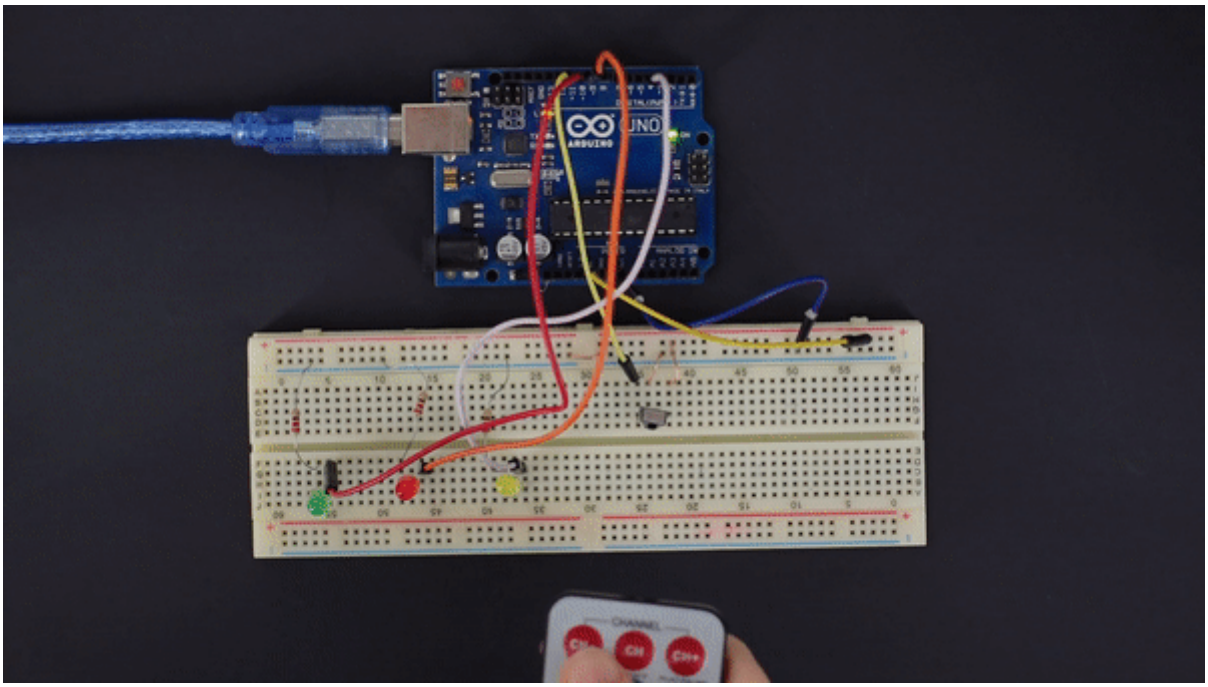
التحكم بالإضاءة باستخدام مستقبل الأشعة تحت الحمراء(IRreceiver)

المقدمة

في هذا الدرس سنتعرف على طريقة برمجة مستقبل الأشعة تحت الحمراء (IR receiver) مع الأردوينو، ليساعدنا في التحكم بإضاءة الثنائي المشع للضوء، حيث نستخدم ريموت تحكم عن بعد لتشغيل الإضاءة، يمكن أن يساعدنا هذا المشروع ، للتحكم في الأجهزة المنزلية عن بعد و لاسلكيا.



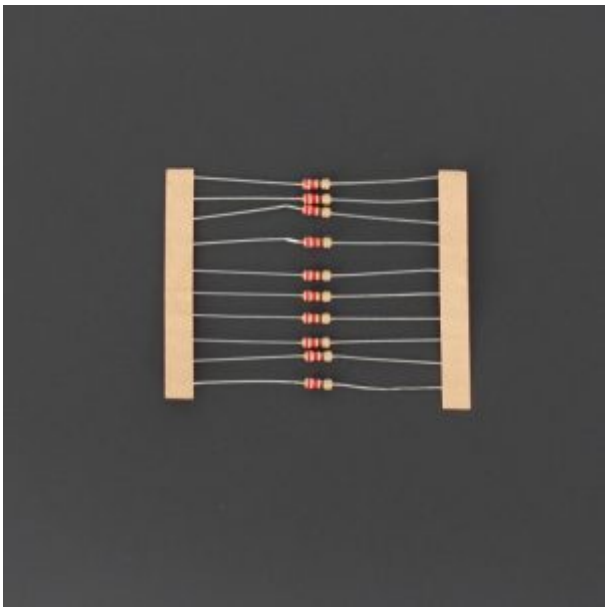
المواد والأدوات

X1 اردوينو أونو

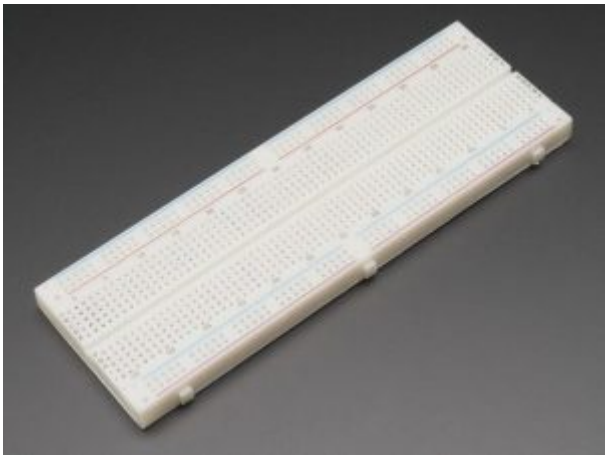
X1 سلك الأردوينو

X1مستقبل الإشارات تحت الحمراء(IR receiver)

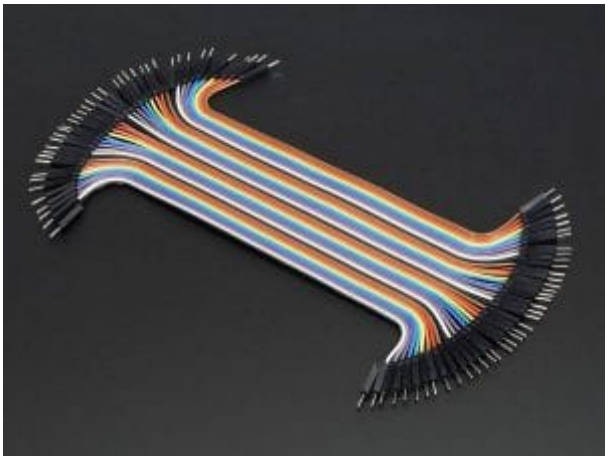
X3 الثنائي المشع للضوء



X3 مقاومة 220أوم



X1 لوحة تجارب



مجموعة أسلاك توصيل (ذكر/ذكر)

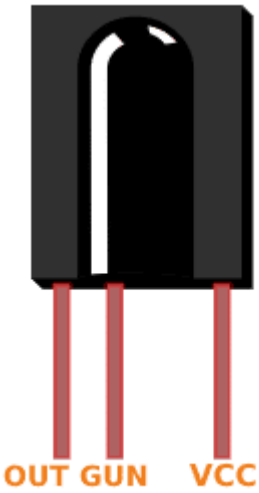
مستقبل الأشعة تحت الحمراء(IRreceiver)

يعد الاتصال بالأشعة تحت الحمراء (IR) تقنية لاسلكية مستخدمة على نطاق واسع وسهلة التنفيذ ولها العديد من التطبيقات المفيدة. من أبرز الأمثلة في الحياة اليومية التلفزيون

الأشعة تحت الحمراء

الأشعة تحت الحمراء هي شكل من أشكال الضوء مشابه للضوء الذي نراه من حولنا. الفرق الوحيد بين ضوء الأشعة تحت الحمراء والضوء المرئي هو التردد وطول الموجة. تقع الأشعة تحت الحمراء خارج نطاق الضوء المرئي ، لذلك لا يمكن للبشر رؤيتها:

مستقبل الأشعة تحت الحمراء



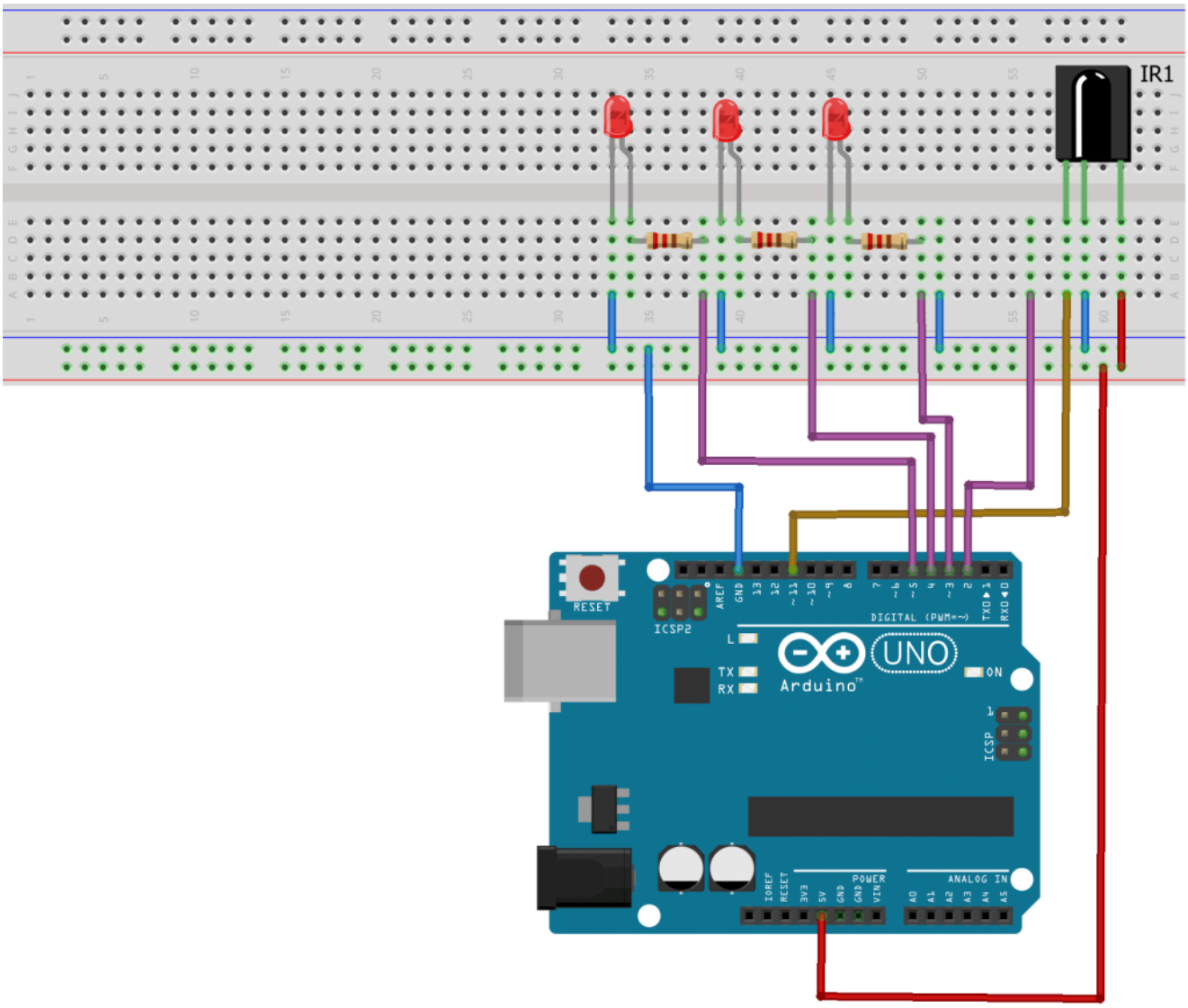
فك اشارات ريموت التحكم باستخدام الأردوينو

Label	Value
-CH	0x00FFA25D
CH	0x00FF629D
+CH	0x00FFE21D

>>|	0x00FF22DD
|<<	0x00FF02FD
||<	0x00FFC23D
+	0x00FFE01F
–	0x00FFA857
EQ	0x00FF906F
0	0x00FF6897
+100	0x00FF9867
+200	0x00FFB04F
1	0x00FF30CF
2	0x00FF18E7
3	0x00FF7A85
4	0x00FF10EF
5	0x00FF38C7
6	0x00FF5AA5
7	0x00FF42BD
8	0x00FF4AB5
9	0x00FF52AD

توصيل الدائرة

قم بتوصيل الدائرة كما هو موضح بالصورة التالية :



الكود البرمجي

```
#include <IRremote.h>
int LED1= 3;
int LED2= 4;
int LED3= 5;

int RECV_PIN = 2;
IRrecv irrecv(RECV_PIN);
decode_results results;

void setup() {
   Serial.begin(9600);
   irrecv.enableIRIn(); // Start the receiver
    /***Pin mode declaration***/
    pinMode(LED1, OUTPUT);
    pinMode(LED2, OUTPUT);
    pinMode(LED3, OUTPUT);
  }
  void loop() {
    if (irrecv.decode(&results)) {
```

```
    switch (results.value)
    {
      case 0xFFA25D:
      digitalWrite(red, HIGH);
      delay(300);
      digitalWrite(red, LOW);
      break;
      case 0xFF629D:
      digitalWrite (green, HIGH);
      delay(300);
      digitalWrite(green, LOW);
      break;
      case 0xFFE21D:
      digitalWrite (blue, HIGH);
      delay(300);
      digitalWrite(blue, LOW);
       break;
}
   irrecv.resume();

    }
 }
```

شرح الكود :

في البداية نقوم بتعريف متغيرات للإشارة إلى أرقام منافذ الأردوينو المستخدمة في المشروع. , واستدعاء مكتبة التحكم ب IRreceiver ، قم بتحميل المكتبة من هنا

```
int red = 3;
int green = 9;
int blue = 10;
#include <IRremote.h>
int RECV_PIN = 11
```

تعريف كائن (object) من النوع IRrecv ، وتعريف متغيرات لحفظ قيم سطوع إضاءة الثلاث ألوان الخاصة بالRGB (أحمر،أخضر،أزرق).

```
IRrecv irrecv(RECV_PIN);
decode_results results;
int redBrightness = 0;
int greenBrightness = 0;
int blueBrightness = 0;
```

في الدلة ()setup ، نقوم بضبط الإعدادات اللازمة مثل ضبط المنافذ الموصلة مع الـ LEDs كمخرج وتهيئة معدل سرعة نقل البيانات عبرالمنفذ التسلسلي (Serial Port) لاستقبال بيانات من الأردوينو :

```
void setup() {
   Serial.begin(9600);
```

```
  irrecv.enableIRIn();

   pinMode(red, OUTPUT);
   pinMode(green, OUTPUT);
   pinMode(blue, OUTPUT);
 }
```

إنشاء دالة ()setcolor والتي تقوم بحساب شدة الإضاءة الخاصة بكل لون في RGB اعتمادا على نوعه إما مصعد مشترك (Common anode) أو مهبط مشترك (Common Cathode)، ثم ارسال القيم عبر المنافذ الخاصة بالـ RGB ليتم إضاءة اللون المطلوب .

```
void setcolor (int redx, int greenx, int bluex){
   Serial.print("Led");
   int red1= 255 - redx ;
   int green1=255 - greenx;
   int blue1=255 - bluex;
   analogWrite(red, red1);
   analogWrite(green, green1);
   analogWrite(blue, blue1);
 }
```

في الدالة loop() نقوم بقراءة القيمة التي يستقبلها IR receiver عند الضغط على أزرار الريموت و تحديد اللون الذي سيظهر على RGB اعتمادا على القيمة المقروءة .

```
void loop() {
    if (irrecv.decode(&results)) {
    switch (results.value)
    {
      case 0xE318261B:
        setcolor (255, 0, 0);
        break;
      case 0x511DBB:
        setcolor (0, 0, 255);
        break;
      case 0xFFE21D:
        setcolor (0, 255, 0);
        break;
      }
   irrecv.resume();

    }
 }
```