

IMPERIAL

Imperial College London

Department of Electrical and Electronic Engineering

EEE Rover - Team 16

Electronics Design Project
ELEC40006

1st Year UG Electrical and Electronic Engineering

Ensar Bati (CID: 02414255)
George Politis (CID: 02388950)
Joel Ng (CID: 02193809)
Koh Pi Rong (CID: 02122328)
Max Ryan (CID: 02462876)
Partha Khanna (CID: 02374670)

Submitted to: Berkay Ozbek

June 5, 2025

Word Count: 9701

Abstract

The aim of this project is to survey several lizards in a given area. The surveying is achieved by using magnetic, ultrasound, infrared and radio sensors to acquire the relevant signals emitted from the lizards that can be used to determine their name and species. The data about the species and name for the lizards must be displayed on a Web-based User Interface. The movement of the rover will be remotely controlled using a control panel hosted on this Interface.

Supplementary Files

Please see the GitHub Repository, containing all the code involved. <https://github.com/parthak314/Team16-EERover2024>

Contents

1	Introduction	3
1.1	Project Breakdown	4
1.2	Planning and Development Strategy	4
1.3	Team members and Roles	6
1.4	Product Procurement and Budgeting	7
2	Implementation	8
2.1	Hardware	8
2.1.1	Ultrasound Sensor	8
2.1.2	Radio Sensor	12
2.1.3	Infrared Sensor	21
2.1.4	Magnetic Sensor	25
2.1.5	Mechanical Design	29
2.2	Software	33
2.2.1	Motor Drive	33
2.2.2	Web User Interface	36
2.3	Integration	38
2.3.1	Physical Integration	38
2.3.2	Software Integration	40
3	Conclusion	43
	References	44
	Appendix	45
	A Risk Assessment	45
	B Schematics	47

1 Introduction

The aim of this project is to design a system capable of moving towards a lizard and identifying its name and species. To achieve this, a rover must be developed that can navigate and detect radio, magnetic (polarity), infrared, and ultrasonic signals, decode these signals, and produce the lizard's name. Key requirements for the rover include:

Mobility: The rover must be capable of precise motion control.

Signal Detection: It should detect and interpret radio, magnetic, infrared, and ultrasonic signals.

Signal Decoding: The rover must decode these signals to extract relevant data.

Remote Control: A web-based user interface (UI) will allow remote control of the rover.

The lizard's four-character name is determined by an ultrasonic transducer, which receives each character encoded in the ASCII character set. These characters are transmitted in UART packets (LSB first) using a carrier frequency of 40 kHz. This setup ensures accurate and efficient decoding of the lizard's name and species information. The species is determined by a combination of the other 3 signals as shown in Table 1.

Species	Infrared	Radio	Magnetic
Abronia	571 Hz		N
Elgaria		120 Hz	N
Dixonius	353 Hz		S
Cophotis		200 Hz	S

Table 1: Species designation

The motion of the rover is controlled by a motor drive chip, from which the following motions states can be defined: forward, reverse, stationary, left turn, and right turn. These manoeuvres are executed using at least two DC motors.

A software-focused solution is also required for the rover's remote control. This involves the development of a web interface that allows the user to control the rover's motion using arrow keys. The interface will also display sensor readings and, based on these values, determine and display the name and species of the lizard. This project aims to comprehensively explore the development of each component: Sensor Creation: Design and implementation of sensors for detecting radio, magnetic, infrared, and ultrasonic signals; Motor Control System: Configuration and programming of the motor drive chip for precise motion control; Web Interface: Development of a user-friendly web interface for remote control and data display; Mechanical Design: Construction of the rover's physical structure, ensuring it is lightweight and functional.

The rover must weigh no more than 750 grams to avoid disturbing the lizards, and the total cost should not exceed £60. With the High Level Requirements [HLRs] listed above, an Acceptance Criteria can be created (Figure 2)

Criteria	
1a	Lizard name understood via an ultrasonic transducer in the required form
1b	Lizard species understood via a combination of magnetic, radio and infrared signals.
2	Financial costs kept to a minimum (under £60) throughout the course of the project
3	Mass of system kept below 750g
4	The rover can be effectively manoeuvred around rocks and objects in the area
5	Remote control interface is easy to use with an interactive UI

Table 2: Acceptance Criteria

The risks associated with this project are detailed in Appendix A - Risk Assessment.

1.1 Project Breakdown

As per the acceptance criteria and the project brief, a project breakdown can be created, as illustrated in Figure 1. This breakdown presents the structure of the project and outlines how each aspect can be delegated to team members. By systematically combining all components, a complete rover can be designed and assembled. The main elements of the project breakdown are covered by sensor design and implementation, Motor control system, Web Interface development, Mechanical Design, System Integration.

By following this structured approach, each team member can focus on their specific tasks, and the integration of all parts will lead to the successful completion of the project.

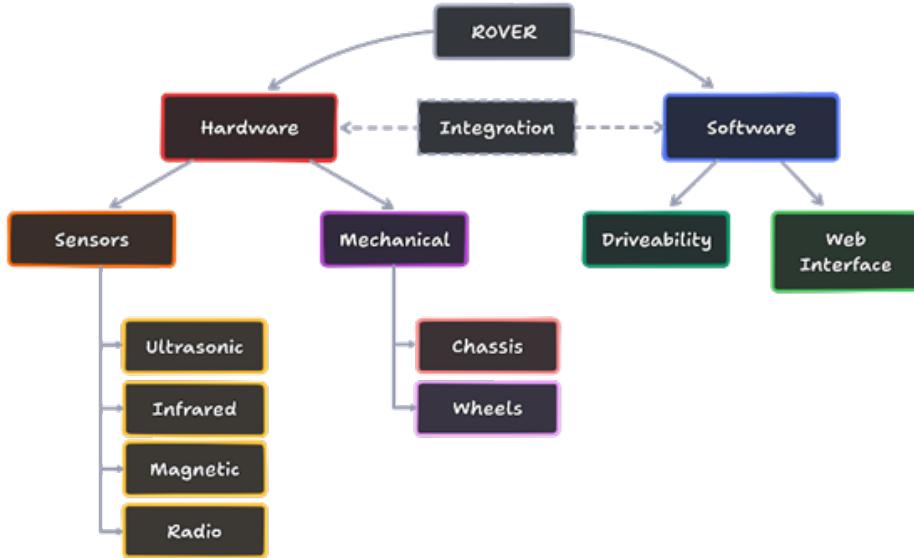


Figure 1: Project Breakdown Diagram.

1.2 Planning and Development Strategy

Throughout this project, the Agile development methodology has been implemented by:

Sprints

The project is broken down into smaller, manageable tasks, which are tackled in short, time-boxed iterations or sprints. With each sprint focusing on the aspects described in Figure 1. This is accompanied with daily stand up meetings to ensure that the progress throughout the report is consistent with the project plan which ensures that each part is developed, tested, and refined before moving on to the next. This promotes the use of cross-functional teams in this project to ensure all aspects are aligned and integrated closely.

Kanban Board

This helps to visualise the workflow and assign tasks to team members. This has been done using a Trello board as can be seen in Figure 2.

This assists in adaptability and flexibility and also promotes dynamic prioritisation. This is due to a new card being issued with a higher priority that a team member responsible for that field can be given. The columns are Backlog (higher priority to-do e.g. if behind on a deadline), to-do (pending tasks), Doing/In progress, Review/Documentation, Testing, Completed. This also helps in reducing the team's workload as a Work in Progress limit can be set.

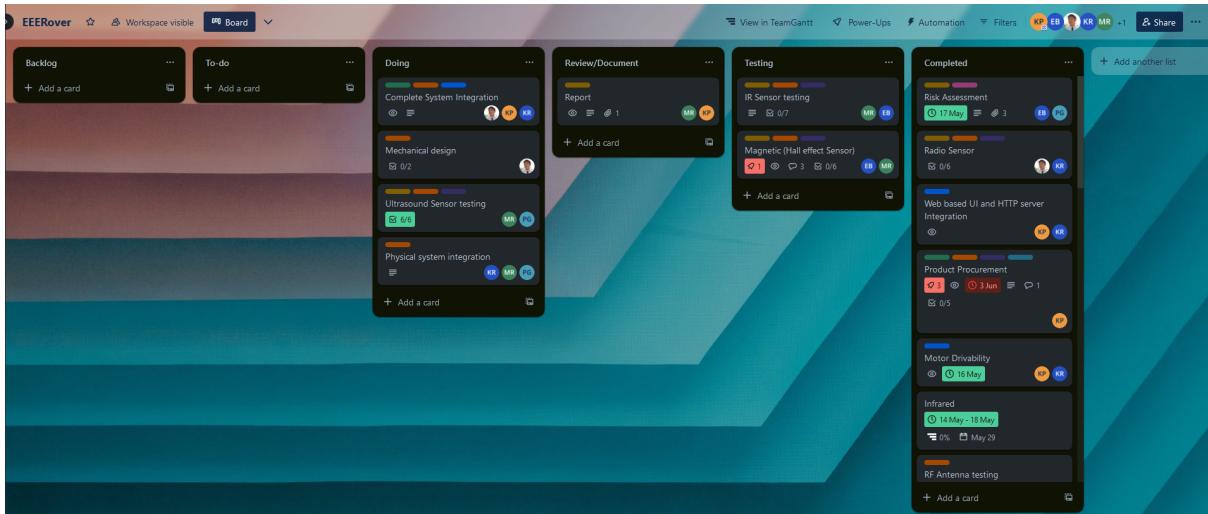


Figure 2: Project Kanban Board implemented using Trello

Gantt Chart

This visualisation shows the timeline of the project and for each of the tasks which are inline with the milestones and deadlines for the key deliverables. This also addresses any dependencies between tasks so that they can be prioritised in the correct order. Retrospective meetings (at the end of the week) can be used to influence any changes that need to be made to the Gantt chart.

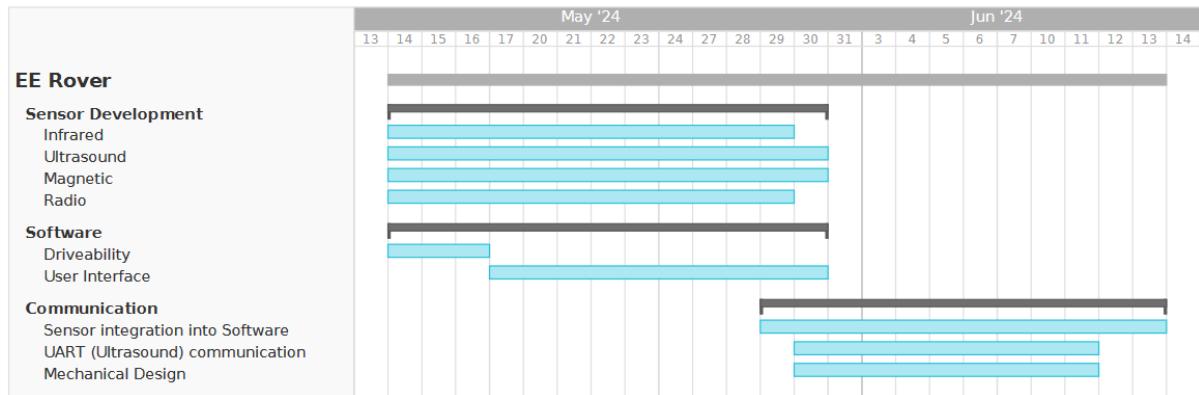


Figure 3: Project Gantt Chart

1.3 Team members and Roles

In this project, roles have been clearly divided to leverage both technical and non-technical expertise effectively, in coordination with each team member's skill set. Technical roles are focused on specific sensors and software integration, ensuring each critical component is developed by specialised team members. Responsibilities include designing and implementing magnetic, radio, infrared, and ultrasound sensors, along with the crucial task of software integration to ensure seamless operation between hardware and software.

Non-technical roles focus on project management and support activities. These include conducting risk assessments and scheduling to keep the project on track, creating detailed schematics for clear and accurate designs, and thorough documentation to capture every aspect of the project's progress. Further additional responsibilities include managing resources effectively as well as ensuring that comprehensive documentation is maintained throughout the project lifecycle.

	Technical Roles	Non-Technical Roles
Ensar Bati	Magnetic Sensor	Risk Assessment
George Politis	Radio Sensor	Risk Assessment and Scheduling
Joel Ng	Radio Sensor and Mechanical Design	Schematics
Koh Pi Rong	Software and Integration	Schematics
Max Ryan	Infrared and Ultrasound Sensor	Documentation
Partha Khanna	Software and Integration	Resources and Documentation

Table 3: Roles and responsibilities

By dividing roles in this manner, the team ensures that both the development and operational aspects of the rover project are handled efficiently, allowing for a coordinated approach to meet project objectives.

1.4 Product Procurement and Budgeting

Product procurement and budgeting are crucial components of this project, ensuring that all necessary sensors and electronic components are acquired within the allocated budget while maintaining high quality and functionality.

The first stage is to identify requirements (for the sensors, op-amps, and any other component required), selecting an appropriate vendor (based on cost, quality and delivery time) from OneCall Farnell, RS and the EED Stores before making a purchase.

Furthermore, a contingency fund was reserved to handle unforeseen expenses such as component failures or any additional parts.

EEE Rover Budgeting Sheet							Provisional Cost	Components		Total
Item	Order Code	Supplier	Order Date	Status	Status last updated	Units	£/unit			
Op-amp (TL072CP)	SI0170	EED Stores	17/05/2024	Received	31/05/2024	£ 10.00	2	£ 0.52	£ 1.04	
Op-amp (MC33078P)	1977383	RS	17/05/2024	Received	24/05/2024	£ -	2	£ 4.11	£ 8.22	
Hall effect Sensor (AH49FZ3-G1)	3942777	OneCall	17/05/2024	Received	24/05/2024	£ 7.50	1	£ 1.00	£ 1.00	
Op-amp (MCP6283-E/P)	1852091	OneCall	27/05/2024	Received	29/05/2024	£ -	3	£ 0.71	£ 2.13	
Op-amp (MCP6002-I/P)	1292245	OneCall	02/06/2024	Received	05/06/2024	£ -	1	£ 0.36	£ 0.36	
Op-amp (MCP6022-I/P)	9758640	OneCall	02/06/2024	Received	05/06/2024	£ -	1	£ 1.49	£ 1.49	
Ultrasonic Sensor (MCUSD16A40S12RO)	2362677	OneCall	04/06/2024	Received	10/06/2024	£ -	1	£ 2.58	£ 2.58	
Op-amp (MCP6022-I/P)	9758640	OneCall	11/06/2024	Placed	11/06/2024	£ -	2	£ 1.49	£ 2.98	
Phototransistor	SD0127	EED Stores	12/06/2024	Placed	12/06/2024	£ -	4	£ 0.21	£ 0.84	
IR receiver (OP505A)	880-1980	RS	12/06/2024	Cancelled	13/06/2024	£ -	0	£ -	£ -	
Phototransistor (TFT4300)	1045526	OneCall	12/06/2024	Placed	12/06/2024	£ -	5	£ 0.60	£ 3.00	
Photodiode (SFH 203 FA)	195-697	RS	13/06/2024	Placed	13/06/2024	£ -	3	£ 0.63	£ 1.89	
Miscellaneous						£ 2.50			£ -	
						ESTIMATE: £ 20.00			SUBTOTAL: £ 25.53	

Figure 4: Project Budgeting sheet

As per the £60 budget, it is critical to estimate the provisional cost of components which was estimated to be around £20. This can be seen in the budgeting sheet and Expense tracker in Figure 4. The actual expenses in the project ended up summing to £25.53. Due to this being higher than the allocated internal budget of £20 by £5.53, the contingency fund was used.

Effective product procurement and stringent budgeting were key to the project's success. By carefully selecting vendors, estimating costs, and managing the budget, the team ensured that all necessary components were procured without exceeding the allocated funds (not exceeding the contingency fund). This disciplined approach not only maintained financial control but also facilitated the timely completion of the project.

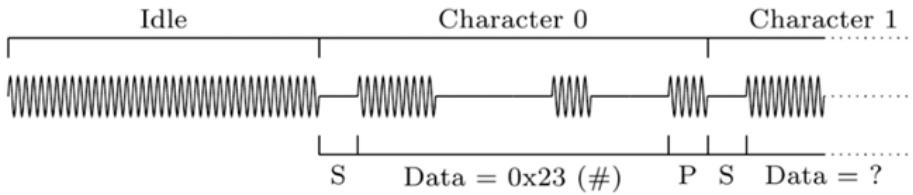
2 Implementation

This explores the conceptualisation, design, development, testing and evaluation for each of the sensors, mechanical design, software and integration.

2.1 Hardware

2.1.1 Ultrasound Sensor

The ultrasound signal uses a carrier frequency of 40kHz which is modulated with a form of amplitude-shift keying. Amplitude-shift keying is a form of amplitude modulation where a high bit is when the amplitude of the signal is not 0 and a low bit is when the amplitude is 0. The data is transmitted at a rate of 600 bits per second using UART packages. Each character is encoded using 10 bits of which the first bit is a start bit which is always low. The next 8 bits represent an ASCII character with the least significant bit first and the final bit is a stop bit which is always high [1]. The name of each lizard is made up of a # followed by 3 characters. The rover must be able to detect the name of the lizard with the sensor placed 2cm from the lizard's mouth where the ultrasound transmitter is located.



S = Start Bit (Logic 0)

P = Stop Bit (Logic 1)

Data transmitted LSB first

Figure 5: Shows how the data is encoded in the ultrasound signal.

An ultrasound transceiver can work as both an emitter and receiver. An ultrasound transceiver works by using a piezoelectric crystal inside the device. When an ultrasonic wave hits this crystal it generates an electrical signal proportional to the pressure of the incoming ultrasonic wave [2]. For this project, it will be used as an ultrasound receiver.

Receiving: The sensor produces a signal with an amplitude of around 200mV as shown in Figure 6. This signal however is too small to pass through an envelope demodulator as this will cause a voltage drop of 0.7 volts due to it having a diode. Unfortunately when the lizard skin is on the module the received amplitude using the sensor when it is 2cm from the lizard's mouth drops down to about 80mV.

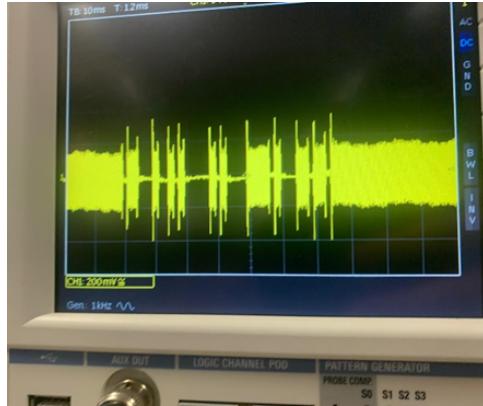


Figure 6: Shows the output signal from the ultrasound transceiver.

Amplification: The signal is then amplified by 63 times using a non-inverting amplifier. A value of 63 was chosen so that the rover could potentially pick up a signal when the sensor is further than 2cm away. In theory the maximum amplification is 42 times as $42 * 0.08 = 3.36V$ so the signal would just about clip. But by using a larger degree of amplification it will enable the rover to pick up the signal from a wider angle and from a further distance. The signal is amplified so that the signal will not be lost when it is passed through the diode. The ultrasonic sensor generates a voltage but a current needs to flow to carry the signal. Since negligible current will flow into the non-inverting op-amp the output of the ultrasonic transducer needs to be connected to ground through a large resistor which has a value of $30k\Omega$. In Figure 7 the blue signal shows the signal after it has passed through the non-inverting amplifier. This amplified signal has an amplitude of 3.3V which is significantly larger than the input signal which had an amplitude of 80mV. In theory this signal should have an amplitude of $0.08V * 63 = 5.04V$ however the power rails were only 3.3V so the signal got clipped. This amplitude of 3.3V is much larger than 0.7V so the voltage drop won't cause the signal to be lost when it passes through the envelope demodulator.

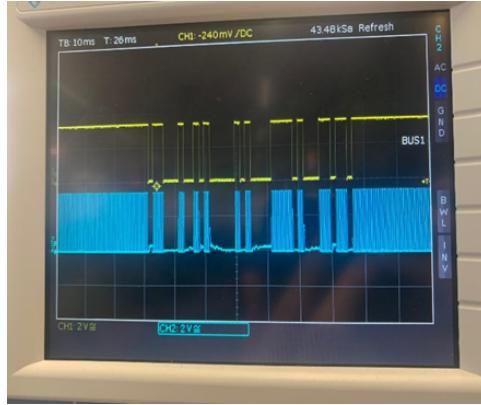


Figure 7: CH2 (blue trace) is the output signal from the non-inverting amplifier. CH1 (yellow trace) is the output signal from the comparator.

Demodulation: The signal is passed through an envelope demodulator which consists of a diode, a $10nF$ capacitor and a $10k\Omega$ resistor as shown in Figure 12. This arrangement gives a corner frequency of 1591Hz (as shown in Figure 8 which is significantly smaller than the carrier frequency of 40kHz and larger than the 600Hz data frequency so it will demodulate the signal well. In Figure 9 the blue signal is the output from the envelope demodulator and it shows that the signal has been demodulated well.

$$f_0 = \frac{1}{2\pi RC} = \frac{1}{2\pi * 10n * 10k} = 1591Hz$$

Figure 8: Formula for envelope demodulator.

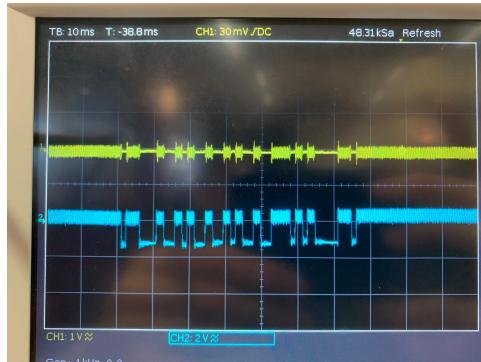


Figure 9: CH2 (blue trace) is the output signal from the envelope demodulator. CH1 (yellow trace) is the output signal from the ultrasound transducer.

Analogue to Digital Conversion: The signal is passed through a comparator so that if the input signal is higher than the threshold V_{out} is set high otherwise V_{out} is low so producing a DC signal which the microcontroller can interpret to find the name of the lizard. The yellow signal in Figure 7 shows the signal after it has passed through the amplitude demodulator and the comparator. This signal is a digital signal which can be decoded using the microcontroller to find the name of the lizard.

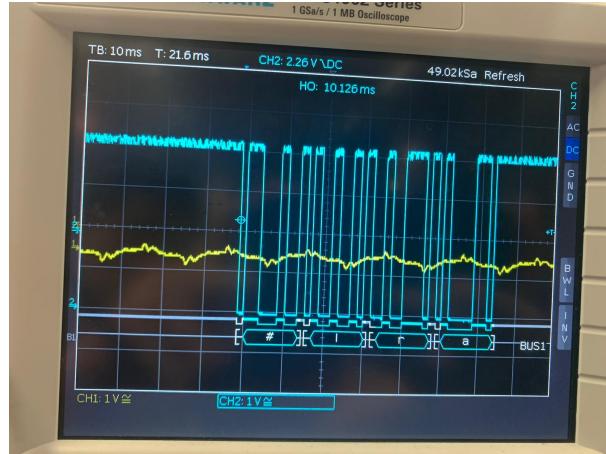


Figure 10: The digital output signal from the comparator interpreted using UART to get the lizards name which was taken with the lizard 8cm away as shown in Figure 11.

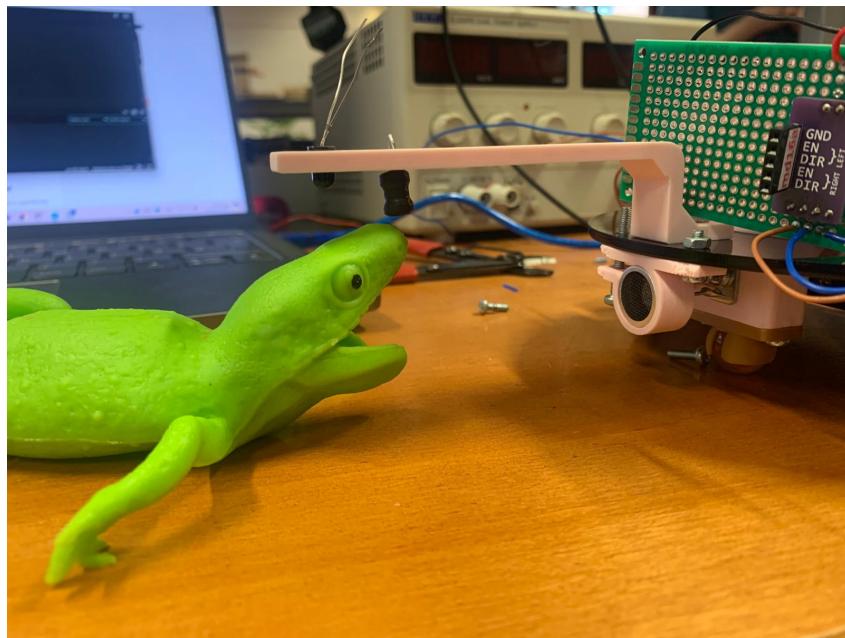


Figure 11: Testing the range of the ultrasonic sensor.

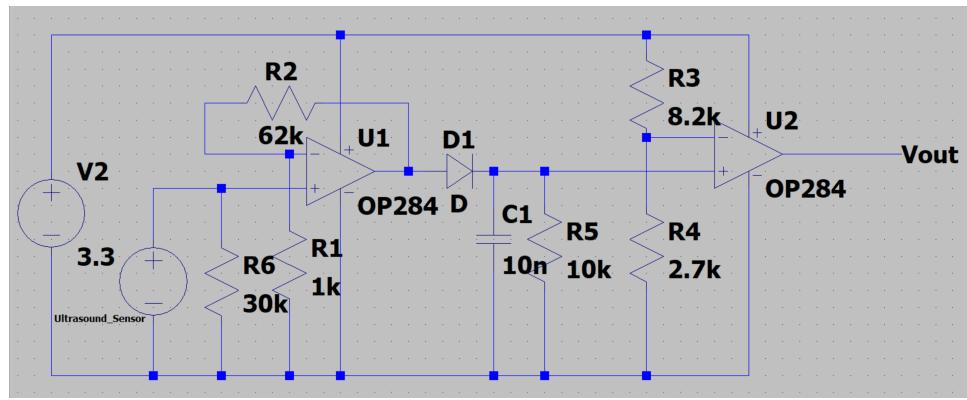


Figure 12: Circuit diagram for the ultrasound amplifier and decoder.

The code to interpret the UART signal uses the `Serial.read()` function to decode each character. For further details on how the code works see the Software Integration section. The ultrasound sensor circuit performs as intended and allows the rover to detect the ultrasound signals very reliably from approximately 10cm away which is significantly further than the 2cm range that was the initial aim.

2.1.2 Radio Sensor

The rover must be able to receive, demodulate and extract information on the lizard species from two Amplitude Modulated (AM) radio signals as seen in Table 4 from a distance of about 5cm.

Species	Carrier Frequency (kHz)	Modulating Frequency (Hz)
Elgaria	89	120
Cophotis	89	200

Table 4: Radio Frequency Encoding

An inductor can be used as the radio receiver, as it is simply a coil, obeying both Faraday's and Lenz's Laws and as such can be used as an antenna to receive the electromagnetic signal. The electromagnetic radio wave can be converted into an electrical signal using an inductor. The antenna works because an incident electromagnetic wave generates an oscillating electric field which induces a current and as a result a voltage. Therefore, the electromagnetic wave can be depicted with voltages which oscillate in the same way the electromagnetic signal does [3]. Having detected the signal, circuit components can be used to extract certain frequency components "hidden" within the signal which in this project is necessary for the species of the lizard to be identified.

An initial implementation was to use a tuned coil wire antenna integrated in the receiver circuit. However after completing many testings on the different combinations of characteristics of the wire based on the inductor characteristics highlighted in this equation,

$$L = \frac{\mu N^2 A}{l}$$

no antenna had an inductance close or equal to the minimum inductance of an inductor found in the laboratory components. As such, based on the Quality factor equation

$$Q = \frac{1}{2\zeta} = \frac{1}{R} \cdot \left(\frac{L}{C} \right)^{1/2}$$

a lower inductance would lead to a more distorted signal; thus, to maximise the clarity of the signal a higher inductance was preferable while maintaining the required resonant frequency and due to the low inductive impedance, this coil wire was replaced with an inductor.

Receiving: An RLC resonant circuit that resonates at 89kHz was used to receive the radio signal through the inductor. Following the corner frequency expression: $\omega_c = \frac{1}{\sqrt{LC}}$, the circuit can be designed to resonate at the right frequency as shown below.

$$\omega_c = \frac{1}{\sqrt{3.3\text{mH} \cdot 1\text{nF}}} = 550481\text{rad/s}$$

$$f = \frac{550481}{2\pi} = 87.6\text{kHz}$$

The resonance frequency of the circuit calculated above deviates 1.6% from the target which is justified by the limited variety of capacitances and inductances available.

The signal received will have a very low amplitude due to the signal generator being located inside of the lizard so the lizard skin absorbs some of the signal as well as there being some distance between the signal generator and the receiver. The value of the resistor was chosen to maximise the Q value of the resonance such that the circuit attenuates sharply at frequencies deviating from 89kHz as can be seen from the equations below:

The transfer function for the receiver stage is:

$$H(jw) = \frac{jwL}{jwL + \left(\frac{1}{jwC}\right) + R} = \frac{(jw)^2 \cdot L \cdot C}{(jw)^2 \cdot L \cdot C + jwRC + 1}$$

From the denominator, let:

$$\begin{aligned} a &= LC, & b &= RC, \\ c &= 1, & \omega_c &= \left(\frac{c}{a}\right)^{1/2} = (LC)^{-1/2} \end{aligned}$$

The damping factor is given by:

$$\zeta = \frac{b}{2a\omega_c} = \frac{RC}{2(LC)^{1/2}} = \frac{R}{2} \left(\frac{C}{L}\right)^{1/2}$$

And the quality factor is:

$$Q = \frac{1}{2\zeta} = \frac{1}{R} \cdot \left(\frac{L}{C}\right)^{1/2}$$

Given:

$$R = 10 \text{ k}\Omega,$$

$$L = 3.3 \text{ mH},$$

$$C = 1 \text{ nF}$$

The following values are calculated:

$$\omega_c = 55.0 \text{ k rad/s},$$

$$\zeta = 2.75 \text{ Ns/m},$$

$$Q = 0.18$$

The RLC circuit design can be seen below in Figure 13 and the output signal from the RLC circuit can be seen in Figure 15 however in this image the lizard module is right next to the inductor but as the lizard module is moved away the amplitude of the signal fall rapidly.

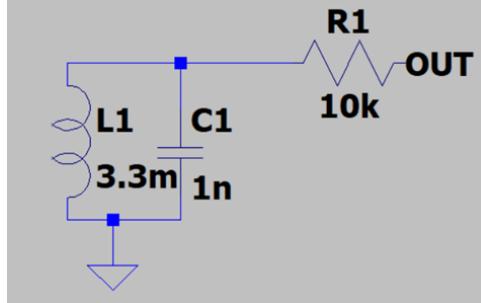


Figure 13: RLC circuit for receiving and attenuating the incoming radio signal

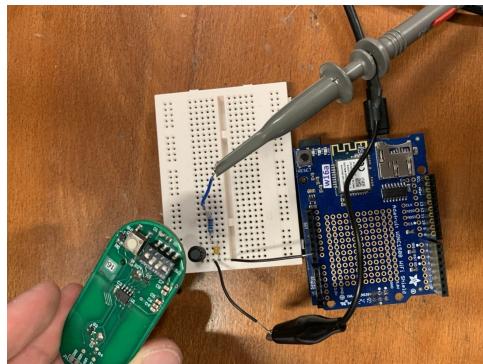


Figure 14: RLC circuit testing

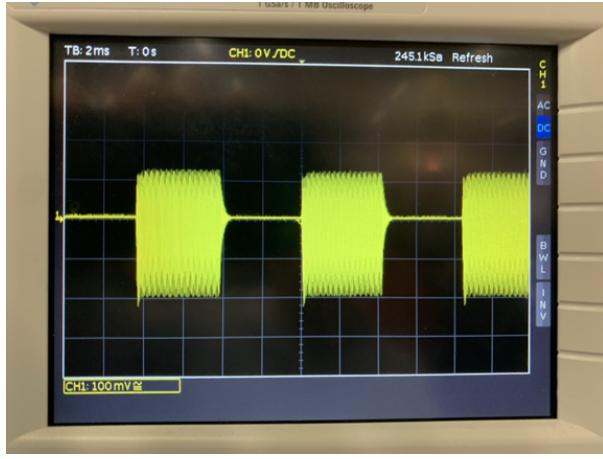


Figure 15: Output from RLC circuit

Amplification: Furthermore, it was necessary to amplify the input signal received from the radio waves, with the aim of being able to read the frequency of the modulated signal in the radio waves from at least 10cm away from the lizard. Noting that the demodulation process would further attenuate the signal. Hence, it is important that the input is attenuated prior to demodulation.

It is important that the amplifier that is chosen has a high slew rate to match the rectangular modulating input. The amplifier must also have a large gain-bandwidth product to ensure that a suitable gain can be achieved, even at the high carrier frequency of the radio waves. A non-inverting amplifier with a gain of 10 was used to amplify the received signal which then passes through the demodulator. This gain was decided since the signal received had an amplitude around 150mV based on Figure 15 and an amplitude of 1V-2V would be adequate for the remaining stages a gain of 10 was selected. The MCP6022 op-amp was chosen for its high gain bandwidth product of 10MHz and high slew rate of $7V/\mu s$.

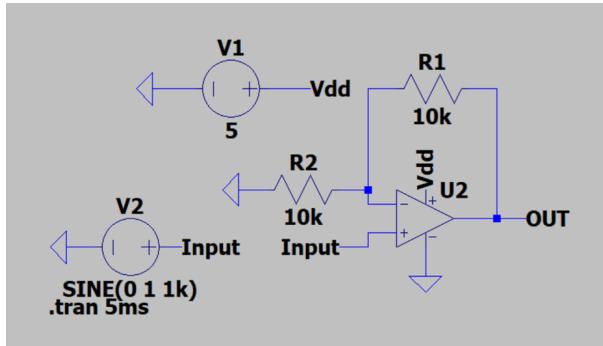


Figure 16: Schematic for the non-inverting amplifier with gain of 2

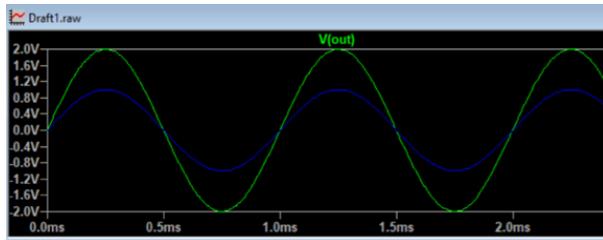


Figure 17: Simulation of the non-inverting amplifier

In the above simulation, a different operational amplifier to the actual was used, due to the limited variety of the simulation software. As such, the amplifier would produce clear signals if low amplitude signals were inserted, hence the simulation was used for demonstration purposes of how a non-inverting amplifier looks and what the relationship between the input and output signals is. In a nutshell, a

different amplifier was used in the simulation to the actual amplifier since it wasn't available in the simulation software and this led to a decrease in the gain from 10 to 2 (solely for the simulation) as the amplifier would distort low amplitude input signals. Therefore, a high amplitude input signal was amplified with a gain of 2 so that the output would be below the 5V positive rail.

The values of the resistors were chosen based on the equation below:

The gain is given by:

$$G = 1 + \frac{R_1}{R_2}$$

Since the gain is 10,

$$R_1 = 9 \cdot R_2$$

Based on the resistances of the resistors available in the laboratory, the resistance of R1 became $R = 27k\Omega$ and the resistance of R2 became $R = 3k\Omega$, since both resistances had to be large enough to avoid short circuits, making the $k\Omega$ band suitable. Adding the non-inverting to the receiver, the following circuit is built.

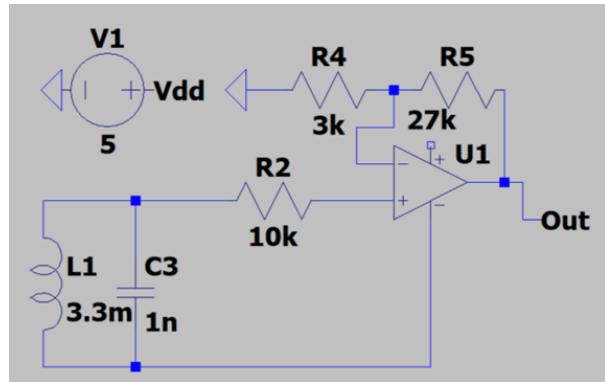


Figure 18: Schematic for the non-inverting amplifier with gain of 9

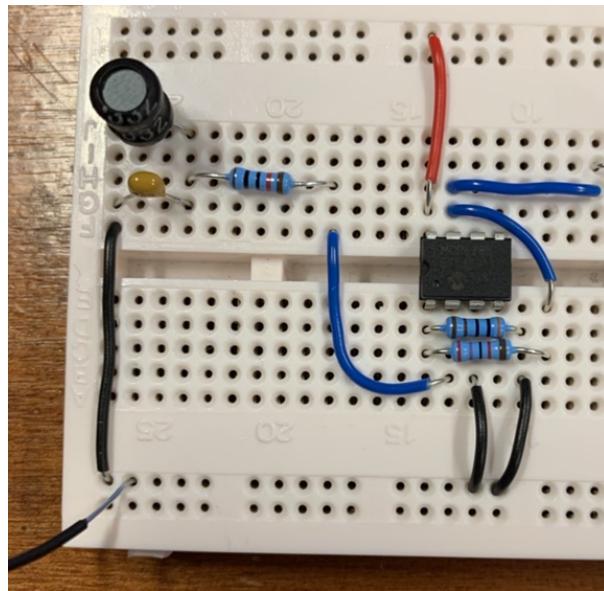


Figure 19: Receiver and amplifier connected

Demodulation: For the demodulation phase, a simple envelope detector circuit was used to extract the modulating frequency from the signal. A simple RC low pass filter has a corner frequency expression of $\omega_c = \frac{1}{RC}$. The corner frequency must be between the modulating frequency and carrier frequency of the signal so that the modulating frequency can be extracted.

$$\omega_c = \frac{1}{2M\Omega \times 150\text{pF}} = 3.33 \text{ k rad/s}$$

The frequency is:

$$f = \frac{\omega_c}{2\pi} = \frac{3.33 \text{ k}}{2\pi} = 531 \text{ Hz}$$

A modulating frequency of 531Hz will demodulate the signal nicely as it will remove the carrier frequency but not affect the modulated signal.

Critically, the envelope detector uses the charge and discharge cycle of the capacitor to “smooth out” the output signal. The resistor is in place to discharge the capacitor to ground. Hence, the resistor value must be chosen carefully. A resistance that is too low would short the entire output signal to ground, while a resistance that is too high would lead to distortion, as the current signal would be very small and prone to interference. A resistor value of $2M\Omega$ was chose as that value was where there was minimal attenuation during simulation which can be seen in Figure 20. Using a modulating signal with an amplitude of 1V the output signal was about 380mV with most of the voltage drop being caused by the diode as can be seen in Figure 22.

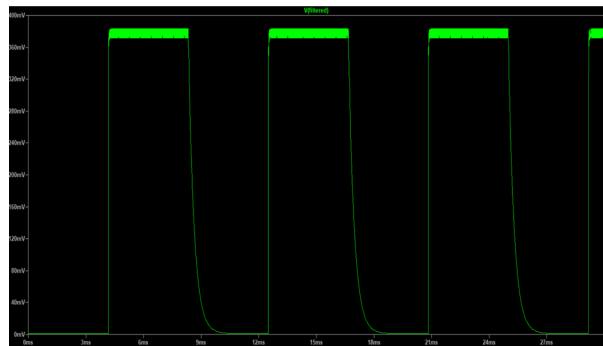


Figure 20: Simulation of circuit in Figure 21

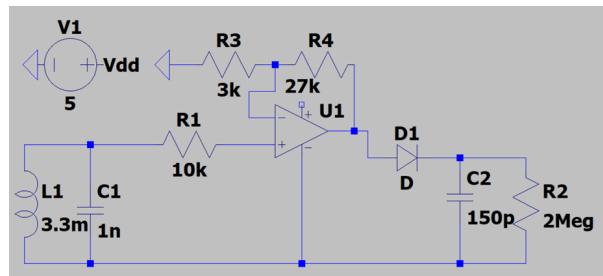


Figure 21: Schematic including the envelope demodulator

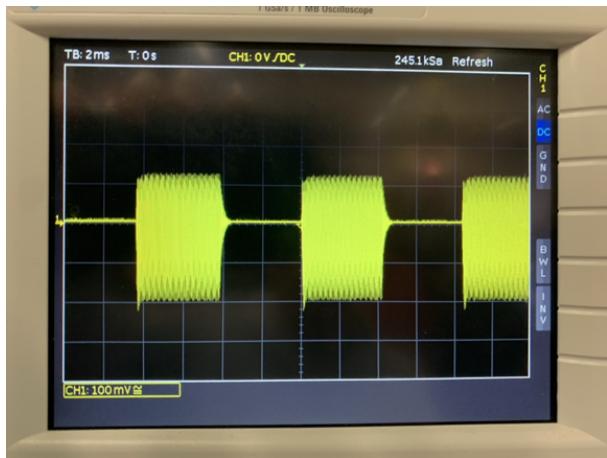


Figure 22: Output from envelope demodulator circuit.

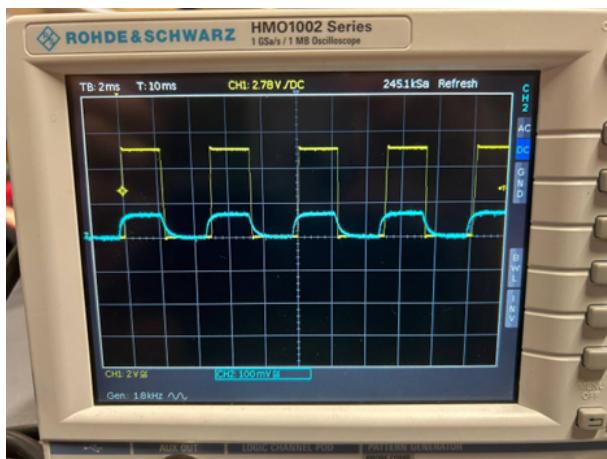


Figure 23: Signal output from the envelop demodulator

However there is some rounding of the rectangular signals due to the transients caused by the capacitor. To rectify this a comparator can be used.

The schematic of the three stages can be seen in Figure 24 below.

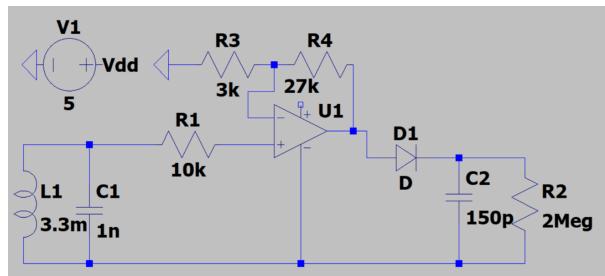


Figure 24: Schematic of circuit with the three stages

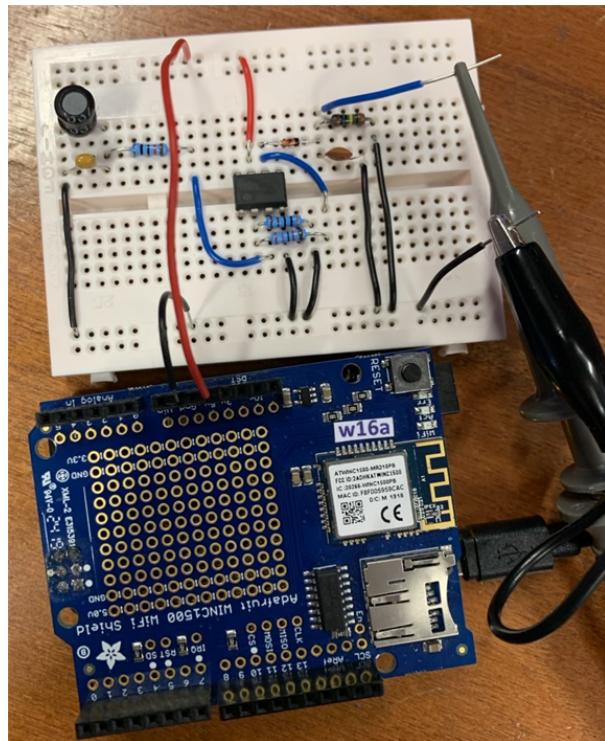


Figure 25: Circuit with the three stages

Analogue to Digital: Finally, to convert the signal into a clean square pulse train to extract the frequency from, a comparator was used to convert the analogue signal to a digital signal by pushing the demodulated signal to either 0 or 5V.

A schematic of the comparator and its simulation can be seen in Figure 26 and Figure 27 respectively below.

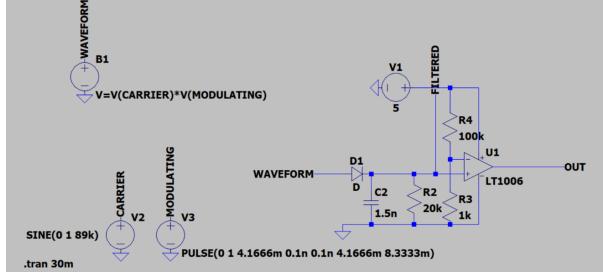


Figure 26: Schematic of the comparator stage



Figure 27: Simulation of the comparator

A threshold value of 50mV was chosen such that it is just low enough to pick up the small output signal from the envelope detector, but large enough such that it does not run the risk of amplifying noise in the circuit.

As can be seen in the output signal of the envelope detector has been successfully amplified converting it into an easy-to-read square pulse train.

The modulating frequencies of 120Hz for the Elgaria species and 200Hz for the Cophotis species can be recovered as seen in Figure 28.

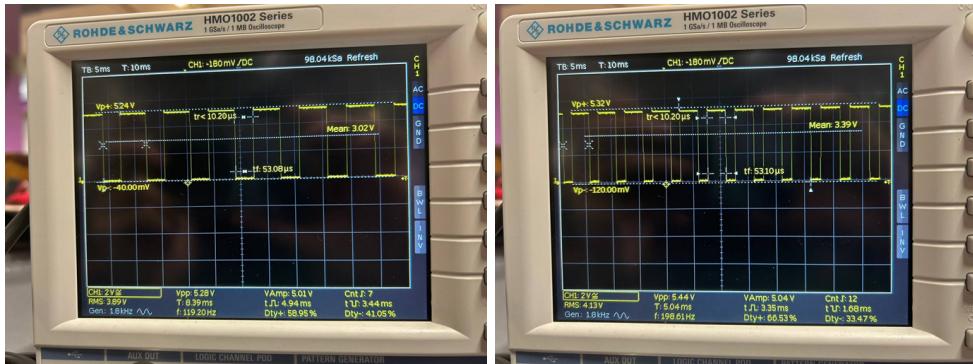


Figure 28: Signal output from comparator for the 2 different species.

It is observed from the oscilloscope that the circuit can receive the expected modulating frequency to a reasonable deviation value. This deviation is small enough that it could be attributed to the accuracy of the oscilloscope. Crucially, the deviation sees no overlap between the Elgaria and Cophotis species, facilitating the identification of each species. In addition when the lizard is brought out of range from our circuit, the entire output immediately collapses to 0V. This is expected behaviour for a comparator, since once the output amplitude falls below 50mV, the entire output should fall to 0V. This is ideal for the metro board, because any noise outside the operating range will not be picked up. The complete circuit of the sensor is shown below in Figure 29.

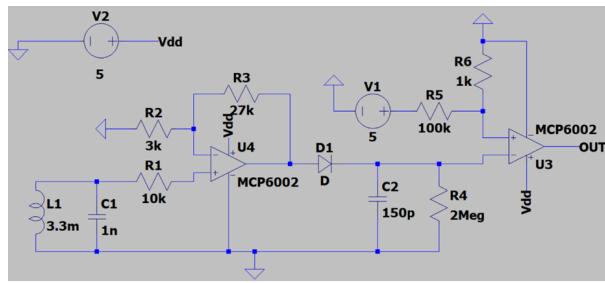


Figure 29: Schematic for the Radio Sensor

The period of the output of the sensor will then be measured using a program, which then will be inverted to calculate the frequency of the signal. Further information about the integration of the radio sensor with the Metro board can be found in the Software Integration section.

2.1.3 Infrared Sensor

The rover must be able to detect the frequency of an infrared pulse with a wavelength of 950nm emitted from the back of the lizard from a distance of about 5cm.

Species	Infrared Frequency
Abronia	571Hz
Dixonius	353Hz

Table 5: Caption

There are two main types of infrared sensors. These are thermal and quantum infrared sensors.

Thermal sensors are designed for detecting infrared signals with a wavelength of about 2 to 14μ and the infrared signal that the rover must detect is 950nm so a thermal infrared sensor is not suitable for this application [4][5].

Quantum infrared sensors however can be used to detect a large range of frequencies but are usually designed for 850nm to 1100nm applications as such a quantum infrared sensor will work very well. Quantum infrared sensors make use of the photovoltaic effect to detect the infrared signals. There are 2 main types of these quantum infrared sensors these being photo-diodes and photo-transistors. The photo-diode has a big advantage over a photo-transistor in that it has a quicker response time and although a photo-transistor is more sensitive, this is just due to the fact that it is a transistor so will naturally amplify any input signal, but another transistor can be used to amplify the signal from the photo-diode. Since the photo-transistor will most likely need amplification as well due to the fact that the infrared signal leaving the lizard is very weak a photo-diode would be better choice. In addition an infrared photo-diode is cheaper than a photo-transistor [6][7]. There are 2 types of photo-diodes PN and PIN photo-diodes. The first has a standard PN junction like a regular diode but a PIN photo-diode has an undoped region in-between the p and n-type doped regions. The advantages of a PIN diode is that they are more sensitive although they do have a slightly slower response time but they are still faster than photo-transistors. In addition PIN photo-diodes will create a cleaner signal due to the having a lower capacitance [8][9].

Photo-diodes are connected in reverse bias. PIN photo-diodes have a PIN junction which is an intrinsic (non-doped) pure silicon region in between the heavily doped p-type and n-type regions. When the photo-diode is connected in reverse bias the depleted region increases in size as the electrons from the n-type region are pulled towards the positive terminal of the photo-diode and the holes in the p-type region are pulled towards the negative terminal. Due to the fact that the depleted region's size is increased it allows for more photons to be absorbed as photons are absorbed in the depleted region. Since more photons can be absorbed the photo-diode becomes more sensitive to light. When photons of sufficient energy hit the depleted region, an electron-hole pair is created and the electron and electron-hole are separated by the electric field within the diode, significantly increases the reverse current flowing through the photo-diode [10][11].

A photo-diode doesn't inherently filter out different wavelengths of light as long as the photons have enough energy to promote an electron from the valence band to conduction band. In order to promote an electron from the valence to conduction band it must have energy greater than 1.12eV as that is the band gap of silicon. A photon which has a wavelength smaller than 1100nm has sufficient energy to promote an electron across the band gap and as such As such a plastic casing is placed around the photo-diode which is designed to only let a certain range of wavelengths through [12]. A photo-diode can be modelled as having a very high resistance of around $200\text{M}\Omega$ when there is no IR light hitting it and when IR light is hitting the photo-diode the resistance is around $4\text{M}\Omega$ but this value varies depending on the intensity of the light.

Receiving: A voltage divider with a pull-up resistor was used and the output was taken across the IR sensor. This means that when there is a spike in the amount of infrared light hitting the IR receiver the voltage across the IR sensor decreases.

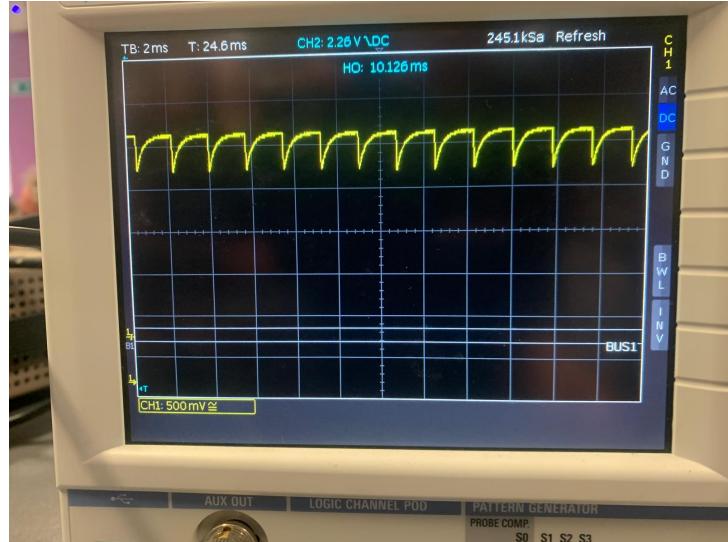


Figure 30: Raw signal output across the photo-diode when it is in the potential divider.

Amplification: Due to the IR LEDs being inside the lizards the amount of IR light that leaves the lizard is significantly reduced. So, it was necessary to amplify the signal. An NPN BJT was used to amplify the signal. As can be seen in Figure 31 once the signal has been amplified by the BJT it is much more visible. Due to the fact that for an NPN BJT, if the base current increases, the current through the collector increases proportionally. This means that if the base current increases the voltage at the emitter decreases as there is a larger voltage drop across the resistor connected between V_{cc} and the emitter. So if the output of the BJT is taken from the emitter when IR light hits the sensor the output from the amplifier will have a spike that starts from 0V and returns to 0V which is the opposite of what was happening from the output of the voltage divider.

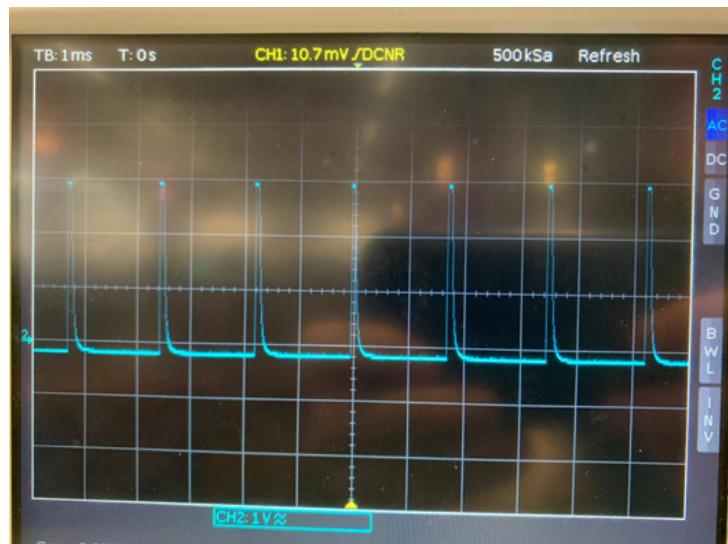


Figure 31: Signal after IR signal has been amplified by the BJT.

Analogue to Digital: Then the output is passed into a comparator. The comparator's output is high once the input signal has passed a preset threshold which allows for the microcontroller to detect when there is a pulse despite the pulse still being quite small when it leaves the BJT if the IR sensor isn't directly above the lizard. This makes the signal a digital signal which makes it easier to calculate the frequency of the signal as on the microcontroller using the `pulseIn(inputPin, HIGH)` and `pulseIn(inputPin, LOW)` functions, the time period of the signal can be calculated which is the reciprocal of the frequency. As such the frequency of the signal can be found. For further details on how the code works see the Software Integration section.

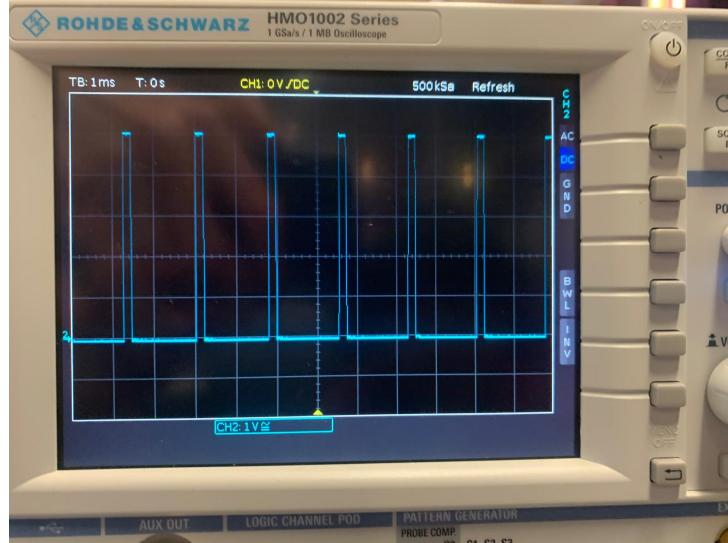


Figure 32: Output from the comparator.

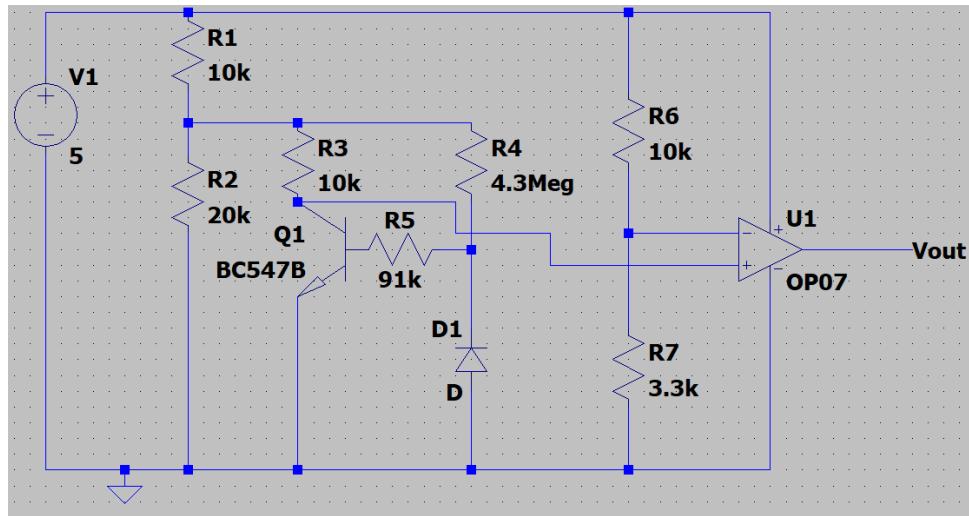


Figure 33: Schematic for the IR sensor.

The rover is able to detect the IR signals from the lizard from a distance of 3cm which exceeds the initial aim of 2cm as can be seen in Figure 34.

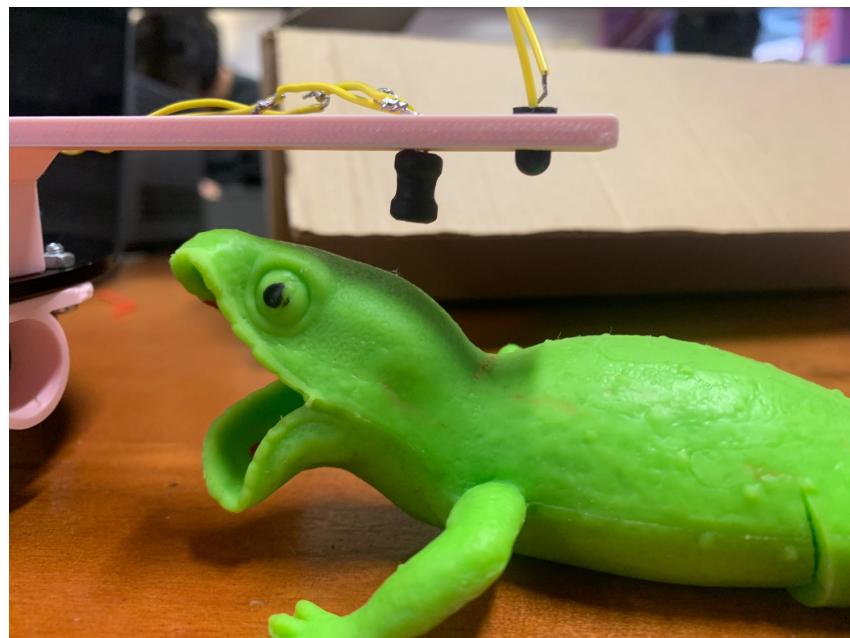


Figure 34: IR sensor range test with the sensor on the probe.

2.1.4 Magnetic Sensor

The rover must be able to detect the polarity of a magnet within the mouth of the lizard from a range of 5cm.

In order to achieve this a magnetic transducer is required. Most magnetic sensors like magnetic resonance sensors and unipolar hall effect sensors only detect the magnitude of the strength of a magnetic field and not the polarity. For the scope of this project, determining the polarity is prioritised.

Bipolar hall effect sensor can detect polarity, however the output only changes when the polarity changes. This would result in unreliability for sensor readings as it is not understandable whether the previous lizard has the same polarity or if the sensor reading has failed. As such a linear hall effect sensor was chosen as they can determine the polarity of a magnetic field as well as being very accurate. The only draw back of hall effect sensors is their relatively short range but this can be remedied by amplifying the output signal [13].

When determining the polarity of a magnetic field, Hall effect sensors stand out as the most effective and practical choice. Unlike magnetoresistive (MR) sensors which detect magnetic fields by changes in electrical resistance or variable reluctance (VR) sensors which rely on changes in magnetic flux and typically require motion to function, Hall effect sensors provide a direct measurement of magnetic polarity.

A linear hall effect sensor relies on the magnetic force that acts perpendicularly on the current flowing through the hall effect sensor which induces a voltage [13]. Furthermore, the Hall effect sensor is less susceptible to environmental conditions such as temperature and ensures consistent performance while being cost effective.

A hall effect sensor has 3 pins which are Vcc, ground and the output pin. When the hall effect sensor is not in the presence of a magnetic field the output pin has a voltage of $\frac{V_{cc}}{2}$. When a south pole is brought near the front side of the hall effect sensor the output voltage increases proportional to the magnetic field strength and the opposite occurs for a north pole. The Type A version of Diodes Inc. AH49FZ3-G1 was chosen as it was a linear hall effect sensor that had a very high output voltage sensitivity with a typical value of about 2.1mV/Gauss as can be seen in the sensor datasheet in Figure 35.

Electrical Characteristics (@ $V_{CC} = 5V$, $T_A = +25^\circ C$, unless otherwise specified.)

Symbol	Parameters	Conditions	Min	Typ	Max	Unit
I_{CC}	Supply Current	—	2	3	4	mA
V_{NULL}	Quiescent Output Voltage	$B = 0$ (Gauss)	2.25	2.5	2.75	V
V_{SEN}	Output Voltage Sensitivity	$B = 0$ to ± 600 (Gauss)	1.7	2.1	2.5	mV/Gauss
V_{OUT_S}	Output Voltage Span	—	1.0 to ($V_{CC} - 1.0$)	0.8 to ($V_{CC} - 0.8$)	—	V
R_{OUT}	Output Resistor	—	—	60	120	Ω
B	Linear Magnetic Range	—	± 500	± 800	—	Gauss
—	Linearity of Span	—	—	0.7	—	%
—	Output Noise	Bandwidth=10Hz to 10kHz	—	90	—	μV

Figure 35: AH49FZ3-G1 hall effect sensor datasheet [14].

Reading sensor values: With no amplification the hall effect sensor was able to determine the polarity of the magnet with a range of 2cm as shown in Figure 36. This range needs to be increased to at least 5cm so that it can be more easily detected by the rover. In order to determine the polarity of the magnet the current value of the voltage output from the hall sensor is compared with the base value of the voltage output of the hall effect sensor and if the current value is higher than the base value the hall effect sensor is facing a south pole and if it is lower then the sensor is facing a north pole.



Figure 36: Hall effect sensor test with no amplification.

Amplification of the signal: The hall effect sensor only used about 2 to 3% of the total range of the voltage that was input centred around $\frac{V_{cc}}{2}$ when the magnet was 2cm away from the lizard due to the low strength of the magnet and the lizard skin reducing the magnetic field strength. By extending this range so that it covered almost all of the voltage range from V_{cc} to 0V it would enable the microcontroller to determine the polarity of a magnet from a greater range. This is necessary as the microcontroller analogue pins only have a resolution of 10 bits which is spread evenly between 0V and 5V. This means that the microcontroller can only measure changes of 3mV using its analogue input pins. So if the hall effect sensor output changes by a value less than 3mV, the microcontroller wouldn't be able to detect it. So it was necessary to amplify the signal so that these small changes to be detected.

In order to achieve this, a TL072 operational amplifier was used as a differential amplifier. Compared to its alternatives such as a non-inverting amplifier with a voltage offset, it provides more reliable data due to its high common-mode noise rejection. For the configuration of the differential amplifier, an option was to use two hall effect sensors inversely located and amplify the difference between them. However, since the rover will not always move directly to the lizard, the sensors' distances to the lizard will vary. Especially in closer ranges, the magnetic field is highly dependent on the distance. This would create a complication that may be solvable but unnecessary, since the range obtained with only one sensor is beyond the aimed range. The configuration decided is by amplifying the difference between the sensor reading and base voltage of 2.5V provided by a potential divider. A feedback resistor is connected to the inverting pin to provide negative feedback to the op-amp, so that the sensor is stabilized around 2.5V. And the resistor values are decided to create a gain of -55 which is calculated by comparing the number of bits utilized and the number of bits the analogue pin provides. The final circuit diagram can be seen in Figure 37. This setup allows for the hall effect sensor to use the full 5V voltage range and values reached out of this scope cause saturation. As a result of this amplification, the range of the magnetic sensor was 12cm which far exceeds our 5cm requirement. On top of this, there is yet more room to improve this range in software.

Noise filtering software: There was noise in the input to the microcontroller due to the large amount of amplification. This noise was in the form of a random slight deviation from the actual value that the sensor should be reading. As such the best way to remove this noise was to use a moving average that constantly updates to give a smoother more stable output by averaging out the noise as can be seen in Figure 38. This managed to increase the range to about 15cm however a slight drawback of this method is that it causes a slight delay in determining the polarity of a magnetic field of about 1 second. The improvement over the base magnetic sensor can be seen in Figure 39 where there is a dramatic improvement in range compared to just the hall sensor with no amplification. For further details on how the code works see the Software Integration section.

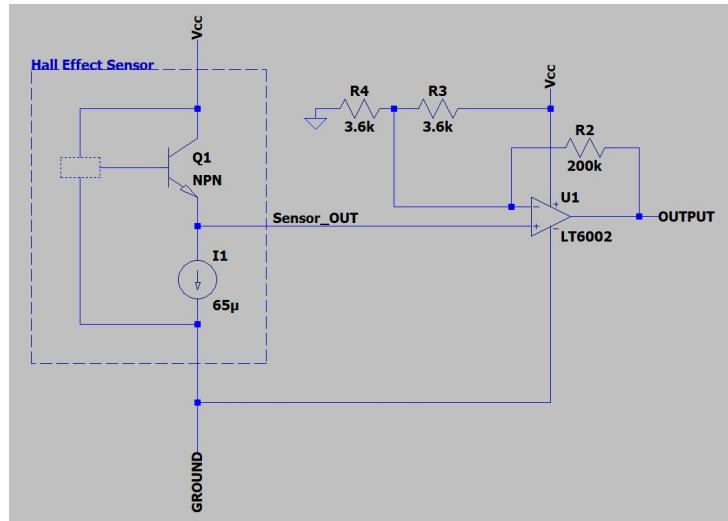


Figure 37: Hall effect sensor schematic.

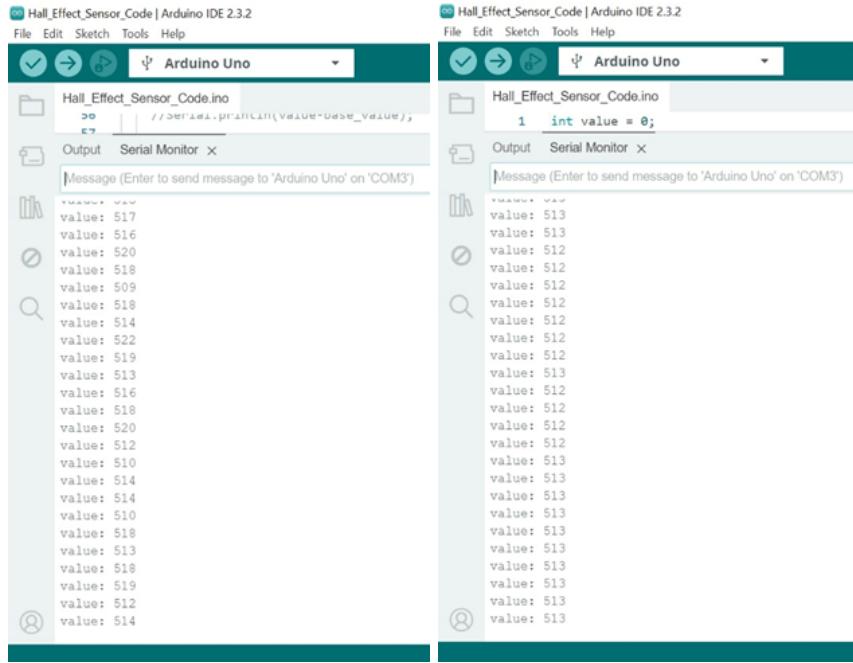


Figure 38: Compares the output readings from the microcontroller with (right) and without (left) the algorithm in the presence of a steady magnetic field.

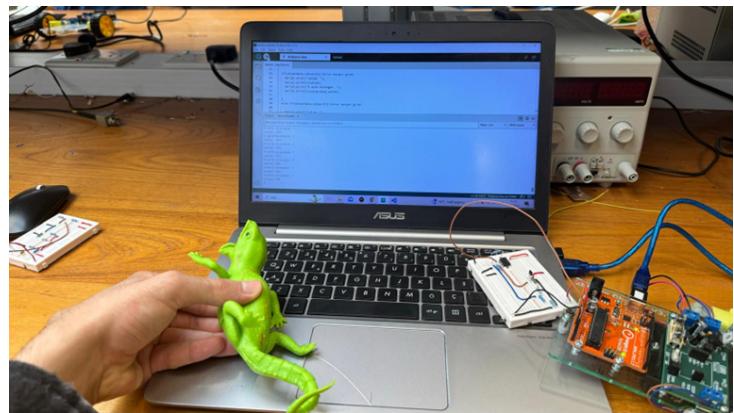


Figure 39: Hall effect sensor test with amplification and the improved software algorithm.

2.1.5 Mechanical Design

When designing the chassis and physical integration of the rover, there were three principal considerations as laid out in Table 6 below.

Principal Consideration	Rationale
Ensure that the sensors can be brought as close as possible to lizard	This ensures that the sensors will be within operating range from the lizard transmitter, allowing us to pick up the signals from the lizard to accurately classify them.
Ensure that the rover is not exceeding 750g in weight	This is a critical requirement from the project brief, where certain lizards will refuse to output their signal when the rover weights more than 750g.
Ensure that the rover is manoeuvrable	As the enclosure is fitted with obstacles, making the rover manoeuvrable is a priority, as it allows us to comfortably negotiate the enclosure. Further, a manoeuvrable rover allows us to control the rover such that the sensors can be brought as close as possible to the lizard.

Table 6: Considerations for the mechanical design

Consideration 1: Minimise sensor proximity to lizard

To ensure that the sensors are placed optimally, the locations of the various signal emitters within the lizard need to be considered. Their locations are as follows: the ultrasonic signal and magnetic signal are radiated from the mouth of the lizard, the radio signal is emitted along the plane of the transmitter, most strongly in the belly of the lizard and the infrared signal is emitted from two different points along the back of the lizard.

With these factors in mind the infrared and radio sensors will be placed on a probe so they can be positioned closer to the belly of the lizard when the ultrasonic sensor is at the lizard's mouth. As the polarity axis of the magnet in the lizard's mouth is from top to bottom, the magnetic sensor will also be mounted on the probe, close to the chassis and facing down, to better capture the magnetic field.

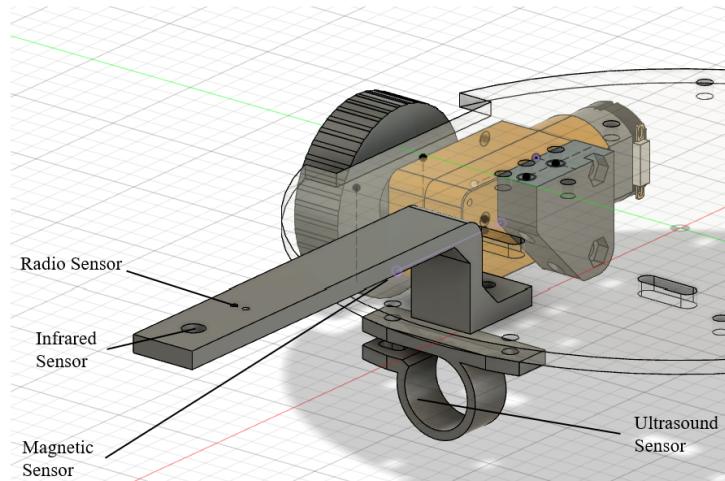


Figure 40: Implementation of probe and ultrasonic sensor mount.

Consideration 2: Rover does not exceed 750g

Motor and Battery: These components are the largest contributors to weight, where the motors and batteries combined contribute close to 200g of weight. These components are positioned in the centre of our rover to ensure that rover is well balanced.

Metro Board, Sensor Electronics and Circuits: The metro board and Wi-Fi attachment weigh in at around 50g. For the sensor circuits, the largest contributor to weight are the breadboards. These boards were hence swapped out for perforated boards, significantly reducing the weight of the circuits.

Chassis: The rest of the weight allowance can be used to design the chassis. A new chassis was designed for better manoeuvrability, as explained in the next section. The chassis is circular, with a radius of 80mm, just marginally larger than the original chassis. However, the design is far easier to steer. The chassis is laser cut from 3mm acrylic, similar to the original chassis.

To connect the perforated boards, motors and metro board to the chassis, supports were 3D printed, ensuring that these connecting blocks are light and inexpensive.

As the chassis has less usable space, it was not possible to lay out the perforated boards and metro board flat on the chassis. The boards were hence mounted vertically.

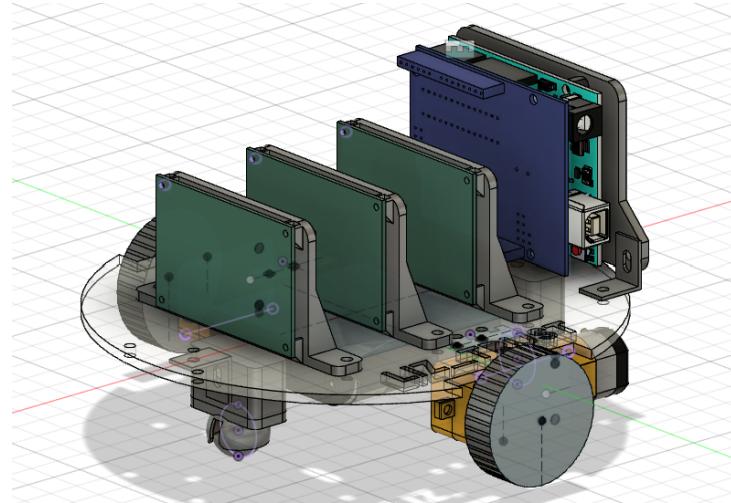


Figure 41: Vertical mounting of metro board and perforated boards.

This design decision was favoured over stacking the boards on top of one another, as it kept the centre of gravity low, increasing the stability of the rover. Furthermore, it ensured that each individual board could be swapped out or removed easily, rather than having to manage boards that were stacked together.

Consideration 3: Rover is manoeuvrable

The large chassis provided by the lab were cumbersome and difficult to steer, where the long body of the chassis made it difficult to estimate where the sensors mounted on the front would end up after turning.

Hence, a circular chassis was chosen, with the wheels mounted central to the chassis, as seen below:

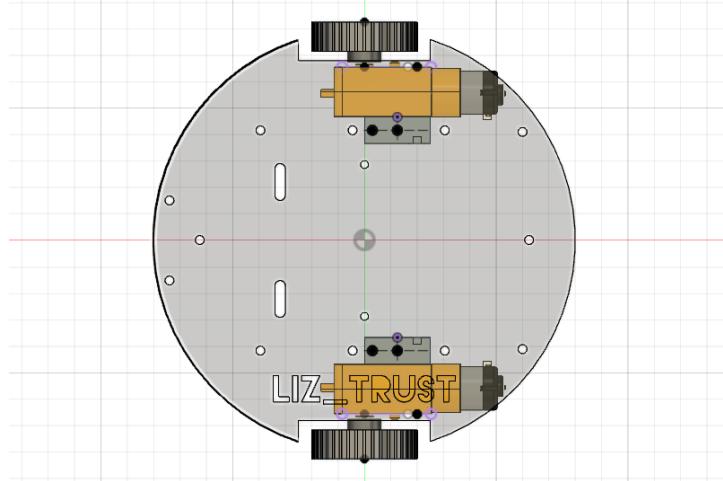


Figure 42: Wheels mounted centrally on the chassis.

This ensured that the sensor positions after turning the rover were highly predictable, as the chassis had equal radius on every side.

Wiring and safety considerations

The 5V power supply from the battery pack is connected to the boards through a centralised screw terminal and fork connectors. This is done similarly for ground on the other side of the rover, where the terminal is then connected to the GND pin on the metro board.

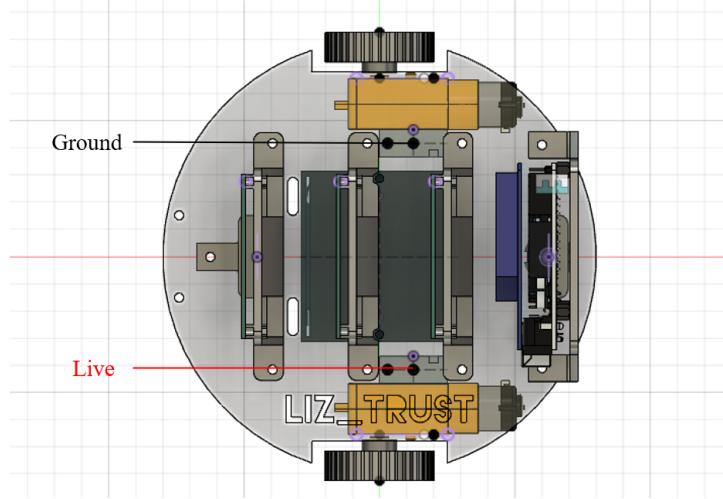


Figure 43: Ground and 5V screw terminal locations.

The biggest safety consideration for the physical integration was ensuring that live wires were properly insulated to prevent electric shocks. Heat shrink tubing and electrical tape were used to cover exposed wires and wire splices.

Furthermore, a plastic shield was 3D printed to prevent users from accidentally coming into contact with the centralised live screw terminal. The shield was printed in bright red to denote the live screw terminal location.

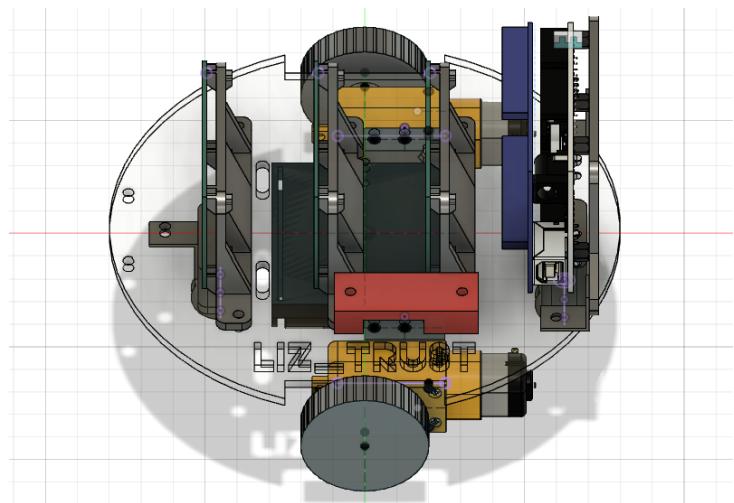


Figure 44: Live shield (in red) to cover live screw terminal.

2.2 Software

2.2.1 Motor Drive

A critical component of the system is the rover's mobility, which is facilitated by the mechanical design. This design incorporates two DC motors, which are controlled by a PWM signal generated by an H-bridge within the motor drive integrated circuit.

To establish a functional system, it is essential to investigate the wireless configuration and design a basic web-based user interface before implementing arrow key controls for rover motion.

This process is split into phases, being initiated by a wireless configuration to turn the in built LED on/off, followed by Rover motor control, then further UI development and finally system integration.

Wireless Connectivity

For wireless functionality, an Adafruit WINC1500 Wi-Fi Shield has been used. This shield connects to the Adafruit Metro M0 micro controller through the 6 pins shown in Figure 45. This system utilises a wired configuration employing the Serial Peripheral Interface (SPI), as evidenced by the MOSI, MISO, and SCK connections. These connections enable the microcontroller, acting as the master, to control data transfer and usage while being synchronized by the clock. Consequently, this configuration ensures that the motors exhibit minimal latency, stopping almost immediately upon receiving a stop command.

The Wi-Fi shield itself, connects to a local network via WPA2, which can be set up via a mobile Hotspot.

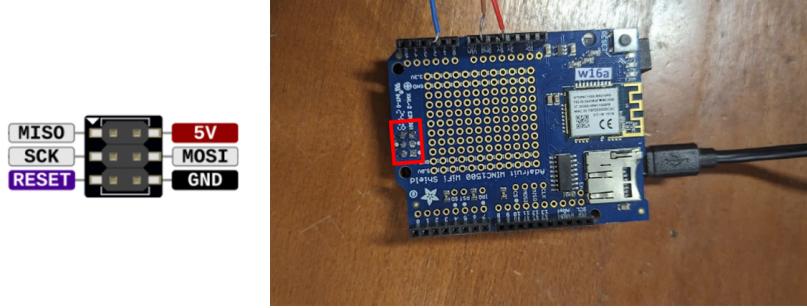


Figure 45: Wi-Fi Shield attachment

LED on/off

By using the code provided [15], including the Wi-FiWebServer library [16], a connection can be established and a control panel can be seen, created in HTML hosted at a static IP as the root URL.



Figure 46: LED on and off control panel

Once the LED is configured as an output, a HTTP server can be initiated, and thus a control panel becomes accessible (Figure 46), allowing the user to toggle the LED on and off. This functionality is achieved through an indefinitely looping server polling function that checks and handles incoming client requests. The HTTP server continuously monitors these requests. From the control panel, clients send HTTP GET requests to retrieve data and POST requests to transmit data, which the microcontroller processes in real-time. This is done through the XMLHttpRequest object [15] with `send()` and `open()` methods and monitored by:

```

xhttp.onreadystatechange = function() {
  if (this.readyState == 4 && this.status == 200) {
    document.getElementById(\"state\").innerHTML = this.responseText;\}
  }

```

Motor control

A similar strategy is used here, but now adding the pin configurations for the motor output signal. Figure 47 elucidates the motor attachment process, outlining the requisite hardware components (motor drive circuit and motors) and the program in the microcontroller to communicate with the Web UI.

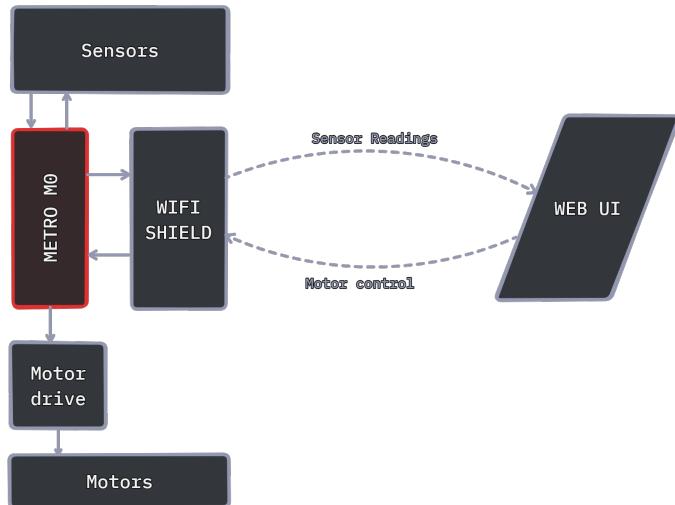


Figure 47: Block diagram for Motor Control

The motor drive chip (Figure 48) illustrates the H-bridge, which is a current driven circuit to power the motors in either direction.

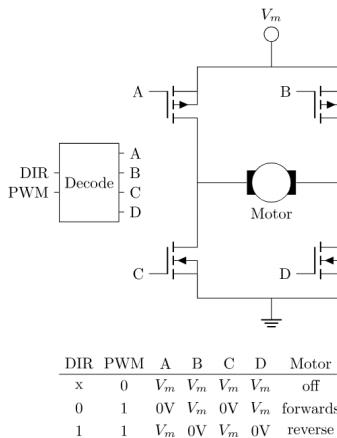


Figure 48: Motor drive circuit

A H-bridge is a circuit that switches the polarity of the voltage applied to a load. In this case, the system contains 2 H-bridges for each motor, with each channel having 2 inputs (DIR and PWM), that control 4 MOSFETs (similar to a switch) and produce an output to the Motors.

DIR controls the direction of the motor:

Motor terminals connected so current flows from V_m (+5V) through MOSFETs B and D to GND for forward motion. (DIR HIGH)

Current passes through A and D for reverse motion. (DIR LOW)

PWM controls the speed of the motor depending on the duty cycle connected to the H-bridge's ENABLE pin. This varies the voltage across the motor so can either be on at any speed (PWM HIGH) or off (PWM LOW).

Through the program, EN is set to HIGH by using `digitalWrite()` and it provides a PWM signal to the DIR pin using `analogWrite()` with a value between 0 and 255.

Where 0 produces a PWM duty cycle at a minimum (0%) resulting in backwards rotation at the highest speed.

Conversely, 255 produces the highest PWM duty cycle (100%) with forwards rotation at the highest speed.

This also means that at 127, the cycle is stationary with a duty cycle at 50%.

When applying this, a simple control panel with buttons is created, which when clicked will initiate a function through the `onClick()` method. So GET requests are handled once again to acquire the user input. This is shown in listings 1 and 2.

Listing 1: Motor control GET requests

```
function forwards() { xhttp.open(\"GET\", \"/fwd\"); xhttp.send();}\\
function backwards() { xhttp.open(\"GET\", \"/bwd\"); xhttp.send();}\\
function stop() { xhttp.open(\"GET\", \"/stop\"); xhttp.send();}\\
function left() { xhttp.open(\"GET\", \"/left\"); xhttp.send();}\\
function right() { xhttp.open(\"GET\", \"/right\"); xhttp.send();}\
```

Listing 2: Register callbacks to respond to requests

```
server.on(F(\"/\"), handleRoot);
server.on(F(\"/fwd\"), forwards);
server.on(F(\"/bwd\"), backwards);
server.on(F(\"/stop\"), stop);
server.on(F(\"/left\"), left);
server.on(F(\"/right\"), right);
```

This is combined with the control panel (Figure 49) to have a working system.

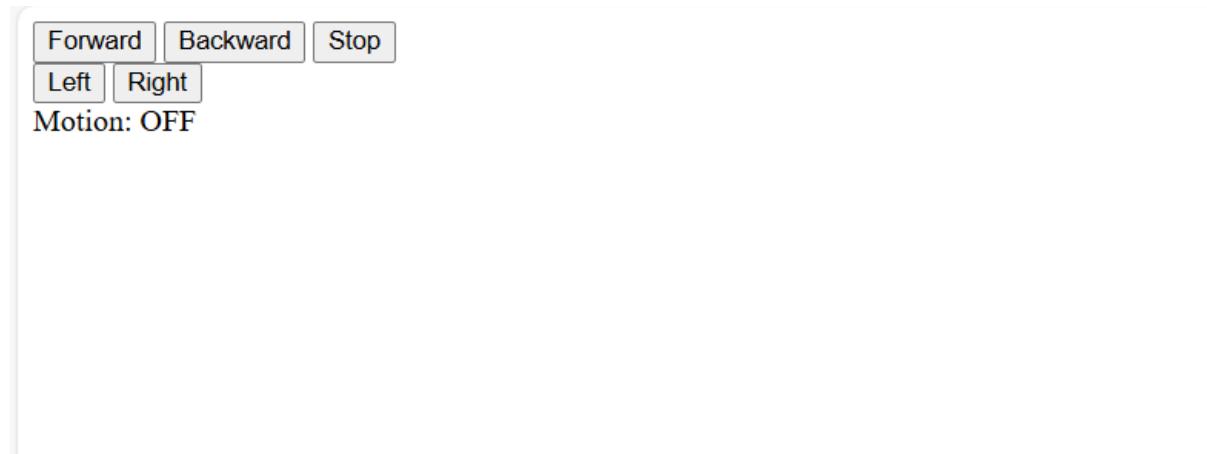


Figure 49: Motor control panel

2.2.2 Web User Interface

With the core control panel functionality established, a more interactive and informative user interface (UI) can be designed. This enhancement is achieved using HTML (providing a structure), CSS (styling and appearance), and JavaScript for handling POST and GET requests, thereby adding interactivity to the control panel.

Leveraging HTML5, CSS3, and JavaScript facilitates the creation of a foundational design, which will be used to develop two prototypes: one utilising arrow keys for rover motion control (Figure 50) and another employing a joystick (Figure 51).

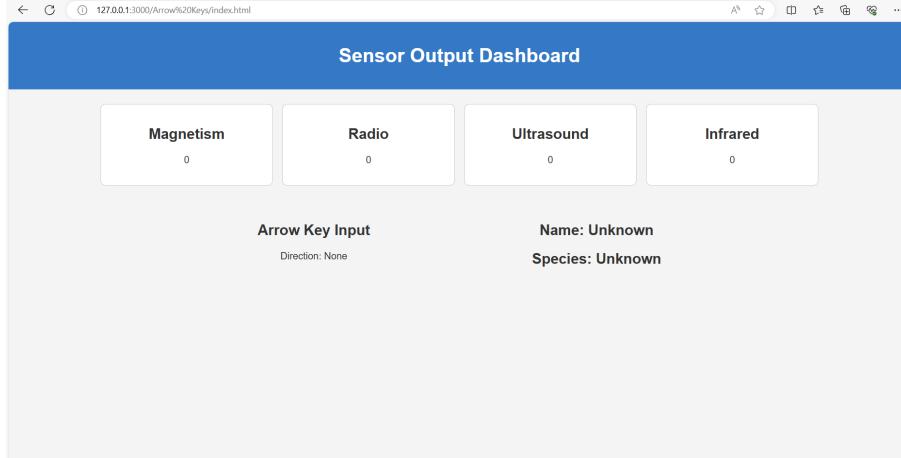


Figure 50: Arrow Keys Web Interface

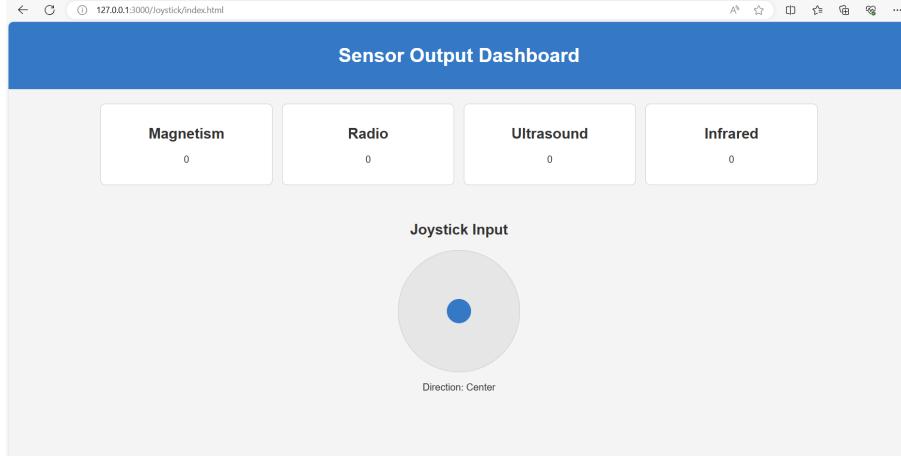


Figure 51: Joystick Web Interface

To implement this, a straightforward locally hosted website will be utilised. Accurate values and configurations will be inserted once all sensors and motors are integrated into the system. This setup will ensure precise and responsive control of the rover, enhancing the overall functionality and user experience of the UI.

The key feature of the JavaScript implementation is the use of:

```
document.addEventListener( "DOMContentLoaded", () )
```

This ensures that the script has an eventhandler to 'listen' for specified events like a key press or mouse movement in this case. This JavaScript code enables the configuration of outputs, including four sensor values, and Lizard parameter values (name and species, determined by the sensor values). Additionally, it processes input from the arrow keys or joystick.

This continuous execution allows the system to regularly update the outputs to reflect the current sensor values. These updates are performed at set time intervals, ensuring that the displayed data remains

accurate and up-to-date. This approach ensures a dynamic and responsive user interface, enhancing the interaction and functionality of the control panel. This can be seen in the JS code with the key structures shown in listing 3:

Listing 3: JavaScript code for the User Interface

```
document.addEventListener("DOMContentLoaded", () => {
  const outputs = {...};
  const parameters = {...};
  const arrowDirection = ...;
  const updateOutputs = () => {};
  const updateParameters = () => {};

  setinterval(updateOutputs, 100);
  setInterval(updateParameters, 100);
  ...
})
```

Any further code can be seen in the complete code in the GitHub repository.

2.3 Integration

This stage requires the combination of the hardware and the software to complete the rover and produce the final system which will be achieved by the physical integration of the circuits onto the mechanical design of the rover and the logical integration of the sensors with the Web UI.

2.3.1 Physical Integration

This section entails the combination of all 4 boards onto a single chassis previously designed as seen in Figure 55.

After testing all circuits on the breadboards, the respective circuits were soldered onto 3 small perforated boards for final production. To save space and ensure that everything fit within the 3 boards, simpler circuits were soldered onto the same board such as the Infrared and Magnetic sensors as seen in Figure 52 and the Ultrasound and motor driver as seen in Figure 54. Larger circuits were soldered individually on a perforated board such as the Radio sensor as seen in Figure 53.

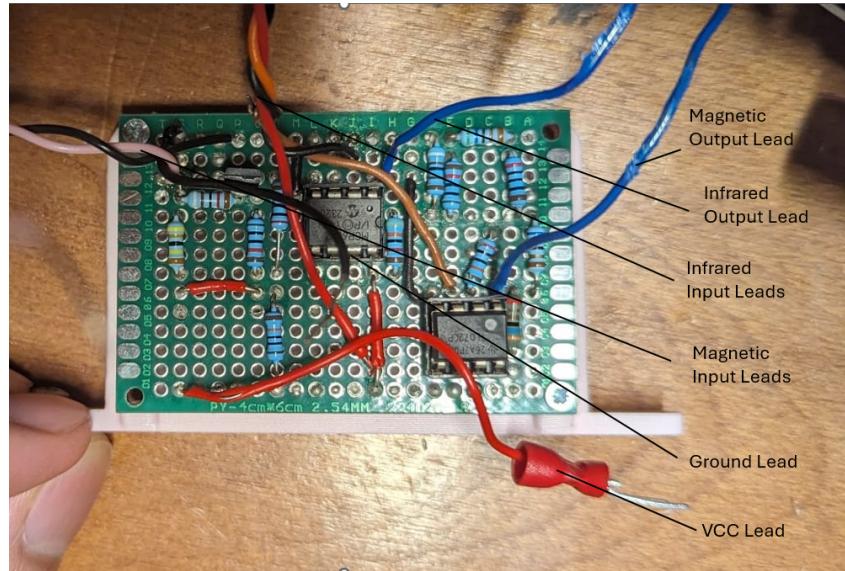


Figure 52: Example Board: Infrared and magnetic sensor circuit board

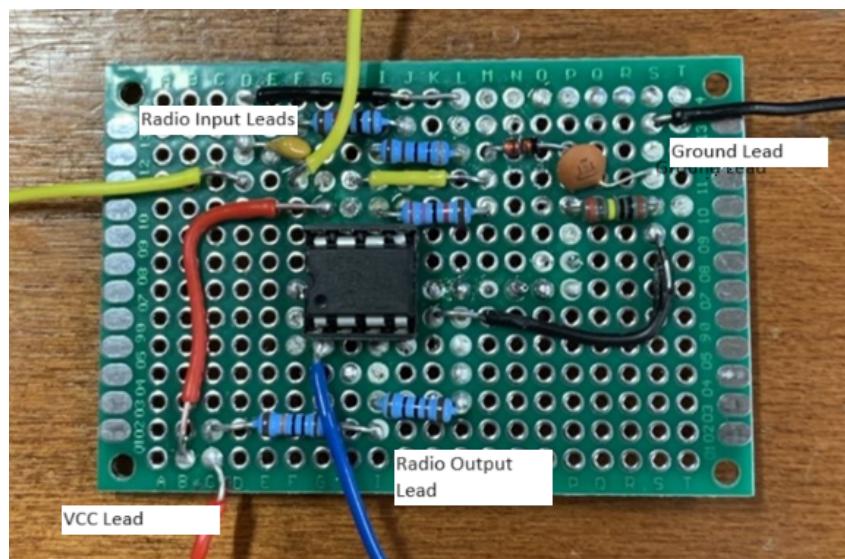


Figure 53: Radio circuit soldered on the perforated board

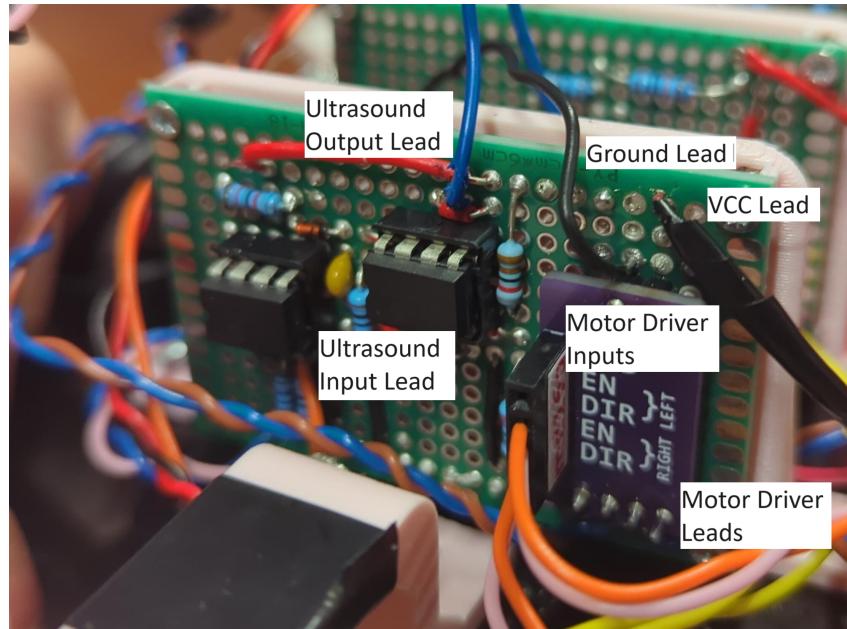


Figure 54: Ultrasound sensor and motor driver on perforated boards

Each board has one VCC and Ground wire leads with a fork connector to be connected to the centralised live and ground screw terminals.

Each circuit also included their various input and output wire leads to be connected to the sensors and metro board.

Finally, the boards are attached to the 3D printed board supports and mounted to the rover.

The rover has the sensors positioned such that it is able to pickup all of the different signals emitted from the lizard effectively. It is able to rotate on the spot due to having the 2 wheels place either side of the rover's centre of gravity. This allows for precise movements to be made making it easier to line up the sensors with lizards.

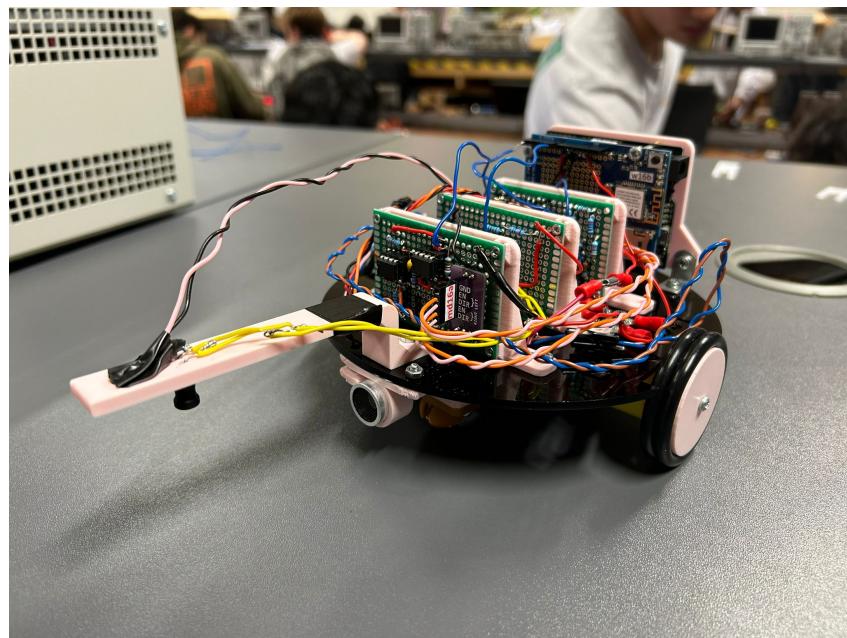


Figure 55: Assembled Rover

2.3.2 Software Integration

With all individual components created, backtracking through the project breakdown (Figure 1 shows that the Hardware and the Software can now be Integrated with all dependant components created. Starting with acquiring the appropriate values from each of the sensors. Then merging all the code into a single file such that when all sensors are physically integrated onto the chassis, with the lizard present, all appropriate values are attained. Then merging the program with the web UI, creating a final product that is read and understood by the user.

The individual programs can be viewed in the GitHub repository.

Ultrasound

The ultrasound sensor is tasked with returning the name of the lizard. The circuit configuration processes UART data packets as input and generates a PWM output. The program subsequently detects if a signal is being transmitted. Upon signal detection, an array of characters stores the first three characters read after the delimiter '#'.

This process involves reading input from the circuit into the microcontroller unit using `Serial1.read()`, which operates at a baud rate of 600 bits per second. The value read on the Rx (receiver) pin ranges between 0 and 255. This range results from the microcontroller handling 10-bit UART frames, which include 1 start bit, 8 data bits, and 1 stop bit, thus producing an 8-bit data width for the actual information content.

The 8-bit value is then mapped to a character using ASCII representation and appended to a string or an array of characters. This approach ensures that the encoded name, transmitted via ultrasound, is accurately captured and processed by the system for subsequent use. The 3 characters obtained are then output to the console.

Radio

The radio sensor is designed to determine the frequency of a signal transmitted by a lizard. The circuit receives an amplitude-modulated (AM) signal, demodulates it, and outputs the original modulating signal. The program measures the duration of the high and low pulses of this signal and sums them to calculate the signal's period. By inverting this period, the program can determine the frequency of the signal.

This process begins with the circuit sending an analogue input to the microcontroller via pin A1. This then measures the duration of each high and low pulse using two functions with a timeout of 10 000 microseconds. If a high or low pulse is not detected within this time frame, the corresponding function returns a value of 0. The two measured durations are summed to obtain the signal's period. If this period is non-zero, the program calculates the frequency by dividing 1 000 000 microseconds by the period.

Finally, the determined frequency is output to the console.

Infrared

This sensor returns either a high or low signal based on the presence of an infrared signal. By measuring the duration of the high and low states, the period can be determined and subsequently the frequency of the infrared signal. This method of detecting and processing the signal is identical to the approach used with the radio sensor.

Magnetic

The sensor operates by taking an initial reading through a Hall sensor, and this is then processed. When taking an input into the Metro board, a 10-bit value is acquired. In decimal form, this value ranges from 0 to 1023. When there is no magnetic field detected, the sensor is expected to have a steady-state value approximately equal to half of the supply voltage $V_{CC}/2$ which is equivalent to 512. However, this value is theoretical and requires adjustment to consider unexpected environmental factors.

The amplified value is fed into the microcontroller through an analogue input pin (A0). The microcontroller reads this value to determine the presence and strength of a magnetic field. The program running on the microcontroller takes 10 readings from the sensor when no magnetic field is present. These readings are averaged to establish a baseline or reference value, representing the sensor output when no magnetic field is detected.

When a magnetic field is present, the sensor's output changes, either increasing or decreasing from the baseline value. The program continuously takes new readings and computes a moving average of the

last 10 values to smooth out any fluctuations and provide a stable reading. The difference between the current moving average and the initial reference value indicates the presence and polarity of a magnetic field. If the difference is positive, the North pole of the magnet is detected. If the difference is negative, the South pole is detected.

Complete Integrated System

For the complete system integration, all sensor code needs to be compiled into a single file. This then needs to be paired with the JavaScript so that the sensor values obtained on the server side can be displayed on the web UI on the client side. In order to do this a python server has been created using flask. Finally, the name and species of the lizard can be displayed appropriately on the UI. This is done using Platform IO due to ease of use (specifically to make use of the HTML, CSS and JS code) as opposed to the Arduino platform.

Merging all sensor code into a single file

The individual sensor programs can then be compiled into one single file for sensor detection merged with the motor control. This file can be used to return the relevant values that can then be displayed on the dashboard once it is configured with the Web UI.

In this file, pins have been allocated for each of the sensors and the motor EN (PWM) and DIR. After this, the individual programs have been merged together.

Python Flask Server

After reconfiguring the Python Flask server to run on a mobile hotspot with the IP 192.168.221.17, the Arduino will attempt to connect to the same mobile hotspot with a hardcoded SSID and password which are found in the preloaded program.

Once the Arduino connects to the Python Flask server, it will send a HTTP POST request to the /data endpoint of the server containing a JSON object. The attributes of the JSON object are the radio frequency, infrared frequency, magnetic value and parsed ultrasound name. The Python Flask server will deserialise the JSON object and update the values in a global variable accessible to all endpoints.

The webpage, which can be accessed through the /dashboard endpoint, will continuously update the sensor values. This is done using a AJAX GET request to the same /data endpoint, which would output the last known sensor values as a JSON object. The JavaScript code on the webpage would then parse and render the values on the webpage.

As for the direction of the robot's movement, the webpage will continuously listen to keystrokes made by the user. If the user presses a direction key, the webpage will send a POST request to the /direction endpoint. Similarly, the Arduino would continuously poll for the direction by sending GET requests to the same /direction endpoint. The entire process is illustrated in greater detail in the figure below.

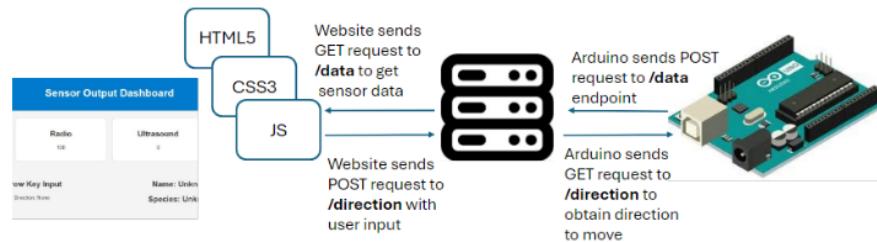


Figure 56: Assembled Rover

JavaScript File

The main body of this file is similar to the one developed in the section for the Web User Interface, with certain changes made to reflect the values attained by the sensors in the C++ script and the python script, thus rendering these values as part of the Web UI. The sensor values are received in this script from the AJAX GET request, which is done every 0.1 seconds (as per the interval that has been defined). This is set to 0.1 seconds to manage the server load and prevent excessive delays in handling client requests. This is specifically used for the 4 sensors with success callback functions to process the response data and update variable values as shown in Listing 4.

Listing 4: AJAX GET request for fetching latest sensor data

```
radio = response["radio_frequency"];
infrared = response["infrared_frequency"];
magnetic = response["magnetic_difference"];
ultrasound = response["ultrasound_name"];
```

This is then rendered to the user interface via the `updateOutputs()` function.

The motor direction is received from the user interface and sent to the server side by an AJAX POST request. This is done when the user presses one of the 4 arrow keys on their keyboard. This is sent to the `/direction` URL where it can be received by the flask server in the form of JSON data. As shown in Listing 5, the headers indicate that the request and response bodies contain JSON data and the data itself is converted to a string using `.stringify()` as part of this POST request.

Listing 5: AJAX POST request for sending motor direction as a JSON string

```
$.ajax({
  url: "/direction",
  type: "POST",
  headers: {
    'Accept': 'application/json',
    'Content-Type': 'application/json'
  },
  data: JSON.stringify(arrowDirection.textContent)
});
```

The complete code is shown in the GitHub repository.

Identifying name and species

As part of the JavaScript code, the species can be determined through a set of if statements with the correct value (± 5) being displayed as per the criteria shown in Figure 1. The name can be simply output to the web UI as this is value acquired by the ultrasound sensor.

Finished Software Integration

After the aforementioned parts being put together, the result can be seen in Figure 57. This illustrates the Name appearing as well as the Magnetic sensor value of North and the Infrared sensor value of 570 Hz when the lizard has a mode of only switch 2 at HIGH.

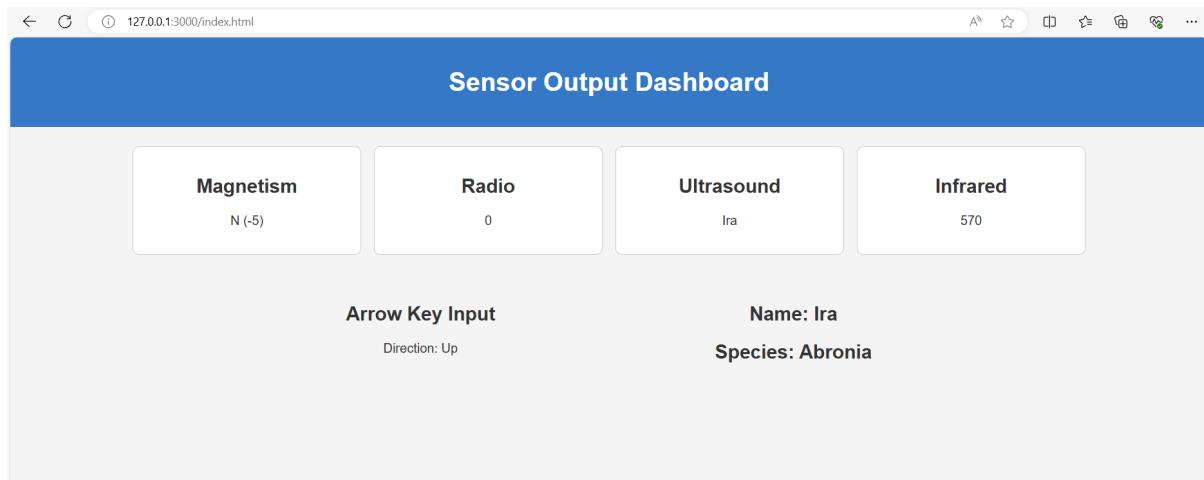


Figure 57: Complete and working User Interface

3 Conclusion

In conclusion, the rover has demonstrated the ability to successfully identify the name and species of lizards, with this information being readable from the web user interface. The project adhered to a budget of under £60 and ensured the rover weighed less than 750g. The rover can be easily driven and maneuvered using the web UI, indicating that all acceptance criteria outlined in Table 2 have been met.

The project successfully integrated various sensors, including Hall effect sensors, ultrasound sensors, infrared sensors, and radio sensors, into a single, cohesive system. The Hall effect sensor accurately determined the magnetic polarity, while the ultrasound sensor decoded UART data packets to reveal the lizard's name. Both the infrared sensor and the radio sensor measured their respective signals' frequency. These capabilities were seamlessly managed by the microcontroller, which processed the data and provided real-time outputs to the user interface.

Technical Achievements

The success of the project is attributed to the effective integration of hardware and software components. The use of a microcontroller to handle multiple sensor inputs and the development of a web-based user interface for remote control and monitoring sensor outputs were key technical achievements. Further, to handle POST and GET requests using AJAX and a Python Flask server is also a key accomplishment.

Furthermore, the system's design focused on minimising weight and cost without compromising functionality. Using lightweight materials such as PLA and PMMA (Acrylic) and an effective rover design, while being careful with components selection and budget management kept the project within financial constraints.

In addition, with all members of the team working in their respective roles to the best of their ability, collectively an asset to the project.

Next Steps and Future Improvements

Despite the success, there are several areas for improvement. The rover could be improved by increasing its speed. This could be achieved by using a combination of faster motors and larger wheels, improving maneuverability but also allow the rover to cover larger areas of land quicker.

A PCB could also be designed to make the electronics to enhance reliability and compactness. Integrating the PCB as part of the chassis would further reduce weight and streamline the design. Thus, simplifying assembly and maintenance.

Another potential enhancement involves attaching the probe for the rover to a servo so that the probe can move along the horizontal plane in front of the rover, allowing for sensors to be positioned accurately so that they perfectly align with the back of the rover, improving detection.

In addition, incorporating the ESP32 into the rover system offers significant improvements over the current Metro M0 microcontroller. This is due to the ESP32 having a dual-core processor, with a clock speed of over 240MHz, significantly enhancing the processing power and speed, with reduced latency in sensor data processing while determining the species of the lizards [17]. It also has expanded GPIO capabilities for additional sensors than would be allowed by the Metro M0 that the rover currently uses, for any further future development where additional sensors may be required. This has built-in Wi-Fi for robust and reliable wireless communication improving data transfer as well as being power efficient which extends the rover battery life.

References

- [1] Tutorials Point, “Amplitude Shift Keying,” https://www.tutorialspoint.com/digital_communication/digital_communication_amplitude_shift_keying.htm, accessed: 2024-06-06.
- [2] S. P. Grogan and C. A. Mount, “Ultrasound physics and instrumentation,” <https://www.ncbi.nlm.nih.gov/books/NBK570593/#:~:text=Ultrasound%20transducers%20contain%20piezoelectric%20crystals,between%200.2mm%20and%202mm.>, accessed: 2024-06-12.
- [3] OpenStax College, “Faraday’s Law of Induction: Lenz’s Law,” <https://courses.lumenlearning.com/suny-physics/chapter/23-2-faradays-law-of-induction-lenzs-law/>, accessed: 2024-06-11.
- [4] Murata, “Basics - Types and comparisons of infrared sensors,” <https://www.murata.com/en-eu/products/sensor/infrared/overview/basic/kind>, accessed: 2024-05-28.
- [5] Infratec, “Pyroelectric Detector,” <https://www.infratec.co.uk/sensor-division/service-support/glossary/pyroelectric-detector/>, accessed: 2024-05-28.
- [6] Electrical Technology, “Difference Between Photodiode and Phototransistor,” <https://www.electricaltechnology.org/2022/01/difference-between-photodiode-phototransistor.html>, accessed: 2024-05-29.
- [7] Circuit Globe, “Difference Between Photodiode and Phototransistor,” <https://circuitglobe.com/difference-between-photodiode-and-phototransistor.html>, accessed: 2024-05-29.
- [8] Team Wavelength, “Photodiode Basics,” <https://www.teamwavelength.com/photodiode-basics/#:~:text=The%20PIN%20photodiode%20is%20similar,between%20the%20two%20doped%20layers.>, accessed: 2024-05-29.
- [9] B. Doherty and M. Watertown, “PIN Diode Fundamentals,” <https://www.microsemi.com/sites/default/files/micnotes/701.pdf>, accessed: 2024-05-29.
- [10] P. Gupta, “IR-Photo Diode Sensor,” <https://forelectronics.wordpress.com/2020/05/28/ir-photo-diode-sensor>, accessed: 2024-06-10.
- [11] K. Fobelets, “EE1 An introduction to semiconductor devices.” 2023.
- [12] RS Components, “How Do Photodiodes Work?” <https://uk.rs-online.com/web/content/discovery/ideas-and-advice/how-do-photodiodes-work>, accessed: 2024-06-10.
- [13] Asahi Kasei Microdevices, “What’s a Magnetic Sensor?” <https://www.akm.com/us/en/products/hall-sensor/tutorial/magnetic-sensor/#:~:text=A%20magnetic%20sensor%20is%20a,sensor%20types%20and%20their%20features>, accessed: 2024-05-23.
- [14] Diodes Inc., “AH49F Datasheet,” <https://4donline.ihs.com/images/VipMasterIC/IC/DIOD/DIOD-S-A0007976879/DIOD-S-A0007976879-1.pdf?hkey=6D3A4C79FDBF58556ACFDE234799DDF0>, accessed: 2024-05-24.
- [15] H. Merdan, “EEERover Specifications and Technical Guide,” <https://github.com/hakanmerdan/EEERover2024>, accessed: 2024-06-10.
- [16] Khoih-prog, “Wifiwebserver,” <https://github.com/khoih-prog/WiFiWebServer>, accessed: 2024-05-22.
- [17] E. Systems, “ESP32 Series Datasheet,” https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf, 2024.

A Risk Assessment

For the project to be safely completed, a risk assessment is essential. Various hazards have been identified that may prove harmful to team members and anyone present in the laboratory while the project is carried out, including laboratory assistants and teaching assistants. The hazards mentioned do not pose the same threat to the users of the laboratory and the majority of them are unlikely to occur due to extensive measures taken by the laboratory assistants. The assessment is performed in Table 11 and is done based on the Likelihood (Table 7) and Severity (Table 8). The combined risk rating is done based on these as shown in the Risk Assessment Matrix 9. The Risk rating and Action table (Table 10) elucidates the reasoning behind the colour scheme implemented.

Likelihood	Guide Description
5	Very likely/imminent - certain to happen
4	Probable – a strong possibility of it happening
3	Possible – it may have happened before
2	Unlikely – could happen but unusual
1	Rare - highly unlikely to occur

Table 7: Likelihood Guide Description

Severity	Guide Description
5	Catastrophic – fatality, catastrophic damage
4	Major – significant injury or property damage, hospitalisation
3	Moderate - injury requiring further treatment, lost time
2	Minor – first aid injury, no lost time
1	Very minor – insignificant injury

Table 8: Severity Guide Description

Likelihood (L)	Severity (S)				
	1	2	3	4	5
5	5	10	15	20	25
4	4	8	12	16	20
3	3	6	9	12	15
2	2	4	6	8	10
1	1	2	3	4	5

Table 9: Risk Assessment Matrix

Risk Rating (RR)	Action
High Risk	Stop the task/activity until controls can be put into place to reduce the risk to an acceptable level
Medium Risk	Determine if further safety precautions are required to reduce risk to as low as is reasonably practicable
Low Risk	No further action, keep under review

Table 10: Risk Rating and Action

Hazards and potential risk	Control Measures	L	S	RR
Soldering iron misuse: Burns to hands/fingers/other body parts and Inhaling the smoke of soldering iron	Use tweezers/pliers or a vice to hold work piece where possible. Always assume that the soldering iron is hot and place it back in its holder when not being used. Switch the iron off when it is not in use and place it in the holder. In case of minor burns use the first aid box available locally for treatment. Wear protective glasses when soldering. Avoid using close to the face.	3	4	12
Misuse of hand tools e.g. cutters pliers blades screwdrivers: Cuts on hands/fingers/other body parts	When trimming component legs point towards the floor or into a waste container/bin. Visually inspect hand tools prior to use. If tools are damaged do not use them and contact technical staff for replacement. Ensure that tools are stored safely and appropriately when not in use. Seek advice from technical staff if necessary. A lab coat may be worn if required to protect skin and clothing.	3	4	12
Electric Shock: Muscle spasms burns to skin	Do not bring any drink to the lab. Ensure that the power is off before handling circuits. Use insulated tools. Use low voltage power supplies. Ensure that no cable is left open-ended.	2	2	4
Battery leakage: Destructive corrosion inhalation irritation contamination	Keep batteries away from humidity or potential water exposure. Periodically inspect batteries for any signs of leakage or corrosion. Store batteries separately from metallic objects to avoid short circuits. Don't recharge the non-rechargeable batteries.	3	3	9
Fire: Faulty equipment component failure sparks skin burns	Always use a damp sponge for wiping the soldering irons tips. Automatic fire detection fitted in all university buildings. Fire extinguishers are provided in all buildings. Check the electronic components regularly to prevent possible overheating issues.	1	4	4

Table 11: Risk Assessment Table

L = likelihood, **S** = severity, **RR** = risk rating

B Schematics

