# Automated Sound Panel Modification for Audio Lab

ECE 445 Design Proposal

—

Team 3 - Rishi Kalluri, Zachary Bryl, Rajat Vora
TA: Shaoyu Meng
Spring 2021

**Table of Contents**

# 1. Introduction

## 1.1 Objective

The Illinois Augmented Listening Laboratory is one of a handful of acoustic labs in the country. In order to enable others to contribute more easily to this field, the lab intends to build a fully automated, remotely accessible, audio lab. Researchers could also submit a configuration to run an experiment, and the experiment will be autonomously set up (speakers, mic arrays, sound panels, etc.) and run to capture data. By constructing a space as described, research on hearing aid technology, the "cocktail party problem" [1], and other acoustics research becomes much easier for any research group to study by using this automated, fully remote, lab facility.

Our goal to contribute to this automated lab space is to construct a system to autonomously set up sound panels around the room based on the incoming configuration. We will construct a motorized shutter system that will sit over each individual sound panel. When the shutter is open, sound is allowed to contact the panel and ends up trapped in the panel's geometry. In the closed position it will lie flat over the shutter and will mimic a wall for the sound to bounce off, not allowing any sound to reach the panel. In any given set of sound panels (some N x N configuration of panels), each panel's shutter will be controlled by a controller, which determines each panel's state (open/closed) based on the experiment's configuration file. This design modifications to the room acoustics from experiment to experiment, allowing the space to simulate different acoustic environments on the fly.

## 1.2 Background

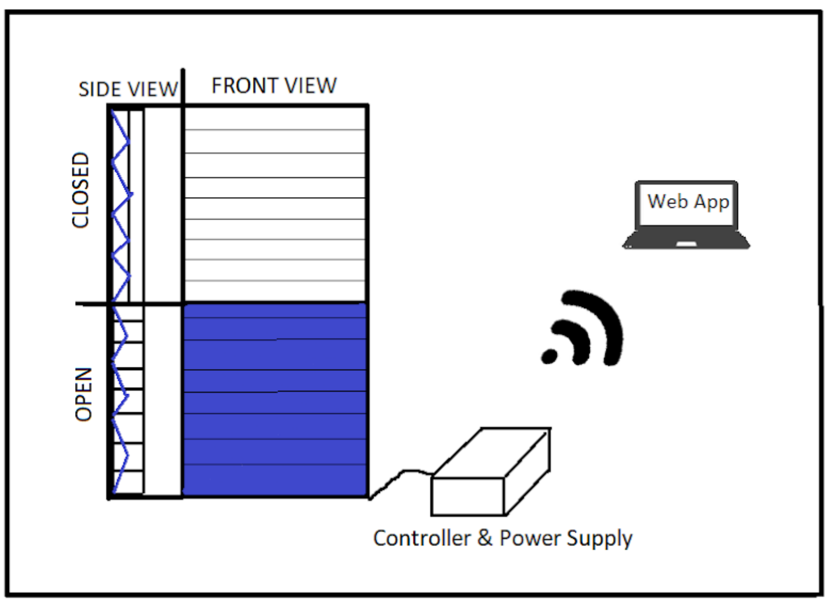
Most hearing aid devices struggle to clearly articulate incoming noise (voice, tv, music, etc.), from surrounding noise, and result in unclear audio assistance. A report published by the *International Journal of Audiology* even lists background noise as the second most contributing reason to the underutilization of hearing aids [2]. A report by the National Institute on Deafness and Other Communication Disorders (NIDCD) states that only 30% of people aged 70 and older with hearing loss have ever worn hearing aids [3]. That number is even lower (roughly 16%) for those aged 20 to 69 [3]. Consequently the NIDCD highlights one of its priorities as improving performance of hearing aids, especially in separating background/ambient noise from relevant input [3].

Solving these problems requires equipment to simulate a variety of environments as well as technology to simulate in-ear acoustics is required. However, equipment of this nature is inherently expensive, making funding and purchasing power a restricting factor in this research. This problem exists in other fields as well, with some having created ways to share their equipment across research spaces. The Robotarium at Georgia Tech is a swarm robotics research space that allows remote access by researchers who want to test new algorithms or experiments, without needing to buy robots or travel to the lab themselves [4]. This idea can now be applied to the acoustics lab space. Providing other researchers remote-access to an autonomous lab, it greatly increases the opportunity to contribute towards research. This innovation facilitates improvements to hearing aid technology and quicker solving of other acoustics problems.

## 1.3 Physical Design

As shown by figure 1, you can see the controller and power supply controlling the state of the blinds. This state is determined based on the configuration that is sent to the controller from the web application.

***Figure 1. High-Level Component Diagram***




## 1.4 High-Level Requirements

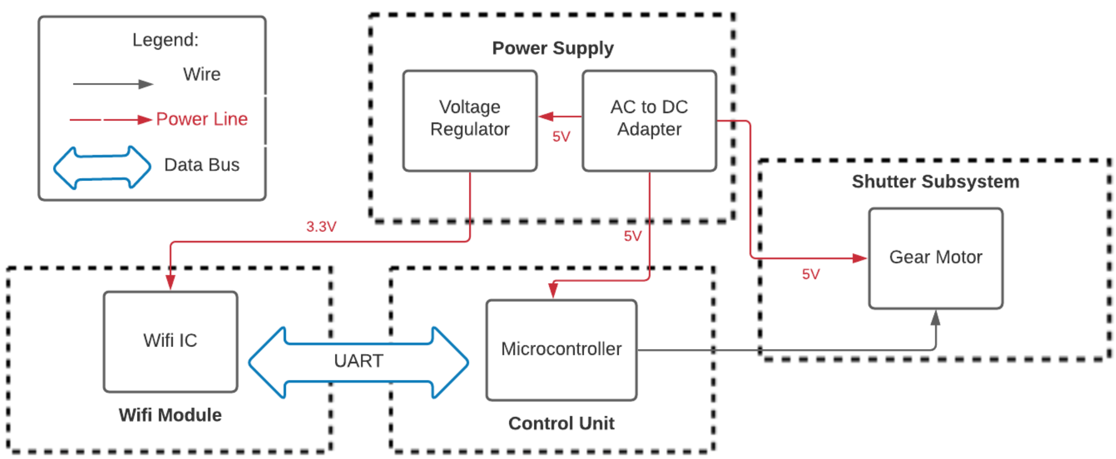
1. The system should significantly change the acoustics of the room, measurable by the T60 Reverberation Time.
2. The time from sending configuration (assuming normal network conditions) to complete execution of instructions on each shutter should be < 60 seconds.
3. The system should be easily scalable up to a 5x5 set of acoustics panels (25 total panels).

# 2. Design

## 2.1 Block Diagram

Our system requires various components for full functionality as shown in Figure 2. These components are the Power Supply, WiFi Module, Control Unit, and Shutter Subsystem. The power supply consists of an AC to DC adapter and a voltage regulator that supplies 3.3 V to various components. We ended up using the adapter so that we would not need to replace batteries. The Control Unit consists of a microcontroller to handle the data from the user and send the corresponding information to the shutter subsystem and the WiFi module. The WiFi module allows for a remote connection to a web application to provide our system with a wireless functionality. The shutter subsystem will consist of a gear motor per panel and allows the shutters to open and close depending on the desired configuration.

***Figure 2. Subsystem Diagram***




## 2.2 Control Unit

The control unit should be able to collect the data from the user via a config file and control the motors. It will also control the data being sent and received from the WiFi module. We will use UART to communicate between the Wifi module and the microcontroller.

#### 2.2.1 Microcontroller

The microcontroller will be an ATMEGA32 Arduino chip on the PCB. The chip will receive the config file data from the user from the WiFi module, communicated via UART. The PIC also has on-chip memory that can be used to store data such as the states of the shutters over each panel and the current experiment metadata. To test the requirements in table 1, we loaded the ATMEGA32 with a configuration file and had the ATMEGA32 perform the necessary program. This also worked when we loaded the configuration file via WiFi.

***Table 1. Microcontroller R&V***

| Requirement(s) | Verification |
|---|---|
| 1. Able to parse configuration file and activate correct motor | 1A. Load configuration file into program locally (not via WiFi), note which motor should activate based on input file<br><br>1B. Run program, compare activated motor to expected motor to ensure they are equivalent |
| 2. Receives configuration file from WiFi module, and activates corresponding motor | 2A. Connect the microcontroller to the WiFi Module, to communicate over UART<br><br>2B. Send configuration file over WiFi to the controller, note which motor should activate based on input file<br><br>2C. Run program, compare activated motor to expected motor to ensure they are equivalent |

## 2.3 Shutter Subsystem

The shutter subsystem is made of the motors which control the position (open/closed) of the shutters.

#### 2.3.1 Gear Motor

The gear motors will be used to open and close the shutters. With the uxcell Gear Motor, the shutters should be able to open and close with good speed as the motor functions at 200RPM. In order for us to pass the requirements that are in table 2, we needed to know if our motor would perform what we needed it to.

Torque requirements can be calculated based on the expected weight of the shutters. A set of aluminum shutters, when cut to the dimensions of a sound panel (1 ft. X 1 ft), at a tenth of an inch, the torque requirements are as follows:

$$\text{Mass of Aluminum} = 44.22526g / in^3$$

$$\text{Mass of shutters} = (44.22526g / in^3) * (12in * 12in * 0.1in) = 636.48g$$

$$\text{Rated Torque of Motors} = 0.2kgf.cm = 200gf.cm$$

$$\text{Stall Torque of Motors} = 0.8kgf.cm = 800gf.cm$$

Based on this analysis, the motor was expected to be able to rotate the weight of the aluminum shutters which it was capable of doing.

***Table 2. Gear Motor R&V***

| Requirement(s) | Verification |
|---|---|
| 1. Must have enough torque to open/close shutters | 1A. Mount motors on shutters (work with ECE machine shop to do this)<br><br>1B. Run test program to rotate motor while connected to shutter and observe if torque is sufficient |
| 2. Should be able to open/close shutters in under 2 seconds | 2A. Mount motors on shutters (work with ECE machine shop to do this)<br><br>2B. Run test program to rotate motor while connected to shutter and record time taken for shutters to oscillate |

## 2.4 Power Supply

The power supply is used to power the components in the system. These components are the motors, and the WiFi IC. After realizing that we would not be able to change the batteries when we needed to along with the cost we decided to have the power supply consist of an AC to DC adapter. We also used a voltage regulator so that our WiFi chip could be powered.

### 2.4.1 Power Supply Unit

The power supply will be a 5V 15A 75W AC to DC adapter, and it will supply power to the motors and other components of the system. As shown in table 3, we just needed our power supply and voltage regulator to supply the correct voltage which we ended up getting 5.19V for our power supply and 3.283V for the voltage regulator.

***Table 3. Power Supply R&V***

| Requirement(s) | Verification |
|---|---|
| 1. The Power Supply must be able to supply a voltage of 5V. | 1A. Measure the power supply with a voltmeter and make sure it is within 5% of 5V. |
| 2. The Voltage Regulator must be able to supply a voltage of 3.3V. | 2A. Measure the power supply with an voltmeter and make sure it is within 5% of 3.3V. |

## 2.5 WiFi Module

The Wifi module will be used to receive information from the microcontroller via UART and then transmit that information across a wireless network in order to create a remote monitoring functionality. The power supply will power the WiFi IC with 3.3V.

### 2.5.1 WiFi IC

The WiFi IC will be implemented using the ESP8266 chip by Espressif Systems. The chip will be able to transmit data through a wireless network via TCP/IP connections.

***Table 4. WiFi IC R&V***

| Requirement(s) | Verification |
|---|---|
| 1. Must be able to receive experiment configuration file (JSON) from server | 1A. Connect the microcontroller to the WiFi Module, to communicate over UART<br><br>1B. Connect the WiFi module to remote server (via program)<br><br>1C. Send configuration file to the chip<br><br>1D. Check that the configuration file was received by the WiFi chip by having some acknowledgement action on the controller (rotate motor, light LED, etc.) |

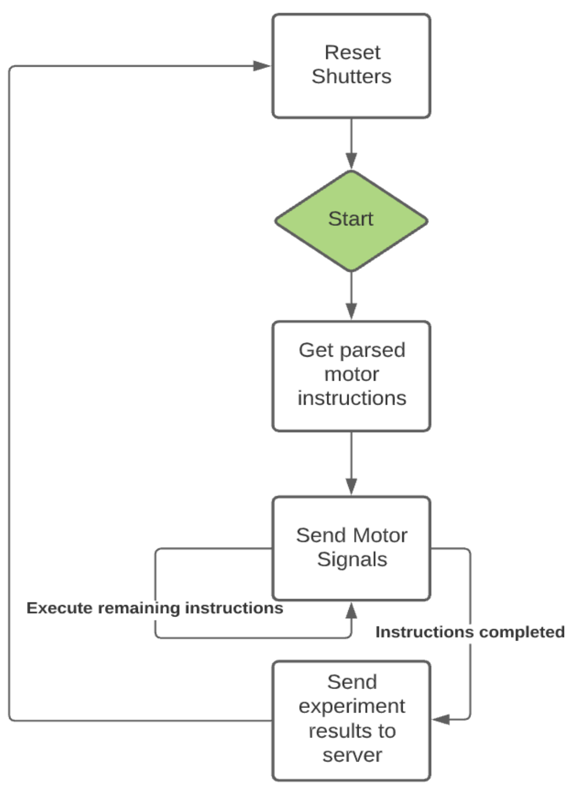
## 2.6 Software

As shown in Figure 6 (below), there are 3 components to the software for this project: the controller (arduino icon), the coordination server (server icon), and the experiment server (cloud icon). Together, these systems allow for collection, distribution, and delegation of the submitted experiment instructions. Users of the system will submit their experiment configuration files to the experiment server where they will be stored until requested by the coordination server. Once there, the coordination server will parse through the instructions and delegate to the controllers so that the correct shutters can be open and closed for the correct duration.

### 2.6.1 Controller

The controller software is responsible for receiving parsed instructions from the coordination server and applying these instructions to the shutter it is responsible for. This involves tracking the current state of the shutter, as well as the current time since each state is associated with a duration, and when the state is supposed to change (per the instructions) the controller is responsible for activating the motor with the correct direction and speed to make the shutter reflect the change in state.

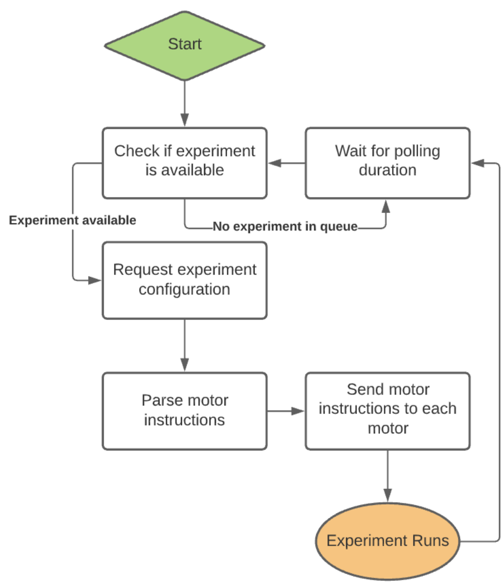
***Figure 3. Controller Software***



### 2.6.2 Coordination Server

The coordination server is the middle layer of communication between the controller and the experiment server. It exposes an API endpoint to the controller that forwards requests to the experiment server. The controller submits a request to check if an experiment is ready. When this happens, the coordination server forwards the request to the experiment server. If there is no experiment available, the coordination server tells the controller to continue waiting. If an experiment exists, the coordination server sends another request to the experiment server to fetch the experiment configuration, and serves it to the controller. In Figure 4, the polling element is removed from the controller and put in the coordination server. This is because when scaled up, the coordination server would be responsible for polling and checking if an experiment is ready. If one is ready, it will fetch the config as described above, parse the motor instructions, and delegate instructions to all the motors in the system. This contrasts what the project is in practice, since the polling code and requests originate at the controller.

***Figure 4. Coordination Server***




### 2.6.2 Experiment Server

The experiment server queues submitted configuration files and serves them to the coordination server when requested. LIke the coordination server, it is a basic API service which has requests submitted to it. When the coordination server wants a configuration file, it submits a GET request and when one wants to submit a configuration file to the system, they submit a POST request. In the future, this experiment server will be connected to the frontend, likely a sort of submission portal, to allow the researchers to easily submit configurations for their experiments.

***Figure 5. Experiment Server***

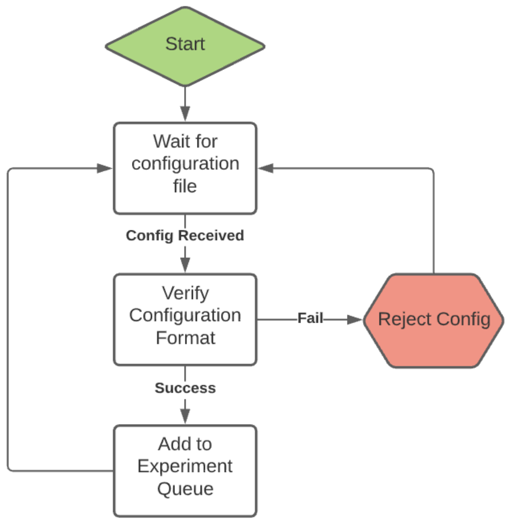



***Figure 6. Software Flow Diagram***

## 2.7 Circuits

**Figure 7. H Bridge Motor Driver**

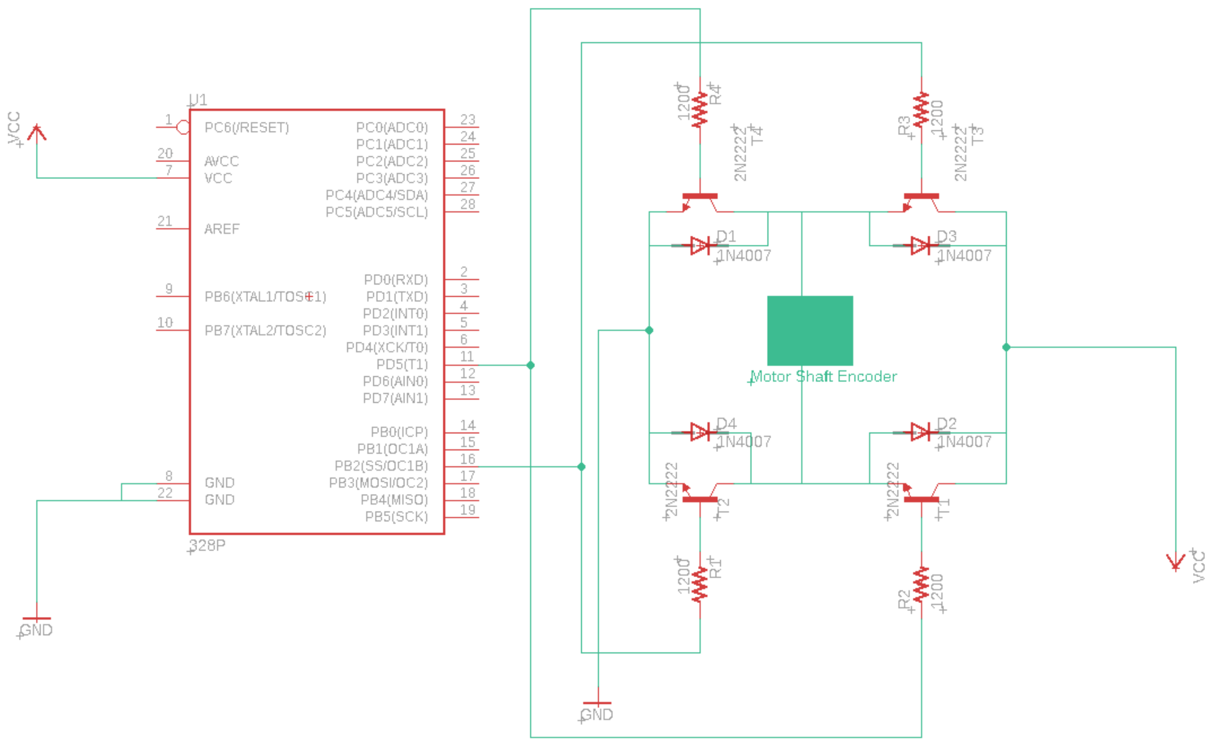
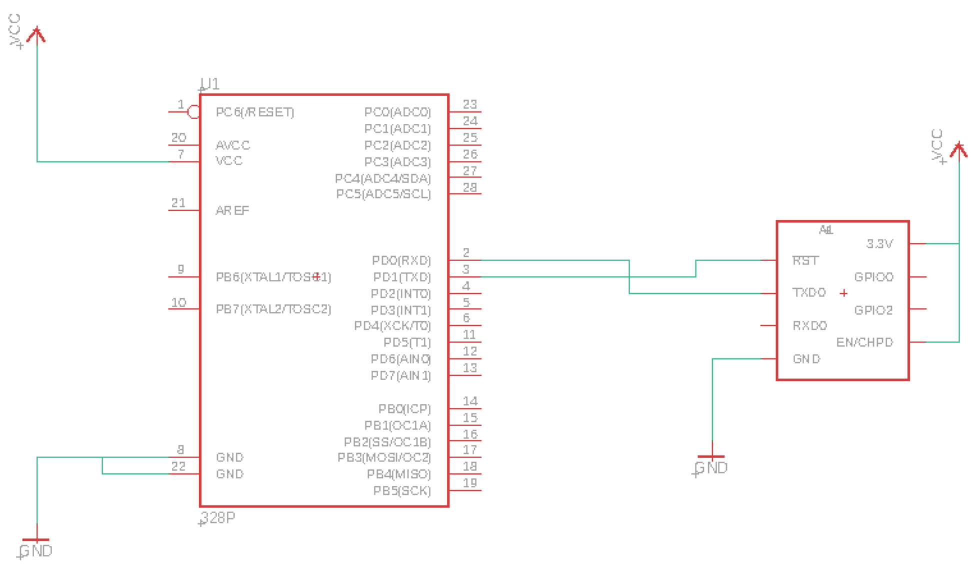



**Figure 8. Microcontroller and WiFi Module Schematic**



# 3. Verifications

## 3.1 T60 Reverberation Verification

One of our high level requirements was for our blinds to make a substantial impact on the acoustics of the room. This was measurable by our T60 reverberation time. T60 reverberation time is the time it takes for sound to decay by 60dB [7]. To test this we were able to put our shutter system into a small closet and took out all of the clothes that were already in the closet. Using a software called Room EQ Wizard we were able to calculate the reverberation time in the closet with different frequencies. As shown in figure 9, with a frequency of about 100 Hz we can see that with the blinds open, where the sound will make it into the sound panels, that the reverberation time was about 3.1 seconds. With that same frequency when the blinds were closed, where the sound bounces off of them as if they are a wall, the reverberation time was about 4.2 seconds as you can see in figure 10. As you can see we were able to improve the T60 reverberation time at a 100Hz frequency by 1.1 seconds. If the closet was completely filled with multiple of our shutter subsystems then the improvement would be expected to be greater.

**Figure 9. Open Blinds T60 Reverberation time vs Frequency**

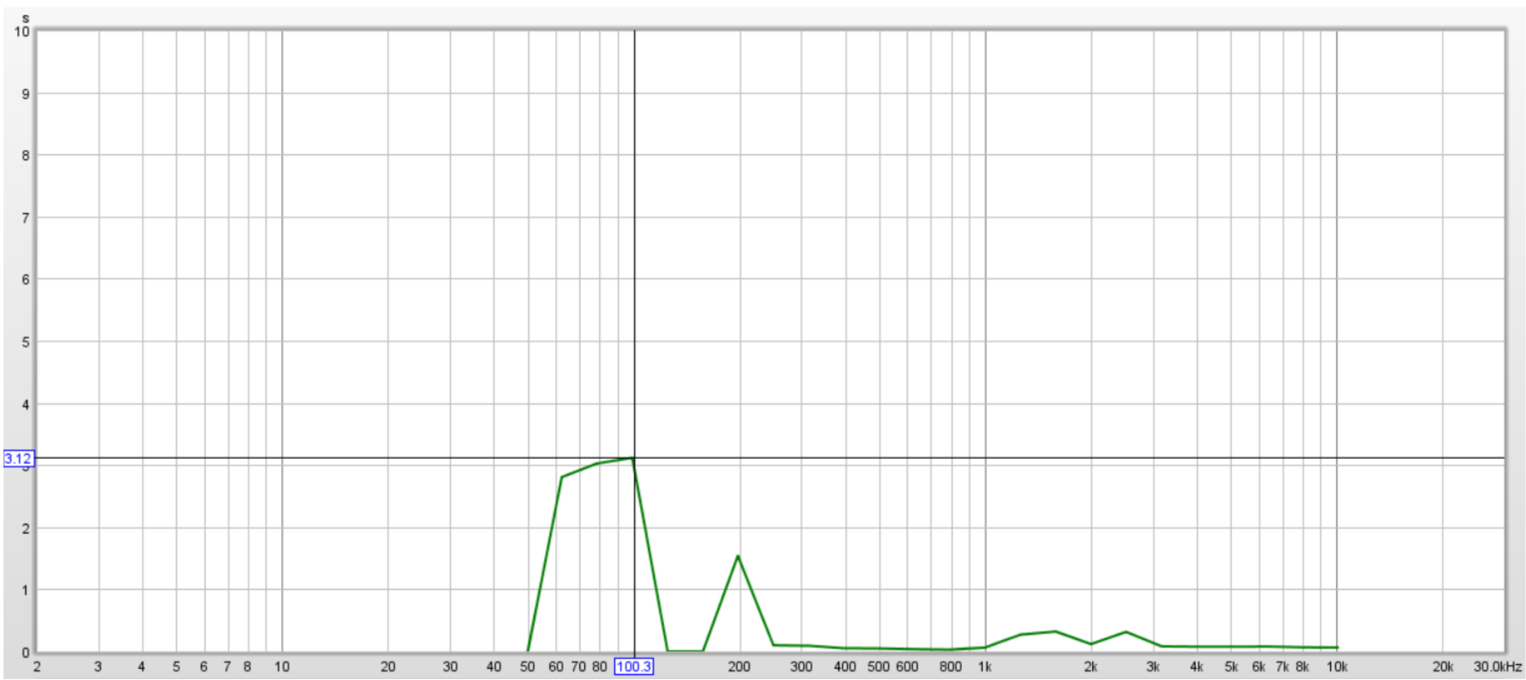
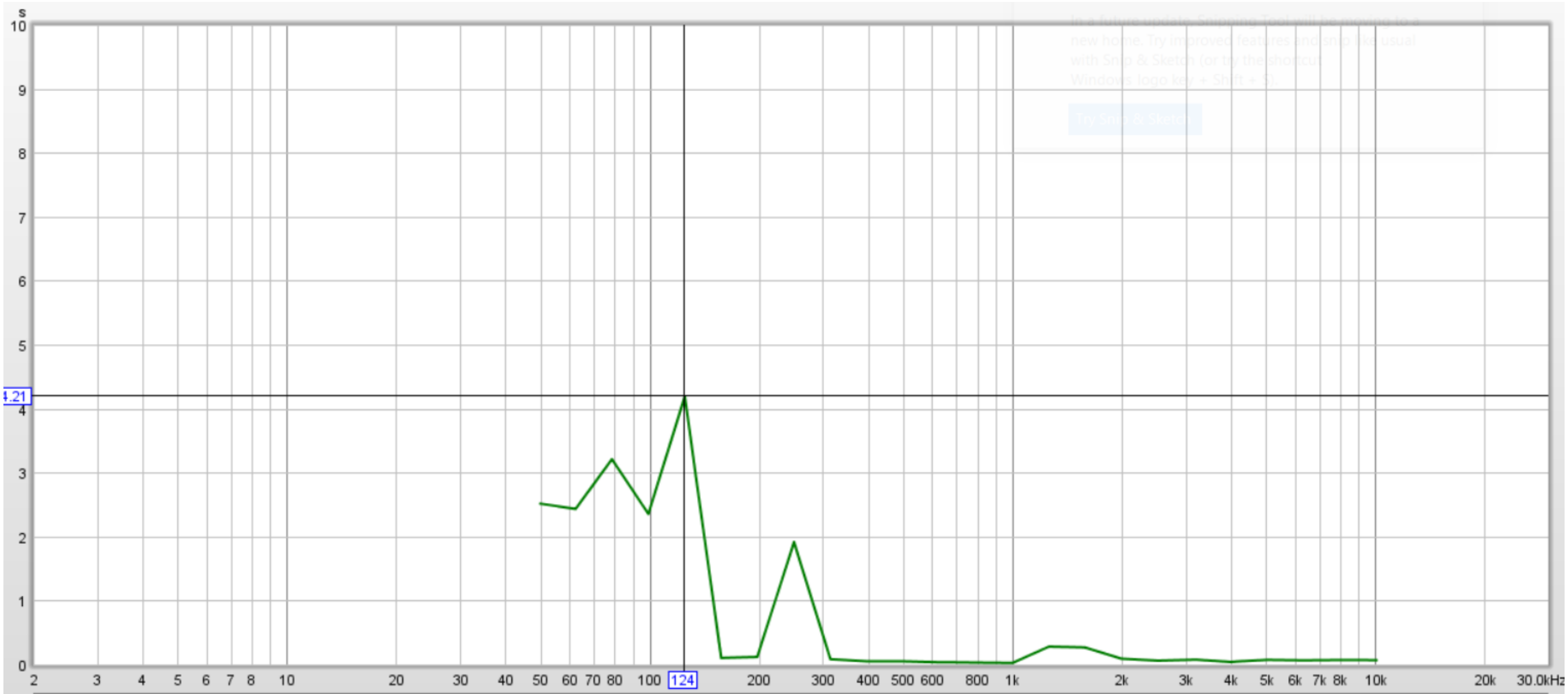


**Figure 10. Closed Blinds T60 Reverberation time vs Frequency**




# 4. Cost and Schedule

## 4.1 Cost Analysis

### 4.1.1 Labor

The labor rate from our team was calculated from the average salaries of Illinois ECE graduates [5]. Since the three members of the team are Computer Engineering, all the hourly rates were calculated based on the Computer Engineering salary, $96, 992. For the majority of the course, the team worked about 15 hours a week on design and construction of the system. This totals to 120 hours worked, and a total labor cost of $5595.6 per employee. The total labor cost is then 3*5595.6= $16786.8.

***Table 5. Personnel Labor Cost***

| Employee | Rate | Hours Worked | Labor Cost |
|---|---|---|---|
| Rishi | $46.63 | 120 | 5595.6 |
| Zachary | $46.63 | 120 | 5595.6 |

| | | | |
|---|---|---|---|
| Rajat | $46.63 | 120 | 5595.6 |

### 4.1.2 Parts

In Table 2 (below), is the initial list of parts that is necessary to complete the project. This does not include cost of construction materials, but focuses on the electronic parts integral to each subsystem. The total cost of the parts is expected to be $110.68.

***Table 6. Parts Cost***

| Module | Part | Quantity | Cost per Unit | Function |
|---|---|---|---|---|
| Control Unit | ATMEGA328P U | 1 | $2.40 | Microcontroller chip which connects components and controls their state |
| Control Unit | TLC5940NT | 1 | $8.95 | IC that allows expansion of PWM pins from ATMEGA, allows for scaling up |
| Shutter Subsystem | Gear Motor + Encoder | 4 | 4x$7.40=$29.60 | Controls shutter movement to open and close |
| Wifi Module | ESP8266 | 1 | $6.95 | Receives configuration file from internet server using WiFi |
| Power Supply | 5V 75W Power Supply | 1 | $26.99 | Provides power to entire system |
| Shutter Subsystem | Aluminum Shutters | 1 | $25.79 | Provides a basic shutter mechanism build off |
| Other | Electronics (Resistors, capacitors, etc.) | N/A | *$10.00<br>*Estimated | Required to create safe, functional circuits |

### 4.1.3 Total Cost

The total cost of the parts is $110.68 and the total cost of labor is $16,786.80 bringing our total cost to be 16,897.48.

## 4.2 Schedule

***Table 7. Weekly Schedule***

| Week | Objective | Rishi | Zachary | Rajat |
|---|---|---|---|---|
| **2/28 - 3/6** | Design Document and Check, order parts | Complete Design Document, work with Zachary on motor design | Research motors and servos and finalize which parts to use, design circuit | Design Circuit Schematic for ESP8266 and ATMEGA |
| **3/7 - 3/13** | Design Review, start designing PCB for ATMega and WiFi Module | Begin web application development, assist with PCB | Start PCB Design, work with Rajat | Start PCB Design, work with Zachary |
| **3/14 - 3/20** | Finish initial designing PCB and place PCB order, make basic prototype with Arduino Board and parts, work with machine shop to mount motors to shutters | Continue web app dev, research connecting applications to ESP8266 WiFi Chip, assist with PCB design as needed, work with machine shop to mount motors | Complete PCB Design, add motors to PCB and any other necessary components, verify PCB design with TA, place PCB order | Work with Zachary on PCB Design, build basic prototype with motors and shutters on Arduino Board, verify torque requirements are met |
| **3/21 - 3/27** | Verify PCB order was correct, order again if not. Begin constructing entire system with arduino board | Compile components onto PCB board (soldering if needed), mount board to system, work with Zachary | Find way to neatly attach board to system and mount board, work with Rishi and Machine Shop | Mount motors onto shutters, find way to attach shutters together neatly, work with Zachary and Machine Shop |

| | | | | |
|---|---|---|---|---|
| **3/28 - 4/3** | Test system to ensure functionality is as intended | Test system, address any issues/bugs that we observe | Test system, address any issues/bugs that we observe | Test system, address any issues/bugs that we observe |
| **4/4 - 4/10** | Provide progress reports, begin working on presentation, continue debugging | Work on presentation, continue debugging | Work on presentation, continue debugging | Work on presentation, continue debugging |
| **4/11 - 4/17** | Continue presentation work, plan demo | Create demo configurations to show functionality, continue working on presentation | Continue working on presentation, distribute responsibilities for presentation | Continue working on presentation, distribute responsibilities for presentation |
| **4/18 - 4/24** | Mock demo | Mock Demo | Mock Demo | Mock Demo |
| **4/25 - 5/1** | Final demo, and mock presentation and report due | Refine presentation, work on final report | Refine presentation, work on final report | Refine presentation, work on final report |
| **5/2 - 5/8** | Final presentation and report due | Complete final report and submit | Complete final report and submit | Complete final report and submit |

# 5. Conclusion

## 5.1 Accomplishments

Overall, our project was successful in meeting the high level requirements that were set initially. Most importantly, our shutter system was able to significantly change the acoustics of the room. This was quantitatively verified by measuring the T60 Reverberation Time when the shutters were open versus closed. Our second point of success was that the end-to-end time it takes from sending a configuration to the complete execution of the shutter is under 60 seconds. Lastly , although we were not able to actually assemble multiple units of our shutter system, we were able to make the system easily scalable up to a 5x5 set of panels.

In terms of the broader impacts of our project, we hope that this innovation will help researchers more efficiently find solutions for hearing aid devices and other audio-related problems. This project will also help researchers be able to run experiments from wherever they want in the world.

## 5.2 Future Work

1. **Move polling software**

As discussed in section 2.6.2, the current implementation of the project has the polling software on the control unit as opposed to in the coordination server. This currently is not an issue since there are no other controllers to make redundant requests. However, when scaled up, all controllers would be polling for experiment configuration files, and this would repeat operations across controllers in terms of parsing the configurations as well as submitting web requests. This can be mitigated by moving the polling to the coordination server. This would allow for only one request for a new experiment as opposed to N (number of controllers/shutters) requests per polling cycle. Additionally, it allows the more powerful computer hosting the coordination server to do the parsing, and delegate only the instructions each controller needs. This contrasts the current solution, where the controller parses the whole config for its own instruction. If multiple shutters existed, each controller would have to parse through the config for its instructions, as opposed to only doing it once at the coordination server level. For these reasons, moving polling to the coordination server is critical in allowing the system to be easily scalable.

2. **Store experiment results**

Currently, experiment data (success/fail) is being dismissed since no actual significant data is being collected. However, this would change once the system becomes integrated with the automated audio lab. The work to add this feature involves adding an API endpoint on the experiment server to submit experiment results as well as a way to store them. Potential solutions involve some sort of database backed server, like a SQL or MongoDB backed system. Additionally, the controller software needs to be updated to collect data and submit a request to the experiment server to store the results.

3. **Synchronize controller clocks**

Much like a distributed system, each controller functions as an independent "node". However, since the states of the shutters are time based, this means that each controller has to be on the same clock. Potential solutions include implementing a software system to ensure the clocks are synchronized, but this likely will not be fast enough to maintain the integrity of the experiment. Consequently, the controllers would have to be wired to an external clock, which is connected to all the shutters in the system.

**4. Frontend submission interface**

For the sake of testing and demoing the system, we were able to submit HTTP requests through an API client (Postman). However, this process is cumbersome and not intuitive, and researchers shouldn't be expected to submit configs this way. Consequently, there is a need for an intuitive client-facing interface that allows authorized researchers to submit configuration files to the experiment server, as well as providing access to results and status updates for their experiment.

## 5.3 Ethics and Safety

The sound shutter project is fully automated and is intended for unsupervised use, and there are a number of associated safety concerns associated with the system. Most of these concerns are related to the motion of the shutters and potential damages to the motors driving the shutters. This is especially a concern because the priority is to contain any potential damages that might occur to just our system, because the lab would be unsupervised.

The most likely failure situations are motor burnout or an obstruction that prevents the shutter from closing. The other concern is motors overheating, but the motor should turn off automatically if it gets too hot, which can be treated as an offline motor. Additionally, the likelihood of this event happening is very low, considering the motors are only rotating back and forth occasionally, and only with a range of 180 degrees.

To handle such situations, the shutter subsystem software will be designed to recognize when a shutter is unable to close, or has lost connection (burnout, offline motor). The program will then mark the experiment as a failure to run, and log the error, specific to the exact panel with a malfunction. This logging will also aid the lab staff in identifying the problem, and ensuring that the issue does not occur again. This is in accordance with IEEE Code of Ethics #5, to acknowledge and correct errors [6].

Additionally, due to the valuable nature of experimental data, results from experiments run using the system should only be accessible by the researcher who commissioned the experiment. This is intended to follow the ACM Code of Ethics, 1.6., to respect privacy [8]. While we cannot verify the intentions of the lab as a whole, our iteration of this system will follow this guideline, since it is our priority to maintain privacy when relevant. Our system would share system reports, configurations, and any gathered data privately with the researcher, to prevent data from spreading in the community.

# 6. References

[1] Mark B., Christophe M., 2008. *The cocktail party problem: what is it? How can it be solved? And why should animal behaviorists study it?* [Online] *Journal of comparative psychology* Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2692487/ [Accessed 15 February 2021]

[2] Abby M., Heather F., 2013. *Why do people fitted with hearing aids not wear them?* [Online] *International journal of audiology.* Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3665209/ [Accessed 14 February 2021]

[3] National Institute on Deafness and Other Communication Disorders, 2017-2021. *Strategic Plan*. [Online] Available at: https://www.nidcd.nih.gov/about/strategic-plan/2017-2021-nidcd-strategic-plan#sp3 [Accessed 15 February 2021].

[4] Daniel M., 2017. *The Robotarium: A remotely accessible swarm robotics research testbed*. [Online] Available at: https://ieeexplore.ieee.org/document/7989200 [Accessed 1 March 2021]

[5] Illinois ECE, 2019. *ECE Salary Averages*. [Online] Available at: https://ece.illinois.edu/admissions/why-ece/salary-averages [Accessed 2 March 2021]

[6] Ieee.org, 2021. *IEEE Code Of Ethics*. [Online] Available at: https://www.ieee.org/about/corporate/governance/p7-8.html [Accessed 16 February 2021].

[7] C. Roberts, "What is reverberation time and how it is calculated?," *NoiseNews*, 28-Apr-2020. [Online]. Available: https://www.cirrusresearch.co.uk/blog/2018/04/what-is-reverberation-time-and-how-it-is-calculated/. [Accessed: 04-May-2021].

[8] Acm.org, 2021. *Code of Ethics*. [Online] Available at: https://www.acm.org/code-of-ethics [Accessed 16 February 2021].