PORTABLE WIDEBAND MICROSEISMIC
DATA ACQUISITION SYSTEM

Thesis submitted to the
College of Graduate Studies and Research
in partial fulfillment of the
requirements for the degree of
Master of Science in the
Department of Electrical Engineering,
University of Saskatchewan

by
Jyothsna Rajan
Saskatoon, Saskatchewan, Canada
Spring, 2000

©Jyothsna Rajan, March 2000. All rights reserved.
To
my Daddy and Mummy
for their love,
inspiration and encouragement
and
my brothers
for their unwavering support
PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a Postgraduate degree from the University of Saskatchewan, it is agreed that the Libraries of this University may make it freely available for inspection. Permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professors who supervised this thesis work or, in their absence, by the Head of the Department of Electrical Engineering or the Dean of the College of Graduate Studies and Research at the University of Saskatchewan. Any copying, publication, or use of this thesis, or parts thereof, for financial gain without the written permission of the author is strictly prohibited. Proper recognition shall be given to the author and to the University of Saskatchewan in any scholarly use which may be made of any material in this thesis.

Request for permission to copy or to make any other use of material presented in this thesis in whole or in part should be addressed to:

Chairman, Dept. of Electrical Engg.
57 Campus Drive, University of Saskatchewan
Saskatoon, Saskatchewan, Canada – S7N 5A9
UNIVERSITY OF SASKATCHEWAN

PORTABLE WIDEBAND MICROSEISMIC DATA ACQUISITION SYSTEM

Submitted by: Jyothsna Rajan
Supervisor: Dr. Brian L.F. Daku

M.Sc Thesis Submitted to the
College of Graduate Studies and Research, March 2000

ABSTRACT

Microseismic monitoring has gained importance ever since a correlation was established between earthquakes and intensive mining operations at various mine sites across the world. A microseismic network is in place at the Cory, Saskatchewan, mine site to record vibrations from minor tremors and to analyse the data to locate zones of impending faults. Precise fault location demands greater accuracy and resolution of recorded data – both of which depend on transducer characteristics. Different transducers can be designed depending on the parameters (displacement, velocity or acceleration) that they sense, the sensing element used and the placement of the sensing element in the transducer. Currently, mass spring transducers that sense velocity are the most common transducers. It is anticipated that, in future, new transducers will be developed for microseismic monitoring applications. Before deploying these sensors in microseismic networks, it is important to test them and validate their performance.

The primary focus of this research was to develop a portable wideband data acquisition system that can be used to acquire data from newly developed transducers and compare the recorded waveforms with that obtained from transducers already in use.

The data acquisition system developed in this research has an antialiasing filter with a cut-off frequency at 2485Hz. The A/D converter has a default sampling 2500 samples per second. The developed wideband data acquisition system can be
assembled and carried in a case and installed at site. The system can be remotely accessed and configured and allows for easy offloading of data to a soft media for further analysis with other software packages. This system is not intended for online analysis but only to acquire data and process it into a form readily readable by other analysis packages.

The developed system was used to acquire test data using a geophone and a piezoelectric accelerometer developed by Dr. Brian Daku. The transducer output was read simultaneously from the data acquisition system and an indicating instrument. The response curves show a close match between these values over the selected frequency range, thus establishing software integrity.

This thesis describes the design procedure and features of the portable wideband microseismic data acquisition system, the experimental setup and benchmarks tests conducted and an interpretation of the results obtained.
ACKNOWLEDGMENTS

I take this opportunity to thank my supervisor, Dr. Brian Daku for his support and guidance throughout my academic work at the University of Saskatchewan. I am also grateful to Telecommunications Research Laboratories (TRLabs), Saskatoon, for showing active interest in and funding the research described herein.

I also thank Cliff, Murat and Kim at Microstar Laboratories for their help. Thanks also to Peter Kosteniuk for his help with many aspects of the software used in the data acquisition programs. I thank my friends, from here and abroad, for their constant encouragement. I also acknowledge all help extended by Doug Bitner, Vic Meyer and Dave Karaloff. I sincerely thank Jim Sadler, B. Abolhossani and Dr. A Prugger for all the willing discussions pertaining to this research. I also acknowledge the support extended by the Electrical Engineering department and Engineering Computer Centre staff.

I shall always be grateful to my parents and brothers for their inspiration and support, for just being there through my travails and successes.
# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERMISSION TO USE</td>
<td>i</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>ii</td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>iv</td>
</tr>
<tr>
<td>TABLE OF CONTENTS</td>
<td>v</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>ix</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>xi</td>
</tr>
<tr>
<td>LIST OF ABBREVIATIONS</td>
<td>xii</td>
</tr>
<tr>
<td>1 Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Effects of Underground Mining</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Microseismic Events</td>
<td>2</td>
</tr>
<tr>
<td>1.3 Rockbursts</td>
<td>4</td>
</tr>
<tr>
<td>1.4 Microseismicity and Potash Mining in Saskatchewan</td>
<td>5</td>
</tr>
<tr>
<td>1.5 Microseismic Networks</td>
<td>6</td>
</tr>
<tr>
<td>1.6 Portable Data Acquisition System</td>
<td>7</td>
</tr>
<tr>
<td>1.7 Motivation</td>
<td>7</td>
</tr>
<tr>
<td>1.8 Research Goals</td>
<td>8</td>
</tr>
<tr>
<td>1.9 Summary</td>
<td>9</td>
</tr>
</tbody>
</table>
2 Data Acquisition Subsystems

2.1 Transducers
   2.1.1 Piezoelectric Accelerometer
   2.1.2 Geophone

2.2 Signal Conditioning
   2.2.1 Signal Amplification
   2.2.2 Filtering

2.3 Data Acquisition Processor
   2.3.1 Signal Multiplexing
   2.3.2 A/D Converter
   2.3.3 On-board Memory for Data Buffering
   2.3.4 High Speed Data Transfer to Host Computer

2.4 Host Computer and Software

2.5 Summary

3 Hardware

3.1 Anti-aliasing Filter
   3.1.1 Filter characteristics
   3.1.2 Cascaded Filters
   3.1.3 Eighth-Order Filter Design
   3.1.4 PC Compatible Filter Board
   3.1.5 Design Observations

3.2 DAP1216a
   3.2.1 Host computer interface
   3.2.2 Pipes and Data Communication
   3.2.3 Tasks and Procedures
   3.2.4 Custom Command
6.0.1 Future Work .................................................. 78

References .................................................. 79

A Program Block Diagrams .................................. 82
  A.1 MISER-Block Diagram ..................................... 83
  A.2 MISCON - Block Diagram .................................. 86
  A.3 MSED - Block Diagram ..................................... 87

B Mass-spring Transducer Response ...................... 90
  B.1 Mass-Spring Transducer .................................. 91

C Bessel Filter .................................................... 94
  C.1 Bessel Transfer Functions ............................... 95
  C.2 Component Values and Scaling ......................... 96
  C.3 Eighth Order Bessel Filter Design Procedure ......... 98
List of Figures

2.1 Functional Blocks in a Data Acquisition Network ........... 12
2.2 Expected Response of a Velocity Transducer ............... 13
2.3 Expected Response of an Accelerometer ............... 14
2.4 Typical Configuration of a Piezoelectric Accelerometer .... 15
2.5 Geophone ........................................ 16
3.1 Second-Order Low Pass Filter ................................ 28
3.2 Magnitude Response of Eighth Order Bessel Filter .......... 29
3.3 Phase Response of Eighth Order Bessel Filter .......... 30
3.4 DAP1216a System .................................. 33
3.5 Dataflow Diagram of Typical Application ................. 36
4.1 Typical Signals Recorded by a Microearthquake Network, USGS, Central California .................. 43
4.2 MAIN Menu ........................................... 51
4.3 SETUP Menu ............................................. 52
4.4 Interprogram Data Transfer ................................ 57
5.1 Test Setup .............................................. 64
5.2 Piezoelectric Transducer Output at Constant Acceleration ........ 70
5.3 Geophone Output at Constant Acceleration ................ 71
5.4 Geophone Response at Constant Velocity ................ 72
5.5 Peak Voltage From Software vs Frequency .................. 73
List of Tables

3.1 Attenuation at Different Frequencies for the Eighth-Order Bessel Filter ........................................... 31
3.2 Component Values for Second-Order Stages of Bessel Filter .......... 32
5.1 Error Percentages for Data Points in Figure 5.5 ......................... 74
List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>A/D</td>
<td>Analog-to-Digital</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>ASCII</td>
<td>American Standard Code for Information Interchange</td>
</tr>
<tr>
<td>CPU</td>
<td>Central Processing Unit</td>
</tr>
<tr>
<td>D/A</td>
<td>Digital-to-Analog</td>
</tr>
<tr>
<td>DAPL</td>
<td>Data Acquisition Processor Language</td>
</tr>
<tr>
<td>DAQ</td>
<td>Data Acquisition</td>
</tr>
<tr>
<td>DI</td>
<td>Digital Input</td>
</tr>
<tr>
<td>DI/O</td>
<td>Digital Input/Output</td>
</tr>
<tr>
<td>DMA</td>
<td>Direct Memory Access</td>
</tr>
<tr>
<td>DO</td>
<td>Digital Output</td>
</tr>
<tr>
<td>EISA</td>
<td>Extended Industry Standard Architecture (32-bit bus)</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transform</td>
</tr>
<tr>
<td>FIFO</td>
<td>First-In-First-Out buffer</td>
</tr>
<tr>
<td>FSR</td>
<td>Full Scale Range</td>
</tr>
<tr>
<td>ISA</td>
<td>Industry Standard Architecture (16-bit bus)</td>
</tr>
<tr>
<td>LSB</td>
<td>Least Significant Bit</td>
</tr>
<tr>
<td>LTA</td>
<td>Long Term Average</td>
</tr>
<tr>
<td>Mbps</td>
<td>Megabytes per second</td>
</tr>
<tr>
<td>OS</td>
<td>Operating System</td>
</tr>
<tr>
<td>PC</td>
<td>Personal Computer</td>
</tr>
<tr>
<td>PCI</td>
<td>Peripheral Component Interconnect (64-bit bus)</td>
</tr>
<tr>
<td>PCS</td>
<td>Potash Corporation Of Saskatchewan</td>
</tr>
<tr>
<td>RAM</td>
<td>Random Access Memory</td>
</tr>
<tr>
<td>SDK</td>
<td>Software Development Kit</td>
</tr>
<tr>
<td>SEGY</td>
<td>Society of Exploration Geophysicists Y format</td>
</tr>
<tr>
<td>STA</td>
<td>Short Term Average</td>
</tr>
<tr>
<td>USGS</td>
<td>United States Geological Survey</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

Interest in microseismic monitoring was not an overnight development. It was a sequence of happenings—potash mines going into active production, occurrence of light tremors in regions close to mines, and the establishment of a correlation between mine operation and frequency of tremors—that led to a growing interest in microseismicity. Once the basics were understood, attempts were made towards finding ways to monitor microseismic events and use the data to locate and learn more about their source.

1.1 Effects of Underground Mining

Increased mining depths, a result of better technology and equipment, has increased the probability of induced seismicity. Since the 1930s, seismic activities induced by mining have caused significant fatalities and damage to underground mines[1]. In June 1975, sudden failure of a 3.35-square-kilometre area at less than one kilometre depth over a potash mine in Werra province, East Germany, caused an earthquake that was felt as far as 180 kilometres from its epicentre [2, 3]. Earthquakes and rockbursts in deep gold mines of South Africa have killed mine workers and caused mines to be shut down [4, 5]. In one of the biggest mine disasters in Canada, severe ground disturbances in a coal mine in Springhill, Nova Scotia, in October
1958 claimed the lives of 75 mine workers and rendered the mine unsuitable for any further operations [6].

Ten mines went into production within eight years of the discovery of rich potash reserves in the Prairie Evaporite belt of Saskatchewan. Increased mining activities led to noticeable ground tremors near the potash mines at Esterhazy and Cory. The proximity of the Esterhazy earthquake to a mine, in an area where previous seismicity was unknown, led to a speculation that the event may have been induced by mining. Seismicity also had not been noticed in the Cory area before the occurrence of an earthquake in 1979. Dr. D. Gendzwill, who undertook to compile a list of seismic events and their epicentres, opined that though earthquakes had not been observed near six of Saskatchewan's ten potash mines, weak seismic events may be occurring at these sites [7].

In addition to generating tremors, intensive mining operations also created problems that posed financial risks to investors. One such problem encountered during mining operations in Saskatchewan mines was the influx of water into passages drilled through or below water bearing strata. Mines had to be abandoned leading to heavy financial loss [8]. A few mines that were rehabilitated, entailed heavy expenditure and manpower investment [8, 9]. Rising concerns that weak seismic events generated by highly stressed rocks might be the cause of water inflow, led geologists in the province to investigate mining-induced seismic activities and to propose solutions to mitigate their damaging influences [10].

1.2 Microseismic Events

Microseismic events are highly localized, low intensity earthquakes occurring more frequently than large scale earthquakes. They measure three or less on the Richter scale. They are the result of accumulated stresses on fracture surfaces (typically in the 1 to 10 meter linear dimension) in areas subject to frequent activities such as underground mining and oil exploration. In undisturbed rock layers, the naturally existing stresses are balanced. As mining operations tend to remove underlying
the exact nature of the fault occurring at the rock surface, the direction of shear and potential high risk areas downstream of the shear propagation. A mining company’s ultimate goal in installing microseismic networks is to be able to locate these fissures in their nascency. As it is growing fissures that lead to rockbursts, locating fissures will help identify hazardous zones in the mine and enable men and machinery to be moved to safer areas in advance.

1.3 Rockbursts

Rockbursts are geological phenomena, typically associated with deep mining, frequently occurring at a depth of approximately 600 feet below surface. A rockburst occurs when rock breaks apart due to excessive stress. The intensity of a rockburst is usually in direct proportion to the depth of the mine. Depth, geologic factors and mining activities contribute to the load on a rock mass. Static loads cause pillars to collapse while active loading induces tremors and quakes. Stresses on rock strata are also responsible for rocks exploding spontaneously.

Continuous rock pressure disintegrates the pillars and rock supporting the mining chambers, leading to collapsed walls, stopes\(^1\) and drifts\(^2\), that shut down mining operations. Sometimes this action may be sudden and violent with little or no warning signs. Pillars and angular projections of stope walls in large, worked-out areas are particularly susceptible to violent bursts. Rockbursts typically occur in series, throughout periods of days, weeks, or months. Tremors have been reportedly felt 12-15 miles (19-24 kilometers) away from the source, although a 3-4 mile (4.8-6.4 kilometer) range is more common [12].

Because underground mining disturbs the natural equilibrium of stresses in the earth’s crust, rockbursts and their after effects cannot be totally eliminated. Rockbursts are costly to human life and mining potential. A single rockburst can deci-

\(^1\)A stope is a step-like excavation underground for the removal of ore that is formed as the ore is mined in successive layers

\(^2\)A drift is a nearly horizontal mine passageway driven on or parallel to the course of a vein or rock stratum
icated that the stress developed in the brittle limestone layer immediately above the potash ore layer could be large enough to trigger the microseismic activity observed at the Cory mine site. Horizontal shear failure along planes of weakness in the carbonate rocks over the mine was the most likely mechanism for these tremors [18, 19]. With a correlation established between mining operations and microseismic events, and a knowledge of the long term consequences of such events, Potash Corporation of Saskatchewan felt it necessary to install a permanent microseismic network to monitor microseismic events.

1.5 Microseismic Networks

Networks that monitor microseismic activities contain transducers that detect, and data processing devices that store and analyse, the weak shock waves emanating from developing fractures in stressed rocks. Such networks are being widely used to detect stresses induced by excavations and controlled explosions in underground mines in Canada. Microseismic monitoring demands use of separate networks independent of existing seismic networks for earthquake monitoring. Seismic networks are primarily designed to monitor active tectonic processes over large areas. Their design parameters—recording and processing schemes, transducer geometry and data logging techniques—are not geared to gather and process the large amount of data generated by the frequent microearthquakes. Their transducers do not respond to low-intensity seismic activities and those that do are typically overdriven (saturated and/or clamped) by higher intensity seismic activities. The transducers must have high sensitivity to detect the low intensity waves and not readily driven to saturation. Also, as microseismic events occur quite frequently in the mines, more data will be generated than that for seismic events. The recording system must have enough memory to store this data.

A comprehensive analysis of a microseismic activity includes pre- and post-event data analysis. Large scale microseismic monitoring networks employ permanently installed host computers, data acquisition devices (filters and A/D converters) and a known configuration of sensors.
1.6 Portable Data Acquisition System

Transducers used in microseismic monitoring can differ in the parameter that they measure. The measured parameter can be the displacement, velocity or acceleration input to the transducer. Transducers can also differ in their construction depending on the way the actual sensing element is mounted inside them. The accuracy of measurement as well as the size of the transducer depends on its construction. Consequently, a variety of transducers can be developed for detecting microseismic events. Before using them in actual microseismic networks, it is important to test these sensors. Using the permanent networks for testing these would entail the tedious process of configuring new sensors and subsequent reconfiguration after the sensor is removed. Any errors committed while doing this will affect the ongoing data acquisition. The necessity for an independent test tool was the motivation behind development of a portable data acquisition system.

A portable data acquisition system was envisaged as a tool to acquire data from test sensors developed for use in microseismic networks. In the laboratory, this system can be used to acquire data from simulated events using the test transducer. Simultaneous data acquisition using the test transducer and a known transducer can help study the performance of the test transducer. The system is devised as a compact, ready to use equipment that can be taken to any mine site and easily installed and configured to acquire data from an actual microseismic event, using the test sensor. This data can later be analyzed, using a variety of software packages, to observe and verify the performance of the test sensor.

1.7 Motivation

Mining induced seismicity cannot be totally eliminated. But with the advanced technologies and instruments it is possible to build systems that can predict the onset of a failure in the rock or detect rock masses under high stress. Adequate warning will help evacuate men and machinery in good time. The fact that microseismic signals are generated in rock under stress is significant because it provides a basis
for a method to detect and delineate areas of high stress. This method does not require any knowledge of the mechanical properties of the rock or the state of stress of the rock.

This research was motivated by a need to test a piezoelectric transducer developed by Dr. Brian Daku and intended for use in the microseismic network installed by PCS. The primary focus was to visualize, design and develop a wideband, data acquisition system completely contained in one portable unit and capable of being remotely accessed and programmed. The general intent was to have this system as a tool for testing the performance of different transducers before using them in operational microseismic networks.

1.8 Research Goals

The goal of this research was to develop a wideband data acquisition system, to acquire test data from transducers and to verify their performance before deploying them in an actual microseismic network. The design requirements for developing such a system were as follows:

- It should be possible to easily set up the data acquisition system at any site and connect the test transducers for data acquisition.

- The recording system must be remotely controlled and administered, capable of continuous uninterrupted operation and must be programmed to allow off-loading of data for further analysis, onto devices such as floppy disks, without interrupting the data acquisition process.

- The data acquisition hardware must be capable of a) eliminating aliasing effects, b) converting the analog data to a digital form, c) processing the data, and d) supporting high speed data transfer to upload data to a remote data processor.

- The embedded software executing on the hardware must be capable of operating in real time. Event mechanisms and triggering algorithms must be user
programmable and configurable online (dynamic configuration). The software should support remote triggers (allowing users to override triggering conditions and record data). The application software must allow users to configure the input channel parameters online. Finally, the software must allow autostart of the data acquisition process. It should do all this while having user friendly features such as ease of installation, good development tools and environments and a short learning curve.

- The acquisition system used for testing sensors must be portable (so that is can be easily transported to any mine site for installation), flexible (to allow data from different types of instruments to be recorded) and robust (to withstand the operating environment inside a mine). The system is not required to analyze the data, but only to record data and merge them into a form suitable for reading with other applications (for example MATLAB).

1.9 Summary

Microseismic events are localized earthquakes occurring in regions of highly stressed rock layers. Fissures generated by these stresses are one of the major sources of microseismic events. These fissures gradually increase in dimensions and may ultimately lead to disasters such as roof fall or cave-in, resulting in loss of miners and machinery. Monitoring ground tremors can give information about the onset of a fissure. This can be used in an early warning system that can avoid damages.

Different types of transducers can be used to monitor the ground tremors arising from microseismic events. Before use in an actual microseismic event, it is important to test each new transducer and verify the integrity and accuracy of the readings obtained from it. Doing this in an actual microseismic network can disrupt ongoing data acquisition. Hence an independent test tool is required, that incorporates all aspects of a microseismic monitoring system but is capable of being easily transported, installed and configured to acquire test data.

The remainder of this thesis describes the various subsystems of a portable, wide-
band microseismic data acquisition system, continues with an explanation of the hardware and software used for data acquisition, describes the experimental setup and concludes with a discussion of results obtained from simulations in the laboratory and suggestions for future work.
Chapter 2

Data Acquisition Subsystems

Data acquisition is the process of converting a physical phenomenon, such as temperature, pressure or acceleration, into an electrical signal and analyzing the signal in order to extract information. PC-based data acquisition systems and plug-in boards are widely used in laboratories and remote sites. A data acquisition system consists of different subsystems, each performing specific functions. A data acquisition network consists of one or more data acquisition systems connected to a network of transducers that sense the physical phenomena. The process of data acquisition consists of the following functionalities: a) sense physical properties via transducers, b) condition the analog output signals from transducers via devices such as amplifiers and anti-aliasing filters, c) digitize the analog signal using A/D converters, and d) process, analyze, record and display data information using a host computer and software.

Figure 2.1 shows these functional units and their role in the data acquisition process. The transducers are stand-alone instruments that must be deployed at site. The filter, A/D converter and the host computer together constitute the portable data acquisition system.
2.1 Transducers

Transducers are elements that directly connect to the physical phenomena being measured. They convert a physical parameter into an equivalent electrical signal. Transducers used for microseismic monitoring can measure either the displacement, velocity or acceleration of the surface to which they are attached.

Most transducers that measure shock and vibration use the response of a mass-spring system. The operating frequency range of the transducer can be found by examining a graph of the sensitivity of the transducer for various values of input frequency. For a transducer measuring displacement, the sensitivity is the ratio $s_0/w_0$, where $s_0$ is the peak value of the transducer output and $w_0$ is the peak value of the input displacement. For a transducer measuring acceleration, the sensitivity is the ratio $s_0/w'_0$, where $s_0$ is the peak value of the transducer output and $w'_0$ is the peak value of the input acceleration. The relation between $s_0/w_0$ and the frequency ratio $\frac{\omega}{\omega_n}$, where $\omega$ is the selected frequency and $\omega_n$ the natural frequency of the transducer, is shown graphically in Figure 2.2. Corresponding curves for $s_0/w'_0$ are shown in Figure 2.3. The numbers in parenthesis indicate the fraction of critical damping for each curve. Both graphs show a distinct peak at specific frequencies. The peak is indicative of the increased amplitude at resonance and the corresponding frequency is the natural frequency of the transducer. The graphs can be divided into two regions - one below the natural frequency and the other above the natural
frequency. For a discussion on the derivation of the equations for sensitivity, the reader is referred to Appendix B.1

A constant value of sensitivity, for different values of input frequency, indicates that the electrical output is proportional to the mechanical input, irrespective of the frequency of the input. An accelerometer measures the acceleration of the surface while a geophone measures the velocity of the surface subjected to this mechanical input. This means that for any transducer if we plot the observed output values per unit of the input sensing parameter (which is acceleration for an accelerometer and velocity for a velocity transducer), the operating frequency range is the region where the graph is a straight line parallel to the frequency axis.

For transducers that measure displacement, it can be seen from Figure 2.2 that:

1. For low frequencies, below the natural frequency of the transducer, the graph is a linearly rising curve.

2. There is a very distinct peak at the natural frequency. The output amplitude at this frequency is a function of the damping—greater damping reduces the output amplitude.
3. For frequencies greater than the natural frequency of the transducer, the graph is a straight line parallel to the frequency axis. This indicates that the operating frequency range of the transducer lies above its natural frequency. The lower the natural frequency of such a transducer, the wider will be its operating frequency range.

For transducers that measure acceleration, it can be seen from Figure 2.3 that:

1. For frequencies above the natural frequency, the output gradually decreases.

2. There is a very distinct peak at the natural frequency. The output amplitude at this frequency is a function of the damping—greater damping reduces the output amplitude.

3. The response curve is fairly flat in the lower frequencies well below the natural frequency of the transducer. The graph indicates that the relative displacement amplitude is directly proportional to the acceleration amplitude of the sinusoidal vibration being measured, at small values of the frequency ratio $\frac{\omega}{\omega_n}$. The operating range for an accelerometer thus lies below its natural frequency.
The higher the natural frequency of an accelerometer, the larger will be its operating range.

2.1.1 Piezoelectric Accelerometer

Typical configuration of an accelerometer is shown in Figure 2.4. This is a seismic transducer utilizing a piezoelectric element in such a way that an electric charge is produced which is proportional to the applied acceleration. The piezoelectric crystal which produces the charge acts as the spring. The displacement of the mass relative to the frame is dependent upon the applied acceleration of the frame, the crystal properties, the mass and the viscous damping between the mass and the frame. The inertial force of the mass causes mechanical strain in the piezoelectric element, which produces an electric charge proportional to the acceleration. Thus the voltage generated is also proportional to the acceleration. Metallic electrodes are applied to the piezoelectric element and electrical leads are connected to them for measurement of the generated voltage.

2.1.2 Geophone

The geophone used in this research is represented by the model shown in Figure 2.5. It consists of a mass attached to a spring with a support to the earth. A coil is attached to the mass and becomes part of the total mass. When the earth moves,
the magnet and support also move. The mass tends to remain stationary and lags behind the motion of the earth. Hence there is relative motion between the coil and the magnetic field. The resultant voltage output is proportional to the rate of displacement of the coil and consequently the velocity (or speed) of motion of the earth. Hence the geophone is referred to as a velocity transducer.

2.2 Signal Conditioning

Many real-world sensors generate output signals that must be modified in some way before a data acquisition system can effectively acquire them. This front-end preprocessing is generally referred to as signal conditioning. Signal conditioning functions incorporated in the portable data acquisition system are signal amplification, filtering, and multiplexing. Signal amplification and filtering requires special circuits. Commercially available data acquisition boards contain multiplexing circuits together with the A/D converter.
2.2.1 Signal Amplification

The most common type of conditioning required is amplification. Transducer outputs are generally in the millivolt range and must be suitably amplified to obtain output in the range acceptable by the data acquisition hardware. For the highest possible accuracy, the signal should be amplified so that the maximum voltage range of the amplified signal equals the full scale range of the A/D converter.

Many commercial data acquisition boards provide multiple input ranges by using software-programmable gain amplifiers. While this allows us to change the input range, it also poses a new problem—amplification of electrical noise added by the measurement system to the signal. If the gain is on the data acquisition board, it amplifies not only the signal but also the noise added by a common measurement system source such as signal conditioner, computer interconnections or the A/D input signal. Where signals are transmitted over long cables, they also pick up electrical noise during transmission. On the other hand if the amplification is done close to the signal source, only the signal is amplified. Source proximal amplification can make a low level signal override the measurement system noise, providing a dramatic improvement in the signal-to-noise ratio.

2.2.2 Filtering

Filtering removes unwanted signals from the signals of interest. Filtering is important for AC signals because of the upper limit on the sampling rate of the A/D converter. Sampling rate refers to the number of samples that the A/D converter generates in one second. To be able to closely reconstruct the input signal from its samples, signals must be sampled at a minimum rate of twice the maximum frequency content in the input signal (Nyquist criterion). Since we have an imposed restriction on the maximum sampling rate, to satisfy the Nyquist criterion it is essential to bandlimit the input signal to a maximum of half the maximum sampling rate of the A/D converter. Low pass filtering, using filters with extremely fast roll-off in the transition band (anti-aliasing filters), is used to achieve this bandlimiting.
Chapter 3 of this thesis describes the underlying principles and design procedures for anti-aliasing filters.

2.3 Data Acquisition Processor

After signal conditioning, the sensor input is fed to the data acquisition processor. This subsystem converts the conditioned voltage or current signal into a digital format which is readable by a PC. PC compatible data acquisition processors are designed to fit into expansion slots in a computer. In addition to supporting analog and digital I/O, data acquisition processor hardware incorporates the following:

- signal multiplexing,
- A/D conversion,
- On-board memory for data storage, and
- High-speed data transfer to the PC.

2.3.1 Signal Multiplexing

Most commercially available data acquisition cards have multiple inputs, typically either 8 or 16. There are two approaches to digitizing multiple analog signals. The first approach is to use an A/D converter for each input channel. Provided sampling is initiated from a single clock source, this method permits simultaneous sampling and ensures that there is no reduction in sampling rate as the number of inputs is increased.

The second approach is to use a single A/D converter and switch each analog signal in turn to the converter. As soon as a reading is taken from one channel, the switch moves on to the next. In turn, the A/D converter processes each reading. On the digital side, the data from the different channels can be directed into different storage locations to sort the consecutive channel readings or can be sequentially
stored and retrieved in that order. The great advantage of analog multiplexing is that it provides for high channel count at a low cost and uses the minimum board space. Typically 8 differential channels can be multiplexed per multiplexing chip. Multiplexing is much simpler to implement than using individual A/D converters and where there are space constraints, as in case of PC compatible data acquisition boards, it is economical to use multiplexers against individual A/D converters for each input channel.

2.3.2 A/D Converter

The primary component in a data acquisition system is the A/D converter that samples a continuous time signal and digitizes it so that it can be transferred to a computer and stored as discrete data values. Successive approximation type A/D converters and flash A/D converters are two common types of A/D converters used in data acquisition systems. A successive approximation type A/D converter provides a fast conversion of momentary value of the input signal. It works by comparing the input with a voltage in successively decreasing proportions of the full scale range. While the signal is being compared, it is held in a sample and hold circuit. This design strikes a balance between the space constraints in compact PC compatible data acquisition boards and the conversion speed. Flash Converters provide higher conversion speeds but for the same resolution, they employ more hardware and occupy more space than successive approximation type A/D converters. Successive approximation type A/D converters are thus preferred for use in PC compatible data acquisition boards. After A/D conversion, the resulting bytes are stored in a buffer. The computer can read the converted value immediately or can allow values to accumulate in the buffer and read them in blocks of data. Quantization error is an issue but can be reduced by increasing the resolution of the A/D converter.

Parameters of particular interest, when selecting an A/D converter board, are the full scale range, resolution, permissible gain, maximum aggregate sampling rate, facility for high speed data transfer to host computer and availability of on-board memory for data buffering.
Full scale range (FSR) is the total voltage capability of the A/D converter from the minimum to the maximum voltage that it can handle. The FSR of a unipolar A/D configured for 0 V to 10 V is 10 V, while the FSR for a bipolar ±10-V A/D is 20 V. Most A/D converters offer selectable ranges so that the signal range can be closely matched with its FSR and better resolution can be obtained.

Resolution determines the smallest signal that the A/D converter can distinguish. The higher the resolution, the smaller the detectable voltage change. It is usually stated as the number of bits that the A/D converter uses to represent the digitized values. For an n-bit converter, the resolution can be calculated using the equation:

\[
\text{Resolution} = \frac{\text{FSR}}{2^n}
\]  

Most A/D converters have two or more permissible gains. The gain value is specified when programming the converter. In order to take maximum advantage of the A/D converter resolution, the input signal range must be amplified to closely match the FSR of the converter. This is achieved by selecting a proper value of the gain.

The Microstar Laboratories Data Acquisition Processor DAP1216a, used in this research, has a 16-bit converter and is configured for ±5 V. For this converter the resolution in accordance with Equation 2.1 is \((10/65536)\) V or \(152\mu\text{V}\).

When an A/D converter uses multiplexed inputs, usually the aggregate sampling rate is specified. The aggregate sampling rate is the total number of samples generated by the A/D converter in each cycle as it successively samples all the configured channels. The sampling rate for individual channels can be obtained by dividing the aggregate sampling rate by the number of channels sampled in each cycle. The number of channels to be sampled is specified in the software for configuring the A/D converter inputs. The DAP1216a has a maximum aggregate sampling rate of 100 kilosamples per second and can be configured for a maximum of 16 input channels. Thus the maximum sampling rate for individual input, assuming all 16 input channels are sampled once during each cycle of A/D conversion, is \(100/16=6.25\) kilosamples per second.
2.3.3 On-board Memory for Data Buffering

The DAP1216a begins the data transfer to the computer by issuing an interrupt request. The delay between the request and the start of a transfer is called interrupt latency. This latency varies from less than a millisecond to many milliseconds. For continuous data acquisition, the A/D converter must continue to sample and store data in a local buffer during this time. Long latency is a serious concern for real-time data acquisition. Whether the PC can accept it or not, data keeps coming in. If the rate at which data is collected is higher than that at which it is transferred to the host computer, some data will inevitably be lost. The only robust solution is for the acquisition equipment to have sufficient on-board memory, so that data is saved until the PC can service the acquisition task. Theoretically a 100-kHz A/D converter must have sufficient on-board memory to hold 100 samples if the interrupt latency is 1 ms, since the converter will generate 100 samples in 1 ms. In practice, the buffer should be several times larger. Average user requirements of current PC technology dictates 64 kB or more.

2.3.4 High Speed Data Transfer to Host Computer

An essential requirement of a data acquisition system is high speed data throughput coupled with simultaneous data processing. In order to carry out system level tasks, it is important not to tie up the PC's central processing unit (CPU) with the task of transferring data from the data acquisition processor to the host computer RAM. The ISA bus uses special circuitry on the computer motherboard to perform direct memory access (DMA) on data to and from PC RAM. DMA gives the system the ability to store samples directly into memory without any intervention by the CPU. While DMA is in process, the data acquisition board communicates directly with the bus using a special set of hardware handshake signals. DMA is a widely accepted method for capturing a continuous stream of data, either to memory or to disk. Single, demand and block-mode transfers may be used with DMA. In single mode transfers, one data value is transferred for each DMA request asserted. This is the slowest method of transfer because the DMA controller must arbitrate for the
system bus with each transfer. Block and demand transfer modes increase system throughput because the DMA controller performs several data transfers once the DMA controller has gained bus access. For block mode transfers, the DMA controller performs the entire DMA sequence as specified by the transfer count register at the fastest possible rate in response to a single DMA request from the data acquisition board. In demand mode transfers, the DMA controller performs data transfers as long as the data acquisition board asserts its DMA request. When the data acquisition board releases this DMA request, transfers are held off.

2.4 Host Computer and Software

The computer used for data acquisition can greatly affect the maximum speeds at which data can be continuously acquired. The host computer can be any of the commercially available high speed personal computers equipped with the Linux operating system, standard ISA bus and capable of DMA transfers. A limiting factor for acquiring large amounts of data is often the hard drive. It is advisable to select a high speed hard drive and make sure there is enough contiguous free space to hold data.

When integrating a third party data acquisition board with a host computer, it is essential to setup proper communication channels and synchronize operations with the operating system resident on the PC to ensure smooth transfer of data to and from the PC. Driver software is the layer of software that directly programs the registers of the data acquisition board, managing its operations and integration with the computer resources such as processor interrupts, DMA and memory. Driver software hides the low-level, complicated details of hardware programming while preserving high performance, providing the user with an easy-to-understand interface. Properly developed driver software delivers an optimal combination of flexibility and performance while also significantly reducing the time required to develop a data acquisition application.

Application software provides for engineer and user interaction with the data ac-
qquisition system. Application software usually is a collection of programs written in a high level language, such as C, that is supported by the operating system. For a typical data acquisition program, the application programs are targeted at allowing users to configure input channels, to specify parameters such as type of triggering algorithm that must be used and time length over which data must be logged, to start/stop data acquisition, to display real time data on screen and to allow offloading of data to another storage medium.

2.5 Summary

Several functional units must be integrated to develop the total data acquisition sub-system. Signal conditioning undertaken for microseismic monitoring is amplification and filtering. Amplifiers are generally integrated with the transducer. The signal is then sent to the filter, and from there to the A/D converter board. The sampled data are then transferred to the host computer. Specifications and desirable features of each functional unit are described in this chapter. Examples, where stated, use specifications of the DAP1216a data acquisition processor which has been used for this research.
Chapter 3

Hardware

Two significant hardware components of the data acquisition system developed during the execution of this project are the Bessel anti-aliasing filter and the DAP1216a. After passing through a high-impedence pre-amplifier, the electrical signals generated by the transducers are routed to an anti-aliasing Bessel filter. The conditioned output from the Bessel filter is processed by the developed data acquisition system. If a triggering event is detected, a collection of digitized data (which includes pre- and post-event data) is transferred to the host computer for further analysis.

3.1 Anti-aliasing Filter

Aliasing is an unwanted effect of sampling a waveform at a rate below its Nyquist frequency [20]. The Nyquist frequency is twice the highest frequency component present in the input waveform. From a practical perspective, the maximum speed at which signals can be sampled is limited by the speed of the converter hardware.

There are two ways to address the problems arising out of aliasing. The first is to sample at the highest rate possible, with the implicit understanding that any aliasing effects introduced by this sampling rate are not noticeable. The second, more practical, solution is to limit the frequency spectrum of the input waveform so that the sampling rate can now be set to a little over twice the frequency limit on the
input. The latter solution is commonly implemented by designing an anti-aliasing filter with a judicious selection of cut-off frequency and roll-off characteristics.

3.1.1 Filter characteristics

Filters differ in their pass band and stop band characteristics, the roll off rate in the transition band and their phase response. Filters can be designed in one of three common forms—Butterworth, Chebyshev or Bessel. For a given order of the filter, a Butterworth filter has the flatest magnitude response in the pass band while a Chebyshev filter has the steepest roll-off.

All filters exhibit a phase shift that varies with frequency. If the phase varies linearly with frequency, its effect is simply to delay the output signal by a constant time period. However, if the phase shift is not directly proportional to the frequency, components of the input signal at one frequency will appear at the output shifted in phase with respect to the other frequencies.

Microseismic monitoring applications require that the phase relationships between the signals received from various sensors be maintained. A primary goal of microseismic data acquisition is to use the data to locate the source. This is done by observing the time delay between signals received from the different sensors and using this information and known velocity of waves to generate a set of equations whose solution would give the source coordinates. It is imperative to preserve the relative phase difference between the waves arriving via different transducers in the network. A microseismic data acquisition system must employ filters that delay all frequency components equally. Bessel filters satisfy this criterion since their design is based on transfer functions that ensure constant delay over a large frequency range. Derivation of transfer functions for Bessel filters is presented in Appendix C.1.
3.1.2 Cascaded Filters

The primitive building blocks of any high order filter are the first- and second-order filters. Increasing the order of a filter gives a better roll-off while preserving the passband characteristics of the chosen filter type. Higher order filters can be generated by cascading the required number of first- and second-order filters. When filters are cascaded, the transfer function of the composite filter is the product of the transfer functions of the individual filters in the cascade.

In first-order filters, the only adjustable parameter is the cutoff frequency and the impedance level. In a second-order filter, it is possible to adjust the impedance level, the cut-off frequency and the quality factor Q. Quality factor sets the peaking or droop of the response at frequencies near cutoff. The performance of the composite filter depends on the sequence in which the first- and second-order filters are cascaded. Performance is best optimized by swapping the Q values and/or editing the cascade order and trying all possible permutations until a configuration that best matches the requirements is found.

A few guidelines that could make the process less tedious are:

- Bandlimiting of noise is best achieved by placing the second-order section with the lowest Q and lowest cut-off frequency \((F_c)\) last in the cascade order i.e., closer to the output. A second-order section with a low Q begins rolling off before \(F_c\). The lower the Q, the farther into the pass band the roll off begins. Second-order filters with high Q on the other hand, have resonance peaks around \(F_c\). As Q increases, the resulting peak also increases. Most noise in the cascaded filter comes from the stages with high Qs, with the noise being greatest near the vicinity of the resonance peaks. By placing the stage with the lowest Q and lowest \(F_c\) last in the cascade order, noise contributed by previous stages lie outside the passband of this final stage, thus resulting in overall reduction in the noise. Because the final stage has a low Q, it contributes relatively lower noise of its own.

- It is essential to give the high Q stages a DC gain of less than one and pro-
portionally increase the gain of subsequent stages. This is because if a high Q stage is given a gain of 1, frequencies in the vicinity of \( F_c \) will receive an additional boost from the resonance peak resulting in a gain greater than one. Depending on the strength of the input signal, the output from the high Q stage may saturate the following input stage, driving it into its non-linear region of operation and thus causing distortion. Setting a gain such that the the peak at \( F_c \) does not exceed 0 dB results in a DC gain of less than one for that stage. This has the effect of significantly attenuating most of the frequencies in the pass band, thereby minimising the excursions of the input amplifier. Although this reduces harmonic distortion, the noise, which increases with Q, also gets amplified in the following stages with gain greater than unity. Harmonic distortion optimization is inimical to noise optimization, so the best cascade is a compromise between these two.

3.1.3 Eighth-Order Filter Design

In designing the anti-aliasing filter for application in microseismic monitoring, the primary consideration was that the filter must have constant delay. A Bessel filter provides constant delay over a wide range of frequencies. A high order of the filter was chosen to get a steeper roll-off, so that frequency components above the cut-off frequency are substantially attenuated. Since the filter board had to be installed in a computer, space constraints did not permit the use of a filter order greater than eight.

Figure 3.1 shows the configuration of a typical second-order filter stage used in designing the composite eighth order filter for this research.

The Q values for the four stages, as listed in Table 10.4 in [21], are \( Q_1 = 1.226 \), \( Q_2 = 0.711 \), \( Q_3 = 0.506 \) and \( Q_4 = 0.056 \). The capacitance values for the second order normalized filter stages are \((2.452, 0.40783), (1.4218, 0.7033), (1.1192, 0.8935), \) and \((1.012, 0.988)\). These values are listed in ordered pairs \((C_1, C_2)\) beginning with the first stage corresponding to \( Q_1 \). The filter was designed in accordance with the procedure explained in Appendix C.1. To facilitate trials with component values,
a MATLAB program was written to calculate the theoretical values and scaled practical values of the components and the new values of \( Q \) for the practical circuit. The program also generated the transfer function and the frequency and phase response for the individual filter stages and the composite eighth-order filter.

The designed filter has a roll-off rate of 160 dB/decade (measured between 10 kHz and 100 kHz) and cut-off frequency at 2485 Hz. Component values obtained from MATLAB were used to simulate the filter circuit using ICAP4/SPICE simulation package. The magnitude and phase response obtained are shown, respectively, in Figure 3.2 and Figure 3.3.

The input voltage \( (V_{\text{in}}) \), and output voltage after passing through the filter \( (V_{\text{out}}) \), are related as:

\[
V_{\text{out}} = V_{\text{in}} \times 10^{\frac{A}{20}}
\]  \hspace{1cm} (3.1)

where \( A \) is the attenuation in decibels.

The attenuation obtained at the various frequencies and the corresponding ratio of the filter output voltage to the filter input voltage are listed in Table 3.1.

The designed filter has a pass band gain very close to unity up to a frequency of
Figure 3.2: Magnitude Response of Eighth Order Bessel Filter

about 700 Hz. Beyond this frequency, the gain starts to drop gradually, till it reaches -3 dB at 2.485 kHz. At frequencies above the cut-off frequency, the filter has a roll-off rate of 160 dB/decade. The filter has linear phase over an appreciable range, from 0 Hz to about 100 Hz.

3.1.4 PC Compatible Filter Board

The developed data acquisition system can handle a maximum of eight input channels at the desired sampling speed of 2500 Hz. Each channel requires an individual anti-aliasing filter. A filter board containing eight composite, eighth-order filters was fabricated by Vic Meyer at the Electrical Engineering Laboratory, University of Saskatchewan.

Motorola MC34084 quad operational amplifiers were used for these filters. The MC34084 was preferred because of its wide gain bandwidth product and high slew rate, low input offset errors and bias currents. Each chip has four operational amplifiers and allows all four stages of the eighth-order filter to be incorporated. The individual second-order filters are built around the typical circuit shown in Figure 3.1. The value of the resistances and capacitances for each stage are shown
in Table 3.2. Stage 1 is the stage closest to the input while stage 4 is the final filter output stage. The resistance values stated above are in kilo-ohms while the capacitance values are in nanofarads.

The filter board has connectors for connecting to the analog inputs on one side and to the Microstar DAP1216a on the other. This filter board plugs into one expansion slot in the host computer.

3.1.5 Design Observations

A few observations emerged out of the various trials for the filter. These are:

1. A high value of $Q$ for the initial stages greatly increased the roll-off rate, but it also increased the overshoot near the cut-off frequency.

2. The capacitance $C_2$ greatly determined the value of the cut-off frequency. Lower value of $C_2$ yields a higher cut-off frequency.

3. Resistance values primarily effect the roll-off. It is good design practice to start with resistances of 1 Ω and scale them to obtain practical values over
<table>
<thead>
<tr>
<th>Frequency</th>
<th>Attenuation</th>
<th>V_{out}/V_{in}</th>
<th>Frequency</th>
<th>Attenuation</th>
<th>V_{out}/V_{in}</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$-1.15 \times 10^{-3}$</td>
<td>0.99987</td>
<td>1100</td>
<td>$-173.0 \times 10^{-3}$</td>
<td>0.98028</td>
</tr>
<tr>
<td>10</td>
<td>$-1.15 \times 10^{-3}$</td>
<td>0.99987</td>
<td>1200</td>
<td>$-231.0 \times 10^{-3}$</td>
<td>0.97376</td>
</tr>
<tr>
<td>100</td>
<td>$-1.15 \times 10^{-3}$</td>
<td>0.99987</td>
<td>1300</td>
<td>$-312.0 \times 10^{-3}$</td>
<td>0.96472</td>
</tr>
<tr>
<td>150</td>
<td>$-1.20 \times 10^{-3}$</td>
<td>0.99986</td>
<td>1400</td>
<td>$-425.0 \times 10^{-3}$</td>
<td>0.95225</td>
</tr>
<tr>
<td>160</td>
<td>$-1.21 \times 10^{-3}$</td>
<td>0.99986</td>
<td>1500</td>
<td>$-531.0 \times 10^{-3}$</td>
<td>0.94070</td>
</tr>
<tr>
<td>180</td>
<td>$-1.26 \times 10^{-3}$</td>
<td>0.99985</td>
<td>1600</td>
<td>$-647.0 \times 10^{-3}$</td>
<td>0.92822</td>
</tr>
<tr>
<td>200</td>
<td>$-1.31 \times 10^{-3}$</td>
<td>0.99985</td>
<td>1700</td>
<td>$-853.0 \times 10^{-3}$</td>
<td>0.90646</td>
</tr>
<tr>
<td>263</td>
<td>$-1.70 \times 10^{-3}$</td>
<td>0.99980</td>
<td>1800</td>
<td>$-1.050$</td>
<td>0.88614</td>
</tr>
<tr>
<td>300</td>
<td>$-2.09 \times 10^{-3}$</td>
<td>0.99976</td>
<td>1900</td>
<td>$-1.230$</td>
<td>0.86796</td>
</tr>
<tr>
<td>400</td>
<td>$-4.05 \times 10^{-3}$</td>
<td>0.99953</td>
<td>2000</td>
<td>$-1.410$</td>
<td>0.85016</td>
</tr>
<tr>
<td>500</td>
<td>$-8.23 \times 10^{-3}$</td>
<td>0.99905</td>
<td>2100</td>
<td>$-1.770$</td>
<td>0.81564</td>
</tr>
<tr>
<td>600</td>
<td>$-16.7 \times 10^{-3}$</td>
<td>0.99808</td>
<td>2200</td>
<td>$-2.110$</td>
<td>0.78433</td>
</tr>
<tr>
<td>700</td>
<td>$-31.0 \times 10^{-3}$</td>
<td>0.99644</td>
<td>2300</td>
<td>$-2.440$</td>
<td>0.75509</td>
</tr>
<tr>
<td>800</td>
<td>$-47.5 \times 10^{-3}$</td>
<td>0.99455</td>
<td>2400</td>
<td>$-2.750$</td>
<td>0.72862</td>
</tr>
<tr>
<td>900</td>
<td>$-80.4 \times 10^{-3}$</td>
<td>0.99079</td>
<td>2485</td>
<td>$-3.00$</td>
<td>0.707795</td>
</tr>
<tr>
<td>1000</td>
<td>$-110.0 \times 10^{-3}$</td>
<td>0.98742</td>
<td>2500</td>
<td>$-3.05$</td>
<td>0.70388</td>
</tr>
</tbody>
</table>

Table 3.1: Attenuation at Different Frequencies for the Eighth-Order Bessel Filter

$10\, \text{k}\Omega$.

Resistance and capacitances refer to the configuration shown in Figure 3.1.

### 3.2 DAP1216a

The DAP1216a (Figure 3.4) is a commercial off-the-shelf data acquisition board manufactured by Microstar Laboratories. An onboard digital-to-analog converter samples the analog input (the output of the anti-aliasing Bessel filter), digitizes the sampled value and transfers the digital data to the host computer.

The DAP1216a is a stand-alone data acquisition processor board typically setup as
Table 3.2: Component Values for Second-Order Stages of Bessel Filter

<table>
<thead>
<tr>
<th></th>
<th>Stage 1</th>
<th>Stage 2</th>
<th>Stage 3</th>
<th>Stage 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>19.1</td>
<td>19.1</td>
<td>22.1</td>
<td>20</td>
</tr>
<tr>
<td>R2</td>
<td>19.1</td>
<td>19.1</td>
<td>22.1</td>
<td>20</td>
</tr>
<tr>
<td>C1</td>
<td>4.7</td>
<td>3.9</td>
<td>2.2</td>
<td>3.9</td>
</tr>
<tr>
<td>C2</td>
<td>1.0</td>
<td>1.5</td>
<td>1.0</td>
<td>2.7</td>
</tr>
</tbody>
</table>

A peripheral device on a host computer's ISA/EISA bus. Its highlights are:

- A 16-bit microprocessor (Intel 80C186XL @20 MHz), a 16-bit A/D converter (AD976), a 16-bit D/A converter (AD669), 1 MB on-board RAM.
- A multi-tasking real-time operating system (DAPL) that executes from the onboard RAM.
- Onboard timers and control logic to offload input processing overheads from the host computer.
- Sixteen single-ended or eight differential inputs separately programmable for selected voltage gains (1x, 10x, 100x, 500x) and input voltage ranges (±5 V, ±10 V).
- 100 kilosamples per second maximum aggregate data sampling rate over all inputs.
- Two analog outputs, sixteen digital input and sixteen digital output channels. Additional boards increase the analog capacity to 512 single-ended inputs and the digital capacity to 128 inputs and 1024 outputs.
- 512-byte high-speed, full-duplex FIFOs to communicate with the host processor. Communication content includes user programs, system commands, binary/text data and error messages. FIFOs allow multiple DAP1216a boards to share interrupts.
3.2.1 Host computer interface

The DAP1216a occupies one expansion slot in the host architecture and communicates with the host over ISA bus. The host used for this project is a 133-MHz, Pentium-based personal computer executing the Linux operating system.

The board is supplied with a proprietary operating system (named DAPL) and software (named accel) that implements a communication interface as standard device drivers on the Linux operating system. The DAPL supports task scheduling and task concurrency. It interacts with users through a simple command interpreter. It has a rich library of built-in commands to configure the board and a mechanism to extend this collection with user defined commands.
3.2.2 Pipes and Data Communication

DAPL tasks communicate via first-in-first-out buffers named pipes. Pipes are an abstraction that support full-duplex communication among generating modules (those that send data into a pipe via input ports) and user modules (those that read data from a pipe via output ports). Pipes are dynamically allocated and hence are capable of resizing to fit all data. Pipes can be text (capable of handling and interpreting character data) or binary (handling raw binary data with no special interpretations). Serial pipes transfer data on a serial port while parallel pipes transfer data over a parallel data bus. Binary pipes are always parallel while text pipes may be serial or parallel.

Communication pipes transfer data between the DAP1216a and the host computer. The DAP1216a has four built-in communication pipes. DAPL commands are read from SYS(IN) while status and error messages are written out on SYS(OUT). Event data is written out on BIN(OUT). Additional communication pipes can be defined to enable several binary and text pipes to transmit and receive simultaneously, but these must be declared in DAPL and defined/configured in the accel driver executing on the host PC.

Channel pipes are pipes used to store data related to DAP1216a's analog and digital inputs and outputs. Input channel pipes (IPIPE) store sampled data from an input pin and while output channel pipes (OPIPE) store data to be sent to an output pin. Channel pipes are assigned to inputs or outputs on a one-to-one basis using the SET command within an input or output procedure.

User defined pipes are used for data transfer between tasks. These pipes must be defined before they are used. When a task places its output in a user defined pipe, other tasks can access the data and operate on them. More than one task can read data from the same pipe.
3.2.3 Tasks and Procedures

User-defined commands are implemented via tasks and procedures that can be specified in the C programming language. A task is any command that is executed by DAPL, and is specified by a task name and its parameter(s). A procedure is a collection of tasks that are executed concurrently. Procedures are classified into one of three forms:

1. Input procedures: Input procedures consist of input configuration commands such as those that set sample rates and associate input channels to communication pipes.

2. Processing procedures: Processing procedures typically consist of tasks that manipulate the input data, such as, for example, to calculate sum and mean or call custom commands that implement specific algorithms.

3. Output procedures: Output procedures typically consist of tasks such as setting output parameters for the waveform generators and associating output channels to the host processor’s communication pipes.

Tasks can be nested within other tasks inside a procedure. Each concurrent task execution generates a separate runtime environment. Tasks are executed whenever the procedure containing that task is executed. The START and STOP commands respectively initiate and terminate procedure execution.

Figure 3.5 is a data flow representation of a typical application. It contains instances of input, processing and output procedures and pipes.

An input procedure specifies that data from (differential) input D0 be routed to channel pipe IP0 and data from (single-ended) input S0 be routed to channel pipe IP1. The processing procedure consists of the ADD, LIMIT, PEAK and INTEGRATE tasks. ADD adds the samples from the two inputs and stores the result in user-defined pipe P1. LIMIT limits the values in the user-defined pipe P1 between a maximum and minimum value (specified as task parameters) and routes the output to the user-defined pipe P2. PEAK obtains the peak value of a block of data stored
in the user-defined pipe P2 and transfers it to the host computer. INTEGRATE integrates the data stored in the user-defined pipe P1 and routes it to the user-defined pipe P3. This data is copied to the output channel pipe OP1. The output procedure routes the data in the output channel pipe OP1 to the analog output.

3.2.4 Custom Command

Custom commands are user-defined processing task commands that extend DAPL by implementing specialized functionality not available with the built-in command set. Custom commands are C-language programs compiled into object code that can execute on the DAP1216a. The software development kit (SDK) supplied with DAPL provides a C language environment. The SDK library has functions that access data, control the DAP1216a hardware and transparently replace the standard runtime C libraries. The SDK also contains a cross compiler that executes on a PC
and generates executables for the DAP1216a. Once downloaded into the DAP1216a, custom commands can be invoked as any other built-in DAPL command.

Custom commands invoke library routines to perform operations specific to the DAPL environment. The most commonly used routines provide access to the DAPL pipes and their data. Other routines configure data access characteristics, control the output hardware, manipulate DAPL software triggers, generate timing events, provide fixed-point arithmetic operations, construct waveforms, perform fast fourier transform, digital filtering operations and multitasking control.

A DAPL configuration file consists of a list of tasks, each task built from pre-defined or custom commands. Additionally, the task definition specifies the data sources and destinations of the command. The DAP1216a activates a custom command on receiving a START command for the processing procedure containing that task. Custom command execution begins with the extraction and validation of the command's parameters followed by the code specified in the initialization section of the task. The main body of the task is a continuous loop that reads data from pipes or variables, processes the data and writes the results to pipes or variables. The task terminates when the DAP1216a is halted.

Custom commands are recommended in the following situations:

1. Several predefined operations need to be merged into a single task. This improves processing efficiency by reducing pipe access overhead.

2. A custom command can read input data and process it as a single task. This minimizes the number of tasks in a system and reduces latency between input and output and is recommended for closed-loop control systems which require fast response times.

3. Tasks that use DAPL expressions can be coded as programs that utilize C-language operators. Compiling such programs and executing them natively on the host processor significantly improves the efficiency of the task and lightens the load on the DAPL.

Users must ensure that all data structures are correctly initialized before real-time
processing is started. A custom command must open a pipe in the read or write mode before performing pipe I/O. Custom commands that use blocks of data must also set up a pointer to a pipe buffer structure that encapsulates the state of a pipe, namely, the amount of data available, a pointer to the storage buffer, and the minimum and the maximum amount of data containable in the buffer. Each operation that fetches data from a pipe or places data in a pipe requires operating system overhead. This overhead limits the maximum rate at which data can be transferred into and out of pipes. Blocked pipe operations increase the pipe input/output rate by operating on blocks of data. The block size can be defined in the commands used for the pipe read and write operations. Blocked pipe operations are conducive to continuous data acquisition systems requiring frequent transfer of large amounts of data.

3.2.5 Device Driver

Application programs on the PC are responsible for user interfaces and other activities such as formatting and logging data to disk files. The DAP1216a handles the data buffering and real-time processing. Programs executing on a host PC communicate with the DAP1216a via accel drivers. The accel driver implements the communication protocol between the operating system and one or more DAP1216a. In the Linux operating environment, the accel driver allows application programs to treat the DAP1216a as a file and access it via standard file access functions. Any programming language that supports file input and output can communicate with the DAP1216a via accel. The accel driver configuration files define communication pipes in the host computer and establish a one-to-one relationship with communication pipes in the DAP1216a.

The accel devices are of two types—numbered or directed. Application programs may choose to use one or the other, but not both at the same time. Numbered devices communicate with specific PC communication pipes (based on their numbers) and are accessed as files by using their device name as the file name. Directed devices communicate with exactly one active input pipe and one active output pipe; application programs are allowed to dynamically redefine active pipes and simulate
multiple connectivities.

Systems with one DAP1216a typically use one output communication pipe '0' and one input communication pipe '1'. Communication on pipe '0' is established by accessing the numbered device accel0 as an output file while communication on pipe '1' is established by accessing accell as an input file. Commands are transmitted to the DAP1216a via communication pipe '0', while data from the DAP1216a is received via communication pipe '1'. Binary communication provides the fastest means for data transfer between the DAP1216a and the application and is preferred. This is because it requires fewer data transfers and eliminates the need for data conversions from binary to ASCII.

3.2.6 Programming Considerations

Application programs that communicate with the DAP1216a must contain code that correctly initializes and establishes all communication channels before performing any data acquisition operations. Application programs must also properly terminate in a well-defined state.

Proper initialization is achieved by issuing a RESET command followed by a forced flushing of all text and binary data. This ensures that there is no stale data that may interfere with the correct execution of a program. A good programming technique is to create a file which has a list of commands that define the input, output and processing procedures for a specific application. This file must be downloaded to the DAP1216a before the start of any application. In applications where the DAP1216a must be customized at run time, this configuration file must be generated from user defined inputs, using programming commands that write to a file. Once generated, it must be loaded to the DAP1216a. C programs typically use the 'sprintf' command to write DAPL commands as text strings to a file and then load the file to the DAP1216a using 'DAPSendFile' command. The defined procedures can then be started or stopped at will by sending the START or STOP commands through the text pipe.
Proper termination is achieved by issuing a STOP command followed by a forced flushing of all active communication pipes. This ensures that stale data is not carried over to the next invocation of an application program.

3.3 Summary

Vibrations generated by microseismic events are detected using transducers which generate proportional analog electrical signals. Anti-aliasing filters are required to condition these signals before they are sampled by the A/D converter. This is done to avoid distortion, so that the original waveform can be closely reconstructed from its samples. Of the various filters available, Bessel filter exhibits a phase shift proportional to the frequency, over the greatest range of frequencies. The filter can be designed from first principles using the roots of the Bessel polynomial. The values of the capacitances and resistances have to be appropriately scaled to obtain a filter with the desired cut-off frequency.

The output from the anti-aliasing filter is sampled by the data acquisition processor. The DAP1216a from Microstar Laboratories is a PC compatible data acquisition board that has a 16-bit A/D converter, multiplexer and on-board memory for sampling and storing data temporarily before it is transferred to a host computer memory. The DAP1216a has its own real-time operating system, the DAPL, which allows continuous sampling and processing of data as defined by custom commands. Familiarity with the concepts of procedures, tasks and buffered data transfer using text and binary pipes is essential to develop the strategy for any application using the DAP1216a.

This chapter provides the design principles for the anti-aliasing filter and also a comprehensive information on the DAP1216a and concepts that form the basis of implementation of software used for microseismic monitoring.
Chapter 4

Algorithms and Software

4.1 Introduction

Previous chapters discuss the hardware required to acquire data from microseismic activity. This chapter presents the algorithm and software that complements the hardware. It describes the event triggering algorithm, the applications running on the host computer, the STA/LTA algorithm for microseismic monitoring, and the setup and installation of the system.

A wideband, microseismic data acquisition system should be capable of distinguishing natural microseismic signals from artificial signals (sudden vibrations generated by a passing vehicle or machines going into operation) and storing only the relevant data. Efficient design of a microseismic data acquisition system must answer the following key questions:

- Does the incoming signal represent an actual microseismic event?
- What is the minimal amount of data to be permanently stored?
- In what format should one store the data?

The solutions are in the choice of an algorithm to trigger events, the time window
over which data must be permanently recorded, and an industry-wide format for data storage.

4.2 Event Detection

On-site transducers are prone to respond to spurious signals not generated by the phenomenon under study. To reduce processing and memory overheads, data acquisition systems incorporate algorithms to identify relevant events and trigger data acquisition on detecting such events.

Triggers may be generated based on an absolute value or a ratio of signal averages over short-term and long-term intervals. The threshold mode compares input data value to a predefined threshold value and generates a trigger if the input is larger than a specified multiple of the threshold. A modification of this mode uses a statistically derived value, such as an average or a median value of the input data values. The ratio mode is based on the ratio of the short term average (STA) of an input to its long term average (LTA). If this ratio is greater than a specified value, a trigger is generated.

4.3 Triggering Algorithm for Microseismic Event Detection

Figure 4.1 is a reproduction of the signals recorded on the short-period, vertical component seismographs of the USGS Central California microearthquake network [22]. The background signals typically have a low amplitude and a low frequency, and their appearance can be either quite periodic (Figure 4.1a) or quite random (Figure 4.1b). Because of instrument noise and activities going on in the vicinity, transient signals over a broad-range of amplitudes and periods are often present. For example, a transient noise from a telephone line is shown in Figure 4.1c. A more emergent type of noise is shown in Figure 4.1d. Its envelope, indicated by the dashed line, is roughly elliptical in shape. Seismic signals are generally classified into three
Figure 4.1: Typical Signals Recorded by a Microearthquake Network, USGS, Central California

types depending on the distance from the source to the recording station. Earthquakes occurring more than 2000 km from a seismic network usually are called teleseismic events. An example of a teleseismic event as recorded by a microearthquake network station is shown in Figure 4.1e. Depending on the size of the event, teleseismic amplitudes can range from barely perceptible to those that saturate the instrument. Their predominant periods are typically a few seconds. Earthquakes occurring within a few hundred kilometres to 2000 km outside the network are called regional events. An example earthquake as recorded by a microearthquake network station is shown in Figure 4.1f. As with the teleseismic event, amplitudes of regional earthquakes can range from barely perceptible to large, but their predominant periods are less than those of the teleseismic events. Earthquakes occurring within a few hundred kilometres of a seismic network are called local events. Local earthquakes are often characterized by impulsive onsets and high-frequency waves as shown in Figure 4.1g and Figure 4.1h. The envelope of local earthquake signals typically has
an exponentially decreasing tail as shown by the dashed line in Figure 4.1g. Another characteristic of all (teleseismic, regional and local) earthquake signals is that the predominant period generally increases with time from onset. On the other hand, a given transient signal originating from instrument noise or human activity, often has the same predominant period throughout its duration.

In short, the incoming signals from a micro-earthquake network can have a wide range of amplitudes, but local earthquakes are characterized generally by their impulsive onsets, high-frequency content, exponential envelope, and decreasing signal frequency with time [22].

If the signal amplitude over time is observed, along with its short-term and long-term moving averages, it is noticed that the difference in the two averages is significant when there is a major seismic event. An event detection program, based on a ratio of moving averages, commonly referred to as the STA/LTA algorithm, has been extensively studied and employed in microseismic studies [23, 24, 25]. McEvilly and Majer [25] have shown that a judicious selection of the ratio and observation windows of STA and LTA agrees with virtually all results of interest in conventional microearthquake surveys.

Using trial-and-error techniques, Dr. A. Prugger [26] showed that with observation windows of 0.3-0.6 seconds for the short term averaging function and 5-10 seconds for the long term averaging function, an STA/LTA ratio between 1.5 and 2.5 would be ideal for detecting significant seismic activities in mines operated by the Potash Corporation of Saskatchewan. If the threshold level is set too low, noise bursts will cause too many false detections. If the threshold level is set too high, the number of false detections will decrease and so will the number of detected earthquakes.

4.3.1 Algorithm Implementation

An actual implementation of the STA/LTA algorithm involves the following steps:
1. The time series $X(t_j)$, representing the input signal, is digitized at a predetermined sampling rate.

2. The mean value is subtracted from each value that is used to generate the sum of samples in a data block.

3. A new time series $X'(t_j)$ is formed by

$$X'(t_i) = \frac{\sum_{j=i-n}^{i} |X(t_j)|}{n}$$

where $n$ is the number of samples per data block.

4. A long term average and a short term average of blocks of data is calculated. Typical values of the window duration for the short term average are below one second. This permits a quick response to changes in the signal level that characterise the onset of an earthquake. Typical values of the window duration for the long term averages depend on the application and range from a few seconds to a few minutes.

5. If the ratio of STA to LTA exceeds a predefined value, then a trigger point is found.

6. If no trigger is found, the program waits until it obtains the next sample and repeats from step 1.

Data from each input channel is tested for a trigger point. In microseismic applications, the transducers are spread over a wide area. It is possible that local activity near one transducer will generate signals that might satisfy the triggering condition for that input and signal an event. To avoid false triggering under such circumstances, microseismic event detection algorithms additionally specify the minimum number of inputs that must be simultaneously triggered to precipitate an event detection.

4.4 Data Acquisition Software

The application software used in this project consists of programs written in the C programming language. MISER (MIcroSeismic Event Recorder) handles data acquisition while MISCON (MIcroSeismic CONsole) handles user interfaces and other
activities needed to establish communication with Miser. The program MSED.R (MicroSeismic Event Data Recorder) acts as custom command and implements the event detection algorithm. The benefits of such a modular architecture are threefold:

- A well-defined interface between the user and the data acquisition program.
- The ability to detach the data acquisition program from the user and execute it as a background process. This allows for unattended program execution for extended durations thereby generating a significant history of data that can be used as a benchmark for future analysis.
- The ability to execute Miser on a hardware platform installed in the field and control its execution from a remote site that is executing MISCON.

4.4.1 Miser

Miser is the program whose key function is data acquisition and formatting. The module executes continuously, providing constant updates on the status of the data acquisition process. It also provides the following application programming interfaces (APIs):

- Configure: This collection of APIs allows users to specify various configuration parameters that are converted to internal data structures and perform the following functions:
  - Set the number of input data acquisition channels, transducer type and positional coordinates.
  - Set the minimum number of channels that must be simultaneously triggered for an event to be recognized.
  - Set triggering modes. Available modes are threshold (a fixed value) or ratio (the ratio of the instantaneous value to the long-term averaged value).
- Set the trigger reference. The threshold and ratio triggering modes use this value to determine when to trigger an event.

- Set time intervals of the STA and LTA windows.

- Set the duration over which the data must be observed and analysed before (pre-trigger window) and after (post-trigger window) a trigger is detected.

This API is invoked using key 'A' from the program MAIN menu.

- Status: When invoked, this API responds with a menu that displays the status of the A/D converter (idle, running or recording an event) and real-time event data as it is being recorded. This API is invoked using key 'B' from the program MAIN menu.

- Event Trigger: This API simulates an external event to the data acquisition system, precipitating data logs that can be used as benchmarks or for post-event analysis. This API is invoked using key 'C' from the program MAIN menu.

- Start/Stop: This API is used to start and stop the data acquisition process. It is invoked using key 'D' from the program MAIN menu.

- Move: This API moves files from the working directory (/miser/data) to another directory (/miser/download) from where they can be offloaded to soft media. This option ensures that there is no disruption in data logging when off-loading event records. This API is invoked using key 'E' from the program MAIN menu.

- Detach: This API forces MISER to detach from a user terminal and execute in the background. Users can reconnect by executing the MISCON command. This API is invoked using key 'F' from the program MAIN menu.

- Waveforms: This API allows users to program the data acquisition system to generate waveforms at the analog output port. User configurable attributes include the envelope (square, triangular, sawtooth, sinusoidal), frequency, and
type (pulsed/periodic). These waveforms are used to self-calibrate the system. This API is invoked using key 'S' from the program MAIN menu.

- Terminate : This API allows user to halt the A/D converter and gracefully stop the data acquisition process. This API is invoked using key 'Q' from the program MAIN menu.

Other modules within the program perform the following functions:

- Set data acquisition start/stop times. This can be used, for example, to specify repeated, periodic execution of data logging programs.

- Set input/output data attributes such as memory locations and buffer sizes.

- Generate filename and formatted data for an event. Data pertaining to each event is stored in a SEGY file whose name is a concatenation of the event number (representing a unique identifier) and the event occurrence time. The formal syntax of such a filename is “Xyyyyddhmmss.sgy” where X is a one letter code, yyyy is the year, ddd is the day of the year(1-366), hh is the hours(0-24), mm the minutes and ss the seconds. The SEGY file name can be easily interpreted by users and batch processing programs responsible for various post-processing activities.

- Upload sampled data to host computer. When a trigger is generated, sampled data corresponding to the pre- and post-event duration is transferred from the DAP1216a on-board memory to the host computer. This action momentarily suspends DAP1216a activity till the transfer is complete.

- Log event and activities to a log file. Event data includes SEGY file name, the number of channels triggered and the peak value for each event. Activity data includes program startup and termination times, error messages, triggers, A/D converter actions, file manipulation errors and memory allocation errors.

A top-level flow chart of the program logic is presented in Appendix A.1. MISER accepts two command line options: '-s', '-d'. The '-s' option allows users to manually
start and stop the DAP1216a; the default is to initiate execution of all input/output procedures at MISER startup. The -d option forces a diagnostic message to be logged whenever a data spike (i.e., two triggering events happening in quick succession) is detected by the DAP1216a.

The program begins by logging the command line options and the start times (obtained by reading the system clock and interpreting that as the start time). Dynamic memory is allocated for the various internal data structures (including the data logging structure). Values in the structures are initialized (some with default values, some with values read from 'miser.set' file, and some others calculated from user specified values). The 'miser.set' file contains information about the transducer types, trigger values and pre- and post-event window length and is generated at run-time based on the values entered in the SETUP menu. Program variables that depend on user inputs are also calculated. Based on all these values, the program generates the DAP1216a configuration file. The DAP1216a configuration file defines the input channel pipes, the sampling rate and the task and procedure definitions. This file is also generated at run time and uses information stored in the 'miser.set' file.

The DAP1216a is initialized by opening the device driver pipes (for reading and writing) and downloading the configuration file to the DAP1216a's on-board RAM. If the -s option was not specified, all procedures are started. FIFO buffers MISERIN and MISEROUT are initialized and the date and status fields in the STATUS menu are updated.

The program enters an infinite loop performing the following activities:

1. Update the current time.

2. Update the time-out reference counter if a keystroke is just received or if the terminal is not connected.

3. Update the STATUS menu display.
4. Check for activity on the input port. If activity is detected, draw the STATUS menu and discard the key. If no activity is detected, proceed to get the data from the converter as specified in “GetAData” function.

5. “GetAData” function is responsible for getting the multiplexed data samples from the A/D converter and generating the SEGY file for the event. This function causes the DAP1216a to check for data in the binary input channel pipe. Block mode is used for data transfer in these programs. In the block mode, the DAP1216a waits until a specified number of samples per channel are collected in the respective input channel pipes and then transfers them together to the program buffer rather than passing each sample as it is generated. For the first data block, a temporary file is opened in the write mode. A sample counter is used to keep track of the number of samples that must be transferred each time. Depending on the pre- and post-event time durations specified in the SETUP menu, a total event may be contained in more than one block of data and the number of samples to be transferred may not be an exact multiple of the predefined block size. Hence it is required to check if the remaining number of samples corresponding to the current event is less than the the block size. If the number of samples to be transferred is indeed less, then the sample counter must be accordingly updated so that the data samples corresponding to another event do not get mixed with the current event samples. After total event data corresponding to the pre- and post-trigger window length is read, communications between the DAP1216a and the host computer is paused. Data is transferred from the data acquisition processor on-board memory to the host computer and converted to the SEGY format. This conversion process returns a status message. If the conversion is successful, it returns value 0. The event log is then updated and a diagnostic message written to the diagnostic log file. If a spike is detected, and the '-d' option is specified, only the diagnostic message is written to the diagnostic log file. Communication with the host computer is resumed and control returns to the MISER program which waits until more samples are gathered. Steps 18 to 26 of the program logic are repeated until the QUIT signal is received from MAIN menu.
6. When the program is terminated by the QUIT signal from MAIN menu, close the diagnostic log file and exit the program.

The MAIN menu options are shown in Figure 4.2.

![Diagram of MAIN Menu]

**Figure 4.2: MAIN Menu**

This menu has eight options. The letters shown by the side of the arrows denote the key used for that option. For example, when in MAIN menu, entering key 'B' displays the STATUS menu. Similarly using key 'M' in the STATUS menu takes the user back to the MAIN menu. The SETUP menu allows the user to specify the transducer type, the triggering algorithm to be used, the time durations before and after the event for which data is to be recorded and the positional coordinates of the transducers. The STATUS menu displays the status of the A/D—whether it is idle (i.e., the DAP1216a procedures have not been started), running (i.e., powered but not collecting data) or recording (i.e., currently sampling data). Entering key 'C' allows the user to override the triggering algorithm and record any signals generated. Key 'D' is used to start or stop the DAP1216a input, output and
processing procedures defined for this application. When required, the user can copy the data in the event log file to a different directory by entering key 'E'. This operation temporarily suspends all communications, moves the files to a different directory (/miser/download) and then resumes communications. Key 'F' is used to drop the terminal connection temporarily so that the user can do other functions. The MISER program continues to execute in the background. To reconnect, the user must invoke the console routine MISCON again. Key 'S' allows the user to output a pulse or periodic waveform on the analog output pin A0. The option toggles between starting and stopping the waveform. Entering key 'Q' causes the program to quit and returns the user to the operating system prompt.

Figure 4.3 shows the options for the SETUP menu.

Figure 4.3: SETUP Menu

The letters indicate the keystroke used to invoke the particular function. The SETUP menu allows the user to read the default values from a file (key 'B'), or save the entered data in a file (key 'A'). From the SETUP menu, the user can nav-
igate to the output configuration menu (key 'C'). This menu offers choice between a wave output (key 'W') and a pulse output (key 'P'). Each of these menu offers the choice to load default values from a file (key 'A') or save the entered values in a file (key 'B'). Key 'M' used within any of these menus returns the user to the MAIN menu.

4.4.2 MISCON

The program MISCON implements and provides a mechanism for the user to initiate and control the functions of the DAP1216a. It sets up two pipes named MISERIN and MISEROUT as communication channels to and from the MISER module. MISCON allows users to reconnect at will to an active MISER process. This allows users to execute a MISER process in the background and reconnect to that process at a later time. MISCON has interrupt handling hooks that shut down the communication pipes and gracefully terminate active MISER processes (whether attached to the MISCON or executing in the background). MISCON also shuts down gracefully if it receives a QUIT signal from MISER.

A block diagram of the program logic is shown in Appendix A.2. When this program is first invoked, it sets up the pipes MISERIN and MISEROUT for data transfer between itself and the program MISER. Since most operations are achieved by entering character code options, the terminal is placed in the character mode. MISCON sends a carriage return to MISER that causes it to display the STATUS menu. Arrow keys used for navigation within the SETUP menu contain an escape sequence. Consequently, if the first character is an escape, an additional two characters are read, and the corresponding key code is generated and passed on to the MISER program. If there is no escape sequence, the entered key is directly passed on to MISER. Steps 5 to 8 of the program logic are continuously executed until the program detects a QUIT signal from MISER. On receiving the QUIT signal, MISCON restores the terminal to the line mode and exits.
4.4.3 Custom Command

The custom command MSEDR implements the selected triggering mode for micro-seismic event detection. This program executes on the DAP1216a and computes a running average on blocks of data calculated from the pre-trigger and post-trigger window lengths specified by the user. While the DAP1216a is collecting data and populating the input buffers via interrupts and DMA transfers, the MSEDR constantly computes the averages and checks for events. MSEDR can implement threshold or ratio mode triggering.

Events are recognized based on the trigger mode specified via the setup menu. In threshold triggering mode, data is transferred from a temporary buffer to a permanent buffer whenever the input signal values for at least the minimum number of channels specified in the setup menu exceed the threshold value. In the ratio mode, data values are transferred when the ratio (STA/LTA) exceeds the user specified value for at least the minimum number of channels specified.

Yet another mode is the forced trigger mode, wherein the user can simulate a trigger and force the program to dump data to the permanent buffer. This option is very useful for routine off-line analysis of data.

The block diagram for the program logic is shown in Appendix A.3. MSEDR is a custom command that is invoked by MISER. The configuration file defines all the configurations and then invokes the custom command with parameters. The passed parameter contains two variables—rmode and cflag—that are respectively manipulated to enable forced triggering of the DAP1216a and to temporarily pause the DAP1216a. The data entered in the SETUP menu is passed on to this program as a vector. The last parameter in the vector is a pipe that is used to send the output to the host computer. All the parameters are cast into appropriate types and their values are stored in local variables on the host computer. Depending on the duration of the pre- and post-trigger window lengths, buffer sizes for the raw data are calculated. The number of input blocks corresponding to the duration of short term average and long term average are calculated. Sampled data is stored
in two pipes named PGA and PGB. While samples stored in pipe PGA are taken at a gain of 1, samples in pipe PGB are taken at a gain of 10. The custom command opens these two pipes for reading and another pipe for writing. For the block read and write operations, the minimum and maximum number of data per block is then set. When collecting multiplexed data from more than one channel, at a fixed number of samples per channel, this value must be set equal to the product of the number of channels and the samples collected per channel. Storage is allocated for the work buffers and the output buffer. Variables are initialized and the processing loop begins. The different operations executed in sequence are:

1. Obtain two consecutive samples from each channel—one at a gain of 1 and the second at a gain of 10.

2. For each channel calculate the maximum and minimum values for the gain 10 data.

3. If the calculated minimum and maximum values exceed the converter range, use the gain 1 sample value.

4. Calculate the sum of the gain 1 or the gain 10 samples, whichever is used.

5. If the gain 1 samples are used, multiply the sample values by 10, so that all sample values are at the same gain.

6. For the first data block (i.e., 100 samples), set the LTA equal to mean of data.

7. Calculate the running average of a block of data corresponding to the specified short term average duration and the long term average duration, by replacing the earliest value with the new sample value. For the very first block of data, these running averages are set to the mean of the sample values.

8. Clear the trigger count and the trigger flag for each channel.

9. If threshold triggering is specified, an event is detected if the difference between the maximum or minimum value of a data block and the long term average exceeds the threshold.
10. If STA/LTA triggering is specified, an event is detected when the ratio of the short term average to the long term average exceeds a specified limit. When a trigger is detected, the trigger flag is set to 1 and the trigger count is incremented.

11. Steps 1 to 10 are repeated for all the channels used. At the end of one cycle through all channels, the trigger count will contain the total number of channels triggered.

12. For each channel, the trigger flag and gain values are merged and placed in a temporary work buffer.

13. Data transfer to the host can be initiated either when there is an actual trigger, recognized if the total number of triggers is greater than the minimum number specified, or forced data transfer is requested from the MAIN menu option. In the former case, the data from the earliest sample available in the work buffer is transferred to the host PC. In the second case, data from the current data block is transferred. The program repeatedly checks for these two types of triggers continuously.

14. When an event is detected, transfer data to the output buffer.

15. If the communication between DAP1216a and the host computer is enabled, transfer data from the output buffer to the host computer memory via the BIN(OUT) communication pipe. If communication has been temporarily stopped, when files are being transferred from one directory to another using the MAIN menu 'transfer files to download directory' option, wait until communication is enabled and then transfer data.

16. Repeat the steps listed above for the next input block.

4.4.4 Software Integration

MISER and MISCON are designed to run on a host computer. The custom command MSED/R executes on the DAP1216a. The host computer is a 133-MHz
Pentium computer operating under the Linux environment. Data acquisition and recording entails extensive transfer of data between programs. This is achieved by establishing communication channels between the programs and initialising them at startup. Under the Linux environment, pipe devices are set up to allow communication between local programs. The accel driver software allows communication with the custom command on the DAP1216a. Figure 4.4 shows the various pipes and direction of data flow.

![Diagram of communication channels between programs and pipes](image)

Figure 4.4: Interprogram Data Transfer

### 4.5 Microseismic Data Storage Standard - SEGY

In the early 1980s, the most common data storage format for microseismic data was SEGY. This is the Society of Exploration Geophysicists Y format which is described in the SEG's publication *Digital Tape Standards*. The format is still widely used today.

In its earliest form, the SEGY data format consists of three parts. The first part is a 3200-byte card image header which contains 40 cards (i.e. 40 lines of text with 80 characters per line) worth of text data describing the tape. The second part
is a 400-byte binary header containing information about the contents of the tape reel. The third part of the SEGY format consists of the actual seismic traces. Each trace has a 240-byte trace header. This is followed by the data written in the binary format.

For the SEGY files created in this application, the card image header contains information about the mine site (name and location). The binary header contains information such as number of traces in the file, sampling interval, data sample format, trace sorting code, gain and amplitude recovery method. The trace header contains the trace sequence number within the line, trace sequence number within the reel, field record number and trace number within field record, elevation, x-coordinate, y-coordinate, trace length in samples, frequency of the filter used, time, day of the year and year that the data is recorded.

4.6 System Setup and Initialization

The following actions must be executed before the system is properly initialized for on-site data acquisition:

- Load the accel device driver under Linux.
- Set up MISERIN and MISEROUT as Linux pipe devices.
- Create directories 'miser', 'miser/data' and 'miser/set' under the root directory.
- Compile all custom commands using the batch file supplied by Microstar Labs. This step generates an executable binary file that can only be executed by the DAP1216a.
- Load the binary file onto the DAP1216a via the COMLOAD program. COMLOAD is a program supplied with the Developer's Toolkit for DAPL and is used to load a custom command onto the DAP1216a on-board RAM.
• Compile MISER and MISCON source programs and copy the executables to a separate directory. Keeping the MISER, MISCON, and COMLOAD programs in a separate directory is advised for easy operation.

It is mandatory to execute MISER (in the background) and then run MISCON. For each specific situation, the user must configure the inputs using the SETUP menu. Depending on the type of triggering mechanism specified, events are recognized and pertinent data is transferred to the host computer where it is formatted to conform to SEGY standards. To avoid memory overflow the DAP1216a is programmed to automatically restart every 24 hours. Files in the data directory are moved to a 'download' directory every time the DAP1216a restarts. These files can then be copied to floppy diskettes as necessary.

4.7 Program Enhancements

Since the early 1990s Saskatchewan has witnessed tremendous growth in the number of software and hardware activities associated with microseismic monitoring. This research builds on a software program originally developed by Peter Kosteniuk, of Kosteniuk Consultants Limited. The software was enhanced to address the needs of a wideband, portable, data acquisition system in the following areas.

Exception Handling: The software should catch and gracefully handle exceptions such as user intervention to abort a program, incomplete initializations, invalid configuration, data acquisition error and memory overflows.

The DAP1216a communicates with the host via inter-process pipes. These pipes must be flushed completely at startup and shutdown; communication integrity cannot be ensured otherwise. Inconsistent pipe states corrupt the communication channel and result in loss and damage to configuration commands sent to and data received from the DAP1216a. Other times, users may want to asynchronously shut down the DAP1216a using operating system commands invoked via actions such as pressing some control keys on the keyboard. Such scenarios also require that the
communication pipes between the DAP1216a and the host be flushed before the system is halted.

User interface exceptions are addressed via signal handling routines provided by the Linux operating system. The function 'mySignalHandler' developed for this project traps such exceptions (control keys), flushes and closes all communication pipes and returns control to the operating system.

**High Frequency Samples:** The input procedures that sample the input can be programmed to sample at different sampling frequencies. By default, the sampling frequency is set to 250 samples/second. To meet the requirements of a wideband data acquisition system, the input procedure was set up to default to a sampling rate of 2500 samples/second. In addition to this, the available options in the DAP1216a configuration file were enhanced to allow users to set a sampling frequency as high as 2500 samples/second.

**Self-Calibration:** A portable system should include features for self-test and self-calibration. For a data acquisition system this implies, at the minimum, the ability to produce electrical waveforms that simulate the output of transducers. In addition, there should be some flexibility in the choice of frequency and envelope of the waveform.

These requirements are satisfied by the waveform generator (continuous and pulsed) developed in this project. The generated outputs are available at the analog output pin A0 on the DAP1216a. Programming interfaces have been developed and incorporated into the MISER module to allow remote access via the following activities at DAP1216a system initialization time.

1. Start and stop waveform generation.
2. Select from sinusoidal, sawtooth, triangular, square waveforms.
3. Specify amplitude and time-period of continuous waveforms.
4. Specify amplitude and duration of pulsed waveforms.

4.8 Summary

Operation of the portable, wideband, microseismic data acquisition system is centred around the use of modular software each achieving a specific purpose. MISCON is the terminal handling program that runs in the foreground. This allows the user to detach from the terminal and perform other functions and reconnect at a later time. MISER running in the background handles the menu displays and formatting of event data to conform to industry standards. Event detection and data transfer to a host computer is achieved by the custom command MSED. Section 4.6 lists the sequence of operations to be followed in installing the software and setting up the system for data acquisition. The portable hardware complemented by the software discussed here can be easily installed and operated.

A typical laboratory setup for simulating vibrations and testing this system is described in Chapter 5.
Chapter 5

Methodology and Results

The experimentation in this research was aimed at evaluating the performance of two distinct aspects of the data acquisition system, namely, the piezoelectric transducer and the data analysis system. The data analysis system includes an analysis algorithm (software) and a low-pass, anti-aliasing, Bessel filter (hardware) that was developed specifically for this research. Henceforth, the developed piezoelectric transducer shall be referred to as the test accelerometer.

Two sets of experiments were performed. One set helped to characterize and evaluate the test accelerometer in detecting and recording microseismic events while the other helped to evaluate the performance of the data acquisition system in acquiring data from a transducer. The first set was performed by observing the data acquisition system’s response over a range of frequencies when equipped with the test accelerometer. The second was performed by observing the data acquisition system’s response over a range of frequencies when equipped with a geophone. Unlike the widely used geophone—a transducer whose electrical output is a reasonably faithful indication of the input velocity—a piezo-electric transducer (in our case the test accelerometer) generates an electrical output indicative of the acceleration it is subjected to.

The set of data obtained from our experimentation was sub-divided into two overlapping subsets. One subset was used to characterize and evaluate the performance
of the test accelerometer while the other was used to evaluate the performance of the hardware and the software implementations. Conclusions and interpretations of the data are presented later in the chapter.

Sinusoidal input of known frequency and amplitude were input to the vibration exciter to generate controlled microseismic events. Over a wide frequency range, such controlled events were used to simulate and test the data acquisition system. Observing the system's response provides sufficient data to verify design functionality. Comparing the response to a known correct reference (also generated under controlled conditions) measures the design accuracy.

5.1 Testing Requirements

The data analysis algorithm relies on being able to detect and distinguish sudden changes in surface movements (such as those precipitated by imminent microearthquakes) from sedentary movements of the surface (such as those generated by usual mining activities). In other words, it depends on being able to measure and quantify the acceleration of the geological surface to which the transducers are attached.

The experiment uses a calibrated piezoelectric accelerometer, henceforth referred to as the reference accelerometer, as a known reference. Accurate characterization of acceleration measuring devices is critical to the success of the software. The input excitation signal to the vibration exciter controls the amplitude and the frequency of vibrations of the table. The test accelerometer and the reference accelerometer measure acceleration—a value that is directly proportional to the product of the amplitude and the frequency. Maintaining a constant acceleration while scanning a range of frequencies during the experiment eliminates a free variable, thereby letting us better focus on evaluating the frequency characteristics of our analysis algorithm. Constant acceleration is maintained by manipulating the amplitude of the vibration exciter vibration so as to maintain a constant reading from the reference accelerometer.
5.2 Test Setup

The experimental setup is illustrated in Figure 5.1. It consisted of a portable recording device, a vibration exciter, a reference accelerometer, a test accelerometer, multimeters and high-impedance amplifiers. The portable recording device was based on a personal computer architecture, with the data acquisition board (Microstar's DAP1216a) occupying one expansion slot and the anti-aliasing, Bessel filter occupying another. The entire recording device is mounted inside an aluminium case for rapid transportation and deployment.

![Figure 5.1: Test Setup](image)

The signal generator was the primary source for variable-frequency sinusoidal signals. Its output was conditioned by a power amplifier (B&K\(^1\)2706) and fed as the excitation input to the vibration exciter (B&K4809).

A connector board connects the output from the test accelerometer (or the geo-

\(^1\)Brüel and Kjær
phone) to the anti-aliasing filter. The output of the filter is fed as input to the data acquisition board. The output of the test accelerometer is simultaneously fed to a voltmeter to record voltage levels during the experimentation. The output of the reference accelerometer is amplified by a charge amplifier and fed to a voltmeter that records the voltage output proportional to the charge generated by the reference accelerometer. Offline, the reference accelerometer is mounted on the calibrator (B&K4294) and its output fed through the charge amplifier to a voltmeter and the sensitivity of the charge amplifier adjusted to produce a voltage of \(10 \text{ V}_{\text{rms}}\) for a \(10 \text{ m/s}^2\) acceleration produced by the calibrator. The Linux operating system, the accel drivers for Linux, the terminal handling routine and the data acquisition program were all loaded onto the computer. The custom command was downloaded to the DAP1216a on-board memory.

5.2.1 Instrumentation and Measurement

Reference Accelerometer: The reference accelerometer is a B&K accelerometer (type 4370), specially designed to measure light industrial vibrations. Type 4370 has a unified charge sensitivity of \(10\ \text{pC/m/s}^2\) with useful operating frequency range up to \(6\ \text{kHz}\). A delta-shear configuration in the accelerometer results in a high resonant frequency and low sensitivity to temperature transients and base strain. This configuration employs three piezoelectric elements arranged in the shear mode around a triangular centre post. The operating temperature range is up to \(250^\circ\text{C}\).

Charge Amplifier: The output of the piezoelectric accelerometer is a low electrical charge insufficient to drive common instrumentation devices. Loading effects manifest as low sensitivity and smaller operating frequency range. A high impedance charge amplifier (B&K Model 504) is a solution to this problem. It is a variable gain, multiple range, charge amplifier designed to isolate and amplify piezoelectric transducer outputs. The variable gain feature, combined with a maximum full scale reading up to \(\pm 10\ \text{V}\) over twelve ranges, make it ideal for conditioning the output from the reference accelerometer.
Power Amplifier: The B&K Power Amplifier Type 2706 is designed to drive small vibration exciters such as the B&K Type 4809. Its maximum gain is 40 dB; the output drive is 75 VA into a 3-Ω exciter; and the usable frequency ranges from 10 Hz to 20 kHz. It has very low harmonic distortion and high tolerance to temperature and voltage fluctuations. Inputs can be attenuated in steps of 10 dB, clipped or amplified via a continuous gain control.

Vibration Exciter: Vibration transducers of any type can be calibrated on the B&K Vibration Exciter Type 4809, and their frequency response can be determined in the frequency range 10 Hz to 20 kHz. Type 4809 is an electrodynamic type with a permanent field magnet. A coil, which is an integral part of the table structure, is flexibly suspended in one plane in the field of the permanent magnet. An AC signal, provided by an external oscillator, is passed through the coil to induce vibrations in the table. The moving element is supported by a robust rectilinear guidance system consisting of grouped radial and transverse flexures. Laminated flexure springs provide high damping to minimize distortion due to axial, transverse and flexural resonances. The bare table has an extremely high first resonance frequency at 20 kHz. The instrument can withstand maximum peak-to-peak displacement of 8 mm. A mounting arrangement was devised so that both the reference accelerometer and the test transducer can be simultaneously mounted on the vibration exciter thus making it possible to obtain the frequency response of the test transducer for a constant acceleration.

Calibration Exciter: The B&K Calibration Exciter Type 4294 is a compact, portable vibration reference source that calibrates vibration measuring and recording systems based on the piezoelectric accelerometers. The calibrator uses an electromagnetic exciter driven by a stabilized oscillator at a frequency of 1000 rad/s. Servo feedback via an accelerometer on the underside of the vibration table maintains a constant vibration level of 10 m/s². The reference accelerometer was mounted on the table with bees-wax or cyanoacrylate adhesive, and its output was connected to a measuring instrument via the charge amplifier. With the calibrator excited,
the sensitivity of the charge amplifier was adjusted to indicate an output of 1 V rms. This calibrates our measurement system to indicate 1 V for 10 m/s² acceleration. Besides being simple, such a calibration system compensates for any inaccuracies introduced by the cabling system or the transducers themselves.

5.3 Test Procedure

Frequency Response: The experimental setup for testing the frequency response of test transducers using a reference accelerometer was as shown in Figure 5.1. In this research two different transducers—a geophone and the test accelerometer, were used as the test transducer. The experiment was conducted as follows:

1. Powered up the host computer, downloaded the custom command (MSEDR) to the DAP1216a and executed the event recording program (MISER) in the background.

2. Started the user interface program (MISCON). The STATUS menu appears on the computer screen.

3. Connected the equipment as shown in Figure 5.1. Calibrated the measurement system offline as indicated in the previous section.

4. Configured input channel 1 as geophone and the trigger type as none. Specified pre-trigger window of 0.5 seconds and post-trigger window length of 6 seconds. Saved the configuration. The A/D is automatically reprogrammed after each save operation.

5. The following steps were performed to obtain the frequency response of the geophone:

(a) Mounted the reference accelerometer and the geophone on the vibration exciter. This ensured that the reference accelerometer and the geophone are subjected to identical inputs.
(b) Connected the output of the geophone to Differential Input D0 on the input connector board.

(c) Zeroed the frequency and amplitude controls in the signal generator. Powered it on, set the mode to sine wave and the frequency to 10 Hz. Adjusted the amplitude to obtain the desired acceleration value. A reading of 0.05 V indicated an acceleration of 0.5 m/s².

(d) From the MAIN menu on the computer, selected option B: Record Background Signals. Switched back to the STATUS menu and noted the peak value and the event number. The peak value is the maximum value of sinusoidal output from the geophone.

(e) Repeated the previous step for a different sine wave frequency. Adjusted the amplitude to maintain a constant voltmeter reading (thereby ensuring a constant acceleration).

6. To obtain the frequency response of the test accelerometer,

(a) Removed the geophone and mounted the test accelerometer.

(b) Connected the output of the test accelerometer to Differential Input D0 on the input connector board.

(c) Configured the detector type as accelerometer from the SETUP menu.

(d) Repeated the entire experiment over a range of frequencies and recorded the output values.

One set of frequency response data (with a constant acceleration of 0.5 m/s²) was obtained for the geophone while two sets of frequency response data (with constant acceleration of 0.5 m/s² and 0.25 m/s²) were obtained for the test accelerometer.

**Software Verification:** Another experiment was conducted as follows to verify that the software accurately measures the input. The experiment was conducted twice, once with an input voltage of 233.1 mV_{rms} and once with an input voltage of 176.6 mV_{rms}. 

68
1. Connected the output of the signal generator to input D0 on the connector board.

2. Adjusted the controls on the signal generator to generate a 10 Hz sine wave with a fixed amplitude.

3. From the MAIN menu on the computer, selected option B: Record Background Signals. Switched back to the STATUS menu and noted the peak value and the event number. This peak value was a measure of the output of the signal generator.

4. Changed the frequency of the sine wave without changing its amplitude and repeated the previous step.

5. Repeated steps 2 and 3 with gradually increasing frequency up to a frequency of 2400 Hz.

5.4 Simulation Results and Discussion

The experimental set up was used to obtain sample data from a geophone and the test accelerometer. The outputs from the test accelerometer and the geophone were observed at constant acceleration.

Test accelerometer response: Figure 5.2 is a plot of the peak values over a frequency range as observed during this experimentation. The solid lines in the graph denote the value obtained from the data recorded by the developed software. The "*" and "." denote the values obtained from the digital multimeter. The output from the test transducer remains fairly constant at lower frequencies until about 100 Hz, followed by a sudden increase and then a sudden decrease to a gradually diminishing level. The maximum output is at 263 Hz, the resonant frequency of the test transducer.

Recall from Section 2.1 that such an an envelope is characteristic of transducers that measure input acceleration. Consequently the test transducer can be used as an accelerometer up to about 100 Hz. The value of 100 Hz was determined by
observing the range over which the graph in Figure 5.2 approximates a line parallel to the x-axis (frequency).

It is also evident from Figure 5.2 that the value obtained from the developed software closely matches the value obtained from an indicating instrument. This demonstrates that the software gives a fairly accurate value of the transducer output, thus validating the use of the software for testing transducers.

**Geophone response:** The geophone is a velocity transducer and generates output proportional to its input velocity. Its sensitivity is the ratio of the observed output to its input velocity. If input velocity is maintained constant during the experiment, sensitivity of the geophone is a constant multiple of its electrical output. The values obtained from this experiment are at constant input acceleration and are plotted in Figure 5.3. If the input \( u \) is sinusoidal, with peak value \( u_0 \), then the velocity and acceleration are found by successive differentiation. The peak value of the velocity
Figure 5.3: Geophone Output at Constant Acceleration

$u''_0$, and that of the acceleration $u''_0$, are related as

$$u''_0 = \omega u'_0 \tag{5.1}$$

where $\omega$ is the input frequency in radians. Substituting Equation 5.1 in Equation B.9 leads to the relation

$$\frac{s_0}{u'_0} = \omega \times \frac{s_0}{u''_0} \tag{5.2}$$

The response to a constant velocity input can, therefore, be derived from the response to constant acceleration, by multiplying the observed data (at a given frequency) with the corresponding frequency. The calculated values are plotted against corresponding input frequency in Figure 5.4.

From the Figure 5.3, it can be seen, that the geophone has a gradually increasing output in the lower frequency range. The output amplitude is maximum at 50 Hz and decreases thereafter. This is the natural frequency of the geophone and agrees with its stated natural frequency. The response of the geophone to a constant velocity input is shown in Figure 5.4. It is seen that the output amplitude initially increases but later assumes a fairly constant value between 70 Hz and 250 Hz. The
graph in Figure 5.4 is similar to the expected graph for a velocity transducer as shown in Figure 2.2. By comparison, the operating frequency range of the geophone lies between 70 Hz and 250 Hz. This is the operating frequency range over which the geophone is currently being used in practical microseismic networks.

The obtained values of the natural frequency for the geophone and the test accelerometer agrees with the requirement that an accelerometer should have a high natural frequency while a velocity transducer should have a low natural frequency.

Software Performance: The software was validated by observing its output when provided with constant amplitude inputs. Graphs A and B in Figure 5.5 represent the response of the data acquisition system for sinusoidal input signals. Graph A was generated with a measured input voltage of 233.1 mVrms and graph B with 176.6 mVrms. The x-axis represents the frequency of the sinusoidal input while the y-axis represents the output indicated by the data acquisition system. The solid lines represent the peak voltage detected by the software while the *'s denote the peak voltage measured at the input to the DAP1216a.

For purposes of analytic accuracy, the input to the DAP1216a was calculated by
measuring the peak amplitude at the output of the Bessel filter and using Table 3.1 to evaluate the actual input. From the graphs it is evident that the observed values from the software (V_{sw}) and the expected values i.e., the output of the Bessel filter (V_{out}), track closely over the operating frequency range of the data acquisition system.

Table 5.1 is a tabular representation of the data from Figure 5.5. An extra column indicates the percentage difference in the expected and observed values. The numbers in this column are calculated as percentages, using Equation 5.3.

\[ P = \left( \frac{V_{out} - V_{sw}}{V_{out}} \right) \times 100 \]  

Consider a sinusoidal signal with an RMS voltage of 233.1 mV_{rms} and a peak amplitude of 329.65 mV (ie. 233.1\sqrt{2}). At 1000 Hz the filter has an attenuation of -0.11 dB (Table 3.1). If the input to the filter is 329.65 mV, the output is 325.36 mV. The value recorded by the data acquisition software was 323.6 mV resulting in an error of 0.58% (from Equation 5.3). The possible sources of the differences between the
<table>
<thead>
<tr>
<th>Frequency (Hz)</th>
<th>Error (%)</th>
<th>Frequency (Hz)</th>
<th>Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Plot A</td>
<td>Plot B</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1.03</td>
<td>0.57</td>
<td>1100</td>
</tr>
<tr>
<td>100</td>
<td>1.06</td>
<td>0.61</td>
<td>1200</td>
</tr>
<tr>
<td>200</td>
<td>1.06</td>
<td>0.61</td>
<td>1300</td>
</tr>
<tr>
<td>300</td>
<td>1.08</td>
<td>0.59</td>
<td>1600</td>
</tr>
<tr>
<td>400</td>
<td>0.94</td>
<td>0.49</td>
<td>1800</td>
</tr>
<tr>
<td>500</td>
<td>0.98</td>
<td>0.45</td>
<td>2000</td>
</tr>
<tr>
<td>600</td>
<td>0.88</td>
<td>0.35</td>
<td>2100</td>
</tr>
<tr>
<td>700</td>
<td>0.66</td>
<td>0.43</td>
<td>2200</td>
</tr>
<tr>
<td>800</td>
<td>0.72</td>
<td>0.19</td>
<td>2300</td>
</tr>
<tr>
<td>900</td>
<td>0.68</td>
<td>0.14</td>
<td>2400</td>
</tr>
<tr>
<td>1000</td>
<td>0.58</td>
<td>0.08</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.1: Error Percentages for Data Points in Figure 5.5

observed and the expected values are:

1. The Instek Model FG-8016G digital function generator has frequency accuracy of ±5% of the full scale.

2. The multimeters have a minimum resolution of 1 mV while the oscilloscope has a voltage resolution of ±1.5% of the reading ±0.4% of full scale.

3. DAP1216a uses a 16-bit converter over a full scale range of ±5 V and has non-linearity error of ±2 LSB. This signifies an error of ±300 µV. The transducers generate signals in the millivolts range. The sampling configuration specifies that signals below 458 mV are sampled at a gain of 10. Given this, significant percentage error contributed due to quantization is \((0.3 \times 100/10 = 3)\).
5.5 **Summary**

A series of tests were conducted to characterize and verify the operation of the test accelerometer and the data analysis/recording software and algorithm. Test results were compared with expected results and demonstrated a good match. Sources of minor mismatches were identified and quantified.
Chapter 6

Summary and Conclusions

This research was motivated by a need to test an accelerometer developed for use in microseismic monitoring. Depending on the construction and principle of operation, it is possible to develop different types of transducers for use in a microseismic network. Prior to actual deployment in a network, it is necessary to test these new transducers and study their characteristics. Permanent microseismic monitoring networks employ a large number of transducers placed in a known configuration. It is not practical to disrupt operations of these networks for testing new transducers. From this stemmed the need to have a stand-alone system that would have the necessary features of a microseismic data acquisition system, but could be easily transported, configured and operated for acquiring data from test transducers. The intent of this research was to develop a generic system that could acquire data from different transducers and organize the data in an industry standard format which could be analyzed to study the transducer behaviour in a simulated environment.

A portable, wideband, microseismic data acquisition system was developed during the course of this research. A commercially available, single-board, data acquisition processor (Microstar's DAP1216a) forms the backbone of the data acquisition system. The DAP1216a can monitor and acquire data from as many as 8 input channels at sampling rate of 2500 kilosamples per second per channel.

Wide bandwidth operation was achieved by implementing an eighth-order Bessel
filter. MATLAB programs and SPICE circuits were used to design and simulate this filter. The filter was implemented on a single PC-compatible board. This filter exhibits a roll-off of 160 dB/decade and has a high cut-off frequency at 2485 Hz.

An in-house experimental setup was used to test the enhanced software and obtain test data from multiple transducers. Tests were conducted to verify the accuracy of data obtained from this system. This involved using a known input signal and verifying that the signal can be reconstructed from the sampled data acquired by the system. The developed system is able to faithfully reproduce the signals generated by a transducer within reasonable tolerance. A second test was undertaken to study the behaviour of the piezoelectric accelerometer and a geophone. The piezoelectric and geophone transducers were subjected to constant acceleration and their outputs recorded by the developed system. Simultaneously, the transducer output data was also recorded via a measuring instrument. The two sets of data were found to match within reasonable tolerances, thereby establishing the design accuracy. The data obtained from the second test was also used to analyze and explain the response of the geophone and the test accelerometer.

In conducting this research we were able to demonstrate that the developed system:

1. Is portable, easy to set-up, and can be accessed, configured and programmed remotely.

2. Offers a wide bandwidth.

3. Gathers and stores data in the standard SEGY format. Data can subsequently be offloaded to removable storage media and analyzed using a suitable analysis package.

4. Provides multiple triggering algorithms that extend the useful domain of applicability. This was tested by generating a simulated event to trigger the STA/LTA algorithm and validating that the recorded data did indeed provide a good indication of the occurrence of the event.

5. Enables recording and reproduction of transducer outputs. The data in the SEGY files can be reconstructed and visualized as waveforms.
6. Has a simple, user-friendly interface. The screens and options are self-explanatory and easy to navigate.

In conclusion, the developed system achieves the design goals set for this research. It is a fully qualified system capable of being used as a stand-alone test tool to evaluate sensors used in microseismic monitoring.

6.0.1 Future Work

The work done in this research is by no means a complete and exhaustive exploration of the various tasks and opportunities available for portable data acquisition systems. Future work can improve the system in a few different ways, some of which are:

1. Verifying the performance of a large set of existing and newer transducers. This can be done by installing the system at a mine site and acquiring data for a single event from different test transducers, comparing it with a known reference data (generated by a reference transducer that has been set up in the same site) and comparing the results.

2. Experimenting and observing the behavior of the system while varying different aspects of the triggering algorithm. The different aspects could include the STA/LTA trigger ratio, the pre- and post-event data gathering window, the transducer sensitivity or the sampling rate.

3. Using the developed system to study the effects of mine geometry and distance limitations on signals transmitted by different transducers.
REFERENCES


Appendix A

Program Block Diagrams
A.1 MISER-Block Diagram

Figure A.1: MISER - Program Block Diagram
MISER block diagram (Contd)

Figure A.2: MISER - Program Block Diagram
MISER block diagram (Contd)

Figure A.3: MISER - Function GetADa
A.2 MISCON - Block Diagram

Figure A.4: Terminal Handling Routine - MISCON
A.3 MSEDRA - Block Diagram

Figure A.5: MSEDRA - Program Block Diagram I
Compute total absolute deviation from the mean

For the first data block, set LTA = mean of data

For subsequent data blocks, compute sum and average by replacing tail values by current value

For first LTA duration blocks, long term sum and average without replacing tail values

For first long term deviation data blocks, compute long term sum and average without replacing tail values

For subsequent data blocks, compute sum and average by replacing tail values by current value

For first short term deviation blocks, compute long term sum and average without replacing tail values

Clear total trigger count and trigger flag for each channel

If threshold triggering specified, set trigger if difference between minimum or maximum value of data block and long term average exceeds the threshold

Figure A.6: MSED - Program Block Diagram II
MSED - block diagram (Contd)

For STA/LTA triggering, set trigger if ratio of STA to LTA exceeds the threshold

Repeat for each channel

Merge trigger flag and gain
If waiting for event to occur and there is room in output buffer and work buffer is full, continue
Else jump to step 35

Check for trigger. If trigger detected, transfer data starting with earliest block in work buffer
Else go to step 34

If DAP procedures not PAUSED, transfer data from work buffer to host computer
Else wait for DAP to RESUME and then transfer data to host

Increment block counter repeat from step 9 to 35

Note: This program executes until the user exits from MISER program.

Figure A.7: MSED - Program Block Diagram III
Appendix B

Mass-spring Transducer Response
B.1 Mass-Spring Transducer

This appendix describes the operation of a typical mass-spring system and the explains the related equations used in expressing its response. A typical mass spring system is show in Figure B.1

![Figure B.1: Mass-Spring Type of Vibration Measuring Instrument](image)

It consists of a mass $M$ suspended from the transducer case $a$ by a spring of stiffness $K$. The motion of the mass within the case may be damped by a viscous liquid, symbolized by a damping coefficient $C$. When the transducer case is attached to the moving part, the transducer may be used to measure displacement, velocity or acceleration based on whether the relative displacement or relative velocity is sensed by the transducing element. Analysis of data obtained from typical mass-spring systems is documented in [27].

Consider a transducer whose case experiences a displacement motion "$u$". While the case moves, because of inertia the mass tends to lag behind in motion and thus there is some relative displacement between the mass and the case. Let the relative displacement between the mass and the case be "$s$". The displacement of the mass with respect to a fixed reference in space is $s - u$ and the applied force is $M[d^2(s - u)/dt^2]$. The force applied by the spring is $-ks$, and the force applied by the damper is $-C(ds/dt)$. Since the system is in equilibrium we obtain

$$-M \frac{d^2(s - u)}{dt^2} - C \frac{ds}{dt} - Ks = 0 \quad (B.1)$$

This equation may be rearranged to obtain:

$$M \frac{d^2s}{dt^2} + C \frac{ds}{dt} + Ks = M \frac{d^2u}{dt^2} \quad (B.2)$$
This is a linear differential equation. If the motion \( u \) is sinusoidal, then the response \( s \) will also be a sinusoid. Let this response be defined as \( s = s_0 \cos(\omega t - \theta) \). The solution of Equation B.2 is

\[
\frac{s_0}{u_0} = \frac{\omega^2}{\sqrt{(\frac{K}{M} - \omega^2)^2 + (\frac{\omega C}{M})^2}} \tag{B.3}
\]

where \( s_0 \) is the maximum relative displacement amplitude. For undamped systems \((C = 0)\) the resonant frequency \( f_n \) is the frequency at which

\[
\frac{s_0}{u_0} = \infty \tag{B.4}
\]

Setting \( C = 0 \) in Equation B.3 results in:

\[
\omega_n = 2\pi f_n = \sqrt{\frac{K}{M}} \text{ rad/sec} \tag{B.5}
\]

The damping in a transducer is specified as a fraction of the critical damping. The critical damping \( c_c \) is the minimum level of damping that prevents a mass-spring transducer from oscillating when excited by a step function or other transient. It is defined by

\[
c_c = 2\sqrt{km} \tag{B.6}
\]

Thus, the fraction of critical damping, \( \zeta \), is

\[
\zeta = \frac{c}{c_c} = \frac{c}{2\sqrt{km}} \tag{B.7}
\]

The excitation frequency, \( \omega \), for a transducer is defined in terms of the undamped natural frequency, \( \omega_n \), by using the dimensionless frequency ratio \( \omega/\omega_n \). Substituting this ratio and the damping factor, Equation B.3 can be rewritten as

\[
\frac{s_0}{u_0} = \frac{\left(\frac{\omega}{\omega_n}\right)^2}{\sqrt{\left[1 - \left(\frac{\omega}{\omega_n}\right)^2\right]^2 + (2\zeta \frac{\omega}{\omega_n})^2}} \tag{B.8}
\]

The graphical representation of Equation B.8 is shown in Figure 2.2. The different curves are obtained using different values for the damping factor, indicated alongside each curve.
The response of the mass-spring transducer given by Equation B.8 may be expressed in terms of the acceleration $u''_0$ of the moving part by substituting $u''_0 = -u_0 \omega^2$. Then the ratio of the relative displacement amplitude $s_0$, between the mass $m$ and the transducer case $a$, to the impressed acceleration amplitude $u'_0$ is

$$\frac{s_0}{u'_0} = -\frac{1}{\omega_n^2} \frac{\left(\frac{\omega}{\omega_n}\right)^2}{\sqrt{1 - \left(\frac{\omega}{\omega_n}\right)^2}^2 + (2\zeta \frac{\omega}{\omega_n})^2}$$  \hfill (B.9)

The graphical representation of Equation B.9 is shown in Figure 2.3. The different curves are obtained using different values for the damping factor, indicated alongside each curve.
Appendix C

Bessel Filter
C.1 Bessel Transfer Functions

Consider a signal $v_1$ that has been delayed by $D$ seconds, as shown in Figure C.1.

This can be mathematically represented as

$$v_2(t) = v_1(t - D) \quad (C.1)$$

If the input is a sinusoidal signal $v_1 = A\sin(\omega t + \phi)$ the output is represented by:

$$v_2 = A\sin(\omega(t - D) + \phi) \quad (C.2)$$

$$\Rightarrow A\sin(\omega t - \omega D + \phi) \quad (C.3)$$

The input and output signal differ only by the phase angle:

$$\theta = -\omega D \quad (C.4)$$

Differentiating Equation C.4 with respect to the frequency, we get

$$D = -\frac{d\theta}{d\omega} \quad (C.5)$$

A filter with the phase response given by Equation C.5 has a constant time delay and a linear phase shift over a frequency range. A plot of $\omega$ vs $\theta$ is a straight line with slope $-D$ where $D$ is commonly referred as envelope or group delay. If the phase is linear with a negative slope, the magnitude is constant and then the delay will be constant. Under such conditions a signal will be delayed without distortion.

A simple method to find the family of transfer functions that would give approximately constant delay was devised by Storch[28]. He used the hyperbolic function identity to obtain a continued fraction which could be simply truncated after $N$ terms. Simplifying the resulting fraction and adding the numerator and denominator generates the denominator for the transfer function for the constant time delay.
delay filter. The polynomials thus obtained resemble Bessel polynomials, hence their name. Bessel polynomials of any order can be generated by using the recursive formula

\[ B_n = (2n - 1)B_{n-1} + s^2B_{n-2} \quad (C.6) \]

and the transfer function is given as

\[ T_n(s) = \frac{B_n(0)}{B_n(s)} \quad (C.7) \]

The roots of \( B_n(s) = 0 \) are determined heuristically by iterative computer based algorithms. Eighth order Bessel filters are designed using the roots of the eighth order Bessel polynomial and working backwards to find the resistance and capacitance value for each stage of the filter.

### C.2 Component Values and Scaling

The equivalent passive network with active sources is shown in Figure C.2. \( V_{in} \) and

![Figure C.2: Equivalent Passive Network With Active Source](image)

\( V_{out} \) refer to the input and output voltages. \( V_a \) in Figure C.2 refers to the voltage at the non-inverting input terminal of the operational amplifier, while \( K \) denotes the amplifier gain. \( R_1 \) and \( R_2 \) denote resistances while \( C_1 \) and \( C_2 \) denote capacitances.

The transfer function for this circuit, which is a ratio of the output to the input, is found to be:
The standard transform for any second order filter with gain $K$ and cut-off frequency $\omega_c$ is given by:

\[
\frac{V_{in}}{V_{out}} = \frac{K/R_1C_1R_2C_2}{s^2 + \left[\frac{1}{R_2C_1} + \frac{1}{R_1C_1} + (1 - K)\frac{1}{R_2C_2}\right]s + 1/R_1C_1R_2C_2} \tag{C.8}
\]

For a normalized, unity gain filter, cut-off frequency $\omega$ is 1 rad/s and gain $K$ is 1. Using these values, Equation C.9 can be rewritten as

\[
\frac{V_{out}}{V_{in}} = \frac{K\omega^2}{s^2 + d\omega^2s + \omega_c^2} \tag{C.9}
\]

\[
\frac{V_{out}}{V_{in}} = \frac{1}{s^2 + ds + 1} \tag{C.10}
\]

Comparing Equation C.10 and Equation C.8, yields

\[
R_1R_2C_1C_2 = 1 \tag{C.11}
\]

\[
\frac{1}{R_2C_1} + \frac{1}{R_1C_1} + (1 - K)\frac{1}{R_2C_2} = d \tag{C.12}
\]

or

\[
\frac{2}{C_1} = d \tag{C.13}
\]

For ease of design, it is assumed that $R_1 = R_2 = 1$. Substituting the damping, $d$, by its more frequently used reciprocal $Q$, the capacitance values for the various stages can be calculated using Equation C.14, if the $Q$ values are known.

\[
C_1 = 2Q \tag{C.14}
\]

\[
C_2 = \frac{1}{2Q} \tag{C.15}
\]

These equations give the value of the filter components for a unity gain filter with cut-off frequency of 1 rad/s and resistance values of 1 $\Omega$. For best performance, the input impedance of an operational amplifier must be 10 k$\Omega$ or greater. Also, capacitance values obtained from these equations are not practically available. To design a practically realisable filter, it is necessary to scale the resistance and capacitance values. To scale from 1 $\Omega$, 1 rad/s to $K_m\Omega$, f Hz, resistances must be multiplied by $K_m$ and capacitances divided by $K_c = 2\pi f K_m f$. 

97
C.3 Eighth Order Bessel Filter Design Procedure

To design an nth-order Bessel filter, the starting point is to find the roots of the nth-order Bessel polynomial and work from there to evaluate the resistance and capacitance for each second order stage. In this research, roots of the eighth-order Bessel polynomial listed in [21] were used.

A second-order filter employing a non-inverting amplifier in the voltage follower configuration is shown in Figure 3.1. With this configuration, for a unity gain filter with a cut-off frequency of 1 rad/s and resistors of 1 Ω, the capacitances can be found using the relation:

\[
C_1 = 2Q \quad \text{(C.16)}
\]
\[
C_2 = \frac{1}{2Q} \quad \text{(C.17)}
\]

The theoretical resistor and capacitor values can then be suitably scaled to obtain practical values for these components. Four such second-order filters are cascaded to achieve the eighth order Bessel filter.

The strategy adopted to design the eighth-order filter is enumerated below.

- Note the value of Q for each stage from a standard table. For this research the values were taken from the tables in [21].
- Calculate the values of the capacitances (in microfarads) for each stage in the second order normalized filter using Equations C.16 and C.17.
- Scale the resistance and capacitance values to obtain the desired cut-off frequency.
- Round off the capacitance values were rounded to the nearest practical values. Recalculate the resistance values using the practical values of capacitors and round off to nearest practical values.
- Plot and observe the frequency response of the individual and composite filters. As practical values and theoretical values of the components vary, the response will not be the same as observed with theoretical values of components.
- Depending on the roll-off and the peak near the cut-off frequency for the composite filter, adjust the Q values to reduce the peak while still achieving a good roll-off. This may be done by adjusting the ratio of the two capacitances while maintaining their product constant at 4Q^2.
- Adjust the absolute value of the resistances and capacitances to get the desired cut-off frequency. Ensure that ratio of the capacitances remains constant, though their absolute values are changed.
• Use the component values that gave the best roll-off rate, with no peaks in the amplitude response and cut-off frequency closest to the desired cut-off frequency to simulate the filter circuit using a simulation package such as ICAP4/SPICE