

VIBRATION MONITOR ANALOG INSTRUMENTATION:
ACCELEROMETERS VS. STRAIN GAGES

A Major Qualifying Project Report:

submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the

Degree of Bachelor of Science

by

James Filice

Naveed Naeem

Kaung Myat Win

Date: April 24, 2013



WPI

Approved: Professor John A. McNeill

Abstract

The goal of the project was to implement a low-cost vibration monitoring device that could be used to detect the presence of a loose screw in a vibrating mechanical system. The instrumentation system was implemented using both strain gages and accelerometers and a performance comparison between the two implementations was made. It was found that with proper signal conditioning, less expensive strain gages showed comparable performance to that of accelerometers.

Table of Contents

1	Introduction	5
1.1	Report Overview	5
1.1.1	Wireless Heart Rate Monitor.....	5
1.1.2	Vibration Monitor.....	8
1.1.3	Radar System	10
1.2	Motivation for Mechanical Vibration Analyses	13
1.3	Why Use the AD8421?.....	13
2	Project System Design Process.....	15
2.1	Project Elements and System Overview.....	15
2.1.1	Analog Front End.....	15
2.1.2	Data Acquisition	15
2.1.3	Signal Processing Back End	16
2.2	Design Ideas Discussion	17
2.2.1	Analog Discovery Based DAQ System with Microcontroller Bridge Offset Nulling..	17
2.2.2	Custom Microcontroller Based DAQ System.....	18
2.2.3	Analog Discovery Based DAQ System with Fixed Potentiometer Bridge Balancing	19
2.3	Project Discussion and Decision	20
3	Device Theory and Operation.....	21
3.1	Strain Gages and Wheatstone Bridges	21
3.2	The Instrumentation Amplifier.....	27
3.3	Accelerometers	28
3.3.1	Post-Amp and Gain for Accelerometers:.....	32
3.4	8-to-1 Multiplexer.....	33
3.5	MATLAB	33
4	System Requirements and Detailed Design.....	35
4.1	Mechanical System	35
4.2	Data Acquisition.....	36
4.2.1	Analog Discovery and Analog Multiplexer.....	36
4.2.2	Synchronization.....	37
4.3	Analog Front End	38
4.3.1	Strain Gage Signal Chain.....	38

4.3.2 Accelerometer Signal Chain.....	41
4.4 Noise Analysis of Analog Front End.....	42
4.5 D/A Converter-Based Wheatstone Bridge Offset Nulling	48
4.6 System Requirements and Final Design.....	52
5 Tests and Results	53
5.1 Signal Verification Tests.....	53
5.1.1 Sinusoidal Input	53
5.1.2 Triangle Wave Input.....	56
5.1.3 Square Wave Input	58
5.2 Full System Test	60
6 Future Work and Recommendations	65
6.1 Data Acquisition.....	65
6.2 Offset Adjustment	65
6.3 Power Requirements.....	66
7 Conclusion.....	67
8 Acknowledgements	68
9 References.....	69
Appendix A.....	70
Appendix B.....	75
Appendix C.....	78

Table of Figures

Figure 1: Data Acquisition Chain Functional Block Diagram	15
Figure 2: System Block Diagram with Analog Discovery with D/A Converter.....	17
Figure 3: System Block Diagram with ADuC7023	18
Figure 4: Analog Discovery System Block Diagram with Trim Pots and a Buffered Reference Voltage.....	19
Figure 5: Material at Rest Dimensions.....	21
Figure 6: Material Experiencing Strain	21
Figure 7: Metal Foil Strain Gage.....	23
Figure 8: Wheatstone Bridge.....	24
Figure 9: Internal Construction of the ADXL001-70	30
Figure 10: Functional Block Diagram of ADXL001-70	31
Figure 11: Preliminary Testing to Determine Accelerometer Gain.....	31

Figure 12: Plot of CMMR as a Function of Frequency for Different Gain Values	32
Figure 13: Mechanical Test Apparatus	35
Figure 14: Control Line Signals	37
Figure 15: Four Resistors in Wheatstone Bridge Configuration.....	38
Figure 16: Strain Gage Signal Chain Post-Amp	40
Figure 17: A Simplified Representation of the Gain Stages with Noise.....	42
Figure 18: Simplified Circuit Diagram for Noise Analysis	45
Figure 19: (left) Total Input Referred Noise Due to R1; (Right) Total Output Vrms Due to Noise Present in the Circuit.....	46
Figure 20: (left) Total Current Consumption from the Bridge; (Right) Figure of Merit Analysis	47
Figure 21: Total Output Noise Density Seen by A/D Converter.....	48
Figure 22: 24-Bit Wide Input Shift Register of AD5684R D/A Converter.....	49
Figure 23: Commands Used to Control the D/A Converter.....	49
Figure 24: Address Bits Used to Select Particular D/A Converter.....	49
Figure 25: Timing Diagram for a Write Operation.....	50
Figure 26: Simplified Circuit Diagram with Offset Adjustment.....	50
Figure 27: System Block Diagram with Signal Levels	52
Figure 28: Output Resulting from 1 kHz Sinusoid Input to Strain Gage Signal Chain	54
Figure 29: Output Resulting from 200 Hz Sinusoid Input to Accelerometer Signal Chain	55
Figure 30: Output Resulting from 1 kHz Triangle Wave Input to Strain Gage Signal Chain.....	56
Figure 31: Output Resulting from 200 Hz Triangle Wave Input to Accelerometer Signal Chain.....	57
Figure 32: Output Resulting from 1 kHz Square Wave Input to Strain Gage Signal Chain	58
Figure 33: Output Resulting from 200 Hz Square Wave Input to Accelerometer Signal Chain	59
Figure 34: Strain Gage Signals with All Four Screws in Place	60
Figure 35: Accelerometer Signals with All Four Screws in Place	61
Figure 36: Strain Gage Signals with Screw 2 Loosened	62
Figure 37: Accelerometer Signals with Screw 2 Loosened	63

Table of Tables

Table 1: Summary of Mechanical System Dimensions and Sensor Placement	36
Table 2: Summary of AD8421 Noise Specs.....	44

1 Introduction

The primary objective of the project was to implement a low cost system for performing vibration analysis on mechanical systems as a tradeshow demo for the AD8421 precision instrumentation amplifier from Analog Devices, Inc. (ADI). The initial project formulation did not specify a single ADI product and a list of possible project ideas were presented from which the project was chosen. The system should ultimately be able to detect a loose screw holding in place a vibrating plate system.

1.1 Report Overview

Initial research included exploring energy harvesting and wireless sensor applications. Project ideas were proposed based on the finding of this research. Additionally, the project was intended to be applications based and the final goal was to create a system that could be used in a tradeshow atmosphere to show the superior capabilities of the products in the Analog Devices catalog. ADI also proposed a set of projects comprised of a heart rate monitor design, a radar system, and a vibration monitor. Based on these suggestions the team began to look at prior art for each project. Measures to determine project desirability and feasibility were devised. The categories that were chosen for evaluating possible projects were ease of implementation, ease of demonstration, “wow factor”, and the ability for the given project to win an MQP award and the overall team interest. The proposed projects are detailed below.

1.1.1 Wireless Heart Rate Monitor

Demo Idea:

For the actual tradeshow demo we will need to highlight best traits (high CMRR, low input offset voltage drift, and low power consumption) of the ADI product being showcased in the wireless heart rate monitor. We propose creating a test board which allows us to choose between multiple signal paths, each one containing a different in-amp. The ECG signal that we obtain will be displayed on a monitor and the differences in clarity of PQRST wave between the different signal paths could be used to demonstrate the superiority of precision ADI in-amps. Additionally, we could generate excessive 60 Hz noise to emphasize the high CMRR of the AD8237 and/or AD8420. Competitors’ products could be used and obscured from sight to conceal their identity. We could also subject the circuit to a range of temperatures to demonstrate the low input offset voltage drift.

Project Appeal:

- Ease of implementation (Highest):
 - Wireless heart rate monitor designs appear to be fairly common. This will give us many resources and references. Overall, this would probably be the easiest of the suggested projects to bring to fruition.
- Ease of demonstration (Highest):
 - If we were to use a wristband wireless electrode, it would be a very tactile tradeshow experience. One thing we could do is display the actual PQRST ECG signal while subjecting the system to various 60 Hz noise conditions to demonstrate the superior CMRR of the ADI in-amp being used. This project
- Wow factor (Middle):
 - While the demonstration would be very interactive, it would not be particularly impressive or memorable. The means that it may not stand out despite highlighting the superior performance of an ADI product.
- MQP winnability and group interest (Lowest):
 - Partly because wireless heart rate monitors have been done so many times, the project would not be as interesting to our team as the vibration sensor and radar projects would be. We would need to do something truly innovative, interesting, or impressive to have a first place MQP.

ADI Product Selection:

- AD8420^[1]:
- Maximum supply current:
80 μ A
- Minimum CMRR: 100 dB (out to 1 kHz)
- Rail-to-rail output
- Input voltage range goes below ground
- Gain set with 2 external resistors
-Can achieve low gain drift at any gain
- Very wide power supply range

- Single supply: 2.7 V to 36 V
- Dual supply: ± 2.7 V to ± 18 V
- Bandwidth (G = 100):
2.5 kHz
- Input voltage noise:
55 nV/ $\sqrt{\text{Hz}}$

The AD8420 is low power, boasts a very high CMRR = 100 dB at DC, and has a more than adequate bandwidth and gain for the purposes of ECG measurements.

AD8237^[2]:

- Gain set with 2 external resistors
 - Can achieve low gain drift at all gains
- Ideal for battery powered instruments
 - Supply current: 115 μA
 - Rail-to-rail input and output
 - Zero input crossover distortion
- Designed for excellent dc performance
 - Minimum CMRR: 106 dB (80 dB out to 1 kHz)
 - Maximum offset voltage drift: 0.3 $\mu\text{V}/^\circ\text{C}$
- Bandwidth mode pin (BW) to adjust compensation
- 8 kV HBM ESD rating
- RFI filter on-chip
- Single-supply operation:
1.8 V to 5.5 V

The AD8237 has power specifications comparable to those of the AD8420, but with less flexibility in supply voltage. While the offset voltage drift for the AD8237 (0.3 $\mu\text{V}/^\circ\text{C}$) is superior to that of the AD8420, the input voltage noise (68 nV/ $\sqrt{\text{Hz}}$) is slightly inferior. Overall, the two chips would be equally viable as ECG in-amps for the purposes of a tradeshow demonstration. The determining factor would be which chip ADI would like to showcase.

Sensor Selection:

An interactive tradeshow demo should be easy to setup, reusable, and safe. Ultimately, this eliminates diagnostic-quality ECG setups as they require multiple electrodes. A better alternative would be to utilize a chest band which contains electrodes, but does not require direct contact with the skin. In a tradeshow environment, even this is undesirable. The best solution would be to utilize a wrist band. A wrist band can be easily put on and removed.

1.1.2 Vibration Monitor

Demo Idea:

The initial idea is to use a motor which is screwed to a board and intentionally loosen the screw to measure the vibration depending on how loose the screw is. The sensor/transducer outputs electrical signal which will be transmitted to the computer to be further processed. The amplitude of vibration as a function of time and frequency as well as other useful information for analysis will be displayed using GUIs such as LABVIEW or MATLAB. There are many books on vibrational analysis on machines, motors and other mechanical parts that we can look into if we are going in this direction.

Project Appeal:

- Ease of implementation (High):
 - Vibrational monitor design is very widespread in manufacturing and automotive industries. In the manufacturing factories, the vibrational sensors are used to monitor the loose screws along the assembly lines, while in automotive industries such as Formula One, vibrational sensors are used to spot mechanical imbalances to ensure safety of drivers. Due to its common use in such industries, this project will have a certain amount of resources available to us. Although there are no mechanical engineers on the team, there will not be any problem in completing the project since the project only needs minimal knowledge on the mechanical side. Additionally, we could always approach professors in Mechanical and Robotics Engineering departments if necessary.
- Ease of demonstration (Highest):

- If we were to implement a monitoring system with a simple motor with a loose screw, it would be simple and effective tradeshow experience. One thing we could do is compare and contrast the performance of chips from different competitors while subjecting the system to various noise conditions to demonstrate the superior CMRR, slew rate, bandwidth and low noise property of AD8421 being used.
- Wow factor (High):
 - Although the appearance is simple, the demonstration will be very persuasive since we are comparing against competitors.
- MQP winnability and group interest (Medium):
 - Although there may be vibrational monitoring systems available, the team has high interest in such a multi-disciplinary project which focuses more on sensor and signal processing aspects. In comparison with two other projects, this project was deemed more appealing.

ADI Product Selection:

AD8421:

Low power

- 2.3 mA maximum supply current

Low noise

- 3.2 nV/ $\sqrt{\text{Hz}}$ maximum input voltage noise at 1 kHz
- 200 fA/ $\sqrt{\text{Hz}}$ current noise at 1 kHz

Excellent ac specifications

- 10 MHz bandwidth (G = 1)
- 2 MHz bandwidth (G = 100)
- 0.6 μs settling time to 0.001% (G = 10)
- 80 dB CMRR at 20 kHz (G = 1)
- 35 V/ μs slew rate

High precision dc performance (AD8421BRZ)

- 94 dB CMRR minimum (G = 1)
- 0.2 $\mu\text{V}/^\circ\text{C}$ maximum input offset voltage drift

- 1 ppm/°C maximum gain drift ($G = 1$)
- 500 pA maximum input bias current

Inputs protected to 40 V from opposite supply

- ± 2.5 V to ± 18 V dual supply (5 V to 36 V single supply)
- Gain set with a single resistor ($G = 1$ to 10,000)

The excellent gain and bandwidth characteristics of the AD8421 would be best suited for the vibration monitor application. The system would be sensing minute changes and also needs high noise immunity. With a gain of 100 the CMRR of the AD8421 is 120dB at 20kHz, which is the best performing in-amp of all Analog Devices options^[3].

Sensor Selection:

The types of sensors that will be used in this application are:

- Strain Gages
- Accelerometers

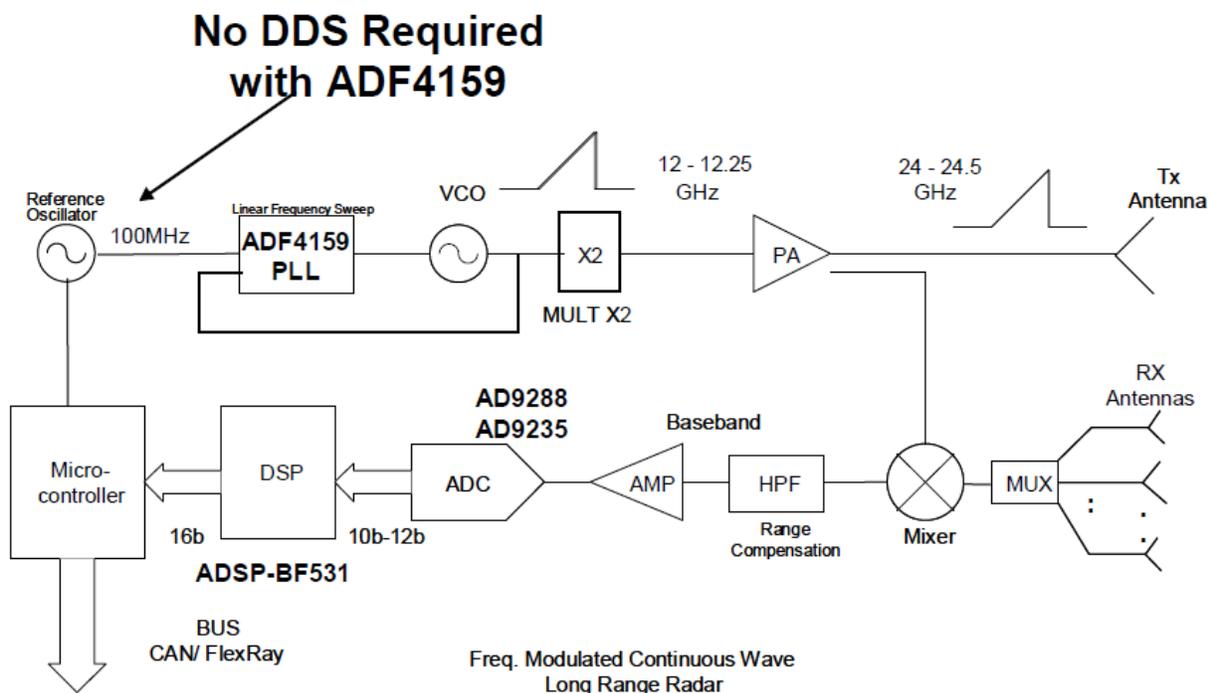
The focus of this project would be to determine mechanical modes of a system and detect them without the need for more expensive equipment such as spectrum analyzers which are typically used to perform this task.

1.1.3 Radar System

Project Appeal:

- Ease of Implementation (Lowest):
 - Radar systems are extremely difficult to realize in short periods of time and require a vast understanding of many aspects of electrical and computer engineering. These include RF circuits, digital signal processing, filtering, embedded systems, and phase-locked loops. If this project is picked it may be difficult to realize given the time constriction on the project. The AD8232 provides a complete analog front end of the radar system providing low-noise and power amplification, anti-aliasing filters and an ADC output. The AD4159 can provide signal generation and phase-locked loop. The other necessary blocks in the radar would be antennas, a voltage controlled

oscillator, TX and RX MMICs, DSP, and a host processor. An example of a block diagram for a radar system is shown below



- Ease of Demonstration (Medium):
 - The type of application that the radar system would be used for if implemented would determine the ease of having a demonstration in a trade-show atmosphere. Some of the type of applications suggested was a proximity detector which would be relatively easy to demonstrate if given an adequate amount of space. At the same time the velocity of a target could be determined. The only problem that may arise is that there are too many people in the vicinity and the amount of allowable radiation people can be exposed to.
- Wow Factor (Highest):
 - Radar is a very interesting application and the quality of ADI chips would contribute to the overall quality of the system. Radar provides a wide range of possible demonstrations and if done properly would be an excellent showcase of the versatility of the Analog Devices Catalog.
- MQP winnability and group interest (Highest):

- Although the project would garner awards if there was enough time to complete it, therein lies the problem which is time. Radar systems usually take teams of tens of peoples to get the system up, running, and functional. The interest is there amongst group members but the feasibility of the project seems implausible given the lack of background knowledge and time available.

ADI Product Selection:

AD8283^[4]:

- Provides LNA, PGA, and Anti-aliasing filter stages than can be multiplexed to support up to 6 inputs.
- SPI-programmable gain = 16 dB to 34 dB in 6 dB steps
- Antialiasing filter (AAF)
- Programmable third-order low-pass elliptic filter (LPF) from 1 MHz to 12 MHz
- Analog-to-digital converter (ADC)
- 12 bits of accuracy up to 80 MSPS
- SNR = 67 dB
- SFDR = 68 dB
- Low power, 170 mW per channel at 12 bits/80 MSPS
- Low noise, 3.5 nV/ $\sqrt{\text{Hz}}$ maximum of input referred voltage noise
- Power-down mode

ADF4159^[5]:

- RF bandwidth to 13 GHz
- High and low speed FMCW Ramps Generation
- 25-bit fixed modulus allows sub-hertz frequency resolution
- PFD Frequencies up to 110MHz
- Frequency and Phase modulation capability
- Sawtooth and triangular waveforms generation
- Parabolic ramp
- Ramp superimposed with FSK

- Ramp with 2 different sweep rates
- Ramp Delay
- Ramp Frequency Readback
- Ramp Interruption
- 2.7 V to 3.3 V analog power supply
- 1.8 V digital power supply
- Programmable charge pump currents
- 3-wire serial interface
- Digital lock detect
- Power-down mode
- Cycle Slip Reduction for faster lock times

After meeting with project advisor, Professor John McNeill, and ADI project liaison, Oljeta Qirko, it was decided to pursue the vibration monitoring system. This decision was reached after weighing the interesting analog design content of the considered projects and their feasibility given the time constraints of the major qualifying project.

1.2 Motivation for Mechanical Vibration Analyses

The primary motivation for measuring vibration in mechanical systems is to detect component faults and to avoid operating under resonant conditions^[6]. Mechanical engineers and technicians measure system vibrations mainly by converting mechanical signals, such as acceleration and strain, into voltage signals and performing a Fourier transform. Spectrum analyzers are the most commonly used tool for performing this task in industry. Due to the need for a spectrum analyzer to perform the necessary analysis it significantly increases the cost of performing these procedures. The purpose of this project is to be able to remove the need for extraneous equipment such a spectrum analyzer and allow the analysis procedure to be performed with the use of a laptop, a MATLAB license, and low-cost accelerometers and strain gages.

1.3 Why Use the AD8421?

Since the system we developed uses strain gages, which converts a mechanical strain into a corresponding change in resistance. This resistance is small and the need to create a differential signal is inherent. The use of an instrumentation-amplifier in tandem with a Wheatstone bridge

allows for differential signal amplification. These small changes in resistance can be converted into detectable changes in voltage. Furthermore, the signal chain needs to have high noise immunity due to the sensitivity of the strain gauges and therefore all components within the signal chain must not inject a significant amount of noise. This is the reasoning behind the choice of the AD8421 instrumentation-amplifier. The $3\text{nV}/\sqrt{\text{Hz}}$ maximum input voltage noise and $200\text{fA}/\sqrt{\text{Hz}}$ current noise at 1kHz are the lowest in the ADI product catalog. At the same time it provides the best gain and bandwidth characteristics among all of the available instrumentation amplifiers. The AD8421 was developed specifically for applications such as medical instrumentation, precision data acquisition, microphone pre-amplification, and vibration analysis. Due to its capabilities it allows us to have a dynamic range of mechanical vibrations that can be detected simply by adjusting the gain.

Although the low-noise, CMRR, PSRR, gain, and bandwidth characteristics are optimal, this is not a rail-to-rail instrumentation amplifier. However, there are no rail-to-rail instrumentation amplifiers that have the capabilities as the AD8421 which is why a post-amplifier with similar noise immunity with rail-to-rail capabilities will be used to allow us to recover almost the entire voltage swing at the output.

2 Project System Design Process

2.1 Project Elements and System Overview

At the onset of the project the intended product was a data acquisition and signal processing system for performing modal analysis on a vibrating plate. We separated the system into three distinct sub-systems: the analog front end, the data acquisition system, and the signal processing back end. Deconstructing the project into these three elements was done with the intention of categorizing sub-systems by their main required design discipline.

We proposed the broad system-level design in Figure 1 for our project.

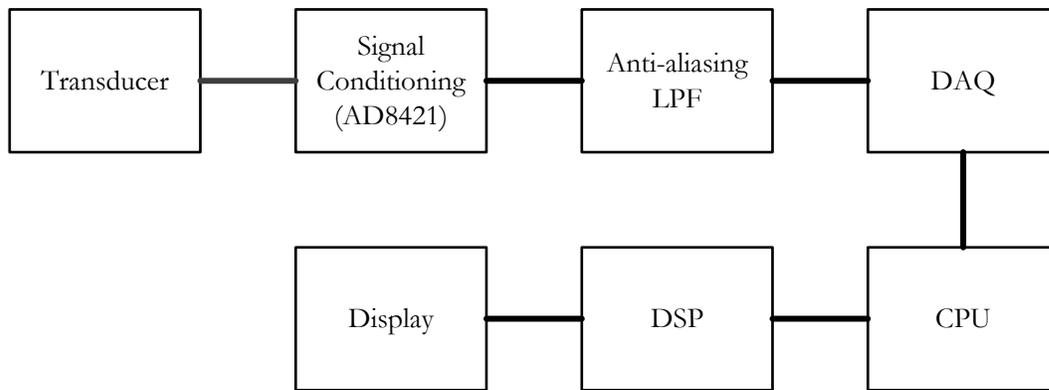


Figure 1: Data Acquisition Chain Functional Block Diagram

2.1.1 Analog Front End

The analog front end is comprised of the transducer, the signal conditioning hardware, and the anti-aliasing low-pass filter. The transducer consists of the four Wheatstone bridges and the four accelerometers. Each of the four Wheatstone bridges must be balanced to prevent the inherent voltage offset from being amplified by the signal conditioning and active low-pass anti-aliasing filter. The low-pass filter limits the bandwidth of the system and prevents aliasing of the sampled transducer outputs as well as limiting noise in the system. For the strain gage sensors the low-pass filter also converts the signal range to utilize the full 5 V range of the analog front end.

2.1.2 Data Acquisition

The data acquisition unit converts the analog signals from the transducers to digital signals that can be manipulated in software. The majority of data acquisition systems use a single channel

multiplexed A/D converter architecture where multiple analog input channels are multiplexed to a single A/D converter channel. This architecture is cost effective, but results in having to split the throughput evenly between each active input channel.

2.1.3 Signal Processing Back End

Initially we considered using LabView to perform the signal processing for our system, but this was dropped in favor of MATLAB for two reasons. The first reason is our familiarity with MATLAB. We already have experience using it for signal processing from previous courses. The second reason was that the mathematical operations necessary to perform modal analysis on a system are not readily performed in LabView whereas MATLAB supports a variety of signal processing functions that are already written for the user.

There are also two other options that must be considered when processing signals for any application—real time signal processing and post-processing. In our case, we explored a real-time implementation and found many limitations which effectively restricted us to post-processing. This, however, is acceptable because we do not need to monitor a signal that evolves in time, rather, we need to only monitor a system driven by a periodic oscillation.

2.2 Design Ideas Discussion

2.2.1 Analog Discovery Based DAQ System with Microcontroller Bridge Offset Nulling

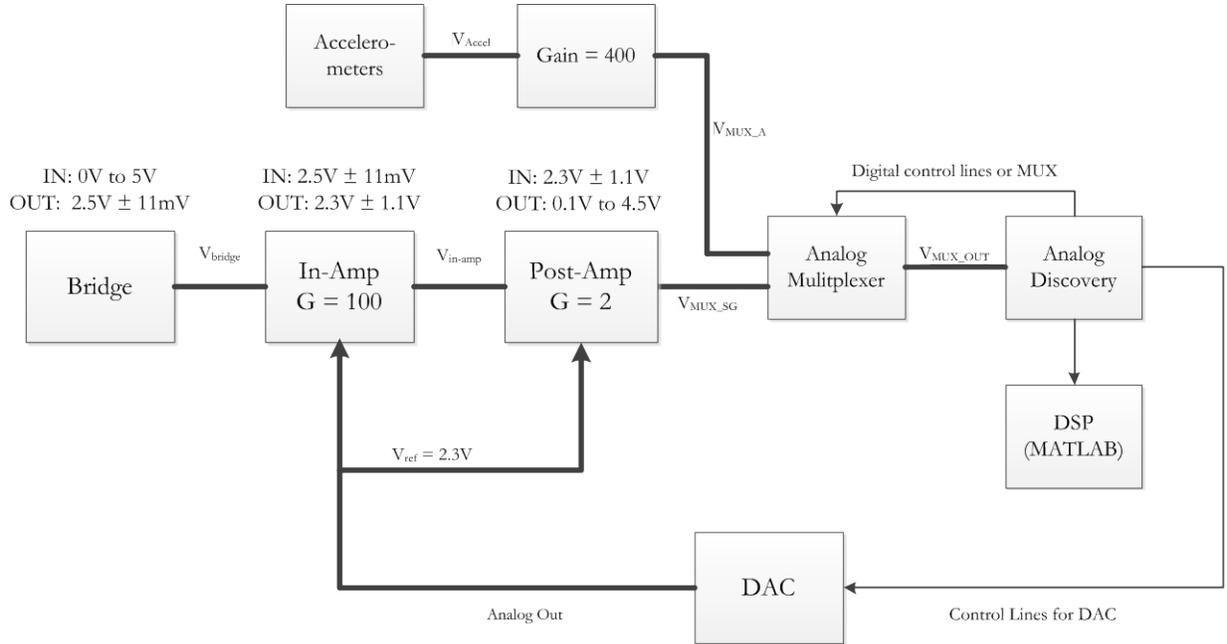


Figure 2: System Block Diagram with Analog Discovery with D/A Converter

This system block diagram describes the hardware implementation of the system we would implement if it is possible to use the 16 digital I/O ports of the Analog Discovery to properly write to the D/A converter such that the Wheatstone bridge offsets can be properly calibrated out. Another option that was suggested was to have an MSP430 microcontroller to control just the D/A convert calibration by using the Launchpad to program the 20 pin DIP and then place it in the circuit. This idea was strongly considered because it is simple to do if we cannot properly control the D/A converter using the Analog Discovery. In the case that using the Analog Discovery to control the D/A converter is feasible, then the system block diagram in Figure 2 would be the implementation and our PCB would solely consist of the analog front end for the analog signal conditioning of the sensor outputs. Due to the fact that the Analog discovery only has two analog inputs that can be used to interface a signal with MATLAB, an analog multiplexer is needed to be able to take information from all eight sensors. By using the waveform generator on the Analog Discovery, three digital control lines can be generated to cycle through the inputs of the multiplexer. The data from each of these sensors will be stored in their own distinct vector within MATLAB which can then be processed accordingly.

Pros:

- More than adequate sample rate for DAQ using the Analog Discovery
- Wheatstone bridge calibration using a microcontroller to drive a D/A converter increases system robustness

Cons:

- Requires extra processing to ensure channel synchronization
- May require additional user input to load .csv files for digital I/O

2.2.2 Custom Microcontroller Based DAQ System

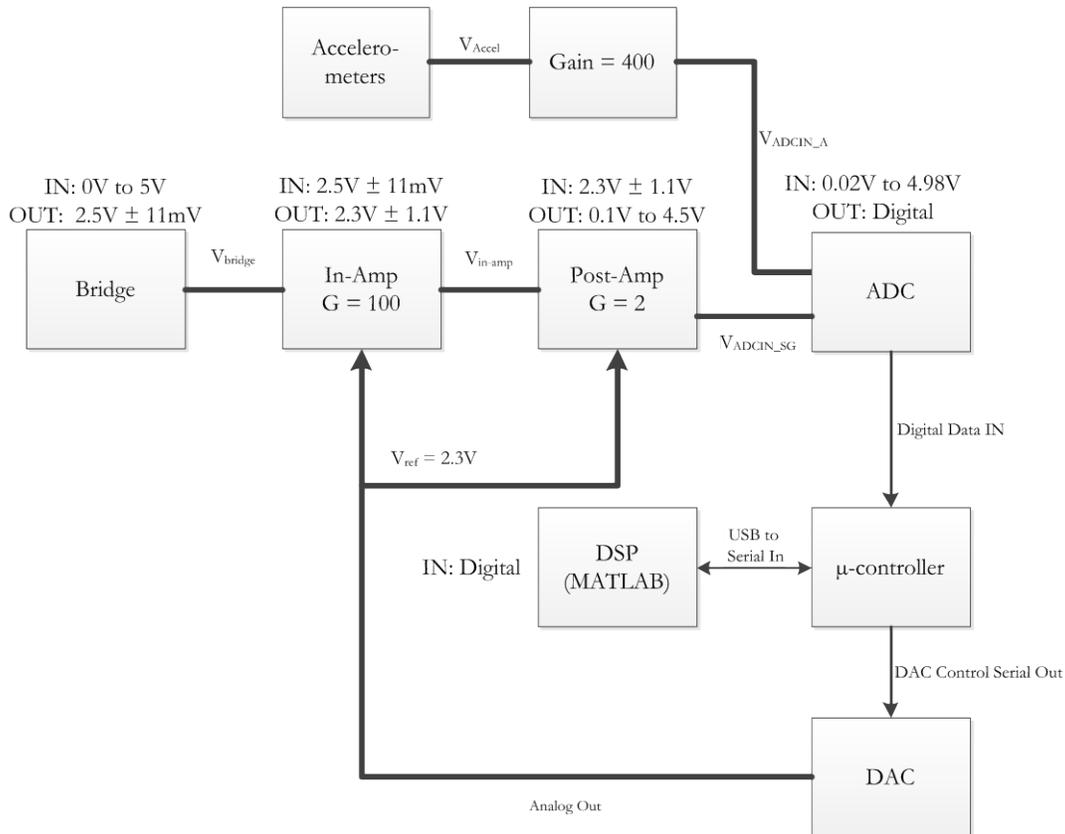


Figure 3: System Block Diagram with ADuC7023

The system block diagram in Figure 3 shows the implementation of our system using the ADuC7023. The analog front end of the system is the same as that in Figure 3. However, the data acquisition portion would have to be purpose-built. Although it allows for easy D/A converter calibration, the process of sending data and acquiring it is complicated by the fact that the

microcontroller is ARM based. None of the team members have familiarity with ARM systems and the development process would be prolonged.

Primarily, data transfer limitations are the limiting factor in designing this system. It was found that USB transfer rates are not sufficient for real time and, additionally, that direct memory access and external RAM would be required to fully realize such a system. Given the time constraints of the project, this option proved to be infeasible, though much time was spent investigating it.

Pros:

- Only requires a microcontroller which is used both for DAQ and for driving the D/A for Wheatstone bridge calibrations
- Requires no user interactions once data collection has begun

Cons:

- Requires building a DAQ system from the ground up which, in turn, requires embedded system expertise not present in the team
- Serial data transmission over USB may pose a significant challenge

2.2.3 Analog Discovery Based DAQ System with Fixed Potentiometer Bridge Balancing

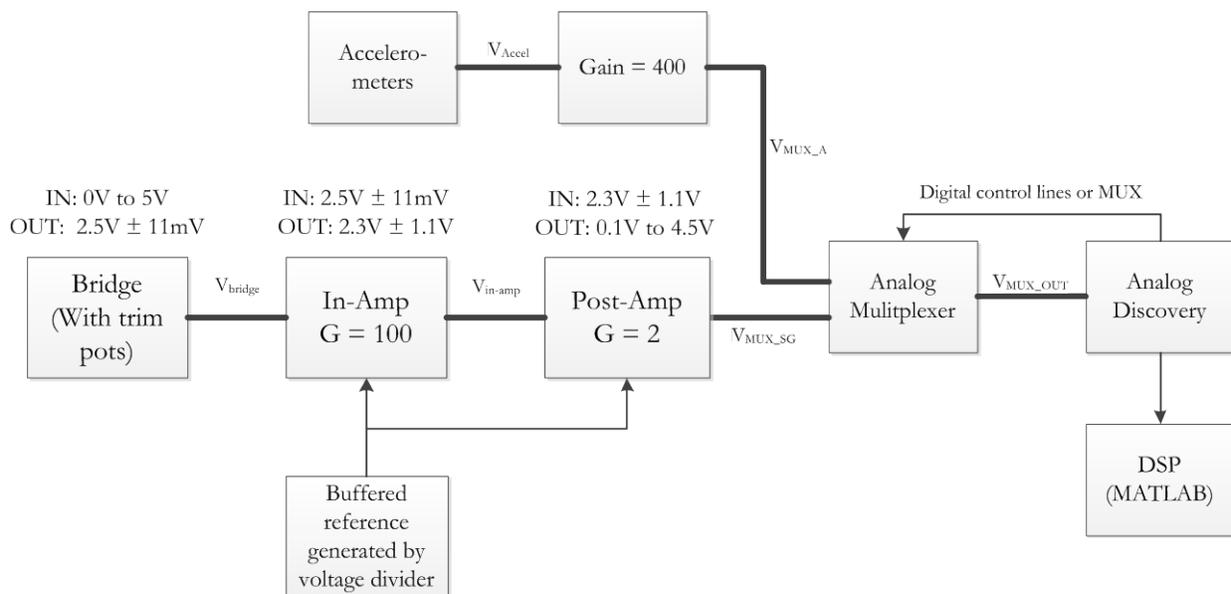


Figure 4: Analog Discovery System Block Diagram with Trim Pots and a Buffered Reference Voltage

The third option considered was that in which the Analog discovery cannot properly communicate with the D/A converter and using the MSP430 to control the D/A converter is not a viable option. The system block diagram for such a system is shown in Figure 4. Instead of using a D/A converter to calibrate out the offset of the Wheatstone bridges, trimming potentiometers would be used to offset adjust the bridge manually. The necessary 2.3V reference voltage for the in-amp would then need to be generated by a dedicated reference. This option is not ideal and would require several extra components but would eliminate the need for a D/A converter.

Pros:

- Eliminates the need for a D/A converter and the microcontroller to drive it
- Does not require a calibration step to balance the Wheatstone bridges prior to gathering data

Cons:

- Increased susceptibility to temperature induced changes in bridge offset voltages leading to decreased system robustness
- Need to generate a single precision reference to bias the instrumentation amplifiers

2.3 Project Discussion and Decision

After receiving the Analog Discovery we constructed a prototype of the DAQ system and found that the system shown in Figure 2 was viable. It was found that by using this design yielded usable data and allowed bridge offset nulling without user intervention, though the cons of the system were found to be accurate.

3 Device Theory and Operation

3.1 Strain Gages and Wheatstone Bridges

Strain gages have an at rest resistance when force is applied to them their dimensions change which results in a corresponding change in resistance. A material at its rest dimensions is presented in Figure 5. The same material experiencing elongation is depicted in Figure 6.

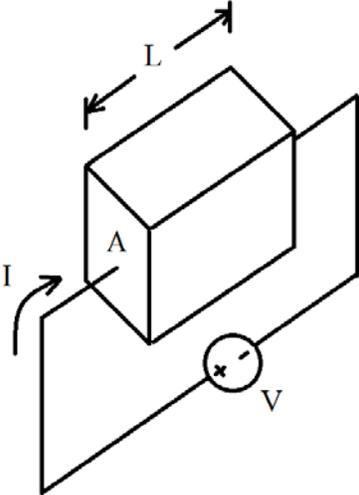


Figure 5: Material at Rest Dimensions

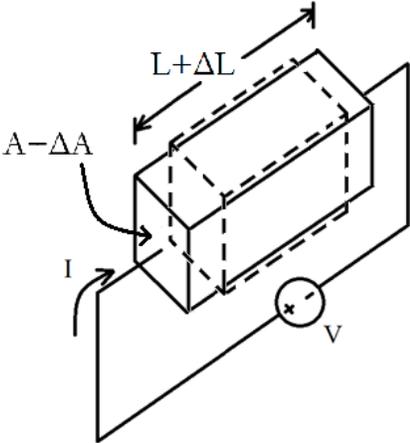


Figure 6: Material Experiencing Strain

Strain gages use the principle of a devices resistance being dependent on the devices geometry to sense the strain applied to them based on the equation^[7]:

$$R = \rho \frac{L}{A} \quad (1)$$

Where the resistance R is equal to the resistivity ρ multiplied the length of the device L and inversely proportional to the area A . A strain applied to the device causes an increase in the length of ΔL of the resistive material and results in an increase in resistance. Mass must be conserved so the area decreases accordingly by ΔA . This process is described by the equation:

$$R = \rho \frac{L+\Delta L}{A-\Delta A} \quad (2)$$

By the same token when a strain is applied in the opposite direction it causes a compression in the material resulting in a decrease in resistance. The resistance can be recast as

$$R = \rho \frac{L}{A} \left(1 + \underbrace{2}_{GF} \frac{\Delta L}{L}\right) \quad (3)$$

Strain gages have a gage factor (GF), which is the ratio of the relative change in electrical resistance to the corresponding applied mechanical strain. The equation for gage factor is given below^[8]:

$$GF = \frac{\Delta R/R_G}{\epsilon} \quad (4)$$

The term ΔR is the change in resistance caused by strain is, R_G is the undeformed resistance of the gage, and ϵ is the strain. The typical gage factor for a metal foil strain gage is 2 while other types of strain gages such as. The gage factor is what determines the sensitivity of the strain gage. The derivation of the number comes from the mercury-in-rubber strain gages where the volume of the strain gage remains that same even though the length changes. The mercury-in-rubber strain gage was used to determine respiration rates and still has some applications in medicine today.

The assumption that the volume of the resistive material will remain constant allows for the derivation of the gage factor for a typical metal foil strain gage. By using equation (1) and multiplying the top and bottom by the length over the length results in:

$$R = \rho \frac{L^2}{V} \quad (5)$$

By taking the derivative of the resistance with respect to the length results in:

$$\frac{dR}{dL} = 2\rho \frac{L}{V} \quad (6)$$

Then by multiplying equation (6) by length over length results in the following expression:

$$\frac{dR}{dL} = 2\rho \frac{L^2}{LV} \quad (7)$$

It can be seen that part of equation (7) takes the form of equation (5), such that equation (5) can now be substituted into equation (7) which results in the following expression:

$$\frac{dR}{dL} = 2 \frac{R}{L} \quad (8)$$

Equation (8) can then be manipulated to achieve the resultant expression which is equation (6) the gage factor:

$$\frac{dR/R}{dL/L} = 2 \text{ or } GF = \frac{dR/R}{\epsilon} \quad (9)$$

Typical strain gage construction consists of a metal foil that is in a pattern similar to that of Figure 7 shown below. This foil is attached to a substrate on a metal beam using a special adhesive. There are also piezoresistor strain gages made from semiconductors have much higher gain factors which result in higher voltage changes due to greater change in resistance. The piezoelectric effect is the phenomenon where an electric signal is produced due to the deformation of a crystal such as quartz. The output level of the signal corresponds to the amount of mechanical stress or strain applied. Unfortunately, piezoresistive strain gages also have higher temperature dependence, and are more fragile and more expensive than their metal foil counterparts^[9].

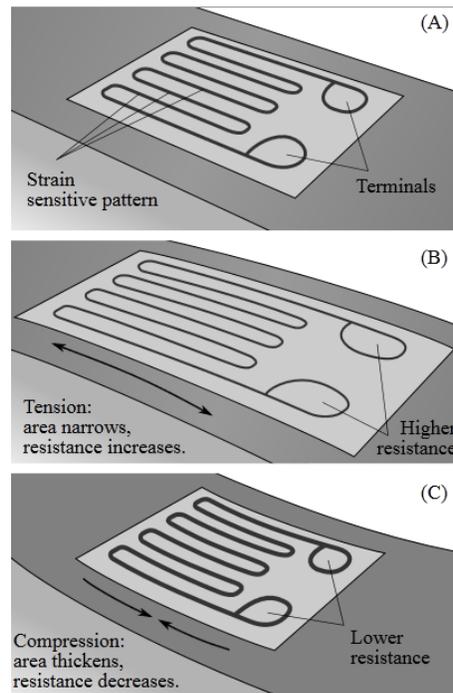


Figure 7^[10]: Metal Foil Strain Gage

Since the strain gage only has a minimal change in resistance they are used in a Wheatstone bridge configuration which is then tuned so that the voltage across the at-rest resistance of the strain gage is equivalent to the voltage across the voltage divider in the other branch of the bridge. The traditional Wheatstone Bridge configuration is shown below where the resistors R1, and R2 are equivalent to split the excitation voltage of the bridge and R3 is typically a potentiometer that is used to calibrate the branch of the bridge with the strain gage such that when the gage is at rest the voltage measured across the bridge is 0V. The resistor R4 is representative of the strain gage.

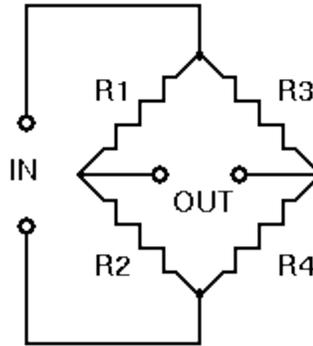


Figure 8^[11]: Wheatstone Bridge

The expression that represents the output voltage of the system is the following, note that R4 will be replaced with R_G to represent the strain gage resistance:

$$V_{out} = \left(\frac{R_G}{R_3 + R_G} - \frac{R_2}{R_1 + R_2} \right) \cdot V_{in} \quad (10)$$

By knowing that $R_G = R_3$, $R_1 = R_2$, and the fact that the resistance of the strain gage will be its nominal resistance plus its change in resistance, through mathematical manipulation results in the following expression:

$$\frac{V_{out}}{V_{in}} = \frac{GF \cdot \epsilon}{4 \left(1 + \frac{GF \cdot \epsilon}{2} \right)} \quad (11)$$

This means that the sensitivity of the system is purely dependent on the gage factor and the supply voltage. These two signals are then inputted to an instrumentation which will reject the balanced DC voltage set by the Wheatstone bridge and amplify the changing voltage. This phenomenon is due to the common-mode rejection behavior of the instrumentation amplifier.

Strain gages have quite a broad frequency response range and can resolve changes between 1Hz and 100kHz.^[12] Due to strain gages relatively light mass they can be used in applications where any

increase in mass will cause a large increase in the dampening factor of the system. Strain gages are typically bonded to surface using cyanoacrylate glues otherwise known as “super-glues.” Generally, a typical single package strain gage can range in price from \$8.35 (£5.20) to \$15 depending on the manufacturer and gage’s size and sensitivity^{[13] [14]}.

Although strain gages have some distinct advantages there are applications in which they may be unsuitable. They cannot sense changes in relative displacement that occur when the entire beam is moving uniformly because there is very little or no strain. In this instance an accelerometer would be necessary to implement to determine if the entire system is moving not just a single portion. Some errors that are associated with strain gages are temperature dependencies that occur and result in a change in the sensitivity of the gage. Linearity errors can occur when the sensitivity of the devices changes over certain portions of the pressure range of the gage. Since the strain gage is essentially a material that is being stressed and relaxed if the applied strain is greater than the material’s elastic modulus it results in plastic deformation. The plastic deformation that results because of overloading will degrade gage performance over time and never recover fully as the result of hysteresis. Strain gages sensitivity can also be affected by the ambient humidity which may cause parasitic resistances and wires and the substrate that the strain gage is adhered to.

In a previous design consideration the team came up with the idea to replace the branch of the Wheatstone bridge used to create the common-mode voltage and replace this input to the instrumentation amplifier with a D/A converter output voltage. At the same time it seemed to be better to move towards a dual-supply voltage because the output range of the AD8421 in-amp that is going to be used is limited to:

$$\text{Output}_{\min} = -V_s + 1.2V$$

And the maximum out voltage is:

$$\text{Output}_{\max} = +V_s - 1.6V$$

This was suggested because it allows for the bridge to be balanced through the use of a microcontroller and not have to worry about the differences cause by the tolerances of the resistors in the Wheatstone bridge. In addition, since the reference input to the AD8421 in-amp needs to be buffered and using the reference voltage reduces the common-mode rejection ratio of the in-amp it would increase of our signal integrity and possibly reducing power consumption.^[15]

However, this setup would put considerably more strain on the D/A converter because it would need more bits to accurately balance the bridge. For example, if the difference in voltage at the Wheatstone bridge had an imbalance of $20\mu\text{V}$, the instrumentation amplifier gain is set to 100V/V , and the supplies were $\pm 5\text{V}$ the necessary number of bits can be determined by the following equation:

$$\text{LSB} = \frac{V_{\text{FSR}}}{2^n - 1} = \frac{(+V_s) - (-V_s)}{2^n - 1}$$

In this instance the LSB is the $20\mu\text{V}$ offset voltage that needs to be calibrated out and the full-scale range voltage is 10V :

$$20\mu\text{V} = \frac{10\text{V}}{2^n - 1}$$

By solving for n yields:

$$n = \frac{\log(500001)}{\log(2)} = 18.93 \approx 19 \text{ bits}$$

There is also a current constraint on the system because although USB will supply a constant 5V supply the guaranteed current is only 100mA (although it is possible to get a maximum of 500mA). Due to this fact the DC-to-DC converters that would be needed to create the dual supplies were a source of inefficiency and added unnecessary complexity to the system.

Upon further consideration, it was decided to return to the single 5V supply and inputting a reference voltage to the AD8421 to have the output ride on an offset so that it can properly oscillate. Since the head-room and leg-room for the AD8421 at a 5V single supply constrained us to a maximum output value of 3.4V and a minimum output of 1.2V without clipping to get an equal amount of swing in both directions the offset needs to be at 2.3V . If we go through the calculation presented previously for determining the number of bits needed to resolve a $20\mu\text{V}$ bridge offset but now adjusting for the offset at the reference pin after the gain results in:

$$\text{LSB} \cdot \text{Gain} = \frac{V_{\text{FSR}}}{2^n - 1}$$

Knowing that the full-scale range voltage $V_{FSR} = 5V$ and that the gain will be $100V/V$ the resulting number of bits needed to resolve this signal is now:

$$20\mu V \cdot 100 = \frac{5V}{2^n - 1}$$

By solving for n yields the number of bits needed for the D/A converter:

$$n = \frac{\log(2500)}{\log(2)} = 11.287 \approx 12 \text{ bits}$$

This significantly reduces the constraints on the D/A converter, although it does require the use of a buffer to the input of the reference pin on the AD8421. However, the resistors in the Wheatstone bridge need to have a high enough resistance such that it reduces power consumption while not introducing too much unwanted noise that could corrupt the data acquisition process. It is also necessary to have the other branch of the Wheatstone bridge to utilize the common mode voltage that it generates because there may be fluctuations on the power supply or other noise on the system that may not be rejected if a D/A converter was used for the other input of the instrumentation amplifier.

3.2 The Instrumentation Amplifier

The instrumentation amplifier is a crucial design aspect in a sensor application such as this one. The instrumentation amplifier takes the differential signal generated by the Wheatstone bridges and amplifies them. The differential-to-single ended amplification performed by the instrumentation amplifier is necessary to strip away the DC offset of the Wheatstone and amplify the signal created by the strain gages.

Due to the sensitive nature of the application it is desired to avoid injecting any additional signal noise into the system. At the same time, having high common-mode rejection across the entire frequency range of interest is needed to ensure that the DC offset voltage inherent to the Wheatstone bridge is removed. The low-noise, high gain, high bandwidth, and low power consumption constraints made the selection of in-amp a key design choice. As a result, project sponsor Analog Devices provided their best instrumentation amplifier available in these areas, the AD8421. The $3nV/\sqrt{Hz}$ maximum input voltage noise and $200fA/\sqrt{Hz}$ current noise at 1kHz are the lowest in the ADI product catalog. At the same time it provides the best gain and bandwidth

characteristics among all of the available instrumentation amplifiers and was developed specifically for applications vibration analysis.

The expected signal range of inputs to AD8421 is typically very small on the order of millivolts. Although, all the other characteristics of the AD8421 are optimal it lacks a rail-to-rail signal swing which is required to fully utilize the signal dynamic range. The maximum output voltage with the given rails is 3.4V and the minimum output voltage is 1.2. In order to get an equal amount of signal swing in both directions it needs to be riding on a 2.3V DC offset which is input to the reference pin of the AD8421. The head and leg room lost with AD8421 is recovered through the use of a rail-to-rail amplifier which will be discussed later. With a $1.1V_{\text{peak}}$ signal swing and gain of $100[V/V]$ the expected max value inputs without clipping is $11mV_{\text{peak}}$. The smallest resolvable signal is determined by the overall gain of the system and the resolution of the A/D converter that it is being input to. The gain can be adjusted if necessary to detect smaller strains.

3.3 Accelerometers

Initially when research was being conducted it was found that the most common sensor for vibration analysis is an accelerometer^[16]. Accelerometers can only resolve certain frequency ranges but the ranges of frequencies that can be sensed are still well within the frequency range of interest for most mechanical systems. However, they are much more expensive than a strain gage. Some industrial accelerometers can be in the range of hundreds of dollars.^[17] Due to the project budget these are not a viable option. There are some MEMs based solutions that are only a few tens of dollars that would be suitable for an application such as vibration analysis. It was determined that the frequency range of interest for this particular application was between 10Hz and 10kHz.^[18] The best fit for the price and application was the ADXL001, a MEMs based accelerometer developed by Analog Devices which has a bandwidth of 22kHz, operates at 5V, and outputs an analog voltage. The ADXL001 has a sensitivity of 16mV/g where 1g is equal to $9.81m/s^2$. As an example, if the acceleration felt by the ADXL001 is 0.1g then the output voltage will be 1.6mV. This sensitivity can be increased through amplifying the output voltage. This accelerometer facilitates inexpensively acquiring acceleration data and showcases another product in Analog Devices' catalog.

At this point, some background information regarding accelerometers will be provided. Accelerometers, in general, are more versatile than metal foil strain gages in the sense that some models may have analog outputs and others may have digital outputs. This provides additional flexibility over a broad range of sensing applications. There are capacitive, piezoresistive and MEMs

based accelerometers. Capacitive accelerometers use silicon micro-machined sensing elements and are unmatched in performance at lower frequencies and operated in servo which is an error-correcting system that uses negative feedback which allows for linear and highly stable operation. MEMs based accelerometers are the simplest accelerometer topology. They consist of a cantilever beam and a test mass that acts as a reference. Piezoresistive accelerometers are the most suitable for high frequency applications and are known for the low package weight and wide temperature range^[19]. Despite their advantages, accelerometers can be cumbersome when trying to implement because they are much heavier than strain gauges. This excess weight can cause unwanted dampening affects in the system skewing data measurement. There are also issues with MEMs based accelerometers in that they require a break-out board. This adds to the complexity of implementation and introduces additional unwanted weight to the system.

The accelerometer is typically more practical in applications where the object in question is moving but the strain on the object may be negligible. Accelerometers measure the “proper acceleration” of an object which is associated with the weight of a test mass. Proper acceleration is the physical acceleration experienced by an object relative to free-fall or some inertial frame of reference. Accelerometers do not feel the acceleration due to gravity so when they are at rest the read zero since gravity is the only acceleration present. Conceptually, an accelerometer functions like a damped mass on a spring. When there is acceleration the mass is displaced, the displacement corresponds to the amount of force applied and the mass is known so the acceleration can be calculated using the formula:

$$F = ma$$

More specifically, the ADX001-70 is a MEMs device which is constructed as shown in Figure 9.

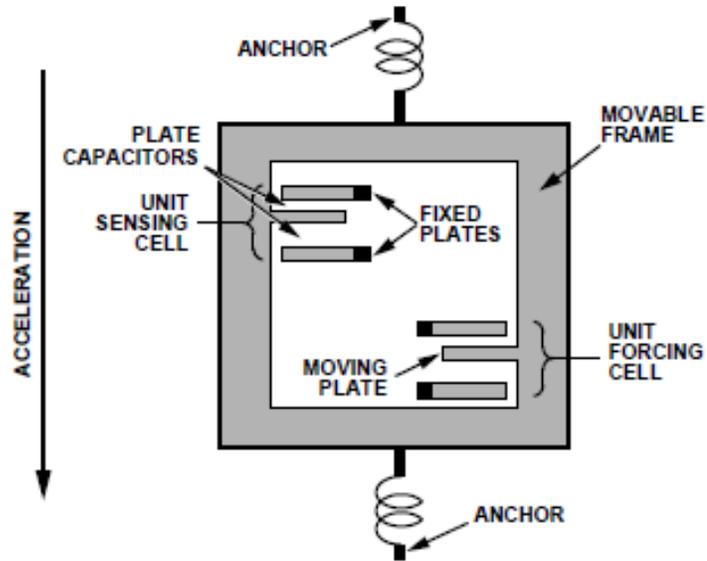


Figure 9: Internal Construction of the ADXL001-70^[20]

Figure 9 shows the damping springs, and the plate connectors which comprise the unit forcing cell. When the device experiences an acceleration, the moving plates cause a change in capacitance in each unit forcing cell. The capacitance changes according to the fundamental equation for a plate capacitor

$$C = \epsilon \frac{A}{d}$$

The area A of the plates remains constant along with the dielectric constant ϵ of the insulator. This results in a capacitance that is proportional to the plate separation d . The overall construction of the ADXL001-70 is shown in Figure 6.

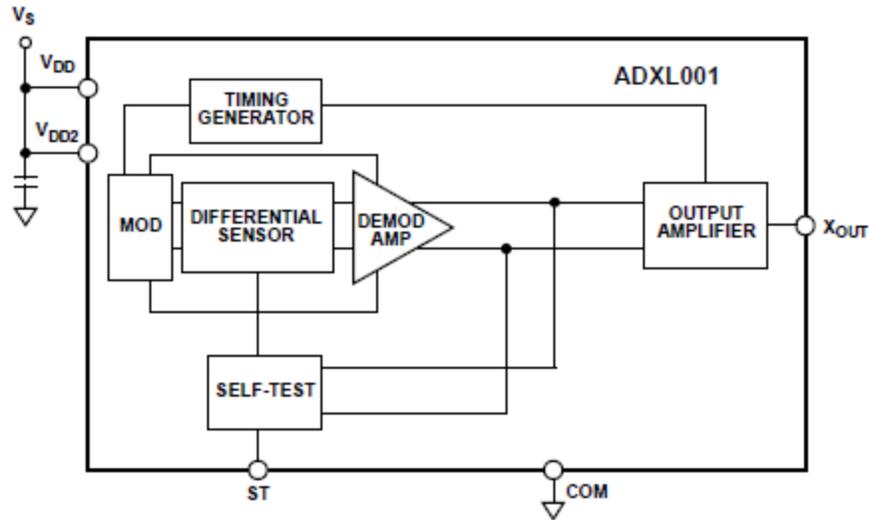


Figure 10: Functional Block Diagram of ADXL001-70^[21]

The output of the differential sensor is amplified internally by the ADXL001-70's output amplifier and fed to the X_{OUT} terminal of the device. This particular model provides an analog signal output rather than a digital output.

To determine the external amplification required, a simple test was performed using the ADXL001 with the use of a fan to excite it and see the resultant waveforms. The output is shown in Figure 11 below. The output of approximately 2mV peak for a low excitation suggested that a gain of 100V/V would be optimal to get a varying range of accelerations.

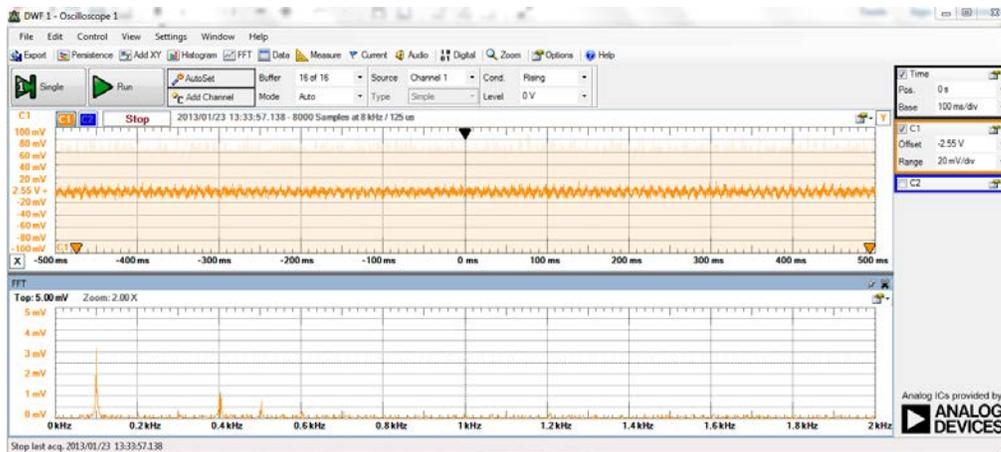


Figure 11: Preliminary Testing to Determine Accelerometer Gain

3.3.1 Post-Amp and Gain for Accelerometers:

Our system has a limited maximum voltage swing of 0 to 5V if the full supply range is utilized. The AD8421, however, is not a rail-to-rail amplifier. To cope with this limitation the, AD8608 quad rail-to-rail op-amp will be implemented to recover the lost head and leg room. This will increase the range of detectable acceleration values. The AD8608 has a 10MHz unity gain frequency, an open-loop gain of $1E+6V/V$, low noise characteristics comparable to the AD8421 of $8nV/\sqrt{Hz}$, and can swing from 0.02V to 4.98V at the output with the previously stated voltage rails.

The post-amp will also be used to implement a bandpass filter that will serve as an anti-aliasing filter for the system. This is necessary to remove any noise that may have been injected into the system that is above our desired signal range of 10Hz-10kHz. It will also serve to strip the initial DC bias from the signal so that a new bias may be added. Similar to the AD8421, a reference voltage needs to be provided to the AD8608 such that it can ride on a DC offset. A 2.3V reference allows for the output voltage swing to utilize the full dynamic range provided by the single-sided supply. A post-amp DC gain of 2V/V allows the majority of the gain to be implemented using the AD8421. This exploits the high common mode rejection ratio of the in-amp which, according to the graph in Figure 12, increases with circuit gain.

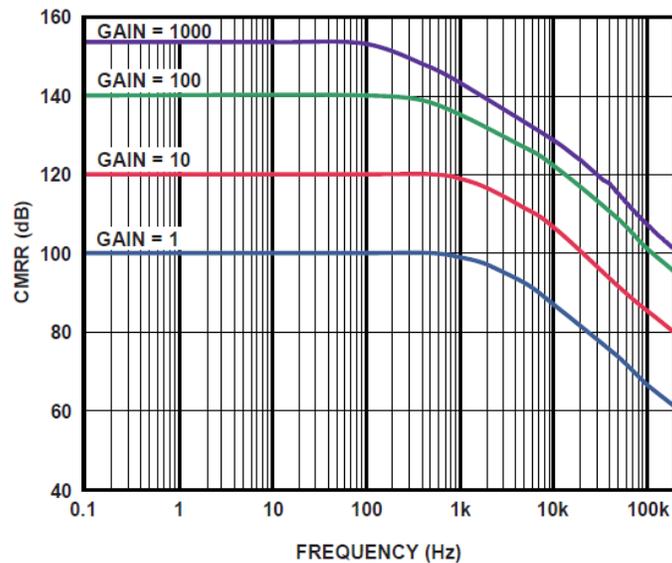


Figure 12: Plot of CMMR as a Function of Frequency for Different Gain Values^[22]

The same post-amp configuration will be used for the accelerometer signal chain, however, the gain will be different. The upper cutoff frequency for the bandpass filter will remain the same. The

gain for the accelerometer will be set to 100V/V initially in order to allow for reasonable amplification while not clipping allowing for the sensing of mid-range accelerations. If the need arises, the gain can be increased to allow for the sensing of even smaller magnitudes of acceleration or decreased to avoid distortion from clipping.

3.4 8-to-1 Multiplexer

Due to the eight sensors and only a single usable analog input channel to the Analog Discovery it is necessary to multiplex the channels to acquire data. Since there are 8 inputs there need to be 3 digital inputs that control which input is selected. The multiplexer needs to be fast enough such that the transition times are small enough to not alter the data that is being sampled. This aspect is key in ensuring that the data for one sensor isn't accidentally being captured in the data for another. Since direct control over the timing is necessary the Analog Discovery will provide the repeating clock pattern to switch between channels 0-7. The other analog input of the Analog Discovery will be used to look at the MSB of the control lines and use that information to synchronize the data acquisition.

The ADG709 8-to-1 analog multiplexer from Analog Devices was chosen to accomplish this task. The ADG708 has a 3Ω on resistance, 3dB bandwidth of 55MHz, maximum signal swings from V_{SS} to V_{DD} , and a power consumption of only $5\mu W$. With the given specifications and the power constraints of the system the ADG708 is the best fit for the system.

3.5 MATLAB

Once the data is acquired and saved into a .csv file MATLAB will be used to post process the data and display the relevant information in a significant manner. The .csv file will contain the data from all sensors and by using the MSB data, MATLAB will determine which sensor corresponds to which samples because a transition from low-to-high will indicate sensor 4 as the data at the corresponding sample and a transition from high-to-low indicates that sensor 0 corresponds to that data sample. Since the channels are known to be changing at a certain rate the number of samples and the samples corresponding to each channel can be determined and put into their own data vectors. Once the data for each sensor has been sorted, it can then be used to determine the frequency at which they are vibrating by taking the Fourier transform and the relative maximum magnitude of each sensor can be compared. For the accelerometers, the sensors that have the highest magnitude will indicate which screws are loose. For the strain gages, the sensors opposite of the loose screw will display the highest magnitude signal. Additionally, the real and imaginary

portions of the magnitude from each sensor can be used to reconstruct the mode shapes of the given plate showing the material behavior at each resonant frequency.

4 System Requirements and Detailed Design

The implementation of each of the system blocks discussed in Chapter 3 will be discussed in detail in this chapter. Special attention will be paid to the operation of each of the sub-circuits within each block. Additionally, expected signal levels and their governing equations will be elaborated upon for the purpose of determining system requirements. These requirements will be used to finalize design details.

4.1 Mechanical System

To simulate a typical mechanical application of the system, a test apparatus was constructed using a steel sheet metal plate as the test surface. A button cell motor was attached to the underside of the plate to mechanically excite the system. Screws were inset 0.5 in. from the corners of the plate.

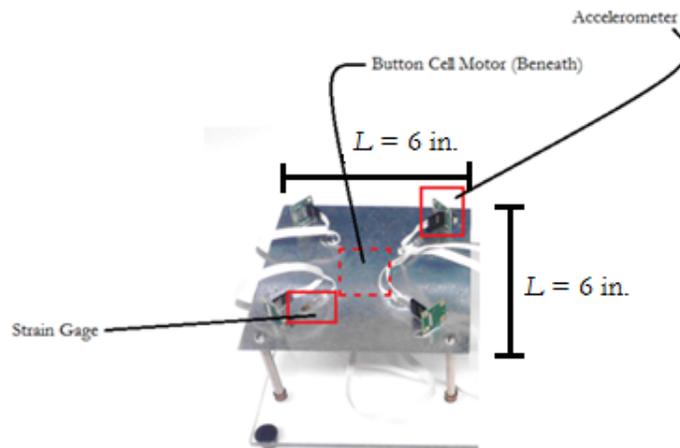


Figure 13: Mechanical Test Apparatus

The plate dimensions were chosen to be $L = 6$ in. Using a piece of acrylic as the base, the system then fixed using the screws in each of the four corners.

Table 1 shows a summary of system dimensions and sensor placement.

Plate Side Lengths	15.24 cm
Screws	1.27 cm inset from plate sides
Strain Gages	3.5 cm from screw along diagonal of plate
Accelerometers	1.5 cm from screw along diagonal of plate

Table 1: Summary of Mechanical System Dimensions and Sensor Placement

4.2 Data Acquisition

The data acquisition unit is the heart of our system. It converts the analog signals to the digital domain so that they may be manipulated in MATLAB. This is done mainly using the Analog Discovery's two analog inputs. One analog input is used to pass the analog signal from the analog multiplexer to the Analog Discovery's A/D converter and the other is used to monitor the MSB of the 3-bit control signal. The MSB data is then used to synchronize the data from the multiplexer output to the particular multiplexer channel.

4.2.1 Analog Discovery and Analog Multiplexer

The analog discovery is essentially an oscilloscope. At its heart is a 14-bit A/D converter with a full scale range of 1.8 V. In front of the A/D converter is a level converter of unspecified type which maps the input signal to the A/D converter's 1.8 V full scale range. The A/D converter has one severe limitation in that it is a single channel type. In order to remedy this an 8:1 analog multiplexer, the AD708G, is used to create a multiplexed A/D architecture of sorts.

Using the facts that the Analog Discovery's A/D converter is a 14-bit device with a 1.8 V reference, it is possible to determine the smallest detectable voltage, the LSB.

$$LSB = \frac{1.8 \text{ V}}{2^{14} - 1} = 110 \mu\text{V}$$

Therefore, for a signal to be visible to the Analog Discovery, it must meet two criteria. It must be larger than the noise floor, which will be discussed in more detail later, and it must be at **110 μV** or greater in magnitude.

The system must also take into account which frequencies are desired in this application. Pre-defined digital signals will be generated using the Analog Discovery via the WaveForms software that comes with the device. The control lines of the AD708G are driven using three digital I/O lines from the Analog Discovery with the LSB line running at 400 kHz. The diagram for the digital control lines is shown in Figure 1.

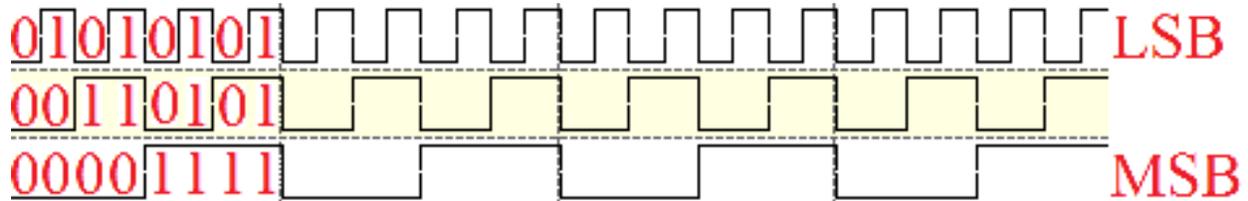


Figure 14: Control Line Signals

The multiplexing frequency of 400 kHz was chosen so that the Nyquist rate condition is satisfied for each individual channel. The highest frequency of interest on each individual channel is 10 kHz. To satisfy Nyquist this means that each channel must be sampled at a minimum of 20 kHz. When using a multiplexed A/D converter architecture a frequency of

$$n \times f_s = f_{mux}$$

where n is the number of MUX channels, f_s is the sample rate require for a single line, and f_{mux} is the sample rate required for the total system. In addition, f_{mux} should be given at least 15% margin. For our system, the minimum f_{mux} needed to satisfy Nyquist would be 160 kHz and our f_{mux} is 400 kHz.

Analog input 1 of the Analog Discovery is connected to the MUX output for acquisition of measured data. Analog input 2 is used to monitor the MSB of the digital signals generated by the Analog Discovery. Data collected from these two inputs are exported in the form of a .csv file using the WaveForms software and imported to MATLAB for parsing signal processing.

4.2.2 Synchronization

The .csv file imported into MATLAB consists of three columns: the time stamp, analog input 1 voltage, and analog input 2 voltage. In order to associate voltage data with a particular sensor or, in other words, a particular channel, some data parsing must be done. A MATLAB function was written to convert the MSB analog voltage levels from analog input 2 into logic levels using the

MUX logic levels for the AD708G from the datasheet. The code then looks for the first transition from 1 to 0 on the MSB and defines this sample as a sample from the 0th channel. The other channel numbers for particular voltages can then be found knowing f_{max} . The MATLAB functions used to implement data parsing can be found in Appendix A.

4.3 Analog Front End

The analog front end of this system consists of the sensors used to convert mechanical quantities into electrical signals and the amplifiers and filters to make those signals usable. This system can be decomposed into 2 separate subsystems: the strain gage signal chain and the accelerometer signal chain. The strain gage signal chain consists of the strain gage Wheatstone bridges, instrumentation amplifiers, and post-amp active filters. The accelerometer signal chain consists of the accelerometers and their post-amp active filters. The signal levels and operation of each of these subsystems will be discussed in this section.

4.3.1 Strain Gage Signal Chain

To create a measurable output signal for the strain gages, the change in resistance must first be converted to a change in voltage. This is accomplished by placing a strain gage in one leg of a balanced Wheatstone bridge such as the one shown in Figure 15.

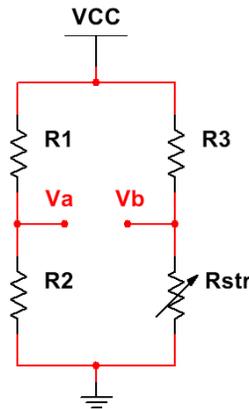


Figure 15: Four Resistors in Wheatstone Bridge Configuration

When the strain gage is in the resting state, the differential output of the bridge should be 0 V. This case is described mathematically by the following conditions

$$R1 = R2 = R3 = Rstr = R_0 \text{ and } Vb - Va = 0$$

When the condition

$$R_{str} = R_0 + \Delta R, \text{ where } \Delta R_{max} < 1\Omega$$

Recalling that

$$V_{diff} = \left(\frac{R_G}{R_3 + R_G} - \frac{R_2}{R_1 + R_2} \right) V_{CC}$$

and that

$$\frac{V_{diff}}{V_{CC}} = \frac{GF \cdot \varepsilon}{4 \left(1 + \frac{GF \cdot \varepsilon}{2} \right)}$$

Therefore the relationship between strain and the differential bridge voltage is

$$V_{diff} = \frac{GF \cdot \varepsilon}{4 \left(1 + \frac{GF \cdot \varepsilon}{2} \right)} V_{CC}$$

The post amplifier that will be used to restore the voltage swing at the output to 0-5V needed to have as stringent noise characteristics as the AD8421 while still having a sufficiently high CMRR and a very low offset voltage and input bias current. This resulted in the AD8608 rail-to-rail CMOS op-amp being selected as the amplifier for this portion of the signal chain. The AD8608 has a max input bias current of 1pA, a max offset voltage of 65 μ V, a CMRR of approximately 85dB at 10 kHz, and an output voltage swing from 0.02 V to 4.98 V. Since the output of the AD8421 will have a reference voltage of 2.3 V and a maximum differential voltage of ± 1.1 V it is necessary to strip away the DC offset such that it is not amplified at the output. This is done by using the op-amp configuration shown in Figure 16.

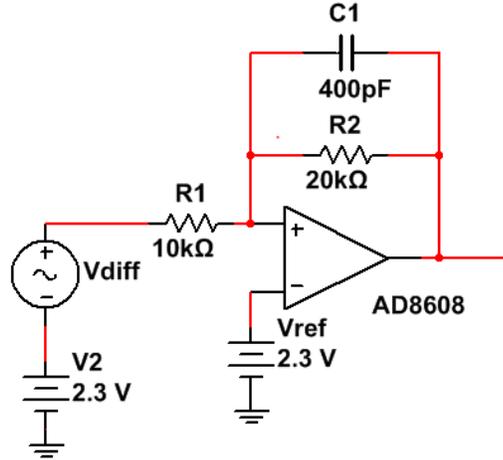


Figure 16: Strain Gage Signal Chain Post-Amp

The derivation of the output voltage V_{out} can be found using KCL. Since the op-amp is connected for negative feedback and the op-amp open-loop gain is sufficiently high the assumption that the op-amp input voltages are equal can be made. This results in the following expression

$$\frac{V_{ref} + V_{diff} - V_{ref}}{R_1} = \frac{V_{ref} - V_{out}}{R_2}$$

Where V_{ref} is the DC offset voltage that the D/A converter outputs to the in-amp reference input, V_{diff} is the voltage generated by the differential voltage caused by the strain gage. By solving for V_{out} the following expression is found which indicates that the output voltage will be the amplified differential voltage riding on the DC offset of V_{ref} .

$$V_{out} = -\left(\frac{R_2}{R_1}\right) * V_{diff} + V_{ref}$$

This means that in order to make sure the DC offset voltage is not amplified the V_{ref} voltage from the in-amp must also be used here. Since the gain of this op-amp will be two the op-amp offset at the output needs to be 2.5V to get the maximum voltage swing of 0.3V to 4.7V. This voltage will then be fed through an anti-aliasing low-pass filter and inputted to the A/D converter.

The analog discovery has an internal 14-bit A/D converter indicating that the least significant bit for the A/D converter is:

$$LSB = \frac{V_{FSR}}{2^n - 1} = \frac{1.8V}{2^{14} - 1} = 110 \mu V$$

This limits the smallest resolvable strain to:

$$G_{in-amp} G_{post-amp} V_{CM} \epsilon_{min} = 110 \mu V$$

Since the gain of both the AD8421 and AD8608 are known to be 200V/V and 2V/V respectively while the common-mode voltage is 2.5V the minimum resolvable strain is equal to:

$$\epsilon_{min} = \frac{110 \mu V}{2 \cdot 200 \cdot 2.5V} = 0.110 \mu \epsilon$$

The maximum resolvable strain can be found because the maximum change in output voltage is known to be 2.2V based on the output limit of the in-amp that will be gained up by the post-amp. The equation for the maximum resolvable strain is found to be:

$$G_{in-amp} G_{post-amp} V_{CM} \epsilon_{max} = 2.2 V$$

By then rearranging the equation to solve for the max strain results in:

$$\epsilon_{max} = \frac{2.2V}{2 \cdot 200 \cdot 2.5V} = 2.2 m\epsilon$$

It should be noted that if we wish to detect even smaller strains the gain can be adjusted as needed such that the system can increase its resolution up to a certain point without causing clipping and distorting the signal.

4.3.2 Accelerometer Signal Chain

Our system has a limited maximum voltage range from 0 to 5 V and the AD8421 would not be able to swing to from rail-to-rail. To cope with this limitation the AD8608 quad rail-to-rail op-amp will be implemented in order to recover the lost head and leg room in the same fashion as the post-amp active filter shown in Figure 2. The AD8608 has a 10MHz unity gain frequency, an open-loop gain of 1E+6 V/V, low noise characteristics comparable to the AD8421 of 8 nV/ \sqrt{Hz} , and can swing from 0.02 V to 4.98 V at the output with the previously stated voltage rails.

The post-amp will also be used to implement the active first-order low-pass filter that will be used as an anti-aliasing filter. This is necessary to remove any noise that may have been injected into

the system that is above our desired signal range of 10 Hz-10 kHz. Similarly to the AD8421 a reference voltage needs to be provided to the AD8608 such that it can ride on a DC offset. Since the 2.3 V reference allows for the output voltage swing to go from a max of 4.5 V down to a minimum of 0.1V given the max expected input voltage swing being 1.1 V and the post-amp having a DC gain of 2 V/V. Although this offset is not in the middle of the voltage swing it still allows for an undistorted output because of its rail-to-rail capability.

The same configuration will be used for the accelerometer, however the gain will be different while the pole frequency for the anti-aliasing low-pass filter will be the same. The gain for the accelerometer will be set to 50 V/V in order to allow for reasonable amplification without clipping allowing for the sensing of mid-range accelerations. If the need arises the gain can be increased to allow for the sensing of even smaller magnitudes of acceleration.

4.4 Noise Analysis of Analog Front End

As mentioned above, our system includes two gain stages: A_1 representing first stage gain and A_2 representing second stage gain. Assume that the input noise to the first stage is e_{n1} and that of the second stage is e_{n2} . e_{n1} encompasses all the noise sources from the bridge as well as the noise source from the in-amp. e_{n2} includes all the noise between the first and the second stages.

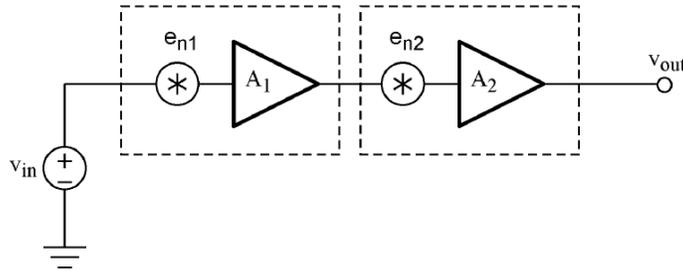


Figure 17: A Simplified Representation of the Gain Stages with Noise

e_{n1} sees the total gain of A_1A_2 while e_{n2} only sees the second stage gain. From these definitions, we can conclude that the output noise due to e_{n1} is:

$$e_{n1(out)} = A_1A_2e_{n1}$$

and that of e_{n2} is:

$$e_{n2(out)} = A_2 e_{n2}$$

The values of the noise sources add in RSS (Root of the Sum Squared) fashion:

$$e_{n(out)} = \sqrt{(A_1 A_2 e_{n1})^2 + (A_2 e_{n2})^2}$$

Referring the output noise to input results in division by the total gain:

$$e_{n(in)} = \frac{1}{A_1 A_2} e_{n(out)}$$

$$e_{n(in)} = \frac{1}{A_1 A_2} \sqrt{(A_1 A_2 e_{n1})^2 + (A_2 e_{n2})^2}$$

Factoring in the total gain into square root gives us:

$$e_{n(in)} = \sqrt{\frac{(A_1 A_2 e_{n1})^2}{(A_1 A_2)^2} + \frac{(A_2 e_{n2})^2}{(A_1 A_2)^2}}$$

which ultimately results in the input referred noise equation of:

$$e_{n(in)} = \sqrt{(e_{n1})^2 + \left(\frac{e_{n2}}{A_1}\right)^2}$$

This equation provides a useful insight for the noise analysis. If the first gain stage A_1 is relatively high, e_{n2} term becomes very minute that it can be ignored from the equation and the first stage gain becomes an approximation for the input referred noise equation. Since our first gain $A_1 = 200$ is relatively large compared to the second stage gain $A_2 = 2$, we can assume that the second stage noise is reduced by a factor of first stage gain and disregard it from the noise analysis. Thus, we consider the effects of the noise sources on the left of the first gain stage which includes the voltage and current noises from AD8421 as well as the thermal noise of the resistors from the Wheatstone bridge. The following curves provide the gain, CMRR, voltage noise, and current noise specifications of AD8421 instrumentation amplifier. The corresponding values are^[23]:

Gain	200 V/V or 46.02 dB @ f = 100 Hz to 100 kHz
CMRR	120 dB @ f = 10 kHz
Voltage noise spectral density	3.5 nV/ $\sqrt{\text{Hz}}$
Current noise spectral density	200 fA/ $\sqrt{\text{Hz}}$

Table 2: Summary of AD8421 Noise Specs.

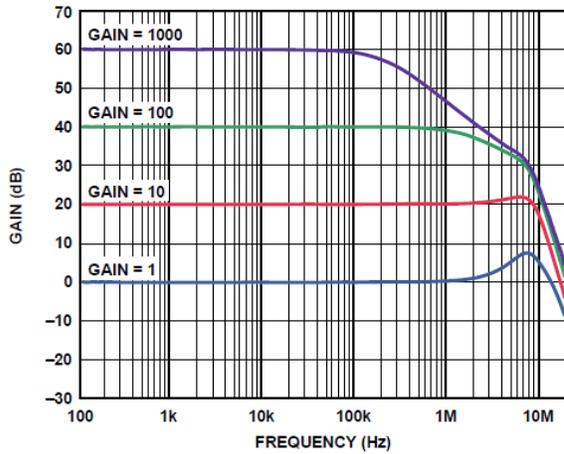


Figure 21. Gain vs. Frequency

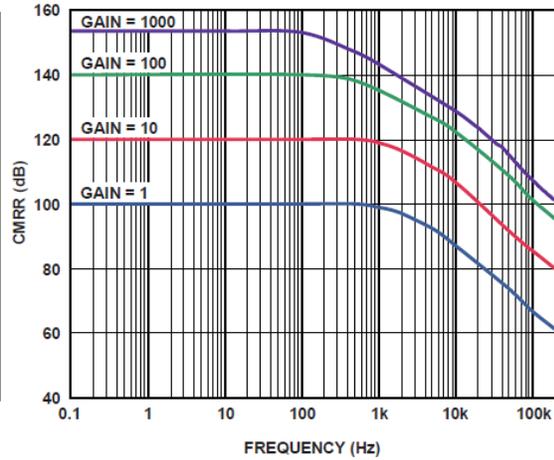


Figure 22. CMRR vs. Frequency

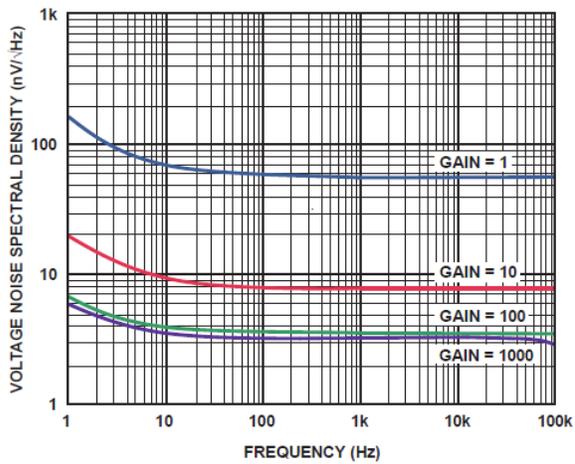


Figure 41. RTI Voltage Noise Spectral Density vs. Frequency

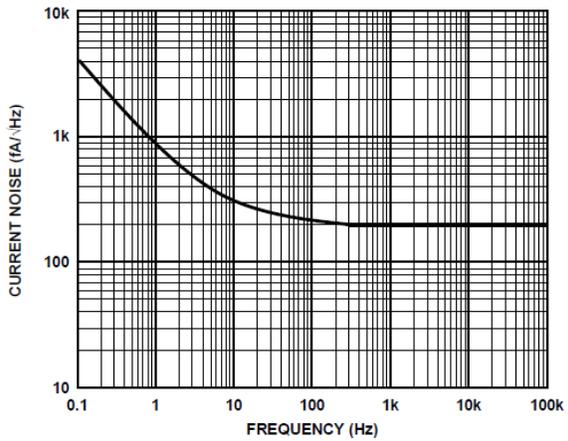


Figure 43. Current Noise Spectral Density vs. Frequency

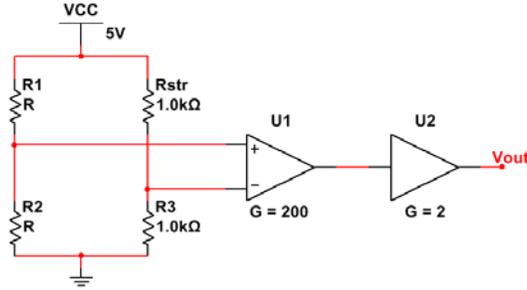


Figure 18: Simplified Circuit Diagram for Noise Analysis

First, assume that the left branch of the Wheatstone bridge uses two equal value resistors with resistance R . Resistors inherit the thermal noise of

$$e_{n(R)} = \sqrt{4kTR}$$

which implies that the amount of thermal noise varies directly with the square root of the value of the resistance. In other words, the smaller the resistance, the less noise it produces. k in the above equation is the Boltzmann constant, T is the temperature in Kelvin, and R is the resistance of the R1 and R2 resistors. Due to the strong dependence on the resistance, care must be taken in choosing the right value resistances. The smaller the resistance is, the lower the voltage we see at the output of the gain stage.

Another important bottleneck in choosing optimum resistance value is the supply current from the USB port. With the total series resistance of right side of the bridge is about 1 kΩ, adding resistor in parallel reduces the 1 kΩ to a lower value. In contrary to the above mentioned noise bottleneck, current limitation requires larger resistor values. The total resistance on the left side of the bridge is constant as we have decided on a 1 kΩ strain-gage.

Therefore, a quick analysis of the bridge resistance dependent output voltage and supply current was performed in MATLAB using the code provided below. Figure 19 (left) shows the total input referred noise due to the noise from resistors as well as op-amp voltage and current noises, assuming input to output signal gain is 400. Note that the noise values add in an RSS fashion.

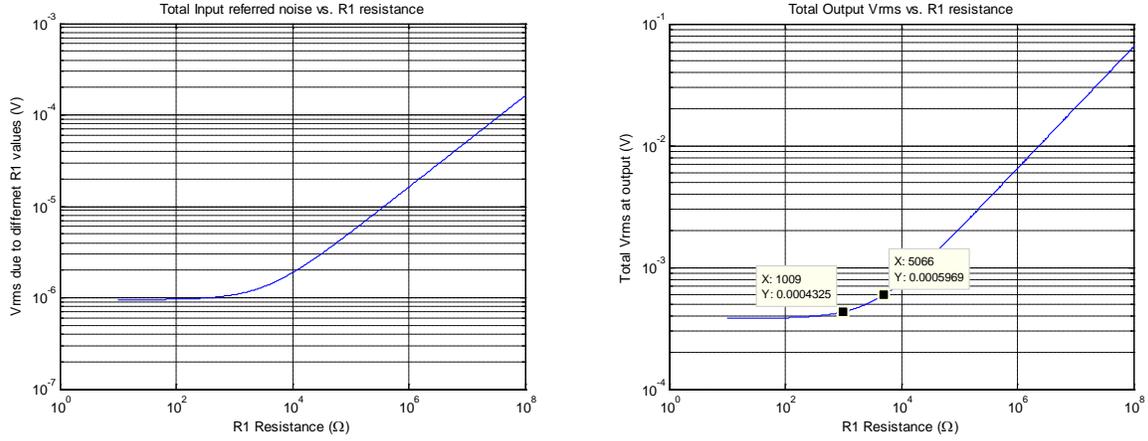


Figure 19: (left) Total Input Referred Noise Due to R1; (Right) Total Output Vrms Due to Noise Present in the Circuit

Figure 19 (left) also suggests that at low R1 values of less than 1 kΩ, noise characteristic is governed by the in-amp noises whereas at high R1 values of greater than 1 kΩ, resistor R1 noise greatly impacts the noise performance. It was calculated that a factor of 10 change in resistance (where R > 5 kΩ) results in a factor of 3 change in the noise performance. Figure 19 (right) shows the total output Vrms due to all noise sources according to the different values of R1. The output RMS voltage of a voltage noise source is calculated as

$$V_{rms,en} = Gain \cdot e_n \cdot \sqrt{\frac{\pi}{2} f_{3db}}$$

This equation suggests that the higher the 3dB frequency point is, the more the noise is present in the spectrum. Thus, an anti-aliasing active low-pass filter is placed at each output of the in-amps to limit the bandwidth of the system at $f_{3db} = 20$ kHz. The output RMS noise must be lower than the LSB of A/D converter which is about $q = LSB = \frac{1.8V}{2^{14}} = 110 \mu V$ for a 14-bit A/D converter. However, with the inherent noise floor of about 400 μV as shown in Figure 19 (right), the system will inevitably experience at least about 4 LSBs error solely from the noise.

Although resistance values lower than 5 kΩ would be preferred in order to minimize the noise effect, we have yet to consider the power supply limitations. The total current supplied by USB connector is very limited. According to Figure 20 (left), the current consumption from four bridges using 5 kΩ resistors for R1 and R2 is about 12 mA. However, this would increase the error due to noise to about 600 μV or about 6 LSBs.

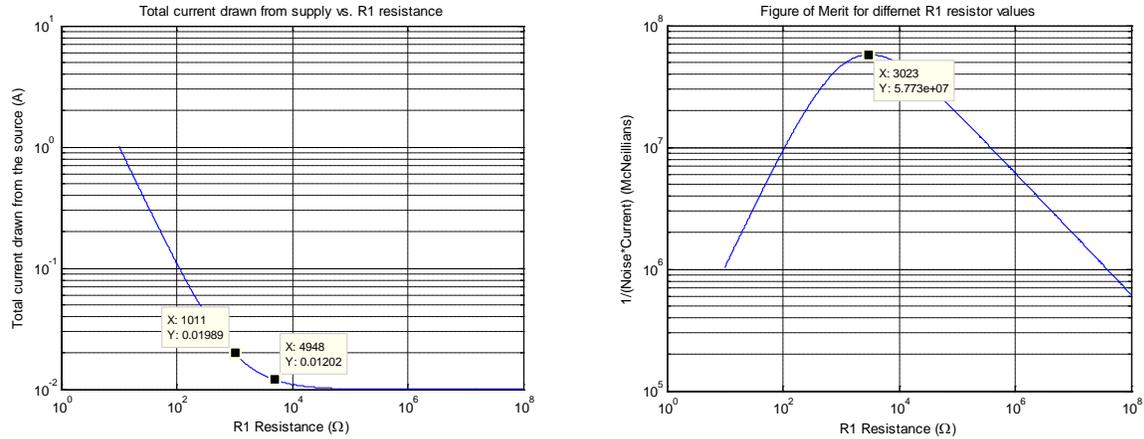


Figure 20: (left) Total Current Consumption from the Bridge; (Right) Figure of Merit Analysis

To find the definite optimum point for the trade-off between noise and the power consumption, a figure of merit analysis was designed with the following equation and plotted on Figure 20(right).

$$FOM = \frac{1}{\text{Input referred noise} \cdot \text{Total current consumption}}$$

It was found that the figure of merit peaks at R1 value of approximately 3 kΩ. Using the output RMS noise will also provide the same result since input referred noise is a fraction of output RMS noise. This analysis can be used to perform trade-offs when the power consumption from other components are known. Since the major portion of the noise comes from the fact that the 3dB bandwidth of in-amp and the gain are significantly high, care must be taken to select the correct resistor values and the total gain of the system. Judging from the analyses above, 1 kΩ resistors would be the best candidate because it would draw 20 mA, which is about 8 mA more than the circuit with 5 kΩ, but the total noise output RMS voltage would be reduced from 6 LSB to 4 LSB.

A/D converter Noise + Noise Analysis with bandwidth limitation

Since A/D converters are used in our system, the noise due to quantization must be considered. The root mean squared error and SNR of an ideal A/D converter are given as^[24]

$$\text{rms quantization noise} = \sqrt{e^2(t)} = \frac{q}{\sqrt{12}}$$

$$SNR = 6.02N + 1.76 \text{ dB}$$

where q is the LSB of the A/D converter and N is the number of bits of the A/D converter.

Since $q = LSB = 110 \mu\text{V}$, RMS quantization noise yields $\sigma = \frac{110 \mu\text{V}}{\sqrt{12}} = 31.7 \mu\text{V}_{\text{rms}}$ for our system. The detailed derivations of the above equations can be found in the reference given in the footnote. The quantization noise density for A/D converter with the new 3 dB bandwidth of 20 kHz becomes

$$\frac{\sigma}{\sqrt{\frac{\pi}{2} \cdot f_{3dB}}} = \frac{31.7 \mu\text{V}}{\sqrt{\frac{\pi}{2} \cdot 20 \text{ kHz}}} = 0.179 \frac{\mu\text{V}}{\sqrt{\text{Hz}}}$$

which sets the maximum noise density limit at the A/D converter input to avoid conversion errors.

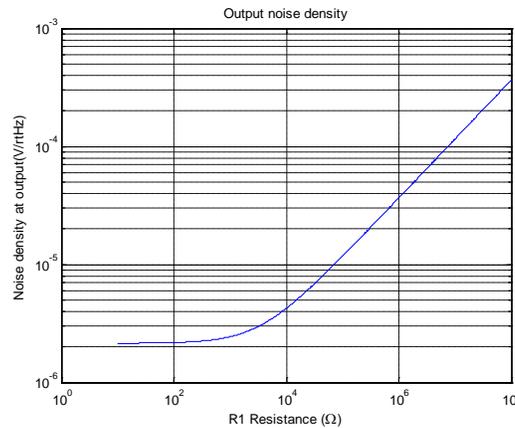


Figure 21: Total Output Noise Density Seen by A/D Converter

The total output noise density plot above shows that choosing any resistor would not improve the output noise density because the noise floor is set by the in-amp noise. In other words, there will always be at least one bit error in converting from an analog signal to a digital signal, although the result would be more accurate than using 12-bit A/D converter.

4.5 D/A Converter-Based Wheatstone Bridge Offset Nulling

In order to produce the voltages necessary for the offset adjustment, a D/A converter with SPI interface is considered. AD5684R^[25] from Analog Devices is a suitable candidate for our system as it is a 12-bit D/A converter with four rail-to-rail analog voltage output pins as there are four instrumentation amplifiers, all of which requires offset adjustment. Similar to typical 3-wire SPI interface devices, AD5684R requires a synchronization pin \overline{SYNC} , a clock source pin $SCLK$, and data input pin $SDIN$.

The AD5684R can be controlled by writing bit values to the 24-bit wide input shift register as can be seen in Figure 22. The first four bits DB23-DB20, counting from MSB, are the command bits, provided in Figure 23, which selects different operations necessary to control the D/A converter. The main command that is used in this project is 0011 which “writes to and updates DAC Channel n”. The channel is selected by the combination of address bits shown in Figure 24.

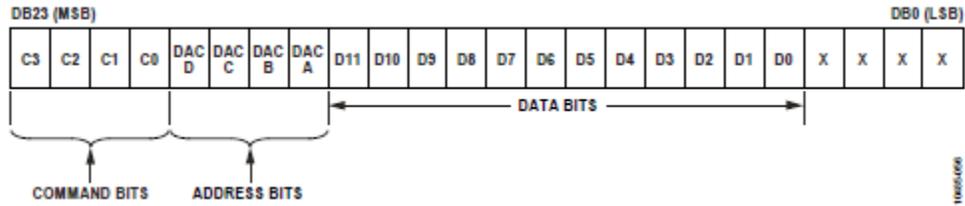


Figure 22: 24-Bit Wide Input Shift Register of AD5684R D/A Converter

Command				Description
C3	C2	C1	C0	
0	0	0	0	No operation
0	0	0	1	Write to Input Register n (dependent on $\overline{\text{LDAC}}$)
0	0	1	0	Update DAC Register n with contents of Input Register n
0	0	1	1	Write to and update DAC Channel n
0	1	0	0	Power down/power up DAC
0	1	0	1	Hardware $\overline{\text{LDAC}}$ mask register
0	1	1	0	Software reset (power-on reset)
0	1	1	1	Internal reference setup register
1	0	0	0	Set up DCEN register (daisy-chain enable)
1	0	0	1	Set up readback register (readback enable)
1	0	1	0	Reserved
...	Reserved
1	1	1	1	Reserved

Figure 23: Commands Used to Control the D/A Converter

Address (n)				Selected DAC Channel ¹
DAC D	DAC C	DAC B	DAC A	
0	0	0	1	DAC A
0	0	1	0	DAC B
0	1	0	0	DAC C
1	0	0	0	DAC D
0	0	1	1	DAC A and DAC B
1	1	1	1	All DACs

Figure 24: Address Bits Used to Select Particular D/A Converter

The following diagram shows the timing specification and the state necessary for each pin, especially \overline{SYNC} , $SCLK$, $SDIN$ and \overline{LDAC} . The synchronization pin, \overline{SYNC} , must go from high to low before data from $SDIN$ is transmitted. \overline{LDAC} must be brought from high to low then to high again to update and write to the selected DAC.

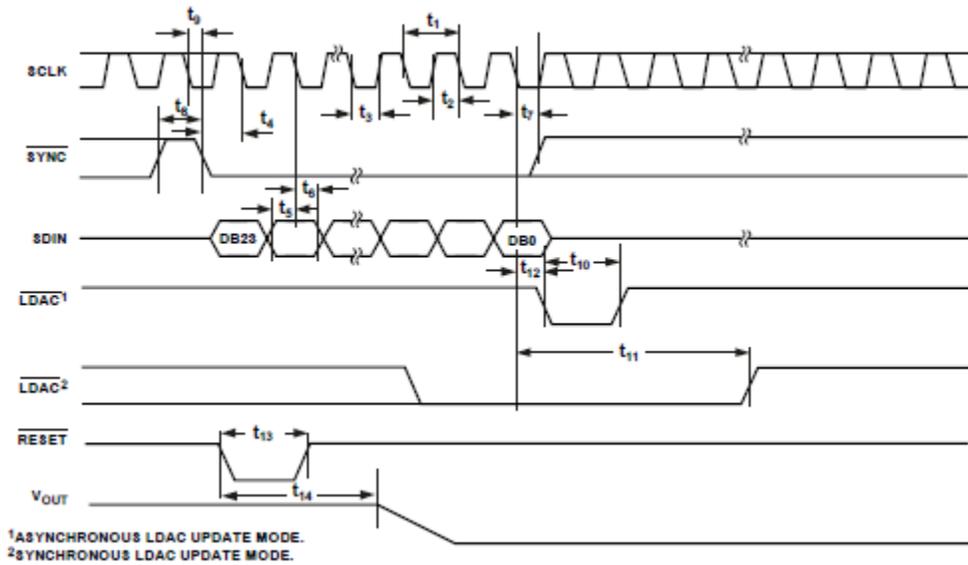


Figure 25: Timing Diagram for a Write Operation

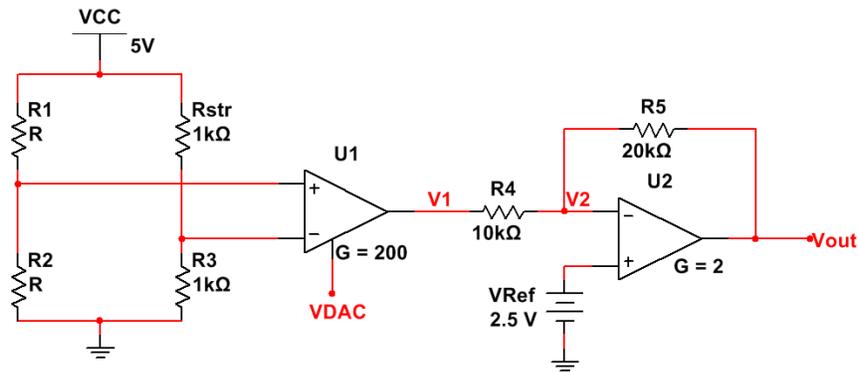


Figure 26: Simplified Circuit Diagram with Offset Adjustment

Figure 26 shows the simplified circuit diagram of the analog front end composed of a Wheatstone bridge, an in-amp, and a post-amp. To calculate the voltages necessary for the offset adjustment, the following calibration algorithm is utilized for each channel. Set the D/A converter output to 2.5V so that the in-amp output is not railing. Measure the voltage at Vout. Since the

voltage drop across R5 is half of R4, the voltage difference between Vout and V2 should be twice that of the voltage difference between V1 and V2, yielding the following equation:

$$V_{out} - V2 = 2 (V2 - V1)$$

Since the post-amp U2 is connected for negative feedback, the voltage at V2 should be close to VRef, which results in

$$V_{out} - 2.5V = 2 (2.5V - V1)$$

If the common mode voltage is present in the Wheatstone bridge, the voltage at the output of the in-amp, V1, will have an initial error of

$$V_{err} = \frac{VDAC}{2.5V} - V1$$

Thus, the voltage required at the VDACC or the reference pin of in-amp to calibrate the Vout to be biased at 2.5V is

$$VDAC_{new} = \frac{V_{out} - \underbrace{VREF}_{2.5V}}{2} + 2.5V$$

The complete microcontroller code that performs the mathematical calculations as well as communication with the D/A converter is provided in Appendix B.

4.6 System Requirements and Final Design

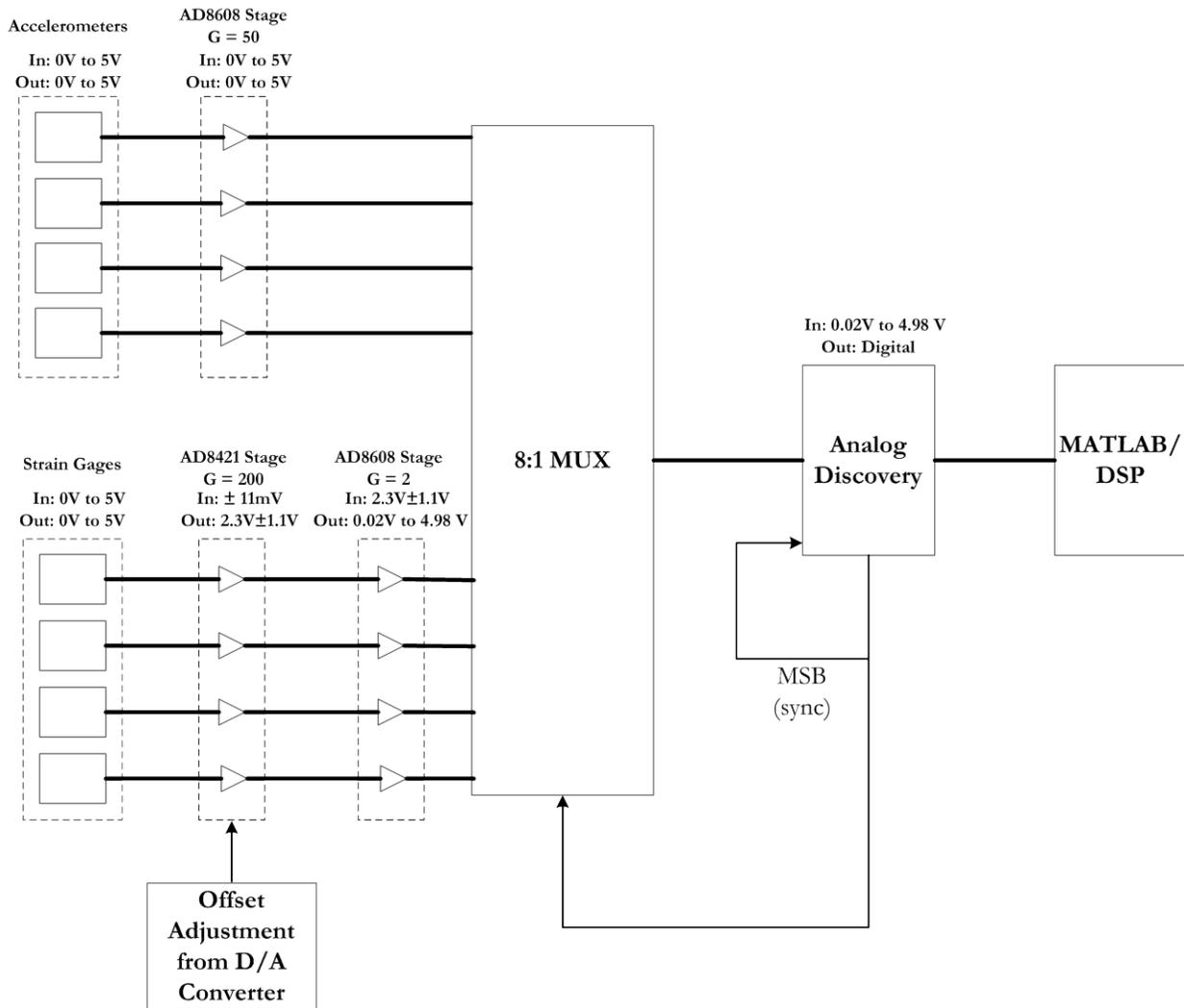


Figure 27: System Block Diagram with Signal Levels

Figure 27 shows the system level block diagram with input and output signal levels for each block. Starting from the A/D converter in the Analog Discovery and the sensor requirements for the system, the requirements of the analog front end were determined using the signal levels shown in Figure 27 in conjunction with the equations discussed in previous sections. Finally, the gain for each signal chain was determined for each of the signal chains.

5 Tests and Results

5.1 Signal Verification Tests

To verify proper operation of system circuitry, a battery of tests was performed. These tests were conducted on both the strain gage and accelerometer signal paths with the sensors disconnected from the system. Three basic waveforms were input to the system: A sinusoid, a triangle wave, and a square wave. For the strain gage channels, it was necessary to run the excitation signal through a pre-attenuator to avoid railing in the in-amp. These tests were performed with a gain of 200 V/V on the strain gage signal path and a gain of 50 V/V on the accelerometer signal path. No attenuation was required for the accelerometer signal chain.

5.1.1 Sinusoidal Input

A 1 kHz sinusoid was input to the four strain gage channels to obtain a $2 V_{pp}$ signal at the output of the signal chain. The signal was then converted to the digital domain using the Analog Discovery and this was plotted in MATLAB. The results are shown in Figure 28.

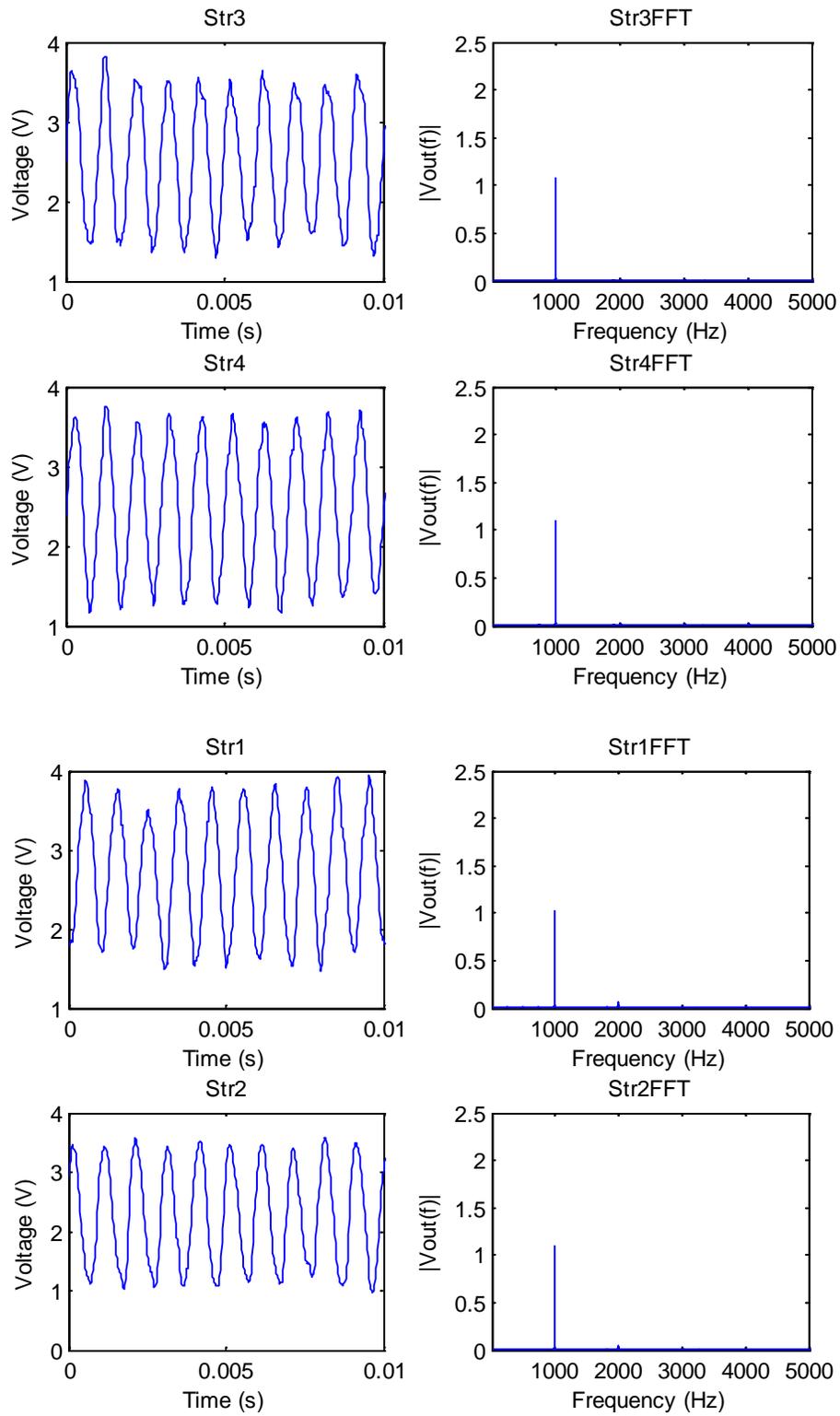


Figure 28: Output Resulting from 1 kHz Sinusoid Input to Strain Gage Signal Chain

It can be seen that when a pure sinusoid at 1 kHz is input to the strain gage signal chain, the output is an undistorted sinusoid in frequency yielding the expected magnitude of 1 V. Distortion, however, is visible in the time domain.

For the accelerometer chain, a 200 Hz sinusoid was input such that the same output voltage was obtained. Figure 29 shows the results of this test.

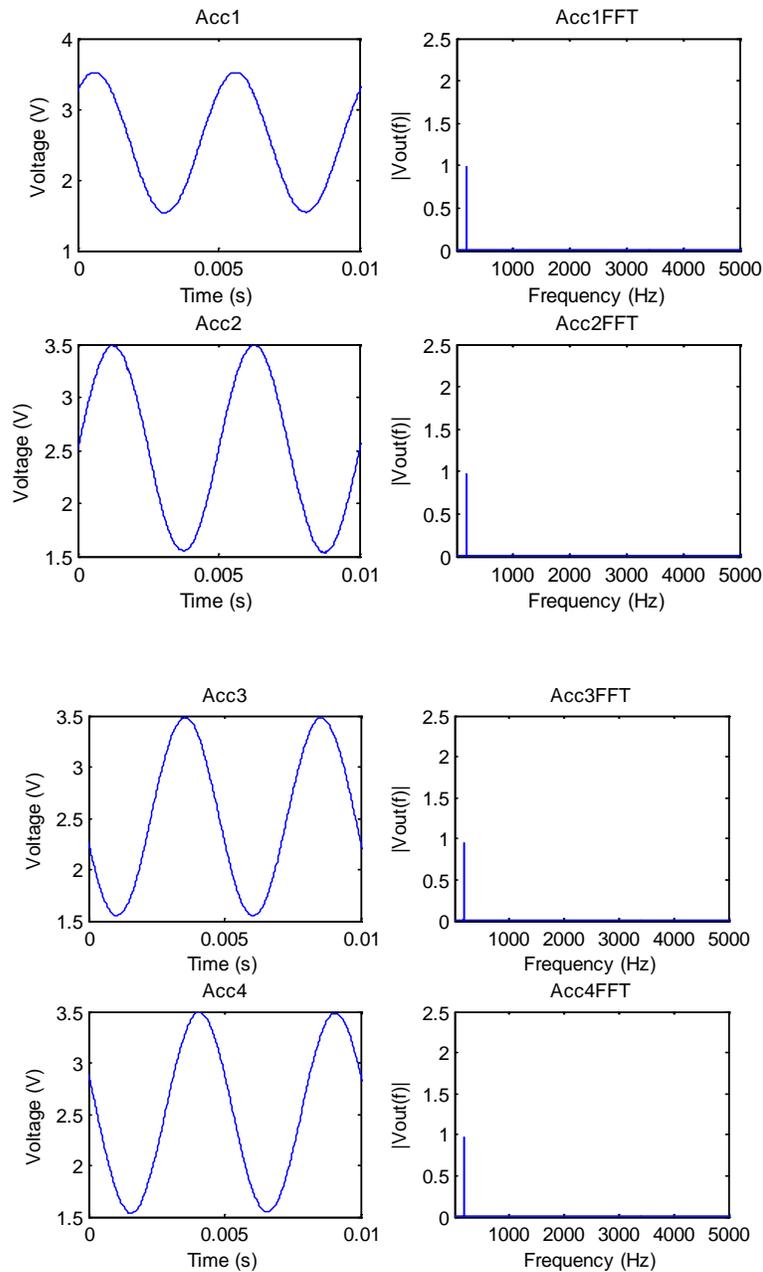


Figure 29: Output Resulting from 200 Hz Sinusoid Input to Accelerometer Signal Chain

It was observed that the accelerometer signal chain displayed undistorted behavior in both time and frequency. A pure tone was observed at 200 Hz and the time-value signal is clearly a sinusoid.

5.1.2 Triangle Wave Input

Performing the same test, but this time with a 1 kHz triangle wave, the outputs were again recorded and can be seen in Figure 30.

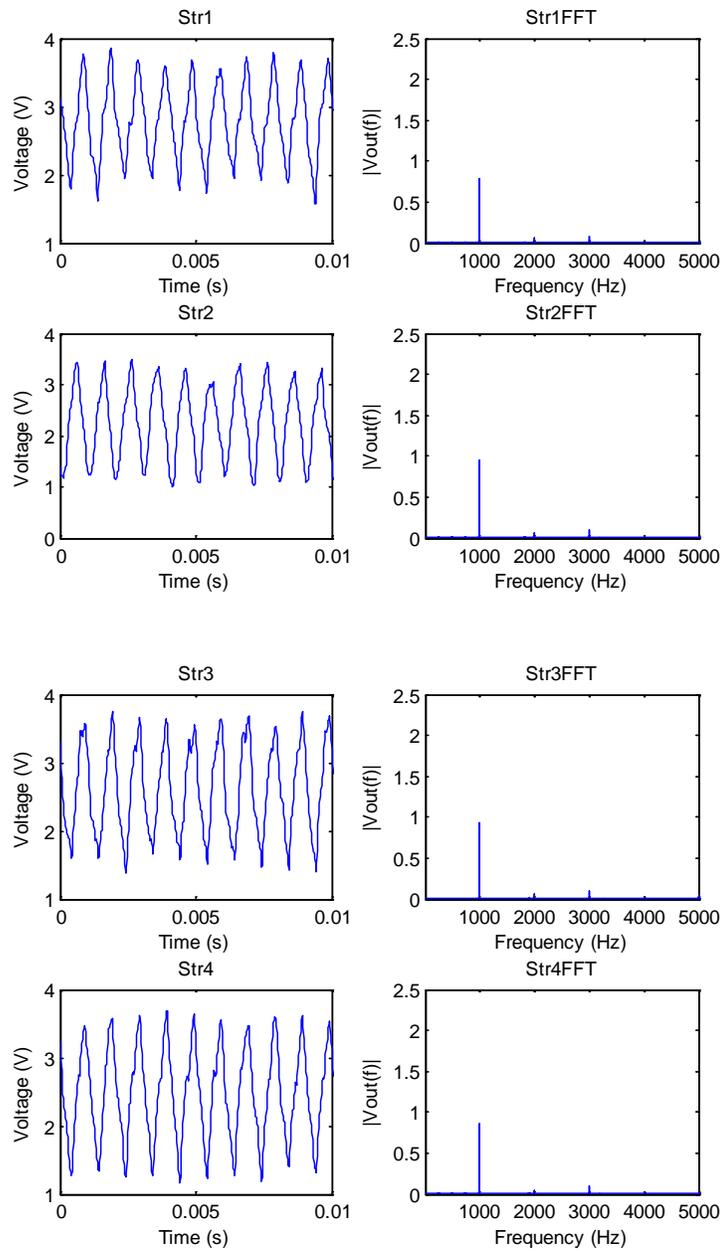


Figure 30: Output Resulting from 1 kHz Triangle Wave Input to Strain Gage Signal Chain

From Figure 30 it can be seen that the strain gage signal chain yields, again, a slightly distorted output in the time domain. An undistorted signal in the frequency domain with the fundamental at nearly 1 V and the expected harmonics were observed.

A 200 Hz triangle wave was then input to each of the accelerometer channels and the outputs from the system were recorded and displayed in Figure 31.

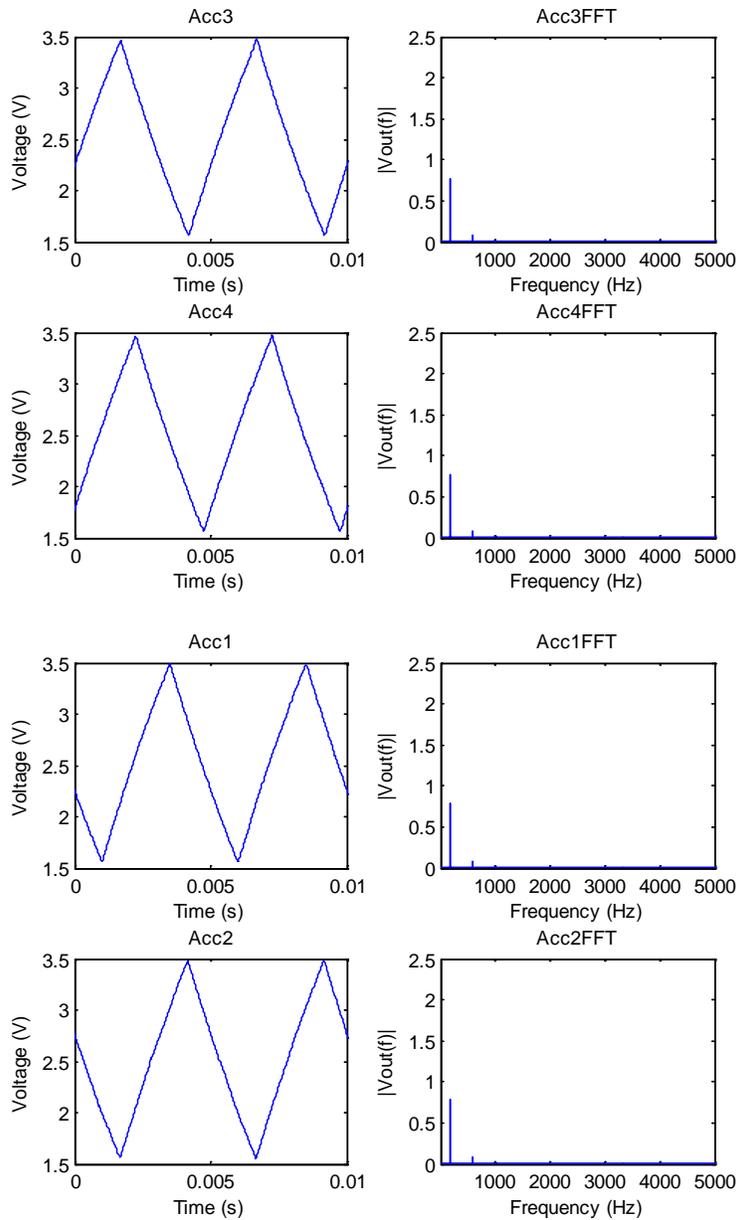


Figure 31: Output Resulting from 200 Hz Triangle Wave Input to Accelerometer Signal Chain

The results of this test indicate an undistorted output for the accelerometer signal chain in the frequency with the expected harmonics present for a triangle wave. In the time domain, a slight capacitive effect was observed stemming from the use of a high-pass filter in the accelerometer chain post-amp.

5.1.3 Square Wave Input

Repeating the same test procedure for a square wave input the following results were obtained:

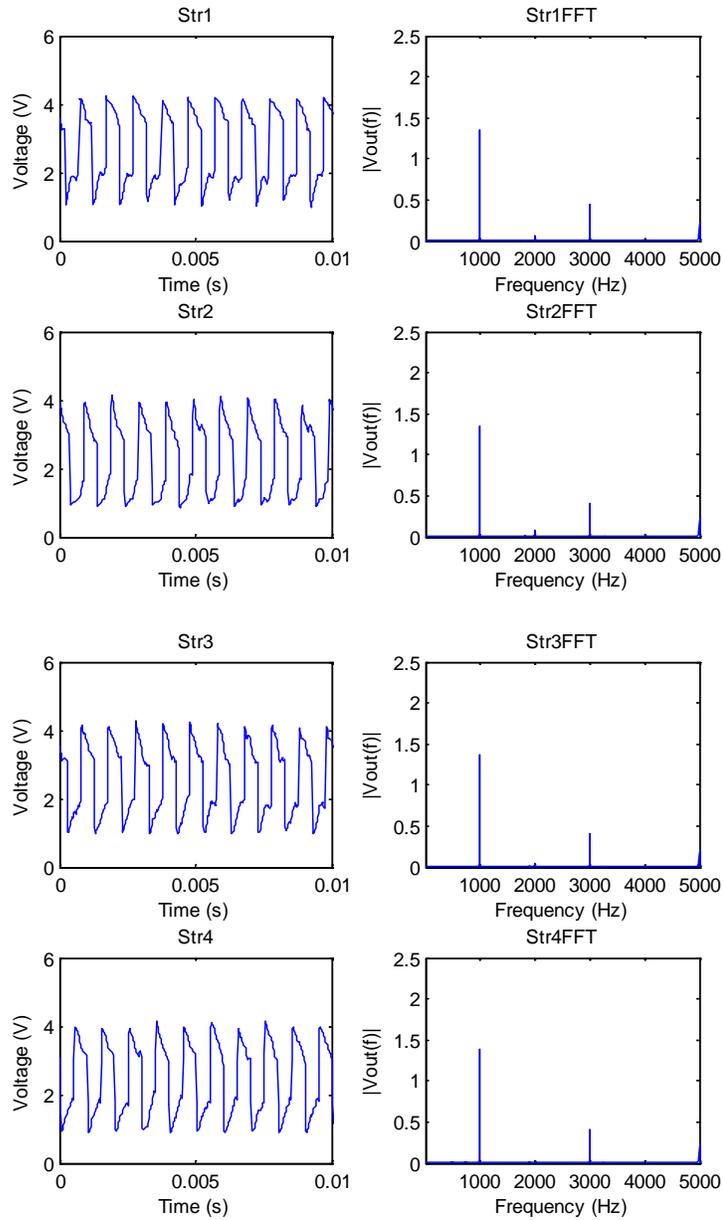


Figure 32: Output Resulting from 1 kHz Square Wave Input to Strain Gage Signal Chain

A pronounced capacitive effect was observed in the time domain in Figure 32. The signal otherwise behaved as expected.

Evaluating the accelerometer signal chain:

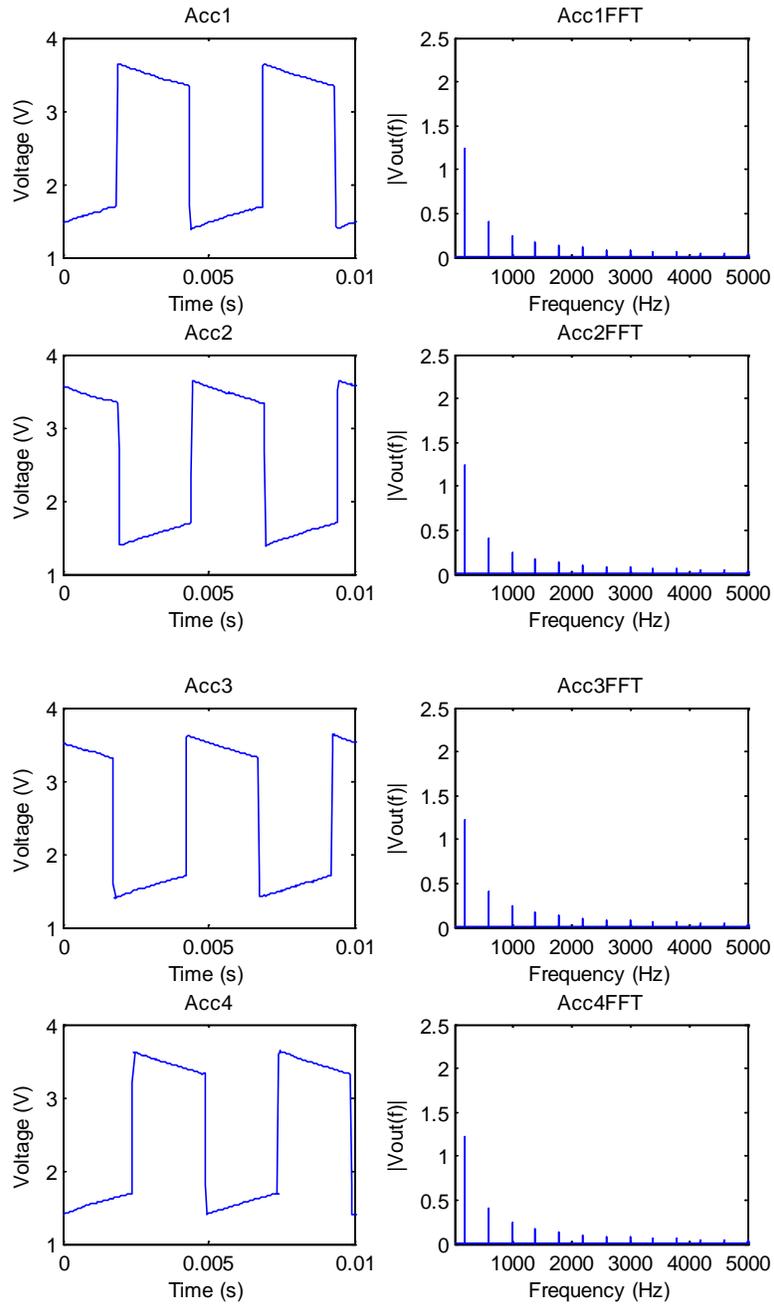


Figure 33: Output Resulting from 200 Hz Square Wave Input to Accelerometer Signal Chain

Figure 33 again shows typical square wave behavior with the exception of the capacitive effects introduced by the high-pass filter present in the accelerometer signal chain post-amp.

5.2 Full System Test

After verifying proper operation of the signal channels for several well-known waveforms a test of the full system was conducted. Initially, the mechanical test system was run with the mechanical excitation applied while all four screws were firmly staked down. The resultant waveforms can be seen in Figure 34 and Figure 35.

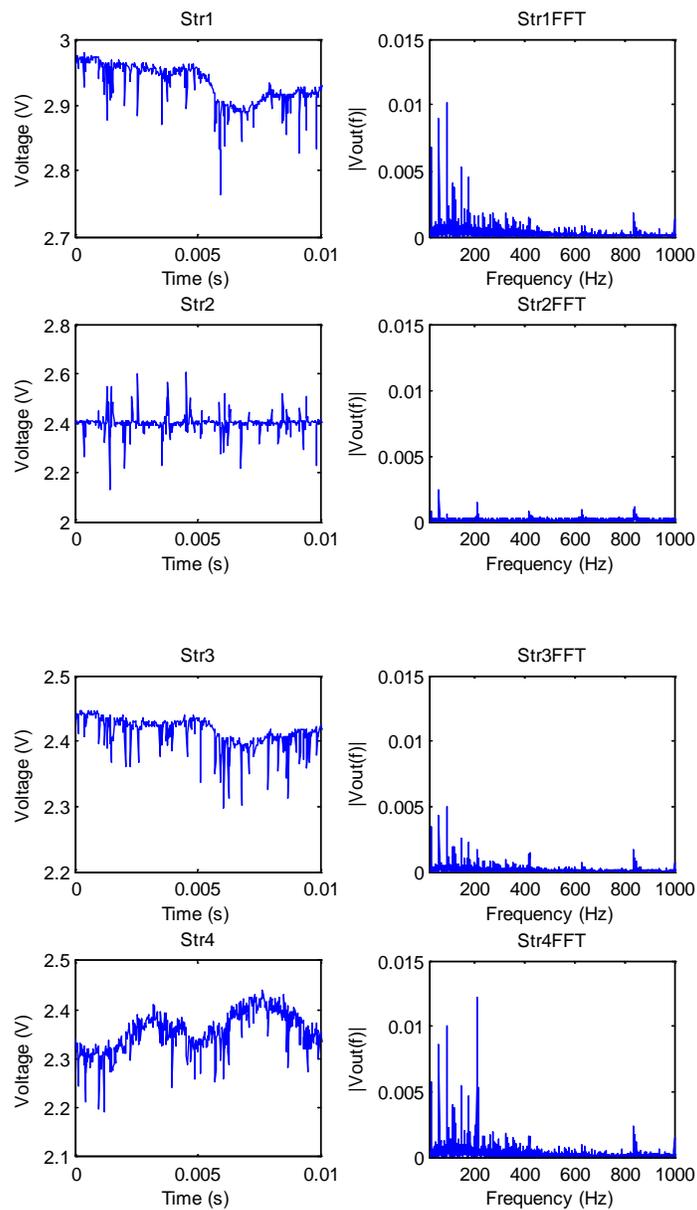


Figure 34: Strain Gage Signals with All Four Screws in Place

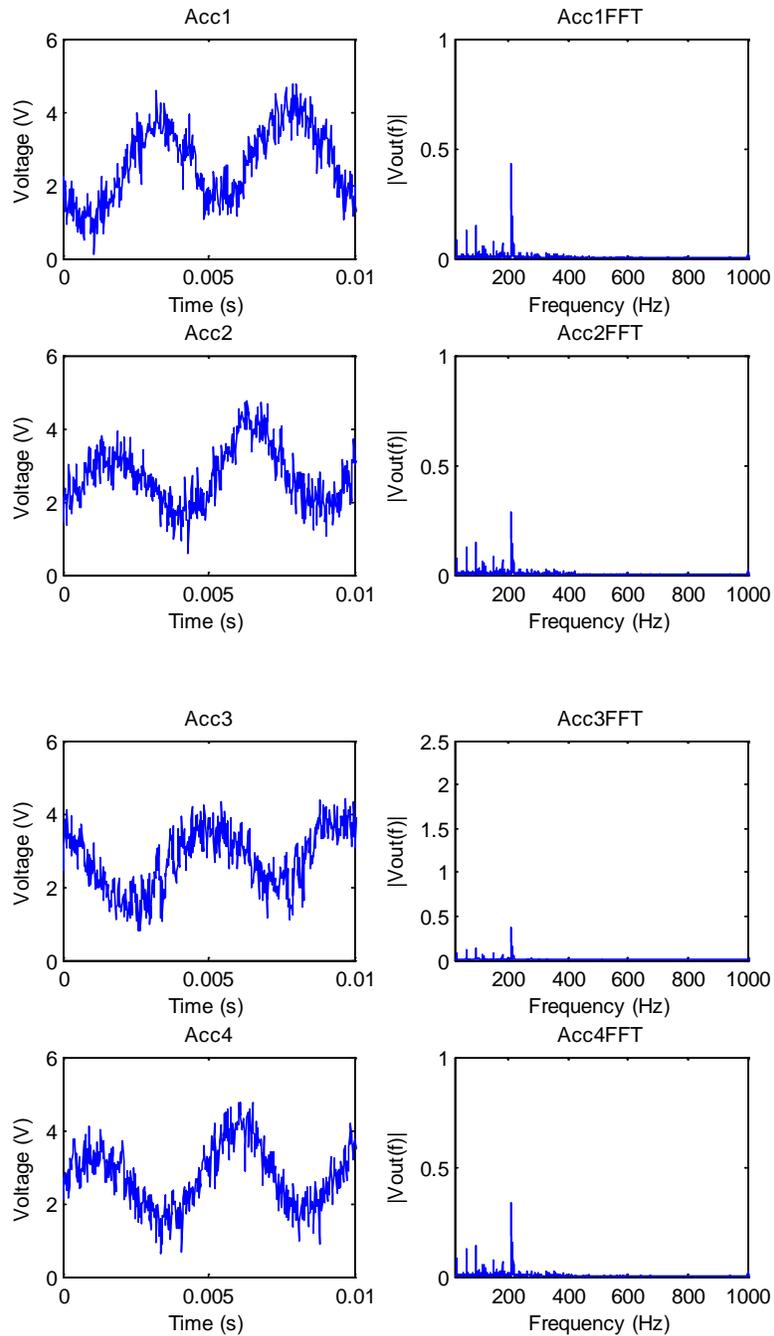


Figure 35: Accelerometer Signals with All Four Screws in Place

The source of the mechanical excitation, rated at approximately 200 Hz, was visible in Figure 35 in both the time and frequency domains. The strain gage signals, however are not as clear, and it can be seen that strain gage channel 4 is behaving as an accelerometer. The source of this behavior has been investigated, but was not discovered. It can be seen that the strain gage detects different modes of vibration that were not visible to the accelerometer.

After testing the system with all of the screws tightly in place, screw 2 was loosened and measurements were again made. Figure 36 shows the resultant strain gage signals.

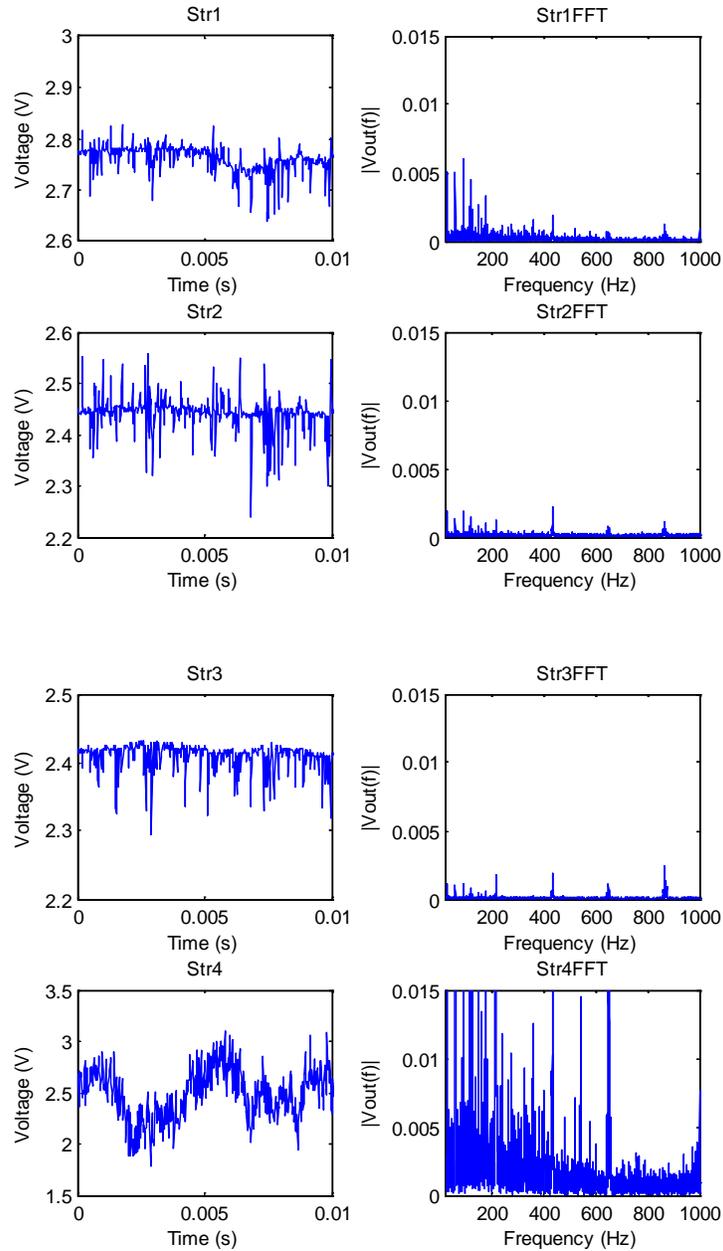


Figure 36: Strain Gage Signals with Screw 2 Loosened

For this experiment, the screw that is in closest proximity to strain gage 2 is loosened while the other three strain gages remain staked down. Due to the nature of the strain, the largest displacement will be seen on the strain gage opposite of the one loosened. This is the result of the lever arm for strain gage 4 being longer and this can be seen from the larger amplitude of the signal seen in the plot in

Figure 36. Despite the larger relative amplitude, the frequency content is severely distorted compared to the strain gages that were staked. The exact cause of this is not known but it is assumed that it results from the plate striking the stand-off of the mechanical system.

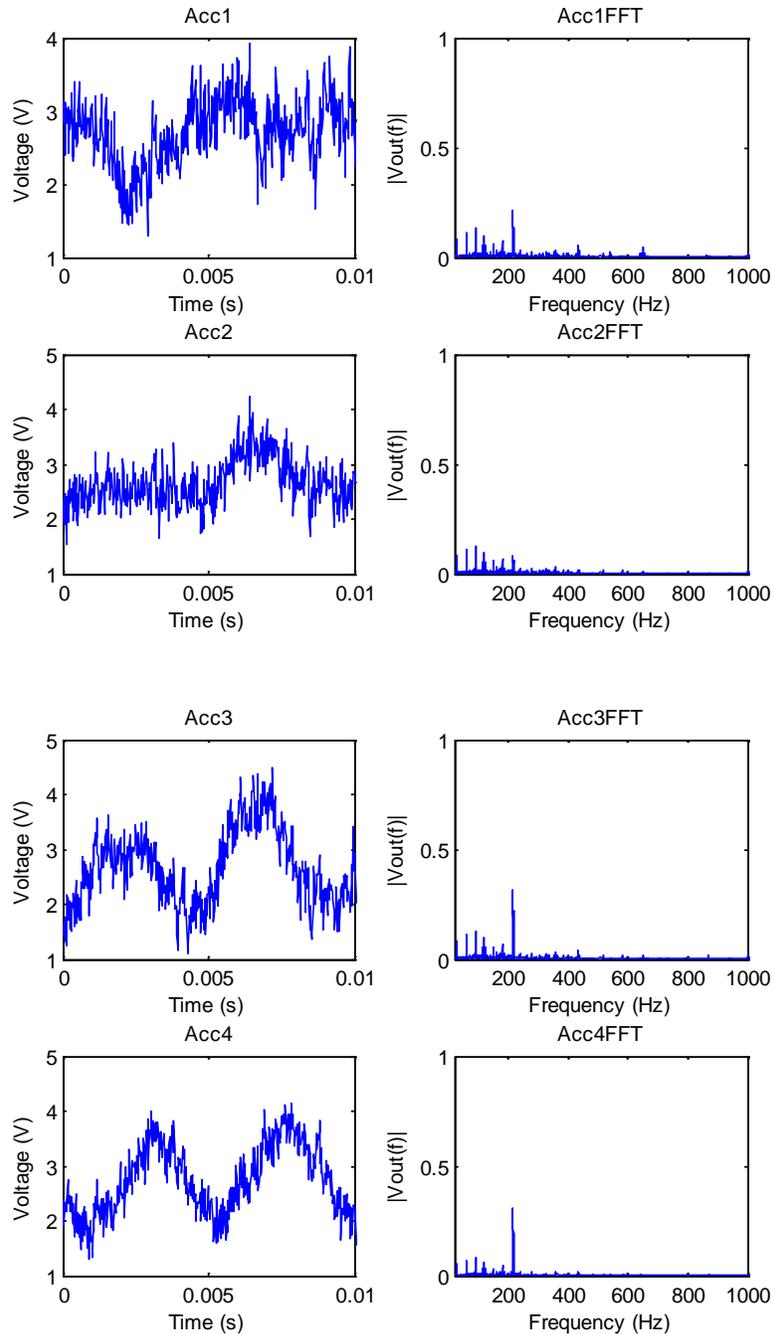


Figure 37: Accelerometer Signals with Screw 2 Loosened

In Figure 37 it was observed that the accelerometer closest to screw 2, accelerometer 2, output a lower magnitude signal than the other accelerometers. This is due to the fact that when a screw is

loose the plate in that region travels over a larger distance before changing direction again. Consequently, the acceleration in that region will be lower. Accelerometers 1, 3, and 4 were fairly close in magnitude.

6 Future Work and Recommendations

Over the course of the project several issues were encountered. These issues and possible solutions to them will be addressed in the following sections. If this project were to be continued it may be of interest to explore implementing these options in order to increase system performance.

6.1 Data Acquisition

The main problem encountered in the design was with data acquisition. Initially we planned to make use of a microprocessor and its integrated A/D converter. It was originally thought that while sampling at 400kHz, data transfer could be done with USB. Two key issues arose from the implementation. The first was that in order to transfer data at this rate, external memory would be required. In addition, direct memory access (DMA) would be required. The second issue was that there was a lack of embedded system expertise on the design team. A microprocessor would have provided a more cost-effective and flexible data acquisition platform than the Analog Discovery in every way.

It was found that deficiencies in the strain gage chain originated in the data acquisition block. The limitations of the Analog Discovery imposed undesired constraints on the system. Primarily, it was found that the Analog Discovery forces the user to implement filtering of collected data. This filtering distorts signal content that cannot be remedied due to the fact the Analog Discovery provides severely limited documentation.

Future changes to the project would ideally begin with selecting one of ADI's data acquisition-oriented microprocessors, such as the ADuC7023. This product offers an eight channel 12-bit A/D converter with a 1MSPS throughput.

6.2 Offset Adjustment

One of the secondary problems encountered with the project was with the offset adjustment scheme. In the current implementation, there is no way of dealing with a negative bridge offset because a single-supply reference D/A converter is being used. Additionally, even with a dual-supply reference D/A converter, the reference on the device would need to be increased from 3.3V to at least 5V.

A more straightforward solution would be to slightly modify the Wheatstone bridges such that, in the strain sensing leg, a fixed 950 Ω resistor in series with a 100 Ω potentiometer is used instead of a single fixed 1k Ω resistor. This would require manual calibration of the system, but would

guarantee that the system could be balanced regardless of resistor tolerances or inherent bending in the strain gages.

6.3 Power Requirements

The last area which presented a problem over the course of the project was the issue of power requirements. Initially, it was believed that a 100mA current draw limitation would be imposed if power over USB was used. This was due to the fact that the USB 2.0 standard supposedly requires device interrogation and a USB high-power request if more than 100mA is desired. It was later found that this is rarely, if ever, enforced. Many USB devices, including mobile phone chargers, routinely draw approximately 500mA without requesting USB high-power. It is assumed that whatever device is supplying power to the USB hub will be capable of delivering the desired current.

7 Conclusion

The original project goal was to design and implement an analog front end for vibration monitoring and analysis in order to detect a loose screw in a mechanical system. Ultimately, this goal was not entirely achieved. However, after testing the analog front end it was found to operate in a satisfactory manner. The system allowed for the proper analog signal conditioning to be performed for the accelerometers, but some deficiencies were found to exist when the strain gage signal chain was tested.

System signal conditioning requirements were successfully determined by considering sensor and data acquisition unit specifications. This allowed the gap between these two systems to be bridged. Detection of a loose screw was successfully done using accelerometers, but not successfully done with strain gages. The AD8421 met performance expectations. It was able to simultaneously provide high gain while operating over a 20kHz bandwidth in addition to furnishing the system with superior CMRR and minimal noise contribution. However, due to the limitations imposed by a single-sided supply from USB the lack of rail-to-rail operation was found to be an issue.

Overall, despite several setbacks the project allowed the team to explore various analog design techniques while also providing a strongly multi-disciplinary challenge. As a result the project was an excellent learning experience and introduced the team to topics such as single-sided supply system design.

8 Acknowledgements

We would like to thank the following people for their contributions to the project:

Professor Stephen Bitar for his helpful insight and generous lending of components.

Xu Chen for his contributions including giving us food and driving us places.

Richard Dennen for sharing his time, expertise, equipment, and humor.

Joseph William D. DiChiara for his useful suggestions.

Jordyn Rombola for providing us with motors and moral support.

Salik Siddiqui for his witticisms and kind contributions.

Dr. Michael Coln (ADI) for his support, life-lessons, and encouragement.

Max Liberman (ADI) for introducing us to the ADX001 accelerometer which were an integral part of this MQP.

Oljeta Qirko (ADI) for helping us with project ideas and providing us with the parts we needed from the ADI catalog.

Finally, we would like to thank Professor John A. McNeill—our advisor. As a result of his careful guidance, analog knowledge, and constant trolling we have become better engineers.

9 References

- [1] http://www.analog.com/static/imported-files/data_sheets/AD8420.pdf
- [2] http://www.analog.com/static/imported-files/data_sheets/AD8237.pdf
- [3] http://www.analog.com/static/imported-files/data_sheets/AD8421.pdf
- [4] http://www.analog.com/static/imported-files/data_sheets/AD8283.pdf
- [5] http://www.analog.com/static/imported-files/data_sheets/ADF4159.pdf
- [6] Steve Goldman *Vibration Spectrum Analysis: A Practical Approach*.
- [7] http://www.efunda.com/designstandards/sensors/strain_gages/strain_gage_theory.cfm
- [8] http://www.efunda.com/designstandards/sensors/strain_gages/strain_gage_sensitivity.cfm
- [9] http://www.efunda.com/designstandards/sensors/strain_gages/strain_gage_selection_matl.cfm
- [10] <http://upload.wikimedia.org/wikipedia/commons/c/c3/StrainGaugeVisualization.svg>
- [11] http://www.play-hookey.com/dc_theory/resistors/images/wheatstone_bridge.gif
- [12] Wilson, Earl J. *Harris' Shock and Vibration Handbook*, pp. 17.1 – 17.15
- [13] <http://uk.rs-online.com/web/p/strain-gauges/0632180/>
- [14] <http://www.digikey.com/product-detail/en/CEA-06-250UW-350/1033-1013-ND/2503704>
- [15] http://www.analog.com/static/imported-files/data_sheets/AD8421.pdf
- [16] Girdhar, Paresh Practical Machinery Vibration Analysis and Predictive Maintenance. pp. 32
- [17] <http://www.omega.com/pptst/ACC103.html>
- [18] Wilson, Earl J. *Harris' Shock and Vibration Handbook*, pp. 17.1 – 17.15
- [19] http://www.new.mmf.de/characteristics.htm#frequ_resp
- [20] <http://www.analog.com/adxl001>
- [21] <http://www.analog.com/adxl001>
- [22] <http://www.analog.com/ad8421>
- [23] Figures and numerical values adapted from AD8421 Datasheet.
- [24] Kester W., *Taking the Mystery out of the Infamous Formula*, “ $SNR = 6.02N + 1.76dB$,” and *Why You should Care*, MT-001 Tutorial, Analog Devices.
- [25] <http://www.analog.com/en/digital-to-analog-converters/da-converters/ad5684r/products/product.html>


```

        end
    end
end

%for first MSB transition from high to low, we get 000 -> channel 0 use
%this as a starting point for channel indexing
for j = 2:r
    if((msbtick(j-1,1) == 1) && (msbtick(j,1) == 0))
        index = j;
        break;
    end
    % if((msbtick(j-1,1) == 0) && (msbtick(j,1) == 1))
    %     index = j;
    %     break;
    % end
end

%indexing off of that initial transition, the LSB ticks once every sample
    %MSB: ____|----|____|----|____
    %MSB: 0000 1111 0000 1111 0000
    %MID: 0011 0011 0011 0011 0011
    %LSB: 0101 0101 0101 0101 0101

%meaning that we are just incrementing the channel number by one for every
%sample after the index sample (the first 1->0 MSB transition

for k = 1:r
    if k <= index
        channum(k,1) = 0;
    elseif k-1 <= index
        channum(k,1) = channum(k-1,1) + 1;
    else
        channum(k,1) = 0;
    end
end

end

% append and output the new matrix with channel numbers in column 4
m = [rawdat channum];
end

% ----- %
function [ C ] = deconvolved_sig(filename, numchannels)
% James Filice, Naveed Naeem, Kaung Myat Win
% ADI Chip MQP Team
% 2/5/2012

%Takes in the filename of a .csv and deconvolves it into individual data
%vectors representing each channel(sensor)

fclose('all');
Chans = chanstamp(filename); %Takes the out of the chanstamp to sort
                             % .csv files into a matrix which assigns
                             % a number to the corresponding sensor
                             % channel

vec_length = length(Chans(:,1));%finds the length of the columns in 'Chans'
C = zeros((vec_length/numchannels), numchannels); %pre-allocates the matrix

for j = 1:numchannels-1
    ChanIDX = find(Chans(1:end,4)==j);
    if isequal(j,1)
        Chan0IDX = ChanIDX-1;
        if Chan0IDX(1)>8
            added0IDX = Chan0IDX(1)-8;
            if added0IDX>8
                added0IDX = [added0IDX-8;added0IDX]; %#ok<AGROW>
            end
            Chan0IDX = [added0IDX;Chan0IDX]; %#ok<AGROW>
        end
    end
    if isequal(Chan0IDX(end),vec_length-numchannels)
        Chan0IDX = [Chan0IDX;vec_length]; %#ok<AGROW>
    end
end

```

```

        end
    end
    if ChanIDX(1)>8
        addedIDX = ChanIDX(1)-8;
        if addedIDX>8
            addedIDX = [addedIDX-8;addedIDX]; %#ok<AGROW>
        end
        ChanIDX = [addedIDX;ChanIDX]; %#ok<AGROW>
    end
    C(:,j+1) = Chans(ChanIDX,2);
end
C(:,1) = Chans(Chan0IDX,2);

% ----- %

% James Filice, Naveed Naeem, Kaung Myat Win
% ADI Chip MQP Team

% Plots the time domain signal given in data, the nth channel and the
% sampling frequency

function plotChannel(data,ChannelNum,freq)

dataCh = data(:,ChannelNum);
numelCh = numel(dataCh);
time = 0:1/freq:(numelCh-1)*1/freq;
xlim([0 0.1]);
plot(time,dataCh);
title('Measured Voltage Vout')
xlabel('Time (s)')
ylabel('Voltage (V)')

% ----- %

% James Filice, Naveed Naeem, Kaung Myat Win
% ADI Chip MQP Team

% Plots the frequency domain signal given in data, the nth channel and the
% sampling frequency

function plotFFT(dataCh,ChannelNum,Fs)

Ts = 1/Fs;
data = dataCh(:,ChannelNum);
N=length(data); % length of the signal x
ssf=(ceil(-N/2):ceil(N/2)-1)/(Ts*N); % frequency vector
fx=fft(data(1:N)); % do DFT/FFT
fxs=fftshift(fx);
plot(ssf,2*abs(fxs)/N);
xlim([20 1000]);
ylim([0 0.3]); % can be done with axis([3 250 0 0.03])
title('Single-Sided Amplitude Spectrum of Vout(t)')
xlabel('Frequency (Hz)')
ylabel('|Vout(f)|')

% ----- %

% James Filice, Naveed Naeem, Kaung Myat Win
% ADI Chip MQP Team

% Example script to perform time domain and FFT analyses on the acquired .csv files

Fs = 400e3/8; %Each channel is sampled at 50kHz each
FsFFT = Fs;

MatOut = deconvolved_sig('scope_recordb4.csv',8);
figure(1)
subplot(2,2,1);plotChannel(MatOut,1,Fs); title('Str1');xlim([0 0.01]);
subplot(2,2,2);plotFFT(MatOut,1,FsFFT); title('Str1FFT');ylim([0 0.015]);

```

```

subplot(2,2,3);plotChannel(MatOut,2,Fs); title('Str2');xlim([0 0.01]);
subplot(2,2,4);plotFFT(MatOut,2,FsFFT); title('Str2FFT');ylim([0 0.015]);
figure(2)
subplot(2,2,1);plotChannel(MatOut,3,Fs); title('Str3');xlim([0 0.01]);
subplot(2,2,2);plotFFT(MatOut,3,FsFFT); title('Str3FFT');ylim([0 0.015]);
subplot(2,2,3);plotChannel(MatOut,4,Fs); title('Str4');xlim([0 0.01]);
subplot(2,2,4);plotFFT(MatOut,4,FsFFT); title('Str4FFT');ylim([0 0.015]);

figure(3)
subplot(2,2,1);plotChannel(MatOut,5,Fs); title('Acc1');xlim([0 0.01]);
subplot(2,2,2);plotFFT(MatOut,5,FsFFT); title('Acc1FFT');ylim([0 1]);
subplot(2,2,3);plotChannel(MatOut,6,Fs); title('Acc2');xlim([0 0.01]);
subplot(2,2,4);plotFFT(MatOut,6,FsFFT); title('Acc2FFT');ylim([0 1]);
figure(4)
subplot(2,2,1);plotChannel(MatOut,7,Fs); title('Acc3');xlim([0 0.01]);
subplot(2,2,2);plotFFT(MatOut,7,FsFFT); title('Acc3FFT');ylim([0 2.5]);
subplot(2,2,3);plotChannel(MatOut,8,Fs); title('Acc4');xlim([0 0.01]);
subplot(2,2,4);plotFFT(MatOut,8,FsFFT); title('Acc4FFT');ylim([0 1]);

MatOut = deconvolved_sig('scope_recordafter.csv',8);
figure(5)
subplot(2,2,1);plotChannel(MatOut,1,Fs); title('Str1');xlim([0 0.01]);
subplot(2,2,2);plotFFT(MatOut,1,FsFFT); title('Str1FFT');ylim([0 0.015]);
subplot(2,2,3);plotChannel(MatOut,2,Fs); title('Str2');xlim([0 0.01]);
subplot(2,2,4);plotFFT(MatOut,2,FsFFT); title('Str2FFT');ylim([0 0.015]);
figure(6)
subplot(2,2,1);plotChannel(MatOut,3,Fs); title('Str3');xlim([0 0.01]);
subplot(2,2,2);plotFFT(MatOut,3,FsFFT); title('Str3FFT');ylim([0 0.015]);
subplot(2,2,3);plotChannel(MatOut,4,Fs); title('Str4');xlim([0 0.01]);
subplot(2,2,4);plotFFT(MatOut,4,FsFFT); title('Str4FFT');ylim([0 0.015]);

figure(7)
subplot(2,2,1);plotChannel(MatOut,5,Fs); title('Acc1');xlim([0 0.01]);
subplot(2,2,2);plotFFT(MatOut,5,FsFFT); title('Acc1FFT');ylim([0 1]);
subplot(2,2,3);plotChannel(MatOut,6,Fs); title('Acc2');xlim([0 0.01]);
subplot(2,2,4);plotFFT(MatOut,6,FsFFT); title('Acc2FFT');ylim([0 1]);
figure(8)
subplot(2,2,1);plotChannel(MatOut,7,Fs); title('Acc3');xlim([0 0.01]);
subplot(2,2,2);plotFFT(MatOut,7,FsFFT); title('Acc3FFT');ylim([0 1]);
subplot(2,2,3);plotChannel(MatOut,8,Fs); title('Acc4');xlim([0 0.01]);
subplot(2,2,4);plotFFT(MatOut,8,FsFFT); title('Acc4FFT');ylim([0 1]);

% ----- %

```

The following code snippet show the noise analysis performed in Section 4.4:

```

% Kaung Myat Win, James Filice, Naveed Naeem
% MQP Noise Analysis
% 11/18/2012
% Updated 4/22/2013

% ----- Only the noise from the bridge ----- %
% Constants
k = 1.38e-23; % Boltzmann's Constant [J/K]
q = 1.6e-19; % charge of an electron [coul]
T = 300; % Temperature [K]

% Input Variables
Vcc = 5; % 5V
Num_Bridges = 4; % Number of bridges
R1 = logspace(1,8,1000); % 10^1 Hz to 10^8
Rstr = 1000; % 1000-Ohm Strain Gage
f3dB = 20E3; % 20kHz bandwidth

```

```

R3 = Rstr; % Matched Resistance

G1 = 200; % First Stage Gain
G2 = 2; % Second Stage Gain
GT = G1*G2; % Total Gain

en_R1 = sqrt(4*k*T*R1)/2; % Voltage Noise of R1
en_Rstr = sqrt(4*k*T*Rstr/2); % Voltage Noise of Strain Gage

% Output
Vrms_R1 = en_R1*GT*sqrt(pi/2*f3dB);
Vrms_Rstr = en_Rstr*GT*sqrt(pi/2*f3dB);

Rthev = 2.*R1.*2.*Rstr./(2.*R1+2.*Rstr);
Current_T = Num_Bridges.*Vcc./(Rthev);
Power_T = Vcc*Current_T;

% ----- Include the op-amp input referred noises ----- %
en_A = 3.5e-9; % Amplifier input voltage noise
in_A = 200e-15; % Amplifier input current noise

Vrms_en = en_A*GT*sqrt(pi/2*f3dB);
Vrms_in = in_A*Rthev*GT*sqrt(pi/2*f3dB);

Vrms_AMP_T = sqrt((Vrms_R1).^2+(Vrms_R1).^2+...
                 (Vrms_Rstr).^2+(Vrms_Rstr).^2+...
                 (Vrms_en).^2+(Vrms_in).^2);

In_Ref_Noise = Vrms_AMP_T/(G1*G2); % Input referred noise

FOM = 1./(In_Ref_Noise.*Current_T); % Figure of merit

figure(1); loglog(R1,In_Ref_Noise); grid on;
title('Total Input referred noise vs. R1 resistance')
xlabel('R1 Resistance (\Omega)')
ylabel('Vrms due to differnet R1 values (V)')

figure(2); loglog(R1,Vrms_AMP_T); grid on;
title('Total Output Vrms vs. R1 resistance')
xlabel('R1 Resistance (\Omega)')
ylabel('Total Vrms at output (V)')

figure(3); loglog(R1,Current_T); grid on;
title('Total current drawn from supply vs. R1 resistance')
xlabel('R1 Resistance (\Omega)')
ylabel('Total current drawn from the source (A)')

figure(4); loglog(R1,FOM); grid on;
title('Figure of Merit for differnet R1 resistor values')
xlabel('R1 Resistance (\Omega)')
ylabel('1/(Noise*Current) (McNeillians)')

% ----- Include ADC noise ----- %

numBits = 14; % 14-bit ADC in Analog Discovery
FSR = 1.8; % FSR of the ADC in Analog Discovery
LSB = FSR/(2^numBits-1);
Vrms_ADC = LSB/sqrt(12);
Out_Noise_Density = Vrms_AMP_T/sqrt(pi/2*f3dB);

figure(5); loglog(R1,Out_Noise_Density); grid on;
title('Output noise density')
xlabel('R1 Resistance (\Omega)')
ylabel('Noise density at output(V/rtHz)')
% ----- %

```

Appendix B – Microcontroller Code

```
// Authors: James Filice, Naveed Naeem, Kaung Myat Win,
// With help and contribution of Richard Dennen and Salik Siddiqui
// Date: 3/21/2013
// Edition: 4/21/2013

#include <msp430.h>

// constants 0xYZ; Y=operation, Z=DACnumber
#define FirstDAC 0x31 // write to first DAC
#define SecondDAC 0x32 // write to second DAC
#define ThirdDAC 0x34 // write to third DAC
#define FourthDAC 0x38 // write to fourth DAC

// function prototypes
void SetDACVal(unsigned char DAC_num, unsigned char MSB_Val, unsigned char LSB_Val);
unsigned char* ReadADC(unsigned int ADC_Chan);

// main function
int main(void)
{
    WDTCTL = WDTPW + WDTHOLD; // Stop watchdog timer

    BCSCTL1 = CALBC1_1MHZ; // Set DCO for 1MHz
    DCOCTL = CALDCO_1MHZ;

    P1SEL = BIT2 + BIT4; // P1.2=TXD, P1.4=SMCLK
    P1SEL2 = BIT2 + BIT4; // P1.2=TXD, P1.4=SMCLK
    P1DIR |= BIT2;
    P1DIR |= BIT0; // P1.0=LED

    P2DIR |= BIT0 + BIT1 + BIT2; // P2.0=LDAC, P2.1=DAC_RST, P2.2=SYNC_IN
    P2DIR &= ~BIT4 + ~BIT5;

    UCA0CTL0 |= UCMSB + UCMST + UCSYNC; // 3-pin, 8-bit SPI master , inactive state = low
    UCA0CTL1 |= UCSSEL_2; // SMCLK
    UCA0BR0 |= 0x04; // divide by 4
    UCA0BR1 = 0; //
    UCA0MCTL = 0; // No modulation
    UCA0CTL1 &= ~UCSWRST; // **Initialize USCI state machine**

    __delay_cycles(100);
    // DAC reset
    P2OUT &= ~BIT1; // SET P2.1=--RESET Low
    __delay_cycles(10);
    P2OUT |= BIT1; // SET P2.1=--RESET High

    unsigned char* bytePtr;
    unsigned char newRefhighByte,newReflowByte;

    __delay_cycles(10000);

    // First DAC
    SetDACVal(FirstDAC, 0xC1, 0xF0);
    __delay_cycles(10000); // 1 delay cycle = 0.88us
    bytePtr = ReadADC(INCH_7);
    newRefhighByte = *bytePtr;
    bytePtr = bytePtr+1;
    newReflowByte = *bytePtr;

    SetDACVal(FirstDAC, newRefhighByte, newReflowByte); /******/
    //SetDACVal(FirstDAC, 0xFF, 0xF0);
    //SetDACVal(FirstDAC, 0xC1, 0xF0);
    __delay_cycles(10000); // 1 delay cycle = 0.88us
}
```

```

// Second DAC
SetDACVal(SecondDAC, 0xC1, 0xF0);
__delay_cycles(10000); // 1 delay cycle = 0.88us
bytePtr = ReadADC(INCH_6);
newRefhighByte = *bytePtr;
bytePtr = bytePtr+1;
newReflowByte = *bytePtr;

SetDACVal(SecondDAC, newRefhighByte, newReflowByte); /*****/
//SetDACVal(SecondDAC, 0xD4, 0x80);
//SetDACVal(SecondDAC, 0xC1, 0xF0);
__delay_cycles(10000); // 1 delay cycle = 0.88us

// Third DAC
SetDACVal(ThirdDAC, 0xC1, 0xF0);
__delay_cycles(10000); // 1 delay cycle = 0.88us
bytePtr = ReadADC(INCH_5);
newRefhighByte = *bytePtr;
bytePtr = bytePtr+1;
newReflowByte = *bytePtr;

SetDACVal(ThirdDAC, newRefhighByte, newReflowByte); /*****/
//SetDACVal(ThirdDAC, 0x99, 0xA0);
//SetDACVal(ThirdDAC, 0xC1, 0xF0);
__delay_cycles(10000); // 1 delay cycle = 0.88us

// Fourth DAC
SetDACVal(FourthDAC, 0xC1, 0xF0);
__delay_cycles(10000); // 1 delay cycle = 0.88us
bytePtr = ReadADC(INCH_3);
newRefhighByte = *bytePtr;
bytePtr = bytePtr+1;
newReflowByte = *bytePtr;

SetDACVal(FourthDAC, newRefhighByte, newReflowByte); /*****/
//SetDACVal(FourthDAC, 0xFF, 0xF0);
//SetDACVal(FourthDAC, 0xC1, 0xF0);
__delay_cycles(10000); // 1 delay cycle = 0.88us

// DAC calibration Finished
P1OUT |= BIT0; // if done, light the red LED
__bis_SR_register(LPM4_bits + GIE); // CPU off, enable interrupts
}

void SetDACVal(unsigned char DAC_num, unsigned char MSB_Val, unsigned char LSB_Val){
    unsigned char TX_Data;
    // LDAC disable
    P2OUT |= BIT0; // !LDAC high

    // Chip select enable (must reset it after LDAC pin has been disabled (Logic High))
    P2OUT |= BIT2; // !SYNC high to flush out everything
    P2OUT &= ~BIT2; // !SYNC low
    //P2OUT &= ~BIT1; // !RESET low

    // write to transmit buffer
    TX_Data = DAC_num; // Initialize data values 0x11 = write to DAC N
    UCA0TXBUF = TX_Data; // Transmit first character
    while (!(IFG2&UCA0TXIFG)); // wait until the transmission is complete

    //P2OUT |= BIT1; // !RESET high

    // 0xB26x = 2.3V
    TX_Data = MSB_Val; // MSB val
    UCA0TXBUF = TX_Data; // Transmit
    while (!(IFG2&UCA0TXIFG)); // wait until the transmission is complete
}

```

```

TX_Data = LSB_Val; // LSB val
UCA0TXBUF = TX_Data; // Transmit
while (!(IFG2&UCA0TXIFG)); // wait until the transmission is complete

// make !LDAC synchronized (brute force method)
__delay_cycles(30); // 1 delay cycle = 0.88us

// SYNC_IN needs to be high to complete writing
P2OUT |= BIT2; // bring !SYNC_IN high to complete writing

// write to output
P2OUT &= ~BIT0; // bring !LDAC low
__delay_cycles(10);
P2OUT |= BIT0; // bring !LDAC high
}

unsigned char* ReadADC(unsigned int ADC_Chan){
    unsigned char newByteArr[2] = {0x00,0x00};
    //unsigned char RefhighByte, ReflowByte;
    unsigned char* arrPtr;
    //unsigned int FirstADCVal, Voffset, RefByte; // holds 1st ADC result
    unsigned int FirstADCVal,RefByte;
    // read the offset voltage
    ADC10CTL1 = ADC_Chan;
    // VR+=Vcc & VR-=Vss, ADC10ON, interrupt disabled
    ADC10CTL0 = SREF_0 + ADC10SHT_3 + REFON + ADC10ON;

    ADC10CTL0 |= ENC + ADC10SC;
    while (!(ADC10IFG & ADC10CTL0)); // First conversion?
    FirstADCVal = ADC10MEM; // Read out 1st ADC value (10 bit result)
    __delay_cycles(10000); // 1 delay cycle = 0.88us

    RefByte = (0x308 - (FirstADCVal>>1))<<6; // 2.5V = 776 = 0x308
    //first step<<4 Shift necessary to interpret
    //second step<<2 10-bit ADC to 12-bit DAC
    //third step>>1 for gain of 2 AD8608

    newByteArr[0] = (unsigned char) ((RefByte >> 8) & 0xff);
    newByteArr[1] = (unsigned char) (RefByte & 0xff);
    arrPtr = &newByteArr[0];
    return arrPtr; // return the high byte and low byte as arrays
}

```

Appendix C – Board Layout and Schematics

