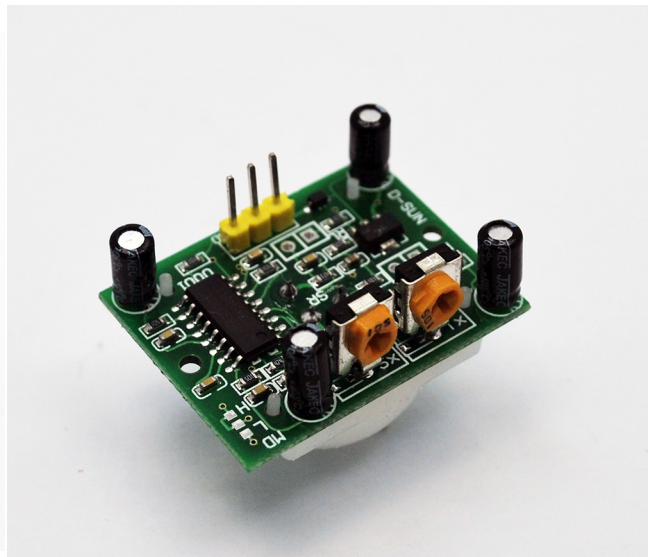
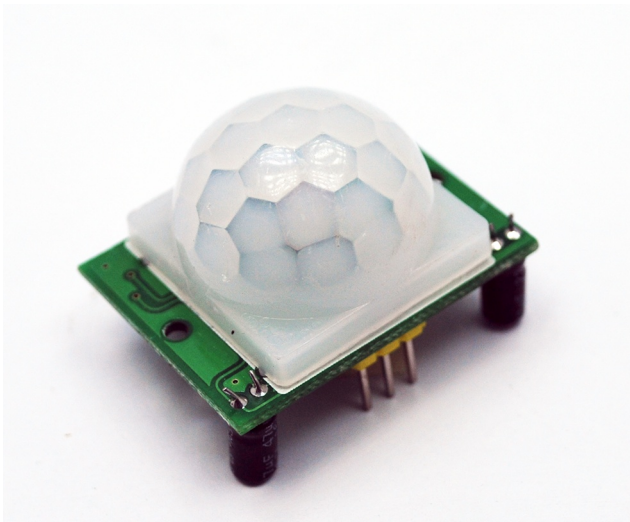


PIR Motion Detector

Technical Manual Rev 1r0

HC-SR501

PIR Motion Detector (HC-SR501) is an automatic control module, high sensitivity, high reliability, ultra-low-voltage operating mode, widely used in various auto-sensing electrical equipment.



FEATURES:

- Gizduino and Arduino Compatible
- Arduino IDE software
- Automatic Induction*
- Photosensitive control*
- Temperature compensation*
- Triggered in two ways*
- With induction blocking time*
- Automatically sensing light for Floor, basement, porch, warehouse, Garage, ventilator, alarm, etc.

* See page 3 (for more information)

GENERAL SPECIFICATIONS:

- Input Supply: 5V ~ 20VDC
- Power Consumptions: 65mA
- TTL Output: 3.3V, 0V
- Delay time: Adjustable (.3->5mins)
- Trigger: L- disable repeat
H - enable repeat
- Sensing range: < 120degreesm within 7 m
- Lock time: 0.2 sec
- PCB Dimensions: 32mmx24mm
- Distance bet. screw: 28mm, M2
- Lens dimension: 23mm diameter

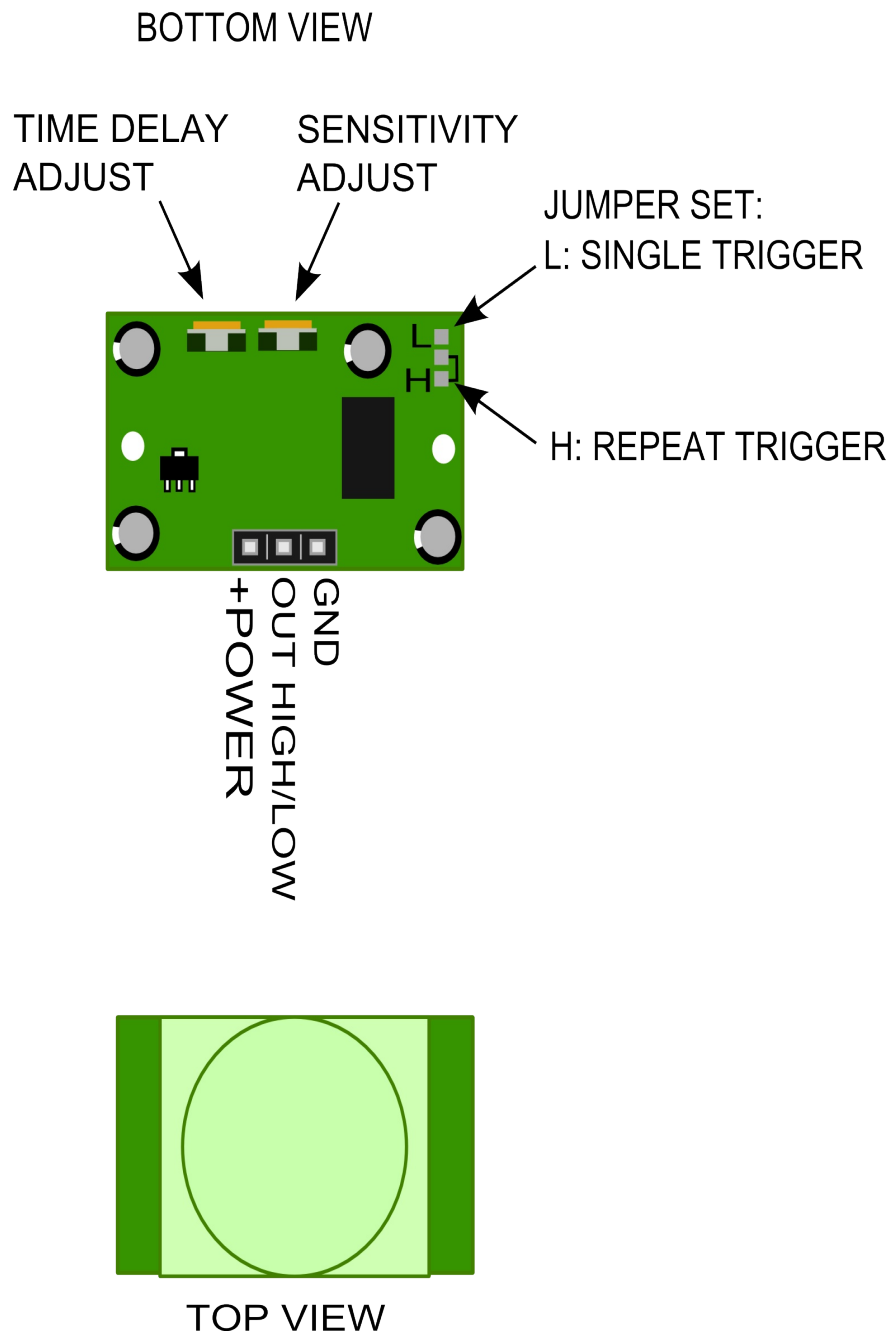


Figure 1. Major parts presentation of PIR Motion Sensor

1. Automatic Induction:

To enter the sensing range of the output is high, the person leaves the sensing range of the automatic delay off high output low.

2. Photosensitive control (optional, not factory-set)

Can be set photosensitive control, day or light intensity without induction.

3. Temperature compensation (optional, factory reset)

In the summer when the ambient temperature rises to 30C to 32C, the direction distance is slightly shorter, temperature compensation can be used for performance compensation.

4. Triggered in two ways: (jumper selectable)

-non-repeatable trigger

-repeatable trigger

5. With induction blocking time(the default setting: 2.5s block time)

6. Wide operating voltage range: default voltage DC4.5V-20v.

7. Micropower consumption:static current <50 microamps.

8. Output high signal

(FOR MORE INFORMATION) Read the HC-SR501 Datasheet.

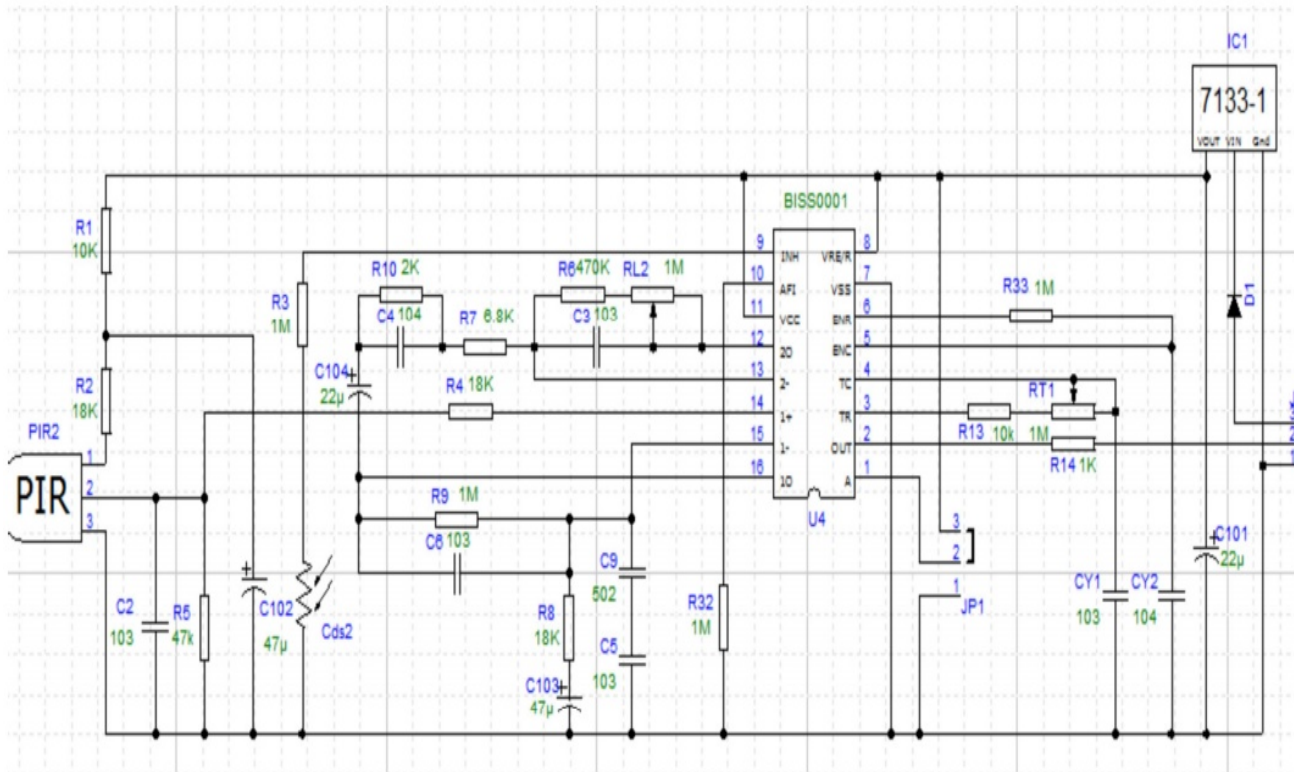
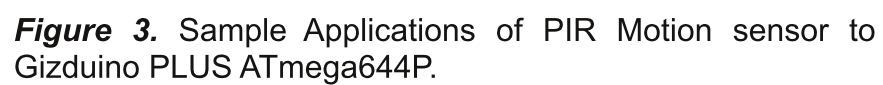


Figure 2. Schematic Diagram of PIR Motion Sensor



```
/*  
PIR Motion sensor HC-SR501  
Sample code  
  
Reads a digital input on pin 7,  
the result is in pin13 LED indicator and  
prints the result to the serial monitor.  
  
e-Gizmo Mechatronix Central  
*/  
//digital pin 7 & 13 assignment  
int input = 7;  
int led = 13;  
  
//setup the routine once  
void setup(){  
  //set serial communication at 9600 b/s  
  Serial.begin(9600);  
  //make the pins as an input/output  
  pinMode(led,OUTPUT);  
  pinMode(input,INPUT);  
}  
//loop routine runs over and over forever  
void loop(){  
  //See the result if LED 13 is HIGH/LOW  
  indicator.  
  digitalWrite(led,digitalRead(input));  
}
```