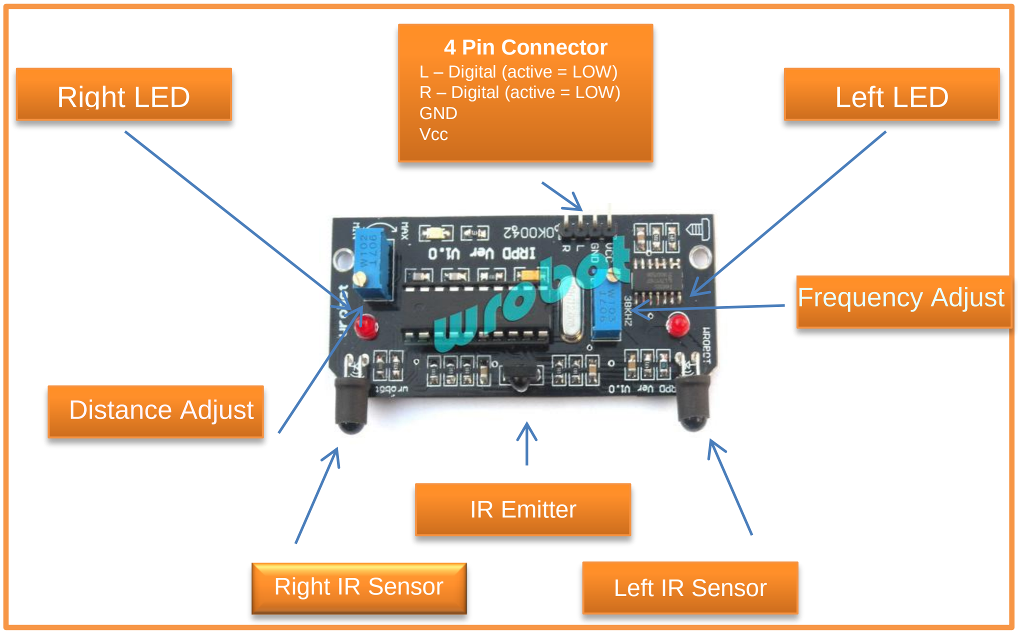
ARDUINO INFRARED OBSTACLE-AVOIDING SENSOR

I am investigating a number of different sensor alternatives to determine cost/benefit as measured against my needs. The device I evaluated in this case is as the title above reads, I purchased it from a distributor in China: www.elechouse.com.



The frequency is well calibrated. Change to it is not recommended.

OVERVIEW

The Device is designed to send out infrared signals and detect the response to determine an object's presence. The specifications from the on-line page are repeated here with some editorial adjustments.

- Adjustable sensitivity: 30mm - 600mm (affected by reflectivity of an object) – my experience using my hand and a piece of paper was about 7 inches (177.8mm). See below on sensitivity adjustment.
- Can detect obstacles in three directions
- Embedded 38Khz generator delivers strong anti-interference capacity

The Device can be installed together with the tracing module to form a tracking and barrier dodging module.

OPERATION

The device drives the L and R pins HIGH until an object is detected on the right or left side. When something is detected it simultaneously drives L and/or R LOW and illuminates a red LED on either/or the left and right sides.

The sensitivity is adjustable. It seems difficult to adjust as I can only derive a maximum detection range of about 7 inches (177.8 mm).

CONNECTING TO ARDUINO

I used the Arduino to interface to the Device; more specifically, the Arduino Mega 2560. I chose to map the L pin to PWM/Digital pin 6, and the R pin to PWM/Digital pin 5. I also sourced Vcc and GND from the Arduino. The Mega was attached to the PC via USB cable.

> *[plug] I have had the pleasure of working with Wilson Shen at Elechouse. He has been extremely responsive, professional and helpful. Good Guy! [\plug]*

That's about it for the hardware. Now on to the tough stuff! Software.

SOFTWARE

I hope this code speaks enough for itself.

```
//  Module to control the WRobot Infrared Barrier Sensor - IRPD V1.0
//
//  Created 3/13/2011 by Agile Robotics.
//  Please send any mods or questions to support@agilerobotics.com
//
//  This code is placed in the public domain by Agile Robotics, LLC
//

int sensorPinLO = 6;    // Right_output, Left_output.  The pin on
                        // the IRPD v1.0 board is labeled 'L' for the
                        // left output and 'R' for the right output
int sensorPinRO = 5;    // These are the pins that we will read
                        // to determine if something is in Range
                        // If something is within Range, it will
                        // LOW on the digital pin, otherwise it
                        // latches high

int ledPin = 13;      // select the pin for the LED - debug purposes
```

```
                      // only.  Helps to see that the Arduino is
                      // seeing the same thing as the sensor is as
                      // shown by its LED.
int sensorValue = 0;  // variable to store the value coming from the sensor

void setup() {
  // Debug Only - Initialize Arduino Serial which will be used to communicate
  // status to the developer.
  Serial.begin(9600);

  // Set the pins correctly, we will be reading from both sides
  pinMode(sensorPinLO, INPUT);
  pinMode(sensorPinRO, INPUT);

  // Debug only - indicate the LED pin is an output and print a
  // message that we are done.
  pinMode(ledPin, OUTPUT);
  Serial.println("finished setup");
}


void loop() {
  int charIn;



  // read the value from the sensor to see if something is on the
  // Right side of us.
  if (digitalRead(sensorPinRO) == LOW)  {

    // Debug Only - this is the place where you would do what
    // you need to do when the sensor detects an object.
    Serial.println("Sensing Right");
    digitalWrite(ledPin, HIGH);
  }

  // read the value from the sensor to see if something is on the
  // Right side of us.
  if (digitalRead(sensorPinLO) == LOW) {

    // Debug Only - this is the place where you would do what
    // you need to do when the sensor detects an object.
    Serial.println("Sensing Left");
    digitalWrite(ledPin, HIGH);
  }

  //  Wait for milliseconds to let the demo operate at a more
  //  human speed.
  delay(1000);

  //  Turn the LED off to show activity
  digitalWrite(ledPin, LOW);

}
```