Home Automation Project Using Voice and App

ABSTRACT

The Aim of the project is to build a complete and secure home automation system Using micro controller that being controlled by Android app and Voice assistant. It uses Wi-Fi as a communication medium in a secure way. Here the ESP32 is used as a micro controller to automate the home appliance.

The Solar energy is used to power up the home because solar power is 100% clean, renewable energy source. It reduces reliance on oil, coal and natural gas for electricity production. The fossil fuels produce harmful emission that affects air, water, and leads to global warming. To get more energy from Solar, the Solar tracking mechanism is used and it is controlled by a micro controller (Arduino uno).

As the internet of things and automation increased the dark side of them also increased. The dark side is called as hacking. To prevent and secure from hackers some modules are implemented like finger print authentication, Encrypted communication and CSRF is used to protect. Some hacking techniques are also performed like DDOS, rolling code attack, Blue borne attack, jamming attack, back door, man in the middle and sniffing. For the physical security we use surveillance camera, motion detector and laser light protection to prevent from any instruction

The person's healthcare is also monitored by voice assistant and reminds them for the checkup. while entering the home the temperature of the person is checked and the person will be sanitized completely.

CONTENT

PROJECT DESCRIPTION

1.1 INTRODUCTION

Home automation is a network of hardware, communication, and electronic interfaces that work to integrate everyday devices with one another via the Internet. Each device has sensors and is connected through Wi-Fi, so you can manage them from your smartphone or tablet whether you're at home, or miles away. This allows you to turn on the lights, lock the front door, or even turn down the heat, no matter where you are. There are three main elements of a home automation system: sensors, controllers, and actuators. Sensors can monitor changes in daylight, temperature, or motion detection. Home automation systems can then adjust those settings (and more) to your preferences. Controllers refer to the devices — personal computers, tablets or smartphones — used to send and receive messages about the status of automated features in your home. Actuators may be light switches, motors, or motorized valves that control the actual mechanism, or function, of a home automation system. They are programmed to be activated by a remote command from a controller.

Home automation refers to the automatic and electronic control of household features, activity, and appliances. In simple terms, it means you can easily control the utilities and features of your home via the Internet to make life more convenient and secure, and even spend less on household bills. The benefits of home automation include Control your home from mobile devices, including your laptop, tablet, or smartphone. Use home automation to make your home a more comfortable, livable space. Preprogram your

thermostat with your preferred settings so that your home is always at a comfortable temperature, set up smart speakers to play music when you get home from work, or adjust your lights to soften or brighten based on the time of day. Program devices to turn on automatically at certain times, or access their settings remotely from anywhere with an Internet connection. When you don't have to remember to lock the door behind you or switch off the lights, you can turn your attention to more important things. Smart fire detectors, carbon monoxide monitors, pressure sensors, and other home automation security features can help protect your home from disaster. Home automation allows you to be more mindful of your power usage. For example, you can save on energy bills by reducing the length of time that lights stay on, or by lowering temperatures when you leave a room. Global Home Automation Market size was valued at $39,607 million in 2016, and is projected to reach at $81,645 million by 2023, growing at a CAGR of 11.2% from 2017 to 2023. The India home automation market size was valued at $1,790.9 million in 2018, and is expected to reach at $13,574.1 million by 2026, growing at a CAGR of 29.8% during the forecast period 2019-2026. In India home automation market is segmented on the basis of product, technology, and end user. By product, the India home automation market is segregated into lighting, security & safety, HVAC, entertainment, and others. The lighting segment is further divided into relays, dimmers, switches, and others. Security & safety is sub-segmented into bells, locks, security cameras, and others. In 2018, the India home automation share is highest among the Security & safety segment, owing to the growth in urbanization. The HVAC segment is further classified into thermostats, sensors, control valves, and others. Entertainment is subdivided into home theater system, audio, volume & multimedia controls, and others. The others segment includes wall panels, ceiling plates, fans, and wires. Depending on technology, the market is bifurcated into wired and wireless. By end use, it is fragmented into residential, commercial, and hospitality.

1.2ABOUT THE PROJECT

Home automation systems are advancement to the mechanization processes where in human efforts are needed with the machinery equipment to operate various loads in homes. Our project aims on providing a low cost efficient and flexible home automation system.

All our home appliances can be controlled using one android device, so that it becomes more convenient for the users to control. Our home automation system not only controls the home loads but it also notifies when the home is in risk. So the user can feel more secured. Control of appliances are done through both our native app and voice command. Our project also demonstrates usage of solar energy which plays a major role in proper usage of renewable energy.

1.3 PROBLEM STATEMENT

The existing system is used only for controlling the home appliances, also they use predecessor micro controller ESP8266 which is less efficient. Home automation not only refers to reduce human efforts but also energy efficiency and time saving. The existing system does not implement any home security. With ever-increasing digital transformation, automation, increasing population and complex lifestyle the world is also facing huge energy crisis due to enhanced carbon emission, climate change like global warming & depletion of the ozone layer and depletion of fossil fuels. So, the importance of renewable and sustainable energy resources is undeniable The energy conservation was also not implemented. Thirdly voice control over appliances was not implemented.

1.4 PROPOSED SYSTEM

The Proposed system focuses on adding additional features to the existing system. It gives us the solution for the drawbacks in the existing system. The missing feature security in the existing system is rectified by implementing security in the proposed system. The security is implemented by using various sensors. The missing feature voice control in the existing system is implemented using Google voice assistant. The missing feature, effective management of energy in existing system is achieved using Solar tracking mechanism

1.5 EXISTING SYSTEM

The existing system aimed on developing a home automation system based on IOT using Wi-Fi based microcontroller. The existing system is concerned, NodeMCU (ESP8266)

microcontroller along with Relays is used to control electrical switches remotely from the server which is built on Node.js. User can control switches using a Web Application after authenticating.

OBJECTIVES

The main objectives of this project is to rectify all the drawbacks in the existing system by implementing the home security, including voice control over appliances and effective usage of renewable energy

The objectives of the project are to design a smart home system which is:

1) Automation
2) Efficient
3) User-friendly
4) Energy saving
5) Cost effective
6) Secured
7) Respond to external environment
8) Intelligent
9) Respond as per user's input

1.6 HARDWARE REQUIREMENT

- PC preferably running windows
- ESP32 micro-controller
- ARDUINO UNO micro-controller
- 4 channel relay
- Servo motor
- Power adapter

- Passive Infrared Sensor
- Light Dependent Resistor
- Infrared sensor
- Gas sensor
- Fire sensor
- Buzzer
- Solar Panel
- LED Light
- Laser Light
- Jumper Wires
- Ribbon Cable
- ON/OFF Switch

1.7 SOFTWARE REQUIREMENT

- Arduino IDE for windows
- C programming
- C++ programming
- Firebase Database

2. SYSTEM DESIGN

2.1 ARCHITECTURE OF ESP32

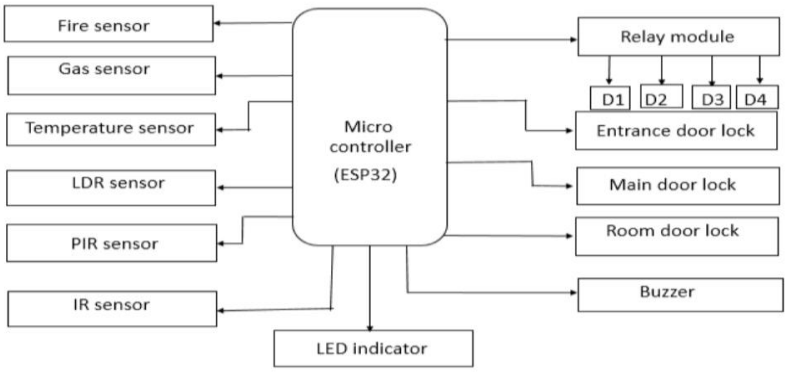


Figure 2.1 Architecture of Esp32

2.2 ARCHITECTURE OF ARDUINO UNO

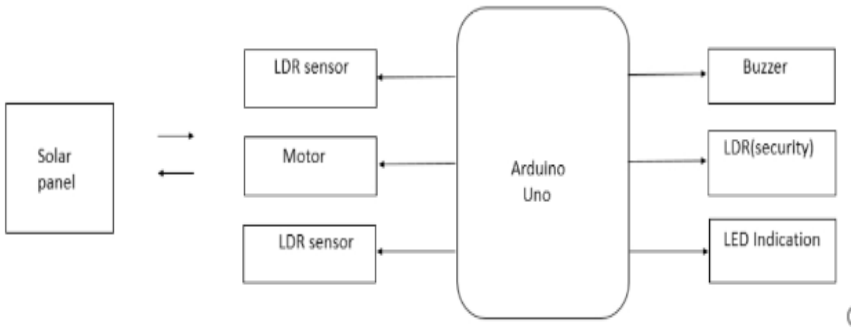


Figure 2.2 Architecture of Arduino uno

2.3 ZERO LEVEL DATA FLOW DIAGRAM

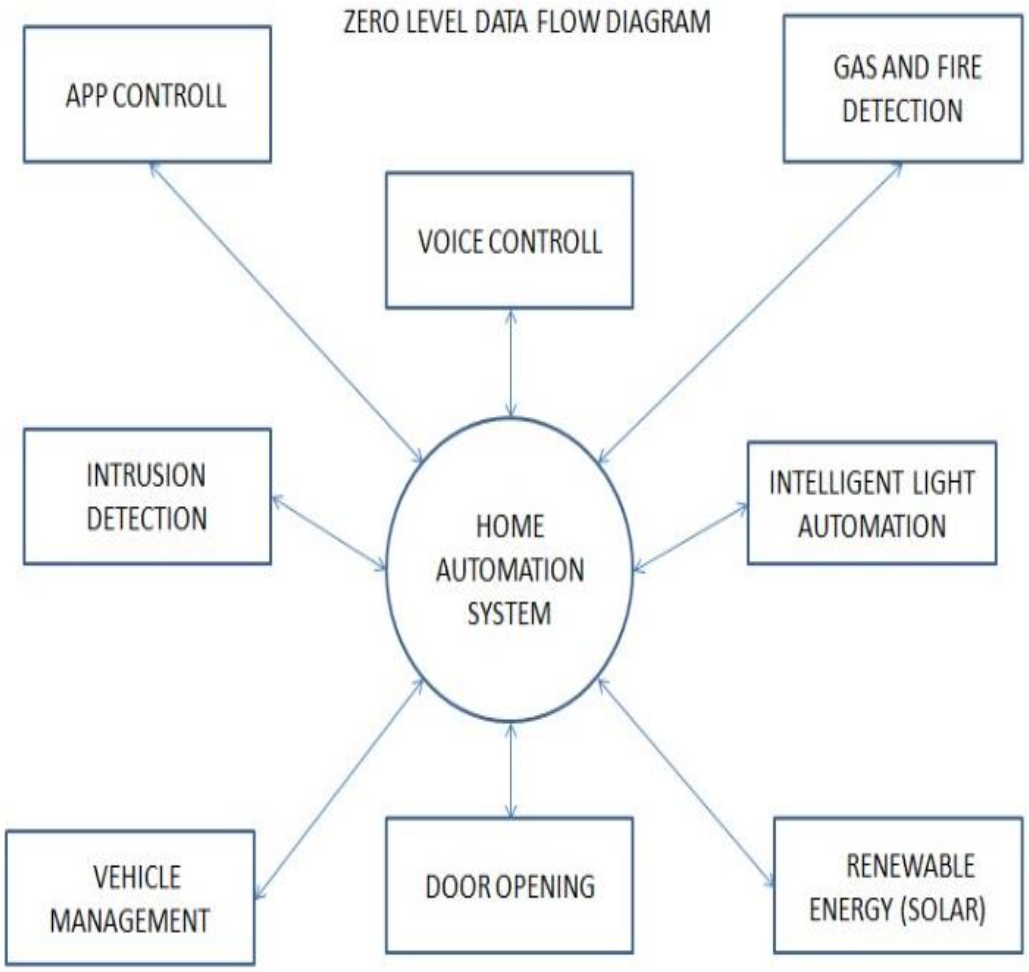


Figure 2.3 Zero level data flow diagram

2.4 CIRCUIT DIAGRAM OF ARDUINO UNO

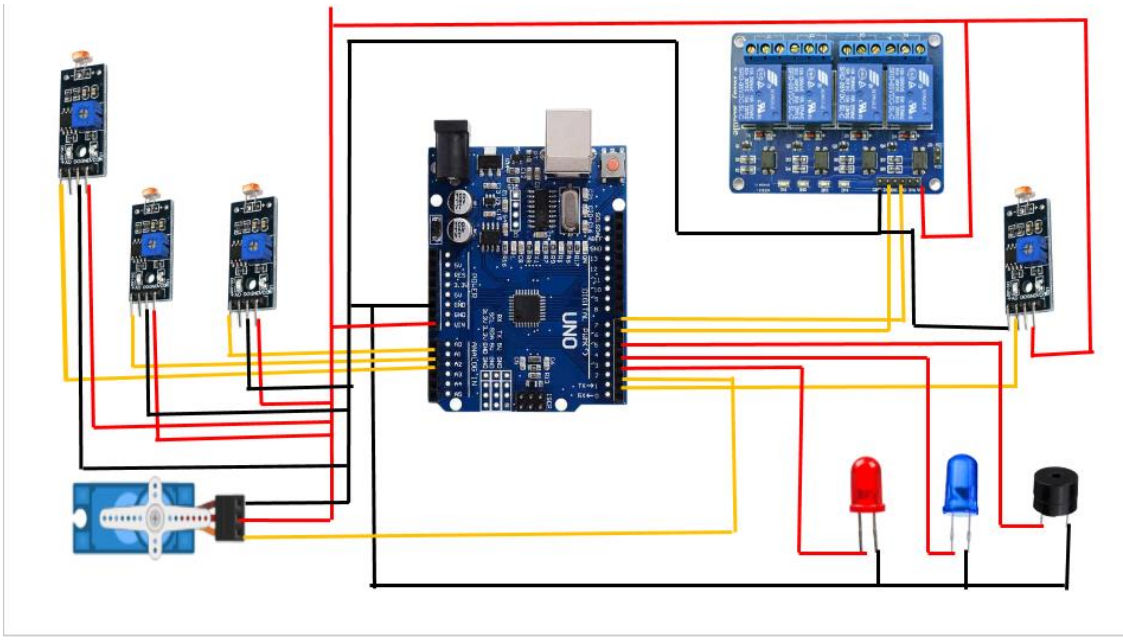


Figure 2.4 Circuit diagram pf Arduino uno

2.5 CIRCUIT DIAGRAM OF ESP32

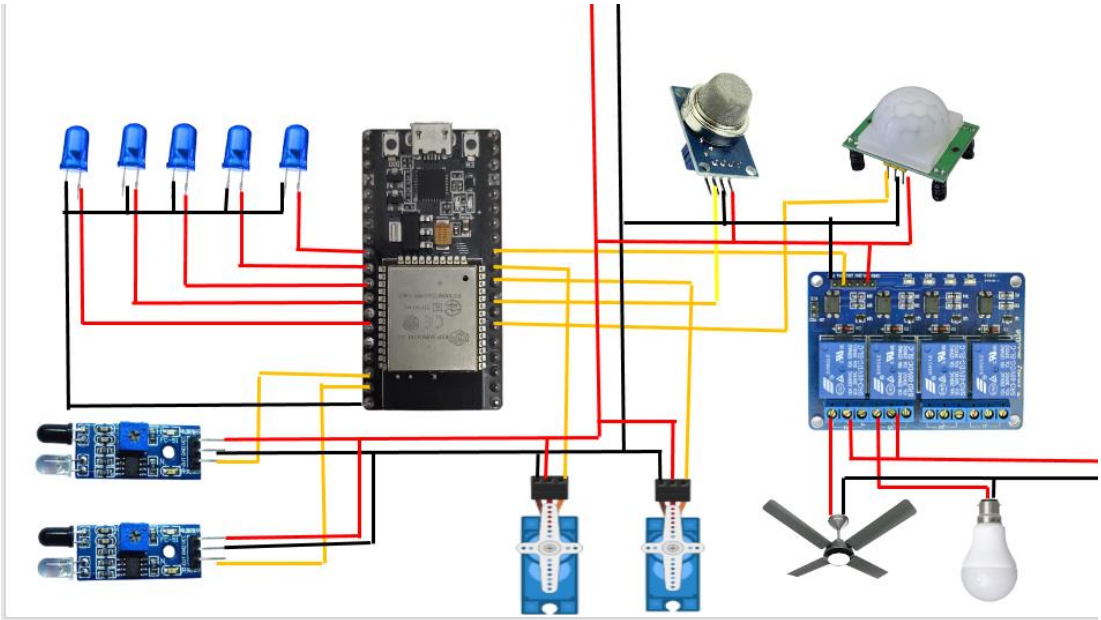


Figure 2.5 Circuit diagram of Esp32

3. PROGRAM DESIGN

3.1 APP CONTROL OVER APPLIANCES

With enhancement in technology, automation has become a need whether it is home, office or some other place. At home we come across many appliances be it fan, AC, TV, lights, etc. What if you could operate all of them with the Android Phone holding in our hand.

In this module we aimed on controlling all our home appliances through our App. Our App takes IP address as input so that it can control the appliances which are already connected through the same WI-FI. Using our app, the appliances can be controlled in a faster and efficient way. The app can be extended to any number of appliances we want to connect. We also added a security feature for

APP CONROL OVER APPLIANCES

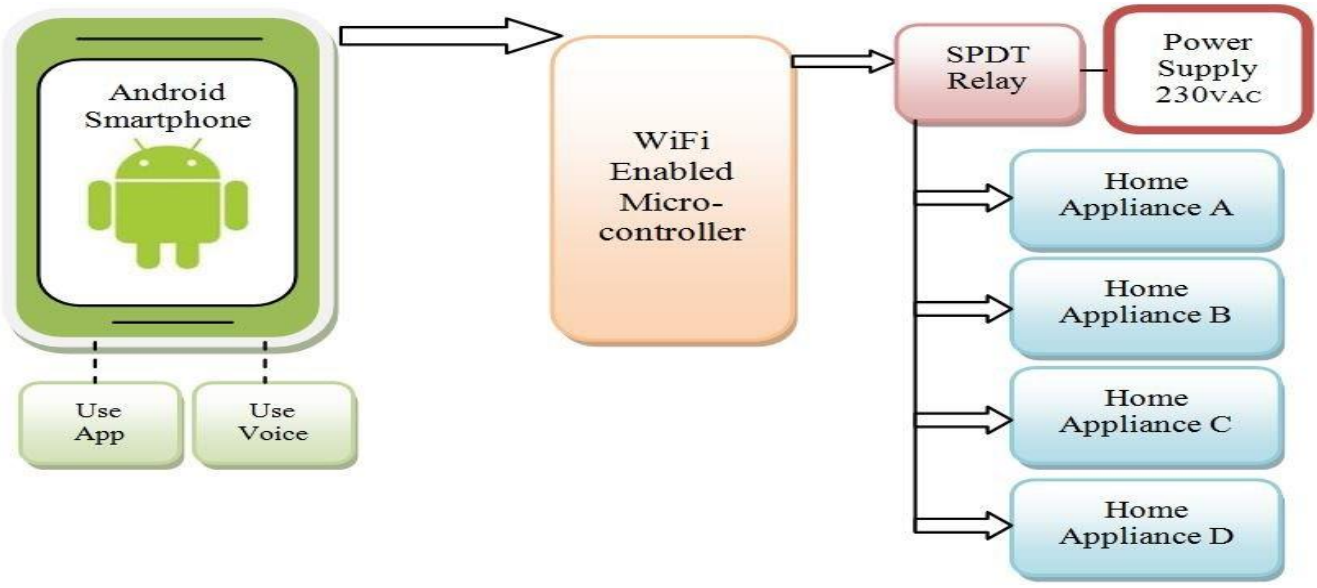


Figure 3.1 App control over appliances

3.2 VOICE CONTROL OVER APPLIANCES

In this module we aimed on controlling all our home appliances through our Voice. For this we are using GOOGLE ASSISTANT. Using IFTTT service this is made Possible. The appliances are connected to ADAFRUIT IO, a cloud service. IFTTT triggers the button in ADAFRUIT IO when the GOOGLE ASSISTANT gets the voice command.

Google assistant is AI (Artificial Intelligence) based voice command service. Using voice, we can interact with google assistant and it can search on the internet, schedule events, set alarms, control appliances, etc. This service is available on smartphones and Google Home devices. We can control smart home devices including lights, switches, fans and thermostats using our Google Assistant.

3.3 CREATING FEEDS ON ADAFRUIT IO

Adafruit IO is a platform designed to display, respond, and interact with our project's data.

- First, created an account at www.Adafruit.io.
- Create a dashboard. This dashboard is a user interface to control things remotely.
- After following the above steps, providing a name to the dashboard and save it.
- Then, creating feed (user interface) to control light On-Off. To create it, just clicking on the **'+'** symbol and selecting the toggle feed.
- After selecting toggle feed, a pop-up window appears Enter the name of our feed create it.
- After creation, select the created feed and then click on the Next step**.** 0(OFF) and 1(ON) text for button and then click on create.
- This will create a toggle button on your dashboard which can be used to control things remotely.
- Now, dashboard is ready for IoT applications like home automation.

3.4 CONFIGURING IFTTT

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. Here, we used IFTTT to use google assistant service and Adafruit service in the chain. So, when I use google assistant to control light of my home by saying Ok Google, turn the light ON or OFF. Then IFTTT interprets the message and can send it to Ad fruit's dashboard as an understandable command to the created feed.

- The first step is creating an account on IFTTT.
- After account creation, clicking on My Applets and then selecting New Applet.
- After selecting a new applet, we get a new page in which we should click on to This
- Then searching for Google Assistant and select it.
- Then, entering voice phrases which we will use as a command for google assistant.
- Then searching for Adafruit and select it.
- After selecting Adafruit, choose action. Now enter what data we need to send to which feed of Adafruit dashboard.
- Clicking on Create Action.

So, when we use Google Assistant on my mobile and give voice command as "Ok Google, Turn LED ON", applet created in IFTTT receives this command and will send data '1' to the Adafruit feed. This will trigger the event on the Adafruit dashboard which is continuously monitored by the microcontroller. This microcontroller will take action as per the data change on the Adafruit dashboard.

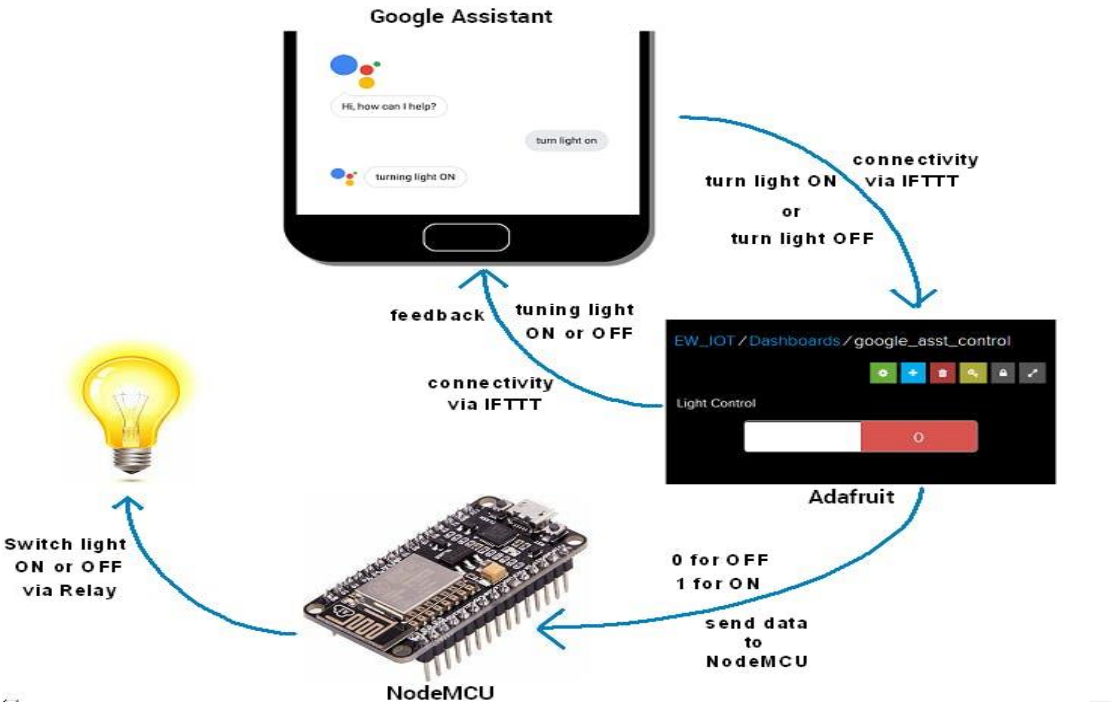


Figure 3.4 voice control over appliances

3.5 SENSOR NOTIFICATION

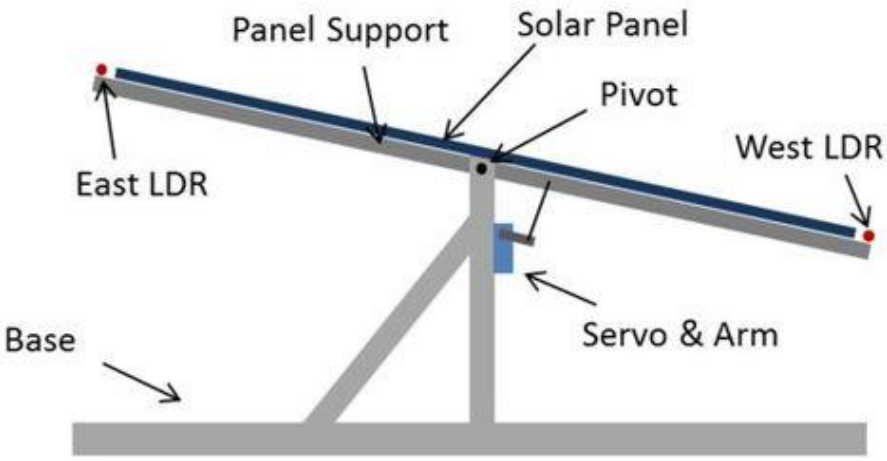
The sensor used in our project includes LDR, IR, PIR, Fire and Gas Sensors. If these sensor senses any intrusion, it notifies us through our native app. To store sensor values we are using firebase database as backend. Using IoT push notifications, we can send notifications to other devices. In this case, we want to send IoT push notifications from a smart object to a mobile smartphone. This project integrates different eco-systems and can have several implementations. For example, we can send a push notification to our smartphone when an alert occurs:

- Fire detection
- Motion detection
- Parking

Figure 3.2 Firebase push notification

3.4 SOLAR TRACKING MECHANISM

Due to increasing energy demand and the decreasing amount of solid fuel, the people are directed to find new energy sources. Thus, the use of solar and wind energy has been increased in recent years and has become widespread. Especially the sun is always seen as a source of energy generating electricity directly from sunlight and an environment friendly energy source that is introduced as an important part of the electricity generation from solar energy today. Electrical energy generation from solar panels varies depending on the angle of irradiation and time of the solar light. The right angle of light delivered to the solar panels and increased duration of illumination provide to increase the quantity of produced energy. Therefore, solar tracking systems are used to increase the amount of electric power generated from solar panels. There have been numerous studies on solar tracking systems in the literature. single-axis solar tracking study, using adrino uno microprocessor and the development control algorithm was performed in. As it was done as a single axis solar tracking system in the east-west direction system in the east-west axis solar tracking system in the east-west direction system in the east-west direction**.**



/

Figure 3.6 Solar tracking mechanism

3.7 INTELLIGENT LIGHT AUTOMATION

In this module we are using LDR sensor. The main idea of this module is to automate the light based on the changing environment. In this module the LDR detects the amount of light energy that it has been receiving or we can also say that it detects the presence of the light that is the reason why it is called as light detecting resistor and when the LDR detects the light it sends a signal to the microcontroller. we have programmed the microcontroller that if the light is detected, the orange color LED will get on. When light is not detected by LDR the blue color LED will gets on. Using this concept at day time orange light is illuminated under the specified time because orange light improves alertness and cognitive brain function. Light, both natural and artificial, affects our circadian rhythms, causing alertness and sometimes upsetting our sleep patterns. At night blue light turns on because blue light messes with your body's ability to prepare for sleep because it blocks a hormone called melatonin that makes you sleepy.

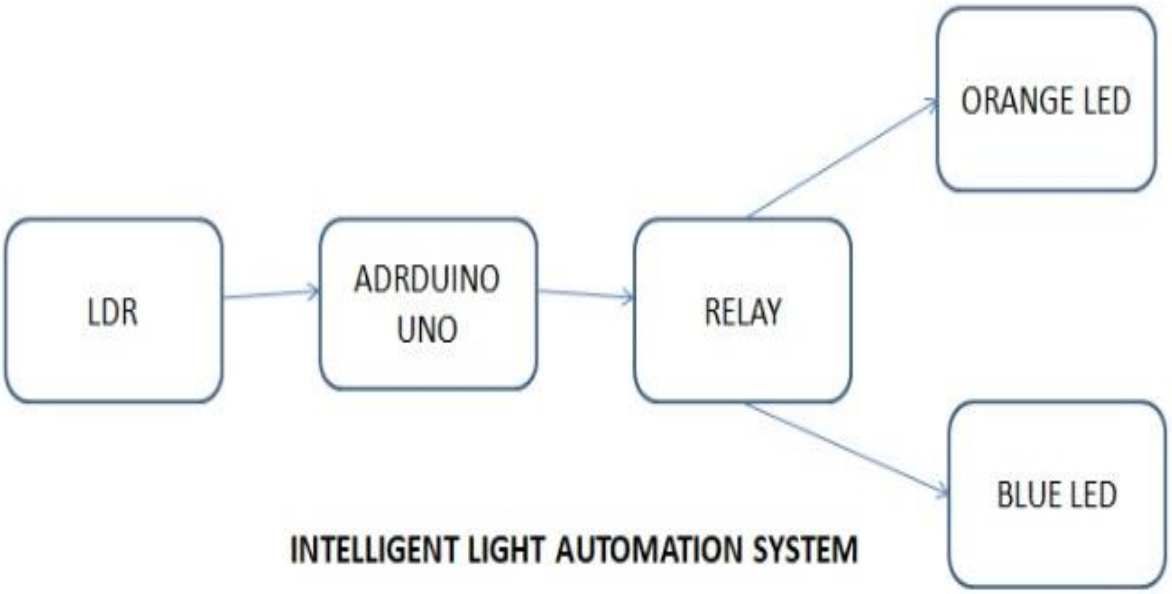


Figure 3.7 Intelligent home automation system

4. HARDWARE AND SOFTWARE DESCRIPTION

4.1 ESP32 MICROCONTROLLER

General Description

ESP32 is a series of low-cost, low-power system on a chip micro-controllers with integrated Wi-Fi and dual-mode Bluetooth. The ESP32 series employs a Tensilica Xtensa LX6 microprocessor in both dual-core and single-core variations and includes built-in antenna switches, RF balun, power amplifier, low-noise receive amplifier, filters, and power-management modules. ESP32 is created and developed by Espressif Systems, a Shanghai-based Chinese company, and is manufactured by TSMC using their 40 nm process. It is a successor to the ESP8266 microcontroller

Product Description

ESP32 has Xtensa® Dual-Core 32-bit LX6 microprocessors, which runs up to 600 DMIPS. The ESP32 will run on breakout boards and modules from 160Mhz up to 240MHz. That is very good speed for anything that requires a microcontroller with connectivity options.

The two cores are named Protocol CPU (PRO_CPU) and Application CPU (APP_CPU). That basically means the PRO_CPU processor handles the Wi-Fi, Bluetooth and other internal peripherals like SPI, I2C, ADC etc. The APP_CPU is left out for the application code. This differentiation is done in the Espressif Internet Development Framework (ESP-IDF). ESP-IDF is the official software development framework for the chip. Arduino and other implementations for the development will be based on ESP-IDF.

ESP-IDF uses free RTOS for switching between the processors and data exchange between them. We have done numerous tutorials on free RTOS and with all the bare-metal programming tutorials for ESP32 we will try and cover this aspect in detail. Although the feature set is great at the price at which the chip is being sold, the complexity is enormous.

For the chip to get widely adopted, it will require huge efforts from Espressif as well as the community.

ESP32 Specification

- Processors:
    - CPU: Xtensa dual-core (or single-core) 32-bit LX6 microprocessor, operating at 160 or 240 MHz and performing at up to 600 DMIPS
    - Ultra low power (ULP) co-processor
- Memory: 520 KB SRAM
- Wireless connectivity:
    - Wi-Fi: 802.11 b/g/n
    - Bluetooth: v4.2 BR/EDR and BLE (shares the radio with Wi-Fi)
- Peripheral interfaces:
    - 12-bit SAR ADC up to 18 channels
    - 2 × 8-bit DACs
    - 10 × touch sensors (capacitive sensing GPIOs)
    - 4 × SPI
    - 2 × I²S interfaces
    - 2 × I²C interfaces
    - 3 × UART
    - SD/SDIO/CE-ATA/MMC/eMMC host controller
    - SDIO/SPI slave controller
    - Ethernet MAC interface with dedicated DMA and IEEE 1588 Precision Time Protocol support
    - CAN bus 2.0
    - Infrared remote controller (TX/RX, up to 8 channels)
    - Motor PWM
    - LED PWM (up to 16 channels)
    - Hall effect sensor

- Ultra low power analog pre-amplifier

- Security:
  - IEEE 802.11 standard security features all supported, including WFA, WPA/WPA2 and WAPI
  - Secure boot
  - Flash encryption

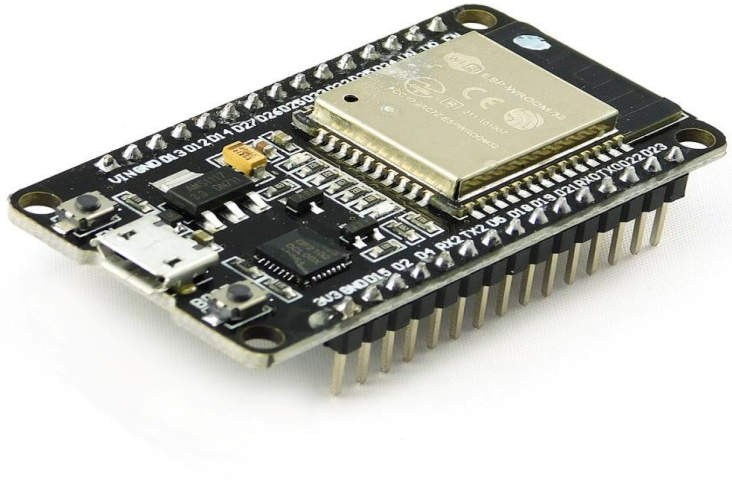


Figure 4.1 Esp32

  - 1024-bit OTP, up to 768-bit for customers
  - Cryptographic hardware acceleration: AES, SHA-2, RSA, elliptic curve cryptography (ECC), random number generator (RNG)

- Power management:
  - Internal low-dropout regulator
  - Individual power domain for RTC
  - 5 μA deep sleep current
  - Wake up from GPIO interrupt, timer, ADC measurements, capacitive touch sensor interrupt

APPLICATIONS

- DIY –Intelligent Robot Arms
- Temperature Measurement Gun
- Weather Monitoring System
- Home Automation System

4.2 ARDUINO UNO

General Description

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc. The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits. The board has 14 digital I/O pins (six capable of PWM output), 6 analog I/O pins, and is programmable with the Arduino IDE (Integrated Development Environment), via a type B USB cable. It can be powered by the USB cable or by an external 9-volt battery, though it accepts voltages between 7 and 20 volts. It is similar to the Arduino Nano and Leonardo. The hardware reference design is distributed under a Creative Commons Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available.

The word "uno" means "one" in Italian and was chosen to mark the initial release of Arduino Software. The Uno board is the first in a series of USB-based Arduino boards; it and version 1.0 of the Arduino IDE were the reference versions of Arduino, which have now evolved to newer releases. The ATmega328 on the board comes preprogrammed with a bootloader that allows uploading new code to it without the use of an external hardware programmer. While the Uno communicates using the original STK500 protocol, differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

Product Description

Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connects to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

ARDUINO UNO SPECEFICATIONS

- Microcontroller: Microchip ATmega328P
- Operating Voltage: 5 Volts
- Input Voltage: 7 to 20 Volts
- Digital I/O Pins: 14 (of which 6 can provide PWM output)
- UART: 1
- I2C: 1
- SPPI: 1
- Analog Input Pins: 6
- DC Current per I/O Pin: 20 mA
- DC Current for 3.3V Pin: 50 mA
- Flash Memory: 32 KB of which 0.5 KB used by bootloader
- SRAM: 2 KB
- EEPROM: 1 KB
- Clock Speed: 16 MHz
- Length: 68.6 mm
- Width: 53.4 mm
- Weight: 25 g

APPLICATIONS

- Arduino Uno is used in Do-it-Yourself projects prototyping.
- In developing projects based on code-based control
- Development of Automation System
- Designing of basic circuit designs.

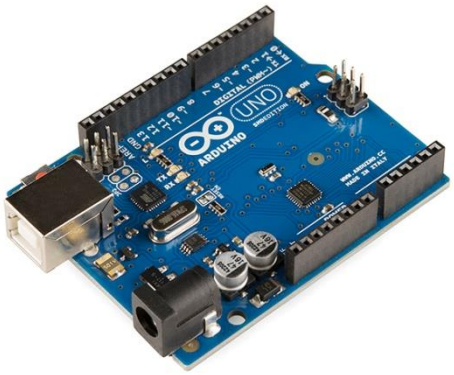


Figure 4.2 UNO

4.3 Four-channel relay module

General Description:

The four-channel can be used to switch multiple loads at the same time since there are four relays on the same module. This is useful in creating a central hub from where multiple remote loads can be powered. It is useful for tasks like home automation where the module can be placed in the main switchboard and can be connected to loads in other parts of the house and can be controlled from a central location using a microcontroller.

Product Description

The four-channel relay module contains four 5V relays and the associated switching and isolating components, which makes interfacing with a microcontroller or sensor easy with minimum components and connections. There are two terminal blocks with six terminals each, and each block is shared by two relays. The terminals are screw type, which makes connections to mains wiring easy and changeable.

The four relays on the module are rated for 5V, which means the relay is activated when there is approximately 5V across the coil. The contacts on each relay are specified for 250VAC and 30VDC and 10A in each case, as marked on the body of the relays. The switching transistors act as a buffer between the relay coils that require high currents, and the inputs which don't draw much current. They amplify the input signal so that they can drive the coils to activate the relays. The freewheeling diodes prevent voltage spikes across the transistors when the relay is turned off since the coils are an inductive load. The indicator LEDs glow when the coil of the respective relay is energized, indicating that the relay is active. The couplers form an additional layer of isolation between the load being switched and the inputs. The isolation is optional and can be selected using the V_{CC} selector jumper. The input jumper contains the main V_{CC}, GND, and input pins for easy connection using female jumper wires.

Relay Specifications

- 5V 4-Channel Relay interface board
- Requires 15-20mA signal drive Current
- TTL logic compatible
- High-current AC250V/10A, DC30V/10A relay
- Status LED
- Equipped with 3.1mm screw holes for easy installation

Applications

- Commonly used in switching circuits.
- For Home Automation projects to switch AC loads
- To Control (On/Off) Heavy loads at a pre-determined time/condition
- Used in safety circuits to disconnect the load from supply in event of failure
- Used in Automobiles electronics for controlling indicators glass motors etc.

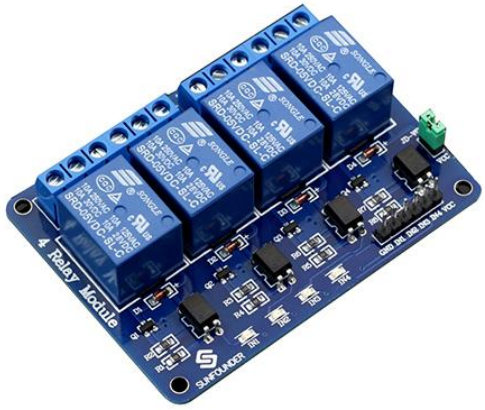


Figure 4.3 Four channel relay

4.5 Servo Motor

General Description

A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration. It consists of a suitable motor coupled to a sensor for position feedback. It also requires a relatively sophisticated controller, often a dedicated module designed specifically for use with servomotors. Servomotors are not a specific class of motor, although the term *servomotor* is often used to refer to a motor suitable for use in a closed-loop control system.

Product Description

Basically **servo motor** is made up of DC **motor** which is controlled by a variable resistor (potentiometer) and some gears. The pulse of 1 ms (1 millisecond) width can rotate the **servo** to 0 degrees, 1.5ms can rotate to 90 degrees (neutral position) and 2 ms pulse can rotate it to 180 degrees. If it's above 5 volts, we need to power them with something other than the Arduino's 5v OUT, and if they draw a lot of current but only use 5V, consider an external power supply for your Arduino board. The current is added whenever we have multiple servos but the voltage required stays the same: for instance, if we have two servos that require 5V voltage but 200mA current, we still only need 5 volts but you need 400mA current.

Servo specifications

- Trigger Voltage (Voltage across coil) : 5V DC
- Trigger Current (Nominal current) : 70mA
- Maximum AC load current: 10A @ 250/125V AC
- Maximum DC load current: 10A @ 30/28V DC
- Compact 5-pin configuration with plastic molding
- Operating time: 10msec Release time: 5msec
- Maximum switching: 300 operating/minute (mechanically)

Application

- Elevator technology
- Robotics
- Cameras, telescopes and antennas
- Industrial production

Figure 4.4 Servo

4.5 Passive Infrared Sensor

General Description

A passive infrared sensor (PIR sensor) is an electronic sensor that measures infrared (IR) light radiating from objects in its field of view. They are most often

used in PIR-based motion detectors. PIR sensors are commonly used in security alarms and automatic lighting applications.PIR sensors detect general movement, but do not give information on who or what moved. For that purpose, an imaging IR sensor is required.PIR sensors are commonly called simply "PIR", or sometimes "PID", for "passive infrared detector". The term passive refers to the fact that PIR devices do not radiate energy for detection purposes. They work entirely by detecting infrared radiation (radiant heat) emitted by or reflected from objects.

Product Description

HC-SR501 PIR Sensor

The Vcc is Input voltage is +5V for typical applications. Can range from 4.5V- 12V. High/Low Output (Dout) is Digital pulse high (3.3V) when triggered (motion detected) digital low(0V) when idle (no motion detected Ground is Connected to ground of circuit.

Specifications

- Wide range on input voltage varying from 4.V to 12V (+5V recommended)
- Output voltage is High/Low (3.3V TTL)
- Can distinguish between object movement and human movement
- Has to operating modes - Repeatable(H) and Non- Repeatable(H)
- Cover distance of about 120° and 7 meters
- Low power consumption of 65mA
- Operating temperature from -20° to +80° Celsius

Application

- All outdoor Lights
- Lift Lobby
- Multi Apartment Complexes
- Common staircases
- For Basement or Covered Parking Area
- Shopping Malls
- For garden lights

Figure 4.5 Passive infrared sensor

4.6 Light Dependent Resistors

General Description

Photo resistors, also known as light dependent resistors (LDR), are light sensitive devices most often used to indicate the presence or absence of light, or to measure the light intensity. In the dark, their resistance is very high, sometimes up to 1MΩ, but when the LDR sensor is exposed to light, the resistance drops dramatically, even down to a few ohms depending on the light intensity. LDRs have a sensitivity that varies with the wavelength of the light applied and are nonlinear devices. They are used in many applications but are sometimes made obsolete by other devices such as photodiodes and phototransistors. Some countries have banned LDRs made of lead or cadmium over environmental safety concerns.

Product Description

An LDR or photo resistor is made any semiconductor material with a high resistance. It has a high resistance because there are very few electrons that are free and able to move - the vast majority of the electrons are locked into the crystal lattice and

unable to move. Therefore, in this state there is a high LDR resistance. As light falls on the semiconductor, the light photons are absorbed by the semiconductor lattice and some of their energy is transferred to the electrons. This gives some of them sufficient energy to break free from the crystal lattice so that they can then conduct electricity. This results in a lowering of the resistance of the semiconductor and hence the overall LDR resistance. The process is progressive, and as more light shines on the LDR semiconductor, so more electrons are released to conduct electricity and the resistance falls further. The active semiconductor region is normally deposited onto a semi-insulating substrate and the active region is normally lightly doped. In many discrete photo resistor devices, an interdigital pattern is used to increase the area of the photo resistor that is exposed to light. The pattern is cut in the metallization on the surface of the active area and this lets the light through. The two metallize areas act as the two contacts for the resistor. This area has to be made relatively large because the resistance of the contact to the active area needs to be minimized.

Specifications

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

Applications

- Automatic Street Light
- Detect Day or Night
- Automatic Head Light Dimmer
- Position sensor
- Used along with LED as obstacle detector
- Automatic bedroom Lights
- Automatic Rear view mirror

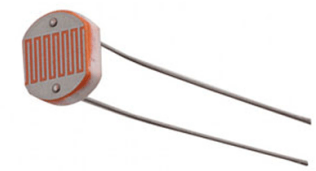


Figure 4.6 Light Dependent Resistor

4.7 Infrared Sensor

General Description

The emitter is simply an IR LED and the detector is simply an IR photodiode. Photodiode is sensitive to IR light of the same wavelength which is emitted by the IR LED. When IR light falls on the photodiode, the resistances and the output voltages will change in proportion to the magnitude of the IR light received. There are five basic elements used in a typical infrared detection system: an infrared source, a transmission medium, optical component, infrared detectors or receivers and signal processing. Infrared lasers and Infrared LED's of specific wavelength used as infrared sources. The three main types of media used for infrared transmission are vacuum, atmosphere and optical fibers. Optical components are used to focus the infrared radiation or to limit the spectral response.

Product Description

These simple devices operate at 940nm and work well for generic IR systems including remote control and touch-less object sensing. Using a simple ADC on any microcontroller will allow variable readings to be collected from the detector. The emitter is driven up to 50mA with a current limiting resistor as with any LED device. The detect is a NPN transistor that is biased by incoming IR light. An infrared sensor is an electronic device that emits and/or detects infrared radiation in order to sense some aspect of its surroundings. Infrared sensors can measure the heat of an object, as well as detect motion. Many of these types of sensors only measure infrared radiation, rather than emitting it.

Specifications

- Operating voltage: 4.5 V to 5.5 V
- Average current consumption: 30 mA (typical)
- Output voltage differential over distance range: 1.9 V (typical)
- Response time: 38 ± 10 ms
- Weight: 3.5 g (0.12 oz)
- Range detection (10-80cm).
- No external circuitry required.
- Infrared Proximity Sensor.
- Analog output inversely proportional to distance.

Radiation Thermometers

- Flame Monitors
- Moisture Analyzers
- IR Imaging Devices

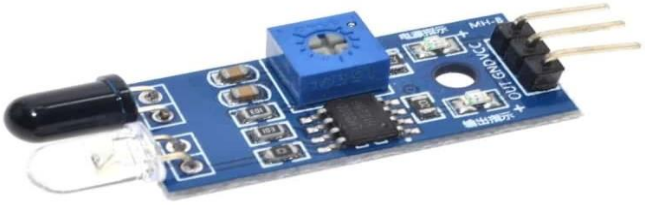


Figure 4.7 Infrared

4.8 Gas Sensor

General Description

A gas sensor is a device which detects the presence or concentration of gases in the atmosphere. Based on the concentration of the gas the sensor produces a corresponding potential difference by changing the resistance of the material inside the sensor, which can be measured as output voltage. Based on this voltage value the type and concentration of the gas can be estimated. The type of gas the sensor could detect depends on the sensing material present inside the sensor. Normally these sensors are available as modules with comparators as shown above. These comparators can be set for a particular threshold value of gas concentration. When the concentration of the gas exceeds this threshold the digital pin goes high. The analog pin can be used to measure the concentration of the gas.

Product Description

Here is an ultra-sensitive LPG sensor that generates loud beeps when it senses any gas leakage. It detects vapors of liquefied petroleum gas anywhere between 200 and 10,000 ppm and drives a piezo buzzer to catch attention for immediate action. The buzzer beeps until the concentration of gas in the air decreases to a safe level. The circuit uses an MQ6 gas sensor, which is designed to sense LPG, propane and isobutene gases. LPG gas is supplied in pressurized steel cylinders. As this gas is heavier than air, when it leaks from a cylinder it flows along floor and tends to settle in low spots such as a basement. This can cause fire or suffocation if not dealt with. Here is a circuit that detects the leakage of LPG gas and alerts the user through audio-visual indications. Fig. 1 shows the circuit of the gas leakage alarm. As per its datasheet, it has high sensitivity to propane, butane, isobutene, LPG and natural gas. The sensor can also be used to detect combustible gases, especially methane. This circuit has been tested with LPG gas and was found to work satisfactorily. Whenever there is LPG concentration of 1000 ppm (parts per million) in the area, the OUT pin of the sensor module goes high. This signal drives IC2, Hex Invert Buffer. cur Specifications Ent-limiting resistors R5 and R4, respectively. LED D1 glows and the alarm sounds to alert the user of gas leakage.

Procedure

- High Sensitivity to LPG, iso-butane, propane.
- Small sensitivity to alcohol, smoke.
- Detection Range: 100 - 10,000 ppm iso-butane propane.
- Fast Response Time: <10s.
- Simple drive circuit.
- Heater Voltage: 5.0V.

Applications

- Used in industries to monitor the concentration of the toxic gases.
- Used in households to detect an emergency incident.
- Used at hotels to avoid customers from smoking.
- Used in air quality check at offices.
- Used in air conditioners to monitor the CO2 levels.

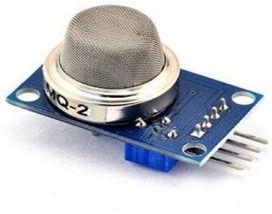


Figure 4.8 Gas Sensor

4.9 Fire Sensor

General Description

A flame detector is a sensor designed to detect and respond to the presence of a flame or fire, allowing flame detection. Responses to a detected flame depend on the installation, but can include sounding an alarm, deactivating a fuel line (such as a propane or a natural gas line), and activating a fire suppression system. When used in applications such as industrial furnaces, their role is to provide confirmation that the furnace is working properly; it can be used to turn off the ignition system though in many cases they take no direct action beyond notifying the operator or control system. A flame detector can often respond faster and more accurately than a smoke or heat detector due to the mechanisms it uses to detect the flame.

Product Description

Flame sensor is the most sensitive to ordinary light that is why its reaction is generally used as flame alarm purposes. This module can detect flame or wavelength in 760 nm to 1100 nm range of light source. Small plate output interface can and single chip can be directly connected to the microcomputer IO port. The sensor and flame should keep a certain distance to avoid high temperature damage to the sensor. The shortest test distance is 80 cm, if the flame is bigger, test it with farther distance. The detection angle is 60 degrees so the flame spectrum is especially sensitive. The detection angle is 60 degrees so the flame spectrum is especially sensitive

Specifications

- On-board LM393 voltage comparator chip and infrared sensing probe
- Support 5V/3.3V voltage input
- On-board signal output indication, output effective signal is high level, and the same time the indicator lights up, output signal can directly connectwith microcontrolle

- Signal detection sensitivity can be adjusted
- Reserved a line voltage compare circuit (P3 is leaded out).
- PCB size: 30(mm) x15(mm)

Application

- Hydrogen stations.
- Gas-fueled cookers
- Industrial heating and drying systems
- Domestic heating systems
- Industrial gas turbines

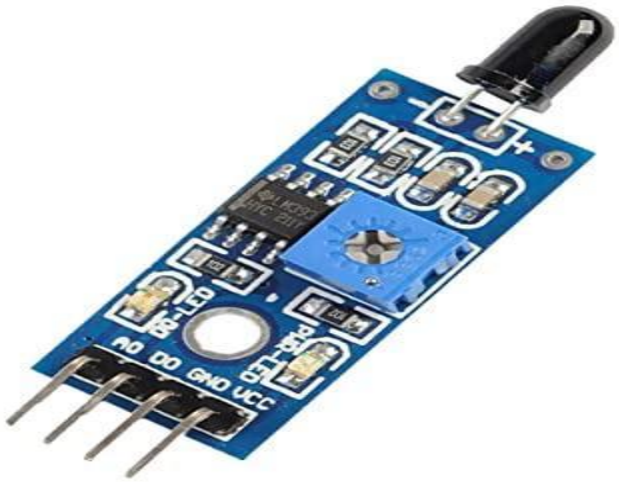


Figure 4.9 Fire Sensor

4.10 BUZZER

Domestic Description

A buzzer or beeper is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric (piezo for short). Typical uses of buzzers and beepers

include alarm devices, timers, and confirmation of user input such as a mouse click or keystroke.

Product Description

A buzzer is a small yet efficient component to add sound features to our project/system. It is very small and compact 2-pin structure hence can be easily used on breadboard, Perf Board and even on PCBs which makes this a widely used component in most electronic applications.

There are two types are buzzers that are commonly available. The one shown here is a simple buzzer which when powered will make a Continuous Beeeeeeppp.... sound, the other type is called a readymade buzzer which will look bulkier than this and will produce a Beep. Beep. Beep. Sound due to the internal oscillating circuit present inside it. But, the one shown here is most widely used because it can be customized with help of other circuits to fit easily in our application.

This buzzer can be used by simply powering it using a DC power supply ranging from 4V to 9V. A simple 9V battery can also be used, but it is recommended to use a regulated +5V or +6V DC supply. The buzzer is normally associated with a switching circuit to turn ON or turn OFF the buzzer at required time and require interval.

Specifications

- Rated Voltage: 6V DC
- Operating Voltage: 4-8V DC
- Rated current: <30mA
- Sound Type: Continuous Beep
- Resonant Frequency: ~2300 Hz
- Small and neat sealed package
- Breadboard and Perf board friendly

Applications

- Alarming Circuits, where the user has to be alarmed about something
- Communication equipments
- Automobile electronics
- Portable equipments, due to its compact size

Figure 4.10 Buzzer

4.11 SOLAR PANELS

General Description

Solar panels are those devices which are used to absorb the sun's rays and convert them into electricity or heat. A solar panel is actually a collection of solar (or photovoltaic) cells, which can be used to generate electricity through photovoltaic effect. These cells are arranged in a grid-like pattern on the surface of solar panels. Thus, it may also be described as a set of photovoltaic modules, mounted on a structure supporting it. A photovoltaic (PV) module is a packaged and connected assembly of 6×10solarcells.When it comes to wear-and-tear, these panels are very hardy. Solar panels wear out extremely slow. In a year, their effectiveness decreases only about one to two per cent. Most solar panels are made up using crystalline silicon solar cells. Installation of solar panels in homes helps in combating the

harmful emissions of greenhouse gases and thus helps reduce global warming. Solar panels do not lead to any form of pollution and are clean. They also decrease our reliance on fossil fuels (which are limited) and traditional power sources. These days, solar panels are used in wide-ranging electronic equipments like calculators, which work as long as sunlight is available.

However, the only major drawback of solar panels is that they are quite costly. Also, solar panels are installed outdoors as they need sunlight to get charged.

Product Description

Solar Panel are made up of many solar cells. Solar cells are made of silicon, like semiconductors. They are constructed with a positive layer and a negative layer, which together create an electric field, just like in a battery. Photovoltaic modules can produce electricity from a range of frequencies of light, but usually cannot cover the entire solar radiation range (specifically, ultraviolet, infrared and low or diffused light). Hence, much of the incident sunlight energy is wasted by solar modules, and they can give far higher efficiencies if illuminated with monochromatic light. Therefore, another design concept is to split the light into six to eight different wavelength ranges that will produce a different color of light, and direct the beams onto different cells tuned to those ranges. This has been projected to be capable of raising efficiency by 50%.

Specifications

- Open Circuit Voltage: 7.7V.
- Peak Voltage: 6.5V.
- Peak Current: 340mA.
- Peak Power: 2.2W.
- Power Tolerance: +/-10%
- Output Cable Length: 26cm.
- Output Cable Color: Red PU Coated.
- Output Cable Plug: Waterproof Male 3.5x1.1mm.

Applications

- Photovoltaic power stations
- Rooftop solar PV systems
- Standalone PV systems
- Solar hybrid power systems

Figure 4.11 Solar panel

4.12 LED Light

General Description

LED stands for light emitting diode. LED lighting products produce light up to 90% more efficiently than incandescent light bulbs. How do they work? An electrical current passes through a microchip, which illuminates the tiny light sources we call LEDs and the result is visible light. To prevent performance issues, the heat LEDs produce is absorbed into a heat sink. The useful life of LED lighting products is defined differently than that of other light sources, such as incandescent or compact fluorescent lighting (CFL). LEDs typically do not “burn out” or fail. Instead, they experience ‘lumen depreciation’, wherein the brightness of the LED dims slowly over time. Unlike incandescent bulbs, LED “lifetime” is established on a prediction of when the light output decreases by 30 percent.

Product Description

The L-53SGD-5V is a 5mm super bright Solid State LED with diffused lens. The super bright source color device is made with gallium phosphate light emitting diode. LEDs come to full brightness immediately with no warm-up delay. Frequent switching on and off does not reduce life expectancy as with fluorescent lighting.] Light output decreases gradually over the lifetime of the LED. The light output in lumens, the power consumption in watts, the color temperature in Kelvin or a color description such as "warm white", "cool white" or "daylight", the operating temperature range, and sometimes the equivalent wattage of an incandescent lamp delivering the same output in lumens.

Specifications

- Superior weather resistance.
- 5mm Round Standard Directivity.
- UV Resistant Epoxy.
- Forward Current (IF): 30mA.
- Forward Voltage (VF): 1.8V to 2.4V.
- Reverse Voltage: 5V.
- Operating Temperature: -30℃ to +85℃
- Storage Temperature: -40℃ to +100℃

Applications

- General purpose lighting luminaires (including LED)
- Electric and electric hybrid vehicles
- Appliances
- Electrical devices

Figure 4.12 Led

4.13 Wires

Wires are used to make frame for the project. It is a type of carbon steel with a low amount of carbon – it is actually also known as “low carbon steel.” It can be used for a variety of purposes in multiple applications. Alongside the uses for low carbon steel as previously mentioned, mild steel can be used for anything from the following: Machinery parts – depending on the ratio of the carbon and iron, the strength, ductility and malleability of steel differs.

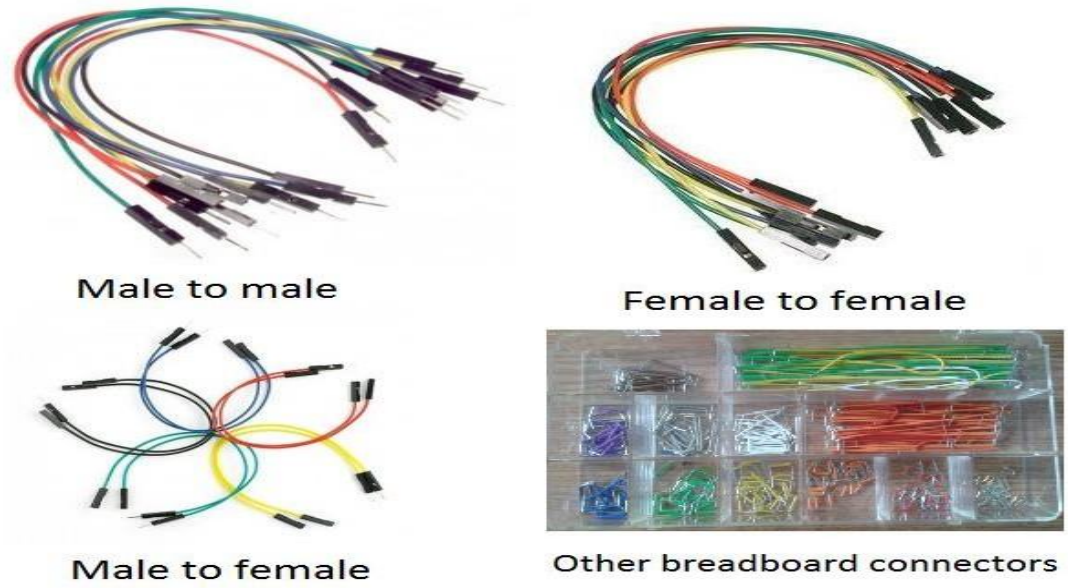


Figure 4.13 Wire

4.14 ON/ OFF SWITCH

A switch is an electrical component that can disconnect or connect the conducting path in an electrical circuit, interrupting the electric current or diverting it from one conductor to another. The most common type of switch is an electromechanical device consisting of one or more sets of movable electrical contacts connected to external circuits. When a pair of contacts is touching current can pass between them, while when the contacts are separated no current can flow.

Switches are made in many different configurations; they may have multiple sets of contacts controlled by the same knob or actuator, and the contacts may operate

simultaneously, sequentially, or alternately. A switch may be operated manually, for example, a light switch or a keyboard button, or may function as a sensing element to sense the position of a machine part, liquid level, pressure, or temperature, such as a thermostat. Many specialized forms exist, such as the toggle switch, rotary switch, mercury switch, push-button switch, reversing switch, relay, and circuit breaker. A common use is control of lighting, where multiple switches may be wired into one circuit to allow convenient control of light fixtures. Switches in high-powered circuits must have special construction to prevent destructive arcing when they are opened.

Figure 4.14 Switch

Ribbon Cable

A ribbon cable (also known as multi-wire planar cable) is a cable with many conducting wires running parallel to each other on the same flat plane. As a result, the cable is wide and flat. Its name comes from its resemblance to a piece of ribbon. Ribbon cables are usually seen for internal peripherals in computers, such as hard drives, CD drives and floppy drives. On some older computer systems (such as the BBC Micro and Apple II series) they were used for external connections as well. The ribbon-

like shape interferes with computer cooling by disrupting airflow within the case and also makes the cables awkward to handle, especially when there are a lot of them; as a result, round cables have almost entirely replaced ribbon cables for external connections and are increasingly being used internally as well.

Arduino IDE

The Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in functions from C and C++. It is used to write and upload programs to Arduino compatible boards, but also, with the help of third-party cores, other vendor development boards. The source code for the IDE is released under the GNU General Public License, version 2. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides

many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub *main()* into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program *avrdude* to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware. By default, avrdude is used as the uploading tool to flash the user code onto official Arduino boards.

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuino hardware to upload programs and communicate with them. Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension. ino. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom right-hand corner of the window displays the configured board and serial port. The toolbar buttons allow us to verify and upload programs, create, open, and save sketches, and open the serial monitor.

LITERATURE SURVEY

International Journal of Computer Applications (0975 – 8887) Volume 116 – No. 11, April 2015 Home Automation Systems - A Study

With the increase in consumption of energy and population, there is a grave need to conserve energy in every way possible. The inability to access and control the appliances from remote locations is one of the major reasons for energy loss. A web or an android application is used by the users to give instructions to these systems. This system can make use of a host of communication methods such as Wi-Fi, GSM, Bluetooth, ZigBee. Different controlling devices and configurations can be found in existing systems. Such systems have been found already in many places for a wide variety of applications. This paper presents a survey of all such systems.

- Appliance can be remotely accessed and controlled using software as an interface, which an Android application and a Web application. Such remotely accessible systems are already available in the market

DISADVANTAGE

- Cost is high

Balakrishna Gokaraju1 , Donald Yessick1 , Jonathan Steel1 , Daniel A. Doss1 and Anish C. Turlapaty2 Integration of Intrusion Detection and Web Service Alarm for Home Automation System Using 'ARM' Microprocessor

This intrusion detection system will be integrated wirelessly to the home Wi-FI system and could initiate an email to the respective authority. There are innumerous commercial home intrusion detection systems readily available in the market, all of which tend to be expensive to install and include contracts with monthly recurring payments. Moreover, these systems have high false alarm rates and unnecessary calls to 911 operators.

The novelty of our present implementation design lies in cost and time effective communication of the intrusion event wirelessly to the homeowners and law-enforcement with a confirmed image of the scene during the intrusion event. The above implementation of the home security system is aimed to compared in terms of design and program implementation against both ARM microprocessor and AT mega micro-controller.

- Works faster and less energy consumption

DISADVANTAGE

- Programming to the micro-controller is so complex

2017Third International Conference on Science Technology Engineering & Management (ICONSTEM) Voice Controlled Home Automation System Using Natural Language Processing (NLP) and Internet of Things (IOT)

The primary objective of this project is to construct a fully functional voice based Home automation system that uses Internet of Things, Artificial Intelligence and Natural Language Processing (NLP) to provide a cost-effective, efficient way to work together with home appliances. There are many smart home solutions in the market that aim to automate the basic operations of these home appliances using various technologies such as GSM (Global System for Mobile), NFC (Near-Field Communication) etc. However, most of these systems focus on mimicking the basic operation of the electrical switch. Our project aims at providing a fully automated voice based solution that our users can rely on, to perform more than just switching on/off the appliances. The user sends a command through speech to the mobile device, which interprets the message and sends the appropriate command to the specific appliance. We plan on implementing four basic home appliances as a “Proof-of-Concept” for this project which includes Fan, Light, Coffee Machine and Door Alarms. The voice command given by the user is interpreted by the mobile device using Natural Language processing. The mobile device acts as a central console; it determines what operation must be completed by which appliance to fulfill the user's

request. The central console might likewise be either a desktop application, web application or a smart phone application as nearly all of the data transferred can be processed by the cloud. However, for the convenience of the user and increased mobile capabilities we will be using a smart phone in this project. The appliances are associated with the mobile device through an Arduino Board that establishes the concept of Internet of Things. The Arduino Boards are interfaced with the appliances and programmed in a manner that they respond to mobile inputs.

- Voice control over appliances

DISADVANTAGE

- Response is slow

ASADULLAH, M., & ULLAH, K. (2017). Smart Home Automation System Using Bluetooth Technology 2017 International Conference on Innovations in Electrical Engineering and Computational Technologies (ICIEECT) Smart Home Automation System Using Bluetooth Technology

In this paper a low cost and user friendly remote controlled home automation system is presented using Arduino board, Bluetooth module, smartphone, ultrasonic sensor and moisture sensor. A smartphone application is used in the suggested system which allows the users to control up to 18 devices including home appliances and sensors using Bluetooth technology. Nowadays, most of conventional home automation systems are designed for special purposes while proposed system is a general purpose home automation system. Which can easily be implement in existing home. The suggested system has more features than conventional home automation systems such as an ultrasonic sensor is used for water level detection and soil moisture sensor is use for automatic plant irrigation system. This paper also describes the hardware and software architecture of system, future work and scope. The proposed prototype of home automation system is implemented and tested on hardware and it gave the exact and expected results.

- Control up to 18 devices including home appliances and sensors using Bluetooth technology.

DISADVANTAGE

- Operates only in shorter range.

Arathi, P. N., Arthika, S., Ponmithra, S., Srinivasan, K., & Rukkumani, V. (2017). Gesture based home automation system. 2017 International Conference on Nextgen Electronic Technologies: Silicon to Software (ICNETS2).

Smart can be a potential application which provides support to elderly or disabled persons. Home automation is the use and control of home appliances remotely or automatically. Day by day the gap between machines and humans is being reduced. Nowadays hand gesture based home automation is getting more importance. Gesture recognition refers to recognizing the motion of the human parts like hand, face, etc. Most of the electronic components manufactures focuses on the hand gesture basis. In this proposed work MATLAB based algorithm is used. This proposed work uses this tool for the recognition and processing of the hand gesture. The algorithm used for this proposed work is the object detection algorithm. At first the image is captured by the camera and it is processed by the MATLAB, if the preloaded gesture is matched with the existing gesture the data will be sent to the microcontroller, then the home appliances are controlled. There are other applications which could be controlled by a gesture such as media players, robots and virtual objects. The hardware module consists of camera, PIC microcontroller, fan, light, power supply, LED, GSM module. This hardware module is communicated with simulation software using a USB to serial converter bus which comes along with driver software.

- Very accurate than hand glove based gesture recognition process

SYSTEM SPECIFICATION

LIMITATIONS OF EXISTING SYSTEM

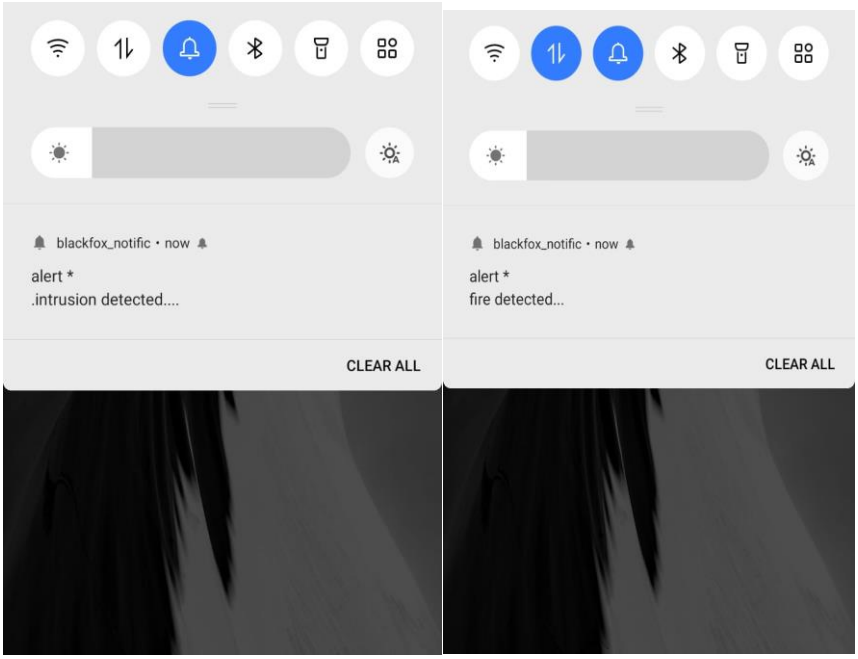
The existing system aimed only on controlling the home appliances. The existing systems does not implement any security. It does only control through app, voice command is not included. The effective utilization of energy is also not included. The existing system uses a low cost microcontroller but it is not so efficient.

BENEFITS OF PROPOSED SYSTEM

The Proposed system gives the access to control devices in our home from a mobile device anywhere in the world. From a home security perspective, various sensors are used so that if any intrusion occurs it notifies us through our native app. As we use solar tracking mechanism and intelligent light automation, more amount of energy is saved and the available energy is efficiently utilized. Voice control feature are very useful for the old age people and physically impaired persons. By implementing all these features we can have built a smart home which is highly secured, user-friendly, efficient, which eases the human effort.

5. TESTING

The results obtained from the procedure described above are illustrated in below. The front door and the main door can be controlled only by finger print authentication for security purpose. The home appliances can be controlled by native app, it can also can be controlled using voice command. The sensors attached will notify us when our home is under risky situation. The PIR sensor attached will notify us when any unwanted intrusion occurs. The fire and gas sensor notify us when any leakage occurs. The solar tracking system tracks the sun's direction so that effective utilization of energy is done. The intelligent light system automatically changes the color of light based on the changing environment. On the specified day time orange led turns on, at night blue light turns on.



6. CONCLUSION

6.1 CONCLUSION

The IoT device market has undergone radical changes in only a few short years. Starting with disparate devices and no ecosystems to speak of, the market has now grown to encompass enterprise players working together to create ecosystems, tailored for mobile technology, which allows IoT devices to become interconnected. Automaton of the home may have once seemed like a peculiar and unlikely concept, but as our devices become smarter and more investment is poured into the development of IOT consumer products, we are likely to see increased competition spur on further innovation in the field.

6.2 FUTURE WORK

IoT is an umbrella term used for all technologies that enable the connection of a device to the Internet. Such systems depend on the collection of data. The data is then used for monitoring, controlling and transferring information to other devices via the internet. This allows specific actions to be automatically activated whenever certain situations arise. It might also send a notification to the user on the same. Now applying the same concept to the entire home and all the devices present. That is a smart home powered by IoT. Instead of manually going up to the device and taking action, those actions can be taken at the press of a button. These days, most smart IoT home automation devices allow you to control them via an app or even via voice commands. Now imagine if we did not even need to undertake such actions. In other words, the smart home will know when to take certain actions and automatically take them. This is where the future of home automation and IoT lies. In the future work the improvement in the security is considered. Various other sensors can be included. Implementing also in other areas like industries, banks etc.

7. REFERENCES

[1] R. S. Ransing and M. Rajput, "Smart home for elderly care, based on Wireless Sensor Network," 2015 International Conference on Nascent Technologies in the Engineering Field (ICNTE), Navi Mumbai, 2015, pp. 1-5.

[2] S. Kumar and S. R. Lee, "Android based smart home system with control via Bluetooth and internet connectivity," The 18th IEEE International Symposium on Consumer Electronics (ISCE 2014), JeJu Island, 2014, pp. 1-2.

[3] Young-Guk Ha, “Dynamic integration of Zigbee home networks into home gateways using OSGI service registry,” IEEE Transactions on Consumer Electronics., vol. 55, Issue. 2, pp.470-476, May. 2009.

[4] J. Stankovic, Q. Cao, T. Doan, L. Fang, Z. He, R. Kiran, “Wireless Sensor Networks for In-Home Healthcare: Potential and Challenges,” Proc. of the Workshop on High Confidence Medical Devices Software and Systems, June 2005.

[5] H. Noguchi, T. Mori and T. Sato, "Human daily behavior accumulation in real home environment via distributed sensors in a long term," 2011 IEEE/SICE International Symposium on System Integration (SII), Kyoto, 2011, pp. 368-374.

[6] S. M. Brundha, P. Lakshmi and S. Santhanalakshmi, "Home automation in client-server approach with user notification along with efficient security alerting system," 2017 International Conference on Smart Technologies for Smart Nation (SmartTechCon), Bangalore, 2017, pp. 596-601.

8. APPENDIX

8.1 Source Code for ESP32

```
#include <WiFi.h>

#include <FirebaseESP32.h>

WiFiClient client;

#include <ESP32Servo.h>

WiFiServer server(80);

#include "Adafruit_MQTT.h"

#include "Adafruit_MQTT_Client.h"

#define AIO_SERVER      "io.adafruit.com"

 #define AIO_SERVERPORT  1883

#define AIO_USERNAME    "ASG"

#define AIO_KEY         "aio_jJCo50Q9I2ZZhfrXfXFuPocSmfj2"

Adafruit_MQTT_Client mqtt(&client, AIO_SERVER, AIO_SERVERPORT, AIO_USERNAME, AIO_KEY);

Adafruit_MQTT_Subscribe relay = Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME "/feeds/relay connectWiFi 1");

Adafruit_MQTT_Subscribe relaym = Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME "/feeds/relay2");
```

```
void MQTT_connect();

void ();

String checkClient();

FirebaseData firebaseData;

/* WIFI settings */

const char* ssid = "POCO M2 Pro";   //WIFI SSID

const char* password = "gopikanth1";    //WIFI PASSWORD

#define FIREBASE_HOST "homeautomation-asg-default-rtdb.firebaseio.com"

#defineFIREBASE_AUTH"******************************”

/* data received from application */

String  data ="";

/* define L298N or L293D motor control pins */

int Relay1 = 15;

int Relay2 = 16;

int Relay3 = 17;

int Relay4 = 18;

static const int servoPin = 26;

static const int servoPin1 = 27;

int g = 14;

int gas;

String car;
```

```
Servo servo1;

Servo servo2;

int angle = 0;

int angleStep =5;

int angleMin = 0;

int angleMax = 90;

void setup()

{

 /* initialize motor control pins as output */

 pinMode(Relay1, OUTPUT);

 pinMode(Relay2, OUTPUT);

 pinMode(Relay3, OUTPUT);

 pinMode(Relay4, OUTPUT);

 pinMode(LED_BUILTIN,OUTPUT);

 pinMode(23,INPUT);

 pinMode(22,INPUT);

 pinMode(13,INPUT);

 pinMode(21,OUTPUT); // voice led hall

 pinMode(25,OUTPUT); //voice fan bedroom

 servo1.attach(servoPin);

 servo2.attach(servoPin1);
```

```
 digitalWrite(Relay1,LOW);
 digitalWrite(Relay2,LOW);
 digitalWrite(Relay3,LOW);
 digitalWrite(Relay4,LOW);
 /* start server communication */
  Serial.begin(9600);
  connectWiFi();
  server.begin();
}
uint32_t x=0;
void loop()
{
 MQTT_connect();
Adafruit_MQTT_Subscribe *subscription;
while ((subscription = mqtt.readSubscription(5000))) {
if (subscription == &relay) {
Serial.print(F("Got: "));
Serial.println((char *)relay.lastread);
if (!strcmp((char*) relay.lastread, "On"))
 {
 digitalWrite(24, HIGH);
```

```
}
 else
 {
digitalWrite(24, LOW);
 }
 }
 if (subscription == &relaym) {
 Serial.print(F("Got: "));
 Serial.println((char *)relaym.lastread);
 if (!strcmp((char*) relaym.lastread, "On"))
 {
 digitalWrite(25, HIGH);
 }
 else
 {
 digitalWrite(25, LOW);
 }
 }
 gas=analogRead(g);
 if(gas>2000)
```

```
{
//Serial.println(gas);
Firebase.setString(firebaseData,"iotproject/gas","gas_leaking")
}
else
{
Firebase.setString(firebaseData,"iotproject/gas","gas_not_leaking");
//Serial.print("no gas leakage");
}
if (digitalRead(13)==HIGH)
{
Firebase.setString(firebaseData,"iotproject/pir","intrusion_not_detected");
}
else
{
Firebase.setString(firebaseData,"iotproject/pir","intrusion_detected");
}
if (digitalRead(23)==HIGH
{
car = "car_not_parked";
Firebase.setString(firebaseData,” iot project/car",car);
```

```
//Serial.println(car);
}
else
{
car = "car_parked";
 Firebase.setString(firebaseData,"iot project/car",car);
//Serial.println(car);
}
if (digitalRead(22)==HIGH)
{
Firebase.setString(firebaseData,"iotproject/fire","fire_not_detected");
}
else
{
Firebase.setString(firebaseData,"iotproject/fire","fire_detected");
}
/* If the server available, run the "checkClient" function */
client = server.available();
if (!client) return;
data = checkClient ();
Serial.print(data);
```

```
/*** Run function according to incoming data from application ****/
 if (data == "Relay1ON")
 {
 digitalWrite(Relay1,HIGH);
 }
 else if (data == "Relay1OFF")
 {
 digitalWrite(Relay1,LOW);
 }
 else if (data == "Relay2ON")
 {
 digitalWrite(Relay2,HIGH);
 }
 else if (data == "Relay2OFF")
 {
 digitalWrite(Relay2,LOW);
 }
 else if (data == "Relay3ON")
{
digitalWrite(Relay3,HIGH);
}
```

```
else if (data == "Relay3OFF")
{
digitalWrite(Relay3,LOW);
}
else if (data == "Relay4ON")
{
 digitalWrite(Relay4,HIGH);
}
else if (data == "Relay4OFF")
{
digitalWrite(Relay4,LOW);
}
else if (data == "RELAY5SUCCESS")
{
for(int angle = 0; angle <= angleMax; angle +=angleStep) {
   servo1.write(angle);
delay(20);
}
delay(5000);
for(int angle = 90; angle >= angleMin; angle -=angleStep) {
servo1.write(angle);
```

```
delay(20);
}
}
else if (data == "RELAY6SUCCESS")
{
for(int angle = 0; angle <= angleMax; angle +=angleStep) {
servo2.write(angle);
delay(20);
}
delay(5000);
for(int angle = 90; angle >= angleMin; angle -=angleStep) {
servo2.write(angle);
delay(20);
}
}
}
}
void connectWiFi()
{
 Serial.println("Connecting to WIFI");
```

```
 WiFi.begin(ssid, password);
while ((!(WiFi.status() == WL_CONNECTED)))
{
 delay(300);
 Serial.print("..");
}
Serial.println("");
digitalWrite(LED_BUILTIN,HIGH);
Serial.println("WiFi connected");
Serial.println("NodeMCU Local IP is : ");
Serial.print((WiFi.localIP()));
Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
Firebase.reconnectWiFi(true);
}
void MQTT_connect() {
int8_t ret;
// Stop if already connected.
if (mqtt.connected()) {
return;
}
Serial.print("Connecting to MQTT... ");
```

```
uint8_t retries = 3;
while ((ret = mqtt.connected()) != 0) { // connect will return 0 for connected
Serial.println(mqtt.connectErrorString(ret));
Serial.println("Retrying MQTT connection in 5 seconds...");
mqtt.disconnect();
delay(5000);  // wait 5 seconds
retries--;
if (retries == 0) {
// basically die and wait for WDT to reset me
while (1);
}
}
Serial.println("MQTT Connected!");
}
/**** RECEIVE DATA FROM the APP ******/
String checkClient (void)
{
while(!client.available()) delay(1);
String request = client.readStringUntil('\r');
request.remove(0, 5);
request.remove(request.length()-9,9);
```

```
return request;

}
```

8.2 Source Code for ARDUINO UNO

```
#include<SoftwareSerial.h>

#include <Servo.h>

Servo myservo;

int initial_position = 90;

int LDR1 = A0;          //connect The LDR1 on Pin A0 ---- solar

int LDR2 = A1;          //Connect The LDR2 on pin A1 ----- solar

int error = 5;

int servopin=3;

int sensorPin= A2;  //automatic on and off  -- A2

int sensorValue = 0;

int ledo = 7;       //automatic on and off  -- orange 12v

int ledb = 8;      //automatic on and off  -- blue 12v

int in = 2;        //physcial intrusion ldr

int LEDr = 4;      // ledred

int LEDb = 5;      // ledblue
```

```
int BUZZ = 6;      //buzzer

void setup() {

  pinMode(in,INPUT);

  pinMode(ledo, OUTPUT);

  pinMode(ledb, OUTPUT);

  digitalWrite(ledo,LOW);

  digitalWrite(ledb,LOW);

  pinMode(LEDr,OUTPUT);

  pinMode(LEDb,OUTPUT);

  pinMode(BUZZ,OUTPUT);

  myservo.attach(servopin);

  pinMode(LDR1, INPUT);

  pinMode(LDR2, INPUT);

  myservo.write(initial_position);   //Move servo at 90 degree

  delay(2000);

  Serial.begin(9600);

}

void loop() {

  // put your main code here, to run repeatedly:
```

```
int R1 = analogRead(LDR1); // read  LDR 1
 int R2 = analogRead(LDR2); // read  LDR 2
 int diff1= abs(R1 - R2);
 int diff2= abs(R2 - R1);
 Serial.println(R1);
 Serial.println(R2);
 if((diff1 <= error) || (diff2 <= error)) {
 } else {
  if(R1 > R2)
  {
    initial_position = --initial_position;
  }
  if(R1 < R2)
  {
    initial_position = ++initial_position;
  }
 }
 myservo.write(initial_position);
 delay(100);
```

```
sensorValue = analogRead(sensorPin);
 Serial.println(sensorValue);
 if(sensorValue < 100)
 {
   Serial.println("LED light on");
   digitalWrite(ledb,LOW);
   digitalWrite(ledo,HIGH);
 }
 else
 {
  digitalWrite(ledo,LOW);
  digitalWrite(ledb,HIGH);
 }
if (digitalRead(in)==HIGH){
  digitalWrite(LEDb,LOW);
  digitalWrite(LEDr,HIGH);
  digitalWrite(BUZZ,HIGH);
}
else
```

```
{
   digitalWrite(LEDr,LOW);
  digitalWrite(BUZZ,LOW);
  digitalWrite(LEDb,HIGH);
}
delay(1000);
}
```

8.3 Screen shot -Application

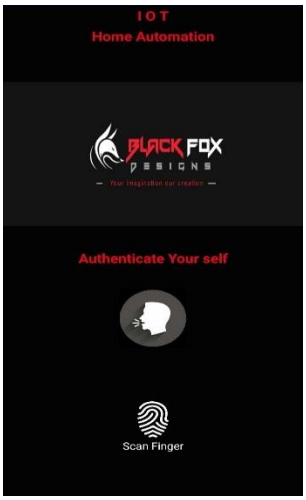


Figure 8.3.1 Authentication for Application Login

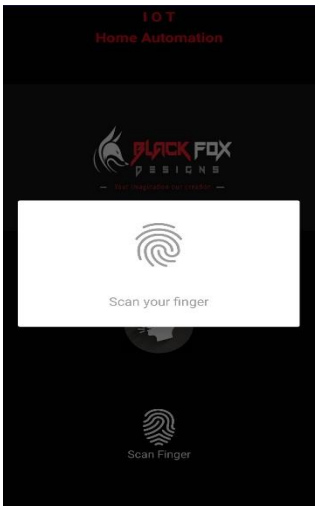


Figure 8.3.2 Authentication Using Fingerprint

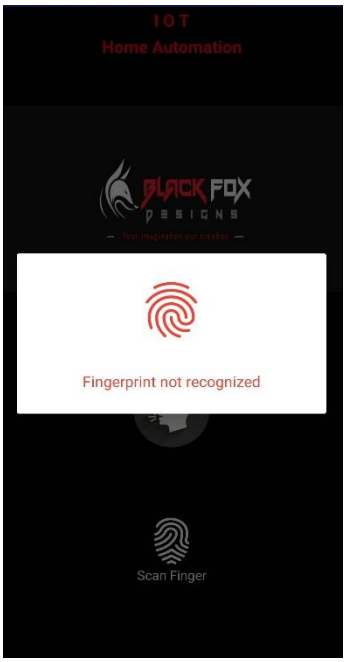


Figure 8.3.3 Fingerprint Authentication Failed

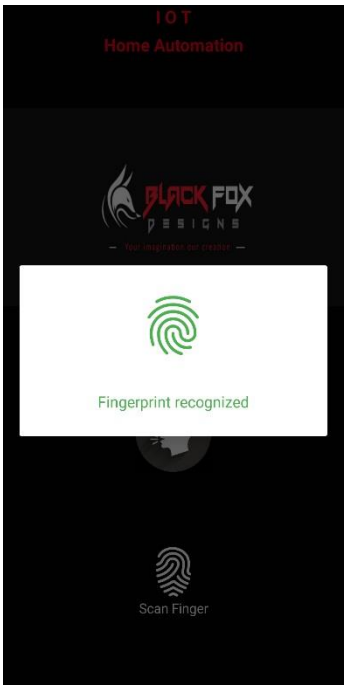


Figure 8.3.4 Fingerprint Authentication Successful

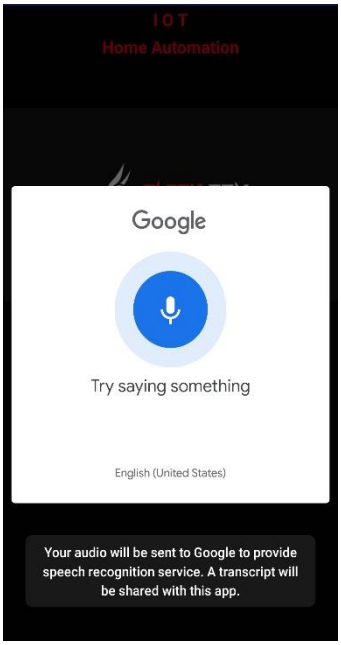


Figure 8.3.5 Authentication Using Voice

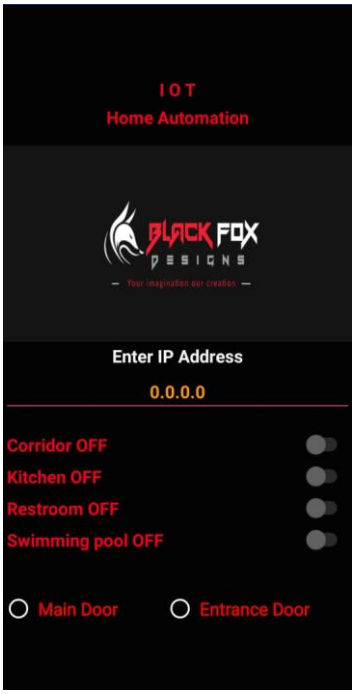


Figure 8.3.6 Main User Interface

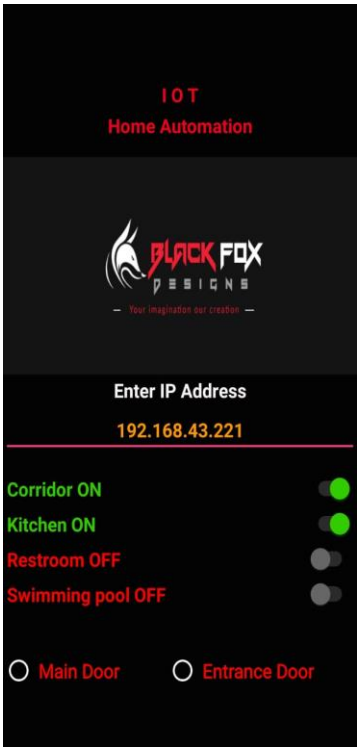


Figure 8.3.7 Switch On and Off to Control Appliances

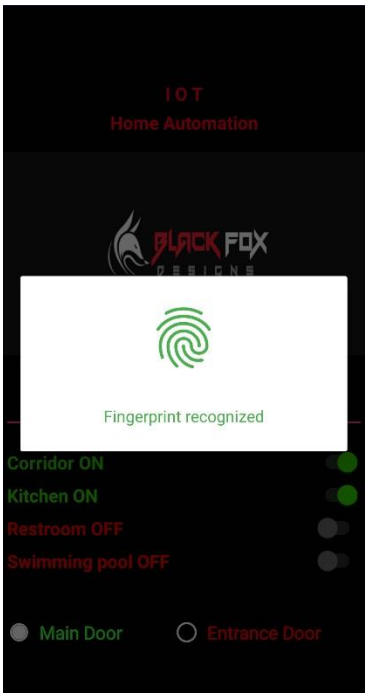


Figure 8.3.8 Door opening using fingerprint

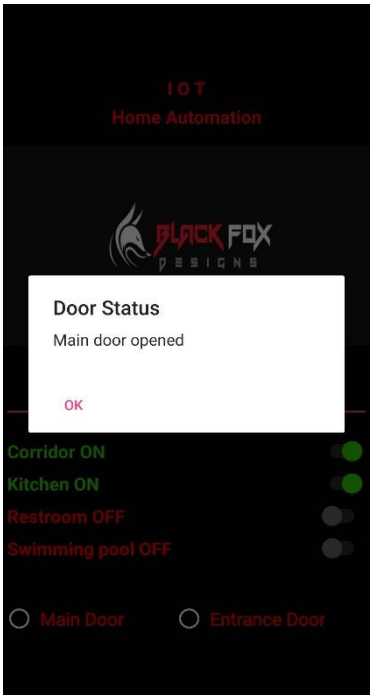


Figure 8.3.9 Notification for Door Opened

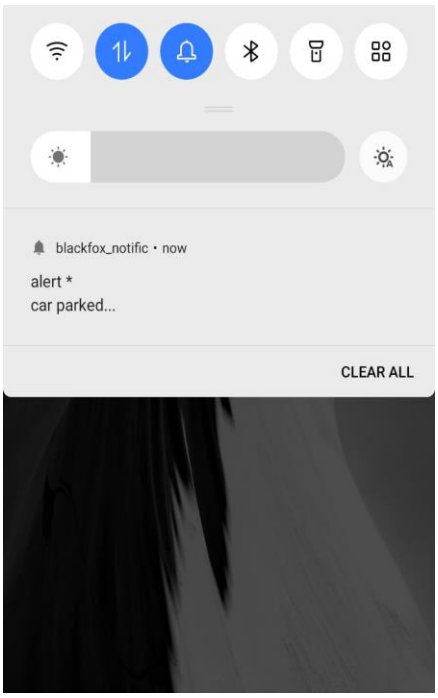


Figure 8.3.10 Notification for Car Parked

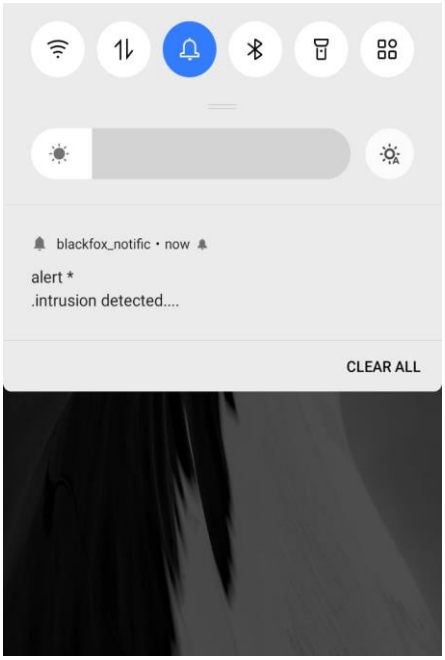


Figure 8.3.11Notification for Intrusion Detected

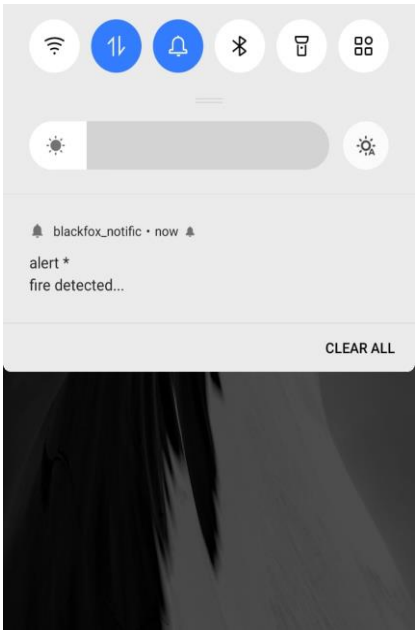


Figure 8.3.12 Notification for Fire Detected

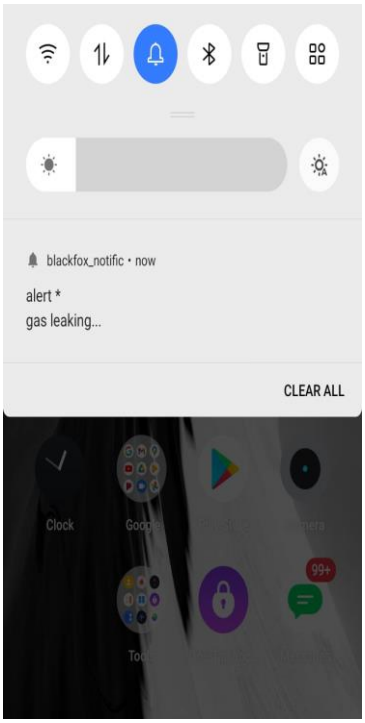


Figure 8.3.13Notification for Gas Leaking

8.4 Screen shot-Setting up Google Firebase

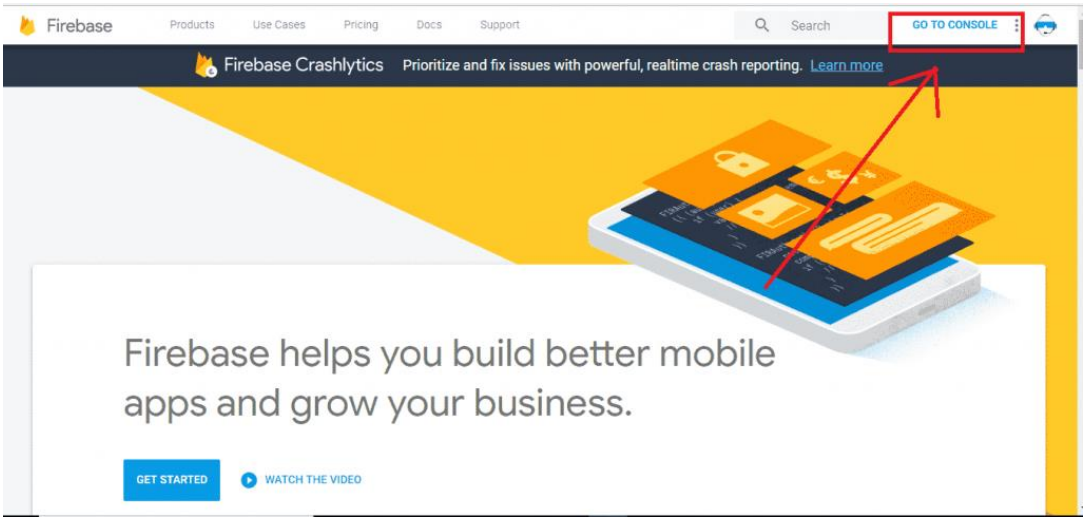


Figure 8.4.1 Going to Console

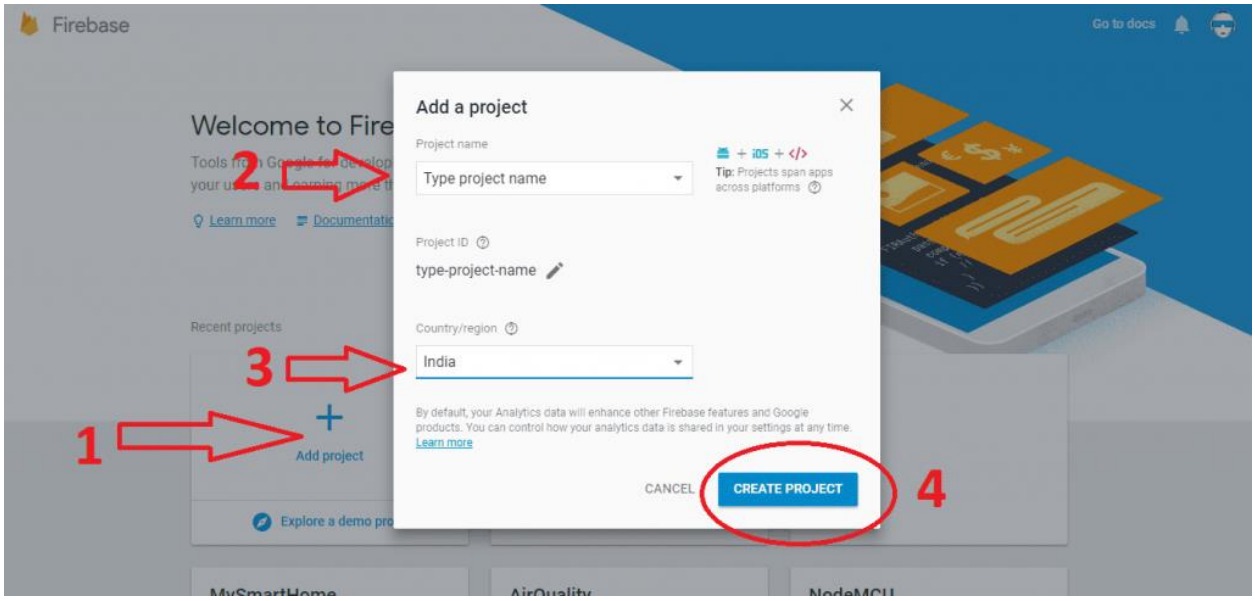


Figure 8.4.2 Creating New Project

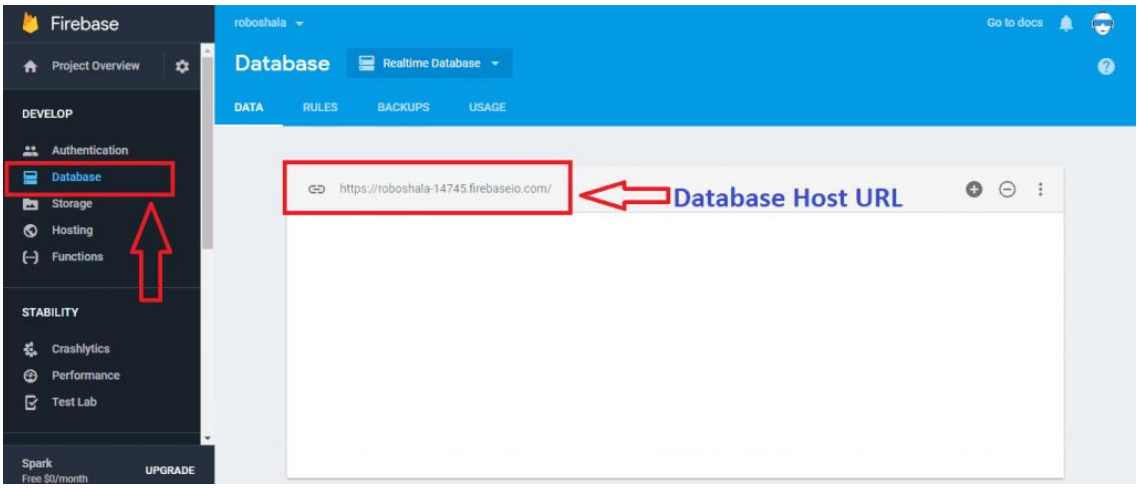


Figure 8.4.3 Creating Database

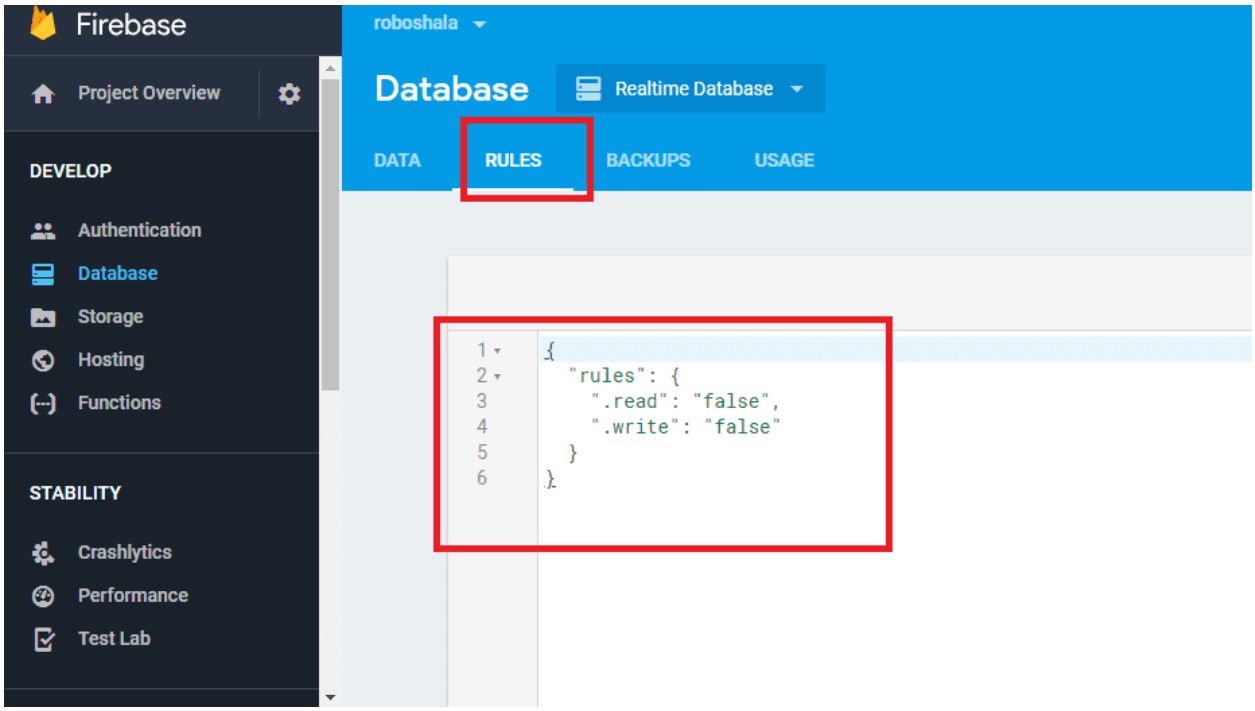


Figure 8.4.4 Setting up Rules

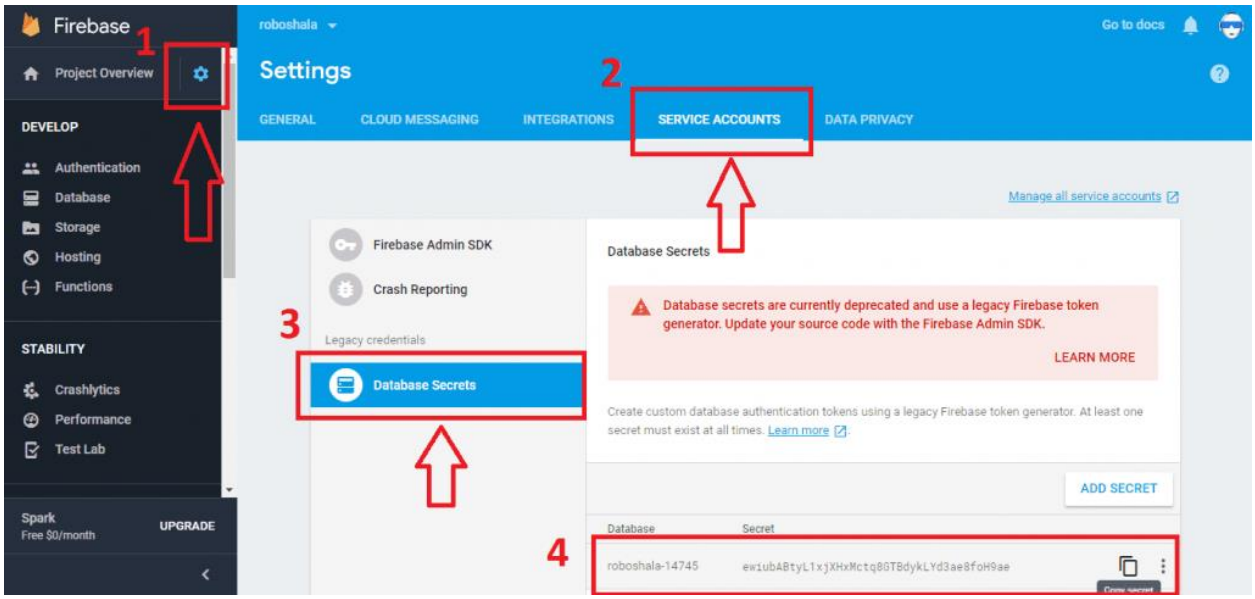


Figure 8.4.5 Getting Project's Secret Key

8.5 Screen shot- Setting up Adafruit and Connecting to Google Assistant

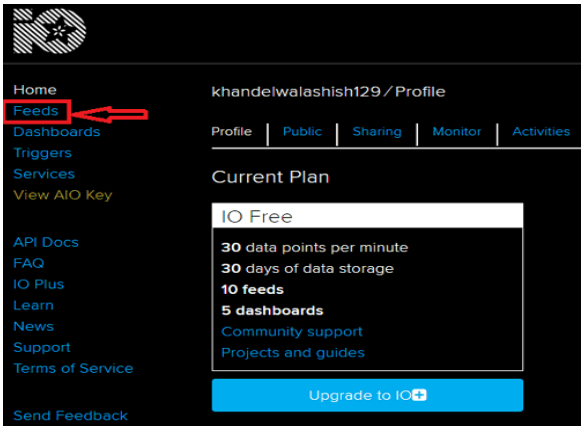


Figure 8.5.1 Clicking on Feeds

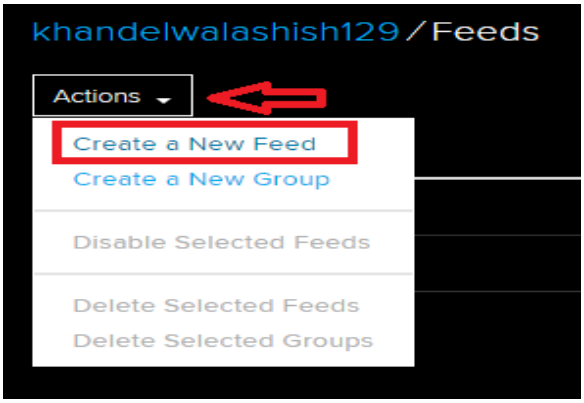


Figure 8.5.2 Creating New Feeds

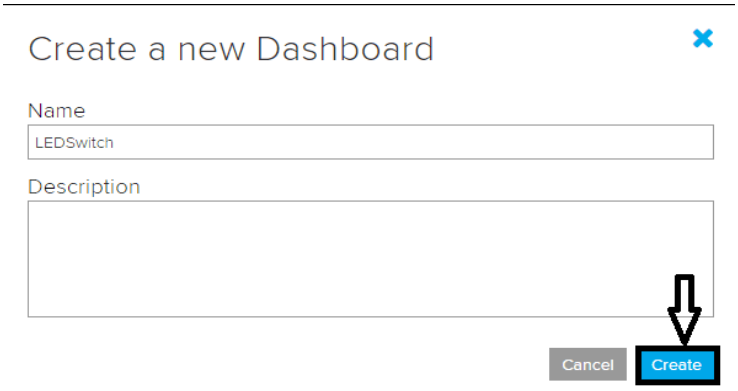


Figure 8.5.3 Creating New Dashboards

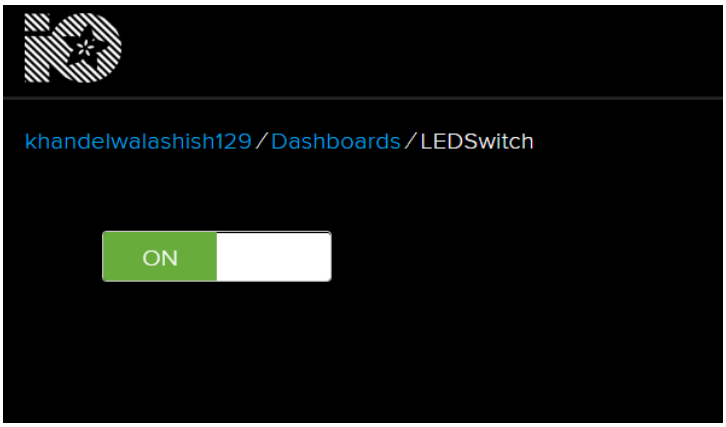


Figure 8.5.4 Creating New Blocks

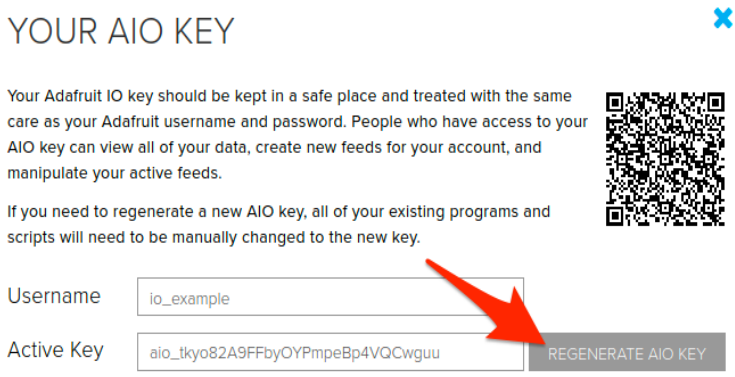


Figure 8.5.5 Getting AIO Key

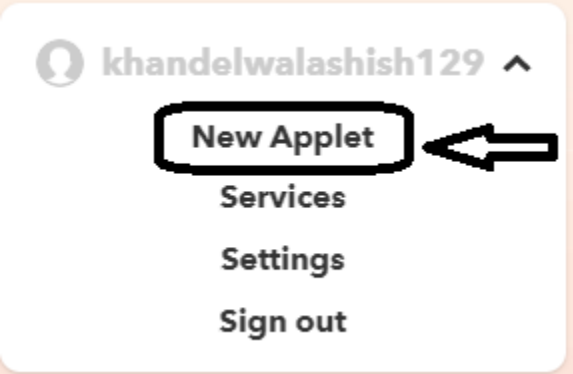


Figure 8.5.6 Creating New Applet in IFTT

Figure 8.5.7 Creating Trigger

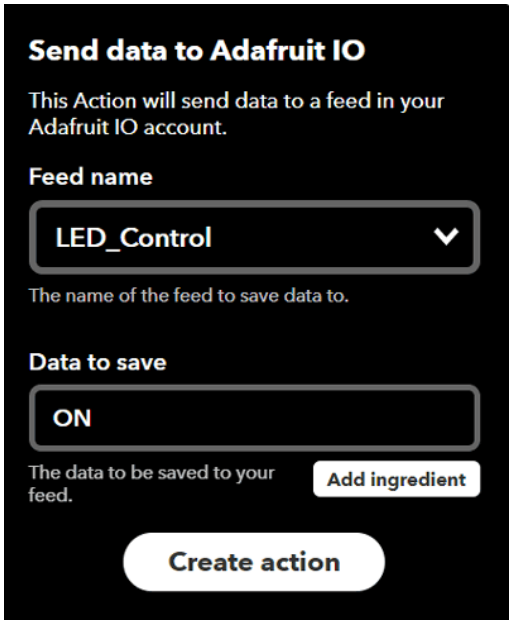


Figure 8.5.8 Sending Data to Adafruit

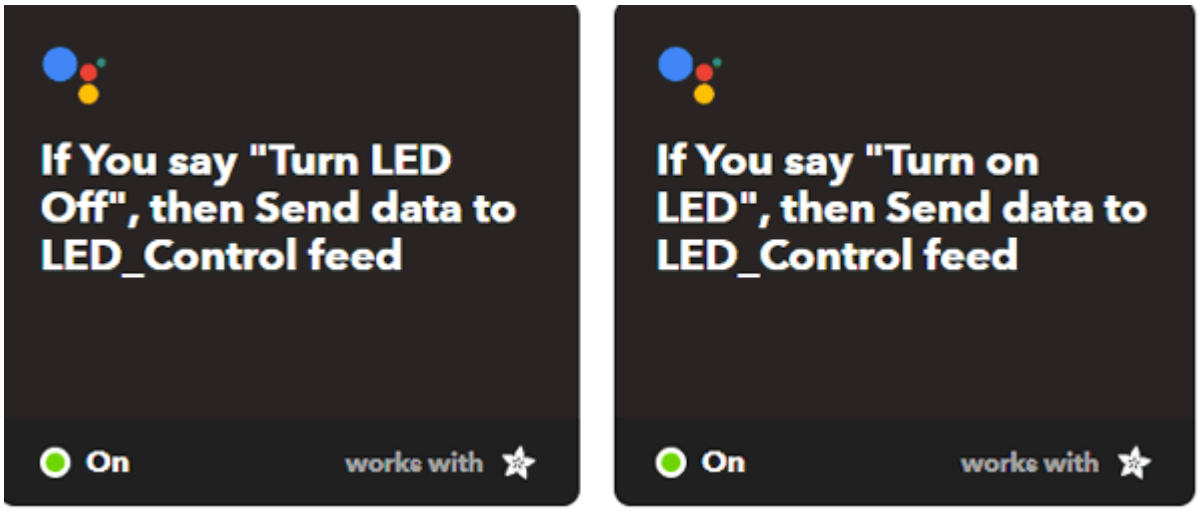


Figure 8.5.9 My Applets