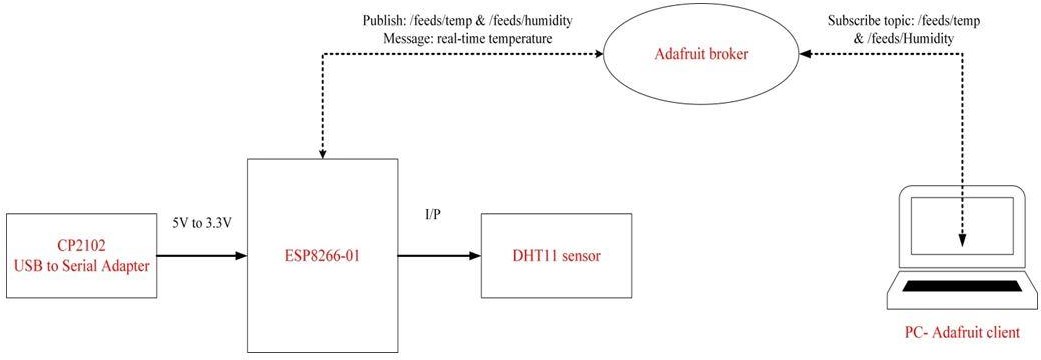
TEMPERATURE SENSOR

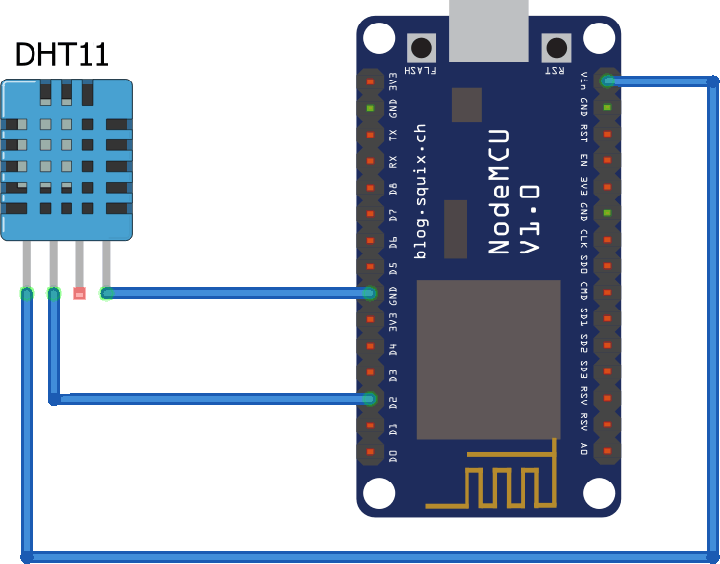
AIM

To create the block diagram and program for Temperature Sensor by using Industrial IOT.

Materials Required:

* Esp 8266
* Jumber wire
* DHT11
* Power source Software required :
* Arduino IDE Block diagram :





Procedure:

* + A temperature sensor is a device, typically, a thermocouple or resistance temperature detector, that provides temperature measurement in a readable form through an electrical signal.
  + A thermometer is the most basic form of a temperature meter that is used to measure the degree of hotness and coolness.
  + A temperature sensor is a device that measures the temperature of an object or its surroundings and converts it into an electrical signal that can be displayed, recorded, or used to control a system.
  + The temperature sensor contains a sensing element that responds to changes in temperature. There are various types of temperature sensing elements, including:
  + A thermistor is a type of resistor whose resistance changes with temperature. Negative Temperature Coefficient (NTC) thermistors decrease in resistance as temperature increases, while Positive Temperature Coefficient (PTC) thermistors increase in resistance with temperature.
  + An RTD is a temperature sensor whose resistance changes in a predictable manner with temperature. RTDs are often made of metals such as platinum, nickel, or copper.
  + IC temperature sensors use semiconductor materials to measure temperature. They often contain a temperature-sensitive voltage reference or a temperature-dependent current source.
  + When the temperature changes, the sensing element responds by altering its electrical properties, such as resistance, voltage, or current.

Program :

#include "DHT.h"

#define DPIN 4 //Pin to connect DHT sensor (GPIO number) D2 #define DTYPE DHT11 // Define DHT 11 or DHT22 sensor type

DHT dht(DPIN,DTYPE);

void setup() { Serial.begin(9600); dht.begin();

}

void loop() { delay(2000);

float tc = dht.readTemperature(false); //Read temperature in C float tf = dht.readTemperature(true); //Read Temperature in F float hu = dht.readHumidity(); //Read Humidity

Serial.print("Temp: "); Serial.print(tc); Serial.print(" C, "); Serial.print(tf); Serial.print(" F, Hum: "); Serial.print(hu); Serial.println("%");

}

# Result

When the temperature is used to measure the heat and cold of the object

PRESSURE SENSOR

Pressure sensor

AIM

To create the block diagram and program for Pressure Sensor by using Industrial IOT.

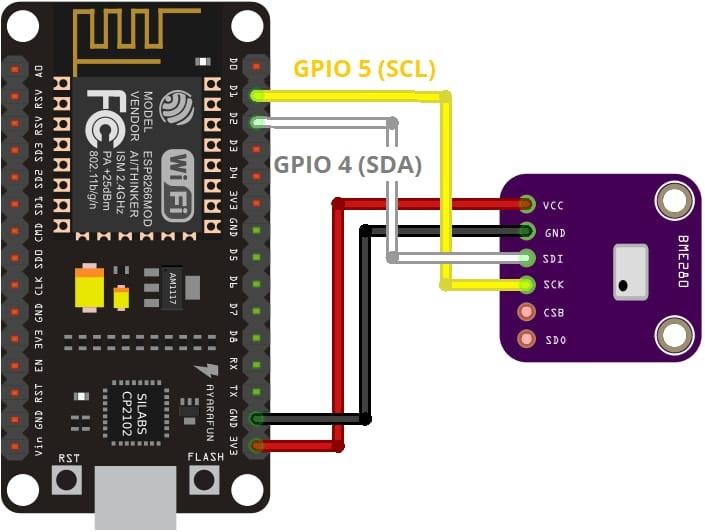
Materials Required:

* Esp 8266
* Jumber wire
* Sensor
* Power source Software required :
* Arduino IDE Block Diagram :

Power source

ESP 8266

LED



Procedure

A pressure sensor is defined as a transducer that converts an input mechanical pressure into an electrical output signal (pressure sensor definition). There are several types of pressure sensors based on size, capacity, measurement method, sensing technology and output requirements.

A [pressure sensor](https://www.techopedia.com/definition/15002/pressure-sensor) is a device that senses and measures pressure. In this case, pressure is defined as the amount of force exerted over an area.

Pressure sensors allow for more specialized maintenance strategies, such as predictive maintenance.

These devices collect real-time data on the conditions of equipment.

By installing pressure sensors to certain tanks and other pressurized assets, the sensors are able to alert maintenance teams when the pressure falls outside of a specified level. This allows the teams to address the issue immediately.

Pressure sensors work by measuring a physical change that happens, as a reaction to pressure differences.

After measuring these physical changes, the information is converted into electric signals. These signals can then be displayed as usable data that the team can then interpret.

Program

#include <ESP8266WebServer.h> #include <Wire.h>

#include <Adafruit\_Sensor.h>

#include <Adafruit\_BMP280.h>

#define SEALEVELPRESSURE\_HPA (1013.25)

#define BMP\_SCK (13)

#define BMP\_MISO (12)

#define BMP\_MOSI (11)

#define BMP\_CS (10)

Adafruit\_BMP280 bmp; // I2C

float temperature, humidity, pressure, altitude;

/Put your SSID & Password/

const char\* ssid = "sathya"; // Enter SSID here

const char\* password = "36113422"; //Enter Password here ESP8266WebServer server(80);

void setup() { Serial.begin(115200); delay(100); bmp.begin(0x76);

Serial.println("Connecting to "); Serial.println(ssid);

//connect to your local wi-fi network WiFi.begin(ssid, password);

//check wi-fi is connected to wi-fi network while (WiFi.status() != WL\_CONNECTED) { delay(1000);

Serial.print(".");

}

Serial.println(""); Serial.println("WiFi connected..!");

Serial.print("Got IP: "); Serial.println(WiFi.localIP()); server.on("/", handle\_OnConnect); server.onNotFound(handle\_NotFound); server.begin();

Serial.println("HTTP server started");

}

void loop() { server.handleClient();

}

void handle\_OnConnect() { temperature = bmp.readTemperature();

// humidity = bmp.readHumidity(); pressure = bmp.readPressure() / 100.0F;

altitude = bmp.readAltitude(SEALEVELPRESSURE\_HPA); server.send(200, "text/html", SendHTML(temperature,pressure,altitude));

}

void handle\_NotFound(){

server.send(404, "text/plain", "Not found");

}

String SendHTML(float temperature, float pressure,float altitude){ String ptr = "<!DOCTYPE html>";

ptr +="<html>"; ptr +="<head>";

ptr +="<title>ESP8266 Weather Station</title>";

ptr +="<meta name='viewport' content='width=device-width, initial- scale=1.0'>";

ptr +="<link href='https://fonts.googleapis.com/css?family=Open+Sans:300,400,600' rel='stylesheet'>";

ptr +="<style>";

ptr +="html { font-family: 'Open Sans', sans-serif; display: block; margin: 0px auto; text-align: center;color: #444444;}";

ptr +="body{margin: 0px;} ";

ptr +="h1 {margin: 50px auto 30px;} ";

ptr +=".side-by-side{display: table-cell;vertical-align: middle;position: relative;}";

ptr +=".text{font-weight: 600;font-size: 19px;width: 200px;}";

ptr +=".reading{font-weight: 300;font-size: 50px;padding-right: 25px;}"; ptr +=".temperature .reading{color: #F29C1F;}";

ptr +=".humidity .reading{color: #3B97D3;}"; ptr +=".pressure .reading{color: #26B99A;}"; ptr +=".altitude .reading{color: #955BA5;}";

ptr +=".superscript{font-size: 17px;font-weight: 600;position: absolute;top: 10px;}";

ptr +=".data{padding: 10px;}";

ptr +=".container{display: table;margin: 0 auto;}"; ptr +=".icon{width:65px}";

ptr +="</style>"; ptr +="</head>"; ptr +="<body>";

ptr +="<h1>BMP280 ESP8266 Weather Station</h1>"; ptr +="<div class='container'>";

ptr +="<div class='data temperature'>"; ptr +="<div class='side-by-side icon'>";

ptr +="<svg enable-background='new 0 0 19.438 54.003'height=54.003px id=Layer\_1 version=1.1 viewBox='0 0 19.438 54.003'width=19.438px x=0px xml:space=preserve xmlns=<http://www.w3.org/2000/svg> xmlns:xlink[=h](http://www.w3.org/1999/xlink)t[tp://www.w3.org/1999/xlink](http://www.w3.org/1999/xlink) y=0px><g><path d='M11.976,8.82v- 2h4.084V6.063C16.06,2.715,13.345,0,9.996,0H9.313C5.965,0,3.252,2.715,3.2

52,6.063v30.982";

ptr

+="C1.261,38.825,0,41.403,0,44.286c0,5.367,4.351,9.718,9.719,9.718c5.368,0, 9.719-4.351,9.719-9.718";

ptr +="c0-2.943-1.312-5.574-3.378-7.355V18.436h-3.914v-2h3.914v-2.808h- 4.084v-2h4.084V8.82H11.976z M15.302,44.833";

ptr +="c0,3.083-2.5,5.583-5.583,5.583s-5.583-2.5-5.583-5.583c0-2.279,1.368-

4.236,3.326-5.104V24.257C7.462,23.01,8.472,22,9.719,22";

ptr

+="s2.257,1.01,2.257,2.257V39.73C13.934,40.597,15.302,42.554,15.302,44.83

3z'fill=#F29C21 /></g></svg>"; ptr +="</div>";

ptr +="<div class='side-by-side text'>Temperature</div>"; ptr +="<div class='side-by-side reading'>";

ptr +=(int)temperature;

ptr +="<span class='superscript'>&deg;C</span></div>"; ptr +="</div>";

ptr +="<div class='data humidity'>";

ptr +="<div class='side-by-side icon'>";

ptr +="<svg enable-background='new 0 0 29.235 40.64'height=40.64px id=Layer\_1 version=1.1 viewBox='0 0 29.235 40.64'width=29.235px x=0px xml:space=preserve xmlns=<http://www.w3.org/2000/svg> xmlns:xlink[=h](http://www.w3.org/1999/xlink)t[tp://www.w3.org/1999/xlink](http://www.w3.org/1999/xlink) y=0px><path d='M14.618,0C14.618,0,0,17.95,0,26.022C0,34.096,6.544,40.64,14.618,40.64s 14.617-6.544,14.617-14.617";

ptr +="C29.235,17.95,14.618,0,14.618,0z M13.667,37.135c-5.604,0-10.162- 4.56-10.162-10.162c0-0.787,0.638-1.426,1.426-1.426";

ptr

+="c0.787,0,1.425,0.639,1.425,1.426c0,4.031,3.28,7.312,7.311,7.312c0.787,0,1

.425,0.638,1.425,1.425";

ptr +="C15.093,36.497,14.455,37.135,13.667,37.135z'fill=#3C97D3

/></svg>";

ptr +="</div>";

ptr +="<div class='side-by-side text'>Pressure</div>"; ptr +="<div class='side-by-side reading'>";

ptr +=(int)pressure;

ptr +="<span class='superscript'>hPa</span></div>"; ptr +="</div>";

ptr +="<div class='data altitude'>";

ptr +="<div class='side-by-side icon'>";

ptr +="<svg enable-background='new 0 0 58.422 40.639'height=40.639px id=Layer\_1 version=1.1 viewBox='0 0 58.422 40.639'width=58.422px x=0px xml:space=preserve xmlns=<http://www.w3.org/2000/svg> xmlns:xlink[=h](http://www.w3.org/1999/xlink)t[tp://www.w3.org/1999/xlink](http://www.w3.org/1999/xlink) y=0px><g><path d='M58.203,37.754l0.007-0.004L42.09,9.935l-0.001,0.001c-0.356-0.543- 0.969-0.902-1.667-0.902";

ptr +="c-0.655,0-1.231,0.32-1.595,0.808l-0.011-0.007l-0.039,0.067c-

0.021,0.03-0.035,0.063-0.054,0.094L22.78,37.692l0.008,0.004";

ptr +="c-0.149,0.28-0.242,0.594- 0.242,0.934c0,1.102,0.894,1.995,1.994,1.995v0.015h31.888c1.101,0,1.994 -

0.893,1.994-1.994";

ptr +="C58.422,38.323,58.339,38.024,58.203,37.754z'fill=#955BA5 /><path d='M19.704,38.674l-0.013-0.004l13.544-23.522L25.13,1.156l- 0.002,0.001C24.671,0.459,23.885,0,22.985,0";

ptr +="c-0.84,0-1.582,0.41-2.051,1.038l-0.016-0.01L20.87,1.114c-

0.025,0.039-0.046,0.082-0.068,0.124L0.299,36.851l0.013,0.004";

ptr

+="C0.117,37.215,0,37.62,0,38.059c0,1.412,1.147,2.565,2.565,2.565v0.015h16

.989c-0.091-0.256-0.149-0.526-0.149-0.813";

ptr +="C19.405,39.407,19.518,39.019,19.704,38.674z'fill=#955BA5

/></g></svg>"; ptr +="</div>";

ptr +="<div class='side-by-side text'>Altitude</div>"; ptr +="<div class='side-by-side reading'>";

ptr +=(int)altitude;

ptr +="<span class='superscript'>m</span></div>"; ptr +="</div>";

ptr +="</div>"; ptr +="</body>"; ptr +="</html>"; return ptr;

}

Result

Pressure sensors can be categorized according to the method they sense

pressure changes. Sensors can observe and measure various physical reactions

PULSE SENSOR

PULSE SENSOR

ESP - 8266

AIM

To create the block diagram and program for Pulse Sensor by using Industrial IOT.

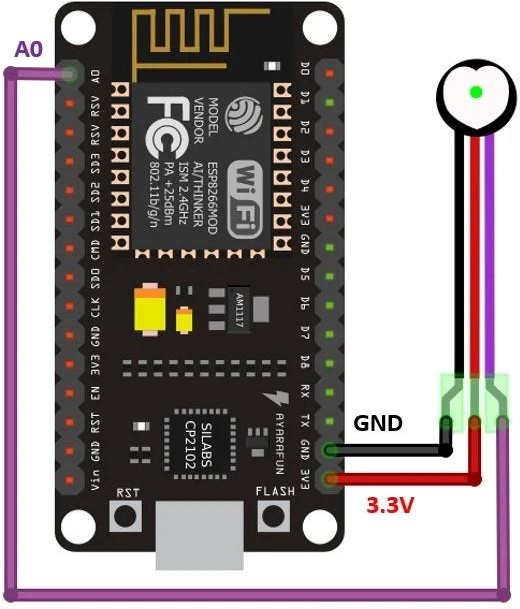
Materials Required:

* Esp 8266
* Jumber wire
* Pulse sensor
* Power source Software required :
* Arduino IDE Block diagram :

LED

POWER SOURCE

THING SPEAK



Program

// Variables

int PulseSensorPurplePin = 0; // Pulse Sensor PURPLE WIRE connected to ANALOG PIN 0

int LED13 = 13; // The on-board Arduion LED

int Signal; // holds the incoming raw data. Signal value can range from 0-1024

int Threshold = 550; // Determine which Signal to "count as a beat", and which to ingore.

// The SetUp Function:

void setup() {

pinMode(LED13,OUTPUT); // pin that will blink to your heartbeat! Serial.begin(9600); // Set's up Serial Communication at certain speed.

}

// The Main Loop Function void loop() {

Signal = analogRead(PulseSensorPurplePin); // Read the PulseSensor's value.

// Assign this value to the "Signal" variable.

Serial.println(Signal); // Send the Signal value to Serial Plotter.

if(Signal > Threshold){ // If the signal is above "550", then "turn-on" Arduino's on-Board LED.

digitalWrite(LED13,HIGH);

} else {

digitalWrite(LED13,LOW); // Else, the sigal must be below "550", so "turn-off" this LED.

}

delay(500);

}

Procedure

Install the Arduino ide software to your computer

Go the preference page paste the URL for installing esp 8266 driver by using Arduino ide

Check the ports is available or instal CH 340 driver from chrome and install that for port

After that install the library files from library manager pressure sensor need a library for pulse sensor.

An alternate name of this sensor is heartbeat sensor or heart rate sensor. The working of this sensor can be done by connecting it from the fingertip or human ear to Arduino board. So that heart rate can be easily calculated.

A [color code](https://www.elprocus.com/capacitor-color-code/) cable is connected to header connectors. So this sensor is easily connected to an Arduino into the project without soldering.

An ear clip size is the same as a heart rate sensor and it can be connected using hot glue at the backside of [the sensor](https://www.elprocus.com/mems-sensor-working-and-its-applications/) to wear on the earlobe.

Two Velcro dots are completely sized toward the sensor at the hook side. These are extremely useful while making a Velcro strap to cover approximately a fingertip.

This is used to cover the Sensor around the finger.

Transparent strikers are protection layers used to protect the sensor from sweaty earlobes and fingers.

This sensor includes three holes in the region of the external edge so that one can easily connect anything to it.

The **pulse sensor working principle** is very simple. This sensor has two surfaces, on the first surface, the [light-emitting diode](https://www.elprocus.com/light-emitting-diode-led-working-application/) & ambient light sensor is connected.

Result

This sensor uses an easy optical pulse sensor along with amplification &

cancellation of noise to make a circuit.

TOUCH SENSOR

AIM

To create the block diagram and program for touch Sensor by using Industrial IOT.

Materials Required:

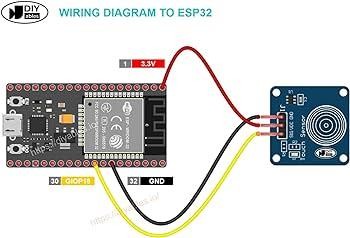
* Esp 8266
* Jumber wire
* Touch sensor
* Power source Software required :
* Arduino IDE Block diagram :

Touch sensor

ESP - 8266

POWER SOURCE

Door



Procedure

Install the Arduino ide software to your computer

Go the preference page paste the URL for installing esp 8266 driver by using Arduino ide

Check the ports is available or instal CH 340 driver from chrome and install that for port

Touch Sensors are the electronic sensors that can detect touch. They operate as a switch when touched.

These sensors are used in lamps, touch screens of the mobile, etc… Touch sensors offer an intuitive user interface.

Touch sensors are also known as Tactile sensors. These are simple to design, low cost and are produced in large scale.

With the advance in technology, these sensors are rapidly replacing the mechanical switches. Based on their functions there are two types of touch sensors- Capacitive sensor and Resistive sensor

Capacitive sensors work by measuring capacitance and are seen in portable devices. These are durable, robust and attractive with low cost. Resistive sensors don’t depend on any electrical properties for operation. These sensors work by measuring the pressure applied to their surface

Touch sensors work similar to a switch. When they are subjected to touch, pressure or force they get activated and acts as a closed switch. When the pressure or contact is removed they act as an open switch.

Program

#define TOUCH\_PIN D1 // Define the GPIO pin connected to the touch sensor void setup() {

Serial.begin(115200); // Initialize serial communication pinMode(TOUCH\_PIN, INPUT); // Set touch pin as input

}

void loop() {

int touchValue = digitalRead(TOUCH\_PIN); // Read the value of touch pin

if(touchValue == HIGH) {

Serial.println("Touch Detected!"); // Print message if touch is detected

// You can add any actions you want to perform when touch is detected

}

delay(100); // Add a small delay to avoid rapid toggling of output

}

Result

Touch Sensors are the electronic sensors that can detect touch. his is a

basic example to get you started. Depending on your specific requirements and the capabilities of your touch sensor, you may need to adjust the code accordingly.

IR SENSOR

AIM

To create the block diagram and program for IR Sensor by using Industrial IOT.

Materials Required:

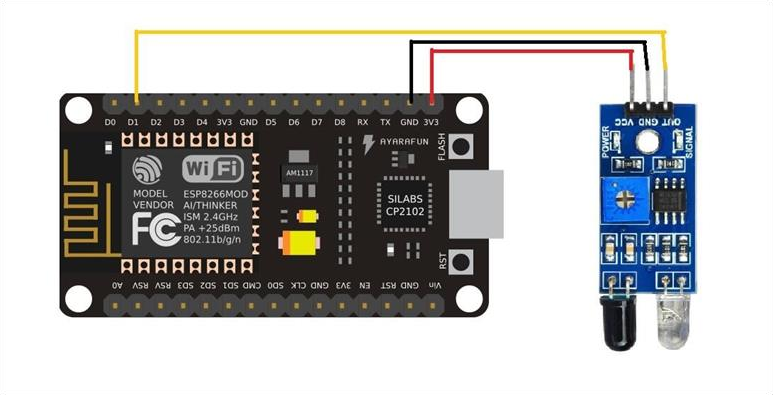
* Esp 8266
* Jumber wire
* IR sensor
* Power source Software required :
* Arduino IDE Block diagram :

IR Sensor

ESP - 8266

POWER SOURCE

Car Parking



Program

#define IR\_PIN D1 // Define the GPIO pin connected to the IR sensor

void setup() {

Serial.begin(115200); // Initialize serial communication pinMode(IR\_PIN, INPUT); // Set IR pin as input

}

void loop() {

int irValue = digitalRead(IR\_PIN); // Read the value of IR pin

if(irValue == LOW) {

Serial.println("Obstacle Detected!"); // Print message if obstacle is detected

// You can add any actions you want to perform when an obstacle is detected

}

delay(100); // Add a small delay to avoid rapid toggling of output

}

Procedure

Install the Arduino ide software to your computer

Go the preference page paste the URL for installing esp 8266 driver by using Arduino ide

Check the ports is available or instal CH 340 driver from chrome and install that for port

IR sensor is an electronic device, that emits the light in order to sense some object of the surroundings.

An [IR sensor](https://robu.in/product-category/sensor/ir-and-pir-sensor/) can measure the heat of an object as well as detects the motion. Usually, in the [**infrared spectrum**](https://en.wikipedia.org/wiki/Infrared_spectroscopy), all the objects radiate some form of thermal radiation.

These types of radiations are invisible to our eyes, but infrared sensor can detect these radiations.

The emitter is simply an IR LED [(Light Emitting Diode](https://robu.in/product-category/display-boards/led/)) and the detector is simply an IR photodiode. Photodiode is sensitive to IR light of the same wavelength which is emitted by the IR LED.

When IR light falls on the photodiode, the resistances and the output voltages will change in proportion to the magnitude of the IR light received.

There are five basic elements used in a typical infrared detection system: an infrared source, a transmission medium, optical component, infrared detectors or receivers and signal processing.

Infrared lasers and Infrared LED’s of specific wavelength used as infrared sources.

The three main types of media used for infrared transmission are vacuum, atmosphere and optical fibers. Optical components are used to focus the infrared radiation or to limit the spectral response.

Result

IR sensor is an electronic device, that emits the light in order to sense

some object of the surroundings. An [IR sensor](https://robu.in/product-category/sensor/ir-and-pir-sensor/) can measure the heat of an object as well as detects the motion

MOTION SENSOR

AIM

To create the block diagram and program for MOTION SENSOR by using Industrial IOT.

Materials Required:

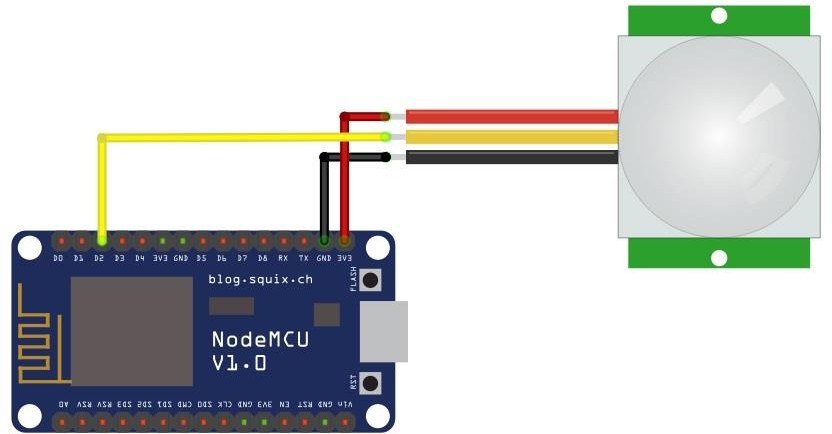
* Esp 8266
* Jumber wire
* Motion sensor
* Power source Software required :
* Arduino IDE Block diagram :

Motion sensor

ESP - 8266

POWER SOURCE

Car



Program

#define PIR\_PIN D1 // Define the GPIO pin connected to the PIR sensor

void setup() {

Serial.begin(115200); // Initialize serial communication pinMode(PIR\_PIN, INPUT); // Set PIR pin as input

}

void loop() {

int pirValue = digitalRead(PIR\_PIN); // Read the value of PIR pin

if(pirValue == HIGH) {

Serial.println("Motion Detected!"); // Print message if motion is detected

// You can add any actions you want to perform when motion is detected

}

delay(100); // Add a small delay to avoid rapid toggling of output

}

# Procedure

Install the Arduino ide software to your computer

Go the preference page paste the URL for installing esp 8266 driver by using Arduino ide

Check the ports is available or instal CH 340 driver from chrome and install that for port

A [passive infrared sensor](https://robu.in/product/pir-motion-sensor-detector-module-hc-sr501/) is an electronic sensor that measures infrared light radiating from objects. PIR sensors mostly used in PIR-based motion detectors. Also, it used in security alarms and automatic lighting applications. The below image shows a typical pin configuration of the PIR sensor, which is quite simple to understand the pinouts. The PIR sensor consist of 3 pins,

Generally, [**PIR sensor**](https://robu.in/product/hc-sr505-mini-infrared-pir-motion-sensor-infrared-detector-module/) can detect animal/human movement in a requirement range. PIR is made of a pyroelectric sensor, which is able to detect different levels of infrared radiation. The detector itself does not emit any energy but passively receives it.

It detects infrared radiation from the environment. Once there is infrared radiation from the human body particle with temperature, focusing on the optical system causes the pyroelectric device to generate a sudden electrical signal.

Simply, when a human body or any animal passes by, then it intercepts the first slot of the PIR sensor. This causes a positive differential change between the two bisects. When a human body leaves the sensing area, the sensor generates a negative differential change between the two bisects.

He passive infrared sensor does not radiate energy to space. It receives the infrared radiation from the human body to make an alarm.

Any object with temperature is constantly radiating infrared rays to the outside world. The surface temperature of the human body is between 36° C - 27

° C and most of its radiant energy concentrated in the wavelength range of 8 um- 12 um.

RESULT

A [passive infrared sensor](https://robu.in/product/pir-motion-sensor-detector-module-hc-sr501/) is an electronic sensor that measures infrared light radiating from objects

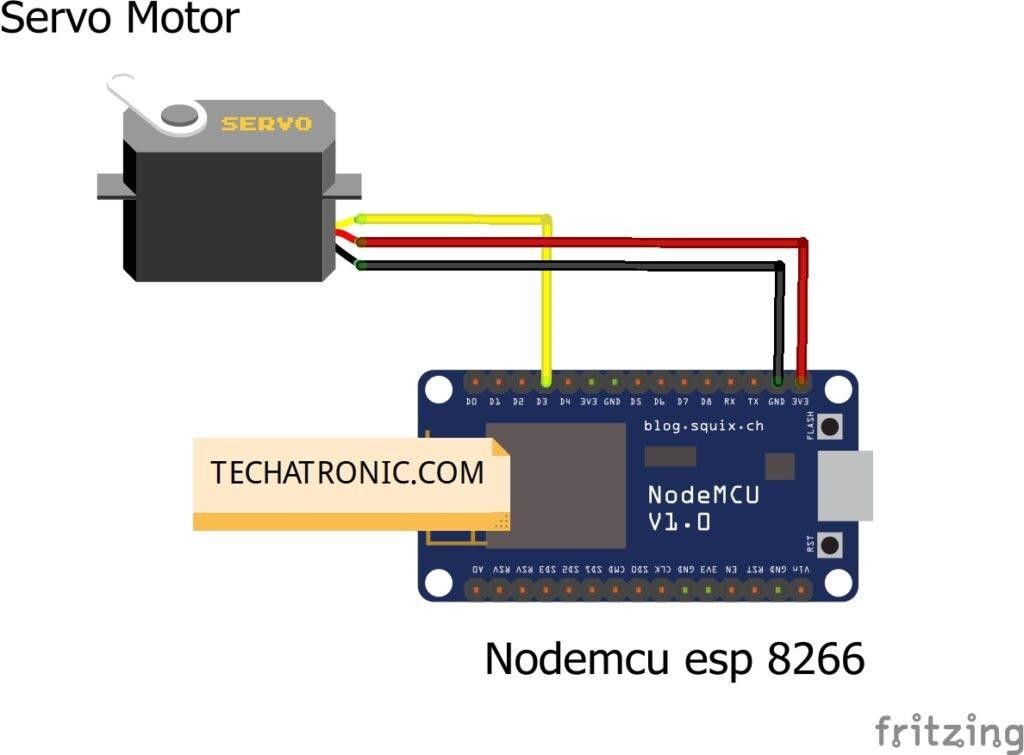
SERVO MOTOR

AIM

To create the block diagram and program for SERVO MOTOR by using Industrial IOT.

Materials Required:

* Esp 8266
* Jumber wire
* SERVO
* Power source Software required :
* Arduino IDE Block diagram



Program

#include <Servo.h>

#define SERVO\_PIN D1 // Define the GPIO pin connected to the servo motor Servo servo; // Create a servo object

void setup() {

servo.attach(SERVO\_PIN); // Attach the servo object to the specified pin

}

void loop() {

// Move the servo to position 0 degrees servo.write(0);

delay(1000); // Wait for 1 second

// Move the servo to position 90 degrees servo.write(90);

delay(1000); // Wait for 1 second

// Move the servo to position 180 degrees servo.write(180);

delay(1000); // Wait for 1 second

}

Procedure

Install the Arduino ide software to your computer

Go the preference page paste the URL for installing esp 8266 driver by using Arduino ide

Check the ports is available or instal CH 340 driver from chrome and install that for port

A **servo motor** is a type of motor that can rotate with great precision. Normally this type of motor consists of a control circuit that provides feedback on the current position of the motor shaft, this feedback allows the servo motors to rotate with great precision.

If you want to rotate an object at some specific angles or distance, then you use a servo motor. It is just made up of a simple motor which runs through a **servo mechanism**.

If motor is powered by a DC power supply then it is called DC servo motor, and if it is AC-powered motor then it is called AC servo motor. For this tutorial, we will be discussing only about the **DC servo motor working**.

Apart from these major classifications, there are many other types of servo motors based on the type of gear arrangement and operating characteristics.

A servo motor usually comes with a gear arrangement that allows us to get a very high torque servo motor in small and lightweight packages. Due to these features, they are being used in many applications like toy car, RC helicopters and planes, Robotics, etc.

A servo consists of a Motor (DC or AC), a potentiometer, gear assembly, and a controlling circuit. First of all, we use gear assembly to reduce RPM and to increase torque of the motor. Say at initial position of servo motor shaft, the position of the potentiometer knob is such that there is no electrical signal generated at the output port of the potentiometer.

Result

If you want to rotate an object at some specific angles or distance, then

you use a servo motor. It is just made up of a simple motor which runs through a **servo mechanism**.

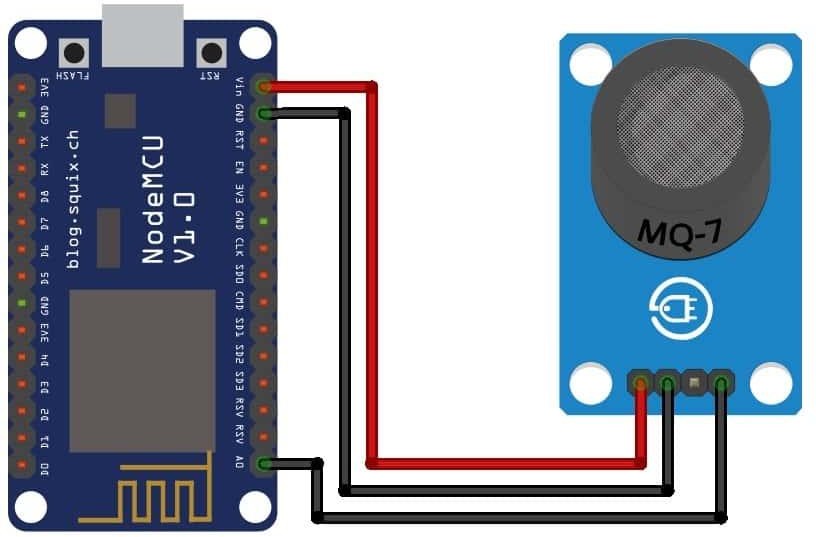
GAS SENSOR

AIM

To create the block diagram and program for GAS SENSOR by using Industrial IOT.

Materials Required:

* Esp 8266
* Jumber wire
* GAS SENSOR
* Power source Software required :
* Arduino IDE Block diagram



# Procedure

Gas sensors use physical or chemical reactions to convert the concentration of various gases into electrical signals, and output values after calculation.

Widely used to detect toxic and harmful gases and natural gas leaks.

The gas sensor is a device used to monitor the presence or level of gas in a stationary environment. Commonly used in coal mines, petroleum, chemical, municipal, medical, transportation, family, and so on. Gas sensors can measure the presence and concentration of combustible, flammable, toxic gases, or oxygen consumption.

**According to different usage methods**: it can be divided into [portable gas](https://www.renkeer.com/product/4-gas-detector/) [detectors](https://www.renkeer.com/product/4-gas-detector/) and [fixed gas detectors](https://www.renkeer.com/product/fixed-gas-detector/).

**According to different sampling methods**: it can be divided into diffusion sensors (the sensor is directly installed in the measured environment, and the measurement gas is in direct contact with the detection unit through natural diffusion), pumping sensors (through the suction pump, etc., the gas is sucked into the detection unit.

According to whether the gas needs to be diluted, it is divided into complete inhalation and diluted inhalation).

**According to different functions**: it can be divided into a [single gas](https://www.renkeer.com/product/single-gas-monitor/) [sensor](https://www.renkeer.com/product/single-gas-monitor/) (only one gas can be detected) and a [composite gas sensor](https://www.renkeer.com/product/air-quality-monitor/) (can detect multiple gases at the same time).

**According to the different detection principles**: it can be divided into semiconductor gas sensor, electrochemical gas sensor, NDIR gas sensor, catalytic gas sensor, thermal conductivity gas sensor, magnetic gas sensor

A semiconductor gas sensor is a device that uses a semiconductor element

as a measuring unit.

Its working principle is that the gas undergoes a redox reaction on the semiconductor, which causes the resistance value to change.

Electrochemical sensors react with the measured gas and generate an electrical signal proportional to the gas concentration. Most electrochemical gas sensors are current sensors, producing a current that is linearly proportional to the gas concentration.

The electrochemical gas sensors working principle: the gas passes through a diaphragm that prevents condensation, and this diaphragm also has a dust-proof effect.

Program

#define GAS\_SENSOR\_PIN A0 // Define the analog pin connected to the gas sensor

void setup() {

Serial.begin(115200); // Initialize serial communication

}

void loop() {

int gasValue = analogRead(GAS\_SENSOR\_PIN); // Read the analog value from the gas sensor

// Convert the analog value to voltage

float voltage = gasValue \* (5.0 / 1023.0); // Assuming ESP8266 operates at 5V Serial.print("Gas Value: ");

Serial.print(gasValue); Serial.print(", Voltage: "); Serial.println(voltage);

// You can add logic here to detect the presence of gas based on the sensor reading

// For example:

// if (voltage > threshold) {

// Serial.println("Gas detected!");

// // Take appropriate action

// }

delay(1000); // Add a small delay before reading again

}

Result

Gas sensors use physical or chemical reactions to convert the concentration

of various gases into electrical signals, and output values after calculation.

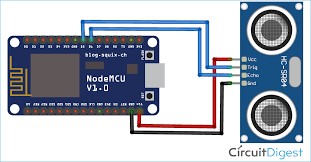
ULTRASONIC SENSOR

AIM

To create the block diagram and program for ultrasonic sensor by using Industrial IOT.

Materials Required:

* Esp 8266
* Jumber wire
* Ultrasonic sensor
* Power source Software required :
* Arduino IDE Block diagram



Procedure

Ultrasonic sensors emit short, high-frequency sound pulses at regular intervals. These propagate in the air at the velocity of sound.

If they strike an object, then they reflected back as an echo signal to the sensor, which itself computes the distance to the target based on the time-span between emitting the signal and receiving the echo.

Ultrasonic [sensors](https://robu.in/product-category/sensor/ultrasonic-sensor/) are excellent at suppressing background interference. Virtually all materials which reflect sound can be detected, regardless of their colour. Even transparent materials or thin foils represent no problem for an ultrasonic sensor.

Microsonic ultrasonic sensors are suitable for target distances from 20 mm to 10 m and as they measure the time of flight they can ascertain a measurement with pinpoint accuracy.

Some of our sensors can even resolve the signal to an accuracy of 0.025 mm. Ultrasonic sensors can see through dust-laden air and ink mists. Even thin deposits on the sensor membrane do not impair its function.

First, need to transmit trigger pulse of at least 10 us to the HC-SR04 Trig

Pin.

Then the HC-SR04 automatically sends Eight 40 kHz sound wave and wait for rising edge output at Echo pin.

When the rising edge capture occurs at Echo pin, start the Timer and wait for a falling edge on Echo pin.

As soon as the falling edge captures at the Echo pin, read the count of the Timer. This time count is the time required by the sensor to detect an object and return back from an object.

Program

#define TRIGGER\_PIN D1 // Define the GPIO pin connected to the trigger pin of the ultrasonic sensor

#define ECHO\_PIN D2 // Define the GPIO pin connected to the echo pin of the ultrasonic sensor

void setup() {

Serial.begin(115200); // Initialize serial communication pinMode(TRIGGER\_PIN, OUTPUT); // Set trigger pin as output pinMode(ECHO\_PIN, INPUT); // Set echo pin as input

}

void loop() {

// Send a 10us pulse to trigger the ultrasonic sensor digitalWrite(TRIGGER\_PIN, LOW); delayMicroseconds(2); digitalWrite(TRIGGER\_PIN, HIGH); delayMicroseconds(10); digitalWrite(TRIGGER\_PIN, LOW);

// Read the time taken for the pulse to return long duration = pulseIn(ECHO\_PIN, HIGH);

// Calculate distance in cm

float distance\_cm = duration \* 0.034 / 2; Serial.print("Distance: ");

Serial.print(distance\_cm); Serial.println(" cm");

delay(1000); // Wait for a short time before next reading

}

Result

Ultrasonic sensors emit short, high-frequency sound pulses at regular intervals. These propagate in the air at the velocity of sound.

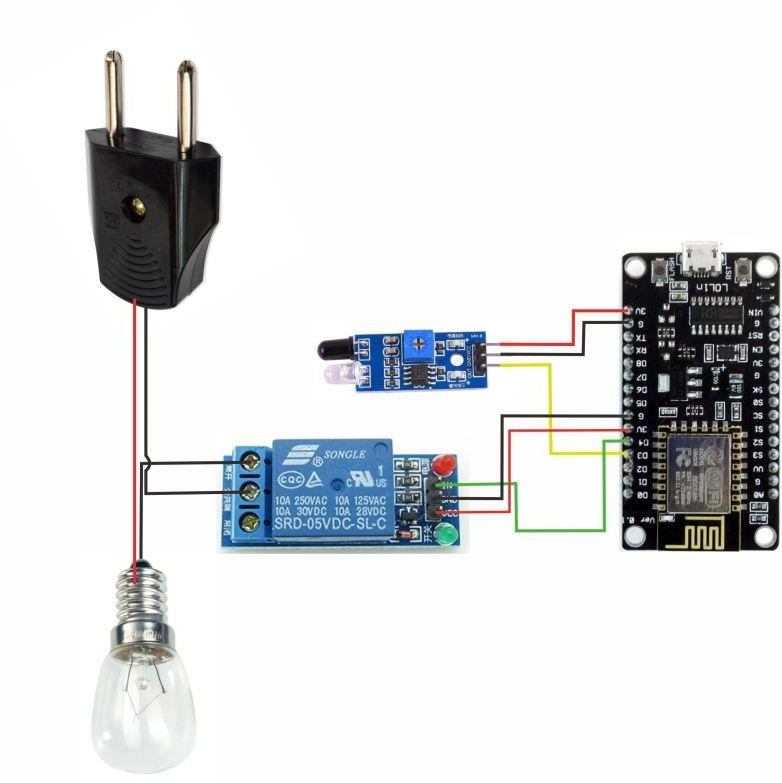
Relay while operating the servo motor or IR Sensor

AIM

To create the block diagram and program for GAS SENSOR by using Industrial IOT.

Materials Required:

* Esp 8266
* Jumber wire
* GAS SENSOR
* Power source Software required :
* Arduino IDE Block diagram



Program

#define IR\_PIN D1 // Define the GPIO pin connected to the IR sensor #define RELAY\_PIN D2 // Define the GPIO pin connected to the relay module

bool relayState = false; // Variable to store the state of the relay void setup() {

Serial.begin(115200); // Initialize serial communication

pinMode(IR\_PIN, INPUT); // Set IR pin as input pinMode(RELAY\_PIN, OUTPUT); // Set relay pin as output digitalWrite(RELAY\_PIN, LOW); // Initially turn off the relay

}

void loop() {

int irValue = digitalRead(IR\_PIN); // Read the value of IR pin

// If IR sensor detects an object, toggle the relay state if (irValue == HIGH) {

relayState = !relayState; // Toggle relay state digitalWrite(RELAY\_PIN, relayState ? HIGH : LOW); // Activate or

deactivate relay based on state

Serial.println(relayState ? "Relay Activated!" : "Relay Deactivated!"); delay(1000); // Add a small delay to avoid multiple detections

}

}

Procedure

Install the Arduino ide software to your computer

Go the preference page paste the URL for installing esp 8266 driver by using Arduino ide

Check the ports is available or instal CH 340 driver from chrome and install that for port

IR sensor is an electronic device, that emits the light in order to sense some object of the surroundings.

An [IR sensor](https://robu.in/product-category/sensor/ir-and-pir-sensor/) can measure the heat of an object as well as detects the motion. Usually, in the [**infrared spectrum**](https://en.wikipedia.org/wiki/Infrared_spectroscopy), all the objects radiate some form of thermal radiation.

These types of radiations are invisible to our eyes, but infrared sensor can detect these radiations.

The emitter is simply an IR LED [(Light Emitting Diode](https://robu.in/product-category/display-boards/led/)) and the detector is simply an IR photodiode. Photodiode is sensitive to IR light of the same wavelength which is emitted by the IR LED.

When IR light falls on the photodiode, the resistances and the output voltages will change in proportion to the magnitude of the IR light received.

There are five basic elements used in a typical infrared detection system: an infrared source, a transmission medium, optical component, infrared detectors or receivers and signal processing.

Infrared lasers and Infrared LED’s of specific wavelength used as infrared sources.

The three main types of media used for infrared transmission are vacuum, atmosphere and optical fibers. Optical components are used to focus the infrared radiation or to limit the spectral response.

Result

IR sensor is an electronic device, that emits the light in order to sense

some object of the surroundings. An [IR sensor](https://robu.in/product-category/sensor/ir-and-pir-sensor/) can measure the heat of an object as well as detects the motion