

Phase 3 : Development Part 1

Building an IoT sensor system with Raspberry Pi integration involves several steps.

1. Select Sensors:

Choose the sensors based on the data you want to collect (e.g., temperature, humidity, motion). Ensure they are compatible with Raspberry Pi.

2. Raspberry Pi Setup:

Get a Raspberry Pi (any model should work). Install an operating system (Raspberry Pi OS is a good choice).

Update and upgrade the system:

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

3. Connect Sensors:

Wire the sensors to the Raspberry Pi's GPIO pins. Refer to the datasheets and documentation for correct wiring.

4. Python Programming:

Write Python code to read data from the sensors. Libraries like GPIO Zero and Adafruit Circuit Python can be helpful.

5. Data Storage:

Decide where to store the sensor data. Options include local storage, cloud services, or databases.

6. Data Transmission:

If you're sending data to the cloud, use IoT protocols like MQTT or HTTP to transmit the data.

7. Data Visualization:

Create a dashboard or use a service to visualize the data. Tools like Grafana or ThingSpeak can help.

8. Automation:

Implement rules and automation based on sensor data (e.g., sending alerts when a threshold is reached).

9. Security:

Secure your Raspberry Pi by changing default passwords and using encryption for data transmission.

10. Power Supply:

Ensure your Raspberry Pi and sensors have a reliable power source.

11. Testing and Calibration:

Test the system, calibrate sensors, and ensure data accuracy.

12. Monitoring:

Implement remote monitoring and error handling for the system.

13. Documentation:

Document your setup, connections, and code for future reference.

14. Scalability:

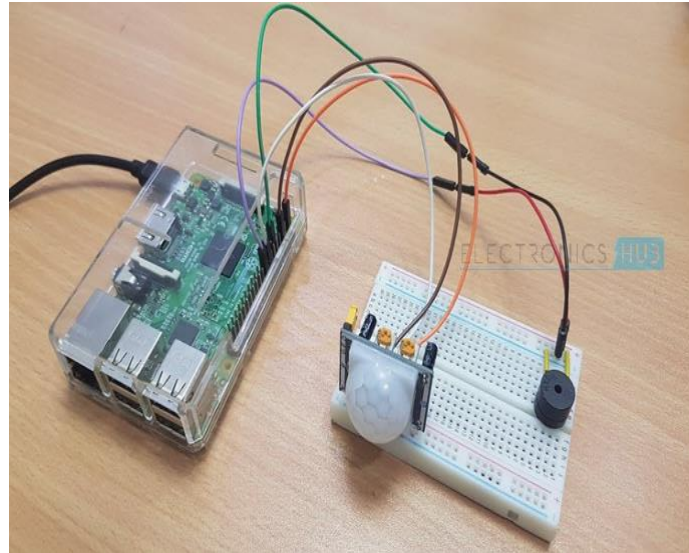
Consider how to scale the system if you want to add more sensors or devices.

15. Maintenance:

Regularly update software and perform maintenance as needed.

CONNECTING PIR MOTION SENSOR USING RASPBERRY PI

In this project, we will see how to implement a PIR Motion Sensor using Raspberry Pi by learning how to interface a PIR Sensor with Raspberry Pi. In this project, when the PIR Sensor detects any human motion, a buzzer is activated. With this interface, you can implement a simple Motion Detection Alarm as a DIY Project.



Outline:

- Overview
- PIR Sensor
- Requirements
- PIR Sensor Adjustments
- PIR Motion Sensor using Raspberry Pi
- Connection
- Circuit Diagram
- Operation
- Consideration
- Working
- Applications

Overview:

Passive Infrared Sensors, often referred to as PIR Sensors (also IR Motion Sensors and Pyroelectric Sensors), are Motion Detectors that basically detect the changes in Infrared Radiations emitted by a person. Passive Infrared Sensors, often referred to as PIR Sensors (also IR Motion Sensors and

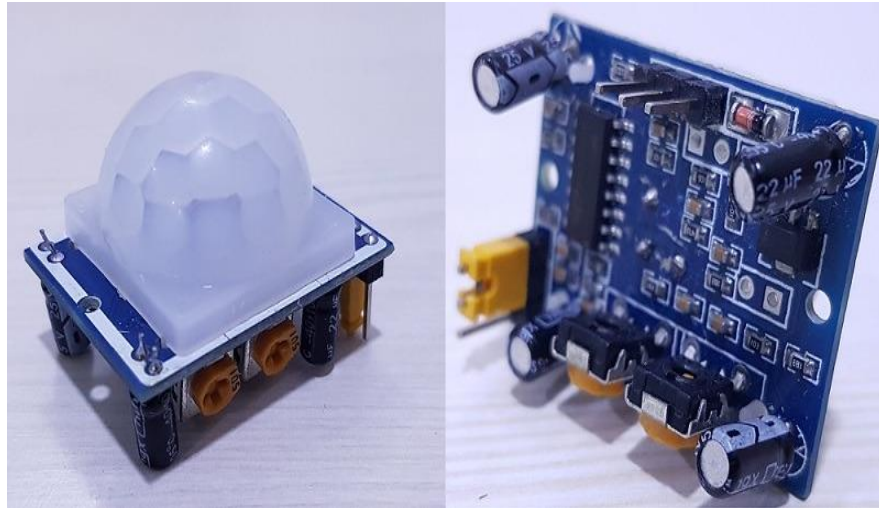
Pyroelectric Sensors), are Motion Detectors that basically detect the changes in Infrared Radiations emitted by a person. Every living and non-living thing which has a temperature greater than absolute zero will emit infrared radiations. Since the emitted energy is in the form of infrared radiation, whose wavelength is greater than that of our visible light, we humans cannot see those radiations.

But PIR Sensors are built to detect those infrared radiations. Hence, they are employed in a variety of applications like Motion Detectors, Security Systems, Intruder Alert and so forth.

The term “Passive” in the PIR Sensor means that the sensor will not emit any infrared energy but rather detects infrared radiations emitted by other objects. This is in contrast to active sensors, which perform both the actions (emitting and detection).

PIR(Passive Infrared) Sensor:

PIR (Passive Infrared) Sensor: A PIR sensor is a motion sensor that detects changes in infrared radiation, primarily the heat emitted by objects, including people and vehicles.



Requirements:

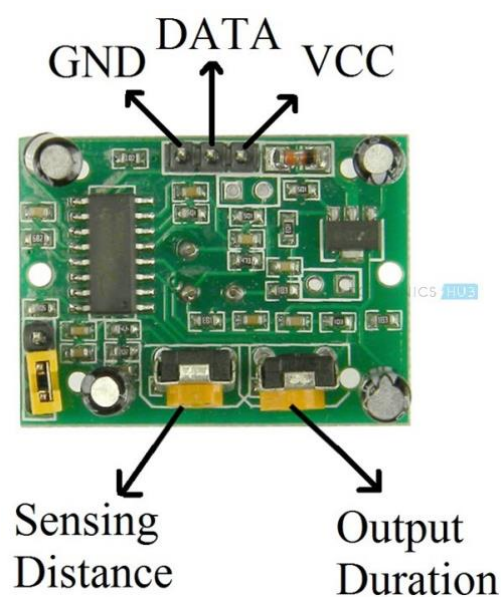
- Raspberry Pi.
- PIR sensor module.
- 5V PIN.
- Jumper wires.
- Power source (like a USB power bank or an external power supply for the Raspberry Pi).

PIR Sensor Adjustments:

As mentioned earlier, there are two potentiometers for manually adjusting the sensitivity and output timing.

With the help of the first potentiometer, you can adjust the Sensitivity i.e. the Sensing Distance of the PIR Sensor. The range can be adjusted from 3 meters up to 8 meters. To increase the sensing distance, turn the POT in clockwise direction and to decrease, turn it in anti-clockwise direction.

Coming to the second potentiometer, you can adjust the duration for which the Output of the PIR Sensor stays HIGH. It can be varied anywhere between 0.3s to 600s. To increase the time, turn the POT in clockwise direction and in anti-clockwise direction to decrease the time.



PIR Motion Sensor using Raspberry Pi:

As mentioned in the introduction, the aim of this project is to interface a PIR Sensor with Raspberry Pi and implement a PIR Motion Sensor using Raspberry Pi and Python Programming.

I have already implemented PIR Motion Sensor Module in several other projects like Automatic Room Lights using Arduino and PIR Sensor, Automatic Door Opener using Arduino and PIR Sensor, Arduino PIR Sensor Tutorial and GSM Based Home Security Alarm System Using Arduino.

The main advantage of the PIR Motion Sensor using Raspberry Pi over all the above mentioned projects is that Raspberry Pi

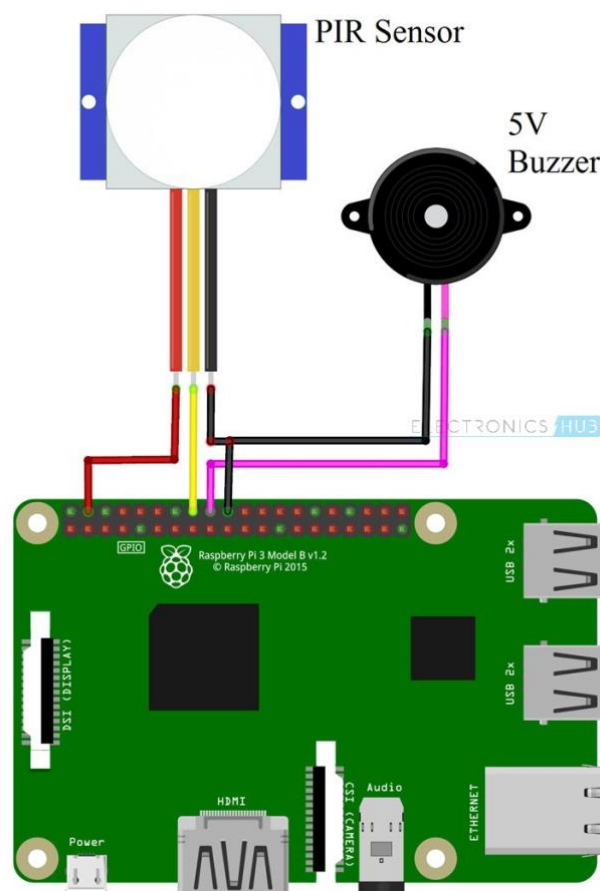
can be easily connected to the Internet and allows IoT based application of the project.

Connections:

Connect the VCC (power) pin of the PIR sensor to a 5V pin on the Raspberry Pi. Connect the GND (ground) pin of the PIR sensor to a ground pin on the Raspberry Pi. Connect the OUT (output) pin of the PIR sensor to a GPIO pin on the Raspberry Pi, such as GPIO17.

Circuit Diagram:

The following Fritzing based images shows all the connections with respect to the PIR Motion Sensor using Raspberry Pi.



Operation:

When a vehicle or a person moves within the PIR sensor's range, it will detect the motion and trigger the sensor's output pin. Your Raspberry Pi will read this signal, and you can use it to track parking space occupancy or trigger other actions, such as capturing an image with a camera module.

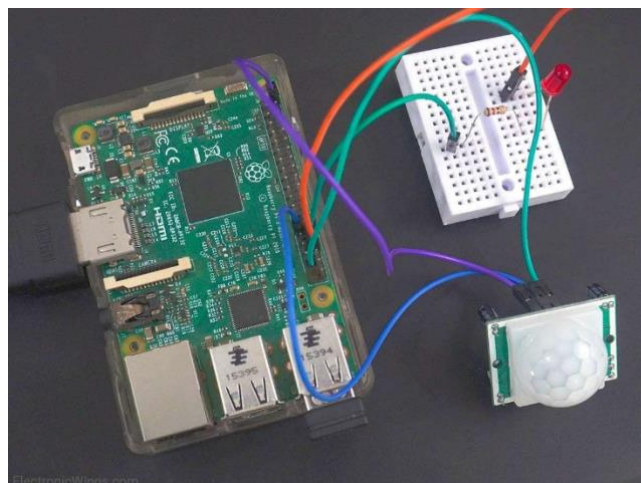
Considerations:

Adjust the PIR sensor's sensitivity and delay settings as needed. Secure the PIR sensor in a way that minimizes false detections from nearby movement. Combine PIR sensors with other sensors or cameras for a more robust smart parking system.

Working:

The working of the PIR Motion Sensor using Raspberry Pi is very simple. If the PIR Sensor detects any human movement, it raises its Data Pin to HIGH.

Raspberry Pi upon detecting a HIGH on the corresponding input pin, will activate the Buzzer.



Conclusion:

Here is the explanation of connecting PIR Sensor using Raspberry Pi for Smart Parking.

By this smart parking process the time were saved and reduction in Green House gases and also the reduction in traffic volumes in built up areas.