**SMART CHICKEN FARMING: AMMONIA and FEEDS MONITORING with REMOTE TEMPERATURE and HUMIDITY REGULATOR**

A Design Project

Presented to the Faculty of the School of Undergraduate Studies

AMA UNIVERSITY

Maximina St. Villa Arca Subd., Project 8 Quezon City

In partial fulfillment

Of the requirements for the Degree of

Bachelor of Science in Electronic Engineering

(BSECE)

By

GARCIA, ERICK

August 2021

# Acknowledgment

First and first, we want to express our gratitude to God for providing us with the wisdom, knowledge, patience, skills, strength, and ability required to complete this research project effectively.

We'd like to thank **Engr. Jeffrey T. Leonen** for his unwavering support, professional advice, understanding, and encouragement throughout our studies and research. Our thesis study would have been a difficult and daunting endeavor if it hadn't been for his great patience and timely insight and guidance.

To our family for their unwavering love, care, and unwavering moral support as a strong foundation for us to be led and succeed forward in life. This study would not be effective or forward-thinking if they were not present.

# ABSTRACT

The environmental conditions monitoring and control’s ability is crucial and demands a good level of research in fields ranging from the change in climatic conditions in agriculture and zoology. According to world’s agricultural produce survey [[1]](#_REFERENCES), chicken is among the most favorite produce, since it is a nutrient rich food providing high protein, low fat, low cholesterol, and low energy than other kinds of poultries.

From last few years worldwide, there has been an increased level of awareness regarding the safety of food products like chickens and there has been a high demand for good quality and quantity chicken food. This research focuses on the integration of wireless sensors and mobile network with a well-known sensors’ integration platform using remote sensing. System initiates the action automatically to control the environmental parameters such as temperature, humidity, ammonia gas (NH₃) then, the control will be based on the set threshold value when there is a sudden change in climate. The proposed solution will decrease the environmental diseases affecting chicken and increase the productivity and eliminate a lot of manpower who can make some human errors. The method can care also about the data analysis.

### TABLE OF CONTENTS

[Acknowledgment i](#_Toc80131388)

[ABSTRACT ii](#_Toc80131389)

[CHAPTER I 1](#_Toc80131393)

[THE PROBLEM AND ITS BACKGROUND 1](#_Toc80131394)

[1.1 Introduction 2](#_Toc80131395)

[1.2. Background of the Study 3](#_Toc80131396)

[1.3. Objectives of the Study 7](#_Toc80131397)

[1.3.1. General Objective 7](#_Toc80131398)

[1.3.2. Specific Objectives 7](#_Toc80131399)

[1.4. Significance of the Study 8](#_Toc80131400)

[1.5. Scope and Delimitation 9](#_Toc80131401)

[1.6 Conceptual Framework 11](#_Toc80131402)

[1.7 Operational Definition of Terms 13](#_Toc80131403)

[Chapter II 16](#_Toc80131404)

[Review of Related Literatures and Studies 16](#_Toc80131405)

[2.1 Monitoring environmental parameters: humidity and temperature using Arduino based microcontroller and sensors 16](#_Toc80131406)

[2.2 A GSM-Based Remote Temperature and Humidity Monitoring System for Granary 17](#_Toc80131407)

[2.3 An Automatic and Realtime Control of Ammonia Concentration in Catfish Pond Water Based on MQ137 Sensor 17](#_Toc80131408)

[2.4. Arduino UNO 18](#_Toc80131409)

[2.5. Sensors 19](#_Toc80131410)

[Chapter III. 20](#_Toc80131411)

[Methodology 20](#_Toc80131412)

[3.1 General Method Used 20](#_Toc80131413)

[3.1.1 Waterfall Methodology 21](#_Toc80131414)

[3.1.1.1 Requirement Phase 21](#_Toc80131415)

[3.1.1.2 Design Phase 22](#_Toc80131416)

[3.1.1.3 Implementation Phase 23](#_Toc80131417)

[3.1.1.4 Verification Phase 24](#_Toc80131418)

[3.1.1.5 Maintenance Phase 24](#_Toc80131419)

[3.2 Procedure 25](#_Toc80131420)

[3.2.1 Requirements Specification 25](#_Toc80131421)

[3.2.2 Requirements Specification 28](#_Toc80131422)

[3.2.2.1 General system architecture 28](#_Toc80131423)

[3.3 Design 29](#_Toc80131424)

[DHT22 Temperature and Humidity Sensor 30](#_Toc80131425)

[MQ137 Ammonia Gas (NH₃) sensor 33](#_Toc80131426)

[HX711 Load Cell Amplifier 38](#_Toc80131427)

[SIM800 GPRS GSM Shield 40](#_Toc80131428)

[3.4 Verification and Testing 44](#_Toc80131429)

[3.4.1 DHT22 Temperature and Humidity Sensor 45](#_Toc80131430)

[3.4.2 MQ137 Ammonia Gas (NH₃) Sensor 46](#_Toc80131431)

[3.4.3 HX711 Load Cell Amplifier 47](#_Toc80131432)

[3.4.4 SIM800 GPRS GSM Shield 49](#_Toc80131433)

[3.5 Unit Testing 50](#_Toc80131434)

[Chapter IV 53](#_Toc80131435)

[Results and Discussions 53](#_Toc80131436)

[4.1 The Developed System 54](#_Toc80131437)

[4.1.1 Arduino IDE code for DHT22 54](#_Toc80131438)

[4.1.2 Arduino IDE code for calibrating MQ137 56](#_Toc80131439)

[4.1.3 Arduino IDE code for calibrating HX711 56](#_Toc80131440)

[4.1.4 Arduino IDE code for sending a sample message using SIM800 GPRS GSM Shield 59](#_Toc80131441)

[4.1.4 Arduino IDE code for the Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator 60](#_Toc80131442)

[4.2 Verification and Testing Result 65](#_Toc80131443)

[4.2.1 Unit Testing 65](#_Toc80131444)

[4.2.2 Acceptance Test 69](#_Toc80131445)

[4.3 Data Result of Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator 71](#_Toc80131446)

[4.4 Data Result of the system when a user sends the prompt message 72](#_Toc80131447)

[4.5 Data Result of a Humid weather 73](#_Toc80131448)

[4.6 Data Result of a Dry weather 74](#_Toc80131449)

[Chapter V 75](#_Toc80131450)

[Summary of Findings, Conclusions and Recommendations 75](#_Toc80131451)

[5.1 Summary of Findings 75](#_Toc80131452)

[5.2 Conclusions 76](#_Toc80131453)

[5.3 Recommendations 77](#_Toc80131454)

### LIST OF FIGURES

[Fig. 1. 1 The DMAIC Process Correspondence to Problem Identification 7](#_Toc80175353)

[Fig. 1. 2 Conceptual Framework of study 11](#_Toc80175354)

[Fig. 2. 1 Arduino UNO 18](#_Toc80175355)

[Fig. 3. 1 Waterfall model development process 21](#_Toc80175356)

[Fig. 3. 2 General System Architecture 28](#_Toc80175357)

[Fig. 3. 3 Level 0 block Diagram of the Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator 29](#_Toc80175358)

[Fig. 3. 4 DHT22 sensor module & Fig. 3.4. 1 Circuit diagram for DHT22 31](#_Toc80175359)

[Fig. 3.4. 2 DHT22 Single-bus communication protocol 32](#_Toc80175360)

[Fig. 3.5. 1 MQ137 Ammonia Sensor Module & Fig. 3.5. 2 MQ137 sensor circuit diagram 33](#_Toc80175361)

[Fig. 3.5. 3 Typical dependence of the MQ137 on Temperature and RH 34](#_Toc80175362)

[Fig. 3.5. 4 MQ137 Sensitivity Characteristics 35](#_Toc80175363)

[Fig. 3.5. 5 Arduino IDE results for the RS, RO at fresh air and VRL after 24 hours preheat time 37](#_Toc80175364)

[Fig. 3.5. 6 Sensor Circuit Simplified & Fig. 3.5. 7 RS and RL combined 37](#_Toc80175365)

[Fig. 3.6. 1 HX711 Breakout board & Fig. 3.6. 2 HX711 Circuit Diagram 40](#_Toc80175366)

[Fig. 3.7. 1 SIM800 GPRS GSM Shield & Fig. 3.7. 2 SIM800 GPRS GSM Shield Circuit Diagram 41](#_Toc80175367)

[Fig. 3.8. 1 Level 1 diagram of the Smart chicken farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator 42](#_Toc80175368)

[Fig. 3.9. 1 System Flowchart 45](#_Toc80175369)

[Fig. 3.10. 1 DHT22 Sensor readings using Arduino IDE 46](#_Toc80175370)

[Fig. 3.10. 2 DHT22 Sensor system architecture 46](#_Toc80175371)

[Fig. 3.11. 1 NH₃ reading using Arduino IDE 47](#_Toc80175372)

[Fig. 3.11. 2 MQ137 Sensor system Architecture 47](#_Toc80175373)

[Fig. 3.12. 1 Test weight using a digital weighing scale 48](#_Toc80175374)

[Fig. 3.12. 2 HX711 acquiring the Calibration Factor 48](#_Toc80175375)

[Fig. 3.12. 3 Actual weighing using HX711 with a 10KG Load cell balancer 49](#_Toc80175376)

[Fig. 3.12. 4 HX711 System Architecture 49](#_Toc80175377)

[Fig. 3.13. 1 Arduino IDE Serial Monitor 50](#_Toc80175378)

[Fig. 3.13. 2 Sample message screenshot from a mobile phone 50](#_Toc80175379)

[Fig. 3.13. 3 SIM800 GPRS GSM Shield system architecture 51](#_Toc80175380)

[Fig. 4. 1 The developed system (back view) 55](#_Toc80175381)

[Fig. 4. 2 The developed system (close-up view) 55](#_Toc80175382)

[Fig. 4.2. 1 The developed system (front view) 56](#_Toc80175383)

[Fig. 4. 3 Data result on Arduino IDE 73](#_Toc80175384)

[Fig. 4. 4 System replied message when a prompt message is received 74](#_Toc80175385)

[Fig. 4.5. 1 High humidity threshold is reached & Fig. 4.5. 2 Fan Heater air reached 42.9C 75](#_Toc80175386)

[Fig. 4.6. 1 The temperature reaches > 32C & Fig. 4.6. 2The mist maker automatically turns on ………………………………………………………………………………76](#_Toc80175387)

[Fig. 4.7. 1 High NH₃ gas detected & Fig. 4.7. 2 Ammonia gas releasing to the sensor 77](#_Toc80175388)

[Fig. 4.8. 1 Chicken feeder status message & Fig. 4.8. 2 A test object is placed on the feeder 78](#_Toc80175389)

### LIST OF TABLES

[**Table I** Requirements specification of the Smart chicken farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator 25](#_Toc80177348)

[**Table II** Technical Specification of the Smart chicken farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator 27](#_Toc80177349)

[**Table III** Level 0 functional requirements of the DHT22 Temperature and Humidity sensor 30](#_Toc80177350)

[**Table IV** Level 0 functional requirements of the MQ137 Ammonia Gas (NH₃) sensor 33](#_Toc80177351)

[**Table V** Level 0 functional requirements of the HX711 Load cell amplifier 39](#_Toc80177352)

[**Table VI** Level 0 functional requirements of the SIM800 GPRS GSM Shield 40](#_Toc80177353)

[**Table VII** Level 1 Functional Requirements of the Smart chicken farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator 43](#_Toc80177354)

[**Table VIII** Table for Unit Testing 52](#_Toc80177355)

[**Table IX** Result of Unit Test of Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator 67](#_Toc80177356)

[**Table X** Result of Integration Test of Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator 69](#_Toc80177357)

[**Table XI** Result of Acceptance Test of Engineering Requirement 1 71](#_Toc80177358)

# CHAPTER I

# THE PROBLEM AND ITS BACKGROUND

In chicken farming, one of the challenges for breeders is to maintain the sanitation in the coop. When chicken excrete, the nitrogen in poultry manure is broken down by bacteria and produces a pungent-smelling gas called ammonia (Chartier, 2020). Monitoring ammonia levels is a year-round problem, and it can be intensified depending on the environmental conditions such as high temperatures and moisture (Howell, 2019).

The toxicity level of ammonia increases as temperature increases (Environmental Protection Agency, n.d.), which is the same with humidity. When humidity rises above the recommended level, it will wet the litter that also contributes to high ammonia levels (Linden, 2015). It can be noted that the high levels of ammonia gas in the broiler farm affect the resistance of chicken to diseases that can lead to high mortality rates. In addition, the high concentration of ammonia in the air can also reduce appetite and slowdown the growth rate, lowering egg production and decrease in meat production of the slaughterhouses that results to decreased return-of-investment. In conclusion, it is very important for breeders to maintain a balanced temperature and humidity in chicken coop to reduce the presence of ammonia in the air.

## 1.1 Introduction

Automation is increasingly important in modern agriculture, reducing dependence on labor and liberating farmers from constant work, increasing management scale and efficiency, fulfilling the precision and consistency of product quality control, enabling enforceable traceability as part of food safety efforts – all of which can help achieve agricultural sustainability [[2]](#_REFERENCES). Poultry farming is the process of raising domesticated birds such as chickens, ducks, turkeys and geese for the purpose of farming meat or eggs for food. This research focused on modern technologies for poultry farming to control all environmental parameters like temperature, humidity, ammonia gas (NH₃) that influences on the growth of the chickens. If the environmental condition is not up to the mark, then there may be harmful for digestive, respiratory and behavioral change in the chickens. If chickens may get suitable atmosphere and proper food then it may grow rapidly and health of chickens will be good so their weights will increase. Climate plays quite important role in the growth of the chicken.

A hatched chick cannot maintain a proper body temperature without help. In the first three weeks of its life, a chick should not be exposed to cool temperatures. Exposing the young bird to cool temperatures (20° C) for a day or two on the farm can cause the bird to die from heart problems later. Hence, heated premises are a must for brooding. Temperature should be reduced by 3° C per week, until the room temperature of 20°C is reached. After six weeks of age, desirable temperature range in 18 to 21° C. The most widely used source of heat, is a heat lamp for small flocks of birds [[3]](#_REFERENCES).

In poultry farm production, humidity and air pollution are critical; during the summer season, birds experience discomfort due to high humidity combined with high temperatures. High CO₂ levels result in lethargic chicks with reduced weight gains, while high ammonia gas (NH₃) level result in poor feed conversion, reduced weight gain and increased susceptibility to disease. Smart poultry farm can be designed in way that the humidity and air pollution can be controlled by ventilations, cooling and heater.

Within this paper, the parameters, like ammonia gas (NH₃), humidity, temperature and a weighing scale for the feeder are monitored and controlled using microcontroller. The transmitted data should be received by receiver through the GSM module and then can turn on or off the attached humidifier and a fan heater inside the chicken coop if the threshold values are surpassed.

.

## 1.2. Background of the Study

Ammonia is released particularly in a hot environment. The factors of temperature and relative humidity that greatly influence NH₃ volatilization are reflection of weather conditions in each season of the year (Mihina et al., 2010).

Ammonia (NH₃) is the primary basic gas in the atmosphere. Elevated concentrations of NH₃ in poultry barns reduce feed intake and impede bird growth rate, decrease egg production, damage the respiratory tract, increase susceptibility to Newcastle disease virus, increase the incidence of air sacculitis and keratoconjunctivitis and increase the prevalence of Mycoplasma gallisepticum (Kristensen, Wathes, 2000). Egg quality may also be adversely affected by high levels of atmospheric ammonia as measured by reduced albumen height, elevated albumen pH and albumen condensation (Xin et al., 2011).

Gaseous NH₃ is the predominant pollutant in poultry systems. Higher concentrations adversely affect bird performance, welfare and human health (Costa et al., 2012). It is highly reactive and deposits readily to vegetation and other surfaces close to its source. Intensive animal husbandry units, such as poultry farms, are major sources of ammonia. The reports of Wheeler et al. (2006), Li et al. (2008) and Li and Xin (2010) showed that 80.9% of total ammonia emissions in USA were from animal husbandry activities, of which 26.7% were from poultry.

Ammonia is released particularly in a hot environment. The factors of temperature and relative humidity that greatly influence NH₃ volatilization are reflection of weather conditions in each season of the year (Mihina et al., 2010). A direct relationship between NH₃ emissions and indoor temperature was also observed. Although indoor temperature was identified as main variable influencing NH₃ emissions, other variables, such as ventilation rate and bird activity, may also be influencing those emissions (Walker et al., 2014). Ammonia emission rates averaged 19.7 and 18.1 mg.h–1 per bird in the summer and winter, respectively, and increased with indoor temperature (r2 = 0.51 in summer; r2 = 0.42 in winter). Emissions are mainly produced at the end of the summer cycle, whereas in winter, their production begins earlier in the cycle (Calvet et al., 2011).

Ammonia emission is affected also by litter material, litter humidity and ventilation flow (Patterson, Adrizal, 2005; Mihina et al., 2012a). NH₃ emissions varied depending on the litter temperature (Mihina et al., 2012b). The use of new litter material in each cycle determines 92 the dynamics of NH₃ production. Likewise, distribution of ammonia within the poultry house depends on the ventilation system, particularly the air circulation as well as poorly maintained water drinkers, bird stocking density and flocking behavior (Kristensen, Wathes, 2000).

Most gaseous pollutants originate from the breakdown of fecal matter and the concentrations depend on the ventilation efficiency and rate, as well as the stocking density and movements of the animals. The litter type, management, humidity and temperature affect the gas concentration and emission from broiler fattening (Redding, 2013). Also, commercial egg production facilities involve variety of housing systems and manure handling practices, which can produce different magnitudes of environmental footprint. However, research information concerning the environmental pollution for various production systems and the system’s ability to maintain the microenvironment that is conducive to poultry welfare and health, conservation of natural resources and production efficiency is not very clear.

In animal housing there are several factors that affect the production and release of harmful gaseous compounds. These are primarily the number and live weight of housed animals, floor surface covered with their excrements, manure storage time in housing area, performance of ventilation system, air temperature, year season, air movement above the litter surface or not bedded barn floor, air penetration through the litter, litter temperature, moisture, pH, the C:N ratio and feed composition (Knowlton, 2000; Wheeler et al., 2003; Coufal, 2006; Mihina et al., 2012a).

Maintaining the correct temperature is critical in chick brooding, especially during the first two weeks of the chick's life. Early in life, the chick is poorly equipped to regulate its metabolic processes to adequately control its body temperature. As a result, the young chick is dependent on environmental temperature to maintain optimal body temperature. Chilling or overheating during this crucial period can result in poor growth, poor feed conversion and increased susceptibility to disease. Proper brooding practices must maintain the chick's body temperature so that it does not have to use energy to lose heat by panting or generate heat through metabolism. (B. Fairchild, 2012)

The monitoring of environmental variables such as temperature and humidity has a long history of development and the variables have shown significant impact in the productivity of plant growth, the quality of food industry and the efficiency of many temperatures and humidity-sensitive equipment (Vleeschouver, et al., 2010). The monitoring of temperature and humidity of laboratories, storages, halls, school and hospitals is important with respect to health and hygiene. The reliable measurement and monitoring are crucial in this competitive era of technology.

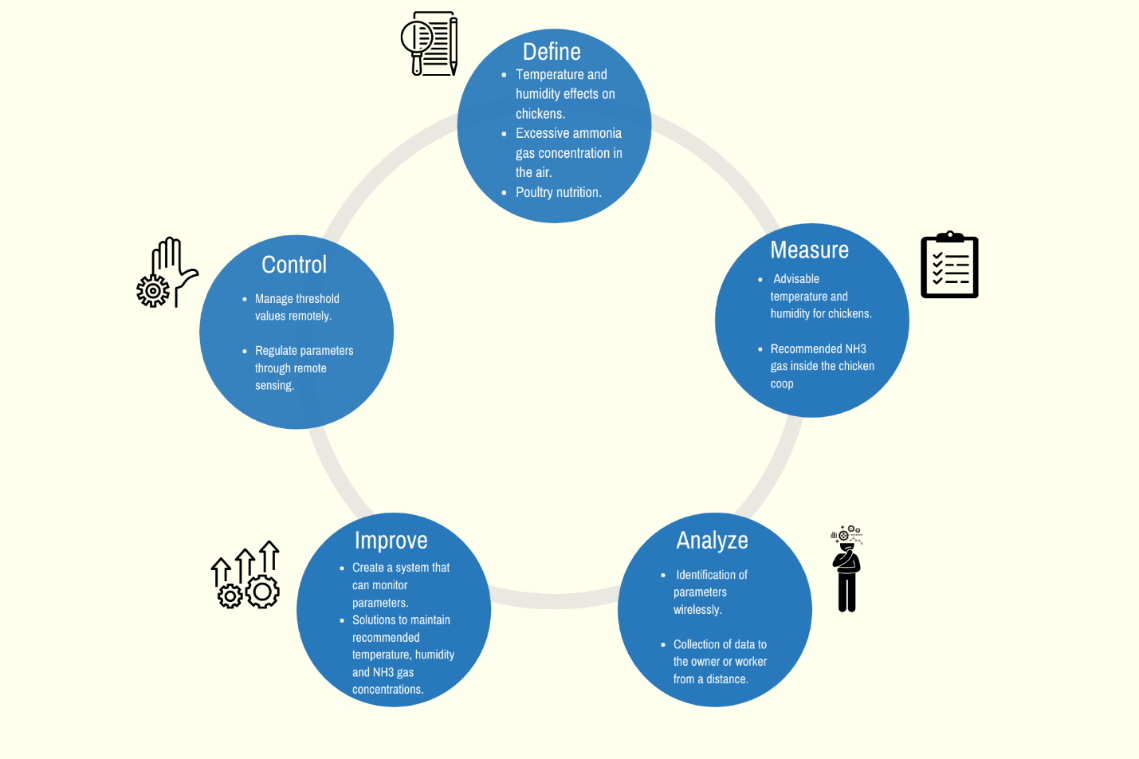


Fig. 1. 1 The DMAIC Process Correspondence to Problem Identification

## 1.3. Objectives of the Study

### 1.3.1. General Objective

Build a device that will simplify the work of monitoring the ammonia levels and reducing its presence in the air by maintaining a balanced temperature and humidity in chicken coop for the breeders with the use of sensors and Arduino UNO– a microcontroller-based electronic system. The device was maximized by adding a chicken feeder weight sensing that will prompt the worker of its present status.

### 1.3.2. Specific Objectives

In line with this, the project aims to achieve the following specific objectives:

* To develop a monitoring device that will notify and send sensor data values to the worker or owner via SMS.
* To control the parameter of the temperature and humidity automatically.
* To activate the humidifier for the temperature and the fan heater for the humidity if the threshold values are exceeded in terms of using relay switches.
* To measure and reduce excessive ammonia gas (NH₃) present inside the chicken coop.
* To ensure that the chicken coop is in the ideal condition by measuring the recommended levels of temperature and humidity for tropical countries like the Philippines.

## 1.4. Significance of the Study

Below is a list of significant contributions of the study.

* In global context, the study will contribute to the modernization of agriculture by creating a device that will help the chicken breeders simplify their work. This project aims to to improve agricultural productivity through modern technology transform agriculture and then increase the income of the owner. This also make differences between urban and rural areas and will make agriculture into a modern agriculture which has the world advanced level.
* In economic context, the study will put considerable effort into methodologies to make food production more efficient and will encourage more farmers to do Smart Farming. This project also provides Food Security - increases the ability of countries to develop their agricultural markets and economies. Thus, access to quality, nutritious food is fundamental to human existence is granted. It also secures access to food can produce wide ranging positive impacts, including economic growth and job creation.
* In environmental context, the study will reduce high levels of noxious gases, especially ammonia gas (NH₃), causing poor habitat conditions for the birds and the workers inside the chicken coop. The project also presents technical options to mitigate environmental impacts, such as improvements to farm management, animal waste management and nutrition management, along with options to reduce the impacts of intensive feed production.
* In societal context, the study will attain social sustainability [[4]](#_REFERENCES) that can be defined as the ability of a community to develop processes and structures that not only meet the needs of its current constituents, but also support the ability of future generations to maintain a healthy community. In the present study, the concept of social sustainability applied included topics such as social equity, livelihood, health equity, community development, labor rights, community resilience and animal welfare.

## 1.5. Scope and Delimitation

The study covers the design of a Smart Chicken Farming composed of DHT22 sensor for monitoring current temperature and humidity inside the chicken coop. The study covers the remote regulation of excess temperature by using of an ultrasonic sonic mist maker attached to the humidifier and unwanted humidity by using of a fan heater. Its purpose is in desire that that the chicken coop is in the ideal condition by measuring the recommended levels of temperature and humidity for tropical countries like the Philippines. By using relay switches, the fan heater and the humidifier inside the chicken coop can be automated remotely if the value of the environmental parameters are reached beyond maximum recommended values. The study also covers the identification of high level of ammonia gas (NH₃) present in the air by using the MQ137 sensor module. It is intended to ensure that the air quality inside the chicken coop must be regulated if the threshold value is surpassed. The study also uses a weight sensing module HX711 for the chicken feeder so that the worker or owner is aware if the feeder is needed to be refilled. In terms of sending of all the sensor data values gathered, the proponents used SIM800 GSM Module by which the receiver will be sending a prompt message to it in order that the data will be sent to the receiver.

Due to the design constraints, this study does not cover call functions for the receiver in order to acquire data sensor values accumulated. This study also does not cover remote areas with no cellular mobile services which the device cannot received a prompt message to send all of the data sensor values to the receiver. The proposed design cannot totally eliminate the ammonia gas present inside the chicken coop in which traces of the said gas is existing in the environment. This study also delimits the replenishing of the chicken feeds where it must be done manually. The design is limited only to the prototype model which is just a small-scale system with small covered area, but the idea of the larger scale for mass production is still reliable with respect to the smaller scale applying the same basic concept and computation.

## 1.6 Conceptual Framework

In order to successfully achieve the desired outcome of this study, certain procedures, requirements and ideas were carefully discussed to conceptualize the project’s design and development. After a long deliberation and brainstorming, one thought was agreed upon in which resulted into one concept.

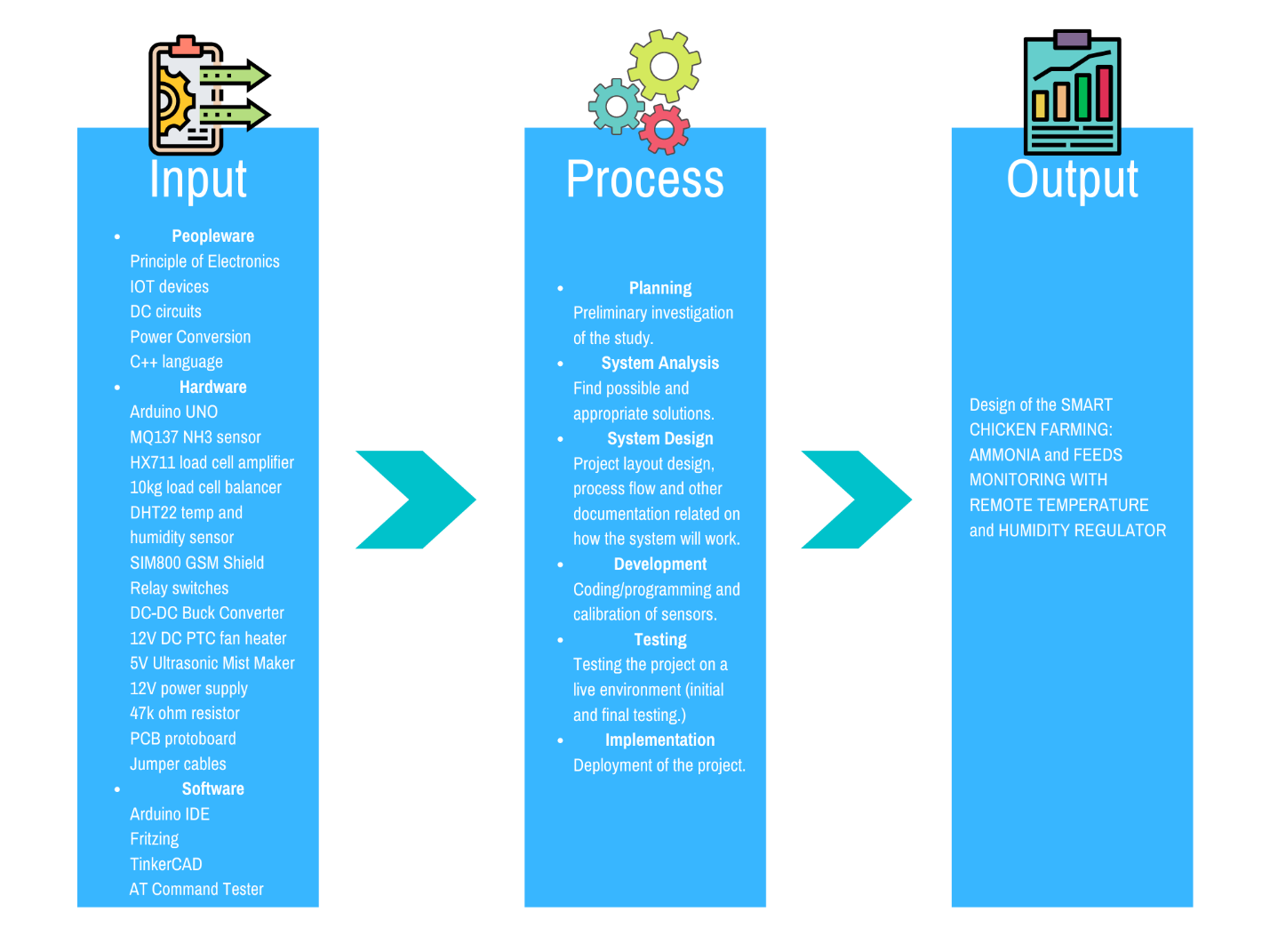


Fig. 1. 2 Conceptual Framework of study

In order to successfully achieve the desired outcome of this study, certain procedures, requirements and ideas were carefully discussed to conceptualize the project’s design and development. After a long deliberation and brainstorming, one thought was agreed upon in which resulted into one concept.

Fig. 1.2 shows the conceptual framework of the study. It covers the three major parts, namely: Input, Process and Output Phases. This serves as a guide that is used by the proponents to make conceptual distinctions and organize ideas. This does not put a limitation to the study and changes can be made. The proponents can have the freedom in adjusting the conceptual framework as it is deem fit during the duration of the study.

The Input phase includes: Peopleware, Hardware, and Software. Under the Peopleware are Principle of Electronics, IOT devices, DC circuits, Power Conversion and C++ language. Hardware consists of Arduino UNO, MQ137, NH₃sensor, HX711 load cell amplifier, 10kg load cell balancer, DHT22 temperature and humidity sensor, SIM800 GSM Shield, Relay switches, DC-DC Buck Converter, 12V DC PTC fan heater, 5V Ultrasonic Mist Maker, 12V power supply, 47k ohm resistor, PCB protoboard and Jumper cables. Lastly, the Software includes Arduino IDE, Fritzing, TinkerCAD, AT Command Tester and PuTTY.

The process phase covers the approach for Planning which making preliminary investigation of the study followed by System Analysis which the proponents find possible and appropriate solutions. Thereafter, System Design will be followed, it consists of project layout design, process flow and other documentation related on how the system will work and then Development of the project will be the next it involves the coding/programming and calibration of sensors for the system. Testing of the project in a live environment must be done in order to obtain data sensor values. Lastly, Implementation or deployment of the project must be made.

The output phase provides the design of the Smart chicken farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator.

## 1.7 Operational Definition of Terms

* *Remote monitoring system* - Global System for Mobile communication (GSM) module is designed for wireless monitoring through Short Messaging Service (SMS). This project is able to receive serial data from monitoring devices such as cellular phones and transmit the data as text SMS to a receiver. It provides two-way communication for data transmission and status query. The project hardware consists of GSM module, SIM circuit and Atmega328P microcontroller. Microcontroller provides control for sending, receiving and AT command processing to GSM module. It processes all incoming SMS, transmits alert/notification SMS when the data reach or exceed threshold value, and transmits SMS data at every fixed interval according to configuration. Integration of this project with monitoring device will create mobile and wireless monitoring system with prompt emergency alert.
* *Computer programming* - Computer programming is the process of designing and building an executable computer program to accomplish a specific computing result or to perform a specific task. Programming involves tasks such as: analysis, generating algorithms, profiling algorithms' accuracy and resource consumption, and the implementation of algorithms in a chosen programming language.
* *GSM communication* Global System for Mobile Communications. It is the digital mobile network that is widely used by mobile phone users in the world. GSM uses a variation of time division multiple access and is the most widely used technology among Time Division Multiple Access (TDMA), Global System for Mobile Communications (GSM) and Code Division Multiple Access (CDMA). GSM phones allow you to simply switch SIM cards and use another GSM device. GSM provides standard features like phone call encryption, data networking, caller ID, call forwarding, call waiting, SMS, and conferencing
* *Sensor* - a device which detects or measures a physical property and records, indicates, or otherwise responds to it.
* *Relay Switches*– are electric switches that use electromagnetism to convert small electrical stimuli into larger currents. These conversions occur when electrical inputs activate electromagnets to either form or break existing circuits. By leveraging weak inputs to power stronger currents, relays effectively act as either a switch or an amplifier for the electric circuit, depending on the desired application.*PCB and Breadboards* - A printed circuit board (PCB) mechanically supports and electrically connects electrical or electronic components using conductive tracks, pads and other features etched from one or more sheet layers of copper laminated onto and/or between sheet layers of a non-conductive substrate.
* *Buck converter* -- A power converter that steps-down the voltage. This will be applied in order to conventionally reduce the voltage to a stable 5V for the ultrasonic mist maker instead of having a variable voltage greater than 5V.
* *Ultrasonic mist maker/humidifier* – Also referred to as ultrasonic atomizer, a piezo atomizer disc/transducer (ceramic humidifier) works by transposing high-frequency sound waves into mechanical energy that is transferred into a liquid, creating standing waves. As the liquid exits the atomizing surface of the disc, it is broken into a fine mist of uniform micron-sized droplets.
* *Positive Temperature Coefficient (PTC) heater* - a self-regulating heaters that run open-loop without any external diagnostic controls and utilizes Positive Temperature Coefficient materials i.e., materials that exhibit a positive resistance change in response to the increase in temperature. As the temperature increases, the electrical resistance of the material also increases, thus limiting the current flow.

# Chapter II

# Review of Related Literatures and Studies

This chapter is designed to identify and enumerate several researches related to the present study. The purpose of this review is to set the current research project within a conceptual and theoretical context. This includes several citations from magazines, journals, newspapers, case studies, theses, research papers and other reliable literature sources.

## 2.1 Monitoring environmental parameters: humidity and temperature using Arduino based microcontroller and sensors

Nagendra Dangi developed the purpose of the system that was to build an Arduino based embedded device for building monitoring (monitoring environmental variables temperature and humidity) and to study the characteristics of its performance. To study its performance characteristics, Arduino was tested in three different temperature and humidity conditions. The different environmental conditions were created by using the device in NTP (normal room temperature/humidity), outdoor temperature/humidity and the insulated wooden box. The developed system is useful 3 in monitoring two variables- temperature and humidity- in a building, laboratory and greenhouse.

## 

## 2.2 A GSM-Based Remote Temperature and Humidity Monitoring System for Granary

Xiao Xi Zheng, Lian Rong Li and Shuai Fei Shao proposed a remote temperature and humidity monitoring system is designed based on the GSM technology and MSP430. With the digital sensor DSB1820 and SHT11, the temperature and humidity of the granary are detected, and these parameters can be adjusted with the controlling system to adapt various working conditions. Through the GSM system, the detected data could be sent to various monitoring devices, such as cellphones and laptops. These data can be used for data display, inquiry, controlling and storage at the remote terminals. The experimental results show that the system is convenient and concise, which meets the remote monitoring demand for the modern granary.

## 2.3 An Automatic and Realtime Control of Ammonia Concentration in Catfish Pond Water Based on MQ137 Sensor

I H Pulungan were explained that ammonia concentration control tool in catfish farming has been successfully tested and applied to catfish culture media in real-time. The test results show that the tool has been able to control the ammonia level of catfish culture water automatically and in real-time. The success of the study is very beneficial for catfish farmers that provide technology to control the ammonia level of catfish water media so that it can improve the quality and quantity of catfish yields.

## 2.4. Arduino UNO

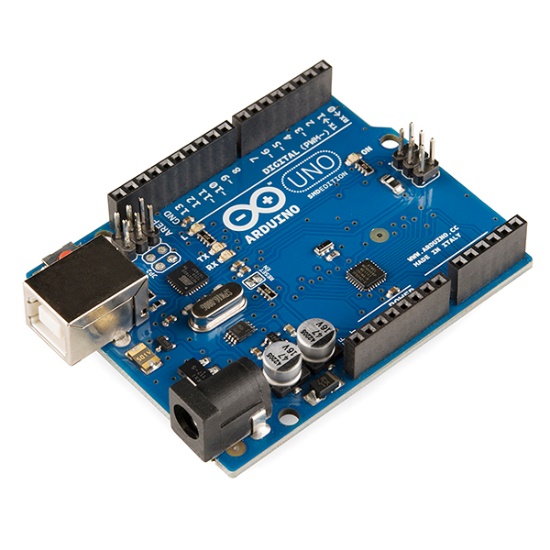
**

Fig. 2. 1 Arduino UNO

The most essential part of designing a monitoring system; the Arduino is an open-source computer hardware and software company, project, and user community that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices and interactive objects that can sense and control objects in the physical world. In other words, it can also be defined as a single board microcontroller for building digital objects and interactive devices. Arduino is designed to sense the environment and/or surrounding by receiving input signal through sensors and communicates with its surrounding through actuators. An actuator could be a simple LED (light emitting diode), a motor or sensors, ethernet or some other electronics depending on the project (Kushner, 2011). The Arduino hardware are available in many formats and design enabling different features. The programming is based on hardware wiring. The Arduino software can be run on Windows, Linux or Mac OS (Sandhu, 2016).

## 2.5. Sensors

A sensor is an electronic device that converts a change in physical phenomenon into an electrical signal. It can send the information to computers or other electronic devices. Therefore, it is a part of the interface between the environment or physical world and the electronics (Kenny, 2005). The function of a sensor is to respond to an input physical signal and to convert it into an electrical signal (voltage). It is a semiconductor device that is designed to respond on change in their resistive or capacitive property depending upon the type of sensor. The performance of sensor is characterized by: Transfer function, Sensitivity, Range, Accuracy/Uncertainty, Hysteresis, Linearity, Noise, Resolution and Bandwidth (Wilson, 2005).

# Chapter III.

# Methodology

This chapter presents and discusses the research technique and processes utilized in the current study in a methodical manner. This section contains information on the many procedures that will be utilized to produce the design project. Similarly, several stages of the study's development are depicted in this section. This chapter also includes a detailed explanation of the project's design and development methods. This section also explains the various assessment and consistency tests that will be performed on the project to guarantee design stability and dependability.

## 3.1 General Method Used

The study incorporates the method of waterfall model is a classical model used in system development life cycle to create a system with a linear and sequential approach. It is termed as waterfall because the model develops systematically from one phase to another in a downward fashion. This model is divided into different phases and the output of one phase is used as the input of the next phase. Every phase has to be completed before the next phase starts and there is no overlapping of the phases. (Aleksandar Olic, 2017)

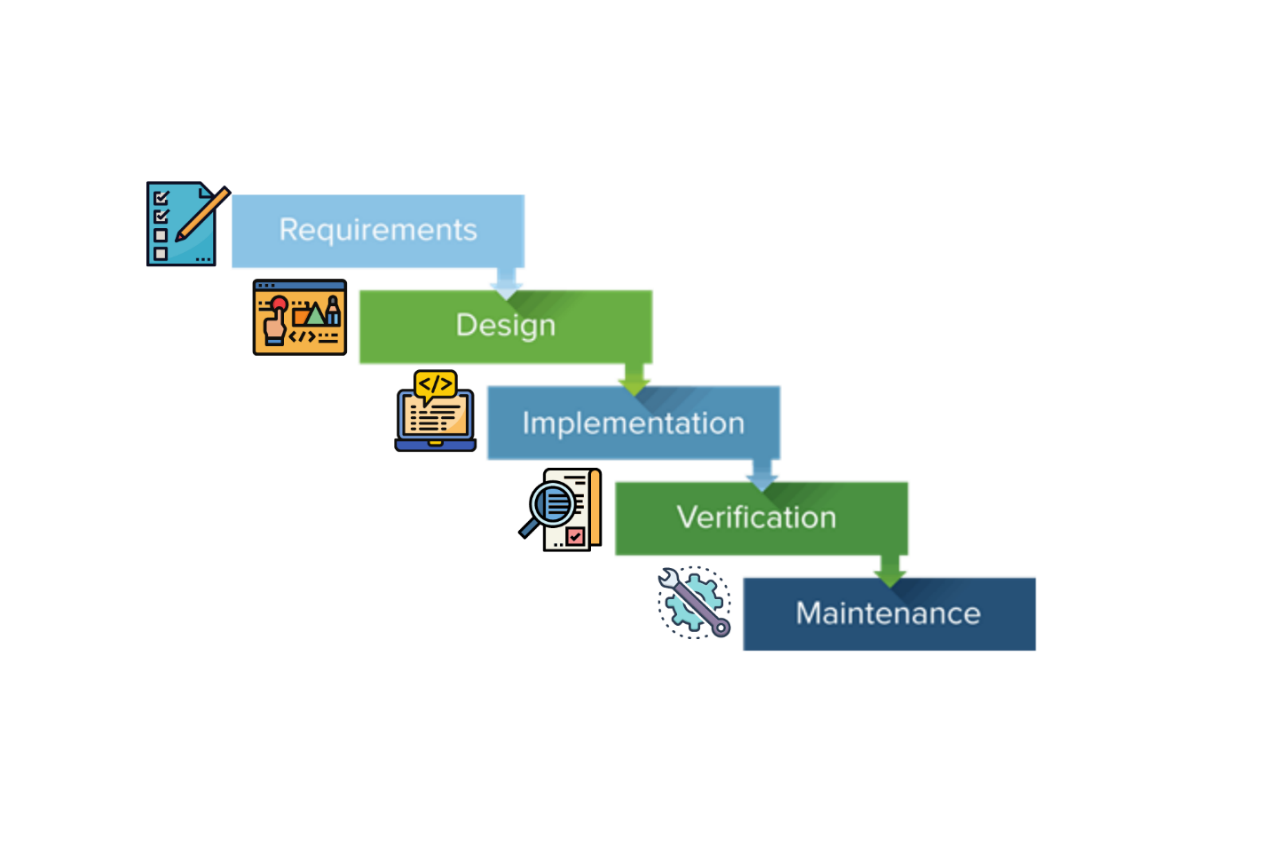


Fig. 3. 1 Waterfall model development process

### 3.1.1 Waterfall Methodology

The Waterfall model is divided into five categories which has a sequential design process in which progress is seen as flowing steadily downwards (like a waterfall) through the phases as needed. The sequence, as shown in Figure 3.1 are: (1) Requirements, (2) Design, (3) Implementation, (4) Verification and (5) Maintenance.

### 3.1.1.1 Requirement Phase

All possible requirements and information gathering of the system to be developed are captured in this phase and documented in a requirement specification document. The proponents accomplished brainstorming and walkthrough in order to understand the specific necessity. In addition, we undertake feasibility test to ensure that the requirements are testable or not. It provides input material to the product being made, and thus the upcoming product is studied, finalized, and marked. It also gives us the extension to decide the hardware or software requirements of the product which will be designed, developed, and captured in all phases.

For this project, it has been identified that temperature, humidity and ammonia concentration in the air has a huge effect on the chickens’ health and nutrition. In addition, several studies indicates that environmental parameters play big role in global, environmental, economic and societal factors.

As such, the main requirement of the project is to develop Smart chicken farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator, that would observe the environmental parameter like temperature, humidity, ammonia gas in the air as well as the chicken feeder weight. Along with this process, these parameters have threshold values so that a humidifier and a fan heater can automatically power up in order to adjust the ideal temperature and humidity requirements inside the chicken coop. With the help of the GSM Communications, the worker or the owner can monitor the present sensor values remotely via sending a prompt message to the system.

### 3.1.1.2 Design Phase

This design phase can be divided into two stages: (1) Logical Design Stage and (2) Physical Design Stage. At the Logical stage, proponents design the system based on the requirements, without regards to hardware and software. In this stage, the study describes the purpose and scope of the project, the flow of each component and integration points. It also implies the proponents to define the plan for a solution which includes logical diagram design and concept design. In essence, this is more on brainstorming and theoretical design stage. Afterwards, at the Physical stage, the proponents transform it into a practical design according to hardware and software specifications. In this stage, it optimizes performance while ensuring data integrity by avoiding unnecessary data redundancies. It ensures that the hardware will comes along with software itself. In conclusion, this stage is when the theoretical thoughts and schemes are made into actual specifications.

After carefully considering the many options, creating a design that will serve as the blueprint for the real product prototype. In order to manipulate the hardware into the software, the proponents use a simulation tool Fritzing for them to analyze the entire circuit while TinkerCAD to build a casing for the sensors.

### 3.1.1.3 Implementation Phase

The requirements and specifications that were developed in the first two phases are now made into an actual building, product, software program or deliverable. This phase is where the real code is written and compiled into an operational application. In other words, it is the process of converting the whole requirements and blueprints into a production environment. All previous planning is put into action during this phase. This phase belongs to the proponents in the Waterfall method, as they take the project requirements and specifications, and code the applications. Proponents take information from the previous stage and create a functional product. Implementing all models, logic stages, and service integrations that were specified in the prior stages are all on this phase. Briefly, this is where all the planning into action.

The proponents use Arduino IDE in order to sum up all the required codes into the system itself. Codes are carefully supervised and written in result of an acceptable and reliable monitoring of environmental parameters.

### 3.1.1.4 Verification Phase

It is also known as verification and validation which is a process for checking that a software solution meets the original requirements and specifications and that it accomplishes its intended purpose. In fact, verification is the process of evaluating software to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase; while, validation is the process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements. Moreover, the verification phase is the outlet to perform debugging in which bugs and system glitches are found, corrected, and refined accordingly.

### 3.1.1.5 Maintenance Phase

It is the process of modifying a software solution after delivery and deployment to refine output, correct errors, and improve performance and quality. Additional maintenance activities can be performed in this phase including adapting software to its environment, accommodating new user requirements, and increasing software reliability

## 3.2 Procedure

The study employed the common elements of engineering design process described by Ford and Coulston.

### 3.2.1 Requirements Specification

**Table I**

*Requirements specification of the Smart chicken farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator*



Marketing Requirements

1. The system can communicate between user and the apparatus.
2. The system should be able to understand prompt messages from the user
3. The system should provide clear and understandable messages.
4. The system should be user-friendly.
5. The system should be able to accurately monitor environmental parameters.
6. The system should be allowed to control of relays when threshold values are reached.
7. The system should be portable.
8. The system should be low-cost in terms of comparison to competitive products.

Table I shows the requirements specification of the Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator. The marketing requirements involves the demand of the user in terms of the systems’ overall performance and quality. The engineering requirements refers to the process of defining, documenting, and maintaining requirements in the engineering design process.

**Table II**

Technical Specification of the Smart chicken farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator



Table II shows the target technical Specification that was derived and determined from the requirements specification and the objectives. The technical specification will provide vision and goal for the final design. Calibration will take place depending on the results, thresholds will then be adjusted to match the actual output.

### 3.2.2 Requirements Specification

### 3.2.2.1 General system architecture

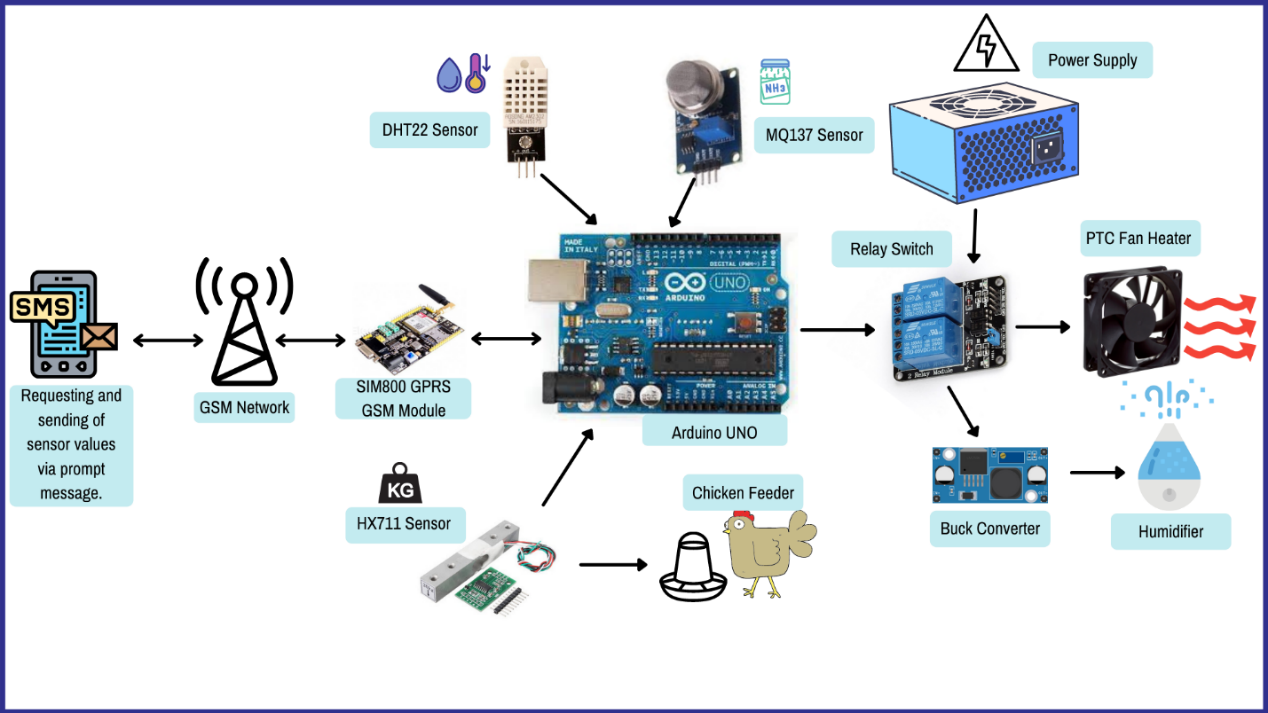


Fig. 3. 2 General System Architecture

The General System Architecture shown in Figure 3.2 discusses the conceptual model that shows the structure and behavior of the system.

It starts when a user is requesting for a message containing the of all the sensor data values gathered inside the chicken coop through sending a prompt message. After that, the message is being delivered to the GSM network and to the GSM Shield. Following to that process, Arduino UNO will gather of all the data of the sensor attached from it. The DHT22 is responsible for the temperature and humidity, MQ137 is in charge for the ammonia gas (NH₃) concentration in the air and HX711 is conducting the sensing of the chicken feeder weight inside the brooder house. If the threshold values for the recommended temperature and humidity inside the coop, the relay switch will power up the humidifier and the fan heater in order to decrease the excess heat or humidity inside the house. Presuming that if the recommended temperature and humidity is attained, the relay switch will cut off the electricity from the power supply with the intention of low power consumption.

## 3.3 Design

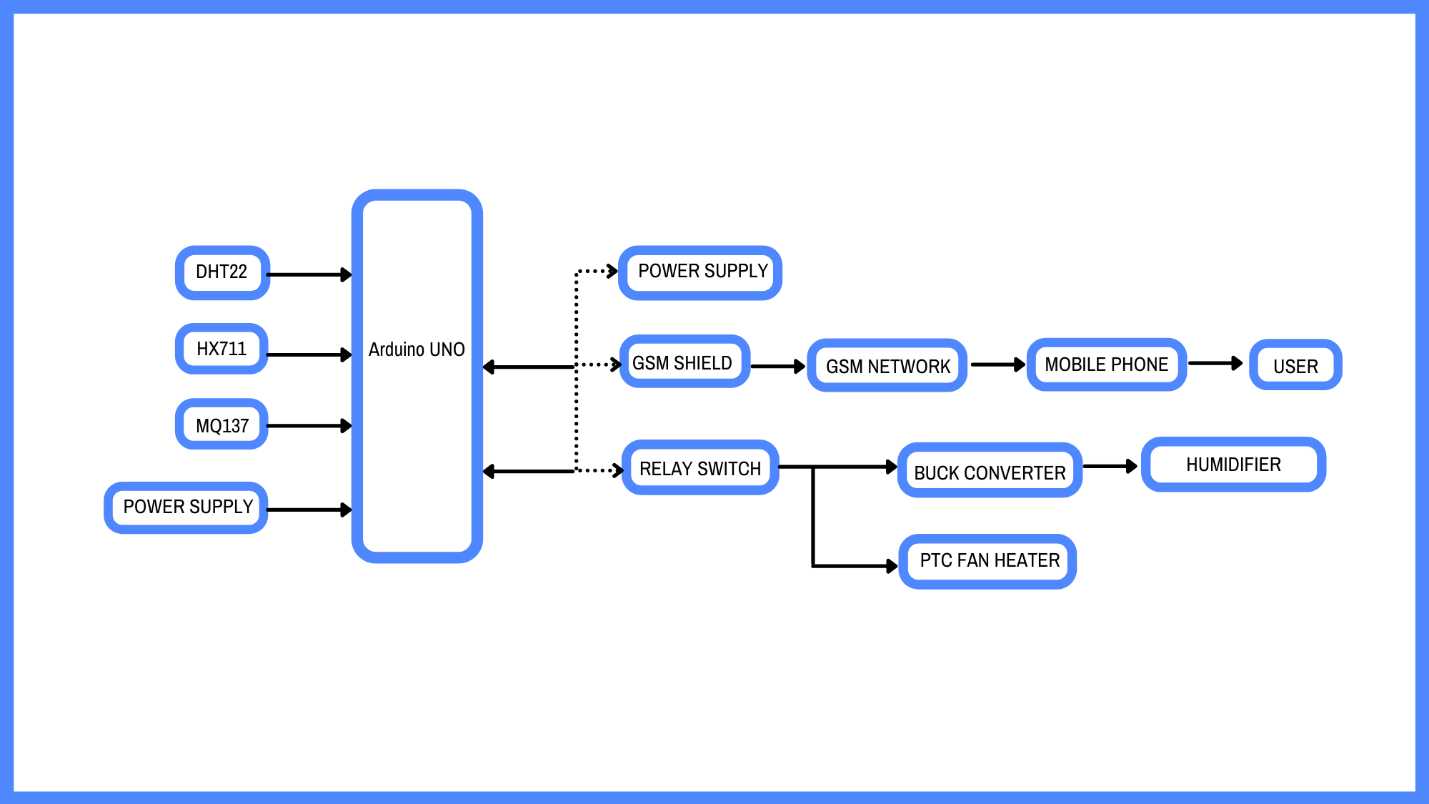


Fig. 3. 3 Level 0 block Diagram of the Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator

Fig. 3.3 Illustrates the Level 0 block diagram of the Smart chicken farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator. This contains the concept alternatives based on the requirement specification discussed previously.

**Table III**

*Level 0 functional requirements of the DHT22 Temperature and Humidity sensor*



### DHT22 Temperature and Humidity Sensor

One of the used sensors in this project is temperature and humidity sensor- DHT22 module also referred to as AM2302. The DHT22 is an analog sensor designed to sense the physical change in heat and moisture when exposed in air with suitable wiring and programming. Its small size, cheap price, low power consumption, quick responses are the characteristics for being one of the best choices for many users. The sensor DHT22 is applicable in HVAC (heating, ventilation and air conditioning), it can be used in testing and inspecting equipment and consumer goods. It is also applicable to use in building a weather station or a humidity regulator. The use of DHT22 sensor has shown its usefulness measuring and controlling temperature and humidity in agriculture, medical, home appliances and many other sectors.

The Figure 3.4 shows the DHT22 temperature and humidity sensor module. The DHT22 sensor module has following performance range and accuracy.

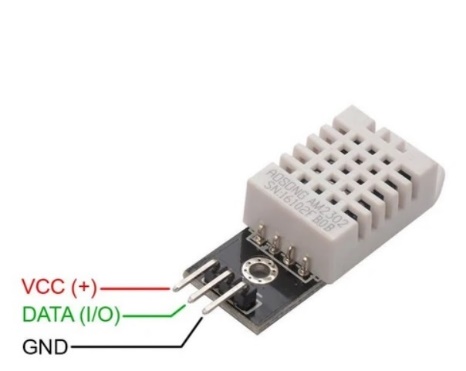
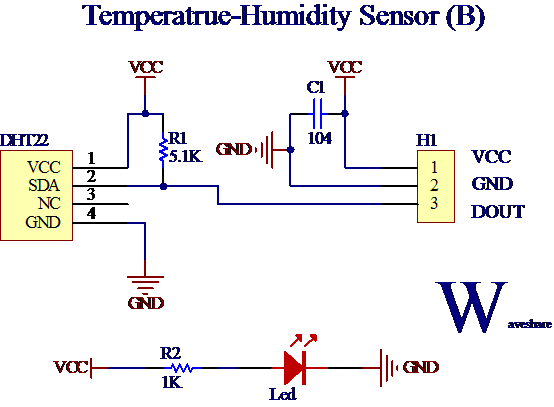
 

Fig. 3. 4 DHT22 sensor module Fig. 3.4. 1 Circuit diagram for DHT22

Humidity is defined as the amount of water vapor contained in air. Usually, it is expressed as absolute humidity, dew point and relative humidity. The sensor used in this thesis project, DHT22, is designed to measure humidity in terms of relative humidity (RH). Relative humidity (RH) is the ratio of the amount of water vapor content of the air to the saturated moisture level at the same pressure or temperature.

Where 𝑅𝐻 is relative humidity, 𝜌𝑤 is the density of water vapor, and 𝜌𝑠 is the density of water vapor at saturation.

DHT22 detects moisture in the air by measuring the electrical resistance between electrodes. It is fabricated with a moisture holding substrate. When substrate absorbs moisture, ionization takes place and results in the increase in conductivity between the electrodes. The relative humidity is proportional to the change in resistance between electrodes due to moisture absorbed. The typical circuit diagram for DHT22 is shown in Figure 3.4.1.

It also uses Single-bus communication protocol (see Fig. 3.4.2) which refers to that only one data line, data exchange system, controlled by the data line to complete. Equipment (microprocessor) through an open-drain or tri-state port connected to the data line to allow the device does not send data to release the bus, while other devices use the bus; single bus usually require an external about 5.1kΩ pull-up resistor, so when the bus is idle, its status is high. Because they are the master-slave structure, only the host calls the sensor, the sensor will respond, so the hosts to access the sensor must strictly follow the sequence of single bus, if there is a sequence of confusion, the sensor will not respond to the host.

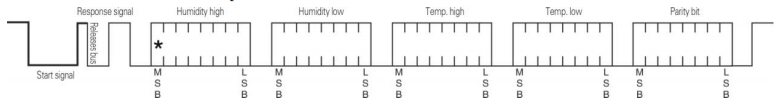


Fig. 3.4. 2 DHT22 Single-bus communication protocol

Single-bus data calculation example

Example 1: 40 Data received:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0000 0010 | 1001 0010 | 0000 0001 | 0000 1101 | 1010 0010 |
| High humidity 8 | Low humidity 8 | High temp. 8 | Low temp. 8 | Parity bit |

Calculate： 0000 0010+1001 0010 +0000 0001+0000 1101= 1010 0010 (Parity bit)

Received data is correct:

Humidity：0000 0010 1001 0010 = 0292H (Hexadecimal)= 2×256 + 9×16 + 2 = 658

Humidity = 65.8%RH

Temp.：0000 0001 0000 1101 = 10DH(Hexadecimal) = 1×256 + 0×16 + 13 = 269

Temp.= 26.9℃

**Table IV**

*Level 0 functional requirements of the MQ137 Ammonia Gas (NH₃) sensor*



### MQ137 Ammonia Gas (NH₃) sensor

Another sensor used in this project is the MQ137 (see Figure 3.5.1) which uses 𝑆𝑛𝑂2 as semiconductor. This is a non-transition-metal oxide. Its work is based on redox reactions [[5]](#_REFERENCES) between the surface of the metal and the measured gas. The molecules of 𝑂- react with the gas and alter the resistance of the sensor (𝑅s), which in turn results in a change in the voltage between the analog signal and the ground pin (see MQ sensor modules circuit diagram in Figure 3.5.2). As such, it is evident that this sensor is best suited for indoor measurements and characterization of emission sources that naturally present much higher concentrations of ammonia including industry or poultry farms.

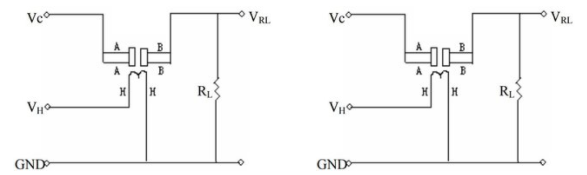
 

Fig. 3.5. 1 MQ137 Ammonia Sensor Module Fig. 3.5. 2 MQ137 sensor circuit diagram

Typical dependence of the MQ137 on Temperature and Relative Humidity

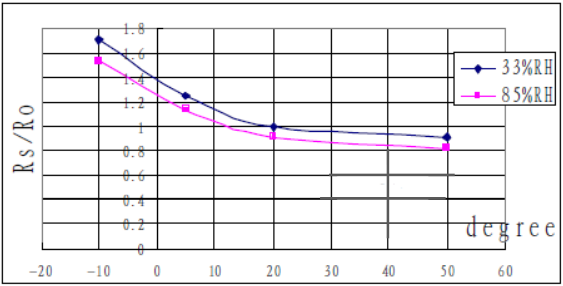


Fig. 3.5. 3 Typical dependence of the MQ137 on Temperature and Relative Humidity

These parameters affect the human levels of comfort, food supplies and many other aspects of life directly and, as such, have been subject to interest since the dawn of civilization. The interest of this study comes in their direct impact on the measurements of ammonia concentration, because of the sensibility of the sensor to temperature and relative humidity variations. This sensibility is due to the variation of the internal resistance (*RS)* with both parameters and the respective change in calculated ammonia concentration. This dependency can be shown in Fig. 3.5.3, from the sensor's datasheet [[6]](#_REFERENCES). It is easily observed in the figure that small variations in temperature and humidity can lead to huge alterations of the sensor's values, being more accentuated with from -10 to 20℃.

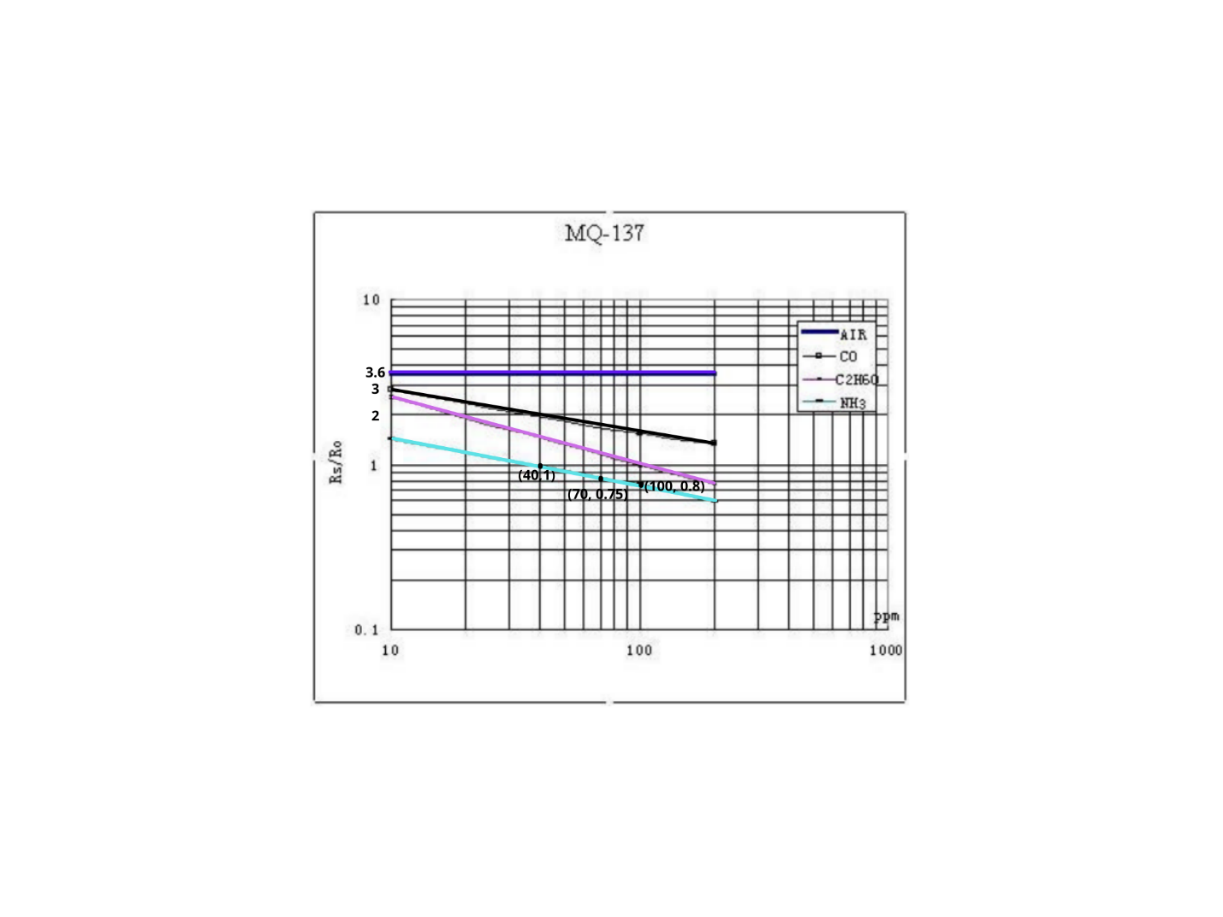


Fig. 3.5. 4 MQ137 Sensitivity Characteristics

Figure 3.5.4 tells us the concentration of a gas in part per million (ppm) according to the resistance ratio of the sensor (RS/R0). RS is the resistance of the sensor that changes depending on the concentration of gas, and R0 is the resistance of the sensor at a known concentration without the presence of other gases, or in fresh air.

Calculations

We can derive a formula to find RS using Ohm's Law:

Where *V* is voltage, *I* is current, and *R* is resistance. Looking back at the circuit from Figure 3.5.2, we can see that *RS*, which is the resistor between pins *A* and *B*, and *RL* are in series. Thus, we can simplify the circuit as shown below and combine the two resistors in a series connection.

Going back to the original circuit from Figure 3.5.2, it can obtain the output voltage at the load resistor using the value obtained for *I* and Ohm’s Law.

*V = I x R*

*VRL = [VC / (RS + RL)] x RL*

*VRL = (VC x RL) / (RS + RL)*

Then derive the above formula to solve for *RS:*

*VRL x (RS + RL) = VC x RL*

*(VRL x RS) + (VRL x RL) = VC x RL*

*(VRL x RS) = (VC x RL) - (VRL x RL)*

*RS = [(VC x RL) - (VRL x RL)] / VRL*

*RS = [(VC x RL) / VRL] – RL*

Where,

*VRL =* Low-Voltage Reference

*RS =* Sensor Resistance

*RL =* Load Resistance

*VC* = Circuit Voltage

This formula will find the values of the sensor resistance for different gases. Figure X.X is shown by using Arduino IDE after 24 hours of preheat time of MQ137 Sensor in order to calculate the *RS, RO* at fresh airand *VRL* with use of C++ programming. The code for calibrating of MQ137 Sensor is indicated in the Chapter 4.1.2

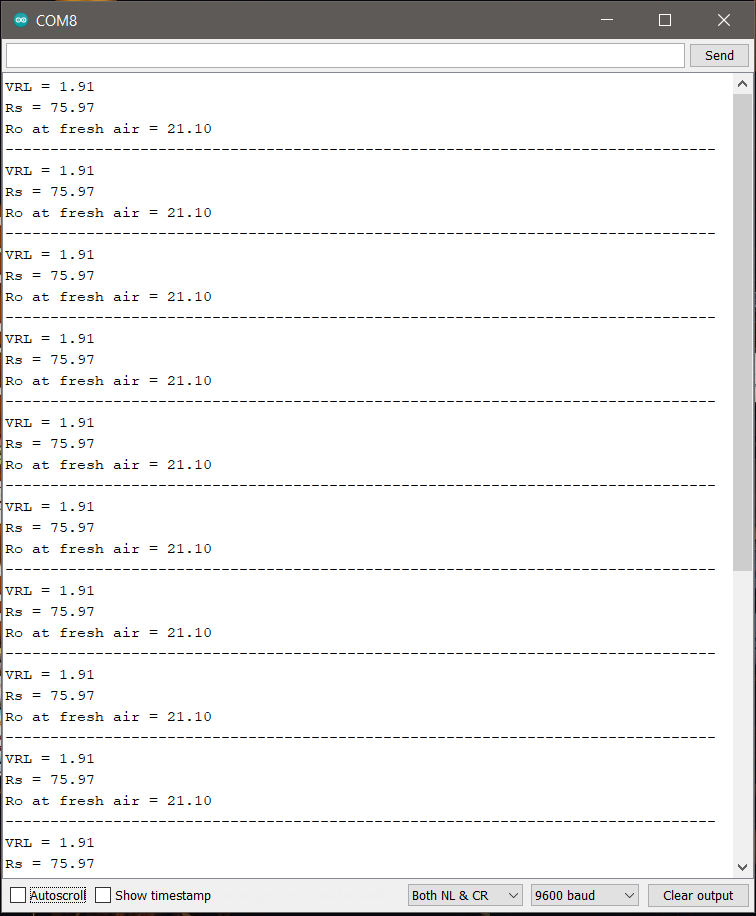


Fig. 3.5. 5 Arduino IDE results for the RS, RO at fresh air and VRL after 24 hours preheat time

Although the NH3 line (cyan color) in figure appears to be linear it is actually not linear. The appearance is because the scale is divided un-uniformly for appearance. Thus, the relating between Rs/Ro and PPM is actually logarithmic which can be represented by the below equation.

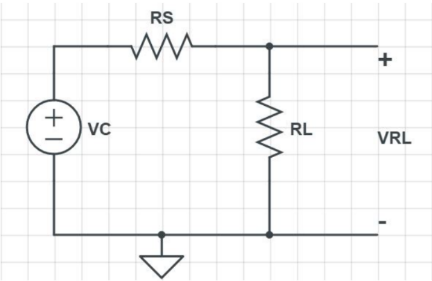
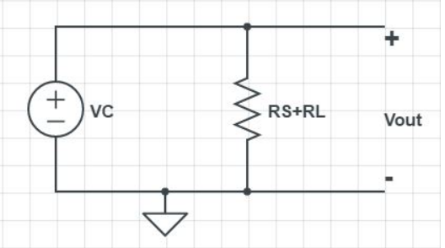
 

Fig. 3.5. 6 Sensor Circuit Simplified Fig. 3.5. 7 RS and RL combined

Where,

*y* = ratio (Rs/Ro)

*x* = ppm

*m* = slope of the line

*b* = intersection point

To find the values of m and b we have to consider two points (*x1*, *y1*) and (*x2*, *y2*) on our gas line. We're dealing with ammonia here, therefore the two points we've examined are (40,1) and (100,0.8).

*m* = [log(*y2*) - log(*y1*)] / [log(*x2*) - log(*x1*)]

*m* = log(0.8/1) / log(100/40)

*m* = -0.243

Similarly, for (b) let’s get the midpoint value (x, y) from the graph which is (70,0.75) as shown in figure 3.6.1 above.

b = log(*y*) - m\*log(*x*)

b = log(0.75) - (-0.243)\*log(70)

b = 0.323

Provided of all of the given values, the formula to equate the value of Rs/Ro to ppm is indicated below:

**Table V**

*Level 0 functional requirements of the HX711 Load cell amplifier*



### HX711 Load Cell Amplifier

A Load Cell Amplifier is a small breakout board for the HX711 Integrated Circuit (IC) that allows to easily read load cells to measure weight. By connecting the amplifier to a microcontroller, it will be able to read the changes in the resistance of the load cell and with some calibration it will be able to get very accurate weight measurements. This can be handy for creating industrial scale, process control, or simple presence detection. The HX711 uses a two-wire interface (Clock and Data) for communication. Any microcontroller's GPIO pins should work and numerous libraries have been written making it easy to read data from the HX711. Load cells uses a four wire Wheatstone bridge [[7]](#_REFERENCES) to connect to the HX711. These are commonly colored Red, Black, White, Green. Each color corresponds to the conventional color coding of load cells:

* Red (Excitation+ or VCC)
* Black (Excitation- or GND)
* White (Amplifier+, Signal+, or Output+)
* Green (A-, S-, or O-)

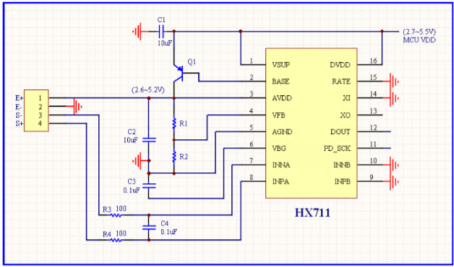
 

Fig. 3.6. 1 HX711 Breakout board Fig. 3.6. 2 HX711 Circuit Diagram

**Table VI**

*Level 0 functional requirements of the SIM800 GPRS GSM Shield*

**

### SIM800 GPRS GSM Shield

The GSM Shield SIM800 provides your Arduino or Microcontroller a way to transmit data wirelessly through SMS, GPRS, or even making a Call using the GSM phone network. it is compatible with most Arduino boards which have the same form factor as the standard Arduino Uno Board. GSM modules or shield such as the SIM800 provides an alternative to RF radio, Bluetooth, and Wi-Fi as a means of wireless communication, where range is a factor, the GSM module will be able to transmit data as long as there is a cellular tower in the area.

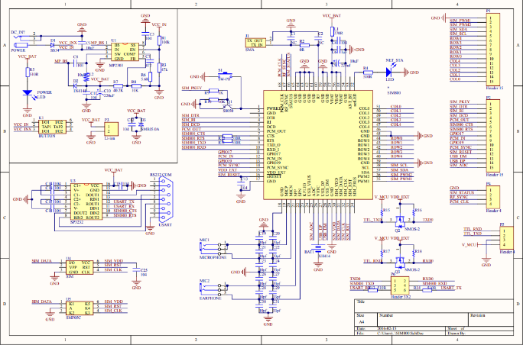
 

Fig. 3.7. 1 SIM800 GPRS GSM Shield Fig. 3.7. 2 SIM800 GPRS GSM Shield Circuit Diagram

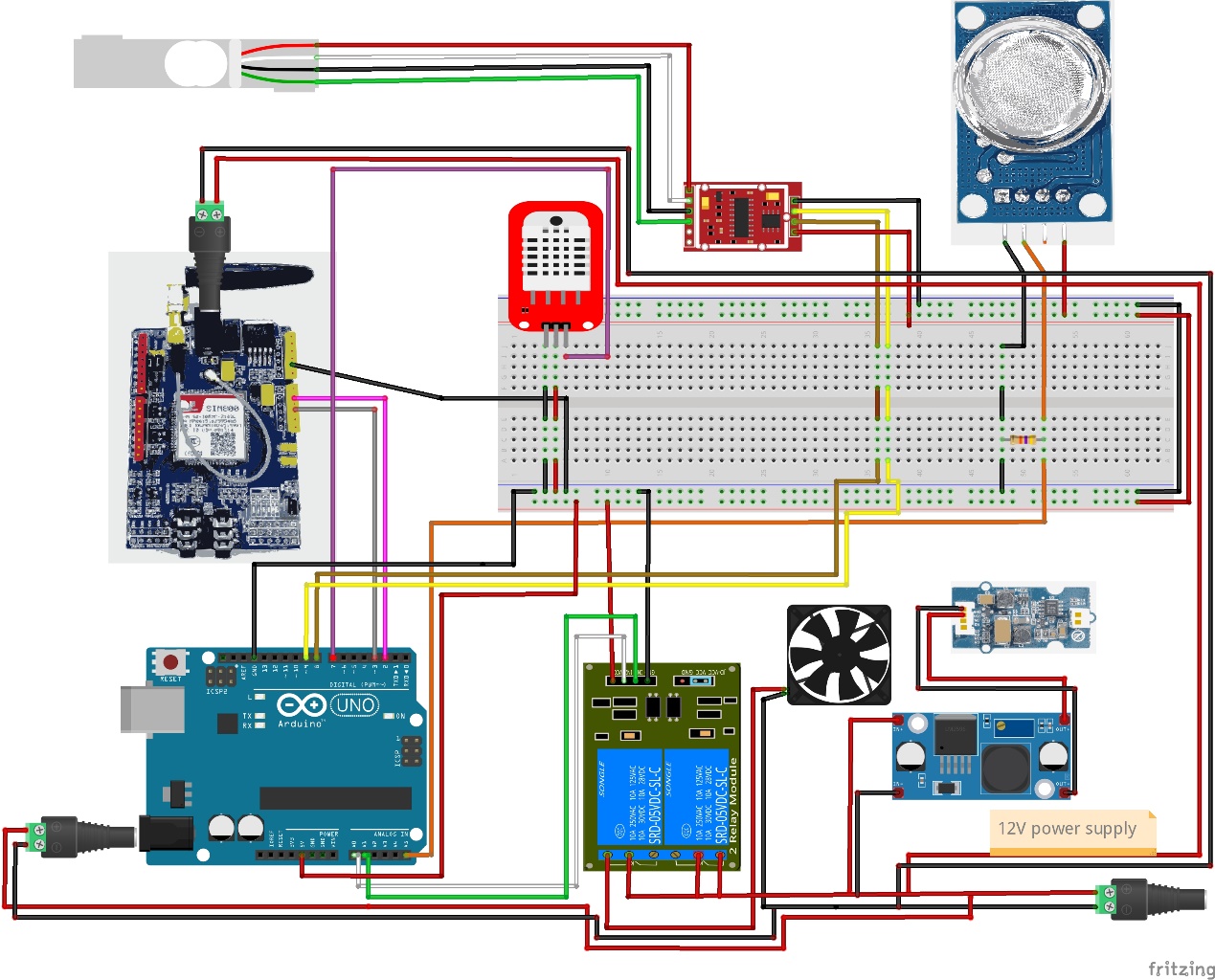


Fig. 3.8. 1 Level 1 diagram of the Smart chicken farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator

Figure 3.8.1 illustrates the level 1 diagram of the Smart chicken farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator. The inputs are Peopleware, Hardware and Software. Under the Peopleware are Principle of Electronics, IOT devices, DC circuits, Power Conversion and C++ language which are required in order to initialize the project that has to be done. Meanwhile, Hardware requirements are peripheral that is being used in this project such as Arduino UNO, MQ137 NH3 sensor, HX711 load cell amplifier, 10kg load cell balancer, DHT22 temp and humidity sensor, SIM800 GSM Shield, Relay switches, DC-DC Buck Converter, 12V DC PTC fan heater, 5V Ultrasonic Mist Maker, 12V power supply, 47k ohm resistor, PCB protoboard and Jumper cables. These components are used to monitor environmental parameters such as temperature, humidity and ammonia gas concentrations in the air. Power supply is being used with the aim to give electricity to the entire system. Eventually, proponents utilized programming software with goal of compilation of written codes into an operational application. It also has the process of converting the whole requirements and blueprints into the production. Proponents used Arduino IDE, Fritzing, TinkerCAD and AT Command Tester. The process phase covered the guide that the researchers did during their study. Which includes Planning, System Analysis, System Design, Development, Testing and Implementation. This process tackles about how the proponents attain the final system so as to achieve a low-cost monitoring system for chicken coops. Lastly, the output phase provided the final output of the study that includes the recommendation and improvement that can be done on the project. It also included a solution to the problem found out during the process phase.

**Table VII**

*Level 1 Functional Requirements of the Smart chicken farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator*

|  |  |
| --- | --- |
| Module | Smart chicken farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator |
| Inputs | * 5V DC power for Arduino UNO * 12V 15A Power supply for PTC fan heater, Humidifier and GSM Shield * Collects all environmental parameter inside the coop * The GSM will notify the microcontroller if a prompt message is received |
| Outputs | * The module will provide necessary actions when an environmental parameter passed it threshold value |
| Functionality | * Sends all the sensor data values when a prompt message from a user is received * Enables remote sensing * Allows of turning on the relay if a certain threshold value is reached |

Table VII shows the level 1 functional requirements of the Level 1 Functional Requirements of the Smart chicken farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator. The second row refers to the inputs of the system which refers to the peripherals need for the system and the processes that system does. The third row refers to the output which declares for what will the system leaves information produced by a product. The last row refers to the functionality which have to do with the uses and tasks that the system will execute depending on the implementation of it.

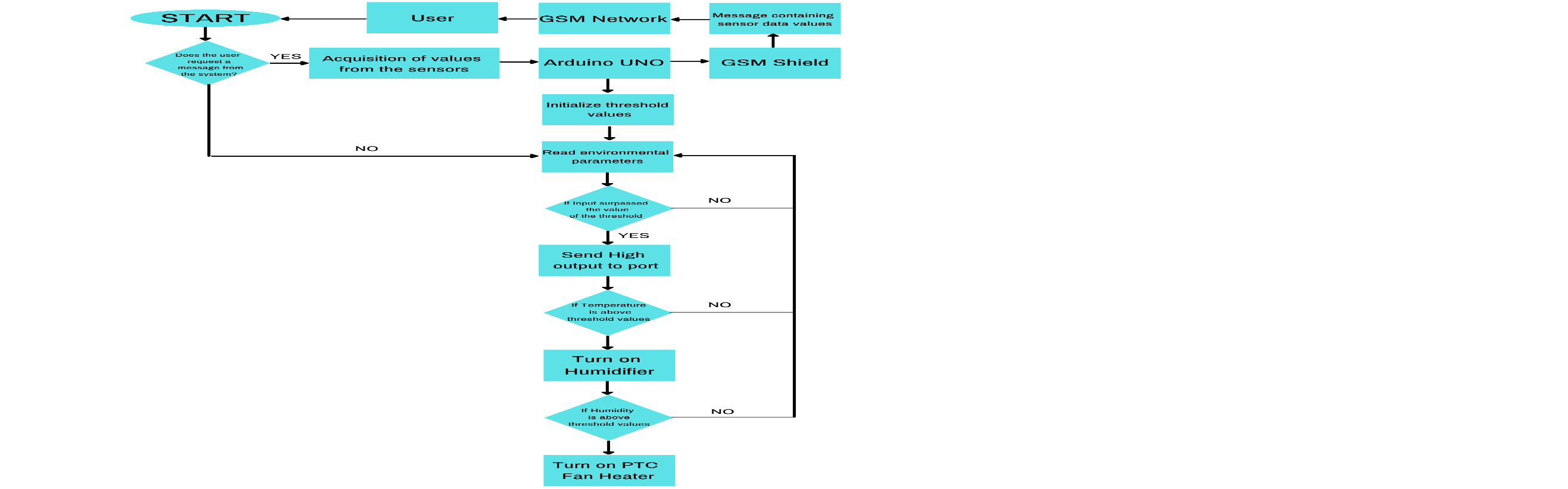


Fig. 3.9. 1 System Flowchart

Fig. 3.9.1. shows how the device works from top to bottom. It demonstrates how the modules start and how they interact with one another. Starting with if the user request for the sensor values and if the threshold values are reached in such a way that relay will automatically turn on remotely. This flowchart ensures that the environmental parameters inside the coop will remain to its recommended values.

## 3.4 Verification and Testing

In this section, all the modules are tested individually to verify functionality, then as a group to test integrated functionality. The main device in this project is the Arduino UNO which serves as the microcontroller. First, the proponents took the modules for calibrating if needed such in a way it would function correctly as it is supposed to be.

### 3.4.1 DHT22 Temperature and Humidity Sensor

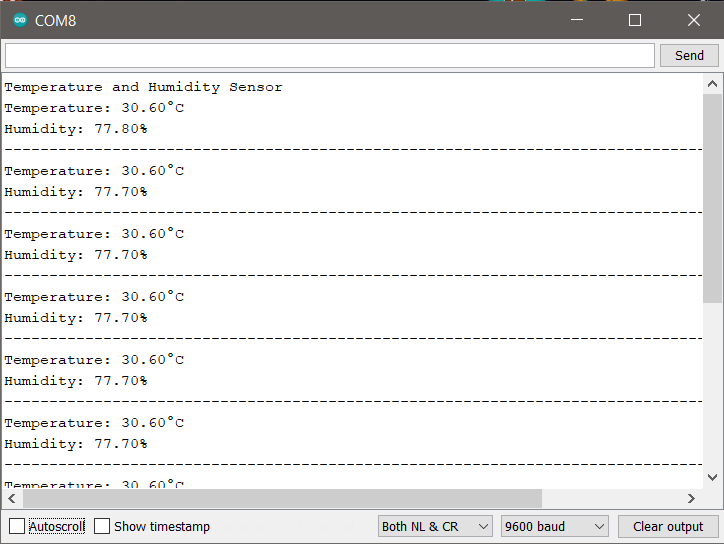


Fig. 3.10. 1 DHT22 Sensor readings using Arduino IDE

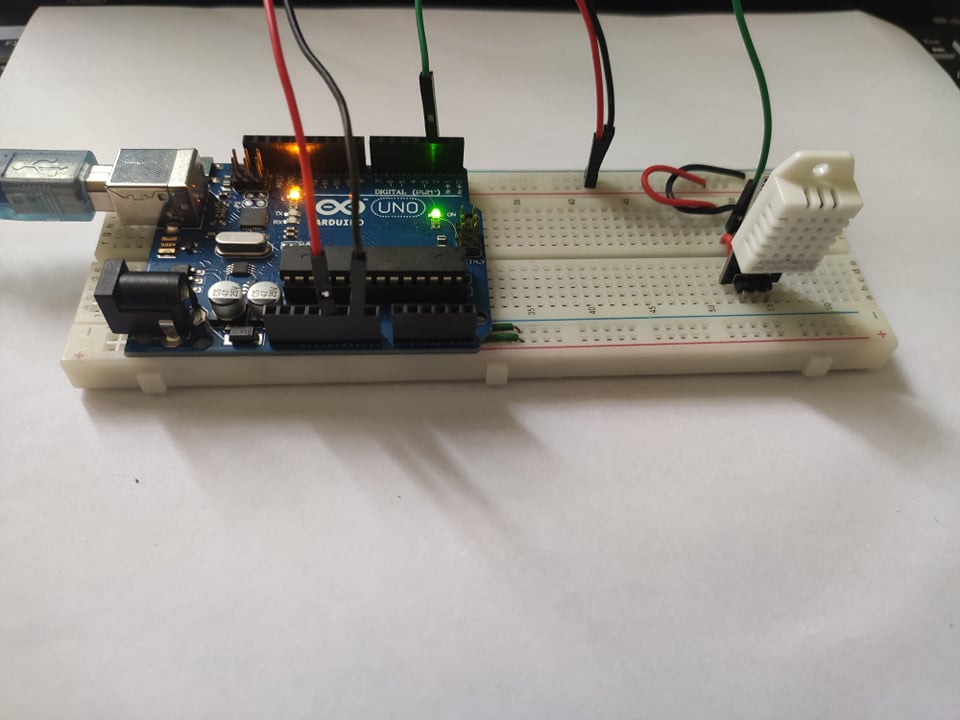


Fig. 3.10. 2 DHT22 Sensor system architecture

### 3.4.2 MQ137 Ammonia Gas (NH₃) Sensor

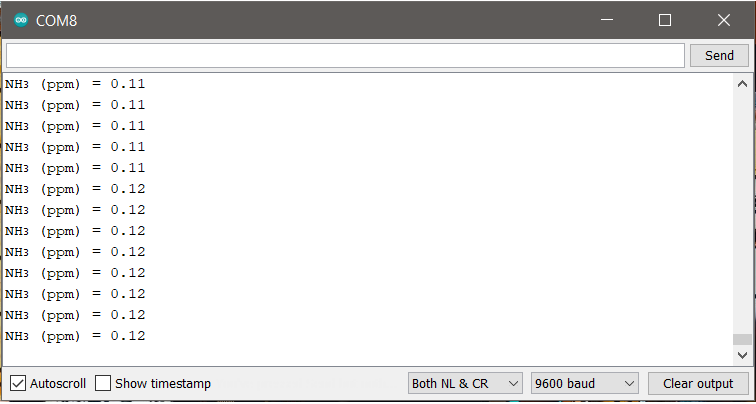


Fig. 3.11. 1 NH₃ reading using Arduino IDE

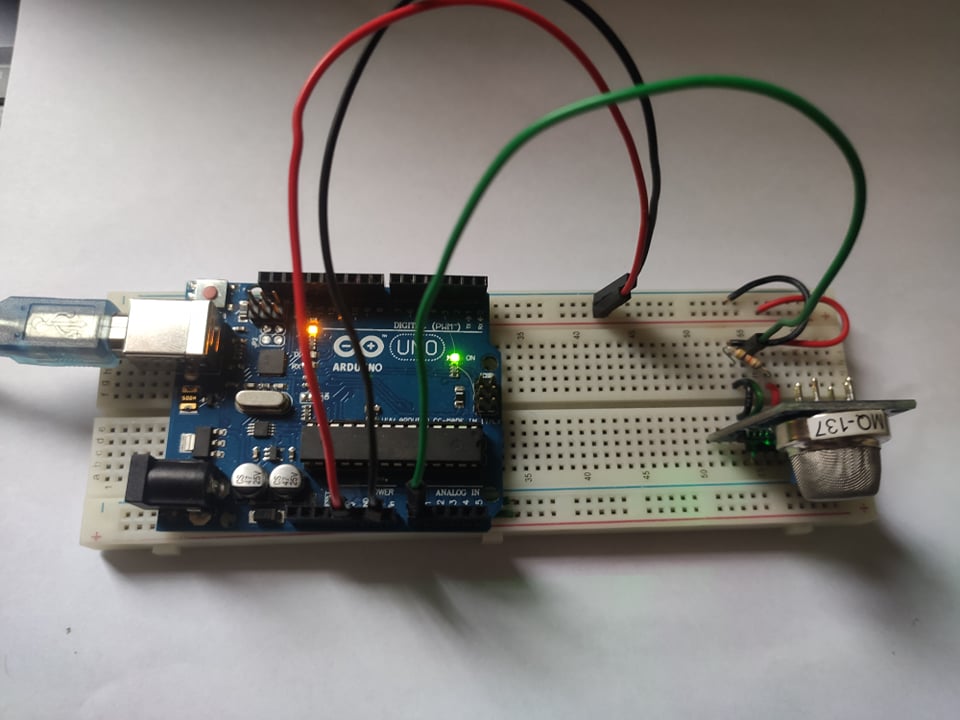


Fig. 3.11. 2 MQ137 Sensor system Architecture

### 3.4.3 HX711 Load Cell Amplifier

**

Fig. 3.12. 1 Test weight using a digital weighing scale

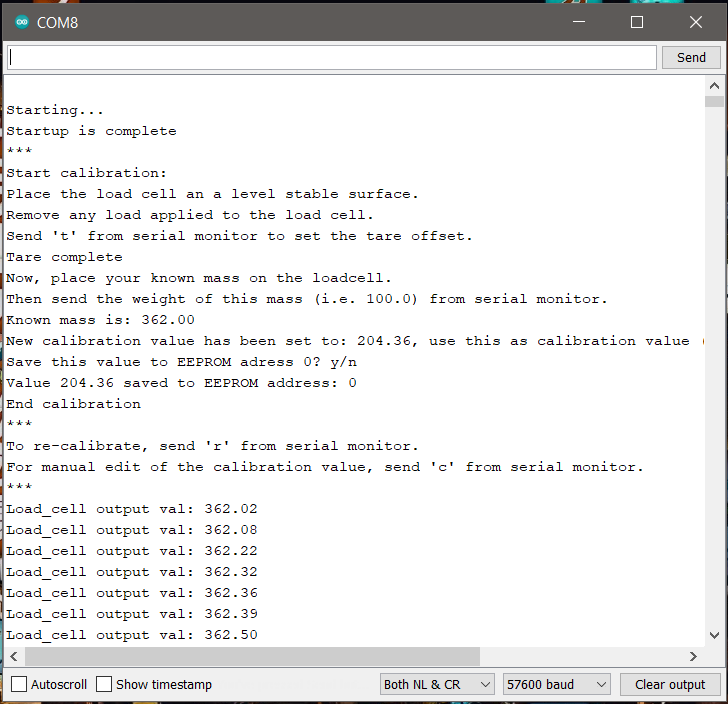
**

Fig. 3.12. 2 HX711 acquiring the Calibration Factor

**

Fig. 3.12. 3 Actual weighing using HX711 with a 10KG Load cell balancer

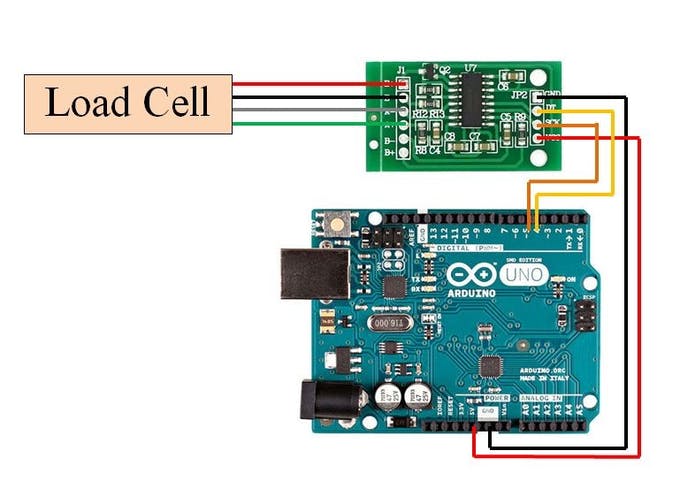
**

Fig. 3.12. 4 HX711 System Architecture

### 3.4.4 SIM800 GPRS GSM Shield

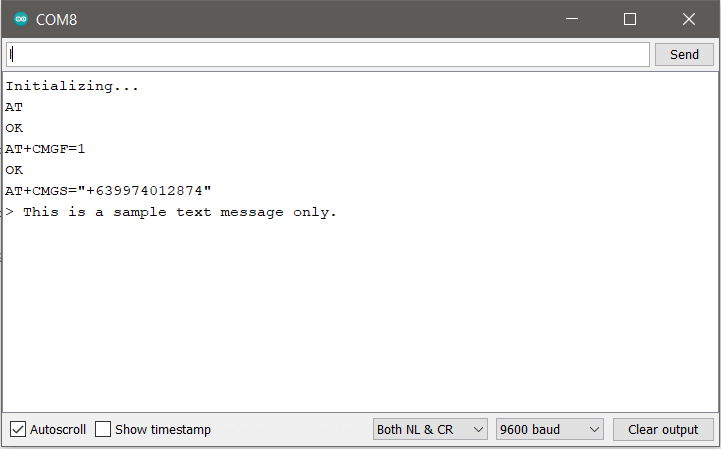
**

Fig. 3.13. 1 Arduino IDE Serial Monitor

**

Fig. 3.13. 2 Sample message screenshot from a mobile phone

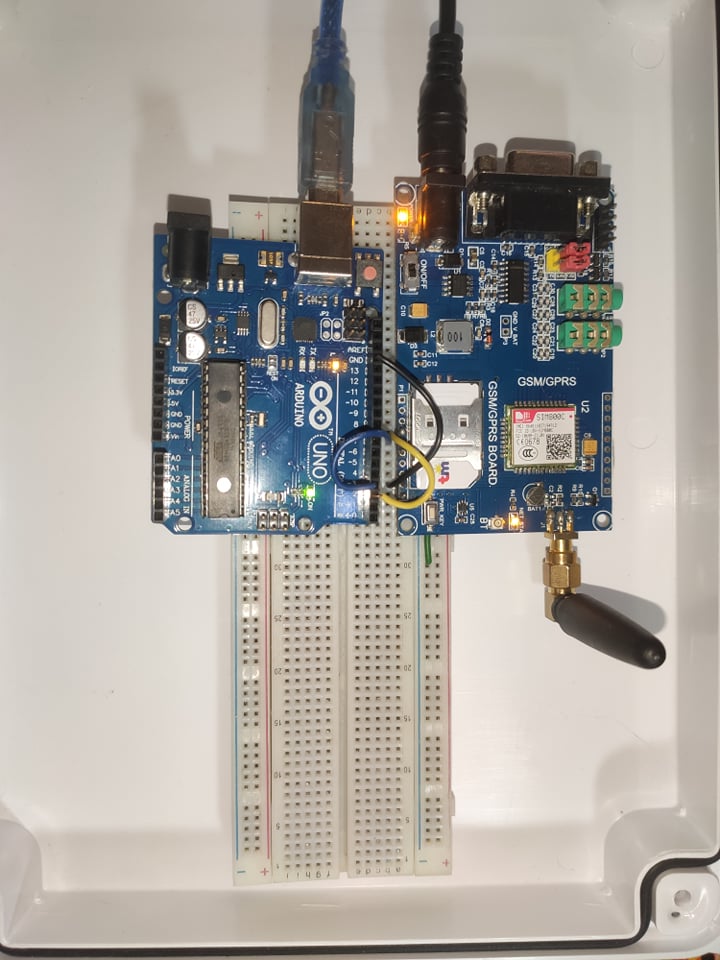
**

Fig. 3.13. 3 SIM800 GPRS GSM Shield system architecture

## 3.5 Unit Testing

Before the modules are being used to a one system, it needs to be calibrated if needed such as to HX711 Load Cell Amplifier. In addition, the modules must test correctly each for what are the environmental parameters are to be observed. For instance, DHT22 sensor is for temperature and humidity; MQ137 for ammonia gas (NH₃) concentration in the air. As proof of when a certain parameter is reached its threshold value, the relay switches will automatically turn on to power up the humidifier and the PTC fan heater.

A unit test verifies that a system module executes a single unit of functionality according to a set of requirements. It’s a standalone test of a system module’s functioning.

**Table VIII**

*Table for Unit Testing*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Description** | **Expected Result** | **Actual Result** | **Date of Testing** |
| **SCF-01** | Tests for DHT22 and MQ137 | The modules used must measure the temperature, humidity and NH₃ concentrations. | The system can monitor environmental parameters remotely. | 07/28/2021 |
| **SCF-02** | Test for HX711 load cell amplifier | The peripheral used must measure weight using load cell balancer. | It measures the chicken feeder weight when it has a mass on it. | 07/28/2021 |
| **SCF-03** | Test for SIM800 GSM GPRS Shield | The component used must communicate with the microcontroller to the user. | It sends sensor data values when it receives a prompt message from the user. | 07/29/2021 |
| **SCF-04** | Test for relay switches | The module used must control the humidifier and PTC Fan Heater | It turns on automatically when a certain threshold value is reached. | 07/30/2021 |

Table VII discusses about the trial and assessment about the modules that are being used by the proponents in this project. It turns that the modules are correctly used and calibrated, as well as it shows sensor values to the Arduino IDE and sends to the user

if a prompt message is received with the use of GSM Shield. The use of relay switches is to remotely activated if a threshold value for a certain environmental parameter is reached. In addition, if the parameter is reached to its minimum value, the power will be cut off with the aim of low power consumption of the project.

# 

# Chapter IV

# Results and Discussions

This chapter goes through the whole system and project in detail, as well as the results of unit, integration, and acceptability testing. After each unit of the prototype has been tested, it provides a detailed description of the results. Similarly, it provides a detailed explanation of the consequences after integrating all of the prototype's parts. Each item is examined and inspected before and after the integration in line with its description and expected consequences. In the future, these results will be used to draw conclusions and offer recommendations.

## 4.1 The Developed System



Fig. 4. 1 The developed system (back view)

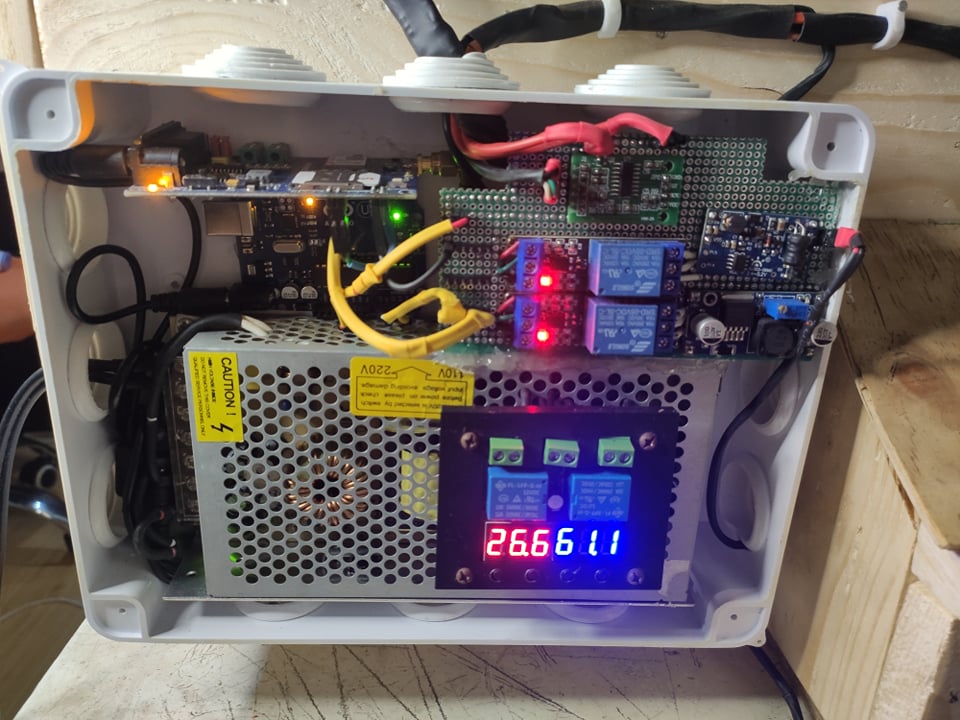


Fig. 4. 2 The developed system (close-up view)



Fig. 4.2. 1 The developed system (front view)

### 4.1.1 Arduino IDE code for DHT22

#include <Adafruit\_Sensor.h>

#include <DHT.h>

#include <DHT\_U.h>

#define DHTPIN 7

#define DHTTYPE DHT22

DHT\_Unified dht(DHTPIN, DHTTYPE);

uint32\_t delayMS;

void setup() {

Serial.begin(9600);

dht.begin();

Serial.println(F("Temperature and Humidity Sensor"));

sensor\_t sensor;

dht.temperature().getSensor(&sensor);

delayMS = sensor.min\_delay / 1000;

}

void loop() {

delay(delayMS);

sensors\_event\_t event;

dht.temperature().getEvent(&event);

if (isnan(event.temperature)) {

Serial.println(F("Error reading temperature!"));

}

else {

Serial.print(F("Temperature: "));

Serial.print(event.temperature);

Serial.println(F("°C"));

}

dht.humidity().getEvent(&event);

if (isnan(event.relative\_humidity)) {

Serial.println(F("Error reading humidity!"));

}

else {

Serial.print(F("Humidity: "));

Serial.print(event.relative\_humidity);

Serial.println(F("%"));

Serial.println("--------------------------------------------------------------------------------------------------");

### 4.1.2 Arduino IDE code for calibrating MQ137

#define RL 47

void setup()

{

Serial.begin(9600);

}

void loop() {

float analog\_value;

float VRL;

float Rs;

float Ro;

analog\_value = 0;

for (int test\_cycle = 1 ; test\_cycle <= 500 ; test\_cycle++)

{

analog\_value = analog\_value + analogRead(A5);

}

analog\_value = analog\_value / 500.0;

VRL = analog\_value \* (5.0 / 1023.0);

Rs = ((5.0 / VRL) - 1) \* RL;

Ro = Rs / 3.6;

Serial.print("VRL = ");

Serial.println(VRL);

Serial.print("Rs = ");

Serial.println(Rs);

Serial.print("Ro at fresh air = ");

Serial.println(Ro);

Serial.println("-------------------------------------------------------------------------------");

delay(1000);

}

### 4.1.3 Arduino IDE code for calibrating HX711

#include <HX711\_ADC.h>

const int HX711\_dout = 8;

const int HX711\_sck = 9;

HX711\_ADC LoadCell(HX711\_dout, HX711\_sck);

unsigned long t = 0;

void setup() {

Serial.begin(57600); delay(10);

Serial.println();

Serial.println("Starting...");

LoadCell.begin();

unsigned long stabilizingtime = 2000;

boolean \_tare = true;

LoadCell.start(stabilizingtime, \_tare);

if (LoadCell.getTareTimeoutFlag() || LoadCell.getSignalTimeoutFlag()) {

Serial.println("Timeout, check MCU>HX711 wiring and pin designations");

while (1);

}

else {

LoadCell.setCalFactor(1.0);

Serial.println("Startup is complete");

}

while (!LoadCell.update());

calibrate();

}

void loop() {

static boolean newDataReady = 0;

const int serialPrintInterval = 0;

if (LoadCell.update()) newDataReady = true;

if (newDataReady) {

if (millis() > t + serialPrintInterval) {

float i = LoadCell.getData();

Serial.print("Load\_cell output val: ");

Serial.println(i);

newDataReady = 0;

t = millis();

}

}

if (Serial.available() > 0) {

char inByte = Serial.read();

if (inByte == 't') LoadCell.tareNoDelay();

else if (inByte == 'r') calibrate();

else if (inByte == 'c') changeSavedCalFactor();

}

if (LoadCell.getTareStatus() == true) {

Serial.println("Tare complete");

}

}

void calibrate() {

Serial.println("\*\*\*");

Serial.println("Start calibration:");

Serial.println("Place the load cell an a level stable surface.");

Serial.println("Remove any load applied to the load cell.");

Serial.println("Send 't' from serial monitor to set the tare offset.");

boolean \_resume = false;

while (\_resume == false) {

LoadCell.update();

if (Serial.available() > 0) {

if (Serial.available() > 0) {

char inByte = Serial.read();

if (inByte == 't') LoadCell.tareNoDelay();

}

}

if (LoadCell.getTareStatus() == true) {

Serial.println("Tare complete");

\_resume = true;

}

}

Serial.println("Now, place your known mass on the loadcell.");

Serial.println("Then send the weight of this mass (i.e. 100.0) from serial monitor.");

float known\_mass = 0;

\_resume = false;

while (\_resume == false) {

LoadCell.update();

if (Serial.available() > 0) {

known\_mass = Serial.parseFloat();

if (known\_mass != 0) {

Serial.print("Known mass is: ");

Serial.println(known\_mass);

\_resume = true;

}

}

}

LoadCell.refreshDataSet();

float newCalibrationValue = LoadCell.getNewCalibration(known\_mass);

\_resume = false;

while (\_resume == false) {

if (Serial.available() > 0) {

char inByte = Serial.read();

}

}

Serial.println("End calibration");

Serial.println("\*\*\*");

Serial.println("To re-calibrate, send 'r' from serial monitor.");

Serial.println("For manual edit of the calibration value, send 'c' from serial monitor.");

Serial.println("\*\*\*");

}

void changeSavedCalFactor() {

float oldCalibrationValue = LoadCell.getCalFactor();

boolean \_resume = false;

Serial.println("\*\*\*");

Serial.print("Current value is: ");

Serial.println(oldCalibrationValue);

Serial.println("Now, send the new value from serial monitor, i.e. 696.0");

float newCalibrationValue;

while (\_resume == false) {

if (Serial.available() > 0) {

newCalibrationValue = Serial.parseFloat();

if (newCalibrationValue != 0) {

Serial.print("New calibration value is: ");

Serial.println(newCalibrationValue);

LoadCell.setCalFactor(newCalibrationValue);

\_resume = true;

}

}

}

\_resume = false;

Serial.println("End change calibration value");

Serial.println("\*\*\*");

}

### 4.1.4 Arduino IDE code for sending a sample message using SIM800 GPRS GSM Shield

#include <SoftwareSerial.h>

SoftwareSerial mySerial(2, 3);

void setup()

{

Serial.begin(9600);

mySerial.begin(9600);

Serial.println("Initializing...");

delay(1000);

mySerial.println("AT");

updateSerial();

mySerial.println("AT+CMGF=1");

updateSerial();

mySerial.println("AT+CMGS=\"+639XXXXXXXXX\"");

updateSerial();

mySerial.print("This is a sample message only.");

updateSerial();

mySerial.write(26);

}

void loop()

{

}

void updateSerial()

{

delay(500);

while (Serial.available())

{

mySerial.write(Serial.read());

}

while(mySerial.available())

{

Serial.write(mySerial.read());

}

}

### 4.1.5 Arduino IDE code for the Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator

#include <SoftwareSerial.h>

#include <Wire.h>

#include "HX711.h"

#include "DHT.h"

#define DHTPIN 7

#define DHTTYPE DHT22

#define DOUT 8

#define CLK 9

#define MQ\_sensor A5

#define RL 47

#define m -0.263

#define b 0.42

#define Ro 82.12

#define RELAY\_FAN\_PIN A1

#define RELAY\_HUMIDIFIER\_PIN A0

const int TEMP\_THRESHOLD\_UPPER = 33;

const int TEMP\_THRESHOLD\_LOWER = 30;

const int HUMID\_THRESHOLD\_UPPER = 60;

const int HUMID\_THRESHOLD\_LOWER = 50;

char Received\_SMS;

char call;

short STATUS\_OK = -1;

String Data\_SMS;

SoftwareSerial mySerial(3, 2);

DHT dht(DHTPIN, DHTTYPE);

HX711 scale(DOUT, CLK);

void setup()

{

mySerial.begin(9600);

Serial.begin(9600);

scale.set\_scale(199.30);

scale.tare();

dht.begin();

Serial.println("Starting ...");

delay(3000);

updateSerial();

ReceiveMode();

}

void loop() {

String RSMS;

while (mySerial.available() > 0) {

Received\_SMS = mySerial.read();

Serial.print(Received\_SMS);

RSMS.concat(Received\_SMS);

STATUS\_OK = RSMS.indexOf("STATUS");

}

if (STATUS\_OK != -1) {

Serial.println("found DHT22, MQ137, HX711");

float humidity = dht.readHumidity();

float temperature = dht.readTemperature();

float VRL;

VRL = analogRead(MQ\_sensor) \* (5.0 / 1023.0);

float Rs;

Rs = ((5.0 \* RL) / VRL) - RL;

float ratio;

ratio = Rs / Ro;

float ppm = pow(10, ((log10(ratio) - b) / m));

float calibration\_factor = 199.30;

float Feeder = scale.get\_units();

if (isnan(temperature)) {

Serial.println("Failed to read from DHT22 sensor!");

} else {

if (temperature > TEMP\_THRESHOLD\_UPPER) {

Serial.println("The humidifier is turned on");

digitalWrite(RELAY\_HUMIDIFIER\_PIN, HIGH);

} else if (temperature < TEMP\_THRESHOLD\_LOWER) {

Serial.println("The humidifier is turned off");

digitalWrite(RELAY\_HUMIDIFIER\_PIN, LOW);

}

}

if (isnan(humidity)) {

Serial.println("Failed to read from DHT22 sensor!");

} else {

if (humidity > HUMID\_THRESHOLD\_UPPER) {

Serial.println("The fan heater is turned on");

digitalWrite(RELAY\_FAN\_PIN, HIGH);

} else if (humidity < HUMID\_THRESHOLD\_LOWER) {

Serial.println("The fan heater is turned off");

digitalWrite(RELAY\_FAN\_PIN, LOW);

}

}

Serial.print("DHT22 Temperature = ");

Serial.print(temperature);

Serial.print("\*C DHT22 Humidity = ");

Serial.print(humidity);

Serial.println(" %");

Serial.print("NH3 = ");

Serial.print(ppm);

Serial.println(" ppm");

Serial.print("Chicken Feeder Weight: ");

Serial.print(scale.get\_units(10), 2);

Serial.println(" grams");

Data\_SMS = "Smart Chicken Farming\nTemperature: " + String(temperature, 1) + "\*C \nHumidity: " + String(humidity, 1) + " % \nNH3: " + String(ppm, 2) + " ppm \nChicken Feeder Weight: " + String(Feeder, 2) + " grams";

updateSerial();

Send\_Data();

ReceiveMode();

STATUS\_OK = -1;

}

}

void updateSerial()

{

delay(500);

while (Serial.available())

{

mySerial.write(Serial.read());

}

while (mySerial.available())

{

Serial.write(mySerial.read());

}

}

void get\_Call()

{

mySerial.println("AT");

updateSerial();

mySerial.println("AT+CMGF=1");

updateSerial();

mySerial.println("AT+CNMI=2,2,0,0,0");

updateSerial();

}

void Serialcom()

{

delay(500);

while (Serial.available())

{

mySerial.write(Serial.read());

}

while (mySerial.available())

{

Serial.write(mySerial.read());

}

}

void ReceiveMode()

{

mySerial.println("AT");

Serialcom();

mySerial.println("AT+CMGF=1");

Serialcom();

mySerial.println("AT+CNMI=2,2,0,0,0");

Serialcom();

}

void Send\_Data()

{

Serial.println("Sending Data...");

mySerial.print("AT+CMGF=1\r");

delay(100);

mySerial.print("AT+CMGS=\"+639XXXXXXXXX\"\r");

mySerial.print(Data\_SMS);

delay(500);

mySerial.print((char)26);

delay(500);

mySerial.println();

Serial.println("Data Sent.");

delay(500);

}

## 4.2 Verification and Testing Result

### 4.2.1 Unit Testing

A unit test verifies that a system module executes a single unit of functionality according to a set of requirements. It is an isolated test of a system module's functionality. In this part, the Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator, the proponents conducted a test to find out if our system was passable or not.

**Table IX**  
Result of Unit Test of Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator



Table VIII shows the result of the unit test of Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator. In step 1, the IDE generated no warning or errors. In step 2, It shows the result fast without time delayed and the output changes based on the detection of humidity and temperature. In step 3, the MQ137 did its task for detected Ammonia gas concentrations in the air. In step 4, with proper calibration, the HX711 perform its weight sensing operations. In step 5, the SIM800 GPRS GSM Shield can successfully send and receive text message when it is needed. In step 6, the humidifier powers up when the threshold value of temperature is reached. Lastly, in step 6, the PTC fan heater starts up when the threshold value of humidity is reached.

The overall test result is that the Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator provided the correct environmental parameter measurements to the MCU.

1. Integration Test
2. The behavior of the integrated system is verified through integration testing. After the system modules have passed a unit test, it is carried out.
3. The proponents conduct a test to see if the result is immediately transferred to the mobile user when a prompt message is received by the system itself.

**Table X**  
Result of Integration Test of Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator



Table IX shows the result of the integration test of Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator. In step 1, the program was precisely gathers all of the sensor data values. In step 2, when a prompt message is sent by the user, the system will reply a message that includes the data gathered. In step 3 and 4, when the weather is humid or hot, the PTC fan heater and the humidifier will automatically turn on and off when the limit values is reached respectfully.

The entire outcome is outstanding in using Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator

### 4.2.2 Acceptance Test

The acceptance test ensures that the system complies with the criteria. After the system has passed an integration test, it is put to the test. The main goal of this test is to determine whether the system complies with the criteria and whether it is suitable for delivery.

**Table XI**

*Result of Acceptance Test of Engineering Requirement 1*



Table X shows the result of the acceptance test of the Engineering Requirement 1. In step 1, the program was statically tested to verify accuracy. In step 2, the result of the test is direct to the mobile user. In step 3, the system can remotely control and regulate environmental conditions. The overall test result receives 99% accurate data from the system.

## 4.3 Data Result of Smart Chicken Farming: Ammonia and Feeds Monitoring with Remote Temperature and Humidity Regulator

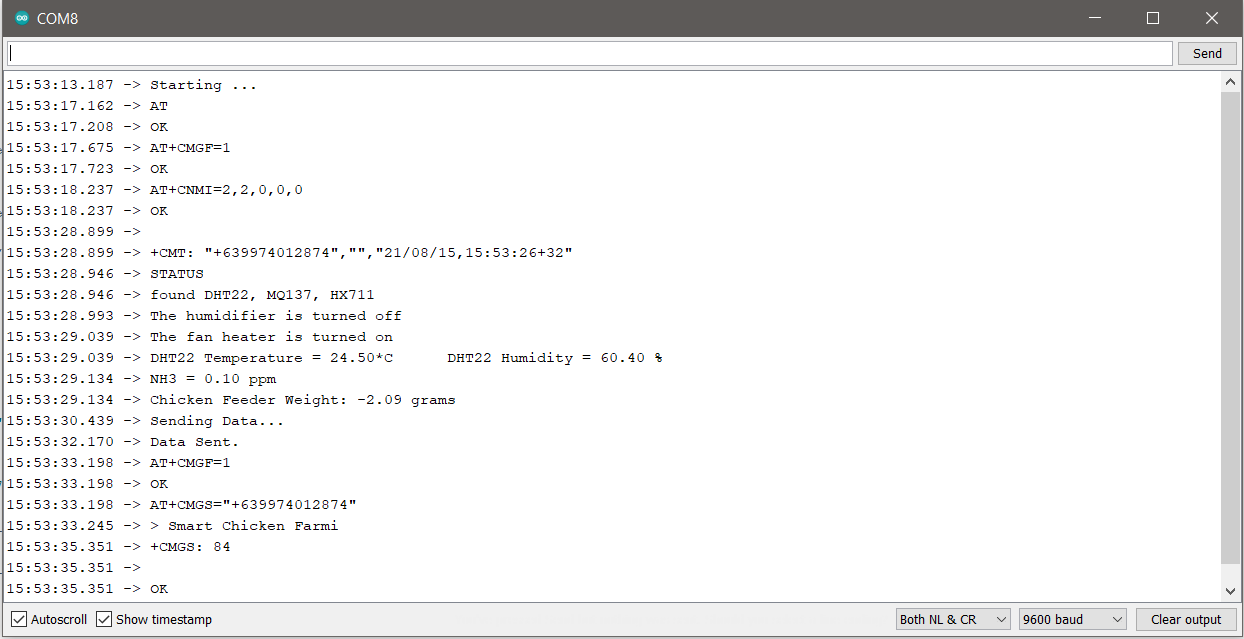


Fig. 4. 3 Data result on Arduino IDE

In this case, the proponents utilized the Arduino IDE's serial monitor to communicate with the computer and the user. The Arduino IDE was also used to see if the status message was the same.

## 4.4 Data Result of the system when a user sends the prompt message

**

Fig. 4. 4 System replied message when a prompt message is received

When the system receives a prompt message, it begins collecting data from the sensor and sending it back to the user. This procedure employs remote monitoring for the user in order to facilitate work.

## 4.5 Data Result of a Humid weather

Fig. 4.5. 1 High humidity threshold is reached Fig. 4.5. 2 Fan Heater air reached 42.9C

When a user sends a prompt message to the system and the device detects a high level of humidity within the chicken coop, the PTC Fan Heater will automatically switch on to achieve the necessary humidity for brooding hens by removing surplus humidity. The fan heater will automatically switch off if the realative humidity falls below 50%, resulting in reduced power usage.

## 4.6 Data Result of a Hot weather

 Fig. 4.6. 1 The temperature reaches > 32C Fig. 4.6. 2 The mist maker automatically turns on

The system has a 32°C temperature threshold, and if the system detects a high temperature within the chicken coop that exceeds this number, the ultrasonic mist maker will immediately switch on to remove the extra heat. The device will considerably aid the health of the hens, allowing the farmers to obtain high-quality meat.

## 4.7 Data Result of a high Ammonia gas concentration in the air

Fig. 4.7. 1 High NH₃ gas detected Fig. 4.7. 2 Ammonia gas releasing to the sensor

The proponents employed a combination of Calcium Hydroxide (Ca(OH)2) and Ammonium Chloride (NH4Cl) in a test bottle that was immersed in hot water to produce gaseous Ammonia gas. Ammonia is created when the nitrogen in poultry manure is broken down by bacteria. It impacts poultry bedding, litter and the overall air quality in chicken houses. The concentration of ammonia in poultry housing is increased by environmental conditions, such as high temperatures and moisture.

## 4.8 Data Result of the Chicken Feeder Weight Sensor

Fig. 4.8. 1 Chicken feeder status message Fig. 4.8. 2 A test object is placed on the feeder

As a consequence of this data, the user will be able to remotely determine whether the chicken feeder needs to be replenished. In some cases, if the feeder is near to be empty it must be refilled in order for the hens to maintain excellent health and nutrition. In addition, the user may determine the hens' average daily food intake.

# Chapter V

# Summary of Findings, Conclusions and Recommendations

This chapter covers three key components of the research. It begins by summarizing the findings from the preceding chapter. Second, it derives inferences from the data. Third, it offers a collection of suggestions that may be utilized to improve the design project even more.

## 5.1 Summary of Findings

The following conclusions are summarized in this study:

* The acceptance test of engineering requirement 1 shows that the program was statistically tested to ensure accuracy.
* The acceptance test of engineering requirement 2 indicates that the result of the tests is direct to the user.
* The acceptance test of engineering requirement 3 proves that the system can remotely control and regulate environmental conditions
* The acceptance test of engineering requirement 4 appears that with the use of proper calibration, it is proved that the sensor is working properly.
* The acceptance test of engineering requirement 5 manifests that the design must be straightforward but effective.
* The acceptance test of engineering requirement 6 shows that the user-friendliness of the system is critical to the user's comprehension and adaption.

## 5.2 Conclusions

The following are the conclusions of this research:

* Concerning with the foremost accuracy with the detection of ammonia gas concentrations in the air, Resistance value of MQ-137 is different to various kinds and various concentration gases. Therefore, when using this components, sensitivity adjustment is very necessary. we recommend that you calibrate the detector for 10ppm NH3 concentration in air and use value of Load resistance that (RL) about 47 KΩ (10KΩ to 100KΩ).
* Based on the temperature and humidity readings, the sensors’ accuracy is very substantial. For instance, the DHT11 is best when measuring with temperature within the range of 0°C to 50°C with ±2°C accuracy and for humidity it is best inside the parameters of 20% to 80% with ±5% accuracy. While DHT22 has the outstanding environmental readings with the range of temperature of -40°C to 80°C with ±0.5°C accuracy and 0% to 100% with ±2-5% accuracy for the humidity.\
* In order to obtain the best possible outcome, the researcher must use the right calibration factor while obtaining the proper weighing procedure.
* The researcher must be aware of the required temperature and humidity for brooding chicken in relation to the country's weather.
* Because mobile signal strength is so important in this study, it is recommended that this project be carried out in an open location with a strong mobile signal.

## 5.3 Recommendations

This study highly recommended to future researchers the following:

* This project aims only when a user sends a prompt message to the system and not recording of all the sensor data values, future researches can use Cloud Storage in order to sum up all the values in a sequential time manner.
* Future researchers can upload this project to Blynk App with the goal of immediate automation as well as the recording of data sensor readings to increase the automation of the humidifier and the PTC fan heater in real-time conditions. This also saves money since it eliminates the requirement for prepaid load on both the user and the system.
* Future researchers can utilize Text-to-Speech (TTS) modules to assist people with disabilities, particularly the visually challenged. This is to make it easier for them to utilize as well as to monitor environmental conditions in real time.
* Future researchers can use USSD (Unstructured Supplementary Service Data), which is seven times quicker than SMS and has interactive navigation for faster two-way communication, to share data quickly. It isn't reliant on mobile software or a SIM card. This means it can function without either and just requires access to the GSM network.

## REFERENCES

[1] Marangoni, F., et. Al (2015). Role of poultry meat in a balanced diet aimed at maintaining health and wellbeing: an Italian consensus document. Retrieved from <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4462824/>

[2] Chiung-Liu, C. (2019). Food Safety and Traceability of Agricultural Products. Retrieved from <https://ap.fftc.org.tw/article/1401>

[3] Manitoba, C.A., (2002). Brooding Temperatures for Small Poultry Flocks. Retrieved from <https://www.gov.mb.ca/agriculture/livestock/production/poultry/booding-temperatures-for-small-poultry-flocks.html>

[4] Eizenberg, E., Jabareen, Y. (2016). Social Sustainability: A New Conceptual Framework. Retrieved from <https://www.mdpi.com/2071-1050/9/1/68/pdf>

[5] Antunes, A. (2019). Study of reliability of electrochemical sensor MQ-135 and MQ-137. Retrieved from <https://docplayer.net/155355931-Study-of-reliability-of-electrochemical-sensor-mq-135-and-mq-137-in-concern-of-ammonia-gas-in-male-urinals-implemented-on-iot-based-system.html>

[6] Winsen Sensors (2015). MQ137 Ammonia Gas Sensor Datasheet. Retrieved from <https://www.winsen-sensor.com/d/files/semiconductor/mq137.pdf>

[7] Load Cell Central (2009). Load Cell and Strain Gauge Basics. Retrieved from <https://www.800loadcel.com/load-cell-and-strain-gauge-basics.html>