# CHAPTER 1 INTRODUCTION

* 1. **GENERAL**

The egg incubator is a device that can control the temperature, humidity and position of the egg during the incubation process. When using the egg incubator, the hen does not incubate the egg. Therefore, the egg incubator can help farmers to incubate eggs to produce large numbers of chickens. It is necessary to investigate and build incubators for eggs from different animals such as hens, quails, turtles, partridges, and others. For the incubation system, it is necessary to automate the temperature adjustment, the humidity, in some cases the position of the eggs, the control by microcontroller and the use of IoT and solar energy. The development of an intelligent incubator to improve the incubation of hen eggs. The incubator can control temperature and humidity based on sensors and a National Instruments data acquisition card. The need for smart incubator is to generally increase hatchability of eggs which leads to the improvement and increase in the production of chicks and eggs for human consumption and the economic market. In this present age of information technology, the control and automation of devices, machines and systems are mostly achieved through mechatronic means with emphasis on soft control, this is mostly achieved by the use of programmed microcontrollers. Furthermore, the incubator is connected to the Internet to help farmers controlling and monitoring the incubator remotely. With these incubators, the farmer can monitor the eggs remotely saving time and money.

# OBJECTIVE

This project is for an egg hatching that works with the present of temperature. This egg hatching system is cheap and works with the principle of maintain certain temperature and humidity. Temperature and humidity are collected from the sensors and maintain the temperature by any tungsten bulb.

# IMPLEMENTATION

In recent years, most of the artificial applications are designed to be controlled remotely via internet to make the access to the machines from anywhere over the world is easier. The egg incubator is an electrical device which imitates the avian incubation by providing the suitable conditions for embryo in the egg to develop into a chick. The controller of egg incubator

56+keeps the temperature and humidity inside the incubator at optimal values and tilts the egg to make sure all parts of the egg is heated by a heater element until hatch the egg. Due to increasing the production of chickens for human consumption and economic market, the researchers are encouraged to develop and improve the artificial egg incubator to increase hatchability of eggs and produce the chickens. Therefore, serious efforts in designing and implementing a suitable controller for egg incubator are spend. High efficiency fans were used to circulate the heated air within the chamber to maintain even temperature.

# MOTIVATION

In olden days opportunity for farmers to produce chicks from egg with the consent of the mother hen, it is only one of the ways of transforming eggs to chicks. During this time, the type of incubation used was manual which meant it was totally depended on mother hen. Temperature changes were affected in the incubator by moving the eggs, adding additional eggs to use the heat of embryological development of older eggs, and by regulating the flow of fresh air through the hatching area. But in the modern era the smart incubation process was autonomous with respect to humidity and temperature control with the help of controllers and sensors. The most important difference between natural and smart incubation is the fact that the natural parent provides warmth by contact rather than surrounding the egg with warm air. But the worked on designing a smart incubator in which they used a node MCU as the wireless communication module for logging the necessary data to the cloud. It was realized that incorporating an incandescent lamp as the heat source reduced substantial power consumption as against using an electric heater.

# CHAPTER 2 LITERATURE REVIEW

1. **Ali. F and Amran. N. A (2016), “Development of an Egg Incubator Using Raspberry Pi for Precision,” International Journal of Agriculture, Forestry and Plantation, vol. 2, pp. 40–45.**

A low-cost, ARM-based computer called the Raspberry Pi was created for use in computer science teaching. It is a powerful credit card-sized computer that can be used for many tasks that a desktop computer can perform, including spreadsheets, word processing, and gaming. This essay will investigate the creation of a mechanism that can handle egg hatching without requiring the broody process. A monitoring system has also been designed so that the user may access the device without difficulty. Its main objective is to establish a setting that makes the process of incubating eggs safer and more organised. This device's creation also makes use of the Arduino UNO platform, which ultimately succeeded in creating the ideal conditions for the egg- hatching process. The Raspberry Pi is low-cost but by using other controllers we can reduce the cost instead of using microprocessor we can use microcontrollers.

# Okpagu. P. E and Nwosu. A. W (2016), “Development and Temperature Control of Smart Egg Incubator System for Various Types of Egg,” European Journal of Engineering and Technology, vol. 4, no.2, pp. 2056-5860.

This work is aimed at modeling, designing and developing an egg incubator system that is able to incubate various types of egg within the temperature range of 35 – 400C. This system uses temperature and humidity sensors that can measure the condition of the incubator and automatically

change to the suitable condition for the egg. Extreme variations in incubation temperature affect the embryo and ultimately, post hatch performance. In this work, electric bulbs were used to give the suitable temperature to the egg whereas water and controlling fan were used to ensure that humidity and ventilation were in good condition. The entire element is controlled using AT89C52 Microcontroller. The temperature of the incubator is maintained at the normal temperature using PID controller implemented in microcontroller. Mathematical model of the incubator, actuator and PID controller were developed. Controller design based on the models was developed using Matlab Simulink. The models were validated through simulation and the Zeigler- Nichol tuning method was adopted as the tuning technique for varying the temperature control parameters of the PID controller in order to achieve a desirable transient response of the system when subjected to a unit step input. After several assumptions and simulations, a set of optimal parameters were obtained at the result of the third test that exhibited a commendable improvement in the overshoot, rise time, peak time and settling time thus improving the robustness and stability of the system.

# Radhakrishnan. K, N. Jose, Sanjay. S. G, Cherian. T and Vishnu. K. R (2014), “Design and Implementation of a Fully Automated Egg Incubator,” International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 3, no. 2, pp. 7686–7690.

The goal of this is to describe the design of an egg incubator system based on an ATmega16 microcontroller that can automatically maintain the conditions best for embryo growth. A microprocessor receives the temperature data from the system's temperature sensor, which can track the temperature both inside and outside the incubator. The microcontroller uses relays to control an incandescent bulb and an air circulator to keep the temperature of

the eggs between 37 and 38.5 °C. Additionally, the temperature inside and outside of the incubator are shown on an LCD monitor. Additionally, the microcontroller has a timer that can be programmed by the user to manage a gear motor that tilts the egg holder.

# Shittue. S, Muhammad. A. S and Jimoh. O (2016), “Development of an Automatic Bird-Egg Incubator,” A Journal of Embedded System & Applications. vol. 5, no.1, pp. 1–10.

This developed an automatic egg incubator and hatchery system with automated humidity and temperature control. The performance evaluation of incubator and hatchery system was carried out. This was with a view to develop an improved incubator system that is capable of hatching large number of chicks at a time and also attaining high hatchability. The main heating elements are three sets of 60 W bulbs used to provide the warmth needed in the incubator. Data from the sensors was processed by the PIC16F877A and the various control elements were activated to moderate the conditions in the incubator (using C codes). The system was powered by electricity from the national grid. The incubator system developed was tested, and temperature and humidity reading on hourly bases were plotted. The graph obtained shows that the incubator will perform its desired function of hatching eggs effectively, because the temperature and humidity range were within the required range of 37– 38°C and 32–35% respectively.

# Tolentino. K. S, Justine. E, Enrico. G, Listanco. R. L. M, Anthony. M, Ramirez. M, Renon. T. L. U, Rikko. B and Samson. B (2018), “Development of Fertile Egg Detection and Incubation System Using Image Processing and Automatic Candling”, TENCON 2018, IEEE Region 10 Conference, pp. 0701-0706.

The development of an incubation system for autonomous temperature and humidity control using Arduino microcontroller interfaced and coded using LabView programming. The proposed system also includes important functions to hatch eggs which are candling through infertile egg identification using basic image acquisition, and egg turning that employs crank-rocker mechanism and a hatching chamber. It revolves around fusing all the elements of egg incubation and turning it into one device. It functions autonomously without having to consistently check and adjust to obtain optimal parameters. By using its monitoring features, the user can have real-time data of the day-to- day status of the incubator’s parameters. The speed of its automatic candling program is 1.129 seconds while the performance of the incubator held an optimal temperature of 36º Celsius with humidity between 40% and 60% with an optimal level of 50%. Lastly, the hatch rate percentage of the incubation using the proposed system is 69.44% while the percentage accuracy result for detecting fertile eggs is 91.43%.

# CHAPTER 3 PROPOSED SYSTEM

* 1. **INTRODUCTION**

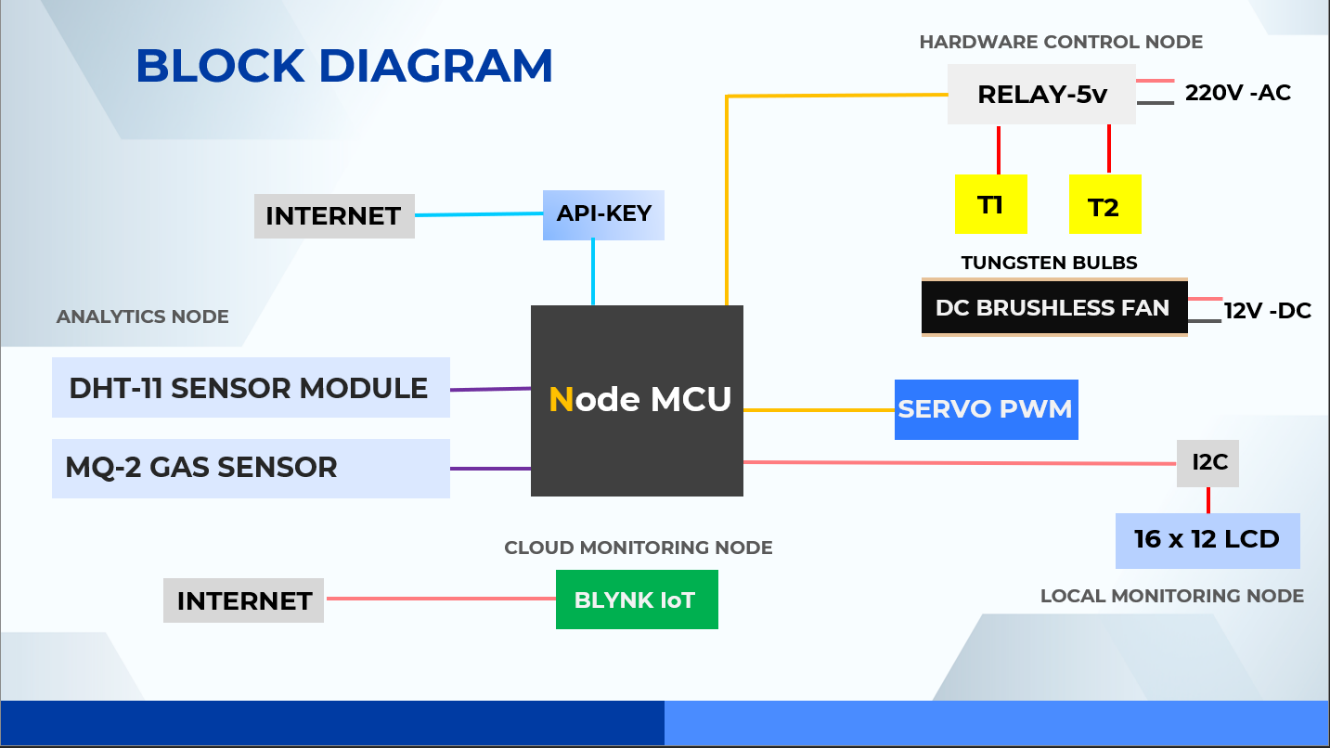
Egg incubator is one of the inventions that provide opportunity especially for those who want to be excellent farmers. The system will automatically control the temperature and humidity of the incubator for various types of eggs. The function of egg incubator is to take over the animal job to incubate an egg to hatching. Eggs have been incubated by artificial means for thousands of years. The system developed, consists of two parts, the mechanical and electronic part. The mechanical part was a mechanism for angular tilting of the egg trays up and down alternatively on hourly bases, using a DC motor and limit switch sensor for angular movement control. The electronic part comprised of the Node MCU(ESP8266), DHT11 sensor, servo motor and LCD display.

# INCUBATOR

Incubators of several types and capacities with adapters for eggs from different species are available. Basically, an incubator is a box that holds and rotates eggs while maintaining appropriate temperature, humidity and oxygen levels. A well-designed incubator should maintain temperature and humidity within 1 degree F wet bulb temperature. Several features are standard in popular or larger incubator models. Automatic turners that turn eggs at least once every 2 hours to 4 hours are recommended. Humidifiers are of several types. Some are actuated by wet bulb systems while others are designed to maintain humidity by a simple water reservoir surface area system. Either of these systems can be used effectively. Temperature can be

controlled by the older wafer system or by newer microprocessor systems. Whatever the system chosen, an incubator with a backup controller set at less than 38.8 degree Celsius can save the hatch if the primary temperature controller ever malfunctions.

# BLOCK DIAGRAM

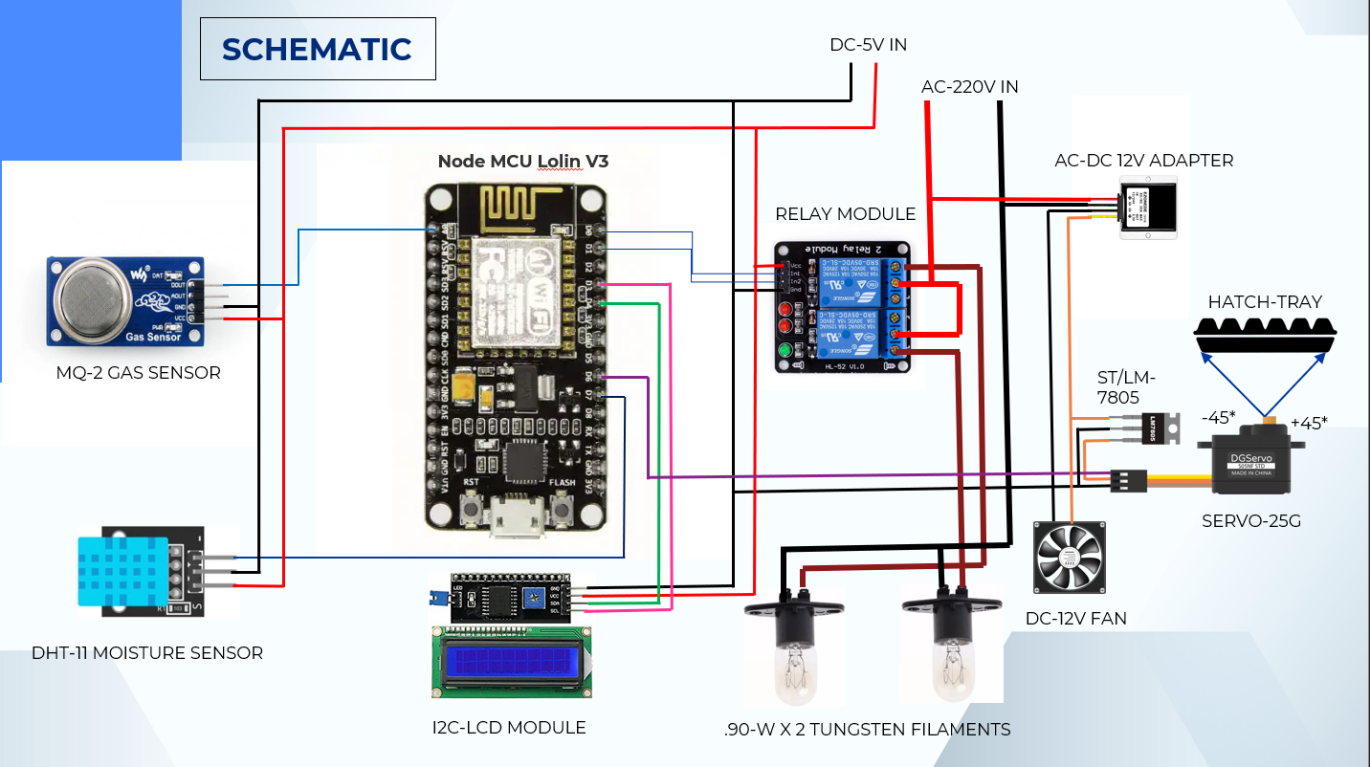


**Figure 3.1 Block Diagram of Incubator**

# EXPLANATION

After connecting all the components according to the circuit diagram, now time to turn on the circuit power supply using turn on the slide switch. When the power supply is initially given the relay is off. We need to turn on the system virtually through blynk application or blynk web dashboard. After turning on the system Node MCU start to collect the data from the Sensors. The value gets from the sensors are transmitted to the blynk cloud with the help of internet. By using the virtual pin, we can monitor the temperature and humidity level. After read the sensor data the controller going to take decision whether the filament bulb want to turn on or off. If the temperature is less than the certain temperature the relay is turn on, the bulb is turn on heat is produced else if the temperature is higher than the certain temperature relay is turn off, so the heat is not produced in the egg hatching system. In additional we can control the rotation of egg buy controlling the servo angle. For that we need to turn on the servo motor and it contains two modes. They are automatic and manual mode. When we choose the automatic mode the angle of eggs is automatically change the angle. In the manual mode the angle is controlled by manually, with the proper time difference it is necessary to change the angle. The all details like temperature, humidity and servo angle were displayed in the LCD display, which is connected to the Node MCU. We can also monitor it through blynk with the help of internet across the world.

# CIRCUIT DIAGRAM



**Figure 3.2 Circuit Diagram**

# CHAPTER 4 HARDWARE DESCRIPTION

* 1. **INTRODUCTION**

Using simple electronic components, we have implemented the smart egg hatching system with sensors and monitoring by the Blynk application.

# HARDWARE COMPONENTS

* + - Node MCU ESP8266
    - SG90 Servo Motor
    - DHT11 Sensor Module
    - MQ2 Gas sensor
    - 5V Relay
    - Tungsten Bulb
    - LCD 16\*2 Display
    - I2C converter
    - Bread Board
    - Connecting Wires
    - Zero PCB Board

# Node MCU ESP8266

NodeMCU is an open-source Lua based firmware and development board specially targeted for IoT based Applications. The choice of the DIP format allows for easy prototyping on [breadboards](https://en.wikipedia.org/wiki/Breadboard). The design was initially based on the ESP-12 module of the [ESP8266](https://en.wikipedia.org/wiki/ESP8266), which is a Wi-Fi SoC integrated with a [Tensilica](https://en.wikipedia.org/wiki/Tensilica) Xtensa LX106 core, widely used in IoT applications. The NodeMCU ESP8266 development board comes with the ESP-12E module containing the ESP8266 chip having Tensilica Xtensa 32-bit LX106 RISC microprocessor. This microprocessor supports RTOS and operates at 80MHz to 160 MHz adjustable clock frequency. NodeMCU has 128 KB RAM and 4MB of Flash memory to store data and programs. Its high processing power with in-built Wi-Fi / Bluetooth and Deep Sleep Operating features make it ideal for IoT projects. NodeMCU can be powered using a Micro USB jack and VIN pin (External Supply Pin). It supports UART, SPI, and I2C interface. This means that if you try to load libraries or use firmware from the internet, it will not work. It also doesn’t work if you try to make HTTP requests to services on the internet like publishing sensor readings to the cloud.

# Table 4.1 Technical specifications

|  |  |  |
| --- | --- | --- |
| **Pin Category** | **Name** | **Description** |
| Power | Micro-USB, 3.3V, GND, Vin | **Micro-USB:** NodeMCU can be powered through the USB port  **3.3V:** Regulated 3.3V can be supplied to this pin to power the board |

|  |  |  |
| --- | --- | --- |
|  |  | **GND:** Ground pins  **Vin:** External Power Supply |
| Control Pins | EN, RST | The pin and the button resets the  microcontroller |
| Analog Pin | A0 | Used to measure analog voltage in the  range of 0-3.3V |
| GPIO Pins | GPIO1 to  GPIO16 | NodeMCU has 16 general purpose input-  output pins on its board |
| SPI Pins | SD1, CMD,  SD0, CLK | NodeMCU has four pins available for  SPI communication |
| UART Pins | TXD0, RXD0, TXD2, RXD2 | NodeMCU has two UART interfaces, UART0 (RXD0 & TXD0) and UART1  (RXD1 & TXD1). UART1 is used to  upload the firmware/program |
| I2C Pins |  | NodeMCU has I2C functionality support but due to the internal functionality of these pins, you have to find which pin is  I2C |



**Figure 4.1 Node MCU ESP8266**

# SG90 Servo Motor

The Servo Motor Micro SG90 work well for basic servo experimentation and can be used in applications where small size is a virtue and that don’t require a huge amount of torque, but they are still pretty strong. Gears are nylon which is the case with most lower cost Servos. Servo motors can be commanded to go to a specific position and so are the usual go-to motor when accurate positioning is needed, such as for turning the front wheels on an RC model for steering or pivoting a sensor to look around on a robotic vehicle. Servo motors are comprised of a DC motor, gears, a potentiometer to determine its position and a small electronic control board. Standard servos have a specified limited range. This is usually specified as 180 degrees. Frequently the actual range is less than the full 180 degrees and is limited by the mechanical gears and potentiometer used for position sensing that is contained in the device. If the motor is run all the way to 0 or 180, it may start making unhappy sounds and start vibrating as it tries to drive to a position that it cannot get to change in angle. This causes a high stall current condition and has the potential of stripping gears and damaging the motor, pulses with values between these can be used to position the shaft arbitrarily.

# Table 4.2 Servo Motor Wire Configuration

|  |  |  |
| --- | --- | --- |
| **Wire Number** | **Wire Colour** | **Description** |
| 1 | Brown | Ground wire connected to the ground  of system. |
| 2 | Red | Powers the motor typically +5V is  used. |

|  |  |  |
| --- | --- | --- |
| 3 | Orange | PWM signal is given in through this wire to drive the motor. |



**Figure 4.2 SG90 Servo Motor**

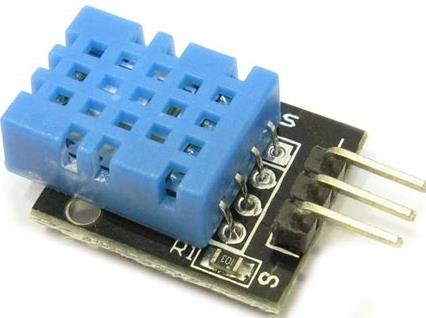
# DHT11 Sensor

The [DHT11 humidity and temperature](https://www.amazon.com/gp/product/B01DKC2GQ0/ref%3Das_li_qf_asin_il_tl?ie=UTF8&tag=circbasi-20&creative=9325&linkCode=as2&creativeASIN=B01DKC2GQ0&linkId=e60f134123941e579cc73e26d34fef60) sensor makes it really easy to add humidity and temperature data to our electronics projects. The DHT11 detects water vapor by measuring the electrical resistance between two electrodes. The humidity sensing component is a moisture holding substrate with electrodes applied to the surface. When water vapor is absorbed by the substrate, ions are released by the substrate which increases the conductivity between the electrodes. The change in resistance between the two electrodes is proportional to the relative humidity. Higher relative humidity decreases the resistance between the electrodes, while lower relative humidity increases the resistance between the electrodes. There are two different versions of the DHT11 you might come across. One type has four pins, and the other type has three pins and is mounted to a

small PCB. The PCB mounted version is nice because it includes a surface mounted 10K Ohm pull up resistor for the signal line.

# Table 4.3 DHT11 Pinout Configuration

|  |  |  |
| --- | --- | --- |
| **Pin No** | **Pin Name** | **Description** |
| 1 | Vcc | Power supply 3.5V to 5.5V. |
| 2 | Data | Outputs both Temperature and Humidity  through serial Data. |
| 3 | NC | No Connection and hence not used. |
| 4 | Ground | Connected to the ground of the circuit. |



**Figure 4.3 DHT11 Sensor Module**

# MQ-2 Gas sensor module

The mq-2 gas sensor is a popular and widely used gas sensor module that is commonly used to detect a variety of gases in the air. It is especially sensitive to flammable gases like methane, propane, butane, and natural gas. The sensor can also detect other gases such as alcohol, smoke, and certain volatile organic compounds (vocs).The mq-2 sensor module consists of a tin dioxide (sno2) semiconductor sensor element and an integrated signal conditioning circuit. The sensor element changes its resistance when it comes into contact with the target gas, and this change in resistance is then converted into an electrical signal by the conditioning circuit. It’s important to note that the mq-2 gas sensor has some limitations. Firstly, it can only detect specific gases and is not suitable for detecting all types of gases. Secondly, its accuracy and sensitivity may vary depending on factors such as temperature, humidity, and calibration. Therefore, it is advisable to calibrate the sensor for accurate readings and consider environmental factors when interpreting the sensor's output. The mq-2 gas sensor is commonly used in various applications, including gas leakage detection systems, industrial safety, fire detection systems, home automation projects, and environmental monitoring.

# Table 4.4 MQ-2 sensor Pinout Configuration



|  |  |  |
| --- | --- | --- |
| **Pin No** | **Pin Name** | **Description** |
| 1 | A0 | Send PPM as output in analog format. |
| 2 | D0 | Send PPM as output in digital format. |
| 3 | Ground | Connected to the ground of the circuit. |
| 4 | Vcc | Power supply 3.5V to 5.5V. |

# Figure 4.4 MQ-2 Gas sensor module

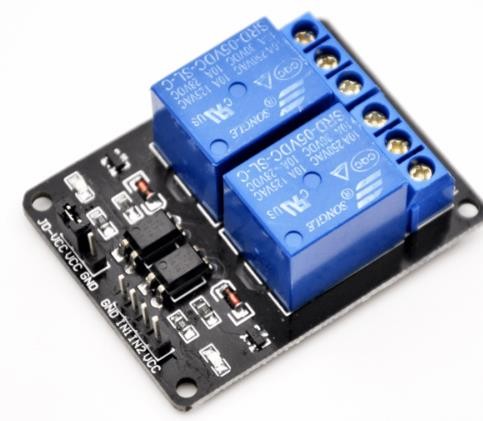
* + 1. **5V Relay**

Relays are most commonly used switching device in electronics. Trigger Voltage, this is the voltage required to turn on the relay that is to change the contact from Common-NC to Common-NO. Our relay here has 5V trigger voltage, but you can also find relays of values 3V, 6V and even 12V so select one based on the available voltage. The other parameter is your Load Voltage & Current**,** this is the amount of voltage or current that the NC, NO or Common terminal of the relay could withstand, in our case for DC it is maximum of 30V and 10A. Make sure the load you are using falls into this range. The relay has 5V trigger voltage we have used a +5V DC supply to one end of the coil and the other end to ground through a switch. We can also notice a diode connected across the coil of the relay, this diode is called the Fly back Diode**.** The purpose of the diode is to protect the switch from high voltage spike that can produced by the relay coil. As shown one end of the load can be connected to the Common pin and the other end is either connected to NO or NC. If connected to NO the load remains disconnected before trigger and if connected to NC the load remains connected before trigger.

# Table 4.5 Dual-Channel Relay Module Pinout

|  |  |  |
| --- | --- | --- |
| **Pin Number** | **Pin Name** | **Description** |
| 1 | JD-VCC | Input for isolated power supply for relay  coils. |
| 2 | VCC | Input for directly powering the relay coils. |
| 3 | GND | Input ground reference. |
| 4 | GND | Input ground reference. |
| 5 | IN1 | Input to activate the first relay. |

|  |  |  |
| --- | --- | --- |
| 6 | IN2 | Input to activate the second relay. |
| 7 | VCC | VCC to power the optocouplers, coil  drivers, and associated circuitry. |



**Figure 4.5 5V Dual channel relay module**

# Tungsten Bulb

The legacy filament bulb is commonly known as the tungsten bulb. The element has for over a century proven to be a good choice made by our forefathers. There is none like tungsten in performance. To date, the majority of electrical lighting uses a tungsten bulb. Though the technology is first changing, I don’t foresee the bulb becoming obsolete. More advancement is being made on the [bulb](https://electricalmag.com/edison-bulb/) that the principle of operation remains the same. Great future the illumination enthusiasts. All sources of light involve chemical reactions that release energy in the form of light. This piece will concentrate on tungsten filament. This is the legacy source of electrical lighting which was invented by Thomas Edison in late 19th century. Tungsten as material was selected as the best among the many that were experimented in the lab. The temperature at which they begin to emit light is different because of their chemical composition. You have

noticed charcoal emitting light when its red hot. The challenge presented with most materials was as they were heated, they melted and then evaporated and therefore ceased to produce light.



# Figure 4.6 Tungsten Bulb

* + 1. **LCD 16\*2 Display**

16×2 LCD has 16 Columns and 2 Rows. So, it will have 32 characters in total and each character will be made of 5×8 Pixel Dots. A Single character with all its Pixels is shown in the below picture. Now, we know that each character has 40 Pixels and for 32 Characters we will have 1280 Pixels. Further, the LCD should also be instructed about the Position of the Pixels. Hence it will be a hectic task to handle everything with the help of MCU, hence an Interface IC like HD44780 is used, which is mounted on the backside of the LCD Module itself. The function of this IC is to get the Commands and Data from the MCU and process them to display meaningful information onto our LCD Screen.

# Table 4.6 16x2 LCD Pinout Configuration

|  |  |  |
| --- | --- | --- |
| **Pin No** | **Pin Name** | **Description** |
| 1 | Vss  (Ground) | Ground pin connected to system ground. |
| 2 | Vdd  (+5 Volt) | Powers the LCD with +5V (4.7V – 5.3V). |
| 3 | VE  (Contrast V) | Decides the contrast level of display. Grounded  to get maximum contrast. |
| 4 | Register  Select | Connected to Microcontroller to shift between  command/data register. |
| 5 | Read/Write | Used to read or write data. Normally grounded  to write data to LCD. |
| 6 | Enable | Connected to Microcontroller Pin and toggled between 1 and 0 for data acknowledgement Connected to Microcontroller Pin and toggled  between 1 and 0 for data acknowledgement. |
| 7-14 | Data Pin 0 to Data Pin 7 | Data pins 0 to 7 forms a 8-bit data line. They can be connected to Microcontroller to send 8- bit data.  These LCD’s can also operate on 4-bit mode in such case Data pin 4,5,6 and 7 will be left free. |
| 15 | LED  Positive | Backlight LED pin positive terminal. |
| 16 | LED  Negative | Backlight LED pin negative terminal. |



**Figure 4.7 LCD 16\*2 Display**

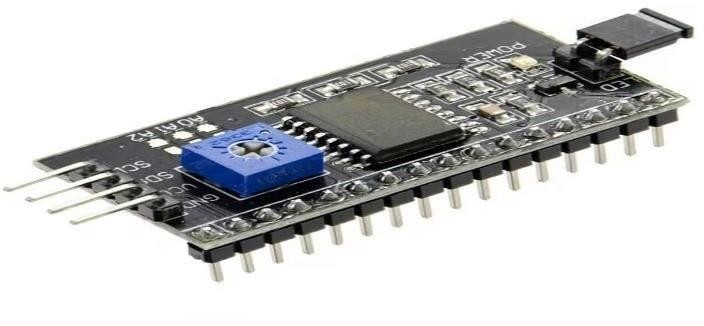
# I2C converter

I2C LCD is an easy-to-use display module, it can make display easier. I2C lcd adapter is a device containing a micro-controller PCF8574 chip. This microcontroller is a I/O expander, which communicates with other micro- controller chip with two wire communication protocol. Using this adapter anyone can control an 16x2 LCD with only two wires (SDA, SCL). It can reduce the difficulty of make, so that makers can focus on the core of the work. We developed the Arduino library for I2C\_LCD, user just need a few lines of the code can achieve complex graphics and text display features.

**Table 4.7 Pin Configuration of I2C Serial Interface Adapter Module**

|  |  |  |
| --- | --- | --- |
| **Pin Name** | **Pin Type** | **Pin Description** |
| GND | Power | Ground |
| VCC | Power | Voltage Input |
| SDA | I2C Data | Serial Data |
| SCL | I2C Clock | Serial Clock |
| A0 | Jumper | I2C Address Selection 1 |
| A1 | Jumper | I2C Address Selection 2 |

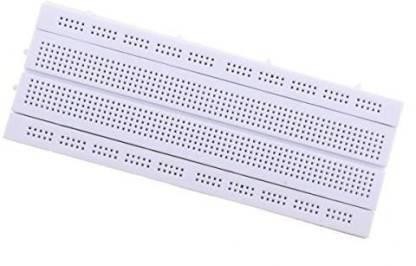
|  |  |  |
| --- | --- | --- |
| A2 | Jumper | I2C Address Selection 3 |
| Backlight | Jumper | Control Backlight of panel |



**Figure 4.8 I2C converter**

# Bread Board

A breadboard is a rectangular board with many mounting holes. They are used for creating electrical connections between electronic components and single board computers or microcontrollers such as Arduino and Raspberry Pi, breadboard, solderless breadboard, or protoboard is a construction base used to build semi-permanent prototypes of electronic circuits. Unlike a perf board or stripboard, breadboards do not require soldering or destruction of tracks and are hence reusable. For this reason, breadboards are also popular with students and in technological education.



# Figure 4.9 Bread Board

* + 1. **Connecting Wires**

Jumper wires are simply wires that have connector pins at each end, allowing them to be used to connect two points to each other without soldering. Jumper wires are typically used with breadboards and other prototyping tools in order to make it easy to change a circuit as needed. Fairly simple. Connecting wires allows an electrical current to travel from one point on a circuit to another because electricity needs a medium through which it can move. Most of the connecting wires are made up of copper or aluminum. A connecting wire is represented by a straight line. It is usually made of copper and is provided with insulation to make electrical connections between two points.

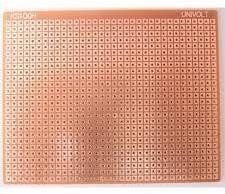


# Figure 4.10 Connecting Wires

* + 1. **Zero PCD Board**

Zero PCB is basically a general-purpose printed circuit board (PCB), also known as preboard or DOT PCB. It is a thin rigid copper sheet with holes predrilled at standard intervals across a grid with 2.54mm (0.1-inch)

spacing between holes. A blank PCB is quite simply an empty circuit board free from any of the components that are installed to create a functioning circuit board. A blank circuit board is sometimes known as a 'copper-clad' circuit board, due to the coating of copper the board has around it.



# Figure 4.11 Zero PCB Board

**CHAPTER 5 SOFTWARE DESCRIPTION**

# ARDUINO IDE

The Arduino Integrated Development Environment or Arduino Software (IDE) contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them. The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used withany Arduino Supported Boards such as ESP32 and other similar boards. It isavailable for all operating systems i.e., MAC, Windows, Linux and runs on the Java Platform that comes with inbuilt functions and commands that play a vital role in debugging, editing and compiling the code. A range of Arduinomodules available including Arduino Uno, Arduino Mega, Arduino Leonardo, [Arduino Micro](https://www.theengineeringprojects.com/2018/09/introduction-to-arduino-micro.html) and code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board. The IDE environment mainly contains two basic parts: Editor and Compiler where former is used for writing the required code and later is used for compiling and uploading the code into the given Arduino Module.



# Figure 5.1 Arduino IDE

* 1. **BLYNK IOT PLATFORM**

To control this Project, we used the BlynkIoT Platform which provides us simplicity to operate complex HVAC systems & the fullback- end IOT cloud support to run our Android and WEB Apps on their platform. It has support for IOS apps also. It’s a digital dashboard where you can build a graphic interface for your project by simply dragging and dropping widgets. Blynk HTTP RESTful API allows users to easily read and write values to/from Pins in Blynk apps and Hardware (microcontrollers and microcomputers like Arduino, Raspberry Pi, ESP8266, Particle, etc.).

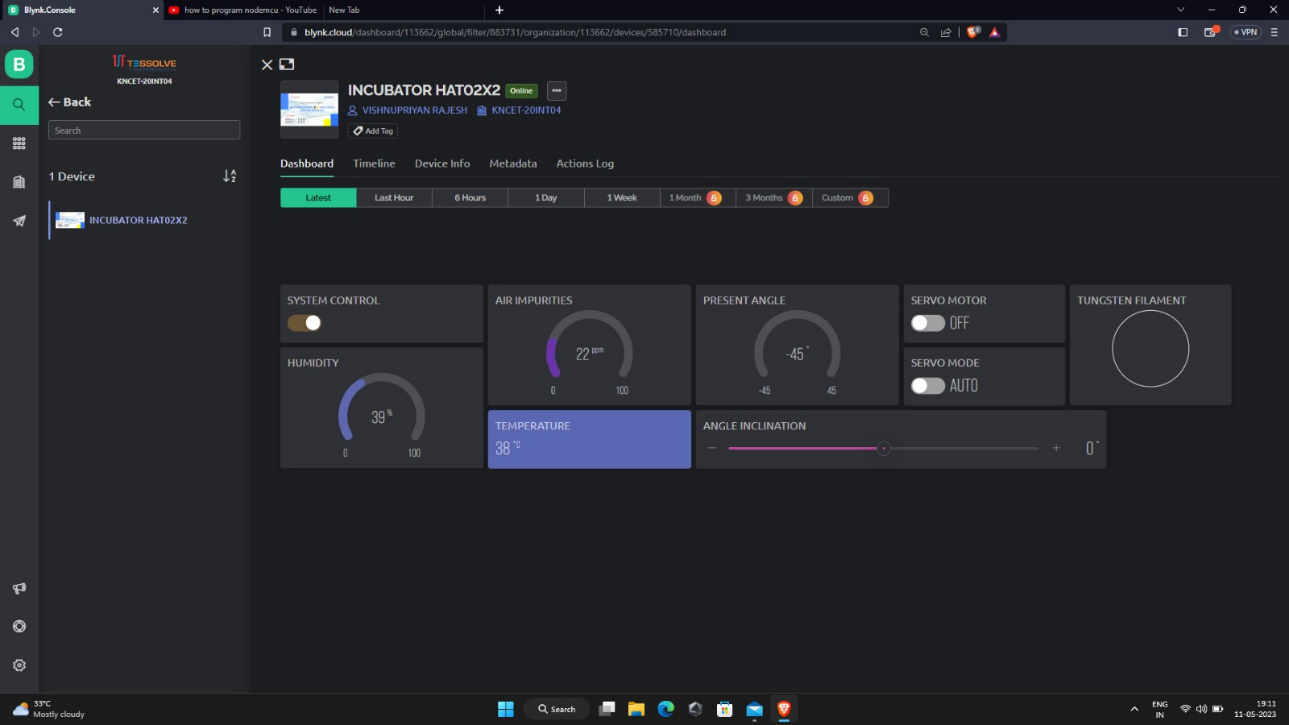
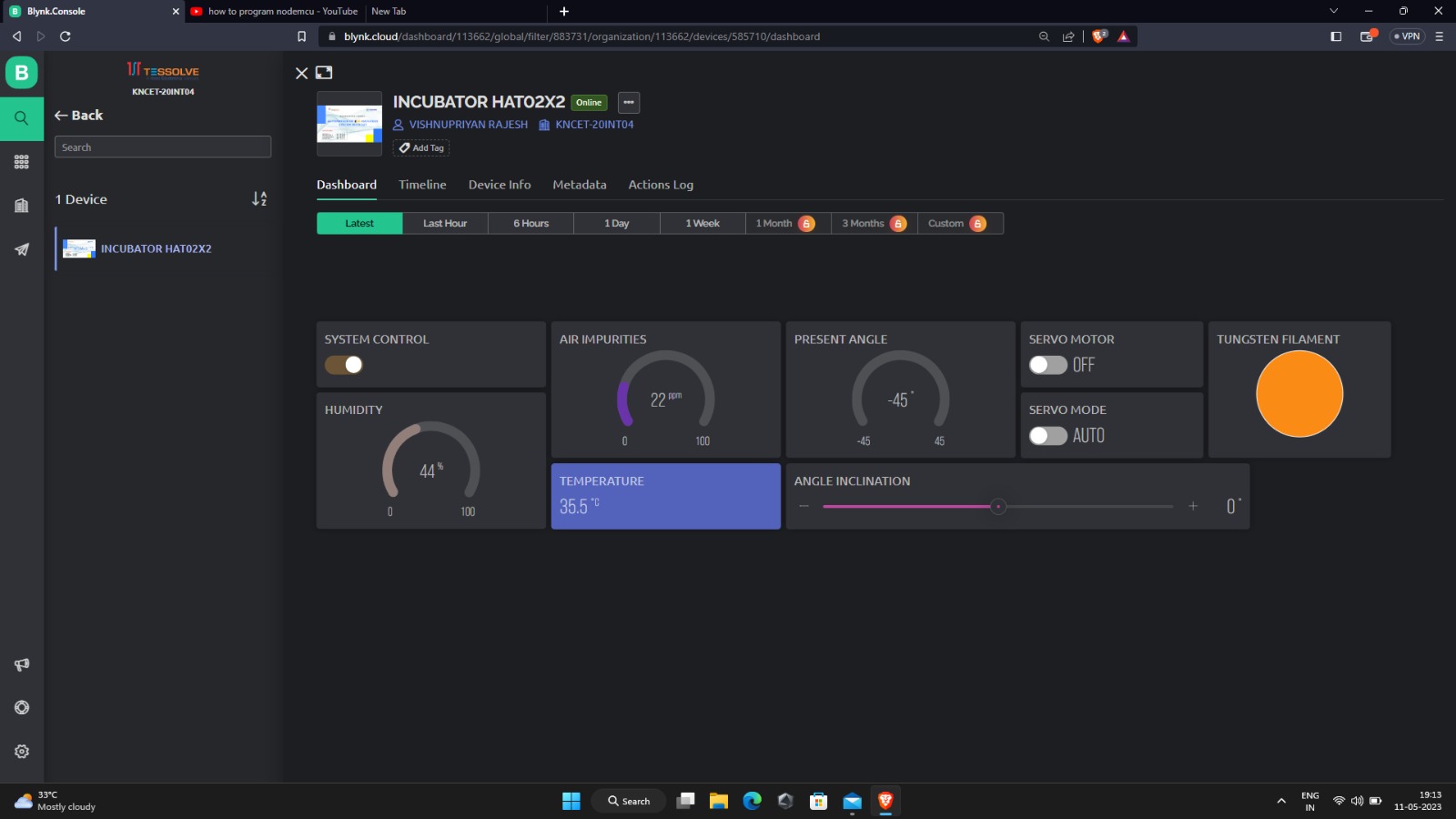


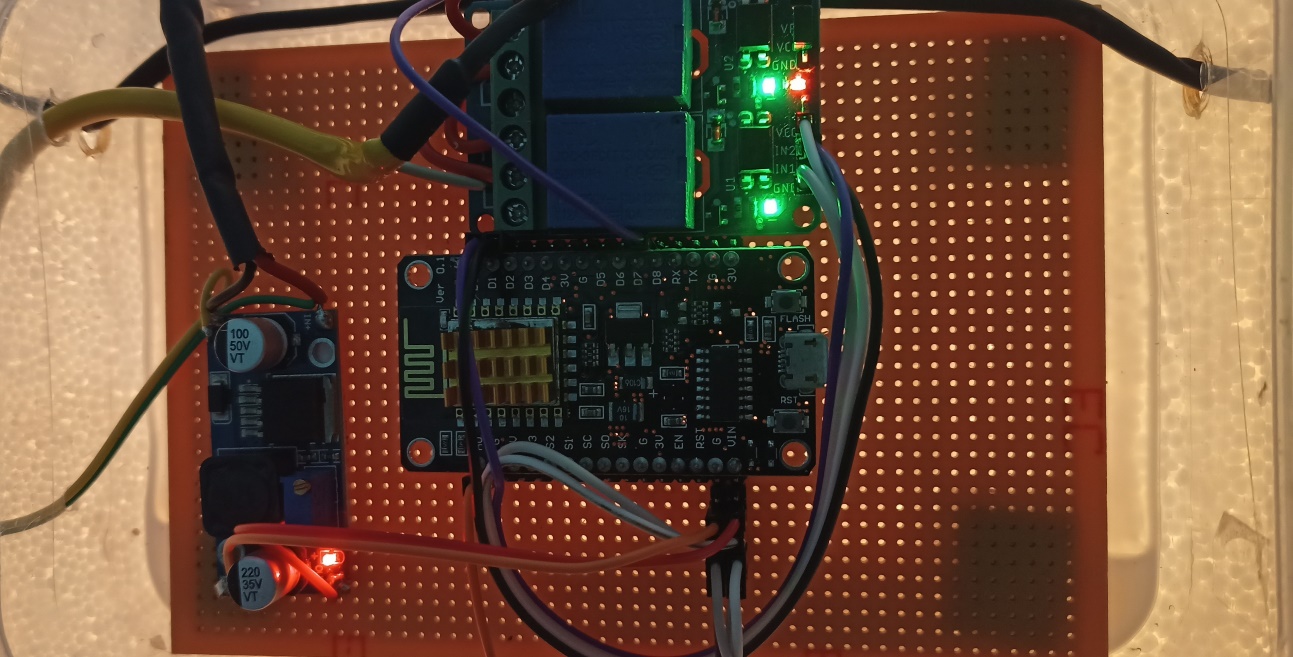
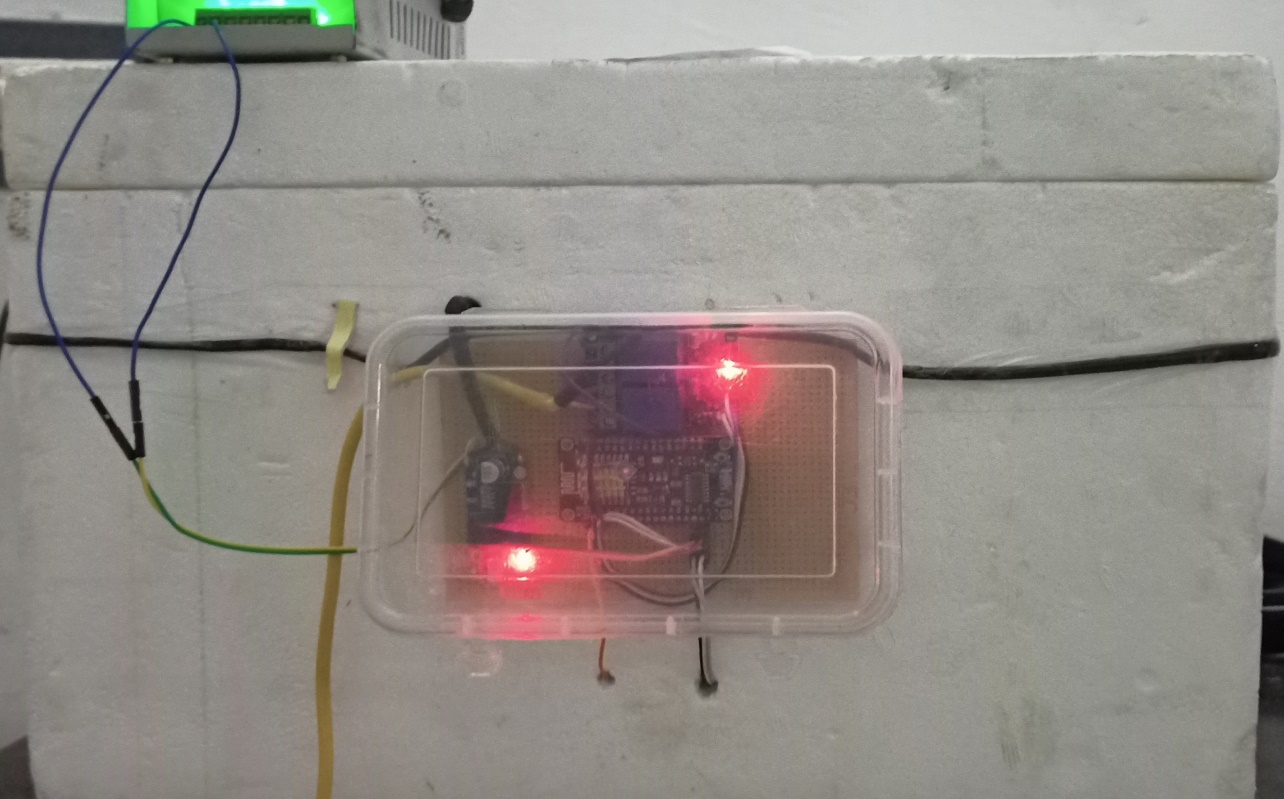
# Figure 5.2 Blynk

Every PUT request will update Pin's state both in apps and on the hardware. Every GET request will return the current state/value on the given Pin. Here developer also provide a simplified API so user can do updates via getting requests.

# CHAPTER 6 RESULT AND DISCUSSION

**HARDWARE AND SOFTWARE IMPLEMENTATION**





**CHAPTER 7**

# CONCLUSION AND FUTURE ENHANCEMENT

Monitoring and controlling a smart egg hatching system via the internet of things is successfully developed and implemented in this work. Smart controller is designed by using fuzzy controller to ensure that the suitable conditions are available inside the incubator for healthy growth of the embryo. By using the IoT technology, the egg incubator can be controlled and monitored by using simple blynk application from any personal computer or smart phone around the world with proper authorization. The designed blynk dashboard has the facility for changing the set point such as temperature set point, humidity set point and the long of incubation period of the local egg incubator. The system proved smart by being able to monitor, report and automatically regulate its internal environment within the required limits without the farmer’s presence and periodically turn eggs inside of it as such, ensuring that the eggs were well incubated thus producing high hatchability. Therefore, the designed smart egg incubator can be used for any type of the egg and can be controlled by anywhere through Internet. Besides that, another recommended is the solar energy should be used as backup power supply, it is because of the untimely failure of electricity to enhance the efficiency of the smart egg hatching system.

# APPENDIX

#define BLYNK\_TEMPLATE\_ID "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

#define BLYNK\_TEMPLATE\_NAME "EGG INCUBATOR"

#define BLYNK\_AUTH\_TOKEN "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

#include <BlynkSimpleEsp8266.h>

#include <ESP8266WiFi.h>

#include "DHT.h"

#define DHTTYPE DHT11

#include <Servo.h>

Servo myservo;

DHT dht(D5, DHTTYPE);

const char\* ssid = "\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

const char\* password = "\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

float t = 0;

int po = 90, h, i = 0, a, sy\_c, sv\_c, sv\_a;

WidgetLED led1(V8);

BlynkTimer timer;

void setup()

{

Serial.begin(9600);

WiFi.begin(ssid, password);

Blynk.begin(BLYNK\_AUTH\_TOKEN, ssid, password);

dht.begin();

myservo.attach(D4);

pinMode(D0, OUTPUT);

pinMode(D1, OUTPUT);

myservo.write(po);

pinMode(A0, INPUT);

}

BLYNK\_WRITE(V0)

{

sy\_c = param.asInt();

}

BLYNK\_WRITE(V3)

{

sv\_c = param.asInt();

}

BLYNK\_WRITE(V4)

{

sv\_a = param.asInt();

}

BLYNK\_WRITE(V5)

{

po = param.asInt();

}

void callsensor()

{

h = dht.readHumidity();

t = dht.readTemperature();

if (t >= 37.5)

{

digitalWrite(D1, HIGH);

digitalWrite(D0, HIGH);

led1.off();

}

else if(t <=35.5)

{

digitalWrite(D0, LOW);

digitalWrite(D1, LOW);

led1.on();

}

Blynk.virtualWrite(V1, h);

Blynk.virtualWrite(V2, t);

}

void servoin()

{

a = map(po, 0, 180, -45, 45);

myservo.write(po);

Blynk.virtualWrite(V6, a);

po += 1;

i++;

}

void servode()

{

a = map(po, 0, 180, -45, 45);

myservo.write(po);

Blynk.virtualWrite(V6, a);

po = 1;

i++;

}

void servo\_control()

{

if(sv\_c==1)

{

if(sv\_a==1)

{

a=map(po,-45,45,0,180);

myservo.write(a);

Blynk.virtualWrite(V6,po);

}

else if(sv\_a==0)

{

if(i<=180)

{

servoin();

delay(500);

}

else if(i<=360)

{

servode();

delay(500);

}

else if(i>360)

{

i=0;

po=0;

}

}

}

else if(sv\_c==0)

{

po=90 ;

a=map(po,0,180,-45,45);

myservo.write(po);

Blynk.virtualWrite(V6,a);

}

}

void read\_MQ2\_sensor()

{

int mq2Value = analogRead(A0);

float ppm = (mq2Value / 10.0) \* 1.5;

Blynk.virtualWrite(V7, ppm);

}

void loop()

{

if (sy\_c==1)

{

callsensor();

servo\_control();

read\_MQ2\_sensor();

}

Blynk.syncVirtual(V0);

Blynk.syncVirtual(V3);

Blynk.syncVirtual(V4);

Blynk.syncVirtual(V5);

Blynk.run();

}

# REFERENCES

* 1. Abdul-Rahaim. L. A, Mohammed. A and Ali. A (2015), “Remote Wireless Automation and Monitoring of Large Farm using wireless sensors networks and Internet,” International Journal of Computer Science & Engineering Technology (IJCSET), vol. 6, no. 03, pp. 118–137.
  2. Ali. F and Amran. N. A (2016), “Development of an Egg Incubator Using Raspberry Pi for Precision,” International Journal of Agriculture, Forestry and Plantation, vol. 2, pp. 40–45.
  3. Catwright. A. L (2000), “Incubating and Hatching Eggs,” The Texas A&M University System, pp. 1–12.
  4. Divoky. G. J and Harter. B. B (2010), “Supernormal Delay in Hatching, Embryo Cold Tolerance and Egg-fostering in the Black Guillemot Cepphus Grylle,” Marine Ornithology 38: 7–10.
  5. Fordjour. H. D, Hamidu. J. A and Adomako. K (2017), “Assessing Incubation and Performance Deficiencies to Boast Broiler Production,” American Research Journal of Agriculture, vol. 3, pp:1-6.
  6. Garcia-hierro. J, Robla. J. I, Barreiro. P, Correa-hernando. E. C and Diezma. B (2012), “Design of a Solar Incubator, Part 1: Monitoring Temperature and Enthalpy Gradients Under Commercial Production,” International Conference of Agricultural Engineering, pp. 1–6.
  7. Harb, Habbib. S. K, Kassem Y. A and Raies. A (2010), “Energy Consumption for Poultry Egg Incubator to Suit Small Farmer,” Egypt Journal Agricultural Research, vol. 88, no.1, pp. 193-210.
  8. Kusi. L. Y, Agbeblewu. A. I and Minta Nyarku. K (2015), “The Challenges and Prospects of the Commercial Poultry Industry in Ghana: A Synthesis of Literature,” International Journal of Management Sciences, vol. 5, no.6, pp.

476-489.

* 1. Mashhadi. S. K. M, Yekan. J. G. D and Dashtaki. M. G. N (2012), “Incubator with Fuzzy Logic,” The Journal of Mathematics and Computer Science, vol. 3, no. 3, pp. 197–204.
  2. Okpagu. P. E and Nwosu. A. W (2016), “Development and Temperature Control of Smart Egg Incubator System for Various Types of Egg,” European Journal of Engineering and Technology, vol. 4, no.2, pp. 2056- 5860.
  3. Olaoye. I. O, Lawal. B. M, Ibrahim. S.O and Sanusi. B. A (2013), “An Electrically Operated Incubator for Household,” Greener Journal of Science and Technological Research, vol. 3, no 5, pp. 160-165.
  4. Oliveira. S. G and Santos. M. V. D (2020), “Effects of different egg turning frequencies on incubation efficiency parameters,” Poultry Science, vol. 99, pp. 4417-4420.
  5. Pereira. G. T, Nakage. E. S and Cardozo. J. P (2003), “Effect of Temperature on Incubation Period, Embryonic Mortality, Hatch Rate, Egg Water Loss and Partridge Chick Weight,” Brazilian Journal of Poultry Science, vol. 5, no.2, pp: 131-135.
  6. Radhakrishnan. K, Jose. N, Sanjay. S. G, Cherian. T and Vishnu. K. R (2014), “Design and Implementation of a Fully Automated Egg Incubator,” International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 3, no. 2, pp. 7686–7690.
  7. Sanjaya. W. S. M (2018), “The development of quail eggs smart incubator for hatching system based on microcontroller and Internet of Things (IoT),” 2018 International Conference on Information and Communications Technology (ICOIACT), Yogyakarta, pp. 407- 411.
  8. Shittue. S, Muhammad. A. S and Jimoh. O (2016), “Development of an Automatic Bird-Egg Incubator,” A Journal of Embedded System &

Applications. vol. 5, no.1, pp:1–10.

* 1. Sunday. A.A , Ogunbode. O. A, Babatunde. E. G and Olalekan. A. M (2020), “Design and Construction of Automated Eggs Incubator for Small Scale Poultry Farmers” International Journal of Technical Research & Science, vol. 5, pp. 2454-2024.
  2. Sunitha. H, Niranjan. L, Rajesh. D. P. B, Pooja. R and Supritha. B. K (2020), “Universal Egg Incubation System for Hatching using Atemga328P, Proteus Design Tool and IoT”, International Journal of Research and Analytical Reviews (IJRAR), vol. 7, pp. 2349-5138.
  3. Tolentino. K. S, Justine. E, Enrico. G, Listanco. R. L. M, Anthony. M, Ramirez. M, Renon. T. L. U, Rikko. B and Samson. B (2018), “Development of Fertile Egg Detection and Incubation System Using Image Processing and Automatic Candling”, TENCON 2018, IEEE Region 10 Conference, pp. 0701-0706.
  4. Umar. A. B, Lawal. K, Mukhtar. M and Adamu. M. S (2016), “Construction of an Electrically Operated Egg Incubator”. International Journal of Modern Engineering Sciences, vol.5, no.1, pp 1-18.