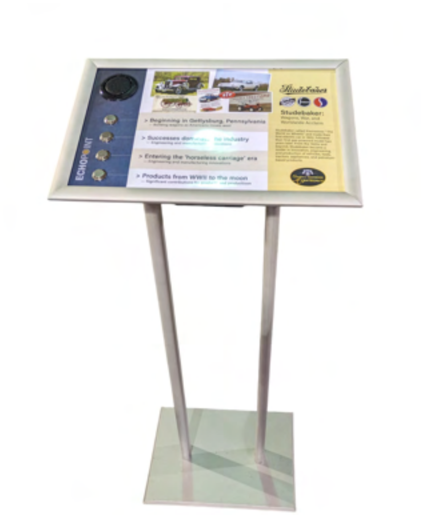
ECHOPOINT


ECHOPOINT
> Beginning in Gettysburg, Pennsylvania
> Entering the 'horseless carriage' era
> Products from WWII to the moon
Studebaker:
Studebaker

Contact: Andrew D. Marques

Website: www.AndrewDMarques.com

Email: ____________________

____

Address: ____________________

____

Phone: ____________________

Version: 5
Updated: 8/9/2024

INTRODUCTION

This manual serves as a comprehensive guide for using and maintaining EchoPoint. EchoPoint is a small device that enables simple audio playback at the push of a button. One of EchoPoint's defining features is that it can be battery operated, or powered using a 6V DC source. Using the supplied 6V 6Ah LiFePO4 battery, an expected battery life is 107 active hours, 6,400 button presses, or 3-5 months in a museum setting before charging. This can be extended to about 214 active hours, 12,800 button presses, or 6-10 months if two 6V 6Ah batteries are used. Batteries recharge in about 5 hours with the supplied battery charger.

Core features include:

1. **Battery Operated** - external power source not required
2. **Deep Sleep Mode** - device enters deep sleep after 1 minute of inactivity
3. **Volume Control** - customizable to most spaces
4. **Customizable Audio** - any mp3 file <28MB large

Contributors include:

Individual	Contribution
Andrew D. Marques	Electronics, software development, hardware, and initial prototyping
"Tiger" Tom Ehrhart	Significant help with stand fabrication and stand completion, topic ideas
Rob Kain	Original idea conception, funding procurement, organizing audio collections, topic ideas
Kelly Vanek	Stand graphic design
Stanley Sipko	Museum curator
Casey Allyn	Voice actor
Lyn Weiss	Voice actor

TABLE OF CONTENTS

QUICKSTART GUIDE

Operating Steps:

1. Press any of the blue buttons to wake up the device and begin hearing audio.
2. Continue to press the other blue buttons to sample audio files, or press the same button to replay the previous message.

Recharge Batteries:

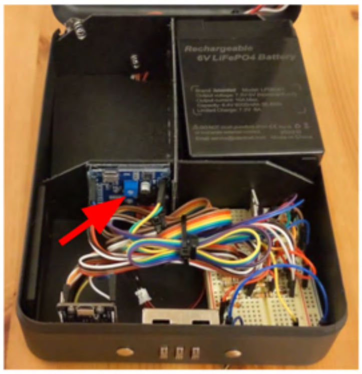
Video Tutorial (link and QR Code):
echopoint_v5.01_recharge-batteries.mp4

1. To recharge the batteries, first remove the batteries from the device.
2. Use the supplied charger to charge the batteries by connecting the positive lead of the charger to the positive and the battery and the negative lead of the charger to the negative lead of the battery.
3. Plug in the charger to the wall.
    a. NOTE: The LED on the charger will be green when unplugged from the wall if connected to the batteries -- this does not mean the battery is charged. The LED color is only valid when the charger is connected to the wall.
4. The light will be red while charging and green when done charging.
5. When the light turns green, disconnect the charger from the wall.
6. Now disconnect the charger from the battery.
7. Place the battery back into the EchoPoint.
    a. NOTE: the leads on the battery must be in contact with the metal springs, but do not have to be compressed against the springs.

Change Volume:

Video Tutorial (link and QR Code):
echopoint_v5.01_change-volume.mp4

8. To change the playback volume, open EchoPoint and locate the audio amplifier.
9. Use a small philips screw driver to turn the volume potentiometer (white screw in the blue box). Clockwise increases the volume and counter clockwise decreases the volume.
    a.

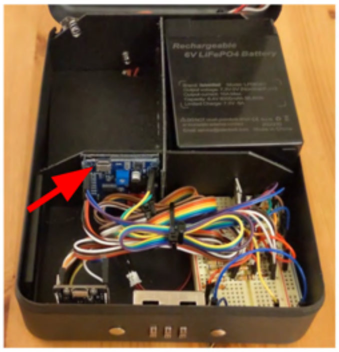


10. Close and lock the box to prevent tampering.
11. The audio is now adjusted.

Change Audio Programs:

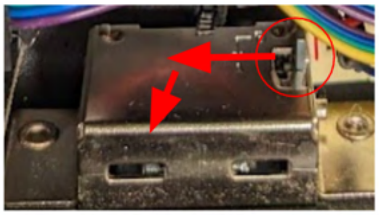
Video Tutorial (link and QR Code):
echopoint_v5.01_change-audio.mp4

1. Adding new programs is easy, begin by opening the housing.
2. Removing the batteries to disconnect the power source..
3. Locate the DV-SV8F board and connect a microUSB cable from a computer to the board.



4. Your computer will treat the DV-SV8F board like a USB drive showing the file contents.
5. Prepare the audio files to be played.
   a. Audio files must be .mp3 format.
   b. Files must be named as "00001.mp3", "00002.mp3", "00003.mp3", or "00004.mp3".
6. Download the audio files of interest to the DV-SV8F board as files "00001.mp3", "00002.mp3", "00003.mp3", or "00004.mp3".
7. Disconnect the DV-SV8F board from your computer by unplugging the microUSB cable.
8. Reconnect the battery power supply.
9. Test your audio files by pushing blue membrane buttons.
10. Close and lock the device.
11. Your audio files are now updated.

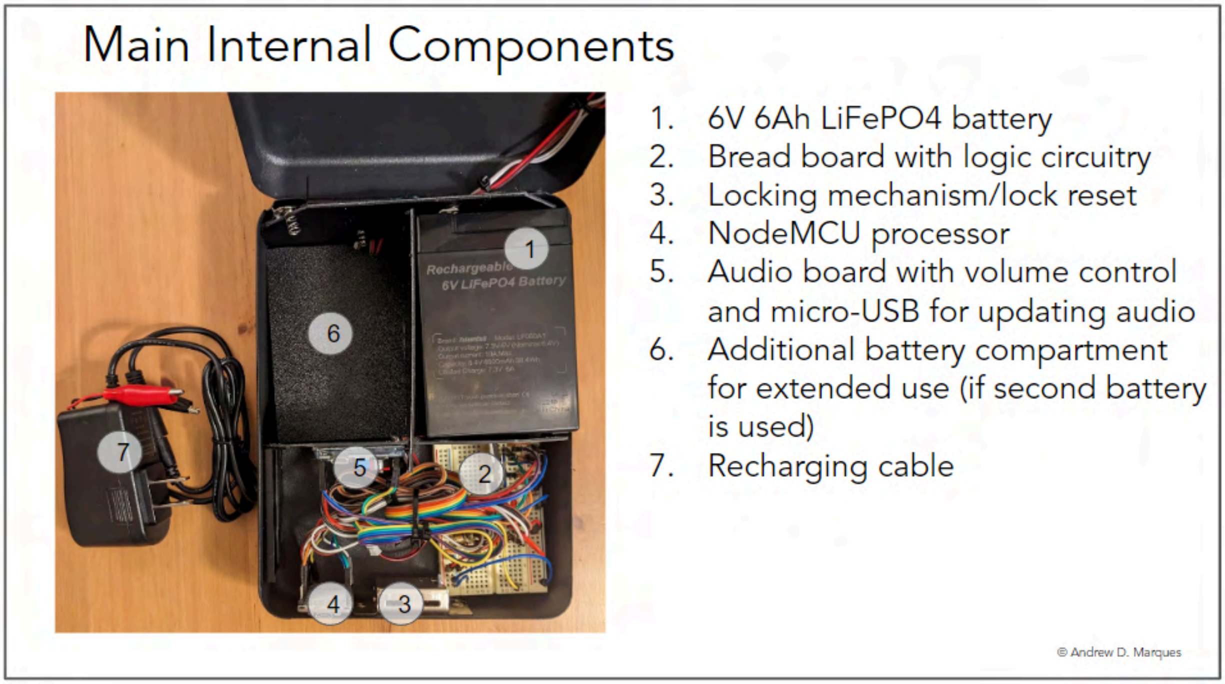
Change Combination:

1. To change the lock combination on the device, first unlock and open the box.
2. There is a lever on the box that can be pushed to the left and toward you (if you are looking down at the box with the lock closest to you.
   a. 
3. Set the new combination using the dial on the outside of the box.
4. Reverse the motion of the lever so that it is back in its home position (as pictured in step 2).
5. Close and lock the device.
6. Your new combination is set.

Long-Term Storage:

1. EchoPoint should be stored in dry cool spaces without rodents. This device should be thought of like a computer, so make sure that it is stored appropriately.
2. Check the integrity of the wiring connections.
3. Consider labeling the box with the combination code so that it can be opened in the future.

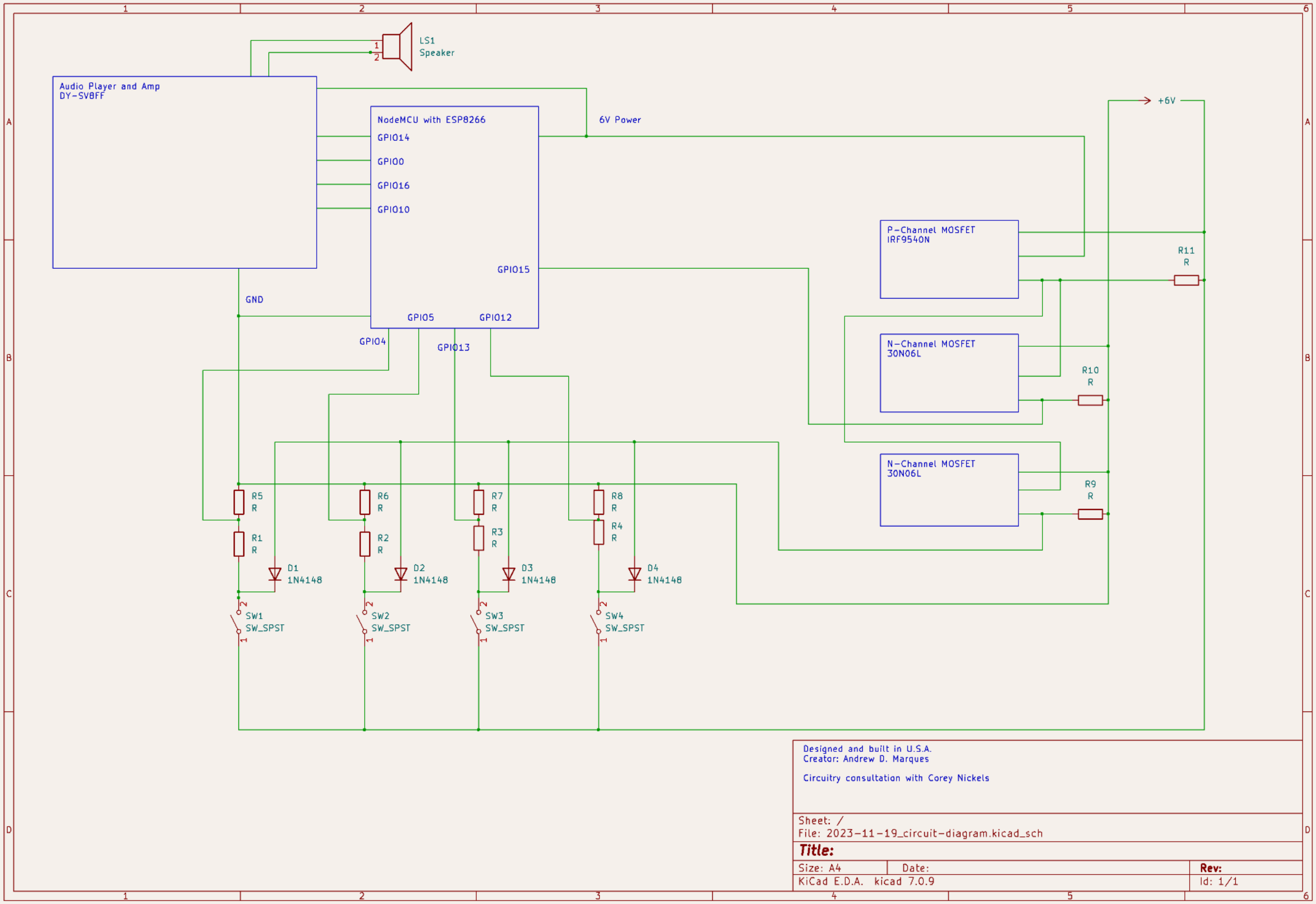
DIAGRAMS



Main Internal Components

1. 6V 6Ah LiFePO4 battery
2. Bread board with logic circuitry
3. Locking mechanism/lock reset
4. NodeMCU processor
5. Audio board with volume control and micro-USB for updating audio
6. Additional battery compartment for extended use (if second battery is used)
7. Recharging cable

© Andrew D. Marques



Designed and built in U.S.A.
Creator: Andrew D. Marques

Circuitry consultation with Corey Nickels

Sheet: /
File: 2023-11-19_circuit-diagram.kicad_sch

Title:

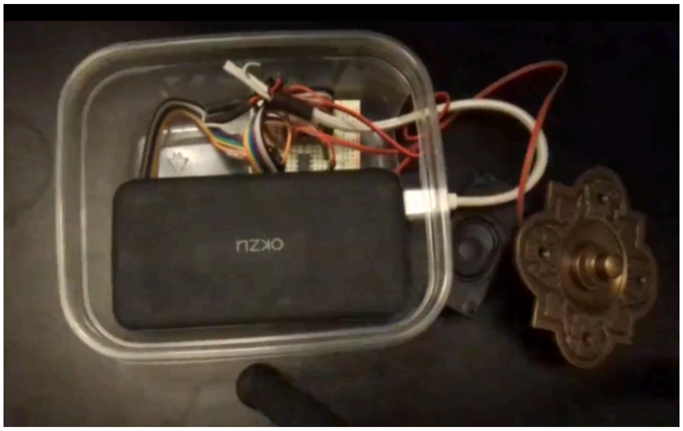
Size: A4 Date: **Rev:**
KiCad E.D.A. kicad 7.0.9 Id: 1/1

BUILD DOCUMENTATION

8/23/2023

Beginning programming with code:
2023-08-23_aaca-button_v1.00

8/27/2023 First "operational" prototype completed. A crude proof of concept without audio amplifier.



9/6/2023

Programming:

```
Sep  6 08:25 aaca_button_v1_06_energery_saver
Sep  6 08:27 aaca_button_v1_07_energery_saver
Sep  6 08:50 aaca_button_v1_08_energery_saver
Sep  6 08:54 aaca_button_v1_09_energery_saver
Sep  6 08:59 aaca_button_v1_10_energery_saver
Sep  6 09:04 aaca_button_v1_11_energery_saver
```

9/12/2023

Programming:

```
Sep 12 18:37 aaca_button_v1_12_energery_saver
Sep 12 19:21 aaca_button_v1_13_relay
Sep 21 21:36 aaca_button_v1_14_back_to_basics
Sep 13 19:01 aaca_button_v2_00_arduino
Sep 13 19:41 aaca_button_v2_01_arduino
Sep 13 19:45 aaca_button_v2_02_arduino-relay
Sep 13 20:04 aaca_button_v2_03_arduino-relay
Sep 13 20:43 aaca_button_v2_04_arduino-two-relay
Sep 13 20:43 aaca_button_v2_05_arduino-two-relay
Sep 13 20:46 aaca_button_v2_06_arduino-single-relay
Sep 13 20:55 aaca_button_v2_07_start-low-relay
```

9/21/2023

Programming:

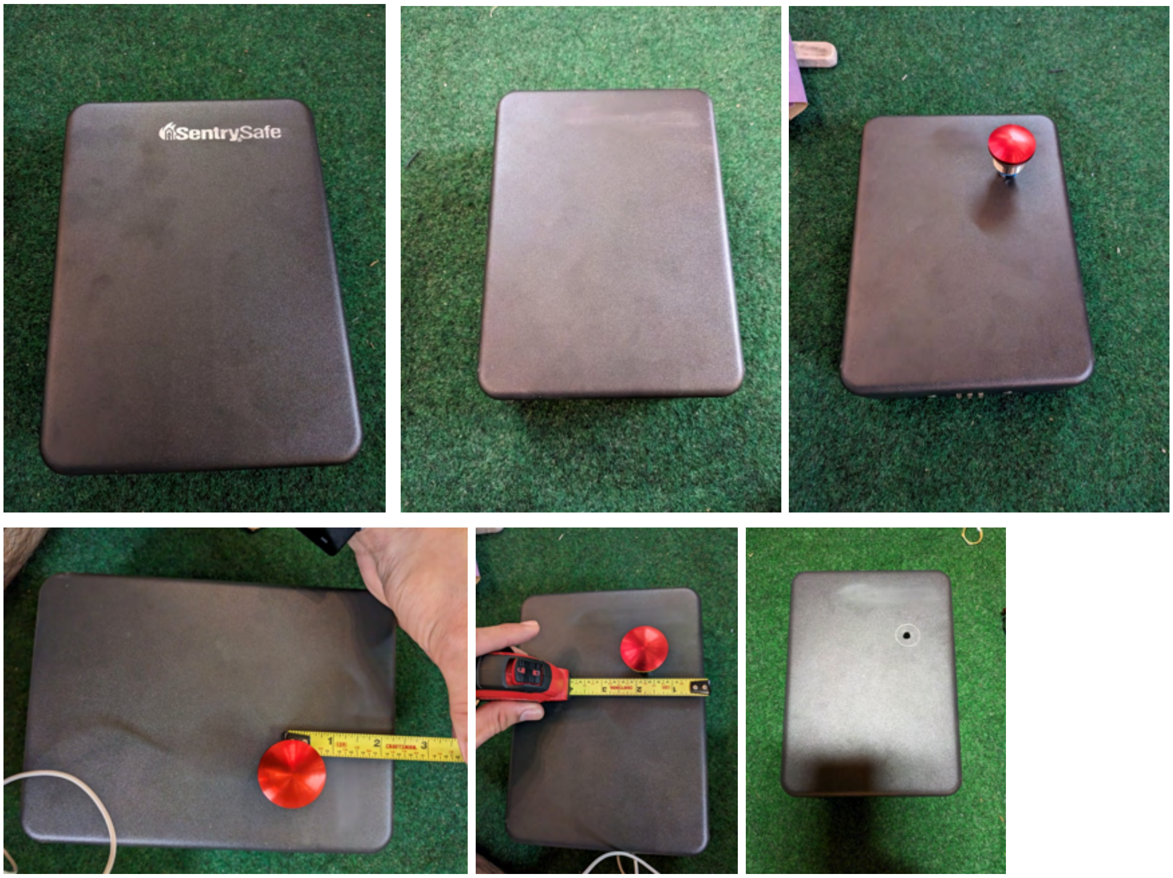
```
Sep 21 08:02 aaca_button_v3_00_attiny85-relay
Sep 21 08:08 aaca_button_v3_01_attiny85-relay
Sep 21 08:10 aaca_button_v3_02_attiny85-relay
Sep 21 08:14 aaca_button_v3_03_attiny85-relay
Sep 21 21:56 aaca_button_v3_04_attiny85-relay
Sep 21 20:53 aaca_button_v3_05_attiny85-relay-with-audio
Sep 21 20:56 aaca_button_v3_06_attiny85-relay-with-audio-module
Sep 21 21:16 aaca_button_v3_07_attiny85-relay-with-audio-module
Sep 21 21:17 aaca_button_v3_08_attiny85-relay-with-audio-module
Sep 21 21:56 aaca_button_v3_09_attiny85-relay
Sep 21 21:24 aaca_button_v1_05
```

9/25/2021

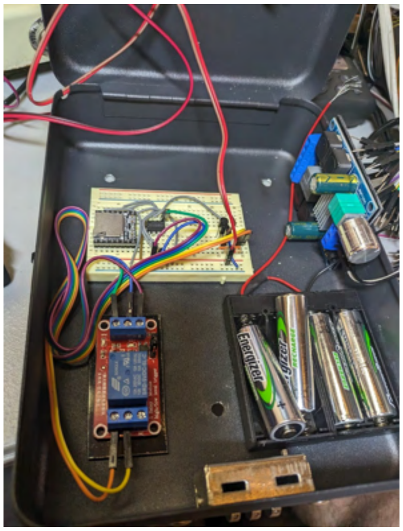
Programming:

```
Sep 25 21:31 aaca_button_v3_10_attiny85-relay
Sep 25 21:50 aaca_button_v3_11_attiny85-relay-loud-speaker
```

9/26/2023 Fabricating case.



9/30/2023 Assembling next iteration of prototype. Placing electronics inside housing and creating opening for speaker.



Programming:

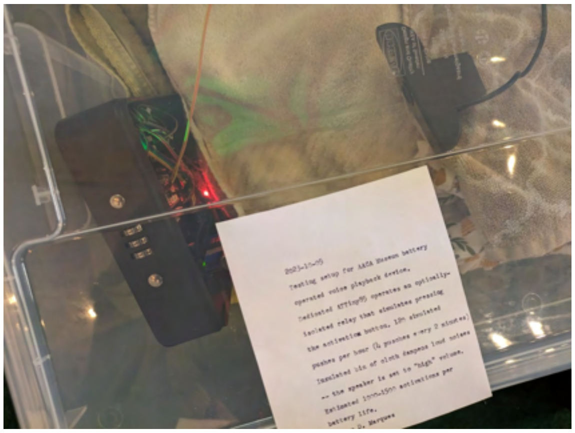
```
Sep 30 22:44 aaca_button_v3_12_attiny85-get-the-number-of-files-to-be-played
```

10/1/2023 Testing current draw on device.

```
Oct  1 18:39 aaca_button_v3_13_attiny85-get-the-number-of-files-to-be-played
Oct  1 18:52 aaca_button_v3_14_attiny85-troubleshooting-get-the-number-of-fi
```

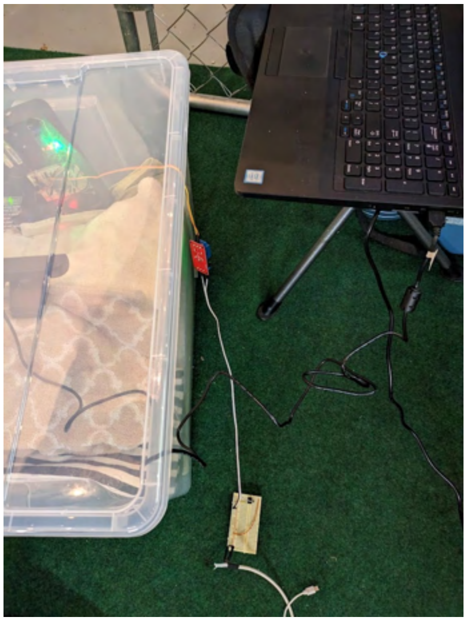
10/2/2023 Completed exterior

10/05/2023 Testing set up for EchoPoint. A dedicated external ATTiny85 operates an optically isolated relay that simulate pressing the activation button. 120 simulated pushes per hour (4 pushes every 2 minute). The device is insulated in a bit of cloth to dampen the loud sounds while it runs for 10-12 hours before draining the batteries. The speaker is set to the highest volume without distortion. Estimated 1000-1500 button pushes per battery life.



2023-10-05
Testing setup for AAGA Museum battery
operated voice playback device.
Dedicated ATTiny85 operates an optically-
isolated relay that simulates pressing
the activation button. 120 simulated
pushes per hour (4 pushes every 2 minutes)
Insulated bit of cloth dampens loud noises
-- the speaker is set to "high" volume.
Estimated 1000-1500 activations per
battery life.
D. Marquez

A laptop and camera is used to continuously record the simulated button pressing over the 10-12 hour testing periods. Any failures would be recorded on camera.



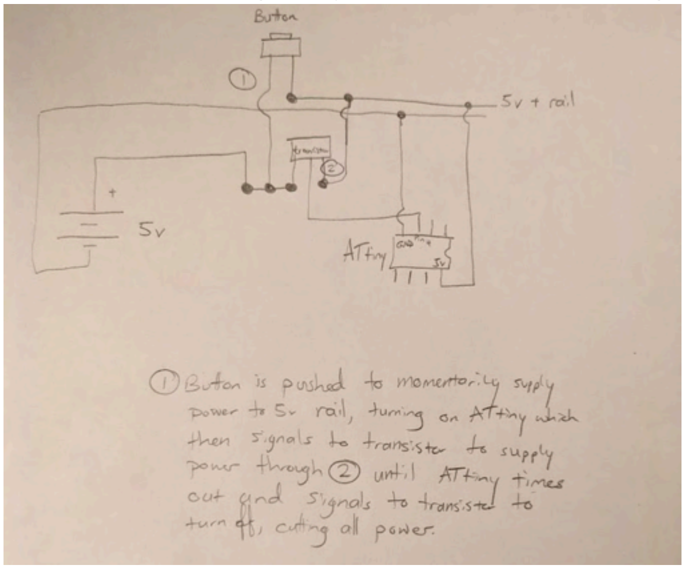
The image above showed the auxiliary optically isolated relay and ATTiny85 simulating button pressing.

10/12/2023

Programming:
Oct 12 07:37 echopoint_v3_15_attiny85-relay-loud-speaker

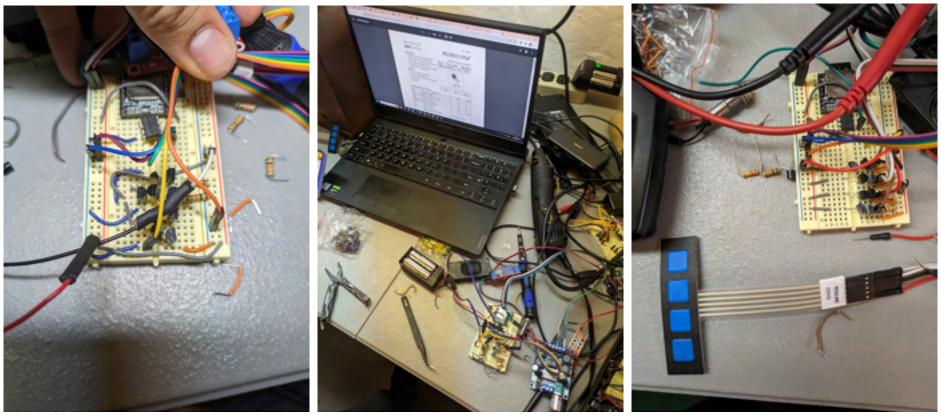
Building documentation and manual.

10/21/2023 Consulting Corey Nickels on transistor design.

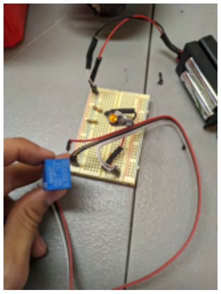


① Buton is pushed to momentorily supply power to 5v rail, turning on ATtiny which then signals to transister to supply power through ② until ATtiny times out and signals to transister to turn off, cutting all power.

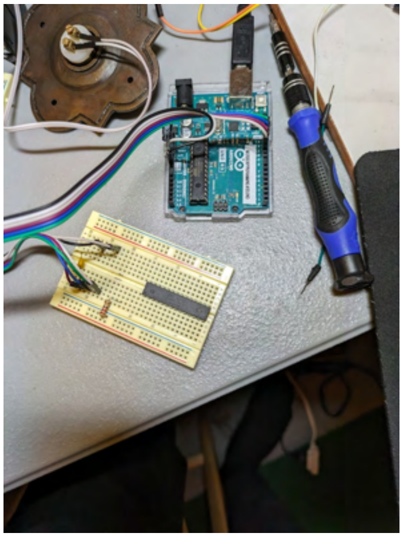
10/22/2023 Modifying transistor circuitry and testing one button one sound membrane buttons.

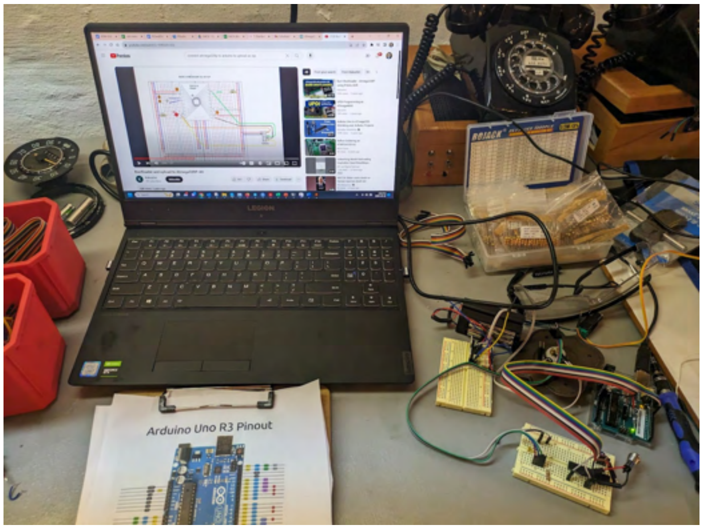


10/24/2023 Comparing power efficiency with naked transistors. This would allow a simpler design, but at a high cost of power consumption.

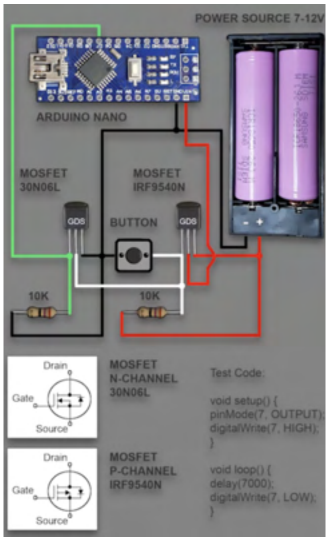


10/25/2023 Attempting to program ATMEGA328P or Arduino UnoR3 (without bootloader) using an Arduino Uno R3 as ISP





10/26/2023 Redesigning power transistor circuitry, turning to the internet as inspiration and finding a circuit that works with our applications (image below comes from @PropDork).

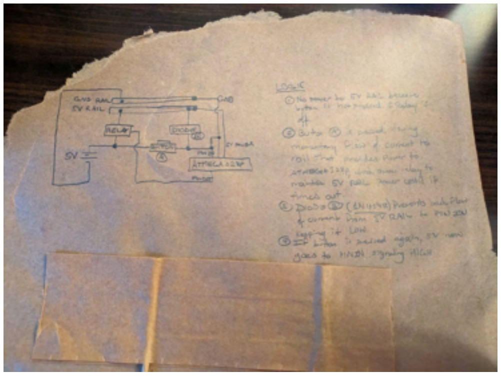


Test Code:

```
void setup() {
pinMode(7, OUTPUT);
digitalWrite(7, HIGH);
}

void loop() {
delay(7000);
digitalWrite(7, LOW);
}
```

10/27/2023 Redesigning transistor circuitry yet again.



LOGIC

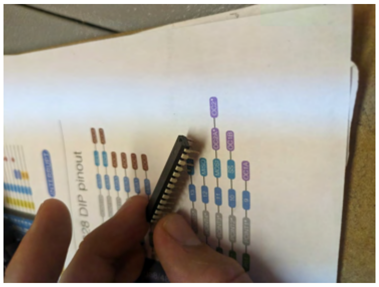
① No power to 5V RAIL because button is not pressed & relay is off

② Button Ⓐ is pressed, causing momentary flow of current to rail that provides power to ATMEGA328P, which then relay to maintain 5V RAIL power until it times out.

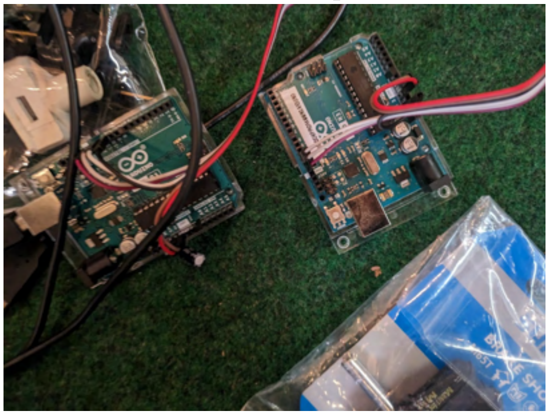
③ Diode Ⓑ (1N4148) prevents back flow of current from 5V RAIL to PIN [illegible] keeping it LOW

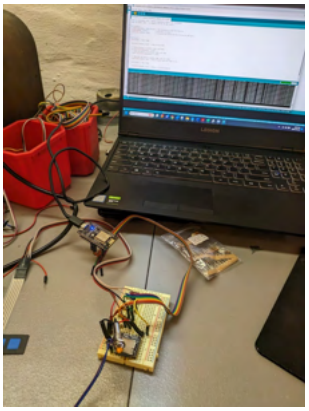
④ If button is pressed again, 5V now goes to PIN [illegible] signaling HIGH

10/31/2023 Still attempting to successfully program ATMEGA328P



11/1/2023 Final attempts to use Arduino Uno R3 without bootloader. First successful attempts to program a NodeMCU with ESP8266 processor.



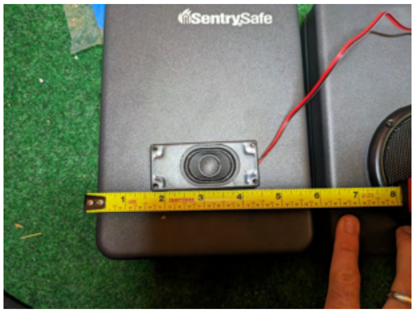


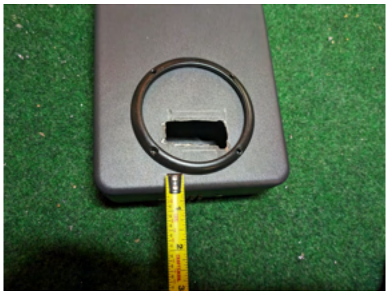
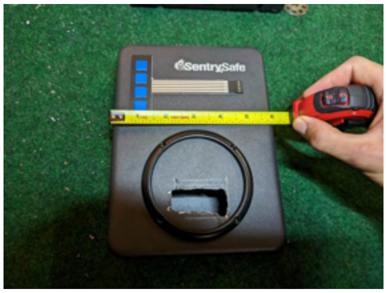
```
Nov  1 19:37 echopoint_v4.00_nodemcu
Nov  1 19:38 echopoint_v4.01_nodemcu
Nov  1 19:43 echopoint_v4.02_nodemcu
Nov  1 19:49 echopoint_v4.03_nodemcu
Nov  1 19:56 echopoint_v4.04_nodemcu_connected_to_dfplayer_mini
Nov  1 20:02 echopoint_v4.05_nodemcu_connected_to_dfplayer_mini
Nov  1 20:03 echopoint_v4.06_nodemcu_connected_to_dfplayer_mini
Nov  1 20:08 echopoint_v4.07_nodemcu_connected_to_dfplayer_mini
Nov  1 20:09 echopoint_v4.08_nodemcu_connected_to_dfplayer_mini
Nov  1 20:15 echopoint_v4.09_nodemcu_connected_to_dfplayer_mini
Nov  1 20:22 echopoint_v4.10_nodemcu_connected_to_dfplayer_mini
Nov  1 20:28 echopoint_v4.11_nodemcu_connected_to_dfplayer_mini
Nov  1 21:13 echopoint_v4.12_nodemcu_connected_to_dfplayer_mini_random
Nov  1 21:14 echopoint_v4.13_nodemcu_connected_to_dfplayer_mini_random
Nov  1 21:18 echopoint_v4.14_nodemcu_connected_to_dfplayer_mini_random
Nov  1 21:22 echopoint_v4.15_nodemcu_connected_to_dfplayer_mini_random
Nov  1 21:28 echopoint_v4.16_nodemcu_connected_to_dfplayer_mini_random
Nov  1 21:33 echopoint_v4.17_nodemcu_connected_to_dfplayer_mini_random
```

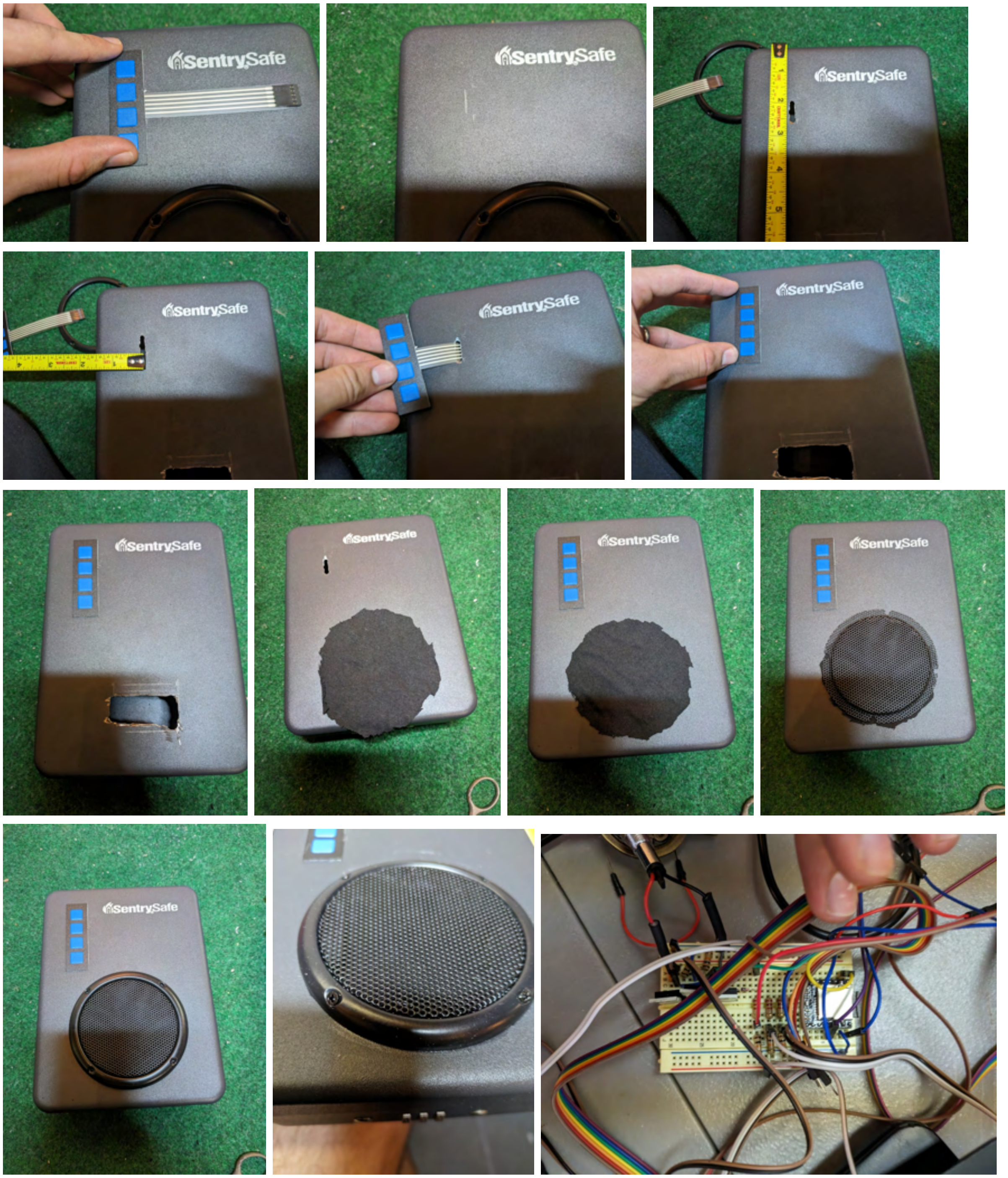
11/2/2023 Programming

```
Nov  2 08:21 echopoint_v4.18_nodemcu_connected_to_dfplayer_mini_new_setup
Nov  2 08:25 echopoint_v4.19_nodemcu_connected_to_dfplayer_mini_new_setup
Nov  2 08:27 echopoint_v4.20_nodemcu_connected_to_dfplayer_mini_new_setup
Nov  2 16:43 echopoint_v4.21_nodemcu_connected_to_dfplayer_mini_io_setup
Nov  2 17:28 echopoint_v4.22_nodemcu_connected_to_dfplayer_mini_io_setup
Nov  2 18:05 echopoint_v4.23_nodemcu_connected_to_dfplayer_mini_io_setup
Nov  2 18:09 echopoint_v4.24_nodemcu_connected_to_dfplayer_mini_io_setup
Nov  2 18:12 echopoint_v4.25_nodemcu_connected_to_dfplayer_mini_io_setup
Nov  2 18:21 echopoint_v4.26_nodemcu_connected_to_dfplayer_mini_io_setup
Nov  2 18:30 echopoint_v4.27_nodemcu_connected_to_dfplayer_mini_io_setup
Nov  2 18:33 echopoint_v4.28_nodemcu_connected_to_dfplayer_mini_test_blink
Nov  2 18:35 echopoint_v4.29_nodemcu_connected_to_dfplayer_mini_test_blink
Nov  2 18:44 echopoint_v4.30_nodemcu_connected_to_dfplayer_mini_test_blink
Nov  2 18:47 echopoint_v4.31_nodemcu_connected_to_dfplayer_mini_test_blink
Nov  2 18:50 echopoint_v4.32_nodemcu_connected_to_dfplayer_mini_test_blink
Nov  2 18:52 echopoint_v4.33_nodemcu_connected_to_dfplayer_mini_test_blink
Nov  2 18:54 echopoint_v4.34_nodemcu_connected_to_dfplayer_mini_blink_work
Nov  2 18:54 echopoint_v4.34_nodemcu_connected_to_dfplayer_mini_test_blink_w
Nov  2 18:56 echopoint_v4.35_nodemcu_connected_to_dfplayer_mini_blink
Nov  2 18:59 echopoint_v4.36_nodemcu_connected_to_dfplayer_mini_blink
Nov  2 20:18 echopoint_v4.37_nodemcu_connected_to_dfplayer_mini_blink
Nov  2 20:24 echopoint_v4.38_nodemcu_connected_to_dfplayer_taking_it_back
Nov  2 20:28 echopoint_v4.39_nodemcu_connected_to_dfplayer_taking_it_back
Nov  2 20:41 echopoint_v4.40_nodemcu_connected_to_dfplayer_taking_it_back
Nov  2 20:45 echopoint_v4.41_nodemcu_connected_to_dfplayer_taking_it_back
Nov  2 20:46 echopoint_v4.42_nodemcu_connected_to_dfplayer_taking_it_back
Nov  2 20:52 echopoint_v4.43_nodemcu_connected_to_dfplayer_serial_monitor
Nov  2 20:56 echopoint_v4.44_nodemcu_connected_to_dfplayer_taking_it_back
Nov  2 21:01 echopoint_v4.45_nodemcu_connected_to_dfplayer_taking_it_back
Nov  2 21:03 echopoint_v4.46_nodemcu_connected_to_dfplayer_taking_it_back
Nov  2 21:10 echopoint_v4.47_nodemcu_connected_to_dfplayer_last_one_worked
Nov  2 21:12 echopoint_v4.48_nodemcu_connected_to_dfplayer_simple_stateNov  2 21:14
echopoint_v4.49_nodemcu_connected_to_dfplayer_simple_stateNov  2 21:21
echopoint_v4.50_nodemcu_connected_to_dfplayer_short_state
Nov  2 21:23 echopoint_v4.51_nodemcu_connected_to_dfplayer_debouncing
Nov  2 21:26 echopoint_v4.52_nodemcu_connected_to_dfplayer_debouncing
Nov  2 21:30 echopoint_v4.53_nodemcu_connected_to_dfplayer_short_state
Nov  2 21:44 echopoint_v4.54_nodemcu_connected_to_dfplayer_cycle_test_serial
Nov  2 21:48 echopoint_v4.55_nodemcu_connected_to_dfplayer_cycle_last_worked
Nov  2 21:50 echopoint_v4.56_nodemcu_connected_to_dfplayer_cycle_last_worked
```

11/3/2023 Building a new housing for the one button one sound design.

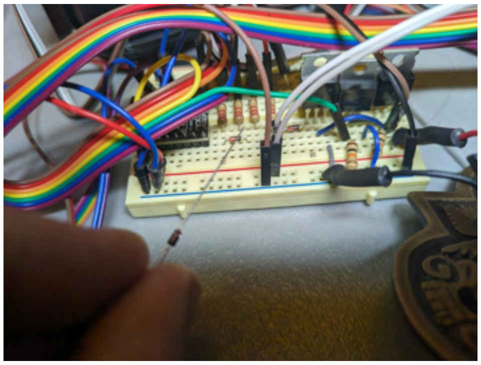
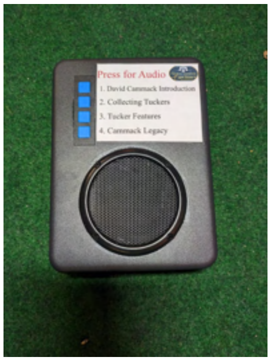


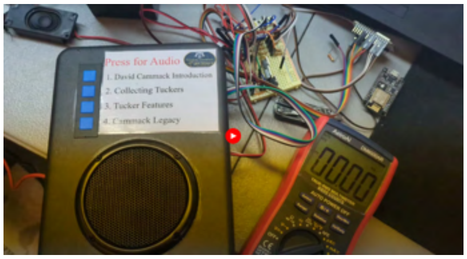





Nov 3 20:15 echopoint_v4.57_nodemcu_connected_to_dfplayer_serial_againNov 3 21:02
echopoint_v4.58_nodemcu_connected_to_dfplayer_cycle_last_worked
Nov 3 21:39 echopoint_v4.59_nodemcu_connected_to_dfplayer_test_shutoffNov 4 18:08
echopoint_v4.60_nodemcu_connected_to_dfplayer_transistor_sleep

11/4/2023 First use of diodes in the project as the circuits are redesigned for buttons to handle independent activation of the processor while have equal activation of powering on the power transistors. Also designing example signage and measuring power consumption.

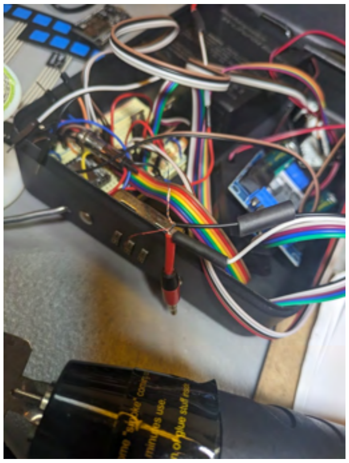


```
Nov  4 18:39 echopoint_v4.61_nodemcu_connected_to_dfplayer_cycle_works
Nov  4 18:42 echopoint_v4.62_nodemcu_connected_to_dfplayer_cycle_works_with_
Nov  4 18:47 echopoint_v4.63_nodemcu_connected_to_dfplayer_shifted_buttons
Nov  4 19:43 echopoint_v4.64_nodemcu_connected_to_dfplayer_shifted_d8
Nov  4 20:12 echopoint_v4.65_nodemcu_connected_to_dfplayer_faster_button_det
Nov  4 20:17 echopoint_v4.66_nodemcu_connected_to_dfplayer_extra_delay
Nov  4 20:22 echopoint_v4.67_nodemcu_connected_to_dfplayer_extra_delay_first
```

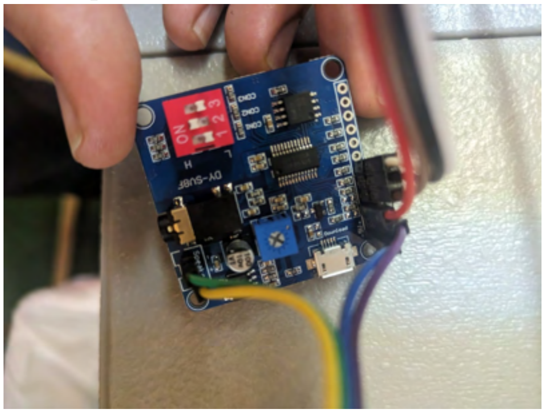
11/5/2023 Troubleshooting output pin issues. Not all pins can work on bootup -- this is not well documented.

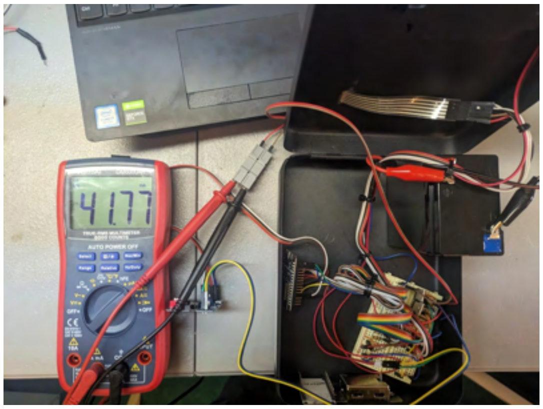
```
Nov  5 08:14 echopoint_v4.68_nodemcu_connected_to_dfplayer_extra_fix_delay
Nov  5 09:22 echopoint_v4.68_nodemcu_connected_to_dfplayer_extra_fix_pin4
Nov  5 09:44 echopoint_v4.69_nodemcu_connected_to_dfplayer_last_one_works
```

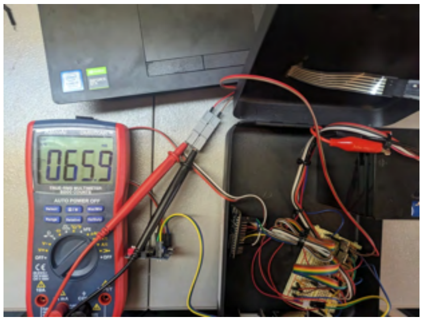
11/6/2023 Updating amplifier and soldering new connections.



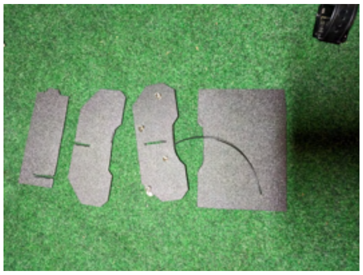
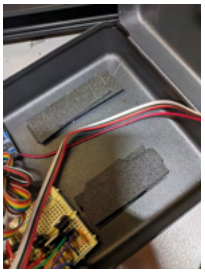
11/11/2023 Replacing previous audio amplifier with one that is more efficient, plays the audio with a dedicated pin for each audio file, but it does not have the same max loudness. Now using DV-SV8F. Total standby power consumption: 41.8mA. Total active power consumption: 66mA-110mA. Deep sleep power consumption (when device is inactive for >1 minute): 0mA.

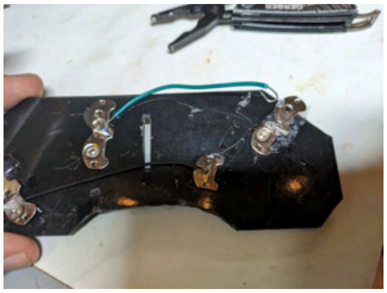






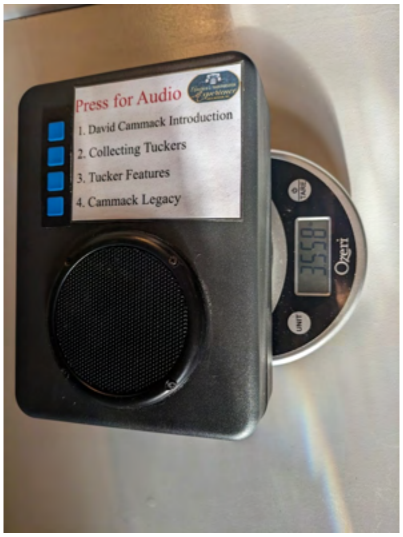
11/12/2023 Designing and building housing for batteries. Housing cut from plastic and fits together like a puzzle. Hot glue was temporarily used to hold some componenets together. Springs from a separate battery housing was used for leads to the batteries. Wires were soldered together. E6000 was used as a final solution to the hot glue.

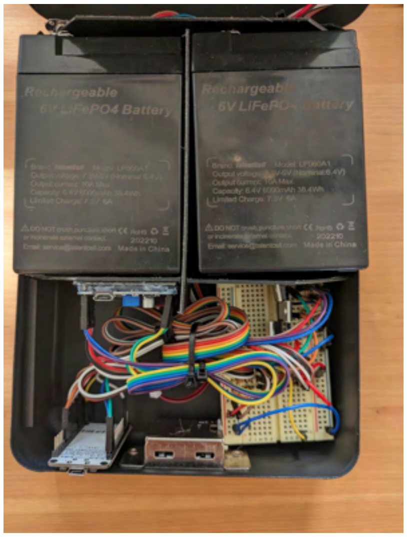



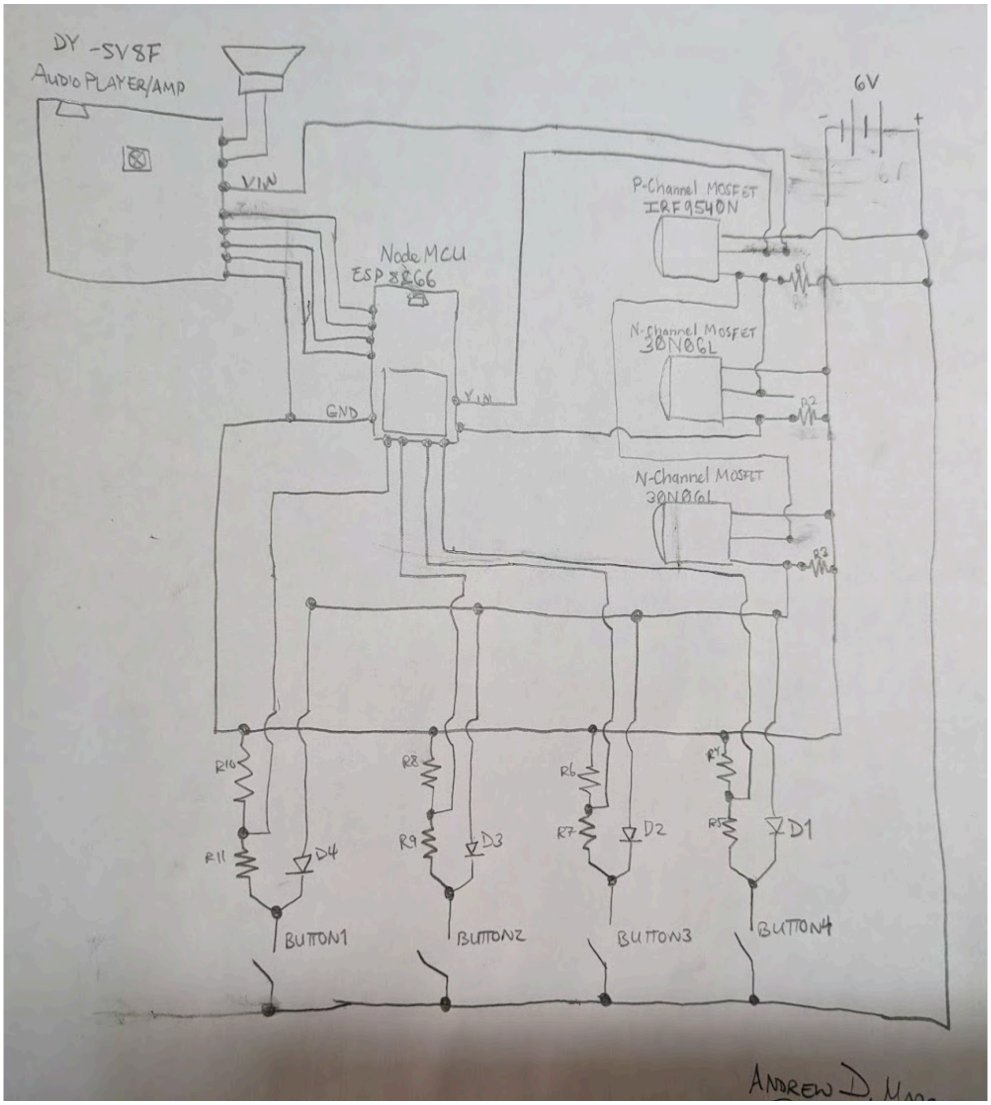


```
Nov 12 09:49 echopoint_v5.02_nodemcu_connected_to_dy_player
Nov 12 10:08 echopoint_v5.03_nodemcu_connected_to_dy_player
Nov 12 10:22 echopoint_v5.04_nodemcu_connected_to_dy_player
Nov 12 10:35 echopoint_v5.05_nodemcu_connected_to_dy_player
Nov 12 10:41 echopoint_v5.06_nodemcu_connected_to_dy_player_last_almost_work
Nov 12 10:50 echopoint_v5.07_nodemcu_connected_to_dy_player_serial
Nov 12 11:01 echopoint_v5.08_nodemcu_connected_to_dy_player_serial
Nov 12 11:01 echopoint_v5.09_nodemcu_connected_to_dy_player_serial
Nov 12 12:19 echopoint_v5.10
Nov 12 12:35 echopoint_v5.11
Nov 12 12:39 echopoint_v5.12_getting_serial_working
Nov 12 12:44 echopoint_v5.13_getting_serial_working
Nov 12 12:50 echopoint_v5.14_getting_serial_working_last_worked
Nov 12 12:50 echopoint_v5.15_getting_serial_working
Nov 12 12:55 echopoint_v5.16_getting_serial_working_last_worked
Nov 12 13:01 echopoint_v5.17_getting_serial_working_last_worked
Nov 12 13:08 echopoint_v5.18_getting_serial_working_multiplexed
Nov 12 13:13 echopoint_v5.19_getting_serial_working_multiplexed
Nov 12 13:16 echopoint_v5.20_getting_serial_working_multiplexed
Nov 12 13:19 echopoint_v5.21_getting_serial_working_multiplexed
Nov 12 13:21 echopoint_v5.22_getting_serial_working_multiplexed
Nov 12 13:26 echopoint_v5.23_getting_serial_working_multiplexed
Nov 12 13:30 echopoint_v5.24_getting_serial_working_multiplexed
Nov 12 13:31 echopoint_v5.25_getting_serial_working_multiplexed
Nov 12 13:32 echopoint_v5.26_getting_serial_working_multiplexed
Nov 12 15:46 echopoint_v5.27_getting_serial_working_multiplexed_last_worked
Nov 12 15:50 echopoint_v5.28_getting_serial_working_multiplexed
Nov 12 15:56 echopoint_v5.29_getting_serial_working_multiplexed
Nov 12 16:00 echopoint_v5.30_getting_serial_working_multiplexed_last_really_
Nov 12 16:04 echopoint_v5.31_getting_serial_working_multiplexed
Nov 12 16:04 echopoint_v5.32_getting_serial_working_multiplexed
Nov 12 16:06 echopoint_v5.33_getting_serial_working_multiplexed_last_worked
Nov 12 16:12 echopoint_v5.34_getting_serial_working_multiplexed_timing
Nov 12 16:19 echopoint_v5.35_getting_serial_working_multiplexed_timing
Nov 12 16:31 echopoint_v5.36_getting_serial_working_initial_pause_last_works
Nov 12 16:36 echopoint_v5.37_getting_serial_working_initial_pause
Nov 12 16:36 echopoint_v5.37_getting_serial_working_initial_pause_last_works
Nov 12 16:45 echopoint_v5.38_getting_serial_working_initial_pause
Nov 12 16:51 echopoint_v5.39_getting_serial_working_initial_pause
Nov 12 16:54 echopoint_v5.40_getting_serial_working_initial_pause
Nov 12 16:58 echopoint_v5.41_getting_serial_working_initial_pause_last_worke
```

11/13/2023 Documenting final wiring diagrams, taking final measurements. Image of the device with two batteries.



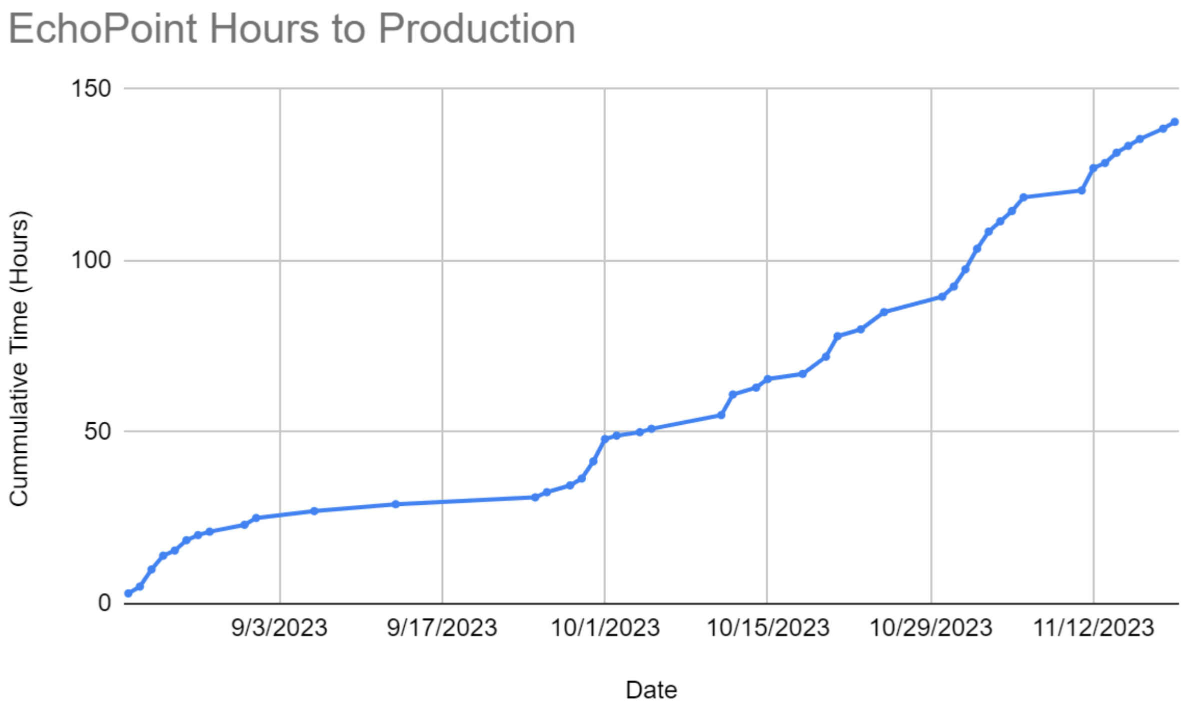




Entry	Date	Time (hours)	Commute (hours)	Cumulative Time (Hours)	Notes
71	8/21/2023	3		3	EchoPoint: developing prototype unit 1
72	8/22/2023	2		5	EchoPoint: developing prototype unit 1
73	8/23/2023	5		10	EchoPoint: developing prototype unit 1
74	8/24/2023	4		14	EchoPoint: developing prototype unit 2
75	8/25/2023	1.5		15.5	EchoPoint: developing prototype unit 2
76	8/26/2023	3		18.5	EchoPoint: developing prototype unit 2 and generating audio files
77	8/27/2023	1.5		20	EchoPoint and planning Tucker Sound Effects Module
78	8/28/2023	1		21	EchoPoint: developing prototype unit 2
79	8/31/2023	2		23	EchoPoint: developing prototype unit 2
80	9/1/2023	2		25	EchoPoint: Attempting to make a more power-efficient unit

82	9/6/2023	2		27	EchoPoint: Attempting to make a more power-efficient unit
86	9/13/2023	2		29	EchoPoint: designing a more efficient module -- using relay
90	9/25/2023	2		31	EchoPoint: researching different amplifiers used to make the sound louder and how to build the housing
91	9/26/2023	1.5		32.5	EchoPoint: building housing
92	9/28/2023	2		34.5	EchoPoint: testing different amplifiers and audio configurations
93	9/29/2023	2		36.5	EchoPoint: testing different amplifiers and audio configurations
94	9/30/2023	5		41.5	EchoPoint: determining how to automatically recognize the number of audio files (failed) and getting the device to make less noise when starting up (success). Begin constructing housing.
95	10/1/2023	6.5		48	EchoPoint: building tool to test playback time, improving housing
96	10/2/2023	1		49	EchoPoint: testing playback time
97	10/4/2023	1		50	EchoPoint: testing playback time
98	10/5/2023	1		51	EchoPoint: testing playback time
100	10/11/2023	4		55	EchoPoint: making manual
101	10/12/2023	6		61	EchoPoint: Making manual
102	10/14/2023	2		63	EchoPoint: planning and ordering parts for longer battery life and "one button one sound"
103	10/15/2023	2.5		65.5	EchoPoint: planning and ordering parts for longer battery life and "one button one sound"
104	10/18/2023	1.5		67	EchoPoint: testing new battery, researching voltage regulators
105	10/20/2023	5		72	EchoPoint: attempting to replace relays with transistors (failed)
106	10/21/2023	6		78	EchoPoint: attempting to replace relays with transistors and getting "one button one sound" wired (failed)
107	10/23/2023	2		80	EchoPoint: planning and researching alternative circuits to replace relay with transistors
108	10/25/2023	5		85	EchoPoint: using Atmega328p processor (failed)
109	10/30/2023	4.5		89.5	EchoPoint: using Atmega328p processor (failed)
110	10/31/2023	3		92.5	EchoPoint: using Arduino Uno processor (failed)
111	11/1/2023	5		97.5	EchoPoint: learning how to use NodeMCU
112	11/2/2023	6		103.5	EchoPoint: learning how to use NodeMCU (success)
113	11/3/2023	5		108.5	EchoPoint: programming NodeMCU and constructing housing for one button one sound

114	11/4/2023	3		111.5	EchoPoint: redesigning circuitry and using diodes to solve problems
115	11/5/2023	3		114.5	EchoPoint: troubleshooting code, not all GPIO pins work as expected
116	11/6/2023	4		118.5	EchoPoint: troubleshooting code and rewiring different amplifiers
117	11/11/2023	2		120.5	EchoPoint: troubleshooting code and measuring current usage of device
118	11/12/2023	6.5		127	EchoPoint: troubleshooting code (success) and building battery housing, and securing internal components with adhesives.
119	11/13/2023	1.5		128.5	EchoPoint: testing and documenting circuitry
120	11/14/2023	3		131.5	EchoPoint: updating build guide
121	11/15/2023	2		133.5	EchoPoint: making tutorial videos for changing, content, charging, etc.
122	11/16/2023	2		135.5	EchoPoint: updating build guide
123	11/18/2023	3	6	138.5	EchoPoint: preparing to mount device, interact with guests, pulled data from VinTEL phone to generate report, fixed coin jam on VinTEL, installed new sign for VinTEL to increase usage.
124	11/19/2023	2		140.5	EchoPoint: preparing build guide and learning to make circuit diagrams digitally



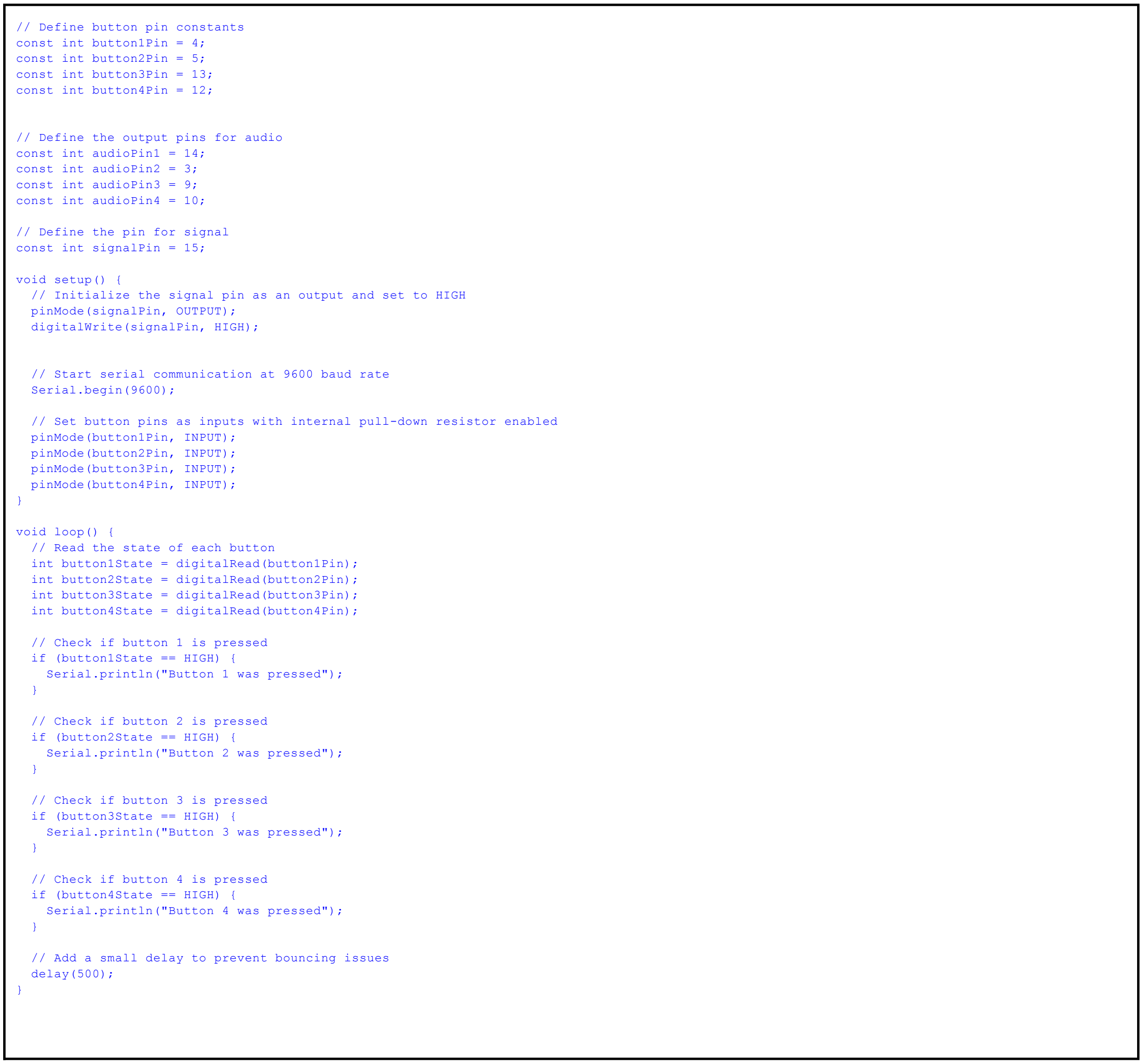
BUILD COSTS

Total: $79.84

- $21.49 Metal case, modified from Sentry Safe Lock Box
  https://www.amazon.com/gp/product/B004LAMJVI/ref=ppx_yo_dt_b_search_asin_title?ie=UTF8&th=1
- $5.00 Metal speaker grill (1x at $5.00 each, $9.99 for 2)
  https://www.amazon.com/gp/product/B0B1LFCDSH/ref=ppx_yo_dt_b_search_asin_title?ie=UTF8&th=1
- $2.75 Speaker (1x at $2.75 each, $10.99 for 4)
  https://www.amazon.com/gp/product/B0B4D2Z35P/ref=ppx_yo_dt_b_search_asin_title?ie=UTF8&th=1
- $2.70 electrical wires (27x at $0.10 each, $11.99 for 120)
  https://www.amazon.com/dp/B07GD1TH2K/ref=redir_mobile_desktop?_encoding=UTF8&ref_=ya_aw_od_pi&th=1
- $7.38 Breadboard
  https://www.amazon.com/dp/B00B8861R4?psc=1&ref=ppx_yo2ov_dt_b_product_details
- $0.10 10K Ohm resistor (10x at $0.01 each, $13.99 for 1000)
  https://www.amazon.com/gp/product/B08FHPKF9V/ref=ppx_yo_dt_b_search_asin_title?ie=UTF8&psc=1
- $0.24 Diodes (4x at $0.06 each, $5.89 for 100)
  https://www.amazon.com/gp/product/B00N1ZKU7E/ref=ppx_yo_dt_b_search_asin_title?ie=UTF8&psc=1
- $1.80 MOSFET Transistor 30NO6L (2x at $0.90 each, $8.99 for 10)
  https://www.amazon.com/gp/product/B07WHSD3GJ/ref=ppx_yo_dt_b_search_asin_title?ie=UTF8&psc=1
- $0.38 MOSFET Transistor IRF9540N (1x at $0.38 each, $18.99 for 50)
  https://www.amazon.com/gp/product/B082J3F8HJ/ref=ppx_yo_dt_b_search_asin_title?ie=UTF8&psc=1
- $7.00 Playback Module DV-SV8F
  https://www.amazon.com/dp/B0BL9SDJPW?ref=ppx_yo2ov_dt_b_product_details&th=1
- $26.99 6V 6Ah LiFePO4 Battery Pack LF060A1
  https://www.amazon.com/dp/B07P8LCSFJ?psc=1&ref=ppx_yo2ov_dt_b_product_details
- Plastic
- $1.47 ABS Plastic
  https://www.amazon.com/gp/product/B08R9W4NZW/ref=ppx_yo_dt_b_search_asin_title?ie=UTF8&th=1
- $2.53 membrane buttons (1x at $ each, $7.59 for 3)
  https://www.amazon.com/gp/product/B076WS74VN/ref=ppx_yo_dt_b_search_asin_title?ie=UTF8&psc=1

OPERATING SYSTEM CODE

<u>**NodeMCU code**</u>: **echopoint_v5.41.ino** is run on the NodeMCU microchip. It communicates with the power transistor circuitry to cut power to itself and the amplifier after 1 minute of inactivity. It also communicates with the audio amplifier and sound board to select which file to play.



```
// Define button pin constants
const int button1Pin = 4;
const int button2Pin = 5;
const int button3Pin = 13;
const int button4Pin = 12;


// Define the output pins for audio
const int audioPin1 = 14;
const int audioPin2 = 3;
const int audioPin3 = 9;
const int audioPin4 = 10;

// Define the pin for signal
const int signalPin = 15;

void setup() {
  // Initialize the signal pin as an output and set to HIGH
  pinMode(signalPin, OUTPUT);
  digitalWrite(signalPin, HIGH);


  // Start serial communication at 9600 baud rate
  Serial.begin(9600);

  // Set button pins as inputs with internal pull-down resistor enabled
  pinMode(button1Pin, INPUT);
  pinMode(button2Pin, INPUT);
  pinMode(button3Pin, INPUT);
  pinMode(button4Pin, INPUT);
}

void loop() {
  // Read the state of each button
  int button1State = digitalRead(button1Pin);
  int button2State = digitalRead(button2Pin);
  int button3State = digitalRead(button3Pin);
  int button4State = digitalRead(button4Pin);

  // Check if button 1 is pressed
  if (button1State == HIGH) {
    Serial.println("Button 1 was pressed");
  }

  // Check if button 2 is pressed
  if (button2State == HIGH) {
    Serial.println("Button 2 was pressed");
  }

  // Check if button 3 is pressed
  if (button3State == HIGH) {
    Serial.println("Button 3 was pressed");
  }

  // Check if button 4 is pressed
  if (button4State == HIGH) {
    Serial.println("Button 4 was pressed");
  }

  // Add a small delay to prevent bouncing issues
  delay(500);
}
```

TROUBLESHOOTING: GENERAL

Your EchoPoint is not working correctly? Start with this checklist!

1. Open the EchoPoint using your combination.
2. Press the red activation button.
3. There should be sound as well as LED lights that flash when some of the buttons are pressed. Continue if either of these do not function as expected.
4. Adjust the volume knob to determine if the volume is just set to be too low.
5. Remove batteries and replace with fully charged batteries.
6. Test the device to determine if the issue is resolved.
7. Check that all wired connections are secure.
8. Again, check if this fixes any issues.
9. If issues persist then, continue to troubleshooting specifics.

TROUBLESHOOTING: SPECIFIC

No sounds and internal lights do not turn on

- Problem: There is no power to the device or any of its components.
- Solution:
    - Replace batteries with fully charged batteries -- the voltage may be high enough to power the indicator lights but not high enough to drive the amplifier.
    - Check that all connections are secure.

LED lights turn on inside the device, but no sound is produced.

- Problem: The device speakers are not making any sounds, but there is power.
- Solution:
    - Replace batteries with fully charged batteries -- the voltage may be high enough to power the indicator lights but not high enough to drive the amplifier.
    - Check that all connections are secure.
    - Test an alternative speaker to determine if the speaker is broken.

Audio too quiet:

- Problem: The audio is too quiet.
- Solution:
    - Change the audio amplifier dial located on the amplifier board near the batteries.
    - The selected audio file might be digitally too quiet if it is a new recording being added, use software like Audacity to increase the digital volume of the file.

DISCLAIMER

The EchoPoint device described in this manual is provided "as is" with limited warranty at the discretion of the creator (Andrew D. Marques). This discretionary limited warranty includes but is not limited to the implied warranties of merchantability, fitness for a particular purpose, or non-infringement. In addition, the user may return the device at any time. Replacement of the device can be discussed at the discretion of the creator.

ABOUT THE CREATOR

Andrew D. Marques is a virologist at the University of Pennsylvania's Perelman School of Medicine. Some of his favorite pastimes include spending time with his wife and family, building projects, programming, film photography, amateur birding, cycling, maintaining and driving cars, and restoring antique electronics by finding ways to incorporate them into modern life. At a young age, he has been interested in taking a holistic perspective of history: where historical and current events are closely connected, and having a physical and mental connection to our past can better shape our present and future.

ACKNOWLEDGEMENTS

The success of this project was made possible by the assistance of numerous friends, whose contributions I would like to acknowledge. Initial conception of a device like EchoPoint is accredited to Rob Kain, "Tiger" Tom Ehrhart, and others at the AACA Museum. I also would like to thank Sean Tucker and Mark Lieberman for their correspondence shaping this project. I extend a special thank you to Carter Merenstein, Corey Nickels, and Prakrati Nickels, for their invaluable troubleshooting help with conceptualization and programming.

Additionally, I owe a debt of gratitude to my wife Caitlyn Mierau-Marques, whose unwavering support was critical to the project's success. Caitlyn was endlessly patient and supportive, providing encouragement throughout the countless nights and many hours spent grinding away using the dremel or on the computer programming.

Their insights, feedback, and support were vital to the completion of this work.