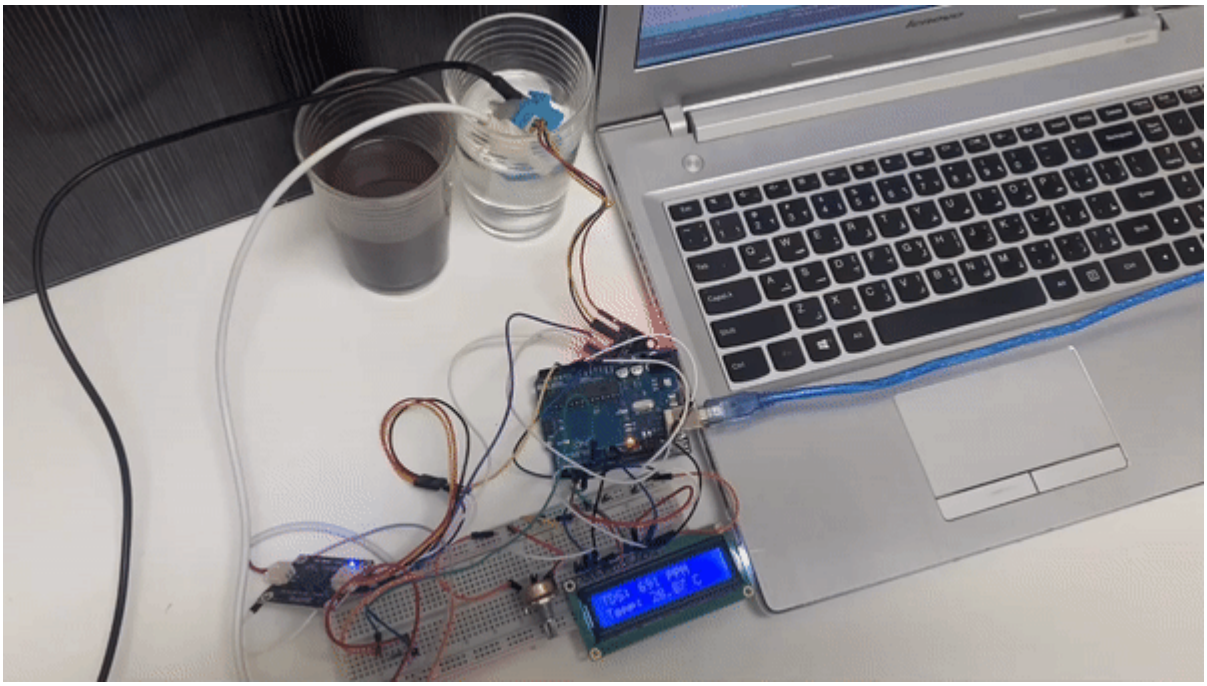
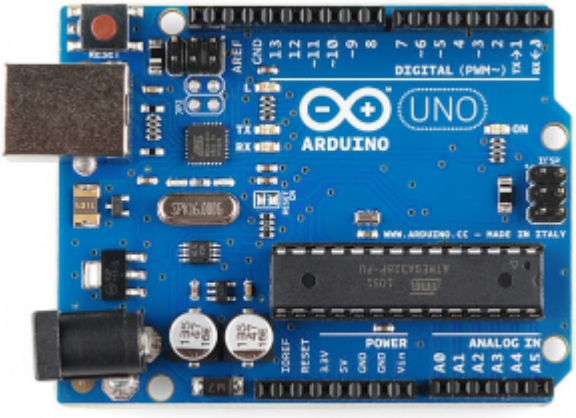
نظام لفحص جودة الماء باستخدام الاردوينو

مقدمة

إجراء اختبارات كيميائية للماء تساعدنا ذلك على تحديد المواد الكيميائية الذائبة في الماء ومصدر الماء وطريقة تنقيته وتحديد الاستخدامات الممكنة له، في هذا الدرس ستتعلم كيف تفحص جودة الماء باستخدام الاردوينو ومختلف الحساسات.

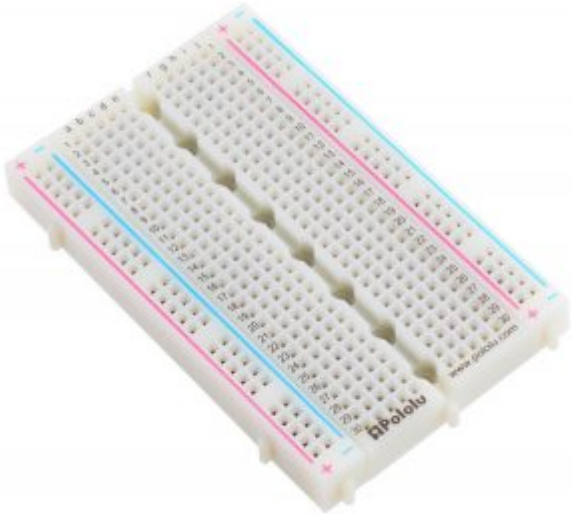


المواد والأدوات



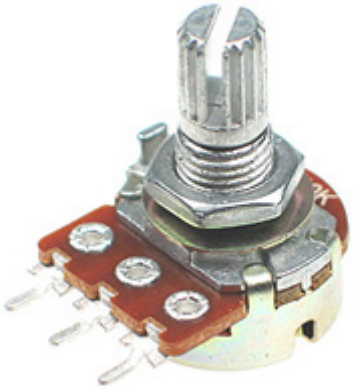
1× اردوينو اونو

1× سلك الاردوينو



1× لوحة تجارب – حجم كبير

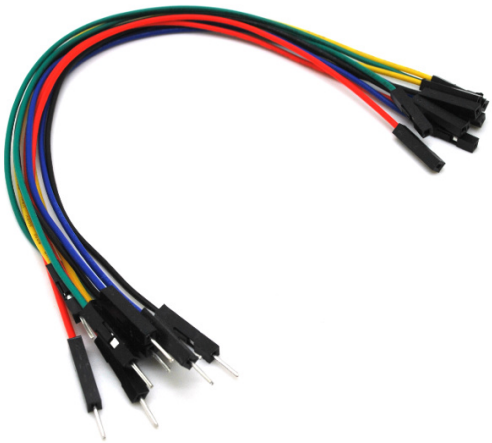
1× شاشة كرستالية



1× مقاومة متغيرة

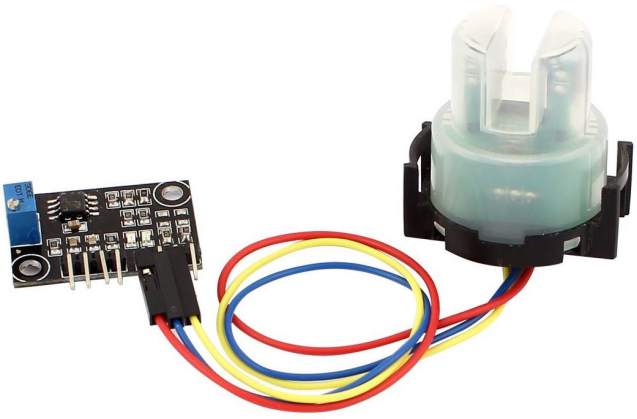
حزمة أسلاك توصيل (ذكر- ذكر)

1× مقاومة 4.7 Ω

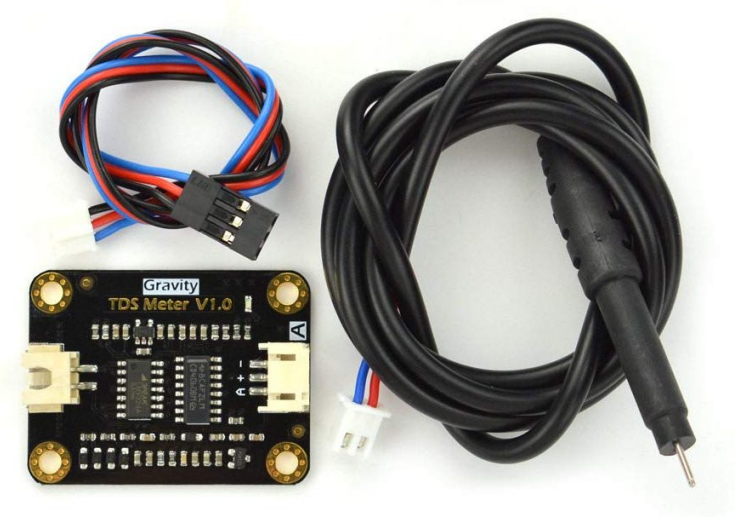


حزمة أسلاك توصيل (ذكر – أنثى)

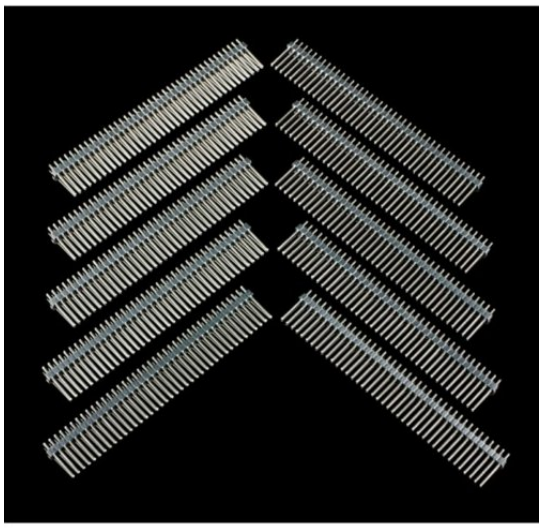
1× مستشعر درجة الحرارة المقاوم للماء



1× مستشعر شفافية الماء



1× مستشعر ملوحة الماء

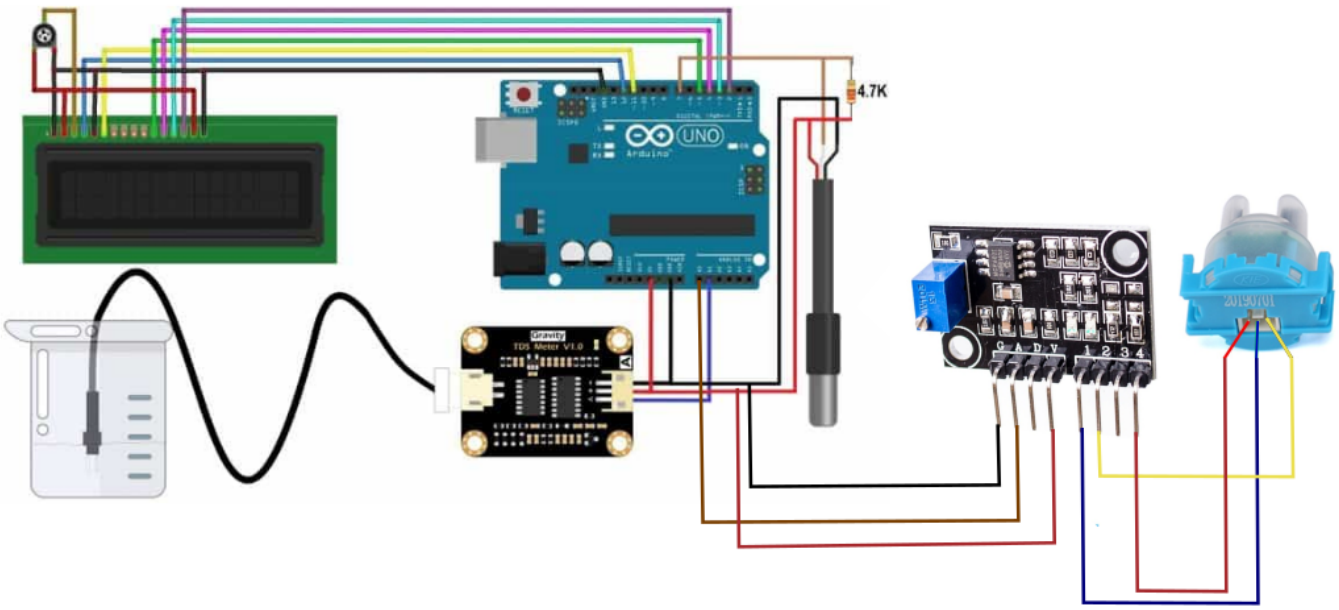


1× 40 رأس دبوس

توصيل الدائرة

لمعرفة المزيد حول الشاشة الكرستالية يمكنك الرجوع للدرس التحكم بالشاشة الكرستالية LCD

لابد من تلحيم المنافذ مع الشاشة الكرستالية، للمزيد حول اللحام يمكنك الرجوع للدرس تعلم كيفية التلحيم – تلحيم القطع باللوحة الإلكترونية



الكود البرمجي

قبل رفع الكود البرمجي على لوحة الاردوينو حمّل المكتبات اللازمة <EEPROM.h> “GravityTDS.h”> <<DallasTemperature.h>.

```
#include <LiquidCrystal.h>
#include <EEPROM.h>
#include "GravityTDS.h"
#include <OneWire.h>
#include <DallasTemperature.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

#define ONE_WIRE_BUS 7
#define TdsSensorPin A1

OneWire oneWire(ONE_WIRE_BUS);
GravityTDS gravityTds;

DallasTemperature sensors(&oneWire);

float tdsValue = 0;

void setup()
{
lcd.begin(16,2);
sensors.begin();
gravityTds.setPin(TdsSensorPin);
gravityTds.setAref(5.0); //reference voltage on ADC, default 5.0V on Arduino UNO
gravityTds.setAdcRange(1024); //1024 for 10bit ADC;4096 for 12bit ADC
gravityTds.begin(); //initialization
}
```

```
void loop()
{
sensors.requestTemperatures();
gravityTds.setTemperature(sensors.getTempCByIndex(0)); // set the temperature and
execute temperature compensation
gravityTds.update(); //sample and calculate
tdsValue = gravityTds.getTdsValue(); // then get the value

lcd.setCursor(0, 0);
lcd.print("TDS: ");
lcd.print(tdsValue,0);
lcd.print(" PPM");

lcd.setCursor(0, 1);
lcd.print("Temp: ");
lcd.print(sensors.getTempCByIndex(0));
lcd.print(" C");

delay(1500);
lcd.clear();

int sensorPin = A0;
int sensorValue = analogRead(sensorPin);
int turbidity = map(sensorValue, 0, 750, 100, 0);

lcd.setCursor(0, 0);
lcd.print("Turbidity:");
lcd.print(turbidity);

delay(1500);
lcd.clear();
}
```

شرح الكود البرمجي

هذا السطر يستدعي مكتبة الشاشة الكرستالية.

نستطيع تحميلها بتتبع المسار التالي:

Sketch > Include libraries > Manage libraries

ثم نكتب بخانة البحث Liquid crystal by Arduino

ثم نضغط على Install.

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

هنا نستدعي مكتبة مستشعر ملوحة الماء.

```
#include "GravityTDS.h"
```

وهنا نستدعي مكتبة مستشعر درجة الحرارة المقاوم للماء.

```
#include <DallasTemperature.h>
```

بعد ذلك تم تعريف المتغيرات اللازمة مثل المتغيرات الخاصة بالشاشة الكرستالية.

```
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

المتغير ONE_WIRE_BUS الخاص بمستشعر الحرارة تم ربطه مع المنفذ الرقمي 7 على لوحة الاردوينو.

والمتغير TdsSensorPin الخاص بمستشعر ملوحة المياه تم ربطه مع المنفذ التناظري A1 على لوحة الاردوينو.

```
#define ONE_WIRE_BUS 7
#define TdsSensorPin A1
```

القيمة الابتدائية لمستشعر ملوحة الماء tdsValue هي صفر.

```
float tdsValue = 0;
```

في الدالة ()setup يتم تهيئة الشاشة الكرستالية لقراءة القيم من المستشعرات.

```
void setup()
{
lcd.begin(16,2);
sensors.begin();
gravityTds.setPin(TdsSensorPin);
gravityTds.setAref(5.0); //reference voltage on ADC, default 5.0V on Arduino UNO
gravityTds.setAdcRange(1024); //1024 for 10bit ADC;4096 for 12bit ADC
gravityTds.begin(); //initialization
}
```

في الدالة ()loop سيتم قراءة القيم من مستشعر ملوحة الماء وسيتم تحديثها للحصول على قراءات مختلفة.

```
void loop()
{
sensors.requestTemperatures();
gravityTds.setTemperature(sensors.getTempCByIndex(0)); // set the temperature and
execute temperature compensation
gravityTds.update(); //sample and calculate
tdsValue = gravityTds.getTdsValue(); // then get the value
```

بعد ذلك سيتم طباعة درجة ملوحة الماء على الشاشة الكرستالية في السطر الأول.

```
lcd.setCursor(0, 0);
lcd.print("TDS: ");
lcd.print(tdsValue,0);
lcd.print(" PPM");
```

وفي السطر الثاني سيتم طباعة درجة حرارة الماء المقروءة من مستشعر الحرارة بوحدة C.

```
lcd.setCursor(0, 1);
lcd.print("Temp: ");
lcd.print(sensors.getTempCByIndex(0));
lcd.print(" C");
```

سيتم عرض درجة الملوحة ودرجة الحرارة لمدة ثانية ونصف ثم سيتم مسح الشاشة الكرستالية لتهيئتها استعدادًا لطباعة قيمة شفافية الماء.

```
delay(1500);
lcd.clear();
```

المتغير sensorPin الخاص بمستشعر شفافية الماء تم ربطه مع المنفذ التناظري A0 على لوحة الاردوينو.

سيتم قراءة شفافية الماء وطباعة القيمة على الشاشة الكرستالية في السطر الأول.

ستقل القيمة إذا زاد التعكر في الماء، وستزيد القيمة إذا زادت شفافية الماء.

```
int sensorPin = A0;
int sensorValue = analogRead(sensorPin);
int turbidity = map(sensorValue, 0, 750, 100, 0);

lcd.setCursor(0, 0);
lcd.print("Turbidity:");
lcd.print(turbidity);
```

تأكد بأن النظام يعمل بالشكل الصحيح.

لا تنسَ فصل وحدة الطاقة بعد الانتهاء من استخدام نظام فحص جودة الماء.