PIR sensors, or Passive Infra-Red sensors, can be great for Arduino and Raspberry Pi projects, and can be bought really cheap – I only paid about a dollar per sensors in bundle of 10 PIR sensors from eBay, but you can get them really cheap at places like Amazon or AliExpress as well.

In this short article we will see how we these sensors work, how the can be used, how we can test them, and possible fine tine them a little bit.

What is a PIR sensor?

A PIR sensor is an electronic sensor that measures infrared light radiating from objects in its field of view. Normally this type of sensor would be used as a motion or proximity sensor.

Quite often they are referred to as:

PIR Sensor

Motion Sensor

Proximity Sensor

Infrared (motion) Sensor

Pyroelectric sensor





The short and simple explanation on how these sensors work

The sensor in the PIR detects or “reads” infrared radiation “emitted” from objects all around us.

Each object with a temperature above absolute zero ( -273.15° Celsius, -459.67° Fahrenheit, or 0 Kelvin) will radiate infrared, even us humans, and even though we mere humans cannot see this.

With special Thermal Infrared Camera’s however, this can be made visible to the human eye.

Note that the PIR just uses a relatively simple sensor – it is most definitely not a camera!

Example of InfraRed radiation

Example of InfraRed radiation

PIRs are called “passive” since they are not assisted by any “helpers” that for example would send some form or shape of “radiation” or “light” to help detect. It’s purely based on what the sensor can pick up out of the environment, what’s being emitted by objects.

PIR’s actually only look at the “difference” between two sensor “halves”. If the difference is too high then it will trigger – it detects “motion”. This is done in a smart way, to avoid false positives caused for example by a brief flash or an increase in room temperature.

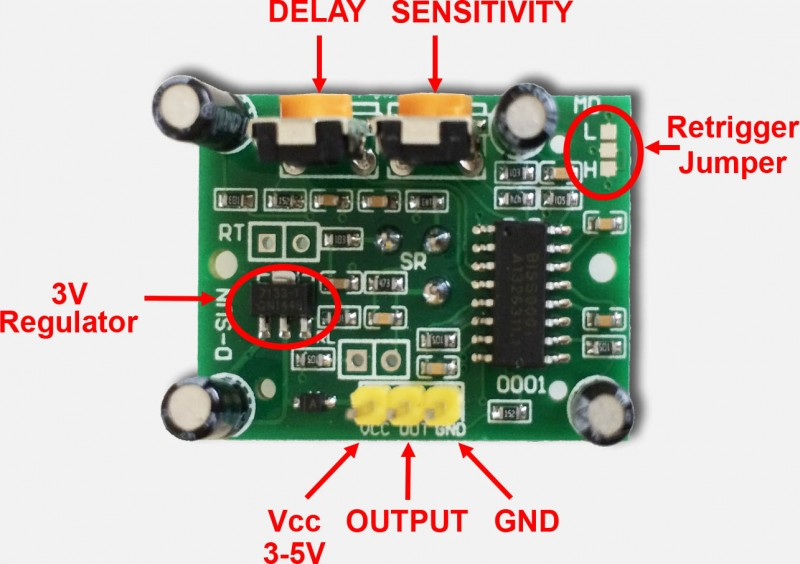
A chip and some discrete electronics handles all this for you.

PIR Lens “Dome”

As you can see in the pictures above, the PIR has some funny dome-like bubble, which is a collection of lenses covering the sensor.

If you look closer, you’ll see that the “dome” is build out of little segments – each being a small plastic Fresnel Lens. These tiny lenses help the sensor to look “around” in one swoop, which would have been impossible with just the flat sensor (see figure 2).

PIR PCB



A few points on the PCB of the PIR are of importance to us:

The most important pins are of course Power supply (Vcc 3 to 5 Volts – it is said that this can even be up to 12V) and GND (Ground).

The OUTPUT pin is the switching pin.

The two potentiometers (orange) allow you to tune Delay time (Tx) and Sensitivity (Sx).

Testing your PIR

If you’d like to experiment a little with your PIR, or test how well it works, then you actually do not even need an Arduino or Raspberry Pi.

All you need is either a power source, a resistor and a LED.

The power can be drawn from either batteries or a power supply – 3 to 9 Volts will work – I used a 5V USB charger.

For the LED, we need to pay attention, since it’s important that we connect the pins right. The longest pin of the LED (Anode, or “plus”) should be connected to the resistor, see the LED drawing below.

The resistor should be 470 Ohms; the colours are Yellow, Violet, and Brown (see also the Tweaking4All resistor calculator).

Summarized:

Connect the PIR Vcc pin to the + of your power source.

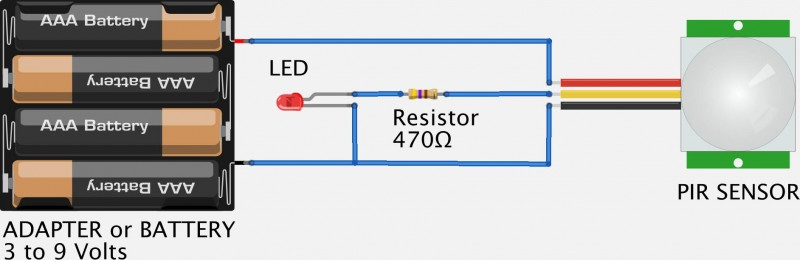
Connect the PIR Output pin to one end of the resistor.

Connect the other end of the resistor to the long leg (Anode) of the LED.

Connect the short leg of the LED (Cathode) to the − of your power source.

Connect the PIR GND to the − of your power source.

And that’s all … you should have something like this now:



Testing your PIR with a Battery, LED and a Resistor

Playing with the “settings”

Now that we have a test setup, time to do some playing.

When the LED goes on, motion is detected.

In the beginning, you might notice some seemingly erratic behaviour – this is perfectly normal. We need to understand a few things before we can tweak the settings.

When connecting the battery, the sensor will take up to 30 to 60 seconds to stabilize (warm up).

Place your setup in such a way that there will be no motion and wait until the LED remains OFF.

Once the LED remains OFF, you can move your hand or anything for that matter, in front of the little dome, and see the LED go ON and OFF.

Depending on what is moving around, the detectable range should be up to 20 feet or about 6 meters.

Also keep in mind that the sensor will remain “ON” for a little bit after it detects motion. At a later time, you can tune the timing with the “Delay Time” potentiometer.

Delay Time

The “Delay Time”, determines how long the PIR will keep the Output “High” (ON) after detecting motion.

For example, when motion has been detected, you could set this somewhere between a few seconds (mine has an approximately minimum delay time of 2 seconds) up to a few minutes (specifications of mine claim about 200 seconds).

When assembling the basic test setup (above) and the LED seems to stay on forever, turn this dial down – the delay might be too long. The lowest setting, when facing the philips screw of the potentiometer, is all the way to left.

When doing your first tests, turn it as low as possible … until you notice there is a need for a time delay.

Sensitivity

With this potentiometer you can determine set the “range” you have in mind.

I have the impression that it regulates how much motion it takes to be “seen”, or maybe more technical terms: how much difference between the two “halves” of the sensor need to see before it’s considered motion.

You’ll have to play a little with this to see what is the appropriate setting for your purposes.

An increased sensitivity can be beneficial for when using a PIR for long range, say up to 20 feet (6 meters) or more. This can however also cause false positives in smaller spaces – i.e. the PIR might trigger when it should not.

A decreased sensitivity is good for a short range, say half of the maximum range or up to 10 feet (3 meters). Which in turn could miss movement at a longer range. Objects further away may need more motion to be detected.