

University of Southern Indiana
Pott College of Science, Engineering, and Education
Engineering Department
8600 University Boulevard
Evansville, Indiana 47712

Autonomous Moving Chicken Coop

Alex Malone, EE, Claribeth De Leon, MFE, Joshua O'Brian, EE, Kennedy Sermersheim, MFET,
Will Craig, ME

Advisor: Bradley L. Kicklighter, PE

ENGR 491 – Senior Design
Spring 2025

ACKNOWLEDGEMENTS

In this section we would like to acknowledge certain individuals whose assistance with this project has proved to be paramount to its success, Professor Bradley Kicklighter for his role as our project advisor, Dr. Ryan Integlia, Dr. Susan Ely, and Dr. Todd Nelson for their instruction in our preliminary design classes, Dr. Paul Kuban and Kelly Certain for their administrative support and parts ordering, Mr. Justin Amos for training our group to use the Applied Engineering Center and fabricating our steel plate for battery box and steel axels, and the USI Solar Splash Club for donating electrical power components for our project. Lastly, we would also like to recognize the University of Southern Indiana, in particular, the entirety of the engineering department and its faculty.

ABSTRACT

The purpose of this project was to develop a system that automates the relocation of a chicken coop, while also automatically feeding and watering the chickens inside, helping farmers reduce the time and effort required for daily chicken care. The key feature of the system is its ability to automatically move the coop across different terrains, ensuring optimal conditions for both the chickens and the land. In addition to mobility, sensors and user alerts were integrated to maintain consistent performance and notify the user of any maintenance needs. By automating multiple aspects of chicken care, this project will provide a modern, efficient solution for small-scale farmers.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	ii
ABSTRACT.....	iii
LIST OF FIGURES/TABLES	vi
1.0 INTRODUCTION.....	1
2.0 BACKGROUND	1
2.1 STATEMENT OF PROBLEM.....	1
2.2 REVIEW OF EXISTING SOLUTIONS.....	2
2.3 PROJECT OBJECTIVE STATEMENT.....	5
3.0 CONCEPTUAL DESIGN	6
3.1 REQUIREMENT SPECIFICATIONS.....	6
3.2 CONSTRAINTS.....	6
3.3 SPECIFIC AIMS	7
3.3.1 AUTOMATED MOVEMENT	7
3.3.2 CHICKEN’S WELL-BEING.....	7
3.3.3 FOOD/WATER DISPLAY	8
3.4 PERFORMANCE MEASURES.....	8
3.5 COOP FRAMEWORK.....	8
4.0 SYSTEM DESIGN.....	10
4.1 SYSTEM HIERARCHY	10
4.2 MOVEMENT	10
4.2.1 MOVEMENT IMPLEMENTATION.....	11
4.3 CONTAINERS.....	23
4.3.1 FEED & WATER IMPLEMENTATION	24

4.4 MONITORING	29
4.4.1 MONITORING IMPLEMENTATION	29
4.5 HOUSING/ENVIRONMENT	33
4.5.1 HOUSING/ENVIRONMENT IMPLEMENTATION	34
4.6 ELECTRICAL POWER	44
4.6.1 ELECTRICAL POWER IMPLEMENTATION	44
5.0 RESULTS	50
6.0 FUTURE IMPROVEMENTS	51
7.0 REFLECTIONS	51
ALEX MALONE	51
CLARIBETH DE LEON	52
JOSHUA O’BRIAN	52
KENNEDY SERMERSHEIM	53
WILL CRAIG	54
8.0 CONCLUSION	55
REFERENCES	56
APPENDIX	62
APPENDIX A: PROJECT TIMELINE	62
APPENDIX B: BILL OF MATIERIALS	63
APPENDIX C: Failure Mode and Effects Analysis (FMEA)	64
APPENDIX D: PERT CHART	65
APPENDIX E: ABET OUTCOME 2, DESIGN FACTOR CONSIDERATIONS & STANDARDS	67
APPENDIX F: SOLIDWORKS’ DRAWINGS	68

Motor Enclosures	68
Wheel Axles	69
Side View of Spool	70
Front View of Spool	70
APPENDIX G: DATASHEET HYPERLINKS	71
APPENDIX I: PROJECT CODE	72

LIST OF FIGURES/TABLES

<i>Figure 1: Camera Project [1]</i>	2
<i>Figure 2: Automatic Feeder [2]</i>	3
<i>Figures 3 & 4: Sensor Detection Project [3]</i>	4
<i>Figure 5: Automatic Barn [4]</i>	5
<i>Figure 6: Back Side of Coop</i>	9
<i>Figure 7: Front Side of Coop</i>	9
<i>Figure 8: Automated Chicken Coop System-Breakdown</i>	10
<i>Figure 9: Movement Subsystem-Breakdown</i>	11
<i>Figures 10 & 11: Caster Wheels</i>	12
<i>Figure 12: Plastic Caster Wheels [9]</i>	13
<i>Figure 13: Ramp Caster</i>	13
<i>Figure 14 & 15: SolidWorks 3D printed model (left) and final steel model (right)</i>	14
<i>Figure 16: Motor hub attached to axle</i>	14
<i>Figure 17: Axle and motor hub paired with motor shaft</i>	15
<i>Figure 18: Lock pin on wheel</i>	16
<i>Table 1: Motors Decision Matrix</i>	16

<i>Figure 19: Wheels Motor [10]</i>	17
<i>Figure 20: Motor housing enclosure</i>	19
<i>Table 2: Microcontroller Decision Matrix</i>	20
<i>Figure 21: Arduino Mega 2560 [14]</i>	21
<i>Table 3: Motor Driver Decision Matrix</i>	22
<i>Figure 22: BTS7960 Motor Driver [16]</i>	22
<i>Figure 23: Food Containers Subsystem-Breakdown</i>	23
<i>Figure 24 & 25: Water and Feed Containers</i>	24
<i>Figure 26: Ultrasonic Feeder Sensor [24]</i>	26
<i>Figure 27: Contact Water Sensor [25]</i>	27
<i>Figure 28: LCD Screen [26]</i>	28
<i>Figure 29: Mounted LCD Screen</i>	28
<i>Figure 30: Monitoring</i>	29
<i>Figure 31: ESP32-CAM Board (with OV2640 camera) and FT23RL USB to TTL Serial Converter [27]</i>	30
<i>Figure 32: Monitoring System Camera Mounted in the Autonomous Chicken Coop</i>	31
<i>Figure 33: ESP32-CAM User SSID and Network Password Input</i>	31
<i>Figure 34: Programming the ESP32-CAM Using the FT2RL</i>	32
<i>Figure 35: Monitoring System User Interface</i>	33
<i>Figure 36: Housing/Environment Subsystem-Breakdown</i>	33
<i>Figure 37: Initial Design of the Drawbridge Door</i>	35
<i>Figure 38: Final Design of the Drawbridge Door</i>	35
<i>Figure 39: Drawbridge Door Motor [28]</i>	36
<i>Figure 40: Frame Reinforcement</i>	37

<i>Figure 41: Housing Reinforcement</i>	38
<i>Figure 42: Caster Wheel Mounts</i>	39
<i>Figure 43: Electrical Enclosures Mounted on the Coop</i>	40
<i>Figure 44: Electrical Component Layout within the Enclosure</i>	41
<i>Figure 45: Interior of the Battery and Solar Controller Enclosure</i>	42
<i>Figure 46: Electrical Conduit Routed on the Side of the Coop</i>	43
<i>Table 4: Coop Decision Matrix</i>	43
<i>Figure 47: Electrical Power Subsystem Diagram</i>	44
<i>Table 5: Autonomous Chicken Coop Power Budget</i>	45
<i>Figure 48: FULLRIVER DC35-12 Battery [29]</i>	46
<i>Figure 49: Renogy 100W Monocrystalline Solar Panel Mounted on the Autonomous Chicken Coop</i>	47
<i>Figure 50: Renogy Wanderer Solar Controller Complete with Solar Panel and Battery Hookup</i>	48
<i>Figures 51 & 52: Screenshots from the Renogy App</i>	49
<i>Table 6: Constraint and Requirement Results</i>	50

1.0 INTRODUCTION

Just like any other pet or animal a person may own; chickens require routine care. This means that farmers and homesteaders must set aside time out of their day to go out to the chicken coop to tend to the needs of their birds. For household pets, such as cats or dogs, there are many automated systems on the market that handle routine care including feeders, waterers, and litter boxes. However, there are few systems like these that are designed for chickens, and they often provide a very limited number of features. Therefore, the design of a self-sustaining chicken coop that automates some of the simple daily tasks associated with caring for chickens will address these issues and offer a solution to farmers who desire a freer schedule.

2.0 BACKGROUND

2.1 STATEMENT OF PROBLEM

There are many tasks involved with raising chickens that farmers and homesteaders must be mindful of. These include maintaining an appropriate supply of feed and water and regulating the temperature of the coop during extreme outdoor temperature conditions. Additionally, it is important to move the coop daily to prevent the birds from getting parasites and supplying them with nutrient-rich soil. These tasks can be time consuming and cumbersome, and therefore it may be beneficial for farmers and homesteaders to take advantage of coops that automate these tasks.

Most automated chicken coops that are currently available for purchase are designed for the commercial market and are therefore costly and larger in scale than what is desired of homesteaders and hobby farmers. The chicken coops that do meet these criteria do not possess a high-level of automation and are often only equipped with an automated door that will close the coop during nighttime hours. As a result, it is necessary to consider the design of a coop that will provide more advanced automated features while maintaining a low price point and smaller size.

The overall goal is to design and construct a chicken coop that automates daily tasks, such as watering, feeding, monitoring, and the moving of the coop, and thus minimizing the amount of work the consumer must do to maintain the well-being of the chickens. To

accomplish this, an automated chicken coop will be designed to follow a pre-determined route set by the operator. Additionally, it will include a feeding, watering, and monitoring system.

2.2 REVIEW OF EXISTING SOLUTIONS

While similar projects have been conducted, few have addressed all the chores faced by chicken farmers as comprehensively as this one. Below are several previous projects, followed by discussions.



Figure 1: Camera Project [1]

One such project aimed to ease the burden on farmers by automating the care processes for chickens. “The previous research was conducted to monitor the locomotion behavior of broiler chickens in the cage. Overall, using a monitoring system increases farmer time efficiency and maintenance costs” [1]. This project enabled remote care for chickens, marking a significant upgrade for farmers who otherwise must attend to their chickens manually. However, it does not fully automate the care process, as users must remember to check the system daily. Moreover, remote care may introduce technical difficulties that could hinder or prevent task completion.

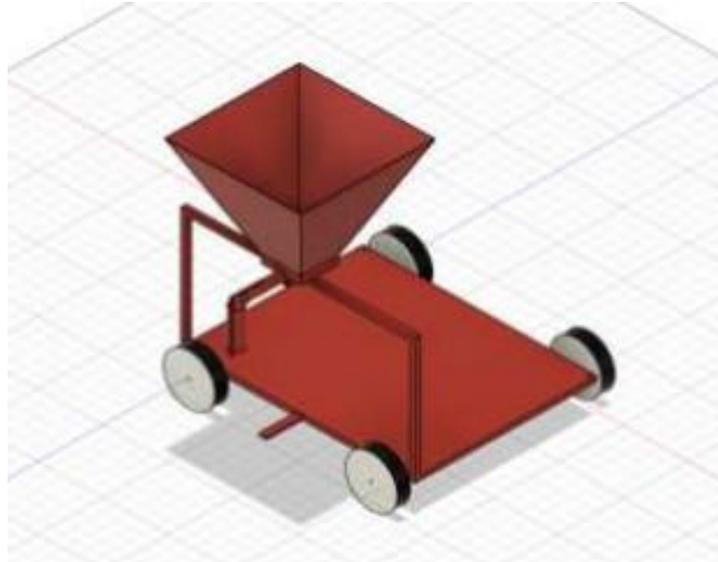


Figure 2: Automatic Feeder [2]

Another project focused on the need for an automatic feeder for chickens, which is also a component that will be included in this project. This feeder is not stationary; it moves within the barn or coop using infrared (IR) sensors. It distributes food evenly and alerts users when feed levels are low. "This feeder is operated by moving around the barn by IR sensors installed in the feeder system. Therefore, the proposed method is more efficient than the manual conventional way of feeding because less effort will be needed in feeding the chickens and less feed will be wasted" [2]. This project will adopt the concept of automatic feeding but design it to be stationary, with a larger food storage capacity. It will also feature an automatic watering system.



Figures 3 & 4: Sensor Detection Project [3]

Additionally, another project employed sensors to automatically supply food and water to chickens and regulate the coop temperature. "The prototype development involved placing the main components in the chicken coop such as the temperature sensor to detect the overall temperature, the ultrasonic sensor placed inside the food container, and the water level sensor inside the water container. There was also another ultrasonic sensor that was placed inside the food container to alert users if the food level is low" [3]. These features can be monitored via a smartphone app, providing valuable insights for this project, particularly regarding the auto-relocating system.



Figure 5: Automatic Barn [4]

The company UKKO Robotics has developed three different ROVA|Barn designs, serving as an example of the team’s project goals. This is a large-scale version of an automated chicken coop, that costs \$15,000 and holds 100 chickens. Their barns feature several operational factors that benefit chicken welfare, including “automatic feeders and watering systems, intelligent ventilation systems, predator-proofing electrified skirts, self-moving capabilities, solar-powered operations, and real-time data (live camera view and temperature controls)” [4]. UKKO Robotics illustrates the benefits of a mobile chicken coop, which aligns with this project’s objectives. Their innovations provide a baseline for the self-moving capabilities the team aims to incorporate.

In summary, while numerous projects have addressed aspects of chicken coops, few have designed entirely automated systems with an auto-relocating feature. It is believed that successfully implementing this feature could significantly enhance the functionality and luxury of automated chicken coops, which is the primary focus of this project.

2.3 PROJECT OBJECTIVE STATEMENT

The objective of this project is to design and construct a small-scale automated chicken coop system by April 2025 and within a budget of \$1000. The system will autonomously

manage relocation, along with feeding and watering, ensuring optimal living conditions for the chickens with minimal human intervention. To further optimize the coop, the electrical components will be powered via a battery and solar panel, and a camera will be implemented for monitoring conditions inside of the coop. The design will prioritize cost-efficiency and eco-friendliness. Deliverables include a formal report, poster, presentation, and a proof-of-concept prototype.

3.0 CONCEPTUAL DESIGN

3.1 REQUIREMENT SPECIFICATIONS

To meet the objectives of this project, several key requirements were established and shall be met for it to be successful. First, the coop should accommodate up to two chickens to provide enough space for comfort while maintaining a manageable size for automation. Second, the coop should move at a safe speed of no more than 1 foot per second, allowing it to navigate without causing harm to the chickens or the environment. Third, the automated system must maintain sufficient battery charge that lasts at least a week. The battery charge will have assistance from a solar panel. Additionally, the system shall be able to perform in chickens' ideal living temperatures of chickens, or 40-85 degrees Fahrenheit [5]. This means that all the components must function properly in these temperatures. Lastly, the entire coop must not exceed 150 pounds of weight. This is important because the motors must be able to pull the weight of the coop with no issue. Together, these requirements form a foundation for creating a safe, functional, and sustainable automated chicken coop.

3.2 CONSTRAINTS

To ensure the project is both practical and successful, a few constraints were identified and integrated into the design:

- Have automated features of movement, and feed/water level monitoring
- Provide information to the user about battery and solar panel status, and have live camera feed the user can access
- Be eco-friendly by use of solar power
- Be cost efficient by being under \$1000 to produce
- Minimize human intervention

- Can navigate semi-flat, well-kept yard terrain efficiently
- Withstand outdoor environment (temperature range, humidity, rain, etc.)

An automated chicken coop should incorporate several key features to optimize functionality and efficiency. A user-friendly and programmable interface is essential, allowing operators with varying levels of technical expertise to easily manage feeding and watering. Eco-friendly design is equally important, minimizing energy consumption and environmental impact through sustainable materials and renewable energy sources. Budget-friendly construction ensures accessibility for a wide range of users, from small-scale hobbyists to larger commercial operations. By minimizing manual labor through automation of essential tasks such as feeding, watering, and waste management, the system reduces human involvement while maintaining consistent care for the chickens. Additionally, the ability to navigate terrain effectively is crucial if the coop is designed to move, enhancing flexibility in managing land and grazing areas. These features contribute to a system that is both practical and sustainable, benefiting users and the environment alike.

3.3 SPECIFIC AIMS

The specific aims for this project were devised prior to its design in order maximize the automation and practicality of the final product for the end user.

3.3.1 AUTOMATED MOVEMENT

The design of the automated movement for the coop needed to be either AI generated utilizing GPS to navigate the user's property or set to follow three different predesigned paths that have been provided to the user.

3.3.2 CHICKEN'S WELL-BEING

The chicken's well-being was another essential factor to be considered in the design of this project. It was decided that this would be handled through the implementation of an automatic feeder and water to provide the chickens with the appropriate level of sustenance, along with cameras to monitor the conditions both inside and outside of the coop. The cameras would also serve as an alert for the user of any environmental hazards within the coop to reduce stress on the chickens. The automated movement also ties into the chicken's well-being by

allowing the chickens to engage in natural behaviors like pecking, scratching, and foraging for insects [6].

3.3.3 FOOD/WATER DISPLAY

Lastly, a food/water display was a function deemed necessary for informing the user of the amount of feed and water available to the chickens. This display was to be mounted onto the coop in a user accessible waterproof container. The data provided to the user would be how long the food and water will last be based on the number of chickens in the coop.

3.4 PERFORMANCE MEASURES

To evaluate the system's effectiveness, specific performance measures have been defined:

- Be at least 8 square feet in area, or four-square feet per chicken [7]
- The coop moves no greater than 1 foot per second
- Battery lasts a weeklong period and plugs into charger at the end of week
- Notifies user when sensors sense that food/water storage reaches a certain level, or if it gets stuck and cannot move
- The maximum weight of coop is only 150 lb.

These performance measures are based on the system's requirements. It can be known that a requirement is satisfied if its corresponding performance measure yields the desired result. Therefore, if the final design of the automated chicken coop has at least 12 square feet of space, moves at a speed of no greater than 1 foot per second, has a battery that runs the system for at least one week without charging, notifies the user when the feed/water level reaches a certain level or the coop itself gets stuck and cannot move, and weighs 150 pounds or less, all of these requirements can be deemed satisfied.

3.5 COOP FRAMEWORK

This is the fully assembled automated chicken coop, which integrates both structural and technological components for enhanced functionality and ease of maintenance. The images below show the coop from two different angles, highlighting a few key features. Future details regarding the design, components, and operation of the coop will be discussed throughout the remainder of the paper.



Figure 6: Back Side of Coop



Figure 7: Front Side of Coop

4.0 SYSTEM DESIGN

4.1 SYSTEM HIERARCHY

The autonomous chicken coop consists of five key systems designed for automation, security, and sustainability. The movement system allows mobility using wheels, axles, and motors. The food container system manages automated feeding and watering with sensors and an LCD screen for displaying the sensor information. The monitoring system includes a camera and live feed for remote observation. The housing/environment system ensures structural integrity with a motorized drawbridge door, reinforcement to make coop sturdier and for mounting electrical components. Finally, the electrical power system supports operations with a battery, solar panel, and status monitoring for battery level. These systems work together to create an efficient and self-sustaining chicken coop.

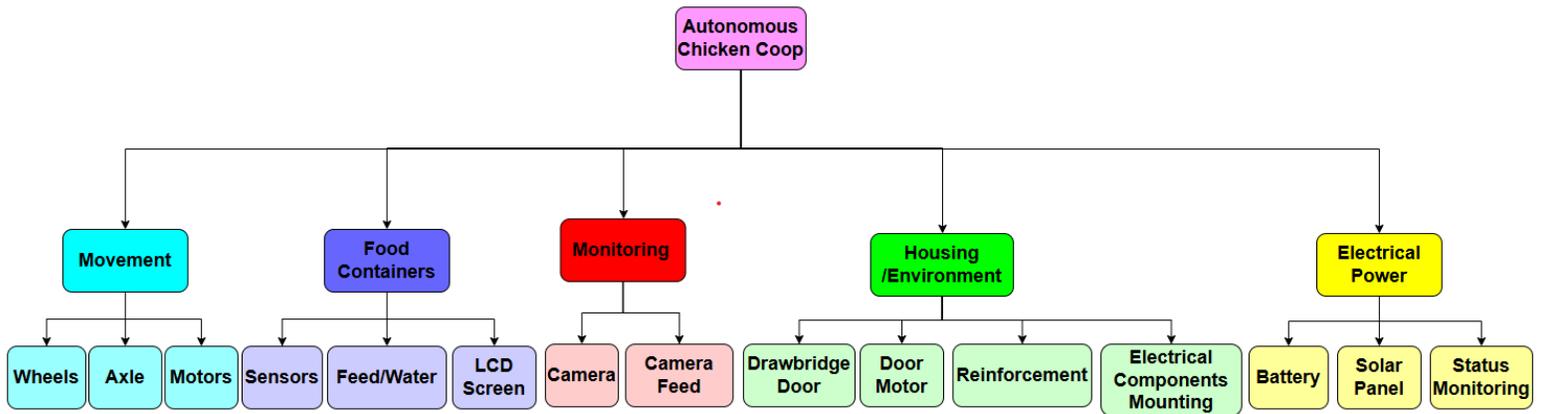


Figure 8: Automated Chicken Coop System-Breakdown

4.2 MOVEMENT

The movement system of the Autonomous Chicken Coop enables mobility through the integration of wheels, axles, and motors. The wheels provide the primary means of movement, designed to handle average yard terrain. They must be durable enough to support the coop's weight while ensuring stability and ease of transport. The axle connects the wheels and provides structural support, allowing synchronized movement and maintaining balance during relocation. Its material and strength are crucial to ensure long-term reliability. The motors automate movement, eliminating the need for manual effort. This system allows the coop to be

repositioned efficiently, whether for better weather conditions, pasture rotation, or environmental adjustments.

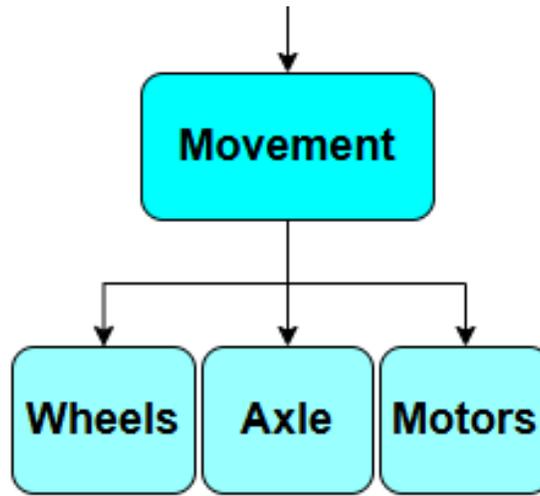


Figure 9: Movement Subsystem-Breakdown

4.2.1 MOVEMENT IMPLEMENTATION

In the design phase, several implementations were discussed, including complex steering patterns and possibly integrating AI algorithms. However, to minimize system complexity and time constraints it was reduced to simple linear movements programmed by the Arduino Mega 2560. This approach was chosen because chickens naturally return to the coop each night, allowing the system to close the door and then proceed with movement.

The door would be closed based on the time that chickens are expected to go in every night [8]. Once the door is closed, it is planned to start moving to the next location based on the width of the coop of 20 inches. It was decided that at midnight every night, the chicken coop would move forward and stop. The timer would then reset and wait till the next night to move again. This program is to repeat however many times the user decides. For example, it could be set to move forward every day for a week and then once completed the next week move backward till it reaches its original location. Limiting the movement to straight-line forward and backward travel eliminates the need for steering corrections, obstacle avoidance, and complex localization. This makes the system overall more robust and easier to maintain. The timing, speed control, and interval counting are managed entirely by the microcontroller through a pre-programmed routine within the Arduino.

4.2.1.1 WHEELS

The chicken coop features a redesigned caster system to improve mobility and reduce overall weight. Originally, the coop was equipped with heavy-duty steel casters with solid, rugged tires, as shown in the first image. While durable, these wheels added considerable weight to the structure. To address this, the team replaced them with lighter plastic casters, shown in the second picture, which significantly reduced the overall load while still providing sufficient strength and stability for movement. This updated caster system makes the coop easier to move without compromising functionality or durability.



Figures 10 & 11: Caster Wheels

The wheels used on the coop are 8-inch-semi-solid plastic caster wheels with polypropylene hubs [9]. Polypropylene is a durable, lightweight thermoplastic known for its resistance to impact and moisture, making it ideal for outdoor environments. These hubs provide a strong yet weather-resistance core that can withstand exposure to the elements without corroding or degrading over time.

The outer tire is made of semi-solid rubber, which offers a balance between firmness and shock absorption. These semi-solid variants are puncture-proof and require no maintenance in terms of air pressure, ensuring consistent performance across smooth yard terrain.

To integrate these wheels into the coop's motorized system, the inner hub was notched to interface directly with the drive axle. This modification allows the axle to lock into the hub, so when the motor turns the axle, the wheels rotate simultaneously. This design provides reliable

propulsion and maneuverability while maintaining the simplicity and robustness needed for long-term outdoor operation.



Figure 12: Plastic Caster Wheels [9]

Additionally, a smaller caster wheel was installed under the ramp, as shown in Figure 13 below. This ramp caster supports the weight of the ramp during relocation, preventing it from dragging or catching on the ground and improving maneuverability.

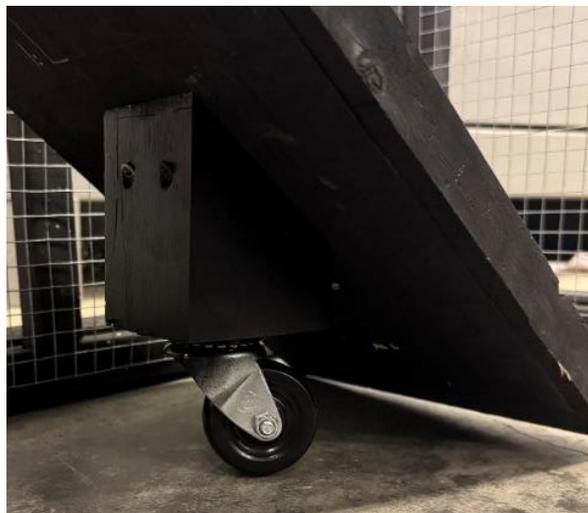


Figure 13: Ramp Caster

4.2.1.2 AXLES

An initial problem encountered by the team was that it was difficult to find an axle that could be adapted to the motor and the wheel, which meant that the motor shafts could spin but the wheels would not. Therefore, a custom axle that could be bolted onto the motor hubs was created on SolidWorks. The axle has two tabs on the shaft directly under the head. Two notches were cut into each motorized wheel so that the axle could slide into the wheel, and the tabs could interlock into the notches. This would allow the wheels to spin whenever the motors run. The SolidWorks model was 3D printed and tested, and the geometry was as desired, so a steel version of the same axle was fabricated for more strength and durability.



Figure 14 & 15: SolidWorks 3D printed model (left) and final steel model (right)

As mentioned previously, a motor hub was bolted to the head of the steel axles by using pre-drilled holes on the bottom of the head. This made the axles compatible with the motor shafts. In Figure 16, the motor hub can be seen attached to the head of the axle.



Figure 16: Motor hub attached to axle

With the motor shaft successfully adapted to the axle, the wheel could be placed onto the axle and engaged with the notches of the wheel (Figure 17). This allowed the motor to drive the wheel. This was the design for both motorized wheels.



Figure 17: Axle and motor hub paired with motor shaft

A hole was drilled into the final steel model so that a lock pin could be inserted into it to prevent the wheel from moving back and forth on it and keep the tabs of the axle engaged with the notches of the wheel, as shown in Figure 18.



Figure 18: Lock pin on wheel

4.2.1.3 MOTORS

Table 1 shows a structured evaluation of the three drive motor considerations for the project. The brands considered were Pololu, Banebot, and Midwest Motion. This table represents a weighted decision matrix in which criteria such as the unit price, system compatibility, and connectivity were considered. Each of the considered categories was assigned a specific “importance factor” (the absolute value of the total of which add up to 1) reflecting its importance in the decision of which component would be selected. The “Total Weight” column represents the overall scores of each component. Here, the values were calculated by multiplying the numbers found for each component in a specific category by the importance factor of that category and then summing the result of each category together.

Table 1: Motors Decision Matrix

Part:	Price Per Unit:	Compatibility	Connectivity	Total Weight
Importance	- 0.1	0.4	0.5	
Pololu	\$51.95	Yes (100)	Yes (100)	80.4
Banebot	\$99.99	Yes (100)	Yes (100)	80.0
Midwest Motion	\$125	No (0)	Yes (100)	37.5

The Pololu 12V 37D Metal Gearmotor was selected because it was overall the best-balanced motor. The performance, compact size, and energy efficiency were a few reasons it was selected. Another reason is that Pololu is compatible with the Arduino libraries, which is the selected microcontroller for this project. The stall torque is approximately 49 kg-cm (4.81 N*m) and a manageable stall current of 5.5 Amps [10]. The coop is expected to traverse over mildly inclined terrain which these motors should be able to perform with minimal issues.



Figure 19: Wheels Motor [10]

Some calculations were performed to see how the motor would stand up to the required torque needed. The motors need to provide enough torque to pull the weight of the coop through grass. Each wheel motor provides 4.8 Newton-meters, and there are two motors, so the motors combine to provide 9.6 Newton-meters of torque. Since the motors will be driving the coop through grass, a coefficient of rolling resistance of 0.1 can be used [11]. The wheels are each 8 inches in diameter, or 4 inches in radius. For the motors to be successful, the required torque to move the coop must be less than what the motors can provide. Here is the analysis to find the torque needed. Each calculation was converted to the same unit to ensure fluency. The estimated weight of the coop was converted from pounds to kilograms to align with the rest of the calculation. This estimated 150 lbs. was the target to stay under for this project.

$$W = \text{weight of coop} = 150 \text{ lbs}$$

The calculation below takes the weight in pounds and converts it into newtons, which is the unit that is required for the rolling force.

$$W = 150 \text{ lbs} \times \left(4.482 \frac{N}{lb}\right) = 667 \text{ N}$$

The equation below calculates the rolling force needed from the motors which the 0.3 is a rolling coefficient in dry grass.

$$\text{Rolling force needed from motors} = F_R = \mu W = (0.3)667 = \mathbf{200.1 \text{ N}}$$

The equation below takes the rolling force needed and applies it to the radius of the wheel which now provides the torque needed from the Pololu.

$$\begin{aligned} \text{Torque needed from motors} &= F_R \times \text{radius of wheel} = 200.1 \text{ N} \times 0.1 \text{ m} \\ &= \mathbf{20.01 \text{ N} \cdot \text{m}} \end{aligned}$$

$$\mathbf{\underline{20.01 \text{ N} \cdot \text{m} \geq 9.6 \text{ N} \cdot \text{m}}}$$

When originally calculated the motors would have worked but after recalculating it was noted that the rolling force was calculated incorrectly which in turns causes an issue with the selected motors. This was a miscalculation which in turns makes sense of why the coop struggled to move in the grass. The coop does move efficiently on the concrete because the rolling force on concrete is smaller allowing the torque required to be less.

While the Banebot PDX16 would provide more robust torque and durability for heavier applications compared to the Pololu, for the use of only using two motors with each motors' stall torque providing 91 in-lb (10.28 N*m) was probably the safer route [12]. With trying to keep the price down the Pololu beats the Banebots. While this would be the better choice for moving the coop, it is expensive, and the compatible motor controller is also expensive.

The Midwest Motion DC Gearmotor offered the highest torque output and is heavy-duty. These motors were the most expensive consideration at 250 dollars, which is far more than what was budgeted for the project's motors [13]. The motors would be able to handle heavier weight and steeper inclines. In this aspect the motors were powered for what was needed for the coop.

There were also not many applications showcased online for this version of the motor in use, which was concern enough to evoke consideration of other options.

4.2.2.1 MOTOR HOUSING

It is important that the motors used to drive the wheels are protected in some way. If left exposed, dirt, dust, or moisture can accumulate in the motors and hurt performance and longevity. In order to solve the problem of the motors being exposed a motor housing was modeled on SolidWorks and 3D printed to make a snug fit on the coop and cover up the motors. A little semi-circle shaped hole was left in the rear side of the housing to allow electrical conduit to be run into it, and a section was left open so the motor shaft could stick out without interfering with the housing.



Figure 20: Motor housing enclosure

4.2.2.2 MICROCONTROLLER

Table 2 shows an evaluation of the three microcontroller considerations for the project. The brands considered were Arduino Mega, Elegoo Mega, and Arduino Giga. This table represents a weighted decision matrix in which criteria such as the unit price, digital Inputs/outputs, power, and accessibility were considered. Each of the considered categories was assigned a specific “importance factor” (the absolute value of the total of which add up to 1)

reflecting its importance in the decision of which component would be selected. The “Total Weight” column represents the overall scores of each component. Here, the values were calculated by multiplying the numbers found for each component in a specific category by the importance factor of that category and then summing the result of each category together.

Table 2: Microcontroller Decision Matrix2

Part:	Price Per Unit:	Digital Input/Outputs	Power	Accessibility	Total Weight
Importance	0.1	0.2	0.3	0.4	
Arduino Mega 2560	\$48.40	Yes(100)	Yes(100)	Yes(100)	94.84
Elegoo Mega 2560	\$21.99	Yes(100)	Yes(100)	No(0)	52.20
Arduino GIGA R1	\$72.82	Yes(100)	Yes(100)	Yes(100)	97.28

The Arduino Mega 2560 has 54 digital input/output pins, 16 analog inputs, and 4 UART (Hardware serial ports) which are needed for a complex project like this one [14]. The Arduino Mega has a widely supported ecosystem with extensive documentation, a large community of developers, and compatibility with a wide range of shields and libraries. This in turn makes it easier to find resources, tutorials, and troubleshooting support when working on a project. While the Elegoo is like the Mega it is an off-brand competitor which could increase issues with library communication. Finally, the Mega is in the median price range of the options.

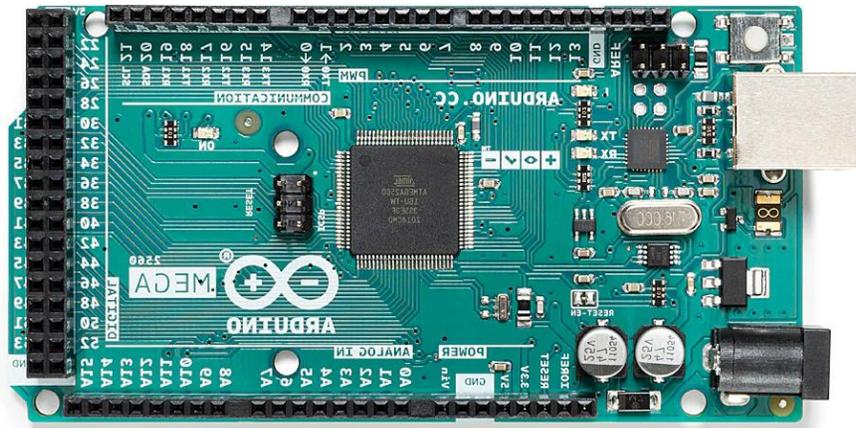


Figure 21: Arduino Mega 2560 [14]

The Elegoo Mega is the same as the Arduino Mega being a little cheaper and capable with the Arduino IDE which was a requirement for the project. The Arduino IDE software was something that several group members were familiar with. However, since this is a secondary company, this microcontroller has several concerns with lower quality components which could in turn affect the project in the long term.

The Arduino Mega is a well-developed board with several resources backing its functionality. The Mega has 54 digital and 16 analog pins, which is more than needed for this project. This microcontroller is reliable and has extensive community support which was a requirement for the project. The Arduino Mega is more expensive than the clone board Elegoo, however has the curial requirements needed for the project.

The Arduino Giga R1 is a more advanced microcontroller compared to the Mega. The Giga also has Wi-Fi and Bluetooth built in, which would be a nice aspect for this project [15]. However, the selected ESP32 Camera board with built in the Wi-Fi and Bluetooth capabilities, so the Wi-Fi capabilities were not needed from this Arduino Giga R1. This kept the price down as well as the Arduino Giga R1 costs more than the Mega which was selected for the price and similar capabilities between the boards researched.

4.2.2.3 MOTOR DRIVER

Table 3 displays an evaluation of the three motor driver considerations for the project. The brands considered were BTS7960, L298N, and VNH5019. This table represents a weighted decision matrix in which criteria such as the unit price, system compatibility, power, and current rating were considered. Each of the considered categories was assigned a specific “importance factor” (the absolute value of the total of which add up to 1) reflecting its importance in the decision of which component would be selected. The “Total Weight” column represents the overall scores of each component. Here, the values were calculated by multiplying the numbers found for each component in a specific category by the importance factor of that category and then summing the result of each category together.

Table 3: Motor Driver Decision Matrix

Part:	Price Per Unit:	Compatibility	Power	Current	Total Weight
Importance	0.1	0.2	0.3	0.4	
BTS7960	\$21.99	Yes(100)	High(100)	43A	69.40
L298N	\$11.49	Yes(100)	Medium(50)	4A	37.75
VNH5019	\$29.95	Yes(100)	Low(0)	30A	35.00

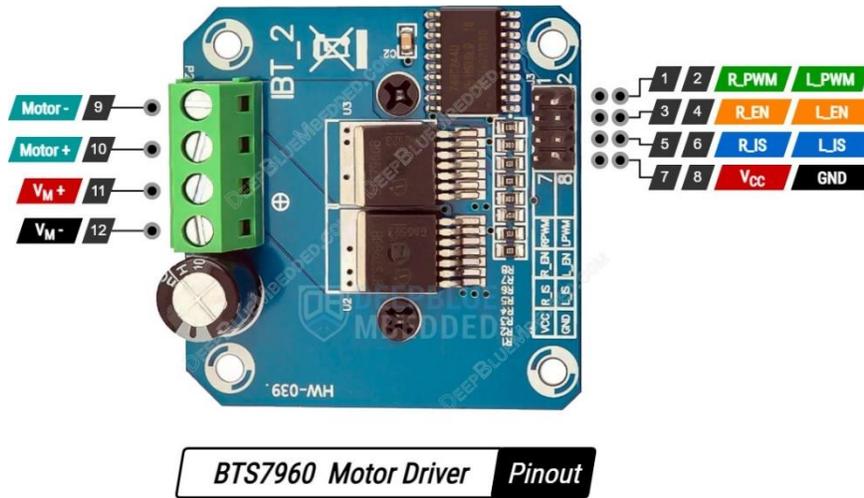


Figure 22: BTS7960 Motor Driver [16]

HiLetgo 4pcs L298N Motor Driver Controller – The L298N is a basic dual H-bridge motor driver modular widely used for low-power robotic projects which would be good if a big motor was not needed. The L298N was researched for its low costs and well-documented libraries suitable for beginners. However, it was determined that each channel can only handle up to 2A of continuous current which is not enough to support the Pololu Motors [17].

BTS7960 43A H-bridge Double High-Power Motor Driver – The BTS7960 is a powerful motor driver capable of handling high-current motors, which makes it ideal for this application. The datasheet says it can handle up to 43A per channel which is more than what is needed for the Pololu Motors, and BTS7960 also has built-in overcurrent and thermal protection [16]. To ensure safe and reliable power delivery between the power source, motor driver, and motors, the wiring was selected based in accordance with UL 1007 [18]. Given the Pololu motor’s stall current of 5.5A, 18 AWG AWM-style wire was chosen, as it meets the requirements outlined in both UL 1007 and ASTM B258-18 [18], which defines the dimensional standards for round copper

VNH5019 Motor Driver Carrier – The VNH5019 is a versatile motor driver that supports medium-power applications and is popular in robotics. The VNH5019 would be able to control the Pololu Motors but under certain conditions it would struggle. For example, for high-stall scenarios a higher-current motor driver is needed. So, because of this and the price being higher, the BTS7960 was selected over the VNH5019.

4.3 CONTAINERS

The food containers systems in the Autonomous Chicken Coop are designed to automate and monitor the feeding and watering process. It includes sensors, a feed/water mechanism, and an LCD screen. The sensors are used to detect the levels of food and water within the containers, ensuring that the chickens always have access to adequate resources. These sensors will display the level of food and water on the LCD screen. The LCD screen provides a user-friendly interface that displays real-time information such as the current levels of the food and water. This system helps maintain consistent care for the chickens with minimal intervention.

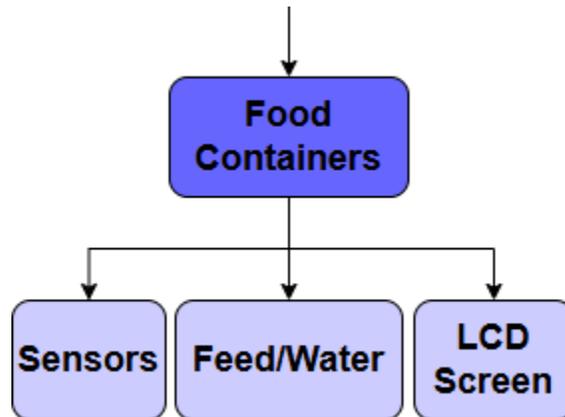


Figure 23: Food Containers Subsystem-Breakdown

4.3.1 FEED & WATER IMPLEMENTATION



Figure 24 & 25: Water and Feed Containers

The chicken feed and water system shown in the pictures above is a simple and efficient setup designed to consistently supply feed and water to backyard or small-scale poultry flocks. The design features a vertical PVC pipe for feed storage, and a water delivery system that includes a PVC pipe connected to a larger storage container mounted on top for increased capacity. The water system uses red chicken water cups, which release water when pecked, ensuring a steady supply with minimal waste. This gravity-fed system is easy to refill and maintain, making it an ideal low-effort solution for effective flock management.

Using the volume equations for both the food and water components, it's clear that the system can hold enough resources to support a chicken for multiple days. Chickens typically consume a consistent amount of water, ranging from 8 to 16 ounces daily [20]. The average feed consumption is about a quarter of a pound per day. However, daily consumption of both water and food can vary depending on the chicken's age, the season, and environmental conditions. Based on the calculated capacities, this setup provides approximately 2 to 3 days' worth of water and 2-3 days' worth of feed, ensuring reliable access to essential nutrients even without daily supervision [21].

Food Volume Equation

$$V = \pi r^2 h$$

$$V = \pi(1.5in)(20in)$$

$$V = 141in^2 = 9.8 \text{ cups}$$

$$9.8 \text{ cups} \times \left(\frac{2oz}{cup}\right) = 19.6 \text{ oz}$$

$$19.6 * \frac{day}{8oz}$$

2 – 3 day supply

Water Volume Equation

$$V = \pi r^2 h$$

$$V = \pi(.25in)^2(12in)$$

$$V = 2.357in^2$$

$$\text{Density of water} = 0.578 \frac{oz}{in^3}$$

$$0.578 \frac{oz}{in^3} = \frac{mass(oz)}{2.357in^3}$$

$$\text{water in pipe} = 1.36oz$$

$$\text{water in bottle} = 30oz$$

$$\text{water total} = 31.36oz$$

2 – 3 day supply

4.3.1.1 SENSORS

For this project it was decided that a food/water level monitoring system should be implemented in the final design of the project. The food and water containers were mounted, and the volume calculations were done above in the previous section. This was important for the

users to have an idea of how long each containment last before going empty. To further notify the users, the implementations of an ultrasonic distance sensor and a contact level sensor were adapted to the containers. The ultrasonic distance sensor determines the distances to a target by measuring time lapses between sending and receiving of the ultrasonic pulse [22]. All the integration is done inside of the module which is then sent to the microcontroller. This simply uses the trigger pin to trigger ultrasonic sound pulse, by setting the pin to high which initiates the ultrasonic burst out [23]. The Echo pin is set to high and remains high until the sensor receives an echo in which it returns to low. Finally, the measured time in which the echo pin stayed high is then calculated into distance [24]. This calculated distance is then compared to the length of the container which then decides how full it is based on the length of the container.



Figure 26: Ultrasonic Feeder Sensor [24]

For the next sensor selected for this project a contact water/ liquid level sensor which reads a 0 in the code when no liquid is detected and reads a 1 when liquid is detected. This sensor was mounted closer to the bottom of the container to read full until it does not detect water which would change to reading close to empty for the user to read. This would give the users enough time to refill before the container goes to empty. The selected sensor is compatible with the Arduino Mega and provides a base code that was used to determine the final code. This sensor is resistant to high temperatures and high pressures, which is more than we need for this project [25]. The selected sensor is in Figure 27.



Figure 27: Contact Water Sensor [25]

4.2.2.2 LCD SCREEN

An LCD screen was chosen to display the food and water levels after encountering challenges with implementing an HMI screen in the initial design. Due to the complexity and difficulty in coding the HMI, the team opted for the LCD as a more practical alternative. The selected Spark Fun display features a 16-character by 2-line interface with blank text on a black background with white lettering [26]. It requires 11 general-purpose I/O pins, which the Arduino Mega is fully capable of supporting. This substitution simplified the display system, eliminating the complications of the HMI and allowing the team to concentrate on the primary goal of the project – achieving motion. The LCD screen used is shown in Figure 28.



Figure 28: LCD Screen [26]

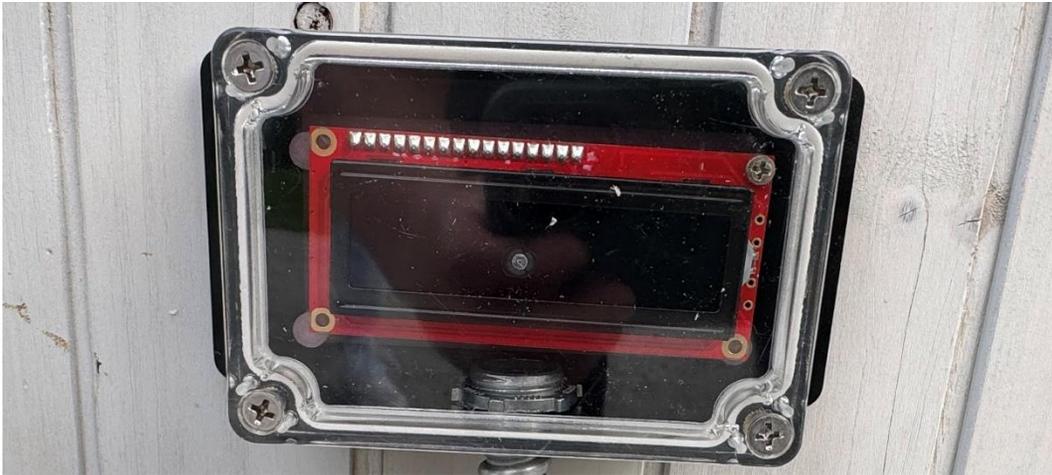


Figure 29: Mounted LCD Screen

4.4 MONITORING

The monitoring system in the Autonomous Chicken Coop is focused on providing real-time observation and enhancing security. It consists of a camera and live camera feed. The camera is installed within or around the coop to continuously capture video footage of the chickens and their environment. This helps in monitoring their behavior, detecting intrusions, and ensuring overall safety. The camera feed allows users to access live videos remotely through a URL link. This feature enables quick responses to any issues, such as equipment malfunctions and reduces the need for constant physical supervision.

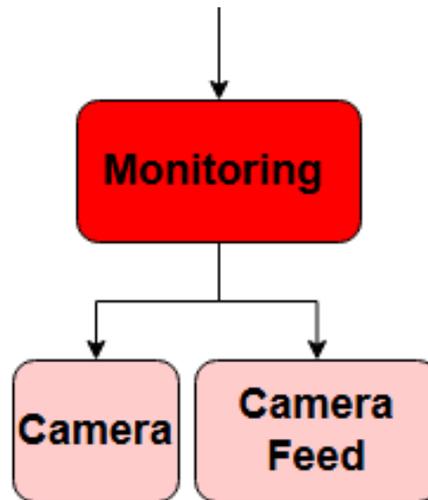


Figure 30: Monitoring

4.4.1 MONITORING IMPLEMENTATION

The implementation of the monitoring subsystem consisted of selecting a camera that could provide reliable access to a real-time video stream of the coop's interior and determining how that video stream would be transmitted and ultimately displayed to the user. This implementation presented its own set of unique constraints. First, the monitoring system, specifically the camera, had to reliably function inside of the coop under normal circumstances. This means the camera needed to be capable of capturing clear video regardless of the air quality or dust level that could be reasonably expected to occur in a housing environment of two chickens. Second, the live camera footage needed to be transmitted in such a way that it could be easily accessed by the user at any time. Finally, the monitoring system needed to be capable of providing the live video stream to the user whenever the coop was a reasonable distance away from the user's location when attempting to view the live stream.

4.4.4.1 CAMERA

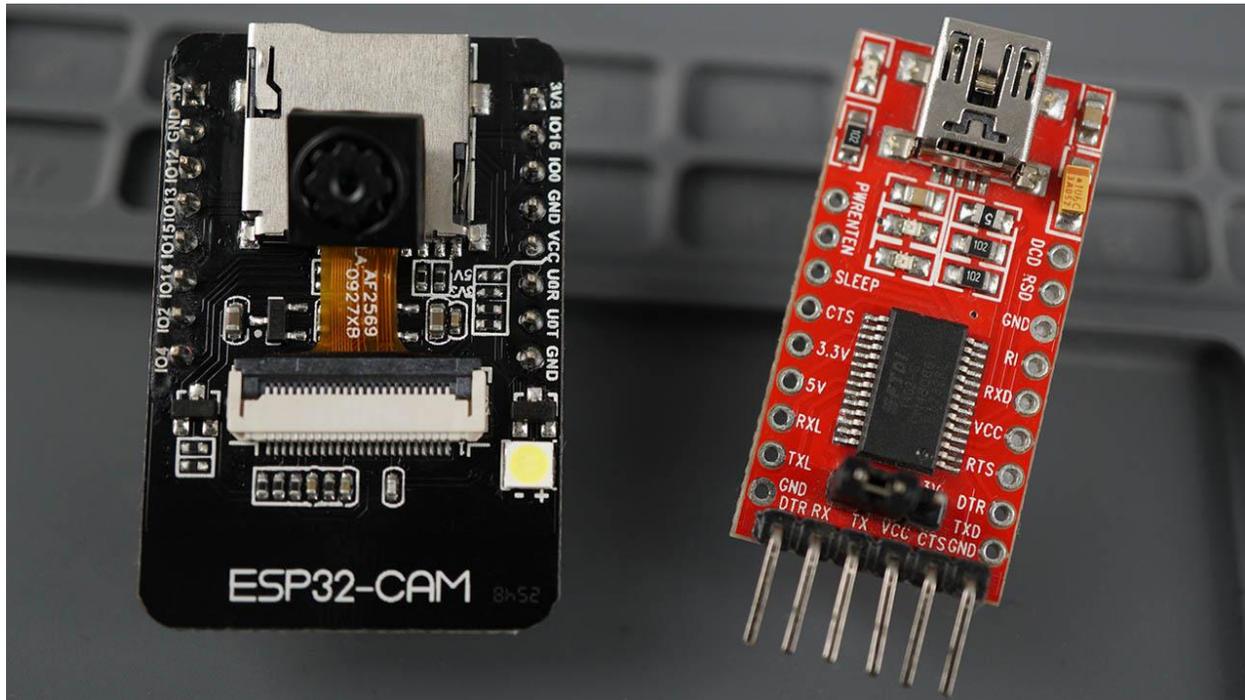


Figure 31: ESP32-CAM Board (with OV2640 camera) and FT23RL USB to TTL Serial Converter [27]

The camera setup chosen for the monitoring subsystem is an ESP32-CAM board paired with an OV2640 2-megapixel camera. The reason for this selection is that the ESP32 includes built-in Wi-Fi and Bluetooth modulation capabilities, making it easy for users to view the live video stream when connected to their home network. Additionally, the ESP32 is an open sourced device with several open-sourced documents available that has greatly assisted in its integration into this project. The ESP32-CAM complete with the OV2640 camera is compact, making it suitable for mounting in small spaces such as the inside of the coop. It also includes an LED on the front of the board that operates as a flash feature that can be switched on for dark areas. Lastly, the ESP32-CAM development board can be programmed using the Arduino IDE, which is the same application that was used to program the Arduino Mega for the other automated features of the coop.

For the physical integration of the camera onto the coop, an open-source design of an enclosure for the ESP32-CAM was 3D printed and mounted high enough on the inside of the coop to prevent the lens from being obstructed. Figure 32 below shows the ESP32-CAM inside

of its enclosure mounted on the inside of the coop. The enclosure ensures that the ESP32-CAM board is sealed from the environmental conditions inside of the coop.



Figure 32: Monitoring System Camera Mounted in the Autonomous Chicken Coop

4.4.4.2 CAMERA FEED

For the user to access the live camera feed, the ESP32-CAM must be programmed to include the user's service set identifier (SSID) and network password. Figure 33 shows the lines of code in Arduino IDE where the user must write in this information.

```
// =====  
// Enter your WiFi credentials  
// =====  
const char *ssid = "*****";  
const char *password = "*****";
```

Figure 33: ESP32-CAM User SSID and Network Password Input

Figure 34 below shows the ESP32-CAM board connected to the FT232RL USB to TTL Serial converter board during programming which was required to establish communication between the ESP32-CAM microcontroller and the user's computer.

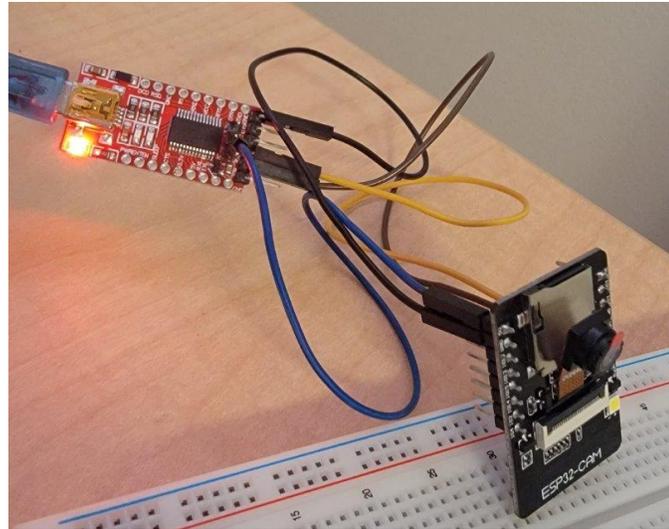


Figure 34: Programming the ESP32-CAM Using the FT232RL

Once the code containing the user's SSID and network password is uploaded to the ESP32-CAM microcontroller, a unique uniform resource locator (URL) will be provided to the user inside of the serial monitor window of the Arduino IDE. The user can input this URL into a web browser of their choice, and it will lead them to a web page where they can view the real-time video stream of the inside of the coop. A screenshot of this webpage is shown in Figure 35 and showcases what the camera feed will look like from the user's perspective. Note the settings on the right-hand side of the web page that can be used to adjust the video for optimized viewability based on the conditions inside of the coop. Once the user has the URL from the serial monitor, the ESP32-CAM board can be disconnected from the FT232RL and installed in the coop where it will run off a 5V DC supply.

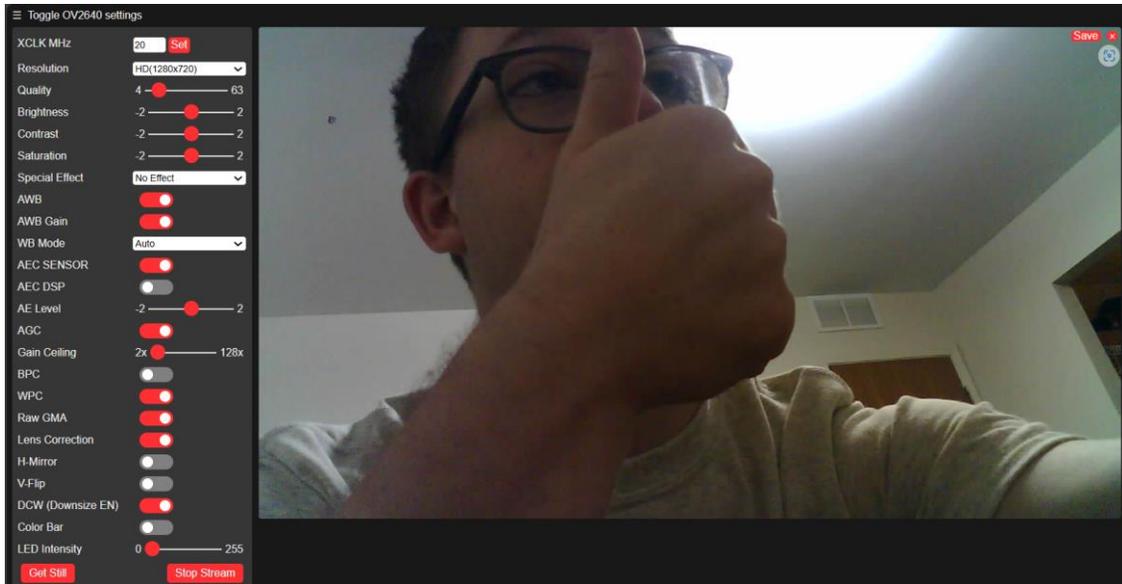


Figure 35: Monitoring System User Interface

4.5 HOUSING/ENVIRONMENT

The housing/environment system of the Autonomous Chicken Coop is designed to ensure a safe, functional, and adaptable living space for the chickens. It includes a drawbridge door, a door motor, reinforcement, and electrical component mounting. The drawbridge door provides controlled access to the coop, allowing chickens to enter or exit at scheduled times, while also protecting them from external threats when closed. The door motor automated this process, enabling timed operation without human intervention. Reinforcement strengthens the overall structure, making it more durable while moving through yard terrain. Finally, the reinforcements allowed a more secure mounting place for the electrical component boxes on the outside of the coop.

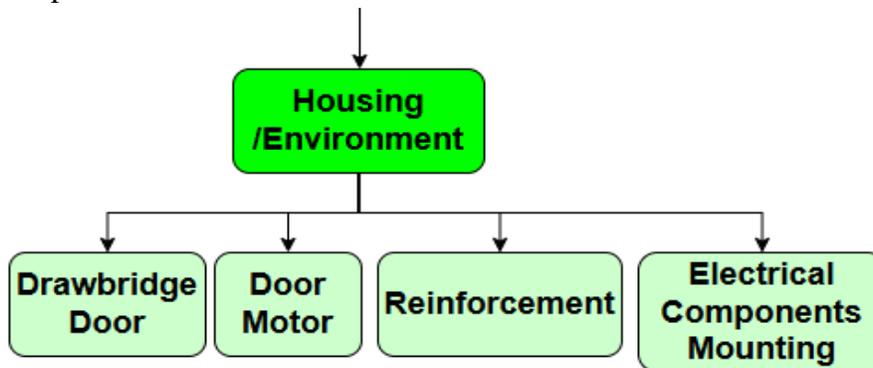


Figure 36: Housing/Environment Subsystem-Breakdown

4.5.1 HOUSING/ENVIRONMENT IMPLEMENTATION

The implementation of the housing or environment was important for the chickens' shelter and safety. First, the team added reinforcement to the coop to make the structure sturdier. Then, a drawbridge door system was added so that the chickens could walk into their housing up the door like a ramp, and then have the door shut at nighttime to keep them safe and secure. The drawbridge door was raised and lowered by the door motor, which runs on a timer to shut the door every night and open it every morning. Along with reinforcement, some more boards were mounted to the inside of the coop to allow a strong mounting spot for the electrical components. All of these things work together to make up the housing and living environment for the chickens and automate it as well.

4.5.1.1 DRAWBRIDGE DOOR

The figures below illustrate the initial and final designs of the drawbridge door system. The purpose of this system is to protect chickens from predators by automatically securing the coop after the chickens enter. Once inside, the door is programmed to close automatically, preventing predators from accessing the housing and keeping the chickens safely enclosed throughout the night.

The initial design featured two spools connected by a single rod, with the motor driving system using fishing line, as shown in Figure 37. In contrast, the final design uses a single spool, a rod with a custom adapter for direct coupling to the motor shaft, and a durable string to link the spool to the drawbridge door, as seen in Figure 38.

The door itself was mounted using two hinges attached to the bottom edge of the door and the coop frame, enabling vertical movement for opening and closing. The spool functions by winding the string to raise or lower the door as needed. To improve alignment with the spool mechanism, the motor assembly was placed on the exterior of the housing. Additionally, a custom 3D-printed enclosure was fabricated to shield the motor from environmental exposure.

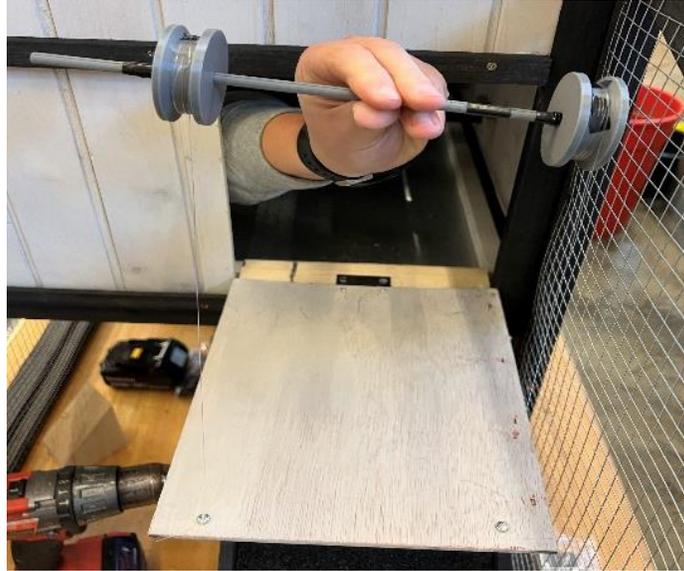


Figure 37: Initial Design of the Drawbridge Door



Figure 38: Final Design of the Drawbridge Door

4.5.1.2 DOOR MOTOR

The drawbridge door is operated by an Ocean 810:1 metal DC gear-reduction motor. This motor supports operation at both 12V and 24V, provides precise position and speed control, and

features a compact design. Its high gear ratio results in reduced rotational speed while delivering increased torque, making it well-suited for applications requiring strength over speed. Moreover, the motor takes approximately one minute to fully open or close the drawbridge door. Once closed, the motor is programmed to hold the door shut throughout the night (approximately 10 hours), then reopen it the following morning to allow the chickens to exit. This motor is very similar to the Pololu motor which was the other selected motor. The reason this motor was not used in movement design was because it is a secondary manufacturer which could result in cheaper internal parts which could lead to failure. While this concern was taken into consideration, knowing this could result in the door not operating as intended. However, in testing the motor performed as intended with little complications.



Figure 39: Drawbridge Door Motor [28]

4.5.1.3 REINFORCEMENT

When the coop was initially put together, the overall stability was a concern, as the frame felt feeble and wobbly. This is mainly due to the fact that the stock wood of the coop is meant to be light, which in turn means it is very weak. Since this coop will be moving automatically, and have other elements mounted to it, such as the feeder, solar panel, and electrical component boxes, it is very important that the structure is strong and sturdy. To fix this, 2x2 inch pieces of lumber were screwed into the sides flush with the bottom. This immediately gave the coop a

studies frame and overall structure. The pieces of lumber used here were spray painted black prior to attaching so they would match the black color of the coop (Figure 40).



Figure 40: Frame Reinforcement

Furthermore, two wood boards were mounted inside the coop's housing on the wall and fastened to the corners (Figure 41). This added even more stability, especially to the housing portion of the coop. The boards mounted to the inside wall also provided a perfect place to drill into and bolt on the electrical component boxes.



Figure 41: Housing Reinforcement

Since the 2x2 reinforcement boards also provided a strong place to mount components, the team created two caster wheel holders out of 2x4 lumber that extend out from the coop and allow the caster wheels to spin freely when mounted onto them (Figure 42). These were able to be screwed vertically into the 2x2s along the bottom of the coop walls and screwed horizontally into the coop's wooden frame. The caster wheel holders were also spray painted black to match the coop, similarly to the 2x2 reinforcement boards.



Figure 42: Caster Wheel Mounts

4.5.1.4 ELECTRICAL COMPONENT LAYOUT

Another important part of the implementation for the housing/environment subsystem consisted of how to preserve the sensitive electrical components that were critical to the autonomous functions of the coop. This included selecting adequate enclosures for the Arduino microcontroller, motor drivers, battery, and solar charge controller as well as how to route the electrical wires in such a manner that they would not be exposed to the outside conditions or the chicken's living environment. Figure 43 shows the two electrical enclosures selected for this project and their locations on the rear side of the chicken coop. Both enclosures meet the requirements set in place by the National Electrical Manufacturers Association's (NEMA) standard for electrical equipment enclosures (ANSI/NEMA 250-2003) with a type 4X rating. The type 4X rating ensures that both enclosures have a degree of protection against an ingress of water (rain, sleet, snow) [29].



Figure 43: Electrical Enclosures Mounted on the Coop

The upper enclosure housed the main electrical components such as the Arduino, motor drivers, and a step-down voltage regulator. The step-down voltage regulator is a Pololu S13V15F5 and is responsible for converting 12 VDC from the battery to 5 VDC to provide consistent power for the motor encoders, the motor controller boards, and the ESP32-CAM microcontroller. This was a necessary feature since the Arduino Mega's onboard 5 V supply did not provide enough current to power all these features at one time. Figure 44 shows the inside of the upper enclosure, and the electrical components mounted to a custom panel.

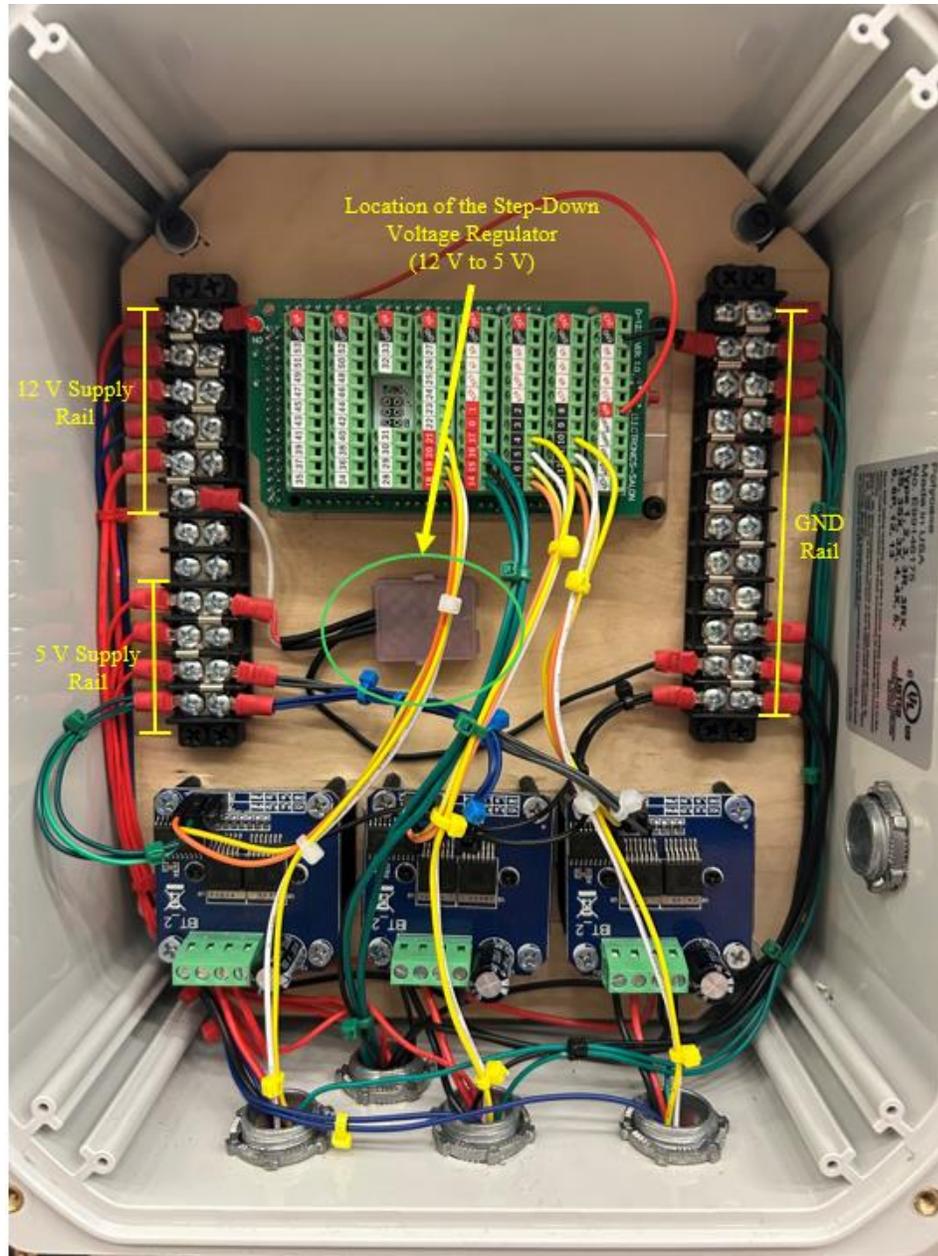


Figure 44: Electrical Component Layout within the Enclosure

The lower enclosure housed the battery and solar charge controller. This enclosure needed to be larger to accommodate the size of the battery. It was desired to have the battery and solar controller inside of a separate enclosure to make them more accessible to the user. A custom panel was created out of an 0.125" thick steel plate to secure the components. Figure 45 shows the inside of this enclosure along with the solar charge controller and battery.



Figure 45: Interior of the Battery and Solar Controller Enclosure

For securing wiring and keeping it shielded from the chickens inside of the coop and the outside environment, electrical conduit was routed to the various electrical components and secured to the structure of the coop. Figure 46 shows a section of the conduit going from the upper enclosure to the right-side drive motor.



Figure 46: Electrical Conduit Routed on the Side of the Coop

4.5.1.5 COOP

Table 44: Coop Decision Matrix

Brand/Model	Price Per unit:	Weight of Coop	Chicken Capacity	Total Weight
Importance	0.3	0.4	0.3	
Coop 1	\$149.00	56	2 to 3	25
Coop 2	\$125.99	60	1 to 4	23
Coop 3	\$100.00	68	3 to 4	23
Coop 4	\$139.99	56	2 to 4	23

To determine the most suitable coop for this project, a decision matrix was developed to evaluate key factors. The main considerations included the coop’s price, its total weight, and its capacity to accommodate the required number of chickens. Upon assessment, coop 1 was found to be the best choice as it offered the lowest weight and met the necessary housing capacity.

Although this option was not the cheapest one compared to the other ones, its positive quality reviews justified additional investment in the coop.

4.6 ELECTRICAL POWER

The electrical system of the Autonomous Chicken Coop is designed to provide reliable and renewable energy while ensuring system health and performance visibility. The solar panel serves as the primary energy source, converting sunlight into electricity during the day. This energy is regulated through a solar charge controller, which manages the voltage and current to safely charge the battery without the risk of overcharging or damaging it. The battery acts as an energy storage unit, supplying power when solar energy is unavailable, such as during the night or a cloudy day. The status monitoring is with the solar panel app allowing the user to look at the battery charge watts/volts and solar information such as solar volts/amps/watts.

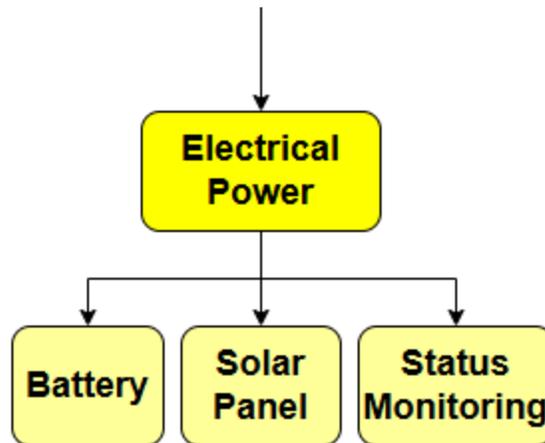


Figure 47: Electrical Power Subsystem Diagram

4.6.1 ELECTRICAL POWER IMPLEMENTATION

The implementation of the electrical power subsystem consisted of selecting an adequate solar panel and battery for providing power to the automated features of the coop as well as determining how the status of both of these items would be communicated to the user. One of the main requirements for this project stated that the battery's capacity needed to be sufficient for providing consistent power to the coop under normal operating conditions for a period of one week when recharged or assisted only by the solar panel. This requirement along with the electrical specifications of the components chosen for this project would influence the selection and implementation methods of the Autonomous Chicken Coop's electrical power source.

Table 5 shows a power budget that was created for the project based on the components chosen. It should be noted here that the “Max Power Consumption” value was calculated under the assumption that all three motors on the coop would be in “locked rotor” state at the same time. This was necessary for determining what the max power for the system could be under the most extreme condition, however it does not reflect the power consumption behavior of the coop during routine operation. Therefore, while this power budget provided an organized way of viewing the electrical requirements of the components selected, the maximum power consumption value calculated from this information did not factor into the choice of battery and solar panel for the coop. Further information behind the electrical components selected including the battery and solar panel can be found in the component datasheets listed in Appendix G.

Table 5: Autonomous Chicken Coop Power Budget

Component Name:	Manufacturer:	Part Number:	Rated Voltage: (V)	Rated Current: (A)	Notes:
Driver Motor (Left)	POLOLU	37D (ITEM #2828)	12	5.5	stall current
Drive Motor (Left) Encoder	POLOLU	37D (ITEM #2828)	5	0.01	
Drive Motor (Right)	POLOLU	37D (ITEM #2828)	12	5.5	stall current
Drive Motor (Right) Encoder	POLOLU	37D (ITEM #2828)	5	0.01	
Drawbridge Door Motor	CQRobot	CQR37D	12	5.5	stall current
Drawbridge Door Motor Encoder	CQRobot	CQR37D	5	0.01	
Coop Camera	DFRobot	ESP32-CAM	5	0.31	flash on, brightness max
Coop Processor	Arduino	MEGA 2560 Rev3	12	0.435	0.075 by itself & 0.04 per I/O pin
5V, 1.5A Step-Up/Step-Down Voltage Regulator	POLOLU	S13V15F5	12	0.016	quiescent current
Water Level Sensor (1)	CQRobot	CQRSENYW002	5	0.012	
Water Level Sensor (2)	CQRobot	CQRSENYW002	5	0.012	
Feed Level Sensor	OSEPP	HC-SR04	5	0.015	
				Total Rated Current:	17.330
				Max Power Consumption: (W)	205.307

4.6.1.1 BATTERY



Figure 48: FULLRIVER DC35-12 Battery [29]

The battery selected for the electrical power subsystem is a FULLRIVER DC35-12 deep cycle battery that was donated to this project by the University of Southern Indiana Solar Splash Club. It is a 12 Volt 35 Amp-hour battery at a 20-hour rate, which means it is capable of supplying 1.75 amps consistently every hour for twenty hours. It being a deep cycle battery meant that it would be sufficient for delivering sustained power over an extended period of time and could discharge a large percentage of its capacity without any permanent faults. Since many of the electrical components on the coop required a minimum of 12VDC for normal operation, this battery was a suitable choice for this project.

4.6.1.2 SOLAR PANEL, CHARGE SYSTEM, AND STATUS MONITORING



Figure 49: Renogy 100W Monocrystalline Solar Panel Mounted on the Autonomous Chicken Coop

The solar panel selected is a Renogy 12 Volt 100-Watt monocrystalline unit. It utilizes Passivated Emitter and Rear Cell (PERC) monocrystalline cells and has an overall efficiency of 17.8% [30]. This means that 17.8% of the sunlight that the panel's surface is exposed to is converted to usable energy. While this number may seem low, it is within the average range of most solar panels currently in production [31]. This solar panel also meets the standards set in place by the International Electrotechnical Commission in IEC 61215-1, which dictates the requirements for the design qualification of terrestrial photovoltaic modules suitable for long-term operation in open-air climates [32]. The solar panel's dimensions are 41.8 x 20.9 x 1.38 inches, and it weighs a total of 14.1 lbs. These dimensions ensured that it would mount onto the coop with minimal modifications to the existing roof structure.

Once the battery and solar panel had been selected, the next step was to choose an adequate solar charge controller. The preferred option was a Renogy Wanderer 10-amp charge controller due to its compact design and straightforward setup when paired with the Renogy solar panel mentioned previously. Figure 50 shows the Renogy Wanderer solar charge controller complete with the wires that connect to the battery and solar panel. Another reason it was an ideal choice was due to the battery's recommended charge current of 3-10 Amps. Since the maximum capable charge current allowed with the Wanderer is 10 Amps, the risk of overcharging the battery was minimized.

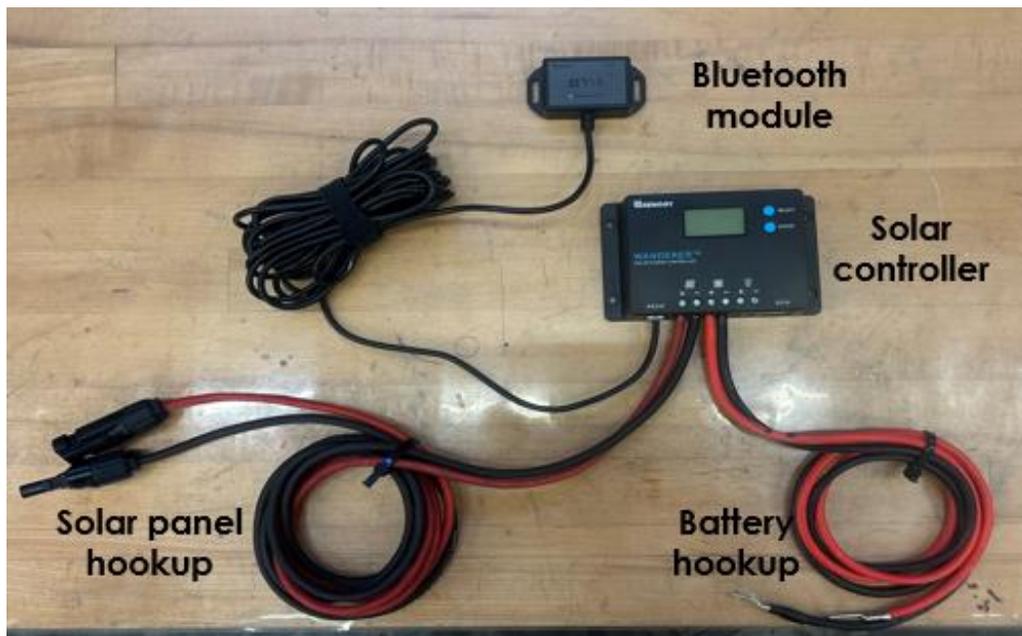


Figure 50: Renogy Wanderer Solar Controller Complete with Solar Panel and Battery Hookup

The Renogy Wanderer also had the option to integrate the BT-1 Bluetooth module for transmitting the status of the solar panel and battery to an app on the user's cell phone. Figures 51 and 52 are screenshots from the Renogy app that showcase the user's perspective when viewing the status of the solar panel and battery.



Figures 51 & 52: Screenshots from the Renogy App

The screenshot on the left in Figure 51 was captured during solar panel testing on a cloudy day, while the screenshot on the right in Figure 52 was captured during testing in full sun. While the testing was not conducted for a significant period of time, it was still concluded that the solar panel would be capable of outputting enough power to maintain sufficient charge on the battery, even on cloudier days.

5.0 RESULTS

Table 6: Constraint and Requirement Results

Constraint/Requirements	Completion	Measurable
Accommodate 2 chickens	YES	2
Move at a speed no greater than 1 foot per second	YES	0.97 ft/sec
Maintain sufficient battery charge for 1 week (with solar panel)	YES	-
Weigh less than 150 pounds	YES	149 <u>lb</u>
Function properly in chicken's ideal living temperature range, or 40-85	N/A	-
Be eco-friendly by using solar power	YES	-
Be cost-efficient by being under \$1000 to produce	YES	\$865.34
Coop relocation via a user programmed route	YES/NO	-
Feed and water level monitoring	Water YES/Feed NO	-
Be able to move effortlessly through semi-flat yard terrain	YES	-
Camera feed	YES/NO	-
Recordable battery level	YES	-
Recordable solar status	YES	-

The table above shows the design constraints and system requirements for this project. As shown, most of the specifications were successfully met. However, two criteria remain either unfulfilled or partially implemented: operation within the optimal temperature range for chickens (40–85°F) and coop relocation via a user-programmed route.

Upon evaluation, the requirement to function within the chicken's ideal temperature range was considered not applicable. The original design intent included integrating a fan and heating system to regulate internal temperature during seasonal extremes. However, space limitations within the coop housing made the inclusion of these components not practical, as they would have occupied the minimum required space for the chickens, thereby compromising animal well-being.

Regarding the user-programmed navigation feature, this functionality was only partially implemented. The intended design was to allow end-users to customize multiple movement paths for the coop. Due to budget and time limitations, a fully functional end-user interface for path programming could not be completed. However, the team successfully configured a single predefined route, demonstrating the feasibility of this feature. Future development can focus on

expanding this capability to allow user-defined route programming through a more intuitive interface.

6.0 FUTURE IMPROVEMENTS

- Add stronger driver motors to increase the torque and allow the coop to move on all terrains effortlessly.
- Add a feature that allows customers to select pre-programmed routes.
- Add temperature control system – fan and heater. The fan will help to cool down the coop during the hot season, and the heaters will help heat up the coop in cold weather.

7.0 REFLECTIONS

ALEX MALONE

The last two semesters of working on the autonomous chicken coop and witnessing its conception, design, and implementation have been an insightful and rewarding experience that I am sure will have a lasting impact on my professional and personal development. Not only have I had the opportunity to utilize the technical skills I have learned throughout my time in college but also sharpen some of the soft skills that are important for working on a large-scale collaborative project such as this one.

Going into pre-senior design in the fall of 2024 I did not have a specific idea in mind for what I wanted my senior project to be. Josh and I had decided to be partners, and he pitched the idea of the autonomous chicken coop to me. Claribeth, Kennedy, and Will joined us shortly after. During that semester I was anxious about how we could accomplish such an ambitious project and how it would be perceived. However, despite my concerns, we never failed to make significant progress from week to week and this is a theme that we carried on into the next semester.

One thing that I realized during the implementation of this project, is that while we were split between different disciplines of engineering, we would often find ourselves working on the same parts of the coop all at the same time. It was never the case where the electrical work that Josh and I did was completely isolated from the structural and mechanical work that our other team members did. From this, I learned that it is important to support all aspects of a project and

not just in specific aspects where you might have the expertise. I believe that this was one of the contributing factors to our success.

CLARIBETH DE LEON

Working on this project helped me strengthen my technical skills and gain hands-on experience in part fabrication, which are both important aspects of my professional development as a future engineer. This project challenged me to use my problem-solving skills and come up with more effective alternatives. From designing and fabricating our own components to finding the most cost-effective and feasible solutions already available on the market.

Learning how to apply soft skills like teamwork and clear communication was another important take away from this project journey. Everyone had different strengths, so figuring out how to divide tasks, stay on the same page, and support each other made a huge difference. Since we were a large, interdisciplinary group, we had to rely on each other's knowledge and stay open to different ideas to find the best solutions. Communication became key, whether we were solving problems, giving updates, or making sure we met deadlines.

Making most of the resources available to us and having the support of our mentors was also essential to completing the project successfully. I'm proud to say our project met the goals we set at the beginning, and overall, it was a fun and rewarding experience. The lessons we learned throughout this journey will help us in the future as engineers.

JOSHUA O'BRIAN

In the two semesters of designing and making this project, I have had a fun and enjoyable time with my group members. At the very beginning, I was excited but also very worried about how things would turn out. For example, when Alex and I first decided to partner up in pre-senior it was just us, the project ideas that we both had were not two-person projects. Luckily our professor brought us some good news about two members who did not have a project yet in a different pre-senior class. Alex and I proposed to the Autonomous Chicken Coop and let them decide if they wanted to join, hinting this was Claribeth and Kennedy. Lo and behold they agreed to join. The last group member to join was Will, he helped extremely with the SolidWorks stuff which was something that I had never learned at the time. So, from the beginning, it felt like a circus making sure we were all on the same page with the project.

After all that fun stuff with pre-senior designing and learning some of the important key factors of project management. The project had finally begun for us, filling out purchase orders with the items that were selected, receiving those items, and opening them up felt like Christmas again. That was all fun but the real enjoyable and learning experience was constructing parts with my group members. We were a multidisciplinary group comprised of Electrical, Manufacturing, and Mechanical engineers. At the start, it was hard to figure out what each group member was going to do but after some discussion, we figured it out. I want to take note of that, we made sure that we each contributed to this project in some way. Which in turn made it a great learning experience with each other.

So, my big takeaway is that things will not always go the way that were intended. For example, there were several times when Alex and I would be working on code, and it wouldn't work as intended. This was infuriating, there were times when we would have to take a short break go for a walk, and come back to it. This is important because it was not just Alex and I having this issue, other group members had the same issues but in different ways. I can proudly say that our group did not give up once on making this project happen. We all did our best to make this happen, which was more enjoyable to see.

KENNEDY SERMERSHEIM

Working on this automated chicken coop project has been a genuinely rewarding and eye-opening experience. From the initial concept ideas to seeing the fully assembled coop in action, the journey came with its challenges, learning moments, and a strong sense of accomplishment. One of the most rewarding aspects of the project was translating our ideas into a real, functioning system combining mechanical, electrical, and software components all come one cohesive design.

Through this project, I gained valuable hands-on experience in fabrication, component selection, and system integration. Designing real-world constraints like weather resistance, mobility, and automation pushed me to think critically and work closely with my team.

Beyond the technical growth, this project also helped strengthen my teamwork, management, and organization skills. I really appreciate working with such a supportive and collaborative team. They were always willing to help and made the process enjoyable.

WILL CRAIG

This project for me was a great learning experience, but also very fun and rewarding. As a mechanical engineer, I mainly worked hands-on with the chicken coop structurally and designed some key components that were implemented. It was neat to see my work being implemented and helping the project come together into its final form. It is one thing to work on problems on a homework assignment, but being able to solve problems in this project felt really rewarding because each thing the team or I accomplished brought us closer to our end goal.

My biggest takeaway from this project is that not everything goes according to plan all the time. It was easy to set a plan in place for what our day of work would entail, but once we began working, we oftentimes realized that more roadblocks than we might have expected would arise. However, this is just the nature of working towards a big goal, and through persistence and teamwork, our team was able to tackle these things and strengthen our bond.

Through all the things our team accomplished and all the adversities we faced during this senior design project, I can genuinely say I enjoyed the whole process and learned many valuable lessons that I will remember throughout the course of my engineering career.

8.0 CONCLUSION

This project aimed to design a self-sustaining chicken coop capable of autonomous relocation, feeding, watering, sheltering, and condition monitoring. The goal was to optimize the chickens' living conditions with minimal human intervention, addressing the labor-intensive nature of traditional chicken farming. A primary objective was to allow the coop to relocate without human assistance while simplifying care tasks for farmers. Additionally, the autonomous design supports rotational grazing, which helps prevent soil degradation and provides chickens with access to fresh forage, enhancing their nutrition. Moreover, key features include predator protection and feeding monitoring. A drawbridge door system was implemented to prevent predators from entering the housing area. Automatic feeders and waterers were integrated to maintain the chickens' well-being and further reduce manual labor. A camera system was also installed to allow remote monitoring of the chickens' health and safety. The project was completed within the \$1,000 budget and finalized by April 2025. Overall, the team considered the project successful, as it met most requirements and constraints, while also identifying areas for future improvement to ensure optimal functionality of the autonomous chicken coop.

REFERENCES

- [1] Sri Wahjuni, et al., “The Development of Chicken Coop Automatic Remote Visual Monitoring System,” *Scientific Journal of Informatics*, vol. 9, no. 2, pp. 161–168, 2022.
- [2] C. Kumar, et al., “Design and Fabrication of Automatic Poultry Foodfeeding System.”
- [3] E. M. M. Yusof, M. F. H. N. Sarkawi, and N. Yahya, “Automated Chicken Coop Management System to Improve the Quality of Chicken Production,” in *Advanced Transdisciplinary Engineering and Technology*, Cham: Springer, 2022, pp. 23–29.
- [4] UKKO Robotics, “Is Mobile Chicken Coop Right for Me?” *UKKO Robotics*, Oct. 1, 2024. [Online]. Available: www.ukkorobotics.com/news-and-resources/is-a-mobile-chicken-coop-right-for-me. [Accessed: Oct. 2, 2024].
- [5] “What Is the Ideal Temperature for a Chicken Coop?” *Chicken Keeping Secrets*. [Online]. Available: www.chickenkeepingsecrets.com/chicken-coops/what-is-the-ideal-temperature-for-a-chicken-coop/. [Accessed: Mar. 23, 2025].
- [6] "Free Range Chicken Care," Lincoln County Farm Center, [Online]. Available: <https://lincolncountyfarmcenter.com/blog/68569/-free-range-chicken-care>. [Accessed: Nov. 17, 2024].
- [7] "Chicken Coop and Run Size Calculator." *Chicken Pets*, ChickenPets.com, <https://chickenpets.com/chicken-coop-run-size-calculator>. Accessed 30 Apr. 2025.

- [8] Timeanddate.com. (2025). *Planets Visible in the Night Sky in Evansville, Indiana, USA*. [Online]. Available:
<https://www.timeanddate.com/astronomy/night/usa/evansville>. [Accessed: Feb. 03, 2025].
- [9] GICOOL. (n.d.). *GICOOL 72-108 Replacement Wheels for Oregon Mower*. [Online]. Available: <https://www.amazon.com/GICOOL-72-108-Wheels-Oregon-Mower/dp/B0DKDYQPZG/> . [Accessed: Mar. 30, 2025].
- [10] Pololu, "Pololu 150:1 Metal Gearmotor 37Dx73L, 64 CPR," TME. [Online]. Available: <https://www.tme.eu/en/details/pololu-2828/dc-motors/pololu/150-1-metal-gearmotor-37dx73l-64-cpr/>. [Accessed: Mar. 26, 2025].
- [11] T.C. Petterson and S.D. Gooch, "Rolling resistance of ATV tyres in agriculture," *ResearchGate*, May 2020. [Online]. Available:
https://www.researchgate.net/publication/342113424_ROLLING_RESISTANCE_OF_ATV_TYRES_IN_AGRICULTURE. [Accessed: Feb. 10, 2025].
- [12] Robot Marketplace. (n.d.). *BaneBots PDX16 - 16:1 Gearmotor*. [Online]. Available:
<https://www.robotmarketplace.com/products/0-pdx16.html>. [Accessed: Jan. 15, 2025].
- [13] eBay. (n.d.). *Vintage Toy Robot - 265206947176*. [Online]. Available:
<https://www.ebay.com/itm/265206947176> . [Accessed: Oct. 17, 2025].
- [14] Arduino. (n.d.). *Arduino Mega 2560 Rev3*. [Online]. Available: <https://store-usa.arduino.cc/products/arduino-mega-2560-rev3>. [Accessed: Jan. 28, 2025].

- [15] Arduino. (n.d.). *Arduino GIGA R1 WiFi*. [Online]. Available: <https://store-usa.arduino.cc/products/giga-r1-wifi>. [Accessed: Jan. 28, 2025].
- [16] AITRIP. (2022). *AITRIP 4 PCS BTS7960 43A High Power Motor Driver Module/Smart Car Driver Module for Arduino*. [Online]. Available: <https://www.amazon.com/dp/B09QKFFDQ3>. [Accessed: Jan. 30, 2025].
- [17] HiLetgo. (n.d.). *HiLetgo 4pcs L298N Motor Driver Controller Board Module Stepper Motor DC Dual H-Bridge for Arduino Smart Car Power UNO MEGA R3 Mega2560*. [Online]. Available: <https://www.amazon.com/dp/B07BK1QL5T>. [Accessed: Jan. 30, 2025].
- [18] Ryan Electronics, "UL 1007 | UL1569 Wire," *Ryan Electronics*, [Online]. Available: <https://www.ryanelectronics.com/products/ul-1007/?utm>. [Accessed: Jan. 30, 2025].
- [19] ASTM International, *Standard Specification for Standard Nominal Diameters and Cross-Sectional Areas of AWG Sizes of Solid Round Wires Used as Electrical Conductors*, ASTM B258-18, 2018. [Online]. Available: <https://store.astm.org/b0258-18.html>. [Accessed: Mar. 22, 2025].
- [20] Miller Manufacturing Company. (n.d.). *Watering Backyard Chickens*. [Online]. Available: <https://miller-mfg.com/blogs/blog/watering-backyard-chickens#:~:text=How%20much%20water%20does%20a,pints%20of%20water%20per%20day>. [Accessed: Mar. 23, 2025].

- [21] "How Much Feed Does a Chicken Eat per Day?" *Happy Chicken Coops*. [Online]. Available: www.happychickencoops.com/how-much-feed-does-a-chicken-eat-per-day/. [Accessed: Mar. 23, 2025].
- [22] DFRobot, "A02YYUW Waterproof Ultrasonic Distance Sensor (3~450cm, UART, IP67)," *DFRobot*, [Online]. Available: <https://www.dfrobot.com/product-1935.html>. [Accessed: Mar. 3, 2025].
- [23] "Distance measurement with ultrasonic sensor HC-SR04 using the Arduino with an UART connection (Python)," *ROS Wiki*. [Online]. Available: <https://wiki.ros.org/Drivers/Tutorials/DistanceMeasurementWithUltrasonicSensorHC-SR04ArduinoUARTPython>. [Accessed: Mar. 5, 2025].
- [24] "Ultrasonic Distance Sensor, UART Controlled 3-450cm Ranging DC 3.3-5V Waterproof Detector," Amazon.com, [Online]. Available: <https://www.amazon.com/dp/B0CGRM8BW2>. [Accessed: Mar. 5, 2025].
- [25] CQRobot, "Contact Water/Liquid Level Sensor SKU: CQRSENYW002," *CQRobot Wiki*, [Online]. Available: http://www.cqrobot.wiki/index.php/Contact_Water/Liquid_Level_Sensor_SKU:CQRSENYW002. [Accessed: Mar. 23, 2025].
- [26] SparkFun Electronics, "SparkFun Basic 16x2 Character LCD - White on Black, 5V (with Headers)," *SparkFun*, [Online]. Available: <https://www.sparkfun.com/products/18160>. [Accessed: Feb. 13, 2025].

- [27] R. R. Tutorials, "How to upload code to ESP32-CAM," *Random Nerd Tutorials*, 2021. [Online]. Available: <https://randomnerdtutorials.com/program-upload-code-esp32-cam/>. [Accessed: Mar. 26, 2025].
- [28] CQRobot, "810:1 Gearmotor," CQRobot Online Shop. [Online]. Available: https://www.cqrobot.com/index.php?route=product/product&path=242_243_244_294&product_id=1206. [Accessed: Jan. 27, 2025].
- [29] National Electrical Manufacturers Association, *NEMA 250-2003: Enclosures for Electrical Equipment (1000 Volts Maximum)*. Rosslyn, VA: NEMA, 2003. [Online]. Available: <https://webstore.ansi.org/standards/NEMA/nema2502003> [Accessed: Feb. 1, 2025]
- [30] IGS Energy, "Solar Panel Efficiency," *Energy Resource Center*, [Online]. Available: <https://www.igs.com/energy-resource-center/energy-101/solar-panel-efficiency>. [Accessed: Feb. 3, 2025].
- [31] Renogy, "100-Watt 12 Volt Monocrystalline Solar Panel - Compact Design," Renogy, 2021. [Online]. Available: <https://www.renogy.com/100-watt-12-volt-monocrystalline-solar-panel-compact-design>. [Accessed: Feb. 3, 2025].
- [32] IGS Energy, "Solar Panel Efficiency," *Energy Resource Center*, [Online]. Available: <https://www.igs.com/energy-resource-center/energy-101/solar-panel-efficiency>. [Accessed: Nov. 10, 2024].
- [33] IEC, "IEC 61215-1:2021 - Terrestrial photovoltaic (PV) modules - Design qualification and type approval - Part 1: Test requirements," International

Electrotechnical Commission, 2021. [Online]. Available:

<https://webstore.iec.ch/en/publication/61345>. [Accessed: Feb. 10, 2025].

APPENDIX

APPENDIX A: PROJECT TIMELINE

Work Performed	Due Date	Hard Deadline
Coop Structure	02/05/2025	No, schedule margin
Finished CAD Models	03/15/2025	No
Trouble Shooting/testing	04/23/2025	No, schedule margin
Final Design Completed	04/25/2025	Yes
Final Presentation	04/25/2025	Yes
Code Completion	05/01/2025	Yes
Redesign if needed	05/02/2025	No, schedule margin

APPENDIX B: BILL OF MATIERIALS

Part:	Retailer:	Price Per Unit:	Quantity Needed:	Total Price:
Motor driver	Amazon	\$21.99 (for 4-pack)	1	\$21.99
Microcontroller	Arduino Store	\$48.40	1	\$48.40
Chicken coop	Amazon	\$155.99	1	\$155.99
Electric motor	Pololu	\$51.95	2	\$103.90
Camera	Amazon	\$12.99	2	\$25.98
Caster wheel	Home Depot	\$4.69	1	\$4.69
Wheels	Harbor Freight		4	\$20.99
Wire	Amazon	\$11.22	1	\$11.22
Wire	Amazon	\$9.61	1	\$9.61
Mounting hub	Pololu	\$9.95	4	\$39.80
Solar kit	Amazon	\$89.99	1	\$89.99
Charge controller	In person	\$26.99	1	\$26.99
Adaptive cables	In person	\$18.99	1	\$18.99
Mounting brackets	In person	\$12.00	1	\$12.00
Enclosure box (small)	Newark	\$26.53	1	\$53.06
Enclosure box (large)	Newark	\$28.12	1	\$28.12
Arduino shield	Amazon	\$29.90	1	\$29.90
Caster wheel	Home Depot	\$21.68	2	\$21.68
L-Brackets for motors	In person	\$9.95	1	\$9.95
Red ring terminals (100 Pack)	Hutch & Son	\$4.80	1	\$4.80
Terminal blocks	Hutch & Son	\$15.27	2	\$30.54
Jumpers	Hutch & Son	\$0.69	24	\$16.50
ESP32 Antenna	Amazon	\$9.99	1	\$9.99
Filament	Amazon	\$25.99	1	\$25.99
Water sensor	Amazon	\$27.98	2	\$27.98
Pololu 5V, 1.5 A Step Up/Down	Pololu	\$26.85	3	\$26.85
Ultrasonic distance sensor	Amazon	\$19.69	1	\$19.69
Total Price:				\$865.34

APPENDIX C: Failure Mode and Effects Analysis (FMEA)

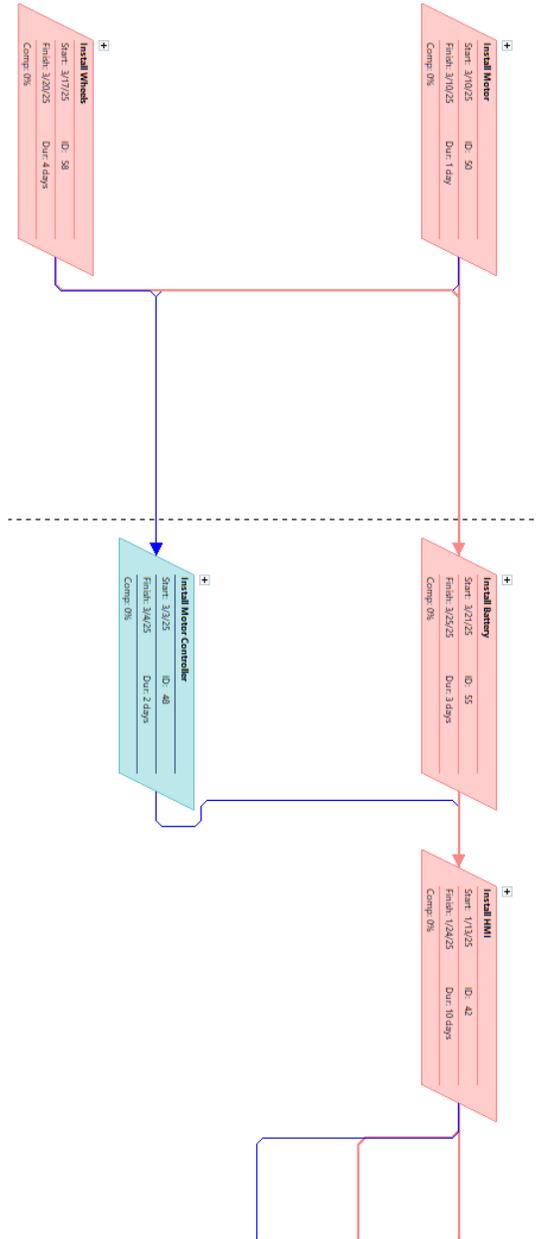
FAILURE MODE AND EFFECTS ANALYSIS

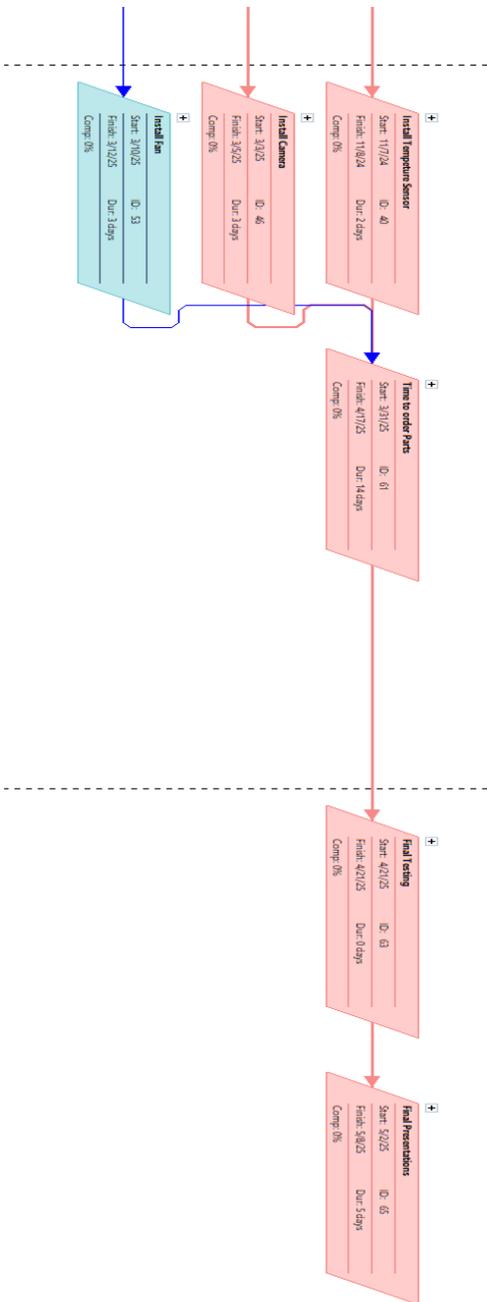
Item: Autonomous Moving Chicken Coop Responsibility: _____
 Model: _____ Prepared by: _____
 Core Team: _____

FMEA number: _____
 Page : 1 of 1
 FMEA Date (Orig): _____ Rev: 1

Process Function	Potential Failure Mode	Potential Effect(s) of Failure	S e v	Potential Cause(s)/ Mechanism(s) of Failure	Occurrence	Current Process Controls	Detec	R P N	Recommended Action(s)	Responsibility and Target Completion Date	Action Results				
											Actions Taken	S e v	O c c	D e t	R P N
Power	no charge in battery	all power systems fail	7	not plugged in	3	a light will show up	8	168	added backup battery			7	2	1	14
			7	no power to electrical emponents (motors)	3		4	84	indicator light			7	2	1	14
			7	wiring malfunctioning	1		10	70	none			4	1	10	40
			7	not holding charge	2		6	84	battery indicator (laptop)			4	2	1	8
Wheels	Coop not able to move	Coop can get stuck	9	Flat tire	8	notification to user	7	504	keep sharp objects out of the path			5	3	3	45
			9	Axle Failure	6		10	540	regular inspection			5	2	3	30
			9	Corrosion or Rust	8		5	360	improve the material properties			4	3	3	36
			9	Broken or Loose Wheel Mounts	6		3	162	regular inspection			5	2	4	40
			9	Jammed or Obstructed Wheels	6		6	324	clean regularly			5	3	2	30
HMI	Breaking	Not getting notifications	8	looses power	6	App status not working	8	384	none			8	6	8	384
			8	weather conditions	9		7	504	locktight box			7	2	4	56
			8	wire malfunction	3		7	168	none			8	3	7	168
Coop Structure	Breaking	Hurting Chickens	10	Preditors	8	Notification to User	7	560	Check Cameras Frequently			10	4	3	120
			10	Material Quality	7		8	560	Regular inspection			7	3	3	63

APPENDIX D: PERT CHART



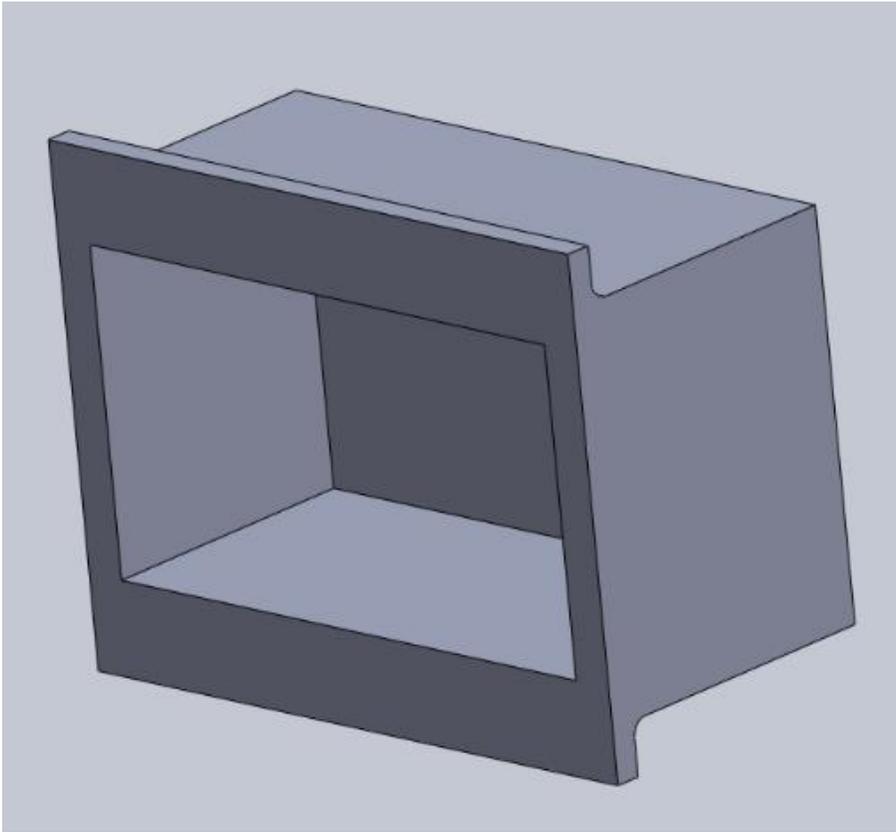
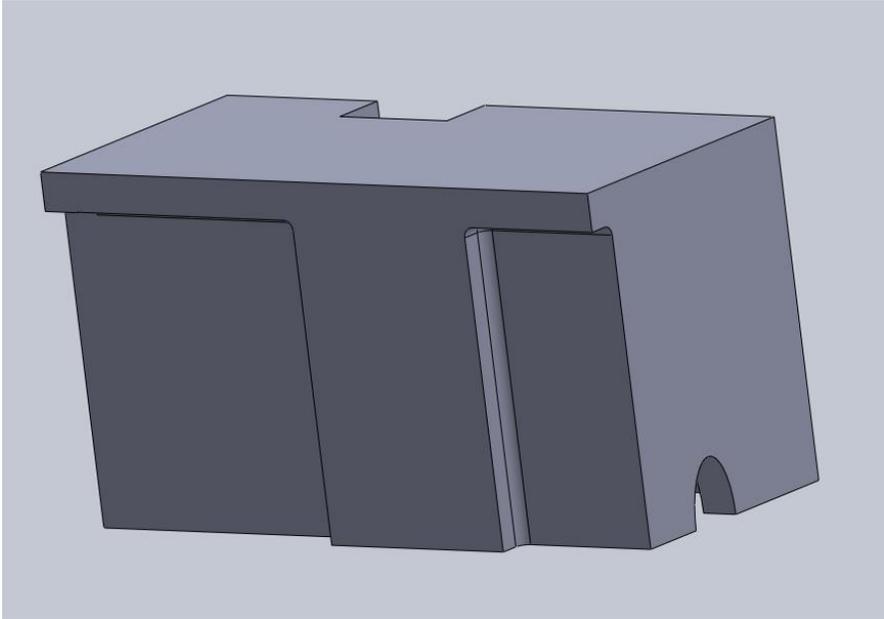


**APPENDIX E: ABET OUTCOME 2, DESIGN FACTOR
CONSIDERATIONS & STANDARDS**

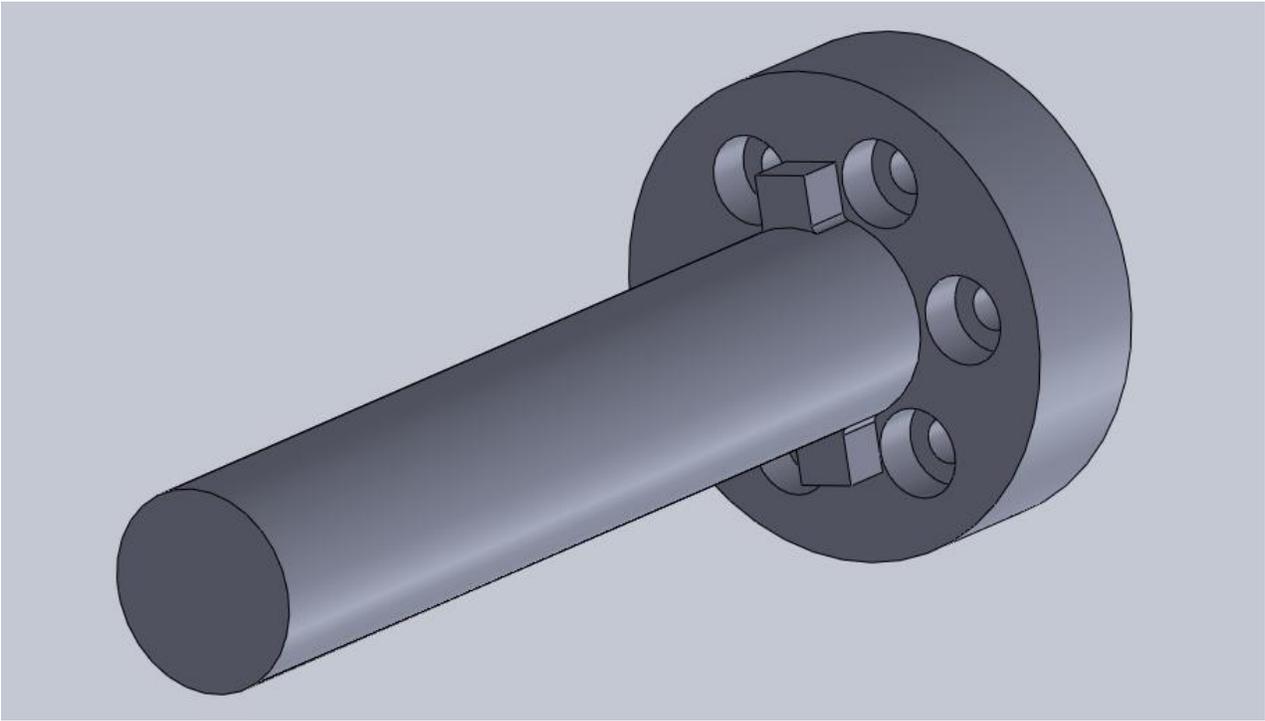
Design Factor	Page number, or reason not applicable
Public health safety, and welfare	Not applicable since this project is not for public use.
Global	Not applicable since the design of this project did not involve global considerations.
Cultural	Not applicable since the design of this project did not involve cultural considerations.
Social	Not applicable since this project contains no social aspects.
Environmental	Page 37 & 38
Economic	Page 45
Ethical & Professional	Page 21 & 22
Reference for Standards	UL 1007 & UL1569 (Page 21 & 22) ASTM B258-18 (Page 21 & 22) NEMA 250-2003 (Page 37 & 38) IEC 61215-1:2021 (Page 45)

APPENDIX F: SOLIDWORKS' DRAWINGS

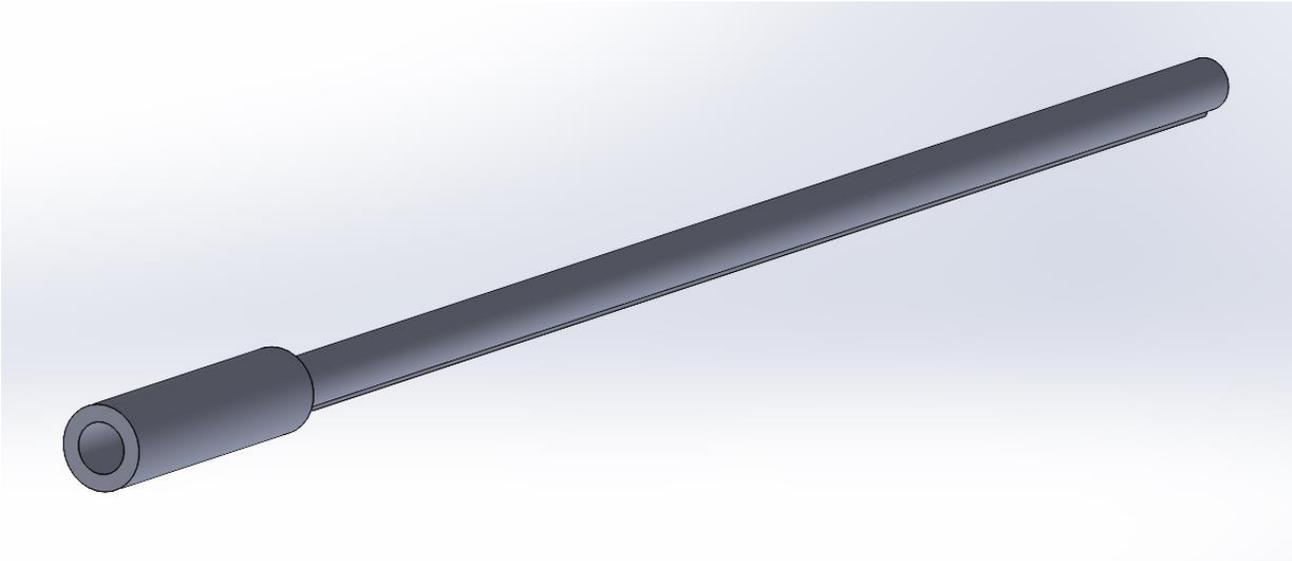
Motor Enclosures



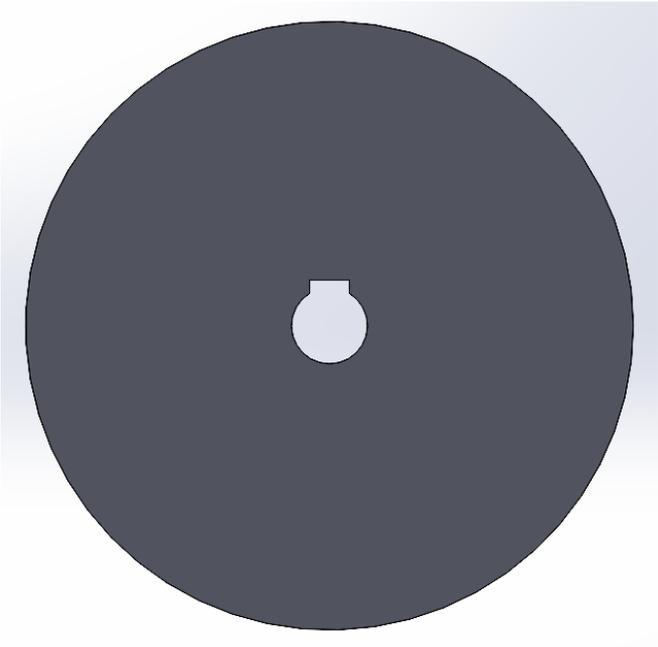
Wheel Axles



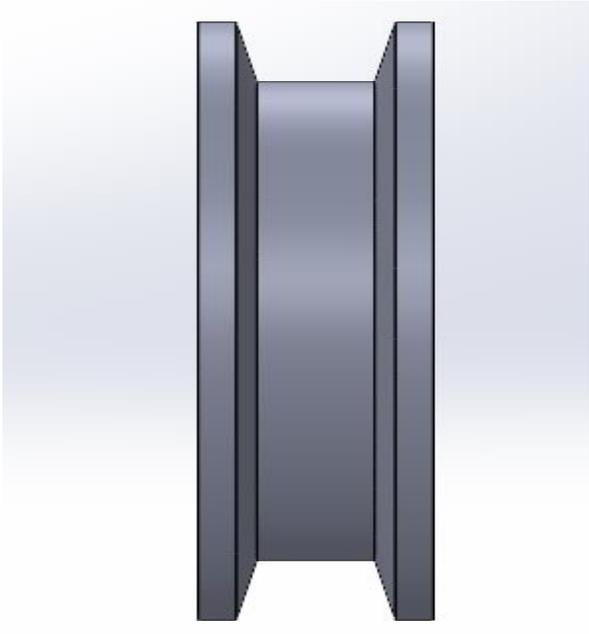
Rod and Motor Adapter



Side View of Spool



Front View of Spool



APPENDIX G: DATASHEET HYPERLINKS

Component:	Hyperlink:
Solar Panel	https://www.renogy.com/content/RNG-100D-SS/RNG-100D-SS-G3%20Datashet
Solar Charge Controller	https://store-fhncf.mybigcommerce.com/content/RNG-CTRL-WND10/WND10-
Battery	https://www.nationwide-battery.com/pdf/DC35-12.pdf
Drive Motors	https://www.pololu.com/file/0J1736/pololu-37d-metal-gearmotors-rev-1-2.pdf
Drawbridge Door Motor	http://www.cqrobot.wiki/index.php/DC_Gearmotor_SKU:_CQR37D
Motor Drivers	https://image.dfrobot.com/image/data/DRI0018/BTS7960.pdf
Step-Down Voltage Regulator	https://www.pololu.com/product/4084
Arduino Mega 2560 Rev3	https://docs.arduino.cc/resources/datasheets/A000067-datasheet.pdf
ESP32-CAM	https://media.digikey.com/pdf/Data%20Sheets/DFRobot%20PDFs/DFR0602_We
FT232RL USB to TTL Serial Board	https://ftdichip.com/wp-content/uploads/2020/08/DS_FT232R.pdf
Feed Sensor	https://www.dypcn.com/uploads/A02-Datasheet.pdf
Water Sensor	http://www.cqrobot.wiki/index.php/Contact_Water/Liquid_Level_Sensor_SKU:
Battery Enclosure	https://www.budind.com/wp-content/uploads/ptr28492.pdf
Regular Enclosure	https://www.budind.com/wp-content/uploads/hbdps28710.pdf

APPENDIX I: PROJECT CODE

```
#include <Wire.h>

#include "RTClib.h"

#include <LiquidCrystal.h>

#include <EEPROM.h>

// ===== LCD Setup =====

LiquidCrystal lcd(5, 6, 7, 8, 9, 10); // RS, E, D4–D7

// ===== RTC =====

RTC_DS3231 rtc;

// ===== Ultrasonic Sensor (Food Level) =====

const int trigPin = 2;

const int echoPin = 3;

// ===== Water Level Sensor =====

const int waterLevelPin = 4;

// ===== Coop Movement Motors (BTS7960) =====

#define RPWM1 13

#define LPWM1 12

#define RPWM2 18

#define LPWM2 19
```

```

// ===== Encoders for Coop Movement =====
#define encoderA1 10
#define encoderB1 11
#define encoderA2 21
#define encoderB2 20

volatile int encoderCount1 = 0;
volatile int encoderCount2 = 0;

// ===== Door Motor (BTS7960) =====
#define RPWM_DOOR 6
#define LPWM_DOOR 7

// ===== Movement Configuration =====
const int coopSpeed = 255;          // Full speed for coop movement
const int doorSpeed = 255;         // Full speed for door motor
const int coopTarget = 3820;       // Encoder count for ~20 inches of travel
const int DAYS_TO_MOVE = 7;        // Days forward before returning
int moveCounter = 0;               // Tracks coop position cycle
bool hasMovedToday = false;
bool openedToday = false;
bool closedToday = false;

// ===== Door Timing (Safety) =====

```

```

const unsigned long moveDuration = 31000; // Normal door movement duration
const unsigned long safetyTimeout = 35000; // Max duration before forced stop

void setup() {
  Serial.begin(115200);
  Wire.begin();
  rtc.begin();

  // Uncomment once to set RTC to compile time
  // rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));

  lcd.begin(16, 2); // Initialize 16x2 LCD

  // Setup sensor pins
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(waterLevelPin, INPUT);

  // Setup motor control pins
  pinMode(RPWM1, OUTPUT); pinMode(LPWM1, OUTPUT);
  pinMode(RPWM2, OUTPUT); pinMode(LPWM2, OUTPUT);
  pinMode(RPWM_DOOR, OUTPUT); pinMode(LPWM_DOOR, OUTPUT);

  // Setup encoder pins with interrupts

```

```

pinMode(encoderA1, INPUT); pinMode(encoderB1, INPUT);
pinMode(encoderA2, INPUT); pinMode(encoderB2, INPUT);
attachInterrupt(digitalPinToInterrupt(encoderA1), readEncoder1, CHANGE);
attachInterrupt(digitalPinToInterrupt(encoderA2), readEncoder2, CHANGE);

// Restore previous movement state from EEPROM
moveCounter = EEPROM.read(0);

Serial.println("System Initialized");
}

void loop() {
  DateTime now = rtc.now();
  int hour = now.hour();
  int minute = now.minute();

  // ===== Door Control Schedule =====
  if (hour == 7 && minute == 0 && !openedToday) {
    openDoor();
    openedToday = true;
    closedToday = false;
  }

  if (hour == 22 && minute == 0 && !closedToday) {

```

```

closeDoor();

closedToday = true;

openedToday = false;
}

// ===== Coop Movement at Midnight =====
if (hour == 0 && minute == 0 && !hasMovedToday) {
  if (moveCounter < DAYS_TO_MOVE) {
    Serial.println("Moving Forward");
    moveForward(coopTarget);
    moveCounter++;
  } else if (moveCounter < 2 * DAYS_TO_MOVE) {
    Serial.println("Moving Backward");
    moveBackward(coopTarget);
    moveCounter++;
  } else {
    Serial.println("Cycle Complete - Resetting Counter");
    moveCounter = 0;
  }
  stopMotors(2000);
  hasMovedToday = true;
  EEPROM.update(0, moveCounter); // Save moveCounter
}

```

```

// ===== Reset Daily Flags at 2:00 AM =====
if (hour == 2 && minute == 0) {
    hasMovedToday = false;
    openedToday = false;
    closedToday = false;
}

// ===== Sensor Monitoring =====

float foodDistance = readUltrasonic();

bool waterDetected = digitalRead(waterLevelPin);

lcd.setCursor(0, 0);
lcd.print("Food: ");
lcd.print(foodDistance, 1);
lcd.print(" cm ");

lcd.setCursor(0, 1);
lcd.print("Water: ");
lcd.print(waterDetected ? "YES " : "NO ");

Serial.print("Food: "); Serial.print(foodDistance, 1);
Serial.print(" cm | Water: "); Serial.println(waterDetected ? "YES" : "NO");

delay(60000); // Check every 60 seconds

```

```

}

// ===== Ultrasonic Sensor Function =====
float readUltrasonic() {
    digitalWrite(trigPin, LOW); delayMicroseconds(2);
    digitalWrite(trigPin, HIGH); delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    long duration = pulseIn(echoPin, HIGH);
    return (duration * 0.034) / 2.0; // Distance in cm
}

// ===== Door Control Functions =====
void openDoor() {
    Serial.println("Opening Door");
    if (runDoorMotor(doorSpeed, true)) Serial.println("Door Opened");
    else Serial.println("⚠ Door Opening Timeout");
    stopDoor();
}

void closeDoor() {
    Serial.println("Closing Door");
    if (runDoorMotor(doorSpeed, false)) Serial.println("Door Closed");
    else Serial.println("⚠ Door Closing Timeout");
    stopDoor();
}

```

```
}
```

```
bool runDoorMotor(int speed, bool open) {
```

```
    unsigned long start = millis();
```

```
    while (true) {
```

```
        if (open) {
```

```
            analogWrite(RPWM_DOOR, speed); analogWrite(LPWM_DOOR, 0);
```

```
        } else {
```

```
            analogWrite(RPWM_DOOR, 0); analogWrite(LPWM_DOOR, speed);
```

```
        }
```

```
        unsigned long elapsed = millis() - start;
```

```
        if (elapsed >= moveDuration) return true;
```

```
        if (elapsed >= safetyTimeout) return false;
```

```
    }
```

```
}
```

```
void stopDoor() {
```

```
    analogWrite(RPWM_DOOR, 0); analogWrite(LPWM_DOOR, 0);
```

```
}
```

```
// ===== Coop Movement Functions =====
```

```
void moveForward(int target) {
```

```
    resetEncoders();
```

```

unsigned long start = millis();
while (encoderCount1 < target && encoderCount2 < target) {
    analogWrite(RPWM1, coopSpeed);
    analogWrite(LPWM1, 0);
    analogWrite(RPWM2, 0);
    analogWrite(LPWM2, coopSpeed);
    if (millis() - start > 30000) break;
}
}

void moveBackward(int target) {
    resetEncoders();
    unsigned long start = millis();
    while (encoderCount1 > -target && encoderCount2 > -target) {
        analogWrite(RPWM1, 0);
        analogWrite(LPWM1, coopSpeed);
        analogWrite(RPWM2, coopSpeed);
        analogWrite(LPWM2, 0);
        if (millis() - start > 30000) break;
    }
}

void stopMotors(int delayMs) {
    analogWrite(RPWM1, 0); analogWrite(LPWM1, 0);

```

```
analogWrite(RPWM2, 0); analogWrite(LPWM2, 0);  
delay(delayMs);  
}  
  
// ===== Encoder Functions =====  
void resetEncoders() {  
    noInterrupts();  
    encoderCount1 = 0;  
    encoderCount2 = 0;  
    interrupts();  
}  
  
void readEncoder1() {  
    encoderCount1 += (digitalRead(encoderB1) == HIGH) ? 1 : -1;  
}  
  
void readEncoder2() {  
    encoderCount2 += (digitalRead(encoderB2) == HIGH) ? 1 : -1;  
}
```