Harrison Ernst

Student-Athlete, Lehigh University

Majors: Electrical Engineering, German

Email: harry@ernst.site

Website: www.ernst.site

Hello

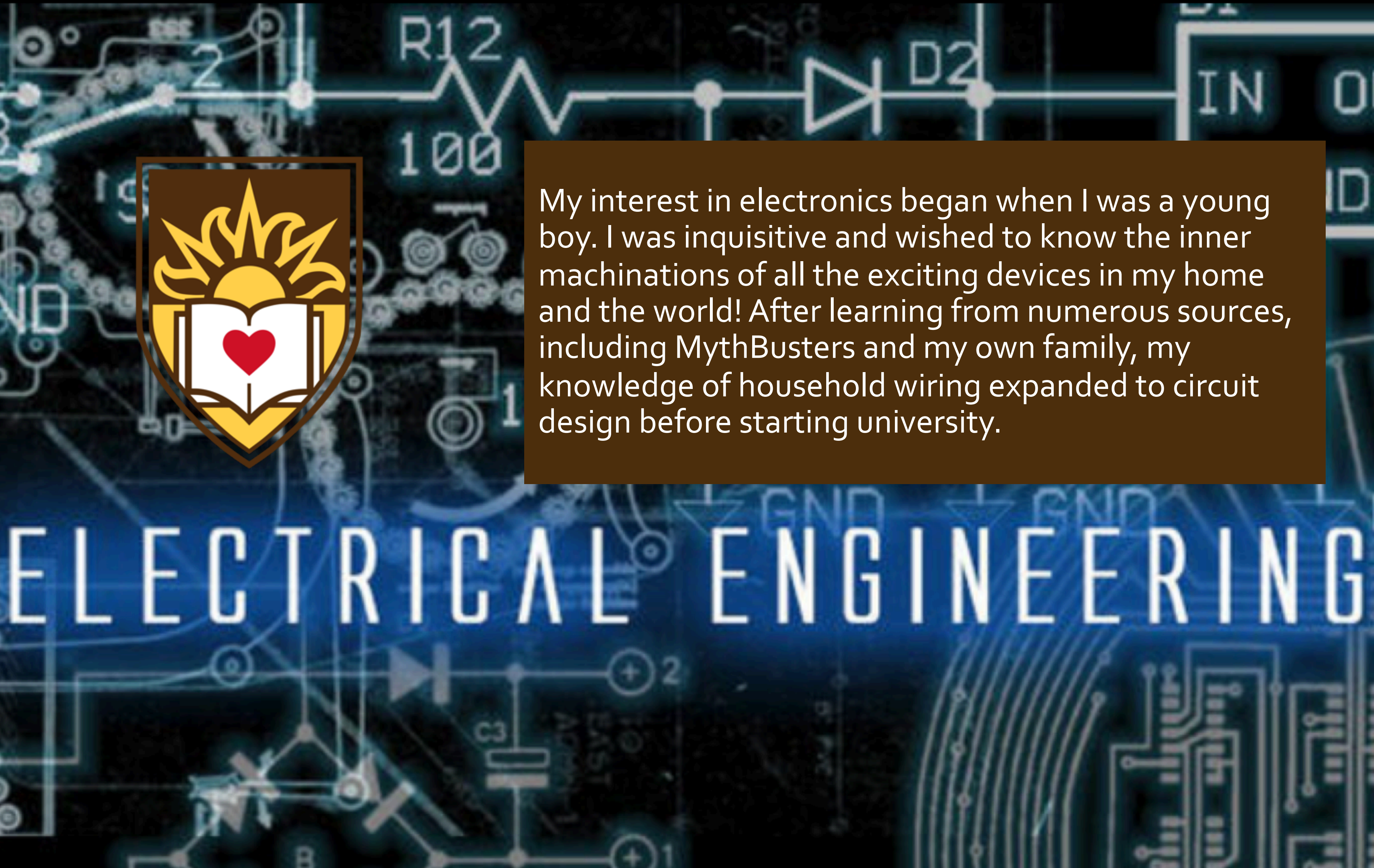
My name is Harry Ernst, and I am a third-year student-athlete at Lehigh University with a plethora of hands-on experience. Being a varsity track and field member, declaring two majors, and studying abroad in Germany has provided me with the sense of intuition and collaborative skills required to solve complex engineering problems with ease. These opportunities have also prepared me to function in the modern workplace independently and as a member of a team.

Table of Contents

Wood & Metalworking

Circuit Components

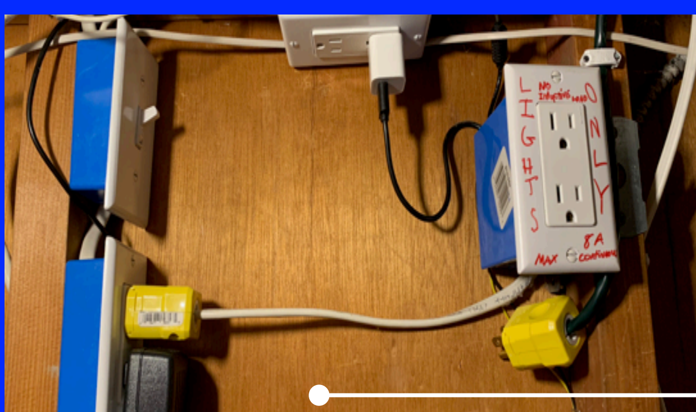
My interest in electronics began when I was a young boy. I was inquisitive and wished to know the inner machinations of all the exciting devices in my home and the world! After learning from numerous sources, including MythBusters and my own family, my knowledge of household wiring expanded to circuit design before starting university.



Building and Repairs

Over the years, I learned a plethora of information regarding home improvement. Starting with roofing repairs and household wiring, I was eager to learn more. I started winterizing sprinkler systems, configuring new WIFI networks and security systems, and expanded to computer repair and upgrades. When I learned how to solder, I immediately knew that electrical engineering was right for me.

MacBook Pro Fan Replacement



Receptacle Installation

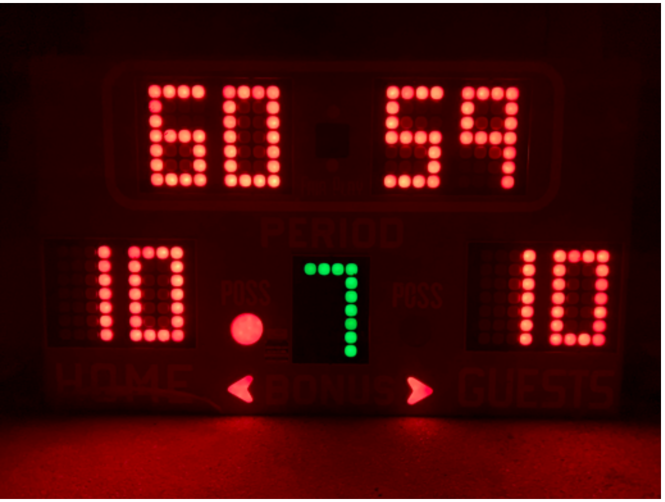
PCB Tree

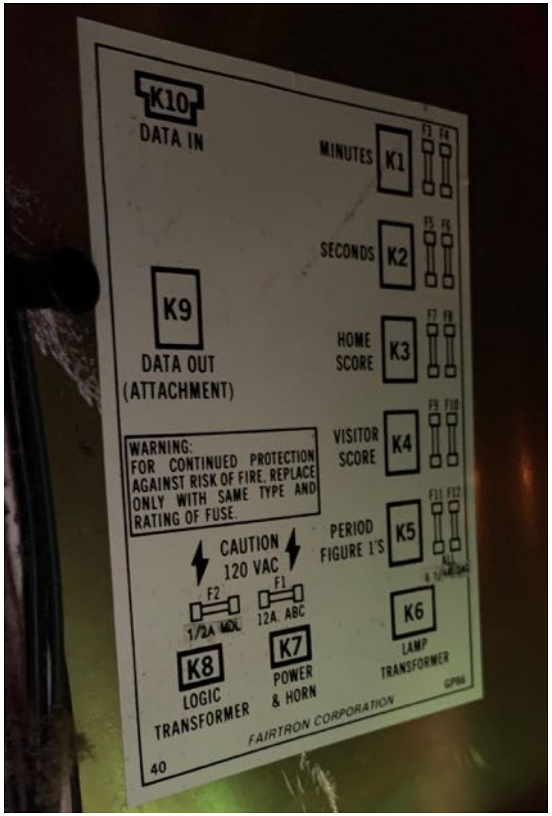
This was my first ever soldering project. I bought a kit for this Christmas tree which contained all of the necessary components including LEDs, transistors, resistors, capacitors and boards. Ever since I completed the project in 2014 the tree has been powered on!

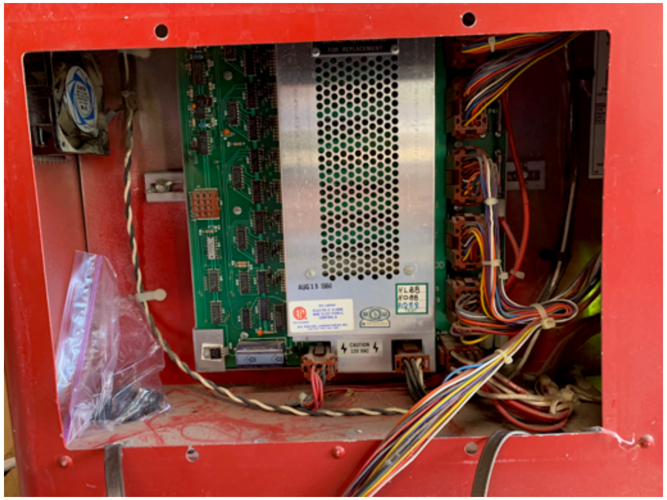
The implementation of different size capacitors in conjunction with transistors allowed all red, then yellow, then green LEDs to illuminate sequentially, with one exception, being the top red LED, which is on continuously.

Scoreboard Repair

My middle school was throwing out the old, broken scoreboard, so I figured I would attempt to save it. Luckily for me, everything was clearly labeled inside the metal casing, and only a few wires were destroyed, so the repair was simple. I have yet to use the finished product.







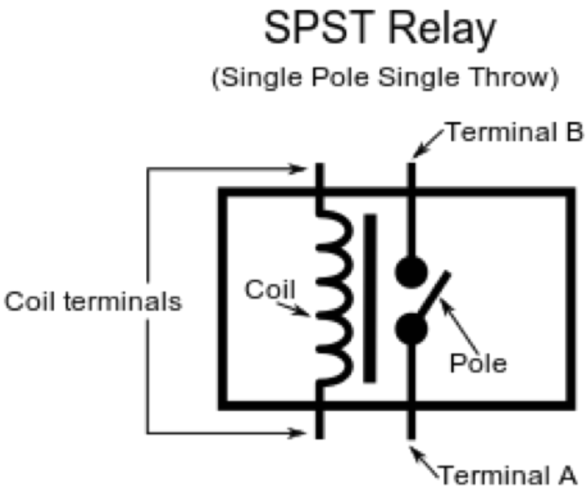
Synchronized Lights to Music Display

Here, you can see my current 3 house display using Light-O-Rama hardware

The First Show

I started making Christmas light shows in 2014 for my friends and neighbors to enjoy during the holiday season. I began with an Arduino, an array of relays, and the software Vixen Lights to make the strands of lights and props come to life. Each prop or strand of lights had its own relay or channel so that the show could be as interactive as possible.

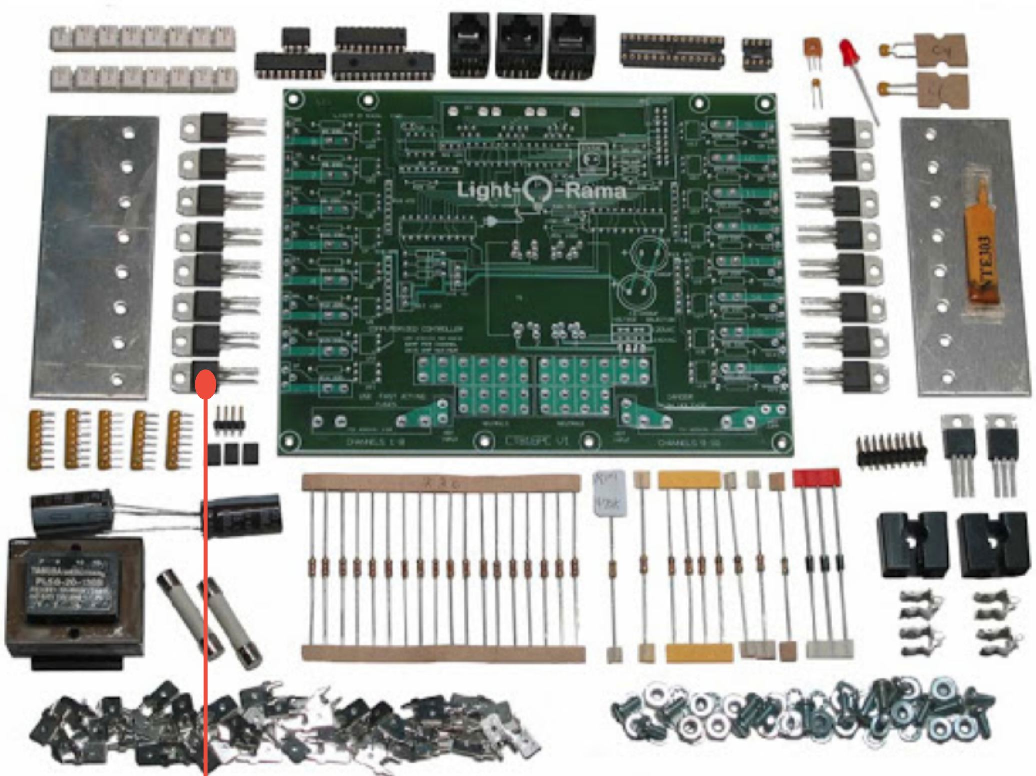
```
// Includes the watchdog timer library
#include <avr/wdt.h>

// This sets how many channels will vixen be sending. Can be set to any number from 1 to 48 for Arduino Mega, and 1 to 18 for Arduino Uno.
#define CHANNEL_COUNT 16

// speed for the com port for talking with vixen. From 9600 to 115200. Use the same speed as set in Vixen.
#define VIXEN_COM_SPEED 57600

// Timeout waiting for serial input before going to random mode (in milliseconds).
#define TIME_OUT 1000

// If the relays turn On and Off opposite to Vixen sequence, change "#define MODE NOT_INVERTED" for "#define MODE INVERTED"
#define NOT_INVERTED 0
#define INVERTED 1
#define MODE NOT_INVERTED

// which pins control which channels
// You can change these assignment to use different pins, but be very careful to not repeat the same pin number for 2 channels.
// DO NOT use pings 0 and 1, as those are for the serial port to talk to the computer.
#define CH01 2
#define CH02 3
#define CH03 4
#define CH04 5
#define CH05 6
#define CH06 7
#define CH07 8
#define CH08 9
#define CH09 10
#define CH10 11
#define CH11 12
#define CH12 13
```

This approach worked very well for the first year since the display only consisted of 16 channels, however as the display grew and even included my neighbors' homes, the amount of time necessary to construct the waterproof boxes containing the relays became extreme. For this reason, as well as the fact that cheaper SPST relays are not very reliable in the harsh NY winters, I switched to Light-O-Rama hardware.

Expansion

The first Light-O-Rama controller I purchased was a kit that required assembly. I went with this option since it was much cheaper than fully assembled controllers, and I enjoyed the soldering! One of the most outstanding qualities of the Light-O-Rama hardware is that it relies on a set of thyristors, or TRIACS (bidirectional triode thyristors), which are solid-state, to control each channel.

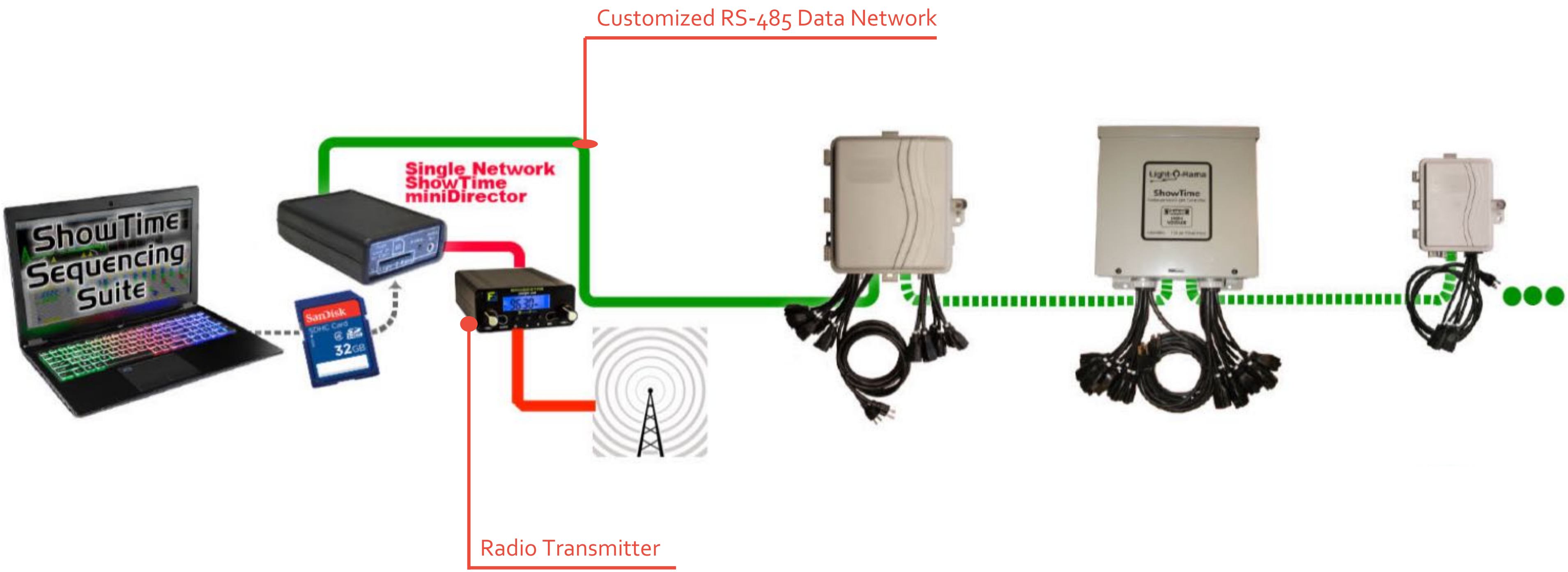


TRIAC

Completed Controller

Current Setup

My show now has ten different controllers, which contribute 148 channels—the props which grab the most attention are my four 12 foot homemade PVC spiral trees and custom made stars. Once the controllers are given power via photocell timers, they contribute voltage to the data network and allow the miniDirector to run the files on the SD card. Thankfully I broadcast the music on the radio, so it is not playing out loud all night long! For the 2020 show, I programmed 24 songs, which took around 12 hours each.

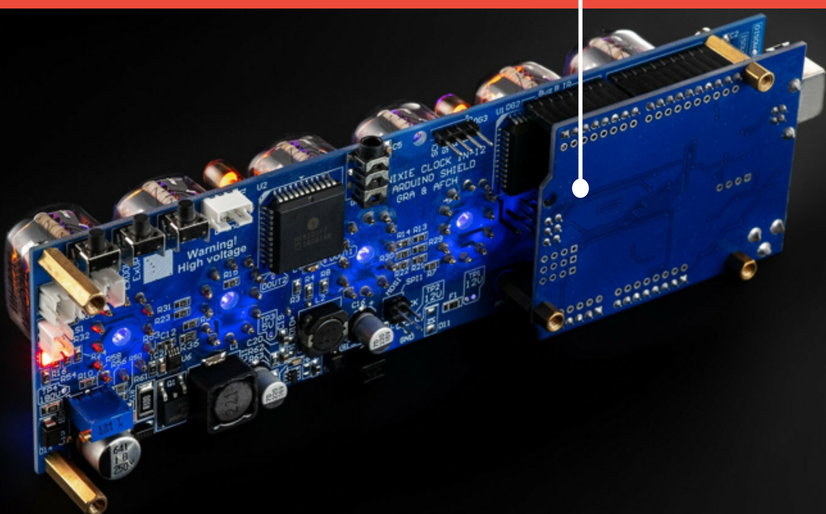


Please visit www.ernst.lighting to see the show!

Nixie Clock

Over the summer quarantine, I came across Nixie Tubes online and immediately fell in love. I did some research and decided on making a clock. Using Arduino and a shield kit, I soldered everything together and, when powered on, was greeted with the amazingly nostalgic glow of the tubes. This was a quick project, but definitely one of my favorite.

Arduino



Single Nixie Tube

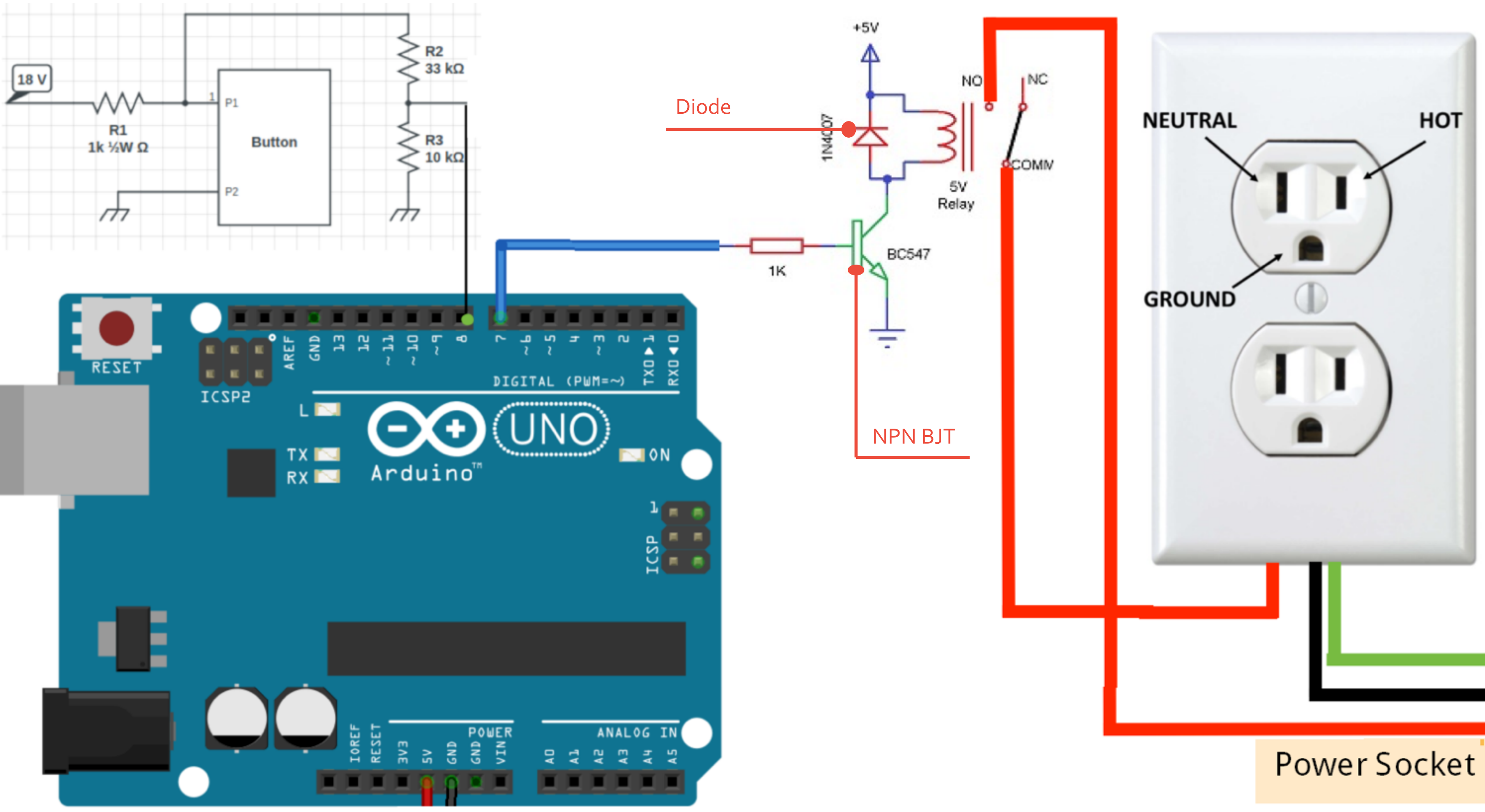
Finished Clock

Pedestrian Push Button Light Switch

As a child, I was immensely intrigued by traffic lights and intersection devices. So when two pedestrian pushbuttons became locally available, I purchased them immediately! I knew these would be excellent light switches; thus, I started designing a circuit. The thing that made this process a bit tricky was that the buttons I bought were recommended to use 18V DC (for the internal pressure sensor, LED, and buzzer), and I wanted them to trigger 120V AC lights.

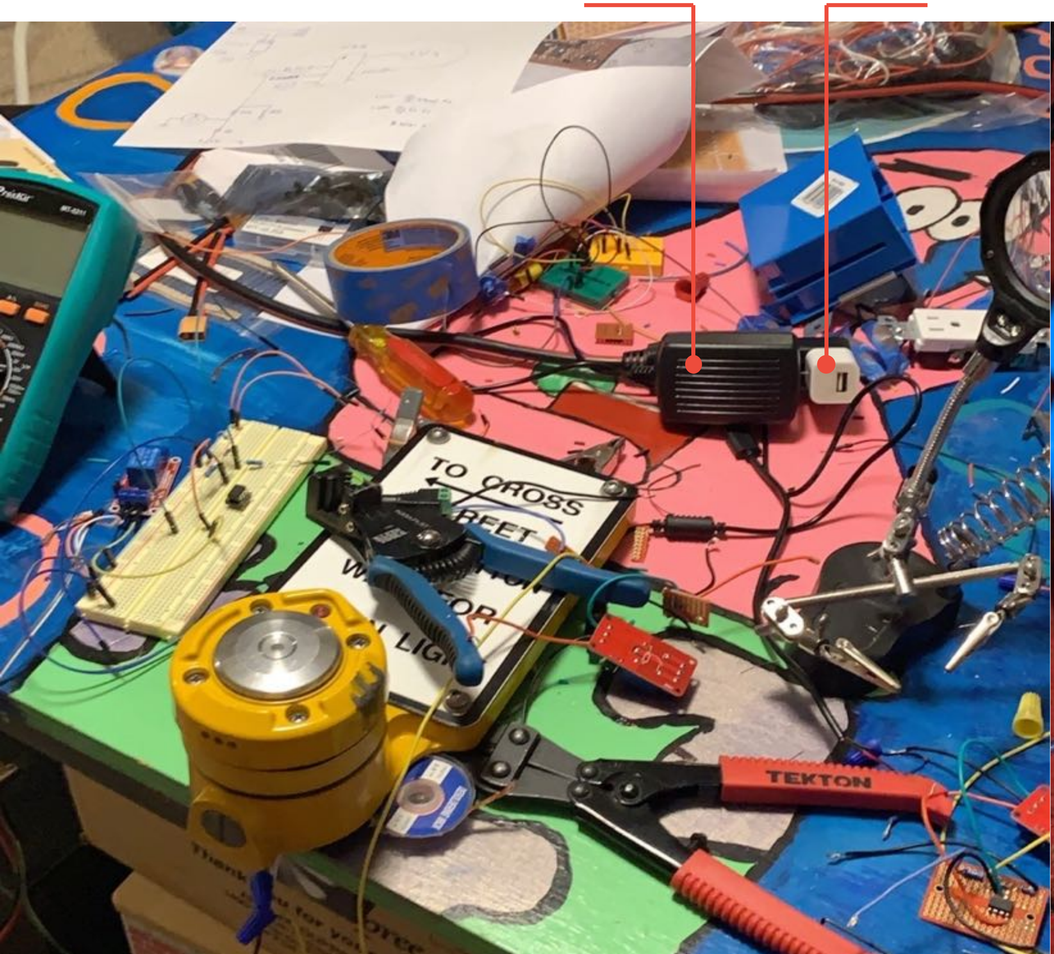
Prototype

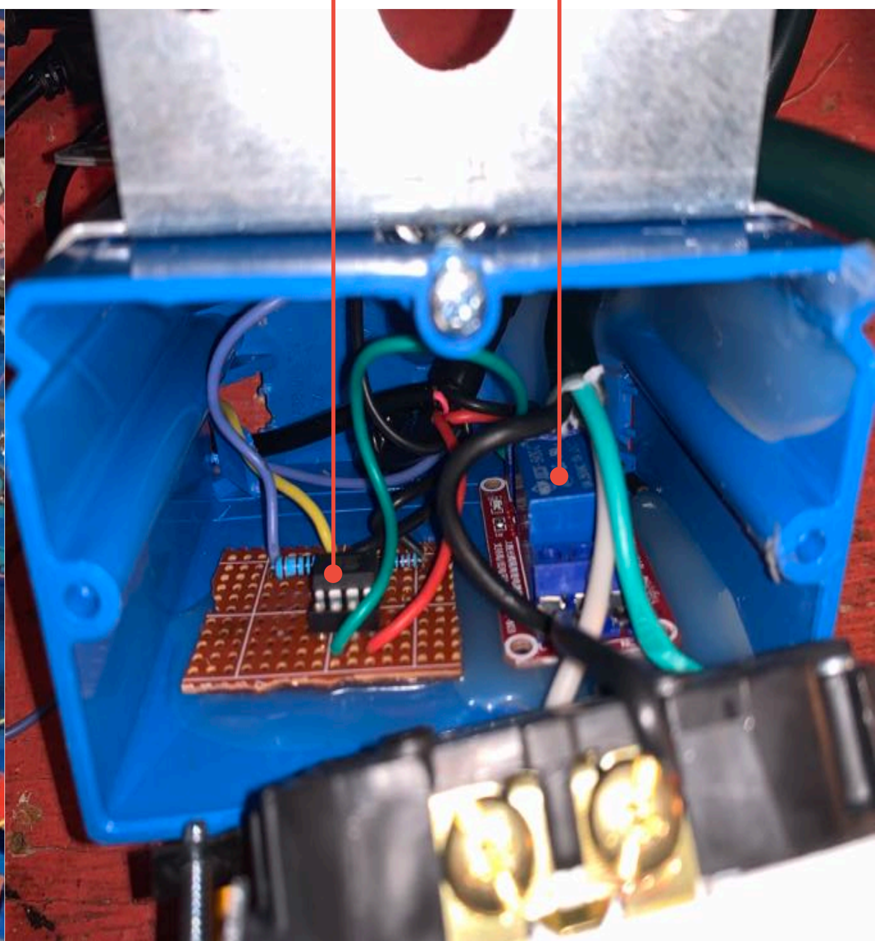
Since the button is 18V DC, but the Arduino input requires around 5V DC, I implemented a voltage divider to drop the voltage. I took advantage of the Arduino's internal pull-up resistors to compensate for noise. On the output side, I used an NPN Bipolar junction transistor to boost the signal for a 5V DC relay with a diode to protect its coil. The relay is used to control the outlet, which will be the source for the lights.



Testing

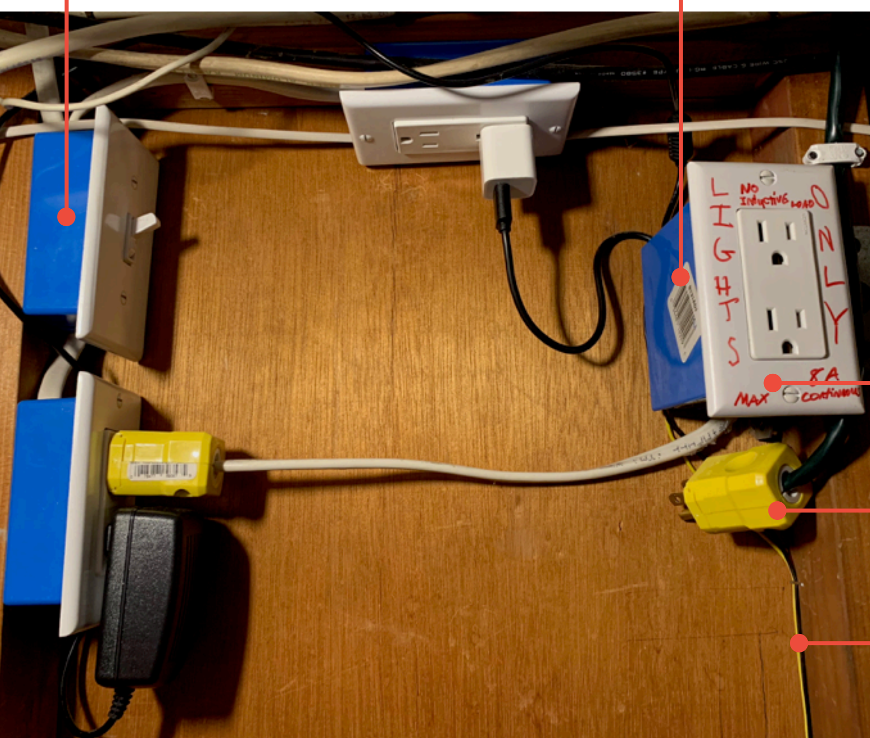
An Arduino Uno would be much too large in a final installation, so I opted to use an AT Tiny 85 instead. I also decided to use a premanufactured relay board, which included a diode and transistor in a small package. I used a wall wart to supply the 18V DC and an old phone charger for 5V DC. I mounted the circuit in a deep-set junction box and screwed the outlet into place. After I was sure the one I would permanently install on the wall worked, I made a second unit that could be retrofitted anywhere.





Final Designs

For the button that I installed on my basement wall, new electrical infrastructure was necessary. The control unit needed three individual outlets for the 120V AC, 18V DC, and 5V DC inputs; I installed two new receptacles with a kill switch behind the wall and tapped into a circuit with a small existing load in respect to the 20A breaker. For my second button, I made a portable unit that can be used practically anywhere!

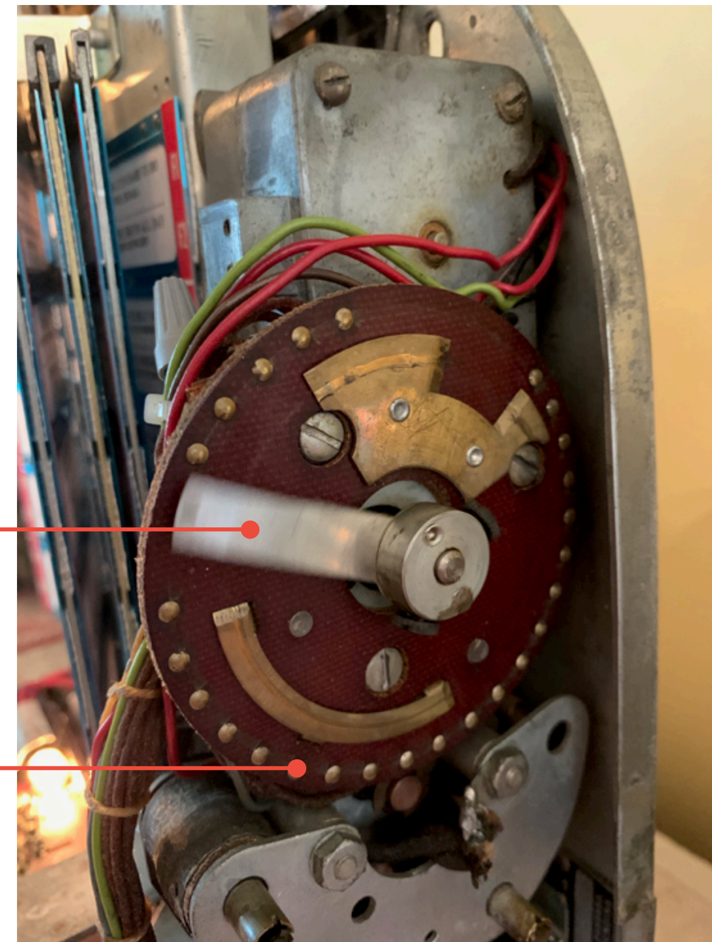


Seeburg Wallbox Restoration

& Jukebox Emulation

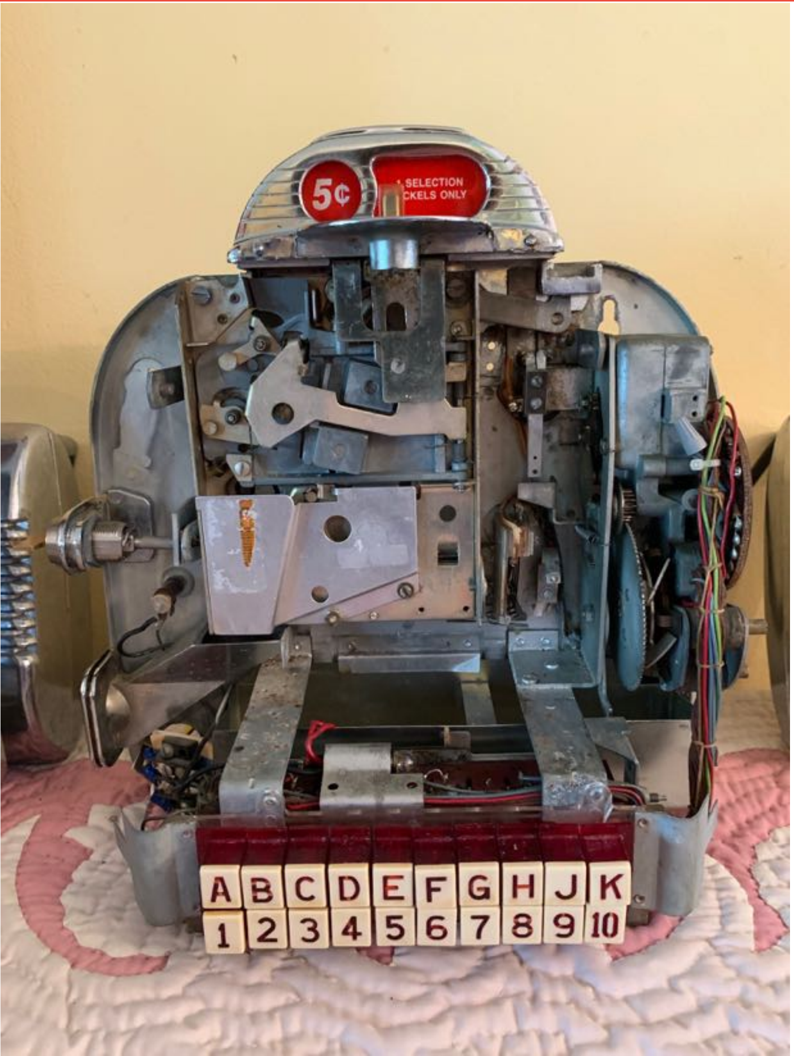
Diagnosis

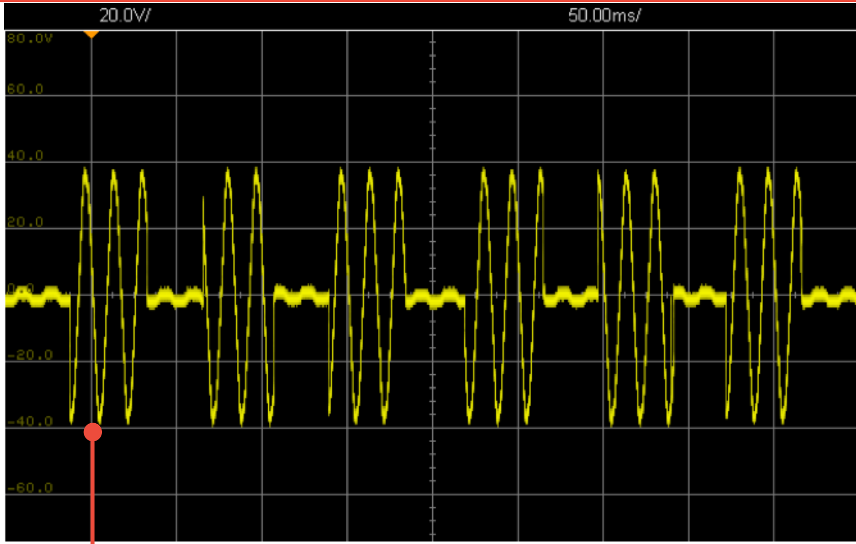
This electromechanical contraption caught my eye at a garage sale and is now one of my favorite possessions. This machine is a Wallbox and works in conjunction with a compatible jukebox. Essentially it is a fancy remote; it has no internal speaker, and no music is stored inside. When a letter and number are selected, different pulses are sent to the jukebox, which must decipher the 25V AC signal. The motor technically creates the pulses since it drives the contact arm. After finding a 25V AC transformer, the lights lit up, but the stepper motor did not turn, and many wires were destroyed. With the instruction manual's help and numerous wiring diagrams, I was able to rewire the machine and unseized the motor. After cleaning the contacts, I needed to see if the proper signals could still be generated.



Testing

This time I used an oscilloscope and compared a few of the alphanumerical combinations with a baseline I found online. Everything checked out, so now it was time for a jukebox emulator. I was going to make one myself, but I knew I would have run into numerous complications. I knew I'd need at least a rectifier (full bridge), voltage regulator, an optoisolator, and Schmitt trigger in addition to capacitors and resistors. However, that only converts the 25V AC pulse to a 5 V DC signal, which doesn't consider an mp3 player, music storage, or a program to play the song corresponding to the alphanumeric selection.





Raw AC Pulse

Completion

I came across a standalone jukebox emulator compatible with my Wallbox, so the choice was obvious. If I had designed my own, I'd use a Raspberry Pi; however, they can be problematic since they have operating systems. After inspecting the Wallbox2mp3 jukebox emulator, it was clear that it used boards similar to Arduino minis, and all music was stored locally on an SD card, which I preferred. This beautiful machine now sits prominently in my family's home.

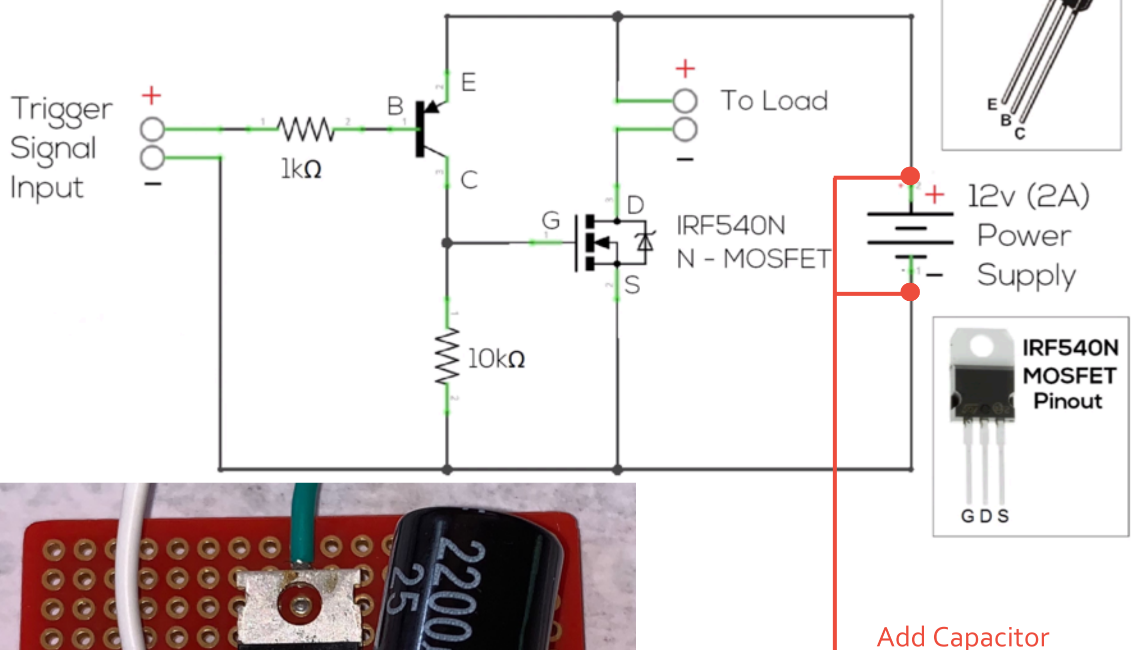
Fire Bell Alarm Clock

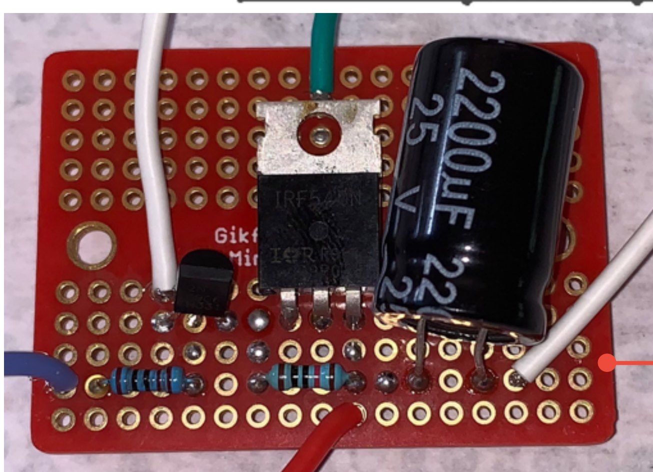
Most alarm clocks are too quiet, so I took it upon myself to solve the problem. I had seen a cheap, mass-produced fire bell alarm clock online, but it was mostly a gimmick since many users noted it broke easily. I knew I could make a much more reliable version, so I got to work.

Design

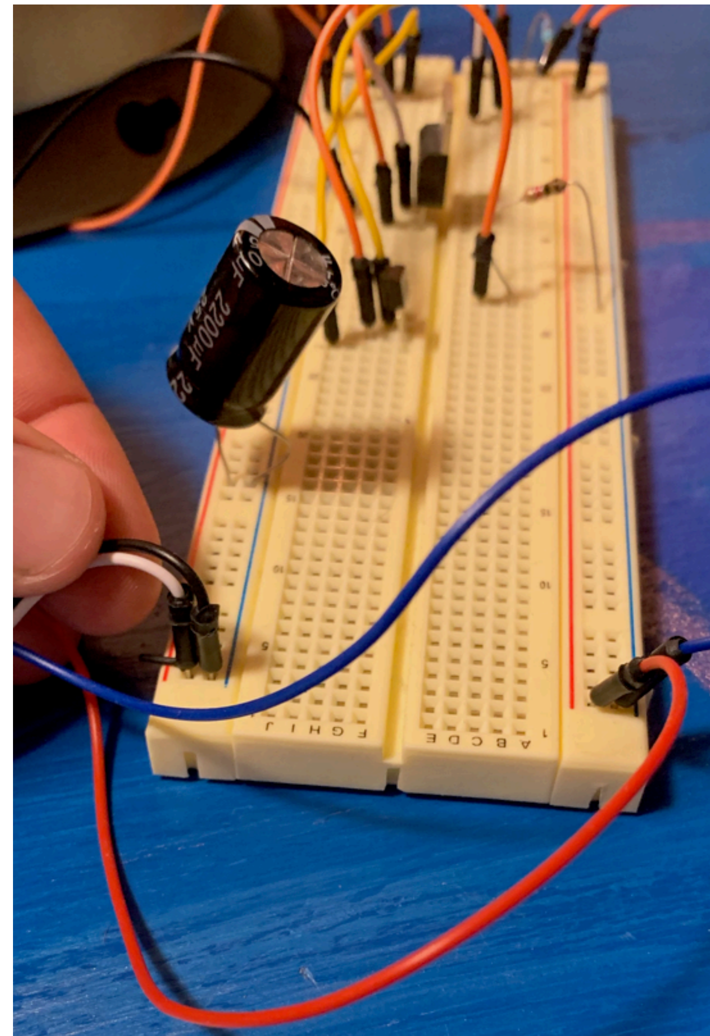
I chose a twin bell style alarm clock because I could easily remove the motor for the bell striker and use that signal as the input for a solid-state relay. I also chose to use a 12V DC bell as I felt it would simplify the circuit compared to the more common 24 V DC fire bells. To make the solid-state relay, I used components readily available to me, including a PNP transistor to boost the clock's signal and an N channel MOSFET to trigger the bell. I ended up adding a capacitor across the terminals of the 12V power supply to ensure the bell would start smoothly.

Solidstate DC Relay



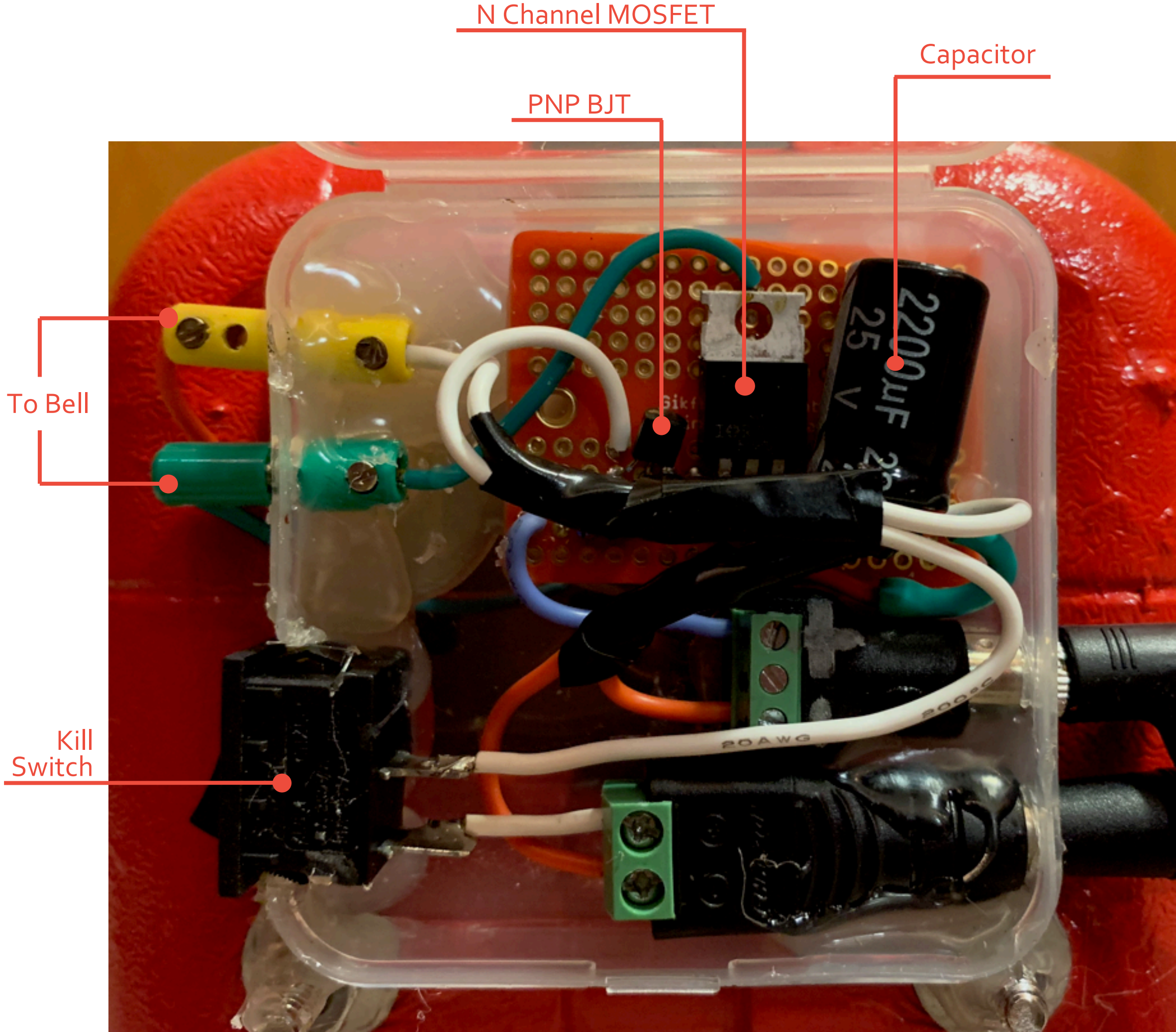


Final Board Design



Finished Clock

I used PVC pipe to make a stand for the bell and connected the clock to the relay circuit with a 3.5mm audio cable, so if the clock breaks, it can be quickly and easily replaced. In fact, all connections to the relay circuit are removable to allow for convenient repair, although I would be astonished if the circuit or bell failed. I use this alarm clock every day and can confidently say it always wakes me up!



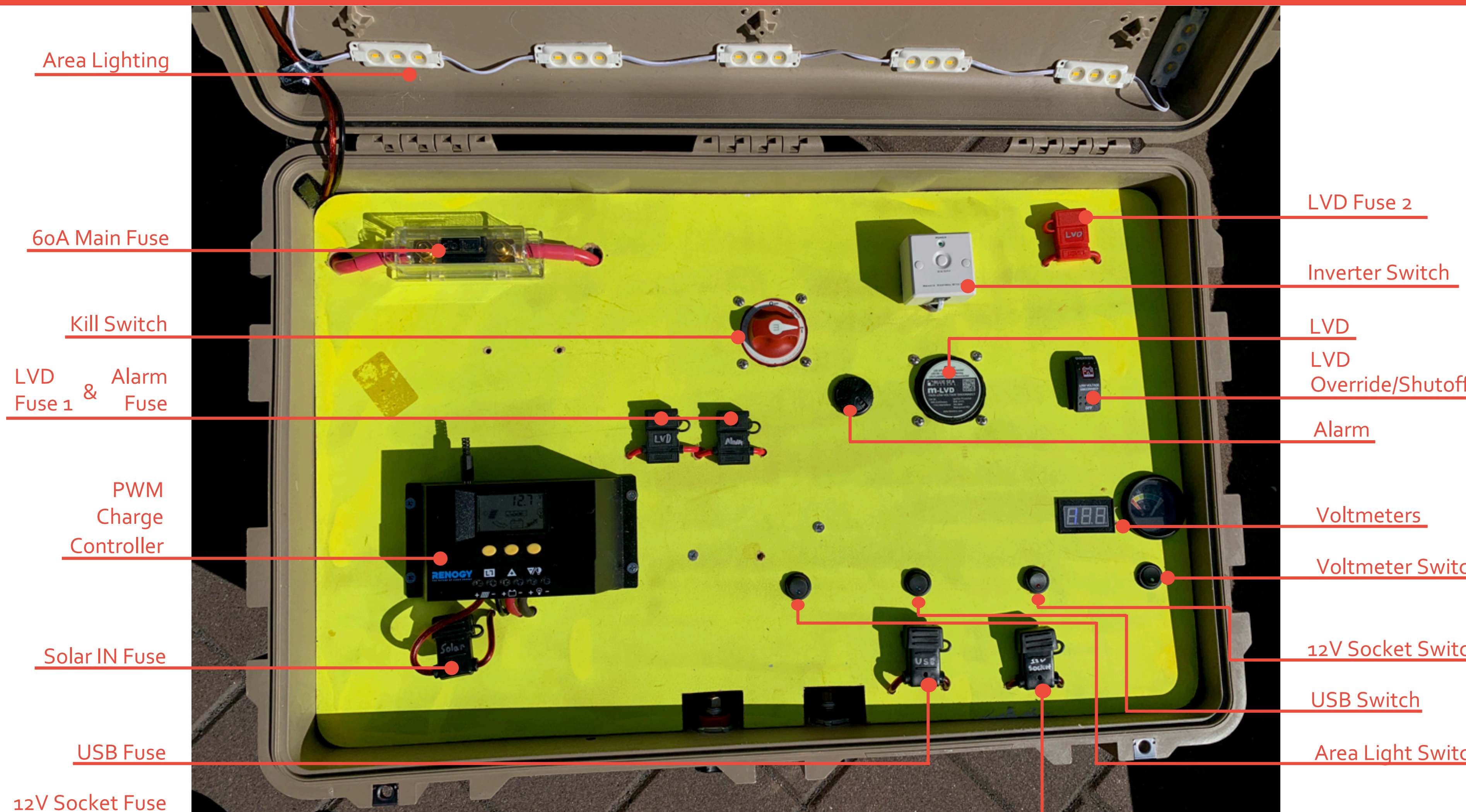

Solar Generator
Battery Backup System

Component Selection

I chose to enclose the system in a waterproof case and used gaskets on all connections passing through the case. The system's capacity is 110 Ah since I used two 12V DC spill-proof absorbed glass mat batteries. I opted for a 12V system since I procured a relatively low-cost 12V pure sine wave inverter. The batteries are protected from over-discharging with a low voltage disconnect (LVD), which happens to be the current bottleneck of the entire system. It only allows a maximum of 60A at 12V DC, so my 1 kW inverter, the lights on the inside of the lid, the 12V automobile socket, and USB ports are all I can implement. I did add a direct connection to the battery that bypasses the LVD to which a larger Inverter or jumper cables could be connected.

More Features

Every component is protected by an appropriate fuse and connected to the main kill switch (except the direct connection to the batteries). The system can be charged either via the sun or the internal battery charger (120V AC). A 30W solar panel is stored inside the case. I implemented a 30A pulse width modulation (PWM) charge controller since it was much cheaper than a maximum power point tracker (MPPT) controller, although I wish to upgrade to a more efficient MPPT controller. I also added an alarm that will sound one minute before the LVD cuts the power; in dire situations, the LVD can be bypassed.



Inputs & Outputs

As stated before, the system can be easily charged via the sun or the internal battery charger (120V AC). The direct connection to the battery could also be used for charging purposes with proper supplies. One 120V two outlet GFCI receptacle is mounted on the case in addition to two USB ports and a 12V DC automotive socket. Although the current capacity only allows an average refrigerator (120V, 200W) to run for a little over six hours (while solar is not connected), my design allows for more batteries to be added effortlessly.


More Projects
Coming Soon!
Remember to visit
www.ernst.lighting