IOT BASED HOME AUTOMATION CIRCUIT USING SOLID STATE RELAY

Submitted to

CHHATTISGARH SWAMI VEVEKANAND TECHNICAL UNIVERSITY BHILAI, (C.G.)

In partial fulfilment for the award of the degree

of

Bachelor of Engineering

in

ELECTRICAL ENGINEERING

By

Geetanjali Nishadraj (Roll No. 300702418006)
Pankaj Kumar Soni (Roll No. 300702418013)
Anjali Madhukar (Roll No. 300702418019)
Abhinay Dev Berma (Roll No. 300702418047)
Ambika Prajapati (Roll No. 300702418049)

Under the Guidance of

Prof. S. K. Dewangan

Department of Electrical Engineering

GOVERNMENT ENGINEERING COLLEGE BILASPUR
Koni, Bilaspur, C.G.

Session: 2021-22

DECLARATION BY THE STUDENT

I the undersigned solemnly declare that the report of the project work entitled **Home Automation Circuit using Solid State Relay**, is based my own work carried out during the course of my study under the supervision of **Prof. S. K. Dewangan.**

I assert that the statements made and conclusions drawn are an outcome of the project work. I further declare that to the best of my knowledge and belief that the report does not contain any part of any work which has been submitted for the award of any other degree/diploma/certificate in this University of any other University.

Geetanjali Nisahdraj

Roll No. – **300702418006**

Enrolment No. – **BF2045**

Pankaj Kumar Soni

Roll No. – **300702418013**

Enrolment No. – **BF2052**

Anjali Madhukar

Roll No. – **300702418019**

Enrolment No. – **BF2058**

Abhinay Dev Berma

Roll No. – **300702418047**

Enrolment No. – **BF2086**

Ambika Prajapati

Roll No. – **300702418049**

Enrolment No. – **BF2088**

CERTIFICATE BY THE SUPERVISIOR

This is to certify that the report of the project submitted is an outcome of the project work entitled **Home Automation Circuit using Solid State Relay** carried out by

Geetanjali Nishadraj	**Roll No. 300702418006**	**Enrolment No.: BF2045**
Pankaj Kumar Soni	**Roll No. 300702418013**	**Enrolment No.: BF2052**
Anjali Madhukar	**Roll No. 300702418019**	**Enrolment No.: BF2058**
Abhinay Dev Berma	**Roll No. 300702418047**	**Enrolment No.: BF2086**
Ambika Prajapati	**Roll No. 300702418049**	**Enrolment No.: BF2088**

Carried out under my guidance and supervision for the award of Degree in Bachelor of Engineering in **Electrical Engineering** of Chhattisgarh Swami Vivekanand Technical University, Bhilai (C.G.), India.

To the Best of my knowledge the report

i) Embodies the work of the candidate him/herself,

ii) Has duly been completed,

iii) Fulfils the requirement of the Ordinance relating to the BE degree of the University and

iv) Is up to the desired standard for the purpose of which is submitted.

(Signature of Guide)

Name: Prof. S.K. Dewangan

Designation: **Assistant Professor**

Department: **Electrical Engineering**

The Project work as mentioned above is hereby being recommended and forwarded for examination and evaluation.

_____

(Signature of HOD with seal)

_____

(Seal of the Department Electrical Engineering)

CERTIFICATE BY THE EXAMINERS

This is to certify that the project report entitled

IOT BASED HOME AUTOMATION CIRCUIT USING SOLID STATE RELAY

which is
submitted by

1. Geetanjali Nishadraj	**Roll No. 300702418006**	**Enrolment No.: BF2045**
2. Pankaj Kumar Soni	**Roll No. 300702418013**	**Enrolment No.: BF2052**
3. Anjali Madhukar	**Roll No. 300702418019**	**Enrolment No.: BF2058**
4. Abhinay Dev Berma	**Roll No. 300702418047**	**Enrolment No.: BF2086**
5. Ambika Prajapati	**Roll No. 300702418049**	**Enrolment No.: BF2088**

has been examined by the undersigned as a part of the examination for the award of Bachelor of Engineering degree in Electronics & Telecommunication Engineering of Chhattisgarh Swami Vivekanand Technical University, Bhilai.

(Signature of the Internal Examiner)	(Signature of the External Examiner)
(Name of the Internal Examiner)	(Name of the Internal Examiner)
Date:	Date:
Designation:	Designation:
Institute:	Institute:.

A C K N O W L E D G E M E N T

A completion with project has been a source of great practical experience and personal pride for us and we take this opportunity to thank to all those who has been us in our endeavor.

I would like to say thank **to Dr. B. S. CHAWLA** principal of , Government Engineering College Bilaspur to facilities my experimental work.

I would like to say thank **to Dr. A. K. SHUKLA** Head of Electrical Engineering department, Government Engineering College Bilaspur to facilities my experimental work.

I would also like to thanks **Prof. S. K. DEWANGAN** professor Electrical Engineering department, Government Engineering College Bilaspur, to provide facilities to carry out the experimental works.

I cannot close this prefatory remark without expressing my deep sense of gratitude and reverence to the various papers. I have used and referred to in order to complete my research work.

Geetanjali Nisahdraj

Roll No. – **300702418006**

Enrolment No. – **BF2045**

Pankaj Kumar Soni

Roll No. – **300702418013**

Enrolment No. – **BF2052**

Anjali Madhukar

Roll No. – **300702418019**

Enrolment No. – **BF2058**

Abhinay Dev Berma

Roll No. – **300702418047**

Enrolment No. – **BF2086**

Ambika Prajapati

Roll No. – **300702418049**

Enrolment No. – **BF2088**

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

CHAPTER-I

INTRODUCTION

1. INTRODUCTION

Home automation is the automatic control of electronic devices in your home. These devices are connected to the Internet, which allows them to be controlled remotely. With home automation, devices can trigger one another so you don't have to control them manually via an app or voice assistant. For example, you can put your lights on schedules so that they turn off when you normally go to sleep, or you can have your thermostat turn the A/C up about an hour before you return to work so you don't have to return to a stuffy house. Home automation makes life more convenient and can even save you money on heating, cooling and electricity bills. Home automation can also lead to greater safety with Internet of Things devices like security cameras and systems.

1.1. Internet of Things vs. Home Automation

The Internet of Things, commonly known as IoT, refers to any device that's connected to the Internet that isn't normally; for example, a smart light bulb that you can turn on and off via an app. All home automation devices are IoT devices, which can be automated to trigger one another. So, while IoT refers to the devices themselves, home automation is what you can do with the IoT devices to make your life just a tad bit easier.

1.2. Solid State Relay

A solid-state relay (SSR) is an electronic switching device that switches on or off when an external voltage (AC or DC) is applied across its control terminals. It serves the same function as an electromechanical relay, but has no moving parts and therefore results in a longer operational lifetime. SSRs consist of a sensor which responds to an appropriate input (control signal), a solid-state electronic switching device which switches power to the load circuitry, and a coupling mechanism to enable the control signal to activate this switch without mechanical parts. The relay may be designed to switch either AC or DC loads.

Packaged solid-state relays use power semiconductor devices such as thyristors, triac and transistors, to switch currents up to around a hundred amperes. Solid-state relays have fast switched speeds compared with electromechanical relays, and have no physical contacts to wear out. Users of solid-state relays must take into consideration an SSR's inability to withstand a large momentary overload the way an electromechanical relay can, as well as their higher "on" resistance.

1.3. Triac

A TRIAC is a bidirectional, three-electrode AC switch that allows electrons to flow in either direction. It is the equivalent of two SCRs connected in a reverse-parallel arrangement with gates connected to each other.

A TRIAC is triggered into conduction in both directions by a gate signal like that of an SCR. TRIACs were designed to provide a means for the development of improved AC power controls.

TRIACs are available in a variety of packaging arrangements. They can handle a wide range of current and voltage. TRIACs generally have relatively low-current capabilities compared to SCRs — they are usually limited to less than 50 A and cannot replace SCRs in high-current applications.

TRIACs are considered versatile because of their ability to operate with positive or negative voltages across their terminals. Since SCRs have a disadvantage of conducting current in only one direction, controlling low power in an AC circuit is better served with the use of a TRIAC.

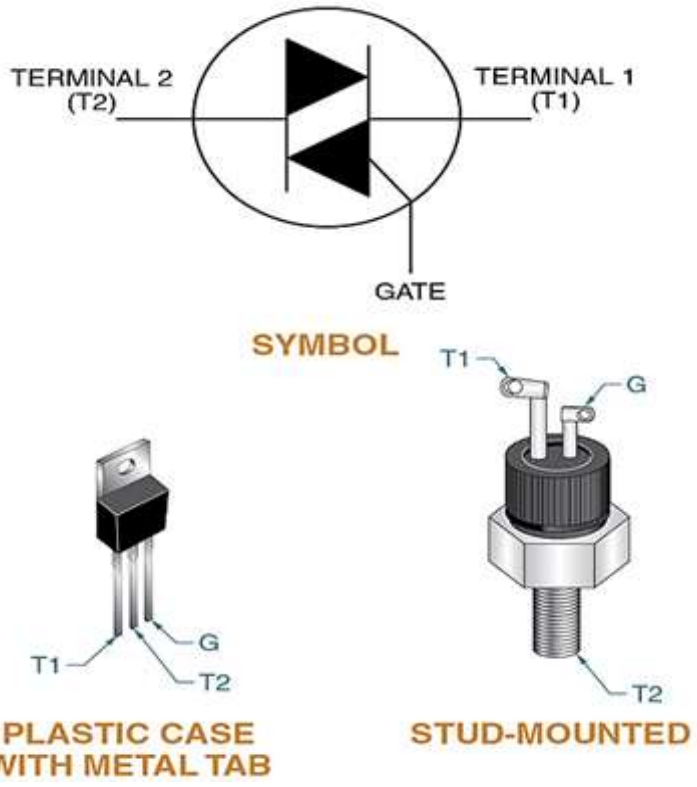


Figure 1.3.1. ***TRIAC terminals include a gate, terminal 1 (T1), and terminal 2 (T2).***

1.4. The Characteristic Curve of TRIACs

The characteristics of a TRIAC are based on T1 as the voltage reference point. The polarities shown for voltage and current are the polarities of T2 with respect to T1. The polarities shown for the gate are also with respect to T1. See Figure 1.2.

Again, the TRIAC may be triggered into conduction in either direction by a gate current (IG) of either polarity.

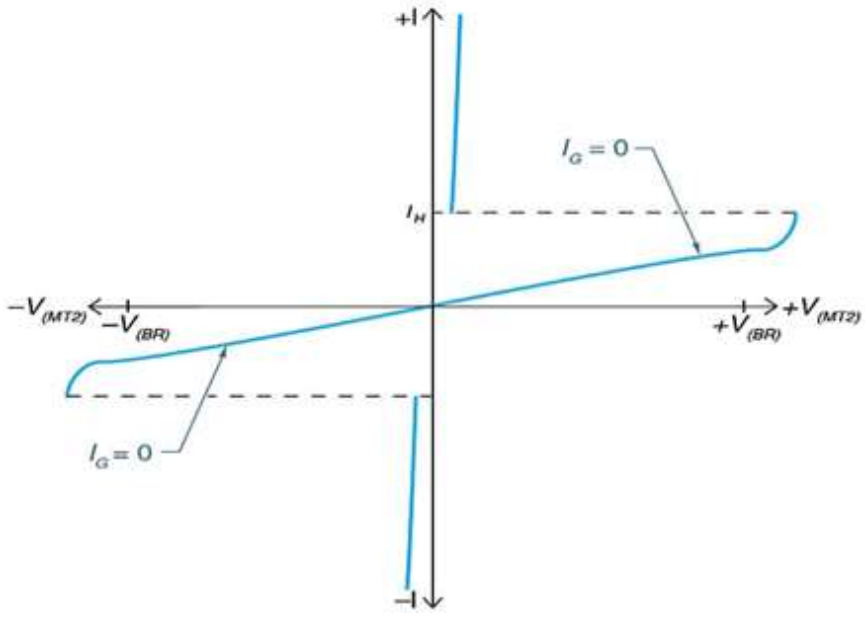


Figure 1.4.1. ***A TRIAC characteristic curve shows the characteristics of a TRIAC when triggered into conduction.***

1.5. Zero Crossing Triac Drivers

The zero-cross family of optically isolated triac drivers is an inexpensive, simple and effective solution for interface applications between low current dc control circuits such as logic gates and microprocessors and ac power loads (120, 240 or 380 volts, single or 3-phase).

These devices provide sufficient gate trigger current for high current, high voltage thyristors, while providing a guaran- teed 7.5 kV dielectric withstand voltage between the line and the control circuitry. An integrated, zero-crossing switch on the detector chip eliminates current surges and the resulting electromagnetic interference (EMI) and reliability problems for many applications. The high transient immunity of 5000 V/μs, combined with the features of low coupling capacitance, high isolation resistance and up to 800 volt specified V DRM ratings qualify this triac driver family as the ideal link between sensitive control circuitry and the ac power system environment.

Optically isolated triac drivers are not intended for standalone service as are such devices as solid-state relays. They will, however, replace costly and space demanding discrete drive circuitry having high component count consisting of standard transistor opt isolators, support components including a full wave rectifier bridge, discrete transistor, trig- ger SCRs and various resistor and capacitor combinations.

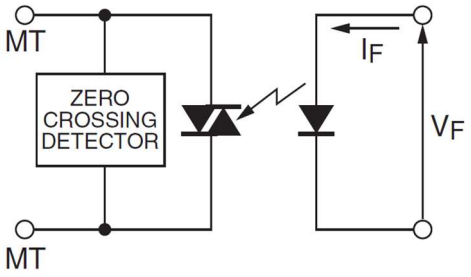


Figure 1.5.1. ***Simplified Schematic of Isolator.***

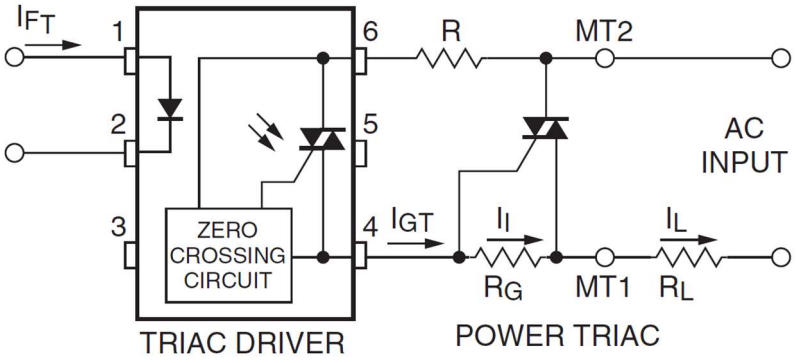


Figure 1.5.2. ***Basic Driving Circuit Triac Driver, Triac and Load.***

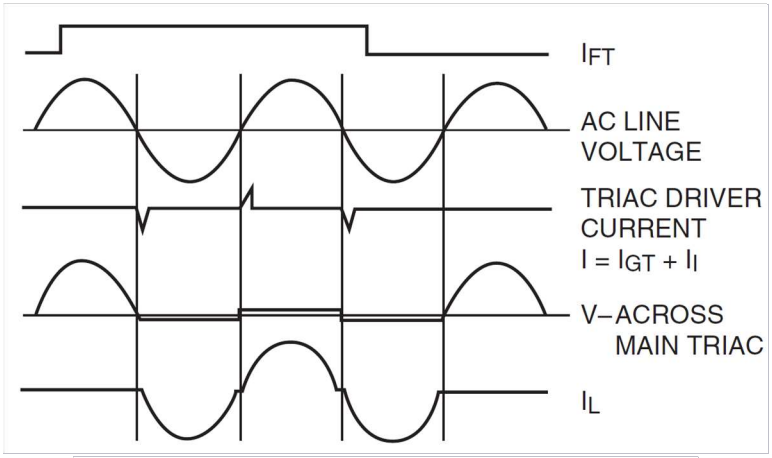


Figure 1.5.2. ***Waveforms of a Basic Driving Circuit.***

1.6. IoT Devices

IoT devices are the nonstandard computing devices that connect wirelessly to a network and have the ability to transmit data, such as the many devices on the internet of things (IoT).

IoT involves extending internet connectivity beyond standard devices, such as desktops, laptops, smartphones and tablets, to any range of traditionally "dumb" or non-internet-enabled physical devices and everyday objects. Embedded with technology, these devices can communicate and interact over the internet. They can also be remotely monitored and controlled.

Example: Connected devices are part of an ecosystem in which every device talks to other related devices in an environment to automate home and industry tasks. They can communicate usable sensor data to users, businesses and other intended parties. The devices can be categorized into three main groups: consumer, enterprise and industrial.

Consumer connected devices include smart TVs, smart speakers, toys, wearables and smart appliances.

In a smart home, for example, devices are designed to sense and respond to a person's presence. When a person arrives home, their car communicates with the garage to open the door. Once inside, the thermostat is already adjusted to their preferred temperature, and the lighting is set to a lower intensity and color, as their smart watch data indicates it has been a stressful day. Other smart home devices include sprinklers that adjust the amount of water given to the lawn based on the weather forecast and robotic vacuum cleaners that learn which areas of the home must be cleaned most often.

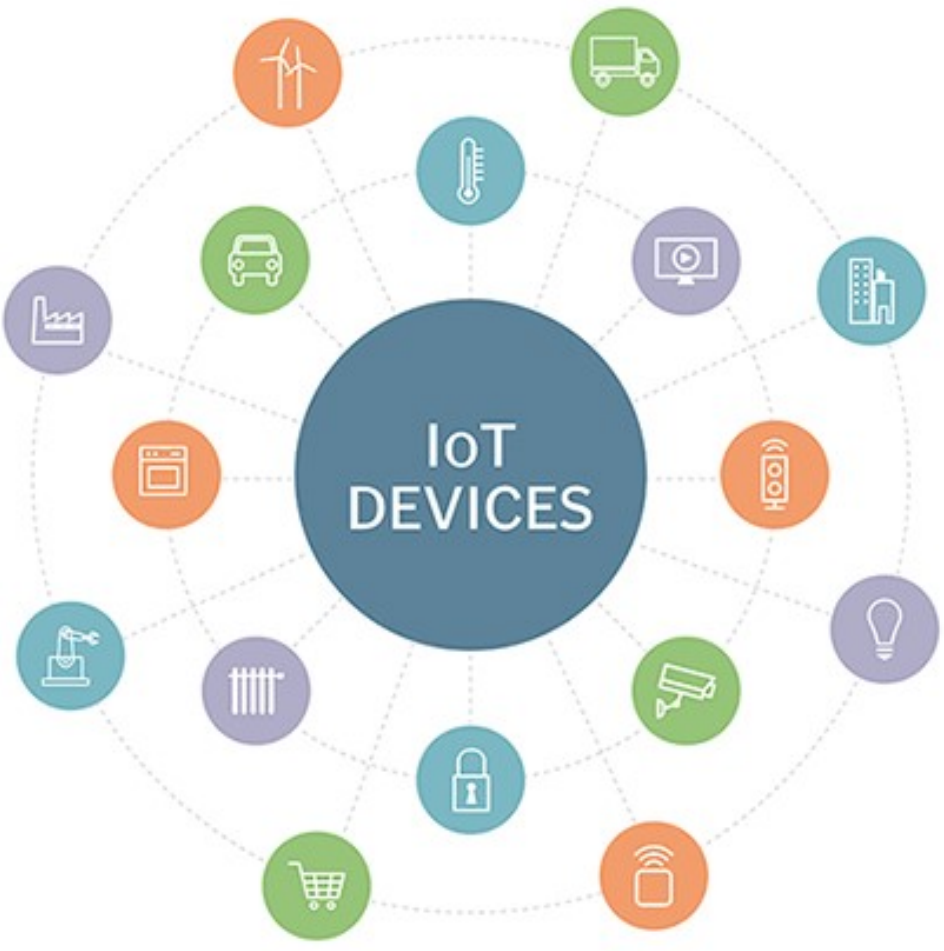


Figure 1.6.1. ***Various IoT Devices.***

1.7. HTML, CSS and Java Script

HTML, CSS, and JavaScript are the main languages of the Web. As a web developer, the three main languages we use to build websites are HTML, CSS, and JavaScript. JavaScript is the programming language, we use HTML to structure the site, and we use CSS to design and layout the web page. These days, CSS has become more than just a design language, though. You can actually implement animations and smooth transitions with just CSS. In fact, you can do some basic programming with CSS too. An example of this is when you use media queries, where you define different style rules for different kinds of screens (resolutions). JavaScript has also grown beyond being used just in the browser as well. We now use it on the server thanks to Node.js.

Together, we use these three languages to format, design, and program web pages. And when you link together some web pages with hyperlinks, along with all their assets like images, videos, and so on that are on the server computer, it gets rendered into a website. This rendering typically happens on the front end, where the users can see what's being displayed and interact with it. On the other hand, data, especially sensitive information like passwords, are stored and supplied from the back end part of the website. This is the part of a website which exists only on the server computer, and isn't displayed on the front-end browser. There, the user cannot see or readily access that information.

1.7.1. HTML

HTML stands for **Hyper Text Markup Language**.

Markup is defined as: *a set of detailed instructions, usually written on a manuscript to be typeset, concerning style of type, makeup of pages, and the like.*

So, you can think of HTML as the language used for creating detailed instructions concerning style, type, format, structure and the makeup of a web page before it gets printed (shown to you). But in the context of web development, we can replace the term ‘printed’ with ‘rendered’ as a more accurate term. HTML helps you structure your page into elements such as paragraphs, sections, headings, navigation bars, and so on.

1.7.2. CSS

While HTML is a **markup language** used to format/structure a web page, CSS is a **design language** that you use to make your web page look nice and presentable. CSS stands for Cascading Style Sheets, and you use it to improve the appearance of a web page. By adding thoughtful CSS styles, you make your page more attractive and pleasant for the end user to view and use. Imagine if human beings were just made to have skeletons and bare bones – how would that look? Not nice if you ask me. So, CSS is like our skin, hair, and general physical appearance. You can also use CSS to layout elements by positioning them in specified areas of your page.

1.7.3. Java Script

Now, if HTML is the **markup language** and CSS is the **design language**, then JavaScript is the **programming language**. JavaScript is a scripting or programming language that allows you to implement complex features on web pages every time a web page does more than just sit there and display static information for you to look at displaying timely content updates, interactive maps, animated 2D/3D graphics, scrolling video jukeboxes, etc.

you can bet that JavaScript is probably involved. It is the third layer of the layer cake of standard web technologies, two of which (HTML and CSS) we have covered in much more detail in other parts of the Learning Area.

CHAPTER-II

LIST OF COMPONENTS

2. LIST OF COMPONENTS

Sr. No.	Name of Component
1	BTB16-800B 16A TRIAC
2	BT136 4A TRIAC
3	MOC3021 Triac Driver IC
4	Plastic Film Capacitor
5	Carbon Film Resistor
6	ESP32 Microcontroller
7	TTP223 Touch Sensor
8	ACS712 Current Sensor
9	ZMPT101 Voltage Sensor
8	JST Connectors
9	Screw Terminals
10	Zero PCB Board

Table 2.0.1. ***List of Component.***

2.1. BTB16-800B 16A TRIAC

The UTC BTB16 is a 16A triacs which can be operated in 4 quadrants, it uses UTC's advanced technology to provide customers with high commutation performances.

The UTC BTB16 is suitable for AC switching application and phase control application such as fan speed and temperature modulation control, lighting control and static switching relay, either in through-hole or surface-mount packages.

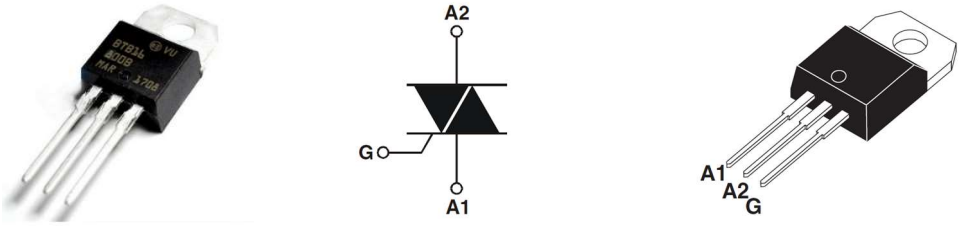


Figure 2.1.1. ***BTB16-800B and its Pinout.***

ELECTRICAL CHARACTERISTICS (T_J =25°C unless otherwise specified.):

PARAMETER	SYMBOL	TEST CONDITIONS		C			B			UNIT
				MIN	TYP	MAX	MIN	TYP	MAX	
Gate Trigger Current (Note 1)	I_{GT}	V_D=12V, R_L=33Ω	I-II-III			25			50	mA
			IV			50			100	mA
Gate Trigger Voltage	V_{GT}		ALL			1.3			1.3	V
Gate Non-Trigger Voltage	V_{GD}	V_D=V_{DRM}, R_L=3.3kΩ, T_J=125°C	ALL	0.2			0.2			V
Holding Current (Note 2)	I_H	I_T=500mA				25			50	mA
Latching Current	I_L	I_G=1.2I_{GT}	I-III-IV			40			60	mA
			II			80			120	mA
Critical Rate of Rise of Off-State Voltage (Note 2)	dV/dt	V_D=67%V_{DRM}, Gate Open, T_J=125°C		200			400			V/µs
Critical Rate of Rise of Off-State Voltage at Commutation(Note 2)	(dV/dt)c	(dI/dt)c=7A/ms, T_J= 125°C		5			10			V/µs

Table 2.1.1. ***Electrical Characteristics of BTB16 Triac.***

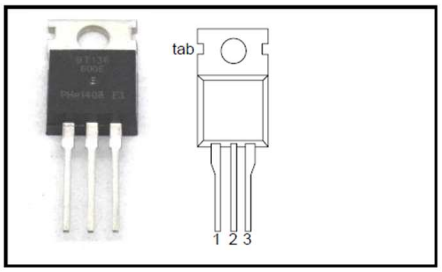
2.2. BTB136 4A TRIAC

The BT136 is TRIAC with 4A maximum terminal current. The gate threshold voltage of the BT136 is also very less so can be driven by digital circuits. Since TRIACs are bi-directional switching devices they are commonly used for switching AC applications. Passivated triacs in a plastic envelope, intended for use in applications requiring high bidirectional transient and blocking voltage capability and high thermal cycling performance. Typical applications include motor control, industrial and domestic lighting, heating and static switching.

PINNING - TO220AB

PIN	DESCRIPTION
1	main terminal 1
2	main terminal 2
3	gate
tab	main terminal 2

PIN CONFIGURATION



SYMBOL

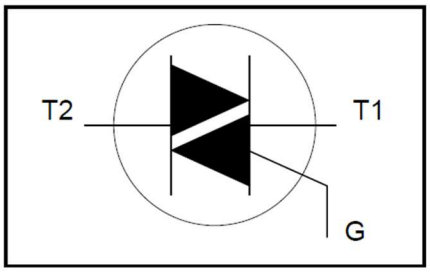


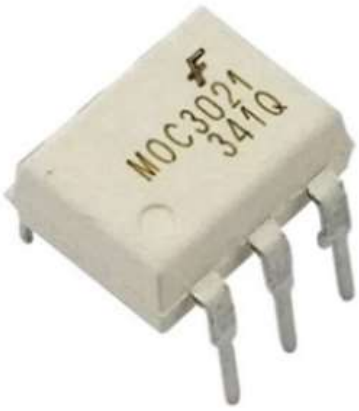
Figure 2.2.1. ***BT136 and its Pinout.***

2.3. MOC3021 TRIAC Driver IC

The MOC3021 is a Zero-Crossing TRIAC driven Optocoupler or Optoisolator. As we know the term Optocoupler/optoisolater means the same that is we use light to indirectly couple to sets of circuits. The specialty of MOC3021 is that it has Zero-Crossing ability and is driven by a Triac.

Since the output is driven by a TRIAC, we can drive loads up to 400V and the triac can conduct in both directions hence controlling AC loads will not be a problem. Also, since it has zero-crossing ability, when the AC load is switched on for the first time the TRIAC will start conducting only after the AC wave reaches 0V this way we can avoid direct peak voltages to the Load and thus preventing it from getting damaged. It also has a decent rise and fall time and hence can be used to control the output voltage.

This features of MOC3021 makes it an ideal choice for controlling high voltage AC loads through digital controllers like MPU/MCU. Since the output is controlled, we can control the intensity of the light or the speed of a AC motor. So, if you are looking for a opto-isolator to control a AC application through DC then this IC might be the right choice for you.



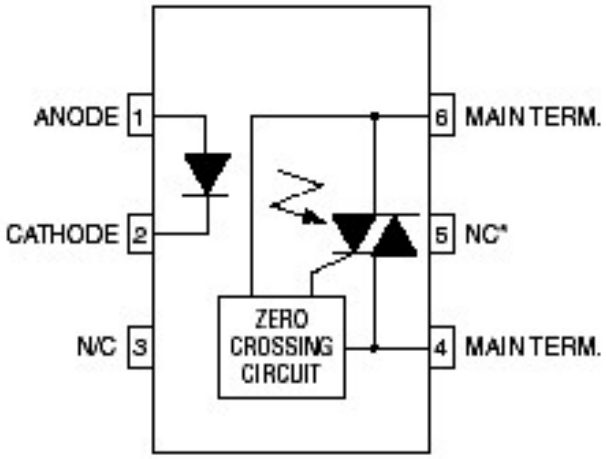


Figure 2.3.1. ***MOC3021 and its Pinout.***

MAXIMUM RATINGS (T_A = 25°C unless otherwise noted)

Rating	Symbol	Value	Unit
INFRARED EMITTING DIODE			
Reverse Voltage	V_R	3	Volts
Forward Current — Continuous	I_F	60	mA
Total Power Dissipation @ T_A = 25°C Negligible Power in Triac Driver Derate above 25°C	P_D	100 1.33	mW mW/°C
OUTPUT DRIVER			
Off–State Output Terminal Voltage	V_{DRM}	400	Volts
Peak Repetitive Surge Current (PW = 1 ms, 120 pps)	I_{TSM}	1	A
Total Power Dissipation @ T_A = 25°C Derate above 25°C	P_D	300 4	mW mW/°C
TOTAL DEVICE			
Isolation Surge Voltage(1) (Peak ac Voltage, 60 Hz, 1 Second Duration)	V_{ISO}	7500	Vac(pk)
Total Power Dissipation @ T_A = 25°C Derate above 25°C	P_D	330 4.4	mW mW/°C
Junction Temperature Range	T_J	−40 to +100	°C
Ambient Operating Temperature Range	T_A	−40 to +85	°C
Storage Temperature Range	T_{stg}	−40 to +150	°C
Soldering Temperature (10 s)	T_L	260	°C

Table 2.3.1. ***Electrical Characteristics of MOC3021 Triac Driver.***

2.4. Plastic Film Capacitor

Film capacitors, plastic film capacitors, film dielectric capacitors, or polymer film capacitors, generically called film caps as well as power film capacitors, are electrical capacitors with an insulating plastic film as the dielectric, sometimes combined with paper as carrier of the electrodes.

The dielectric films, depending on the desired dielectric strength, are drawn in a special process to an extremely thin thickness, and are then provided with electrodes. The electrodes of film capacitors may be metallized aluminum or zinc applied directly to the surface of the plastic film, or a separate metallic foil. Two of these conductive layers are wound into a cylinder-shaped winding, usually flattened to reduce mounting space requirements on a printed circuit board, or layered as multiple single layers stacked together, to form a capacitor body. Film capacitors, together with ceramic capacitors and electrolytic capacitors, are the most common capacitor types for use in electronic equipment, and are used in many AC and DC microelectronics and electronics circuits.

Advantages:

- Polypropylene film capacitors can qualify for Class 1 applications
- Very low dissipation factors (tan δ), high quality factors (Q) and low inductance values (ESL)

- No microphonics compared with ceramic capacitors
- Metallized construction has self-healing properties
- High rated voltages, up to the range of kV possible
- Much higher ripple current, compared with electrolytic capacitors
- Much lower aging, compared with electrolytic capacitors of similar values
- High and very high surge current pulses possible

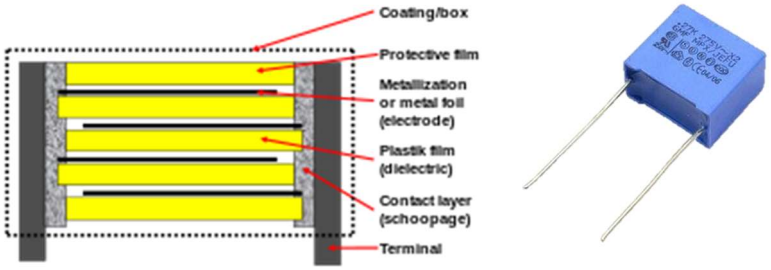


Figure 2.4.1. ***Plastic Film Capacitor.***

2.5. Carbon Film Resistor

A carbon film resistor is a type of film resistor, which is also called thermal decomposition carbon film resistor. It's a thin-film resistor in which the carbon that is thermally decomposed from a hydrocarbon at high temperatures is deposited on a substrate. It adopts high-temperature vacuum coating technology to tightly attach carbon to the surface of a porcelain rod to form a carbon film, and coats the surface with an epoxy resin to protect the film. Its surface is often coated with green protective paint. The thickness of the carbon film determines the resistance value. And usually, the thickness of the film and the groove are used to control the resistor. Besides, it has a low price, stable performance, a wide range of resistance and power.

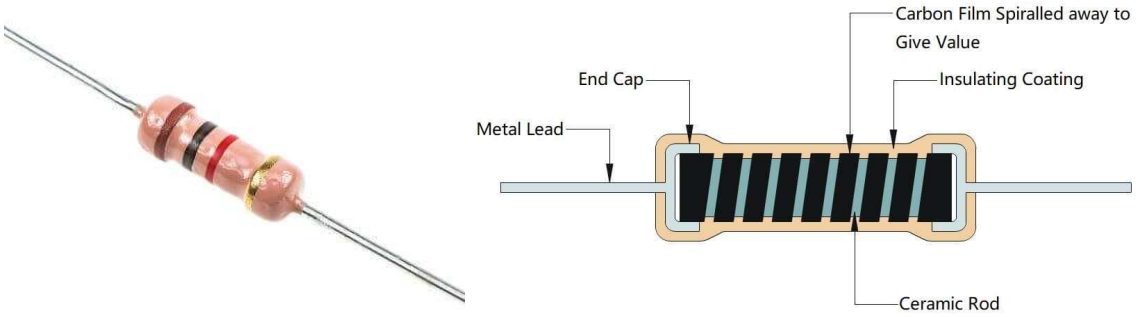


Figure 2.5.1. ***Carbon Film Resistor.***

2.6. ESP32 Microcontroller

Espressif Systems ESP32-WROOM-32 MCU Modules are powerful and generic Wi-Fi/BT/BLE MCU modules that target a wide variety of applications. These modules target applications ranging from low-power sensor networks to the most demanding tasks, such as voice encoding, music streaming, and MP3 decoding. The ESP32-WROOM-32U module is different from ESP32-WROOM-32D as it integrates a UFL connector which needs to be connected to an external IPEX antenna.

The ESP32-WROOM-32 MCU Modules use the ESP32-D0WD chip at its core. This SoC features two CPU cores that can be individually controlled, and the CPU clock frequency is adjustable from 80MHz to 240MHz. The ESP32 integrates a rich set of peripherals, ranging from capacitive touch sensors, Hall sensors, SD card interface, Ethernet, high-speed SPI, UART, I2S, and I2C.

Features:

1. **CPU and On Chip Memory**

   • ESP32-D0WD-V3 embedded, Xtensa dual-core 32-bit LX6 microprocessor, up to 240 MHz

   • 448 KB ROM

   • 520 KB SRAM

   • 16 KB SRAM in RTC WiFi

   • 802.11b/g/n

   • Bit rate: 802.11n up to 150 Mbps

   • A-MPDU and A-MSDU aggregation

   • 0.4 µs guard interval support

   • Center frequency range of operating channel: 2412 ~ 2484 MHz

2. **Bluetooth**

   • Bluetooth V4.2 BR/EDR and Bluetooth LE specification

   • Class-1, class-2 and class-3 transmitter

   • AFH

   • CVSD and SBC

3. **Peripherals**

   • SD card, UART, SPI, SDIO, I2C, LED PWM, Motor PWM, I2S, IR, pulse counter, GPIO, capacitive touch sensor, ADC, DAC, TWAI® (Compatible with ISO 11898-1, i.e. CAN Specification 2.0)

4. **Integrated Components on Module**

   • 40 MHz crystal oscillator

   • 4/8/16 MB SPI flash

5. **Antenna Options**

   • ESP32-WROOM-32E: On-board PCB antenna

   • ESP32-WROOM-32UE: external antenna via a connector

6. **Operating Conditions**

   • Operating voltage/Power supply: 3.0 ~ 3.6 V

   • Operating ambient temperature: – 85 °C version: –40 ~ 85 °C – 105 °C version: –40 ~ 105 °C. Note that only the modules embedded with a 4/8 MB flash support this version.

7. **Certification**

   • Bluetooth certification: BQB

   • RF certification: FCC/CE RED/SRRC

   • Green certification: REACH/RoHS

8. **Reliability Test**

   • HTOL/HTSL/uHAST/TCT/ESD

Figure 2.6.1. ***ESP 32 Microcontroller and its PINOUT.***

2.7. TTP223 Touch Sensor

The TTP223 Touch Key Module is based on a touch-sensing IC (TTP223) capacitive touch switch module. A capacitive touch sensor module based on the dedicated TTP223 touch sensor IC. The module provides a single integrated touch sensing area of 11 x 10.5mm with a sensor range of ~5mm.

An onboard LED will give a visual indication of when the sensor is triggered. When triggered the output of the module will switch from its idle low state to high (default operation). Solder jumpers allow for reconfiguring its mode of operation to be either active low or toggle output.

TTP223 Capacitive Touch Sensor Module. The TTP223 is a touchpad detector IC replicating a single tactile button. The TTP223 is a touchpad detector IC replicating a single tactile button. This touch detection IC is designed for replacing traditional direct button key with diverse pad size.

AB=00: No-lock High TTL level output;
AB=01: Self-lock High TTL level output;
AB=10: No-lock Low TTL level output;
AB=11: Self-lock Low TTL level output;

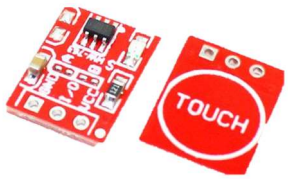


Figure 2.7.1. ***TTP223 Touch Sensor.***

2.8. ACS712 Current Sensor

Measurement of current is necessary for the proper working of devices. Measurement of voltage is Passive task and it can be done without affecting the system. Whereas measurement of current is an Intrusive task which cannot be detected directly as voltage. For measuring current in a circuit, a sensor is required. ACS712 Current Sensor is the sensor that can be used to measure and calculate the amount of current applied to the conductor without affecting the performance of the system.

The ACS712 is a fully integrated, hall effect-based linear current sensor with 2.1kVRMS voltage isolation and a integrated low-resistance current conductor. Technical terms aside, it's simply put forth as a current sensor that uses its conductor to calculate and measure the amount of current applied.

2.8.1. Features of ACS712 Current Sensor

- Low-noise analog signal path
- Device bandwidth is set via the new FILTER pin
- 5 µs output rise time in response to step input current
- 80 kHz bandwidth
- Total output error 1.5% at TA = 25°C
- 1.2 mΩ internal conductor resistance
- 2.1 kVRMS minimum isolation voltage from pins 1-4 to pins 5-8
- 5.0 V, single supply operation
- 66 to 185 mV/A output sensitivity
- Output voltage proportional to AC or DC currents
- Extremely stable output offset voltage

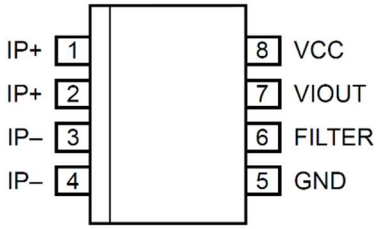


Figure 2.8.1. ***TTP223 Touch Sensor.***

Number	Name	Description
1 and 2	IP+	Terminals for current being sensed; fused internally
3 and 4	IP-	Terminals for current being sensed; fused internally
5	GND	Signal ground terminal
6	FILTER	Terminal for external capacitor that sets bandwidth
7	VIOUT	Analog output signal
8	VCC	Device power supply terminal

Table 2.8.1. ***Pinout of ACS712.***

2.8.2. Working Principle

Current Sensor detects the current in a wire or conductor and generates a signal proportional to the detected current either in the form of analog voltage or digital output. Current Sensing is done in two ways – Direct sensing and Indirect Sensing. In Direct sensing, to detect current, Ohm's law is used to measure the voltage drop occurred in a wire when current flows through it. A current-carrying conductor also gives rise to a magnetic field in its surrounding.

In Indirect Sensing, the current is measured by calculating this magnetic field by applying either Faraday's law or Ampere law. Here either a Transformer or Hall effect sensor or fiberoptic current sensor are used to sense the magnetic field. ACS712 Current Sensor uses Indirect Sensing method to calculate the current. To sense current a liner, low-offset Hall sensor circuit is used in this IC. This sensor is located at the surface of the IC on a copper conduction path. When current flows through this copper conduction path it generates a magnetic field which is sensed by the Hall effect sensor. A voltage proportional to the sensed magnetic field is generated by the Hall sensor, which is used to measure current.

2.8.3. Operating Characteristics

Characteristic	Symbol	Test Conditions	Min.	Typ.	Max.	Units
ELECTRICAL CHARACTERISTICS						
Supply Voltage	V_{CC}		4.5	5.0	5.5	V
Supply Current	I_{CC}	V_{CC} = 5.0 V, output open	6	8	11	mA
Output Zener Clamp Voltage	V_Z	I_{CC} = 11 mA, T_A = 25°C	6	8.3	–	V
Output Resistance	R_{IOUT}	I_{IOUT} = 1.2 mA, T_A=25°C	–	1	2	Ω
Output Capacitance Load	C_{LOAD}	VIOUT to GND	–	–	10	nF
Output Resistive Load	R_{LOAD}	VIOUT to GND	4.7	–	–	kΩ
Primary Conductor Resistance	$R_{PRIMARY}$	T_A = 25°C	–	1.2	–	mΩ
RMS Isolation Voltage	V_{ISORMS}	Pins 1-4 and 5-8; 60 Hz, 1 minute, T_A=25°C	2100	–	–	V
DC Isolation Voltage	V_{ISODC}	Pins 1-4 and 5-8; 1 minute, T_A=25°C	–	5000	–	V
Propagation Time	t_{PROP}	I_P = I_P(max), T_A = 25°C, C_{OUT} = open	–	3	–	µs
Response Time	$t_{RESPONSE}$	I_P = I_P(max), T_A = 25°C, C_{OUT} = open	–	7	–	µs
Rise Time	t_r	I_P = I_P(max), T_A = 25°C, C_{OUT} = open	–	5	–	µs
Frequency Bandwidth	f	–3 dB, T_A = 25°C; I_P is 10 A peak-to-peak	50	–	–	kHz
Nonlinearity	E_{LIN}	Over full range of I_P	–	±1	±1.5	%
Symmetry	E_{SYM}	Over full range of I_P	98	100	102	%
Zero Current Output Voltage	$V_{IOUT(Q)}$	Bidirectional; I_P = 0 A, T_A = 25°C	–	V_{CC} × 0.5	–	V
Magnetic Offset Error	V_{ERROM}	I_P = 0 A, after excursion of 5 A	–	0	–	mV
Clamping Voltage	V_{CH}		Typ.–110	V_{CC} × 0.9375	Typ.+110	mV
	V_{CL}		Typ.–110	V_{CC} × 0.0625	Typ.+110	mV
Power-On Time	t_{PO}	Output reaches 90% of steady-state level, T_J=25°C, 20 A present on leadframe	–	35	–	µs
Magnetic Coupling[2]			–	12	–	G/A
Internal Filter Resistance[3]	$R_{F(INT)}$			1.7		kΩ

Table 2.8.2. ***Electrical Characteristics of ACS712.***

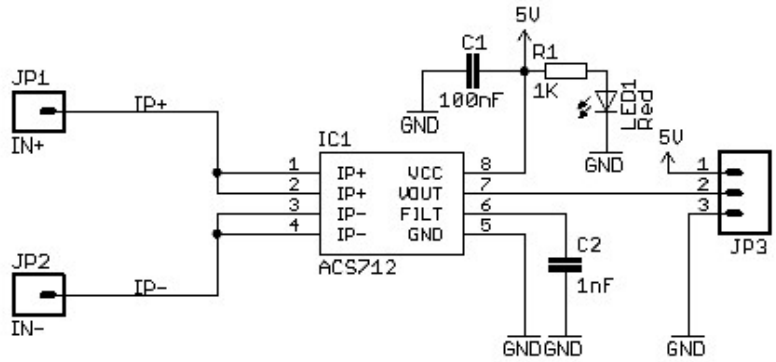


Figure 2.8.2. ***Application Circuit Used in Project.***

2.9. ZMPT101 Voltage Sensor

ZMPT101B AC Voltage Sensor is the best for the purpose of the 1ph AC voltage measurement, where we need to measure the accurate AC voltage with the voltage transformer. This is an ideal choice to measure the AC voltage using Arduino/ESP8266/Raspberry Pi like an open-source platform. In many electrical projects, engineer directly deals with measurements with few basic requirements like:

- High galvanic isolation
- Wide Range
- High accuracy
- Good Consistency

ZMPT 101B is a high-precision voltage Transformer. This module makes it easy to monitor AC mains voltage up to 1000 volts.

2.9.1. Features of ZMPT101 module

- Voltage Transformer: Onboard Precision Micro Voltage Transformer
- Operational amplifier circuit: high-precision on-board amplifier circuit, the signal to do the exact sampling and appropriate compensation.
- Input/Output: the module can measure AC voltage within 250V, the corresponding output voltage amplitude can be adjusted by using a potentiometer (multi-turn trim pot).
- Output signal: the output signal for the sine wave, the waveform of the median DC component (offset of 2.5V at 5V input supply)
- Supply voltage: 5V
- Operating temperature: 40ºC ~ + 70ºC

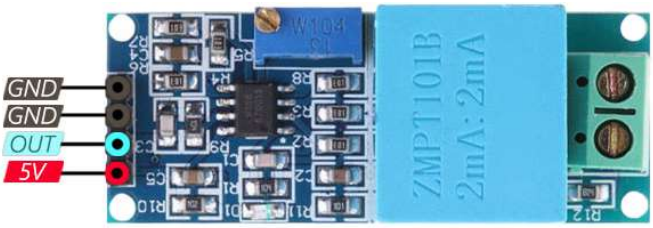


Figure 2.9.1. ***ZMPT101 Voltage Sensor.***

2.10. JST Connectors

JST connectors are electrical connectors manufactured to the design standards originally developed by J.S.T. Mfg. Co. (Japan Solderless Terminal). JST manufactures numerous series (families) and pitches (pin-to-pin distance) of connectors.

JST connectors are used in many types of products, and commonly used by electronics hobbyists and consumer products for rechargeable battery packs, battery balancers, battery eliminator circuits, 3D printers, and radio-controlled servos.

The term "JST" is sometimes incorrectly used as a vernacular term meaning any small white electrical connector mounted on PCBs.

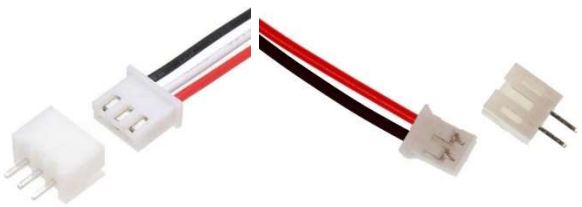


Figure 2.10.1. ***JST Connectors.***

2.11. Screw Terminal

A screw terminal is a type of electrical connection where a wire is held by the tightening of a screw. The wire may be wrapped directly under the head of a screw, may be held by a metal plate forced against the wire by a screw, or may be held by what is, in effect, a set screw in the side of a metal tube. The wire may be directly stripped of insulation and inserted under the head of a screw or into the terminal. Otherwise, it may be either inserted first into a ferrule, which is then inserted into the terminal, or else attached to a connecting lug, which is then fixed under the screw head.

Depending on the design, a flat-blade screwdriver, a cross-blade screwdriver, hex key, Torx key, or other tool may be required to properly tighten the connection for reliable operation.

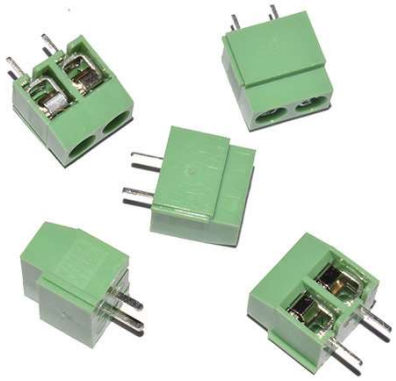


Figure 2.11.1. ***Screw Terminals.***

2.12. Zero PCB

Zero PCB is basically a general-purpose printed circuit board (PCB), also known as perfboard or DOT PCB. It is a thin rigid copper sheet with holes pre-drilled at standard intervals across a grid with 2.54mm (0.1-inch) spacing between holes. Each hole is encircled by a round or square copper pad so that component lead can be inserted into the hole and soldered around the pad without short-circuiting the nearby pads and other leads. For connecting the lead of component with another lead, solder these together or join these using a suitable conducting wire.

Veroboard or stripboard is also a general-purpose PCB characterized by a 2.54mm regular (rectangular) grid of holes, with parallel strips of copper cladding running in one direction across one side of the board. This general-purpose PCB is known by the name of the original product, Veroboard, a trademark of British company Vero Technologies Ltd and Canadian company Pixel Print Ltd.

Both zero PCB and Veroboard are popular among hobbyists, beginners, and students for rapid prototyping and project works.

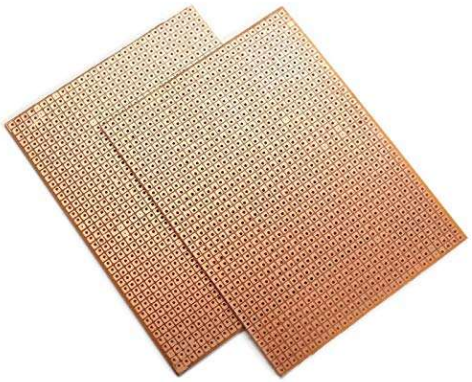


Figure 2.12.1. ***Zero PCB.***

CHAPTER-III

LIST OF SOFTWARE

3. LIST OF SOFTWARE

Sr. No.	Software
1	Arduino IDE
2	Autodesk EAGLE

Table 3.0.1. ***List of Software.***

3.1. Arduino IDE

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals - has gathered around this open-source platform, their contributions have added up to an incredible amount of accessible knowledge that can be of great help to novices and experts alike.

Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. The software, too, is open-source, and it is growing through the contributions of users worldwide.

Figure 3.1.1. ***Arduino IDE Snap.***

3.2. Autodesk EAGLE

EAGLE is a scriptable electronic design automation (EDA) application with schematic capture, printed circuit board (PCB) layout, auto-router and computer-aided manufacturing (CAM) features. EAGLE stands for Easily Applicable Graphical Layout Editor (German: Einfach Anzuwendender Grafischer Layout-Editor) and is developed by CadSoft Computer GmbH. The company was acquired by Autodesk Inc. in 2016.

EAGLE contains a schematic editor, for designing circuit diagrams. Schematics are stored in files with .SCH extension, parts are defined in device libraries with .LBR extension. Parts can be placed on many sheets and connected together through ports.

The PCB layout editor stores board files with the extension .BRD. It allows back-annotation to the schematic and auto-routing to automatically connect traces based on the connections defined in the schematic.

EAGLE saves Gerber and PostScript layout files as well as Excellon and Sieb & Meyer drill files. These are standard file formats accepted by PCB fabrication companies, but given EAGLE's typical user base of small design firms and hobbyists, many PCB fabricators and assembly shops also accept EAGLE board files (with extension .BRD) directly to export optimized production files and pick-and-place data themselves. EAGLE provides a multi-window graphical user interface and menu system for editing, project management and to customize the interface and design parameters. The system can be controlled via mouse, keyboard hotkeys or by entering specific commands at an embedded command line. Keyboard hotkeys can be user defined. Multiple repeating commands can be combined into script files (with file extension .SCR). It is also possible to explore design files utilizing an EAGLE-specific object-oriented programming language (with extension .ULP).

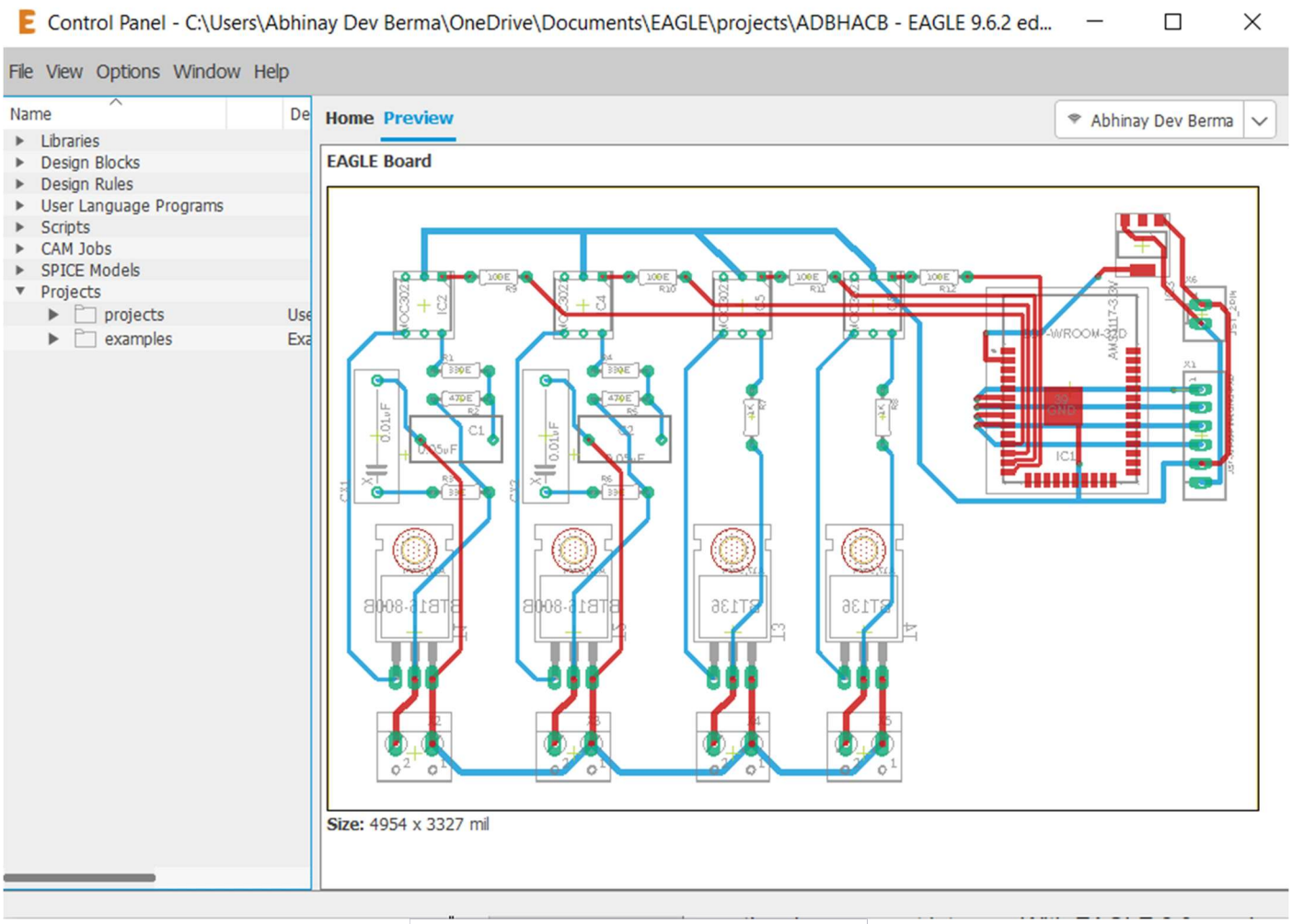


Figure 3.2.1. ***Autodesk EAGLE Snap.***

3.3. Visual Studio Code

Visual Studio Code, also commonly referred to as VS Code is a source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality. In the Stack Overflow 2021 Developer Survey, Visual Studio Code was ranked the most popular developer environment tool, with 70% of 82,000 respondents reporting that they use it.

Visual Studio Code was first announced on April 29, 2015, by Microsoft at the 2015 Build conference. A preview build was released shortly thereafter. On November 18, 2015, the source of Visual Studio Code was released under the MIT License, and made available on GitHub. Extension support was also announced. On April 14, 2016, Visual Studio Code graduated from the public preview stage and was released to the Web. Microsoft has released most of Visual Studio Code's source code on GitHub under the permissive MIT License, while the releases by Microsoft are proprietary freeware.

3.3.1. Features

Visual Studio Code is a source-code editor that can be used with a variety of programming languages, including Java, JavaScript, Go, Node.js, Python, C++ and Fortran. It is based on the Electron framework,[20] which is used to develop Node.js Web applications that run on the Blink layout engine. Visual Studio Code employs the same editor component (codenamed "Monaco") used in Azure DevOps (formerly called Visual Studio Online and Visual Studio Team Services). Out of the box, Visual Studio Code includes basic support for most common programming languages. This basic support includes syntax highlighting, bracket matching, code folding, and configurable snippets. Visual Studio Code also ships with IntelliSense for JavaScript, TypeScript, JSON, CSS, and HTML, as well as debugging support for Node.js. Support for additional languages can be provided by freely available extensions on the VS Code Marketplace.

Instead of a project system, it allows users to open one or more directories, which can then be saved in workspaces for future reuse. This allows it to operate as a language-agnostic code editor for any language. It supports many programming languages and a set of features that differs per language. Unwanted files and folders can be excluded from the project tree via the settings. Many Visual Studio Code features are not exposed through menus or the user interface but can be accessed via the command palette.

Visual Studio Code can be extended via extensions, available through a central repository. This includes additions to the and language support. A notable feature is the ability to create extensions that add support for new languages, themes, debuggers, time travel debuggers, perform static code analysis, and add code linters using the Language Server Protocol.

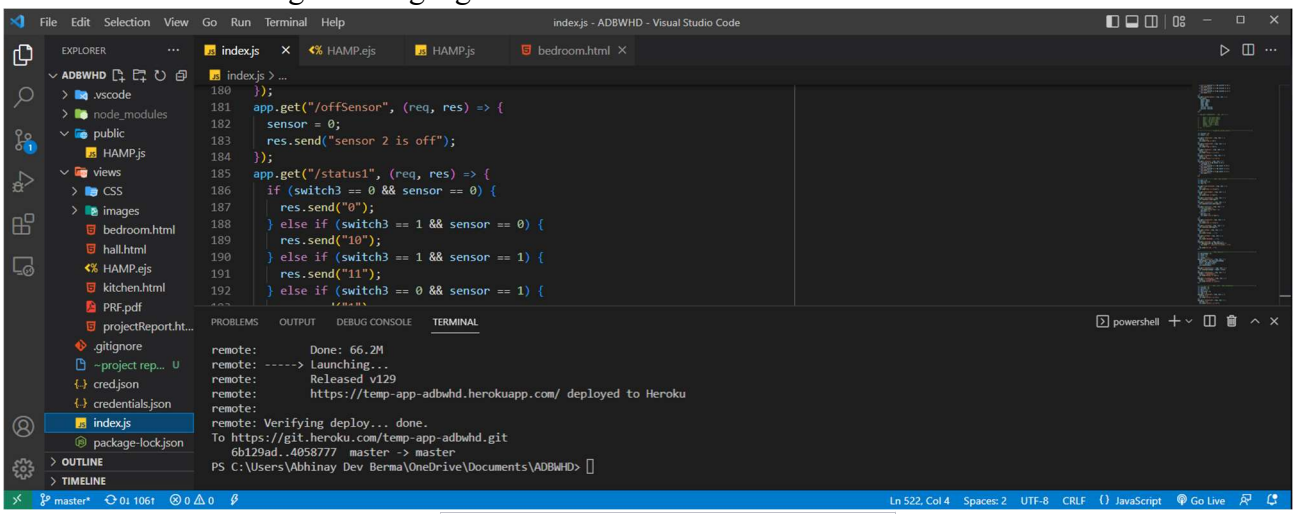


Figure 3.3.1. ***Visual Studio Code Snap.***

CHAPTER-IV

LITERATURE SURVEY

4. INTRODUCTION

Automation performs an increasingly vital role in daily experience and global economy. Engineers strive to combine automated devices with mathematical and organizational tools to create complex systems for a rapidly expanding range of applications and human activities. The concept of home automation has been around since the late 1970s. But with the enhancement of technology and smart services, people's expectations have changed a lot during the course of time to perfectly turn the traditional house into smart home, and also think that what a home should do or how the services should be provided and accessed at home to became a smart home and so has the idea of home automation systems. A home automation system means to grant the endusers to manage and handle the electric appliances. If we look at different home automation systems over time, they have always tried to provide efficient, convenient, and safe ways for home inhabitants to access their homes. Regardless of the change in user's hope, growing technology, or change of time, the appearance of a home automation system has remained the same. Many existing, well-established home automation systems are based on wired communication such as Arduino based and raspberry pi-based home automation systems. This does not pose a problem until the system is planned well in advance and installed during the physical construction of the building. But for already existing buildings the implementation cost goes very high. In contrast, Wireless systems can be of great help for automation systems like Bluetooth, Wi-Fi and IOT based home automation systems. With the advancement of wireless technologies such as Wi-Fi, cloud networks in the recent past, wireless systems are used every day and everywhere.

4.1. Bluetooth based home automation system using cell phones

In Bluetooth based home automation system the home appliances are connected to the Arduino BT board at input output ports using relay. The program of Arduino BT board is based on high level interactive C language of microcontrollers; the connection is made via Bluetooth. The password protection is provided so only authorized user is allowed to access the appliances. The Bluetooth connection is established between Arduino BT board and phone for wireless communication. In this system the python script is used and it can install on any of the Symbian OS environment, it is portable. One circuit is designed and implemented for receiving the feedback from the phone, which indicate the status of the device.

4.2. Zigbee based home automation system using cell phones

To monitor and control the home appliances the system is designed and implemented using Zigbee. The device performance is record and store by network coordinators. For this the Wi-Fi network is used, which uses the four switch port standard wireless ADSL modern router. The network SSID and security Wi-Fi parameter are preconfigured. The message for security purpose first process by the virtual home algorithm and when it is declared safe it is re-encrypted and forward to the real network device of the home. Over Zigbee network, Zigbee controller sent messages to the end. The safety and security of all messages that are received by the virtual home algorithm. To reduce the expense of the system and the intrusiveness of respective installation of the system Zigbee communication is helpful.

4.3. GSM based home automation system using cell phones

Because of the mobile phone and GSM technology, the GSM based home automation is lure to research. The SMS based home automation, GPRS based home automation and dual tone multi frequency (DTMF) based home automation, these options we considered mainly for communication in GSM. In figure shows the logical diagram the work of A. Alheraish, it shows how the home sensors and devices interact with the home network and communicates through GSM and SIM (subscriber identity module). The system use transducer which convert machine function into electrical signals which goes into microcontroller. The sensors of system convert the physical qualities like sound, temperature and humidity into some other quantity like voltage. The microcontroller analysis all signal and convert them into command to understand by GSM module. Select appropriate communication method among SMS, GPRS and DTFC based on the command which received GSM module.

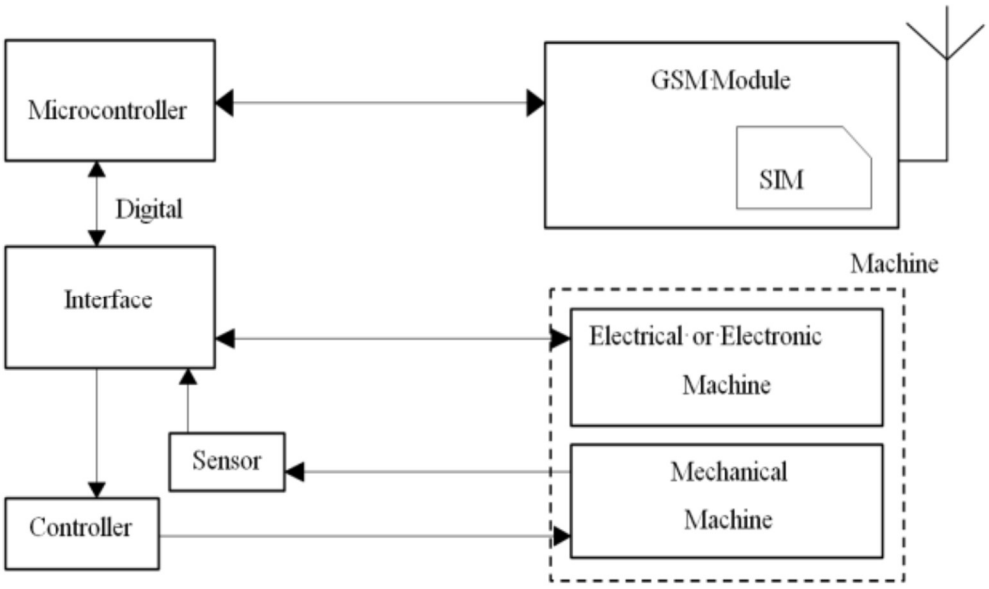


Figure 4.3.1. ***GSM and Mobile-based home automation system.***

4.4. Wi-Fi based home automation system using cell phones

Wi-Fi based home automation system mainly consist three modules, the server, the hardware interface module, and the software package. The figure shows the system model layout. Wi-Fi technology is used by server, and hardware Interface module to communicate with each other. The same technology uses to login to the server web-based application. The server is connected to the internet, so remote users can access server web-based application through the internet using compatible web browser. Software of the latest home automation system is split to server application software, and Microcontroller (Arduino) firmware. The Arduino software, built using C language, using IDE comes with the microcontroller itself. Arduino software is culpable for gathering events from connected sensors, then applies action to actuators and pre-programed in the server. Another job is to report the and record the history in the server DB. The server application software package for the proposed home automation system, is a web-based application built using asp.net. The server application software can be accessed from internal network or from internet if the server has real IP on the internet using any internet navigator supports asp.net technology. Server application software is culpable of, maintain the whole home automation system, setup, configuration. Server use database to keep log of home automation system components, we choose to use XML files to save system log.

Figure 4.4.1. ***Wi-Fi Mobile-based home automation system.***

CHAPTER-V

PROBLEM IDENTIFICATION

5. PROBLEM IDENTIFICATION

Home automation systems suffers four main challenges; these are poor manageability, inflexibility, difficulty in achieving security and high cost of ownership, The main objectives of this research is to design and implement a home automation system using IoT that is capable of controlling and automating most of the house appliances through an easy manageable web interface. The proposed system has a great flexibility by using Wi-Fi technology to interconnect its distributed sensors to home automation server. This will decrease the deployment cost and will increase the ability of upgrading, and system reconfiguration.

Today people are looking at ways and means to better their life-style using the latest technologies that are available. Any new facility for home appliance that promises to enhance their life-style is grabbed by the consumers. The more such facilities and appliances are added, it becomes inevitable to have easy and convenient methods and means to control and operate these appliances.

Conventional wall switches are located in different parts of a house and thus necessitates manual operations like to switch on or off these switches to control various appliances. It gets virtually impossible to keep track of appliances that are running and also to monitor their Performances. And Aim is to Build an system which controls home appliances with less efforts, like control using mobile….

Visit the Tata Power EZ Home Store

Tata Power EZ Home 2 Gang 5A Modular Retrofit Smart Switch, No Re-Wiring for Switches, Overload Protection, Tracks Power Usage

★★★★☆ 22 ratings

-48% ₹2,080^{00}

M.R.P.: ~~₹3,999.00~~

Inclusive of all taxes

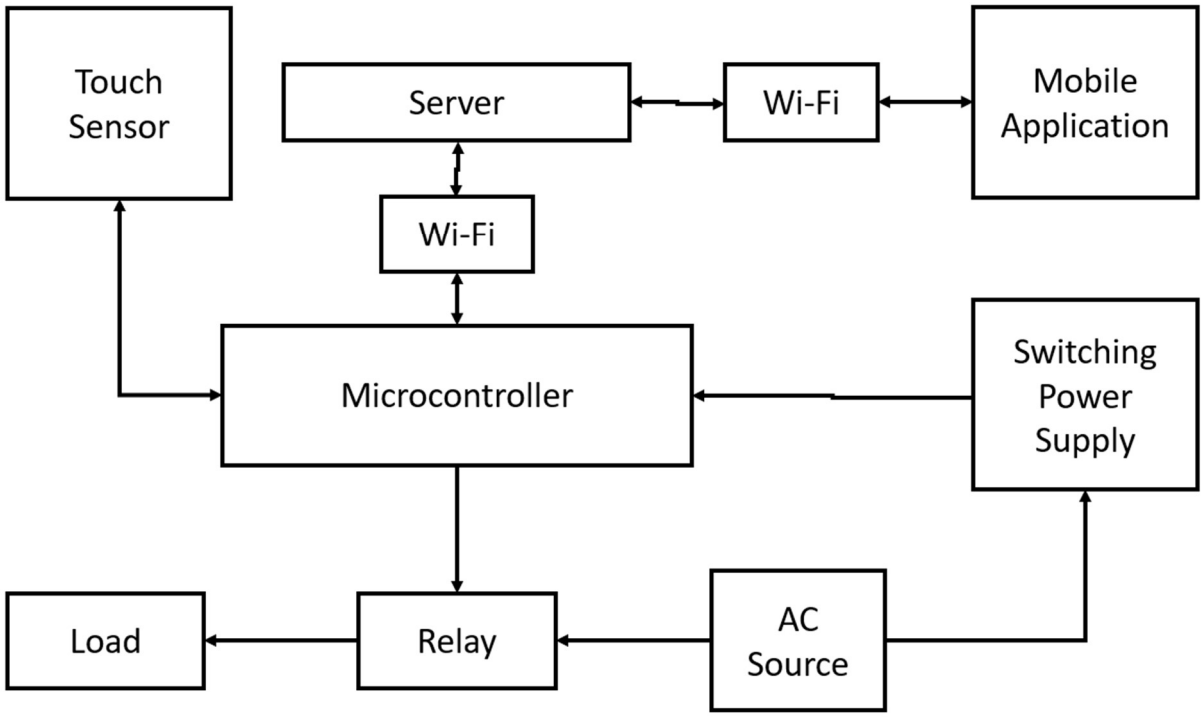
CHAPTER-VI

METHODOLOGY

6. METHODOLOGY

1. Make a Solid-State Relay as per the Circuit Diagram.
2. Make Connection with ESP32 Microcontroller.
3. Connect 230V AC to 5V DC Switching Power Supply to ESP32.
4. Program the ESP32 Microcontroller using Arduino IDE Software.
5. Connect Touch Switches to ESP32 Microcontroller.
6. Connect Loads and Supply to Solid State Relay.
7. Turn on the Supply.
8. ESP32 Will automatically connect to home Wi-Fi network.
9. Now we can operate appliances with touch switches.
10. To operate appliances remotely, open the application on your mobile phone and operate via switches.

6.1. BLOCK DIAGRAM



Solid State Relay: Two types of relays is used in this project, one is for small loads(4A) and second is for Heavy Loads (16A).

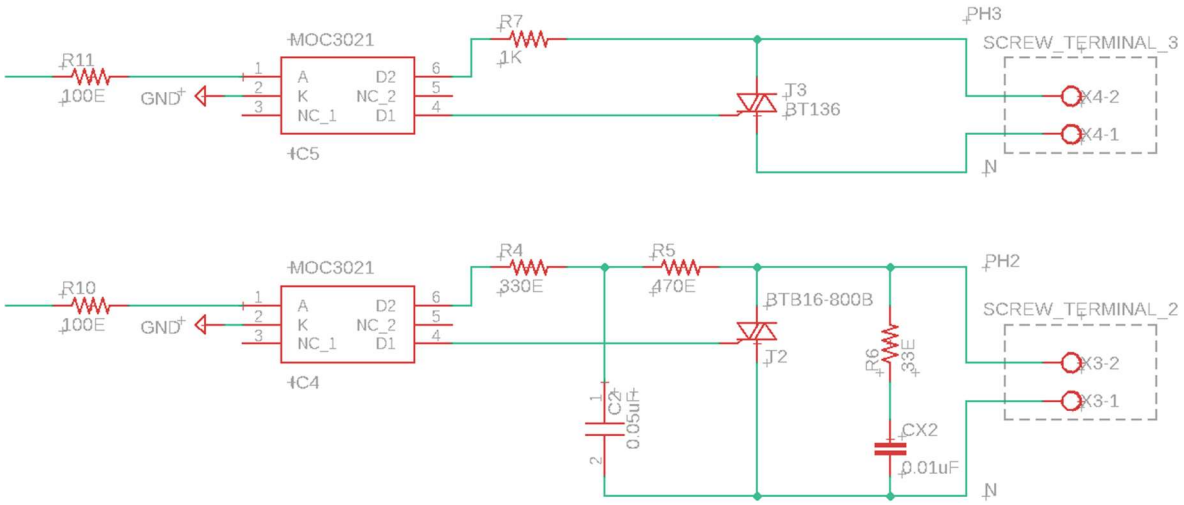
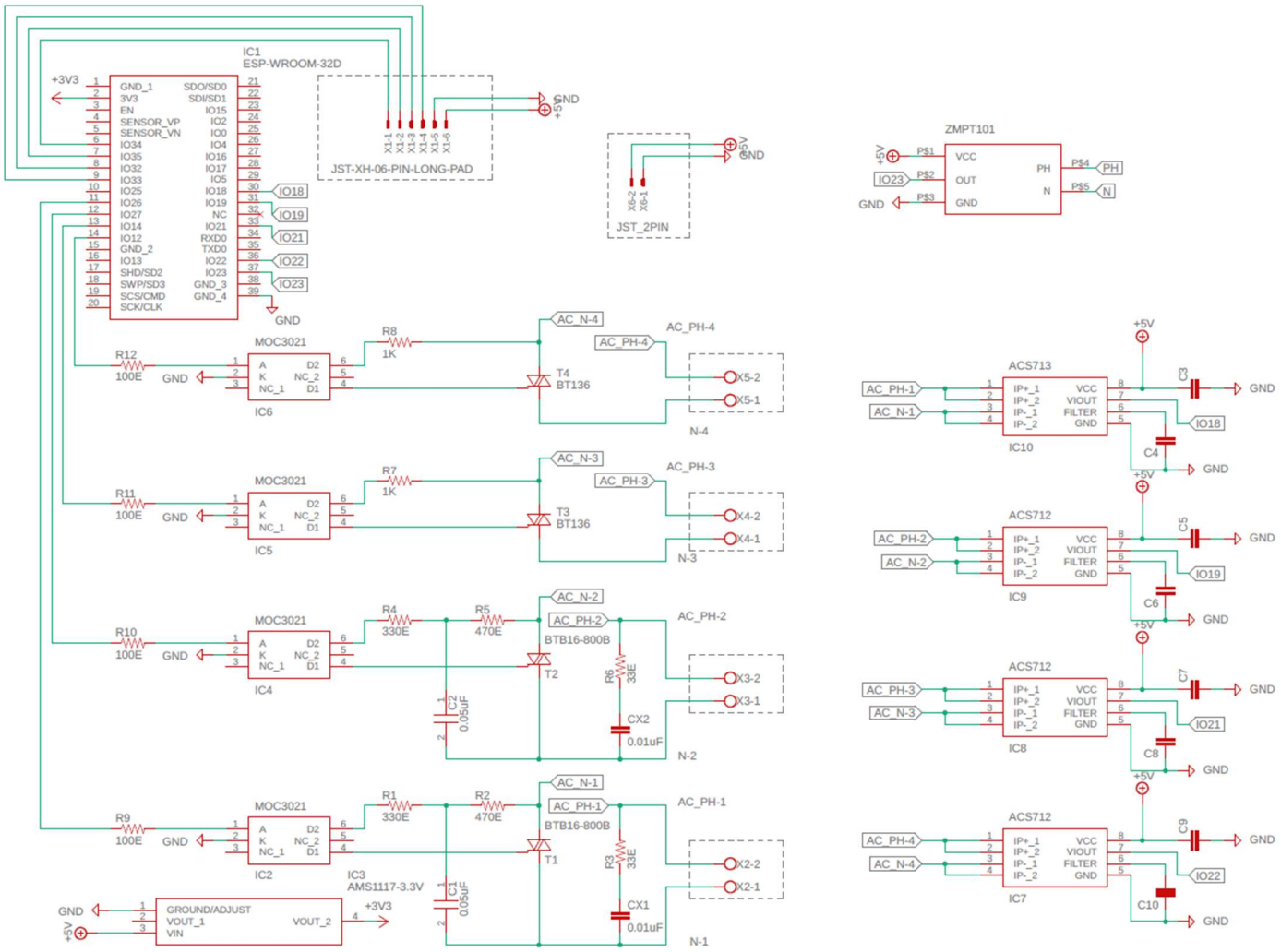


Figure 6.1.1. ***Solid State Relay Circuit.***

6.2. CIRCUIT DIAGRAM



6.3. WORKING

1. When all the connection is done as per the circuit diagram and supply is on, then ESP32 automatically connect to Wi-Fi Network.
2. After Connecting to Wi-Fi network, it communicates with the server and read the appliances last status and turn ON/OFF accordingly.
3. Read the current and voltage of appliances and update the server continuously.
4. Now it enters the loop where it frequently checks for any update via mobile application present in server and change in touch sensor switch.
5. If any changes to ON/OFF appliances via touch sensor ESP32 Microcontroller sense it and turn ON/OFF appliances accordingly and update the server via HTTP protocol.
6. If any changes to ON/OFF appliances via mobile application, then it updates server via HTTP protocol.
7. And then in the next checkup of microcontroller it turns ON/OFF appliances.
8. This loop (from point 3) continues forever.

6.4. PHYSICAL HARDWARE

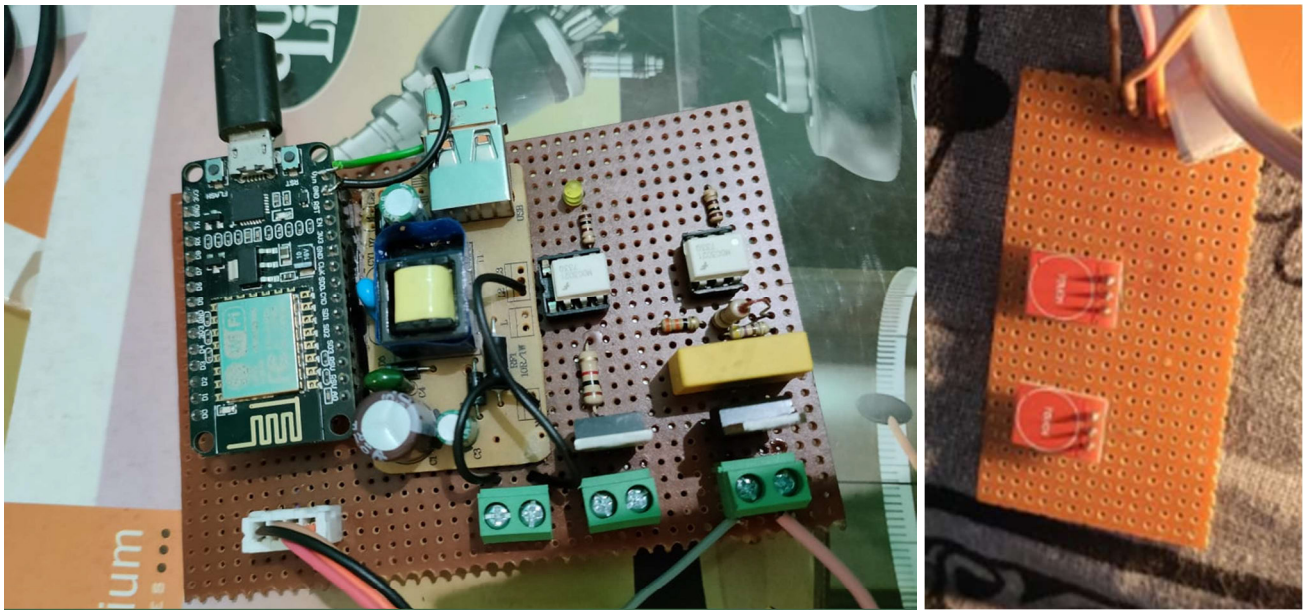


Figure 6.4.1. ***Solid State Circuit in Zero PCB.***

6.5. PCB LAYOUT

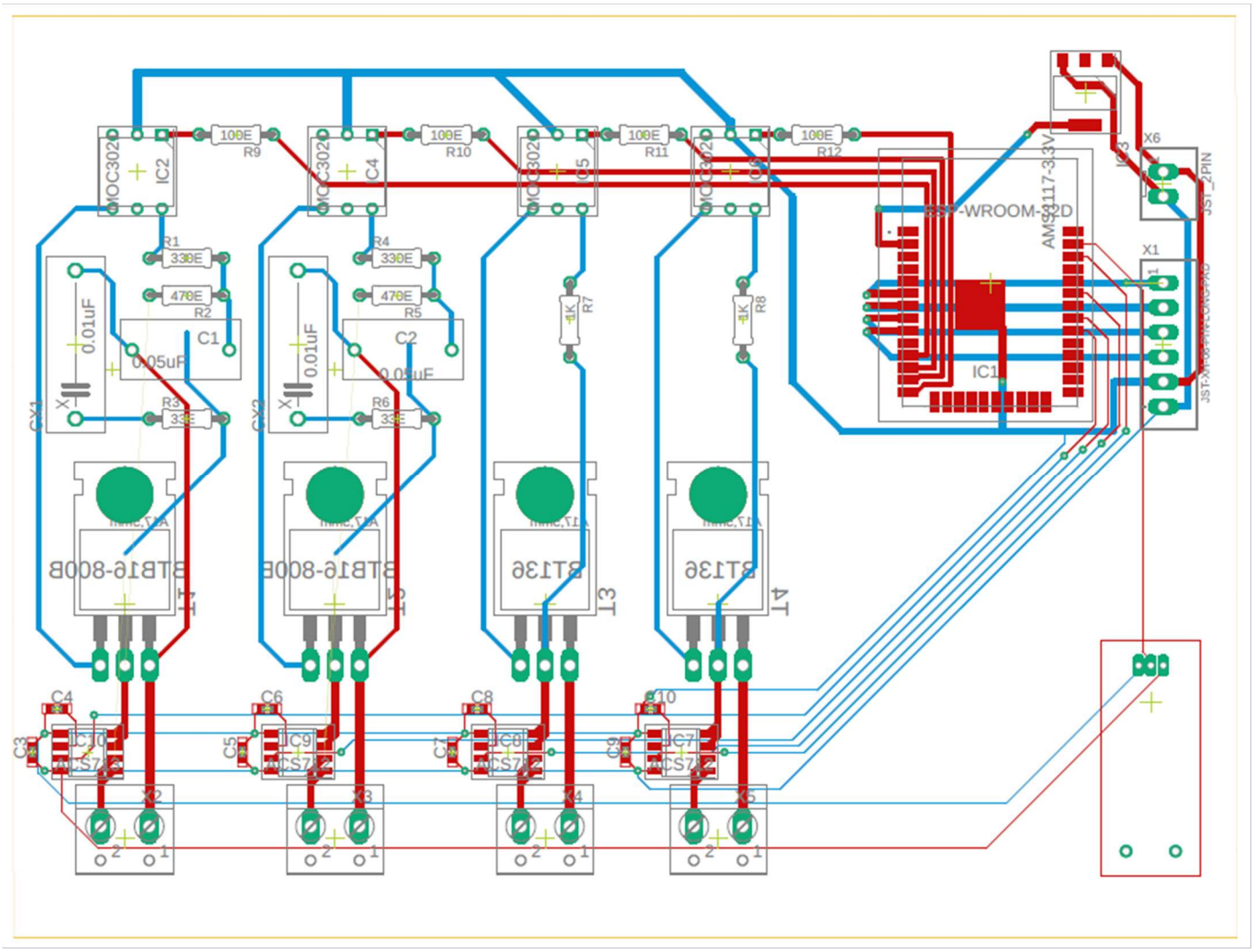


Figure 6.5.1. ***PCB Layout Top Side.***

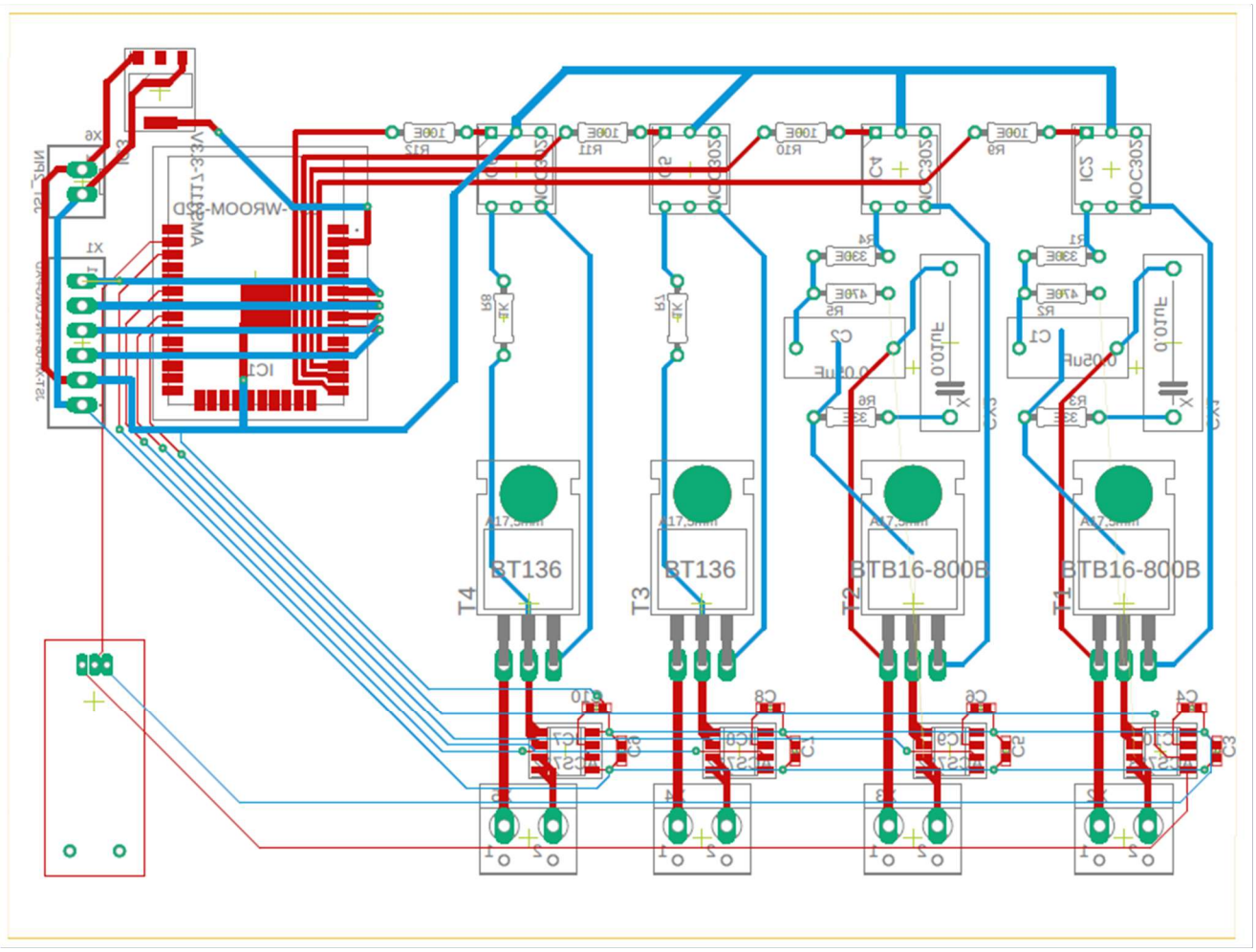
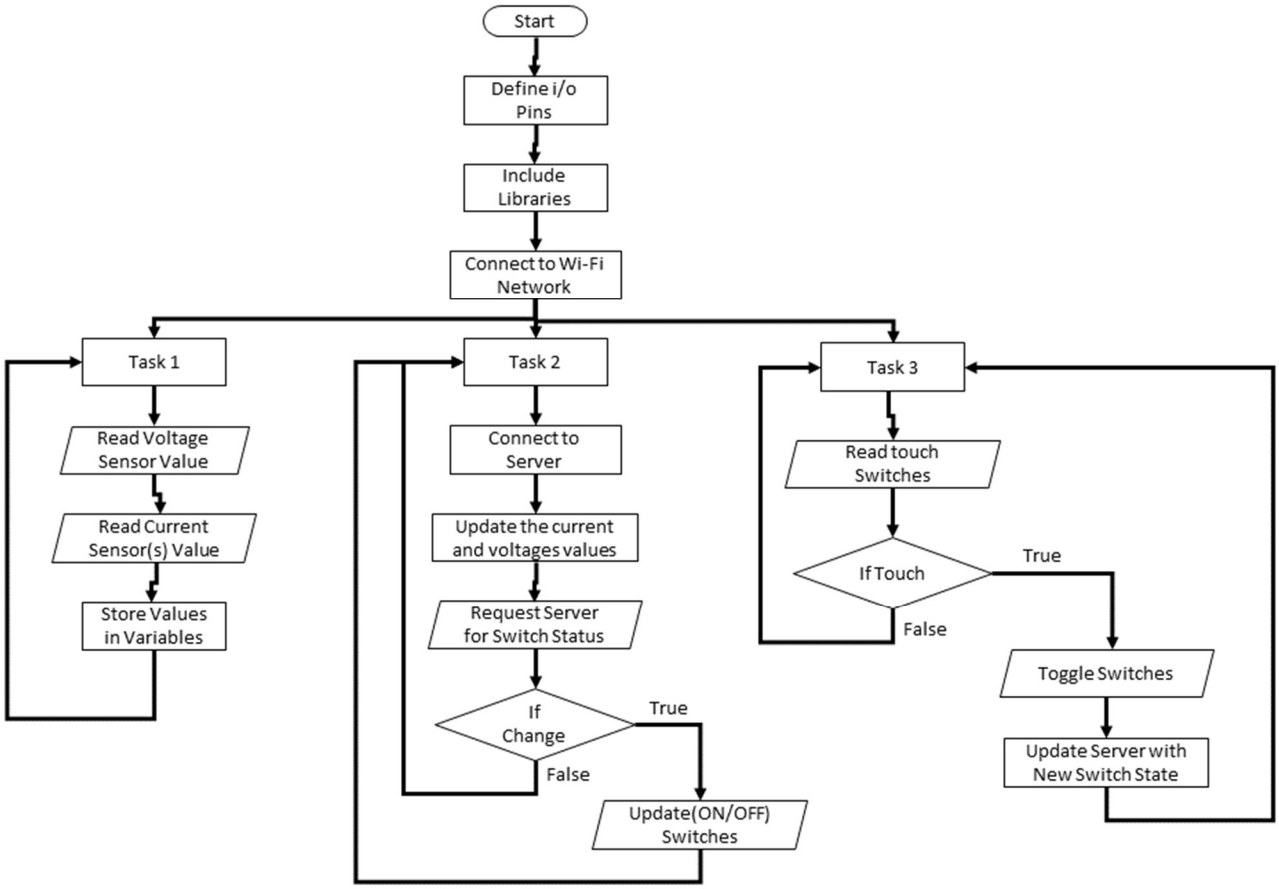


Figure 6.5.2. ***PCB Layout Bottom Side.***

6.6. PROGRAMMING FLOW CHART



6.7. CODING

```
/* <----Defining Pins for Sensor and Switches----> */
#define CS1 34
#define CS2 35
#define VS 32
#define SSR1 5
#define SSR2 17
#define TS1 16
#define TS2 4
#define VOLT_CAL 148
#define CURRENT_CAL 3
#define CURRENT_CAL_1 1
/* <----Including Libraries for WiFi and Sensors----> */
#include <WiFi.h>
#include <HTTPClient.h>
#include <WiFiClient.h>
#include "EmonLib.h"
/* <----Initialising Multitasking----> */
TaskHandle_t Task1;
TaskHandle_t Task2;
/* <----WiFi Credentials----> */
const char* ssid = "realme X";
const char* password = "13140711";
/* <----Useful Variables----> */
int switchstate = 22;
String res = "";
String serverName = "http://temp-app-adbwhd.herokuapp.com";
int sta = 0;
bool switchOperate = false;
float current1Draw = 0;
float current2Draw = 0;
float supply1Voltage = 0;
/* <----Objects for Current sensor and voltage----> */
EnergyMonitor emon1;
EnergyMonitor emon2;
/* <----Program Starts----> */
void setup() {
  Serial.begin(115200);     //Bordrate fo serial monitor
  //Initialising i/o pins
  pinMode(SSR1, OUTPUT);
```

```
  pinMode(SSR2, OUTPUT);
  pinMode(TS1, INPUT);
  pinMode(TS2, INPUT);
  /* <----Connecting to WiFi----> */
  WiFi.begin(ssid, password);
  Serial.println("Connecting to Network....");
  while (WiFi.waitForConnectResult() != WL_CONNECTED) {
   Serial.println("Unable to connect wifi, Rebooting...");
   delay(5000);
   ESP.restart();
  }
  Serial.print("Connected to WiFi network with IP Address: ");
  Serial.println(WiFi.localIP()); //Printing local IP
  /* <----Updating Server as all appliances is off intialy----> */
  res = server("/offSwitch1");
  Serial.println(res);
  res = server("/offSwitch2");
  Serial.println(res);
  switchstate = 0;

  emon1.voltage(VS, VOLT_CAL, 1.7);  // Voltage: input pin, calibration, phase_shift
  emon1.current(CS1, CURRENT_CAL);       // Current: input pin, calibration.
  emon2.voltage(VS, VOLT_CAL, 1.7);  // Voltage: input pin, calibration, phase_shift
  emon2.current(CS2, CURRENT_CAL_1);       // Current: input pin, calibration.


  //create a task that will be executed in the Task1code() function, with priority 1 and
executed on core 0
  xTaskCreatePinnedToCore(
    Task1code,   /* Task function. */
    "Task1",     /* name of task. */
    10000,       /* Stack size of task */
    NULL,        /* parameter of the task */
    1,           /* priority of the task */
    &Task1,      /* Task handle to keep track of created task */
    0);          /* pin task to core 0 */
  delay(500);

  //create a task that will be executed in the Task2code() function, with priority 1 and
executed on core 1
  xTaskCreatePinnedToCore(
```

```
    Task2code,   /* Task function. */
    "Task2",     /* name of task. */
    10000,       /* Stack size of task */
    NULL,        /* parameter of the task */
    1,           /* priority of the task */
    &Task2,      /* Task handle to keep track of created task */
    1);          /* pin task to core 1 */
  delay(500);

}

/* <----Task 1 Code----> */
void Task1code( void * pvParameters ) {
  Serial.print("Task1 running on core ");
  Serial.println(xPortGetCoreID());

  for (;;) {
      delay(200);

    res = server("/status?adbc1=" + String(current1Draw) + "&adbc2=" +
String(current2Draw) + "&adbv=" + String(supply1Voltage) + "&adbp=" +
String(((current1Draw + current2Draw) / 1000) * supply1Voltage));
    if (res.length() < 3) {
      sta = res.toInt();
      if (!switchOperate) {
        switch (sta) {
          case 0:
            digitalWrite(5, LOW);
            digitalWrite(17, LOW);
            break;

          case 1:
            digitalWrite(5, LOW);
            digitalWrite(17, HIGH);
            break;

          case 10:
            digitalWrite(5, HIGH);
            digitalWrite(17, LOW);
            break;
```

```
          case 11:
            digitalWrite(5, HIGH);
            digitalWrite(17, HIGH);
            break;
        }
       }
     }
   }
}
/* <----Task 1 Code----> */
void Task2code( void * pvParameters ) {
  Serial.print("Task2 running on core ");
  Serial.println(xPortGetCoreID());

  for (;;) {
    if (digitalRead(4) == 1) {
      switchOperate = true;
      if ((sta == 0) || (sta == 10)) {
        //res = server("/status");
        while (digitalRead(4) == 1) {
          delay(44);
        }
        digitalWrite(17, HIGH);
        res = server("/onSwitch2");
        if (sta == 0) {
          sta = 1;
        } else {
          sta = 11;
        }
      } else {
        while (digitalRead(4) == 1) {
          delay(44);
        }
        digitalWrite(17, LOW);
        res = server("/offSwitch2");
        if (sta == 1) {
          sta = 0;
        } else {
          sta = 10;
        }
      }
```

```
      switchOperate = false;
    }
    if (digitalRead(16) == 1) {
      switchOperate = true;
      if ((sta == 0) || (sta == 1)) {
        //res = server("/status");
        while (digitalRead(16) == 1) {
          delay(44);
        }
        digitalWrite(5, HIGH);
        res = server("/onSwitch1");
        if (sta == 0) {
          sta = 10;
        } else {
          sta = 11;
        }
      } else {
        while (digitalRead(16) == 1) {
          delay(44);
        }
        digitalWrite(5, LOW);
        res = server("/offSwitch1");
        if (sta == 10) {
          sta = 0;
        } else {
          sta = 1;
        }
      }
      switchOperate = false;
    }
  }
}

void loop() {
  emon1.calcVI(20, 100);     // Calculate all. No.of half wavelengths (crossings), time-out
  emon2.calcVI(20, 100);     // Calculate all. No.of half wavelengths (crossings), time-out
  current1Draw = emon1.Irms * 1000;   //extract Irms into Variable
  current2Draw = emon2.Irms * 1000;   //extract Irms into Variable
  supply1Voltage   = emon1.Vrms;      //extract Vrms into Variable
//  current1Draw = (current1Draw - 114.27) * 0.3505696757;    //Calibrated Irms
//  current2Draw = (current2Draw - 25.51) * 2.105263158;      //Calibrated Irms
```

```
    current1Draw = (current1Draw - 93.27) * 0.71942446043;      //Calibrated Irms
   current2Draw = (current2Draw - 56.51) * 1.10497237560;      //Calibrated Irms
   if (current1Draw < 150) {
    current1Draw = 0;
   }
   if (current2Draw < 150) {
    current2Draw = 0;
   }
}
String server( String req) {

  if (WiFi.status() == WL_CONNECTED) {
    WiFiClient client;
    HTTPClient http;
    Serial.println(req);
    String serverPath = serverName + req;

    // Your Domain name with URL path or IP address with path
    http.begin(client, serverPath.c_str());

    // Send HTTP GET request
    int httpResponseCode = http.GET();

    if (httpResponseCode > 0) {
      String payload = http.getString();
      Serial.println(payload);
      return payload;
    }
    else {
      return "error";
    }
    // Free resources
    http.end();
  }
  else {
    return "disconnected";
  }
}
```

CHAPTER-VII

RESULT

7. RESULT

Result:

1) Prototype of commercial PCB model is created for demonstration purpose and testing purpose which is working without any problem.

PCB is created for commercial use.

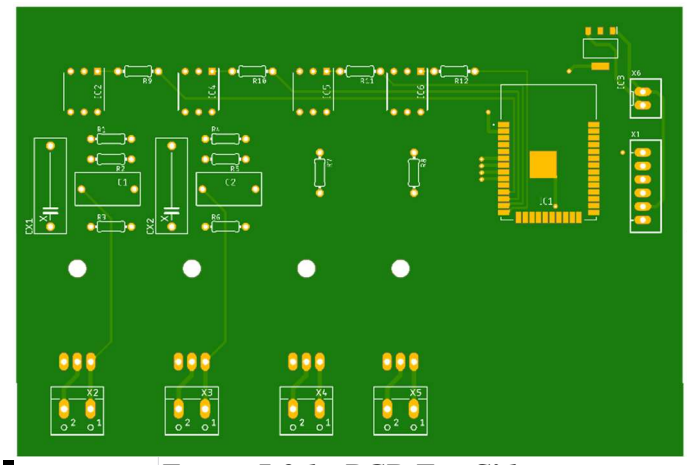


Figure 7.0.1. ***PCB Top Side***

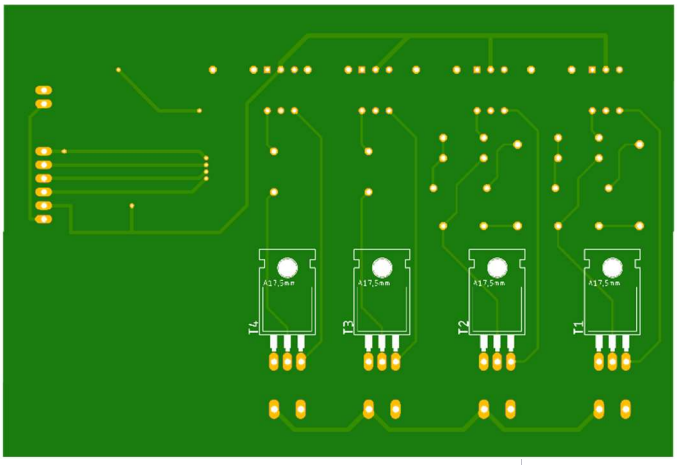


Figure 7.0.1. ***PCB Bottom Side.***

2) We Designed the whole website which show the instantaneous values of current, voltage and power of appliances via graph whose link is https://temp-app-adbwhd.herokuapp.com/

Figure 7.0.2. ***PCB Top Side***

3) We Can control the appliances of from website and app together. All are sync together.

CHAPTER-VIII

FUTURE SCOPE

8. FUTURE SCOPE

Future Scope:

1. The going with stage for home robotization advertise will happen subject to a couple of key overhauls in the progression open in Automation, for example, improvement in Wireless Automation blueprints and moreover bringing down of regard appears as the market starts perceive home mechanization use in more noteworthy volumes. A couple of examples that we foresee for this time of the business are,

2. Big associations like Philips, Siemens and Schneider will as time go on bring out truly mass-market mechanization things with interfacing with UI in any case at lower esteem point as contrast with today, and more people will be able to bear the cost of the things.

3. Solution commitments will bit by bit move to an all the more straightforward structure, where next to two or three key parts, customers will have the ability to buy and use the Automation things themselves without the guide of any specific ace.

4. Future Scope for the home automation system involves making homes even smarter. Homes can be interfaced with the sensors including the motion sensors, light sensors and temperature sensors and thus this may provide the automatic toggling of the devices according to the conditions

5. More energy can be conserved by ensuring occupation of the house before turning on devices and checking the brightness and turning off the light if not necessary.

6. The system can be integrated closely with the home security solutions enhancing the safety for home owners.

CHAPTER-IX

REFERENCE

9. REFERENCE

Reference:

1. https://www.security.org/home-automation/
2. https://en.wikipedia.org/wiki/Solid-state_relay
3. https://en.wikipedia.org/wiki/TRIAC
4. https://eepower.com/technical-articles/an-introduction-to-triac-basics/#
5. https://www.onsemi.com/pub/collateral/an-3004.pdf
6. https://internetofthingsagenda.techtarget.com/definition/IoT-device
7. https://www.onsemi.com/products/interfaces/triac-driver-optocouplers#overview
8. https://www.elprocus.com/acs712-current-sensor-working-and-applications/
9. https://pdf1.alldatasheet.com/datasheet-pdf/view/16753/PHILIPS/BT136-600.html
10. https://pdf1.alldatasheet.com/datasheet-pdf/view/85275/STMICROELECTRONICS/BTB16.html
11. https://www.evelta.com/esp32-wroom-32d-wi-fi-bt-ble-mcu-module-spi-flash-4mb-pcb-antenna/?gclid=Cj0KCQiAi9mPBhCJARIsAHchl1yAi73sqL3CGR5RwAgynkaGPDZ_0mUAebUeCWu2zadlYl-c-2KENH0aAjeOEALw_wcB
12. https://innovatorsguru.com/zmpt101b/
13. https://en.wikipedia.org/wiki/JST_connector
14. https://en.wikipedia.org/wiki/Screw_terminal
15. https://robu.in/product/ttp223-touch-key-module-2pcs/
16. https://www.electronicsforu.com/technology-trends/learn-electronics/veroboard-zero-pcb-difference
17. https://en.wikipedia.org/wiki/Arduino_IDE
18. https://en.wikipedia.org/wiki/EAGLE_(program)