

EEE4113F Literature Review



Prepared by:

Karan Abraham

Lundi Mabasa

Mpilo Nxasana

Michael Van Staden

Prepared for:

EEE4113F

Department of Electrical Engineering

University of Cape Town

Declaration

1. I know that plagiarism is wrong. Plagiarism is to use another's work and pretend that it is one's own.
2. I have used the IEEE convention for citation and referencing. Each contribution to, and quotation in, this report from the work(s) of other people has been attributed, and has been cited and referenced.
3. This report is my own work.
4. I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as their own work or part thereof.



May 12, 2024

Karan Abraham

Date



May 12, 2024

Lundi Mabasa

Date



May 12, 2024

Mpilo Nxasana

Date



May 12, 2024

Michael Van Staden

Date

Contents

| | |
|---|------------|
| List of Figures | vi |
| Abbreviations | vii |
| 1 Introduction | 1 |
| 1.1 Background | 1 |
| 1.1.1 Problem Statement | 1 |
| 1.2 Objectives | 1 |
| 1.3 System Requirements | 1 |
| 1.4 Scope & Limitations | 1 |
| 1.5 Report Outline | 1 |
| 2 Problem Analysis | 2 |
| 2.1 Design School Activities | 2 |
| 3 Literature Review | 3 |
| 3.1 Introduction | 3 |
| 3.2 The Importance of Monitoring Avian Weight Changes | 3 |
| 3.3 Current Weighing Methods | 3 |
| 3.3.1 Spring Scales | 4 |
| 3.3.2 Electronic Scales | 4 |
| 3.3.3 Perching Scales | 4 |
| 3.4 Data Transmission and User Interface | 4 |
| 3.5 Power supply | 5 |
| 3.5.1 Wall Power | 5 |
| 3.5.2 Energy Harvesting | 6 |
| 3.5.3 Limitations | 6 |
| 3.6 Challenges and Considerations | 6 |
| 3.6.1 Size and Species | 6 |
| 3.6.2 Environmental Conditions | 6 |
| 3.6.3 Material Considerations | 7 |
| 3.6.4 Ethical Considerations | 7 |
| 3.7 Conclusion | 7 |
| 4 Sensing Subsystem (NXSMPI001) | 8 |
| 4.1 Introduction | 8 |
| 4.2 Requirements Analysis | 8 |
| 4.3 Design Process | 9 |

| | | |
|----------------------|---|-----------|
| 4.3.1 | Microcontroller Unit (MCU) | 9 |
| 4.3.2 | Weight Sensor | 9 |
| 4.3.3 | Sensor Amplifier | 10 |
| 4.3.4 | Analogue to Digital Converter (ADC) | 13 |
| 4.3.5 | Weight Measurement Algorithm | 14 |
| 4.3.6 | Tare Functionality | 15 |
| 4.4 | Acceptance Test Procedure (ATP) | 16 |
| 4.4.1 | Load Cell and Amplifier tests | 16 |
| 4.4.2 | Weight Measurement Test | 17 |
| 4.4.3 | Tare Function Test | 17 |
| 4.5 | Results and Discussion | 18 |
| 4.5.1 | Load Cell and Amplifier test | 18 |
| 4.5.2 | Weight Measurement Test | 18 |
| 4.5.3 | Tare Function Test | 18 |
| 4.5.4 | Acceptance Test Results | 18 |
| 5 | Conclusions | 21 |
| Bibliography | | 22 |
| A Appendix A4 | | 24 |
| A.0.1 | Amplifier Test | 24 |
| B Appendix A5 | | 26 |
| C Appendix A6 | | 27 |
| C.1 | GA Table: VSTMIC004 | 27 |

List of Figures

| | | |
|------|---|----|
| 3.1 | Overall Performance of each Protocol [1] | 5 |
| 4.1 | Circuit Schematic of Instrumentation Amplifier | 10 |
| 4.2 | Output Voltage of the Instrumentation Amplifier | 11 |
| 4.3 | Circuit Schematic of Low Pass Filter | 11 |
| 4.4 | Input and Output Waveform of Filter | 12 |
| 4.5 | Input and Output Waveform of the Final Filter | 12 |
| 4.6 | Schematic of Split Supply Circuit | 12 |
| 4.7 | Relationship between Weight and Voltage | 13 |
| 4.8 | Circuit Schematic of the ADS1115 | 14 |
| 4.9 | Code Snippet of the Function that converts voltage to a weight | 14 |
| 4.10 | Code Snippet of the Function that takes the Moving Average of the weights | 15 |
| 4.11 | Code Snippet of Interrupt Service Routine for push button | 16 |
| 4.12 | Image of split supply test | 16 |
| 4.13 | Image of mock scale | 17 |
| 4.14 | Image of load cell and amplifier test | 17 |
| 4.15 | Image of Weight measurement in terminal | 18 |
| 4.16 | Plot of the output voltage against the weight | 18 |
| 4.17 | Image of load cell and amplifier test | 19 |
| A.1 | Image of Amplifier test when input is 1mV | 24 |
| A.2 | Plot of Output voltage against input voltage from the oscilloscope | 25 |
| B.1 | LM395 Electrical Specifications | 26 |

Abbreviations

Chapter 1

Introduction

Philosophers have hitherto only interpreted the world in various ways;
the point is to change it.

—Karl Marx

1.1 Background

1.1.1 Problem Statement

Sally, a researcher at the FitzPatrick Institute, needs a way to weigh the red-winged starlings without startling them, because interestingly, they record their weight manually using a kitchen scale.

1.2 Objectives

1.3 System Requirements

1.4 Scope & Limitations

1.5 Report Outline

Chapter 2

Problem Analysis

2.1 Design School Activities

Chapter 3

Literature Review

3.1 Introduction

Weight is an important metric for evaluating the overall health, behavior and ecology of a bird in ornithological research. However obtaining this weight data presents several challenges due to their size, fragility and often rapid movements. This literature aims to explore and evaluate the different techniques and technologies researchers employ to obtain weight data, the different ways this data could be transmitted and recorded, and the types of devices needed to power such a system. This literature review will also consider the challenges described above and, other environmental and ethical factors that must be taken into account when weighing birds in the wild.

3.2 The Importance of Monitoring Avian Weight Changes

According to Clark, “Weight summarizes the total biomass of an individual and is probably the most convenient standard of energetic comparisons.” [3], which emphasizes how valuable the weight data of an individual bird can be. Baldwin and Kendeigh state that “The weight of birds and the variations and fluctuations of these weights furnish criteria of considerable importance in the understanding of the physiological and ecological reactions of the bird as a living organism [4], which further establishes how important the tracking of an individual bird’s weight data can be. Both of these statements cement how valuable weight data can be for ornithologists studying individual birds.

Clark goes on to say, “Weights have been used in analyses of the factors that influence differences in species diversity between communities.” [3]. They also state that “Weights and census data have often been combined to calculate the total biomass of a particular species or group of species in an area”. These statements add that weight data could also be invaluable for the analysis of an entire community of birds as opposed to just monitoring individuals.

3.3 Current Weighing Methods

This section examines the different methods and tools used to obtain weight data in ornithological research today.

3.3.1 Spring Scales

Spring scales measure weight based on the extension of a spring when a force (the weight of the bird) is applied. Their main advantage, as described by Manolis [5], is that they are “relatively inexpensive and sufficiently portable to suffice for short-term field project[s]”. However, within the same study, the scale was only accurate to within 0,5g and when smaller birds can weigh less than 50g, spring scales may lack the precision for such research applications.

3.3.2 Electronic Scales

Electronic scales utilize load cells or strain gauges to convert the weight of the bird into an electronic signal, which can then be displayed on a digital screen. These scales offer precise measurements as shown by Carpenter et al. [6], where they were able to improve the precision of their measurements from 0,05g to 0,01g, by simply replacing their spring scales with electronic ones. Another advantage over spring scales, is that they do not have to be recalibrated after moving [6] and they tend to come with features such as taring functions to account for the weight of the housing holding the bird.

3.3.3 Perching Scales

Perching scales integrate a weighing platform into an artificial perch or nest. In Poole and Shoukimas' [7] study, birds landing on perch would deflect a transducer (a metal beam with 4 strain gauges bonded to it), thus generating an electronic signal. Reid et al. [8] used artificial nests rigged with a load cell in much the same way. In both studies, these electronic signals would then be recorded via some kind of electronic storage medium. This meant the birds could be weighed remotely, which minimizes stress and reduces the risk of injury, making perching scales particularly useful for long-term monitoring studies or behavioural observations. However, for such long-term studies, researchers would need to keep track of a large number of birds, which would also result in a large amount of data that needs to be stored.

Manolis [5] provides a solution to these issues by urging other researchers to make use of telemetry. One such technology is Radio Frequency Identification (RFID) which enables researchers to track individual birds and record their weight automatically. Wang et al. [9] made use of RFID by attaching two transponders to each bird, which would be detected by antennas placed under the perches. When a tagged bird interacts with the RFID reader, its unique identifier and weight are recorded electronically, making the data much easier to organise. It also reduces the volume of data created as the weight is only taken when the birds are on top of the perch. This allows researchers to collect data on a larger scale but this data need not be stored locally.

3.4 Data Transmission and User Interface

The method that the FitzPatrick Institute currently have for reading the bird weight is having one of the researchers go up to the scale and read off the screen. They then record the weight into *Cybertracker* [10], a mobile app that creates an Excel spreadsheet for them to analyze later. This highlights a need for a way to access that data remotely; or perhaps, a way to send that data directly to the Cybertracker app.

There are many communication protocols for transmitting data from a microprocessor. In a comparative performance study by Eridani et al., three protocols were compared: Bluetooth, Wi-Fi direct, and ESP-NOW ('a new protocol that allows multiple devices to communicate with each other without the use of Wi-Fi, with low power consumption' [1]). Five metrics were used in the tests: maximum range, transmission speed, latency, power usage, and signal resistance to obstructions [1]. A brief summary of the performance of each protocol is shown below in Figure 3.1.

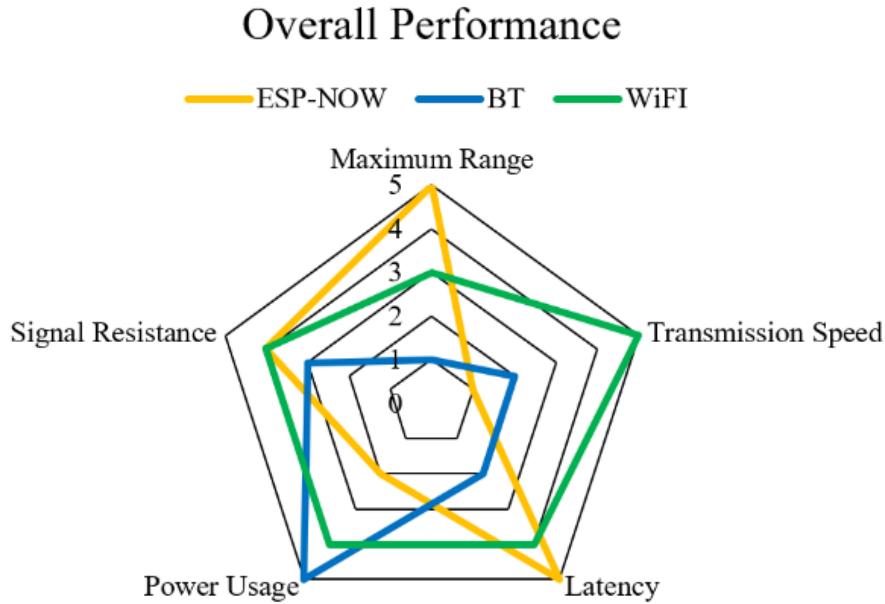


Figure 3.1: Overall Performance of each Protocol [1]

ESP-NOW performs best in range and latency; Bluetooth in power usage; and Wi-Fi in transmission speed.

Budoyo and Andriana used the internet when designing a prototype of a digital scale to measure the weight of onions. [11]. They interfaced the microcontroller (an ATMega2560) to the internet using an ESP8266 Wi-Fi module. The weight data is sent to a website where it is stored in a database. A database is useful in creating an Excel spreadsheet with many fields which is the end product that the client requires.

3.5 Power supply

Traditional weighing scales have relied on either battery or electric power sources for operation. Battery-powered scales utilize internal batteries, commonly alkaline or lithium-ion, to supply the necessary electrical power. This section will discuss the different types of power supplies available for bird scales and the power limitations on what type of power source the final design can use.

3.5.1 Wall Power

For indoor scales, which are in a fixed to one spot, the electric-type scales are directly connected to a power source via a cord, typically drawing from AC power provided by a wall outlet. These type of

scales are used in laboratories and residential environments.

3.5.2 Energy Harvesting

One other method researchers use in low powered bird scales is Energy Harvesting. Energy Harvesting is used to extend the lifespan of the scale and sensing devices, however the process is not always effective [12]. An example of energy harvesting would be using wind or solar as a power source to the scale device. This concept is useful in situations where the scale is left in the field for data capture, and only accessed after a prolonged period of time. It is worth noting that low-power weighing scales are an existing topic, where in some cases there are scales and sensors that are able to take measurements, read and communicate the data in real time [13].

3.5.3 Limitations

The reliance on electrical power poses serious constraints, particularly in terms of mobility. This limitation becomes pronounced in specialized applications such as bird weighing scales, especially for very mobile birds such as the starlings. A better solution is to use a rechargeable battery source, such as the alkaline batteries mentioned above. Environmental conditions also pose a risk to the battery lifespan. Solar panels can pose collision risks for birds, particularly if the panels are highly reflective. Some birds may collide with solar panels while flying, leading to injury or mortality.

3.6 Challenges and Considerations

While modern weighing methods offer significant advantages in terms of accuracy, convenience, and animal welfare, researchers must consider several factors when selecting the most appropriate technique for their study.

3.6.1 Size and Species

The size and behaviour of the target bird species may influence the suitability of different weighing methods. Some birds may become skittish around researchers which would make measurements unreliable. In Manolis' case [5], they had to use binoculars to take readings of the scale from afar; an inconvenience that is entirely removed from the solutions presented by Poole and Shoukimas [7] and Reid et al. [8]. Smaller birds may require scales with higher precision, while larger species may benefit from perching scale systems capable of handling multiple subjects simultaneously.

3.6.2 Environmental Conditions

Field studies often expose equipment to challenging environmental conditions. For example, Manolis [5] had to keep swaying to a minimum to get accurate readings, hence spring scales would not be suitable in windy conditions. Rain can seriously damage electronic components so it was important for Reid et al. [8] to house the amplifier unit in “a small water-resistant case with a sealed connection to the data logger cable”. Researchers must choose weighing methods that are robust and reliable under these circumstances, with weather-resistant features where necessary.

3.6.3 Material Considerations

An important consideration that our design would need to fulfill is that it is safe for use on birds. Given that they will be in direct contact with the scale, toxicity is a primary concern when considering the materials used to construct the scale.

In the design of an electronic scale, artificial materials such as plastics are an attractive option due to their naturally weatherproof properties. However, caution must be exercised when considering specific materials. Artificial materials such as polytetrafluoroethylene (PTFE) are a common source for airborne toxicity in avians [14]. However, in a paper by Kroshefsky [15] this material is only cause for toxicity concern when exposed under high temperatures. This is because PTFE begins to decompose in air at around 200°C. Even if the scale is placed outdoors, the ambient temperature will be well under this temperature limit, making this material a viable choice in the housing.

Heavy metal poisoning is another important concern when it comes to our choice of materials. The most common occurrences of which come from the ingestion of lead [16]. When lead is ingested, it can be absorbed in the gastrointestinal tract and then taken up by soft tissues and eventually bone [16]. The paper by C. Pollock includes a list of common sources of lead in household items, the most notable of which is solder. Thus some kind of insulation is required around soldered circuitry to avoid any trace amounts of lead affecting the birds.

3.6.4 Ethical Considerations

Ethical guidelines emphasize the importance of minimizing stress and harm to the animals being studied. As such researchers should prioritize methods that avoid the need to handle the birds, as this will minimize the risk of injury to the researchers and the birds. This means that the red-winged starlings will need to be lured such that their weight measurements can be taken. This can be achieved using the many perching scale solutions described earlier, but for more traditional weighing methods, Manolis [8] provides a solution. In their study they used sunflower seeds to entice the birds to land on the scale, and since red-winged starlings tend to scavenge for food, this will prove useful for this application as well.

An automatic feeder device could be implemented as a way to streamline this process. However, researchers have found that the introduction of automated feeders tend to reduce species variety in a given area [17]. Automated feeders are also highly inconsistent with seasonal change, which would result in it being a highly inconsistent form of luring birds [17]. The introduction of an automated feeding strategy would have a noticeable effect on the ecological and weight cycles of local birds, especially those that would be studied.

3.7 Conclusion

Based on the reviewed literature, it can be seen that accessing data remotely is very much feasible using accessible technology such as Wi-Fi. The importance of gathering weight data was established and how it would be of benefit to ornithologists was explored. Various challenges and considerations have been presented in the reviewed literature and the importance thereof will be taken into account going forward.

Chapter 4

Sensing Subsystem (NXSMPI001)

4.1 Introduction

In order store the weight data of the Starlings in a database, the weight of the birds must first be obtained. That is the purpose of this subsystem, to translate the weight force of each bird on the scale into a digital reading that can be transmitted. This section will go through the design process involved in constructing the circuitry that turns the weight into an electrical signal, and developing the algorithms that will convert this signal into a usable measurement. As such, this section not only encompasses the sensors but also the microcontroller and the code responsible for allowing the two to communicate.

4.2 Requirements Analysis

From the user requirements discussed in Section 1.3, the non-functional and functional specifications for this specific subsystem were derived. They are shown in Tables 4.1 and 4.2 respectively.

Table 4.1: Non-functional Specifications of the Sensing Subsystem

| User Requirement | Specification Description | Specification no. |
|-------------------|--|-------------------|
| Portable | The final circuitry must be able to fit in a box that is 100x100x50mm. | SS1 |
| Long battery life | The final circuitry should consume less than 30mA. | SS2 |

Table 4.2: Functional Specifications for Sensing Subsystem

| User Requirement | Specification Description | Specification no. |
|---|---|-------------------|
| The scale must measure weights of up to 500g. | The weight sensor must have a maximum capacity greater than 750g (1.5 times safety factor). | SS3 |
| | The sensor and amplifier must output a voltage proportional to the weight force applied up to a weight of 500g. | SS4 |

| | | |
|---|---|------|
| | The microcontroller must run an algorithm that converts output codes from the ADC into a weight measurement. | SS5 |
| The scale must measure weight accurate to 0.1g. | The ADC must be able to resolve voltage changes from weight changes that are less than 0.1g. | SS6 |
| | The ADC must have a full-scale error less than a voltage change resulting from a change in weight of 0.1g. | SS7 |
| The scale must have a tare function. | The microcontroller must have a digital input pin to read the user's inputs from a button or switch. | SS8 |
| | The microcontroller must subtract the current weight from all subsequent measurements when the digital pin receives an input. | SS9 |
| | There must be an LED that indicates when the scale is in tare mode. | SS10 |
| The scale must be battery-powered. | The microcontroller and all the surrounding circuitry must be able to operate on one positive rail. | SS11 |

4.3 Design Process

4.3.1 Microcontroller Unit (MCU)

The Arduino was chosen as its Integrated Development Environment (IDE) has ample support and libraries which will make interfacing with all the different modules simple and straightforward. Within the Arduino family the Arduino Nano was initially chosen as it was one of the cheapest Arduino and it came in a small form factor. However, the User Interface subsystem required a WiFi or Bluetooth module so the Arduino Nano 33 IoT was chosen instead. Although BLE and BLE Sense also meet these requirements, they come with additional sensors that are unnecessary. All the Arduino chips also come with several low power modes that can be leveraged to reduce power consumption.

4.3.2 Weight Sensor

A strain gauge is an electrical component whose resistance changes when a force is applied to it. Strain gauges work on the principle that the resistance of a conductor is proportional to its length, as shown in the equation below.

$$R = \rho \frac{L}{A}$$

One solution is to put a strain gauge in series with another resistance, then place the strain gauge on a beam. When the beam deflects under the bird's weight, the change in voltage across the strain gauge

can be measured. The issue with this setup is that the change in resistance, and thus the subsequent change in voltage, will be very small. This means a very high resolution ADC will be required to resolve these small changes in voltage. The resolution required could be reduced by amplifying the signal, however this would also amplify the DC offset introduced by the voltage divider, quickly saturating the output.

A better solution is a load cell which has 4 strain gauges in a Wheatstone configuration. This means that when the load cell has no load on it, the voltage will be zero, and when the device deflects, there will be a slight voltage difference between its 2 output terminals. As discussed above, this output can be sent through an amplifier thus reducing the resolution required for the ADC. To meet the sensor specifications, a 1kg load cell will be used.

The specifications for one such load cell by HKD is shown in Table 4.3 below.

Table 4.3: Table on load cell specifications

| | |
|------------------|---------------------------|
| Rated Load | 1kg |
| Rated Output | $1.0 \pm 0.15\text{mV/V}$ |
| Zero Output | $\pm 0.1\text{mV/V}$ |
| Input Impedance | $1115 \pm 10\Omega$ |
| Output Impedance | $1000 \pm 10\Omega$ |

At the rated load, the output will be $0.001V_{cc}$, hence the amplifier needs a gain of 1000.

4.3.3 Sensor Amplifier

From Table 4.3, the output impedance of the load cell is quite significant, meaning there will need to be an input buffer between it and the amplifier to avoid loading. The instrumentation amplifier is thus ideal circuit for achieving this and it is shown in Figure 4.1 below.

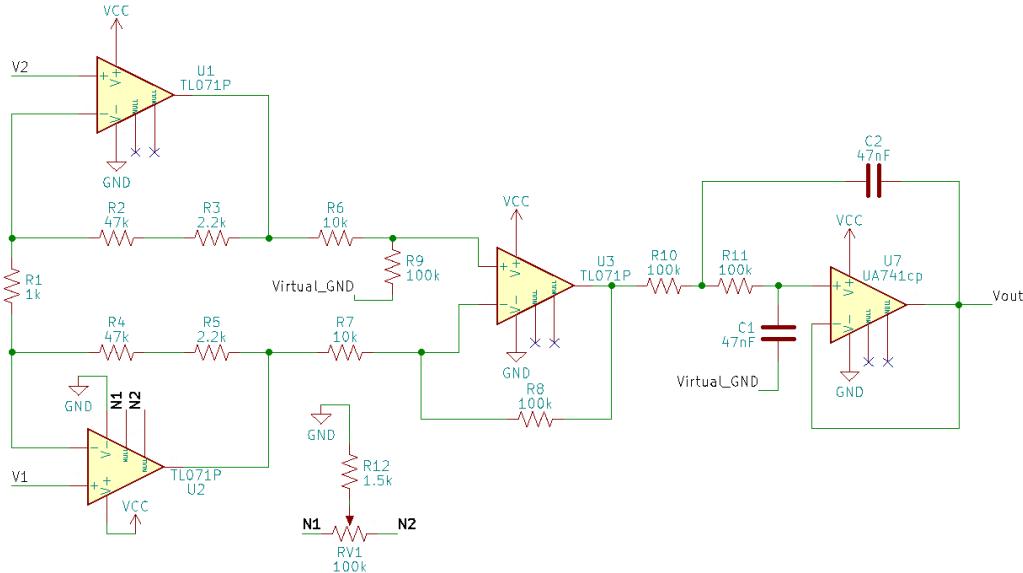


Figure 4.1: Circuit Schematic of Instrumentation Amplifier

The circuit has three stages. The first stage has two input buffers which also amplify the input signal.

The second is a differential amplifier which is a circuit whose output is proportional to the difference between the two inputs. The final stage is a low pass filter. The final output voltage is related to the input voltage by the expression below.

$$V_{out} = (V_2 - V_1) \left(1 + \frac{2(R_2 + R_3)}{R_1} \right) \left(\frac{R_9}{R_6} \right)$$

Figure 4.2: Output Voltage of the Instrumentation Amplifier

From the expression above, when the load cell is connected to the two input terminals, its output will be amplified by a factor of 994, which is close to the gain required. The amplifier have such a large gain presents two issues.

The first is that real op-amps have an input offset voltage. As the offsets from the input stage propagate through the circuit, they are amplified resulting in the output having a large bias and saturating for very small weights, hence the op-amps used were the TL071P. These are JFET op-amps meaning they have a very low input offset voltage, in this case, of 1mV. This is still large in comparison to the input, but they also come with two NULL pins which allows the input offset to be adjusted, and thus reduced to 0. The is the purpose of potentiometer RV1 in Figure 4.1. Another reason for choosing TL071P is that their minimum recommended supply voltage is 4.5V which means unlike other JFET op-amps they can operate at lower supply voltages. This advantageous since the scale will be battery powered so there will not be a large supply.

The second is that noise from the input will also be amplified as it propagates through circuit, making the final output difficult to measure. The low pass filter in the final stage addresses this. Since output is a DC voltage, ideally the cutoff frequency should be as low as possible to attenuate the most amount of noise, but this would have a negative impact on the rise time. A lower cutoff frequency would also require larger capacitors. The sample rate for final system will be 10Hz (discussed later). This equates to a period of 0.1s and ideally the output should settle within half that time. It takes 5 time constants for the output to settle to 99% of its final value. This means that $5RC = 0.05s$ or $RC = 0.01s$. If a $100k\Omega$ resistor is used then the capacitor would need $100nF$. The filter also needs a steep roll-off to ensure a clean output, so a second stage can be added at the input, to make it a second order filter. The input stage of this filter needs to have much lower impedance than output stage to avoid loading, which would resulting in the filter having a larger cutoff frequency than was calculated. Using a $10k\Omega$ resistor, the capacitor needed would be $1uF$. The equates to a cutoff frequency of 16Hz. It is difficult to know the exact rise time for higher order filters from calculation alone, as such, this filter was simulated in LTSpice. The circuit diagram is shown in Figure 4.3 below.

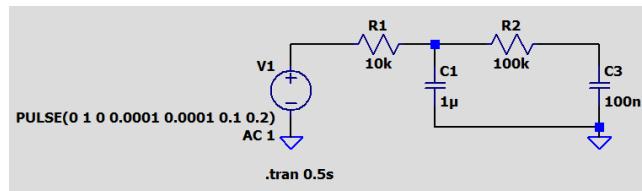


Figure 4.3: Circuit Schematic of Low Pass Filter

The input was set to a $1V_{pp}$ square wave with a frequency of 5Hz. Figure 4.4 below shows the input and output of the circuit.

4.3. Design Process

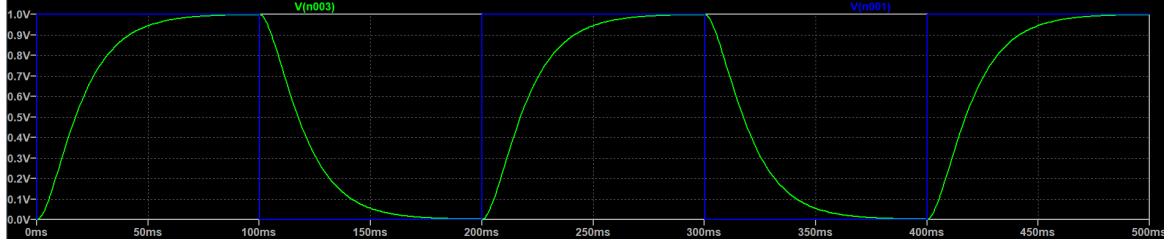


Figure 4.4: Input and Output Waveform of Filter

From the waveform above it can be seen that the rise time is too large, as the output (in green) is barely settling in time for the next half-cycle. This can be rectified by halving the size of the capacitors to 470nF and 47nF, as seen in Figure 4.1. The new output is shown in Figure 4.5 below.

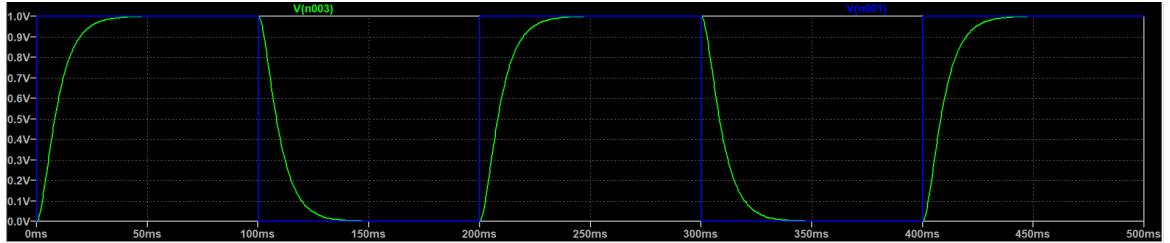


Figure 4.5: Input and Output Waveform of the Final Filter

As seen above, the filter now meets the speed requirements. Since the instrumentation amplifier has op-amps, it needs 2 rail voltages, a positive and a negative. Unfortunately there is only a single supply, however this supply can be split in two with a simple op-amp circuit, as shown in Figure 4.6 below.

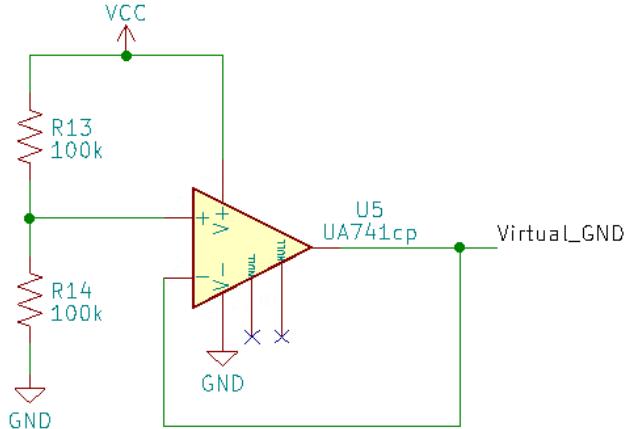


Figure 4.6: Schematic of Split Supply Circuit

If the new reference point is made to be ‘Virtual GND’, then two rail voltages equal to $\pm \frac{V_{cc}}{2}$ are obtained. This does mean that output of the amplifier will have an offset of $\frac{V_{cc}}{2}$, but this can be stepped down using a voltage divider as to not damage the input to the microcontroller. In testing, a 5V supply was initially used but this resulted in the op-amps saturating. In the early stages of the design process, the amplifier was tested using $\pm 3.3V$ and this worked fine so in the end a 6.6V supply was chosen. This will be the supply voltage required from the Power Subsystem. The Arduino Nano 33 IoT has an operating voltage of 3.3V so the voltage divider must have a gain of 0.5.

Finally, because the output from the amplifier will no longer go directly to the microcontroller, an op-amp must be included in the low pass filter to buffer the input of the voltage divider. The new active low pass filter will be in a Sallen-Key configuration with $R = 100k\Omega$ and $C = 47nF$ as shown in Figure 4.1.

4.3.4 Analogue to Digital Converter (ADC)

Since the weight force on the load cell is proportional to output voltage out of the load cell, the change in weight is equal to the change voltage as shown in the equation below.

$$\frac{W_1}{W_2} = \frac{V_1}{V_2}$$

Figure 4.7: Relationship between Weight and Voltage

By substituting in the rated load, rated output and the minimum weight, the smallest change in the output can be determined.

$$\frac{0.1g}{1000g} = \frac{V_{min}}{0.5(994(3.3mV))}$$

$$V_{min} = 0.164mV$$

As such, 1 LSB (Least Significant Bit) of ADC must be less than 0.164mV. Since the supply voltage from the Arduino is 3.3V, an ADC with a minimum resolution of 15 bits is required. The Arduino only comes with a 12-bit ADC so an external module will be needed. The module that was chosen is the ADS1115 16 bit ADC its specification for this application are summarized in Table 4.4 below.

Table 4.4: Table on ADS1115 specifications

| | Min | Typical | Max | Unit |
|-----------------------|------------|----------------|------------|-------------|
| Supply Voltage | 2 | - | 5.5 | V |
| Data Rate | 8 | - | 860 | SPS |
| Offset error | - | ± 3 | - | LSB |
| Gain error | - | 0.01 | 0.15 | % |

From Table 4.4, the offset error is equivalent to 0.151mV which is less than V_{min} . The maximum gain error is quite large, translating to an offset of 4.95mV at the last output code. Even under typical conditions the offset will be 0.33mV. However this is not a big issue, as the scale only needs to be accurate over half the range and the gain error can be compensated for in software. The noise performance is also great as at low data rates, not a single bit of resolution is lost to quantization noise. The final reason this module was chosen is that most ADC come standalone in a SIOC package, but this one comes as a development board kit with headers allowing for easy soldering onto a Veroboard. The circuit diagram for this module is shown in Figure 4.8 below. The ADC communicates with Arduino the I2C communication protocol, hence their SDA and SCL lines are connected. The Arduino has internal pull-up resistors on these lines so there's no need for external ones. The output from the amplifier is halved by the voltage divider and sent to channel 0 on the ADC. This is necessary as the ADC can only take 0.3V above the supply. In this case the supply will be the regulated 3.3V from the Arduino.

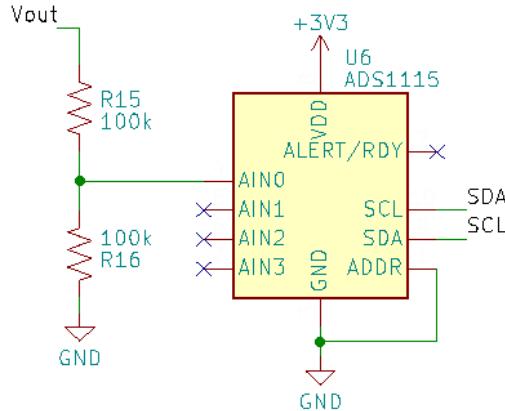


Figure 4.8: Circuit Schematic of the ADS1115

4.3.5 Weight Measurement Algorithm

The ADC was read from in code using the ADS1X15 Arduino library by Rob Tillaart [18]. The ADS1115 has a programmable gain amplifier (PGA) that allows it take more precise measurements at the cost of a reduced input voltage range. The different options for the gain with corresponding full scale range and precision are shown in 4.5 below.

Table 4.5: Table on ADS1115 gain options

| FSR [V] | LSB Size [uV] |
|-----------|---------------|
| +/- 6.144 | 187.5 |
| +/- 4.096 | 125 |
| +/- 2.048 | 62.5 |
| +/- 1.024 | 31.25 |
| +/- 0.512 | 15.625 |
| +/- 0.256 | 7.8125 |

The second option for the gain was chosen since the input is expected to be between 1.65V and 3.3V. The LSB size for this gain also meets the accuracy requirements. The data rate was also set to 16 samples per second as the sample rate for the system will 10 samples per second, and the lower the sample rate, the lower the noise. These were both set in code using the setGain() and setDataRate() functions from the library. Once setup, the sampled value would be read from the ADC using the readADC() function. The ADC value was then converted into voltage in Volts by multiplying it by the value returned by the toVoltage() function from the same library. That voltage would then be converted to a weight in grams using the voltsToGrams() function shown in Figure 4.9 below.

```
float voltsToGrams(float V) {
    float Vin = (2*V - 3.3)/994;           // Determine input voltage from output voltage
    float weight = 1000*Vin/(3.3*0.001);   // Determine weight from input voltage
    // float weight = (V-1.5935)/0.0015;   // Linear regression model
    return weight;
}
```

Figure 4.9: Code Snippet of the Function that converts voltage to a weight

From Equation 4.2, the output voltage of the amplifier is...

$$V_{out} = \frac{1}{2} (994V_{in} + 3.3)$$

And from this, the input voltage from the load cell can be determined. The weight is then calculated using Equation 4.7, and this was implemented in code as shown in 4.9 above.

The final outputted weight value was the ten-point moving average of all the measurements. The purpose of this was to reduce the effect of the bird's rapid movements on the weight measurement, as a moving average changes slower when compared to the input. This algorithm is implemented in code by the movingAverage() function shown in Figure 4.10 below.

```
float movingAverage(float* values, int index)
{
    float sum = values[SIZE];
    sum += values[index];
    if(index+1<SIZE) {
        sum -= values[index+1];
    }
    else {
        sum -= values[0];
    }
    values[SIZE] = sum;
    return sum/SIZE;
}
```

Figure 4.10: Code Snippet of the Function that takes the Moving Average of the weights

If 'SIZE' is set to 10, the 'values' array is a circular buffer that stores last 10 weight measurements, along with the sum of the whole array. The index increases incrementally with each new measurement and wraps back around to 0 once it goes past 9. This means that the latest value is stored at position 'index' within the array, while the value one position ahead of it is the oldest. When a new value is added to the array, it is added to the sum while the oldest value is removed from the sum. The sum divided is subsequently by 'SIZE' or 10 and that is the average that gets returned. By storing the sum in memory it removes the need to add up all the values in the array with every new measurement, making the program run more efficiently.

4.3.6 Tare Functionality

The tare function was implemented using an interrupt tied to the input of a push button. The button is made active low by connecting an internal pull up resistor in code. The interrupt is set to trigger whenever there is a falling edge from the button, which occurs when it is initially pressed. The corresponding Interrupt Service Routine (ISR) toggles the LED when this occurs as shown in Figure fig:S16 below. Within the main loop the change in the LED state is detected and the current weight is stored and added to an offset value. When the LED is toggled ON, all subsequent measurement will now be the difference between the weight and this offset, unless the offset is larger, in which it is just 0. When a button is pressed it does not close shut immediately. Instead its contacts bounce off one another many times within a split second, resulting in the microcontroller reading multiple inputs from

one button press. Hence a delay of 250ms is added to allow the button to settle before taking input from it again.

```
void buttonPressedISR()
{
    if(millis() - last_press > DEBOUNCE_DELAY){      // Button debouncing
        digitalWrite(LED_PIN, !digitalRead(LED_PIN)); // Toggle LED
        last_press = millis();
    }
}
```

Figure 4.11: Code Snippet of Interrupt Service Routine for push button

4.4 Acceptance Test Procedure (ATP)

4.4.1 Load Cell and Amplifier tests

The amplifier circuit built on the breadboard and connected to a 6.6V power supply. The split supply was tested by measuring the voltage of the positive and negative rail with respect to ‘Virtual GND’ using a multimeter as shown in Figure 4.12 below.

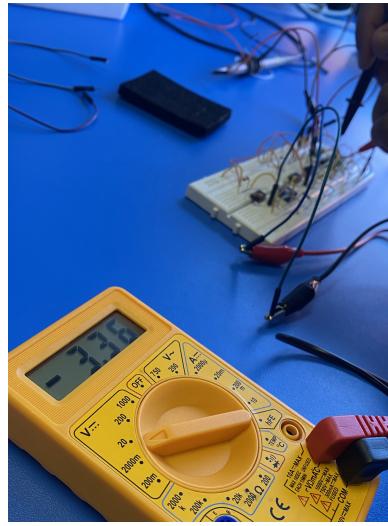


Figure 4.12: Image of split supply test

The load cell was attached to a mock scale as shown in Figure 4.13 below. Header pins were soldered to each of leads from the load cell. Within the same figure, the red and black leads were connected to V_{cc} and $Virtual_GND$ respectively, while the white and green wires leads connected to the V_1 and V_2 inputs of the amplifier respectively.

A kitchen scale was then used to measure and record the weight of an empty water bottle. This bottle was then placed on the front of the scale above the nails. The output with respect to ‘Virtual GND’ was measured using the oscilloscope and the output with respect to GND was measured using the multimeter (this is what the ADC would measure). This is shown in Figure 4.13 below. This process was repeated for different weights up to just before where the amplifier’s output saturates. The weight of the bottle was varied by filling it up with different amounts of water. The bottle was placed at

4.4. Acceptance Test Procedure (ATP)

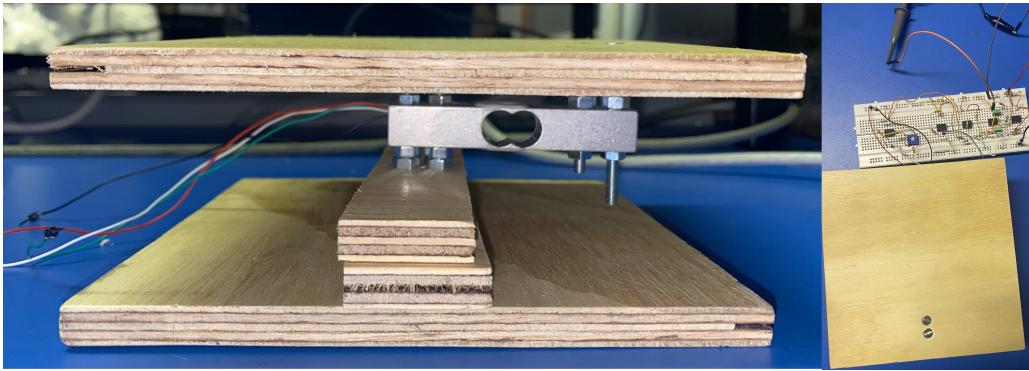


Figure 4.13: Image of mock scale

the same position on the scale in each run to ensure that the bottle's distance from the other end, or moment arm, was the same.

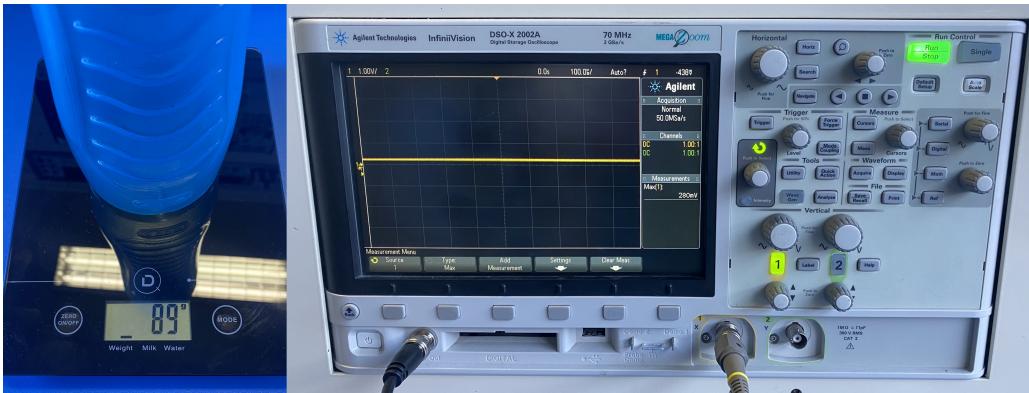


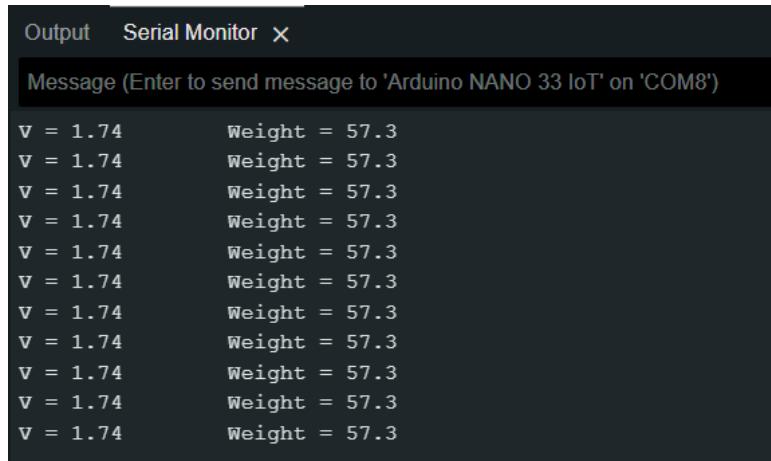
Figure 4.14: Image of load cell and amplifier test

4.4.2 Weight Measurement Test

The ADC was connected to the Arduino as shown in Figure 4.8 on the breadboard. The output from the voltage divider in Figure 4.13 was connected to channel 0 on the ADC. The Arduino was connected to the PC, the weight measurement program was run and a serial connection was established to see the output in the terminal. Once again, a bottle filled with different amounts water was placed on top of the load cell. The weight of the bottle measured by the scale and the weight measurement determined in code were both recorded. The measurement for the empty 66g bottle is shown in Figure 4.15 below.

4.4.3 Tare Function Test

The push button was now connected to pin D2 on the Arduino while the LED was placed in series with a $100\ \Omega$ resistor and connected to pin D3. The button was pressed and the resulting weight change in the terminal was observed.



The screenshot shows the Arduino Serial Monitor window. At the top, there are tabs for 'Output' and 'Serial Monitor'. Below the tabs, a message box says 'Message (Enter to send message to 'Arduino NANO 33 IoT' on 'COM8')'. The main area of the window displays a series of 13 lines of text, each consisting of two columns: 'V = 1.74' and 'Weight = 57.3'. This indicates that the Arduino is outputting a constant voltage of 1.74V and a weight of 57.3g.

Figure 4.15: Image of Weight measurement in terminal

4.5 Results and Discussion

4.5.1 Load Cell and Amplifier test

The plot of the output voltage against the weight is shown in Figure 4.16 below. It can be seen that the output voltage has linear response with the weight.

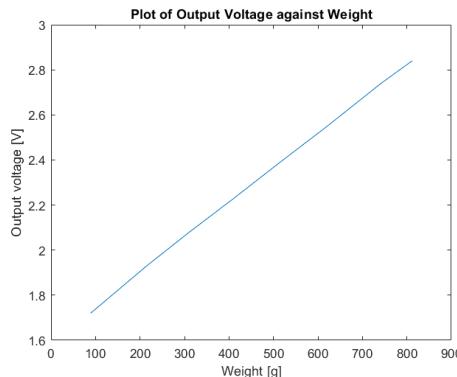


Figure 4.16: Plot of the output voltage against the weight

4.5.2 Weight Measurement Test

Table 4.6 below shows the measured weight of the bottle against its actual weight.

4.5.3 Tare Function Test

The output in the terminal when the button is pressed is shown figure 4.17 below.

4.5.4 Acceptance Test Results

Table 4.7 shows the results of the different acceptance tests.

Table 4.6: Table on Weight Measurements from Arduino

| Actual Weight [g] | Measured Weight [g] |
|-------------------|---------------------|
| 66 | 57.3 |
| 191 | 111.5 |
| 301 | 160.0 |
| 400 | 202.8 |
| 484 | 239.7 |
| 622 | 299.9 |
| 725 | 345.0 |
| 805 | 379.7 |



The screenshot shows the Arduino Serial Monitor interface. The title bar says "Output" and "Serial Monitor X". Below the title bar is a message box containing the text "Message (Enter to send message to 'Arduino NANO 33 IoT' on 'COM8')". The main window displays a series of weight measurements. Each measurement consists of two lines: a variable value (v) followed by a weight value. The variable v is consistently shown as 2.05, while the weight values fluctuate between approximately 26.3 and 283.0.

```

v = 2.05      Weight = 283.0
v = 2.05      Weight = 283.0
v = 2.05      Weight = 283.0
v = 2.05      Weight = 282.9
v = 2.05      Weight = 254.4
v = 2.05      Weight = 225.9
v = 2.05      Weight = 197.3
v = 2.05      Weight = 168.8
v = 2.05      Weight = 140.3
v = 2.05      Weight = 111.8
v = 2.05      Weight = 83.3
v = 2.05      Weight = 54.8
v = 2.05      Weight = 26.4
v = 2.05      Weight = 26.4
v = 2.05      Weight = 26.4
v = 2.05      Weight = 26.3

```

Figure 4.17: Image of load cell and amplifier test

Table 4.7: Acceptance Test Procedures

| Specification no. | Acceptance criterion | Result |
|-------------------|---|-----------------------------------|
| SS1 | Soldered circuit can fit in 100x100x50mm box. | Failed to test. Did not solder |
| SS2 | Less than 30mA is drawn from a 6.6V supply. | |
| SS3 | Load cell operates normally while under a 750g load or heavier. | Passed |
| SS4 | A linear response in the output voltage is observed. | Passed |
| SS5 | The microcontroller outputs a weight measurement in the terminal. | Passed |

| | | |
|-----------|--|---|
| SS6 & SS7 | The outputted weight from the terminal must match the given the weight to within 0.1g. | Failed |
| SS8 | The microcontroller detects when push button is pressed | Passed |
| SS9 | When the button is pressed once, the outputted weights have been offset by the weight measured when the button was pressed. When the button is pressed twice, the offset is removed. | Failed. The offset is not correct. Could be due to some precision errors in the way the Arduino handles floating point. |
| SS10 | The LED is toggled on and off with each button press. | Passed. |
| SS11 | The Arduino and external circuitry operate off of 6.6V | Passed. |

Chapter 5

Conclusions

The same rule holds for us now, of course: we choose our next world through what we learn in this one. Learn nothing, and the next world is the same as this one.

—Richard Bach, *Jonathan Livingston Seagull*

The purpose of this project was to...

This report began with...

The literature review was followed in Chapter...

The bulk of the work for this project followed next, in Chapter...

In Chapter...

Finally, Chapter... attempted to...

In summary, the project achieved the goals that were set out, by designing and demonstrating...

Bibliography

- [1] D. Eridani, A. F. Rochim, and F. N. Cesara, “Comparative performance study of esp-now, wi-fi, bluetooth protocols based on range, transmission speed, latency, energy usage and barrier resistance,” in *2021 international seminar on application for technology of information and communication (iSemantic)*. IEEE, 2021, pp. 322–328.
- [2] “Arduinoble.” [Online]. Available: <https://www.arduino.cc/reference/en/libraries/arduinoble/>
- [3] G. A. C. Jr., “Body weights of birds: A review,” *The Condor*, pp. 193–202.
- [4] S. C. K. S. Prentiss Baldwin, “Variations in the weights of birds,” *The Auk*.
- [5] T. Manoils and A. Manolis, “A simple technique for weighing birds at feeders without recapture,” *North American Bird Bander*, vol. 6, no. 3, p. 4, 2024.
- [6] F. L. Carpenter, D. C. Paton, and M. A. Hixon, “Weight gain and adjustment of feeding territory size in migrant hummingbirds,” *Proceedings of the National Academy of Sciences*, vol. 80, no. 23, pp. 7259–7263, 1983.
- [7] A. Poole, “A scale for weighing birds at habitual perches,” *Journal of Field Ornithology*, vol. 53, no. 4, pp. 409–414, 1982.
- [8] K. Reid, G. M. Liddle, P. A. Prince, and J. P. Croxall, “Measurement of chick provisioning in antarctic prions pachyptila desolata using an automated weighing system,” *Journal of Avian Biology*, pp. 127–134, 1999.
- [9] K. Wang, K. Liu, H. Xin, L. Chai, Y. Wang, T. Fei, J. Oliveira, J. Pan, and Y. Ying, “An rfid-based automated individual perching monitoring system for group-housed poultry,” *Transactions of the ASABE*, vol. 62, no. 3, pp. 695–704, 2019.
- [10] CyberTracker, “Cybertracker wiki.” [Online]. Available: <https://cybertrackerwiki.org/getting-started/>
- [11] Y. Budoyo and A. D. Andriana, “The digital weight scale of iot system using load cell sensor in ud. pangrakti tani,” *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*.
- [12] L. Meile, A. Ulrich, and M. Magno, “Wireless power transmission powering miniaturized low power iot devices: A revie,” in *2019 IEEE 8th International Workshop on Advances in Sensors and Interfaces (IWASI)*, June 2019, pp. 312–317.
- [13] J. Paek, J. Hicks, S. Coe, and R. Govindan, “Image-based environmental monitoring sensor application using an embedded wireless sensor network,” *Sensors*, vol. 14, no. 9, pp. 15 981–16 002, 2014. [Online]. Available: <https://www.mdpi.com/1424-8220/14/9/15981>

- [14] T. L. Lightfoot, “Pet bird toxicity and related environmental concerns,” *Veterinary Clinics of North America: Exotic Animal Practice*, pp. 229–259, 2008.
- [15] R. D. Kroshefsky, “Teflon poisoning: How dangerous is your cooking to your birds?” *AFA Watchbird Magazine*, pp. 30–33, 1980.
- [16] C. P. Cynthia Chow, “Heavy metal poisoning in birds,” pp. 1–9, 2009.
- [17] J. R. B. K. P. M. C. S. Josie A. Galbraith, Darryl N. Jones, “Urban bird feeders dominated by a few species and individuals,” *Frontiers in Ecology and Evolution*.
- [18] R. Tillaart, “Ads1x15: Arduino library for ads1015 = i2c 12 bit adc and ads1115 = i2c 16 bit adc,” 2022. [Online]. Available: <https://github.com/RobTillaart/ADS1X15>

Appendix A

Appendix A4

A.0.1 Amplifier Test

The input into the amplifier was connected to the waveform generator on the oscilloscope. The input was set to 0V DC and the potentiometer was adjusted until the measured output on the oscilloscope was also 0. The output to the Arduino was then measured using a multimeter. This was repeated for different input values until the output saturated. This was also compared to the ideal output given by the following expression.

$$V_{out} = \frac{1}{2} (994V_{in} + 3.3)$$

The measurement for the output when the input was 1mV DC can be seen in Figure A.1 below.

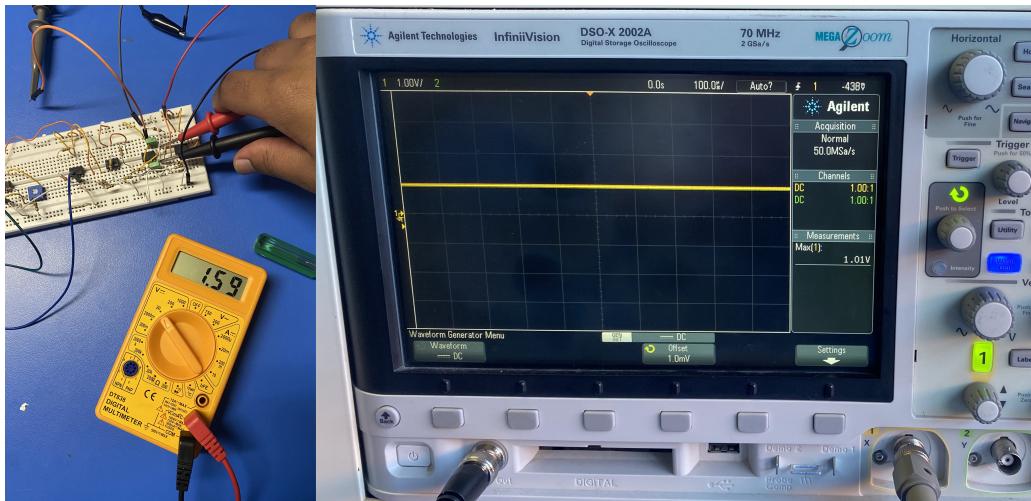


Figure A.1: Image of Amplifier test when input is 1mV

The resulting plot of the output voltage against the input voltage from the oscilloscope is shown in Figure A.2 below.

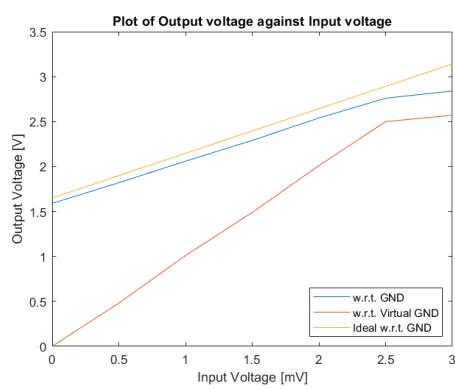


Figure A.2: Plot of Output voltage against input voltage from the oscilloscope

Appendix B

Appendix A5

| ELECTRICAL CHARACTERISTICS | | | | | | | | | |
|-------------------------------------|---|-------|-----|-----|-------|-----|-----|---------|--|
| Parameter | Conditions | LM195 | | | LM395 | | | Units | |
| | | Min | Typ | Max | Min | Typ | Max | | |
| Collector-Emitter Operating Voltage | $I_C \leq I_{C\text{MAX}}$ | | | 42 | | | 36 | V | |
| Base to Emitter Breakdown Voltage | $0 \leq V_{BE} \leq V_{BE\text{MAX}}$ | 42 | | | 36 | 60 | | V | |
| Collector Current | | | | | | | | | |
| TO-3, TO-220 | $V_{CE} \leq 15V$ | 1.2 | 2.2 | | 1.0 | 2.2 | | A | |
| TO-5 | $V_{CE} \leq 7.0V$ | 1.2 | 1.8 | | 1.0 | 1.8 | | A | |
| Saturation Voltage | $I_C \leq 1.0A, T_A = 25^\circ C$ | | 1.8 | 2.0 | | 1.8 | 2.2 | V | |
| Base Current | $0 \leq I_B \leq I_{B\text{MAX}}$ | | 3.0 | 5.0 | | 3.0 | 10 | μA | |
| Quiescent Current (I_Q) | $V_{CE} = 0V, T_A = 25^\circ C$ | | 2.0 | 5.0 | | 2.0 | 10 | mA | |
| Base to Emitter Voltage | $I_B = 1.0A, T_A = +25^\circ C$ | | 0.9 | | | 0.9 | | V | |
| Switching Time | $V_{CE} = 36V, R_L = 360\Omega, T_A = 25^\circ C$ | 500 | | | 500 | | | ns | |

(1) Unless otherwise specified. These specifications apply for $-55^\circ C \leq T_J \leq +150^\circ C$ for the LM195 and $0^\circ C \leq +125^\circ C$ for the LM395.
(2) Selected devices with higher breakdown available.

Submit Documentation Feedback

Figure B.1: LM395 Electrical Specifications

Appendix C

Appendix A6

C.1 GA Table: VSTMIC004

Table C.1: GA Table for VSTMIC004

| Housing Subsystem | Where met |
|---|--|
| GA 3: Engineering Design | Sections 3 and 6 |
| GA 7: Sustainability and Impact of Engineering Activity | Section 6 |
| GA 8: Individual, Team and Multi-Disciplinary Working | See Teams group for meeting minutes |
| GA 10: Engineering Professionalism | All submission activities met including final report a |