University of Southern Queensland

Faculty of Health, Engineering and Sciences

Automated Mammal Nest Box Protection

A dissertation submitted by

Scott Essery

In fulfilment of the requirements of

ENP4111 Professional Engineering Research Project

Towards the degree of

Bachelor of Engineering (Honours) (Mechatronic)

Submitted November, 2024

Abstract

The continued reduction of native forests and habitats, coupled with a booming population of exotic predators and competing feral animals, is putting significant stress on small native Australian animals that require a nest to reproduce and thrive. The purpose of this study is to investigate the feasibility of leveraging new technologies to provide support to a target species of animal in their natural environment.

By reviewing past research projects in wildlife conservation which utilised electronic and automated systems and tools, this study collates common benefits and challenges indicated by the researchers in order to identify key criteria that a successful system would be required to meet. With key criteria identified, design and construction of a system designed to support nesting mammals was performed.

Tests of the various electronic components indicated that a nest box could be feasibly protected by an automated system that leveraged new advancements in compact computing, object and image recognition, and wireless reporting systems.

This study only analysed the technical capability to design and construct a system that could autonomously perform object detection, animal recognition, and operate the locking mechanism. Further study into how a live untrained animal would interact with a system is required to analyse whether the animal would both approach the nestbox, and whether they would positively interact with the nest box door.

University of Southern Queensland

Faculty of Health, Engineering and Sciences

ENG4111/ENG4112 Research Project

Limitations of Use

Certification

I certify that the ideas, designs and experimental work, results, analyses and conclusions set out in this dissertation are entirely my own effort, except where otherwise indicated and acknowledged.

I further certify that the work is original and has not been previously submitted for assessment in any other course or institution, except where specifically stated.

Scott Essery

Student Number: [redacted]

Acknowledgements

I would like to give thanks and acknowledge the contributions of those who have helped and supported my efforts in the project.

I would first like thank my thesis advisor Dr. Derek Long for his project proposal and for supporting my efforts throughout this project. His guidance, critique and counselling throughout the year has been invaluable, helping me improve with both my technical research and providing much needed guidance on how to produce a professionally structured report.

I would also like to thank Dr. Meg Edwards for making herself available to discuss the project and taking providing her valuable time to review and discuss my work. Her kindness and willingness to help, even with content outside of her area of expertise, was greatly appreciated.

Finally, I would give a huge thank you to my wife, Heather, who put up with my long nights at the desk, the weekends alone with our son, and who provided me with never ending support to strive on to complete my studies.

Contents

Table of Figures

List of Tables

Abbreviation, Acronyms and Notations

API .... Application Programming Interface

AWS .... Amazon Web Services

CPU .... Central Processing Unit

CSI .... Camera Serial Interface

DoD .... Depth of Discharge

GPIO .... General-Purpose Input/Output

IC .... Integrated Circuit

LEUART .... Low Energy Asynchronous Receiver/Transmitter

MCU .... Microcontroller Unit

PCB .... Printed Circuit Board

PIR .... Passive Infra-red

RFID .... Radio Frequency Identification

SELV .... Safety Extra-Low Voltage

SBC .... Single Board Computers

SMART .... Spatial Monitoring and Reporting Tools

V (AC) .... Volts (Alternating Current)

V (DC) .... Volts (Direct Current)

VHF .... Very High Frequency

1. Introduction

1.1 Background

Native mammals are under increasing pressure from human redevelopment of their natural living and breeding grounds(Woinarski et al., 2015). As bushlands and forests are removed, so too are suitable nesting habitats that are crucial to the reproductive process of these animals. Exotic predators and smaller exotic nesting animals are adding yet another layer of pressure on target native wildlife populations (Aretino et al., 2001). These predators may target native nesting species as an easy source of food. Other exotic species of animal may inhabit the already rare nesting areas required by the native species to survive and breed (Rogers, 2019). With less available nesting sites available, and increasing predation pressure, additional conservation efforts are required to help the dwindling native populations of nesting species of animals survive and grow (Saunders et al., 2022).

Advances in electronics, robotics, and computer vision technology are being used to help wildlife conservation in a variety of modes, allowing for new and effective means of monitoring, intervention, and protection (Verdict, 2022). Drones and Spatial Monitoring and Reporting Tools (SMART) have been deployed to track illegal poaching in areas that would normally be nearly impossible to monitor (Heft, 2020). Microchips for access automation have been leveraged in combination with animal training to create safe havens for bandicoots (Edwards et al. 2020). Finally, with the recent leaps in AI and computer vision processing tools, animal detection and classification can be automated and carried out at the same instant an animal is detected and captured on a digital camera. Services such as Wildlife Insights can provide an online platform for image submission and processing, as well as a database of images containing the sought-after target species, providing a valuable dataset for computer learning algorithms to refine their species identification capacity (Wildlife Insights, 2024). Beyer and Goldingay (2006) performed a review of published works regarding nesting boxes, and identified species detection, species ecology and the preference in box design in different species as prominent applications for research. This research project proposes to provide a system of access automation to nesting boxes and leverage computer vision systems, create monitoring and reporting tools, and to determine the effectiveness of the access systems.

1.2 Aims and Objectives

This report aims to investigate whether newer technologies in compact computers, sensors, cameras and image recognition software can be utilised to provide a benefit to a target animal species in the form of access control to a nesting box. The supporting team at the School of Agriculture and Environmental Science have requested that the project be targeted to generate a positive response when detecting a Sugar Glider (*Petaurus breviceps*). This requirement will facilitate a smooth transition into further projects and investigations by the team.

1.2.1 tasks:

- Design and construct a gating system for a nesting box. The system is to be triggered by an animal identification program.
- Develop a monitoring system to collect data on various events and parameters, e.g. nesting box usage, predator attacks, utilising both internal memory storage as well as leveraging IoT technologies for remote monitoring.
- Simulate the nesting box system in an enclosure for a target species (sugar glider) to test the box and control system.
- Perform system tests on the box by simulating a non-target species animal attempting to enter the enclosure.
- Ensure that all designs are safe to interact with both the target and non-target animals. The nesting boxes must not be able to trap an animal inside the box in the event of a hardware or software failure. Designs shall be suitable to be installed in harsh outdoor conditions and must reasonably resist encroachment or attack by animals.

1.2.2 Expected Outcomes:

Key criteria

- Demonstrate that there is no risk of harm to any animal interacting with any component of the system.
- Demonstrate that an autonomous nesting system can either allow, inhibit or deter the entry of various species of animals, depending on system requirements.
- Demonstrate that the nest box can function autonomously, requiring limited or no support by staff to maintain functionality.
- Demonstrate that nesting data can be collected and reviewed.

1.3 Limitations and specifications

1.3.1 Technical Limitations

The selection of the camera system and image recognition software is not included in the scope of this project and has been managed by a separate team. For this project, the imaging team has specified that the software "Ultralytics YOYOv8" shall be implemented to perform the image recognition function. Through consultation with the School of Agriculture and Environmental Science team, the imaging team have identified specific camera technologies that shall be implemented.

Hardware and software requirements for both the imaging software and the camera systems shall be investigated in this report, to ensure that the system design satisfies their specifications.

1.3.2 Ethics

This project is being conducted for UniSQ, who are based in Queensland Australia. As such, this project is required to be in compliance with the rules, regulations, standards and approvals of the Animal Ethics Committee of Queensland.

1.3.3 Project Limitations

The original intention was for the nest box to be deployed in a controlled environment where interactions with live animals would occur. Unfortunately, this was not able to be fully realized over the course of the project. A broader range of tests to simulate animal interactions are to be performed to mitigate the impact of the loss of live test data. Simulations of animal interaction are a poor substitute for the interaction of a live wild animal. As such, the outcome of this project may need to be reviewed against the results of animal trials if or when animal interactions are able to be performed.

1.4 Approach

1.4.1 Literature review

This report will first review the available literature regarding past projects that involve the use of technology and animal interactions, in order to understand the successes, challenges encountered and future opportunities. A review of emerging technologies will be performed to identify potential opportunities to leverage those new technologies to increase the performance of the proposed system.

1.4.2 Concept Design

Using the project specifications and lessons learnt from the literature review, the projects key criteria shall be developed, as fundamental decision drivers for development through the Detailed design stage. Once these criteria have been established, they shall be utilised, along with the input from the image recognition team to drive the concept design stage. In this stage, hardware and software components shall be selected to meet the functionality requirements of the project and must comply with all key criteria.

1.4.3 Subsystem and component design and selection

Once the concept design has been completed, system components selected in the concept design stage shall designed, with concepts being created in a temporary manner (solderless breadboards, small individual scripts etc) to test and confirm functionality and suitability. Each design shall ensure that all design parameters meet the key criteria outlined in section 1.3.2.

When the sub-system has been evaluated to meet design requirements, the design shall be finalised, with any circuits being created on small solder breadboards, hardware being installed or sourced, and scripts being saved in the final master control script.

1.4.4 Assembly

With all components of the nest box system selected or identified, all components of the project shall be created and assembled, and all necessary scripts written. Once assembly is complete, each individual system shall be subject to performance tests to ensure proper functionality.

1.4.5 Test and Commissioning:

While all sub-systems shall be tested independently, a series of wholistic tests shall be performed to ensure that there are no unforeseen conflicts created by connecting the entire mechanical and electronic environment, and that the system as a whole behaves in such a way as to meet both the project specifications and all key requirements identified in the concept design stage.

1.4.6 Results, conclusions, and proposed future works:

A final chapter shall be created to document the results, discuss the project, including the successes, challenges and lessons learnt, and to proposed future works that can either improve on the project, or guide future projects in a similar space.

Report Structure

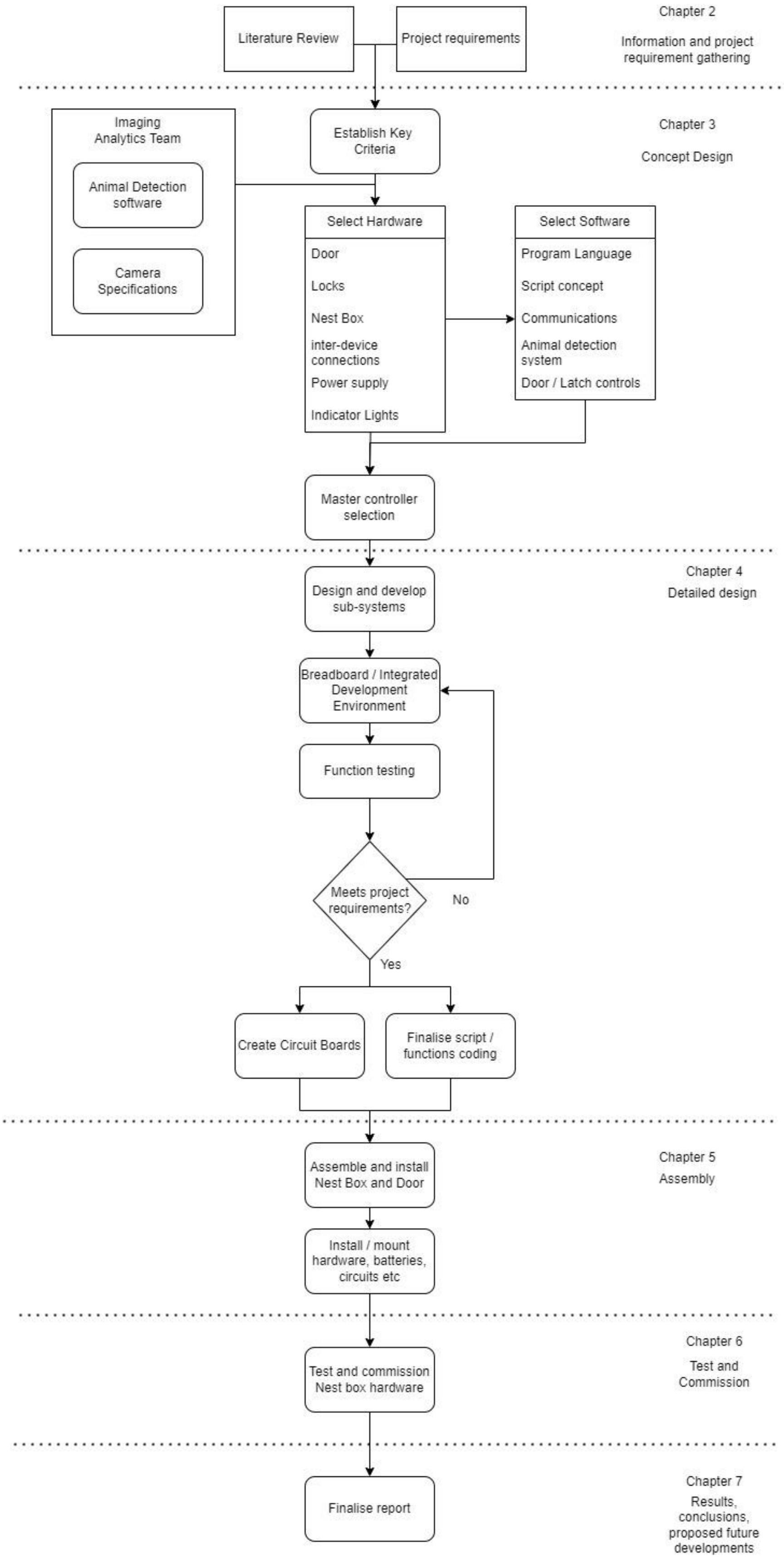


Figure 1 Overall report structure

2 Literature Review

2.1 Introduction

This literature review aims to establish a baseline understanding of the environment, habitat, nesting requirements and predators (both natural and exotic) relating to Australia's nesting mammals and marsupials, with particular notes on the sugar glider (*Petaurus breviceps*).

This review will analyse relevant completed projects into Australian native wildlife conservation and resilience, pertaining both to the target species listed above, and to projects involving non-target species but where the intent, techniques or potential outcomes align with the project goal. These projects will be inspected to identify lessons learned, the challenges, limitations and any potential negatives effects of implementing those conservation works. Emerging technologies shall be investigated to identify potential avenues for further investigation, with speculation on their suitability for a proposed conservation project. Finally, the review will review the viability of the proposed project, and if there is sufficient support to proceed with the project, establish a list of key criteria the project must adhere to.

2.2 Australia's changing natural environment

Australia is facing unprecedented levels of human driven change, both in the increase in global temperatures, and reshaping or removal of much of the natural environments that Australian native animals call home (Burnham, 2021, Dickman and Lindenmayer, 2021, Dales, 2011, Melbourne-Thomas et al., 2024). Australia's natural forests, which consist of Eucalypt forests (75%), Acacia forests (8%), Melaleuca forest (5%) and Rainforest (2%), cover approximately 17% of Australia's total landmass (ABARES, 2019).

A 2022 review of our forests showed that 95.5 kha had been destroyed or removed, equating to a 25% reduction in total forest area (Global Forest Watch, 2023). This reported percentage loss can be supported by reviewing the Australian Governments 2004 and 2019, shown in figure 1 and Figure 2, which reports on the natural forest landmass's (ABARES, 2004, ABARES, 2019), which equates to a reduction of natural forest area across that time period of 19.9%. This rapid loss of forests is driven by agriculture, forestry, mining and urban development (Wilderness Society, 2024). More broadly, Australia's all native vegetation wilderness areas have been reduced by 33% (National Land and Water Resources Audit, 2001), primarily in areas surrounding in the coastal regions of Australia.

Australia's forests in summary	
National Statistics	
Total land area	768 million ha
Total population	20 million
Forest as % of land area	21%
Total forest area	164.3 million ha
Forest per capita	8.2 ha
Native forest area	162.7 million ha
Plantation forest area	1.6 million ha
New area of plantation planted (average 1998–2002)	87 000 ha/year
Forest area in nature conservation reserves	21.5 million ha
Forest area covered by Regional Forest Agreements	23.2 million ha

Figure 2 Forest review 2004 (ABARES, 2004)

Australia's forests in summary	
Total land area	**768.9 million hectares**
Total forest area	**134.0 million hectares**
Forest area as a proportion of land area	17%
Native forest area	131.6 million hectares
Forest area on nature conservation reserve tenure	21.8 million hectares
Native forest area managed for the protection of biodiversity	46.0 million hectares
Native forests available and suitable for commercial wood production	28.1 million hectares
Public native forests available and suitable for commercial wood production	6.3 million hectares
Leasehold and private native forests available and suitable for commercial wood production	21.8 million hectares
Area of commercial plantation forest	1.94 million hectares

Figure 3 Forest Review 2019 (ABARES, 2019)

While forests only cover a relatively small percentage of Australia's landmass, many of Australia's native animal species rely on those forests, with over 30% of Australian mammal species and 40% of native birds living in the Rainforest biomes alone (Forests Australia, 2024).

The loss of habitat has been flagged as the primary driving factor in the loss of plant and animal species around the world (Sala et al., 2000). In Australia, 61% of all animals currently listed as threatened are listed as being primarily affected by habitat loss (Ward et al., 2019).

While there are many rehabilitation programs and groups working to restore Australia's natural ecosystem, performing various restoration project all over Australia (AABR, 2024), the species that rely on mature forests are still vulnerable. While trees that are replanted in order to rehabilitate an area may begin to restore the habitat, the hollows that are required by nesting animals can take up to 100 years to form in trees naturally (Carritt, 1999).

Although loss of habitat is major contributor to the decline of native species, predation by exotic animals is now the leading cause of population decline and extinction of Australian native species, with 17 of 25 extinctions mainly due to invasive species (Low & Booth 2023). The World Wildlife fund (2024) concurs, flagging feral cats and the red fox as the two primary species responsible for the loss of many mammal and bird species, and whose ongoing presence in the environment present a constant and severe threat. Of the animals threatened by exotic predators, mammals, especially rodent and marsupial species have a relatively higher rate of loss, compared to larger mammals (Woinarski et al., 2015).

Predation is not the only mode in which exotic species negatively affect the native animal populations in Australia. Exotic animals often out-compete the natives for food and nesting resources, further depriving those animals of the critical components needed to thrive.

Pell and Tidemann (1997) conducted an investigation of the potential effects of exotic birds on the Canberra's native nesting parrots – Crimson Rosella (*Platycercus elegans), Eastern Rosella (Platycercus eximius)*, and Red-Rumped Parrot (*Psephotus haematonotus)*. The researchers installed approx. 60 nesting boxes between two sites and performed an extensive survey of the wooded area to detect hollows that could be utilised for nests. The study identified that two exotic birds*, the common Myna (Acridotheres tristis)* and the common Starling (*Sturnus vulgaris)* posed a significant threat to the nesting success of the native Parrots. These two exotic bird species utilised the same hollows to form nests and would often successfully outcompete the native parrots for the available nests on account of their increased aggressive nature.

Interactions between bird species around nesting boxes were observed and recorded. From the tabulated results below, the Myna bird prevailed in the majority of their aggressive interactions, with the native red-Rumped Parrot conceding the most.

	Winner					
Loser	Starling	Myna	C rosella	E rosella	RR parrot	Total lost
Starling	—	18	5	5	1	29
Myna	0	—	10	4	0	14
C rosella	4	18	—	1	0	23
E rosella	11	28	34	—	1	74
RR parrot	1	0	0	5	—	6
Total won	16	64	49	15	2	146

Figure 4 Aggressive interactions between bird species.

The inspections also noted that the exotic species would nest earlier in the year than the native birds, occupying nesting sites before the native parrots even began to enter into their mating and nesting periods.

2.3 Conservation Efforts

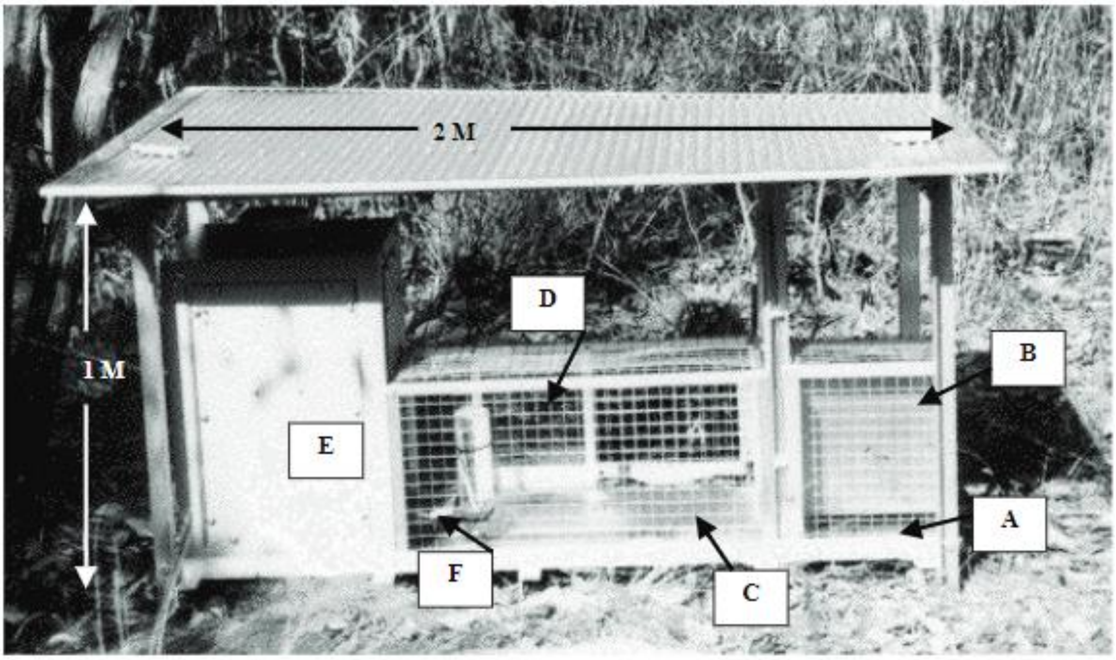
2.3.1 Food supplementation

Food supplementation is an animal conservation technique, whereby food sources are artificially supplied to a target animal species. This typically occurs where it is believed that the animals natural food sources are degraded to such an extent that it is negatively affecting the health of the animal population (Ewen et al., 2015). It is hoped that by providing an artificial food source, the target animals population may recover. However, non-target animals accessing the food source, the food inducing migration of other animals into the area, and adverse animal behaviours such as a larger aggressive animal protecting the food source from smaller individuals can confound the effectiveness of supplementary food provision (Isaac et al. 2004).

To mitigate the negative effects of food supplementation, various projects have created a system that could feed specific individuals, while prohibiting food access to other animals.

In a program by Isaac et al. (2004), wild brushtail possums (*Trichosurus vulpecula*) were captured and fitted with a collar, then released. A control group of Possums were also captured and tagged but were not given access to the food system. Five food delivery systems were installed in the vicinity the possums were captured and released in. A pressure plate at the entry of the system activated the electronics used to detect the radio signal. The controller would open the gate when a collar was detected. A scale, which would weight each animal when they were within the device was installed, along with a data logger, which would record the times, dates, weight and radio ID of each animal that entered the food delivery system were also installed. Once inside the system, the animal would be issued a set amount of food.

The study demonstrating that the use of supplemented foods to possums resulted in weight gain, while the control group were shown to have lost weight over that period.



A: Entrance area with pressure switch; B: weatherproof plastic box containing the transponder scanner; C: the balance scale in the main feeder area; D: 'cat-flap'-type exit door; E: weather-proof food and electronics enclosure; F: feed tray.

Figure 5 Brushtail Possum feeder system (Isaac et al., 2004).

The trial demonstrated that providing a controlled food source was beneficial to the target species, and that automation could be utilised to aid with conservation programs. The authors noted the benefits of the data loggers captured data in analysing and reviewing the health of the specimens, as it greatly reduced the labour hours of the researchers to collect that vital information.

Jewell (2013) wrote an essay that posed the question "how does the technique of monitoring animals affect those animal's natural behaviour?". The paper details the compounding effects of stress on the animals resulting from intrusive modes of investigation (capture, tag, release, recapture). The investigation into various intrusive modes on animal monitoring have detected a change in animal behaviour, which calls into doubt the validity of the data collected. Negative results on animal welfare are rarely reported, as it could jeopardize future funding. Non-invasive monitoring techniques were viewed more favourably by the pubic, increasing public engagement and project sustainability. Human-animal interactions during trials to capture, tag and measure the animal should be considered and monitored to ensure that they do not adversely affect the animal behaviour in such a way that would either invalidate the test results, or cause significant harm or stress onto the animal.

In a study by Watson et al. (2021), a wild-caught brush-tailed phascogale (*Phascogale tapoatafa*) was trained in captivity to use a microchip-automated feeder. This study provided a proof of concept that controlling food supplements with access-controlled doors was viable as it was demonstrated that wild native animals would approach something as foreign as a door system, and that they were able to learn to interact with the automated doors.

2.3.2 Nest Boxes

The replanted trees and shrubs in cleared areas are not suitable to provide nesting sites until a sufficient amount of time has elapsed for the natural hollows to occur (Arlettaz et al., 2010). As new trees that are planted during rehabilitation do not create the natural hollows for many years, the installation of nest-boxes is often a required component of habitat rehabilitation to meet the needs of the nesting animals.

When designing nesting boxes, a common strategy to help target a specific animal is to manipulate the nest-box entrance size, to reduce the likelihood of invasive or common animals using nest boxes (Lambrechtsetal.2012, Charter et al. 2016). Modification of nest-box design is not always effective however, as the physical requirements of the targeted and competitive species may be similar (Kiss et al. 2017).

When planning or performing nest-box installation to help support a target species, studies suggest that providing as many nest-boxes as possible can only benefit the target species. An investigation by Berthier et al. (2012) was performed into monitoring the introduction of a 700 nest-boxes for the Hoopoe bird in Switzerland. The team was tasked with recording the bird population and observing for any potentially negative effects. The team observed that the introduction of the nest-boxes resulted in a population boom resulting in the high availability of nest boxes. The team did not detect any negative effects to the overall health, genetic diversity or social structures of the birds resulting from the rapid increase of population, or the introduction of so many nest boxes.

To improve the efficacy of nest-box uptake, it is recommended that the installers alter or adjust the design of a nest-box to replicate the target animals natural requirements (Beyer and Goldingay, 2006). Lawton et al. (2022) conducted a project to install 450 nesting boxes targeting brush tail phascogales, in order to investigate how often these nesting boxes were utilised, and detect what factors, if any, affected the likelihood of a nesting box being utilised by the target species. The nest box inspections spanned over a period of 5 years and concluded that nesting boxes in Stringybark type trees were utilised more often than those mounted in gum bark or box trees, and that nest boxes installed on forest slopes were favoured over those installed in gullies.

2.3.3 Animal shelters with automated doors

One method of providing targeted access and reducing non-target use of nest boxes is to use an automated door. Edwards et al. (2020) performed a study to investigate the effectiveness of training wild-caught bandicoots to utilise microchip enabled doors to access food. The goal of the training was to analyse the viability of training animals in captivity to use a microchip enabled door to access an enclosure, which could provide food, a nesting site, or act as a safe haven – somewhere the newly released animal could go to escape predation.

The team captured a group of wild northern brown bandicoots, with each animal then being implanted with a microchip. The bandicoots were kept in a contained area with a nestbox, and through the use of food lures were enticed to interact with the door. The bandicoots were successfully trained to approach and enter the closed door of the box to access the food.

Figure 6 Northern brown bandicoot bunker (Edwards et. al. 2020)

The test demonstrated that animals could learn to approach a door system and be taught to push the door to gain access. The authors highlight that this could provide a mode of delivering supplemental food sources to target species without inadvertently feeding and attracting other animals. The trial noted that the electronics system required a connection to a wi-fi network to access information on individual animal access, reducing the designs feasibility in remote locations.

Using these same microchip enabled doors, a project to test whether wild animals could learn to utilise automated nest boxes in-situ was performed by Watson et al. (2022), whereby the team captured brushtail possums and brush-tailed phascogales and fitted each with a VHF tag and a microchip. Each microchip was married to a specific automated nest box, which was installed in the specific animals' major den sites. The animals were trained to use the doors that were held ajar, and through the use of food lures, taught to interact with the door and nest box.

Figure 7 Animal inside an automated nest box (Watson et al., 2022)

The authors discuss the costly nature of training animals in captivity to use microchip-controlled systems. The paper discusses the induced stress on the animal(s) brought about by the capture and relocation required for the training, mirroring concerns in a research paper by Jewell (2013). Finally, the authors highlight the potential for disease to be transmitted both to and from the wild animals that are captured and brought to the training facility.

2.3.4 Detection and monitoring systems in conservation projects

Detection and identification of both native and invasive species within an environment can provide researchers, caretakers, and management authorities with vital information to help drive effective flora and fauna management. Wildlife cameras, in collaboration with image processing programs are being deployed by conservation teams to survey large areas of land to detect and monitor the animal populations, which is utilised to inform and guide conservation plans and programs (Wildlife Insights, 2024). Early detection of pest species in an area can lead to a faster response from any management programs in place, and result in a better result for pest suppression or elimination (Epanchin-Niell et al., 2012). Animal survey data has proven such a critical resource that teams have even begun developing unmanned aerial drones in an attempt to gather survey data over large areas (Linchant et al., 2015)

2.4 Emerging Technologies

Computer technology sees continuous growth in complexity, capability and availability throughout the world. From the humble beginnings of relays performing additions in 1937 (Zeren, 2024) to self-guided space missions and the birth of complex AI computing, the effect of computing technology on the world is nothing short of astonishing. With advancements in manufacturing and a saturation in the global marketplace, powerful computing technology is more readily available than ever before. By enabling innovation, development and collaboration, the computing industry, through to hobbyists and enthusiasts have driven a boom in new computing system capability.

Emerging technologies in the computing, Internet or things (IoT) and object detection spaces shall be reviewed to identify any possible opportunities of project integration.

2.4.1 Computing

Cloud Computing

Advancements in internet capabilities has enabled the use of computer systems to be used remotely. Companies like Amazon Web Services (AWS) are able to provide a service whereby a user may rent out Amazons computing capabilities accessible via an online connection (AWS, 2024). This arrangement can facilitate a user gaining access to significantly powerful computing capabilities without the need to acquire and set up the computing systems locally. There is also significant potential for cost savings, as the user may only need to pay for a short period of time to utilise the technology, instead of needing to invest in the initial installation of the system (Azure, 2024).

Microcontroller boards

A microcontroller unit (MCU) is an integrated circuit with an embedded Central Processer Unit (CPU), onboard memory, various input/output terminals and may be created with various supplementary capabilities that allow the MCU to perform various computing and control tasks (Hashemi-Pour, 2024). As these chips became more capable in the tasks and functions they could perform, bespoke boards used to boost the functionality of these chips began to be developed. The Arduino is a one such platform, whereby a primary MCU is utilised to provide the computing capacity to a board that has been fitted out with a variety of components to streamline it’s use. The Arduino was originally designed to be used by students and hobbyists to learn how to use small computing platforms, however, the modularity and flexibility of the platform, supported by a strong community who develop open-source programs and extension, have elevated the Arduino to be a legitimate computing solution for a variety of real-world applications (Arduino, 2024). While Microcontroller boards, such as the Arduino are cost effective devices, often with a great deal of online support, the small MCU does limit the systems overall computing power and capability, trading simplicity and cost for CPU capability, RAM capacity and typically do not include any Graphics Processing capability.

Single Board Computers

Single Board Computers (SBC) exist in the middle ground between the Microcontroller board and the Personal Computer (PC) as they are typically understood of today. An SBC is a compact computer, with all vital components being installed on a single circuit board. Depending on the individual system, these SBC's can be utilised to provide a significant amount of computing capability, as the boards may include dedicated graphics processing capability, will often have a considerable amount of onboard RAM and storage memory, and will host a dedicated CPU (BAE systems, 2023). The where the Microcontrollers will typically be controlled via a dedicated interface, uploaded a specified script that would be run indefinitely, SBC's can be controlled using an operating system (typically Linux). Utilising a dedicated operating system allows for very complex control and calculations to be performed. While they significantly outperform their smaller Microcontroller board cousins for computing capacity, they do draw a great deal more energy to function and cannot be as quickly or easily powered down and back up as needed by the system, meaning that power supply becomes a significant consideration (MonoDAQ, 2024).

2.4.2 Object detection and AI

The ever-increasing processing power and complexity of modern computers has unlocked their potential to perform object detection and image recognition of content within photos, videos, or live feeds from digital cameras. A computational technique denoted as "deep learning" involves processing information through multiple stages of calculations and analytics. Each stage of computation is called a Neural Network. Deep learning leverage advances in computing capability to chain together many layers of networks, creating sophisticated analytical tools (Holdsworth and Scapicchio, 2024). Using these neural networks and by providing images of a specific object, the deep learning system can develop the capability to detect that object in other images.

Services such as Wildlife Insights provide an online service of animal recognition, whereby photos or video footage may be submitted for review to their online portal. Wildlife Insights utilise their broad dataset of animal photos to provide their animal detection software with a powerful capability to help process, detect and tag animals (Wildlife Insights, 2024).

Smaller, more specific image recognition software is beginning to enter the marketplace, enabling object detection capability for the hobbyists, researchers and small-scale industry, without the need for larger servers or powerful computing machines.

2.5 Findings

The literature review highlights a growing need to support our native species of mammals and birds as they struggle against the pressures imposed on them from the reduction of their natural environments, compounded by the effects of exotic animals both over-competing for the existing resources, or targeting them as easy prey.

Past projects have highlighted that the provision of nest-boxes or food supplements are valid strategies for supporting native animals by observing an increase in the health of populations that receive these supplements.

Later projects have utilised new technologies in automation and access control to more effectively deliver food or shelter, reducing the side effects of inadvertently aiding non-target animal species. These studies have highlighted that Australian native mammals are inquisitive and/or bold enough to approach and interact with quite foreign and artificial environments and systems, and that there is proven possibility for these animals to learn how to effectively utilise these automated systems.

The above findings support the proposed initiative to develop a system that can detect animals and allow or inhibit access, while collecting animal information. Throughout the previous studies, it is commonly noted that minimal or no human interaction with the animals is preferred, a recommendation driven by cost, practicality, minimising the risk of spreading illness among the animal population, or to avoid causing undue stress to the animals involved.

2.6 Establish Key Criteria

Based on the literature review, there is sufficient evidence to support the proposal to further develop tools and strategies in the environmental conservation space. From the review, and preliminary discussions with stakeholders within the Environmental Science team, the proposed nest box system shall:

a. **Behave positively with the interaction of a Sugar Glider.**
   This criterion establishes the requirement as set out by the Environmental Science project client.
b. **Facilitate minimal, if any, direct human interaction with any animals.**
   A key finding from previous animal studies was that human interaction with the animals was almost always detrimental to some degree, either through the increase of stress on the animal(s), adding time and labour burdens onto the research team members, or by providing a vector for disease transmission.

   For this project, both the literature review and the stakeholders' directions have indicated that the system should be designed with the intention to either minimise or eliminate the need to interact with the target animals, i.e. to not require capturing, tagging, or RFID tag implanting of animals.

c. **Pose no direct or indirect risk of harm to any animal interacting with the system**.
All research projects and proposals must stand up to scrutiny by the Animal Ethics Committee, who evaluate and approve / decline research proposals (Government, 2016). It is also imperative that the device cannot accidentally, carelessly or maliciously injure or kill an animal, as it is firstly morally wrong, and that it would breach the Animal Care and Protection Act 2001, potentially resulting in fines or imprisonment.

d. Allow, inhibit or deter access to the nest-box.
From the literature review, providing a supplement food or shelter to a specific animal can often inadvertently support a non-target species, creating additional pressures on the target species. To provide maximum benefit to the target animal, while ensuring that non-target species cannot interfere, a method of access control is required, to allow or inhibit access to the nest box from animals.

e. Control access depending on the animal that is attempting to access the nest box.

Following on from criteria d, a method of selectively allowing access to a certain animal is required to operate the access control system.

f. Be designed to be installed and operate in the natural habitats of the animals.

One of the common findings from the literature review is that minimum human interaction is a desirable feature for the system. From this finding, the following assumptions can be made:

- The system must operate autonomously, without the requirement for human operator interactions.
- The system is intended to be located in the animal's natural environment – i.e. deployed in either wild or urban habitats, and not within a controlled environment, so as to eliminate the requirement for animal capture and re-location.

g. Record or report animal interactions for behavioural and system analytics.

Animal data has been identified as a crucial component to any environmental conservation program. Any data that the system could collect may be valuable to researchers both in analysing the effectiveness of the nest box itself, as well as detecting what other animals are in proximity to the box, be they predators looking to raid the nest, or competing species of animals looking to take over habitation of the box.

In order to analyse the effectiveness of the nest box and study the animal behaviours while interacting with the system, a mode of data reporting shall be required. Inclusions of system health monitoring would also be prudent in the design, to facilitate the research team responding to potential faults in a timely manner.

3 Concept Design

3.1 Criteria Analysis

To create the design criteria for the project, each key criteria highlighted in the Literature review shall be reviewed. Assumptions shall be listed, and potential options analysed to determine their suitability for project implementation.

3.1.1 Behave positively with the interaction of a Sugar Glider

The research team have identified that the system shall first be designed and calibrated with the intention to interact with the Sugar Glider. The following section shall investigate the characteristics, behaviours and nesting requirements of the Sugar Glider in order to suitably design the system by meeting their key criteria.

Animal Description and behaviours

Sugar gliders are small marsupials, typically growing to around 25 cm long and weighing around 100 grams. They are mostly nocturnal, nesting in tree hollows during the day and foraging for their food among the trees at night. Their diet consists of a variety of foods, ranging from nuts, seeds, flower and berries to insects, eggs and small birds (Australian Wildlife Conservancy, 2024). As their name suggests, sugar gliders are capable of jumping from one tree to another, stretching their arms and legs outwards to create a sort of sail.

Predators and competitors

Given their small size, they are seen as a prey item for a variety of native animals, including snakes, quolls, owls, and Kookaburras, along with the exotic cats, dogs and foxes (Barritt M., 2024)

Design criteria

- A wooden nest box simulating a tree hollow is appropriate to meet the nesting needs of the Sugar Glider.
- Their arboreal behaviour highlights their dexterity for climbing trees and branches, therefore it shall be assumed that the animal may approach the nest box from any direction, potentially climbing on or around any exposed surface of the nest box.
- Their nocturnal behaviour indicates that the camera system will need to be fully functional in the dark, requiring minimal natural or artificial visible lighting to aide in photo quality.
- The access system shall be required to either positively identify the Sugar Glider approaching the door or shall be required to detect and identify the predators listed above (at a minimum) to activate the lock.

3.1.2 Facilitate minimal direct human interaction with animal(s)

Given that minimal interaction with animals is required for the system to operate, the following assumptions can be made:

- The system shall not be dependent on any device that is to be attached or implanted into the animal, removing the need for animals to be captured and released.

- The system shall be designed to be fully autonomous, being able to be deployed and left unattended, to eliminate the animals from being in proximity to the researchers.

- There must be minimal effort on behalf of the animal to operate the system, therefore the system must operate intuitively, eliminating the need to capture and relocate an animal to a training facility.

Design criteria

- No tagging or collars shall be utilised
- Operate autonomously, with minimal human interaction or interference. Initial target aim shall be 2-3 days of autonomous operation.
- No required behaviour by the target animal is to be performed to trigger the system. The system must trigger with the animal(s) interacting with the system naturally and intuitively.

3.1.3 No risk of animal harm

Electrical risk

Any system that can perform advanced automated functions will require a source of electrical energy to both power any computing components or operate any motors or actuators needed to latch doors or hatches. Any other energy source (chemical, potential, spring etc) will be either impractical or too convoluted to safely, accurately and repeatedly achieve the requirements of the project.

This project shall ensure that all electrical energy utilised anywhere within the system shall meet the requirements of Safety Extra-Low Voltage (SELV), which is specified as a requirement by Australian Standards where electrical equipment could potentially pose a significant risk or hazard (Guide, 2024). To meet this standard, electrical potential anywhere in the system shall not exceed 50 V AC or 120 V DC. To achieve this, the equipment shall be operated by battery power. By choosing appropriate battery supplies, it can be ensured that no over-voltage may occur.

Ensnarement risk

It must be ensured that there is minimal risk to an animal entering the nest-box and consequently being unable to exit at any time. Well intentioned nest boxes made of hard plastic materials have resulted in animal becoming trapped, as their claws and/or beaks were unable to get a firm purchase on the hard smooth material (McGlashan A., 2000). It is recommended that steps or a ladder structure also be installed inside the box to aide in animals climbing to the door (Cherriman, 2015).

Given the concept design nature of the digital control system, it cannot be assured that the control programming will operate flawlessly at all times. Unintended code glitches or electronics failures can occur; therefore, it must be assured that the safety of the animals is not reliant on the system operating normally.

Design Criteria

- The system shall use Extra-Low Voltage energy potential.
- Egress control from the box shall be either uncontrolled (always allowed), or operated by simple, non-automated functions, ensuring that an unforeseen system error does not cause the lock mechanism to fail and trap an animal.
- Doorways or hatches shall be unencumbered by obstacles, pinch points, or componentry, ensuring reliable operation of the door without the risk of the door fouling and becoming stuck.
- The interior of the nest box shall be made of a material that is suitable for animals to climb on or gain a firm foothold. Smooth plastics are not appropriate as a nest box construction material.

3.1.4 Operate in the natural habitats of the animals

In order to provide the most effectiveness to the target native animal as a nest box, the boxes are to be designed ready for deployment in natural environments of the animals.

Given the range of animal that the boxes could be calibrated to provide shelter to, it shall be assumed that most outdoor locations around Australia could be selected as a potential install site.

Weatherproofing

The system shall be installed in an open environment, therefore adequate protection from the weather and local climate shall be required. For the initial system design, it shall be assumed that the system shall be appropriately prepared in it's entirely for installation in the field i.e. no additional weatherproofing shall be needed to protect the system from dust, rain or wind, to simplify the installation, and minimise any disturbance to the location the nest box is to be installed. Protection of the ingress of insects shall also be considered.

Given the box is to be designed to mee the needs of a sugar glider, the box is to be installed at a height above the ground, therefore it shall be assumed that the box will not be exposed to flooding or emersion in water from below. It shall be a requirement of the installation team to assure that the box shall be installed in such a way as to prevent water pooling at the base of the box, and that water is allowed to run off freely.

Communications

While Australia's urban areas are serviced by telecommunications network coverage, there are still significant portions of regional areas and remote locations that have either poor of non-existent connections to the cellular network (Infrastructure Australia, 2023). Given the potential target location for the nest box may be in a remote location, it shall be assumed that a consistent connection to Australia's mobile network carriers should not be relied upon to operate the system.

Design Criteria

- A consistent high-speed cellular connection shall not be required to operate effectively.
- The system as a whole shall be rated to Australia Ingress Protection rating of IP54 (Dust protection and water splashing at any angle).

3.2 Hardware and Software Design

3.2.1 Compile design criteria

From 3.2.1

- A wooden nest box simulating a tree hollow is appropriate to meet the nesting needs of the Sugar Glider.
- Their arboreal behaviour highlights their dexterity for climbing trees and branches, therefore it shall be assumed that the animal may approach the nest box from any direction, potentially climbing on or around any exposed surface of the nest box.
- Their nocturnal behaviour indicates that the camera system will need to be fully functional in the dark, requiring minimal natural or artificial visible lighting to aide in photo quality.
- The access system shall be required to either positively identify the Sugar Glider approaching the door or shall be required to detect and identify the predators listed above (at a minimum) to activate the lock.

From 3.2.2

- No tagging or collars shall be utilised
- Operate autonomously, with minimal human interaction or interference. Initial target aim shall be 2-3 days of autonomous operation.

No required behaviour by the target animal is to be performed to trigger the system. The system must trigger with the animal(s) interacting with the system naturally and intuitively.

From 3.2.3

- The system shall use Extra-Low Voltage energy potential.
- Egress control from the box shall be either uncontrolled (always allowed), or operated by simple, non-automated functions, ensuring that an unforeseen system error does not cause the lock mechanism to fail and trap an animal.
- Doorways or hatches shall be unencumbered by obstacles, pinch points, or componentry, ensuring reliable operation of the door without the risk of the door fouling and becoming stuck.
- The interior of the nest box shall be made of a material that is suitable for animals to climb on or gain a firm foothold. Smooth plastics are not appropriate as a nest box construction material.

3.2.2 Access control method investigations

A variety of access control methods and techniques shall be analysed to determine their project suitability. Each option shall be listed in Table 1, identifying their pros and cons with supporting references where required. Once completed, each option shall be weighed against the projects key criteria, where it shall be identified which option(s) meet all of the identified key criteria. Failure to meet one or more of the requirements will exclude the potential system from being selected for implementation.

Table 1 - Access Control methods review

Control system	Description	Pros	Cons
Noise deterrent	A noise deterrent system could be triggered when an unapproved animal approaches the nest box.	Easily controlled by electronic system Battery controlled No risk of animal entrapment Cost effective Simple design	Repeated exposure to noise can dimmish the deterrent response (Kok A. et al., 2021) Nearby target species may be inadvertently deterred Motivated predators may disregard the noise deterrent
Chemical deterrent	A spray of aerosol chemical could be used to deter unapproved animals from approaching the nest-box	Controllable by electronic system No risk of animal entrapment Simple design Low power consumption to operate	Potential for animal harm – dosage may be incorrect, animal may receive the spray to the eye or directly to the face, potential for allergic reaction. Target species may be inadvertently deterred by chemical repellent lingering in the box or area.

			Chemical may be ineffective against all potential non-target species. Chemical would require reloading with high use. Would require reliable actuator device to deliver chemical dose.
Mechanical restriction	A door or hatch may be installed on the nest box to provide access control	Controllable by electronic system Low power consumption to operate. Battery powered No risk of inadvertently deterring target species when non-target species interacts with the system	Potential risk for animal entrapment* Potential for mechanism failure, rendering the latch ineffective * Mechanical restriction can be designed in such a way as to always allow for animal egress upon system failure
Decoy / Fake predators	Installation of fake or decoy predators in the vicinity of the nest box to deter animals	No power consumption No possibility of mechanical or electrical failure. Always on Cheap and robust	Indiscriminate deterrent. Potential to deter target species Unable to adjust or calibrate controls to allow / disallow other species Effectiveness of deterrent fades over time (DiLonardo M., 2022)

		Strong initial results (Blackwell M., 2024)	Potentially difficult to deter all potential non-target species while not deterring target species.
Electric shock deterrent	An electric shock could be administered to deter non-targe species	Controllable by electronic system Battery powered No risk of inadvertently deterring target species when non-target species interacts with the system No risk of animal entrapment	High potential for animal harm High power consumption Complex system to apply electrical contact to non-target species Poor electrical connection may not adequately deter animal

3.2.2.1 Access system selection

As previously identified, the relevant key criteria for an effective system are as follows:

1. No potential for animal harm
2. Is able to selectively allow or deter target animals
3. Maintains effectiveness over time
4. Does not cause a lingering or unintended deterrent

Table 2 - Deterrent or access control criteria compatibility

System / Requirement	1	2	3	4	System Suitability?
Noise deterrent	YES	YES	NO	NO	NO
Chemical deterrent	NO	YES	YES	NO	NO
Mechanical restriction	YES	YES	YES	YES	YES
Decoys	YES	NO	NO	NO	NO
Electric shock	NO	YES	YES	YES	NO

Result

A mechanical mode of access control shall be selected for the system. This may be in the form of lockable swinging of sliding doors, auto-opening doors etc. Given the risks highlighted in the previous section, any mechanical door shall only require ELV power supply. The designers shall ensure that any door or latch implemented cannot create an entrapment risk should the system fail to control the door as required.

3.2.3 Animal identification options

Human operator input

Having an operator or observer provide the system with instructions is the simplest mode of animal detection. The operator would communicate with the system with either a wired or wireless communication device, sending the signal to allow or prohibit an animal. The operator is likely to provide very accurate animal detection, depending on the training and skill of the observer.

While this option provides a high level of detection confidence, this option requires a great deal of operator time and concentration and may likely be too taxing on the research team to be viable. The design also contravenes a key requirement, as the human operator would likely need to be in the vicinity of the device.

RFID

A collar or tag I attached to the target animal that is fitted with an Radio-Frequency-Identification (RFID)circuit. This Circuit can be detected and identified by nearby RFID readers.

Given requirement E discussed above, RFID is not an available method to provide the system with animal identification as that method requires the capture, tagging and releasing of the animal(s).

Pressure Switch

The feeder box project by Isaac et al. (2004) utilised a pressure plate to activate the detection device. A pressure plate may be viable for nest boxes located on the ground but may be ineffective for nest boxes installed in trees to accommodate arboreal species of mammals, or hollow nesting birds. As discussed in an earlier section, the Sugar Glider is adept at climbing, therefore it cannot be guaranteed that the glider will traverse over a pressure plate.

PIR sensor

Passive Infra-Red (PIR) sensors could be utilised to detect animals at the location of the nest box. They are simple and robust, and operate by detecting heat energy within close proximity to the sensor, which is emitted by animals and birds (Arrow, 2018). PIR sensors are not able to determine what animal is being detected, only that an animal is likely to be in the vicinity of the detector. PIRs are unlikely to provide the full solution but may be implemented as a supplementary detection system.

Image Recognition

Continuous growth in computing capability, coupled with the development of AI software has enabled the use of the training of image recognition software to detect and recognise specific species of animals captured by a digital camera. This image recognition can be utilised to detect a target species of animal and provide a signal or specific data output depending on the image analysis.

This system allows for specific control methods depending on the species of animal identified. It is however reliant on accurate AI models correctly identifying the animal, as well as a fixed camera capturing a clear photo of the animal for the image recognition software to process.

3.2.3.1 Animal Detection System Selection

Utilising the key criteria identified in the concept design, the following relevant key criteria shall be utilised to determine system suitability:

1. Behave positively with the interaction of a Sugar Glider.
2. Facilitate minimal, if any, direct human interaction with any animals.
3. Control access depending on the animal that is attempting to access the nest box.
4. Be designed to be installed and operate in the natural habitats of the animals.

Table 3 - Animal Detection criteria compatibility

Process / Requirement	1	2	3	4	Meets all criteria?
Human Operator	YES	NO	YES	YES	NO
RFID	YES	NO	YES	YES	NO
Pressure Switch	NO	YES	NO	NO	NO
PIR Sensor	YES	YES	NO	YES	NO
Image Recognition	YES	YES	YES	YES	YES

Result

From the analysis above, image recognition provides an appropriate mode of animal detection for the proposed system and shall be implemented as the method of identification.

3.2.4 System power

Given the requirements for remote installation of the system, it shall be assumed that a connection to a permanent power grid is not practical or always available. A method for delivering power in remote locations shall be investigated below:

Solar Power

Solar panels may be able to be deployed to provide system power in remote locations. Off-grid systems typically leverage solar energy technologies to provide remote locations with an energy source (ARENA, 2024).

Pros	Cons
- Portable system - Quiet	- Daytime power supply only, unless battery is also implemented - Shade significantly decreases the energy generated - Panel size may become quite large, depending on Wattage required - Potential for animal damage - Wiring needed from the panels to the nest box.

Power Generator

A small portable generator could be deployed to provide electrical energy to the system

Pros	Cons
- Consistent energy output	- Potential noise pollution. - Potential air pollution. - Requires refuelling - Potential for animal damage - Wiring needed from the panels to the nest box.

Battery Power

Batteries could be installed to provide electrical energy to the system.

Pros	Cons
- Not reliant on Day/Night cycles - Able to be installed into the system, minimising cabling and animal damage potential	- Requires recharging - Potentially heavy, depending on power duration requirements.

3.2.4.1 *Power System Selection*

Utilising the key criteria identified in the concept design, the following relevant key criteria shall be utilised to determine system suitability:

1. Facilitate minimal, if any, direct human interaction with any animals.
2. Be designed to be installed and operate in the natural habitats of the animals.
3. Pose no direct or indirect risk of harm to any animal interacting with the system.

Table 4 – Power supply criteria compatibility

Process / Requirement	1	2	3	Meets all criteria?
Solar Panels	YES	NO	YES	NO
Generator	YES	NO	NO	NO
Battery	YES	YES	YES	YES

Result

From the analysis above, battery power provides an safe and effective power supply to a remotely installed system. Solar panels were deemed not to be effective given the potential tree coverage, however, they may be applicable as a supplemental supply to extend battery life.

3.2.5 Record or report animal interactions

Animal interactions records are required to be kept for analysis and review. It would also be prudent for the system to provide some self-reporting to monitor system health, to detect any possible malfunctions or needs for maintenance.

In order to provide data on the status of the system, and send data collected in the field, a communications system is to be implemented into the system. The system shall provide a minimum of one report a day, which should ideally include any alarms or warnings regarding the system (e.g. battery failure), any logged events, and may potentially include photos. The data is to be received and processed to an online platform, which shall be accessible by the research team(s).

The locations for the system to be installed has not yet been finalised, therefore it shall be assumed that the location of installation may range from dense urban areas to remote wilderness.

The following is an analysis of the potential data management and communications systems

Local Storage of data

A simple method of reporting is creating a data file held internally to the control system, which the researcher can access, either on site or remotely, to review the event logs.

Pros	Cons
- Simple to set up - Does not require extra communications equipment.	- Cannot remotely monitor the device, must attend site and connect directly to the system / device.

Wired Communications

A wired communications system may be implemented to transfer data from the device to a computer or network port. Depending on the complexity of the signal or message, various wiring systems could be implemented.

Pros	Cons
- Simple communications system, very likely the computing system has native communications protocols available (UART, Ethernet network etc.)	- Receiving system must decode the signals to generate the desired message. - Cables could be exposed to the animals and elements. - Cable resistance limits cable length, depending on signal type. Receiver would need to be in close proximity (~100 m Max for Cat 5E digital signal) - Very remote box installations could be impractical to connect to via wired communications.

Wireless Communications

There is a broad range of wireless communications technologies available today for commercial and private use that could be leveraged to provide wireless communication capability. Short range communications, Infra-Red signal, Bluetooth communications protocol, or short-range radio device could be implemented. For longer ranges, telecommunications networks may be used to carry communications signals, either via their 3G/4G/5G cellular signals, or through a low bandwidth nb-IoT network. For long range or remote communications, Satellite technology may be required (Connected Platforms, 2024). It may also be possible to established an independent communications network, utilising Long Range Radio, or with a system of repeater stations and a wired or wireless internet connection to provide data communications (The Things Network, 2024).

Each of the options above have strengths and constraints that are to be analysed to ascertain their suitability for the project.

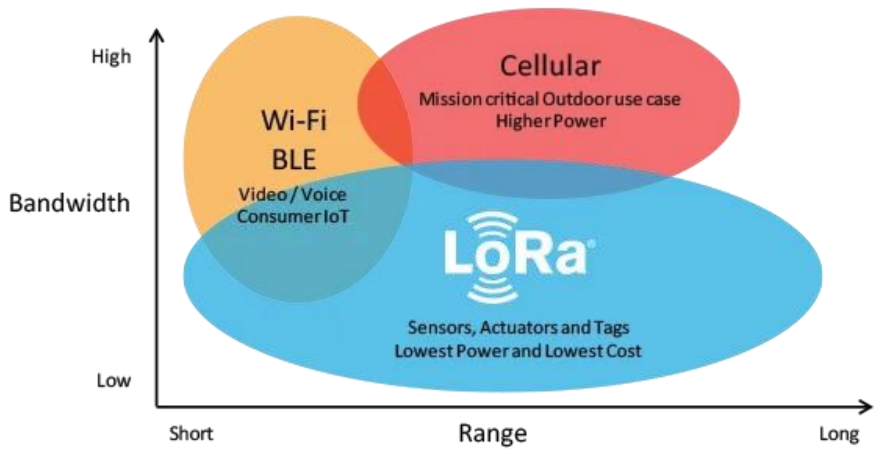


Figure 8 Transmission modes with transfer distance (The Things Network, 2024)

Table 5 - Wireless communication system comparison

Range	System	Pros	Cons
Short	**Bluetooth**	- Components are widely available - Low cost to set up - Able to handle high bandwidth	- Short range (typically up to 10 m, Long-range vriant can manage 100 m)
	IR	- Components are widely available - Well understood by the robotics and tinkering community – Libraries for control are freely available.	- Very short range (~ 1 – 2 m) - Obstructions (leaves branches etc) can interfere with signal.
	Independent Network*	- Very customisable - Can be built to provide high speed data connectivity to a target site.	-fixed location, relocating the nest box may require augmenting the system. - High upfront cost to establishing the network - Network maintenance must be arranged by the researchers.
↓	**3G / 4G / 5G**	- Network is established in urban and some rural areas. - Maintenance of the network is not the responsibility of the researchers - Data easily accessed and organised through telecom companies.	- Network location availability is fixed by the carriers - Remote locations may not have adequate signal strength. Consistent cellular connectivity tends to be heavy on power draw.
	Long Range Radio	- Able to broadcast up to 16 km (at 25 W signal power) - Repeaters may be implemented to boost signal distance. - Signal sender can be relocated easily (provided it is still in range of the receiver)	- Sender and receiving tower will need to be installed by the research team. - Line of sight to signal receive is typically needed. - Terrain and weather may affect connectivity.
Long	**Satellite**	- Australia wide coverage - Highly customisable messages can be created. - Does not require any extra network equipment (towers, repeaters etc)	- Small message packets - Message configuration is static in the field. - Clear line of sight to the open sky required. - High upfront cost.

Discussion and results: As the location of the proposed nest box installation has not been confirmed at this stage, no assumptions can be made as to the availability of telecommunications networks that may be utilised. It also cannot be assured that the nest box will be installed in a single fixed location for the duration of the research, and hence, the suggestions of establishing either a bespoke independent communications network cannot be made.

To enable the team to install the nest box without any undue limitations, satellite communications should be investigated as a mode of communications for the project.

3.2.6 Final Design Criteria

Driven from the design investigations above, the following system specs have been determined:

Table 6 - Concept design system specifications

Component / Function	Details
Animal Safety	The nest box shall be made of wood, to facilitate a firm safe foothold for animals climbing inside and outside of the box. A smooth plastic material shall not be used, removing the chance of animal entrapment. Easy Egress shall always be available. Electrical potential shall be kept as low as possible, with a max allowable voltage of 50 VAC , 100 VDC
System Control	Compact Computing technology shall be utilised to provide on-site intelligence, analytics, reporting and access control
Nest Box Access	A mechanically actuated door or hatch shall be implemented to provide access to the nest box
Animal recognition	Image recognition shall be utilised to provide animal data input to the logic system to determine access permission.
Access Control	Given the access and identification system requirements, access must be controllable by electronic signal or power, i.e. a lock or catch must be electrically actuated.
Power	Battery power shall be supplied to provide electrical energy to the system
Data collection	On-board data recording shall be implemented, and wireless satellite communications system shall be implemented to provide research data and system information.

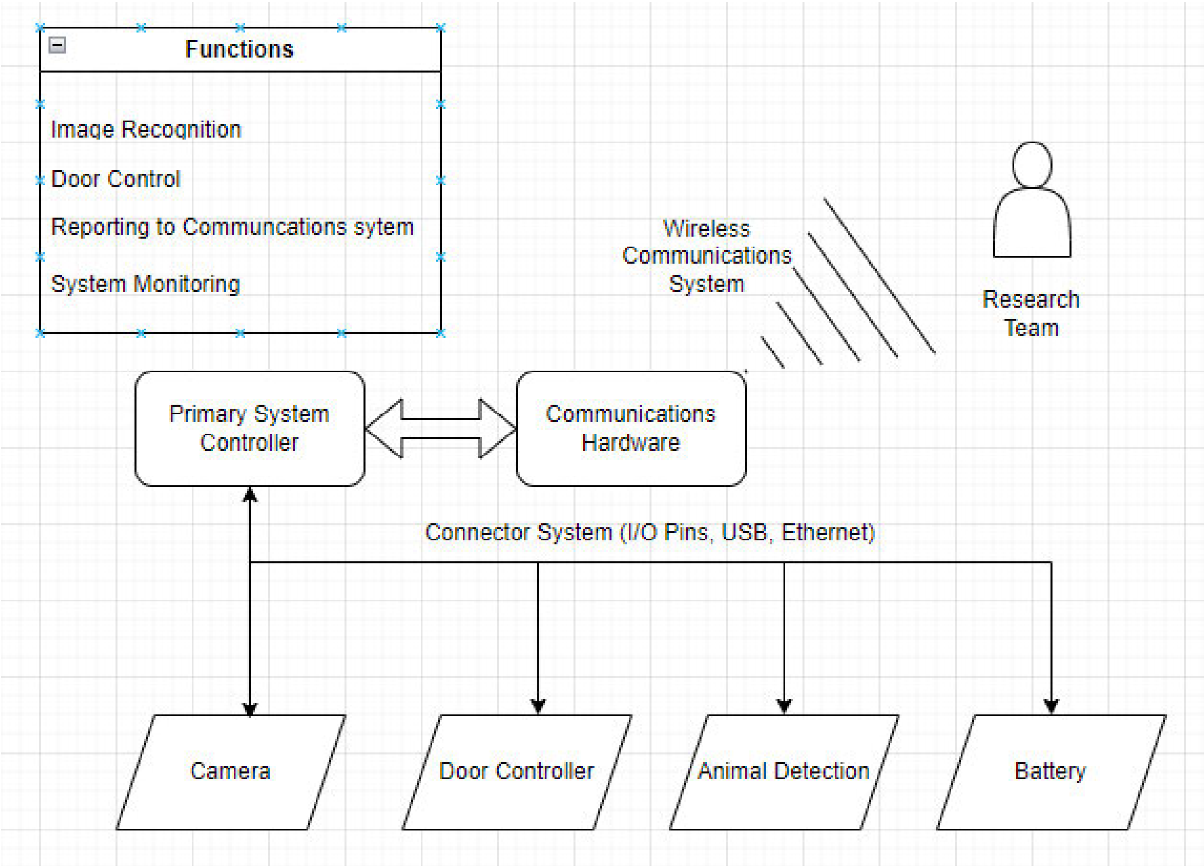


Figure 9 Concept Design

4 Detailed Design

4.1 Introduction and structure

With the structure of the project now established through the concept design, each sub-system shall be designed, meeting the requirements of the overall project, and ensuring that dependencies with sub-systems that share a connection or relationship are met. Each chapter shall focus on the sub-system indicated in the flow chart shown below. Lower systems have dependencies on the system(s) above.

System design

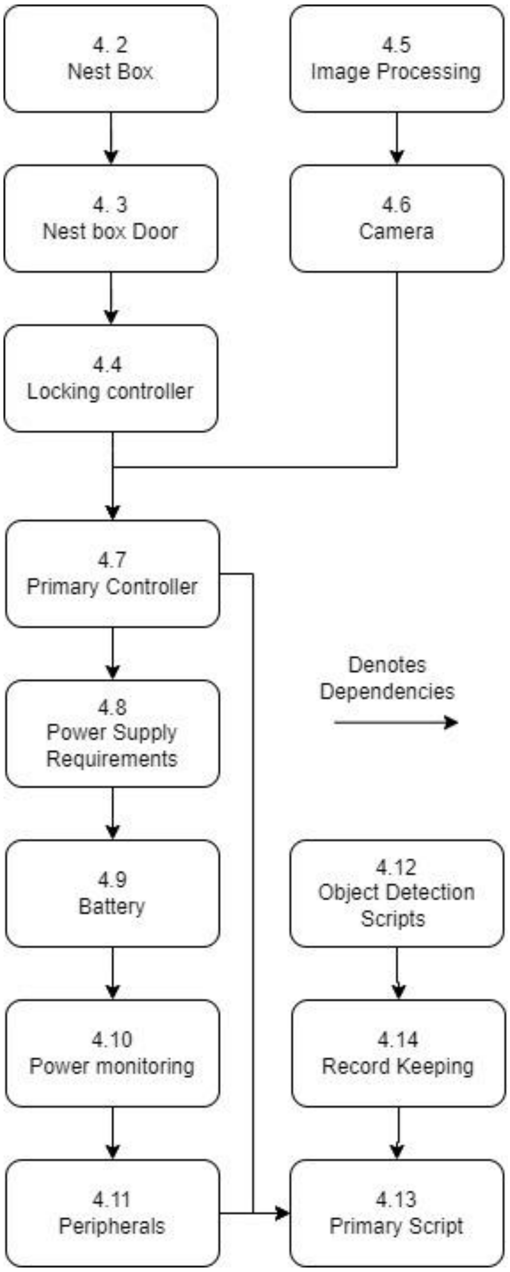


Figure 10 Detailed design structure

Design rationale

1. The nest box shall be selected as the structural foundation of the project. This box shall then enable the selection of the nest box door, which will need to match the physical dimensions of the nest to which it is installed.
2. Nest box door shall be selected. With the nest box selected, an investigation into the door components can be completed. Any hardware or software requirements for operating the door locking component can then be used to inform the design for the controlling system.
3. With the door system selected, a method of control shall be designed. The linear structure of the report is not entirely fit for purpose regarding the design for this subsystem. The capabilities of the main controller could potentially affect the final design for the locking controller. A basic design shall be created, then an amendment to the section, where required, providing any updates resulting from the selection of the main controller.
4. The partner design team are to supply the technical details for the image processing systems, for integration into the main system. This section shall only be identifying hardware and software requirements to enable the imaging software to function effectively.
5. As above, the camera technical details are to be provided by the partner team. Similarly, no designing shall be required, only an understanding of the technical requirements.
6. With the system requirements documented, a primary controller system can now be selected for implementation.
7. With a controller selected, power requirements will need to be well understood in order to calculate battery requirements.
8. Battery technology shall be reviewed to ensure the selection meets both power demands and overall project requirements.
9. To ensure that the operators of the system are able to replace any depleted batteries before the system loses power, a method of monitoring battery power shall be implemented.
10. A supplemental animal detection system shall be investigated to potentially reduce the overall power load on the system, and any other required peripheral systems shall be investigated.
11. A review of the supplied object detection scripts shall be performed to identify script functionality and investigate system integration.
12. Considerations into data keeping shall be performed, to ensure a record of all interactions the system detects are kept for review by the research team.
13. Finally, the primary script to perform all controls shall be created to meet all requirements identified above.

4.2 Nest Box

Given the project scope to analyse the effectiveness of controlling access to nest boxes, this project shall attempt to match as near as possible to the conventional nest box arrangements. This shall be implemented to minimise any unforeseen behavioural changes to the animals interacting with a nest box arrangement that could adversely affect any detected behaviours, potentially skewing test results.

To that end, the following characteristics shall apply:

Table 7 - Nest box characteristics

Material	Shape	Dimensions (L,W,H) (mm)	Entrance
Softwoods	Rectangular Prism	370, 240, 280[1]	N/A [2]

Notes:

1. The above dimensions are larger than the recommendations for a Sugar Glider (Nest Box Tales, 2024), however, smaller dimensions may not be able to accommodate the door / latch mechanism
2. The nest box entrance shall not follow the recommendations for nest box entrances. Nest box entrances are typically sized to prohibit predators, which is redundant in this instance.

Box design

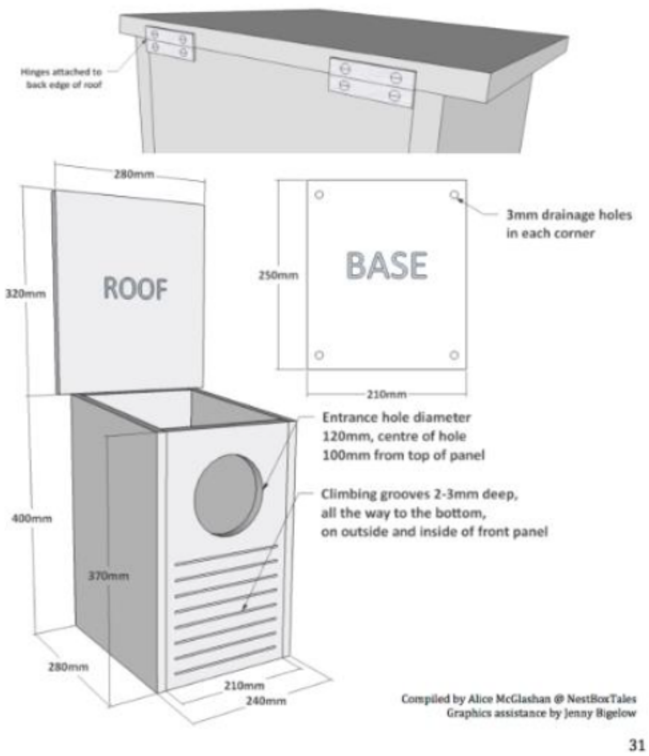


Figure 11 Typical Nest box arrangement (Nest Box Tales, 2024)

4.3 Door / Latch system

Given the requirements formulated in the concept design, the following key criteria apply to the door or gate mechanism:

- The door shall always allow for egress of the nest box, regardless of the state or setting of the control system.
- The door shall lock or unlock utilising an electrical or electronic control.
- The door shall operate on Extra-Low-Voltage potential.

From reviewing the installation method from Edwards et al. (2020) and (Watson et al., 2022), it can be identified that a commercially available pet door system was implemented, providing the access control mechanism.

This method of access shall be implemented for this nest box system for the following reasons:

- Simplified design process – Utilising a commercially available door saves time in the design and manufacturing steps, eliminating the need to design and perform stress / function testing.
- Cost effective – A commercially available door is likely to be at least as cheap than a product specifically built for the project. While there may be some savings in materials, labour costs to design, build and test a bespoke door will typically end up costing more than the purchased door.
- Commercially available doors allow for simplified repeatability of additional boxes by eliminating the requirement to fabricate a custom door.
- Product reliability – a commercially available door is assumed to provide reliable and repeatable functionality. In the case of the project, the door freely swinging outward from the nest box is a critical requirement must be assured. From the above-mentioned projects utilising the pet door, there was no mention of a failure of the door to operate as required, therefore it is assumed that the product performed as required being utilised as a nest box door.
- The door meets all design criteria for this project, as it is operated by a small DC motor to actuate the locking mechanism, and that the door is designed for unrestricted egress by a pet.

Door design

A pet door targeted for the use by cats shall be utilised, as the dimensions shown in the figure below are the smallest door dimensions available, matching as near as possible the potential appropriate door size for the target animal(s).

The RFID componentry for the door shall be disconnected, and a bespoke connection to the internal motor shall be made to facilitate locking control. The battery components shall also be disconnected to simplify the system, as the battery capacity available within the door enclosure is likely not going to be adequate to supply the entire system.

The pet door shall be retrofit to the Possum nest box in the location of the original Entrance hole.

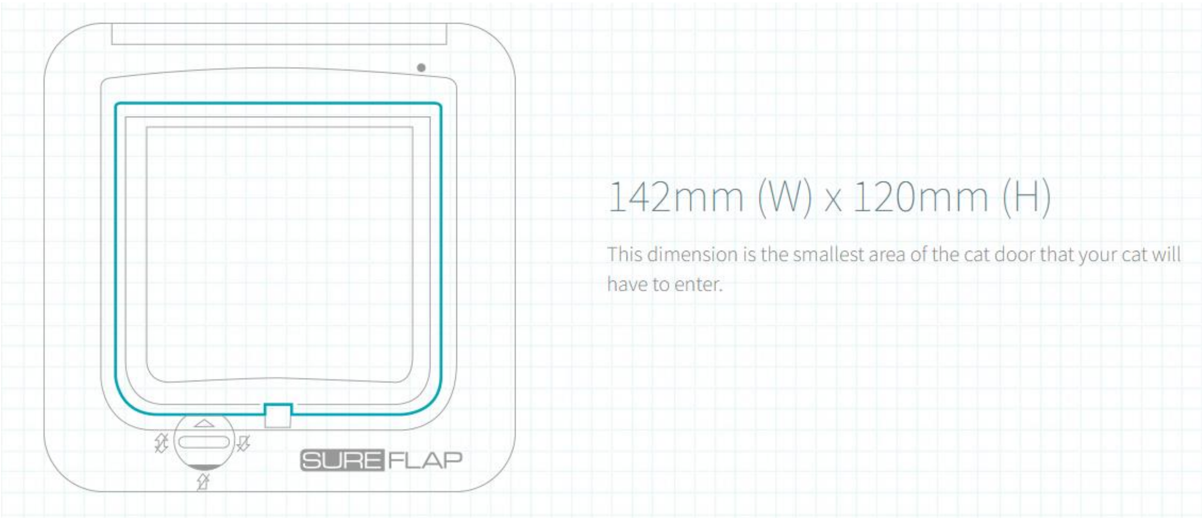


Figure 12 Pet door dimensions (SurePet, 2024)

4.4 Locking mechanism

Visual inspection of the dismantled pet door identified that a small DC motor provided the lock actuation. There were no markings or numbers on the body of the motor to confirm a model number.

4.4.1 Motor Power supply

While the motor may be unmarked, and a model number unavailable, there are some assumptions that can safely be made:

The battery compartment for the cat door requires 4 x AA batteries. A typical AA battery typically deliver 1.5 V. Where batteries are connected in Series, the voltage across both batteries in the sum of the two battery Voltages. Where batteries are connected in parallel, the voltage across the batteries remains the same as each individual battery (assuming the batteries are of matching voltages) (Renogy, 2023).

The possible battery configurations, and their associated voltages are shown in the table below:

Table 8 - Battery connection arrangement calculations

Battery configuration	Voltage Calculation
4 x batteries in series	$V_{out} = 4 * 1.5$ $V_{out} = 6\ V$
Parallel pair of 2 x batteries in series	$V_{out} = 2 * 1.5$ $V_{out} = 3\ V$
4 x batteries in parallel	$V_{out} = 1.5\ V$

Given the above potential battery configurations, the range of voltage to consider is 1.5 V to 6 V.

The simplistic design of a small, brushed DC motor, overvoltage for very short periods of time shall not present a risk of damage to the motor. The 5.5 V power supply to the system shall be an acceptable Voltage to operate the DC motor actuating the locking mechanism.

Small DC motors between 1.5 and 5 V typically draw 50 – 100 mA at no load, and up to 500 mA when in a stalled state (hub360, 2024). Current outputs of common Microcontrollers or SBC’s are typically very low (>5 mA), therefore a motor controller shall be required to provide both sufficient amperage to the motor, and to protect the control system from damage.

4.4.2 Motor Driver Selection

For this project, the motor controller IC L9110H shall be implemented. An investigation into the system requirements against the performance characteristics of the motor controller has been performed below.

Table 9 - Motor Controller requirements / L9110H comparison

	Input Voltage	Input Current	Output Voltage	Output Current	Logic Voltages	
System Requirement	5 V DC [1]	2 mA max [2]	6 V Max [3]	500 mA max [4]	Low=0 V [5]	High = 3 V [5]
L9110H motor controller performance characteristics (Components, 2024)	2.5 – 12 V;	350 µA	Input Voltage - 1.5 V	800 mA	Low=0.7 V MAX	High = 2.5 V MIN

1. Input Voltage to be compatible with supply voltage to the system for design simplicity.
2. 2 mA current is the max current the GPIO pins may supply.
3. Calculated maximum for the door motor, as per section 4.4.1
4. Max allowable current, as per section 4.4.1
5. Output voltage of LOW logic level of GPIO pins.
6. Output voltage of HIGH logic level of GPIO pins.

Connections and control

The pinout for the L9110H shown below shall been included to help inform the installers identify each pins function.

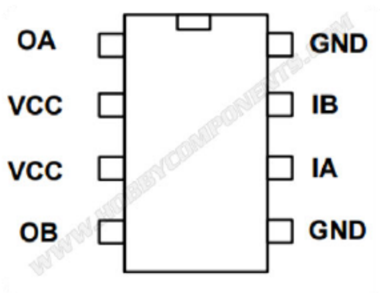


Figure 13 L9110H motor driver IC pinout

The motor controller shall be wired as shown below.

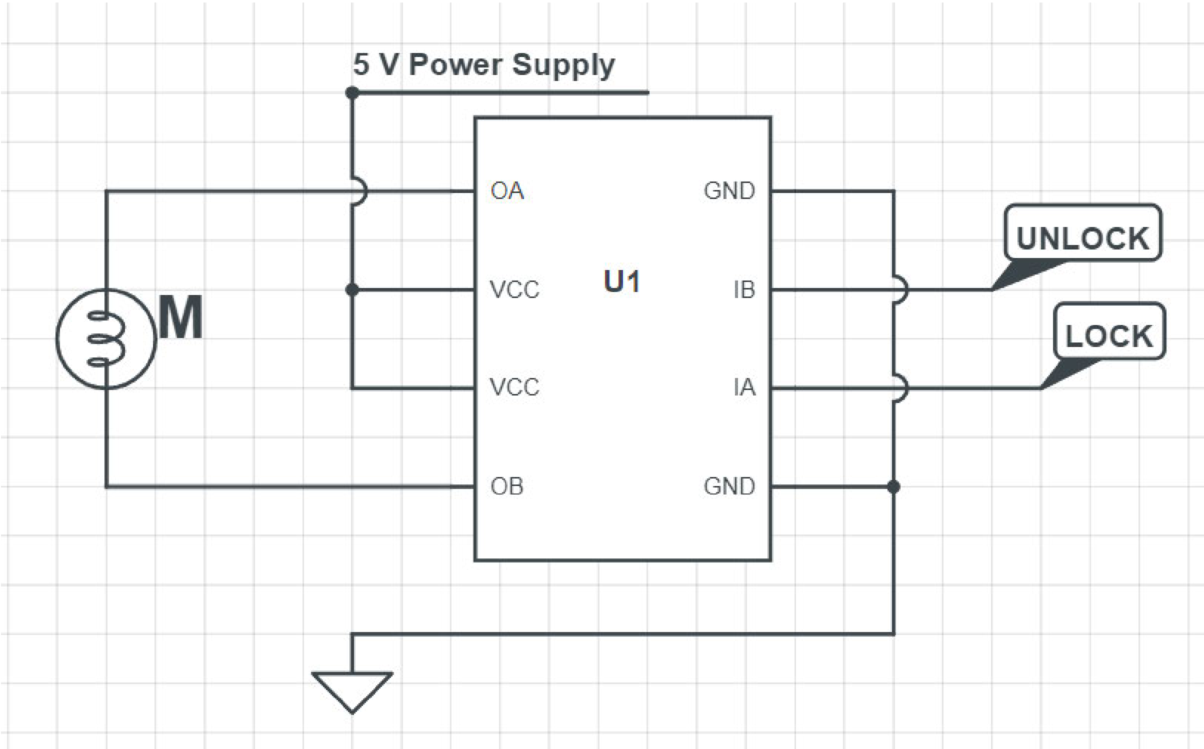


Figure 14 L9110H wiring diagram

To provide control to the motor, the logic levels below must be satisfied.

IA	IB	OA	OB
H	L	H	L
L	H	L	H
L	L	L	L
H	H	L	L

Figure 15 L9110H logic relationships (Components, 2024)

4.4.3 Locking control

A visual inspection of the motor and locking mechanism identified that the lock is engaged and disengaged by the motor rotating clockwise and counterclockwise. There are no switches or toggles in the mechanism to indicate to the existing control circuit that the locking mechanism has travelled the appropriate distance. It must be assumed that to lock or unlock the door, the motor is simply powered on for a short period of time, which is long enough to rotate the mechanism to the lock/unlock position, but not long enough to cause damage to the motor by driving the motor in a stalled condition.

Investigation and testing

An initial test to apply a 100 mS pulse of power was performed, to prove the electrical connection arrangement, and to validate the voltage assumptions of the motor. Pins IA and IB are labelled “Lock” and “Unlock” respectively in the Python script. A Lock door and Unlock door scripts were written, energising the motor controllers forward and reverse signal for a period of 100 mS. Both the lock and unlock scripts have been defined as functions, making repeat usage of the functionality simple by using either lockDoor() or unlockDoor() commands. Utilising functions streamlines the overall script, reducing the risk of typos or errors, and increases overall script readability.

Firstly, the script imported the built-in function to operate the GPIO pins and labelled pins 18 and 19 as the connectors to the motor driver IC. As per the logic table above, the initial state of the signal pins must be LOW, delivering 0 V output. Only the signal to lock or unlock shall go HIGH for a short duration.

```
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)
lock = 18
unlock = 19
GPIO.output(lock,0)
GPIO.output(unlock,0)
```

The script then includes commands to set the output of the lock pin to HIGH, generating a 3 V output to the motor controller IC. The program performs a short stop while the motor rotates, before setting the output to LOW.

```
def lockDoor():
  GPIO.output(lock,1)
```

```
    time.sleep(0.1)
    GPIO.output(lock,0)
```

The motor was observed to energise, rotating the locking mechanism towards its locked state.

The second unlockDoor function was written to reverse the motor, driving the locking mechanism to the unlocked state.

```
def unlockDoor():
    GPIO.output(unlock,1)
    time.sleep(0.1)
    GPIO.output(unlock,0)
```

During the bench lock and unlock function tests, it was observed that the mechanism would not always correctly complete the transition from one state to the other, resulting in a failed lock/unlock transition. The mechanism was sometimes observed to bounce off of the lock position, resulting in the mechanism potentially fouling on the door tab. This behaviour prompted follow-up trials to determine the most reliable lock / unlock control sequence that would ensure reliable and repeatable door performance.

It should be noted that the door mechanism was initially inspected while dismantled on the test bench to facilitate simple tests of the motor. Subsequent tests were performed on the lock after the door system was re-assembled and installed into a wooden panel, to most accurately simulate the final system conditions.

Tests

Setup – A simple script was written to control the door lock and unlock signal; with times the signal is on being easily modified for ease of testing.

```
import time
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)
lock = 18
unlock = 19

GPIO.setup(lock,GPIO.OUT)
GPIO.setup(unlock,GPIO.OUT)
GPIO.output(lock,0)
GPIO.output(unlock,0)

doorWait = 2
motMoving = 0.1

def lockDoor():
GPIO.output(lock,1)
```

```
print("locking")
time.sleep(motMoving)
GPIO.output(lock,0)
print("lock stopped")

def unlockDoor():
GPIO.output(unlock,1)
print("unlocking")
time.sleep(motMoving)
GPIO.output(unlock,0)
print("unlock stopped")


for I in range(0,21,1):
unlockDoor()
time.sleep(doorWait)
lockDoor()
```

This script was modified to meet the requirements of each test.

Test 1 – 100 mS signal

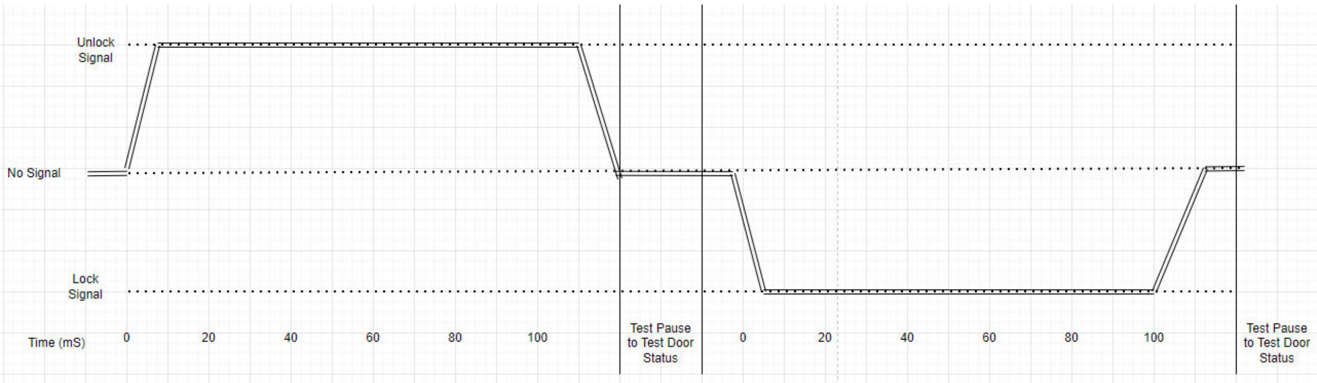


Figure 16 10 mS test signal pattern

As shown above, the first test was a simple on – wait – off signal to drive the locking mechanism to the lock and unlock positions. Test 1 will trial a 100 mS signal time. A pause in the sequence has been included between each transition to allow for a door lock confirmation. It is believed that this pause will not affect the test results.

An unlock/lock sequence will be tested for 20 cycles, with each lock and unlock transition being judged as either successful (✓) or incomplete (×)

Table 10 Door Lock/Unlock test 1 results

1		2		3		4		5		6		7		8		9		10	
Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
11		12		13		14		15		16		17		18		19		20	
Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Test 1 demonstrated repeated smooth operation of the locking mechanism, both in locking and unlocking. No failures were detected

Test 2 – 50 mS signal

Signal shall be reduced to 50 mS to whether shorter pulse cycles may change the locking behaviour.

Table 11 Door Lock/Unlock test 2 results

1		2		3		4		5		6		7		8		9		10	
Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
11		12		13		14		15		16		17		18		19		20	
Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Similarly to test 1, each unlock and lock sequence performed well. No faults detected

Test 3 – 10 mS signal

Signal shall be further reduced to 10 mS.

Table 12 Door Lock/Unlock Test 3 results

1		2		3		4		5		6		7		8		9		10	
Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo	
✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	×	×	✓	✓	✓	✓	✓	✓	✓
11		12		13		14		15		16		17		18		19		20	
Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo		Un / Lo	
✓	✓	×	✓	✓	✓	×	✓	✓	✓	×	×	✓	✓	×	✓	✓	✓	✓	✓

A motor pulse width of 10 mS performed poorly. Unlock / Lock sequence was not consistent. It is also noted that some unlock sequences resulted in the mechanism being lowered enough to unlock, but not enough to completely clear the door hatch.

Observation: During this test it was noted that the noise from the locking mechanism was reduced. A re-test of the locking mechanism at 50 mS confirmed that the locking mechanism sounded harsh. While this is not a definitive test result, it shall be noted that a quiet locking mechanism may be beneficial, both in reducing any potential scaring of the animal(s), and in reducing any mechanical strain on the motor and locking mechanism. Further tests shall now be performed to identify how short a signal can be used before any potential unlock/lock failure occurs.

Table 13 Pause timing test results

Test No.	4	5	6	7
Pause time (mS)	10	20	15	12 *
Test Result	FAIL	PASS	PASS	PASS

*while the locking mechanism did function properly, it was observed that the mechanism was beginning to interfere with the door panel.

Observation– If the mechanism is locking while the door is returning to the locked position, there is a potential that the mechanism may not move fully to the LOCKED position. Given the unpredictable nature of animal interactions with the door, the following lockDoor behaviour modifications shall be made:

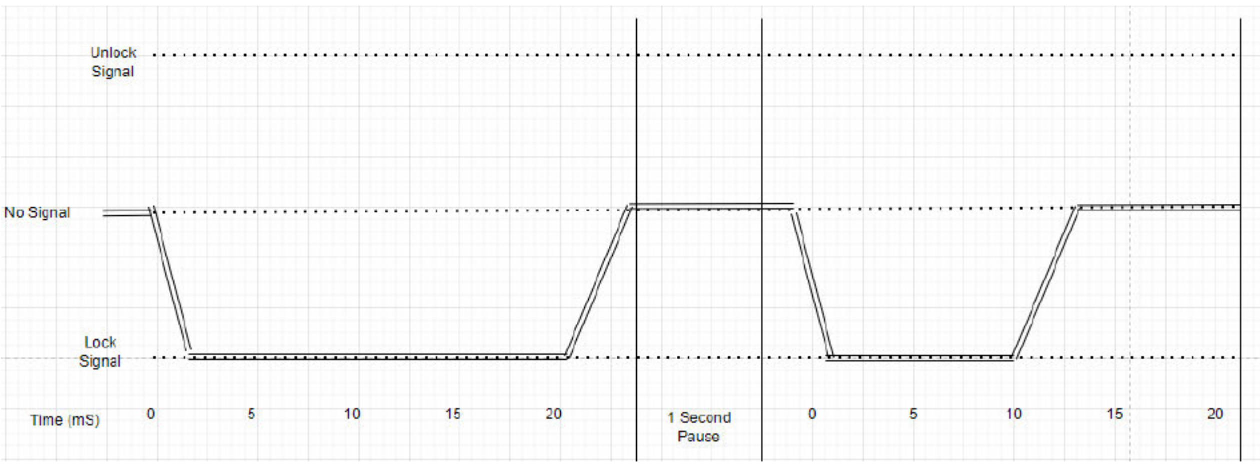


Figure 17 Modified lockDoor behaviour

The initial door lock pulse shall be initiated, at the tested 20 mS duration. Once the initial pulse has been completed, a wait signal for 1 S shall be called to allow for the door to complete it's movement to the closed position. Given that the locking mechanism has already travelled the majority of the movement to the LOCKED position, a shortened second pulse shall be initiated to ensure the lock is fully engaged.

```
def lockDoor():
GPIO.output(lock,1)
time.sleep(motMoving)
GPIO.output(lock,0)
time.sleep (1)
GPIO.output(lock,1)
time.sleep(0.01)
GPIO.output(lock,0)
```

Testing the modification above confirmed that the locking mechanism reliably completed the full movement, however, the harsh locking noise noted earlier returned when the second pulse was called. A second test reducing the second pulse width to 8 mS removed the harsh noise but retained lock reliability and repeatability.

Final control solution

Given the results and observations above, a 15 mS unlock period shall be utilised, in combination with a 15 mS initial pulse / 8 mS secondary pulse pattern. 15 mS was selected as testing demonstrated that the pulse width was reliable, while minimising energy use, mechanism wear and noise. A test of 100 unlock/lock cycles was performed to ensure that the pattern remained reliable. No errors were detected during the test.

4.5 Image Processing

The key criterion for hardware requirements is the image recognition software. This software cannot be supported by remote computing systems and must be stored and ran locally on the supplied computer platform. The program must run at sufficient speed as to provide a rapid and accurate output to the control script to promptly allow or block entrance to the nest-box. A delayed response from the system may block a target species from entering the nest-box while the animal is initially interacting with the box, leading to a poor learning outcome. Insufficient computing power may also lead to inaccurate results from the computations, incorrectly identifying animals, and negatively affecting the performance of the system.

Computer vision is the digital system that processes and analyses data from a camera and interprets the content in such a way that can be managed by a computer program (Sharma, 2021). The three basic components of the system are:

Image acquisition – Camera(s) used to capture the image and deliver the data to the processing computer.

Image processing –Various algorithms and learning models are used on the image.

Image identification – Data collected from the image processing is used to identify the contents of the image.

As outlined, the technical requirements for the image recognition is provided by the parter team. The image recognition team have indicated Ultralytics "YOLOv8" as the selected software being utilised to perform the object detection and recognition.

Online datasets and FAQ responses from the developers (Ultralytics, 2024) of the software have been reviewed, and the following system requirements have been have been identified:

Table 14 - Ultralyticsv8 system requirements

Programming Language	Programming tools	RAM	Disk Space	GPU
Python 3.7 (Min) Python 3.8 (Recommended)	PyTorch 1.7	8 GB	50 GB	NVIDIA GPU with CUDA 11.2

4.6 Camera

The partner team, in collaboration with the Environment and Science team have identified the requirement for a camera with Infra-red capability, that can capture acceptable images in night conditions.

Table 15 - Camera specifications

Part Number	Supply Voltage	Data interface	Compatible systems	Night Vision?
IMX219-160	3.3 V	Camera Serial Interface (CSI), 15 Pin, 1 mm	Jetson Nano* Raspberry Pi*	Yes

*adapter cable may be required to interface with later variants of both Jetson and Raspberry hardware.

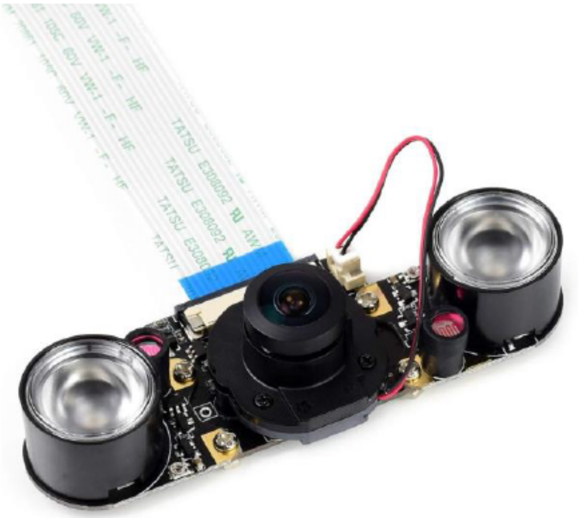


Figure 18 Infra-Red camera (Amazon, 2024)

4.7 Primary Controller

4.7.1 Introduction

A central computing hub is required to perform the image recognition, and control the camera, communication systems, and sensors required to meet the project needs.

The processor selection shall be driven by the system requirements identified in previous chapters. Online Services, Microcontroller boards and single board computers shall be compared to analyse their suitability for the project. Given the many variations of each computing option, it is not practical to perform an exhaustive review on every variant of each option. To simplify the investigation, four products shall be selected to represent each general section of the computing space.

Cloud Computing: Cloud computing shall be represented by Amazon Web Services, a large service provider, boasting an extensive range of products. System performance shall be derived from their online documentation (AWS, 2024).

Microcontroller Board: The Arduino series is a well-known and well-developed Microcontroller board in the hobby, study and tinkering space. The open-sourced platform has various boards available, therefore, both the Arduino Uno and Arduino Mega shall be considered. Hardware specifications shall be derived from their online hardware catalogue (Arduino, 2022).

Single Board Computer: Given the range in computing power, cost and capabilities, this category shall be sub-divided into two. The Raspberry Pi is considered to be a board more suited to learners and tinkerers (Raspberry Pi, 2024), where the Orin Nano, a product developed by NVIDIA, has been specifically designed to provide a compact, high powered computing system (NVIDIA Developer, 2024b).

Table 16 - Main processor requirements

Source	System requirements		AWS	Arduino	Raspberry Pi	Orin Nano
Image Processing	8 GB RAM		Up to 24 TB	8 KB	8 GB	8 GB
	50 GB onboard memory		TB's of memory available	256 kB memory	SD card	SD card
	GPU		High performance GPUs available	N/A	Yes	Yes
	Python v3.8		Python 3.12 available	No – C++ variant required	Yes	Yes
Door controller	GPIO outputs, controllable with scripts		No – a relay system would be required.	Yes	Yes	Yes
Camera	USB or CSI connector input		USB re-direct available. CSI connectivity available through APIs	No CSI connector. USB available, but no camera libraries exist.	Yes	Yes
	Image processing capability		Yes	No	Yes	Yes
Recording and reporting	Onboard writeable memory		Yes	Memory available.	Yes	Yes
	UART communication capable		Communications available through online API's	Yes	Yes	Yes
System design requirements	Not reliant on consistent internet connectivity		No -Must maintain an internet connection	Yes	Yes	Yes
Meets all requirements?			NO	NO	YES	YES

From the investigation above, a Single Board Computer shall be selected to provide the computing capability of the system.

Through consultation with the imaging software team, a Jetson Orin Nano development platform shall be utilised as the central computer for the system. It may be possible to implement this system with the smaller Raspberry Pi system, this may be an avenue for further investigation on a later project.

4.7.2 Jetson Orin Nano

System overview

The Jetson Orin Nano is a compact computer, with a dedicated Graphics Processing Unit (GPU), microUSB memory storage, camera connectively natively available and various USB ports available (NVIDIA Developer, 2024a). While this platform may be more expensive than other developer or hobby robotics kits, the processing power, available peripheral connections, and onboard storage make the development of the computing system for the project far simpler. The dedicated GPU provides the processing power to perform image recognition in real-time within the platform itself, and therefore will not rely on external systems to perform animal recognition. Once the final design has been implemented, a review of the system requirements may be performed to analyse whether a smaller controller / computer may be utilised.

Programming Language

Programming language is the style and structure in which the computing instructions are written and formatted (Hemmendinger, 2024). Each programming language has their own styles, strengths and weaknesses, especially in the field of image recognition. For the purpose of image recognition, it is generally recommended that the programming language Python or C++ be utilised, as they are perform their functions quickly, and are open source – meaning that many image recognition functions and libraries already exist that can be utilised (Sakovich, 2023, Pape, 2023).

Specifications

Hardware:

Table 17 - Jetson Orin Nano Hardware Specifications (NVIDIA Developer, 2024b)

Storage	GPIO	Display	Power			Camera Connectivity
			Connector	V in	Consumption	
SD card slot	40-pin expansion Header	DisplayPort Output	5.5 mm, 2.5m DC Power Jack	9-19 V	7 W, 15 W options	USB, CSI 22 pin, 0.5 mm*

*Camera CSI not natively compatible with the selected camera. Adapter cable required to interface selected camera and Jetson Orin Nano board.

Software:

Table 18 - Jetson Orin Nano Software Specifications

Memory (RAM)	Operating System (OS)	Python	CUDA
8 GB	Jetpack 6.0 DP	3.11.5	12.2.140

4.8 Power Supply Requirements

4.8.1 Introduction

The nest box and electronics system are to be installed in remote settings and must operate continuously as a stand-alone system. Power supply to the computer platform, camera, and any peripherals will need to be supplied by a battery that is capable of meeting the Voltage requirements of the system as well as the battery life needs for the project.

Recharging the system with a solar panel may be possible, however, given the nest box is potentially to be mounted in trees with a dense canopy, a solar panel may not get enough direct sunlight to make the recharging of the battery viable. It is also to be noted that a solar panel in the vicinity of these nest boxes may expose the panel to damage from the animals or become soiled and need cleaning. A remotely located solar panel may be considered, although this will necessitate the use of an electrical cable to be run from the panel to the system, which, if left unprotected, could be damaged by wildlife.

4.8.2 Power consumption

NVIDIA documentation states two power modes can be selected, Low and High modes. Standby or idle power consumption is not well documented.

Voltage input: 9 V

Low power mode: 7 W

High power mode: 15 W

It is not documented whether the controller draws a continuous 7 W of energy when awake, or if this stated figure is the maximum allowable draw during operation. It is also not known what affect (if any) performing calculations or running Python scripts have on energy consumption.

An alternative to idle power consumption would be to have the hardware unpowered during periods of inactivity. NVIDIA support teams quote a 300 mS delay between when 5V is applied to the POWER_EN power before the WAKE is enabled. Bench testing may be required to ascertain whether the device can operate in a timely manner from a cold start when an animal is detected.

Power Consumption Bench Test

An understanding of the Jetson Orin Nanos power consumption while powered on and performing calculations is required to accurately determine the battery capacity needed to supply sufficient power supply over time.

Method:

Power the Jetson Orin Nano utilising available Lithium-Ion batteries (18 V, 5 Ah Makita), while the Jetson continuously runs a Python script. This will be a lab simulation of the expected conditions for the final system. The python script will write the time to a text file every minute until the power supply in the battery is no longer sufficient to provide enough energy to the Jetson to continue operation.

Components:

- 3 x Makita 18 V, 5 Ah batteries. 3 batteries will be tested to prevent a potentially defective or aged battery from skewing the test results. All three batteries are > 10 years old.
- Soldered PerfBoard.
- Jumper wires

Theory:

The Jetson Orin Nano will be set to 7 W power mode. Initial calculations will assume a full 7 W energy consumption at all times @ 5.5 V DC input, therefore an assumed continuous consumption of 1.2 A.

$$Battery = energy\ consumption$$

$$18\ V * 5\ Ah = 5.5\ V * 1.2\ A * x\ h$$

Solving for hours: $h = \frac{90}{6.6}$ = 13.63 hours

If the Jetson Orin does draw a continuous 7 W when running a Python script, the expected battery life is approx. 13 hours.

Bench Test Arrangement

Script:

A short script was written to repeatedly check the time. If the “Seconds” was equal to 0, the script would open the text file “Battery Report”, write the date and time, and pause for one second to avoid duplicate time entries, then continue the loop. The script is to run continuously until battery power fails.

```
$ from datetime import datetime #date and time functionality
$ import time
$ print("battery test routine")
$ now = datetime.now().time()
$ print(now)
$ time.sleep(5)

$ with open('battery_report','a') as f:
$       f.write("Battery Endurance Test  " + 'ʃdate:%d-%m-%Y_%H:%M:%Sʃ'.format(date = datetime.now()))
$       f.write("\n")
$while True:
$       now = datetime.now().time()
$       print(now)
$       if now.second == 0 :
$       with open('battery_report','a') as f:
$       f.write('ʃdate:%d-%m-%Y_%H:%M:%Sʃ'.format(date = datetime.now()))
$       f.write("\n")
$       time.sleep(1)
```

The image below is the display output of the script running. The terminal is the right-hand side window, displaying the time repeatedly called. The left-hand window is the text file, which is updated with the most recent minute.

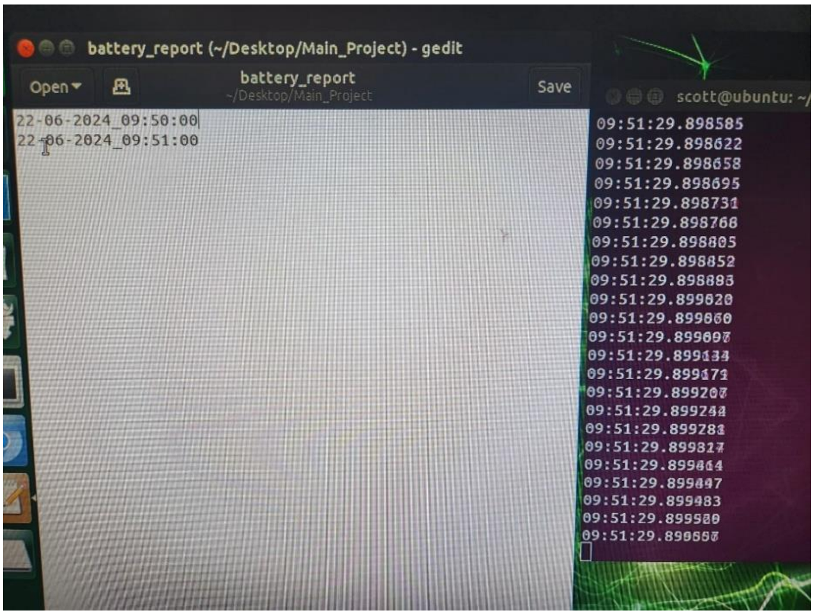


Figure 19 Battery test script terminal output

Bench Setup

The 18 V Makita battery was connected to the input terminals of the Jetson Orin via a small soldered perfboard, used to ensure sound electrical connectivity as shown in the image below. A monitor, keyboard and mouse were initially connected to the Jetson to begin the test script. Once the script was observed to be running, the monitor and peripherals were disconnected in order to remove any potential power draw on the system that did not match the intended field conditions.

The external camera was attached to capture power drain from the camera.

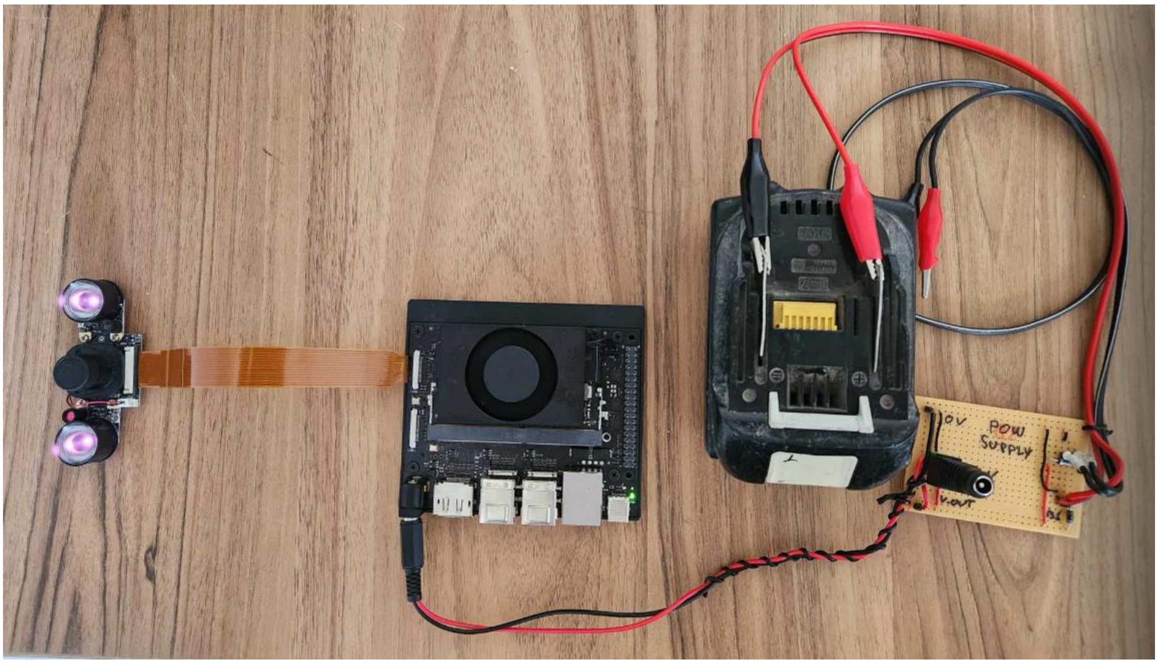


Figure 20 Battery Bench test arrangement

Results:

Table 19 - Jetson Orin Nano power consumption test results

Test	Battery	Start Voltage (V)	End Voltage (V)	Time Start	Time End	Duration
1	Battery 1	19.2	14.25	14.46	02.31	11 h 45 m
3	Battery 2	20.35	0.11	07.15	18.54	11 h 39m
2	Battery 1	20.27	0.1	07.17	16.42	9 h 25 m

Observations:

Battery flat voltage levels oddly vary between approx. 14.46 V and 0.11 V. It is speculated that the Makita batteries have a protection circuit within the battery that prevents the batteries from being damaged from draining too deep of a discharge. There is little information from the manufacturers regarding any potential protection circuit integrated into the batteries. It may be prudent to repeat the power supply test with another type of battery to see if the power duration results are similar.

Conclusion:

Tests suggest that the Jetson Orin Nano consumes consistent wattage, at or slightly above 7 W. This result may be slightly skewed by battery health but is near enough to the calculated estimate of 13 hours to be an acceptable conclusion.

4.9 Battery Supply

As discussed in the previous section, this project shall require battery power to operate in the intended remote locations.

Battery Type

A review of the market availability for large batteries identified Lead-Acid as the first viable option for battery technology to be utilise. While Lead-acid batteries are typically more readily available and often the cheaper option, their performance compared to a Lithium battery of similar size is significantly less. (Schiavone, 2022). The project shall consider Nickel Metal Hydride (NiMH) rechargeable and lithium batteries, which maintain a consistent voltage until they no longer provide adequate current and voltage output. This power output behaviour results in a consistent voltage supply over the period of discharge, compared to a decrease in Voltage provided by alkaline batteries. Lithium batteries are more expensive than lead-acid, but they also last considerably longer and won't fail in cold weather. NiMH batteries will also last longer than lead-acid, but suffer in temperatures above 80 degrees. (Genzel, 2021). While it is not considered likely that the battery shall be exposed to 80 degrees, it is likely that the battery will be exposed to potentially significant temperature swings, and can potentially be exposed to temperatures ranging from below 0 deg C to above 40 deg C (Australian Wildlife Journeys, 2024).

Depth of discharge

Typically, rechargeable batteries are not designed to be fully discharged, i.e. 100 % of their energy carrying capacity should not be used, before recharging. To maintain the required voltage output and maximise the life of the battery, discharging of the battery should be ceased once the battery's energy levels have met is Depth of Discharge (Waag and Sauer, 2009).

Depth of discharge (DoD) can be calculated by comparing the amount of energy used, by the total amount of energy the battery is able to carry.

$$DoD = \frac{removed\ amount\ of\ charge\ (Qd)}{Maximum\ available\ amount\ of\ Charge\ (C)} * 100$$

Battery manufacturers will typically state the recommended DoD for the battery types. Where this information does not exist, it is generally recommended that Lithium batteries should not be discharged below 80% (Xiong et al., 2020).

Battery sizing

To Estimate the size of the battery bank required, the following shall be performed (Schiavone, 2022):

5. Determine Energy Use (in wattHours)
6. Determine number of days the system should operate without recharge (days of autonomy)
7. Review the thermal losses information to determine Temperature compensation factor.
8. Determine depth of discharge allowable of the battery

$$\frac{\text{daily electrical load} * \text{ Days of autonomy} * \text{ Temperature compensation factor}}{Voltage * depth\ of\ discharge}$$

From the previous section, a consistent draw of 7 W is expected, at a power supply of 5.5 V

Energy consumption in 24 hours

$\frac{W}{V} * hours = Amp.Hours$ Amp.Hours * 1000 = milliamp Hours

$$\frac{7}{5.5} * 24 = 30\ Ah\ ;\ \ 30000\ mAh$$

If no recharging system exists, battery capacity to operate the Jetson Nano for 1 day, 1 week, 1 month:

Table 20 Energy requirements

Power usage (Ah)	1 day (Ah)	1 week (Ah)	1 month (Ah)
	30	210	900

A 12 V 340 Ah battery weighs approximately 36 kg, and is 522 mm x 268 mm x 220 mm (Battery World, 2024). This battery could supply the system for approximately 20 days. If two batteries were installed in parallel, this can provide power to the system for 40 days.

Given the Jetson Orin Nanos broad range of input voltage (9 V to 19 V), the research crew have a broad range of batteries available to them to balance their battery life with cost.

4.10 Battery Level Monitoring

In order to alert the team of low battery, and to reduce the potential for battery damage by surpassing the recommended Depth of Discharge, battery monitoring and control circuitry shall be required.

4.10.1Theory

The voltage level within a lithium battery does not discharge in a linear manner from full charge to 0 V. The batteries are able to maintain a reasonably consistent voltage throughout its discharge cycle. Therefore, careful consideration of the discharge characteristics shall be required to calculate the voltage level of a Lithium battery nearing full discharge.

Makita 18 V Li-Ion batteries comprise of 10 x 3.6 V, 18650 cells, arranged in 2 parallel sets of 5 (Wikimedia, 2022). Total voltage output is equal to Cell Voltage x number of cells in series.

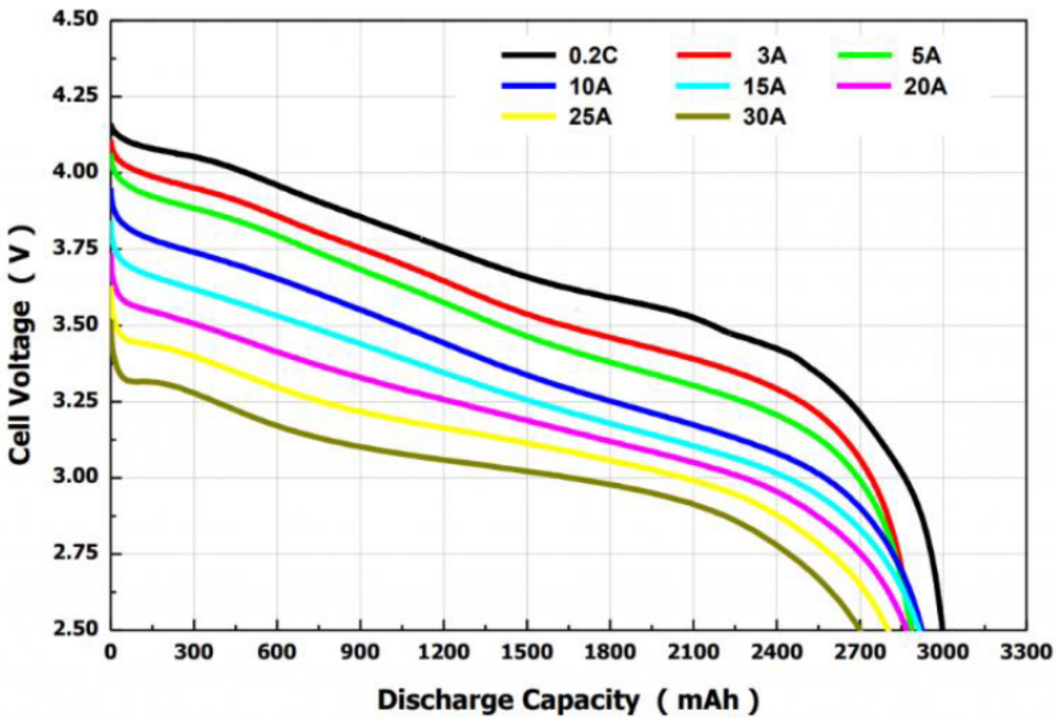


Figure 21 Discharge Curve for 18650 cell (MakerMax, 2024)

Assuming a discharge rate of 1 A:

Table 21 - Lithium-ion battery % charge voltages

Cell Arrangement / Discharge	0% (full)	50%	80%	100% (empty)
Single Cell	4.2 V	3.5 V	3.4 V	2.5 V
5 cells in series	21 V	17.5	17	12.5

4.10.2 Voltage level detector

To detect the battery voltage, a voltage window detector arrangement shall be established, utilising op-amps and fixed reference voltage(s). A “50%” battery supply warning shall be established, as well as a “Low Battery” alarm.

As discussed in section 4.4, the motor controller shall require a 5 V supply to operate. The 5 V supply shall also be utilised to provide the consistent reference voltage required to monitor the battery voltage levels.

Utilising 10 kΩ resistors, implement a voltage divider to establish 2 fixed voltages. Note, high resistor values have been selected to reduce current drain on the overall system.

R1 = R2 = R3 = 10 kΩ

First offtake:

$$V_{out_{50}\%} = V_{in} * \frac{R1}{R1 + R2 + R3}$$

$$V_{out_{50}\%} = 5 * \frac{10}{30}$$

$$V_{out_{50}\%} = 3.33\ V$$

Second offtake:

$$V_{out\_Low} = V_{in} * \frac{R1 + R2}{R1 + R2 + R3}$$

$$V_{out\_Low} = 5 * \frac{20}{30}$$

$$V_{out\_Low} = 1.667\ V$$

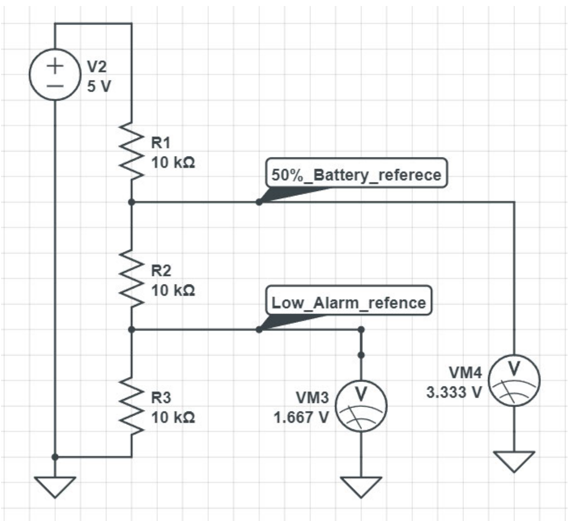


Figure 22 Battery monitor level arrangement and simulation

As above, 50% battery = 17.5 V. Low battery shall be arbitrarily set to 17.2 V.

Table 22 – Battery 50% and Low alarm calculations

Charge and voltage levels	Voltage divider calculations
50% charge; V_in = 17.5 V V_out = 3.33 V	$$3.33 = 17.5 * \frac{R2}{R1 + R2}$$ $$R1 + R2 = \frac{17.5 * R2}{3.33}$$ $$R1 + R2 = 5.255 * R2$$ $$R1 = 4.255 * R2$$ Assume R1 = 10 kΩ R2 = 10 kΩ / 4.255 = 2350 kΩ
Low battery; V_in = 17.2 V V_out = 1.667 V	$$1.667 = 17.2 * \frac{R2}{R1 + R2}$$ $$R1 + R2 = \frac{17.2 * R2}{1.667}$$ $$R1 + R2 = 10.317 * R2$$ $$R1 = 9.317 * R2$$ Assume R1 = 10 kΩ R2 = 10 kΩ / 9.317 = 1073 kΩ

Given the exact resistance values, it may be prudent to utilise potentiometers for R2 and adjust the exact value to achieve the desired voltage divider.

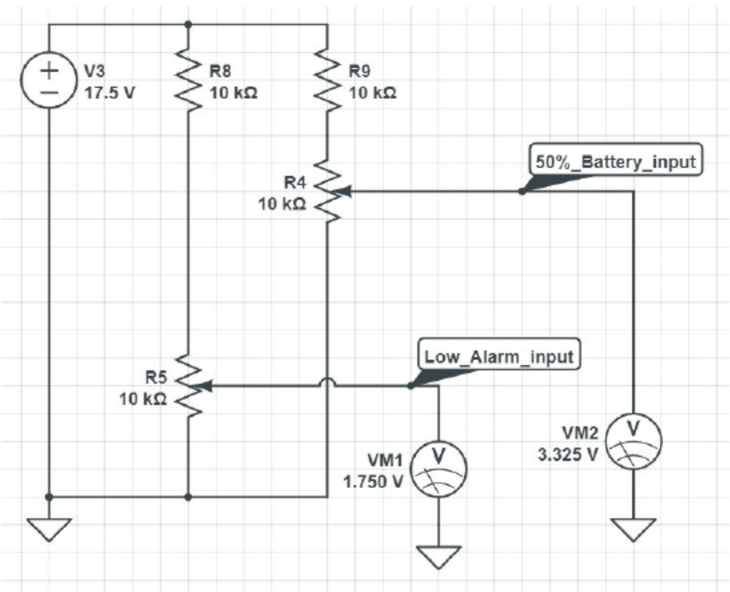


Figure 23 Voltage divider circuit

Proposed circuit

The proposed circuit is an amalgamating of the voltage divider circuit and the two potentiometers shown above, a op amp is to be included to function as a comparators, detecting whether the voltage levels in the voltage divider has been met. The output of the two op amps will supply a HIGH signal to the controller when the battery power exceeds the setpoints.

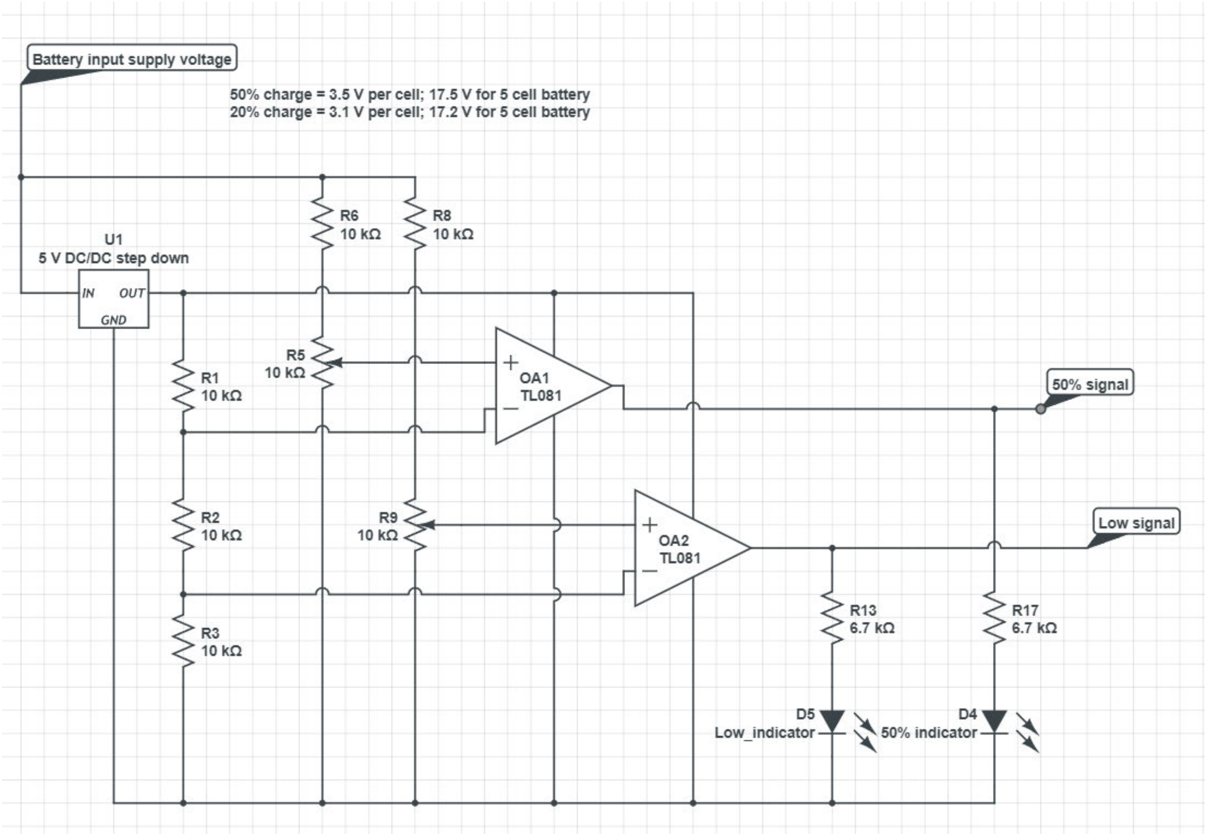


Figure 24 Voltage level detector circuit

4.11 Peripherals

4.11.1 PIR signal booster

While the primary method of animal detection shall be the object detection and image recognition software, a backup system shall be implemented as a system check and potentially utilised as a power saving wake signal. Passive InfraRed (PIR) sensors shall be installed near to the door entrance, detecting the presence of any infrared signals. Infrared light is typically emitted by animals and people, therefore any movement of an animal near to the nest door shall be detected by the PIR sensor.

GPIO input HIGH logic levels for the Jetson Orin Nano is 3.3V. There are conflicting datasheets that indicate whether the PIR delivers 3 V or 3.3V to the sensor output pin. A control circuit shall be implemented to ensure that the inconsistent output voltage from the PIR is boosted to provide consistent triggering of the Jetson input GPIO signal.

As a 5 V power supply is available, a high side switching arrangement shall be implemented, utilising a PNP and NPN transistor arrangement. This arrangement can also power two indicator LED's to help the user identify if a PIR is faulty.

Proposed circuit

Circuit shown below indicates a full 5 V output given an input signal voltage of 3 V. When signal is applied, approx. 5 mA is utilised in the system, drawing minimal current from the overall supply.

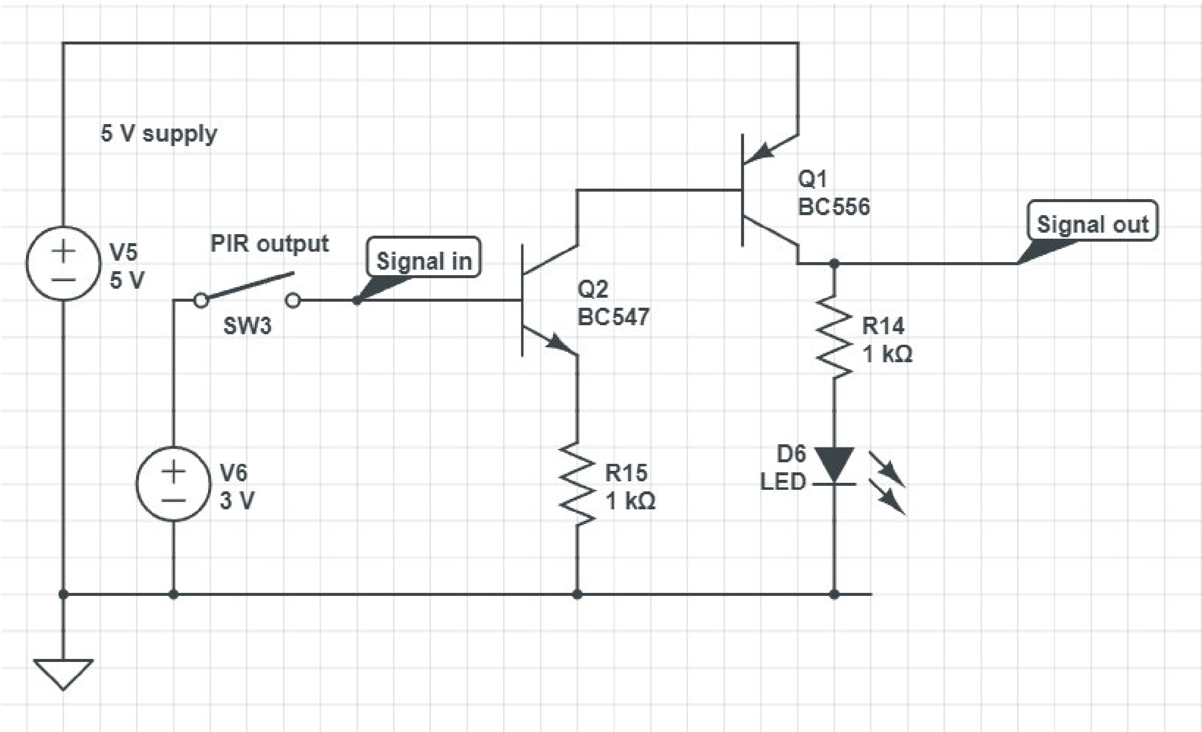


Figure 25 Proposed High-side switch

Design Calculations

PNP transistor = Q1; NPN transistor = Q2

Table 23 Signal Boost transistor calculations

Known values	Calculations
h_{fe} (DC current Gain): 110 min V_{be} = 0.7 V Vce = 0.09 V typ. 0.25 V max V_drop_led = 1.6 V approx	Q2: $V_e = power\ supply - V_{be}$ $V_e = 3 - 0.7$ $V_e = 2.3\ V$ $I_e = \frac{V_e}{R15}$ $I_e = \frac{2.3}{1000}$ $I_e = 2.3\ mA$ $I_{ce} = I_{be} * hfe\ \therefore\ I_{be} = \frac{I_{ce}}{hfe}$ $I_{be} = \frac{2.3\ mA}{110}$ $I_{be} = 20.9\ \mu A$ Q1: $Q1\_I_{eb} = Q2\_I_{ce}$ $I_{eb} = 2.3mA$ $I_e = \frac{V_e - V_{ec} - V_{led}}{R14}$ $I_e = \frac{5 - 0.25 - 1.6}{1000}$ $I_e = 3.15mA$ $I_{eb} *$ hfe = 2.3 mA * 100 = 230 mA Current I_e < I_{eb} * hfe, transistor Q1 is fully saturated, therefore $V_c = V_e - V_{ec}$ $V_c = 5 - 0.25$ $V_c = 4.75\ V$

Confirm Calculations

Utilising circuit simulation software CircuiLab, high side switching calculations can be confirmed. A signal voltage generates an output signal of 4.75 V min.

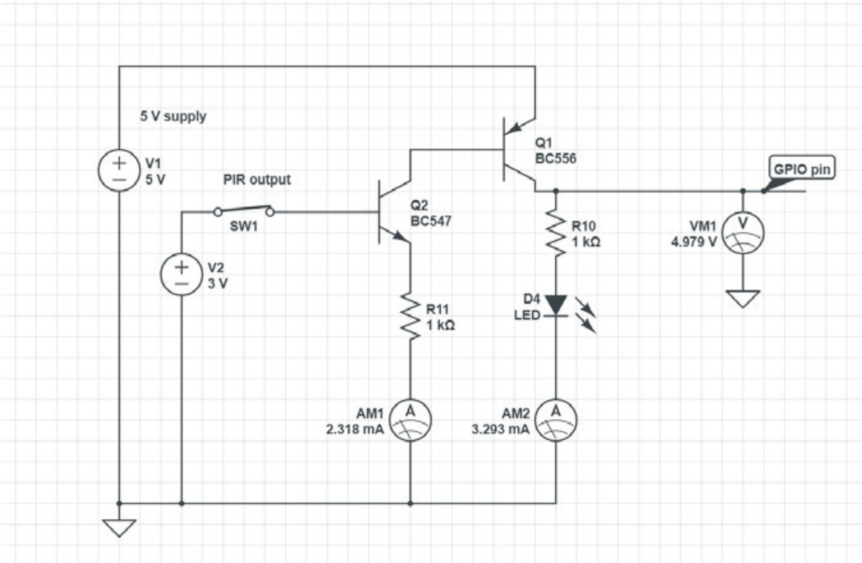


Figure 26 PIR control circuit - 3 V signal applied, 5 V output

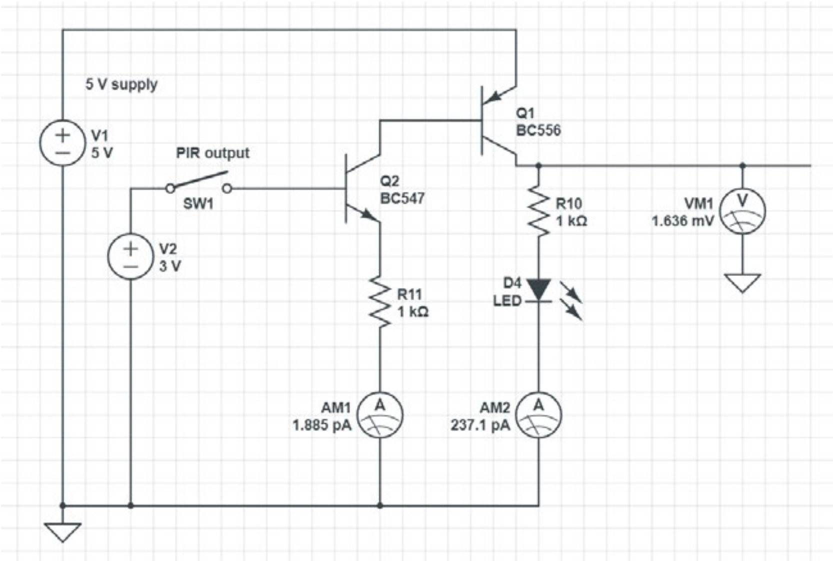


Figure 27 PIR control circuit - no input signal. No output voltage

4.11.2 LED indicator

A tri-colour LED shall be included in the design to aid both the system development and testing, as well as indicate system status for the research team. The LED shall be able to output Red, Green, Blue, or any combination of those colours.

A common Cathode Tri-colour LED shall be utilised.

Circuit design

LED circuit shall be powered by a 5 V supply, independent of the Orin Nanos GPIO pins.

LED's shall be controlled by 3 x NPN transistors. A resistor on the common Cathode was installed to reduce the overall brightness and power demand of the circuit.

3 to 5 V input signal is able to switch the NPN transistor, supplying voltage to the colour.

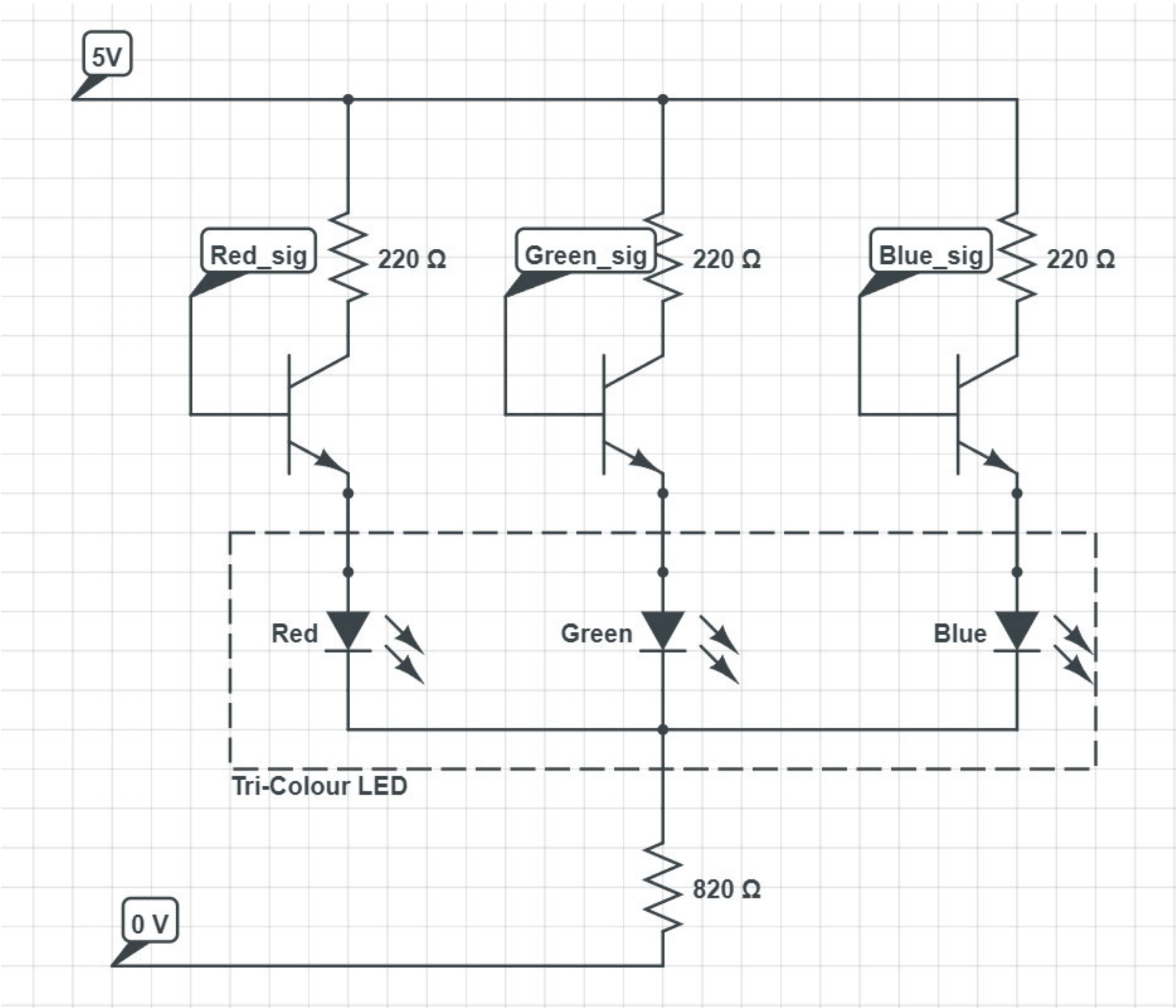


Figure 28 Tri-colour LED circuit

4.12 Object Detection Scripts

The partner team have supplied a package of Python scripts and accompanying instructions to install and launch the object detection models and animal identification tools. These scripts have been supplied to be used and modified as required, with full permissions by the authors of the scripts - Mr Arien Westerman and Dr Derek Long. A copy of the original primary script can be found in Appendix B.

Script investigation

A review of the script has identified the following key components that must be understood and modified in order to integrate the object detection model:

$ Detection_Classes = ["person", "cell phone"] # Which things do you want to activate
$ detection

Detection classes identify what objects will trigger actions further in the script. Classes can be added to the above list to flag them as an item of interest.

$ Detection_Class_Minimum_Confidence = { # Confidence level required for activation
$ "person": 0.9,
$ "cell phone": 0.6
$ }

Minimum confidences can be adjusted to determine how tight of a fit the computing match needs to be for a trigger to be actioned. This value can be refined, to suit the systems capability to accurately detect a specific species. If the system is struggling to confidently identify a specific species, it may be allowable to relax this confidence level, provided that either it is unlikely that other similar animals will interact with system, or that there are no similar animals that may pose a threat if mis-identified and allowed access.

$ # Define Classes
$ classNames = ["

This section of script contains the labels of items that the script can identify. It is suspected that future revisions of the script will contain unique tags for the specific animals that the researchers require to identify.

$ if flag_classTrigger:
$ if Detection:
$ # Put activate code here
$ print("Trigger" , ",".join(class_Detection_Info))
$ else:
$ print("No trigger")

This if statement triggers when an object in the detection class is identified, and the identification meets the minimum confidence threshold. This trigger shall be the primary function that provides the animal control.

Supporting Scripts

In addition to the primary Python script examined above, several other small scripts were provided as part of the object identification package. While these files are required for the object detection to operate fully, their individual function(s) do not affect the system design, and are all unmodified. It is sufficed to say that the final system shall ensure that all files shall be installed on the primary controller in the file structure as outlined by the partner team. A review of files did not identify any additional design requirements above those identified in earlier chapters.

4.13 Record keeping

4.13.1 Onboard records

A function shall be created to write a message to a file saved on the primary controller. This file can be used both as a master record or as a backup if the Satellite communications system fails.

The first script below opens, or creates, a master file named “onboard_report”, and writes the time and date, as well as whatever string it is passed when the function is called. This script keeps adding to the original file.

The second script breaks up the recording into individual files based on the date, i.e. a file will contain all recordings in that particular day. This hopefully prevents a large loss of data if the primary file is corrupted or lost.

Script

```
with open('onboard_report','a') as f:
    f.write('ʃdate:%d-%m-%Y_%H:%M:%Sʃ'.format(date = datetime.datetime.now()) +"; " + f"ʃanimalʃ detected")
    f.write("\n")

with open('ʃdate:%d_%m_%Yʃ'.format(date = datetime.datetime.now()),'a') as daily:
    daily.write('ʃdate:%d-%m-%Y_%H:%M:%Sʃ'.format(date = datetime.datetime.now()) +"; " + f"ʃanimalʃ detected")
    daily.write("\n")
```

4.13.2 Wireless Communications

As discussed in the Concept Design, satellite communications enable the nest box system to be deployed in remote locations where there may not be adequate cellular connectivity to obtain a reliable data connectivity. Through consultation with the requesting team, it was decided that a “Myriota Satellite Communications” system shall be deployed for the project. The Myriota device is a standalone device, with an independent battery supply and a set of GPIO pins. The service provider indicate that a pair of AA batteries is sufficient to supply power to the system for years (Myriota, 2024), and therefore shall not be included as a requirement of the power supply system design.

System architecture

To communicate between the master controller and the Myriota device, Low Energy Universal Asynchronous Receiver/Transmitter (LEUART) messaging protocol shall be utilised to send a short message packet between the two devices. LEUART connection using the onboard UART protocol and a 2-wire connection utilising the GPIO pins on both the primary controller and the Myriota device.

The primary controller shall initialise the LEUART connection with the Myriota and send the message. The Myriota device is then responsible for sending the messages to the satellite communication system. The satellite receivers pass the data packets through to the service server, where the information is received and processed by their web-based portal.

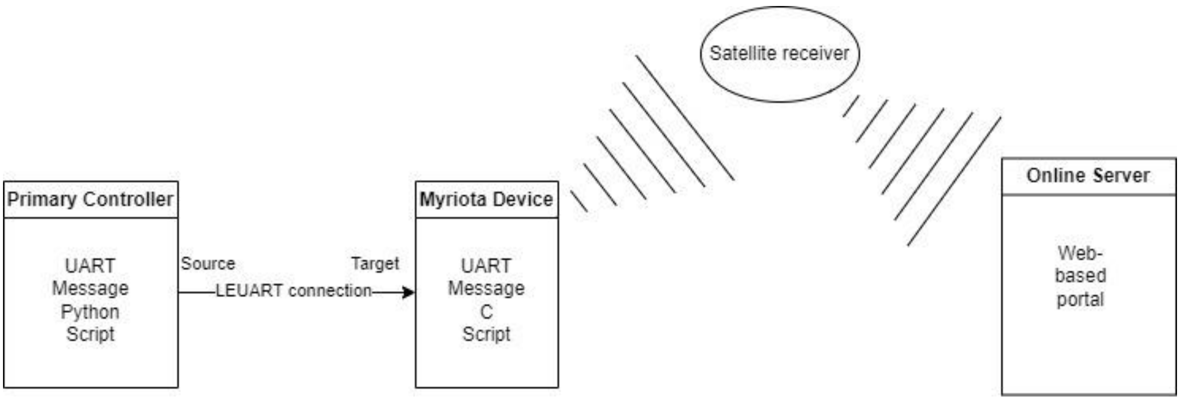


Figure 29 UART message process

Master controller

A Python script on the primary controller shall be called as a function. This script shall establish the Once the connection is established, the script shall create a message using whatever information is passed to the function via a string variable. The Python LEUART script can be reviewed in Appendix C.

Myriota device

Myriota device has a script, that is written in C, which is saved on its internal memory. This script handles the incoming messages from the LEUART coming in via the GPIO pins, formats the message, and queues the message for transmission. The Myriota device waits for a satellite connection to be established and sends as many messages as it can during the window of connectivity. Once satellite connectivity is lost, the Myriota device continues to receive and store messages until it is able to make a new connection. At present, there are approx. 3 to 4 windows of opportunity for message upload over the east coast of Australia each day. The Myriota script can be reviewed in Appendix D.

Myriota message

The Myriota's message is comprised of up to 40 hex characters (0 to 9 plus a to f) to the online Device Manager platform. The online platform manages the hardware registration and message collection and is able to direct the message onto a secondary data lake. The raw Hex file is communicated to an online service called "Tago.IO", where a script intercepts the raw data and transformed to form a message.

Hex to Message

Each two characters from the Hex string form a pair, which is translated into 20 positions inside a buffer object (position 0 to 19). Each Hex pair is translated into a decimal value. These decimal values can then be queried by the script, which can then be either loaded directly into variables, or have further calculations performed to acquire the desired value. If the message sent to Tago.IO does not match the expected format by the script, the message will not be accepted.

Example Hex to Decimal translation: 0000e01a2eeb80ae9c5257047c664a2c00000000

Table 24 HEX value calculations

Myriota total character limit:	40		Example Hex message: 0000e01a2eeb80ae9c5257047c664a2c0000							
Buffer Position	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
HEX	00	00	e0	1a	2e	eb	80	ae	9c	52
Decimal value	0	0	224	26	46	234	128	172	156	82
Buffer Position	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**
HEX	57	4	7c	66	4a	2c	00	00	00	00
Decimal value	87	4	124	102	74	44	10	0	0	0

4.14 Primary Script

It was initially planned that a primary script would be written to perform the basic functions required by the system, and a request to the object detection script would be made to check for any animals being identified by the camera. Bench tests of the object detection scripts identified that the startup time was too great to be effective for animal identification and access control, i.e. if the target species approached the door, the delay in computing may be such that the animal could have already tried and failed to access the door before the script finalised and the door unlocked.

To overcome this, the object detection script shall be modified to act as the primary script, with supplemental controls (locks, lights, reporting etc) being written in secondary partner script, to be called in as functions when required.

Orin Nano GPIO Pinout

The GPIO pins on the Orin Nano shall be utilised to provide Input and Output signals for the various functions, as well as the communications connection. The excerpt of the Jetson Orin nano Datasheet shown below shall be utilised to identify which pins on the GPIO 40-pin header are suitable to be utilised.

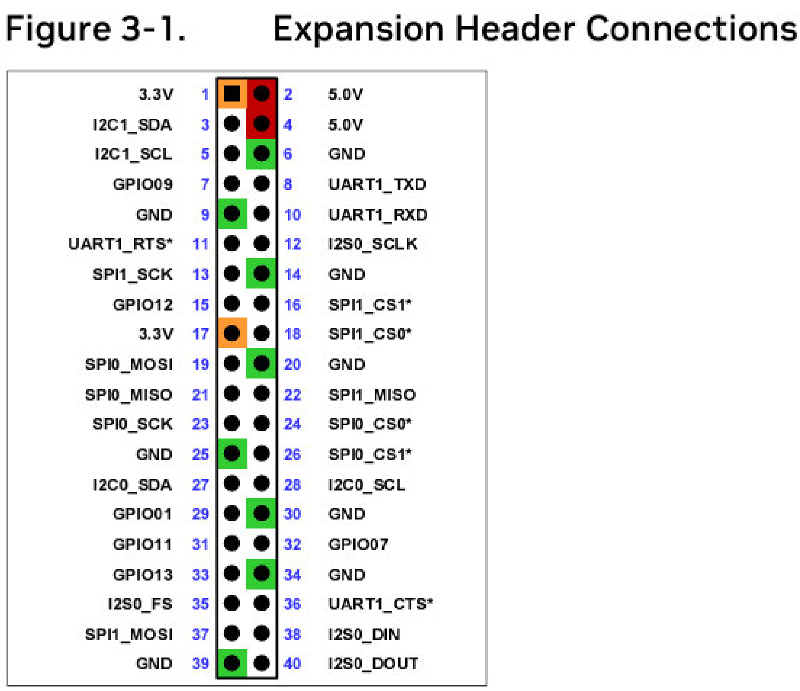


Figure 30 Jeston Orin Nano GPIO pinout (NVIDIA Developer, 2024b)

UART:

Pins 8 and 10 are the only two pins designated for UART receive (RXD) and transmit (TXD) communications and so must be utilised as such.

Signal inputs and outputs

The datasheet notes:

> *"All the interface signal pins (I2S, I2C, SPI, UART, and AU clock) can also be configured as GPIOs."*

It is preferred to utilise the pins designated as GPIO only first, before re-configuring other pins, in the event that those functions be utilised at a later date.

Pins labelled as GPIO only are: 7, 15, 29, 31, 32 and 33.

Care shall be taken not to use any pin identified as 0 V (9, 14, 20, 25, 30, 34, 39) or those listed as power supplies (1, 2, 4, 17).

Pinout

From the designs outlined in the previous chapters, the following Inputs and outputs are required:

Motor control: 2 pins, Motor + and Motor – output

PIR sensor: 1 pin, Sensor Input

LED monitor: 3 pins, LED outputs

Battery Monitor: 4 pins, Battery 1 and battery 2 high and low signal input.

Table 25 - Orin Nano pin designation

Source	Motor Controller		Peripheral Board				Battery Monitor			
Function	Mot (+)	Mot (-)	PIR Sensor Input	LED out (R)	LED out (G)	LED out (B)	Batt1_high	Batt1_low	Batt2_high	Batt2_low
GPIO pin	7	15	29	31	32	33	19	21	23	24

Main Program

The main script shall perform as follows:

1. Initialise the object detection, as provided by the partner team
2. Initialise all GPIO pins, variables, and settings to perform the required processes
3. Main Loop:
    a. Run the object detection
    b. Check to see if an object has been identified
        i. If yes, move to step c.
        ii. If no, move to step e.
    c. Check to see if the animal meets the minimum confidence requirement.
        i. If yes, move to step d.
        ii. If no, move to step e.
    d. If minimum confidence has been met, check to see if the detected animal is allowed.
        i. If yes, call the unlock function, then call the record keeping function, passing the animal details to record function.
        ii. If no, call the record keeping function, passing the animal details to the record function.
    e. Check to see if the PIR sensor has detected any movement.
        i. If yes, call the record keeping function, passing a PIR detection result
        ii. If no, move to step f.
    f. Check to see if the 6 hourly check-in timer has been met
        i. If yes, call the record keeping function, passing timer-checking result
        ii. If no, end loop, return to start.

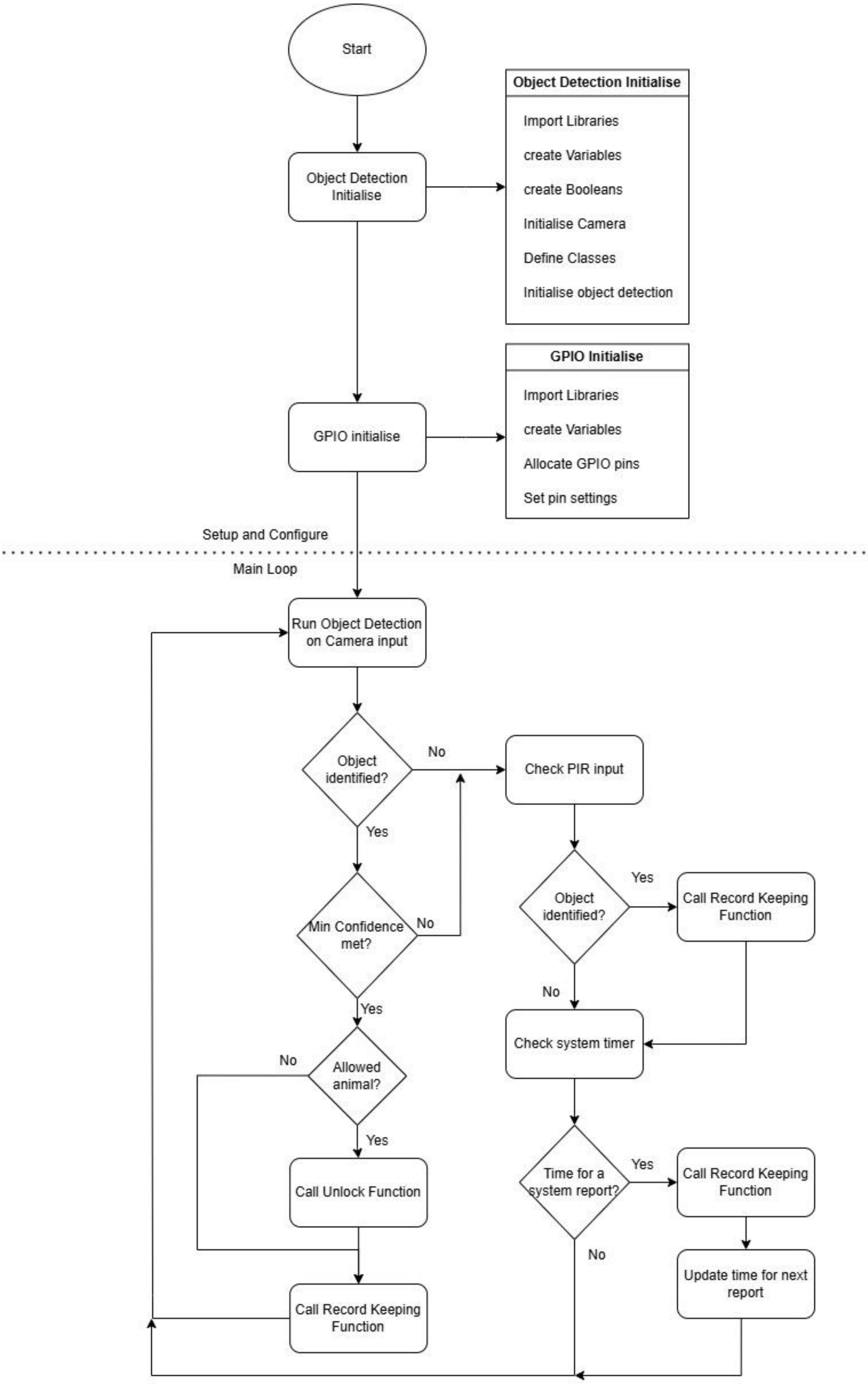


Figure 31 Main Script Flow Diagram

Define Functions

Locking and Unlocking

As per Section 4.4.2, the lock and unlock functions shall be defined as follows:

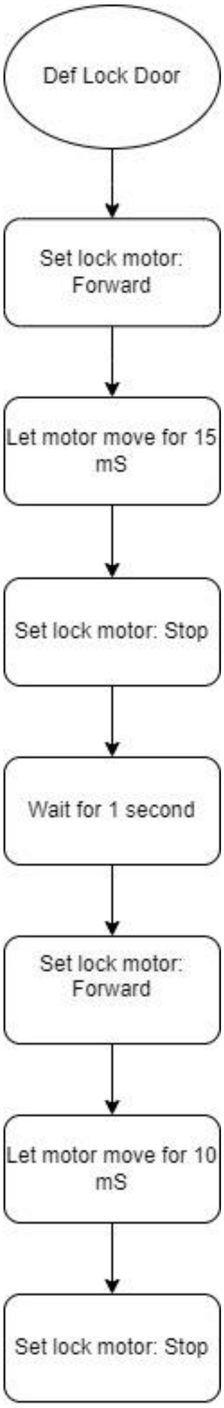


Figure 32 Lock script sequence

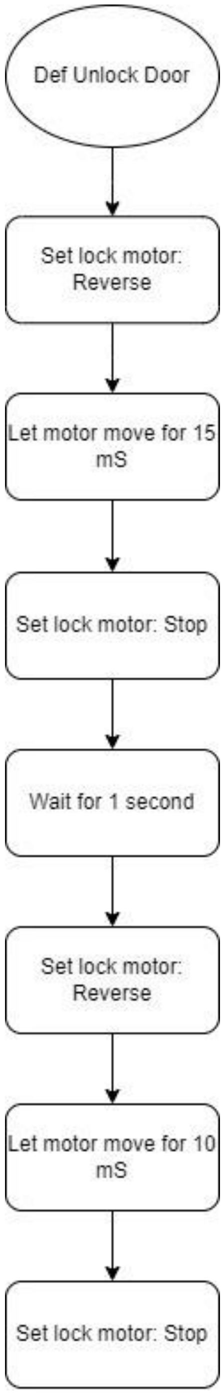


Figure 33 Unlock script sequence

Script

Table 26 - Lock and Unlock functions

Function	Script
Lock the door	def lockDoor(): GPIO.output(lock,GPIO.HIGH) time.sleep(0.015) GPIO.output(lock,GPIO.LOW) time.sleep (1) GPIO.output(lock,GPIO.HIGH) time.sleep(0.01) GPIO.output(lock,GPIO.LOW)
Unlock the door	def lockDoor(): GPIO.output(lunock,GPIO.HIGH) time.sleep(0.015) GPIO.output(unlock,GPIO.LOW) time.sleep (1) GPIO.output(unlock,GPIO.HIGH) time.sleep(0.01) GPIO.output(unlock,GPIO.LOW)

5 Construction and Assembly

5.1 Nest Box and Door

The nest box has been assembled as instructed by the supplier.

The pet door has pulled apart to expose the wiring of the motor and electronic board. The electronics controlling the motor have been discarded. The 2-pair cable to the motor have been extended. The pet door has been fitted to the front of the nest box door, ensuring that the side intended for “inside” is installed to be internal to the nest box. PIR sensor is fitted to the top of the pet door entrance. Wiring to the PIR are installed, with sufficient length to reach the side and back of the nest box.

Figure 34 Nest box front

Figure 35 Nest box internal

5.2 Power Supply Board

A board has been created to manage the parallel battery connections, provide the 18 V DC output to the Jetson Orin Nano, and provide 18 V / 0 V male pins for the easy connection to the 5 V DC/DC stepdown board.

Should the user wish to power the device with a reliable ongoing power supply, the board has incorporated a female 5 mm x 2.5 mm Power Jack to receive an 18 V supply, typically supplied by a transformer, plugged into mains power.

Male pins shall be installed before the diodes to measure individual battery charge.

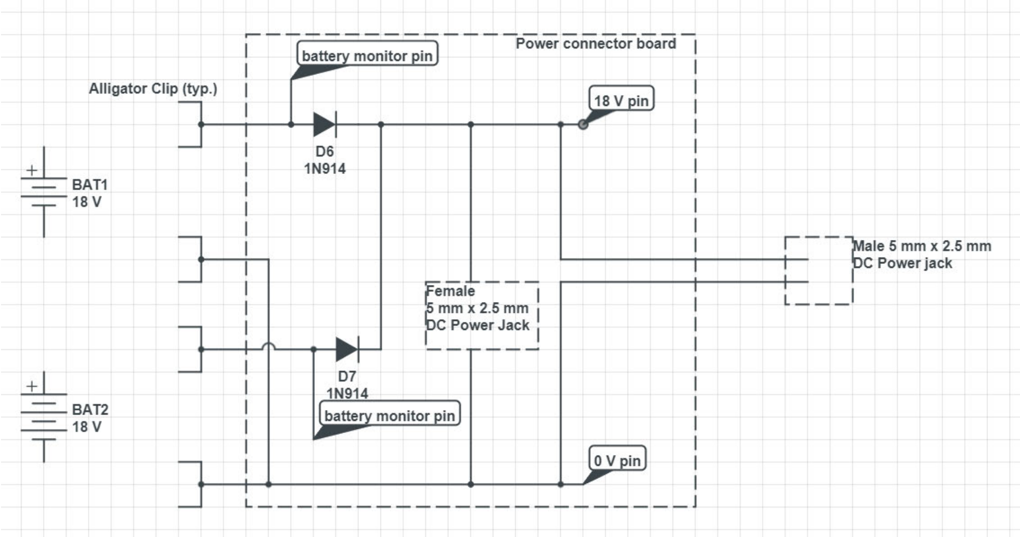


Figure 36 Power supply circuit design

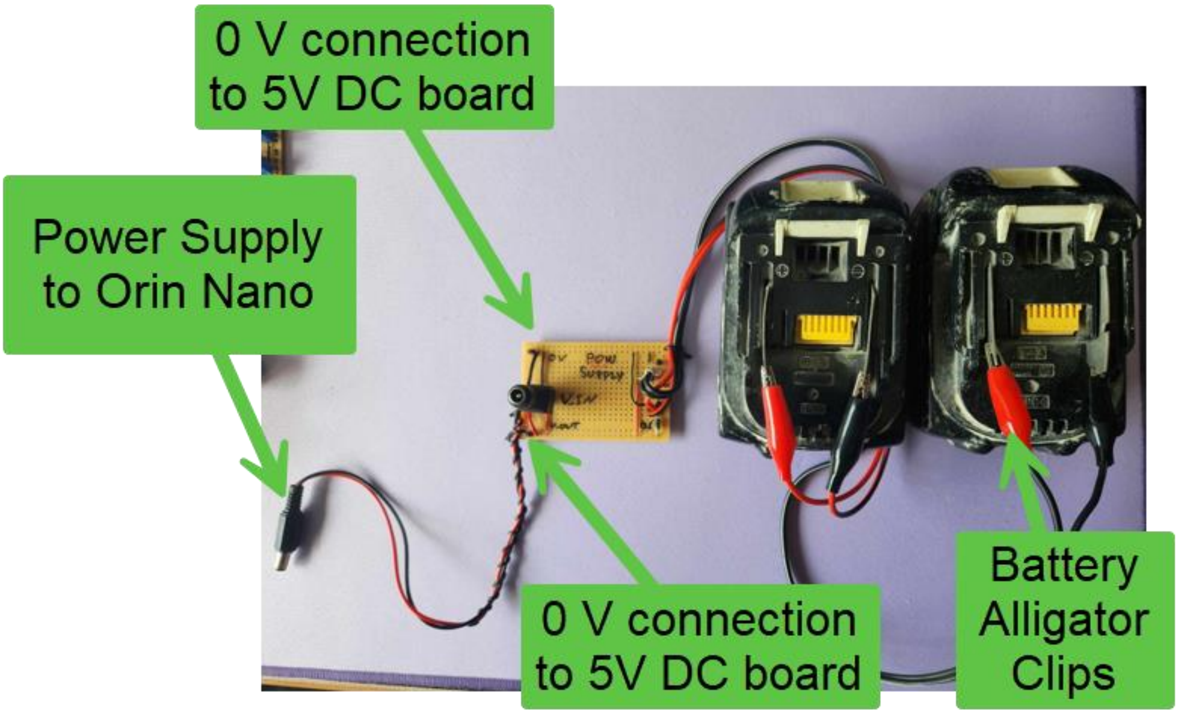


Figure 37 Power Supply circuit and parallel battery supply

5.3 5V Power Management

As power management is provided by a bespoke circuit, 5 V power supply available on the Orin shall not be utilised. Pin 6 shall be used to connect a common ground to the 5 V power supply board, as this ensures a common ground reference across all sub-systems.

To supply the motor, PIR sensor, LED's and battery monitoring boards, a 5 V DC supply board has been created. This board receives ~19 V from the Power supply circuit, and steps down to 5 V DC. 6 pairs of 5 V / 0 V male pins are available. As with the power supply, a female DC jack is available to supply a steady 5 V DC supply if required for bench testing. An indicator LED is installed to indicate when 5 V supply is available.

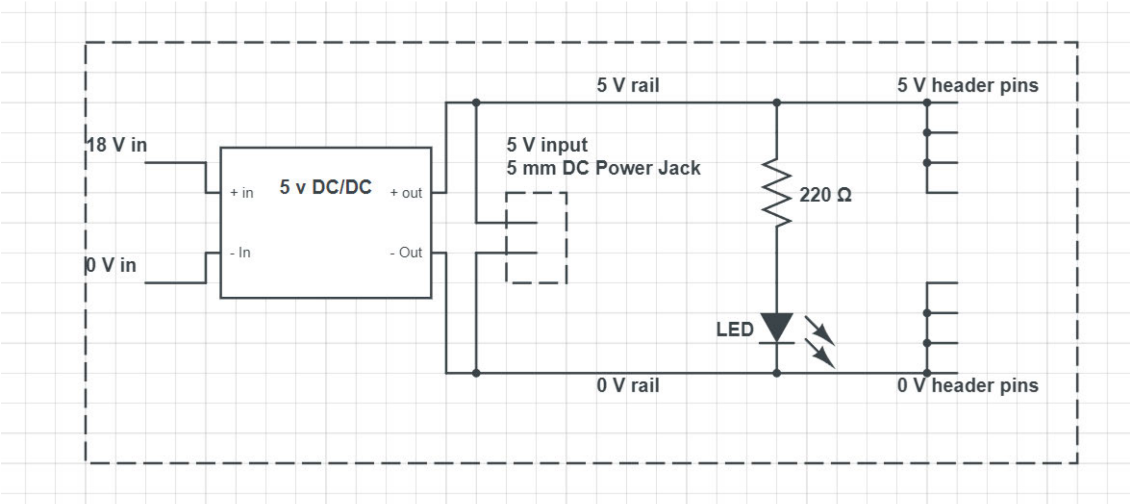


Figure 38 5 V power management circuit design

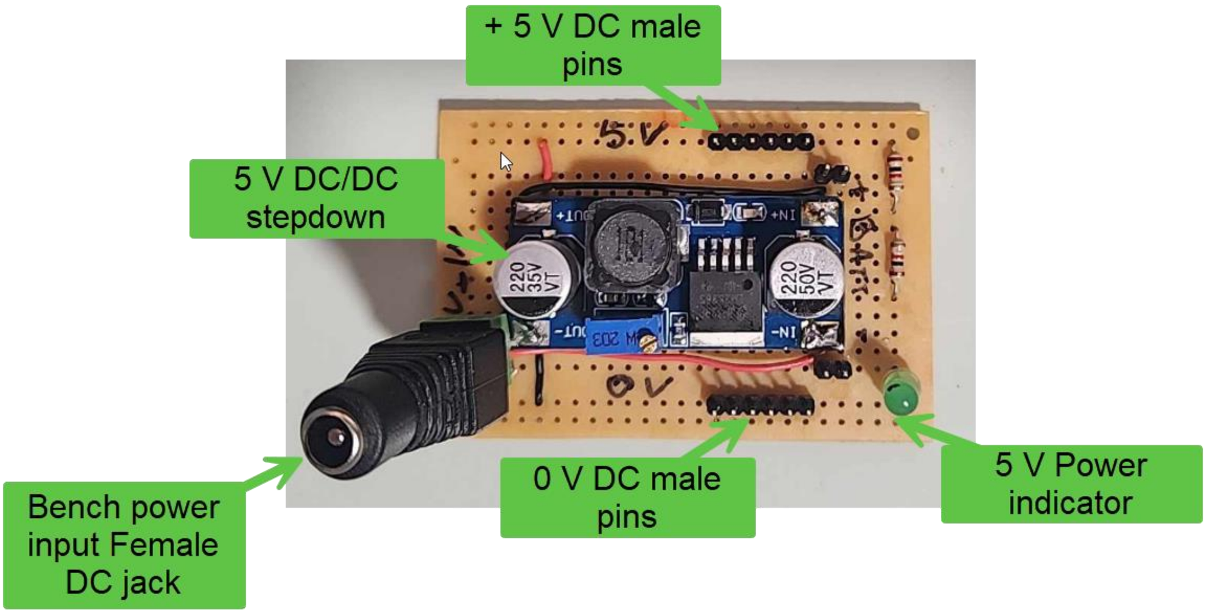


Figure 39 5V Power management board.

5.4 Battery Monitoring Circuit

As discussed in Section 4.10, a battery monitor board shall be created. An LM324 Quad Op Amp has been selected to provide two sets of Op-Amps that are used as comparators.

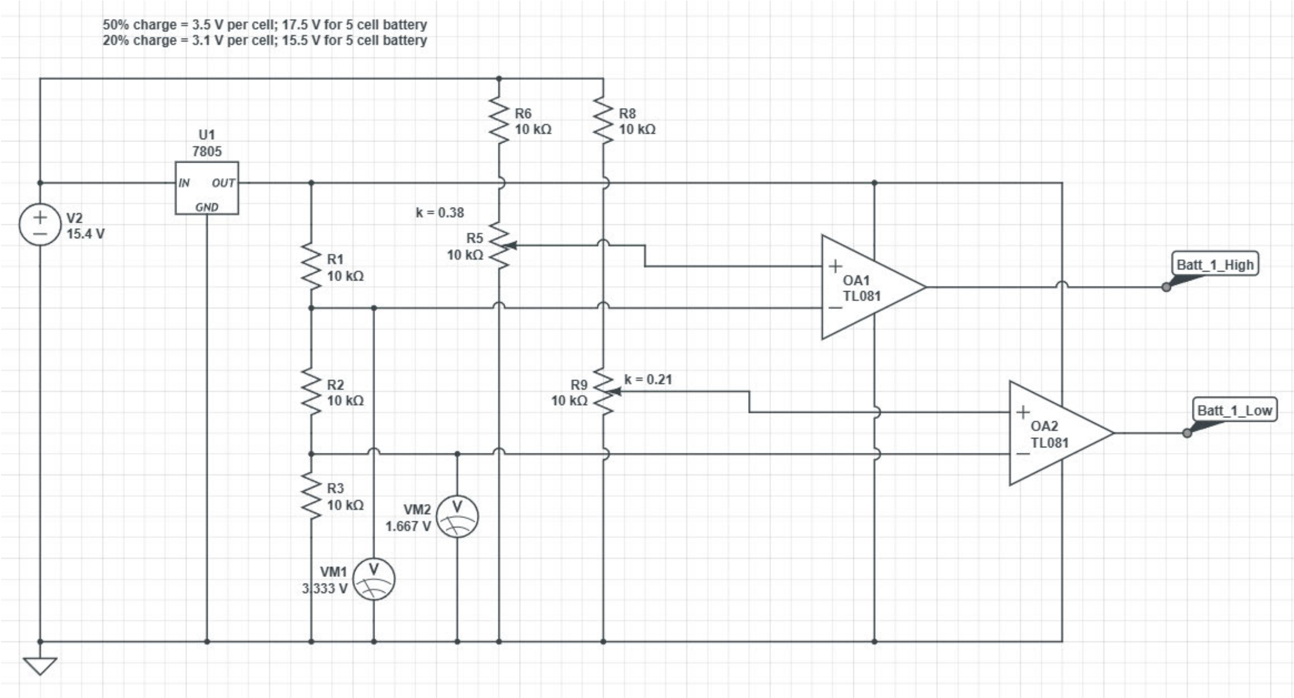


Figure 40 Battery Monitor Circuit Design

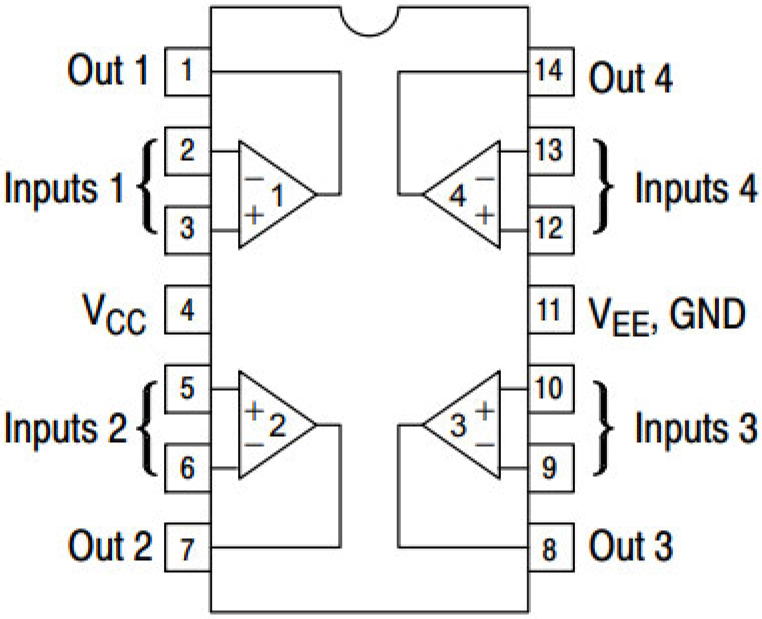


Figure 41 LM324 pinout

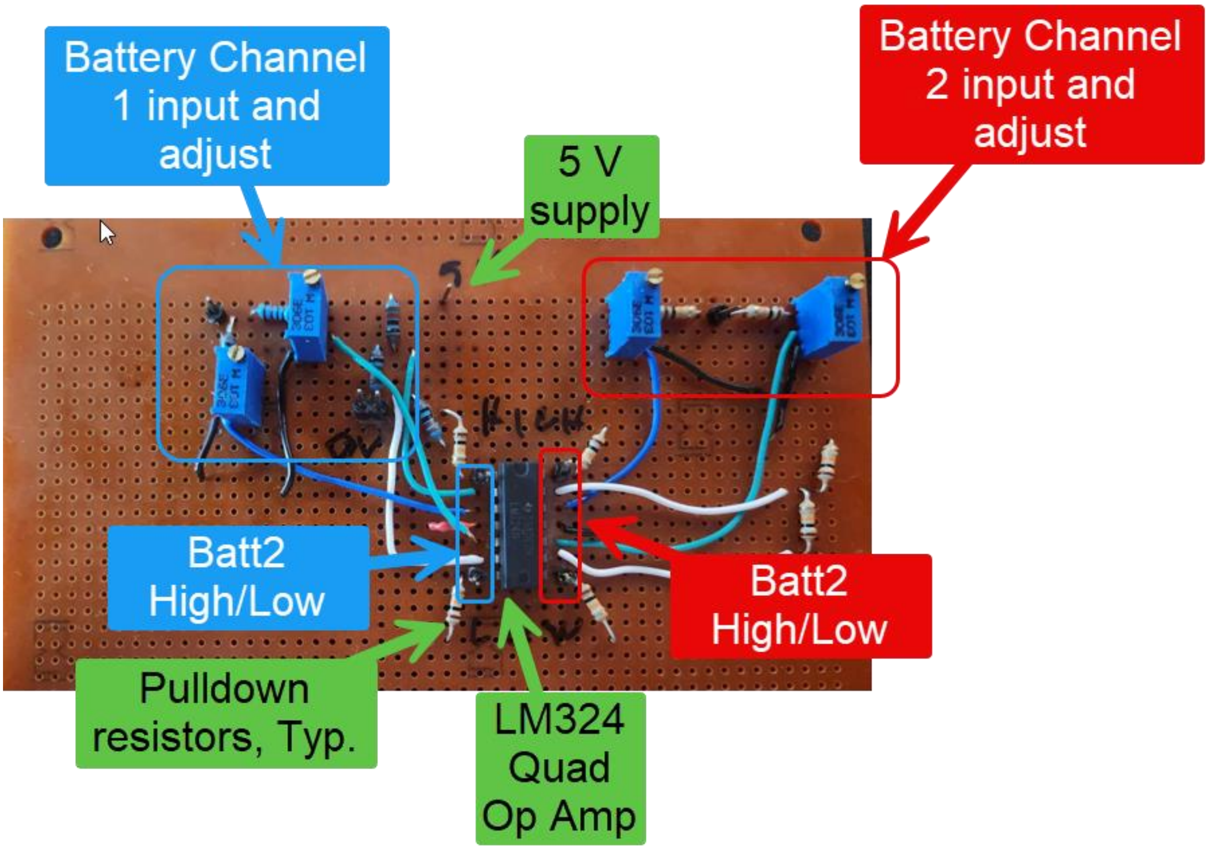


Figure 42 2 Channel Battery Supply Circuit

Calibration

Utilising a potentiometer arrangement, adjust the battery input sensor to the required 17.5 V (simulating 50% charge). Adjust the trim potentiometer until the comparator is just at the on/off voltage, where a slight adjustment up or down changes the state of the comparator output. Repeat for 15.5 V (20% battery simulation) to calibrate the second comparator.

Repeat this process to calibrate the second channel of the voltage level detector.

5.5 PIR and Indicator LED’s

The PIR and LED circuits have been consolidated into a single board. A 5 V input supplies both circuits. Input male pins provide the required connections to the Orin Nano.

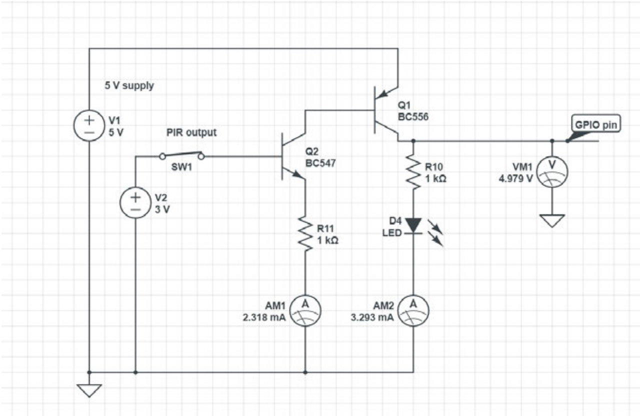


Figure 43 PIR signal booster design

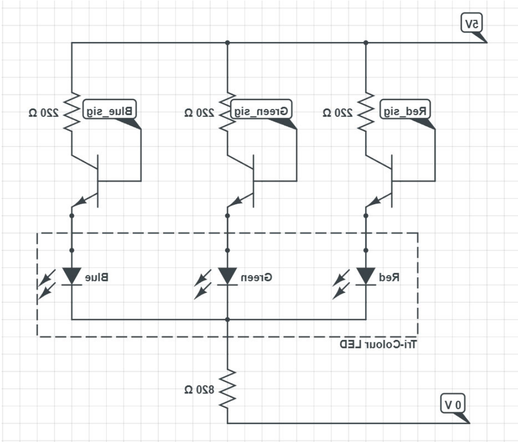


Figure 44 Multi-colour LED design

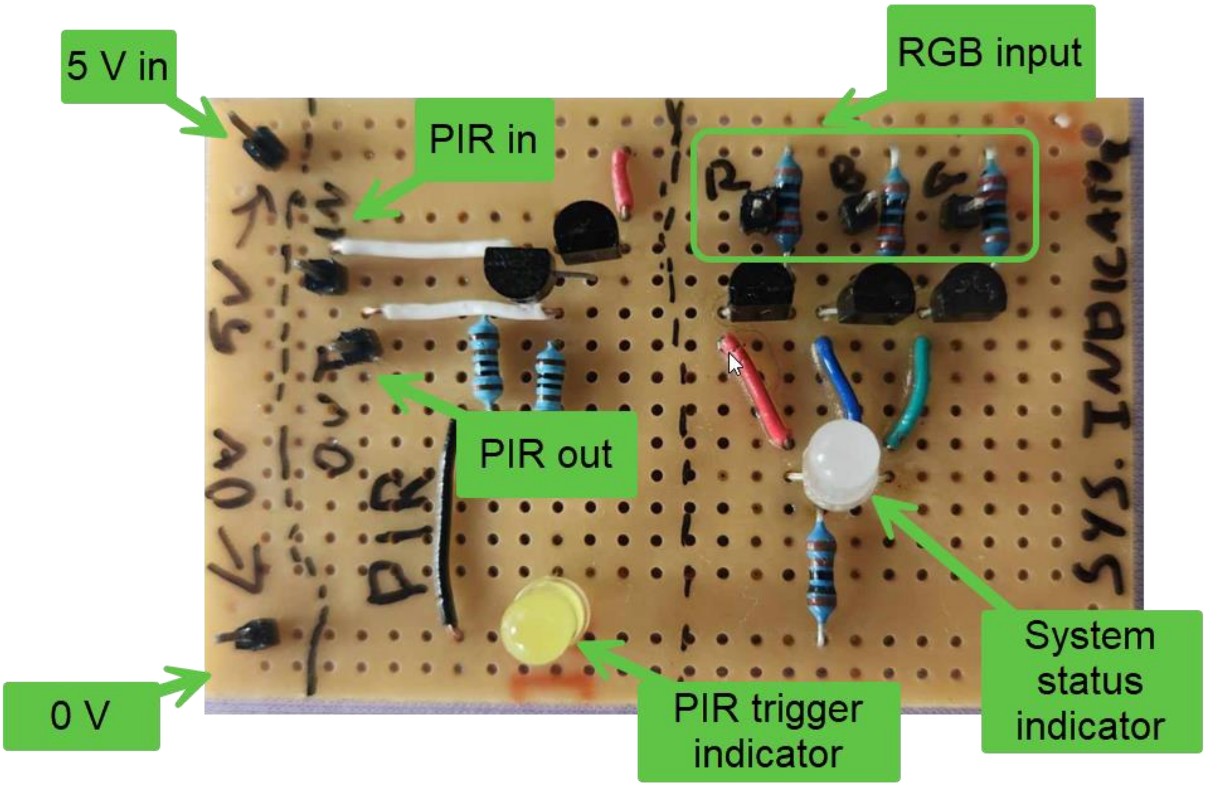


Figure 45 Combined PIR and LED circuit

5.6 Motor Controller Board

As per section 4.4, a motor controller board is required to safely operate the DC motor. The two Vcc input pins have been bridged, requiring only a single 5 V input cable. Similarly, the GND pins have also been bridged, with a single 0 V connection required. The motor coil is to be connected two the pins indicated on the board.

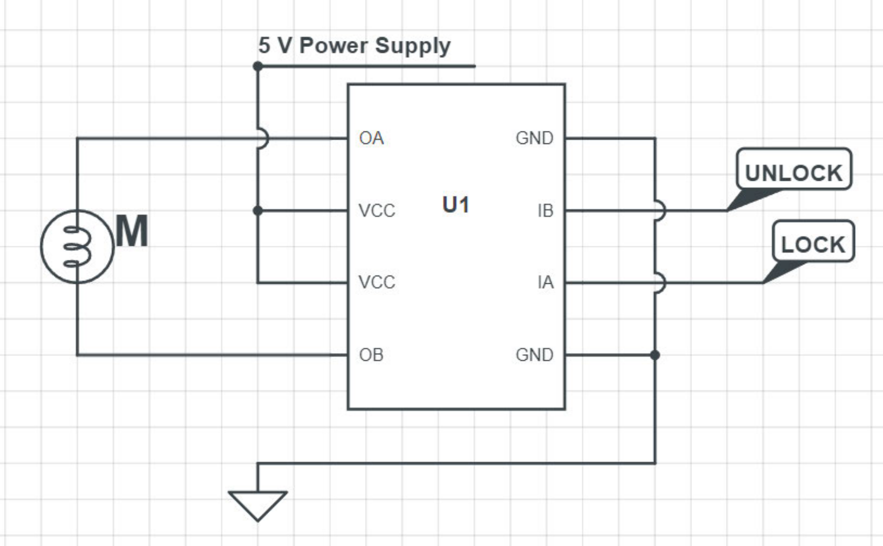


Figure 46 Motor Controller circuit Layout

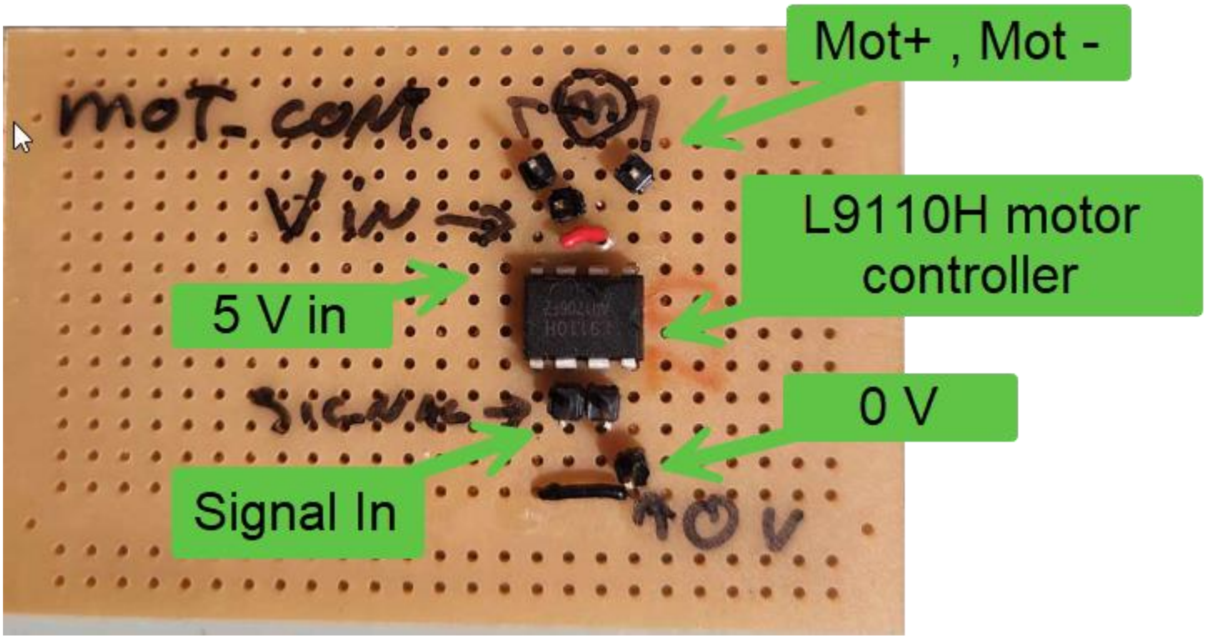


Figure 47 Motor controller circuit

5.7 Assembly

5.7.1 Power supply - Wiring Diagram

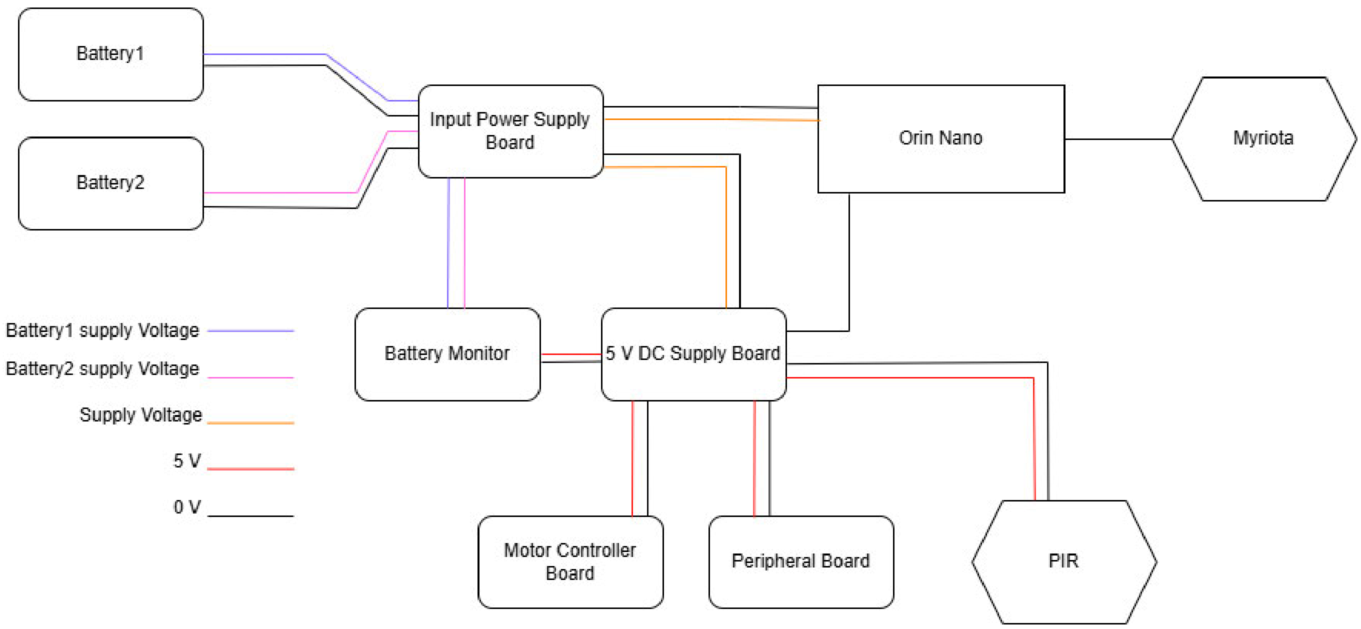


Figure 48 Power supply wiring diagram

5.7.2 Signal connections – Wiring Diagram

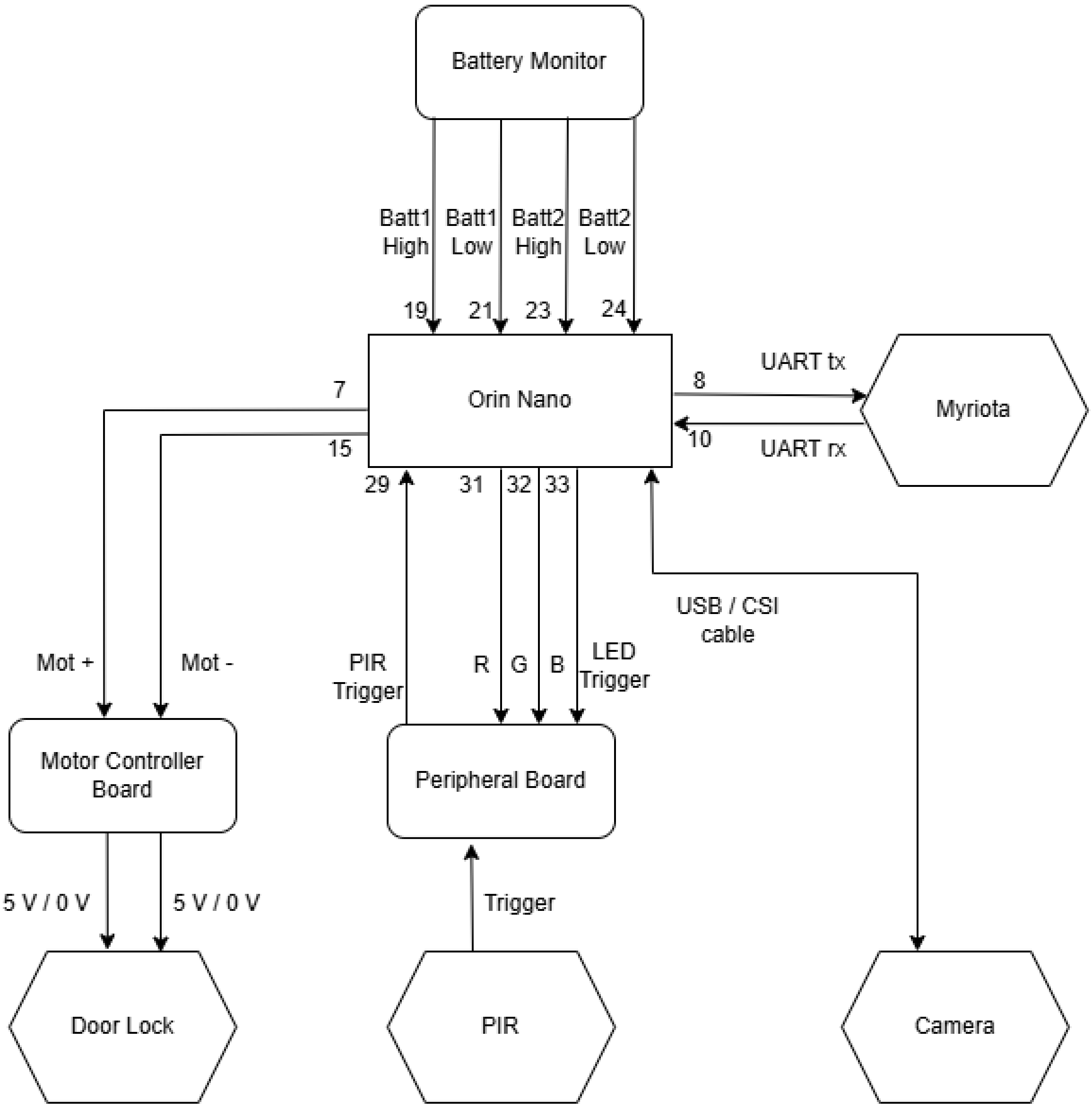


Figure 49 Data connections Wiring Diagram

6 Test and Commissioning

6.1 Functionality Testing

The table below outlines the various tests on each system. Each test details the actions performed, the pass criteria and the result of the test(s).

Table 27 - System Tests

System	Category	Test	Pass Criteria	Result
Door	Safety	Inspect for obstructions, cabling. Test door operation in all stages of the program, and when powered down.	Door freely operates during ingress and egress, and is free of any obstructions, including cabling and sensors	PASS
		Test to determine whether door is always able to be operated regardless of system status	Egress is permitted	PASS
	Functionality	Test door consistently locks and unlocks, as performed in chapter 4.4.3	Door consistently locks firmly when required	PASS
		Test door consistently locks and unlocks, as performed in chapter 4.4.3	Door consistently unlocks completely when required	PASS
		Door reacts quickly when a permitted animal is detected	Door unlocks within 1 second of detection	PASS
Power Supply Circuit	Electrical arrangement	Connect battery 1 only, inspect the power status of the entire system.	Battery 1 connection alone adequately powers the system	PASS
		Connect battery 2 only, inspect the power status of the entire system.	Battery 2 connection alone adequately powers the system	PASS

		Connect both batteries	Battery 1 and 2 connected together adequately powers the system.	PASS
		With both batteries connected, and the system energised and operating, disconnect and reconnect battery 1	System stable when battery 1 is “Hot Swapped”	PASS[1]
		With both batteries connected, and the system energised and operating, disconnect and reconnect battery 2	System power supply stable when battery 2 is “Hot Swapped”	PASS[1]
		With battery 1, battery 2 , then both batteries connected, inspect voltage at DC jack output for Orin	19 V available at the output DC jack	PASS
		With battery 1 connected, inspect voltage at level sensor pin for Batt1	Battery 1 Voltage available at Male Pin output	PASS
		With battery 2 connected, inspect voltage at level sensor pin for Batt2	Battery 2 Voltage available at Male Pin output	PASS
		Inspect sensor pin for Batt 2 when battery 1 only is connected. Repeat test for Battery 2 and Pin1	No interconnectivity for Male pin outputs	PASS
		With no batteries connected, supply 19 V DC to the female DC socket on the power supply board.	Input Female DC socket energises the system	PASS
	Electrical safety	Connect both batteries. Power supply must not exceed the voltage of a single battery (Parallel, not series connected)	Output voltage when both batteries connected is equal to ~ 19 V	PASS
		With the system energised, inspect voltage levels on the battery clips (while they are not connected to a battery)	No voltage detected on battery leads when system is energised from any other source	PASS
	Electrical arrangement	Inspect voltage available across the 5 V DC male pins.	5 V available on output pins	PASS

5 V DC Supply board		With no voltage supplied from the battery system, connect a 5 V DC supply to the female DC jack on the 5 V supply board.	Input Female DC socket supplies 5 V to output pins	PASS
	Indicator	Inspect LED indicator lamp on the 5 V DC board when energised / not energised	LED illuminates when 5 V is available on output pins	PASS
Peripheral board	PIR	Inspect voltage available on the output pin when PIR input is LOW	Output pin = 0 V when input signal is LOW	PASS
		Inspect voltage available on the output pin when PIR input is HIGH	Output pin = 5 V when input signal is HIGH	PASS
	LED	Inspect Indicator LED when system is powered down, or when the system is energised, but not yet running the program.	No LED output when all inputs are LOW	PASS
		Inspect Indicator LED when door is transitioning to the “LOCK” state	RED LED illuminates only when Red input is HIGH	PASS
		Inspect Indicator LED when door is transitioning to the “UNLOCK” state	GREEN LED illuminates only when Green input is HIGH	PASS
		Inspect Indicator LED when the systems is monitoring.	BLUE LED illuminates only when Blue input is HIGH	PASS
Battery Monitor	Functionality	Inspect the system report over a day of use running on battery supply.	Battery 1 HIGH level triggers appropriately	PASS
			Battery 1 LOW level triggers appropriately	PASS
			Battery 2 HIGH level triggers appropriately	PASS
			Battery 2 LOW level triggers appropriately	PASS
		Adjust the set level of the battery trigger, observe the system report for change	Trigger voltage levels adjustable	PASS
Motor controller board	Functionality	Inspect door lock motor while system is not performing a Lock or Unlock function	Motor not energised when both inputs = LOW	PASS
		Inspect door lock motor while system is performing a Lock function	Motor actuates locking mechanism to the LOCK position when LOCK input is energised	PASS

		Inspect door lock motor while system is performing a Unlock function	Motor actuates locking mechanism to the UNLOCK position when UNLOCK input is energised	PASS
Script	Animal detection	Observe monitor output during script operation.	Camera successfully operated by script	PASS
		Insert objects into cameras Field of View, observe monitor and terminal output.	Objects identified by script	PASS
		Insert objects into cameras Field of View, observe monitor and terminal output.	Object names captured by script	PASS
		Set script to unlock given a specific object. Insert that object into the Field of View of the camera. Observe monitor and terminal output, as well as door lock.	Unlock triggered by specified object	PASS
		Set script to unlock given a specific object. Insert a different object into the Field of View of the camera. Observe monitor and terminal output, as well as door lock.	Unlock Not triggered by non-specified object	PASS
	Reporting	Observe animal, PIR and system check reports.	Accurate Date and time captured in report	PASS[2]
		Observe animal, PIR and system check reports. Compare with battery in-build level sensor.	Battery status captured in report	PASS[2]
		Set script to unlock given a specific object. Insert that object into the Field of View of the camera. Observe on-board report.	On-board report generated when specified object detected	PASS[2]
		Set script to unlock given a specific object. Insert a different object into the Field of View of the camera. Observe on-board report.	On-board report generated when non-specified object detected	PASS[2]
		Wave hand in the detection zone of the PIR. Observe on-board report.	On-board report generated when PIR sensor trigger	PASS
		Have the system energised for at least 12 hours. Observe on-board report.	On-board report generated when check-in timer trigger	PASS

		Have permitted animal report triggered. Observe Terminal output.	UART script called when on-board report generated when specified object detected	PASS
		Have non-permitted animal report triggered. Observe Terminal output.	UART script called when on-board report generated when non-specified object detected	PASS
		Have PIR report triggered. Observe Terminal output.	UART script called when on-board report generated when PIR sensor trigger	PASS
		Have check-in report triggered. Observe Terminal output.	UART script called when on-board report generated when check-in timer trigger	PASS
		Have UART report called. Observe output from UART pins, or test results in Myriota device	UART message received by Myriota device	FAIL

Notes:

1. System power is stable during hot swapping of batteries provided the second battery contains adequate charge to maintain system power.
2. An excerpt of the on-board report can be found in Appendix G

6.2 Results

6.2.1 Door

The door operates as required. The locking mechanism operates in such a way that egress is not controlled by the system, and therefore egress from the box is always permitted. The door does not foul on any components.

6.2.2 Power supply

The system is able to be operated with one or two 18 V batteries. Both batteries can be connected at the same time, with the systems power supply remaining stable, regardless of battery connection configuration. An operator is able to safely remove and replace a battery while the system is energised, as no power is present on the battery clips while a battery is not connected. The system is able to be live and operating during the battery swap, provided that the other battery has sufficient energy to maintain the required voltage over the battery swap out. Both the power supply and 5 V supply board function as required.

The battery monitor system functions adequately. The current system has been configured to measure Makita 18 V battery(ies). If the operator wishes to change what batteries are used, they may need to perform the calibration steps as outlined in an earlier chapter. The status of both batteries is saved into each report the system creates.

6.2.3 Motor control

The motor controller board functions as required. A defect in the door motor may result in damage to the motor controller or 5 V power supply board, keeping the Jetson Orin safe from electrical damage. The door lock operates consistently and reliably.

6.2.4 Script

The script is able to successfully connect to the camera and receive a video input.

To test the object detection functions, a toothbrush was assigned as a permitted object to the nest box. A toothbrush, a keyboard, and a person were brought into the Field of View of the camera and the actions taken by the system were observed.

When a toothbrush was detected, the system unlocked the door with extremely minimal delay (<0.5 Seconds). The object (toothbrush) was recorded, and a report was logged on the on-board report document, and a UART message passed to the UART script.

When a person or keyboard was presented to the camera, the door was not unlocked. The objects were recorded and a report was logged on the on-board report document. A UART message was passed to the UART script.

6.2.5 UART messaging

While the UART script was able to be called by the main script without an error, the output from the Jetson Orin was not able to be confirmed.

Suspected faults:

- The receiving PC may not have been properly programmed.
- TTL to USB cable may have been defective.
- UART output on the Orin may not have been properly calibrated within the device.

Without a confirmed UART output, the Myriota device connection could not be confirmed. The Orin / Myriota connection was achieved by the partner team, confirming that the connection is possible, therefore it is assumed that the error exists within the specific Jetson Orin running the script. Further testing should take place on the connected Orins of the partner team, running the final scripts on their system.

7 Conclusions

This project designed a system to support the conservation efforts of the School of Agriculture and Environmental Science at UniSQ through the use of computer vison, image processing AI tools and robotics. This project was successful in designing a system that leveraged electronics and computer vision to provide an easily operated, safe, autonomous platform, that can be deployed in remote regions of Australia, and potentially provide a safe haven to small Australian Mammals.

The system designed in this process has been demonstrated to reliably meet the required key criteria established through discussions with the research team and through lessons learnt by previous research papers. The material of the nest box being a soft wood meets the nesting needs of a Sugar Glider, with the nest box design suitable for installation in trees or wooded areas outdoors. A lightweight lockable door of an appropriate size provides a reliable method of entry and exit of the nestbox. The door is able to be locked and unlocked consistently by a monitoring system, ensuring that the nest is protected from animals that may wish to prey on the Sugar Glider, or those who may wish to take over and inhabit the nest themselves. The control system has been demonstrated to be easily configurable, enabling the research team to change the desired animal allowed, or even to set multiple animals that are allowed through the door. Twin battery supply enables the system to be installed in a remote location and operate autonomously for an extended period of time and enables the research team to safely swap batteries without risk of electrical injury or powering down the system. Finally, on-board record keeping and UART output enables researchers to review field performance data of the box, enabling performance and effectiveness to be evaluated.

Delays in component acquisition and identifying and correcting a significant oversight by the manufacturer of the Primary Controller unit resulted in the original planned scope having to be re-evaluated. A completed system was not able to be assembled in time to be delivered to the research team for live trials, therefore no results of real-world performance can be given. Results from previous animal trials utilising identical nest boxes and door systems suggest that Australian mammals could learn to utilise the system confidently given some training. The question of whether a wild animal would utilise the system without any prior coaching or coaxing has yet to be determined and may be the topic of another research proposal.

Further Works

Construction: The original intent of this project was to construct and deliver the system to the research team of the Agricultural and Environmental Science for a supervised live test with the target animal – a Sugar Glider. Unplanned delays in appropriating the primary driver postponed the project such that a finished construction could not be completed.

Animal detection script: The script provided does successfully detect object and provide an output to the system, however, preliminary tests with photos of sugar gliders resulted in an erroneous response of "Cat". Further development and testing of the animal detection is recommended to tighten up the detection rate of various animals. The script in this project is written such that the object detection components can easily be updated without corrupting or needing to modify the primary control script.

Camera location optimisation: Once the object detection has been re-calibrated where a Sugar Glider is successfully and reliably detected, testing should take place to identify the optimal location of the detection camera. As identified earlier in this report, Sugar Gliders are adept climbers and may approach the nest box door from many different directions. Trial and error of various camera locations and potentially the installation of obstacles may be required to determine an optimal solution.

Myriota: As discussed in the results, the Myriota UART connection could not be adequately confirmed during testing. The partner team was able to confirm this connection, therefore it is hoped that the connection fault can be rectified. The UART script is successfully being called by the primary script and should not need much modification to work with the amended system.

Additional control features: The modular design of the system allows for additional components to be added to the system. If the research team wanted to include RFID readers and collared animals into the scope of the research, the amendment both to the hardware and software of the system should be simple.

Amalgamation of circuit boards: It was the initial intention of the design to have a single circuit board provide all of the functionality provided by all individual boards. During the test and design stage, a single board proved to be a significant hinderance in the design and iterate phase, as a seemingly small change to a circuit design could affect the main board in such a way as to require significant re-work, wasting time and materials. Now that each individual board is complete, a consolidated Printed Circuit Board could be considered. It is recommended that the board be designed with digital circuit design tools and fabricated by PCB printing service providers to ensure reliable and accurate circuitry.

References

AABR. 2024. *Groups and Organisations* [Online]. Australian Association of Bush Regenerators. Available: https://www.aabr.org.au/connect/organisations-groups/ [Accessed 23 March 2024 2024].

ABARES 2004. Australia's forests at a glance 2004. Australian Bureau of Agriculture and Resource Economics and Sciences.

ABARES 2019. Australia's forests at a glance 2019. Australian Bureau of Agriculture and Resource Economics and Sciences.

AMAZON. 2024. *waveshare Compatible with NVIDIA Jetson Nano Camera IMX219-160 8MP IR-Cut Infrared Night Vision Camera Module for Jetson Nano and Raspberry Pi Compute Module,162° FOV with IMX219 Sensor* [Online]. Amazon Prime. Available: https://www.amazon.com.au/Compatible-Camera-IMX219-160-Module-Raspberry/dp/B08RBVTLGH [Accessed 29 September 2024 2024].

ARDUINO. 2022. *Arduino Hardware* [Online]. Available: https://www.arduino.cc/en/hardware [Accessed 21 September 2024 2024].

ARDUINO. 2024. *What is Arduino?* [Online]. Available: https://www.arduino.cc/en/Guide/Introduction [Accessed 12 March 2024 2024].

ARENA. 2024. *Off Grid* [Online]. Australian Renewable Energy Agency. Available: https://arena.gov.au/renewable-energy/off-grid/#:~:text=Electricity%20users%20or%20systems%20that,usually%20diesel%20or%20natural%20gas. [Accessed 24 August 2024 2024].

ARETINO, B., HOLLAND, P., PETERSON, D. & SCHUELE, M. 2001. Creating Markets for Biodiversity: A Case Study of Earth Sanctuaries Ltd. *In:* COMMISSION, P. (ed.). Canberra: Media and Publications.

ARLETTAZ, R., SCHAUB, M., FOURNIER, J., REICHLIN, T. S., SIERRO, A., WATSON, J. E. M. & BRAUNISCH, V. 2010. From Publications to Public Actions: When Conservation Biologists Bridge the Gap between Research and Implementation. *BioScience,* 60**,** 835-842.

ARROW. 2018. *The Right Tool for the Job: Active and Passive InfraRed Sensors* [Online]. Arrow Electronics. Available: https://www.arrow.com/en/research-and-events/articles/understanding-active-and-passive-infrared-sensors#:~:text=How%20Do%20PIR%20Sensors%20Work,)%2C%20the%20sensor%20will%20engage. [Accessed 18 August 2024 2024].

AUSTRALIA, W. 2024. *Introduced Predators* [Online]. Available: https://wwf.org.au/what-we-do/species/introduced-predators/ [Accessed 17 March 2024 2024].

AUSTRALIAN WILDLIFE CONSERVANCY. 2024. *Sugar Glider* [Online]. Australian Wildlife Conservancy. Available: https://www.australianwildlife.org/wildlife/sugar-glider/ [Accessed 11 September 2024 2024].

AUSTRALIAN WILDLIFE JOURNEYS. 2024. *Climate and Seasons* [Online]. Australian Wildlife Journeys. Available: https://australianwildlifejourneys.com/plan-your-trip/climate-and-seasons#:~:text=Temperatures%20range%20from%2040%C2%B0,the%20tropical%20regions%20of%20Australia. [Accessed 28 September 2024 2024].

AWS. 2024. *What is Cloud Computing?* [Online]. Amazon Web Services. Available: https://aws.amazon.com/what-is-cloud-computing/ [Accessed 21 September 2024 2024].

AZURE. 2024. *What is Cloud Computing?* [Online]. Microsoft. Available: https://azure.microsoft.com/en-au/resources/cloud-computing-dictionary/what-is-cloud-computing#:~:text=Simply%20put%2C%20cloud%20computing%20is,resources%2C%20and%20economies%20of%20scale. [Accessed 21 September 2024 2024].

BAE SYSTEMS. 2023. *What are single-board computers?* [Online]. BAE systems. Available: https://www.baesystems.com/en-us/definition/what-are-single-board-computers [Accessed 21 September 2024 2024].

BARRITT M. 2024. *Sugar Glider* [Online]. Wildlife Perservation Society of QLD. Available: https://wildlife.org.au/news-resources/educational-resources/species-profiles/mammals/gliders/sugar-gliders/#:~:text=Native%20predators%20in%20their%20range,goannas%2C%20snakes%2C%20and%20quolls. [Accessed 11 September 2024 2024].

BATTERY WORLD. 2024. *ROLLS 6V 415AH RE AGM DEEP CYCLE BATTERY* [Online]. Battery World. Available: https://www.batteryworld.com.au/Rolls-6V-415Ah-RE-AGM-Deep-Cycle-Battery [Accessed 29 Mar 2024 2024].

BERTHIER, K., LEIPPERT, F., FUMAGALLI, L. & ARLETTAZ, R. 2012. Massive nest-box supplementation boosts fecundity, survival and even immigration without altering mating and reproductive behaviour in a rapidly recovered bird population. *PLoS One,* 7**,** e36028.

BEYER, L. & GOLDINGAY, T. 2006. The value of nest boxes in the research and management of Australian hollow-using arboreal marsupials. *Wildlife Research,* Vol 33**,** 161-174.

BLACKWELL M. 2024. *What Really Scares Birds Away?* [Online]. Project Multi Pest. Available: https://pmpest.co.uk/what-really-scares-birds-away/#:~:text=interesting%20FAQs%20below%3A-,Do%20Fake%20Owls%20Really%20Keep%20Birds%20Away%3F,area%20can%20show%20immediate%20results. [Accessed 18 August 2024 2024].

BURNHAM, B. 2021. Comedy. *Inside.* Netflix.

CARRITT, R. 1999. Natrual Tree Hollows. NSW Environment and Heritage.

CHERRIMAN, S. 2015. *Building your own nest-box for wildlife* [Online]. Insight Ornithology. Available: https://www.armadale.wa.gov.au/sites/default/files/assets/documents/docs/Environmental_Management/NestBox.pdf [Accessed 18 August 2024 2024].

COMPONENTS, H. 2024. *L9110H Motor Driver IC* [Online]. Available: https://hobbycomponents.com/ics/834-l9110h-motor-driver-ic [Accessed 26 August 2024 2024].

CONNECTED PLATFORMS. 2024. *A Guide to Types of Wireless Technologies* [Online]. Connected Platforms. Available: https://connectedplatforms.com.au/types-of-wireless-technologies/#:~:text=Types%20of%20Wireless%20Communication,-The%20days%20of&text=Some%20of%20these%20terms%20may,different%20uses%20in%20some%20cases. [Accessed 14 September 2024 2024].

DALES, J. T. 2011. Death by a thousand cuts: incorporating cumulative effects in Australia's Environment Protection and Biodiversity Conservation Act. *Pac. Rim L. & Pol'y J.,* 20**,** 149.

DICKMAN, C. R. & LINDENMAYER, D. B. 2021. Australia's Natural Environment: A Warning for the World. *Sustainability and the New Economics: Synthesising Ecological Economics and Modern Monetary Theory.* Springer.

DILONARDO M. 2022. *Do Fake Owls and Other Decoys Work?* [Online]. Treehugger. Available: https://www.treehugger.com/how-do-fake-owls-and-other-decoys-work-4863566 [Accessed 18 August 2024 2024].

EDWARDS, M. C., HOY, J. M., FITZGIBBON, S. I. & MURRAY, P. J. 2020. Bandicoot bunkers: training wild-caught northern brown bandicoots (Isoodon macrourus) to use microchip-automated safe refuge. *Wildlife Research,* 47**,** 239.

EPANCHIN-NIELL, R. S., HAIGHT, R. G., BEREC, L., KEAN, J. M. & LIEBHOLD, A. M. 2012. Optimal surveillance and eradication of invasive species in heterogeneous landscapes. *Ecol Lett,* 15**,** 803-12.

EWEN, J. G., WALKER, L., CANESSA, S. & GROOMBRIDGE, J. J. 2015. Improving supplementary feeding in species conservation. *Conservation Biology,* 29**,** 341-349.

FORESTS AUSTRALIA. 2024. *Rainforest* [Online]. Available: https://www.agriculture.gov.au/abares/forestsaustralia/australias-forests/profiles/rainforest-2019 [Accessed 23 March 2024 2024].

GENZEL, J. 2021. *4 Technical Aspects of Trail Cameras You Should Know Before Buying One* [Online]. OutdoorLife. Available: https://www.outdoorlife.com/story/hunting/technical-aspects-of-trail-cameras/#:~:text=But%20the%20fact%20is%2C%20most,of%20the%20larger%20file%20size. [Accessed 31 Mar 2024 2024].

GLOBAL FOREST WATCH. 2023. *Australian Forests* [Online]. Available: https://www.globalforestwatch.org/dashboards/country/AUS/?category=undefined [Accessed 02/03/2024 2023].

GOVERNMENT, Q. 2016. *Animal Ethics Committee* [Online]. Business Queensland, Queensland Government. Available: https://www.business.qld.gov.au/industries/farms-fishing-forestry/agriculture/animal/health/welfare/science/ethics [Accessed 14 September 2024 2024].

GUIDE, E. I. 2024. *Extra Low Voltage (ELV)* [Online]. Schneider Electric. Available: https://www.electrical-installation.org/enwiki/Extra_Low_Voltage_(ELV) [Accessed 17 August 2024 2024].

HASHEMI-POUR, C. 2024. *What is a microcontroller (MCU)* [Online]. TechTarget. Available: https://www.techtarget.com/iotagenda/definition/microcontroller [Accessed 21 September 2024 2024].

HEFT, M. 2020. *How Technology Can Transform Wildlife Conservation'* [Online]. Conservation Guide. Available: https://www.conservationguide.org/news/how-technology-can-transform-wildlife-conservation [Accessed 11 Feb 2024 2024].

HEMMENDINGER, D. 2024. *Computer programming language* [Online]. Britannica. Available: https://www.britannica.com/technology/computer-programming-language [Accessed 6 April 2024 2024].

HOLDSWORTH, J. & SCAPICCHIO, M. 2024. *What is deep learning?* [Online]. IBM. Available: https://www.ibm.com/topics/deep-learning [Accessed 28 September 2024 2024].

HUB360. 2024. *5 V DC motor* [Online]. hub360. Available: https://hub360.com.ng/product/5v-dc-motor/ [Accessed 19 August 2024 2024].

INFRASTRUCTURE AUSTRALIA. 2023. *Mobile telecommunications coverage in regional and remote areas* [Online]. Infrastructure Australia. Available: https://www.infrastructureaustralia.gov.au/map/mobile-telecommunications-coverage-regional-and-remote-areas [Accessed 21 September 2024 2024].

ISAAC, J. L., JOHNSON, C. N., GRABAU, P. J. & KROCKENBERGER, A. K. 2004. Automated feeders: new technology for food supplementation experiments with mammals. *Wildlife Research,* 31**,** 437.

JEWELL, Z. 2013. Effect of monitoring technique on quality of conservation science. *Conserv Biol,* 27**,** 501-8.

KOK A., VAN HULTEN D., TIMMERMAN K., LANKHORST J., VISSER F. & SLABBEKOORN H. 2021. Interacting effects of short-term and long-term noise exposure on antipredator behaviour in sand gobies. *Animal Behaviour,* 172**,** 93-102.

LAWTON, J. A., HOLLAND, G. J., TIMEWELL, C., BANNON, A., MELLICK, E. & BENNETT, A. F. 2022. Citizen science and community action provide insights on a threatened species: nest box use by the brush-tailed phascogale (Phascogale tapoatafa). *Wildlife Research,* 49**,** 513-528.

LINCHANT, J., LISEIN, J., SEMEKI, J., LEJEUNE, P. & VERMEULEN, C. 2015. Are unmanned aircraft systems (UASs) the future of wildlife monitoring? A review of accomplishments and challenges. *Mammal Review,* 45**,** 239-252.

MAKERMAX. 2024. *Discharge Characteristics of Lithium-ion Batteries* [Online]. MakerMax. Available: https://makermax.ca/articles/discharge-characteristics-of-lithium-ion-batteries/ [Accessed 29 September 2024 2024].

MCGLASHAN A. 2000. *Materials to Use or Not to Use to Make Nest Boxes for Wildlife* [Online]. NestBox Tales. Available: https://nestboxtales.com/wp-content/uploads/2020/01/Nest-Box-Materials.pdf [Accessed 14 September 2024 2024].

MELBOURNE-THOMAS, J., LIN, B., SYME, L. & HOPKINS, M. 2024. *Climate change will strike Australia's precious World Heritage sites – and Indigenous knowledge is a key defence* [Online]. CSIRO. Available: https://www.csiro.au/en/news/All/Articles/2024/February/climate-change-heritage-sites [Accessed 17 Mar 2024 2024].

MONODAQ. 2024. *Single board computer power consumption and heatsink temperature* [Online]. MonoDAQ. Available: https://www.monodaq.com/applications/single-board-computer-power-consumption-heatsink-temperature/ [Accessed 21 September 2024 2024].

MYRIOTA. 2024. *FAQ* [Online]. Myriota. Available: https://myriota.com/faqs/#:~:text=The%20intelligent%20design%20of%20both,battery%20life%20in%20the%20field. [Accessed 26 October 2024 2024].

MYRIOTA DEVELOPER. 2024. *Module Code Examples* [Online]. Myriota. Available: https://support.myriota.com/hc/en-us/articles/6158658553743-Module-Code-Examples [Accessed 26 October 2024 2024].

NATIONAL LAND AND WATER RESOURCES AUDIT 2001. Australian native vegetation assessment. Canberra, Australia.

NEST BOX TALES. 2024. *Sugar Glider* [Online]. Nest Box Tales. Available: https://nestboxtales.com/nest-box-designs/designs-sugar-glider/ [Accessed 8 September 2024 2024].

NVIDIA DEVELOPER. 2024a. *Jetson Nano 2GB Developer Kit User Guide* [Online]. Available: https://developer.nvidia.com/embedded/learn/jetson-nano-2gb-devkit-user-guide [Accessed 14 March 2024 2024].

NVIDIA DEVELOPER. 2024b. *Jetson Orin Nano Developer Kit Carrier Board* [Online]. NVIDIA. Available: https://developer.download.nvidia.com/assets/embedded/secure/jetson/orin_nano/docs/Jetson-Orin-Nano-DevKit-Carrier-Board-Specification_SP-11324-001_v1.2.pdf?hFzM_63O0xYgOXjYINrGL3yTm1gtNu8MXg--6lTl1oZkehJR9TFXL8M-i4GCo16JxsOtAQrHE44_lQcxHUaq5aWrPrHQQVL5mkkLUYoUyVkR84ByWJIqhu4ONTd-ZFZMXnKH3Ex5y4kajrzd_CdVDSwOtsqmRhbSoA_E6WP4v9ZHj2Qjnc_OgsPM8K8ohnvS8XA2xta0y2z7EpZYHflO-8PSN2Uhq_2Ni6mTY1tZpGK4jlhA4A==&t=eyJscyI6ImdzZW8iLCJsc2QiOiJodHRwczovL3d3dy5nb29nbGUuY29tLyJ9 [Accessed 21 September 2024 2024].

PAPE, C. 2023. Best programming language for image analysis in science - the winner is…. Neuroscience and Beyond.

PELL, A. S. & TIDEMANN, C. R. 1997. The impact of two exotic hollow-nesting birds on two native parrots in savannah and woodland in eastern Australia. *Biological Conservation,* 79**,** 145-153.

RASPBERRY PI. 2024. *Raspberry Pi Datasheets* [Online]. Raspberry Pi. Available: https://datasheets.raspberrypi.com/ [Accessed 21 September 2024 2024].

RENOGY. 2023. *Batteries in Series Vs Parallel* [Online]. Renogy. Available: https://ca.renogy.com/blog/batteries-in-series-vs-parallel-what-are-the-differences/ [Accessed 19 august 2024 2024].

ROGERS, A. 2019. *Gimmi shelter: Conserving hollow-nesting birds* [Online]. Threatened Species Recovery Hub. Available: https://www.nespthreatenedspecies.edu.au/news-and-media/latest-news/gimme-shelter-conserving-hollow-nesting-birds [Accessed 23 March 2024 2024].

SAKOVICH, N. 2023. *Image Recognition: Which Programming Language to Choose?* [Online]. SaM Solutions. Available: https://www.sam-solutions.com/blog/image-recognition-programming-language/ [Accessed 6 April 2024 2024].

SALA, O. E., STUART CHAPIN, F., ARMESTO, J. J., BERLOW, E., BLOOMFIELD, J., DIRZO, R., HUBER-SANWALD, E., HUENNEKE, L. F., JACKSON, R. B., KINZIG, A., LEEMANS, R., LODGE, D. M., MOONEY, H. A., OESTERHELD, M. N., POFF, N. L., SYKES, M. T., WALKER, B. H., WALKER, M. & WALL, D. H. 2000. Global Biodiversity Scenarios for the Year 2100. *Science,* 287**,** 1770-1774.

SAUNDERS, D. A., DAWSON, R. & MAWSON, P. R. 2022. Artificial nesting hollows for the conservation of Carnaby's cockatoo Calyptorhynchus latirostris: definitely not a case of erect and forget. *Pacific Conservation Biology,* 29**,** 119-129.

SCHIAVONE, D. 2022. *How to design and size a solar battery system* [Online]. EnergyUME. Available: https://www.youtube.com/watch?v=FtO1sunTHOE [Accessed 14 April 2024 2024].

SHARMA, R. 2021. *Computer Vision and Computing Power — What processing power do you need to do Computer Vision ?* [Online]. Medium. Available: https://medium.com/analytics-vidhya/computer-vision-and-computing-power-what-processing-power-do-you-need-to-do-computer-vision-c5c45e51a837 [Accessed 31 Mar 2024 2024].

SUREPET. 2024. *Microchip Cat Door* [Online]. Available: https://www.surepetcare.com/en-au/pet-doors/microchip-cat-flap [Accessed 21 September 2024 2024].
THE THINGS NETWORK. 2024. *What are LoRa and LoRaWAN* [Online]. Available: https://www.thethingsnetwork.org/docs/lorawan/what-is-lorawan/ [Accessed 17 March 2024 2024].
TUAMA, D. 2024. *Understanding Libraries in Python: A Comprehensive Guide* [Online]. Code Institute. Available: https://codeinstitute.net/global/blog/what-are-libraries-in-python/#:~:text=What%20is%20python%20Library%3F,without%20a%20library%20like%20this. [Accessed 6 October 2024 2024].
ULTRALYTICS. 2024. *Ultralytics YOLO Frequenty Asked Questions (FAQ)* [Online]. Ultralytics. Available: https://docs.ultralytics.com/help/FAQ/ [Accessed 8 September 2024 2024].
VERDICT. 2022. *SlothBots, SnotBots, and Robobees: The role of robotics in conservation* [Online]. Verdict. Available: https://www.verdict.co.uk/robotics-in-conservation/?cf-view [Accessed 23 March 2024 2024].
WAAG, W. & SAUER, D. U. 2009. SECONDARY BATTERIES – LEAD– ACID SYSTEMS | State-of-Charge/Health. *In:* GARCHE, J. (ed.) *Encyclopedia of Electrochemical Power Sources.* Amsterdam: Elsevier.
WARD, M. S., SIMMONDS, J. S., RESIDE, A. E., WATSON, J. E. M., RHODES, J. R., POSSINGHAM, H. P., TREZISE, J., FLETCHER, R., FILE, L. & TAYLOR, M. 2019. Lots of loss with little scrutiny: The attrition of habitat critical for threatened species in Australia. *Conservation Science and Practice,* 1.
WATSON, S. J., HOY, J. M., EDWARDS, M. C. & MURRAY, P. J. 2021. First use of a microchip-automated nest box. *Australian Mammalogy,* 44**,** 139-142.
WATSON, S. J., HOY, J. M., EDWARDS, M. C. & MURRAY, P. J. 2022. First use of a microchip-automated nest box in situ by a brush-tailed phascogale (Phascogale tapoatafa). *Australian Mammalogy,* 44**,** 139-142.
WIKIMEDIA. 2022. *Makita LXT* [Online]. Wikipedia. Available: https://commons.wikimedia.org/wiki/Category:Makita_LXT#:~:text=Batteries%20are%20constructed%20with%20multiple,four%2018650%20cells%20in%20series). [Accessed 23 June 2024 2024].
WILDERNESS SOCIETY. 2024. *Deforestation explained* [Online]. Wildnerness Society. Available: https://www.wilderness.org.au/protecting-nature/deforestation/deforestation-explained [Accessed 23 March 2024 2024].
WILDLIFE INSIGHTS. 2024. *Bringing Cutting-Edge Technnology to Wildlife Conservation* [Online]. Wildlife Insights. Available: https://www.wildlifeinsights.org/ [Accessed 24 March 2024 2024].
WOINARSKI, J. C. Z., BURBIDGE, A. A. & HARRISON, P. L. 2015. Ongoing unraveling of a continental fauna: Decline and extinction of Australian mammals since European settlement. *Proceedings of the National Academy of Sciences,* 112**,** 4531-4540.
XIONG, R., PAN, Y., SHEN, W., LI, H. & SUN, F. 2020. Lithium-ion battery aging mechanisms and diagnosis method for automotive applications: Recent advances and perspectives. *Renewable and Sustainable Energy Reviews,* 131**,** 110048.
ZEREN, B. 2024. *The Evolution of Computers and Computer Timeline* [Online]. Book Your Data. Available: https://www.bookyourdata.com/blog/computers-changed-history [Accessed 21 September 2024 2024].

Appendix A: Specification and Work plan document submission 12/02/2024

Introduction and Background

The School of Agriculture and Environmental Science has sought help to design and construct a gating and/or deterrent solution for nesting boxes and develop a monitoring system to analyze the target and non-target species behavior in relation to these boxes.

Native mammals and marsupials are under increasing pressure from human redevelopment of their natural living and breeding grounds. As bushlands and forests are removed, so too are suitable nesting habitats that are crucial to the reproductive process of these animals. Exotic predators and smaller exotic nesting animals are adding yet another layer of pressure on target native wildlife populations. These predators may target native nesting species as an easy source of food. Smaller exotic species of animal may inhabit the already rare nesting areas required by the native species to survive and breed. With less available nesting sites available, and increasing pressure of predation, additional conservation efforts are required to help the dwindling native populations of nesting species of animals survive and grow.

Advances in electronics, robotics, and computer vision technology are being used to help wildlife conservation in a variety of modes, allowing for new and effective means of monitoring, intervention, and protection. Drones and Spatial Monitoring and Reporting Tools (SMART) have been deployed to track illegal poaching in areas that would normally be nearly impossible to monitor (Heft 2020). Microchips for access automation have been leveraged in combination with animal training to create safe havens for bandicoots (Edwards et al. 2020). Finally, with the recent leaps in AI and computer vision processing tools, animal detection and classification can be automated and carried out at the same instant an animal is detected and captured on a digital camera. Services such as Wildlife Insights can provide an online platform for image submission and processing, as well as a database of images containing the sought-after target species, providing a valuable dataset for computer learning algorithms to refine their species identification capacity (Wildlife Insights n.d).

Beyer and Goldingay (2006) performed a review of published works regarding nesting boxes, and identified species detection, species ecology and the preference in box design in different species as prominent applications for research. This research project proposes to provide a system of access automation to nesting boxes and leverage computer vision systems, create monitoring and reporting tools and, in order to study both the effectiveness the access systems and analyze potential changes in behavioral patterns of animals interacting with the system.

Objectives and Aims:

Specific Objectives:

Design and construct a gating and/or deterrent system for a nesting box. The system is to be triggered by a separate animal identification system (developing the animal identification system is not in the scope of this project).

Develop a monitoring system to collect data on various events and parameters, e.g. nesting box usage, predator attacks.

Set up a remote reporting system to collate and send nesting performance data for research and analysis.

Install and monitor inert variants of the nesting box in the same vicinity of the automated boxes.

(Optional) Install and monitor inert variants of the nesting box in locations that are not in proximity to the automated nesting boxes.

(Optional) Install and monitor different inert variants of nesting box designs to investigate animal preferences for nest box design and design suitability for nesting, predation protection etc.

Ensure that all designs are safe to interact with both the target and non-target animals. The nesting boxes must not be able to trap an animal inside the box in the event of a hardware or software failure. Designs shall be suitable to be installed in harsh outdoor conditions and must reasonably resist encroachment or attack by animals.

Expected Outcomes:

Key criteria

Demonstrate that nesting box efficacy for target species can be improved through the use of an automation system.

Demonstrate that target animal species can utilise the nesting box, while non-target species are prohibited from entry through the use of the automated system.

(Optional) Investigate if the use of access control and/or deterrents on controlled nesting boxes can cause a change in animal behaviour in relation to the other inert nesting boxes.

Work Plan

Month	Objectives	Key Targets
Jan - Mar	Develop project key criteria. Submit proposal to supervisor and team for approval and review. Set up CAD software. Begin literature review – nesting boxes, safety in design, automation in harsh conditions, IoT systems. Review hardware and software either provided or suggested by supervision team.	12 Feb – Specification and Work Plan
April	Finalise literature review. Discuss literature review and methodology with supervisor and Environment team. Develop Methodology Begin prototyping (3D printer or hand built) proof on concept designs for door mechanisms. Review IoT devices for reporting system, acquire IoT device(s) and online platform(s) for testing and commissioning	2 Apr – Literature Review
May	Finalise methodology and seek final review. Finalise door and/or deterrent design. Begin construction of nesting boxes	20 May – Methodology
June	Finalise construction of nesting boxes Acquire all monitoring hardware. Install automated and non-automated nesting boxes. Install monitoring systems.	
July – Aug	Download and review monitoring data as it is collected. Monitor for system failures, reactively repair or replace if required. Monitor nesting box interactions daily to ensure no animals are trapped or harmed. Seek immediate advice if adverse effects on animals are detected.	
September	Continue to download and collate nesting box results. Collated, review and draft results.	9 Sep – Draft Dissertation, preliminary results
October	Recover nesting boxes and monitoring systems. Prepare presentation. Prepare Reflection.	14 Oct – Presentation 21 Oct - Reflection
November	Finalise and submit dissertation	4 Nov - Submit Dissertation

Resources Required:

Equipment:

3D printer for prototyping.

Hand tools and basic building materials for nest box construction

Actual or simulated connection with required camera and/or processing unit for system integration.

RFID door for modification

Animal/Trail camera for an independent system to perform prototype testing and monitoring.

IoT device for recording and broadcasting data.

Software:

Autodesk AutoCAD and Inventor (or similar. Could potentially be sourced from my place of work.)

3D printer interface software.

Online IoT hosting, data lake, reporting tool.

Access:

CSIRO publishing access for literature review.

USQ library, repository of reports, studies, journals and previous dissertations.

Appendix B: Original Object Detection Script

Original Object detection script supplied to the project by the partner team.

Authors: Mr Arien Westerman, Dr. Derek Long, 2024

```
from ultralytics import YOLO
import cv2
import math
import time
import logging
import argparse

Minimum_Confidence = 0

# Flags
default_printTerminal = True # Prints inference time, classes, confidence, etc to terminal if 'True'
default_display = True # Brings up webcam feed if 'True'
default_logOutput = True # Enable output logging to 'detection.log' while 'True'
default_classTrigger = False # This enables the bottom section of code that runs code based on
detection when set to 'True' e.g. when 'True' 'if Detection = True \ open_door()'


# key vars
Model_Path = "/home/unisq/Documents/yolo-weights/yolov8n.engine" # Model Location
Detection_Classes = ["person", "cell phone"] # Which things do you want to activate detection
Detection_Class_Minimum_Confidence = { # Confidence level required for activation
"person": 0.9,
"cell phone": 0.6
}
# Target Resolution             # you can change these to suit your camera
Camera_X_Resolution = 1280
Camera_Y_Resolution = 720


# handling of command line arguments
parser = argparse.ArgumentParser(description='Process some flags.')

# Define the command-line arguments
parser.add_argument('--printTerminal', type=bool, default=default_printTerminal,
        help=f'Set flag_printTerminal (default: {default_printTerminal})')
parser.add_argument('--display', type=bool, default=default_display,
        help=f'Set flag_display (default: {default_display})')
parser.add_argument('--logOutput', type=bool, default=default_logOutput,
        help=f'Set flag_logOutput (default: {default_logOutput})')
parser.add_argument('--classTrigger', type=bool, default=default_classTrigger,
        help=f'Set flag_classTrigger (default: {default_classTrigger})')

# Parse the arguments
```

```
args = parser.parse_args()

# Assign the parsed values to the flags
flag_printTerminal = args.printTerminal
flag_display = args.display
flag_logOutput = args.logOutput
flag_classTrigger = args.classTrigger



#### Into the script proper ####
# Configure Logging
if flag_logOutput:
logging.basicConfig(
filename='/home/unisq/Documents/logs/detection.log',  #this is hardcoded sthe script can find the folder no matter where it is being called from e.g. Service
level=logging.INFO,
format='%(asctime)s - %(levelname)s - %(message)s'
)

# Start Camera
print("Starting Camera")
cap = cv2.VideoCapture(1)
cap.set(3, Camera_X_Resolution)
cap.set(4, Camera_Y_Resolution)

# Point to model with respect to detect.py
print("Loading Model")
trt_model = YOLO(Model_Path)

# Define Classes
classNames = ["person", "bicycle", "car", "motorbike", "aeroplane", "bus", "train", "truck",
"boat","traffic light", "fire hydrant", "stop sign", "parking meter", "bench", "bird", "cat",
"dog", "horse", "sheep", "cow", "elephant", "bear", "zebra", "giraffe", "backpack", "umbrella",
"handbag", "tie", "suitcase", "frisbee", "skis", "snowboard", "sports ball", "kite", "baseball bat",
        "baseball glove", "skateboard", "surfboard", "tennis racket", "bottle", "wine glass", "cup",
        "fork", "knife", "spoon", "bowl", "banana", "apple", "sandwich", "orange", "broccoli",
        "carrot", "hot dog", "pizza", "donut", "cake", "chair", "sofa", "pottedplant", "bed",
        "diningtable", "toilet", "tvmonitor", "laptop", "mouse", "remote", "keyboard", "cell phone",
        "microwave", "oven", "toaster", "sink", "refrigerator", "book", "clock", "vase", "scissors",
        "teddy bear", "hair drier", "toothbrush"
        ]

# Begin inference
print("Starting Detection")

if flag_display:
cv2.startWindowThread()
cv2.namedWindow("Webcam")
```

```
while True:
success, img= cap.read()
results = trt_model(img, stream=True, verbose=False) # Can change to true to see some of the raw
results output
Detection = False
Detection_Info = []
class_Detection_Info = []
for r in results:
boxes = r.boxes
for box in boxes:


# confidence
confidence = math.ceil((box.conf[0]*100))/100
#print("Confidence --->",confidence)
# class name
cls = int(box.cls[0])
#print("Class name -->", classNames[cls])
class_name = classNames[cls]

Detection_Info.append(f"{class_name} {confidence:.2f}")


if class_name in Detection_Classes:
Minimum_Confidence = Detection_Class_Minimum_Confidence.get(class_name, 0.0)
if confidence > Minimum_Confidence:
Detection = True
class_Detection_Info.append(f"{class_name} {confidence:.2f}")

if flag_display:
# bounding box
x1, y1, x2, y2 = box.xyxy[0]
x1, y1, x2, y2 = int(x1), int(y1), int(x2), int(y2) # convert to int values

# put box in cam
cv2.rectangle(img, (x1, y1), (x2, y2), (255, 0, 255), 3)

# object details
org = [x1, y1]
font = cv2.FONT_HERSHEY_SIMPLEX
fontScale = 1
color = (255, 0, 0)
thickness = 2

label = f"{classNames[cls]} {confidence:.2f}"

cv2.putText(img, label, org, font, fontScale, color, thickness)
```

```
#coming out of loop and actioning on overall results

if flag_printTerminal:
print("Detected objects:" , ",".join(Detection_Info))

if flag_logOutput:
logString = ",".join(Detection_Info)
logging.info(logString)

if flag_classTrigger:
if Detection:
# Put activate code here
print("Trigger" , ",".join(class_Detection_Info))
else:
print("No trigger")
if flag_display:
cv2.imshow('Webcam', img)
cv2.waitKey(20)
if cv2.waitKey(1) == ord('p'):
break

cap.release()
cv2.destroyAllWindows()
```

Appendix C: LEUART script

The script below was created by Mr Arien Westerman and given to the project with permission given to utilise and modify as required.

```
import time
import serial
import threading

class UARTCommunication(threading.Thread):
  def __init__(self, port, baudrate, interval_seconds=300):
    super().__init__()
    self.port = port
    self.baudrate = baudrate
    self.interval_seconds = interval_seconds
    self.running = True
    self.counter = 1

  def run(self):
    while self.running:
      try:
        with serial.Serial(
          port=self.port,
          baudrate=self.baudrate,
          bytesize=serial.EIGHTBITS,
          parity=serial.PARITY_NONE,
          stopbits=serial.STOPBITS_ONE
        ) as serial_port:
          while self.running:
            payload_message = f"Count:{self.counter:04d}\n"
            serial_port.write(payload_message.encode())
            print(f"Sent payload: {payload_message.strip()}")
            self.counter += 1
            time.sleep(self.interval_seconds)
      except serial.SerialException as e:
        print(f"Serial port error: {e}")
```

```
                self.running = False
            except KeyboardInterrupt:
                print("Exiting Program")
                self.running = False
            except Exception as exception_error:
                print("Error occurred. Exiting Program")
                print(f"Error: {exception_error}")
                self.running = False

    def stop(self):
        self.running = False

if __name__ == "__main__":
    print("UART Demonstration Program")
    print("NVIDIA Jetson Nano Developer Kit")
    interval_seconds = 10
    uart_thread = UARTCommunication("/dev/ttyTHS1", 9600, interval_seconds)
    uart_thread.start()

    try:
        uart_thread.join()
    except KeyboardInterrupt:
        print("Main program interrupted")
    finally:
        uart_thread.stop()
        uart_thread.join()
```

Appendix D: Myriota Onboard Script

The script below is freely available from the Myriota Development team as part of their Software Development Kit (Myriota Developer, 2024)

```
#include <string.h>
#include <stdio.h>  // For printf
#include <time.h>   // For time_t
#include "myriota_user_api.h"

#define RECEIVE_TIMEOUT 1000  // [ms]
#define OVERLOAD_STRING "\nOVERLOADED\n"
#define MAX_MESSAGE_SIZE 20  // Define maximum message size

static void *leuart_handle = NULL;
// Read string from UART with timeout, return number of bytes read
int UARTReadStringWithTimeout(void *Handle, uint8_t *Rx, size_t MaxLength) {
  const uint32_t start = TickGet();
  int count = 0;
  while (TickGet() - start < RECEIVE_TIMEOUT) {
    uint8_t ch;
    if (UARTRead(Handle, &ch, 1) == 1) {
      if (count < MaxLength - 1) {  // Leave space for null terminator
        Rx[count] = ch;
        count++;
      }
    }
  }
  Rx[count] = '\0';  // Null-terminate the string
  return count;
}

// Handle incoming payload
void HandlePayload() {
  uint8_t Rx[MAX_MESSAGE_SIZE] = {0};
    int len = UARTReadStringWithTimeout(leuart_handle, Rx, MAX_MESSAGE_SIZE);
```

```
    if (len > 0) {
        if (len > MAX_MESSAGE_SIZE - 1) {  // Adjust for null terminator
            printf("LEUART RX buffer overloaded\n");
            UARTWrite(leuart_handle, (uint8_t *)OVERLOAD_STRING, strlen(OVERLOAD_STRING));
        } else {
            printf("Received payload message: %s\n", Rx);
            // Add debug output to ensure `ScheduleMessage` is called
            printf("Calling ScheduleMessage\n");
            ScheduleMessage(Rx, len);
            printf("Message Scheduled\n");
        }
    } else {
        printf("No data received\n");
    }
    // Get the amount of message space left
    size_t freeSpace = MessageBytesFree();
        // Print the amount of message space left
    printf("Message space left: %d bytes\n", freeSpace);
}
static time_t UartComm() {
    while (1) {
        HandlePayload();
        // This loop will continue indefinitely; add break or condition as needed
        break;
    }
        return OnLeuartReceive();
}
void AppInit() {
    // Initialize UART
    leuart_handle = UARTInit(LEUART, 9600, 0);
    if (leuart_handle == NULL) {
        printf("Failed to initialise leuart\n");
        return;
    }
```

```
    // Verify the UART initialization status
  printf("LEUART initialized successfully.\n");
  // Schedule the UartComm job
  ScheduleJob(UartComm, OnLeuartReceive());
}
```

Appendix E: Python Main Script

As per Appendix A, original script written by Dr Derek Long and Arien Westerman.

All modifications and additions written by Scott Essery

```
######## Import libraries ############

from ultralytics import YOLO
import cv2
import math
import time
import datetime
from datetime import timedelta
import logging
import argparse

import Jetson.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)

######## establish connection to partner script  ############
#light function accepts the strings "red" "green" "blue" to control LED
#door function accepts the string "unlock" to cycle the door unlock
#PIR returns a value of 1 or 0, depending on the PIR input status

#Record keeping function. Call this function with 3 arguments.
#Arg 1:  "1" is camera detection, "2" is PIR, "3" is System checkin
#Arg 2: string from the camera. enter NA is not using type 1
#Arg 3 is the UART message counter, increment numerical value.
#Record captures the date and time, as well as the battery levels.

from detectPartner import light, door, PIR, battCheck, record
```

```
######## Establish variables, pins, timers etc ############

#Controllable timers

#how often do you want the system to do a check-in?(in minutes)
checkin_timer = 30

#how long should the PIR wait before triggering again? (in seconds)
PIR_timer = 30

#how long should the camera system wait before triggering again? (in seconds)
#note, a minimum of approx 15 seconds is required to avoid multiple reports being generate for a single camera sighting.
trigger_timer = 30

Initial_Minimum_Confidence = 0
object_found = 0
hourly_timer = datetime.datetime.now() + datetime.timedelta(minutes=checkin_timer)
buffer_timer = datetime.datetime.now()
PIR_buffer_timer = datetime.datetime.now()
UART_counter = 0

#list of allowed object to unlock the door
allowed_objects = ["toothbrush","keyboard"]

# key vars
Model_Path = "/home/unisq/Documents/yolo-weights/yolov8n.engine" # Model Location
Detection_Classes = ["keyboard", "person", "cell phone", "toothbrush", "book"] # Which things do you want to activate detection
Detection_Class_Minimum_Confidence = { # Confidence level required for activation
        "person": 0.9,
        "cell phone": 0.6,
```

```
        "toothbrush": 0.6,

        "book" : 0.8,

        "keyboard" : 0.8

}


# Flags

default_printTerminal = False # Prints inference time, classes, confidence, etc to terminal if 'True'

default_display = True # Brings up webcam feed if 'True'

default_logOutput = False # Enable output logging to 'detection.log' while 'True'

default_classTrigger = True # This enables the bottom section of code that runs code based on detection when set to 'True' e.g. when 'True' 'if Detection = True \ open_door()'

default_object_itendifiedTrigger = True



# Target Resolution           # you can change these to suit your camera

Camera_X_Resolution = 1280

Camera_Y_Resolution = 720



# handling of command line arguments

parser = argparse.ArgumentParser(description='Process some flags.')


# Define the command-line arguments

parser.add_argument('--printTerminal', type=bool, default=default_printTerminal,

        help=f'Set flag_printTerminal (default: {default_printTerminal})')

parser.add_argument('--display', type=bool, default=default_display,

        help=f'Set flag_display (default: {default_display})')

parser.add_argument('--logOutput', type=bool, default=default_logOutput,

        help=f'Set flag_logOutput (default: {default_logOutput})')

parser.add_argument('--classTrigger', type=bool, default=default_classTrigger,

        help=f'Set flag_classTrigger (default: {default_classTrigger})')


# Parse the arguments

args = parser.parse_args()


# Assign the parsed values to the flags

flag_printTerminal = args.printTerminal
```

```
flag_display = args.display

flag_logOutput = args.logOutput

flag_classTrigger = args.classTrigger




#### Into the script proper ####

# Configure Logging

if flag_logOutput:

	logging.basicConfig(

		filename='/home/unisq/Documents/logs/detection.log',  #this is hardcoded sthe script can find the folder no matter where it is being called from e.g. Service

		level=logging.INFO,

		format='%(asctime)s - %(levelname)s - %(message)s'

	)


# Start Camera

print("Starting Camera")




cap = cv2.VideoCapture(0)

cap.set(3, Camera_X_Resolution)

cap.set(4, Camera_Y_Resolution)


# Point to model with respect to detect.py

print("Loading Model")

trt_model = YOLO(Model_Path)


# Define Classes

classNames = ["person", "bicycle", "car", "motorbike", "aeroplane", "bus", "train", "truck", "boat","traffic light", "fire hydrant", "stop sign", "parking meter", "bench", "bird", "cat",

"dog", "horse", "sheep", "cow", "elephant", "bear", "zebra", "giraffe", "backpack", "umbrella", "handbag", "tie", "suitcase", "frisbee", "skis", "snowboard", "sports ball", "kite", "baseball bat",

    "baseball glove", "skateboard", "surfboard", "tennis racket", "bottle", "wine glass", "cup",

    "fork", "knife", "spoon", "bowl", "banana", "apple", "sandwich", "orange", "broccoli",

    "carrot", "hot dog", "pizza", "donut", "cake", "chair", "sofa", "pottedplant", "bed",

    "diningtable", "toilet", "tvmonitor", "laptop", "mouse", "remote", "keyboard", "cell phone",
```

```
		"microwave", "oven", "toaster", "sink", "refrigerator", "book", "clock", "vase", "scissors",
		"teddy bear", "hair drier", "toothbrush"
		]

# Begin inference
print("Starting Detection")

if flag_display:
	cv2.startWindowThread()
	cv2.namedWindow("Webcam")

while True:
	success, img= cap.read()
	results = trt_model(img, stream=True, verbose=False) # Can change to true to see some of the raw results output
	Detection = False
	Detection_Info = []
	class_Detection_Info = []
	for r in results:
		boxes = r.boxes
		for box in boxes:


			# confidence
			confidence = math.ceil((box.conf[0]*100))/100
			#print("Confidence --->",confidence)
			# class name
			cls = int(box.cls[0])
			#print("Class name -->", classNames[cls])
			class_name = classNames[cls]

			Detection_Info.append(f"{class_name} {confidence:.2f}")



			if class_name in Detection_Classes:
				Minimum_Confidence = Detection_Class_Minimum_Confidence.get(class_name, 0.0)
				if confidence > Minimum_Confidence:
```

```
                    Detection = True
                    class_Detection_Info.append(f"{class_name} {confidence:.2f}")
                    object_found = class_name
                    print("detected : " + object_found)

            else:
                Minimum_Confidence = Initial_Minimum_Confidence

            if flag_display:
                # bounding box
                x1, y1, x2, y2 = box.xyxy[0]
                x1, y1, x2, y2 = int(x1), int(y1), int(x2), int(y2) # convert to int values

                # put box in cam
                cv2.rectangle(img, (x1, y1), (x2, y2), (255, 0, 255), 3)

                # object details
                org = [x1, y1]
                font = cv2.FONT_HERSHEY_SIMPLEX
                fontScale = 1
                color = (255, 0, 0)
                thickness = 2

                label = f"{classNames[cls]} {confidence:.2f}"

                cv2.putText(img, label, org, font, fontScale, color, thickness)

#coming out of loop and actioning on overall results

if flag_printTerminal:
    print("Detected objects:" , ",".join(Detection_Info))

if flag_logOutput:
    logString = ",".join(Detection_Info)
    logging.info(logString)
```

```
if flag_classTrigger:
    if Detection:

        #The main space to trigger actions if an animal is detected ############
        print("Trigger" , ",".join(class_Detection_Info))
        current_time = datetime.datetime.now()
        if buffer_timer < current_time:
            if object_found in allowed_objects:
                door("unlock")
                record(1,object_found,UART_counter)
                UART_counter = UART_counter + 1
                buffer_timer = datetime.datetime.now() +
datetime.timedelta(seconds=trigger_timer)
            else:
                record(1,object_found,UART_counter)
                UART_counter = UART_counter + 1
                buffer_timer = datetime.datetime.now() +
datetime.timedelta(seconds=trigger_timer)



        #The main space to trigger actions if an animal is detected ############
    else:

        print("No trigger")

if flag_display:
    cv2.imshow('Webcam', img)
    cv2.waitKey(20)
    if cv2.waitKey(1) == ord('p'):
        break
```

```
		#PIR detector
		sense = PIR()
		current_time = datetime.datetime.now()
		if PIR_buffer_timer < current_time:
			if sense == 1:
				record(2,"NA",UART_counter)
				UART_counter = UART_counter + 1
				PIR_buffer_timer = datetime.datetime.now() + datetime.timedelta(seconds=PIR_timer)



		#system check in, calls the record keeping function to report in.
		current_time = datetime.datetime.now()
		if current_time > hourly_timer:
			record(3,"NA",UART_counter)
			UART_counter = UART_counter + 1
			hourly_timer = datetime.datetime.now() + datetime.timedelta(minutes=checkin_timer)
			print("check-in")




cap.release()
cv2.destroyAllWindows()
```

Appendix F: Python Secondary Script

This partner script is to be used in conjunction with the primary script above.

Author: Scott Essery

```
import Jetson.GPIO as GPIO

import time

import datetime

import serial

GPIO.setmode(GPIO.BOARD)


##Set up the LEDs###


Rled = 31

Bled = 33

Gled = 32

GPIO.setup(Rled, GPIO.OUT, initial=GPIO.LOW)

GPIO.setup(Bled, GPIO.OUT, initial=GPIO.LOW)

GPIO.setup(Gled, GPIO.OUT, initial=GPIO.LOW)


###Set up the motor control###


lock = 7

unlock = 15

GPIO.setup(lock, GPIO.OUT, initial=GPIO.LOW)

GPIO.setup(unlock, GPIO.OUT, initial=GPIO.LOW)


#how long does the door remain unlocked?

unlock_time = 5 #time in seconds


# Set up PIR


pir = 29

GPIO.setup(pir, GPIO.IN)


#set up Battery monitor

B1_high = 19
```

```
B1_low = 21
B2_high = 23
B2_low = 24

GPIO.setup(B1_high, GPIO.IN)
GPIO.setup(B1_low, GPIO.IN)
GPIO.setup(B2_high, GPIO.IN)
GPIO.setup(B2_low, GPIO.IN)


def light(colour):
    print("controlling LED")

    if colour == "green":
        GPIO.output(Gled, GPIO.HIGH)
        GPIO.output(Bled, GPIO.LOW)
        GPIO.output(Rled, GPIO.LOW)
    if colour == "red":
        GPIO.output(Gled, GPIO.LOW)
        GPIO.output(Bled, GPIO.LOW)
        GPIO.output(Rled, GPIO.HIGH)
    if colour == "blue":
        GPIO.output(Gled, GPIO.LOW)
        GPIO.output(Bled, GPIO.HIGH)
        GPIO.output(Rled, GPIO.LOW)

def door(key):
    print("door control")
    if key == "unlock":
        light("green")
        GPIO.output(unlock,1)
        time.sleep(0.015)
        GPIO.output(unlock,0)
        time.sleep (1)
        GPIO.output(unlock,1)
        time.sleep(0.01)
```

```
        GPIO.output(unlock,0)
        time.sleep(unlock_time)
        GPIO.output(lock,1)
        time.sleep(0.015)
        GPIO.output(lock,0)
        time.sleep (1)
        GPIO.output(lock,1)
        time.sleep(0.01)
        GPIO.output(lock,0)
        light("red")
        time.sleep(1)
        light("blue")

def PIR():
    signal = GPIO.input(pir)
    return signal

def battCheck():
    battery1level = 0
    battery2level = 0
    B1H = GPIO.input(B1_high)
    B1L = GPIO.input(B1_low)
    B2H = GPIO.input(B2_high)
    B2L = GPIO.input(B2_low)
    if B1H == 1:
        battery1level = "full"
    elif B1L == 1:
        battery1level = "half"
    else:
        battery1level = "low"

    if B2H == 1:
        battery2level = "full"
    elif B2L == 1:
        battery2level = "half"
    else:
```

```
        battery2level = "low"

    return (battery1level,battery2level)


'''
Record keeping function. Call this function with 3 arguments.
Arg 1:  "1" is camera detection, "2" is PIR, "3" is System checkin
Arg 2: string from the camera. enter NA is not using type 1
Arg 3 is the UART message counter, increment numerical value.
Record captures the date and time, as well as the battery levels.
'''
def record(record_type,seen,counter):

    print("record keeping")

    battery1level, battery2level = battCheck()

    if record_type == 1:
        print("animal note")
        message = '{date:%d-%m-%Y_%H:%M:%S}'.format(date = datetime.datetime.now()) +"; " + f"{seen} detected, Batt1 {battery1level}; Batt2 {battery2level}"
        with open('onboard_report','a') as f:
            f.write(message)
            f.write("\n")
        uart_message(message,counter)

    if record_type == 2:
        print("PIR note")
        message = '{date:%d-%m-%Y_%H:%M:%S}'.format(date = datetime.datetime.now()) +"; " + f"PIR triggered, Batt1 {battery1level}; Batt2 {battery2level}"
        with open('onboard_report','a') as f:
            f.write(message)
            f.write("\n")
        uart_message(message,counter)

    if record_type == 3:
```

```
        print("system checkin note")
        message = '{date:%d-%m-%Y_%H:%M:%S}'.format(date = datetime.datetime.now()) +"; " + f"sys check, Batt1 {battery1level}; Batt2 {battery2level}"
        with open('onboard_report','a') as f:
            f.write(message)
            f.write("\n")
        uart_message(message,counter)



#UART message, takes the message from the record Function, along with the message counter and sends it to UART.
def uart_message(message,count):
    try:
        with serial.Serial(
            port="/dev/ttyTHS0",
            baudrate=9600,
            bytesize=serial.EIGHTBITS,
            parity=serial.PARITY_NONE,
            stopbits=serial.STOPBITS_ONE
        ) as serial_port:

            payload_message = f"{message} Count:{count:04d}\n"
            serial_port.write(payload_message.encode())
            print(f"Sent payload: {payload_message.strip()}")


    except serial.SerialException as e:
        print(f"Serial port error: {e}")
    except KeyboardInterrupt:
        print("Exiting Program")
    except Exception as exception_error:
        print("Error occurred. Exiting Program")
        print(f"Error: {exception_error}")
```

Appendix G: On-board report excerpt

The following is a screenshot from the text file generated by the primary script. It demonstrates how object, PIR and system check-in reports are saved, as well as demonstrate the battery levels being shown.

onboardReport.txt:

```
30-10-2024_05:50:09; sys check, Batt1 full; Batt2 full
30-10-2024_06:20:09; sys check, Batt1 full; Batt2 full
30-10-2024_06:50:09; sys check, Batt1 full; Batt2 full
30-10-2024_07:20:10; sys check, Batt1 full; Batt2 full
30-10-2024_07:50:10; sys check, Batt1 full; Batt2 full
30-10-2024_08:20:10; sys check, Batt1 full; Batt2 full
30-10-2024_08:50:10; sys check, Batt1 full; Batt2 full
30-10-2024_09:20:10; sys check, Batt1 full; Batt2 full
30-10-2024_09:50:10; sys check, Batt1 full; Batt2 full
30-10-2024_10:20:10; sys check, Batt1 full; Batt2 full
30-10-2024_10:30:39; PIR triggered, Batt1 full; Batt2 full
30-10-2024_10:32:05; PIR triggered, Batt1 full; Batt2 full
30-10-2024_10:50:11; sys check, Batt1 full; Batt2 full
30-10-2024_11:20:11; sys check, Batt1 full; Batt2 full
30-10-2024_11:50:11; sys check, Batt1 full; Batt2 full
30-10-2024_12:20:11; sys check, Batt1 full; Batt2 full
30-10-2024_12:50:11; sys check, Batt1 half; Batt2 full
30-10-2024_13:20:11; sys check, Batt1 half; Batt2 full
30-10-2024_13:50:11; sys check, Batt1 half; Batt2 full
30-10-2024_14:20:12; sys check, Batt1 half; Batt2 full
30-10-2024_14:50:12; sys check, Batt1 half; Batt2 full
30-10-2024_15:20:12; sys check, Batt1 half; Batt2 full
30-10-2024_15:50:12; sys check, Batt1 half; Batt2 full
30-10-2024_16:20:12; sys check, Batt1 half; Batt2 full
30-10-2024_16:50:13; sys check, Batt1 half; Batt2 full
30-10-2024_17:20:13; sys check, Batt1 half; Batt2 low
30-10-2024_17:20:41; PIR triggered, Batt1 half; Batt2 low
30-10-2024_17:50:13; sys check, Batt1 half; Batt2 low
30-10-2024_18:20:13; sys check, Batt1 low; Batt2 low
30-10-2024_18:46:13; PIR triggered, Batt1 low; Batt2 low
30-10-2024_18:50:14; sys check, Batt1 low; Batt2 low
30-10-2024_19:20:14; sys check, Batt1 low; Batt2 low
03-11-2024_11:51:13; toothbrush detected, Batt1 full; Batt2 full
03-11-2024_11:52:06; keyboard detected, Batt1 full; Batt2 full
03-11-2024_11:52:08; PIR triggered, Batt1 full; Batt2 full
03-11-2024_11:52:40; person detected, Batt1 full; Batt2 full
03-11-2024_11:58:44; PIR triggered, Batt1 full; Batt2 full
03-11-2024_11:59:42; PIR triggered, Batt1 full; Batt2 full
03-11-2024_12:13:45; PIR triggered, Batt1 full; Batt2 full
03-11-2024_12:20:00; sys check, Batt1 full; Batt2 full
03-11-2024_12:50:00; sys check, Batt1 full; Batt2 full
03-11-2024_13:06:33; PIR triggered, Batt1 full; Batt2 full
03-11-2024_13:09:35; PIR triggered, Batt1 full; Batt2 full
03-11-2024_13:20:00; sys check, Batt1 full; Batt2 full
03-11-2024_13:30:57; cell phone detected, Batt1 full; Batt2 full
03-11-2024_13:50:00; sys check, Batt1 full; Batt2 half
03-11-2024_14:12:35; PIR triggered, Batt1 full; Batt2 full
03-11-2024_14:17:35; PIR triggered, Batt1 full; Batt2 full
03-11-2024_14:20:01; sys check, Batt1 full; Batt2 full
03-11-2024_14:50:01; sys check, Batt1 full; Batt2 full
03-11-2024_15:20:01; sys check, Batt1 full; Batt2 full
03-11-2024_15:50:02; sys check, Batt1 full; Batt2 full
03-11-2024_16:20:02; sys check, Batt1 full; Batt2 full
03-11-2024_16:43:45; PIR triggered, Batt1 full; Batt2 full
03-11-2024_16:50:02; sys check, Batt1 full; Batt2 full
```