RAYO

TUNE INTO
YOUR HEART

Design Package

William Cai, Julian Maravilla, Owen Ou, Michael Quach

Table of Contents

RAYO's Purpose

For many years, the mechanical stethoscope has been the conventional device for auscultation, which is the medical examination of bodily acoustics. Despite its simplicity and usefulness, the stethoscope has many drawbacks caused by its purely mechanical nature. Some notable drawbacks include inaccurate sound perception, uncontrollable sound amplification, and the lack of acoustic data collection. These issues can significantly hinder doctors' efforts to treat patients, as doctors are prone to misdiagnosis two out of five times and can receive permanent hearing damage due to the lack of volume control. Furthermore, without previously collected data to reference, doctors may diagnose patients inconsistently. RAYO is an electronic alternative to the mechanical stethoscope that aims to mitigate these shortcomings. Not only will RAYO provide top quality source isolation using state-of-the-art beamforming techniques, it will also allow users to easily control audio amplification as well as to collect or view acoustic data via a touch screen. Just like a traditional stethoscope, RAYO is designed to be accessible and portable.

RAYO is designed around the sum-and-delay Beamforming technique, which utilizes time delays between input signals to focus and isolate specific sound sources in space. The technique in reverse allows the user to input their own time delays between signals to create a desired angle of arrival. This creates a beam towards the direction of interest and allows the user to recreate the signal emitted from the desired source. Beamforming is widely used by modern smart devices such as routers and speakers to locate signal sources and achieve higher quality transmissions. As RAYO's main objective is to achieve high quality source localization, our application of Beamforming is of the near-field domain and is focused on isolating and retrieving acoustics from organs such as the heart and lungs. By optimizing the algorithm to very small degrees of variability, RAYO can be used to distinguish signals emitted from the obscure parts of vital organs and detect serious health issues such as heart murmurs and fluid accumulation in the lungs. Outside of medical applications, RAYO can also serve as a Beamforming testing platform used for newer, more developed Beamforming algorithms.

Our device will consist of a microphone array, a multi-channel analog-to-digital converter (ADC), a microcontroller, and a touch screen for user interface. Each microphone is encased in a transceiver, acting as a substitute to the typical stethoscope head which prevents additional noise from the area. The four-element microphone array is mounted upon a circular mount and is meant to be placed upon a person's chest or back. These microphones collect and amplify acoustic data from the body. Multiple acoustic signals from the several channels are then sampled and digitized by the ADC. The digitized signals are then fed into a microcontroller (Raspberry Pi) where signal processing algorithms (correlation, FFT, phase-shifting, etc.) are used to determine parameters such as heart rate. Users of the product will be able to access visualization and acoustic playbacks of the recorded signals. In addition, the users are also given access to the heart rate and a contour map indicating the estimated body location of the measured signals.

The design of the product is depicted in Figure 1 (shows final design/prototype of the product). The device is to be placed upon the chest of a patient lying flat and facing up on a bed. Using the

touchscreen, the medical expert or user can manually begin recording the heart beats with the same monitor displaying the heartbeat, heart rate, and estimated location of the heart. Once finished, the recording of the data can be stopped by the user and the data can be saved to the microcontroller. If the patient moves or the device is in a noise environment, the collected signals are susceptible to error and inaccuracy. In the future, RAYO can be further developed to gather heart and lung acoustics, ranging from low to high frequencies. In addition, it can also be developed to be able to detect heart problems automatically like heart murmurs through advanced software implementation.

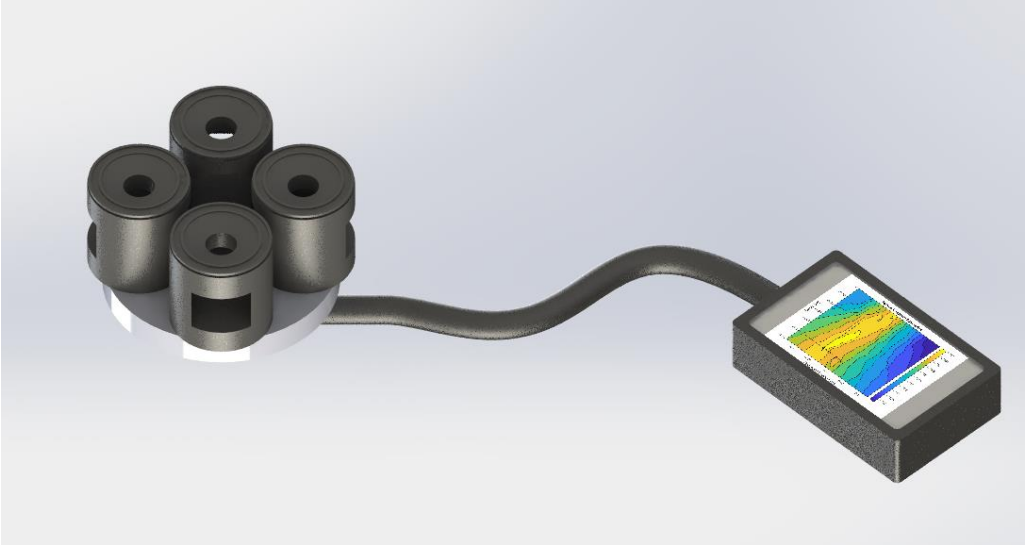


Figure 1: RAYO's Final Development Stage Product (after Capstone)

Target Specifications

In order to define our target specifications and to introduce our source localization algorithm, a quick introduction to delay and sum beamforming will be presented.

This derivation will involve a thought experiment. A point source in the near field emits a spherical wave front defined as,

$$f(t,r) = \frac{s(t - r/c)}{r}.$$

Where r is the distance between the source and the receiver of interest, $s(t)$ is the emitted signal, and c is the speed of sound in the environment. With a linear array of M elements, each of the received signals will be delayed with respect to their distance away from the source.

$$f_m(t,r_m) = \frac{s(t - r_m/c)}{r_m}$$

With proper delay, each of the received signals can be delayed in software so that their received signals line up in time. With this alignment, the signals can be summed together to reveal an amplified version of the emitted signal $s(t)$. This signal is defined as,

$$z(t) = \frac{s(t - r^o/c)}{r^o} \sum_{i=0}^{\mathrm{M}-1} \frac{w_{\mathrm{i}}}{r_{\mathrm{i}}}$$

where r^o is the center of the linear array, and w_m is a weight term applied to each receiver. This is the classical method of delay and sum beamforming. However, our algorithm exploits the backward propagation phenomenon. We can guess the location of the source by adjusting the time delays that corresponds to different points in space. When the time delays are at the approximate location of the source, the delayed and summed signal will have the highest energy when compared to another combinations of delays. Therefore, we can generate a contour map that describes where we expect the signal to be located. The combination of delays that reveals the highest energy signal will be the set of delays used to record the heart.

To define the necessary delay(s) (Δ_m) for each point on the grid, the law of cosines can be used to derive the necessary delay at each receiver with respect to the desired point on the grid.

$$\Delta_m = \frac{r^o}{c}\left[1 - \sqrt{1 + \frac{\left|\left(\frac{1-M}{2} + m\right)d\right|}{(r^o)^2} - 2\frac{|r^o - r^m|}{(r^o)^2}}\right]$$

The quantity r^m defines the distance between the m^{th} receiver and the desired point on the grid. A graphical representation of the backward propagation source localization algorithm is shown in Figure 2.

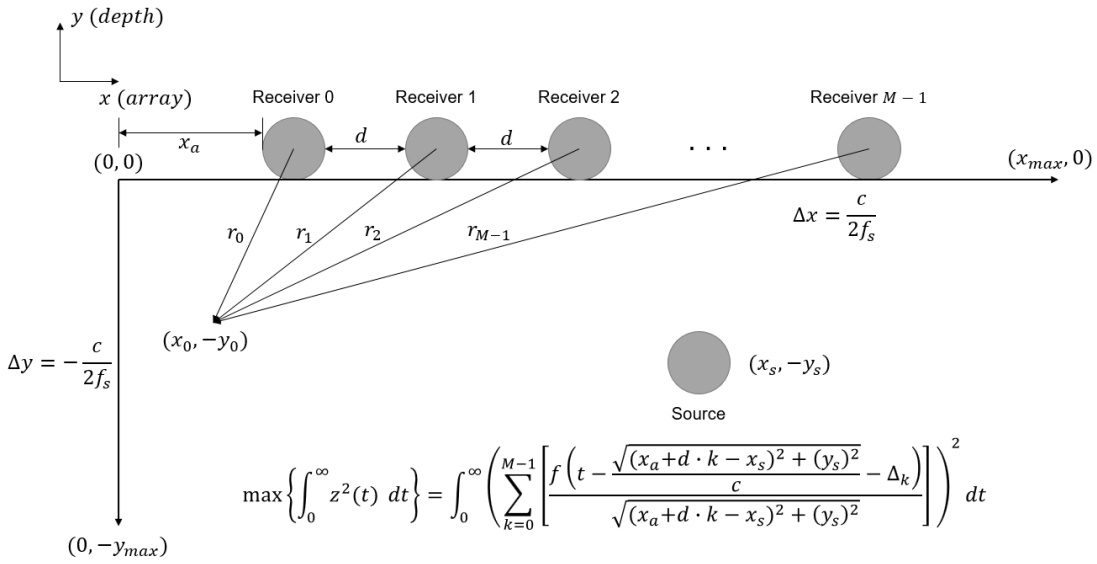


Figure 2: Backward Propagation Algorithm

Now, imagine a combination of these facts used on a linear array placed on the chest. The heart and linear array are assumed to lie on a plane, with that plane being the inside of the chest. Running our source estimation algorithm eliminates the doctor's necessity to adjust the location of the stethoscope head as the linear array finds the optimal location for recording heart acoustics via software. By producing a highly amplified signal, a clean heart acoustic signal can be recorded for extended periods of time allowing the doctor to continuously monitor the patient for potential heart problems that were not present during a physical examination. RAYO also solves other issues associated with current stethoscopes. It reduces the risk of misdiagnosis and reduces the risk of potential ear damage as all signal interpretation and recording is done electronically.

Our algorithm works great for continuous time signals. However, RAYO deals with discrete time signals. Therefore, adequate sample rate and compute power are necessary to produce a versatile stethoscope array. Our desired resolution and compute speed control the target specifications. Table 1 shows the base target specifications for RAYO.

Specification	**Requirement**
Spatial Resolution	±3.5cm
Storage	8 GB
Connectivity	WIFI, Bluetooth, SPI
Playback	.wav, .mp3, etc.
Display	5-inch or Greater Multi-Touch Display
Micro-Controller Core Count	2 cores @ 1 GHz
Micro-Controller RAM	1 GB
Input Signal SNR	>40dB
Number of Transceivers	>1

Table 1: Base Target Specifications

To achieve a spatial resolution of ±3.5cm, an adequate sampling rate is required. An equation used to find our spatial resolution and distance between receivers is derived using the model shown in Figure 3.

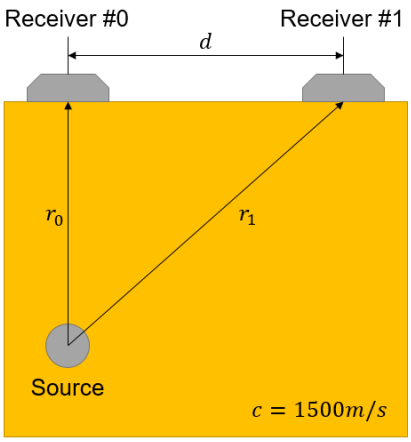


Figure 3: Spatial Resolution Model

From this model, we will assume that r_0 represents the expected depth of the heart within the chest (5cm). Also, the source will stay fixed beneath Receiver #0 and the position of Receiver #1 will be adjusted to find the required sampling rate. The time delay between Receiver #0 and Receiver #1 is,

$$\frac{r_1 - r_0}{c} = \Delta t.$$

Basic trigonometry can be used to relate the distance between the receivers and the time delay of the signal.

$$d = \sqrt{r_1^2 - r_0^2}$$

Our desired spatial resolution is 3.5cm which is related to $|r_1 - r_0|$,

$$3.5cm = |r_1 - r_0| = \Delta tc, \qquad \Delta t_{min} = 23.3\mu s.$$

Now that we have defined in minimum time step, this related to the sampling by means of the following equation.

$$\Delta t_{min} = \frac{1}{f_s}, \quad f_s = 42.86kHz$$

The minimum required sampling rate is $f_s = 42.86kHz$, and the distance between the receivers should at least be

$$r_1 = 8.5cm, \quad d = 6.874cm$$

for the minimum sampling rate. If we make the sampling rate $f_s = 100kHz$, the distance we need between the receivers reduces to,

$$d = \sqrt{\left(\frac{c}{f_s} + r_0\right)^2 - r_0^2} = 4.1533cm.$$

With a sampling rate of $f_s = 100kHz$, a source located 5cm below the chest/air boundary, and receiver separation of 6cm, the spatial resolution of RAYO is

$$\Delta r = \pm 1.5cm.$$

This resolution exceeds our desired spatial resolution. The Texas Instrument ADS1278 ADC was selected for its maximum sampling rate of $144kHz$ and its ability to operate with multiple receivers. The ADS1278 also uses SPI for easy integration with a Raspberry Pi or similar microcontrollers. To meet our storage, display, playback, micro-controller, and the rest of our connectivity specifications, the Raspberry Pi 4B will be used.

With the Raspberry Pi 4B we can run a display, interface with the ADC, quickly run the algorithm, record heart acoustics, and display a contour map of the approximate location of the heart. The expandable storage of the Raspberry Pi allows for microSD cards up to 2 TB in size. Therefore, storage is not an issue. Alongside the Raspberry Pi, a 7-inch touch screen display will be used to display the contour map and control the recording duration.

To receive signals from the chest/air barrier, a custom transceiver was designed to house a microphone such that it is as close to the boundary as possible while providing adequate shielding for reducing external noise recording. This allows for a high SNR. The transceiver head is shown below in Figure 4 and more details on each individual component is reported in the following sections. Table 2 shows the components that meet our target specifications.

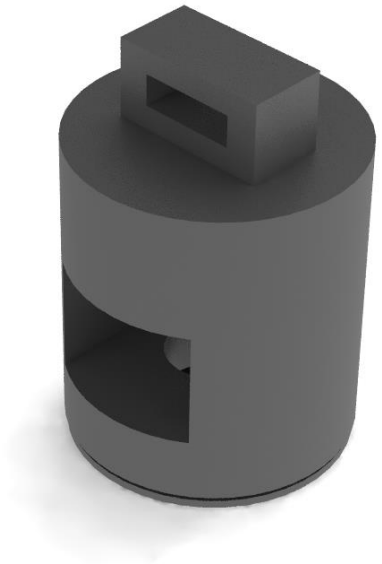


Figure 4: Transceiver Head

Component	Satisfied Specifications
TI ADS1278 ADC	Spatial Resolution, Multi-channel capability
Raspberry Pi 4B	Meets all storage, connectivity, playback, and computer capability specifications
7-inch Multi-touch Display	Meets all display/user control specifications
Transceiver Head	Allows for good signal coupling between chest/air boundary and allows for a high SNR

Table 2: Components that Satisfy Target Specifications

Testing Platform/Final Prototype Specifications

Figure 5 shows a rendered CAD drawing of our final testing platform to be shown at the Capstone Fair this Spring. For Spring Quarter's demonstration, a four-element linear array will be constructed using our custom transceiver heads, and our compute system comprising of the ADC, Raspberry Pi, and touch-screen display will run our recording/source localization algorithm. Gelatin has similar acoustic properties when compared to human tissue, therefore a speaker will be embedded at the center of our gelatin block at a depth of 12cm. Figure 6 shows the desired contour map that demonstrates RAYO's ability to estimate the location of the source. We will also playback the heart acoustic emitted from the speaker embedded within the gelatin block.

List of Desired Final Prototype Specifications

- Ability to find the optimal/near optimal time delays necessary to reconstruct the signal emitted from the source (heart).
- Compute a contour plot in a timely manner while recording heart acoustics in a separate process.
- Comprised of a linear array with 4 elements each separated by 6cm.
- Transceiver can couple well to the gelatin/air boundary and reproduce signals with a high SNR.
- Testing platform should be modular, with the ability to quickly adapt to new array designs.
- Compute system should be easy to work with and the ADC should be accessible for wiring changes.
- Display will show the contour map with the estimated source location.

All these goals will be met with the chosen components, and prior analysis.

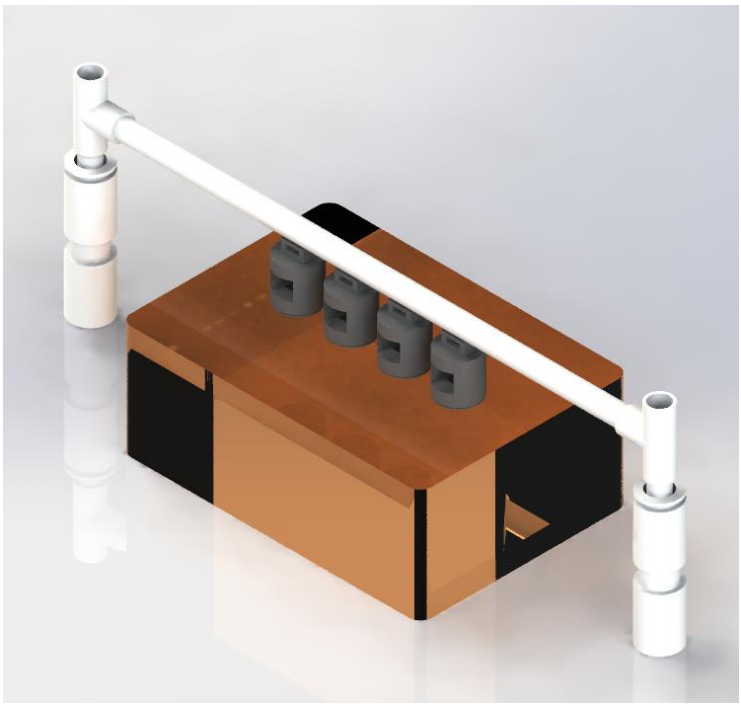


Figure 5: Final Prototype Rendering

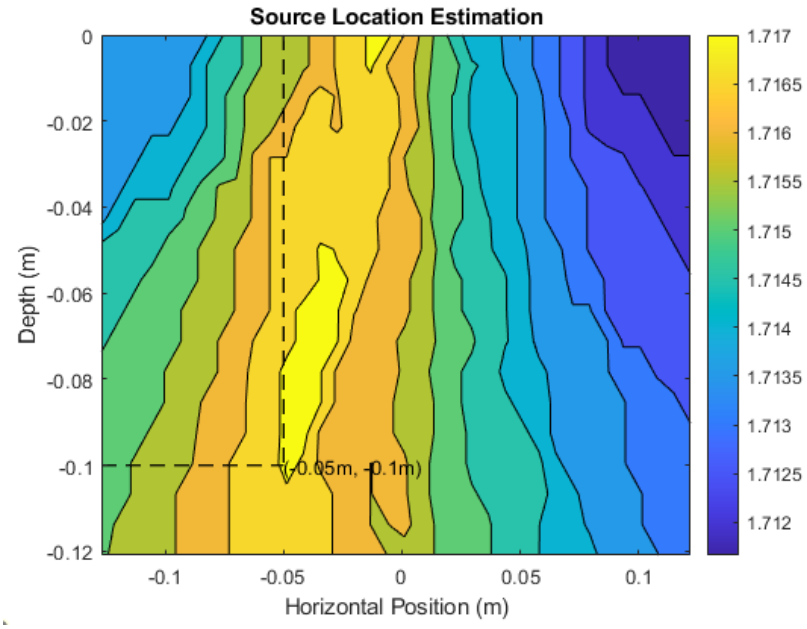
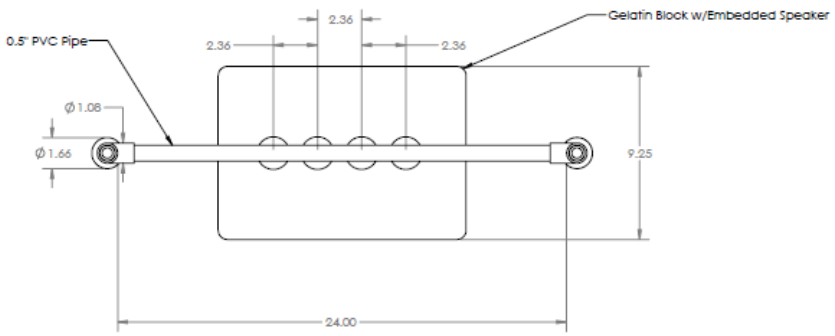
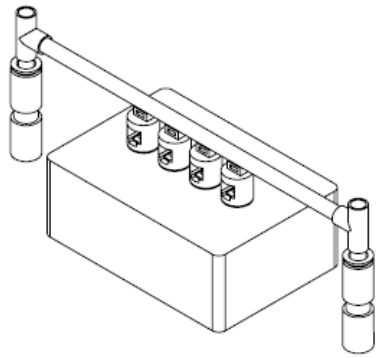
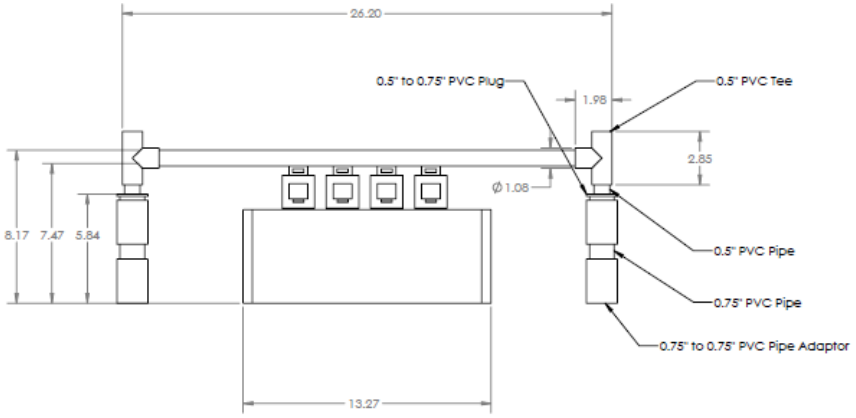
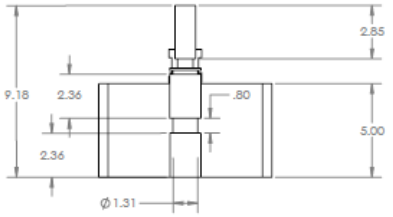


Figure 6: Localization Algorithm Contour Map









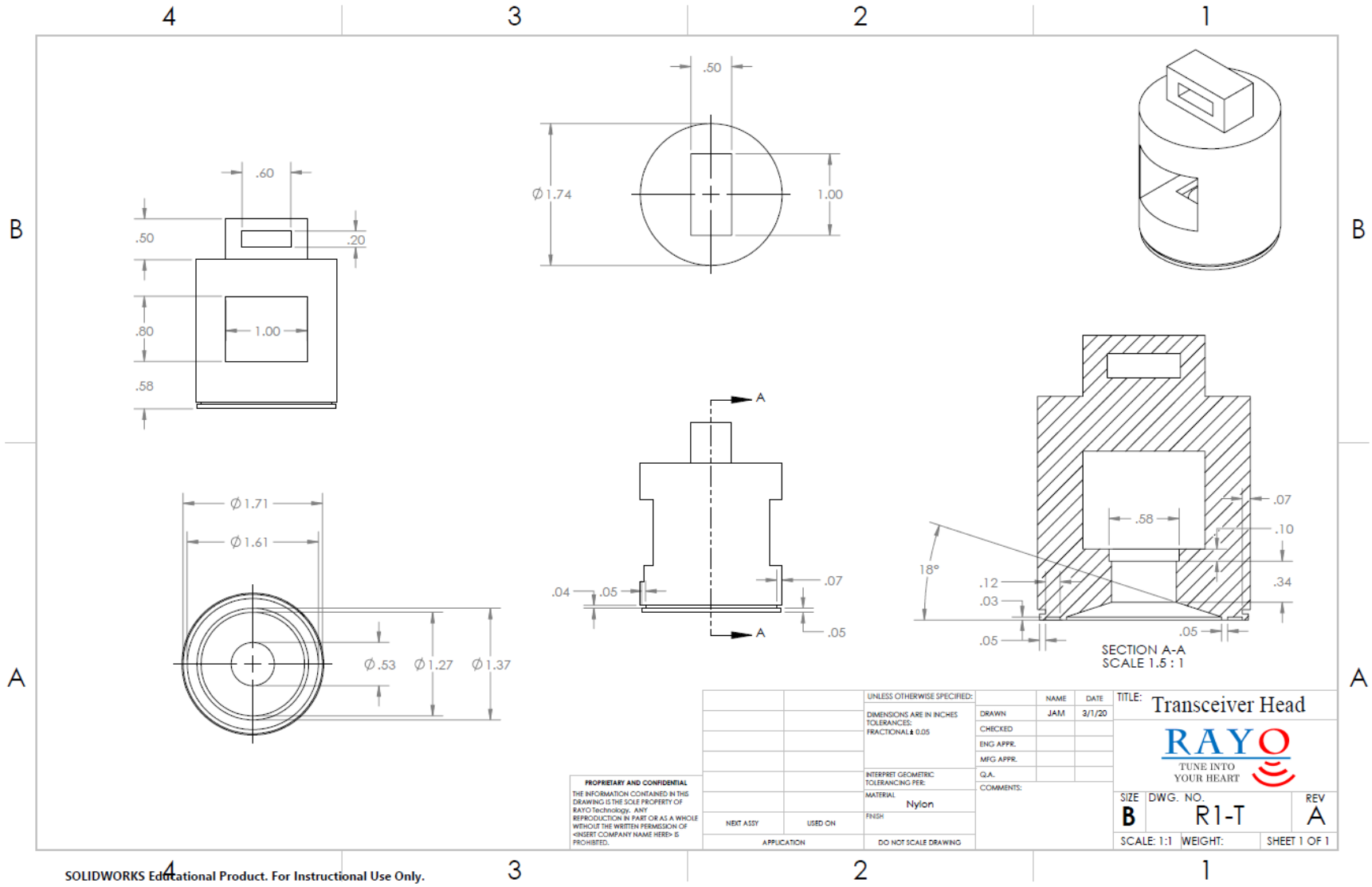
		UNLESS OTHERWISE SPECIFIED:		NAME	DATE
		DIMENSIONS ARE IN INCHES	DRAWN	JAM	3/3/20
		TOLERANCES: FRACTIONAL ± 0.1	CHECKED		
			ENG APPR.		
			MFG APPR.		
		INTERPRET GEOMETRIC TOLERANCING PER:	Q.A.		
		MATERIAL	COMMENTS:		
		FINISH			
NEXT ASSY	USED ON				
APPLICATION		DO NOT SCALE DRAWING			

TITLE: Test Platform Assembly

RAYO
TUNE INTO YOUR HEART

SIZE	DWG. NO.	REV
C	R1-TP	A

SCALE: 1:4 | WEIGHT: | SHEET 1 OF 1



SOLIDWORKS Educational Product. For Instructional Use Only.

System and Sub-System Block Diagrams

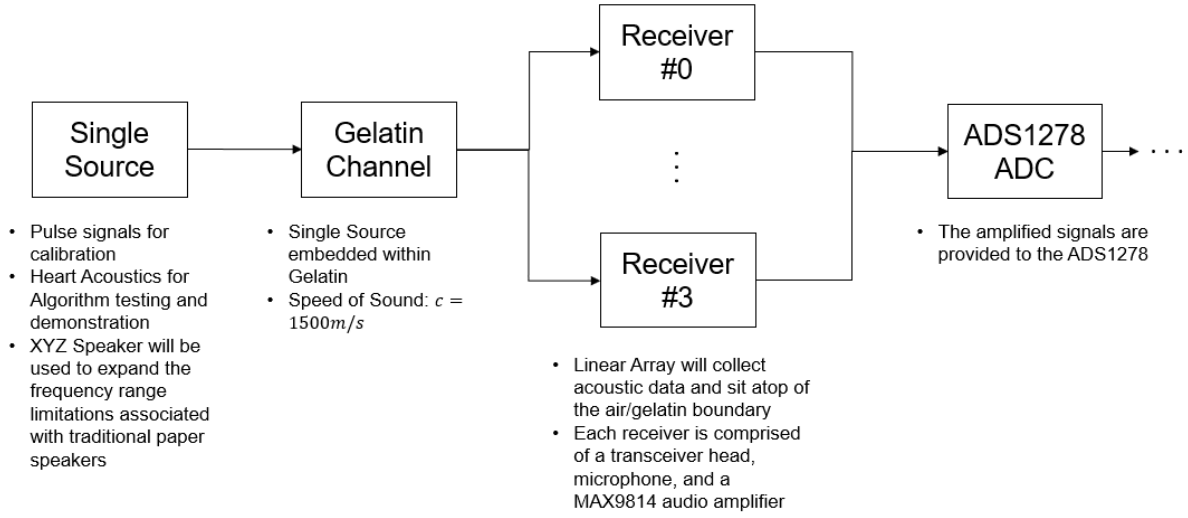


Figure 7: Source and Microphone Configuration

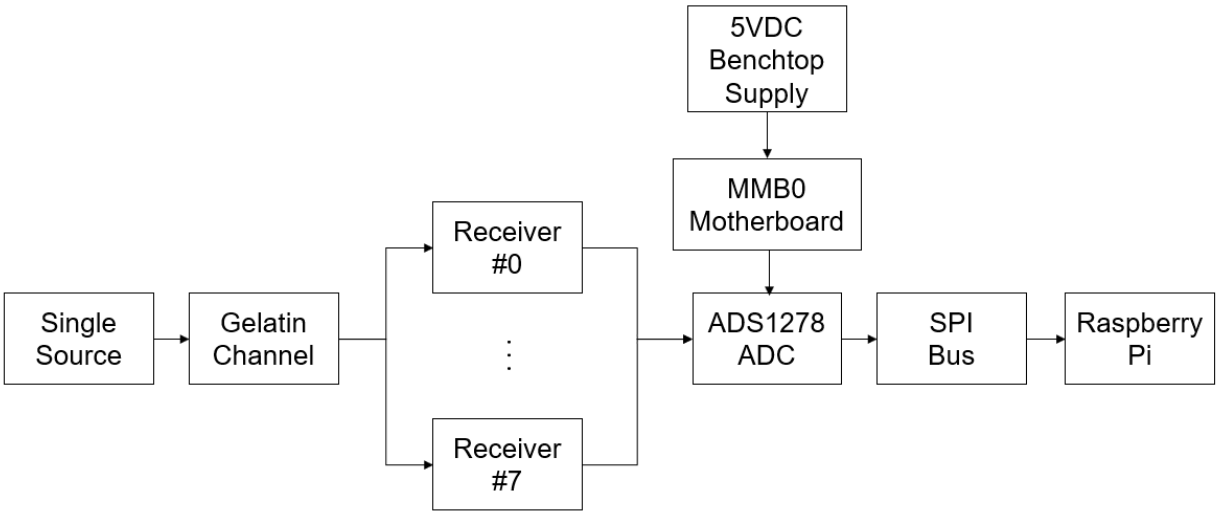


Figure 8: ADC and Raspberry Pi High-Level Block Diagram

Operation Flowcharts

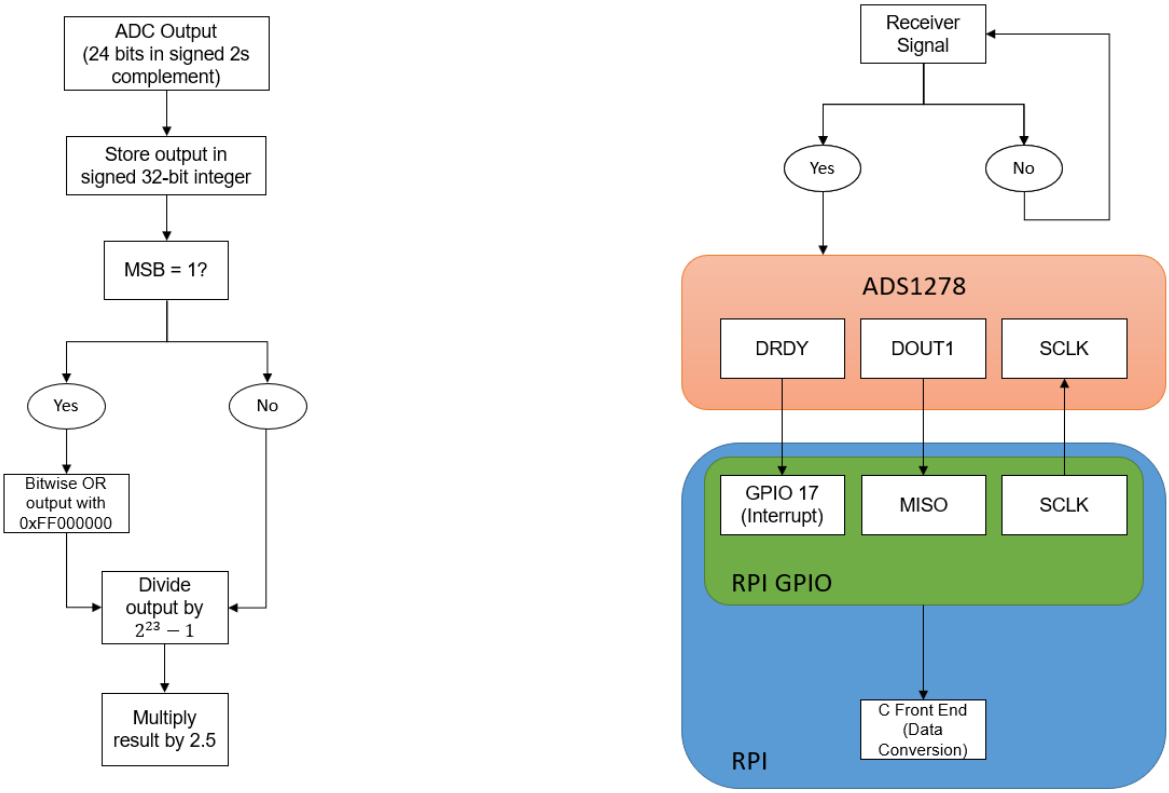


Figure 9: Data Conversion from ADC

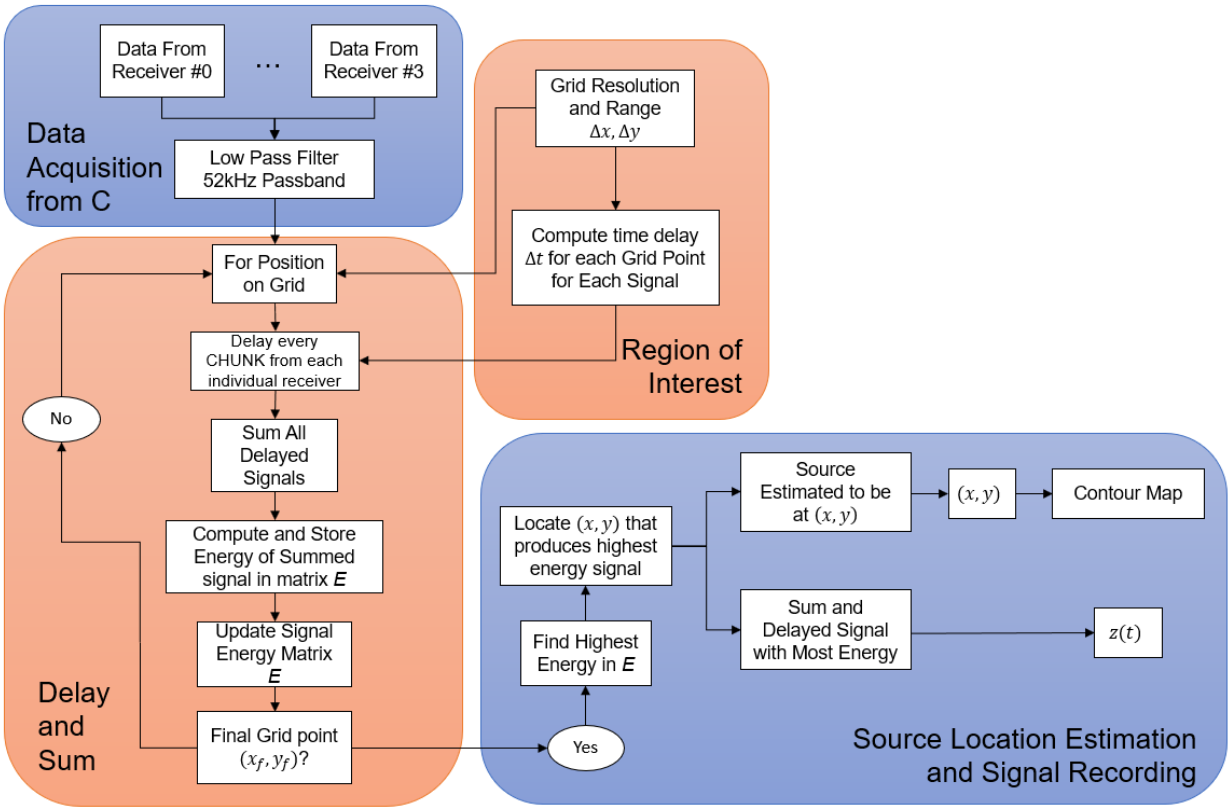
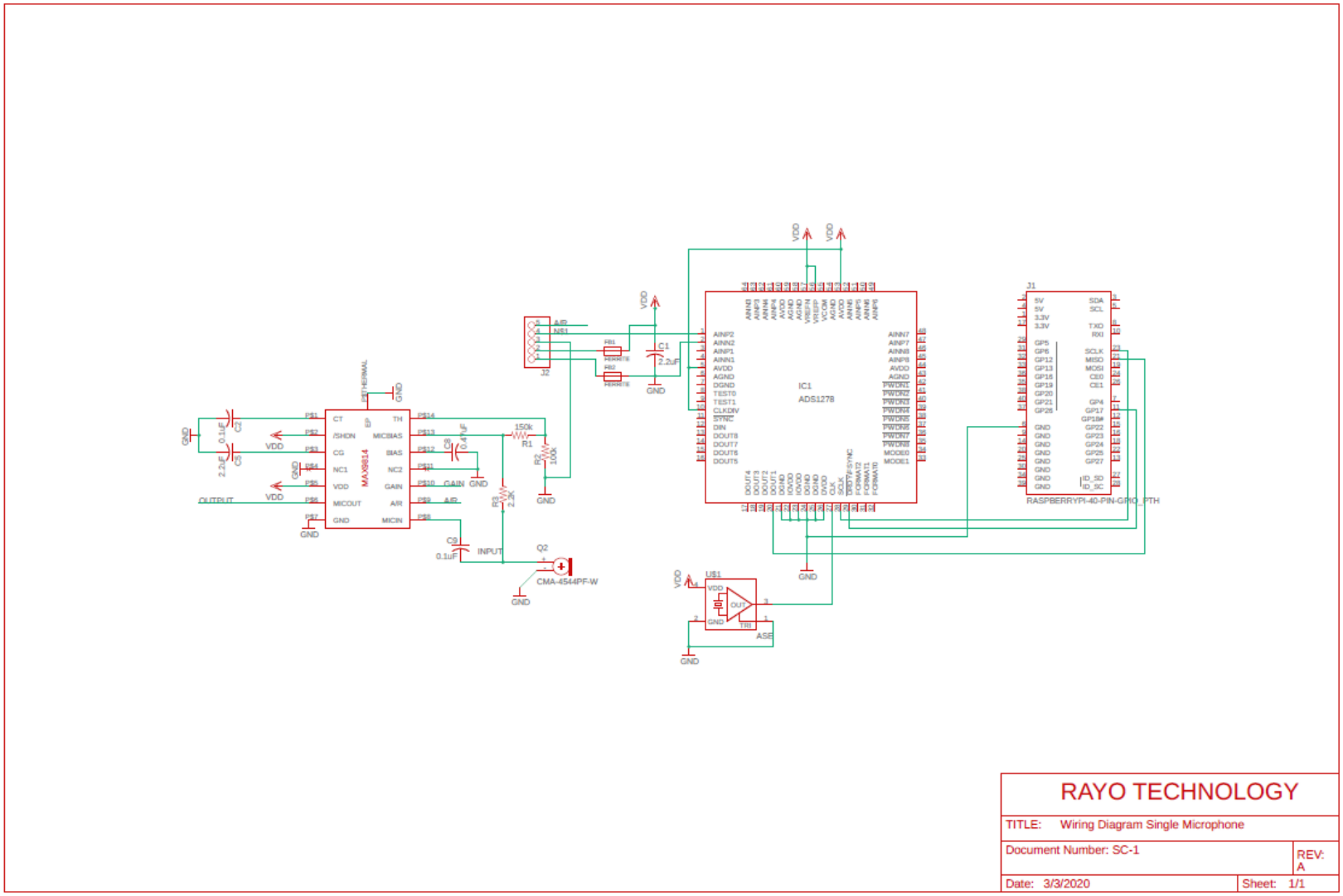


Figure 10: Source Estimation/Recording Algorithm

Circuit Schematic and Wiring Diagram



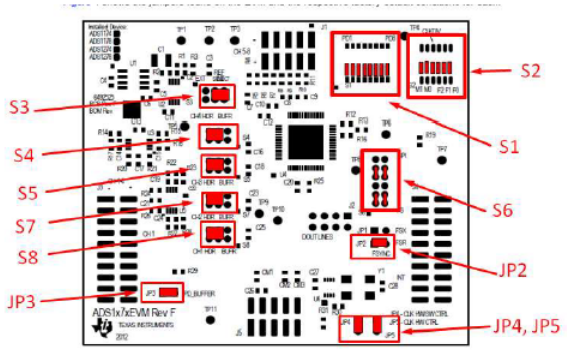
ADS1278 Eval Board

	J3	
Pin 1 -->	AINN4	AINP4
	AINN3	AINP3
	AINN2	AINP2
	AINN1	AINP1
MODE 1 (J4)	AGND	NC
FORMAT2 (J4)	AGND	AGND
FORMAT1 (J4)	AGND	NC
	NC	NC
	AGND	EXTREFN
	AGND	EXTREFP

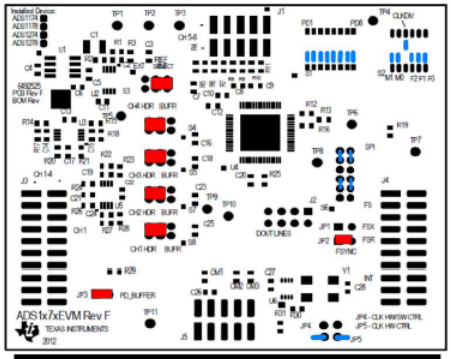
J2				
DOUT7	DOUT5	DOUT3	DOUT1	<-- Pin 1
DOUT8	DOUT6	DOUT4	DOUT2	

	J5				
	Not used	Not used	AGND	Not used	N/A
Pin 1 -->	Not used	+5VA	DGND	1.8VD	+3.3VD

	J4	
Pin 1 -->	SYNC	MODE0
	SCLK	DGND
	CLKR	MODE1
	DRDYSRC1	FORMAT0
	DRDYSRC2	DGND
	DIN	FORMAT1
	DOUT1	FORMAT2
	DRDY	SCL
	CLK	DGND
	CLK Sel	SDA



ADS1278-EVM-PDK Default Jumper Configuration



RAYO Configuration

	Raspberry Pi 4	
Pin 1 -->	3.3V	5V
	GPIO02	5V
	GPIO03	GND
	GPIO04	GPIO14 (TXD0, UART)
	GND	GPIO15 (RXD0, UART)
DRDY (J4	GPIO17	GPIO18 (PWM0)
	GPIO27	GND
	GPIO22	GPIO23
	3.3V	GPIO24
	GPIO10 (SPI0_MOSI)	GND
DOUT1 (J	GPIO09 (SPI0_MISO)	GPIO25
SCLK (J4	GPIO11 (SPI0_CLK)	GPIO08 (SPI0_CE0_N)
	GND	GPIO07 (SPI0_CE1_N)
	GPIO00 (SDA0_I2C)	GPIO01 (SCL0, I2C)
	GPIO05	GND
	GPIO06	GPIO12 (PWM0)
	GPIO13(PWM1)	GND
	GPIO19	GPIO16
	GPIO26	GPIO20
AGND (J3	GND	GPIO21

MMB0 Motherboard

J14					
-VA	+VA	+5VA	-5VA	+5VD	GND

BENCHTOP 5V AND GND; NOT THE SAME AS +5VA AND GND!

	J5				
	Not used	Not used	AGND	Not used	N/A
Pin 1 -->	Not used	+5VA	DGND	1.8VD	+3.3VD

RAYO
TUNE INTO YOUR HEART

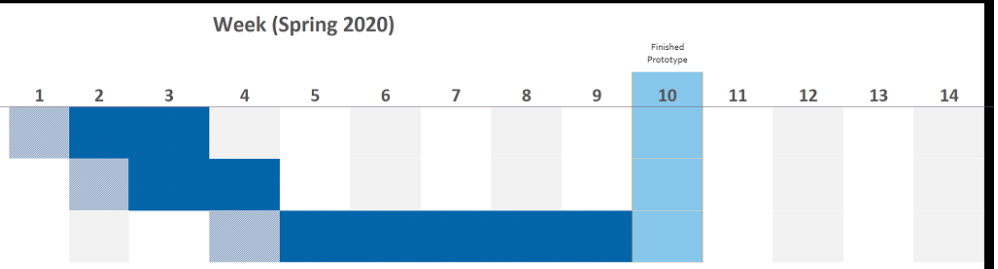
Bill of Materials/Budget

Item #	Name	Mfg	Mfg Part #	Distributor	Part #	Qty	Price	Description
1	Raspberry Pi 4B/2GB	Raspberry Pi	RASPBERRY PI 4B/2GB	Digikey	1690-RASPBERRYPI 4B/2GB-ND	1	$ 56.25	BCM2711 Raspberry Pi 4 Model B 2GB - ARM® Cortex®-A72 MPU Embedded Evaluation Board
2	ADS1278 ADC	Texas Instruments	ADS1278EVM-PDK	Digikey	296-30637-ND	2	$ 154.72	KIT EVAL/DEMO FOR ADS1278
3	UCTRONICS 7 Inch IPS Touch Screen for Raspberry Pi 4	UCTRONICS	B07VWDDWQ9	Amazon	N/A	1	$ 59.99	Display for system
4	EVAL BOARD FOR MAX9814	Adafruit	1713	Digikey	1528-1713-ND	9	$ 7.95	MAX9814 Microphone Amp Audio Evaluation Board
5	Transceiver Heads	Xometry	1D03B-15003	Xometry	N/A	7	$ 48.71	3D Print of Custom Transceiver Head
6	EXCITER 5W, 8 OHM, 44MM X 14.5MM	Soberton Inc.	E-4408	Digikey	433-1171-ND	2	$ 8.29	8 Ohms Exciter Speaker 5W 0Hz ~ 20kHz Top Round
7	3M Littmann Stethoscope Spare Parts Kit	3M Littmann	40022	Amazon	B014GVCGLE	7	$ 13.74	Transceiver Cover
8	3/4 in. PVC Pipe 2ft.	VPC	N/A	Home Depot	N/A	1	$ 1.52	PVC for testing platform
9	3/4 in. x 1/2 in. PVC Reducer	DURA	N/A	Home Depot	N/A	2	$ 0.83	PVC for testing platform
10	3/4 in. PVC Coupling	VPC	N/A	Home Depot	N/A	4	$ 0.35	PVC for testing platform
11	1/2 in. PVC Pipe 2ft.	VPC	N/A	Home Depot	N/A	2	$ 1.27	PVC for testing platform
12	1/2 in. PVC Tee	Charlotte Pipe	N/A	Home Depot	N/A	2	$ 0.45	PVC for testing platform
13	3/4 in. - 1-3/4in. Hose Clamp	Everbilt	N/A	Home Depot	6720595	7	$ 1.25	Hose Clamp for Transceiver mounting
14	64 Qt. Latching Storage Box	Sterilite	14978006	Home Depot	206721480	1	$ 7.98	Mold for Gelatin Medium
15	Unflavored Gelatin	Knox	N/A	Amazon	B001UOW7D8	6	$ 14.99	Gelatin for encasing speakers
Total Price:	$ 1,065.65							

Prototype Schedule Schedule (Spring 2020)

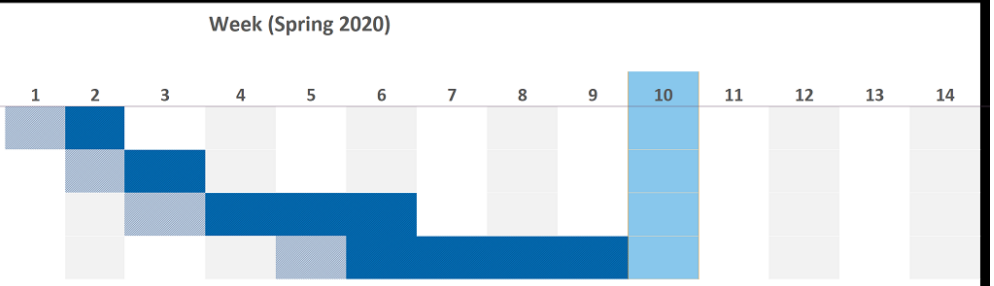
Gelatin and Speaker Team

	PLAN START	PLAN DURATION
Gelatin Speed of Sound Test	1	3
Heartbeat/Impulse/Step/Sine Input Tests	2	2
Final Prototype Setup (Gelatin + Speaker)	4	6



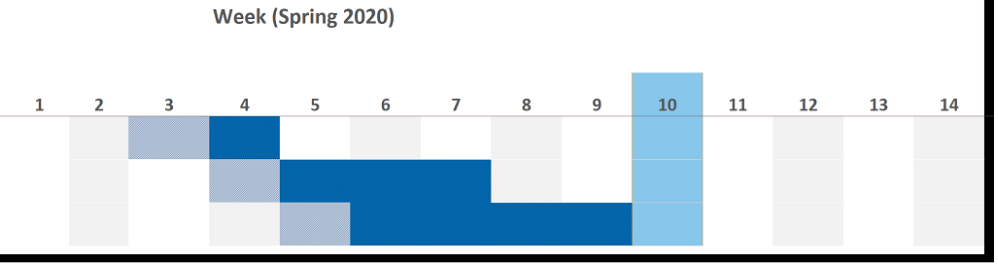
Transceiver/Array Design

	PLAN START	PLAN DURATION
Synchronization Test	1	2
Tranciever Comparison	2	2
Multichannel Tranciever Test w/ ADC	3	4
Final Prototype Array Setup and Tests	5	5



Algorithm Implementation

	PLAN START	PLAN DURATION
MATLAB to C++, w/ wrapper to C	3	2
Implement different beamforming techniques	4	4
Final Prototype Array Beamforming Tests	5	5



Prototypes and testing

ADC Functionality/Testing

The ADS1278 is not a typical SPI device because it is a read-only device and lacks a Slave Select/Chip Select (CS) pin. Instead of prompting data transaction via the CS pin, the device simply reads input signals and transfers data continuously. To collect data from the ADS1278, we rely on the ADC's DRDY output pin, which indicates data availability whenever a falling edge is observed on the signal. When the falling edge occurs, 24 bits of data will be available for collection. The data must be collected within 37ns, after which the next falling edge occurs, and the next 24 bits of data will overwrite any pre-existing data.

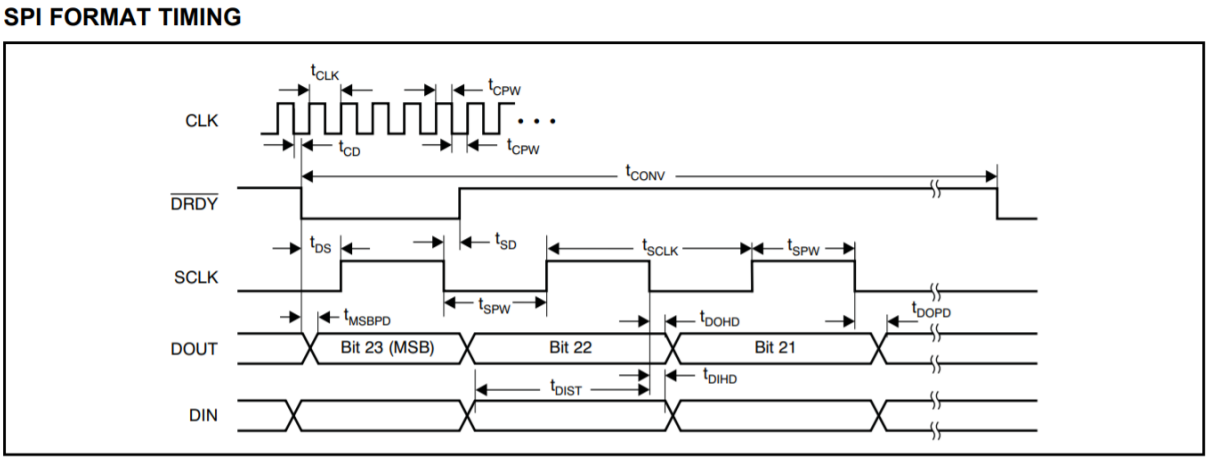


Figure 11: ADS1278 SPI Timing Format

For our purposes, we need the highest sampling rate the device can offer, which is 144,531 samples per second (SPS). However, the sampling rate of the ADS1278 is determined by the input clock we use. Since we are using the 27MHz from the MMB0 motherboard, our sampling rate is limited to 105,469 SPS. To enable this option, we configured the CLKDIV pin to HIGH to enable the selection of high-speed mode and both the MODE[1:0] pins to LOW to specifically select high-speed mode. We also tied the FORMAT[2:0] pins to LOW, which configures the device for SPI communication as well as data output via Time-Division Multiplexing (TDM) mode. This setting also causes the device to operate in Dynamic Mode, which eliminates any dead time from channels with no input signals. The reference voltage was chosen to be 2.5V (default value). This meant that to test the device with single-ended signals, +2.5V would have to be connected to the negative input of the ADS1278 and sinusoids would need to have a DC bias of +2.5V. This allows us to utilize the full-scale voltage range, as determined by the reference voltage.

FORMAT[2:0]	INTERFACE PROTOCOL	DOUT MODE	DATA POSITION
000	SPI	TDM	Dynamic

MODE[1:0]	MODE SELECTION	MAX f_{DATA} (1)
00	High-Speed	144,531

Figure 12: ADS1278 Format and Mode Selection

To easily test our design and to avoid the hassle of implementing PCBs, we utilized the ADS1278 evaluation board. The evaluation board kit includes a breakout board of the ADS1278 as well as a MMB0 motherboard, which provides analog and digital power rails as well as the +2.5V reference voltage to the ADS1278 and allows us to configure the settings via the ADCPro software. However, due to incompatibility issues, we were unable to use the software and instead configured the device by manually driving the pins. After successfully configuring the device, we connected it to the Raspberry Pi. C code was implemented to facilitate signal propagation from the input of the ADS1278 to the Pi.

For our tests so far, we've only been using one of eight channels on the ADS1278. A 2.5VDC sinusoid generated by a function generator is input to the ADS1278. If the communication between the ADS1278 and the Pi is established correctly, we should see the exact same signal (maybe with added noise) on the output. Indeed, we do receive output data that looks quite like the input. We believe some of the discrepancies between the input and output data sets are caused by the long wires and bad signal connections. This will be improved upon in future iterations of the test setup.

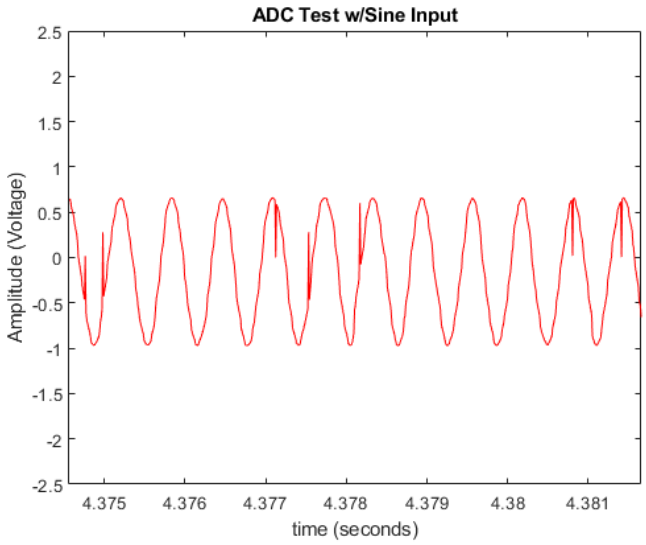


Figure 13: ADC Test Output

Speed of Sound/Synchronization Test

To demonstrate our ability to synchronize the source and microphone receivers, a speed of sound test was conducted in air. With this experiment, we also record data on which waveforms optimize our ability to measure the speed of sound in different medium (gelatin). Figure 14 shows our testing platform for the speed of sound test. A Digilent Analog Discovery 2 development kit was used as a function generator and oscilloscope. To conduct the experiment a pulsed waveform is emitted from the source. The source is aligned to the two-element array such that only the distance between the two elements is necessary to estimate the speed of sound. The pulsed signal is recorded on the microphones and the time between them is synchronized to minimize timing errors.

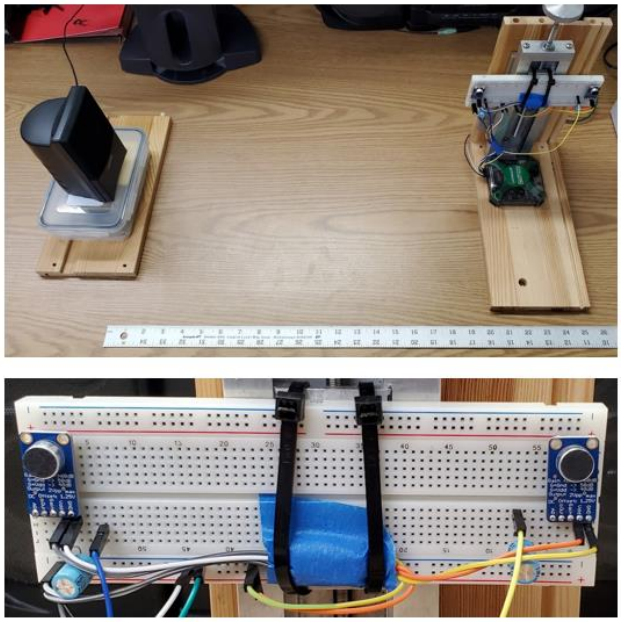


Figure 14: Speed of Sound Test Setup

The recorded signals are filtered to increase the SNR of the signals. Then, the received signals are cross correlated to reveal the time delay between the two received signals. To compute the speed of sound, the distance between the two elements is divided by the estimated time delay. Figure 15 shows the block diagram of the algorithm used to estimate the speed of sound. Figure 16 shows the resulting speed, correlation, and time/frequency domain representations of the received signals.

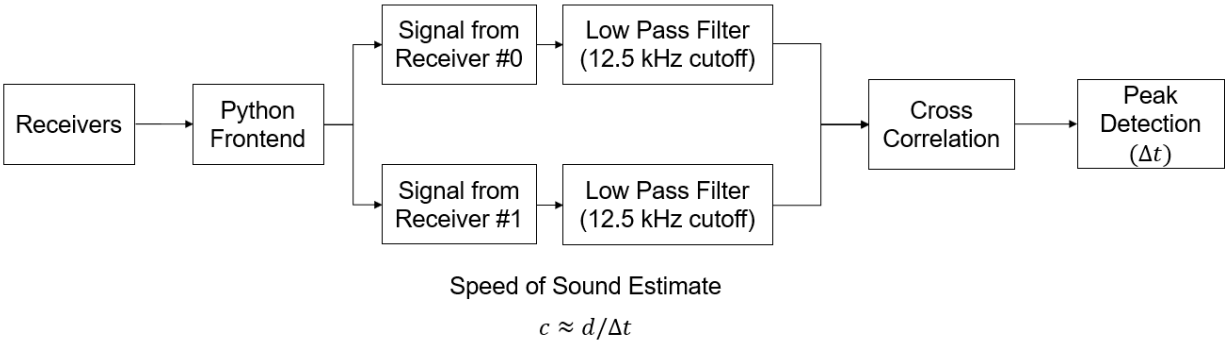


Figure 15: Speed of Sound Test Block Diagram

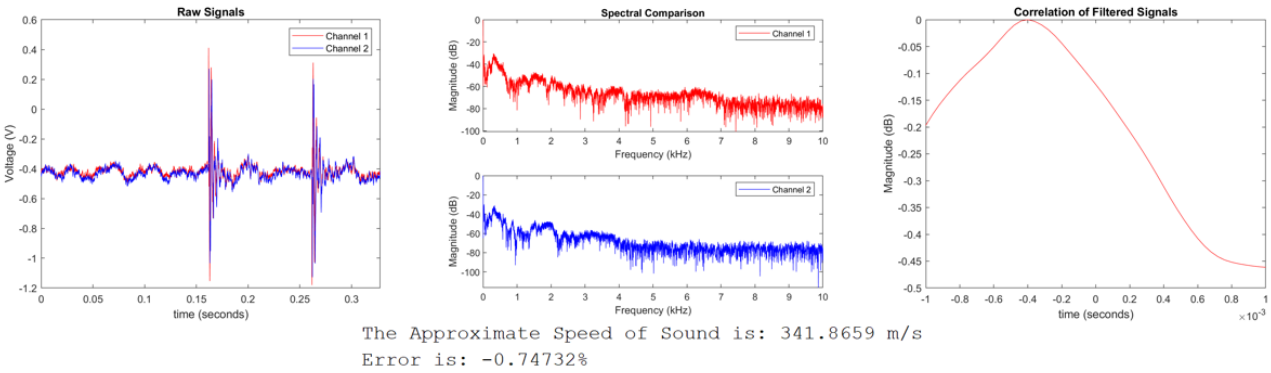


Figure 16: Resulting Speed and Signal Information

From this experiment, we were able to successfully demonstrate our ability to synchronize the source and receivers. The waveforms and algorithm studied in this experiment will pay dividends when conducting this same experiment with gelatin next quarter.

Gelatin Test

The propagation of sound through human tissue is like the propagation through polymer gelatin. Average speed of propagation in body tissue is around 1540m/s. At room temperature, the speed of sound through gel can range between 1460m/s to 1650 m/s. Gelatin serves as a controllable variable as the speed can vary between different body types. In addition, primary gelatin trials will confirm the validity of gelatin as a test platform before actual testing on people. If successful, the proceeding gelatin tests also allow for better results when confirming the validity of the beamforming techniques.

Using these considerations, the gelatin was tested to ensure good playback ability and a large signal to noise ratio (SNR) when emitting and receiving sound. In the experiment, a speaker was embedded at the bottom of a small block of gelatin at room temperature and the 1kHz tone emitted from the speaker was recorded through the microphone transceiver. A foam cup served as the mold and the solution was placed within a fridge overnight. The mixture ratio of gelatin and water was recorded (9:1). A function generator provided a 1kHz sinusoid input into the speaker as the transceiver was placed directly on top of the block.

Figure 17: Gelatin Block w/Embedded Speaker

From the collection of data found from the microphone, it was concluded that the signal received contained minimal noise and high energy. Gelatin was successful in producing a high SNR from our trials. This confirms the original credibility of gelatin as a medium for sound transmission. In the future, the speed of sound in gelatin will be tested to confirm it being an effective substitute to the human body.

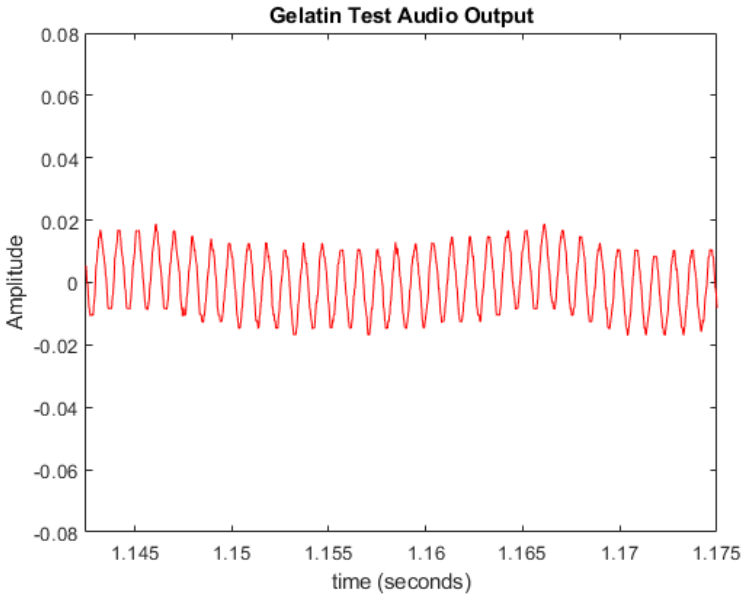


Figure 18: Audio Signal from Transceiver

Grid Resolution Test

To compute and test the computational load of the source localization algorithm, three different grid sizes were tested. The nominal case for step size is the spatial resolution of our system, 1.5cm. A grid step size of 3cm and 0.75cm will be used to compare the computational load and compare the resulting contour maps with the true location of the source. Table 3 shows the computational load in terms of the nominal case.

Grid Step Size	Computational Load
3cm	ε/4
1.5cm	ε
0.75cm	4ε

Table 3: Grid Size V. Computational Load

The following Figures show the resulting contour map for a source located at a depth of 10cm with a grid area of 264cm squared. The test involves two transceivers separated by 6cm.

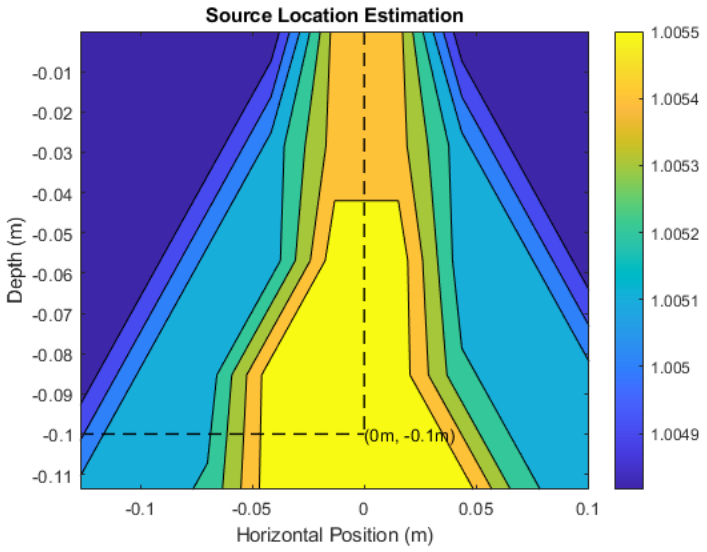


Figure 19: 3cm Grid Step

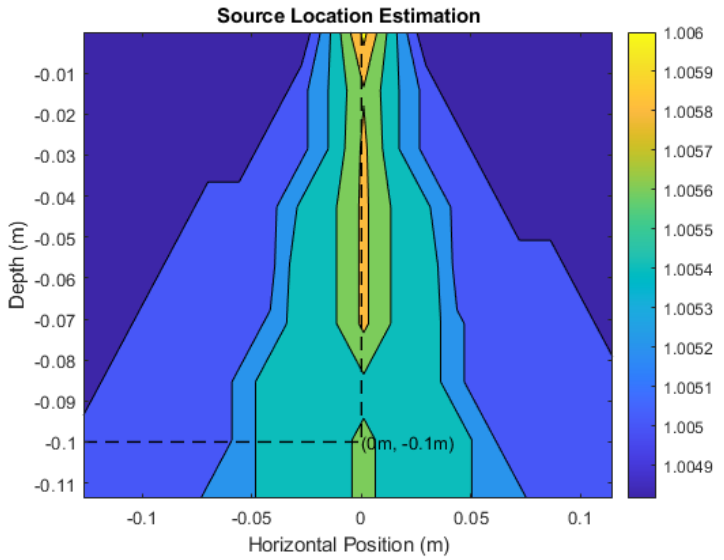


Figure 20: 1.5cm Grid Step

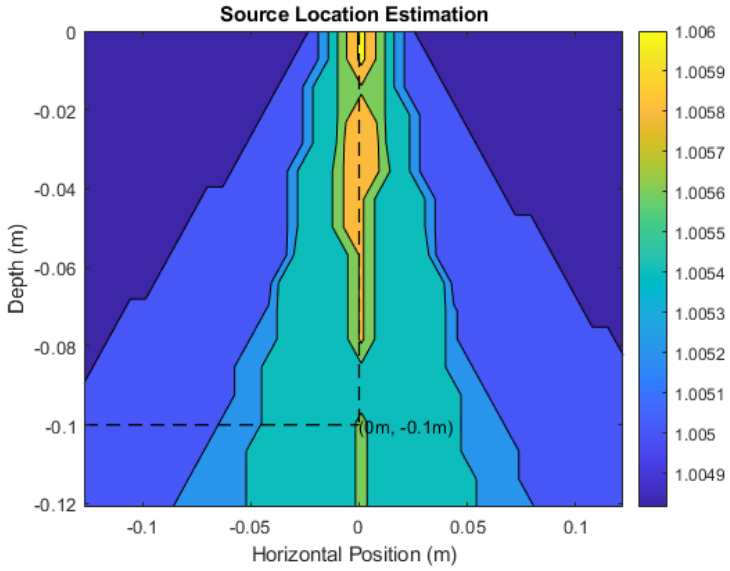


Figure 21: 0.75 Grid Step

From the resulting contour maps, having a coarse grid step will still reveal an accurate bearing angle estimation and source location estimation. Therefore, we will be using a coarse step size of 3cm for all testing and demo implementation.

Future Prototypes

Multi-Channel ADC Implementation

Our goal for the ADS1278 is to be able to read signals from all eight inputs. By configuring the ADS1278 to operate in TDM mode, we can retrieve data from multiple channels from just a single output pin, DOUT1. Thanks to this feature, we only must tune our code to correctly separate and interpret the data. To test our setup, we will input different pure sinusoids to each of the eight inputs. If our implementation is correct, we should expect the exact same signals on the output.

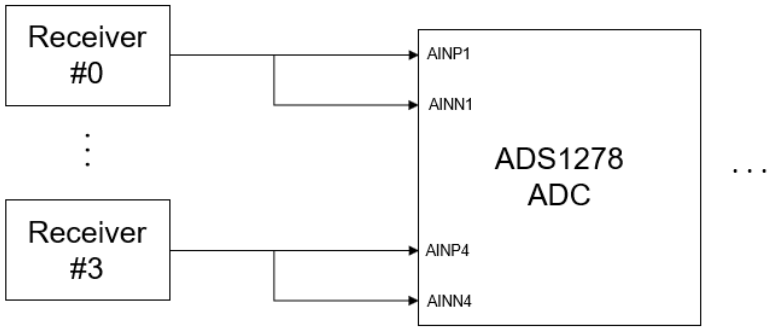


Figure 22: Multi-Channel Block Diagram

Fast Algorithm Implementation

To insure the fastest possible update to our contour map, our MATLAB code used to test our algorithm will be converted to C++. Once in C++, the code will run faster as MATLAB is higher level and requires more complex compliers when compared to C++. Implementation in C++ will also allow the code to interface directly with the C code used to capture the data from the ADC.

C code will be generated to simulate the code used to interpret the SPI channel received from the ADC. Once that code is created, we will implement our algorithm in C++ which will plot a contour map and record data in the background once the ideal time delays are computed using our algorithm. The multiple cores on the Raspberry Pi 4B allow us to run parallel processes allowing us to record and run our algorithm at the same time. Another grid step test will be performed using the C++ code.

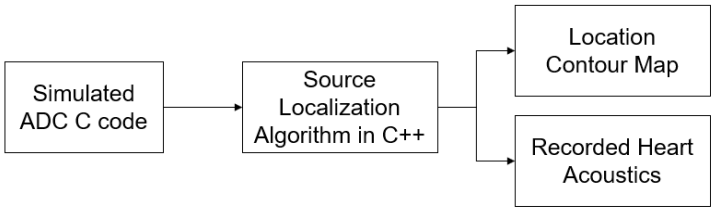


Figure 23: Fast Algorithm Implementation Test

Both prototypes will be used to help us achieve our goals for RAYO, the assembly drawing and the final prototype section of the design package give more detail on the final product to be shown at the Capstone Fair.

Breadboarding of the Components

To ease debugging, our components will be breadboarded together in order to ensure the shortest possible connection between the components. This will remove any potential unwanted inductances and other unwanted parasitics that attribute to the noise of the system.

As mentioned before, for Capstone we are not using PCB design for developing the system. Our goal is to demonstrate the capabilities of beamforming stethoscope arrays. Therefore, in future developments of RAYO (after Capstone), a PCB design can ease integration into a smaller package. We are constructing a testing platform in our first year of development. Figure 24 shows the breadboarding layout.

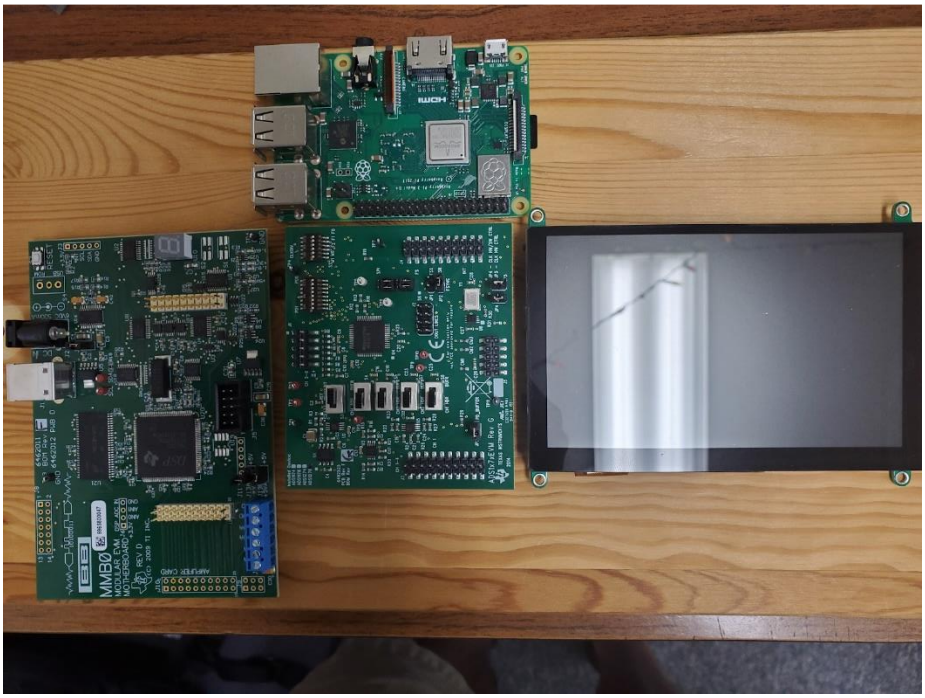


Figure 24: Potential Breadboarding Layout