HOME AUTOMATION USING IOT

*Summer Internship Report Submitted in partial fulfillment*

*of the requirement for under graduate degree of*

**Bachelor of Technology**

In

**Electrical and Electronics Engineering**

**/Electronics and Communication Engineering**

By

**V.SAI SHIVA RAM REDDY**

**221710401064**

*Under the Guidance of*

**Mr. CHANDRU**

Assistant Professor



Department of Electrical, Electronics and Communication Engineering

GITAM School of Technology

GITAM Deemed to be University

Hyderabad Campus -502329

August 2020

**DECLARATION**

I submit this Summer Internship entitled "**HOME AUTOMATION USING INTENET OF THINGS"** to GITAM School of Technology, GITAM Deemed to be University, Hyderabad campus in partial fulfillment of the requirements for the award of the degree of “Bachelor of Technology” in **“Electronics and Communication Engineering / Electrical and Electronics Engineering ”.**

I declare that it was carried out independently by me under the guidance of CHNADRU ASST PROFESSORGITAM School of Technology, GITAM Deemed to be University, Hyderabad, India.

The results embodied in this report have not been submitted to any other University or Institute for the award of any degree or diploma.

Place: HYDERABAD V.SAI SHIVA RAM REDD

Date: 15-09-2020 221710401064



GITAM Deemed to be University

Hyderabad Campus-502329, India

: 15-09-2020

**CERTIFICATE**

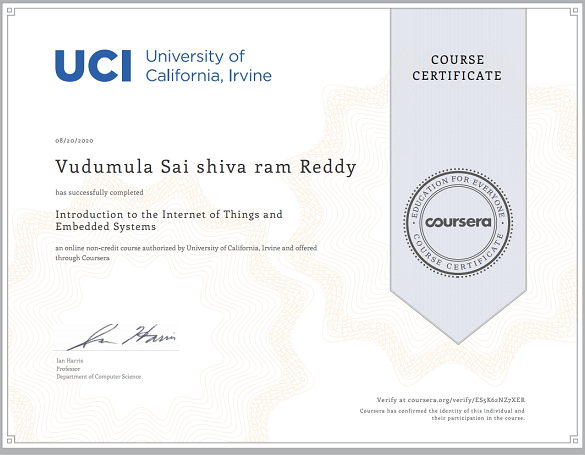
This is to certify that the Summer Internship Report entitled **“ HOME AUTOMATION USING INTERNET OF THINGS”** is being submitted by V.SAI SHIVA RAM REDDY(221710401064) in partial fulfillment of the requirement for the award of **Bachelor of Technology** **in “Electronics and Communication Engineering / Electrical an~~d~~ Electronics Engineering”.** at GITAM University, Hyderabad.

It is faithful record work carried out by her at the Electrical, **Electronics & Communication Engineering Department**, GITAM School of Technology, GITAM Deemed to be University ,Hyderabad Campus under my guidance and supervision.

**Mr. CHNADRU Dr.K.MANJUNATHCHARI**

Assistant Professor Professor and HOD

Department of EECE Department of EECE

****

**ACKNOWLEDGMENT**

Apart from my effort, the success of this internship largely depends on the encouragement and guidance of many others. I take this opportunity to express my gratitude to the people who have helped me in the successful competition of this internship.

I would like to thank respected **Dr. N. Siva Prasad**, Pro Vice Chancellor, GITAM Hyderabad and **Dr. N. Seetharamaiah,** Principal, GITAM Hyderabad.

I would like to thank respected **Dr. K. MANJUNATHCHARI.**  Head of the Department of Computer Science Engineering for giving me such a wonderful opportunity to expand my knowledge for my own branch and giving me guidelines to present a internship report. It helped me a lot to realize of what we study for.

I would like to thank the respected faculty **R.CHANDRU.** who helped me to make this internship a successful accomplishment.

I would also like to thank my friends who helped me to make my work more organized and well-stacked till the end.

V.SAI SHIVA RAM REDDY

221710401064

**Abstract:**

Home automation has become more and more popular in recent years. It aims at helping people manage the home appliances freely and build an autonomous environment in home. The aim of this project is the home automation with full security and controlling the home appliances using wireless communication as Wi-Fi. We design this smart home system with the implementation of related software and hardware The temperature and humidity of the each room is monitored and maintained at room temperature using temperature and humidity sensors which activates the exhaust fan to maintain the temperature. The water level sensor is used to fill the overhead water tank without wasting the water. For these control purposes Arduino mega 2560 and ESP8266 is used because the arduino has the advantages of ease understandability and easily modifiable. The arduino board is specially designed circuit board for programming and prototyping with ATMEL microcontroller. The microcontroller used in this arduino is ATmega 328 which is in-built in arduino board and the coding are done in java script.

**LIST OF CONTENTS**

1. Introduction
2. Block Diagram
3. Arduino UNO
4. Detectors
5. Bluetooth &Wi-fi Modules
6. Components
7. DC Motors
8. Source Code

IOT BASED HOME AUTOMATION

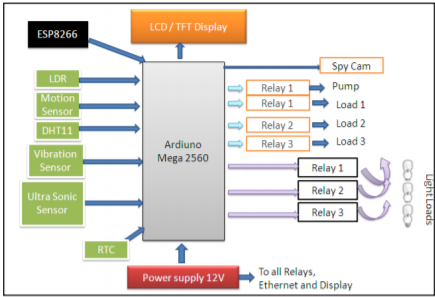
**Abstract:**

Home automation has become more and more popular in recent years. It aims at helping people manage the home appliances freely and build an autonomous environment in home. The aim of this project is the home automation with full security and controlling the home appliances using wireless communication as Wi-Fi. We design this smart home system with the implementation of related software and hardware The temperature and humidity of the each room is monitored and maintained at room temperature using temperature and humidity sensors which activates the exhaust fan to maintain the temperature. The water level sensor is used to fill the overhead water tank without wasting the water. For these control purposes Arduino mega 2560 and ESP8266 is used because the arduino has the advantages of ease understandability and easily modifiable. The arduino board is specially designed circuit board for programming and prototyping with ATMEL microcontroller. The microcontroller used in this arduino is ATmega 328 which is in-built in arduino board and the coding are done in java script.

**1.Introduction:**

Home automation is providing home safety for dwellers. It automatically turn lights on in closets, stairways, and other dark places. Thus accidentally tripping or running into thing is decreased. Everywhere environmental issues are raised before introducing any technology. In this regard home automation provides a better solution. Devices included in home automation consume less power. Besides, it saves energy. Thus home automation technology is so far environmentally suitable. Moreover, the technology keeps mind in peace. In most cases, guardians face and always they keep tensioning for the safety of their children staying in home. In home automation system internet access is used to control from far away. For years, internet is used only for surfing pages, searching information and downloading software and other things. Advancement of technology is forcing to make interaction internet with machineries and devices. In home automation system comfort and security of houses have been enhanced. Besides, people are concerning over costs. In offices, a division of people are employed only to make supervision of some manual means typed work. Home automation is replacing those arrangements. For this, cost is highly reduced. Besides, for manual labor engaged to control appliances waste energy in cases. It is seen that appliances continue to run though people are not present in their respective places. For this energy cannot stop consuming. If this happens for a long time then there have possibility to misuse energy in a huge amount. To overcome this obstacle home automation is encouraged to apply. Home automation does that challenging work. That‘s why; home automation is presented as energy efficient. In recent years home automation is gaining much popularity. The trend is also in favor of using home automation technology. If we look around residences, malls, offices, use of home automation systems will draw attention.

**2. Block Diagram:**

****

The proposed block diagram is shown in figure. The main block of our project is the arduino module which falls next in line. There are relays to serve the purpose of on and off. The power supply provided for arduino is 5V.It is given through an adapter. The power supply given to the relays is 12V and it is given from a step down transformer. The relays used act as Main Switches. The relays are programmed to operate without delay. The signals for the relay are given from arduino board. The actuators are connected to output of the relay. The sensors are directly interfaced to the arduino and they are used for detecting the parameters like temperature, Light intensity, Humidity, sound and obstacle detection.

**3. Arduino UNO:**

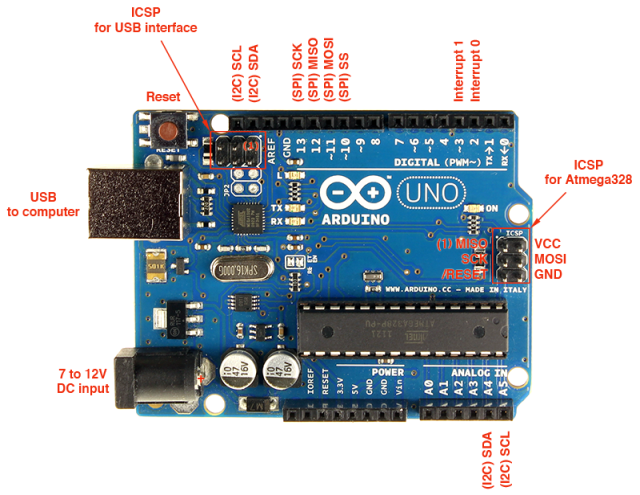


Fig 3.3: ARDUINO UNO R3

Arduino is a single-board microcontroller meant to make the application more accessible which are interactive objects and its surroundings.

Arduino board designs use a variety of [microprocessors](https://en.wikipedia.org/wiki/Microprocessor) and controllers. The boards are equipped with sets of digital and analog [input/output](https://en.wikipedia.org/wiki/Input/output) (I/O) pins that may be interfaced to various expansion boards ('shields') or [breadboards](https://en.wikipedia.org/wiki/Breadboards) (For prototyping) and other circuits. The boards feature serial communications interfaces, including [Universal Serial Bus](https://en.wikipedia.org/wiki/Universal_Serial_Bus) (USB) on some models, which are also used for loading programs from personal computers. The microcontrollers can be programmed using [C](https://en.wikipedia.org/wiki/C_(programming_language)) and [C++](https://en.wikipedia.org/wiki/C%2B%2B) [programming languages](https://en.wikipedia.org/wiki/Programming_language). In addition to using traditional [compiler](https://en.wikipedia.org/wiki/Compiler) [toolchains](https://en.wikipedia.org/wiki/Toolchains), the Arduino project provides an [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) based on the [Processing](https://en.wikipedia.org/wiki/Processing_(programming_language)) language project.

**PIN DIAGRAM:**

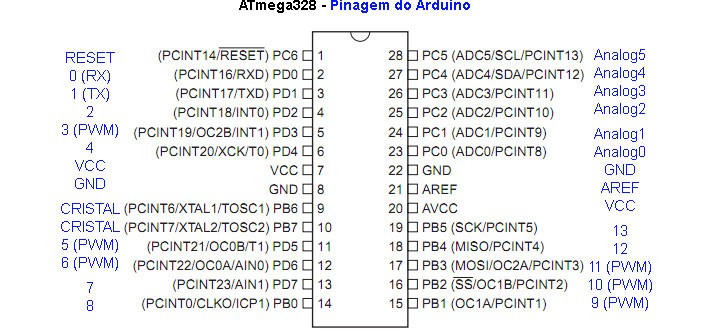


Fig-3.3.1 Pin Diagram of Arduino

**Features of the Arduino Uno Board:**

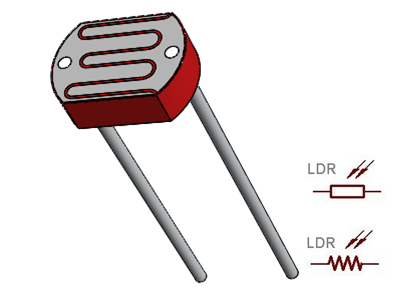
* It is an easy USB interface. This allows interface with USB as this is like a serial device.
* The chip on the board plugs straight into your USB port and supports on your computer as a virtual serial port. The benefit of this setup is that serial communication is an extremely easy protocol which is time-tested and USB makes connection with modern computers and makes it comfortable.
* It is an open source design and there is an advantage of being open source is that it has a large community of people using and troubleshooting it. This makes it easy to help in debugging projects.
* It is a 16 MHz clock which is fast enough for most applications and does not speeds up the microcontroller.
* It is very convenient to manage power inside it and it had a feature of built-in voltage regulation. This can also be powered directly off a USB port without any external power. You can connect an external power source of up to 12v and this regulates it to both 5v and 3.3v.
* 13 digital pins and 6 analog pins. This sort of pins allows you to connect hardware to your Arduino Uno board externally. These pins are used as a key for extending the computing capability of the Arduino Uno into the real world. Simply plug your electronic devices and sensors into the sockets that correspond to each of these pins and you are good to go.
* It has a 32 KB of flash memory for storing your code.
* An on-board LED is attached to digital pin 13 to make fast the debugging of code and to make the debug process easy.
* Finally, it has a button to reset the program on the chip.

**Technical Specifications**:

* Microcontroller ATmega328
* Operating Voltage 5V
* Input Voltage (recommended) 7-12V
* Input Voltage (limits) 6-20V
* Digital I/O Pins 14 (of which 6 provide PWM output)
* Analog Input Pins 6
* DC Current per I/O Pin 40 mA
* DC Current for 3.3V Pin 50 mA
* Flash Memory 32 KB of which 0.5 KB used bootloader
* SRAM 2 KB
* EEPROM 1 KB
* Clock Speed 16 MHz

4. Detectors:

i)LDR:



A **photoresistor** or **LDR** (Light Dependent Resistor), as the name suggests will change it resistance based on the light around it. That is when the resistor is placed in a dark room it will have a resistance of few Mega ohms and as we gradually impose light over the sensor its resistance will start to decrease from Mega Ohms to few Ohms.

This property helps the LDR to be used as a **Light Sensor**. It can detect the amount of light falling on it and thus can predict days and nights. So if you are looking for a sensor to sense light or to distinguish between days and nights then this sensor is the cheap and modest solution for you

### LDR Pin Description

* VCC = 3.3V to 5V DC
* GND = Ground
* DO = Digital Output
* AO = Analog Output.

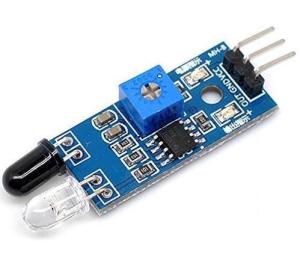
### LDR Features

* Can be used to sense Light
* Easy to use on Breadboard or Perf Board
* Easy to use with Microcontrollers or even with normal Digital/Analog IC
* Small, cheap and easily available
* Available in PG5 ,PG5-MP, PG12, PG12-MP, PG20 and PG20-MP series

### How to use a LDR sensor

As said earlier a LDR is one of the [different types of resistors](https://components101.com/articles/resistor-basics-types-and-uses), hence using it is very easy. There are many ways and different circuit in which an LDR can be used. For instance it can be used with Microcontroller Development platforms like [Arduino](https://components101.com/microcontrollers/arduino-uno), [PIC](https://components101.com/pic16f877a-pin-diagram-description-features-datasheet) or even normal Analog IC’s like Op-amps. But, here we will use a very simple circuit like a potential divider so that it can be adapted for most of the projects.

A potential Divider is a circuit which has two resistors in series. A constant voltage will be applied across the both the resistor and the output voltage will be measured from the lower resistor. In our case, the lower resistor will be a **LDR**and the constant voltage will be +5V. The set-up is shown below

ii) IR-SENSOR: 

IR Sensor or Infrared Sensor has two main parts. IR Transmitter and IR Receiver. The work of IR transmitter or Infrared transmitter is to transmit the infrared waves whereas the work of IR receiver is to receive these infrared waves. IR receiver constantly sends digital data in the form of 0 or 1 to Vout pin of the sensor.

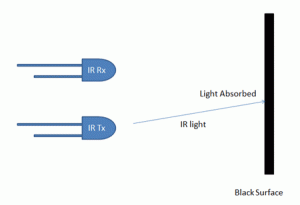
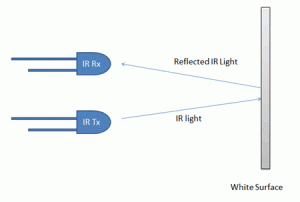
[](https://circuitdigest.com/microcontroller-projects/line-follower-robot-using-arduino)

Fig a:Working of IR Tx& Rx on White Surface Fig b:Working of IR Tx& Rx on Black Surface

Pins of IR sensor:

1)  Ground

2)  5 volt

3)  Enable

4)  Vout

iii) DHT-11 Sensor:

The **DHT11**is a commonly used **Temperature and humidity sensor.** The sensor comes with a dedicated NTC to measure temperature and an 8-bit microcontroller to output the values of temperature and humidity as serial data. The sensor is also factory calibrated and hence easy to interface with other microcontrollers.

The sensor can measure temperature from 0°C to 50°C and humidity from 20% to 90% with an accuracy of ±1°C and ±1%. So if you are looking to measure in this range then this sensor might be the right choice for you.

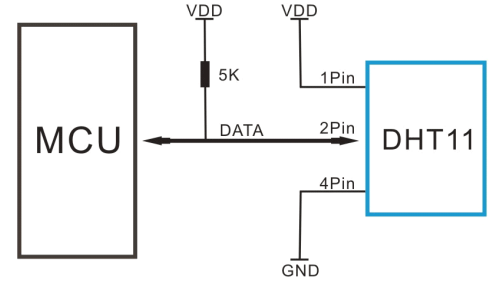
Pin Identification and Configuration:

****

|  |  |  |
| --- | --- | --- |
| **No:** | **Pin Name** | **Description** |
| **For DHT11 Sensor** | | |
| 1 | Vcc | Power supply 3.5V to 5.5V |
| 2 | Data | Outputs both Temperature and Humidity through serial Data |
| 3 | NC | No Connection and hence not used |
| 4 | Ground | Connected to the ground of the circuit |

How to use DHT11 Sensor:

The DHT11 Sensor is factory calibrated and outputs serial data and hence it is highly easy to set it up. The connection diagram for this sensor is shown below.



**Applications:**

* Measure temperature and humidity
* Local Weather station
* Automatic climate control
* Environment monitoring

iv) ULTRASONIC SENSOR:



### HC-SR04 Sensor Features:

* Operating voltage: +5V
* Theoretical  Measuring Distance: 2cm to 450cm
* Practical Measuring Distance: 2cm to 80cm
* Accuracy: 3mm
* Measuring angle covered: <15°
* Operating Current: <15mA
* Operating Frequency: 40Hz

**Ultrasonic Sensor Pin Configuration**

|  |  |  |
| --- | --- | --- |
| **Pin Number** | **Pin Name** | **Description** |
| 1 | Vcc | The Vcc pin powers the sensor, typically with +5V |
| 2 | Trigger | Trigger pin is an Input pin. This pin has to be kept  high for 10us to initialize measurement by sending US wave. |
| 3 | Echo | Echo pin is an Output pin. This pin goes high for a period of time which will be  equal to the time taken for the US wave to return back to the sensor. |
| 4 | Ground | This pin is connected to the Ground of the system. |



**How to use the HC-SR04 Ultrasonic Sensor**

**HC-SR04 distance sensor** is commonly used with both microcontroller and microprocessor platforms like Arduino, ARM, PIC, Raspberry Pie etc. The following guide is universally since it has to be followed irrespective of the type of computational device used.

  Power the Sensor using a regulated +5V through the Vcc ad Ground pins of the sensor. The current consumed by the sensor is less than 15mA and hence can be directly powered by the on board 5V pins (If available). The Trigger and the Echo pins are both I/O pins and hence they can be connected to I/O pins of the microcontroller. To start the measurement, the trigger pin has to be made high for 10uS and then turned off. This action will trigger an ultrasonic wave at frequency of 40Hz from the transmitter and the receiver will wait for the wave to return. Once the wave is returned after it getting reflected by any object the Echo pin goes high for a particular amount of time which will be equal to the time taken for the wave to return back to the sensor.

The amount of time during which the Echo pin stays high is measured by the MCU/MPU as it gives the information about the time taken for the wave to return back to the Sensor. Using this information the distance is measured as explained in the above heading.

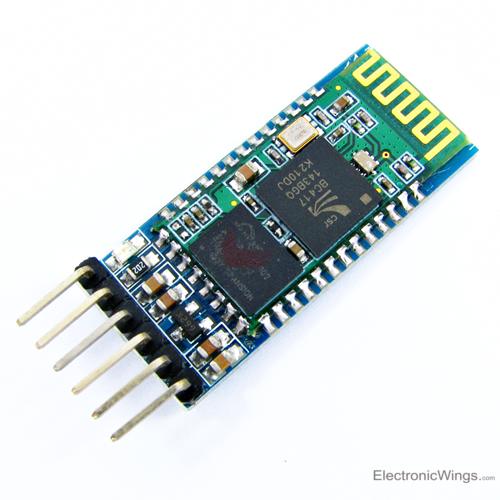
**Applications**

* Used to avoid and detect obstacles with robots like biped robot, obstacle avoider robot, path finding robot etc.
* Used to measure the distance within a wide range of 2cm to 400cm
* Can be used to map the objects surrounding the sensor by rotating it
* Depth of certain places like wells, pits etc can be measured since the waves can penetrate through water

5. **Bluetooth & Wi-Fi Modules**:

# i) Bluetooth Module HC-05:

HC-05 is a Bluetooth module which is designed for wireless communication. This module can be used in a master or slave configuration.

It has 6 pins,

1.  **Key/EN:** It is used to bring Bluetooth module in AT commands mode. If Key/EN pin is set to high, then this module will work in command mode. Otherwise by default it is in data mode. The default baud rate of HC-05 in command mode is 38400bps and 9600 in data mode.

HC-05 module has two modes,

          1.  **Data mode:**Exchange of data between devices.

          2.  **Command mode:**It uses AT commands which are used to change setting of HC-05. To send these commands to module serial (USART) port is used.

2.  **VCC:**Connect 5 V or 3.3 V to this Pin.

3.  **GND:**Ground Pin of module.

4.  **TXD:**Transmit Serial data (wirelessly received data by Bluetooth module transmitted out serially on TXD pin)

5.  **RXD:** Receive data serially (received data will be transmitted wirelessly by Bluetooth module).

6.  **State:**It tells whether module is connected or not.

**Pair HC-05 and smartphone**:

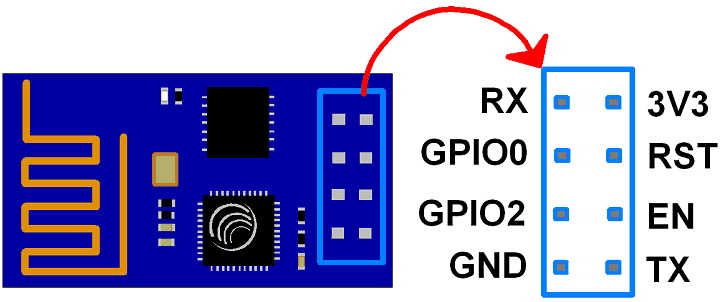
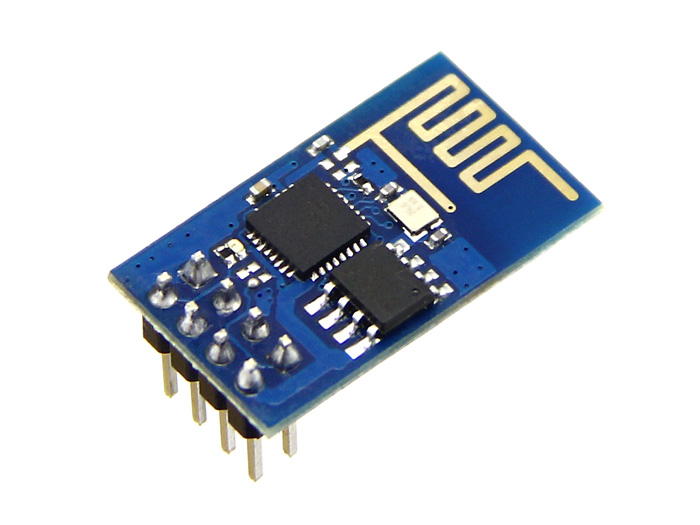
1. Search for new Bluetooth device from your phone. You will find Bluetooth device with “HC-05” name.
2. Click on connect/pair device option; default pin for HC-05 is 1234 or 0000.

After pairing two Bluetooth devices, open terminal software in PC, and select the port where we have connected USB to serial module. Also select default baud rate of 9600 bps.

In smart phone, open Bluetooth terminal application and connect to paired device HC-05.

It is simple to communicate,we just have to type in the Bluetooth terminal application of smartphone. Characters will get sent wirelessly to Bluetooth module HC-05. HC-05 will automatically transmit it serially to the PC, which will appear on terminal. Same way we can send data from PC to smartphone.

ii) **Wi-Fi Module ESP8266:**

SP8266 is Wi-Fi enabled system on chip (SoC) module developed by Espressif system. It is mostly used for development of IoT (Internet of Things) embedded applications 

**3V3**: - 3.3 V Power Pin.

**GND**: - Ground Pin.

**RST**: - Active Low Reset Pin.

**EN**: - Active High Enable Pin.

**TX**: - Serial Transmit Pin of UART.

**RX**: - Serial Receive Pin of UART.

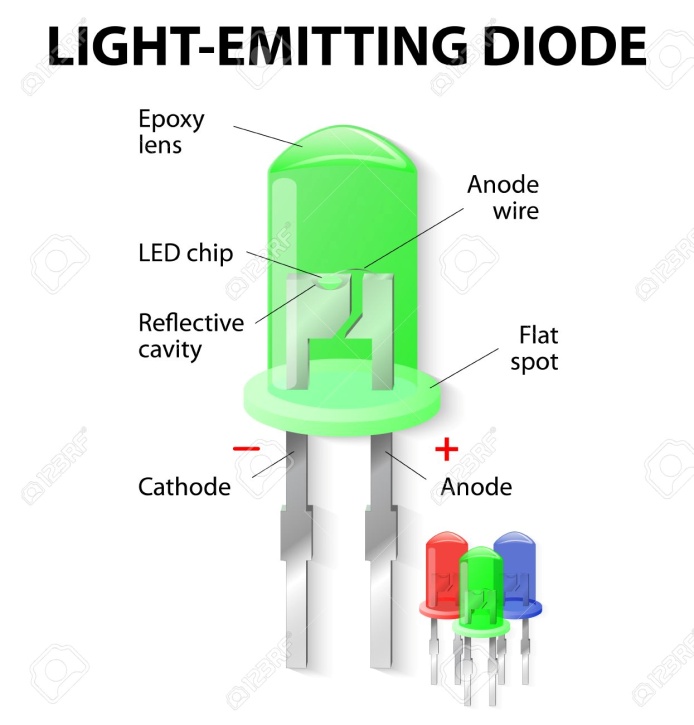
**GPIO0 & GPIO2**: - General Purpose I/O Pins. These pins decide what mode (boot or normal) the module starts up in. It also decides whether the TX/RX pins are used for Programming the module or for serial I/O purpose.

To program the module using UART, Connect GPIO0 to ground and GPIO2 to VCC or leave it open. To use UART for normal Serial I/O leave both the pins open (neither VCC nor Ground).

**6) Components:**

LED’s:

A light-emitting diode is a two-lead semiconductor light source. It is a p–n junction diode that emits light when activated. When a suitable voltage is applied to the leads, electrons are able to recombine with electron holes within the device, releasing energy in the form of photons. This effect is called electroluminescence, and the color of the light (corresponding to the energy of the photon) is determined by the energy band gap of the semiconductor.



**Electromechanical Relay:**

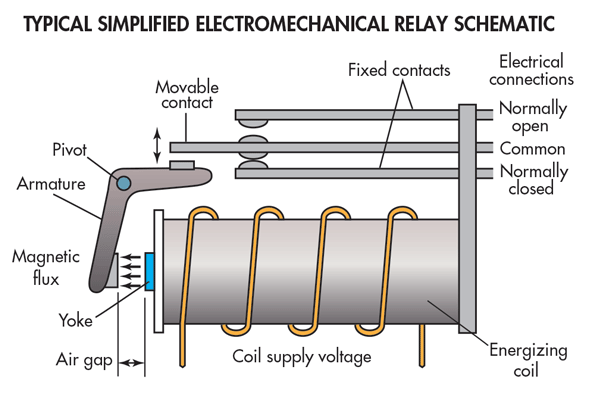
An electrical relay is an electromagnetically operated electrical switch - an electromechanical switch. A relatively small current is used to create a magnetic field in a coil within a magnetic core and this is used to operate a switch that can control a much larger current.

In this way an electromechanical relay or electrical relay can use a small current to switch a much larger current and enable both circuits to be electrically isolated from each other.

Electrical relays come in a variety of different sizes and they can be of a variety of different types using slightly different technologies, although they all use the same basic concept.

Although electromechanical relays may be considered in some respects to use old technology, and solid state relays / solid state switches might be thought to be a more effective means of switching electrical current.

Nevertheless electromechanical relays have some unique properties that make them ideal for many applications, where other types may not be as effective. That said, solid state switches, solid state relays or electronic switches are widely used and have taken over in many areas where electromechanical relays were previously used as electrical switches.



**POTETIOMETER:**

A potentiometer is a three-terminal resistor with a sliding or rotating contact that forms an adjustable voltage divider. If only two terminals are used, one end and the wiper, it acts as a variable resistor or rheostat.

The measuring instrument called a potentiometer is essentially a voltage divider used for measuring electric potential (voltage) the component is an implementation of the same principle, hence its name.

Potentiometers are commonly used to control electrical devices such as volume controls on audio equipment. Potentiometers operated by a mechanism can be used as position transducers, for example, in a joystick. Potentiometers are rarely used to directly control significant power (more than a watt) , since the power dissipated in the potentiometer would be comparable to the power in the controlled load. Potentiometers consist of a resistive element, a sliding contact (wiper) that moves along the element, making good electrical contact with one part of it, electrical terminals at each end of the element, a mechanism that moves the wiper from one end to the other, and a housing containing the element and wiper.



Fig: Potentiometer

**7) DC Motor:**

The **DC Motor** or **Direct Current Motor** to give it its full title, is the most commonly used actuator for producing continuous movement and whose speed of rotation can easily be controlled, making them ideal for use in applications were speed control, servo type control, and/or positioning is required. A DC motor consists of two parts, a “Stator” which is the stationary part and a “Rotor” which is the rotating part. The result is that there are basically three types of DC Motor available.

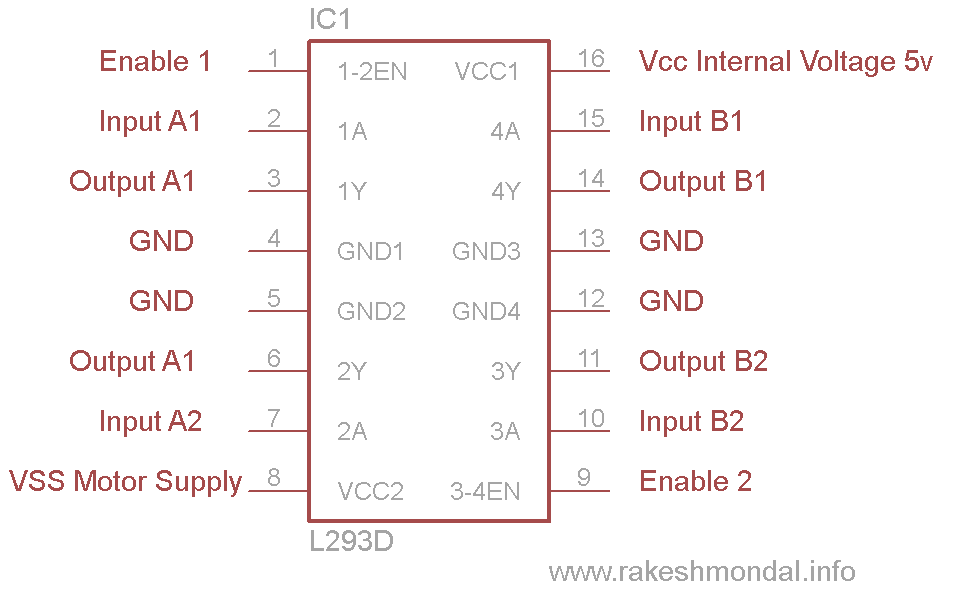
* Brushed Motor – This type of motor produces a magnetic field in a wound rotor (the part that rotates) by passing an electrical current through a commutator and carbon brush assembly, hence the term “Brushed”. The stators (the stationary part) magnetic field is produced by using either a wound stator field winding or by permanent magnets. Generally brushed DC motors are cheap, small and easily controlled.
* Brushless Motor – This type of motor produce a magnetic field in the rotor by using permanent magnets attached to it and commutation is achieved electronically. They are generally smaller but more expensive than conventional brushed type DC motors because they use “Hall effect” switches in the stator to produce the required stator field rotational sequence but they have better torque/speed characteristics, are more efficient and have a longer operating life than equivalent brushed types.
* Servo Motor – This type of motor is basically a brushed DC motor with some form of positional feedback control connected to the rotor shaft. They are connected to and controlled by a PWM type controller and are mainly used in positional control systems and radio controlled models.



Fig: DC Motor

**L293D Motor Driver**:

L293D is a typical Motor driver or Motor Driver IC which allows DC motor to drive on either direction. L293D is a 16-pin IC which can control a set of two DC motors simultaneously in any direction. It means that you can control two [DC motor](https://www.rakeshmondal.info/High-Torque-Motor-Low-RPM-Motor) with a single L293D IC



**Working of L293D**:

There are 4 input pins for l293d, pin 2,7 on the left and pin 15 ,10 on the right as shown on the pin diagram. Left input pins will regulate the rotation of motor connected across left side and right input for motor on the right hand side. The motors are rotated on the basis of the inputs provided across the input pins as LOGIC 0 or LOGIC 1, In simple you need to provide Logic 0 or 1 across the input pins for rotating the motor.

**Source Code**:

int led=13;

int motor1=12;

int relay=11;

int x;

intch,ldr,ir,ultra,distance,time;

intir\_pin=4;

intldr\_pin=5;

inttrigger\_pin = 6;

intecho\_pin = 7;

void setup()

{

pinMode(led,OUTPUT);

pinMode(trigger\_pin,OUTPUT);

pinMode(motor1,OUTPUT);

pinMode(ir\_pin,INPUT);

pinMode(ldr\_pin,INPUT);

pinMode(echo\_pin,INPUT);

pinMode(relay,OUTPUT);

digitalWrite(relay,HIGH);

Serial.begin(9600);

wifi\_init();

}

void loop()

{

ldr=digitalRead(ldr\_pin);

ir=digitalRead(ir\_pin);

ultra=digitalRead(echo\_pin );

if (ir==0)// IR Sensor value exceed threshold limit

{

digitalWrite(led, HIGH);// led will be off

digitalWrite(motor1, HIGH);// buzzer will sound

}

else

{

digitalWrite(led, LOW);//

digitalWrite(motor1, LOW);

}

//LDR starts

if (ldr==1)//Sensor value exceed threshold limit

{

digitalWrite(led, HIGH);// led will be off

}

else

{

digitalWrite(led, LOW);//

}

digitalWrite (trigger\_pin, HIGH);//ultrasonic sensor

digitalWrite (trigger\_pin, LOW);

time = pulseIn (echo\_pin, HIGH);

distance = (time \* 0.034) / 2;

if (distance <= 5)

{

Serial.println (" tank's full");

Serial.print (" Distance= ");

Serial.println (distance);

digitalWrite (relay, LOW);

}

else

{

Serial.println (" tank's empty");

Serial.print (" Distance= ");

Serial.println (distance);

digitalWrite (relay, HIGH);

}

upload\_iot(ir,ldr,ultra);

x=read\_iot();

Serial.println(char(x));

if(x=='1')

{

digitalWrite(led,1);

}

if(x=='2')

{

digitalWrite(led,0);

}

if(x=='3')

{

digitalWrite(motor1,1);

}

if(x=='4')

{

digitalWrite(motor1,0);

}

}

voidwifi\_init()

{

Serial.println("AT+RST");

Serial.println("AT+CWMODE=3");

Serial.print("AT+CWJAP=");

Serial.write('"');

Serial.print("srcesolutions"); // ssid/user name

Serial.write('"');

Serial.write(',');

Serial.write('"');

Serial.print("srcesolutions"); //password

Serial.write('"');

Serial.println();

//delay(100);

}

voidupload\_iot(intx,inty,int z) //ldr copied int to - x and gas copied into -y

{

String cmd = "AT+CIPSTART=\"TCP\",\"";

cmd += "184.106.153.149"; // api.thingspeak.com

cmd += "\",80";

Serial.println(cmd);

//delay(150);

String getStr ="GET /update?api\_key=J9CKY6YWMTADQJ8R&field1=";//paste sensor monitering write api key

getStr += String(x);

getStr +="&field2=";

getStr += String(y);

getStr += "\r\n\r\n";

getStr += String(z);

getStr += "\r\n\r\n";

cmd = "AT+CIPSEND=";

cmd += String(getStr.length());

Serial.println(cmd);

//delay(250);

Serial.println(getStr);

}

intread\_iot() //ldr copied int to - x and gas copied into -y

{

intch;

String cmd = "AT+CIPSTART=\"TCP\",\"";

cmd += "184.106.153.149"; // api.thingspeak.com

cmd += "\",80";

Serial.println(cmd);

//delay(150);

String getStr ="GET /channels/814291/fields/1/last?api\_key=U3CP5H6UNZLF39BN\r\n\r\n";//paste device control channel id and then read api key

// send data length

cmd = "AT+CIPSEND=";

cmd += String(getStr.length());

Serial.println(cmd);

//delay(100);

Serial.print(getStr);

//delay(200);

ch=0;

if(Serial.find("~"))

{

//delay(10);

ch=Serial.read();

}

returnch;

}