 MLR Institute of Technology

(Autonomous)

**(Affiliated to JNTUH, Hyderabad)**

**Dundigal, Hyderabad-500043**

**SMART CIRCUIT BREAKER USING IoT**

A

MINI PROJECT REPORT

**Submitted in the partial Fulfillment of the requirements for the award of the**

**Degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**ELECTRICAL AND ELECTRONICS ENGINEERING**

**Submitted By**

1. **Name:S.Sai Bhargav RollNo.20R21A0228**
2. **Name:M.V.Phaneendra RollNo.20R21A0223**
3. **Name:G.Sampath RollNo.20R21A0213**

**4.Name:K.Kunal RollNo.20R21A0234**

**UNDER THE GUIDANCE OF**

**Dr.A.Sudhakar**

**(HOD of EEE )**

 MLR Institute of Technology

(Autonomous)

**(Affiliated to JNTUH, Hyderabad)**

**Dundigal, Hyderabad-500043**

**DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING**

***CERTIFICATE***

This is to certify that the project entitled “Smart circuit breaker using IoT” is the bonafide work done by S.Sai Bhargav (20R21A0228), M.V.Phaneendra (20R21A0223), G.Sampath Kumar (20R21A0213), K.Kunal (20R21A0234) in partial fulfilment of the requirement for the award of degree of b.tech in Electric and Electronic Engineering during academic year 2022-2023.

**Internal Guide Head of Department**

**External Examiner**

ACKNOWLEDGEMENT

We express our profound thanks to the management of **MLR Institute of Technology**, Dundigal, Hyderabad, for supporting us to complete this project.

We take immense pleasure in expressing our sincere thanks to **Dr.K.Srinivasa Rao**, Principal, MLR Institute of Technology, for his kind support and encouragement.

We are very much grateful to **Dr.A.Sudhakar**, Professor & Head of the Department, MLR Institute of Technology, for encouraging us with his valuable suggestions.

We are very much grateful to **Internal Guide Dr.A.Sudhakar,Head of the Department** for his unflinching cooperation throughout the project.

We would like to express our sincere thanks to the teaching and non teaching faculty members of EEE Dept., MLR Institute of Technology, who extended their help to us in making our project work successful.

Project associates:

S.SaiBhargav(20R21A0228)

M.V.Phaneendra (20R21A0223)

G.Sampath Kumar(20R21A0213)

K.Kunal (20R21A0234)

**ABSTRACT**

**SMART CIRCUIT BREAKER USING IOT**

Nowadays, electrical accidents of the line man are increased day by day due to electrical shocks, while repairing the electrical lines and this happens due to the lack of communication between the electrical substation and maintenance staff. Thus, to ensure the safety of line man a system is proposed using ESP 8266 to control i.e., ON/OFF the electrical lines. In this system the maintenance staff or lineman has to login into ThingSpeak Application to ON/OFF the electrical line. If there is any fault in electrical line then lineman will switch off the power supply to the line by using ThingSpeak application and comfortably repair the electrical line, and after completing the maintenance, the lineman will switch on the supply to the particular line by using ThingSpeak application. The relay ON/OFF operation will be indicated by the LEDs, also it displays the status on LCD. As soon as the maintenance work is finished the line man can log out the ThingSpeak Application.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S. No** | **Student Name** | **Roll No** | **Class** | **Phone No** | **Email** |
| 1 | S.SAI BHARGAV | 20R21A0228 | EEE | 7013598150 | [20R21A0228@mlrinstitutions.ac.in](mailto:20R21A0228@mlrinstitutions.ac.in) |
| 2 | M.V,PHANEENDRA | 20R21A0223 | EEE | 9000451808 | [20R21A0223@mlrinstitutions.ac.in](mailto:20R21A0223@mlrinstitutions.ac.in) |
| 3 | G,SAMPATH KUMAR | 20R21A0213 | EEE | 9346848391 | [20R21A0213@mlrinstitutions.ac.in](mailto:20R21A0213@mlrinstitutions.ac.in) |
| 4 | K.KUNAL | 20R21A0234 | EEE | 8096490340 | [20R21A0234@mlrinstitutions.ac.in](mailto:20R21A0234@mlrinstitutions.ac.in) |

**Guide:** Dr.A.SUDHAKAR

**TABLE OF CONTENTS:**

**Chapter 1**

|  |  |
| --- | --- |
| **Content** | **Page no**. |
| Introduction | 09 |
| Literature Review | 10 |

**Chapter 2**

|  |  |
| --- | --- |
| **Content** | **Page no**. |
| Proposed Methodology | 11-12 |
| Actual Proposed System | 13 |

**Chapter 3**

|  |  |
| --- | --- |
| **Content** | **Page no**. |
| Wifi Module Code | 14-15 |
| Arduino Code | 16-17 |

**Chapter 4**

|  |  |
| --- | --- |
| **Content** | **Page no**. |
| Introduction to Arduino UNO | 18-20 |
| Introduction to Relays | 21-22 |
| WiFi Module ESP8266 | 23-25 |
| Regulated Power Supply | 26-28 |

**Chapter 5**

|  |  |
| --- | --- |
| **Content** | **Page no**. |
| Liquid Crystal Display | 29-32 |
| ThingSpeak | 33-35 |
| Result & Discussion | 36-37 |
| Conclusion | 38 |
| Future Scope | 38 |

**List of Figures:**

**Chapter 2**

|  |  |
| --- | --- |
| **Figures** | **Page no**. |
| 2.1 Proposed Block Diagram | 12 |
| 2.2 Circuit Diagram | 12 |
| 2.3 Pin Diagram | 13 |

**Chapter 4**

|  |  |
| --- | --- |
| **Figures** | **Page no**. |
| 4.1 Arduino | 20 |
| 4.2 Relay | 21 |
| 4.3 SPDT (Relay) | 21 |
| 4.4 Circuit Diagram (Relay) | 22 |
| 4.5 ESP8266 | 23 |
| 4.6 Hardware Setup | 24 |
| 4.7 Block Diagram (RPS) | 26 |
| 4.8 Full Wave Rectifier | 27 |
| 4.9 Bridge Rectifier | 28 |

**Chapter 5**

|  |  |
| --- | --- |
| **Figures** | **Page no**. |
| 5.1 MxN LCD | 29 |
| 5.2 LCD | 30 |
| 5.3 Pin Diagram | 30 |
| 5.4 Data Aggregation & Analytics | 33 |
| 5.5 ThingSpeak+Arduino | 34 |
| 5.6 ThingSpeak+C# | 35 |
| 5.7 Results & Discussions | 36 |
| 5.8 Result Snapshot | 37 |

**CHAPTER 1**

**Introduction**

Security is the main concern in our day-to-day life while performing any activity. Circuit breakers are mostly used for protecting & switching industrial electrical systems. Hence, reliability is most important for this. Failure of circuit breaker results in huge damage including revenue loss & fatality. circuit breakers are subjected to stress during their operation because it contains many mechanical as well as electrical components, which carry rated or fault power & participates in arc quenching methods. so the concern regarding reliability of circuit breaker operation raises issue. In order to discover reliability of circuit breaker, it is general practice to carry out preventive maintenance at fixed time intervals. This proposed system is designed to ensure the safety by controlling the circuit breaker by using ThingSpeak application. The proposed system facilitates automated circuit breaker. This is a shift in maintenance paradigm from time-based maintenance to as needed maintenance. This shift ensures the benefit to the safety of lineman by reducing overall maintenance cost and unnecessary downtime. In our system automatic circuit breaker operated electrical switch which is designed to protect lineman from damage caused by an overload or short circuit. With Fuse which operates once and then need to be replaced, but with an automated circuit breaker, it can be used to set or reset automatically to resume normal operation. The proposed system aims to solve this issue by making the use of Internet of Things (IOT). Proposed system is with high response time, which uses the interconnection network (internet) to control electrical loads. This system uses a Wi-Fi module paired with ESP 8266 microcontroller. LCD display is used in our system which shows the state of each load connected to the system, either ON or OFF. A user or operator at the other end can remotely connect with the system using a wireless device and an interface named “ThingSpeak Application”. From the ThingSpeak App interface, the user can control the state ON/OFF the connected loads remotely. The proposed system build using has an advantage over other remotely operable circuit breakers as it uses a faster medium which is the wireless network to connect with the loads. This proposed system is indispensable when industrial safety comes to play, as use of such circuit braking systems removes any chance of common accidents, such as electrical shocks

**Literature Review**:

In past few years many theories were proposed about the circuit breaker system using IoT.

Mane Kirti M[1] proposed a system that access only specified password to control the circuit breaker by authorized person only. It is fully controlled by the microcontroller which has an 8KB of ROM and matrix keypad was used to enter the password and relay driver to switch ON/ OFF loads through relays. Bhagwan Kharat[2] proposes a system that presents the architecture of an online monitoring and diagnosis System of an electrical equipment which has role to acquire, transfer and process information about monitored equipment. An interface is designed on top of which different local as well as system applications can be recorded. Controller will continuously transmit parameters of circuit breaker to control room and also displayed on Monitor of computer as well as after C.B trip SMS get sent to the registered mobile number. Once message is received, the operator or any authorized person will give command to set or reset the breaker. It minimizes the fault clearing time and improves maintenance method which increases life time and reliability of the circuit breaker. Abhijit Das [3] proposed a system that develops monitoring & control scheme of a typical circuit breaker using Arduino Mega 2560 embedded microcontroller along with Ethernet Shield for integration of IOT. The system facilitates diagnose of the electrical and mechanical health of circuit breaker in real time. It uses Open-source platform which eliminates the concern regarding reliability & security of the safety/safety related/strategic application as complete source code implementation is open & fully accessible to the user. Lakshmi Devdas [4] proposed an idea that put forwards a more secured operating 8 condition for the line man and ensures unauthorized access and that a switching of the line does not occurs. This is implemented using a password verification method. With the help of matrix keypad user can enter password which is compared with a pre-set password. If it matches, the line man 11 can operate the line according the requirement. Parameters like voltage and current with ON/OFF status are accessed on a webpage, connected via an IOT Athira Nair [5] have proposed a system that focuses on the safety of the lineman while working so they do not feel the sudden electric shock. With proper coordination among lineman and substation, lot of accidents can be avoided. The project aimed to provide the solution that ensures the safety of maintenance staff. The lineman detects the fault in the electric line, SMS get sent to the substation staff, who switch off the line and turn it on when the fault is resolved. Application whether to ON/OFF the circuit. After Receiving the input, the microcontroller sends output to the Relay Module to ON/OFF the circuit. It also displays the status of the load by using the LCD display module. Relay module is connected with the external AC supply to power the load at normally open pin. A relay is working on electromagnetic principles. As soon as it gets instruction from the microcontroller it completes or breaks the circuit by attracting and repelling the armature and load get ON/OFF without any human efforts.

**Chapter 2**

PROPOSED METHODOLOGY

ESP8266 microcontroller is the main component of the proposed system which passes the information to switch the load ON / OFF by the input given by the user through IoT. Many loads can be controlled at a time by using a Wi-Fi module with the help of Thing Speak Application. It uses Mobile hotspot to get connected with the Microcontroller. ThingSpeak Application is user friendly. An Account must be created In the ThingSpeak Application. In the ThingSpeak Application widget must be created to control the Load. With the help of Arduino Software ESP8266 is programmed to work accordingly. To instruct the ESP8266 about ThingSpeak Interface, ThingSpeak library is included in the Arduino Software. The ESP8266 microcontroller receives input from the ThingSpeak Application whether to ON/OFF the circuit. After Receiving the input, the microcontroller sends output to the Relay Module to ON/OFF the circuit. It also displays the status of the load by using the LCD display module. Relay module is connected with the external AC supply to power the load at normally open pin. A relay is working on electromagnetic principles. As soon as it gets instruction from the microcontroller it completes or breaks the circuit by attracting and repelling the armature and load get ON/OFF without any human efforts.

The circuit breaker system being operated using IoT is carried out by performing certain steps as given below:

(1) ESP8266 is used as it has micro controller with free Wi-Fi module. It is very reliable for interfacing with ThingSpeak Application without using external Wi-Fi module. ESP8266 can be programmed by Arduino IDE

(2) Relay Module is reliable for AC load. It can be used for Controlling AC loads using low voltage. It is powered with 3.3 V pin of ESP8266.

(3Mobile interface is used for displaying the status of Load and the Connection of ThingSpeak Application.

(4) Load here can be any AC devices.

(5) ThingSpeak Application is used to control the state (ON/OFF) of the connected loads remotely.

**Proposed Block Diagram**:

Fig.2.1

ARDUINO

LCD 16X2

Relay2

Relay1

Power Supply

WIFI  
ESP8266

LOAD1

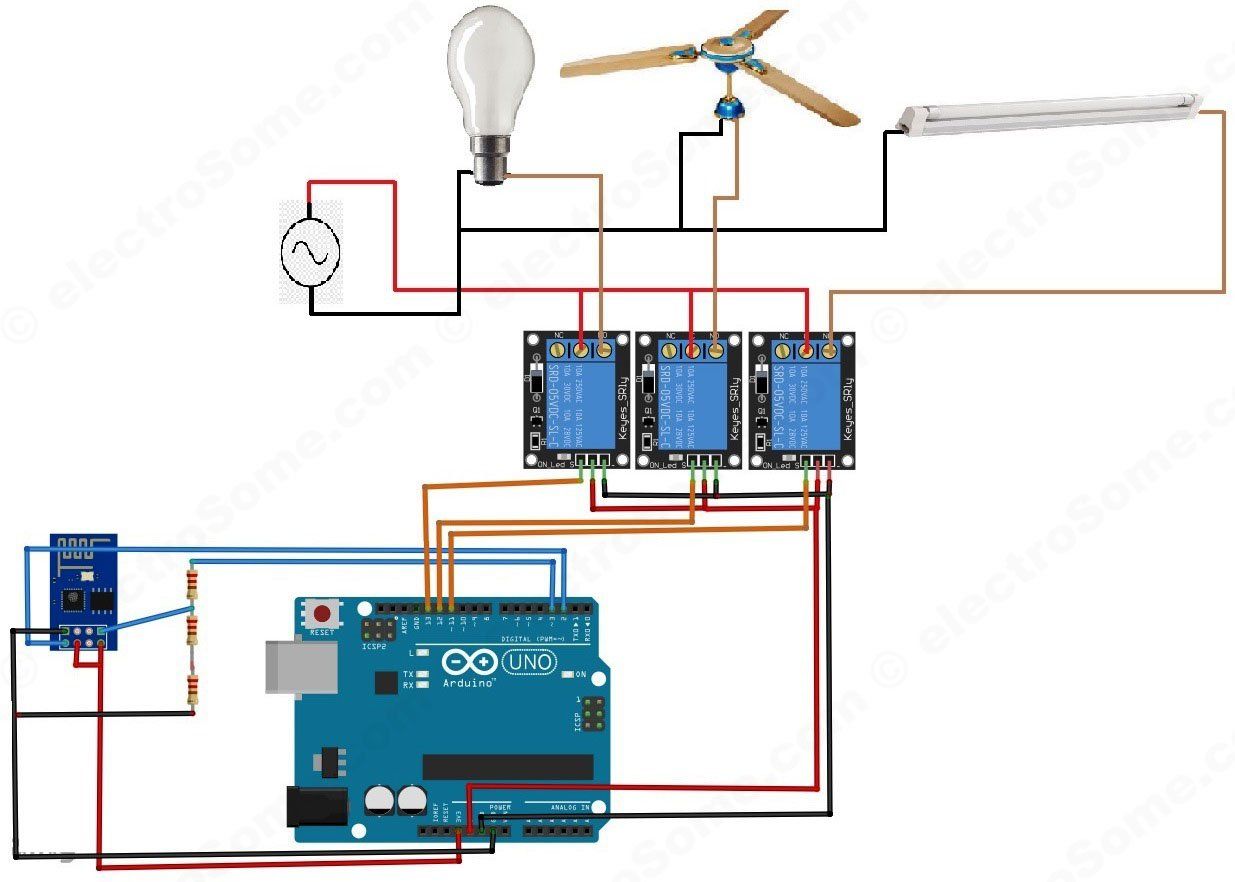
Load2

CLOUD

MOBILE

**Circuit Diagram:**

Fig.2.2



**Actual Proposed System:**

**The working of the proposed system is as listed below:**

(1) The proposed system supports Wireless Local Area Network (WLAN) architecture. ESP 8266 is the Access Point (AP).

(2) The proposed system is automated circuit breaker.

(3 ESP8266 is powered by 230VAC to 5VDC converter through Vin.

(4) Pin D6 is connected to IN pin of Relay module.

(5)I2C bus and relay module both are powered by 5V convertor and ground pin is connected to GND pin of ESP8266.

(6)I2C bus is connected to LCD display matrix for reducing the pin load on ESP8266.

**PIN DIAGRAM**:

Fig.2.3

D0

D1

D2

D3

D4

ESP 8266 GND

D5

D6

D7

D8

GND

VCC

4-Channel Relay

GND CH1

IN CH2

VCC CH3

CH4

**Chapter 3**

**Wifi module code:**

#include "ThingSpeak.h"

#include <ESP8266WiFi.h>

//Replace wifi credentials here

const char\* ssid = "TEST1234";//Replace with Wifi Name

const char\* password = "test1234";// Replace with wifi Password

//change channel number here

unsigned long channel =419422;

unsigned int sen1 = 1;

unsigned int sen2 = 2;

WiFiClient client;

void setup() {

pinMode(D1, OUTPUT);

pinMode(D2, OUTPUT);

digitalWrite(D1, 0);

digitalWrite(D2, ,0);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

}

WiFi.localIP();

WiFi.subnetMask();

WiFi.gatewayIP();

ThingSpeak.begin(client);

}

void loop() {

//get the last data of the fields

int sen\_1 = ThingSpeak.readFloatField(channel, sen1);

int sen\_2 = ThingSpeak.readFloatField(channel, sen2);

if(sen\_1 == 1){

digitalWrite(D1, 1);

}

else if(sen\_1 == 0){

digitalWrite(D1, 0);

}

if(sen\_2 == 1){

digitalWrite(D2, 1);

}

else if(sen\_2 == 0){

digitalWrite(D2, 0);

}

delay(5000);

}

**Arduino code:**

#include <LiquidCrystal.h>

char inChar;

const int load1=4;

const int load2=5;

int sen1=6;

int sen2=7;

LiquidCrystal lcd(13, 12, 11, 10, 9, 8); digitalWrite(load1,HIGH)); digitalWrite(load2,LOW);

void setup() {

// setup code here, to run once:

pinMode(load1, OUTPUT);

pinMode(load2, OUTPUT);

digitalWrite(load1,HIGH);

digitalWrite(load2,HIGH);

pinMode(sen1,INPUT);

pinMode(sen2,INPUT);

lcd.begin(16, 2);

lcd.clear();

lcd.setCursor(0,0);

lcd.print("IOT BASED HOME");

lcd.setCursor(0,1);

lcd.print("AUTOMATION");

delay(500);

void loop() {

// main code here, to run repeatedly:

if(digitalRead(sen1)==LOW){

lcd.setCursor(0,0);

lcd.print("LOAD1: ON");

delay(500);

digitalWrite(load1,HIGH);

}

if(digitalRead(sen1)==HIGH){

lcd.setCursor(0,0);

lcd.print("LOAD1: OFF ");

delay(500);

digitalWrite(load1,LOW);

}

if(digitalRead(sen2)==LOW){

lcd.setCursor(0,1);

lcd.print("LOAD2: ON ");

delay(500);

digitalWrite(load2,HIGH);

}

if(digitalRead(sen2)==HIGH){

lcd.setCursor(0,1);

lcd.print("LOAD2: OFF ");

delay(500);

digitalWrite(load2,LOW);

}

}

**Chapter 4**

# INTRODUCTION TO ARDUINO UNO

Arduino is a tool for making computers that can sense and control more of the physical world than on a desktop computer. It's an open-source physical computing platform based on a simple microcontroller board, and a development environment for writing software for the board.

Arduino can be used to develop interactive objects, taking inputs from a variety of switches or sensors, and controlling a variety of lights, motors, and other physical outputs. Arduino projects can be stand-alone, or they can be communicating with software running on a computer. The boards can be assembled by hand or purchased preassembled; the open-source IDE can be downloaded for free. The Arduino programming language is an implementation of Wiring, a similar physical computing platform, which is based on the Processing multimedia programming environment.

**What is Arduino?**

Arduino is an open source electronics platform accompanied with a hardware and software to design, develop and test complex electronics prototypes and products. The hardware consists of a microcontroller with other electronic components which can be programmed using the software to do almost any task. The simplicity of the Arduino language makes it very easy for almost everyone who has an interest in electronics to write programs without the understanding of complex algorithms or codes.

Arduino is intended for an artist, tinker, designer or anyone, interested in playing with electronics without the knowhow of complex electronics and programming skills. Arduino is an excellent designed open source platform. It has specially designed boards which can be programmed using the Arduino Programming Language (APL).

The presence of Arduino is not only spreading between hobbyists, but it has also expanded its roots in industries and used by experts for making prototypes of commercial products. Arduino takes off the efforts required in complex coding and designing hardware.

The open source nature of Arduino has been the main reason for its rapid horizontal growth. Since it is an Open Source project, all the files related to hardware and software is available for personal or commercial use. The development cost of the hardware is very small as against the costly similar proprietary products by the industrial giants. The open source nature doesn’t require any licenses to develop, use, redistribute or even sell the product. But the Arduino name is trade mark protected (Arduino™) i.e., these are free to sell the Arduino board under any other name however in order to sell it under the name “Arduino” need to take permission from the founders and follow their quality terms.

The Software files which includes all the source code library are also open sourced. A user can modify them to make the project more versatile and improve its capabilities. This provides a strong online community support.

**Concept of Arduino:**

The root of Arduino goes deep down to the development of Processing Language by MIT researchers. Processing language is an open source language designed to introduce the software development environment for the artistic people without the need of deep knowledge of programming of algorithms. Processing is based on java.

  In early year of 21st century, designing an electronics gadget was nearly impossible for a common man. The requirement of specific skill set and hefty prices of software and hardware created a full stop in the path of their creativity.

  In year 2003 Hernando Barragan, a programmer developed an open source electronics development platform with software IDE, where anyone with a small knowledge in electronics and programming could use his project to give wings to their creativity. His focus was to reduce the burden of complexity in designing electronics hardware and software. The project was named as Wiring. The software IDE of the Wiring used processing language to write the codes.

  As the program written in C\C++ is named as Project, in the same way the code written in Wiring (even in Processing and Arduino) is termed as *Sketch*. The name sketch gives a familiar look for an artist.

The principle idea behind Wiring is that one can make the sketch of their idea on Wiring software and implement it using specially designed Wiring board. This need to write a few lines of codes on the software IDE and then download the program to the onboard microcontroller to see the output.

  Wiring has predefined libraries to make the programming language easy. Arduino uses these libraries. The predefined libraries are written in C and C++. One can even write his software in C\C++ and use them on *wiring* boards. The difference between writing a program in C/C++ and Wiring is that the Wiring *Application Programmable Interface* (API) has simplified programming style and the user doesn’t require detailed knowledge of the concepts like classes, objects, pointers, etc. While sketching hardware need to call the predefined functions and rest will be handled by the Wiring software.

The basic difference between the Processing and the Wiring is that the Processing is use to write the program which can be used on other computers while Wiring program is used on microcontrollers.

**History:**

Wiring is the predecessor of Arduino. Arduino was developed in lvrea, Italy by Massimo Banzi and David Cuartielles in year 2005. The Project was named after Arduin of lvrea (King of Italy). The project Arduino uses the Wiring language. The concept of Wiring Language was created by Hernando Barragan, and under his supervision Massimo Banzi and David Cuartielles developed the Project Arduino.

**Open Source License**

Arduino is an open source project which is probably the root cause reason for its popularity. Arduino hardware design is an Open Source Hardware, distributed under ***Creative Common****Attribution Share-Alike license.* Creative Common, a non-profitable organization has released several copyleft-licenses as free of charge, so that the creativity/ knowledge can be shared to the rest of the world while having the copyright to the authorized person. The originally designed files, like layout and schematics of Arduino products are available as Eagle CAD files.

The source code for its IDE and libraries are also available and released under GUN General Public License (known as GPL). The GPL is the first copyleft license for general use. The license is granted for the software to ensure the copyleft freedom.



Fig.4.1

**RELAYS**

**Introduction:**

A relay is an electrical switch that opens and closes under the control of another electrical circuit. In the original form, the switch is operated by an electromagnet to open or close one or many sets of contacts. A relay is able to control an output circuit of higher power than the input circuit, it can be considered to be, in a broad sense, a form of an electrical amplifier.

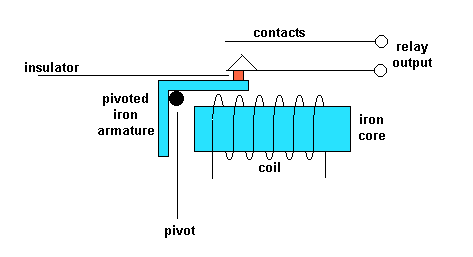
****

Fig.4.2

Relays are usually SPDT (single pole double through switch) or DPDT (double pole double through switch) but they can have many more sets of switch contacts, for example relays with 4 sets of changeover contacts are readily available.

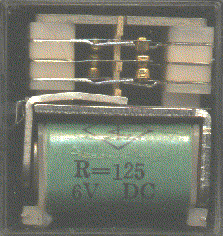
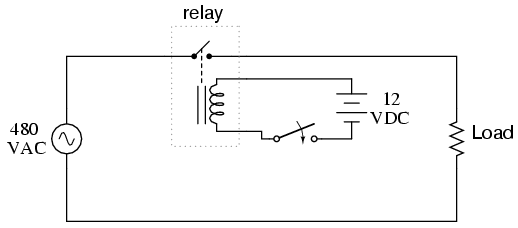


Fig.4.3

**Basic operation of a relay:**

An electric current through a conductor will produce a magnetic field at right angles to the direction of electron flow. If that conductor is wrapped into a coil shape, the magnetic field produced will be oriented along the length of the coil. The greater the current, the greater the strength of the magnetic field, all other factors being equal.



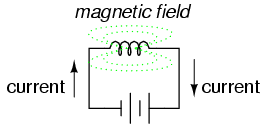


Fig.4.4

Inductors react against changes in current because of the energy stored in this magnetic field. While constructing a transformer from two inductor coils around a common iron core, using this field to transfer energy from one coil to the other. However, there are simpler and more direct uses for electromagnetic fields than the applications have seen with inductors and transformers. The magnetic field produced by a coil of current-carrying wire can be used to exert a mechanical force on any magnetic object, just as it can use a permanent magnet to attract magnetic objects, except that this magnet (formed by the coil) can be turned on or off by switching the current on or off through the coil.

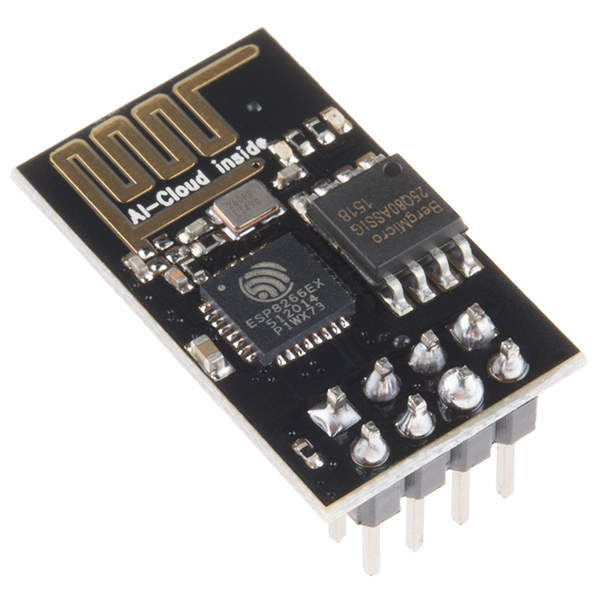
While placing a magnetic object near such a coil for the purpose of making that object move when energize the coil with electric current, is called a solenoid. The movable magnetic object is called an armature, and most armatures can be moved with either direct current (DC) or alternating current (AC) energizing the coil. The polarity of the magnetic field is irrelevant for the purpose of attracting an iron armature. Solenoids can be used to electrically open door latches, open or shut valves, move robotic limbs, and even actuate electric switch mechanisms and are used to actuate a set of switch contacts.

**Wi-Fi Module - ESP8266:**

**Description:**

The ESP8266 Wi-Fi Module is a self contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to Wi-Fi network. The ESP8266 is capable of either hosting an application or offloading all Wi-Fi networking functions from another application processor. Each ESP8266 module comes pre-programmed with an AT command set firmware, meaning, it can simply hook this up to the Arduino device and get about as much Wi-Fi-ability as a Wi-Fi Shield offers (and that’s just out of the box)! The ESP8266 module is an extremely cost effective board with a huge, and ever growing, community.

Fig.4.5



This module has a powerful enough on-board processing and storage capability that allows it to be integrated with the sensors and other application specific devices through its GPIOs with minimal development up-front and minimal loading during runtime. Its high degree of on-chip integration allows for minimal external circuitry, including the front-end module, is designed to occupy minimal PCB area. The ESP8266 supports APSD for VoIP applications and Bluetooth co-existance interfaces, it contains a self-calibrated RF allowing it to work under all operating conditions, and requires no external RF parts.

There is an almost limitless fountain of information available for the ESP8266, all of which has been provided by amazing community support. In the *Document* section below will find many resources to aid in using the ESP8266, even instructions on how to transforming this module into an IoT (Internet of Things) solution!

**Note:** The ESP8266 Module is not capable of 5-3V logic shifting and will require an external [Logic Level Converter](https://www.sparkfun.com/products/12009). Please do not power it directly from 5V dev board.

**Note:** This new version of the ESP8266 WiFi Module has increased the flash disk size from 512k to 1MB.

**Features:**

* 802.11 b/g/n
* Wi-Fi Direct (P2P), soft-AP
* Integrated TCP/IP protocol stack
* Integrated TR switch, balun, LNA, power amplifier and matching network
* Integrated PLLs, regulators, DCXO and power management units
* +19.5dBm output power in 802.11b mode
* Power down leakage current of <10uA
* 1MB Flash Memory
* Integrated low power 32-bit CPU could be used as application processor
* SDIO 1.1 / 2.0, SPI, UART
* STBC, 1×1 MIMO, 2×1 MIMO
* A-MPDU & A-MSDU aggregation & 0.4ms guard interval
* Wake up and transmit packets in < 2ms
* Standby power consumption of < 1.0mW (DTIM3)

#### Hardware setup:

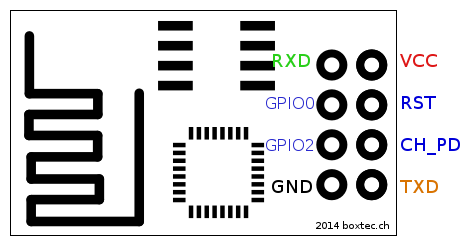


Fig.4.6

The hardware connections required to connect to the ESP8266 module are fairly straight-forward but there are a couple of important items to note related to power:

* The ESP8266 requires 3.3V power–do not power it with 5 volts!
* The ESP8266 needs to communicate via serial at 3.3V and does not have 5V tolerant inputs, so it need level conversion to communicate with a 5V microcontroller like most Arduinos use.

However, if anyone are to a adventurous and have no fear possibly getting away with ignoring the second requirement. But nobody takes any responsibility for what happens if they do. Here are the connections available on the ESP8266 WiFi module

**WEB SERVER:**

A **web server** is a computer system that processes requests via [HTTP](https://en.wikipedia.org/wiki/HTTP), the basic [network protocol](https://en.wikipedia.org/wiki/Network_protocol) used to distribute information on the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web). The term can refer to the entire system, or specifically to the [software](https://en.wikipedia.org/wiki/Software) that accepts and supervises the HTTP requests.

Web servers are computers that deliver (serves up) Web pages. Every Web server has an IP address and possibly a domain name. For example, if anyone enter the URL http://www.webopedia.com/index.html in browser, this sends a request to the Web server whose domain name is webopedia.com. The server then fetches the page named index.html and sends it to browser.

Any computer can be turned into a Web server by installing server software and connecting the machine to the Internet. There are many Web server software applications, including public domain software and commercial packages.

Every Website sits on a computer known as a Web server. This server is always connected to the internet. Every Web server that is connected to the Internet is given a unique address made up of a series of four numbers between 0 and 255 separated by periods. For example, 68.178.157.132 or 68.122.35.127.

A register web address, also known as a domain name, such as tutorialspoint.com have to specify the IP address of the Web server that will host the site. It can load up with Dedicated Servers that can support web-based operations.

**REGULATOR POWER SUPPLY**

A **regulated power supply** is an embedded circuit; it converts unregulated AC into a constant DC. With the help of a rectifier it converts AC supply into DC. Its function is to supply a stable voltage (or less often current), to a circuit or device that must be operated within certain power supply limits. The output from the regulated power supply may be alternating or unidirectional, but is nearly always DC.

A regulated DC power supply is also called as a linear power supply; it is an embedded circuit and consists of various blocks.

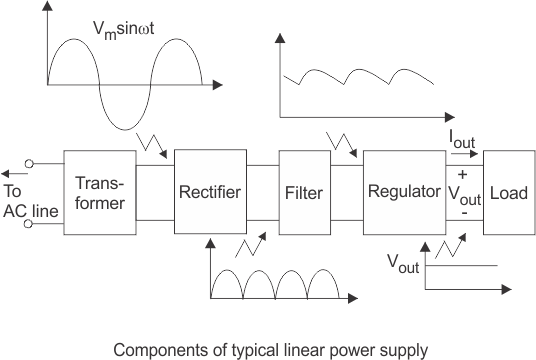
The regulated power supply will accept an AC input and give a constant DC output. Figure below shows the block diagram of a typical regulated DC power supply. 

Fig.4.7

The basic building blocks of a regulated DC power supply are as follows:

1. A step down transformer
2. A rectifier
3. A DC filter
4. A regulator

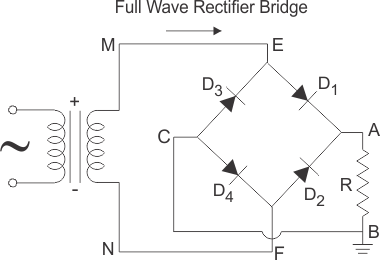
**Operation of Regulated Power Supply:**

**Step Down Transformer:**

A [step down transformer](https://www.electrical4u.com/step-down-transformers/) will step down the voltage from the ac mains to the required voltage level. The turn’s ratio of the transformer is so adjusted such as to obtain the required voltage value. The output of the [transformer](https://www.electrical4u.com/what-is-transformer-definition-working-principle-of-transformer/) is given as an input to the rectifier circuit.

**Rectification:**

Rectifier is an electronic circuit consisting of [diodes](https://www.electrical4u.com/diode-working-principle-and-types-of-diode/) which carries out the rectification process. Rectification is the process of converting an alternating voltage or current into corresponding direct (DC) quantity. The input to a rectifier is ac whereas its output is unidirectional pulsating DC. Usually a [full wave rectifier](https://www.electrical4u.com/full-wave-rectifiers/) or a bridge rectifier is used to rectify both the half cycles of the ac supply (full wave rectification). Figure below shows a [full wave bridge rectifier](https://www.electrical4u.com/full-wave-diode-rectifier/).

Fig.4.8

A bridge rectifier consists of four [p-n junction diodes](https://www.electrical4u.com/p-n-junction-diode/) connected in the above shown manner. In the positive half cycle of the supply the voltage induced across the secondary of the [electrical transformer](https://www.electrical4u.com/what-is-transformer-definition-working-principle-of-transformer/) i.e. VMN is positive. Therefore point E is positive with respect to F. Hence, diodes D3 and D2 are reversed biased and diodes D1 and D4 are forward biased. The [diode](https://www.electrical4u.com/diode-working-principle-and-types-of-diode/) D3 and D2 will act as open switches (practically there is some [voltage drop](https://www.electrical4u.com/voltage-drop-calculation/)) and diodes D1 andD4 will act as closed switches and will start conducting. Hence a rectified waveform appears at the output of the rectifier as shown in the first figure. When voltage induced in secondary i.e. VMN is negative than D3 and D2 are forward biased with the other two reversed biased and a positive [voltage](https://www.electrical4u.com/voltage-or-electric-potential-difference/) appears at the input of the filter

**DC Filtration:**

The rectified voltage from the rectifier is a pulsating DC voltage having very high ripple content. But this is not a pure ripple free DC waveform. Hence a filter is used. Different types of filters are used such as [capacitor](https://www.electrical4u.com/what-is-capacitor/) filter, LC filter, Choke input filter, π type filter. Figure below shows a capacitor filter connected along the output of the rectifier and the resultant output waveform.

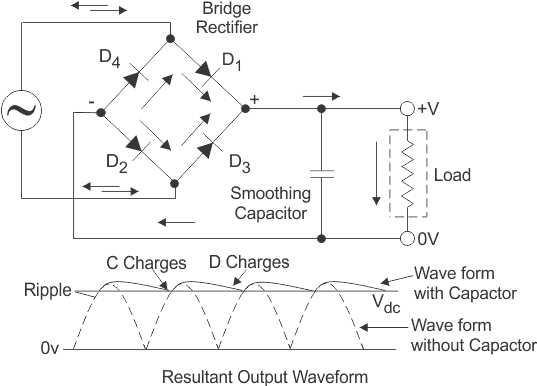


Fig.4.9

As the instantaneous voltage starts increasing the [capacitor charges](https://www.electrical4u.com/charging-a-capacitor/), it charges till the waveform reaches its peak value. When the instantaneous value starts reducing the [capacitor](https://www.electrical4u.com/what-is-capacitor/) starts discharging exponentially and slowly through the load (input of the regulator in this case). Hence, an almost constant DC value having very less ripple content is obtained.

**Chapter 5**

**LIQUID CRYSTAL DISPLAY**:

LCD stands for liquid crystal display. Character and graphical LCD’s are most common among hobbyist and DIY electronic circuit/project makers. Since their interface serial/parallel pins are defined so it’s easy to interface them with many microcontrollers. Many products that are seen in our daily life have LCD’s with them. They are used to show status of the product or provide interface for inputting or selecting some process. Washing machine, microwave, air conditioners and mat cleaners are few examples of products that have character or graphical LCD’s installed in them. In this tutorial it discuss about the character LCD’s. How they work? Their pin out and initialization commands etc.

Character LCD’s come in many sizes 8x1, 8x2, 10x2, 16x1, 16x2, 16x4, 20x2, 20x4, 24x2, 30x2, 32x2, 40x2 etc. Many multinational companies like Philips, Hitachi, and Panasonic make their own custom type of character LCD’s to be used in their products. All character LCD’s performs the same functions (display characters numbers special characters, asci characters etc.).Their programming is also same and they all have same 14 pins (0-13) or 16 pins (0 to 15).

In an mxn LCD. M denotes number of coulombs and n represents number of rows. Like if the LCD is denoted by 16x2 it means it has 16 coulombs and 2 rows. Few examples are given below. 16x2, 8x1 and 8x2 LCD are shown in the picture below. Note the difference in the rows and coulombs.

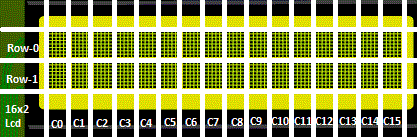


Fig.5.1

On a character LCD a character is generated in a matrix of 5x8 or 5x7. Where 5 represents number of coulombs and 7/8 represent number of rows. Maximum size of the matrix is 5x8. It cannot display character greater then 5x8 dimension matrix. Normally, display a character in 5x7 matrixes and left the 8th row for the cursor.Generally use the 8th row of the matrix for the character display, then there will be no room for cursor. The picture on the right side shows the 5x8 dot matrix pixels arrangement.  To display character greater than this dimension, it need to switch to graphical LCD’s.

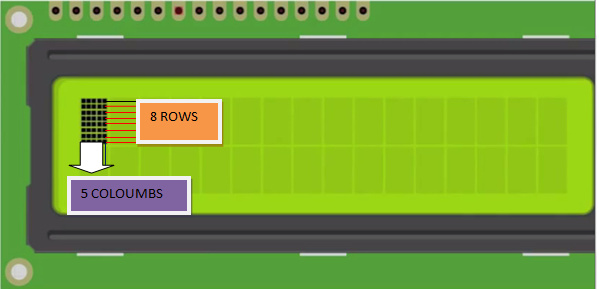


Fig.5.2

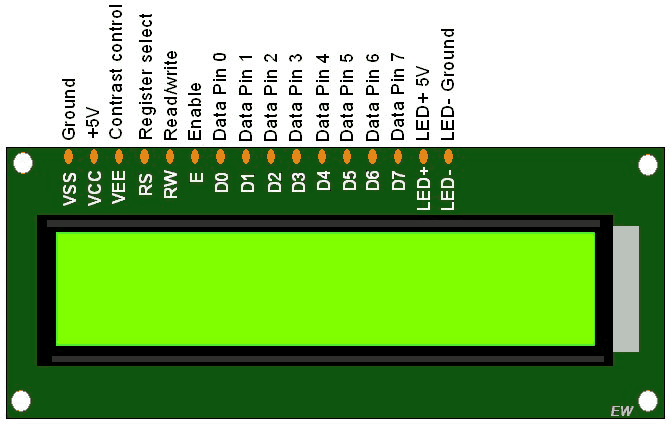


Fig.5.3

All character LCD’s have

* Eight (8) data pins D0-D7
* Vcc (Apply +5 volt here)
* Gnd (Ground this pin)
* Rc (Register select)
* Rw (read – write)
* En (Enable)
* V0 (Set LCD contrast)

The picture on the left side shows the pin out of the character LCD. Almost all the character LCD’s are composed of the same pin out. LCD’s with total pin count equal to 14 does not have back light control option. They might have back light always on or does not have a back light. 16 total pin count LCD’s have 2 extra A and K pins. A means anode and K cathode use these pins to control the back light of LCD.

Character LCD’s have a controller build in to them named HD44780. Actually talk with this controller in order to display character on the LCD screen. HD44780 must be properly handled and initialized before sending any data to it. HD44780 has some registers which are initialized and  manipulated for character displaying on the LCD. These registers are selected by the pins of character LCD.

**Rs(Register select)**  
Register select selects the HD44780 controller registers. It switches between Command and data register.

* Command Register
* Data Register

**Command Register**

When a command send to LCD these commands go to Command register and are processed there. Commands with their full description are given in the picture below. When Rs=0 command register is selected.

**Data Register**

When a send Data to LCD it goes to data register and is processed there. When Rs=1 data register is selected.

**Rw(Read – Write)**

Rw pin is used to read and write data to HD44780 data and command registers. When Rw=1, it can read data from LCD. When Rw=0 can write to LCD.

**En(Enable signal)**

When the register Rs(Command and Data) is select and set Rw(read -  write) and placed the raw value on 8-data lines, now it’s time to execute the instruction. By instruction, it mean the 8-bit data or 8-bit command present on Data lines of LCD. For sending the final data/command present on the data lines use this enable pin. Usually it remains en=0 and when it want to execute the instruction make it high en=1 for some mills seconds. After this,again make it ground en=0.

**V0 (Set LCD contrast)**

To set LCD display sharpness use this pin. Best way is to use variable resistor such as potentiometer a variable current makes the character contrast sharp. Connect the output of the potentiometer to this pin. Rotate the potentiometer knob forward and backward to adjust the LCD contrast.

**NOTE:** It cannot send an integer, float, long, double type data to LCD because LCD is designed to display a character only. Only the characters that are supported by the HD44780 controller. See the HD44780 data sheet to find out what characters can display on LCD.  The 8 data pins on led carries only Ascii 8-bit code of the character to LCD. However it can convert our data in character type array and send one by one our data to LCD. Data can be sent using LCD in 8-bit or 4-bit mode. If 4-bit mode is used, two nibbles of data (First high four bits and then low four bits) are sent to complete a full eight-bit transfer. 8-bit mode is best used when speed is required in an application and at least ten I/O pins are available. 4-bit mode requires a minimum of seven bits. In 4-bit mode, only the top 4 data pins (4-7) are used.

**ThingSpeaks:**

•ThingSpeak is a Web Service (REST API) that lets to collect and store sensor data in the cloud and develop Internet of Things applications. • It works with Arduino, Raspberry Pi and MATLAB (premade libraries and APIs exists). • But it should work with all kind of Programming Languages, since it uses a REST API and HTTP.

## ThingSpeak:

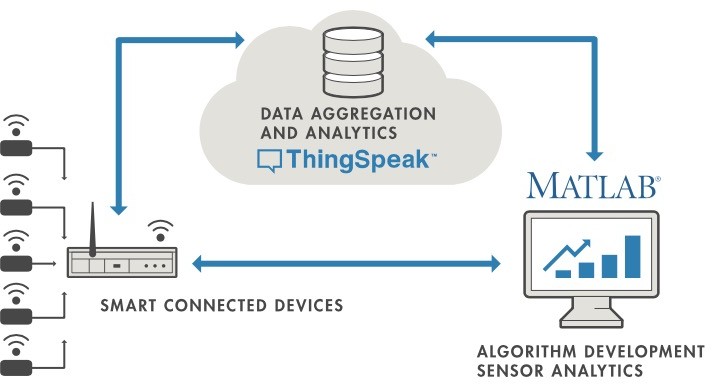


Fig.5.4

ThingSpeak:

ThingSpeak is an IoT analytics platform service that lets to collect and store sensor data in the cloud and develop Internet of Things applications.

The ThingSpeak service also perform online analysis and act on data. Sensor data can be sent to ThingSpeak from any hardware that can communicate using a REST API

ThingSpeak is a Web Service (REST API) that collects and store sensor data in the cloud and develop Internet of Things applications.

ThingSpeak + Arduino:

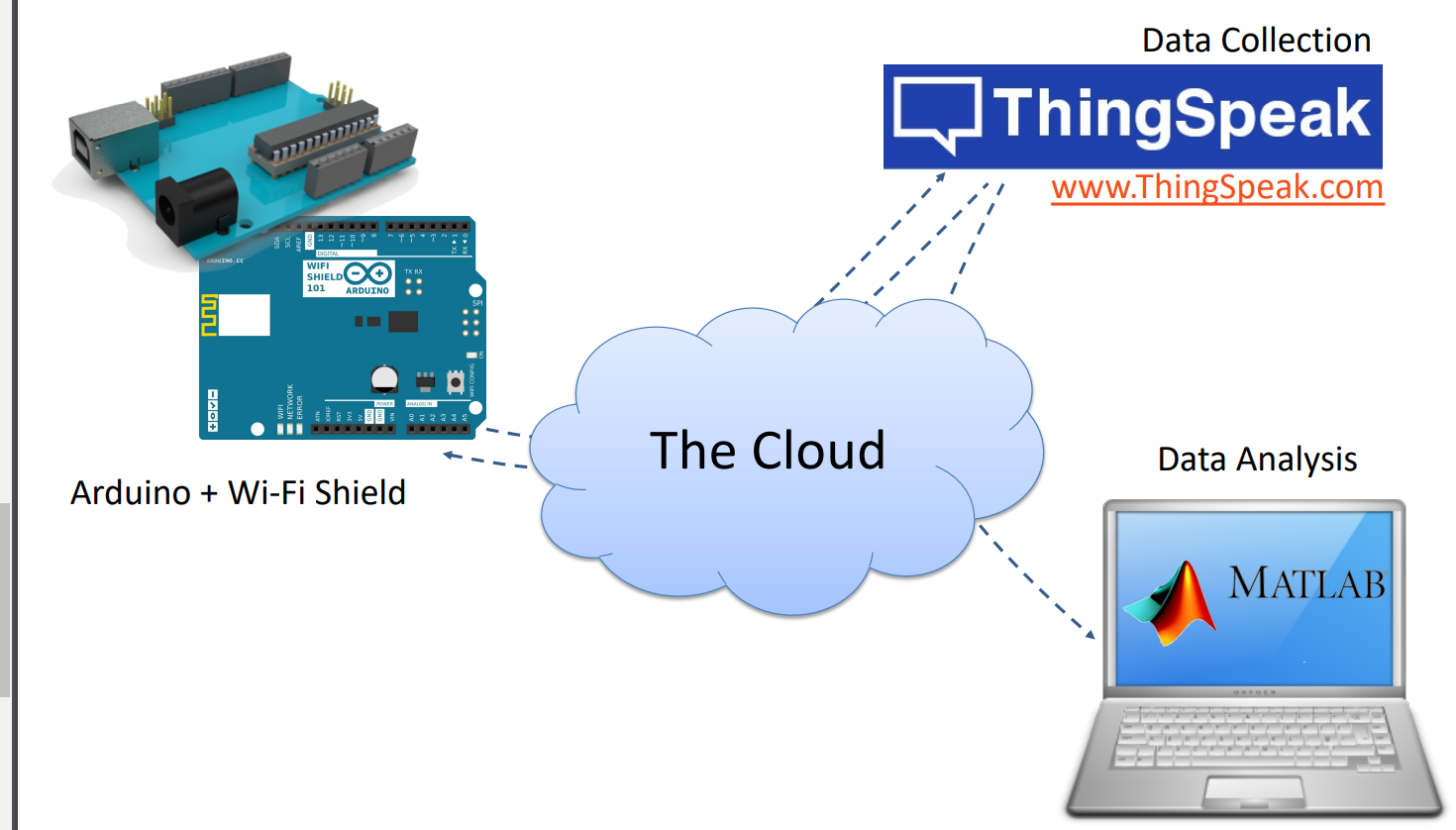
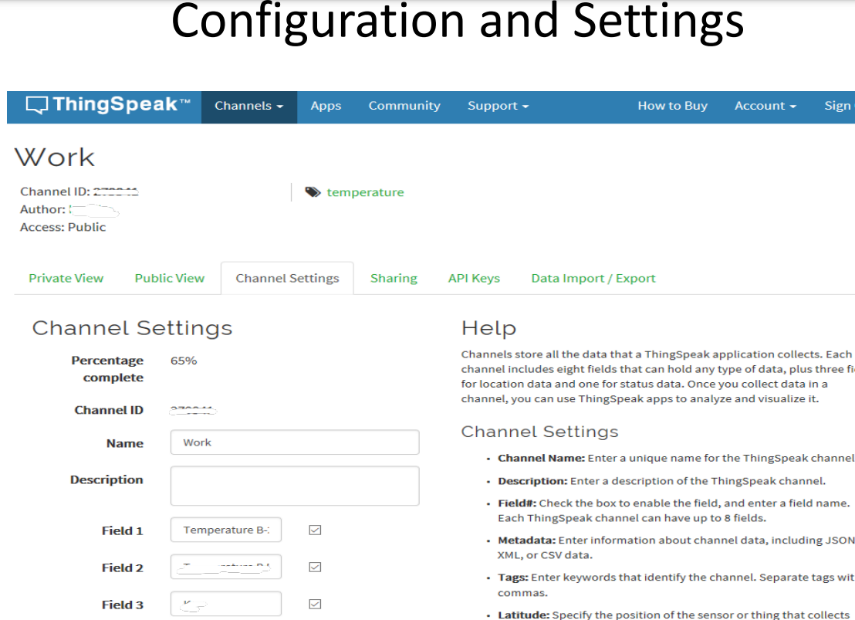


Fig.5.5

ThingSpeak + C#:



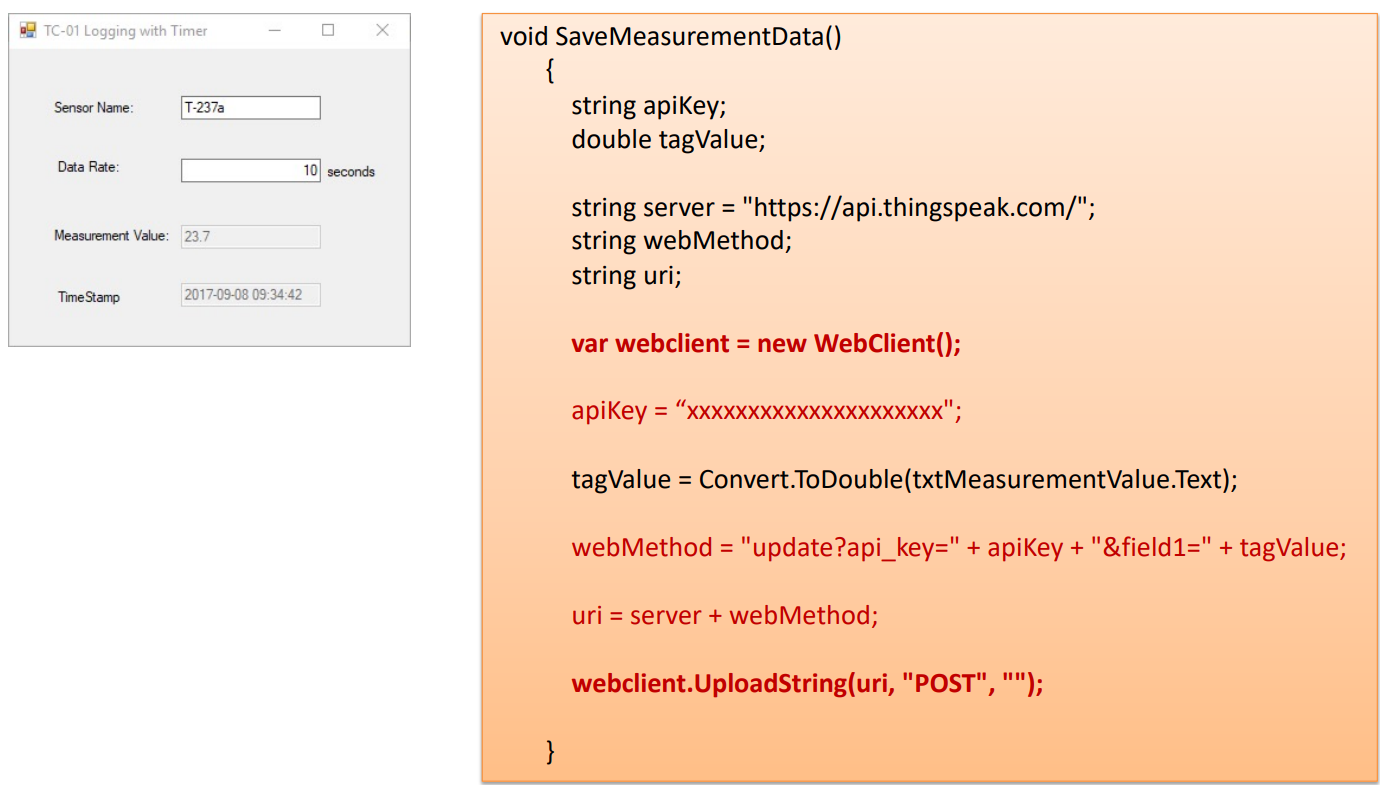


Fig.5.6

**RESULTS & DISCUSSION:**

**Analytical results of the proposed system**

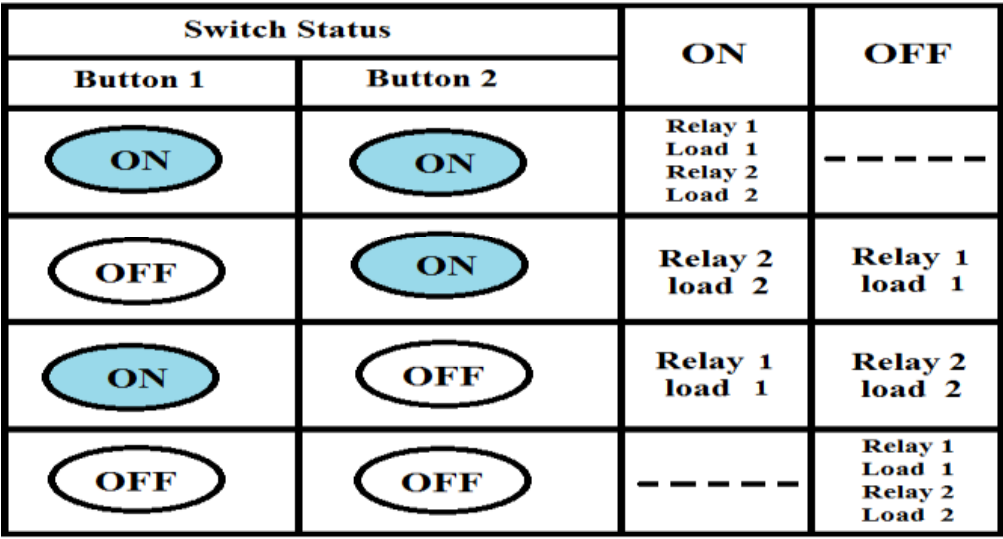
****

Fig.5.7

**Case Study:**

**Case 1:** When both the switches are ON, then the ESP 8266sends 3.3V to both the channels. Hence, the resultant effect of this action is seen on the bulbs i.e., both blubs are glowing.

**Case 2:** When clicked on switch 1, the ESP 8266 sends 0 V to IN1 of the first channel of relay which leads to a power cut at Load 1. Hence the Load 1 is turned OFF.

**Case 3:** When clicked on switch 2 ESP 8266 sends 0 V to IN2 of the second channel of relay. Thus Load 2 is turned OFF.

**Case 4:** When both switches are off, ESP 8266 sends 0V to both the channels. Hence resultant effect is seen on both Loads i.e. Both Load 1 & Load 2 are OFF.

Result Snapshot:

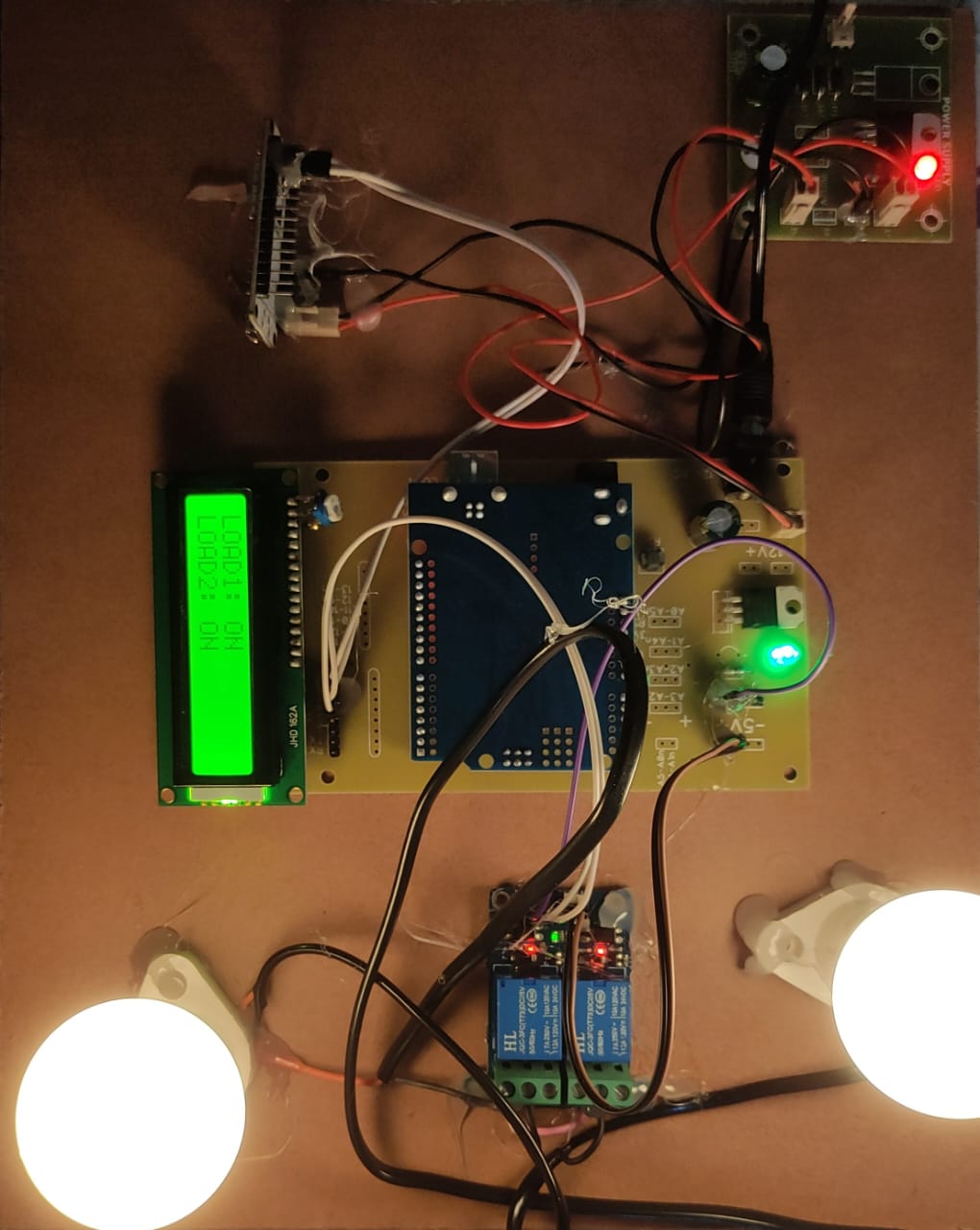


Fig.5.8

**CONCLUSION**

Thus, with the implementation of IOT based circuit breaker the safety of the line man is ensured while working on high power lines by manually controlling the circuit system using ThingSpeak application. The overall circuit breaker performance is maintained while reducing maintenance costs and unnecessary downtime. Also, the restriction of the unauthorized and non-interference of the outsiders with the supply circuit can be achieved. In this way it can proposed a smart circuit breaker by making it very reliable for the user to handle it using IoT.

**FUTURE SCOPE**

A modification to this project is possible developing a webpage that can be accessed by the substation personal anywhere. Proposed system can be enhanced by using an EPROM for user to change the password for a more secured system. It can be interfaced with a GSM Modem for remote controlling of the circuit breaker via SMS.

**REFERENCES:**

[1] Mane Kirti M., Attar Arifa U, Dandile Aishwarya A., Ghogale Pragati S.,Prof. Jagtap Sujit P.5, "Password Based Circuit Breaker", International Journal for Research in Applied science and Engineering Technology (IJRASET),Volume 6 Issue IV, April 2018, pp. 2229-2233.

[2] Bhagwan Kharat, Durvankur Sarwade, Dhananjay Bidgar, Prof. Bhushan Kadu,“Internet of Thing (I.O.T) Base Controlling & Monitoring of Circuit Breaker”, International Research Journal of Engineering and Technology (IRJET), Volume: 04 Issue: 05 | May - 2017,pp. 668-673.

[3] Abhijit Das & Dr. K. Kalimuthu, S.S. Biswas, “IOT Based Circuit Breaker Monitoring& Control”, International Journal of Applied Engineering Research, Volume 13, Number10 (2018), pp. 7806-7810.

[4] Lakshmi Devdas, Sreelakshmi V.G, Amal Kottor, Jinendhu Justin, “Automated Circuit Breaker with IOT Interfacing”, IOSR Journal of Engineering, Issue-2018, pp.29-32.

[5] Athira Nair, Josephin J, Anjana A S, Athira C P, “Electric Line Man Safety System with OTP Based Circuit Breaker”, IJRET, April 2015, pp 23-26.