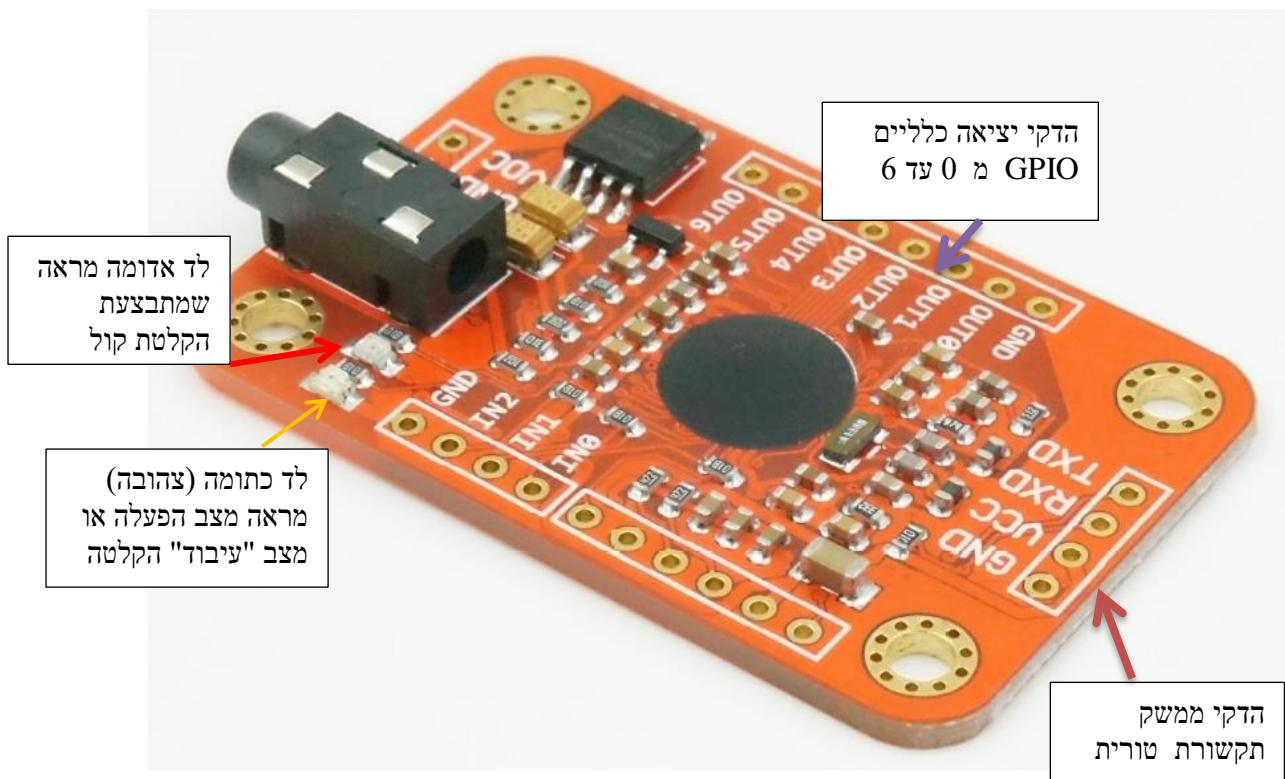


מודול זיהוי דיבור ELECHOUSE V3.1

א. מבוא

מודול זיהוי קול הוא כרטיס "שיודע בדיוק מה אתה אומר". בכרטיס זיהוי קול יש חומרה ותוכנה המסוגלים להבין דיבור אנושי. הכניסה לכרטיס מגיעה על ידי הקלטה של מילה או ביטוי שהמשתמש מדבר אל מיקרופון והפלט הוא ביית (או מספר בייתים) בתקשורת טורית המודיעים האם בוצע זיהוי של מילה או ביטוי שהוקלטו ממיקודם, אילו הקלטות קיימות במעגל זיהוי הנקרא RECOGNIZER ועוד. פלט נוסף הוא דרך הדקי ה GPIO הממוספרים מ OUT0 ועד OUT6 (סה"כ 7 יציאות). ניתן לקבוע מה יהיה בכל אחת מהיציאות כאשר יתבצע זיהוי של אחת מ 7 הקלטות קול שנמצאות ב RECOGNIZER (הכוונה האם ביציאה יהיה 0 או 1 או פולס שלילי). לדוגמה ניתן לקבוע שאם יש זיהוי של ההודעה הראשונה אז ביציאת OUT0 נקבל פולס שלילי שאת רוחבו ניתן לקבוע. יישומי הכרטיס רבים ודוגמאות לשימוש הן שליטה על מכונית או הפעלת ציוד ביתי בעזרת קול. הכרטיס נראה באיור 1 של חברת ELECHOUSE. זוהי גרסה הנקראת v3. מודול כזה עולה בין 20 ל 30 דולר באינטרנט. הכרטיס תומך ויודע לזהות עד 80 פקודות קוליות בסך הכול כאשר ברגע מסוים הוא יודע לזהות 7 פקודות קוליות בו זמנית, כמות המספיקה להרבה יישומים.



איור מספר 1 א : מודול זיהוי קול של חברת ELECHOUSE - voice recognition v3 צד רכיבים

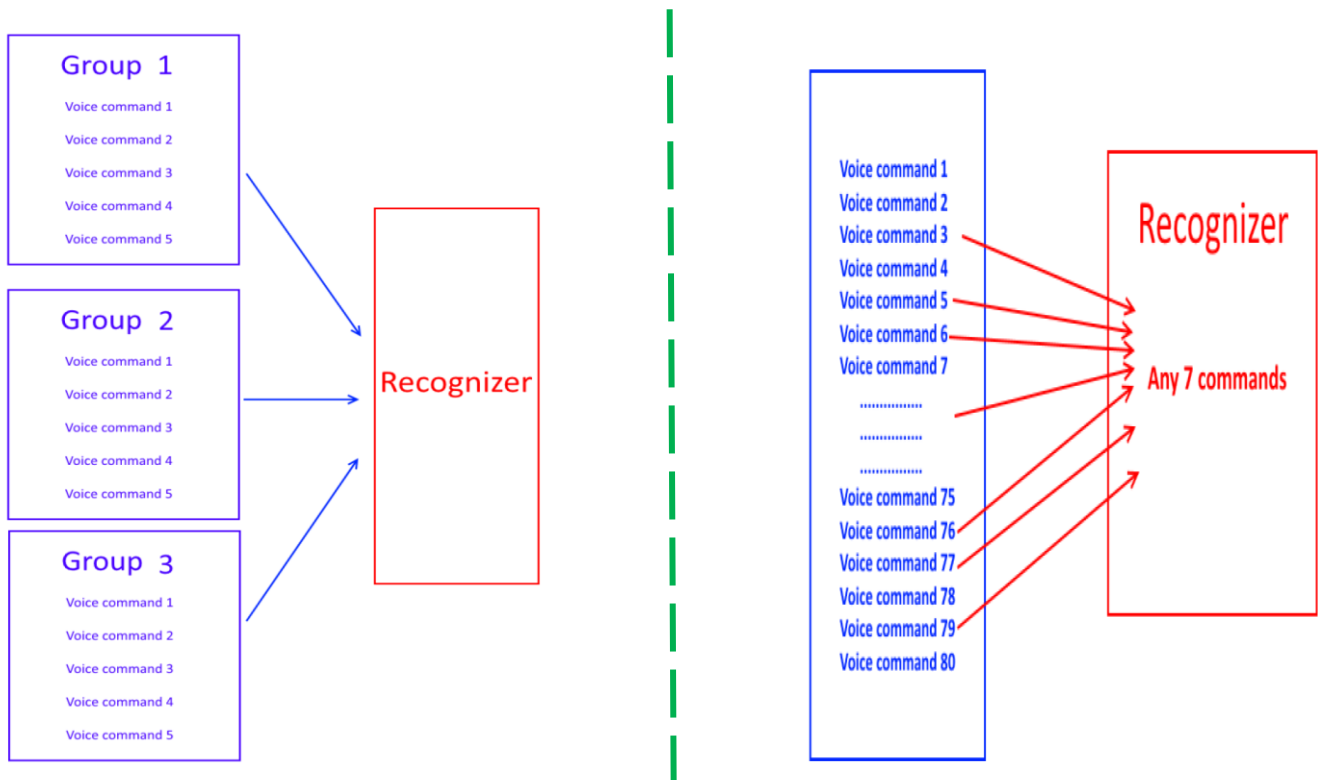


איור 1 ב : צד המעגל המודפס

המודול מקבל פקודות קונפיגורציה (תצורה – פקודות אתחול שאומרות לו כיצד לעבוד) או פקודות הפעלה דרך הממשק של הפורט הטורי. המודול יכול לשמור עד 80 הקלטות קוליות בזיכרון ה-FLASH שלו (כל הקלטה של 1.5 שנייה). יש לו מערכת זיהוי קול הנקראת RECOGNIZER שניתן להכניס אליה שבע הקלטות קול ואז המערכת יכולה לזהות רק את שבע ההקלטות האלו בזמן מסוים. כמובן שניתן לשנות את אחת או יותר מההקלטות בקבוצה של השבע. בהתחלה מקליטים את הודעות הקול אחת אחרי השנייה. בהמשך כאשר רוצים שהמערכת תזהה את ההודעות שהקלטנו נעביר את הקלטות הקול הרצויות לנו (עד 7 הקלטות) אל ה-RECOGNIZER ואם נשמיע מילה או ביטוי, באותו הקול שהקלטנו את אחת משבע הקלטות אלו, נקבל הודעה שיש זיהוי של הודעה ואת מספר ההודעה. במידה והמערכת לא זיהתה את המילה או הביטוי הנאמר היא מודיעה על שגיאה.

בגרסה קודמת של מערכת זיהוי קול (גרסה V2) היו 3 קבוצות של 5 הודעות כל אחת ובזמן נתון היינו מעבירים ל-RECOGNIZER קבוצה אחת מתוך השלוש. העיקרון היה דומה, אם השמענו קול ובוצע זיהוי של אחת מ-5 הקלטות הקול של הקבוצה שב-RECOGNIZER קיבלנו הודעה שיש זיהוי ואת מספר ההודעה. במידה והמערכת לא זיהתה אחת מההודעות היא מודיעה על שגיאה.

נסביר את ההבדלים בין גרסה v2 לגרסה v3 בעזרת איור 2.



איור 2 - ההבדלים בין גרסה V2 (מצד שמאל) לגרסה V3 (מצד ימין).

המודול בגרסה קודמת ל v3 הוא "speaker independent" או בתרגום פשוט "בלתי תלוי ברמקול", כלומר ישנה אפשרות שכאשר מישוהו אחר – ולא זה שהקליט - אומר את המילה או הביטוי יהיה זיהוי נכון.

בגרסה v3 המודול הוא "speaker dependent" כלומר המערכת מזהה את המילה שהוקלטה רק אם נאמרה בקולו של מקליט ההודעות. למרות מה שנכתב ראיתי שכאשר אני מקליט הודעות בקולי ומישוהו אחר אומר את המילים המוקלטות בחלק מהמקרים היה זיהוי. כמובן שחשוב מאד שבזמן ההקלטה נשתמש ברמקול רגיש ושבסביבה לא יהיו רעשי רקע ! הדבר משפיע מאד על הזיהוי של המילה.

ב. נתונים טכניים כלליים של הכרטיס

1. מתח הפעלה : 4.5 – 5.5 V .
2. זרם הפעלה קטן מ 40mA .
3. ממשק דיגיטאלי : UART עם רמות TTL (לא לחבר לרמות של RS232 !!) וגם עם GPIO - General Purpose Input Output - הכוונה להדקי קלט פלט למטרה כללית .
4. ממשק אנאלוגי : קונקטור מיקרופון 3.5 מ"מ מונו + מיקרופון מתאים.
5. מידות פיזיות: 31mm * 50mm .
6. דיוק זיהוי : 99% בסביבה אידיאלית (ללא רעשי רקע או קולות נוספים בסביבה).

ג. מאפיינים

1. תמיכה ב 80 פקודות (הקלטות) קוליות, כל אחת של 1.5 שניות (מילה או שתיים).
2. מקסימום 7 הודעות ניתנות לזיהוי באותו הזמן.
3. בקרה קלה של תקשורת טורית או GPIO.
4. הדקי יציאה כלליים שנשלטים על ידי המשתמש.

ד. מונחים (טרמינולוגיה)

1. VR3 - Voice Recognition module V3 – מודול זיהוי קול גרסה V3.
2. Recognizer - כלי קיבול שאליו מטעינים 7 פקודות קול (7 הקלטות). הדבר דומה לקבוצת כדורסל שבה 12 שחקנים אבל רק 5 שחקנים יכולים להיות במגרש ולהשתתף בזמן מסוים במשחק.
3. Recognizer index - אינדקס זיהוי – מספר בין 0 ל 6 המתאים ל 7 ההקלטות שב Recognizer.
4. Train - אימון (הכשרה) - התהליך של הקלטת הפקודות.
5. Load - טעינה = הטענת הפקודות ל Recognizer.
6. Voice Command Record - רשומת הקלטות הקול - ההקלטות/פקודות שהקלטנו ונמצאות בזיכרון ה FLASH (ממוספרות מ 0 ועד 79).
7. Signature - חותמת - תגובת הטקסט להקלטה.
8. Group - קבוצה - עוזרת לנהל את הרשומות, כל רשומה של 7 הקלטות. קיימת תמיכה בקבוצת מערכת וקבוצת משתמש.

ה. הפעלת הכרטיס

נציג את ספריית הארדואינו ופרוטוקול VR3 עבור ארדואינו. לשם כך נצטייד ב :

- כרטיס voice recognition V3 עם מיקרופון
- לוח ארדואינו אונו או תואם
- תכנה של ארדואינו Arduino IDE
- ספריית Voice Recognition V3 מהאתר

<https://github.com/elechouse/VoiceRecognitionV3/archive/master.zip>

לאחר שנשים את הקבצים בספריית libraries של הארדואינו נחבר את החומרה הבאה בין כרטיס

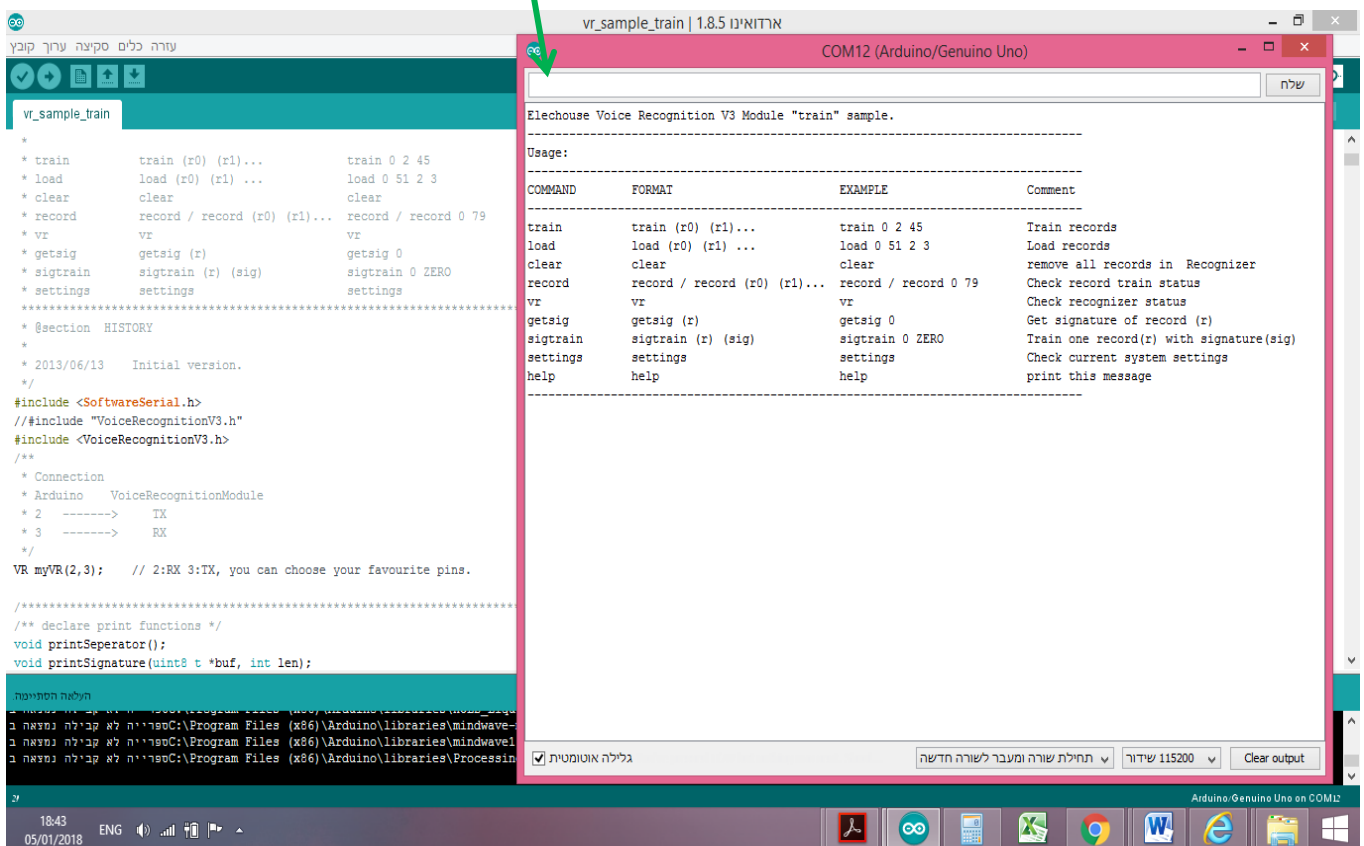
הארדואינו וכרטיס VR :

Arduino	VR Module
5V -----	5V
2 -----	TX
3 -----	RX
GND -----	GND

1.ה הקלטת ההודעות (פקודות) קוליות :

1. הפעל את הקובץ `vr_sample_train.ino` הנמצא בספריית ה `libraries` של תכנת הארדואינו במסלול (תכנת הארדואינו IDE נמצאת במחשב שלי בכונן C ב `Program Files(x86)`) :
`C:\Program Files (x86)\Arduino\libraries\VoiceRecognitionV3-master\examples\vr_sample_train`
2. יש לבחור ב `tools` (כלים) את הלוח הנכון (ארדואינו אונו) ו `port` (פורט) את `COM` הנכון ואז להעלות את התכנה לכרטיס הארדואינו.
3. פתח את ה `Serial Monitor` (המוניטור הטורי) . קבע את קצב התקשורת הטורית ל `115200` (ביטים בשנייה) עם `Newline or Both NL & CR` . נקבל את המסך שבאיור 3 :
4. רשום את המילה `settings` בשורת הפקודה ולחץ `send` (שלה) .

המקום שבו נרשום את הפקודה ונלחץ על `ENTER` או `SEND` (מימין) -



איור 3 : המסך הטורי עם מסך התפריט המתקבל בתכנית `vr_sample_train`

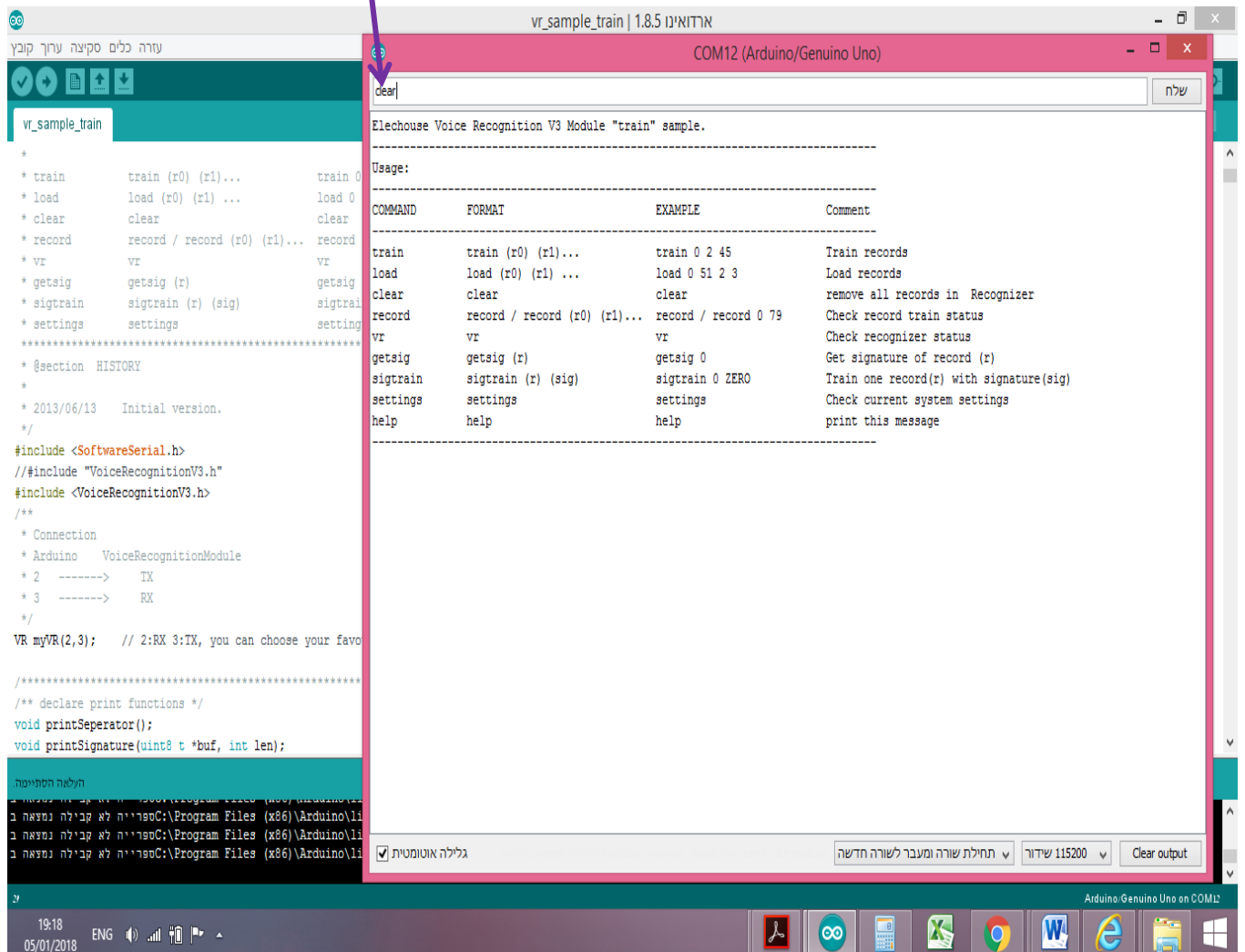
- יש לרשום במסך את הפקודה הרצויה בצד שמאל למעלה של המסך הטורי (כמו שנראה באיור 3) וללחוץ על ENTER או SEND (שלח). אפשרויות העבודה שבתפריט הן:
- train - הכשרה (אימון) "מלמדים" את המודול לקלוט את המילה שאמרנו. לדוגמה: אם נרצה להקליט הודעה 5 נרשום 5 train ונלחץ ENTER או SEND (שלח). אם נרצה יותר מהקלטה אחת נוכל לרשום 5 4 3 1 train ונוכל להקליט 4 הקלטות.
 - load - טעינה של ההקלטות הרצויות ל RECOGNIZER. לדוגמה 5 4 2 0 load טוען את הקלטות מספר 0, 2, 4 ו 5.
 - clear - מנקה (מוחק) את כל ההקלטות שב recognizer.
 - record - מראה את הסטטוס של כל הקלטה רצויה. האם קיימת הקלטה או לא. לדוגמה: אם רשמנו 5 record אז נקבל trained (יש הקלטה) או none (אין הקלטה).
 - vr - מראה אילו הקלטות נמצאות ב RECOGNIZER
 - getsig - קיצור של get signature – קבל חתימה - מראה האם נתנו שם להקלטה מסוימת. לדוגמה 2 getsig תודיע לנו האם להקלטה מספר 2 נתנו שם (לדוגמה RED).
 - sigtrain - הפקודה מקליטה את ההודעה ונותנת לה שם. לדוגמה: 3 BLUE sigtrain מקליט את הפקודה שנאמר (נגיד את המילה כחול) והיא נקראת BLUE.
 - settings - מראה את ההגדרות הנוכחיות של המערכת (קצב תקשורת וכדומה).
 - help - מדפיס שוב את מסך ההודעות שבאיור 3.

ה.2 דוגמאות:

ה.2.א ניקוי ה RECOGNIZER

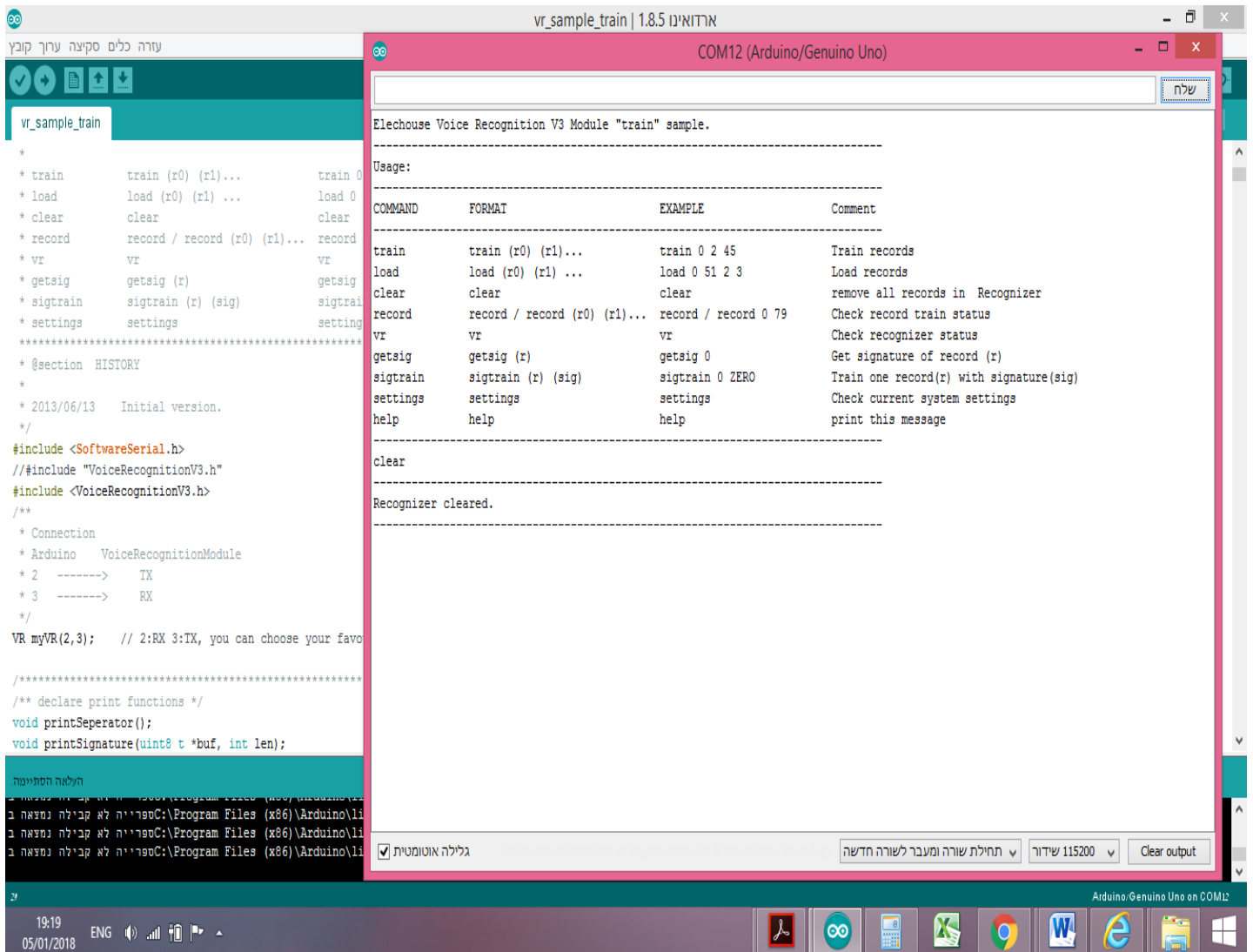
נתחיל עם ניקוי ההודעות שנמצאות ב RECOGNIZER (לא מוחקים את ההודעות שבזיכרון ה FLASH). הדבר מוסבר באיורים 4 ו 5.

פקודת clear לניקוי ההודעות
שב RECOGNIZER



איור 4 : מסך שבו רשמנו את הפקודה clear .

אחרי שרשמנו clear לחצנו ENTER או SEND (שלח). הפקודה הועברה לכרטיס והתשובה שקיבלנו מהכרטיס נראית באיור 5 .



איור 5 : תשובת הכרטיס לפקודה clear .

יש לשים לב שהלד הכתומה דולקת קבוע ולא מהבהבת.

ה.2. ב הקלטה של הודעה

בבצע הקלטה של הודעה . נניח שרוצים להקליט הודעה מספר 0 . לשם כך נרשום את הפקודה train 0 ונלחץ ENTER או SEND . תהליך ההקלטה הוא כזה : הלד הכתומה מהבהבת ומציינת "היכון להשמיע את המילה הרצויה". מיד לאחר מכן נרשם במסך הטורי speak now (דבר עכשיו) , נכבית הלד הכתומה ונדלקת הלד האדומה. כשהלד האדומה נדלקת יש לומר את המילה הרצויה. לאחר מכן נכבית האדומה ושוב מהבהבת הכתומה ואז נרשם במסך הטורי speak again (דבר שוב), הלד הכתומה

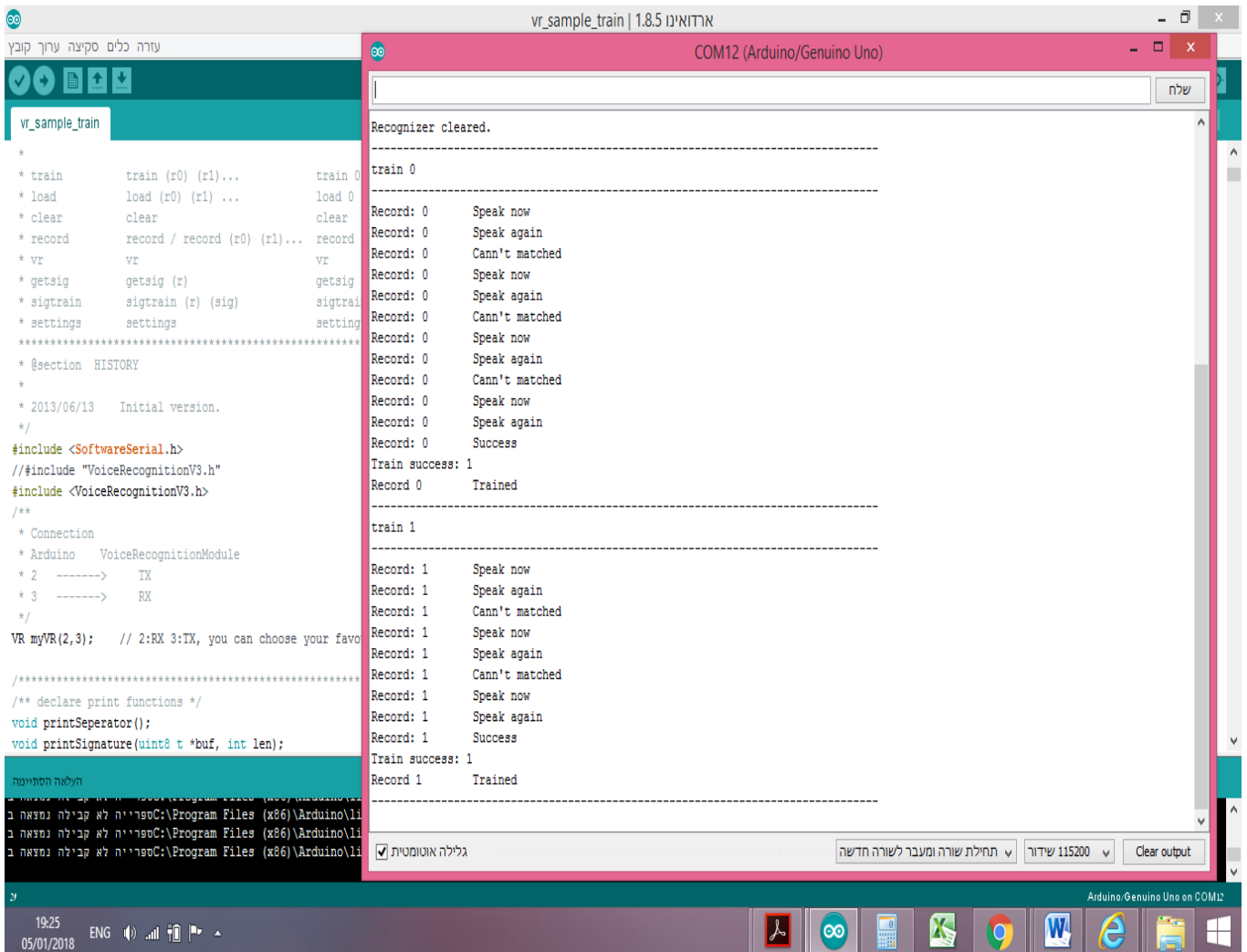
נכבית ונדלקת האדומה. יש לחזור על המילה פעם נוספת. לאחר מכן נכבית האדומה. אם בוצעה התאמה (כלומר הכרטיס הצליח להבחין שיש שזהות במאפיינים של שתי המילים) הלבד הכתומה דולקת קבוע. במידה ולא הייתה התאמה הכרטיס חוזר על תהליך ההקלטה פעם נוספת (מבקש להשמיע פעם אחר פעם את המילה, עד שמתבצעת התאמה. איור 6 מדגים את התהליך.

The screenshot displays the Arduino IDE interface. On the left, the source code for 'vr_sample_train' is visible, including headers for `SoftwareSerial` and `VoiceRecognitionV3`, and a `VR myVR(2,3);` declaration. The right pane shows the serial monitor output for 'COM12 (Arduino/Genuino Uno)'. The output includes a command table, a 'clear' command resulting in 'Recognizer cleared.', a 'train 0' command, and a series of 'Record: 0' messages with responses like 'Speak now', 'Speak again', and 'Cann't matched'. The process concludes with 'Train success: 1' and 'Record 0 Trained'. Hebrew annotations with arrows point to specific parts of the output: 'clear' (התגובה של הכרטיס לפקודת clear ששלחנו מקודם. נרשם מה הייתה הפקודה ששלחנו ומה התוצאה.), 'train 0' (מראה מה הפקודה שנשלחה), 'Speak now' (קיבלנו את מספר ההקלטה והודעה לדבר. יש להשמיע את המילה הרצויה), 'Cann't matched' (התבקשנו להשמיע שוב את המילה), 'Cann't matched' (הכרטיס מודיע שלא הצליח לבצע התאמה בין המילים), 'Cann't matched' (פעמיים נוספות התבקשנו להגיד את המילה והכרטיס לא מצא התאמה.), and 'Success' (בוצעה התאמה).

איור 6 : תשובה לבקשת train 0

ה.2.ג. הקלטה של הודעה מספר 1

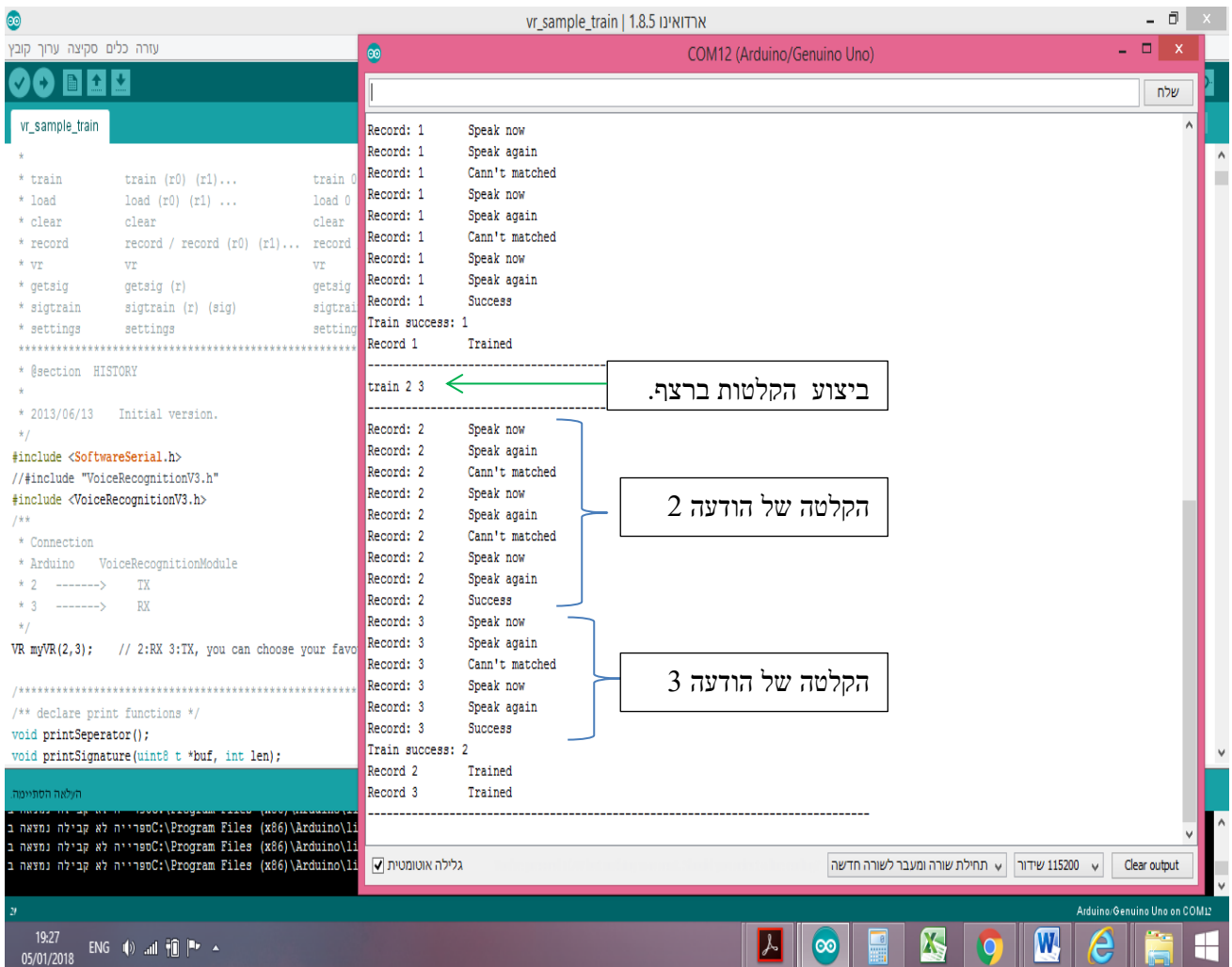
אם נרצה להקליט הודעה מספר 1 נרשום `train 1` ונקבל את המסך שבאיור 7. בחלק העליון רואים עדיין את הקלטת הודעה 0 שהייתה מקודם. בחלק השני רואים את הקלטת הודעה 1. שוב רואים שבוצעה התאמה רק בפעם השלישית.



איור 7 : בקשה להקליט הודעה מספר 1 .

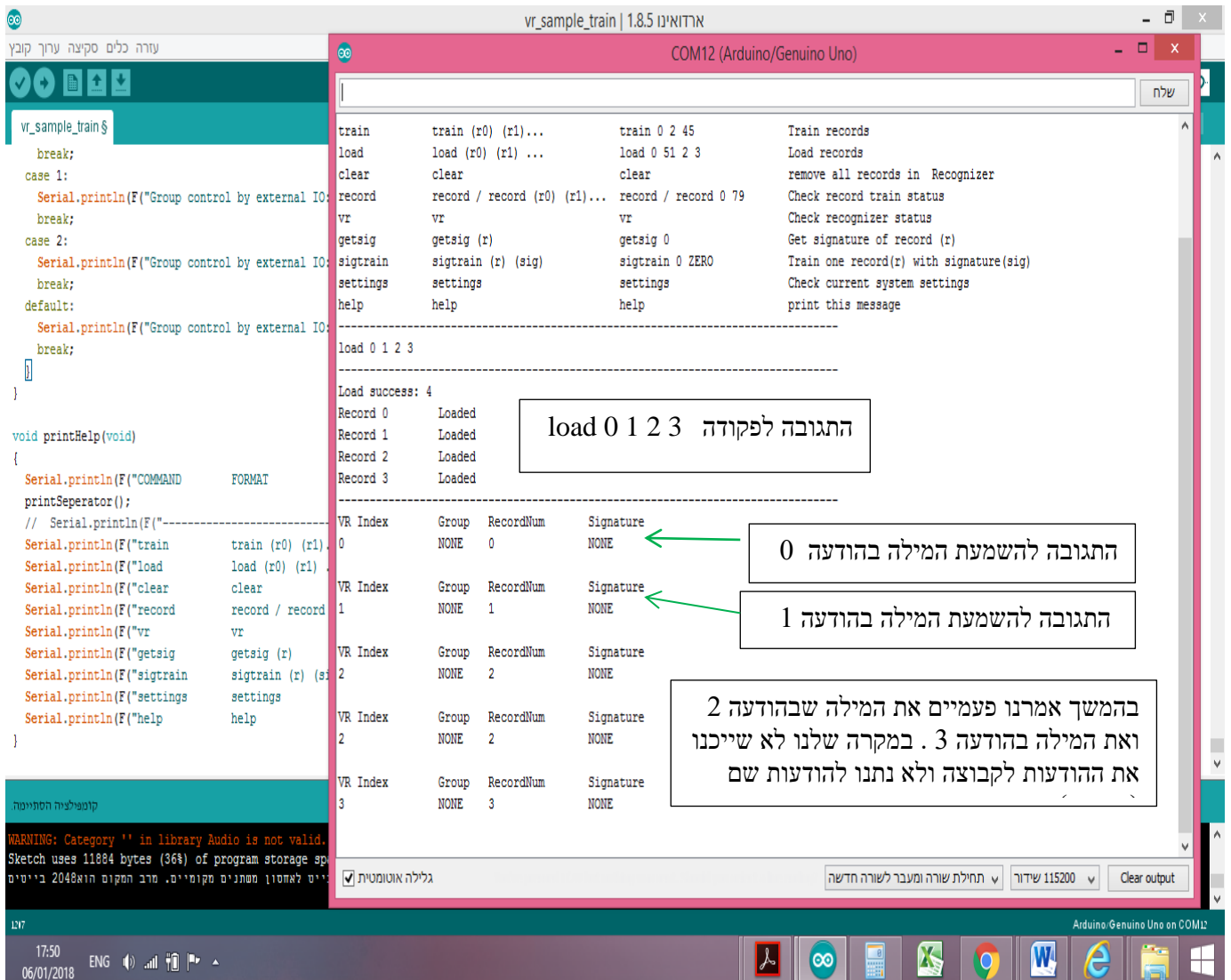
ה.2.ד. הקלטה של מספר הודעות ברצף

כאשר נרצה לבצע מספר הקלטות ברצף נרשום : `train 2 3` (לבצע הקלטות להודעות 2 ו 3) . המסך שיתקבל נראה באיור 8 . במסך רואים שבהקלטה מספר 2 הצלחנו לבצע התאמה רק בפעם השלישית ובהקלטה 3 הצלחנו לבצע התאמה כבר בפעם השנייה.



איור מספר 8 : ביצוע הקלטות ברצף

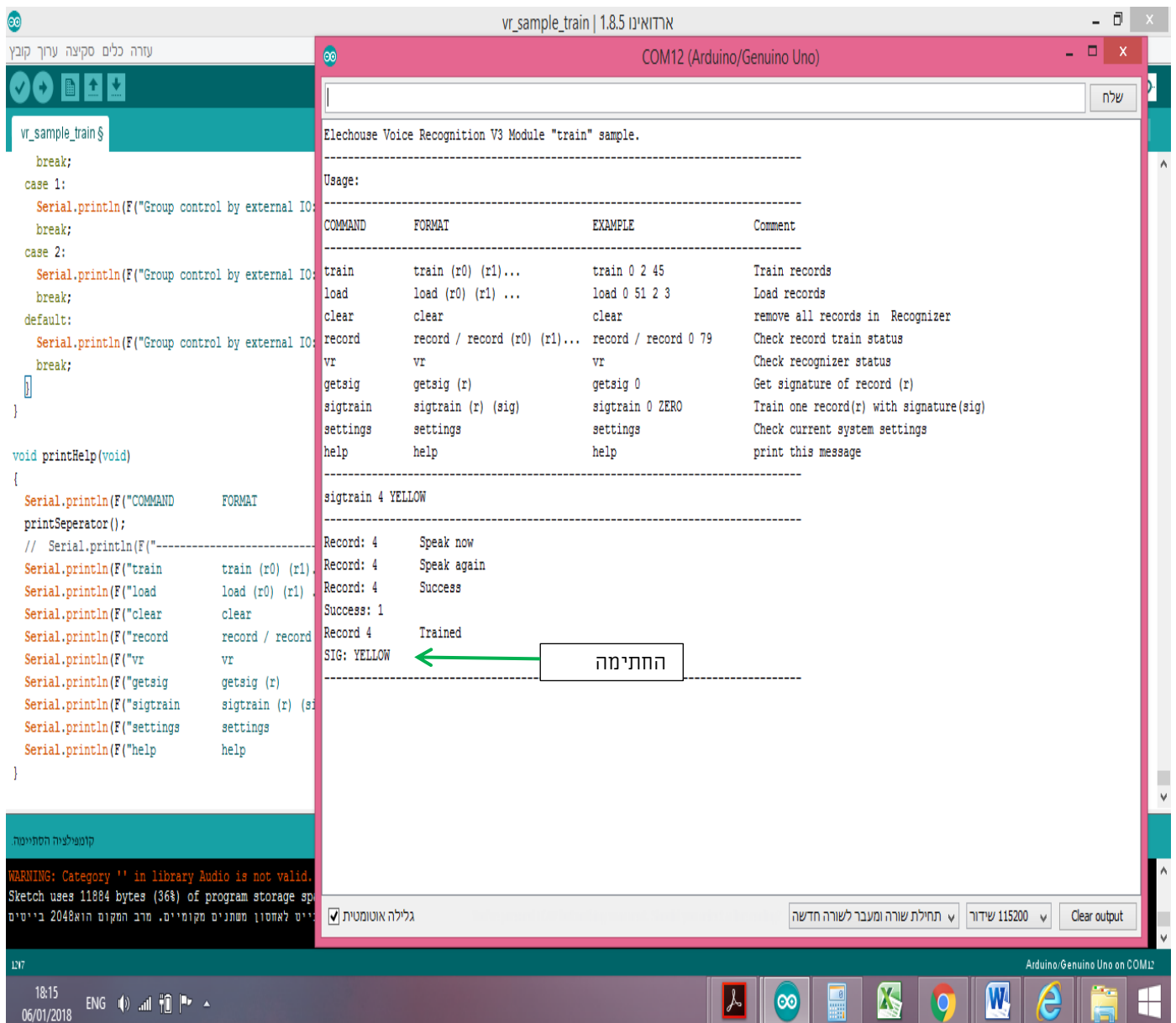
אחרי שהקלטנו מספר הודעות ניתן לטעון אותן אל ה- RECOGNIZER ולבדוק האם הוא מזהה את מה שהקלטנו. נתחיל עם הטענת ההודעות שהקלטנו ל RECOGNIZER. לשם כך נרשום load 0 1 2 3 ונלחץ על ENTER או SEND. התשובה שהכרטיס מחזיר היא אילו הודעות נטענו והלד הכתומה מהבהבת בקצב של שנייה לציון שהיא מתכוננת לקבלת מילה/ כעת ניתן לומר את המילה הרצויה ובמקרה שהכרטיס מזהה מילה הוא מודיע על זיהוי ובמסך הטורי נוכל לדעת איזו הודעה זוהתה. מילה שלא זוהתה איננה נרשמת במסך כטעות. באיור מספר 9 צירפנו תיאור של 2 תהליכים יחד. בהתחלה את קבלת פקודת טעינת ההודעות ל RECOGNIZER ולאחריו זיהוי ההודעות המתאימות.



איור 9 : טעינת הודעות ל RECOGNIZER וזיהוי הודעות

ה.2. הודעה עם חותמת (שם)

אם נרצה לתת להודעה מסוימת חתימה (שם) נרשום את הפקודה `sigtrain 4 YELLOW` כלומר לבצע הכשרה (הקלטה) מספר 4 ולקרא לה `YELLOW`. איור מספר 10 מראה את ביצוע הקלטה מספר 4. במקרה עכשיו בוצעה התאמה כבר בפעם הראשונה.

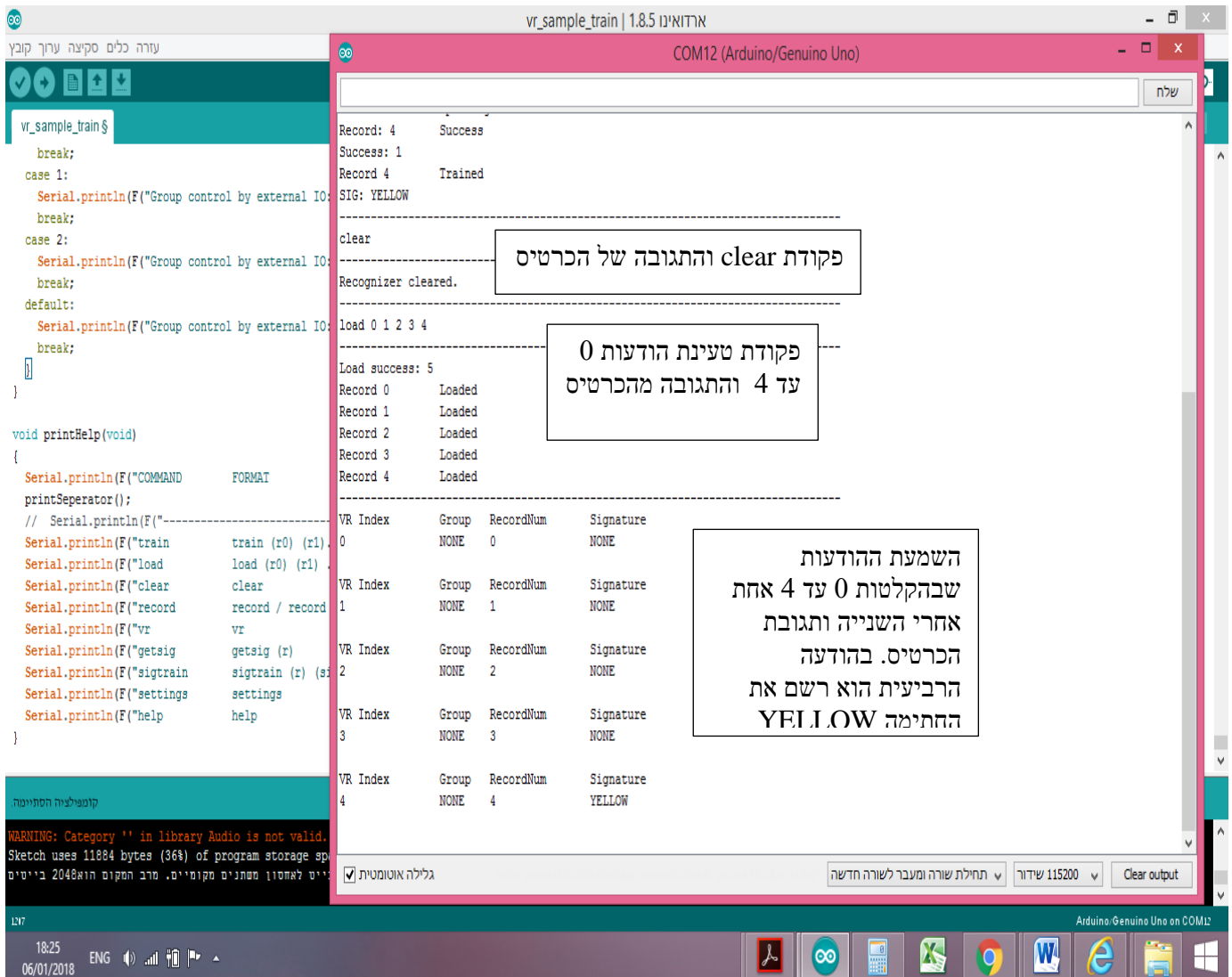


איור 10 : ביצוע הקלטה עם חתימה - sigtrain

כאשר נרשום תכניות שרוצות לזהות איזו הודעה זוהתה אז בתשובה של הכרטיס שהיא מספר בתים נקבל את מספר ההודעה שזוהתה וגם את תווי האסקי של YELLOW .

ה.2.1

כעשיו נבצע את 3 הפעולות הבאות אחת אחרי השנייה : א. נקה את ה RECOGNIZER ב. נטען אליו הודעות 0 עד 4 ג. נאמר את המילים 0 עד 4 . איור 10 מתאר את 3 התהליכים אחד אחרי השני.



איור 10 : פקודת clear לאחר מכן load ולאחר מכן אמרנו את המילים בהודעות 0 עד 4

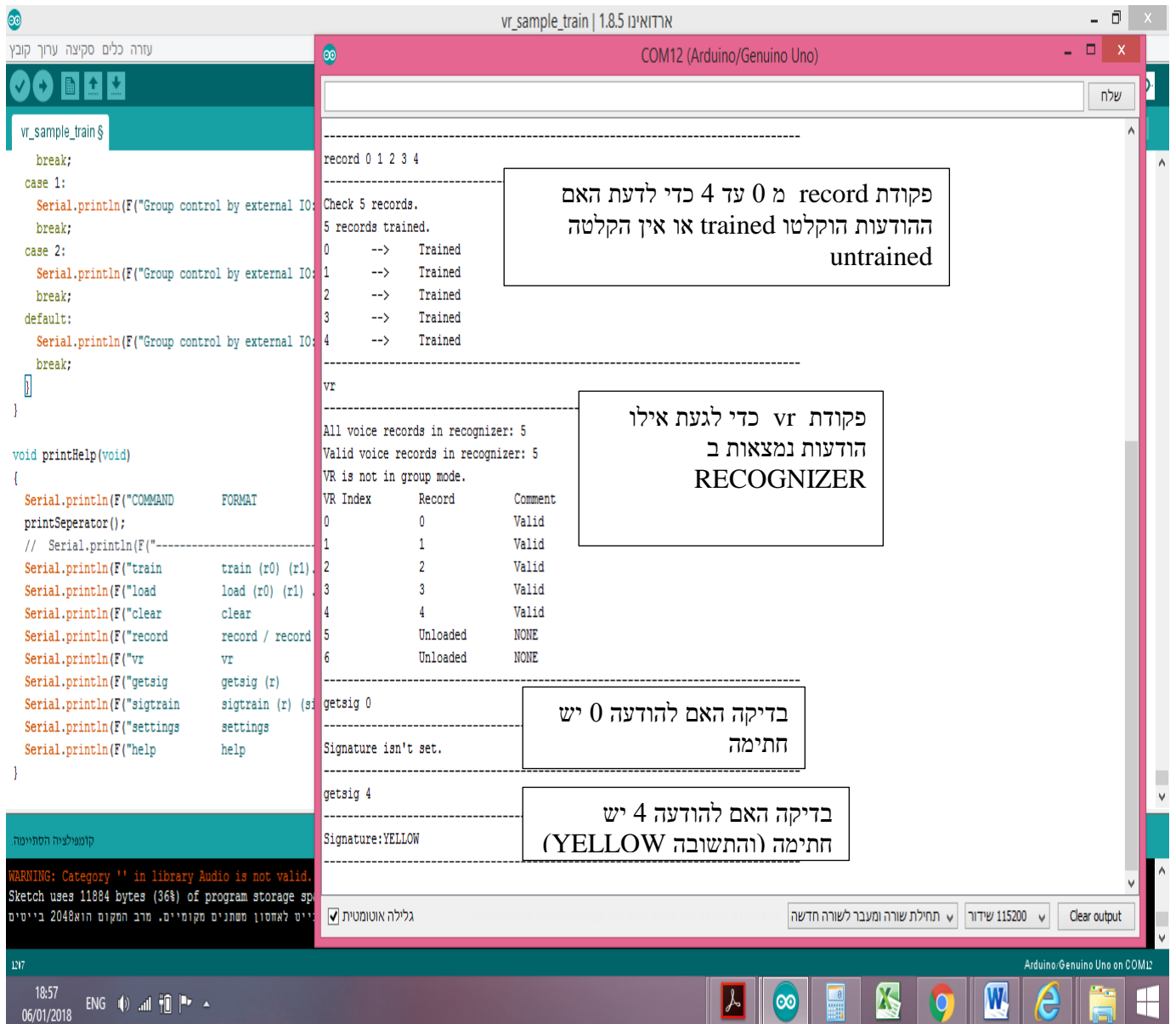
כעת נעבור על פעולות אחרות האפשריות בתפריט של תכנית זו.

א. record - שאומרת האם קיימת הקלטה להודעה מסוימת

ב. vr שאומרת אילו הודעות נמצאות ב RECOGNIZER

ג. getsig - אומרת האם להודעה מסוימת יש חתימה ומה החתימה או שאין להודעה חתימה.

איור 11 מראה את ביצוע 3 הפקודות אחת אחרי השנייה.



איור 11 : פקודות record לאחריה vr ולאחריה getsig

2 המצבים האחרונים בתפריט פקודת settings - מדפיסה למסך את ההגדרות של הכרטיס (קצב תקשורת וכו'). . פקודת help מדפיסה שוב במסך הטורי את התפריט .

1. קבצים נוספים בספרייה

- בספרייה \libraries\VoiceRecognitionV3-master\examples יש 5 קבצים. את הקובץ vr_sample_train שעוזר לבצע הקלטות הסברנו בסעיפים הקודמים.
- קבצים נוספים הם :
- **vr_sample_bridge** מראה כיצד ניתן לשלוט על לד בעזרת הכרטיס. כמו כן הוא מאפשר ניקוי ה RECOGNIZER וטעינת קבצים אליו.
 - **vr_sample_check_baud_rate** - בודק את קצה התקשורת עם הכרטיס.
 - **vr_sample_control_led** - מדגים גם הוא שליטה על הלד שבהדק 13 בעזרת הכרטיס.
 - **vr_sample_multi_command** - מדגים כיצד ניתן להשתמש ביותר מ 7 פקודות לזיהוי.

2. פרוטוקול

אם נרצה לכתוב תכניות יישום שלנו ולא אלו הרשומות בקבצים שבסעיף הקודם, יש להכיר את פרוטוקול התקשורת.

לרכיב מתקשרים בקצב של 9600 ביטים בשנייה, ללא ביט זוגיות עם ביט סיום אחד. כל פקודה של שידור או קליטה מתחילה בערך AAH שנקרא ראש - Head, ומסתיימת בתו 0AH שנקרא סוף - End. ה H בסוף כל תו מראה שזה ערך הקסה-דצימלי. הפורמט של הפקודה ששולחים לכרטיס הוא (משמאל לימין) :

| Head (0AAH) | LENGTH | Command | Data | End (0AH)

תו סיום | נתונים | הפקודה | אורך | תו התחלה

תו ההתחלה ותו הסיום קבועים.

האורך הוא כמות הבתים הכוללת את : ביית האורך, ביית הפקודה והבתים של הנתונים. הפקודה היא תמיד ביית אחד. הנתונים יכולים להיות ביית אחד או יותר.

1.2 הקודים

קבוצת הבתים שנשלחים או מתקבלים נקראת מסגרת Frame. את הקודים נרשום בהקסה דצימלי (אפילו שלא נציין H). בהתחלה נרשום בטבלה את הקודים ולאחריה נרחיב ונפרט על כל פקודה.

1.1.2

פקודות הקוד מחולקות ל 5 קבוצות לפי הטבלה הבאה – טבלה 1 :

מספר הקבוצה	הקבוצה - GROUP	הפקודה בהקסה	הפקודה	תרגום לעברית
1	Frame Code	AA	Frame Head	ראש המסגרת. התו הראשון.
		0A	Frame End	סוף מסגרת. התו האחרון.
2	Check	00	Check System Settings	בדיקת הגדרות המערכת
		01	Check Recognizer	בדיקה של ההקלטות הנמצאות ב RECOGNIZER
		02	Check Record Train Status	בדיקת מצב של ההקלטות
		03	Check Signature of One Record	בודק האם להקלטה מסוימת יש חתימה
		20	Train One Record or Records	הקלטה של מילה פקודה אחת או כמה פקודות
		21	Train One Record and Set Signature	הקלטה של פקודה אחת עם חתימה (שם)
		22	Set signature for Record	תן שם (חתימה) להקלטה מסוימת
4	RECOGNIZER CONTROL	30	Load Record or Records to Recognizer	טען הקלטה או הקלטות ל RECOGNIZER
		31	Clear Recognizer	ניקוי ה RECOGNIZER
		32	Group Control	בקרה על קבוצה
5	RETURN MESSAGE	0A	Prompt	הנחייה
		0D	Voice Recognized	הקול זוהה
		FF	Error	שגיאה

טבלה 1 - פקודות הקוד

ז.1.ב - פרוט והרחבה על הפקודות

• **Check System Settings (00) - בדיקת הגדרות המערכת.**

הפקודה בודקת את ההגדרות הנוכחיות של הכרטיס הכולל את : קצב התקשורת, באיזה אופן נמצאות היציאות , רוחב פולס היציאה , טעינה אוטומטית ופונקציית קבוצה group .

הפורמט : | AA | 02 | 00 | 0A |

AA הוא התו הראשון Head שתמיד נשלח . 02 הוא האורך (כמות הבתים ששולחים ללא תו התחלה וללא תו סיום). במקרה שלנו זה 2 כי זה כולל את הבייט של האורך (כמות הבתים) ועוד הבייט הבא (00) ללא ה 0A - ביית סיום – END .
00 היא הפקודה לבדיקת הגדרות המערכת.
הכרטיס יחזיר את התשובה :

| AA | 08 | 00 | STA | BR | IOM | IOPW | AL | GRP | 0A |

STA הוא קיצור של Trained STatus

untrained - 0

trained - 1

FF – אומר שהערך בהקלטה מחוץ לתחום (אולי כי המילה או הביטוי ארוכים מידי).

BR – Baud Rate - קצב הביטים בשנייה.

אם התשובה 0 או 3 הקצב 9600.

אם התשובה 1 אז הקצב 2400.

אם התשובה 2 אז הקצב 4800

אם התשובה 4 אז הקצב 19200

אם התשובה 5 אז הקצב 38400 .

IOM - Output IO mode - אופן עבודה של היציאות (היציאות מוסברות בפרוט המשך)

0 – פולס

1 – Toggle החלפת מצב

2 – Clear – בהדק יהיה 0 כאשר יש זיהוי

3 – SET – בהדק יהיה 1 כאשר יש זיהוי

IOPW - IO Pulse Width - רוחב הדופק בהדק (רוחב דופק יוסבר בפרוט בהמשך)

המספר שנקבל הוא בין 1 ל 15 .

Power On Auto Load - AL - טעינה אוטומטית בהפעלת הכרטיס. (יוסבר בפרוט בהמשך)

- 0 – אין טעינה אוטומטית בהפעלת חשמל
 - 1 – יש טעינה אוטומטית בהפעלת החשמל
- GRP – Group Control by external IO - בקרה על קבוצה בעזרת הדק חיצוני. (יוסבר בפרוט בהמשך)
- 0 – disable - אין אפשר
 - 1 - System Group
 - 2 - User Group

לדוגמה: אם ב RECOGNIZER אין קבצים ונרשום את הפקודה : AA 02 00 0A
נקבל את התשובה : AA 08 00 00 00 00 00 00 0A

• Check Recognizer (01) - בדיקת ה Recognizer

הפקודה בודקת אילו הקלטות נמצאות ב Recognizer
הפורמט של הפקודה :

| AA | 02 | 01 | 0A |

התשובה שתחזור מהכרטיס היא :

| AA | 0D | 01 | RVN | VRI0 | VRI1 | VRI2 | VRI3 | VRI4 | VRI5 | VRI6 | RTN |

VRMAP | GRPM | 0A |

RVN - number of valid records in recognizer - מספר ההקלטות ב Recognizer
(מקסימום 7).

VRI_n (n=0~6) - ההקלטה הנמצאת ב Recognizer. האינדקס n הוא האינדקס של ההקלטה ב Recognizer (מספר מ 0 ועד 6) ולא מספר ההקלטה (מ 1 עד 80).

VRTN - number of total records in recognizer - מספר ההקלטות הכולל שיש כרגע ב Recognizer.

VRMAP - valid record bit map for VRI0~VRI6 - מפת ביטים של הקלטות עבור VRI0 עד VRI6.

GRPM - group mode indicate - מציינן אופן עבודה בקבוצה :

FF - לא בעבודה בקבוצה

00 ~ 0A מספר הקבוצה ב System Group (קבוצת מערכת).

80 ~ 87 מספר הקבוצה ב USER Group (קבוצת משתמש).

• **Check Record Train Status (02) - בדיקה האם הודעה או הודעות**

הוקלטו .

הפורמט :

בדיקה של כל ההקלטות :

| AA | 03 | 02 | FF | 0A |

התו FF מראה שבדקים את כל ההקלטות.

בדיקה של הקלטות מסוימות :

| AA | 03+n | 02 | R0 | ... | Rn | 0A |

הבית של כמות התווים הוא כי יש אפשרות להתחיל מהקלטה 0 עד n שזה n+1 תווים. אם רוצים 5 הקלטות אז רושמים בתו האורך 7 (זה כולל את 5 ההקלטות, את הפקודה 02 ואת תו האורך. פורמט התשובה המתקבלת מהכרטיס יהיה :

| AA | 5+2n / 02 / N / R0 / STA / ... / Rn / STA / 0A /

N הוא המספר של ההקלטות שבוצעו.

R0~Rn מספר הקלטה

STA - קיצור של סטטוס. האם יש הקלטה

0 - אין הקלטה

1 - יש הקלטה

FF - ההקלטה מחוץ לתחום.

• **Check Signature of One Record (03) - בדיקה האם להקלטה מסוימת**

יש חתימה (שם).

הפורמט :

| AA | 03 | 03 | Record | 0A |

Record היא מספר ההקלטה שרוצים לבדוק אם יש לה חתימה.

פורמט התשובה :

| AA | 03 | 03 | Record | SIGLEN | SIGNATURE | 0A |

SIGLEN - כמות התווים במחרוזת של השם

SIGNATURE - המחרוזת של השם (השם שנתנו להקלטה).

• Restore System Settings (10) - שחזור הגדרות מערכת

החזרת הכרטיס למצב ברירת המחדל שלו.

פורמט הפקודה :

| AA | 02 | 10 | 0A |

פורמט התשובה :

| AA | 03 | 10 | 00 | 0A |

• Set Baud Rate (11) - קבע קצב תקשורת

הפקודה קובעת את קצב התקשורת של הכרטיס. שולחים אותה לאחר שהכרטיס קיבל restart .

פורמט הפקודה :

| AA | 03 | 11 | BR | 0A |

BR - הוא קיצור של Baud Rate . המספר שנשים ב BR קובע את הקצב בסיביות לשנייה :

9600 - 0 2400 - 1 4800 - 2 9600 - 3 19200 - 4 38400 - 5

פורמט התשובה :

| AA | 03 | 11 | 00 | 0A |

• Set Output IO Mode (12) - קביעת אופן העבודה בהדקי ה IO

הפקודה קובעת את אופן העבודה עם הדקי ה IO של הכרטיס. הפקודה משפיעה מיד אחרי ביצוע הפקודה.

פורמט הפקודה :

| AA | 03 | 12 | MODE | 0A |

MODE - אחד מ 4 אופני העבודה של הדקי ה IO של הכרטיס :

0 - pulse ההדק יוציא בזמן זיהוי מילה פולס בהדק ה IO המתאים להודעה שב

RECOGNIZER (יש 7 הדקי IO מ 0 ועד 6). הפולס הוא שלילי כאשר את הרוחב שלו ניתן לקבוע

בפקודה הבאה. לדוגמה אם השמענו מילה המתאימה להודעה החמישית שב RECOGNIZER אז בהדק

IO4 (המתאים להודעה החמישית כי מתחילים מ 0) יהיה פולס שלילי.

1 - Toggle - כל זיהוי של המילה יהפוך את מצב היציאה המתאימה. לדוגמה: הדק IO4 נמצא ב 0 וזוהתה מילה שנמצאת בהודעה החמישית RECOGNIZER אז ההדק יעבור ל 1. בזיהוי הבא הוא יעבור ל 0 וכך הלאה.

2 - Set - ההדק המתאים עובר ל 1 בזיהוי מילה.

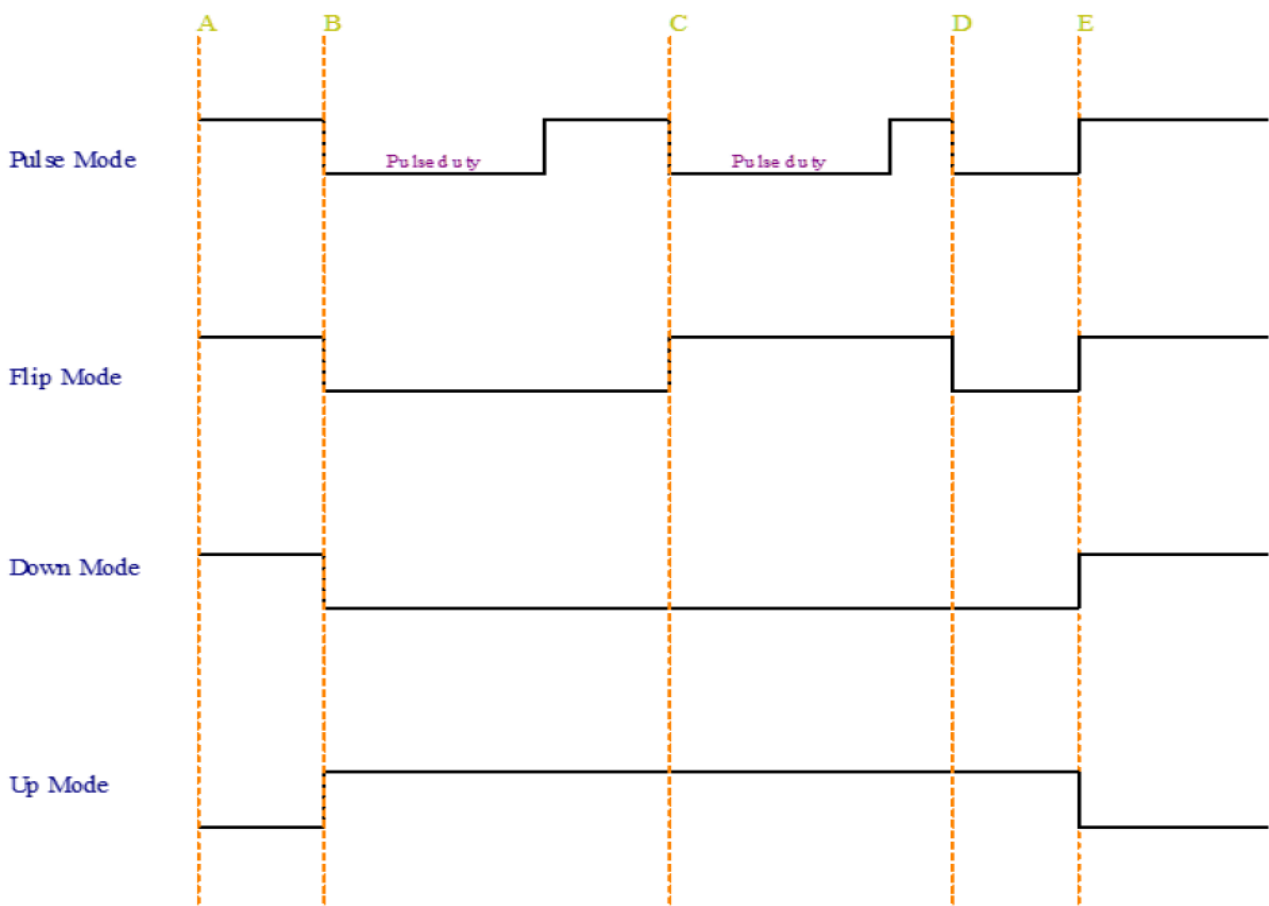
3 - Clear - ההדק המתאים יורד ל 0 בזיהוי המילה.

יש לזכור שבמצבים 2 ו 3 יש לתת reset ליציאה כדי לחזור למצב שלפני זיהוי המילה אחרת הוא יישאר במצב זה.

פורמט התשובה:

| AA | 03 | 12 | 00 | 0A |

איור 12 מסביר את כל אחד מ 4 אופני העבודה. נקודה A היא נקודת התחלה. הנקודות B C D הן נקודות בהן בוצע זיהוי קול נכון. בנקודה E נתנו פקודת reset ליציאה.



איור 12: 4 מצבים אפשריים עבור כל יציאה OUT0 עד OUT6

• (13) Set Output IO Pulse Width - קביעת רוחב הדופק בהדק יציאה

הפקודה משפיעה אם קבענו אם קבענו אופן עבודה של פולס.

פורמט הפקודה :

| AA | 03 | 13 | LEVEL | 0A |

LEVEL - רוחב פולס רצוי . יש 16 זמנים לרוחב הפולס הרצוי מ 10 מילי ועד 1 שנייה לפי :

- 00	10ms
- 01	15ms
- 02	20ms
- 03	25ms
- 04	30ms
- 05	35ms
- 06	40ms
- 07	45ms
- 08	50ms
- 09	75ms
- 0A	100ms
- 0B	200ms
- 0C	300ms
- 0D	400ms
- 0E	500ms
- 0F	1s

פורמט התשובה :

| AA | 03 | 13 | 00 | 0A |

• (14) Reset Output IO - איפוס הדק יציאה IO

הפקודה משמשת באופן עבודה של Set או Clear של ה IO (ראה פקודה 12 שקובעת אופן

עבודה) והיא מחזירה את ההדק למצב ההתחלה של לפני זיהוי המילה.

פורמט הפקודה לאיפוס כל יציאות ה IO:

| AA | 03 | 14 | FF | 0A |

פורמט לאיפוס יציאות IO רצויות :

| AA | 03+n | 14 | IO0 | ... | IO_n | 0A |

IO_n מסמל את מספר יציאת ה IO הרצויה.

לדוגמה : איפוס יציאות 0, 2 ו 4 נרשום : AA 05 14 02 4

פורמט התשובה :

| AA | 03 | 14 | 00 | 0A |

• Set Power On Auto Load (15) - קבע טעינה אוטומטית בהפעלת

חשמל

הפקודה מאפשרת לטעון אוטומטית הודעות רצויות אל ה RECOGNIZER בהפעלת חשמל ללא צורך בפקודות של המשתמש לטעון את ההודעות בעזרת פקודות טעינה.

פורמט הפקודה למנוע טעינת קבוצה אוטומטית בהפעלת חשמל:

| AA| 03 | 15 | 00 | 0A |

פורמט הפקודה לטעינה אוטומטית של הודעות רצויות בהפעלת חשמל :

| AA| 03+n | 15 | BITMAP | R0 | ... | Rn | 0A |

BITMAP - היא מפת ביטים האומרת כמה הודעות לטעון :

0 - אין הודעות, לא לאפשר טעינה אוטומטית.

01 – הקלטה אחת

03 – 2 הקלטות

07 – 3 הקלטות

0F - 4 הקלטות

1F – 5 הקלטות

3F – 6 הקלטות

7F – 7 הקלטות

R0 ~ Rn - איזו הקלטה לטעון

פורמט התשובה :

| AA| 04+n | 15 | 00 |BITMAP | R0 | ... | Rn | 0A |

דוגמה : בניח שרוצים לטעון את ההקלטות 0, 1, 4, 7, 8 באופן אוטומטי בהפעלת חשמל.

הפקודה שנשלח היא : AA 08 15 1F 0 1 4 7 8 0A

• (20) Train One Record or Records – הקלטה של הקלטה אחת או מספר הקלטות

ניתן לבצע הקלטה רצופה של מספר הקלטות, אחת אחרי השנייה.

פורמט הפקודה :

| AA| 03+n | 20 | R0 | ... | Rn | 0A |

R0 ~ Rn מהו מספר ההקלטה

פורמט התשובה אם שלחנו פקודה להקלטה יחידה :

| AA| LEN | 0A | RECORD | PROMPT | 0A |

הפורמט אם שלחנו פקודה למספר הקלטות :

| AA| 05+2n / 20 / N / R0 / STA0 / ... / Rn / STAn / SIG / 0A /

N - כמות ההקלטות המוצלחות שבוצעו

SIG - החתימה של ההקלטה

Rn - מספר ההקלטה

STA – תוצאת ההקלטה : 0 – הצלחה. 1 – timeout (עבר הזמן שמאפשר ניסיון הקלטה) . 2 – ההקלטה מחוץ לתחום (דיברנו זמן ארוך מידי) .

דוגמה: פקודת הקלטה להודעות 2, 4 ו 5 : AA 05 20 02 04 05 0A

• (21) Train One Record and Set Signature – הקלט הודעה אחת

וקבע את החתימה (השם)

פורמט :

| AA| 03+SIGLEN | 21 | RECORD | SIG | 0A |

RECORD – מספר ההקלטה הרצויה

SIGLEN - כמות התווים במחרוזת של השם

SIG - חתימה - מחרוזת תווים של השם שנקרא להקלטה

פורמט התשובה :

| AA | LEN | 0A | RECORD | PROMPT | 0A |

PROMPT

STA - הסטטוס (התוצאה) : 0 - הצלחה. 1 - timeout (עבר הזמן שמאפשר ניסיון הקלטה) . 2 -

ההקלטה מחוץ לתחום (דיברנו זמן ארוך מידי) .

| AA | 05+SIGLEN | 21 | N | RECORD | STA | SIG | 0A |

N - מספר ההקלטה

• Set Signature for Record (22) - קבע חתימה להקלטה

הפקודה נותנת שם להקלטה רצויה.

פורמט הפקודה :

| AA | 03+SIGLEN | 22 | RECORD | SIG | 0A |

SIGLEN - כמות התווים בשם

SIG - השם

פורמט התשובה :

| AA | 04+SIGLEN | 22 | 00 | RECORD | SIG | 0A |

• DELET SIGNATURE(22) - ביטול שם להקלטה

כמו הפקודה למעלה רק כדי לבטל שם להקלטה שנתנו לה שם נרשום :

| AA | 03 | 22 | RECORD | 0A |

התשובה שנקבל :

| AA | 04 | 22 | 00 | RECORD | 0A |

• Load a Record to Recognizer (30) - טעינת הקלטה ל RECOGNIZER

הפקודה טוענת ל RECOGNIZER את אחת ההקלטות הרצויות. מרגע זה ניתן להשמיע את ההודעה ויתבצע זיהוי או חוסר זיהוי.

פורמט הפקודה :

| AA | 2+n | 30 | R0 | ... | Rn | 0A |

n - מציין את כמות ההקלטות שנטען

פורמט התשובה :

| AA | 2+n | 30 | N | R0 | STA0 | ... | Rn | STAn | 0A |

N - מספר ההקלטות שנטענו בהצלחה

STA₀-STAn תוצאת הטעינה : 0 - הטעינה הצליחה. FF - ההקלטה מחוץ לתחום (ארוכה מידי) .

FE - אין הקלטה. FD - ה RECOGNIZER מלא. FC - ההקלטה נמצאת כבר ב

RECOGNIZER /

• (31) Clear Recognizer - נקה את ה RECOGNIZER

הפקודה מוחקת את כל ההקלטות שב RECOGNIZER .

פורמט הפקודה :

| AA | 02 | 31 | 0A |

פורמט התשובה :

| AA | 03 | 31 | 00 | 0A |

• (32) GROUP CONTROL - בקרת קבוצה

א. בחירת קבוצה

קבע אופן עבודה בקבוצה. האפשרויות הן : disable - לא לעבוד בקבוצה. System - קבוצת מערכת . user - קבוצת משתמש . אם פעולת בקרה קבוצתית לא ב disable אז השליטה על הכרטיס מבוקרת על ידי הדקי ה IO החיצוניים .

פורמט הפקודה :

| AA | 04 | 32 | 00 | MODE | 0A |

ה MODE - אופן העבודה החדש : 00 - disable (לא מאופשר שיוך לקבוצה) . 01 - קבוצת מערכת system . 02 - קבוצת משתמש user . FF - בדיקת הקבוצה .

פורמט התשובה :

| AA| 03 | 32 | 00 | 0A |

אם ביקשנו לבדוק את הקבוצה (שלחנו FF) נקבל את התשובה :

| AA| 05 | 32 | 00 | FF | MODE | 0A |

ב. Set User group - קבע קבוצת משתמש

פורמט הפקודה : כדי למחוק קבוצת משתמש (Delete UGRP) נרשום

| AA| 03 | 32 | 01 | UGRP | 0A |

UGRP - User Group - מספר קבוצת משתמש

פורמט הפקודה כדי לקבוע קבוצת משתמש - (Set UGRP) :

| AA| LEN | 32 | 01 | UGRP | R0 | ... | Rn | 0A |

R0 ~ Rn אינדקס מספר ההקלטה (מ 0 עד 6).

פורמט התשובה (הפקודה הצליחה):

| AA| 03 | 32 | 00 | 0A |

ג. Load System Group - טען קבוצת מערכת

הפקודה טוענת קבוצת מערכת אל ה RECOGNIZER . הפקודה תנקה את ה RECOGNIZER .

פורמט הפקודה :

| AA| 04 | 32 | 02 | SGRP | 0A |

SGRP – מספר קבוצת המערכת.

פורמט התשובה :

| AA| 04 | 32 | SGRP | VRI0 | VRI1 | VRI2 | VRI3 | VRI4 | VRI5 | VRI6 | RTN |
VRMAP | GRPM | 0A |

VRI_n - הקלטה הנמצאת ב RECOGNIZER . n הוא האינדקס של ההקלטה ב RECOGNIZER .

RTN - מספר ההקלטות הכולל ב RECOGNIZER .

VRMAP - מפת ביטים של הקלטה תקפה מ VRI0 ~ VRI6 .

GRPM - מציין את מספר קבוצת המערכת מ 0 ועד 0A (סה"כ אפשרי 11 קבוצות).

7. Load user group - טען קבוצת משתמש

טען קבוצת מערכת ל RECOGNIZER . הפקודה מנקה את ה RECOGNIZER .

פורמט הפקודה :

| AA | 04 | 32 | 03 | UGRP | 0A |

UGRP - מספר קבוצת המשתמש .

פורמט התשובה :

| AA | 04 | 32 | UGRP | VRI0 | VRI1 | VRI2 | VRI3 | VRI4 | VRI5 | VRI6 | RTN |
VRMAP | GRPM | | 0A |

VRIn - הקלטה הנמצאת ב RECOGNIZER . n הוא האידיקס של ההקלטה ב RECOGNIZER .

RTN - מספר ההקלטות הכולל ב RECOGNIZER .

VRMAP - מפת ביטים של הקלטה תקפה מ VRI0 ~ VRI6 .

GRPM - מציין את מספר קבוצת המערכת מ 0 ועד 0A (סה"כ אפשרי 11 קבוצות).

ה. Check user group - בדיקת קבוצת משתמש

בודק את תוכן קבוצת המשתמש

פורמט הפקודה לבדיקת כל קבוצת המשתמש (check all user group):

| AA | 04 | 32 | 04 | 0A |

פורמט הפקודה לבדיקת קבוצה או קבוצות רצויות (check user group) :

| AA | 04 | 32 | 04 | UGRP0 | ... | UGRPn | 0A |

פורמט התשובה :

| AA | 0A | 32 | UGRP | R0 | R1 | R2 | R3 | R4 | R5 | R6 | 0A |

UGRP – מספר קבוצת המשתמש

R0 ~ R6 - כל הקלטה

• **Prompt (0A) - הנחייה לכתיבה**

משתמשים בפקודה כאשר רוצים לקבל מהכרטיס נתון בזמן הקלטה.

פורמט הפקודה: שולחים 0A.

פורמט התשובה:

| AA | 07 | 0A | RECORD | PROMPT | 0A |

RECORD – ההקלטה שמבצעים

PROMPT – מחרוזת.

• **Voice Recognized (0D) - זוהה קול**

הנחייה לכרטיס להחזיר נתונים כאשר יש זיהוי קול.

פורמט הפקודה: 0D

פורמט התשובה:

| AA | 07 | 0D | 00 | GRPM | R | RI | SIGLEN | SIG | 0A |

GRPM – מציין אופן הקבוצה. FF – לא בקבוצה. 0A ~ 00 אופן קבוצת מערכת. 87 ~ 80 אופן קבוצת משתמש.

R - ההקלטה שזוהתה

RI - מספר האינדקס של ה RECOGNIZER להקלטה שזוהתה.

SIGLEN - כמות התווים במחרוזת השם של ההקלטה שזוהתה. 0 – אומר שאין שם למחרוזת שזוהתה.

SIG – תוכן השם (התווים בשם)

• **Error (FF) - שגיאה**

הפקודה משמשת להגיד לכרטיס להחזיר סטטוס שגיאה. משתמשים בה כאשר נתנו פקודה והכרטיס לא מבצע את הפקודה ורוצים לדעת מדוע.

פורמט הפקודה : FF

פורמט התשובה :

| AA | 03 | FF | ECODE | 0A |

- ECODE - קוד השגיאה : FF – הפקודה לא מוגדרת . FE – שגיאה באורך הפקודה . FD -
- שגיאת נתון . FC - שגיאה בתת פקודה . FB – שימוש לא חוקי בפקודה.

Library Reference

Index

```

int setBaudRate(unsigned long br);
int setIOMode(io_mode_t mode);
int resetIO(uint8_t *ios=0, uint8_t len=1);
int setPulseWidth(uint8_t level);
int setAutoLoad(uint8_t *records=0, uint8_t len = 0);
int disableAutoLoad();
int restoreSystemSettings();
int checkSystemSettings(uint8_t* buf);
int recognize(uint8_t *buf, int timeout = VR_DEFAULT_TIMEOUT);
int train(uint8_t *records, uint8_t len=1, uint8_t *buf = 0);
int train(uint8_t record, uint8_t *buf = 0);
int trainWithSignature(uint8_t record, const void *buf, uint8_t len=0, uint8_t *retbuf = 0);
int load(uint8_t *records, uint8_t len=1, uint8_t *buf = 0);
int load(uint8_t record, uint8_t *buf = 0);
int clear();
int setSignature(uint8_t record, const void *buf=0, uint8_t len=0);
int deleteSignature(uint8_t record);
int checkSignature(uint8_t record, uint8_t *buf);
int checkRecognizer(uint8_t *buf);
int checkRecord(uint8_t *buf, uint8_t *records = 0, uint8_t len = 0);

int setGroupControl(uint8_t ctrl);
int checkGroupControl();
int setUserGroup(uint8_t grp, uint8_t *records, uint8_t len);
int checkUserGroup(uint8_t grp, uint8_t *buf);
int loadSystemGroup(uint8_t grp, uint8_t *buf=0);
int loadUserGroup(uint8_t grp, uint8_t *buf=0);

void send_pkt(uint8_t *buf, uint8_t len);
void send_pkt(uint8_t cmd, uint8_t *buf, uint8_t len);
void send_pkt(uint8_t cmd, uint8_t subcmd, uint8_t *buf, uint8_t len);
int receive(uint8_t *buf, int len, uint16_t timeout = VR_DEFAULT_TIMEOUT);
int receive_pkt(uint8_t *buf, uint16_t timeout = VR_DEFAULT_TIMEOUT);

```

באתר [libref.pdf](#) נמצא הסבר מפורט על כל פונקציה.

ד. מה מציינים הלדים שבכרטיס ?

בכרטיס 2 לדים (ראה איור מספר 1): אדומה וכתומה . הן עוזרות לדעת מהו המצב שבו עובד כרטיס.

ד.1 בהקלטה :

1. חייווי (ציון) הקלטה : הלב האדומה D1 מהבהבת 3 פעמים בתוך 600 מילי שניות ואז היא נכבית ל 400 מילי שניות ואז מהבהבת 4 פעמים בתוך 600 מילי שניות. עכשיו החיווי של הקלטה מסתיים.
2. התחל דיבור – הלב האדומה D1 לא דולקת לזמן של 400 מילי שניות ואז נדלקת. הקול מוקלט בזמן שהלב דולקת.
3. הקלטה מוצלחת של הודעה קולית בפעם הראשונה : הלב האדומה D1 לא דולקת. הלב הכתומה D2 דולקת לזמן של 300 מילי שניות.
4. הקלטה מוצלחת של הודעה קולית בפעם הראשונה : הלב האדומה D1 לא דולקת. הלב הכתומה D2 דולקת לזמן של 700 מילי שניות.
5. הקלטה נכשלה : הלב הכתומה D2 מהבהבת 4 פעמים בתוך 600 מילי שניות. כישלון בהקלטה קורה במקרים שבהם התגלו 2 פקודות שמע לא תואמות , או שהשמע ארוך מידי, או שאין שמע. יש להתחיל את תהליך ההקלטה לפקודה זו.

ד.2 אופן המתנה

בזמן המתנה – **waiting mode** - הלב הירוקה D2 לא דולקת והלב האדומה D1 מהבהבת בקצב מהיר של 5 פעמים בשנייה כאשר 80 אלפיות שנייה מכל 200 מילי שניות היא דולקת. במצב זה אין זיהוי של הודעות קוליות אלא המתנה לפקודות של תקשורת טורית.

ד.3 – זיהוי

בזיהוי הלב הירוקה D2 בחושך והלב האדומה D1 מהבהבת בקצב איטי. היא דולקת לזמן של 100 מילי שניות כל 1500 מילי שניות. במצב זה המודול מעבד את הקול שנקלט ואם יש התאמה היא תתקבל מידית דרך התקשורת הטורית.

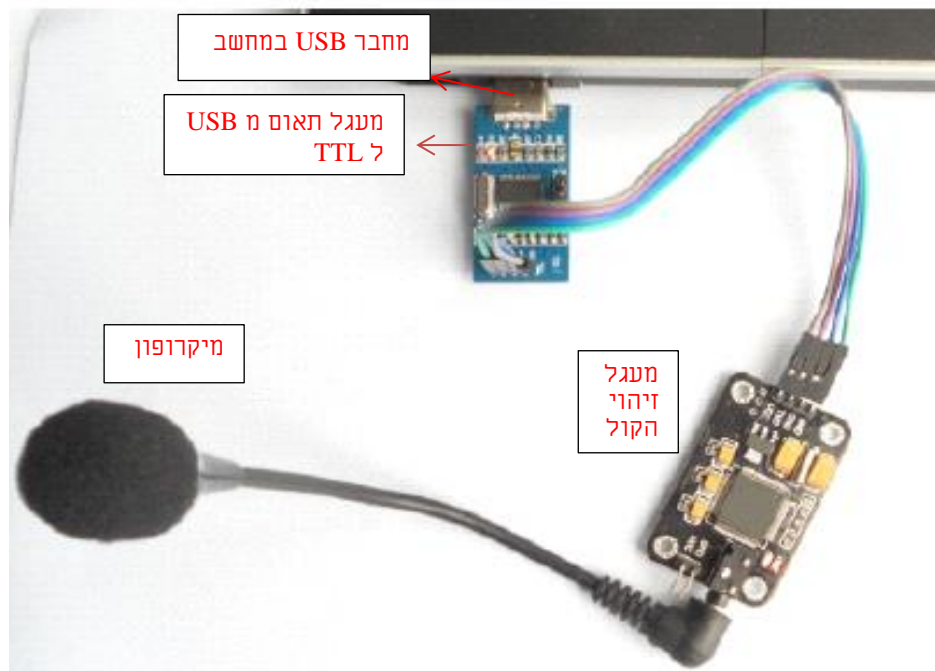
הערה : באופן מעשי קשה לעבוד בזמן ההקלטה עם הלדים. עדיף לקרא על המחשב את המצב הנוכחי כפי שיוסבר בהמשך.

ה. הקלטה

לפני השימוש בכרטיס יש להקליט אליו פקודות קוליות. לכל פקודה קולית יש אורך מקסימאלי של 1300 מילי שניות שמבטיח שניתן להקליט את רוב המילים. כאשר התחלנו להקליט לא ניתן להפסיק את תהליך ההקלטה עד שסיימנו להקליט את כל 5 הפקודות הקוליות של הקבוצה. בזמן ההקלטה נמחק כל התוכן הקודם של אותה הקבוצה. בזמן הקלטה הכרטיס איננו עונה לפקודות תקשורת טורית אחרות.

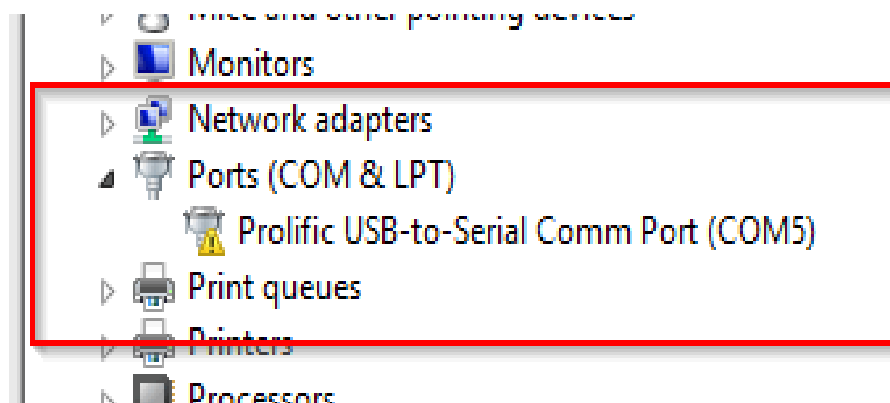
ה.1 דוגמא להקלטה

בתמונה מספר 1 מתואר מעגל הקלטה עם הכרטיס ומעגל תאום המעביר מ USB ל TTL . את מעגל התאום ניתן להשיג באינטרנט בדולרים בודדים.



תמונה מספר 1 : מעגל ההקלטה

הערה: עבור גרסאות windows שונות ועבור 32 או 64 ביט יש להתקין את הדרייבר המתאים עבור המתאם מ USB ל TTL. אם מקבלים במסך מנהל ההתקנים של לוח הבקרה במחשב את ההודעה הבאה עם משולש צהוב וסימן קריאה זה אומר שיש בעיה בדרייבר ולא נוכל לבצע הקלטות !!



באתר

<http://www.totalcardiagnostics.com/support/Knowledgebase/Article/View/92/20/prolific-usb-to-serial-fix-official-solution-to-code-10-error>

ניתן למצוא הסבר על התקלה ואיך לפתור אותה. רק אחרי טעינת הדרייבר המתאים נוכל לבצע הקלטות. נייער בתוכנה הנקראת **AccessPort** שנותנת למשתמש להתקשר לפורט הטורי של המחשב בצורה קלה.

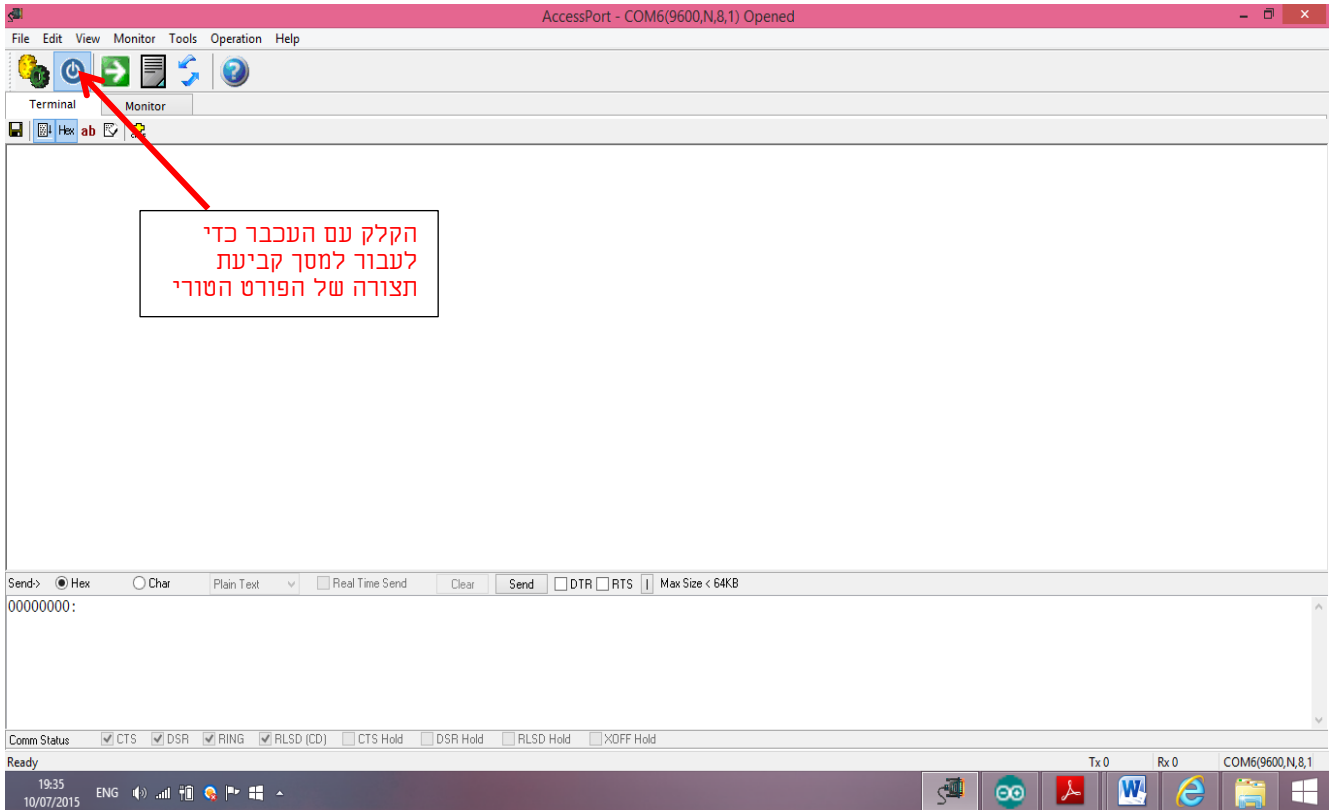
את התוכנה ניתן להוריד בחינם באתר : <http://www.sudt.com/en/ap/download.htm> . יש

להוריד את הגרסה המתאימה למערכת ההפעלה שבמחשב שלך.

החיבור בין מעגל התאום מ USB ל TTL לכרטיס זיהוי הקול:

יש לחבר 4 חיבורים בצורה הבאה : א. חוט של ה 5 וולט ב. חוט האדמה ג. קו ה TX בכרטיס המתאם ל RX בכרטיס זיהוי הקול ד. קו RX בכרטיס המתאם ל TX בכרטיס זיהוי הקול. ברגע שהכרטיסים חוברו והכרטיס המתאם הוכנס ל USB הלד האדומה בכרטיס הזיהוי תהבהב.

נפעיל את התכנית AccessPort ונקבל את המסך שבתמונה 2 :

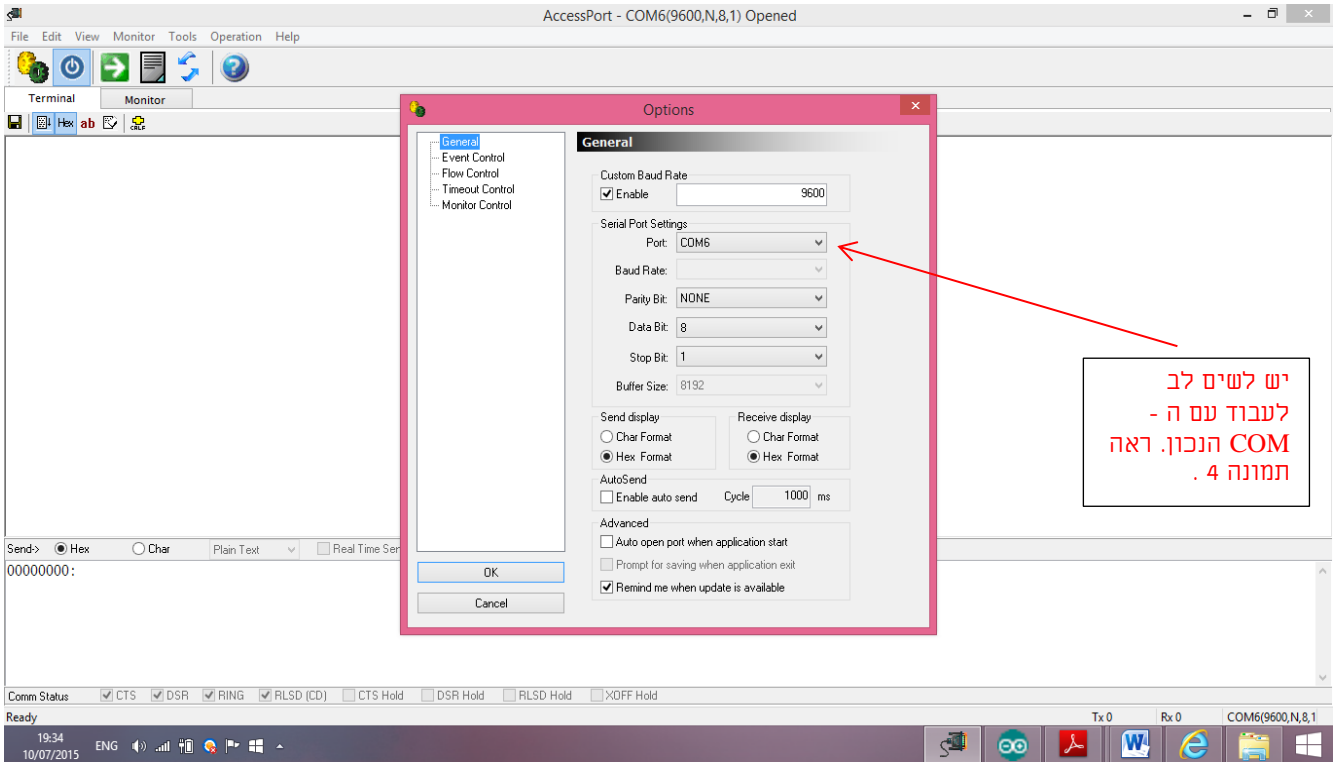


תמונה 2 : מסך התכנית AccessPort

נקליק עם העכבר על האובייקט של 2 גלגלי השיניים שנמצא למעלה מצד שמאל ומסומן בעזרת חץ ונקבל את המסך שבתמונה 3 . נשנה את הנתונים במסך לפי הנאמר בסעיף ה.1.א .

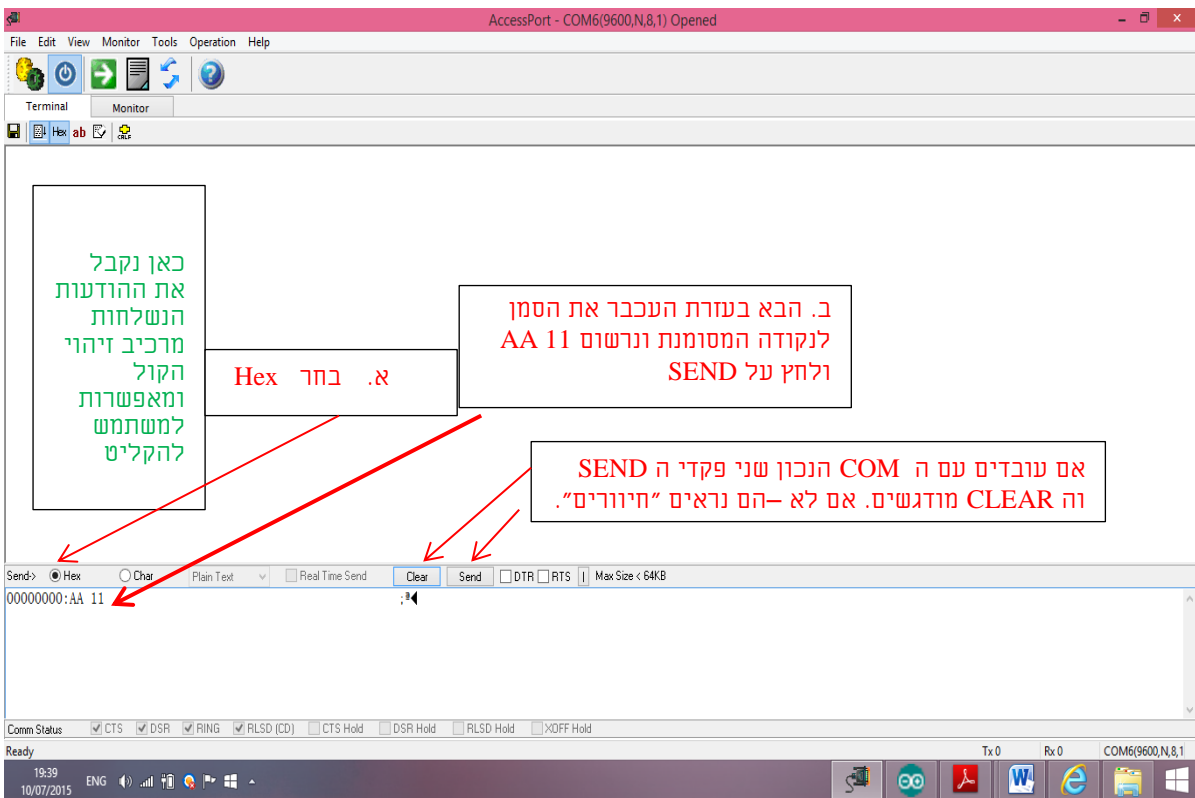
ה.1.א קביעת העבודה עם פורט התקשורת הטורית

1. קצב תקשורת 9600 . (Baud :9600).
2. ללא ביט זוגיות. (Parity:None)
3. נתון של 8 ביט. (Data bit:8)
4. ביט סיום אחד (Stop bit:1)
5. שליחת נתון ב הקסה דצימלי . Send format:Hex
6. האות הנקלט הוא תווי. (Receive format: Char)



תמונה 3 : קביעת העבודה עם פורט התקשורת הטורית

נלחץ על פקד OK בצד שמאל למטה ונחזור בחזרה למסך שבתמונה 2.
במסך זה נבצע את הדברים המוסברים בתמונה מספר 4



תמונה מספר 4 : מסך ההפעלה

בזמן ההקלטה הכרטיס מבקש מהמשתמש להקליט כל הודעה **פעמיים**. מרגע ששידרנו בעזרת הפקד SEND את הפקודה AA 11 האומר שרוצים להקליט 5 הודעות עבור הקבוצה הראשונה, כרטיס זיהוי הקול שולח הודעות למשתמש מה הפעולה שהוא צריך לבצע. כאשר עובדים באופן עבודה COMPACT ההודעות שניתן לקבל הן :

- START** - השמע את ההודעה הקולית שאתה רוצה
- Again** - חזור. כאשר יופיע שוב START יש לחזור ולהשמיע שוב את ההודעה
- Finish one** - הסתיימה אחת ההקלטות של ההודעה הקולית. בסיום כל אחת מ 5 ההקלטות של הקבוצה מופיעה ההודעה הזו .
- Group1 finished** - בסיום הקלטת כל 5 ההודעות של קבוצה מספר 1 תופיע ההודעה של סיום קבוצה 1
- No voice** - אין הקלטה – אם היינו אמורים להשמיע הודעה ובזמן שהיינו אמורים לדבר ולהקליט לא השמענו שום הודעה. מיד אחרי הודעה זו נקבל שוב START
- Different** - כאשר מקליטים פעם שנייה את אותה ההודעה והיא שונה מהראשונה.

אם ההקלטה מתבצעת בראוי אז נקבל את ההודעות הבאות :

START

המשתמש אומר את המילה הרצויה

Again

היכון לחזור על המילה הקודמת

START

המשתמש חוזר על המילה הקודמת

Finish one

הכרטיס אומר שהסתיימה הקלטת המילה הראשונה

START

המשתמש אומר את המילה הרצויה

Again

היכון לחזור על המילה הקודמת

START

המשתמש חוזר על המילה הקודמת

Finish one

הכרטיס אומר שהסתיימה הקלטת המילה הבאה

START

המשתמש אומר את המילה הרצויה

Again

היכון לחזור על המילה הקודמת

START

המשתמש חוזר על המילה הקודמת

Finish one

הכרטיס אומר שהסתיימה הקלטת המילה הבאה

START

המשתמש אומר את המילה הרצויה

Again

היכון לחזור על המילה הקודמת

START

המשתמש חוזר על המילה הקודמת

Finish one

הכרטיס אומר שהסתיימה הקלטת המילה הבאה

START

המשתמש אומר את המילה הרצויה

Again

היכון לחזור על המילה הקודמת

START

המשתמש חוזר על המילה הקודמת

Finish one

הכרטיס אומר שהסתיימה הקלטת המילה הבאה

Group1 finished

הסתיימה ההקלטה של כל 5 המילים של קבוצה אחת.

עכשיו המשתמש יכול להקליט את 5 המילים עבור קבוצה מספר 2 על ידי שליחת הפקודה : AA 12
ושוב נקבל את אותן ההודעות (חוץ מההודעה הסופית שהסתיימה הקלטת קבוצה 2).
עבור הדוגמא שניתן בסעיף הבא הקלטנו את המילים : 1. "חושך" 2. "אדום" , 3. "צהוב" 4. "ירוק"
5. "כולן",
ניתן להקליט את כל 3 הקבוצות אחת אחרי השנייה או רק אחת מהקבוצות.

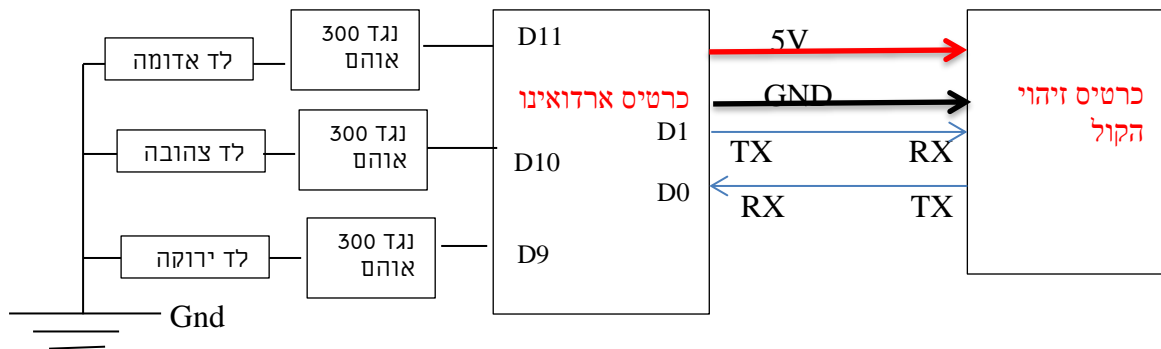
ה.2 דוגמא : הדלקת לדים בעזרת פקודות קוליות

בדוגמא הקודמת הקלטנו לקבוצה 1 את ההודעות: 1. "חושך" 2. "אדום" , 3. "צהוב" 4. "ירוק"

5. "כולן", ונפעיל/נכבה 3 לדים בהתאם לפקודות שנשמיע.

לשם כך נחבר את כרטיס זיהוי הקול לארדואינו אוננו (או כל כרטיס ארדואינו אחר) כאשר צורת החיבור

נראית באיור מספר 2 :



איור מספר 2

התכנית תראה כך :

```
int red = 11; // הLED האדומה מתחברת להדק דיגיטאלי 11 בכרטיס הארדואינו
int yellow = 10; // הLED הצהובה מתחברת להדק דיגיטאלי מספר 10 בכרטיס הארדואינו
int green = 9; // הLED הירוקה מתחברת להדק דיגיטאלי 9 בכרטיס הארדואינו
byte com = 0; // הבייט שיחזור מכרטיס זיהוי הקול
void setup()
{
  Serial.begin(9600); // אתחול התקשורת הטורית לקצב 9600
  pinMode(red, OUTPUT); // קביעת מצב ההדק שבו הLED האדומה כפלט
  pinMode(yellow, OUTPUT); // קביעת מצב ההדק שבו הLED הצהובה כפלט
  pinMode(green, OUTPUT); // קביעת מצב ההדק שבו הLED הירוקה כפלט
  // בדיקה שכל הLEDים מחוברות נכון - הדלקה וכיבוי שלהן
  digitalWrite(red,1);
  digitalWrite(yellow, 1);
  digitalWrite(green, 1);
  delay(1000);
  digitalWrite(red,0);
  digitalWrite(yellow, 0);
  digitalWrite(green, 0);
}
```

```
Serial.write(0xAA); // ( Head) שליחת הודעה לכרטיס הקול ששולחים לו פקודה
Serial.write(0x37); // Compact "מרוכז" לעבוד בכרטיס לעבוד במצב
delay(1000); // השהייה של שנייה
Serial.write(0xAA); // ( Head) שליחת הודעה לכרטיס הקול ששולחים לו פקודה
Serial.write(0x21); // פקודה לכרטיס לייבא את קבוצה 1 ולהיות מוכן לפקודה קולית
}
void loop()
{
  while (Serial.available()) // כאשר נקלט תו בתקשורת טורית
  {
    com = Serial.read(); // com הכנסת התו שהתקבל בתקשורת טורית למשתנה
    switch (com) // האם התו שהתקבל הוא ?
    {
      case 0x11: // האם הכרטיס זיהה שאמרנו "חושך"
      {
        digitalWrite(red, 0); // כיבוי הled האדומה
        digitalWrite(yellow, 0); // כיבוי הled הצהובה
        digitalWrite(green, 0); // כיבוי הled הירוקה
      }
      break;
      case 0x12: // האם הכרטיס זיהה שאמרנו "אדום"
      {
        digitalWrite(red, 1); // הדלקת הled האדומה
        digitalWrite(yellow, 0);
        digitalWrite(green, 0);
      }
      break;
      case 0x13: // האם הכרטיס זיהה שאמרנו "צהוב"
      {
        digitalWrite(red, 0); // כיבוי הled האדומה
        digitalWrite(yellow, 1);
        digitalWrite(green, 0);
      }
      break;
    }
  }
}
```



```
case 0x14: // האם הכרטיס זיהה שאמרנו "ירוק"
{
    digitalWrite(red, 0); // כיבוי הLED האדומה
    digitalWrite(yellow, 0);
    digitalWrite(green, 1);
}
break;
case 0x15: // האם הכרטיס זיהה שאמרנו "כולן"
{
    digitalWrite(red,1);
    digitalWrite(yellow, 1);
    digitalWrite(green, 1);
}
break;
}
}
```