A Project Report on

**IOT BASED UNLOCKING OF HOME AUTOMATION SYSTEM AND FACE DETECTION USING ESP32**

Submitted in partial fulfillment of the requirement for the award of the degree of

# BACHELOR OF TECHNOLOGY

# In

# ELECTRONICS AND COMMUNICATION ENGINEERING

By

K. MANIKANTA G.M.L. BHAVANI BH. SRI MADHURI (18A81A0426) (18A81A0416) (18A81A0466)

M. PURNASRI A. HEMANTH SREE SAI T. SUDHEER BABU

(18A81A0433) (18A81A0402) (18A81A0454)

Under the Esteemed Guidance of

## Dr.M.KOTESWARARAO M.Tech, Ph.D,

**Professor**



**DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING**

# SRI VASAVI ENGINEERING COLLEGE

**(Autonomous )**

(Affiliated to JNTUK & Accredited by NAAC- ‘A’ Grade)

Pedatadepalli, Tadepalligudem, West Godavari, AP -534101 **2018-2022**



DEPARTMENT OF

ELECTRONICS AND COMMUNICATION ENGINEERING

**CERTIFICATE**

This is to certify that the project report entitled “IOT BASED UNLOCKING OF HOME AUTOMATION SYSTEM AND FACE DETECTION USING ESP32”submitted by K.MANIKANTA (18A81A0426), G.M.L.BHAVANI (18A81A0416), Bh.SRIMADHURI (18A81A0466), M.PURNASRI (18A81A0433), A.HEMANTH SREE SAI (18A81A0402), T.SUDHEER BABU (18A81A0454) in partial fulfilment of the requirements for award of the Degree of Bachelor of Technology in Electronics and Communication Engineering, from Sri Vasavi Engineering College, Tadepalligudem affiliated to JNTU Kakinada and NAAC with ‘A’ grade is a record of bonafide work carried out by them under my guidance and supervision.

**PROJECT GUIDE HEAD OF THE DEPARTMENT**

**(Dr.M. KOTESWARA RAO) (Sri Dr.E. Kusuma Kumari)**

**EXTERNAL EXAMINER**

**DECLARATION**

We hereby declare that the project entitled **“IOT BASED UNLOCKING OF HOME AUTOMATION SYSTEM AND FACE DETECTION USING ESP32 ”** is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology in Electronics and Communication Engineering under the esteemed supervision of **Dr.M. KOTESWARA RAO M.Tech, Ph.D, Professor, Department of ECE.**

This a record of work carried out by us and results embodied in this project report have not been submitted to any other university for the award of any degree.

## 

## Project associates

K.MANIKANTA (18A81A0426)

G.M.L.BHAVANI (18A81A0416)

BH.SRI MADHURI (18A81A0466)

M.PURNASRI (18A81A0433)

A.HEMANTH SREE SAI (18A81A0402)

T.SUDHEER BABU (18A81A0454)

**ACKNOWLEDGEMENT**

We take this opportunity to express our profound sense of gratitude in all its humbleness to our beloved **Project Guide**, **Dr.M.KOTESWARA RAO, M.Tech, Ph.D Professor** in Electronics and Communication Engineering, Sri Vasavi Engineering College, Tadepalligudem. For his excellent guidance, meticulous care and enthusiastic encouragement in motivating us to take up this challenging task and helped us in the completion of the project.

We would like to express our gratitude to **Project Coordinator Dr.PURNIMA K SHARMA, M. Tech, Associate Professor** of our college for the excellent support and encouragement throughout this project.

We are indebted to **Dr.E.KUSUMA KUMARI, Ph.D., Professor & Head of the Department** of Electronics and Communication Engineering, for her constant encouragement during execution of the project.

We express our deep sense of gratitude to **Dr.G.V.N.S.R. RATNAKAR RAO, Principal, and Management of Sri Vasavi Engineering College** for providing necessary facilities to carry out our project work successfully.

We would like thank to our **Teaching and Nonteaching Staff** of Department of ECE, who have been directly and indirectly part of this journey, for their encouragement to complete our project work.

An Endeavour over long period can also be successful by constant effort and encouragement. We wish to take this opportunity to express our deep gratitude to all the people who have extended their cooperation in various ways during our project work. It is our pleasure to acknowledge the help of all those respected individuals.

## Project associates

K.MANIKANTA (18A81A0426)

G.M.L. BHAVANI (18A81A0416)

BH.SRI MADHURI (18A81A0466)

M.PURNASRI (18A81A0433)

A.HEMANTH SREE SAI (18A81A0402)

T.SUDHEER BABU (18A81A0454)

**INDEX**

**CONTENTS Page.No**

Abstract i

**CHAPTER-1: INTRODUCTION 1 - 7**

1.1 Embedded System Implementation……………………..1

1.2 Micro Processor vs Micro Controller ………………...3

1.3 Embedded System Software …………………………...4

1.4 Applications of Embedded Systems…………………....6

1.5 Implementation Flow …………………………………...6

**CHAPTER-2: 2.1 LITERATURE SURVEY 7 - 8**

**CHAPTER-3: WORKING DESCRIPTION 9 - 34**

3.1 Existing system…………………………………………..9

3.2 Proposed system………………………………………....9

3.3 Nodemcu ………………………………………………..10

3.4 Power Supply…………………………………………...14

3.5 Transformer……………………………………………...15

3.6 Rectifier……………………………………………….....16

3.7 Capacitor…………………………………………………17

3.8 Voltage Regulator……………………………………….17

3.9 ESP32 AI Thinker Cam………………………………..18

3.10 Relay……………………………………………………26

3.10.1 Relay basics………………………………………….29

3.10.2 Energized Relay (ON)……………………………….30

3.10.3 De-Energized Relay (OFF)………………………….31

3.11 Pole and Throw………………………………………..32

3.11.1 Relay Applications………………………………….33

3.11.2 Applications of Overload Relay…………………...33

3.11.3 Relay Selection……………………………………...34

**CHAPTER-4: SOFTWARE REQUIREMENT 35 - 41**

4.1 Adafruit cloud server…………………………………….35

4.2 Adafruit IO Setup………………………………………...36

4.3 IFTTT………………………………………………………41

**CHAPTER-5: RESULTS 46 - 47**

**CHAPTER-6: CONCLUSION AND FUTURE SCOPE**

6.1 Conclusion ………………………………………………46

**CHAPTER-7:-REFERENCES 47 - 48**

**ABSTRACT**

In this fast paced life of 21stcentury, automation plays a key role in human life. Home automation allows us to control household electrical appliances like light, door, fan, AC etc. It also provides home security and emergency system to be activated. Home automation not only refers to reduce human efforts but also energy efficiency and time saving. The main objective of home automation and security is to help handicapped and old aged people who will enable them to control home appliances and alert them in critical situations.

This project put forwards the implementation of home automation and security system using NodeMCU and ESP32 Camera. Home appliances are connected to the microprocessor and communication is established between the NodeMCU and Android mobile device or tablet via Cloud. This project would develop an authentication to the system for authorized person to access home appliances.

## **CHAPTER 1**

## **INTRODUCTION**

Today’s homes require sophistication control in its different gadgets which are basically electronic appliances. This has revolutionized the area of home automation with respect to an increased level of affordability and simplicity through the integration of home appliances with smart phone and tablet connectivity. Smart phones are already feature-perfect and can be made to communicate to any other devices in an ad hoc network with a connectivity options like cloud server. With the advent of mobile phones, Mobile applications development has seen a major outbreak.

**1.1 Embedded system implementation**

**Introduction:**

An embedded system is one kind of a computer system mainly designed to perform several tasks like to access, process, and store and also control the data in various electronics-based systems. Embedded systems are a combination of hardware and software where software is usually known as firmware that is embedded into the hardware. One of its most important characteristics of these systems is, it gives the o/p within the time limits. Embedded systems support to make the work more perfect and convenient. So, we frequently use embedded systems in simple and complex devices too. The applications of embedded systems mainly involve in our real life for several devices like microwave, calculators, TV remote control, home security and neighborhood traffic control systems, etc.

User interface

Embedded system

Software

Hardware

Inputs

Outputs

Link to other systems

Fig: Overview of embedded system

**Embedded system:**

Embedded system includes mainly two sections, they are

1. Hardware

2. Software

**Embedded System Hardware:**

As with any electronic system, an embedded system requires a hardware platform on which it performs the operation. Embedded system hardware is built with a microprocessor or microcontroller. The embedded system hardware has elements like input output (I/O) interfaces, user interface, memory and the display. Usually, an embedded system consists of:

* Power Supply
* Processor
* Memory
* Timers
* Serial communication ports
* Output/Output circuits
* System application specific circuits

Embedded systems use different processors for its desired operation. Some of the processors used are

1. Microprocessor

2. Microcontroller

3. Digital signal processor

**1.2 Micro Processor vs. Micro Controller**

**Microprocessor**

* **CPU** on a chip.
* We can attach required amount of ROM, RAM and I/O ports.
* Expensive due to external peripherals.
* Large in size
* general-purpose

**Microcontroller**

* **Computer** on a chip
* fixed amount of on-chip ROM, RAM, I/O ports
* Low cost.
* Compact in size.
* Specific –purpose

**1.3 Embedded System Software:**

The embedded system software is written to perform a specific function. It is typically written in a high level format and then compiled down to provide code that can be lodged within a non-volatile memory within the hardware. An embedded system software is designed to keep in view of the three limits:

* Availability of system memory
* Availability of processor’s speed
* When the system runs continuously, there is a need to limit power dissipation for events like stop, run and wake up.

**Bringing software and hardware together for embedded system:**

To make software to work with embedded systems we need to bring software and hardware together .for this purpose we need to burn our source code into microprocessor or microcontroller which is a hardware component and which takes care of all operations to be done by embedded system according to our code.

Generally we write source codes for embedded systems in assembly language, but the processors run only executable files. The process of converting the source code representation of your embedded software into an executable binary image involves three distinct steps:

1. Each of the source files must be compiled or assembled into an object file.
2. All of the object files that result from the first step must be linked together to produce a single object file, called the re-locatable program.
3. Physical memory addresses must be assigned to the relative offsets within the re-locatable program in a process called relocation.

The result of the final step is a file containing an executable binary image that is ready to run on the embedded system.

Source code

Assembler

Locator

Executable file

Linker

Processor

Flow of burning source code to processor

**1.4 Applications:**

Embedded systems have different applications. A few select [applications of embedded systems](https://www.elprocus.com/embedded-systems-real-time-applications/) are smart cards, telecommunications, satellites, missiles, digital consumer electronics, computer networking, etc.

[Embedded Systems in Automobiles](http://www.edgefx.in/importance-of-embedded-systems-in-automobiles-with-applications/)

* Motor Control System
* Engine or Body Safety
* [Robotics](http://www.edgefx.in/top-list-robotics-projects-for-engineering-beginners/) in Assembly Line
* Mobile and E-Com Access

Embedded systems in Telecommunications

* Mobile computing
* Networking
* [Wireless Communications](http://www.edgefx.in/multiple-input-and-multiple-output-mimo-wireless-communications/)

Embedded Systems in Smart Cards

* Banking
* Telephone
* [Security Systems](http://www.edgefx.in/microcontroller-based-projects-on-car-security-systems-using-gsm/)

**1.5 Implementation flow:**

**Stage 1:**

Considering the problems of existing methods and giving solution to that problem by considering the basic requirements for our proposed system

**Stage 2:**

Considering the hardware requirement for the proposed system

For this we need to select the below components:

1. Microcontroller

2. Inputs for the proposed system (ex: sensors, drivers etc..,)

3. Outputs (ex: relays, loads)

**Stage 3:**

After considering hardware requirements, now we need to check out the software requirements. Based on the microcontroller we select there exists different software for coding, compiling, debugging. we need to write source code for that proposed system based on our requirements and compile, debug the code in that software .

After completing all the requirements of software and hardware we need to bring both together to work our system. For this we need to burn our source code into microcontroller, after burning our source code to microcontroller then connect all input and output modules as per our requirement.

## CHAPTER 2

## LITERATURE SURVEY

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.

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.

## CHAPTER 3

## WORKING DESCRIPTION

## 

**3.1 EXISTING SYSTEM**

In the existing system, home appliances are controlled through Zigbee and other devices. But they are limited to certain area. In the proposed system we have designed so as to control home appliances from anywhere in the world.

DTMF can also be used but it won’t come for practically and also not reliable.

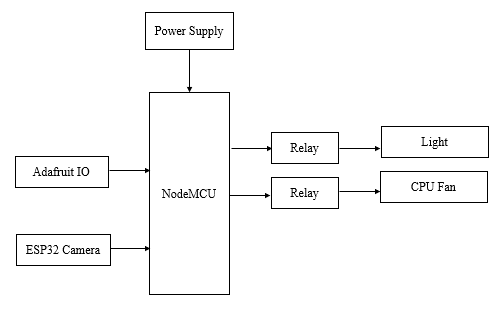
**Drawbacks:**

* Long distance communication is not done
* No Android application is involved so we cannot operate from long ranges.

**3.2 PROPOSED SYSTEM**

In this project we are using NodeMCU as the main microcontroller and that is connected to cloud server which is Adafruit IO. Through that home appliances will be controlled. Here we are implementing face recognition also in order to know whether the person is authorized or not.

**Block Diagram:**



**3.3 NodeMCU:**

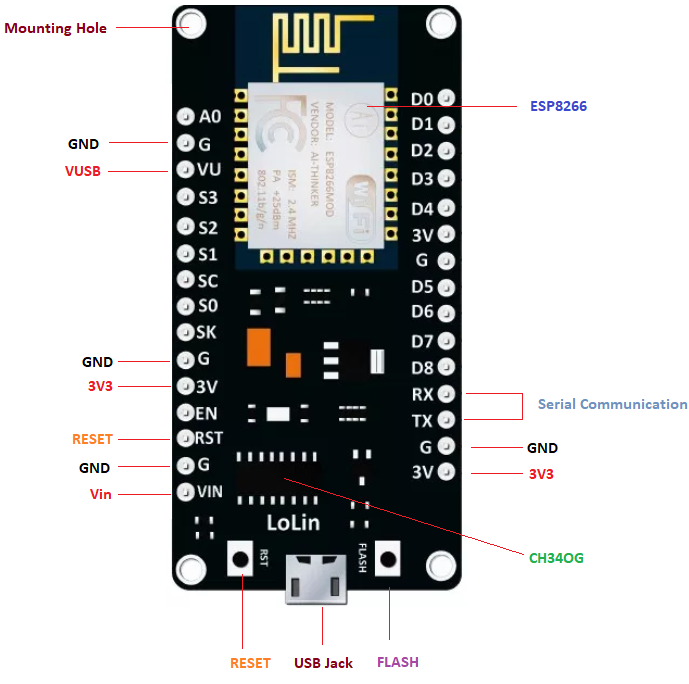
**Introduction to NodeMCU**

NodeMCU is an open-source firmware and development kit that plays a vital role in designing your own IoT product using a few Lua script lines.

Multiple GPIO pins on the board allow you to connect the board with other peripherals and are capable of generating PWM, I2C, SPI, and UART serial communications.

* The interface of the module is mainly divided into two parts including both Firmware and Hardware where former runs on the ESP8266 Wi-Fi SoC and later is based on the ESP-12 module.

The firmware is based on Lua – A scripting language that is easy to learn, giving a simple programming environment layered with a fast scripting language that connects you with a well-known developer community.



And open source firmware gives you the flexibility to edit, modify and rebuilt the existing module and keep changing the entire interface until you succeed in optimizing the module as per your requirements.

* USB to UART converter is added on the module that helps in converting USB data to UART data which mainly understands the language of serial communication.

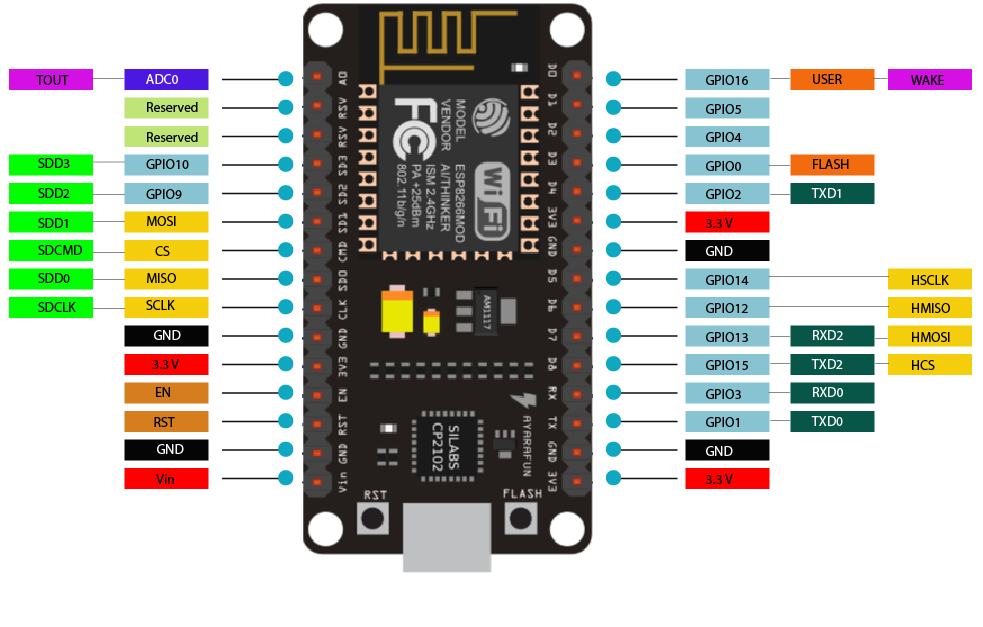
Instead of the regular USB port, MicroUSB port is included in the module that connects it with the computer for dual purposes: programming and powering up the board.

* The board incorporates status LED that blinks and turns off immediately, giving you the current status of the module if it is running properly when connected with the computer.

The ability of module to establish a flawless WiFi connection between two channels makes it an ideal choice for incorporating it with other embedded devices like Raspberry Pi.

**NodeMCU Pinout:**

NodeMCU comes with a number of GPIO Pins. Following figure shows the Pinout of the board.



* There is a candid difference between VIN and VU where former is the regulated voltage that may stand somewhere between 7 to 12 V while later is the power voltage for USB that must be kept around 5 V.

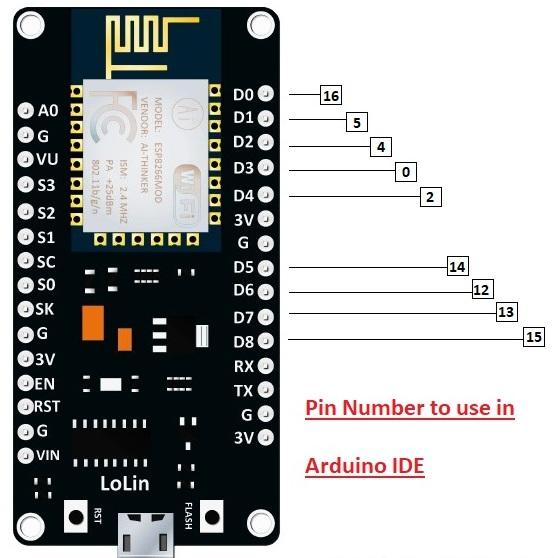
**Features:**

* Open-source
* Arduino-like hardware
* Status LED
* MicroUSB port
* Reset/Flash buttons
* Interactive and Programmable
* Low cost
* ESP8266 with inbuilt wifi
* USB to UART converter
* GPIO pins

As mentioned above, a cable supporting micro USB port is used to connect the board. As you connect the board with a computer, LED will flash. You may need some drivers to be installed on your computer if it fails to detect the NodeMCU board. You can download the driver from [this](https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers) page.

**Note:** We use [Arduino IDE](https://www.theengineeringprojects.com/2018/10/introduction-to-arduino-ide.html) software for programming this module. It is important to note that the pin configuration appearing on the board is different from the configuration we use to program the board on the software i.e. when we write code for targeting pin 16 on the Arduino IDE, it will actually help is laying out the communication with the D0 pin on the module.

Following figure the shows the pin configuration to use in Arduino IDE.



**How to Power NodeMCU**

You can see from the pinout image above, there are five ground pins and three 3V3 pins on the board. The board can be powered up using the following three ways.

**USB Power.** It proves to an ideal choice for loading programs unless the project you aim to design requires separate interface i.e. disconnected from the computer.

**Provide 3.3V.** This is another great option to power up the module. If you have your own off-board regulator, you can generate an instant power source for your development kit.

**Power Vin.** This is a voltage regulator that comes with the ability to support up to 800 mA. It can handle somewhere between 7 to 12 V. You cannot power the devices operating at 3.3 V, as this regulator unable to generate as low as 3.3V.

**Applications:**

NodeMCU V3 is mainly used in the WiFi Applications which most of the other embedded modules fail to process unless incorporated with some external WiFi protocol. Following are some major applications used for NodeMCU V3.

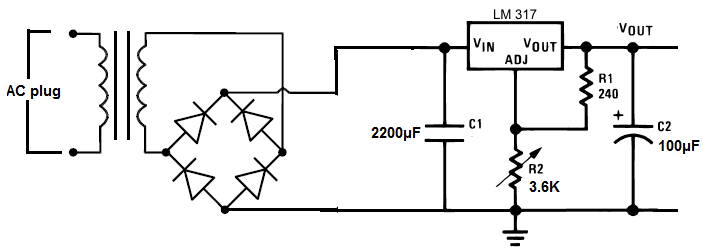
* Internet Smoked Alarm
* VR Tracker
* Octopod
* Serial Port Monitor
* ESP Lamp
* Incubator Controller
* IoT home automation
* Security Alarms

**3.4 Power supply:**

A power supply is a component that provides at least one electrical charge with power. It typically converts one type of electrical power to another, but it can also convert a different Energy form in electrical energy, such as solar, mechanical, or chemical.

A power supply provides electrical power to components. Usually the term refers to devices built into the powered component. Computer power supplies, for example, convert AC current to DC current and are generally located along with at least one fan at the back of the computer case.

Most computer power supplies also have an input voltage switch that, depending on the geographic location, can be set to 110v/115v or 220v/240v. Due to the different power voltages supplied by power outlets in different countries, this switch position is crucial.

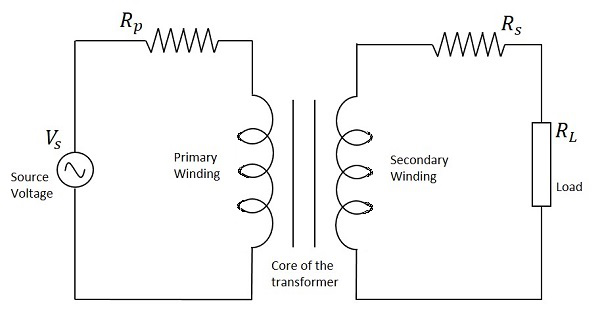


Some basic components used in the supply of power:

**3.5 Transformer:**

A transformer is a static electrical gadget that exchanges control between at least two circuits. A fluctuating current creates a changing attractive motion in one transformer curl, which thus actuates a differing electromotive power over a second loop twisted around a similar center.

Without a metallic association between the two circuits, electrical vitality can be exchanged between the two loops. The enlistment law of Faraday found in 1831 portrayed the impact of prompted voltage in any curl because of the changing attractive flux surrounded by the coil.



**Circuit of transformer**

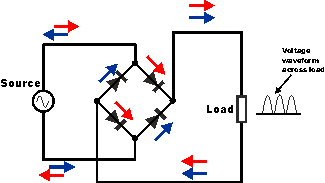
****

**Transformer**

**3.6 Rectifier:**

A **rectifier** is an electrical device that [converts](https://en.wikipedia.org/wiki/Electric_power_conversion) [alternating current](https://en.wikipedia.org/wiki/Alternating_current) (AC), which periodically reverses direction, to [direct current](https://en.wikipedia.org/wiki/Direct_current) (DC), which flows in only one direction. The process is known as *rectification*, since it "straightens" the direction of current.

Rectifiers have many uses, but are often found to serve as components of DC power supplies and direct power transmission systems with high voltage. Rectification can be used in roles other than direct current generation for use as a power source.

****

**Circuit of rectifier**

****

**Rectifier**

**3.7 Capacitors:**

Capacitors are used to attain from the connector the immaculate and smoothest DC voltage in which the rectifier is used to obtain throbbing DC voltage which is used as part of the light of the present identity. Capacitors are used to acquire square DC from the current AC experience of the current channels so that they can be used as a touch of parallel yield.

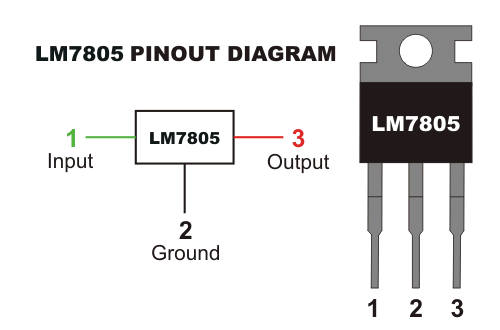


**Capacitor**

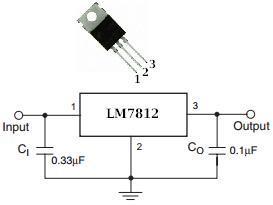
**3.8 Voltage regulators:**

The 78XX voltage controller is mainly used for voltage controllers as a whole. The XX speaks to the voltage delivered to the specific gadget by the voltage controller as the yield. 7805 will supply and control 5v yield voltage and 12v yield voltage will be created by 7812.

The voltage controllers are that their yield voltage as information requires no less than 2 volts. For example, 7805 as sources of information will require no less than 7V, and 7812, no less than 14 volts. This voltage is called Dropout Voltage, which should be given to voltage controllers.



**7805 voltage regulator with pinout**



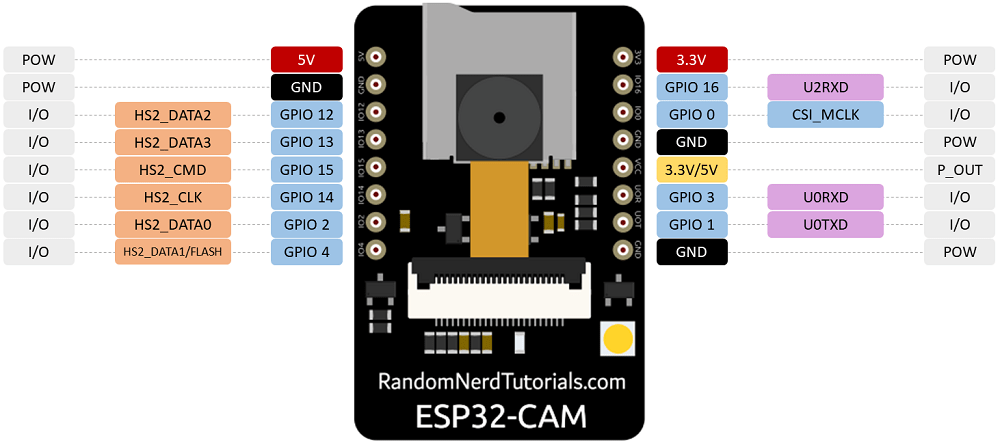
**7812 voltage regulator with pinout**

**3.9 ESP32 AI THINKER BOARD:**

The ESP32-CAM is a development board with an ESP32-S chip, an OV2640 camera, microSD card slot and several GPIOs to connect peripherals.

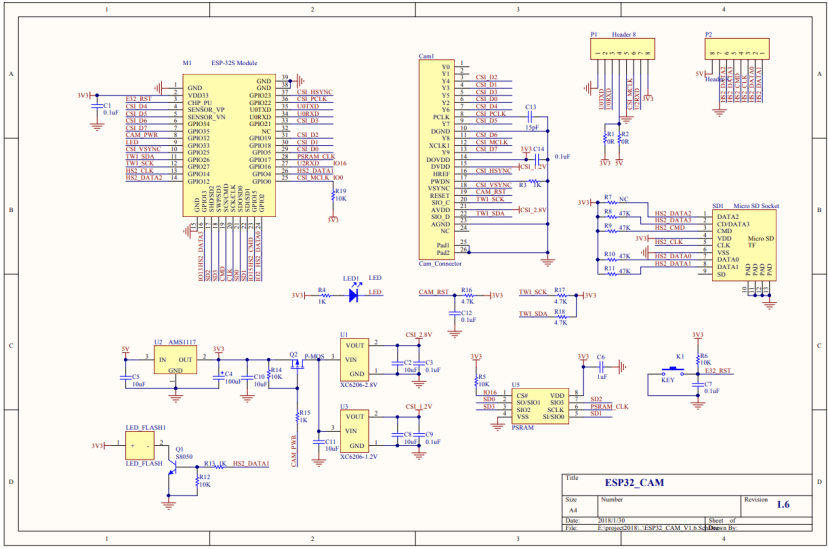
## Pinout Diagram

The following image shows the pinout diagram for the [ESP32-CAM AI-Thinke](https://makeradvisor.com/tools/esp32-cam-external-antenna/)r



## Schematic Diagram

The following figure shows the schematic diagram for the ESP32-CAM.



## Power Pins

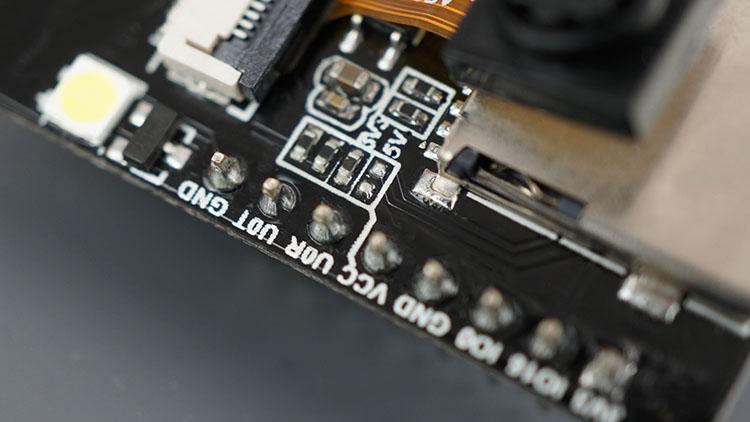
The ESP32-CAM comes with three GND pins (colored in black color) and two power pins (colored with red color): 3.3V and 5V.

You can power the ESP32-CAM through the 3.3V or 5V pins. However, many people reported errors when powering the ESP32-CAM with 3.3V, so we always advise to **power the ESP32-CAM through the 5V pin**.

### Power output pin

There’s also the pin labeled on the silkscreen as **VCC** (colored with a yellow rectangle). You should not use that pin to power the ESP32-CAM. That is an output power pin. It can either output 5V or 3.3V.

In our case, the ESP32-CAM outputs 3.3V whether it is powered with 5V or 3.3V. Next to the VCC pin, there are two pads. One labeled as 3.3V and other as 5V.



If you look closely, you should have a jumper on the 3.3V pads. If you want to have an output of 5V on the VCC pin, you need to unsolder that connection and solder the 5V pads.

## Serial Pins

GPIO 1 and GPIO 3 are the serial pins (TX and RX, respectively). Because the ESP32-CAM doesn’t have a built-in programmer, you need to use these pins to communicate with the board and upload code.

The best way to upload code to the ESP32-CAM is using an [FTDI programmer](https://makeradvisor.com/tools/ftdi-programmer-board/).

[Learn how to upload code to the ESP32-CAM AI-Thinker.](https://randomnerdtutorials.com/program-upload-code-esp32-cam/)

You can use GPIO 1 and GPIO 3 to connect other peripherals like outputs or sensors after uploading the code. However, you won’t be able to open the Serial Monitor and see if everything is going well with your setup.

## GPIO 0

GPIO 0 determines whether the ESP32 is in flashing mode or not. This GPIO is internally connected to a pull-up 10k Ohm resistor.

When GPIO 0 is connected to GND, the ESP32 goes into flashing mode and you can upload code to the board.

GPIO 0 connected to GND » ESP32-CAM in flashing mode.To make the ESP32 run “normally”, you just need to disconnect GPIO 0 from GND.

## MicroSD Card Connections

|  |  |
| --- | --- |
| **MicroSD card** | **ESP32** |
| CLK | GPIO 14 |
| CMD | GPIO 15 |
| DATA0 | GPIO 2 |
| DATA1 / flashlight | GPIO 4 |
| DATA2 | GPIO 12 |
| DATA3 | GPIO 13 |

The following pins are used to interface with the microSD card when it is on operation.

If you’re not using the microSD card, you can use these pins as regular inputs/outputs. You can take a look at the [ESP32 pinout guide](https://randomnerdtutorials.com/esp32-pinout-reference-gpios/) to see the features of these pins.

All these GPIOs are RTC and support ADC: GPIOs 2, 4, 12, 13, 14, and 15.

## Flashlight (GPIO 4)

The ESP32-CAM has a very bright built-in LED that can work as a flash when taking  
photos. That LED is internally connected to GPIO 4.

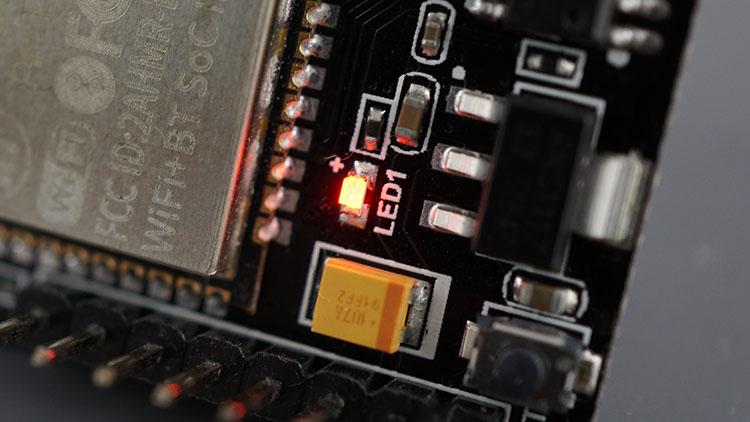
That GPIO is also connected to the microSD card slot, so you may have troubles when trying to use both at the same time – the flashlight will light up when using the microSD card.

**Note:** one of our readers shared that if you initialize the microSD card as follows, you won’t have this problem because the microSD card won’t use that data line.\*

SD\_MMC.begin("/sdcard", true)

\* we found that this works and that the LED will not make that flash effect. However, the LED remains on with low brightness – we’re not sure if we are missing something.

## GPIO 33 – Built-in Red LED



Next to the RST button, there’s an on-board red LED. That LED is internally connected to GPIO 33. You can use this LED to indicate that something is happening. For example, if the Wi-Fi is connected, the LED is red or vice-versa.

That LED works with inverted logic, so you send a LOW signal to turn it on and a HIGH signal to turn it off.

## Camera Connections

The connections between the camera and the ESP32-CAM AI-Thinker are shown in the following table.

|  |  |  |
| --- | --- | --- |
| **OV2640 CAMERA** | **ESP32** | **Variable name in code** |
| D0 | GPIO 5 | Y2\_GPIO\_NUM |
| D1 | GPIO 18 | Y3\_GPIO\_NUM |
| D2 | GPIO 19 | Y4\_GPIO\_NUM |
| D3 | GPIO 21 | Y5\_GPIO\_NUM |
| D4 | GPIO 36 | Y6\_GPIO\_NUM |
| D5 | GPIO 39 | Y7\_GPIO\_NUM |
| D6 | GPIO 34 | Y8\_GPIO\_NUM |
| D7 | GPIO 35 | Y9\_GPIO\_NUM |
| XCLK | GPIO 0 | XCLK\_GPIO\_NUM |
| PCLK | GPIO 22 | PCLK\_GPIO\_NUM |
| VSYNC | GPIO 25 | VSYNC\_GPIO\_NUM |
| HREF | GPIO 23 | HREF\_GPIO\_NUM |
| SDA | GPIO 26 | SIOD\_GPIO\_NUM |
| SCL | GPIO 27 | SIOC\_GPIO\_NUM |
| POWER PIN | GPIO 32 | PWDN\_GPIO\_NUM |

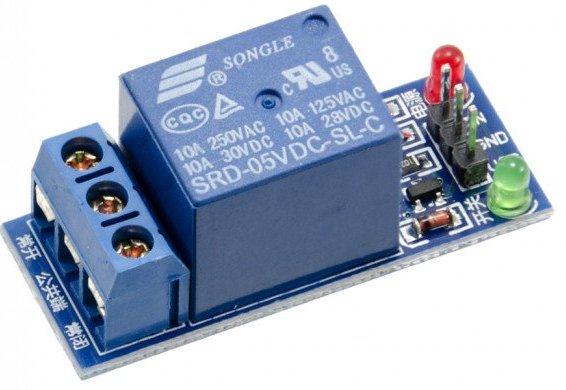


**3.10 Relay:**

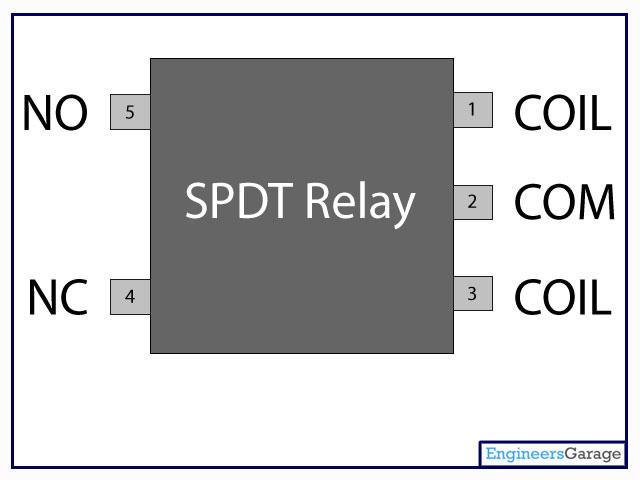
**What is a relay?**

A relay is an electromagnetic switch that is used to turn on and turn off a circuit by a low power signal, or where several circuits must be controlled by one signal.

Most of the high end industrial application devices have relays for their effective working. Relays are simple switches which are operated both electrically and mechanically. Relays consist of an electromagnet and also a set of contacts. The switching mechanism is carried out with the help of the electromagnet. There are also other operating principles for its working. But they differ according to their applications. Most of the devices have the application of relays.



### **Pin Diagram:**



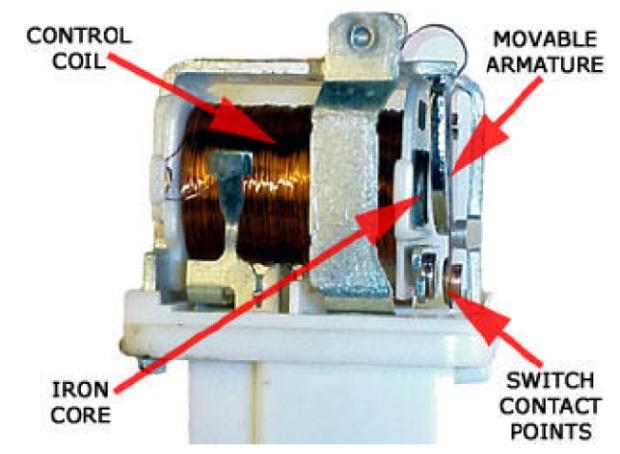
**Why is a relay used?**

The main operation of a relay comes in places where only a low-power signal can be used to control a circuit. It is also used in places where only one signal can be used to control a lot of circuits. The application of relays started during the invention of telephones. They played an important role in switching calls in telephone exchanges. They were also used in long distance telegraphy. They were used to switch the signal coming from one source to another destination. After the invention of computers they were also used to perform Boolean and other logical operations. The high end applications of relays require high power to be driven by electric motors and so on. Such relays are called contactors.

### **Relay Design**

* There are only four main parts in a relay. They are
* Electromagnet
* Movable Armature
* Switch point contacts
* Spring

The figures given below show the actual design of a simple relay.



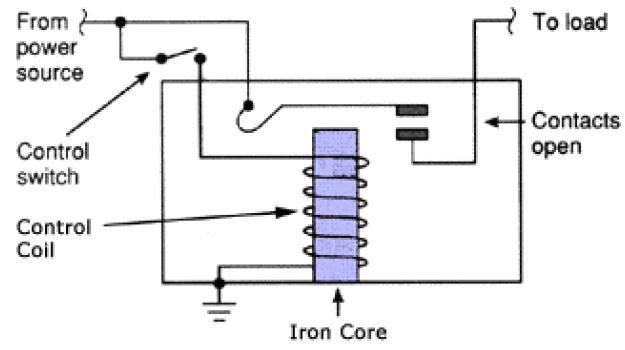
**Relay Construction**

It is an electro-magnetic relay with a wire coil, surrounded by an iron core. A path of very low reluctance for the magnetic flux is provided for the movable armature and also the switch point contacts.

The movable armature is connected to the yoke which is mechanically connected to the switch point contacts. These parts are safely held with the help of a spring. The spring is used so as to produce an air gap in the circuit when the relay becomes de-energized.

### **How relay works?**

The relay function can be better understood by explaining the following diagram given below.



**Relay Design**

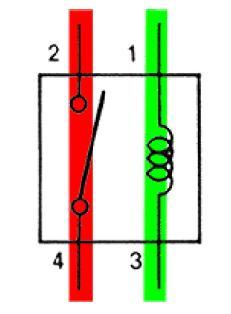
The diagram shows an inner section diagram of a relay. An iron core is surrounded by a control coil. As shown, the power source is given to the electromagnet through a control switch and through contacts to the load. When current starts flowing through the control coil, the electromagnet starts energizing and thus intensifies the magnetic field. Thus the upper contact arm starts to be attracted to the lower fixed arm and thus closes the contacts causing a short circuit for the power to the load. On the other hand, if the relay was already de-energized when the contacts were closed, then the contact move oppositely and make an open circuit.

As soon as the coil current is off, the movable armature will be returned by a force back to its initial position. This force will be almost equal to half the strength of the magnetic force. This force is mainly provided by two factors. They are the spring and also gravity.

Relays are mainly made for two basic operations. One is low voltage application and the other is high voltage. For low voltage applications, more preference will be given to reduce the noise of the whole circuit. For high voltage applications, they are mainly designed to reduce a phenomenon called arcing.

### **3.10.1 Relay Basics**

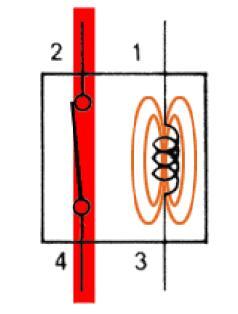
The basics for all the relays are the same. Take a look at a 4 pin relay shown below. There are two colors shown. The green color represents the control circuit and the red color represents the load circuit. A small control coil is connected onto the control circuit. A switch is connected to the load. This switch is controlled by the coil in the control circuit. Now let us take the different steps that occur in a relay.



**Relay operation**

* **3.10.2 Energized Relay (ON)**

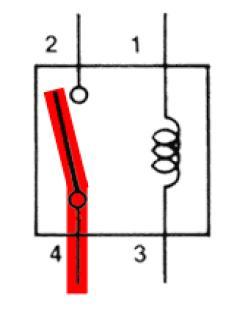
As shown in the circuit, the current flowing through the coils represented by pins 1 and 3 causes a magnetic field to be aroused. This magnetic field causes the closing of the pins 2 and 4. Thus the switch plays an important role in the relay working. As it is a part of the load circuit, it is used to control an electrical circuit that is connected to it. Thus, when the electrical relay in energized the current flow will be through the pins 2 and 4.



**Energized Relay (ON)**

* **3.10.3 De – Energized Relay (OFF)**

As soon as the current flow stops through pins 1 and 3, the relay switch opens and thus the open circuit prevents the current flow through pins 2 and 4. Thus the relay becomes de-energized and thus in off position.



**De-Energized Relay (OFF)**

In simple, when a voltage is applied to pin 1, the electromagnet activates, causing a magnetic field to be developed, which goes on to close the pins 2 and 4 causing a closed circuit. When there is no voltage on pin 1, there will be no electromagnetic force and thus no magnetic field. Thus the switches remain open.

### **3.11 Pole and Throw**

Relays have the exact working of a switch. So, the same concept is also applied. A relay is said to switch one or more poles. Each pole has contacts that can be thrown in mainly three ways. They are

* **Normally Open Contact (NO):** NO contact is also called a make contact. It closes the circuit when the relay is activated. It disconnects the circuit when the relay is inactive.
* **Normally Closed Contact (NC):** NC contact is also known as break contact. This is opposite to the NO contact. When the relay is activated, the circuit disconnects. When the relay is deactivated, the circuit connects.
* **Change-over (CO) / Double-throw (DT) Contacts:** This type of contacts are used to control two types of circuits. They are used to control a NO contact and also a NC contact with a common terminal. According to their type they are called by the names **break before make** and **make before break** contacts.

Relays can be used to control several circuits by just one signal. A relay switches one or more poles, each of whose contacts can be thrown by energizing the coil.

Relays are also named with designations like

* **Single Pole Single Throw (SPST)**: The SPST relay has a total of four terminals. Out of these two terminals can be connected or disconnected. The other two terminals are needed for the coil to be connected.
* **Single Pole Double Throw (SPDT):** The SPDT relay has a total of five terminals. Out of these two are the coil terminals. A common terminal is also included which connects to either of two others.
* **Double Pole Single Throw (DPST):** The DPST relay has a total of six terminals. These terminals are further divided into two pairs. Thus they can act as two SPST which are actuated by a single coil. Out of the six terminals two of them are coil terminals.
* **Double Pole Double Throw (DPDT)**: The DPDT relay is the biggest of all. It has mainly eight relay terminals. Out of these two rows are designed to be change over terminals. They are designed to act as two SPDT relays which are actuated by a single coil.

### **3.11.1 Relay Applications**

* A relay circuit is used to realize logic functions. They play a very important role in providing safety critical logic.
* Relays are used to provide time delay functions. They are used to time the delay open and delay close of contacts.
* Relays are used to control high voltage circuits with the help of low voltage signals. Similarly they are used to control high current circuits with the help of low current signals.
* They are also used as protective relays. By this function all the faults during transmission and reception can be detected and isolated.

#### **3.11.2 Application of Overload Relay**

Overload relay is an electro-mechanical device that is used to safeguard motors from overloads and power failures. Overload relays are installed in motors to safeguard against sudden current spikes that may damage the motor. An overload relay switch works in characteristics with current over time and is different from circuit breakers and fuses, where a sudden trip is made to turn off the motor. The most widely used overload relay is the thermal overload relay where a bimetallic strip is used to turn off the motor. This strip is set to make contact with a contactor by bending itself with rising temperatures due to excess current flow. The contact between the strip and the contactor causes the contactor to de-energize and restricts the power to the motor, and thus turns it off.

Another type of overload motor is the electronic type which continuously watches the motor current, whereas the thermal overload relay shuts off the motor depending on the rise of temperature/heat of the strip.

All overload relays available to buy comes in different specifications, the most important of them being the current ranges and response time. Most of them are designed to automatically reset to work after the motor is turned back on.

### **3.11.3 Relay Selection**

You must note some factors while selecting a particular relay. They are

* Protection Different protections like contact protection and coil protection must be noted. Contact protection helps in reducing arcing in circuits using inductors. Â Coil protection helps in reducing surge voltage produced during switching.
* Look for a standard relay with all regulatory approvals.
* Switching time Ask for high speed switching relays if you want one.
* Ratings There are current as well as voltage ratings. The current ratings vary from a few amperes to about 3000 amperes. Â In case of voltage ratings, they vary from 300 Volt AC to 600 Volt AC. There are also high voltage relays of about 15,000 Volts.
* Type of contact used whether it is a NC or NO or closed contact.
* Select Make before Break or Break before Make contacts wisely.
* Isolation between coil circuit and contacts

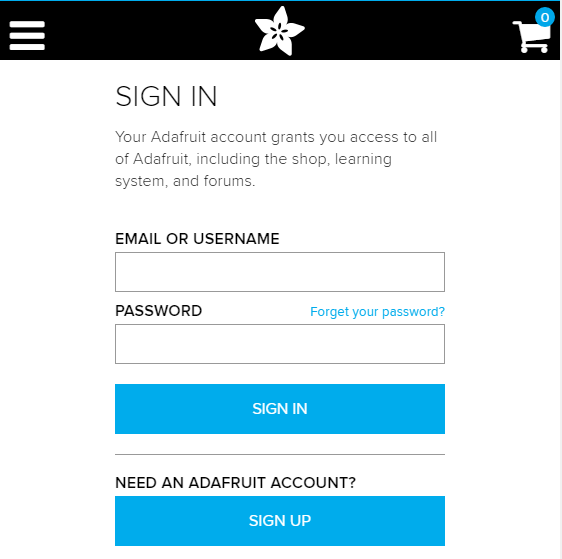
## CHAPTER 4

## SOFTWARE REQUIREMENT

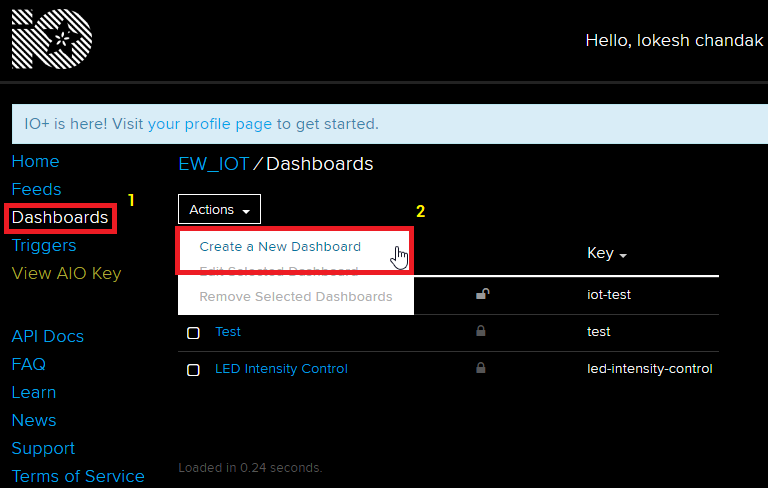
**4.1 Adafruit**:

Adafruit Industries is an open-source hardware company based in New York City. It was founded by Limor Fried in 2005. The company designs, manufactures and sells a number of electronics products, electronics components, tools and accessories. It also produces a number of learning resources, including live and recorded videos related to electronics, technology, and programming. MQTT, or message queue telemetry transport, is a protocol for device communication that Adafruit IO supports. Using a MQTT library or client you can publish and subscribe to a feed to send and receive feed data.

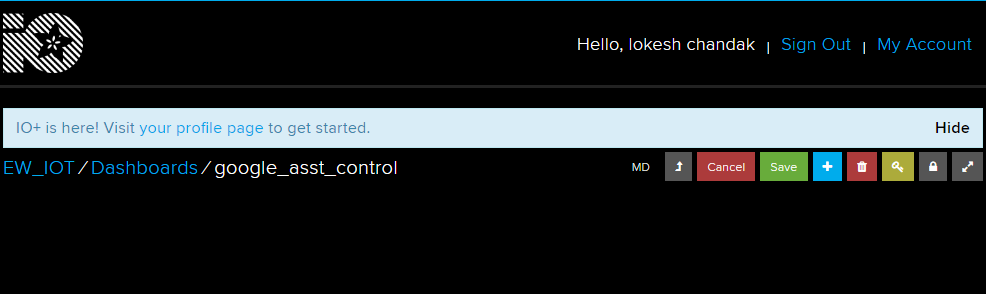
# **4.2 Adafruit IO Steps**

First, created account at [www.Adafruit.io](http://www.adafruit.io/)

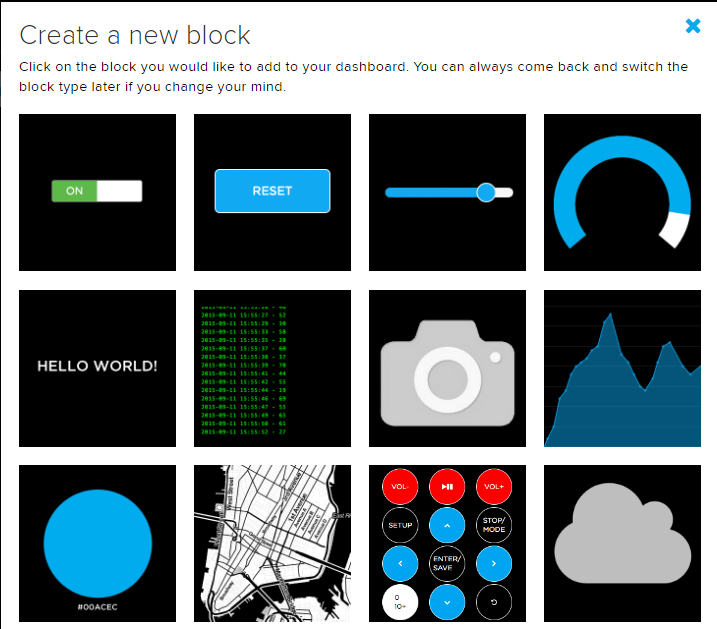
Now, create dashboard at Adafruit. This dashboard is a user interface to control things remotely.



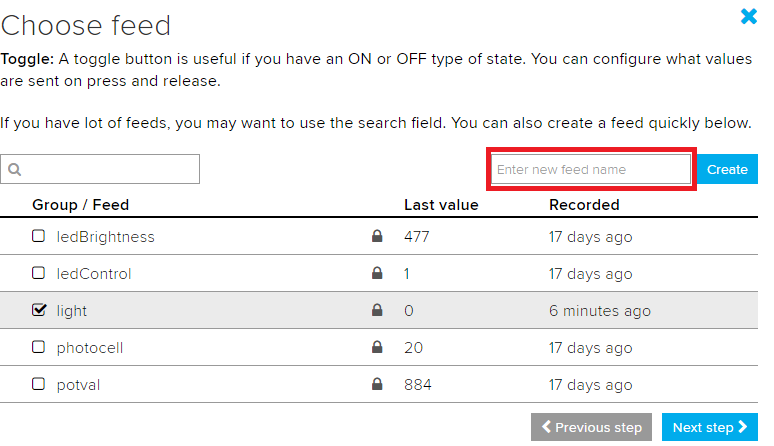
After following above steps, provide name to the dashboard and save it. We can see our dashboard as follows,



Now, create feed (user interface) to control light On-Off. To create it, just click on **‘+’** symbol and select toggle feed shown below,

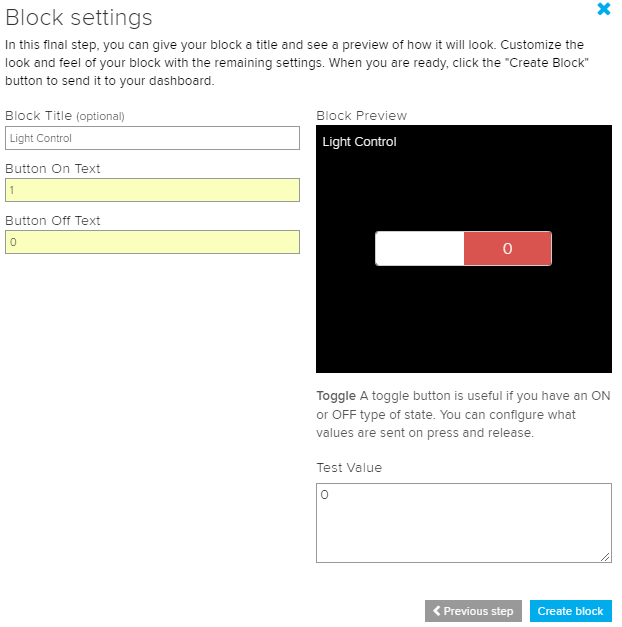


After selecting toggle feed, pop-up window appears as shown below.

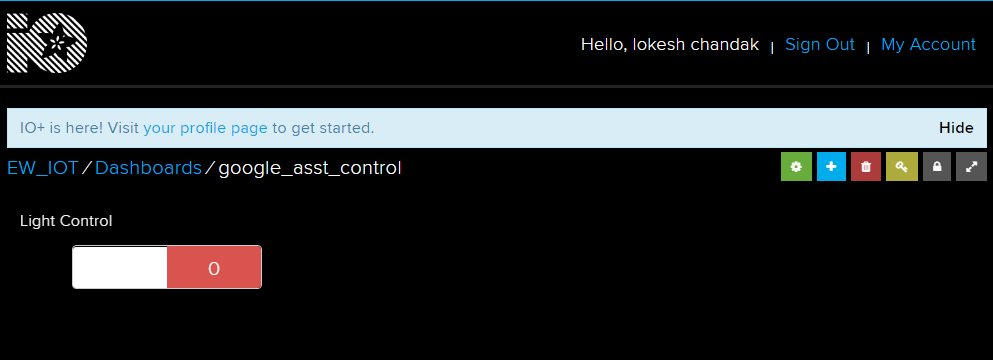


Enter name of our feed (shown in red box) and create it. After creation, select the created feed (here mine is **light**) and then click on **Next step.**

In the next step configure the feed which is shown below,



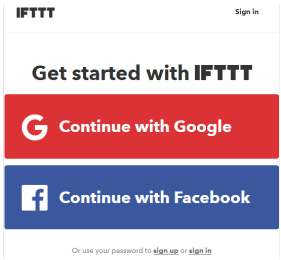
Here, I used **0**(OFF) and **1**(ON) text for button and then click on create. This will create toggle button on your dashboard which can be used to control things remotely.



**4.3 IFTTT:**

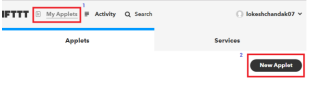
IFTTT (If This Then That) If This Then That, also known as IFTTT is a free web-based service to create chains of simple conditional statements, called applets. An applet is triggered by changes that occur within other web services such as Gmail, Facebook, Telegram, Instagram, or Pinterest. For example, an applet may send an e-mail message if the user tweets using a hashtag or copy a photo on Facebook to a user's archive if someone tags a user in a photo. Here, IFTTT is used to use google assistant service and Adafruit service in chain. So, Google assistant is used to control light of my home by saying Ok google, turn the light ON or OFF. Then IFTTT interpret the message and can send it to Adafruit’s dashboard as an understandable command to the created feed.

First step is creating account on IFTTT.

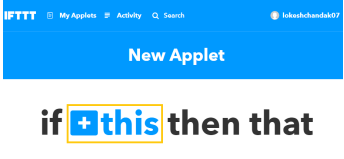


Note: Create account on IFTTT by using same e-mail id which have been used for Adafruit.

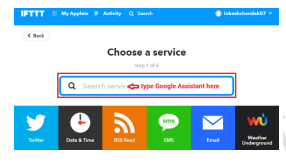
After account creation, click on My Applets and then select New Applet shown below,



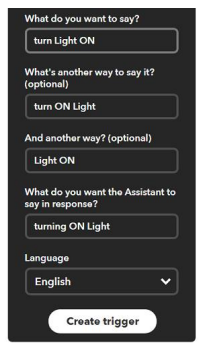
After selecting a new applet, we get a new page in which we should click on to This



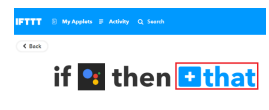
Then search for Google Assistant and select it.



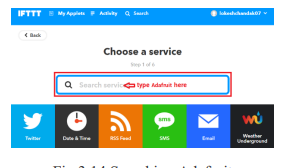
Searching Google assistant Now, enter voice phrases which will be used as a command for google assistant.



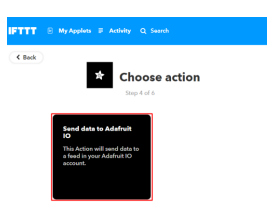
Any phrase can be entered as per the application. The phrases entered in the showing trigger fields is for making Light ON. For making Light OFF, another applet with different phrases have to be created. Now, another page will be shown as shown in Fig 3.13 in which user have to click on that option which is used to connect Google Assistant with Adafruit.



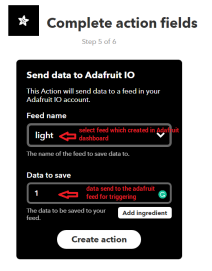
Then search for Adafruit and select it.



After selecting Adafruit



Now enter what data needed to send to which feed of Adafruit dashboard.



Click on Create Action. So, when Google Assistant is used on my mobile. This will trigger the event on Adafruit dashboard which is continuously monitored by the microcontroller (here NodeMCU). This microcontroller will take action as per the data change on the Adafruit dashboard.

**ADVANTAGES**

* Portable
* Low cost
* Easy to use

**APPLICATIONS**

* Home automation – This project can be used to control various Home Appliances
* We can control device from a long distance; thus, it gives ease of access.
* Faster operation and efficient.
* No need to carry separate remote or any other controlling unit.

## CHAPTER 5

## RESULTS/FUTURE SCOPE

**Circuit Connection Diagram**

 When system detects the authorized face, it shows like the “Hello subject 0”.



When system detects the unauthorized face it will give Intruder Alert.



CHAPTER 6

## CONCLUSION & FUTURE SCOPE

By the end of successful completion of this work it was concluded that through IOT we can connect multiple input/ output devices, multiple sensors and actuators in a network so that they can talk to each other, data acquired from these things can be used to keep a log or monitor or control the other things without human intervention and much more. Thus, IOT is like global networks which give the communication between things to things, human to things and human to human. IOT is the development of existing internet facility to manage everything which exists in the world or exists in the future.

**FUTURE SCOPE :**

The various future applications may be used by controlling various household devices of house with internet.In the future work, the persons other than the residents will also be considered. o Industrial automation and management through internet.Improvement of security problems Used in extremely restricted areas

**REFERENCES**

[1] M. Daneshvar, M. Pesaran, and B. Mohammadi-Ivatloo, ‘‘Transactive

energy in future smart homes,’’ in The Energy Internet. Amsterdam,

The Netherlands: Elsevier, 2019, pp. 153–179.

[2] R. Sinha, S. Patil, L. Gomes, and V. Vyatkin, ‘‘A survey of static formal

methods for building dependable industrial automation systems,’’ IEEE

Trans. Ind. Informat., vol. 15, no. 7, pp. 3772–3783, Jul. 2019.

[3] W. A. Jabbar, M. H. Alsibai, N. S. S. Amran, and S. K. Mahayadin, ‘‘Design

and implementation of IoT-based automation system for smart home,’’ in

Proc. Int. Symp. Netw., Comput. Commun.(ISNCC), Jun. 2018, pp. 1–6.

[4] H. Ning, F. Shi, T. Zhu, Q. Li, and L. Chen, ‘‘A novel ontology consistent

with acknowledged standards in smart homes,’’ Comput. Netw., vol. 148,

pp. 101–107, Jan. 2019.

[5] W. Li, T. Logenthiran, V.-T. Phan, and W. L. Woo, ‘‘A novel smart energy

theft system (SETS) for IoT-based smart home,’’ IEEE Internet Things J.,

vol. 6, no. 3, pp. 5531–5539, Jun. 2019.

[6] W. A. Jabbar, W. K. Saad, and M. Ismail, ‘‘MEQSA-OLSRv2:

A multicriteria-based hybrid multipath protocol for energy-efficient and

QoS-aware data routing in MANET-WSN convergence scenarios of IoT,’’

IEEE Access, vol. 6, pp. 76546–76572, 2018.

[7] D. Anandhavalli, N. S. Mubina, and P. Bharathi, ‘‘Smart home automation

control using Bluetooth and GSM,’’ Int. J. Informative Futuristic Res.,

vol. 2, no. 8, pp. 2547–2552, 2015.

[8] B. Davidovic and A. Labus, ‘‘A smart home system based on sensor

technology,’’ Facta Universitatis, Ser., Electron. Energetics, vol. 29, no. 3,

pp. 451–460, 2015.

[9] N. David, A. Chima, A. Ugochukwu, and E. Obinna, ‘‘Design of a home

automation system using arduino,’’ Int. J. Sci. Eng. Res., vol. 6, no. 6,

pp. 795–801, 2015.

[10] R. K. Kodali and S. Soratkal, ‘‘MQTT based home automation system

using ESP8266,’’ in Proc. IEEE Region 10 Humanitarian Technol. Conf.

(R10-HTC), Dec. 2016, pp. 1–5.

[11] A. ElShafee and K. A. Hamed, ‘‘Design and implementation of a WIFI

based home automation system,’’ World Acad. Sci., Eng. Technol., vol. 68,

pp. 2177–2180, Aug. 2012.