 06 05 04 03 02 30 אומרת לטעון הודעות 02 03 04 05 06 07 08 (מקסימום 7 הודעות !) . בהקלטות שביצעתי יש את ההודעות הבאות:

02 SHALOM

03 KADIMA

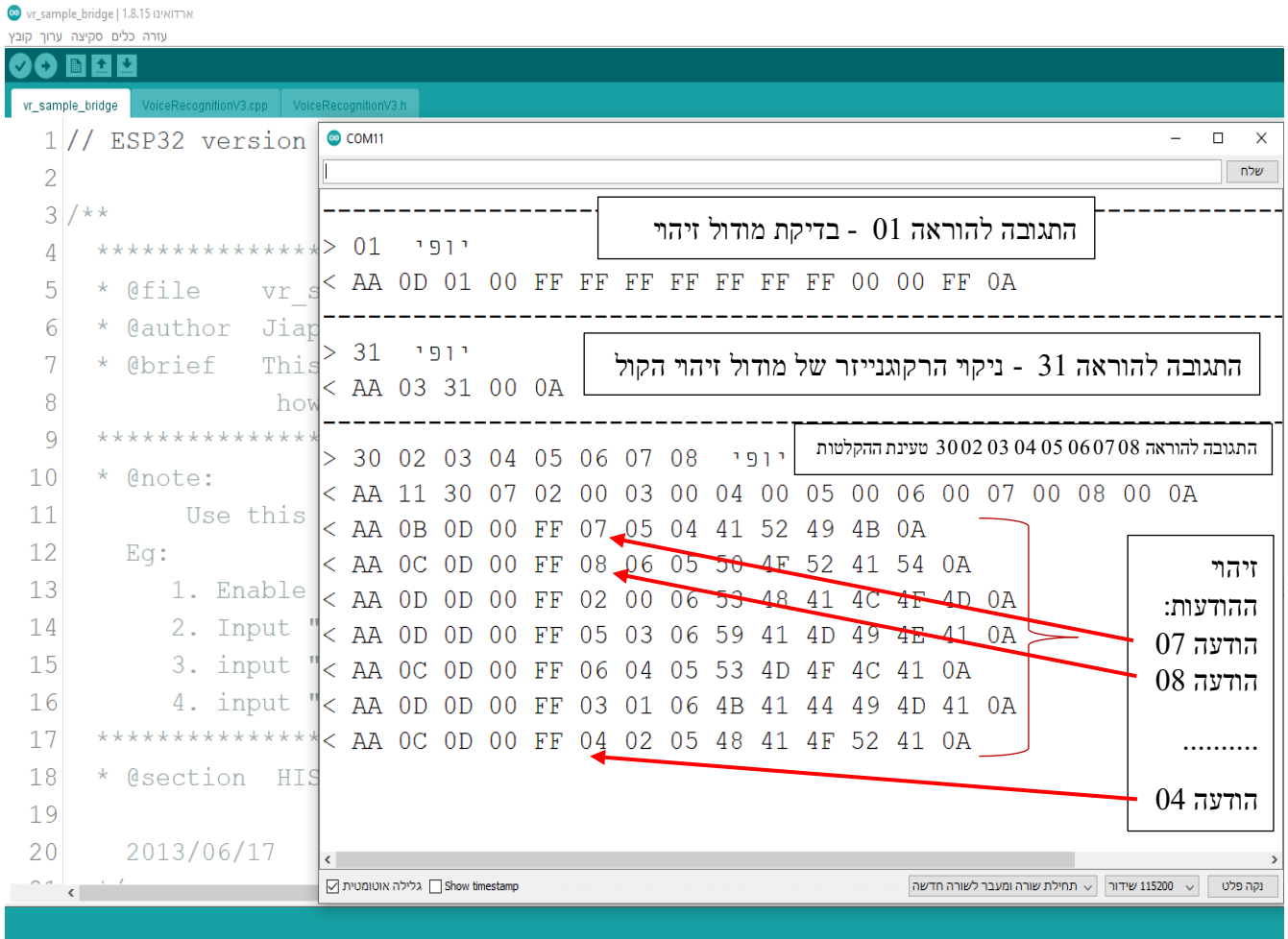
04 HAORA05 YAMINA

06 SMOLA

07 ARIK

08 PORAT

המסך שנקבל עם השמעת ההודעות הרצויות נראה באיור הבא :



איור 7 : מסך זיהוי מספר ההודעה .

עבור כל הודעה המתקבלת במחרוזת תשובה מהמודול זיהוי קול רואים באיבר השישי את מספר ההודעה.

6. ביבליוגרפיה (קישור לאתררים) :

1. <https://blog.frankvh.com/2022/02/21/esp32-support-for-elechouse-voice-recognition-module-v3/>
2. [GitHub - elechouse/VoiceRecognitionV3: Arduino library for elechouse Voice Recognition V3 module](https://github.com/elechouse/VoiceRecognitionV3)
3. [libref.pdf](#)