National Library of Canada

Acquisitions and Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale du Canada

Direction des acquisitions et des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Canada

Biological Signal Data Acquisition and Processing

Attila Horvath

A Thesis
in
The Department
of
Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Applied Science at
Concordia University
Montreal, Quebec, Canada

February 1992

Copyright Attila Horvath, 1992

National Library of Canada
Acquisitions and Bibliographic Services Branch
395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale du Canada
Direction des acquisitions et des services bibliographiques
395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-80959-0

Canada

ABSTRACT

Biosignal Data Acquisition and Processing

Attila Horvath

Most clinical biosignal systems still consist of analog systems that acquire and process EMG signals and display them on a chart recorder. By using microcontrollers and Personal Computers much more functionality can be realized. The storage and display flexibility allows for easier analysis and interpretation. Eliminating many of the analog processing elements and replacing them by Digital Signal Processing (DSP) software gives the system adaptability to change and allows the user to discover new processing techniques. Two systems were designed, one remote system and one PC based system. The remote system is a portable battery powered microcontroller based unit that gives the subject maximum freedom. The remote unit is compact and contains sufficient RAM to allow stand-alone acquisition of data. The PC then can communicate through the serial port to the remote unit to download data and upload commands. FIR filters and IIR filters are available for filtering. Rectification and envelope detection routines allow for traditional EMG processing. Integration is available for EMG signal quantification. During signal acquisition the signal spectrum can be displayed and the signal can be filtered in real time.

Acknowledgements

I would like to thank my supervisors Dr. Gouw and Dr. Comeau for their help, encouragement, and patience. I wish to acknowledge the valuable input from Ron Rehel. Finally. I owe a lot to my parents for their support.

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

The human body produces many signals that can be used to analyze a condition or function. Many of these signals are of electrical nature. Most clinical systems still consist of analog systems that acquire and process signals and display them on a chart recorder (Fig 1.1). This thesis concerns the design, development and implementation of a Bio-signal Data Acquisition and analysis System (BDAS).

1.1 Design Goals

With the advent of relatively inexpensive computing power, new possibilities in the area of biological signal processing can be created. The medical field still uses expensive monitoring devices that are outdated and cumbersome to use. By using basic Digital Signal Processing (DSP) techniques coupled with the storage and retrieval of a microcomputer, biosignals can be acquired and analyzed in a very efficient way.

Many existing systems are composed of different blocks as outlined in Fig 1.1. The chart recorder can be replaced by disk storage, printer/plotter, and computer screen. The analog filters and rectifier subsystem can be replaced by DSP software.

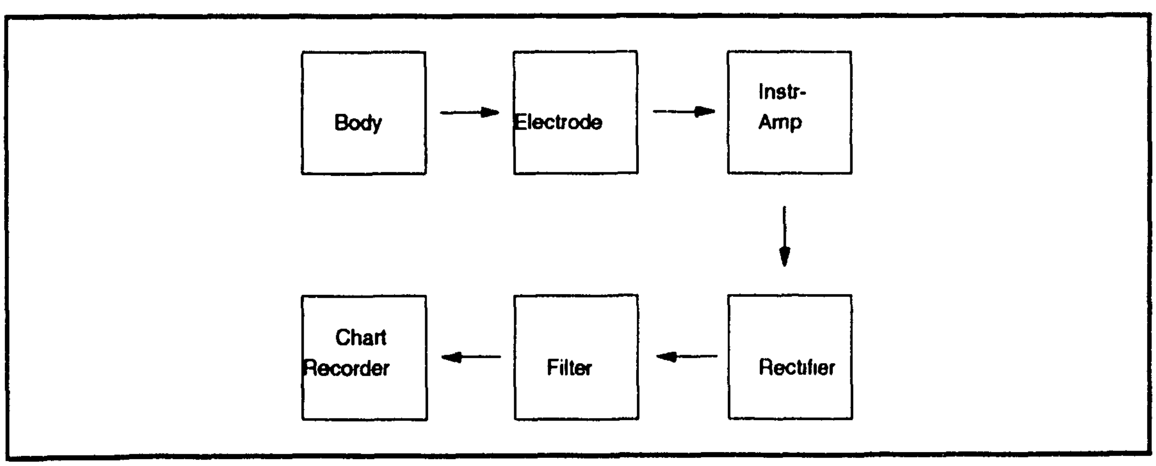

Fig 1.1 Typical existing system

The Bio-signal Data Acquisition System (BDAS) developed consists of both hardware and software. One design criterion is to minimize hardware processing on the signal in order to gain maximum flexibility by having the processing done in software. Due to the nature (level, frequency) of most biological signals and the modularity and flexibility of the developed system BDAS can be modified to acquire and analyze most types of biological signals.

One goal of this system is to provide a way to acquire biosignals simply and to provide means to analyze these signals. Another is to use existing technology (mainly off the shelf components) to show that rather advanced systems can be constructed with minimal cost using readily available technology. The components will be chosen based on availability, cost, ease of use and functionality. Since the remote device is portable, and will run on batteries, power consumption will determine component choice is a design criteria.

BDAS is a functioning prototype. The goal is provide a complete system for EMG signal acquisition and processing. The hardware is designed using some basic and simple implementations. The design goal for each block is not to get the "best" design but a functional and working unit. Each of the blocks can be later enhanced as the need arises. The system is unique because it provides a flexible working environment. Another design goal is the adaptability for bio-signals other than EMG.

There are very few portable EMG / General purpose bio-signal systems. Thought Technology has developed a portable unit that can be used for EMG. Here are some of the features :

- 4 channels
- Sampling rate is 20 Hz / channel
- Hardware rectification and filtering (envelope detection)
- Fixed 7 Hz cutoff filter

• Uses active electrodes

The following popular systems are not portable. They consist of three units namely the preamp, coupler/integrator and chart recorder.

The Beckman type 9852 Input Coupler

• integrates (after rectification) or direct output

• integration has fixed 40 Hz low pass filter

The Beckman R511A preamplifier has the following specs

• 2, 3, or 4 channels

• Input resistance 2 megohms minimum.

• Weight 40.82 kg

• Common-Mode Rejection Ratio (CMRR)(Max gain inputs shorted) 100 dB (60 Hz)

• Input noise 1 uV RMS maximum

DISA 15C01 EMG amplifier by DANTEC.

• Input impedance > 1500 Megohms

• Noise 0.7 uV RMS 2 Hz to 10 kHz

• CMRR from cable to amp : 55 dB

direct > 100 dB

GRASS Medical Instruments Differential AC Microelectrode Preamplifier

• Gain of 10, 100 or 1000

• Input resistance of 200 Megohms differential, capacitively coupled.

• Noise 20 uV, peak to peak, wideband input shorted

• Frequency response is 0.1 Hz to 50 KHz.

• Low cutoff filters .1, .3, 1, 3, 10, 30, 100, 300

• High cutoff from 10 Hz to 50 kHz

BDAS will be designed with cost and portability as major criteria. The system must cost less than $700 and be portable enough to use with a laptop. Common Mode Rejection Ratio (CMRR) which is the ability to reject common-mode noise needs to be high (around 100 dB).

Chapter 2 Biosignal Data Acquisition System (BDAS)

The system that was designed, developed and implemented is called the Biosignal Data Acquisition System (BDAS). The system can acquire and store biological signals. The signal can then be viewed, processed and analyzed.

2.1 Overview

BDAS consists of both hardware and software. There are two different hardware versions of BDAS. Version 1 is a PC card that plugs into an IBM compatible PC, whereas version 2 is a stand-alone remote unit. There are three software blocks: main system on the PC, interface on the PC, and remote interface running on the remote unit. The main system is identical for both hardware versions, but there is a separate interface portion for each version.

The different versions were designed for different applications. The PC card version was designed for standard clinical applications. The remote version was designed for non-clinical applications. The remote version allows the subject complete freedom to be tested in a "natural" environment such as a runner on the track rather than on a treadmill in the laboratory. Figure 2.1 shows the equipment overview.

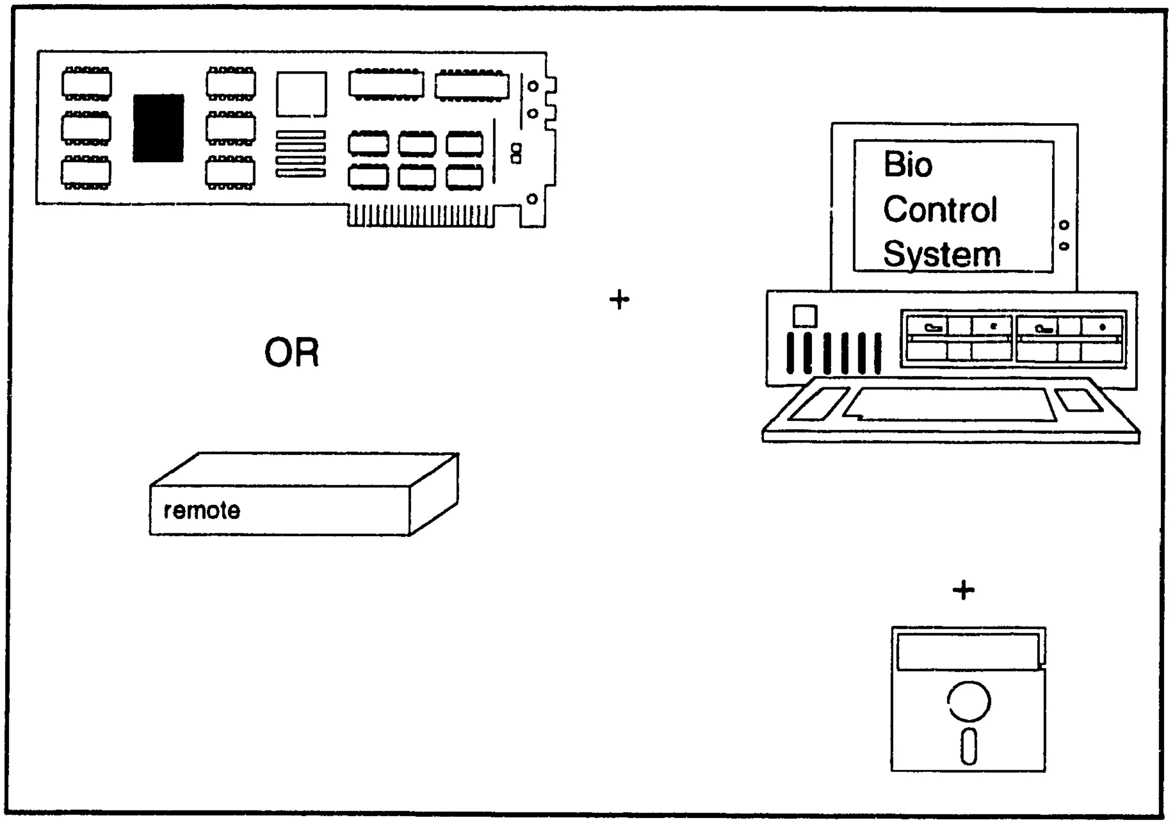

Fig 2.1 Equipment Overview

The system allows the user to acquire biological signals. It can acquire data from eight different channels. Four channels have differential inputs and the other four can have differential inputs using an external instrumentaion amplifier adapter or can be used directly. The direct channels are useful in connecting external devices that vary a resistance or voltage. Each channel has a programmable gain.

Both versions have the same front-end subsystem. The electrodes are plugged into the front end which consists of the preamplifiers, filters and channel multiplexer.

2.2 System requirements

The minimum hardware requirements needed to run BDAS consist of an IBM PC or compatible with 640kb memory and an EGA video card. The software is divided into the acquisition subsystem and the post processing subsystem. The acquisition subsystem requires the version 1 BDAS plug-in card or version 2 BDAS remote unit. The post processing software can be used by itself requiring no extra hardware.

2.3 Design

Most available computerized bio-signal acquisition systems store the signal after it has been processed. This restricts the analysis possibilities. BDAS acquires the signal and stores the signal in "raw" (no processing done) form.

The system needs to be easy to use. Therefore, the main system is a menu-driven application using simple pop-up menus. Most of the available systems allow the viewing of only one group of signals. Since different groups of signals sometimes need to be viewed at the same time, multiple windows are included in BDAS. In each window, a group of signals may be viewed.

An A/D converter is needed to convert the analog bio-signal to a digital stream. The bio-signals of interest are in the range of 1 uV to 2.5 mV. In order to have approximately 1 uV accuracy when sampling the 2.5 mV signal, at least 12 bits are needed. With 12 bits each step is

$$\frac{2.5mV}{2^{12}} = 0.61\frac{uV}{step}$$

Noise is a major problem during signal acquisition. Instrumentation amplifiers will reduce the contamination by noise. Most A/Ds have input ranges from 0 to 5V to 0 to 15 Volts single-ended or from $\pm 5V \quad to \quad \pm 15V$ bipolar. Since bio-signals are bipolar, a bipolar $\pm 5V$ A/D was chosen. The A/D device chosen has a 13 bit output. 12 of the bits will be used, with the most significant bit ignored. This gives an input range of $\pm 2.5V$. The bio-signals need to be amplified and filtered before being connected to the A/D. One A/D can service many channels when a multiplexer is used. This whole subsystem from electrode to A/D input is called the BDAS front end. The op-amps will be also powered from $\pm 5V$. A large voltage swing near the power supply rails can cause distortion and clipping. To avoid this, the system allows $\pm 2.5V$ and not the full $\pm 5V$ voltage swing at the input to the A/D. The preamplifiers have gains ranging from 1000 to 16000; therefore, the maximum input signal from the electrodes is

$$\frac{2.5V}{1000} = 2.5mV$$

In the majority of cases, the signals will be less than 1 mV in amplitude.

The A/D is then read at a rate corresponding to the sampling frequency. The system buffers the signal samples and then sends packets of information to the PC. The PC takes these packets of information and stores them in a file or buffer. At the same time the PC can display the incoming waveform. Once the bio-signal is stored on disk, Digital Signal Processing (DSP) techniques can be used to manipulate the signal to extract useful information.

The system is complete, with the only extra component being an IBM compatible PC. It contains all the necessary hardware and software for the acquisition, processing and analysis of biological signals. The software is menu driven using a hierarchy of pop-up menus.

2.4 System Functional Specifications

- 8 input channels divided into two groups of 4 channels
- 4 differential input channels with programmable gain
- 4 direct or differential (with external instrumentation amplifiers) input channels with programmable gain
- Input signal amplitude up to 2.5 mV using the instrumentation amplifiers.
- Input signal amplitude up to 250 mV direct.
- 8 kHz maximum sampling rate (total of 8 channels)

- IBM PC compatible software
- "raw" signal storage
- Menu driven software
- Multiple viewing and printing options
- Digital filtering available in postprocessing
- Postprocessing spectral analysis
- Real-time digital filtering
- Real-time FFT analysis
- Portable system which runs on batteries up to 12 hours on one charge
- Rechargeable
- Sound and visuals for biofeedback applications

Chapter 3 Biological Signals

In order to understand the electrical characteristics of the Human Body, an overview of the pertinent chemistry and physics is needed.

3.1 Basic Chemical and Electrical Characteristics

The forces holding atoms together in compounds are called chemical bonds. There are basically two kinds of bonds namely *Ionic (electrovalent) and Covalent.* Usually most chemical bonds are partially ionic and partially covalent. When a salt (ionic compound) dissolves in water, the ions separate. Substances that exist as ions in solutions are called electrolytes. Since the human body is composed mainly of water and electrolytes such as KCl and NaCl the electrolytes break up into Na+, K+, Cl- ions [Tortora, 1981].

A volume conductor is a conductor that allows conduction in three dimensions. An example of a volume conductor is glass full of water that contains ions. The *Human body* due to its chemical composition is mainly water containing many different ions and therefore is a volume conductor. In general, current generated in one part of the body usually can reach any other part of the body. Current is defined as a flow of electric charge (charge per unit time). Therefore, the flow of ions constitutes current [Tortora, 1981].

3.2 Cell Potential

A polarized cell is a resting cell that has a potential difference across the cell membrane. The potential difference is due to imbalance of potassium (K+) and sodium (Na+) ions between the inside and outside of the cell. The membrane allows K+ ions but does not allow Na+ to move across it. There are more Na+ ions on the outside of the cell than on the inside. The K+ ions migrate to the inside of the cell to try to balance the Na+ potential on the outside.

Not enough K+ ions are able to migrate. Therefore, a potential is created between the inside and outside. Since the potential is measured from the inside of the cell relative to the outside and there is less positive charge on the inside, the cell's resting potential is negative (between -70 and -100 mV). Depolarization consists of sodium going into the cell and potassium coming out. When the cell is stimulated, the membrane suddenly lets Na+ ions pass through, resulting in an influx of Na+ ions to the cell, with a +20mV potential (the depolarized potential [action potential] is usually +20mV because the entering Na+ ions travel faster than the leaving K+ ions). In the resting state, there is an excess of sodium on the outside and an excess of potassium on the inside. Groups of cells depolarize when one cell receives a stimulus. The stimulated cell's depolarization then stimulates another cell and proceeds to depolarize the whole group, creating a domino effect [Tortora, 1981].

3.3 Signal Characteristics

Muscles are activated and deactivated when the body moves. ElectroMyoGraphy (EMG) is the measurement of electrical signals generated in the muscles and it can be used to determine the pattern of muscle activation for a specific movement. Muscle cells are activated by nerve cells [Basmajian, 1985].

When groups of cells depolarize one after another, they create a travelling wave. A device (electrode) can be placed on the surface of the skin overlying an active muscle to record this wave. The recording will be the vector sum of the many depolarizations taking place. If more than one muscle (one or more motor units) is depolarizing (which is the usual case) then the recording will contain the sum of all these signal sources. This interference of one muscle group by another is called crosstalk which is a source of biological noise [Basmajian, 1985].

3.3.1 EMG Signal levels

The actual electrical levels generated by muscles contracting depend on their source. EMG signal levels range from 10 uV to 5 mV peak for the raw signal. The frequency response is from 10Hz to 10kHz. The bulk of the signal energy is under 1 kHz and in a 2 mV peak to peak range [Basmajian, 1975].

3.3.2 Electrical Signals

The cells that generate electrical activity by depolarizing usually do so in groups. Different labels have been given to the acquisition and recording of biological signals depending on the location and function of these cell groups. For example, if the signals recorded originate from the heart, they are called ElectroCardioGrams (ECG).

In general, the different signals have different frequency responses and different voltage levels. However, they can be described as being relatively low-frequency signals with most of them being under 1 kHz and with voltage levels in the range from 1 uV to 5 mV.

Table 3.1 lists some biological signals of electrical nature with their pertinent characteristics. In practice, most of these signals have levels between ±2.5 mV and have relatively low frequency, mainly under 1 kHz [Gowitzke, 1988].

Table 3.1 Electrical signals [Bukstein, 1973]

Signal Name	Signal Level	Frequency Range
ElectroCardioGram (ECG)	10uV to 5mV	0.05Hz to 85Hz
ElectroEncephaloGram (EEG)	10uV to 200uV	DC to 100Hz
ElectroMyoGram (EMG)	20uV to 5mV	10Hz to 10kHz
ElectroRetinoGram (ERG)	0 to 5mV	DC to 25Hz
ElectroOculoGram (EOG)	50uV to 5mV	DC to 1Hz

3.4 Electrodes

Converting an ion flow to an electrical voltage is the purpose of the electrode. An electrode is an object that is placed on the skin to sense current (it usually consists of metal with conductive gel). The electrodes function like the electrodes in a battery. At the electrode site, a redox (reduction-oxidation) process is occurring. It is important that the electrode have this property in order to sense the proper voltage. Wire is attached to the electrode to pass the current to amplifiers. After the voltage is amplified the voltage is then sent to the computer through isolation circuits. Current toward a positive electrode will register a positive voltage, whereas current away from the electrode will register a negative voltage [Basmajian, 1985].

Electrodes come in two basic varieties, namely active and passive. Active electrodes capacitively couple the signal and provide amplification at the source. On the other hand, passive electrodes on the other hand lower the impedance, usually with an electrolyte gel.

This allows a redox system to form at the electrode-skin contact. The disadvantage of this system is that the contact area has to be prepared and gel applied, whereas in the active system, no preparation is needed.

A surface electrode is placed on the skin and is a non-invasive electrode. It picks up signals from the surrounding tissue and, therefore, care must be taken to place the electrodes such that they pick up signals from one particular muscle without other muscle signals interfering [Basmajian, 1985].

3.5 Noise

We can define noise as any unwanted signal. There are many different types of noise, usually identified from their source.

Typically, we are interested in a specific muscle or muscle group. When other muscles in the neighbourhood are active as well, their signals may also be picked up and can interfere with signals that are under observation. By proper electrode placement, cross-talk can be reduced.

Narrow-band noise comes from signal sources that produce narrow frequency band signals. The power lines are a major source of unwanted noise, producing a 60Hz noise signal. Radio and television stations also produce narrow-band signals, although these signals are usually too high in frequency to cause a problem. Notch filters can reduce the power line interference.

3.6 Electromyography (EMG)

Electromyography consists of graphing the electrical activity of muscles (myo). Most of the time, the recording of a combination of motor units is done. This raw EMG signal is very complex compared to a cardiac signal, where the depolarization starts from one point and proceeds relatively smoothly. The complex EMG signal is due to many motor units firing asynchronously with all these signals travelling through a volume conductor to the electrode.

Extracting useful information from a raw EMG signal involves signal processing. The two basic questions are: is the muscle active and at what level is the muscle being used? Quantifying the amount of use is a problem. Integrating a signal over a period of time is not a sufficient measure of the signal, since the signal has a zero-mean value. There are two common solutions to this problem.

1. Rectify, filter and then integrate the signal.

2. Square the signal, then take the avc age (mean) and finally take the square root, more commonly called RMS.

Both methods distort the signal since both methods involve non-linear processes [Basmajian, 1975].

A typical system would consist of electrodes which pick up the signal. Since the signal levels are too small to drive an output device (plotter), the signal is amplified. Then the signal would be rectified and filtered. Finally the filtered signal is fed to a chart recorder.

Chapter 4 Main Software

The software system can be divided into three main sections namely acquisition, processing and analysis. The software was written in C++ and assembler. C++ is ideal for this application due to its high-level structures and its ability for easy low-level access. Some assembler portions were needed to achieve maximum throughput.

The system comes on one 1.44Mb floppy diskette. It consists of two program files and some utility files. On startup, the system tries to autodetect the graphics card on the computer. It displays the type of card and allows the user to change the type of graphics display.

The physical screen consists of rows of pixels (dots) which form the graphic display. An autodetect routine displays the detected resolution of the screen. The most common resolutions are EGA = 640 x 350 and VGA = 640 x 480.

The system is a menu-driven application. The mouse or the keyboard can be used to control the system. The screen can be divided into different windows. In each window, one or more signals can be displayed. There is no limit to the amount of windows except limits due to screen size or memory constraints. The window dimensions and location can be specified. The windows can overlap. The different menu options are show in Fig 4.1.

```
Main   |- Exit
Menu   |- Window Menu   |- Exit
       |                |- New Window
       |                |- Delete Window
       |                |- Zoom Window
       |                |- Unzoom Window
       |                |- View Window -------------   |- Exit
       |                |-                             |- Next
       |                |- Set Current Window          |- Prev
       |                |- Set Current Plot            |- Zoom
       |                |- Separate Plot               |- Goto
       |                |- Single Plot                 |- Autoscale
       |                |- Combine Plot
       |                |- Set View FFT
       |                |- Set View Input
       |                |- Set View Output
       |                |- Define Signal
       |                |- Position
       |                |- Integrate
       |
       |- Process Menu  |- Exit
       |                |- Full Wave Rectify
       |                |- Envelope
       |                |- Filter Response
       |                |- Delete Response
       |                |- Filter Channel
       |                |- IIR Filter Menu ---------   |- Exit
       |                |                              |- Band Pass
       |                |                              |- Band Stop
       |                |                              |- High Pass
       |                |                              |- Low Pass
       |                |                              |- Notch
       |                |- FIR Filter Menu ---------   |- Exit
       |                |                              |- Band Pass
       |                |                              |- Band Stop
       |                |                              |- High Pass
       |                |                              |- Low Pass
       |                |                              |- Notch
       |                |- Load User Filter
       |                |- User Filter Menu
```

Fig 4.1 Menu System Overview

```
Main
Menu    |- Capture Menu    |- Exit
(Cont.) |                  |- Monitor
        |                  |- Monitor and Capture
        |                  |- Capture to File
        |                  |- Download From Remote
        |                  |- Start Remote Capture
        |                  |- Config Capture
        |                  |- Config Com port
        |                  |- Real Time FFT
        |
        |- File Menu       |- Exit
        |                  |- Save Channel
        |                  |- Save Buffer
        |                  |- Append Buffer
        |                  |- Load Set
        |
        |
        |- Generate Menu   |- Exit
        |                  |- Sin Wave
        |                  |- Square Wave
        |- Print Plot
```

Fig 4.1 Menu System Overview (cont.)

The system starts up by typing BIO_GM. The system autodetects the video and the user confirms that the display driver is correct by pressing *enter*. The default configuration file (bio.cfg) is loaded and the tables for FFT and the filters are generated. Next, the copyright notice (Fig 4.2) is displayed. After the user presses *enter*, the **Main Menu** appears.

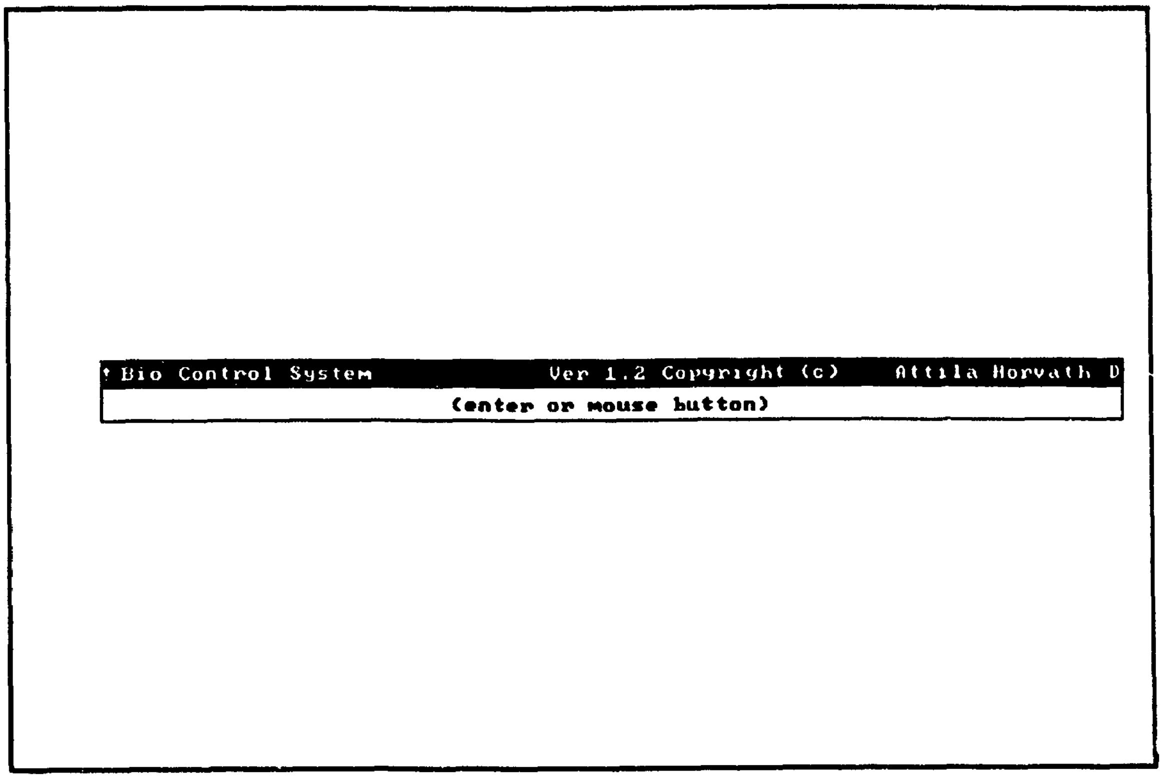

Fig 4.2 Copyright

4.1 Main Menu

The Main Menu is the heart of BDAS (Fig 4.3). All the major submenus can be accessed from the main menu. The system can only be exited from the main menu. The system will be described with the help of an example. The example consists of a 10 Hz sine wave signal with a 60 Hz sine wave noise component (Fig 4.4). The first step is to acquire the signal. This is done by selecting the **Capture Menu.**

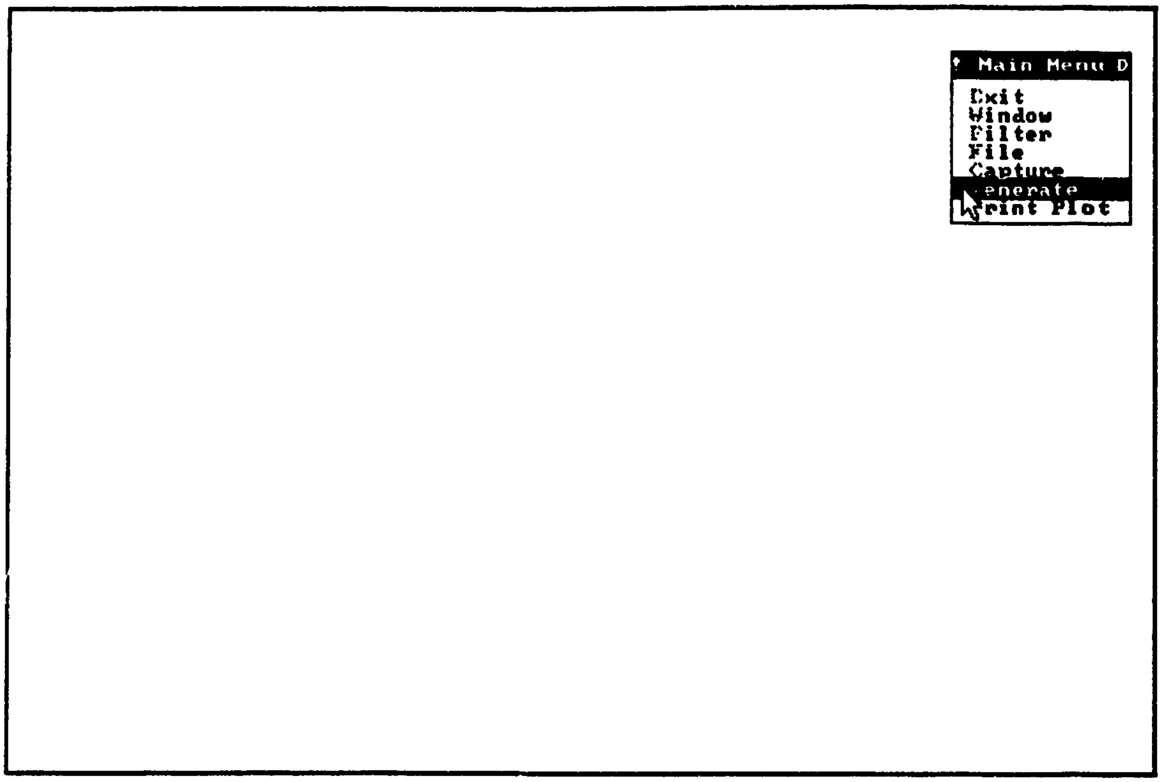

Fig 4.3 Main Menu

4.2 Capture Menu

The *Capture Menu* is where signals are acquired through the remote device. The system communicates to the remote device using the serial port. The first step in using the system is to configure the communication port by choosing the *Config Comm Port* in the *Capture Menu.* The choice of serial port is between **Com1** and **Com2.** The other parameter that needs to be specified is **Baud Rate.** After the **Baud Rate** is entered, the remote unit should be turned on. The system will try to establish communications with the remote device.

The next step is to configure the input channels by choosing the *Config Capture* in the *Capture Menu.* The following characteristics need to be entered.

• ***Front end filter cutoff*** (less than half the sampling frequency)

There are 7 different cutoff frequencies ranging from 50 Hz to 4000 Hz. Choose the filter with the lowest cutoff frequency that allows the highest signal frequency to pass. Use this filter for single channel captures. Otherwise, enter **None.**

• ***Input scale***

This is a multiplication factor that is used to determine the actual value from the stored A/D value. This parameter takes into account the preamplifier gain. If the signal is amplified before the BDAS connection, then the input scale parameter should be increased by the amplified amount.

• ***Sample frequency of a channel*** (valid for one channel only)

Should be at least twice the highest frequency of interest.

• ***Starting Channel***

The starting channel number which defaults to channel 0

• ***Number of signals*** (input channels)

BDAS supports up to 8 input channels, 4 internal and 4 external.

• ***Tone y/n***

Switch tone option on for channel 0. Tone sounds when the A/D value is higher than the input_start value. The tone frequency is equal to

$$output\_freq = \frac{(A/D\_value - input\_start)}{input\_step} x freq\_step + starting\_freq$$

• ***Starting Frequency (tone option)*** (Hz)

The tone frequency that corresponds to the input_start value.

• ***Frequency Step (tone option)*** (Hz)

For each input_step, the amount of frequency the tone increases from the starting_frequency.

• ***Input Start (tone option)*** (uV)

The minimum input value needed to turn on the tone.

• ***Input Step (tone option)*** (uV)

The amount the input has to be above the input start value to increase the starting frequency by the frequency step.

• ***Fixed Gain***

This gain is fixed by the instrumentation amplifiers (IA) and filters. It should only change when the external channels (4 to 7) are used or the IAs are modified.

For each channel channel the following needs to be specified.

• ***Real-time Filter***

A choice of real-time filters is given such as low pass Butterworth filter, envelope detection, notch filter and rectification. The cutoff frequency (low pass, envelope) or the center frequency and bandwidth (notch) can be specified.

• ***Gain*** (can be different for each channel)

While viewing a signal as it is being captured the gain can be adjusted to get a full dynamic range. The gain can be from 1 to 16. The total gain is the fixed gain x channel gain.

• ***Channel Description***

A short title for the signal should be entered.

The first option in the capture menu is *Monitor*, which captures a signal to memory and displays it on the screen at the same time. This option should be used to make sure the signal characteristics are correct. Once the signal characteristics have been entered and checked with *Monitor*, the *Capture to a File* or *Monitor and Capture* option can be chosen. The capture terminates by pressing any key. The *Remote Capture* option starts the remote device capturing to the on-board memory. The *Download* option downloads the data from the on-board remote memory that was previously started with remote capture. **Real Time FFT** allows channel 0 to display its frequency spectrum in real time.

4.3 Generate Menu

Test waveforms can be generated to test the system. The *sin wave* option, chosen from the *Generate* sub menu, generates a sine wave with a optional noise wave (also a sine wave). In Fig 4.4 there is a signal waveform at 10 Hz and a noise signal at 60 Hz. A *square wave* can also be generated by specifying the on time and its value and the off time and its value. The screen buffer was saved to a file, using the *Save Buffer* option of the *File Menu* (Fig 4.1), for further processing.

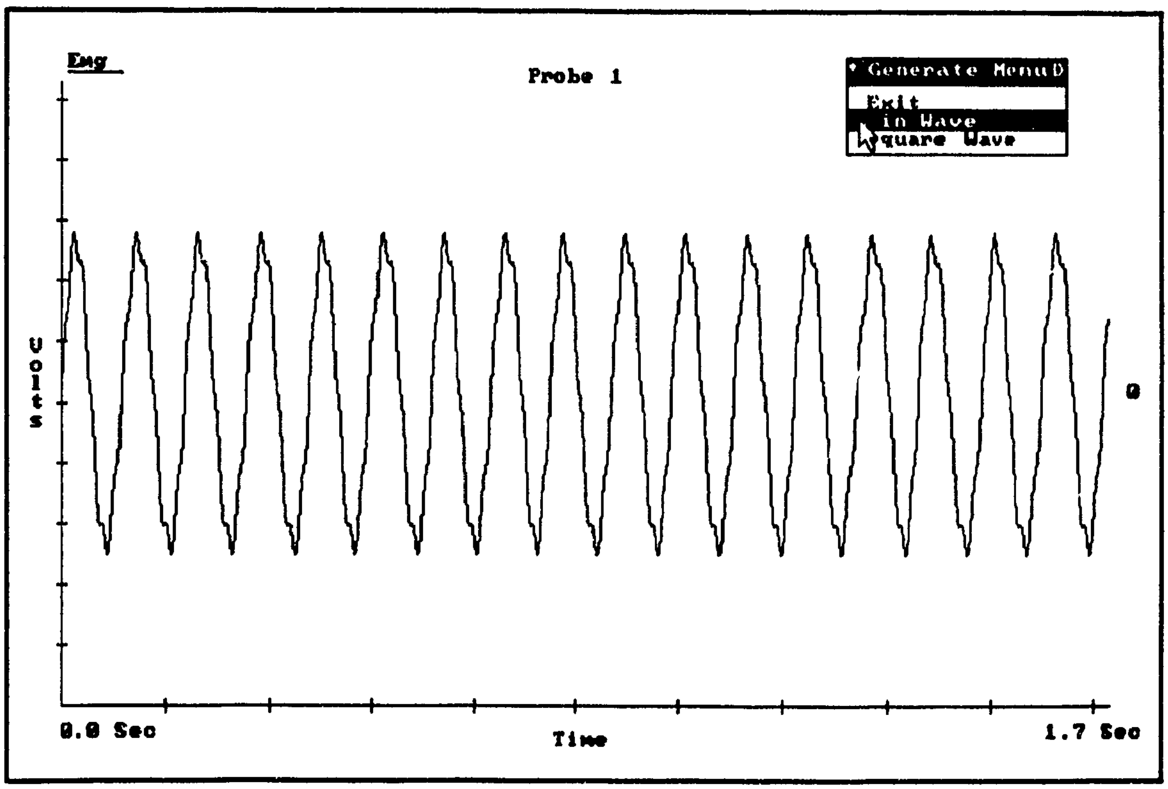

Fig 4.4 Sine Wave with Noise

4.4 Window Menu

Access to window and plot manipulation is from the *Window* sub menu. A new window can be created by specifying size and location. The last window that was manipulated is the *Current Window*. The current window can be zoomed to full screen size and back to the original size. The plots in a window can be viewed individually or together. When viewed together, the plots can be either separated or viewed on top of each other. The plot's raw signal, FFT or filtered output can be selected (Fig 4.1). The numeric values of the plots in a window can be displayed using the *Position* option. *Integration* integrates a plot between two specified markers. Multiple plots are shown in Fig 4.4. Multiple windows are shown in Fig 4.5. Overlapping windows are shown in Fig 4.6.

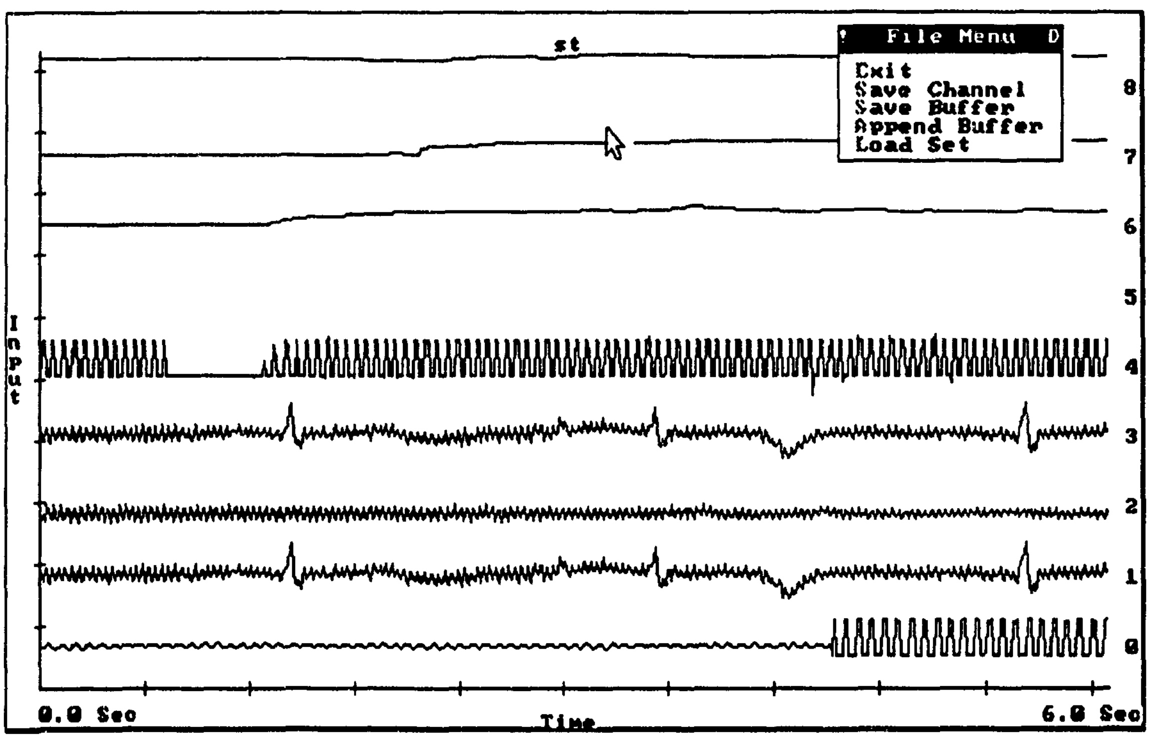

Fig 4.5 Multiple Plot View

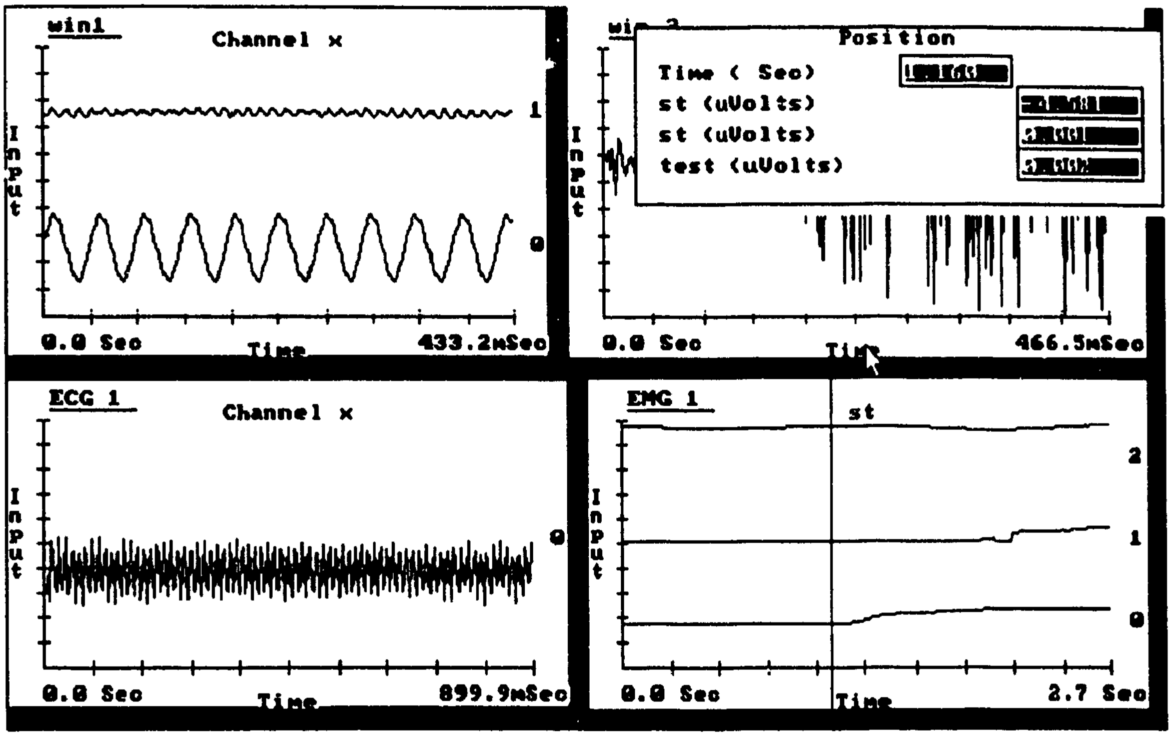

Fig 4.6 Multiple Windows

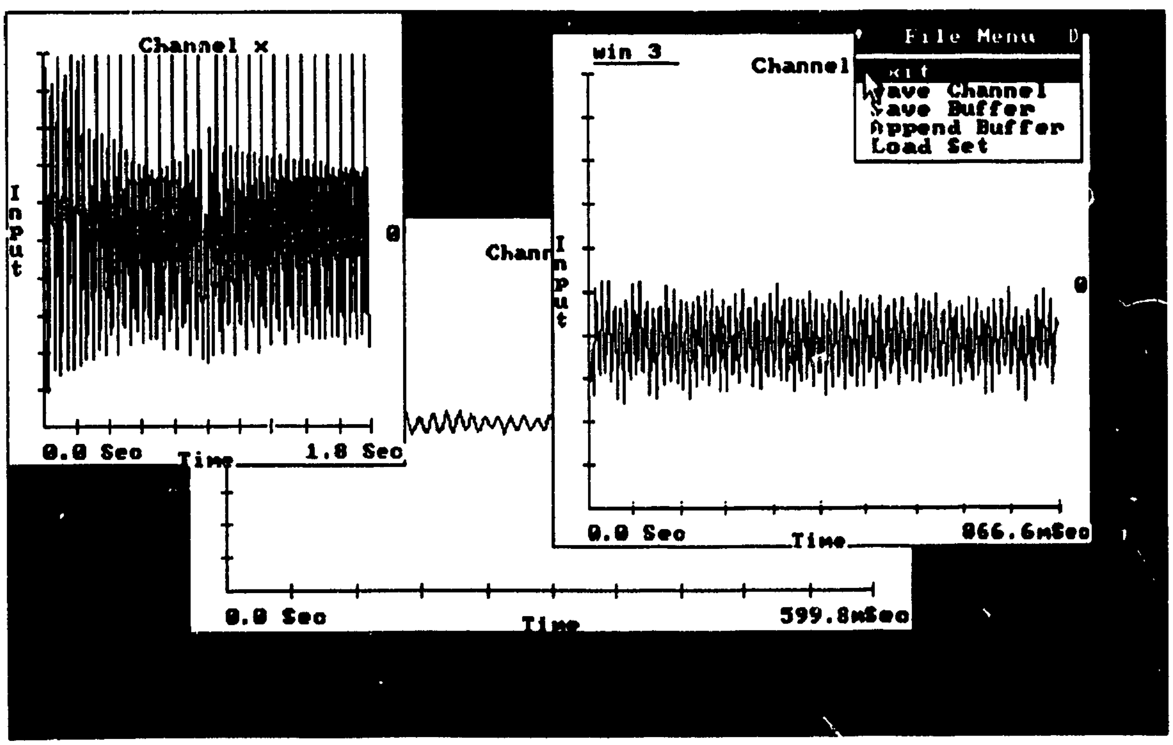

Fig 4.7 Overlapping Windows

4.4.1 Defining a window

In order to analyze a signal, a window must be first created using *New Window* in the *Window Menu* (Fig 4.1). A *Window* is a rectangular portion of the physical screen. It is defined by specifying the top left corner and the bottom right corner. The number of windows is unlimited.

• From the main menu choose the window menu.

• From the window menu choose new window.

• Enter the X coordinate for the top left corner of the window.

• Enter the Y coordinate for the top left corner of the window.

• Enter the X coordinate for the bottom right corner of the window.

• Enter the Y coordinate for the bottom right corner of the window. For the current example the default window will suffice.

• Enter window title

4.4.2 Viewing a signal

A signal's data displayed in a window is called a ***plot***. A plot must exist in a previously defined *Window*. More than one plot can be displayed in a window. Use of windows allows plots to be grouped by function. In the example the generated waveform is loaded into the default window. From the *File Menu* select the *Load Set* option.

• First, enter the file specification (wildcards are allowed). A list of files will then be displayed corresponding to the file specification.

• Choose the file from the selection list

• Finally, select the window where the signal is to be displayed.

A signal has three possible views namely raw input, filtered output and FFT (see **Set View** options in Fig 4.1). Only one view can be displayed at one time. If more than one view of a signal needs to be displayed the signal can be loaded as another plot in the window.

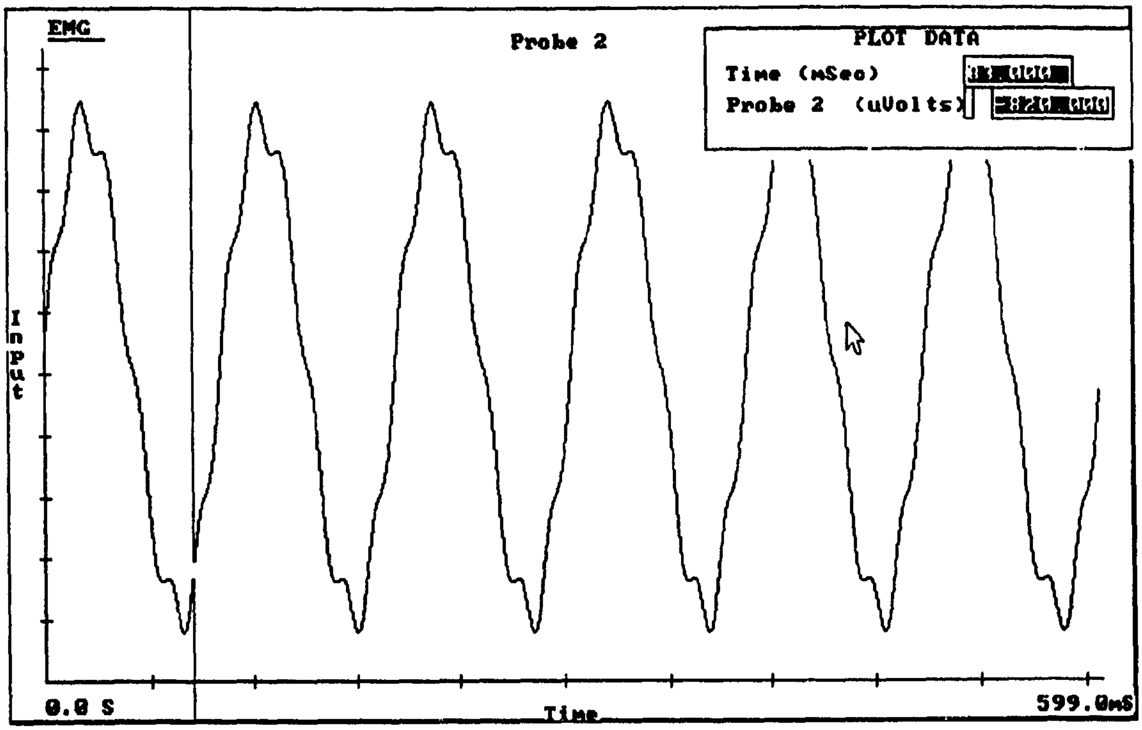

Fig 4.8 Position Option, Display Values

The *Position* option in the *Window* menu displays the actual values of the waveform at the marker position as shown in Fig 4.8. Use the arrow keys to move the marker. To exit *Position* mode press enter.

The FFT shows the spectrum of a signal. The number of points used to calculate the FFT and the sampling frequency determine the spectrum resolution. Continuing with the example, the FFT of the signal shown in Fig 4.4 is displayed in Fig 4.9. The 10 Hz signal and 60 Hz noise portions can easily be identified.

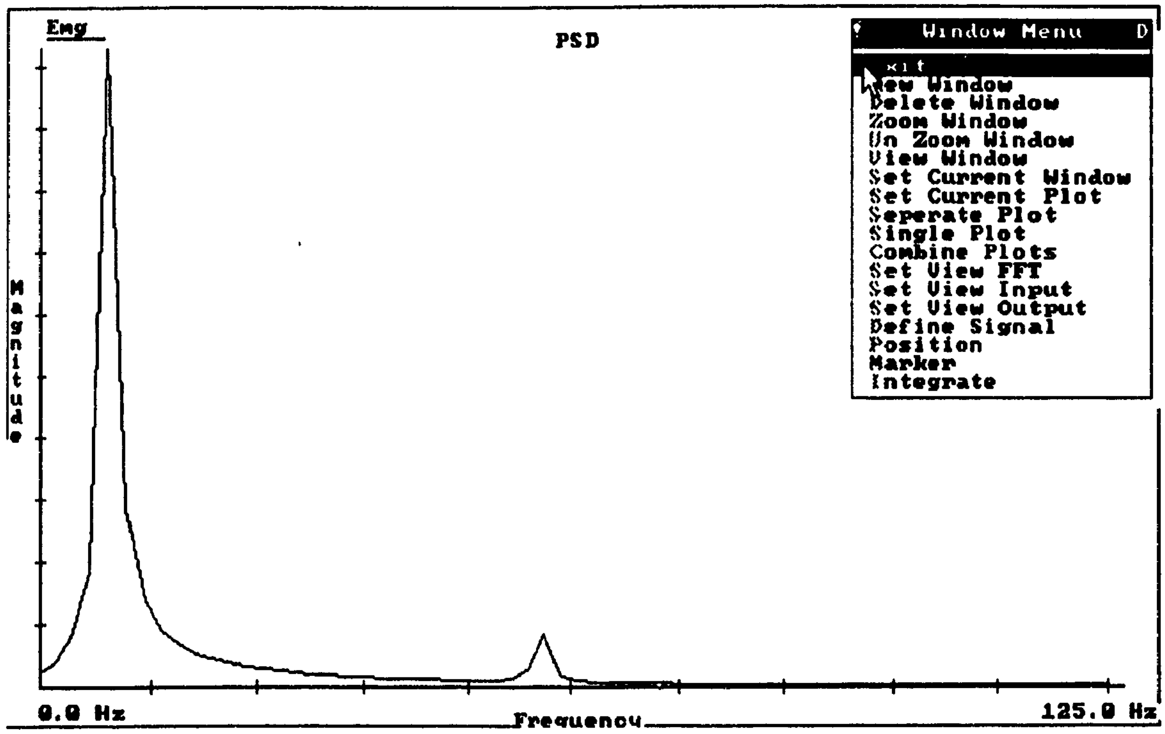

Fig 4.9 FFT of Sin Wave with Noise

4.4.3 View Window Menu

The *View Window* under the *Window Menu* can be used to view different portions of the file corresponding to a window (Fig 4.1). In most cases a captured file is too big to fit completely on the screen therefore only a **window** into the **.set** (see 4.6) file is v..wable.

Next and *Prev* shift the window by one screenful forward and backward respectively. *Goto* is used to place the left-hand corner of the window anywhere in the file. The time offset in the file needs to be specified. *Zoom* will zoom in or out the time scale (x-axis). A positive value less than one specifies a zoom in (more detail). A positive number greater than one indicates a zoom out (less detail). *Autoscale* allows signals that have very small amplitude to be zoomed vertically.

4.5 Process Menu

There are four basic types of filters available namely low pass, high pass, band pass, and notch (band stop). They are available as IIR or FIR filters. FIR filters have the property of linear phase and stability, although the computational complexity is usually higher than an IIR filter. The order and destination window for the filter can be chosen. Any plot on the screen can be filtered. The filtered data can then be viewed and or stored in a file. The *filter response* (spectrum) of the chosen filter can be viewed (Fig 4.10). There are three different filter sub-menus: FIR, IIR, and User Filter Menu.

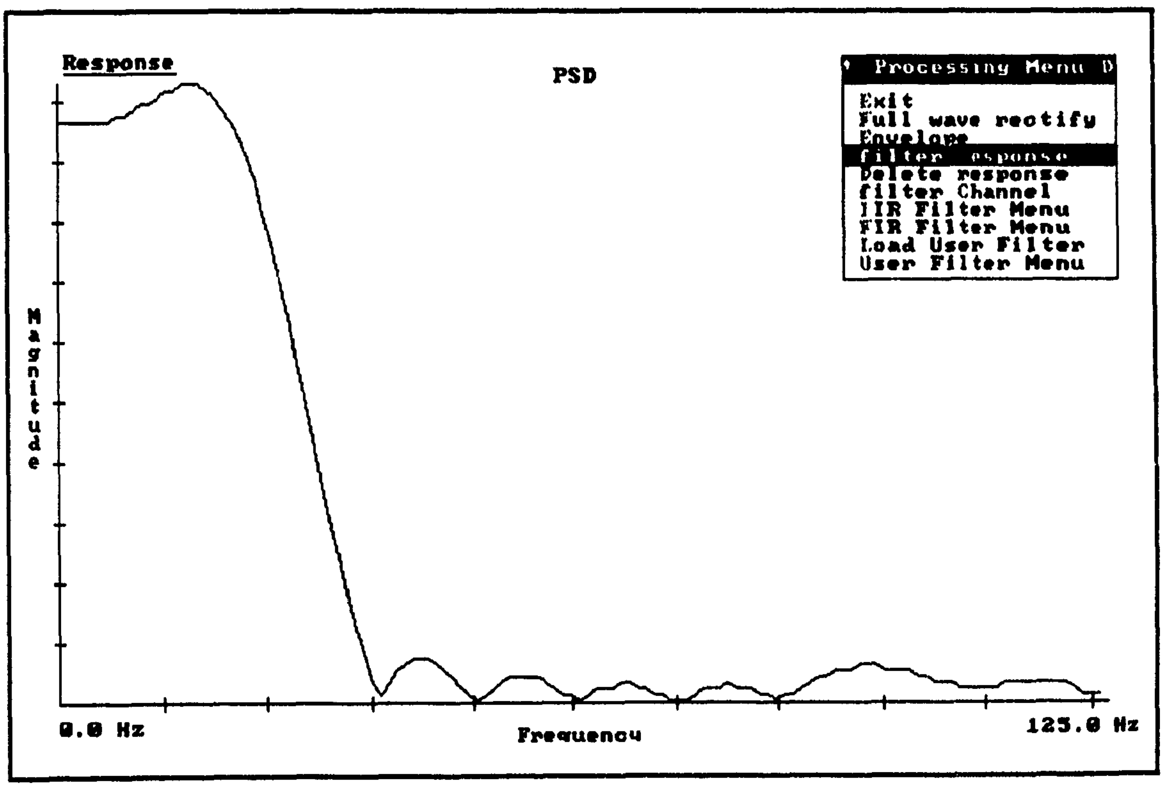

Fig 4.10 Filter Response

Filter channel will filter one channel of an entire file with a selected filter. The output will be stored in a file. This file has the same format as a .set file.

Envelope does a full-wave rectification and then performs low pass filtering.

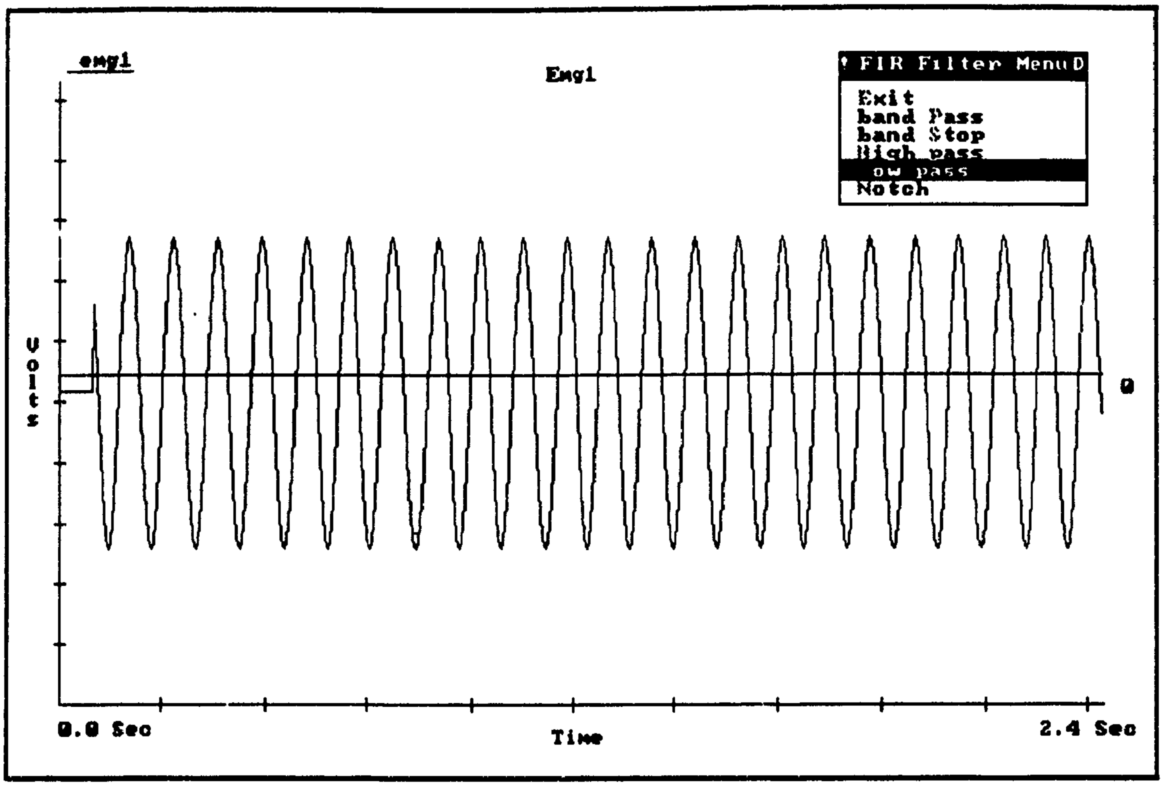

Fig 4.11 Filtered Output

In the example, the 60 Hz noise portion needs to be filtered out. In the *FIR Menu* of the *Process Menu*, the *Low Pass* option will filter the data using an FIR low pass filter with a selectable order. In the example, the upper frequency of the low pass filter is 20 Hz and the order is 10. Figure 4.10 shows the example signal after being filtered. The noise at the beginning of the filtered output (Fig 4.11) is due to the transient response of the filter. The FFT of the filtered signal is shown in Fig 4.12. The 60 Hz noise component has been removed.

In order to allow customized processing, BDAS allows for user filters specification to be loaded from files. The filters specifications are loaded using the *Load User Filter* option of the *Process Menu*. The filters can then be accessed under the *User Filter Menu* option of the *Process Menu*.

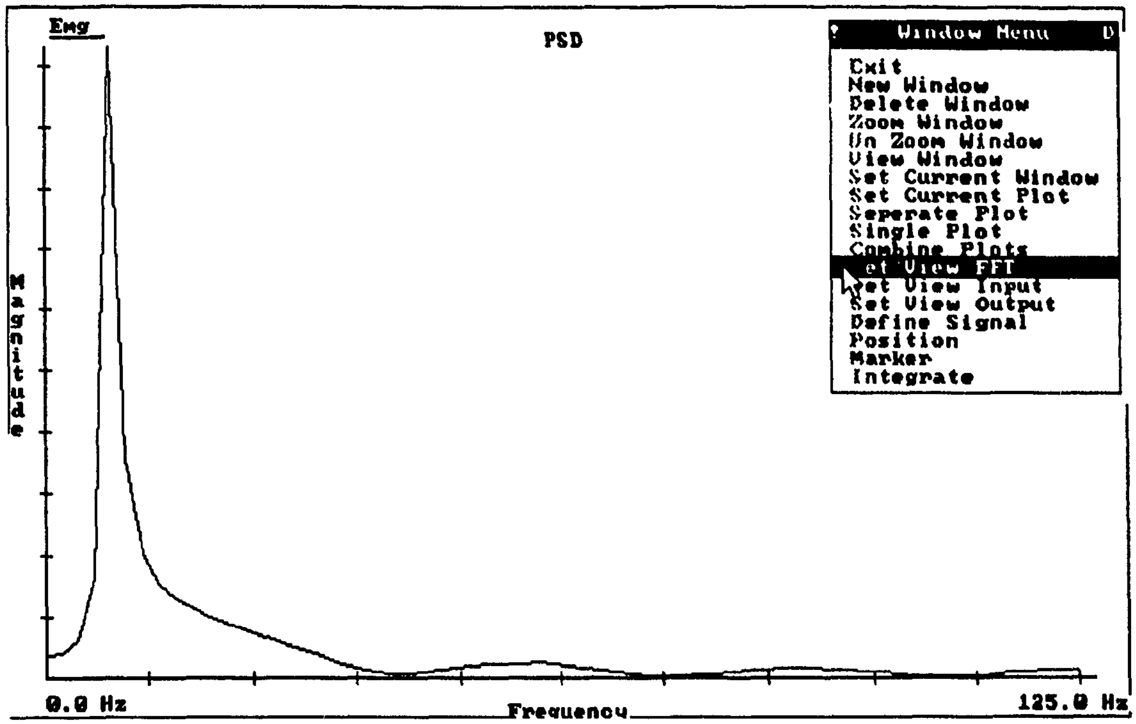

Fig 4.12 FFT of Filtered Output

The user filters implement a difference equation of the form

$$\sum_{k=0}^{N} a_k y(n-k) = \sum_{r=0}^{M} b_r x(n-r)$$

The filter specification in the file has three lines. The first line is the filter description. The first number on the second line is the number of b's (M in the difference equation). The second line specifies the b's starting with b_0 in the difference equation. The first number on the third line is the number of a's (N in the difference equation). The third line specifies the a's starting with a_1 in the difference equation.

For example, the difference equation for the filter

$$yn = 0.1yn - 1 + 0.2yn - 2 + 0.01xn + 0.15xn - 2 + 0.45xn - 3$$

has the following file description:

```
Test filter                 // filter title
4 0.01 0 0.15 0.45          // b's (multipliers of the x[n-k]'s)
2 0.1 0.2                   // a's (multipliers of the y[n-k]'s)
```

4.6 File Menu

There are two types of signal files, namely: ASCII (.ASC) and **.set** files. The ASCII file is used for exchange between other programs. Other data acquisition systems can be used and the data can be exchanged through the ASCII file format. **.set** files are created by the data acquisition system part. They contain one or more signals. BDAS can only work with **.set** files. There are two programs provided which convert ASCII files to **.set** files and vice-versa.

The **.set** files have a specific header before the data. A copy of the 'C' header file shows the contents of the header. For each channel, there are two bytes for the gain followed by a null terminated string, which is the description. Since the strings are of variable length, the start of the data is not fixed in the file. A copy of the 'C' header file is shown below.

```
//----------------------------------------------------------+
// file header
//----------------------------------------------------------+

struct head {
      int     identifier;           // file identifier
      int     tot_channels;
      int     sample_freq;
      int     num_bits;             // number of bits per sample
      int     input_scale;          // voltage level per step
FOR EACH Channel :
      char    *channel_desc;
      int     gain;
      int     offset;               // voltage offset
};
```

A **.set** file can contain many channels. The *save channel* option in the *File Menu* (Fig 4.7) can extract a channel from a **.set** file that was loaded into a window. This file can then be manipulated as any other **.set** file.

Save Buffer can be used to extract sections of a channel into a file. Different sections can be concatenated into one file using the *append buffer* option.

There are two parameter files types, template (**.tpl**) and config (**.cfg**). Both these files are **ASCII** files which can be edited. When using the system, many windows might be specified and signals loaded into these windows. The window specifications and the files that were open in the windows can be saved in a template file using the *Save Template* option in the *File Menu.* A previously saved template file can be loaded using the *Load Template* option in the *File Menu.* Figure 4.13 shows a sample **.tpl** file.

The capture parameters and system parameters are contained in **.cfg** files (Fig 4.14). When the system starts up it reads a file called *bio.cfg*. This system config file contains the default options for the capture parameters such as sampling frequency, tone parameters, number of channels etc. The *bio.cfg* config file also contains a default template which will be loaded on system startup.

A sample user config file looks like the following (without comments)

```
2                      // number of windows
0 0 630 230 1 ECG 1    // top left (x0 y0) bottom right (x1 y1) window name
heart1.set             // signal file
0 240 630 470 1 ECG 2  // top left (x0 y0) bottom right (x1 y1) window name
heart2.set             // signal file
```

Fig 4.13 Sample template file

A sample *bio.cfg* file looks like the following (without comments)

```
n          // Mouse  y or n
2-1.tpl    // default user config file or template
1          // number of channels
250        // sampling frequency
12         // number of bits for the A/D
2500       // maximum input to A/D in mV
1000       // Fixed Gain
0          // starting channel
1          // Gain
0          // offset
4          // real version of butterworth realtime filter cutoff
16         // integer version of butterworth realtime filter cutoff
60         // notch filter center frequency
5          // notch filter bandwidth
5          // envelope detector upper frequency cutoff
300        // starting frequency
20         // frequency step
400        // input start
200        // input step
30         // Tone sounds for a minimum of nn samples.
y          // Tone on y or n
```

Fig 4.14 Sample config file

4.7 Data Acquisition

For the PC card version, the heart of the data acquisition is the Interrupt Service Routine (ISR) that services the timer interrupt. The user decides the sampling frequency and the timer is programmed to interrupt the CPU at that rate. The data is first written to one of two buffers. When the buffer is full, it is written out to disk. As one buffer is being written to disk, the other is being filled by the A/D samples. If the sampling rate is slow enough, the data is also plotted on the screen or printed out on a printer.

In the remote unit version, the data is acquired through the serial port. The heart of data acquisition is the Interrupt Service Routine (ISR) that services the serial port interrupt. The data bytes are buffered. The system requests data from the buffer routines. The remote software used here is discussed in more detail in Chapter 5.

4.8 Processing summary

The acquisition system delivers, in a buffer or a file, a raw signal. Depending on the analysis, different processing options can be used such as filtering to remove noise and linear envelope detection to quantify EMG activity. There are different filters available which can be customized. BDAS has a set of FIR filters to perform the basic types of filtering such as Low Pass, High Pass, Bandpass, and BandStop. Also included is envelope detection processing which takes the raw signal and passes it through a notch filter to filter out the 60 Hz powerline noise. The signal is then passed through a full wave rectifier and, finally, the signal is low pass filtered. Digital Signal Processing (DSP) is discussed in more detail in Chapter 6.

Chapter 5 Remote Software

The software portion on the remote system consists of an interrupt driven main loop that controls the remote data acquisition. There is a routine that controls each device connected to the CPU such as the serial port and A/D.

The system was written in 6811 assembler code for the MC68HC11 microcontroller. One of the five timers is programmed to interrupt the CPU at the sampling rate. At each interruption, the system reads the A/D port and stores the data in a buffer. The multiplexer is then switched to get the input from the next channel. The programmable gain amplifier is programmed to the gain of the next channel. Between the interrupts the system sends data to the serial port. The data is formatted in packets of information. Figure 5.1 shows the remote software overview.

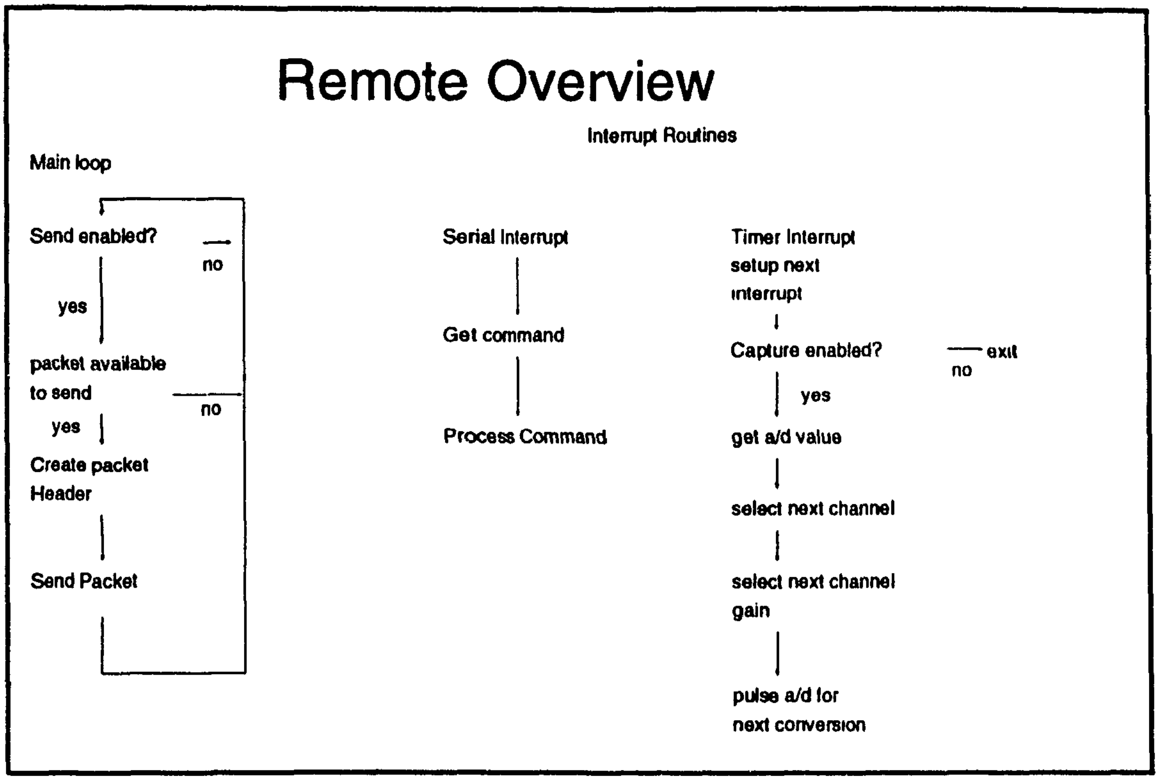

Fig 5.1 Remote Software Overview

5.1 Command Control

The system can be configured through the serial port. The number of channels and sampling rate are some of the parameters that can be modified. The PC sends a command packet to the remote unit. When a command packet is received by the remote unit data acquisition is suspended until a start acquisition command is received.

The remote commands are shown in table 5.1.

The command value is first sent. Data if any is returned. All commands return 00H to indicate that the command was processed.

Table 5.1 Remote commands

Command Value	Brief Description	Parameters / Return Values
01	Start send	
02	Stop send	
03	Set Freq	LSB then MSB
04	Set Channels	byte = number of channels
05	Set Gain	channel #, gain = 0 to 15 (x1 --> x16)
06	Set Filter	0 --> 7 (one of 8 cutoff freq.)
07	Start Download	
08	Get status	41H if O.K
09	Start Remote Capt.	
10	Change Baud Rate	6811 Baud Rate
11	Get setup	
12	Reset Buffers	

The most common MC68HC11 baud rates with an 8MHz crystal are show in table 5.2. This value is used in the Change Baud Rate command (10 in table 5.1). The communication baud rate between the PC and remote unit can be configured in the **Capture Menu.** The default baud rate is 1200.

Table 5.2 MC68HC11 Baud Rates

Baud Rate	Command Parameter Value
9600	30H
4800	31H
2400	32H
1200	33H (default)

The get setup command returns the structure as shown in table 5.3.

Table 5.3 Get Setup Structure

Byte Offset	Size(bytes)	Value
0	2	Sampling Rate
2	1	Send flag
3	1	Number of Channels
4	2	Buffer Size
6	2	Next Buffer
8	2	Send Offset
10	2	Buffer Offset
12	2	Buffer Start
14	2	Buffer End
16	1	Buffer Full Flag
17	2	Send Buffer Start
19	1	Current Channel
20	1	Capture Flag

5.2 Buffer Management

One of the crucial elements of the remote software is the buffer management. The buffers hold the values acquired from the A/D. Buffers are needed in the system because the rate of acquisition from the A/D may be higher than the rate of transmission. Buffering is also used when the remote unit runs in stand-alone mode. There is a large area of RAM reserved for the buffers. The buffers are of fixed length, although their length is configurable. The start and end of the buffer area is also configurable. The buffers are managed as a circular queue. The buffer format is shown in Fig 5.2. There is a pointer called the buffer_offset which is the location of the next available storage location for the A/D value. Send_start is the pointer to the buffer that is currently being sent. Send_next is a pointer to the next available buffer to be sent.

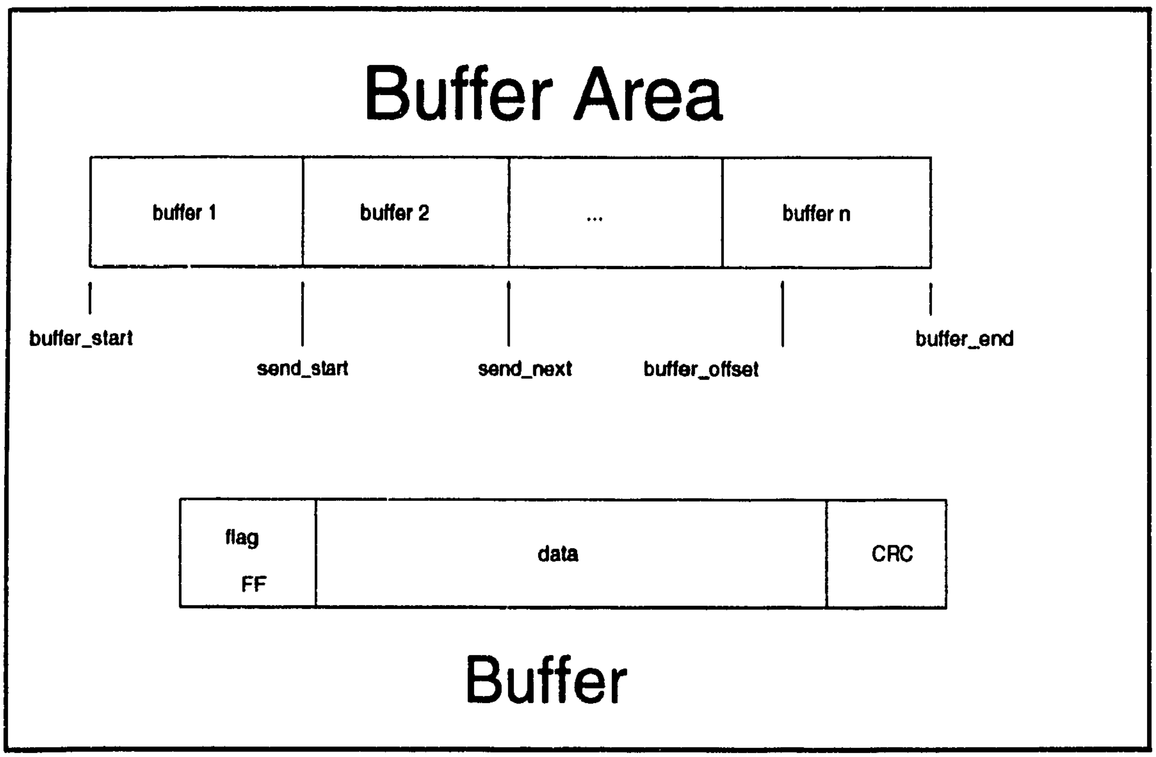

Fig 5.2 Buffer Format

The system can be in one of four states. The first state occurs when the buffer_offset is filling the next buffer to send. The buffer_offset is greater than send_start and less than send_next. No buffer is available and a buffer is currently being filled. In this state, the system waits untill the buffer is full.

The second state occurs when the buffer_offset pointer is greater than or equal to send_next. In this state the system can send the next available buffer while it is filling the next buffer.

The third state occurs when sending is much slower than acquisition. The time will come when buffer_offset will equal the send_start pointer. The buffer is full. In this state the acquisition will stop and the system will wait until a certain number (high water mark) of the buffers have been sent before resuming. The high water mark (variable) is presently set so that acquiring can start before all the buffers have been sent.

The fourth state occurs when the buffer_offset is less than send_start. This occurs when the buffer_offset has wrapped around the buffer queue. The system just continues sending the next available buffer.

Chapter 6 Digital Signal Processing (DSP)

BDAS implements a basic set of filters. In basic EMG processing, this set should be adequate. This section summarizes the filter design that is used in BDAS. More sophisticated filters can be incorporated in BDAS as was shown in section 4.

A discrete time signal is a sequence of numbers x[.], where x[n] denotes the n'th value of the sequence. Most signals from the real world are continuous signals. By taking samples of a continuous signal $x_a[t]$ at a certain interval T, a sequence can be obtained.

$$x[n] = x_a[nT]$$

Discrete signal processing involves transforming an input sequence input[n] to a output sequence out[n] using a transfer function T.

$$out[n] = T\{input[n]\}$$

BDAS uses digital signal processing to attenuate noise, eliminate undesirable frequency components, display frequency spectrum, quantify signal and other processing.

6.1 Digital Filtering

Frequency selective filters let certain frequencies pass and attenuate others. There are two major type of filters namely Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filter. The impulse response identifies the type of filter. The design goal is to obtain a difference equation which can be programmed on a computer. BDAS uses FIR and IIR filters for postprocessing. It uses IIR filters for real-time processing of signals. The IIR filters were used when computational speed was critical.

For Linear Time Invariant (LTI) filters, the transformation from an input sequence x[n] to an output sequence y[n] can be described by a difference equation having the form

$$\sum_{k=0}^{N} a_k y(n-k) = \sum_{r=0}^{M} b_r x(n-r)$$

The goal in filter design is to find a difference equation satisfying the filter requirements.

6.1.1 Finite Impulse Response (FIR) Filters

FIR filters occur mostly in discrete-time implementations. Most FIR designs have a linear phase constraint. One method of FIR design involves specifying the impulse response of the ideal frequency response of the filter we want to design. Usually this impulse response is infinite and noncausal (h[n] > 0 for n < 0). One approach to solve this problem is to truncate the ideal impulse response. This approach is called Impulse Invariance. The truncation of the ideal impulse response in order to get the practical impulse response is equivalent to multiplying the ideal impulse response by a window. One window is the rectangular window which has the value 1 from -a to +a and zero otherwise. The practical impulse response is equal to the values of the ideal impulse response from -a to +a and is zero otherwise.

6.1.1.1 Fourier Series Method

Starting with an ideal frequency response and applying the inverse Discrete Time Fourier Transfer (DTFT), an impulse response is obtained. The impulse response is given by

$$h[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} H(e^{j\omega})d\omega \qquad for \quad n = 0, \pm 1, \pm 2 \ldots$$

where $H(e^{j\omega}) = \sum_{n=-\infty}^{\infty} h[n]e^{-j\omega n}$

The digital filter corresponding to the impulse response h[n] will have the frequency response $H(e^{j\omega})$. There are two problems in implementing this filter. First the sequence h[n] is infinite and second h[n] is noncausal. Truncating the sequence at M samples will alleviate the first problem but will introduce ripples in the stop band. The sequence also needs to be shifted by M/2 samples to make it causal.

One of the filters implemented in BDAS is a band pass filter designed using the Fourier series approximation method. For an ideal band pass filter $H(e^{j\omega}) = 1 \quad for \quad \omega 1 < \omega < \omega 2.$

Taking the inverse DTFT gives

$$h[n] = \frac{1}{2\pi}\int_{\omega 1}^{\omega 2} H(e^{j\omega n})d\omega + \frac{1}{2\pi}\int_{-\omega 1}^{-\omega 2} H(e^{j\omega n})d\omega$$

$$= \begin{cases} \frac{1}{\pi n}(\sin(\omega 2n) - \sin(\omega 1n)) & n \neq 0 \\ \frac{1}{\pi}(\omega 2 - \omega 1) & n = 0 \end{cases}$$

The values for h[n] can be calculated using the above equation. To create a low pass filter make $\omega 1 = 0, \omega 2 = cutoff$. For a high pass filter $\omega 1 = cutoff, \omega 2 = \pi$.

When truncating an impulse response, ripples will occur in the stopband. The simplest form of truncation is the rectangular window which multiplies the sequence by another sequence which is equal to one for M samples and zero for the rest. BDAS gives the choice of window to allow for different filtering requirements.

6.1.1.2 Frequency Sampling

Another FIR filter design method is to specify the desired frequency response and perform an inverse DFT (IDFT). By using an iterative method the coefficients from the IDFT can be modified. The IDFT produces filters with large errors at certain sections of the frequency response. By "spreading" out these errors, the worse case error can be reduced.

Minimizing the maximum error is called the Chebyshev criterion which leads to an equal-ripple approximation. Parks and McClellan have a filter design algorithm based on the Chebyshev criterion. BDAS implements the Parks and McClellan algorithm for FIR filters [Parks, 1987].

6.1.2 Infinite Impulse Response (IIR) Filters

BDAS uses IIR filters for the real-time filtering option during acquisition. IIR filters use less computations for the same order filter than FIR filters.

One method of designing IIR filters is to design a continuous-time filter and then adapt it to the discrete case. The main reason for starting the digital filter design using the continuous case is that the continuous case is well developed and the design procedures have been used and tested for a long time.

6.1.2.1 Design of a notch filter

The 60Hz power line interference is a major source of noise when acquiring biological signals. A notch filter is used to attenuate signals at a specific frequency, in this case 60Hz. In BDAS, a notch filter is available for both real-time and post processing.

The system function of a (LTI) digital filter can be expressed as a ratio of two polynomials with the form

$$H(z) = \frac{(z - z_1)(z - z_2)}{(z - p_1)(z - p_2)} K$$

z_1, z_2 are called zeros because they make *H(z)* equal to zero. p_1, p_2 are called poles, because they make *H(z)* infinite. K is the gain factor. Polynomial coefficients are usually real. Therefore, poles and zeros should occur in conjugate pairs. A second order basic notch filter can be designed so that at 60Hz the value of $H(z) = 0$; therefore, a zero occurs at 60Hz. The sampling frequency determines the value of z for which the zero occurs. If the sampling frequency is 600 Hz then on a normalized frequency scale this is equal to 2π. 60Hz then would correspond to $2\pi \cdot \frac{60}{600} = \frac{\pi}{5}$. At $\frac{\pi}{5}$ is one zero and at $-\frac{\pi}{5}$ is the conjugate pair. The problem with this filter is that it attenuates the frequencies around 60Hz too much and emphasis is added to the signal around π. By adding poles near the zeros, the cutoff near 60Hz is much sharper. The phase response is non-linear around the cutoff frequency. The closer a pole or zero is to the unit circle the larger the change in group delay. Group delay is the derivative of the phase response. It shows how linear the response is.

The notch filter has one zero, one pole and their conjugates. The zero lies on the unit circle. The distance of the pole from the zero determines the 3 dB points of the notch filter. The radius of the circle is 1 and the distance of the pole from the unit circle is r then distance from the **zero** is 1 - r. 1 - r represents the magnitude of the pole when the system function is equal to zero, this is the response peak value of the pole. The 3 dB points occur when the response is $\frac{1}{\sqrt{2}}$ of the peak value. The 3 dB point occurs when the distance from the unit circle is $\sqrt{2}.(1-r)$. Assuming the distance d, between a 3 dB point and the zero, on the unit circle is straight (valid if distance is small) then d equals 1 - r. The bandwidth of the notch filter is then 2d. The circumference is equal to 2π; therefore, $\frac{d}{2}\pi$ *radians* can be used to calculate the pole position (1 - r). In summary, if the notch pole frequency is ω_n, then

$$\frac{2.(1-r)}{2\pi} = \omega_n$$

[Lynn, 1989]

Given a relative notch pole frequency ω_n then r can be calculated. r can be substituted in the system function and the system response *H(z)* can then be calculated. Next the difference equation can be derived from *H(z)*.

Some assumptions are, the response peak is due to the pole only, not its conjugate or the zeros. d on the unit circle is straight, this is reasonable if r > 0.9.

The system function is

$$H(z) = \frac{(z - z_1)(z - z_2)}{(z - p_1)(z - p_2)}$$

The frequency response is obtained by substituting $z = e^{j\omega}$ and $p = e^{j\omega}$

$$H(e^{j\omega}) = \frac{(e^{j\omega} - z_1)(e^{j\omega} - z_2)}{(e^{j\omega} - p_1)(e^{j\omega} - p_2)}$$

The difference of two complex numbers such as $(e^{j\omega} - z_i)$ or $(e^{j\omega} - p_i)$ can be expressed in polar form

$$(e^{j\omega} - z_i) = |e^{j\omega} - z_i| e^{j\alpha_i \omega}$$

where

$e^{j\omega}$ is on the unit circle. The magnitude of the distance of the poles or zeros from the unit circle will determine the frequency response. The angle between the horizontal starting at the pole or zero and the unit circle will determine the phase response.

The frequency response can be summarized to be

$$|H(e^{j\omega})| = \frac{\prod distances\,zeros\,are\,from\,unit\,circle}{\prod distances\,poles\,are\,from\,unit\,circle}$$

The phase response is equal to the sum of the angles from zeros to unit circle minus sum of the angles to poles on the unit circle.

A zero causes a dip in the frequency response at the zero's frequency. A pole causes a peak at the pole's frequency. Poles and zeros can cancel each other out. In the notch filter case, for frequencies away from the notch frequency ω_n the pole and zero cancel each other giving a flat response. When the frequency equals the ω_n, the notch frequency, the zero makes the frequency response zero.

The zeros are $z_1 = e^{j\omega_s}$ and $z_2 = e^{-j\omega_s}$

The poles are $p_1 = Pe^{j\omega_s}$ and $p_2 = Pe^{-j\omega_s}$ where ω_s is the normalized sampling frequency and P is the pole magnitude.

Substituting in

$$H(z) = \frac{Y(z)}{X(z)} = \frac{(z - z_1)(z - z_2)}{(z - p_1)(z - p_2)}$$

gives

$$H(z) = \frac{(z - e^{j\omega_n})(z - e^{-j\omega_n})}{(z - Pe^{j\omega_n})(z - Pe^{-j\omega_n})} = \frac{z^2 - z(e^{j\omega_n} + e^{-j\omega_n}) + 1}{z^2 - zP(e^{j\omega_n} + e^{-j\omega_n}) + P^2}$$

$$= \frac{z^2 - 2zA + 1}{z^2 - 2zPA + P^2}$$

where $A = \cos(\omega_s)$.

Using the z-transform time shift property to convert to the time domain yields

$$y[n+2] - 2PAy[n+1] + P^2 y[n] = x[n+2] - 2Ax[n+1] + x[n]$$

Subtracting 2 from bracketed items (valid because system Linear Time Invariant (LTI)) and rearranging, we obtain

$$y[n] = 2PAy[n-1] - P^2 y[n-2] + x[n] - 2Ax[n-1] + x[n-2].$$

Given any sampled waveform, its sampling frequency, notch filter frequency and notch bandwidth, the coefficients for the above difference equation can be calculated. The impulse response can be obtained by determining the response to a unit impulse.

6.1.2.2 Analog Butterworth Filter

Design of the IIR filters in BDAS starts with an analog design to determine the transfer function $H_a(s)$. In the case of Butterworth filters, the frequency response is

$$|H_n(j\Omega)|^2 = \frac{1}{1+\left(\frac{\Omega}{\Omega_c}\right)^{2n}}$$

BDAS uses Butterworth filters because they have a flat passband.

By applying transformations on a normalized low pass Butterworth filter high pass, bandpass, and low pass filters can be generated.

The frequency response of an analog filter is obtained by letting $s = j\Omega \quad \therefore \quad \Omega = \frac{s}{j}, \Omega_c = 1$. This yields

$$H_n(s)H_n(-s) = \frac{1}{1+\left(\frac{s}{j}\right)^{2n}}$$

Ω is the analog frequency variable. ω is the analog frequency variable.

The roots of the denominator determine the poles. The roots are

$$p_k = (-1)^{\frac{1}{2n}}(j\Omega_c) = \Omega_c e^{\left(\frac{j\pi}{2n}\right)(2k+n-1)}, \quad k = 0, 1, \ldots, 2n-1$$

The poles of the magnitude square function always occurs in pairs: with one pole of a pair in the left s plane and the other in the right half s plane. Therefore, in order for the filter to be stable and causal the LHS (Left Hand Side) poles are chosen for the function $H_a(s)$ and the RHS poles are chosen for $H_a(-s)$.

In the case of a normalized Butterworth filter of order 2 the transfer function $H_2(s)$ has poles at angles $\frac{\pi}{4}, 3\frac{\pi}{4}, 5\frac{\pi}{4}, 7\frac{\pi}{4}$. The poles at $3\frac{\pi}{4}, 5\frac{\pi}{4}$ are chosen for $H_2(s)$ and the poles at $\frac{\pi}{4}, 7\frac{\pi}{4}$ are chosen for $H_2(-s)$. The transfer function becomes

$$H_2(s) = \frac{1}{(s-p_1)(s-p_2)}$$

$$= \frac{1}{s^2 + \sqrt{2}s + 1}$$

6.1.2.3 Impulse Invariance

Impulse invariance consists of sampling the impulse response of a continuous time system. Since continuous time filter design is highly advanced, there exists many design procedures.

When designing a filter using impulse invariance the continuous time system function *H(s)* is needed. Once this function is obtained, the continuous time impulse response $h_c(t)$ is obtained. Sampling the continuous time impulse response finally gives the discrete time impulse response.

$$h[n] = T_d h_c(nT_d)$$

The system function *H(z)* is the z-transform of h[n].

The generalized form of a second-order filter transfer function is

$$H(s) = \frac{a_1 s + a_0}{b_2 s^2 + b_1 s + b_0}$$

$$H(z) = \sum_{n=0}^{\infty} Ke^{pnT}z^{-n} + \sum_{n=0}^{\infty} Le^{qnT}z^{-n}$$

The geometric series sum to

$$H(z) = \frac{K}{1 - e^{pnT}z^{-1}} + \frac{L}{1 - e^{qnT}z^{-1}}$$

Combining the partial fractions gives

$$H(z) = \frac{(K+L) - (Ke^{pT} + Le^{qT})z^{-1}}{1 - (e^{pT} + e^{qT})z^{-1} + e^{(p+q)T}z^{-2}} \quad .$$

Let

$$f_2 = 0, \quad f_1 = -(Ke^{pT} + Le^{qT}), \quad f_0 = K + L$$

$$g_2 = e^{(p+q)T}, \quad g_1 = -(e^{pT} + e^{qT}), \quad g_0 = 1$$

Using time-shift property of z-transform gives the difference equation,

$$y[n] - g_1 y[n-1] + g_2 y[n-2] = f_0 x[n] + f_1 x[n-1]$$

Using the values calculated for p, q, K, and L the difference equation can be derived.

In the previous section, it was established that a Butterworth filter of order two has the system transfer function

$$H_2(s) = \frac{1}{s^2 + \sqrt{2}s + 1} \quad .$$

Therefore, in this case

$$a_0 = 1, \quad a_1 = a_2 = 0$$

$$b_0 = 1, \quad b_1 = \sqrt{2} \quad b_2 = 1$$

The normalized transfer function to general low-pass transfer function transformation is

$$s \rightarrow \frac{s}{f_r}$$

where

$$f_r = \frac{f_c}{f_s}.2\pi = \frac{cutoff \quad frequency}{sampling \quad frequency}.2\pi$$

in the case of the 2nd order Butterworth filter, the coefficients change to

$$b_2 = \frac{1}{f_r^2}$$

$$b_1 = \frac{\sqrt{2}}{f_r}$$

6.1.2.4 Bilinear Transformation

Aliasing occurs when a high frequency signal is misinterpreted as a low frequency signal. This can happen when the highest input signal frequency is greater than half the sampling rate. Bilinear transformation avoids the aliasing problems of the other methods although it includes a possible problem of it's own. The transformation distorts the frequency axis by mapping the continuous frequency axis, which is infinite, to the unit circle in the Z plane. $H_a(s)$ denotes the analog filter and H(z) denotes the digital filter. The bilinear transformation consists of replacing s by

$$s = \frac{2}{T_d}\left(\frac{1 - z^{-1}}{1 + z^{-1}}\right)$$

The relationship between ω and Ω can be derived by substituting $z = e^{j\omega}$ in the above equation giving

$$s = \frac{2}{T_d}\left(\frac{1-e^{-j\omega}}{1+e^{-j\omega}}\right)$$

$$s = \sigma + j\Omega = \frac{2}{T_d}\left(\frac{2e^{-j\frac{\omega}{2}}\left(j\sin\frac{\omega}{2}\right)}{2e^{-j\frac{\omega}{2}}\left(\cos\frac{\omega}{2}\right)}\right)$$

$$= \frac{2j}{T_d}\tan\left(\frac{\omega}{2}\right)$$

$$\therefore \sigma = 0,$$

As θ varies from 0 to π, the magnitude varies from zero to infinity [Opp].

The 2nd order Butterworth filter has the form

$$\frac{1}{as^2 + bs + 1}$$

substituting $s \rightarrow \dfrac{2(1-z^{-1})}{(1+z^{-1})}$

gives

$$H(z) = \frac{(1+2z^{-1}+z^{-2})}{z^{-2}(2a-2b+1)+z^{-1}(2-4a)+(2a+2b+1)}$$

The difference equation is,

$$y[n] = \frac{x[n]+2x[n-1]+x[n-2]-(2a-2b+1)y[n-2]-(2-4a)y[n-1]}{(2a+2b+1)}$$

In the case of a second order Butterworth filter the transformations from normalized low pass to general low pass is the same as was stated in the discussion on Impulse Invariant filters. Therefore,

$$a = \frac{1}{f_r^2}$$

$$b = \frac{\sqrt{2}}{f_r}$$

where f_r = relative cutoff frequency.

6.2 Rectification

Most of the signals acquired are bipolar (i.e values above and below zero). As mentioned previously, the "amount" of a signal can be calculated by integrating over a period of time. Unfortunately, this would lead to a zero value due to the bipolar nature of the signal. Some of the ways of getting just a positive signal are half-wave rectification, full-wave rectification and RMS. Each of these ways introduces non-linear distortion. BDAS offers full wave rectification and integration.

6.3 DFT and FFT

Frequency analysis is an important part of signal processing. The Discrete Fourier Transform (DFT) is the discrete-time equivalent of the continuous-time Fourier transform. It computes the "amount" of a signal at discrete frequency points. The Fast Fourier Transform (FFT) is a computationally-efficient algorithm that computes the DFT.

The DFT is defined as

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk} \quad \text{for k= 0 to N-1}$$

where

$$W_N = e^{\frac{-j2\pi}{N}}$$

A total of N^2 complex multiplications and N^2 complex additions are needed to compute an N-point DFT. FFT takes advantage of the symmetry of W_N to reduce the number of complex multiplications substantially [Oppenheim, 1989].

The two basic types of FFT are Decimation In Time (DIT) and Decimation In Frequency (DIF). They both divide the problem into two smaller problems. These then get subdivided until the FFT is computed on a simple case of two numbers. This simple case is called the butterfly. The FFT consists of stages (subdivisions) which are made up of groups of butterfies.

BDAS implements the FFT using a DIT algorithm. A floating point and fixed point version was developed. Many PC's don't have a floating-point coprocessor which would slow floating-point calculations dramatically. The floating point version is slower but produces a smoother output because of less roundoff errors when compared to integer versions.

6.3.1 Overflow

When using integer formats, the butterfly calculation may cause overflow. Some methods to eliminate overflow are input data scaling, unconditional block floating-point, and conditional block floating-point. Input data scaling reserves a set number of bits for overflow. This method is the fastest of the three since no manipulation is needed except for a final scaling at the end. The problem is that depending on the size of the FFT and the number of bits in each input word there may not be enough bits available.

Block floating point is a mix between fixed-point and floating-point number representations. The number is represented in fractional form with the implied decimal point at the left. There is an exponent for the whole group of numbers. As long as the whole group of numbers stays within a range determined by the number of bits available, this representation is a very good compromise between floating-point and integer. Conditional block floating-point only scales when absolutely necessary. It is slightly slower and produces marginally better results.

BDAS uses unconditional block floating point which s 'es the output after each iteration. In order to speed up calculations memory is sacrificed for speed. Look-up tables are used whenever possible. Sin, Cos, and bit reverse tables are used. When the system first starts up, these tables are generated for the largest FFT, 1024 points.

6.4 Fixed point arithmetic

In order speed up calculations, fixed-point 2's complement arithmetic is used. The binary point is implied in the fixed-point format. The binary point can affect quantization errors and can cause distortion due to overflow. The binary point determines the range of numbers that can be stored without overflow and the precision these numbers can have. A fixed-point binary number can be referred to as W-n format where W is the word length in bits and n is the number of bits after the binary point. For a 16-bit word with 10 bits after the binary point, the format is 16-10. The range of numbers is -32 to +31. When adding or subtracting using a W-n format, the W-n format does not change. The only problem that can occur is overflow. When multiplying a W-n by a W-m format, the result is 2W-(m+n). When the result is converted back to W-n format, quantization error occurs. These errors can be minimized by using 2W format for as long as possible, and only after many multiplications have been added together. Converting a floating-point number to W-n format is simply multiplying by 2^n which is done in a CPU by a simple shift. BDAS uses a 16-16 format, a fully-fractional representation. A 32-bit long integer format is used in intermediate steps to minimize roundoff errors.

Chapter 7 Front End Hardware

In the overview on BDAS, two versions of the hardware were briefly described. The hardware front end is common to both versions. This chapter deals with the common hardware portions. Figure 7.1 shows the major building blocks for the front end.

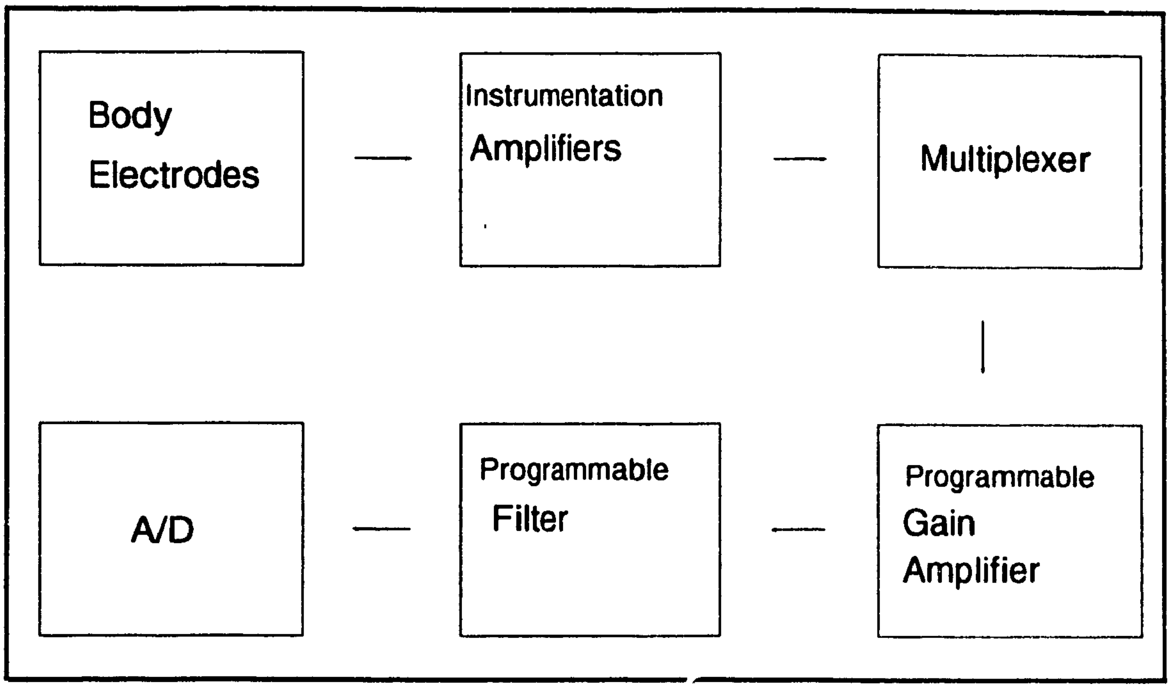

Fig 7.1 Front End Overview

7.1 Operational Amplifiers

The front end for the A/D subsystem is composed of instrumentation amplifiers, filters, and gain amplifiers. The major building block for these units is the operational amplifier (op amp).

7.1.1 Ideal Op Amp

The ideal op amp is a very simple three terminal device. Two of the terminals are inputs v_+, v_- and the third is the output v_{out}. They are related by

$$v_{out} = A(v_+ - v_-)$$

where A is the loop gain.

In an ideal op-amp the current into the input terminals is zero. This gives an infinite input impedance. The bandwidth is from zero to infinity with constant gain A. A is infinity.

Using the ideal definition of an op amp

$$v_{out} = A(v_+ - v_-)$$

$$(v_+ - v_-) = \frac{v_{out}}{A} \approx 0$$

7.1.2 Non Ideal Characteristics

Actual op-amps are non-ideal. Non ideal characteristics are the slew rate, common mode gain and offset voltage. This a low frequency application therefore, high frequency op-amp problems are not significant.

7.1.2.1 Slew Rate

Slew rate is the maximum possible rate of change of the op-amp output voltage. Slew rate limiting will cause non-linear distortion. It is usually specified in V/us. The higher the slew-rate, the larger the bandwidth.

$$SlewRate = S = \left(\frac{\delta V}{\delta t}\right)_{max}$$

An op amp has internal capacitances. These capacitances will need to be charged and discharged. They therefore limit that rate of change of the output.

Slew rate differs from frequency response in that slew rate measures large signal distortion whereas frequency response is a measure of small signal gain over frequency. Slew rate is a function of frequency and output voltage. For instance, given a sinusoidal signal

$V = V_p \sin(2\pi f t)$.

where V_p is the peak voltage and differentiating gives

$$\left(\frac{\delta V}{\delta t}\right) = 2\pi f V_p \cos(2\pi f t) \quad .$$

The maximum occurs when $\cos(2\pi f t) = 1$. Therefore

$$\left(\frac{\delta V}{\delta t}\right)_{max} = 2\pi f V_p = S$$

In the BDAS system, one of the op amps used is the TL084 which is rated at 13 V/us. The peak voltage is 2.5 Volts. Using this information, the maximum frequency before distortion occurs because of slew rate is

$$S = 2\pi f V_p$$

$$f = \frac{S}{2\pi V_p} = \frac{13V/us}{5\pi V} = 827kHz$$

The TL064 has a slew rate of 6 V/us which is a little less than half the slew rate of the TL084. The maximum frequency for the TL064 is 382 kHz. Since the maximum frequency in the BDAS system is 5 kHz both these op amps will be adequate.

7.1.2.2 Common-Mode Gain

If the two input terminals of an op amp are tied together and a signal is applied to the input, the output ideally should be zero. The ratio of the common-mode input voltage to output voltage is the common-mode gain. Practical op amps, however, have a nonzero *Common Mode Gain*. The measure of ability to reject common-mode signals is expressed in the *Common Mode Rejection Ratio* (CMRR) which is usually specified in dB. When acquiring bio-signals, noise is a major problem. The bio-signal of interest can be very small when compared to the noise signal. High CMRR is needed to reduce the noise signal and extract the signal of interest. BDAS amplifier characteristics concerning CMRR are discussed in 7.2

7.1.2.3 Offset Voltage

Offset voltage occurs when both inputs are tied to zero and the output is nonzero. This voltage occurs due to non ideal characteristics of the differential amplifier input stage. Bias current is the current needed to drive the input stage of the op amp. The offset current is the difference in the bias currents. Since the instrumentation amplifiers are capacitively coupled, there is no problem with offset voltage.

7.2 Instrumentation Amplifiers

The basic amplifier block used in acquiring bio-signals is an Instrumentation Amplifier (IA). The IA amplifies the difference between the two inputs and rejects signals common to both inputs. This property allows for , mall signals in a noisy environment to be amplified with the noise being rejected.

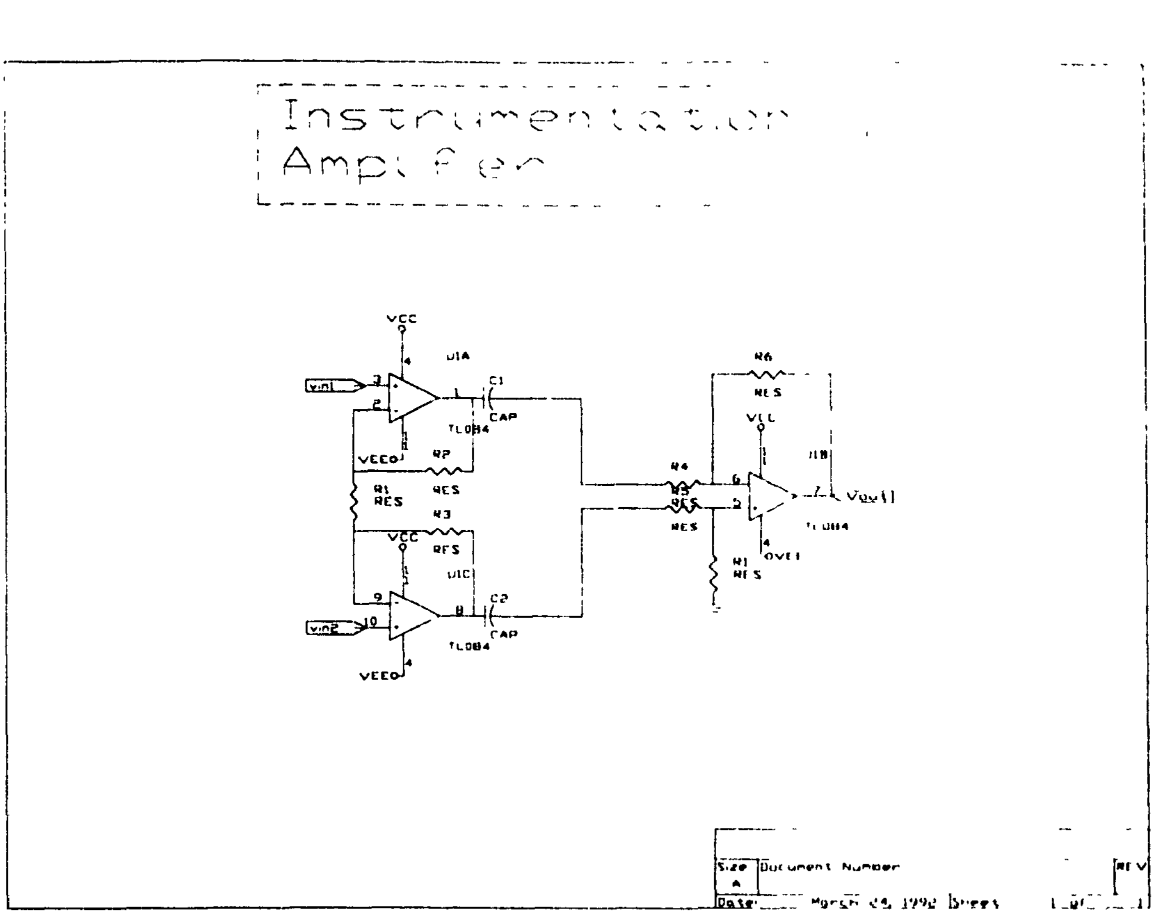

Fig 7.2 Instrumentation Amplifier

The main component of an IA is the differential amplifier. The problem with the differential amplifier is the input impedance is low. One solution is to add voltage followers to the inputs of the differential amplifier. The voltage followers should have high input impedance, due to a JFET first stage. The buffers in front of the differential amplifier are connected in such a way as to provide gain.

The TL064 op-amp was chosen with the key criterion being the input impedance. The TL064 has very high input impedance on the order of $10^{12} ohms$. Other criteria were cost and availability. The TL0XX series is widely available and with minimal cost.

vin1 and vin2 are across r1. The voltage v_{diff} at the inputs of the differential amplifier is calculated as follows:

The current through r3 is

$$I_{r1} = \left(\frac{v_{in1} - v_{in2}}{r1} \right)$$

$$V_{out1} = v_{in1} + I_{r1} r2$$

$$V_{out1} = v_{in1} + \left(\frac{v_{in1} - v_{in2}}{r1} \right) r2$$

$$V_{out1} = v_{in1}\left(1 + \frac{r2}{r1} \right) - v_{in2}\left(\frac{r2}{r1} \right)$$

similarly

$$V_{out2} = v_{in2}\left(1 + \frac{r3}{r1} \right) - v_{in1}\left(\frac{r3}{r1} \right)$$

The input of the differential amplifier then becomes, (assume r2 = r3)

$$V_{diff} = V_{out2} - V_{out1}$$

substituting

$$V_{diff} = \left(1 + \frac{2r2}{r1} \right)(V_{out2} - V_{out1})$$

A mismatch between r2 and r3 leads to gain error but does not reduce CMRR which is crucial. The minimum gain once adjusted in BDAS is usually fixed unless the system is used for signals with much different voltage levels than EMG or ECG. For example, EEG would need higher gain front ends due to its very-low signal levels. In BDAS the resistor r1 is variable so that the gain can be modified.

In order to calculate the CMRR the same signal will be used for differential-mode input and common-mode input. Figure 7.3 shows the $\pm 10mV$ input signal.

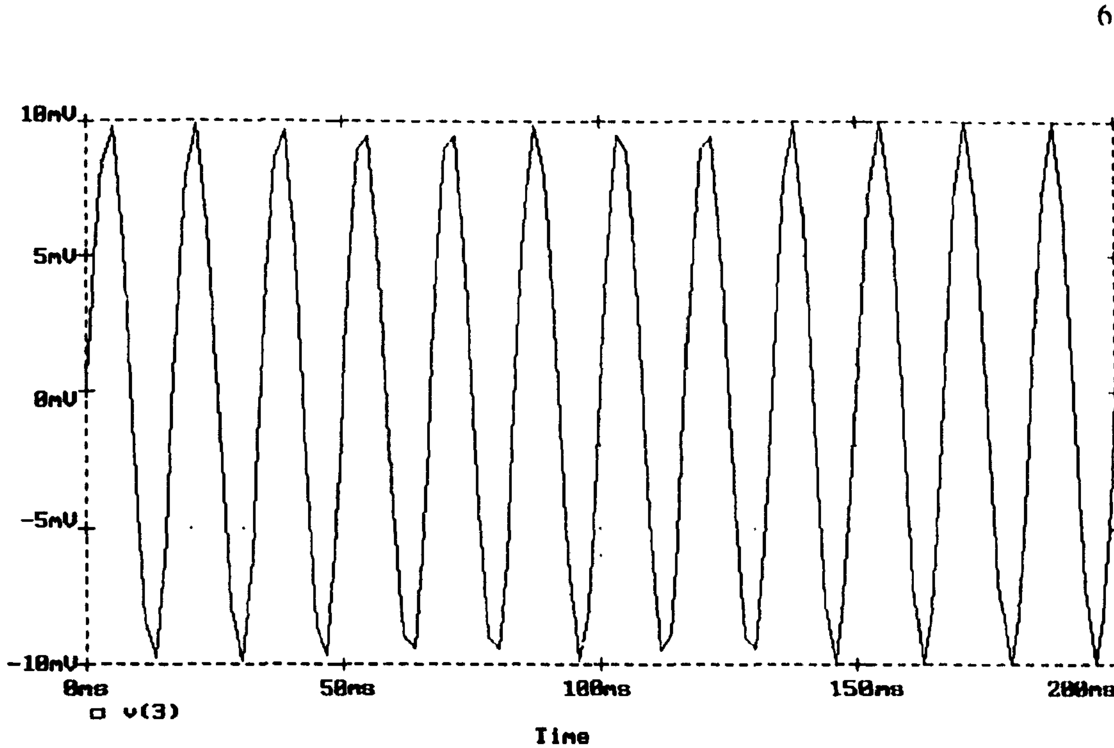

Fig 7.3 Input to IA

Figure 7.4 shows the output of the instrumentation amplifier when the input is applied differentially. In BDAS, the gain is set at 100 giving, in this case, an output of $\pm 1\,V$. Figure 7.5 shows the output of the instrumentation amplifier when the input signal is applied to both inputs of the amplifier. The output is $\pm 5uV$ with a DC offset of $11uV$

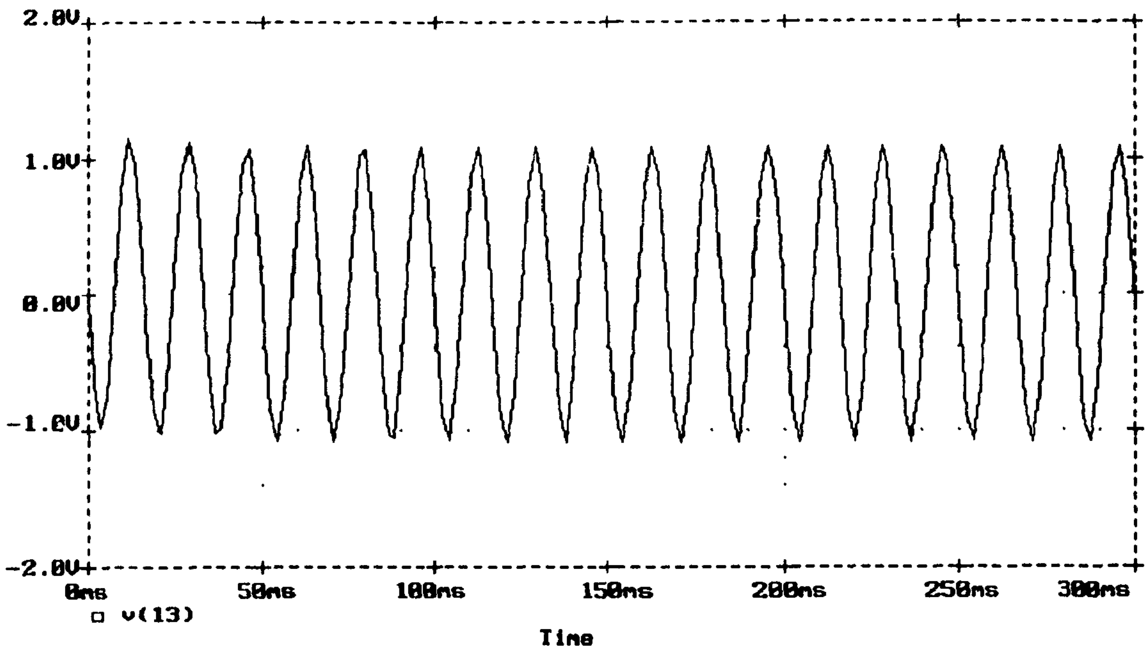

Fig 7.4 Output of Differential Input

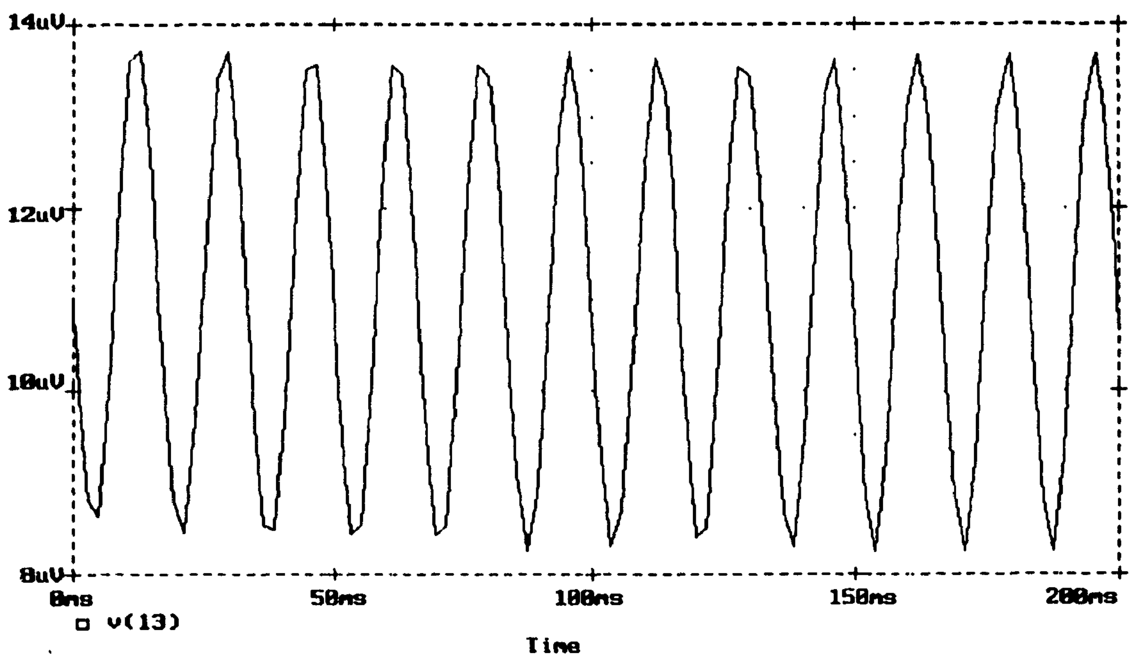

Fig 7.5 Output of Common Mode Input

The CMRR is defined as $CMRR(dB) = 20\log\left(\frac{Differential\_gain}{Common\_mode\_gain}\right)$

The differential gain in this case is 100. The common mode gain is 0.005. Therefore $CMRR = 20\log(100/0.005) = 86dB$

Figure 7.6 shows a 1 mV at 10 Hz signal riding on top of a noise signal of 100 mV at 60 Hz. Figure 7.7 shows the output of the noisy signal, which is now 10Hz at around 100mV (differential gain of 100). The instrumentation amplifier has effectively extracted the smaller input signal from the larger noisy signal.

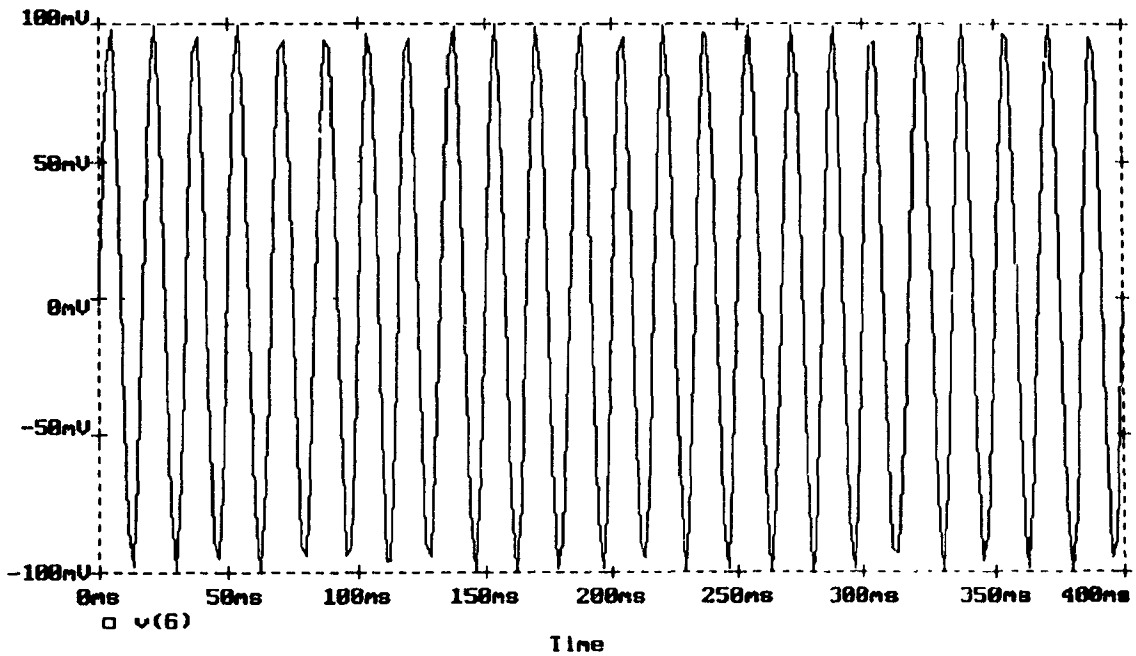

Fig 7.6 Signal plus Noise Input

If a $\pm 2.5mV$ signal is input into the instrumentation amplifiers the maximim noise that can be tolerated is calculated as follows. The 12 bit resolution divides the signal into 2048 steps. For a maximum input signal (to the gain and filter amplifiers) of 250 mV, the step

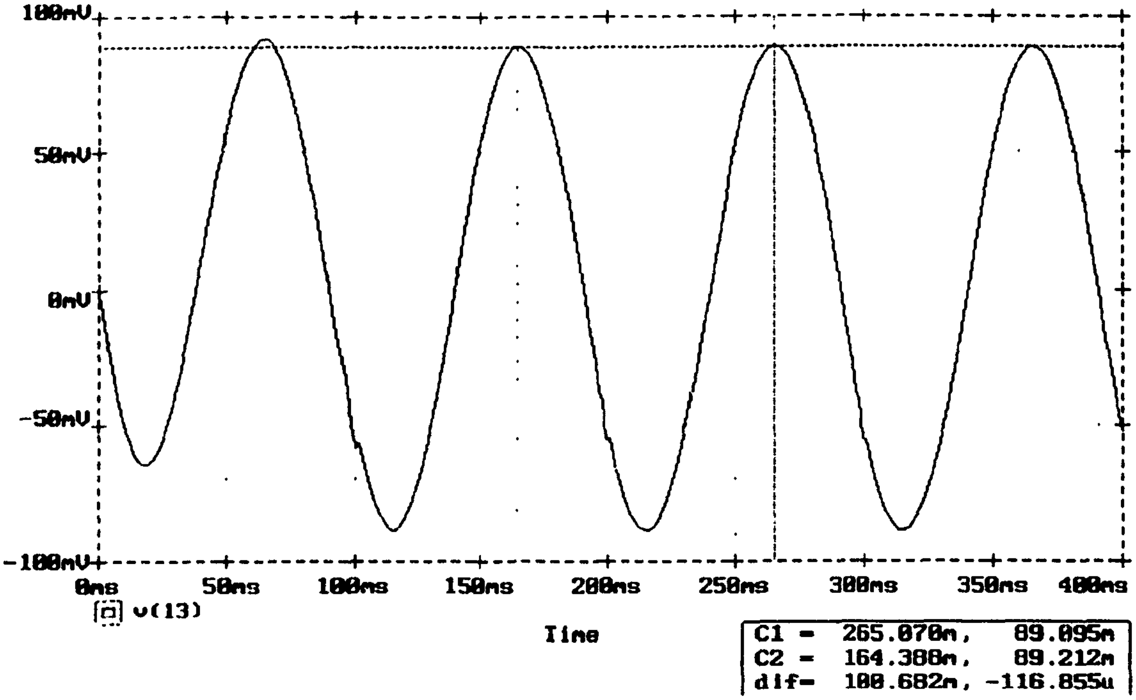

Fig 7.7 Output of Noisy signal

size is around 122 uV. The noise will be a problem at half the step size namely 61 uV. The common-mode rejection of the signal allows the noise to be 2000 times or 122 mV before affecting the output signal.

7.3 Programmable Gain Amplifier (PGA)

The Programmable Gain Amplifier (PGA) uses an inverting gain amplifier as the base. It has the following gain response

$$\frac{v_{out}}{v_i} = 1 + \left(\frac{R_2}{R_1}\right)$$

In BDAS, R2 is fixed and R1 is switch-selectable thus providing different gains (Fig 7.13). There are four switches with the values R2, R2/2, R2/4 and R2/8. By turning on different combinations of switches, different gains can be programmed. The gains range from 2 to 16.

7.4 Isolation Amplifiers

The human body is very susceptible to electrical current. Even currents as low as a few milliamps can be dangerous. In order to avoid hazardous currents if components fail in the data acquisition system, isolation is required for safety reasons.

Isolation means that there exists no direct connection between the subject and the power lines. Should some components fail, the power lines should not short out through the subject. Optocouplers isolate one part of a system from another part by communicating using light (infrared). Transformers can also be used for isolation, since the primary and secondary coils are not physically connected.

The isolated circuits can be powered using one of two methods. The first method is to use a separate battery power source. Another solution consists of using isolation amplifiers which derive their power from the main power source through a transformer. BDAS uses battery power since the remote unit is independent of the PC. This solves the need for special isolation amplifiers. In the remote unit, the connection is through the serial port which needs to be isolated. The serial port is opto-isolated from the remote unit.

In the PC Card version, the input to the A/D is opto-isolated from the A/D chip. The A/D front end runs off battery power.

7.5 Front end filtering (anti-aliasing)

The sampling theorem states that the highest frequency that can be sampled is one half the sampling rate. When frequencies higher than half the sampling rate are sampled, aliasing errors occur. Aliasing will make these higher frequency signals appear as low frequency signals, thus corrupting the data. To avoid aliasing, a front-end low-pass filter is needed. Since low-pass filters do not have ideal cutoff characteristics, the sampling rate should be increased to more than twice the highest frequency. The amount of increase depends on the actual implementation of the front end filter. In BDAS, the recomended sampling rate is 2.5 times the maximum frequency.

BDAS has different options to solve the aliasing problem. The human body, which is the signal generator in this case, bandlimits the bio-signals it generates. If only one channel is being used then the programmable front-end Sallen and Key filter can be used. If more than one channel is used, the filter CANNOT be shared. It must be disabled and the filtering done separately for each channel. In the case of EMG, frequencies above 500 Hz have power less than 1/1000 the maximum value of the EMG spectrum [Mathieu, 1990]. If sampling is done at over 1000 Hz, then the corruption caused by the signals above 500 Hz is minimal. One solution to anti-aliasing, therefore, is to sample at a high rate without the need for an anti-aliasing filter. If frequencies higher than half the sampling rate are present, then BDAS can have fixed filters on each of the instrumentation amplifiers. Presently three out of the four channels have a 200 Hz cutoff filter. This can easily be changed by replacing the feedback capacitor on the differential amplifier in the IA.

7.5.1 Low Pass filters

The system contains programmable low pass filters. The need for different low pass filters is due to the variability of the sampling rate. When the user choses a sampling rate, the appropriate filter is automatically chosen. This can be overridden when there is a need for oversampling. The low-pass filters are of the Sallen-Key variety. By using the 4051 digital controlled analog switch, different combinations of resistors and capacitors can be chosen which in turn specify the filter characteristics. An easier implementation would be to use switched capacitor filters (SCF) and control the cutoff using the clock rate. Each of the channels could have a SCF after the IA and before the gain amp. Unfortunately, the availability necessitated an alternate design with available components.

The front-end programmable low-pass filter is a two-pole Sallen and Key equal component filter (Fig 7.9). The equal component Sallen and Key filter was chosen because of ease of design and availability of components. A switched capacitor filter would be more practical and have less components. The higher noise of the switched capacitor filter is not a problem, since the signal has been previously amplified. Unfortunately, the switched capacitor filters were unavailable at implementation time. The Butterworth filter has a flat passband and stopband which is what is needed in this application. A second-order Butterworth filter has a damping factor $\alpha = 1.414$ and frequency ratio

$$\frac{f_{(3db)}}{f_c} = 1$$

then

$$f_{(3db)} = f_c = \frac{1}{2\pi R C}$$

Since this is an equal-component design, r1 = r2 = R and c1 = c2 = C [Fau]. R_a is chosen arbitrarly and

$$R_b = R_a(2 - \alpha)$$

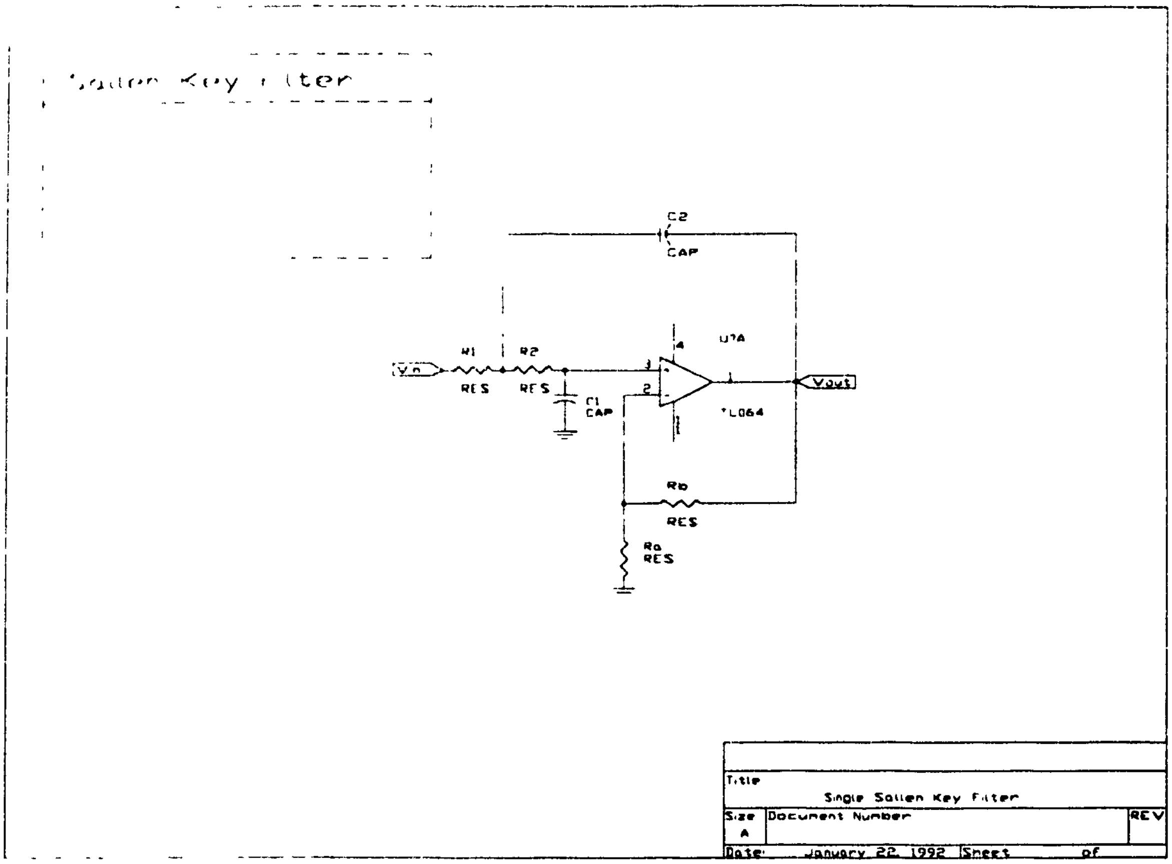

Fig 7.8 Sallen Key filter

The passband gain is fixed by the filter type

$$Gain = \left(\frac{R_b}{R_a} \right) + 1 = 1.59$$

BDAS has a 4-pole sallen-key fixed filter which gives a gain of about 2.5. With the programmable filter this gain becomes about 4.

Figure 7.9 shows a sample front-end filter response

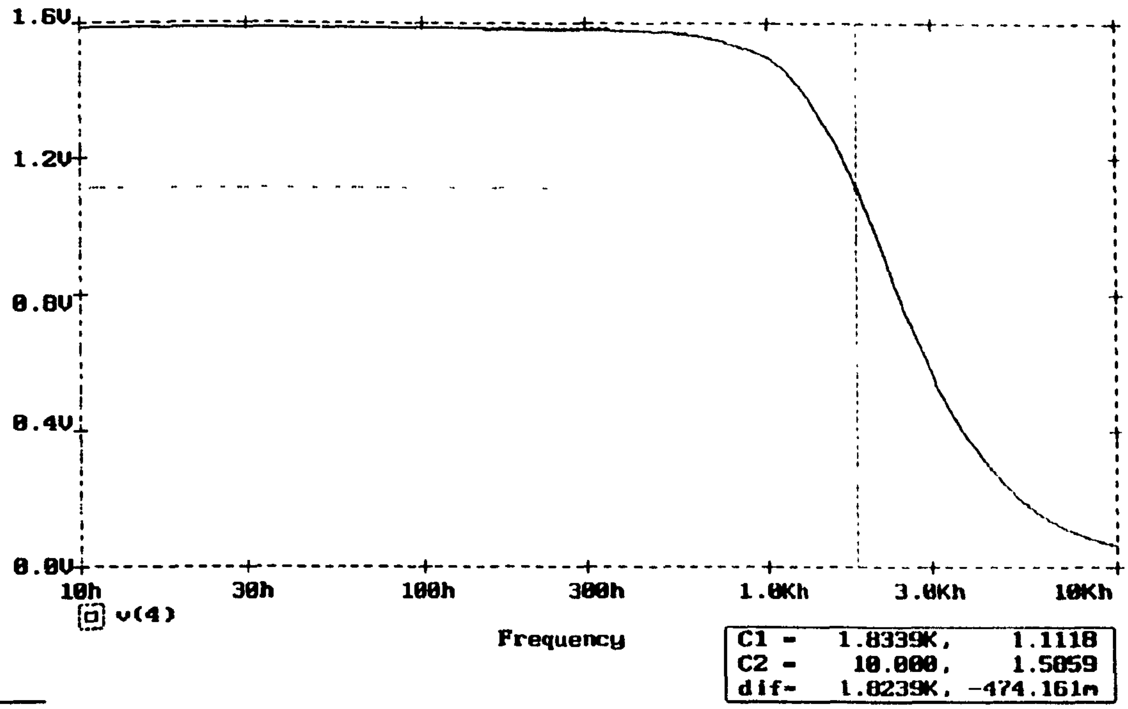

Fig 7.9 Front-End Filter Frequency Response

7.6 A/D Subsystem

Two important parameters that classify A/D converters are conversion time and number of bits. In the case of BDAS, the highest sampling frequency is 8 kHz giving a maximum input frequency of 4 kHz. After the conversion time, the system switches the channel and waits for the gain and filter amplifiers to settle down. Both decrease the maximum conversion time needed. With an 8 kHz sampling frequency, the conversion time is 125 us. Switching of the multiplexer and settling of the gain and filter amplifiers each increases the conversion time by 10 us.

Since the A/D chip is being used in a battery-powered device, power consumption is a major criterion when choosing an A/D chip. The ADC1241 from National is a chip that runs off a +5V power supply and can convert an input signal that ranges from -5V to +5V. The power consumption is 2 mA which is low compared to other chips. The ADC1241 converts 12 bits plus a sign bit. DBAS uses a 12 bit 2's complement number by using the least significant 11 bits and the sign bit making the input range to the A/D chip from -2.5V to +2.5V. Since the signals of interest are bipolar the system has an on board DC to DC converter which converts a +5V input to -5V which is needed by the A/D converter and the front-end subsystem.

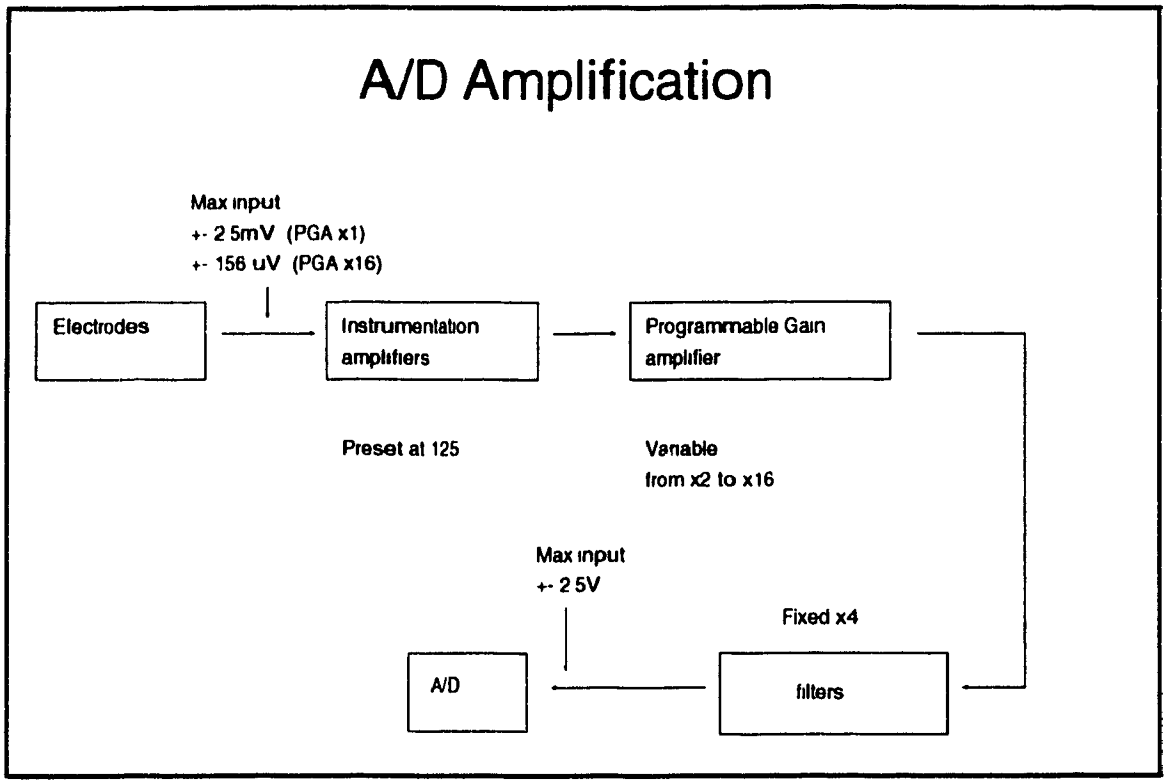

Fig 7.10 A/D preamplification

The biological signals being captured are usually below 1 mv. Since the resolution of the A/D is $\frac{2.5V}{2048} \approx 1mV$, this requires the signals to be amplified. Figure 7.10 gives an overview of the A/D preamplification. The minimum amplification is 1000 which gives a maximum input signal to the instrumentation amplifiers of -2.5 mV to +2.5 mV. Amplification of 1000 gives a minimum resolution of about 1 uV.

7.6.1 Sample and Hold

An A/D converter requires a finite amount of time to digitize a signal. During the conversion the signal should be held constant. Otherwise, errors could occur in the conversion process. The subsystem that performs this task is called the sample and hold.

The sample and hold has two states. The first is sample, which means the output tracks the input. The second is the hold state where the output remains equal to the input the instant the hold signal was applied.

The A/D converter chip fortunately has an internal sample and hold. Therefore, no external sample and hold is needed.

7.7 BDAS front end

The BDAS front end consists of two pc boards. One board contains the instrumentation amplifiers (IA) and charger. It consists of four IA's each having a two input jumper (Fig 7.11). The second board is the multiplexer board (Fig 7.12). It contains the Programmable Gain Amp (PGA) (Fig 7.13), the channel switcher (MUX) and a Programmable Filter (PF) (Fig 7.14).

The IA board connects to the mux board tnrough an 8-pin connector. The mux board connects to the CPU board through a 16-pin connector.

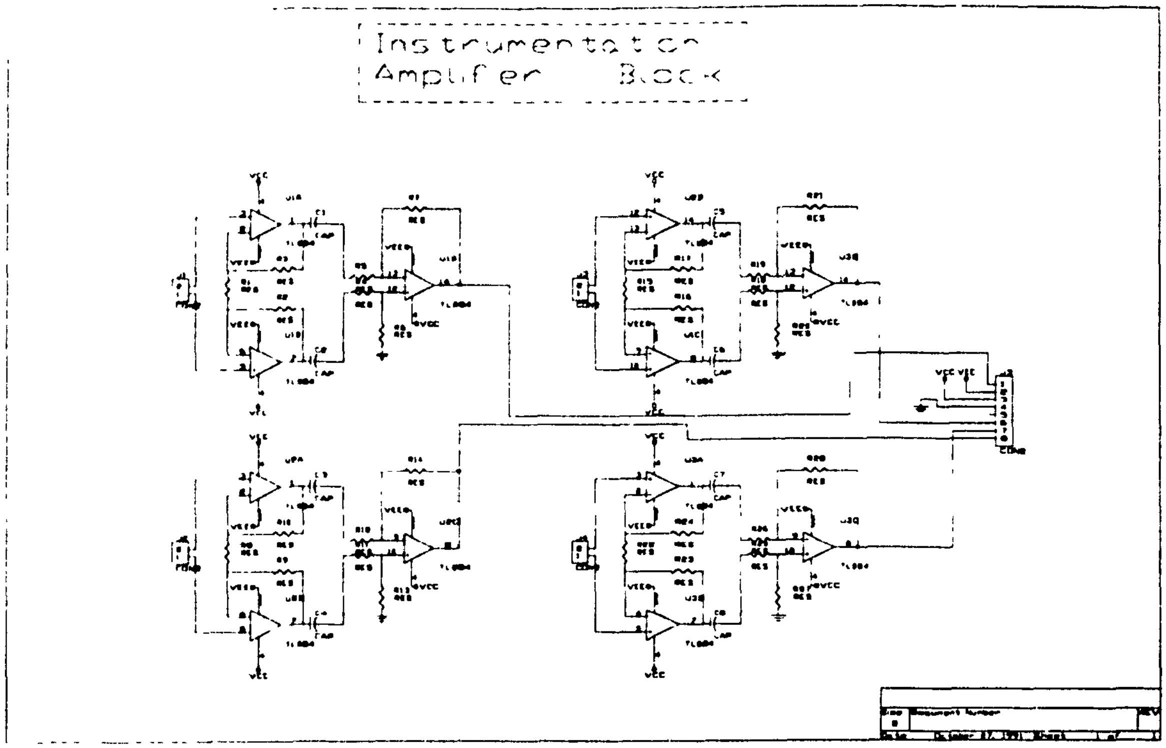

Fig 7.11 IA Board

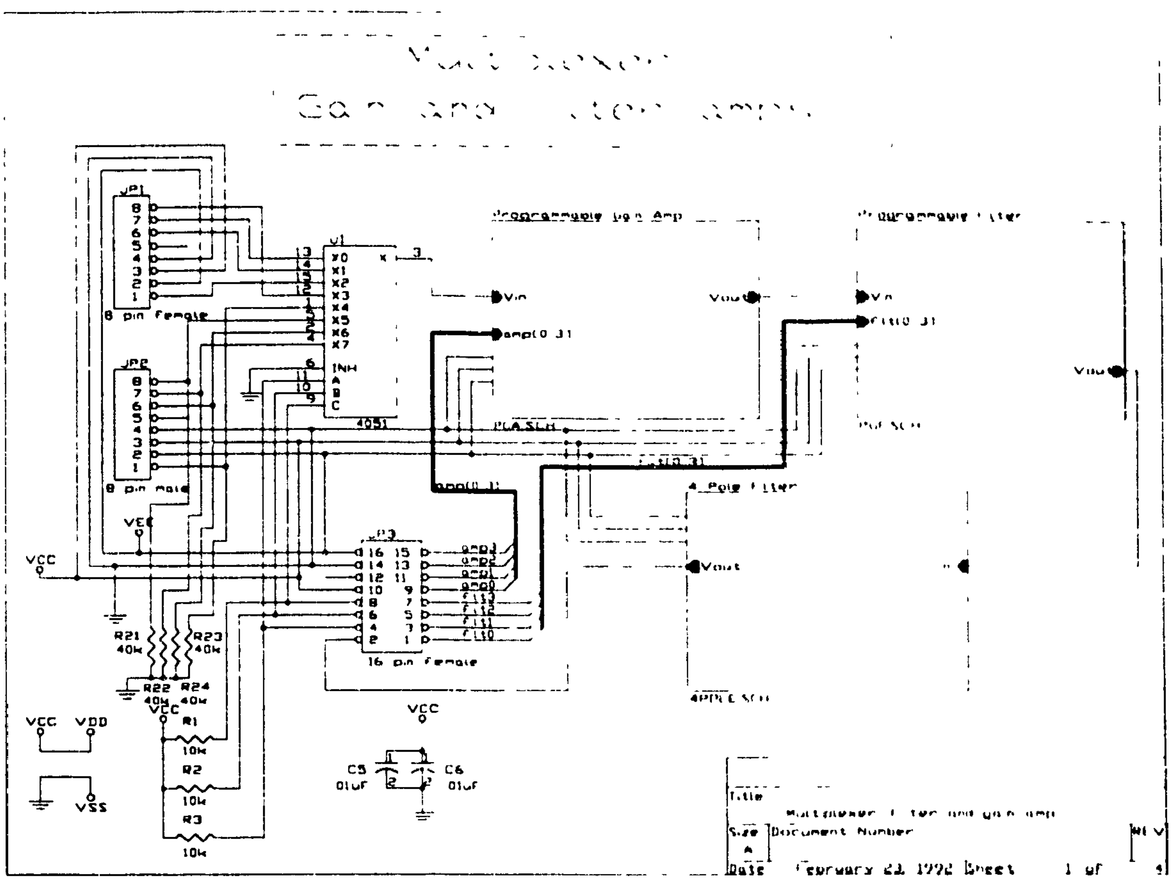

Fig 7.12 Multiplexer Board

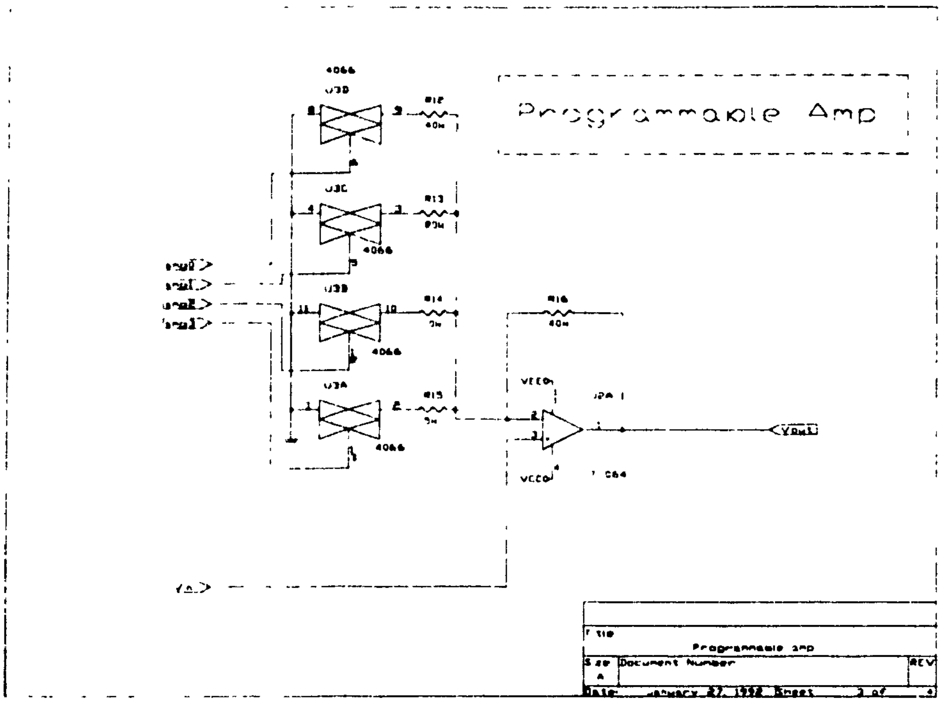

Fig 7.13 PGA Board

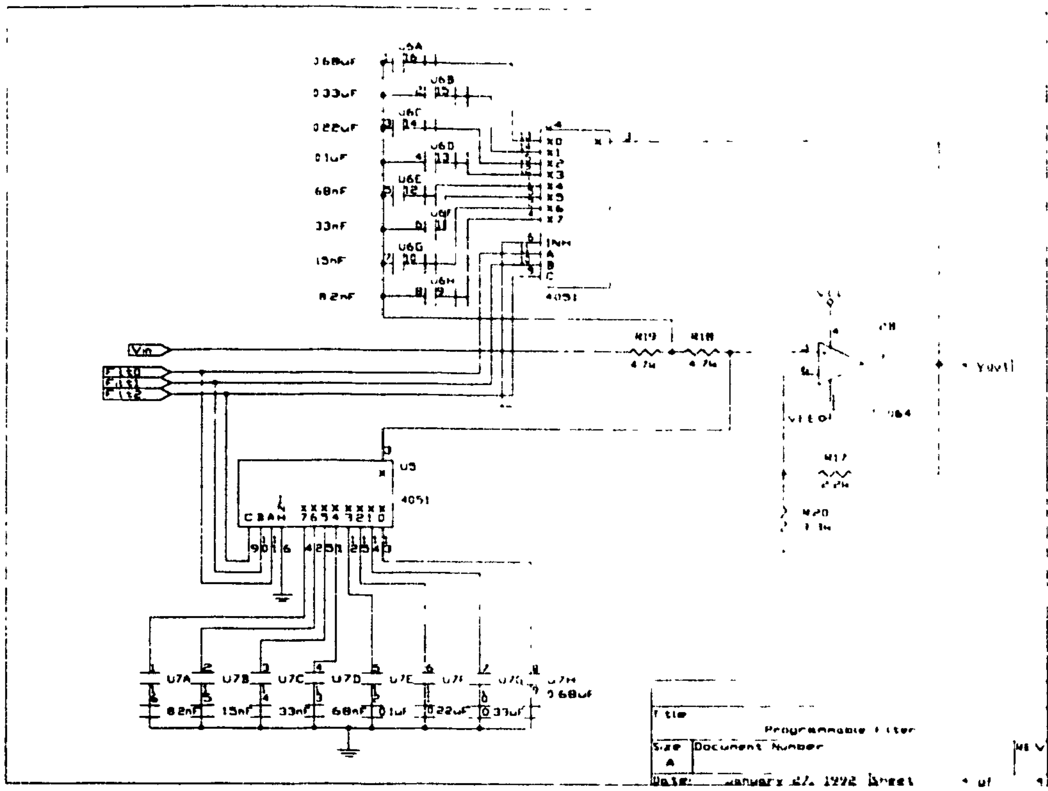

Fig 7.14 PF Board

7.7.1 Charger Subsystem

The portable system runs on rechargable batteries. The voltage that is needed is a min of 4.8V and a maximum of 6.0V. The system uses 4 standard AA batteries which are 1.25V each and have a C = 450 mA-hr rating. The normal charge rate is a C/10 rate which is 45 mA-hrs. Using the C/10 rate, the batteries can be left on charge indefinitely. The charger input is 12V. The diode is used to protect the batteries from discharge. The LED indicates the system is under charge. The resistor limits the current to approximately the C/10 rate. The main switch will switch the battery to the system or the charger. For safety reasons, charging is disabled when the unit is in use. Figure 7.15 shows the charger subsystem.

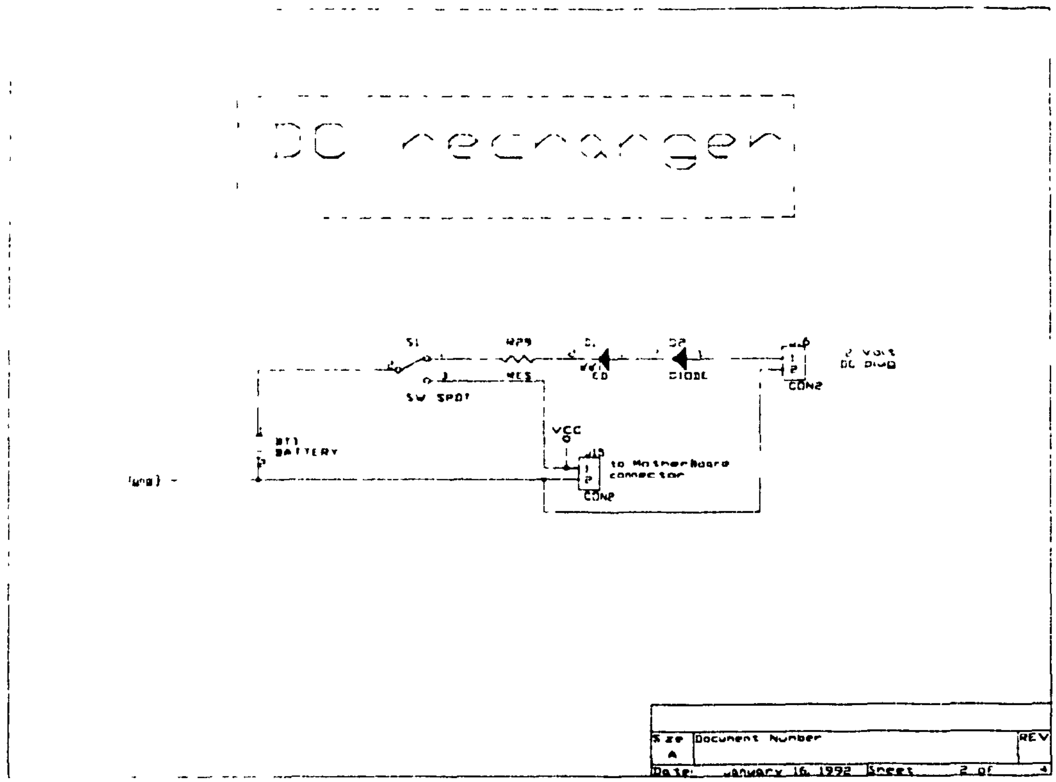

Fig 7.15 Charger Subsystem

7.8 External Port

BDAS has 8 channels, 4 internal and 4 external. The external channels can be accessed through a 9 pin DB9 connector. The maximum input for the external channels is ±250mV. The following lists the pin definitions

Table 7.1 External Port

Pin Number	Descrition
1	Channel 5
2	Channel 6
3	GND
4	-5V DC
5	not connected
6	Channel 7
7	not connected
8	+5V DC
9	Channel 4

7.9 Shielding

When acquiring bio-signals noise is a major problem. Shielding can reduce the noise problem. A shield is a metallic partition placed between two regions of space. Electro-magnetic radiation can be kept out by using proper shielding.

With high speed digital signals the shield should be connected to a noise-free ground at both ends. Shields should be grounded only at one end, for low frequency signals such as those under 1 MHz. Connecting the shield to ground at more than one end will cause ground currents to flow.

7.10 Decoupling

When digital ICs switch, they draw a lot of current. The power supply path has inductance in it. When the IC switches this momentary draw of current will cause a voltage drop along the power supply path. This power supply transient can be reduced in one of two ways, either decrease the power supply inductance or decrease the current. By using a power plane or grid the inductance can be minimized. The transient current can be eliminated by supplying the current from another source close to the IC which does not have a long path.

The IC decoupling capacitors supply the intermittent current needs for the digital ICs. By using large bulk capacitors the decoupling capacitors can be recharged. As a rule of thumb, the bulk capacitors should be ten times the sum of all decoupling capacitors and located near the power supply. Since these capacitors are usually quite large, the choice is between the electrolytic types tantalum and aluminum. Since aluminum has large inductance, the tantalum type is more appropriate [Ott, 1987].

Decoupling capacitors must supply ICs switching at very high speeds (in BDAS up to 2 MHz) therefore they should be low-inductance high-frequency types. Ceramic types are most often used.

Larger than necessary capacitors should not be used because of the increased inductance which will cause a lower self-resonant frequency. Above self resonance, the circuit becomes inductive with its impedance increasing as the frequency increases, causing a voltage drop, therefore noise, at the V_{CC} input of the IC. With this in mind, capacitors greater than 0.01 uF should rarely be used except for devices with large switching current demands such as large RAMS.

The decoupling capacitors should be placed as close to the IC as possible. The longer the traces on the circuit board, the larger the trace inductance.

7.11 Signal Ground

Ideally ground is an equipotential reference point but since current flows through different ground paths each ground point could have a different potential. Signal ground can be defined as a low impedance path for current to return to the source. There are three categories of ground, namely: single-point, multi-point and hybrid.

Single point ground can have two configurations namely: parallel and serial. Parallel connections, meaning separate ground connections are available, has the advantage of no cross coupling currents. The disadvantage is the need for separate paths requiring more wiring. Series connections are the easiest from the wiring standpoint but have additive current crosstalk which can cause large ground potential differences.

Multipoint ground systems are composed of components connecting to a low impedance ground plane. The connections are kept as short as possible to minimize their impedances. At high frequencies, the current flows only on the surface due to skin effect. Therefore the thickness of the ground plane is not critical.

BDAS uses both parallel and series single point grounds. For noncritical ground circuits, such as most of the digital portion of the system, the multipoint ground connections were used. The parallel connections were used in the analog sections which can be very sensitive to grounding potential differences. The system uses two separate ground returns. There is one ground for the digital circuits, and one for the analog portion. They are connected at one point.

7.12 Components

Components come in many different varieties. The application determines the component type or family.

7.12.1 Capacitors

Capacitors are characterized by their dielectric material. Each type of material has different properties which should be taken into account when choosing a capacitor for a specific application. Frequency is the major criterion when choosing the capacitor type. The equivalent circuit for a non-ideal capacitor is an inductor in series with a resistor in series with a parallel combination of an ideal capacitor and another resistor. Due to the inductance the non-ideal capacitor becomes self-resonant. Above the self-resonant frequency, the capacitor has inductive reactance and an impedance increasing with frequency. The capacitor's inductance determines the operating range of the capacitor. In summary the lower the inductance of a capacitor, the higher the operating frequency.

Aluminum electrolytes are usually large and have large inductances. Therefore, they should only be used in the low frequency range. Mica and ceramic capacitors have low series resistance and low inductances. Therefore, they are good at high frequencies. Polystyrene capacitors have very low series resistances. Table 7.1 shows different capacitor frequency ranges [Ott, 1987].

Table 7.2 Capacitor Frequency Ranges

Type	Frequency Range
Al Electrolyte	1 Hz to 1 MHz
Tantalum Electrolyte	1 Hz to 50 kHz
Paper	100 kHz to 5 MHz
Mica Glass and ceramic	1 kHz to 10 GHz
Mylar	100 Hz to 10 MHz
Polystyrene	1 kHz to 10 GHz

In the BDAS system, the frequencies range is from 1 Hz to 10 kHz. For coupling, tantalum electrolytes will be used. Although tantalum capacitors cost more, they have low inductance and are used in the IA and, as recommended by the manufacturer, in the A/D power lines. For power supply bypassing, ceramic capacitors are used.

7.12.2 Digital ICs

There are many different IC families. Since remote BDAS is battery powered, power consumption is critical. This will be the criterion for choosing the appropriate IC family. Version 1 PC Card does not have the power concerns of the remote unit so that the LSTTL family is used. There are two basic gate groups MOS and BIPOLAR. For low power consumption the CMOS families are better suited. There are many different CMOS logic families [Hodges, 1988].

The MC68HC11 is a HCMOS CPU. Therefore, interfacing using CMOS ICs is straightforward. The LS TTL family cannot interface directly because the 6811 min

$V_{IH} = 0.7x V_{DD}$ (minimum input voltage considered a one) which, with a 5 volt supply is 3.5 volts and the LS TTL series V_{OH} (minimum output voltage for a 1) is 2.7 volts. Most of the digital ICs used in BDAS are CMOS devices, with no interfacing problems [Hodges, 1988].

Chapter 8 Remote Version Hardware

The remote version of BDAS contains a microcomputer that controls the acquisition of the bio-signals. The MC68HC11 microcontroller is the heart of the remote unit. It is a self-contained battery-powered unit that communicates to the PC through a serial port. The remote unit can function independently from the PC.

BDAS consists of three boards, the main board, multiplexer board and the front-end IA board. The front-end IA board and the multiplexer board were discussed in section 7. An overview of the main board hardware is shown in Fig 8.1. The individual building blocks are shown in more detail in Figures 8.2 - 8.5.

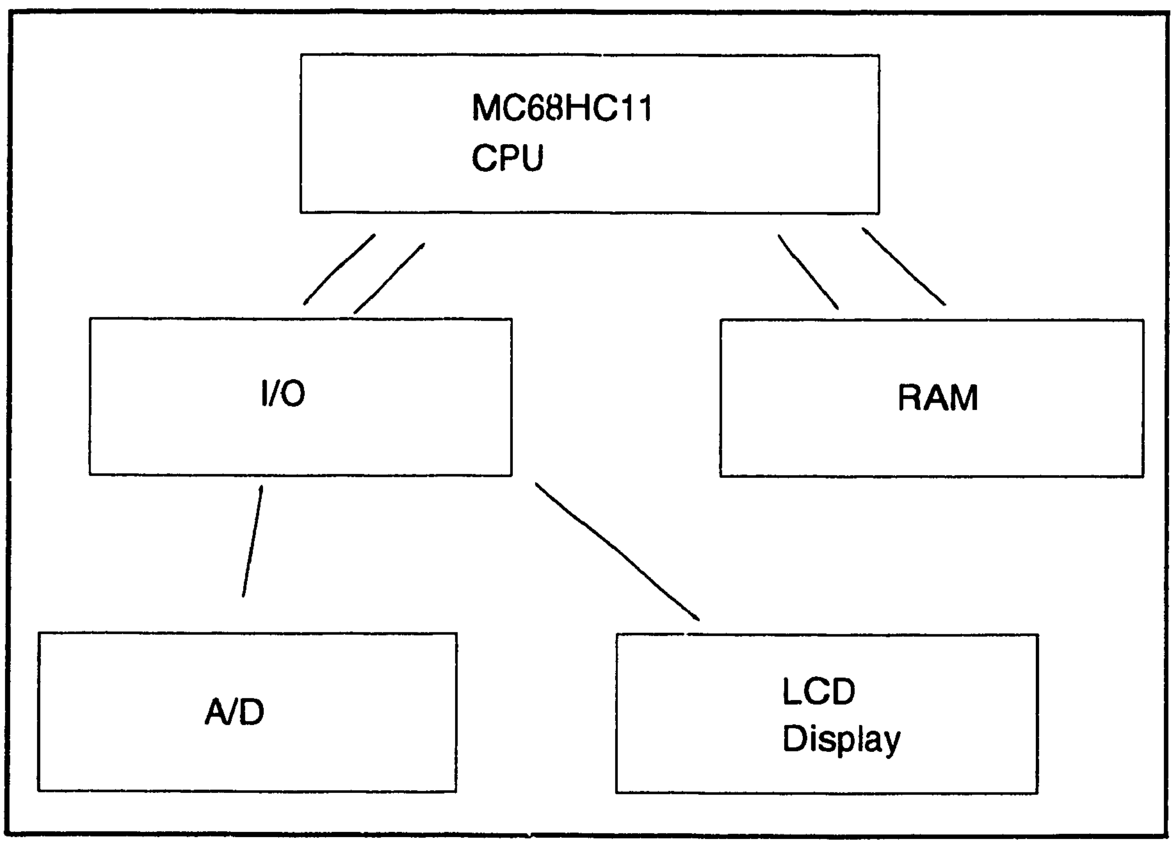

Fig 8.1 Remote Hardware Overview

8.1 MC68HC11 Microcontroller

The remote unit needs a microcontoller. Different microcontrollers were considered such as the Intel 8051, 8031, and 8096. The 68xx series from Motorola was also considered. The MC68HC11 microcontroller is an advanced 8-bit low cost microcontroller which had some unique characteristics which the others did not have. It has 40 I/O ports, 5 timers, serial port and on board A/D subsystem. The on board A/D is an 8-bit converter with 8 multiplexed channels. The MC68HC11 can be operated in two normal modes namely: single chip and expanded mode. In single chip mode, all the I/O ports are available and no external RAM can be accessed. In expanded mode, the MC68HC11 can access 64kb of RAM with some of the I/O ports converted to data and address buses. There are 22 I/O ports left in expanded mode. Since the remote unit can be used stand alone for acquisition it contains RAM to store the acquired values. The MC68HC11 has an internal 2kb EEPROM which contains the system program. The system runs at 8.0 MHz with a bus cycle time of 2 MHz.

The 68HC11 uses memory-mapped I/O which allows the use of the full instruction set when accessing I/O devices. BDAS I/O ports are input or output buffers that are decoded like memory. The decoding logic consists of a 74LS682 octal compare chip and a 74LS138 3 to 8 decoder. The input ports use the 74HC244 octal buffer. The output ports use the 74HC374 latch (Fig 8.5).

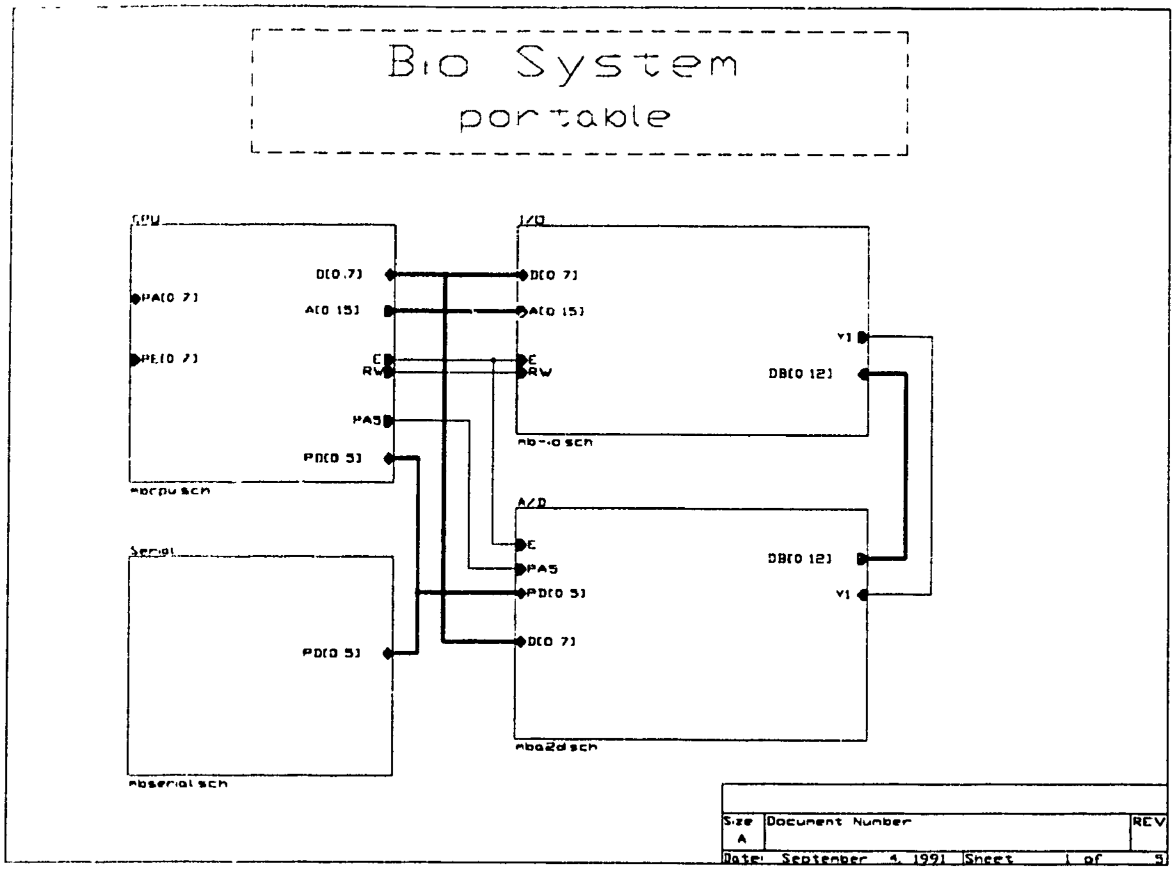

Fig 8.2 Main Board

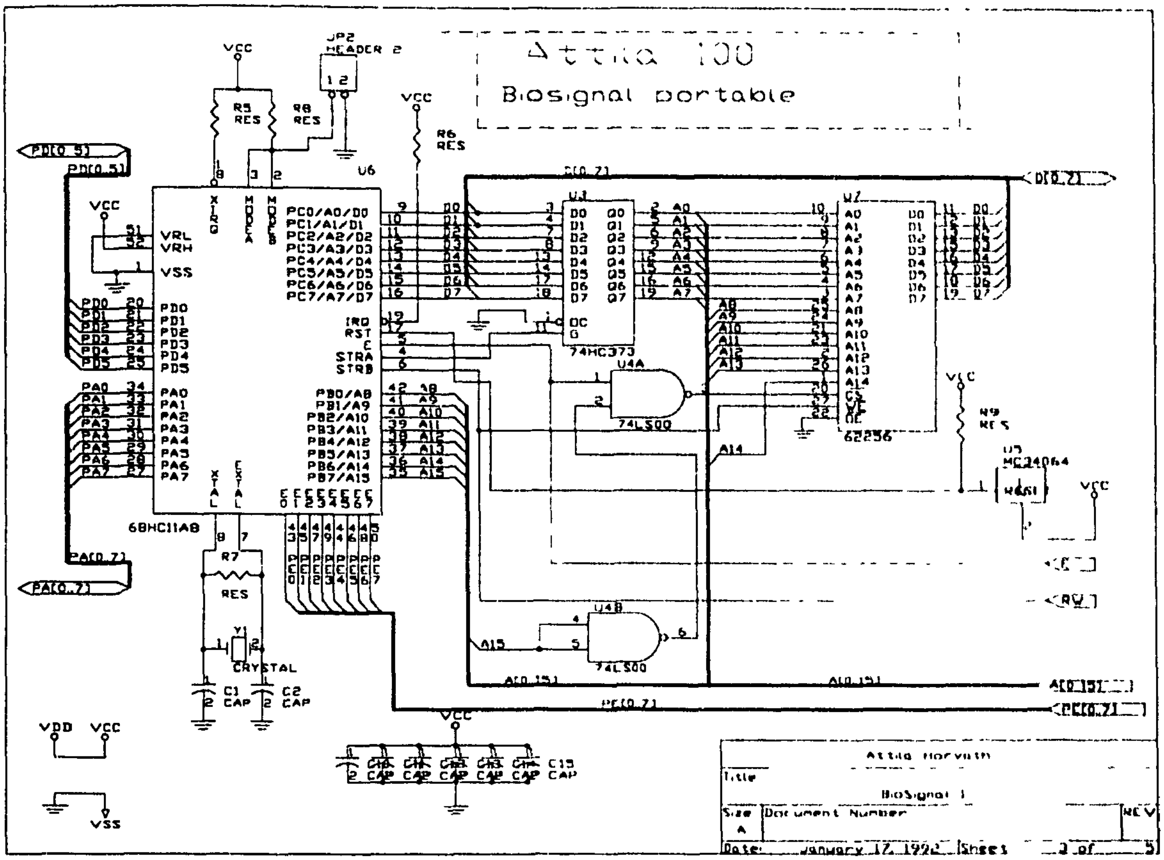

Fig 8.3 CPU Subsection

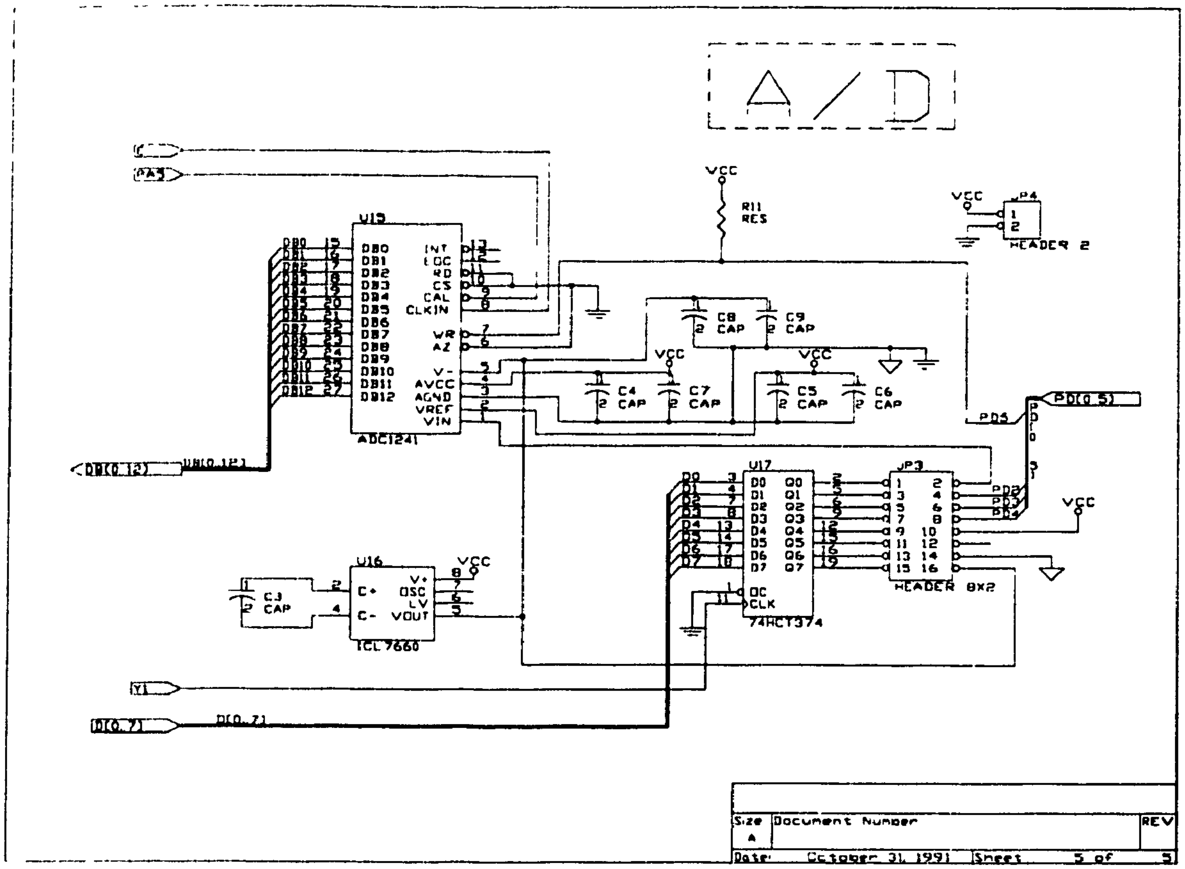

Fig 8.4 A/D Subsection

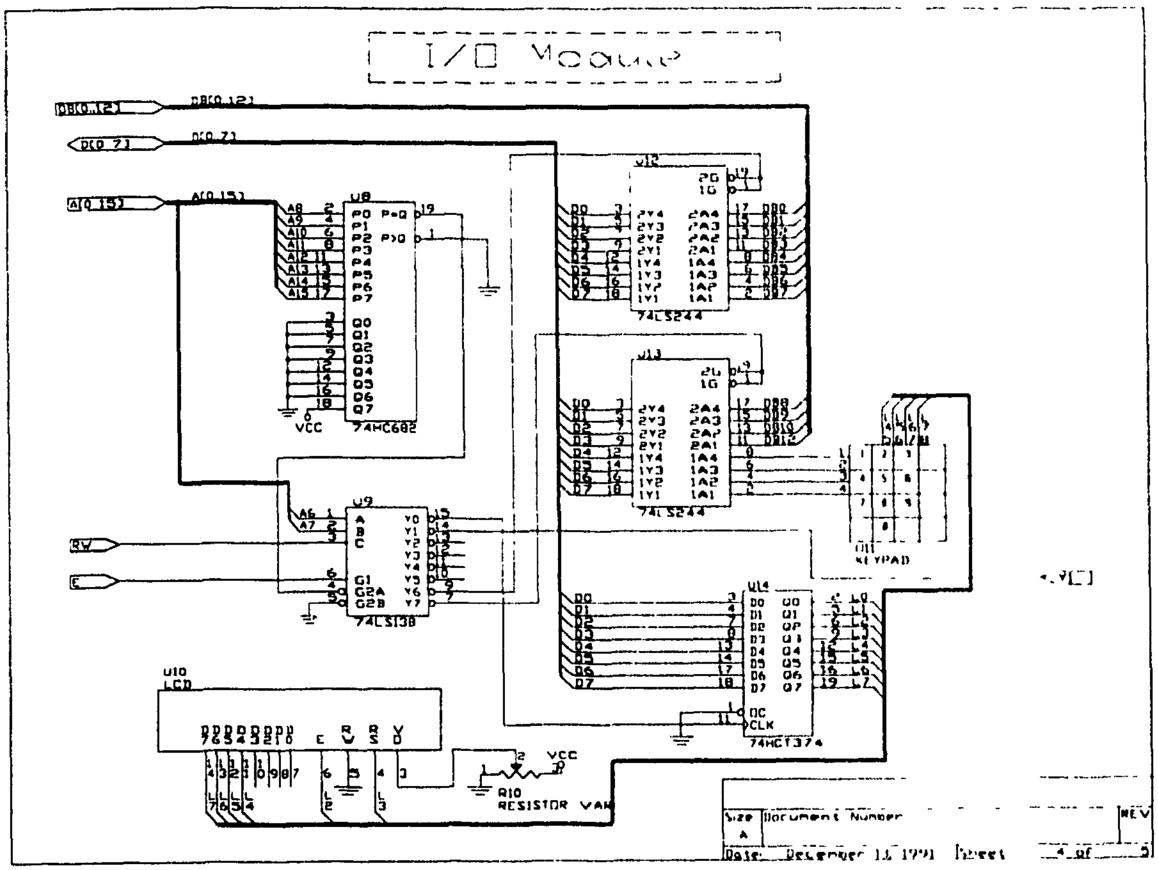

Fig 8.5 I/O Subsection

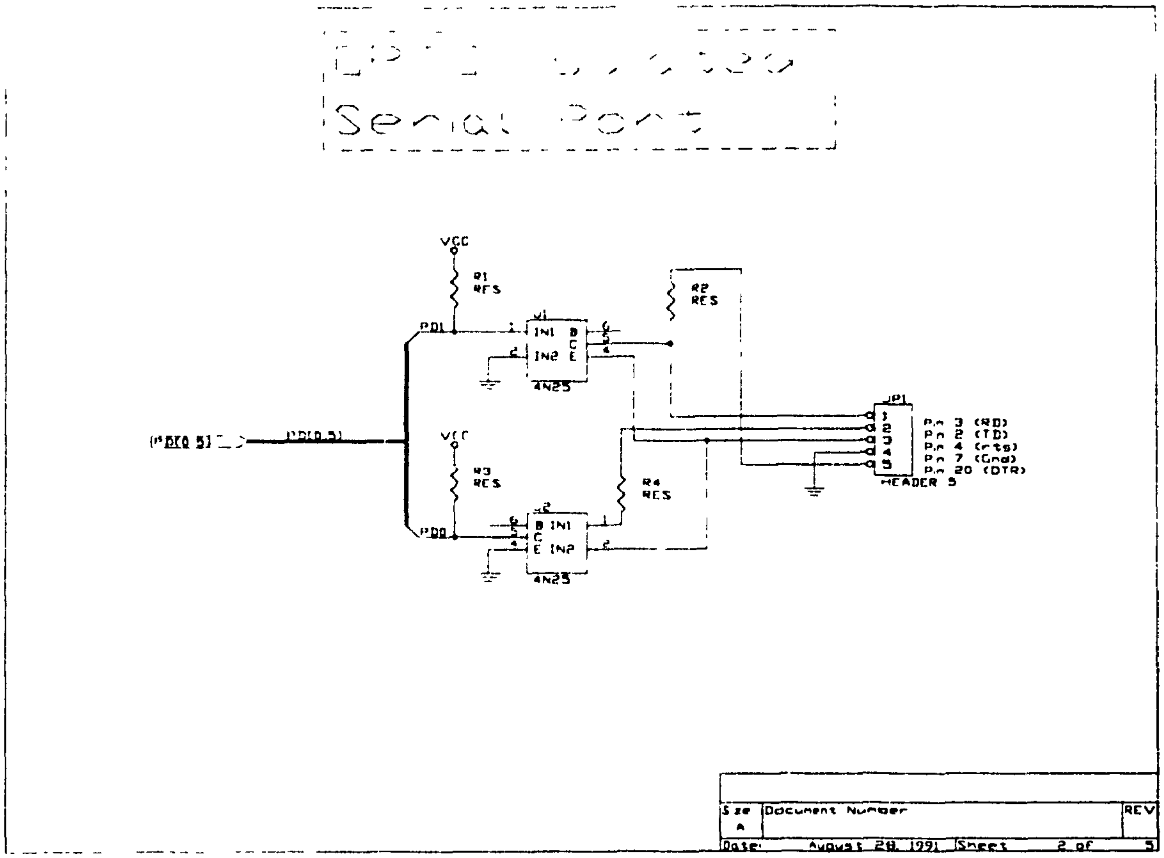

Fig 8.6 Serial Port Subsection

8.2 RAM interfacing

The 6811 has a bus cycle (E clock) of 2 MHz or period of 500 ns which is 25% the speed of the system clock. Table 8.1 shows the timing characteristics of the MC68HC11. The bus is multiplexed with the data and lower 8 address lines sharing the same pins. On the first half of the E clock cycle the address is put on the multiplexed pins. A 74HC373 8-bit transparent data latch is used to latch the address (Fig 8.3). On the second half of the E clock cycle the data is read or written. The minimum memory cycle time is 250 ns. The non-multiplexed address lines (upper 8 lines A8 - A15) and the R/W lines are valid 94 ns (tAV) before the second half of the bus cycle (rise of E clock). The multiplexed address valid time to E clock rise is 84 ns. The slowest memory that can be interfaced to the 6811 running at 8.0 MHz (2.0MHz bus) can be calculated as follows. Since the bus is multiplexed, the complete address is valid 84 ns before the second half of the E clock cycle. This extra time before the E clock rise can be used for read only devices like EPROMs. With an EPROM, this extra 84 ns can be used for chip select. Output enable should be synchronized with the rising edge of the E clock. With RAM chips the extra 84 ns cannot be used on write. The rising edge of the E clock cycle is used for synchronization, without which the RAM chip could be selected inadvertently by an invalid address, as the address lines are settling during the first half of the E clock cycle. This could be disastrous if a write was done with an invalid address and/or data. The RAM must have a minimum cycle time of 500 ns (when E clock 2 MHz). The access time (Table 8.1) must be shorter than 192 ns, therefore the RAM chip select plus address decoding must be less than 192 ns. CMOS 74HCxx gates have a typical propagation of 10 ns. In this case only one level of NAND gates is used therefore the max chip select time for RAM reduces to 182 ns.

Table 8.1 MC68HC11 Timing Characteristics

Address Hold time	tAH	33 ns MIN
Read Data Setup Time	tDSR	30 ns MIN
Read Data Hold Time	tDHR	10 ns MIN 83 ns MAX
Data Delay Write Time	tDDW	128 ns MAX
Data Hold Write Time	tDHW	33 ns
Pulse Width E high	PWEH	222 ns
MPU Access Time	tACCE	192 ns MAX
NON-Mux addr, R/W valid before E rise	tAV	94 ns MIN

In the write case, the maximum write setup time (tDW)(hold time on write) the RAM device must have is PWEH - tDDW + tDHW which in this case is 222ns - 128ns + 33ns= 127ns. Also from the E positive edge, the decoding logic delay time and RAM chip select time to end-of-data write must be less than PWEH - RAM Hold time (222ns - hold). The tDH (Data Hold RAM) must be less than tDHW (CPU). tDHW is the time after the E clock goes low (in this case that means the RAM's chip select is off) that the data is still valid.

The 62256L-100 CMOS static RAM is the system memory. Static RAM was chosen to eliminate the need for refresh circuits. The CMOS version was chosen because the system runs on batteries and low power is an important feature. The 62256L-100 meets all the timing requirements to interface to the MC68HC11. The cycle time for the 62256L-100 is 100 ns which is well below the calculated 6811 maximum .

For reads the maximum chip select time before data is valid (tDDR) is 100ns, again well below the 182 ns maximum. For writes, the write pulse width must be at least 60 ns (RAM) which, in this case, is governed by E clock pulse width high (PWEH = 222 ns). The RAM requires a minimum of 35 ns of Data Valid to End-of-Write (tDVWH). The 6811 provides 127 ns [Motorola, 1989c].

8.3 Data Acquisition

Data can be captured in three different ways. In the first way the remote system continuously captures the data in on-board 32k of static RAM and at the same time transmits the data to the PC. The limitation of this continuous capture is the speed of the serial data link (RS232). The second way is to capture intermittently with continuous transmission to the PC. The advantage of this method is that higher data acquisition rates can be used but only for short periods of time. The third way is to use the remote device in stand-alone mode, where the device continuously captures the data to the on-board RAM.

8.3.1 Continuous Capture

The limiting factor for the maximum sampling rate in this case is the serial port. The serial transmissions are presently limited to 9600 baud which is roughly equivalent to 800 12 bit samples per second. This means the 8 channel bandwidth presently is limited to 400 Hz or 50 Hz per channel.

8.3.2 Intermittent Capture

Transmission medium limits throughput but capture rates do not have to correspond to transmission rates. Most types of biomedical applications that require high-speed capture rates need them only for a short period of time. By having on-board RAM, the system can capture at high speeds to the RAM and then continuously send out data at the slower transmit speeds. One tradeoff is that capture rates depend on the amount of on-board RAM and length of capture. Another tradeoff is that data capture is not continuous; after the acquisition of one buffer, the system must wait for the buffer to be transmitted before it can acquire more data.

For example, if the system is used to capture at a high rate of 8kHz, then the buffer is full after approximately 2.6 seconds. At a transmission rate of 9600 Baud or 960 Bytes/sec the system takes approximately 40 seconds to send the buffer. The remote system can also start a capture while it is transmitting, thus decreasing the time between acquisitions.

8.3.3 Remote Capture

Remote capture can capture continuously at any rate up to the maximum sample rate of 8kHz. The limiting factor is the buffer size. This mode of capture will usually be used at slow sample rates.

8.3.4 RS232 port

The remote unit uses a simple 5-wire interface for communication. These are the DTR, RTS, Rx, Tx and ground (pins 2,3,4,7 and 20). A straight through cable is all that is needed. A NULL modem cable is NOT necessary. Pin 20 is set to +10V and pin 4 is set to -10V. Pin 7 is signal ground. Pin 2 is transmit and pin 3 is receive. The opto-isolators are necessary because short circuits at the PC end could couple dangerous voltage levels to the serial RS232 port. The opto-isolators used are 4N27s. They have an isolation surge voltage of 7500 Vac.

8.4 Power Consumption

Power consumption of the remote unit was calculated using an ammeter at the battery terminal. Power consumption was measured to be 30 mA. Four AA NiCad batteries are needed to operate the system. These batteries have a 450 mA-hr capability. The system can, therefore, run about 15 hours on one charge.

Chapter 9 PC Card Version Hardware

The PC Card version (Fig 9.1) of BDAS has many similarities with the remote unit. The front end system of both versions are the same. The A/D subsystem is almost identical to the remote unit. The difference being isolation circuits connecting the front end to the A/D subsystem. The PC card version consists of I/O ports that are connected to the A/D subsystem. Instead of the microcontroller reading the data from I/O ports and then transmitting the information serially to the PC, the PC system reads the I/O ports on the PC Card directly. The main difference between the PC Card Version and the remote unit consists of the PC interface.

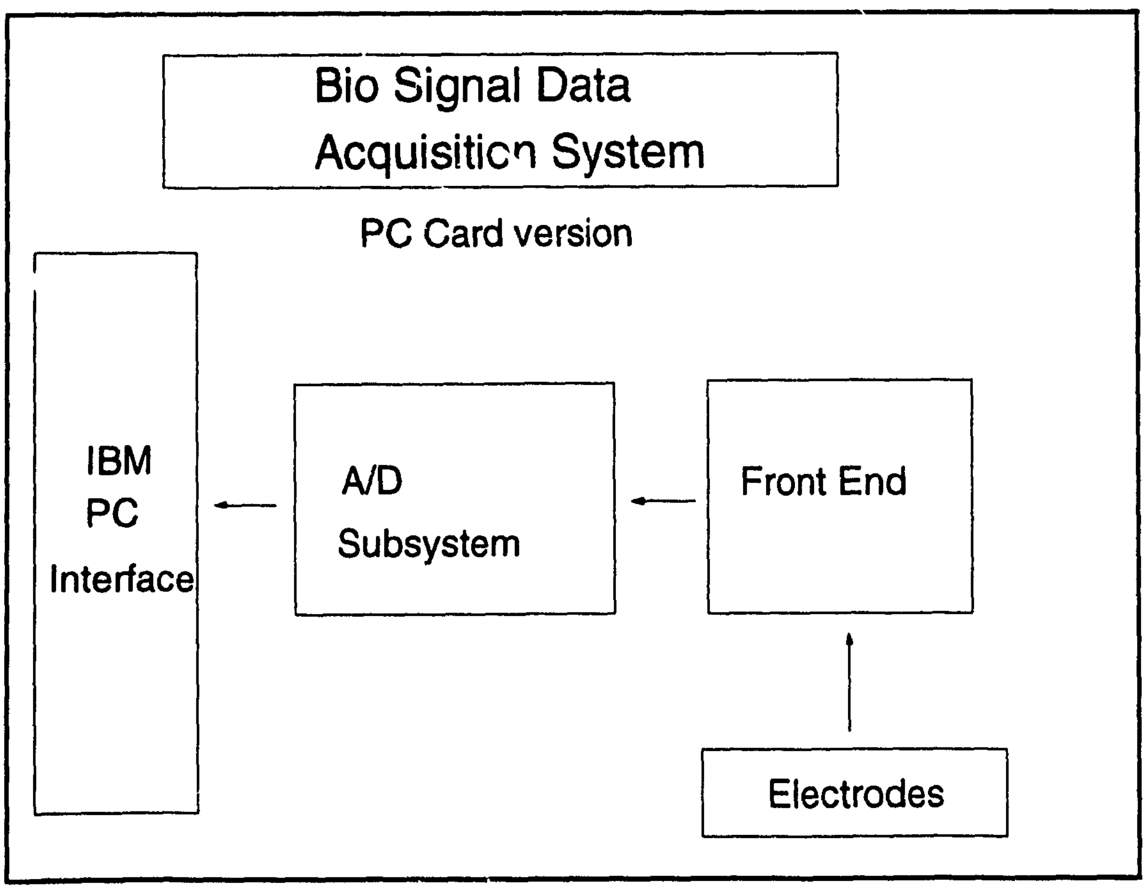

Fig 9.1 PC Card Overview

The PC card version consists of an 8-bit 1/2 card that plugs into the standard IBM PC bus. The IBM PC interface consists of a card that plugs into the standard IBM PC bus. The card does the necessary address decoding to interface the A/D circuits to the PC. The IBM PC has a separate I/O address space consisting of 1k addresses. Each interface that uses the I/O address space must decode at least the first 10 address lines. The data bus from each connector can be loaded by a maximum of two TTL devices. If more devices are needed to interface with the data bus, then the bus has to be repowered with a buffer.

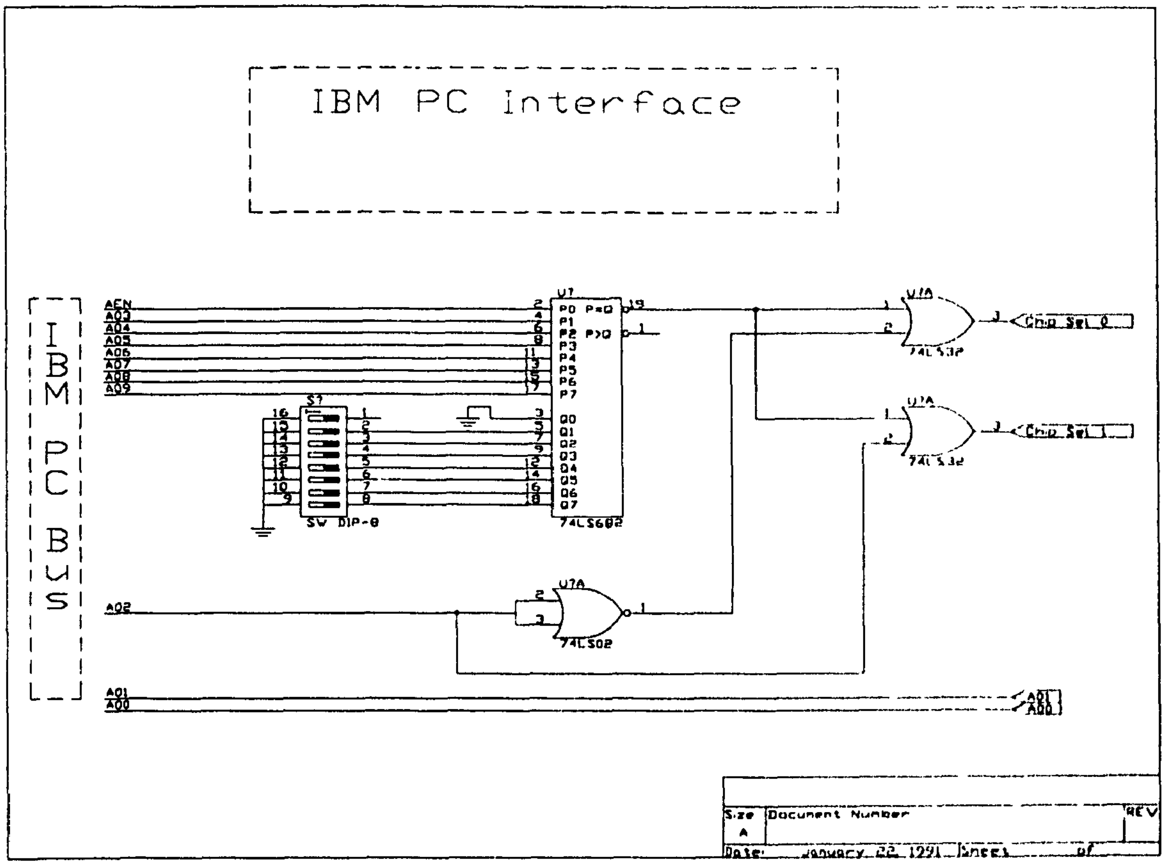

Fig 9.2 PC Interface

In summary, both remote and PC Card versions use the same front-end hardware and A/D. The addressing of the I/O ports (Fig 9.2) and opto-isolation of the A/D input are the major hardware differences.

Chapter 10 Case Study ElectroCardioGraphy (ECG)

A simple test was constructed to capture ElectroCardioGrams (ECG) using the remote unit. The same setup is used for EMG and ECG with differences being the electrode placement, amplifier gain and filtering.

10.1 Startup

Connect the remote unit to the PC using a RS 232 cable.

Power up the PC and the remote unit.

Start the BDAS system on the PC by typing >BIO_GM

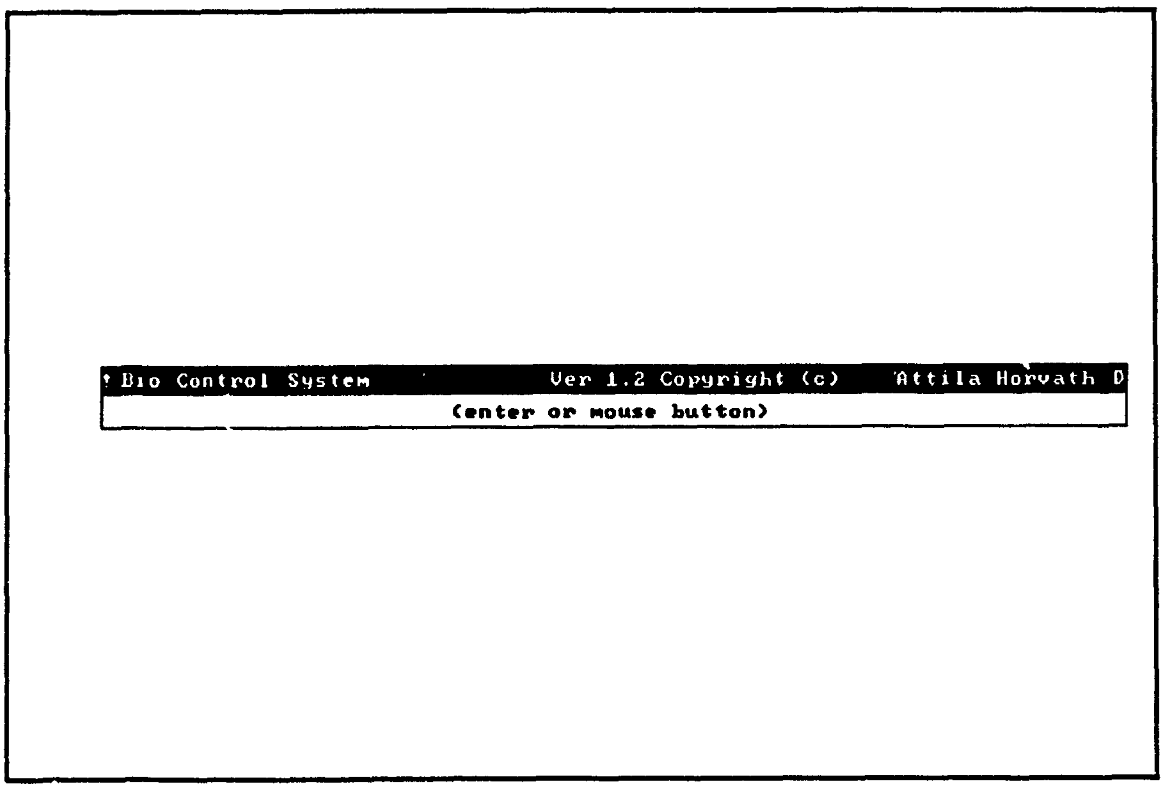

Fig 10.1 Copyright

Press enter after the copyright notice and the Main Menu appears.

10.2 Acquisition

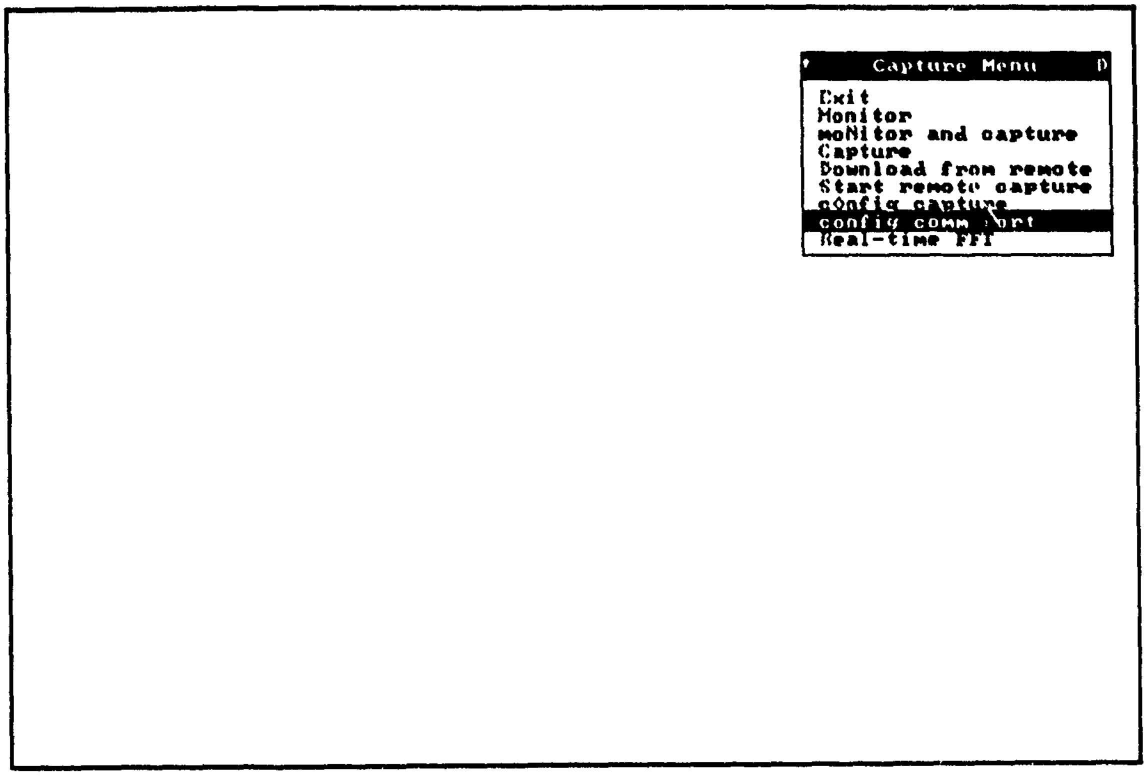

Fig 10.2 Capture Menu

From the Main Menu choose the Capture Menu. Next Choose : Config Comm Port and proceed to configure the communication channel (Fig 10.2) by specifying com1 or com2 and the baud rate. The system then checks the communication between remote unit and PC.

If the message "Remote Unit Not Responding" appears, check to see if the remote unit is powered on and the LCD has the message "Bio Control Sys".

Next choose **Config capture** in the **Capture Menu.**

The system gives a choice for the **Cutoff Frequency.** There are 8 possible cutoff frequencies for the front end filter located before the A/D. Choose the filter with the *lowest cutoff* frequency that will pass the highest input frequency of interest. In this case the filter chosen was the one with a cutoff of 100 Hz.

The **input scale** is the maximum input signal, in mV, to the A/D . The default 2500 is chosen.

The **number of bits** is the A/D output. The system assumes the output of the A/D is bipolar and the negative numbers are stored in 2's complement. The default 12 is chosen.

The **sampling frequency** should be at least twice the cutoff frequency of the front end filter. In order to compensate for nonideal front end filter characteristics the minimum sampling value should be at least 2.5 times the cutoff frequency. In this case, the sampling frequency was chosen to be 300 Hz.

The **Starting channel** indicates which port to start sampling from. In this case only one channel was used. The input was connected to port 0.

The **number of channels** indicates which channels to sample from. The number of channels specifies the number of channels to sample from, starting from the starting channel. For this example, only one channel was used.

The **Tone** turns on the sound which varies in frequency with input amplitude.

The **Real-time Filter** was selected to apply a notch filter. The option to display both the filtered and raw data is selected.

Gain controls the Programmable Gain Amplifier (PGA) at the output of the 4051 channel multiplexer. Gain from the preamp is incorporated into the gain figure. Since the preamps are initially set for a gain of 1000, the minimum gain is 1000. The PGA has a gain from 1 to 16. Each channel can have a different gain. The gain and channel description is prompted for each channel. In this case, the gain for channel 0 was 4.

Channel description gives a label to the signal in the capture file. Each channel will be prompted for a description. The name given to the signal was ECG.

Choose : *Monitor and Capture* in the capture menu. This option will allow the user to view the captured signal while it is being stored in a file. The system prompts for a **file name**. This is where the data will be saved. The last prompt "Ready to Capture" indicates the PC is ready to capture. Channel 0 is the raw signal and channel 1 is the notch filtered signal.

After the capture has been configured the system is ready to start capturing. Choose **Monitor** in the **Capture Menu.** Press any key to start the capture.

To stop the capture, press any key. The capture will also stop if the remote unit is turned off. A prompt appears which allows the capture to continue or return to the capture menu (Fig 10.4). When a key is pressed, the capture can be restarted or terminated. This start/restart feature allows different non successive captures to be stored in the same capture file.

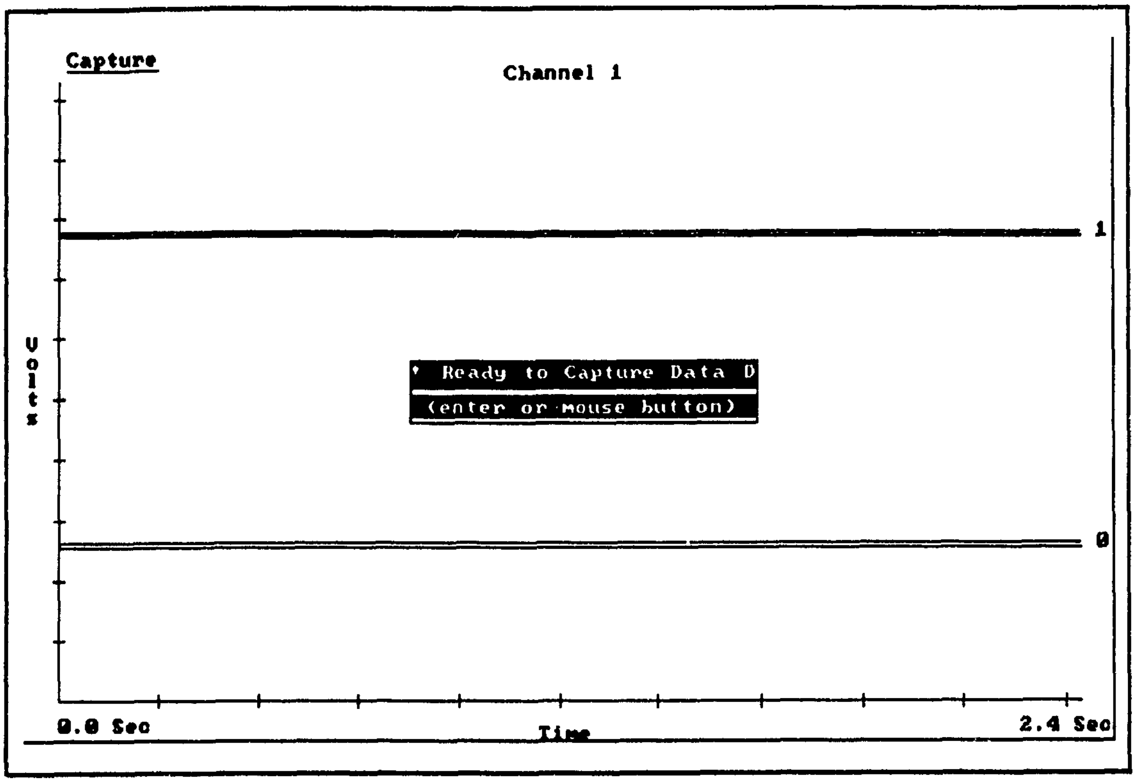

Fig 10.3 Capture start

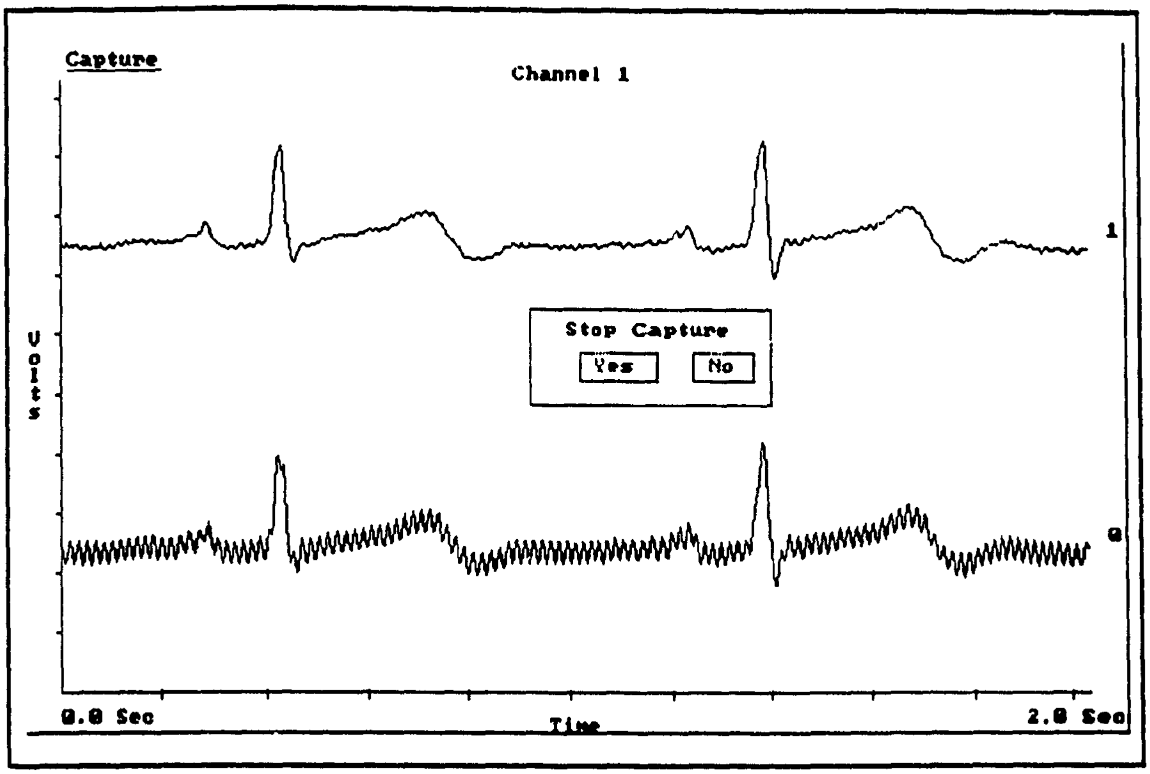

Fig 10.4 Capture end

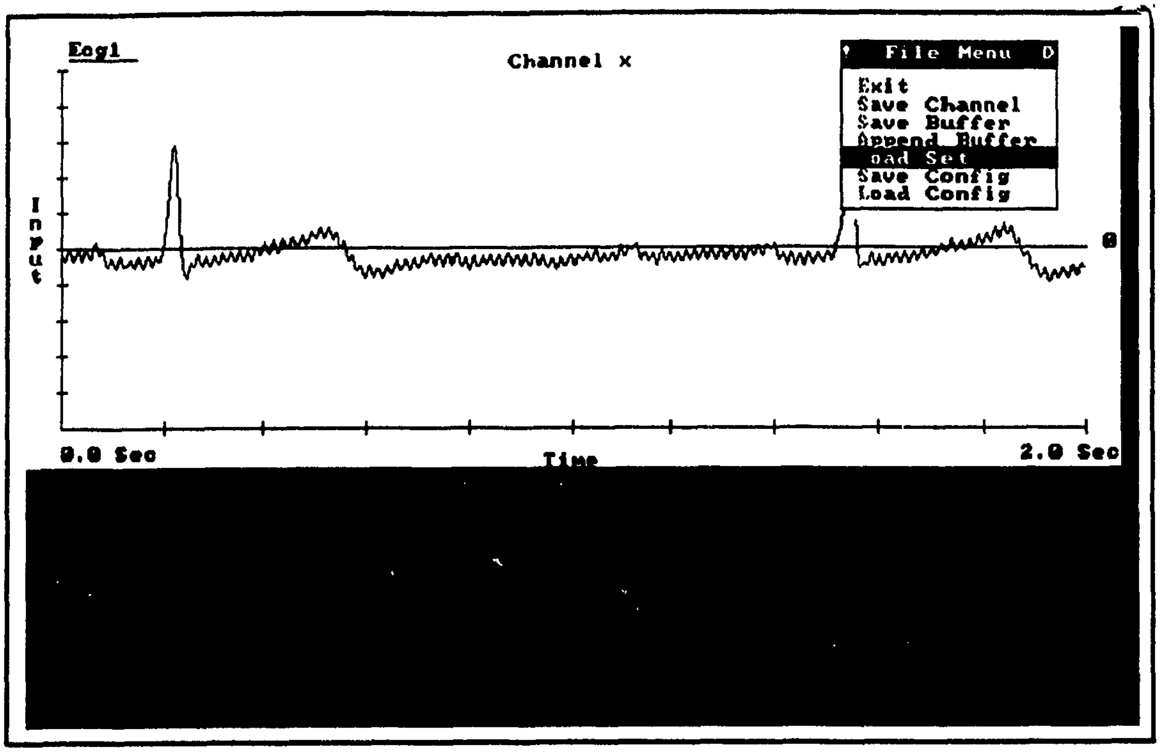

Fig 10.5 File Menu

Upon entering yes to stop capture, the data file (.set) on disk is closed. Once the capture is over, the .set file is closed.

10.3 Analysis

To analyze the data it needs to be loaded into a window. In the ***File Menu*** the ***load set*** option is used to load the previously captured file into a window. The file name is entered and the window is chosen. The window can be zoomed to full screen size from the ***Zoom Window*** in the ***Window Menu*** .

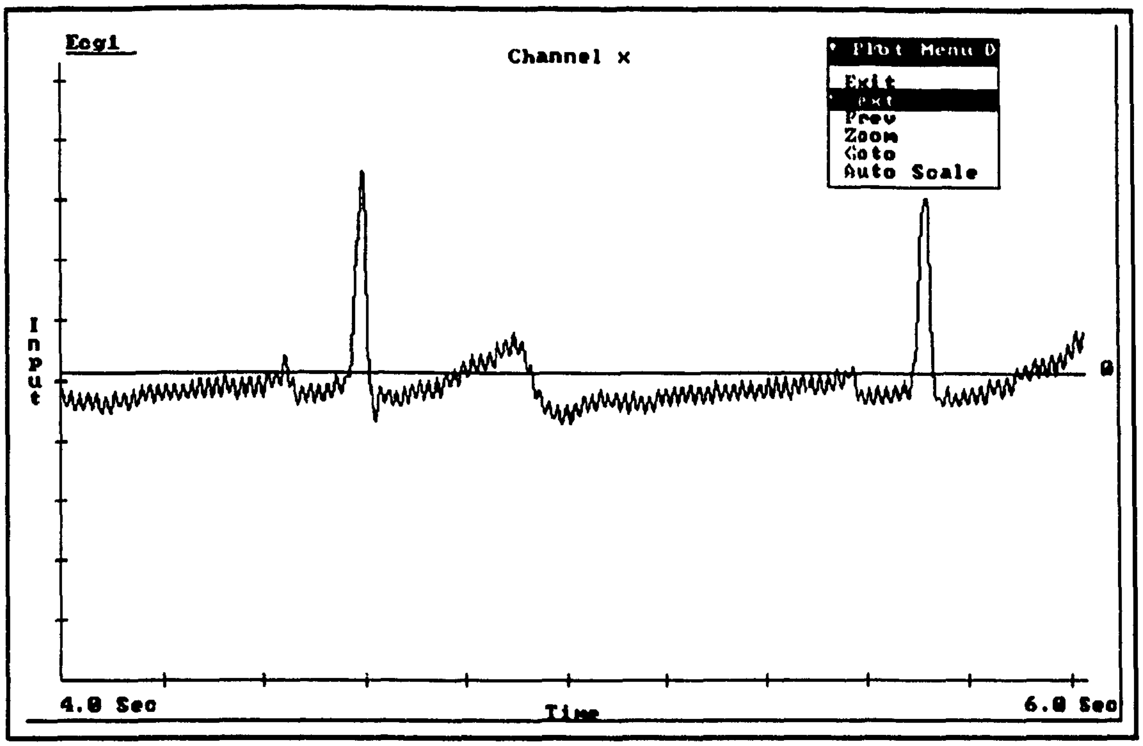

Fig 10.6 Plot Menu

The signal in a window can be filtered. The filter menu is accessed from the main menu.

In this example, low pass filtering is done on the ECG signal. The sampling frequency is displayed. Usually this parameter should be left alone as it corresponds to the sampling frequency at capture time. For a low-pass filter, the upper frequency needs to be specified. It should always be less than half the sampling frequency. The **order** is the order of the digital FIR filter. The higher the order, the better the filter (sharper cutoff). The tradeoff is that the higher the order the longer it takes to filter.

There is a notch filter that can be enabled to filter the data before the other filtering takes place. This is useful when the 60 Hz powerline interference is a problem.

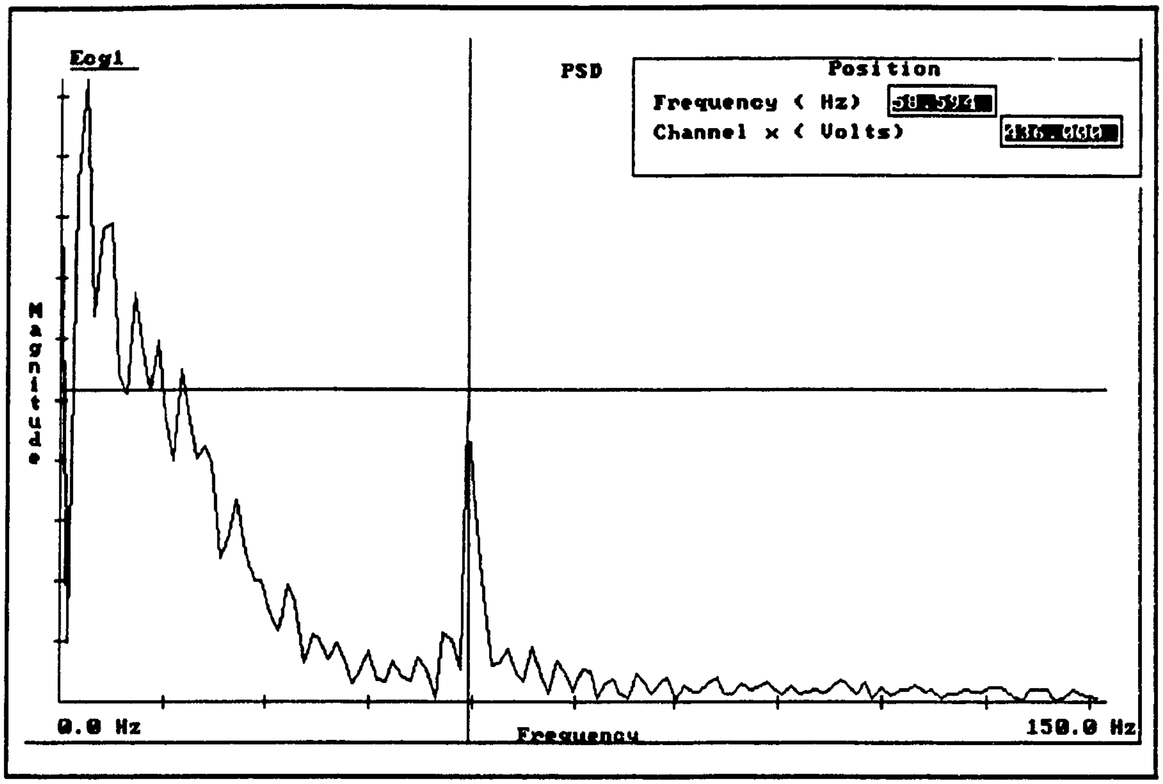

Fig 10.7 FFT of Input

The plot menu has options that will view different portions of a signal in a window. In Fig 10.6, the 4 to 6 sec portion of the raw (unfiltered) ECG is shown. Fig 10.7 shows the FFT of the input signal. When an FFT is selected there are different sizes to choose from. The size is the number of samples used in computing the FFT. In this case a 256-point FFT was chosen. The larger the size, the more detailed the FFT but the longer it takes to perform the FFT. Note the large 60 Hz component in the spectrum. In Fig 10.8 shows a relatively smooth signal after the high-frequency noise components have been filtered out. There is a noise component at the beginning of the filtered output due to the startup loading of the coefficients in the digital filter.

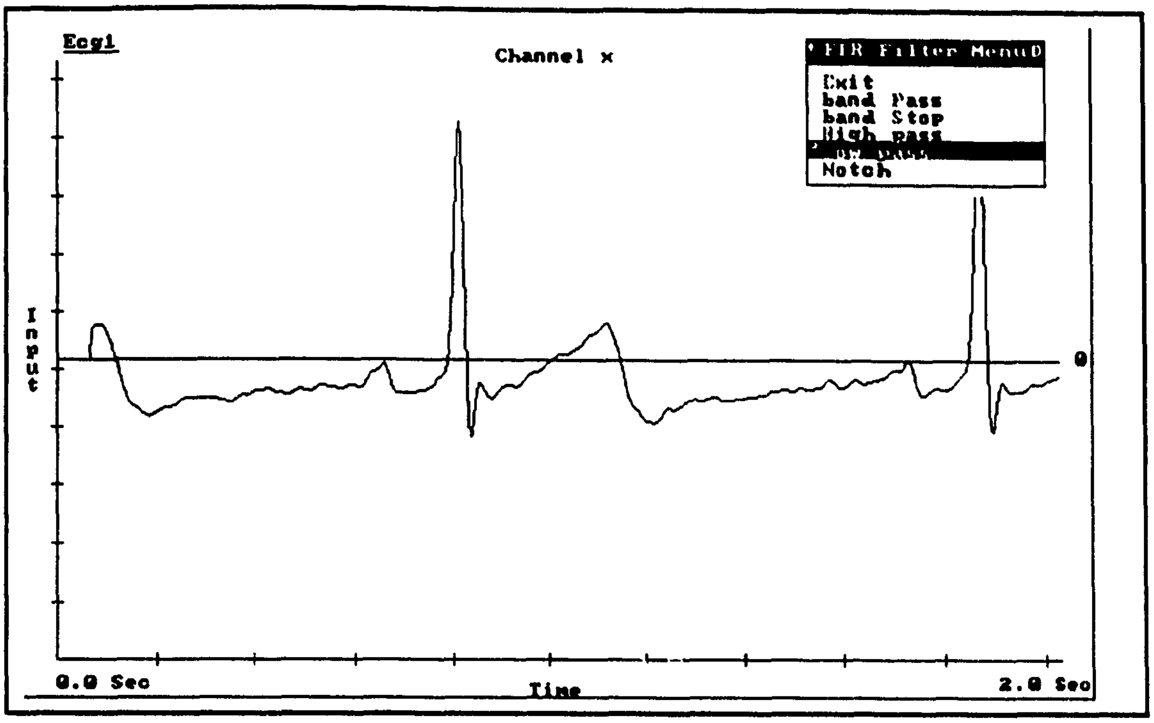

Fig 10.8 Filtered Output

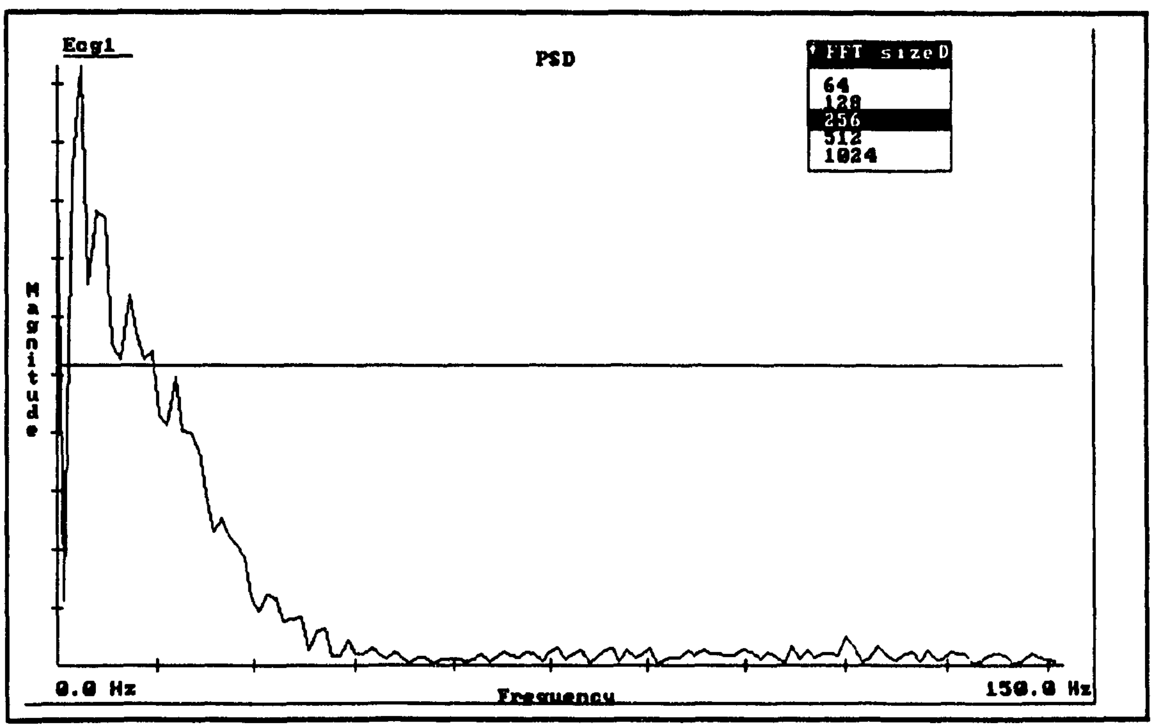

Fig 10.9 FFT of output

The FFT's of the signal input and filtered output is shown in Fig 10.9.

The *Integrate* option in the window menu integrates a portion of the displayed signal. It also can be used to find the time between two events. In the ECG signal, the time between two consecutive major peaks (QRS peaks) can be used to calculate the heart rate. First, the left border needs to be chosen (see Fig 10.11). Position the border using the arrow, home and end keys. The shift key in conjunction with the arrow keys allows for quicker travel.

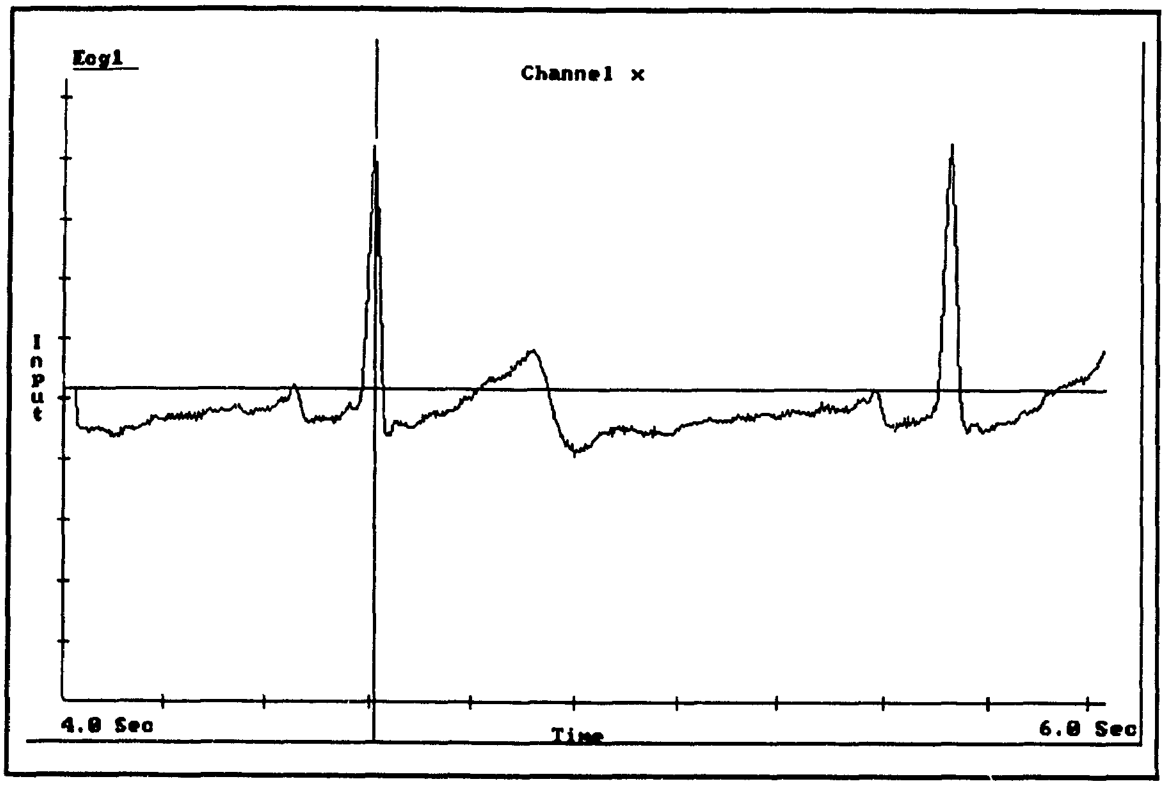

Fig 10.10 Integrate Left Marker

Position the right border using the arrow keys. The time difference is updated as the position changes (see Fig 10.12). In this case the time difference is 1.1 sec which corresponds to a heart rate of

$$\frac{60}{1.1} \approx 55\frac{beats}{\min}$$

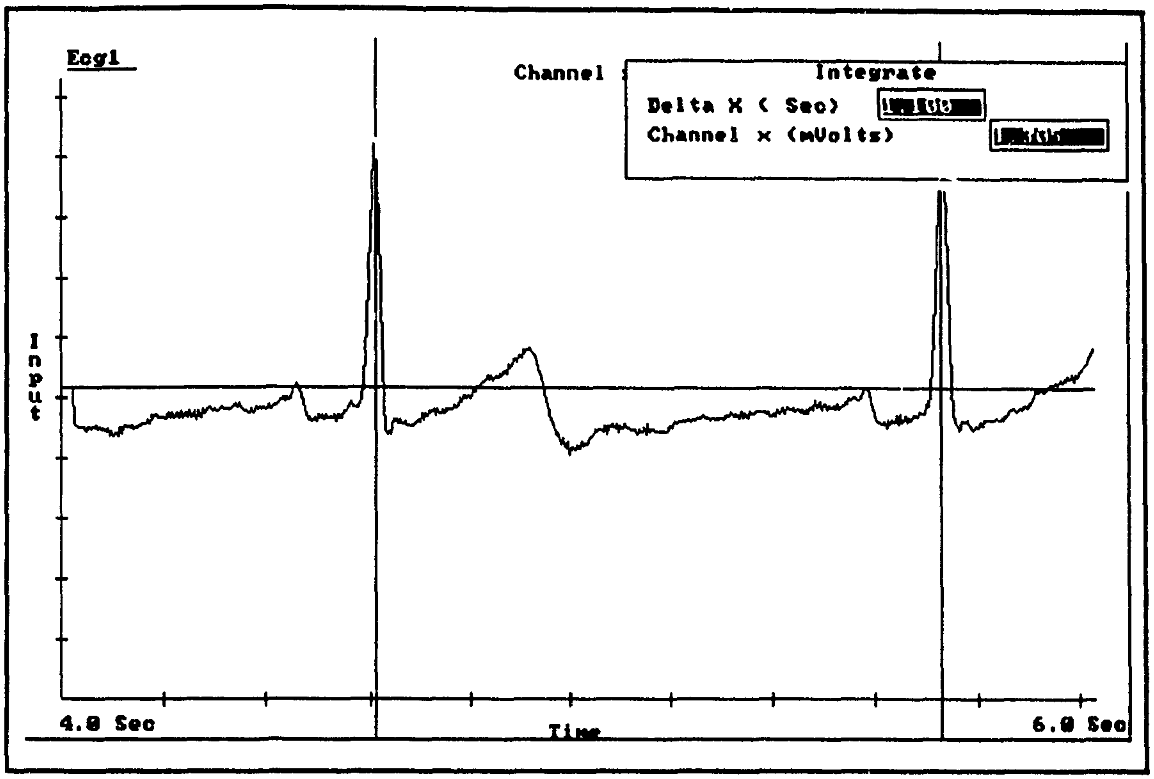

Fig 10.11 Integrate Right Marker

The window menu allows signals to be viewed individually, one above the other, or overlayed. The input, output or FFT of the input or output can be viewed. The value of the signal at a certain time can viewed using the ***position*** option. A portion of the signal can be integrated. ***View window*** is used to specify what portion of the signal from the file is viewed.

To exit the system, choose exit from the main menu (see Fig 10.12).

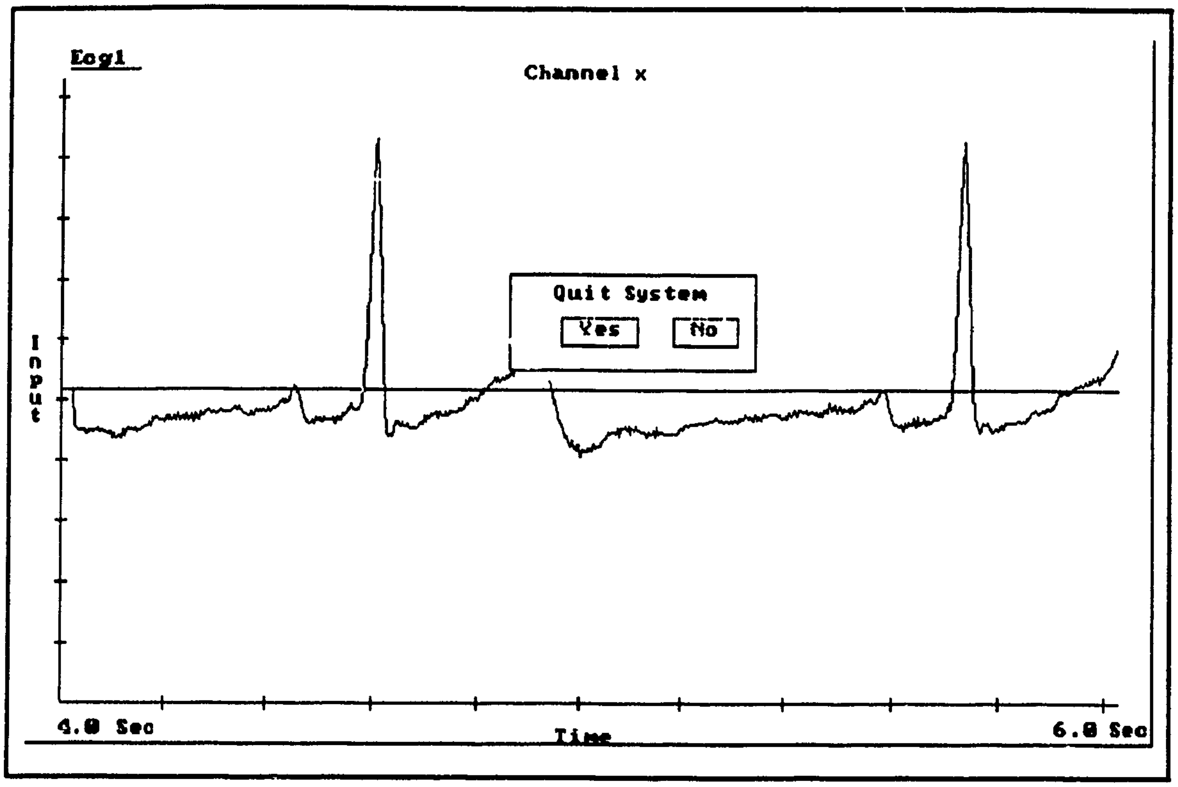

Fig 10.12 Exit System

Chapter 11 Case Study EMG

The EMG study starts with the same setup as the ECG study. The only difference is changing the sampling frequency to 300.

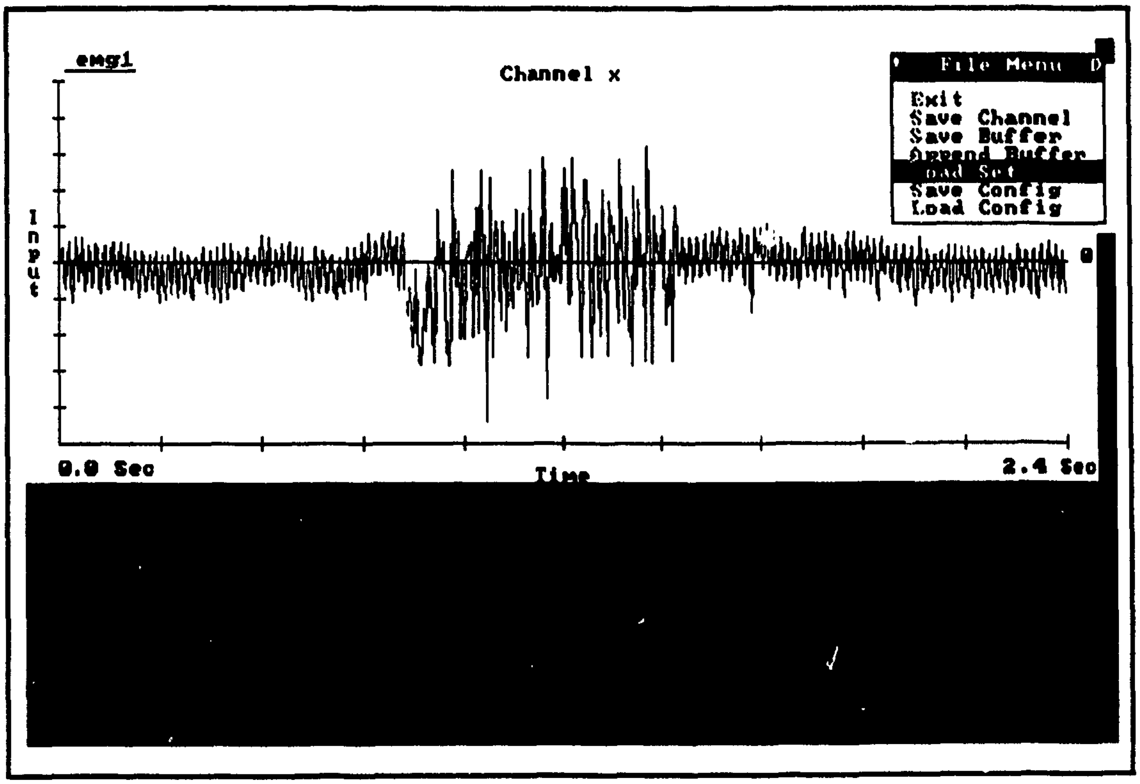

Fig 11.1 Emg Signal

From the file menu, load the captured signal four times into the same window.

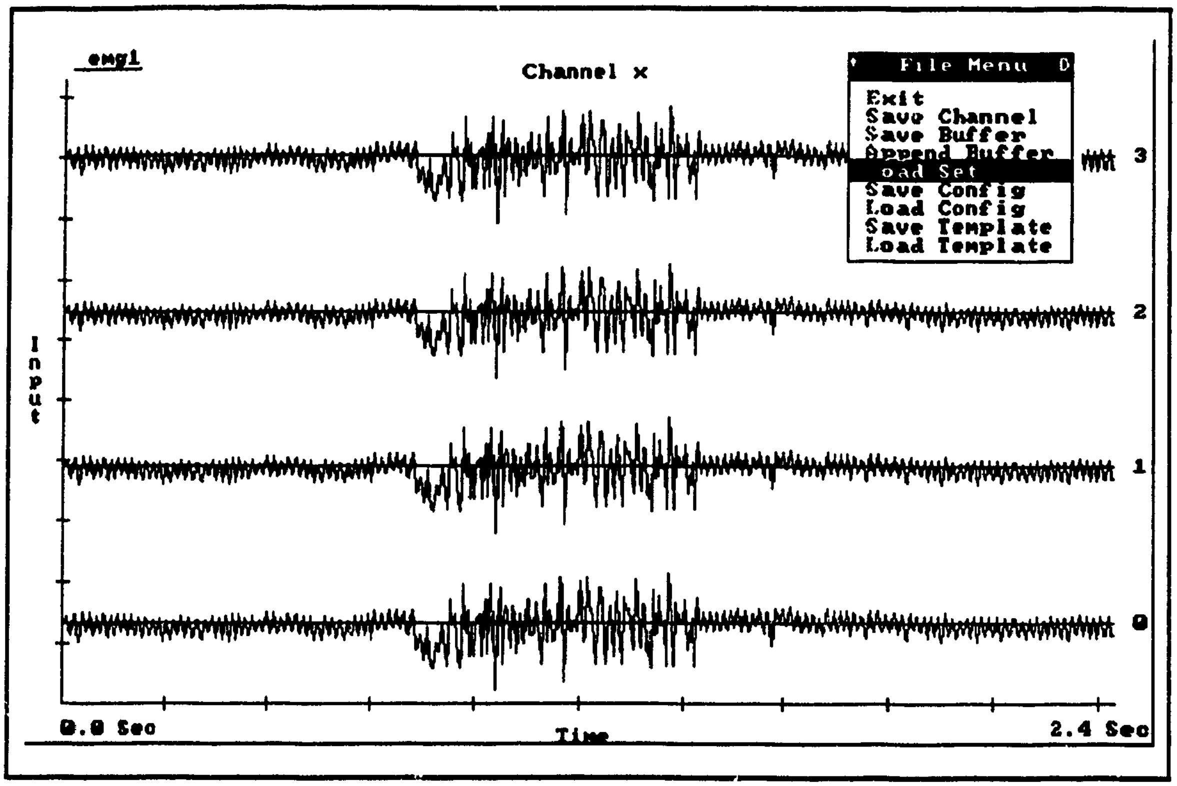

Fig 11.2 Emg Signal 4 copies

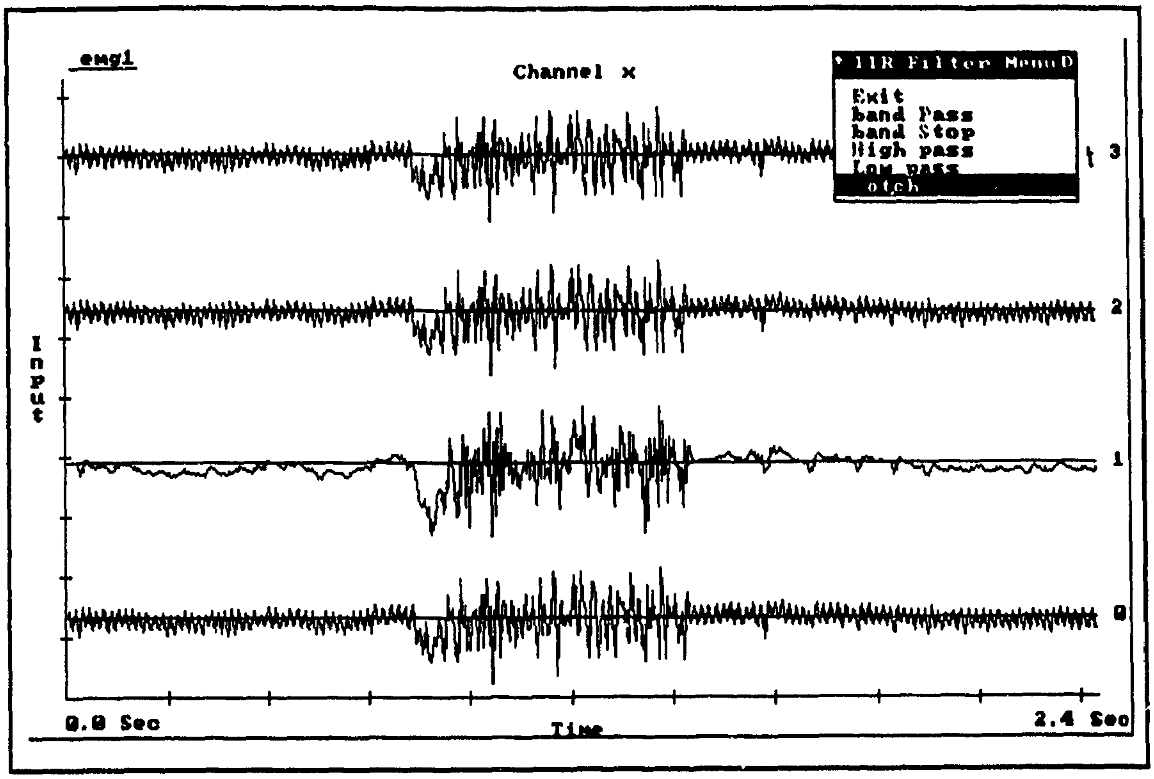

Fig 11.3 Emg Signal notch filtered

Channel 1 is notch filtered. The notch filter removes the 60 Hz noise.

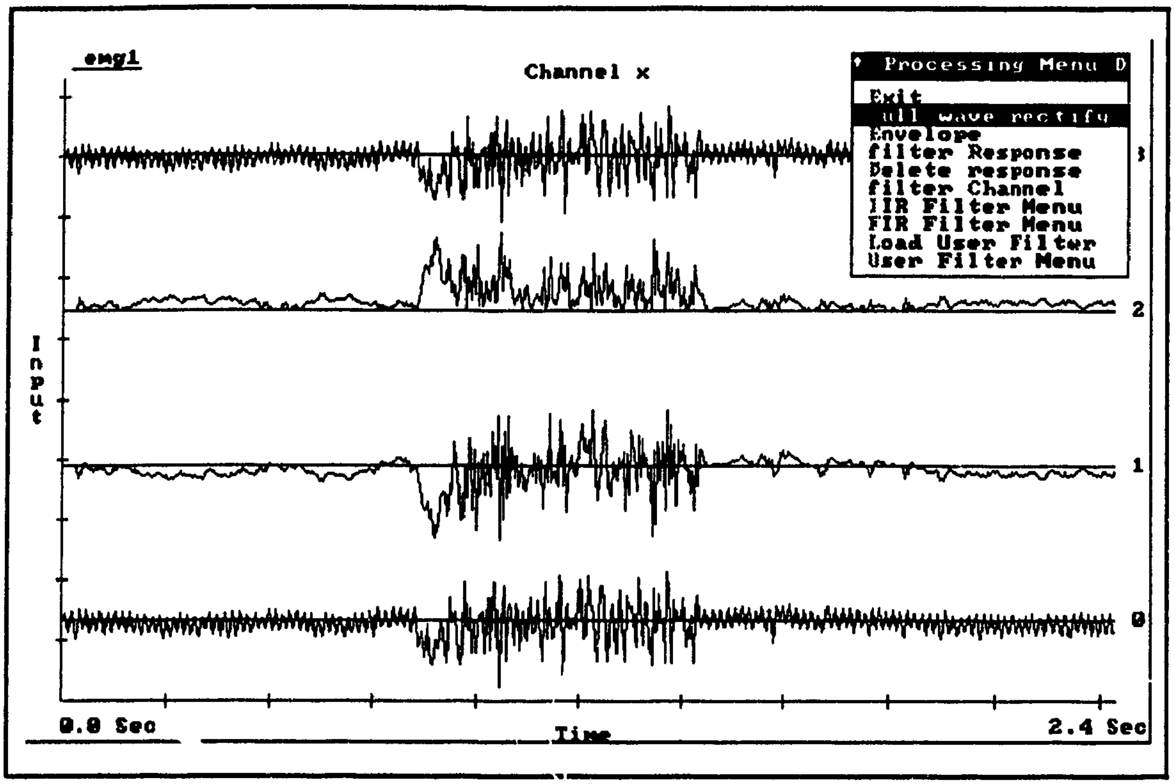

Fig 11.4 Emg Signal rectified

The notch filtered signal is then rectified as shown in channel 2.

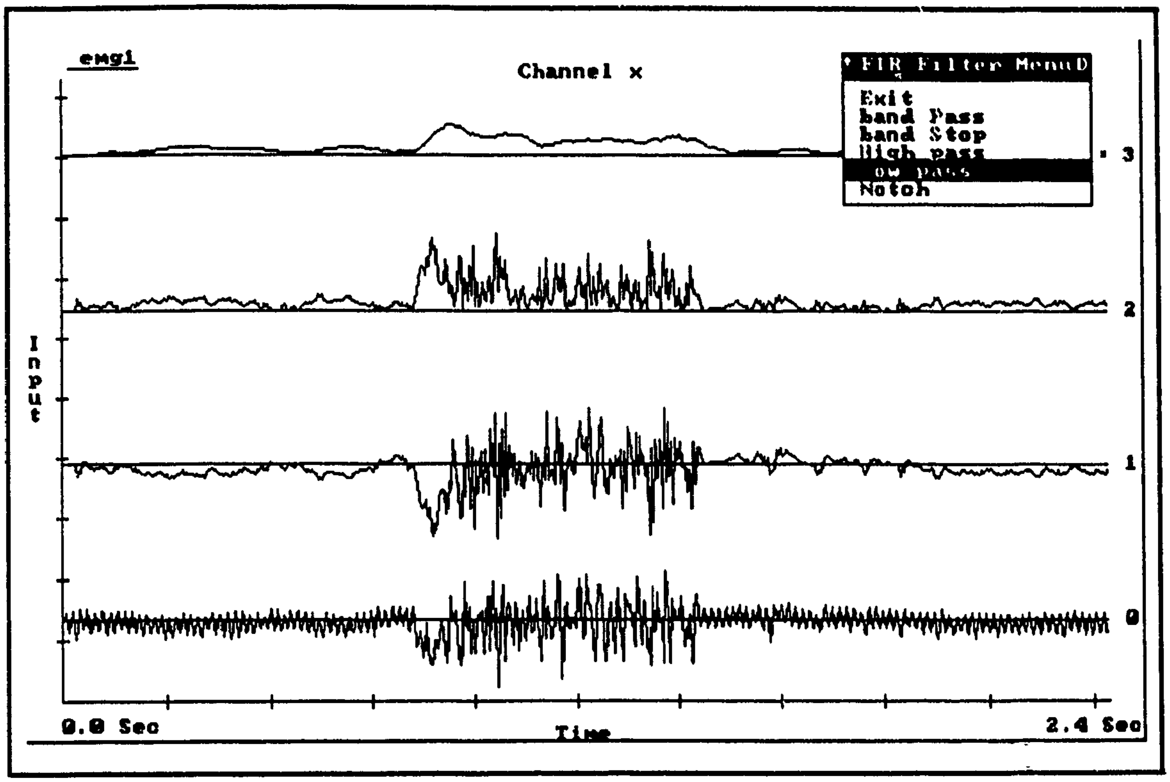

Fig 11.5 Emg Signal Low pass filtered

Finally, the rectified signal is low-pass filtered. This whole sequence can be done in one step by choosing ***Envelope*** option in the ***Process Menu.***

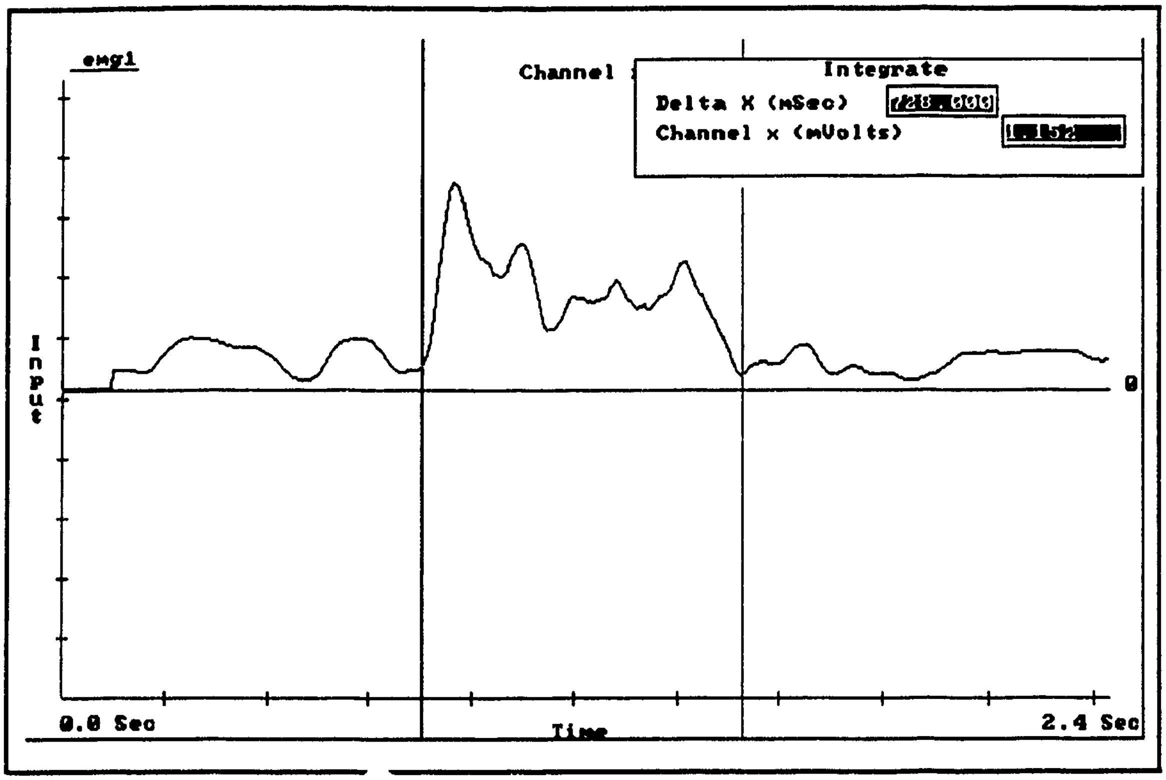

Fig 11.6 Emg Signal integrated

The integrate option is used to get the "amount" of muscle activity. Channel 3 was isolated from the other channels to get a better view.

Chapter 12 Specifications

The system clearly can acquire biological signals. Are the signals being displayed accurate? Do the filters distort the signal? What are the limitations? These questions can be answered by testing the device.

12.1 Testing

There are two groups of tests; general tests and demanding tests. The purpose of the general tests was to see if the device worked. The demanding tests pushed the device to the limits, to find the device limitations.

12.1.1 General Tests

Sine waves of different frequencies and amplitudes were input into the system. The amplitude and frequency of the displayed sine wave was equal to the input sine wave.

Square waves of different amplitudes and frequencies were input into the system. The amplitude and frequency of the displayed square wave was equal to the input square wave.

The FFT routine was tested by using a function generator and viewing the spectrum. The frequency was varied and the spectrum was found to correspond to the input frequency.

Signal and noise were generated, as described in section 5, from the different options in the Generate Menu. These waveforms were used to test the different filters. The FFT was tested a second time by displaying the noisy signal and viewing the spectrum. The noisy signal was filtered and the spectrum of the filtered signal displayed. It was clear that the noise was removed.

The system passed the general tests.

12.1.2 Demanding Tests

The demanding tests are used to gain the actual specifications of the device.

A channel is configured with one of the eight front-end filters. A sine wave of an amplitude A is input into the system. The frequency is varied and the frequency response is plotted (A vs Frequency). This is done for all the front-end filters.

While a noisy signal is input into the system the noise amplitude is varied untill the signal of interest is masked out completely from the noise.

The phase response and frequency response is displayed for the different types of filters. Since some of the filters distort the signal (i.e the IIR filters don't have linear phase). It is up to the user to decide whether the filter characteristics are appropriate for the particular application.

12.2 Current Specifications

- 8 channels
- 8 kHz aggregate sampling rate
- Preset IA gain from 100 to 1000 (total gain 1000 to 10000)
- 86 dB CMRR (total gain of 1000)
- 100 dB CMRR (total gain of 5000)
- 106 dB CMRR (total gain of 10000)
- 8 front-end filters (cutoff range from 50 Hz to 4000 Hz) available for anti-aliasing of one channel.
- 30 mA supply current
- 5V 500 mA-Hr rechargable batteries (around 15 hours continuous use)

12.3 Comparison

Table 12.1 Comparison

Parameter	BDAS	Beckman R511A	Thought Tech	Grass P15	DISA 15C01 EMG Amplifier
CMRR	100 dB Max Gain	>100 dB Max Gain (60Hz)	>140 dB	Unavail	> 100 dB
Input Impedance	>1000000 Megohms	> 2 Megohms	> 1000000 Megohms	> 200 Meg-ohms	> 1500 Megohm
Input	AC	DC	AC	AC	AC
Noise	5.5 uVpp	1 uV RMS	Unavail	20 uVpp	0.7uV RMS
Channels	8	2, 3, or 4	4	1	1
Power	Battery	120 Vac	Battery	Battery	120 Vac
cutoff filters	8 from 50 Hz to 2000 Hz	DC - 125 Hz DC - 70 Hz	fixed 7 Hz	0.1 Hz - 50 kHz	0.05 Hz - 30 Khz
Amplification	1000 to 16000 (preamp + amp)	output : 1 Volt RMS full scale	Unavail	10, 100, 1000	output : > 10 Vpp
Sampling Rate	Max 8 kHz / all channels	Not Applicable	20 Hz / Channel	Not Applicable	Not Applicable

Except for the Thought Technology system all the other devices would view the output using an oscilloscope or chart recorder. The CMRR of BDAS is average and if needed can easily be increased with the use of higher-quality IA's.

Thought Tech.'s system is very limited. No raw signal acquisition is possible. Hardware based rectification and filtering limit the analysis and processing possibilities. The software available is limited due to the hardware implementation. It has very good active electrodes that have high CMRR. BDAS can use active electrodes connected through the external port. These active electrodes can be purchased separately from manufacturers.

DBAS is portable and can be used easily with a laptop to provide a mobile system. Only Thought Tech.'s system is portable, but se limited by hardware.

12.4 Limitations

Although one of the case studies involved ECG, this device is not designed for accurate ECG measurements. In many ECG devices there is a lead connected to the right leg. The lead connected to the right leg is a combination of the three limb leads 180 degrees out of phase is feed back to the subject through the right leg lead. This significantly reduces noise. BDAS is a general purpose device and does not contain this composite right leg lead.

The BDAS IA's are capacitively coupled. Therefore DC measurements are not possible through the IA's. However, the external port has four channels available for direct DC measurements.

12.5 Future Enhancements

The system is modular so that parts can be enhanced or replaced easily. For example, the IA's are very simple and basic amplifiers. If better performance is needed (i.e noise rejection) then that unit can be replaced easily because it is on a separate board. There are many IA amplfiers that are contained completely on one chip. Those IA's have better performance but they are relatively expensive.

If sampling frequency is not high enough, a faster microcontroller could be substituted such as the MC68HC16. Serial transfer rates can be increased, as needed for multi-channel acquisitions. If the serial port is still too slow then parallel port transfers could be used to increase throughput.

BDAS presently comes with a variety of simple filters. More elaborate and efficient filters can be added to increase the processing capabilities.

The external channels provide an easy way for improvement without the need to change existing product.

Chapter 13 Conclusion

The microprocessor revolution gives us the opportunity to use relatively inexpensive microcomputer systems to monitor and analyze biological systems. Digital signal processing techniques allow for different analyses based on the same raw data without having to resort to reacquiring the data.

BDAS was designed to acquire biological signals. Two different hardware versions were designed and developed. Version 1 is a plug in card for the PC and version 2 is a stand alone remote unit. The remote unit can capture data without being connected to the PC. This mobility allows subjects to be observed in actual environments, as opposed to an artificial laboratory setting.

The software for both versions is the same with the exception of the hardware interface routines. The software consists of over 11,000 lines of C++ code, 1,500 lines of MC68HC11 assembler code and 1,000 lines of 80386 assembler code. The software is divided into two parts; data acquisition and post processing. Different options such as sampling frequency, number of channels and signal level can be configured. Post processing of the signal can extract different information from the acquired signal. By doing as much processing in software as possible, a lot of flexibility is achieved. During acquisition, the signal can be filtered and both the raw and filtered signals displayed and stored in a file. Real-time spectrum analysis can be done on one channel. A tone can be switched on to vary in frequency with the input signal magnitude. This can be used in bio-feedback applications.

BDAS is a general purpose device. The design is modular and flexible, allowing the system to be modified easily for different applications.

Chapter 14 References

Basmajian, J.V. Computers in Elecıromyography. Reading, Mass. Butterworth & Co (Publishers) Ltd, 1975.

Basmajian, J.V., and Deluca, C.J. Muscles Alive. Williams & Wilkins, 1985.

Berry, John. The Waite Group's C++ Programming. Indianapolis, IN. Howard W. Sams & Co. 1988.

Budd, Timothy. Introduction to Object-Oriented Programming. Addison-Welsley 1991.

Bukstein, Edward J., Introduction to Biomedical Electronics. Indianapolis, IN. Howard W. Sams & Co. 1973.

Campbell, Joe. C Programmers Guide to Serial Communications. Indianapolis, IN. Howard W. Sams & Co. 1988.

Eggebrecht, Lewis C. Interfacing to the IBM Personal Computer. Indianapolis, IN. Howard W. Sams & Co., 1986.

Ellis, Margaret A., and Stroustrup, Bjarne. The Annotated C++ Reference Manual. AT&T Bell Telephone Laboratories, 1990.

Faulkenberry, Luces M. An Introduction to Operational Amplifiers with Linear IC Applications. John Wiley & Sons, Inc., 1982.

Goldman, Marvin J. Principles of Clinical Electrocardiography. 12th ed. LANGE Medical Publications, 1986.

Gowitzke, Barbara A., Milner, Morris, Scientific Bases of Human Movement. Baltimore, Williams & Wilkins, 1988. p342 - 348.

Greenfield, Joseph D. Practical Digital Design using ICs. John Wiley & Sons Inc., 1983.

H''dges, Hodges A., and Jackson, Horace G. Analysis and Design if Digital Integrated Circuits. McGraw-Hill Inc., 1988.

Lemone, Karen A. Assembly Language & Systems Programming For the IBM PC. Little Brown and Company, 1985.

Lockhart, Gordon, Basic Digital Signal Processing , 1989.

Lynn, Paul A., and Fuerst, Wolfgang. Introductory Digital Signal Processing. John Wiley & Sons, 1989.

Mathieu, Pierre A., and Sullivan, John S. Frequency Characteristics of Signals and Instrumentation: Implication for EMG Biofeedback Studies Biofeedback and Self-Regulation, Vol 15, No. 4, 1990, 335-352.

Mano, Morris M. Digital Design. Englewood Cliffs, New Jersey Prentice-Hall Inc., 1984.

Metzger, Daniel L. Microcomputer Electronics. Englewood Cliffs, New Jersey Prentice-Hall Inc., 1989.

Motorola, Optoelectronics Device Data. 1989a.

Motorola, HC11 M68HC11 Reference Manual. Englewood Cliffs, New Jersey Prentice-Hall Inc., 1989b.

Motorola, Fast and LS TTL Data, 1989c.

Motorola, CMOS Logic Data, 1990a.

Motorola, Linear and Interface Integrated Circuits, 1990b.

Oppenheim, Alan V., and Schaefer, Ronald W. Discrete-Time Signal Processing. Prentice Hall Inc, 1989.

Ott, Henry W., Noise Reduction Techniques in Electronic Systems. New York, John Wiley & Sons, 1987.

Parks, T. W., and Burrus, C. S., Digital Filter Design New York, John Wiley & Sons, 1987.

Rector, Russel, and Alexy, George. The 8086 Book, Berkely, California, Osborne/McGraw-Hill, 1980.

Sargent III, Murray, and Shoemaker, Richard L. The IBM PC from the Inside Out. Addison-Welsley, 1986.

Sedra, Adel S., and Smith, Kenneth C. Microelectronic Circuits New York, N.Y., CBS College Publishing, 1987.

Sheingold, Daniel H. Transducer Interface Handbook, Norwood, Ma, Analog Devices, 1981.

Skalak, R., and Chien, S., Handbook of Bioengineering McGraw-Hill Inc., 1987, 34.1 - 34.20.

Tortora, Gerard J., and Anagnostakos, Nicholas P. Principles of Anatomy and Physiology. New York, N.Y., Harper & Row, 1981, 32-49, 288-293.