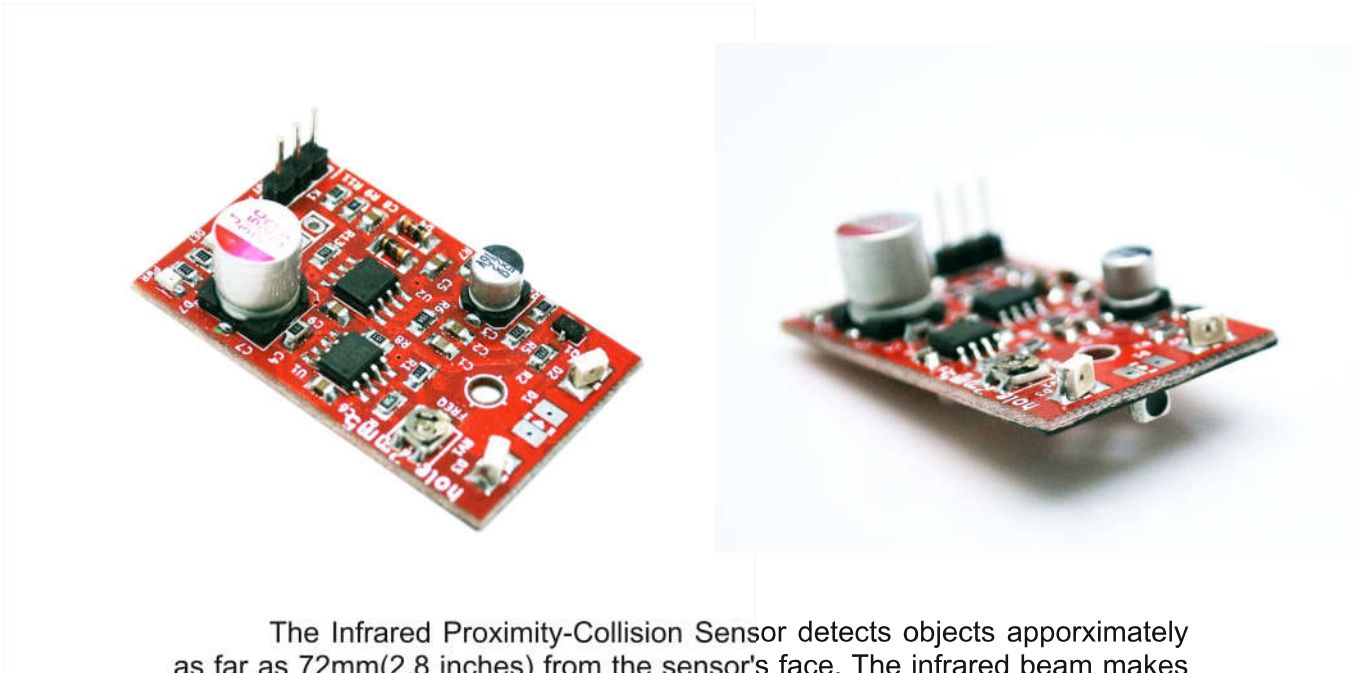# Compact Proximity Collision Sensor Module

Technical Manual Rev 1r0

The Infrared Proximity-Collision Sensor detects objects apporximately as far as 72mm(2.8 inches) from the sensor's face. The infrared beam makes it relativley insensitive to ambient light and color (of the target object). Applications include non contact object detection and collision sensor for mobile robots.

FEATURES:

- 72mm(2.8 inches) typical detection range
- 3IRDA sensors (2 for transmition and the other for receiving).
- 1 TTL active HIGH logic out put (logic goes HIGH when an object is detected).

GENERAL SPECIFICATION:

Supply input: +5V DC
Interface: Arduino Compatible
PCB Dimension: 24 mm x 41 mm

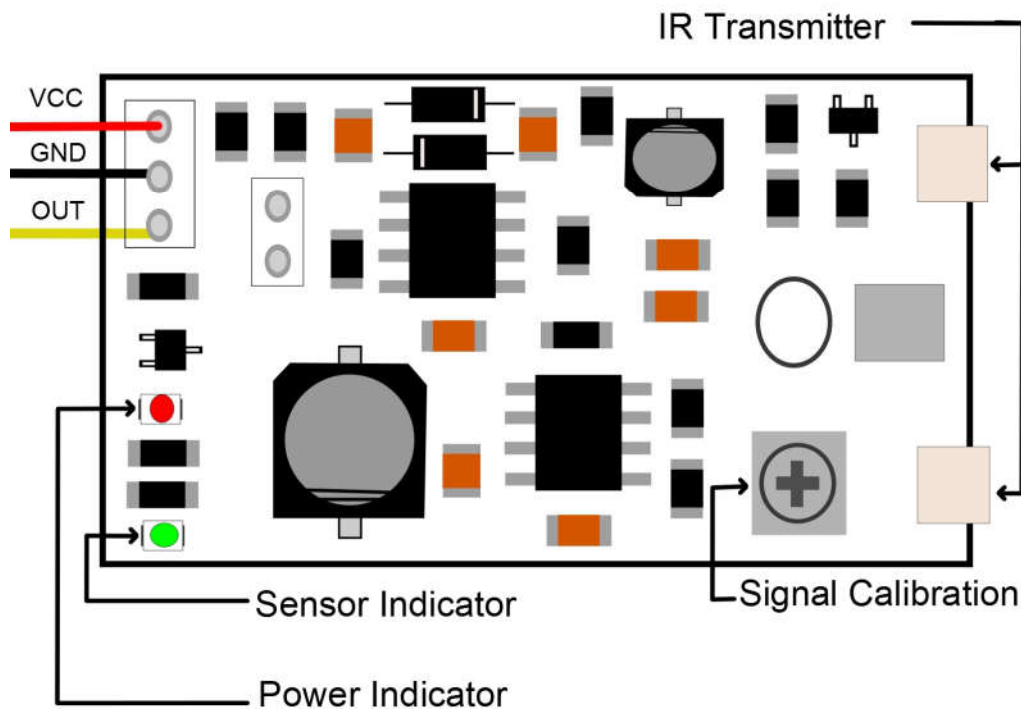e-Gizmo Compact Proximity Sensor

Figure 1. *Compact Proximity Collision sensor &*
its major Components.

        To test the proximity collision sensor, first attach +5V power supply to the Red wire and negative to the Black wire the Yellow to the 'Logic Output', logic goes HIGH when an object is detected and can be connected to microcontrollers or through PC printer port. Now the power indicator will light (Green LED), then place your hand or any object in the front of the infrared transmitter or receiver (2.8 inches maximum) w/o calibration, if the sensor indicator (Red LED) doesn't light, we will need the 'Signal Calibration' rotate the signal calibration
partially either ºCW or ºCCW, then place your hand or any object infront fo the sensor, do this until the sensor indicator has been lit.
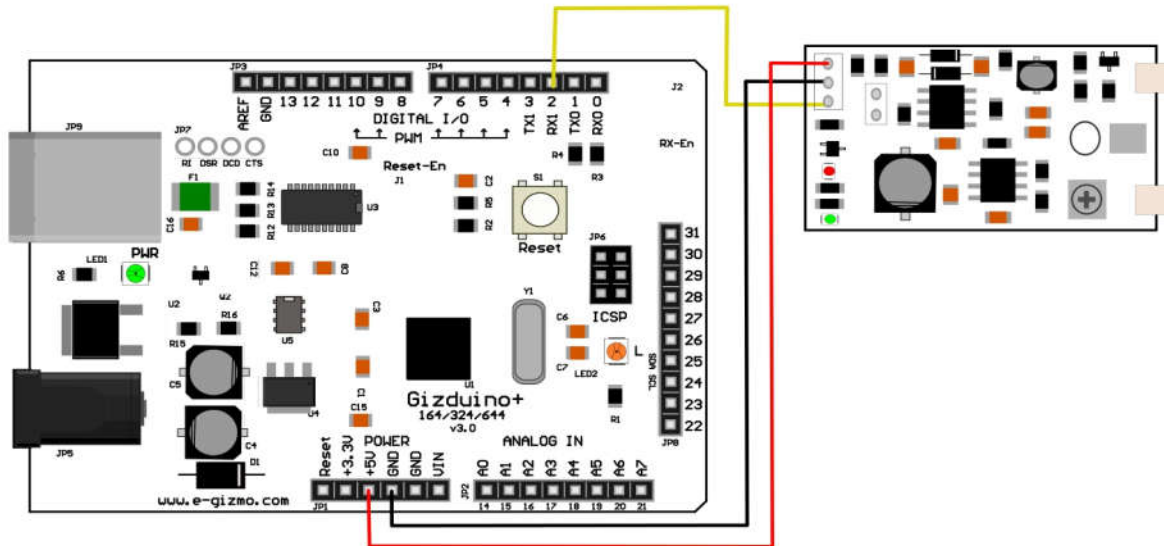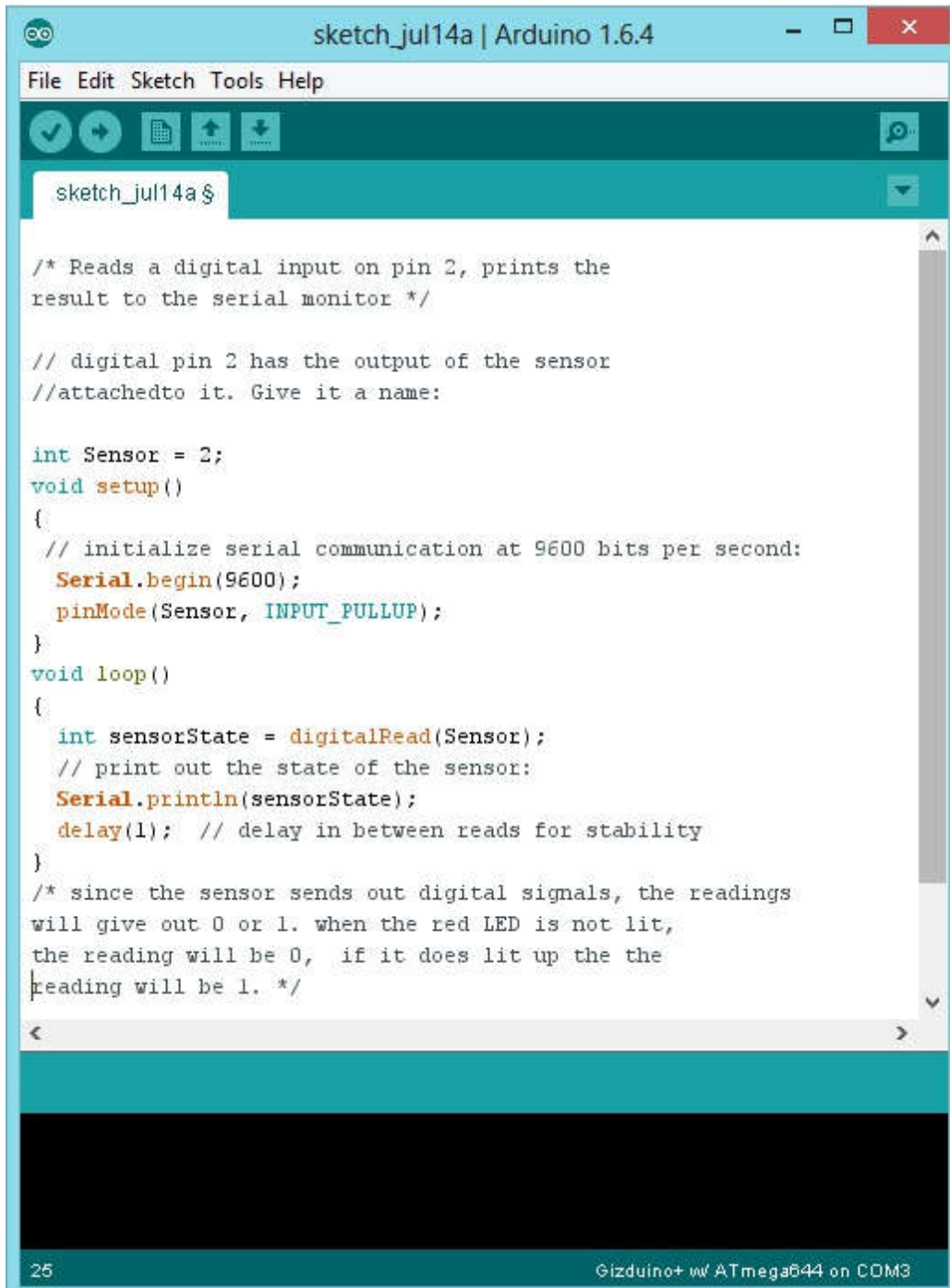
Figure 2. *Compact Proximity Collision Sensor Connected to a gizdDuino + w/ ATmega644P microcontroller.*

To connect the proximity collision sensor to the microcontroller: connect the red wire to the +5V, connect the black wire to the GND, and connect the yellow wire to pin number 2. The connect the USB cable to PC with gizDuino + ATmega644P. Select the correct board Tools>Boards>Gizduino+ ATmega644P and Select the COM PORT number. Upload the Sample code and Click the Serial Monitor to see the result.

```
/* Reads a digital input on pin 2, prints the
result to the serial monitor */

// digital pin 2 has the output of the sensor
//attachedto it. Give it a name:

int Sensor = 2;
void setup()
{
 // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
  pinMode(Sensor, INPUT_PULLUP);
}
void loop()
{
  int sensorState = digitalRead(Sensor);
  // print out the state of the sensor:
  Serial.println(sensorState);
  delay(1);  // delay in between reads for stability
}
/* since the sensor sends out digital signals, the readings
will give out 0 or 1. when the red LED is not lit,
the reading will be 0,  if it does lit up the the
reading will be 1. */
```

25   Gizduino+ w/ ATmega644 on COM3

          e-Gizmo Compact Proximity Sensor