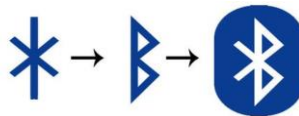


## בלוטות – HC-05 ו HC-06

### 1.א. היסטוריה ומבוא כללי

המילה -בלוטות - Bluetooth היא בעברית שן כחולה. זוהי רשת אלחוטית המשתמשת בתדר גבוה מאד - UHF - של 2400-2483.5 מגה הרץ . היא משמשת לתקשורת נתונים למרחקים קצרים (עד כ 10 מטר) ובהספק נמוך . ניתן לחבר אליה עד 8 מכשירים ברשת פיקונט - piconet - ( פיקונט – פיקו זה  $10^{-12}$  ומציין רשת קטנה מאד) , והיא שימושית לקשר בין מחשב לציווד היקפי ( לדוגמה , בין מחשב לרמקול בלוטות או עכבר בלוטות , או ברכב בין טלפון סלולארי לדיבורית). בשנת 1999 פורסם התקן לבלוטות , פרי שילוב של חברת אריקסון עם IBM , אינטל , נוקיה וטושיבה. התקן בא להחליף את תקן ה RS232 הטורי , שמתבצע עם כבלים. השם בלוטות היא על שמו של האראלד בלאטלנד הראשון מלך דנמרק שאיחד את השבטים הסקנדינביים וכינויו היה כחול השן. פרוטוקול התקשורת ניסה לאחד פרוטוקולים שונים לסטנדרט יחיד כמו שעשה האראלד. הלוגו של בלוטות  הוא שילוב של האות הראשונה של השם הפרטי של המלך ושם המשפחה שלו לפי האלף בית הגרמני וארצות השפלה במאה הראשונה של הספירה.



### 2.א. קצת אלקטרוניקה

תחום התדר של בלוטות הוא תחום חופשי של רשויות תקשורת בעולם (אין צורך באישור או רישיון של רשות כלשהי כדי להשתמש בו) ומשמש בתחומי תעשייה, מחקר, רפואה ומחשבים. הבלוטות משתמש בשיטת **FHSS - Frequency-Hopping Spread Spectrum** שהוא ספקטרום דילוג תדר מפורז. המידע מחולק לחבילות (מנות) כאשר כל חבילה משודרת באחד מתוך 79 ערוצים שונים (מיתוג מנות). לכל ערוץ יש רוחב פס של 1 מגה הרץ . המעבר בין הערוצים הוא כ 1600 פעמים בשנייה. שיטת ה FHSS היא שיטה לפיזור תדר המשדר כך שידלג בקצב גבוה (עד 1600 קפיצות בשנייה) , בין מספר רב של תדרים בצורה אקראית הידועה למשדר ולמקלט. האפנון של המידע היה בהתחלה בשיטת GFSK- Gaussian frequency-shift keying . קצב השידור היה 1 מגה ביט בשנייה ונקרא **Base Rate** – קצב בסיס . המושג **EDR – Enhanced Data Rate** – קצב נתונים מועשר - עובד באפנון  **$\pi/4$ -DQPSK** ו **DPSK8** והחל מגרסה EDR+2.0 מגיעים ל 2 ו 3 מגה ביטים בשנייה. בחודש דצמבר 2014 יצאה גרסה 4.1 . בלוטות הוא פרוטוקול שבו מבנה של עבד – אדון Master-Slave - מסטר סלייב. כל Master יכול להתחבר עם 7 עבדים - Slaves כאשר כל ה Slaves עובדים לפי השעון של המסטר. המסטר הוא זה היוזם את חילוף הנתונים. העבדים לא מתקשרים אחד עם השני. חילוף המנות מבוסס על שעון המסטר שעובד בקצב 312.5

מיקרו שניות. 2 פעימות שעון מהוות חריץ נתונים אחד של 625 מיקרו שניות ( $325 = 1/1600$ ) מיקרו שניות (שניות). במקרה הפשוט של חבילות בחריץ אחד המסטר משדר בזוגות וקולט ביחידים ובהתאמה – העבד משדר ביחידים וקולט בזוגות. חבילה יכולה להיות באורך של 1 או 3 או 5 חריצים אבל בכל מקרה האדון מתחיל את השידור בחריצים זוגיים והעבד צריך לשדר בבודדים. במהלך ההתקשרות יכולים 2 מכשירים להחליף ביניהם תפקידים כעבד ואדון. ניתן לחבר בין מספר פיקונטים יחד. נרשום את היתרונות והחסרונות בטבלה 1:

יתרונות	חסרונות
אפשרות לתקשר עם עד 8 מכשירים במקביל לכל רשת	הפרעות מצד מכשירים אלקטרוניים שפועלים באותו תחום תדרים
הספק חשמל נמוך שמהווה גורם משמעותי בהתקנים ניידים כמו טלפונים סלולריים	רוחב פס נמוך יחסית שמונע אפשרויות של העברת קבצים גדולים וזרימה של וידאו
יכולת אבטחה (הצפנה) טובה	
אין צורך בקו ראייה (לעומת אינפרא אדום)	
חיבור אוטומטי בין המכשירים השונים (לעומת תקן WI-FI)	

טבלה 1 : יתרונות וחסרונות של בלוטות

על מנת לחבר בין המכשירים, צריך אחד הצדדים לערוך סריקה. מציאת מכשירים קרובים בזמן סריקה, תלויה באישור הצד השני, הבוחר בהגדרות המכשיר האם להיות גלוי לכל סורק או לא להופיע כלל.

כאשר התקן Bluetooth אחד נמצא בתחום הפעולה של התקן שני נוצרת תקשורת ביניהם ללא צורך נתינת פקודה או לחיצה על לחצן כלשהו. באופן זה נוצרת רשת תקשורת ללא קשר אם ההתקנים הם מחשבים או ציוד אחר. מערכת Bluetooth יוצרת Personal Area Network (PAN) או פיקונט piconet.

התקשורת בטכנולוגיית Bluetooth מאובטחת באמצעות אלגוריתם הנקרא Safer+ (בטוח יותר) המייצר "מפתח אבטחה". לכן בזמן חיבור בין מכשירים יש צורך להזין סיסמה המכונה בשם "סימת זיווג" או "תאום זוגיות" (pairing). כבירת מחדל "סימת תאום הזוגיות" – pairing password - היא 1234 אבל מומלץ לבחור סיסמה אחרת שתהיה קשה לניחוש.

לכל רכיב יש את מצבי העבודה הבאים :

א. standby שבו הוא מכה להתחבר עם רכיב אחר.

ב. inquire - מחפש רכיבי בלוטות קרובים. זה נקראה גם inquire scan

ג. **page** - מתחבר אל רכיב אחר.

ד. **connect** - מצב של חיבור והעברת נתונים.

ה. **hold and park** - מחובר ברשת פיקונט וממתין לחילופי נתונים (הספק נמוך).

## א.2.1 - מבנה של מנה

כל מנה מכילה שלשה חלקים (או 4 חלקים בבלוטות EDR):

- **Access code** - קוד גישה של 72 ביטים. היחידה הקולטת משתמשת במידע זה לזיהוי שידור נכנס.
- **Header** - כותרת – של 54 ביטים. מגדיר את סוג המנה ואת אורכה.
- **Payload** - "מטען" – הנתונים עצמם שהם בין 0-2745 ביטים.
- בבלוטות EDR קיים חלק רביעי **Inter-Packet Guard Band** – מרווח ביטחון בתוך המנה. המקלט חוזר לתדר הבא.

### א.2.1.א Access Code (קוד גישה) - מיועד לתזמון סנכרון של קוד גישה.

**קוד גישה כללי General Inquiry Access Code - GIAC** הוא קבוע: 0x9E8B33. אם רוצים קוד גישה ספציפי ניתן לקבוע אותו בין 0x9E8B00 עד 0x9E8B3F והוא נקרא **DEDICATED IAC**.

### א.2.1.ב Header (כותרת) - מכיל מידע על המנה, מספר המנה, כתובת המנה ובדיקת שגיאה של הכותרת.

**א.2.1.ג Payload** - מכיל שדה דיבור, שדה נתונים או שניהם יחד. אם הוא מכיל שדה נתונים אזי הוא מכיל גם כותרת נוספת.

באיור הבא מתוארת מנה ברשת פיקונט. השליחה משמאל לימין:

Access Code (72 bits)	Header (54 bits)	Payload (0-2475 bits)
-----------------------	------------------	-----------------------

### א.2.1.ד - פרוטוקולים של בלוטות

קיימים מספר פרוטוקולים התלויים בסוג השימוש של רשת הבלוטות. עבודה עם רכיב HC-05 הוא בפרוטוקול **SPP - Serial Port Profile** - כמו תקשורת טורית רגילה של UART או RS232. דוגמה לפרוטוקול נוסף הוא **HID - Human Interface Device** – רכיב ממשק אנושי כמו עכבר או לוח מקשים של מחשב.

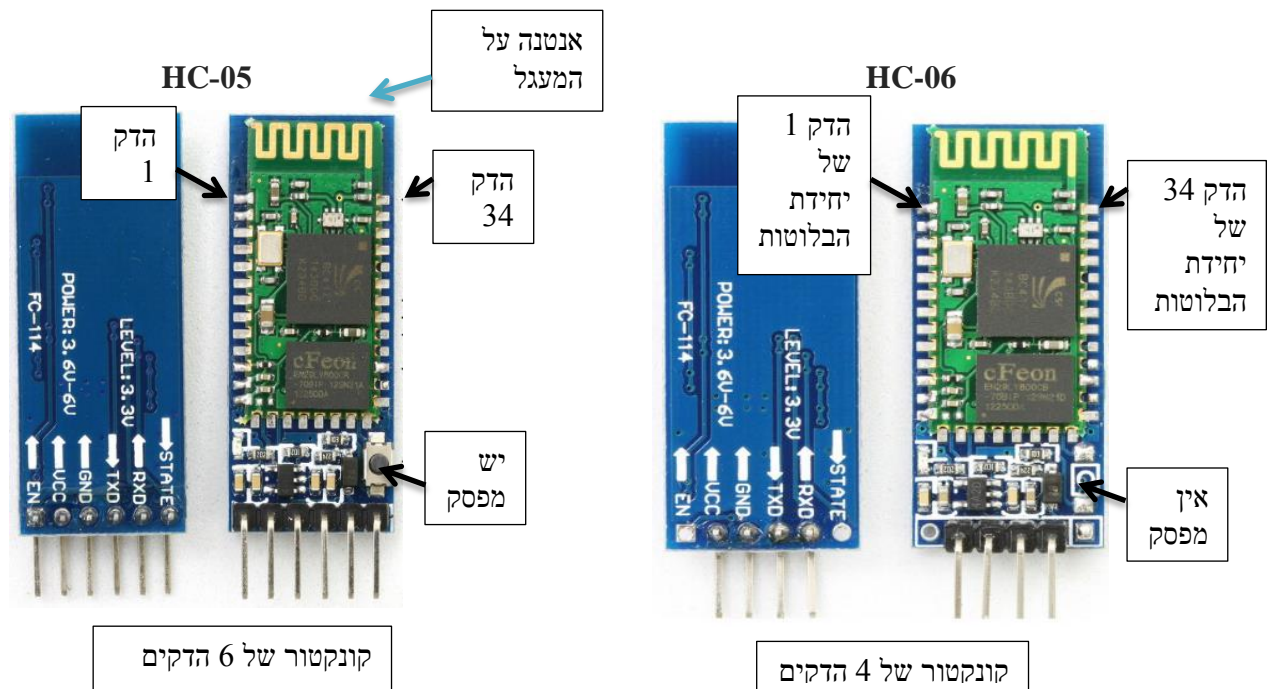
### 3.א - מבוא ליחידות בלוטות HC

מוצרי בלוטות טוריים מסדרת HC מורכבים מ 2 יחידות :

א. מודול ממשק בלוטות טורי

ב. ממתאם בלוטות

מודול ממשק בלוטות הוא החלק הירוק בכל אחת מהיחידות שבאיור 1 הכוללת 2 ג'וקים ואנטנה והמתאם הוא החלק הנוסף עם רכיבי תאום וקונקטור להתחברות.



איור 1 : תמונת כרטיסי HC-05 ו HC-06 כולל ציון ההבדלים ביניהם

כאמור יחידת הבלוטות היא הרכיבים הנמצאים על החלק הירוק כולל האנטנה ויחידת התאום היא כל הרכיבים שמתחת למלכן הירוק כולל ההדקים.

#### 3.1.א - כרטיסי בלוטות HC-05 ו HC-06

ניתן למצוא באינטרנט כרטיסי בלוטות במחיר של שקלים בודדים. הם נראים כמו באיור 1. כדאי לשים לב שלמודול HC-05 יש מחבר של 6 הדקים לעומת HC-06 שלו מחבר של 4 הדקים. כמו כן למודול HC-05 יש מפסק (נראה בחלק הימני התחתון של הכרטיס). ניתן לראות שבכל מודול יש משטח בצבע ירוק ועליו את 2 הרכיבים העיקריים שהם ג'וק הבלוטות (הגבוה יותר וקצת יותר גדול) ומתחתיו זיכרון FLASH של 8 מגה ביט. רואים שליחידה הירוקה שנקרא לה יחידת הבלוטות, יש 34 הדקים המתחברים על יחידה נוספת שנקראת יחידת התאום ובה יש את הקונקטור, מתאמי רמות מתח, מייצב מתח ל 3.3 וולט ולד.

### 3.2.א - מודולי ממשק בלוטות טורי :

תעשיה : Hc-03, HC-04 (HC-04-M, HC-04-S)

אזרחי : HC-05, HC-06 (HC-06-M, HC-06-S)

HC-05-D, HC-06-D ( עם לוח בסיסי לבדיקות והערכה).

ה M וה S מציינים האם המודול עובד כ Master או Slave .

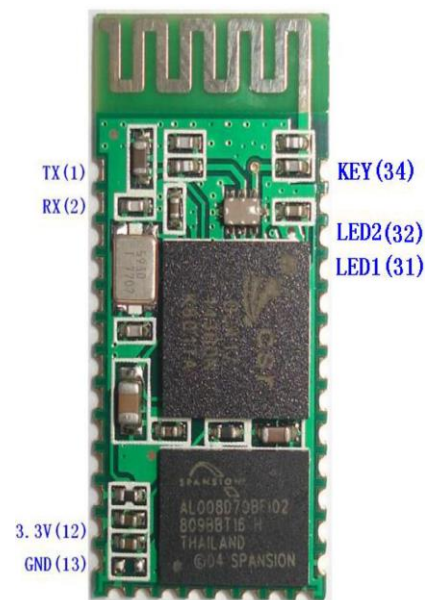
### 3.3.א - מתאמי בלוטות

HC-M4, HC-M6

איור 2 מראה מודול ממשק בלוטות (החלק הימני) ומצד שמאל מופיע כרטיס HC-05 הכולל גם מתאם . הפלסטיק השקוף שבכרטיס נותן הגנה פיזית.

**אזהרה :** על גב המעגל רשום שהדק RX יכול לקבל רמות של 3.3 וולט. יש סכנה שאם נחבר את הדק TX של מיקרו בקר שעובד עם 5 וולט, ייגרם נזק לרכיב. במקרה כזה כדאי לקחת מחלק מתח ביחס של 1:2 ולחבר את הדק RX לנגד עליו נופל 2/3 מהמתח .

אני השתמשתי ביחידות בלוטות HC-05 ו HC-06 שקניתי דרך האינטרנט ולא שמתי מחלק מתח והיחידות עובדות ללא דופי.



איור 2 : מודול בלוטות מימין והמודול עם המתאם בכרטיס HC-05 ( 2 התמונות שמשמאל) .

באיור 2 נראה בצד ימין יחידת בלוטות ללא מתאם. במרכז יש מודול HC05 שיש עליו הגנת פלסטיק מוקשח שקופה. מצד שמאל יש כרטיס HC-05 ללא הגנת הפלסטיק השקוף.

בטבלה 2 מתואר כיצד לחבר את הדקי הכרטיס.

שם ההדק	HC-06	HC-05	לאן ההדק מחובר
STATE	לא קיים	state	לא לחבר. הדק יציאה. כאשר אין התאמה יש בהדק פולסים. כאשר בוצע תאום זוגיות יש בהדק גבוה.
RXD	Serial Receive	Serial Receive	להדק TXD במיקרו בקר
TXD	Serial Transmit	Serial Transmit	להדק RXD במיקרו בקר
GND אדמה	Ground	Ground	
Vcc 5V	Power	Power	
Enable	Wakeup או Key En	Wakeup או Key En	אם נשים את ההדק בגבוה לפני הפעלת חשמל הרכיב ייכנס למצב AT. בחלק מהכרטיסים יש מפסק שניתן ללחוץ עליו לפני שמחברים את הכרטיס לחשמל ואז הכרטיס עובר למצב AT.

טבלה 2 : חיבור ההדקים

נתאר מודול בלוטות טורי המשמש להמרה של פורט טורי לבלוטות. למודול כזה יש 2 אופנים : אדון – מסטר – MASTER, ועבד – SLAVE. הרכיב עם המספר זוגי ( HC-04, HC-06 ) יוצא מבית היצרן כשהוא מתוכנן להיות המסטר או העבד ולא ניתן לשנות זאת. לרכיב עם מספר אי זוגי (HC-05, HC-03, HC-04 ו- HC-06), ניתן לקבוע האם הרכיב יעבוד כעבד או אדון עם פקודות AT. הרכיבים HC-04 ו- HC-06 יוצאים מבית היצרן כאשר ברירת המחדל היא עבד – Slave. אם רוצים מסטר יש לציין זאת בהזמנת המודול.

המודולים HC-03 ו- HC-05 יוצאים מבית היצרן כאשר חלק מהפרמטרים מכוונים להפעלת המודול. אופן העבודה לא נקבע והמשתמש יכול לקבוע בעצמו בעזרת פקודות AT. התפקיד העיקרי של המודול הטורי הוא להחליף את התקשורת הטורית. לדוגמה:

- יש 2 מיקרו בקרים שרוצים להתקשר ביניהם. אחד מתחבר לרכיב המסטר והשני מתחבר לרכיב העבד. רק אחרי ביצוע תאום זוגיות הם יכולים להתחבר ביניהם. חיבור בלוטות אקוויולנטי לפורט טורי עם הדקים RXD ו- TXD והם מתחברים להדקי הבלוטות להתקשרות ביניהם.

2. כאשר למיקרו בקר יש מודול בלוטות מסוג עבד הוא יכול להתחבר למתאם בלוטות של מחשב או טלפון חכם.
3. רוב רכיבי הבלוטות בשוק הם רכיבי עבד, כדוגמת מדפסות, GPS וכו'. ניתן להשתמש ברכיב מסטר להתקשרות איתם.
- כדי להתחבר בין 2 רכיבי בלוטות צריך לפחות 2 תנאים הכרחיים ( והם לא תמיד מספיקים):
1. התקשורת תהיה בין אדון ועבד.
  2. הסיסמה חייבת להיות נכונה.

### א.3.4 - טבלת השוואה בין HC-05 ו HC-06

ל 2 היחידות HC-05 ו HC-06 יש אלקטרוניקה כמעט זהה, נציג טבלת השוואה של הפרמטרים העיקריים של הרכיבים אחרי יציאה מבית היצרן. ההשוואה בטבלה 3 .

HC-05	HC-06
ניתן להחליף בין אופני אדון ועבד	לא ניתן להחליף בין מצבי אדון ועבד
שם הבלוטות: HC-05	שם הבלוטות: linvor
סיסמה: 1234	סיסמה: 1234
אם הוא אדון - לא צריך לזכור את הרכיב האחרון. יכול לבצע תאום זוגיות לכל רכיב עבד(הוא יוצא מבית היצרן AT+COMMAND=1). אם נרשום AT+COMMAND=0 הוא יבצע תאום זוגיות רק עם רכיב העבד האחרון.	אם הוא אדון - חייב לזכור את רכיב העבד האחרון שעשה אתו התאמה ורק איתו עושה תאום זוגיות, אלא אם הדק KEY (הדק 26) מותנע על ידי רמה של '1'. ברירת המחדל בהדק זה היא '0'.
רכיב המסטר יכול לא רק ליצור תאום זוגיות עם רכיב בלוטות עם כתובת מסוימת כמו טלפון סלולארי מתאם מחשב ורכיב עבד, אלא יכול לחפש וליצור תאום זוגיות עם עבד אוטומטית. בתנאים מסוימים האדון והעבד יכולים לבצע תאום זוגיות אוטומטית (זוהי ברירת המחדל).	בעבודה כמסטר, הוא מחפש את העבד ועושה תאום זוגיות אוטומטית. בתנאים מסוימים האדון והעבד יכולים לבצע תאום זוגיות אוטומטית.
התקשרות עם מספר רכיבים: יש תקשורת קר בין נקודה לנקודה בין מודולים אבל מתאם יכול להתקשר עם מספר מודולים	התקשרות עם מספר רכיבים: יש תקשורת קר בין נקודה לנקודה בין מודולים אבל מתאם יכול להתקשר עם מספר מודולים
באופן 1 של AT: אחרי הדלקת חשמל, ניתן להיכנס למצב AT על ידי התנעת PIN34 ברמה גבוהה. אז קצב	אופן AT: לפני תאום זוגיות הוא נמצא במצב AT. אחרי התאום זוגיות יש תקשורת רגילה.

התקשורת לפקודות AT שווה לקצב התקשורת (לדוגמה 9600). באופן 2 של AT: יש לשים ב PIN34 רמה גבוהה וקצב התקשורת בשימוש הוא 38400 ביטים בשנייה. יש לשים לב שכל פקודות ה AT מבוצעות כאשר PIN34 ברמה גבוהה. רק חלק מפקודות ה AT יתבצעו אם נכנסים למצב AT והרמה ב PIN34 לא נשמרת בגבוה. במודול יש הדק גישה נוח להתחברות ל PIN34.	
בזמן תהליך תקשורת הרכיב איננו יכול להיכנס למצב AT. ל 0 המודול יכול לחזור למצב ההתקשורת.	בזמן תהליך תקשורת הרכיב איננו יכול להיכנס למצב AT.
קצב תקשורת ברירת המחזל : 9600, 4800-1.3M	קצב תקשורת ברירת המחזל : 9600, 1200-1.3M
KEY : PIN34 - להיכנס למצב AT.	KEY : PIN 26 - להזניח את זיכרון המסטר.
LED1 : PIN31 מראה את אופן העבודה. קצב נמוך של 1 הרץ מציין כניסה למצב AT אופן 2. קצב גבוה יותר של 2 הרץ מציין כניסה למצב AT אופן 1 או מצב של תקשורת תאום זוגיות. 2 הבהובים בשנייה מציינים גם שהתאום זוגיות בוצעה וניתן להתקשר עם הרכיב. LED2 : PIN32 - לפני התאום זוגיות יש בו רמה נמוכה ואחרי התאום זוגיות רמה גבוהה.	LED : קצב ההבהוב של רכיב עבד הוא 102 מילי שניות. אם לרכיב מסטר יש את כתובת האדון אז בתאום זוגיות ביניהם קצב ההבהוב הוא 110 אלפיות שנייה. אם לא – או המסטר רוקן את הזיכרון שלו, קצב ההבהוב יהיה 750 מילי שניות. אחרי תאום זוגיות, לא משנה אם זה מסטר או עבד, הדק ה LED הוא בגבוה.
תצרוכת הספק: בזמן התאום זוגיות הזרם קופץ בתחום של 30-40 מילי אמפר. הזרם הממוצע 25 מילי אמפר. אחרי התאום זוגיות יש זרם של 8 מילי אמפר. אין מצב sleep – שינה.	תצרוכת הספק: בזמן התאום זוגיות הזרם קופץ בתחום של 30-40 מילי אמפר. הזרם הממוצע 25 מילי אמפר. אחרי התאום זוגיות יש זרם של 8 מילי אמפר. אין מצב sleep – שינה.
RESET : PIN11 – פעיל בנמוך.	RESET : PIN11 – פעיל בנמוך.
רמה - אזרחית	רמה - אזרחית

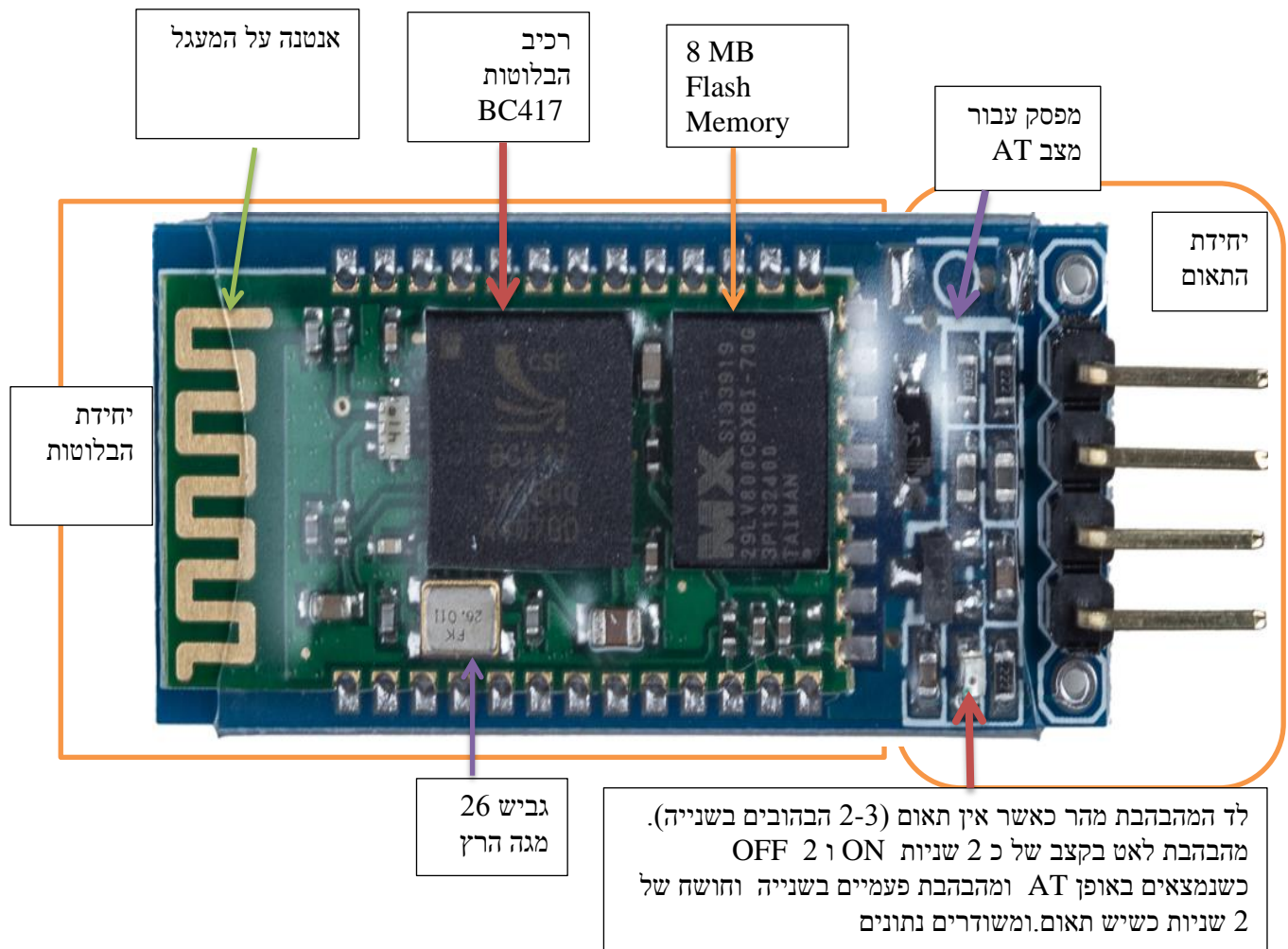
טבלה 3 – טבלת השוואה בין HC-05 ו HC-06



## ב. מודול שידור קליטה בלוטות HC-05 – ו HC-06 - הרחבה

### **1.ב - כרטיסי בלוטות HC-06 ו HC-05 - הרכיבים בכרטיס**

ניתן למצוא באינטרנט כרטיסי בלוטות במחיר של שקלים בודדים. הם נראים כמו באיור 1. כדאי לשים לב שלמודול HC-05 יש מחבר של 6 הדקים לעומת HC-06 שלו מחבר של 4 הדקים. כמו כן למודול HC-05 יש מפסק (נראה בחלק הימני התחתון של הכרטיס). ניתן לראות שבכל מודול יש משטח בצבע ירוק ועליו את 2 הרכיבים העיקריים שהם ג'וק הבלוטות (הגבוה יותר וקצת יותר גדול) ומתחתיו זיכרון FLASH של 8 מגה ביט. רואים שליחידה הירוקה שנקרא לה יחידת הבלוטות, יש 34 הדקים המתחברים על יחידה נוספת שבה יש את הקונקטור, מתאמי רמות מתח, מייצב מתח ל 3.3 וולט ולד. באיור 3 מתוארים 2 מרכיבי הכרטיס והרכיבים שבכרטיס HC-06.



איור 3 - מודול HC-06 : חלוקת הכרטיס ל 2 יחידות והרכיבים על הכרטיס.

## 2. – מאפיינים טכניים

המודולים HC-05 או HC-06 הוא מודול בלוטות SPP ( Serial Port Protocol) - פרוטוקול פורט טורי - קל לשימוש שתוכנן להעברת תקשורת אלחוטית טורית. המודול הוא V2.0+EDR (Enhanced Data Rate) בקצב של 3 מגה ביט בשנייה בתדר של 2.4 ג'יגה הרץ. יש בו רכיב בטכנולוגית CMOS ( צריכת הספק נמוכה ) שנקרא Bluecore 04 - CSR BC417 - ובתכונת דילוג תדר מסתגל Adaptive Frequency Hopping Feature - AFH . הגודל שלו קטן 12.7\*27 mm

## 3. מאפייני חומרה

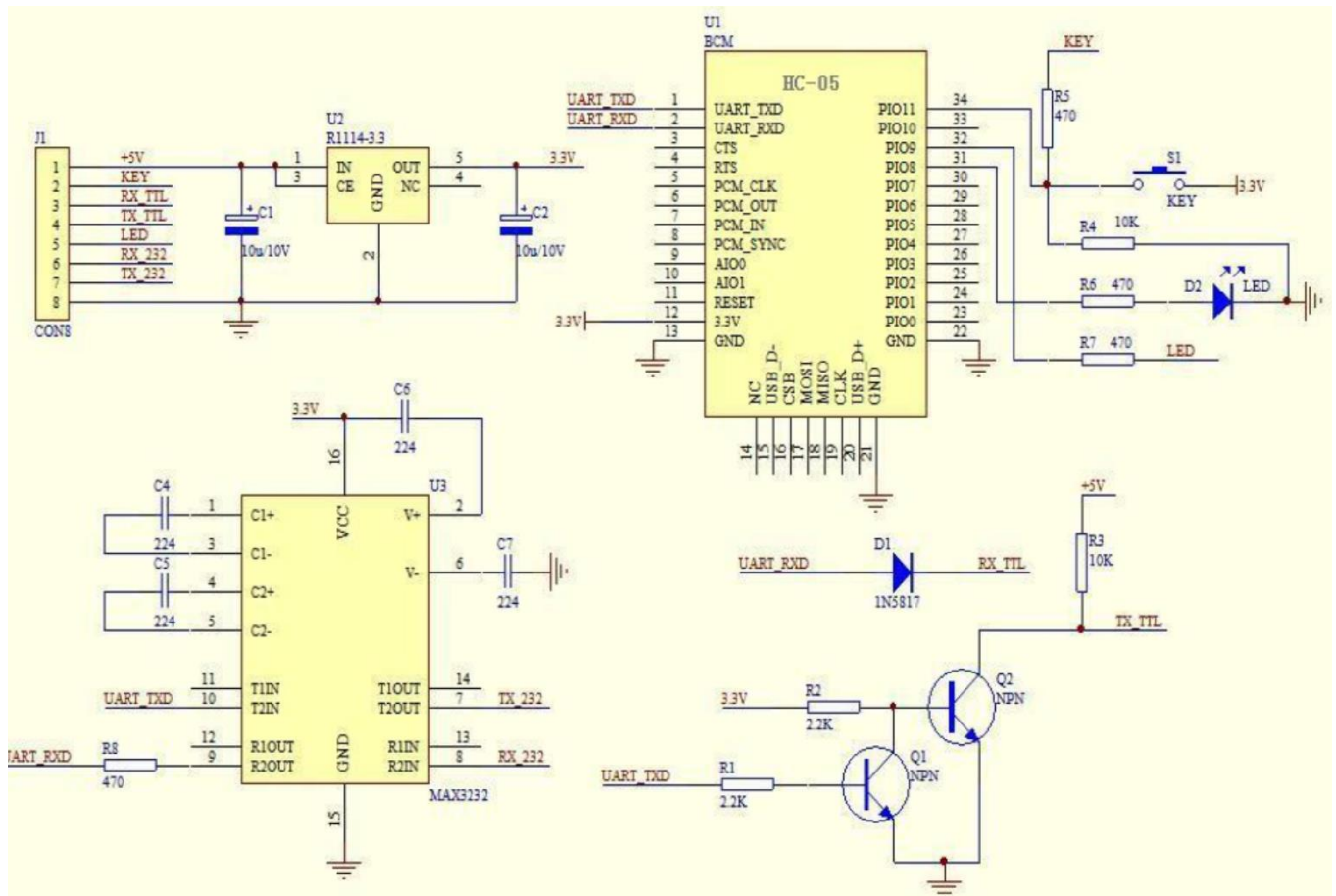
- \* רגישות -85dBm .
- \* הספק שידור RF של עד +40dBm .
- \* אנטנה על הכרטיס עצמו.
- \* פעולה מ 1.8 ועד 3.6 וולט.
- \* כרטיס FLASH של 8 מגה ביט על הכרטיס.
- \* ממשק UART עם קצב תקשורת שניתן לתכנות.
- \* . מידות 26.9 מ"מ אורך, 13 מ"מ רוחב ו 2.2 מ"מ גובה.

## 4. מאפייני תוכנה

- \* התקשורת הטורית : קצב תקשורת ברירת המחדל היא 38400, 8 ביט, 1 ביט סיום, ללא זוגיות. ניתן לתכנת את קצב התקשורת ל 9600, 19200, 38400, 57600, 115200, 230400, 460800 .
- \* אם ניתן פולס עולה מ 1 ל 0 ב PIO0 הרכיב מתנתק.
- \* PIO1 הוראת סטאטוס : 0 – מנותק. 1 – מחובר .
- \* PIO10 ו PIO11 יכולים להתחבר בנפרד ללד האדומה וללד הכחולה. כאשר יש תאום זוגיות בין עבד
- לאדון הלד האדומה דולקת פעם אחת בכל 2 שניות וכאשר אין תאום זוגיות הלד הכחולה דולקת כל חצי שנייה.
- \* חיבור אוטומטי לרכיב האחרון בהפעלת חשמל.
- \* ברירת המחדל של קוד התאום זוגיות הוא "0000" .
- \* חיבור חוזר אוטומטי בתוך 30 דקות כאשר יש ניתוק כתוצאה של טווח חיבור גדול יותר.

## ג. – סכימה חשמלית של מודול הבלוטות

באיור 4 מתוארת הסכימה החשמלית של המודול.



איור 4 : סכימה חשמלית של יחידת הבלוטות (ללא זיכרון ה FLASH)  
 בצד שמאל למעלה רואים את ספק הכוח המקבל מתח של 5 וולט דרך הדק מספר 1 ומייצב אותו ל 3.3 וולט בעזרת U2.

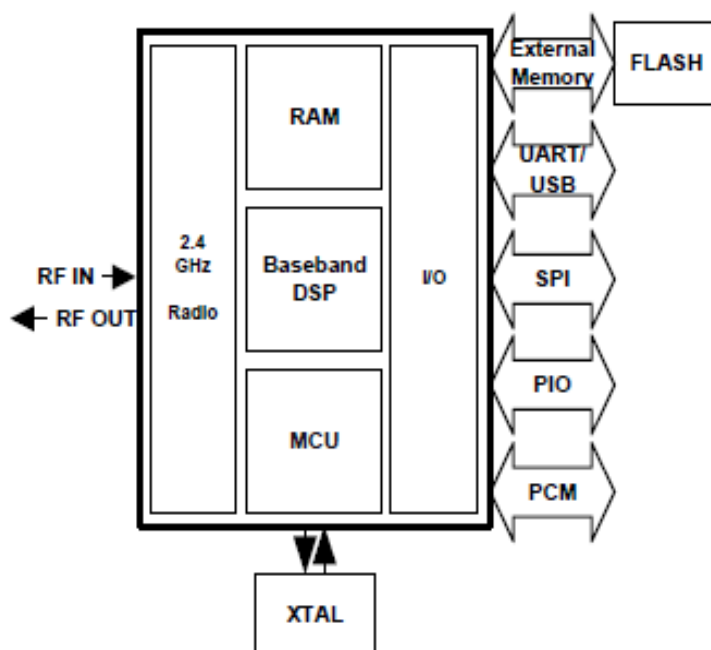
למעלה נראה הרכיב של הבלוטות. איור 4 שבהמשך, מתאר את הארכיטקטורה של רכיב הבלוטות

BC417 של חברת Cambridge Silicon Radio והוא יוסבר בהמשך.

הדקים 1 ו 2 של הרכיב הם הדקי התקשורת הטורית UART כאשר הדק 1 של הרכיב הוא השידור הטורי TXD והדק 2 הקליטה RXD. למטה משמאל נראה את רכיב התקשורת max232 המתאם רמות מתח של RS232 לרמות TTL. מימינו יש 2 טרנזיסטורים המתאמים את רמת המתח של השידור מ 3.3 וולט לרמות של TTL. כאשר יש 0 בבסיס Q1 מהדק TxD של הרכיב HC-05 אז Q1 בקטעון ולכן Q2 ברוויה ובהדק TX\_TTL יש '0'. כאשר יש '1' בבסיס Q1 מהדק TxD של uart אז Q1 ברוויה ולכן Q2 בקטעון ובהדק TX\_TTL יש '1' לוגי ברמה של 5 וולט שהיא רמת TTL.

## 7. הארכיטקטורה של רכיב BC417

באיור 5 מתוארת הארכיטקטורה של רכיב הבלוטות.



איור 5 – ארכיטקטורה פנימית של רכיב הבלוטות BC417

נתאר את המלבנים העיקריים:

**MCU - Micro Controller Unit** - יחידת מיקרו בקר המנהלת את כל התהליכים במערכת. ביחידה זו קיים זיכרון תכנית פנימי (לא נראה באיור) וזיכרון נתונים RAM.

**XTAL - Crystal** - גביש. כאן מחובר גביש בתדר 26MHz הקובע את תדר העבודה של ה-MCU.

**Baseband DSP** – ( DSP-Digital Signal Processing - עיבוד אות ספרתי ). היחידה הזו ביחד עם יחידת ה-MCU קובעות את האפנון ואת התדר שישודרו/ייקלטו.

**2.4GHz Radio** - זוהי יחידת השידור של תדר הבלוטות.

**I/O** - קלט פלט. הדקי קלט או פלט "רגילים" או הדקים המבצעים ממשק לסוגי תקשורת טורית שונים. בחלק הימני עליון ניתן לראות הדקי קלט פלט להתחברות לזיכרון FLASH חיצוני.

**UART/USB** - הדקים לתקשורת טורית רגילה או USB .

**SPI - Serial Peripheral Interface** - ממשק טורי היקפי – סוג נוסף של תקשורת טורית.

**PIO - Programmable Input Output** - קלט פלט מתוכנת - שיטה נוספת להעברת נתונים בין מעבד ורכיב היקפי.

**PCM - Pulse Code Modulation** - אפנון קוד פולס שהיא שיטת ייצוג דיגיטאלית של אות אנאלוגי.

סט פקודות AT אפשר למצוא באתר :

[http://www.sgbotic.com/products/datasheets/wireless/HC05\\_AT\\_Command.pdf](http://www.sgbotic.com/products/datasheets/wireless/HC05_AT_Command.pdf)

הכרטיס HC-06 יכול לקבל רק חלק קטן מהפקודות . הפקודות מתאימות ל HC-05 .

## ה. פקודות בסיסיות באופן AT

ניתן להכניס פקודות AT כדי לקבוע את הקונפיגורציה (תצורה) . כל פקודת AT מסתיימת עם <CR> ו <LF> ( שהם <ENTER> שהוא שילוב של 0Dh ואחריו 0Ah ) . כל פקודת AT נענית בתגובת "OK" של הכרטיס.

כאשר נרצה לקבל תשובה לגבי חלק מהמאפיינים רושמים את המאפיין עם סימן שאלה ( ? ) בסוף המאפיין ואז מקבלים את המצב האחרון שהשתמש קבע לגבי המאפיין הזה.

ניתן לרשום באותיות גדולות או קטנות. נרשום מספר פקודות בסיסיות שכדאי לדעת.

### 1. AT

נוהגים לבדוק האם ניתן ליצור קשר עם הרכיב . התשובה המתקבלת היא OK

### 2. AT+RESET

איפוס המודול וחזרה לנתוני מצב ברירת המחדל ( default ) שלו וקצב התקשורת הוא 38400 למרות שהוא נשלח מבית היצרן עם קצב של 115200 .

### 3. AT+ROLE?

שואלים את המודול האם הוא מתפקד כמסטר או עבד . הרכיב מחזיר 0 אם עובד כעבד או 1 אם עובד כאדון . יכול להחזיר 2 אם עובד כעבד ב loopback (לבדוק תקשורת חוזרת).

#### 4. AT+ROLE=<פרמטר>

קביעת מצב העבודה של הרכיב. הפרמטר יכול להיות 0 – עבד או 1 – אדון או 2 עבודה כעבד ב loopback. כדי ליצור תקשורת בין 2 יחידות יש לקבוע יחידה אחת כאדון ואת השנייה כעבד כדי לבצע תאום זוגיות. אם שמנו '0' אז הוא מחכה שרכיב מסטר יתקשר אליו. אם שמים '1' אז הוא המסטר ומחפש את הרכיב שאליו יתחבר. אם שמים '2' אז הרכיב הוא עבד וכל מה שהוא קולט הוא משדר בחזרה.

#### 5. AT+CMODE?

שואלים את המודול האם הוא מתקשר עם כתובת בלוטות מסוימת? (התשובה מהרכיב היא 0) או עם כל כתובת בלוטות? (התשובה מהרכיב היא 1) או כעבד ב loopback? (התשובה מהרכיב היא 2).

#### 6. AT+CMODE = <פרמטר>

קובעים למודול האם הקשר הוא עם כתובת בלוטות מסוימת ( הפרמטר הנשלח 0 ), או כל כתובת בלוטות ( הפרמטר הנשלח הוא 1) או כעבד ב loopback (שולחים לו 2).

#### 7. AT+PSWD?

שואלים את המודול מה הסיסמה שלו.

#### 8. AT+PSWD = <פרמטר>

קובעים למודול את סיסמת התאמת הזוגיות.

#### 9. AT+UART?

שואלים את המודול מהי תצורת ה UART שלו. התשובה תהיה בפורמט: קצב תקשורת, כמות ביטים של סיום והאם יש זוגיות ואיזה זוגיות

#### 10. AT+UART=<פרמטר1>,<פרמטר2>,<פרמטר3>

קביעת תצורת התקשורת הטורית. פרמטר 1 הוא קצב התקשורת הטורית. הפרמטר השני – כמות ביטים של סיום. כשנרשום 0- יש ביט סיום 1. כשנרשום 1 – 2 ביטים של סיום. פרמטר 3 הוא לגבי ביט הזוגיות. 0 – ללא ביט זוגיות. 1 – זוגיות אי זוגית – ODD, 2 – זוגיות זוגית – EVEN.  
ברירת המחדל: 9600,0,0 כלומר 9600 ביטים בשנייה, סיבית סיום אחת וללא ביט זוגיות.

## 11. AT+ORGL

חזור למצב ברירת המחל default שיצאת מבית החרושת. מצב עבד, קצב תקשורת 38400, ביט סיום 1 ללא ביט זוגיות, סיסמה – 1234, Inquiry code: 0x009e8b33, שהוא קוד כללי ויכול לגלות או להתגלות לכל רכיב בלוטות. (ניתן לתת קוד "מוגבל" עבור התקשרות מהירה ספציפית). הרכיב יוצא מהמפעל עם הנתונים הבאים:

1. Device class: 0
2. Inquiry code: 0x009e8b33
3. Device mode: Slave mode
4. Binding mode: SPP
5. Serial port: 38400 bits/s; 1 stop bit, no parity
6. Pairing code: "1234"
7. Device name: "HHW-SPP-1800-2"

## 12. AT+PAIR=<פרמטר2>,<פרמטר1>

קביעת הרכיב לביצוע תאום זוגיות. הפרמטר הראשון הוא מספר הרכיב לתאום הזוגיות והפרמטר השני הוא פסק הזמן לביצוע התאום, כלומר כמה זמן לנסות לבצע את תאום הזוגיות. נניח שקיים רכיב שמספרו : 12:34:56:ab:cd:ef לכן נרשום: at+pair=1234,56,abcdef,20\r\n (יש לשים לב לאופן הרישום : 4 ספרות אחר כך 2 ספרות ואחר כך 6 ספרות. המספר 20 הוא הזמן לביצוע תאום זוגיות. הרכיב מחזיר OK או FAIL.

## 13. AT+LINK=<פרמטר>

קביעת הרכיב להתקשרות. הפרמטר הוא הכתובת של הרכיב אליו יש להתקשר כהמשך לרכיב שאליו התקשרנו בסעיף הקודם. לדוגמה:

AT+LINK=1234,56,abcd,ef\r\n

הרכיב מחזיר OK או FAIL (או שלא עונה או שמחזיר מספר שגיאה). הפקודה אומרת לרכיב שהוא ברשימה ולא צריך לחקור אותו וניתן להתחבר אליו ישירות.

## AT+BIND = < כתובת הרכיב >

לדוגמה:

AT+BIND=1234,56,abcdef\r\n

הפקודה שימושית רק במצב עבודה של כתובת ספציפית.

### AT+IAC= 9e8b33

ניתן לקבוע את הקוד של המודול שבו יגלה או יתגלה עם רכיבי בלוטות קרובים. ניתן לקבוע קוד גישה כללי GIAC – General Inquire Access Code ואז קוד הגישה הכללי הוא 9e8b33. ניתן לקבוע קוד גישה לא כללי - LIAC כמו 928b3f ואז הרכיב יתגלה מהר יותר.

### AT+CLASS= < פרמטר >

קובעים את סוג הרכיב והשרות שתומך בו. למעשה קובעים האם לסנן ביעילות את הרכיב הקרוב ולמצוא את רכיב הבלוטות שהשתמש קבע בסעיף הקודם של AT+IAC. הפרמטר הוא 32 ביטים (את המספר כותבים בהקסה דצימלי). אם רושמים 0 (שזה גם ברירת המחדל) זה אומר שאין רכיב ספציפי ובודקים את כל רכיבי הבלוטות.

### AT+INQM= < פרמטר 3 > , < פרמטר 2 > , < פרמטר 1 >

הפרמטר הראשון קובע איזו צורת תשאול יבצע רכיב הבלוטות. אם רושמים 0 אז תשאול סטנדרטי. אם רושמים 1 אז התשאול נקרא rssi - Received Signal Strength Indication – מציין את עצמת אות הקליטה באנטנה. המספר הוא באחוזים. ככל שהמספר גדול יותר האות שנקלט חזק יותר. הפרמטר השני הוא max response number - כמות רכיבי הבלוטות שנתחבר אליהם. הפרמטר השלישי הוא זמן התשאול בתאום הזוגיות. זהו זמן בין 1.28 שניות עד 61.44 שניות. רושמים מספר ואת המספר מכפילים ב 1.28 שנייה. לדוגמה: אם נרשום 48 אז זמן התשאול עד לביצוע תאום זוגיות יהיה  $48 * 1.28 = 61.44$  שניות.

לדוגמה: `at+inqm=1,9,48\r\n`

אומר: Inquire mode: RSSI, max number 9, timeout 48

את סוגי הבלוטות השונים וכתובתם מכניסים בעזרת פקודות AT+INQ

### AT+INQM= < פרמטר 3 > , < פרמטר 2 > , < פרמטר 1 >

הפרמטר הראשון הוא הכתובת. הפרמטר השני הוא ה class של הרכיב והשלישי הוא RSSI.



בטבלה הבאה מצוינים מספרי שגיאה עם הסבר מה אומרת כל שגיאה.

#### ERROR code decoder

Error_code (hex)	Explanation
0	AT command error
1	The result is default value
2	PSKEY write error
3	Device name is too long (more than 32 bytes)
4	Device name is 0 byte
5	Bluetooth address: NAP is too long
6	Bluetooth address: UAP is too long
7	Bluetooth address: LAP is too long
8	PIO port mask length is 0
9	Invalid PIO port
A	Device class is 0 byte
B	Device class is too long
C	Inquire Access Code length is 0
D	Inquire Access Code is too long
E	Invalid Inquire Access Code
F	Pairing password is 0
10	Pairing password is too long (more than 16 bytes)
11	Role of module is invalid
12	Baud rate is invalid
13	Stop bit is invalid
14	Parity bit is invalid
15	No device in the pairing list
16	SPP is not initialized
17	SPP is repeatedly initialized
18	Invalid inquiry mode
19	Inquiry timeout
1A	Address is 0
1B	Invalid security mode
1C	Invalid encryption mode

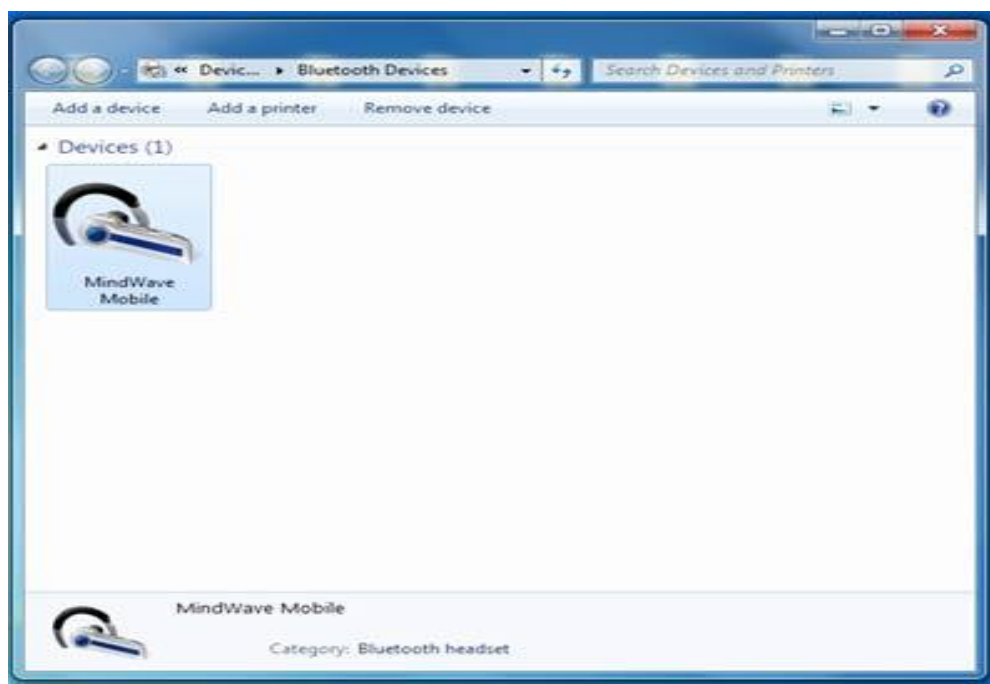
## 1. חיבור HC-05 אל MINDWAVE של חברת Neurosky

קורא גלי המוח MindWave של חברת Neurosky קורא את גלי המוח שאנו משדרים ושולח אותם בתקשורת בלוטות. רכיב הבלוטות בכרטיס קורא המחשבות משמש כעבד – Slave. מכאן שאת רכיב ה HC-05 שיקלוט אותו יש לתכנת כאדון – Master. לא נוכל להשתמש בכרטיס HC-06 כי הוא מגיע כ slave גם הוא ולכן לא יוכל "לדבר" עם קורא המחשבות.

ה HC-05 יוצא מבית החרושת כ slave ולפני שמחברים אותו לקורא המחשבות יש לתכנת אותו לעבוד כ master ולהגיד לו לאיזה רכיב בלוטות הוא צריך להתחבר. כדי לתכנת את ה HC-05 יש להעביר אותו לאופן שנקרא AT. באופן זה שולחים אליו פקודות הקובעות את צורת העבודה שלו- אדון או עבד - ועם אילו רכיבי בלוטות הוא יתחבר. לכל רכיב בלוטות יש קוד זיהוי id code או מזהה ייחודי שלו.

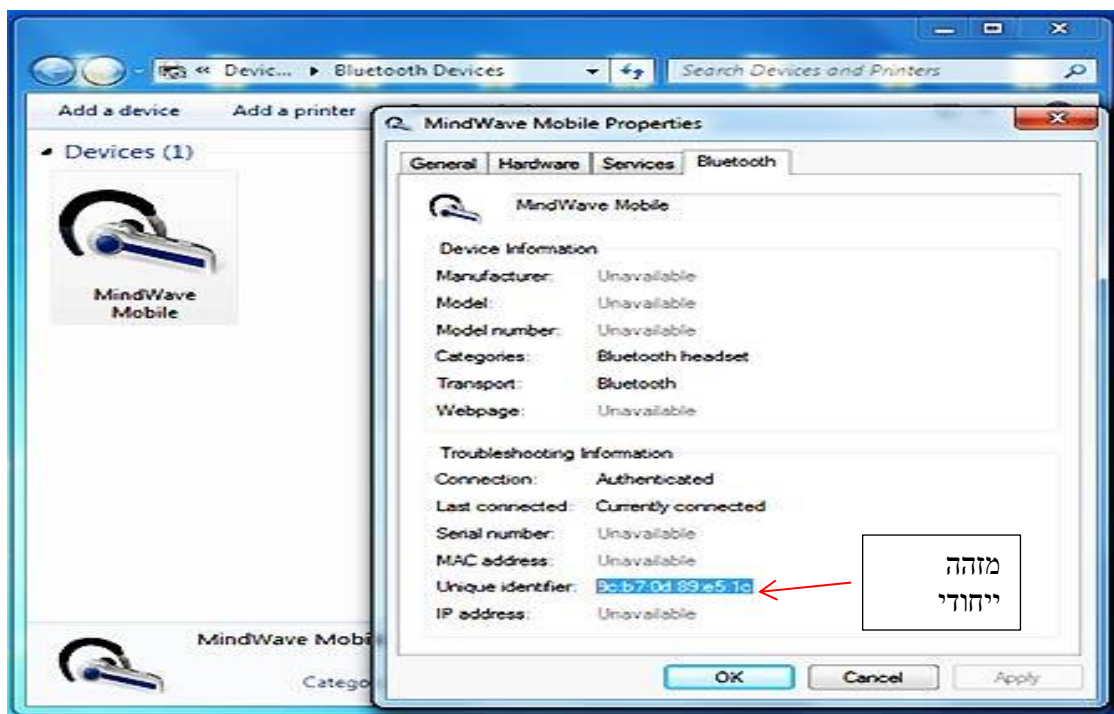
כדי לדעת מהו קוד הזיהוי של הבלוטות בכרטיס קורא המחשבות ניתן לעשות זאת במספר דרכים. חלק מהאפשרויות הן עם המחשב האישי וחלקן עם אחת מאפליקציות הבלוטות של הטלפון הסלולארי שניתן להוריד בחינם. נתאר אחת מהאפשרויות להתחברות עם המחשב האישי. נבצע את המהלכים הבאים:

ניכנס ללוח הבקרה <--- התקנים ומדפסות <-- ונקבל מסך עם ההתקנים המתחברים למחשב שלנו. נפעיל את קורא המחשבות ונעביר למספר שניות את המפסק למצב העליון שלו עד שיתבצע תאום זוגיות עם מערכת הבלוטות של המחשב ונקבל תמונה הדומה לתמונה שבאיור 6:



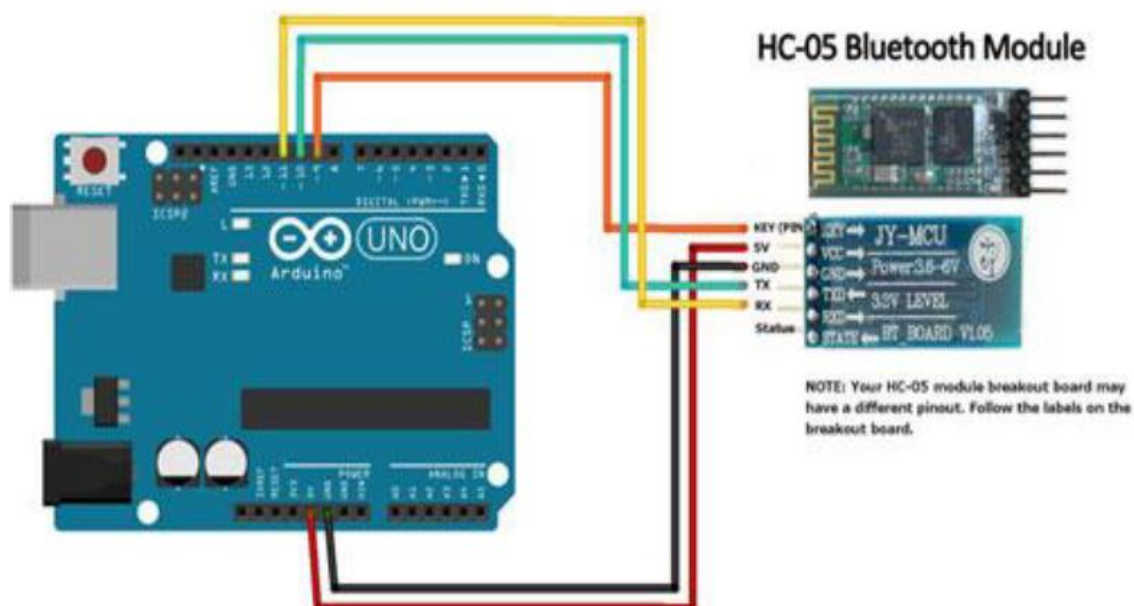
איור 6 : תמונת Mindwave Mobile במסך התקנים ומדפסות במחשב האישי.

נקליק על תמונת ה mindave פעמיים ואז בחלון שנפתח נלחץ על התווית Bluetooth ונקבל מסך ושם נראה את המספר המזהה הייחודי כמו בתמונה הבאה:



איור 7 : מסך הבלוטות והמזהה הייחודי.

במכשיר קורא המחשבות שלנו המזהה הייחודי הוא : 20:68:9d:4c:0c:d8 . אחרי שיוודעים את המזהה הייחודי unique identifier שלנו נוכל לעבור ולתכנת את רכיב הבלוטות HC-05 . לשם כך נחבר את הכרטיס אל ארדואינו לפי השרטוט הבא :

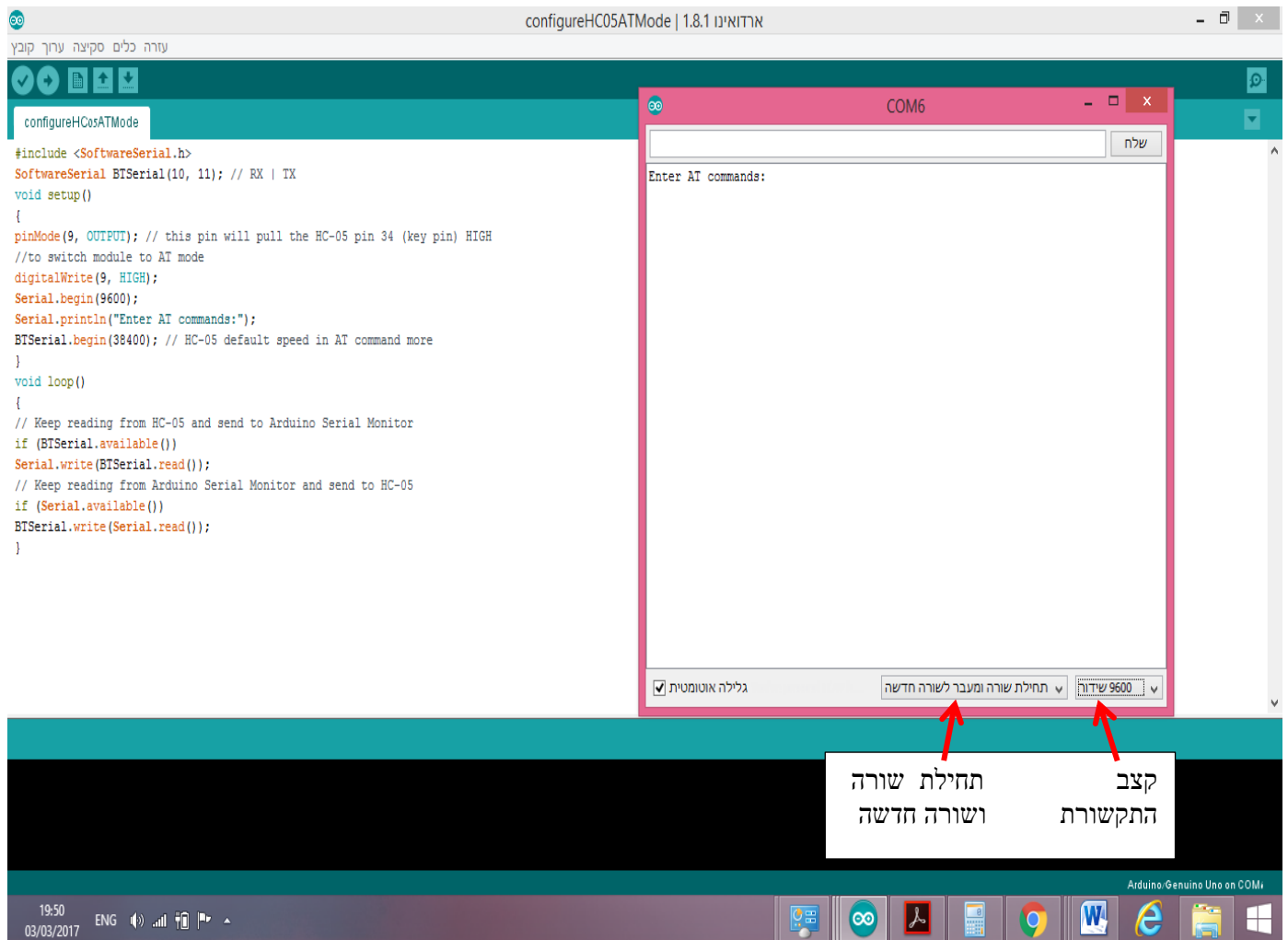


איור 8 : סכמת החיבורים להפעלת ה HC-05 באופן AT .

נרשום את התכנית הבאה ונוריד אותה אל כרטיס הארדואינו.

```
#include <SoftwareSerial.h>
SoftwareSerial BTSerial(10, 11); // RX | TX
void setup()
{
  pinMode(9, OUTPUT); // this pin will pull the HC-05 pin 34 (key pin) HIGH
  //to switch module to AT mode
  digitalWrite(9, HIGH);
  Serial.begin(9600);
  Serial.println("Enter AT commands:");
  BTSerial.begin(38400); // HC-05 default speed in AT command mode
}
void loop()
{
  // Keep reading from HC-05 and send to Arduino Serial Monitor
  if (BTSerial.available())
    Serial.write(BTSerial.read());
  // Keep reading from Arduino Serial Monitor and send to HC-05
  if (Serial.available())
    BTSerial.write(Serial.read());
}
```

כאשר התוכנית נמצאת בכרטיס הארדואינו , נראה שהלד בכרטיס הבלוטות מהבהבת בקצב גבוה של כ 1.5 פעמים בשנייה. במצב זה הכרטיס לא עובד במצב AT וגם לא נמצא במצב של תאום זוגיות. כדי להיכנס למצב AT יש להוציא את חוט ה 5 וולט המתחבר אל כרטיס HC-05 , ללחוץ על המפסק בכרטיס הבלוטות ותוך כדי הלחיצה לחבר את חוט ה 5 וולט . ניתן להפסיק את הלחיצה רק אחרי שהלד שינתה את מהירות ההבהוב שלה למהירות נמוכה יותר. עכשיו מפעילים את תוכנת המוניטור הטורי. יש לכוון בחלונות של קצב התקשורת לקצב 9600 ביטים בשנייה ולקבוע בחלונות שמשמאל לקצב התקשורת : "תחילת שורה חדשה ומעבר לשורה חדשה " או באנגלית Both NL & CR (העזר באיור 9 שבעמוד הבא):



איור : קביעת קצב התקשורת ותחילת שורה ושורה חדשה במסך המוניטור הטורי.

כעת נרשום את הפקודות הבאות (אפשר באותיות גדולות או קטנות) כשמודול ה HC-05 עונה OK:  
תחילה ניתן שם לבלוטות.

AT+NAME = Arik

עכשיו נקבע את קצב התקשורת ל 57600 , ה '0' הראשון אומר שיש ביט סיום אחד (אם היינו רושמים '1' היו 2 ביטים של סיום), הביט '0' הבא קובע שאין בדיקת זוגיות.

AT+UART=57600,0,0

נקבע את הרכיב כאדון – MASTER :

AT+ROLE= 1

נקבע את סיסמת ההתקשרות ל neurosky mindwave ל 0000 :

AT+PSWD=0000

נקבע למודול שהקשר הוא עם כתובת בלוטות מסוימת ( הפרמטר שנשלח 0 ) (המצבים האחרים הם כל כתובת בלוטות - הפרמטר הנשלח הוא 1 - או כעבד ב loopback - שולחים לו 2 ).

AT+CMODE=0

נקבע עם איזו כתובת ספציפית הוא יתחבר (במקרה שלנו את המזהה הייחודי של ה mindwave). יש לשים לב שרושמים 4 תווים, פסיק, 2 תווים, פסיק ולבסוף 6 תווים.

AT+BIND=2068,9d,4c0cd8

נקבע קוד גישה כללי GIAC – General Inquire Access Code והוא 9e8b33

AT+IAC=9E8B33

נקבע שבודקים את כל סוגי רכיבי הבלוטות.

AT+CLASS=0

נקבע מצב עבודה RSSI עם עד 9 רכיבי בלוטות וה timeout יהיה  $1.28 * 48 = 61.44$  שניות.

AT+INQM=1,9,48