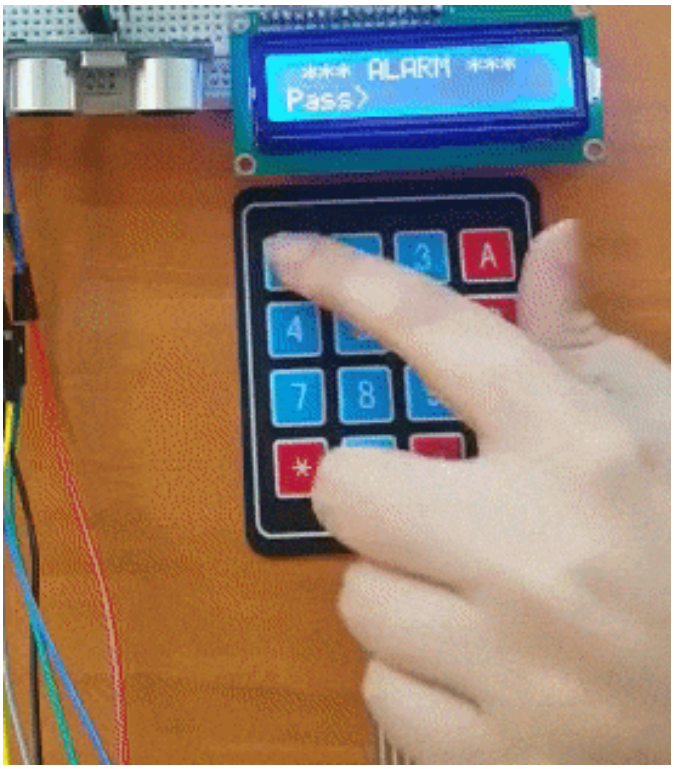
# نظام حماية الخزنة

كما نعلم جميعا أن أنظمة الأمان و الحماية من أهم متطلبات العصر، في هذا المشروع ستتعلم كيفية عمل نظام الحماية و الإنذار من خلال استخدام الاردوينو.




## القطع المطلوبة:

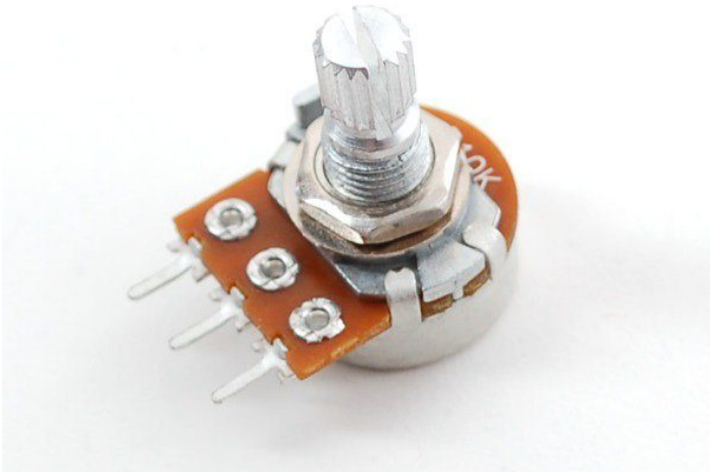
الأدوات التي تحتاجها لهذا المشروع :

Matrix Keypad 4×4

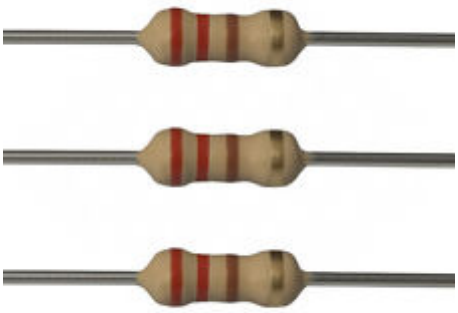
مستشعر الموجات فوق الصوتية (Ultrasonic Sensor HC-SR04)

Piezo sounder

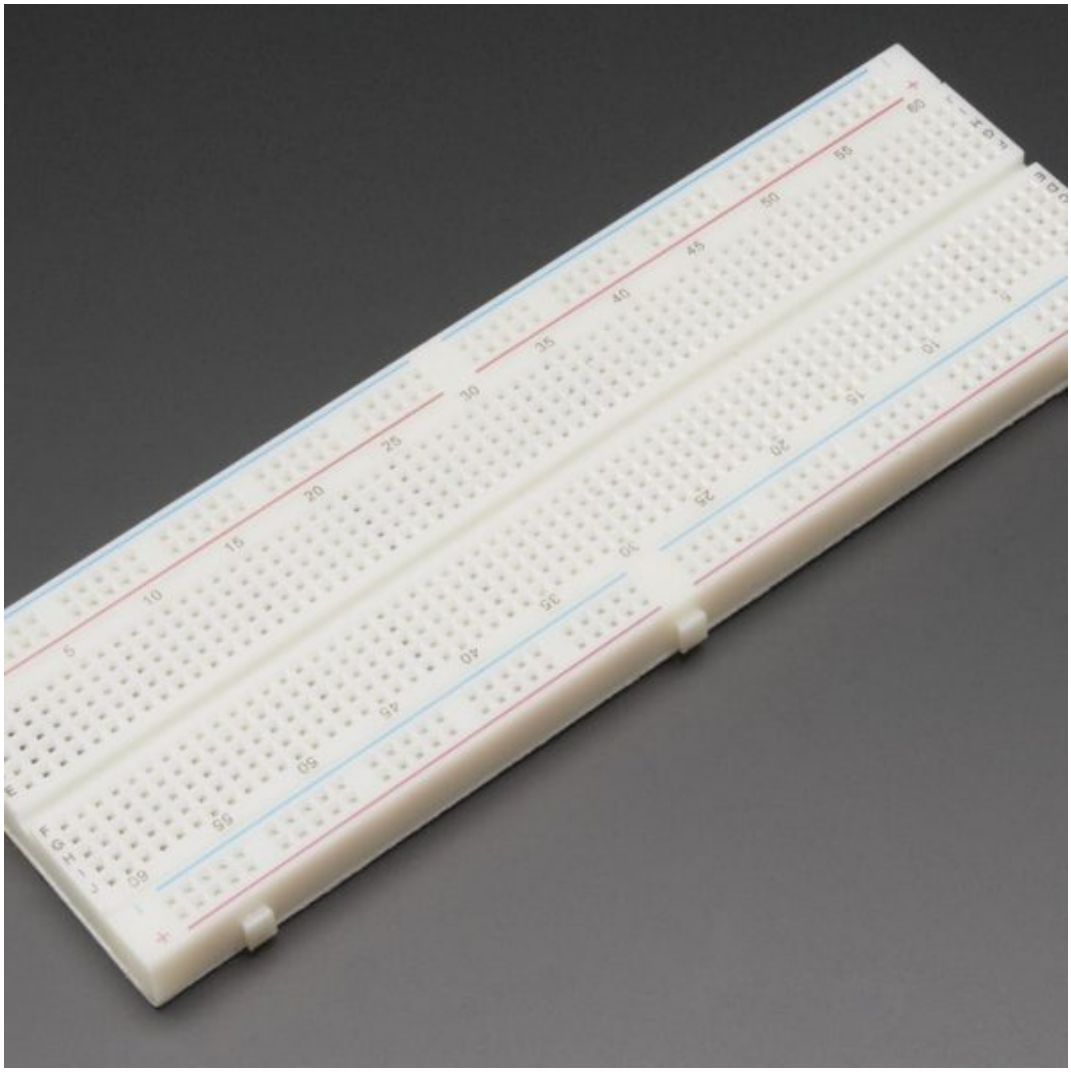
شاشة LCD 16×2



Potentiometer 10K Ω

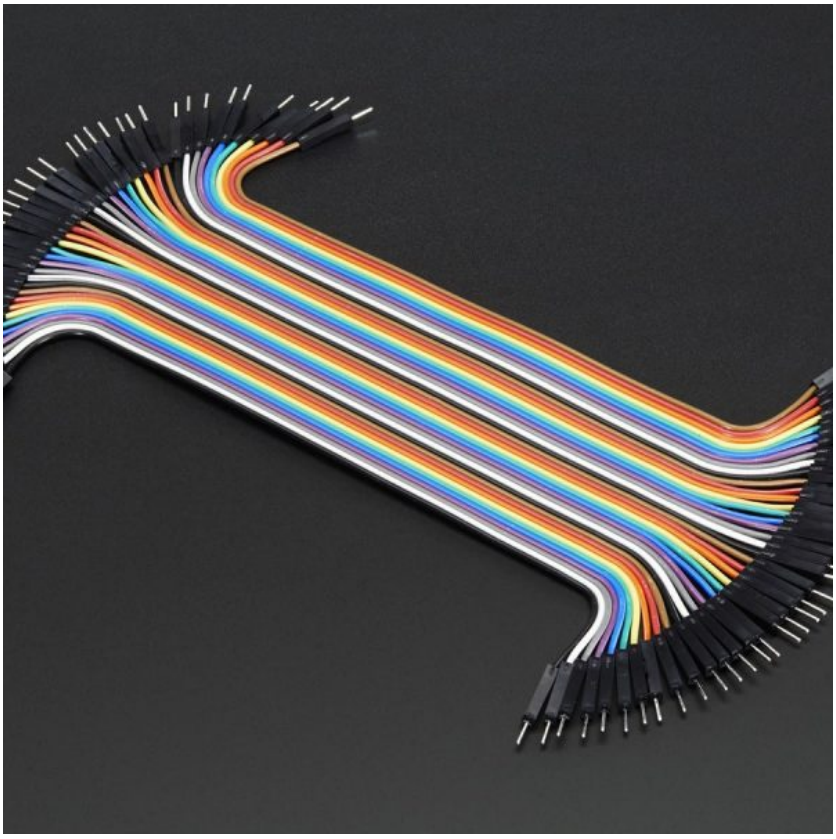


مقاومة 220 اوم



لوح تجارب كبير (Full size breadboard)

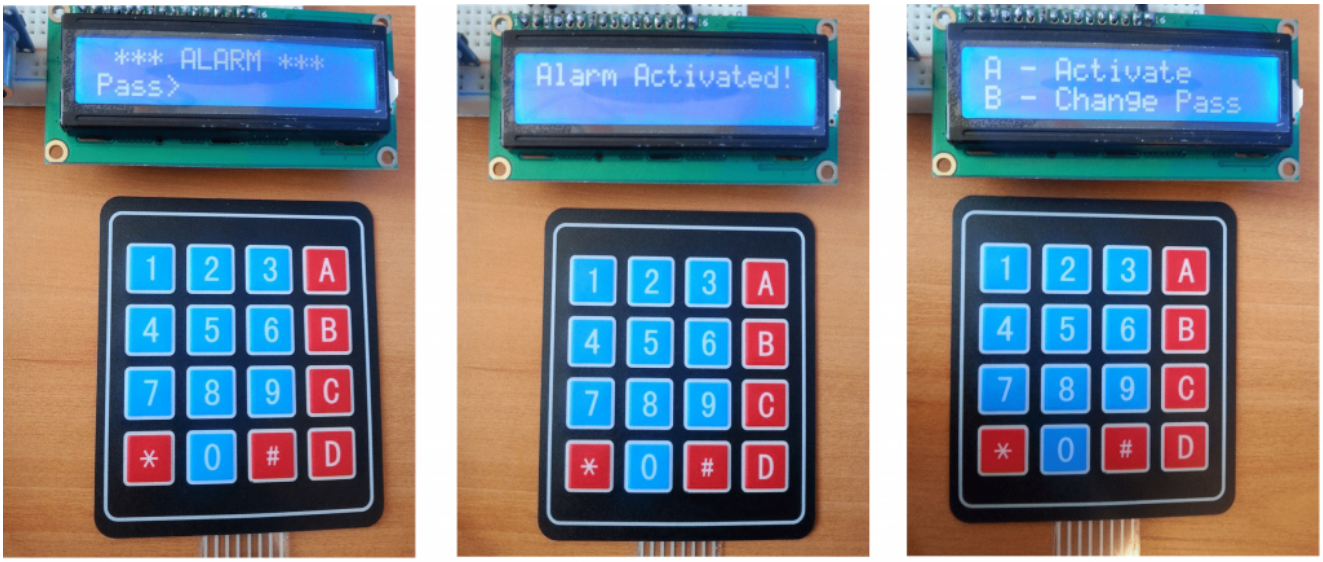
Arduino Uno R3



اسلاك توصيل ذكر/ذكر (Jumper Wires Male Male)

## نظرة عامة

لتفعيل نظام الحماية  في البداية يتم تفعيل أنظمة الانذار بعد 10 ثواني من الضغط على الزر A. وللكشف عن الأجسام يتم استخدام جهاز استشعار الموجات فوق الصوتية (Ultrasonic Sensor) ، في حال الكشف عن وجود جسم قريب يبدأ تفعيل صفارة الانذار (Buzzer) . ولإيقاف صفارة الانذار أنت بحاجة لادخال كلمة المرور المكونة من 4 أرقام ثم الضغط على زر النجمة للتحقق ما اذا كانت الكلمة المدخلة صحيحة او لا . تم وضع كلمة السر الحالية  1234 ولكن لديك ايضا امكانية تغيرها.



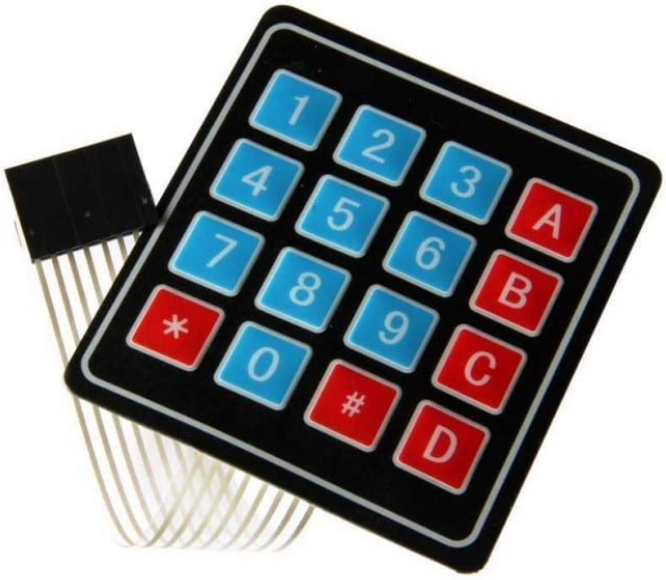

عن طريق الضغط على زر B سيتم الدخول على قائمة تغيير كلمة المرور ، أولا انت بحاجة إلى إدخال كلمة المرور الحالية من أجل المواصلة في إجراءات تغير كلمة المرور، ومن ثم يطلب منك النظام إدخال كلمة المرور الجديدة مكونة من اربع أرقام .

في حال تغيير كلمة المرور, اذا تم تفعيل الانذار سوف تكون قادر على إيقافة عن طريق إدخال كلمة المرور الجديدة فقط. إذا تم إدخال كلمة مرور خاطئة ستظهر رسالة تسمع بإعادة المحاولة.

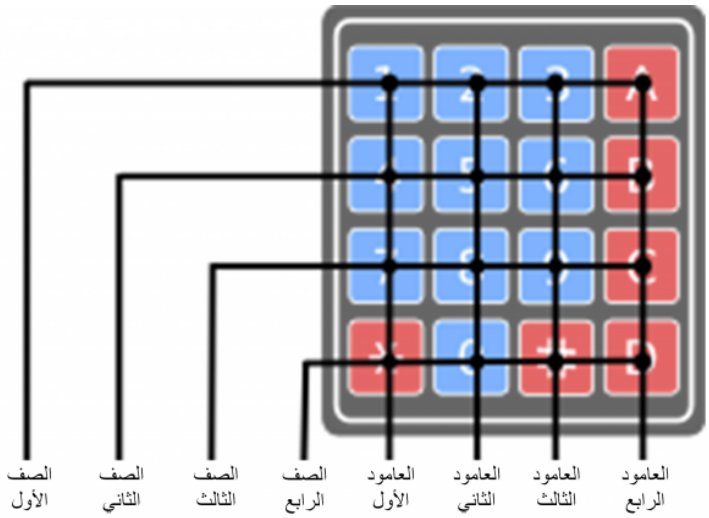
## لوحة المفاتيح Keypad

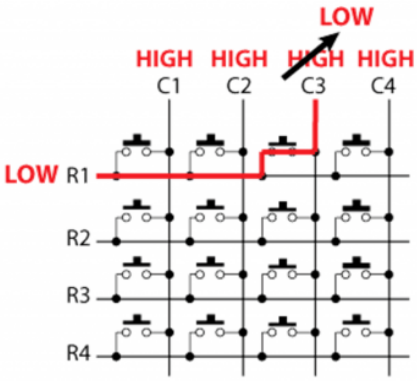
تعتبر لوحة المفاتيح من اهم وسائل الادخال حيث تمكنك من إدخال الارقام و الرموز و بعض الأحرف من خلالها. و يتم إستخدامها بكثرة في مشاريع المتحكمات الدقيقة وأنظمة الحماية و ستجدها حولك في العديد من الاجهزة الالكترونية مثل الهاتف المنزلي وغيره.

تختلف اشكال لوحات المفاتيح تبعا لحجمها و عدد الارقام المتاحة. في هذا المشروع، سوف يتم إستخدام اللوحة المرنة القابلة للطي بسهولة والتي تصنع بالعادة من نوع خاص من البلاستيك المرن.




لوحة المفاتيح 4×4 تتكون من Pins-8 ، أربعة منهم للصفوف و أربعة منهم للأعمدة . كل زر بالواقع عبارة عن مفتاح (Push button)عند الضغط يتم التوصيل بين الصف و العامود.

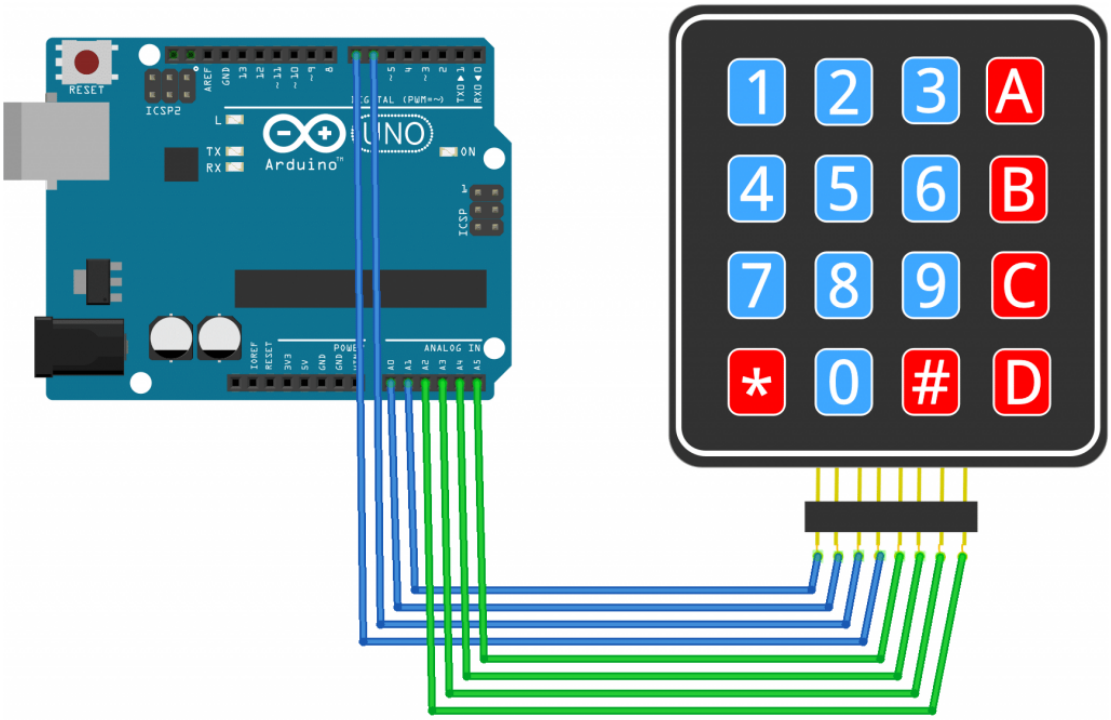



على سبيل المثال : اذا وضع على الصف الأول (Row 1قيمة LOW) ، وجميع الأعمدة قيمة HIGH ، عندما يتم الضغط على الزر الثالث ، سيتم التوصيل بين الصف الأول والعامود الثالث فسوف تصبح قيمة العامود الثالث LOW، وبالتالي يمكننا معرفة الزر الذي تم الضغط عليه.

## تصميم لوح التجارب

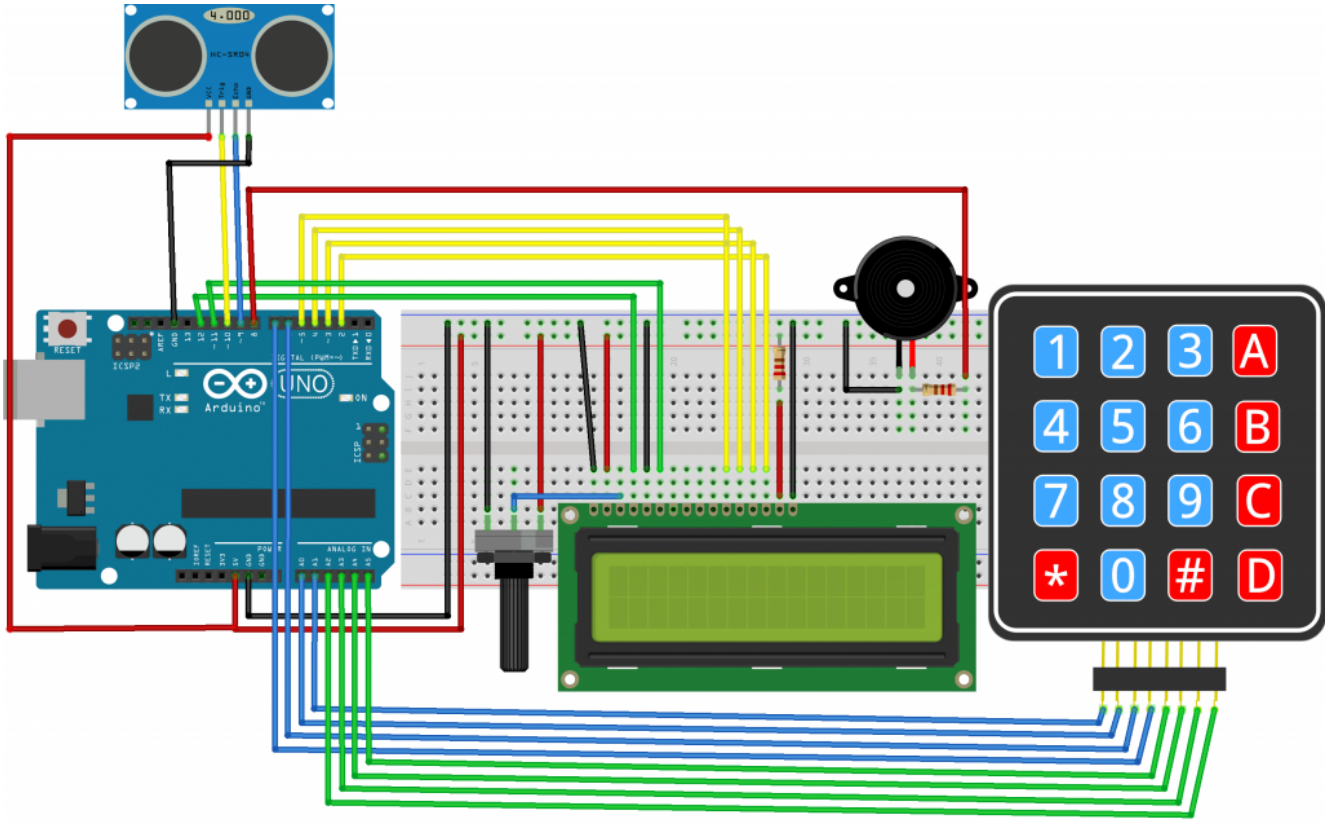
قم بتوصيل الدائرة الكهربائية للوحة المفاتيح كما هو موضح بالصورة :




توصيل لوحة المفاتيح :

| الأردوينو | لوحة المفاتيح |
|---|---|
| Pin A0 | الصف الأول |
| Pin A1 | الصف الثاني |
| Pin 6 | الصف الثالث |
| Pin 7 | الصف الرابع |
| Pin A5 | العامود الأول |
| Pin A4 | العامود الثاني |
| Pin A3 | العامود الثالث |
| Pin A2 | العامود الرابع |

قم بإضافة توصيل باقي القطع الإلكترونية كما هو موضح بالصورة التالية :




**توصيل شاشة الـ LCD :**

| LCD | Arduino |
|---|---|
| RS pin | Pin 12 |
| Enable pin | Pin 11 |
| D4 pin | Pin 5 |
| D5 pin | Pin 4 |
| D6 pin | Pin 3 |
| D7 pin | Pin 2 |

توصيل حساس الموجات فوق الصوتية (Ultrasonic Sensor) :

| Arduino | Ultrasonic Sensor |
|---|---|
| 5V+ | VCC |
| Pin 10 | Trig |
| Pin 9 | Echo |
| GND | Ground |

**توصيل Buzzer:**

| Arduino | Buzzer |
|---|---|
| GND | Shorter lead |
| Pin 8 | Longer lead |

# الكود البرمجي للأردوينو

لفهم الكود بشكل أفضل ، سيتم شرح الكود على أقسام وفي النهاية سأقوم بوضع الكود الكامل.

قم بتنزيل المكتبة الخاصة بلوحة المفاتيح KeyPad من هنا_او من خلال الرابط المباشر هنا .

أولا نحن بحاجة الى مكتبة الـ LCD و مكتبة لوحة المفاتيح. ثم تحديد و تعريف المتغيرات لدبابيس جهاز الانذار (Buzzer)و جهاز استشعار الموجات فوق الصوتية (Ultrasonic)، و تعريف بعض المتغيرات اللازمة لهذا المشروع.

```
#include <LiquidCrystal.h> // includes the LiquidCrystal Library
#include <Keypad.h>
#define buzzer 8
#define trigPin 10
#define echoPin 9
long duration;
int distance, initialDistance, currentDistance, i;
int screenOffMsg =0;
String password="1234";
String tempPassword;
boolean activated = false; // State of the alarm
boolean isActivated;
boolean activateAlarm = false;
boolean alarmActivated = false;
boolean enteredPassword; // State of the entered password to stop the alarm
boolean passChangeMode = false;
boolean passChanged = false;
const byte ROWS = 4; //four rows
const byte COLS = 4; //four columns
char keypressed;
//define the cymbols on the buttons of the keypads
char keyMap[ROWS][COLS] = {
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};
byte rowPins[ROWS] = {A0,A1,6,7};
byte colPins[COLS] = {A5,A4,A3,A2};  //Column pinouts of the keypad
Keypad myKeypad = Keypad( makeKeymap(keyMap), rowPins, colPins, ROWS, COLS);
LiquidCrystal lcd(12, 11, 5, 4, 3, 2); // Creates an LC object. Parameters: (rs,
enable, d4, d5, d6, d7)
```

في قسم الأعداد (setup)، نحن بحاجة لتهيئة الـ LCD وتحديد ما اذا كان الـ Pin الخاص بالمستشعر و جهاز الانذار مدخل او مخرج .

```
void setup() {
  lcd.begin(16,2);
  pinMode(buzzer, OUTPUT); // Set buzzer as an output
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin, INPUT); // Sets the echoPin as an Input
}
```

في القسم Loop، نحن أولا بحاجة للتحقق ما اذا تم تفعيل أنظمة الانذار او لا. فإذا لم يتم التفعيل ، سيتم ظهور القائمة الرئيسية على شاشة الـ LCD التي تقدم خيارين ، خيار A لتفعيل الانذار وخيار B لتغير كلمة المرور.
ثم استخدام myKeypad.getKey() لقراءة الزر الذي تم الضغط عليه من لوحة المفاتيح ، اذا تم الضغط على زر A ، يتم تفعيل جهاز الانذار لمدة200 milliseconds و تصبح قيمة المتغير activeAlarm= true.

```
if (!alarmActivated) {
if (screenOffMsg == 0 ){
lcd.clear();
lcd.setCursor(0,0);
lcd.print("A - Activate");
lcd.setCursor(0,1);
lcd.print("B - Change Pass");
screenOffMsg = 1;
}
keypressed = myKeypad.getKey();
if (keypressed =='A'){ //If A is pressed, activate the alarm
tone(buzzer, 1000, 200);
activateAlarm = true;
}
```

في حالة تم الضغط على زر A يتم تفعيل الانذار و يتم طباعة الرسالة "Alarm will be activated in" على شاشة الـ LCD و عن طريق استخدام while loop يتم انشاء عداد تنازلي لمدة 9 ثواني قبل تفعيل الانذار .
ثم يتم ظهور الرسالة "Alarm Activated " على الشاشة ويتم حساب المسافة الأولية بين نظام الامن (بإستخدام حساس الموجات فوق الصوتية ) و اي جسم امامه .

```
if (activateAlarm) {
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Alarm will be");
lcd.setCursor(0,1);
lcd.print("activated in");
int countdown = 9; // 9 seconds count down before activating the alarm
while (countdown != 0) {
lcd.setCursor(13,1);
lcd.print(countdown);
countdown--;
tone(buzzer, 700, 100);
delay(1000);
}
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Alarm Activated!");
initialDistance = getDistance();
activateAlarm = false;
alarmActivated = true;
```

في الخطوة التالية يقوم جهاز استشعار الموجات فوق الصوتية بالتحقق باستمرار ما إذا كانت المسافة المقاسة حاليا أصغر من المسافة الأولية ، فهذا يدل على وجود كائن اما جهاز الاستشعار فيتم تفعيل الانذار. يتم استخدام الدالة ()tone لتفعيل الـ Buzzer و ثم استدعاء الدالة ()enterPassword .

```
if (alarmActivated == true){
currentDistance = getDistance() + 10;
if ( currentDistance < initialDistance) {
tone(buzzer, 1000); // Send 1KHz sound signal
lcd.clear();
enterPassword();
}
}
```

enterPassword() هذة الدالة تقوم بطباعة رسالة على الشاشة توضح بها أن جهاز الانذار مفعل , وأننا بحاجة إلى إدخال كلمة السر من أجل إيقاف الانذار. وبإستخدام الـ while Loop سيتم التحقق باستمرار ما اذا تم ضغط زر على لوحة المفاتيح ، وكل زر يتم ضغطة يتم اضافته إلى متغير tempPassword. اذا تم ادخال اكثر من 4 ارقام او رمز # يتم مسح الادخال القديم و يمكنك اعادة ادخال كلمة المرور.

```
void enterPassword() {
  int k=5;
  tempPassword = "";
  activated = true;
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(" *** ALARM *** ");
  lcd.setCursor(0,1);
  lcd.print("Pass>");
      while(activated) {
      keypressed = myKeypad.getKey();
      if (keypressed != NO_KEY){
        if (keypressed == '0' || keypressed == '1' || keypressed == '2' ||
keypressed == '3' ||
            keypressed == '4' || keypressed == '5' || keypressed == '6' ||
keypressed == '7' ||
            keypressed == '8' || keypressed == '9' ) {
          tempPassword += keypressed;
          lcd.setCursor(k,1);
          lcd.print("*");
          k++;
        }
      }
      if (k > 9 || keypressed == '#') {
        tempPassword = "";
        k=5;
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print(" *** ALARM *** ");
        lcd.setCursor(0,1);
        lcd.print("Pass>");
      }
      if ( keypressed == '*') {
        if ( tempPassword == password ) {
          activated = false;
          alarmActivated = false;
          noTone(buzzer);
```

```
          screenOffMsg = 0;
        }
        else if (tempPassword != password) {
          lcd.setCursor(0,1);
          lcd.print("Wrong! Try Again");
          delay(2000);
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print(" *** ALARM *** ");
          lcd.setCursor(0,1);
          lcd.print("Pass>");
        }
      }
    }
}
```

و من ناحية أخرى إذا تم الضغط على زر النجمة سوف يتم التحقق ما اذا كانت الكلمة المدخلة صحيحة او لا. اذا كانت الكلمة صحيحة سيتم إيقاف الانزار و سيتم الرجوع الى الشاشة الرئيسية على شاشة الـ LCD. واذا تم ادخال كلمة مرور خاطئة يتم ظهور الرسالة (Wrong! Try Again!) و سيكون لدينا محاولة لإدخال كلمة المرور الصحيحة مرة أخرى .
لتغير كلمة المرور نستخدم طريقة مماثلة . أولا سنحتاج إلى ادخال كلمة المرور الحالية لتكون قادر على تعيين كلمة المرور الجديدة.

الكود التالي هو البرنامج الكامل قم برفعه على الاردوينو:

```
#include  // includes the LiquidCrystal Library
#include
#define buzzer 8
#define trigPin 10
#define echoPin 9
long duration;
int distance, initialDistance, currentDistance, i;
int screenOffMsg =0;
String password="1234";
String tempPassword;
boolean activated = false; // State of the alarm
boolean isActivated;
boolean activateAlarm = false;
boolean alarmActivated = false;
boolean enteredPassword; // State of the entered password to stop the alarm
boolean passChangeMode = false;
boolean passChanged = false;
const byte ROWS = 4; //four rows
const byte COLS = 4; //four columns
char keypressed;
//define the cymbols on the buttons of the keypads
char keyMap[ROWS][COLS] = {
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};
byte rowPins[ROWS] = {A0,A1,6,7};
byte colPins[COLS] = {A5,A4,A3,A2};  //Column pinouts of the keypad
```

```
Keypad myKeypad = Keypad( makeKeymap(keyMap), rowPins, colPins, ROWS, COLS);
LiquidCrystal lcd(12, 11, 5, 4, 3, 2); // Creates an LC object. Parameters: (rs,
enable, d4, d5, d6, d7)
void setup() {
  lcd.begin(16,2);
  pinMode(buzzer, OUTPUT); // Set buzzer as an output
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin, INPUT); // Sets the echoPin as an Input
}
void loop() {
  if (activateAlarm) {
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Alarm will be");
    lcd.setCursor(0,1);
    lcd.print("activated in");
    int countdown = 9; // 9 seconds count down before activating the alarm
    while (countdown != 0) {
      lcd.setCursor(13,1);
      lcd.print(countdown);
      countdown--;
      tone(buzzer, 700, 100);
      delay(1000);
    }
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Alarm Activated!");
    initialDistance = getDistance();
    activateAlarm = false;
    alarmActivated = true;
  }
  if (alarmActivated == true){
      currentDistance = getDistance() + 10;
      if ( currentDistance < initialDistance) { tone(buzzer, 1000); // Send 1KHz
sound signal lcd.clear(); enterPassword(); } } if (!alarmActivated) { if
(screenOffMsg == 0 ){ lcd.clear(); lcd.setCursor(0,0); lcd.print("A - Activate");
lcd.setCursor(0,1); lcd.print("B - Change Pass"); screenOffMsg = 1; } keypressed =
myKeypad.getKey(); if (keypressed =='A'){ //If A is pressed, activate the alarm
tone(buzzer, 1000, 200); activateAlarm = true; } else if (keypressed =='B') {
lcd.clear(); int i=1; tone(buzzer, 2000, 100); tempPassword = "";
lcd.setCursor(0,0); lcd.print("Current Password"); lcd.setCursor(0,1);
lcd.print(">");
      passChangeMode = true;
      passChanged = true;
      while(passChanged) {
      keypressed = myKeypad.getKey();
      if (keypressed != NO_KEY){
        if (keypressed == '0' || keypressed == '1' || keypressed == '2' ||
keypressed == '3' ||
            keypressed == '4' || keypressed == '5' || keypressed == '6' ||
keypressed == '7' ||
            keypressed == '8' || keypressed == '9' ) {
         tempPassword += keypressed;
```

```
         lcd.setCursor(i,1);
         lcd.print("*");
         i++;
         tone(buzzer, 2000, 100);
        }
      }
      if (i > 5 || keypressed == '#') {
        tempPassword = "";
        i=1;
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("Current Password");
        lcd.setCursor(0,1);
        lcd.print(">");
      }
      if ( keypressed == '*') {
        i=1;
        tone(buzzer, 2000, 100);
        if (password == tempPassword) {
          tempPassword="";
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print("Set New Password");
          lcd.setCursor(0,1);
          lcd.print(">");
          while(passChangeMode) {
            keypressed = myKeypad.getKey();
            if (keypressed != NO_KEY){
              if (keypressed == '0' || keypressed == '1' || keypressed == '2' ||
keypressed == '3' ||
                  keypressed == '4' || keypressed == '5' || keypressed == '6' ||
keypressed == '7' ||
                  keypressed == '8' || keypressed == '9' ) {
                tempPassword += keypressed;
                lcd.setCursor(i,1);
                lcd.print("*");
                i++;
                tone(buzzer, 2000, 100);
              }
            }
            if (i > 5 || keypressed == '#') {
              tempPassword = "";
              i=1;
              tone(buzzer, 2000, 100);
              lcd.clear();
              lcd.setCursor(0,0);
              lcd.print("Set New Password");
              lcd.setCursor(0,1);
              lcd.print(">");
            }
            if ( keypressed == '*') {
              i=1;
              tone(buzzer, 2000, 100);
```

```
              password = tempPassword;
              passChangeMode = false;
              passChanged = false;
              screenOffMsg = 0;
            }
          }
        }
      }
    }
   }
 }
}
void enterPassword() {
  int k=5;
  tempPassword = "";
  activated = true;
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(" *** ALARM *** ");
  lcd.setCursor(0,1);
  lcd.print("Pass>");
      while(activated) {
      keypressed = myKeypad.getKey();
      if (keypressed != NO_KEY){
        if (keypressed == '0' || keypressed == '1' || keypressed == '2' ||
keypressed == '3' ||
            keypressed == '4' || keypressed == '5' || keypressed == '6' ||
keypressed == '7' ||
            keypressed == '8' || keypressed == '9' ) {
          tempPassword += keypressed;
          lcd.setCursor(k,1);
          lcd.print("*");
          k++;
        }
      }
      if (k > 9 || keypressed == '#') {
        tempPassword = "";
        k=5;
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print(" *** ALARM *** ");
        lcd.setCursor(0,1);
        lcd.print("Pass>");
      }
      if ( keypressed == '*') {
        if ( tempPassword == password ) {
          activated = false;
          alarmActivated = false;
          noTone(buzzer);
          screenOffMsg = 0;
        }
        else if (tempPassword != password) {
          lcd.setCursor(0,1);
```

```
          lcd.print("Wrong! Try Again");
          delay(2000);
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print(" *** ALARM *** ");
          lcd.setCursor(0,1);
          lcd.print("Pass>");
        }
      }
    }
}
// Custom function for the Ultrasonic sensor
long getDistance(){
  //int i=10;
  //while( i<=10 ) {
  // Clears the trigPin
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  // Sets the trigPin on HIGH state for 10 micro seconds
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  // Reads the echoPin, returns the sound wave travel time in microseconds
  duration = pulseIn(echoPin, HIGH);
  // Calculating the distance
  distance = duration*0.034/2;
  //sumDistance += distance;
  //}
  //int averageDistance= sumDistance/10;
  return distance;
}
```