



Palestine Polytechnic University
College of Information Technology and Computer
Systems Engineering

IoT Wi-Fi Based Smart Cage

By

Yazeed Taweel

Anas Darabee

Supervisor

Eng. Mazen Kayed Zalloum

A project submitted in fulfillment for the Degree of Bachelor in
Computer Systems Engineering

May 2018

Contents

AcknowledgmentV
Abstract.....VI
List of Figures.....VII
List of Tables.....IX

1 Introduction

1.1 Overview of the project.....1
1.2 Motivation.....1
1.3 Goals and objectives.....1
1.4 Short description of the project.....2
1.5 List of requirements.....2
1.6 Expected results.....3
1.7 Limitation.....3
1.8 Literal review.....3

2 Theoretical Background

2.1 Overview.....5
2.2 Hardware equipment.....5
 2.2.1 Arduino Uno.....5
 2.2.2 Grove-Moisture sensor.....7
 2.2.3 Light Dependent Resistor sensor (LDR).....8
 2.2.4 PIR Motion sensor.....9
 2.2.5 LM35 Temperature sensor.....10
 2.2.6 Bluetooth adapter HC-06.....11

2.2.7 Solenoid Valve	13
2.2.8 L298N Dual H Bridge.....	15
2.2.9 Light Source.....	16
2.2.10 Stepper motor.....	16
2.2.11 Fan.....	17
2.2.12 Wi-Fi Shield.....	18

3 System Design

3.1 Overview.....	19
3.2 Theatrical background.....	19
3.3 Methodology.....	19
3.4 Sub Circuits.....	20
3.4.1 Soil-Moisture sensor.....	20
3.4.2 LDR sensor.....	21
3.4.3 PIR motion sensor.....	21
3.4.4 LM35 Temperature sensor.....	22
3.4.5 Bluetooth HC-06.....	22
3.4.6 Water Valve.....	23
3.4.7 RN-171 WI-FI shield.....	23
3.4.8 Stepper motor.....	24
3.4.9 Fan.....	25
3.4.10 Circuit Diagram.....	26
3.5 Block Diagram.....	27
3.6 System Flowchart.....	29

3.6.1 Soil-Moisture flowchart.....	29
3.6.2 LDR flowchart.....	30
3.6.3 Motion sensor flowchart.....	31

4 Hardware & Software Implementation

4.1 Overview.....	32
4.2 Operating System for mobile	32
4.3 Software Implementation Tools.....	32
4.3.1 Android Studio IDE.....	32
4.3.2 Arduino Application.....	34
4.4 Hardware Implementation.....	35
4.4.1 Motion sensor configuration	35
4.4.2 Grove Soil-Moisture sensor configuration	36
4.4.3 Temperature sensor configuration	37
4.4.4 Stepper motor configuration	39
4.4.5 Bluetooth adapter HC-06 configuration	40
4.4.6 LDR sensor configuration	41
4.4.7 Solenoid Valve configuration	42
4.4.8 Fan connections	43
4.4.9 Integrated system	44

5 Testing

5.1 Overview.....	45
5.2 Unit testing.....	45
5.2.1 Hardware components testing	45
5.2.2 Android Application Testing	45
5.3 Integration testing	45
5.3.1 Android – Arduino system integration test	46

6 Conclusion & Future work

6.1 Overview.....	47
6.2 Implementation challenges and issues	47
6.2.1 Hardware challenges	47
6.2.2 Software challenges	47
6.3 Final Result	47
6.4 Future Work	48

Acknowledgment

All praise is due to Allah who assist us in completion our graduation project.

We present thank for our supervisor Eng. Mazen Zalloum for his support and time and effort he spent with us during all phases of the project. We also want to express our special gratitude to our instructors, course supervisor in the PPU for their patience and effort given to us during our study in the PPU.

Finally, we are thankful to our family, especially our parents for their help, patience, and financial and incorporeal support during our whole life.

Abstract

As we notice the world is moving toward technology in most of life aspects since the time cost and human efforts saving that technology provides. So why not to exploit technology in serving animals and birds.

In this documentation we will present our project that aims to design and implement an automatically bird feeder system based on Wi-Fi technique. The system shall provide feeding and watering operation for birds, also it keeps detecting the motion of the bird to inform the breeder in risk status.

The bird feeder system is build using an Arduino microcontroller, some sensors like Light Dependent Resistor and Soil-Moisture sensor, and other needed hardware equipments. Each sensor has a special function to do, basically sending values to the microcontroller which in turn receives them and execute the desired function based on them. Every unit function in the project was successfully build.

The system was verified successfully, so it provides the required functions in a good way. Since the project is based on the Wi-Fi, it should provide big comfort for the breeders who have only to use an android application to control the system remotely. And from this point we expect that our project can be a successful product.

List of Figures

Figure 2.1 Arduino Uno.....	6
Figure 2.2 Grove Soil-Moisture sensor.....	7
Figure 2.3 FC-28 Soil-Moisture sensor-Specifications	8
Figure 2.4 LDR sensor.....	9
Figure 2.5 PIR motion sensor.....	9
Figure 2.6 PIR sensor-simple schematic.....	10
Figure 2.7 LM35 Temperature sensor	11
Figure 2.8 Bluetooth adapter HC-06.....	11
Figure 2.9 Bluetooth adapter HC-06 schematic.....	12
Figure 2.10 Solenoid Valve.....	13
Figure 2.11 Solenoid Valve work principle	13
Figure 2.12 Step-up Chopper circuit	14
Figure 2.13 Chopper device	14
Figure 2.14 L298N H Bridge	15
Figure 2.15 Stepper motor	16
Figure 2.16 ULN2003-Driver	16
Figure 2.17 Stepper motor-ULN2003-Arduino connections	17
Figure 2.18 Wi-Fi Shield	18
Figure 3.1 Soil sensor connections	20
Figure 3.2 LDR connections	21
Figure 3.3 PIR motion sensor circuit	21
Figure 3.4 LM35 Temperature sensor circuit	22
Figure 3.5 Bluetooth module circuit	22

Figure 3.6 Solenoid Valve circuit	23
Figure 3.7 Wi-Fi shield	23
Figure 3.8 Wi-Fi shield connection	24
Figure 3.9 Stepper motor connections	24
Figure 3.10 Fan connections	25
Figure 3.11 Overall system circuit	26
Figure 3.12 Block diagram 1	28
Figure 3.13 Soil-Moisture sensor flowchart	29
Figure 3.14 LDR sensor flowchart	30
Figure 3.15 Motion sensor flowchart	31
Figure 4.1 Searching Bluetooth device interface	33
Figure 4.2 Wi-Fi Network configurations interface	33
Figure 4.3 Server settings interface	34
Figure 4.4 System control interface	34
Figure 4.5 Motion working principle	35
Figure 4.6 Motion sensor connection	35
Figure 4.7 Motion sensor schematic diagram	36
Figure 4.8 Grove Soil-Moisture sensor connection	36
Figure 4.9 Grove Soil-Moisture sensor schematic	37
Figure 4.10 Temperature sensor connection	37
Figure 4.11 Temperature sensor schematic diagram	38
Figure 4.12 Stepper motor connection	39
Figure 4.13 Stepper motor schematic diagram	39
Figure 4.14 HC-06 Bluetooth adapter connection	40
Figure 4.15 HC-06 Bluetooth adapter schematic	40
Figure 4.16 LDR sensors connection	41
Figure 4.17 LDR sensors schematic diagram	41

Figure 4.18 Solenoid Valve working principle	42
Figure 4.19 Solenoid Valve schematic diagram	42
Figure 4.20 Fan connections	43
Figure 4.21 Fan schematic diagram	43
Figure 4.22 Complete system	44
Figure 4.23 Complete system schematic diagram	

List of Tables

Table 2.1 Arduino Uno specifications.....	6
Table 2.2 L298N Input pins	15

Chapter 1

Introduction

1.1 Overview of the project

Microcontrollers are used to develop smart systems which can be in various applications in our daily life needs. One of the microcontroller applications is to design and implement a smart system for breeders of animals and birds. With the emergence of smartphone in our daily life, developers and designers are motivated to make a connection between smart mobiles and microcontrollers using wireless links.

The main problem in our project is to design and implement a smart care system which will assist animals and bird breeders in feeding, monitoring, and watching their health state. Our project will initially concentrate on designing the feeding smart system using microcontroller and sensors, then developing an android application that will be the mean to connect to the system to make easy control of the system. The system will also take considerations for another issue like clearing, watching bird motion. So, in the final form we would have a systematic care system acts as a human.

1.2 Motivation

Intelligent systems have become dramatically required in recent days, it intervenes in most aspects of our life, because it often makes our life easier and more manageable. Care Smart System for bird breeder is an interesting application in our life, it will reduce time and efforts spent by the bird breeders.

1.3 Goals and Objectives

The goal of this project is to build a smart care cage that could achieve the following objectives:

Project objectives:

1. Design a system that monitors water and food elements, so the system has the ability to fill the pots of each element in case that there is much loss in the quantity of each one. This is going to be powered in three modes, automatically along the time, a given time period, and manually by sending a command from the user.
2. Interconnection with the smartphone to send information such as the amount of consumed water and food, state of life (alive or not) and temperature degrees of the place.
3. Controlling the whole system through internet.

1.4 Short description of the project

The system generally monitors the state of the animal and surrounding environmental conditions to attempt control and adjust them as required, and inform the user by the condition continuously. We will make the system acts as absolutely as the human do.

On the other hand, the system provides the animal needs of food, water and others according to mechanism determined by the user of the system where the user adjusts the system on a particular mode and leaves it to be applied systematically as required.

1.5 List of Hardware components

Our hardware project requirements to implement the system are:

1. Light Dependent Resistor (LDR), motion and temperature sensors.
2. Grove-Moisture sensor.
3. Arduino microcontroller.
4. Wi-Fi Shield.
5. Bluetooth adapter.
6. Stepper motor.
7. Light Source (Laser).
8. Solenoid Valve
9. L298N Dual H-Bridge
10. Step up Chopper

1.6 Expected Results

We expect to accomplish the following at the end of the project:

1. A simple and low cost Automatic Care System for bird breeders will be build.
2. The user will be informed on the general state of the system and other issues.
3. The system is achieving the comfort for the breeders.

1.7 Limitations

1. Putting the cage in a place that the motion sensor has to detect only the bird motion.
2. Bird's health information.
3. Internet existence at the breeder.

1.8 Literature Review

The idea of our project comes from the need to achieve the rest for the breeders, and during the research on similar ideas we found the two implemented following ideas:

1. Electronic Bird feeder, developed by Ergo Systems Inc. The system has the ability to manage only feeding operation without controlling from the breeder. It provides fixed amount of seeds depending on a timer up to eight times per filling process. The system needs four separated batteries as power supply, the whole system is controlled by a digital circuit with timer and special processing unit.

Our project has more advantages compared with the previous project. It supports watering operation and can be controlled based on smartphone application manual or depending on sensors as need. Also, our project uses programmable microcontroller that can be easily updated to change the mechanism of the system. And supports many useful functions -mentioned previously- not available in Electronic Bird feeder.

2. Bird Auto Feeder developed by Super-Feed Enterprise Inc. This project supports Wi-Fi technique to make the ability of controlling the system wirelessly via smartphone. It manages feeding operation by providing food three times per filling process. This project has the advantage of the ability of distributing the system which means that you can install the same system on different locations and still controlled as one system.

Our project has more advantages compared with the previous project. It manages the system using many sensors, timer in addition to manual controlling via smartphone. Also, our project gives information about cage status to the smartphone user, but this doesn't support this advantage and too many functions in our project as shown in goals.

Chapter 2

Theoretical Background

2.1 Overview

This chapter introduces the theoretical background of the project, description of some hardware components that will be used in our project. Finally, design specification, limitation and constraints will be addressed.

2.2 Hardware equipment

This section describes all hardware used in our project, it presents a figure for each one with short description about its work principle and why it is used in the system.

2.2.1 Arduino Microcontroller

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board.

Technology has been developing many microcontrollers that can be used to launch and process most of the functions that computer can do. In our project we selected the Arduino microcontroller. This microcontroller has less cost and it is very easy to deal with, it doesn't require a special operating system like Raspberry pi microcontroller, just an application to communicate with the components, there is a flexibility to write code so you can write your own code for a specific project and manipulate it easily at any time.

There are many types of Arduino microcontroller such as UNO, Mega, Lily Pad and Arduino BT. The desired type that are going to use in our project is **Arduino Uno** as shown in figure 2.1. This type has the full capability to handle our system in the best way.

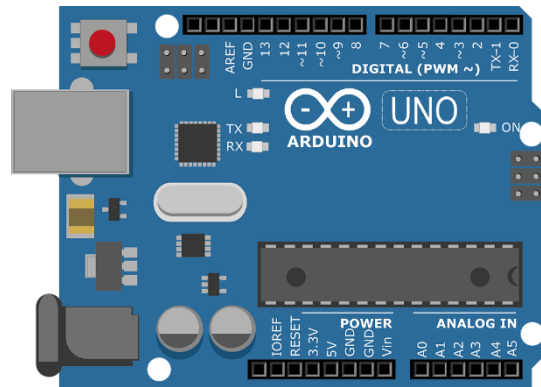


Figure 2.1 Arduino Uno

Table 2.1 shows some specifications of Arduino UNO type.

Table 2.1 Arduino Uno specifications

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	20mA
DC Current for 3.3V Pin	50mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm

2.2.2 Soil-Moisture sensor

The desired sensor can measure and report the humidity of the soil and air. This sensor has the function to make the water addition decision through inserting it directly into the water pot.

The soil moisture sensor consists of two probes which are used to measure the volumetric content of water. The two probes allow the current to pass through the soil and then it gets the resistance value to measure the moisture value.

When there is more water, the soil will conduct more electricity which means that there will be less resistance. Therefore, the moisture level will be higher. Dry soil conducts electricity poorly, so when there will be less water, then the soil will conduct less electricity which means that there will be more resistance. Therefore, the moisture level will be lower.

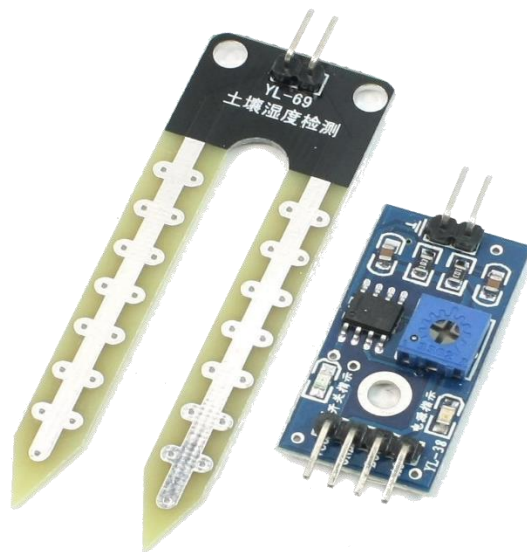


Figure 2.2 FC-28 Soil-Moisture sensor

Figure 2.3 describes the pins of the sensor and the driver FC-28 that takes the value from the sensor and sends it to the Arduino microcontroller.

The Module in figure 2.3 contains a potentiometer which will set the threshold value and then this threshold value will be compared by the LM393 comparator. The output LED will light up and down according to this threshold value.

The FC-28 sensor has four pins as shown in figure 2.3

- VCC: Power supply
- GND: Ground
- A0: Analog output
- D0: Digital output



Figure 2.3 FC-28 Soil-Moisture sensor-Specifications

2.2.3 Light Dependent Resistor sensor (LDR)

An LDR is a component that has a variable resistance that changes with the light intensity that falls upon it. This allows them to be used in light sensing circuits.

When the fallen light intensity increases, the resistance changes by decreasing which causes new output voltage. The goal of the use of LDR is to detect the shortfall of food, the sensor will be inserted inside the food and there will be a source light focused on the food pot, so when the food decreases, the LDR will be exposed to the light and this indicates that there is need to add more food.

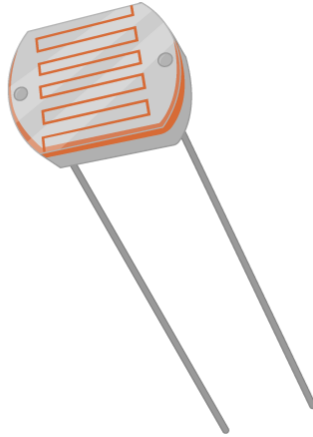


Figure 2.4 LDR sensor

2.2.4 PIR Motion sensor

PIR sensors (Passive Infrared) allow you to sense motion, almost always used to detect whether a human has moved in or out of the sensors range. They are small, low-power and easy to use.



Figure 2.5 PIR Motion sensor

The PIR sensor itself has two slots in it, each slot is made of a special material that is sensitive to IR. When the sensor is idle, both slots detect the same amount of IR and the ambient amount radiated from the room or walls or outdoors.

When a warm body like a human or animal passes by, it first intercepts one half of the PIR sensor, which causes a positive differential change between the two halves.

So, this sensor is very useful to detect the motion of the bird which helps in the state knowledge of it, and based on the motion we can decide whether the bird in a normal state or not.

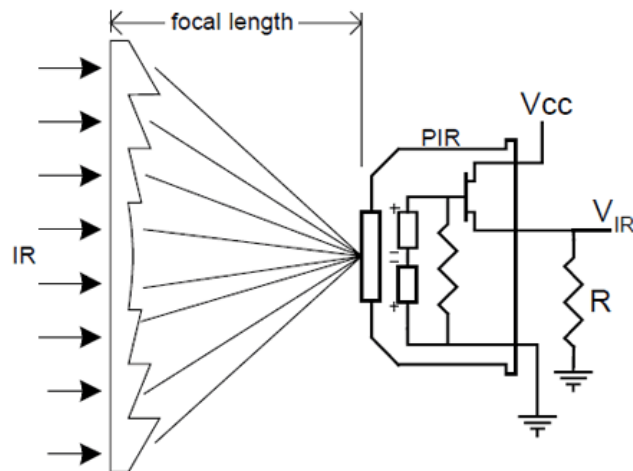


Figure 2.6 PIR sensor-simple schematic

The PIR acts as a digital output so all you need to do is listen for the pin to flip high (detected) or low (not detected). Power the PIR with 5V and connect ground to ground. Then connect the output to a digital pin.

The PIR motion sensor has three pins:

- VCC: Power (5-12V)
- GND: Ground
- OUT: Digital output signal (0 or 1)

2.2.5 LM35 Temperature sensor

LM35 is a precision IC temperature sensor with its output proportional to the temperature in C. The operating temperature range is from -55 to 150.

This sensor is used in the project for providing information to the breeder about the surrounded temperature of the cage place.

A temperature sensor is a thermocouple or a resistance temperature detector (RTD) that gathers the temperature from a specific source and alters the collected information into

understandable type for an apparatus or an observer. Temperature sensors are used in several applications namely HV system and AC system environmental controls, medical devices, food processing units, chemical handling, controlling systems, automotive under the hood monitoring and etc.

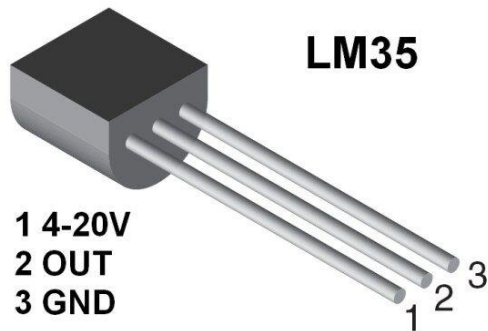


Figure 2.7 LM35 Temperature sensor

LM35 pins:

- VCC: 4-20V
- GND: Ground
- OUT: Analog output signal (0-1023)

2.2.6 Bluetooth adapter HC-06

The HC-06 module is the most important component used in the project at the first phase of triggering the system, it receives the required data from smartphone to setting up the Wi-Fi shield.



Figure 2.8 Bluetooth adapter HC-06

HC-06 Bluetooth module-Pins:

- VCC
- GND
- RX
- RT

RX is the receive pin that receive data from paired device, it is connected to the communication pin RT on the Arduino or any digital pin, and the RT is connected to pin RX on Arduino or any digital pin.

The following figure shows the architecture of the HC-06 module.

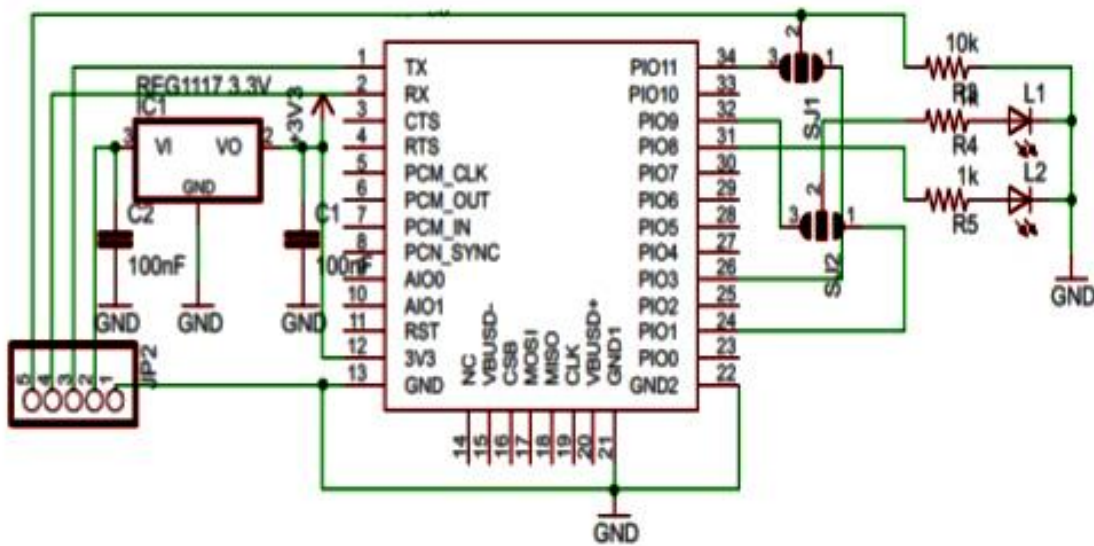


Figure 2.9 HC-06 schematic

Specifications of HC-06:

- Allows your device to both send and receive the TTL data via Bluetooth technology without connecting a serial cable to your computer.
- Works with any USB Bluetooth adapters.
- Built in antenna.
- Operating voltage: 3.3V
- Default Baud Rate: 9600
- Signal coverage: 30ft
- Item size: 4.3 1.6 0.7cm

2.2.7 Solenoid Valve

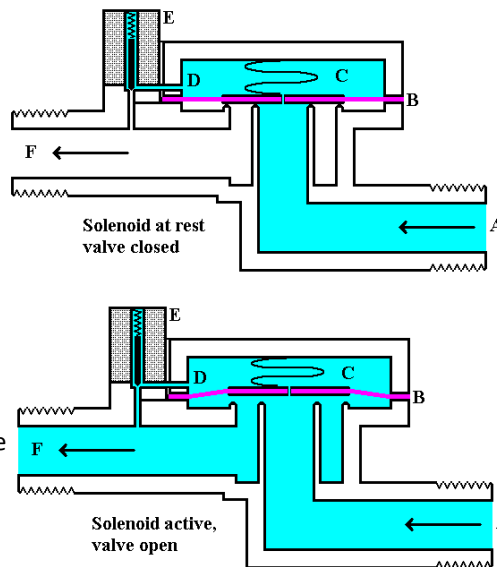
A solenoid valve is an electromechanically operated valve. The valve is controlled by an electric current through a solenoid: in the case of a two-port valve the flow is switched on or off; in the case of a three-port valve, the outflow is switched between the two outlet ports. Multiple solenoid valves can be placed together on a manifold.

Solenoid valves are the most frequently used control elements in fluidics. Their tasks are to shut off, release, dose, distribute or mix fluids. They are found in many application areas.



Figure 2.10 Solenoid Valve

Figure 2.11 describes the principle work of Solenoid Valve. At the top figure is the valve in its closed state. The water under pressure enters at **A**. **B** is an elastic diaphragm and above it is a weak spring pushing it down. The diaphragm has a pinhole through its center which allows a very small amount of water to flow through it to **C** so the pressure is equal on both sides.



- A.** Input side
- B.** Diaphragm
- C.** Pressure chamber
- D.** Pressure relief passage
- E.** Electro Mechanical Solenoid
- F.** Output side

Figure 2.11
Solenoid Valve work principle

Once the diaphragm closes the valve, the pressure on the outlet side of its bottom is reduced, and the greater pressure above holds it even more firmly closed. Thus, the spring is irrelevant to holding the valve closed.

The above all works because the small drain passage **D** was blocked by a pin which is the armature of the solenoid **E** and which is pushed down by a spring. If current is passed through the solenoid, the pin is withdrawn via magnetic force, and the water in chamber **C** drains out the passage **D** faster than the pinhole can refill it. The pressure in chamber **C** drops and the incoming pressure lifts the diaphragm, thus opening the main valve. Water now flows directly from **A** to **F**.

In our project this is used in order to make control of water, it opens when it get a 12V power to pass water into the water pot. And it will be connected to a relay in order to get a 12V.

2.2.8 Step up Chopper

Step-up chopper is an electronic device that results a DC output voltage higher than the input voltage and it is used in our project to supply the water valve 12V with an input voltage source 5V.

The step-up chopper circuit is based on *Inductor, Switch & Diode* as shown on the circuit in figure 2.12

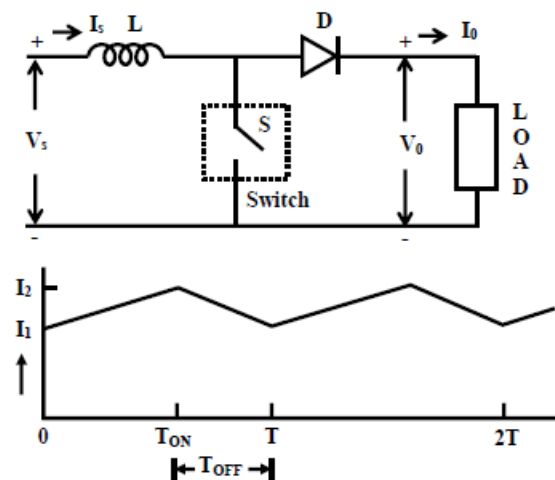


Figure 2.12 Step-up chopper circuit

- When the switch is ON, the inductor L is connected across the supply. The inductor current I rises and the inductor stores energy during the ON time of the chopper, T_{ON} .
- When the chopper is off, the inductor current I is forced to flow through the diode D and load for a period, T_{OFF} .



Figure 2.13 Chopper device

2.2.8 L298N Dual H Bridge

It allows you to control the speed and direction of two DC motors, or control one bipolar stepper motor with ease. The L298N H-bridge module can be used with motors that have a voltage of between 5 and 35V DC. When you want to guarantee that your motor will work then the L298N is used since it supplies the motor with the need current.

Figure 2.14 shows the details of the L298N H Bridge:

- Four pins considered as output pins to the connected motors called OUT1, OUT2, OUT3 and OUT4. Each motor is connected to two pins.
- Three power pins: GND, 5V and 12V. You connect 5 or 12 but not the two.
- Input pins as in the following table

○ Table 2.2 L298N Input pins

Pin	Name	Description
1	ENA	Input to enable Motor A
2	IN1	Input to control Motor A
3	IN2	Input to control Motor A
4	IN3	Input to control Motor B
5	IN4	Input to control Motor B
6	ENB	Input to enable Motor B

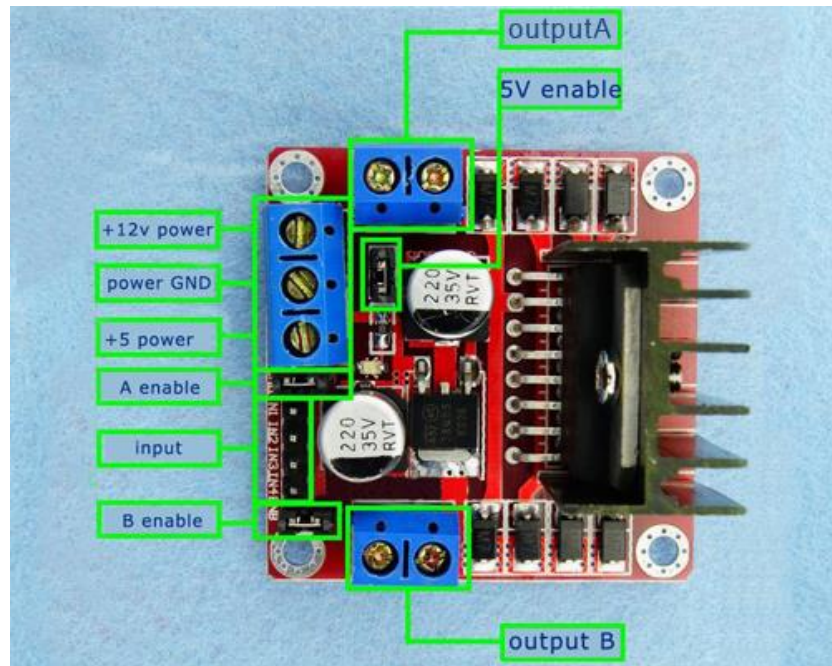


Figure 2.14 L298N Dual H Bridge

2.2.9 Light source (Led)

This is complementary to the LDR sensor which helps in food shortfall detection as mentioned previously in the LDR section 2.2.3

2.2.10 Stepper motor

Stepper Motor is a motor controlled by a series of electromagnetic coils. The center shaft has a series of magnets mounted on it, and the coils surrounding the shaft are alternately given current or not, creating magnetic fields which repulse or attract the magnets on the shaft, causing the motor to rotate.

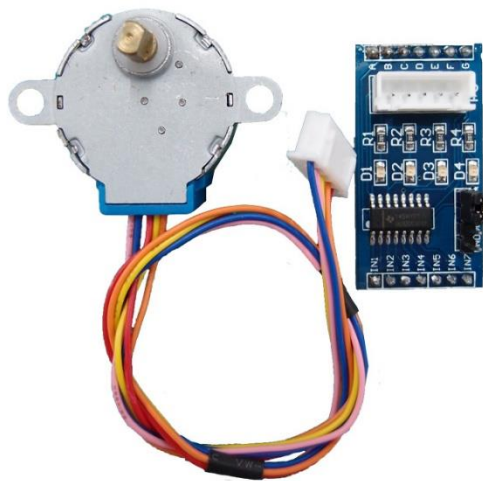


Figure 2.15 Stepper motor

Stepper motor needs a driver to rotate which is ULN2003-Driver as shown in figure 2.16

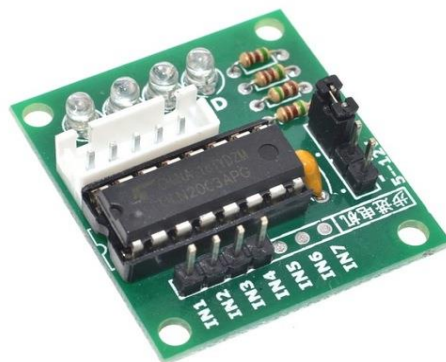


Figure 2.16 ULN2003-Driver

Figure 2.17 shows the connections of stepper motor and ULN2003-Driver. The stepper motor has five pins:

- Four pins called coils.
- VCC pin.

And the ULN2003-Driver has the following pins:

- Input pins **IN1, IN2, IN3, IN4, IN5, IN6** and **IN7**. The first four pins are used to be connected to Arduino as shown in figure 2.15
- Output pins: **OUT1, OUT2, OUT3** and **OUT4** connected with stepper.
- Control pins: GND and VCC (5V or 12V).

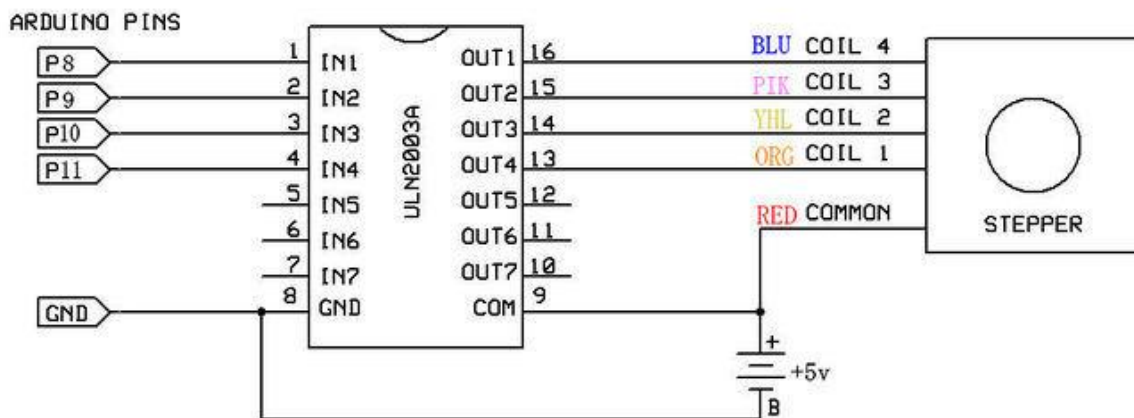


Figure 2.17 Stepper motor-ULN2003-Arduino connections

Stepper motor is used in our project in order to make control of adding food operation, it will open and close a simple piece that considered as a gate for passing food to the food pot.

2.2.11 Fan

The fan will be used in order to get rid of the peels produced from eating seeds. It will be placed almost above the food pot. It can be triggered in different times. For example the user can trigger it every 10 minutes.

2.2.12 Wi-Fi Shield

The Arduino Wi-Fi Shield allows an Arduino board to connect to the internet using the 802.11 wireless specification (Wi-Fi). It is based on the HDG204 Wireless LAN 802.11b/g System in-Package. An AT32UC3 provides a network (IP) stack capable of both TCP and UDP. The Wi-Fi shield connects to an Arduino board using long wire-wrap headers which extend through the shield.

This shield will be the adapter between the smartphone and the Arduino. It will work as a server and the smartphone as a client.



Figure 2.18 Wi-Fi Shield

There is an onboard micro-SD card slot, which can be used to store files for serving over the network. It is compatible with the Arduino Uno and Mega. The onboard micro SD card reader is accessible through the SD Library.

The shield contains a number of informational LEDs:

- L9 (yellow): this is tied to digital pin 9
- LINK (green): indicates a connection to a network
- ERROR (red): indicates when there is a communication error
- DATA (blue): indicates data being transmitted/received

Chapter 3

System Design

3.1 Overview

This chapter introduces the theoretical background of the project, provides a description of the methodologies of how our project will meet the needs of birds, and some description of hardware and software component used in the system, finally some description of design specification and constrains.

3.2 Theoretical background

The problem with the birds' care is that it needs periodic monitor, effort and time from owner. Therefore, the system has to behave like human in order to achieve the comfort for the owner by determining what and when to do. In addition, birds care in terms of feeding, surrounded temperature and clearing the cage base using the sensors and some motor.

3.3 Methodology

Our System consist of microcontroller and set of sensors that receive the configurations from the user through his mobile via Internet, and according to these configurations the Arduino will trigger the sensors, motors, and other elements to run the system as required.

The following steps are demonstrating the mechanism of system work:

- We will build mobile application that view all options to the user, there will be more than one mode for some options that can be controlled (ex: add water manually or automatically every 1 hour). When the user selects an option with specific mode the mobile application encodes and sends commands to the microcontroller via internet Connection.
- After the microcontroller receives and decode commands, it edits its default settings according to the new commands.
- The microcontroller reads data from sensors and make decisions to control the system depending on current settings which the user has chosen.
- When the mobile user become offline with the system, then it will be still use the last setting it was set from the user.

3.4.2 LDR Sensor

The circuit in figure 3.2 shows the connections of the LDR sensors. It is used to get the food status in the cage based on a light source, it indicates if there is a need to add food or not. This sensor measures the intensity of the light and out an analog value, so one leg of each sensor is connected to an analog pin in the Arduino as shown in the following figure. The first one is connected to **A2** and the second to **A3**

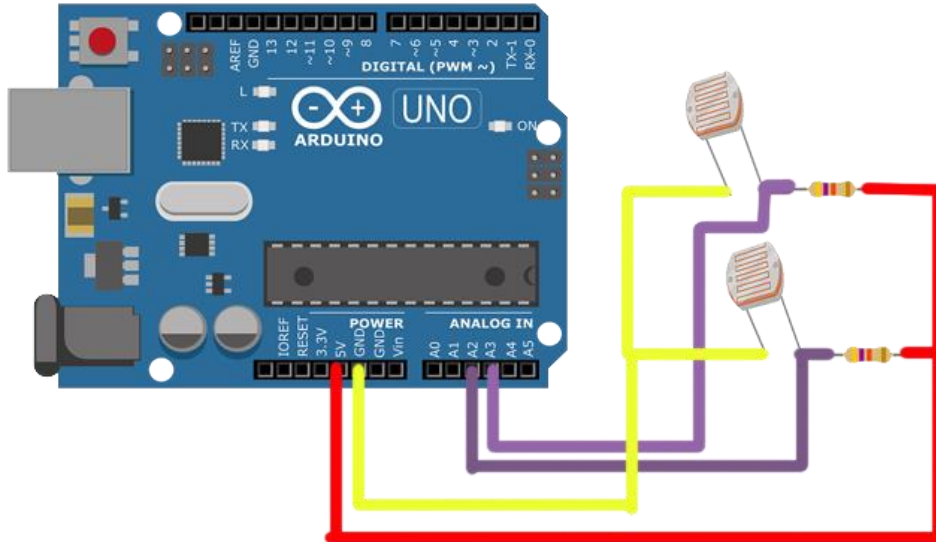


Figure 3.2 LDR sensor connections

3.4.3 PIR Motion sensor

The circuit in figure 3.3 below shows the connections of the Motion sensor (PIR), this sensor is used to detect the last motion of the bird and send it to the user mobile application when the user requests the system status. PIR produces a digital output: 1 if it detects a motion and 0 otherwise. So the output pin of the PIR will be connected to a digital pin on the Arduino and it was connected to pin **2** as shown in figure 3.3.

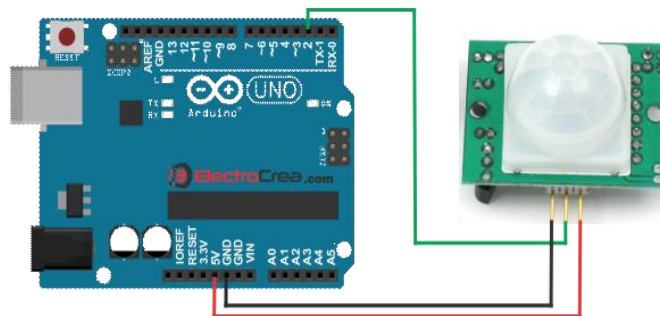


Figure 3.3 Motion sensor circuit

3.4.4 LM35 Temperature Sensor

The circuit in figure 3.4 below shows the connections of the Temperature sensor (LM35), this sensor is used to measure the temperature of the environment surrounding the system and sends it to the user mobile application when he requests the system status. The output pin of LM35 is an analog pin so it will be connected to an analog pin on the Arduino. It was connected to pin **A0**.

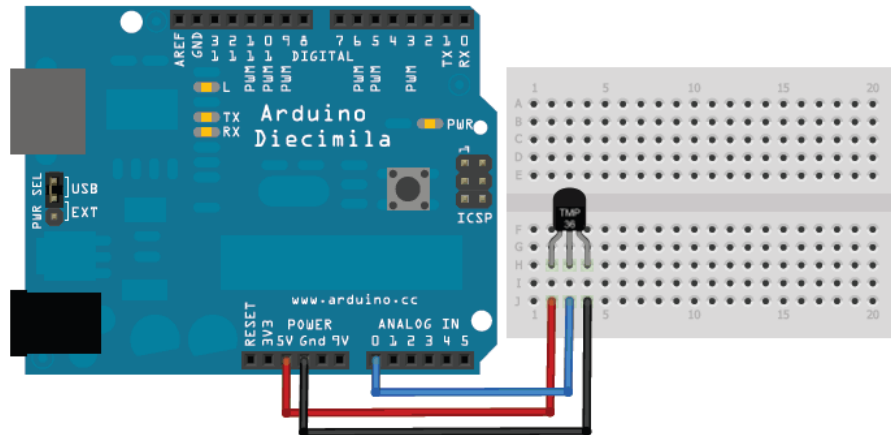


Figure 3.4 Temperature sensor LM35 circuit

3.4.5 Bluetooth HC-06

The circuit shown in figure 3.5 shows the connections of the Bluetooth module HC-06. This module allow user to send the WI-FI Configuration information using his mobile application and therefore allow the Wi-Fi shield to connect to the desired network in the correct format. The RX and TX pins on the HC-06 usually connected to pins **0** and **1** on the Arduino, but it can be connected to another pins, and in our case the two pins are connected to pins **10** and **11** with some configurations in the Arduino code.

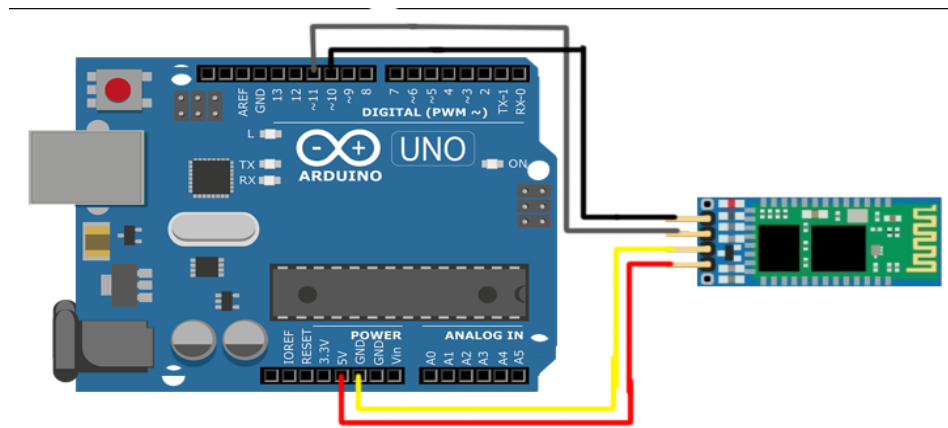


Figure 3.5 Bluetooth module circuit

3.4.6 Water Valve

The circuit shown in figure 3.6 shows the connections of the water valve. That allows the system to add water when needed. We used a relay to pass 5V to **Chopper** which in turn supplies the water valve enough voltage since it requires 12V. Relay has three input pins: VCC, GND and Data. Data pin is connected to the pin 3 in the Arduino, the output pins of the relay (green color) is connected to one of the pins in water valve.

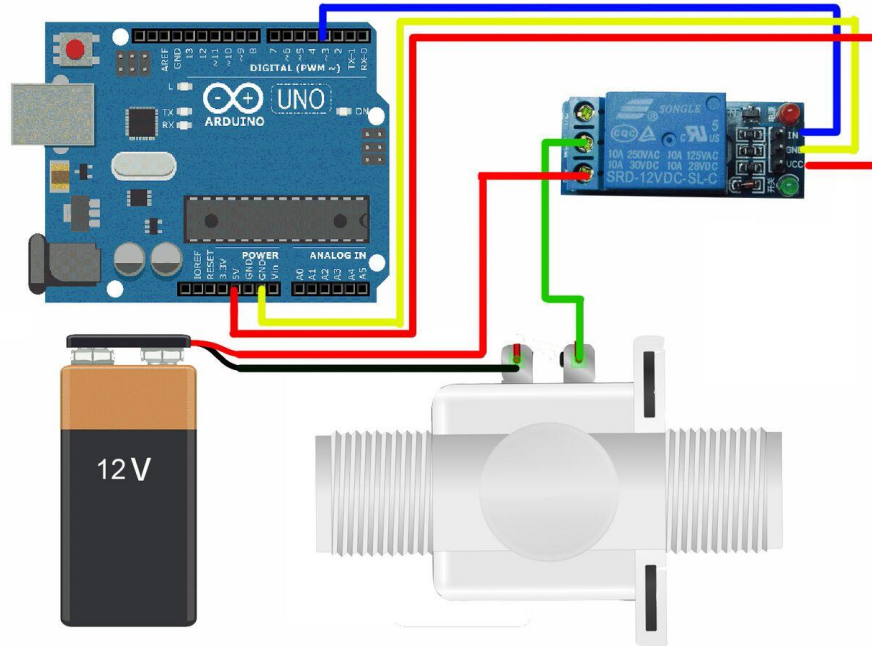


Figure 3.6 Solenoid Valve circuit

3.4.7 RN-171 WI-FI shield

This module shown in figure 3.7 considered as a server that receives the requests from the user mobile application and sends replay or do some functions.



Figure 3.7 Wi-Fi shield

The Figure below shows the connection of Wi-Fi shield to the Arduino UNO. It has pins that makes it able to be placed on the Arduino pins. Physically all wires of the other hardware components will be connected on the Wi-Fi Shield but logically they are connected to the Arduino.



Figure 3.8 Wi-Fi shield connection

3.4.8 Stepper Motor

The circuit shown in figure 3.9 shows the connections of the stepper motor. Stepper motor needs a driver which called ULN2003, and this driver will be connected the Arduino, it has two control pins and four data pins which are connected to the pins **8 9 12 13**.

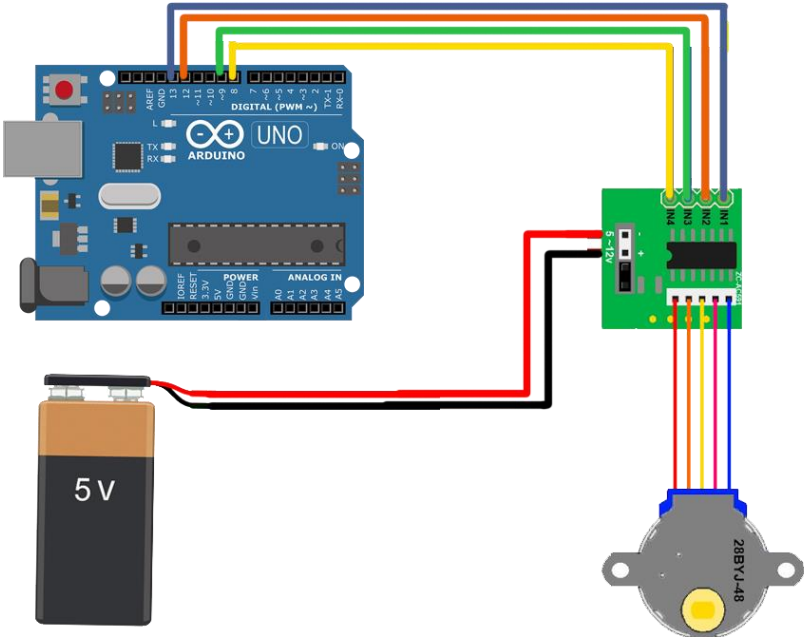


Figure 3.9 Stepper motor connections

3.4.10 Circuit Diagram

The following figure shows the whole system integrated together.

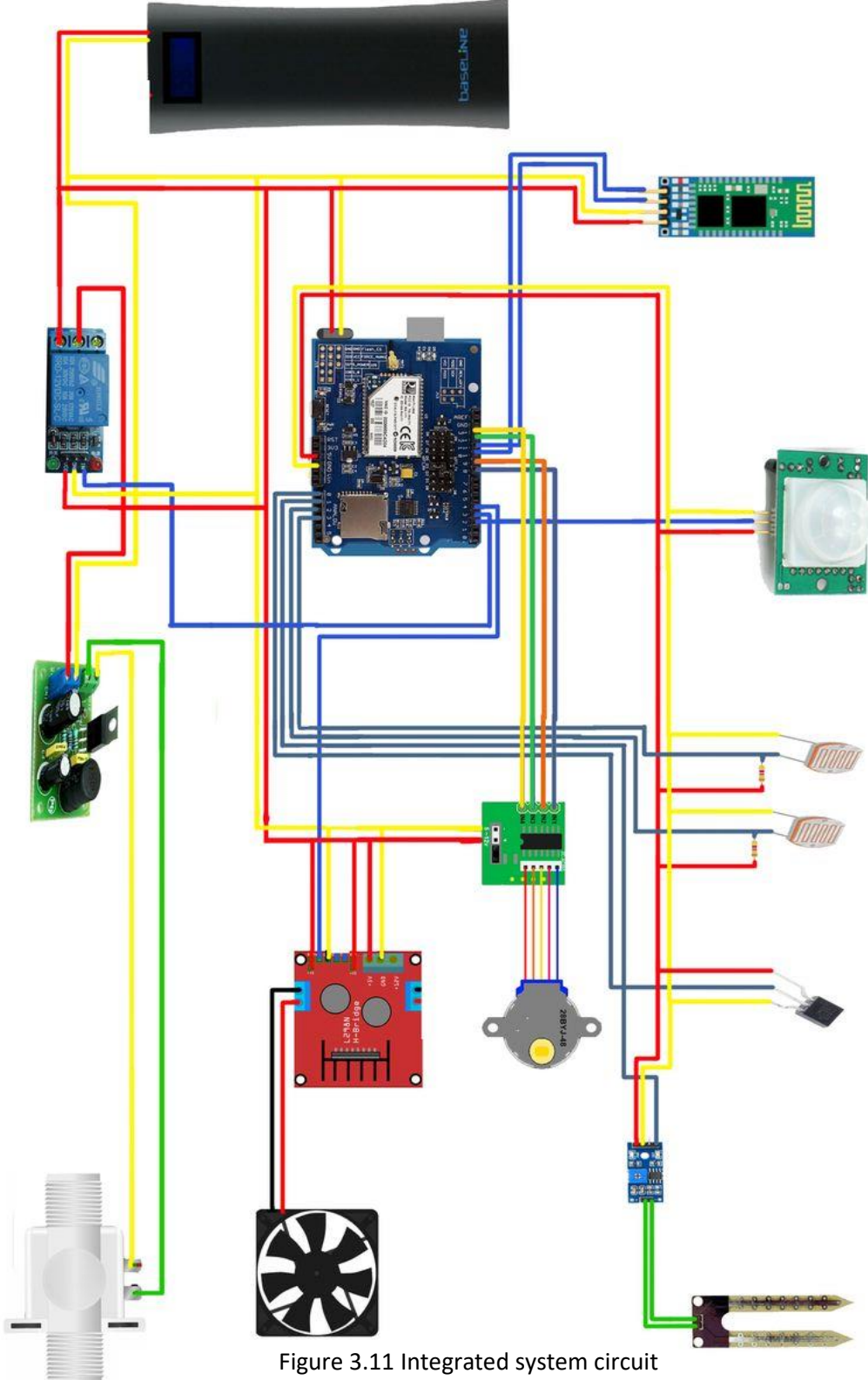


Figure 3.11 Integrated system circuit

3.5 Block Diagram

Figures 3.12a and 3.12b show how the user interact with the system and how the system itself interacts together in general diagram.

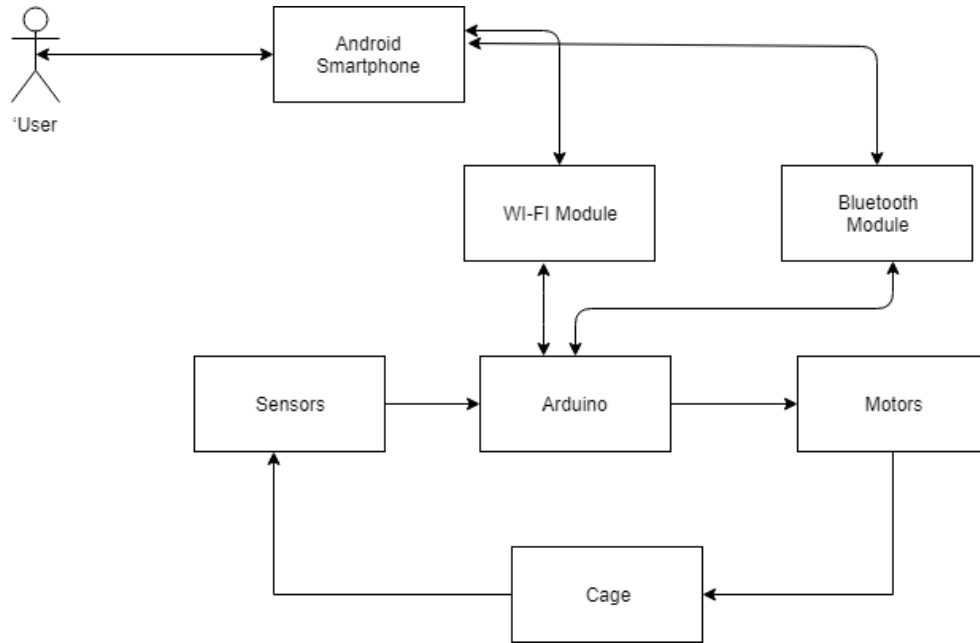


Figure 3.12a Block diagram 1

- User: control the system and receive information from Arduino using his smartphone.
- Smartphone: a user interface connects between user and the system.
- Wi-Fi module: an adapter between the smartphone and the Arduino.
- Bluetooth module: setting the setting for the Wi-Fi module.
- Arduino: a microcontroller that receives data from sensors, Bluetooth and WIFI module to make processing based on them.
- Soil-Moisture: sensor that checks the water level.
- Motion sensor: detects if there is any motion in the cage.
- LDR sensor: indicates if there is need to add more food or not.
- Motors: add water, food and run fan.

This figure shows the general block diagram of the whole system.

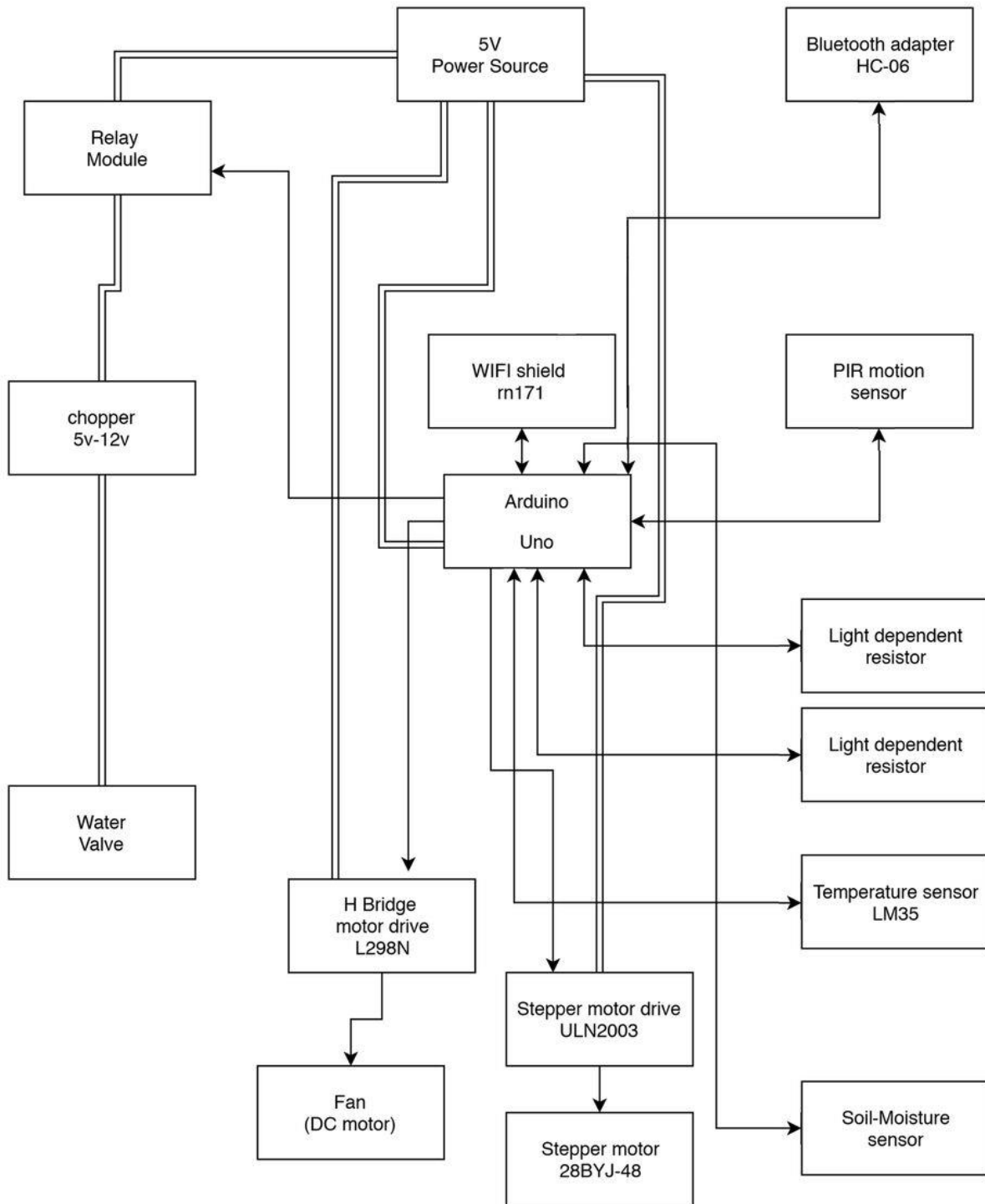


Figure 3.12b Block diagram 2

3.6 System Flowchart

This section shows the flowchart of the three basic sensors (LDR, Soil-Moisture and motion sensor).

3.6.1 Soil-Moisture flowchart

Figure 3.13 shows the work cycle of the soil-moisture sensor, it measures the humidity percentage and then sends the value to the Arduino which in turn decides if there is a need to add water or not based on the read value after comparing it with a stored value called threshold value, so if the value is less than threshold then the Arduino direct the system to add water otherwise it reads another value from the sensor after period of time.

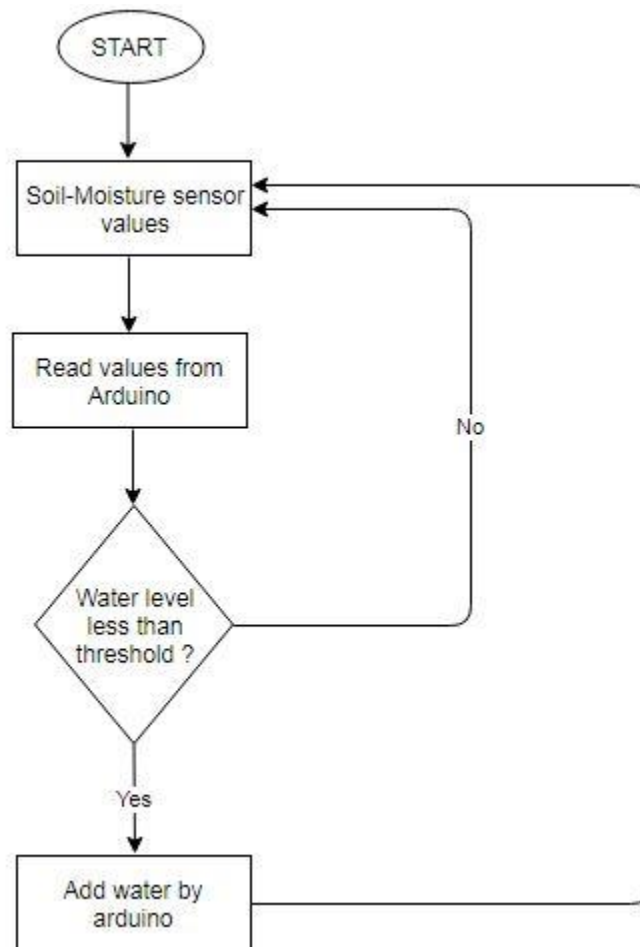


Figure 3.13 Soil-Moisture sensor flowchart

3.6.2 LDR Flowchart

Figure 3.14 shows the work cycle of the LDR sensor, it measures the light intensity and then sends the value to the Arduino which in turn decides if there is a need to add food or not based on the read value after comparing it with a stored value called threshold value, so if the value is less than threshold then the Arduino direct the system to add food otherwise it reads another value from the sensor after period of time.

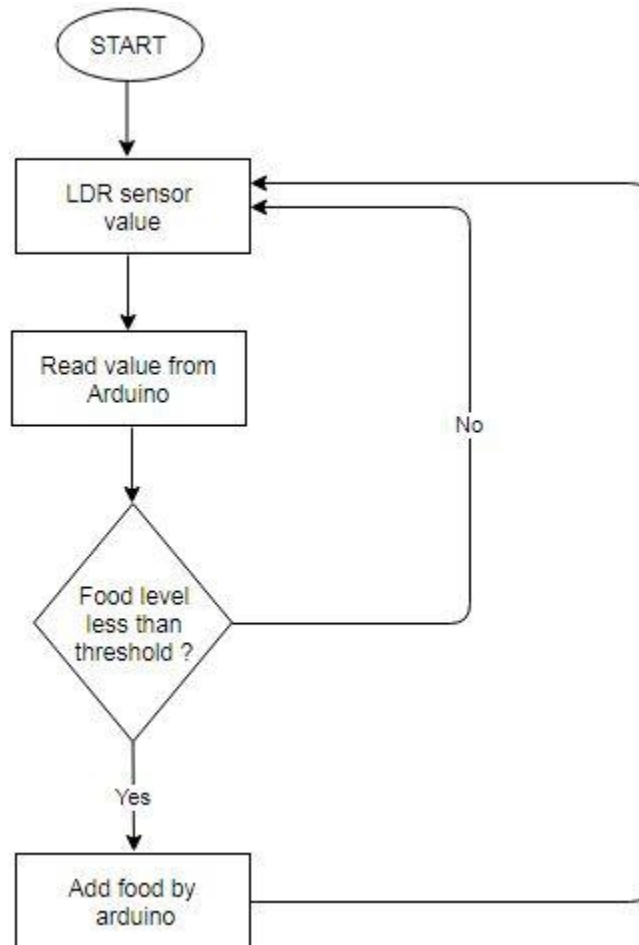


Figure 3.14 LDR sensor flowchart

3.6.3 Motion sensor flowchart

Figure 3.15 shows the work cycle of the PIR-motion sensor, it detects the bird motion and then sends the value to the Arduino, if the read value was 1 which means there is a motion it reread new value until it reads 0. If the value was 0, the Arduino starts counting time while the value is 0, finally if the counting time reaches more than the threshold (9 hours) the system alerts the user that there were no motion for more than 9 hours.

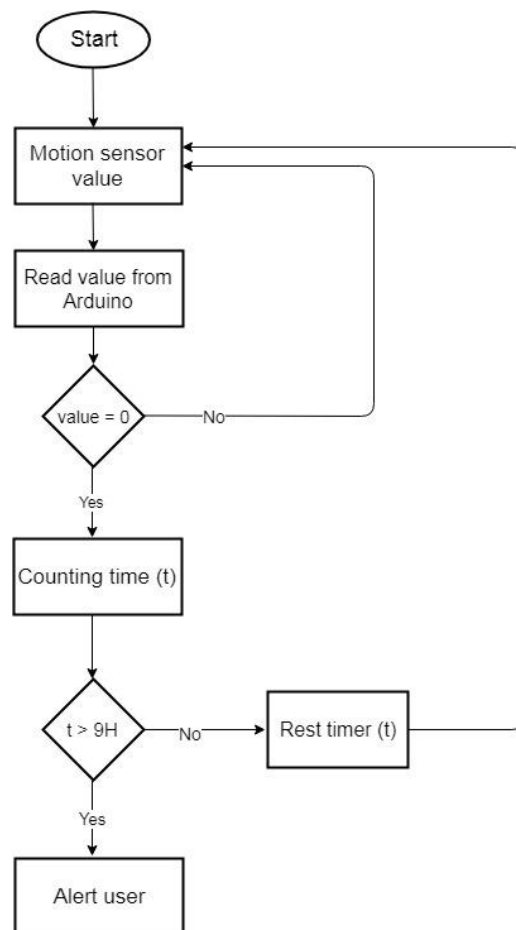


Figure 3.15 Motion sensor flowchart

Chapter 4

Hardware & Software Implementation

4.1 Overview

In this chapter, we describe the implementation of software and hardware components that are used in our project, such as the mobile OS and IDE's.

4.2 Operating System for mobile

There were many operating systems for mobile to be used to handle the software and hardware resources of our system such as Android and IOS. We have chosen android OS rather than IOS OS because it is the world's familiar mobile operating system. In addition, it is easy to be used and work with an Android device, because it is working perfectly with Windows OS and it needs only to download Android studio program with unity android support.

4.3 Software Implementation Tools

This section will provide some information about the main programs and software technologies used in our project.

4.3.1 Android studio IDE

Android Studio is the official Integrated Development Environment (IDE) for Android app development we have used this IDE to develop our application.

Our application consists of four user interfaces, which are:

- Searching Bluetooth devices interface **figure 4.1**. After searching and making a successfully connection with the desired Bluetooth module, the application moves to the Wi-Fi Network configurations.
- Wi-Fi Network configurations interface **figure 4.2**. Consists of a listView that views all available networks and an EditText to enter the password, and a button to create connection and send the name and the password of the selected network to the Arduino in order to make the Wi-Fi Shield create a connection with network.

- Server settings interface **figure 4.3**. In this interface the user has to enter the external IP of the Wi-Fi Shield and the port number opened in the router that the Arduino is connected with to make connection between the mobile and the server (Arduino).
- System Control interface **figure 4.4**. This is the last interface, after a successfully settings up the user is able to start controlling the cage by the interface shown in figure 4.4 which has the functionality controls of the overall system.

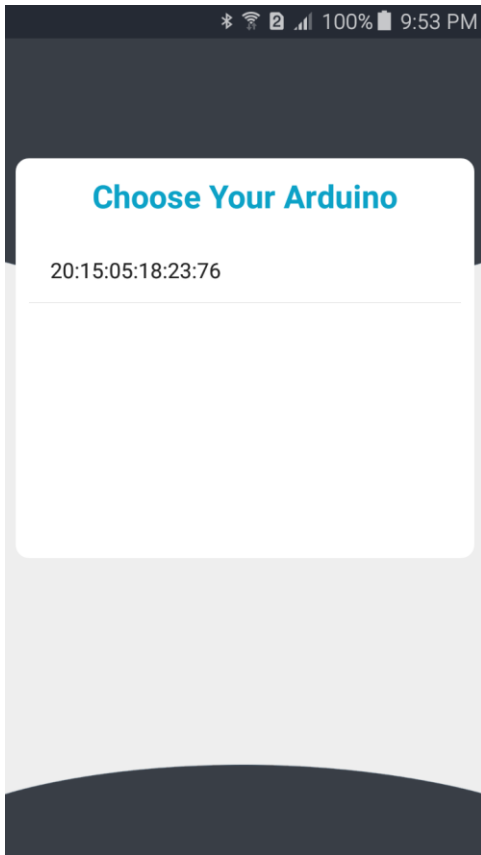


Figure 4.1 Searching Bluetooth device interface

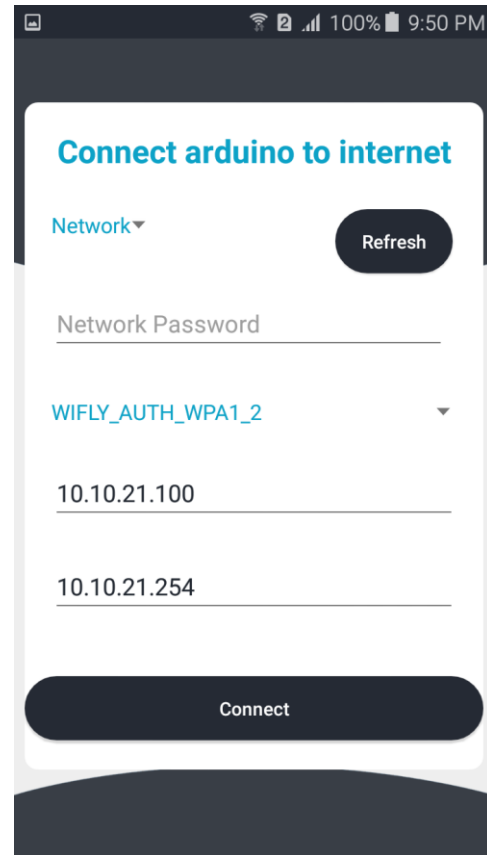


Figure 4.2 Wi-Fi Network configurations interface

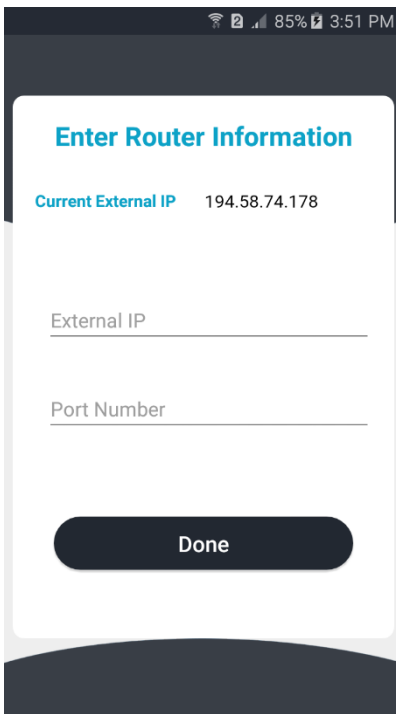


Figure 4.3 Server settings interface

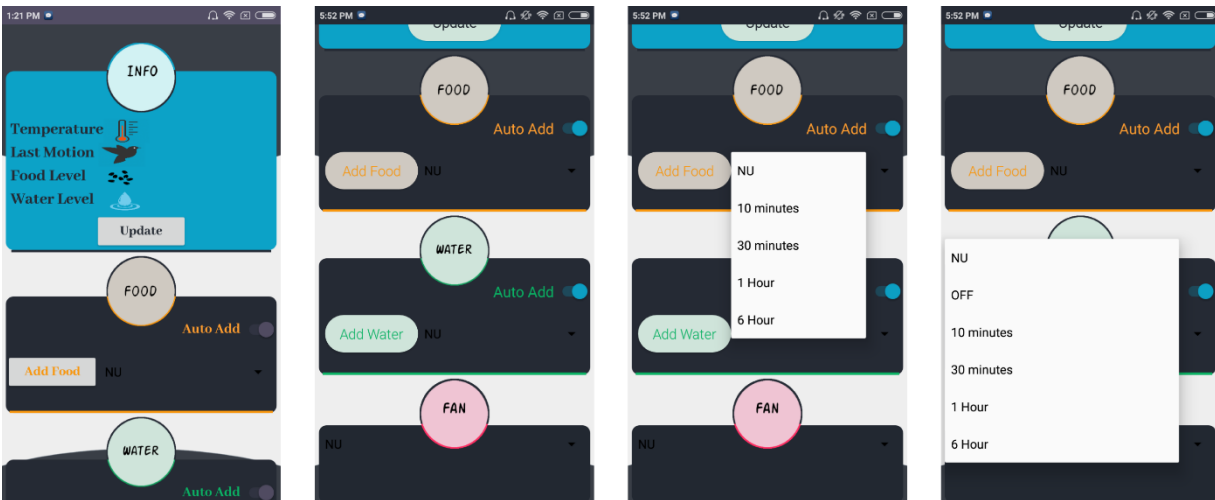


Figure 4.4 System Control interface

4.3.2 Arduino Application

This application based on C/C++ programming languages and we used it to program the Arduino microcontroller and all connected sensors and Wi-Fi Shield.

4.4 Hardware Implementation

This section provides some information about the hardware implementations done through our project.

4.4.1 Motion sensor configuration

For the system, we have used Motion sensor to detect if there is any motion in the cage or not, by this sensor work by two slots when the sensor is idle the slots have the same value of IR, when there is a movement it causes a difference change between IR of the two slots.

Figure 4.1 shows the principle of the sensor working.

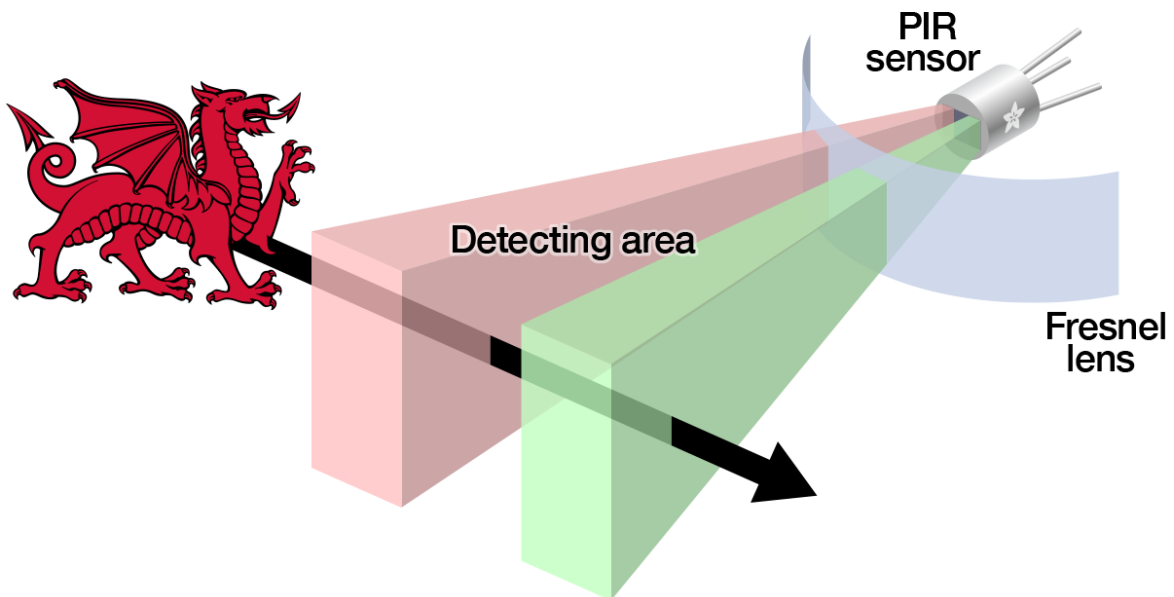


Figure 4.5 Motion working principle

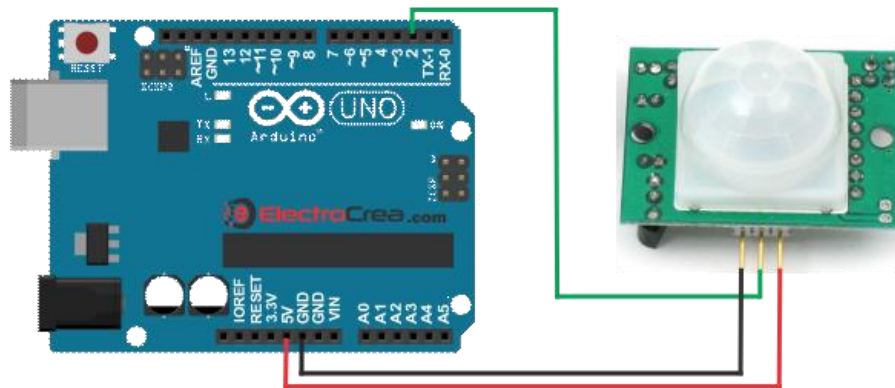


Figure 4.6 Motion sensor connection

Figure 4.7 shows the connections of the motion sensor in a schematic diagram.

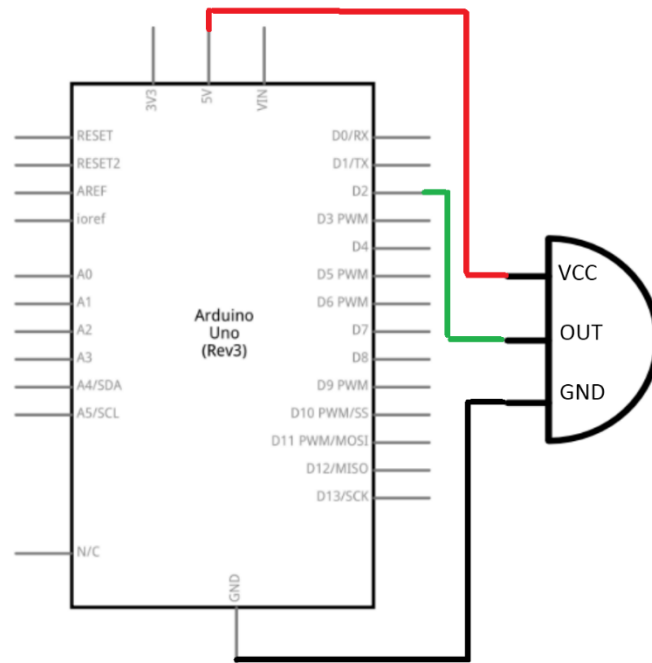


Figure 4.7 Motion sensor schematic diagram

4.4.2 Grove Soil-Moisture sensor configuration

This sensor is used to read the percentage of the water in the pot. The two probes are acting as a variable resistor – more water in the soil means better conductivity and results in a lower resistance and a higher SIG out.

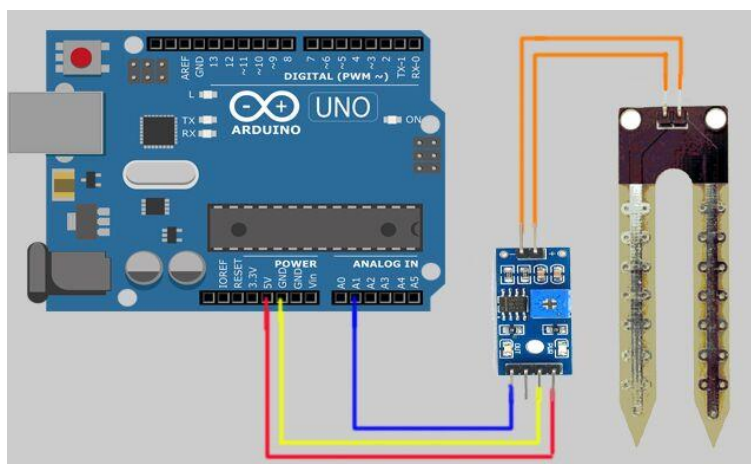


Figure 4.8 Grove Soil-Moisture sensor connection

Figure 4.9 shows the connections of the Soil-Moisture sensor in a schematic diagram.

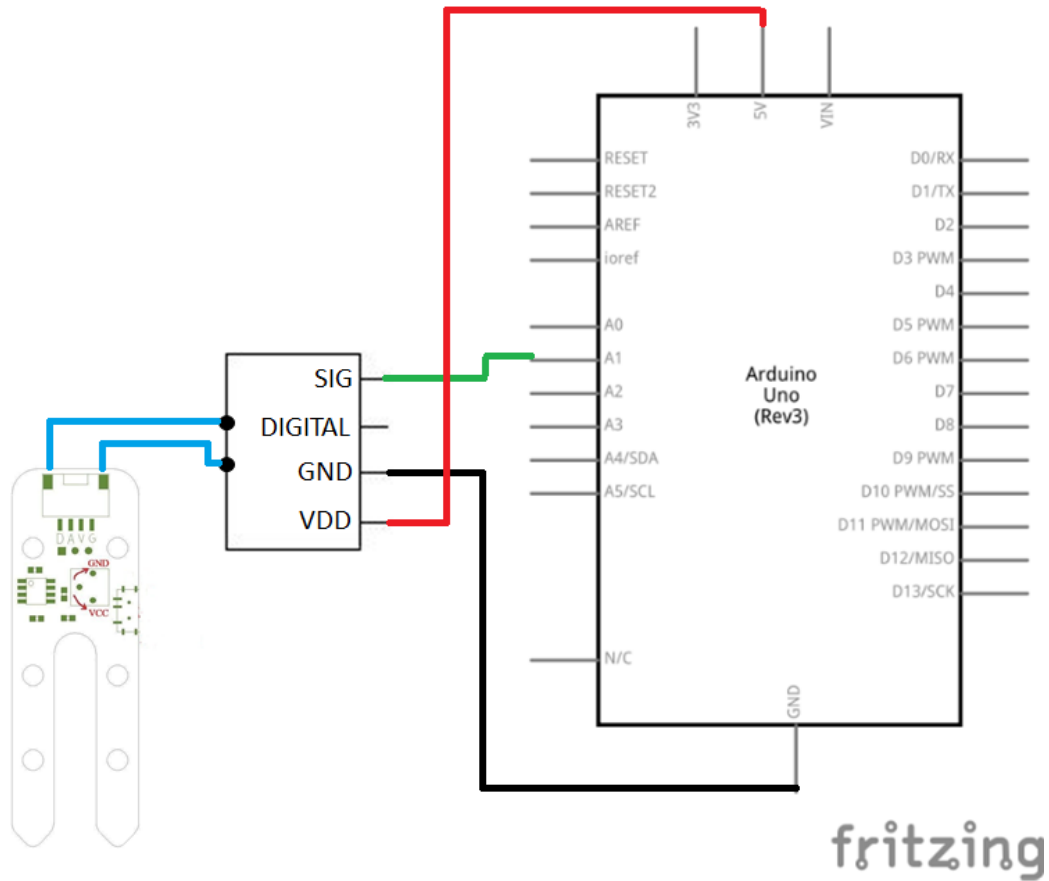


Figure 4.9 Grove Soil-Moisture sensor schematic diagram

4.4.3 Temperature sensor configuration

This sensor is used as an informative sensor, it sends a notification to the android app about the average temperature degree.

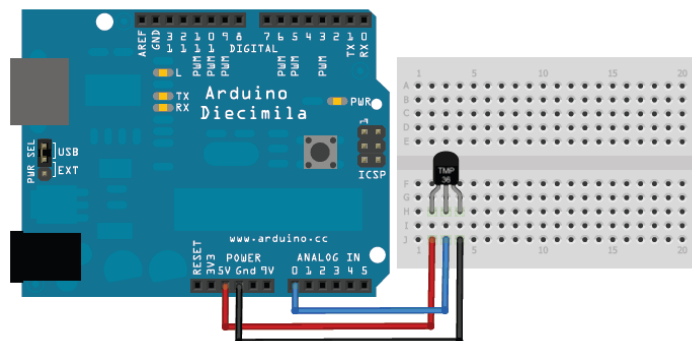


Figure 4.10 Temperature sensor connection

Figure 4.11 shows the connections of the LM35 temperature sensor in a schematic diagram.

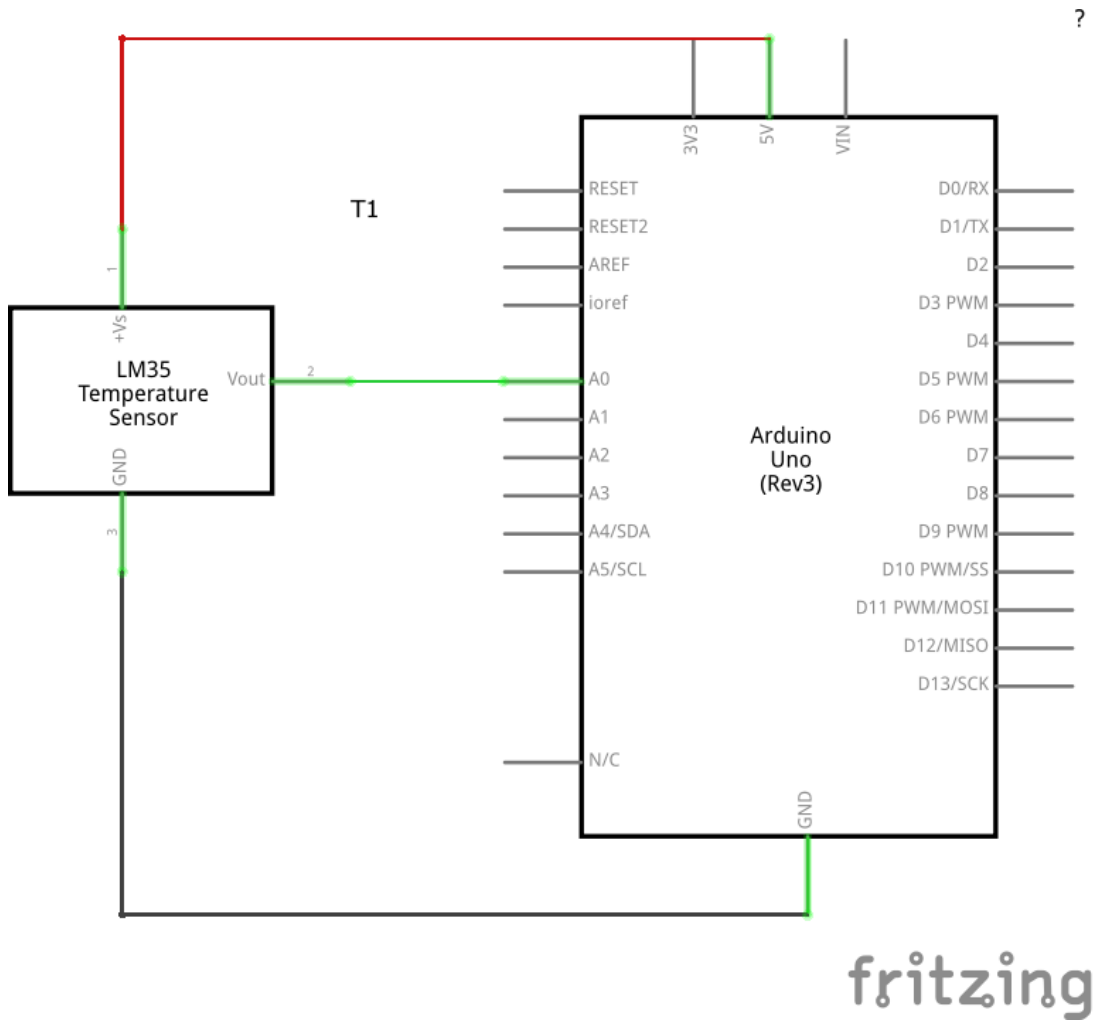


Figure 4.11 Temperature sensor schematic diagram

4.4.4 Stepper motor configuration

Figure 4.12 shows the stepper motor with its connections to the Arduino. This motor related in the addition food operation.

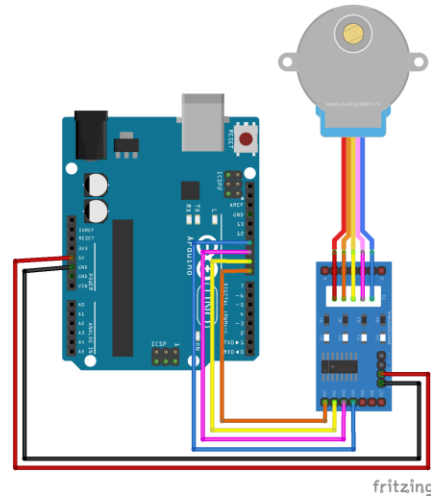


Figure 4.12 Stepper motor connection

The following figure shows all connections of the stepper motor and the ULN2003 driver to the Arduino in a schematic diagram.

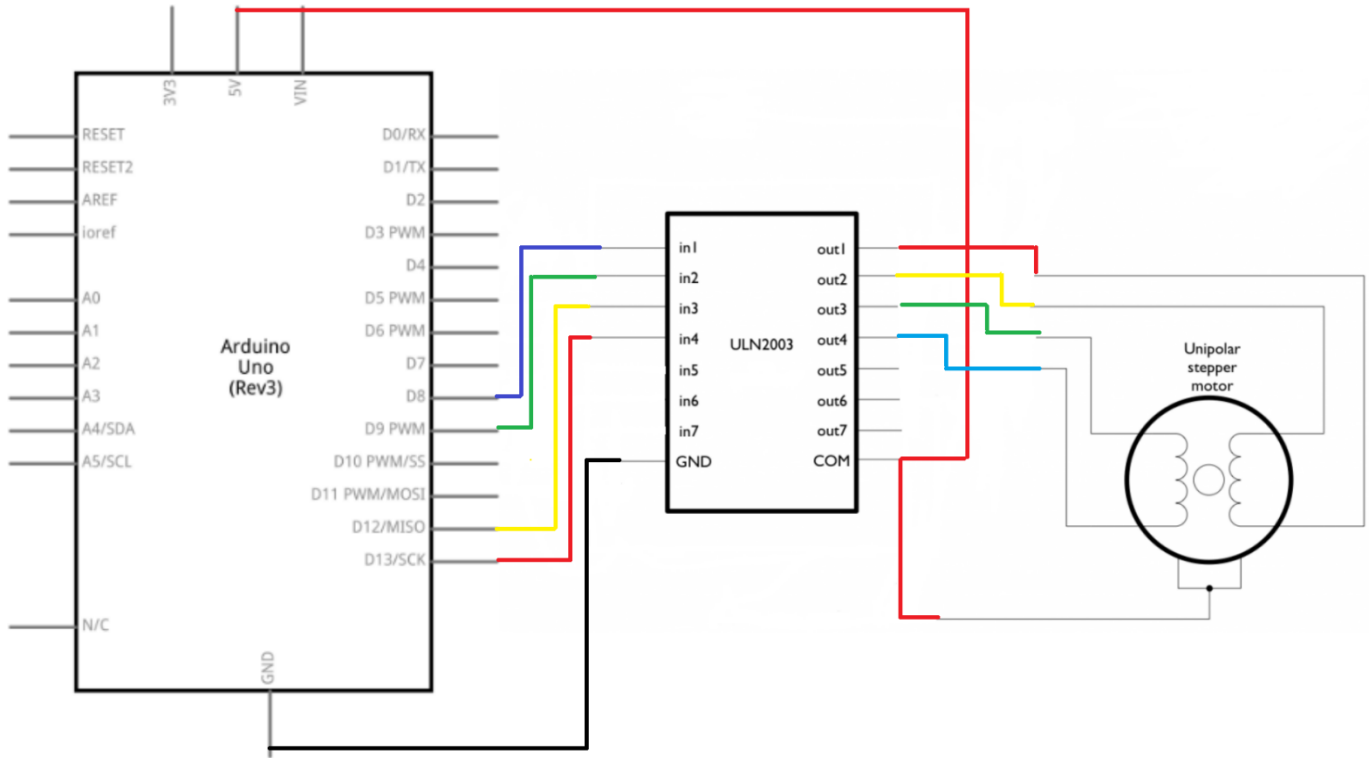


Figure 4.13 Stepper motor schematic diagram

4.4.5 Bluetooth adapter HC-06 configuration

The Bluetooth adapter is used to set up the Wi-Fi shield settings by sending the port number and the IP address from mobile to Bluetooth adapter. Figure 4.14 shows connections of HC-06.

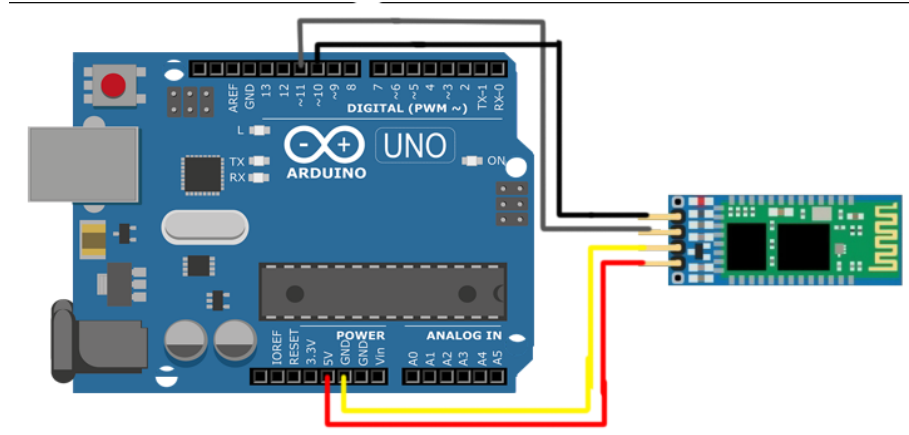


Figure 4.14 HC-06 Bluetooth adapter connection

The following figure shows the connections of the Bluetooth adapter in a schematic diagram.

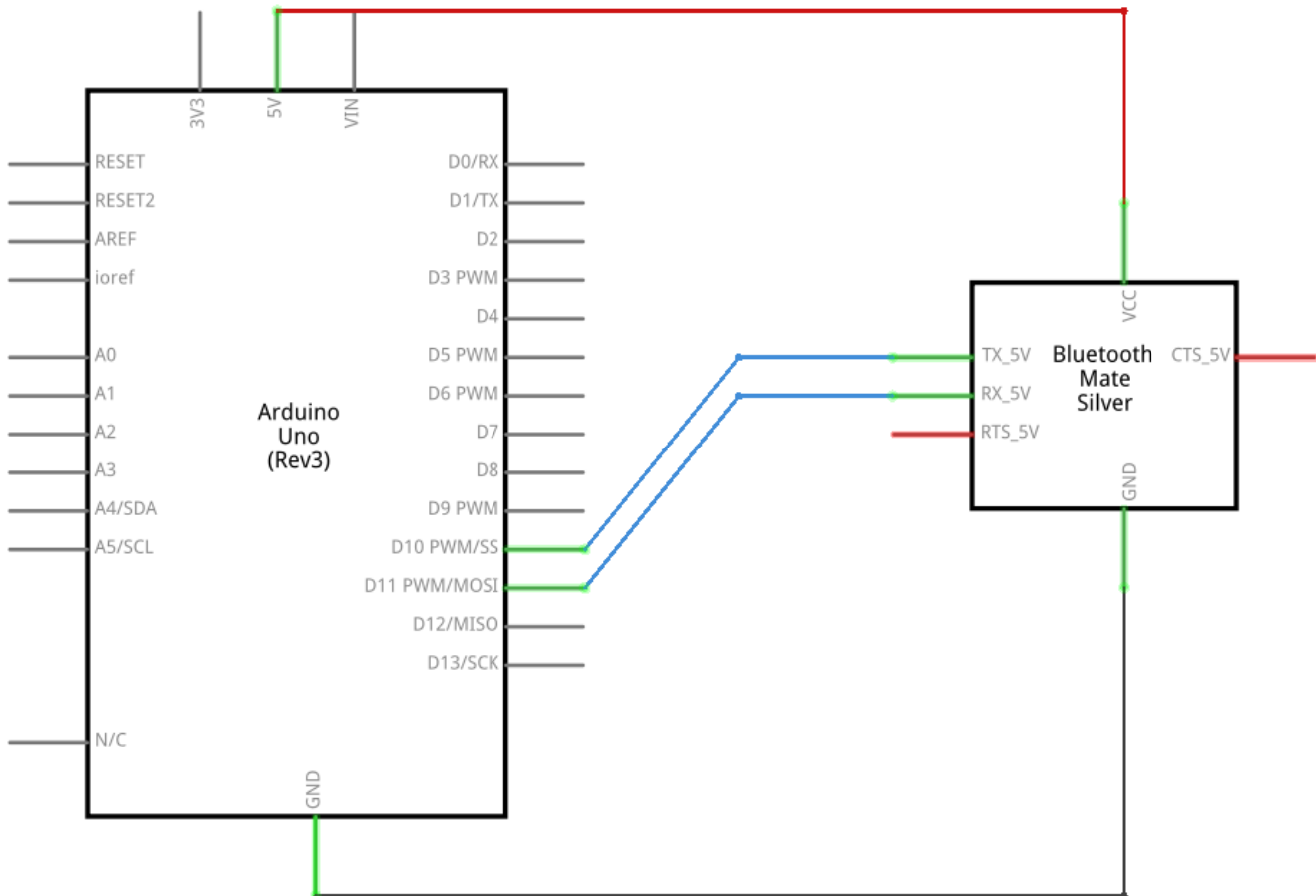


Figure 4.15 HC-06 Bluetooth adapter schematic diagram

4.4.6 LDR sensor configuration

LDR sensors are used to make decision of adding food to the food pot.

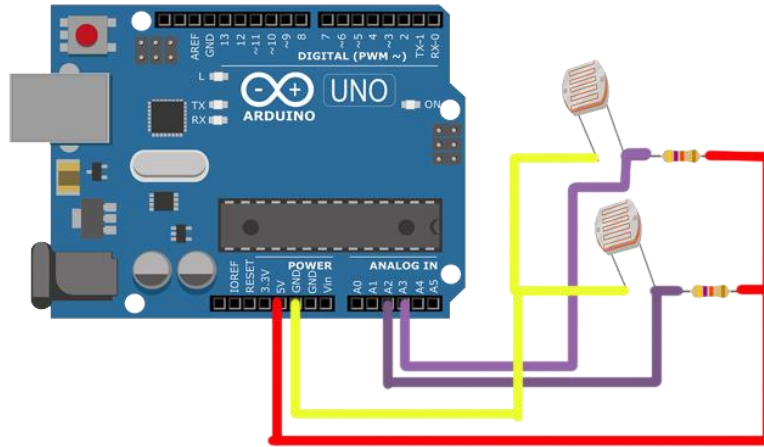


Figure 4.16 LDR sensors connection

Figure 4.17 shows the connections of the LDR sensors in a schematic diagram.

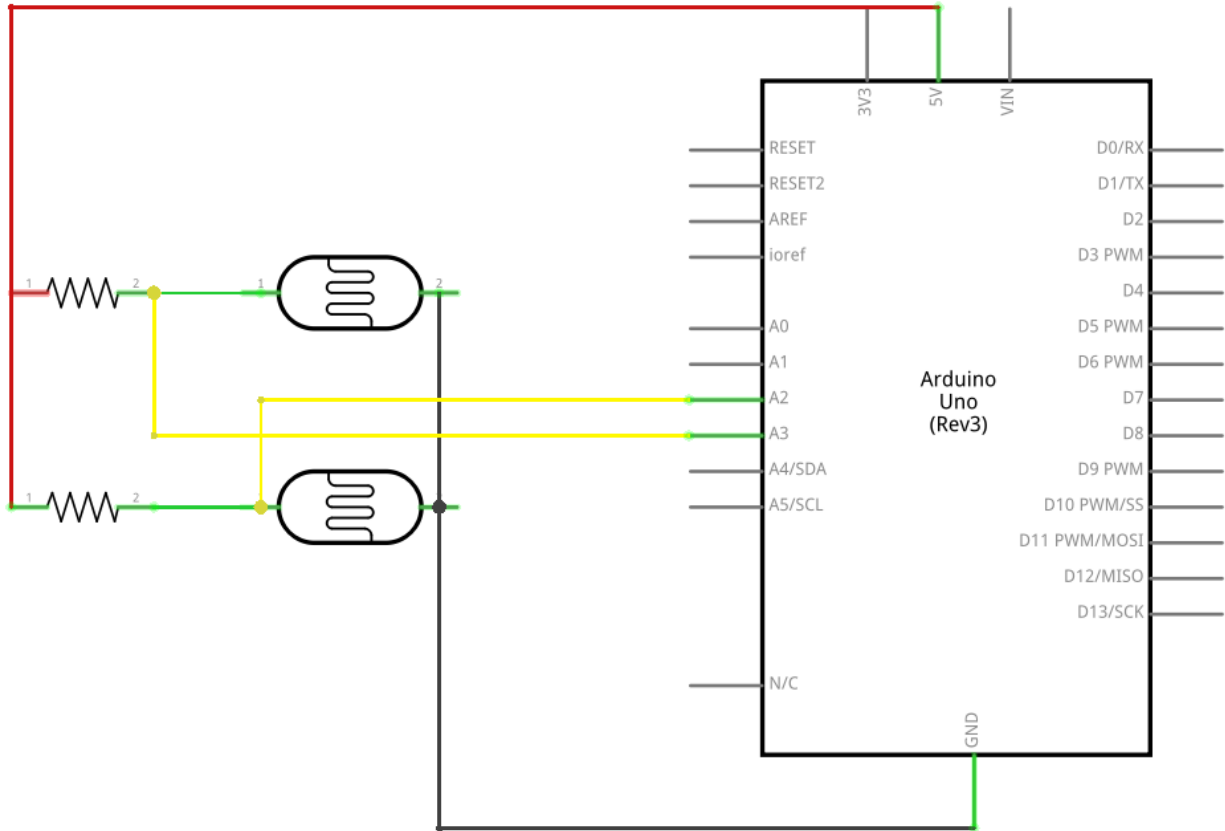


Figure 4.17 LDR sensors schematic diagram

4.4.7 Solenoid Valve configuration

It is used to pass water into the water pot based on a diaphragm that opens and closes according to the water pressure and the armature of the solenoid as shown in figure 4.10

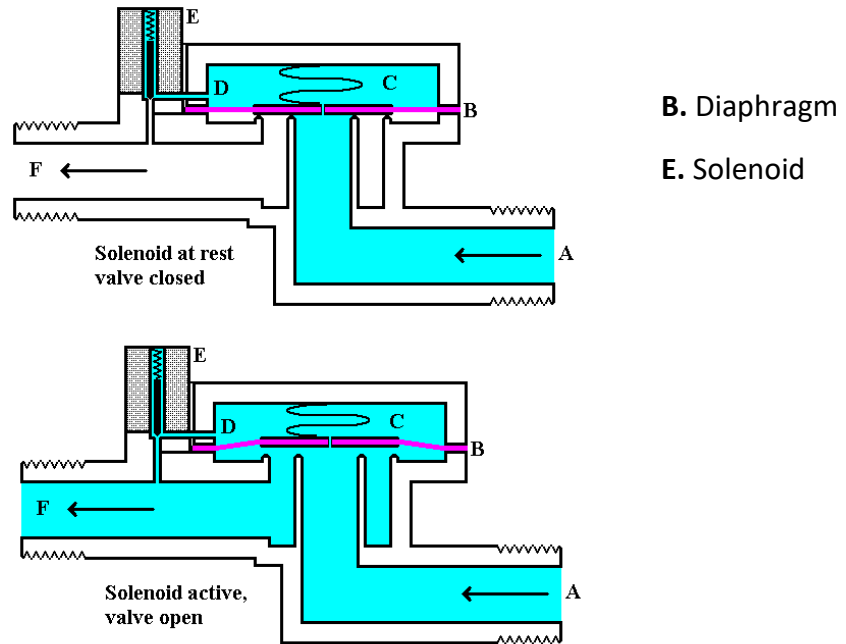


Figure 4.18 Solenoid Valve working principle

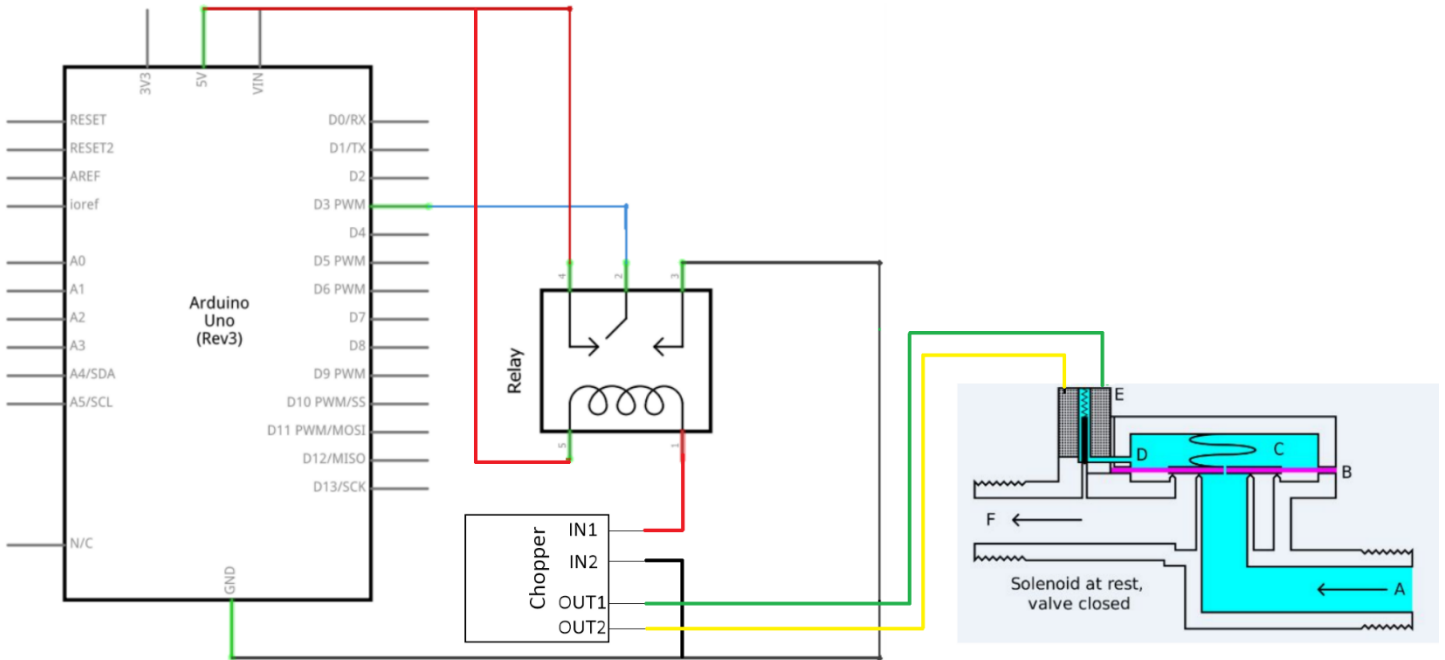


Figure 4.19 Solenoid Valve schematic diagram

4.4.8 Fan connections

Since the food (seeds) produces peels it should be getting rid of the peels, so we used a fan to do this job. And the two figures below show the configuration of the fan.

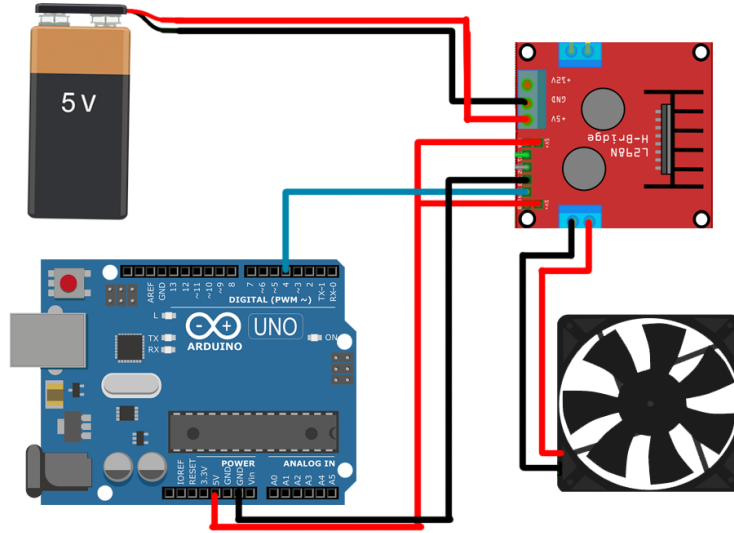


Figure 4.20 Fan connections

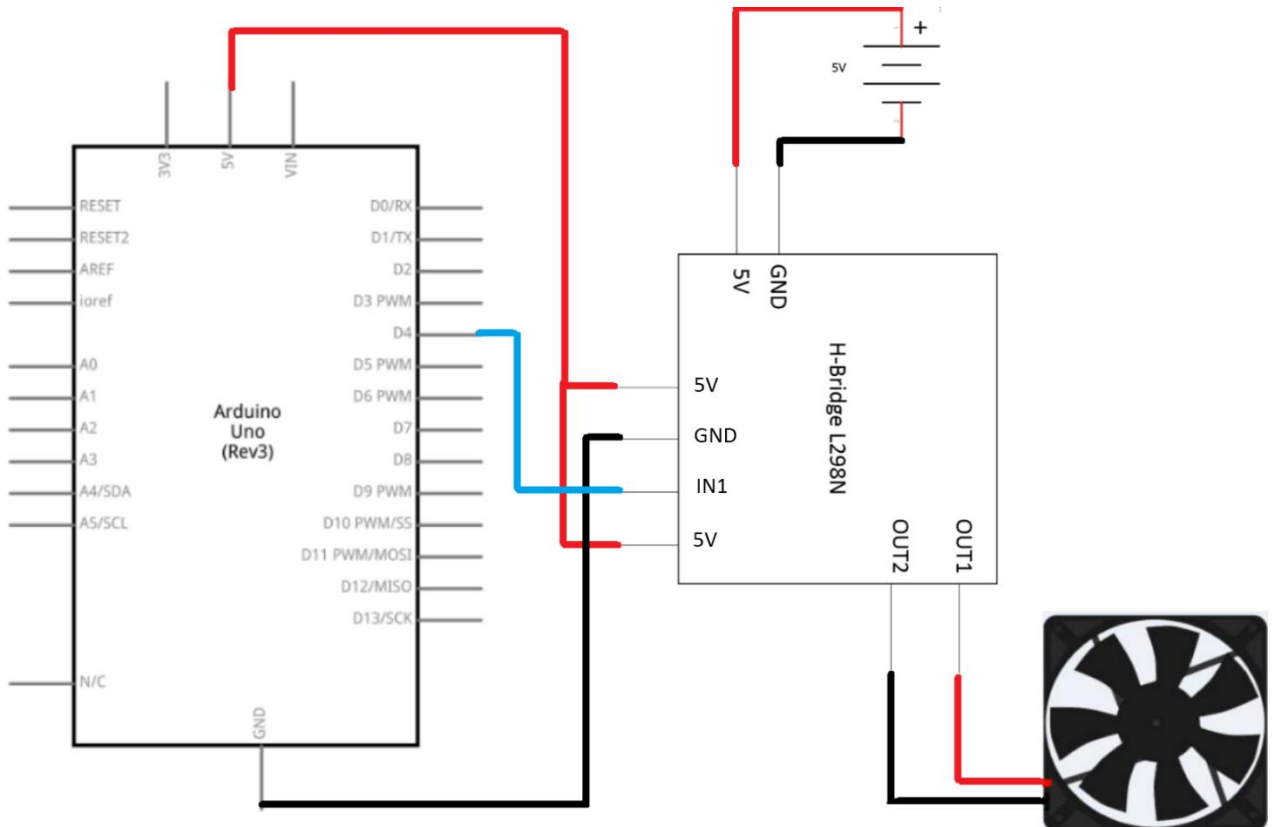


Figure 4.21 Fan schematic diagram

4.4.9 Integrated System

Figure 4.22 shows the implementation of the integrated system of our project. **Figure 4.23** shows the schematic diagram for the integrated system.

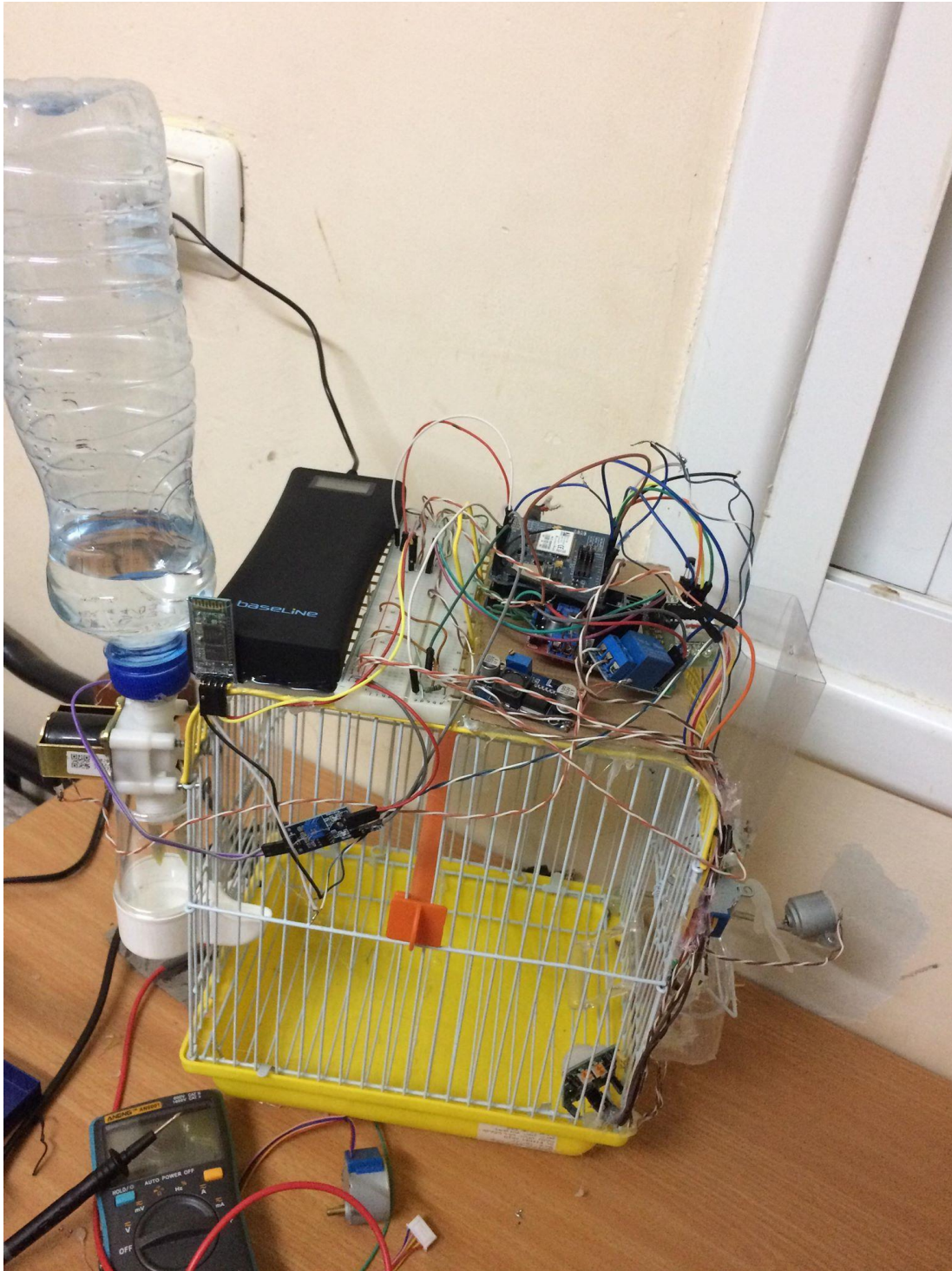


Figure 4.22 Complete system

Chapter 5

Testing

5.1 Overview

In this chapter, we will describe and discuss the steps, results of the implementation and the testing of the system.

5.2 Unit Testing

We starts by testing all the parts separately to ensure that all of the functions works perfectly and without errors.

5.2.1 Hardware components testing

1. Testing the Bluetooth adapter: it was tested successfully and it reads data from mobile.
2. Testing the Grove-Moisture and temperature sensor: all of them tested successfully.
3. Testing the Motion sensor: it was tested successfully.
4. Testing the Stepper motor: it was tested successfully.
5. Testing the Fan: it was tested successfully.
6. Testing the Water Valve: it was tested successfully.

5.2.2 Android Application Testing

1. Testing the android SDK
2. Testing the design of each activity

5.3 Integrated testing

After ensuring that all the parts is working well, we started assembling and integrating the parts with each other to make the system ready to operate.

5.3.1 Android – Arduino system integration test

After sending all types of the commands that the android application supports to the Arduino we checked out that each of the following unit function works well:

- Food addition operation automatically and manually.
- Water addition operation automatically and manually.
- Getting rid of unwanted seeds by triggering the fan.
- Receiving the following information in different conditions was successfully:
 1. Temperature of the surrounding place of the cage
 2. Food status (Good, Low or Empty)
 3. Water status (Good, Low or Empty)
 4. Last detected motion

Our system was completed and tested successfully and it works as required without any problems.

Chapter 6

Conclusion & Future work

6.1 Overview

In this chapter, we will conclude the challenges, final result and future work of our project.

6.2 Implementation challenges and issues

We faced several problems while working on the project either on Software and Hardware.

6.2.1 Hardware challenges

- The ESP8266 Wi-Fi module doesn't work after much efforts to trigger it, so we had to replace it by Wi-Fi Shield and we took long time to trigger it.
- When the integrated test started we discovered that some of the components doesn't work because of the current shortfall so we had to by new components and starts new work.

6.2.2 Software challenges

- Android auto receive data: we faced the port problem, we wanted to receive data from the server without opening a port in the router of the user so we had to find a code that accept data from the server without need of the port, we find many codes but all of them failed despite they work correctly with others but in the end after long searching we found a code as we need.
- Wi-Fi Shield triggering: this was very new to us, so we had to learn about it and searching the code that trigger it, it took hard effort till we got what we want.

6.3 Final Result

The system was able to measure many values through sensors then move these values to Arduino to process them to do different functions. On the other hand, the android

application was able to send commands and receive information through internet without any problem.

The main system's components, which are: Arduino, Wi-Fi Shield, sensors and android application were all combined together, and formed an integrated system that reads the values of the sensors by the Arduino and processes them to do the required functions.

6.4 Future Work

1. Cleaning the base of the cage.
2. Covering the cage when the temperature reaches less than a threshold.
3. Making the system distributed so control many cages.