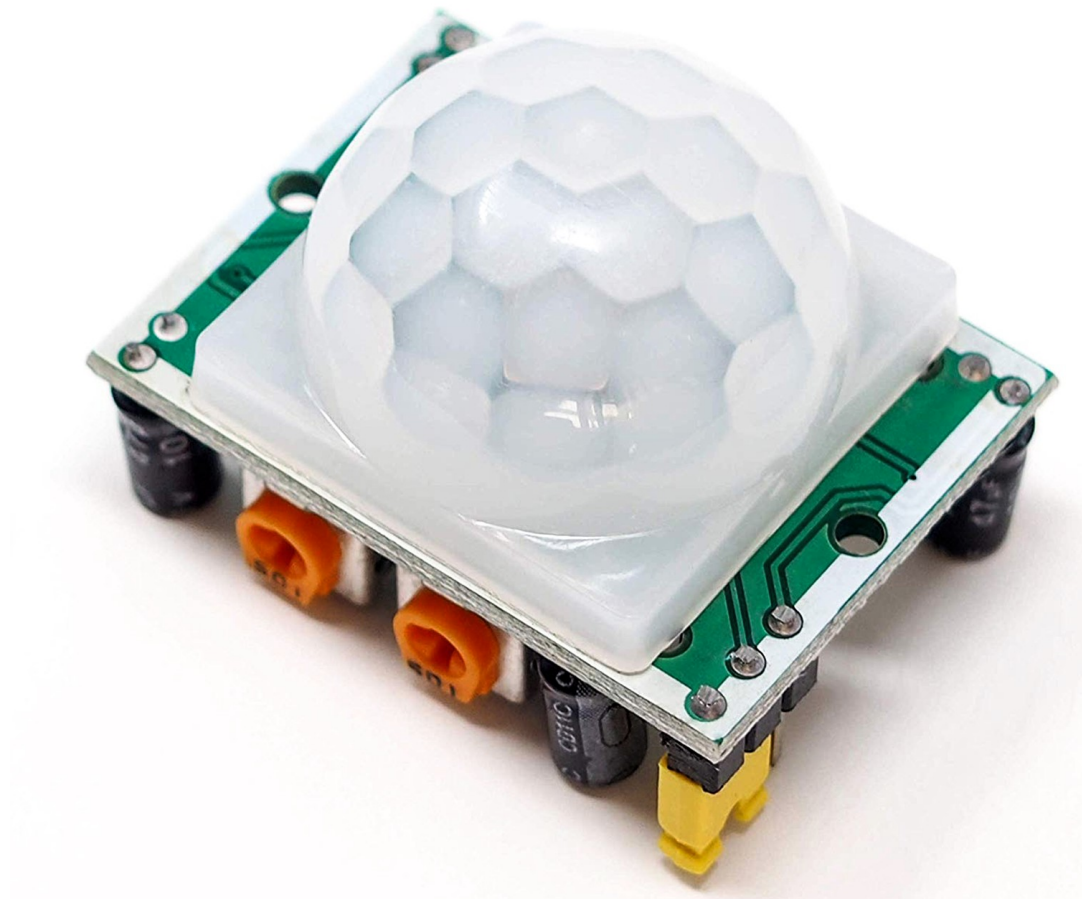


# AZ-Delivery

## Welcome!

Thank you very much for purchasing our AZ-Delivery PIR motion sensor module. On the following pages, we will introduce you to how to use and setup this handy device.

**Have fun!**





The functionality of our PIR motion detection module on circuit board with switching contact is easily explained. It is a pyroelectric sensor which can detect infrared radiation. Similar sensors are used in contactless infrared thermometers, for example. A small plastic sphere is mounted above the module, which scatters the infrared radiation absorbed by the sensor. Thus the area in which movement is detected is enlarged and steered. On the circuit board there are two potentiometers used for control, as well as a jumper bridge. The two potentiometers are for setting the sensitivity and the switching time of the timer integrated in the module, the jumper bridge is used for configuring the desired output format (single trigger, repeat trigger).

## **The most important information in a nutshell**

- » Dimensions: 33mm x 25mm x 25mm
- » Connection:
  - 3.3 - 5 V - Vcc
  - GND - Ground
  - S - DigitalOut
- » Temperature range: -65 ÷ 150 °C
- » Programming via Digital Pin



On the following pages you will find information about the hardware setup and a manual for the reading of the sensor data..

This guide assumes that you know how to upload sketches to an Arduino and use the Serial Monitor! You can find more information in our Arduino E-Book.

This guide further assumes that you know how to run a Raspberry and use the GPIOs. More information can be found in the RaspberryPi e-book.



## All links at a glance

### Libraries

- » Library (Rpi): <https://github.com/WiringPi/WiringPi>

### Programming interfaces

- » Arduino IDE: <https://www.arduino.cc/en/Main/Software>
- » Web-Editor: <https://create.arduino.cc/editor>
- » Arduino extension for SublimeText: <https://github.com/Robot-Will/Stino>
- » Arduino extension "VisualMicro" for Atmel Studio or Microsoft Visual Studio: <http://www.visualmicro.com/page/Arduino-for-Atmel-Studio.aspx>

### Arduino Tutorials, Examples, Reference, Community

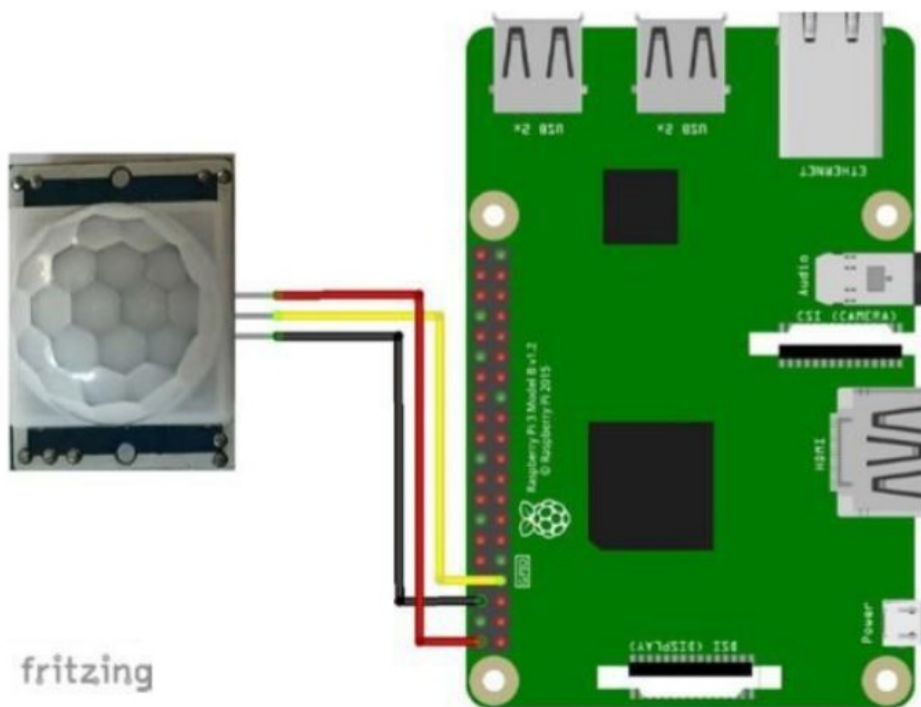
- » <https://www.arduino.cc/en/Tutorial/HomePage>
- » <https://www.arduino.cc/en/Reference/HomePage>

### Interesting from AZ-Delivery

- » Arduino Accessories:  
<https://az-delivery.de/collections/arduino-zubehor>
- » AZ-Delivery on Facebook:  
<https://www.facebook.com/AZDeliveryShop/>

## Hardware setup

Thanks to the built-in switching contact with integrated timer, this sensor is very controller-friendly and can be operated with  $3.3 \div 5V$ . On the switching module, the threshold value for the switching contact and the switching time can be set using the potentiometer. The signal pin provides a high signal, the interval for this can be set via the jumper bridge. If the jumper is at the standard position (on the outside of the board) the pin is switched and returns to low after the timer has expired. If the bridge is between pin 2 and 3, your sensor works in repeat mode and continuously shows changes depending on the movement. Any digital pin on the Arduino can be used.





## Example code at Raspberry Pi

For reading the GPIO we use the GPIO library *wiringPi*, open terminal and write these commands:

```
sudo apt-get -y install git-core
```

 $cd \sim$ 

```
git clone git://git.drogon.net/wiringPi
```

```
cd wiringPi
```

```
./build
```

After installing *wiringPi* we can test it right away:

```
gpio - v gpio readall
```

```
pi@raspberrypi3p:~/wiringPi $ gpio readall
```

Pi 3+											
BCM	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	BCM	
		3.3v			1	2		5v			
2	8	SDA.1	IN	1	3	4		5v			
3	9	SCL.1	IN	1	5	6		0v			
4	7	GPIO. 7	IN	1	7	8	0	IN	TxD	15	14
		0v			9	10	1	IN	RxD	16	15
17	0	GPIO. 0	IN	0	11	12	0	IN	GPIO. 1	1	18
27	2	GPIO. 2	IN	0	13	14		0v			
22	3	GPIO. 3	IN	0	15	16	0	IN	GPIO. 4	4	23
		3.3v			17	18	0	IN	GPIO. 5	5	24
10	12	MOSI	IN	0	19	20		0v			
9	13	MISO	IN	0	21	22	0	IN	GPIO. 6	6	25
11	14	SCLK	IN	0	23	24	1	IN	CE0	10	8
		0v			25	26	1	IN	CE1	11	7
0	30	SDA.0	IN	1	27	28	1	IN	SCL.0	31	1
5	21	GPIO.21	IN	1	29	30		0v			
6	22	GPIO.22	IN	1	31	32	0	IN	GPIO.26	26	12
13	23	GPIO.23	IN	0	33	34		0v			
19	24	GPIO.24	IN	0	35	36	0	IN	GPIO.27	27	16
26	25	GPIO.25	IN	0	37	38	0	IN	GPIO.28	28	20
		0v			39	40	0	IN	GPIO.29	29	21
BCM	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	BCM	

# Az-Delivery

## Reading the sensor data

Open terminal and write these commands:

```
cd ~
```

```
touch pir.py
```

```
nano pir.py
```

in new nano editor, write these pythin commands:

```
import RPi.GPIO as GPIO
```

```
import time
```

```
SENSOR_PIN = 4
```

```
GPIO.setmode(GPIO.BCM)
```

```
GPIO.setup(SENSOR_PIN, GPIO.IN)
```

```
def movement(channel):
```

```
    print 'There was a movement!'
```

```
try: GPIO.add_event_detect(SENSOR_PIN, GPIO.RISING, callback = movement)
```

```
    while True:
```

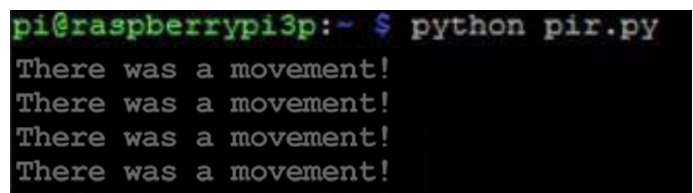
```
        time.sleep(100) except KeyboardInterrupt:
```

```
            print 'Program was terminated.'
```

```
GPIO.cleanup()
```

And in terminal type this command to run this script (output on image below):

```
python pir.py
```



```
pi@raspberrypi3p:~ $ python pir.py
There was a movement!
There was a movement!
There was a movement!
There was a movement!
```



## Reading the data on an Arduino

And continue with the reading of the data on an Arduino. To connect you can use any digital pin, in our example we use Pin 3. We connect Vcc with 5V of Uno, GND with GND of Uno and S with D3 of Uno.

And the example code looks like this:

```
int PIR=3;
int led=13;
void setup() {
    pinMode(PIR, INPUT);
    pinMode(led, OUTPUT);
    Serial.begin(9600);
}
void loop() {
    if(digitalRead(5) == HIGH) {
        digitalWrite(13, HIGH);
        Serial.println("motion detected");
    }
    else {
        digitalWrite(13, LOW); // setng led to low
        Serial.println("scanning");
    }
}
```

The output digital value can be displayed on the serial monitor or via Pin 13 (Built-In LED).

**You've done it, you can now use your sensor for your projects.**





Now it is time to learn and make the Projects on your own. You can do that with the help of many example scripts and other tutorials, which you can find on the internet.

**If you are looking for the high quality products for Arduino and Raspberry Pi, AZ-Delivery Vertriebs GmbH is the right company to get them from. You will be provided with numerous application examples, full installation guides, eBooks, libraries and assistance from our technical experts.**

<https://az-delivery.de>

Have Fun!

Impressum

<https://az-delivery.de/pages/about-us>