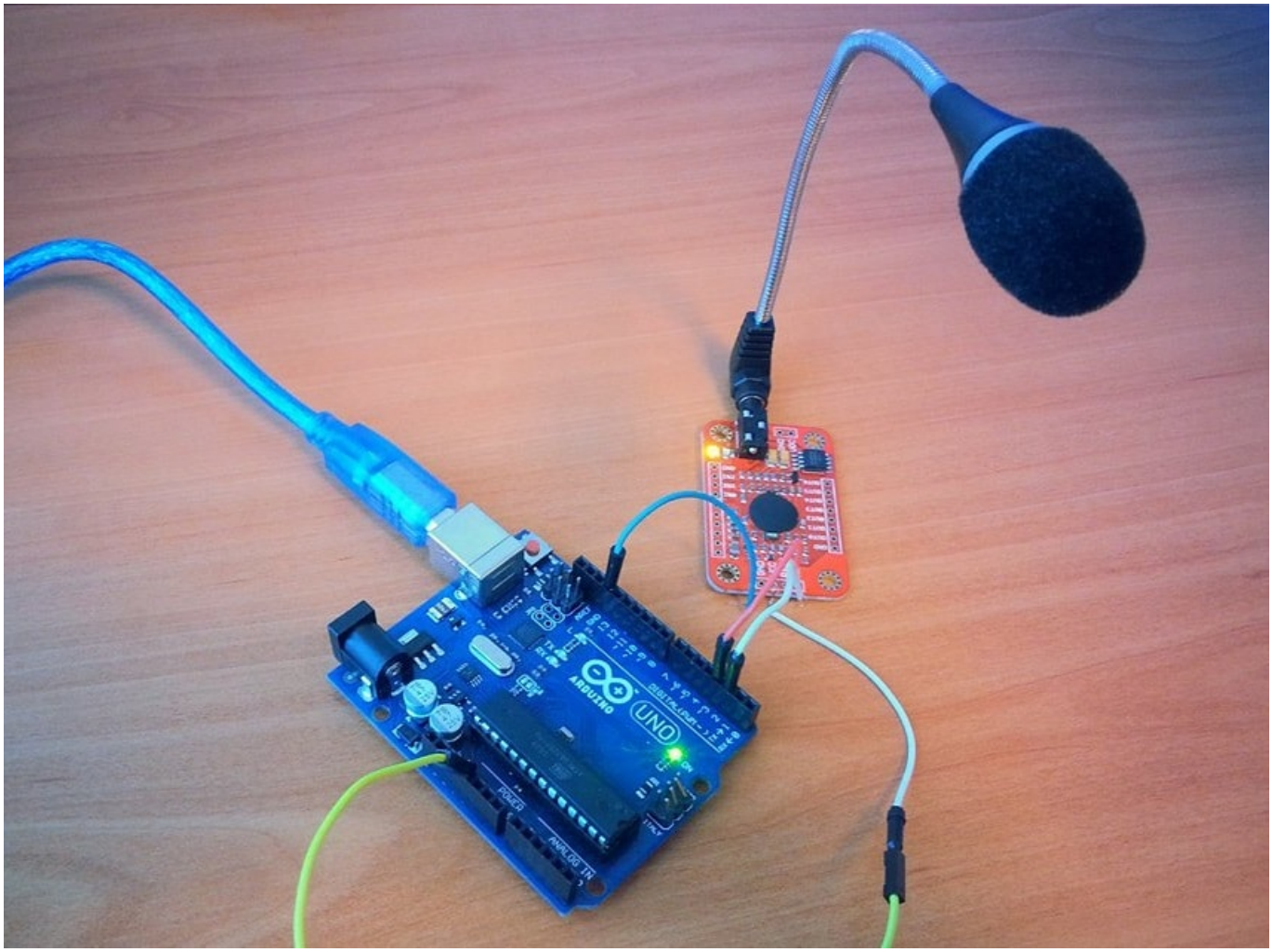
التحكم بالـ LED عبر الأوامر الصوتية

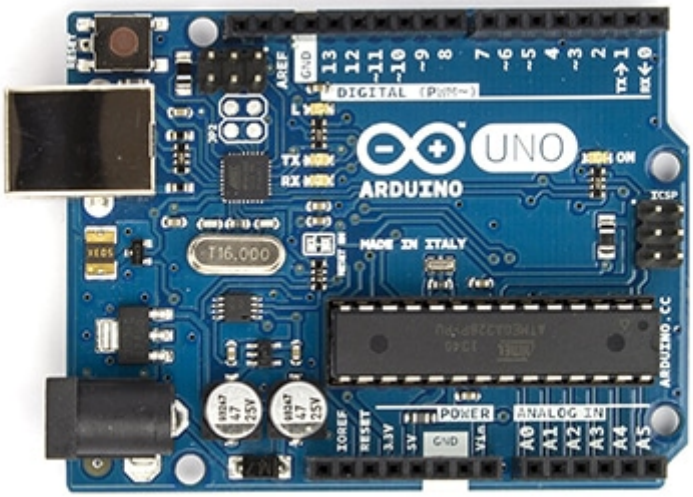
في هذا المشروع، سنقوم بتعلم كيفية تحويل الصوت إلى أوامر تحكم . وتعرف على وحدة التعرف على الصوت (Voice Recognition)، وكيفية تشغيل/ إيقاف الـ LED باستخدام الأوامر الصوتية. هناك عدة طرق أخرى لتنفيذ التعرف على الصوت في مشروعك من خلال هاتف android إلى Alexa أو Raspberry pi أو اي طريقة أخرى.



القطع المطلوبة

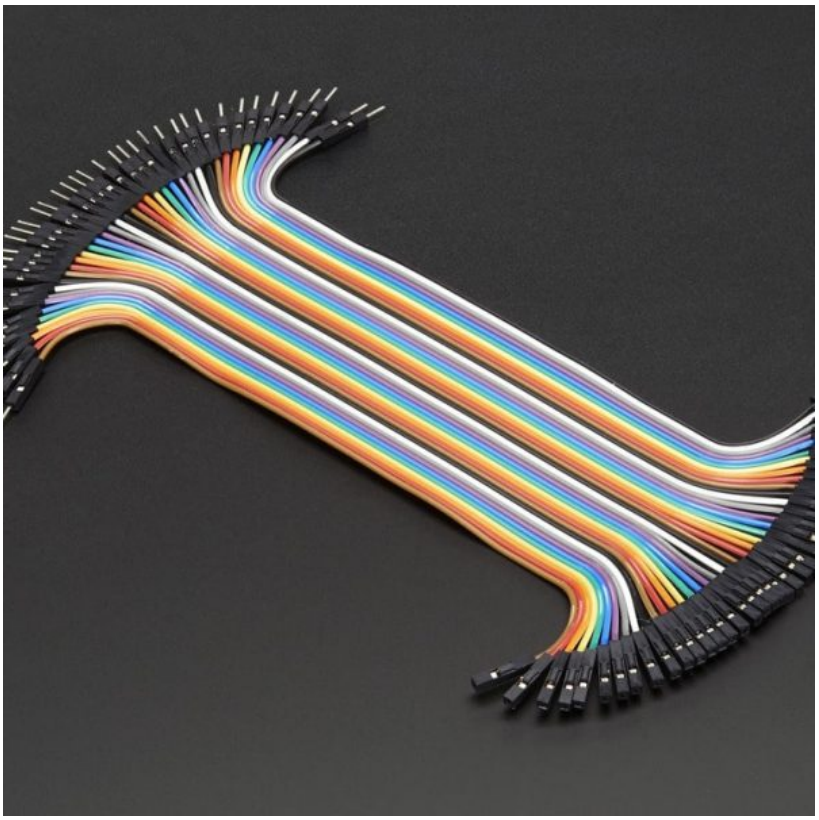
الأدوات التي سيتم استخدامها لهذا المشروع :

Voice Recognition Module v3



Arduino Uno R3

Pin Male Headers



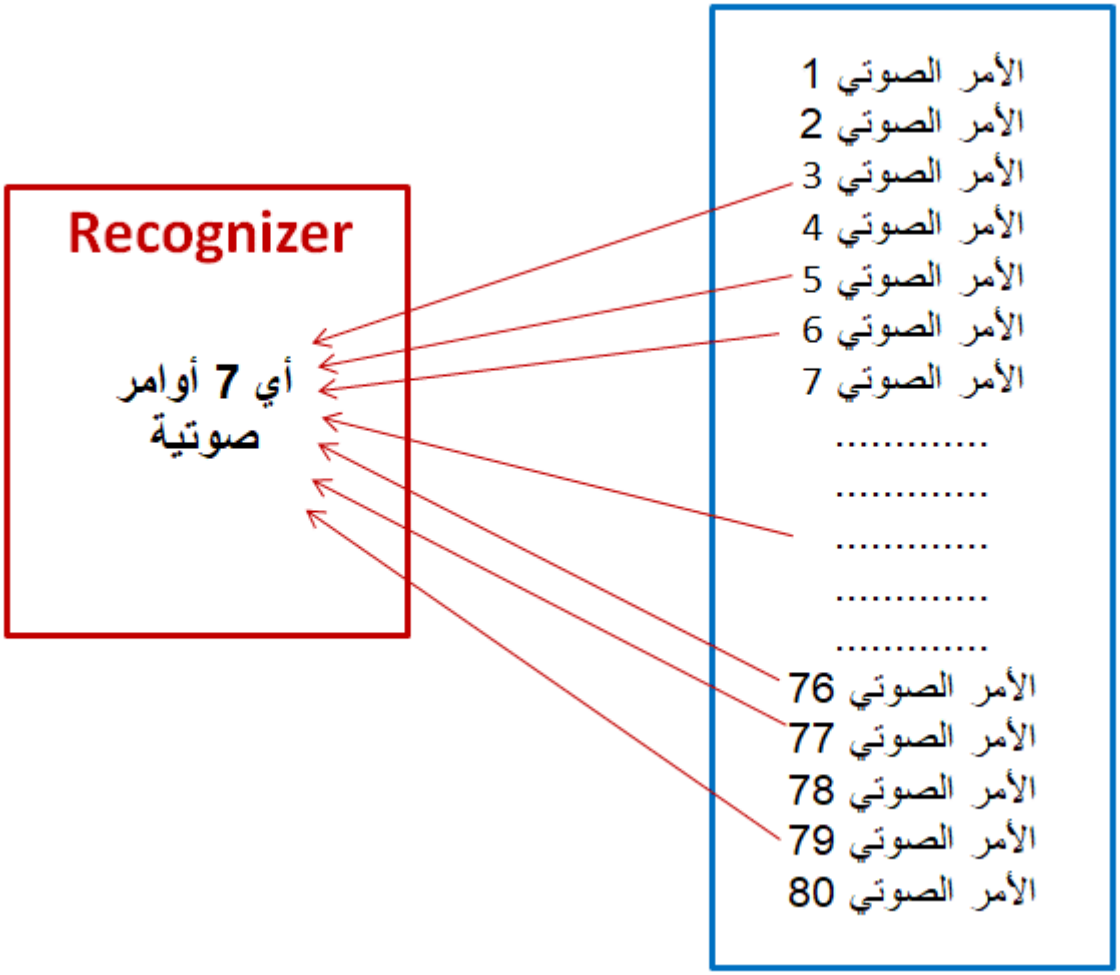
اسلاك توصيل أنثى/ذكر (Jumper Wires Female/male)

وحدة التعرف على الصوت (Voice Recognition) :

وحدة Voice Recognition هي عبارة عن لوحة مدمجة سهله التحكم، تستخدم للتعرف على الصوت/ الكلام للتحكم في المشاريع المختلفة.

لدى وحدة V3 القدرة على تخزين ما يصل إلى 80 أمرا صوتياً لكل منها مدة 1500 milliseconds (اي ما يقارب كلمة أو كلمتين لكل امر). يتم تخزينها في مجموعة واحدة كبيرة مثل المكتبة. ولكن خلال عملها لا تستطيع التمييز بين الـ 80 صوتا في نفس اللحظة، لذلك يتم استيراد/استدعاء سبعة أوامر صوتية في المكتبة إلى أداة التعرف(**Recognizer**) . وهذا يعني أن الحد الأقصى للأوامر

الصوتية هي سبع أوامر فعالة في نفس الوقت.



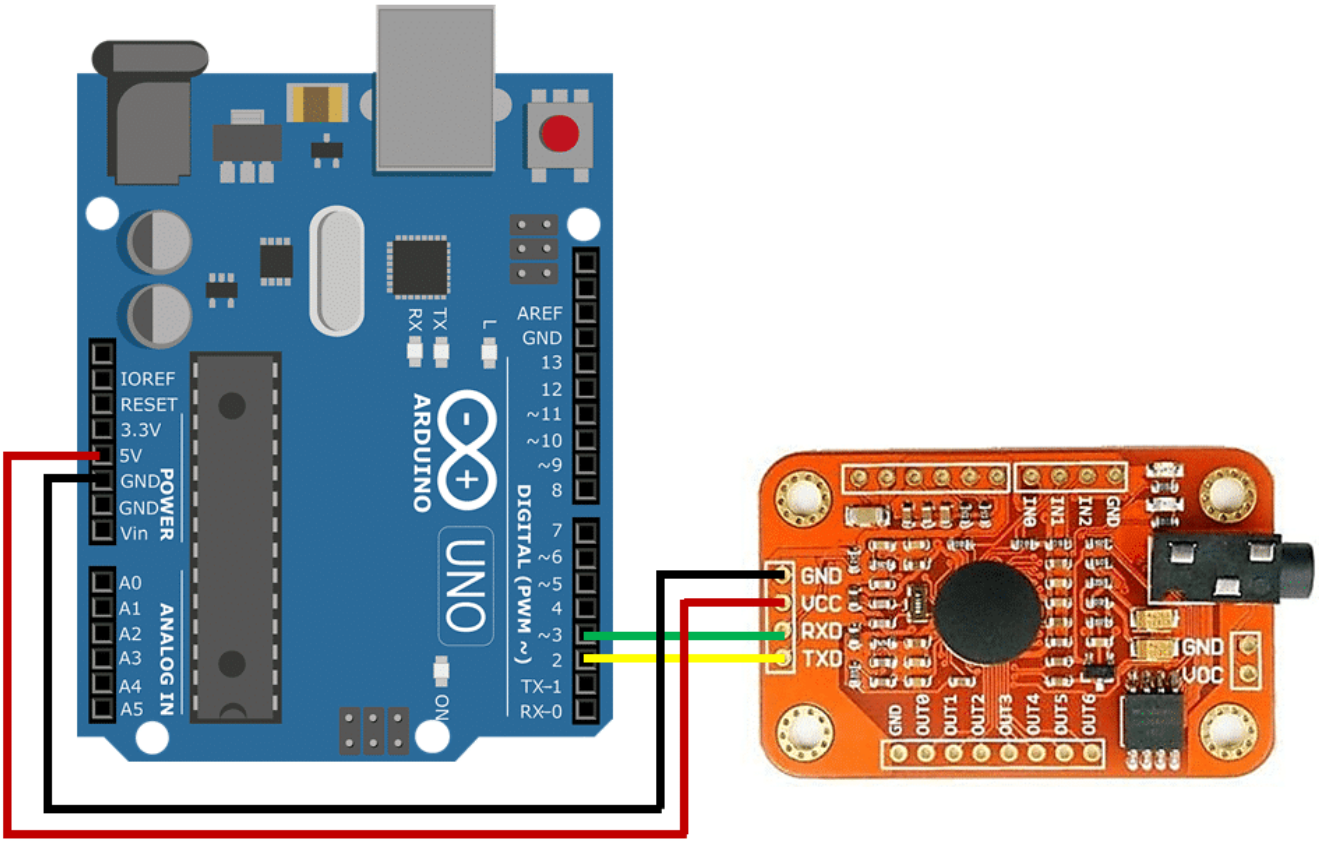
لا تقوم هذه الوحدة بتحويل الأمر إلى نص ولكن ستقوم بمقارنة الصوت القادم من الميكروفون بمجموعة من الأصوات المسجله مسبقا. لذلك من الناحية الفنية لا توجد حواجز لغوية لاستخدام هذا المنتج. يمكنك تسجيل الأمر بأي لغة .

كما انها تحتاج إلى تهيئة وتسجيل الأصوات أولاً قبل اعطاء الأوامر الصوتية للتحكم بالمشاريع.

تصميم لوح التجارب :

هناك طريقتان لإستخدام والتحكم بهذه الوحدة، أولا باستخدام UART أو من خلال دبابيس GPIO المدمجة. سيتم في هذا المشروع استخدام UART.

قم بتوصيل الدارة كما هو موضح بالصورة التالية :



Voice Recognition وحدة	الاردوينو
5v	5v
Tx	2
Rx	3
GND	GND

تهيئة وحدة التعرف على الصوت (Voice Recognition)

ليتم برمجة وحدة Voice Recognition ، والتحكم بالمختلف المشاريع عبر الأوامر الصوتية، قم بإتباع التالي :

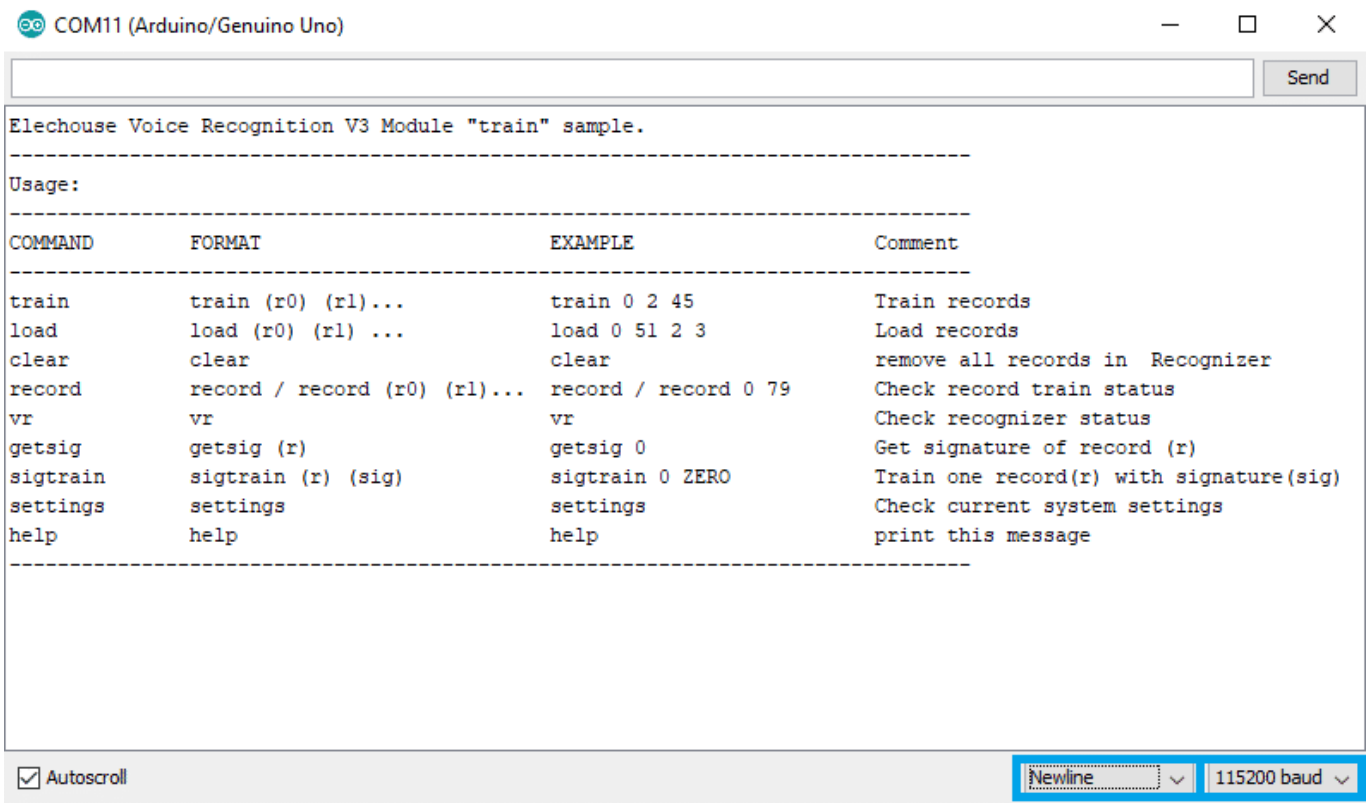
– أولا قم بتثبيت Arduino IDE على جهازك، يمكنك الاطلاع على درس **تنصيب الأردوينو على جهازك** .

– ثم قم بتنزيل مكتبة VoiceRecognitionV3 (قم بتنزيل ملف zip أو استخدم git clone command https://github.com/elechouse/VoiceRecognitionV3.git)

في حالة استخدام ملف zip ، قم باستخراج VoiceRecognitionV3.zip إلى اردوينو Sketch\libraries ، وإذا استخدمت gitclonecommand قم بنسخ VoiceRecognitionV3 إلى ArduinoSketch\libraries .

تحتاج الوحدة إلى تهيئة وتخزين الأوامر الصوتية ثم استدعائها، قم بإتباع الخطوات التالية :

1. فتح كود vr_sample_train). File -> Examples -> VoiceRecognitionV3 -> vr_sample_train)
2. اختيار لوحة التحكم الصحيحة (Tool -> Board, UNO recommended) ، ثم اختيار المنفذ الصحيح.
3. رفع الكود إلى لوحة الأردوينو.
4. بعد ذلك يتم فتح الشاشة التسلسلية (Serial Monitor) . واختيار الـ Baund rate = 115200 ، وتحديد الإرسال مع Newline أو Both NL & CR

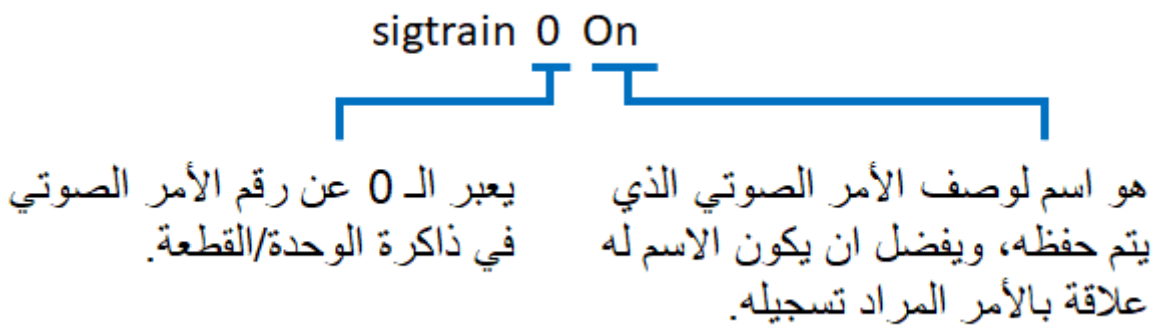


الأوامر الخاصة بوحدة الـ Voice Recognition :

يتم التحقق من إعدادات وحدة الـ Voice Recognition من خلال الأمر **settings** ثم الضغط على ارسال send او على مفتاح Enter .

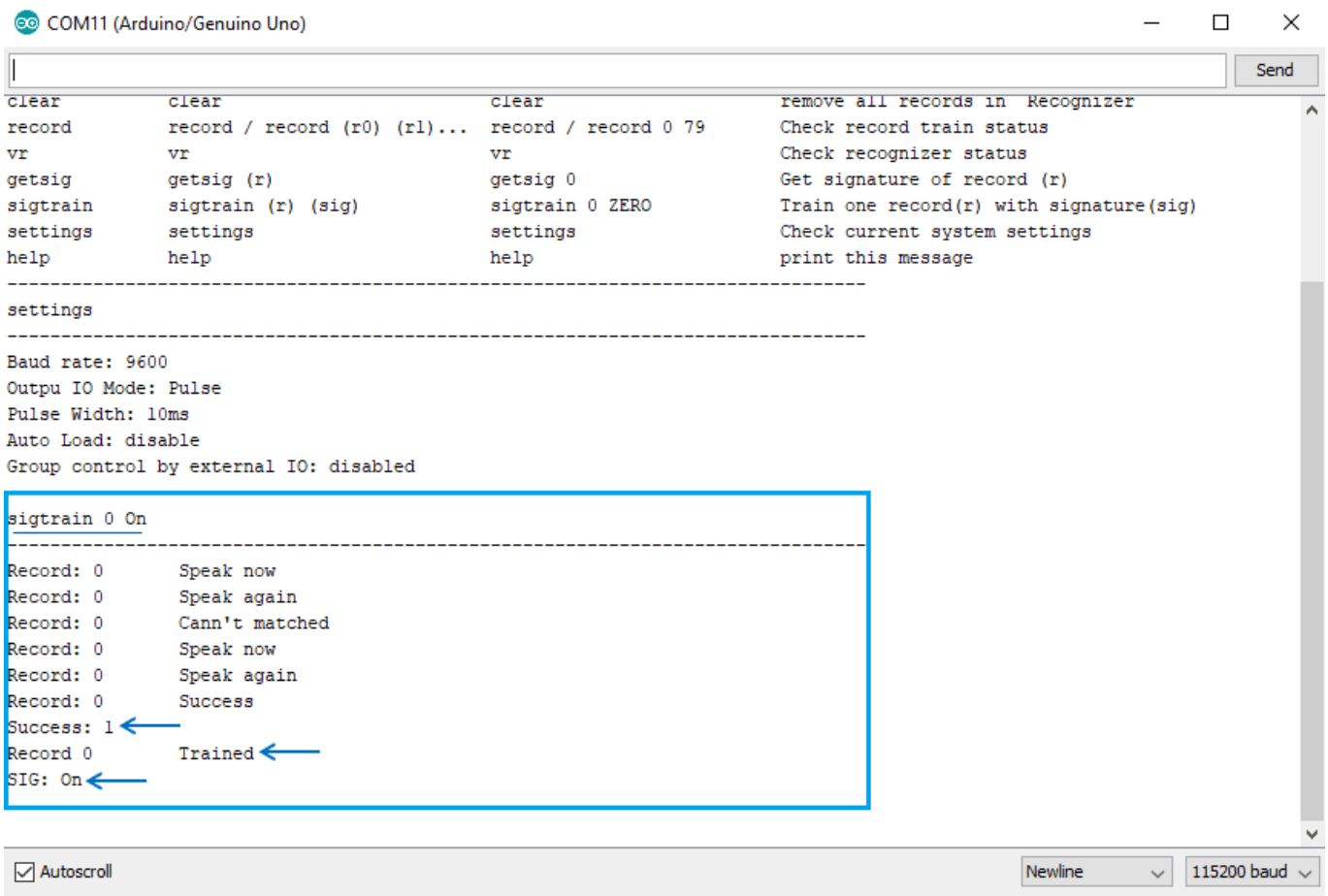
```
COM11 (Arduino/Genuino Uno)
settings                                                    Send
Elechouse Voice Recognition V3 Module "train" sample.
--------------------------------------------------------------------------------
Usage:
--------------------------------------------------------------------------------
COMMAND        FORMAT                        EXAMPLE                    Comment
--------------------------------------------------------------------------------
train          train (r0) (r1)...            train 0 2 45               Train records
load           load (r0) (r1) ...            load 0 51 2 3              Load records
clear          clear                         clear                      remove all records in  Recognizer
record         record / record (r0) (r1)...  record / record 0 79       Check record train status
vr             vr                            vr                         Check recognizer status
getsig         getsig (r)                    getsig 0                   Get signature of record (r)
sigtrain       sigtrain (r) (sig)            sigtrain 0 ZERO            Train one record(r) with signature(sig)
settings       settings                      settings                   Check current system settings
help           help                          help                       print this message

settings
--------------------------------------------------------------------------------
Baud rate: 9600
Outpu IO Mode: Pulse
Pulse Width: 10ms
Auto Load: disable
Group control by external IO: disabled

Autoscroll                                   Newline        115200 baud
```

يتم تسجيل وتخزين الأوامر الصوتية على وحدة Voice Recognition بإسم معين، من خلال ادخال الأمر **sigtrain 0 On** ثم الضغط على ارسال send او على مفتاح Enter :

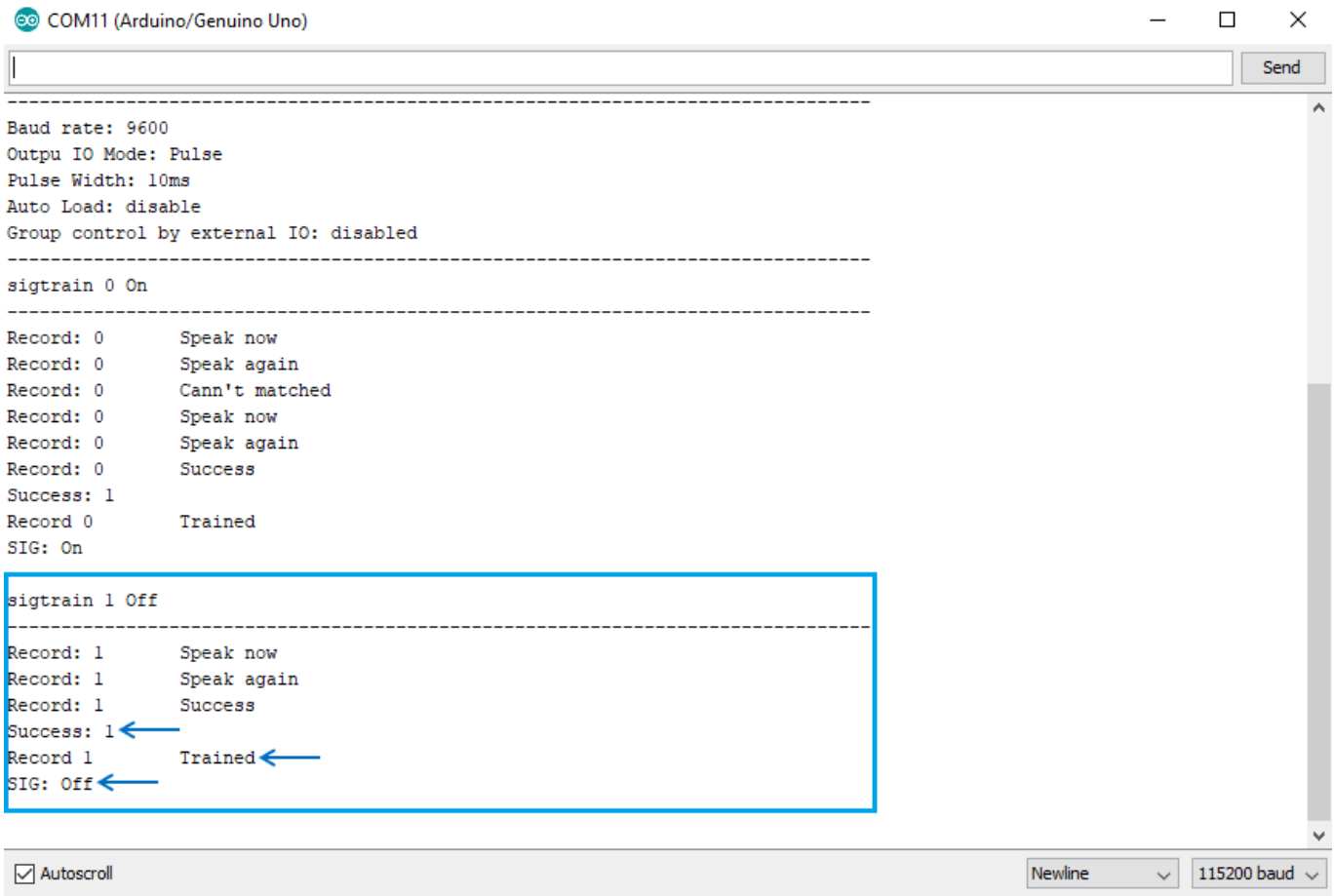


بعد ادخال الأمر السابق والضغط على ارسال send، عند طباعة "Speak now" على الشاشة التسلسلية (Serial Monitor) ، انت بحاجة إلى نطق أمر بصوتك (يمكنك قول اي كلمة ، يوصى باستخدام كلمات ذات صلة بالمشروع ، هنا ستكون كلمة الأمر الصوتي هو "ON")، عند طباعة "Speak again" على الشاشة التسلسلية (Serial Monitor) ، تحتاج إلى إعادة نطق الأمر الصوتي مرة اخرى.

إذا تمت مطابقة هذين الصوتين، سيتم طباعة "Success" على الشاشة التسلسلية (Serial Monitor), ويتم تخزين الأمر الصوتي في سجل 0 "record 0" , وإذا لم يتم التطابق، فكرر نطق الكلام حتى تنجح العملية.



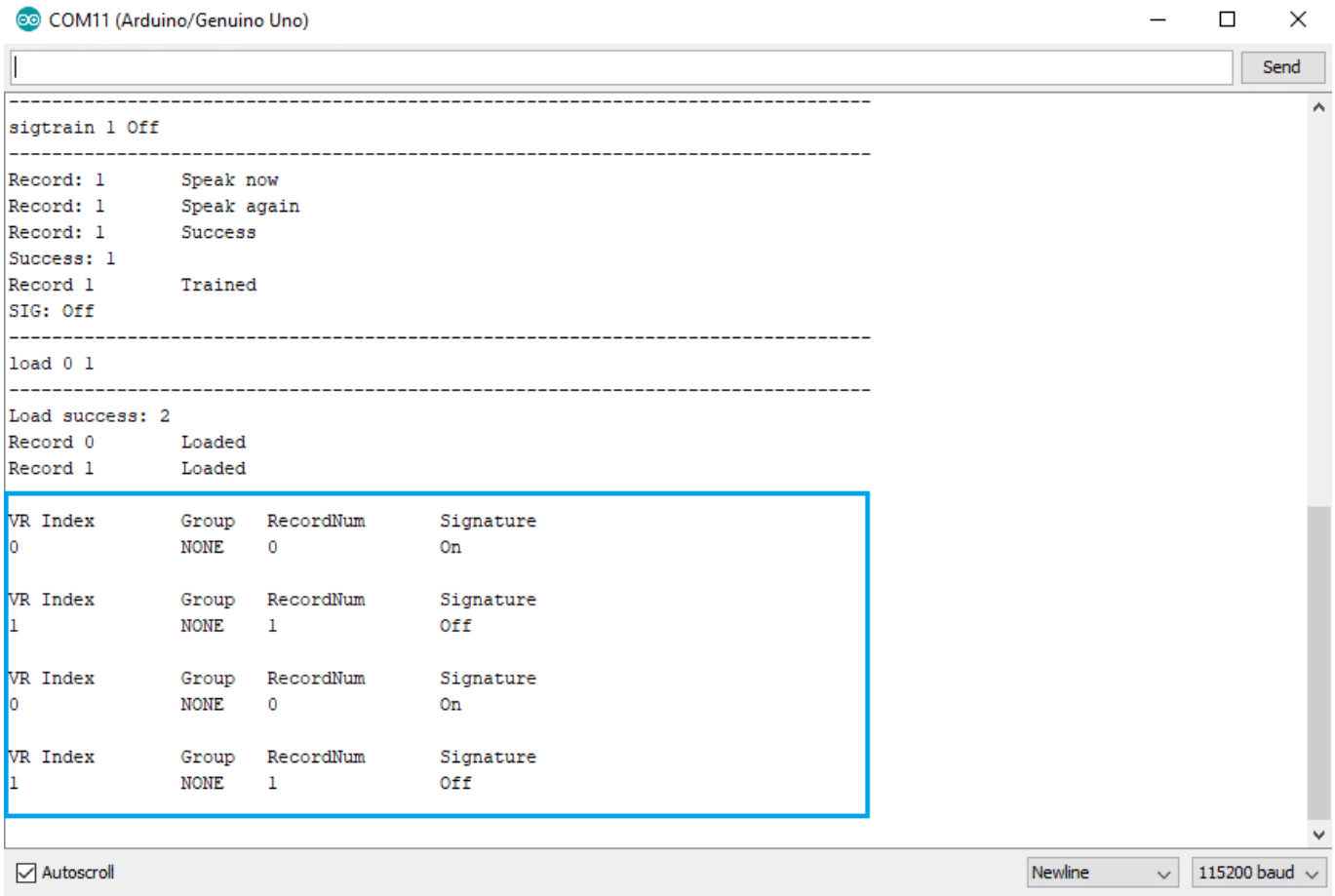
بعد ذلك نقوم بإرسال الأمر sigtrain1Off للتخزين على السجل 1 (record 1) مع اسم "Off" . قم بنطق كلماتك التي ترغب باستخدامها لإيقاف الـ LED .



عند ارسال الأمر **load 0 1** يتم استدعاء/تحميل الأوامر الصوتيه للسجل 0 و 1. كما ذكرنا مسبقا يمكنك استدعاء سبعة أصوات في نفس اللحظة.

```
COM11 (Arduino/Genuino Uno)
Send
--------------------------------------------------------------------------------
sigtrain 0 On
--------------------------------------------------------------------------------
Record: 0       Speak now
Record: 0       Speak again
Record: 0       Cann't matched
Record: 0       Speak now
Record: 0       Speak again
Record: 0       Success
Success: 1
Record 0        Trained
SIG: On
--------------------------------------------------------------------------------
sigtrain 1 Off
--------------------------------------------------------------------------------
Record: 1       Speak now
Record: 1       Speak again
Record: 1       Success
Success: 1
Record 1        Trained
SIG: Off
--------------------------------------------------------------------------------
load 0 1

Load success: 2
Record 0        Loaded
Record 1        Loaded

Autoscroll                                  Newline    115200 baud
```

الآن يمكنك نطق الكلمات التي تم تسجيلها واختبار ما إذا تم التعرف على الصوت. يمكنك أن ترى التالي على الشاشة التسلسلية :



برمجة الأردوينو

هنا نقوم بعرض مثال بسيط يوضح كيفية التحكم بالـ LED المدمج على لوحة الـ Arduino (الـ LED الموصل على المنفذ الرقمي 13) من خلال الأوامر الصوتية. في هذا المثال تحتاج إلى تهيئة وحدة التعرف على الاصوات أولاً (تم عرض الطريقة في الخطوة السابقة) .
سيتم استخدام الأوامر التالية :

– تخزين امر صوتي لإستخدامه في تشغيل الـ LED من خلال الأمر sigtrain 0 on .
– تخزين امر صوتي لإستخدامه في إيقاف الـ LED من خلال الأمر sigtrain 1 off .

الآن نقوم بفتح الملف <- (File -> Examples -> VoiceRecognitionV3 vr_sample_control_led
(vr_sample_control_led ، والذي يحتوي على كود أدناه للتحكم بالـ LED .

```
#include
#include "VoiceRecognitionV3.h"

VR myVR(2,3);    // 2:RX 3:TX, you can choose your favourite pins.

uint8_t records[7]; // save record
uint8_t buf[64];

int led = 13;

#define onRecord    (0)
```

```
#define offRecord   (1)

void printSignature(uint8_t *buf, int len)
{
  int i;
  for(i=0; i<len; i++){
    if(buf[i]>0x19 && buf[i]<0x7F){
      Serial.write(buf[i]);
     }
    else{
      Serial.print("[");
      Serial.print(buf[i], HEX);
      Serial.print("]");
    }
  }
}
/** @brief Print signature, if the character is invisible, print hexible value
instead. @param buf -->  VR module return value when voice is recognized.
             buf[0]  -->  Group mode(FF: None Group, 0x8n: User, 0x0n:System
             buf[1]  -->  number of record which is recognized.
             buf[2]  -->  Recognizer index(position) value of the recognized record.
             buf[3]  -->  Signature length
             buf[4]~buf[n] --> Signature
*/
void printVR(uint8_t *buf)
{
  Serial.println("VR Index\tGroup\tRecordNum\tSignature");

  Serial.print(buf[2], DEC);
  Serial.print("\t\t");

  if(buf[0] == 0xFF){
    Serial.print("NONE");
  }
  else if(buf[0]&0x80){
    Serial.print("UG ");
    Serial.print(buf[0]&(~0x80), DEC);
  }
  else{
    Serial.print("SG ");
    Serial.print(buf[0], DEC);
  }
  Serial.print("\t");

  Serial.print(buf[1], DEC);
  Serial.print("\t\t");
  if(buf[3]>0){
    printSignature(buf+4, buf[3]);
  }
  else{
    Serial.print("NONE");
  }
  Serial.println("\r\n");
```

```
}

void setup()
{
  /** initialize */
  myVR.begin(9600);
  Serial.begin(115200);
  Serial.println("Elechouse Voice Recognition V3 Module\r\nControl LED sample");
  pinMode(led, OUTPUT);
  if(myVR.clear() == 0){
    Serial.println("Recognizer cleared.");
  }else{
    Serial.println("Not find VoiceRecognitionModule.");
    Serial.println("Please check connection and restart Arduino.");
    while(1);
  }
  if(myVR.load((uint8_t)onRecord) >= 0){
    Serial.println("onRecord loaded");
  }
  if(myVR.load((uint8_t)offRecord) >= 0){
    Serial.println("offRecord loaded");
  }
}

void loop()
{
  int ret;
  ret = myVR.recognize(buf, 50);
  if(ret>0){
    switch(buf[1]){
      case onRecord:
        /** turn on LED */
        digitalWrite(led, HIGH);
        break;
      case offRecord:
        /** turn off LED*/
        digitalWrite(led, LOW);
        break;
      default:
        Serial.println("Record function undefined");
        break;
    }
    /** voice recognized */
    printVR(buf);
  }
}
```

شرح الـ Code :

اولا يتم انشاء متغير led يحتوي على المنفذ 13 الخاص بالـ led المدمج بلوحة الاردوينو.

```
int led = 13;
```

ثم يتم تعريف اثنين من السجلات الأول onrecord (لتشغلي الـ LED) و offrecord(لإيقاف تشغيل الـ LED) . قم بتغير قيمة السجلات حسب العنوان/الرقم الذي تم تخزين أوامر التشغيل والايقاف عليها. في هذا المشروع تم استخدام السجل رقم 0 و 1.

```
#define onRecord (0)
#define offRecord (1)
```

في الـ setup()، نقوم بتهيئة وحدة Voice Recognition ، والشاشة التسلسلية .

```
myVR.begin(9600);
Serial.begin(115200);
Serial.println("Elechouse Voice Recognition V3 Module\r\nControl LED sample");
```

تهيئة المنفذ الموصل بالـ LED كمخرج .

```
pinMode(led, OUTPUT);
```

التحقق ما إذا تم توصيل وحدة التعرف على الصوت بشكل صحيح إلى الأردوينو.

```
if(myVR.clear() == 0){
Serial.println("Recognizer cleared.");
}
else{
Serial.println("Not find VoiceRecognitionModule.");
Serial.println("Please check connection and restart Arduino.");
while(1);
}
```

استيراد/نسخ الأمر الصوتي الموجود في السجل 0 (onRecord) إلى recognizer .

```
if(myVR.load((uint8_t)onRecord) >= 0){
Serial.println("onRecord loaded");
}
```

استيراد/نسخ الأمر الصوتي الموجود في السجل 1 (offRecord) إلى recognizer .

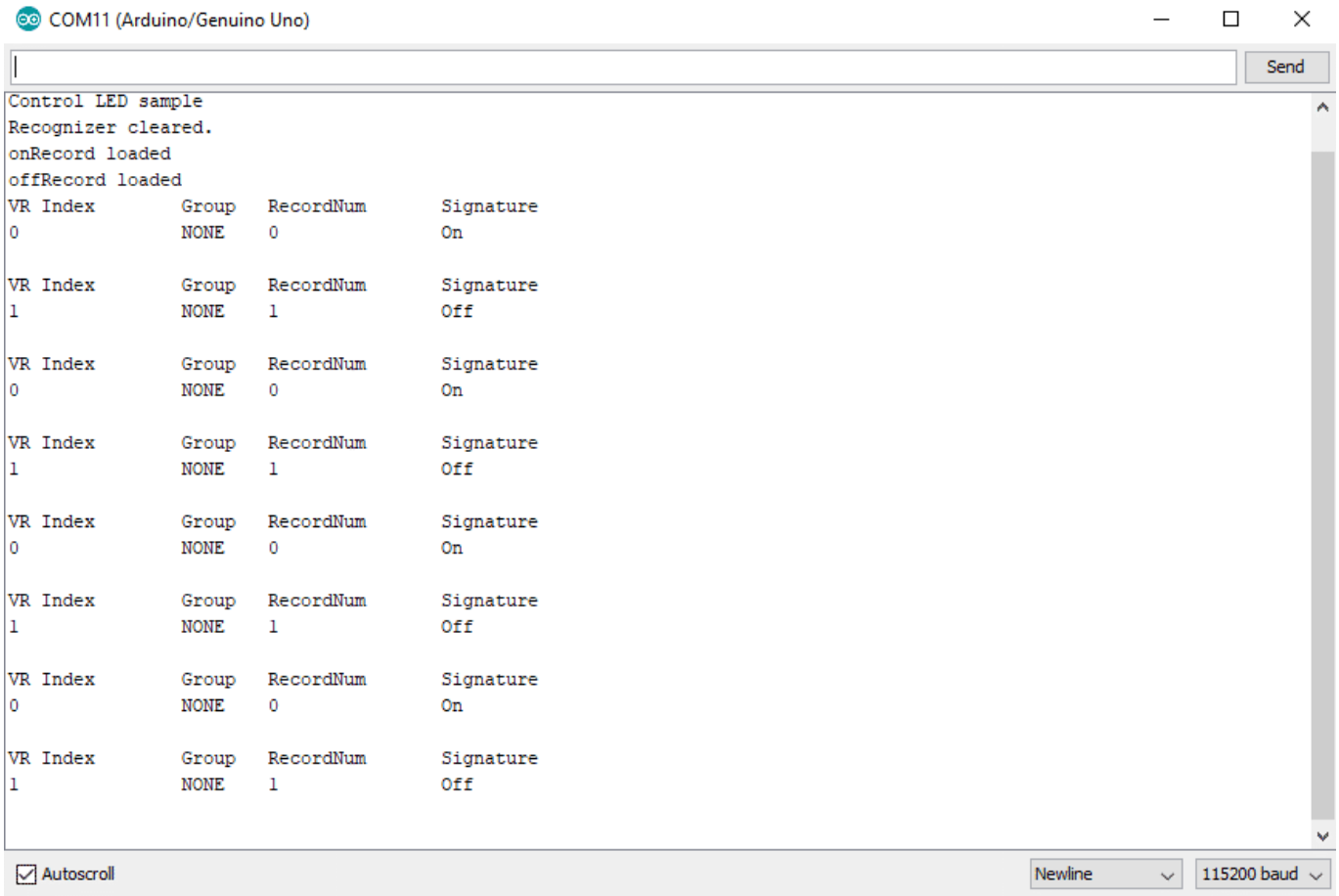
```
if(myVR.load((uint8_t)offRecord) >= 0){
Serial.println("offRecord loaded");
}
```

في loop()، نقوم بمحاولة استقبال الصوت من الميكروفون مع تحديد قيمة مهلة الانتظار timeout

```
int ret;
ret = myVR.recognize(buf, 50);
```

يحتوي buf[1] على رقم السجل الذي تم مطابقته مع السجلات التي تم نسخها على recognizer. لذالك نقوم بمقارنته من السجلات التي تم انشائها onRecord و offRecord، واعتمادا على النتيجة نقوم بالتحكم بالـ LED.

```
switch(buf[1]){
case onRecord:
/** turn on LED */
digitalWrite(led, HIGH);
break;
case offRecord:
/** turn off LED*/
digitalWrite(led, LOW);
break;
default:
Serial.println("Record function undefined");
break;
}
```

وأخيراً، يتم استدعاء دالة printVR و دالة printSignature ليتم طباعة النتيجة والسجلات التي تم التطابق معها.

عند رفع الكود على الأردوينو والبدء بالتحدث سيتم طباعة التالي على الشاشة التسلسلية :