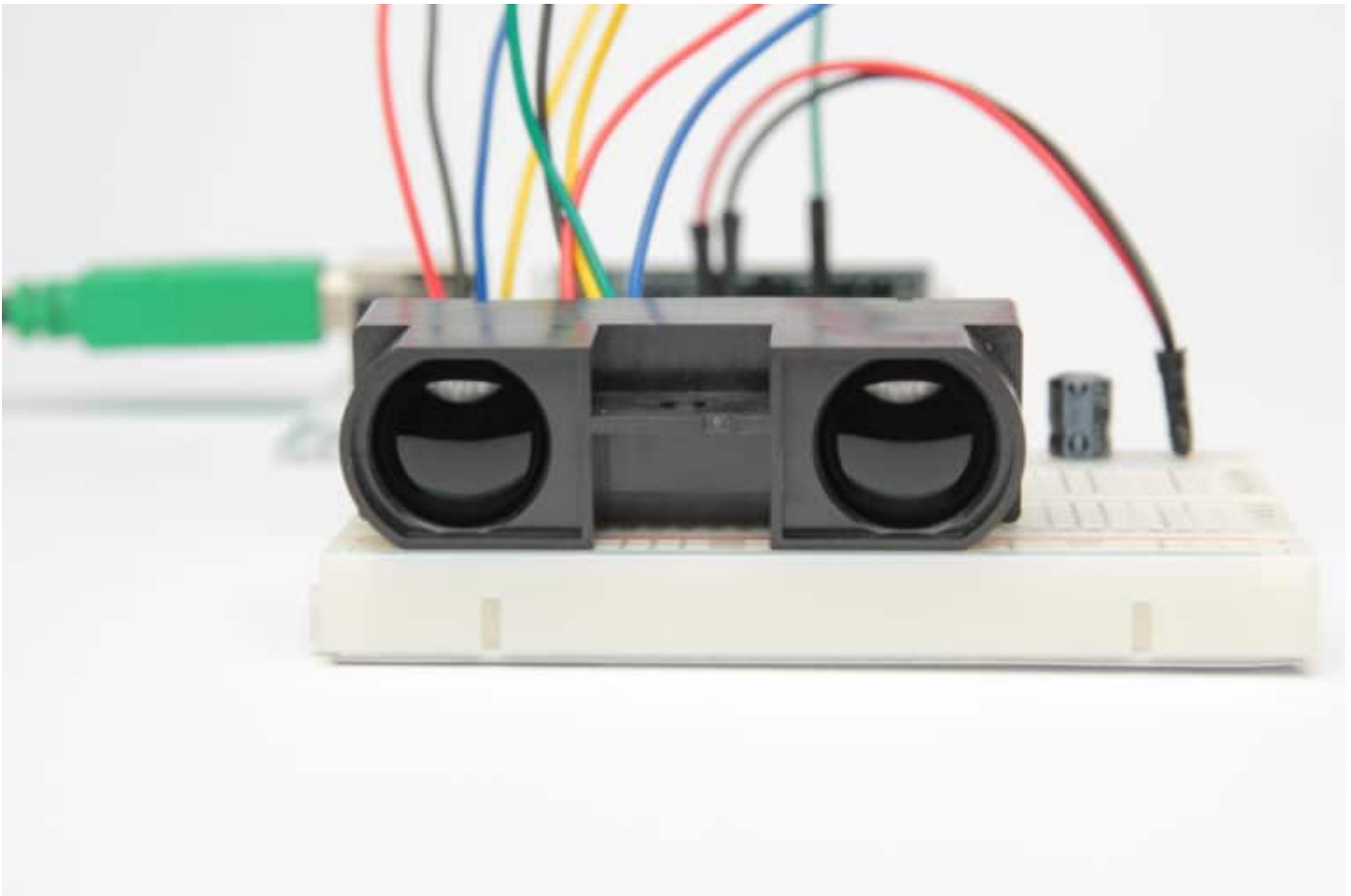


Makerguides.com

How to use a SHARP GP2Y0A710K0F IR Distance Sensor with Arduino

Written by Benne de Bakker (<https://www.makerguides.com/author/benne-de-bakker/>)



The SHARP GP2Y0A710K0F (<https://amzn.to/32P4VJ2>) is a relatively easy to use IR distance sensor with an extra long range of 100 – 500 cm. This extra long range makes it a good alternative for ultrasonic distance sensors.

In this tutorial, you will learn how the sensor works and how to use it with Arduino. I have included a wiring diagram and example code so you can start experimenting with your sensor.

If you are looking for a more affordable or waterproof distance sensor, take a look at the HC-SR04 or JSN-SR04T. In the articles below I explain how these distance/proximity sensors work and how you can use them with Arduino.

Other distance/proximity sensors:

- How to use an HC-SR04 Ultrasonic Distance Sensor with Arduino (<https://www.makerguides.com/hc-sr04-arduino-tutorial/>)
- Waterproof JSN-SR04T Ultrasonic Distance Sensor with Arduino Tutorial (<https://www.makerguides.com/jsn-sr04t-arduino-tutorial/>)
- How to use a SHARP GP2Y0A21YK0F IR Distance Sensor with Arduino (<https://www.makerguides.com/sharp-gp2y0a21yk0f-ir-distance-sensor-arduino-tutorial/>)

Supplies

Hardware components

	SHARP GP2Y0A710K0F IR distance sensor (https://amzn.to/2ZGMZif)	×	Amazon (https://amzn.to/2ZGMZif)
(https://amzn.to/374aJjX)	Arduino Uno Rev3 (https://amzn.to/374aJjX)	×	Amazon (https://amzn.to/374aJjX)
	Breadboard (https://amzn.to/2MQHicc)	×	Amazon (https://amzn.to/2MQHicc)

Jumper wires (https://amzn.to/2EG9wDc)	~ 10	Amazon (https://amzn.to/2EG9wDc)
Capacitor (https://amzn.to/2I8TZAD) (10 μ F or more)	× 1	Amazon (https://amzn.to/2I8TZAD)
USB cable type A/B (https://amzn.to/34SBuXf)	× 1	Amazon (https://amzn.to/34SBuXf)

Software

Arduino IDE (<https://www.arduino.cc/en/Main/Software>)

Makerguides.com is a participant in the Amazon Services LLC Associates Program, an affiliate advertising program designed to provide a means for sites to earn advertising fees by advertising and linking to products on Amazon.com.

How does an IR distance sensor work?

An IR distance sensor uses a beam of infrared light to reflect off an object to measure its distance. The distance is calculated using triangulation of the beam of light. The sensor consists of an IR LED and a light detector or PSD (Position Sensing Device).

When the beam of light gets reflected by an object, the reflected beam will reach the light detector and an 'optical spot' will form on the PSD.

When the position of the object changes, the angle of the reflected beam and the position of the spot on the PSD changes as well. See point A and point B in the image below.

Note the change in the angle of the reflected beam and the position of the 'optical spot' on the PSD.

The sensor has a built-in signal processing circuit. This circuit processes the position of the optical spot on the PSD to determine the position (distance) of the reflective object. It outputs an analog signal which depends on the position of the object in front of the sensor.

How to read an IR distance sensor?

IR distance sensors output an analog signal, which changes depending on the distance between the sensor and an object. From the datasheet, you can see that the output voltage of the SHARP GP2Y0A710K0F ranges from 2.5 V when an object is 100 cm

away to 1.4 V when an object is 500 cm away. The graph also shows why the usable detection range starts at 100 cm. Notice that the output voltage of an object that is 30 cm away is the same as the output voltage for an object that is 160 cm away. The usable detection range therefore start after the peak at roughly 100 cm or 2.5 V.

Distance measuring characteristics (output)

The graph also shows the drawback of these sensors, the response is non-linear. In other words, a big change in the output voltage does not always correspond to a big change in range. In order to determine the distance between the sensor and an object, you need to find a function that converts the output voltage into a range value.

If you plot the output voltage against the inverse distance, you get a mostly linear relationship. For distances > 100 cm you get the following curve.

Output voltage vs Inverse number of distance $1/L$ (1/cm)

Based on the data from the SHARP datasheet, you can calculate the linear function:

$$y = 137500x + 1125$$

With y equal to the output voltage in mV and x equal to $1/\text{distance}$ in cm.

This results in the following formula for the distance between the sensor and an object:

$$\text{Distance (cm)} = 1/((\text{Output_voltage_mV} - 1125)/137500)$$

This is the function that is used in the **SharpIR library**, which we will be using later. Note that this function is based on data from the SHARP datasheet only. The output characteristics of the sensor will vary slightly from sensor to sensor so you might get inaccurate readings.

GP2Y0A710K0F Specifications

Operating voltage	4.5 to 5.5 V
Operating current	30 mA
Measuring range	100 to 550 cm
Output type	Analog
Dimensions	58 x 17.6 x 22.5 mm
Mounting hole	4.2 mm
Cost	Check price (https://amzn.to/2Lg2VlO)

For more information you can check out the datasheet here.

GP2Y0A710K0F Datasheet 

Wiring – Connecting GP2Y0A710K0F IR sensor to Arduino

The wiring diagram below shows you how to connect the GP2Y0A710K0F IR distance sensor to an Arduino. **Please note: the wire colors are not intuitive!**

GP2Y0A710K0F IR distance sensor with Arduino wiring diagram

These types of distance sensors tend to be a bit noisy, so it is recommended to add a capacitor between Vcc and GND. The datasheet suggests a capacitor of 10 μ F or more. Connect the positive lead of the capacitor to the Vcc wire connection and the negative lead to the GND wire connection (see picture). Capacitors are often marked with a stripe which indicates the negative lead. The positive lead is often longer than the negative lead.

GP2Y0A710K0F Connections

GP2Y0A710K0F	Arduino
1 (Red)	GND
2 (Black)	5V
3 (Yellow)	5V
4 (Green)	A0
5 (Blue)	GND

If your sensor comes with different colored wires, be sure to check the pinout below. The Vo pin is connected to the analog in of the Arduino (A0).

GP2Y0A710K0F IR distance sensor pinout

The connector of the sensor is not marked, the order of the pins corresponds to the sensor with the logo facing up.

Now that you have wired up the sensor it is time to look at some example code.

Installing the SharpIR Arduino library

The **SharpIR** library written by Guillaume Rico and Thibaut Mauon makes working with SHARP IR sensors a lot easier. It includes the formulas that are needed to convert the measured output voltage to a distance in centimeters. Currently, the library supports the following sensors: GP2Y0A02YK0F, GP2Y0A21YK0F, GP2Y0A710K0F, and GP2YA41SK0F. The latest version of the library can be downloaded here on GitHub (<https://github.com/guillaume-rico/SharpIR>) or click the button below.

SharpIR-master.zip 

You can install the library by going to **Sketch > Include Library > Add .ZIP Library** in the Arduino IDE.

Installing a .ZIP Arduino library.

The author of the library noticed that the readings from the sensor can fluctuate quite a bit. The library solves this problem by taking multiple readings in a row, discarding outliers, and taking the mean to get a more stable distance reading. Currently, the library takes the mean of 25 readings, which takes roughly 53 ms.

Example code for SHARP GP2Y0A710K0F IR distance sensor with Arduino

The example code below can be used with the GP2Y0A710K0F sensor and displays the measured distance in centimeters in the serial monitor.

You can copy the code by clicking the button in the top right corner of the code field.

```
1.  /*SHARP GP2Y0A710K0F IR distance sensor with Arduino and SharpIR library example code.
    More info: https://www.makerguides.com */
2.
3.  // Include the library:
4.  #include <SharpIR.h>
5.
6.  // Define model and input pin:
7.  #define IRPin A0
8.  #define model 100500
9.
10. // Create variable to store the distance:
11. int distance_cm;
12.
13. /* Model :
14.    GP2Y0A02YK0F --> 20150
15.    GP2Y0A21YK0F --> 1080
16.    GP2Y0A710K0F --> 100500
17.    GP2YA41SK0F --> 430
18. */
19.
20. // Create a new instance of the SharpIR class:
21. SharpIR mySensor = SharpIR(IRPin, model);
22.
23. void setup() {
24.     // Begin serial communication at a baud rate of 9600:
25.     Serial.begin(9600);
26. }
27.
28. void loop() {
29.     // Get a distance measurement and store it as distance_cm:
30.     distance_cm = mySensor.distance();
```

```
31.
32.     // Print the measured distance to the serial monitor:
33.     Serial.print("Mean distance: ");
34.     Serial.print(distance_cm);
35.     Serial.println(" cm");
36.
37.     delay(1000);
38. }
```

Note that we have called the sensor 'mySensor' in this example. If you want to use multiple IR distance sensors, you can create another sensor object with a different name: `SharpIR mySensor2 = SharpIR(IRPin2, model);` Note that in that case you also use a different input pin for the second sensor.

When the output voltage is under 1.4 V or higher than 3.3 V, you will see "Mean distance: 0 cm" in the serial monitor.

Conclusion

In this article, I have shown you how the SHARP GP2Y0A710K0F IR distance sensor works and how you can use it with Arduino. I hope you found it useful and informative. If you did, please share it with a friend that also likes electronics!

I would love to know what projects you plan on building (or have already built) with this IR distance sensor. If you have any questions, suggestions, or if you think that things are missing in this tutorial, **please leave a comment down below.**

Note that comments are held for moderation to prevent spam.

Beginner



What to read next?

LM35 analog temperature sensor with Arduino tutorial
(<https://www.makerguides.com/lm35-arduino-tutorial/>)

TMP36 analog temperature sensor with Arduino tutorial
(<https://www.makerguides.com/tmp36-arduino-tutorial/>)

Arduino Nano Board Guide (Pinout, Specifications, Comparison)
(<https://www.makerguides.com/arduino-nano/>)

The complete guide for DS18B20 digital temperature sensors with Arduino
(<https://www.makerguides.com/ds18b20-arduino-tutorial/>)

How to use an IR receiver and remote with Arduino
(<https://www.makerguides.com/ir-receiver-remote-arduino-tutorial/>)

Comments

Oliver says

May 29, 2019 at 7:47 am (<https://www.makerguides.com/sharp-gp2y0a710k0f-ir-distance-sensor-arduino-tutorial/#comment-73>)

Hello

I would like to use this sensor. But for my project it would help if I could use mm output instead of cm – is this possible?

Reply

Benne de Bakker says

May 29, 2019 at 8:19 am (<https://www.makerguides.com/sharp-gp2y0a710k0f-ir-distance-sensor-arduino-tutorial/#comment-74>)

Hi,

There is no default mm function in this library. You would need to change the SharpIR.cpp file of the library. This is where the raw data is transformed to cm. You should be able to find the file in your libraries folder or via the GitHub link. But I have to say that I don't think the sensor is accurate enough to measure in mm.

Reply

Trackbacks

MaxBotix MB7389 Weather Resistant Ultrasonic Sensor Tutorial

(<https://www.makerguides.com/maxbotix-mb7389-arduino-tutorial/>) says:

September 26, 2019 at 5:55 pm (<https://www.makerguides.com/sharp-gp2y0a710k0f-ir-distance-sensor-arduino-tutorial/#comment-654>)

[...] How to use a SHARP GP2Y0A710K0F IR Distance Sensor with Arduino [...]

How to use SHARP IR Distance Sensor with Arduino (GP2Y0A21YK0F)

(<https://www.makerguides.com/sharp-gp2y0a21yk0f-ir-distance-sensor-arduino-tutorial/>) says:

June 29, 2019 at 6:47 pm (<https://www.makerguides.com/sharp-gp2y0a710k0f-ir-distance-sensor-arduino-tutorial/#comment-172>)

[...] How to use a SHARP GP2Y0A710K0F IR Distance Sensor with Arduino [...]

How to use HC-SR04 Ultrasonic Sensor with Arduino (5 examples)

(<https://www.makerguides.com/hc-sr04-arduino-tutorial/>) says:

June 17, 2019 at 11:43 pm (<https://www.makerguides.com/sharp-gp2y0a710k0f-ir-distance-sensor-arduino-tutorial/#comment-115>)

[...] How to use a SHARP GP2Y0A710K0F IR Distance Sensor with Arduino [...]

(<https://www.makerguides.com/l298n-stepper-motor-arduino-tutorial/>)


Control a stepper motor with L298N motor driver and Arduino
(<https://www.makerguides.com/l298n-stepper-motor-arduino-tutorial/>)

(<https://www.makerguides.com/maxbotix-mb7389-arduino-tutorial/>)

MaxBotix MB7389 weather-resistant distance sensor tutorial
(<https://www.makerguides.com/maxbotix-mb7389-arduino-tutorial/>)

(<https://www.makerguides.com/tb6600-stepper-motor-driver-arduino-tutorial/>)

TB6600 Stepper Motor Driver with Arduino Tutorial
(<https://www.makerguides.com/tb6600-stepper-motor-driver-arduino-tutorial/>)

 Ezoic (<https://www.ezoic.com/what-is-ezoic/>)

[report this ad](#)

© 2021 Makerguides.com - All Rights Reserved