

So-Lo

Midyear Design Review Report

Andy Weng (CSE), Dan-Michael Tiamzon (EE), Suzet Nkwaya (EE), Ming Shuai Chen (CSE)

¹**Abstract**—So-Lo is a device used to record meetings with friends or co-workers. Short for “Sound Locator”, So-Lo uses a sound location technique known as time difference of arrival to pinpoint the location of the source of a sound. When the sound is located, a motor equipped with a camera will turn towards the direction the sound came from. Whenever someone speaks, the camera will turn towards the speaker. This creates an automatic device that will record video and audio of the person speaking, enhancing the quality of a recorded meeting.

I. INTRODUCTION

Meetings occur in everyday life in which people gather and exchange essential information. Research found that the top five reasons why groups meet are: reconcile conflict, reach a group judgment or decision, solve a problem, ensure that everyone understands a specific topic, and facilitate staff communication [4]. Research has shown that most companies spend between seven and fifteen percent of their personnel budget on meetings [6]. Given the resources that are put into organizing meetings, it is imperative that all participants stay focused and remember the maximum amount of information discussed. In fact, according to a 3M study, “Unproductive meetings may cost organizations more than wasted dollars; time may be lost, morale may decline, and productivity may be reduced”[5].

The core takeaway from a meeting is information. The most common way information is recorded is through writing. Writing down information has its own problems such as “it may lose accuracy since the minute preparer may not remember or interpret correctly or is biased”[7]. Video and audio recording devices have been beneficial to keep information correct and be accessed easily. There are products on the market for this function however, most of the affordable options are either manually controlled, offer only audio, or a stationary camera. The high-end products are incredibly expensive. For instance, Polycom’s “CX5500 Unified Conference Station” costs around \$5000 [10]. Our project, So-Lo, proposes a solution to this problem which delivers quality audio and video, easily accessible recordings, and is inexpensive.

So-Lo is a device that records information in video and audio format from a conference. So-Lo uses the Time Difference of Arrival (TDOA) technique to determine the location of the person speaking, and then focus the camera on

the person. Three microphones and a Raspberry Pi are used to implement TDOA in So-Lo. Once the location of the speaker is determined, a motor equipped with a camera will point in the direction of the speaker. This will give a clear and focused recording of the person speaking so that playback will provide good quality and focused recording of the person speaking. The scope and implementation of So-Lo can go beyond the conference rooms. It can be used for security purposes. For example, the device can be installed in a store. During after hours, if someone were to break in, they would make a loud noise, causing the camera to face the intruder.

After extensive research and consulting with our advisor and other professors we have laid out our system requirements as shown in Table 1.

Requirement	Specification
Real-Time Sound Locating	Determine the source of a sound given as a relative angle instantly
Simple Setup	No setup required.
Microphone Array	3 electret microphones configured in an equilateral triangle.
Detect sound from at least 3’ away.	3 omnidirectional electret microphones with adjusted gain to detect sound within a certain distance.
Detect human speaking voice.	Assign a noise level for the system to filter out unwanted sounds.
Prevent wires from entangling due to motor	Use a slip ring to address the issue of wire entanglement.

Table 1. System Requirements

II. DESIGN

Overview: So-Lo is designed to work in quiet, small to mid sized meeting rooms; where it will be assumed that only one person is talking.

So-Lo used TDOA technique to obtain the sound source location. Compared to other sounds localization

¹A. Weng majors in Computer Systems Engineering.
D. M. Tiamzon majors in Electrical Engineering
M. S. Chen majors in Computer Systems Engineering
S. Nkwaya majors in Electrical Engineering

techniques, TDOA is relatively easy and cheap to implement. Some of the other methods of obtaining either sound source direction or sound source location include particle velocity or intensity vector and triangulation. For instance triangulation requires to know the angle at which the sound is arriving at the microphone

To implement TDOA, So-Lo uses an array of three omnidirectional electret microphones to detect various sounds: in this case the project focuses on detecting human voice. The microphones will be placed one foot away from each other, forming an equilateral triangle. When a person speaks the sound will travel and get to the 3 microphones at different times. This time difference will then be captured, and used by a custom software stored in a Raspberry Pi to determine the order in which the microphones detected the sound and use this data to calculate approximately the angle from which the sound came. The angle obtained is then used to point the camera in the speaker direction via a DC motor.

Block Diagram

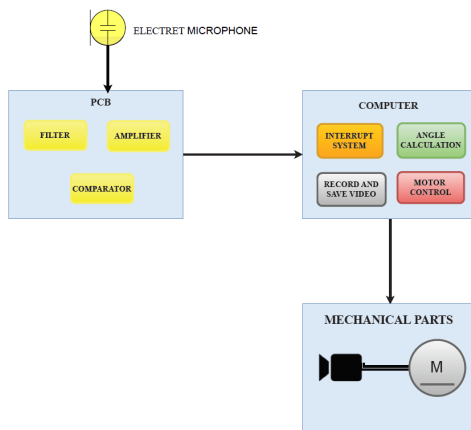


Figure 1. Block Diagram

A. Filter

The sound detected by the electret microphones will go through different stages of analog signal processing before being used by the Raspberry Pi. The first step is the filtration of the signal. Figure 1 shows the filtering circuit used in So-Lo. The filter's goal is to reduce the amount of noise and focus on the human voice, the sound first goes through a lowpass filter that has a cutoff frequency of 500Hz. The signal is then passed through a highpass filter with cutoff frequency of 100Hz. The combination of the high-pass and low-pass filter makes a 100Hz-500Hz bandpass filter, that is good enough to cover the human voice frequency range.

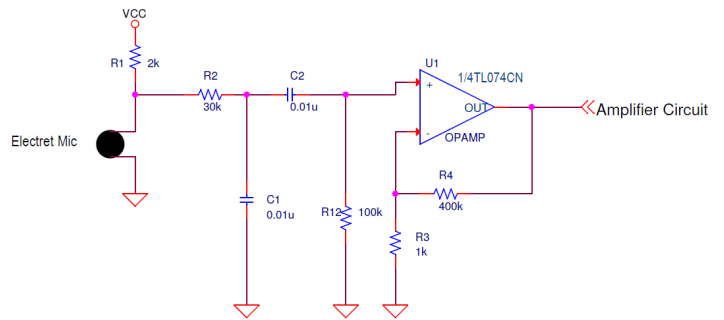


Figure 2. Filter circuit

B. Amplifier

Because the signal coming from the microphone is very small (a few millivolts), it has to be amplified in order to make the processing easier to be used as the input for the rest of the system. The amplification is done by the TL074CN chip, which contains four independent op-amps. So-Lo has two amplification stages (see Figure 3 below). The first stage has a 500 gain, the second stage has a gain that can vary from 5 to 2.5. In addition to increasing the overall gain of the circuit, the second stage gives the ability to tune the amplification of the system. When the electret microphone are not connected to an amplifier, the output was in order of millivolts. The team tested the amplification of the signal coming from the microphone by speaking or playing music then recording the value of the voltage at the output of the amplifier. The amplification stage allowed the output to go up to 15 V. Because not two components have exactly the same properties, even if a the same circuit it replicated for all three microphones, they may end up having different gains. Having the ability to tune the gain makes it possible to match them. After the Amplification stages, the signal goes through another high-pass filter to remove the DC offset.

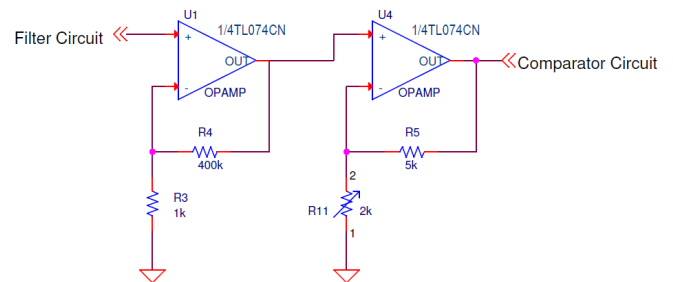


Figure 3. Amplifier Circuit

C. Comparator

Originally, after filtering and amplifying the signal, our team planned to use an analog to digital converter (ADC). However after testing, it was concluded that the ADC was not ideal for this project. The problem with ADCs is that to be able to convert a sound signal, they need a high sampling rate, and their implementation can be challenging. That's why the ADC was replaced with an LM311N comparator (see Figure

4 for comparator circuit). This project requires to detect sounds, therefore the comparator is set up to output 0 V when there is no sound and 15 V when a sound is detected (when someone speaks). In order to achieve this, the comparator will take in the amplified sound signal as an input and compare it to a reference voltage that corresponds to the voltage created by the ambient noise. During lab testing, when there was just ambient noise in the vicinity of microphone, the output of the comparator was 0 V; however, when a person spoke or clapped the output of the comparator went from zero to about 15 V, this change was displayed on an oscilloscope. Since all meeting rooms will have different settings, and therefore different ambient noise levels, the reference voltage can be set to a desired value using a potentiometer. When the comparator outputs 15 V signal, it first goes through a voltage divider before going to the GPIO of the Raspberry Pi because the latter functions with voltages ranging from 3V-5.5V. This signal is then used by the Raspberry Pi to trigger an interrupt system.

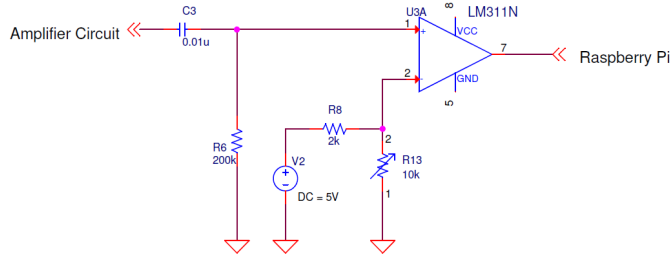


Figure 4. Comparator Circuit

D. Interrupt System

The system is required to pick up sounds from three microphones. It requires three microphones which produces the time differences of arrival between the 1st and 2nd microphone and the 1st and 3rd microphone. Since the sound can originate from anywhere, the system must know which microphone received the signal first, second, and third. The Interrupt System is designed on the Raspberry Pi in a way that the microphones are connected to 3 GPIO (General Purpose Input Output) channels which the Pi determines which channel received the signal first. The channels are set up at logic low and receive a signal from the microphone array circuit. When a sound is picked up by the microphones, they produce a high voltage, which is what the interrupts are looking for.

The instant a noise is picked up by the microphones, three signals will be sent to the GPIO channels. The channel that receives a signal first will have its time of arrival marked and the program will mark that this time is the first microphone used. The same procedure will apply to the remaining two channels that are waiting for an interrupt. Once all times and order of microphones have been recorded, the system will take the time difference between the 1st and 2nd microphone and the 1st and 3rd microphone. These are the time differences needed for TDOA.

This subsystem was tested using a switch circuit connected to the Raspberry Pi. Figure 5 visualizes the test implementation of this subsystem. 3 input pins of a dip-switch is powered up with 3V from the Raspberry Pi and the corresponding output pins are connected to 3 GPIO pins of the Raspberry Pi. These switches simulate 3 microphones connected to the GPIO pins and when the microphone receives a sound, it sends a voltage to the GPIO pins. All switches are flipped from the off position to the on position because in the real microphone system, the three microphones will receive a sound at very similar times with time differences of arrivals in the microsecond range and so by flipping all the switches at the same time mimics the effect of the microphone subsystem.

The software in the Raspberry Pi displays which pin receives the signal first, second, and third as well as what time the voltage arrived at the GPIO pin. The software also displays the time differences between the 1st and 2nd microphones and the 1st and 3rd microphones.

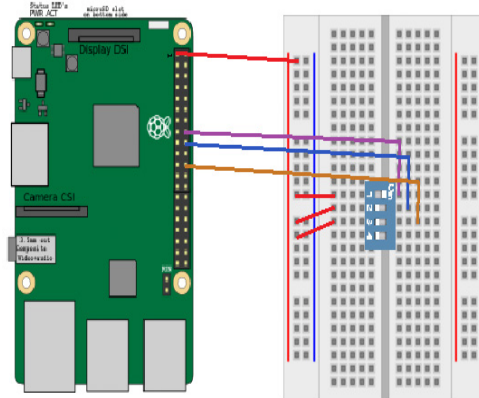


Figure 5. Interrupt System Simulation

E. Angle Calculations

The angle calculation portion of this project is very essential to the overall system. It is purely software based and it takes in an output from the interrupt subsystem using nonlinear calculation methods and finally the pythagorean theorem. It outputs an angle which directs the motor.

The subsystem works by utilizing the Time Difference of Arrival (TDOA) technique which is also used in the interrupt subsystem. Because there are three microphones placed a certain distance away from each other in a triangle formation, there will be a time delay for when each of the mic picks up the source of sound. Each microphone will yield an equation of a circle which all have different center points and radiuses. Looking at Figure 6. Concept of TDOA, there will be three nonlinear equations that all have three unknowns, (x_0, y_0, r) . Where x_0 and y_0 is the position on the coordinate plane and r is the distance to the closest microphone, but we don't use r .

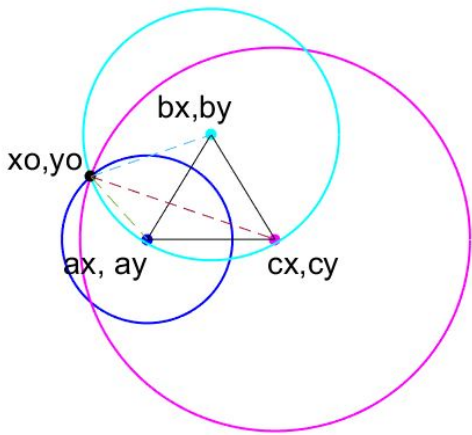


Figure 6. Concept of TDOA

Sequentially, after the (x0,y0) position of the source of sound is known, the angle can be calculated. Figure 7. Example Calculations of TDOA demonstrates how the angle is calculated. First, an imaginary coordinate plane is drawn with the origin centered at the center of the equilateral triangle above. Using offsets, the angle can be calculated using the Pythagorean theorem.

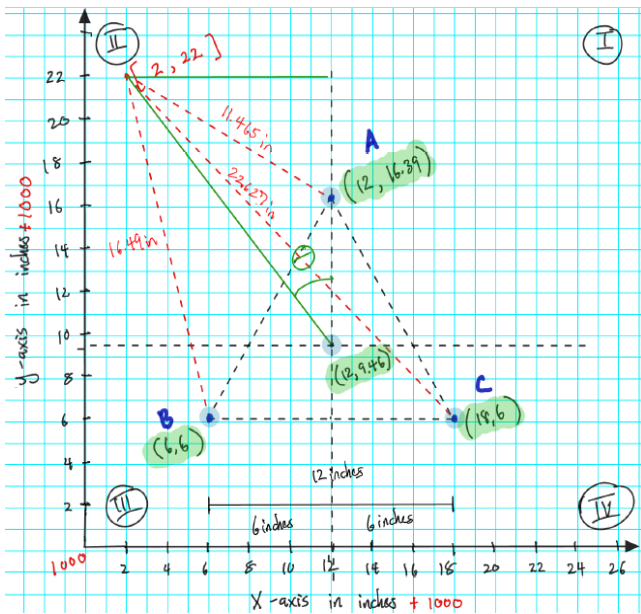


Figure 7. TDOA: Example Calculations

There are a few things that should be noted here. The position of the center of the triangle is at (1012, 1009.46). The units are in inches and the imaginary coordinate plane centered at the triangle is only used for angle calculations. The main graph itself is at the upper right hand first quadrant because the graph starts at (1000,1000). The reason for that is so negative numbers don't come into the equation. Figure 8.

TDOA Equations are the three main equations used to solve this problem.

$$\begin{aligned} (ax - x_0)^2 + (ay - y_0)^2 &= r^2 \\ (bx - x_0)^2 + (by - y_0)^2 &= (r + 346bs)^2 \\ (cx - x_0)^2 + (cy - y_0)^2 &= (r + 346cs)^2 \end{aligned}$$

$346m/s \Rightarrow 13622inches/sec$

Figure 8. TDOA Equations

Looking at Figure 8, the green colored variables are what is known, (ax,by) is the position of microphone A, and so on. The variable “bs” is the difference in time it takes the source of sound to get to microphone B and A and the variable “cs” is the difference in time it takes the source of sound to get to microphone A and C. Note that “bs” and “cs” is not always going to be related to microphone B and microphone C. These variables can change depending on the order of microphone which received the source of sound first, (bs comes before cs always). The two variables are the outputs from the interrupt subsystem and is the only input for this subsystem required to make an angle calculation output for the motor. The red colored variables are the three unknowns that need to be calculated. (x0,y0) is the position of the source of sound.

The technology used to implement the calculations is the Python language which is flexible because it can run on the Raspberry Pi 3 OS and PC/MAC OS. Moreover, to solve a system of linear equations, several Python packages had to be installed such as numpy and scipy. These modules can accurately solve systems of nonlinear equations. It can be ran on both Python 2 and 3.

A test implementation was conducted during MDR presentation for this subsystem. The code in Python asks for user input of the source of sound (x0,y0). The code then outputs the calculated distances from the source of sound to each of the three microphones as well as the time it takes to reach the three microphones. The code then outputs the (x0, y0) which mentioned above is the source of the sound. This position should be precisely the same as what the user inputted, which makes the check successful. The program then outputs the angle that the motor should turn. This angle assumes that the imaginary x-axis centered on the triangle is the 0° mark. If you look at Figure 7, the user input is (2,22). The program would output about 128° (90° from first quadrant and 38° from second quadrant).

The results of these tests show that under ideal circumstances and assuming that the speed of sound at room temperature is 346m/s, the calculations based on the ideal bs and cs will give the correct user position (source of sound) and it will output the correct angle. [11]

F. Motor

The purpose of the motor is to direct the camera towards the direction of the speaker. This part of the system will use a motor with slip ring, a motor driver, and an encoder. The motor that will be used for So-Lo is the DC Maxon Motor 310005 (shown in Figure 9) [3]. The motor driver that will be used is the L293D motor driver chip. The encoder that will be used is the E5 Optical Encoder. The purpose of the motor is to change the angle the camera will be facing. By itself, the motor is only capable of spinning in one direction. This could be a slight problem in the functionality of our device because there can be a shorter path to a certain angle if the motor spins in the other direction. For example, if the motor is facing 0° and someone speaks at 90° , the motor would turn clockwise 270° to face that angle. To address this problem, the L293D motor driver chip was used.



Figure 9. Maxon DC Motor

The L293D chip, shown in Figure 10, is designed to control the direction the motor turns by reversing the current through the motor. Its secondary purpose is to supply power to the motor since the Raspberry Pi is incapable of supplying enough power to the motor directly. [2]

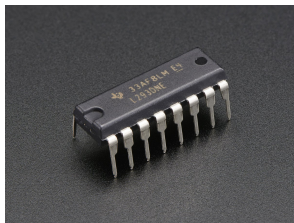


Figure 10. L293D Motor Driver Chip

In order for the motor to accurately determine the direction it is facing, an encoder is used. The encoder of choice is the E5 Optical Encoder. The encoder contains a disk that is wrapped around a shaft. As the motor spins, the shaft spins which causes the disk on the encoder to spin. There are opaque and transparent regions on the disk. A light shines through the disk and into a sensor on the other side. If the sensor senses the light shining on it, pin A on the encoder outputs a 1. As the disk spins, the region that the light shines through alternates between opaque and transparent. The result

is a series of pulses from channel A and B. A diagram of the basics of an encoder is shown on Figure 11. Both channels together are needed to determine the direction the motor is spinning but since we are already able to control the direction the motor spins, only one channel is required. By counting the number of pulses, the motor can determine the direction it is facing relative to its initial position. When the motor makes one revolution, the encoder outputs ~ 18000 pulses. To turn the motor to 90° , just turn the motor counter-clockwise until 4500 pulses are counted since 90° is a quarter of 360° . [1]

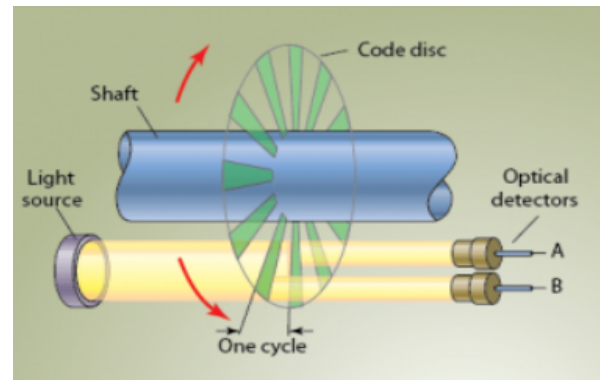


Figure 11. Fundamentals of an optical encoder

The motor driver and encoder is programmed using Python code on the Raspberry Pi. The techniques to programming on the Raspberry Pi to control the motor driver and encoder will be similar to what we have learned in ECE 353 and ECE 354, Computer Systems Engineering I & II. In those courses, we gained experience with embedded systems and how to program an FPGA. The Raspberry Pi is significantly different from an FPGA, however. Therefore, we would need to learn the layout of the functionality of the Raspberry Pi to build this block.

To test the motor functionality, we programmed motor to accept a series of angles that will come from the angle calculation block. The motor's initial position is 0° . We input the angles 90° , 300° , 210° , and 20° . This will test if the motor can turn a certain angle, turn clockwise and counter-wise, and take the shortest path to the specified angle. The motor passed the test with flying colors.

III. PROJECT MANAGEMENT

The project has four subsystems, distributing one subsystem to each team member. S. Nkwaya designed the microphone array circuit, D. Tiamzon designed the interrupt system, M. Chen designed the angle calculation software, and A. Weng programmed the motor.

The team succeeded in completing our MDR deliverables. Table 2 provides the deliverables proposed for MDR and the results. The main purpose of our MDR Deliverables was to demonstrate the concept of TDOA.

Each subsystem was distributed to suit the expertise

and interests of each member. S. Nkwaya is an electrical engineering major whose forte is signal processing and therefore most comfortable with working on the microphone subsystem. D. Tiamzon is also an electrical engineering major who is interested in interfacing with hardware and software; so the interrupt system best suited his interests. M. Chen is a computer systems engineering major who likes programming so the angle calculation was best suited for him. A. Weng is also a computer system engineering major who likes in programming in the Raspberry Pi so the motor subsystem was best suited for him.

Despite our expertise in our individual parts, there were unavoidable challenges where the team needed to help each other out. Anytime any of us was stuck on a problem, we would ask the group for advice on a solution or workaround or even tackle the problem together.

Our team met on a weekly basis with our advisor to present updates and concerns. From September to December, we would all be present in the lab most days of the week. We often found ourselves working individually on each of our subsystems but at least twice a week we would all be present at the lab to collaborate.

Proposed Deliverable	Result
Design microphone array to detect sound.	Success. Microphones detect human voice displayed on a logic analyzer.
Determine the time difference of the time it takes for the sound to travel to each microphone.	Success. Software displays time differences between microphones and the order of which GPIO received the signal first.
Use TDOA to estimate the the source of sound as a relative angle.	Success. Software calculates and displays the relative angle of sound source.
Control the motor to turn to a specific angle.	Success. Given a specific angle, the motor will turn the amount of angle given.

Table 2. MDR Deliverable Table

IV. CONCLUSION

The team has designed four subsystems so far such as the microphone array circuit, interrupt system, angle calculation, and motor. The microphone subsystem is working properly, however it has only been tested in a straight line arrangement. The interrupt system is working properly with the manual switches. The angle calculation code is working properly given manual time delay inputs. The motor

subsystem is working properly, although it is rotating at a very slow pace. These subsystems demonstrated successfully during the MDR Presentation.

Our next step is system integration. The microphones need to be set up and working properly in a triangular arrangement. Then the interrupt system will be tested in conjunction with the microphones. When the microphones and interrupt system are working properly together, the angle calculation software can be integrated. The motor rotation needs to be sped up significantly while maintaining its accuracy. Afterwards, the motor will be integrated with the other subsystems. When the four subsystems are working together properly, the camera will be mounted on the motor and programmed to record and save video to the SD card on the Raspberry Pi. A power supply for the entire system will also be built on a PCB.

By the end of spring we hope to have our entire system functioning completely in which the microphones can detect sound from three feet away, record video and audio for one minute, and accurately point to the source of sound with minimal error in the angle calculation.

Acknowledgment

We would like to thank Professor Polizzi and Professor Ciesielski for their feedback that helped us improve our project. We would also like to thank Professor Soules who took the time to meet with us each week, and helped us stay on track and set our goals high.

REFERENCES

- [1] US Digital, "E5 Optical Kit Encoder" E5 datasheet [Accessed 20-Dec- 2016].
- [2] Texas Instruments, "L293x Quadruple Half-H Drivers" L293D datasheet September 1986 – Revised January 2016
- [3] Maxon Motors, "30mm, Graphite Brushes, 60 Watt" Maxon DC Motor 310005 datasheet May 2008
- [4] Monge, P.R., McSween, C., & Wyer, J. A profile of meetings in corporate America: results of the 3M meeting effectiveness study, Annenberg school of communications, University of Southern California, Los Angeles, CA, 1989.
- [5] 3M Meeting Management Team, and Drew, J. Mastering meetings : discovering the hidden potential of effective business meetings. New York: McGraw-Hill, 1994.
- [6] Doyle, M., & Straus, D. How to make meeting work: the new interaction method. New York, NY: Jove Books, 1982
- [7] Alex Waibel, Michael Bett, and Michael Finke, "Meeting browser: Tracking and summarising meetings," in Proceedings of the DARPA Broadcast News Workshop, 1998.
- [8] Texas Instruments, "TL07xx Low-Noise JFET-Input Operational Amplifiers " TL074CN datasheet September 1978– Revised June 2015
- [9] Texas Instruments "LM111-N/LM211-N/LM311-N Voltage Comparator " LM311-N datasheet mayr 199– Revised March 2013
- [10] Polycom "Polycom CX5500 Unified Conference Station" Polycom CX5500 datasheet [Accessed 5- Feb- 2017]
- [11] Process91, "Sound Triangulation," in StackExchange, 2012. [Online]. Available: <http://math.stackexchange.com/questions/187819/sound-triangulation> . [Accessed 8-Nov-2016]