SWiTCHeROO

make the switch

Video demo: https://www.youtube.com/watch?v=9oVj5mh1Zas

Team Switcheroo
Citrus Hack 2018
Best Beginner Hack Winner

Members:
Diego Torres
Morgan Lytle
George Toc
Rick Jiang

Abstract

Switcheroo is a low cost home automation system, that consists of an Android application, and an electrical relay system. This allows users to wirelessly control home appliances from their phone.

Motivation

The motivation of this project is to create a low cost home automation solution, that is robust, easy to use, and can control a wide variety of household devices. By allowing the system to connect to a standard 120 VAC GFCI outlet, anything that can be controlled by a switch on a wall can be wirelessly controlled from a smartphone. This is applicable to lights, fans, heaters, sound systems, and many other things. Furthermore, the components of this system are very affordable and common, which allows for low cost, and easy repairs.
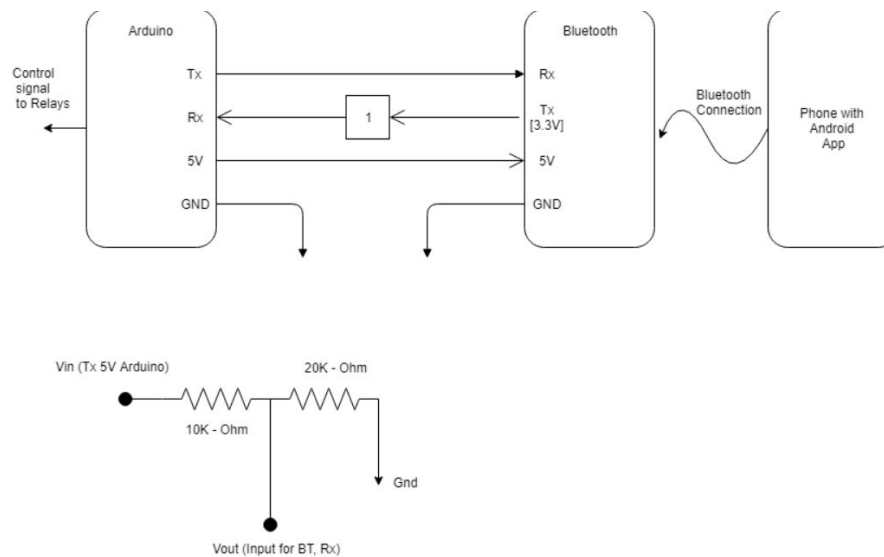
Circuit diagrams and Hardware Implementation

Figure 1: Power subsystem flow diagram
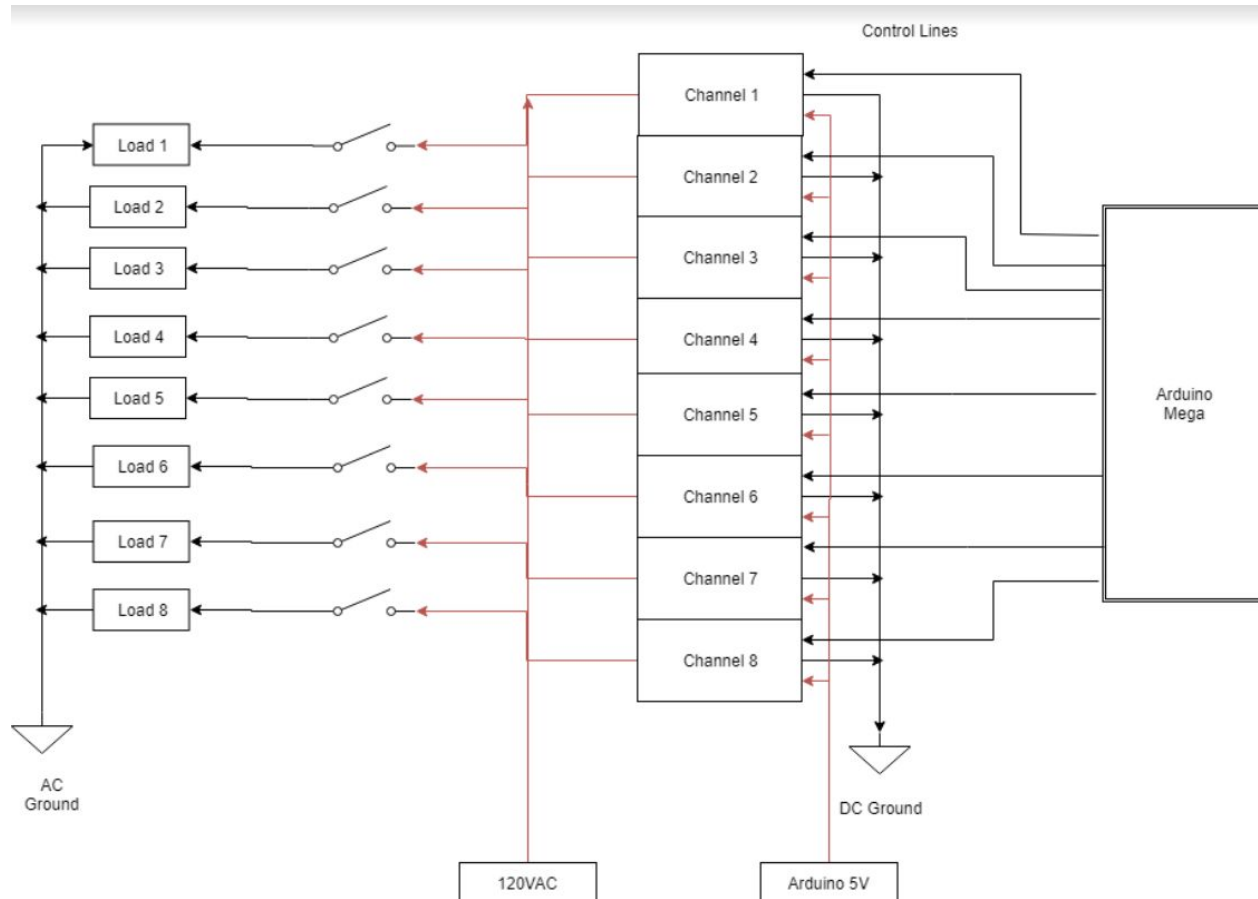
Relays used were the "ELEGOO 8 Channel DC 5V Relay Module with Optocoupler"

Figure 2: Bluetooth and Arduino subsystem schematic

-The Android application connects via bluetooth to the Arduino Mega microcontroller. By doing this, we are allowed to send signals to the arduino, which gets transmitted to the relay system. These signals take the form of the ASCII character in the range of '1' to '8'. These integers are representative of each of the relays.

-Once the signal is received by the arduino, a control signal is passed to the relay system. For example: if channel 1 is off, and a '1' was received, then the channel 1 relay will be turned on. If another '1' is received, then channel 1 will be turned off. This process is uniform throughout all 8 channels.

-The household devices are connected to the "Normally open" port, so the high voltage device is initialized to the "off" state. When the 5V arduino signal is sent to the relay, an internal electromagnet is energized, which causes the switch to close. This causes the load circuit to be completed, and allows the load to receive power. In order to have the desired effect, the device's internal power switch must be closed. Otherwise, the device will have power but will not turn on, which defeats the purpose of this project.

Safety Measures
In order to ensure safe operation of this system, all components in the power circuit were rated to withstand more than 120 VAC. The wire was taken from a household extension cord, and the devices were connected to a GFCI outlet, which has surge protection. Furthermore, all exposed wire connections were covered with insulation before power was connected.

Android App Model and Software Implementation

The Android app was first tested using the Android 7.1 version emulator.
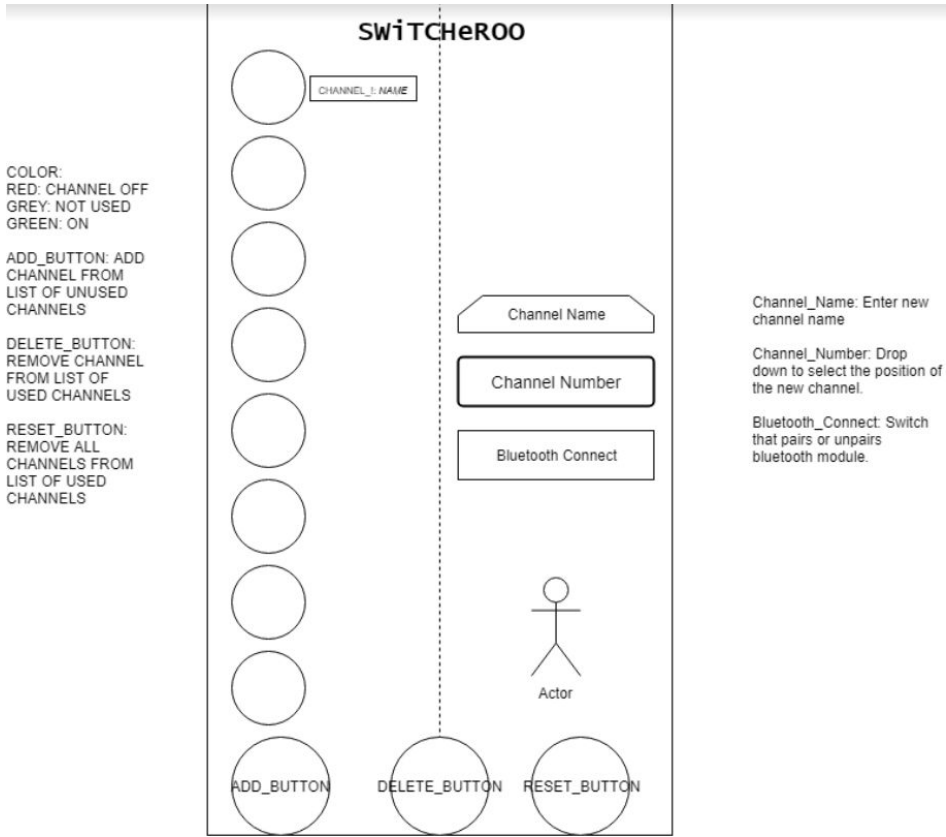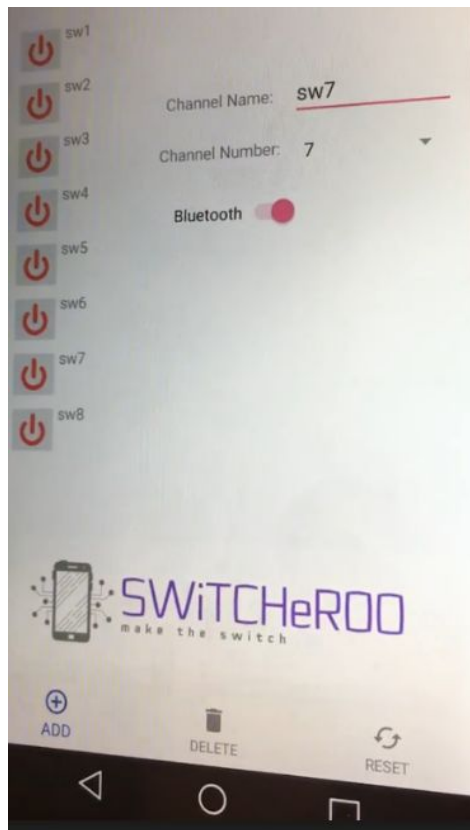
Figure 3: Application Model

Figure 4: Application implementation

GUI Implementation (See figure 4)

-The 8 red power symbols on the left are buttons that represent the current status of the 8 relay channels. These buttons are initialized to being off, which is why they are red. When the user presses these buttons, they turn green and a control signal is sent to the Arduino. The device then turns on.

-While each of the channels are named sw1-sw8, they are easily configured by typing the desired name in the "Channel Name" text field. Furthermore, the "Channel Number" drop down menu allows the user to select which channel their device is connected to. A drop down was used instead of another text field to avoid any ambiguous names that the user may select, which would make the mapping to the channels difficult.

-Switching the bluetooth switch instantiates the bluetooth connection. This utilizes the MAC address of our bluetooth module[*] to avoid interference from other bluetooth devices.

-Once the user enters the information they want they can press the "Add" button, and the channel will be created.

-Pressing the "reset" button will reset all channels to their initialized state.

-Lastly, the "delete button" allows the user to reset a single channel.

Difficulties

On the software level, there were some difficulties when it comes to developing the Android app on Android Studio mainly due to the lack of familiarity of the program. As time progresses, we found ways to implement the ON/OFF buttons for each of the channels along with the text box for the name of the channels. After the graphical user interface was done for the app, there was the issue of constructing a bluetooth socket on Android Studio and establishing communication between the app and the bluetooth. After a few hours browsing references and examples online, we realized that we can preset the address of the bluetooth module to the app so that the app can connect directly to the bluetooth.

Discussion

While we created an easy to use, low cost home automation system at Citrus Hack, there are some ways that we can improve the system, given more time. Future optimizations of the Android application may include more customization, and presets that save once the app is closed. A useful customization would be to make it so the user can type in the MAC address of their device so it isn't hardcoded in the manufacturing process. This would allow for more convenient repairs and replacement of parts.

As far as the hardware improvements are concerned, it would be useful to separate the 8 relays into multiple stations, connected with multiple bluetooth modules. If this was done, then it would be possible to distribute the automated devices across the space where it is being used. Each relay station would be plugged into its own outlet. Since less devices are connected at each station, this expands what is possible for the loads that are automated. This is because the current is distributed across multiple household outlets. In addition to the convenience that this would introduce, it will also add to the safety of the system. It is also critical for the usability of Switcheroo, that a more permanent apparatus is constructed for minimal hardware setup by the consumer. Even though, these hardware changes would entail some changes to the overall product design, they are the next step in the development of Switcheroo.

Conclusion

At Citrus Hack 2018, we designed, and implemented a low-cost and easy-to-use home automation system in 24 hours. While there are some improvements needed to make this a complete product, we have created a solid platform for future development, and gained valuable experience in rapid prototyping, and team collaboration. We also learned the utility of Android Studios, and smart phone bluetooth resources for the integration of hardware and software. By doing this, we have started the development of a home automation solution that is more convenient than what is currently available in the consumer electronics marketplace.