AudioDAQ: Turning the Mobile Phone's Ubiquitous Headset Port into a Universal Data Acquisition Interface

by

Sonal Verma

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Engineering
(Electrical Engineering)
in The University of Michigan
2012

Thesis Committee:

Assistant Professor Prabal Dutta, Chair
Assistant Professor Zeeshan Syed
Assistant Professor Zhengya Zhang

Dedicated to my family for their love and support

ACKNOWLEDGEMENTS

This dissertation marks the single most challenging yet memorable period of my life. Through the many experiences in my graduate work, I not only gained scientific perspective but also grew into a better person, instilled with patience and confidence. I certainly could not have been successful without the help of many wonderful people who I acknowledge here.

First, I would like to express my sincerest gratitude to my advisor Prabal Dutta. I consider myself truly lucky to have worked with him these past years. He showed faith and patience in me as an incoming graduate student. He created an environment which allowed me to be comfortable, to grow and learn. He always used to be concerned about my well-being. He was flexible to allow me to explore and work on the hardware side of things. He defined the scope for this project and his expertise and invaluable advice helped me immensely to complete this work. He has provided me with every opportunity to succeed in the field.

Special thanks goes to Andrew Robinson with whom I have worked jointly on the AudioDAQ project. He worked on sending sensor data to cloud for storage and processing and data reconstruction of the analog signals on the mobile phone. His work made a significant impact in the overall success of this project. We worked together in evaluating the system,

running experiments and writing a paper.

My heartfelt gratitude also goes to Thomas Schmid, my mentor and a great friend with whom I started working when I first joined the lab. He was always very approachable and available for discussion. He was a constant source of knowledge and inspiration. Without him my graduate experience would not have been so rewarding.

I thank my committee members Professor Zeeshan Syed and Professor Zhengya Zhang for their time and effort in evaluating my dissertation and for participating in my thesis defense. I would also like to thank the professors in the EECS for helping me build a strong foundation through coursework which helped me solve problems in research.

I needed some people to take my mind off research and for that I would like to thank Lauri K Johnson. She was not only a great help with the administrative work associated with my research and conference travel but was also a great friend with whom I could share my thoughts and feelings at our weekly lunches. I would also like to acknowledge the help I received from Beth Stalnaker, who was remarkably understanding of my situations and flexible with my paperwork.

My time at Michigan would not have been enjoyable at all if it had not been for my talented group mates: Ye-Sheng Kuo, Aaron Schulman, Lohit Yerva, Sam DeBruin, Pat Pannuto, William Huang, Trey Grunnagle, Brad Campbell, Ben Kempke and Maya Spivak. You guys made coming to work a lot of fun. I will take back fond memories of our discussions, lunches and trips to conferences.

I thank my fiancé, Rahul Mathur, who stood by me in thick and thin. I cannot wait to marry him and share my rest of the life with him. I thank my little sister, Anjali, whom I feel lucky to have every single day. She is someone who I can always count on. Last

but not the least, I express my utmost gratitude to my parents, Surender and Alka, for their unending love, support and encouragement. I can never repay for the sacrifices they have made for me. But as the smallest token of my appreciation, I dedicate this thesis to them.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

LIST OF APPENDICES

Appendix

ABSTRACT

AudioDAQ: Turning the Mobile Phone's Ubiquitous Headset Port into a Universal Data Acquisition Interface

by

Sonal Verma

Chair: Prabal Dutta

We present AudioDAQ, a new platform for continuous data acquisition using the headset port of a mobile phone. AudioDAQ differs from existing phone peripheral interfaces by drawing all necessary power from the microphone bias voltage, encoding all data as analog audio, and by making use of the phone's built-in voice memo application for continuous data collection. These differences enable low overall system power, make the design more universal among smart and feature phones, and enable simple peripherals without requiring a microcontroller. In contrast with prior designs, AudioDAQ works with a wide range of smart and feature phones, requires no hardware or software modifications on the phone, uses significantly less power, and allows continuous data capture over an extended period of time. The design is efficient because it draws all necessary power from the microphone bias signal, and it is general because this voltage is present in nearly every headset port. We show the viability of our architecture by evaluating an end-to-end system that can capture

EKG signals continuously for hours, process the data locally, and deliver the data to the cloud for storage, processing, and visualization.

CHAPTER I

Introduction

The mobile phone has become a ubiquitous element of modern life. It provides many of the features that made personal computers popular in a convenient, slim form-factor, coupled with a rechargeable power source and bidirectional radio. There has been considerable research in last decade in making mobile phone an ideal platform for continuous sensing applications and exploring a universal purpose-built peripheral interface port for communicating with external hardware devices.

An obvious candidate to fill this void is the headset port, which has been made (mostly) standard across many mobile phones by necessity to ensure compatibility with a broad range of handsfree and headphone audio devices. Current smartphone peripherals such as the Square card reader and RedEye mini confirm this suspicion and suggest a growing interest in using the headset port for more than just headsets.

For reasons of cost, simplicity, elegance, and ubiquity using the headset port as a general-purpose interface for transferring power and data to peripheral devices is an attractive option. However, while at first glance the port seems well suited, it is not perfect. While most headsets are indeed designed around a few common phone jacks, there is con-

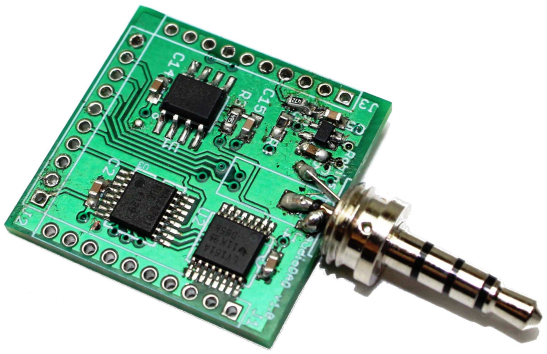


Figure 1.1: The AudioDAQ board, a square-inch mobile peripheral capable of providing a power and communication channel to a variety of sensor peripherals by (a) modulating an analog signal into the audio passband and (b) harvesting a small amount of regulated energy from the microphone bias voltage for powering external active amplifiers. This board is inexpensive in volume, making use of common, mass-produced analog components and can easily be miniaturized.

siderable variability in the passband characteristics, power delivery capability, signal pin outs, and microphone biasing voltage among phones. This leads us to conclude that contrary to recent claims, the headset port is not as universal as one might hope. Designs built to work with the iPhone, for example, may fail to work on Android and Windows phones, and vice versa. Designs built to work with smartphones may fail to work with the far more numerous feature phones.

It is ubiquitous without being universal. It can transfer power but it does so inefficiently. It has bandpass characteristics that severely limit the frequency range of passable signals. Features that at first seem to suggest great utility, such as the high sampling rate required for capturing audio, and the ability to create arbitrary waveforms on the output audio channels turn out to be poorly suited for use in peripherals that do more than record or play audio, due to details in their audio-specific implementations.

Recent work on creating peripherals for this port has focused on delivering large amounts of power for sporadic sensing activities, however what works for sporadic activities doesn't

often work for continuous sensing, where total power consumption has a much higher impact on the utility of the application. We recognize that there exists a class of sensors that require a continuous, yet minuscule amount of power to operate over extended intervals. This class includes biometric sensors such as EKG monitors and environmental sensors, such as electrochemical gas detectors and soil moisture probes.

We claim that a miniaturized, low-cost, low-power interface platform can be designed which will draw all the necessary power from the microphone channel of any mobile phone and enable interfacing of a class of low-power sensor peripherals. This can enable continuous sensing applications for extended periods of time without requiring any software or hardware modifications on the phone.

By targeting a subclass of sensing peripherals we limit the power delivery and data transfer requirements, giving us greater flexibility when designing against the headset port. We perform a survey of existing mobile phones and based on this survey create a design optimized around headset port parameters with the least variability among phones that are also best suited for long term, low-power sensing.

In this thesis, we present AudioDAQ (Figure 1.1), a system designed to provide *both* continuous, low-power capture of analog sensor data and a small amount of power, harvested from the microphone bias voltage, to power active sensors. We justify our design with a brief survey of existing headset port energy harvesters, and the trade-offs of each. Finally we evaluate our design, characterizing signal recovery and distortion across diverse operating conditions, the energy costs of signal processing on the phone as compared to processing on a remote server, and we present a strong example application, a low-power two-lead EKG sensor capable of taking data continuously for extended sampling periods.

CHAPTER II

Design

AudioDAQ is an end-to-end system for capturing, annotating, storing, sending, and processing analog sensor data. Figure 2.1 presents the general system architecture. The system is broken apart into four blocks:

Hardware Interface: A linear regulator ensures that the analog sensors are supplied with a stable voltage reference. Circuitry described in Section 2.2 modulates the analog data into the audio passband.

Mobile Phone: The raw audio data is captured by the mobile phone, which acts as an encoder and a storage device. We make use of the built-in voice recording application to capture data, which is generally efficient, allowing for extended capture periods. The data are later transmitted to a remote server for processing.

Processor: The encoded audio data is received by a remote server and decoded. The server runs an edge detection algorithm on the data, finds the correct framing, and reconstructs the original analog signals.

End User Applications: After being reconstructed, the data is exported and application specific code can take an appropriate action.

Our system supports low-power sensors that output analog voltage signals. To allow for amplification of low voltage signals, we provide enough regulated voltage to power a small amount of active circuitry. The signal from the sensors is modulated into the audio passband using a simple modulation scheme.

The multiplexed (or modulated) signal outputted by the hardware interface is fed into the next block, the mobile phone. Inside the phone, the signal is conditioned by the analog audio front-end, which at times works against us, and is captured using the built-in audio recording application. The signal is compressed efficiently, in most cases using hardware assisted encoding machinery for storage.

After being stored in the mobile phone, the user later sends the data to a server for processing. On the server the data goes through multiple stages of processing: normalization, edge-detection, framing, and reconstruction. If more than one signal has been multiplexed into the composite waveform, it is demultiplexed at this point.

Finally, the sensor data reaches the data application layer of the architecture. Here any implementation-specific transformations are performed and the data is formatted for use by the end user. In a typical application, this stage will process the data, extract domain specific key features, and generate useful plots. In our example application, an EKG sensor, this step extracts key metrics such as pulse-rate and the data is plotted.

2.1 Microphone Bias as Power Source

Recent work in creating peripheral devices for the headset port has focused on using the audio output driver to provide system power. It is the obvious candidate: by being

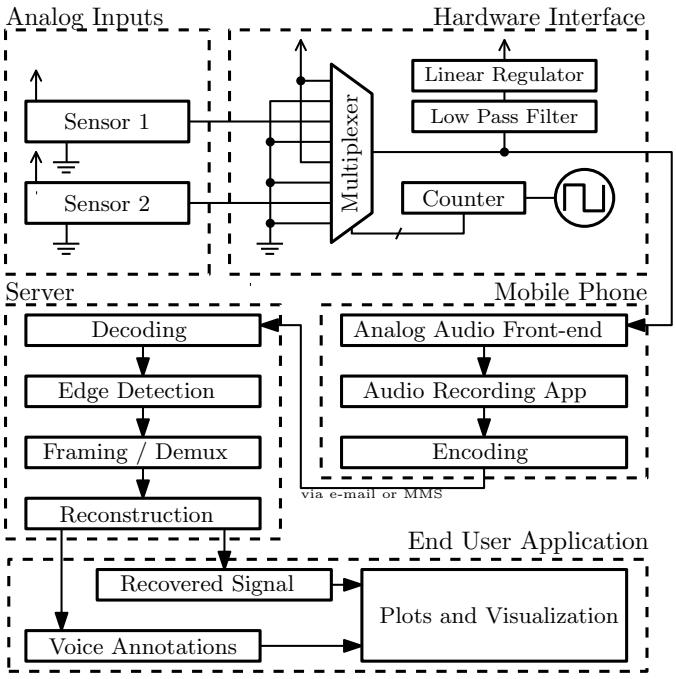


Figure 2.1: System architecture. By using the built-in audio recording utility and offloading the processing to cloud based servers, the load placed on the mobile phone is significantly reduced and the sample time is extended.

necessarily powerful enough to drive earphones and speakers it can deliver plenty of energy. Using it in practice presents serious design challenges. While the audio output driver is capable of delivering large amounts of energy, it does so at a very low voltage, significantly lower than typical voltage level required for operation of power switching electronics, and requires this energy to be exported as a sinusoidal wave within audible frequencies.

A typical approach to harvest power from this output is seen on HiJack [43]. Software on the phone generates a sinusoidal waveform that is sent to the audio output driver, exported over an audio channel, fed through a micro-transformer to reach a voltage level high enough to switch a FET, and finally rectified using a rectifier and regulated to a stable DC voltage.

Using the audio output driver in this manner has significant drawbacks. Generating

the sinusoidal waveform requires custom written phone software and consumes significant amounts of power. Moreover, converting the output of the audio driver into a usable DC voltage requires sophisticated circuitry and often results in poor efficiency. Finally, while the typical audio driver can deliver a significant amount of power as compared to the microphone bias voltage, there is a high degree of variability between phones, making it difficult to design a circuit that is universal enough to work across many headsets.

In AudioDAQ, we instead opt to exclusively use the microphone bias voltage to power our headset peripheral and attached sensors. This voltage has so far been overlooked as a viable power source by previous work. Intended to power the small amplifying FET found in electret condenser microphones, it is only capable of delivering minuscule amounts of current. Many applications would find this to be insufficient. With AudioDAQ, the simplicity of the modulation circuitry lends itself well to being powered by the bias voltage, consuming approximately 110 uW of power, well below the maximum power delivery of a typical headset port.

Figure 2.3 shows the maximum deliverable power and the optimal point in the P-I-V space for both the microphone bias voltage and the audio output driver. Less phones were surveyed when measuring the audio output driver's parameters due to the difficulties in developing custom software for each phone, which is a major limitation of using the audio driver as a power source. While the microphone bias voltage delivers significantly less power, the variability in implementations is less, simplifying the design. The open-circuit voltage of the microphone bias line in the phones surveyed ranged from 1.7 V to 2.7 V. The circuitry on AudioDAQ requires 1.8 V to operate, making it compatible with most but not all of these phones without adding complex voltage boosting circuitry.

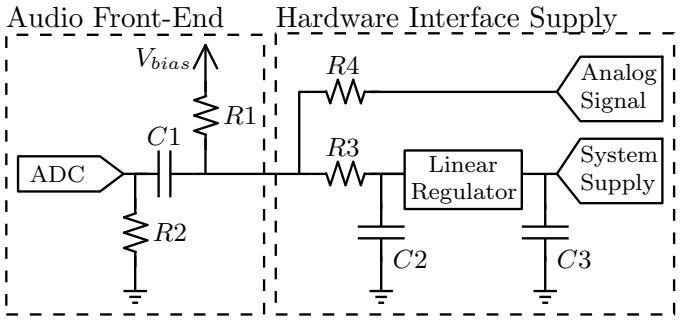
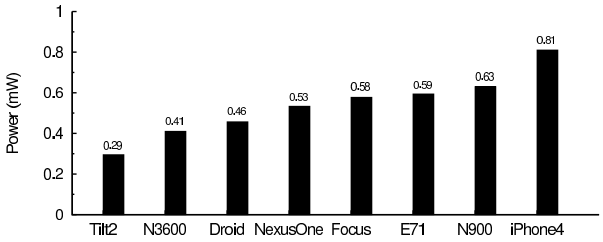
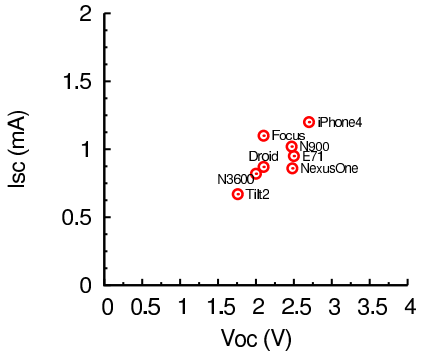


Figure 2.2: Microphone harvester schematic, including a model of the phone audio front-end which shows both how the microphone bias voltage is generated, and the filtering done on the input signal.

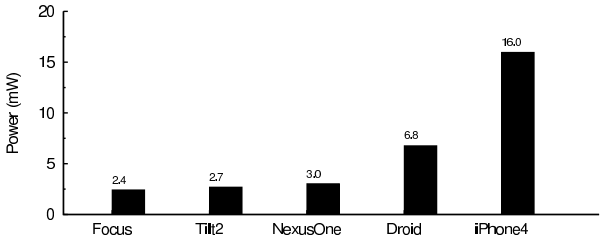
Using the microphone bias voltage as a power source offers a new set of design challenges. Because we must also use this microphone channel to transmit sensor data back to the phone, the power supply and data transfer characteristics are deeply coupled. Figure 2.2 shows a model of the phone circuitry responsible for processing microphone data and generating the microphone bias voltage connected to the power supply of our hardware interface board, and where we inject the analog signal back into the microphone line. $R3$ and $C2$ form a single order RC filter to stabilize the linear regulator and prevent regulator control loop noise from reducing the fidelity of the signal, which is of an extremely low amplitude (10mV peak-to-peak) to be compatible with the phone's audio processing circuitry. The cut off frequency for this low-pass filter is set to 50 Hz which is far below the modulation frequency of the analog signals. The bias voltage is a high-impedance voltage source and its output current limited by $R1$ in Figure 2.2. Because of this great care must be taken when using it as a power source. Components cannot draw large transient currents without proper decoupling capacitors, doing so causes large voltage drops, but the capacitances must be kept small to ensure that they do not attenuate the low amplitude modulated signal.



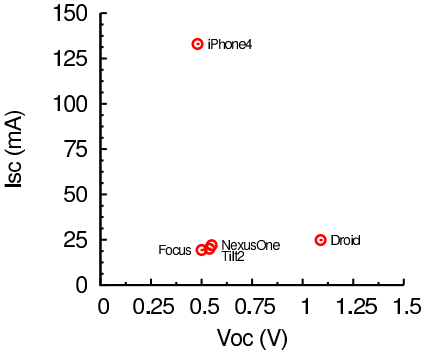
(a) Maximum Output Power from Microphone Channel



(b) Microphone Parameters



(c) Maximum Output Power from Audio Output



(d) Audio Output Parameters

Figure 2.3: Output characteristics of the audio interface on several mobile phones. The maximum output power of the microphone bias line, as calculated from the short circuit current and open circuit voltage in 2.3a is much lower on average than the maximum output power deliverable via the left and right audio channels 2.3c. Fortunately it is high enough to power many small active transducers and the circuitry required to modulate their output signal into the audio passband. The significantly lower variability in maximum output power across phones, and the optimal current-voltage point (2.3b vs. 2.3d) required to achieve this power across many phones makes designing power electronics for this channel easier.

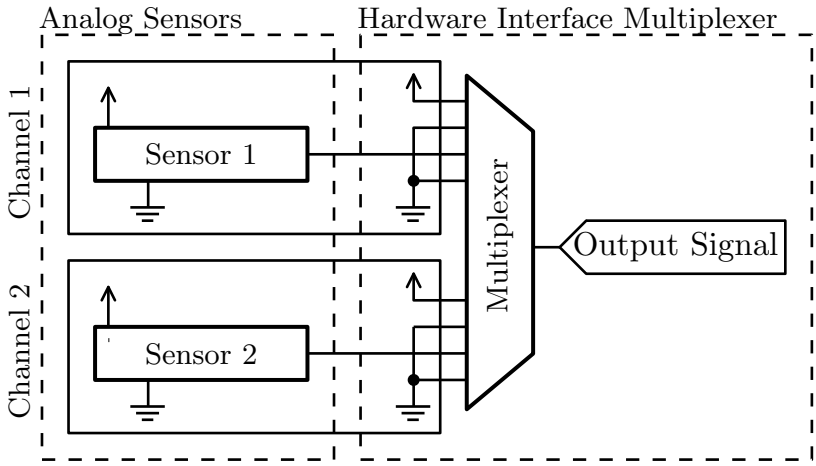


Figure 2.4: Acquiring analog sensor data using 8:1 analog multiplexer. We switch between analog inputs at a frequency of 1.2 kHz to create a composite signal within the audio passband. To recover the DC value of the analog signal a reference voltage of a known value is added to the analog multiplexer inputs. Due to the high-impedance nature of both the reference voltage and the analog sensor signal, every other input is connected to a low-impedance ground source, which prevents any bias charge from building up on the mobile phone's internal high-pass filtering capacitor.

2.2 Acquiring Analog Sensor Data

The typical audio front end in a mobile phone is optimized to acquire signals with amplitudes around 10 mV peak-to-peak, and frequencies within the 20 Hz to 20 kHz audible range. But, many signals either have principal frequency components below 20 Hz (e.g. EKG signals) or are purely DC in nature. This makes it impossible to pass them through this bandlimited microphone channel. To overcome this limitation, we use an analog multiplexer as a poor man's modulator and modulate the analog signal into the audio passband by rapidly switching between signal and ground, as Figure 2.4 shows. The analog multiplexer is driven from a counter, clocked with a low-power RC oscillator running at 1.2 kHz.

We expect the signal from analog sensors attached to our system to be a high-impedance voltage anywhere between system ground and the reference voltage of 1.8 V. The magnitude of this signal is too large to be fed directly into the microphone line. To fit within

the 10 mV limit, we add a scaling resistor between the output of the multiplexer and the microphone bias line. This resistor is identified as $R4$ in Figure 2.2. Calculations indicated that sizing the resistor around 200 $k\Omega$ scales the signal into an amplitude range that doesn't overwhelm the audio processing electronics for most mobile phones.

Variability in the microphone bias voltage causes variations in the exact amplitude of the multiplexed signal, and the mapping of signal amplitude to ADC counts among headsets is inconsistent, with each phone having a slightly different scaling factor when capturing an audio signal. These variations make it impossible to directly recover the magnitude of the analog sensor. To estimate the absolute voltage of analog signal, we add a reference voltage, generated by the linear regulator, to the multiplexer. This effectively time-division multiplexes the signal we are exporting on the microphone line between ground, the analog signal, and a known reference voltage, and allows us to later recover the DC value by scaling the analog sample signals with respect to the reference voltage and ground.

At this point in the design, the connections to the input of the multiplexer are, in order, a voltage reference, ground, the analog signal, and ground again. Switching to a low impedance ground after each voltage signal helps to remove any residual charge that has built up on the capacitor of the high-pass filter on the mobile device, which is labeled as $C1$ in Figure 2.2.

The final step in creating a flexible analog input design is allowing for the simultaneous capture of multiple input channels. We realize this feature in our design by simply duplicating this four signal block on the multiplexer for each additional channel we want to capture. The current design in Figure 2.4 supports simultaneous capture of two channels. If it is required to capture only a single channel, the two inputs are tied together with an

on-board jumper.

2.3 Power Efficiency

Traditionally, power efficiency is examined as the loss of total delivered power as compared to usable power by the system. In this regard, AudioDAQ is adequate but not exceptional, operating with a reasonably efficient linear regulator and incurring small amounts of loss from the input filters adding small amounts of resistance to the power path to prevent corruption of the analog output signals. The power efficiency of AudioDAQ, which consumes less than 1 mW of total system power, is almost inconsequential to the bottom line when compared to the much larger power consumption of the entire mobile phone. Of far greater interest is how the design decisions of the AudioDAQ interface influence the power consumption of other system components, such as the power draw from CPU usage and storage access. These components use orders of magnitude more power than AudioDAQ and as a result have a far greater effect on the effective sampling time when operating off of the internal battery, which is the true net gain of having a highly power efficient system.

To better understand how the design of a headset peripheral influences power draw and to avoid optimization of components which had relatively minor contributions to the power budget of the mobile phone we took the HiJack platform, and using the publicly available source code selectively disabled parts of the system to measure the approximate power draw of each component. The results of this experiment are summarized in Table 2.1. While this data is specific to the HiJack platform and iPhone device, we expect that it generalizes well to other devices and platforms.

From Table 2.1 it is clear that, with the exception of the screen, which is easily disabled by the depression of the power button on most mobile devices, there is no obvious candidate for optimization. Most identifiable components draw amounts of power roughly similar to one another. From this we took a strategy to optimize where we could, not targeting any specific component.

By using the microphone bias voltage to power our system instead of the audio output driver we eliminate power required to generate the output audio waveform, and reduce the amount of power required for I/O to the audio channels. We effectively eliminate the output entirely, allowing the phone to keep inactive a large portion of the audio acquisition interface.

By encoding sensor data in the audio passband, and simply recording it for later remote processing we reduce the power required to process the input signal. The highly efficient hardware encoding mechanisms found in phones, often responsible for compressing voice for transmission over the network makes this possible.

Subsystem	**Average Power Usage**
Audio Channel IO	114 mW
Input Signal Processing	106 mW
Output Signal Generation	60.0 mW
Data Visualization	19.1 mW
Screen	333 mW
Application Overhead	59.5 mW
Base System	68.5mW

Table 2.1: Power consumption of the HiJack platform on an Apple iPod, measured by selectively disabling certain components of the software system via direct modification of the source code. With the exception of the screen, no single component stands out as using a disproportionately large amount of power.

2.4 Capturing and Storing Data Efficiently

For long-term data acquisition, data capture and storage must be efficient. We do not process the data on the phone so the entire audio data must be stored. Storing raw data would be space prohibitive, so we employ the built in compression utilities found on the phone. Almost all mobile phones come bundled with a voice memo application that makes use of these algorithms to record low-quality audio suitable for voice memos.

On Apple iOS devices the voice memo application stores data in Advanced Audio Coding (AAC) format, which is a standard widely used by Apple, at 64 kbps. Samples are taken at 44.1 kHz from a single channel. On Google Android phones the built in application uses the Adaptive Multirate (AMR) encoding with an 8 kHz sample rate by default. Many other formats including AAC are available as part of the API, and sound recording applications that produce higher quality records do exist. Many feature phones also use the AMR encoding, having been specially designed for mobile applications and having seen wide adoption.

All of these codecs will sufficiently compress audio down into file sizes practical to store on a mobile device. Both smart and feature phones often come with built-in hardware support for these compression algorithms. On iOS and Android specific media subsystems are exposed to the developers that allow for hardware-enhanced encoding to take place, and are certainly used by the built in voice memo application. On feature phones the CPUs often have special multimedia modules, and implementations of the codecs is done either completely in hardware, or in heavily optimized low-level programming languages.

Because of this, storing the audio data is surprisingly efficient across most phones, and

the codecs do a good job of compressing the raw audio data into reasonable file sizes.

2.5 Processing Sensor Data

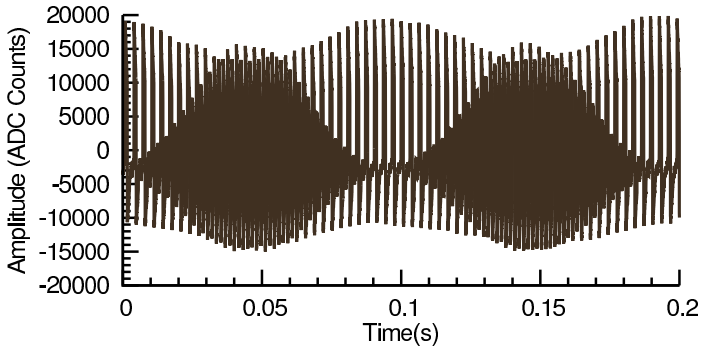
AudioDAQ processes the multiplexed analog sensor data to extract the original signals on a remote server. The audio files are uploaded to this remote server via e-mail where they are immediately processed and the data is extracted. This makes AudioDAQ general. Because many feature and smart phones manufactured today have the software capabilities to record and send a voice memo, AudioDAQ can literally work out of the box with the majority of mobile phones in use.

2.5.1 Signal Reconstruction

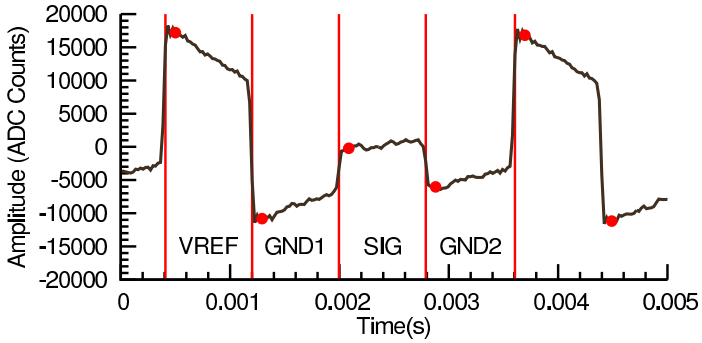
The signal reconstruction algorithm is currently implemented in Python, making use of external libraries to perform codec specific decoding of the compressed audio files. It is designed to examine data in a rolling window to allow for both online and offline processing and create robustness to noisy input data, by relying only on local data points and simply discarding signals that are too noisy for proper reconstruction.

The simplicity of the hardware interface poses a challenge for the signal reconstruction algorithm. Because the analog multiplexer can send no channel delimiter, the framing information must be implicitly determined. Signal reconstruction in the AudioDAQ system occurs in five stages:

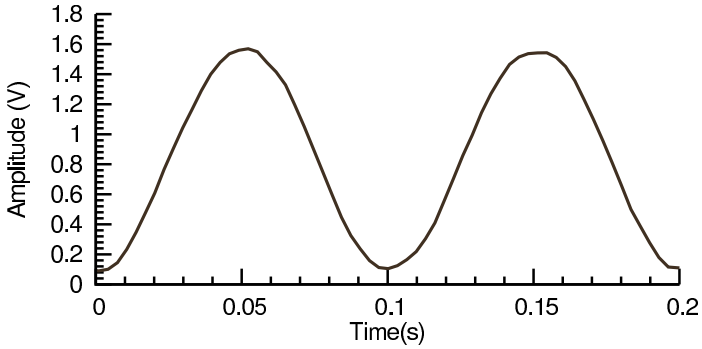
Decoding: The audio encoding format is deduced from the file extension and encoding specific magic bytes. The appropriate decoder is run on the data and the raw audio



(a) Signal Modulated by Multiplexer



(b) Reconstruction of Decoded Audio Data



(c) Reconstructed Signal

Figure 2.5: The raw signal captured by the phone's audio front-end, recorded, transmitted to a computer, and decoded as a sine wave 2.5a is fed into the reconstruction algorithm. Edge detection (the vertical red bars) and value extraction (the red dots) takes place and is shown in 2.5b, where a small segment of the signal has been magnified and annotated with the debug output of the algorithm. The extracted values are then used to compute the original input signal 2.5c. A small, periodic drifting bias in the absolute value of the signal can be observed in 2.5a. This bias comes from residual charge building up on the high-pass input filter's capacitor inside the mobile phone. The same effect can be seen on a smaller time scale in 2.5b, where after switching the signal follows an exponential drift towards zero-offset.

information is extracted.

Edge Detection: The transition edges that are created when the multiplexer switches signals are detected and marked on the data set.

Value Estimation: The regions between the edges are evaluated. Extreme outliers are discarded and an estimate of the value of the signal in that region is obtained.

Framing: Based on previously processed frames, the framing is deduced. Framing is important to determine which values corresponds to ground, the voltage reference, and the actual analog signal. Each frame of input data consists of the four multiplexer signals discussed in Section 2.2.

Calculation: The actual analog signal voltage for the frame is calculated by expressing the signal value as a point between the ground and voltage reference value and then scaling it to the known voltage reference.

A secondary benefit of including the ground to reference voltage transition is that it gives a reliable, repeating signal to help frame our data. Because the analog sensor signal could be at a value close to ground, it is impossible to reliably detect the edge transition between it and ground. In a frame the only two transitions we can reliably detect are the transition from the previous frame's ground to the reference voltage, and the transition from the reference voltage to ground. These transitions are detected by finding the maxima and minima of the first-order numerical derivative of the signal. The distance between these two transitions is calculated and used to estimate the final two transitions of the frame, between the unknown analog signal and ground. The vertical bars in Figure 2.5b show the detected and calculated edge transitions for a short period of input signal.

After edges have been detected the regions between the edges are evaluated. The high-

pass filter capacitor starts to immediately affect the signal after switching so the left-most region of the signal is used to estimate the nominal value of the signal for each region. A small number of points are averaged to a single value. These values are also plotted in Figure 2.5b.

Finally, the analog signal value is extracted. Up until now all processing has been done with ADC counts. To obtain the actual real-valued voltage, we express the analog signal as a value between the voltage reference and ground. The modulation scheme produces two ground values per frame, so the average is used, although in practice they do not vary much. After obtaining this decimal number, we can then multiple it by the known reference voltage (1.8 V in our system) to obtain the original analog signal.

2.5.2 Offline Processing

The most common scenario for data processing involves collecting data to a compressed file, and sending the file to a server for processing at a later point in time. This has the advantage of mitigating the power cost of transmitting the data to the remote location by delaying it until a point in time that it is convenient for the operator, such as when near a charger or connected with a faster wireless network such as 802.11b wireless.

An alternative offline processing scheme that was considered but not explored involved recording the data using the voice memo hardware and then periodically reading it in and doing a high-performance, faster than real-time computation on the data. This batch processing avoids the high idle cost of the CPU and the high wakeup cost of going from sleep to wake up mode but still offers semi-real-time performance. Because one major strength of the AudioDAQ system is its compatibility with almost any hardware without requiring

additional software, we chose not to explore this option.

2.5.3 Online Processing

A less common scenario involves processing the data in real time. This is useful for demonstrative purposes and wireless transmission of sensor data to a remote host for real time display. Assuming a sufficient data connection is available it is possible to remotely stream data. Audio encoding algorithms for VoIP systems such as Speex, which has a mobile port available make this possible over TCP, and even a simple telephone call could provide the bandwidth necessary to stream the data. Streaming in real time would dramatically reduce the sampling period due to the much larger power demands of the wireless radio in the mobile phone.

2.6 Capturing Voice Annotations

An obvious addition to an analog sensor capture system is a method to notate the samples. Because we are limited by the phone's built in voice recording application, we do not have the ability to allow for the user to input text directly to be stored alongside the collected data, however, we can collect voice annotations.

Mobile phones typically detect the presence of a microphone not by the physical presence of a jack, but by the DC voltage present on the microphone bias line. Most microphones pull the voltage of the bias line down past a certain threshold. When the phone detects this, it automatically shifts between the internal microphone and the externally connected peripheral. We exploit this behavior by adding a momentary disconnect switch to

Pinout	Tip	Ring 1	Ring 2	Sleeve
(a) 3.5 mm	L	R	G	M & C
(b) 3.5 mm	L	R	G	M
(c) 2.5 mm	L	R	M	G
(d) 2.5 mm	S+	M+	S–	M–

Table 2.2: Audio headset pinout of several common mobile phones. (a) Apple, Blackberry, HP, Samsung; (b), (c) Nokia smartphone headset connector; (d) Nokia headset socket. Left audio output (L), right audio output (R), microphone input (M), control (C) and ground (G) are common on many smartphone headset ports. Some simple phone headsets offer speaker positive (S+), speaker negative (S–), microphone positive (M+), and microphone negative (M–). Although many smartphones use the same headset interface, there are important pinout differences between device classes, and even within some manufacturer's product lines, showing that the ubiquitous headset port is not universal.

the peripheral, effectively allowing the user to pause data collection, and record a short audio annotation in line with the collected data.

Because AudioDAQ has a distinctive principle switching frequency, it is easy to algorithmically detect the difference between voice data and analog sensor data. We do this server-side, and then, using open-source voice recognition libraries, convert the speech into text which is paired with the reconstructed data.

2.7 Mechanical Design

We chose a 3.5 mm headset jack because it has quickly become the standard for smartphones with the introduction of applications such as music playback that make use of existing standard 3.5 mm headphones. The mapping of pins to logical signals in 3.5 mm headset jack implementations is more consistent when compared to the now less common 2.5 mm interface, for which we found two common, yet separate, mappings. Table 2.2 shows the different pinout configurations for audio jacks across various phones. Inexpensive adapters and "proprietary" headset connectors [8] are available to connect our 3.5 mm peripheral to

existing 2.5 mm ports.

We chose a square-inch form factor for AudioDAQ and a pinout that is directly compatible with existing HiJack sensors. If necessary, however, the circuit could be compacted or even built into a single integrated circuit and incorporated into a molded audio headset jack itself. The present square-inch form factor gives a good trade-off between small size and ease of development and debugging.

CHAPTER III

Low-Power EKG Sensor

3.1 Introduction

Mobile devices have the potential to seamlessly blend health care into our daily lives [11], [22]. Evolving mobile health care [24], [40] technology can help professional caregivers to better monitor their patients, detect problems earlier, and reduce the need for hospital visits which are often expensive and always inconvenient. In addition, mobile health care can empower individuals to better monitor and manage their own health, encouraging them to live healthier lifestyles and prevent many health problems before they begin by providing methods for early detection. For these reasons, to demonstrate the potential and practicality of AudioDAQ, we have developed a low-power, low-cost, portable EKG monitor. This battery-free EKG monitor enables monitoring of an individual's cardiac activity for extended periods of time using relatively inexpensive electronics and existing mobile phones. It allows for hassle free data collection across a wide variety of phones, and for transmission of the data to a remote location where it is analyzed by automated algorithms for abnormalities, or by doctors for specific diagnostic purposes.

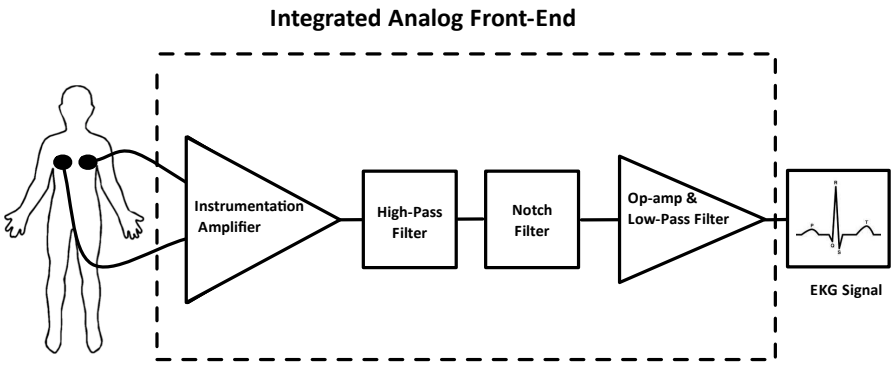


Figure 3.1: Block diagram of the EKG sensor system. Amplification is done in two stages to prevent the large amount of noise from pushing the signal too close to the supply rails, and destroying the EKG signal in the process. A micro-power differential amplifier first amplifies the signal received via electrodes placed on two body locations and removes common mode noise. This amplified signal is then passed through a notch filter to remove 60 Hz interference ambient power line noise. The final operational amplifier provides an appropriate gain, amplifying the signal to a voltage level usable by AudioDAQ and biasing it to half the supply voltage.

3.2 Design of Two-Lead EKG Sensor

The EKG sensor is a two lead device, with the leads attached to the subject's body using conductive hydrogel electrodes either on the left and right arm or wrist or directly across the chest. The signal is passed through two stages of amplification with active filters between to remove noise, as seen in Figure 3.1.

Amplifying cardiac signals [39], which are in the range of -5.0 mV to +5.0 mV and filtering out stray environmental noise captured by the human body poses a significant design challenge, made more difficult by the power budget constraints imposed by the AudioDAQ system. Instrumentation amplifiers were chosen with exceptionally low-power draw.

The first stage of amplification uses a differential operational amplifier, which rejects common mode noise found across the entire human body, leaving only the differential signal from muscle contractions, and amplifies this signal by a gain factor of five. This

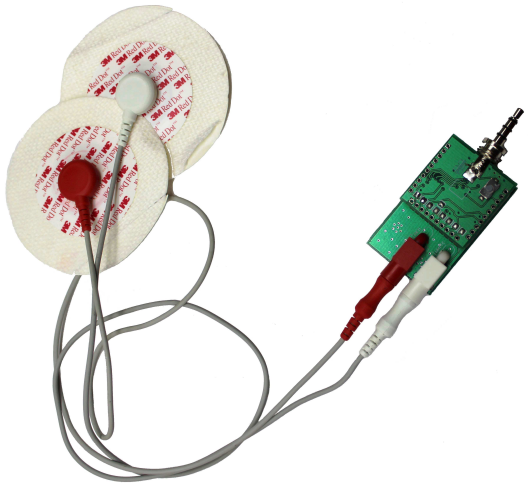


Figure 3.2: The AudioDAQ board interfaced with a 2-lead EKG sensor. The sensor draws 120 μA of power and provides filtering and necessary amplification.

amplifier has a high common mode rejection ratio of 95 dB which provides for sufficient common-mode noise reduction. It is integrated with a high-pass feedback filter that dynamically corrects DC shift in the ungrounded differential signal captured from the body. The human body acts as a large antenna around modern electrical grids so the amplified signal is passed through a notch filter specifically designed to remove 60 Hz common mode interfering noise. Finally the signal is fed through the last operational amplifier, which acts as a low-pass filter and amplifies the filtered signal with a gain of approximately twenty to bring it into the realm of voltage amplitudes commonly seen by analog to digital converters and to be used by AudioDAQ.

The useful bandwidth of an EKG signal is between 0.05 Hz to 150 Hz, so the high and low-pass filters act together as a bandpass filter configured to this frequency range. The output EKG signal is biased around half of the supply voltage using a voltage divider to minimize the possibility of the large voltage spike seen when heart muscles contract being chopped by the limitations of the operational amplifiers operating at their supply rails. The

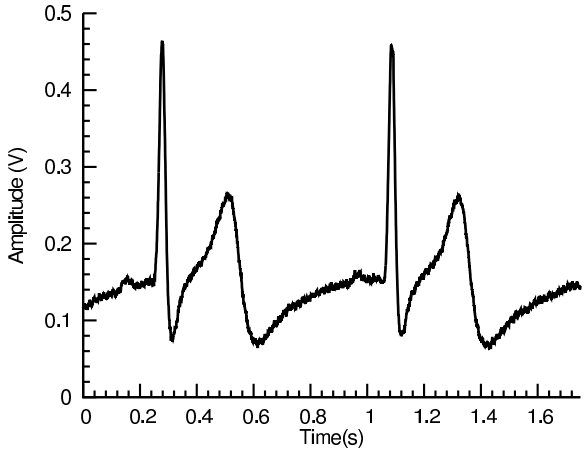


Figure 3.3: Real-time EKG trace of an individual. The observed 450 mV amplitude of the signal makes all the critical features (P wave, QRS complex, T wave) of a cardiac signal clear.

gains are set such that with good conductivity and lead placement, the EKG signal will have an amplitude of approximately 500 mV.

3.3 EKG Sensor with AudioDAQ

The sensor module is interfaced with the AudioDAQ platform to deliver the resulting EKG trace. Figure 4.1 shows the real time EKG waveform of an individual captured with this system. A typical EKG tracing of the cardiac cycle (heartbeat) consists of a P wave, a QRS complex, a T wave, all of which can be easily identified on our EKG monitor.

The EKG sensor interfaces with square-inch AudioDAQ base platform and consumes only 216 μW. Table B.4 shows the cost breakdown of an EKG sensor module. The marginal cost is $24, of which the majority is spent on leads, which could be commoditized at larger volumes for a significant cost reduction.

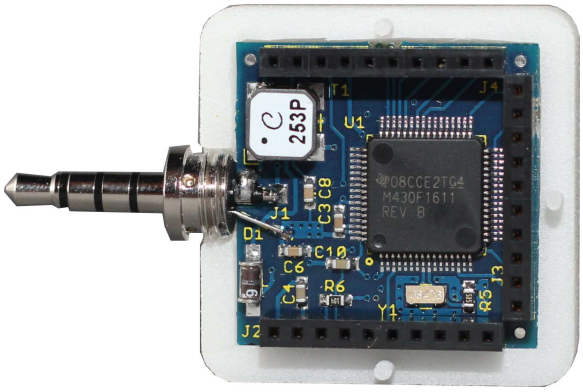


Figure 3.4: HiJack base platform, with a square-inch footprint, offer 7.4 mW power, analog (2x12-bit), digital(1xGPIO), serial (1xI2C and 1xUART) interfaces, exported via connectors, and all multiplexed over the headset port. This board provides the functionality needed to build a variety of external sensor interfaces for the mobile phone.

3.4 HiJack

3.4.1 Background

HiJack (Figure 3.9) is a hardware/software platform for creating cubic-inch sensor peripherals for the mobile phone. It harvests power and use bandwidth from the mobile phone's headset interface. The HiJack energy harvester can supply 7.4 mW to a load with a 47% power conversion efficiency. The harvester is driven by a 22 kHz tone from the output from a single audio channel on the headset port. The HiJack communications layer offers two data transfer schemes. The first allows 300 baud data transfer using Bell 202 FSK signaling. The second offers 8.82 kbaud using a Manchester-encoded, direct-digital communication using hardware accelerators on the HiJack microcontroller and a software-defined, digital radio modulator/demodulator on the phone. Thus, HiJack platform enables a new class of small and cheap phone-centric sensor peripherals that support plug and play operation. HiJack is compatible with iPhone 3G/3GS/4G [6], iPod Touch, and iPad devices [7].

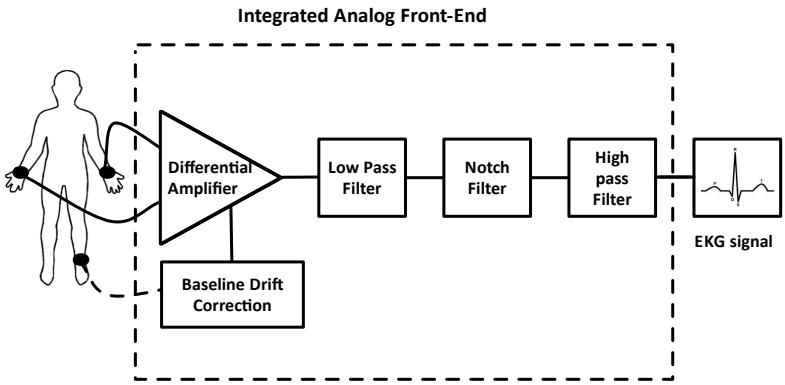


Figure 3.5: Block diagram of the EKG sensor system. Amplification is done in two stages to prevent the large amount of noise from pushing the signal too close to the supply rails, and destroying the EKG signal in the process. A micro-power differential amplifier first amplifies the signal received via electrodes placed on two body locations and removes common mode noise. This amplified signal is then passed through a notch filter to remove 60 Hz interference ambient power line noise. The final operational amplifier provides an appropriate gain, amplifying the signal and biasing it to half the supply voltage.

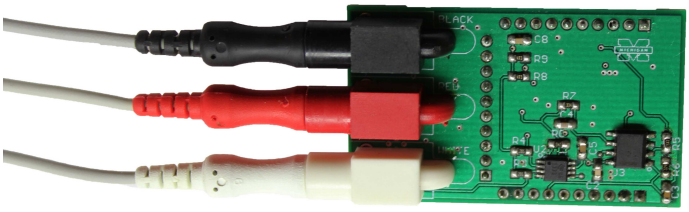


Figure 3.6: A square-inch, 3-lead EKG sensor module which draws only 120 μA at 2 V. This EKG monitor is an example of a small-size, low-power, low-cost sensor peripheral which can interface with smartphones using AudioDAQ platform for continuous cardiac monitoring for extended periods of time.

3.4.2 Design of Three-lead EKG Sensor

The major design challenges involve adequate amplification of cardiac signals (which are in the range of -5.0 mV to +5.0 mV) and filtering out noise. Figure 3.5 presents the basic building blocks of the analog front end of EKG sensor system. It is a 3-lead monitoring device, where the red and white leads receive EKG signals from two locations on patient's body using the hydrogel electrodes. These two locations can be either left arm and right arm or left wrist and right wrist or any two locations close to patient's heart. The black

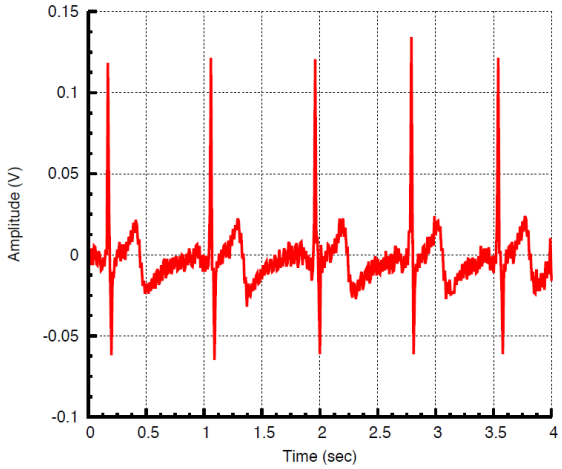


Figure 3.7: Real-time EKG trace of an individual. All the critical features (P wave, QRS complex, T wave) of a cardiac signal clear.

lead is optional and is connected to the right leg to reduce the baseline drift. The EKG signal is then passed through a micro-power instrumentation amplifier, which amplifies the differential signal by a factor of 5. This amplifier also offers a high CMRR of 95 dB which rejects most of the common mode noise. It is integrated with a high-pass feedback filter that dynamically corrects the DC shift. The amplified signal is next passed through a notch filter to remove 60 Hz common mode interfering noise. The final operational amplifier amplifies the filtered signal with an appropriate gain of 20X and also acts as a low pass filter. The output EKG signal is biased around 1.5V to minimize the possibility of large signals being clipped against the supply lines.

3.4.3 EKG Sensor with HiJack

Hospitals, clinics and urgent care facilities in developed regions have access to a battery of medical instruments unavailable to health care professionals in developing regions. One example of such an instrument is the EKG monitor. EKG signals are used to diagnose a wide range of medical conditions but they are often unavailable in all but the most

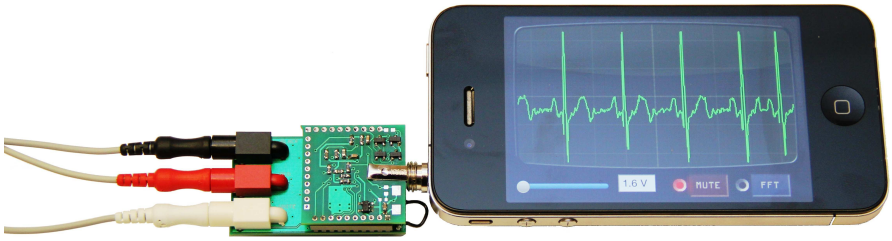


Figure 3.8: A 3-lead phone powered EKG signal conditioning board that interfaces to HiJack, consumes 120 μA and provides amplification, filtering and baseline drift correction.

advanced hospitals in developing regions. We argue, in this section, that this need not be the case. Rather, any hospital, village, clinic or doctor can have access to advanced health care instruments for little more than cost of a mobile phone. We illustrate our claim by designing a low-cost, low-power EKG monitor that uses HiJack for power and communications, and the mobile phone for visualizing the EKG waveform as shown in Figure 3.9. The EKG monitor attaches to the square-inch form factor HiJack base node and extends the PCB to provide space for connectors to attach EKG leads in a 3-lead configuration (Figure A.5). The EKG board draws less than 120 μA, amplifies the input signal, passes it through a notch filter, and corrects for baseline drift. The mobile phone visualizes the EKG and can optionally transmit the data to cloud for storage or real-time remote display (including another mobile phone or even a Facebook application). The EKG board led to an iteration in HiJack baseboard whereby the baseboard now provides regulated power and a wider interface on perimeter pins.

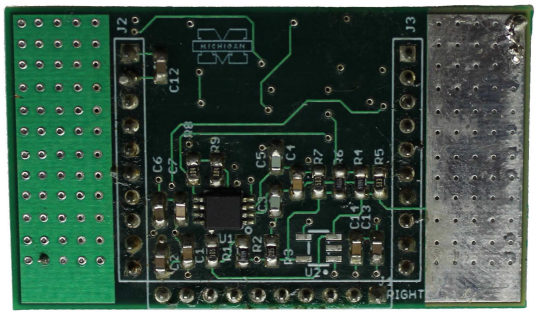


Figure 3.9: EKG sensor for acquiring cardiac signals from finger tips [35]

3.5 Acquiring EKG Signal from Finger Tips

The standard locations for placing electrodes for obtaining a person's EKG are his extremities and chest. We explore a design idea from TI [35], which uses a person's finger tips as the differential point of contact with conductive pads to acquire the EKG signal. This promised to be a low-cost, easy to use, heart-rate monitoring solution.

The system consists of two rectangular pads connected on top and bottom sides of the PCB board which serve as contacts for left and right hand finger tips. Appendix A shoes the entire circuit design and the layout. The design uses INA321 [36] as a differential amplifier configured to a gain of 10 V/V. TLV2765 [37] is configured to implement a fourth-order Sallen Key low-pass filter with a gain of 8.5 at each stage. The line frequency noise is at the frequency of 60 Hz. To eliminate this interference noise, the low pass filter is tuned to a frequency of 16 Hz. The components used in this design run at a minimum supply voltage of 2.7 V and the whole circuit has a gain of 722.5.

When we evaluated a PCB based on this design, we could not get a clean EKG wave. It seems that the signal obtained from the finger tips was too weak and could not be sig-

nal conditioned by this design. Lack of proper amplification and noise filtering circuitry rendered a noisy EKG wave at circuit's output.

CHAPTER IV

Evaluation

In this section, we evaluate the performance of the AudioDAQ system. We examine the major claims of universalness and of an extended data collection period and evaluate the system's signal distortion characteristics, power delivery capabilities, and cost effectiveness. We also justify our design decision to process data remotely by comparing the power consumption of local vs. remote processing.

4.1 System Functionality

AudioDAQ works with a wide range of phones, provides a small amount of energy suitable to power a large range of active sensors, and allows for an extended sampling period. In Table 4.1 the AudioDAQ system is evaluated against three smartphones, and two feature phones. The system operates on all phones, varying only in the sample period and available energy for additional sensor peripherals. Signal reconstruction was performed with multiple phones and no significant difference was seen in accuracy, despite differences in audio recording circuitry and recording format.

Phone	Works	Deliverable Power	System Power Draw	Battery Size	Estimated Sample Period
iPhone4	Yes	322 μW	149 mW	5254 mWh	35 hr
Nexus One	Yes	168 μW	267 mW	5180 mWh	19.5 hr
Galaxy Nexus	Yes	247 μW	301 mW	6475 mWh	22 hr
N900	Yes	120 μW	270 mW	4884 mWh	18 hr
E71	Yes	90 μW	189 mW	5550 mWh	29 hr
N3600	Yes	80 μW	178 mW	3182 mWh	18 hr

Table 4.1: Results of testing the AudioDAQ system across a subset of mobile phones. The platform works on every phone tested, showing that it is indeed universal, with variations in the maximum power delivery and total estimated sample period.

Depending on the specific audio encoding used, the phone will generate between 48 to 600 kilobytes per minute of data to be stored and eventually transmitted. For short data collections of up to 10 minutes in length, data files took a few minutes to transfer over a modern cellular network, with all phones having sufficient storage to hold these files and mobile networks being sufficiently fast enough to quickly transfer them. Extended capture intervals required large amounts of storage, and were only practical to be transmitted over a Wi-Fi connection or physical USB cable. Many current phones have gigabytes of internal storage and Wi-Fi connectivity, making long-term captures viable.

4.2 Collection Period

An extended capture was done on the iPhone4 to verify the estimated sample periods. The sample time for each phone is a function of both the efficiency of recording audio on that phone and the battery capacity. During the capture, at the 13 hr 30 min mark, the recording stopped. We assume that this is a hard file size limit built into the audio recording application. We restarted a second capture and allowed the phone to continue for another 13 hr 30 min period, at which point the capture ended again. The total capture lasted 27 hr, with enough battery remaining to verify the captures were successful and transfer the files to a computer over a Wi-Fi connection. The audio files were 400 MB in size each.

4.3 Power Delivery to Load

AudioDAQ includes an ultra-low dropout linear regulator which regulates the microphone bias voltage to provide a stable power supply to external analog sensors. This regulator consumes only 2.5 μW while achieving an efficiency of 66% and an output voltage ripple of less than 5 mV in typical operating conditions, despite the high-impedance nature of the microphone bias voltage. The maximum deliverable power is heavily dependent on the microphone bias characteristics of the host phone, with the voltage quickly dropping below the required 1.8 V as the load increases beyond a few hundred micro-watts. The AudioDAQ system draws 110 μW of the available power, with the remaining power headroom made available for use by analog sensors which is shown in Table 4.1 across a variety of smartphones and feature phones.

The EKG sensor presented in Section III draws 216 μW. While the AudioDAQ system worked on all tested phones, the EKG sensor was only able to successfully work on two of the five phones in Table 4.1. This design was chosen as a strong example application because of its complexity, so this is understandable. With additional design work, it may be possible to further reduce the power consumption of the EKG sensor, making it compatible with a larger number of phones. Figure 4.1 shows the real-time EKG signal obtained by integrating AudioDAQ enabled EKG sensor with mobile phone. A second demonstration board that includes both an ambient light and temperature sensor was designed and prototyped with a power draw of only 58 μW, and could be successfully powered by all tested phones.

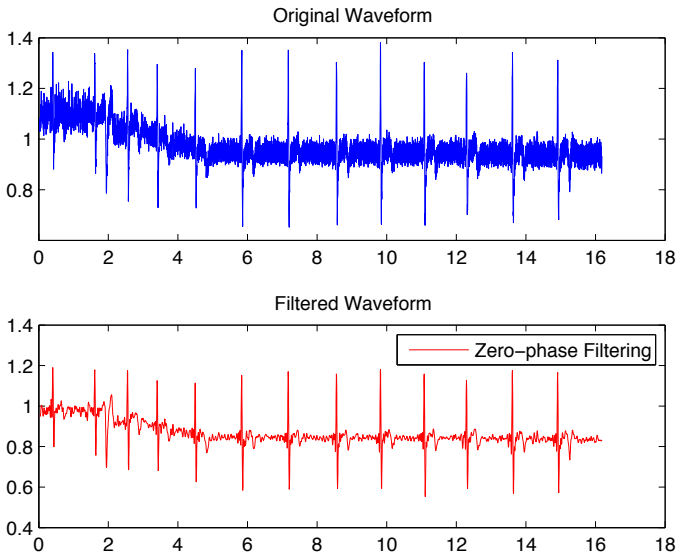


Figure 4.1: (a) Real-time EKG trace obtained by integrating EKG sensor with AudioDAQ platform and plugging the whole system into mobile phone. (b) The EKG trace is passed through zero-phase filtering algorithm to remove the noise.

4.4 Power Draw from Phone

The power draw from the phone is orders of magnitude more than the power delivered to the AudioDAQ board, and after taking into account the battery capacity of the phone, is the sole determiner of total sample period. Table 4.1 presents the total power draw across the tested phones. The iPhone4 had the lowest power draw, most likely due to the compiled nature of its applications and the design focus put on ensuring hardware accelerated multimedia processing is available to the system. On both Google Android and the Nokia feature phones we evaluated, applications are JIT compiled and run in Dalvik or Java virtual machine, which results in more inefficiencies at runtime. The iOS developer's API specifically makes hardware-assisted encoding methods available, and focuses on documenting how to effectively minimize power draw while processing audio. Google Android APIs are not as forthcoming, with hardware-assisted encoding being a platform dependent option, and no

clear documentation on best practices for streamlining audio recording.

To justify our decision to process data server-side, we implemented the signal reconstruction algorithm on iOS and measured the power draw. While running signal reconstruction on the phone and writing the results to storage, we found the phone used 203 mW, which is greater than the 149 mW used when simply recording the signal.

4.5 Signal Recovery and Distortion

AudioDAQ is well-suited for a wide range of analog signals. The majority of analog sensing applications involve reading a slowly-varying DC signal with great accuracy. To characterize the linearity and precision of the system when receiving such a signal, we performed a slow voltage sweep with a period of 1 s from ground to the supply voltage. The results of this sweep are presented in Figure 4.2 and show that AudioDAQ is linear throughout the entire voltage range. The confidence interval was also calculated, and shown to be fairly constant until a voltage above 1.2 V was passed through the system. For applications that require high DC precision, we recommend limiting the voltage range to 0-1.2 V, and taking an average of multiple samples.

To demonstrate AudioDAQ's ability to simultaneously capture multiple signals and to determine if any coupling between the channels does exist, we fed a sinusoidal wave with an amplitude of 1.8 V and a period of 2 s through the system on channel one, while feeding a constant DC signal valued at half the regulated supply voltage through the second channel. The goal of the experiment was to evaluate the reconstructed value of the constant DC signal and see if it varied correspondingly to the sinusoidal wave with time. Figure 4.3 shows the magnitude of the error introduced in the constant DC signal, over the period of

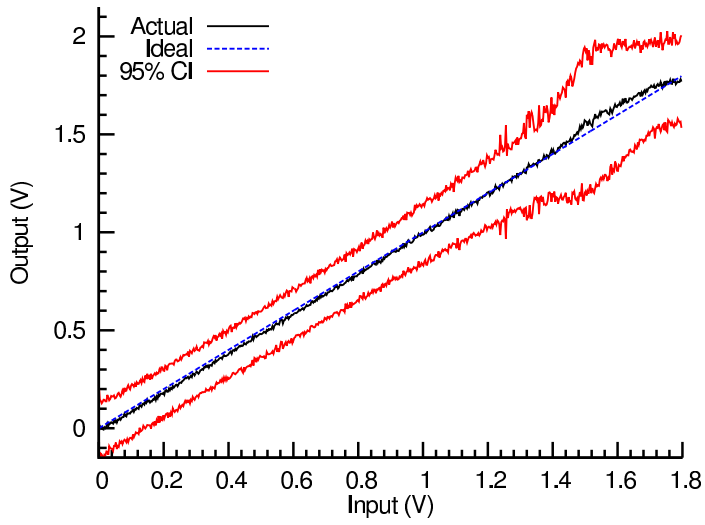


Figure 4.2: A voltage was performed from ground to the reference voltage (1.8V) to determine the linearity of the reconstructed signal across the entire input voltage range. The results of the sweep were collected for 130 runs and the mean was found. A 95 confidence interval was computed. For applications where accuracy of the reconstructed signal is critical, keeping the range of the sensor between 0-1.2 V is suggested to minimize the variance in sampled data.

the sinusoidal wave, averaged across 23 periods. The amplitude of the error never exceeds 40 mV, demonstrating that while the channels are coupled, the coupling is minimal.

When evaluating our system with the EKG demonstration board, the 300 Hz sampling rate was sufficient to capture all the important features of the EKG waveform and to provide a faithful signal reconstruction. Software was developed to extract the subject's pulse rate from the raw data and plot it alongside the EKG waveform.

4.6 Marginal Cost

The hardware in AudioDAQ exclusively uses analog components that have been in mass production for years. These components are not high-performance devices and do not have significant costs associated with them, making AudioDAQ extremely inexpensive. Table B.1 shows the price breakdown for AudioDAQ. The total marginal cost is $5 for 10 k units. The cost includes the prices for headset plug and printed circuit board.

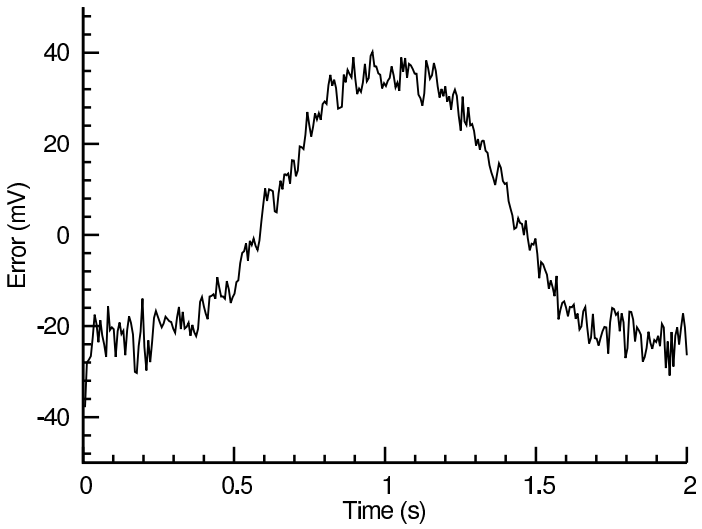


Figure 4.3: The error component of a constant-valued 0.9 V DC signal when multiplexed with a sinusoidal wave with a period of two seconds. The error was calculated from the mean value over 23 sinusoidal periods and is graphed over the period of the wave. The signals are weakly dependent on each other, with a maximum observed average error of 40 mV seen at the peak of the wave.

Other costs are harder to quantify. Supporting a server to store and process the audio data from AudioDAQ devices can create overhead and costs, but the algorithm is reasonably efficient, running at approximately 20x real-time on a single core modern machine, allowing for a single server to support a large number of AudioDAQ devices sporadically sending data files.

The lack of development time required to write custom software for each supported platform will result in significant cost savings, and reduced support and ongoing maintenance, which can largely offset any cost incurred in maintaining a single server for processing.

CHAPTER V

Discussion

In this section, we focus on the design challenges faced when designing a peripheral for the mobile phone's headset port. The fundamental coupling between power and data when transferred together over the microphone channel and the constraints of the channel itself significantly affect the design of AudioDAQ. As AudioDAQ was designed, a clear trade-off between analog signal reconstruction accuracy and maximum power delivery emerged.

5.1 Limitations

The AudioDAQ system is limited by the channel used for data transmission and energy harvesting. The microphone line is not designed to support an arbitrary hardware peripheral. It has been tightly optimized to achieve the lowest possible cost while efficiently providing the minimal power and bandwidth required to extract a voice signal from a condenser microphone.

During the design process, the significance of the trade-off between creating a stronger design by adding additional active components that could produce cleaner signals or filter better, and keeping the system power draw low to provide more headroom for peripheral sensors became clear. Requiring an extremely low energy consumption, on the order of

hundreds of micro-joules, caused us to rethink the design and forced us to adopt simpler passive techniques to solve filtering and signal conditioning issues, at the expense of the fidelity of the signal itself.

5.1.1 Power Budget

We can harvest a maximum power between 0.29 mW to 0.81 mW from the microphone channel, dependent on the specific phone. This power is sufficient to drive many low-power circuits, including the simple analog circuitry on the AudioDAQ board. Since there is a large class of sensing tasks that require small, passive transducers and relatively simple amplifying circuitry, this does not severely limit the usefulness of the system within the chosen design space. However, some sensors, including many used for air quality and gas detection applications, require large amount of current to drive heaters or pulsed infrared lamps and cannot be powered by AudioDAQ.

The energy that can be harvested is constrained by both the high impedance of the microphone bias voltage, set by an internal resistor ($R1$) that can be seen in Figure 2.2, and by the design of the power supply. The ultra-low power requirements of system are below the optimized design space of many commercially available power supplies, making it difficult to find devices that can deliver reasonable efficiencies. This problem is amplified by the fact that we can tolerate very little ripple on the input supply rail. Noise introduced into the supply voltage is not a commonly minimized design parameter. The goal of most power supplies is to provide a stable system output voltage, even if significant noise is introduced into the input rail. In our system a quiet noise floor on the input side of the regulator is more important than having an extremely stable output voltage. The multiplexed signal is

overlaid directly on top of the microphone bias voltage and is of an extremely low amplitude. Any noise on the microphone line adds significant ripple to this signal and quickly makes it unrecoverable.

Adding an external power source is a simple way to avoid the design challenges of harvesting energy from the mobile phone, which, as indicated from this work and previous work, is challenging to get right and presents a number of fundamental design problems. In early prototype versions of AudioDAQ, we worked on the viability of modulating a signal across the audio channel before power supply designs were considered, using an external battery to power sensors. Environmental harvesting devices such as solar cells are viable solutions, however they would add bulk to the AudioDAQ system without delivering significantly more energy, making it worthwhile to work on harvesting energy from headset port.

Harvesting energy from the audio driver output is another viable solution, and was employed successfully in HiJack. The amount of power deliverable from the audio driver output is much greater than what can be obtained from the microphone bias voltage, as Figure 2.3 shows. Delivering power from this source however requires custom software to be written for each feature and smartphone platform, sacrificing the universalness of AudioDAQ. This approach also increases the power draw of phone significantly.

5.1.2 Coupling of Power and Data

Carrying over from the power budget limitations, there is a seemingly fundamental coupling between the amount of power that can be delivered and the magnitude of noise introduced in the data channel. The voltage source for powering our system has an extremely

high impedance. The measured value for the biasing resistor that sits between the microphone bias voltage and the microphone channel is between 2 $k\Omega$ to 2.5 $k\Omega$. This makes the voltage source heavily dependent on the system load. Transient spikes caused by the switching of digital components can create large ripples in the input voltage. This problem is further exasperated by the extremely small amplitude of the signal (10 mV peak-to-peak) required to successfully pass through the audio front end of the phone.

The obvious solution is to add additional decoupling capacitance to the circuit to smooth the noise introduced by switching circuits. Placement of these decoupling capacitors is critical to maintaining the multiplexed signal. If the capacitors are not properly isolated from the microphone channel, they will effectively create a low-pass filter and attenuate the signal. To isolate the decoupling capacitors from the microphone channel, a low-pass filter as seen in Figure 2.2 was added, at the cost of increasing resistance and sacrificing total power output.

5.1.3 Sampling Rate

The sampling frequency for acquiring the analog sensor data is limited by the pass-band characteristics of the microphone channel. The upper limit of the audio pass-band bounds the maximum rate of sampling that can be achieved. The microphone channel is generally designed to support sampling frequencies that support capturing of signals within the audible range of human hearing (20 Hz-20 kHz).

The compression scheme used in many mobile phones further limits the range of captured frequencies. While the iPhone uses a lossless encoding scheme, many other phones do not. Some phones employ adaptive multirate encoding (AMR) to compress voice data,

which has a sampling frequency of 8 kHz and is highly optimized for capturing the vocal range of the audio spectrum (80 Hz-1100 Hz). The AudioDAQ system is optimized for this frequency range, with the multiplexer switching at a frequency of 1.2 kHz, which sits at the far right end of the vocal frequency range.

Reconstructing the original signal faces some Nyquist imposed demands on the number of samples required as well. The lowest frequency compression schemes operate at a sampling rate of 8 kHz, giving us approximately 7 samples for each multiplexer signal, which is the lower limit of the number of samples required by the reconstruction algorithm to properly recover the original signal. Operating at much higher frequency would first stress the limits of the recovery algorithm, and finally push against the Nyquist limited number of samples required to recover the multiplexer waveform at all.

On smartphone platforms, which are often capable of capturing audio at rates such as 44.1 kHz on the iPhone and Android devices that can capture the entire audible spectrum, the switching frequency can be increased. However, doing so will proportionally increase the power consumed by the system and make it unusable by phones with slower sampling rates.

To preserve the universalness of AudioDAQ, we keep power draw low, and to ensure that the data transfer is optimized for compression schemes that favor vocal audio, which are likely to be used on mobile devices, we keep the switching frequency at 1.2 kHz. Because recovering a single analog value requires four values, as explained in Section 2.5, the effective sampling frequency is 300 Hz which is equal to the switching frequency divided by a factor of four. Capturing multiple channels requires time-division multiplexing each signal which further reduces the sampling frequency.

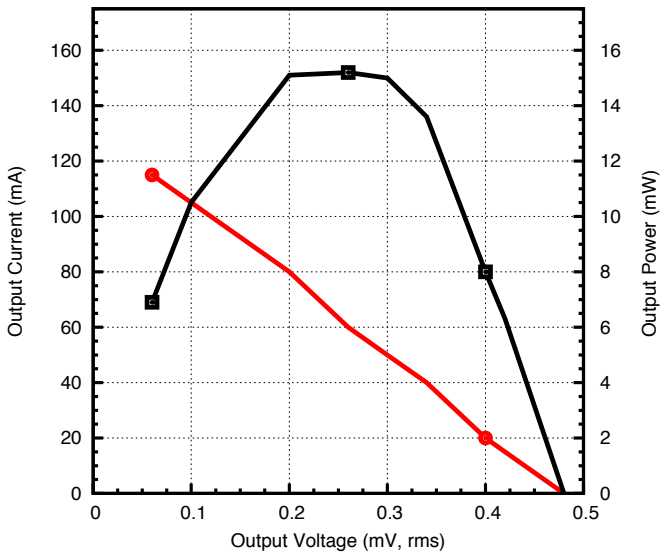


Figure 5.1: Available power from iPhone headset jack

5.2 Synchronous Rectification

In this section, we explore the problem of efficiently harvesting energy from the left and right audio channels of the headset port. In order to find the maximum power that can be efficiently harvested, the following experiment is set up. A variable load is connected between the right audio channel line and the common of the headset port. The load is varied from 0 Ω to 15 KΩ and the output voltage and load current is measured at several points. From the data collected, the power transfer curve for the right audio channel is plotted. For example, figure 5.1 is the plot of the power transfer curve for an iPhone. We can see that the open circuit voltage is less than 500 mV and maximum power point voltage occurs at 240 mV. These voltages are far below the typical starting voltages of switching regulators (generally between 800 mV and 900 mV). Even the ultra-low voltage step-up DC-DC converter like Seiko S-882Z [31] requires a start up voltage of 300 mV. Rectification losses can be high in both high-power and low-voltage systems. In case of

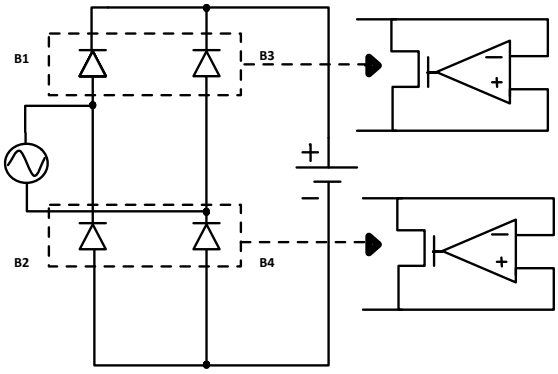


Figure 5.2: Synchronous rectifier circuit [20]

iPhone, maximum power transfer is achieved at RMS current of 66 mA. Even when a low forward voltage drop Schottky diode like the DFLS120L [21] is used for rectification, a 200 mV forward voltage drop occurs. This means that 80% of the power is lost during rectification, and only 20% can be delivered to the load. The conduction loss of diode rectifier contributes significantly to the overall power loss in a power supply, especially in low output voltage applications. Synchronous rectification is sometimes used to reduce losses where a FET is used instead of a diode. As seen in section 2.1, we have a sufficiently high microphone bias voltage which can be used to drive the gate voltage to turn ON the FET switch and other active circuitry. This approach allows us to achieve efficient rectification as compared to diode rectification.

5.2.1 Design

Diode-based rectification often results in high-voltage drop and power loss in low voltage systems. A synchronous rectifier may be used to achieve higher efficiency in comparison to a diode rectifier. A diode rectifier can be converted into a synchronous rectifier by simply replacing each diode with a block of FET integrated with a comparator.

Figure 5.2 shows the basic design of a synchronous rectifier in bridge configuration.

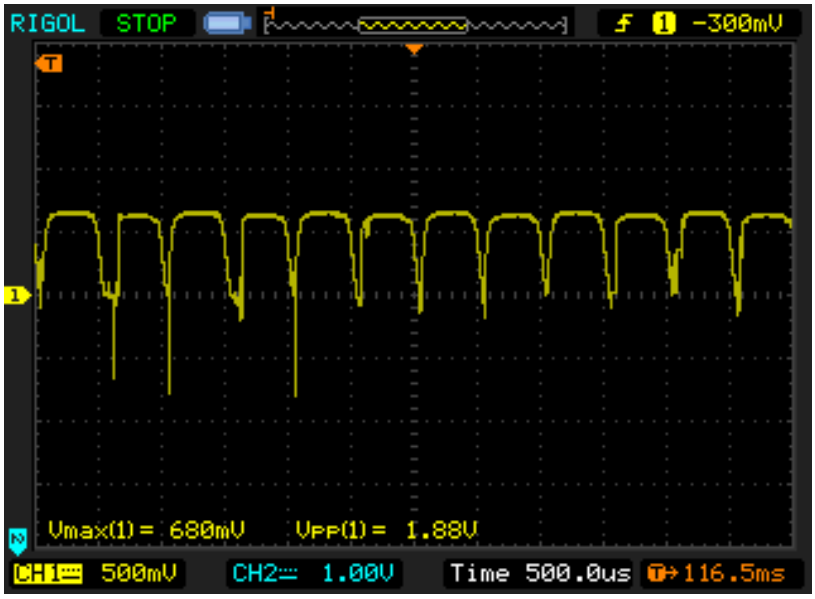


Figure 5.3: Synchronous rectifier output before passing through a smoothing capacitor

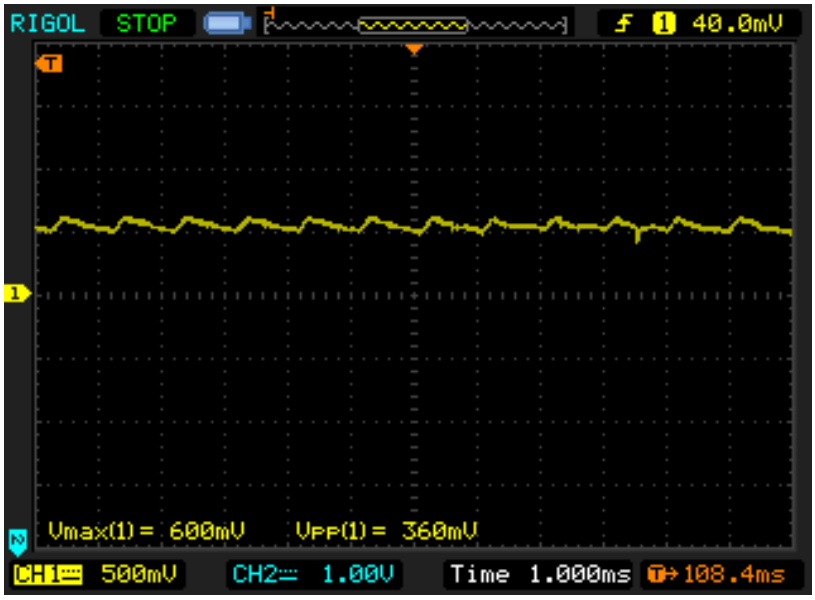


Figure 5.4: Rectified output after passing through a smoothing capacitor

It uses active devices such as op-amps and feedback to efficiently rectify the AC voltage. The FETs in this configuration are controlled by the comparators. These comparators are powered from the microphone bias voltage. The gate of the FET is connected to the comparator's output. The comparator evaluates the voltage across the FET continuously to

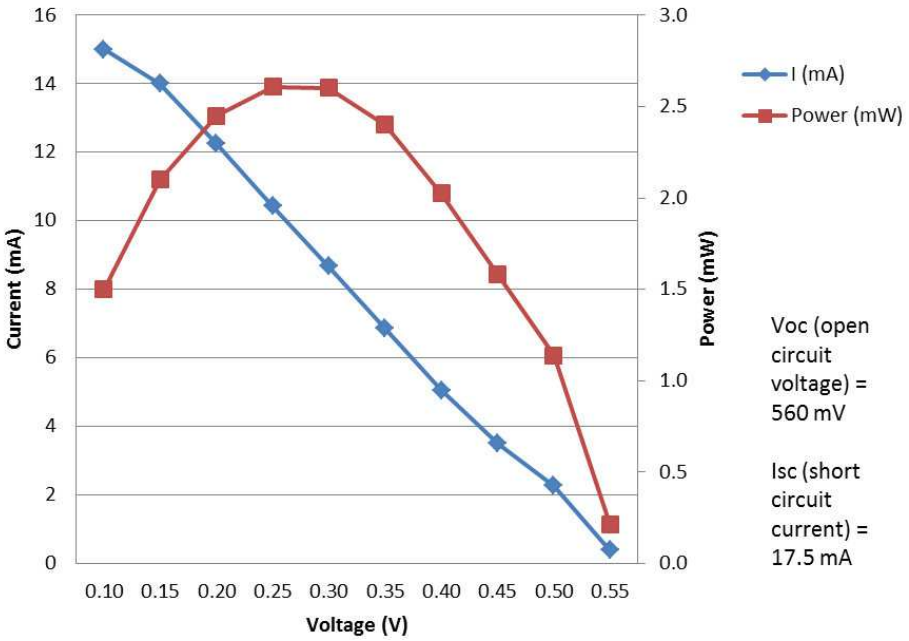


Figure 5.5: P-I-V curve at syncronous rectifier output in Nexus One

turns it ON and OFF accordingly. During the positive half cycle of the sinusoidal wave at the input of the synchronous rectifier, FETs B1 and B4 conduct and the current flows from the positive supply to negative supply. During the negative half cycle, FETs B2 and B3 conduct and the current reverses its direction. We achieved a power transfer efficiency of 70% by using this rectifier.

Figure 5.3 shows the full wave rectified output for a input of 1.64 V pk-pk audio tone of 1 kHz frequency. As seen in the figure, there is still some noise at the rectifier output. This noise contributes to the fact that we were testing our circuit on breadboard. Figure 5.4 shows the DC signal of 540 mV after passing through the smoothing capacitor.

In order to eveluate, if the maximum power transferred from synchronous rectifier output is dependent on the frequency of audio tone generated from right audio channel, we varied the frequency from 500 Hz to 22 kHz in steps of 1 kHz. The maximum power transferred from the audio headset port comes out to be independent of frequency. The

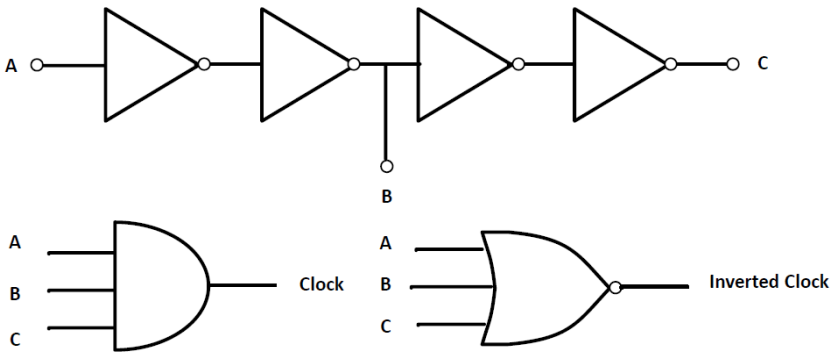


Figure 5.6: Non-overlapping clocks for 1:2 switched capacitor circuit

maximum power transfer after passing through the synchronous rectifier and smoothing capacitor occurs at 280 mV when delivering 9.2 mA for a load impedance of 30.4 ohm in case of Nexus One (Figure 5.5).

5.2.2 Voltage Converter

After the synchronous rectification, a 1:2 switched capacitor converter is used to step up the voltage to a DC level which meets the start up voltage requirements of commercially available switching regulators. We need non-overlapping clocks to ensure the correct operation of this circuit. If the clocks overlap, all the FETs may conduct momentarily which may result in glitches and voltage drop at circuit output. A low-power RC oscillator generates the clock signal A. As shown in Figure 5.6, this clock signal is fed through four inverters in series and the non-overlapping and non-adjacent subsets of this clock signal are picked off. The propagation delay through the AND and the NOR gates must be same. Figure 5.8 and Figure 5.9 show the rising and falling transitions of the two non-overlapping clocks.

Comparative analysis of a number of switched capacitor converter topologies are de-

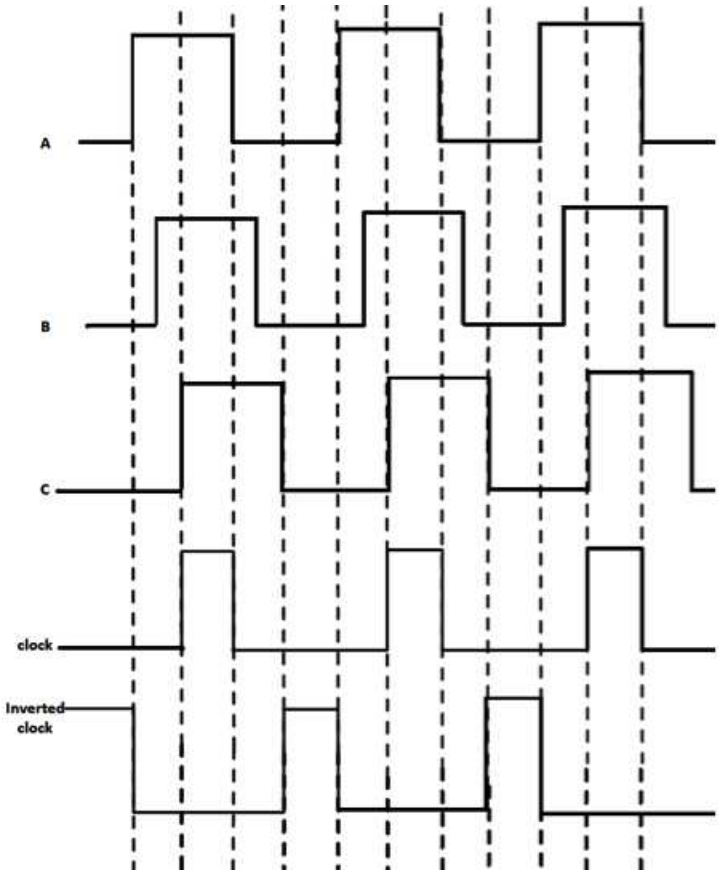


Figure 5.7: Operation of non-overlapping clocks circuit

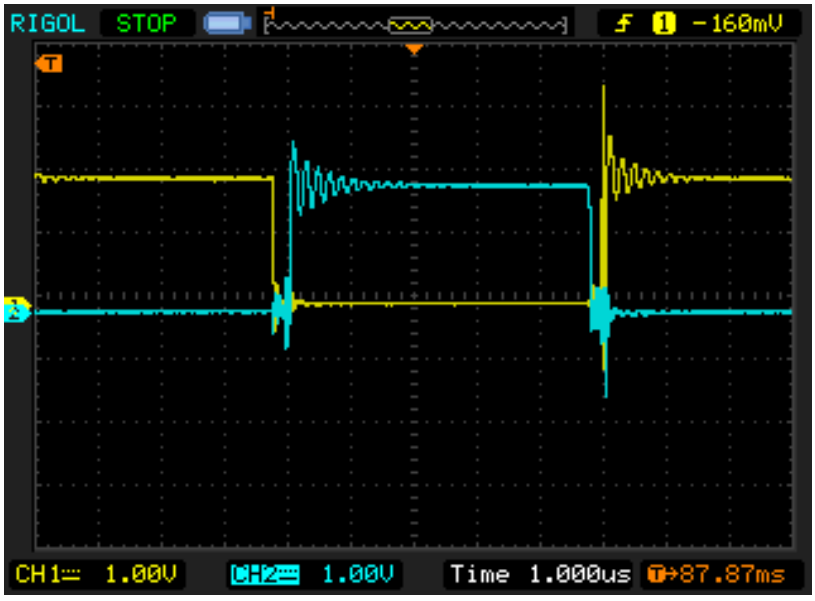


Figure 5.8: Oscilloscope capture of clock generation during rising transition

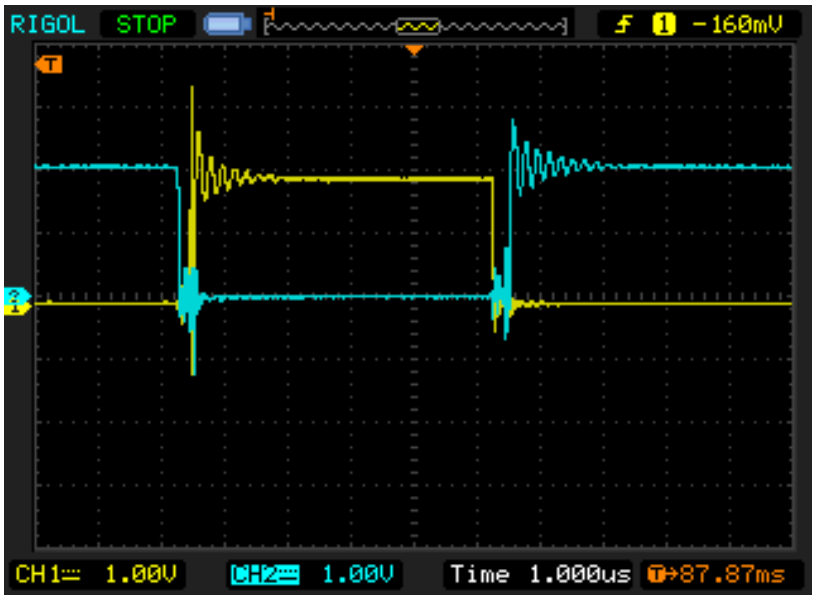


Figure 5.9: Oscilloscope capture of clock generation during falling transition

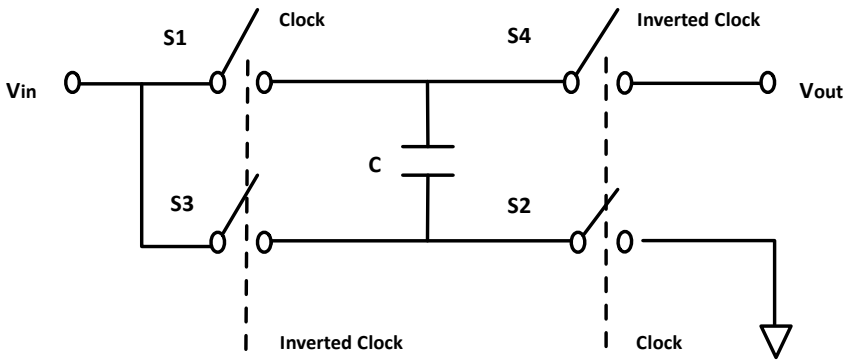


Figure 5.10: 1:2 Switched capacitor converter [17]

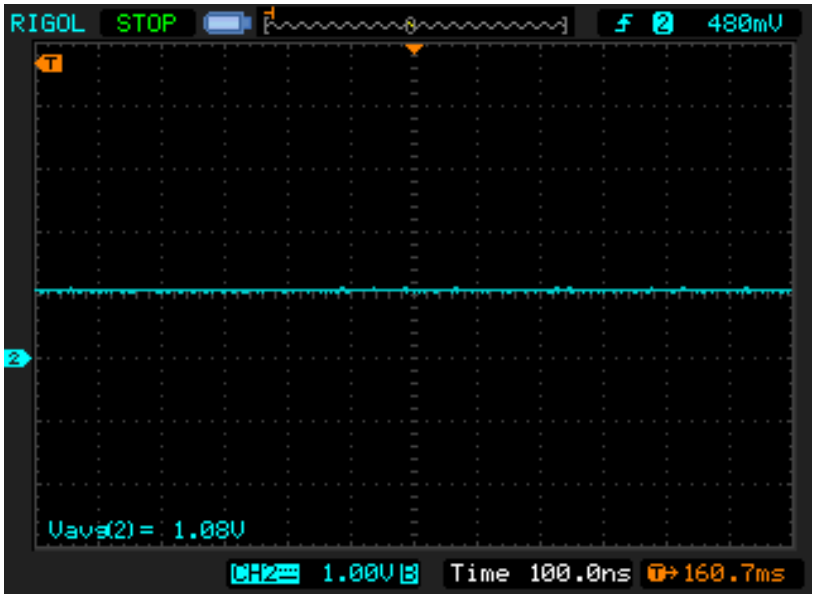


Figure 5.11: Oscilloscope capture of voltage doubler output

scribed in [18], [19]. We use the doubler topology from [18] as shown in Figure 5.4. During phase 1, switches S1 and S2 are ON and charge the capacitor C to the input voltage. During phase 2, switches S3 and S4 are ON while S1 and S2 are turned OFF. As a result, the output capacitor gets charged to the sum of input voltage and voltage at capacitor C. Therefore, the voltage we get at the circuit output is double the input voltage. We fed the 540 mV DC signal from output of the synchronous rectifier into the input of the 1:2 switched capacitor converter described above. Four switches are implemented using nfets whose ON resistance is 45 mΩ. Clock is generated from the low-power oscillator with a frequency of 50 kHz. Capacitors values are choose to meet the charging and level shift requirements during each clock cycle. We get a a doubled voltage of 1.08 V at the circuit's output.

The voltage doubler output is high enough to meet the start up voltage requirements of commercially available boost regulators. The output from the switched capacitor is fed to

the boost converter LT1615 [15] which steps up this voltage to a stable supply voltage of 2.7 V. The entire schematic and layout of the circuit of modulation scheme described in section 2.2 with synchronous rectification is shown in Figure A.9 and Figure A.10.

5.3 Future Work

By far the biggest limitation of the AudioDAQ system is the relatively small amount of power that can be delivered, and the effect it has on the fidelity of the recovered signal. The current power supply design involves a passive first order filter. This filter works but we do not claim it is optimal. Noise from the power supply regulation loop and from digital powered circuitry account almost entirely for the error in the recovered signal. A more effective filter could reduce this noise and result in much better analog signal recovery. An active filter could yield a much better response, but would introduce additional complexity into the design, and require sacrificing a portion of the power budget itself. This filter would be required to serve double duty by both reducing the ripple in the microphone channel created by the power supply and digital switching circuitry, and preventing the signal generated by the multiplexer from in turn creating feedback loops and destabilizing the circuitry generating it.

Another area of future improvement involves the power supply itself. Keeping the microphone bias voltage above the threshold voltage required to drive typical commercial logic gates (1.8 V) is not optimal. The power delivery of the microphone bias voltage could be improved by increasing the load and allowing the voltage to drop lower. Boosting this voltage back up by introducing a switching supply could increase the maximum power delivery capabilities of the system. Early designs of AudioDAQ utilized such a switch-

ing supply, but the noise introduced into the microphone channel was unacceptable and severely affected the quality of the recovered analog signal. Due to the small amounts of power required by the system, almost all commercially available boost regulators will operate in discontinuous mode to conserve power. This resulted in large transient current spikes when the regulator switched, which, due to the high-impedance nature of the microphone bias voltage supply, causes a large voltage spike. It was difficult to create designs where the regulators fell into a mode of stable continuous operation, and even when stability was achieved, the analog signal was severely compromised. Designing an efficient switching supply without this noise problem could improve the AudioDAQ design and provide greater available power for sensing peripherals.

Power supply design traditionally focuses on reducing output ripple. Keeping input ripple low, especially when designing with high-impedance voltage source is not a commonly desired design parameter, and few existing solutions target this goal.

Finally, combining all of this circuitry into a single piece of silicon could solve a number of the present design challenges. Creating a chip small enough to be injection-molded directly in the headset port plug itself would miniaturize the peripheral size and further expand the capabilities of the system.

CHAPTER VI

Related Work

The AudioDAQ architecture, design, and implementation have been influenced by a large body of prior commercial products and academic projects focused on developing mobile phone peripherals [9], [28], [22], [16]. This body of work includes audio headset peripherals, wired/wireless peripherals, and vital signs monitors (e.g. EKG) [2], [3], [1]. AudioDAQ is motivated in part by the opportunity to leverage the billions of already-deployed mobile phones in the world today (many of which are in developing regions). Moreover, energy harvesting has been of considerable research interest in recent years. There have been many designs which explore radio frequency [4], piezo vibration [42], [30], outdoor solar [41] and indoor lighting [13].

6.1 Audio Headset Peripherals

HiJack is an open source energy harvesting platform for iOS devices [43]. It harvests power from the audio output driver, instead of the microphone bias voltage, making it capable of providing 7.4 mW to a load, with an efficiency of 47%. It allows for bi-directional communication between the phone and an on board MSP430 microcontroller at 8.82 kbps using Manchester encoding. AudioDAQ improves on HiJack by extending the sampling

Figure 6.1: Square Card Reader

period, a direct result of simplifying the design by trading the flexibility of a microcontroller for a more power-efficient analog solution after recognizing that such a system is adequate for a large class of sensing applications.

Similar to HiJack, the RedEye Mini also uses the audio output driver for power, driving an LED that acts as a TV remote control [38]. It integrates an upgradeable microcontroller and similar power harvesting circuitry. It provides a significantly higher amount of system power by utilizing both left and right audio output channels, but as a result causes the phone to consume even more energy.

Square is a credit card reader which interfaces to Android and iOS devices through the 3.5 mm audio jack to allow for mobile payments [32]. Based on our own tear down, we found it detects the voltage spikes (30 mV peak-to-peak) generated by oscillating magnetic field created when a credit card is pushed through the magnetic read head. It has a robust algorithm to extract the data and uses error correction schemes to improve detection rates.

Figure 6.2: RedEye Mini

While the Square card reader is a completely passive device, requiring no power from the headset port, we borrow from it the idea that a simple hardware design can be augmented with robust algorithms to achieve complex functionality.

Table 6.1 compares power draw of HiJack, the RedEye Mini, and AudioDAQ, and estimated the sample period of each on an iPhone4. Although both HiJack and the RedEye Mini can provide significantly more output power, it is provided at a great cost to overall system power draw, and is not necessary for a large class of sensing applications and not practical for long-term collection.

Other audio headset peripherals include My TV Remote [29], a simple purpose-built television remote control lacking the complexity of the RedEye Mini, SwitchScience SoftModem [33], a general purpose development board for implementing communication similar to that found in HiJack, iData [5], a PIC-based platform with limitedly available technical details that most likely uses an external power supply, and H4I [27], a development platform also using communications similar to those found in HiJack, offering integration

Peripheral Device	Run Power	Sample Power	Sample Time
AudioDAQ	0.109 mW	149 mW	35 hr
HiJack	0.247 mW	346 mW	15 hr
RedEye mini	0.493 mW	493 mW	10 hr

Table 6.1: Power draw breakdown of RedEye mini, HiJack, and AudioDAQ. For sensors requiring small amounts of power AudioDAQ allows for extended sampling periods.

with existing commercial devices. All these peripherals prove that there is interest in the headset port, however they differ significantly from AudioDAQ. My TV Remote is much simpler than AudioDAQ, only allowing limited control of an infrared transmitting diode directly connected to the audio output channel. The SoftModem, iData, and H4I all provide bi-directional communication but none are self-power or optimized for extended data collection.

6.2 Non-Headset Peripherals

Several designs have also been created to enable external sensor peripherals to communicate with mobile phones that do not make use of headset port, instead opting to use Bluetooth, USB or other novel interface methods.

FoneAstra is a sensing platform developed by researchers at University of Washington in collaboration with Microsoft Research [11]. It is a low-cost programmable hardware plug-in that enables sensing applications on low-tier mobile phones. The system consists of an ARM7 microprocessor that connects and communicates to phone using serial I/O. It can be powered either from mobile phone's battery or by using a separate external battery.

Urban Sensing has been of great interest in sensor network community in recent years [9], [**?**], [23]. The N-SMARTS project aims to build hardware and software platforms to enable data collection from real-time sensors using mobile phones [28]. However, their

Figure 6.3: FoneAstra

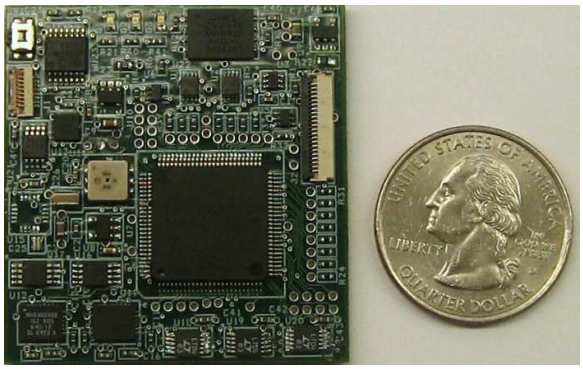


Figure 6.4: Little Rock

work utilizes the GPS and Bluetooth capabilities of cell phones.

Little Rock proposes a new architecture for mobile phones where the sampling and processing of sensor data is offloaded to a secondary low-power microprocessor connected directly to the primary processor [26]. This approach allows the primary processor to go into sleep mode, enabling long-term sensing. If successfully integrated into new mobile phones Little Rock would achieve longer sample intervals with much greater efficiencies, however AudioDAQ has the advantage of working with the phones of today, requiring no hardware modifications.

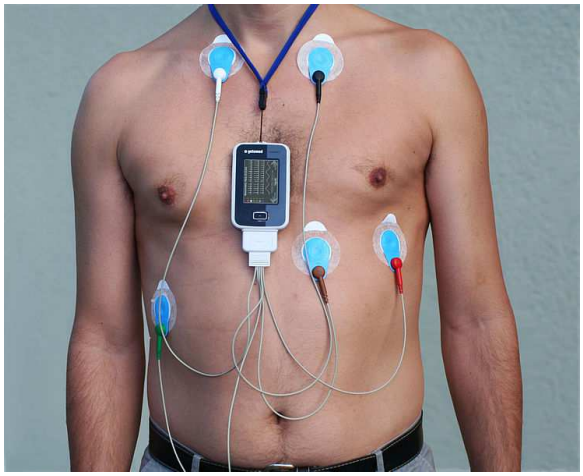


Figure 6.5: Holter Monitor

6.3 EKG Monitors

There has been considerable research in making EKG monitors increasingly low-power, wearable and portable [10]. Our work builds on the prior efforts in this area and introduces a phone-powered EKG monitor, which harvests energy from the microphone channel of the phone using AudioDAQ and consumes small amount of power while using only commercially available components.

An effort by L. Fay, V. Misra, and R. Sarpeshkar resulted in a micropower EKG amplifier [14]. This amplifier requires a remarkably small 2.8 μW, far lower than the power draw of the sensor we present, however it is implemented using chip-level fabrication, making it expensive if not mass produced. The sensor we present can be built using completely off-the-shelf components and is assembled on a standard PCB.

Holter monitor [12] is an EKG recording device, which continuously monitors heart activity of a patient over a period of 24 to 48 hours. It was invented in 1949 but its clinical use started in early 1960s. It consists of three to eight electrodes that are placed on the patient's chest and are connected to recording equipment that he carries around in his

Figure 6.6: Polar Heartstrap

pocket or around his neck. This device is battery operated and continuously stores the cardiac activity on an electronic media. The patient is also required to keep a log of his activities during this recording period. At the end, the cardiac is data is then analyzed on a computer by a cardiologist. However, the success of EKG analysis depends upon the signal quality. The signal quality in turn depends on whether the electrodes are properly attached to the patient's body and the patient not engaging in any strenuous activity, which could result in distortion of EKG data. Only advanced and expensive Holter monitors are able to display the real-time EKG signal or transmit the data over internet. Since, heart abnormalities occur only a few times in a year, they are unlikely to be captured during a 24 to 48-hour period. Hence, an EKG sensor, which provides long-term heart monitoring and has the capability to display and transmit the EKG data over internet is more effective for early detection and timely diagnosis of cardiac problems. Our AudioDAQ enabled square-inch, battery-free EKG monitor tries to incorporate all these features. It can also optionally transmit the collected EKG data to the cloud for storage or real-time remote display where the doctor can analyze the data and take fast preventive action in case of an emergency.

The wireless EKG system built by T. R. F. Fulford-Jones, G.-Y. Wei, and M. Welsh [34] can be attached to a Mica2 wireless mote and is built using commercially available components, but has a much higher power draw than our sensor, using 60 mW of power.

The Polar Heartstrap is a popular wearable heart rate monitor [25]. It consists of a chest strap (transmitter) which includes electrodes and cardiac data acquisition circuitry. The strap wirelessly sends cardiac data measurements to receivers. Integrated circuits exist to receive and decode this signal, although they are power hungry, using 100 mW to receive signals. These monitors do not provide full electrocardiograms, only the heart-rate and consume significantly more power.

CHAPTER VII

Conclusion

We present AudioDAQ, a new platform which harvests energy from the microphone bias signal of a mobile phone, provides both a power and communications channel to external sensor peripherals, and enables a new class of sensing applications that require extended periods of data collection. We envision that with the AudioDAQ platform, it will soon be possible to design peripherals for feature phones with a level of functionality that until now has been limited to smartphones. By using the microphone line for power and communication, and using the phone's built in voice memo application for data capture, we dramatically lower the barrier to entry for phone-centric continuous sensor data collection on all phones. AudioDAQ achieves this feat by using a simple, ubiquitous, and universal hardware interface, and eliminating the custom software development traditionally required to achieve similar levels of functionality. We show the feasibility of our approach with a low-cost, low-power EKG monitor based on the AudioDAQ architecture that captures cardiac signals continuously for extended periods of time, and delivers this data to the cloud for processing, storage, and visualization. Thus, our work shows how to leverage the billions of already deployed mobile phones as easy-to-use data collection devices.

APPENDICES

APPENDIX A

Schematics

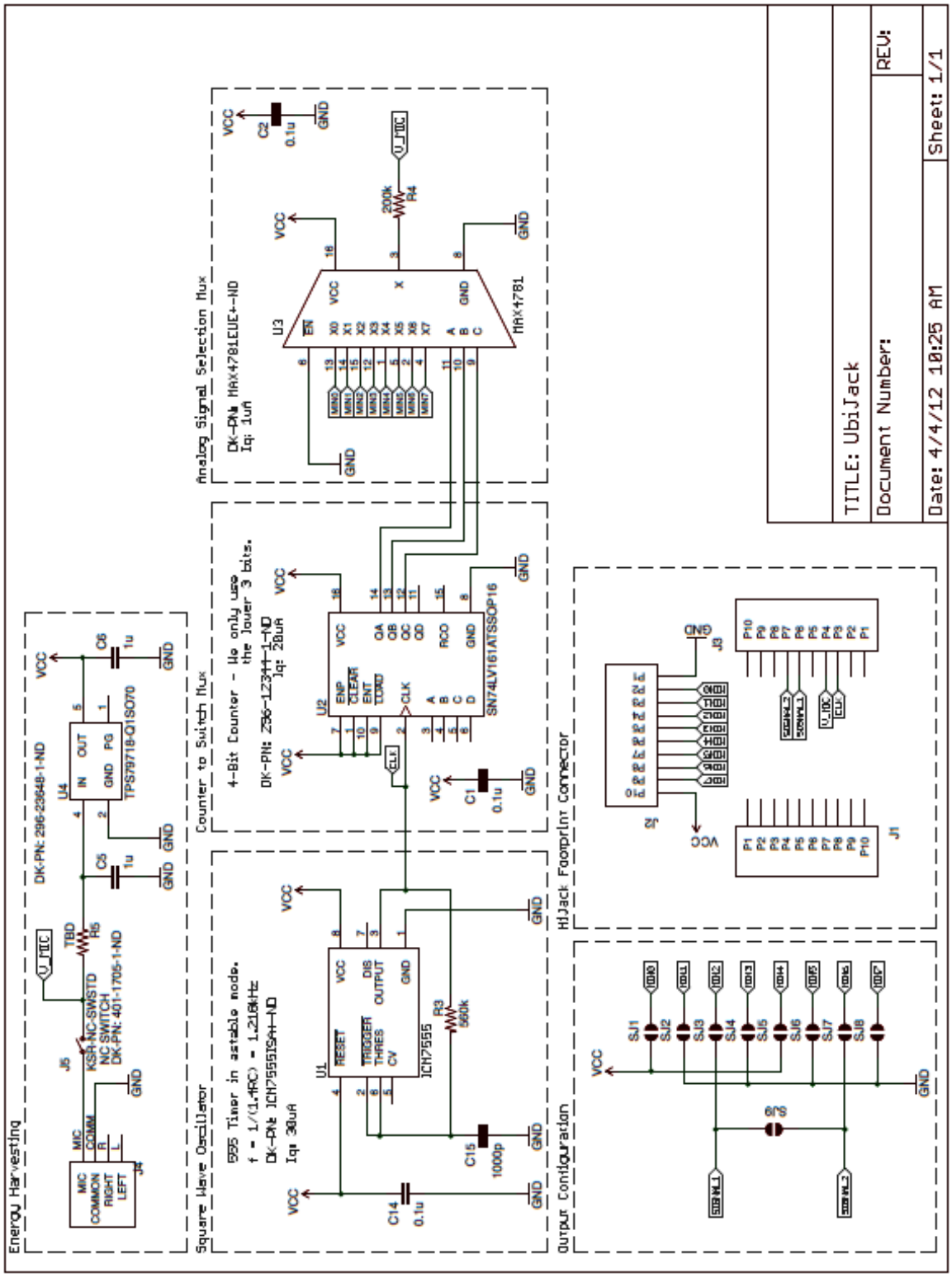


Figure A.**1:** AudioDAQ schematic

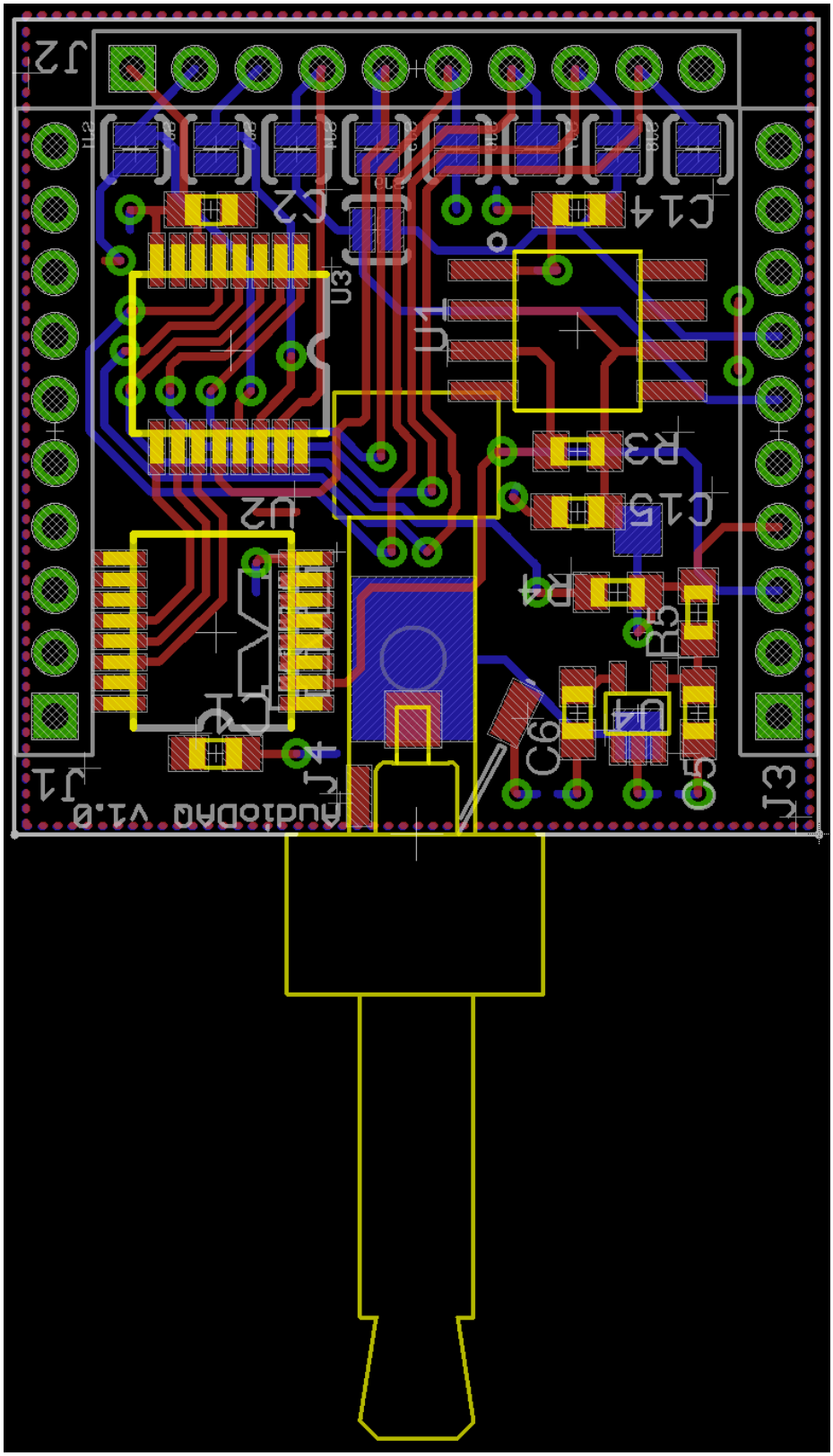


Figure A.**2:** AudioDAQ layout

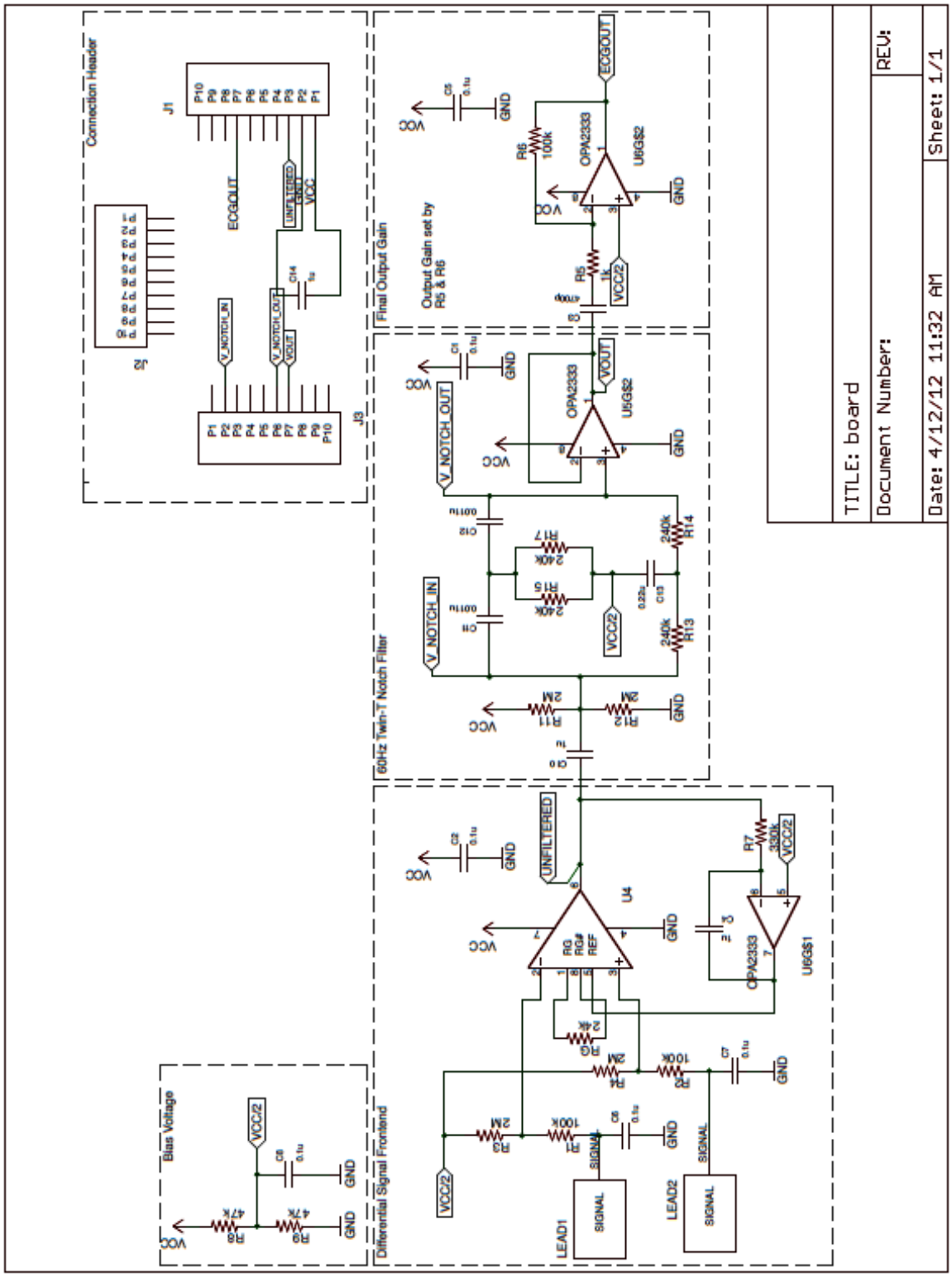


Figure A.**3:** 2-lead EKG sensor schematic

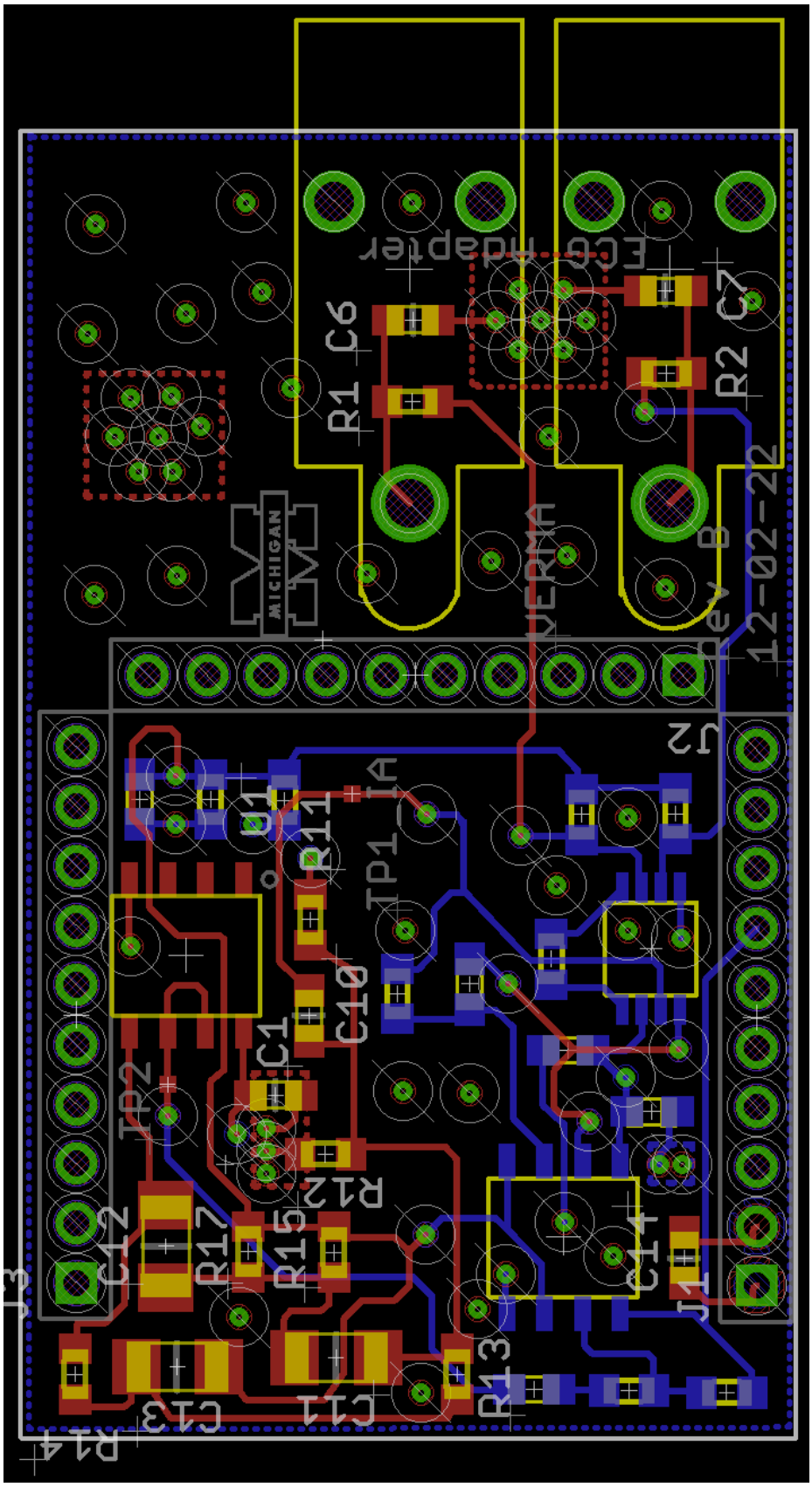


Figure A.4: 2-lead EKG sensor layout

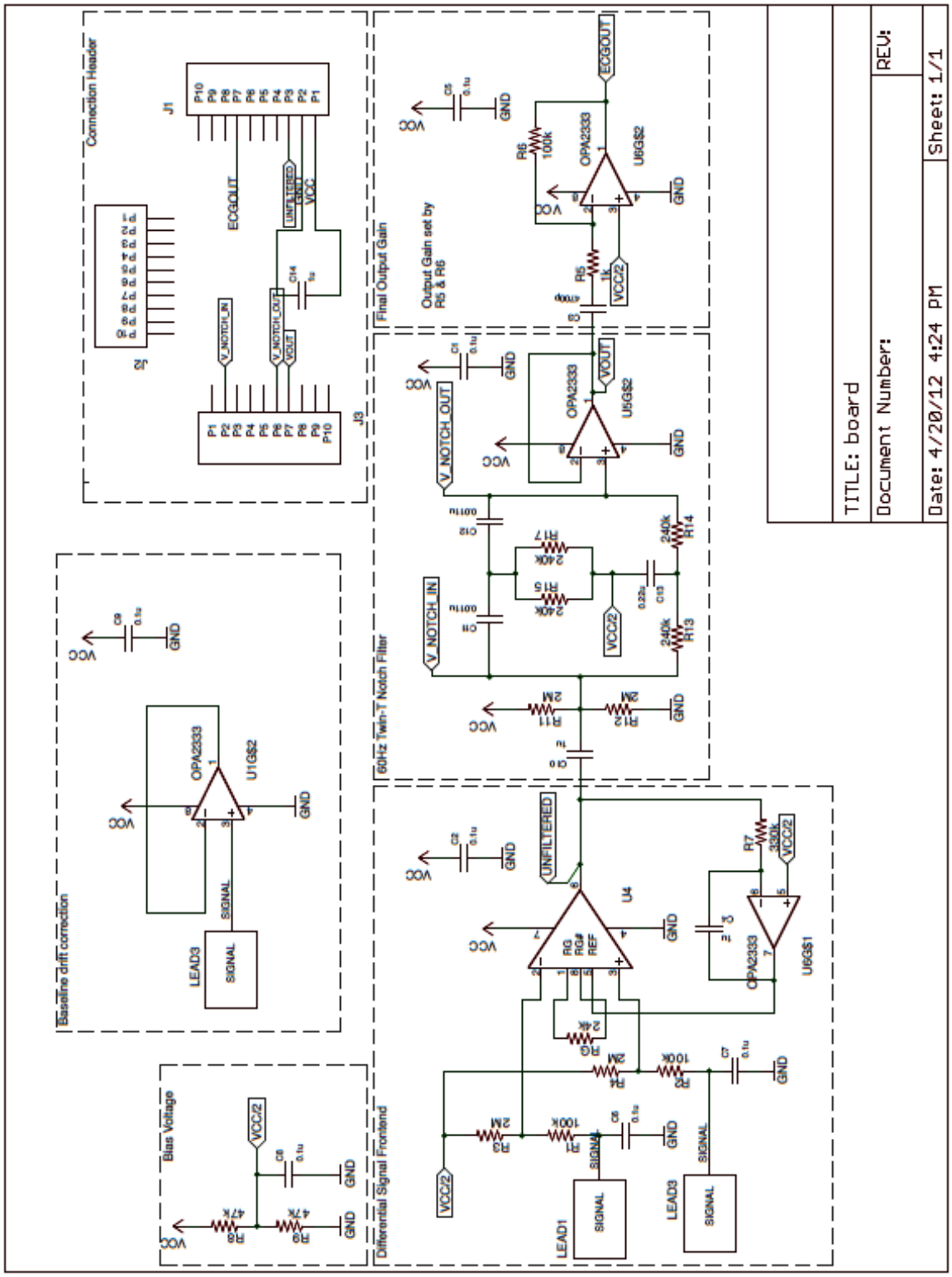


Figure A.**5:** 3-lead EKG sensor schematic

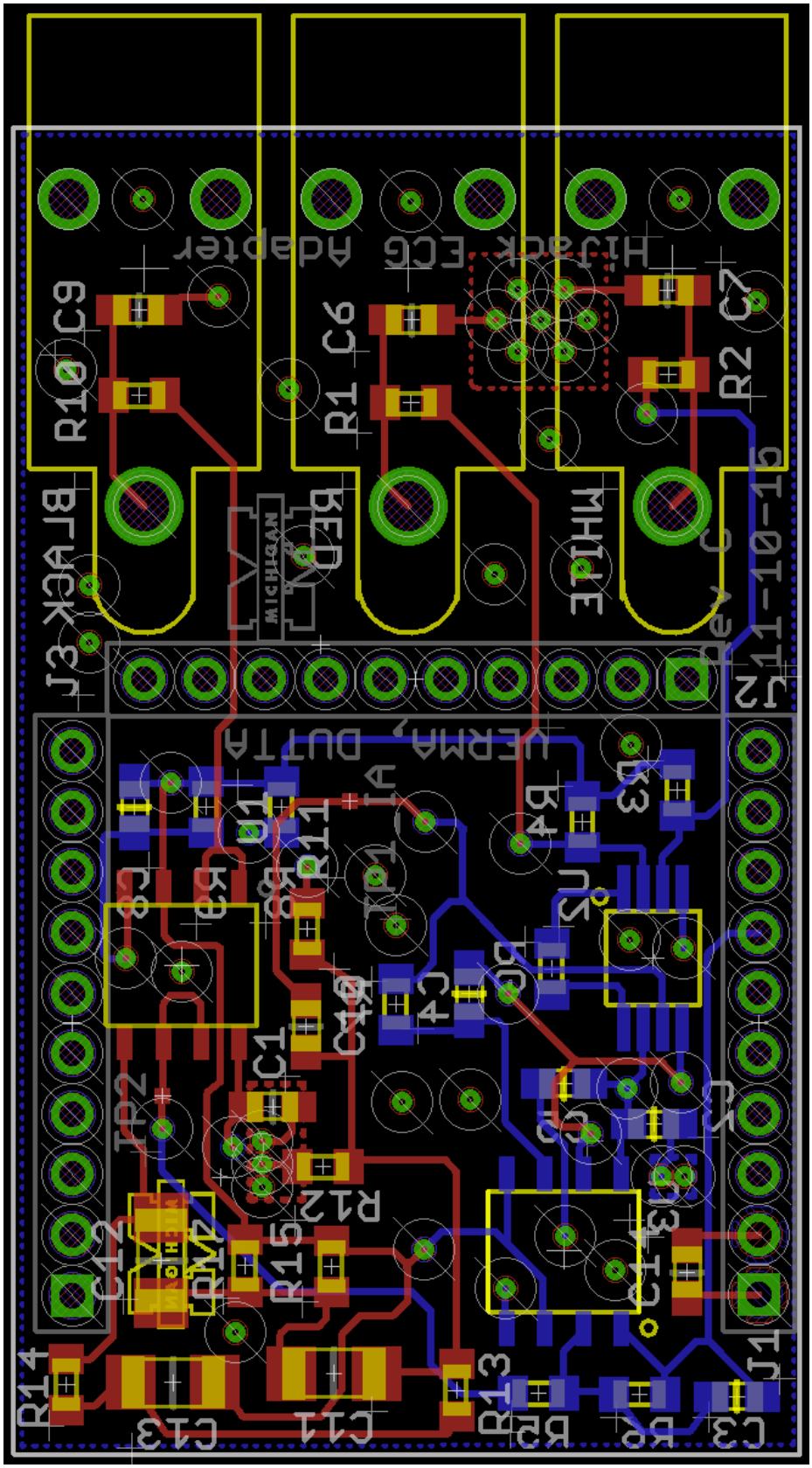


Figure A.**6:** 3-lead EKG sensor layout

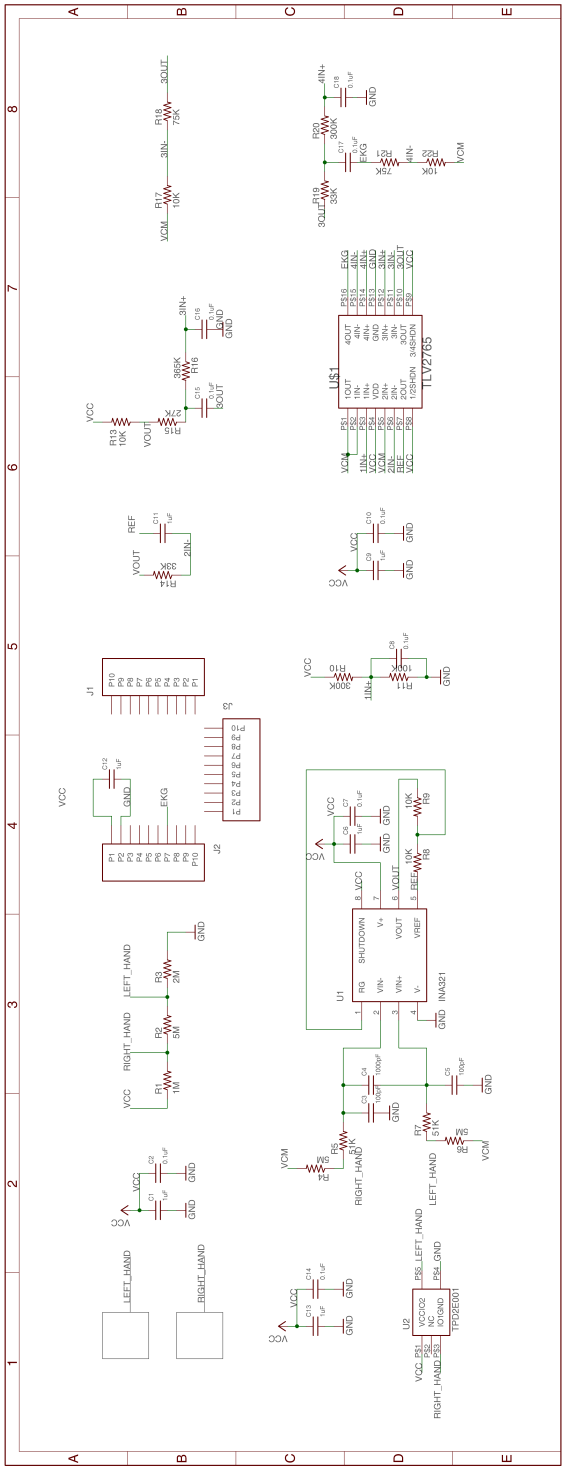


Figure A.**7:** Acquiring EKG signal from finger tips schematic

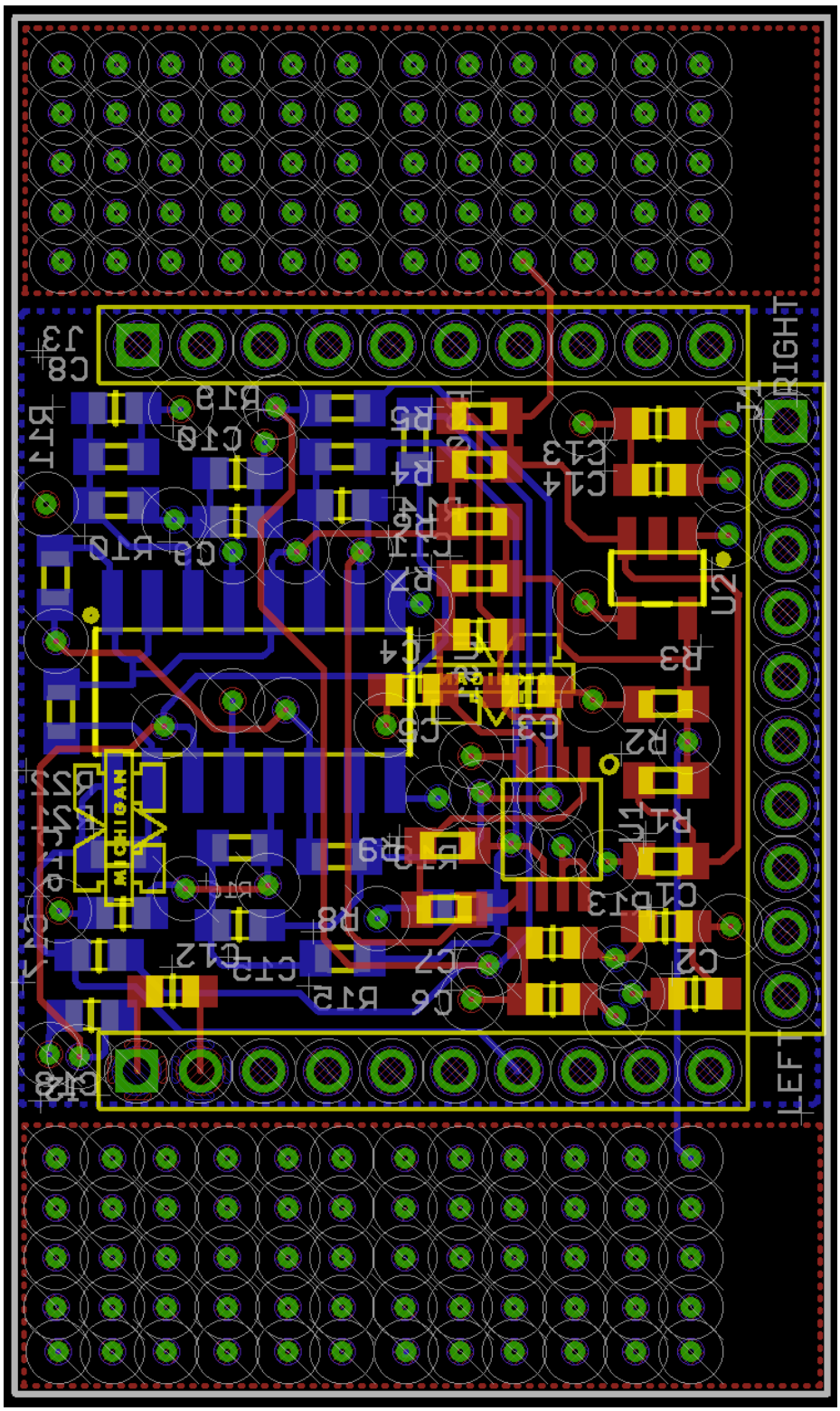


Figure A.**8:** Acquiring EKG signal from finger tips layout

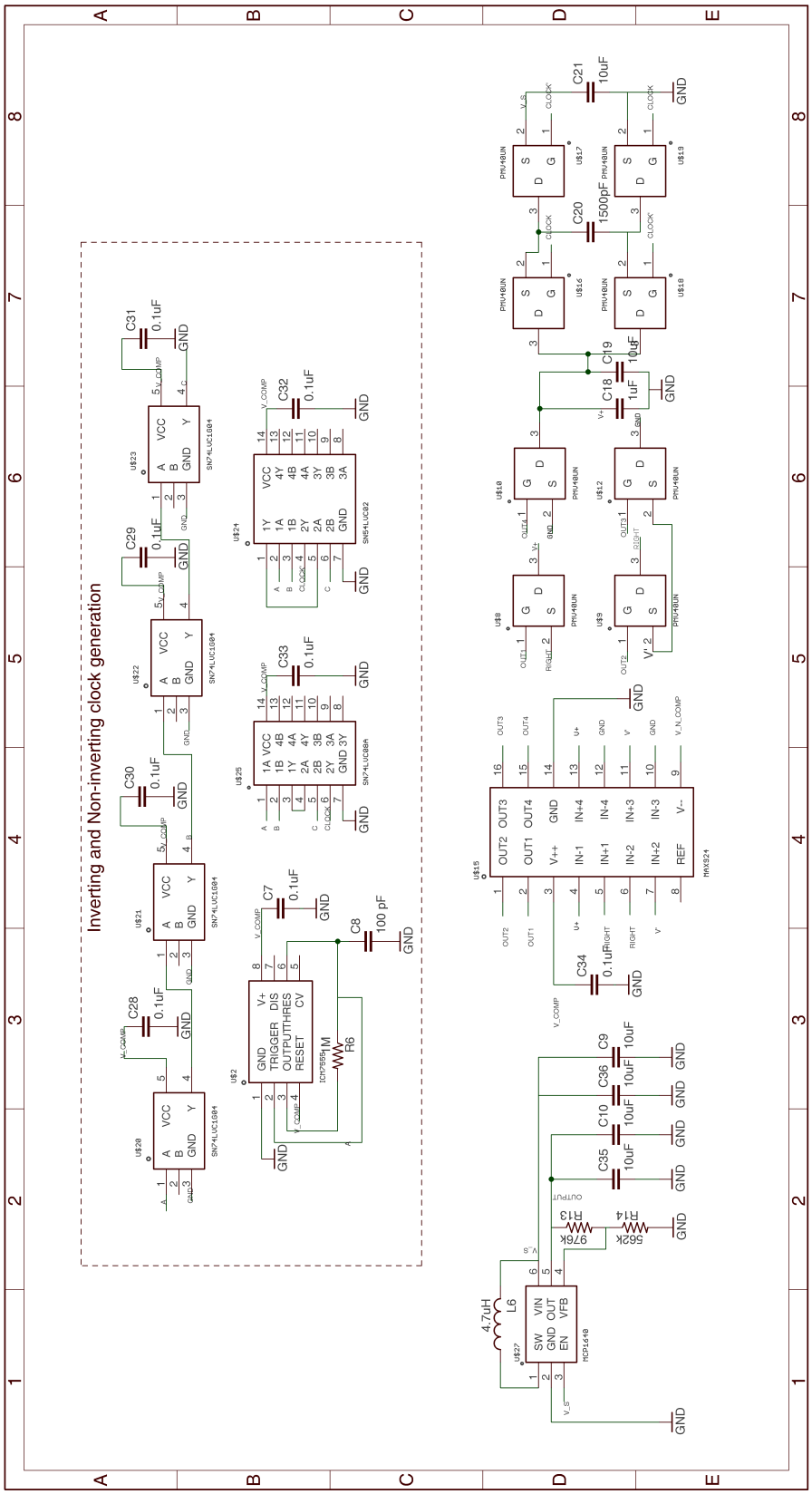


Figure A.**9:** AudioDAQ using synchronous rectification schematic

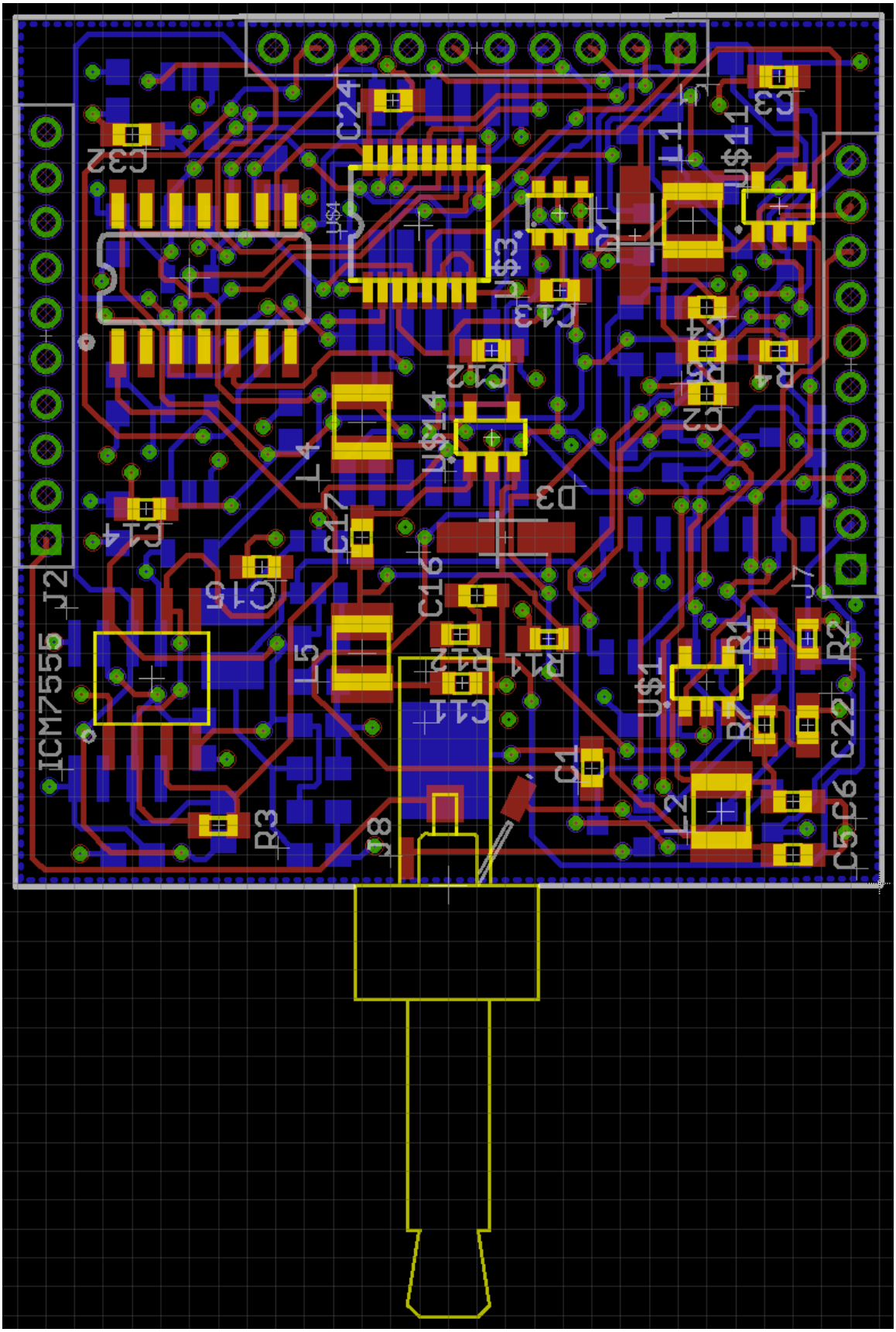


Figure A.**10:** AudioDAQ using synchronous rectification layout

APPENDIX B

BOM

Description	Manufacturer	Part No.	Cost
Plug	Kobiconn	171-7435-EX	$0.92
Regulator	TI	TPS79718	$0.50
RC Timer	Maxim	ICM7555	$0.13
Counter	TI	SN74LV161	$0.10
Multiplexer	Maxim	MAX4781	$1.77
Switch	C & K	KSR223	$0.50
PCB	Sunstone	n/a	$0.30
Passives	n/a	n/a	$0.75

Table B.1: AudioDAQ cost breakdown including the cost of the PCB and headset plug. It costs about $5 for 10 k units.

Description	Manufacturer	Part No.	Cost
Diff Op Amp	TI	INA333	$2.23
Op Amp	TI	OPA2333	$1.86
Wire Leads	WelchAllyn	008-0323-00	$16.0
Sockets	PlasticOne	8R003661901F	$3.00
PCB	Sunstone	n/a	$0.26
Passives	n/a	n/a	$0.75

Table B.2: 2-lead EKG sensor cost breakdown including the cost of the PCB board, major active components, and a small dollar value for passive components such as resistors and capacitors. The entire assembly costs approximately $24 in bulk, with the majority of the cost going towards specialized wire leads, which at larger production volumes could be commoditized and priced lower.

Description	Manufacturer	Part No.	Cost
Diff Op Amp	TI	INA333	$2.23
Op Amp	TI	OPA2333	$1.86
Wire Leads	WelchAllyn	008-0323-00	$24.0
Sockets	PlasticOne	8R003661901F	$4.50
PCB	Sunstone	n/a	$0.26
Passives	n/a	n/a	$0.75

Table B.3: 3-lead EKG sensor cost breakdown including the cost of the PCB board, major active components, and a small dollar value for passive components such as resistors and capacitors. The entire assembly costs approximately $33.5 in bulk, with the majority of the cost going towards specialized wire leads, which at larger production volumes could be commoditized and priced lower.

Description	Manufacturer	Part No.	Cost
Diff Op Amp	TI	INA321	$1.6
Op Amp	TI	TLV2765	$1.88
PCB	Sunstone	n/a	$0.26
Passives	n/a	n/a	$0.75

Table B.4: Cost of EKG sensor which uses left hand and right hand thumbs as electrode contacts. Total price is $4.75 for 10 k units.

BIBLIOGRAPHY

[1] http://www.devtoaster.com/products/rev/index.html, Apr. 2011.

[2] Android meets Arduino. http://developer.android.com/guide/topics/usb/adk.html, Apr. 2011.

[3] iCouchPotatoe. http://www.hackint0sh.org/f131/35975.html, Apr. 2011.

[4] A. P. Sample, D. J. Yeager, P. S. Powledge, A. V. Mamishev, and J. R. Smith. Design of an rfid-based battery free programmable sensing platform. In *IEEE transactions on Inst. and Meas.*, Nov. 2008.

[5] Alex Winston, LTD. iData (FSK): Freeing the iPhone and iPod. https://sites.google.com/a/alexwinston.com/www/idata, Apr. 2011.

[6] Apple, Inc. Apple. http://www.apple.com/iPhone/iPhone-3GS/, Apr. 2011.

[7] Apple, Inc. Apple. http://www.apple.com/iPad/, Apr. 2011.

[8] Apple, Inc. Apple iPhone TTY adapter. http://store.apple.com/us/product/MA854G/A, Apr. 2011.

[9] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. B. Srivastava. Participatory sensing. In *Workshop on World Sensor-Web: Mobile Device Centeric Sensor Networks and Applications*, Dec. 2006.

[10] C. Van Mieghem, M. Sabbe, and D. Knockaert. The clinical value of the EKG in noncardiac conditions. In *Chest 125 (4): 156176*, Sept. 2004.

[11] R. Chaudhri, G. Borriello, R. Andesron, S. McGuire, and E. O'Rourke. FoneAstra: Enabling remote monitoring of vaccine cold-chains using commodity mobile phones. In *DEV'10: Proceedings of the First Annual Symposium on Computing for Development*, dec 2010.

[12] H. Kennedy. Ambulatory electrocardiography including Holter recording technology. Philadelphia, Pa. : Lea and Febiger, 1981.

[13] J. Gummeson, S. S. Clark, K. Fu and D. Ganesan. On the limits of effective hybrid micro-energy harvesting on mobile crfid sensors. In *In MobiSys'10: Proceedings of the 8th Annual International Conference on Mobile Systems, Applications and Services*, 2010.

[14] L. Fay, V. Misra, and R. Sarpeshkar. A micropower electrocardiogram amplifier. In *IEEE transactions, Biomedical Circuits and Systems*, Sept. 2009.

[15] Linear Technology. LT1615. http://cds.linear.com/docs/Datasheet/16151fas.pdf, Apr. 2011.

[16] H. Lu, J. Yang, Z. Liu, N. D. Lane, T. Choudhury, and A. T. Campbell. The jigsaw continuous sensing engine for mobile phone applications. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, SenSys '10, pages 71–84, New York, NY, USA, 2010. ACM.

[17] M. D. Seeman. Analytical and practical analysis of switched-capacitor dc-dc converters. In *Technical Report, EECS Department, UC Berkeley*, 2006.

[18] M. D. Seeman, S. R. Randers. Analysis and optimisation of switched-capacitor dc-dc power converters. In *IEEE COMPEL*, 2006.

[19] M. S. Makowski and D. Maksimovic. Performance limits of switched-capacitor dc-dc converters. In *IEEE Power Electronics Specialists Conference*, 2005.

[20] M. Seeman, S. Sanders, and J. Rabaey. An ultra-low-power management ic for wireless sensor nodes. In *In IEEE Custom Integrated Circuits Conference*, Apr. 2007.

[21] Mouser Inc. Mouser. http://www.mouser.com/ProductDetail/Diodes-Inc/DFLS120L-7/?qs=JV7lzlMm3yKwTZC4liApr. 2011.

[22] P. Dutta, and L. Subramanian. Human-enabled microscopic environmental mobile sensing and feedback. In *In AI-D'10: Proceedings of the AAAI Spring Symposium on Artificial Intelligence for Development*, Mar. 2010.

[23] P. Dutta, J. Taneja, J. Jeong, X. Jiang, and D. E. Culler. A building bloack approach to sensornet systems. In *In proceedings of the 6th international conference on embedded networked sensor systems*, Nov. 2008.

[24] P. Germanakos, C. Mourlas, and G. Samaras. A mobile agent approach for ubiquitous and personalized ehealth information systems. In *UM'05: Proceedings of the Workshop on Personalization for eHealth of the 10th International Conference on user Modeling*, July 2005.

[25] Polar. Polar Wear+Hybrid transmitter. http://www.polar.fi/en/products/accessories/, Apr. 2011.

[26] B. Priyantha, D. Lymberopoulos, and J. Liu. Enabling energy efficient continuous sensing on mobile phones with littlerock. In *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, IPSN '10, pages 420–421, New York, NY, USA, 2010. ACM.

[27] Prological Solutions, LLC. H4I: Homemade for Integration. http://www.progical.com/, Apr. 2011.

[28] R. Honicky, E. A. Brewer, E. Paulos, and R. White. N-smarts: Networked suite of mobile atmospheric real-time sensors. In *SIGCOMM: Proceedings of the second ACM SIGCOMM workshop on Networked system for developing regions*, Dec. 2008.

[29] Ryz Media. My TV Remote. http://www.ryzmedia.com, Apr. 2011.

[30] S. Roundy et. al. Improving power output for vibration-based energy scavengers. In *IEEE Pervasive Computing*, 2005.

[31] seiko. Seiko. http://www.siic-ic.com/en, Apr. 2011.

[32] Square, Inc. Square card reader. http://www.squareup.com, Apr. 2011.

[33] SwitchScience. Audio jack modem for iPhone and Android. http://www.switch-science.com/trac/wiki/ARMS22-SOFTMODEM-HOWTO, Apr. 2011.

[34] T. R. F. Fulford-Jones, G.-Y. Wei, and M. Welsh. A portable, low-power, wireless two-lead EKG system. In *Proc. IEEE 26th Annu. Int. Conf. Engineering in Medicine and Biology Soc.*, Sept. 2004.

[35] Texas Instruments. Ekg-based heart-rate monitor implementation on the launchpad value line development kit using the msp430g2452 mcu. http://www.ti.com/lit/an/slaa486a/slaa486a.pdf, Apr. 2011.

[36] Texas Instruments. Ina321. http://www.ti.com/lit/ds/symlink/ina321.pdf, Apr. 2011.

[37] Texas Instruments. Tlv2765. http://www.ti.com/lit/ds/symlink/tlv2765.pdf, Apr. 2011.

[38] I. ThinkFlood. RedEye Mini universal remote for iPhone, iPod Touch & iPad. http://thinkflood.com/products/redeye-mini/, Apr. 2011.

[39] V. Fuster (ed.). Hurst's. The Heart, Tenth Edition, New York, NY. Mc-Graw-Hill Medical Publishing, ch. 11, 2011.

[40] Vital Wave Consulting. mHealth for development: The opportunity of mobile technology for healthcare in the developing world. United Nations Foundation, Vodafone Foundation, Feb. 2009.

[41] X. Jiang, J. Polastre and D. Culler. Perpetual environmentally powered sensor networks. In *In IPSN'05: The fourth international conference on information processing in sensor networks: Special track on platform tools and design methods for network embedded sensors*, Apr. 2005.

[42] Y. K. Ramadass, A. P. Chandrakasan. An efficient piezoelecyric energy harvesting interface circuit using a bias-flip rectifier and shared inductor. In *IEEE Journal of Solid-State Circuits*, Jan. 2010.

[43] T. S. Y.-S. Kuo, S. Verma and P. Dutta. Hijacking power and bandwidth from the mobile phone's audio interface. In *DEV'10: Proceedings of the First Annual Symposium on Computing for Development*, dec 2010.

ABSTRACT

AudioDAQ: Turning the Mobile Phone's Ubiquitous Headset Port into a Universal Data Acquisition Interface

by

Sonal Verma

Chair: Prabal Dutta

We present AudioDAQ, a new platform for continuous data acquisition using the headset port of a mobile phone. AudioDAQ differs from existing phone peripheral interfaces by drawing all necessary power from the microphone bias voltage, encoding all data as analog audio, and by making use of the phone's built-in voice memo application for continuous data collection. These differences enable low overall system power, make the design more universal among smart and feature phones, and enable simple peripherals without requiring a microcontroller. In contrast with prior designs, AudioDAQ works with a wide range of smart and feature phones, requires no hardware or software modifications on the phone, uses significantly less power, and allows continuous data capture over an extended period of time. The design is efficient because it draws all necessary power from the microphone bias signal, and it is general because this voltage is present in nearly every headset port. We show the viability of our architecture by evaluating an end-to-end system that can capture EKG signals continuously for hours, process the data locally, and deliver the data to the

cloud for storage, processing, and visualization.