

IoT Real World Programmable Blocks



A Project Submitted in Partial Fulfillment of the Requirements for the Degree
of Bachelor of Science in Computer Science and Engineering
Under National University

Submitted By

Md. Antor Ahmed

Reg. No: 16502000527

Session: 2016-17

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
NATIONAL UNIVERSITY, DHAKA, BANGLADESH**

September 2022

APPROVAL

The project is titled “**IoT Real World Programmable Blocks**” submitted by Reg No: **16502000527** to the development of Computer Science and Engineering. Dhaka City College, Dhaka, Bangladesh has been accepted as satisfactory for partial fulfillment of the required studio style and content for a Bachelor’s Degree in Computer Science and Engineering.

**Signature of Internal
Examiner**

**Signature of External
Examiner**

Supervisor
Md. Shahiduzzaman
Assistant Professor
Department of CSE
Dhaka City College

Head of the Department
Md. Mustafa Kamal
Associate Professor
Department of CSE
Dhaka City College

DECLARATION

I hereby, declare that the work presented in this project “**IoT Real World Programmable Blocks**” is the outcome of the investigation performed by use under the supervision of **Md. Shahiduzzaman**, Assistant Professor, CSE Department, Dhaka City College, Dhaka, Bangladesh. We also declare that no part of this project and thereof has been or is being submitted elsewhere for the award of any degree or diploma.

Supervised by:

Md. Shahiduzzaman

Assistant Professor

Department of Computer Science and Engineering

Dhaka City College

Submitted by:

Md. Antor Ahmed

Reg. No.: 16502000527

ABSTRACT

The Internet of Things, or IoT, refers to the billions of physical devices around the world that are now connected to the internet, all collecting and sharing data. Thanks to the arrival of super-cheap computer chips and the ubiquity of wireless networks, it's possible to turn anything, from something as small as a pill to something as big as an aero plane, into a part of the IoT. Connecting up all these different objects and adding sensors to them adds a level of digital intelligence to devices that would be otherwise dumb, enabling them to communicate real-time data without involving a human being. The Internet of Things is making the fabric of the world around us smarter and more responsive, merging the digital and physical universes.

ACKNOWLEDGEMENTS

All praises are for the almighty Allah for giving us strength, without which I could not afford to attempt this research work.

I would like to express my sincere and heartiest gratitude to my honorable thesis supervisor **Md. Shahiduzzaman**, Assistant Professor, Dept. of Computer Science and Engineering, Dhaka City College, for his continuous motivation, guidance and keen encouragement, which helped me throughout the time of my research work. Nothing is comparable to his keen advice.

I would like to thank all the members of the board examiner for their precious time in understanding my work and their insightful comment. I would like to thank all of my friends for their co-operation. Finally, yet importantly, I am grateful to my parents for their continuous support and co-operation.

I also express our sincere thanks and gratitude to Prof. **Bedar Uddin Ahmed** (Principal) and **Md. Mustafa Kamal** (Head of the Department) for their support and guidance and constant encouragement for completion of project report.

I also thankful to other respected teachers of the department, who helped me in a number of ways by providing various resources and moral support.

Any omission in this brief acknowledgment does not mean lack of gratitude.

Table of Contents

APPROVAL	ii
DECLARATION.....	iii
ABSTRACT.....	iv
ACKNOWLEDGEMENTS	v
Table of Contents	vi
List of Figures.....	viii
List of Tables:.....	ix
<i>Chapter 1</i>	<i>10</i>
<i>Introduction.....</i>	<i>10</i>
1.1 Introduction.....	11
1.2 Motivation for the work	11
1.3 Project Objectives	11
1.4 Project Goals	12
1.5 Application technology	12
1.6 Application server	12
<i>Chapter 2</i>	<i>14</i>
<i>Components of Project.....</i>	<i>14</i>
2.1 ESP8266-12E.....	15
2.2 AMS1117 3.3V	18
2.3 DHT11 – Humidity	19
2.4 LED.....	19
2.5 Resistor (1K, 220ohm).....	21
2.6 Buzzer	23
2.7 Power Supply	24
2.8 Battery	25
2.9 TP4056.....	26
2.10 Transistor (BC547)	28
<i>Chapter 3.....</i>	<i>30</i>
<i>Requirements & Analysis.....</i>	<i>30</i>
3.1 Problem Statement	31

3.2 Drawbacks of Existing System	31
3.3 Methodology	32
3.4 Projects features	32
3.5 Some Applications of this project:.....	32
Chapter 4.....	33
Design	33
4.1 System Architectural Design	34
4.2 Circuit Schematic Design	35
4.3 E-R Diagram	36
Chapter 5	38
Implementation	38
5.1 Introduction.....	39
5.2 PCB Design.....	39
5.3 3D Modelling	41
5.4 Circuit Board Assemble	44
5.5 Future Works	48
Appendix.....	52
Server Code.....	52
Python Module.....	53
ESP8266 Firmware Source Code.....	55
Conclusion	57
References	58

List of Figures

Figure 2.1 ESP8266-12E	15
Figure 2.2 AMS1117	18
Figure 2.3 DHT11	19
Figure 2.4 LED	20
Figure 2.5 Working of LED.....	20
Figure 2.6 Resistor	21
Figure 2.7 Resistor in parallel.....	22
Figure 2.8 Resistor in series.....	22
Figure 2.9 Buzzer.....	23
Figure 2.10 Buzzer pinout.....	23
Figure 2.11 Buzzer working principle	24
Figure 2.12 DC power supply	24
Figure 2.13 Battery	25
Figure 2.14 Battery Discharge profile	26
Figure 2.15 TP4056	27
Figure 2.16 BC547.....	28
Figure 2.17 Transistor Symbol	29
Figure 4.1 Block Diagram.....	34
Figure 4.2 Schematic Design	35
Figure 4.3 Database Design	37
Figure 4.4 E-R Diagram.....	37
Figure 5.1 PCB Design	40
Figure 5.2 3D Model Top View.....	41
Figure 5.3 3D model Side View	42
Figure 5.4 3D Model Side View 2	42

Figure 5.5 3D Model Inside View 1	43
Figure 5.6 3D Model Inside View 2	43
Figure 5.7 3D Model Inside Top View	44
Figure 5.8 Implemented Block - Relay	45
Figure 5.9 Implemented Block – Buzzer	46
Figure 5.10 Implemented Block - Temperature.....	46
Figure 5.11 Implemented Block - Button or LED	47
Figure 5.12 Implemented Block - Outside.....	47
Figure 5.13 Implemented Block - Side View	48

List of Tables:

Table 2.1 ESP8266 Pinout	15
--------------------------------	----

Chapter 1

Introduction

Chapter 1

Introduction

1.1 Introduction

Building an IoT [1] gadget requires a ton of knowledge, but with IoT Blocks [8] there is no need for expertise in circuits [7] or network programming [3]. With an IoT block in every project, home, office we can add smart features like motion-sensitivity, remote control [5], orientation monitoring, voice commands, notifications, text messaging, and more.

1.2 Motivation for the work

Once you've determined your motivation, goals and scouted potential locations before finally setting on one. It's time to start thinking about envisioning what you want that space to communicate to your prospective devices and what action you want them to take. If you've only been used to communicating with your devices through your physical action, it sometime difficult to do. After all, there is still a world of difference between a server and a physical space. However, the interesting challenge here will be trying to bridge the gap and creating a consistent expensive across the board.

1.3 Project Objectives

- The system should have a user credentials. Every device uses user information to work with his devices.
- The user is the only person who can operate.
- The user should maintain property. User identify property type as it is residential or commercial property.
- The user can inform their agents for regarding to property and update the information regarding property.
- The system is very useful for the companies or builders that can post and edit their properties and their personal info and admin can monitor records of all of them.
- The system is also useful which also keeps track of Account details of users and devices and also RES Industry.

1.4 Project Goals

- **Planned approach towards working:** - The working in the organization will be well planned and organized. The data will be stored properly in data stores, which will help in retrieval of information as well as its storage.
- **Accuracy:** - The level of accuracy in the proposed system cannot be decided. Because here user buy and another user build the home. There is no guarantee.
- **Reliability:** - The reliability of the proposed system will be high due to the above stated reasons. The reason for the increased reliability of the system is that now there would be proper storage of information.

1.5 Application technology

For server technology we use PHP and we also applied MySQL database in our project.

Back End: In this project MySQL is used as Back End. This is used to store database of the project. MySQL in web is the world's most widely used open-source relational database management system (RDBMS) that runs as a server providing multi-user access to a number of databases. It is also a popular choice of database for use in web applications.

1.6 Application server

The project requires the following software tools in order to function properly.

- **Arduino IDE:**

The Arduino Integrated Development Environment is a cross-platform application that is written in functions from C and C++. It is used to write and upload programs to Arduino compatible boards, but also, with the help of third-party cores, other vendor development boards.

- **Python & IDE:**

Python is an interpreted high-level general-purpose programming language. Its design philosophy emphasizes code readability with its use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

- **MySQL:**

MySQL is an open-source relational database management system. Its name is a combination of "My", the name of co-founder Michael Wideness's daughter, and "SQL", the abbreviation for Structured Query Language.

- **PHP:**

PHP is a general-purpose scripting language geared towards web development. It was originally created by Danish-Canadian programmer Rasmus Lerdorf in 1994. The PHP reference implementation is now produced by The PHP Group.

- **Draw Back:**

Enterprise Architect & Smart draw are used to create or draw Various types of Diagrams and Models.

Chapter 2

Components of Project

Chapter 2

Components of Project

2.1 ESP8266-12E



Figure 2.1 ESP8266-12E

ESP-12E is a miniature **Wi-Fi module** present in the market and is used for establishing a wireless network connection for microcontroller or processor. The core of ESP-12E is **ESP8266EX**, which is a high integration wireless SoC (System on Chip). It features ability to embed Wi-Fi capabilities to systems or to function as a standalone application. It is a low-cost solution for developing IoT applications.

2.1.1 Pin Configuration

The ESP-12E module has twenty-two pins and we will describe function of each pin below.

Table 2.1 ESP8266 Pinout

Pin	Name	Description
1	RST	Reset Pin of the module
2	ADC	Analog Input Pin for 10-bit ADC (0V to1V)
3	EN	Module Enable Pin (Active HIGH)

4	GPIO16	General Purpose Input Output Pin 16
5	GPIO14	General Purpose Input Output Pin 14
6	GPIO12	General Purpose Input Output Pin 12
7	GPIO13	General Purpose Input Output Pin 13
8	VDD	+3.3V Power Input
9	CS0	Chip selection Pin of SPI interface
10	MISO	MISO Pin of SPI interface
11	GPIO9	General Purpose Input Output Pin 9
12	GPIO10	General Purpose Input Output Pin 10
13	MOSI	MOSI Pin of SPI interface
14	SCLK	Clock Pin of SPI interface
15	GND	Ground Pin
16	GPIO15	General Purpose Input Output Pin 15
17	GPIO2	General Purpose Input Output Pin 2
18	GPIO0	General Purpose Input Output Pin 0
19	GPIO4	General Purpose Input Output Pin 4
20	GPIO5	General Purpose Input Output Pin 5
21	RXD0	UART0 RXD Pin
22	TXD0	UART0 TXD Pin

2.1.2 Features and Electrical Characteristics

- Wireless Standard: IEEE 802.11 b/g/n protocol
- Frequency Range: 2.412 - 2.484 GHz
- Serial Transmission: 110 - 921600 bps, TCP Client 5
- SDIO 2.0, SPI and UART Interface available
- PWM available
- One ADC channel available
- Programmable GPIO available
- Wireless Network Type: STA / AP / STA + AP
- Security Type: WEP / WPA-PSK / WPA2-PSK
- Encryption Type: WEP64 / WEP128 / TKIP / AES
- Network Protocol: IPv4, TCP / UDP / FTP / HTTP
- Operating Voltage: 3.3V
- Maximum current allowed to draw per pin: 15mA
- Power down leakage current of < 10uA
- Integrated low power 32-bit MCU
- Onboard PCB Antenna
- Wake up and transmit packets in < 2ms
- Standby power consumption of < 1.0mW
- Operating Temperature: -40°C to +125 °C

2.1.3 Overview of ESP-12E

ESP-12E is a member of 'ESP-XX' series. Although all of them are based on ESP8266 SoC they differ in on output pins, flash memory and antenna type. These modules numbered from **ESP-01 to ESP-15** and are best in performance and cost. Many engineers use these modules to setup a wireless communication between two applications. For data sharing and IoT you will find these modules Ideal.

2.1.4 How to use the ESP-12E

This module does not have complex circuitry or programming so using this module is very easy. We will construct a simple application circuit for understanding the module working.

Steps for setting up a simple application circuit:

- Connect positive +3.3V power to the module.
- Interface module to a microcontroller or ARDUINO using UART (Connect RXD of ESP to RXD of μ C & TXD of ESP to TXD of μ C).
- Download the libraries for the module from the internet. For ARDUINO, the IDE will have pre-installed libraries. If you do not have them just update the libraries from ARDUINO website.
- Write the program for setting up the baud rate and data exchange.
- Send data to the module for transmitting through Wi-Fi or Receive data from the module that was transmitted via Wi-Fi.
- There is another way for setting up the module which is to bypass microcontroller and directly connect the module to PC using FTDI. After interface you can use serial monitor to communicate with the module.

2.2 AMS1117 3.3V



Figure 2.2 AMS1117

The AMS1117 series of adjustable and fixed voltage regulators are designed to provide up to 1A output current and to operate down to 1V input-to-output differential. The dropout voltage of the device is guaranteed maximum 1.3V, decreasing at lower load currents. On-chip trimming adjusts the reference voltage to 1.5%. Current limit is set to minimize the stress under overload conditions on both the regulator and power source circuitry. The AMS1117 devices are pin compatible with other three-terminal SCSII regulators and are offered in the low-profile surface mount SOT-223 package, in the 8L SOIC package and in the TO-252 (DPAK) plastic package.

2.3 DHT11 – Humidity

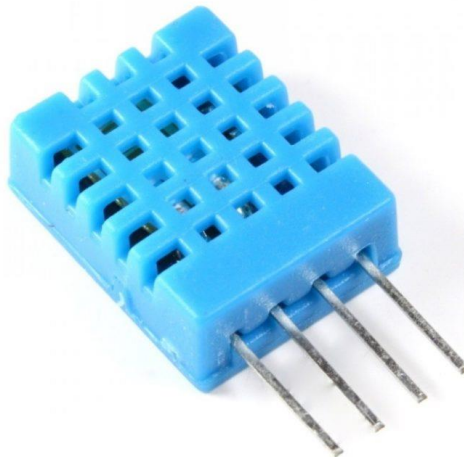


Figure 2.3 DHT11

The **DHT11** is a commonly used **Temperature and humidity sensor**. The sensor comes with a dedicated NTC to measure temperature and an 8-bit microcontroller to output the values of temperature and humidity as serial data. The sensor is also factory calibrated and hence easy to interface with other microcontrollers.

The sensor can measure temperature from 0°C to 50°C and humidity from 20% to 90% with an accuracy of $\pm 1^\circ\text{C}$ and $\pm 1\%$. So, if you are looking to measure in this range then this sensor might be the right choice for you.

2.4 LED

The LED is a PN-junction diode which emits light when an electric current passes through it in the forward direction. In the LED, the recombination of charge carrier takes place. The electron from the N-side and the hole from the P-side are combined and gives the energy in the form of heat and light. The LED is made of semiconductor material which is colorless, and the light is radiated through the junction of the diode.



Figure 2.4 LED

The LEDs are extensively used in segmental and dot matrix displays of numeric and alphanumeric character. The several LEDs are used for making the single line segment while for making the decimal point single LED is used.

2.4.1 Working of LED

The working of the LED depends on the quantum theory. The quantum theory states that when the energy of electrons decreases from the higher level to lower level, it emits energy in the form of photons. The energy of the photons is equal to the gap between the higher and lower level.

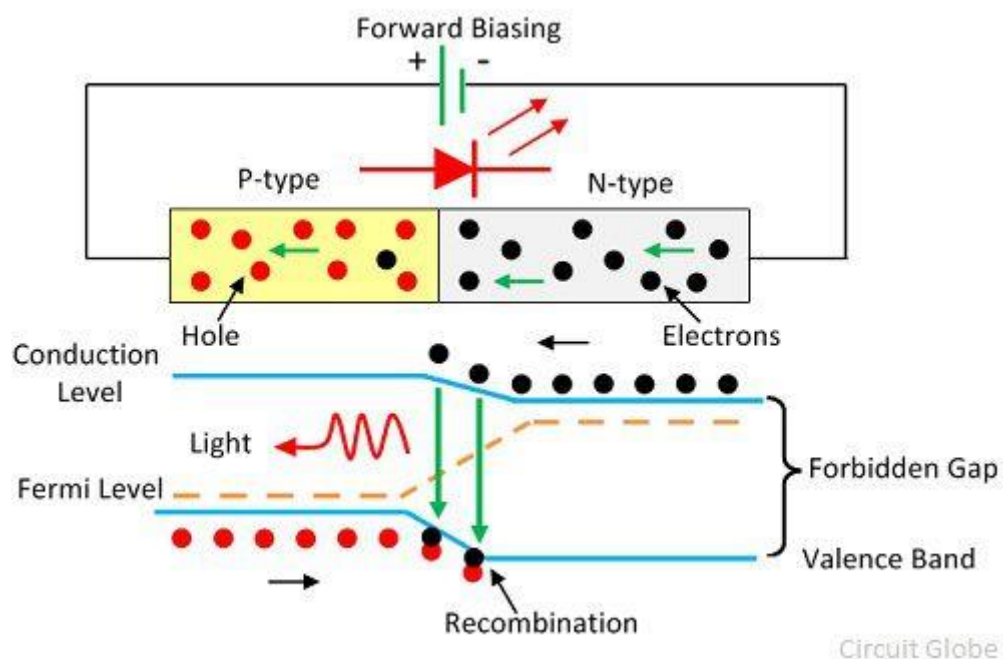


Figure 2.5 Working of LED

The LED is connected in the forward biased, which allows the current to flows in the forward direction. The flow of current is because of the movement of electrons in the opposite direction. The recombination shows that the electrons move from the conduction band to valence band and they emit electromagnetic energy in the form of photons. The energy of photons is equal to the gap between the valence and the conduction band.

2.5 Resistor (1K, 220ohm)

Resistor is an electrical component that reduces the electric current.

The resistor's ability to reduce the current is called resistance and is measured in units of ohms (symbol: Ω).

If we make an analogy to water flow through pipes, the resistor is a thin pipe that reduces the water flow.



Figure 2.6 Resistor

Ohm's law

The resistor's current I in amps (A) is equal to the resistor's voltage V in volts (V)

divided by the resistance R in ohms (Ω):

$$I = \frac{V}{R}$$

The resistor's power consumption P in watts (W) is equal to the resistor's current I in amps (A) times the resistor's voltage V in volts (V):

$$P = I \times V$$

The resistor's power consumption P in watts (W) is equal to the square value of the resistor's current I in amps (A)

times the resistor's resistance R in ohms (Ω):

$$P = I^2 \times R$$

The resistor's power consumption P in watts (W) is equal to the square value of the resistor's voltage V in volts (V)

2.5.1 Resistors in parallel:

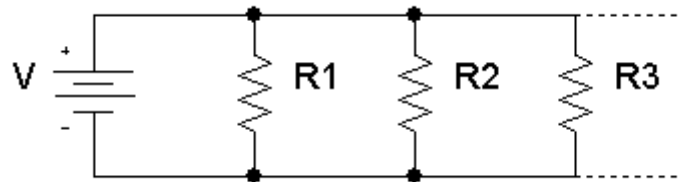


Figure 2.7 Resistor in parallel

The total equivalent resistance of resistors in parallel R_{Total} is given by:

$$\frac{1}{R_{Total}} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} + \dots$$

So, when you add resistors in parallel, the total resistance is decreased.

2.5.2 Resistors in series:

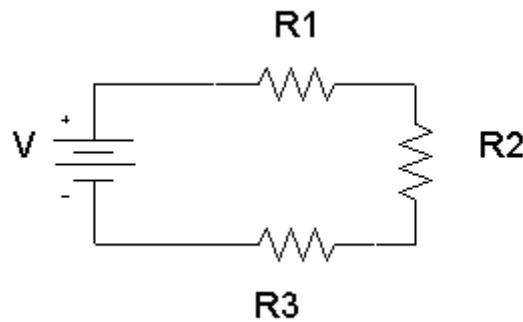


Figure 2.8 Resistor in series

The total equivalent resistance of resistors in series R_{total} is the sum of the resistance values:

$$R_{total} = R_1 + R_2 + R_3 + \dots$$

In this project we are using 1kilo-ohm resistor for Buzzer and 220ohm resistor for LED.

2.6 Buzzer

Buzzer is a kind of electronic sound receiver with integrated structure. It is widely used as a voice device in electronic products like computers, printers, copying machines, alarm apparatus, electronic toys, auto electronic devices, telephones, etc. In this experiment, we are going to use micro: bit to drive buzzer and make its sound circulate between high frequency and low frequency just like alarm song. And we will present its sound frequency on micro: bit with bar chart format.



Figure 2.9 Buzzer

Buzzer is a kind of voice device. It is made of vibration and resonance device. According to the difference of control method, we can divide buzzer into active type and passive type.

2.6.1 Working Principle of Buzzer

Because active buzzer has integrated amplify sampling circuit and resonance system, when DC power input passes through active buzzer, it will make resonance device generate sound signal. We can see the schematic diagram below for the working principle of active buzzer:

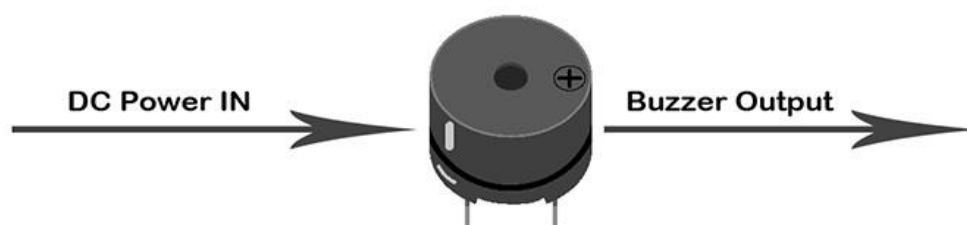


Figure 2.10 Buzzer pinout

When square wave signal passes through the buzzer, its resonance device will transform the square wave signal input into sound signal output. Below is the schematic diagram for the working principle of passive buzzer:

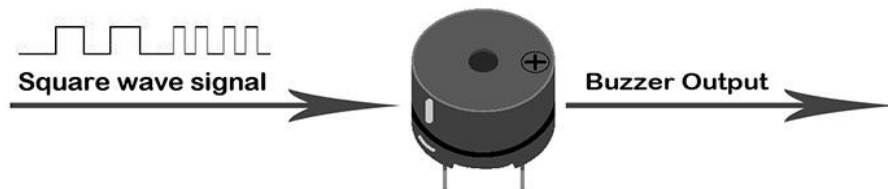


Figure 2.11 Buzzer working principle

In this Project we are using active Buzzer.

2.7 Power Supply

5V power supplies (or 5VDC power supplies) are one of the most common power supplies in use today. In general, a 5VDC output is obtained from a 50VAC or 240VAC input using a combination of transformers, diodes and transistors. 5V power supplies can be of two types: 5V regulated power supplies, and 5V unregulated power supplies. 5V regulated power supplies come in three styles: Switching regulated AC to DC, Linear regulated AC to DC, and Switching regulated DC to DC.



Figure 2.12 DC power supply

Specification:

Module Properties: non-isolated step-down constant current, constant voltage module (CC CV)

Input voltage: AC 220V

Output voltage: DC 5V

Continuously adjustable Output Current: 2.5A

Turn lamp current: constant current value (0.1), turn the lamp current and constant value linkage, such as constant current Value is 3A, turn the lamp current is set to a constant current is 0.1 times ($0.1 * 3A = 0.3A$).

Lowest pressure: 1V

Output Power: Maximum power is about 12W

Conversion efficiency: up to about 95%

Operating frequency: 300KHZ

Output ripple: 20M bandwidth Input 5V

Output 5V 2.5A ripple around 50mV (Excluding noise)

Output short circuit protection: Yes, constant current Input

Reverse Polarity Protection: None

Output prevent backflow: None

Wiring: Terminal

No-load current: Typical 20mA (5V switch 5V)

Load regulation: 1% (constant)

Voltage regulation: 1%

2.8 Battery

Depending on the design and chemistry of your lithium cell, you may see them sold under different nominal "voltages". For example, almost all lithium polymer batteries are 3.7V or 4.2V batteries. What this means is that the maximum voltage of the cell is 4.2v and that the "nominal" (average) voltage is 3.7V. As the battery is used, the voltage will drop lower and lower until the minimum which is around 3.0V. You should see the number 3.7V written on the battery itself somewhere.

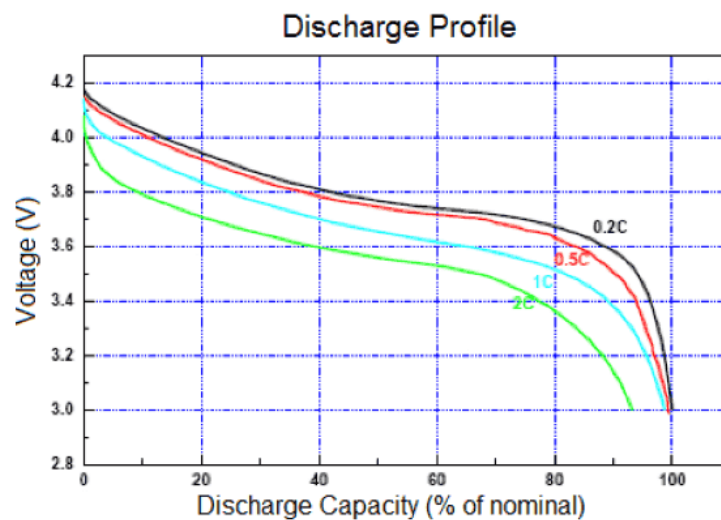


Figure 2.13 Battery

For example, here is a profile of the voltage for a 'classic' 3.7V/4.2V battery. The voltage starts at 4.2 maximum and quickly drops down to about 3.7V for the majority of the battery life. Once you hit 3.4V the battery is dead and at 3.0V the cutoff circuitry disconnects the battery (more on that later).

You may also run across 4.1V/3.6V batteries. These are older than 4.2V/3.7V - they use a slightly different chemistry and you'll see the 3.6V marking on the cell.

Nowadays you may also be able to purchase 4.35V cells! These are the latest chemistry; they



Discharge: 3.0V cutoff at room temperature.

Figure 2.14 Battery Discharge profile

have a little more power as indicated by the voltage being higher than 4.2V. They tend to be cylinder lithium-ions used for laptop batteries, and lights so it's not terribly likely you'll just run into one unless you're looking for it.

Make sure when you're buying batteries and chargers to match them up! Overcharging a 3.6V battery by attaching it to a 4.2V battery charger can at the very least permanently damage your battery and at worst cause a fire!

2.9 TP4056

The TP4056 is a complete constant-current/constant-voltage linear charger for single cell lithium-ion batteries. Its SOP package and low external component count make the TP4056

ideally suited for portable applications. Furthermore, the TP4056 can work within USB and wall adapter. No blocking diode is required due to the internal PMOSFET architecture and have prevent too negative Charge Current Circuit. Thermal feedback regulates the charge current to limit the die temperature during high power operation or high ambient temperature. The charge voltage is fixed at 4.2V, and the charge current can be programmed externally with a single resistor. The TP4056 automatically terminates the charge cycle when the charge current drops to 1/10th the programmed value after the final float voltage is reached. TP4056 Other features include current monitor, under voltage lockout, automatic recharge and two status pins to indicate charge termination and the presence of an input voltage.



Figure 2.15 TP4056

2.9.1 FEATURES:

- Programmable Charge Current Up to 1000mA
- No MOSFET, Sense Resistor or Blocking Diode Required
- Complete Linear Charger in SOP-8
- Package for Single Cell Lithium-Ion Batteries
- Constant-Current/Constant-Voltage
- Charges Single Cell Li-Ion Batteries Directly from USB Port
- Preset 4.2V Charge Voltage with 1.5% Accuracy
- Automatic Recharge

- Two Charge Status Output Pins
- C/10 Charge Termination
- 2.9V Trickle Charge Threshold (TP4056)
- Soft-Start Limits Inrush Current
- Available Radiator in 8-Lead SOP Package, the Radiator need connect GND or impending

2.10 Transistor (BC547)

The transistor is a semiconductor device which transfers a weak signal from low resistance circuit to high resistance circuit. The words trans mean transfer property and sitar mean resistance property offered to the junctions. In other words, it is a switching device which regulates and amplify the electrical signal likes voltage or current.

The transistor consists two PN diode connected back-to-back. It has three terminals namely emitter, base and collector. The base is the middle section which is made up of thin layers. The right part of the diode is called emitter diode and the left part is called collector-base diode. These names are given as per the common terminal of the transistor. The emitter-based junction of the transistor is connected to forward biased and the collector-base junction is connected in



Figure 2.16 BC547

reverse bias which offers a high resistance.

There are two types of transistors, namely NPN transistor and PNP transistor. The transistor which has two blocks of n-type semiconductor material and one block of P-type semiconductor material is known as NPN transistor. Similarly, if the material has one layer of N-type material and two layers of P-type material then it is called PNP transistor. The symbol of NPN and PNP is shown in the figure below.

The arrow in the symbol indicates the direction of flow of conventional current in the emitter with forward biasing applied to the emitter-base junction. The only difference between the NPN and PNP transistor is in the direction of the current.

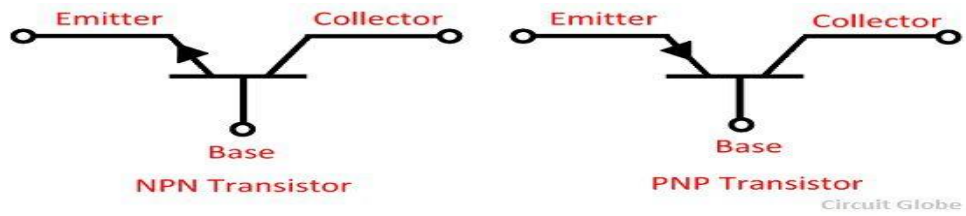


Figure 2.17 Transistor Symbol

Chapter 3

Requirements & Analysis

Chapter 3

Requirements & Analysis

3.1 Problem Statement

As the Internet of things facilitates a set of benefits, it also creates a significant set of challenges. Some of the IoT challenges are given below:

- **Security:** As the IoT systems are interconnected and communicate over networks. The system offers little control despite any security measures, and it can be leading the various kinds of network attacks.
- **Privacy:** Even without the active participation on the user, the IoT system provides substantial personal data in maximum detail.
- **Complexity:** The designing, developing, and maintaining and enabling the large technology to IoT system is quite complicated.

3.1.1 Existing Systems

- IoT Smart Home [1]
- IoT in Agriculture Monitoring [6]
- Google Home Voice Controller [4]
- Amazon Dash Button
- Smart door Lock
- Air Quality Monitor

3.2 Drawbacks of Existing System

- The major problem of IoT devices they are totally independent, that's mean each device do a specific predefined task always.
- Less configurable options.
- For multiple tasks, need multiple apps.
- Customization in device function isn't allowed.

3.3 Methodology

- We use MySQL as a database.
- Programming languages like C++, Python, PHP are used generally.
- Arduino IDE for hardware level programming.
- Basic electronics knowledge.
- 3D modelling with fusion 360.
- 3D prototype casing printing.
- API development using lumens framework.

3.4 Projects features

- Easily can integrate to any environment.
- Can apply any logical expression, iteration.
- Blocks can be used as single or combined together.
- Can be re-programmed on demand.
- Its Increases productivity of our daily life.
- Teaching programming logics easily to high school students.
- Simplify large tasks into smaller.
- This is very useful for security system in home and office.
- Replacement for current smart home automation devices.
- No limitation of use.

3.5 Some Applications of this project:

- Automation of Train Crossing
- Automation of Traffic signals
- Home Automation
- Factory/Office Automation

Chapter 4

Design

4.1 System Architectural Design

A block diagram is a diagram of a system in which the principal parts or functions are represented by blocks connected by lines that show the relationships of the blocks. They are heavily used in engineering in hardware design, electronic design, software design, and process flow diagrams.

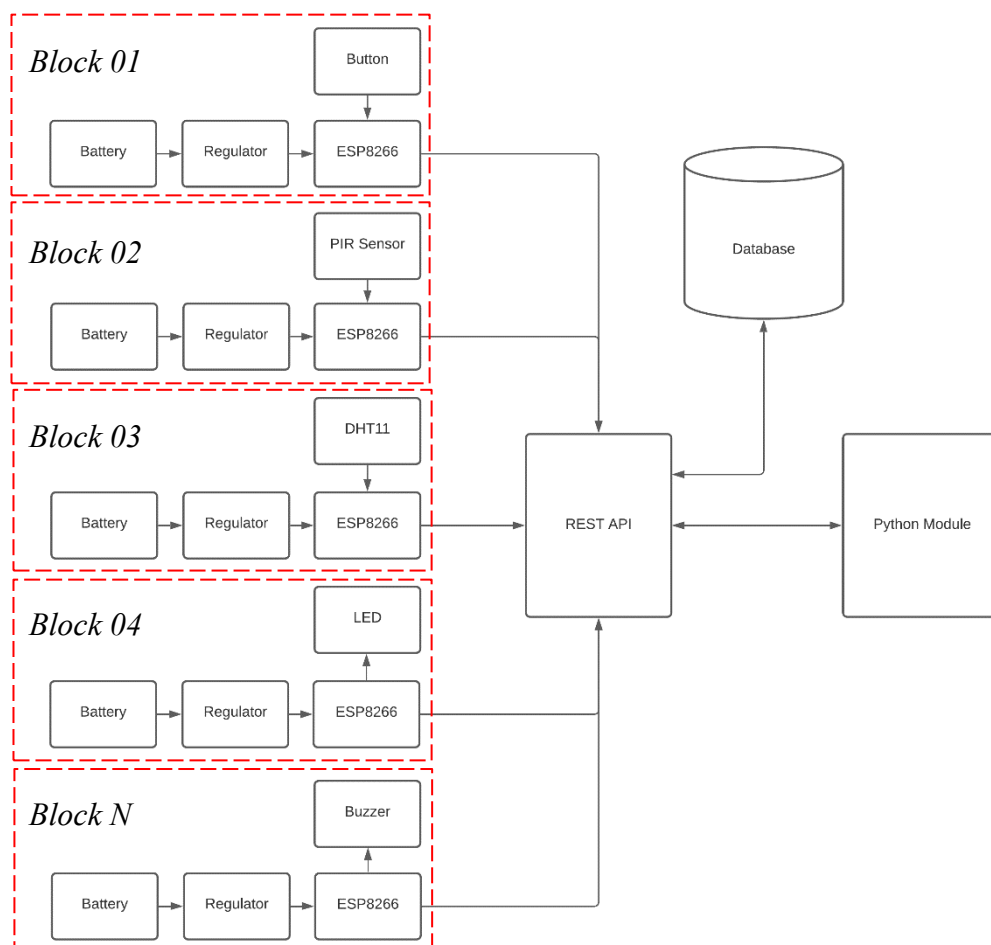


Figure 4.1 Block Diagram

4.2 Circuit Schematic Design

Schematic design is a rough construction drawing that offers a general overview of a project's basic features and construction cost estimates, allowing you to determine if your concept fits within the project budget. With schematic designs, your team turns ideas into physical drawings that you can look at and edit to help craft your construction project, and prepare you for the next phase of your architectural plan. Schematic design is the first phase of the architectural design process, which includes design development, construction documents, bidding, and construction administration.

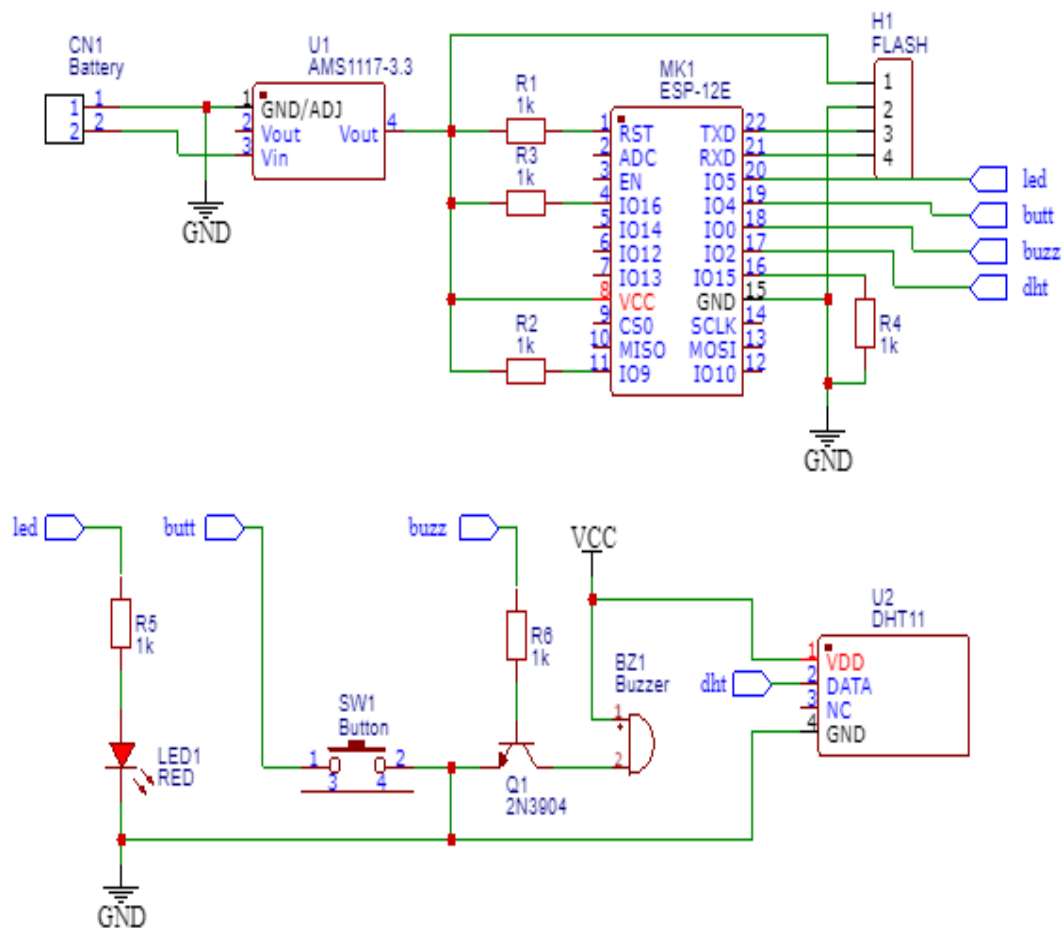


Figure 4.2 Schematic Design

4.3 E-R Diagram

An **Entity–relationship model (ER model)** describes the structure of a database with the help of a diagram, which is known as **Entity Relationship Diagram (ER Diagram)**. An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set.

An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database.

Here are the geometric shapes and their meaning in an E-R Diagram. We will discuss these terms in detail in the next section (Components of an ER Diagram) of this guide so don't worry too much about these terms now, just go through them once.

Rectangle: Represents Entity sets.

Ellipses: Attributes

Diamonds: Relationship Set

Lines: They link attributes to Entity Sets and Entity sets to Relationship Set

Double Ellipses: Multivalued Attributes

Dashed Ellipses: Derived Attributes

Double Rectangles: Weak Entity Sets

Double Lines: Total participation of an entity in a relationship set

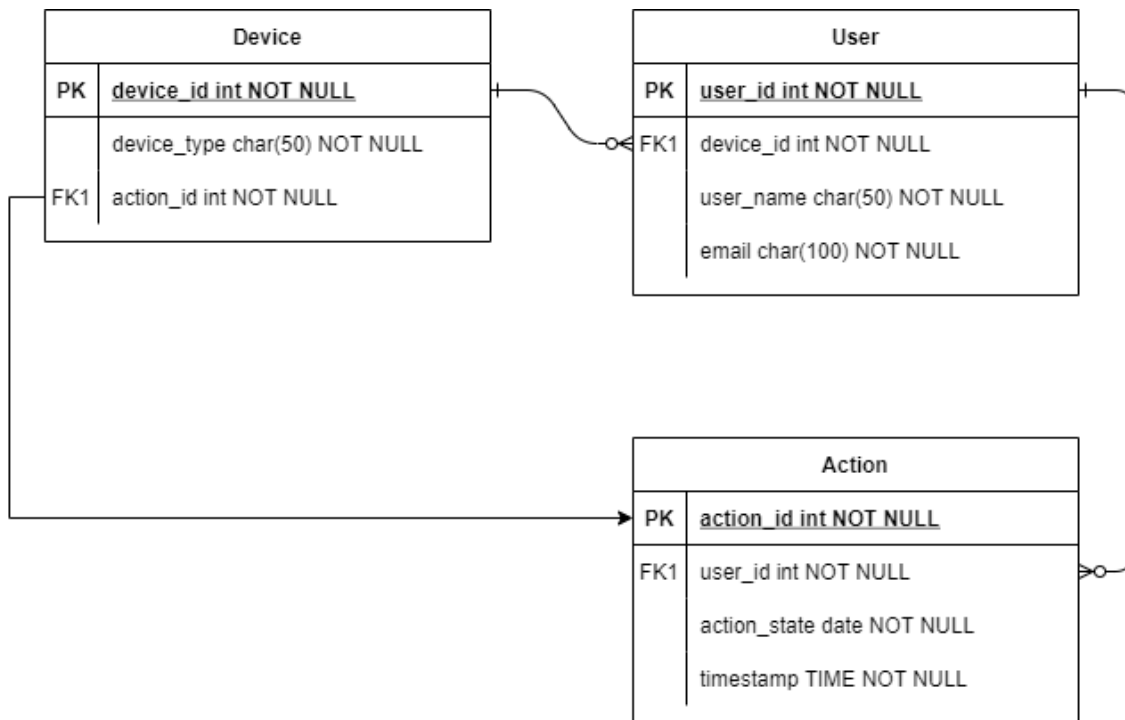


Figure 4.3 Database Design

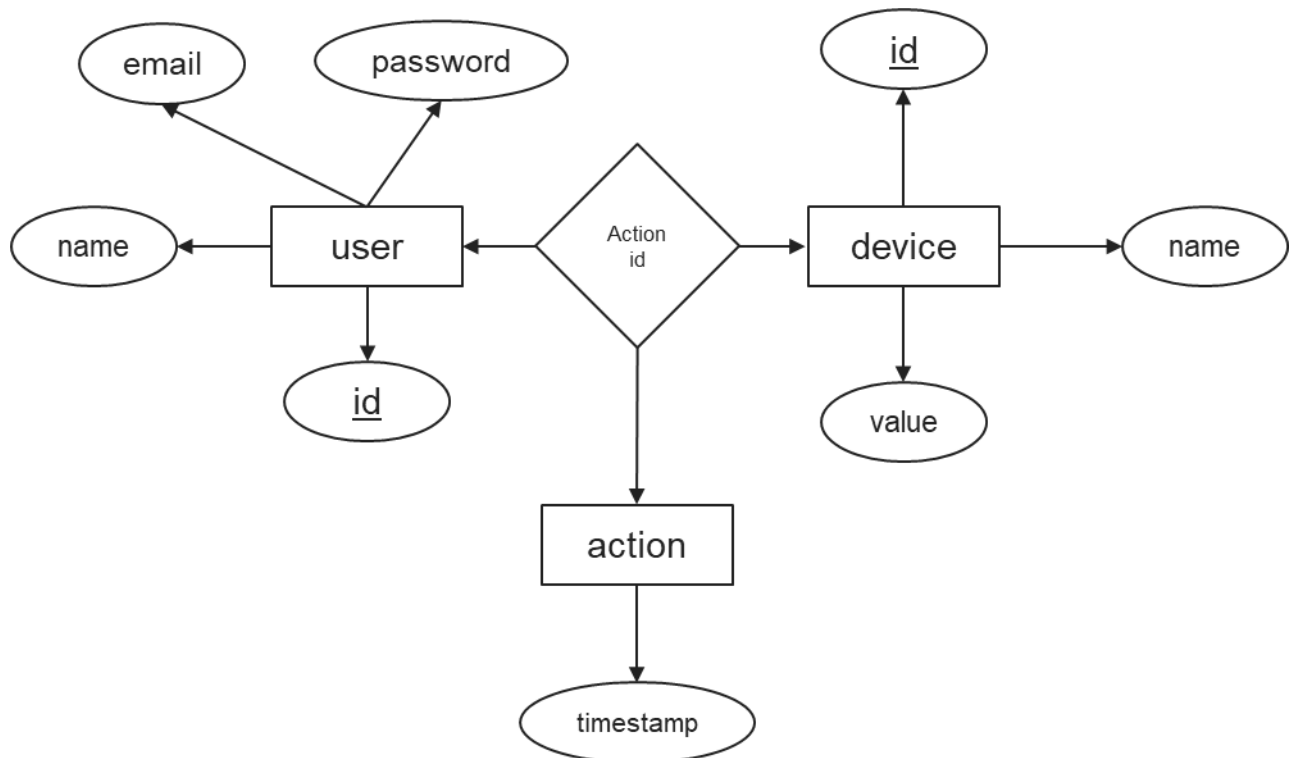


Figure 4.4 E-R Diagram

Chapter 5

Implementation

Chapter 5

Implementation

5.1 Introduction

Internet of Things has emerged as a leading technology around the world. It has gained a lot of popularity in lesser time. Also, the advancements in Artificial Intelligence and Machine Learning have made the automation of IoT devices easy. Basically, AI and ML programs are combined with IoT devices to give them proper automation. Due to this, IoT has also expanded its area of application in various sectors. Here, in this section, we will discuss the applications and the future scope of IoT in healthcare, automotive, and agriculture industries.

5.2 PCB Design

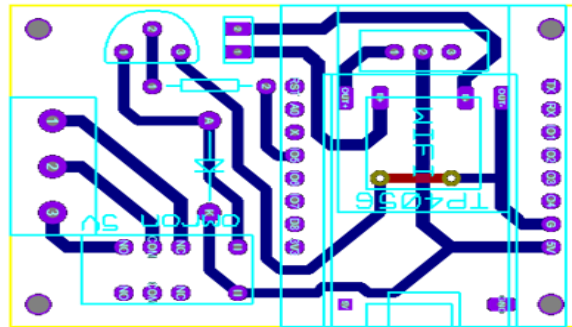
Printed circuit board is the most common name but may also be called "printed wiring boards" or "printed wiring cards". Before the advent of the PCB circuits were constructed through a laborious process of point-to-point wiring. This led to frequent failures at wire junctions and short circuits when wire insulation began to age and crack.

A significant advance was the development of wire wrapping, where a small gauge wire is literally wrapped around a post at each connection point, creating a gas-tight connection which is highly durable and easily changeable.

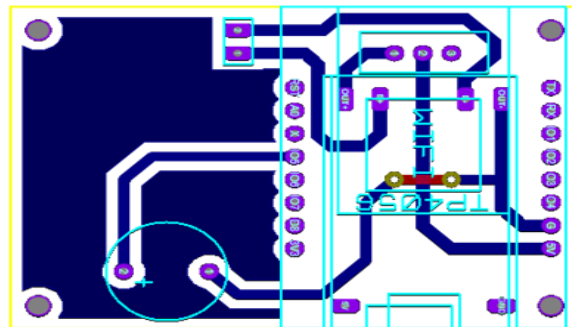
As electronics moved from vacuum tubes and relays to silicon and integrated circuits, the size and cost of electronic components began to decrease. Electronics became more prevalent in consumer goods, and the pressure to reduce the size and manufacturing costs of electronic products drove manufacturers to look for better solutions. Thus, was born the PCB.

It's important to communicate with the PCB or flex application engineer during the beginning of any PCB design job and discuss everything you need or require for your project. Open and constant communication with your PCB provider ensures that you're both on the same page, which can reduce the need for multiple extensive revisions.

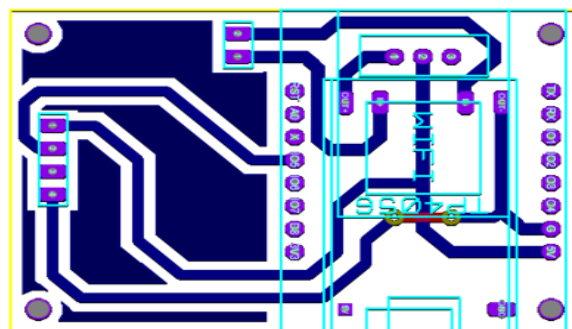
While it is rare that a printed circuit board is 100% perfectly designed on the first try, communication will help ensure that your PCB is properly designed from the start and that any revisions are not major ones that will derail your project and timeline. It's also important to consider the environment in which the product has to function.



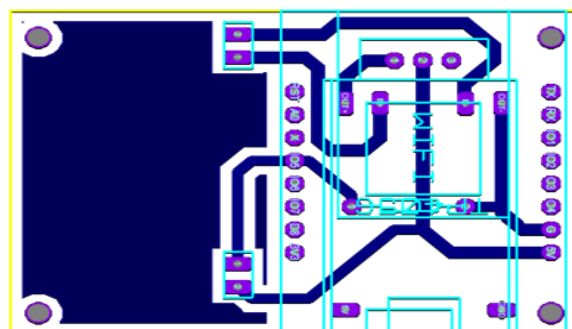
Relay Block



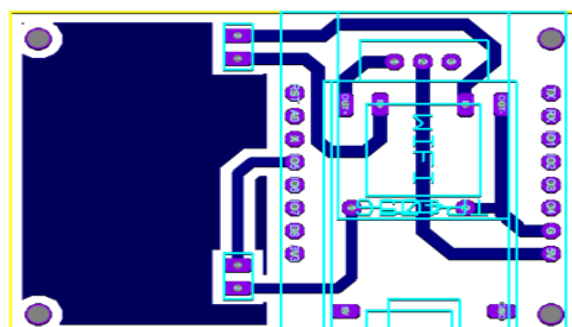
Buzzer Block



Temp Block



LED Block



Button Block

Figure 5.1 PCB Design

5.3 3D Modelling

A 3D model is essentially made up of vertices, which come together to form a mesh and act as the core of the 3D model. Each point on the model can be manipulated to change the shape. By using coordinate data, the software identifies the location of each vertical and horizontal point, all relative to a reference point.

The most common way to begin making a 3D model is to start with a basic shape – a cube, box, sphere, or whatever you think is best suited. From your starter shape, you can start molding and refining it into what you desire.

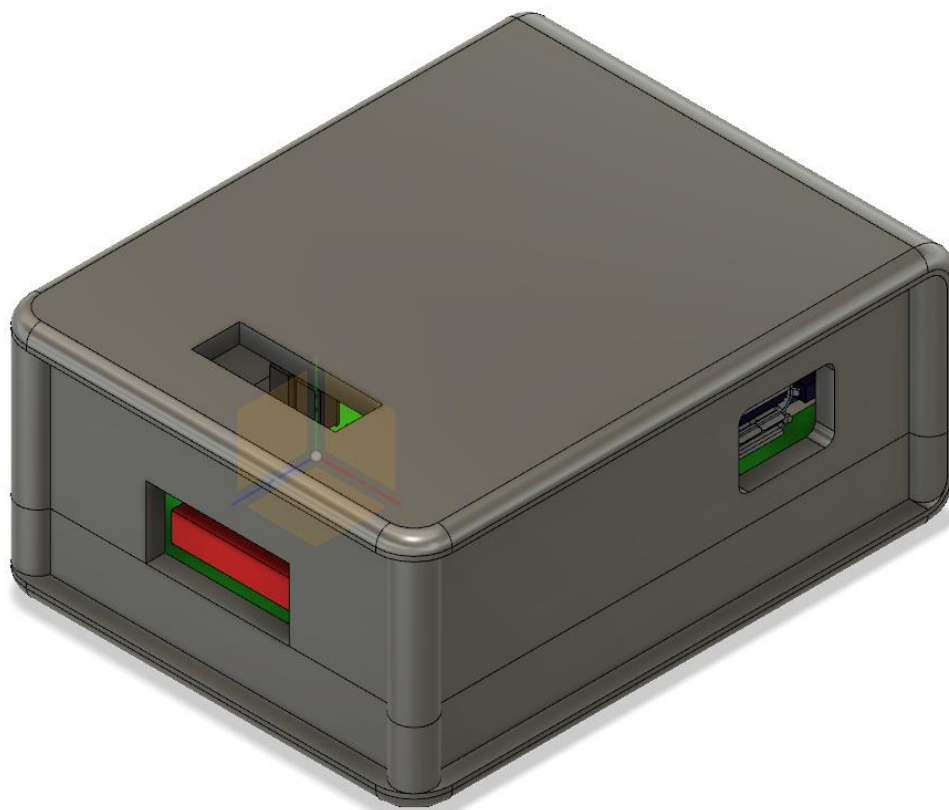


Figure 5.2 3D Model Top View



Figure 5.3 3D model Side View

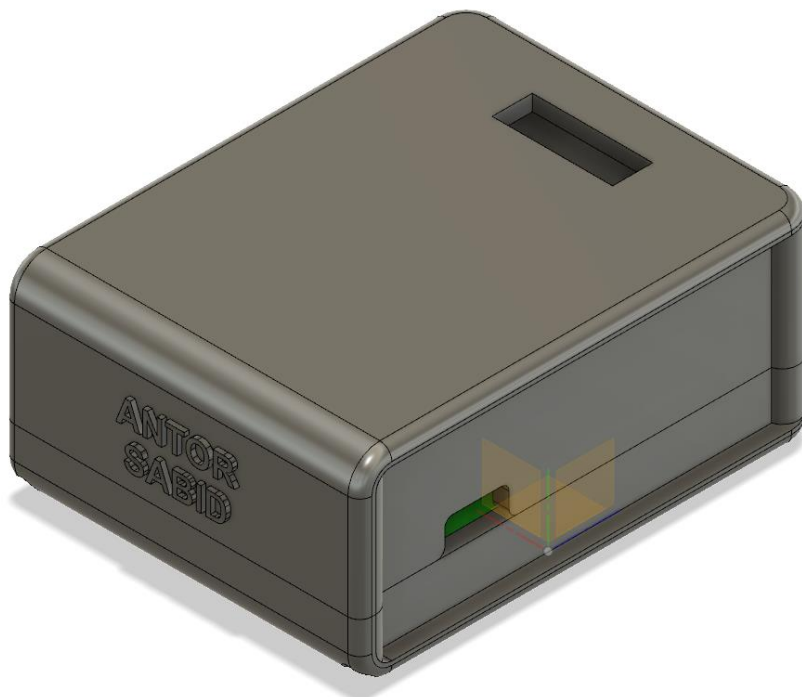


Figure 5.4 3D Model Side View 2

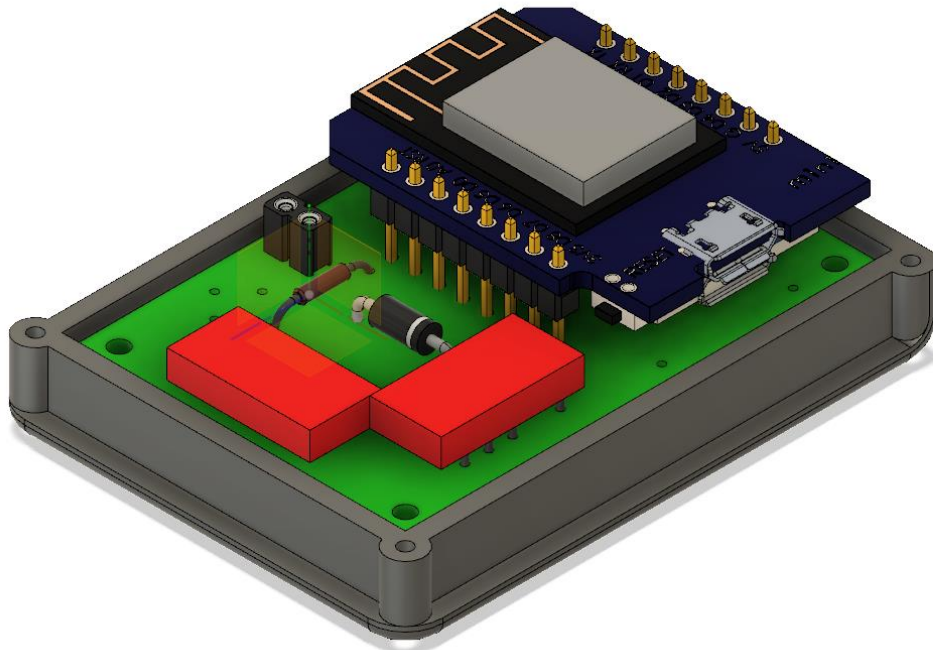


Figure 5.5 3D Model Inside View 1

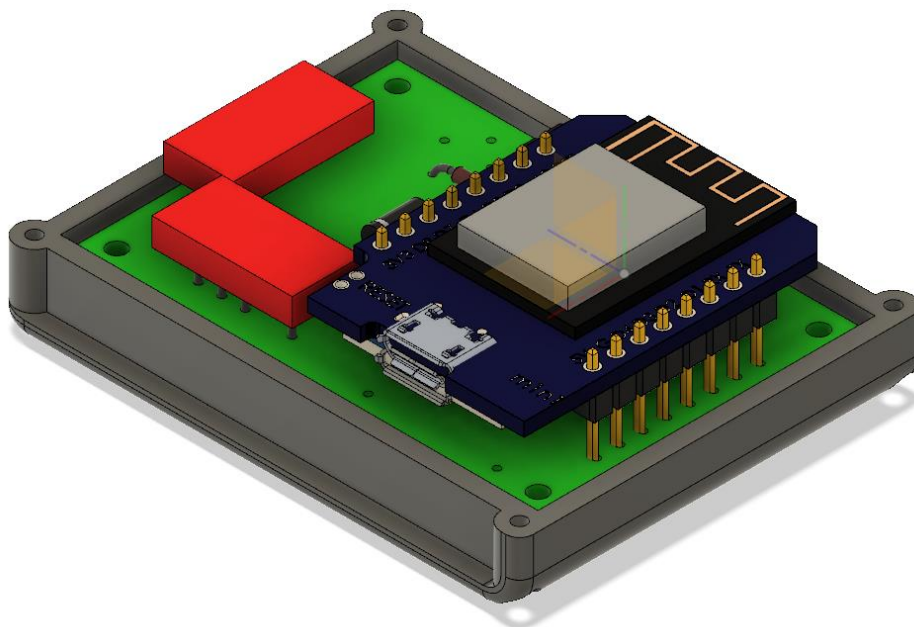


Figure 5.6 3D Model Inside View 2

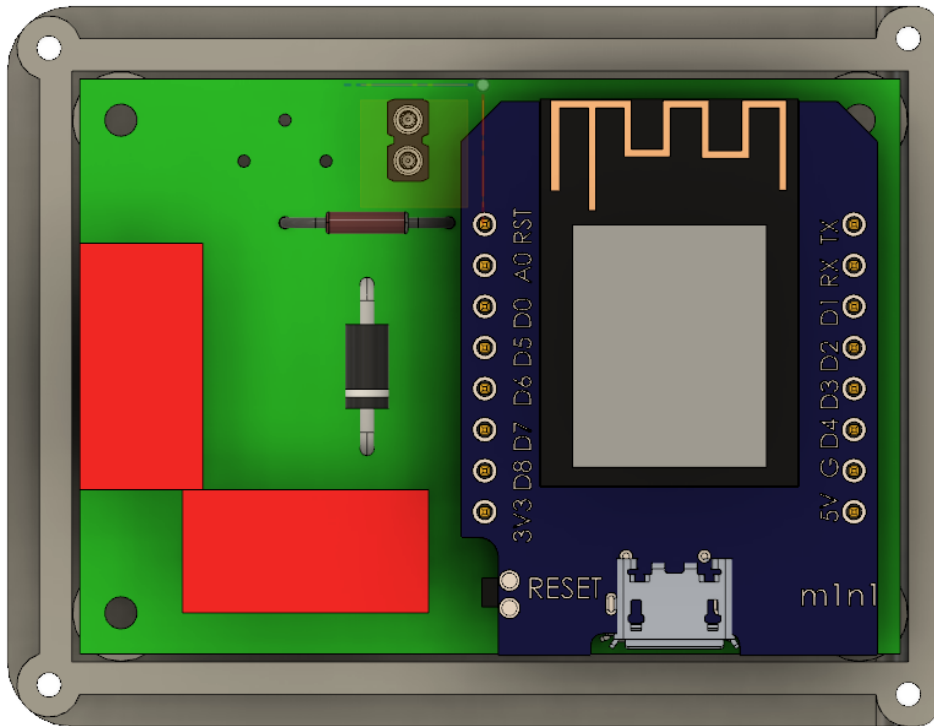


Figure 5.7 3D Model Inside Top View

5.4 Circuit Board Assemble

Here is a step-by-step process of PCB board assembly:

Step 1: Apply solder paste to the circuit board

Place the thin, stainless-steel stencil over the board using a mechanical fixture. Solder paste should be applied evenly to the circuit board in the exact locations needed.

Step 2: Pick and place the machine

SMDs, or surface mount components, should be placed on a prepared PCB by a robotic device. Then, the components need to be soldered onto the circuit board surface.

Step 3: Let the solder paste solidify

In order to adhere the components to the PCB, the solder paste needs to reflow and remain in place for an extended period of time.

Step 4: Inspect the PCB assembly

After the reflow, process is complete and the mount components are soldered into place, comes the PCB inspection. The assembled board should be tested and inspected for functionality.

Step 5: Insert the plated through-hole component

A plated through-hole, or PTH, component is a hole in the PCB that is plated through the board. Rather than soldering paste, more specialized soldering method is required for PTHs.

Manual soldering: A manual, through-hole insertion.

Wave soldering: The automated version on manual soldering where a wave of molten solders all the holes in the bottom of the board at once.

Step 6: Complete a final inspection

Once the soldering process of the PCB board assembly is complete, it is time to do a final inspection and functional test. Run power and simulated signals to test the PCBs electrical characteristics. A sign that the PCB has failed is when it shows the fluctuation of electrical signals during the test. If the PCB fails the final inspection, it should be scrapped. And the process begins all over until a successful PCB is produced.

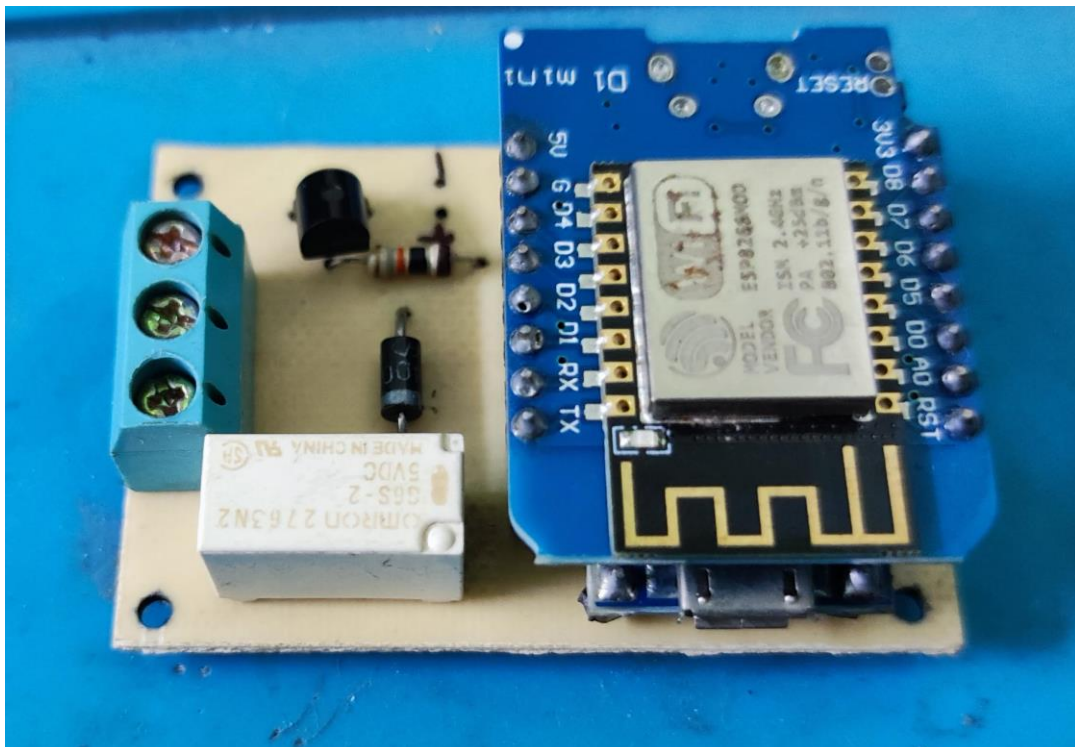


Figure 5.8 Implemented Block - Relay

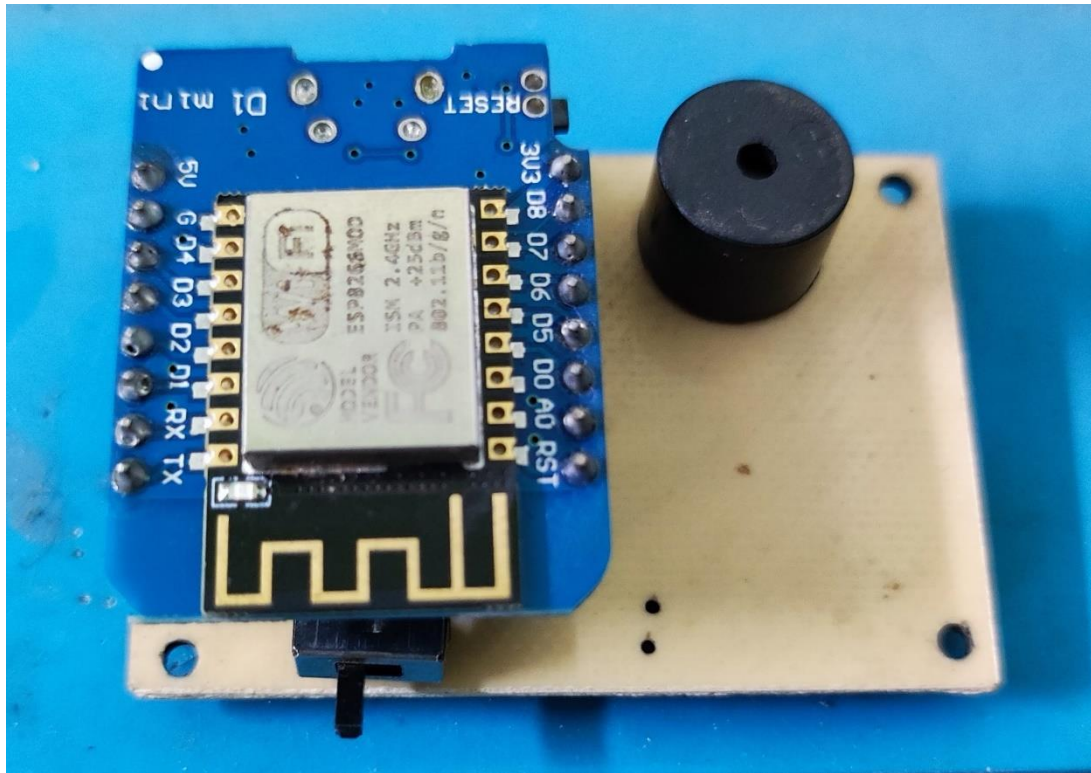


Figure 5.9 Implemented Block – Buzzer

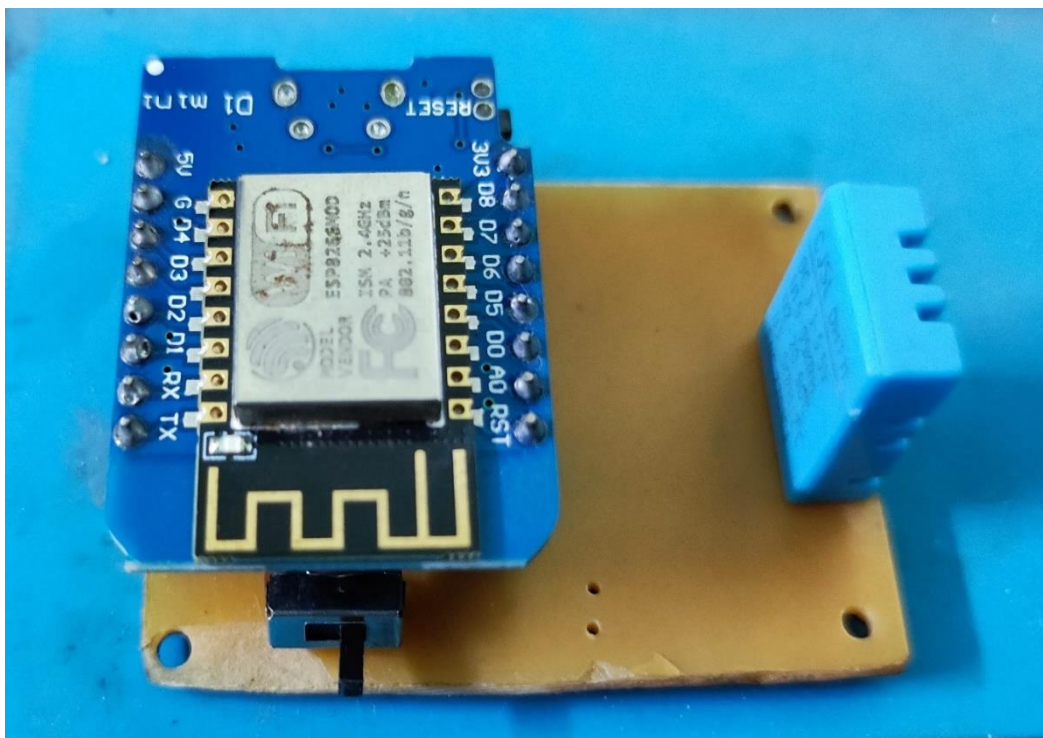


Figure 5.10 Implemented Block - Temperature

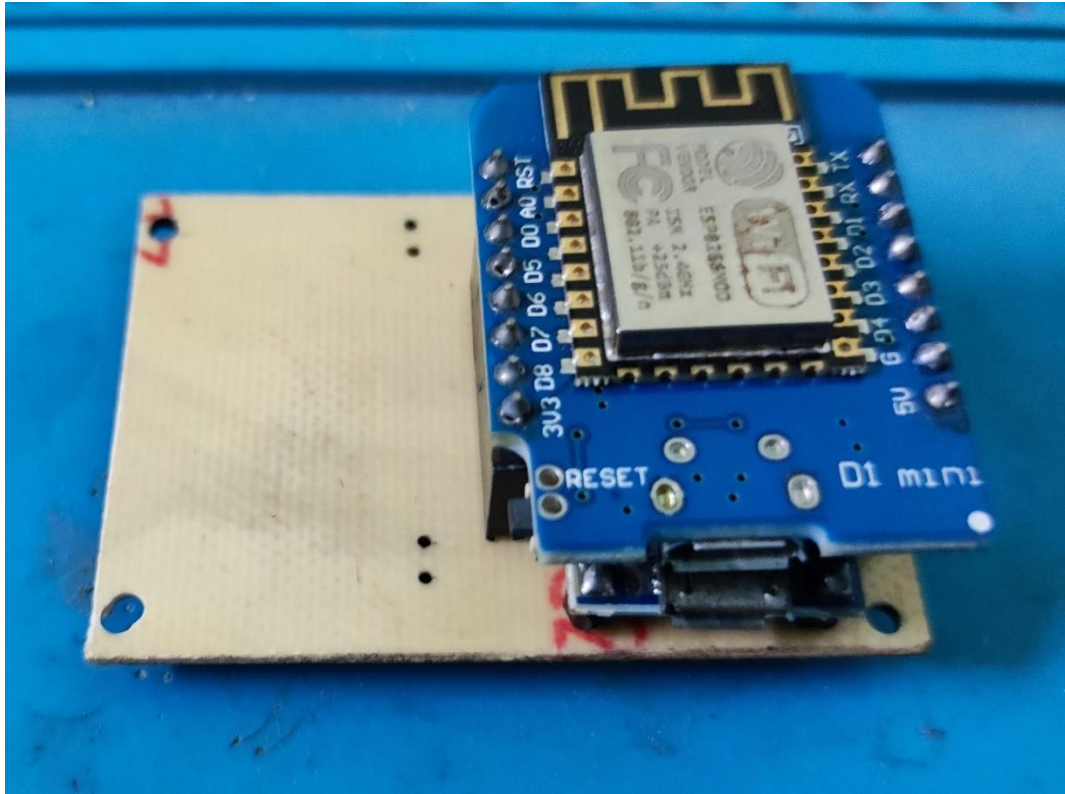


Figure 5.11 Implemented Block - Button or LED



Figure 5.12 Implemented Block - Outside

5.5



Figure 5.13 Implemented Block - Side View

Future Works

- A.I can be added so far device can recognize human pattern for controlling any blocks.
- Machine learning algorithm can be improved its performance.
- Security for authentication can be improve.
- Device data logger can be added.
- Anti-hack security like brute force attack should be prevent.

5.5.1 Healthcare

In this section, we will see the future of IoT in healthcare sector. IoT has proved to be one of the best tools for the healthcare industry. It helps provide advanced healthcare facilities to patients, doctors, and researchers. These facilities include smart diagnosis, wearable devices for tracking health, patient management, and many more. Furthermore, IoT devices have reduced unnecessary strain on the healthcare system.

5.5.2 IoT in healthcare

The healthcare devices can directly send the patients' data health to doctors over a safe network. This allows the doctors to diagnose the patients from remote locations.

Reasons for IoT being beneficial in the healthcare industry are as follows:

1. Error reduction in treatments

With the help of IoT devices, there is a decrease in manual errors in diagnosing patients. Thus, patients can get proper treatment on time. Also, the 24/7 diagnosis through gadgets gives a clearer idea about patients' health when compared to manual diagnosis.

2. Decrease in the cost of treatments

Manual diagnosis requires time and the use of various types of costly equipment, along with other hospital charges. Due to this, the overall cost of the treatment increases. We can reduce these costs by using IoT gadgets. Also, the hospital charges and congestion in hospitals can be reduced as patients can be diagnosed from their respective locations.

3. Availability of specialists in remote locations

Internet of Things solves one of the major problems of the healthcare industry that is the availability of doctors, especially specialists, in remote locations. With the help of IoT devices, the treatment of patients in the physical absence of doctors has become possible. The patients just need to wear the device. Then, the device will send all the real-time data of patients' health to the respective doctors for analyzing the condition. In this way, the scope of Internet of Things is helping the healthcare sector give proper treatment to the needy.

5.5.3 Agriculture

One of the three basic human needs is food. To fulfill the need for food, we do farming. However, now, as the population of the world is increasing, the agricultural industry is facing many challenges. Also, changes in weather conditions and climate hugely impact the agricultural industry. To meet the rising demand for food, the industry has hence adopted technology to increase productivity. It includes the use of precision farming, agricultural drones, and smart farming applications.

All these are built on top of the application of Internet of Things. Now, let us discuss how precision farming, smart farming applications, and agricultural drones help in increasing the productivity of the land.

5.5.4 Precision farming

In agriculture, Information and Communication Technology is a tool used for smart farming. With the help of IoT-based devices, crop fields are observed. The technology uses sensors to calculate the moisture of soil, humidity, and temperature. Also, it uses an automated irrigation system to make efficient use of water. Precision farming helps farmers monitor their fields and boost productivity.

Agricultural drones

Drones used for agriculture and farming are one of the best applications of Internet of Things. They are used to enhance agricultural processes. We use agricultural drones for planting crops, irrigating fields, spraying of pesticides, and monitoring the fields. With the help of drones, it becomes easier to evaluate the health of crops. This is all possible with the help of smart IoT-based devices that are used to make agricultural drones.

5.5.5 Smart greenhouses

Farmers use greenhouse farming to enhance the productivity of crops. In greenhouse farming, the environmental factors that affect the growth of crops are controlled by manual intervention. However, manually controlling the mechanism for the growth of crops is less productive. The emergence of IoT and technological advancements has led to the creation of IoT-based greenhouses that consist of various devices such as sensors, climate controllers, etc.

These IoT devices help in measuring the various environmental conditions according to the requirements of plants. As all sensors and devices connect over the Internet servers, they provide accurate information on the environmental conditions. Then, the devices activate actuators to control heaters, fans, windows, and lighting of greenhouses to set according to the environment.

This is how the scope of IoT is enhancing the productivity of the agricultural industry.

5.5.6 Automotive Industry

In the 21st century, the application of IoT is revolutionizing the automotive industry. One of the major applications is the creation of self-driving cars that has changed the trends of the automotive industry. Engineers have created self-driving cars to reduce manual errors and ensure a safe drive. Various companies around the world are creating self-driving vehicles including Google, Tesla, Mercedes-Benz, Volvo, Audi, and many more. These self-driving

cars use various technologies such as Data Science, Artificial Intelligence, Deep Learning, and IoT. IoT devices are programmed in such a way that they assist in creating an automated system for self-driving cars.

5.5.7 Self-driving cars

These IoT devices consist of HD cameras, thermal sensors, smart navigators, speed controller, rain sensors, wireless connectivity, and proximity sensors. While using these cars, you need to enter your location and the destination. Then, the navigator helps locate the destination and tries to find the shortest path. After that, the IoT-based HD cameras help in getting the visuals of the surrounding and send the data to the AI-based systems. These systems analyze and visualize the data of the surrounding and accordingly fix the response of the self-driving cars. Also, there are IoT-based speed controllers that help regulate the speed of these cars according to the traffic and congestion. This is how the scope of IoT is changing the trends in the automotive industry.

Appendix

Server Code

database.php

```
<?php
    $DBhost = "localhost";
    $DBuser = "esinebdc_projects";
    $DBpass = "QyO2P7h{e;DBW)o!7)Of";
    $DBname = "esinebdc_iotblocks";

    $database = new mysqli($DBhost, $DBuser, $DBpass, $DBname);

    if ($database->connect_error) {
        die("Connection failed: " . $database->connect_error);
    }
?>
```

index.php

```
<?php
include 'database.php';

if(isset($_GET["get"]) && isset($_GET["deviceId"])) {
    $email = $_GET["email"];
    $pass = $_GET["pass"];
    $deviceId = $_GET["deviceId"];

    $sql = "SELECT id,deviceId,state FROM blocks WHERE email='$email' AND
password='$pass' AND deviceId='$deviceId'";
    $result = $database->query($sql);

    $arr = array();
    while($row = $result->fetch_assoc()) {
        $arr[] = $row;
    }

    echo json_encode($arr);
}
else if(isset($_GET["get"])) {
    $email = $_GET["email"];
    $pass = $_GET["pass"];

    $sql = "SELECT id,deviceId,state FROM blocks WHERE email='$email' AND
password='$pass'";
    $result = $database->query($sql);

    $arr = array();
    while($row = $result->fetch_assoc()) {
        $arr[] = $row;
```

```

    }

    echo json_encode($arr);
}
else if(isset($_GET["set"])) {
    $email = $_GET["email"];
    $pass = $_GET["pass"];
    $deviceId = $_GET["deviceId"];
    $state = $_GET["state"];

    $sql = "UPDATE blocks SET state='$state' WHERE email='$email' AND
password='$pass' AND deviceId='$deviceId'";
    if($database->query($sql)) echo '{"status":"OK"}';
    else echo '{"status":"ERROR"}';
}
?>

```

Python Module

IoTBlocks.py

```

import requests

host = "https://esinebd.com/projects/IoTBlocks/index.php"
did = ['button', 'led', 'buzzer', 'temp', 'relay']
email = ""
password = ""
param = { }

def setUser(em, pa):
    email = em
    password = pa
    param['email'] = email
    param['pass'] = password
    print(f"Credential set as: {param}")

def button():
    reqParam = param
    reqParam['get'] = "
    reqParam['deviceId'] = did[0]
    resp = requests.get(host, params=reqParam)
    json = resp.json()
    return json[0]['state']

def LED(*arg):
    if(len(arg) == 0):
        reqParam = param
        reqParam['get'] = "
        reqParam['deviceId'] = did[1]

```

```

    resp = requests.get(host, params=reqParam)
    json = resp.json()
    return json[0]['state']
else:
    reqParam = param
    reqParam['set'] = "
    reqParam['deviceId'] = did[1]
    reqParam['state'] = arg[0]
    resp = requests.get(host, params=reqParam)
    json = resp.json()
    return json['status']

def buzzer(*arg):
    if(len(arg) == 0):
        reqParam = param
        reqParam['get'] = "
        reqParam['deviceId'] = did[2]
        resp = requests.get(host, params=reqParam)
        json = resp.json()
        return json[0]['state']
    else:
        reqParam = param
        reqParam['set'] = "
        reqParam['deviceId'] = did[2]
        reqParam['state'] = arg[0]
        resp = requests.get(host, params=reqParam)
        json = resp.json()
        return json['status']

def temp(*arg):
    if(len(arg) == 0):
        reqParam = param
        reqParam['get'] = "
        reqParam['deviceId'] = did[3]
        resp = requests.get(host, params=reqParam)
        json = resp.json()
        return json[0]['state']

def relay(*arg):
    if(len(arg) == 0):
        reqParam = param
        reqParam['get'] = "
        reqParam['deviceId'] = did[4]
        resp = requests.get(host, params=reqParam)
        json = resp.json()
        return json[0]['state']
    else:
        reqParam = param
        reqParam['set'] = "
        reqParam['deviceId'] = did[4]

```

```

reqParam['state'] = arg[0]
resp = requests.get(host, params=reqParam)
json = resp.json()
return json['status']

```

ESP8266 Firmware Source Code

```

#include <ESP8266WiFi.h>
#include <ESP8266WiFiMulti.h>
#include <ESP8266HTTPClient.h>
#include <ArduinoJson.h>
#include "DHT.h"

const char ssid[] = "Hello";
const char pass[] = "11223344";
const char host[] = "http://esinebd.com/projects/IoTBlocks/index.php";
const char deviceType[] = "led";

#define pin D5

ESP8266WiFiMulti wifi;
WiFiClient client;
HTTPClient http;
DHT dht(pin, DHT11);

String link;
int i;

void setup() {
  Serial.begin(9600);
  WiFi.mode(WIFI_STA);

  dht.begin();
  pinMode(pin, OUTPUT);
  pinMode(pin, INPUT);

  wifi.addAP(ssid, pass);
  wifi.run();
}

void loop() {
  float hum = dht.readHumidity();
  float tmp = dht.readTemperature();

  link = (String)host + "?get&email=antor.mee@gmail.com&pass=123456&deviceId=" +
deviceType;
  if (http.begin(client, link)) {
    int httpCode = http.GET();
    if (httpCode == HTTP_CODE_OK || httpCode ==
HTTP_CODE_MOVED_PERMANENTLY) {

```

```

int len = http.getSize();
String input = http.getString();
StaticJsonDocument<200> doc;
DeserializationError error = deserializeJson(doc, input.c_str(), len);

if (error) {
  Serial.print(F("JSON FAILED: "));
  Serial.println(error.f_str());
}
else {
  const char state = doc["state"].as<const char *>();
  if(state == "1") digitalWrite(pin, 1);
  else digitalWrite(pin, 0);
}
}
else Serial.println((String)"ERROR " + httpCode);
}
else Serial.println("CONNECT ERROR!");
}

```


Conclusion

Nowadays, technologies are growing rapidly and it is same goes to the IoT automation. As for this project it highly recommended for everyone in this world especially for a user who is with disabilities and for the householder too. This recommendation will lead to a green world which is it can help to save and reduce on electricity bill. Plus, it will help and lead the disabilities person can work independently and help them to manage their house safety in more organized way. It will help to prevent any loss to a user or unwanted circumstances to a user.

References

1. Teymourzadeh, Rozita, et al. "*Smart GSM Based Home Automation System.*" Systems, Process & Control (ICSPC), 2013 IEEE Conference on. IEEE, 2013.
2. Singh, Navab, et al. "*Remotely controlled home automation system.*" Advances in Engineering and Technology Research (ICAETR), 2014 International Conference on. IEEE, 2014.
3. Pavithra D, IoT based Monitoring and Control System for Home Automation, Proceedings of 2015 Global Conference on Communication Technologies (GCCT 2015)
4. Wenbo, Yan, Wang Quanyu, and Gao Zhenwei. "*Smart home implementation based on Internet and WiFi technology.*" Control Conference (CCC), 2015 34th Chinese. IEEE, 2015.
5. Gurek, Alper, et al. "*An android based home automation system.*" High-Capacity Optical Networks and Enabling Technologies (HONET-CNS), 2013 10th International Conference on. IEEE, 2013.
6. Beza Negash Getu, Hussain A. Attia, "*Automatic Control of Agricultural Pumps Based on Soil Moisture Sensing*", IEEE conference publication, 2015.
7. HemlataChannel¹, SukheshKothari², DipaliKadam³, "*Multidisciplinary Model for Smart Agriculture using Internet-of-Things (IoT), Sensors, Cloud-Computing, Mobile-Computing & Big-Data Analysis*", International Journal of Int.J. Computer Technology & Applications, Vol 6 (3), May-June 2015.
8. Joaquín Gutiérrez, Juan Francisco Villa-Medina, Alejandra Nieto Garibay and Miguel Ángel Porta-Gadara, "*Automated Irrigation System Using a Wireless Sensor Network and GPRS Module*", IEEE Transaction on Instrumentation and Measurement, Vol.63, No.1, January 2014.

9. Meena Kumari G, Dr. V.Vidya Devi, “*Real- Time Automation and Monitoring System for Modernized Agriculture*”, International Journal of Review and Research in Applied Sciences and Engineering (IJRRASE) Vol3 No.1. PP 7-12 March 2013.
10. Nikhil Agrawal, Smitan Singhal, “*Smart Drip Irrigation System using Raspberry pi and Arduino*”, International Conference on Computing, Communication and Automation (ICCCA2015)