گزارش دوم گروه ۳

سپهر امینی افشار

متینا مهدی زاده

**خواسته‌های کارفرما**

* در این پروژه ما این قابلیت را به رزبری پای همراه سرباز اضافه کنیم به طوری که وقتی دکمه x را بزند عکس بگیرد و وقتی دکمه y را بزند فیلم

گرفتن شروع شود تا وقتی که دکمه z زده شود تا فیلم گرفتن پایان یابد. در هر دوی این حالات بایستی که اطلاعات ذخیره شده فشرده شود و از طریق شبکه ارتباطی بین سربازان و رهبر گروه منتقل شود.

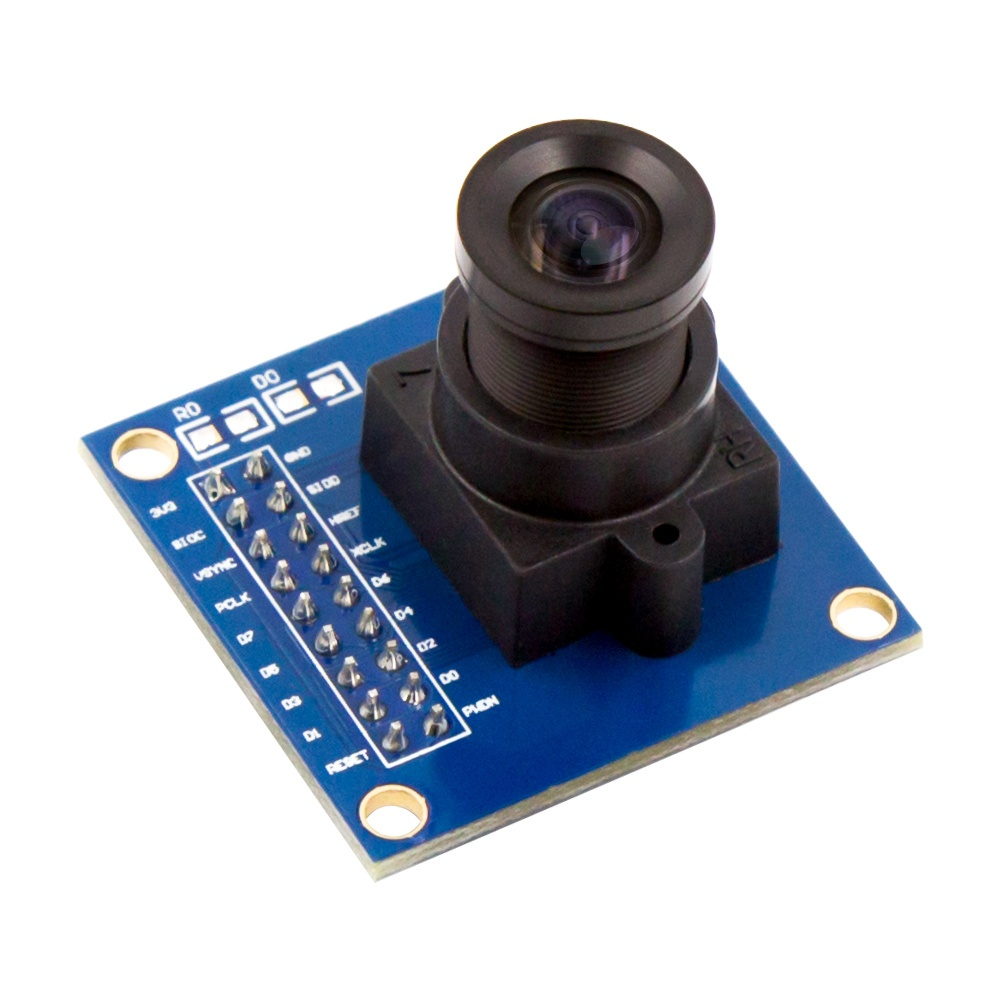
**محدودیت‌های سخت افزاری**

* از آنجایی که این وسیله قرار است با سرباز حمل شود،‌ باید تا حد امکان کم وزن باشد تا مشکلی برای سرباز ایجاد نکند. همچنین باید فضای کمی را اشغال کند تا هم در مسیرهای سخت سالم بماند هم میزان تشخیص‌پذیری او را افزایش ندهد.
* همچنین چون دوربین‌های موجود در بازار صدا را ضبط نمی‌کنند لازم است تا ماژول میکروفون هم به سرباز وصل شود و به همراه فایل فیلم ارسال شود.
* ما از برد Arduino UNO استفاده می‌کنیم و ماژول‌های انتخابی باید با این مدل سازگار باشد.

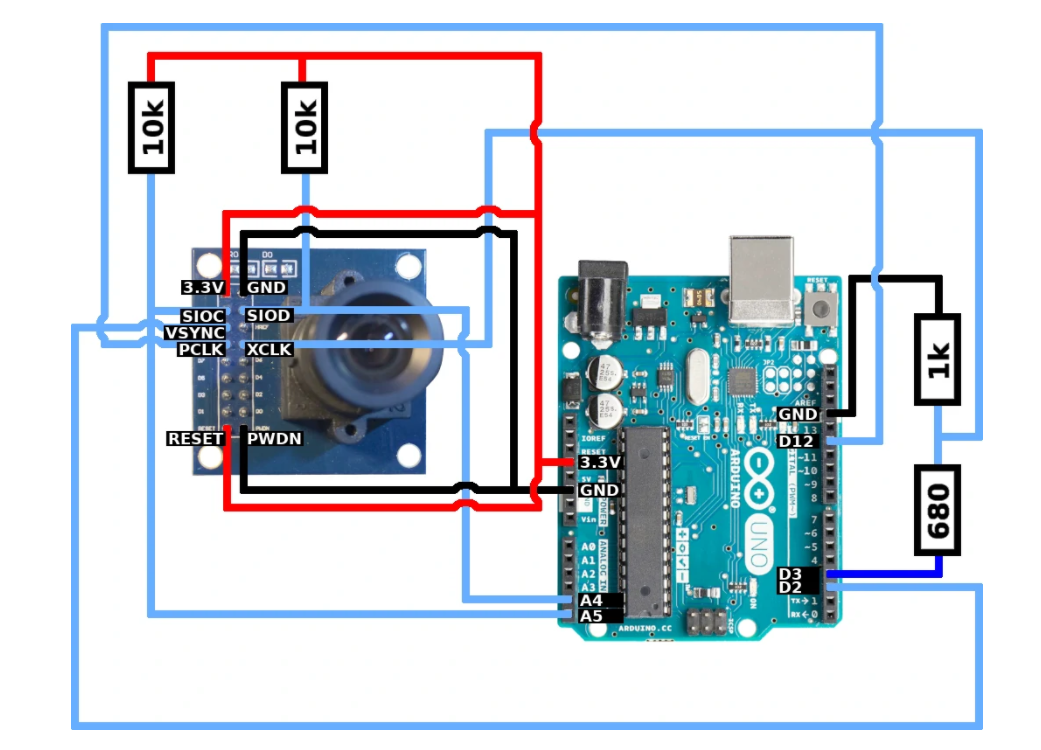
**ماژول‌های مورد استفاده**

* ماژول دوربین رزبری پای مدل [OV7670](https://www.itbazar.com/p_6833.aspx)
* ماژول بازر KY-012
* ماژول تشخیص فاصله SRF05 - Ultrasonic Ranger

**ماژول دوربین مدل OV7670**

****

تصویر۱: ماژول دوربینOV7670

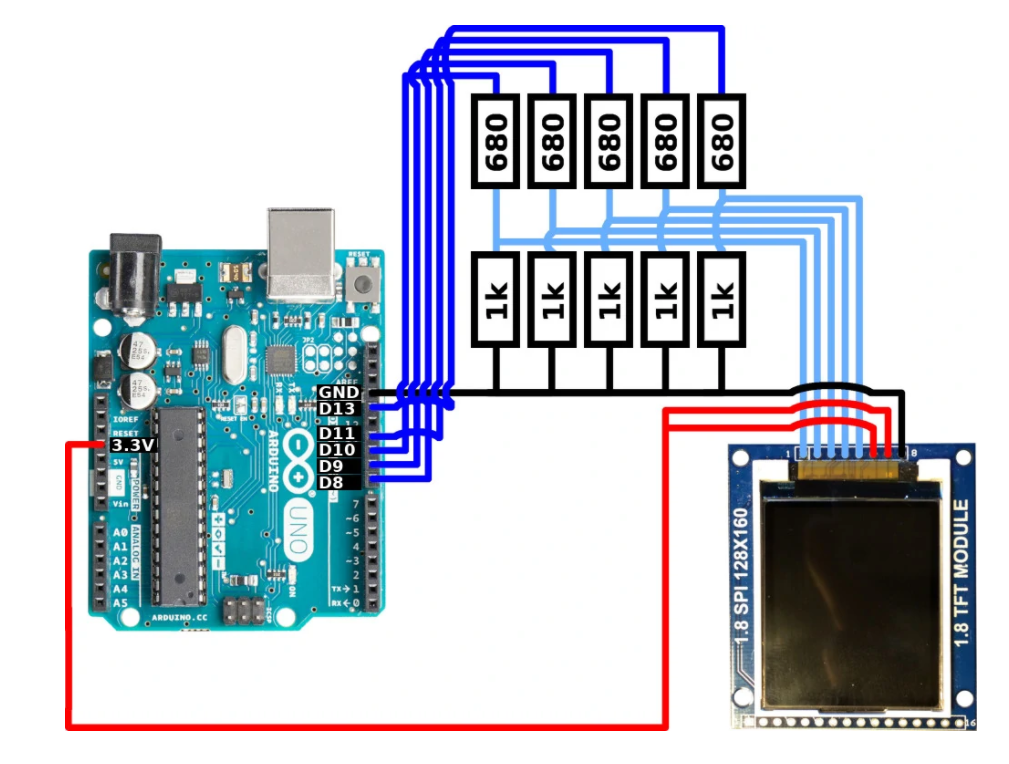
* ابعاد:‌ ۳۵ \*‌ ۳۴ \*‌ ۲۹ میلی متر
* وزن: ۳۰ گرم
* ولتاژ مورد نیاز: ۳.۳ ولت
* نحوه اتصال: 

تصویر۲:نحوه اتصال ماژول دوربینOV7670 به آردوینو UNO

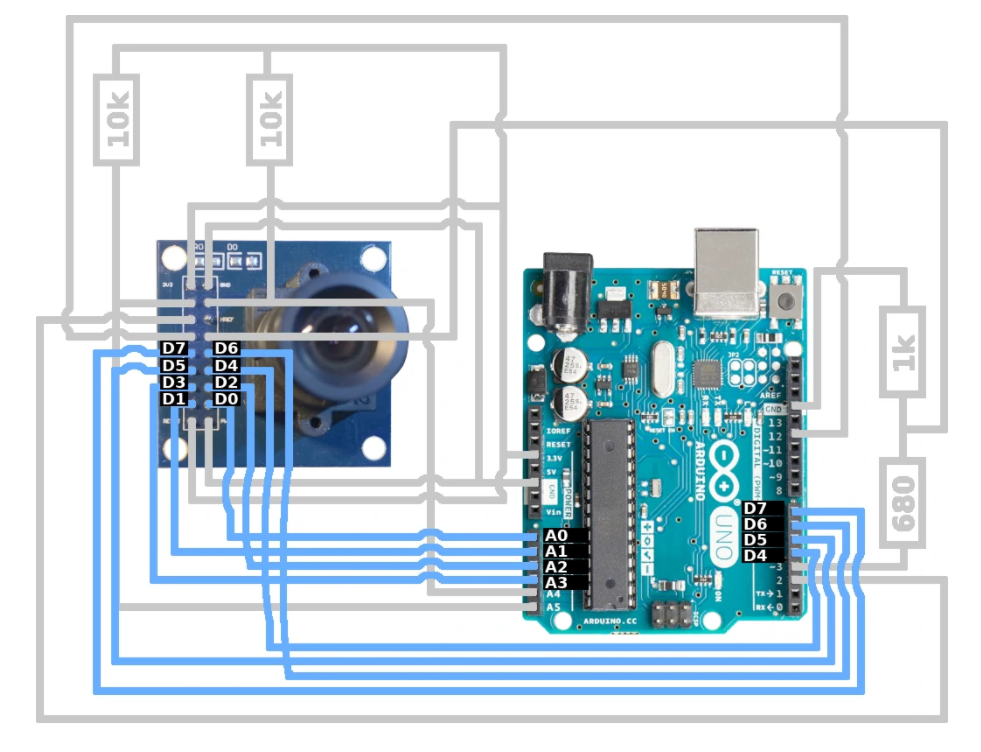
جدول۱: لیست پین‌های OV7670

| **Description** | **Type** | **Pin Name** | **Pin No.** |
| --- | --- | --- | --- |
| 3.3v Power supply | POWER | VCC | 1 |
| Power ground | GROUND | GND | 2 |
| Two-Wire Serial Interface Clock | input | SCL | 3 |
| Two-Wire Serial Interface Data I/O | Bi-directional | Sdata | 4 |
| Active High: Frame Valid; indicates active frame | output | VSYNC | 5 |
| Active High: Line/Data Valid; indicates active pixels | output | HREF | 6 |
| Pixel Clock output from sensor | output | PCLK | 7 |
| Master Clock into Sensor | input | XCLK | 8 |
| Pixel Data Output 9 (MSB) | output | Dout9 | 9 |
| Pixel Data Output 8 | output | Dout8 | 10 |
| Pixel Data Output 7 | output | Dout7 | 11 |
| Pixel Data Output 6 | output | Dout6 | 12 |
| Pixel Data Output 5 | output | Dout5 | 13 |
| Pixel Data Output 4 | output | Dout4 | 14 |
| Pixel Data Output 3 | output | Dout3 | 15 |
| Pixel Data Output 2 (LSB) | output | Dout2 | 16 |

* همانطور که مشاهده می کنید این دوربین ۳ پایه دیجیتال و ۲ پایه آنالوگ را به خود اختصاص می‌دهد. از هیچ درایور دیگری لازم نیست استفاده کنیم.
* **توجه:** با تحقیقاتی متوجه شدیم اگر با اتصال این ماژول به کامپیوتر دارای محدودیت در نرخ انتقال داده هستیم. کامپیوتر ها از حدی بیشتر نرخ انتقال داده را اجازه نمی دهند. بنابراین اگر بخواهیم نتیجه را در کامپیوتر به صورت زنده مشاهده کنیم، از fps خوبی برخوردار نخواهیم بود. برای رفع این مشکل می توانیم فیلم گرفته شده را در lcd وصل شده به همین بورد به صورت زنده نمایش دهیم.



تصویر۳: نحوه اتصال ماژول ال سی دی به ماژول آردوینو UNO



تصویر۴: نحوه اتصال پایه های انتقال داده دوربین به برد آردوینو

**ماژول بازر KY-012**

****

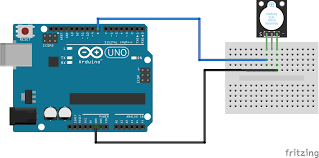
تصویر۶: ماژول بازر KY-012

* ابعاد:‌ 18.5 \* 15 میلی متر
* وزن: 11 گرم
* نحوه اتصال:

Sensor Signal = [Pin 13]

Sensor [N.C] =

Sensor GND = [Pin GND]

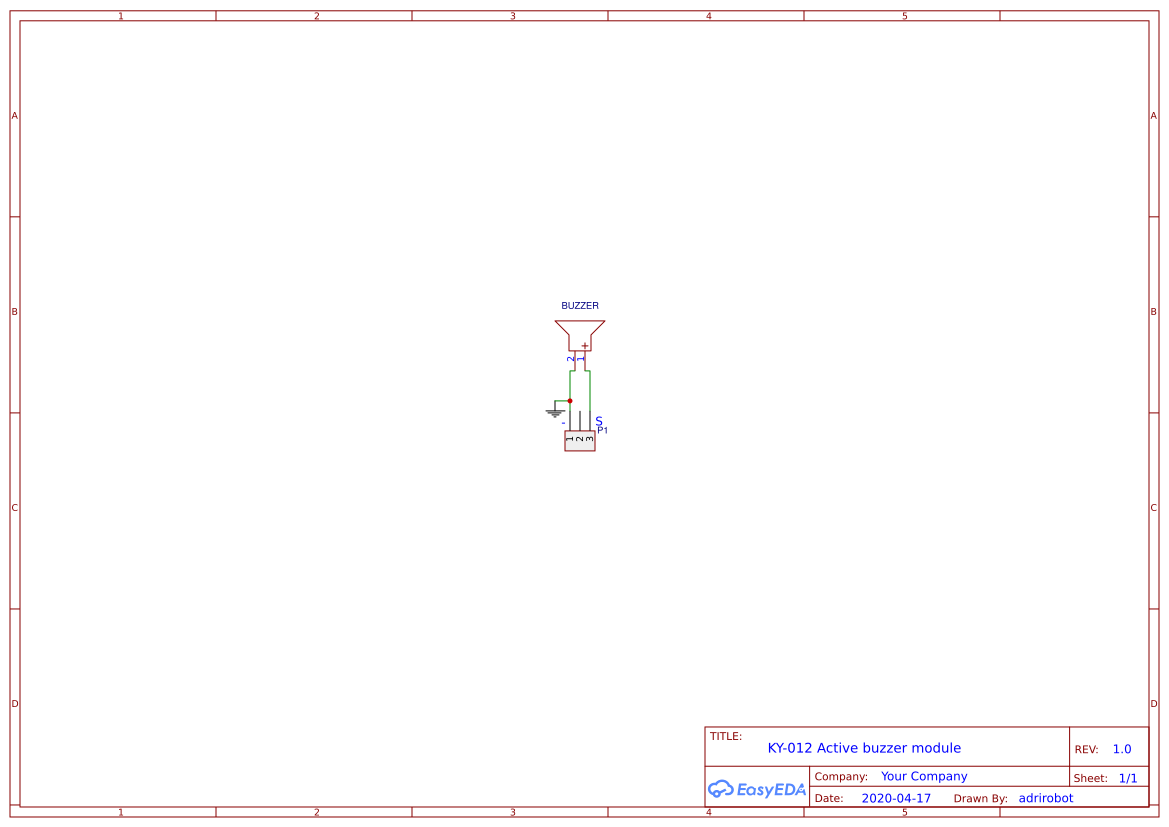


تصویر7: نحوه ی اتصال به برد آردوینو

این ماژول بازر ولتاژ حداقل 3.3 ولت را گرفته و صوتی با فرکانس 2.5KHz را تولید می کند.

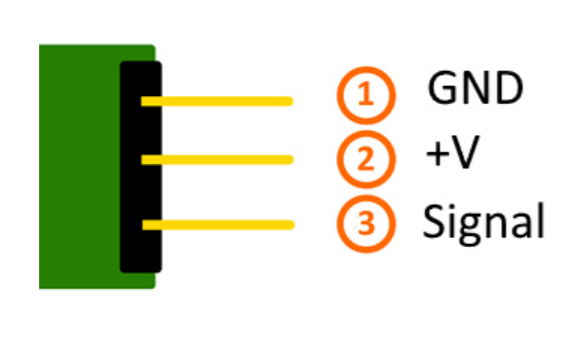
* تکنولوژی استفاده شده در ماژول:

این ماژول از یک بازر تشکیل شده است که زمانی که ولتاژ وارد آن می شود با یک فراکانس یکنواخت صدا تولید می کند.



تصویر8: نمای داخلی ماژول KY-012

* پین های ماژول:



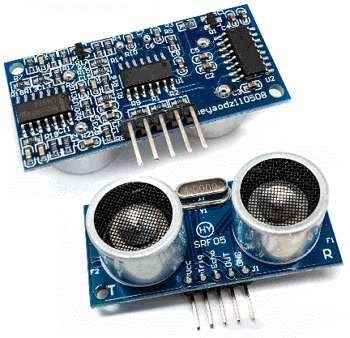
تصویر9: پین های ماژول KY-012

* کد نمونه:

یک نمونه کد برای آزمایش این ماژول در زیر آورده شده است:

| int Buzzer = 13;  void setup () {  pinMode (Buzzer, OUTPUT);  // Output pin initialization for the buzzer  }  void loop () {  digitalWrite (Buzzer, HIGH); // Buzzer will be on  delay (4000); // Waitmode for 4 seconds  digitalWrite (Buzzer, LOW); // Buzzer will be off  delay (2000); // Waitmode for another 2 seconds in which the  buzzer will be off  } |
| --- |
|  |

**ماژول تشخیص فاصله SRF05 - Ultrasonic Ranger**

****

تصویر10: ماژول تشخیص فاصله SRF-05

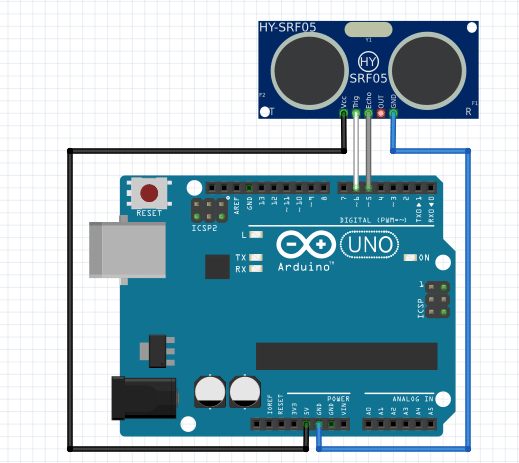
* ابعاد:‌ 43\*20\*17 میلی متر
* وزن: 10 گرم
* نحوه اتصال:

Echo = [Pin 5]

Tring = [Pin 6]

VCC = [Pin 5V]

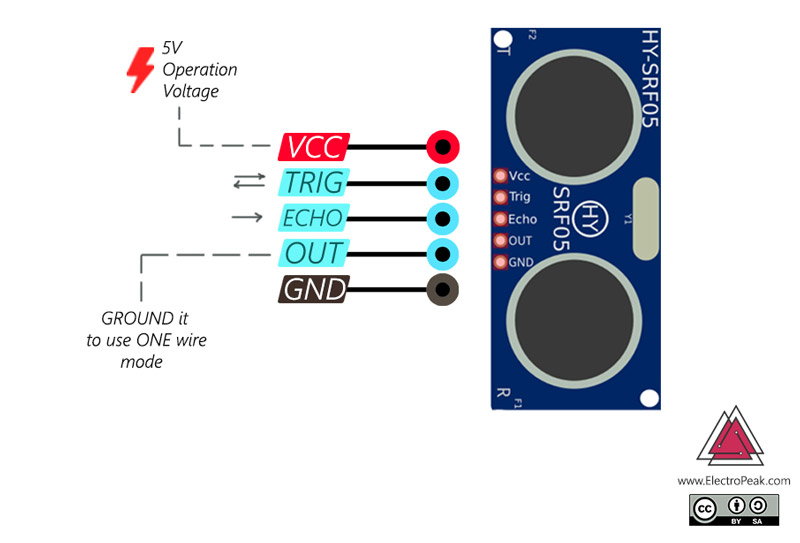
GND = [Pin GND]



تصویر11: نحوه ی اتصال به برد آردوینو

این ماژول تشخیص فاصله توانایی تشخیص جسمی از 2 سانتی متر تا 4.5 متر با دقت 3 میلی متر را دارد. برای استفاده از این ماژول باید یک پالس 10uS را به پین trigger وصل کرده و خروجی حاصل را ضبط کنیم. عرض موجی که از پین echo ضبط میشود با فاصله ی جسم با ماژول تغییر میکند و با استفاده از آن می توانیم فاصله را تشخیص دهیم.

* پین های ماژول:



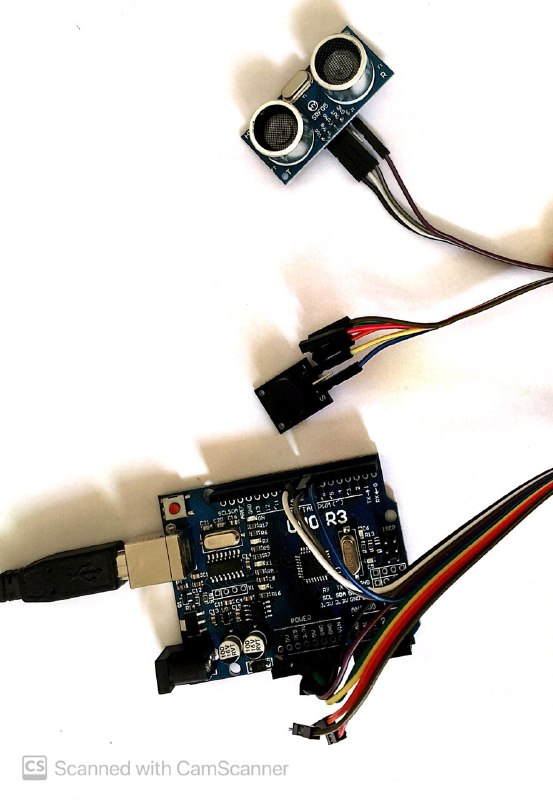
تصویر13: پین های ماژول KY-012

* کد نمونه:

یک نمونه کد برای آزمایش این ماژول در زیر آورده شده است:

| #include <NewPing.h>  #define TRIGGER\_PIN 12 *// Arduino pin tied to trigger pin on the ultrasonic sensor.*  #define ECHO\_PIN 11 *// Arduino pin tied to echo pin on the ultrasonic sensor.*  #define MAX\_DISTANCE 200 *// Maximum distance we want to ping for (in centimeters). Maximum sensor distance is rated at 400-500cm.*  NewPing **sonar**(TRIGGER\_PIN, ECHO\_PIN, MAX\_DISTANCE); *// NewPing setup of pins and maximum distance.*  **void** **setup**() {  Serial.begin(115200); *// Open serial monitor at 115200 baud to see ping results.*  }  **void** **loop**() {  delay(50); *// Wait 50ms between pings (about 20 pings/sec). 29ms*  *should be the shortest delay between pings.*  **unsigned** **int** uS = sonar.ping(); *// Send ping, get ping time in*  *microseconds (uS).*  Serial.print("Ping: ");  Serial.print(sonar.convert\_cm(uS)); *// Convert ping time to distance*  *and print result (0 = outside set distance range, no ping echo)*  Serial.println("cm");  } |
| --- |
|  |

در ادامه این دو ماژول آخر را به هم وصل کردیم تا به ازای فاصله های نزدیک بوق هایی با فاصله های کمتر بزند.

****

تصویر14: اتصال دو ماژول نهایی به اردوئینو

برای وصل این دو ماژول به هم از کد زیر استفاده شده است.

| #define trigPin 8  #define echoPin 7  #include <NewPing.h>  void setup() {  Serial.begin (9600);  pinMode(6, OUTPUT);  }  NewPing sonar(trigPin, echoPin, 200);  void loop() {  delay(50);  unsigned int uS = [sonar.ping](http://sonar.ping/)(); // Send ping, get ping time in  Serial.print("Ping: ");  float distance = sonar.convert\_cm(uS);  Serial.print(distance); // Convert ping time to distance and print result (0 = outside  set distance range, no ping echo)  Serial.println("cm");  delay(distance \* 100);  digitalWrite(6, HIGH);  delay(distance \* 100);  digitalWrite(6, LOW);      } |
| --- |
|  |