CZECH TECHNICAL UNIVERSITY IN PRAGUE

FACULTY OF ELECTRICAL ENGINEERING
Department of Microelectronics

Diploma Thesis

Security System Using ESP32

Author: Ashish Garg
Supervisor: prof. Ing. Miroslav Husák, CSc.
Prague, 2020

MASTER'S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Garg Ashish** Personal ID number: **481069**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Microelectronics**

Study program: **Electronics and Communications**

Specialisation: **Electronics**

II. Master's thesis details

Master's thesis title in English:

Security system using ESP32

Master's thesis title in Czech:

Security system using ESP32

Guidelines:

1. Discuss the possibilities of using ESP32 to provide basic security applications: Switch-on/off, vehicle theft, security for garden gate, fire alarm (motion and magnetic sensors).
2. Design a simple security system with ESP32 control. The system must be able to provide: Switch-on/off, vehicle theft security, keypad security for garden gate, fire alarm, door and window security, use magnetic and PIR motion sensors to design of the alarm system.
3. Evaluate the parameters of the proposed system and compare it with commercial systems.

Bibliography / sources:

1. S. Sanjay Kumar, A. Khalkho, S. Agarwal, S. Prakash, D. Prasad, and V. Nath, "Design of smart security systems for home automation," in Lecture Notes in Electrical Engineering, 2019.
2. A. Anitha, "Home security system using internet of things," in IOP Conference Series: Materials Science and Engineering, 2017.
3. S. Tanwar, P. Patel, K. Patel, S. Tyagi, N. Kumar, and M. S. Obaidat, "An advanced Internet of Thing based Security Alert System for Smart Home," in IEEE CITS 2017 - 2017 International Conference on Computer, Information and Telecommunication Systems, 2017.

Name and workplace of master's thesis supervisor:

prof. Ing. Miroslav Husák, CSc., Department of Microelectronics, FEE

Name and workplace of second master's thesis supervisor or consultant:

Date of master's thesis assignment: **10.02.2020** Deadline for master's thesis submission: ________

Assignment valid until: **30.09.2021**

prof. Ing. Miroslav Husák, CSc.	prof. Ing. Pavel Hazdra, CSc.	prof. Mgr. Petr Páta, Ph.D.
Supervisor's signature	Head of department's signature	Dean's signature

III. Assignment receipt

The student acknowledges that the master's thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the master's thesis, the author must state the names of consultants and include a list of references.

Date of assignment receipt ________ Student's signature ________

Declaration:

I hereby declare that I have completed this thesis independently and that I have used only the sources (literature and webpages) listed in the enclosed bibliography.

Date Signature

Acknowledgment:

I would first like to thank my supervisor, prof. Ing. Husák Miroslav, CSc for guidance and motivation to make my thesis success. He always helped me when I ran into the problem by giving valuable advice. I would also like to thank my parents, family members, and friends for encouragement and moral support.

Abstract:

Home security is a critical issue for people living not only in cities but also in urban and rural areas. Now with the advancement in the Internet of Things (IoT), it is possible to monitor your house from anywhere in the world. This Master's thesis deals with designing the Home Electronic Security System using ESP32 as a main controller of the full design. In the theoretical part of the thesis, first basics of the Electronic Security System and the Internet of Things (IoT) are discussed. After that work of some researchers from the past couple of years in this field is summarized. A comparison with different commercially available products is made. In the practical part of the thesis, a prototype of the house is made, and different sensors connected with ESP8266 are installed at the possible places of the house to protect from an intruder. MQTT protocol, Ubidots IoT platform for data visualization, and for control of system both from web and smartphone are used. IFTTT online service is used for notification and data storage on Google sheets.

Keywords: Electronic Security System, MQTT, Internet of Things (IoT), ESP32, ESP8266

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Abbreviations

IoT: Internet of Things

MQTT: Message Queue Telemetry Transport

QoS: Quality of Service

IFTTT: If This Then That

ESS: Electronic Security System

IDE: Integrated development environment

BLE: Bluetooth Low Energy

PIR: Passive Infrared sensor

IEEE: Institute of Electrical and Electronics Engineers

SPI: Serial Peripheral Interface

SPIFFS: Serial Peripheral Interface Flash File System

GPIO: General Purpose Input Output

ISM: Industrial, Scientific and Medical band

1. Introduction

“Security” is one of the essential things for human beings. Human from the ancient time practice the various method to be secure by family point of view and also individually. Like ancient Humans used to keep the dog with them or burn the fire.

Today in the era of science, there are many methods we use to protect our house, and one of the most used methods is the lock. We use the mechanical locks in front of the doors. And we have the keys to unlock. But nowadays only locking the door is not sufficient as the intruders easily break the door lock and enter the house.

So the electronic security system is one of the solution. The electronic security and fire alarm industry’s remarkable history dates back to the mid-1850s when pioneers led by John Gamewell and Edwin Holmes transformed inventions into business that protected property and life as never imagined [1].

The first electro-magnetic alarm system in the world was patented on 21 June 1853 in the name of a man called Augustus Russell Pope, an inventor from Sommerville in Boston [2]. After that, Edwin Holmes acquired Pope’s patent rights in 1857 and manufactured the device in his factory in Boston, Massachusetts. He began to sell them in 1858 [3].

John Nelson Gamewell saw the Channing and Farmer fire alarm system, recognized its value, and in 1855 purchased the rights to construct the fire alarm. In 1859 he purchased all of the patents and launched his career in the Fire Alarm Telegraph field devoted his entire business life to its introduction and improvement [4]. So like this, the electronic security systems has a long history.

Today internet technology has revolutionized security systems communications and monitoring capabilities. With the advent of digital communication in the 1970’s, central station signaling finally took a major evolutionary step toward the technology we recognize today. That technology continues to advance, with wireless communications now joined by IP connectivity [1].

Any systems that provide its owner/user with a reasonable degree of protection against one or more real or imagined dangers, threats, or nuisances can be described as a “security” system [5]. An electronic security system is one in which the systems’ actions are heavily dependent on electronic circuitry like keypad door locks, burglar alarms, etc. [5].

Today’s modern electronic security circuits and systems use different wireless communications to control the devices remotely. It consists of automated systems comprises sensors, actuators, and controllers to enhance the safety and security of the residents. But most of the available systems are not suitable for many users due to non-user-friendly interface, maintenance difficulty, and high cost.

2. Electronic Security Systems

There are so many electronic security systems in the market, with some of them having extra features and some not. But as per SafeWise website, a security system review website suggests that motion sensors, indoor and outdoor cameras, glass break detectors, door and window sensors, yard signs and window stickers, are some of the main features/components which together makes a complete Home Security System [6].

A residence that is visible to neighbors and the street discourages the potential intruder. There are certain number of things that can discourage the intruder from entering in the building such as [7], [8]:

- Visibly installed alarm
- Fixed window grilles
- Massive entrance door
- Quality locks
- Free view of the property
- Lighting of the property and the house
- Dog on the property
- Warning signs with different warning information
- Presence of residents
- Trim bushes/shrubs so that windows and doors are visible, and due to that intruder have to work in open whose detection is easy.
- Do not leave the ladder or any tool outside, which can be used by intruder to enter in the house.
- Do not hide keys outside the residence as the hiding places are quite obvious, and intruder can easily locate those spots.
- Do not reveal personal information on your mailbox or doorbell.
- Keep garage doors locked.

And certain number of things which attracts intruder such as [7]:

- covered view of the property (high fences, high walls, high growth of grasses/bushes)
- obscured view of places of possible penetration into the object signs of long-term absence (not picking up a mailbox, clearing snow or leaves, non-smoking chimneys, etc.)
- carelessness and negligence (leaving doors and windows open, hiding keys, loose tools, etc.)
- Discrete entrance to house
- Lack of visible security measures
- A fixed routine of going outside.

2.1 Protection against the entrance to the building

There are 4 levels of protection [9] as shown in **Figure 1** :

1. **Perimeter (1st Level):** It is the protection around the house.
2. **Building Wall (2nd level):** It is the protection in the form of Doors, Windows.
3. **Space (3rd Level):** After entering into the house, the protection of the space like corridors.
4. **Object (4th Level):** It is the protection of the objects/ valuable assets in the house.

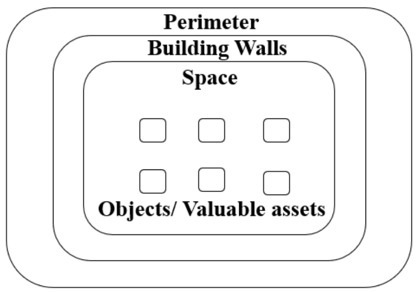


Figure 1: Levels of Protection [9]

We can install different types of sensors at every level of protection like for perimeter, we can use a camera, microwave barrier. For Building walls, we can use magnetic switches on doors, glass break sensors on windows. For Space protection, we can use motion detectors, ultrasound detectors. For Object protection, we can use seismic sensors, capacitive sensors.

All security system consists of the essential elements, as shown in **Figure 2**. One or more danger sensing units like different types of sensors are used, which generate some electrical output, and that output is fed into the decision-making signal processing unit, or we can say Central Control Unit (CCU) via data link. Data Link can be wired, wireless or mix type. The output from the CCU is fed into the danger response unit, such as a buzzer, any trigger, etc. [5].

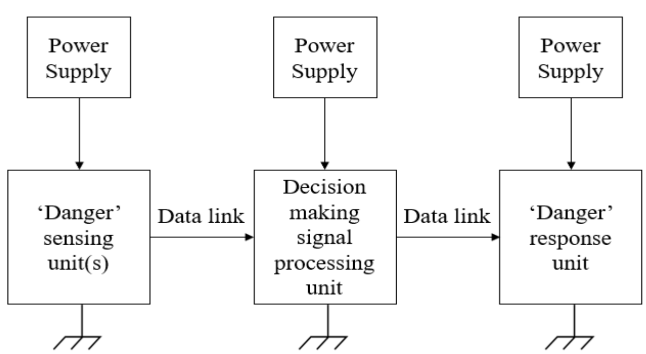


Figure 2: Basic elements of an Electronic Security System [5]

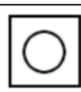
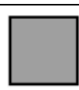
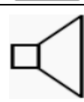
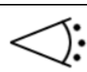
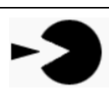
2.2 Reliability of the Security System

One of the most important factors in the Electronic Security System is the reliability of the system in performing the tasks. It can be defined as the probability of the system that does not fail to do the designated task in the prescribed period of time [5], [10]. The reliability of the Electronic Security System is dependent on the reliability of the components used in the system, for example, the danger sensing unit (sensors), the data link, etc. and can vary with the level of security that the system is designed to provide [5]. For example, reed switches, pressure pads, electronic keypad security switch are more reliable [5].

2.3 Security Symbols

There are several symbols used in the electronic security system to indicate the sensors in the design of the flat. Those symbols (only a few) are as follows (**Table 1**):

Table 1: Few symbols used in Electronic security systems [11]

Symbol	**Meaning**
	Panic Button
	PIR sensor
	Magnetic Contact
	Burglar alarm
	Speaker/ Siren alarm
	Central monitoring station
	Keypad
	Infrared motion sensor
	Ultrasonic motion sensor
	Broken glass sensor

3. Internet of Things (IoT)

According to the International Telecommunication Union (ITU), the Internet of Things (IoT) is ‘‘a global infrastructure for the Information Society, enabling advanced services by interconnecting (physical and virtual) things based on, existing and evolving, interoperable information and communication technologies [12]. In the past years, there is tremendous growth in the interest in the IoT. Companies introduce numerous products in the market such as Smart Home, Smart Security, Smart Meters, Smart transport including vehicle fleet tracking, mobile tracking, Smart health solutions, real-time monitoring, Smart Industries under the name Industry 4.0, and many more [13]. International Data Corporation (IDC) predicts the IoT global market revenue to reach approximately US$1.1 trillion by 2025. Global IoT connections are predicted to increase with 17% CAGR (Compound Annual Growth Rate) from 7 billion to 25 billion approximately from 2017 to 2025 [14]. In 2018, Home Security Market was of USD 45.52 billion and is expected to reach USD 74.75 billion by 2023 [15].

Components of the Internet of Things (IoT) is shown in **Figure 3**.

Figure 3: Components of IoT [16]

Explanation of **Figure 3** [16]:

1. **Devices** and smart sensors continuously collect data from the environment and transmit to Gateway via different wireless technologies such as Wi-Fi, ZigBee, Bluetooth, Z-wave, LoRAWAN, etc. Each of these wireless technologies has its pros and cons.

2. IoT **Gateway** manages the bidirectional data traffic between different networks and protocols. Another function of the gateway is to translate different network protocols and make sure the interoperability of the connected devices and sensors.
3. A massive amount of data is collected from devices, applications, and users, which has to be managed efficiently. **IoT cloud** offers tools to collect, process, manage, and store massive amounts of data in real-time. Industries can efficiently access these data.
4. Applying smart **analytics** to convert the analog data from smart devices to a useful format that can be interpreted efficiently.
5. **User interfaces** are the visible part of the IoT system. Designers have to make them a well suitable interface that can be easily accessed by the user.

3.1 Overview of Different Wireless Technologies

There are many technologies available for designing smart services, but everyone has its advantages and disadvantages.

3.1.1 Wi-Fi

Wi-Fi is the wireless networking technology that allows different devices such as laptops, printers, smartphones, etc. to interface with the internet and allows the exchange of the data between them. It is based on IEEE 802.11 standard and commonly used for Local Area Networking of devices and internet access. It works in 2.4 and 5 GHz range. WiFi is the trademark of the WiFi Alliance, which is responsible for the WiFi certification and testing of the devices. WiFi has high bandwidth, and due to which it has a signal loss, and it is power-intensive. However, IEEE 802.11ah and 802.11 ax attracted for the IoT applications due to low power consumption[17][18].

3.1.2 LoRa

It is a 'Long Range' wireless modulation technique. It uses license-free (ISM band) sub-gigahertz radio frequency bands like 433 MHz, 868 MHz (Europe), and 915 MHz (Australia and North America). It is presented in two parts: LoRa, the physical layer using the Chirp Spread Spectrum (CSS) radio modulation technique, and MAC layer protocol (LoRaWAN (Long Range Wide Area Network), the upper layers) [19]. LoRaWAN provides a medium access control mechanism, enabling many end-devices to communicate with a gateway using the LoRa modulation, and the LoRaWAN is an open standard being developed by the LoRa Alliance [19]. LoRa can be deployed where end devices have limited energy available, no need of more byte transfer and where data traffic can be initiated either by the end-device (such as when the end-device is a sensor) or by an external entity wishing to communicate with the end-device (such as when the end-device is an actuator) [19]. The long-range, low cost, more battery lifetime and low-power nature of LoRa makes it an interesting candidate for smart sensing technology in such as health monitoring, smart metering, environment monitoring, etc. as well as in industrial applications [19].

3.1.3 Sigfox

It is an excellent example of LPWAN (Low Power Wide Area Network) on an unlicensed spectrum. It is a proprietary technology based on France, which provides end to end IoT connectivity. It uses 868 MHz ISM band to transmit messages via ultra-narrowband (UNB) modulation (100 Hz) to base stations in a star topology at data rates varying between 100 and 600 bits per second (depending on the region) [20]. From 2009 it is growing rapidly and, as per Nov 2018, covers almost 53 countries with an approximate coverage area of 5 million square kilometers [21]. Key features of the Sigfox are Long Range, High energy efficiency, simple connectivity, low cost [20], [21].

3.1.4 NB-IoT

Narrowband IoT (NB-IoT), is an LPWAN developed by 3GPP (3rd Generation Partnership Project) supported by cellular network operators to meet the coverage requirements in rural and deep indoors locations set by IoT devices and it supports a lower power IoT connection [20]. As it uses the licensed frequency bands, so there are no duty cycle limitations. Thus, it can offer vastly higher data throughput than Sigfox and LoRaWAN [20]. Key features of NB-IoT are indoor coverage, low cost, long battery life, and high connection density.

3.1.5 ZigBee

Zigbee is a wireless mesh network by the Zigbee Alliance for low-cost, low-power, low data rate, and short-distance wireless control and monitoring solutions. It is based on the IEEE 802.15.4 standard for Personal Area Network and operates on the 2.4 GHz ISM band and 868/915 MHz. It is used in IoT for small area remote monitoring, security, automatic household control, and other places which deployed sensor network-based applications [17] [22].

3.1.6 Bluetooth Low Energy (BLE)

It is a Short-range, Low power, low-cost Wireless Personal Area Network communication technology. It is based on the IEEE 802.15.1 standard and operates in the 2.4 GHz ISM band. The energy needed for data transfer is low. Bluetooth Low Energy is now standardized in all smartphones, tablets, and laptops, in addition, a wide range of other devices [17].

A comparison between different wireless technologies is shown in **Table 2**.

Table 2: Comparison between different technologies

Parameters	Wi-Fi	LoRa	Sigfox	NB-IoT	ZigBee	Bluetooth (BLE)
Standard	802.11	LoRa-Alliance	Sigfox company is collaborating with ETSI on the standardization of Sigfox-based network	3GPP	802.15.4	802.15.1
Data Rate	Up to 72 Mbps	50kbps	100bps	200kbps	Up to 250 kbps	Up to 2 Mbps
Range	100m	2-5km (urban), 15-20km (rural)	3-10km (urban), 30-50km (rural)	1 km (urban), 10 km(rural)	100m	10m
Network Topology	Star/Mesh	Star	Star	Cellular network	Star/Mesh	Point to Point
Frequency	2.4 and 5 GHz	Unlicensed ISM bands	Unlicensed ISM bands (868 MHz in Europe, 915MHz in North America, and 433 MHz in Asia)	Cellular Band (Licensed LTE frequency bands)	2.4 GHz and 868/915 MHz	2.4 GHz
Power Consumption	High	Low	Low	Medium	Low	Medium
Cost	Low	Medium	Medium	High	Medium	Low

4. Literature Review

In this section, previous work by some researchers is presented.

After the inception of the Internet of Things (IoT), Smart Homes (SHs) have been widely accepted by the people due to its advantages like comfort, Security of the Homes and due to this, there is an increase in the market revenue of the Smart Home [23].

For resolving the above problems, many researchers proposed and designed the Smart Home security system.

In [23], the author proposed a low cost and minimum delay, advanced IoT based home security system. They used the Raspberry Pie as a main computational device, Pyroelectric Infrared (PIR) module, and Webcam. The system assumes an internet connection and also email the resident real-time data. The system compares the previously stored data and new data, and if it is different, then it sends the email. Webcam and PIR sensor detect any suspicious activity when the user is not at home and sends the signals and captured images to Raspberry Pie, and then it sends the captured images to the owner via email. In [24], the author used Raspberry Pie, IP Web-camera and Fog Computing for limiting the delay during sending the alert message and increasing the security level. The webcam captures live video and matched with the registered sample. If the sample is mismatched, then an email alert is sent to the owner. In [25], the author designed a cost-effective smart door using Reed Switch, Elegoo Mega 2560 microcontroller, Raspberry Pi 2 board for communicating with the webserver that uses RESTful API. RF transmitter is connected with the Elegoo board and receiver with Raspberry Pi 2. When the reed switch is opened, the Raspberry Pie sends HTTP POST request to the webserver, which pushes the information so that the user views the door events from the developed Android application. In [26], the author proposed a Smart and secure Home Automation system using Raspberry Pi 3 as Central Hub, Arduino -nano board for various nodes around the house, NRF24L trans-receiver for communication with sensor nodes and central hub. In the central hub, the author used Debian based OS runs on Eclipse Smart Home with Java Virtual Machine (JVM). Nodes senses the data and sends it to a central hub via NRF24L than it sends the data to the MQTT channel where MQTT binding picks it. Then, an alert message is displayed on the web browser and Smartphone. Their system can also communicate with Z-Wave, ZigBee, Smart TVs, etc. the data that is received is saved and plotted using the SQL database. This data is monitored for some time and then predicts the future. On every switch, energy consumption is monitored and compared with the stored value and declare theta whether an appliance is consuming more power or not. They used the PIR sensor, MQ5 for smoke detection, DHT22 for temperature and humidity, LDR for light, and Reed Switch for door. In [27], the author proposed a voice-controlled multifunctional Secure home using Voice Recognition Module V3, RF Module, Arduino Uno, Raspberry Pi Model B+, DHT11 sensor for temperature and humidity and PIR sensor for motion detection. RF transmitter is connected to the Arduino Uno board and receiver to Raspberry Pi board. Once the user speaks out the voice command, it is recognized by the Voice Recognition module and command encoded into unique decimal value. This value is transmitted to the receiver. The receiver module is connected to all the sensors used in the system. It will decode the received value and execute the corresponding operation. The receiver side is connected to the internet, and all the data is obtained is sent to the cloud and stored in the form of Google sheets. In [28], the author proposed a low cost and multifunctional Smart House prototype using Arduino Uno, TMP temperature sensor, Pyroelectric infrared (PIR) motion sensor, and servo motors. Their prototype aims to control the doors and windows by the use of servo motors, and also turn the lights on/off, measuring temperature and electricity by using Arduino Uno and controlling them by utilizing the mobile interface. They created the mobile interface with the help of the MIT App Inventor. The proposed cost of their prototype, including components, is 100 pounds. For controlling remotely, in [29], the author used the Adafruit IO and MQTT Dash mobile Application. They receive the notification on Mobile via IFTTT, and they can also command the appliances using Google Voice Assistant. The used NodeMCU as the main controller as it sends the collected data to the MQTT server and responds to the commands given by the user from the server to the system such as ON/OFF switching. Users can monitor the data on the server. RFID is used to control the door relay. In [30], the author used a free Blynk Application and server. Arduino as the main controller and for communicating wirelessly, ESP8266, and Wi-Fi. The author used the Reed Sensor module to monitor the status and a buzzer for sounding the alarm. In [31], the author designed Home security and monitoring system using Arduino-nano and Node MCU 8266 as main controller. The other peripheries are RFID reader, numerical code to open the door, PIR sensor, DHT22 sensor for temperature and humidity sensing, rain sensor, fire sensor and LDR sensor to monitor the lights. For entering the house user have to punch the RFID card to the card reader, and if the tag is accepted, then the user enters the numerical code. If the code is incorrect, then the buzzer will alarm and sends the email notification to the owner’s email ID. If fire, rain, moving object is detected, the owner will get the notification on the smartphone as per respective installed sensors. Author used Blynk application. In [32], the author implemented a Door Security system using ESP32 as the main controller, PIR sensor to detect the motion near the door, Magnetic sensor, and MQTT cloud as communication protocol between smartphone and door lock system. touch sensor is also Implemented on the door handle to recognize the human hand. If the door is opened by the force than the alarm will ring and sends the notification to the owner. In [33], the author used ESP32 development board and for using camera module, Arducam ESP32 UNO board is used which comes in the shape of standard Arduino UNO R3 board but instead of ATmega328P microcontroller, it has ESP-WROOM-32. Their prototype acquire continues video, display on SPI TFT module and transmit to the receiving end. In [34], the author used ESP32, ESP8266 and CoAP protocol to implement the smart home automation including security and measurement of temperature. CoAP server is used to update the data on cloud and accessing over local network.

Many wireless standards can be used for the Smart Security systems like ZigBee, Bluetooth Low Energy (BLE), LoRa [35], Z wave, Wi-Fi [23], [29], [30], [32], NB-IoT, etc. In [35], the author discussed LoRa technology, which is suitable for low power consumption, long-range, low cost, less delay, and long battery lifetime. The author also discussed the MQTT protocol, and with these, the author got promising results.

Above mentioned literature also includes some literature discussing both home automation and home security. So in **Table 3**, a comparison is presented on the features and used protocols in the literature focused mainly on Home security.

Table 3: Comparison of previously done work

Security systems	**Indoor Control**	**Outdoor control**	**Wireless interface**	**Controller**	**Implementation**	**Smartphone**	**Web-based**	**MQTT**
Tanwar, et al.[23]	✓	-	Wi-Fi	Raspberry Pie	✓	-	-	-
Khanum, et al.[24]	✓	✓	Wi-Fi	Raspberry Pie	✓	-	✓	-
Sai Vineeth, et al.[27]	✓	-	Wi-Fi	Raspberry Pi	✓	-	-	-
Anitha, et al.[30]	✓	✓	Wi-Fi	Arduino	✓	✓	✓	-
Taryudi, et al.[31]	✓	✓	Wi-Fi	Arduino-nano & NodeMCU 8266	✓	✓	✓	-
Andreas, et al.[32]	✓	✓	Wi-Fi	ESP32	✓	✓	✓	✓
Rai et.al.[33]	-	-	Wi-Fi	ESP32 & Arducam ESP32 UNO	✓	-	-	-
Designed Prototype	✓	✓	Wi-Fi	ESP32 & ESP8266	✓	✓	✓	✓

The purpose of this literature review is to view the current trend in the field of Home Security System, mainly about the kind of components, protocols are in use, and the challenges faced while developing the system. It is clear that there is a huge implementation of IoT concepts in this field. Despite being less power consumable devices, still, they (ESP32 & ESP8266) are not used extensively in the development of the IoT products as Home Security System. The main challenges/ problems mentioned by researchers in developing the Home Security System are delay [23], difficult to handle or can say no user friendly [23], [24], [26], [29], non-web empowered [23], [24], high cost [25], [26], [29], [33], limited wireless range [29], [35], more power consumption [35], [36], use of proprietary protocols which are not compatible with other devices [26], in the voice-controlled system, it is possible to mimic the voice [27] and many more. Many researchers did not use the MQTT protocol in their home security design. It is an excellent protocol for transferring information and has many advantages for IoT based applications. So, by observing the above problems and challenges, it is required to develop an Electronic Home Security System for a complete home with the proper utilization of sensors, latest IoT based microcontrollers like ESP32, ESP8266, protocols like MQTT and can resolve the above-stated problems and challenges.

5. Designed System

5.1 Designed Prototype

In this section, block diagram, placement of sensors in the designed prototype and model of the system is presented.

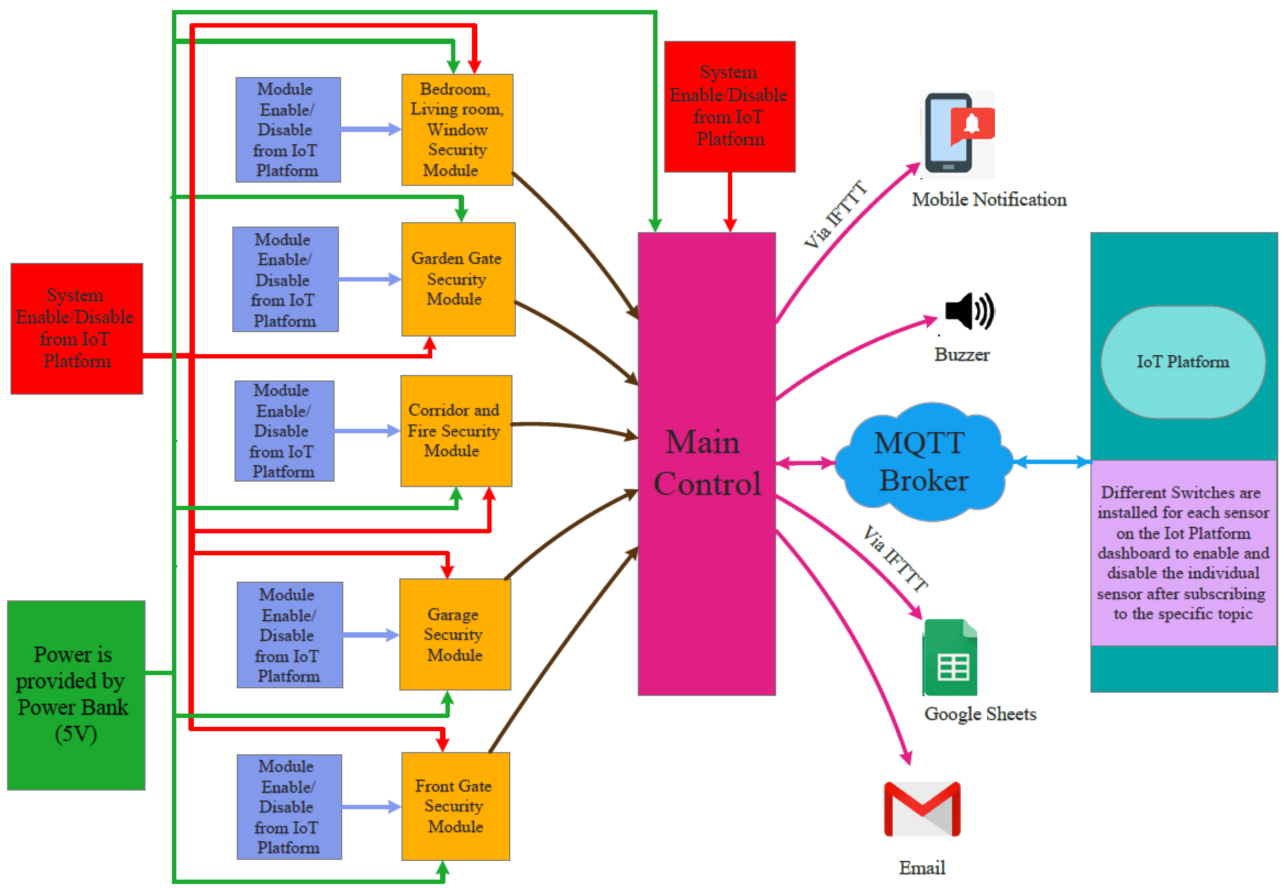


Figure 4: Block Diagram of the designed prototype.

In **Figure 4**, block diagram is shown. Power is provided by Power Bank of 5V. In case of power cut, inverters connected with the power supply will work as a power backup. Main control publishes messages with the different topics to the MQTT broker and each sensor module is subscribed to the specific topic. There is a control switch on IoT platform to enable and disable the whole system and each sensor module after subscription to the specific topic can be enabled or disabled individually. Main control also sends the notification on mobile, email, log data on Google sheets and trigger the buzzer.

Different positions of the sensor in the house is shown in **Figure 5**. The prototype is drawn in the free software **'SWEET HOME 3D'** [37]. This prototype has all the basic portions of the house.

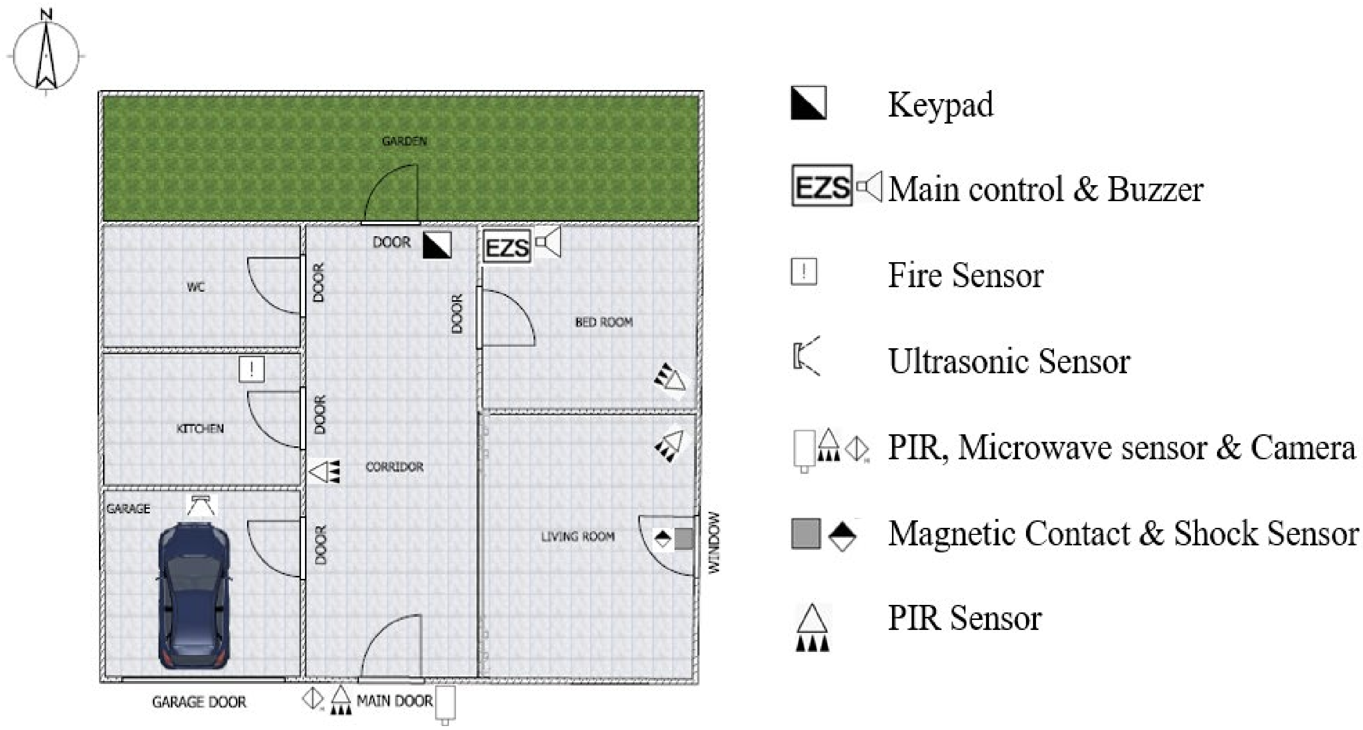


Figure 5: Designed Prototype with symbol description

In the designed system, **Figure 5**, a garage, kitchen, living room, bedroom, washroom, corridor, and a garden is shown. A camera, microwave sensor, and PIR sensor are installed to protect the house from an intruder entering from the main door. A Magnetic switch and Shock sensor (as a glass break sensor) are installed on the window to protect the house from an intruder entering from the window. For the protection of the vehicle parked in the garage, an Ultrasonic sensor is installed. A Fire sensor is installed in the Kitchen to give the information to the owner in case of Fire. A PIR sensor is installed on the ceiling of the corridor for the corridor protection. A keypad is installed to protect the garden from unauthorized entry. Buzzer and Main control is installed in the Bedroom. In each Bedroom and Living room, the PIR sensor is installed.

Figure 6 and **Figure 7** shows the designed model of the system.

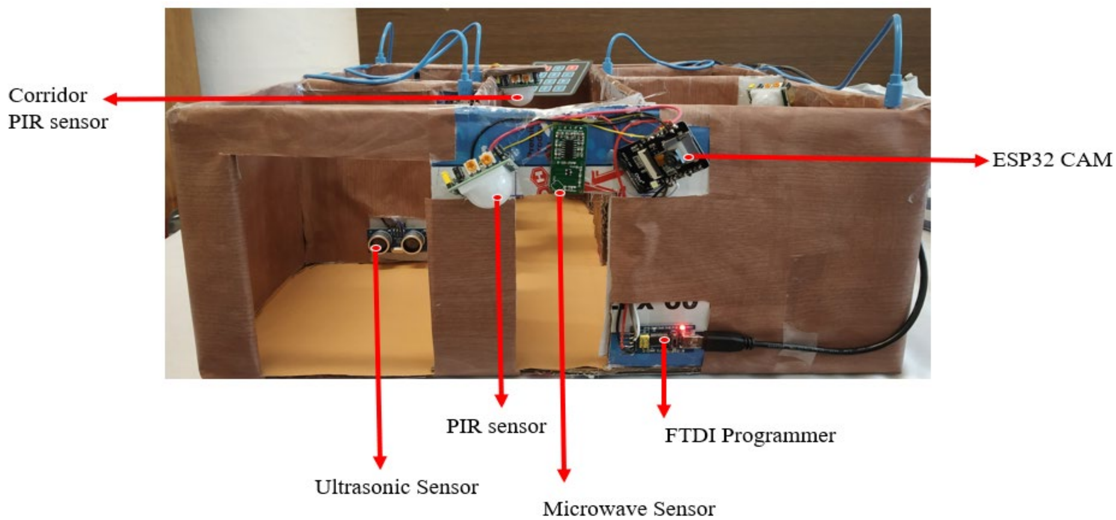


Figure 6: Front view of the designed model

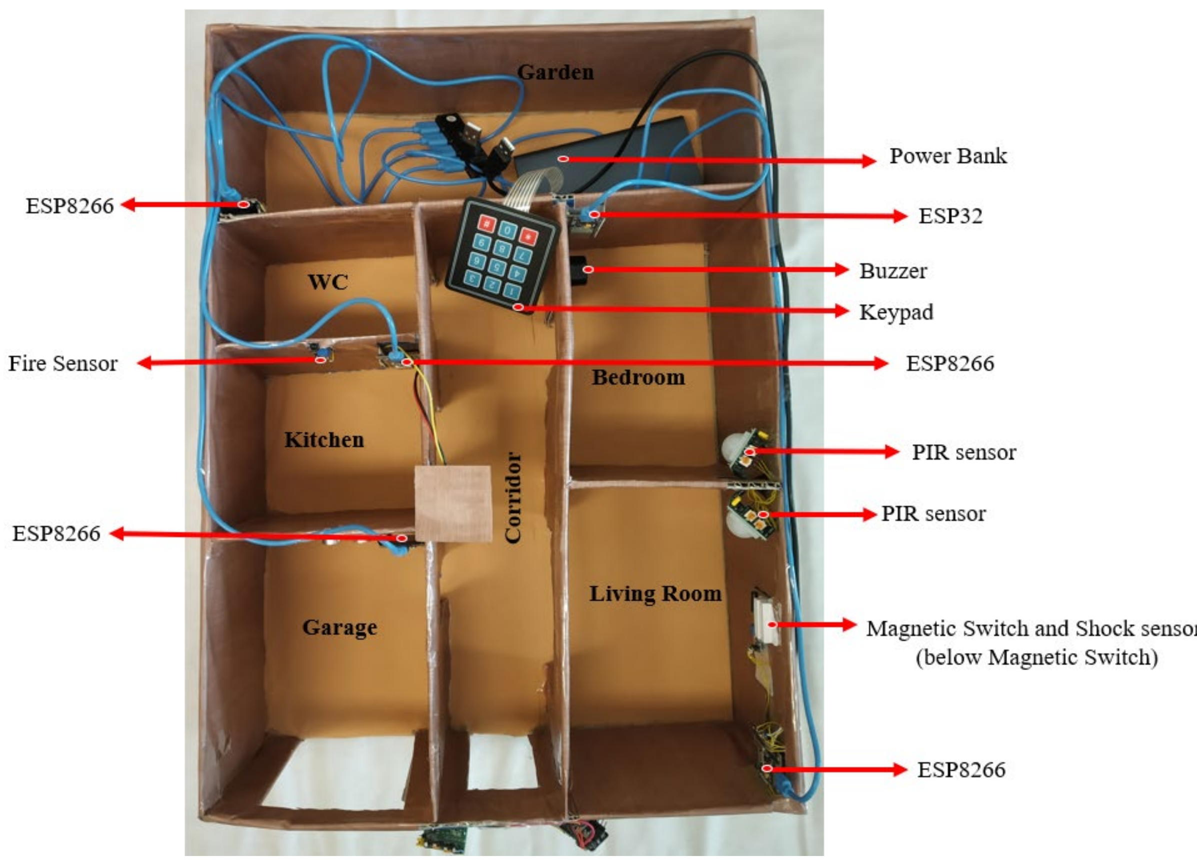


Figure 7: Top view of the designed model

5.2 Components

5.2.1 ESP32

ESP32 is the successor of the ESP8266 integrated circuit from a Chinese company Espressif Systems, which at the time of its launch, experienced enormous a rapid increase in popularity. ESP32 was launched in 2016. ESP32 is a single 2.4 GHz Wi-Fi and Bluetooth combo chip. It is a low cost, low power system on a chip (SoC) series. It has Ultra low power co-processor, which allows doing ADC conversions, computation, and level thresholds while in a deep sleep. It is designed to achieve the best power and RF performance, robustness, versatility, and reliability in a wide variety of applications and different power profiles. At its heart, there is a dual-core or single-core Tensilica Xtensa LX6 microprocessor with a clock rate of up to 240 MHz. It is integrated with antenna switch, RF balun, power amplifier, low noise receive amplifier, filters, and power management modules. Its functional block diagram is shown in **Figure 8**. ESP-IDF is the official development framework for the ESP 32. But it is not simple to program it for the beginner. So suitable IDE for the beginners is Arduino IDE despite its limited functionality. Arduino IDE is supported by the user donated quality libraries and have an extensive online community for help. [38]
Pin description and specifications can be referred from Appendix section 11.1.1.1.

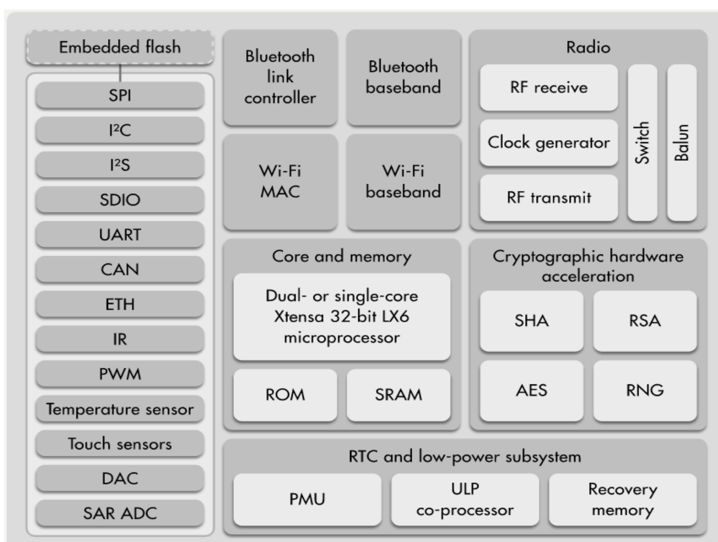


Figure 8: ESP32 Functional Block Diagram [38]

5.2.2 ESP8266

ESP8266 is the WiFi System on Chip (SoC) produced by Espressif System Company. It was launched in 2014. It consists of a Tensilica L106 32-bit microcontroller unit (MCU) and a Wi-Fi transceiver. It can

perform either as a standalone application or as a slave to a host MCU. Also, ESP8266 can be applied to any microcontroller design as a Wi-Fi adaptor through SPI/SDIO or UART interfaces. It integrates antenna switches, RF balun, power amplifier, low noise receive amplifier, filters, and power management modules [39]. Its functional block diagram is shown in **Figure 9**. It can be programmed in Lua programming language, Arduino IDE, and Micro Python. In this project, the Arduino IDE is used. [39]
Pin description and specifications can be referred from Appendix section 11.1.1.2.

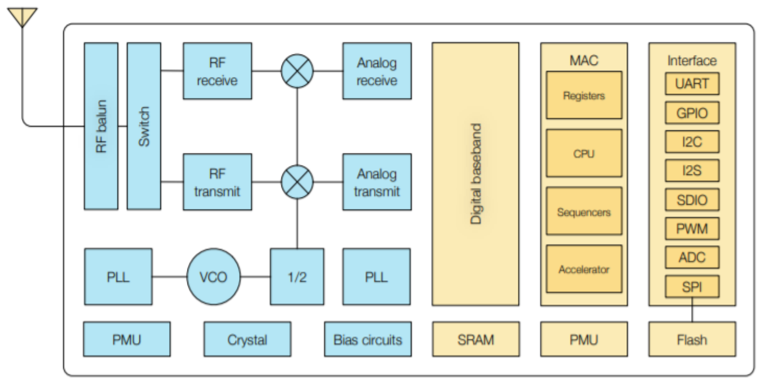


Figure 9: ESP8266EX Functional Block Diagram [39]

5.2.3 ESP32CAM Module

ESP32CAM is based on an ESP32 module, which has Wi-Fi and Bluetooth, with an additional 4MB of external RAM. It is produced by the AI Thinker company. Also, on the board is a memory card slot and a camera connector that can take an OV2640 or OV7670 camera module. [40]
Pin description and specifications can be referred from Appendix section 11.1.1.3.

5.2.4 Fire Sensor

The flame sensor can detect flame and infrared (IR) light sources with wavelengths ranging from 760 nm to 1100 nm. It uses the LM393 comparator chip. Flame sensors can be used in fire alarms and other fire detecting devices. [41]
Pin description and specifications can be referred from Appendix section 11.1.1.4.

5.2.5 PIR Sensor

PIR sensors (HC-SR501) is used to sense motion when a human or any animal has moved in or out of the sensors range. It is referred to as Passive Infrared sensor. It has two slots in it, and each slot is sensitive to IR. When the sensor is idle state, both slots detect the same amount of IR radiation. When a human or animal passes by, it first intercepts one half of the PIR sensor, which causes a positive differential change between the two halves. When the warm body leaves the sensing area, the reverse happens, whereby the sensor generates a negative differential change. [42]

These change pulses are detected, as shown in **Figure 10**. Pin description and specifications can be referred from Appendix section 11.1.1.5.

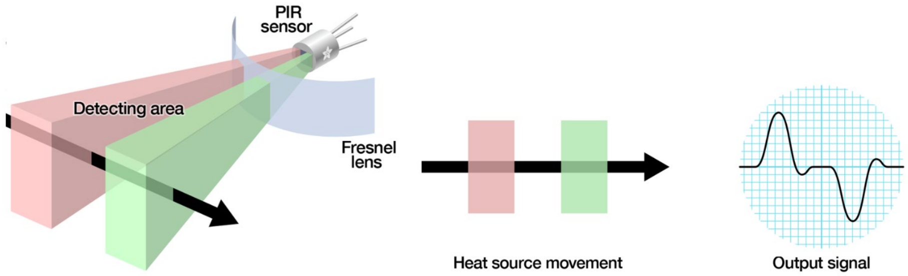


Figure 10: Functioning of the PIR sensor [42]

5.2.6 Magnetic Switch

A typical magnetic switch has 2 parts one is the magnet, and the other is the magnetically sensitive part (usually a reed switch enclosed within a glass envelope). Switches may be either normally open (close on alarm) or normally closed (open on alarm). **Figure 11** shows the working of the Magnetic Switch. When the magnet is away, two contacts of the reed switch is away from each other. When the magnet is near, two contacts of the reed switch get attracted to the magnet, and they come in contact with each other. [43]

Pin description and specifications can be referred from Appendix section 11.1.1.6.

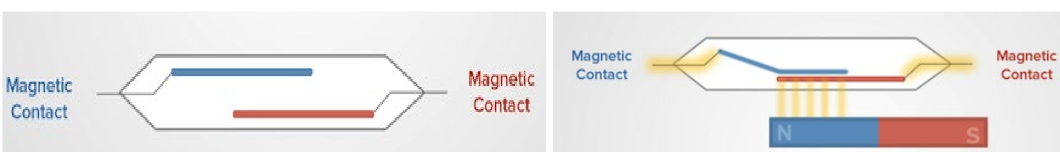


Figure 11: When Magnet is near *(left),* When Magnet is away *(right)* [44]

5.2.7 Ultrasonic Sensor

It consists of both receiver and transmitter. The time taken by ultrasonic sound to send and receive the signal back tell about the distance of the target. It is a non-contact technology that gives results with good accuracy. They provide distance measurement and wider angle of detection. The transmitter (trig pin) of the sensor sends a high-frequency sound signal. When those sound signal finds any object, signals get reflected and Receiver (echo pin) receives that signal. Pin description and specifications can be referred from Appendix section 11.1.1.7.

5.2.8 Microwave Sensor

Actual name of this sensor is Microwave Doppler Radar RCWL-0516. It is a small proximity sensor which uses the Doppler effect to detect the presence of intruder. It detects the movement in the detection range. Doppler effect is the change in the frequency of a wave in when observer and wave source have relative motion between each other. RCWL-0516 supports 360-degree detection area with no blind spot. Best sensitivity is component side. A light dependent resistor (LDR) can be attached to the CDS pad on the sensor PCB or to the CDS pin. Pin description and specifications can be referred from Appendix section 11.1.1.8

5.2.9 Shock Sensor

It detects shocks, impact, or vibration. It has high sensitivity vibration switch (SW-18010P). The sensor consists of a terminal that forms a center post and a second terminal that is a spring, surrounding the center post. If a sudden impact with sufficient force is applied to the sensor, the terminal consisting of the spring moves and shorts both terminals together [45]. Specifications can be referred from Appendix section 11.1.1.9.

5.2.10 Buzzer

Buzzer (Piezo Electric Buzzer) is used just to get the sound indication when any event happens. Specifications can be referred from Appendix section 11.1.1.10.

5.2.11 Servo Motor

Servo motors are used to set a certain position of the actuated mechanism and then hold it in that position. Their main advantage is their small size and low weight with relatively high power. These motors usually do not allow rotation over and over but maintain a set angle of rotation. The angle is usually in the range of 0° to 180°. [46]
Specifications can be referred from Appendix section 11.1.1.11.

5.2.12 Keypad

The keyboard is suitable for all applications where there is a requirement to enter a numeric or other input. The used keypad is 3x4 consisting of 4 rows and 3 columns. Pressing any key shorts, the corresponding row, and column allowing the current to flow. For example, if key 5 is pressed, it will short the second row and second column, as shown in **Figure 12**. Specifications can be referred from Appendix section 11.1.1.12.

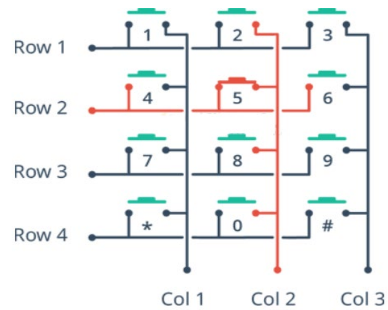


Figure 12: Working of Keypad [47]

5.3 Technology Used

5.3.1 Wi-Fi

General idea about the Wi-Fi is already explained in section **3.1.1** on Page number **6,** but in this section, it is explained how it is used in the designed system. ESP32 and ESP8266 support Station only mode (Wi-Fi client mode), AP only mode (Soft-AP mode or Access Point mode), and Station-AP combined mode [39], [48]. In station mode, ESPs connect to Access point like a wireless router or local network, and they (ESPs) are set as a station. In AP mode, ESPs create their own Wi-Fi network, and to this Wi-Fi network, other devices can connect to it in the station mode without the need of a router. In combined AP-Station mode, ESPs can connect to other Access Point while running their own network.

In the designed system, both ESP8266, ESP32 CAM and ESP32 (main controller) are connected to the mobile hotspot. The mobile hot spot is acting as a Wi-Fi Access point, and ESP32, ESP8266 and ESP32 CAM are in Station mode, as shown in **Figure 13**. So, ESP32, ESP32 CAM and ESP8266 can update the data on the IoT platform. But as per the basic elements of the Electronic Security System, there should be a decision-making unit (main controller), which can control all the functions. For this purpose, ESP32 is programmed as the main controller (so as the Thesis title), and all the sensors with ESP8266 are connected with ESP32 via its IP address. And as per the sensed data, they (ESP 32 CAM & ESP8266 with sensors) send the command to ESP32, which further completes its functions such as sending the emails, notifications, turning ON the buzzer, and updating the data on the IoT platform as shown in **Figure 14**.

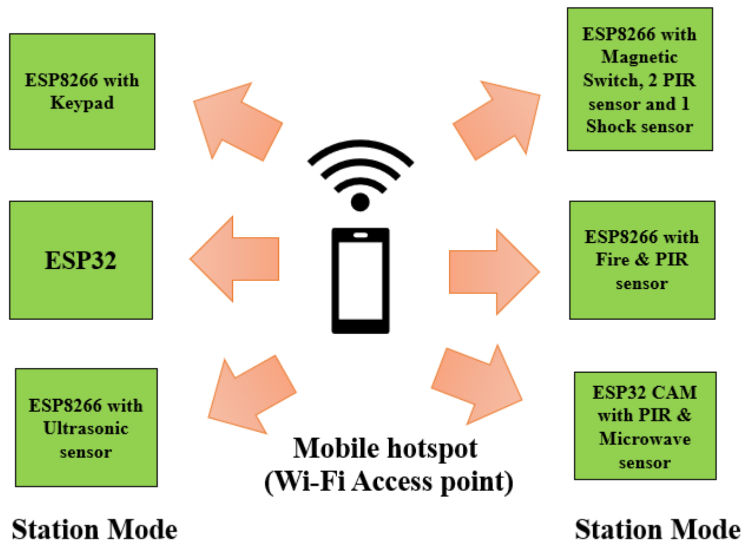


Figure 13: Showing Access point and Station mode

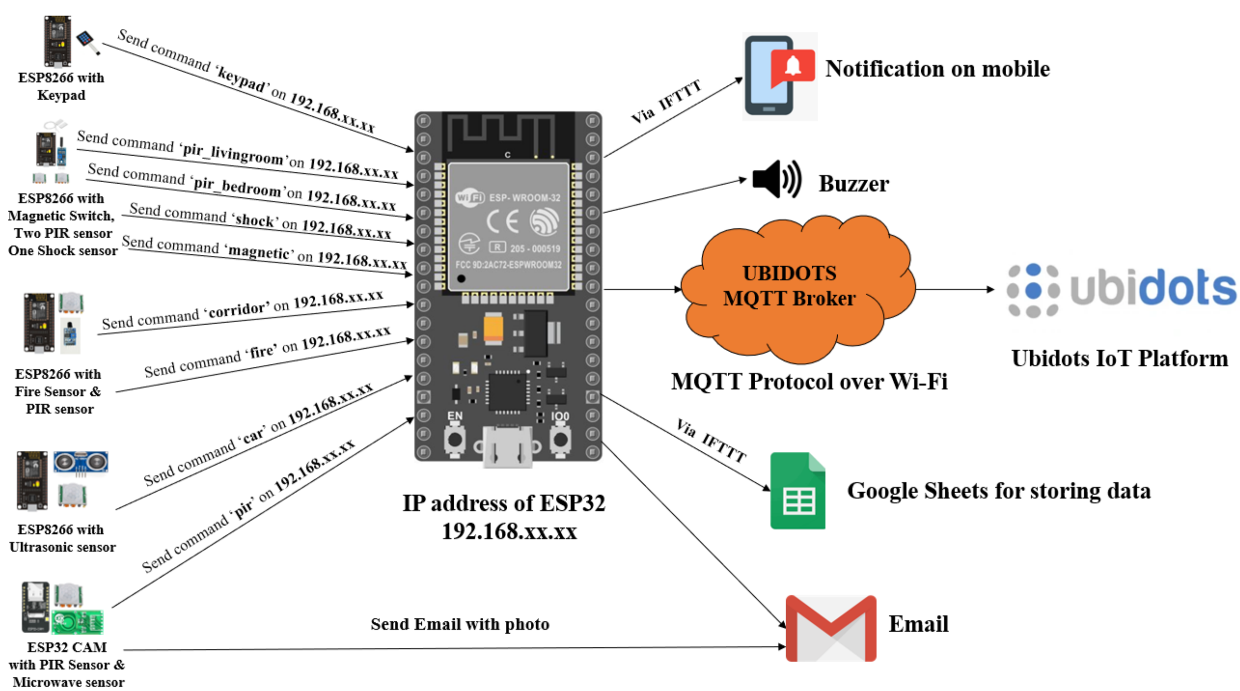


Figure 14: Flow of commands in the designed system

5.3.2 MQTT (Message Queue Telemetry Transport)

It is a simple and lightweight messaging protocol designed for constrained devices and low-bandwidth, high-latency, or unreliable networks [49]. This protocol is based on the principle of publishing messages and subscribing to the topics. Clients connect to the broker and publish the messages to the topic. Publishers are the source of data. And the clients who are interested in the same topic subscribe to it. A single subscriber can subscribe to more than one topic. Topics are treated as a hierarchy, using a slash (/) as a separator, much in the same way as a filesystem [50].

For example, in **Figure 15**, there are 3 sensors, and they are considered as a publisher. They send their data to the broker with different topics, i.e., Temp, Humidity, Motion, respectively (These topics are not structured in the slash (/) format, as they are just for demo purpose). MQTT Broker sends the value to the subscriber as per their subscribed topic. Subscriber 3 is subscribed to more than more topics. This protocol is ideal for emerging “machine-to-machine” (M2M) or “Internet of Things” world of connected devices [49].

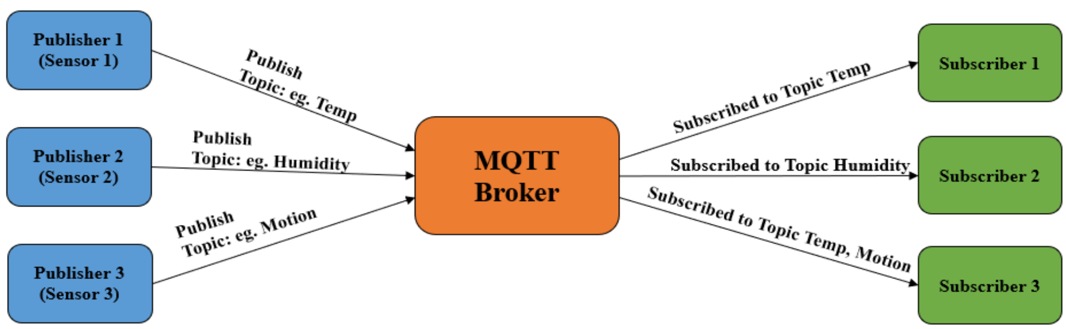


Figure 15: MQTT Example

Table 4: Table showing Publish device, Topic, and Subscribe device for MQTT Example.

Publish Device	Topic	Subscribe Device
Publisher 1	Temp	Subscriber 1 Subscriber 3
Publisher 2	Humidity	Subscriber 2
Publisher 3	Motion	Subscriber 3

In the designed system, Ubidots MQTT Broker is used, and the MQTT Publish and subscribe flow is shown in **Figure 16**. Ubidots MQTT Broker is available at the **industrial.api.ubidots.com** and port is 1883 [51]. To interact with it, TOKEN is needed that can be obtained from ‘API credentials’ on the Ubidots account page. MQTT username is the TOKEN, and the password is kept empty. After successful connection, we can publish and subscribe to the MQTT topics.

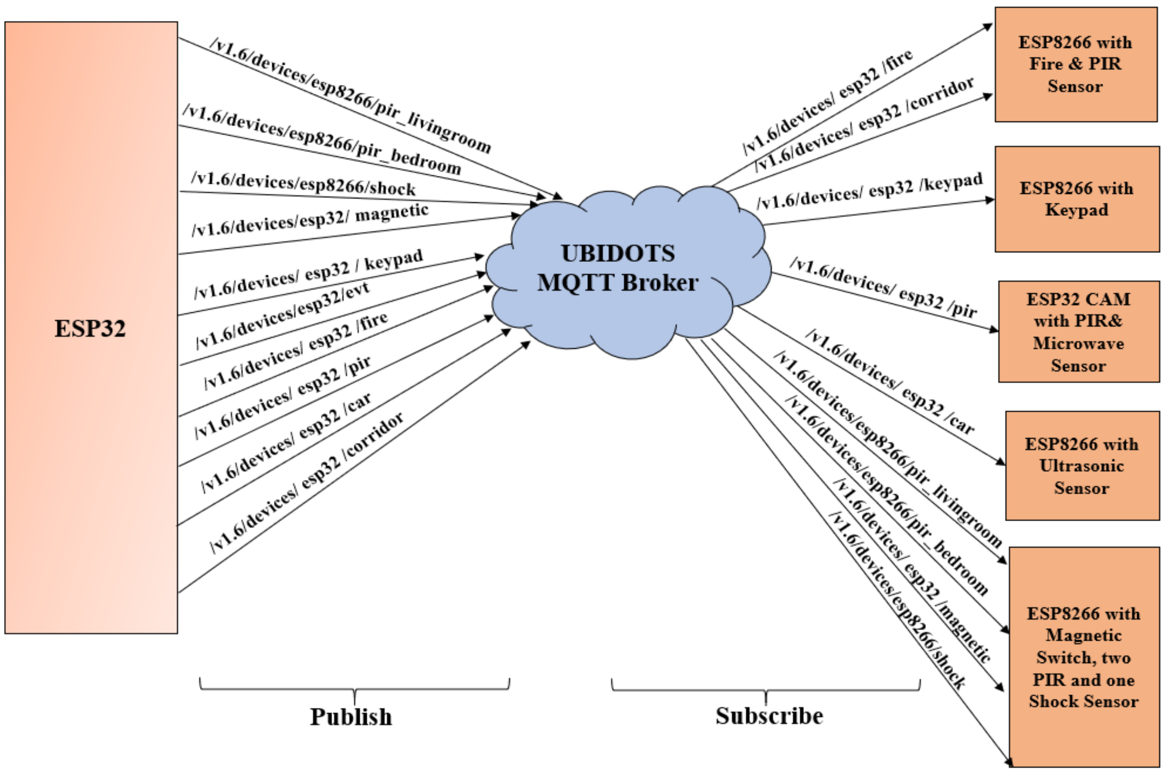


Figure 16: MQTT Flow for Designed System

Table 5: MQTT Publish and subscribe for Designed System

Publish Device	Topic	Subscribe Device
ESP32	/v1.6/devices/ esp32 /magnetic /v1.6/devices/esp8266/pir_livingroom /v1.6/devices/esp8266/pir_bedroom /v1.6/devices/esp8266/shock	ESP8266 with Magnetic Switch, two PIR sensor and one Shock Sensor
	/v1.6/devices/ esp32 /keypad	ESP8266 with Keypad
	/v1.6/devices/ esp32 /fire /v1.6/devices/ esp32 /corridor	ESP8266 with Fire & PIR sensor
	/v1.6/devices/ esp32 /car	ESP8266 with Ultrasonic sensor
	/v1.6/devices/ esp32 /pir	ESP32 CAM with PIR & Microwave sensor
	/v1.6/devices/esp32/evt	It is only for displaying the data on IoT Platform

Topic format of Ubidots MQTT Broker is **/v1.6/devices/{LABEL_OF_DEVICE}/{LABEL_OF_VARIABLE}** [51]. For example, the topic of the data received from Magnetic Switch will be **/v1.6/devices/ esp32 /magnetic**. **LABEL_OF_DEVICE** & **LABEL_OF_VARIABLE** is up to the user means they can keep any name.

In **Figure 16** and **Table 5**, it is shown that ESP32 is the Publishing device with different topics to the Ubidots MQTT Broker. And different sensors with ESP8266 and ESP32 CAM subscribe to a specific topic. It is also seen in **Figure 16** & **Table 5** that the topic (/v1.6/devices/esp32/evt) is not subscribed by any device. It is only published to the broker to show all the published messages in the single column as shown in **Figure 17**.

Values Table

DATE	EVT (ESP32) CONTEXT DATA
04/24/2020 14:14	Window: Intrusion detected
04/24/2020 14:11	Window: Intrusion detected
04/24/2020 14:10	Window: Intrusion detected
04/24/2020 13:57	Window: Intrusion detected
04/24/2020 13:54	Window: Intrusion detected
04/24/2020 13:50	Window: Intrusion detected
04/24/2020 13:49	Window: Intrusion detected
04/24/2020 13:47	Window: Intrusion detected
04/22/2020 18:04	Window: Intrusion detected

Figure 17: All the data in a single column on Ubidots IoT Platform

For testing Publish and Subscribe, MQTT.FX [52] is used. We can test by using any of the components from the designed system and try to publish and subscribe to the topic. So, the magnetic switch of the window is taken with the topic **/v1.6/devices/ esp32 /magnetic.** So, when we trigger the magnetic switch, the message should arrive on both Ubidots and MQTT.FX with value '1' and Content 'magnetic: Window Intrusion Detected,' shown in **Figure 18** (arrival time can be matched, i.e., 01:24).

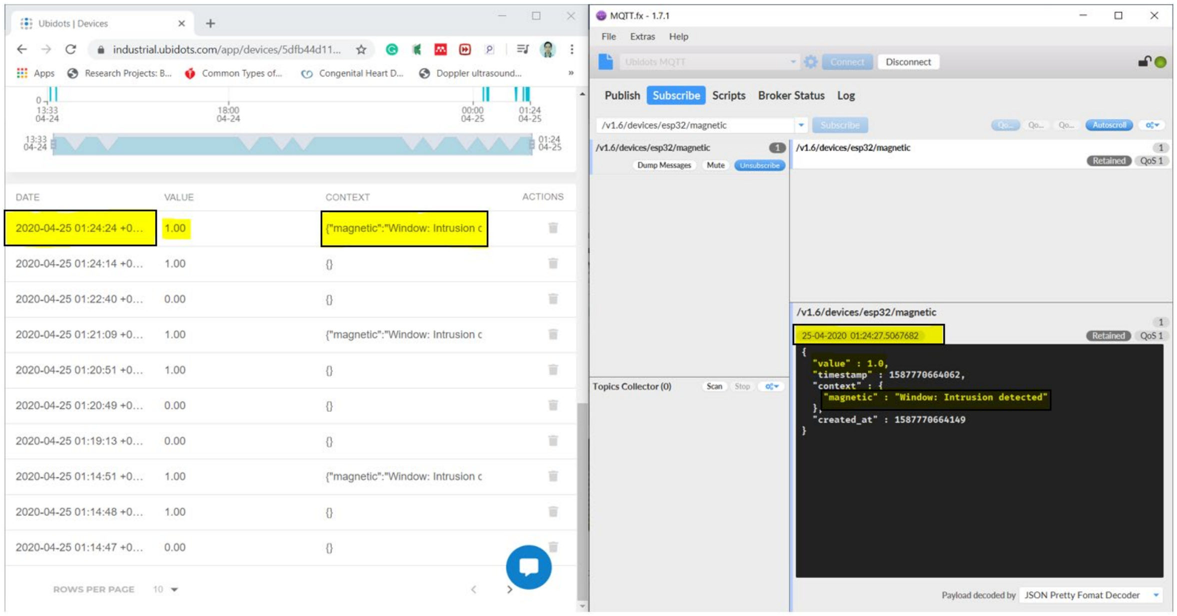


Figure 18: Subscribed to Topic and publishing to Ubidots

5.3.2.1 Quality of Service (QoS)

One of the best features of the MQTT is the Quality of Service (QoS). It means the reliability or the guarantee level of the message delivery between the sender and receiver. There are 2 QoS, i.e., QoS-0, QoS-1, and QoS-2 [50].

1. **Quality of Service (QoS)-0:** At most once. It means that the broker does not acknowledge whether the message received or not.
2. **Quality of Service (QoS)-1:** At least once. It means that the broker acknowledges whether the message received or not. The client can send the message multiple times until it receives the acknowledgment.
3. **Quality of Service (QoS)-2:** Exactly once or assured delivery. It is the highest level of Quality of Service. Despite being assured delivery, it is slower than QoS-0 and QoS-1. It involves the multiple flows between client and broker for the single message, but there is message loss.

5.3.3 Dual Technology

As discussed in section **2.2,** reliability is an important factor for the Electronic Security System. To enhance reliability, dual technology can be used. A dual-technology sensor combines two basic sensing technologies in one unit to enhance performance [53]. For example, a PIR sensor can be combined with a Microwave sensor and Ultrasonic sensor to enhance the reliability in the form of prevention against the false triggering. In a garage, if the owner visit for just taking some items, and only a PIR sensor is installed, then it will

detect the motion and trigger the alarm. It is the case of false triggering. In the Front door, the PIR sensor can detect the temperature change, e.g., from sunshine, which can trigger the alarm.
So in the designed system, this concept is used. On the Front door, the PIR sensor and Microwave sensor are combined to prevent false triggering and by which reliability is enhanced. And the alarm will trigger only if both PIR and Microwave sensors detect the change.

5.3.4 Ubidots

Ubidots is a cloud service (IoT platform) that offers a friendly and intuitive interface to store and analyze sensor data in real-time. It provides device friendly APIs accessed over HTTP/MQTT/TCP/UDP protocols due to which it is easy to set secure connection for sending and retrieving data to/ from cloud service to real-time, display sensor data through widgets, trigger alerts in the form of Email and SMS when a sensor data hits a value [54]. It also has its own mobile application by which we can view the real-time data on mobile without the need of logging to a laptop or computer. And as per the survey conducted by the hardware community Hackster in 2016, Ubidots is in the top 10 list of the preferred IoT platform [55]. We can add a couple of widgets like an indicator, switch, graph, gauge, slider, and many more widgets to create an interactive dashboard. **Figure 19** shows the dashboard of the designed system consisting of Switches and Value table. In the designed system, Ubidots is used as MQTT Broker, an IoT platform for data visualization, which can also be done on its mobile application.

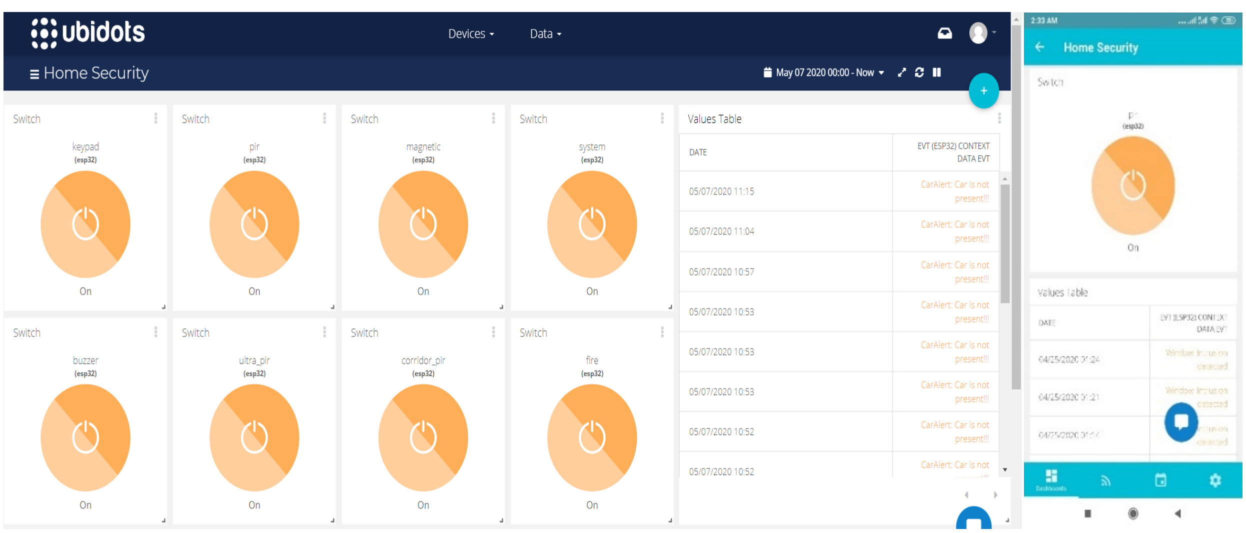


Figure 19: Dashboard and Mobile application of the designed system

Ubidots STEM account is used in the designed system, as it is free for the personal education, IoT research, or DIY projects up to the certain limits that are shown below [56]:

1. A shared server and computational resources across all STEM users. Speed and reliability based on total platform requests at any given second.
2. Non-commercial use ONLY (personal education, IoT research, or DIY projects)
3. 1 dashboard (static only)
4. Up to 3 devices, each device can have up to 10 variables
5. 1 month of data retention
6. 4,000 dots per day (Every time a device updates a sensor value in a variable, a data-point or "dot" is created. Ubidots stores dots that come from devices inside variables, and these stored dots have corresponding timestamps)
7. Up to 30 dots per minute across all of your devices
8. 10 Free SMS and 1 voice call (US & Canada) to kick off your projects. We can add balance to our account anytime to send more alerts, including international SMS and voice calls.
9. 100 Email or Telegram alerts.
10. Maximum 10 widgets per dashboard
11. REST API and MQTT Support
12. SMS, Voice call, Email, Webhook, Slack & Telegram Alerts

5.3.5 Arduino IDE

Arduino Integrated Development Environment is one of the famous IDE for developers and beginners as it is an open-source, simple, and clear programming environment. Arduino language is just a set of C/C++ functions, and after minor changes in codes, they are passed to the C/C++ compiler [57]. It contains a text editor, a message area, a text console, a toolbar. The editor has the facility of cutting, pasting, searching, and replacing the text. Many libraries are available for programming the ESP8266 and ESP32. In the designed system, the Arduino IDE is used for writing the codes.

5.3.6 IFTTT (If This Then That)

It derives its name from the conditional programming statement “if this, then that.” It is a Web service that helps to connect other Web services. It combines two Web-based services using a concept called recipes, which accomplish a task [58]. It allows its end-user to create, customize, and enable a chain of conditional statements. First, make an account in the IFTTT, then we have to select the trigger channel like Instagram and then select the action channel like Dropbox. It can work as saving the liked photos on Instagram in Dropbox. In the designed system, when any sensor is triggered, IFTTT sends the notification to the owner’s mobile and stores the data on Google Sheets. For getting notification, its mobile application is required to be installed on the mobile.

6. Working

To communicate with ESP32, IP address of the ESP32 module is added in all the ESP8266 and ESP32 CAM modules code. So that when any sensor is triggered, then they (ESP8266s and ESP32 CAM) connect to the IP address of ESP32 and then send the command.

For the security of full house, the system is divided into 6 modules which are as follows:

1) For Controlling all the parts, ESP32 is working as a Main controller and connected with Buzzer.
2) ESP8266 with Magnetic Switch and Shock Sensor for Window Security, Bedroom PIR Sensor and Living Room PIR Sensor
3) For protection against Fire and for the security of Corridor, Fire Sensor and PIR sensor is connected with ESP8266
4) For Vehicle theft Security, Ultrasonic Sensor is connected with ESP8266
5) For Garden gate Security, 4*3 Matrix Keypad is connected with ESP8266
6) For Front Gate Security, Microwave and PIR sensor is connected with ESP32 CAM

All these modules can be disarmed individually and can be disarmed collectively from the main controller as per the requirement from the Ubidots IoT platform.

Working of all the individual module are described in the following section:

6.1 Main Controller (ESP32)

ESP32 is connected with a common buzzer that will trigger when any command will arrive from the different modules. If any command arrives from the modules, Buzzer will continue to blow the sound until we switch OFF from the Ubidots IoT Platform. The connection of buzzer with ESP32 is shown in **Figure 20**.

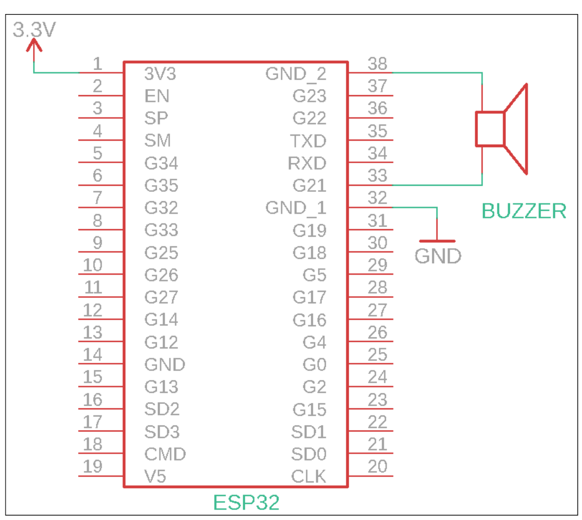


Figure 20: Connection of ESP32 with buzzer

Figure 21 shows the flow chart of the ESP32 (main controller). If the system is enabled and any security module is triggered, then it will send the command to ESP32.

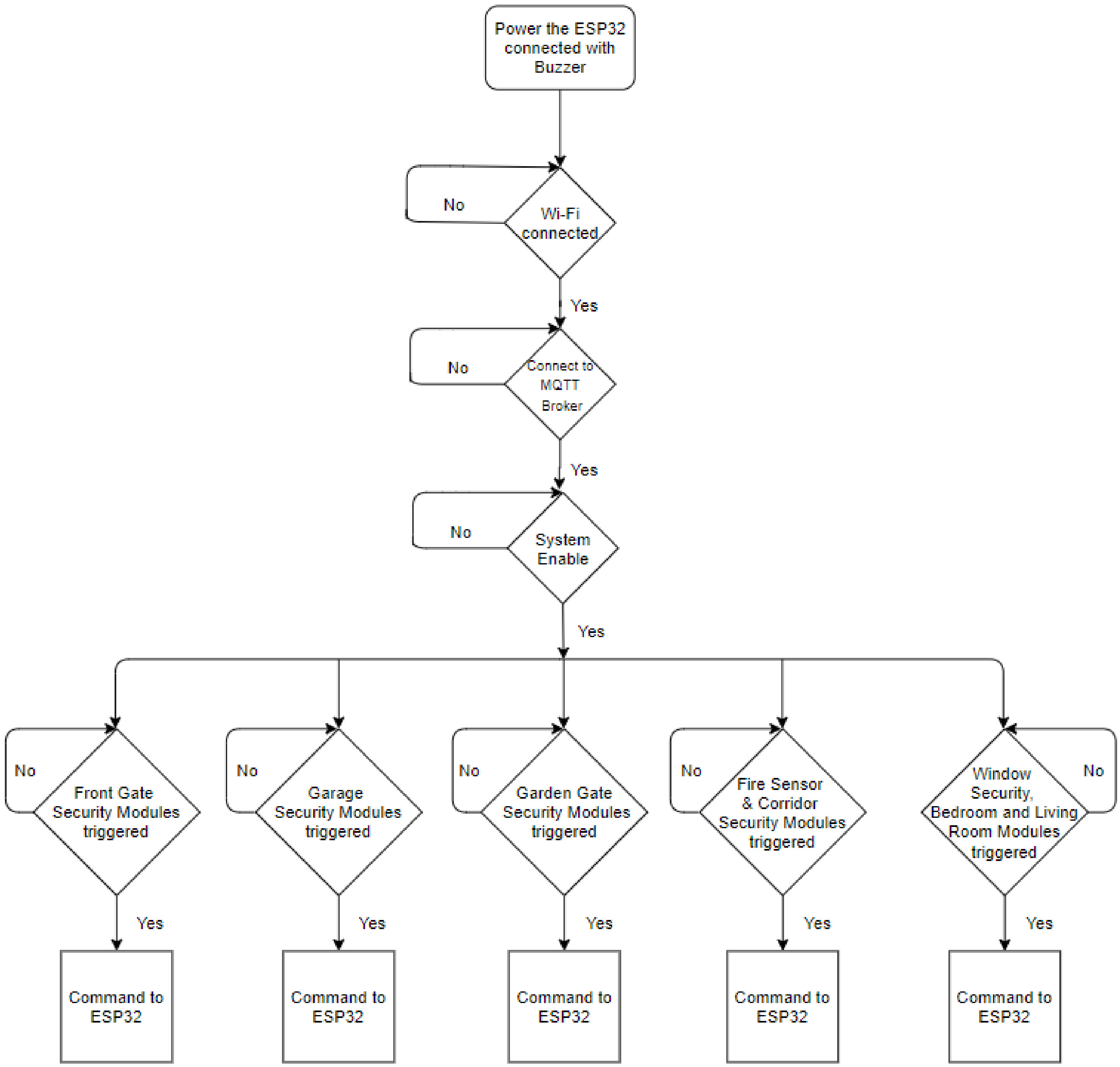


Figure 21: Flow chart of main controller (ESP32)

6.2 ESP8266 with Magnetic Switch and Shock Sensor for Window Security, Bedroom PIR Sensor and Living Room PIR Sensor

The magnetic switch, shock sensor is installed to protect the home from intrusion from the window. If intruder tries to enter in the house by opening the window, then Magnetic switch will detect the intrusion. If intruder tried to break the glass than shock sensor will detect the shock/vibration. With the same ESP8266 two PIR sensors are also connected. One for Living room motion detection and other for bedroom motion detection. The circuit diagram is shown in **Figure 22**.

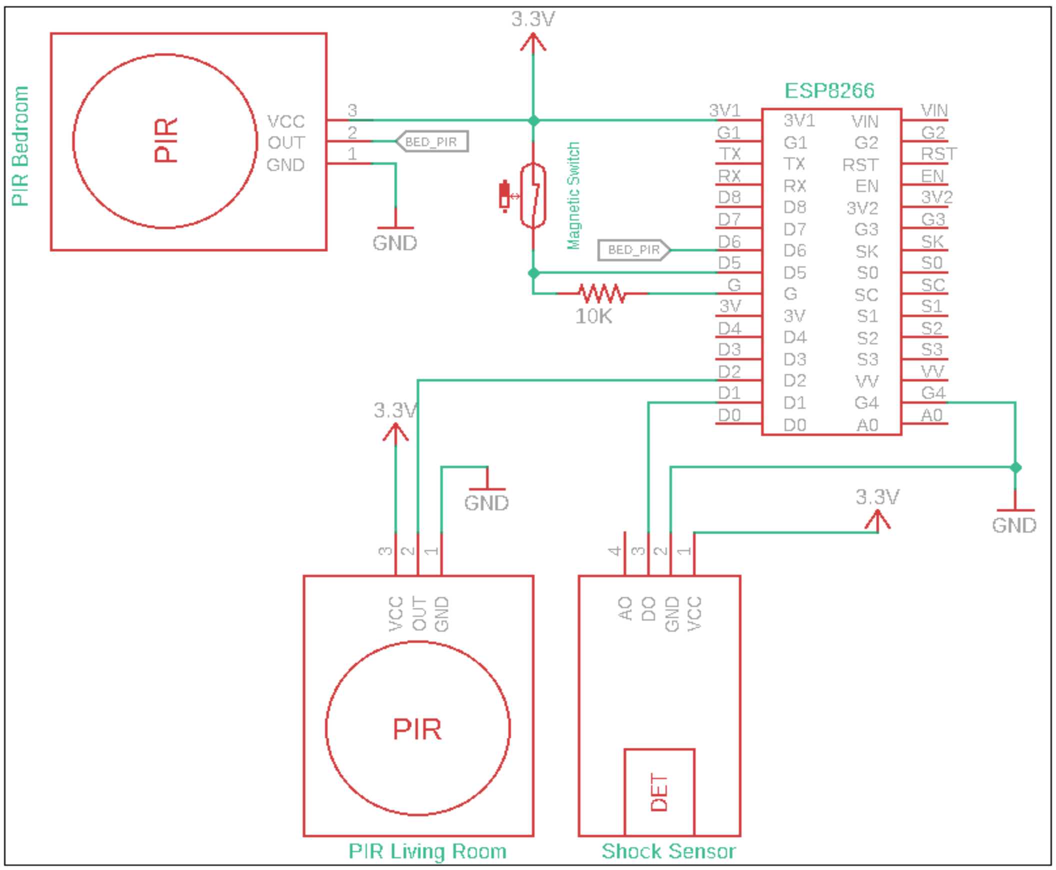


Figure 22: Connection of Magnetic Switch, Shock sensor, and two PIR sensors with ESP8266

The complete flow is shown in **Figure 23.** It shows that for informing the main controller about the intrusion, each sensor should be enabled with System enable. All the four sensors do not interfere with each other's working.

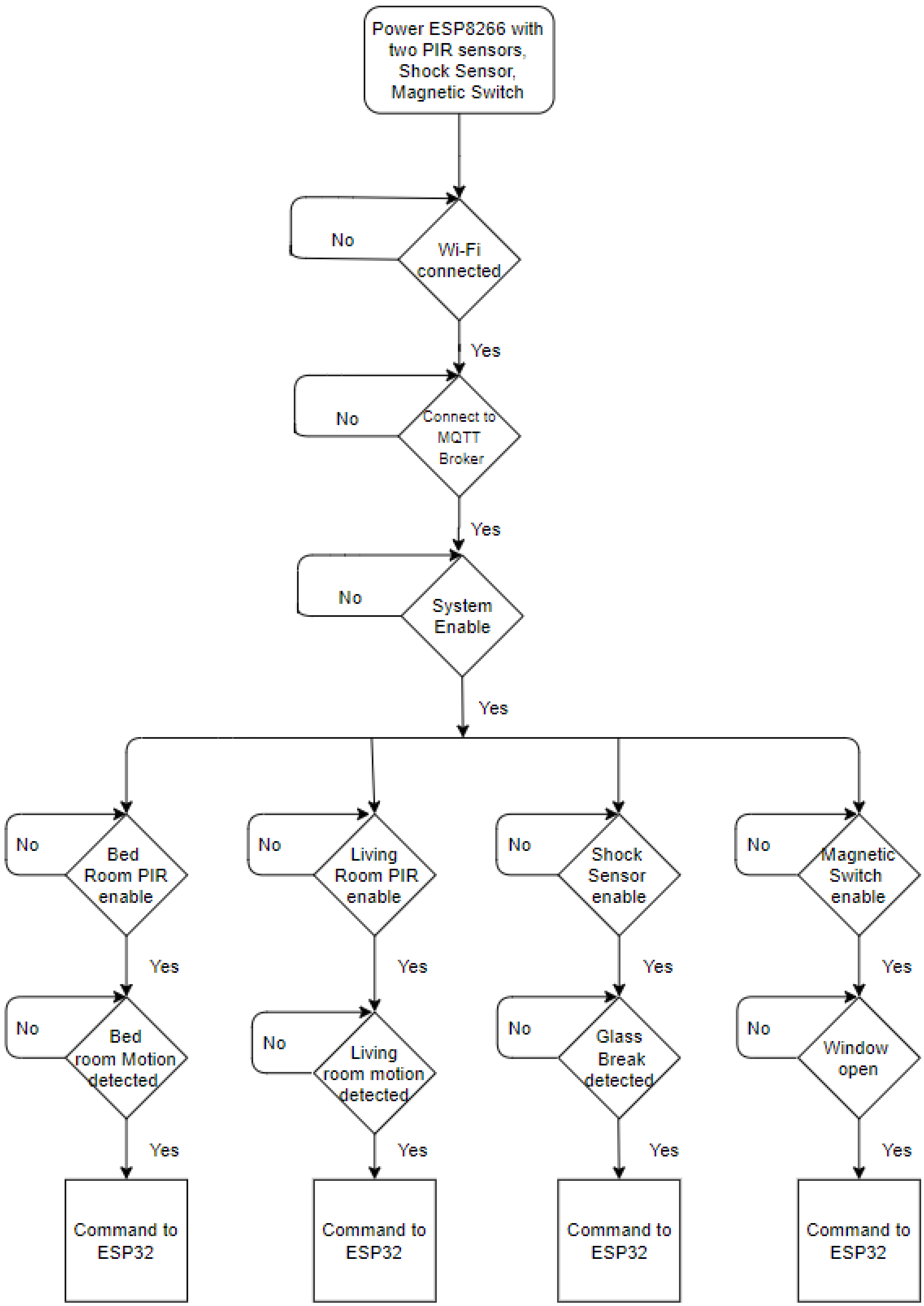


Figure 23: Flow chart of Magnetic Switch, Shock sensor, and two PIR sensors

6.2.1 Magnetic Switch

Table 6 shows the possibilities and effects of System enable/disable, and Magnetic Switch enable/disable on full house security.

Table 6: Control of Magnetic Switch (When Magnetic Switch triggered)

System enable	Magnetic Switch enable	ESP8266 with magnetic switch	ESP32 (Main Controller)	Security of Window	Full house Security	Refer
Yes	Yes	Detect change and send command to ESP32	Received command from Magnetic Switch	ON	ON	Figure 24, Figure 25, Figure 26
Yes	No	Didn't detect any change	Didn't received command	OFF	ON	Appendix 11.2.1.1.1
No	Yes	Didn't detect any change	Didn't received command	OFF	OFF	Appendix 11.2.1.1.2
No	No	Didn't detect any change	Didn't received command	OFF	OFF	Appendix 11.2.1.1.3

In **Figure 24**, the magnetic button and system button is in ON state.

Labelled 1: It is the serial monitor of ESP32, it shows the new client request as magnetic, sending the notification, sending data to Ubidots IoT platform, Switch on the buzzer, sending data on Google sheets. And Buzzer is switched OFF by button (buzzer) on Ubidots.

Labelled 2: It is the serial monitor of ESP8266 with a Magnetic Switch. It shows the connection to ESP32 and then sending the command to ESP32.

Labelled 3: It is the interface of Ubidots, showing the data. And highlighted part verifies the data that arrived at the same time as the magnetic switch triggered.

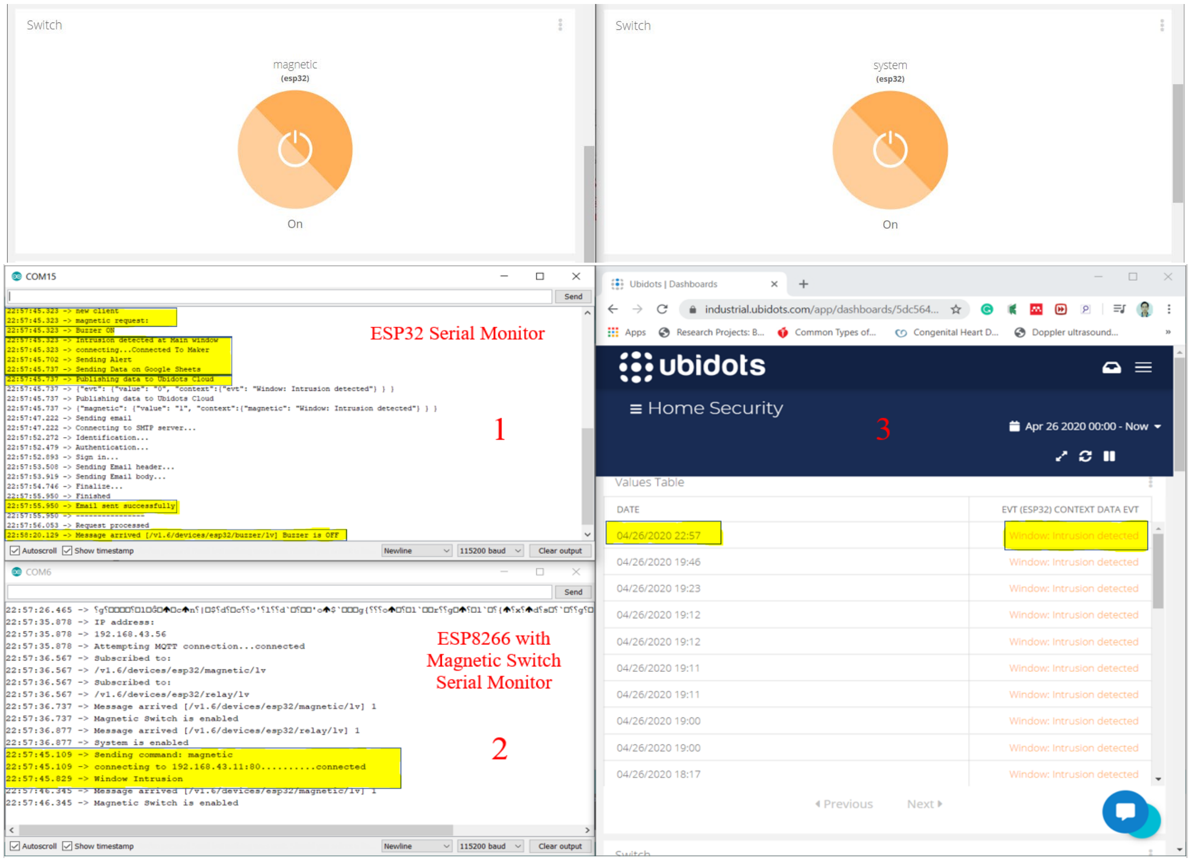


Figure 24: When magnetic button and system button is in ON state

Figure 25 shows the data logged on Google sheet and notification on mobile via IFTTT. And highlighted part verifies the data that arrived at the same time as the magnetic switch triggered.

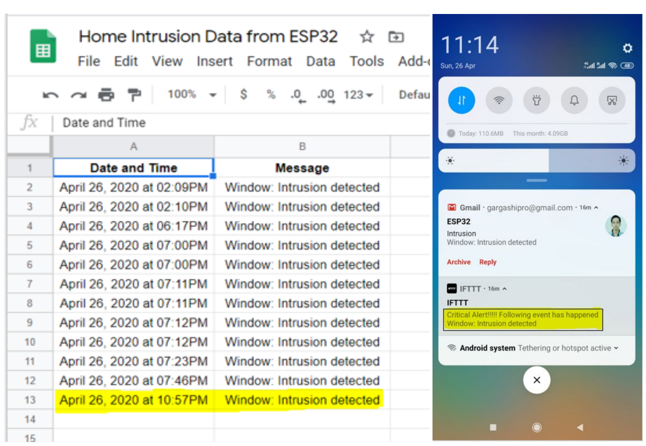


Figure 25: Data on Google sheet and notification

Figure 26 shows the received email about the intrusion

Figure 26: Received email

6.2.2 Shock sensor

Table 7 shows the possibilities and effects of System enable/disable, and Shock Sensor enable/disable on full house security.

Table 7: Control of Shock Sensor (When Shock Sensor is triggered)

System enable	Shock Sensor enable	ESP8266	ESP32 (Main Controller)	Security of Window	Full house Security	Refer
Yes	Yes	Detect change and send command to ESP32	Received command from Shock sensor	ON	ON	Figure 27 Figure 28 Figure 29
Yes	No	Didn't detect any change	Didn't received command	OFF	ON	Appendix 11.2.1.2.1
No	Yes	Didn't detect any change	Didn't received command	OFF	OFF	Appendix 11.2.1.2.2
No	No	Didn't detect any change	Didn't received command	OFF	OFF	Appendix 11.2.1.2.3

In **Figure 24**, the shock button and system button are in ON state.

Labelled 1: It is the serial monitor of ESP32, it shows the new client request as shock, sending the notification, sending data to Ubidots IoT platform, Switch on the buzzer, sending data on Google sheets. And Buzzer is switched OFF by button (buzzer) on Ubidots.

Labelled 2: It is the serial monitor of ESP8266 with a Shock sensor. It shows the connection to ESP32 and then sending the command to ESP32.

Labelled 3: It is the interface of Ubidots, showing the data. And highlighted part verifies the data that arrived at the same time as the shock sensor triggered.

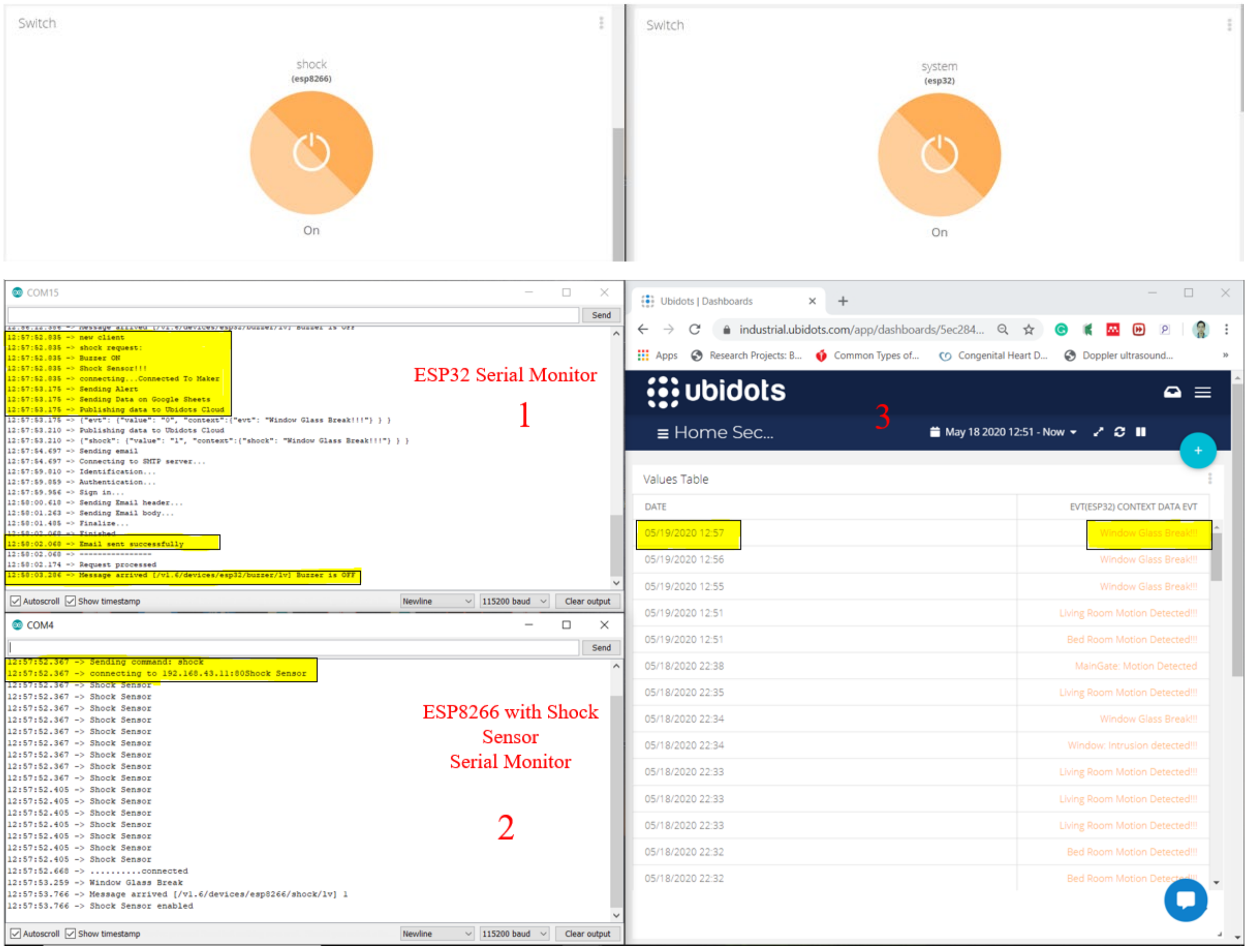


Figure 27: When shock button and system button is in ON state

Figure 28 shows the data on Google sheets and mobile notification via IFTTT.

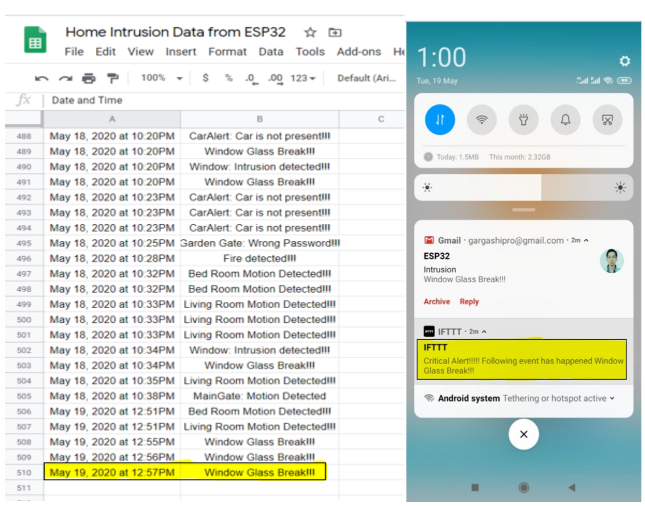


Figure 28: Data on Google sheet and mobile notification

Figure 29 shows the received email about the intrusion.

Figure 29: Email regarding Intrusion

6.2.3 Bedroom PIR sensor

Table 8 shows the possibilities and effects of System enable/disable, and Bedroom PIR sensor enable/disable on full house security.

Table 8: Control of Bedroom PIR Sensor (When Bedroom PIR Sensor is triggered)

System enable	Bedroom PIR Sensor enable	ESP8266	ESP32 (Main Controller)	Security of Window	Full house Security	Refer
Yes	Yes	Detect change and send command to ESP32	Received command from Bedroom PIR Sensor	ON	ON	Figure 30 Figure 31 Figure 32
Yes	No	Didn't detect any change	Didn't received command	OFF	ON	Appendix 11.2.1.3.1
No	Yes	Didn't detect any change	Didn't received command	OFF	OFF	Appendix 11.2.1.3.2
No	No	Didn't detect any change	Didn't received command	OFF	OFF	Appendix 11.2.1.3.3

In **Figure 30**, the pir_bedroom button and system button is in ON state.

Labelled 1: It is the serial monitor of ESP32, it shows the new client request as pir_bedroom, sending the notification, sending data to Ubidots IoT platform, Switch on the buzzer, sending data on Google sheets. And Buzzer is switched OFF by button (buzzer) on Ubidots.

Labelled 2: It is the serial monitor of ESP8266 with a Bedroom PIR sensor. It shows the connection to ESP32 and then sending the command to ESP32.

Labelled 3: It is the interface of Ubidots, showing the data. And highlighted part verifies the data that arrived at the same time as the shock sensor triggered.

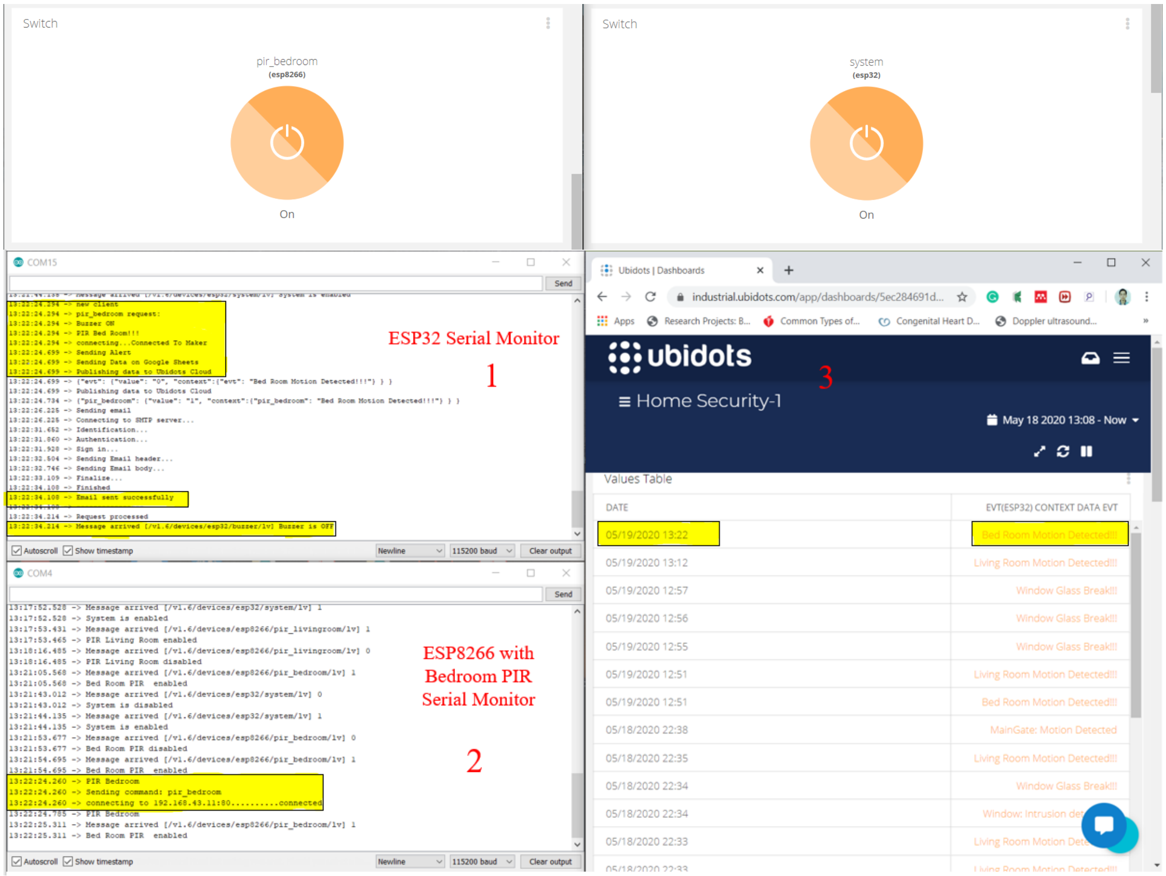


Figure 30: When pir_bedroom button and system button is in ON state

Figure 31 shows the data on Google sheets and mobile notification via IFTTT.

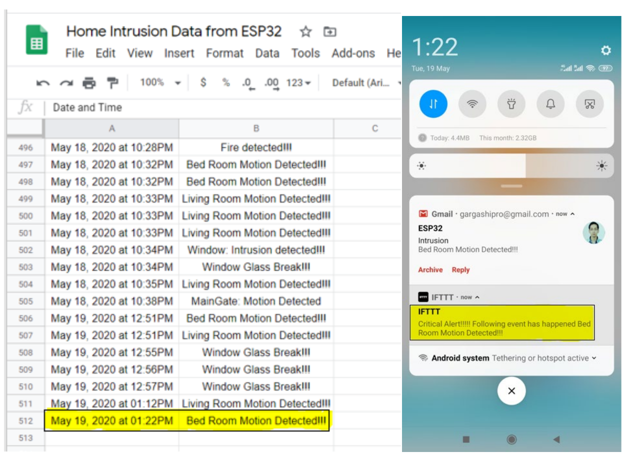


Figure 31: Data on Google sheet and mobile notification

Figure 32 shows the received email about the intrusion.

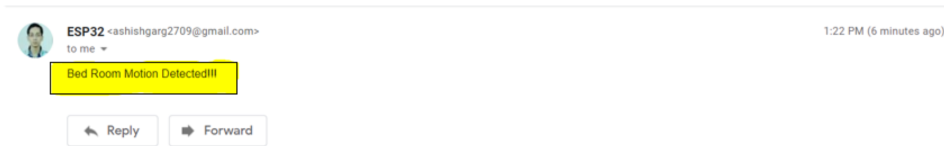


Figure 32: Email regarding Intrusion

6.2.4 Living Room PIR sensor

Table 9 shows the possibilities and effects of System enable/disable and Living Room PIR Sensor enable/disable on full house security.

Table 9: Control of Living Room PIR Sensor (When Living Room PIR Sensor is triggered)

System enable	**Living Room PIR Sensor enable**	**ESP8266**	**ESP32 (Main Controller)**	**Security of Window**	**Full house Security**	**Refer**
Yes	Yes	Detect change and send command to ESP32	Received command from Living Room PIR Sensor	ON	ON	Figure 33 Figure 34 Figure 35
Yes	No	Didn't detect any change	Didn't received command	OFF	ON	Appendix 11.2.1.4.1
No	Yes	Didn't detect any change	Didn't received command	OFF	OFF	Appendix 11.2.1.4.2
No	No	Didn't detect any change	Didn't received command	OFF	OFF	Appendix 11.2.1.4.3

In **Figure 33**, the pir_livingroom button and system button is in ON state.

Labelled 1: It is the serial monitor of ESP32, it shows the new client request as pir_bedroom, sending the notification, sending data to Ubidots IoT platform, Switch on the buzzer, sending data on Google sheets. And Buzzer is switched OFF by button (buzzer) on Ubidots.

Labelled 2: It is the serial monitor of ESP8266 with a Bedroom PIR sensor. It shows the connection to ESP32 and then sending the command to ESP32.

Labelled 3: It is the interface of Ubidots, showing the data. And highlighted part verifies the data that arrived at the same time as the shock sensor triggered.

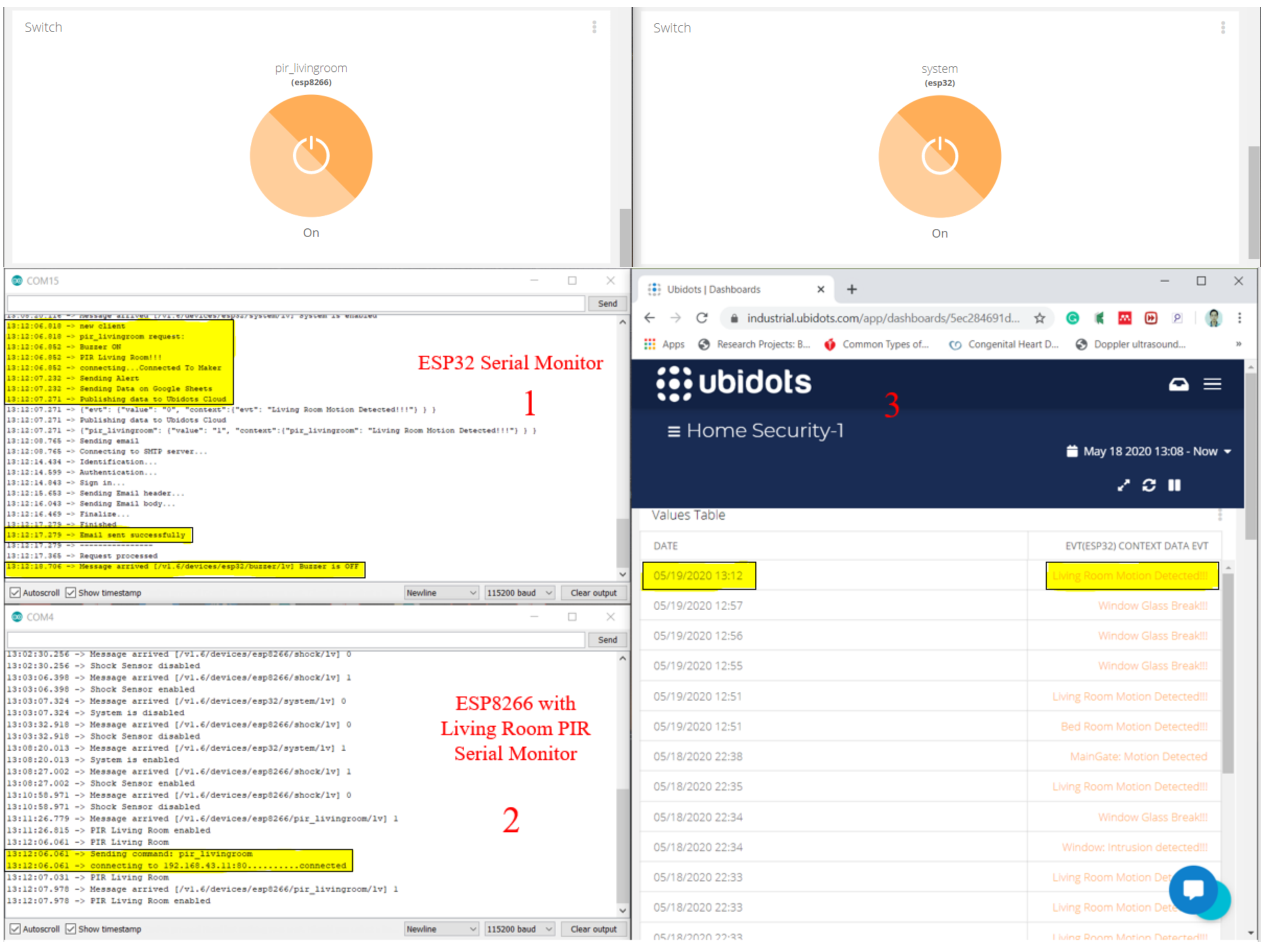


Figure 33: When pir_livingroom button and system button is in ON state

Figure 34 shows the data on Google sheets and mobile notification via IFTTT.

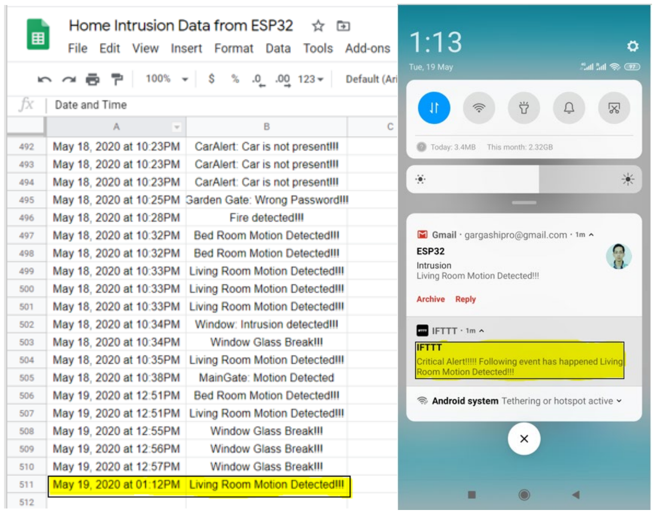


Figure 34: Data on Google sheet and mobile notification

Figure 35 shows the received email about the intrusion.

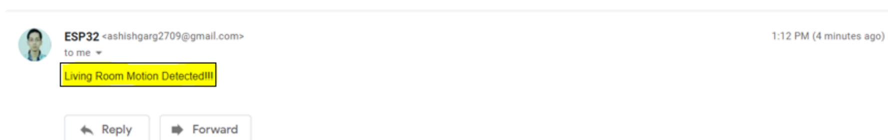


Figure 35: Email regarding Intrusion

6.3 ESP8266 with Fire Sensor and PIR sensor for Corridor Security

The fire sensor is installed to protect the home from fire. With the fire sensor, a PIR sensor is also connected to the same ESP8266 for corridor security. But they both didn't interfere with their functions. The circuit diagram is shown in **Figure 36**.

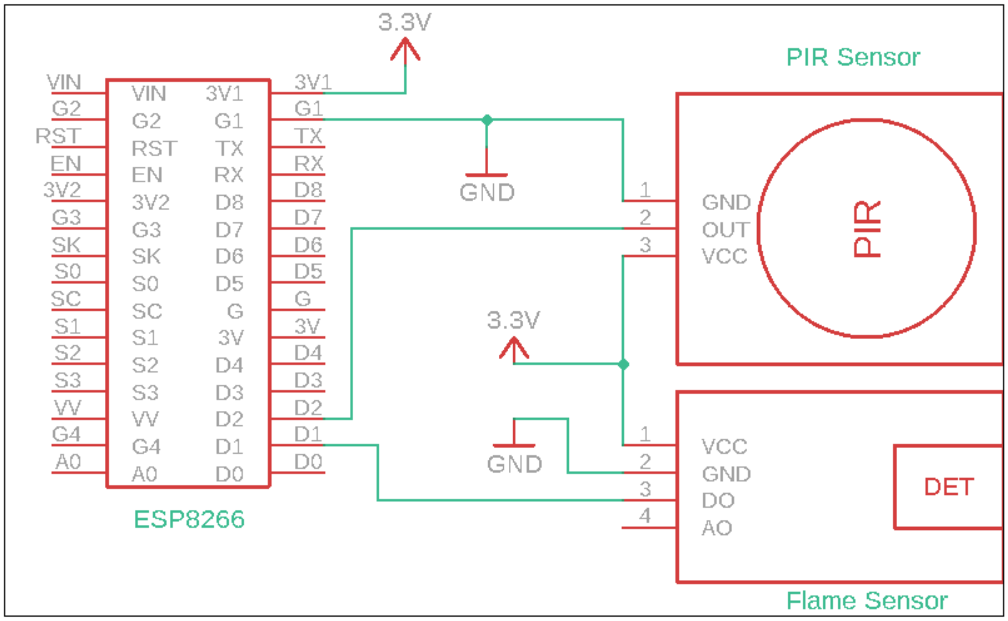


Figure 36: Connection of fire sensor with ESP8266

The complete flow of fire security and corridor security is shown in **Figure 37.** It shows that for informing the main controller about the fire, there should be Flame Sensor Enable and System Enable. If they both are enabled, then after triggering the fire sensor, the command will be sent to the main controller. For corridor PIR sensor also it shows that for informing the main controller about the motion, there should be Corridor PIR Enable and System Enable. If they both are enabled, then after triggering the PIR sensor, the command will be sent to the main controller. All other possibilities and effects of System enable/disable, corridor PIR sensor enable/disable, and flame sensor enable/disable on full house security, corridor security, and fire security is described in **Table 10**.

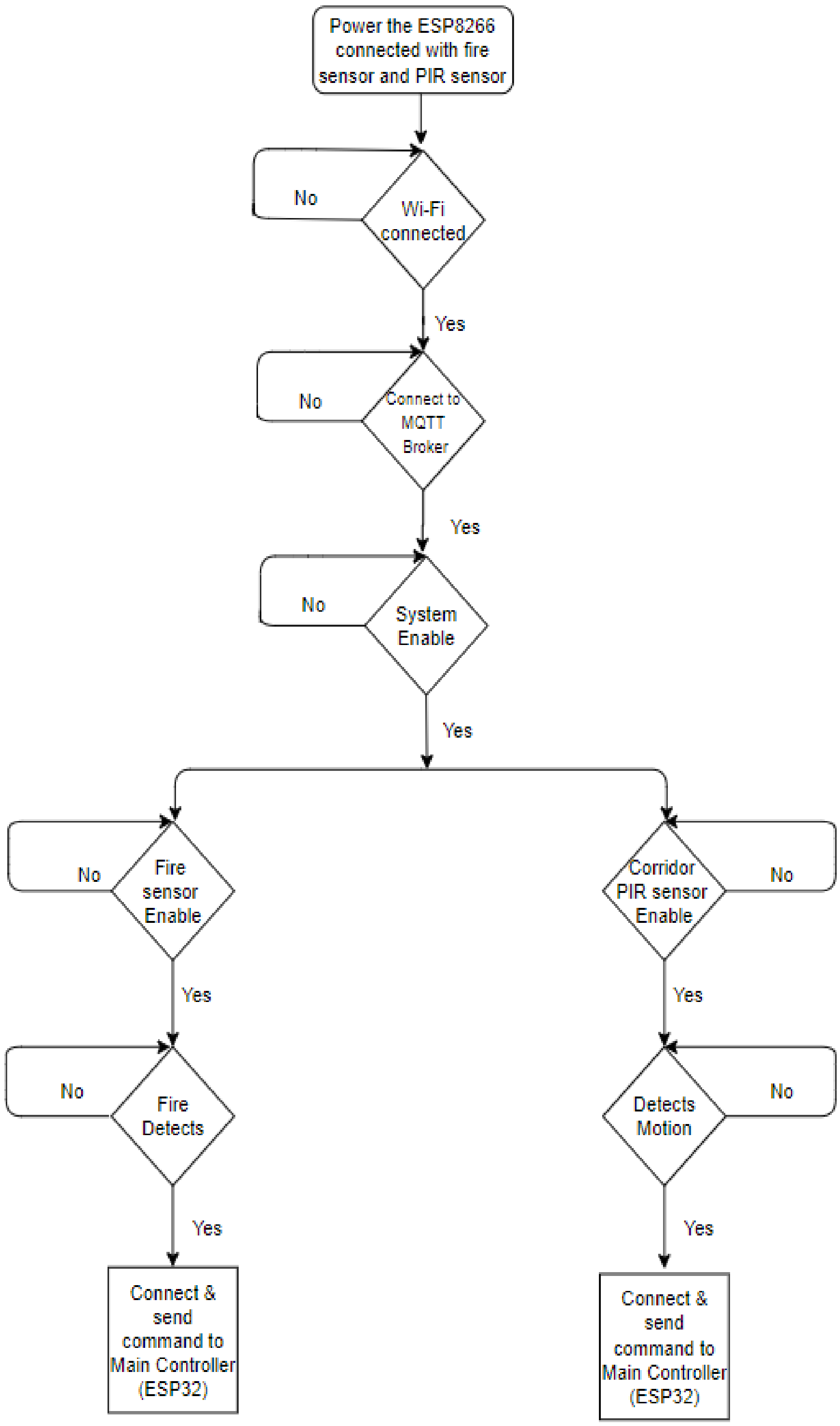


Figure 37: Flow chart of fire security and Corridor security

Table 10: Control of fire sensor and corridor PIR sensor

System enable	Fire sensor enable	Corridor PIR sensor enable	ESP8266 with fire sensor and PIR sensor	ESP32 (Main Controller)	Fire Security	Motion Detection	Full house Security	Refer
Yes	Yes	Yes	Can detect fire, motion and send command to ESP32	Can receive command if fire and motion detected	ON	ON	ON	Figure 38 to Figure 41
Yes	No	Yes	Can detect Motion but not fire	Can receive command if motion detected	OFF	ON	ON	Appendix 11.2.2.1
Yes	Yes	No	Can detect fire but not motion	Can receive command if fire detected	ON	OFF	ON	Appendix 11.2.2.2
No	Yes	Yes	Didn't detect any change	Didn't received command	OFF	OFF	OFF	Appendix 11.2.2.3
No	No	No	Didn't detect any change	Didn't received command	OFF	OFF	OFF	Appendix 11.2.2.4

In **Figure 38**, the fire, corridor PIR, and system button are in ON state.

Labelled 1: It is the serial monitor of ESP32, it shows the new client request as fire and corridor, sending the notification, sending data to Ubidots IoT platform, Switch on the buzzer, sending data on Google sheets. And Buzzer is switched OFF by Ubidots.

Labelled 2: It is the serial monitor of ESP8266 with a Fire Sensor and Corridor PIR sensor. It shows the connection to ESP32 and then sending the command to ESP32.

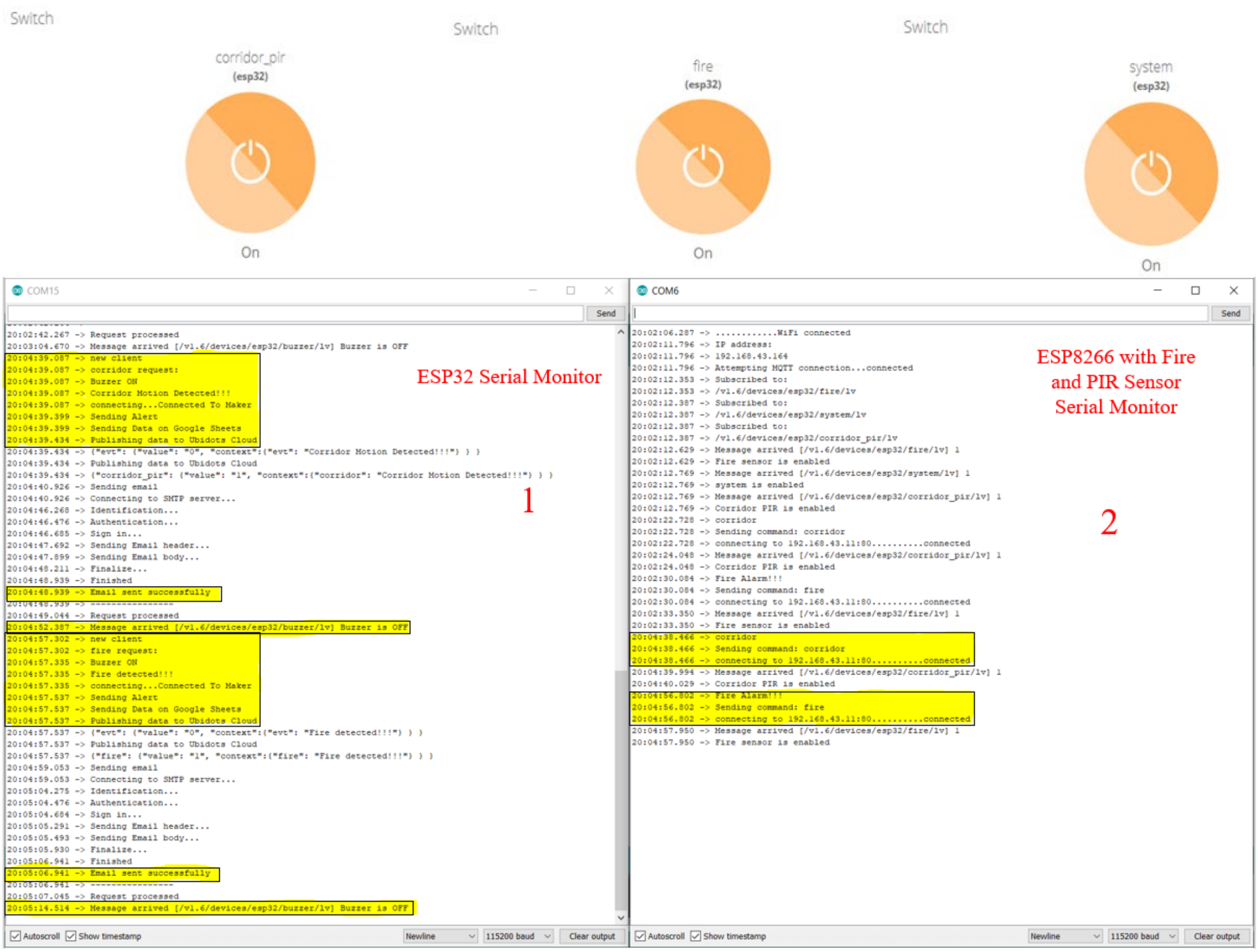


Figure 38: When fire button, corridor PIR button and system button are in ON state

Figure 39 shows the data on Ubidots and mobile notification.

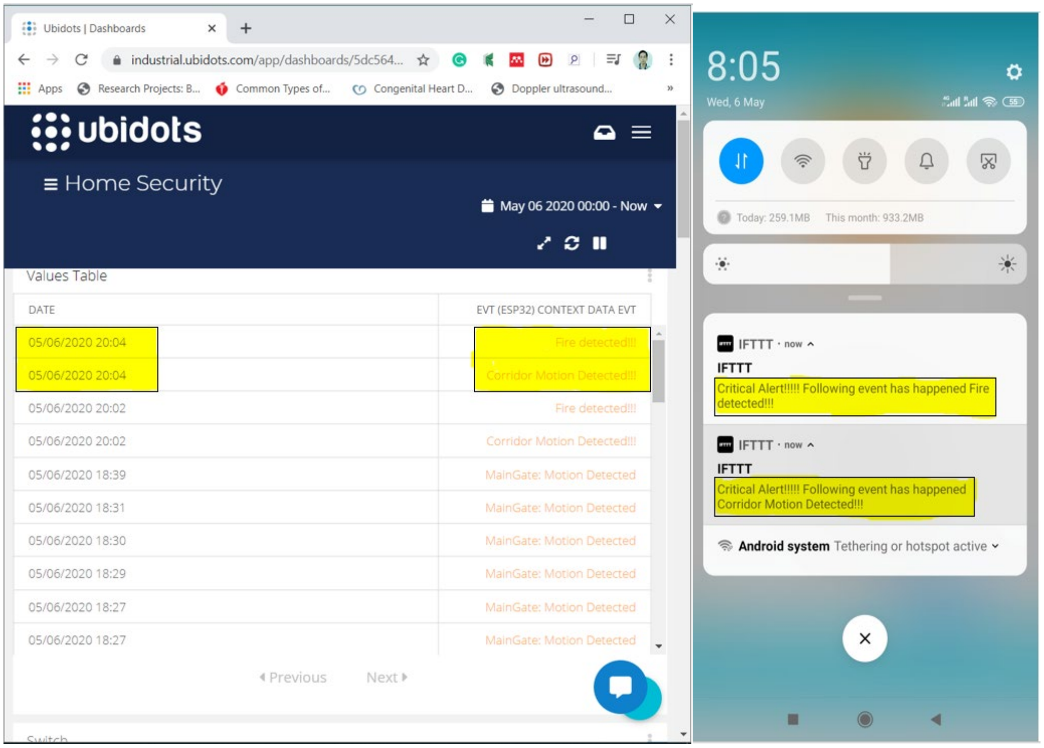


Figure 39: Ubidots Interface and mobile notification

Figure 40 shows the data logged on Google sheet. And highlighted part verifies the data that arrived at the same time as the fire sensor and corridor PIR triggered.

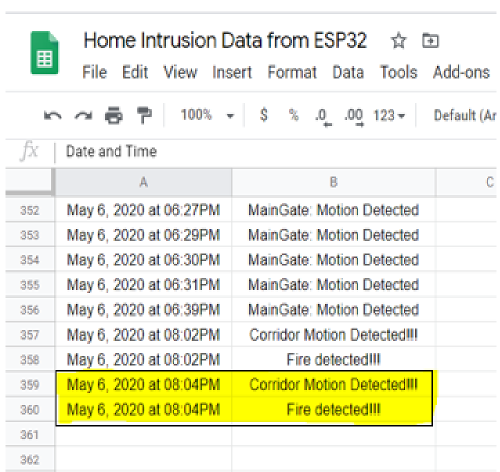


Figure 40: Data on Google Sheets

Figure 41 shows the received email about the fire and Corridor Motion detection.

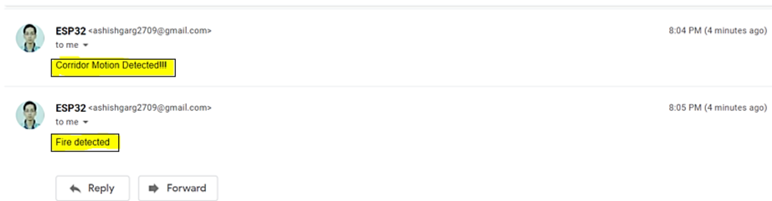


Figure 41: Email received

6.4 ESP8266 with Ultrasonic Sensor and PIR sensor for Garage Security

For the protection of Garage from intrusion, Ultrasonic sensor is installed in the garage. **Figure 42** shows the connections of the Ultrasonic sensor with ESP8266.

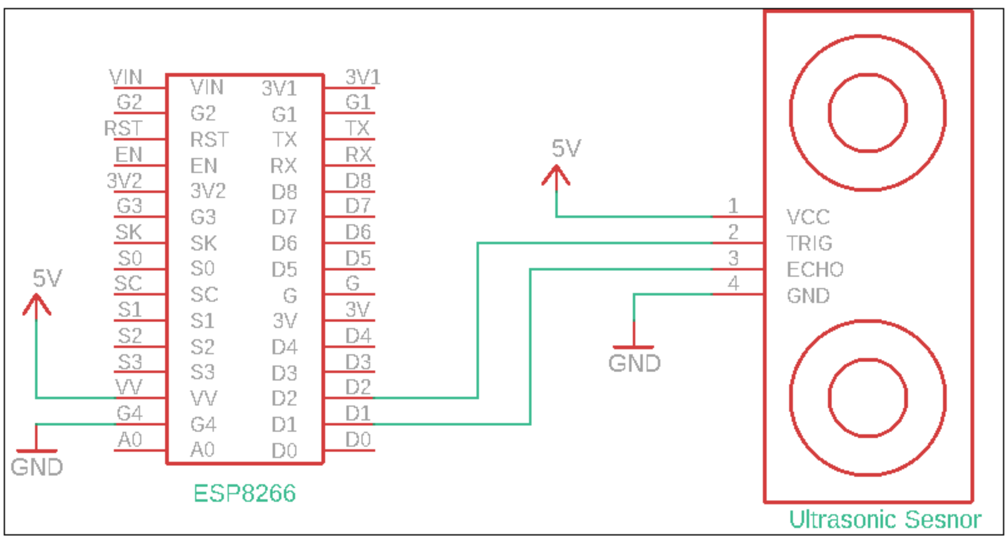


Figure 42: Connections of PIR and Ultrasonic sensor with ESP8266

The complete flow of the process is shown in **Figure 43.** It shows that for informing the main controller about the intrusion, there should be Garage Security Enable and System Enable. When the Ultrasonic sensor detects the movement of the vehicle, then the command will be sent to the main controller. All other possibilities and effects of System enable/disable, and Garage security enable/disable on full house security, and Vehicle security is described in **Table 11**.

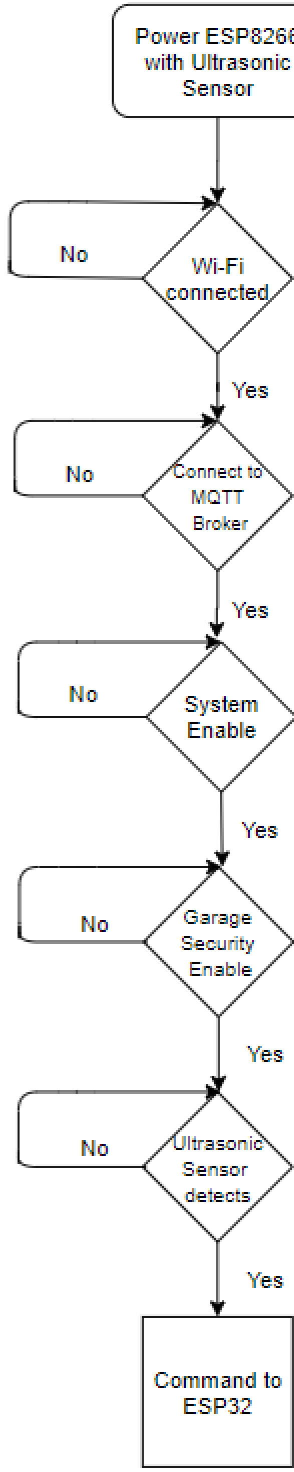


Figure 43: Flow chart of Garage Security

Table 11: Control of Ultrasonic sensor for Garage Security (When triggered)

System enable	Ultra-enable	ESP8266 ultrasonic sensor	ESP32 (Main Controller)	Garage Security	Full house Security	Refer
Yes	Yes	Detect change and send command to ESP32	Received command from ESP8266	ON	ON	Figure 44 Figure 45 Figure 46
Yes	No	Didn't detect any change	Didn't received command	OFF	ON	Appendix 11.2.3.1
No	Yes	Didn't detect any change	Didn't received command	OFF	OFF	Appendix 11.2.3.2
No	No	Didn't detect any change	Didn't received command	OFF	OFF	Appendix 11.2.3.3

In **Figure 44**, the ultrasonic button and system button is in ON state.

Labelled 1: It is the serial monitor of ESP32, it shows the new client request as 'car,' sending the notification, sending data to Ubidots IoT platform, Switch on the buzzer, sending data on Google sheets. And Buzzer is switched OFF by button (buzzer) on Ubidots.

Labelled 2: It is the serial monitor of ESP8266 with an ultrasonic sensor. It shows the connection to ESP32 and then sending the command to ESP32.

Labelled 3: It is the interface of Ubidots, showing the data. And highlighted part verifies the data that arrived at the same time when intrusion is detected.

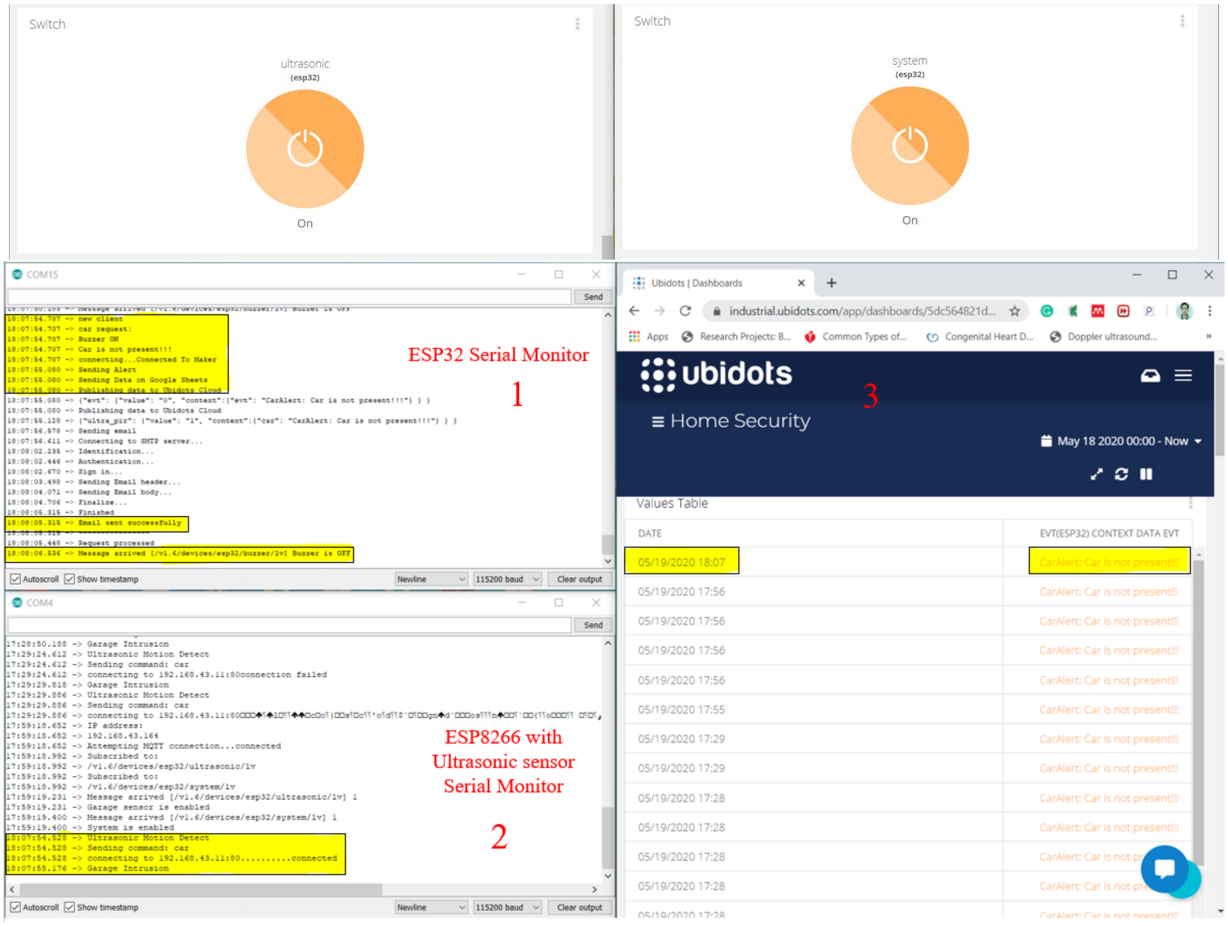


Figure 44: When ultrasonic button and System button is in ON state

Figure 45 shows the data logged on Google sheet and mobile notification via IFTTT. And highlighted part verifies the data that arrived at the same time when intrusion is detected.

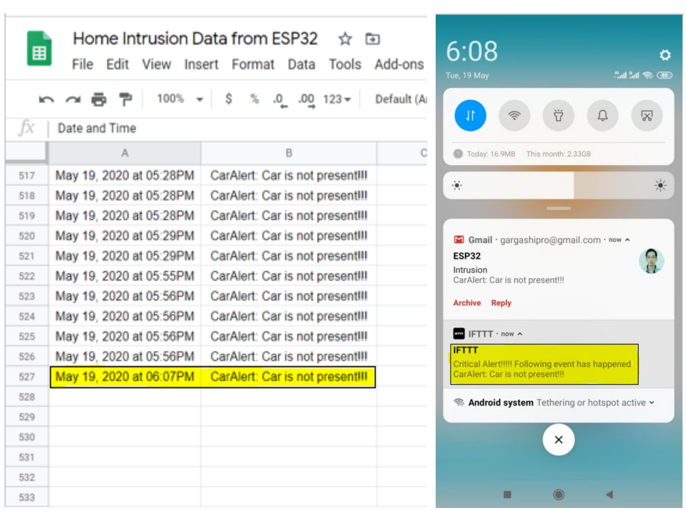


Figure 45: Data on Google sheet and mobile notification

Figure 46 shows the received email about the intrusion.

Figure 46: Received email

6.5 ESP8266 with Keypad for Garden Security

The keypad is installed to protect the garden from unauthorized entry. The keypad is connected with the ESP8266. The circuit diagram is shown in **Figure 47**. Rows of the keypad are connected to pin D7, D6, D5, and D4 and columns to pin D0, D1, and D2. The motor is connected between pin D8 and ground. Power to ESP8266 is provided by USB cable.

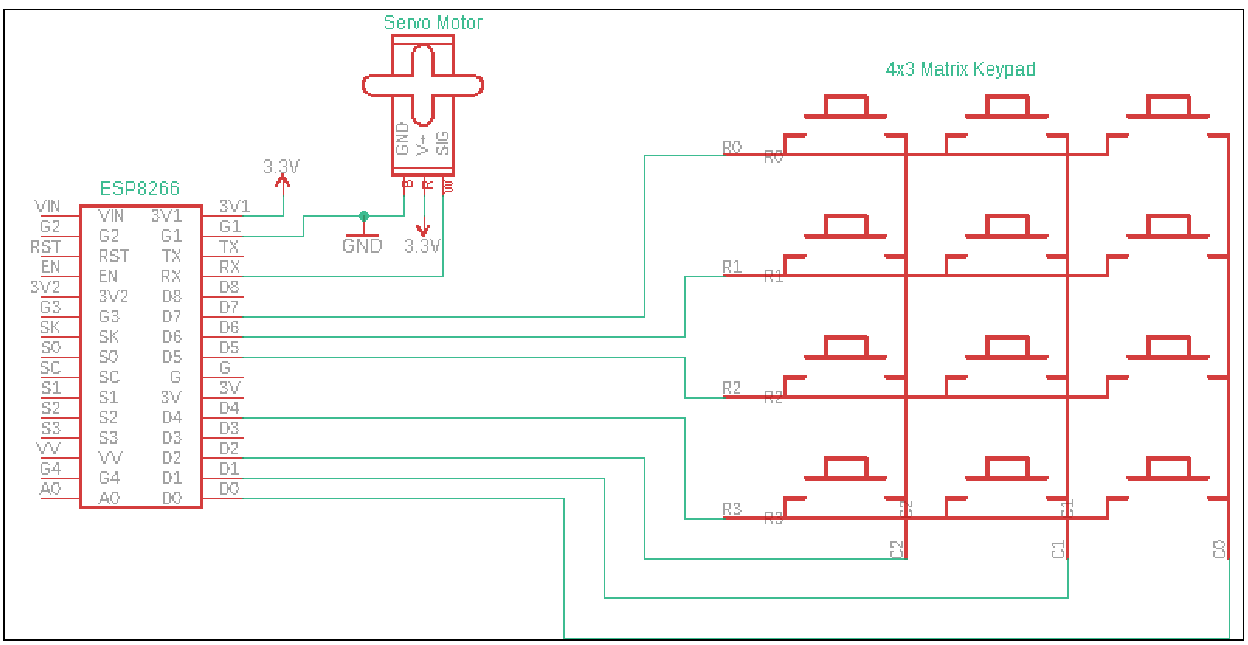


Figure 47: Connection of Keypad and motor with ESP8266

The complete flow of Garden security is shown in **Figure 48.** It shows that for informing the main controller about the unauthorized entry, there should be keypad Enable and System Enable. If they both are enabled, and someone entered the password wrong 3 times, then the command will be sent to the main controller. All other possibilities and effects of System enable/disable, and Keypad enable/disable on full house security, and Garden security is described in **Table 12**.

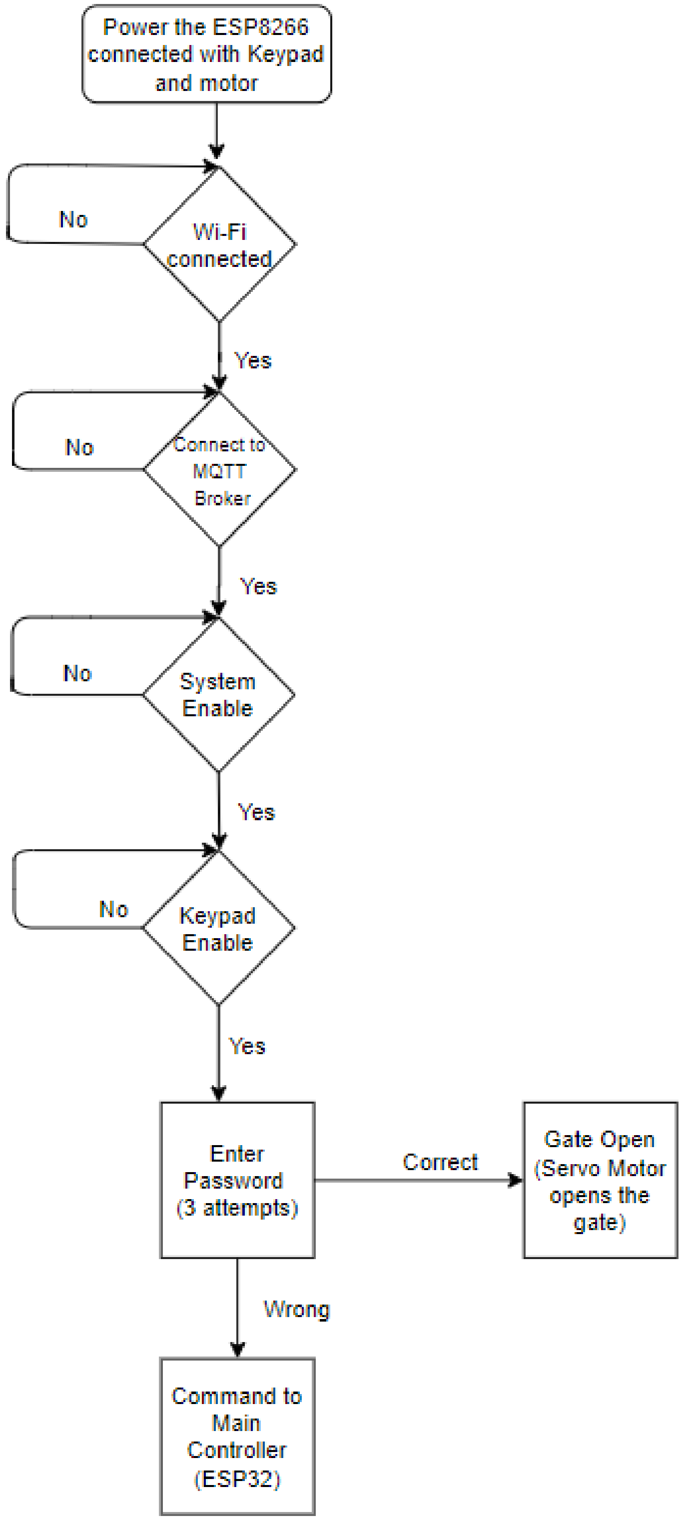


Figure 48: Flow chart of the keypad

Table 12: Control of keypad (When the wrong password is entered for 3 times)

System enable	Keypad enable	ESP8266 with keypad	ESP32 (Main Controller)	Security of Garden	Full house Security	Refer Figure
Yes	Yes	Detect the wrong password (3^{rd} time) and send command to ESP32	Received command from Keypad	ON	ON	Figure 49 to Figure 52
Yes	Yes	If correct password entered, then no command to ESP32, it opens the gate.	Didn't received command	ON	ON	Figure 53
Yes	No	Didn't detect	Didn't received command	OFF	ON	Appendix 11.2.4.1
No	Yes	Didn't detect	Didn't received command	OFF	OFF	Appendix 11.2.4.2
No	No	Didn't detect	Didn't received command	OFF	OFF	Appendix 11.2.4.3

In **Figure 49**, both the system button and keypad button are in ON state.

Labelled 1: It is the serial monitor of ESP32, it shows the new client request as 'garden,' sending the notification, sending data to Ubidots IoT platform, Switch on the buzzer, sending data on Google sheets.

Labelled 2: It is the serial monitor of ESP8266 with a keypad. It shows that after entering the wrong password 3^{rd} time, it sends the command 'garden' to ESP32.

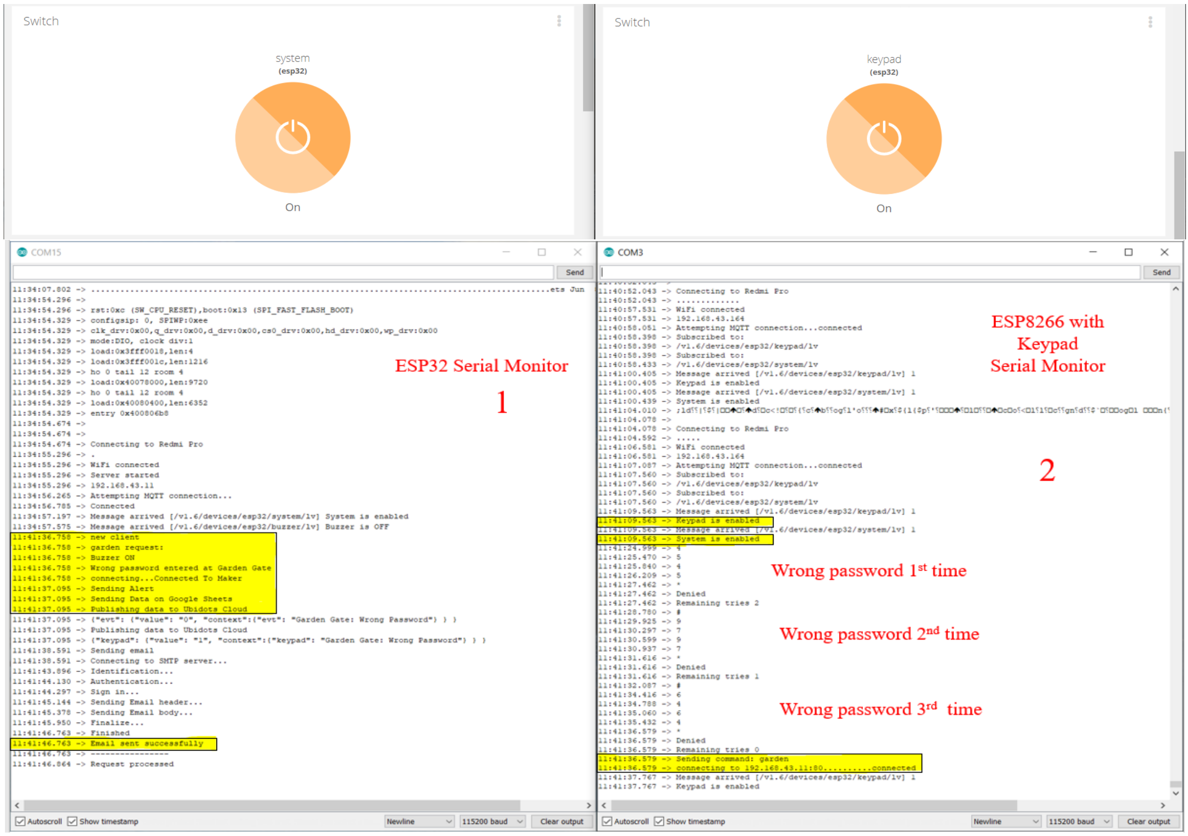


Figure 49: When keypad button and system button is in ON state

In **Figure 50**, both the keypad button and system button are in ON state.

It shows the Ubidots interface, and the highlighted part indicates the data arrived on Ubidots after entering the wrong password 3rd time.

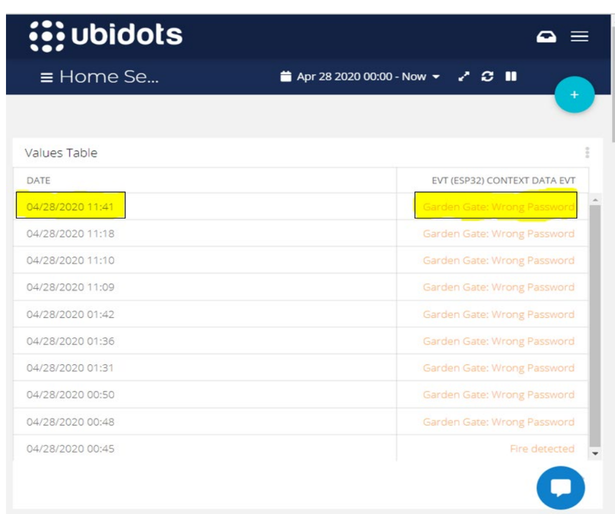


Figure 50: Ubidots interface

Figure 51 shows the email arrived after entering wrong password 3rd time.

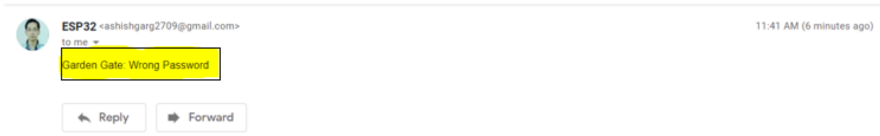


Figure 51: Email regarding Intrusion

Figure 52 shows the data on Google sheets and mobile notification via IFTTT.

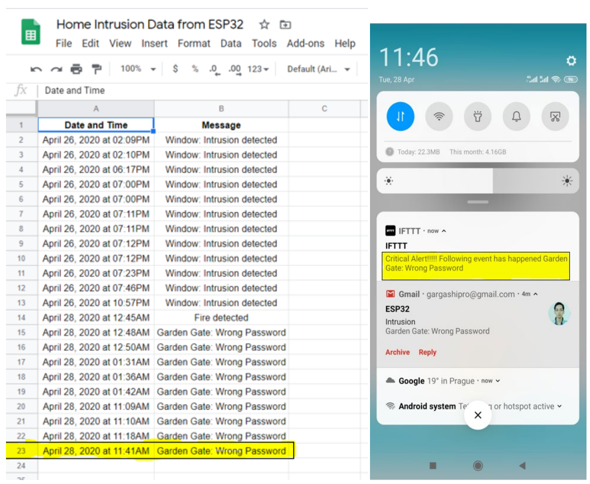


Figure 52: Data on Google sheets and mobile notification

Figure 53, shows that when keypad button and system button is in ON state, and we enter the correct password, it will accept the password and servo motor will open the gate.

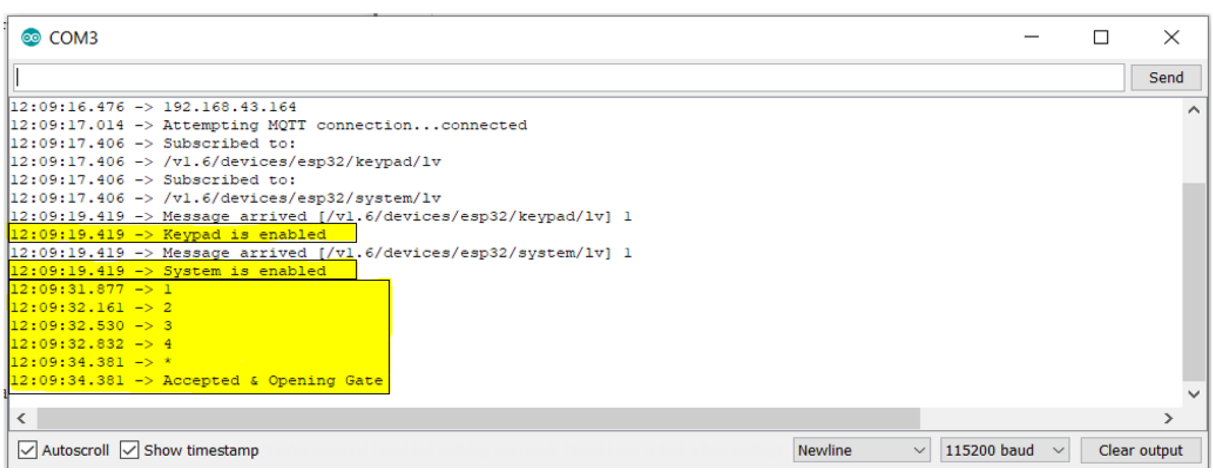


Figure 53: When both system button, keypad button is in ON state and password is correct

6.6 ESP32 CAM with PIR sensor and Microwave Sensor for Front Gate Security

For the protection of the house from intruder from the front gate, dual sensor technology is used. The PIR sensor and Microwave sensor is installed at the main gate of the house to protect from intrusion. ESP32 CAM doesn't have a built-in programmer, so the FTDI programmer is required to upload the sketch on ESP32 CAM. Its connections are shown in **Figure 54**. After uploading the sketch to ESP32 CAM, the IO0 pin should be disconnected from Ground. In the designed system with ESP32 CAM, Serial Peripheral Interface Flash File System (SPIFFS) is used. SPIFFS is a lightweight filesystem created for microcontrollers with a flash chip, which is connected by the SPI bus, like the ESP32 flash memory. When the PIR sensor and Microwave Sensor both detect the motion, ESP32 CAM takes a photo saves it into the SPIFFS and then attach it with email. It always overwrites the previous image in memory. There is another option of saving the image on the microSD card, then attaching the image with email and entered into the deep sleep mode. It was also tried but encountered many problems like getting a problem in getting connected with Wi-Fi after waking up from deep sleep mode, getting Guru meditation error, camera not supported error, and with using microSD card very less number of pins are left as GPIOs. So, to avoid this, SPIFFS is preferred.

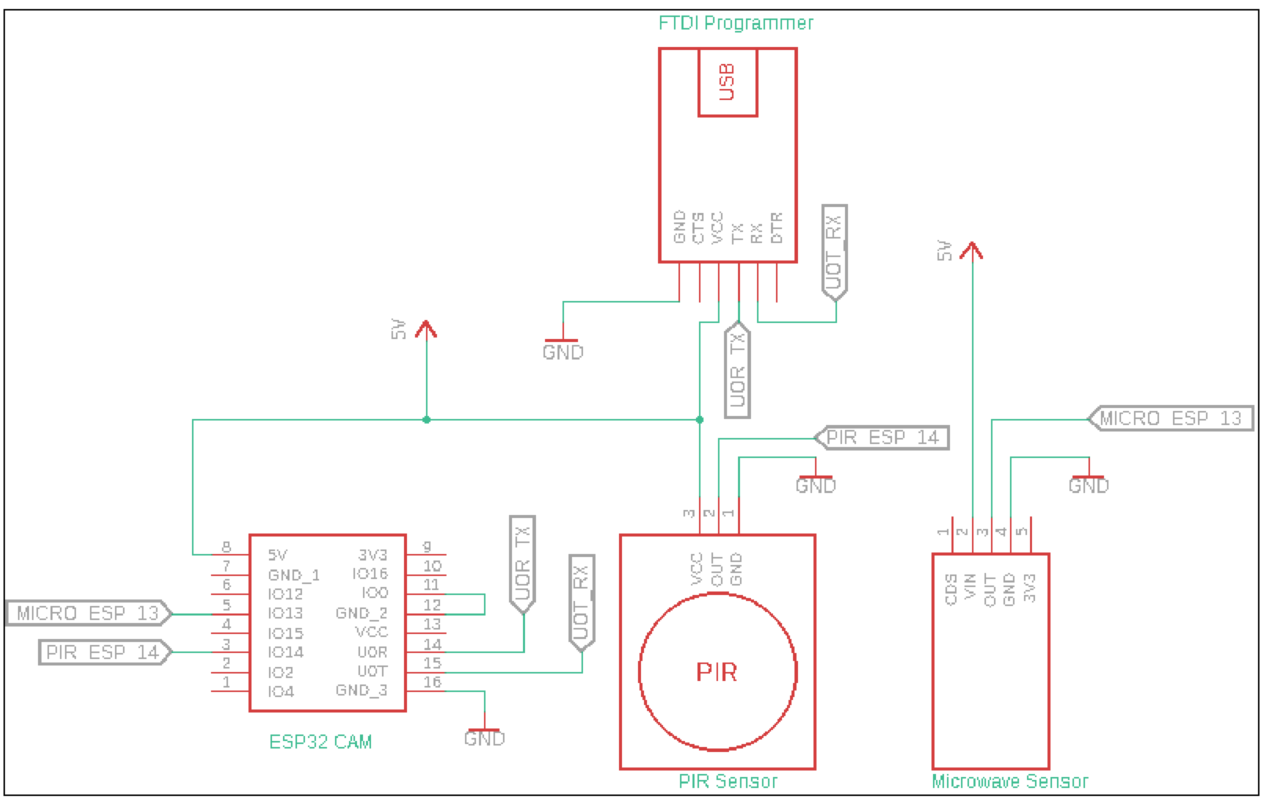


Figure 54: Connection of FTDI programmer with ESP32 CAM, PIR sensor & Microwave Sensor

The complete flow of the front gate security is shown in **Figure 55.** It shows that for informing the main controller about the intrusion, there should be PIR Enable and System Enable. If they both are enabled, and motion is detected in front of the main gate by both Microwave and PIR sensor, then the command will be sent to the main controller. The owner will receive two emails, one from ESP32 CAM with a photo of the intruder and another from ESP32 informing about the intrusion. All other possibilities and effects of System enable/disable, and PIR enable/disable on full house security, and main gate security is described in **Table 13.**

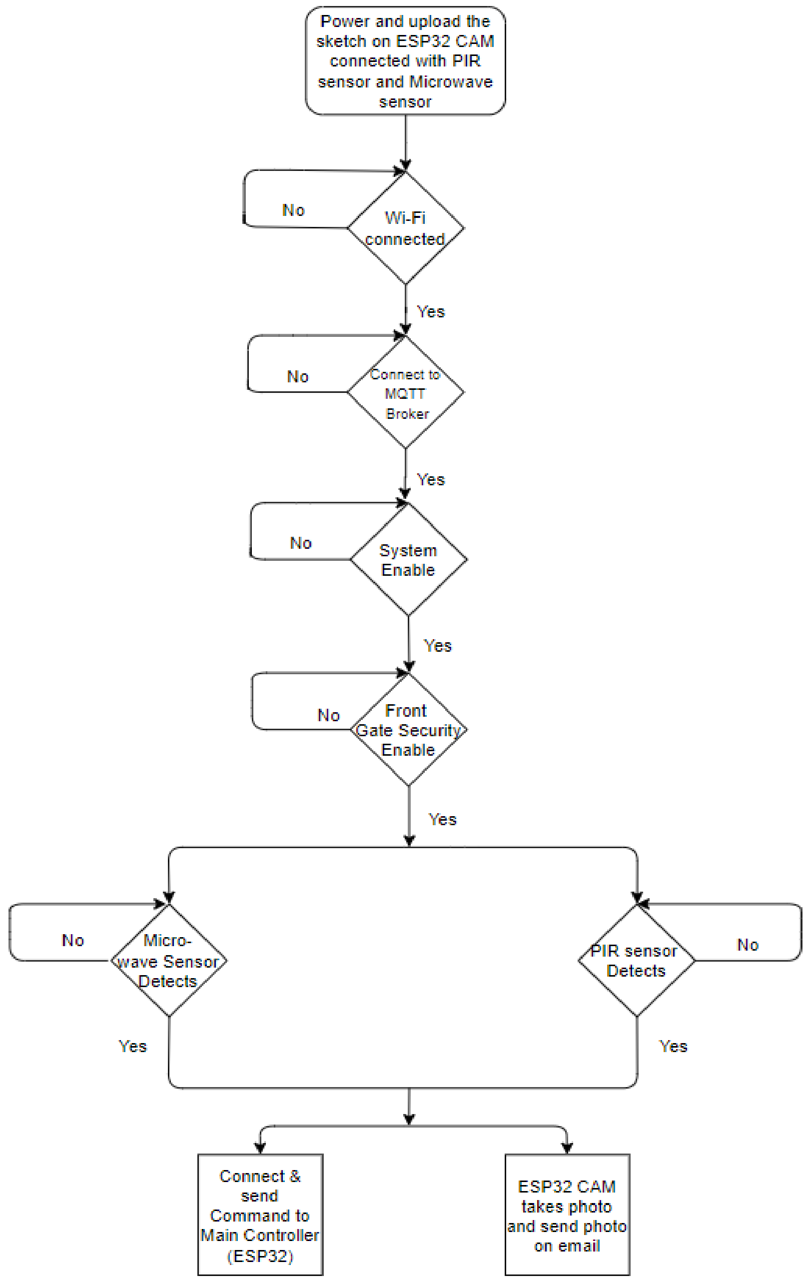


Figure 55: Flow Chart of Front Gate Security

Table 13: Control of PIR sensor (When sensor is triggered means motion is detected)

System enable	**PIR enable**	**ESP32 CAM with PIR sensor and Microwave sensor**	**ESP32 (Main Controller)**	**Main Gate Security**	**Full house Security**	**Refer Figure**
Yes	Yes	Detect motion and send command to ESP32 and send Photo on email	Received command from PIR sensor	ON	ON	Figure 56 to Figure 60
Yes	No	Didn't detect	Didn't received command	OFF	ON	Appendix 11.2.5.1
No	Yes	Didn't detect	Didn't received command	OFF	OFF	Appendix 11.2.5.2
No	No	Didn't detect	Didn't received command	OFF	OFF	Appendix 11.2.5.3

In **Figure 56**, both the system button and pir button are in ON state.

Labelled 1: It is the serial monitor of ESP32, it shows the new client request as 'pir', sending the notification, sending data to Ubidots IoT platform, Switch on the buzzer, sending data on Google sheets. And then buzzer is switched OFF from Ubidots.

Labelled 2: It is the serial monitor of ESP32 CAM. It shows that after detection of motion by both the Microwave sensor and PIR sensor, it sends command 'pir' to ESP32, then take a photo and send the photo on email. It will not send the command to the ESP32 if only one sensor detects the motion.

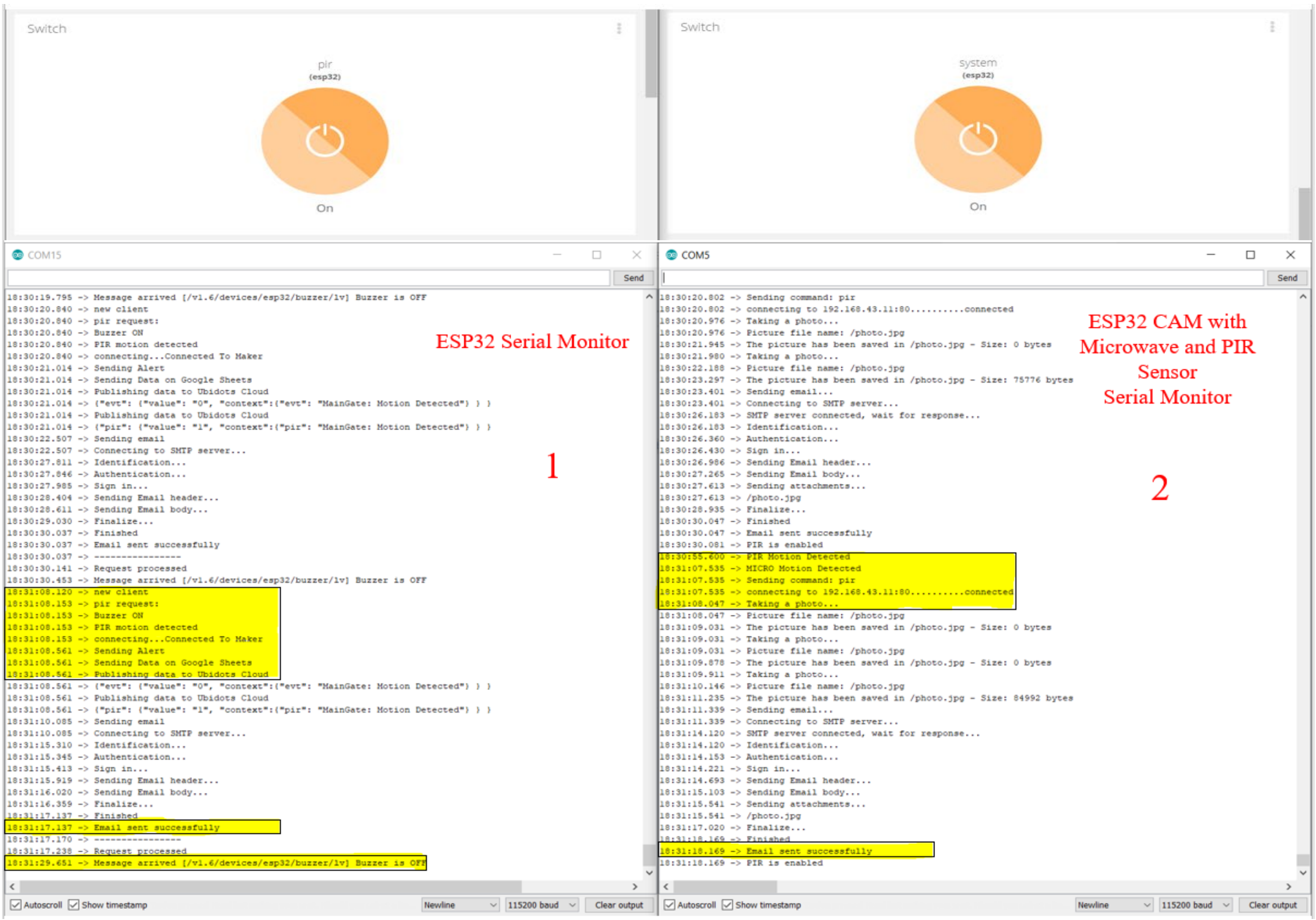


Figure 56: When both system button and pir button are in ON state

In **Figure 57,** both system button and pir button are in ON state. It shows the Ubidots interface, and the highlighted part indicates the data arrived on Ubidots after motion detection.

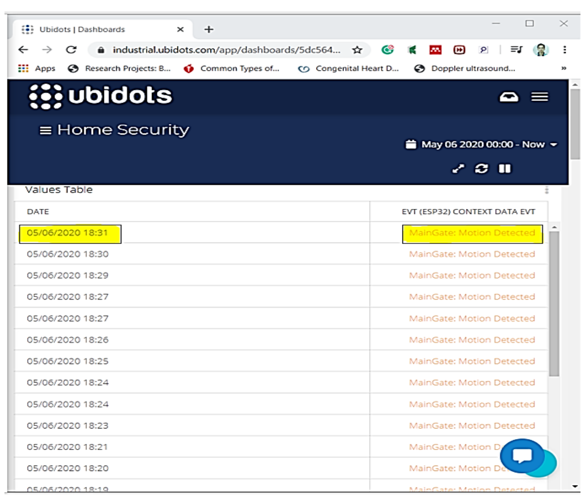


Figure 57: Ubidots interface

Figure 58 shows the data on Google sheets and mobile notification via IFTTT.

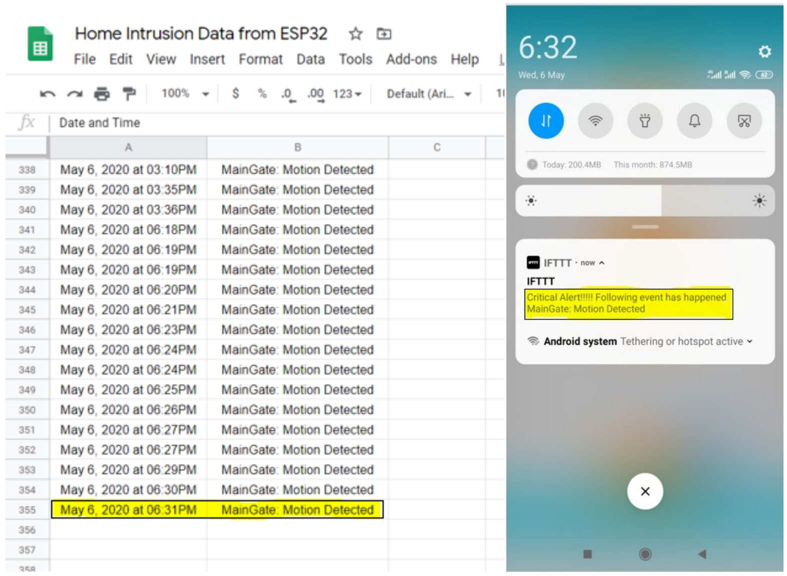


Figure 58: Data on Google sheets

Figure 59, shows the email with attached Photo arrived after motion detection from ESP32 CAM.

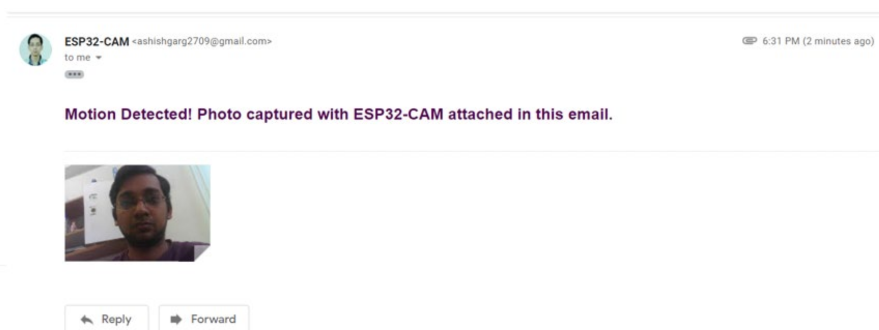


Figure 59: Email from ESP32 CAM with Photo

Figure 60, shows the email arrived after motion detection from ESP32.

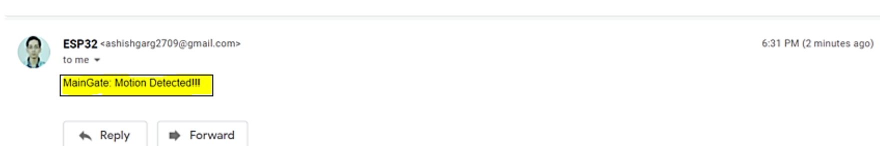


Figure 60: Email from ESP32

7. Economic Analysis

The cost shown in **Table 14** is the cost of the components purchased from the local market. So, the prices are quite high. But if they were brought in bulk and from other sources, then the prices might be low. The prices shown do not include labor costs, which adds significantly to the total cost.

Table 14: Price List of different components

Component	Quantity	Price (per module) in CZK	Total (in CZK)	Link
ESP32	1	519	519	https://www.gme.cz/esp32-esp-32s-2-4ghz-dual-mode-wifi-bluetooth
ESP8266	4	240	960	https://www.gme.cz/nodemcu-esp8266-lua-wifi-v3-ch340
ESP32 CAM	1	529	529	https://arduino-shop.cz/arduino/7587-esp32-cam
PIR Sensor	3	65	195	https://www.gme.cz/pir-modul-hc-sr501
Flame Sensor	1	20.34	20.34	https://www.gmelectronic.com/infrared-flame-sensor
Ultrasonic Sensor	1	79	79	https://www.gme.cz/modul-ultrazvukoveho-merice-vzdalenosti-hc
Microwave sensor	1	35	35	https://www.gme.cz/detektor-pohybu-mikrovlnny-doppler-radar-rcwl-0516
Magnetic Switch	1	69.45	69.45	https://www.gmelectronic.com/magnetic-reed-switch-ls-311-b38
Shock Sensor	1	34	34	https://www.gme.cz/modul-sw-18010p-otresove-cidlo-analog-a-digital-vystup
Keypad	1	34.17	34.17	https://www.gmelectronic.com/membran-keyboard-3x4
FTDI Programmer	1	105	105	https://www.gme.cz/ftdi-prevodnik-s-mini-usb-a-spi
Buzzer	1	59	59	https://www.gme.cz/sirenka-kpk20a25w
Total Price	16		**2638.96**	

This designed prototype is not ready for the series of production by any company because there is a need for certain improvements such as the proper enclosures for the components, certifications, approvals, proper powering of the modules, and removal of other shortcomings. After including these operations, the pricing of the designed system will increase significantly. For demonstration purposes powering of the microcontrollers is done by power bank, USB Hub, and USB cables, but those are not included in the price list.

8. Commercial Products available in the market

Many companies have their product in the market for Home Security, such as:

8.1 SmartThings ADT Home Security

It is one of the most trusted brands in the Home Security Market. It has a wireless touch screen panel and supports many third-party smart home devices. There is 24/7 monitoring with support of remote device control through the mobile application. However, it is expensive. It combined with Samsung SmartThings. SmartThings app allows ADT security to arm and disarms the system. Its starter kit costs 550 USD. [59], [60]

8.2 Vivint Smart Home

In this system, we can customize the product as per our needs. Wall-mounted touch screen, and it supports the Alexa/Google Assistant smartphone app. They also offer 24/7 security monitoring. Its equipment pricing starts at 599 USD. [59], [61]

8.3 SimpliSafe Home Security System

It is a well-known brand in the market and a good alternative to ADT products. It has affordable monthly fees, quick and easy installation, cellular, Wi-Fi connectivity, and even 24/7 professional monitoring. Its starter kit costs 230 USD. [59]–[61]

8.4 Ring Alarm Security Kit

It is easy to install, supports multiple wireless platforms. It offers affordable professional monitoring. It is a better option than the Honeywell Security system and Nest Secure. It can work with Alexa. Its basic price is 179 USD. [59], [61]

8.5 Honeywell Smart Home Security Kit

It is one of the well- known brands in the market and is a DIY (Do It Yourself) kit. There is no need for professional installation. It is built-in Alexa voice service. It supports Google Assistant, IFTTT, and Z-Wave Plus. It includes motion detection, face recognition, cloud storage. Professional monitoring is not available. Its starter kit price is 500 USD. [59], [60]

8.6 Abode iota All-In-One Security Kit

It has all standard sensors and devices like Alexa, IFTTT, voice-enabled control, Zigbee, and Z-Wave. On-demand professional monitoring. It is equipped with a 1080p security camera, motion sensor, and many more can be added as per requirement. Its price is 229 USD. [59]–[61]

8.7 Nest Secure

The nest is part of Google. Easy to install and can work with many third-party devices. It provides a hub with a motion sensor, two Tag key fobs, and nest detects sensor for windows door, and other areas. It is expensive, and its initial cost is 399 USD. [59]–[61]

8.8 Scout Alarm Home Security System

It can be customized as per our demand. It works with Amazon Alexa. Moreover, professional monitoring is available on a monthly basis. Its initial cost is 247 USD. [59], [60]

The photos of the few Commercial Products described above is shown in **Figure 61.**

Honeywell Smart Home Security Kit [60]

SimpliSafe Home Security System [60]

Abode iota All-In-One Security Kit [59]

Ring Alarm Security system[59]

SmartThings ADT Security system panel [60]

Vivint Security system panel [61]

Figure 61: Few Commercial Products

.

A comparison between the above-described products is shown in **Table 15**.

Table 15: Comparison of different commercial products [59]–[61]

	Best For	Contract	Installation	Price	Support
ADT	Full-Featured, Professional Home Monitoring	Required	Professional	550 USD	Amazon Alexa, Google Assistant, Z-wave
Vivint Smart Home	All-In-One Security and Home Automation	Required	Professional	599 USD	Amazon Alexa, Google Assistant, Z-wave
SimpliSafe Home Security System	No-Contract Home Monitoring	Month to month	DIY (Do it Yourself)	230 USD	Amazon Alexa, Google Assistant
Ring Alarm Security Kit	Affordable Professional Monitoring	Not Required	DIY (Do it Yourself)	179 USD	Amazon Alexa, Google Assistant, Z-wave
Honeywell Smart Home Security Kit	Flexible Security	Not Required	DIY (Do it Yourself)	500 USD	Amazon Alexa, Google Assistant, Z-wave, IFTTT
Abode iota All-In-One Security Kit	Flexible All-In-One Security	Not Required	DIY (Do it Yourself)	229 USD	Amazon Alexa, Google Assistant, Z-wave, IFTTT
Nest Secure	Google-Powered Home Security	Not Required	DIY (Do it Yourself)	399 USD	Amazon Alexa, Google Assistant, Z-wave
Scout Alarm Home Security System	Customize and flexible option	Month to month	DIY (Do it Yourself)	247 USD	Amazon Alexa
Designed Prototype	Customize and flexible option	Not Required	DIY (Do it Yourself)	2638.96 CZK (105.31 USD)	IFTTT, Google Sheets

There are a lot of other commercial products available in the market, but only a few of them are discussed above. All products support many facilities like integration with Amazon Alexa and etc. Some are available with professional monitoring, and some are not. In some, we need to pay month by month. So, it's hard to choose from the wide variety of products in the market. Some problems are face by the customers in the commercially available products such as no coverage for the full house, the problem with calibration with other sensors, high cost, false alarm, use of proprietary hardware. As per the home security system market, high cost in installing and maintenance is the crucial factor for limiting the market growth [15]. Customers are aware of the benefits of the Home Security System, but the expensive hardware and cost of ownership restrain them in the adoption of the Home Security System. These are some major issues of the commercially available systems. But the designed system, is not costly, designed with the locally available sensors, it is fully customized so that we can add more sensors as per the size of the house, it can be easily integrated with other sensors. The false alarm issue can be resolved by installing the sensors at a suitable height in case of the motion sensor, and if we are at home, we can easily disarm the particular sensor through mobile so that alarm cannot be triggered due to such as pets. For the false alarm issue, dual technology is used in the designed system. For front gate motion detection, PIR and Microwave sensor are used. And there is no proprietary hardware, or the software is used in the system.

9. Conclusion

The thesis begins with an explanation of the Electronic Security System (ESS) and its basic elements. After that, a short description of the Internet of Things (Chapter 3), and about the different wireless technologies that are available and can be used in the IoT applications was presented.

For the first point of the assignment of the thesis, a Literature Review (Chapter 4) was done related to the Home Security System, more focused on ESP32. But several other literatures were also added. The literatures of Home Security System were compared with each other on the basis of the functionality, and the used technology. The challenges that were stated in the literature were also mentioned.

For the second point of the assignment, a Home Security System was proposed (Chapter 5). As per the challenges mentioned in the first point, relevant components and technology were selected. Properties of each component and the technology were mentioned. Components includes-ESP32, ESP8266, and ESP32 CAM microcontrollers. Other components including sensors like-Flame Sensor, PIR Sensor, Magnetic Switch, Ultrasonic Sensor, Microwave Sensor, Keypad, Buzzer, and Servo Motor. ESPs (ESP32, ESP8266, and ESP32 CAM) are Wi-Fi compatible devices. All the communication is done over the MQTT protocol. MQTT is a very lightweight Protocol which has the publish and subscribe pattern and is most suited for the IoT application. Sensors, Keypad, and servo motor are connected with ESP8266 and ESP32 CAM (PIR Sensor and Microwave Sensor). A buzzer is connected with ESP32. When a sensor is triggered means intrusion occurs, then it sends the specific command to the ESP32, which publishes that (event) on the MQTT Broker on a specific Topic. Ubidots is the MQTT broker and is a cloud service (IoT platform) that offers a friendly and intuitive interface to store and analyze sensor data in real-time. For every sensor as per their topic, a control widget (Switch) is added on the Ubidots Dashboard. If the subscriber is subscribed to that same topic, it will receive the messages, as per our toggling (On/OFF) of Switch. For example, for Window Security, a Magnetic Switch is installed on the Window. If any intrusion occurs from the window, then the Magnetic Switch module (Magnetic Switch connected with ESP8266) will send the command "magnetic" to the ESP32. Then ESP32 will publish on Ubidots MQTT Broker with topic /v1.6/devices/ esp32 /magnetic. We will see the data on the Ubidots IoT Platform with the timestamp. Magnetic Switch with ESP8266 is subscribed to that topic. Now, if we want to disarm the Magnetic Switch module, we can Switch OFF the Switch that was installed on the Ubidots IoT Platform. After disarming the Magnetic Switch Module, it will not sense the intrusion. But the rest of the system will continue to work. In the designed system, each module can be disarmed individually from Ubidots Dashboard, and the full system can also be disarmed collectively. Ubidots also have its mobile application, from which also we can control our system. Besides of publishing to MQTT Broker, ESP32 is connected with buzzer. When any intrusion occurs, buzzer will be triggered, and can only be switch OFF from Ubidots dashboard. ESP32 also sends information about the intrusion to the owner via email and notification. Notification is sent via IFTTT ("if this, then that), which is a Web service that helps to connect other Web services. As per the limitation of Ubidots, the data on it can be stored only for one month, so the data regarding the intrusion events is also stored on the Google Sheets with the date and time. And it is also done by IFTTT. All the programming of the designed system is done on the Arduino IDE platform. For solving the issue of False Triggering, dual technology sensor is used. In this technology, multiple sensing technologies are combined in a single module. For motion detection, PIR and Microwave sensors are combined. And for motion to be detected, both the sensors should be activated. It is installed on the front gate. When both sensors detect motion, then only ESP32 CAM take a photo and send that photo to the owner's email account. Other sensors those were installed at different places of the house are: a PIR sensor with ESP8266 is installed in the corridor for corridor security, a fire sensor is installed for fire security, a magnetic switch, shock sensor (as glass break sensor) for protection intrusion from the window, a PIR sensor in both Living room and Bed room, an ultrasonic sensor is used in Garage for vehicle theft security, a keypad is installed with ESP8266 for garden security, which needs a correct password for opening the gate with the help of servo motor. If a person enters the wrong password 3 times, then the command will be sent to the ESP32, indicating the intrusion. The working of each individual sensor module is explained in Chapter 6.

A short description of the pricing is given in Chapter 7. The total cost of the used components is 2638.96 CZK (105.31 USD). But the price will be higher because, for the final product, proper certifications and approval are needed, which significantly increases the total price of the product.

For the third point of the assignment, certain commercial products were compared in Chapter 8. The pricing of commercial products is higher than the designed system. There are several issues in commercial systems like high cost, false alarm, and many more. Different solutions were adopted for these issues, like the use of dual technology sensors, using locally available sensors.

In its current form, this designed system can be used after some modifications like a proper enclosure of the devices and certifications. Further advancement can be done by using more sensors and adding more security features like RFID, fingerprint sensor. Information and cybersecurity in the Internet of Things are one of the future aspects. Use of Artificial Intelligence and Machine Learning for facial and Voice recognition for differentiating between known and unknown persons.

10. Bibliography

[1] "DGA | Alarm Industry History," *DGA Security*. [Online]. Available: https://www.dga.com/company/alarm-industry-history/. [Accessed: 03-May-2020].

[2] "History of the Alarm System - Alarm systems - Break-in." [Online]. Available: https://www.abus.com/eng/Guide/Break-in-protection/Alarm-systems/History-of-the-alarm-system. [Accessed: 03-May-2020].

[3] "Edwin Holmes (inventor) - Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/Edwin_Holmes_(inventor). [Accessed: 03-May-2020].

[4] "The Fire Alarm Telegraph System – Firefighters Hall & Museum." [Online]. Available: http://firehallmuseum.org/about-us-2/about-us/the-fire-alarm-telegraph-system/. [Accessed: 03-May-2020].

[5] R. M. MARSTON, *Security Electronics Circuits Manual*, First. Newnes, 1998.

[6] "What Are Common Components of a Security System?" [Online]. Available: https://www.safewise.com/home-security-faq/security-system-components/. [Accessed: 20-Mar-2020].

[7] Miroslav Husak, "System Approach to the Protection of Objects and Theft." [Online]. Available: https://moodle.fel.cvut.cz/pluginfile.php/225593/mod_resource/content/1/02_System approach to object protection_2019.pdf. [Accessed: 21-Mar-2020].

[8] F. J. D'Addario, *Personal Safety and Security Playbook Risk Mitigation Guidance for Individuals, Families, Organizations, and Communities*. Elsevier, 2014.

[9] Miroslav Husak, "Protection against Penetration (Perimeter, Wall)." [Online]. Available: https://moodle.fel.cvut.cz/pluginfile.php/225595/mod_resource/content/1/03_Perimeter%2Bwall_protections_2019.pdf. [Accessed: 21-Mar-2020].

[10] "RELIABILITY." [Online]. Available: http://web4.uwindsor.ca/users/f/fbaki/85-222.nsf/0/b7d85091e772a10185256f84007be5c1/$FILE/Lecture_07_ch6_222_w05_s5.pdf. [Accessed: 05-May-2020].

[11] "Symbols in Security Systems." [Online]. Available: https://www.electrical-symbols.com/electric-electronic-symbols/security-systems-symbols.htm. [Accessed: 03-May-2020].

[12] ITU Corporation, "Internet of Things Global Standards Initiative," *Internet of Things Global Standards Initiative*, 2015. [Online]. Available: http://www.itu.int/en/ITU-T/gsi/iot/Pages/default.aspx. [Accessed: 03-May-2020].

[13] F. Wortmann and K. Flüchter, "Internet of Things: Technology and Value Added," *Business and Information Systems Engineering*. 2015.

[14] S. R. Rishi Rahul, “Future of IoT.” [Online]. Available: http://ficci.in/spdocument/23092/Future-of-IoT.pdf. [Accessed: 03-May-2020].

[15] “Home Security System Market | Size, Growth, Trend and Forecast to 2023 | MarketsandMarkets.” [Online]. Available: https://www.marketsandmarkets.com/Market-Reports/home-security-system-market-205573901.html. [Accessed: 08-May-2020].

[16] “What are the major components of Internet of Things - RF Page.” [Online]. Available: https://www.rfpage.com/what-are-the-major-components-of-internet-of-things/. [Accessed: 03-May-2020].

[17] S. S. I. Samuel, “A review of connectivity challenges in IoT-smart home,” in *2016 3rd MEC International Conference on Big Data and Smart City, ICBDSC 2016*, 2016.

[18] “IEEE 802.11ax: The Sixth Generation of Wi-Fi.” [Online]. Available: https://www.cisco.com/c/dam/en/us/products/collateral/wireless/white-paper-c11-740788.pdf. [Accessed: 03-May-2020].

[19] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, “A study of Lora: Long range & low power networks for the internet of things,” *Sensors (Switzerland)*, 2016.

[20] T. G. Durand, L. Visagie, and M. J. Booysen, “Evaluation of next-generation low-power communication technology to replace GSM in IoT-applications,” *IET Commun.*, vol. 13, no. 16, pp. 2533–2540, 2019.

[21] J. Rubio-Aparicio, F. Cerdan-Cartagena, J. Suardiaz-Muro, and J. Ybarra-Moreno, “Design and implementation of a mixed IoT LPWAN network architecture,” *Sensors (Switzerland)*, vol. 19, no. 3, 2019.

[22] Y. Yang, X. Luo, X. Chu, and M.-T. Zhou, “IoT Technologies and Applications,” in *Fog-Enabled Intelligent IoT Systems*, Cham: Springer International Publishing, 2020, pp. 1–37.

[23] S. Tanwar, P. Patel, K. Patel, S. Tyagi, N. Kumar, and M. S. Obaidat, “An advanced Internet of Thing based Security Alert System for Smart Home,” in *IEEE CITS 2017 - 2017 International Conference on Computer, Information and Telecommunication Systems*, 2017.

[24] A. Khanum and V. Rekha, “An enhanced security alert system for smart home using IOT,” *Indones. J. Electr. Eng. Comput. Sci.*, vol. 13, no. 1, pp. 27–34, Jan. 2019.

[25] C. Davidson, T. Rezwana, and M. A. Hoque, “Smart Home Security Application Enabled by IoT:: Using Arduino, Raspberry Pi, NodeJS, and MongoDB,” in *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*, 2019.

[26] A. Panwar and A. Singh, “Eyrie Smart Home Automation using Internet of Things,” in *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, 2017, no. July, pp. 1368–1370.

[27] K. V. Saivineeth, B. Vamshi, and V. K. Mittal, "Wireless voice-controlled multi-functional secure eHome," in *2017 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2017*, 2017.

[28] A. Howedi and A. Jwaid, "Design and implementation prototype of a smart house system at low cost and multi-functional," in *FTC 2016 - Proceedings of Future Technologies Conference*, 2017, pp. 876–884.

[29] W. A. Jabbar *et al.*, "Design and Fabrication of Smart Home with Internet of Things Enabled Automation System," *IEEE Access*, 2019.

[30] A. Anitha, "Home security system using internet of things," in *IOP Conference Series: Materials Science and Engineering*, 2017.

[31] Taryudi, D. B. Adriano, and W. A. Ciptoning Budi, "Iot-based Integrated Home Security and Monitoring System," in *Journal of Physics: Conference Series*, 2018.

[32] Andreas, C. R. Aldawira, H. W. Putra, N. Hanafiah, S. Surjarwo, and A. Wibisurya, "Door Security System for Home Monitoring Based on ESP32," *Procedia Comput. Sci.*, vol. 157, pp. 673–682, 2019.

[33] P. Rai and M. Rehman, "ESP32 based smart surveillance system," in *2019 2nd International Conference on Computing, Mathematics and Engineering Technologies, iCoMET 2019*, 2019.

[34] R. K. Kodali, B. Yatish Krishna Yogi, G. N. Sharan Sai, and J. Honey Domma, "Implementation of Home Automation Using CoAP," in *IEEE Region 10 Annual International Conference, Proceedings/TENCON*, 2019.

[35] E. Gambi, L. Montanini, D. Pigini, G. Ciattaglia, and S. Spinsante, "A home automation architecture based on LoRa technology and Message Queue Telemetry Transfer protocol," *Int. J. Distrib. Sens. Networks*, 2018.

[36] S. Sanjay Kumar, A. Khalkho, S. Agarwal, S. Prakash, D. Prasad, and V. Nath, "Design of smart security systems for home automation," in *Lecture Notes in Electrical Engineering*, 2019.

[37] "Sweet Home 3D - Draw floor plans and arrange furniture freely." [Online]. Available: http://www.sweethome3d.com/. [Accessed: 22-Apr-2020].

[38] "ESP32 Series Datasheet." [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf. [Accessed: 15-Apr-2020].

[39] "ESP8266EX Datasheet," 2020. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf. [Accessed: 15-Apr-2020].

[40] "esp32-cam-ai-thinker/ESP32-CAM_Product_Specification.pdf at master · raphaelbs/esp32-cam-

ai-thinker · GitHub.” [Online]. Available: https://github.com/raphaelbs/esp32-cam-ai-thinker/blob/master/assets/ESP32-CAM_Product_Specification.pdf. [Accessed: 15-Apr-2020].

[41] “Infrared flame sensor | GM electronic COM.” [Online]. Available: https://www.gmelectronic.com/infrared-flame-sensor. [Accessed: 15-Apr-2020].

[42] “Overview | PIR Motion Sensor | Adafruit Learning System.” [Online]. Available: https://learn.adafruit.com/pir-passive-infrared-proximity-motion-sensor?view=all. [Accessed: 15-Apr-2020].

[43] “Magnetic Switch - an overview | ScienceDirect Topics.” [Online]. Available: https://www.sciencedirect.com/topics/engineering/magnetic-switch. [Accessed: 15-Apr-2020].

[44] “Door Sensor Secrets: What They Are, How They Work & 6 Unexpected Uses.” [Online]. Available: https://simplisafe.com/blog/door-sensor-secrets. [Accessed: 08-Mar-2020].

[45] S. Sendra, L. Garcia, J. M. Jimenez, and J. Lloret, “Low-cost vehicle driver assistance system for fatigue and distraction detection,” in *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*, 2017.

[46] “Servo motor Arduino guides.” [Online]. Available: https://navody.arduino-shop.cz/arduino-projekty/servo-motor.html. [Accessed: 16-Apr-2020].

[47] “In-Depth: Interface 4x3 & 4x4 Membrane Keypad with Arduino.” [Online]. Available: https://lastminuteengineers.com/arduino-keypad-tutorial/. [Accessed: 23-Apr-2020].

[48] “Wi-Fi - ESP32 - — ESP-IDF Programming Guide latest documentation.” [Online]. Available: https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/network/esp_wifi.html. [Accessed: 25-Apr-2020].

[49] “FAQ - Frequently Asked Questions | MQTT.” [Online]. Available: http://mqtt.org/faq. [Accessed: 19-Feb-2020].

[50] “MQTT man page | Eclipse Mosquitto.” [Online]. Available: https://mosquitto.org/man/mqtt-7.html. [Accessed: 24-Apr-2020].

[51] “Ubidots MQTT Broker | Ubidots Help Center.” [Online]. Available: https://help.ubidots.com/en/articles/570008-ubidots-mqtt-broker. [Accessed: 30-Apr-2020].

[52] “MQTT.FX.” [Online]. Available: https://mqttfx.jensd.de/index.php/download. [Accessed: 24-Apr-2020].

[53] X. Guo, D. K. Tiller, G. P. Henze, and C. E. Waters, “The performance of occupancy-based lighting control systems: A review,” *Light. Res. Technol.*, 2010.

[54] “IoT platform features.” [Online]. Available: https://ubidots.com/platform/. [Accessed: 24-Apr-2020].

[55] “Top IoT Platforms for Makers.” [Online]. Available: https://ubidots.com/blog/top-iot-platforms/.

[Accessed: 24-Feb-2020].

[56] "What is the difference between Ubidots and Ubidots STEM? | Ubidots Help Center." [Online]. Available: https://help.ubidots.com/en/articles/639806-what-is-the-difference-between-ubidots-and-ubidots-stem. [Accessed: 11-Mar-2020].

[57] "Arduino - FAQ." [Online]. Available: https://www.arduino.cc/en/main/FAQ. [Accessed: 16-Apr-2020].

[58] S. Ovadia, "Automate the Internet With 'If This Then That' (IFTTT)," *Behavioral and Social Sciences Librarian*. 2014.

[59] "The best home security systems in 2020." [Online]. Available: https://www.digitaltrends.com/home/best-home-security-systems/. [Accessed: 14-Apr-2020].

[60] "Best DIY home security system of 2020." [Online]. Available: https://www.cnet.com/news/best-diy-home-security-system-of-2020-simplisafe-nest-secure/. [Accessed: 14-Apr-2020].

[61] "The Best Smart Home Security Systems for 2020." [Online]. Available: https://www.pcmag.com/picks/the-best-smart-home-security-systems. [Accessed: 14-Apr-2020].

[62] "ESP32S-HiLetgo Dev Boad with Pinout Template - parts submit - fritzing forum." [Online]. Available: https://forum.fritzing.org/t/esp32s-hiletgo-dev-boad-with-pinout-template/5357. [Accessed: 27-Mar-2020].

[63] "ESP8266 NodeMcu Pinout - ESP8266 Shop." [Online]. Available: https://esp8266-shop.com/esp8266-guide/esp8266-nodemcu-pinout/. [Accessed: 27-Mar-2020].

[64] "ESP32-CAM AI-Thinker Pinout Guide: GPIOs Usage Explained | Random Nerd Tutorials." [Online]. Available: https://randomnerdtutorials.com/esp32-cam-ai-thinker-pinout/. [Accessed: 27-Mar-2020].

[65] "PIR modul HC-SR501 | GM electronic, spol. s.r.o." [Online]. Available: https://www.gme.cz/pir-modul-hc-sr501. [Accessed: 15-Apr-2020].

[66] "Jazýčkový magnetický senzor 100VDC/0,5A 1x spínací LS-311-B38 | GM electronic, spol. s.r.o." [Online]. Available: https://www.gme.cz/magneticky-jazyckovy-kontakt-ls-311-b38. [Accessed: 16-Apr-2020].

[67] "Modul ultrazvukového měřiče vzdálenosti HC-SR05 | GM electronic, spol. s.r.o." [Online]. Available: https://www.gme.cz/modul-ultrazvukoveho-merice-vzdalenosti-hc-sr05. [Accessed: 05-May-2020].

[68] "RCWL-0516 Doppler RADAR Motion Sensor." [Online]. Available: https://www.epitran.it/ebayDrive/datasheet/19.pdf. [Accessed: 05-May-2020].

[69] "Module SW-18010P, shock sensor, analog and digital output | GM electronic COM." [Online]. Available: https://www.gmelectronic.com/module-sw-18010p-shock-sensor-analog-and-digital-

output. [Accessed: 19-May-2020].
[70] "Piezosirénka KPK20A38W | GM electronic spol. sro." [Online]. Available: https://www.gme.cz/sirenka-kpk20a38w. [Accessed: 16-Apr-2020].
[71] "Amazon.com: RC Model Mini Servo, 4.3g Mini Micro Analog Servo with Servo Arm for RC Boat Drone Helicopter Airplane Component Replacement Part Accessory: Toys & Games." [Online]. Available: https://www.amazon.com/Helicopter-Airplane-Component-Replacement-Accessory/dp/B07MXHWPM3. [Accessed: 16-Apr-2020].
[72] "Membran 3x4 keyboard | GM electronic COM." [Online]. Available: https://www.gmelectronic.com/membran-keyboard-3x4#product-detail. [Accessed: 16-Apr-2020].

11. Appendix

11.1 Appendix A

11.1.1 Pin description and Specifications of the used components

11.1.1.1 ESP32

Figure 62 shows the pin description of the ESP32S development module, which is used in the Designed Prototype as the main controller.

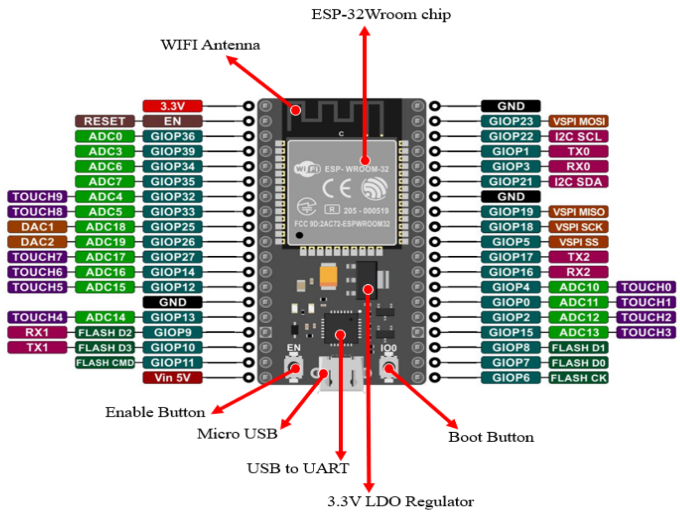


Figure 62: Pin Description of ESP32S Development Module [62]

Specifications [38]

- **Processors:** Tensilica Xtensa 32-bit LX6 microprocessor
- **Wireless connectivity: Wi-Fi:** 802.11 b/g/n/e/i (802.11n @ 2.4 GHz up to 150 Mbit/s)
  Bluetooth: v4.2 BR/EDR and Bluetooth Low Energy (BLE)
- **Typical Frequency:** 160MHz
- **ROM:** 448 KB
- **SRAM:** 520 KB
- **Flash:** SPI Flash up to 16 Mbytes
- **GPIO:** 36
- External 2 MHz ~ 60 MHz crystal oscillator (40 MHz only for Wi-Fi/BT functionality)
- SPI, I2C, I2S, UART, CAN, IR, PWM
- **Clock frequency:** up to 240 MHz
- **Operating Temperature:** -40 - 125°C

11.1.1.2 ESP8266

Figure 63 shows the pin description of the ESP8266 development module, which is used in the Designed Prototype with different sensors to send the sensor data to the main controller (ESP32S).

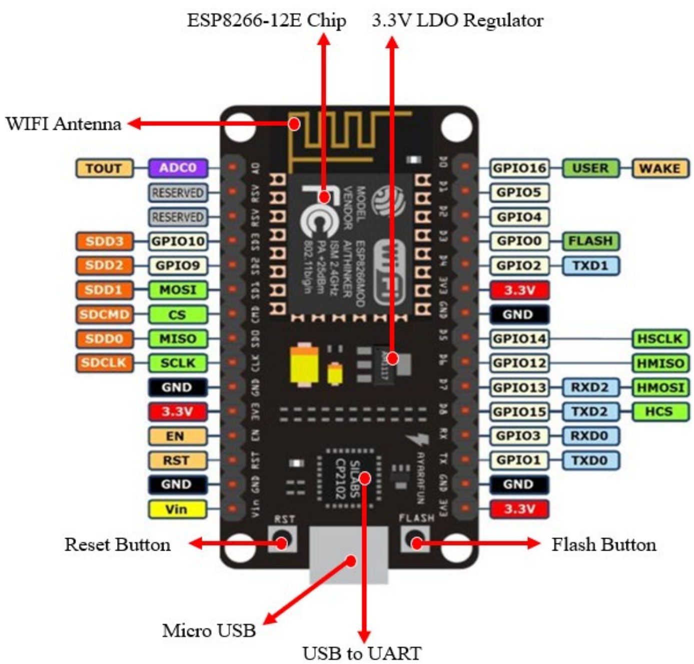


Figure 63: Pin Description of ESP8266 [63]

Specification [39]

- **CPU:** Tensilica Xtensa LX106 32-bit processor
- **Memory:** 160KB
- **Frequency Range:** 2.4 GHz ~ 2.5 GHz (2400 MHz ~ 2483.5 MHz)
- **Operating Voltage:** 2.5 V ~ 3.6 V
- **External flash:**512 KB to 1 MB (up to 16 MB is supported)
- **GPIO:** 16
- **Wi-Fi:** 802.11 b/g/n support, 802.11 n support (2.4 GHz), up to 72.2 Mbps
- **Peripheral Interface:** UART/SDIO/SPI/I2C/I2S/IR Remote Control
  GPIO/ADC/PWM/LED Light & Button
- **Operating Temperature Range:** –40 °C ~ 125 °C

11.1.1.3 ESP32 CAM

Figure 64 shows the ESP32 CAM module and Camera.

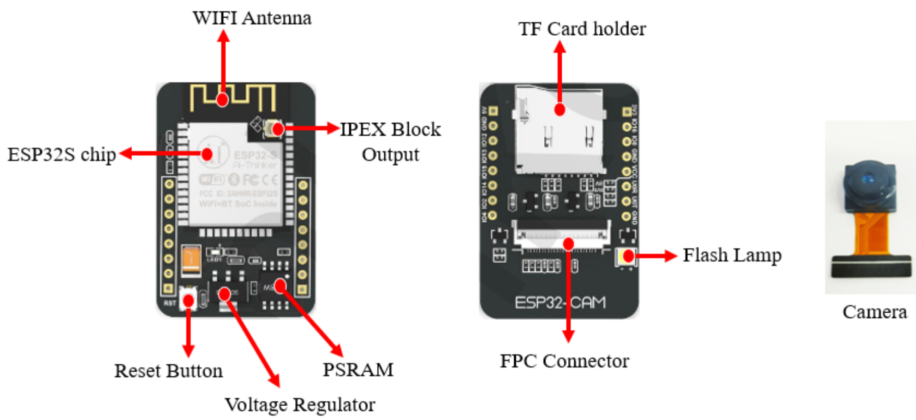


Figure 64: ESP32 CAM Module and Camera [40]

Figure 65 shows the pin description of ESP32 CAM development module.

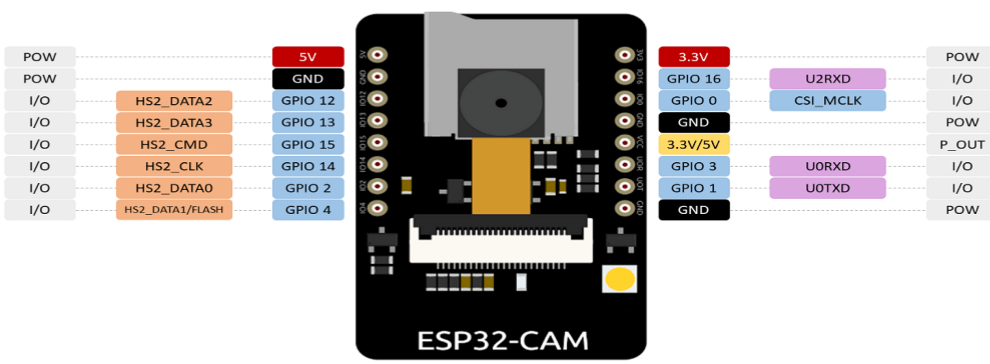


Figure 65: Pin description of ESP32 CAM [64]

Specification [40]

- **MCU:** ESP32
- **SPI Flash:** Default 32 Mbit
- **RAM:** 520KB SRAM+4M PSRAM
- **Wireless Connectivity: Wi-Fi:** 802.11 b/g/n

  Bluetooth: Bluetooth 4.2 BR/EDR and BLE standards
- **Support interface:** UART/SPI/I2C/PWM/ADC/DAC
- **GPIO:** 9

- **Power Supply Range:** 5V
- **Operating Temperature range:** –20 °C ~ 85 °C
- **Image Output format:** JPEG (OV2640 support only), BMP, GRAYSCALE

11.1.1.4 Fire Sensor

Figure 66 shows the actual Flame sensor used in the Designed Prototype. It has 4 pins VCC, GND, digital output, and analog Output. Some modules come with 3 pins, VCC, GND, and digital output. This module is based on the YG1006 sensor, which is an NPN phototransistor. It is coated with black epoxy, due to which it is sensitive to IR radiation. A 10K potentiometer is used to adjust its sensitivity. Red LED glows, indicating that the sensor is connected to the power supply. And when it detects Fire, its Green LED glows.

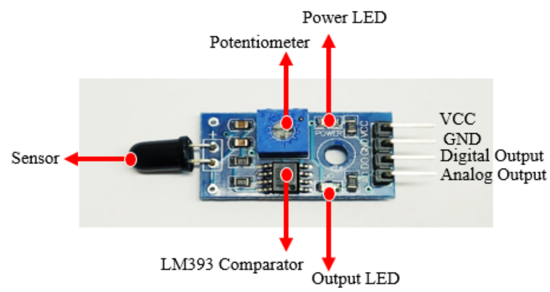


Figure 66: Flame Sensor [41]

Specifications [41]

- **Operating Voltage:** 3.3 V to 5 V
- **Digital Outputs:** 0 and 1
- Adjustable sensitivity via potentiometer
- **LED lights indicators:** Power (red) and digital switching output (green)

11.1.1.5 PIR Sensor

Figure 67 shows the PIR sensor from the top and bottom. This inexpensive PIR sensor comes with BISS0001 ("Micro Power PIR Motion Detector IC"), which is very inexpensive. This chip has certain features such as low powered, Bi-directional level detector, Excellent noise immunity, dual mode (retriggerable & non-retriggerable). Fresnel Lens on the top helps to condenses the light to provide a large range of IR to the sensor.

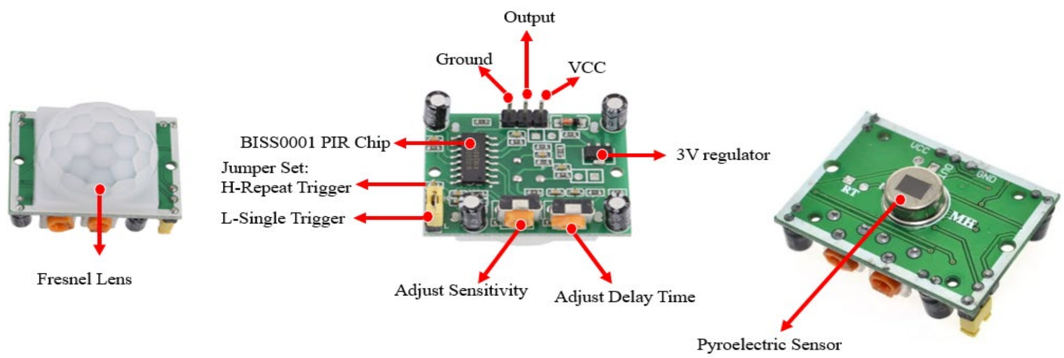


Figure 67: PIR Sensor [65]

Specifications [42]

- **Operating Voltage:** 5V-12V
- **Temperature:** – 15 to +70 °C
- **Output:** High: 3.3V, Low: 0V
- **Delay Time:** 5 - 300s (Adjustable)
- **Adjustable Trigger:** L: non-repeatable trigger, H: repeatable trigger
- **Sensing range:** less than 120 degree, within 7 meters

11.1.1.6 Magnetic Switch

Figure 68 shows the used Magnetic Switch in the Designed Prototype.

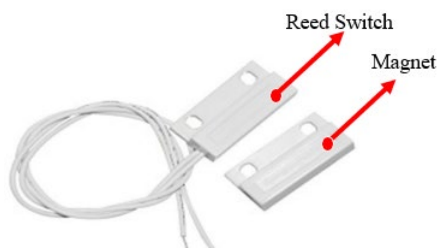


Figure 68: Magnetic Switch [66]

Specifications [66]

- **Maximum Current:** 0.5A
- **Maximum DC Voltage:** 100 V
- **Power:** 10W
- **Housing material:** ABS plastic

11.1.1.7 Ultrasonic Sensor

Figure 69 shows the Ultrasonic sensor used in the designed prototype.

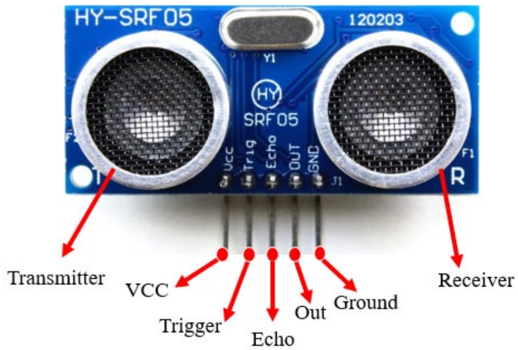


Figure 69: Ultrasonic Sensor [67]

Specifications [67]

- **Operating Voltage:** 5V DC
- **Measurement Angle:** Upto 15 degree
- **Ranging Distance:** 2cm – 400 cm
- **Sound Frequency:** 40 kHz

11.1.1.8 Microwave Sensor

Figure 70 shows the used Microwave sensor.

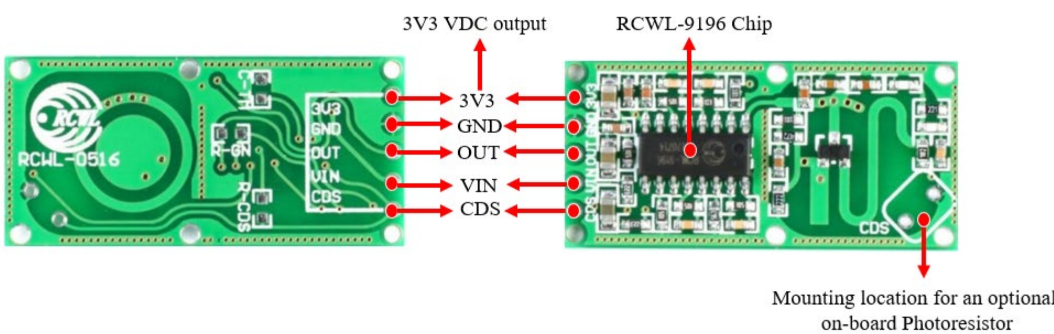


Figure 70: Microwave Doppler Radar RCWL-0516 [68]

Specifications [68]

- **Operating Voltage:** 4-28 VDC
- **Detection Range:** 5-9m

- **Frequency:** 3.2GHz
- **Operating Temperature:** – 20 to +80 °C
- **Output Timing:** 2sec Retrigger with motion

11.1.1.9 Shock Sensor

Figure 71 shows the used shock sensor in the designed system. It has 4 pins VCC, GND, digital output, and analog Output.

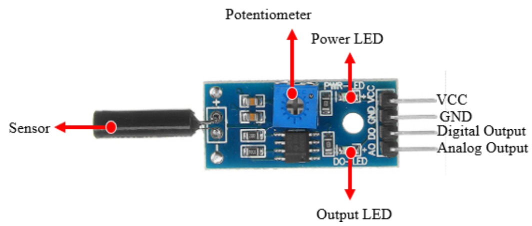


Figure 71: Shock Sensor [69]

Specifications:

- **Operating Voltage:** 3.3 V to 5 V
- **Digital Outputs:** 0 and 1
- Adjustable sensitivity via potentiometer
- **LED lights indicators:** Power (red) and digital switching output (green)

11.1.1.10 Buzzer

Figure 72 shows the Buzzer used in the Designed Prototype.

Figure 72: Buzzer [70]

Specifications [70]

- **Max Voltage:** 12 V
- Constant Tone

- **Oscillation Frequency**: 3.8kHz
- **Sound Pressure Level**: 95 dB

11.1.1.11 Servo Motor

Figure 73 shows the actual servo motor used in the designed prototype.

Figure 73: Servo Motor [71]

Specifications [71]

- Mini Micro Servo Motor 4.3 g
- **Servo Type:** Analog Servo
- **Operating voltage:** 4.8 - 7.2 V
- **Operating Current:** 100mA
- **Range of temperature: -**30 to +60°C

11.1.1.12 Keypad

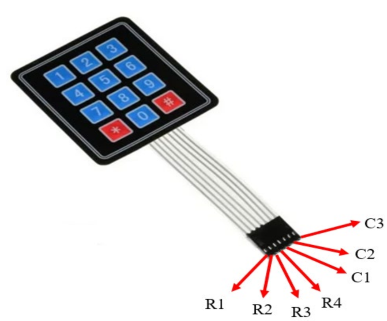


Figure 74: Used keypad in the designed system [47]

Specifications [72]

- Matrix 3x4 Keyboard
- **Number of keys:** 12
- **Contact resistance:** 500 Ohm
- **Insulation resistance:** 100M Ohm
- **Feedback:** 1 ms
- **Operating temperature:** -20 to +40 ° C

11.2 Appendix B

11.2.1 ESP8266 with Magnetic Switch and Shock Sensor for Window Security, Bedroom PIR Sensor and Living Room PIR Sensor

11.2.1.1 Magnetic Switch

11.2.1.1.1 When magnetic is in OFF state, and the system is in ON state

In **Figure 75,** magnetic button is in OFF state, and the system button is in ON state.

Labelled 1: It is the serial monitor of ESP32, it doesn't show any change, or we can say that it didn't receive any command as magnetic button is in OFF state.

Labelled 2: It is the serial monitor of ESP8266 with Magnetic Switch. It shows that when we switched OFF the magnetic button, we received the message indicating magnetic switch is OFF.

Labelled 3: It is the magnetic button on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as ON.

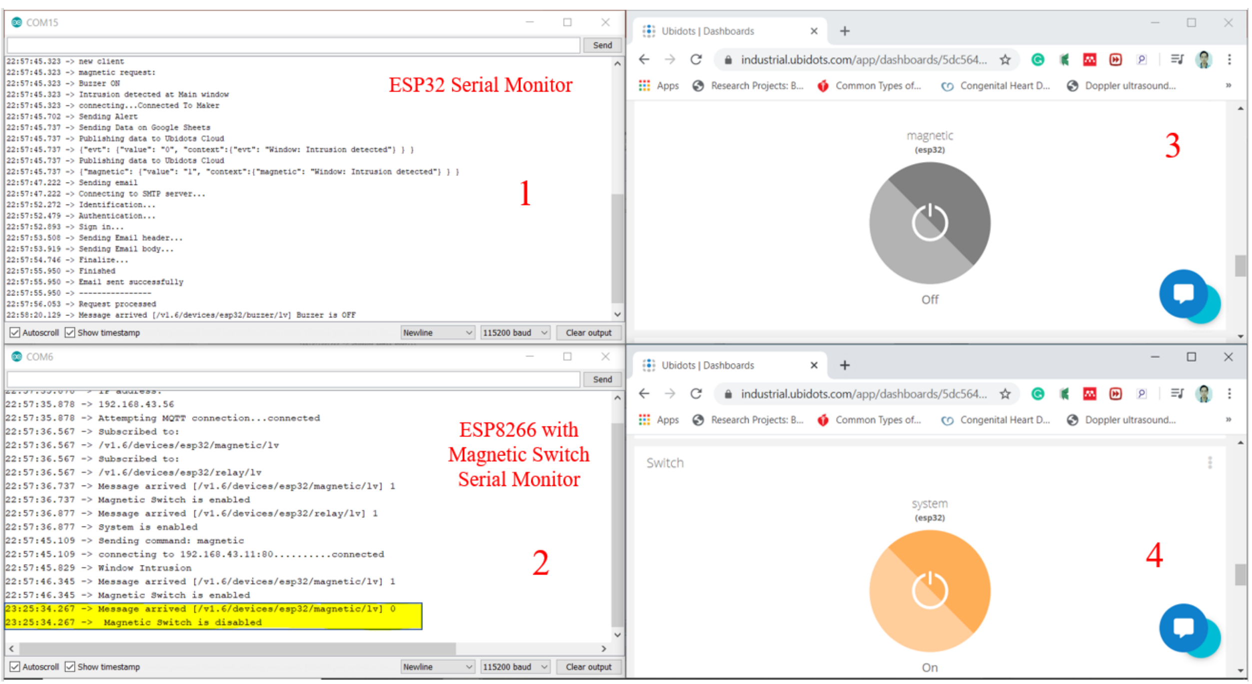


Figure 75: When magnetic button is in OFF state, and the system button is in ON state

11.2.1.1.2 When magnetic button is in ON state, and system button is in OFF state

In **Figure 76**, magnetic button is in ON state, and the system button is in OFF state.

Labelled 1: It is the serial monitor of ESP32. The highlighted part indicates that the system button is in OFF state. So it will not receive any command.

Labelled 2: It is the serial monitor of ESP8266 with Magnetic Switch. The highlighted part indicates that the magnetic button is in ON state, and the system button is in OFF state.

Labelled 3: It is the magnetic button on the Ubidots dashboard, which is shown as ON.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as OFF.

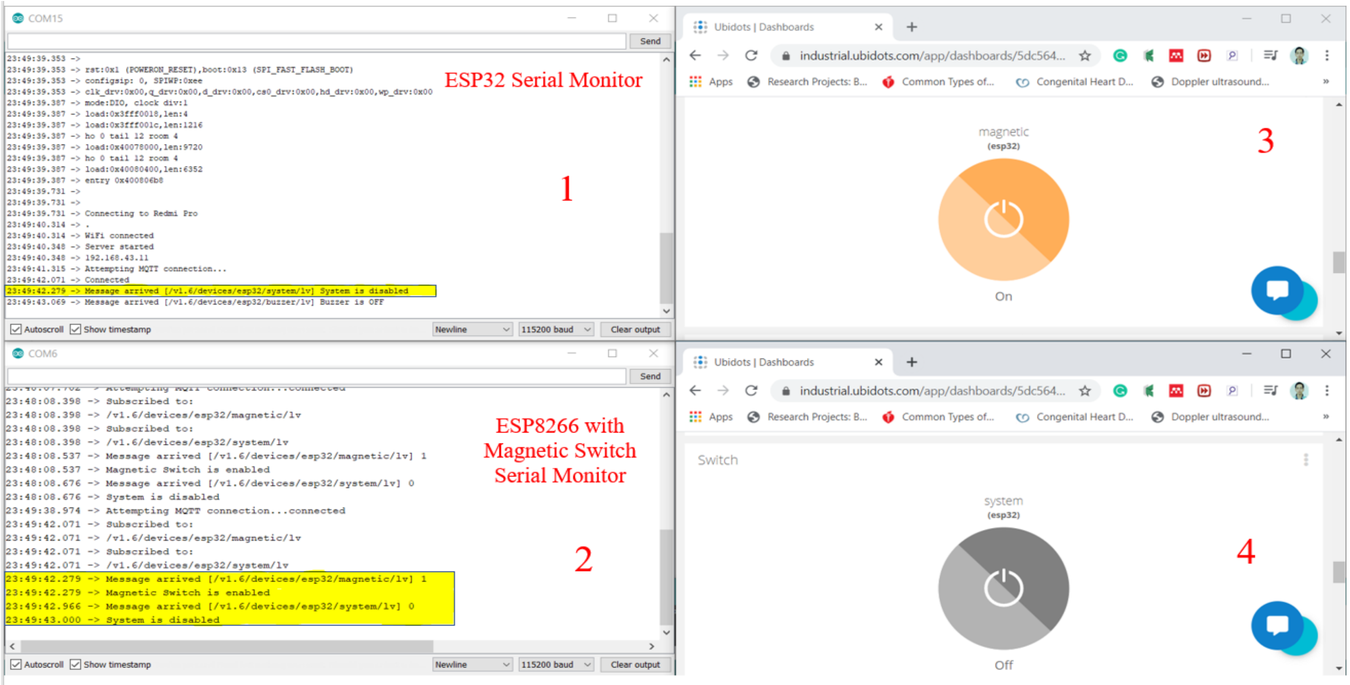


Figure 76: When magnetic is in ON state and system is in OFF state

11.2.1.1.3 When system button and magnetic button are in OFF state

In **Figure 77**, magnetic button is in OFF state, and the system button is in OFF state.

Labelled 1: It is the serial monitor of ESP32. The highlighted part indicates that the system button is in OFF state.

Labelled 2: It is the serial monitor of ESP8266 with Magnetic Switch. The highlighted part indicates that magnetic button and the system button are in OFF state.

Labelled 3: It is the magnetic button on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as OFF.

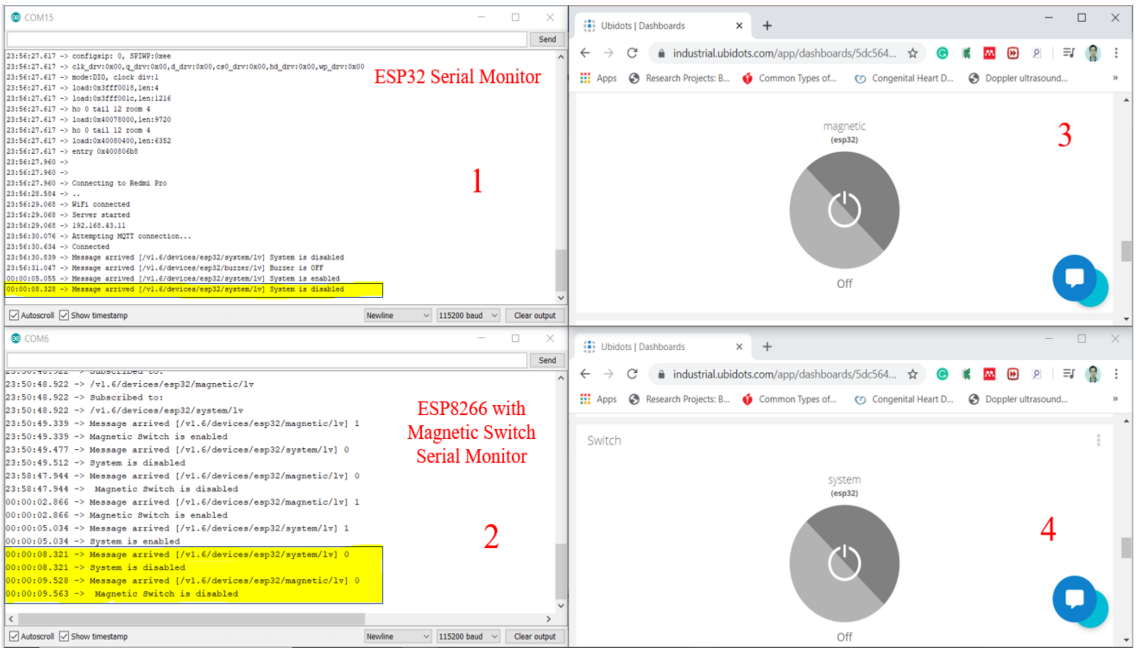


Figure 77: When system and magnetic are in OFF state

11.2.1.2 Shock Sensor

11.2.1.2.1 When shock button is in OFF state, and the system is in ON state

In **Figure 75,** shock button is in OFF state, and the system button is in ON state.

Labelled 1: It is the serial monitor of ESP32, it doesn't show any change, or we can say that it didn't receive any command. Because shock button is in OFF state.

Labelled 2: It is the serial monitor of ESP8266 with Shock Sensor. It shows that when we switched OFF the shock button, we received the message indicating shock sensor is OFF.

Labelled 3: It is the shock button on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as ON.

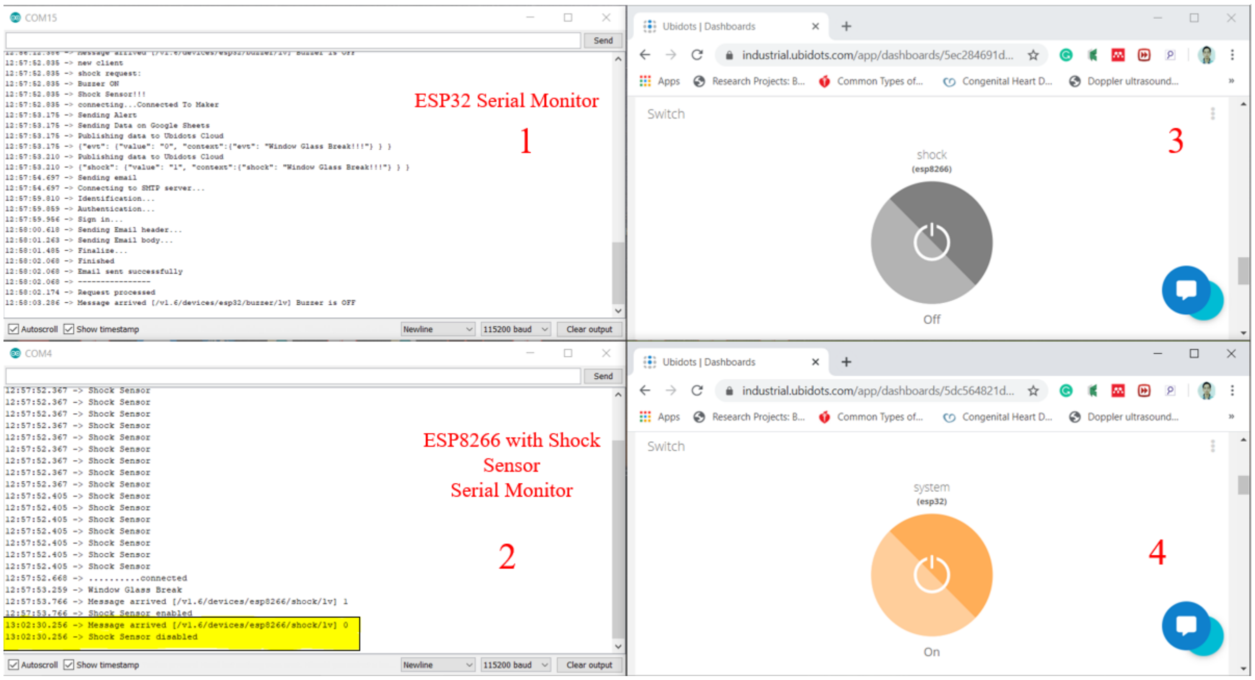


Figure 78: When shock is in OFF state, and the system is in ON state

11.2.1.2.2 When shock button is in ON state, and the system button is in OFF state

In **Figure 79**, shock button is in ON state, and the system button is in OFF state.

Labelled 1: It is the serial monitor of ESP32. The highlighted part indicates that the system button is in OFF state. So, it will not receive any command.

Labelled 2: It is the serial monitor of ESP8266 with Shock Sensor. The highlighted part indicates that the shock button is in ON state, and the system button is in OFF state.

Labelled 3: It is the shock button on the Ubidots dashboard, which is shown as ON.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as OFF.

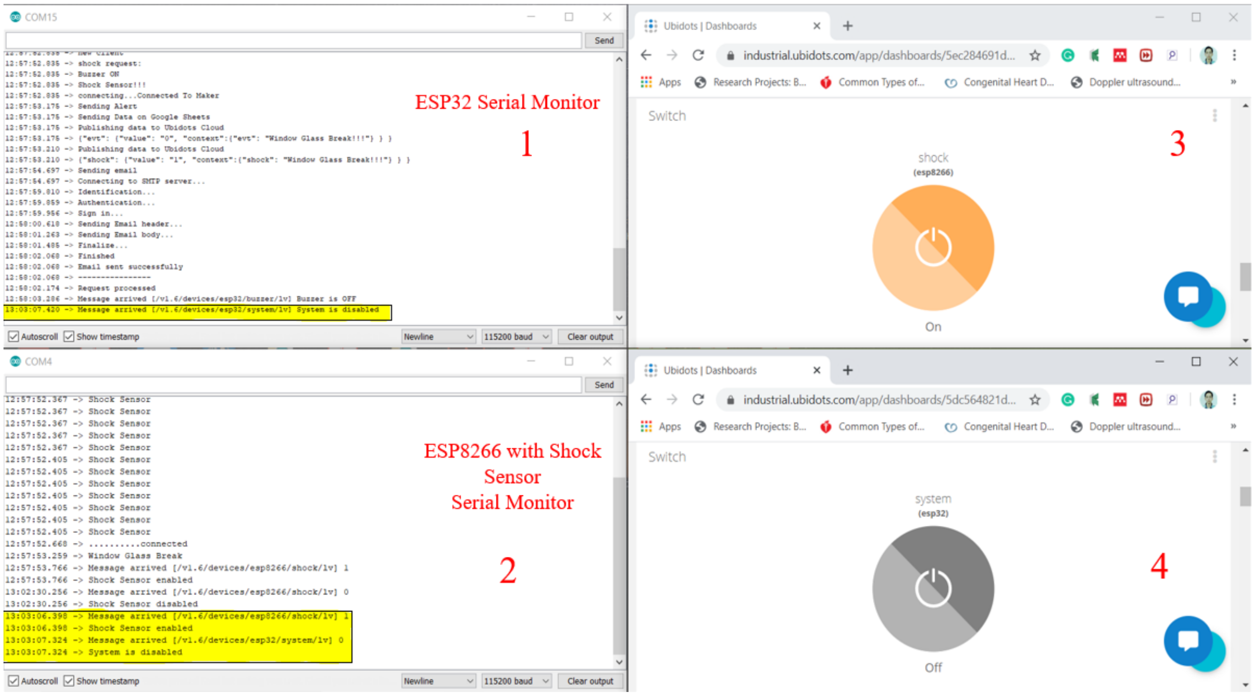


Figure 79: When shock button is in ON state, and the system button is in OFF state

11.2.1.2.3 When shock and system button is in OFF state

In **Figure 80**, shock and system button are in OFF state.

Labelled 1: It is the serial monitor of ESP32. The highlighted part indicates that the system button is in OFF state.

Labelled 2: It is the serial monitor of ESP8266 with Shock Sensor. The highlighted part indicates that shock and system button is in OFF state.

Labelled 3: It is the shock button on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as OFF.

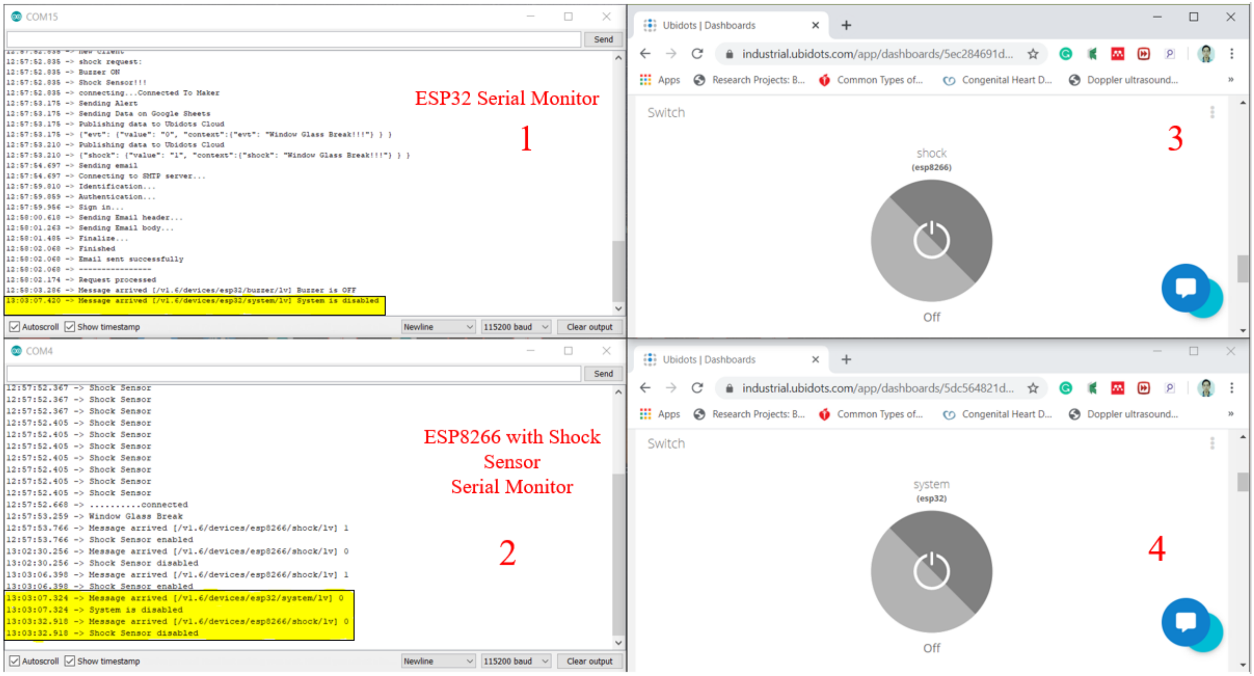


Figure 80: When shock and system button is in OFF state

11.2.1.3 Bedroom PIR sensor

11.2.1.3.1 When pir_bedroom button is in OFF state, and the system button is in ON state.

In **Figure 75,** pir_bedroom button is in OFF state, and the system button is in ON state.

Labelled 1: It is the serial monitor of ESP32, it doesn't show any change, or we can say that it didn't receive any command. Because pir_bedroom button is in OFF state.

Labelled 2: It is the serial monitor of ESP8266 with Bedroom PIR Sensor. It shows that when we switched OFF the pir_bedroom button, we received the message indicating Bedroom PIR sensor is OFF.

Labelled 3: It is the pir_bedroom button on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as ON.

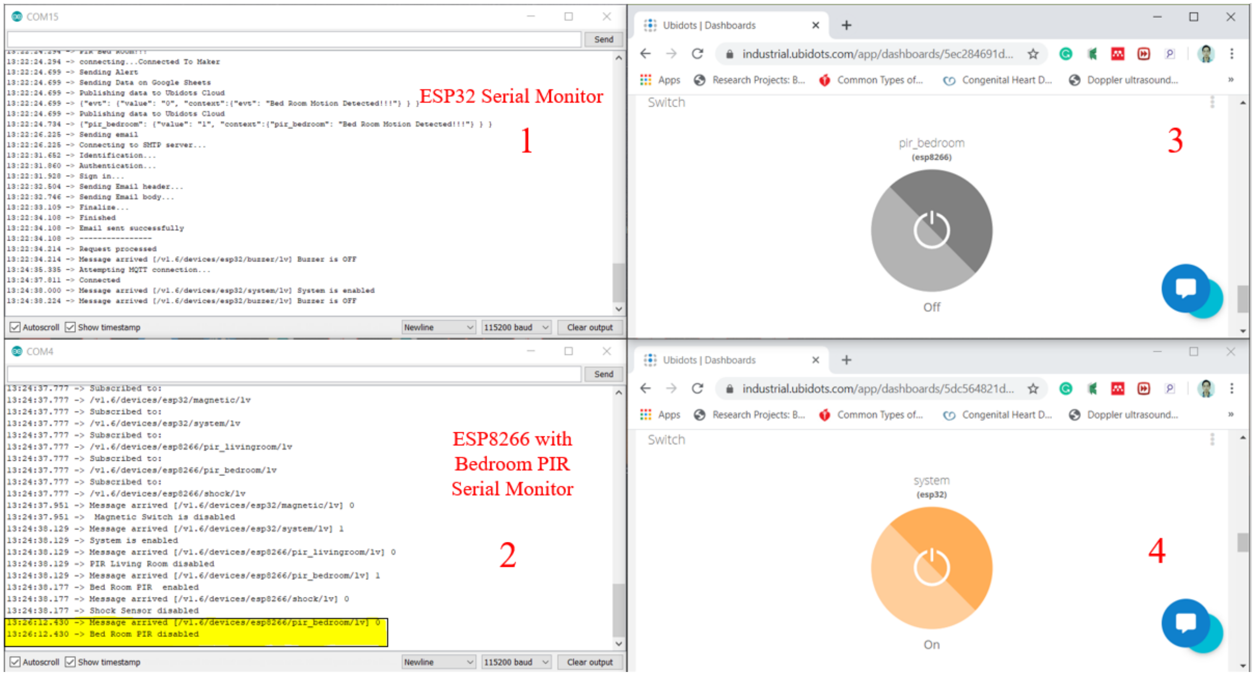


Figure 81: When pir_bedroom button is in OFF state, and the system button is in ON state

11.2.1.3.2 When pir_bedroom button is in ON state, and the system button is in OFF state

In **Figure 82**, pir_bedroom button is in ON state, and the system button is in OFF state.

Labelled 1: It is the serial monitor of ESP32. The highlighted part indicates that the system button is in OFF state. So, it will not receive any command.

Labelled 2: It is the serial monitor of ESP8266 with Bedroom PIR Sensor. The highlighted part indicates that the pir_bedroom button is in ON state, and the system button is in OFF state.

Labelled 3: It is the pir_bedroom button on the Ubidots dashboard, which is shown as ON.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as OFF.

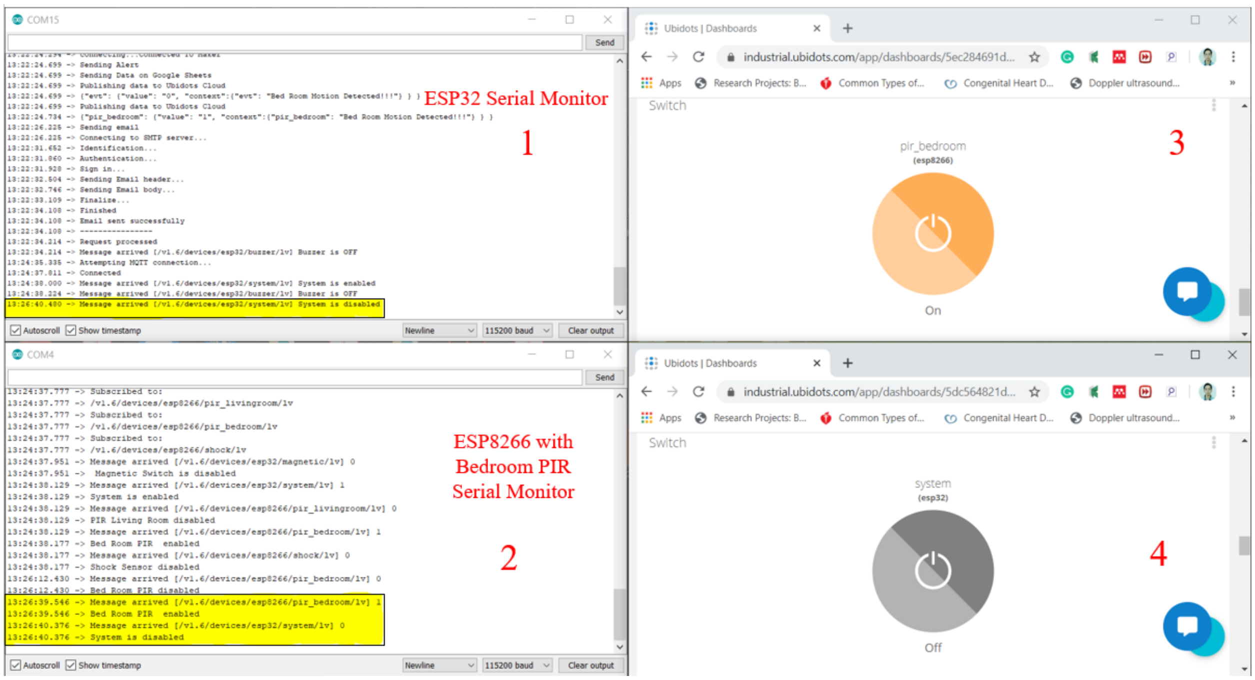


Figure 82: When pir_bedroom button is in ON state, and the system button is in OFF state.

11.2.1.3.3 When pir_bedroom and system button are in OFF state.

In **Figure 83**, pir_bedroom and system button are in OFF state.

Labelled 1: It is the serial monitor of ESP32. The highlighted part indicates that the system button is in OFF state.

Labelled 2: It is the serial monitor of ESP8266 with Bedroom PIR Sensor. The highlighted part indicates that pir_bedroom and system button is in OFF state.

Labelled 3: It is the pir_bedroom button on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as OFF.

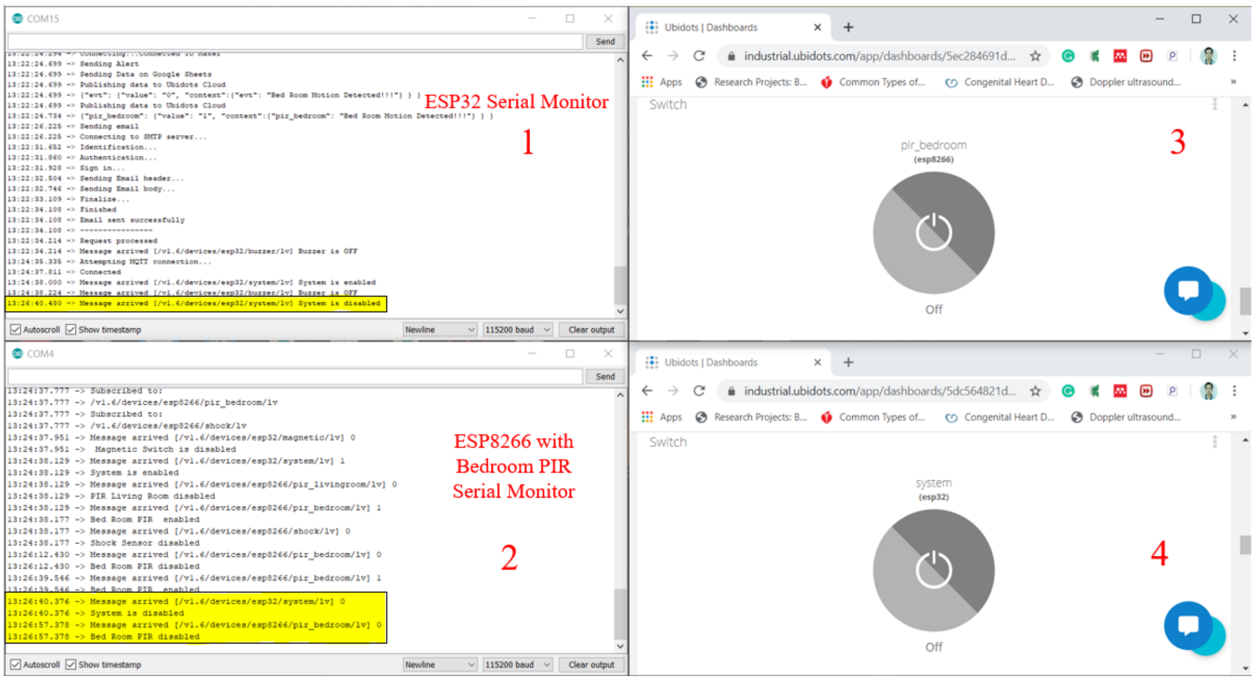


Figure 83: When pir_bedroom and system button are in OFF state.

11.2.1.4 Living Room PIR sensor

11.2.1.4.1 When pir_livingroom button is in OFF state, and the system button is in ON state

In **Figure 84,** pir_livingroom button is in OFF state, and the system button is in ON state.

Labelled 1: It is the serial monitor of ESP32, it doesn't show any change, or we can say that it didn't receive any command. Because pir_livingroom button is in OFF state.

Labelled 2: It is the serial monitor of ESP8266 with Living Room PIR Sensor. It shows that when we switched OFF the pir_livingroom button, we received the message indicating Living PIR sensor is OFF.

Labelled 3: It is the pir_livingroom button on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as ON.

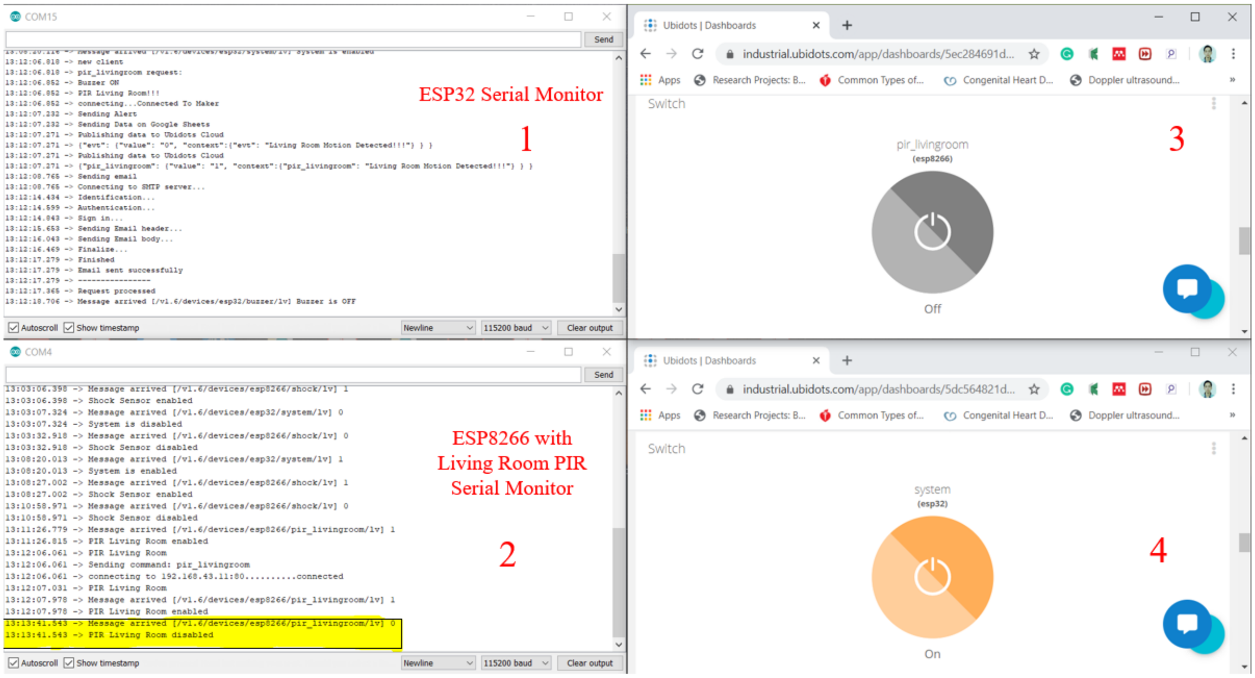


Figure 84: When pir_livingroom button is in OFF state, and the system button is in ON state

11.2.1.4.2 When pir_livingroom button is in ON state, and the system button is in OFF state

In **Figure 85**, pir_livingroom button is in ON state, and the system button is in OFF state.

Labelled 1: It is the serial monitor of ESP32. The highlighted part indicates that the system button is in OFF state. So, it will not receive any command.

Labelled 2: It is the serial monitor of ESP8266 with Living Room PIR Sensor. The highlighted part indicates that the pir_livingroom button is in ON state, and the system button is in OFF state.

Labelled 3: It is the pir_livingroom button on the Ubidots dashboard, which is shown as ON.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as OFF.

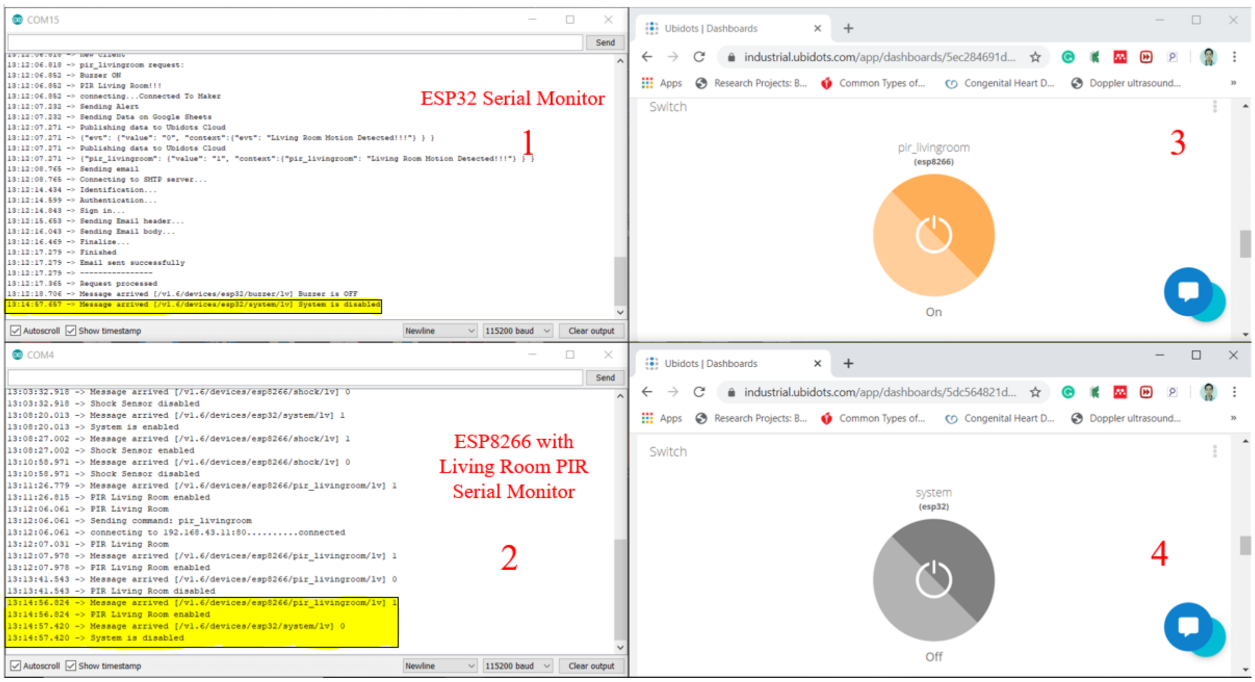


Figure 85: When pir_livingroom button is in ON state, and the system button is in OFF state

11.2.1.4.3 When pir_livingroom and system button are in OFF state

In **Figure 86**, pir_livingroom and system button are in OFF state.

Labelled 1: It is the serial monitor of ESP32. The highlighted part indicates that the system button is in OFF state.

Labelled 2: It is the serial monitor of ESP8266 with Living Room PIR Sensor. The highlighted part indicates that pir_livingroom and system button is in OFF state.

Labelled 3: It is the pir_livingroom button on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as OFF.

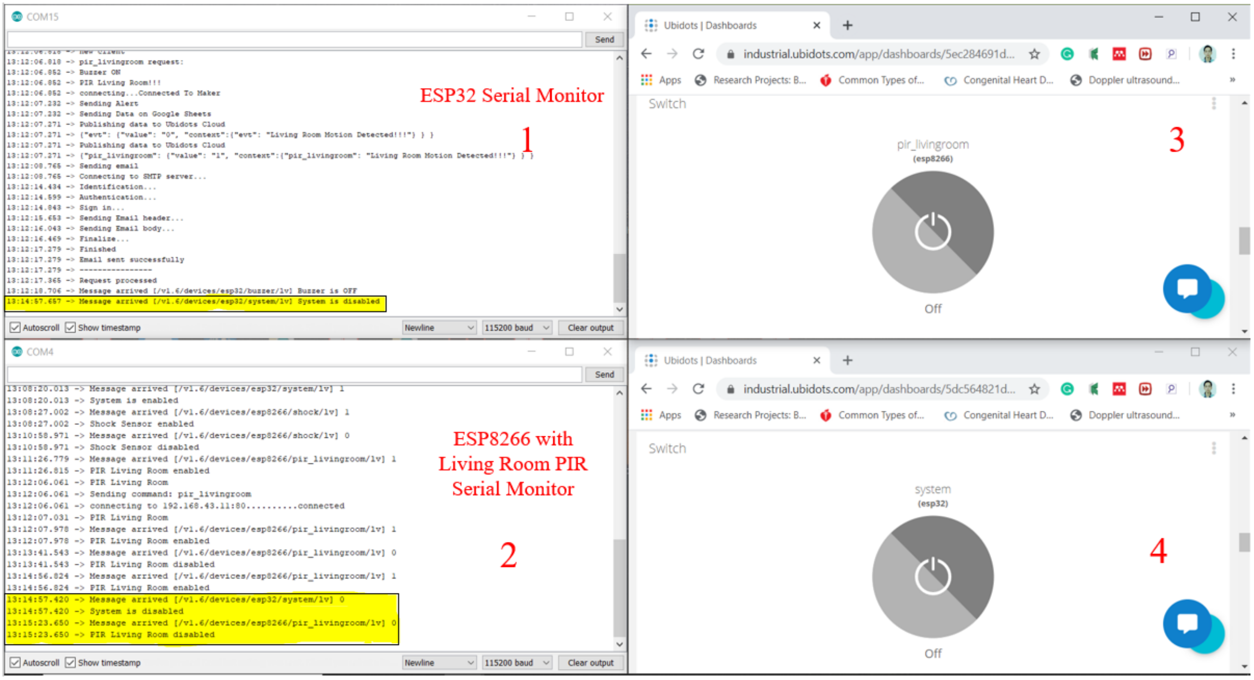


Figure 86: When pir_livingroom and system button are in OFF state

11.2.2 ESP8266 with Corridor PIR Sensor and Fire Sensor

11.2.2.1 When corridor PIR, system button are in ON state, and fire button is in OFF state

In **Figure 87,** fire button is in OFF state, and both Corridor PIR and system button are in ON state.

Labelled 1: It is the serial monitor of ESP32. It will not get any request from the fire sensor, because it is OFF, but can get a request from corridor PIR as its button is in ON state.

Labelled 2: It is the serial monitor of ESP8266. The highlighted part shows that the fire sensor is switched OFF.

Labelled 3: It is the corridor PIR button on the Ubidots dashboard, which is shown as ON.

Labelled 4 : It is the fire button on the Ubidots dashboard, which is shown as OFF.

Labelled 5: It is the system button on the Ubidots dashboard, which is shown as ON.

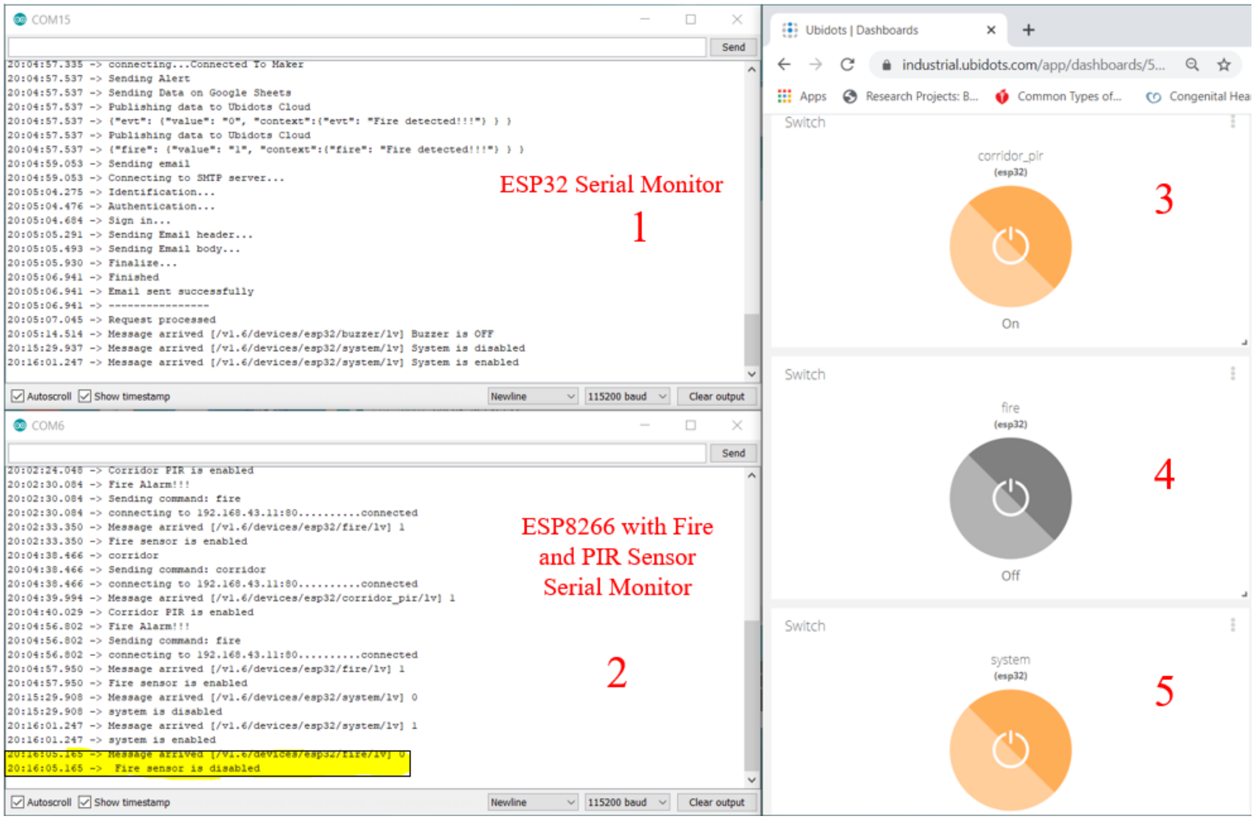


Figure 87: When corridor PIR, system button are in ON state and fire button is in OFF state

11.2.2.2 When fire, system button is in ON state, and Corridor PIR button is in OFF state

In **Figure 88**, both fire and system button is in ON state, and Corridor PIR button are in OFF state.

Labelled 1: It is the serial monitor of ESP32. It will not get any request from corridor PIR, because it is in OFF state but can get a request from the fire sensor as the fire button is in ON state.

Labelled 2: It is the serial monitor of ESP8266. The highlighted part shows that the corridor PIR sensor is switched OFF.

Labelled 3: It is the corridor PIR button on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the fire button on the Ubidots dashboard, which is shown as ON.

Labelled 5: It is the system button on the Ubidots dashboard, which is shown as ON.

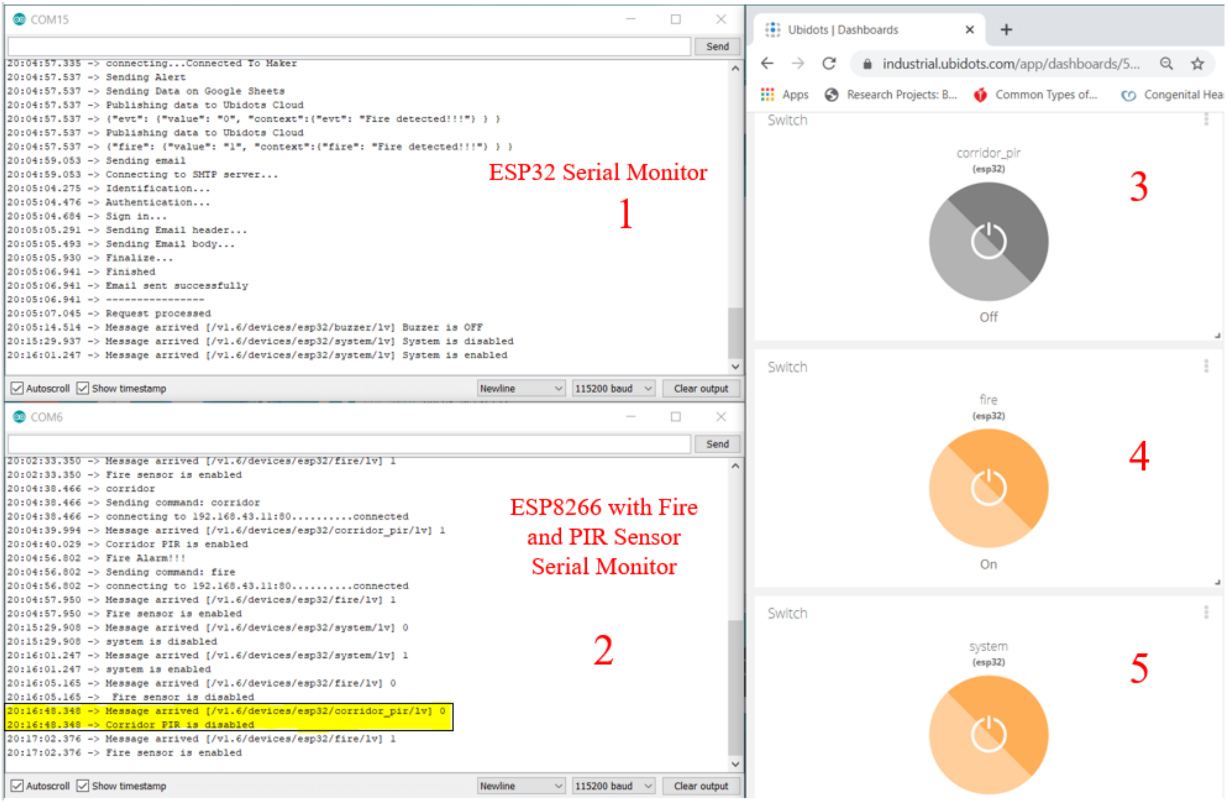


Figure 88: When fire, system button are in ON and Corridor PIR button is in OFF state

11.2.2.3 When corridor PIR, fire button are in ON state and system button is in OFF state

In **Figure 89**, system button is in OFF state, and both Corridor PIR and fire button are in ON state.

Labelled 1: It is the serial monitor of ESP32. It will not get any request neither from the fire sensor nor from corridor PIR because system button is in OFF state.

Labelled 2: It is the serial monitor of ESP8266. The highlighted part shows that the system button is switched OFF.

Labelled 3: It is the corridor PIR button on the Ubidots dashboard, which is shown as ON.

Labelled 4: It is the fire button on the Ubidots dashboard, which is shown as ON.

Labelled 5: It is the system button on the Ubidots dashboard, which is shown as OFF.

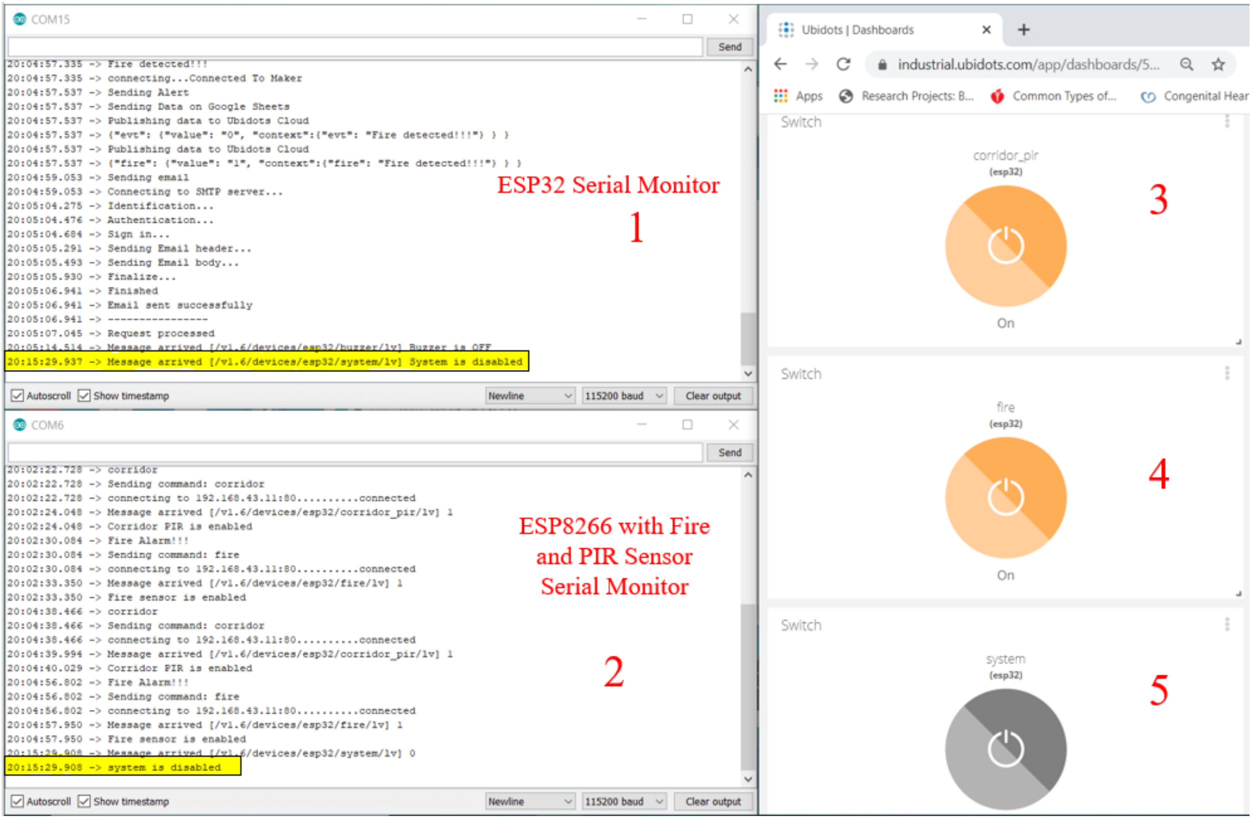


Figure 89: When corridor PIR, fire button are in ON state and system button is in OFF state

11.2.2.4 When corridor PIR, fire button and system button is OFF

In **Figure 90**, corridor PIR, fire button, and system button are in OFF state.

Labelled 1: It is the serial monitor of ESP32. It will not get any request from any sensor as full system is OFF, and individual sensors are also switched OFF.

Labelled 2: It is the serial monitor of ESP8266. The highlighted part shows that all three buttons are in OFF state.

Labelled 3: It is the corridor PIR button on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the fire button on the Ubidots dashboard, which is shown as OFF.

Labelled 5: It is the system button on the Ubidots dashboard, which is shown as OFF.

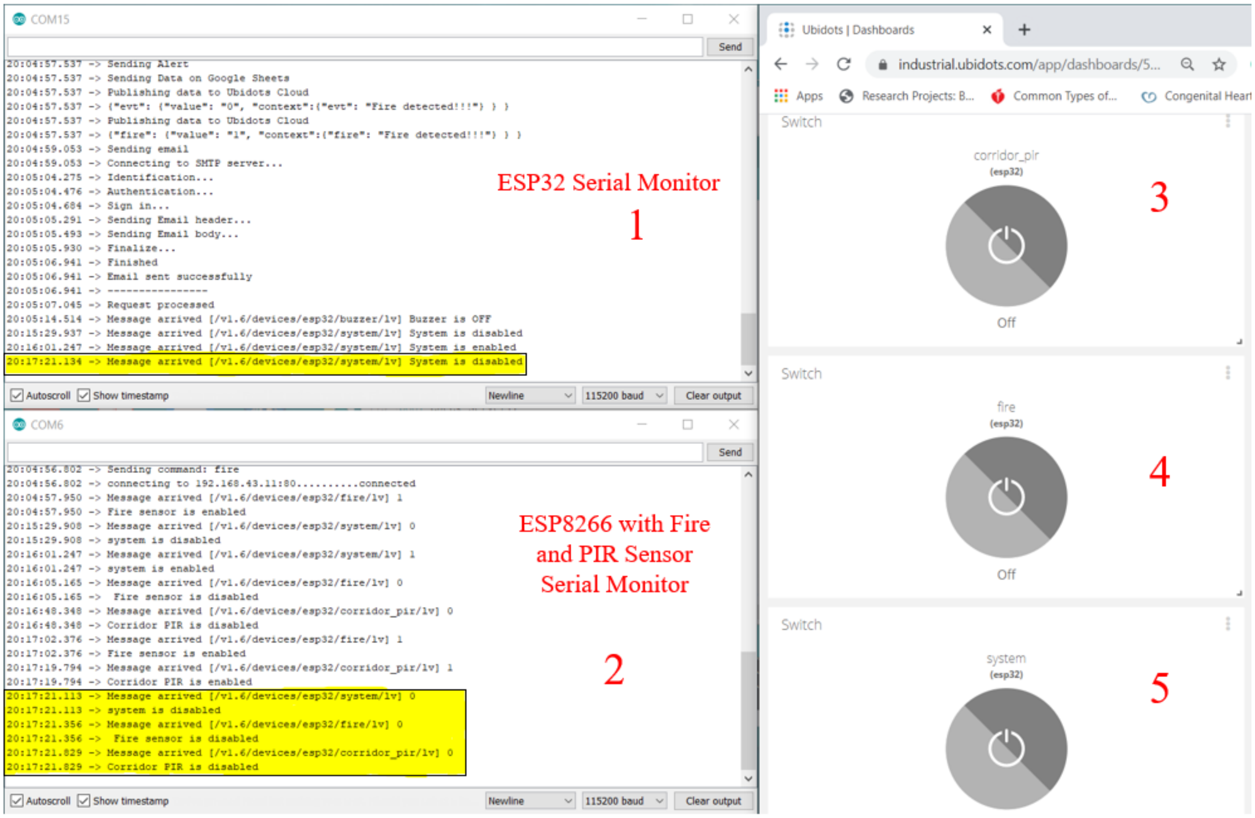


Figure 90: When corridor PIR, fire button and system button are in OFF state

11.2.3 ESP8266 with Ultrasonic Sensor

11.2.3.1 When ultrasonic button is in OFF state, and the system button is in ON state

Figure 91 ultrasonic button is in OFF state, and the system button is in ON state.

Labelled 1: It is the serial monitor of ESP32, it doesn't show any change, or we can say that it didn't receive any command as the ultrasonic button is in OFF state.

Labelled 2: It is the serial monitor of ESP8266 with an ultrasonic sensor. It shows that when we switched OFF the ultrasonic button, we received the message indicating the Garage sensor is OFF.

Labelled 3: It is the ultrasonic button on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as ON.

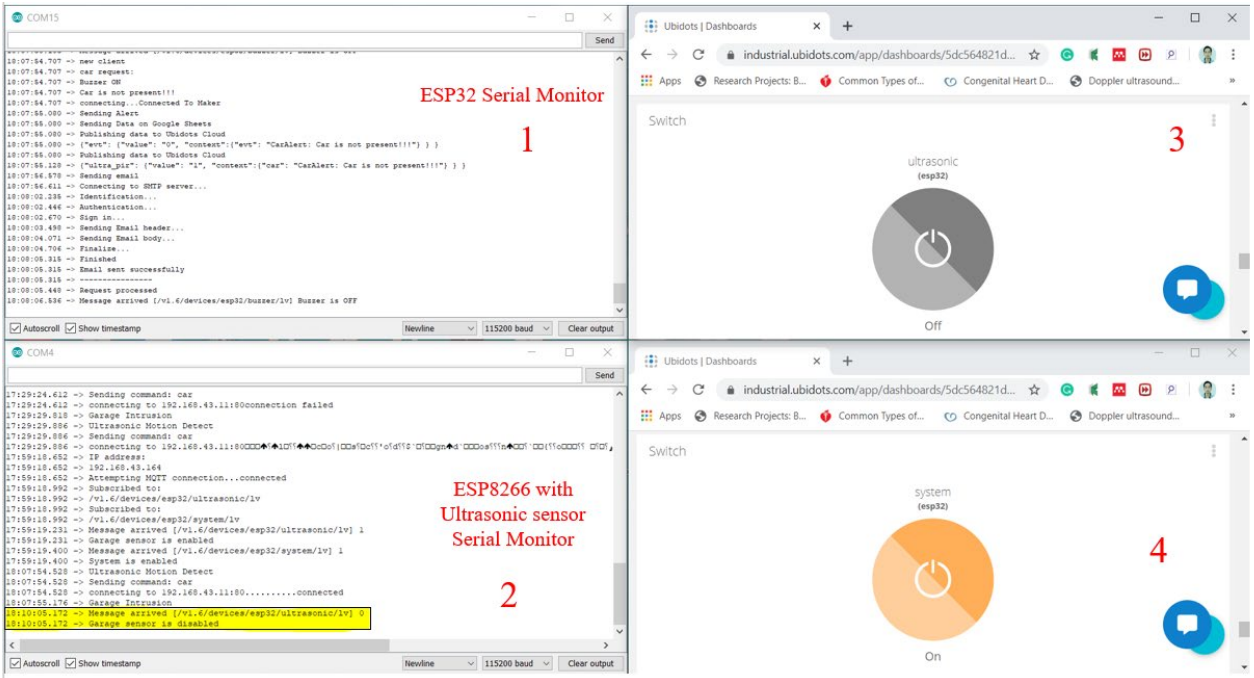


Figure 91: When ultrasonic button is in OFF state, and the system button is in ON state

11.2.3.2 When ultrasonic button is in ON state, and system button is in OFF state

In **Figure 92** ultrasonic button is in ON state and system button is in OFF state.

Labelled 1: It is the serial monitor of ESP32. The highlighted part indicates that system button is in OFF state, so it will not detect any intrusion.

Labelled 2: It is the serial monitor of ESP8266 with ultrasonic sensor. The highlighted part shows that the ultrasonic sensor is ON, and the System is OFF.

Labelled 3: It is the ultrasonic button on the Ubidots dashboard, which is shown as ON.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as OFF.

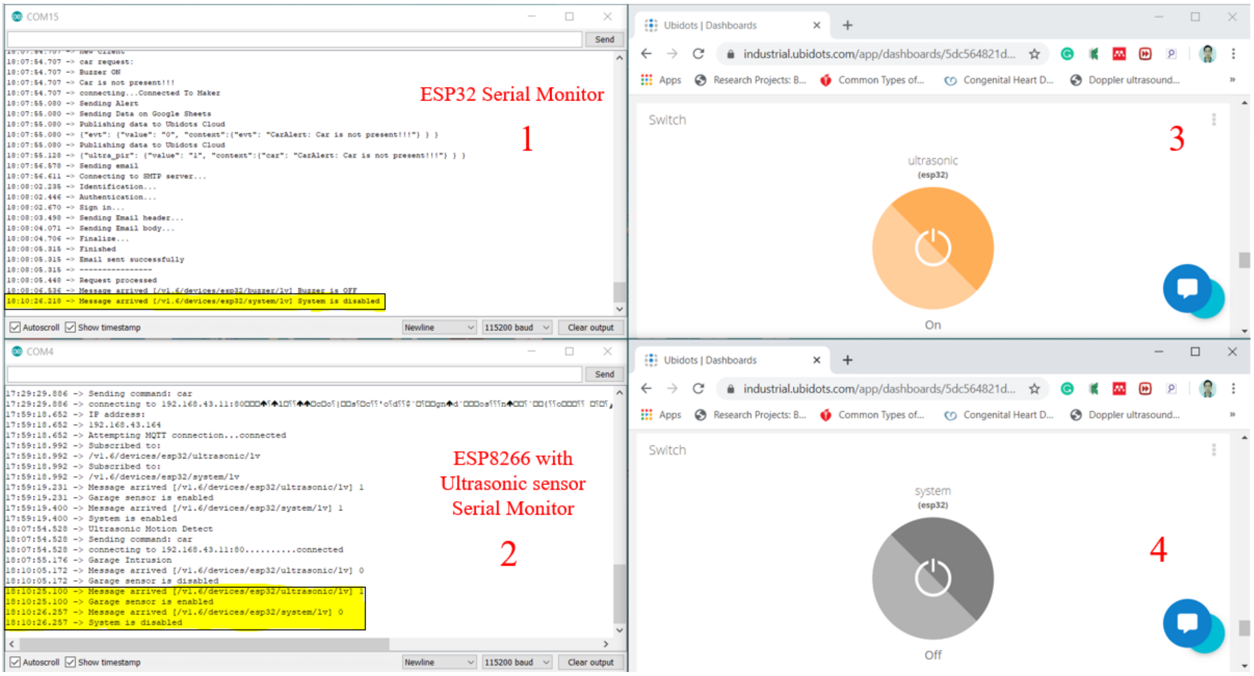


Figure 92: When ultrasonic button is in ON state, and system button is in OFF state

11.2.3.3 When both ultrasonic button and system button is in OFF state

In **Figure 93**, both ultrasonic button and system button is in OFF state.

Labelled 1: It is the serial monitor of ESP32. The highlighted part indicates that the system is OFF, so it will not detect any intrusion.

Labelled 2: It is the serial monitor of ESP8266 with ultrasonic sensor. The highlighted part indicates that the ultrasonic and system button is in OFF state, so it will not detect any intrusion.

Labelled 3: It is the ultrasonic button on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as OFF.

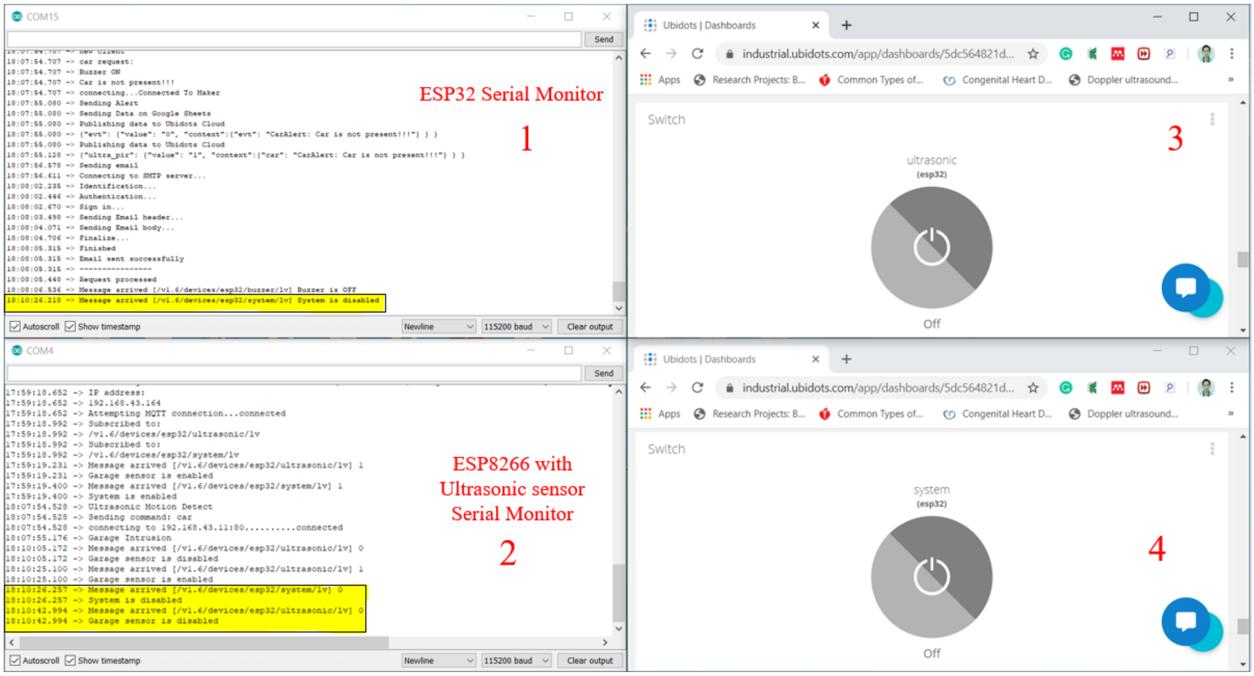


Figure 93: When both ultrasonic button and system button is in OFF state

11.2.4 ESP8266 with Keypad and Motor

11.2.4.1 When System switch is in OFF state, and Keypad switch is in ON state

In **Figure 94**, system is in ON state, and the keypad is in OFF state.

Labelled 1: It is the serial monitor of ESP32. It shows the system is in ON state. And no command received even after entering the wrong password, because keypad is OFF and during that keypad will not take any input.

Labelled 2: It is the serial monitor of ESP8266 with the keypad. The highlighted part indicates that the system is ON, and the keypad is OFF. In the OFF condition, it will not take any input.

Labelled 3: It is the system switch on the Ubidots dashboard, which is shown as ON.

Labelled 4: It is the keypad switch on the Ubidots dashboard, which is shown as OFF.

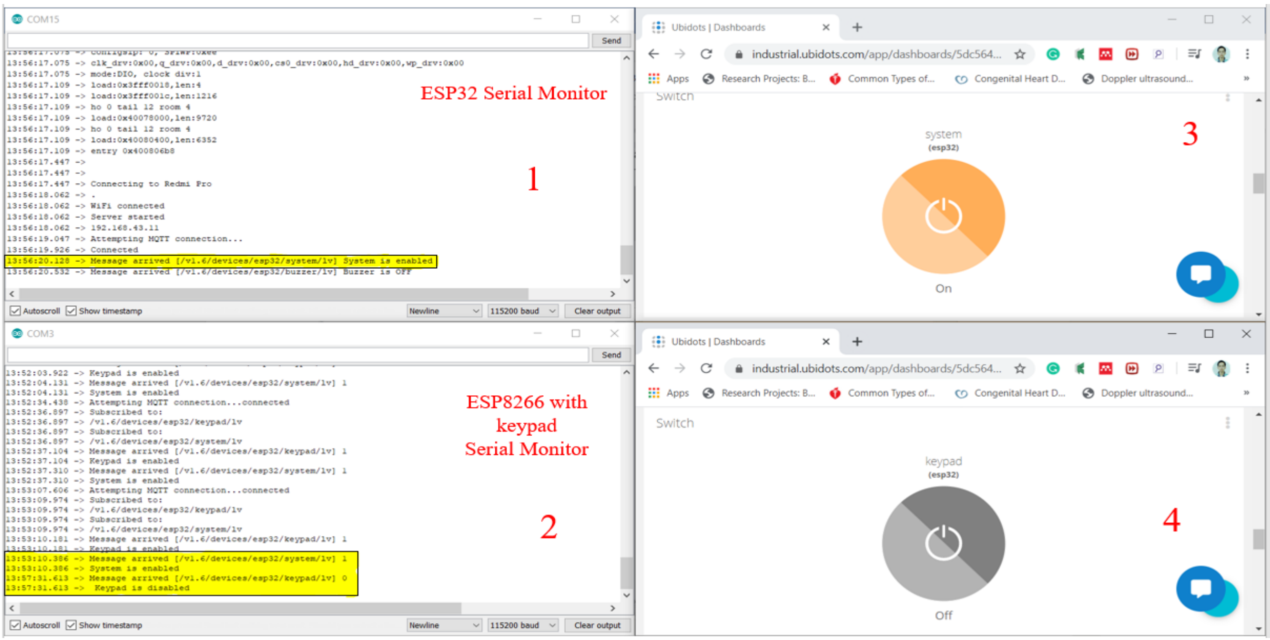


Figure 94: When system is in ON state, and keypad is in OFF state

11.2.4.2 When System switch is in OFF state, and Keypad switch is in ON state

In **Figure 95**, the system is in OFF state, and the keypad is in ON state.

Labelled 1: It is the serial monitor of ESP32. It shows the system is in OFF state. And no command received even after entering the wrong password, because it is OFF and during that keypad will not take any input.

Labelled 2: It is the serial monitor of ESP8266 with the keypad. The highlighted part indicates that the system is OFF, and the keypad is ON. But still, it will not take any input because system is OFF.

Labelled 3: It is the system switch on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the keypad switch on the Ubidots dashboard, which is shown as ON.

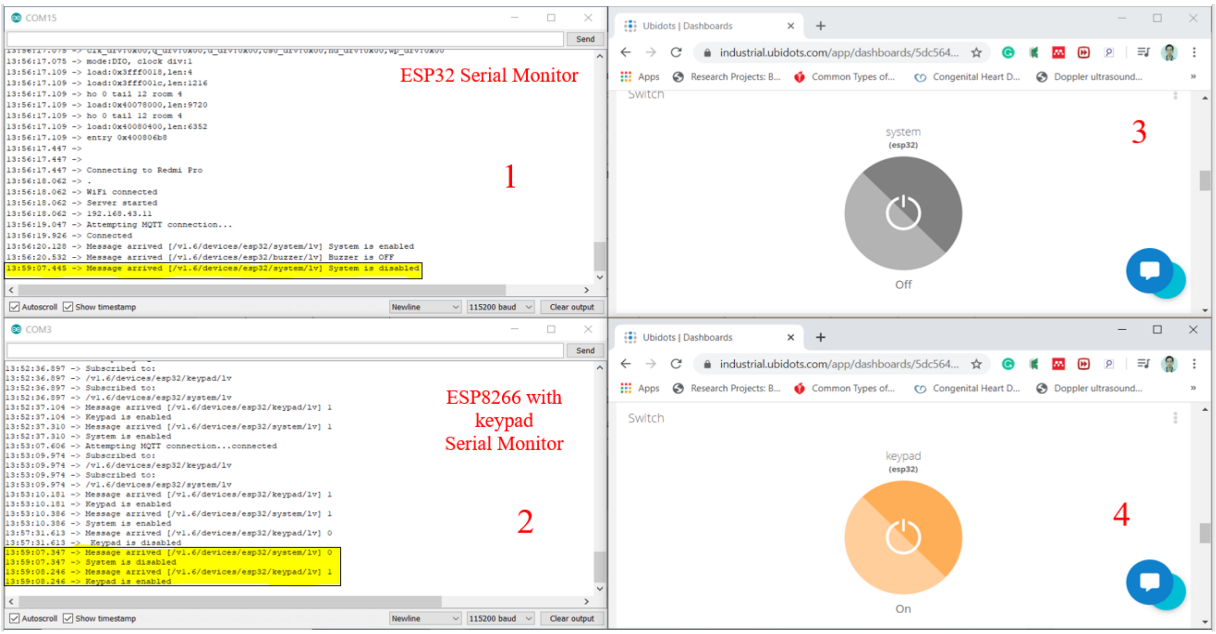


Figure 95: When system is in OFF state, and keypad is in ON state

11.2.4.3 When both System switch and Keypad switch are in OFF state

In **Figure 96**, both system and the keypad are in OFF state.

Labelled 1: It is the serial monitor of ESP32. It shows the system is in OFF state. And no command received even after entering the wrong password, because both system and keypad are OFF.

Labelled 2: It is the serial monitor of ESP8266 with the keypad. The highlighted part indicates system and keypad is in OFF state. It will not take any input because both system and keypad are in OFF state.

Labelled 3: It is the system switch on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the keypad switch on the Ubidots dashboard, which is shown as OFF.

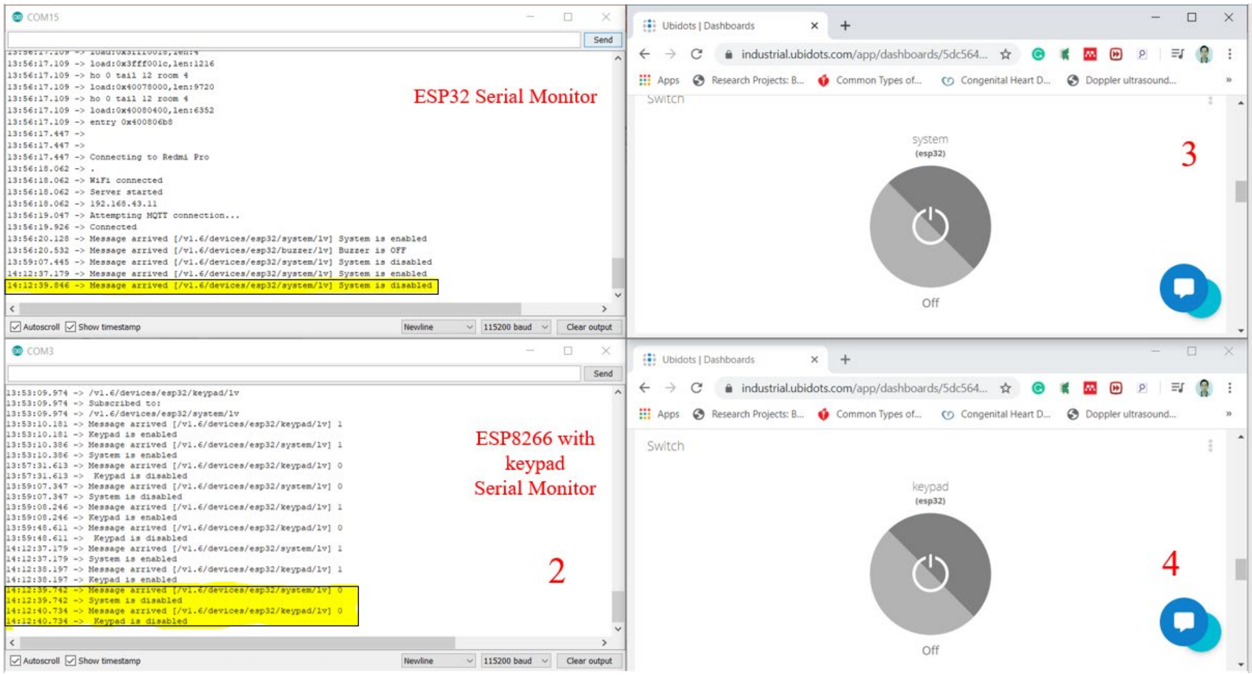


Figure 96: When both system and keypad are in OFF state

11.2.5 ESP32 CAM with Microwave sensor and PIR sensor

11.2.5.1 When system is in ON state, and pir is in OFF state

In **Figure 97**, the system is in ON state, and the keypad is in OFF state.

Labelled 1: It is the serial monitor of ESP32. No command received even after detecting motion because pir button is in OFF state.

Labelled 2: It is the serial monitor of ESP32 CAM. The highlighted part indicates that the pir button is in OFF state. In the OFF condition, it will not detect motion.

Labelled 3: It is the pir button on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as ON.

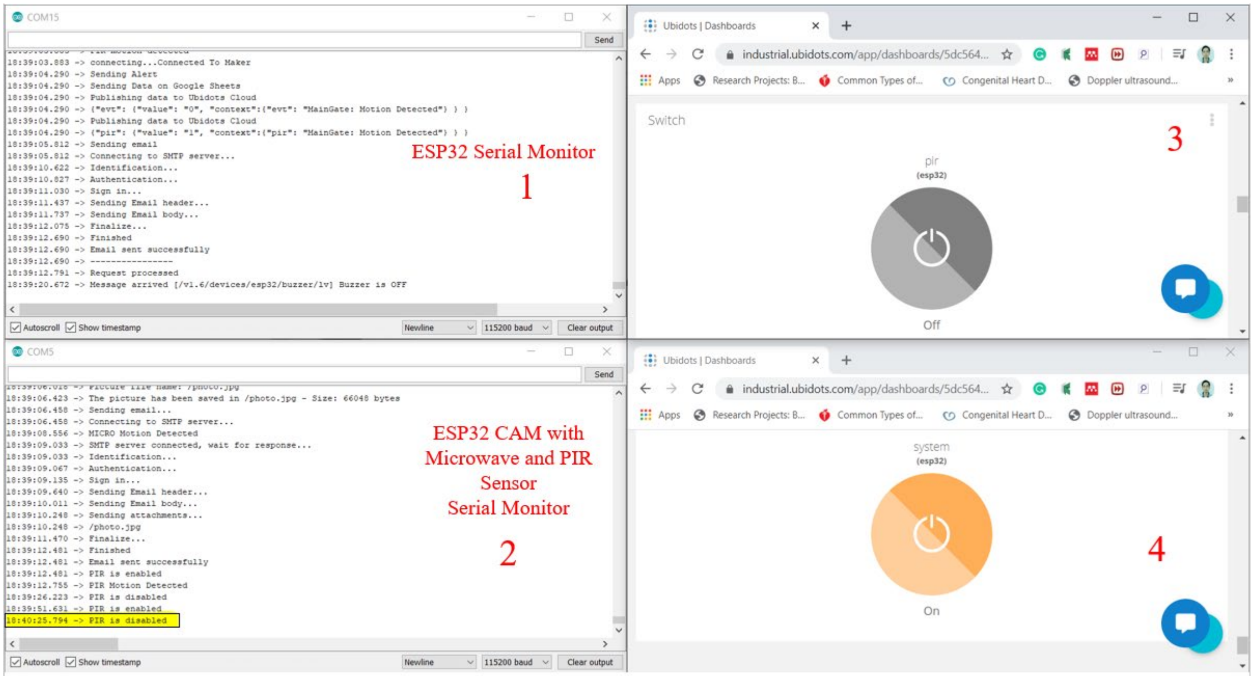


Figure 97: When system is in ON state and pir is in OFF state

11.2.5.2 When system button is in OFF state, and pir button is in ON state

In **Figure 98**, system button is in OFF state, and the pir button is in ON state.

Labelled 1: It is the serial monitor of ESP32. No command received even after detecting motion because system button is in OFF state.

Labelled 2: It is the serial monitor of ESP32 CAM. The highlighted part indicates that the system button is in OFF state, and the pir button is in ON state. But still, it will not detect motion because the system button is in OFF state.

Labelled 3: It is the pir button on the Ubidots dashboard, which is shown as ON.

Labelled 4: It is the system button on the Ubidots dashboard, which is shown as OFF.

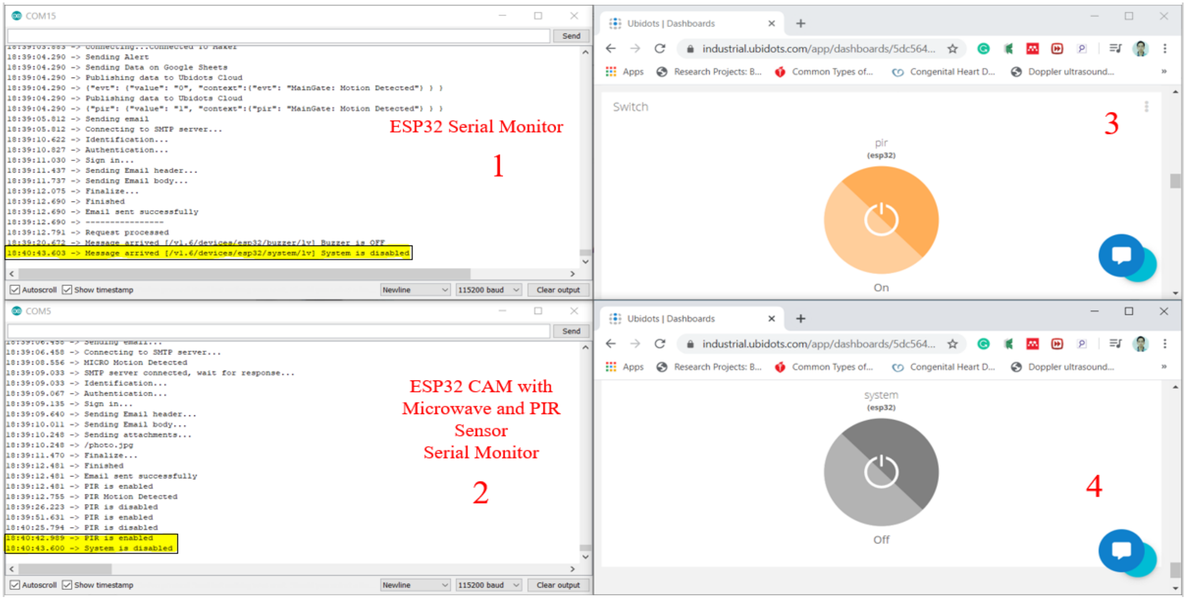


Figure 98: When system button is in OFF state and pir button is in ON state

11.2.5.3 When both system button and pir button are in OFF state

In **Figure 99**, both system switch and the pir switch are in OFF state.

Labelled 1: It is the serial monitor of ESP32. It is the serial monitor of ESP32. No command received even after detecting motion because system is OFF.

Labelled 2: It is the serial monitor of ESP32 CAM with the PIR sensor. The highlighted part indicates system and pir are in OFF state. It will not detect motion because system and pir both are in OFF state.

Labelled 3: It is the system switch on the Ubidots dashboard, which is shown as OFF.

Labelled 4: It is the pir switch on the Ubidots dashboard, which is shown as OFF.

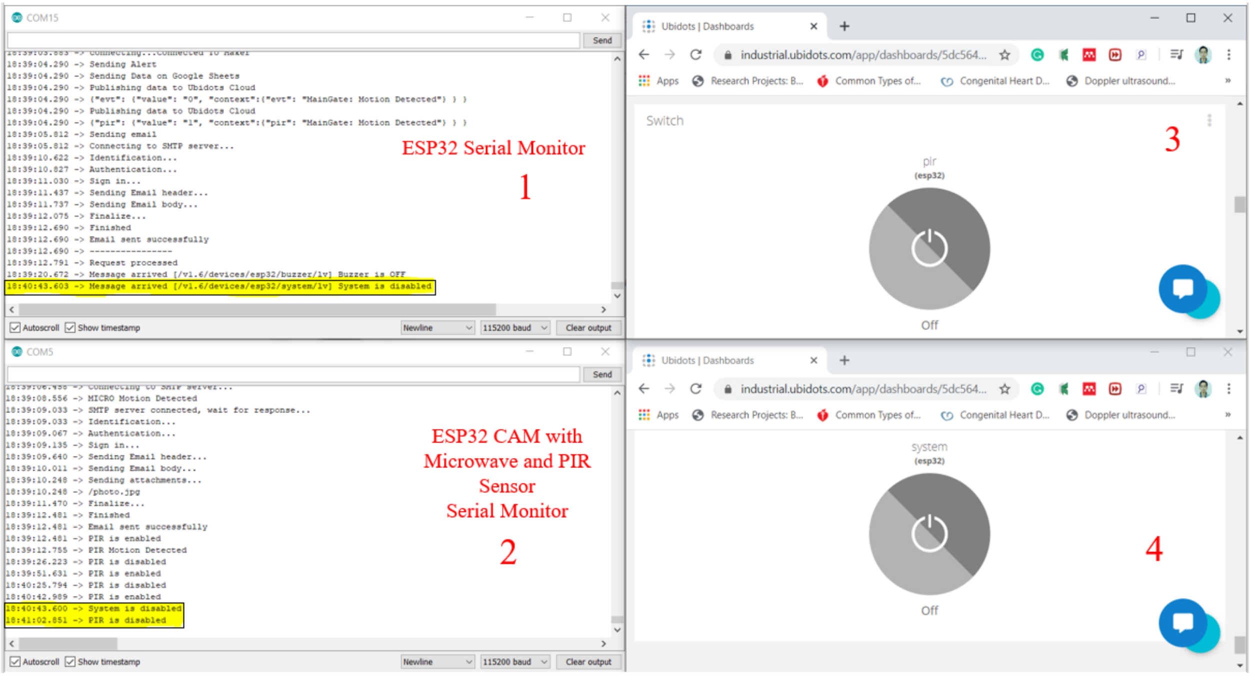


Figure 99: When both system button and pir button are in OFF state

11.3 Appendix C

All the source code files are in the folder Source_code_file and are arranged as follows:

1. **flame_corridor_pir:** For the Flame Sensor and Corridor PIR sensor.
2. **garden_gate_final:** For Garde Gate (Keypad)
3. **main_control:** For Main controller (with buzzer)
4. **PIR_Microwave_Camera:** For Front Gate Security (PIR and Microwave sensor)
5. **shock_pir_pir_mag:** For Living room and Bedroom PIR sensor, Magnetic Switch and shock sensor for window security.
6. **Ultra_Garage:** For Garage Security (Ultrasonic Sensor)
7. **Instructions to use Source Code.pdf:** Instructions for making changes before flashing the codes.