**ARDUINO:-**

Arduino is an open source programmable circuit board that can be integrated into a wide variety of makerspace projects both simple and complex. This board contains a microcontroller which is able to be programmed to sense and control objects in the physical world. By responding to sensors and inputs, the Arduino is able to interact with a large array of outputs such as LEDs, motors and displays. Because of it’s flexibility and low cost, Arduino has become a very popular choice for makers and makerspaces looking to create interactive hardware projects. Arduino was introduced back in 2005 in Italy by Massimo Banzi as a way for nonengineers to have access to a low cost, simple tool for creating hardware projects. Since the board is open-source, it is released under a Creative Commons license which allows anyone to produce their own board. If you search the web, you will find there are hundreds of Arduino compatible clones and variations available but the only official boards have Arduino in it’s name.

**DIGITAL PINS:-**

Digital is a way of representing voltage in 1 bit: either 0 or 1. Digital pins on the Arduino are **pins designed to be configured as inputs or outputs according to the needs of the user**. Digital pins are either on or off. When ON they are in a HIGH voltage state of 5V and when OFF they are in a LOW voltage state of 0V.

**ANALOG PINS:-**

A/D converter  
  
While the main function of the analog pins for most Arduino users is to read analog sensors, the analog pins also have all the functionality of general purpose input/output (GPIO) pins (the same as digital pins 0 - 13).

**RASBERRY PI:-**

The Raspberry Pi is a low cost, **credit-card sized computer** that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python. It’s capable of doing everything you’d expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and playing games.

**JUMPER WIRES:-**

A jumper wire is **an electric wire that connects remote electric circuits used for printed circuit boards**. By attaching a jumper wire on the circuit, it can be short-circuited and short-cut (jump) to the electric circuit.

Jumper wires typically come in three versions: male-to-male, male-to-female and female-to-female. The difference between each is in the end point of the wire. Male ends have a pin protruding and can plug into things, while female ends do not and are used to plug things into. Male-to-male jumper wires are the most common and what you likely will use most often. When connecting two ports on a breadboard, a male-to-male wire is what you’ll need.

**ULTRA SONIC SENSOR:-**

An ultrasonic sensor is an electronic device that measures the distance of a target object by emitting ultrasonic sound waves, and converts the reflected sound into an electrical signal. Ultrasonic waves travel faster than the speed of audible sound (i.e. the sound that humans can hear).

Ultrasonic sensors are used primarily as proximity sensor. They can be found in automobile self-parking technology and anti-collision safety systems. Ultrasonic sensors are also used in robotic obstacle detection systems, as well as manufacturing technology.

**TEMPERATURE SENSOR:-**

A temperature sensor is **a** device used to measure temperature. This can be air temperature, liquid temperature or the temperature of solid matter. There are different types of temperature sensors available and they each use different technologies and principles to take the temperature measurement.

**BREADBOARD:-**

A breadboard consists of plastic block holding a matrix of electrical sockets of a size suitable for gripping thin connecting wire, component wires or the pins of transistors and integrated circuits **(ICs)**. The sockets are connected inside the board, usually in rows of five sockets.

**PIR SENSOR:-**

A device used to detect motion by receiving infrared radiation. When a person walks past the sensor, it detects a rapid change of infrared energy and sends a signal. PIR sensors are used for applications such as automatically turning on lights when someone enters a room or causing a video camera to begin operating.

**BUZZER:-**

An audio signaling device like a beeper or buzzer may be electromechanical or piezoelectric or mechanical type. The main function of this is to convert the signal from audio to sound. Generally, it is powered through DC voltage and used in timers, alarm devices, printers, alarms, computers, etc. Based on the various designs, it can generate different sounds like alarm, music, bell & siren.

**RESISTOR:-**

A resistor is an electrical component that limits or regulates the flow of electrical current in an electronic circuit. Resistors can also be used to provide a specific voltage for an active device such as a transistor.

**LED:-**

A light-emitting diode (LED) is a semiconductor light source that emits light when current flows through it. Electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons.

**SERVO MOTOR:-**

A servo motor is a rotary actuator that allows for precise control of angular position. It consists of a motor coupled to a sensor for position feedback. It also requires a servo drive to complete the system. The drive uses the feedback sensor to precisely control the rotary position of the motor.

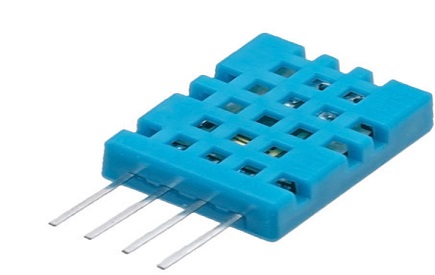
TEMPERATURE SENSOR:- (DHT11)

DHT11 is a low-cost digital sensor for sensing temperature and humidity.  This sensor can be easily interfaced with any micro-controller such as Arduino, Raspberry Pi etc… to measure humidity and temperature instantaneously.

DHT11 sensor consists of a capacitive humidity sensing element and a thermistor for sensing temperature.  The humidity sensing capacitor has two electrodes with a moisture holding substrate as a dielectric between them. Change in the capacitance value occurs with the change in humidity levels. The IC measure, process this changed resistance values and change them into digital form.

For measuring temperature this sensor uses a Negative Temperature coefficient thermistor, which causes a decrease in its resistance value with increase in temperature. To get larger resistance value even for the smallest change in temperature, this sensor is usually made up of semiconductor ceramics or polymers.

The temperature range of DHT11 is from 0 to 50 degree Celsius with a 2-degree accuracy. Humidity range of this sensor is from 20 to 80% with 5% accuracy. The sampling rate of this sensor is 1Hz .i.e. it gives one reading for every second.  DHT11 is small in size with operating voltage from 3 to 5 volts. The maximum current used while measuring is 2.5mA.

DHT11 Sensor

DHT11 sensor has four pins- VCC, GND, Data Pin and a not connected pin. A pull-up resistor of 5k to 10k ohms is provided for communication between sensor and micro-controller.

PIN DIAGRAM:-



**CODE:-**

#include <dht.h>

#define dht\_apin A0 // Analog Pin sensor is connected to

dht DHT;

void setup(){

Serial.begin(9600);

delay(500);//Delay to let system boot

Serial.println("DHT11 Humidity & temperature Sensor\n\n");

delay(1000);//Wait before accessing Sensor

}//end "setup()"

void loop(){

//Start of Program

DHT.read11(dht\_apin);

Serial.print("Current humidity = ");

Serial.print(DHT.humidity);

Serial.print("% ");

Serial.print("temperature = ");

Serial.print(DHT.temperature);

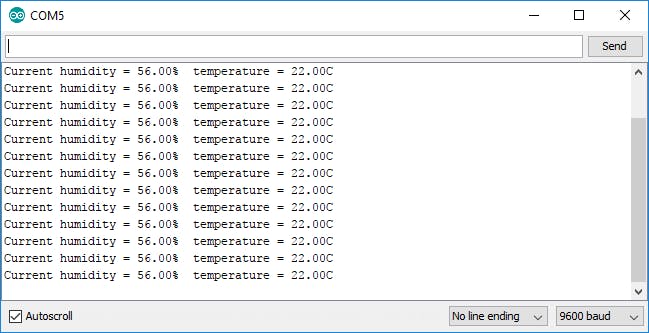
Serial.println("C ");

delay(5000);//Wait 5 seconds before accessing sensor again.

//Fastest should be once every two seconds.

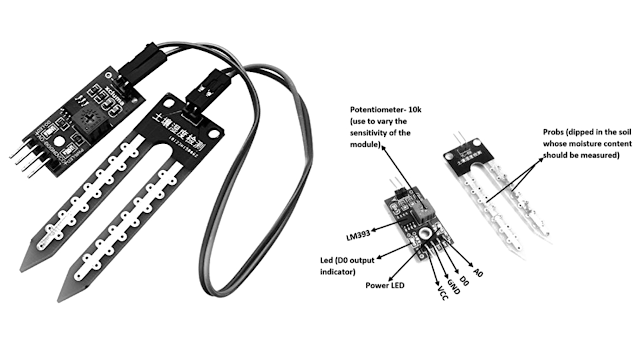
}// end loop(

OUTPUT:-



**SOIL MOISTURE SENSOR:-**

The soil moisture sensor is one kind of sensor used to gauge the volumetric content of water within the soil. As the straight gravimetric dimension of soil moisture needs eliminating, drying, as well as sample weighting. These sensors measure the volumetric water content not directly with the help of some other rules of soil like dielectric constant, electrical resistance, otherwise interaction with neutrons, and replacement of the moisture content.



SOIL MOISTURE SENSOR WORKING:-

The Soil Moisture Sensor uses capacitance to measure dielectric permittivity of the surrounding medium. In soil, dielectric permittivity is a function of the water content. The sensor creates a voltage proportional to the dielectric permittivity, and therefore the water content of the soil.

The sensor averages the water content over the entire length of the sensor. There is a 2 cm zone of influence with respect to the flat surface of the sensor, but it has little or no sensitivity at the extreme edges. The figure above shows the electromagnetic field lines along a cross-section of the sensor, illustrating the 2 cm zone of influence.

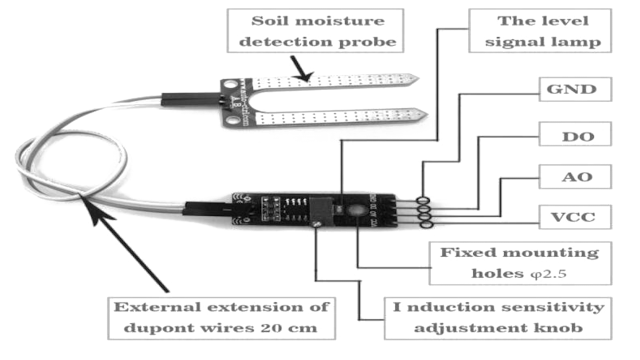
**SOIL MOISTURE SENSOR WORKING PRINCIPLE:-**

This sensor mainly utilizes capacitance to gauge the water content of the soil (dielectric permittivity). The working of this sensor can be done by inserting this sensor into the earth and the status of the water content in the soil can be reported in the form of a percent.

This sensor makes it perfect to execute experiments within science courses like environmental science, agricultural science, biology, soil science, botany, and horticulture.

* The required voltage for working is 5V
* The required current for working is <20mA
* Type of interface is analog
* The required working temperature of this sensor is 10°C~30°C

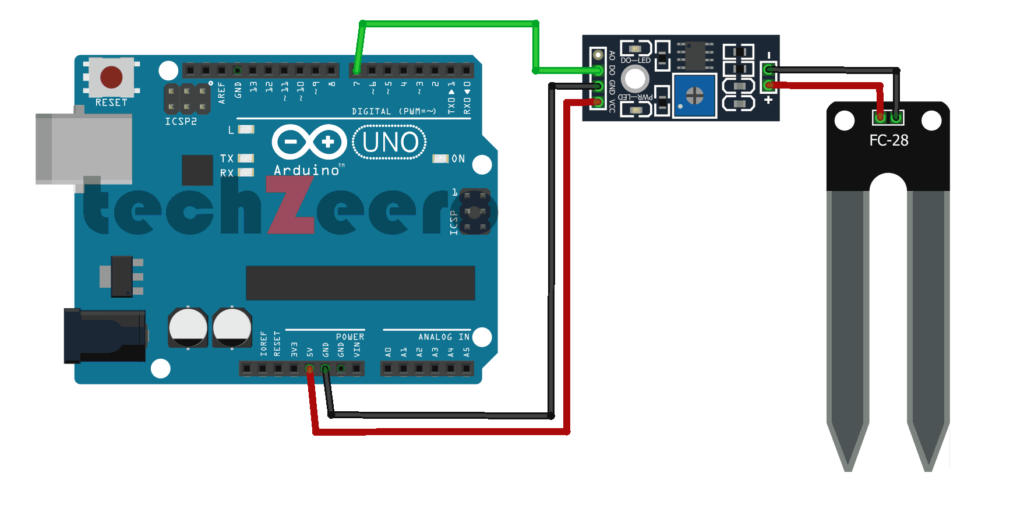
**PIN DIAGRAM:-**



**CODE:-**

|  |
| --- |
| \*/  Soil Moisture with Arduino - Analog Output |
| For more details, visit: https://techzeero.com/arduino-tutorials/soil-moisture-sensor-arduino/ |
| \*/ |
|  |
| int sensorPin = A0; |
| int outputValue ; |
|  |
| void setup() |
| { |
| Serial.begin(9600); |
| Serial.println("Reading Data From the Sensor ..."); |
| delay(2000); |
| } |
|  |
| void loop() |
| { |
| outputValue= analogRead(sensorPin); |
| outputValue = map(outputValue,550,0,0,100); |
|  |
| Serial.print("Moisture Value : "); |
| Serial.print(outputValue); |
| Serial.println("%"); |
| delay(1000); |
| } |

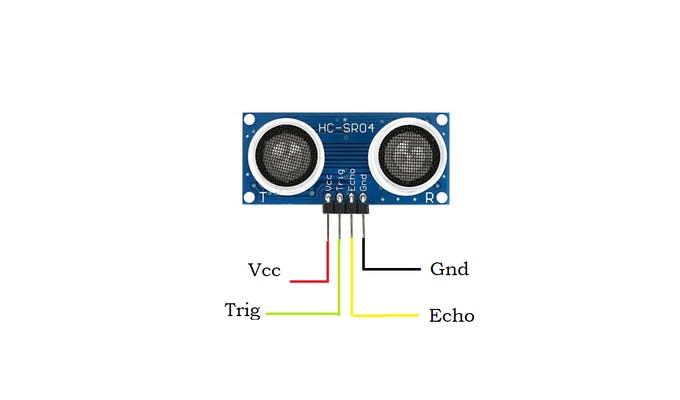
**ANALOG OUTPUT:-**



**ULTRA SONIC SENSOR:-**

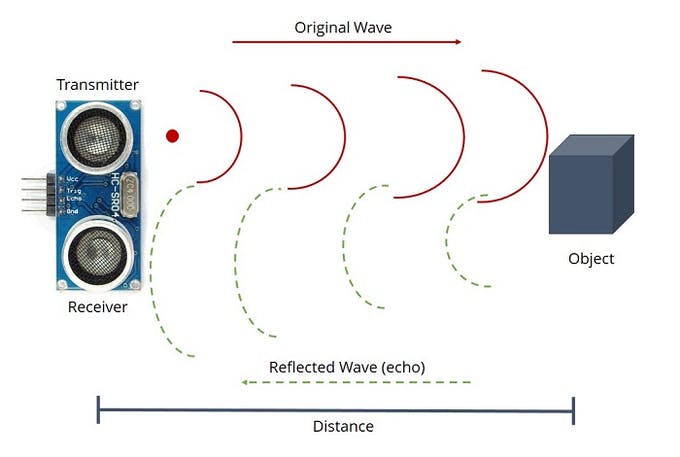
Ultrasonic Sensor HC-SR04 is a sensor that can measure distance. It emits an ultrasound at 40 000 Hz (40kHz) which travels through the air and if there is an object or obstacle on its path It will bounce back to the module. Considering the travel time and the speed of the sound you can calculate the distance.

The configuration pin of HC-SR04 is VCC (1), TRIG (2), ECHO (3), and GND (4). The supply voltage of VCC is +5V and you can attach TRIG and ECHO pin to any Digital I/O in your Arduino Board.



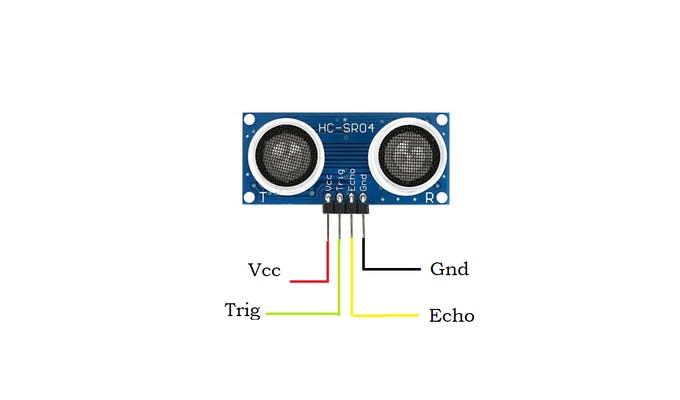
**WORKING MECHANISM:-**

An ultrasonic sensor module uses a transducer to send and receive ultrasonic pulses. The working principle of this module is simple. It sends an ultrasonic pulse out of trigger pin at 40kHz which travels through the air and if there is an obstacle or object, it will bounce back to the sensor at echo pin. By calculating the time travelled by wave and the speed of sound, the distance of the object is calculated.

[](javascript:openLightBox('cc86904286',%200);)

Ultrasonic sensor module has four pins namely Gnd, Vcc, Echo and Trigger. Gnd is considered as the negative pin and it is connected to the ground of the system. Vcc powers the sensor. It typically requires 3.3V. Trig (Trigger) pin is used to trigger the ultrasonic sound pulses. Echo pin produces a pulse when the reflected signal is received.

PIN DIAGRAM:-



CODE:-

#define echoPin 12 //connect echo pin of ultrasonic sensor to D12 of Arduino

#define trigPin 10 //connect trigger pin of ultrasonic sensor to D10 of Arduino

long duration; // declare variables to hold duration and distance

int distance;

void setup() //setup() is used for initialization

{

Serial.begin(9600); //set the baud rate of serial communication to 9600

pinMode(trigPin,OUTPUT); //set trigPin as output pin of Arduino

pinMode(echoPin,INPUT); //set echoPin as output pin of Arduino

}

void loop(){

digitalWrite(trigPin,LOW); //generate square wave at trigger pin

delayMicroseconds(2);

digitalWrite(trigPin,HIGH);

delayMicroseconds(10);

digitalWrite(trigPin,LOW);

duration=pulseIn(echoPin,HIGH);//calculation of distance of obstacle

distance=(duration\*0.034/2);

Serial.print("Distance : ");

Serial.print(distance);

Serial.println(" cm ");

delay(1000);

}

**OUTPUT:-**



**IR SENSOR:-**

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.



There are two types of IR sensors are available and they are,

* Active Infrared Sensor
* Passive Infrared Sensor

Active infrared sensors consist of two elements: infrared source and infrared detector. Infrared sources include the LED or infrared [**laser diode**](https://robu.in/product-category/electronic-module/laser-module/). Infrared detectors include photodiodes or phototransistors. The energy emitted by the infrared source is reflected by an object and falls on the infrared detector.

Passive infrared [**sensors**](https://robu.in/product-category/sensor/) are basically Infrared detectors. Passive infrared sensors do not use any infrared source and detector. They are of two types: quantum and thermal. Thermal infrared sensors use infrared energy as the source of heat. [**Thermocouples**](https://robu.in/product/max6675-thermocouple-sensor-module/), pyroelectric detectors and bolometers are the common types of thermal infrared detectors. Quantum type infrared sensors offer higher detection performance. It is faster than thermal type infrared detectors. The photo sensitivity of quantum type detectors is wavelength dependent.

WORKING MECHANISM:-

There are different types of infrared transmitters depending on their wavelengths, output power and response time. An IR sensor consists of an IR LED and an IR Photodiode, together they are called as PhotoCoupler or OptoCoupler.

**IR Transmitter or IR LED**

Infrared Transmitter is a light emitting diode (LED) which emits infrared radiations called as IR LED’s. Even though an IR LED looks like a normal LED, the radiation emitted by it is invisible to the human eye.

The picture of an Infrared LED is shown below.

[](https://robu.in/wp-content/uploads/2020/05/51fibl-5xL._SX342_.jpg)

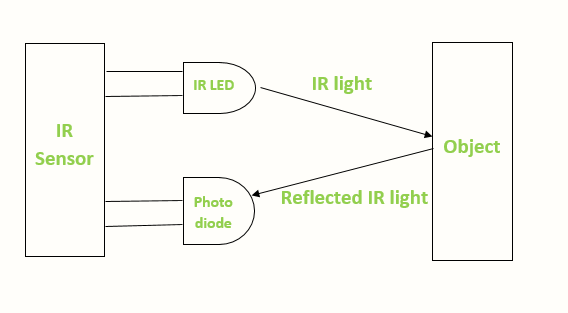
**IR Receiver or Photodiode**

Infrared receivers or infrared sensors detect the radiation from an IR transmitter. IR receivers come in the form of photodiodes and phototransistors. Infrared Photodiodes are different from normal photo diodes as they detect only infrared radiation. Below image shows the picture of an IR receiver or a photodiode,

[](https://robu.in/wp-content/uploads/2020/05/SN-IR-R-0-1-1-800x800-1.jpg)

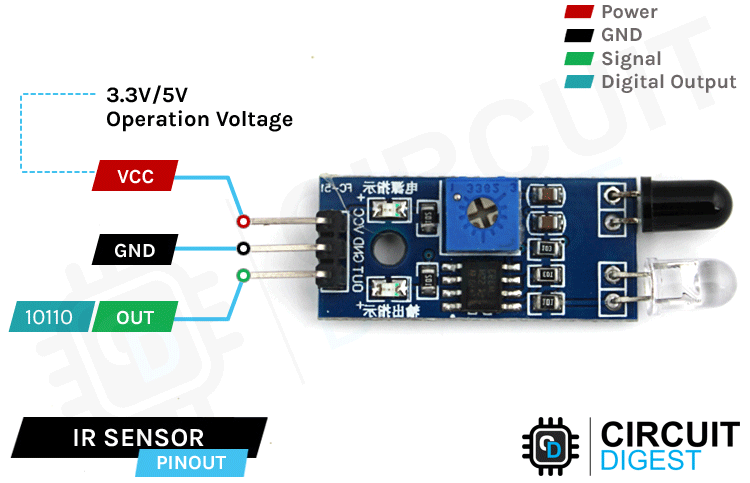
Different types of IR receivers exist based on the wavelength, voltage, package, etc. When used in an infrared transmitter – receiver combination, the wavelength of the receiver should match with that of the transmitter.

The emitter is an IR LED and the detector is an IR photodiode. The IR photodiode is sensitive to the IR light emitted by an IR LED. The photo-diode’s resistance and output voltage change in proportion to the IR light received. This is the underlying working principle of the IR sensor.

[](https://robu.in/wp-content/uploads/2020/05/IR-sensor-Working.png)

When the IR transmitter emits radiation, it reaches the object and some of the radiation reflects back to the IR receiver. Based on the intensity of the reception by the IR receiver, the output of the [**sensor**](https://robu.in/product-category/sensor/) defines.

PIN DIAGRAM:-



CODE:-

int IRSensor = 2; // connect ir sensor to arduino pin 2  
int LED = 13; // connect Led to arduino pin 13

void setup()  
{  
pinMode (IRSensor, INPUT); // sensor pin INPUT  
pinMode (LED, OUTPUT); // Led pin OUTPUT  
}

void loop()  
{  
int statusSensor = digitalRead (IRSensor);

if (statusSensor == 1)  
{  
digitalWrite(LED, LOW); // LED LOW  
}

else  
{  
digitalWrite(LED, HIGH); // LED High  
}

}

OUTPUT:-

