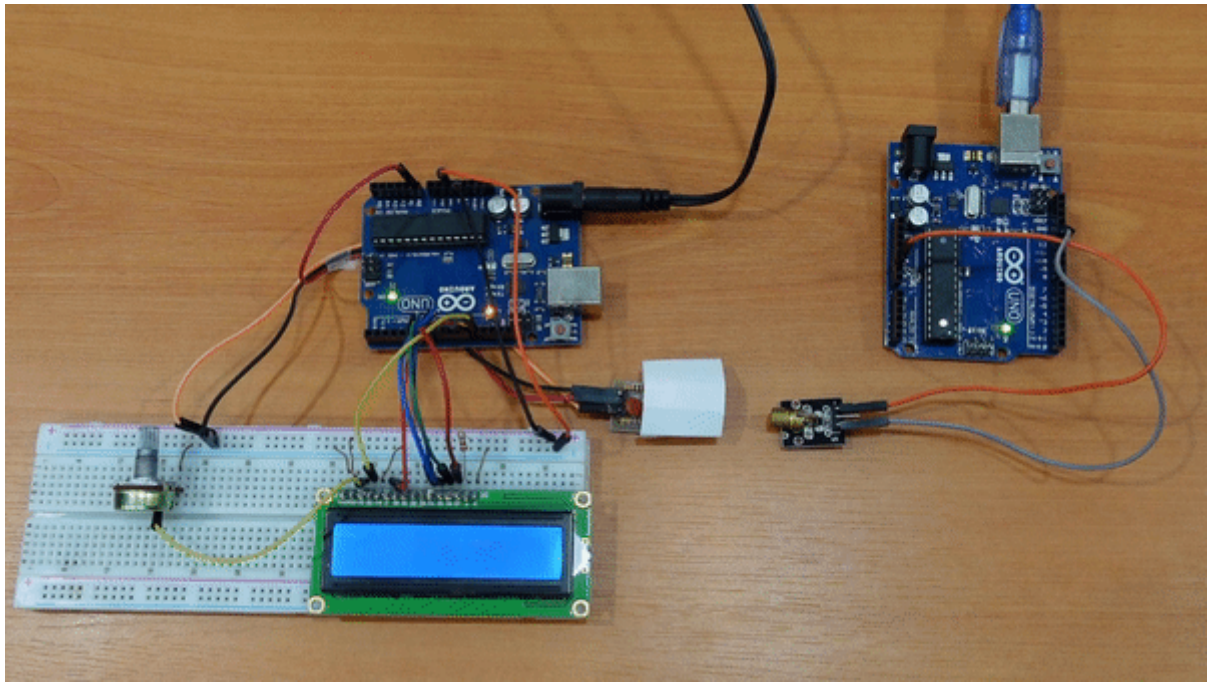
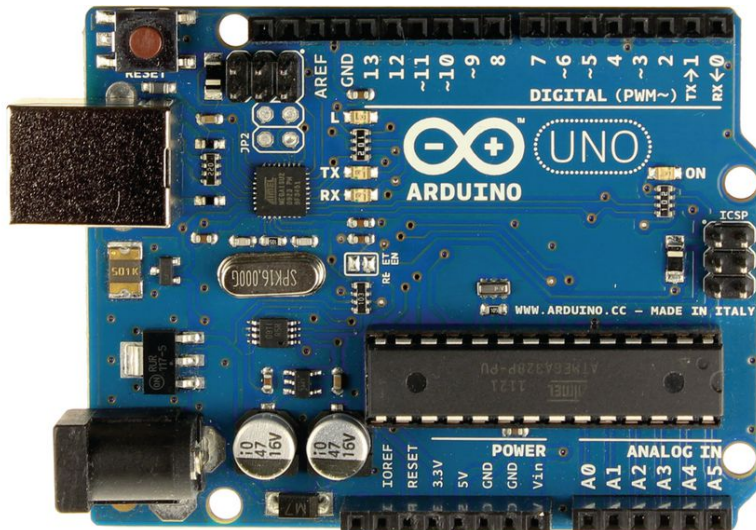


نقل البيانات بين 2 أردوينو باستخدام الليزر

يعتبر الضوء من الوسائل الأكثر سرعة في نقل البيانات، في هذا الدرس سنتعلم نقل البيانات بين جهازين أردوينو لا سلكيا و باستخدام أشعة الليزر بحيث يكون الازدوينو الأول جهاز ارسال يعمل على تشفير النصوص ، و الثاني جهاز استقبال يعمل على فك شفرة النصوص و طباعتها على الشاشة الكرسالة



المواد و الأدوات



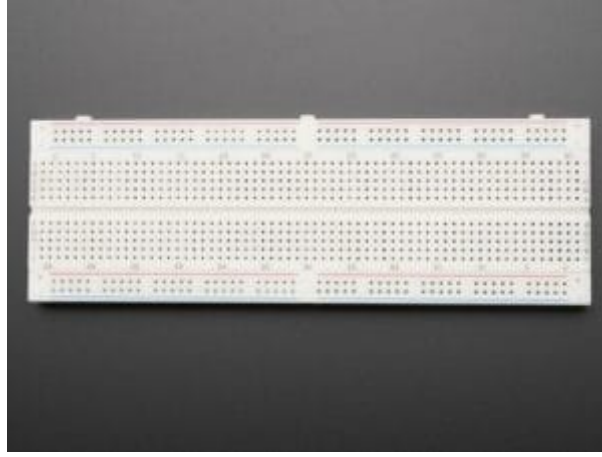
2 X اردوينو أونو



1 X سلك أردوينو



1 X محول طاقة



1 X لوحة تجارب

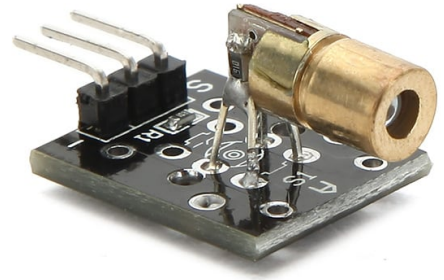


1 X شاشة كرسطالية

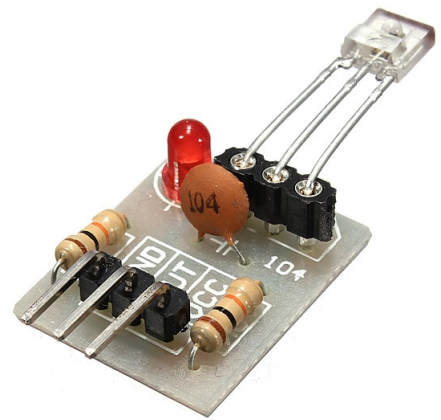
1X رأس دبوس



1 X مقاومة 220 أوم



1 X مرسل ليزر



1 X مستقبل ليزر



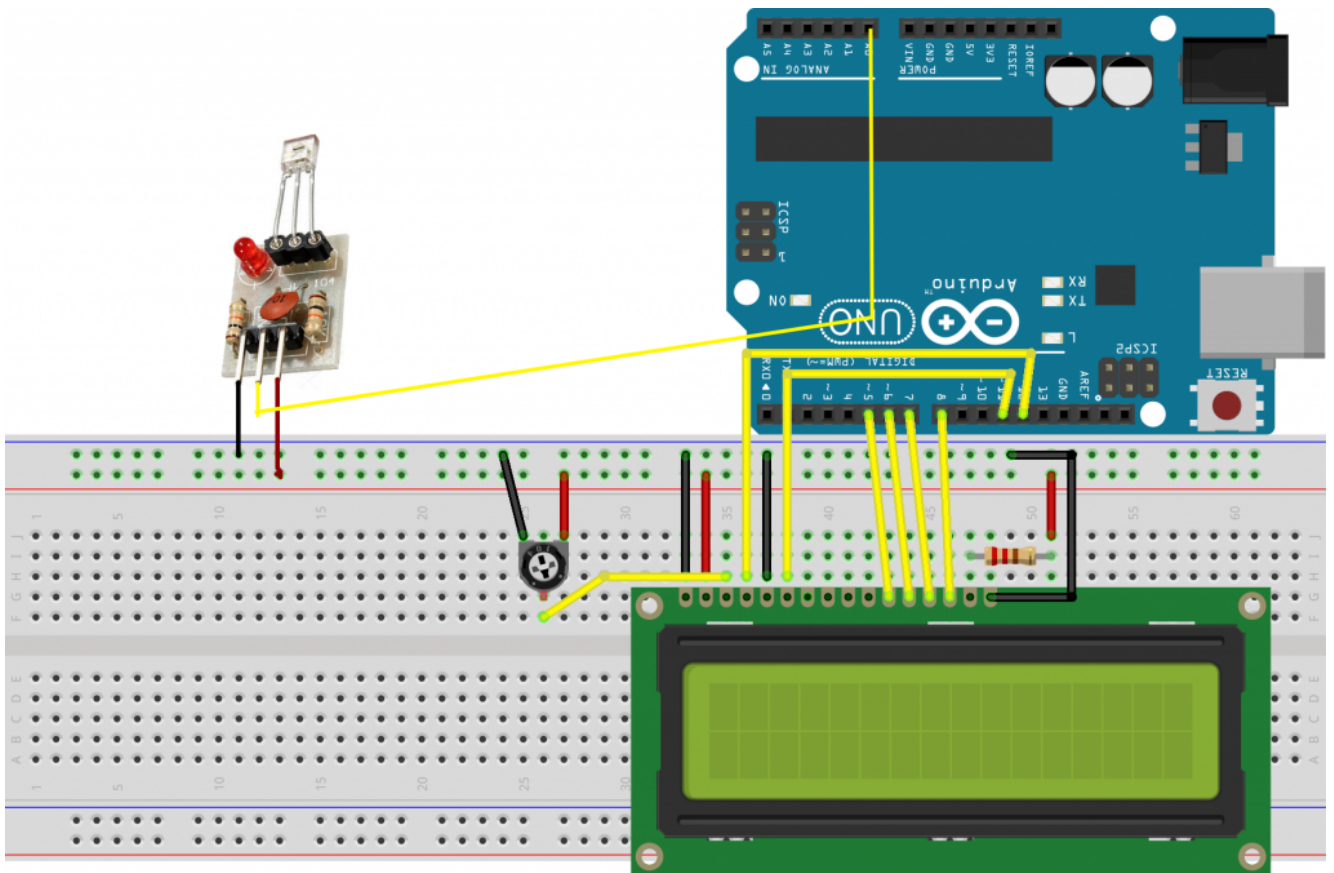
أسلاك توصيل (أنثى/ذكر)



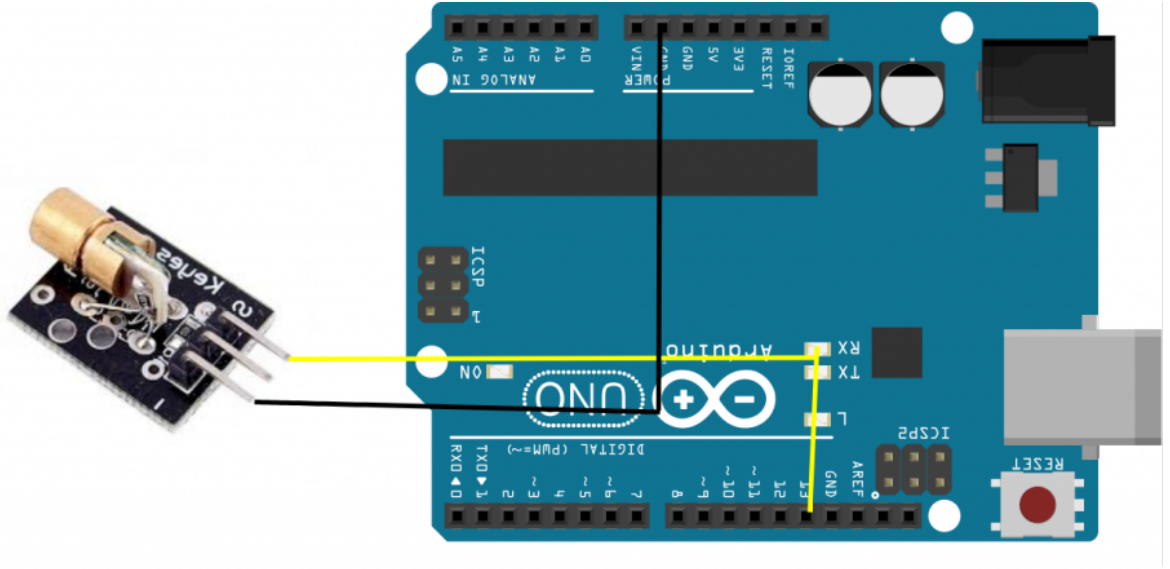
أسلاك توصيل (ذكر/ذكر)

توصيل الدائرة

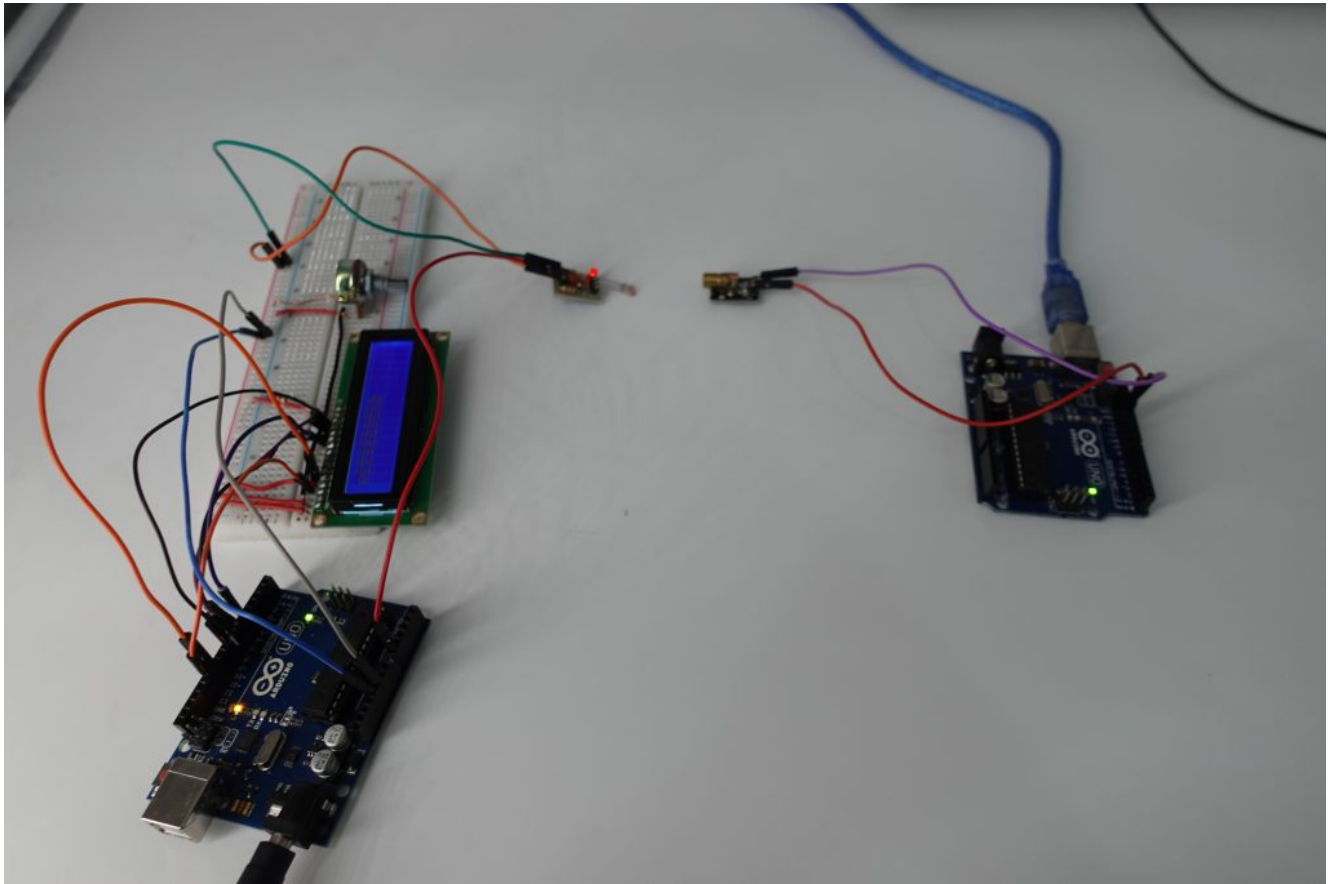
دائرة المستقبل :



دائرة المرسل:



يجب أن يثبت المرسل و المستقبل على خط واحد كما هو ظاهر بالصورة



الكود البرمجي

الكود البرمجي للمرسل يتم رفعه على الأردوينو الذي سيرسل البيانات

```
int ledPin = 13;

void setup() {
  pinMode(ledPin, OUTPUT);
}
```

```

    Serial.begin(9600);
}

void loop() {
    byte input;
    while (Serial.available() > 0) {
        input = Serial.read();
        digitalWrite(ledPin, HIGH);
        delay(10);
        digitalWrite(ledPin, LOW);

        for (int i = 0; i < 8; i++) {
            digitalWrite(ledPin, (input & (1 << i)) >> i);
            delay(100);
        }
        digitalWrite(ledPin, LOW);
        delay(10);
    }
}

```

الكود البرمجي للمستقبل يتم رفعه على الأردوينو الذي يستقبل بيانات

```

#define NUM_SAMPLES 10
int sensorPin = A0;    // select the input pin for the potentiometer
int sensorValue = 0;   // variable to store the value coming from the sensor
double average;
#include <LiquidCrystal.h> //Load Liquid Crystal Library
LiquidCrystal lcd(12, 11, 5, 6, 7, 8);

void setup() {
    lcd.begin(16,2);
    Serial.begin(9600);
    sensorValue = analogRead(sensorPin);
    average = sensorValue;
}

void loop() {
    // read the value from the sensor:
    sensorValue = analogRead(sensorPin);
    boolean high = false;
    high = isSignalHigh(average, sensorValue);
    if (high) {
        byte incoming = 0;
        delay(10);
        for (int i = 0; i < 8; i++) {
            incoming |= ( isSignalHigh(average, analogRead(sensorPin)) << i);
            delay(100);
        }
        lcd.print((char)incoming); //Print measured distance
        Serial.print((char)incoming);
    }
}

```

```

    average = approxRollingAverage(average, sensorValue);
}

double approxRollingAverage(double avg, double new_sample) {

    avg -= avg / NUM_SAMPLES;
    avg += new_sample / NUM_SAMPLES;

    return avg;
}

boolean isSignalHigh(double average, double sample) {
    if (sample - average > 10)
        return true;
    return false;
}

```

شرح الكود البرمجي

شرح الكود البرمجي للمرسل

نعرف المنفذ الذي سيتم توصيله مع مرسل أشعة الليزر به باسم (ledpin) الذي تم توصيله مع منفذ رقم 13

```
int ledPin = 13;
```

نعين المنفذ (ledpin) كمخارجات
و نفعل الاتصال التسلسلي

```

void setup() {
// initialize the digital pin as an output.
pinMode(ledPin, OUTPUT);
Serial.begin(9600);
}

```

في الدالة المتكرره نفعل القراءة من شاشة الاتصال التسلسلي

```

void loop() {

byte input;

// Read character from serial
while (Serial.available() > 0) {
input = Serial.read();
}

```

نعطي أوامر بتشغيل الليزر لارسال البيانات من أردوينو باستخدام الليزر

```

digitalWrite(ledPin, HIGH);
delay(10);

```



```
digitalWrite(ledPin, LOW);
```

نحدد تشغيل الليد بناء على معادلة بين متغير له قيم بين 0 و 8 و المدخلات من شاشة الاتصال التسلسلي

```
for (int i = 0; i < 8; i++) {  
digitalWrite(ledPin, (input & (1 << i)) >> i);  
delay(100);  
}  
digitalWrite(ledPin, LOW);  
  
delay(10);  
}  
}
```

شرح الكود البرمجي للمستقبل

نعرف منفذ مستقبل الليزر و منافذ شاشة الـ LCD

```
#define NUM_SAMPLES 10  
int sensorPin = A0;  
int sensorValue = 0;  
double average;  
#include <LiquidCrystal.h> //Load Liquid Crystal Library  
LiquidCrystal lcd(12, 11, 5, 6, 7, 8);
```

نحدد العدد الأقصى لخانات السطر الواحد وعدد السطور الأقصى التي تسمح بها الشاشة .
و نحدد متغير باسم sensorValue للقيم التي يتم قراءتها من الحساس

```
void setup() {  
lcd.begin(16,2);  
Serial.begin(9600);  
sensorValue = analogRead(sensorPin);  
average = sensorValue;  
}
```

لقراءة البيانات المرسلة من أردوينو باستخدام الليزر و تحويلها إلى نصوص و عرضها على الشاشة الكرسطالية

```
void loop() {  
  
sensorValue = analogRead(sensorPin);  
boolean high = false;  
high = isSignalHigh(average, sensorValue);  
if (high) {  
byte incoming = 0;  
delay(10);  
for (int i = 0; i < 8; i++) {  
incoming |= ( isSignalHigh(average, analogRead(sensorPin)) << i);  
delay(100);  
}  
lcd.print((char)incoming); //Print measured distance  
Serial.print((char)incoming);  
}
```

```
}  
average = approxRollingAverage(average, sensorValue);  
}  
  
double approxRollingAverage(double avg, double new_sample) {  
  
    avg -= avg / NUM_SAMPLES;  
    avg += new_sample / NUM_SAMPLES;  
  
    return avg;  
  
}  
  
boolean isSignalHigh(double average, double sample) {  
    if (sample - average > 10)  
        return true;  
    return false;  
  
}
```