Heart and Lung Sound-sensing shirt

Team 81 Members: Abhiyash Nibber, Hesham Al Ramahi, Marc Backas

TA: Hershel Rege

ECE 445, February 9, 2019

Introduction

Objective

According to the World Bank and WHO, half the world lacks access to essential health services and 100 million are pushed into extreme poverty because of health expenses. There are many factors for this such as the lack of good doctors and the poor quality of medical diagnostic devices. Often, this causes people to travel large distances to talk to a doctor, which can be hard at times. This inspired us to take on a senior design project that may potentially solve this problem. Our goal is to design and build a shirt that is capable of detecting heart and lung sounds which a user can access on his smartphone and then send to a professional doctor.

Background

Many people may be stressed, sometimes even afraid to go to the doctors, even if it's just for a checkup. In many countries, people do not have access to quality healthcare and often have to travel large distances meet a doctor. Even after investing time and money, they might not be able to get good quality treatment. In this case, many people would benefit if they had were able to consult a doctor without physically going there. Therefore, this project will provide convenience for many people and cause them less stress when the need to see a doctor arises. In the market, there are shirts that can detect a user's breathing and heart rate. Other products like Littmann Electronic stethoscopes offer good biological sound detection but do no capture the sounds from different areas in the body at the same time like our device. These stethoscopes are also significantly expensive than standard auscultation devices as well as our product. This shirt will detect the sounds of both the heart as well as lung, and by interfacing the SD card with a microcontroller, the sounds can be accessed by the user on a smartphone and sent to a doctor for analysis.

High-level requirements

- The digital microphones must be able to record heart sounds between 60 - 250 Hz and lung sounds between 50 - 2500 Hz with a Signal-to-noise ratio of 20 dB.
- The 24-bit, 16 kHz WAV files* generated should contain noise-free audio without aliasing.
- The doctor should be able diagnose the patient's medical condition in atleast 69% of cases [8], which will make it as good as high-end stethoscopes.

*The WAV files can be easily upsampled to 48 kHz for better audio quality

Design

Block Diagram

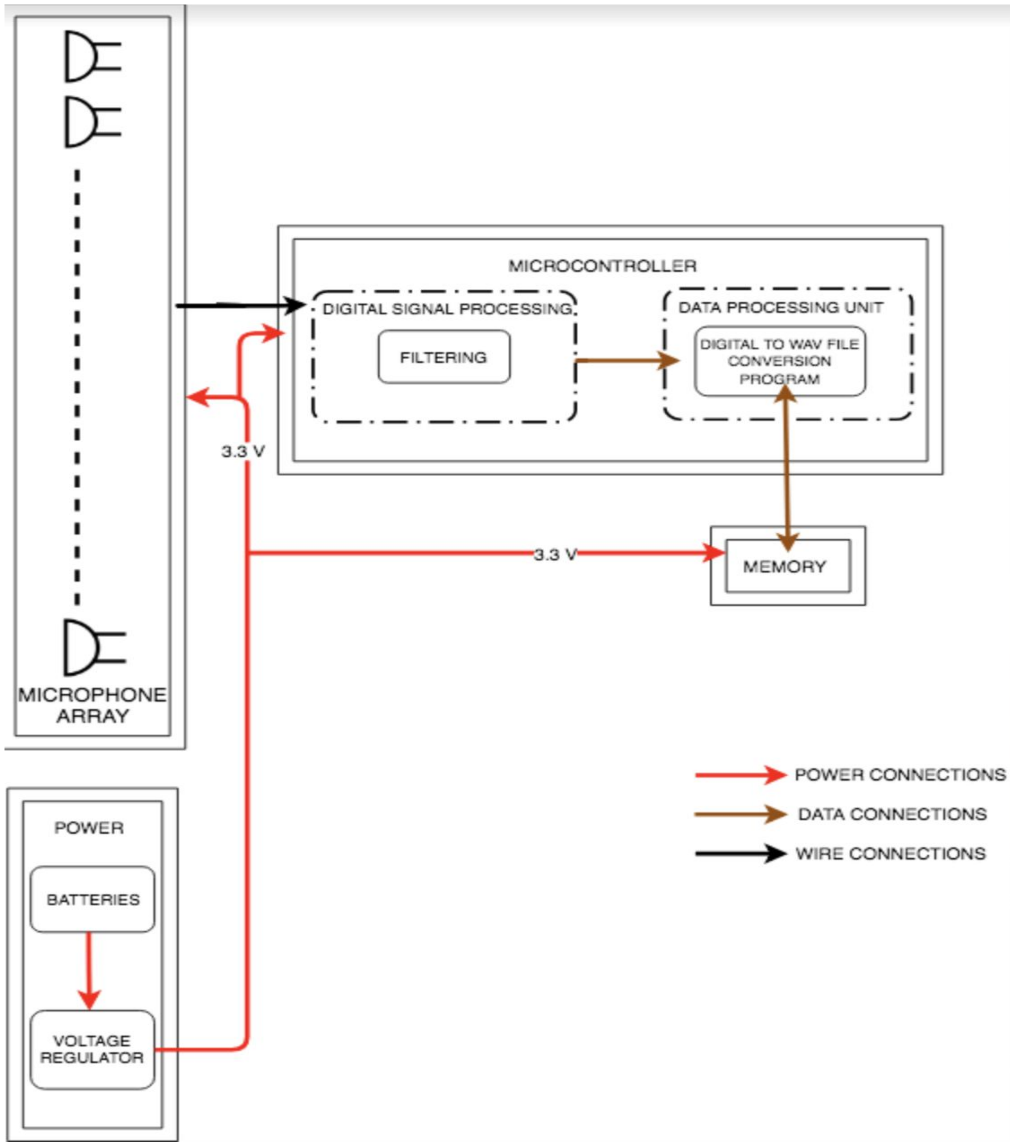

Figure 1: Block Diagram

The Block diagram in the above figure demonstrates how different units in the system interact with each other to fulfill the high-level requirements of this project. The 6 microphones will be placed at different positions on the shirt. The raw audio sample from 4 of these microphones will be filtered and converted to a WAV file by a microcontroller. The microcontroller is interfaced with an SD card that stores the WAV files. These files can then be accessed by the user on their smartphone and can be sent to a doctor for analysis.

Physical Design

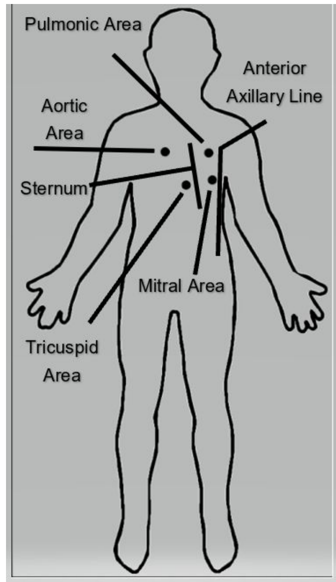

Figure 2: Physical diagram

The figure above demonstrates the locations of four of the digital microphones on the chest and abdominal area of the shirt to detect the heart as well as lung sounds. Another two will be placed placed near Mitral Area and Pulmonic Area to capture ambient noise. The shirt would consist of two layers. The microphones, power circuit and microcontroller will be placed between the inner and the outer layer to isolate the patient's skin from the electronics. The microphones will be soldered to an Evaluation board as shown in Figure 3 and daisy-chained with each other as shown in Figure 6. The microphone heads will be placed in 3-D printed headers which will make contact with the patient's skin to capture signals and the shirt will be snug enough to keep the microphones in place.

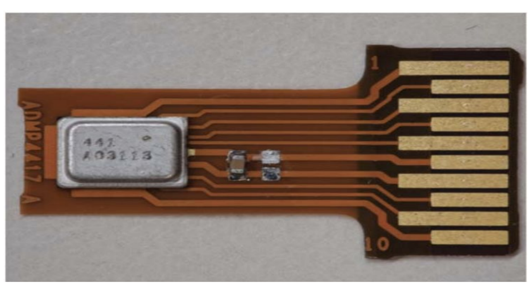

Figure 3: Top view of Evaluation board [8]

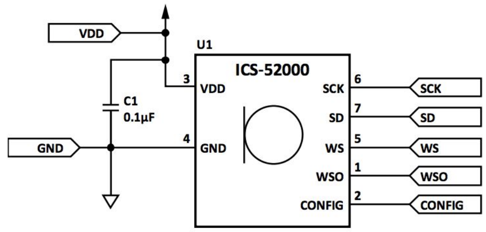

Figure 4: Evaluation board schematic [8]

Functional Overview

Microphone Array

Microphones are used in our system to sense the heart and lung sounds. 6 TDM MEMS microphones will be placed in an array on the chest as well as upper back region of the shirt and interfaced with a microcontroller as shown in Figure 6. All microphones in the array sample their acoustic signals synchronously, enabling precise array processing. Statistical analysis showed that the major concentration of energy, for both first heart sound (S1) and second heart sound (S2), is below 150 Hz which may indicate that both sounds are caused by vibrations within the same structure, possibly the entire heart. However, S2 spectra have greater amplitude than S1 spectra above 150 Hz, which may be due to vibrations within the aorta and pulmonary artery. In subjects with healthy lungs, the frequency range of the vesicular breathing sounds extends to 1000 Hz, whereas the majority of the power within this range is found between 60 Hz and 600 Hz. Other sounds, such as wheezing or stridor, can sometimes appear at frequencies above 2000 Hz. Based on this data, the microphones should be able to detect sounds within the frequency range 50 Hz - 2500 Hz. Two of the microphones will be used to capture ambient noise. The digital output from the microphones will be sent to the microcontroller for filtering, noise cancelling and data processing. The following block diagram in Figure 5 was taken from the datasheet for Invensense ICS-52000 [6].

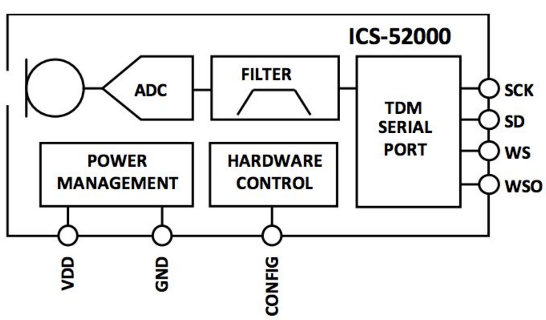

Figure 5: ICS-52000 Block Diagram [6]

Pin Name	Function
WSO	WS output, connected to the the WS of the next ICS-52000 in the array
CONFIG	Pulled to VDD. The state of this pin is used at power-up
GND	Connected to ground
VDD	Power, 3.3 V. This pin will be decoupled to GND with a 0.1 µF capacitor
WS	Serial Data-Word Select for TDM Interface
SCK	Serial Data Clock for TDM Interface
SD	Serial Data Output for TDM Interface. This pin tri-states when not actively driving the appropriate output channel. The SD trace will have a 100 kΩ pulldown resistor to discharge the line during the time that all microphones on the bus have tri-stated their outputs.

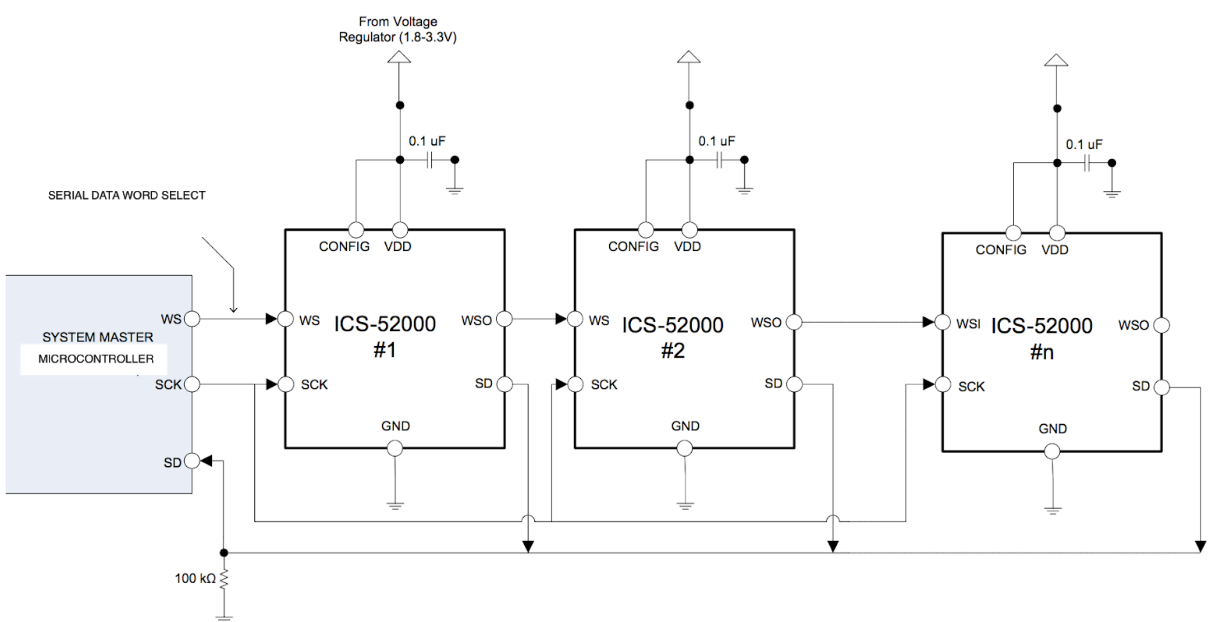

Figure 6: The Microphone array interfaced with the Microcontroller

Requirements	Verification
● Have a Frequency response in the range 50 Hz - 20 kHz. ● Operate on 3.3 V +/- 0.1 V.	● A stepped sine sweep is played from 50 Hz to 20 kHz through a source and the raw audio samples can be recorded using the microcontroller with a sampling rate of 48 kHz. The Frequency response can be plotted from the obtained data. ● Power the microphones with voltage in the range 3.2 V - 3.4 V.

Power Unit

The Power unit consists of a single power line which supplies 3.3 V in order to power the microcontroller, microphones and the SD card. We have a 5 V battery and a voltage regulator with output voltage 3.3 V to ensure there are no voltage spikes which can damage the components of our system. The total power drawn from this unit is calculated by adding the power consumed by the microcontroller and the 6 microphones.

$$P_{\text{microphone}} = P_{\text{microphone, active}}$$

$$= (1.5 \text{ mA}) \times (3.3 \text{ V})$$

$$= 4.95 \text{ mW}$$

$$P_{\text{microcontroller}} = P_{\text{microcontroller,active worst case}}$$

$$= (90 \text{ mA}) \times (3.3 \text{ V})$$

$$= 0.297 \text{ W}$$

$$P_{\text{total}} = P_{\text{microphone}} + P_{\text{microcontroller}}$$

$$= 0.297 \text{ W} + 4.95 \text{ mW}$$

$$= 311.95 \text{ mW}$$

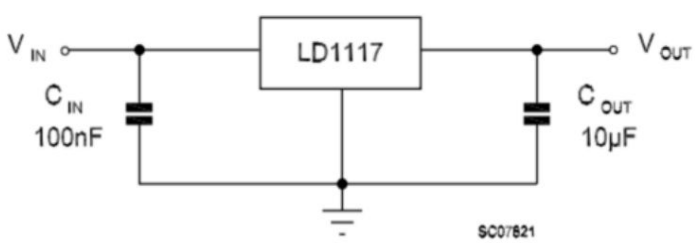

Figure 7: Power supply circuit [4]

Requirements	Verification
• Generate 3.3 V +/- 0.1 V. • Can operate at currents 0-100 mA. • Batteries provide 1000 mAh of power.	• Measure the output voltage from the voltage regulator and ensure that it stays within the 3.2 - 3.4 V range. • Use a constant current circuit to draw 100 mA from the power supply and voltage regulator. • Ensure that the batteries run for 10 hours at maximum current (100mA).

Signal Processing Unit

The microcontroller generates an 16 kHz WS signal [6] and a 4096 kHz SCK signal [6] required to retrieve audio samples synchronously from the microphones. The Digital Signal Processing Unit for filters out the ambient noise. For the heart, the frequencies of the sounds we want to detect in lie in the range 60 - 250 Hz and for the lungs, the signals of interest are in the range 50 - 2500 Hz. This can be achieved by implementing an adaptive Digital Bandpass filter with cut-off frequencies between 50 and 2500 Hz using the microcontroller. The outputs from 4 MEMS microphones and the ambient noise captured from 2 microphones are fed into an adaptive filtering unit in the microcontroller. The signals of interest are compared with the noise and the ambient noise is subtracted. Figure 8 shows an adaptive filter, which we will be implemented using Least Mean Square algorithm (LMS), to get a better estimate of the signal by changing the value of the filter coefficients. LMS starts by filtering the reference input using weights (w) of the adaptive filter and creates an estimate of the primary input. It then creates an error signal using the equation,

$$e(n) = d(n) + r(n) - y(n) \tag{1}$$

where e(n) is the error estimation, d(n) + r(n) is the output from the microphone contaminated with noise and y(n) is the output of the filter. The error signal is then sent into the adaptive filter to update the weights (w) and increasingly make the algorithm more efficient. The weights are updated every using the equation,

$$w(n+1) = w(n)*c*e(n)*r'(n)$$

where the w(n+1) is the new updated weight vector of the filter, w(n) is the current weight vector, c is the step size, r'(n) is the estimate of the instantaneous amplitude of the ambient noise, e(n) is as defined in (1). The weights are updated every iteration.

We finally get an mth order filter as

$$y(n) = \sum_{k=0}^{m-1} w(n) * r'(n-k) \tag{3}$$

where w(n) is the adjustable weight, r'(n) is as defined in (2) and y(n) is as defined in (1).

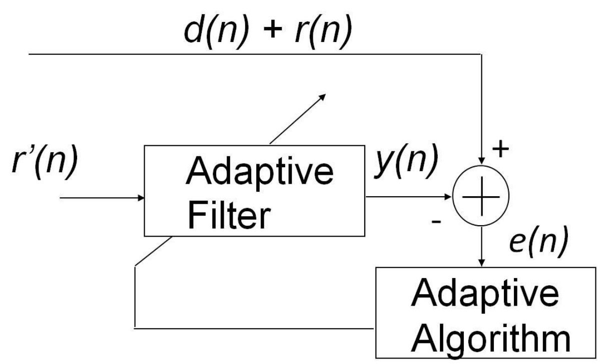

Figure 8: Block diagram of adaptive filter [2]

Requirements	Verification
• -3 dB Frequency Response below 60 Hz and above 250 Hz for filtering heart sounds. • -3 dB Frequency Response below 50 Hz and above 2500 Hz for filtering lung sounds. • Microcontroller should generate 16 kHz WS signal [6]. • Microcontroller should generate 4096 kHz SCK signal [6] • Startup time to valid data is 256 ms [6].	• Generate stepped sine sweep* from 0-60 Hz and record the audio samples using the microcontroller. Measure Frequency response to verify it is -3 dB below. Repeat the process for signals 250 Hz and above by generating sine sweep from 250 to 2500 Hz. • Generate stepped sine sweep* from 0-50 Hz and record the audio samples using the microcontroller. Measure Frequency response to verify it is -3 dB below. Repeat the process for signals 2500 Hz and above by generating sine sweep from 2500 Hz to 3500 Hz . • Verify the frequency of WS signal using oscilloscope. • Verify the frequency of SCK signal using oscilloscope. • Run code to capture data 256 milliseconds after initial power up and ensure 1 raw audio sample

	corresponding to $1/(16 \times 10^3)$ seconds = 0.0625 milliseconds is captured.

*The sine sweep audio can be played using a smartphone

Data Processing Unit

In addition to Signal processing, the microcontroller performs data processing by converting the audio samples from the filter to a 1 minute WAV file and saving it to the SD card. The header of the WAV file is 44 bytes long and has the structure shown in Figure 8. The outputs of the microphones contain 24-bit raw audio data sampled at 16 kHz, which means the sampling rate specified for our 24-bit WAV file would be 16 kHz. The size of each WAV file would be 2.8125 MB calculated as:

1 minute audio file size

= (sample rate) x (sample bit) x (# of channels) x time in seconds / (8 x 1024)

= (16000) x (24) x (1) x (60) / (8 x 1024)

= 2812.5 KB = 2.8125 MB

Positions	Sample Value	Description
1 - 4	"RIFF"	Marks the file as a riff file. Characters are each 1 byte long.
5 - 8	File size (integer)	Size of the overall file - 8 bytes, in bytes (32-bit integer). Typically, you'd fill this in after creation.
9 -12	"WAVE"	File Type Header. For our purposes, it always equals "WAVE".
13-16	"fmt "	Format chunk marker. Includes trailing null
17-20	16	Length of format data as listed above
21-22	1	Type of format (1 is PCM) - 2 byte integer
23-24	2	Number of Channels - 2 byte integer
25-28	44100	Sample Rate - 32 byte integer. Common values are 44100 (CD), 48000 (DAT). Sample Rate = Number of Samples per second, or Hertz.
29-32	176400	(Sample Rate * BitsPerSample * Channels) / 8.
33-34	4	(BitsPerSample * Channels) / 8.1 - 8 bit mono2 - 8 bit stereo/16 bit mono4 - 16 bit stereo
35-36	16	Bits per sample
37-40	"data"	"data" chunk header. Marks the beginning of the data section.
41-44	File size (data)	Size of the data section.
Sample values are given above for a 16-bit stereo source.		

Figure 9: WAV file structure [5]

Requirements	Verification
● WAV file conversion of the raw audio sample. ● Processes only 4 WAV files at a time and stops conversion right after.	● Run a raw audio sample through the data conversion code and ensure output is a 24 bit, 16 kHz WAV file. Verify the sampling rate corresponding to the WAV file using MATLAB. ● Convert four 1-minute audio samples to WAV files and check if exactly 11.25 MB SD card memory was used.

Memory

An external memory will store the patients data. The data stored will contain the heart and lung audio file of the patient as well as the time it was recorded. This unit is needed so that the doctor can access the patient's history at any given time. The SD card has read and write speeds of 16 MB/s, which means it will take 8 seconds to fill the 128 MB memory. This unit will store 11.25 MB of WAV files corresponding to four microphones calculated as:

= 4 WAV files x (2.8125 MB / WAV file) = 11.25 MB

Requirements	Verification
● Have a total usable memory of 11.25 MB to store a minimum of 4 WAV files ● Operate on 3.3 V +/- 0.1 V	● Put 11.25 MB file on SD card, put the card in a smartphone and check if the "Memory card full" message appears on the screen when we try to store more files. ● Attempt to power the chip with 3.2 - 3.4 V.

Additional Circuits

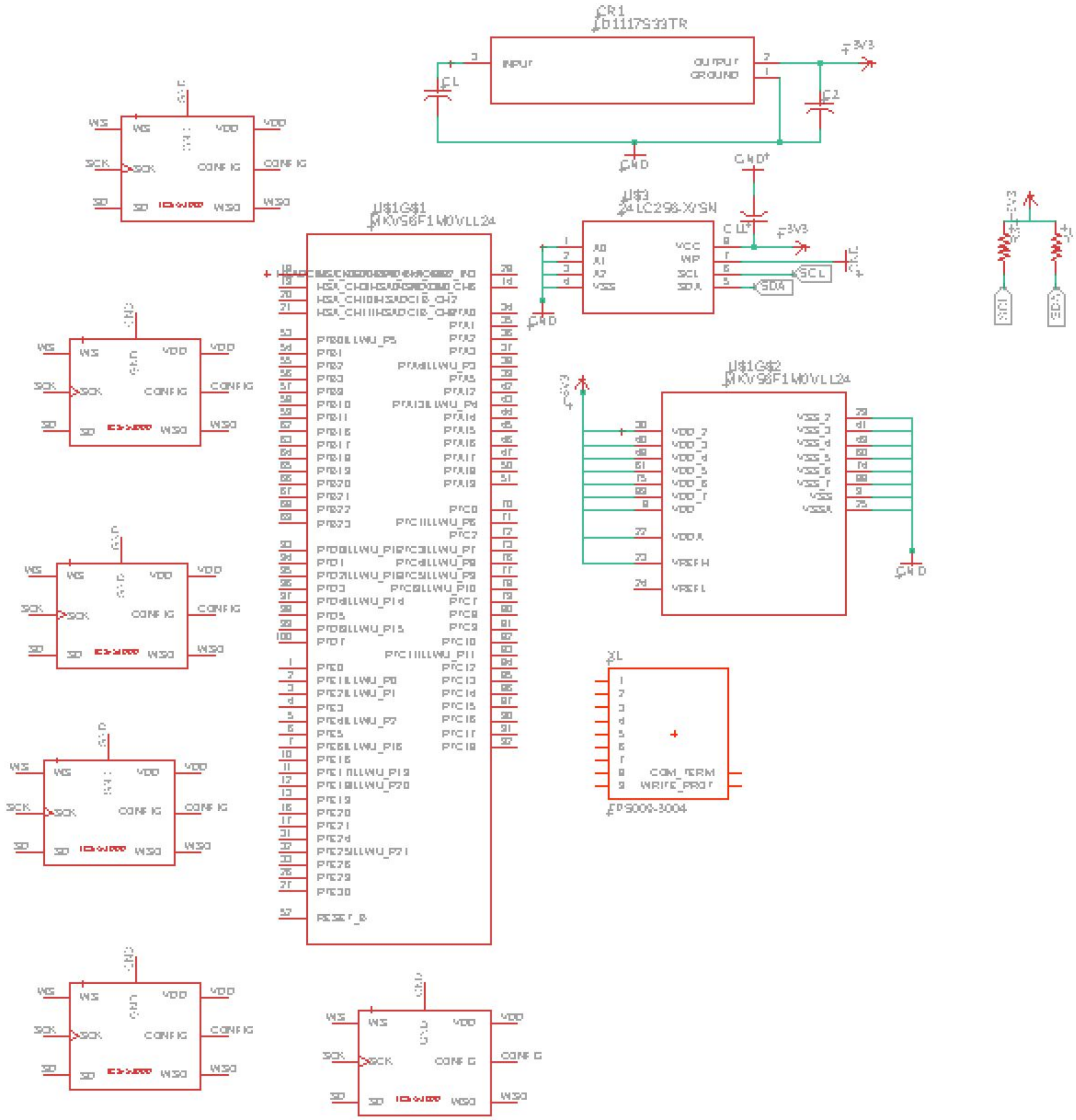

Figure 10: Microcontroller + Memory circuit

Tolerance Analysis

The outputs from the digital microphones in our project are crucial for the success of this project. By design, all of the ICS-52000 microphones on a common bus will sample their acoustic signals simultaneously and output their individual data words in their respective time slots in the TDM bus. Synchronized sampling is critical for the performance of a multi-microphone array because the algorithms typically used with these arrays require accurate synchronization between signals for optimum performance. Without the ICS-52000's integrated synchronization at start-up, these DSP algorithms would require a complicated series of delays and/or buffers to pre-process the signals to achieve the best algorithm performance.

It is also very important to consider the Total Harmonic Distortion of the output of the microphone in our analysis. A higher THD measurement indicates a higher level of harmonics present at the output of the microphone. The THD of the MEMS microphones is calculated from the first five harmonics of the fundamental. The input signal for this test is typically at 105 dB SPL, which is 11 dB above the reference SPL of 94 dB. THD is measured at a higher SPL than other specifications because, as the level of the acoustic input signal increases, the THD measurement typically increases as well. A rule of thumb is that the THD triples with every 10 dB increase in input level. Therefore, THD less than 3% at 105 dB SPL means that the THD will be less than 1% at 95 dB SPL.

The sounds we need to record occur in 50 Hz - 2500 Hz frequency range. This means we need a minimum sampling rate of 2 x 2500 Hz = 5000 Hz to prevent aliasing following Nyquist's Theorem. The audio data from the microphone is sampled at 16 kHz [6] and the ARM Cortex M7 has clock speed of 100 MHz, when using the internal RC oscillator, which means we need to divide the clock frequency by $(100 \times 10^6)/(16000) = 6250$ to achieve the desired sampling rate.

We also need a clock frequency of 256 x 16 Hz = 4096 kHz [6] for operating the microphone, which can be generated by dividing the clock by 100 /(4096000) = 24.414. We can only divide the clock frequency by a positive integer and 25 is a good option. This gives us an SCK frequency of 4000 Hz which is within +/- 2.34375 % of 4096 Hz and will work for our design.

The 128 MB SD card used in our design can store 11 (128/11.25 = 11.37) sets of 4 WAV files.

Cost and Schedule

Cost Analysis

Labor

Our fixed developmental costs are about $40/hr, 10 hrs/week for three laborers.

$$\frac{\$40}{1\ hr} * \ 2.5\ * 10\ weeks\ * \ \frac{10\ hours}{1\ week} = \frac{\$10{,}000}{laborer}$$

Considering we have 3 laborers, our total labor cost is,

$$\frac{\$10{,}000}{laborer} * 3\ laborers = \ \$30,000.$$

Parts

Description	Manufacturer	Model #	Units	Units cost	Total
Microcontroller	NXP	Kinetis KV5x	1	$16.72	$16.72
Microphones	Invensense	ICS - 52000	10	$2.85	$28.5
SD card	SanDisk	SDSDB-128	1	$9.98	$9.98
Resistors	Various	Various	20	$0.10	$2.00
Capacitors	Various	N/A	24	$0.30	$7.2
Voltage Regulator	STMicroelectronics	LD1117	1	$1.95	$1.95
PCB	PCBWay	N/A	1	$4.00	$4.00
Cotton Roll	Curad	N/A	1	$5.23	$5.23
Lithium Coin Battery	Renata	CR2477N.IB	5	$1.74	$8.7
Connectors	Various	Various	4	$1.00	$4.00
Oscillator	Various	Various	2	$1.40	$2.8

Total Cost					$91.08

Grand Total *= Labor Cost + Parts Cost*

= $30,000 + $91.08

= $30,091.08

Schedule

Deadline	Abhiyash	Hesham	Marc
3/4	Build the Power circuit and test the voltage regulation.	Order the microphones, bluetooth module, microcontroller and memory chip.	Build the Power circuit and test the voltage regulation.
3/11	Verify the microphones, test the Frequency response.	Program the DSP Filtering and test it with sample digital signals.	Program the DSP filtering and test it with sample digital signals.
3/18	Program the data conversion of digital signal into WAV format and test memory circuit.	Solder the microphones onto Evaluation boards.	Program test benches for data conversion.
3/25	Build the microcontroller-SD card interface.	3-D print the microphone headers and stitch the two-layered shirt.	Program test benches for data conversion.
4/1	Transfer test WAV files to SD card and check for aliasing as well as noise.	Perform verification for each component of the project and include data in report.	Perform verification for each component of the project and include data in report.
4/8	Test the final system	Test the final system	Test the final system

	and ask doctor for feedback on the shirt.	and ask doctor for feedback on the shirt.	and ask doctor for feedback on the shirt.
4/15	Prepare final presentation	Prepare final presentation	Prepare final presentation
4/16-4/20	Mock demonstrations	Mock demonstrations	Mock demonstrations
4/23-4/25	Demonstrations	Demonstrations	Demonstrations
4/26-4/27	Mock presentation	Mock presentation	Mock Presentation
4/30	Presentation	Presentation	Presentation

Ethics and Safety

There are several aspects of the project that can pose a safety hazard to the user. The Power Unit consists of batteries providing a total of 3.3 V to our system and it can easily overheat causing discomfort to the user and correlates to the IEEE Code of Ethics #9 [3].
Another possible hazard is water which can short the components on the circuit board and even harm the user which again correlates with the IEEE Code of Ethics #9 [3]. We need to ensure that the user is not sweating while wearing this shirt and it is not worn for prolonged periods of time.
The material of our shirt is another safety concern as certain materials can cause allergic reactions in some people. We are using a shirt made of 100% cotton for this project and are going to ensure that the volunteers testing this suit do not have such allergies. We will make sure that the user is aware of the materials used to build the suit following IEEE Code of Ethics #3 [3].

A crucial factor in determining the success of the suit is whether doctors are able to detect the relevant sounds from the sound files generated by the system. Therefore, we need to make sure that we work closely with a doctor and a patient to make our device better in accordance with IEEE Code of Ethics #7 [3].

Since this shirt is a medical device, we need to make sure Food and Drug Authority (FDA) medical device regulations are satisfied and the need for human participants in this project requires us to follow the Institutional Review Board (IRB) guidelines [1].

References

[1] " A. Paulson, "Institutional Review Board (IRB)," American Public University System (APUS), 09-Mar-2017. [Online]. Available: http://www.apus.edu/academiccommunity/research/institutional-review-board/index. [Accessed: 10-Feb-2018].

[2] J. Gerardo Avalos, Juan C. Sanchez and Jose Velazquez, "Applications of Adaptive Filtering".

[3] " Ieee.org, "IEEE IEEE Code of Ethics", 2016. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 10- Feb-2019].

[4] ST Microelectronics, "Low Drop Fixed and Adjustable Positive Voltage Regulators," LD1117 Series datasheet, [Revised Dec. 2005].

[5] Digital Audio - Creating a WAV (RIFF) file. Available: http://www.topherlee.com/software/pcm-tut-wavformat.html

[6] TDK InvenSense., " Low-Noise Microphone with TDM Digital Output", ICS-52000 datasheet, April 2017.

[7] Dogan Ibrahim, "SD Card Projects Using The PIC Microcontroller". Available: https://m.eet.com/media/1114317/1515_ch3.pdf

[8] TDK Invensense., " I2S/TDM Output MEMS Microphone Flex Evaluation Boards", ICS-52000 datasheet, April 2017

[9] Mansoor Mehmood, Hazem L Abu Grara, Joshua S Stewart, and Faisal A Khasawneh, "Comparing the auscultatory accuracy of health care professionals using three different brands of stethoscopes on a simulator". Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4140709/