Automated Pet Door with LED Collar

Daniel Hemmerde, Davis Rozel, and Christopher Vanle

University of Central Florida, Department of Electrical and Computer Engineering and College of Optics and Photonics, Orlando, Florida, 32816, U.S.A.

***Abstract* — Automation has grown and been implemented to complement more of our daily lives to aid in the relationship between owner and pet, such as automated pet feeders and toys. Our project centers our attention on an automated doorway to come back into the house after a long day of playing and exploring outside. With the use of optics and electronics, this door aims to use the pet's collar to act as a key that will activate a sliding door. This will only allow the pet in while keeping out unwanted rodents or other intruders.**

***Index Terms* — motion detection, DC motors, power conversion, animals.**

I. Introduction

Automated pet accessories have become the go-to for pet owners when it comes to caring for their loved ones. Now a days, there are many price point options for automated pet accessories, such as an automated pet feeder. Taking advantage of microchips many owners implant into their pets is one popular option that many accessories utilize, but these require the added cost of surgery to microchip their pet. Other products will use RFID chips that are sometimes sold separately from the product. Our project focuses on automated pet doors while using an infrared LED on the pet's collar to act as a key to open the door, rather than relying on the microchip and includes the LED collar. The product aims to assist pet owners whose pets like to go outside often. The IR LED on the collar utilizes binary outputs to act as the unique key for each pet. The core idea of using an IR LED was inspired by the technology behind TV remotes which also utilizes binary outputs. This technology doubles as a security feature, which is discussed later. Once the key is recognized, a motor will bring a door in the frame upwards, opening the entry way for the pet to enter through. This design aims to give pet owners the relief and satisfaction that their pet can make it back home safely, especially when the pet owner is not available to let their loved ones back in the house.

II. System Components

Our design is composed of several subsystems from different disciplines that come together to make up our unique automated pet door. This section will briefly explain each subsystem used for our design.

A. Microcontroller

Our system utilizes two microcontrollers, one for the main board in the door frame, and a smaller microcontroller for the pet collar. For the door frame, the main board houses an ESP-WROOM-32D microcontroller due to its programming capabilities. Although a bit more expensive than other options, this microcontroller gives the most flexibility for our use case with enough flash memory to cover our needs. For the collar, our design utilizes the ATtiny85. This option has the lowest idle current to help us conserve battery on the collar while having the most RAM compared to our other options. With these two, our goal of creating a unique key using binary outputs to communicate between the collar and door frame can be accomplished.

B. Motor

The main mechanism of our design is a motor used to bring our door for the entry way up and down. Between a DC motor and an AC motor, our design utilizes a DC motor since we only need it to lift a 9" x 9" x ¼" acrylic door no more than 9 inches high. The DC motor is the more power efficient option since our load is constant and not heavy, while it is also very cost effective.

C. Power Supply

Usually, the easiest solution would be to plug it straight into the wall outlet. But since our design is meant to fit at the front or back door of a house, running a wire to an outlet is not always an option. Taking advantage of the outside exposure, we opted to use solar panels as our main power supply to charge our batteries for the door frame. Ideally, the door will not operate often enough to drain the battery of our automated pet door. So, the energy gathered from the solar panels should be enough even if there are little sun hours throughout the day. For days with no sun hours, our design will still include a wall outlet option to charge the batteries.

D. Power Manager

Since our design will have two power sources, a power manager is necessary not to overload our batteries if both inputs are combined. The consumer can control the pin out voltage of the MOSFETs, allowing the consumer to choose which energy input they prefer.

E. Motion Sensor

To help conserve battery for our door frame, our design only starts up when our motion sensor detects something in front of it. This allows our main microcontroller to stay in an idle state; thus, saving energy. Our design uses an active infrared motion sensor since it has a better range than a passive infrared motion sensor. Of course, our motion sensor will pick up everything that steps in front of the door and turn on our main microcontroller. But this is why a unique key is necessary to avoid unwanted guests such as rodents.

F. Occupancy Sensor

To ensure that the pet is not in danger while entering through the pet door, an occupancy sensor consisting of IR LEDs and photodiodes are used to tell the main microcontroller if there is something in the doorway. To put it simply, if an animal is blocking the path of the light from the IR LED to the photodiode, there is something in the doorway and our door should not close. If a pet decides to sit in the doorway for an extended period, the owner should have the ease of mind that the door will not close on top of their pet.

G. Infrared Communication

Our collar and door frame will communicate using the NEC Communication Protocol which is widely used and supported. In addition, our design will use a similar encryption scheme as discussed in "Eavesdropping Vulnerability and Countermeasure in Infrared Communication for IoT Devices" [1]. This is necessary to create a unique key for our pet collars to avoid unwanted guests from coming through the doorway and any person tampering with our door if they happen to have a light source operating at the same wavelength as our photodiodes.

To avoid any further complications if a single subsystem were to fail, each subsystem has its own dedicated PCB. This allows us to make any changes to each subsystem without the stress of having to re-order and re-solder one huge PCB.

III. System Concept

It is necessary to discuss the operation of our design, from the operation of the door frame to the collar communicating to the door frame. The flowchart of the door frame is shown in Fig. 1, and the flowchart of the collar in Fig. 2.

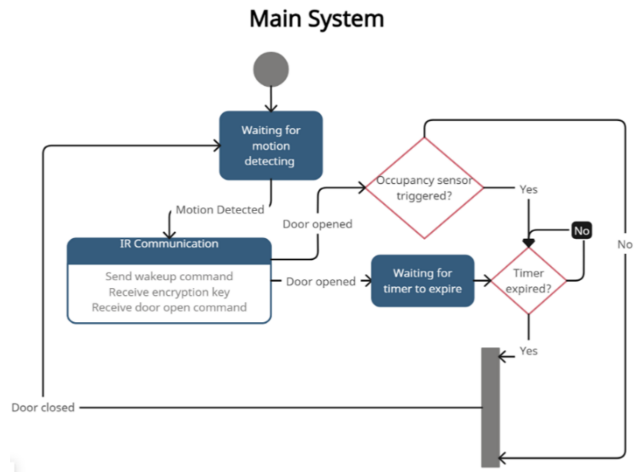

Fig. 1. Operation flowchart of door, motion sensor, collar communication, and occupancy sensor.

As seen in the main system flowchart, the main microcontroller will turn on once the motion sensor detection is triggered. This will prompt communication between the pet collar and door frame and will open the door once conditions are satisfied. Followed by the door closing after a set time expires while opened or after the occupancy sensor is triggered.

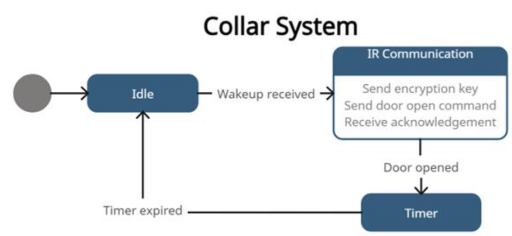

Fig. 2. Operation flowchart of IR communication from collar to door frame.

As shown in the collar flowchart, the collar is in an idle state while the pet is outside until it is woken up when in proximity to the door frame. An encrypted key is sent between the collar and door frame, then the collar will return to its idle state after a set time.

The automated pet door begins operation once the motion sensor detection is triggered. Range for the motion sensor is very important for the pet door to work properly, otherwise the pet door will only turn on when the pet is very close to the door frame. Although the range of our active infrared motion sensors is poor, the range is improved using optics.

After detection, the main microcontroller awakens from its idle state and begins communication with the pet collar. The pet collar's microcontroller will awaken from its idle state and a randomized communication protocol is sent from the main microcontroller to the collar's microcontroller. The protocol is sent back to the main microcontroller as a means of verifying who is trying to access the pet door, in this case the owner's pet wearing the integrated collar.

The IR communication between the collar and door frame is the most crucial aspect of our design. As mentioned before, the NEC Communication Protocol is used to create unique keys for the collar and doubles as a security feature for pet owners. Without it, anyone who happens to own a light source that emits the same wavelength necessary for the photodiodes is able to open the door frame. They can simply shine the light source into the door frame's photodiodes that are meant for the IR light coming from the pet collar. Only the pet collar that programmed with that specific key can activate the door of the door frame. Since each key is unique, multiple collars can be made so that more than one pet can take advantage of the automated pet door.

After verification, the door is pulled upwards by the DC motor and opens the doorway for the pet. A countdown starts as soon as the door is at its peak and will close once the timer runs out. This timer is paused when the occupancy sensor in the entry way is blocked, notifying the main microcontroller that the pet is walking through the doorway. Once the occupancy sensor is no longer blocked, the timer is reset. After the countdown reaches zero, the door will then close. Without the occupancy sensor, the door might close on the pet, which is a huge safety concern. Some pets are reluctant to pass through the doorway immediately after opening, so a timer and occupancy sensor are necessary to give the pet enough time to make their decision.

This concept can be used to allow the pet inside the house or out of the house, but our design will only focus on letting the pet back into the house. Although having both features gives additional safety for the pets in rare events the house is on fire and the pet needs to escape.

A. System Hardware

After discussing how the overall idea of the design works, we need to discuss how all the hardware is integrated and how each communicates to one another. The collar is more than just the microcontroller and IR LED, a battery is needed. A block diagram is used to showcase how each major hardware component is implemented in our design, as shown in Fig. 3.

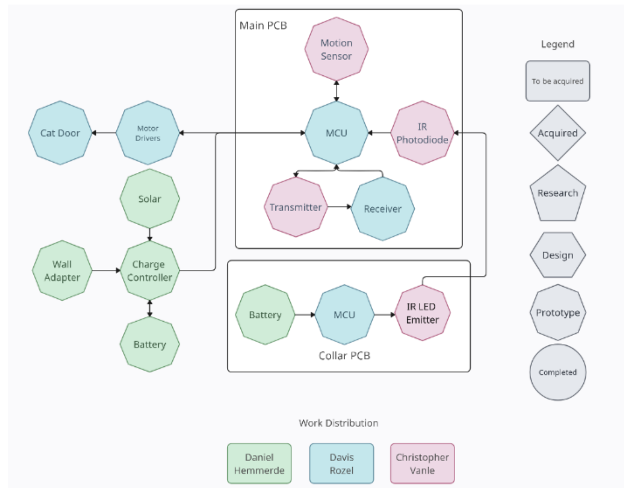

Fig. 3. Block diagram showing major hardware components and team member responsible for its integration.

Although only the main and collar PCB are highlighted in Fig. 3, each subsystem is housed on its own dedicated PCB to make troubleshooting a lot easier, as mentioned before.

IV. Hardware Detail

Although the main components were discussed previously, we need to go in detail with the requirements of each main component used for our design.

A. Main and Collar Microcontrollers

On the main PCB, the microcontroller used is the ESP-WROOM-32D. Since the main microcontroller is responsible for most of the major operations of the system, it must meet the necessary specifications and requirements of the components of our system. These specifications include:

(1) Pin count – enough to connect all the components of the design plus extra for flexibility.
(2) Clock speed – faster clock speed results in faster processing, but also increases power consumption.
(3) Storage – size of storage to handle the size of our code and rate of data transfer to improve response time.

For the collar PCB, the microcontroller used is the ATtiny85. In addition to the requirements for the main microcontroller, the collar microcontroller must also satisfy specifications such as:

(1) Size – it should not be too big since it is hung off a pet's collar.
(2) Weight – cannot weigh too much since it is on a pet's collar.

B. Door Power Supply

The main source of power for our pet door is solar. We are using a solar panel manufactured by FellDen that provides the ideal power output to cost ratio that we need for our design. Factors that were considered for our solar panels were:

(1) Initial Cost – cannot be expensive due to our budget.

(2) Reliability and maintenance – last for a very long time without the need to replace or maintain often, at most cleaning the panels.

(3) Efficiency – provides a constant charge on a sunny day and is enough to provide enough energy for the short burst of energy from the door operating.

(4) Size – mounted on the door frame, they cannot be too large.

For the secondary source of power, we went with a power barrel power adapter produced by N\C. At the lowest cost while supplying the highest maximum current compared to other options makes it the ideal power adapter for our pet door. As such, the only factors we were concerned about for a secondary source of power were:

(1) Compatibility – adapter choice.

(2) Cost – should be a cheap and easily replaceable option.

Of course, the door will need batteries since solar power is our main source of power. Thus, we are using Lithium-Ion batteries for the door frame as it has the highest storage to cost ratio compared to other battery options. They also require the least amount of maintenance and will last for a long time. The important factors we must consider are:

(1) Lifespan – survive long periods where it will not receive a charge.

(2) Safety – although enclosed in the frame, will have to endure the outside temperatures.

(3) Cost – due to budget.

C. Collar Power Supply

In addition to the requirements necessary for the door battery, the collar battery will also need to consider:

(1) Weight – light enough for a pet to carry on a collar.

(2) Safety – due to outside exposure and is worn by the pet.

The battery for the collar needs to be lightweight since an animal is carrying it around, while also being the least likely to cause any fire hazards for the animal. Our collars will use Lithium Polymer batteries due to their high charge density and considerably low weight. They also have the same charge patterns as the Lithium-Ion batteries of the door frame; thus, the same charging circuits can be used to charge both batteries if necessary. The housing for the pet collar must be durable to prevent anything happening to the collar since Lithium Polymer batteries can catch fire if punctured, so the housing quality must be accounted for. The last thing we want to do is to potentially harm the animal.

D. Battery Charge System

To prevent the two input sources of power for the door from combining, our design utilizes a charging system consisting of three N channel MOSFETs that will only allow input from one source at a time. For the circuit, NMOS 2 and 3 are controlled by a pin out voltage from the microcontroller. NMOS 1 inverts the effect of NMOS 3, creating the effect of only NMOS 1 or 2 being active at a time. This halts all forward current from one of the two sources of power while the other works at full capacity. A Schottky diode is placed at the mouth of the solar input and wall input to ensure that the forward voltage drop is minor, resulting in less power loss. As an added feature, the voltage that controls the MOSFETs can be controlled by the consumer, allowing them to pick which input they prefer to use. The circuit can be seen on Fig. 4.

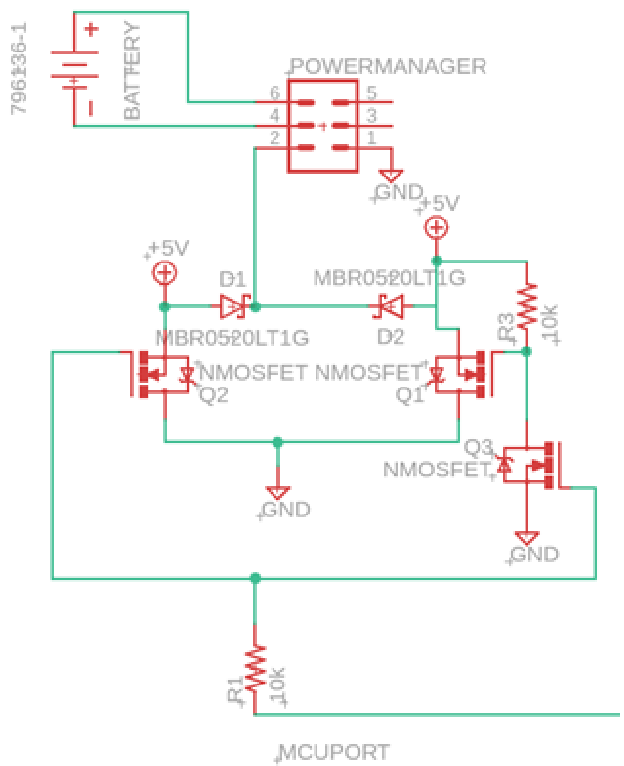

Fig. 4. Battery charging system circuit.

E. Motor

The DC motor was also found on Amazon from CHANCS and is a means of pulling the pet door up and down in the door frame. The door itself is acrylic, so the load is constant and does not weigh a significant amount. Thus, the DC motor is perfect for the job. Some factors we considered are:

(1) Speed – able to pull the door up quickly.

(2) Torque – high enough to be able to pull our acrylic door.

(3) Cost – fits in our budget without overspending.

The CHANCS motor had the highest torque compared to the rest with a suitable RPM of 60 and the lowest cost when shipping cost is considered. Factors such as voltage and shaft length can be customized to fit our desired mechanical needs.

F. Motor Driver

To control the speed and direction of our motor's rotation, our design needs a motor driver consisting of an H-bridge made up of four MOSFETs. This controls the direction of the current based on which MOSFETs are on and off. A PWM signal from the microcontroller is used to change the time each MOSFET is on, allowing us to change the voltage going to the motor for speed control. Our circuit consists of a DRV8220 that can supply more than 12 V and 0.33 A to our motor. The circuit schematic can be seen on Fig. 5.

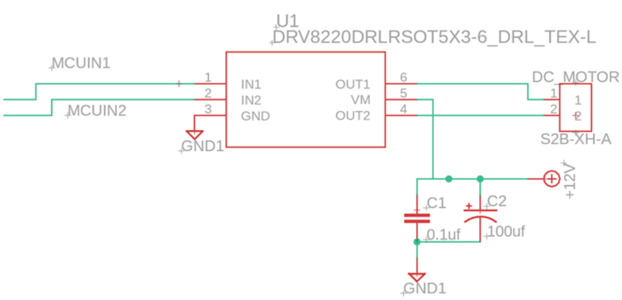

Fig. 5. Motor driver circuit schematic.

G. Motion Sensor

We will use an active infrared motion sensor rather than a passive infrared motion sensor. This requires the use of IR LEDs and photodiodes which were found on Amazon from Gikfun. They operate at 940nm, which is common for most cheap IR LEDs found on the market. Some important factors to consider for the motion sensor include:

(1) Voltage – operates at less than 3V per LED.

(2) Detection Range – should be able to detect an object up to 1 meter away.

In testing, we found these IR LEDs and photodiode pairs have a very small detection range when relying on the IR light that is reflected off an object, in this case, a pet with fur. So, the use of optics is implemented to improve the range of our motion sensor. A Fresnel lens is used to collimate the light coming from the IR LED to avoid losing any light from how the light is spread due to the shape of the LED. Motion sensor circuits are simple and can be easily replaced or fixed if any problems were to occur.

H. Occupancy Sensor

As mentioned before, the occupancy is necessary to ensure the safety of the pet as they walk through the doorway. To keep our design simple and cost efficient, we considered the same factors as the motion sensor when it comes to voltage and detection range. Since the occupancy sensor is in the entry way and the pair is pointed directly at each other, its short range is not an issue. Through testing, we found that the pair pointed directly at each other can still be detected up to 1 meter apart. As such, we will also use the IR LED and photodiode pairs from Gikfun found on Amazon. The IR LEDs and photodiodes can also be used to detect if the door is open or closed, which is helpful for the microcontroller.

I. Door Frame and Door Materials

For the door frame, it is important to pick a material that can meet our requirements:

(1) Cheap – our design should compete with affordable products already on the market.

(2) Durable – must withstand the outside environment and changes in weather conditions.

For the door frame, we went with plywood since it was given to us for free and can withstand the outside conditions as mentioned above. 3D printing the door frame is possible with the right equipment. But due to our limited printing surface and number of printers available to us, we decided not to go the 3D printing route. For mass production, it is possible to outsource the printing.

For the door that is pulled up and down, our main requirements include:

(1) Weatherproofing – since it is exposed to the outside environment.

(2) Durable – can withstand moderate physical force to avoid break-ins.

(3) Lightweight – so that our motor can easily pull and hold the weight.

Finding the balance between strength and weight for our door is important to give the owner the peace of mind that their pet door will not be a fault in their home security. For this reason, we found the acrylic door to be the most suitable for our design. The acrylic door has the best average tensile strength compared to the other materials and can withstand all weather conditions. As a bonus, it is not susceptible to yellowing due to UV exposure, unlike materials such as polycarbonate.

V. Infrared Communication

As mentioned before, the collar and door will communicate with each other using infrared communication, like how TV remotes communicate with the TV. This is done by pulsing an IR LED on and off quickly to communicate data to an infrared photodiode. This method was chosen since it is very cheap, only requiring an IR LED and photodiode, and quite power efficient. IR communication normally lacks security, which is why it's necessary to use the NEC Communication Protocol along with using an encryption key.

A. XOR Operation

An encryption key is generated based on the count of the timer on the collar microcontroller. In addition to the encryption key, there is a counter counting by 1 every time a command is transmitted. By adding the encryption key with the counter count followed by doing an XOR operation with the key + counter and the command, we create an encrypted packet. This encrypted packet is sent from the collar to the door and the same method is done in reverse to decrypt the packet. This method makes our system more impervious to foreign control, allowing only the collar and door to communicate with one another. More information on using XOR operations to create encryption keys can be found in [1].

B. NEC Communication Protocol

This section will discuss in detail how the NEC Communication Protocol functions. This standard was developed by NEC (now Renesas) using pulse distance encoding to transmit bits. A logical '0' is made up of a 562.5 µs pulse followed by a 562.5 µs space, totaling 1.125 ms. Then a logical '1' is made up of a 562.5 µs pulse followed by a 1.6875 ms space, totaling 2.25 ms. The protocol operates using a carrier frequency of 38 kHz, with modulation and demodulation being the most effective at a carrier frequency of 38.22 kHz. This protocol transmits messages in 32-bit frames, with 16-bits for the address field and the other 16-bits for the command field. The address field is occupied by the address of the device, which is our door frame. A typical pulse train is shown on Fig. 6.

Fig. 6. Typical NEC Communication Frame, with permission from SB-Projects by San Bergmans.

The pulse train shown in Fig. 6 starts with a 9 ms pulse burst to show the start of a message, followed by a 4.5 ms space. Next, the address of the device is transmitted followed by the inverse of the address immediately after. Then the command and inverse of the command are transmitted. Lastly, a 562.5 µs burst ends the pulse train to signify the end of the message. There are two reasons why the invert address and command are transmitted. The first reason is that the address can be verified and checked for any errors by comparing the non-inverted and inverted message. The second reason is that each message frame is the same length in time, with the same amount of 1's and 0's transmitted, the total message frame adds to 67.5 ms (excluding the last 562.5 µs bursts that signifies the end of the message).

The NEC protocol supports a repeat command. This tells the receiver to repeat the previous command. It is comprised of a 9 ms pulse burst followed by a 2.5 ms space. This repeat command is used in IR TV remotes when a button is held down to repeat the command every 110 ms and continuously transmits if the button is held down. The repeat command frame can be seen in Fig. 7.

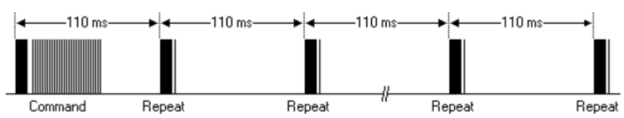

Fig. 7. NEC repeat command frame, with permission from SB-Projects by San Bergmans.

An extended form of the NEC protocol can be used to transmit more addresses by using the inverted address frame to transmit another 8 bits of address, totaling a 16-bit address. This extends the possible addresses from 256 values to 65,536 possible values. One major issue that comes from this method is that the address can no longer be verified by comparing the non-inverted address to the inverted address like before. The second major issue is that the message frame is no longer a constant time but can be fixed by making sure the amount of 0's and 1's transmitted in the address are the same. Although, this ends up reducing the number of possible addresses to 13,000. An extended protocol frame can be seen in Fig. 8.

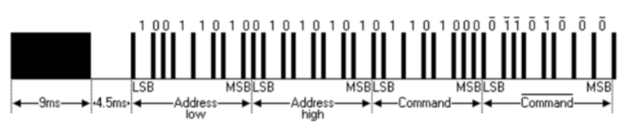

Fig. 8. Extended NEC protocol frame, with permission from SB-Projects by San Bergmans.

The process of generating a random encryption key is based off a timer that counts by 1 every time the wake-up command on the collar is received by the door. The key is then transmitted to the door system so that both parties have the same encryption key.

For each transmission sent, the command is XOR’d with the encryption key. The receiver decrypts the command by doing another XOR operation using the encryption key and the 8-bit command field to extract the command. An 8-bit counter is used to provide additional security. The counter value that is increased every time a command is sent is placed in the address field of the transmitted packet, specifically in the inverted address field using the extended NEC protocol. The encryption key and counter are added together to create a new 8-bit key that is XOR’d with the command.

During decryption, the counter is extracted from the address field, then added to the previously received encryption key. The door system then extracts the command field and performs an XOR operations with the 8-bit command and the new key. A transmission frame of a transmitted encryption key and command from collar to door is seen in Fig. 9 and 10.

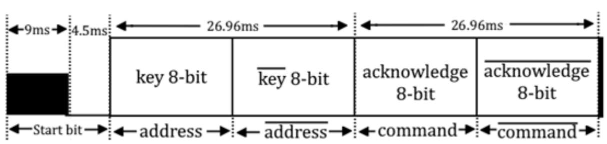

Fig. 9. Transmission frame of sending encryption key from collar to door.

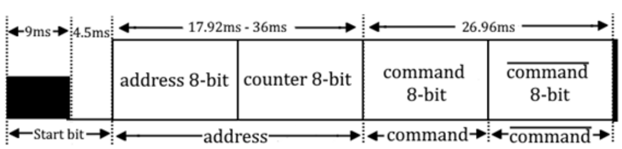

Fig. 10. Transmission frame of sending command from collar to door.

To show how an encrypted command is transmitted, a simplified code is shown below to demonstrate the encryption method.

```
// Encryption key is only generated once after collar wakes
up, not for every
transmission
enc_key = rand(timer_value);
counter = 8-bit integer
address = 8-bit device address;
command = 8-bit command;
key = enc_key + counter;
enc_command = key ^ command;
// Create 32-bit data frame according to NEC protocol
16-bit_address = (address << 8) + counter;
16-bit_command = (enc_command << 8) +
~enc_command;
32-bit_message = (16-bit_address << 16) + 16-
bit_command;
// Send message via IR communication
IR_send(32-bit_message);
counter++;
```

VI. Software Detail

To program our microcontrollers, we chose a language that is not power or performance intensive while still capable of carrying out all the operations necessary for our components. For this design, we use C++ to program our microcontrollers.

This language is widely supported when it comes to programming microcontrollers. It is a high-level language and compiles closer to machine code in comparison to other high-level languages. We use Arduino IDE because it gives us access to many public libraries that are widely supported.

A. Class Diagram

Our system is broken into six classes:

(1) door controller
(2) collar
(3) charge controller
(4) motion sensor
(5) occupancy sensor
(6) motor

The IR transmitters are included within the collar and door controller classes since both systems contain IR transmitters. The door controller is the main element of our system, communicating with every element of our system; thus, it is connected to every class.

The controller communicates with the motor to tell when to open or close the door. The motor communicates with the door controller to tell whether the door is open. The motor also tells the current position of the door and how much it needs to rotate to get into position, as well as detecting any resistance on the door determined by the current running through the motor. The occupancy sensor tells the controller if the door is blocked, and the controller tells the occupancy sensor when it should be on or off. The motion sensor can detect if there is anything in front of it and returns this as a Boolean value which communicates with the controller and wakes up the controller if true. The charge controller informs the charge of the system to the controller. That information is used to switch the source of power for the system based on the charge amount. Finally, the collar is woken up by the door controller through an IR signal, changing the power mode of the collar from idle to active. The door controller and collar communicate with each other through NEC Communication Protocols, sending encryption keys and door opening commands.

The class diagram of all six classes and their connections is shown on Fig. 11 below.

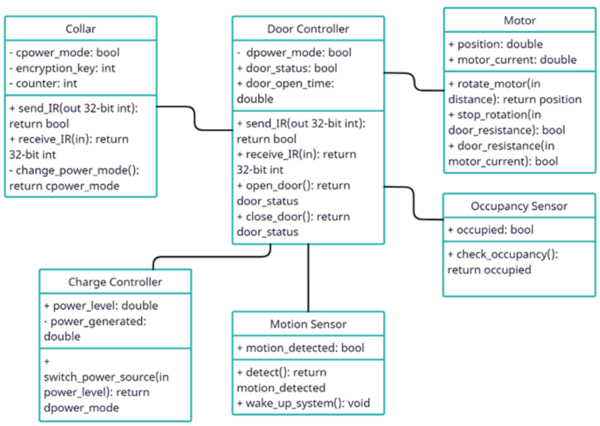

Fig. 11. Class diagrams and communication to each other.

VII. Conclusion

Although voluntary, our design follows standards to account for safety. IEC 61730-1:2023 accounts for safety requirements of solar panels and 10 CFR Part 430 covers the safety and regulations of the battery charger. UL 2592 covers the regulations for proper protection for wires coming from LEDs. ANSI/IES LM-80-21 covers the methods of measuring LED luminous, photon flux, and color maintenance.

Biography

Daniel Hemmerde is an Electrical Engineering student in the power and renewable energy track at the University of Central Florida. He is a senior and plans to graduate in December 2024. Daniel hopes to pursue a career in the nuclear or renewable energy generation fields.

Davis Rozel is currently a senior attending the University of Central Florida, majoring in Electrical Engineering. He intends to receive an undergraduate degree in December of 2024. Davis' future goals are to secure a career in the embedded systems field.

Christopher Vanle is an undergraduate Photonics and Optics Engineering student at the University of Central Florida. He intends to graduate in December 2024. Christopher plans to pursue a career in opto-mechanical design. He currently works as an Optical R&D Intern at Lightpath Technologies.

Acknowledgement

The authors wish to acknowledge the assistance and support of Dr. Suboh Suboh, Dr. Stephen Kuebler, and Dr. Weeks of the University of Central Florida. Including the assistance of Jake Carr of Lightpath Technologies. And San Bergmans for the use of his graphics on NEC Communication Protocols [2].

References

[1] Kim M, Suh T. Eavesdropping Vulnerability and Countermeasure in Infrared Communication for IoT Devices. Sensors (Basel). 2021 Dec 8;21(24):8207. doi: 10.3390/s21248207. PMID: 34960299; PMCID: PMC8706134.

[2] NEC Communication Protocol figures from https://www.sbprojects.net/knowledge/ir/nec.php