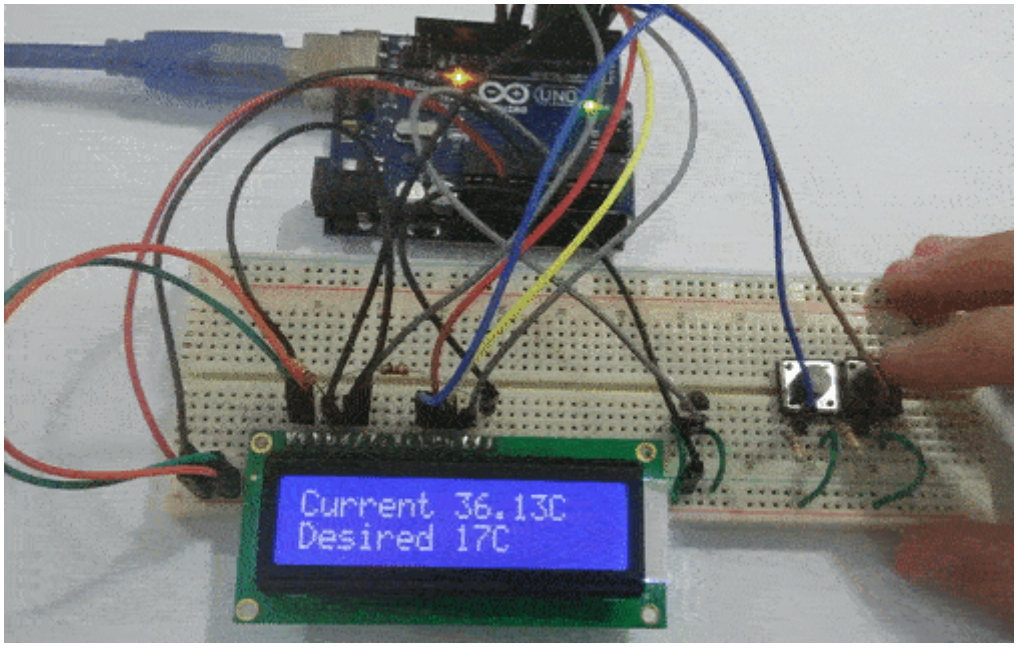
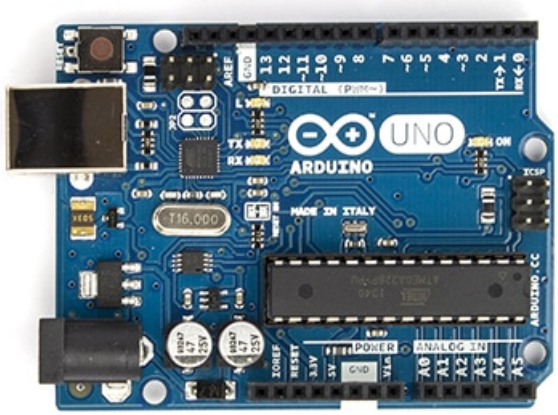
جهاز للتحكم وقياس درجة الحرارة

في هذا المشروع سنقوم بعمل جهاز لمراقبة وقياس درجة الحرارة . سيتم قياس درجة الحرارة بإستخدام حساس درجة الحرارة LM35، وعرض درجة الحرارة الحالية والمطلوبة من خلال شاشة العرض.



المكونات المطلوبة



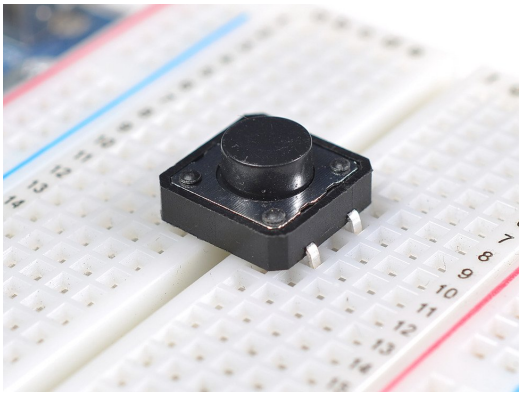
Arduino Uno

LM35

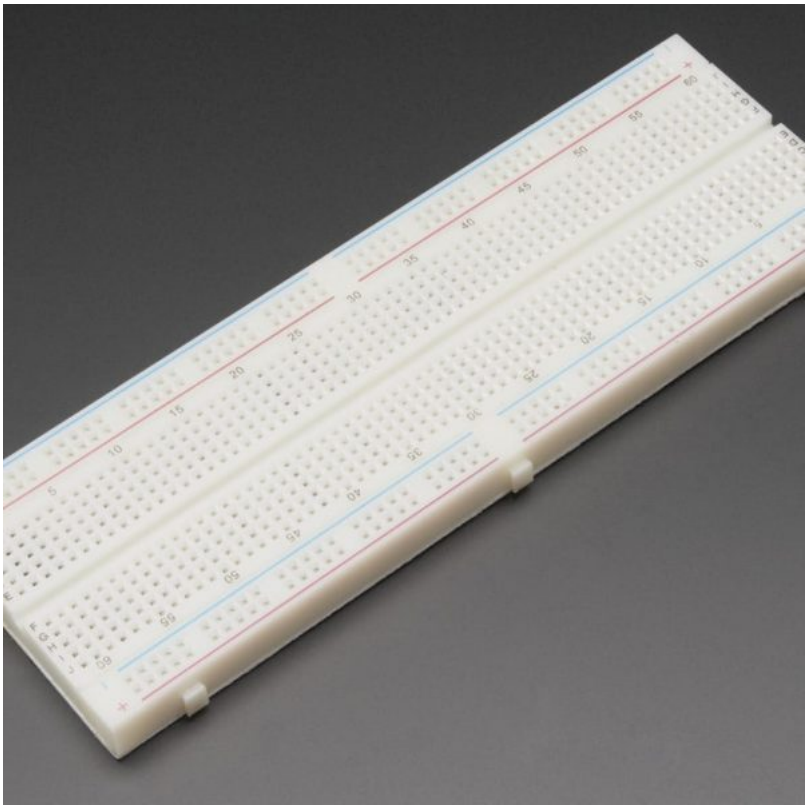
LCD 16×2

مقاومة 220 اوم

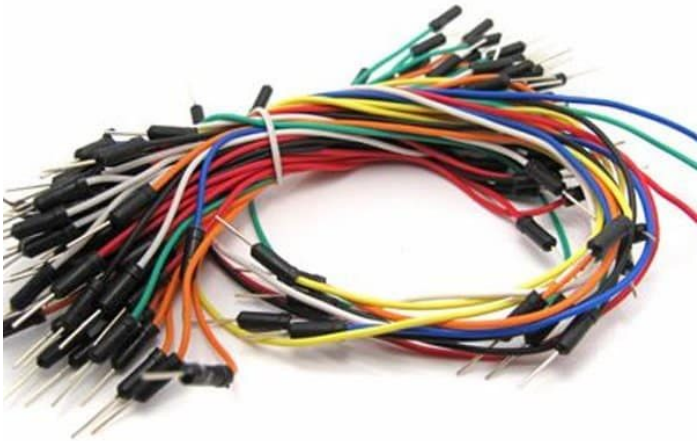
10K Ohms Resistors



Push Buttons

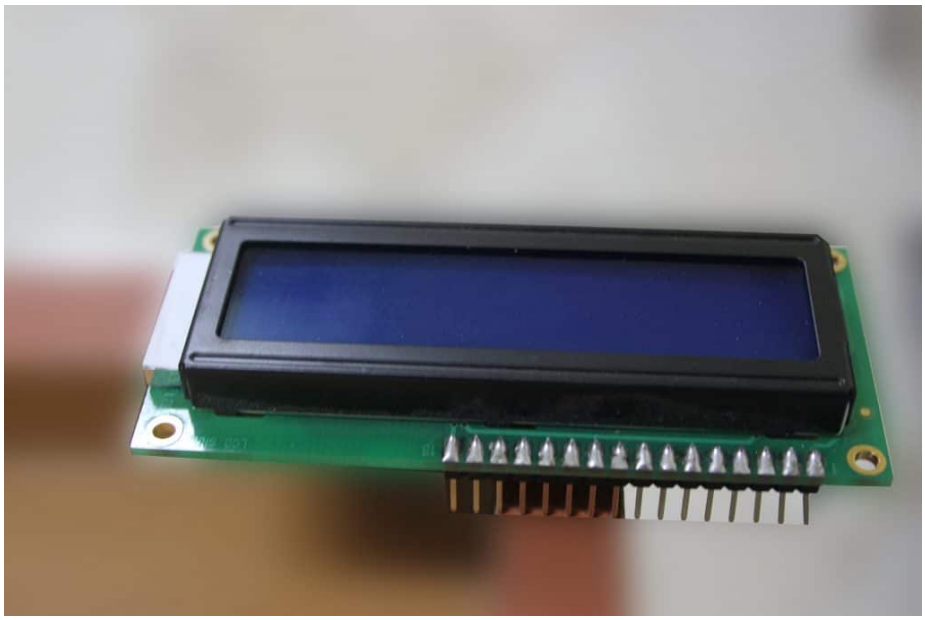


Breadboard



Wires

الشاشة LCD



تعمل الشاشة في احد الحالات التالية:

استقبال أمر من الأردوينو وتنفيذه، على سبيل المثال : أمر التهيئة ومسح الشاشة:

```
lcd.begin(16,2);
lcd.clear();
```

استقبال معلومات من الأردوينو وعرضها، على سبيل المثال : كتابة جملة معينة :

```
lcd.print("Hello");
```

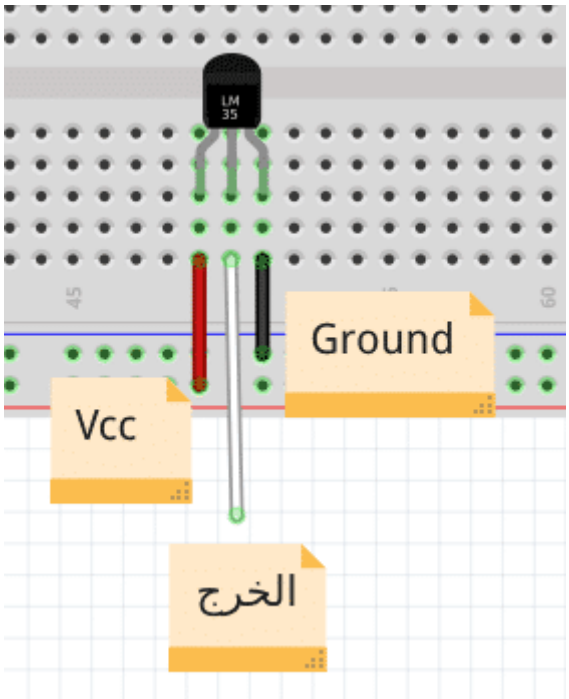
للإطلاع على المزيد حول شاشة الـ LCD قم بالإطلاع على درس التحكم بشاشة LCD

حساس الحرارة LM35

هو عبارة عن عنصر إلكتروني يتأثر بالحرارة ويعطي خرج كهربائي على شكل فولت يمكننا قياسه. أي أن الجهد الكهربائي الناتج منه يتناسب طرديا مع درجة الحرارة فكلما كانت درجة الحرارة عالية كلما كانت الفولتية الناتجة منه عالية.

توصيله في الدارة :

مُخرج هذا الحساس يكون قيمة تناظرية (Analog) ، اي نحتاج إلى توصيله على أحد المنافذ التناظرية (Analog) في الأردوينو.
الأطراف التناظرية في الأردوينو من A0 إلى A5 .



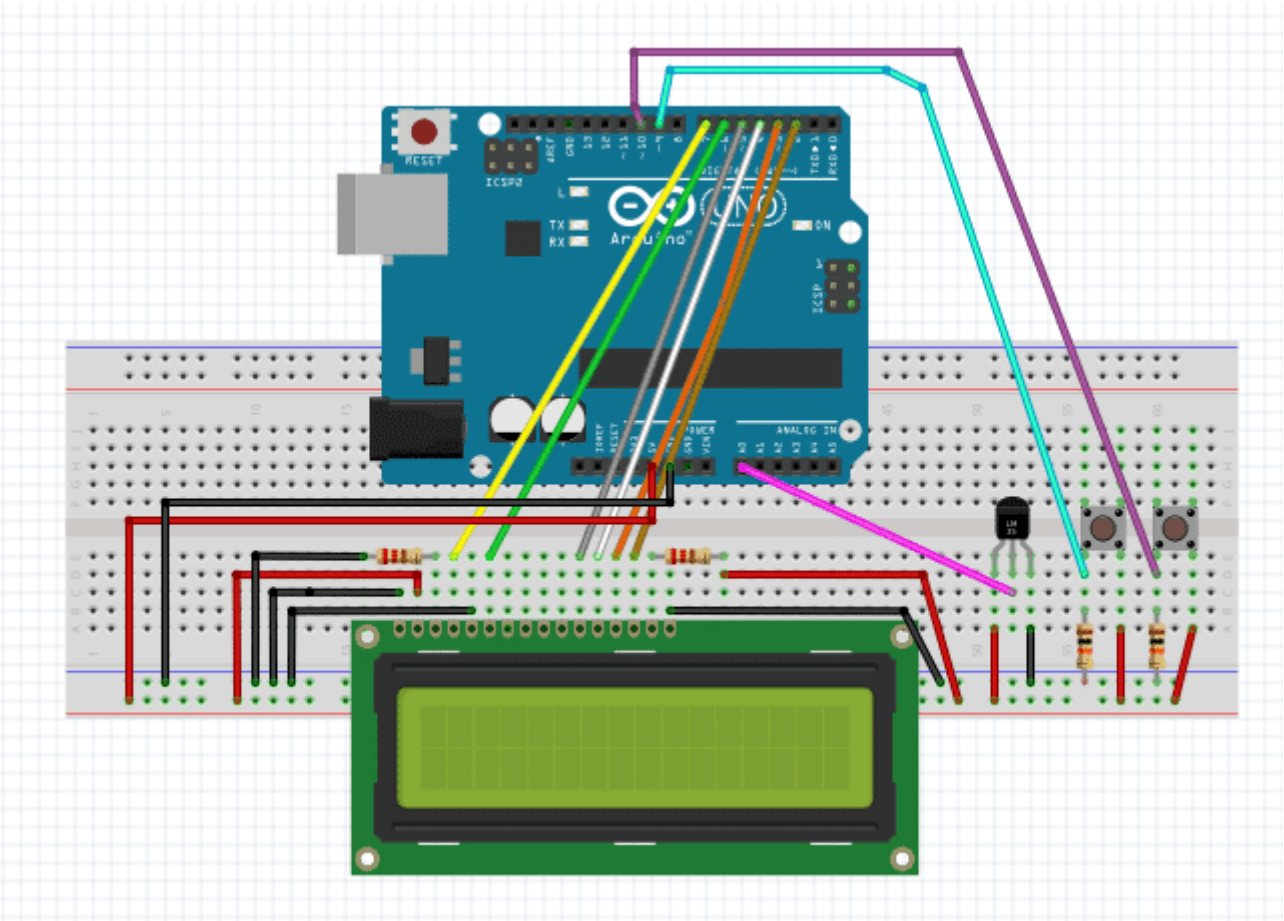
الطرف (بداية من اليسار)	التوصيل
1	Vcc
2	Output To Arduino
3	Ground

ملاحظة هامة :

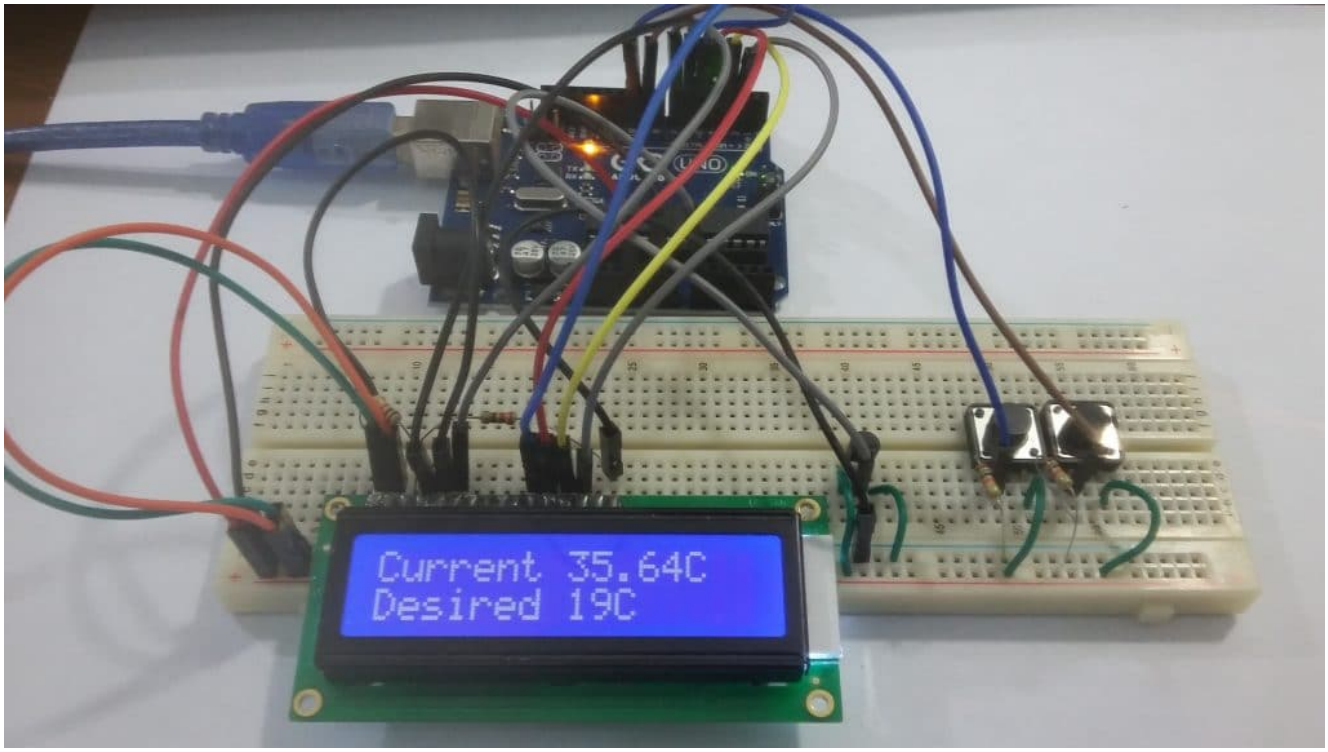
يجب التدقيق في عملية التوصيل، لأنه في حال توصيل الأطراف بشكل خاطىء قد يتسبب في تلف العنصر. ولاحظ عند توصيل العنصر يتم وضعه بحيث تكون الناحية المسطحة مواجهة لنا.

شرح الدارة

قم بتوصيل الدارة كما هو موضح بالصورة التالية :



سيقوم الحساس LM35 بقياس درجة الحرارة بإستمرار وعرضها على شاشة الـ LCD وبإستخدام مفتاح التحكم (Push Buttons) يمكننا التحكم في رفع أو خفض درجة الحرارة المطلوبة في المكان.



الكود البرمجي :

قم بكتابة الكود التالي ورفعه على الأردوينو :

```
#include <LiquidCrystal.h>
#define tempPin A0
#define tup 10
#define tdown 9

float temp;
int defC = 23;
int upstate = 0;
int downstate = 0;

LiquidCrystal lcd(7, 6, 5, 4, 3, 2);

void setup() {
  lcd.begin(16, 2);
  lcd.setCursor(0, 0);
  lcd.print("Config..");
  delay(900);
  lcd.clear();
  pinMode(tup, INPUT);
  pinMode(tdown, INPUT);
}

void loop() {
  upstate = digitalRead(tup);
  downstate = digitalRead(tdown);
  int tvalue = analogRead(tempPin);
  if (upstate == HIGH) {
```

```
    defC = defC + 1;
  }
  if (downstate == HIGH) {
    defC = defC - 1;
  }
  temp = tvalue * 0.48828;

  lcd.setCursor(0, 0);
  lcd.print("Current ");
  lcd.print (temp);
  lcd.print ('C');
  lcd.setCursor (0, 1);
  lcd.print ("Desired ");
  lcd.print (defC);
  lcd.print ('C');

  delay(200);
}
```

شرح الكود :

في البداية، نقوم بإضافة المكتبة الخاصة بشاشة الـ LCD :

```
#include <LiquidCrystal.h>
```

ثم نقوم بتسمية منافذ الأردوينو المستخدمة في المشروع :

```
#define tempPin A0
#define tup 10
#define tdown 9
```

بعد ذلك، نقوم بالإعلان عن المتغيرات التي سيتم استخدامها في البرنامج لتسجيل قيم درجات الحرارة :

```
float temp;
int defC = 23;
int upstate = 0;
int downstate = 0;
```

يتم تسجيل درجة الحرارة الناتجة من حساس الحرارة على المتغير temp. واستخدام المتغير defC لتسجيل درجة الحرارة المطلوبة. والمتغير upstate لتسجيل حالة المفتاح الأول الخاص برفع درجة الحرارة المطلوبة. والمتغير downstate لتسجيل حالة المفتاح الثاني الخاص بخفض درجة الحرارة المطلوبة.

ثم نقوم بإنشاء المتغير الخاص بشاشة الـ LCD وتحديد الأرجل التي سيتم توصيلها مع الأردوينو :

```
LiquidCrystal lcd(7, 6, 5, 4, 3, 2);
```

في الدالة ()setup، نقوم بضبط الإعدادات اللازمة، كإعدادات شاشة الـ LCD وضبط المفاتيح (Push Buttons) كمخرج :

```
void setup() {
```

```
  lcd.begin(16, 2);
  lcd.setCursor(0, 0);
  lcd.print("Config..");
  delay(900);
  lcd.clear();
  pinMode(tup, INPUT);
  pinMode(tdown, INPUT);
}
```

في الدالة ()loop، نقوم بقراءة المفاتيح (Push Buttons) ، وقراءة قيمة الجهد الناتج من الحساس :

```
  upstate = digitalRead(tup);
  downstate = digitalRead(tdown);
  int tvalue = analogRead(tempPin);
```

ثم نقوم بإختبار ما إذا تم الضغط على المفاتيح. فعند الضغط على مفتاح زيادة درجة الحرارة المطلوبة نقوم بإضافة 1 إلى درجة الحرارة المسجلة في المتغير defC. ويتم العكس عند الضغط على مفتاح خفض درجة الحرارة المطلوبة :

```
   if (upstate == HIGH) {
    defC = defC + 1;
  }
  if (downstate == HIGH) {
    defC = defC - 1;
  }
```

ونقوم بإستخدام قيمة الجهد الناتج من الحساس لإيجاد قيمة درجة الحرارة عن طريق المعادلة التالية :

```
   temp = tvalue * 0.48828;
```

وأخيرا، يتم عرض درجة الحرارة الحالية وأيضا درجة الحرارة المطلوبة على شاشة الـ LCD :

```
  lcd.setCursor(0, 0);
  lcd.print("Current ");
  lcd.print (temp);
  lcd.print ('C');
  lcd.setCursor (0, 1);
  lcd.print ("Desired ");
  lcd.print (defC);
  lcd.print ('C');

  delay(200);
```