**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | **PAGE NO.** |
|  | **ABSTRACT** | **iv** |
|  | **TABLE OF CONTENTS** | **vi** |
|  | **LIST OF FIGURES** | **viii** |
|  | **LIST OF ABBREVIATIONS** | **ix** |
|  |  |  |
| **1** | **INTRODUCTION** | **9** |
|  | 1.1 GENERAL | 9 |
|  | 1.2 PROBLEM STATEMENT | 11 |
|  | 1.3 OBJECTIVES | 11 |
|  | 1.4 ORGANIZATION OF THE THESIS | 12 |
| **2** | **LITERATURE SURVEY** | **14** |
| **3** | **SYSTEM ANALYSIS** | **24** |
|  | 3.1 EXISTING SYSTEM | 24 |
|  | 3.2 DRAWBACKS | 24 |
|  | 3.3 PROPOSED BLOCK DIAGRAM | 25 |
|  | 3.4 PROPOSED SYSTEM EXPLANATION | 25 |
|  | 3.4 APPLICATIONS | 26 |
| **4** | **SYSTEM SPECIFICATION** | **29** |
|  | 4.1 SOFTWARE REQUIREMENTS | 29 |
|  | 4.2 SOFTWARE ENVIRONMENT | 29 |
|  | 4.2.1 ARDUINO SOFTWARE (IDE) | 29 |
|  | 4.2.2 EMBEDDED C | 31 |
|  | 4.2.3 PROTEUS | 33 |
| **5** | **SYSTEM DESIGN** | **35** |
|  | 5.1 SYSTEM ARCHITECTURE | 35 |
|  | 5.2 DATA FLOW DIAGRAM | 38 |
| **6** | **SYSTEM IMPLEMENTATION** | **40** |
|  | 6.1 HARDWARE REQUIREMENTS | 40 |
|  | 6.2 HARDWARE ENVIRONMENT | 40 |
|  | 1. POWER SUPPLY UNIT | 40 |
|  | 1. NODE MCU V3 | 46 |
|  | 1. LCD DISPLAY | 49 |
|  | 1. DHT 11 HUMIDITY & TEMPERATURE SENSOR | 51 |
|  | 1. SERVO MOTORS | 53 |
|  | 1. RELAY | 56 |
|  | 1. D.C MOTOR | 58 |
| **7** | **MODULE DESCRIPTION** | **63** |
|  | * 1. MODULES | 63 |
| **8** | **SYSTEM TESTING** | **66** |
|  | 8.1 TYPES OF TESTS | 66 |
|  | 8.1.1 UNIT TESTING | 66 |
|  | 8.1.2 INTEGRATION TESTING | 66 |
|  | 8.1.3 VALIDATION TESTING | 67 |
| **9** | **IMPLEMENTATION AND RESULTS** | **69** |
|  | 9.1 HARDWARE PHOTOGRAPHY | 69 |
| **10** | **CONCLUSION AND FUTURE SCOPE** | **72** |
|  | 10.1 CONCLUSION | 72 |
|  | 10. 2 FUTURE SCOPE | 72 |
|  | **REFERENCES** | **73** |
|  | **APPENDIX** | **75** |

**ABSTRACT**

****

**ABSTRACT**

This paper presents the design and implementation of an automated poultry farming system aimed at improving operational efficiency and ensuring optimal living conditions for poultry.

The proposed system integrates several key components, including a pump motor spray system for humidity control, moisture sensors to regulate AC lamps and maintain appropriate lighting and warmth, DC fans and exhaust units for effective ventilation, and a timer-based automated feeding mechanism to ensure consistent and timely nourishment.

By continuously monitoring and adjusting temperature, humidity, and air quality, the system minimizes the need for manual intervention. This not only enhances the overall efficiency of the farming process but also promotes the healthy growth of poultry by maintaining a stable and comfortable environment. The automation ensures precise control over critical environmental factors, reduces human error, and supports scalable and sustainable poultry farming operations.

**LIST OF ABBREVIATIONS**

|  |  |  |
| --- | --- | --- |
| **FIG NO.** | **TITLE** | **PAGE NO.** |
|  | Main System Goals. |  |
|  | Existing Model Diagram |  |
|  | Proposed Model Diagram |  |
|  | System Architecture |  |
|  | Software Installing |  |
|  | Architecture |  |
|  | Diagrams For Power Supply |  |
|  | Fixed Voltage Regulators |  |
|  | Block Diagram Of Power Supply |  |
|  | Node MCU V3 Pinout |  |
|  | Pin Configuration To Use In Arduino IDE |  |
|  | Liquid Crystal Display |  |
|  | Dth 11 Sensor |  |
|  | Dth 11 Sensor Pin Out |  |
|  | Servo Motor |  |
|  | Woring Of Servomotor |  |
|  | Controller Of Servo Motor |  |
|  | Circuit Diagram For Relay Off |  |
|  | Circuit Diagram For Relay On |  |
|  | Dc Motor |  |
|  | The Basic Principle Of Motor Action |  |
|  | The Armature Winding |  |

**LIST OF ABBREVIATIONS**

|  |  |
| --- | --- |
| **ACRONYM** | **ABBREVIATIONS** |
| IDE | Integrated Development Environment |
| AVR | Alf and Vegard’s RISC processor |
| COBOL | Common Business Oriented Language |
| RMS | Root Mean Squared |
| ADC | Analog to Digital Converter |
| LNG | Liquefied Natural Gas |
| USB | Universal Serial Bus |
| PWM | Pulse Width Modulation |
| ICSP | In Circuit Serial Programming |
| MEMS | Micro Electro Mechanical Systems |
| UART | Universal Asynchronous Reception and Transmission |
| MCU | Micro Controller Unit |
| IR | Infrared Ray |
| LI-ION | Lithium Ion |
| SVM | Support vector machine |
| MFCC | Mel Frequency Cepstral Coefficients |

# INTRODUCTION

****

**CHAPTER 1**

**INTRODUCTION**

**1.1 GENERAL**

Poultry production alleviates protein deficiency and improves the economic standing of the agricultural system. Hatching eggs can be carried out on a small, big, or commercial scale. In rural areas, farmers transport the eggs to a commercial hatchery and pick up the chicks 21 days later for a price based on an agreed-upon cost per hatching egg. An alternative option is using electrical incubators to simulate and imitate the function of the mother hen in natural incubation. Electrical incubators provide fertile eggs with optimal environmental conditions to stimulate embryonic development until hatching [6]. The hatchery process uses an artificial method of egg incubation that can completely mimic the hen's behavior during the development phase[7]. Research suggests that heat stress causes significant economic losses in the poultry industry annually[8]. Regular checking of temperature and humidity are maintained at a predetermined level to form embryonic fluids. A poultry egg electrical incubator creates an artificial environment with the necessary temperature control method for developing embryos in fertilized eggs to mimic natural brooding. Small and medium-sized poultry farmers may use ineffective manual procedures to manage their operations [9]. The Internet of Things (IoT) is a rapidly developing technology gaining significance in smart farming. The IoT is a network of interconnected devices that automate various processes while communicating over the Internet. As IoT gains importance in various smart agricultural applications, leading technology companies invest in and support agricultural innovation development [10]. In the Fourth Industrial Revolution (4IR), a solution for the incubator's control and monitoring system can be found by implementing the Internet of Things[11]. Smart temperature-controlled incubation of poultry eggs necessitates a sustainable energy supply for optimal performance, operation, and profitability. The egg incubation process covers the management of fertilized poultry eggs to a satisfactory level of development, resulting in healthy chicks [12]. Environmental parameters are monitored in the system as described in this study [13]. The poultry farm management system saves time and money by reducing labor, improving the health and growth of chicks, and increasing egg production. A Cloud-based IoT monitoring solution for poultry farming was introduced in [14]. Aside from the lowered costs of operating a chicken farm, an IoT system leverages a remote server that can analyze and compute data graphically. This study aims to develop a temperature and humidity monitoring system for poultry egg incubators with IoT. The project was developed to assist small-scale poultry producers with monitoring the parameters involved in the incubation phase of poultry eggs. A cycle between design, development, and testing was executed to produce the final output of the study. The loop on the three processes enabled a series of iterations to satisfy the specified requirement for the system. Experimentation within the context of an innovation process using prototypes.

****

**Figure 1: Main System Goals.**

**1.2 PROBLEM STATEMENT**

In poultry farming, maintaining optimal conditions for egg incubation is crucial for maximizing hatch rates and ensuring healthy chick development. Traditional incubation methods require constant manual monitoring of temperature, humidity, and feeding schedules, which can be labor-intensive and prone to human error. Inconsistent temperature and humidity levels can lead to poor hatching rates, while irregular feeding can impact the growth and well-being of newly hatched chicks. To address these challenges, this project proposes the development of an automated Chicken Incubator Monitoring System that integrates a temperature sensor to regulate a water pump for humidity control, a humidity sensor to activate a warm lamp for maintaining ideal incubation conditions, and a timer-based system for automated feeding. By automating these essential incubation parameters, the system aims to improve efficiency, reduce human intervention, and ensure a more controlled and stable environment for egg incubation and chick rearing.

**1.3OBJECTIVES**

1. **Monitor and Control Temperature** – Use a temperature sensor to detect fluctuations and activate a water pump to regulate the incubator’s internal temperature.
2. **Maintain Optimal Humidity** – Implement a humidity sensor to monitor moisture levels and trigger a warm lamp when necessary to maintain proper incubation conditions.
3. **Automate Feeding Mechanism** – Use a timer-based system to dispense food at scheduled intervals, ensuring chicks receive adequate nutrition.
4. **Ensure Consistent Egg Incubation Conditions** – Maintain stable environmental conditions to maximize egg hatchability and chick survival rates.
5. **Real-time Data Collection** – Continuously collect and log temperature and humidity data for analysis and performance tracking.
6. **User-friendly Interface** – Provide an easy-to-use monitoring interface (LCD, mobile app, or web-based) to allow farmers to track incubation parameters.
7. **Alarm and Alert System** – Notify users through alarms or notifications when temperature or humidity levels exceed safe thresholds.
8. **Energy-efficient and Reliable Operation** – Design the system to function efficiently with minimal energy consumption while ensuring continuous operation.

**1.4 ORGANIZATION OF THE THESIS**

This project report has been organized into six chapters. The first chapter described the introduction, problem statement, and objectives of the work. The remaining chapters are given below.

**Chapter 1:** To provide an introduction of the project.

**Chapter 2**: Provides the various literature works reviewed.

**Chapter 3:** Describes the existing system relevant to

**Chapter 4:** Covers the presents the block diagram, explains about the proposed work, hardware and description used for this work.

**Chapter 5:** Presents implementation of the hardware and the results.

# LITERATURE SURVEY

****

**CHAPTER 2**

**LITERATURE SURVEY**

Several studies and innovations have contributed to the development of automated incubation systems. Research highlights the importance of maintaining a stable temperature range (typically between 37.5°C and 38°C) for optimal embryonic development. Various sensor-based solutions, such as DHT22 for humidity control and DS18B20 for temperature regulation, have been implemented in previous works to maintain optimal incubation conditions. Automation in poultry incubation has led to higher hatchability rates and reduced chick mortality by ensuring precise environmental control.

Recent advancements in IoT (Internet of Things) have further improved incubator efficiency by enabling real-time monitoring and remote control via mobile applications. Studies have demonstrated that integrating water pumps with temperature sensors helps maintain the required humidity levels inside incubators, preventing dehydration of eggs. Additionally, timer-based food dispensers have been successfully used in poultry farming to provide chicks with regular feeding schedules, promoting healthy growth. These findings indicate that an integrated system combining temperature, humidity, and feeding automation can significantly enhance poultry incubation efficiency and overall farm productivity.

**CONVENTIONAL METHOD**

**[1] AN IOT MONITORING ASSISTANT FOR CHICKEN LAYER FARMS - Ron Daniel M. Nicolas, Wei S. Zhou, Shota C. Kitamura, Mary Jane C. Samonte - 2019 International Conference on Information and Communication Technology Convergence (ICTC)**

Chicken eggs are one of the most common staple foods being consumed almost all over the world. Optimization of egg production is important to address the possibility of shortage in the near future. The application of IoT and its possible benefits to the poultry industry could serve as a base for cost-effective solutions and as a proof-of-concept for scalable monitoring systems that farm owners may use in lieu of already existing methods. This study includes development of a device that is connected to an application to serve as a monitoring assistant for chicken layer farms. The system displays humidity and temperature and controls the lighting of an enclosed vent style coop. The data aggregated is used to compare to historical weather conditions to serve as a decision-support for chicken egg for a better plan of chicken warehouse. Upon testing of the developed system results show relevant advantage in poultry cooling and management plan in maintaining optimum conditions of chicken layer farms.

**[2] ANIMAL BEHAVIOR FOR CHICKEN IDENTIFICATION AND MONITORING THE HEALTH CONDITION USING COMPUTER VISION: A SYSTEMATIC REVIEW - Md Roman Bhuiyan, Philipp Wree - IEEE Access ( Volume: 11)**

Poultry farming needs to be productive and profitable if it is to help with food security issues like cost. To make money economic management and the use of productivity standards like the Feed Conversion Ratio (FCR) calculation are essential. So the best way to improve the performance of chickens is to use best management practices while they are growing. The paper also gives a broad overview of the vital ways in which recognized digital technologies are used to control the well-being of chickens. This review article gives a thorough look at the research with a focus on sensor-based AI applications for analyzing chicken behavior and keeping track of their health. This study focused on the welfare of poultry because there are currently gaps in the literature regarding the development of universally accepted criteria for assessing poultry well-being and developing trustworthy monitoring strategies most notably for the health of broiler chickens and the prevention of disease outbreaks. This paper is focused on the current condition of the smart farming and how to improve the smart farming sector using computer vision with IoT. The current and future operations for poultry management provide a huge opportunity for intelligent automation which if implemented would make it possible to produce chicken that is both high-quality and affordable. As a result this research provides a thorough analysis of the most cutting-edge Internet of Things (IoT) systems with Artificial intelligence (AI) capabilities as well as their most recent developments in the creation of intelligent systems in this field. The paper also explores the benefits and problems that AI and IoT provide for chicken farming.

**[3] RESEARCH ON INTELLIGENT CONTROL TECHNOLOGY OF CHICKEN HOUSE ENVIRONMENT IN LARGE-SCALE CHICKEN FARMS - Qingle Quan, Thelma D. Palaoag, Hanqing Sun - 2024 7th International Conference on Communication Engineering and Technology (ICCET)**

In order to solve the problem of the influence of dust particles and toxic gases on the growth of chickens in chicken houses faced by large-scale chicken farms this study aims to develop an intelligent control system for chicken house environment in chicken farms and to explore the solution of intelligent control of chicken house environment by utilizing the Internet the Internet of Things (IoT) and high-voltage electrostatic and plasma technologies.This study uses a function-driven development approach using basic hardware such as sensor fusion hardware communication equipment high-voltage electrostatic precipitator and plasma discharge device and completes the system function development using the ROS (Robot Operating System) embedded development platform and the Java language to determine the usability and reliability of the system by running tests in chicken farms.The core functions of the system include environmental de-dusting and purification by adjusting the strong discharge field and deodorization and disinfection purification by the plasma reactor. The high-voltage electrostatic discharge technology designed in this study can realize the automatic adjustment function and the discharge docking area and discharge gap can be changed by rotating the button so as to change the discharge field strength and achieve different purification design requirements. The plasma reactor adopts a honeycomb structure the blocking medium is enameled in order to ensure uniform discharge the conductive layer is plated onto the enamel plate the other side of the electrode is made of planed 316 stainless steel plate the high-speed phase-locked loop MM74HC4046 is used to lock the operating frequency and the power is automatically adjusted using the PFM method and the resonant capacitor adopts the new type of capacitor that has a large capacity and a small ESR of the equivalent series resistance. Through the actual operation test in the chicken farm users are satisfied with the usability ease of use and reliability of the system and equipment.

**[4] RESEARCH AND DESIGN OF INTELLIGENT INSPECTION ROBOT FOR LARGE-SCALE CHICKEN FARMS - Qingle Quan, Thelma D. Palaoag, Hanqing Sun - 2024 5th International Conference on Machine Learning and Human-Computer Interaction (MLHMI)**

In order to solve the problems of chicken status monitoring and farming decision making faced by intelligent livestock and poultry farming this study aims to develop an intelligent inspection robot for chicken farms to explore the solution of intelligent inspection of chicken coops using Internet information technology and Internet of Things (loT) technology. This study uses a function-driven development approach using basic hardware such as LiDAR remote sensing technology devices sensor fusion hardware communication devices power supply motors and actuators and ROS (Robot Operating System) embedded development platform and Python language to complete the functional development of the system and through the running test in the chicken farm to determine the systems usability and The system was tested in a chicken farm to determine the usability and reliability of the system.Functionality includes the use of LiDAR infrared sensors ultrasonic sensors etc. to realize basic data acquisition in the chicken house and a sensor fusion unit to integrate data from different sensors into a common environment model to create a comprehensive view of the robots surroundings. The robot uses the sensor data to determine its current position which is typically accomplished through LiDAR and map matching. The robot monitors the sensor data in real time to detect the presence of obstacles or changes in position for real-time obstacle avoidance. The robot can communicate with a remote operator or a centralized control system to transmit real-time data and receive task instructions. The robot can feedback the task execution and collected data to the operator or central system for further analysis and decision making. Users were satisfied with the usability ease of use and reliability of the system through actual operational tests on the farm.

**[5] SMART CHICKEN FARMING; CASE STUDY IN NORTH RIFT KENYA - Allan K. Koech, Fidel Makatia, Valery Chebet - 2023 IEEE Conference on AgriFood Electronics (CAFE)**

Agriculture has long been the backbone of survival for many families living in remote areas of Kenya. Among these there has been small- to mid-scale chicken rearing for broiler and egg production. However despite the traditional practices used to boost productivity there still needs to be more timely feeding cleaning of water sources and medication of the chicks. In addition with predator birds such as hawks preying on the chicks the survival of such birds in the community increases the burden of continuous monitoring as they feed in the open. As a result the population that survives the growth stage diminishes resulting in low productivity in the long run. This paper addresses this through smart chicken rearing for small- and mid-scale farmers who rely on this for survival. This seeks to develop a model system that will allow control of feeding watering security and heating systems allowing the collection of vital data to be used in monitoring productivity and easing physical monitoring for the chicks.

**[6] CHICKEN FARM MONITORING SYSTEM - Nur Syamimi Amir, Abdul Muiz Fathi Md. Abas, Nur Anis Azmi, Zulkifli Zainal Abidin, Amir Akramin Shafie - 2016 International Conference on Computer and Communication Engineering (ICCCE)**

In this paper the chicken farm monitoring system is proposed and developed based on wireless communication unit to transfer data by using the wireless module combined with the sensors that enable to detect temperature humidity light and water level values. This system is focused on the collecting storing and controlling the information of the chicken farm so that the high quality and quantity of the meal production can be produced. This system is developed to solve several problems in the chicken farm which are many human workers is needed to control the farm high cost in maintenance and inaccurate data collected at one point. The proposed methodology really helps in finishing this project within the period given. Based on the research that has been carried out the system that can monitor and control environment condition (temperature humidity and light) has been developed by using the Arduino microcontroller. This system also is able to collect data and operate autonomously.

**[7] ACOUSTIC BASED CHICKEN HEALTH MONITORING IN SMART POULTRY FARMS - Abhinay Bhandekar, Venkanna Udutalapally, Debanjan Das - 2023 IEEE International Symposium on Smart Electronic Systems (iSES)**

Poultry farmings pivotal role in meeting global protein demand emphasizes the need for chicken health assurance. This study introduces real-time acoustic monitoring for poultry farms bridging the gap left by labor-intensive delayed health issue detection that undermines welfare and profits. We devised a system to capture and analyze farm audio data by harnessing audio analytics and machine learning. Targeted microphones strategically captured chicken vocalizations with collected audio undergoing preprocessing for Mel-Frequency Cepstral Coefficients (MFCCs). Utilizing a comprehensively trained Support Vector Machine (SVM) model our solution achieved a remarkable accuracy of 95.66% in real-time detection of abnormal sounds. Rigorous experimentation showcased SVMs proficiency in distinguishing between normal and abnormal further classifying the chickens activities and updating the user every 30 minutes.

**[8] DEVELOPMENT OF THE SMART CHICKEN EGGS INCUBATOR BASED ON INTERNET OF THINGS USING THE OBJECT ORIENTED ANALYSIS AND DESIGN METHOD - Suryo Budi Santoso, Satriyo Adhy, Nurdin Bahtiar, Indra Waspada - 2020 4th International Conference on Informatics and Computational Sciences (ICICoS)**

Smart Chicken Eggs Incubator is one of the applications of the Internet of Things in the field of animal husbandry specifically on the hatching of chicken eggs. Breeders carry out the process of hatching chicken eggs using artificial incubators and in the process of hatching chicken eggs these problems arise in the process of monitoring the environment inside the incubator such as temperature humidity and egg changes which are still done manually. This research explains the development of a prototype Smart Chicken Eggs Incubator System by implementing the Internet of Things which has three subsystems namely embedded systems web-based applications and Telegram bot. Web application software on the Smart Chicken Eggs Incubator System was developed using the Object-Oriented Analysis and Design (OOAD) method. The web apps built can be used to monitor the conditions of the incubator based on sensor data that has been sent during the egg hatching process. While in the Telegram bot real-time conditions of temperature humidity egg transfer and photos of the situation inside the incubator can be monitored as well as message notifications if there are conditions on the incubator that change beyond the specified limits. Prototype testing was carried out for 21 days with the amount of data entered as many as 3402 records with time intervals every 5 minutes and get the optimal time for each entry is 9 minutes to 10 minutes.

**[9] THE IMPLEMENTATION OF MAMDANIS FUZZY MODEL FOR CONTROLLING THE TEMPERATURE OF CHICKEN EGG INCUBATOR - Indri Nurfazri Lestari, Edi Mulyana, Rina Mardi - 2020 6th International Conference on Wireless and Telematics (ICWT)**

Today development of technology is rapidly growing in many fields including in animal husbandry. This technology could be applied in the process of hatching eggs in chicken farms. Since the process of hatching egg was become an important process that correlated with the failure rate in hatching eggs. In order to get an optimal process of hatching eggs temperature must be controlled as closely as an ideal temperature which is around 37-40 Celsius. This study builds the prototype of chicken egg incubator system whose temperature was controlled using a Mamdani logic fuzzy control so its can run optimally. We use DFTII sensor to get the temperature and humidity of incubator as an input of fuzzy logic. Using a fuzzy logic to control the temperature the output of this system is the speed of fan (PWM). Furthermore we use 2 lamps to make the temperature warmer so that the temperature of this incubator could be control in an ideal temperature. We do software testing and overall system performance testing. Software testing is carried out to see whether the implemented fuzzy logic is in accordance with the expected by comparing the results obtained from a system built with manual calculations and simulation calculations. Based on the test results it is found that the fuzzy system has been implemented successfully with 47.36% success. While the results of the overall system performance test show that the system being built has worked well.

**[10] DESIGN OF A CYBER-PHYSICAL SYSTEM USING STEM: CHICKEN EGG INCUBATOR - Pirapat Tangsuknirundorn, Pitikhate Sooraksa - 2019 5th International Conference on Engineering, Applied Sciences and Technology (ICEAST)**

In this paper STEM approach to study design and implement chicken egg incubator as a cyber-physical system is demonstrated. In science viewpoint (S) bringing a life to earth can provide magnificent feeling that inspires students to learn more about life sciences. In views of technology (T) the students learn how to design and use micro-controllers sensors and other components. Specifically the students also develop programming skills in controlling optimal ventilation temperature and relative humidity for incubators in order to hatch chicken eggs. In Engineering and Mathematically (E&M) Kalman filter is adopted to estimate the states from a series of noisy measurements. According to our experiment based on the survey questionnaire the learning outcomes are satisfied.

# SYSTEMANALYSIS

****

**CHAPTER 3**

**SYSTEM ANALYSIS**

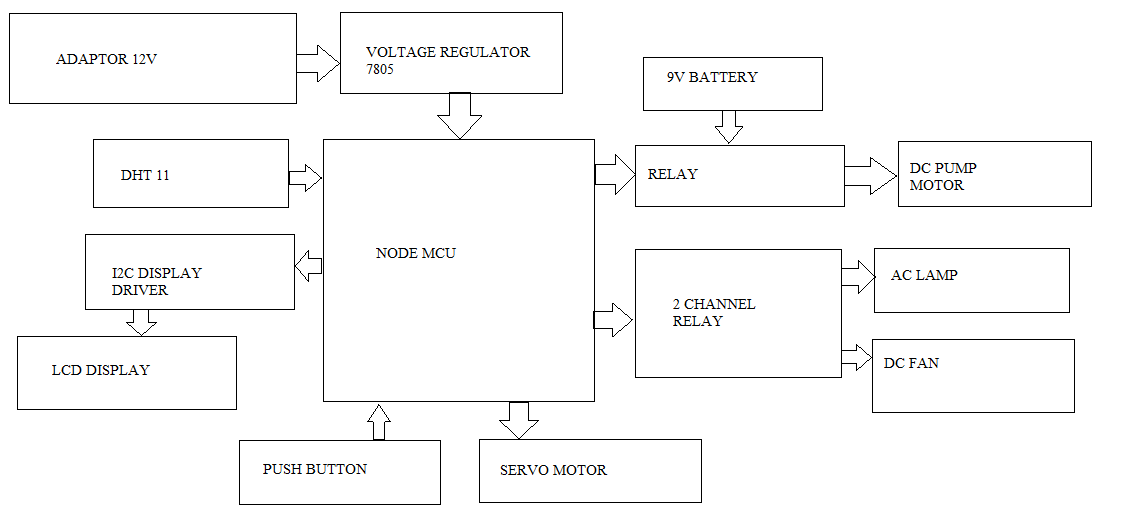
**3.1 EXISTING SYSTEM**

Currently, chicken incubation systems often rely on manual monitoring and control of temperature, humidity, and feeding schedules. Traditional incubators may use basic thermostats and humidity controllers, but they lack automation, requiring frequent human intervention to maintain optimal conditions for egg incubation. Farmers must manually adjust the environment, increasing the risk of inconsistent conditions that may lead to lower hatch rates. Additionally, feeding the chicks after hatching is often done manually, which can be inefficient and time-consuming.

**3.2 DRAWBACKS**

* Manual Monitoring – Requires constant human supervision, increasing labor costs and the risk of human error.
* Inconsistent Temperature and Humidity Control – Fluctuations in environmental conditions can negatively impact hatch rates.
* Lack of Automation – No automated water pumping, humidity regulation, or feeding system, making the process inefficient.
* Higher Energy Consumption – Inefficient temperature and humidity regulation may lead to excessive power usage.
* Time-Consuming Process – Farmers need to check and adjust conditions regularly, reducing efficiency.
* Limited Accuracy – Manual methods may not provide precise environmental control, affecting embryo development.
* Potential Egg Loss – Poor management of incubation conditions can lead to increased embryo mortality.

**3.3 PROPOSED BLOCK DIAGRAM**



**Figure 2: Proposed Model Diagram**

**3.4 PROPOSED SYSTEM EXPLANATION**

The proposed **Chicken Incubator Monitoring System** is an automated solution designed to maintain optimal hatching conditions by regulating temperature, humidity, and feeding schedules. The system integrates a **temperature sensor** that monitors the incubator’s heat levels and activates a **water pump** to cool the environment when temperatures exceed the set threshold. A **humidity sensor** is used to control a **warm lamp**, ensuring that the humidity levels remain ideal for egg incubation. Additionally, a **timer-based feeding mechanism** is implemented to dispense food at scheduled intervals, ensuring proper nourishment for the chicks once hatched. This automated system enhances incubation efficiency, reduces human intervention, and increases the success rate of hatching by maintaining precise environmental conditions.

**3.4 APPLICATIONS**

A Chicken Incubator Monitoring System with a Temperature Sensor for Water Pumping, Humidity Sensor for Warm Lamp, and Timer-Based Food Application is an automated system designed to optimize incubation conditions and chick feeding. Below are 10 key points about this system:

**1. Temperature Sensor for Water Pumping**

* A temperature sensor (e.g., DS18B20 or DHT11) monitors the incubator's temperature.
* If the temperature exceeds the set limit, the system triggers a water pump to spray water and cool the incubator.

**2. Humidity Sensor for Warm Lamp Control**

* A humidity sensor (DHT11 or DHT22) measures the moisture level inside the incubator.
* When humidity drops below a certain level, the system turns on a warm lamp to maintain proper incubation conditions.

**3. Timer-Based Feeding Mechanism**

* A microcontroller-based timer (e.g., Arduino RTC module) automates the food dispensing process.
* Feeds chicks at set intervals to ensure consistent nutrition.

**4. Microcontroller-Based System**

* Uses a microcontroller like Arduino or Raspberry Pi for system control.
* Reads sensor data and triggers actuators based on programmed conditions.

**5. LCD or OLED Display for Monitoring**

* Displays real-time temperature, humidity, and feeding status for easy monitoring.
* Alerts users if conditions go out of the optimal range.

**6. Buzzer and Alert System**

* A buzzer or notification system alerts when conditions exceed safe limits.
* Can be integrated with an IoT module (e.g., Wi-Fi/ GSM) for remote alerts.

**7. Power Backup System**

* Uses a battery backup to ensure continuous operation during power outages.
* Essential for stable incubation and feeding.

**8. Smart Mobile App or Web Integration**

* Optional IoT integration allows remote monitoring through a mobile app or web interface.
* Users can adjust temperature, humidity, and feeding schedules remotely.

**9. Energy Efficiency**

* Uses low-power components to minimize energy consumption.
* Ensures cost-effectiveness for poultry farmers.

**10. Improved Hatch Rate and Chick Growth**

* By maintaining optimal temperature, humidity, and feeding cycles, this system enhances chick survival rates and improves poultry farming efficiency.

# SYSTEM SPECIFICATION

****

**CHAPTER 4**

**SYSTEM SPECIFICATION**

**4.1 Software Requirements**

* Operating system : Windows 7/10.
* MC Programming Language : Embedded ‘C’ Language
* IOT PLATFORM : Things Speak Website

**4.2 SOFTWARE ENVIRONMENT**

**4.2.1 Arduino Software (IDE)**

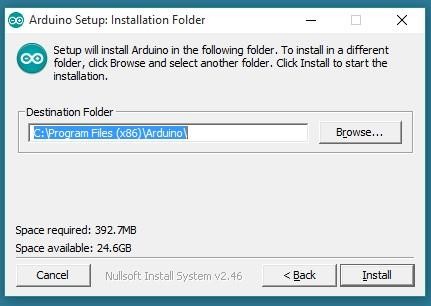
Arduino is an open-source electronics platform based on easy-to-use hardware and software[.](https://www.arduino.cc/en/Main/Products) [Arduino boards](https://www.arduino.cc/en/Main/Products) are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a Motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the [Arduino programming language](https://www.arduino.cc/en/Reference/HomePage) (based on [Wiring),](http://wiring.org.co/) and [the Arduino Software (IDE),](https://www.arduino.cc/en/Main/Software) based on [Processing.](https://processing.org/)

Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals - has gathered around this open-source platform, their contributions have added up to an incredible amount of [accessible knowledge](http://forum.arduino.cc/) that can be of great help to novices and experts alike.

Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. The [software,](https://www.arduino.cc/en/Main/Software) too, is open-source, and it is growing through the contributions of users worldwide.

Arduino also simplifies the process of working with microcontrollers, but it offers some advantage for teachers, students, and interested amateurs over other systems:

* + - **Inexpensive** - Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than $50.
    - **Cross-platform** - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
    - **Simple, clear programming environment** - The Arduino Software (IDE) is easy-touse for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.
    - **Open source and extensible hardware** - The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it. Even relatively inexperienced users can build the breadboard version of the modul[e](https://www.arduino.cc/en/Main/Standalone) in order to understand how it works and save money.

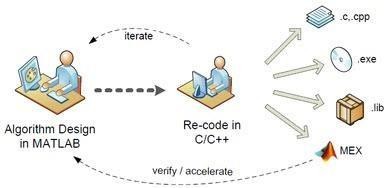
**Figure 3: Software Installing**

The process will extract and install all the required files to execute properly the Arduino Software (IDE).

### 4.2.2 Embedded C

Embedded C is a set of language extensions for the C programming language by the C Standards Committee to address commonality issues that exist between C extensions for different embedded systems. Embedded C programming typically requires nonstandard extensions to the C language in order to support enhanced microprocessor features such as fixed- point arithmetic, multiple distinct memory banks, and basic I/O operations. In 2008, the C Standards Committee extended the C language to address such capabilities by providing a common standard for all implementations to adhere to. It includes a number of features not available in normal C, such as fixed-point arithmetic, named address spaces and basic I/O hardware addressing. Embedded C uses most of the syntax and semantics of standard C, e.g., main() function, variable definition, datatype declaration, conditional statements (if, switch case), loops (while, for), functions, arrays and strings, structures and union, bit operations, macros, etc.

Embedded C Programming is the soul of the processor functioning inside each and every [embedded system](https://www.elprocus.com/ieee-projects-on-embedded-systems/) we come across in our daily life, such as mobile phone, washing machine, and digital camera. Each processor is associated with an embedded software.



**Figure 4: Architecture**

Earlier, many embedded applications were developed using assembly level programming. However, they did not provide portability. This disadvantage was overcome by the advent of various high level languages like C, Pascal, and COBOL. However, it was the C language that got extensive acceptance for embedded systems, and it continues to do so. The C code written is more reliable, scalable, and portable; and in fact, much easier to understand.

C language is a middle-level language as it supports high-level applications and low-level applications. Before going into the details of embedded C programming, we should know about RAM memory organization.

* + - C language is a software designed with different keywords, data types, variables, constants, etc.
    - Embedded C is a generic term given to a programming language written in C, which is associated with a particular hardware architecture.
    - The microcontroller 8051 #include<reg51.h> is used.

The embedded system designers must know about the hardware architecture to write programs. These programs play prominent role in monitoring and controlling external devices.

**4.6.3 Proteus**

Proteus is the best simulation software for various designs with microcontrollers. It is mainly popular because of the availability of almost all microcontrollers in it. So, it is a handy tool to test programs and embedded designs for electronics hobbyists. You can simulate your programming of microcontrollers in Proteus Simulation Software. After simulating your circuit in Proteus Software, you can directly make a PCB design.

**SYSTEM DESIGN**

****

**CHAPTER-5**

**SYSTEM DESIGN**

System design is the phase that bridges the gap between the problem domain and the existing system in a manageable way. It is the process of developing, expressing, documenting, and communicating the realization of the architecture of the system through a complete set of design characteristics described in a form suitable for implementation.

**5.1 SYSTEM ARCHITECTURE**

This system automates the incubation process for chicken eggs by maintaining optimal temperature, humidity, and feeding schedules using sensors and actuators. Below is a breakdown of its architecture:

**1. System Components**

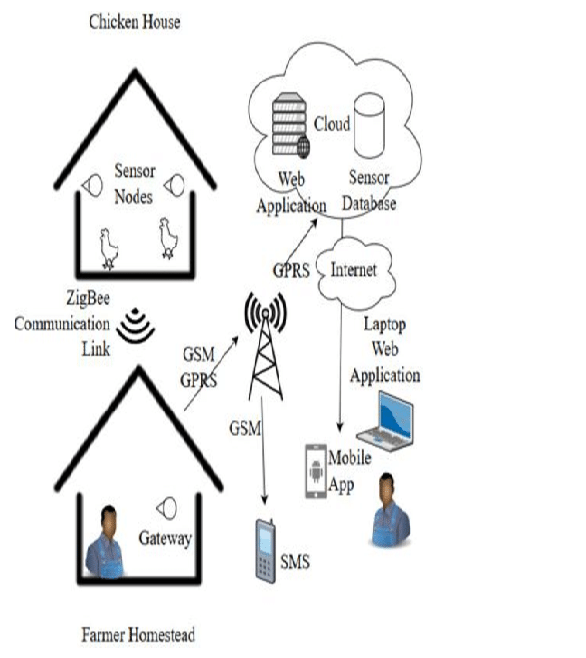
* Microcontroller (e.g., Arduino, ESP32, Raspberry Pi)
  + Acts as the central processing unit to collect sensor data and control actuators.
* Temperature Sensor (e.g., DHT22, DS18B20)
  + Monitors the incubator temperature. If the temperature is too high, a cooling mechanism (water pump) is activated.
  + If too low, the heating system (warm lamp) is adjusted.
* Humidity Sensor (e.g., DHT22, AM2302)
  + Detects humidity levels inside the incubator. If humidity drops below the threshold, the warm lamp is adjusted or a water spraying system is triggered.
* Water Pump System
  + Controls moisture levels by spraying water when humidity is too low.
* Heating System (Warm Lamp / Heating Element)
  + Maintains required incubation temperature.
* Timer-based Feeding Mechanism
  + Dispenses food at scheduled intervals after hatching.
* LCD Display / OLED Screen
  + Shows real-time temperature, humidity, and feeding schedule.
* Buzzer / Alarm System
  + Alerts if the temperature or humidity goes beyond critical levels.
* Wi-Fi / Bluetooth Module (optional: ESP8266/ESP32)
  + Enables remote monitoring and control via a mobile app or web interface.

**2. System Workflow**

1. Monitoring Phase
   * The temperature and humidity sensors continuously monitor conditions.
   * Data is displayed on the LCD screen or sent to a remote monitoring system.
2. Control Phase
   * If the temperature is too high, the water pump activates to cool down the incubator.
   * If the temperature is too low, the warm lamp is activated.
   * If humidity is too low, water spraying is triggered.
3. Feeding System (Post-Hatching)
   * A timer-based food dispensing system ensures regular feeding.
4. Alert & Notifications
   * If parameters go out of range, an alarm or a remote notification is triggered.

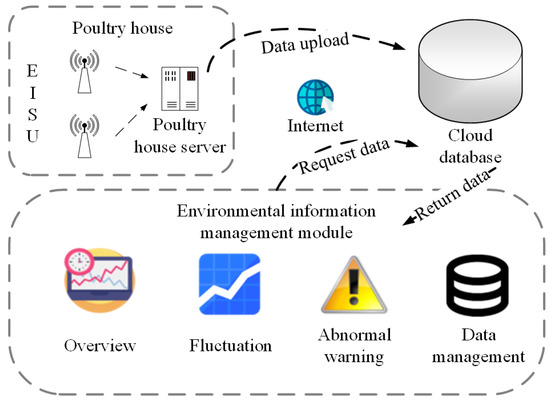
**3. Technologies Used**

* Microcontrollers: Arduino Uno, ESP32, Raspberry Pi
* Programming Languages: C++, Python (for Raspberry Pi), Embedded C
* Sensors: DHT22 (Temp & Humidity), DS18B20 (Temp)
* Communication: Wi-Fi, Bluetooth (optional for mobile monitoring)
* Power Source: 12V DC / Battery Backup

****

**Figure 5: SYSTEM ARCHITECTURE**

**5.1 DATA FLOW DIAGRAM**

****

**Figure 6: DATA FLOW DIAGRAM**

**SYSTEM IMPLEMENTATION**

****

**CHAPTER 6**

**SYSTEM IMPLEMENTATION**

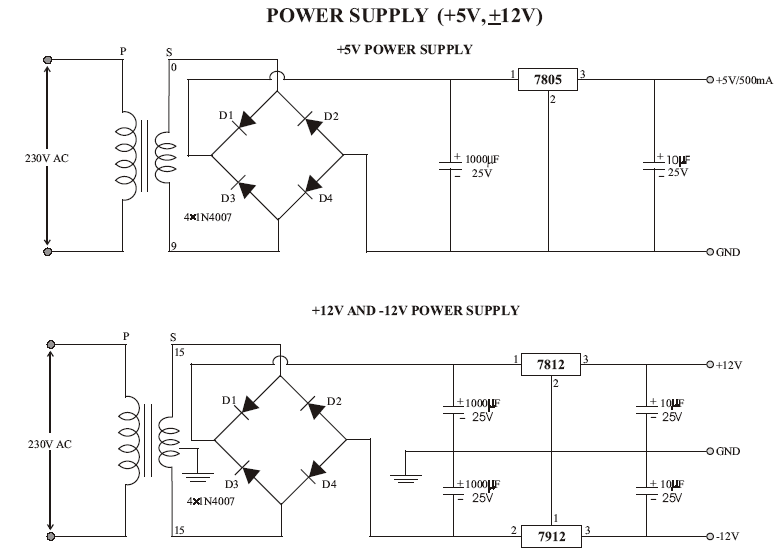
**6.1 Hardware Requirements:**

* LCD
* Transformer
* Voltage Regulator
* NODE MCU
* LCD display
* Relay
* DC Pump Motor
* DTH11 Sensor

**6.2 HARDWARE ENVIRONMENT**

**6.2.1 Power Supply Unit**

The present chapter introduces the operation of power supply circuits built using filters, rectifiers, and then voltage regulators Starting with an AC voltage a steady DC voltage is obtained by rectifying the AC voltage then filtering to a DC level, and finally, regulating to obtain a desired fixed DC voltage. The regulation is usually obtained from an IC voltage regulator unit, which takes a DC voltage and provides a somewhat lower DC voltage, which remains the same even if the input DC voltage varies, or the output load connected to the DC voltage changes.

****

**Figure 7: Diagrams for Power Supply**

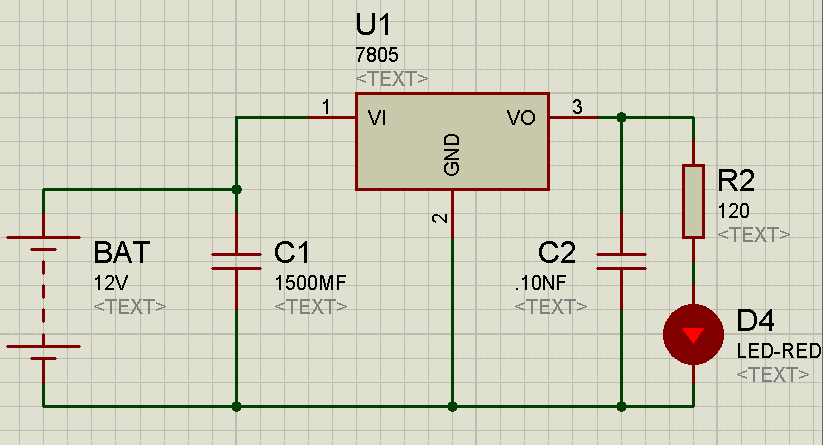
**IC Voltage Regulators**

Voltage regulators comprise a class of widely used ICs. Regulator IC units contain the circuitry for reference source, comparator amplifier, control device, and overload protection all in a single IC. Although the internal construction of the IC is somewhat different from that described for discrete voltage regulator circuits, the external operation is much the same. IC units provide regulation of either a fixed positive voltage, a fixed negative voltage, or an adjustable set voltage. A power supply can be built using a transformer connected to the AC supply line to step the AC voltage to desired amplitude, then rectifying that AC voltage, filtering with a capacitor and RC filter, if desired, and finally regulating the DC voltage using an IC regulator. The regulators can be selected for operation with load currents from hundreds of Millis amperes to tens of amperes, corresponding to power ratings from mill watts to tens of watts.

**Three-Terminal Voltage Regulators**

The basic connection of a three-terminal voltage regulator IC to a load. The fixed voltage regulator has an unregulated DC input voltage, VI, applied to one input terminal, a regulated output DC voltage, Vo, from a second terminal, with the third terminal connected to ground The specifications also list the amount of output voltage change resulting from a change in load current (load regulation) or in input voltage (line regulation).

**Fixed Positive Voltage Regulators**



**Figure 8: Fixed Voltage Regulators**

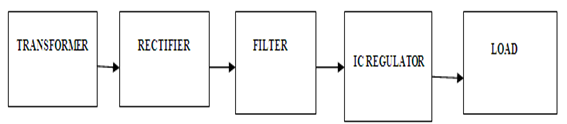
The series 78 regulators provide fixed regulated voltages from 5 to 24 V shows how one such IC, a 7812, is connected to provide voltage regulation with the output from this unit of +12V Dec. An unregulated input voltage VI is filtered by capacitor C1 and connected to the IC’s IN terminal. The IC’s OUT terminal provides a regulated + 12V which is filtered by the capacitor C2 (mostly for any high-frequency noise). The third IC terminal is connected to ground (GND).

While the input voltage may vary over some permissible voltage range, and the output load may vary over some acceptable range, the output voltage remains constant within specified voltage variation limits. These limitations are spelled out in the manufacturer’s specification sheets.

**Block Diagram Of Power Supply**

The AC voltage, typically 220V RMS, is connected to a transformer, which steps that AC voltage down to the level of the desired DC output. A diode rectifier then provides a full-wave rectified voltage that is initially filtered by a simple capacitor filter to produce a DC voltage. This resulting DC voltage usually has some ripple or AC voltage variation.

A regulator circuit removes the ripples and also remains the same DC value even if the input DC voltage varies, or the load connected to the output DC voltage changes.



**FIGURE 9: Block Diagram of Power Supply**

**Transformer**

The potential transformer will step down the power supply voltage (0-230V) to (0-6V) level. Then the secondary of the potential transformer will be connected to the precision rectifier, which is constructed with the help of op–amp. The advantages of using a precision rectifier are it will give a peak voltage output as DC, the rest of the circuits will give only RMS output.

**Bridge Rectifier**

When four diodes are connected as shown in the figure, the circuit is called as a bridge rectifier. The input to the circuit is applied to the diagonally opposite corners of the network, and the output is taken from the remaining two corners.

Let us assume that the transformer is working properly and there is a positive potential, at point A and a negative potential at point B. The positive potential at point A will forward bias D3 and reverse bias D4. The negative potential at point B will forward bias D1 and reverse D2. At this time the D3 and D1 are forward biased and will allow current flow to pass through them; D4 and D2 are reverse biased and will block current flow. The path for current flow is from point B through D1, up through RL, through D3, through the secondary of the transformer back to point B.

This path is indicated by the solid arrows. Waveforms (1) and (2) can be observed across D1 and D3.One-half cycle later the polarity across the secondary of the transformer reverse, forward biasing D2 and D4 and reverse biasing D1 and D3.

The current flow will now be from point A through D4, up through RL, through D2, through the secondary of T1, and back to point A. This path is indicated by the broken arrows. Waveforms (3) and (4) can be observed across D2 and D4. The current flow through RL is always in the same direction. In flowing through RL this current develops a voltage corresponding to that shown waveform (5). Since current flows through the load (RL) during both half cycles of the applied voltage, this bridge rectifier is a full-wave rectifier.

One advantage of a bridge rectifier over a conventional full-wave rectifier is that with a given transformer the bridge rectifier produces a voltage output that is nearly twice that of the conventional full-wave circuit. This may be shown by assigning values to some of the components shown in views A and B. Assume that the same transformer is used in both circuits. The peak voltage developed between points X and y is 1000 volts in both circuits.

Since only one diode can conduct at any instant, the maximum voltage that can be rectified at any instant is 500 volts.. In the bridge rectifier shown in view B, the maximum voltage that can be rectified is the full secondary voltage, which is 1000 volts. Therefore, the peak output voltage across the load resistor is nearly 1000 volts. With both circuits using the same transformer, the bridge rectifier circuit produces a higher output voltage than the conventional full-wave rectifier circuit.

**IC Voltage Regulators**

Voltage regulators comprise a class of widely used ICs. Regulator IC units contain the circuitry for reference source, comparator amplifier, control device, and overload protection all in a single IC. IC units provide regulation of either a fixed positive voltage, a fixed negative voltage, or an adjustable set voltage.

The regulators can be selected for operation with load currents from hundreds of Milli amperes to tens of amperes, corresponding to power ratings from milliwatts to tens of watts. A fixed three-terminal voltage regulator has an unregulated DC input voltage, VI, applied to one input terminal, a regulated DC output voltage, Vo, from a second terminal, with the third terminal connected to ground. The series 78 regulators provide fixed positive regulated voltages from 5 to 24 volts. Similarly, the series 79 regulators provide fixed negative regulated voltages from 5 to 24 volts.

**6.2.2 Node MCU V3**

The best way to develop quickly an IoT application with less Integrated circuits to add is to choose this circuit “NodeMCU”. Today,we will give a detailed Introduction on NodeMCU V3. It is an open-source firmware and development kit that plays a vital role in designing a proper IoT product using a few script lines. The module is mainly based on ESP8266 that is a low-cost Wi-Fi microchip incorporating both a full TCP/IP stack and microcontroller capability. It is introduced by manufacturer Espressif Systems. The ESP8266 NodeMcu is a complex device, which combines some features of the ordinary Arduino board with the possibility of connecting to the internet.

Arduino Modules and Microcontrollers have always been a great choice to incorporate automation into the relevant project. But these modules come with a little drawback as they don’t feature a built-in WiFi capability, subsequently, we need to add external WiFi protocol into these devices to make them compatible with the internet channel.

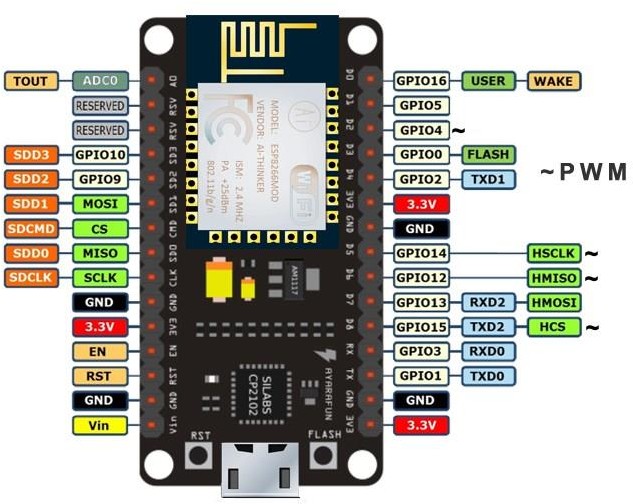
This is the famous NodeMCU which is based on ESP8266 WiFi SoC. This is version 3 and it is based on ESP-12E (An ESP8266 based WiFi module). NodeMCU is also an open-source firmware and development kit that helps you to prototype your IOT product within a few LUA script lines, and of course you can always program it with Arduino IDE. In this article, We will try present useful details related to this WiFi Development Kit, its main features, pinout and everything we need to know about this module and the application domain.

# Introduction Node MCU V3

Node MCU V3 is an open-source firmware and development kit that plays a vital role in designing an IoT product using a few script lines. Multiple GPIO pins on the board allow us to connect the board with other peripherals and are capable of generating PWM, I2C, SPI, and UART serial communications.The interface of the module is mainly divided into two parts including both Firmware and Hardware where former runs on the ESP8266 Wi-Fi SoC and later is based on the ESP-12 module.

**Node MCU V3 Pinout**

NodeMCU V3 comes with a number of GPIO Pins. Following figure shows the Pinout of the board.



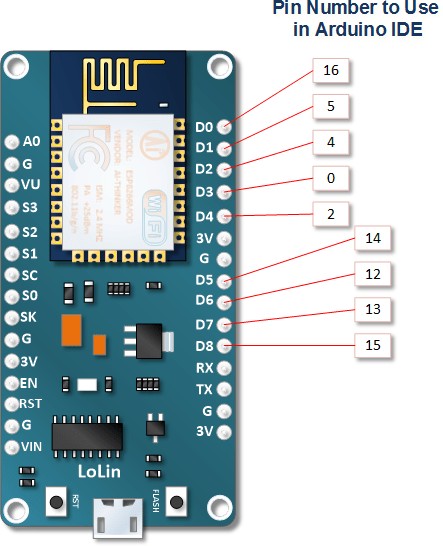
**Figure 10: NodeMCU V3 Pinout**

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

**Features**

1. Open-source
2. Arduino-like hardware
3. Status LED
4. MicroUSB port
5. Reset/Flash buttons
6. Interactive and Programmable
7. ESP8266 with inbuilt wifi
8. USB to UART converter
9. GPIO pins
10. Arduino-like hardware IO

As mentioned above, a cable supporting micro USB port is used to connect the board. As you connect the board with a computer, LED will flash. You may need some drivers to be installed on your computer if it fails to detect the NodeMCU board. You can download the driver from this page.



**Figure 11: Pin Configuration to Use In Arduino IDE**

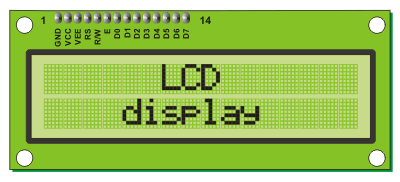
**How to Power Node MCU V3**

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

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

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

**6.2.3 LCD Display**



**Figure 12: Liquid Crystal Display**

A liquid crystal display (LCD) is a flat panel display, electronic visual display, or video display that uses the light modulating properties of liquid crystals. Liquid crystals do not emit light directly. LCDs are available to display arbitrary images (as in a general-purpose computer display) or fixed images which can be displayed or hidden, such as preset words, digits, and 7-segment displays as in a digital clock. They use the same basic technology, except that arbitrary images are made up of a large number of small pixels, while other displays have larger elements.

LCDs are used in a wide range of applications including computer monitors, televisions, instrument, aircraft cockpit displays, and signage. They are common in consumer devices such as video players, gaming devices, clocks, watches, calculators, and telephones, and have replaced cathode ray tube (CRT) displays in most applications. They are available in a wider range of screen sizes than CRT and plasma displays, and since they do not use phosphors, they do not suffer image burn-in. LCDs are, however, susceptible to image persistence.

The LCD screen is more energy efficient and can be disposed of more safely than a CRT. Its low electrical power consumption enables it to be used in battery-powered electronic equipment. It is an electronically modulated optical device made up of any number of segments filled with liquid crystals and arrayed in front of a light source(backlight) or reflector to produce images in color or monochrome. Liquid crystals were first discovered in 1888.

1.Polarizing filter film with a vertical axis to polarize light as it enters.

2.Glass substrate with ITO electrodes. The shapes of these electrodes will determine the shapes that will appear when the LCD is turned ON. Vertical ridges etched on the surface are smooth.

3.Twisted hematic liquid crystal.

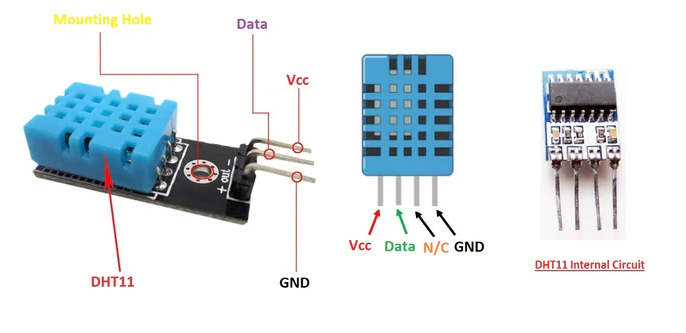
4.Glass substrate with common electrode film (ITO) with horizontal ridges to line up with the horizontal filter.

5.Polarizing filter film with a horizontal axis to block/pass light.

6.Reflective surface to send light back to viewer. (In a backlit LCD, this layer is replaced with a light source.)

**6.2.4 DHT 11 HUMIDITY & TEMPERATURE SENSOR**

DHT11Sensor features a temperature & humidity sensor complex with a calibrated digital signal output. By using the exclusive digital-signal-acquisition technique and temperature & humidity sensing technology, it ensures high reliability and excellent long-term stability. This sensor includes a resistive-type humidity measurement component and an NTC temperature measurement component, and connects to a high- performance 8-bit microcontroller, oﬀering excellent quality, fast response, anti-interference ability and cost-eﬀectiveness.

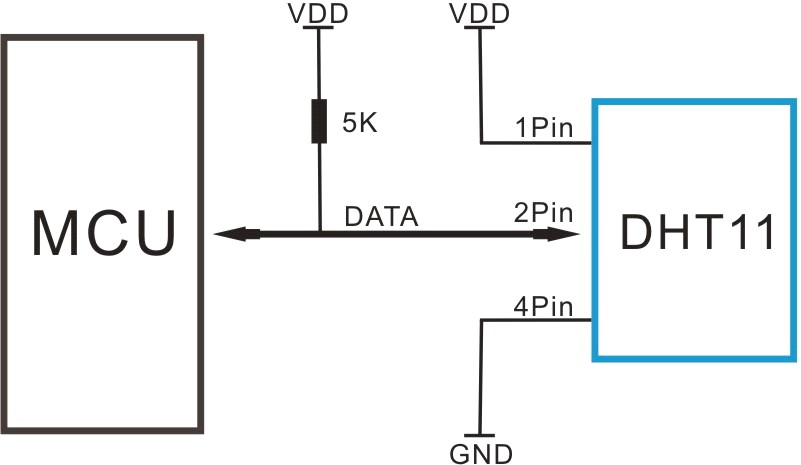


**Figure 13: DTH 11 SENSOR**

Each DHT11 element is strictly calibrated in the laboratory that is extremely accurate on humidity calibration. The calibration coefficients are stored as programmes in the OTP memory, which are used by the sensor’s internal signal detecting process. The single-wire serial interface makes system integration quick and easy. Its small size, low power consumption and up-to-20 meter signal transmission making it the best choice for various applications, including those most demanding ones. The component is 4-pin single row pin package. It is convenient to connect and special packages can be provided according to users’ request.

**Technical Specifications:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item** | **Measurement**  **Range** | **Humidity**  **Accuracy** | **Temperature**  **Accuracy** | **Resolution** | **Package** |
| DHT11 | 20-90%RH  0-50 ℃ | ±5％RH | ±2℃ | 1 | 4 Pin Single Row |

****

**Figure 14: DTH 11 SENSOR PIN OUT**

**Typical Application**

The connecting cable is shorter than 20 metres, a 5K pull-up resistor is recommended; when the connecting cable is longer than 20 metres, choose a appropriate pull-up resistor as needed.

**POWER AND PIN**

DHT11’s power supply is 3-5.5V DC. When power is supplied to the sensor, do not send any instruction to the sensor in within one second in order to pass the unstable status. One capacitor valued 100nF can be added between VDD and GND for power filtering. Communication Process: Serial Interface (Single-Wire Two-Way)

Single-bus data format is used for communication and synchronization between MCU and DHT11 sensor. One communication process is about 4ms.

Data consists of decimal and integral parts. A complete data transmission is 40bit, and the sensor sends higher data bit first.

Data format: 8bit integral RH data + 8bit decimal RH data + 8bit integral T data + 8bit decimal T data + 8bit check sum. If the data transmission is right, the check-sum should be the last 8bit of "8bit integral RH data + 8bit decimal RH data + 8bit integral T data + 8bit decimal T data".

**6.2.5 SERVO MOTORS**



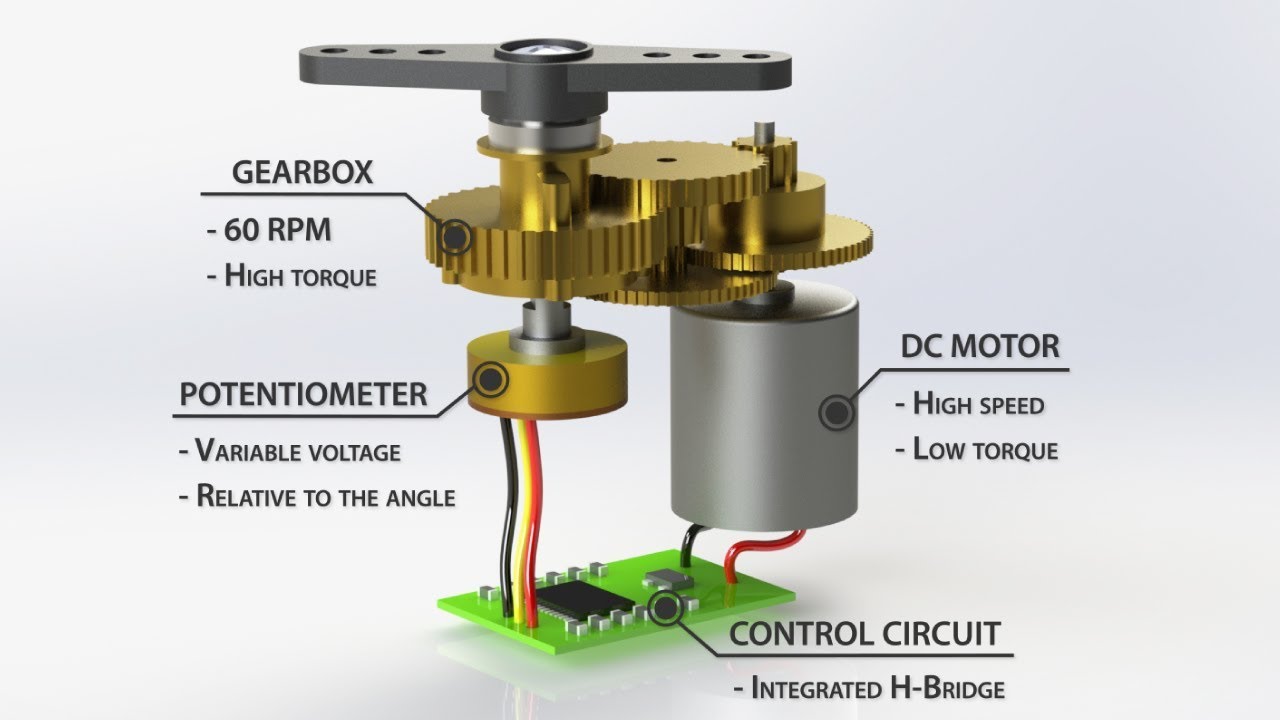
**FIGURE 15: SERVO MOTOR**

How does a servo motor work?

A servo motor is an electromechanical device that produces torque and velocity based on the supplied current and voltage. A servo motor works as part of a closed loop system providing torque and velocity as commanded from a servo controller utilizing a feedback device to close the loop. The feedback device supplies information such as current, velocity, or position to the servo controller, which adjusts the motor action depending on the commanded parameters.

Servo motors are available in an extensive variety of types, shapes and sizes. The term servo was first used in 1859 by Joseph Facort, who implemented a feedback mechanism to assist in steering a ship with steam to control the rudders. A servo motor is part of a servo mechanism consisting of three key elements – a motor, a feedback device, and control electronics. The motor can be AC or DC, brushed or brushless, rotary or linear, and of any size.

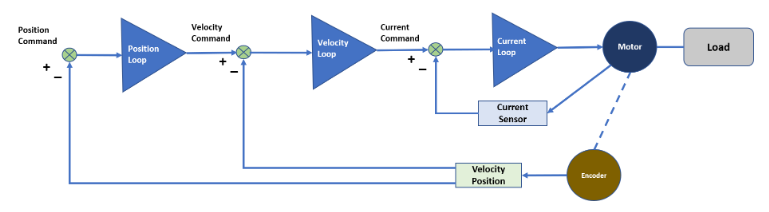
The feedback device can be a potentiometer, Hall-effect device, tachometer, resolver, encoder, linear transducer, or any other sensor as appropriate. Completing the servo system is the control electronics that powers the motor and compares the feedback data and command reference to verify that the servo motor is operating as commanded. There are many types of servo motor applications, from simple DC motors used in hobby applications (such as model airplanes) to sophisticated brushless motors driven by complex motion controllers used for multi-axis machining centers. One example of a common servo mechanism is a vehicle cruise control that consists of an engine (the motor), a speed sensor (feedback), and electronics to compare the vehicle speed with the set speed. If the vehicle slows down, the sensor feeds this data to the electronics which, in turn, increases the gas to the engine to step up the speed to the desired set point – a simple closed loop system.



**FIGURE 16: WORING OF SERVOMOTOR**

 simple industrial servo motor consists of a permanent magnet DC motor with an integral tachometer that provides an output voltage proportional to speed. The drive electronics delivers the necessary voltage and current to the motor based on the voltage fed back from the tachometer. In this example, a commanded speed (represented as a command reference voltage) is set in the driver, then the circuitry in the driver compares the tachometer feedback voltage and determines if the desired speed has been accomplished - known as a closed velocity loop. The velocity loop is monitoring the commanded velocity and tachometer feedback, while the driver adjusts the power to the motor to maintain the desired commanded velocity.

In a more sophisticated servo system, multiple embedded loops are tuned for optimal performance to provide precision motion control. The system consists of current, velocity, and position loops that utilize precision feedback elements. Each loop signals the subsequent loop and monitors the appropriate feedback elements to make real time corrections to match the commanded parameters.

****

**FIGURE 17: CONTROLLER OF SERVO MOTOR**

The base loop is the current or torque loop. Current is proportional to torque in a rotary motor (or force in a linear motor), which provides acceleration or thrust. A current sensor is the device that provides feedback related to the current flowing through the motor. The sensor sends a signal back to the control electronics - typically an analog or digital signal proportional to the motor current. This signal is subtracted from the commanded signal. When the servo motor is at the commanded current, the loop will be satisfied until the current drops below the commanded current. The loop will then increase current until the commanded current is reached, with the cycle continuing at sub second update rates.

**6.2.6 RELAY**

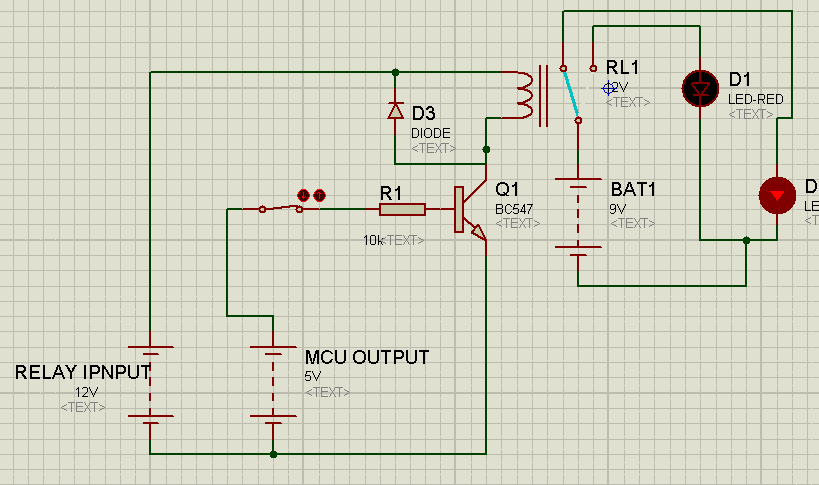


**Figure 18: CIRCUIT DIAGRAM FOR RELAY OFF**

A relay is an electrically operated switch. Current flowing through the coil of the relay creates a magnetic field which attracts a lever and changes the switch contacts. The coil current can be on or off so relays have two switch positions and they are double throw (changeover) switches.

Relays allow one circuit to switch a second circuit which can be completely separate from the first. For example, a low voltage battery circuit can use a relay to switch a 230V AC mains circuit. There is no electrical connection inside the relay between the two circuits; the link is magnetic and mechanical.

The coil of a relay passes a relatively large current, typically 30mA for a 12V relay, but it can be as much as 100mA for relays designed to operate from lower voltages. Most ICs (chips) cannot provide this current and a transistor is usually used to amplify the small IC current to the larger value required for the relay coil. The maximum output current for the popular 555 timer IC is 200mA so these devices can supply relay coils directly without amplification.



**Figure 19: CIRCUIT DIAGRAM FOR RELAY ON**

Relays are usually SPDT or DPDT but they can have many more sets of switch contacts, for example relays with 4 sets of changeover contacts are readily available. For further information about switch contacts and the terms used to describe them please see the page on switches. Most relays are designed for PCB mounting but you can solder wires directly to the pins providing you take care to avoid melting the plastic case of the relay.

**6.2.7 D.C MOTOR**

The electrical motor is an instrument, which converts electrical energy into mechanical energy. According to faraday’s law of Electromagnetic induction, when a current carrying conductor is placed in a magnetic field, it experiences a mechanical force whose direction is given by Fleming’s left hand rule.

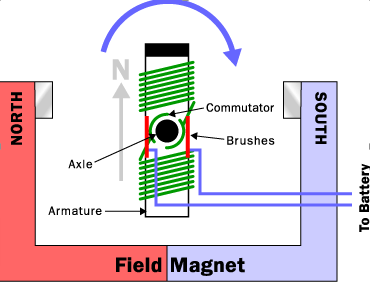


**Figure 20: DC Motor**

Constructional a dc generator and a dc motor are identical. The same dc machine can be used as a generator or as a motor. When a generator is in operation, it is driven mechanically and develops a voltage. The voltage is capable of sending current through the load resistance. While motor action a torque is developed.

* DC Motor capacity : 12V
* Un loading : 130rpm
* Loading : 90rpm

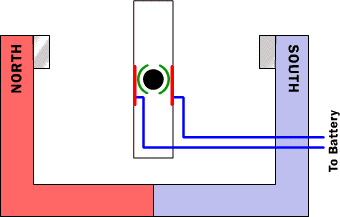
The motor run’s according to the principle of Fleming’s left hand rule. When a current carrying conductor is placed in a magnetic field is produced to move the conductor away from the magnetic field. The conductor carrying current to North and South poles is being removed

**Figure 21:** **Construction of DC Motors**

If the current in the conductor is reversed, the strengthening of the flux lines occurs below the conductor, and the conductor will be pushed upwards An electric motor is all about magnets and magnetism: a motor uses magnets to create motion. If you have ever played with magnets you know about the fundamental law of all magnets: Opposites attract and likes repel. So if you have 2 bar magnets with their ends marked north and south, then the North end of one magnet will attract the South end of the other. On the other hand, the North end of one magnet will repel the North end of the other (and similarly south will repel south). Inside an electric motor these attracting and repelling forces create rotational motion. To understand how an electric motor works, the key is to understand how the electromagnet works. An electromagnet is the basis of an electric motor. You can understand how things work in the motor by imagining the following scenario.

Say that you created a simple electromagnet by wrapping 100 loops of wire around a nail and connecting it to a battery. The nail would become a magnet and have a North and South Pole while the battery is connected. Now say that you take your nail electromagnet, run an axle through the middle of it, and you suspended it in the middle of a horseshoe magnet as shown in the figure below. If you were to attach a battery to the electromagnet so that the North end of the nail appeared as shown, the basic law of magnetism tells you what would happen The North end of the electromagnet would be repelled from the north end of the horseshoe magnet and attracted to the south end of the horseshoe magnet

The armature takes the place of the nail in an electric motor. The armature is an electromagnet made by coiling thin wire around two or more poles of a metal core. The armature has an axle, and the commentator is attached to the axle. In the diagram above you can see three different views of the same armature: front, side and end-on. In the end-on view the winding is eliminated to make the commentator more obvious. You can see that the commentator is simply a pair of plates attached to the axle. These plates provide the two connections for the coil of the electromagnet.

 The "flipping the electric field" part of an electric motor is accomplished by two parts: the commentator and the brushes. The diagram at the right shows how the commentator and brushes work together to let current flow to the electromagnet, and also to flip the direction that the electrons are flowing at just the right moment. The contacts of the commentator are attached to the axle of the electromagnet, so they spin with the magnet. The brushes are just two pieces of springy metal or carbon that make contact with the contacts of the commentator.

**Figure 22 :The armature winding**

In this figure, the armature winding has been left out so that it is easier to see the commentator in action. The key thing to notice is that as the armature passes through the horizontal position, the poles of the electromagnet flip. Because of the flip, the North Pole of the electromagnet is always above the axle so it can repel the field magnet's North Pole and attract the field magnet's South Pole. If you ever take apart an electric motor you will find that it contains the same pieces described above: two small permanent magnets, a commentator, two brushes and an electromagnet made by winding wire around a piece of metal. Almost always, however, the rotor will have three poles rather than the two poles as shown in this article. There are two good reasons for a motor to have three poles:

**MODULE DESCRIPTION**

****

**CHAPTER - 7**

**MODULE DESCRIPTION**

* 1. **MODULES**
* Temperature Sensor Module (for Water Pumping)
* Humidity Sensor Module (for Warm Lamp Control)
* Timer Module
* Control and Display Module
* Alarm and Notification System
* Power Supply Module

The **Chicken Incubator Monitoring System** is designed to maintain optimal hatching conditions by automatically regulating temperature, humidity, and timing functions. The system consists of the following key modules:

**1. Temperature Sensor Module (for Water Pumping)**

* Utilizes a temperature sensor (e.g., DHT11, DHT22, or DS18B20) to monitor the incubator's internal temperature.
* If the temperature exceeds a predefined threshold, the system activates a water pump to increase humidity and regulate the temperature.
* Ensures stable conditions for embryo development and prevents overheating.

**2. Humidity Sensor Module (for Warm Lamp Control)**

* Uses a humidity sensor (e.g., DHT11 or DHT22) to measure the incubator's humidity levels.
* If humidity drops below the required level, a warm lamp is turned on to maintain proper moisture levels.
* Helps in preventing excessive dryness, which can affect egg viability.

**3. Timer Module**

* Incorporates a real-time clock (RTC) or microcontroller-based timer to manage egg-turning intervals.
* Automatically triggers a motor or actuator to turn the eggs at scheduled times, ensuring even heat distribution.
* Can also be used for timing the incubation period, alerting users when hatching is near.

**4. Control and Display Module**

* A microcontroller unit (e.g., Arduino, ESP32, or Raspberry Pi) processes sensor data and controls system components.
* LCD or OLED display shows real-time temperature, humidity, and system status.
* Buttons or a touchscreen interface allow users to set incubation parameters.

**5. Alarm and Notification System**

* If temperature or humidity deviates from the ideal range, an alarm or buzzer notifies the user.
* Can include an optional IoT module (WiFi/Bluetooth) to send alerts to a smartphone or computer.

**6. Power Supply Module**

* The system runs on a regulated power supply (e.g., 12V/5V DC adapter or battery backup) to ensure uninterrupted operation.
* Features an emergency backup to prevent power failures from affecting incubation.

**SYSTEM TESTING**

****

**CHAPTER - 8**

**SYSTEM TESTING**

The process of testing an integrated hardware and software system is an implementation process that helps ensure that the system works correctly and consistently before the start of a live operation. Testing is crucial to the success of the system. System Testing is a logical assumption that the goal would be accomplished if all parts of the device are correct.

**8.1 TYPES OF TESTS**

* + - * Unit Testing
      * Integration Testing
      * Validation Testing

**8.1.1 Unit Testing**

The unit testing phase ensures that each individual component of the Chicken Incubator Monitoring System functions as expected. The temperature sensor is tested to verify accurate readings and proper activation of the water pumping mechanism when temperatures exceed or fall below the predefined range. Similarly, the humidity sensor is tested to confirm its ability to detect changes in moisture levels and trigger the warm lamp accordingly. The timer functionality undergoes rigorous testing to ensure it correctly schedules and maintains incubation cycles. Each module is tested independently to validate data accuracy, responsiveness, and seamless integration with the control system. Any detected errors or inconsistencies are identified and resolved before moving to the system testing phase, ensuring a reliable and efficient incubation process.

**8.1.2 Integration Testing**

Integration testing ensures that all components of the Chicken Incubator Monitoring System work together seamlessly. The system comprises a temperature sensor for water pumping, a humidity sensor for the warm lamp, and a timer to regulate incubation cycles. During testing, each sensor is verified for accurate readings and proper response to environmental changes. The temperature sensor triggers water pumping when the heat level exceeds the preset threshold, preventing overheating. Similarly, the humidity sensor activates the warm lamp to maintain optimal moisture levels. The timer is tested to confirm precise incubation scheduling. Data logging and real-time monitoring functionalities are also examined to ensure reliability. Any discrepancies are addressed by debugging and reconfiguring sensor thresholds. Successful integration guarantees a stable and automated incubation process, promoting ideal hatching conditions.

**8.1.3 Validation Testing**

To The Chicken Incubator Monitoring System was subjected to a series of validation tests to ensure its accuracy, reliability, and efficiency in maintaining optimal incubation conditions. The temperature sensor was tested to verify its responsiveness in triggering the water pumping mechanism when the temperature exceeded the preset threshold. Similarly, the humidity sensor was evaluated for its precision in activating the warm lamp when humidity levels dropped below the required range. The system’s timer was also assessed to confirm proper scheduling and automated operation. Each component was tested under various environmental conditions to simulate real-world incubation scenarios. The results demonstrated that the system effectively maintained stable temperature and humidity levels, ensuring a suitable environment for egg incubation. Minor adjustments were made to optimize sensor sensitivity and response times. The successful validation confirms the system's functionality and reliability in automated incubation control.

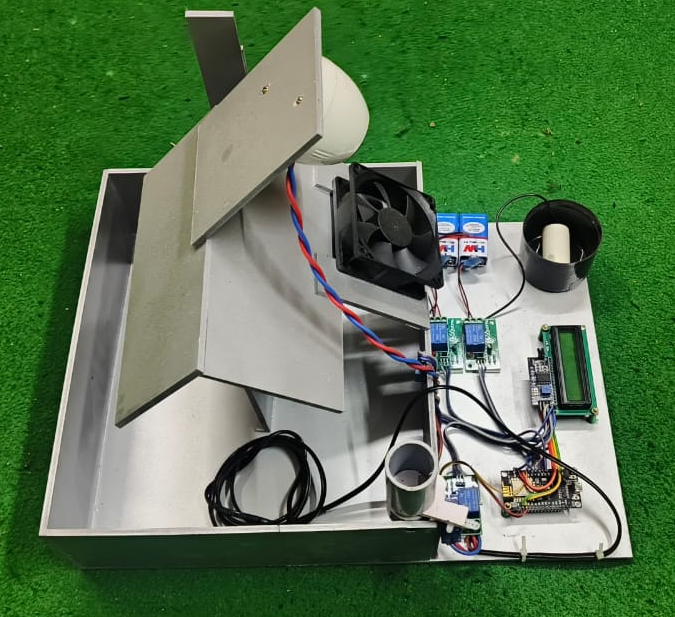
**IMPLEMENTATION AND RESULTS**

****

**CHAPTER 9**

**IMPLEMENTATION AND RESULTS**

**9.1 HARDWARE PHOTOGRAPHY**

****

The experimental setup for the Chicken Incubator Monitoring System consists of an enclosed incubation chamber equipped with a temperature sensor, a humidity sensor, a warm lamp, a water pumping system, and a microcontroller for automated monitoring and control. The system is designed to maintain optimal conditions for egg incubation by regulating temperature and humidity levels. A DHT22 sensor is used to measure both temperature and humidity, while a water pump is activated when the temperature exceeds a set threshold to cool the environment. Similarly, a heat lamp is controlled based on humidity readings to ensure proper moisture levels are maintained for egg development.

The microcontroller, typically an Arduino or Raspberry Pi, serves as the central processing unit, continuously collecting real-time data from the sensors and triggering the appropriate actuators. A relay module is used to switch the heat lamp on or off based on humidity levels, while a solenoid valve or pump controls water flow to regulate temperature. The system also integrates a timer to manage incubation duration, ensuring each stage of development is accurately tracked. The data can be displayed on an LCD screen or transmitted wirelessly to a monitoring device for remote supervision.

To validate the system's efficiency, the incubator is tested under controlled conditions, simulating real incubation processes. The temperature and humidity readings are logged at regular intervals to assess the system's responsiveness. Various test scenarios, such as sudden temperature rises or drops, are introduced to evaluate the accuracy and reliability of the automated controls. The collected data is then analyzed to ensure the system maintains optimal incubation conditions, ultimately improving hatch rates and efficiency compared to traditional incubation methods.

**CONCLUSION AND FUTURE SCOPE**

****

**CHAPTER 10**

**CONCLUSION AND FUTURE SCOPE**

**10.1 CONCLUSION**

In conclusion, the Chicken Incubator Monitoring System successfully integrates a temperature sensor for water pumping, a humidity sensor for the warm lamp, and a timer to create an efficient and automated incubation process. By maintaining optimal temperature and humidity levels, the system enhances hatchability rates and reduces manual intervention, ensuring a stable environment for developing embryos. This smart system not only improves efficiency but also offers a cost-effective and reliable solution for poultry farmers. Future improvements could include remote monitoring and control features to further enhance usability and precision.

**10.** **2 FUTURE SCOPE**

The Chicken Incubator Monitoring System with a temperature sensor for water pumping, a humidity sensor for a warm lamp, and a timer-based food supply has significant potential for future advancements. Future improvements could include integrating IoT (Internet of Things) capabilities, enabling remote monitoring and control via a mobile application. Machine learning algorithms can be incorporated to optimize temperature and humidity settings dynamically, improving hatch rates and efficiency. Additionally, solar power integration can make the system more sustainable and suitable for remote areas with limited electricity. Enhancing the system with real-time alerts and automated troubleshooting features can further increase reliability. Future versions may also support scalability for larger poultry farms, making it adaptable to commercial hatcheries.

**REFERENCES**

[1] CHICKEN FARM MONITORING SYSTEM - Nur Syamimi Amir, Abdul Muiz Fathi Md. Abas, Nur Anis Azmi, Zulkifli Zainal Abidin, Amir Akramin Shafie - 2016 International Conference on Computer and Communication Engineering (ICCCE) - 2016

[2] AN IOT MONITORING ASSISTANT FOR CHICKEN LAYER FARMS - Ron Daniel M. Nicolas, Wei S. Zhou, Shota C. Kitamura, Mary Jane C. Samonte - 2019 International Conference on Information and Communication Technology Convergence (ICTC) - 2019

[3] DESIGN OF A CYBER-PHYSICAL SYSTEM USING STEM: CHICKEN EGG INCUBATOR - Pirapat Tangsuknirundorn, Pitikhate Sooraksa - 2019 5th International Conference on Engineering, Applied Sciences and Technology (ICEAST) - 2019

[4] DEVELOPMENT OF THE SMART CHICKEN EGGS INCUBATOR BASED ON INTERNET OF THINGS USING THE OBJECT ORIENTED ANALYSIS AND DESIGN METHOD - Suryo Budi Santoso, Satriyo Adhy, Nurdin Bahtiar, Indra Waspada - 2020 4th International Conference on Informatics and Computational Sciences (ICICoS) - 2020

[5] THE IMPLEMENTATION OF MAMDANIS FUZZY MODEL FOR CONTROLLING THE TEMPERATURE OF CHICKEN EGG INCUBATOR - Indri Nurfazri Lestari, Edi Mulyana, Rina Mardi - 2020 6th International Conference on Wireless and Telematics (ICWT) - 2020

[6] ACOUSTIC BASED CHICKEN HEALTH MONITORING IN SMART POULTRY FARMS - Abhinay Bhandekar, Venkanna Udutalapally, Debanjan Das - 2023 IEEE International Symposium on Smart Electronic Systems (iSES) - 2023

[7] ANIMAL BEHAVIOR FOR CHICKEN IDENTIFICATION AND MONITORING THE HEALTH CONDITION USING COMPUTER VISION: A SYSTEMATIC REVIEW - Md Roman Bhuiyan, Philipp Wree - IEEE Access ( Volume: 11) - 2023

[8] SMART CHICKEN FARMING; CASE STUDY IN NORTH RIFT KENYA - Allan K. Koech, Fidel Makatia, Valery Chebet - 2023 IEEE Conference on AgriFood Electronics (CAFE) - 2023

[9] RESEARCH ON INTELLIGENT CONTROL TECHNOLOGY OF CHICKEN HOUSE ENVIRONMENT IN LARGE-SCALE CHICKEN FARMS - Qingle Quan, Thelma D. Palaoag, Hanqing Sun - 2024 7th International Conference on Communication Engineering and Technology (ICCET) - 2024

[10] RESEARCH AND DESIGN OF INTELLIGENT INSPECTION ROBOT FOR LARGE-SCALE CHICKEN FARMS - Qingle Quan, Thelma D. Palaoag, Hanqing Sun - 2024 5th International Conference on Machine Learning and Human-Computer Interaction (MLHMI) - 2024

**APPENDIX**

**SOURCE CODE**

#include <ESP8266WiFi.h>

#include "secrets.h"

#include "ThingSpeak.h" // always include thingspeak header file after other header files and custom macros

char ssid[] = SECRET\_SSID; // your network SSID (name)

char pass[] = SECRET\_PASS; // your network password

int keyIndex = 0; // your network key Index number (needed only for WEP)

WiFiClient client;

unsigned long myChannelNumber = SECRET\_CH\_ID;

const char \* myWriteAPIKey = SECRET\_WRITE\_APIKEY;

String myStatus = "";

#include <LCD\_I2C.h>

LCD\_I2C lcd(0x27, 16, 2);

#include <dht.h>

#define dht\_apin 16

dht DHT;

void Humidity();

int h, ht;

void Update\_Data();

int b=0;

// Timer variables

unsigned long lastTime = 0;

unsigned long timerDelay = 30000;

int Relay1 = 14; //DC Fan

int Relay2 = 12; //Pump Motor

int Relay3 = 13; //AC Light

#include <Servo.h> // include Servo library

Servo Feed; // horizontal servo

int pos;

int pos\_move;

int delay\_move ;

int T\_Sec, T\_Min, Timer = 0;

void setup()

{

pinMode(Relay1,OUTPUT);

digitalWrite(Relay1,LOW);

pinMode(Relay2,OUTPUT);

digitalWrite(Relay2,LOW);

pinMode(Relay3,OUTPUT);

digitalWrite(Relay3,LOW);

lcd.begin();

lcd.backlight();

lcd.setCursor(0,0);

lcd.print(" CHICKEN FARMING MONITORING "); // Start Printing

lcd.setCursor(0,1);

lcd.print(" SYSTEM "); // Start Printing

for (int positionCounter = 0; positionCounter < 85; positionCounter++) {

lcd.scrollDisplayRight();

delay(120);

}

delay(500);

lcd.clear();

Serial.begin(115200);

WiFi.mode(WIFI\_STA);

ThingSpeak.begin(client);

Wi\_Fi\_Inti();

Feed.attach(0);

Feed.write(0);

for (pos = 0; pos <= 180; pos += 1) {

Feed.write(pos);

delay(15);

}

for (pos = 180; pos >= 0; pos -= 1) {

Feed.write(pos);

delay(15);

}

Feed.write(180);

}

void loop()

{

b=b+1;

Humidity();

if(ht > 32)

{

digitalWrite(Relay2,HIGH);

}else{

digitalWrite(Relay2,LOW);

}

if(h < 26)

{

digitalWrite(Relay1,HIGH);

digitalWrite(Relay3,HIGH);

}else{

digitalWrite(Relay1,LOW);

digitalWrite(Relay3,LOW);

}

T\_Sec = T\_Sec +1;

if(T\_Sec >= 60)

{

T\_Min = T\_Min + 1;

T\_Sec = 0;

}

if(T\_Min == 5)

{

Feed.write(160);

T\_Min = 0;

T\_Sec = 0;

}else{

lcd.setCursor(0,0);

lcd.print("Feeder = ");

lcd.print(T\_Min);

lcd.print(":");

lcd.print(T\_Sec);

Feed.write(180);

}

delay(500);

if(b>=15){

ThingSpeak.setField(1, ht);

ThingSpeak.setField(2, h);

ThingSpeak.setStatus(myStatus);

ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);

Serial.println("\nUploaded");

lcd.setCursor(0,1);

lcd.print("Uploaded..., ");

delay(500);

b=0;

}else{

lcd.setCursor(0,1);

lcd.print(" ");

}

}

void Humidity()

{

DHT.read11(dht\_apin);

lcd.setCursor(0,1);

h = DHT.humidity/2;

ht = DHT.temperature;

lcd.print("H="); lcd.print(h); lcd.print("% ");

lcd.print("T="); lcd.print(DHT.temperature ); lcd.print("c ");

delay(250);

}

void Wi\_Fi\_Inti()

{

if(WiFi.status() != WL\_CONNECTED){

Serial.print("Attempting to connect to SSID: ");

Serial.println(SECRET\_SSID);

Serial.print("PASS = ");

Serial.println(SECRET\_PASS);

lcd.setCursor(0,0);

lcd.print("SSID = ");

lcd.print(SECRET\_SSID);

lcd.print(" ");

lcd.setCursor(0,1);

lcd.print("PASS = ");

lcd.print(SECRET\_PASS);

lcd.print(" ");

delay(5000);

lcd.clear();

if (WiFi.status() != WL\_CONNECTED){

WiFi.begin(ssid, pass);

Serial.print("\*");

lcd.setCursor(0,0);

lcd.print(" Searching..., ");

lcd.print(" ");

delay(5000);

}

Serial.println("\nConnected.");

lcd.setCursor(0,0);

lcd.print(" Connected..., ");

delay(2000);

lcd.clear();

}

}