**Design Project User and Product Manual**

**B.R.U.C.E. USER MANUAL**

Submitted by:

Screws Loose, C9

Amin Khaled, 300351693

Mohammad Salem 300375174

Hossam Alshaiba 300269076

Waleed Sakalla 300375933

J’afar Assaf 300354072

GNG 1103

December 8th, 2024

University of Ottawa

**Table of Contents**

[**1. Introduction 6**](#_heading=h.4d34og8)

[**2. Overview 6**](#_heading=h.17dp8vu)

[2.1 Conventions 9](#_heading=h.3rdcrjn)

[2.2 Cautions & Warnings 9](#_heading=h.26in1rg)

[**3. Getting Started 11**](#_heading=h.lnxbz9)

[3.1 Configuration Considerations 11](#_heading=h.35nkun2)

[Tools Needed 11](#_heading=h.mkb1d4l9xk17)

[Connections 11](#_heading=h.dcht2betuekr)

[3.2 User Access Considerations 12](#_heading=h.1ksv4uv)

[3.3 Accessing/setting up the System 13](#_heading=h.44sinio)

[3.4 System Organization & Navigation 18](#_heading=h.2jxsxqh)

[3.4.1 Main Component: Arduino MKR Zero 18](#_heading=h.qmthaoepol1v)

[3.4.2 Infrared (IR) Break Beam Sensors 18](#_heading=h.ikbr5nbsj9hx)

[3.4.3 Temperature and Humidity Sensor (DHT11) 18](#_heading=h.8orfg1hgqj2q)

[3.4.4 Data Storage: MicroSD Card 18](#_heading=h.985jx9yr1x85)

[3.4.5 Wireless Communication: Bluetooth Module (HC-05) 19](#_heading=h.cih03uoddbrc)

[3.4.6 Enclosure 19](#_heading=h.tet084ml09m8)

[3.4.7 Power Supply 19](#_heading=h.4n1lsifmvs0h)

[3.5 Exiting the System 19](#_heading=h.z337ya)

[**4. Using the System 20**](#_heading=h.3j2qqm3)

[4.1 Setting Up The System 20](#_heading=h.4i7ojhp)

[4.2 Tracking Bat Activity 20](#_heading=h.c5j4edml5pkk)

[4.3 Accessing Stored Data 21](#_heading=h.bg0c2iubb0eg)

[**5. Troubleshooting & Support 27**](#_heading=h.1ci93xb)

[5.1 Error Messages or Behaviors 27](#_heading=h.3whwml4)

[5.2 Special Considerations 27](#_heading=h.2bn6wsx)

[5.3 Maintenance 28](#_heading=h.qsh70q)

[5.4 Support 28](#_heading=h.3as4poj)

[**6. Product Documentation 29**](#_heading=h.1pxezwc)

[6.1 Mechanical Design 29](#_heading=h.49x2ik5)

[6.1.1 BOM (Bill of Materials) 29](#_heading=h.2p2csry)

[6.1.2 Equipment list 29](#_heading=h.147n2zr)

[6.2 Electronics and Power 29](#_heading=h.fvswgnjmu4ui)

[6.2.1 BOM (Bill of Materials) 29](#_heading=h.a1ltg18u5ybh)

[6.2.2 Equipment list 30](#_heading=h.arj7bwry6sam)

[Assembly and Prototyping Equipment 30](#_heading=h.8xcqfhj3rfd4)

[Prototyping Tools 30](#_heading=h.bj1o2al7vueo)

[Testing and Measurement Equipment 30](#_heading=h.67se7kwd5rui)

[Programming and Debugging Tools 30](#_heading=h.vg07x351u955)

[Power Supply and Debugging 30](#_heading=h.qk0wqeon65cs)

[Enclosure and Mounting Tools 31](#_heading=h.msmvqxaasy4w)

[6.3 Software 31](#_heading=h.7u9ebi3i1fec)

[6.3.1 BOM (Bill of Materials) 31](#_heading=h.f0fiqz90fafn)

[6.3.2 Equipment list 31](#_heading=h.by41b0earhw)

[6.3.3 Instructions 32](#_heading=h.3ewskqnresyf)

[6.4 Testing & Validation 44](#_heading=h.23ckvvd)

[1. Core Function: Sensor Performance 44](#_heading=h.n3fdeg9u5vgw)

[2. Data Processing and Storage 44](#_heading=h.wk1ke2ncn9sa)

[3. System Durability and Enclosure Integrity 45](#_heading=h.otn4melnqwu9)

[4. Cable Management and Ease of Maintenance 45](#_heading=h.m3jh47drqqe6)

[6. Special Requirements for Sustained Usage 45](#_heading=h.wbyr072cjdei)

[**7. Conclusions and Recommendations for Future Work 47**](#_heading=h.ihv636)

[**APPENDIX I: Design Files 48**](#_heading=h.41mghml)

**List of Figures**

**Figure 1.0**: Final Prototype 7

**Figure 2.0**: Block Diagram of how B.R.U.C.E. functions 9

**Figure 3.0**: Enclosure Mounted on the Side of the Bat Box 13

**Figure 4.0**: Close-up of the bottom of the bat box 14

**Figure 5.0**: The interior of the enclosure showing the Arduino MKR Zero with the battery connected 15

**Figure 6.0**: Showcasing the on state for the switch 16

**Figure 7.0**: Showcasing the off state for the switch 16

**Figure 8.0**: Screenshot of the application logo 17

**Figure 9.0**: Inside the application 17

**Figure 10.0**: Inside the Enclosure 21

**Figure 11.0**: Highlighting the Switch and BT module 22

**Figure 12.0**: BT module LED 22

**Figure 13.0**: BLE Terminal Scanning for Devices 23

**Figure 14.0**: Empty BLE Terminal 24

**Figure 15.0**: Highlighting the Yellow Button 24

**Figure 16.0**: Active BLE Terminal Running Code 25

**Figure 17.0**: Github Repository 31

**Figure 18.0**: Navigation to Final.ino File 31

**Figure 19.0**: Viewing the Source Code 32

**Figure 20.0**: Copy Button in GitHub 32

**Figure 21.0**: Creating a New Sketch in Arduino IDE 33

**Figure 22.0**: Overwriting Template Code 33

**Figure 23.0**: Library Manager in Arduino IDE 34

**Figure 24.0**: Installing the DHT Sensor Library 35

**Figure 25.0**: Install All Confirmation in Arduino IDE 36

**Figure 26.0**: Installation Success for DHT Library 36

**Figure 27.0**: Installing RTCZero Library 37

**Figure 28.0**: Installation Success for RTCZero Library 37

**Figure 29.0**: Installing SD Library 38

**Figure 30.0**: Installation Success for SD Library 38

**Figure 31.0**: Selecting Arduino MKR Zero Board 39

**Figure 32.0**: Arduino MKR Zero Connection Confirmation 39

**Figure 33.0**: Uploading Code to Arduino MKR Zero 40

**Figure 34.0**:Upload Success Message in Arduino IDE40

**List of Tables**

[Table 1. Acronyms](#_heading=h.3dy6vkm) 07

[Table 2. Mechanical Parts](#_heading=h.2grqrue) 27

Table 3. Electronic Components 29

Table 4. Software Components 30

**List of Acronyms**

**Table 1. Acronyms**

| **Acronym** | **Definition** |
| --- | --- |
| B.R.U.C.E | Bat Reporting Universal Counting Enclosure |
| BT | Bluetooth |
| BLE | Bluetooth Low Energy |
| IR | Infrared |

# Introduction

This User and Product Manual (UPM) serves as a comprehensive guide for the use, maintenance, and potential improvement of the B.R.U.C.E.. Designed to monitor bat activity, the system aims to support conservation efforts for endangered bat species. This manual assumes that the device will be used in field conditions where bats are active, and under typical environmental factors such as varying weather or light conditions. The UPM is organized into subsections to provide all the necessary information for seamless use by non-engineers, while also offering essential details for those looking to recreate or modify the device. It includes a walkthrough for setup and operation, safety and maintenance instructions, troubleshooting tips, and detailed documentation of the prototype's construction.

# Overview

Bats are essential for maintaining ecosystems by controlling pests, pollinating plants, and dispersing seeds. However, many bat species, especially in Ontario, are endangered due to habitat loss, climate change, and other environmental challenges. To support conservation efforts, bat boxes—artificial structures designed to serve as roosting sites—are being installed to provide safe habitats for bats. Effective monitoring of bat activity is crucial to evaluate the success of these boxes. Without accurate data on their usage, it is difficult to determine if bat boxes are helping to prevent the decline in bat populations or if their design needs improvement.

The bat monitoring device, B.R.U.C.E, is designed to meet functional and practical needs for effective bat conservation efforts. It accurately counts bats entering and leaving the bat box while ensuring that the sensors are non-intrusive. The device is user-friendly, requiring minimal maintenance with a one-month battery life to ensure continuous operation. Data will be securely stored on-site on an SD card, and Bluetooth functionality will be enabled to facilitate easy access to the data. The internal conditions of the bat box will also be monitored to provide insights into habitat suitability. The design prioritizes sustainability by using environmentally friendly materials and keeping production costs low, ensuring accessibility and ease of installation for a wide range of users.

B.R.U.C.E. stands out from other competitors due to its universal and battery powered design, making it compatible with various locations and types of bat boxes. The device also allows for quick and easy data retrieval, which ensures efficient access to information. The device's high accuracy allows our device to be more reliable than other products. The temperature and humidity sensor provided precise readings with a margin of ±2℃ for temperature and ±5% for humidity. The IR sensors have a 96% accuracy rate for detecting bat movements (i.e. bat entrance and exit). These features make B.R.U.C.E a reliable and user-friendly device for monitoring bat activity effectively.

**Figure 1.0**

*Final Prototype*

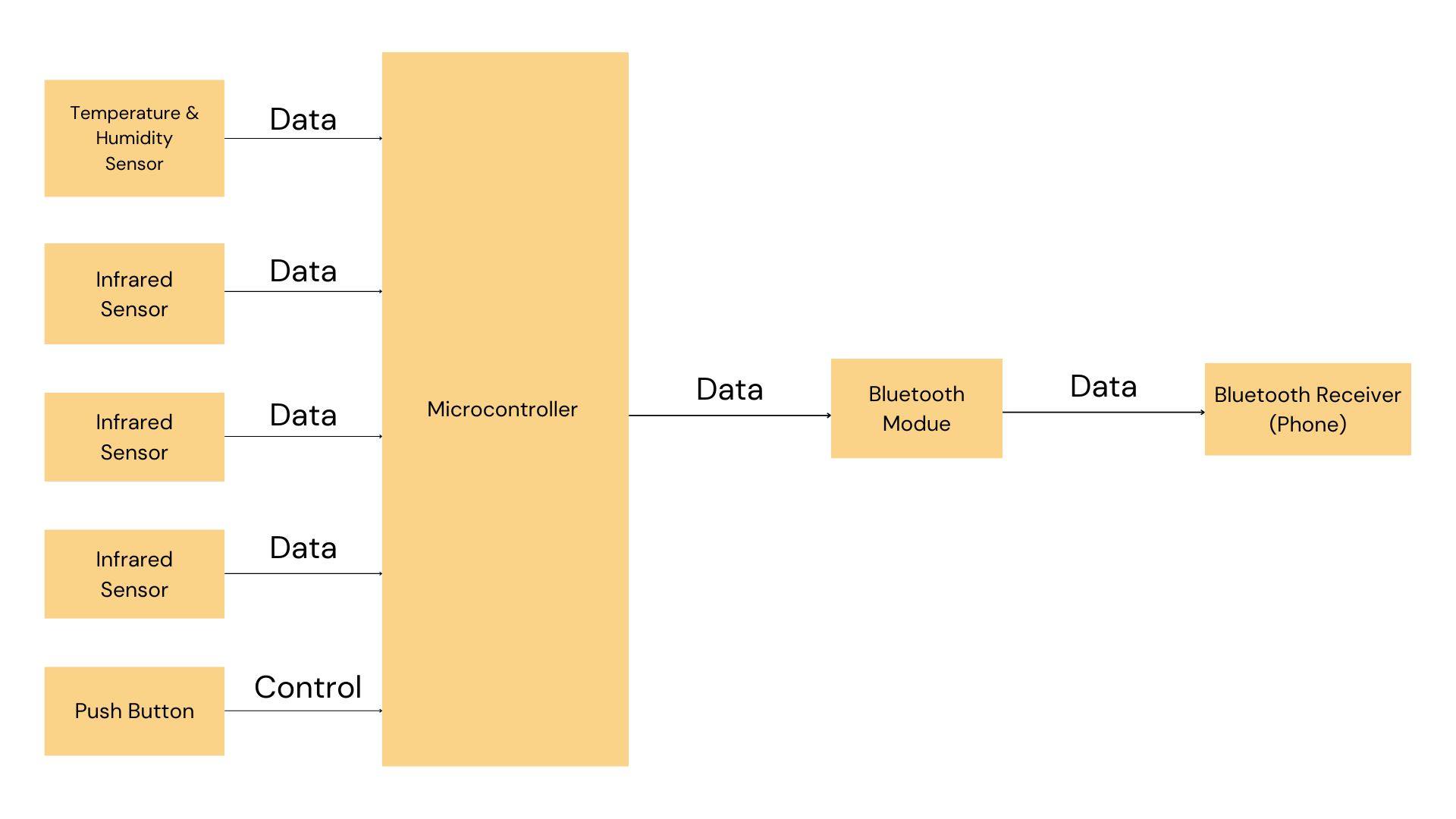
*Note.* The bat box is not a part of the final prototype. The bat box is there to show a representation of the full system.

B.R.U.C.E. can accurately count bats entering and leaving the bat box while not obstructing the bats natural movements. The device is user-friendly, and requires minimal maintenance due to its one-month battery life, which ensures continuous operation. The internal temperature and humidity of the bat box are also monitored, which can provide an insight into habitat suitability. The data is securely stored on a microSD card, which can then be transmitted through a bluetooth module to an online bluetooth terminal or through an app on the user's personal device (i.e. mobile phone, laptop, etc.).

The system uses an Arduino MKR Zero microcontroller to collect, process, and store the data collected by the various sensors. Each sensor is strategically placed to insure that they don’t interfere with the bats natural behaviour. The device involves the use of a button, which allows for the transfer of data from the microcontroller to the user’s personal device, and the use of a switch, which can turn off the bluetooth functionality of the device to maintain battery life. The device runs on battery power with a life of at least one month, making it suitable for remote locations without constant supervision.

The microcontroller, battery, and all other necessary components are housed in a recyclable, water tight, and durable enclosure, which ensures protection in harsh outdoor conditions. The enclosure is designed to be easily opened, to allow access to the button, switch, and battery. The sensors that are responsible for tracking the activity of the bats are protected via a custom made enclosure. Additional wooden pieces are securely placed on the bat box, which can be used to hold the sensors in place, using either an interference fit or screws as needed.

**Figure 2.0**

*Block Diagram of how B.R.U.C.E. functions*

## 2.1 Conventions

**Measurement Units**: Centimeters (cm); inches (in); volts (V); ohms (Ω); milliamps (mA); milliamp hours (mAh)

**Data Formats**: All data is stored in a **.txt** file

**Colour Coding**: Red wires for power, black wires for ground

## 2.2 Cautions & Warnings

It is important to handle the battery with care to ensure safety and functionality of the device. The user must avoid exposing the battery to extreme temperatures, water, or direct sunlight, as it may cause damage or pose safety risks. Do not puncture, disassemble, or attempt to modify the battery. If there are any signs of damage on the battery, such as swelling, or leakage, please discontinue use and immediately dispose of it responsibly. Keep the battery out of reach of children.

The user is responsible to ensure that the bluetooth module is turned off when not in use, as it may quickly drain the capacity of the battery.

# Getting Started

B.R.U.C.E. is designed to track bat movements entering and exiting a bat box, while also monitoring the internal temperature and humidity of the bat box. The system involves the use of 3 IR sensors, a temperature and humidity sensor, a BT module and an Arduino MKR Zero microcontroller to collect and store data about bat activity, helping researchers understand bat behavior. B.R.U.C.E. is powered via two 3.7V 12000 mAh batteries. The microcontroller, BT module, batteries and all necessary components are stored inside a durable, and watertight enclosure.

## 3.1 Configuration Considerations

The system consists of three main components: the bat box, the monitoring system, and the user interface. Each part works together to track bat activity, monitor environmental conditions, and provide data access via Bluetooth.

1. **Monitoring System**
   * The Arduino MKR Zero is housed in a durable, waterproof ABS enclosure mounted on the side of the bat box.
   * Three infrared (IR) sensors are positioned inside and outside the box to detect bat entry and exit movements.
   * Sensors and the DHT11 temperature and humidity sensor are connected to the Arduino through organized cables routed into the enclosure.
   * The system is powered by a rechargeable lithium-ion battery pack located inside the enclosure.
2. **User Interface**
   * Data collected by the system is stored on a microSD card and can be transmitted wirelessly to a smartphone or computer via Bluetooth.
   * Users can view bat activity and environmental data on a mobile app or software compatible with the HC-05 Bluetooth module.

#### Tools Needed

* **Phillips Screwdriver**: For securing the bat box and the ABS enclosure.
* **Battery connector**: For powering the Arduino
* **Mobile Device or Laptop**: To receive data via Bluetooth.

#### 

#### Connections

* **IR Sensors**: Connected to the Arduino MKR Zero via cables routed through the PG 11 cable fittings.
* **DHT11 Sensor**: Placed inside the bat box and connected to the Arduino for environmental monitoring.
* **Power Supply**: Battery connected to the Arduino MKR Zero
* **Bluetooth Module**: Integrated with the Arduino to enable wireless communication with a user device.

## 3.2 User Access Considerations

**B.R.U.C.E** has been designed with multiple user groups in mind, each with distinct levels of access and responsibilities. Accessibility and usage restrictions have been implemented to ensure the system operates effectively while safeguarding the data and hardware.

1. **Researchers and Conservationists**

**Role**: Primary users responsible for monitoring bat activity, collecting data, and maintaining the system.

**Access**:

* Full access to the system, including the Bluetooth module, SD card, and maintenance tasks.
* Can retrieve and analyze data logs for research purposes.
* Responsible for calibrating the system and verifying sensor functionality during installation or maintenance.

**Restrictions**:

* Must ensure proper handling of components to avoid hardware damage.
* Data logs should not be shared publicly without proper authorization to protect sensitive ecological research.

1. **Maintenance Teams**

**Role:** Secondary users responsible for hardware upkeep, including battery replacement, cleaning, and sensor alignment.

**Access:**

* Limited access to the system, focusing on hardware maintenance and basic troubleshooting.
* Can verify that the system is operational but may not retrieve or interpret the data logs.

**Restrictions:**

* Should not tamper with system calibration settings or access detailed data logs unless authorized.
* Maintenance tasks must be performed following the guidelines outlined in the user manual to prevent misalignment or damage to the sensors.

1. **Clients or Sponsors**

**Role:** Stakeholders interested in system performance and high-level outcomes.

**Access:**

* Limited to reviewing summarized data or reports provided by researchers.
* No direct interaction with hardware or data collection processes.

**Restrictions:**

* Cannot access raw data logs or make changes to the system configuration.
* Restricted from performing maintenance tasks to ensure system integrity and avoid potential misoperation.

1. **General Public or Unauthorized Users**

**Role:** Individuals with no defined role in system operations, such as passersby or on-site visitors.

**Access:**

No access to the system’s components or data.

**Restrictions:**

* The enclosure is locked and securely mounted to prevent tampering or accidental damage.
* Unauthorized attempts to access the system may result in data corruption or operational failure.

## 3.3 Accessing/setting up the System

Step 1: Setting Up the System

1. **Mount the Enclosure**:
   * Mount the enclosure on either side of the bat box using using ½ inch #8 screws. The exact position of the enclosure is dependent on the bat box used.

**Figure 3**.**0**

*The enclosure mounted on the side of the bat box.*

Step 2: Installing the Sensors

1. **Position the Sensors**:

***It is important to note that the installation of the sensors is easier when the bat box is not fully assembled.***

* + Attach **Sensor 1** inside of the bat box at the bottom entrance & exit.
  + Attach **Sensor 2** outside of the bat box at the bottom entrance & exit.
  + Place **Sensor 3** at the top exit to track bats leaving.
    1. Sensor 1 must be held in place with the use of screws.
    2. Sensors 2 and 3 must be attached to the bat box with the use of wooden mounts. The wooden mounts must be attached onto the bat box with the use of wood glue, and those sensors can fit in the position shown through an interference fit.
  + Place the Temperature and Humidity sensor on the inside of the roof of the bat box.
    1. To hold the sensor in place, use a hot glue gun to glue the wires to the bat box.
  + Neatly route the wires from each sensor through the bat box to avoid causing interference with bat movement.

**Figure 4.0**

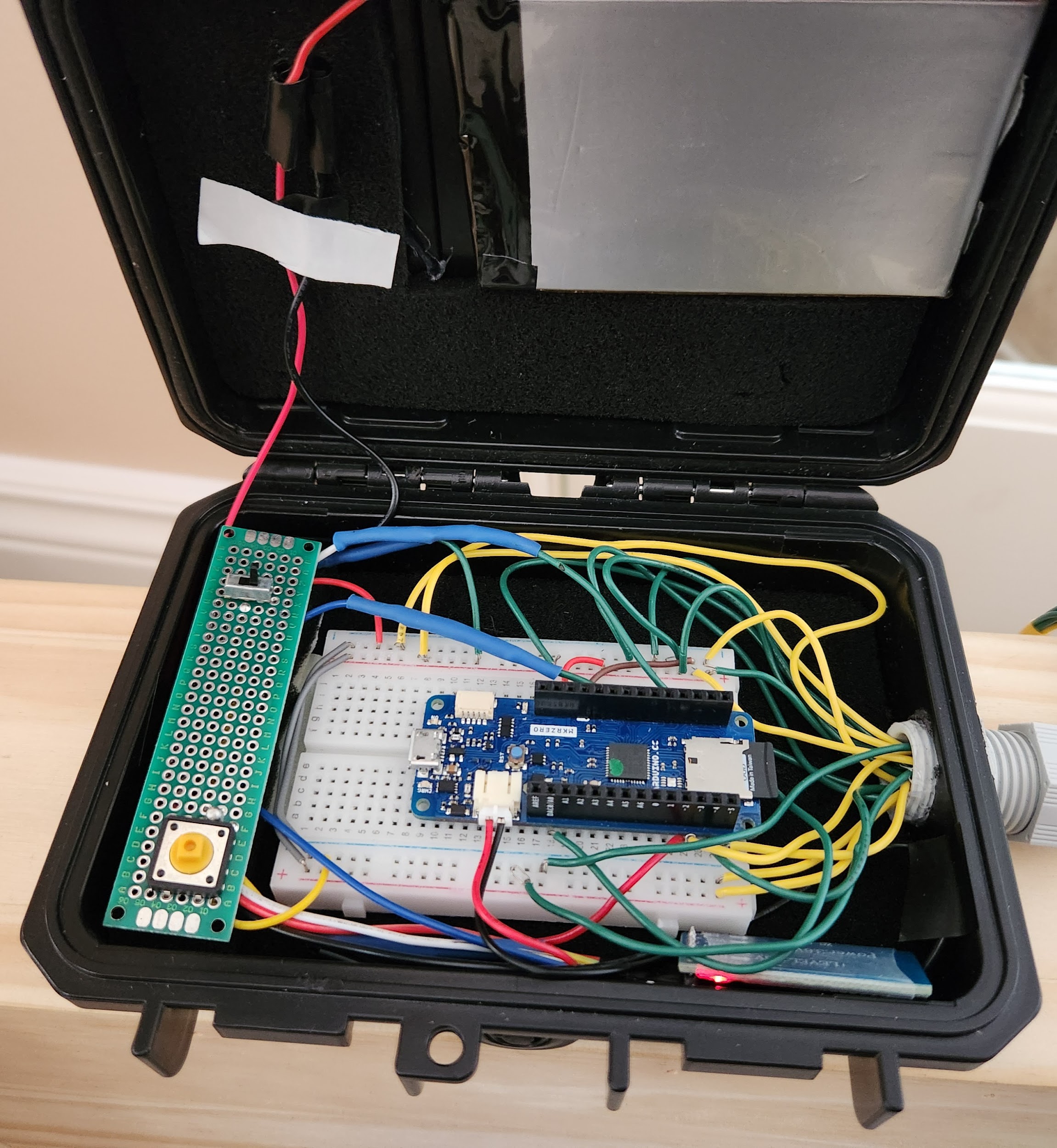
*IR sensor placements*

*Note.* Close-up of the bottom of the bat box

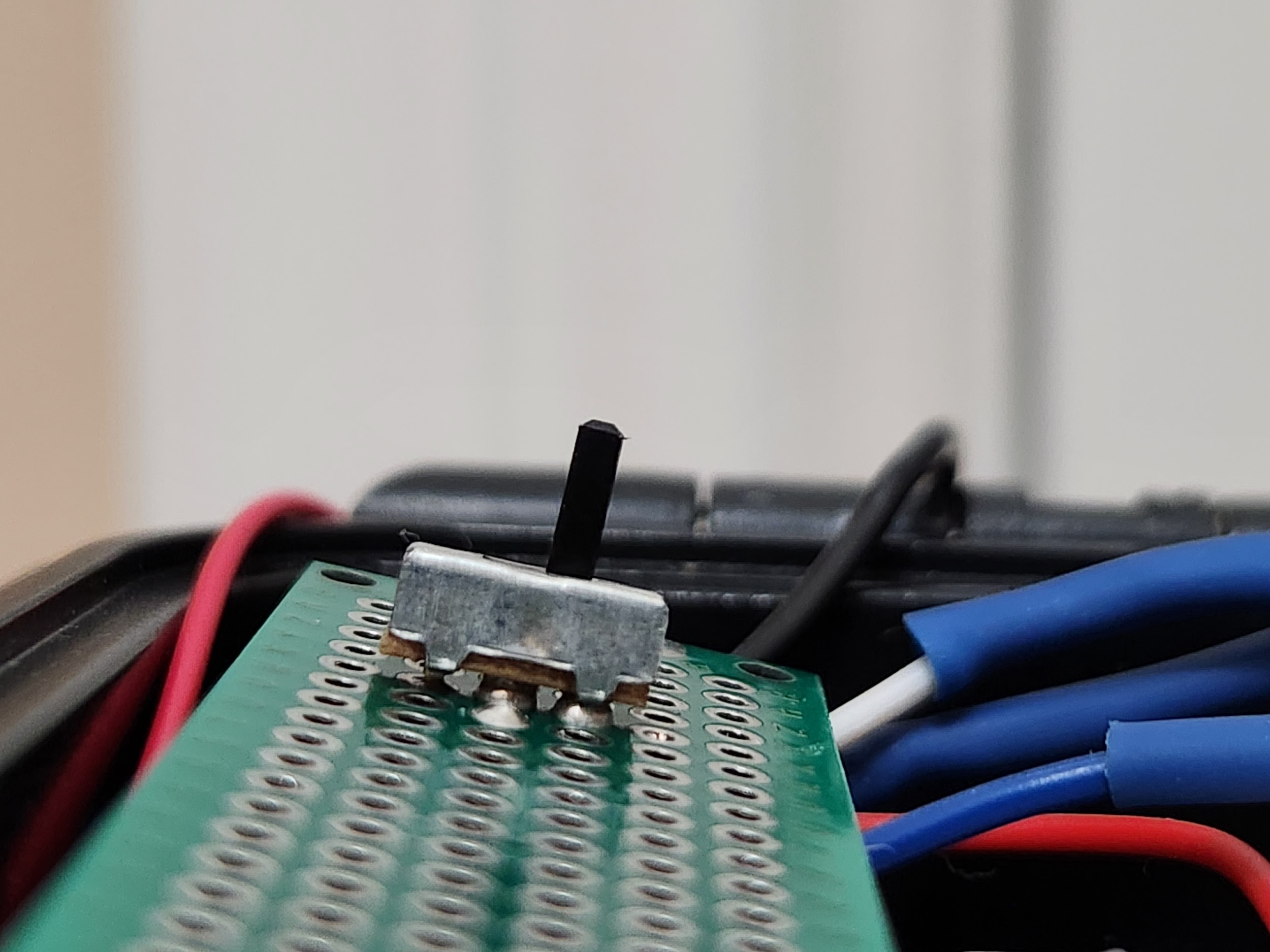
Step 3: Powering the System

1. **Attach Power Source**:
   * Open the enclosure and plug in the battery connector in the port shown in Figure 3.
   * Once the BT module LED is flashing, slide the switch to turn off the BT module to ensure the 30-day battery life (refer to Figure 4.0 and Figure 5.0).
   * Once the system is powered on and the bluetooth module is turned off, the system will begin to function without any further adjustments.

**Figure 5.0**

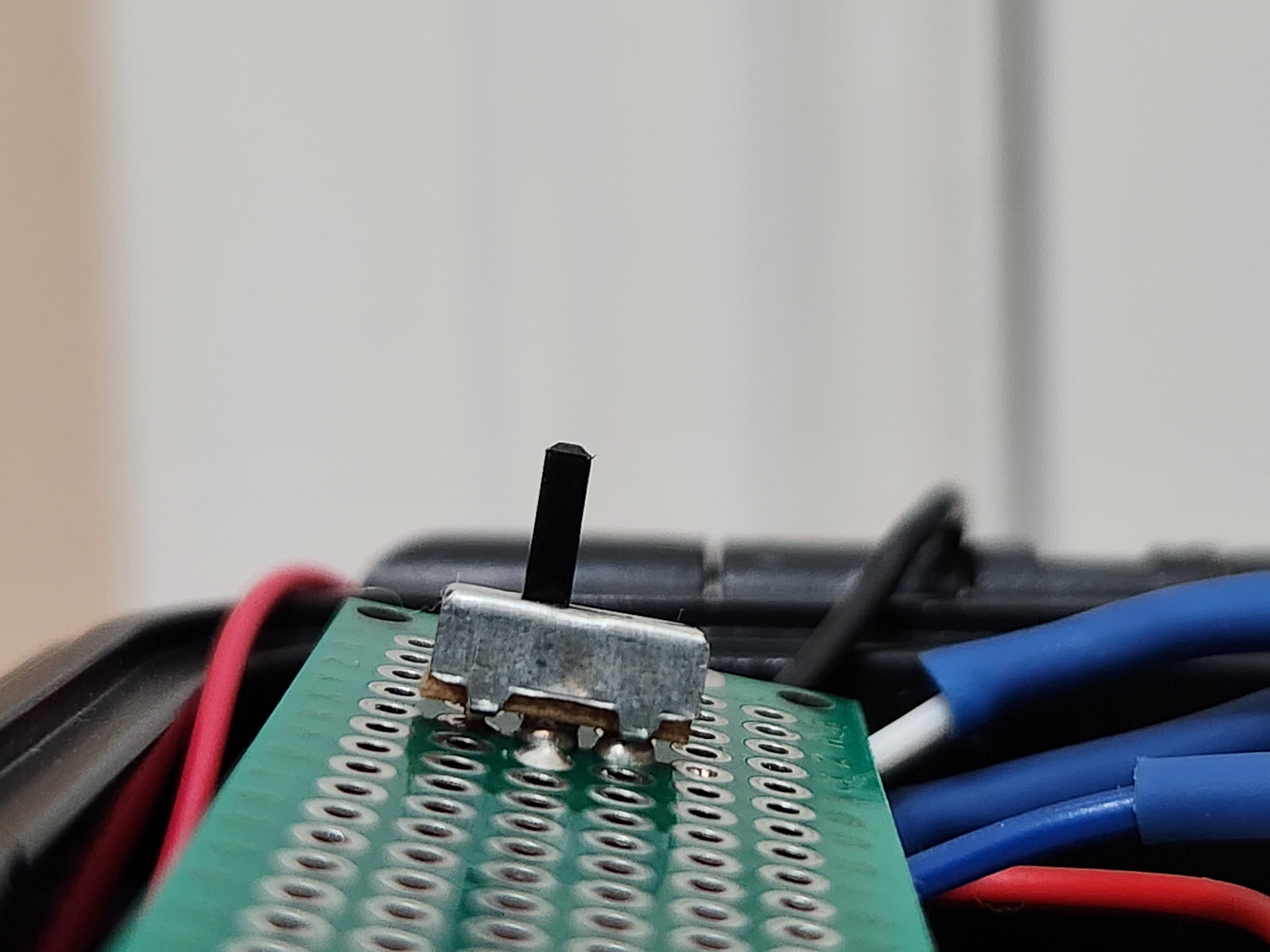
*The interior of the enclosure showing the Arduino MKR Zero with the battery connected*

**Figure 6.0**

*Showcasing the on state for the switch*

*Note.* The left side of the image is the top of the enclosure, while the right side in the bottom of the enclosure

**Figure 7.0**

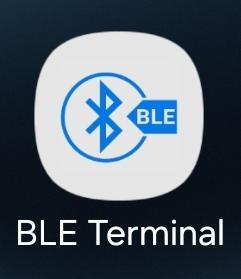
*Showcasing the off state for the switch*

*Note.* The left side of the image is the top of the enclosure, while the right side in the bottom of the enclosure

Step 6: Retrieving Data

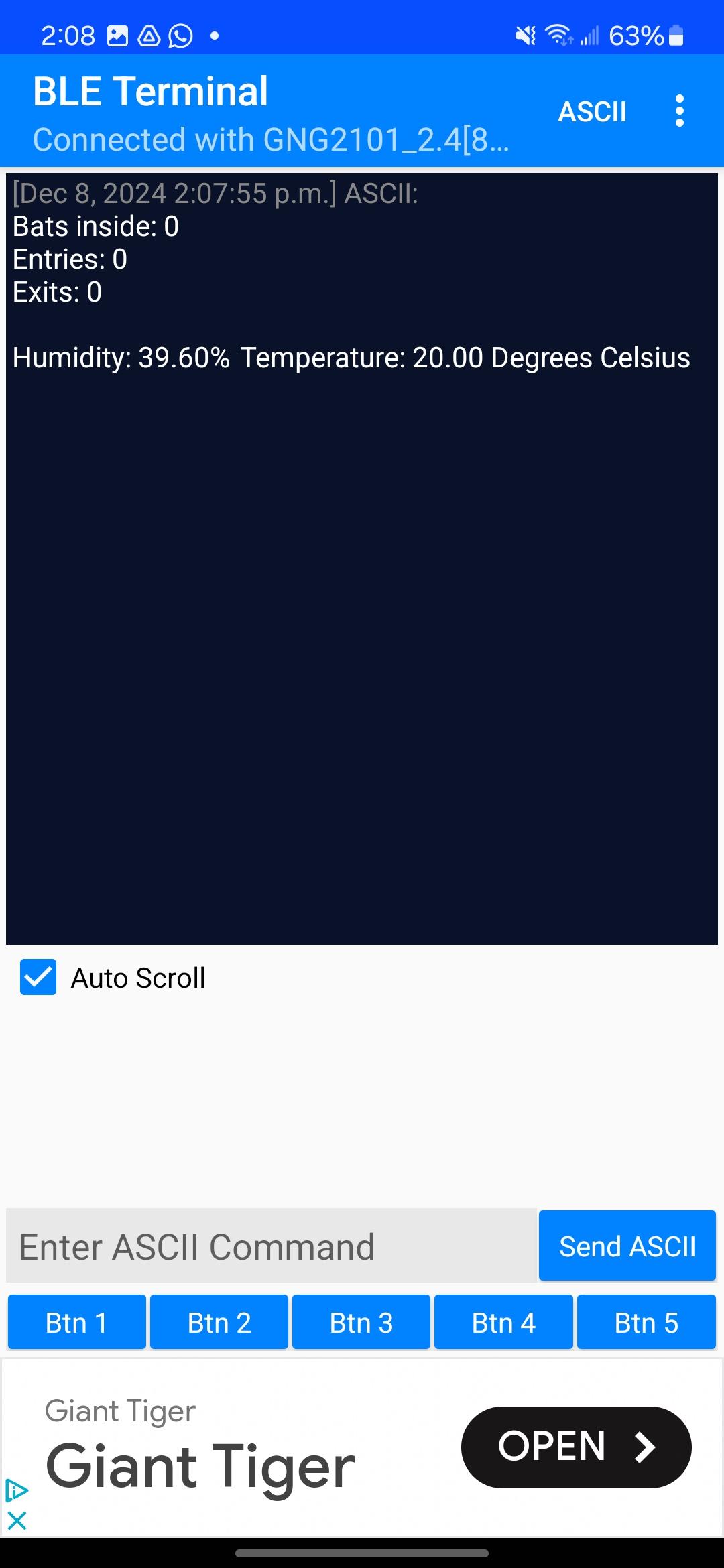
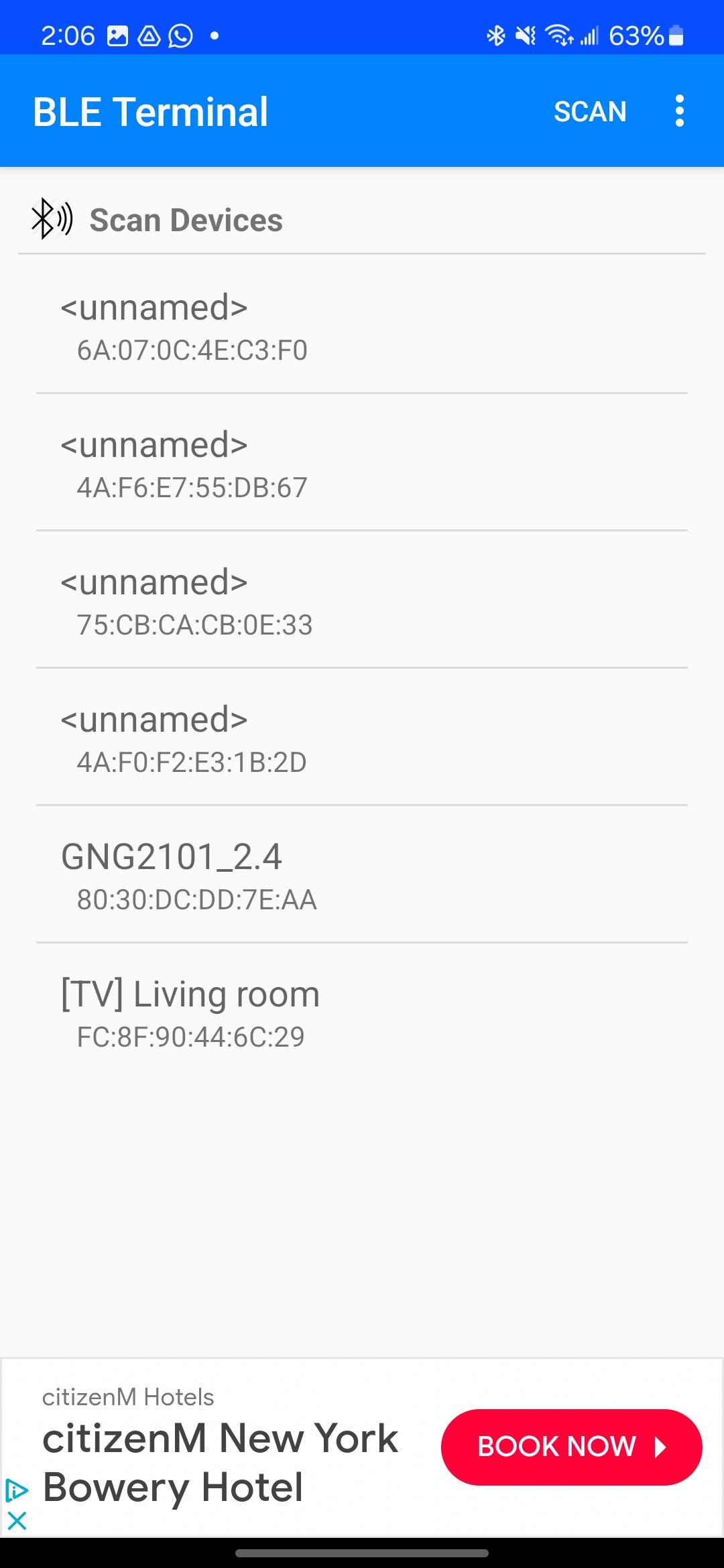
1. **Access the Stored Data**:
   * Flick the switch to the on state (refer to Figure 4.0). The LED on the BT module should be turning on and off.
   * Download the BLE terminal application on your mobile device (refer to Figure 6.0).
   * Open the application and search for a device called GNG2101\_2.4. (Refer to Figure 9)
   * Once the BT is connected, press the yellow button to send the data onto the BLE terminal.
   * If necessary, the microSD card can be removed to gain direct access to the data into a personal computer.

**Figure 8.0**:

*Screenshot of the application logo*.

## 

**Figure 9.0:**

*Inside the application*

## 3.4 System Organization & Navigation

#### 

#### 3.4.1 Main Component: Arduino MKR Zero

The Arduino MKR Zero serves as the central processing unit of the system, responsible for coordinating all sensor inputs, managing data storage, and enabling Bluetooth communication.

* **Connections**:
  + Receives inputs from the IR break beam sensors.
  + Collects temperature and humidity data from the DHT11 sensor.
  + Stores collected data on its onboard microSD card.
  + Sends data via Bluetooth to the user’s device using the HC-05 Bluetooth module.

#### 

#### 3.4.2 Infrared (IR) Break Beam Sensors

The system includes three IR break beam sensors strategically placed within the bat box to monitor bat activity:

1. **Sensor 1**: Positioned at the bottom entrance inside the box, detecting bats entering the enclosure.
2. **Sensor 2**: Positioned at the bottom entrance outside the box, working with Sensor 1 to determine entry/exit sequences.
3. **Sensor 3**: Positioned at the top of the box, solely monitoring bats exiting the box.

* **Connections**:
  + Each sensor is wired to the Arduino MKR Zero via designated digital input pins.
  + The sensors are powered through a shared 3.7V supply to ensure consistent operation.

#### 3.4.3 Temperature and Humidity Sensor (DHT11)

The DHT11 sensor monitors environmental conditions within the bat box, such as temperature and humidity, providing critical data for studying bat behavior relative to climatic factors.

* **Connections**:
  + Wired directly to the Arduino MKR Zero via an analog input pin.
  + Shares the 3.7V power supply with other components.

#### 

#### 3.4.4 Data Storage: MicroSD Card

The system utilizes the Arduino MKR Zero’s onboard microSD card slot for data logging. This feature ensures that all bat activity and environmental data are recorded for future analysis.

* **Connections**:
  + Integrated into the Arduino MKR Zero, eliminating the need for external breakout boards.
  + Data is logged in a structured format to ensure easy retrieval and analysis.

#### 3.4.5 Wireless Communication: Bluetooth Module (HC-05)

The HC-05 Bluetooth module enables wireless transmission of recorded data from the system to the user’s smartphone or other Bluetooth-compatible devices. This feature enhances user accessibility and data analysis convenience.

* **Connections**:
  + Connected to the Arduino MKR Zero.
  + Operates on the shared 3.7V power supply.

#### 3.4.6 Enclosure

The enclosure, made of laminated whitewood, houses the Arduino MKR Zero and all attached components, protecting them from environmental factors such as moisture, dust, and physical impact. PG 11 cable fittings are used to route cables securely, ensuring durability and ease of maintenance.

* **Design Features**:
  + **Waterproofing**: Tested for water resistance, ensuring no internal damage during outdoor use.
  + **Cable Management**: Organized routing to prevent tangling and simplify maintenance.
  + **Access Ports**: Accessible openings for battery replacement and maintenance tasks.

#### 3.4.7 Power Supply

The system is powered by two 3.7 V 12000mAh lithium-ion batteries, wired in parallel, providing reliable power for extended operation.

## 3.5 Exiting the System

To turn off the device, all they must do is unplug the battery that is connected to the Arduino MKRZero. To disassemble the system, carefully follow the previously outlined setup instructions in reverse order.

# 4. Using the System

The following subsections provide step-by-step instructions on how to use the features of **B.R.U.C.E**. Each section explains the user interaction, system responses, and any necessary troubleshooting steps.

## 4.1 Setting Up The System

Before operation, **B.R.U.C.E** must be installed and calibrated properly.

* **Input**: Secure the system enclosure to the bat box using ½ inch #8 screws. Ensure the IR break beam sensors are correctly aligned at the entrance of the bat box for accurate detection.
* **Output**: When powered, the system will emit a brief LED flash to indicate that the sensors are operational and Bluetooth communication is active.

**Steps**:

1. Mount the enclosure to the side of the bat box using #8 x 1/2-in Flat Head Square Drive Steel Wood Screws and a screwdriver.
2. Plug in the provided rechargeable battery.
3. Connect to the system’s Bluetooth module via a smartphone or laptop, then press the yellow button inside the enclosure to verify that the system is operational (Figure 9.0).
4. After setup, turn the Bluetooth module "OFF" by using the toggle switch. This will be indicated by the LED on the Bluetooth module turning off **(**Figure 11.0).

### 

## 4.2 Tracking Bat Activity

The IR break beam sensors track bats entering and exiting the bat box.

* **Input**: Bats passing through the break beam disrupt the IR sensor field. No manual input is required.
* **Output**: The system counts each disruption and updates the tally, storing the data on the internal SD card.

**Steps:**

1. Ensure the sensors are unobstructed and correctly aligned.
2. Bats will automatically be counted as they cross the sensor field.

**Note**: If using a smartphone, it must be an Android device, as the required app is not available on the iOS App Store (see Figure 8.0).

## 4.3 Accessing Stored Data

The system logs data on the SD card, which can be accessed and downloaded for analysis. Additionally, the data can be retrieved wirelessly via the Bluetooth terminal.

* **Input**: Open the enclosure to remove the SD card or connect via Bluetooth to download data.
* **Output**: Data is saved as a **.txt** file titled “**bat\_activity\_log.txt.”**

**Steps**:

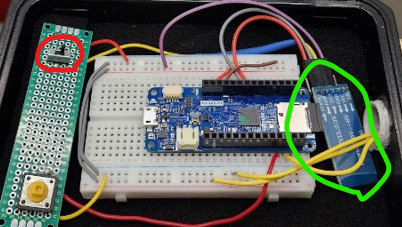
1. **Access via SD Card**:
   1. Open the enclosure and power off the system by unplugging the battery.
   2. Eject the SD card.
   3. Insert the SD card into a computer to retrieve the data file.

**Figure 10.0**: *Inside the enclosure*

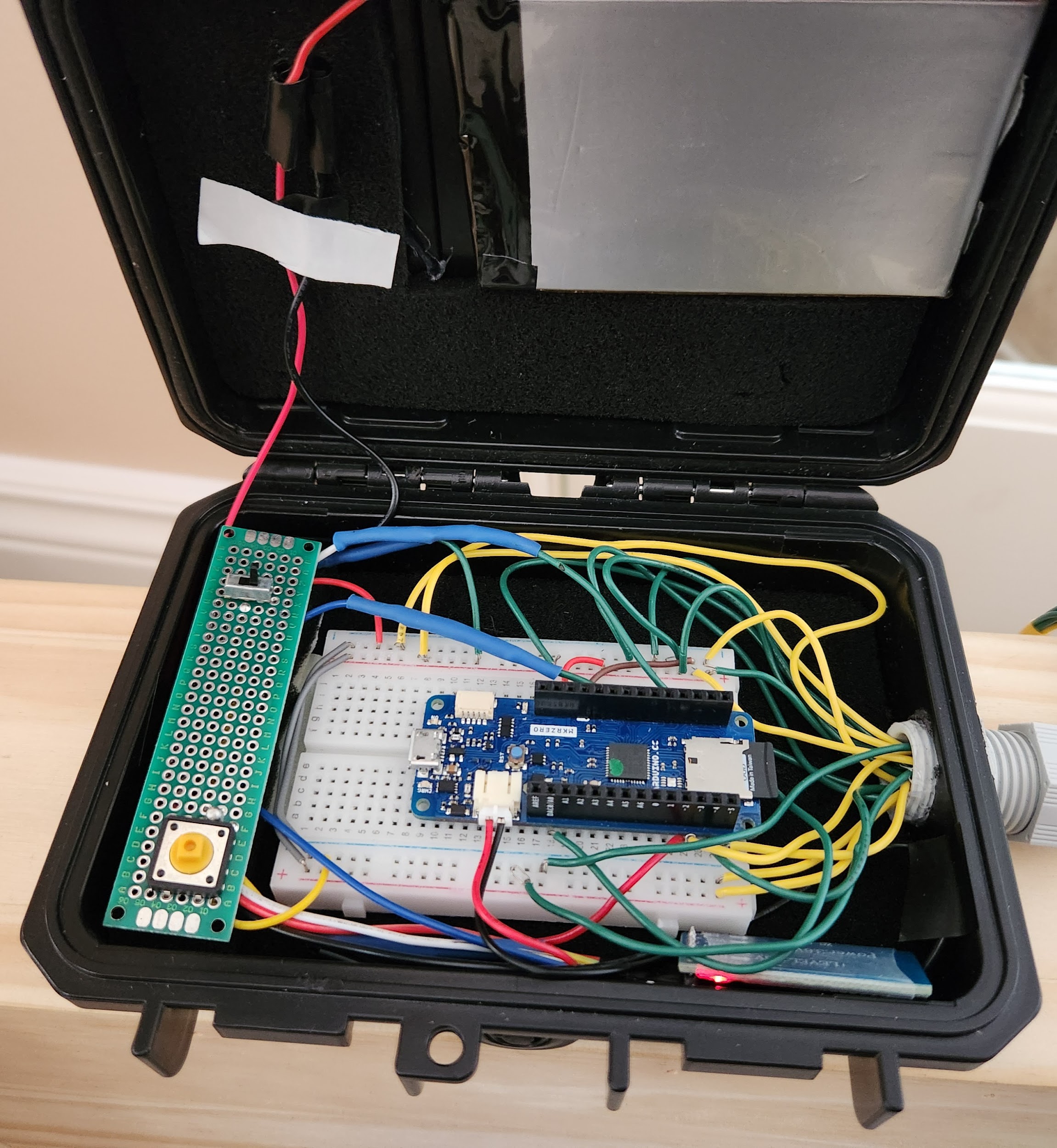


1. **Access via Bluetooth Terminal**:
   1. Ensure the system is powered on and turn on the Bluetooth module by sliding the switch (refer to Figure 10.0 with the red circle showcasing the switch) to either one of the sides; the module is active if a red LED is turned on (see figure 11)
   2. Connect to the HC05 Bluetooth module using a Bluetooth terminal app on your smartphone or computer.
   3. Once connected, press the yellow button (circled below in thick orange) within the enclosure to send data to the terminal.
   4. The system will transmit the current tally and all recorded readings at that moment.
   5. Save the transmitted log as a **.txt** file using the app's save function.
   6. Turn off the bluetooth module by sliding the switch in Figures 4 and 5, and such that the module is not lit anymore.

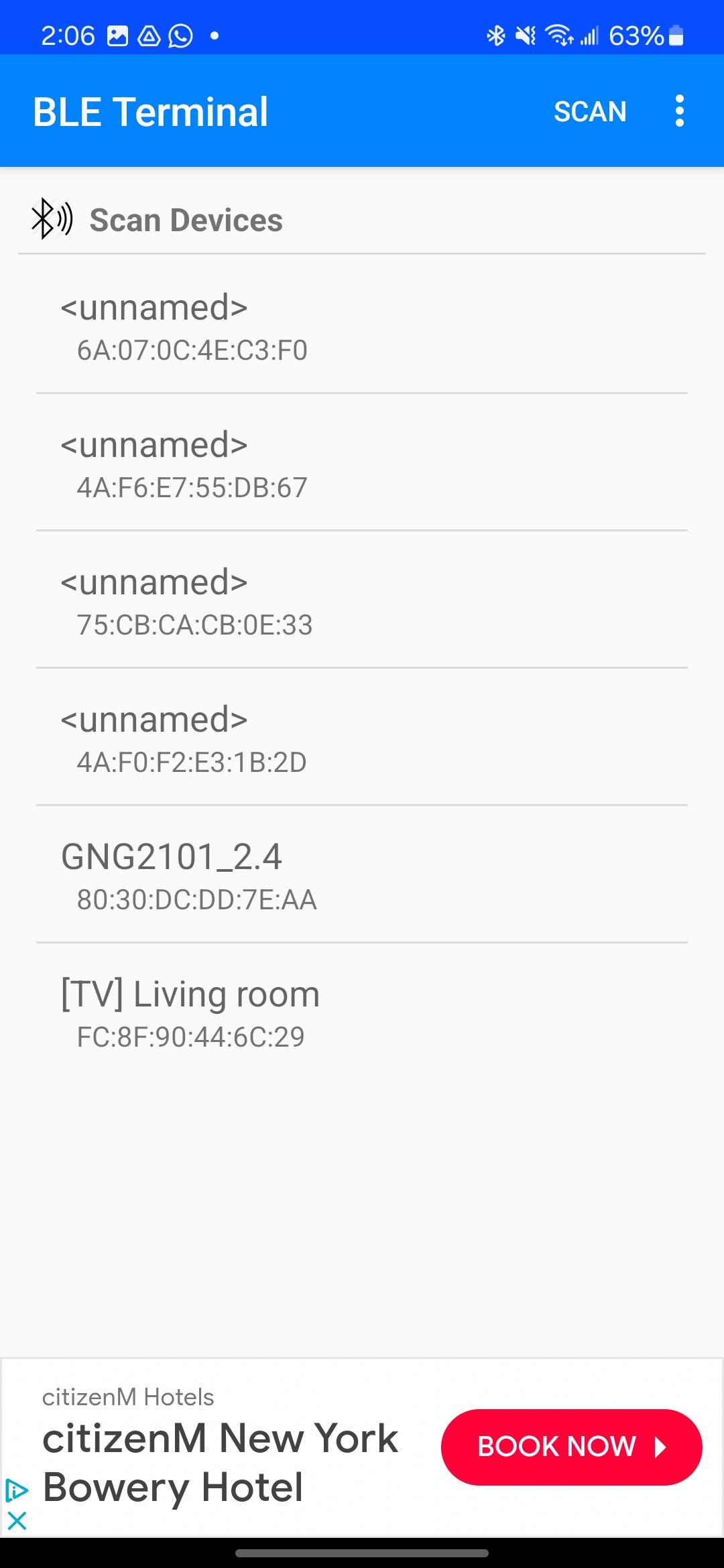
**Figure 11.0**: *Highlighting the Switch and BT module*



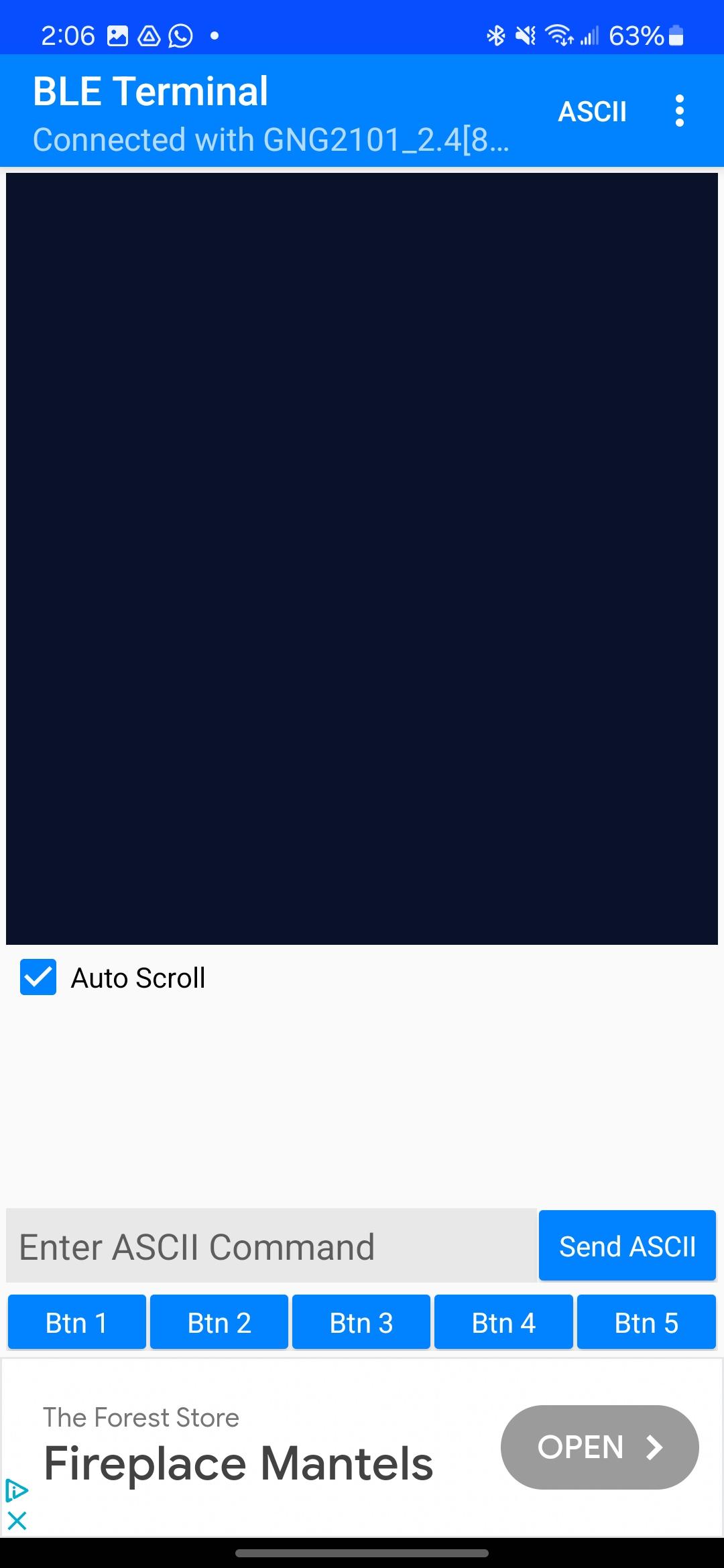
**Figure 12.0**: BT module LED



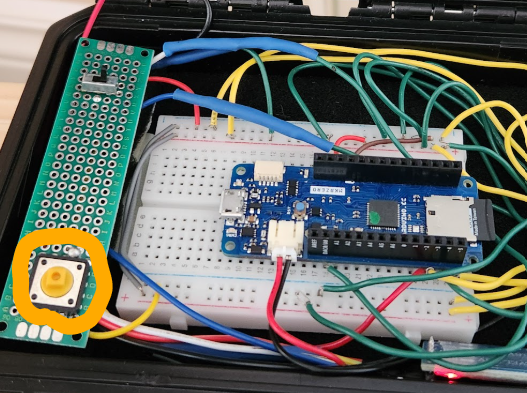
**Figure 13.0**: *BLE Terminal Scanning for Devices*



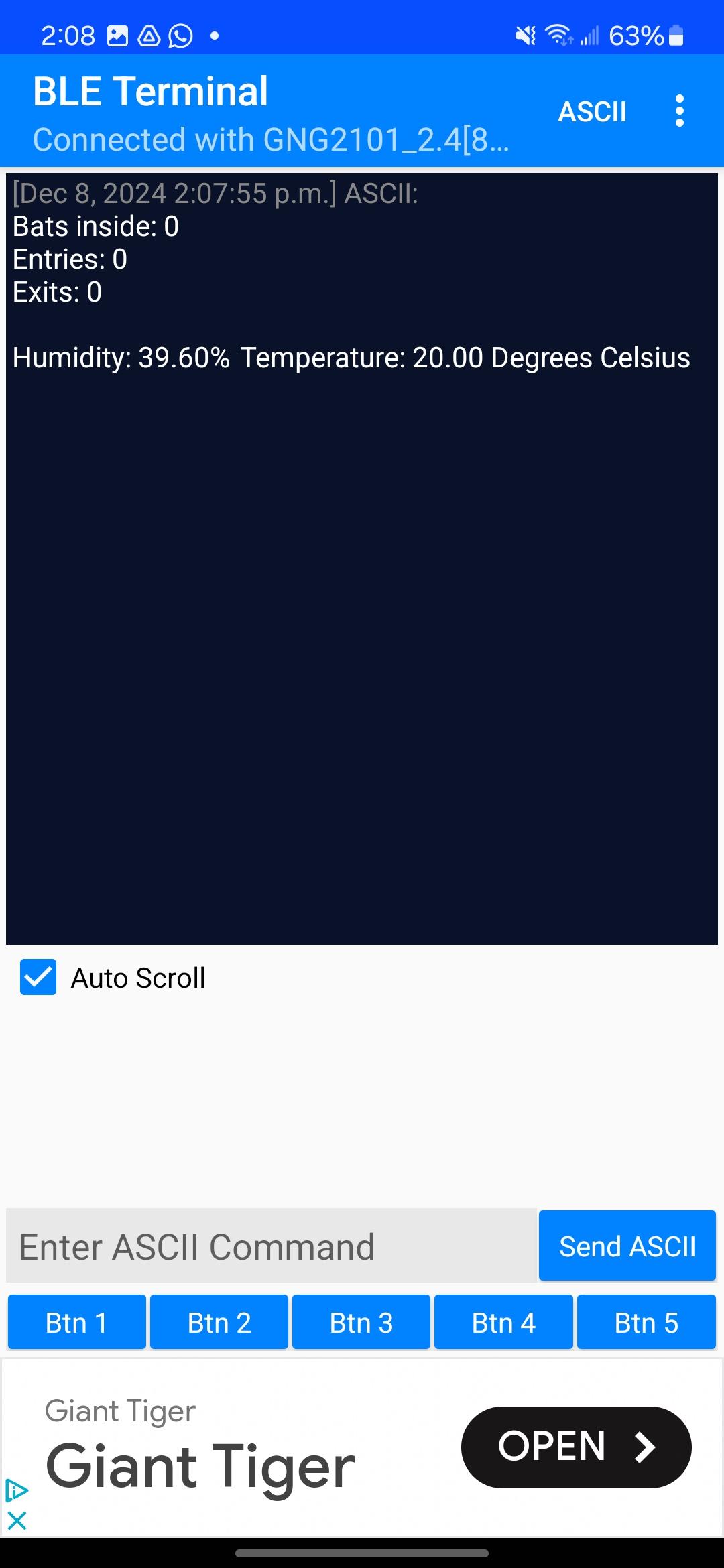
**Figure 14.0**: *Empty BLE Terminal*



**Figure 15.0**: *Highlighting the Yellow Button*



**Figure 16.0**: *Active BLE Terminal Running Code*



**Caveats**:

* Ensure the Bluetooth terminal connection is stable before requesting the data log.
* Power off the system before removing the SD card to prevent data corruption

# 5. Troubleshooting & Support

## 5.1 Error Messages or Behaviors

1. **IR Sensors Not Detecting Bats**

* **Likely Cause**:
  + Sensors are misaligned or blocked by dirt/debris.
* **Corrective Action**:
  + Visually inspect the IR sensors to ensure they are clean and free from obstructions.
  + Check that the sensors are securely mounted and properly aligned.

1. **Inaccurate Bat Counting**

* **Likely Cause**:
  + Sensors are detecting overlapping movement or are obstructed.
* **Corrective Action**:
  + Ensure that nothing blocks the line of sight for the sensors.
  + Confirm that the sensors at the bottom are positioned to detect bats entering and exiting, and the top sensor is correctly mounted to track exits only.

1. **Temperature and Humidity Readings Seem Wrong**

* **Likely Cause**:
  + The sensor is improperly placed or affected by its surroundings.
* **Corrective Action**:
  + Check that the temperature and humidity sensor is not near heat sources or airflow disturbances.

1. **System Not Powering On**

* **Likely Cause**:
  + Loose power connections or insufficient power supply.
* **Corrective Action**:
  + Ensure the power cable is securely connected.
  + Verify that the power source (e.g., battery or external adapter) is functional.

## 5.2 Special Considerations

**Android-only devices**

* Bluetooth terminal apps (BLE) have only been tested on Android devices due to the lack of such apps in the Apple store. However, for Apple users, there exists web-based BLE applications that could be used instead, with the trade off of having to connect to the Internet when using the app.

**Turn Off Bluetooth Module**

* When not in use (i.e. to send data to a nearby device), turn off the bluetooth module (see [4.3](#_heading=h.bg0c2iubb0eg)). The module consumes too much power relative to the rest of the system components, and having it running in the background for more than the maintenance period can drain the battery over a short period of time.

**Ensure Proper Wiring**

* Upon building the device, double-check all sensor and module connections to the Arduino MKR Zero. Loose or incorrect connections are common sources of issues.

## 

## 5.3 Maintenance

The device must be maintained at least once a month. In particular, the two 3.7V 12000mAh Lithium ion battery cells have to be recharged to ensure abundant power supply for the device over the course of another month. The process boils down to unplugging the batteries from the Arduino board, recharging the batteries using proper techniques, and plugging them back into the Arduino. Besides, data collection (refer to [4.3](#_heading=h.bg0c2iubb0eg)) can be done during the monthly maintenance visit.

## 

## 5.4 Support

For emergency assistance and system support, contact the contributors to the project:

* Amin Khaled - [akhal134@uottawa.ca](mailto:akhal134@uottawa.ca)
* Mohammad Salem - [msale126@uottawa.ca](mailto:msale126@uottawa.ca)
* Hossam Alshaiba - [halsh081@uottawa.ca](mailto:halsh081@uottawa.ca)
* Waleed Sakalla - [wsaka038@uottawa.ca](mailto:wsaka038@uottawa.ca)
* J’afar Assaf - [jassa077@uottawa.ca](mailto:jassa077@uottawa.ca)

Allow 3-7 business days at most before hearing back from the contributors.

# 6. Product Documentation

## 6.1 Mechanical Design

### 

### 6.1.1 BOM (Bill of Materials)

**Table 2.0**

*Mechanical Part*

| Parts and Components | Unit Cost | Quantity | Total Cost |
| --- | --- | --- | --- |
| [Enclosure](https://www.aliexpress.com/item/1005006891497903.html?src=google&gQT=1) | $ 6.16 | 1 | $ 6.16 |
| [1/4" PG11 Cable Fitting](https://www.accessotronik.com/) | $ 5.20 | 1 | $ 5.20 |
|  | Total | 2 | $ 11.36 |

### 6.1.2 Equipment list

**Assembly Tools:**

* + Power drill
  + Screwdriver

## 

## 6.2 Electronics and Power

### 

### 6.2.1 BOM (Bill of Materials)

**Table 3.0**

*Electronic components*

| Parts and Components | Unit Cost | Quantity | Total Cost |
| --- | --- | --- | --- |
| [3.7V 12000mAh Lithium Ion Battery](https://www.aliexpress.com/item/1005007348720830.html?spm=a2g0o.productlist.main.45.1f904NSB4NSBBz&algo_pvid=7e193194-c09a-4757-b303-b0c1624b3504&algo_exp_id=7e193194-c09a-4757-b303-b0c1624b3504-22&pdp_npi=4%40dis%21CAD%2110.24%218.19%21%21%2151.53%2141.22%21%4021030ea417298962135003816e1950%2112000040366349780%21sea%21CA%210%21ABX&curPageLogUid=xK1vLOKedbai&utparam-url=scene%3Asearch%7Cquery_from%3A) | $ 15.98 | 2 | $ 31.96 |
| [Arduino MKR Zero](https://www.pishop.ca/product/arduino-mkr-zero-i2s-bus-sd/) | $ 36.95 | 1 | $ 36.95 |
| [DHT11 Temperature and Humidity Sensor](https://www.pishop.ca/product/arduino-compatible-digital-temperature-humidity-sensor-module-dht11/) | $ 2.95 | 1 | $ 2.95 |
| [Wire Spool](https://www.pishop.ca/product/solid-core-wire-spool-25ft-22awg-yellow/) | $ 3.95 | 1 | $ 3.95 |
| [Battery Connector](https://www.pishop.ca/product/jst-to-breadboard-jumper-3-pin/) | $ 2.77 | 1 | $ 2.77 |
|  | Total | 6 | $ 78.58 |

### 

### 6.2.2 Equipment list

The following equipment was used for assembling, prototyping, testing, and installing B.R.U.C.E. This list includes tools required for circuit assembly, sensor testing, enclosure preparation, and system debugging.

#### 

#### Assembly and Prototyping Equipment

**Soldering Tools**:

* + Soldering iron
  + Solder
  + Flux (optional, for stronger joints)
  + Desoldering pump or wick (for corrections)
  + Soldering stand or helping hands

**Wire Preparation Tools**:

* + Wire strippers and cutters
  + Heat shrink tubing and heat gun (optional, for insulation)
* **Basic Tools**:
  + Screwdrivers (for assembling the housing and securing components)
  + Needle-nose pliers (for wire adjustments)

#### Prototyping Tools

* Breadboard (for circuit validation)
* Jumper wires (for temporary connections during testing)

#### Testing and Measurement Equipment

* Multimeter (to test electrical connections, voltage, and continuity)
* Reflective surface or material (to verify IR sensor sensitivity and range)

#### Programming and Debugging Tools

* Computer or Laptop (for coding and uploading sketches to the Arduino MKR Zero)
* Micro USB Cable (to connect the Arduino for programming and power)
* Arduino IDE Software (installed on the computer for programming the Arduino)

#### Power Supply and Debugging

* Battery pack or power supply
* Li-ion battery (for long-term system operation)
* Bench power supply (optional, for controlled testing)

#### 

#### 

#### 

#### 

#### Enclosure and Mounting Tools

**ABS Plastic Cutting Tools**:

Rotary tool or drill (to create openings for wires and ventilation)

Sandpaper or file (to smooth edges after cutting)

**Fasteners and Adhesives**:

Screws, nuts, and bolts (to secure components inside the enclosure)

Zip ties or cable clips (for organizing wires)

**Mounting Tools**:

Drill and bits (to secure the container to the bat box)

Adhesive or sealant (to weatherproof wire entry points)

## 

## 6.3 Software

### 6.3.1 BOM (Bill of Materials)

**Table 4.0**

*Software components*

| Platforms and Libraries | Unit Cost | Quantity | Total Cost |
| --- | --- | --- | --- |
| [Arduino IDE](https://www.arduino.cc/en/software) | $0 | 1 | $0 |
| DHT [[1]](#footnote-0) Library | $0 | 1 | $0 |
| RTCZero Library | $0 | 1 | $0 |
| SD Library | $0 | 1 | $0 |
| SPI Library | $0 | 1 | $0 |
|  | Total | 5 | $0 |

### 6.3.2 Equipment list

To accomplish this section, the user needs a laptop or a PC with a USB port. Note that validating the code can be done independent of other subsystems, by just compiling and uploading the code (see next section) to the Arduino MKRZero board; however, the user would not be able to verify the expected behaviour of the IR sensors and the temperature/humidity sensor unless a proper circuit is connected to the board.

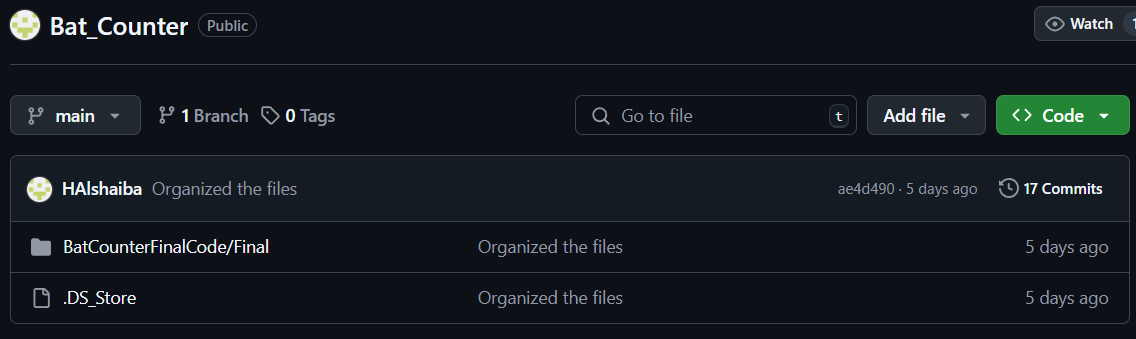
### 6.3.3 Instructions

If the Arduino IDE is installed, proceed with the next steps. Otherwise, visit the following link for installation instructions before continuing: <https://www.arduino.cc/en/software>.

The code that would be uploaded to the Arduino MKRZero board can be found as an open-source on the following GitHub repository: [HAlshaiba/Bat\_Counter](https://github.com/HAlshaiba/Bat_Counter/tree/main).

Clicking on the link redirects to a page that would look as follows:

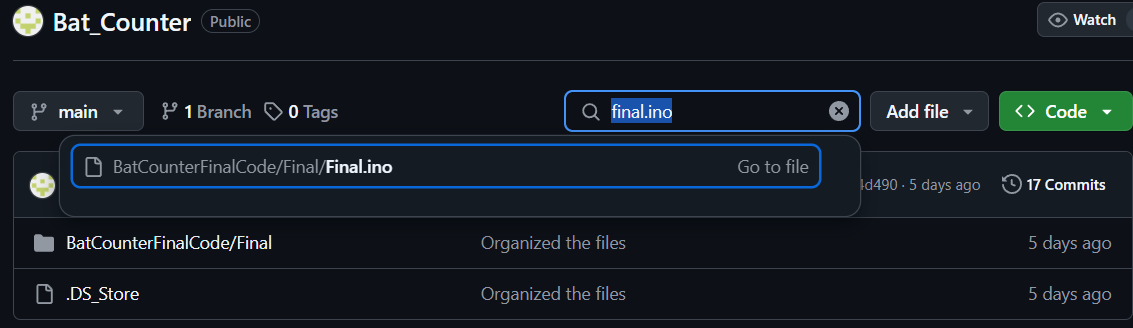
**Figure 17.0**: *Github Repository*



Click in the “Go to file” bar, and type in Final.ino.

**Figure 18.0**:

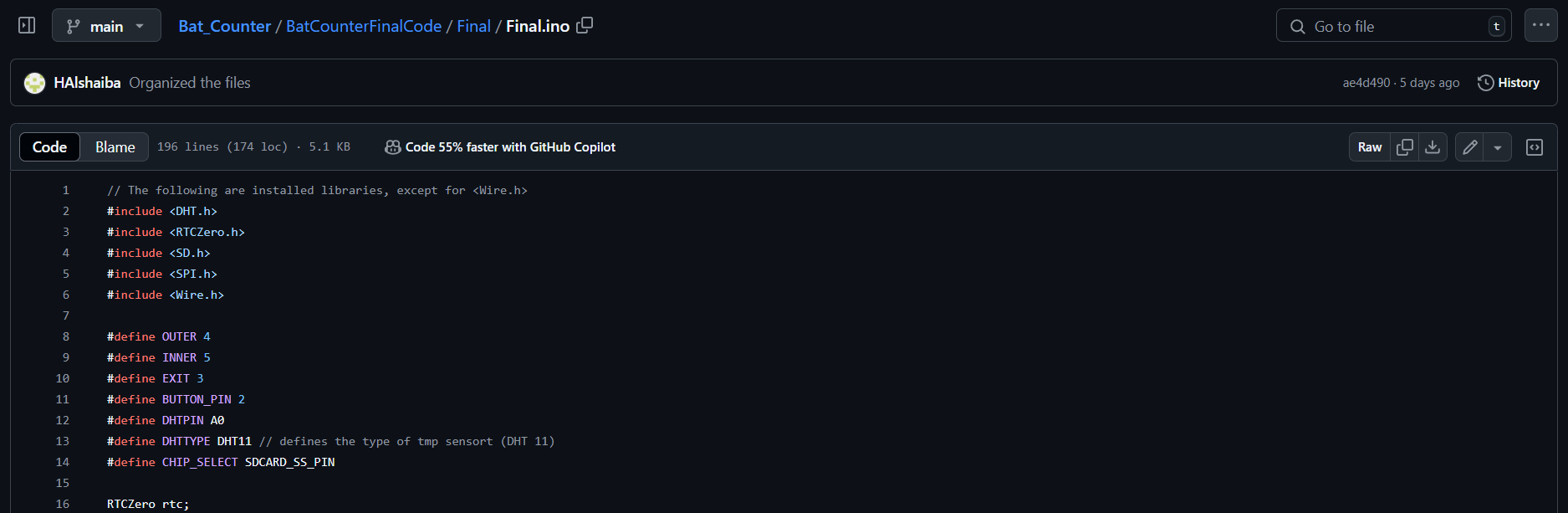
*Navigation to Final.ino File*



Clicking on “Go to file” redirects to the source code page:

**Figure 19.0**:

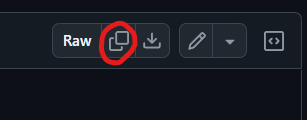
*Viewing the Source Code*



To copy the code, navigate to the top right section of the page and click on the copy symbol, as can be seen:

**Figure 20.0**:

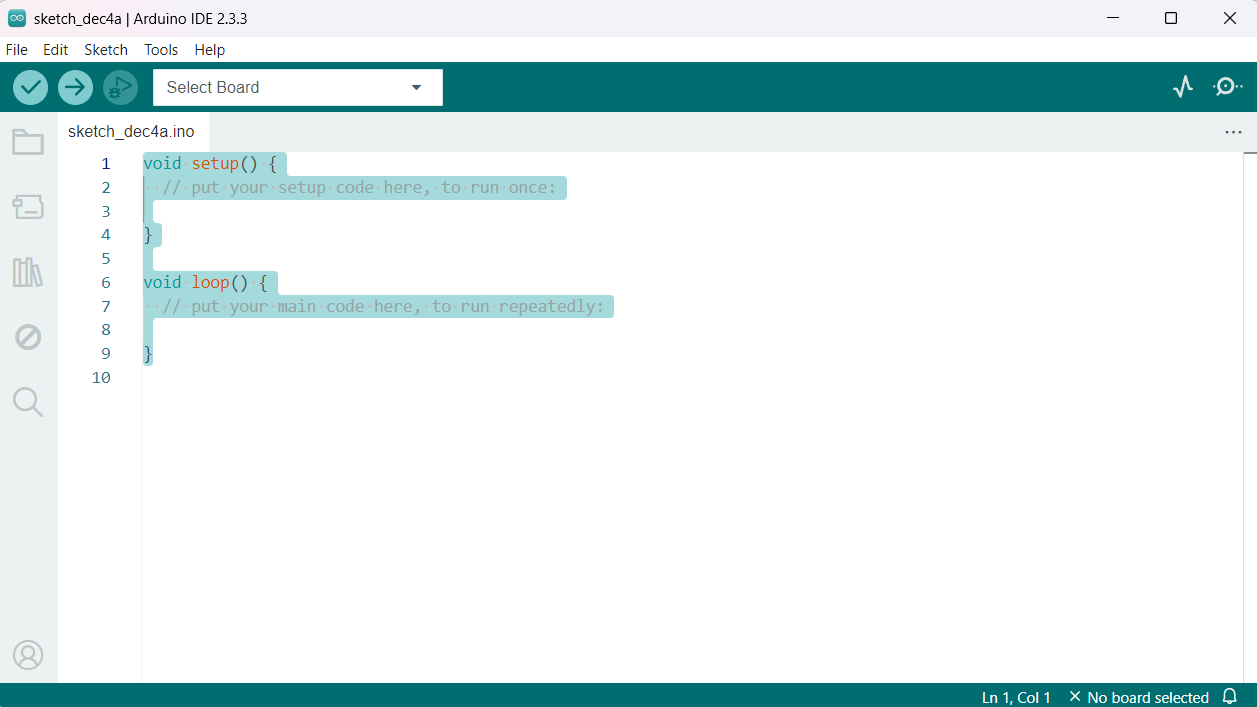
*Copy Button in GitHub*



Open Arduino IDE and create a new sketch:

**Figure 21.0**:

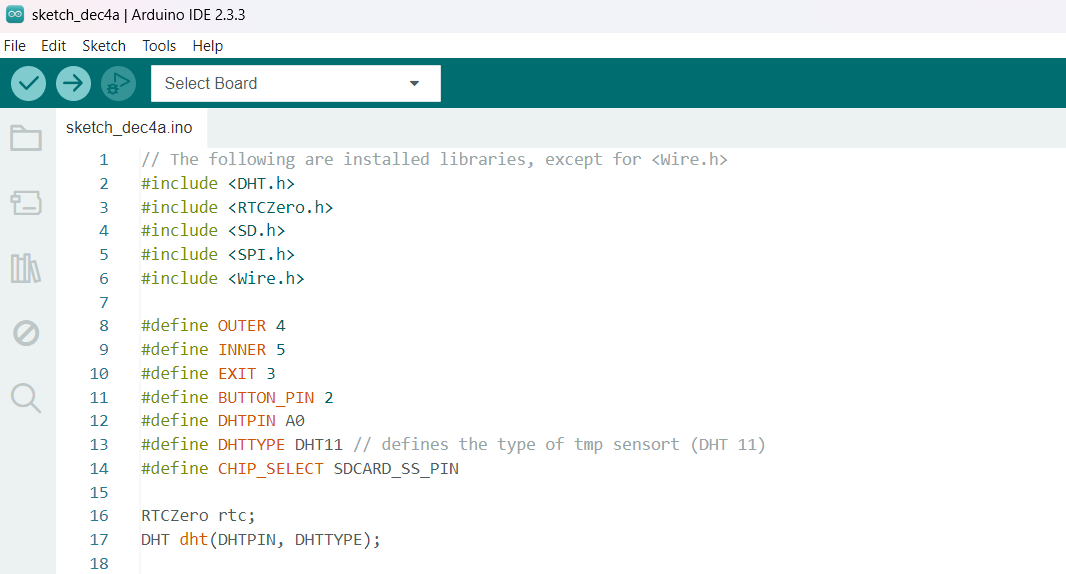
*Creating a New Sketch in Arduino IDE*



Override the template sketch by the code copied from the GitHub repository:

**Figure 22.0**:

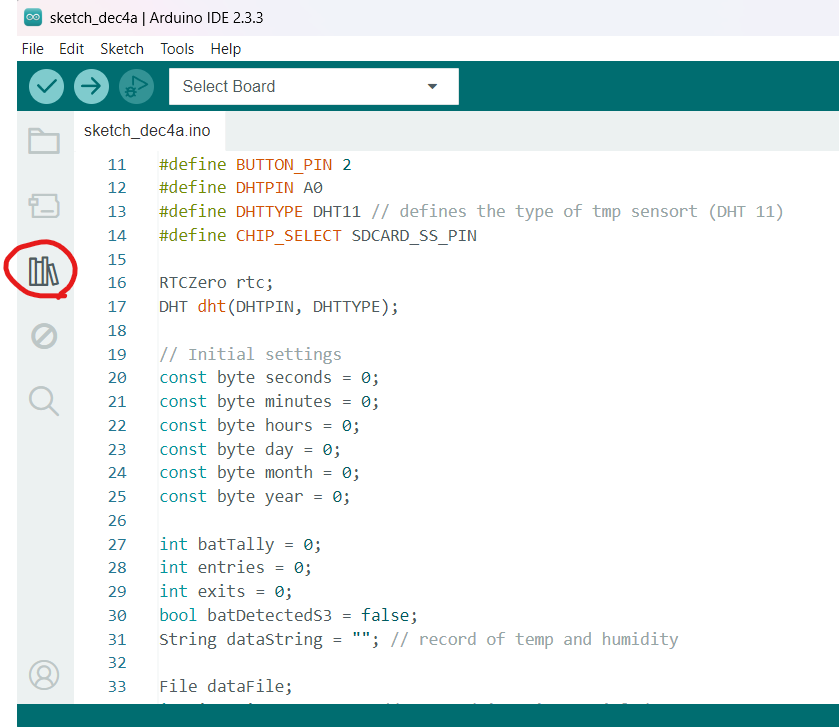
*Overwriting Template Code*



Click on Library Manager on the left:

**Figure 23.0**:

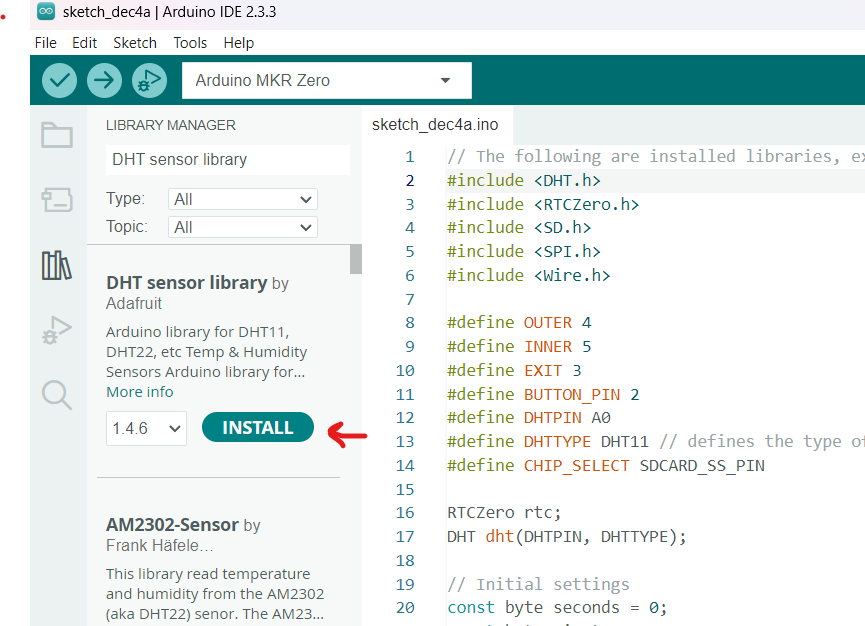
*Library Manager in Arduino IDE*



Type into the search bar “DHT sensor library”. Once found click on the install button:

**Figure 24.0**:

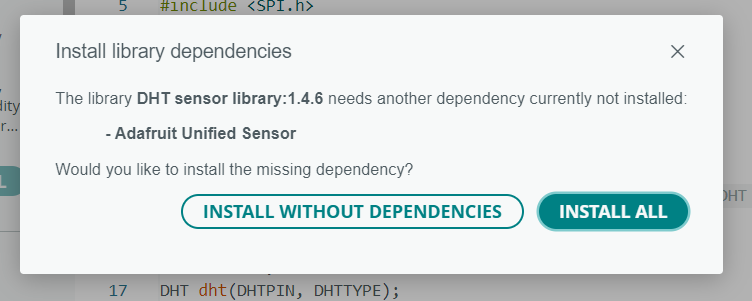
*Installing the DHT Sensor Library*



Click on “INSTALL ALL”,

**Figure 25.0**:

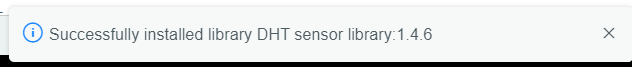
*Install All Confirmation in Arduino IDE*



and wait for the following message at the bottom right of the IDE window:

**Figure 26.0**:

*Installation Success for DHT Library*

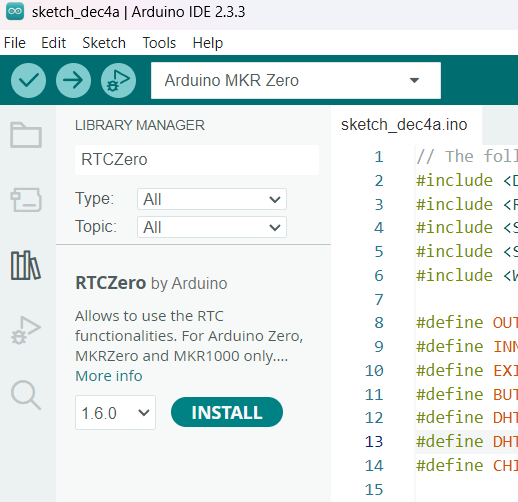


Go back to the Library Manager and type “RTCZero” into the search bar.

Click on install for the library “RTCZero by Arduino”:

**Figure 27.0**:

*Installing RTCZero Library*

~~~~

Wait for the following message at the bottom right of the IDE:

**Figure 28.0**:

*Installation Success for RTCZero Library*

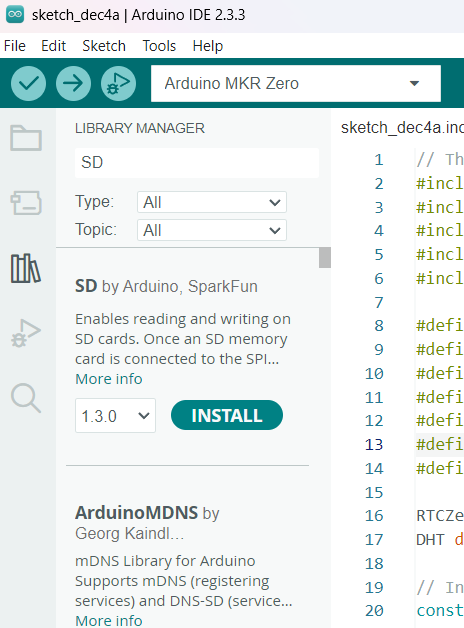


Go back to the Library Manager and type “SD” into the search bar.

Click on install for the library “SD by Arduino, SparkFun”:

**Figure 29.0**:

*Installing SD Library*



Wait for the following message at the bottom right of the IDE:

**Figure 30.0**:

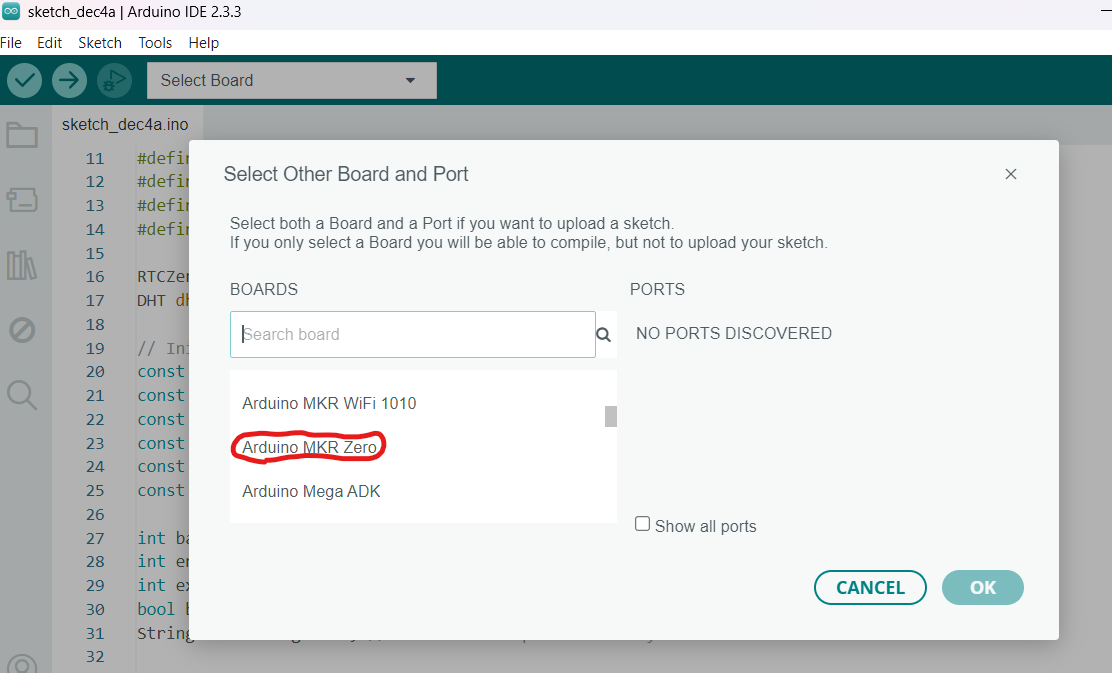
*Installation Success for SD Library*



Click on “Select Board” and choose Arduino MKRZero.

**Figure 31.0**:

*Selecting Arduino MKR Zero Board*

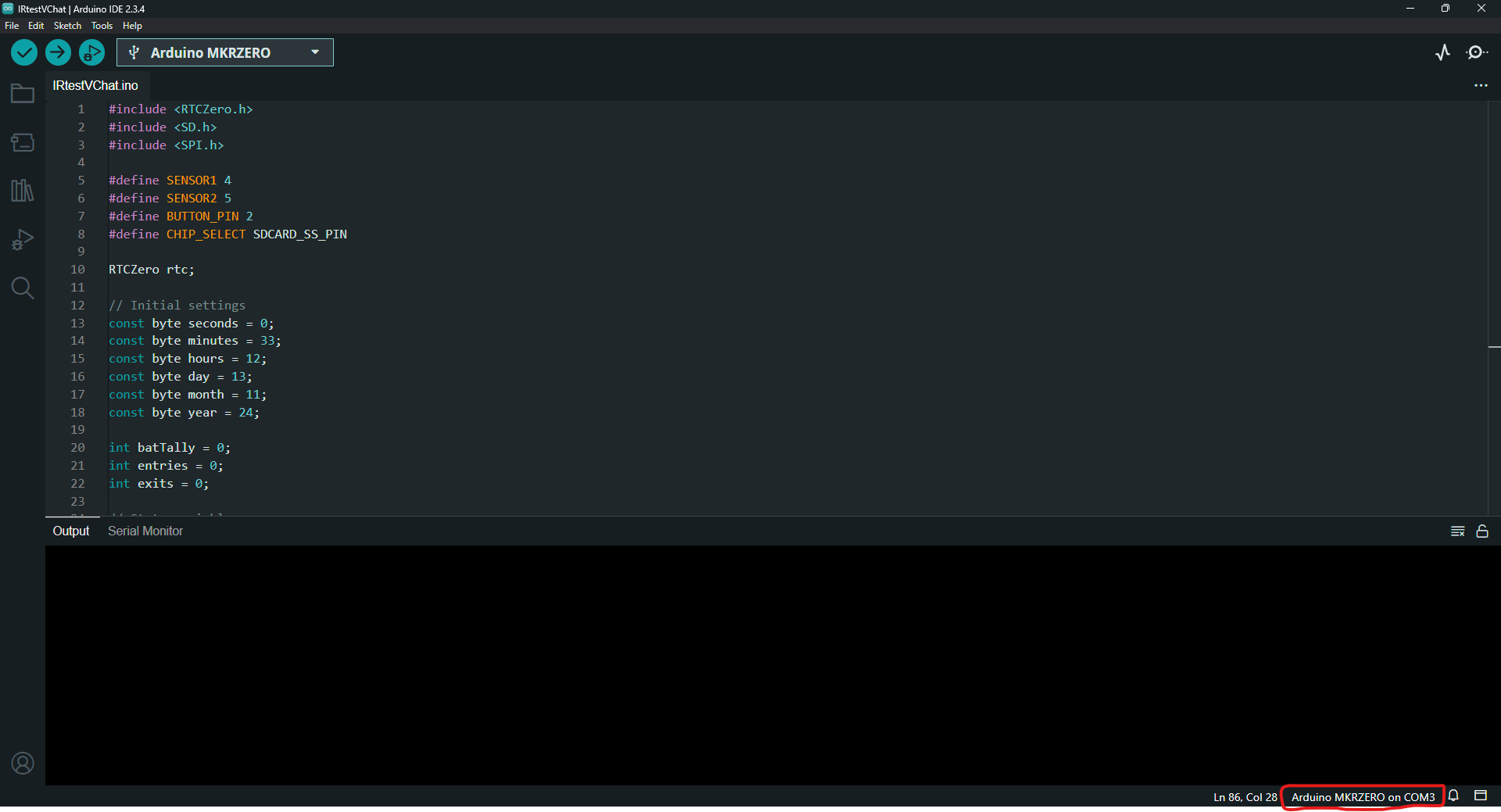


Hit “OK”.

Connect the Arduino board to the device (i.e. laptop or PC). A label on the bottom right of the IDE window should be Arduino MKRZero [connected]

**Figure 32.0**:

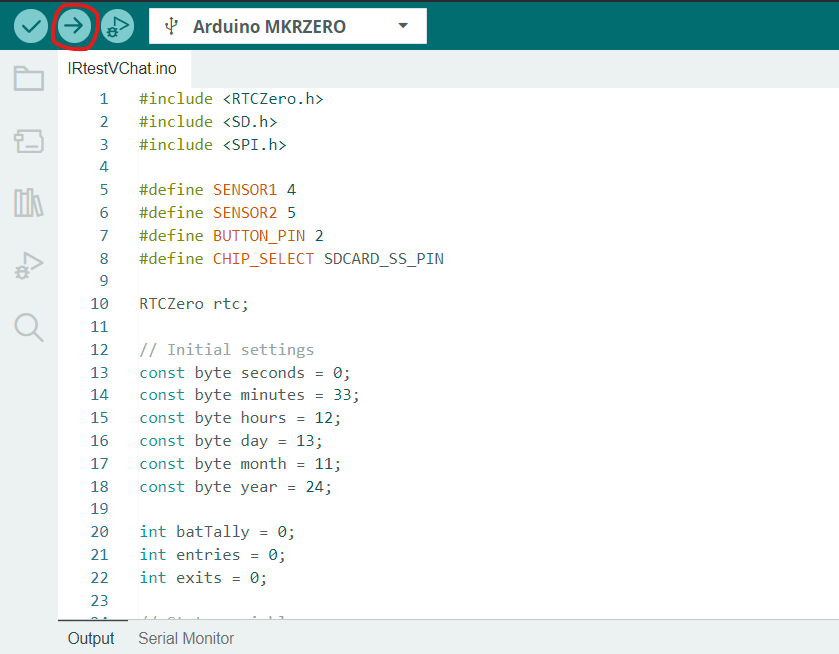
*Arduino MKR Zero Connection Confirmation*



Click on the Upload button to upload the code to the Arduino board:

**Figure 33.0**:

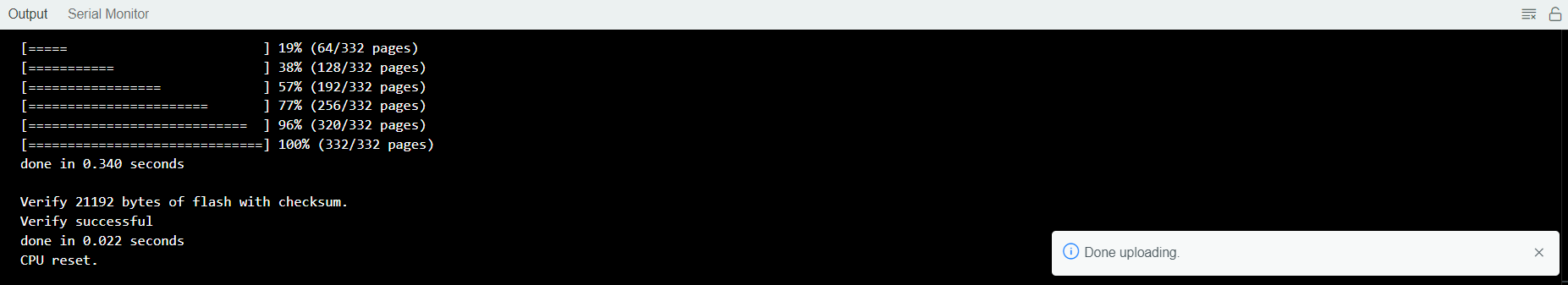
*Uploading Code to Arduino MKR Zero*



A success message should appear in the bottom right of the window with “Done uploading”:

**Figure 34.0**:

*Upload Success Message in Arduino IDE*



***In the event of errors while uploading the code to the Arduino board, refer to*** [***section 5***](#_heading=h.1ci93xb) ***on Troubleshooting and Support***

## 6.4 Testing & Validation

The validation of this bat box monitoring system was carried out through a series of comprehensive tests on the individual components, integrated systems, and the overall design. These tests aimed to assess the functionality, accuracy, and reliability of the sensors, microcontroller, and enclosure, ensuring the final design would meet the required performance standards for sustained usage. The testing was conducted in multiple phases across three prototype iterations, with a focus on sensor performance, data accuracy, system durability, and ease of maintenance.

#### 1. Core Function: Sensor Performance

The primary objective was to ensure that the sensors used in the system (IR break beam sensors) functioned as expected in tracking bat activity.

* **Test Method**: IR break beam sensors were tested for their ability to track bat entry and exit by detecting the sequence in which they were triggered.
* **Results**:
  + The IR break beam sensors achieved a 96% accuracy rate in detecting entrance and exit movements. The system was able to detect the sequence of events reliably, with a near-perfect detection rate for both entrances and exits.
  + **Issues**: None. The IR break beam sensors provided the desired accuracy and speed.

#### 2. Data Processing and Storage

Data accuracy and processing were critical to ensure the system could record bat activity reliably.

* **MicroSD Card Logging**:
  + **Test Method**: The system was tested by logging sensor data (bat entries and exits) to a microSD card. The data was also transferred via Bluetooth to a local device.
  + **Results**:
    - The microSD card successfully recorded sensor data in the correct format. **Bluetooth** We verified the file structure and confirmed the proper storage of data logs.
    - **Data Transfer**: After simulating 25 bat entries and 25 exits, the system demonstrated an accuracy rate of 96% in both data recording and transmission to a local device.
    - **Issues**: None.

#### 3. System Durability and Enclosure Integrity

The system's physical durability was a key factor for sustained outdoor usage. The enclosure needed to protect the sensors, microcontroller, and other components from environmental factors such as water, impact, and temperature fluctuations.:

* + **Test Method**: The bat box enclosure was subjected to a series of environmental tests, including drop tests and waterproofing tests.
  + **Results**:
    - **Drop Test**: The enclosure was dropped five times from a height of 2.2 meters onto a concrete floor. No damage was observed, and the system remained functional.
    - **Waterproofing Test**: The enclosure was submerged in water for 24 hours. Upon removal, no water had entered the enclosure, confirming its watertight integrity.
    - **Issues**: None. The enclosure successfully met all durability and waterproofing requirements.

#### 4. Cable Management and Ease of Maintenance

One of the design goals was to ensure that the system was easy to maintain, particularly in terms of cable management and battery replacement.

* + **Test Method**: The system's cable arrangement was tested for ease of access during battery replacement and maintenance.
  + **Results**:
    - The battery was replaced in under 5 minutes, significantly faster than the anticipated 30-minute maintenance window.
    - Cable management was optimized to prevent tangling, and connectors were designed for easy removal and reattachment.
    - **Issues**: No significant issues were encountered. The system's modular

#### 6. Special Requirements for Sustained Usage

Based on the testing results, the following considerations should be made for sustained use of the bat box monitoring system:

* **Environmental Conditions**: The system's enclosure is well-suited for outdoor use, but attention should be paid to temperature extremes and high humidity environments. The system has passed rigorous waterproofing and impact tests but may need additional ventilation to prevent overheating during prolonged use.
* **Power Supply**: The system’s battery life can be extended with solar panels, which would provide an eco-friendly, sustainable solution for powering the system in outdoor settings.
* **Regular Maintenance**: Though maintenance is straightforward, the sensors should be cleaned periodically to ensure optimal performance. Regular testing should be conducted to verify data accuracy, especially if the system is used for extended periods.

# 7. Conclusions and Recommendations for Future Work

The prototyping process for B.R.U.C.E. was a significant learning experience for our team, particularly regarding sensor selection and system optimization. After exploring several options, we ultimately settled on three sets of break beam IR sensors. This decision came after evaluating other possibilities and facing limitations. Initially, we were using passive infrared (PIR) sensors due to their cost-effectiveness and low power consumption. However, the PIR sensors available to us had delays exceeding 1 second, making them unsuitable for accurately detecting rapid bat movements. While we discovered advanced PIR sensors with delays as low as 2 milliseconds that would have been ideal, shipping constraints prevented their timely integration, but given some extra time would be a swift and substantial addition to our current design.

Additionally, we envisioned incorporating a load cell to track changes in mass within the bat box, which could have provided researchers with valuable data about bat entries and exits, along with enhancing the overall accuracy of our system. Unfortunately, time limitations prevented the implementation of this feature. Given more time, we would have explored the use of multiple load cells to accurately track the changes in mass caused by the bats.

Our power management system also presented an opportunity for further development. With additional time, we could have integrated more kernel-level power saving features, potentially extending the maintenance window for the device from one month to nearly two. These refinements, while not possible within the project's constraints, represent an opportunity for future work and an upgrade path for our tracking system.

**APPENDICES**

# APPENDIX I: Design Files

<https://makerepo.com/WaleedSakalla/2216.gng1103c2-screws-loose>

1. The libraries DHT, RTCZero, SD and SPI are installed from the Arduino IDE (see section 6.3.3) [↑](#footnote-ref-0)