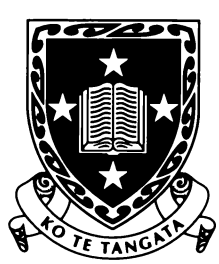
The Auditory Evoked Response as an Indicator of Stress in Free-Ranging Animals

A thesis submitted to the
University of Waikato
for the degree
of
Doctor of Philosophy in Physics
by

Philip James Harris



The University of Waikato
Te Whare Wānanga o Waikato

Hamilton, New Zealand
March 1999

"Hallo!" said Piglet, "what are *you* doing?" ...

"Tracking something," said Winnie-the-Pooh very mysteriously.

"Tracking what?" said Piglet, coming closer.

"That's just what I ask myself. I ask myself, What?"

"What do you think you'll answer?"

"I shall have to wait until I catch up with it," said Winnie-the-Pooh.

Winnie-the-Pooh, by A.A. Milne

The Free-Range Physiological Monitor recording Auditory Evoked Response from a freely-behaving sheep

Abstract

Technology that allows remote monitoring of animals in the field, while freely behaving, would remove many of the problems imposed upon stress studies by experimenter intervention. The acquisition of data from free-ranging animals is difficult, but has significant advantages for studies of stress, nutrition, and behaviour in domestic animals.

To perform this remote monitoring function I developed the Free-Range Physiological Monitor, the FRPM. The FRPM is a state-of-the-art ambulatory physiological monitor, designed to enable researchers to record aspects of the stress response simultaneously and on a number of free-ranging animals. The FRPM incorporates a powerful microprocessor to analyse the recorded data on the animal itself, and can operate indefinitely in the field by solar recharging of its internal batteries. It incorporates a high-speed radio modem with a range of up to 2km to allow the transmission of measurements in real-time to a base computer and to communicate with the researcher.

In its current configuration the FRPM enables the assessment of central nervous system activity through the recording of both the electroencephalogram and the Auditory Evoked Response (AER), allows the monitoring of cardiac (heart) activity through the electrocardiogram (ECG), and provides support for measurement of additional stress parameters such as temperature, respiration rate, and physical activity. The long-range and extended duration capability, and the types of information that can be measured, combine to make the FRPM a unique tool for the welfare assessment of free-ranging animals. The potential of the technology for application in animal welfare research is widespread.

Following development of the FRPM it was considered necessary to validate its design in an experimental setting. As an initial application, the use of the AER as a potential indicator of stress in freely behaving sheep was investigated. This application was considered "high-risk" in that although there was no unequivocal evidence of such a link, there was considered sufficient evidence in the literature to justify a study into the possibility. This application also provided an ideal environment to test the capabilities and reliability of the FRPM.

As the AER originates from the brain itself, it was hypothesised that any change in the operation of the brain, such as that associated with stress, might cause the AER to be modified. To test this hypothesis 12 Romney-cross sheep were individually instrumented with the FRPM, and AER responses recorded as each animal was exposed to potential stressors. The AER results were complemented by heartrate, also recorded by the FRPM, and behaviour observations which helped to provide additional insight into the animals' response to the stressors. In total, each animal was exposed to four separate stressors. These comprised of Control (no stressor), Distraction (a moving, noisy water sculpture), Isolation (separation from the flock), and Fear (exposure to a dog) stressors.

Based on the data acquired in this study, the use of the AER as an indicator of stress appears to be valid, however a larger study would be required to establish a robust relationship. Waveform sequences acquired from individual animals subjected to stressors showed considerable modification during exposure to the stressor and, in several cases, suggested evidence of recovery-type trends upon stressor application and removal. However, this was not the case for all animals in the experimental group, neither was it the case with the acquired heartrate and behavioral data.

These results suggest that different subjects can show quite individual responses to the same stressor, making the creation of a general model for changes in the AER as a quantifier of stress difficult. However, the AER may be a useful measure within an individual and when combined with more conventional measures such as heartrate and behaviour, which in themselves can be difficult to interpret, the AER should provide additional insight into the stress responses of freely-behaving animals.

Acknowledgements

There are many people to whom I am very grateful and without whose help the success of this research would not have been possible.

First, I would like to thank my supervisors, Christian Cook, Peter Schaare, and Jon Henderson for their guidance throughout my research. I am particularly indebted to Christian for his support and encouragement, and for suggesting an investigation into the link between stress in the Evoked Response as an initial application of the FRPM. I would also like to thank Andrew McGlone for many thorough and helpful discussions on data analysis, and Neville Gregory for reviewing this manuscript and making many helpful suggestions.

I would like to thank both Kelly Drake and Steve Payne for their many long hours of assistance during the animal trials involved in this research.

The financial support from both the Agricultural and Marketing Research and Development Trust (AGMARDT), and the Horticultural and Food Institute of New Zealand (HortResearch), is gratefully acknowledged. Without their support this research would not have been possible.

I would also like to thank Kevin O'Donnell for providing me access to the resources of the Technology Development Group, HortResearch, and the staff of the Technology Development Group for their encouragement and help. Without access to these resources the technology aspects of my research would have been severely restricted.

I would like to thank my family, particularly Mum and Dad, for their continued encouragement and support throughout my university education. The encouragement and support of the Crocker family is also greatly appreciated.

Finally, much acknowledgement must go to Melanie, without whose endless patience, encouragement, and support this would have been a much harder road to travel.

Contents

List of Figures

List of Tables

List of Acronyms

AC	Alternating Current
ADC	Analog to Digital Converter
AER	Auditory Evoked Response
ANOVA	Analysis of Variance
BAEP	Brainstem Auditory Evoked Potential
CMRR	Common-mode Rejection Ratio
CNV	Contingent Negative Variation
CPU	Central Processing Unit
DC	Direct Current
DCRR	DC Rejection Ratio
DRAM	Dynamic RAM
DSP	Digital Signal Processor
ECG	Electrocardiogram
EEG	Electroencephalogram
EMG	Electromyogram
EP	Evoked Potential
ER	Evoked Response
FFT	Fast Fourier Transform
FRPM	Free-Range Physiological Monitor
GBP	Gain-Bandwidth Product

GND	Ground
IrDA	Infra-red Data Association
kbs^{-1}	Kilobits per second
LCD	Liquid Crystal Display
mBGA	Mini-Ball Grid Array
MIPS	Million Instructions per Second
NiMH	Nickel Metal Hydride
NSD	Noise Spectral Density
PCA	Principal Components Analysis
RAM	Random Access Memory
RISC	Reduced Instruction Set Computer
RMS	Root Mean Square
ROM	Read-Only Memory
SNR	Signal to Noise Ratio
SPICE	Simulation Program with Integrated Circuit Emphasis
SPL	Sound Pressure Level

Chapter 1 Introduction

The cooler climates of Northern Hemisphere markets for our agricultural products result in intensive farming practices that differ from the extensive outdoor farming practised in warmer Southern Hemisphere countries, including New Zealand. Outdoor farming generally has a favourable welfare image largely due to the freedom of animals to perform a wide range of species-specific behaviours, with the perception of less stress and fewer health problems[1]. However, there are a number of dairy-related issues such as induction of calving, lameness, transport of calves, and thermal stresses which are of concern. Similar welfare issues also exist for the extensive farming of sheep and beef cattle, examples including tail docking, de-horning, and transportation. Putative stresses are not only of importance for welfare reasons: it is also likely that stress itself is a major cost to production[2], is detrimental to product quality and, because of its energetic wastage[3], a substantial sustainability issue. Stress and welfare related issues conservatively cost NZ animal industries NZ$70 M pa in loss of prime export[4] largely due to product quality and yield loss from uncontrolled and unrecognized stress. Potentially this loss could be far greater if consumers feel welfare has been compromised. Stress research on farm animals is therefore becoming increasingly important not just as a result of public concern[5,6] for the welfare of animals, but also in the ongoing need to increase the efficiency of animal production[7].

It is therefore essential to research welfare issues, in particular those unique to extensive farming practice. The study of the free-ranging stock in an extensive farming environment is inherently more difficult than that of confined animals. For example, the stresses imposed on stock by conventional yarding and handling would be difficult to study in confined stock. The aim of this project was to develop tools for the specific purpose of making measurements on free-ranging animals and to then make these tools available to other researchers, for the study of welfare issues related to extensive farming.

Stress is difficult to measure directly, and is often quantified by measuring its effects. However, if it is possible to identify a measure which changes in a consistent and repeatable manner in response to a treatment which is generally recognised as a stressor (e.g. bringing a dog near

sheep), then it can be argued that a reliable indicator of stress for that situation has been identified.

However, behavioral measures alone pose a conundrum because of their subjective anthropomorphic interpretative nature[8,9,10]. To understand, quantitatively measure, and thereby provide a basis for controlling animal stress requires both objective technology and objective markers of stress which allow measurements to be made from free ranging and behaving animals in their natural environment[9,10]. Further, this technology must allow a relative scale of stress to be constructed. Good evidence suggests that the relationship of stress to both welfare and production/product outcomes is an inverted U relationship[11,12] such that too little stress, such as boredom, can be detrimental to a similar extent to that of too much stress.

As a result of these and other physiological factors, estimates of stress which consider only systemic physiological measurements and behaviour have faced difficulties in interpretation. Stress studies need to address the complete axis from brain functions of perceiving and responding to a stressor to the systemic outputs of this central processing and the feedback from these actions. My approach was to design a system which could simultaneously examine each of these links in the stress response path; brain function and processing, systemic response and behavioural responses. Furthermore, each of these aspects of stress response had to be able to be measured on free-ranging animals to allow general application to the study of a wide range of potential stressors.

Technology that allowed remote monitoring of animals in the field, freely behaving, would also remove many of the problems imposed upon stress studies by researcher intervention adding additional uncontrolled stress to the situation. The acquisition of data from free-ranging animals is difficult, but has significant advantages for studies of stress, nutrition and behaviour in domestic animals.

To perform this remote monitoring function the Free-Range Physiological Monitor (FRPM) was developed to enable the researcher to record several aspects of stressor response simultaneously and on a number of free-ranging animals. The FRPM is capable of measuring central nervous system function, heart activity, and with suitable sensors, hormone levels and behavioural activity. The FRPM incorporates a powerful microprocessor to analyse the recorded data on the animal itself, while a radio telemetry system provides the ability to transmit measurements in real-time to a base computer and to communicate with the researcher.

The long-range and extended duration ambulatory capability, and the types of information that can be measured by the FRPM, combine to make the FRPM a unique tool for welfare assessment on free-ranging animals. The potential of the technology for application in animal welfare research is very broad.

Following development of the FRPM it was considered necessary to validate its design in an experimental setting. As an initial application, the use of the Auditory Evoked Response[13] (AER) as a potential indicator of stress in freely behaving sheep was investigated. This application

provided an ideal environment to test the capabilities and reliability of the FRPM. The use of the AER for the measurement of stress in free ranging animals was initially suggested by Dr Christian Cook (HortResearch TDG, Hamilton, New Zealand) based on publications linking changes in the AER with workload stress in human subjects[77,78].

The Evoked Response consists of a series of electrical peaks that occur in the electroencephalogram (EEG) of a subject under test at consistent time intervals after a stimulus is presented to the subject. These peaks are usually grouped into three ranges according to their latency, the time at which they occur following the stimulus. These are short latency, the mid latency and the late latency wave peaks. The short and mid latency components generally represent the conduction of neural information from the stimulus to the cortex of the brain while the late latency components appear indicative of stimulus processing at higher cortical levels of the brain and are very much influenced by psychological state. By presenting varying stimuli to the subject or by changing the subject's environment, the relative importance of stimuli can change dramatically as the processing of stimuli within the central nervous system adapts to the changing situation.

The AER was suggested interesting as a potential measure of stress as it was considered in the literature to be robust, repeatable and the stimulus easy to apply by a speaker or earpiece in the ear of the subject. If changes in the AER could be linked to stressors imposed on an animal, then these features would help make the AER a suitable alternative to more conventional physiological indicators for long-term ambulatory monitoring of stress in free-ranging, freely-behaving animals.

Therefore the following aims were set for this thesis:

1. Development of a flexible instrument platform that could be attached to a free-ranging animal for the ambulatory acquisition and analysis of arbitrary physiological signals
2. To investigate the use of the AER as a measure of stress in free-ranging animals, in this case sheep, with the developed instrument providing the enabling technology to allow such complex recordings to be made in the field.

The FRPM and the Evoked Response projects were funded by the Foundation for Research Science and Technology as part of the 'Measuring Stress In Unconfined Animals' program being undertaken at the Technology Development Group (TDG), Ruakura. TDG is a science group of The Horticulture and Food Research Institute of New Zealand (HortResearch).

1.1 Thesis Outline

This thesis is divided into eight chapters. Chapter 2 introduces the Evoked Response and its generation in the brain and suggests that the Evoked Response could potentially be used as an indicator of anxiety in an instrumented subject. Chapter 3 describes the development of the Free-Range Physiological Monitor, or FRPM, from initial requirements to implementation and application. Chapter 4 describes the design of an amplifier suitable for measuring Evoked Response signals from initial simulation by computer and theoretical performance through to final measured performance. Chapter 5 discusses the development of two algorithms, the first for peak detection and the second for statistical analysis of waveform shape, used for analyzing Evoked Response waveforms. Chapter 6 describes the experimental methodology used in an experiment to determine if changes in the Auditory Evoked Response in free-ranging sheep can be used as an indicator of stress. Chapter 7 presents the results of this experiment. Finally, Chapter 8 presents an overview and discussion of the work presented in this thesis, together with suggestions for improvements, and outlines directions for future work.

The first appendix, Appendix 1, contains the abstracts of publications describing the work contained in this thesis that were presented at both national and international conferences.

The final two appendices attached to this thesis provide more detail on the design and implementation of the FRPM. Appendix 2 describes the FRPM hardware in detail while Appendix 3 describes the FRPM software architecture.

Chapter 2 The Auditory Evoked Response and the Measurement of Anxiety

The earliest known record of electrical activity in the brain was made by Robert Caton who in 1875 reported that "Feeble currents of varying direction pass through the multiplier when electrodes are placed on two points of the external surface [of the brain]"[13]. Caton's subjects were cats, monkeys and rabbits and his recording apparatus consisted of a galvanometer with optical magnification. This electrical activity in the brain is now called the electroencephalogram or EEG. In the following years several researchers, often working completely independently, investigated the EEG in both animal and human subjects and noted changes in electrical activity due to external stimuli as well as what now would be called evoked responses.

2.1 Physical Structure of the Brain

The brain[14] (Figure 2.1) consists of four anatomically distinct parts, these being the left and right hemispheres, the cerebellum, and the brainstem. The two hemispheres are separated by the longitudinal fissure but are connected by a large connective band of fibers called the corpus callosum. The outer surface of the cerebral hemispheres are composed of neurons that form the cerebral cortex. The surfaces of the hemispheres are highly convoluted, increasing the surface

area (and thus the number of neurons) to more than twice that of a smooth sphere of the same size. Beneath the surface of the cortex nerve fibers lead to the other parts of the brain and the body. Because of their colour the regions composed of neurons are known as grey matter while the fibrous connective tissue is known as white matter.

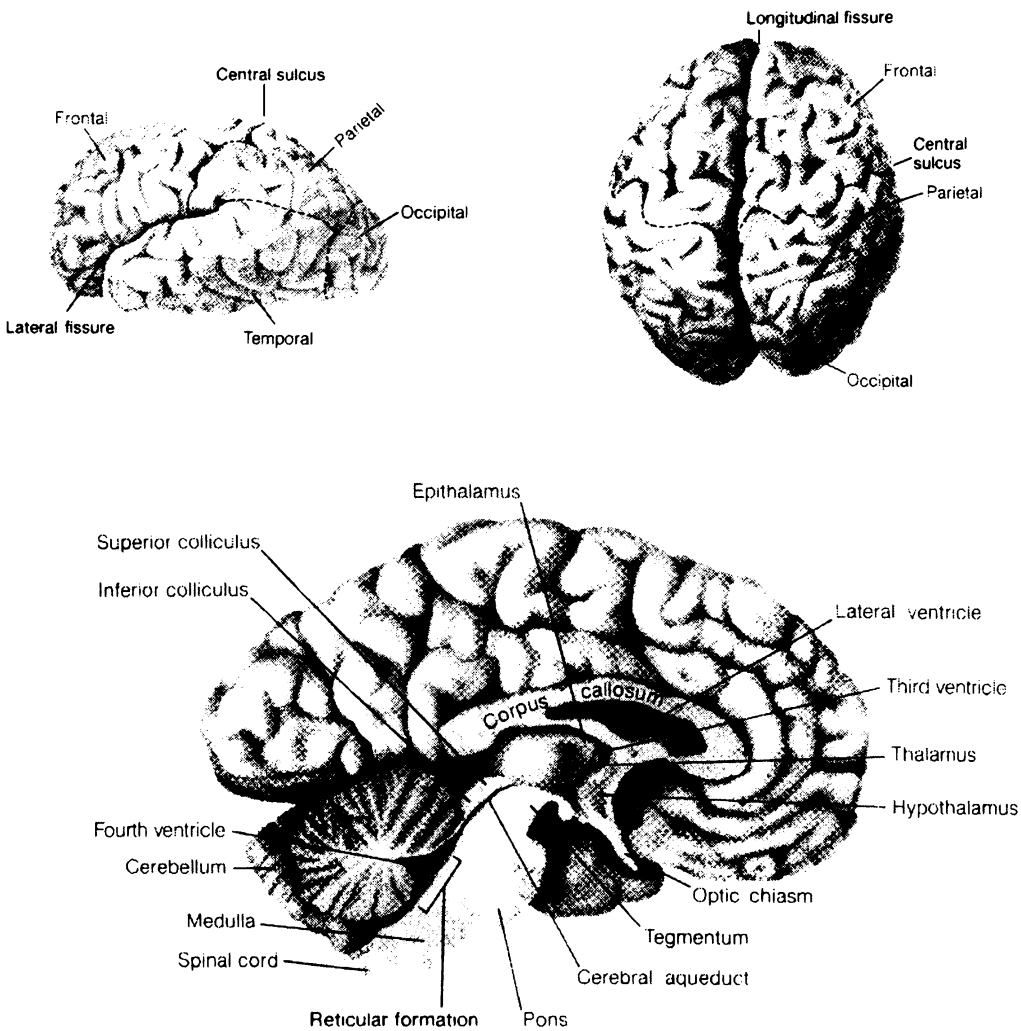


Figure 2.1 – The Brain (adapted from Kolb, 1990[15])

Certain functions have been differentially located to different regions of the cortex[16]. For example, a strip of cortex residing between the central sulcus and the post-central sulcus in the pariatel lobe of the cortex mediates the initial reception of sensory information[17].

Microscopic examination of dyed sections of brain reveal the cerebral cortex as an intricate network of fibers and neurons. The white matter is seen to be composed of fibers each wrapped in an insulating sheath of myelin. This insulating sheath of myelin is responsible for the white color of the connective tissue. There are several different types of neurons but the basic structure is similar to that shown in Figure 2.2. The dendrites serve as connections to other neurons and can spread through a considerable volume of the cortex.

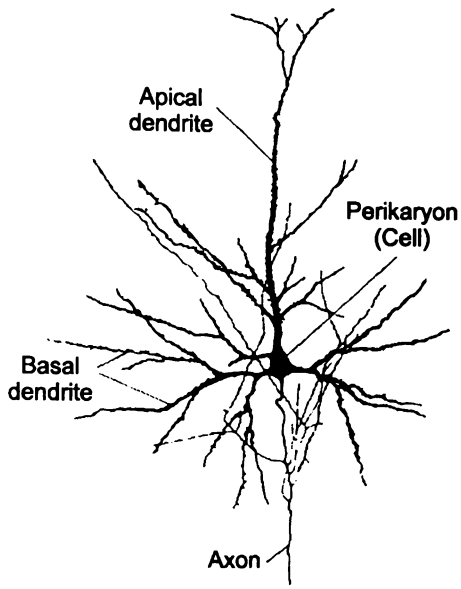


Figure 2.2 – A Pyramidal Neuron from the Cat (adapted from Scholl, 1956[18])

2.2 Electrical Activity in the Brain

The basic information processing unit in the neurophysiological system of any living organism is the nerve cell, or neuron. The function of the neuron is the acquisition, processing, and transfer of information[19]. A schematic representation of a neuron is shown in Figure 2.3.

Signals from the sense organs are transmitted to the brain along nerve fibers as a series of pulses. The intensity of the external stimulus determines the frequency of these pulses. As these nerve fibers enter the cerebral cortex extensive branching occurs, spreading the incoming pulses over an appreciable area of the cortex. The branched fibers do not connect directly to the neurons but instead terminate on cell bodies and dendrites by way of small swellings called synaptic knobs. Each neuron has many synaptic knobs and can receive impulses from many fibers. The pulses are transmitted from the fibers, across the synaptic membrane into the cell structure. Because of the extensive branching, a single pulse can act with varying degree on as many as 5000 neurons which in turn can act on many more neurons.

The pulses transmitted along a fiber take the form of a transient disturbance of the resting potential along the fiber. This action is caused by a very rapid change of membrane permeability to sodium ions followed by a recovery period, the whole process taking less than a millisecond. As the propagated action potential reaches the cell the permeability of the cell membrane can increase suddenly causing the cell to emit (or 'fire') a pulse of its own. This pulse spreads along the dendritic branches and causes the release of transmitter substances where the branches terminate on other cells.

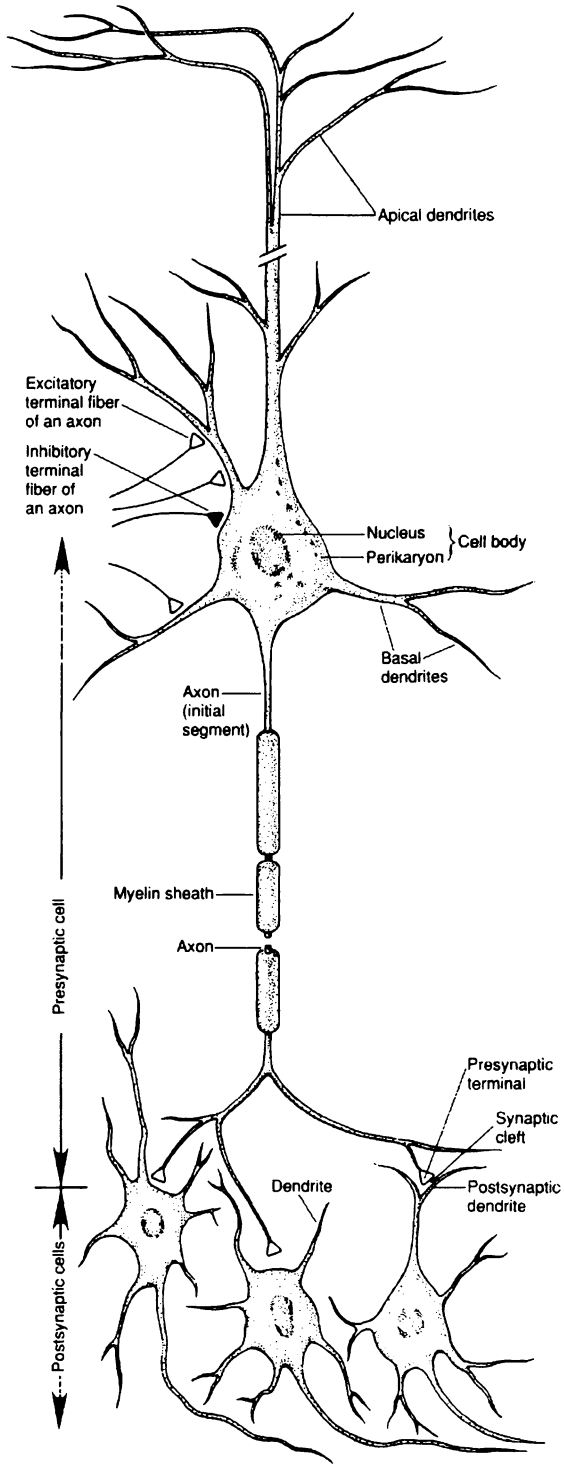


Figure 2.3 – The Neuron (adapted from Kandel, 1991[20])

The transmitter substance can be either inhibitory, making the receptor neuron less likely to fire, or excitatory, increasing the probability that the receptor neuron will emit a pulse of its own. Since many neurons can be connected to a single receptor neuron a change at a single synapse is unlikely to change the operation of the neuron, however change becomes more likely when many neurons release transmitter substances simultaneously. The actions of the transmitter substances

can cause intracellular postsynaptic potentials as large as 30mV, causing external currents to flow between the upper and lower layers of the cortex[21].

2.3 Origins of the Cortical Electroencephalogram and its Relation to the Scalp Electroencephalogram

Action potentials are recorded using microelectrodes placed either very close to or within an individual cell. The action potentials are considerably larger in amplitude and have a shorter time-scale than the features that can be observed in an EEG trace. Even very small EEG electrodes placed on the cortex must, by nature of their size, cause any electrical activity recorded to be the average activity of thousands of cells. The perfectly synchronized firing of this many cells seems unlikely, making it doubtful that the cortical EEG is comprised of large numbers of action potentials. Li and Jasper[22] provided support for this view by showing that in cats the EEG could still be recorded under deep anesthetic when action potentials were abolished. Most evidence suggests that the cortical potentials are due to the excitatory or inhibitory postsynaptic potentials developed by the cell body and the large dendritic structures of pyramidal neurons[23]. The relatively long time constants of these potentials are comparable to features in the EEG and their summation is aided by the columnar structure of these neurons reaching as they do from upper to lower layers of the cortex.

The electrical fields that generate the cortical EEG are very localized due to poor conductivity in the cortical tissue. This can produce large differences in recorded electrical activity even if the recording electrodes are only 1 or 2mm apart. However, this effect does not apply to scalp recordings since the EEG recorded from the scalp is the average of many small active areas on the cortical surface underneath the recording electrode. If the cortical activity is synchronous over a large area underneath the electrode the amplitude of the scalp EEG is similar to that of the cortical EEG. However, if the area of synchronous active cortical tissue is small the amplitude of the resulting scalp EEG will be much smaller than a recording made from the cortex[24].

The scalp EEG has a bandwidth range from DC to around 100Hz, with the major power distributed in the range of 0.5 to 60Hz. The amplitude of the scalp EEG is usually in the range of 2 to 100μV. The EEG power spectral density varies greatly with physical and behavioral states of the subject. For this reason the EEG has been a major tool in neurological diagnosis for many years[25].

2.4 The Evoked Response

The electrical activity of the brain evoked by a sensor stimulus is known as the Evoked Response (ER) or Evoked Potential (EP)[26]. It is usually recorded from electrodes positioned over the sensory region of the brain corresponding to the type of stimulus[27]. A sensory stimulus results in two kinds of potential changes in the EEG. The nonspecific response is a low-voltage transient, having its maximum amplitude in the region of the vertex. This response is similar for all types of stimulus but becomes less marked when the same stimulus is repeated due to adaptation[28] in the subject. The specific response occurs at some latency after the stimulus has been applied and has a maximum amplitude in the cortical area appropriate to the type of stimulation. The Evoked Response is very low in amplitude, typically in the range of 0.1 to 10μV (although the Visual Evoked Response can be much larger), and is usually obscured by the ongoing EEG which may be an order of magnitude larger.

There are three general types of Evoked Response, categorized by the type of stimuli used to evoke them:

The Auditory Evoked Response (AER) is typically evoked by brief auditory stimuli (clicks or short-duration tone bursts) which activate the neural pathways responsible for auditory processing. The latency, amplitude, and shape of components of the AER are disturbed in certain neurological disorders and have also been used for intraoperative monitoring of subjects since the early components are not significantly altered by anesthetic agents[29].

The Visual Evoked Response (VER) is typically evoked by either stroboscopic flashes or high-contrast pattern reversal (e.g. a checkerboard pattern where the black and white squares alternate position) which causes a high level of electrical activity in the neural pathways responsible for visual processing. Due to the high repeatability of the pattern reversal technique across normal subjects, the VER has become a useful clinical tool for the detection of cerebral dysfunction[30].

The Somatosensory Evoked Response (SER) is typically evoked by tactile stimulation, usually on the limbs of the subject. The stimulus can be provided by a mechanical device or by electrical stimulation, sometimes directly at the target nerve. This technique differs from both the AER and VER in that it also provides information on neural activity outside the brain. In particular, it allows measurement of the nerve conduction velocities of afferent impulses in the peripheral nerve fibers in cases of neuropathy. This makes it clinically useful for distinguishing organic and functional sensory loss, and also for the diagnosis of disorders such as multiple sclerosis and peripheral nerve lesions[31].

An Evoked Response waveform consists of a number of components, the appearance or amplitude of which might be dependent on the conditions under which the waveform was evoked. Each component has a certain latency caused by the nerve conduction velocities and the distance any signal has to travel. The nerve conduction velocity can be affected by such environmental factors as temperature[32], subject attention, or fatigue[77]. The various Evoked Responses are largely

similar between species with most of the difference being due to differences in the nervous system path lengths[33].

Generally the short latency components are more dependent on the physical characteristics of the stimuli (e.g. brightness, loudness, etc.) while the longer latency components are more dependent on the conditions of presentation.

Separation of the Evoked Response from the other electrical activity in the brain is traditionally accomplished by signal averaging synchronized with the presentation of the stimulus[34]. Since the electrical response of the brain to the stimulus occurs at consistent latencies from the time of presentation of the stimulus, whereas the background activity does not, the averaging effectively extracts the stationary Evoked Response signal from the non-stationary background noise. Stimuli are presented repetitively until the averaging process produces a sufficiently clear Evoked Response waveform. This technique gives a significant improvement in the signal-to-noise ratio and permits the recording of very small Evoked Responses with amplitudes of less than 1μV in the presence of EEG with amplitudes of several hundred μV. However, extraction of the Evoked Response by averaging makes three assumptions[35]:

1. The signal and noise sum linearly to produce the recorded waveform
2. The signal waveform is the same for each repetition of the stimulus
3. The noise waveforms are sufficiently irregular from stimulus to stimulus that they can be considered as statistically independent samples of a random process

For the purposes of averaging the 'ideal' noise is stationary (i.e. different segments of the noise have similar means and variances), and independent (i.e. the data points in any segment of a recording do not co-vary with those in another segment). Under these conditions the averaging will reduce such an ideal noise to a level in the final waveform that is directly proportional to the root-mean-square (RMS) value of the original noise and inversely proportional to the square-root of the number of repetitions. This is the 'square-root' rule of averaging[36].

In real situations, however, the background noise does not always fulfill the assumptions of averaging. The noise is often non-stationary because of occasional traces in which the noise has a much higher amplitude then usual. For example, some traces may be contaminated by high levels of muscle artifact due to movement, coughing or swallowing. If these traces occur only infrequently they can be rejected from the averaging process to significantly improve the result. If these abnormal traces occur frequently the only course of action is to consider them part of the background noise and increase the number of traces obtained to compensate for the increased noise.

Another potential problem is that the different samples of noise may not be independent because of rhythmic activity during the recording process. Sources of such activity include chewing, electrode motion due to an ambulatory subject, or external electrical interference. One approach

to this problem is to randomize the inter-stimulus interval over a range greater than the period of any repetitive signal in the background noise. Another solution is to use a constant inter-stimulus interval that is not equal to any multiple of the period of the repetitive noise[37].

The Evoked Response signal itself may also change during the acquisition process due to boredom, fatigue or adaptation on the part of the subject[38]. These changes distort the resulting averaged waveform and therefore counter any enhancement of the signal-to-noise ratio brought about by increasing the number of stimuli presented. However, it is possible to increase the number of traces without increasing the duration of the recording by decreasing the inter-stimulus interval. For most Evoked Responses, increasing the stimulus rate results in a decrease in the amplitude of the response. Therefore a trade-off must be made between total recording time, the number of stimuli presented to the subject, and the resulting signal-to-noise ratio.

Among the three general classes of Evoked Responses (Auditory, Visual, and Somatosensory Evoked Responses), the AER stands out as being the most suitable for use in free-ranging animals. Recording the Visual Evoked Response from ambulatory animals presents the difficult task of suspending a visual stimulator in front of an animal that may be grazing, or rubbing against trees or fences in the field, while recording the Somatosensory Evoked Response requires the attachment of the (preferably electrical) stimulator over a nerve, probably on the limb of the animal, and maintaining reliable electrical contact for good, repeatable, stimulation. In comparison, recording the Auditory Evoked Response only requires placing a miniature audio speaker in the ear of the animal, and making sure the ear canal is free of obstructions such as wax build-up, before beginning recording. Such an arrangement avoids the risk of the stimulator becoming caught or damaged by objects in the field and largely reduces such concerns as waterproofing and ensuring reliable stimulation.

2.5 The Auditory Evoked Response

The Auditory Evoked Response (AER) is an Evoked Response generated by auditory stimulus to the subject[39]. The stimulus usually consists of either a click or a tone presented via a speaker to either or both ears of the subject. In some studies the stimulus is presented to the ear via bone conduction of the sound through the skull of the subject.

In most species the AER generally takes the shape of the stylized waveform in Figure 2.4 (note the logarithmic time and amplitude scales)[40]. The very small components that occur shortly after the stimulus are the part of the response due to brainstem activity (typically called waves I through VI). Then follows a number of small responses before the large component known as the N100 which represents the first level of cognitive processing within the brain itself. If the auditory stimulus is used as a warning stimulus to prepare the subject for a motor action when a second stimulus occurs the N100 and P300 components are followed by the Contingent Negative

Variation (CNV) which indicates the readiness of the subject to perform some action[49]. In brain death the components beyond wave I disappear[41].

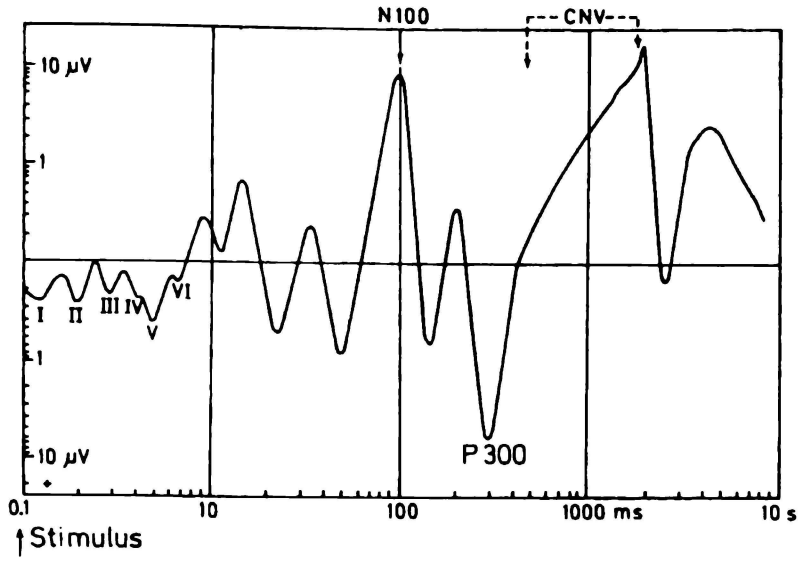


Figure 2.4 – Stylized Auditory Evoked Response (adapted from Cooper, 1980[42])

It is interesting to compare the stylized waveform of Figure 2.4 with a typical AER waveform from a sheep, as shown in Figure 2.5. Note that the time axis only shows the first 100ms and that the amplitude axis is linear, unlike Figure 2.4. Clear similarities between the two waveforms can be seen: a strong positive peak at 10ms latency followed by a negative peak at 20ms are the most obvious, however brainstem activity can also be seen before 8ms, and peaks at 40ms and 60ms are also comparable to the stylized waveform. The subtle differences in latency can be attributed to the variation in the length of the neural pathways between species, while amplitude variation is most likely due to differing electrode placement.

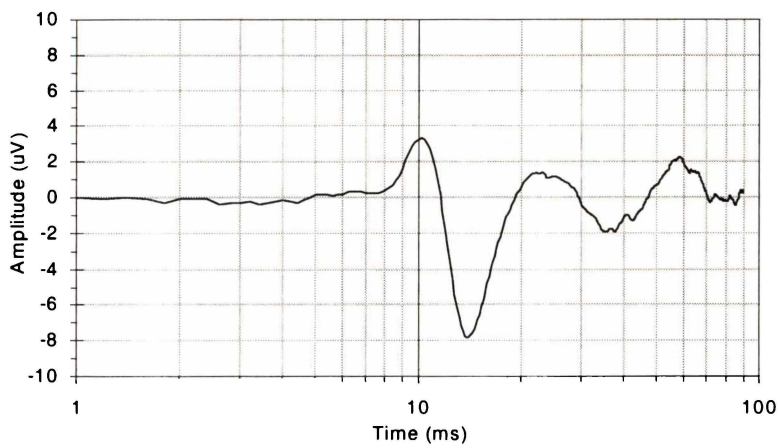


Figure 2.5 – Example Sheep Auditory Evoked Response

Two main types of AER are commonly recorded. These are the Brainstem Auditory Evoked Response (BAER) which records activity up to around 20ms after the stimulus, and a longer-duration recording that records activity up to 1000ms after the stimulus to investigate the long latency components of the Evoked Response such as the P300 or CNV.

The BAER components are very small in amplitude (usually less than 1μV) and are typically evoked by brief auditory stimuli (usually clicks). These components contain relatively high-frequency components and require amplifiers with bandwidths from about 100Hz to 2000Hz to be recorded accurately. This bandwidth requirement usually makes commercial EEG amplifiers unsuitable for the task and special amplifiers must be used instead. Because of the small amplitude of the response large amplifier gains must also be used. Since there is little recovery time after the response, stimuli can be presented at relatively high rates of up to 30 clicks per second. However, care must be taken that any response still present after the trace period does not 'wrap-around' and contaminate the recorded waveform. This can be avoided by random inter-stimulus intervals or careful selection of the amplifier bandwidth.

The amplitudes of the BAER components are dependent on the sound intensity of the click. It is often necessary to use click amplitudes of up to 110dB peak SPL to obtain sufficient resolution of the components in the resulting waveform[43].

The BAER has been used to investigate causes of deafness[44] due to neurological damage, pre- and post-natal neurological development[45,46], depth of anesthesia[47], and the effects of hyperthermia[48], among other applications.

The components consisting of the activity around the N100 and P300 are considered to be part of the cognitive processing of the stimulus by the subject[49]. The N100 in particular can be easily evoked by auditory, visual, or tactile stimuli and has been called the non-specific response. The amplitude of this component is dependent on many factors including the intensity of the stimulus and the attention the stimulus receives. Habituation rapidly reduces the amplitude of this component if the stimuli are presented in a regular manner; however, it never disappears completely. If the subject can be asked, or trained in the case of animals, to pay attention to a particular type of stimulus presented among others (the so-called 'odd-ball' paradigm) the amplitude of the N100 component is increased[50]. The potential field of the N100 is widespread in frontal regions and is maximum at the vertex.

The P300 component has been linked to a variety of decision-making tasks, this being first described by Sutton[51] in 1965. Cooper[52] showed that this component is most easily evoked in a subject by a target selection paradigm in which the subject has to detect infrequent 'target' stimuli which are randomly interspersed with non-target stimuli. The P300 accompanies the detection of the target stimuli with its amplitude dependent on the confidence of detection and the subjective likelihood of the occurrence, this being larger if the targets are infrequent but easily distinguished. The occurrence of the P300 component seems to be dependent on the successful matching of

incoming information with some stored information. The potential field of the P300 component is maximum in the central-parietal regions.

The CNV was originally discovered by Walter *et al.* in 1964[53]. While investigating the evoked responses to pairs of stimuli separated by about one second they discovered a small increase of negativity at the vertex occurred in the inter-stimulus interval. This was only present if the subject had to respond to the second stimulus. This potential was thus called the contingent negative variation. The scalp distribution of the CNV is widespread with a maximum at the vertex.

There have been many attempts to determine the relationships between the CNV amplitude and psychological variables such as attention, anxiety, motivation, and expectancy[54,55,56]; however, none have yet been conclusive. This may be due in part to the extended duration of the experiments (well over an hour in some cases) as a result of the long inter-stimulus intervals required. This could result in 'boredom', fatigue, or adaptation in the subject, potentially changing the experimental environment. Another difficulty lies in the measurement of the baseline psychological variables of the subject. This in itself can take a considerable amount of time.

From the description of the various components of the AER it can be seen that as the latency of the component from the stimulus gets longer, the potential for variation and uncertainty in the component gets larger. This is reflected in published normative data[57]. At the shorter latencies of up to 100ms the neural pathways are typically 'hard-wired' with little variation between individuals, and even between species[33], other than those that reflect the length of the neural pathways and the presence or absence of myelination. This in turn leads to well-defined and repeatable waveforms. With the appearance of the longer-latency components comes a greater dependence of the waveform on the individual subject, the species, and the 'state' of the subject. Reasons for this include differences in the cognitive processing of individual subjects, differences in the relative importance of the sensory pathway for the species, and 'drift' in the cognitive processing due to changes in the psychological state of the subject during extended recording times.

2.6 The Auditory Evoked Response as a Measure of Stress

Stress has long been a major focus among researchers interested in environmental and psychosocial influences on health, both in human subjects and in animals. Despite this interest, stress is a concept that is not rigorously defined. Stress, in a relatively narrow sense, is anything that challenges the maintenance of homeostasis in an organism, whether this challenge is a life-and-death situation, illness, or just a novel environment[58]. The stress response occurs at a number of levels within the body and comprises of the set of neural, endocrine and behavioural adaptations that rise to meet this challenge. At the highest level[59,60] the central nervous system

(CNS), stress related stimuli are received, processed, and integrated with other information to initiate a response "plan" for the organism. This plan is then effected through areas including the neuroendocrine system[61], the autonomic nervous system, motor control systems and behavioural actions, and in the function of the CNS itself (e.g. alerting of the animal)[62]. These in turn produce changes in mobilisation of energy, cardiovascular function, digestion, growth, reproduction, immune response and pain perception[63]. All of these feed back onto the CNS, curtailing or exaggerating the responses, and changing future responses[64]. In this way the stress reaction allows the organism to adapt to the challenge of the stressor: to prepare to fight, to flight, or to learn[65].

Although stress is highly adaptive and in moderate levels provides a mechanism to allow an organism to adjust to a changing environment[66], a process necessary for both survival and evolutionary processes, it is most usefully described from an animal welfare and animal production viewpoint as a pathological process. For example, a useful definition for production animal issues describes stress as *"a process in which environmental demands tax or exceed the adaptive capacity of an organism, resulting in psychological and biological changes that may place the organism at risk from disease*[67]*"*. This definition is useful because it describes stress as depriving the organism of energy that could have been used for growth or disease resistance, making it particularly relevant for use in investigations into the effects of stress on animal welfare and production.

Because stress is difficult to measure directly it is often quantified by measuring its effects. If it is possible to identify a measure which changes in a consistent and repeatable manner in response to a treatment which is generally recognized as a stressor, it is possible to argue that a reliable indicator of stress has been identified. The difficulties of this are:

1. That it is generally difficult to prove that the observed response is indeed caused by stress
2. The measurement may be confounded by other phenomena, e.g. increased physical activity
3. The degree of response to any stressor may be reduced by habituation or increased by facilitation

Because of these and other physiological factors, estimates of stress which consider only physiological measurements and behavior have faced difficulties in interpretation. Stress studies need to address the complete axis from brain functions of perceiving and responding to a stressor to the outputs of this central processing and the feedback from these actions. My approach was to design a system which could simultaneously examine each of these links in the stress response path; brain function and processing, systemic response and behavioral responses. Furthermore, it had to be possible to measure each of these aspects of stress response on free-ranging animals.

The most commonly used psychophysiological methods suitable for measurement of stress in free-ranging animals, are electromyographic[68], cardiovascular, and electrodermal[69] (polygraph) measures. While not commonly used in human studies, the analysis of blood or urine samples for the presence of such stress-related substances as epinephrine, norepinephrine, and cortisol have frequently been used in animal studies but instrumentation for the measurement of these substances in free-ranging animals has only recently been made available[70] and remains bulky and limited in application.

Of the various cardiovascular measures, heartrate is arguably the most frequently used[71]. Although it is subject to environmental conditions and perceptual and physical activity, heartrate is less sensitive than most other visceral responses to measurement artifact and is relatively easy to monitor. While it is generally assumed that exposure to a stressor increases heartrate, this is not always true[72]. Some subjects can show a decrease in heartrate to a stimulus that invokes fear, or even a diphasic pattern in which the heartrate first increases and then sharply decreases[73]. This can make interpretation of heart rate profiles taken during periods of stress difficult, as the response often varies between individual subjects.

There have also been many studies that indicate that the Evoked Response may be used as a measure of stress. The earliest indication of this was a study by Hernandez-Peon *et al.*[74] that showed auditory responses from the cochlear nucleus of a cat were greatly reduced when a cat was presented with live mice but returned to normal when the mice were removed. In human subjects, studies by Drake *et al.*[75] and Bond *et al.*[76] have shown that peak latencies in the recorded BAER of subjects suffering from chronic anxiety were significantly shorter than those of normal subjects. The Evoked Response has also been used successfully as a measure of cognitive processing, or workload, during such activities as display monitoring[77] and air-to-ground attack mission training[78]. These studies show that the Evoked Response also varies with short-term stress, in this case workload.

Human studies that attempt to match anxiety with changes in the Auditory Evoked Response typically consider only the P300 component of the response[79,80,81] and show that the P300 component increases with anxiety. The anxiety is usually due to the threat of mild electric shock, once again suggesting that the Auditory Evoked Response can be used as an indicator of short-term stress.

Although the human studies relating stress and workload with changes in the AER contain no unequivocal evidence of a link between changes in the AER and stress, there was considered sufficient evidence in the literature to justify a study into the possibility of using the AER on free-ranging animals to investigate its potential use as an indicator of stress. Therefore the following hypothesis was proposed for this thesis: **"Does the AER, as recorded from free-ranging sheep, change when the animal is subjected to common stressors?"**.

Few references are available that mention the recording of the AER in sheep. Hill *et al.*[82] has recorded the AER from adult Dalesbred sheep using implanted electrodes and click stimulus. This study states that the AER in sheep is similar to that in other mammals, although there

appears to be a large variability between animals, particularly for components after about 5ms latency from the stimulus. Cook *et al.*[83], Pierson *et al.*[45], and Griffiths *et al.*[46], have performed studies that between them map the changes in the BAER from five weeks prior to birth to seven weeks of age in chronically instrumented lambs to develop an understanding of fetal developmental physiology.

The only reference to the AER being recorded from sheep under any form of stress was a study performed by Mustafa *et al.*, investigating changes in the BAER brought on by hyperthermia[84], in particular changes that might immediately precede heat-induced collapse. However, this was mainly aimed at understanding the onset of heat stroke and death in man.

The Auditory Evoked Response has several potential advantages as an indicator of stress in free-ranging animals:

1. Stimuli can be presented by small speaker or earphone that can be placed discretely in the ear canal
2. The Evoked Response can be measured from a few (usually 3) small electrodes affixed to the skull of the animal with minimal invasiveness
3. The Evoked Response is generated as a result of activity in the brain itself rather than as a secondary effect (for example heart rate) and should therefore be directly affected by psychological state
4. With adequate recording duration, the recorded Evoked Response waveform can display evidence of sensory, cognitive, and motor control processing as a result of a single stimulus

Among the disadvantages of using the AER as an indicator of stress in free-ranging animals lie the following problems:

1. The electrical signal recorded from the electrodes is very small (1-10μV) and can be easily contaminated by noise from muscle activity, damaged electrodes, and electrical interference
2. The presence of large amounts of noise in the recorded signal requires significant signal processing to extract a good estimate of the AER
3. Due to noise, many stimuli may be needed to be presented to the subject over several minutes to obtain a good estimate of the AER by averaging. Therefore the obtained AER may not exhibit short-duration changes in the real AER
4. The extra complexity in stimuli generation, small signal amplification, and signal processing algorithms requires relatively sophisticated instrumentation, at least as compared with conventional stress measures such as heartrate or body temperature

Because animals (sheep) were to be used as the experimental subjects in the present experiment, the parameters of the experiments performed on human subjects had to be modified. In particular the investigation of the P300 as an indicator of stress had to be eliminated. The P300 is an indicator of cognitive processing in the subject[46] and is typically evoked by instructing the subject to perform some task on presentation of the stimulus. Often this task is to count the number of stimulus presented or to identify an unusual stimulus among the stimuli presented[52] (the 'odd-ball' paradigm). In the case of animal subjects this is difficult since the subject must be first trained to perform the task that will evoke the P300 component. Instead it was decided to investigate the early components present at up to 100ms latency from the stimulus as indicators of stress, as these are largely independent from higher brain functioning[85].

Investigating these earlier components also had the advantage of little variation between individuals and the ability to present many stimuli in a short period of time to avoid any drift in the 'state' of the subject affecting the result. The presentation of many stimuli in a short period would also provide an improved signal-to-noise ratio for the recording and help to make the recording more robust against noise caused by movement of the free-ranging animal.

The other major difference from human stress studies was that the stimuli were changed from short-duration tones to clicks. This change was to allow the use of a higher peak stimulus intensities than could be used with continuous tones to ensure a robust response to the stimuli. The short duration click would also help to prevent masking of the early components (<10ms) by stimulator artifact.

If we could develop the Auditory Evoked Response as a measure of how an animal is processing sensory information while freely behaving in the field and we could measure how this changes with behavior and putative stressors then we may achieve three goals:

1. Identification of stress in an objective, graded manner
2. Determination of the relativity of different stressors i.e. how much is normal sensory processing changed?
3. And potentially, an understanding of how stress is processed and perceived

The tool used to attempt an answer the hypothesis presented above would be the FRPM. The following chapters describe both the development of instrumentation suitable for extended duration use recording physiological signals from free-ranging animals, and the development algorithms to search the recorded Evoked Response waveforms for any indicators of stress.

Chapter 3 The Free-Range Physiological Monitor

A fundamental problem in the recording of biological signals from freely-moving subjects over extended periods of time is the large volume of data that can be collected. For example, signals such as electroencephalogram (EEG), electrocardiogram (ECG), and electromyogram (EMG) must be sampled at relatively high sample rates for long periods of time. This problem is further exacerbated when a number of subjects must be monitored simultaneously.

In animal welfare studies researchers are often interested in measuring several such signals simultaneously, for example when investigating models of central nervous system activity or cardiovascular system response. For such studies several animals must be instrumented and the data collected simultaneously from each while the animals are subjected to various stressors. The duration of these recordings can reach several weeks.

The volume of data being recorded poses a number of technical difficulties: any storage medium on the subject is soon filled, the capacity of a radio data channel is soon exceeded, constant transmitter operation soon drains battery power, and the volume and subsequent processing of raw data after the experiment becomes formidable. A solution to these problems is to provide the ability to process the raw data at the subject as it is recorded, with only the summary results being sent back to the researcher.

For example, consider Evoked Response recordings where storage of the raw EEG waveforms from which a single Evoked Response waveform is reconstructed can easily exceed 2MB, collected over a 3 minute period. For continuous Evoked Response recording this may reach 40MB per hour, per animal. If the necessary signal processing could be performed on the animal as the data was recorded this storage requirement can be reduced to around 2kB per recording.

With the addition of a communications link this processed data can be sent back as it is collected, eliminating the need for animal handling to retrieve the recording.

Faced with the problem of remote recording of Evoked Response, I implemented a solution in a physiological recorder, the Free Range Physiological Monitor, or FRPM. The FRPM was developed for initial application to the recording of remote Evoked Response with further potential for future applications including microclimatic monitoring, feeding pattern analysis, and sleep state analysis. Figure 3.1 shows a complete FRPM unit attached to a sheep.

Figure 3.1 – The Free-Range Physiological Monitor

The only other known use of a telemetric system for recording Evoked Response waveforms was a system published by Walter *et al.* in 1967[86] which used a commercial radio-controlled model aircraft system to transmit the live EEG signal to the receiver for recording. A separate radio-control link allowed stimulus clicks and tones to be generated in the subjects ear via an earphone in synchronization with the averaging process required to extract the Evoked Response waveform. This system also allowed the recording of both heart rate and respiration rate.

Compared with the limited instrumentation available for ambulatory physiological recording, features of the FRPM include its robustness, recording flexibility, and high-speed data processing. The ability to communicate with many units simultaneously over the same digital telemetry link, combined with indefinite use via solar recharging, make the FRPM unique among available ambulatory physiological instruments.

3.1 System Requirements

The initial requirements for the FRPM were that it could be attached to the back of an animal for several weeks at a time and that multiple FRPM units could be used to simultaneously record biological data from several animals, with the data communicated back to the researcher via radio link. The FRPM was also required to be able to perform such computationally intensive tasks as EEG sleep-state analysis.

As the FRPM was to be carried on farm animals in typical farming situations the size and weight of the unit was critical. Potential farming situations could involve exposure to temperature extremes, rain and dust, as well as physical abuse due to animals rolling over the unit, or rubbing it against trees or fence posts. For these reasons the FRPM had to be sealed and of robust construction.

3.2 Enclosure

The FRPM is currently housed in a rugged aluminum case to provide adequate protection from the animal and its environment. The case is constructed from a length of 2mm thick tubular aluminum extrusion with 6mm thick end-caps and is sealed against dust and water by O-ring seals in the end-caps providing IP67 protection with submergence of up to 15cm. Waterproof connectors and switches complete the sealed assembly.

Completely assembled the unit measures approximately 260mm by 120mm by 30mm and weighs approximately 980g, a weight easily supported by most farmed animals. Of this total weight around 650g is the enclosure alone, so switching to a plastic case would give a considerable weight saving.

The unit is attached to the animal by Velcro® and a single girth strap. The animal is first clipped and the Velcro is then glued to the clipped wool. Over 3-4 weeks the Velcro naturally detaches from the animal as the wool grows.

3.2.1 Enclosure Specifications

Table 3.1 presents the physical specifications of the FRPM unit in its current configuration.

Size	260mm by 120mm by 30mm
Weight	980g total assembled weight (the case makes up approximately 650g of this total)
Case	2mm thick aluminum extrusion with 6mm thick end caps. O-ring sealed case with sealed connectors gives IP67 protection with submergence of up to 15cm
Connectors	Lemo® EGG.OK.307.CLL[i] 7-pin combined power and RS232 connector DB25 sensor interface connector DB9 EEG/ECG electrode connector Huber+Suhner® SMA-50-2-46C/133[ii] aerial connector
Other Features	Recessed "wakeup" pushbutton Status LED

Table 3.1 – FRPM Physical Specifications

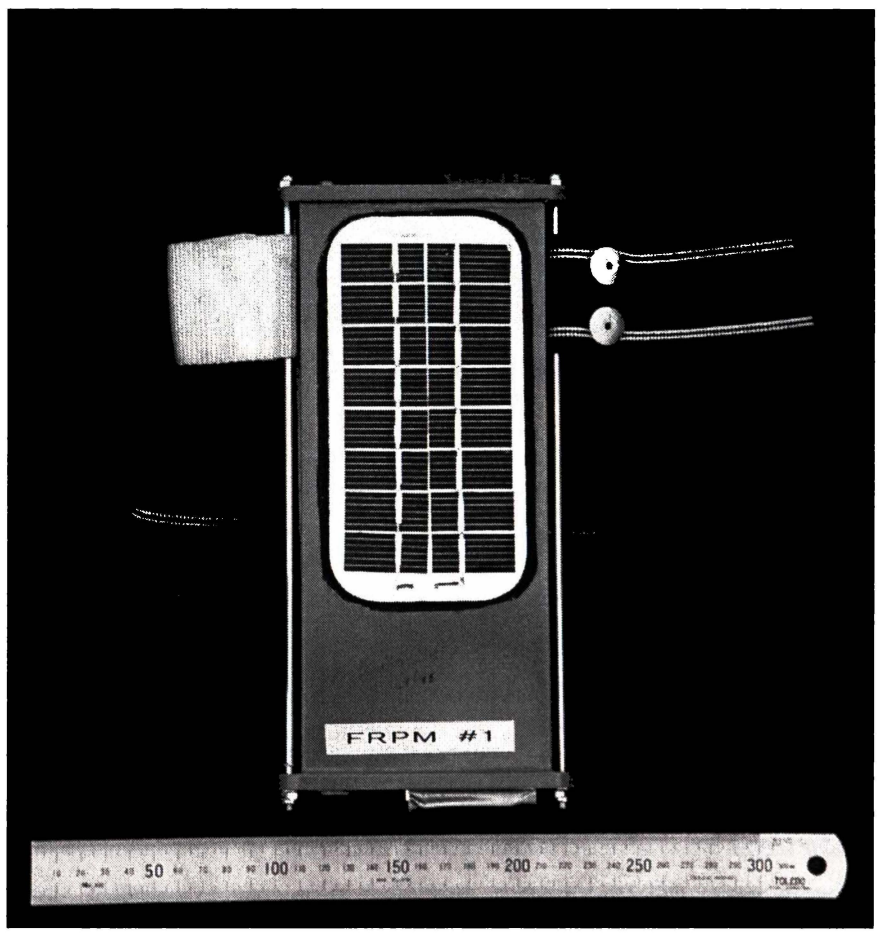


Figure 3.2 – The Free-Range Physiological Monitor

[i] Lemo S.A., Chemin des Champs-Courbes 28, CH-1024, Ecublens, Switzerland
[ii] Huber+Suhner AG, 8330 Pfäffikon/ZH, Switzerland

3.3 The FRPM Main Board

The requirement that the FRPM be able to process raw data on the subject to produce only summary results of recordings for the researcher, and that the FRPM be able to execute relatively complex signal processing algorithms, required the use of both a powerful microprocessor and a large amount of memory in the unit. Figure 3.3 shows a block diagram of the Main Board of the FRPM unit. A more detailed discussion of the FRPM hardware design can be found in Appendix 2.

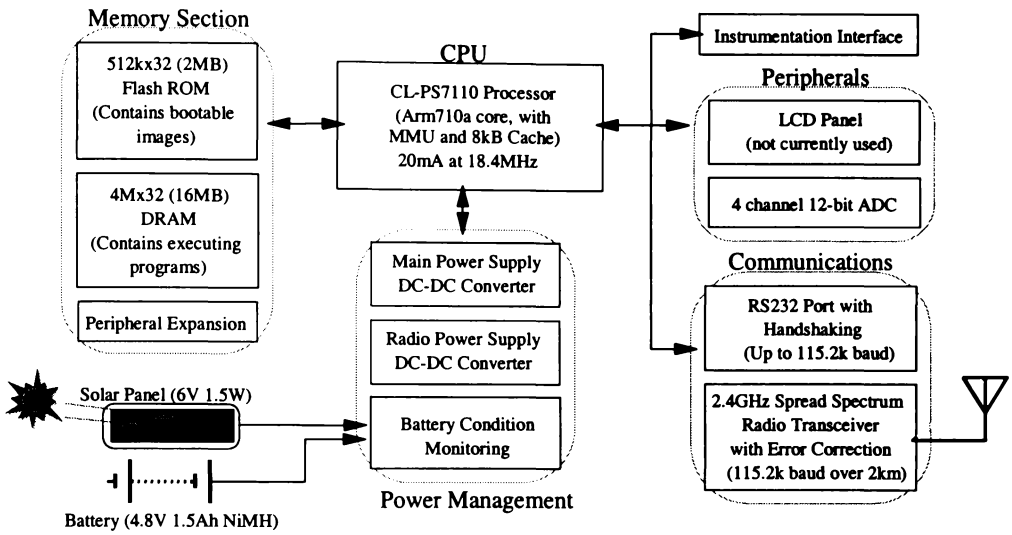


Figure 3.3 – FRPM Main Board Block Diagram

The use of a Digital Signal Processor (DSP), although excellent for complex data analysis, was excluded because the available DSP processors had minimal power management hardware, poor support for on-board peripherals and generally little input/output capability. To obtain both the desired low power consumption and high performance, together with power management and a system with minimal external peripherals, the CL-PS7110[i] processor was chosen.

This processor is based around an ARM710a[ii] core, a 32-bit RISC CPU with integrated cache and memory management. This core gives 15 Vax™-MIPS of performance at 18.432MHz, typically matching the performance of a 33MHz Intel® i486, while consuming only 60mW of battery power. During standby the processor current draw is 3μA.

A 16MB, 32-bit wide, DRAM module was chosen for system memory. The type of module chosen supports "self-refresh" operation allowing the memory to maintain its contents even if the processor is inactive while drawing only 3mA. All program instructions are fetched from this DRAM module although a 2MB, 32-bit wide, Flash ROM memory is used for the initial startup of the system software and as a non-volatile store for vital experiment data.

---

[a] Cirrus Logic Inc., 3100 West Warren Avenue, Fremont, CA 94538, USA
[ii] Advanced RISC Machines Ltd., 90 Fulbourn Road, Cherry Hinton, Cambridge CB1 4JN, UK

Peripherals available on the FRPM include a real-time clock, two timer/counter channels, LCD display support (only used for unit diagnostics), serial communications, and a synchronous serial port connected to an 8 channel, 12-bit, analog-to-digital converter. Of the 36 general purpose input/output pins available, 19 are available for expansion purposes while the rest are occupied by FRPM system management.

The real-time clock and it associated alarm are used for the sequencing and synchronization of experiments both within a single unit and between multiple units each on a separate subject.

The four 12-bit ADC channels available on the basic unit can obtain an aggregate sample rate of 4kHz and are typically used for recording slow activity such as body temperatures, respiration rate, digestive activity (via microphone), and motion (via an accelerometer).

The serial communications port can operate at up to 115.2kbs^{-1} and is multiplexed between three possible communications channels: an RS232 interface with full handshaking, an IrDA (Infrared Data Association) interface and a spread-spectrum radio module.

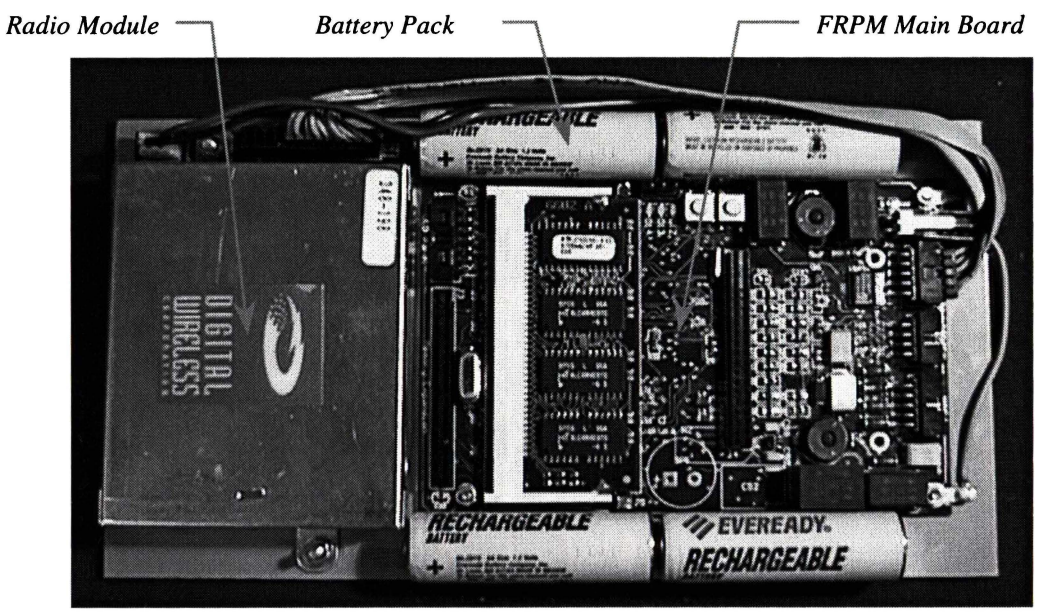


Figure 3.4 – The FRPM Main Assembly

The radio module is a 2.4GHz spread-spectrum design[i]. This module can support a network of up to 255 modules with a network data rate of up to 115.2kbs^{-1}. Up to 16 networks can be in operation simultaneously. The radio modules support full error correction and data buffering with transmission retry if a module temporarily drops from the network. The modules typically consume 180mW of power while linked to the network and up to 850mW during transmission and have proved in practice to operate at ranges of up to 2km even around obstacles such as low hills and buildings. Typically when waiting between experiments the FRPM uses the radio in 10

[i] Digital Wireless Corporation, 1 Meca Way, Norcross, GA 30093, USA.

second bursts every few minutes to "check-in" to the base station in order to reduce the unit's average power consumption.

3.3.1 Processor Board Specifications

The FRPM processor board specifications are presented in Table 3.2.

Size	175mm by 95mm by 18mm
Weight	52g for assembled printed circuit board 108g for battery pack 125g for radio module 285g total assembled weight
Construction	6 layer printed circuit board using surface mount components
Power Consumption	6.25Wh NiMH battery pack with 1.5W solar panel Peak power consumption is 1W Idle power consumption is 200mW Suspend power consumption is 15mW With periodic radio check-in FRPM will typically run for up to 7 days without solar charging before batteries are exhausted
Processor	Cirrus Logic CL-PS7110 clocked at 18.432MHz giving 15 Vax™-MIPs performance
Memory	16MB 60ns DRAM with 2MB Flash Memory
Serial communications	Multiplexed between RS232, radio module and infrared transmission (IrDA) at rates of up to 115.2kbs^{-1}
Radio module	Digital Wireless Corporation WIT2400M 2.4GHz spread-spectrum modem giving a range of up to 2km with full error-correction and retry capability. Radio is addressable allowing up to 255 FRPM units to operate simultaneously
Expansion capabilities	Memory expansion is 4kB byte-wide interface with system control designed for direct connection to Analog Devices ADSP2181 Digital Signal Processor Interface expansion is 16-bit digital input/output with 8 key keypad interface, system control, and four 12-bit analog inputs
Counter/Timers	Real-time clock with alarm Two 16-bit general purpose counter/timers clocked at 512kHz or 2kHz
Display	External 1024x768 by 16 colors LCD panel (typically used only for diagnostics purposes as its use requires the FRPM case to be opened)

Table 3.2 – FRPM Processor Board Specifications

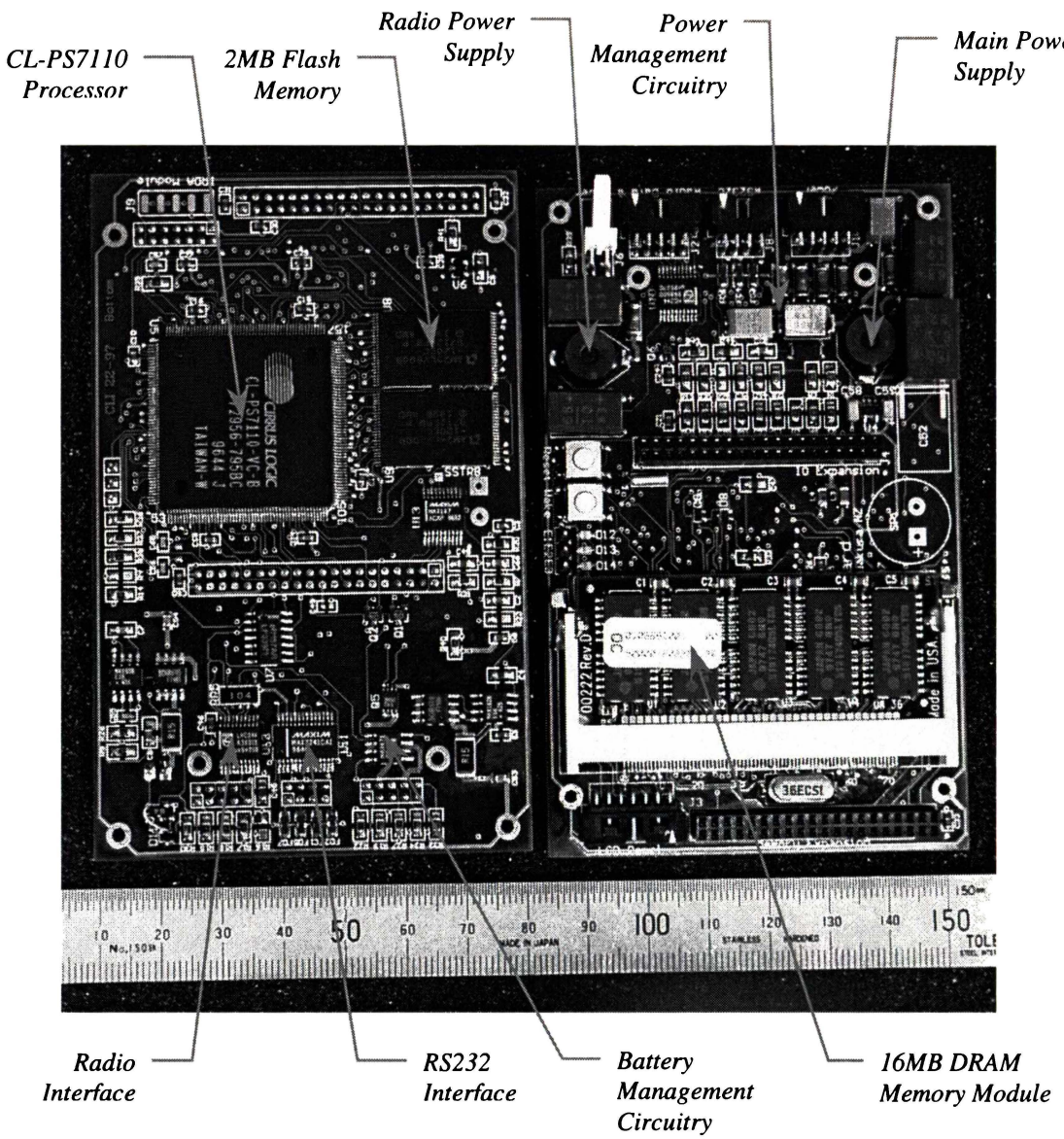


Figure 3.5 – The Free-Range Physiological Monitor Main Board

3.4 FRPM Power Management

Because of the long periods for which the FRPM is required to be attached to a subject, power management was a critical part of the instrument design. The FRPM uses a solar array as its main power source. Measurements have shown that on an average day in August (mid-winter in New Zealand) around 2.5Wh of energy can be collected from the array. The typical FRPM power consumption in any 24 hour period is 1.5Wh, leaving ample energy for experiments. The excess energy is used to charge a 6.25Wh Nickel-Metal Hydride battery pack that provides power for overnight operation and also helps support the unit against days where light levels are too low to allow the solar array to provide a power surplus. The battery alone can power the unit for at least 7 days.

Figure 3.6 shows an energy usage profile for a five day period beginning Monday 11th August 1997. For this test the FRPM was operated on a profile of 30s of radio contact every 5 minutes. Good charging of the FRPM batteries can be seen on four of the five days. Tuesday resulted in little charging due to a heavy overcast. However, sufficient charge was in the batteries to enable the FRPM to continue operation. Overall, energy storage from the solar panel well exceeds energy usage by the FRPM.

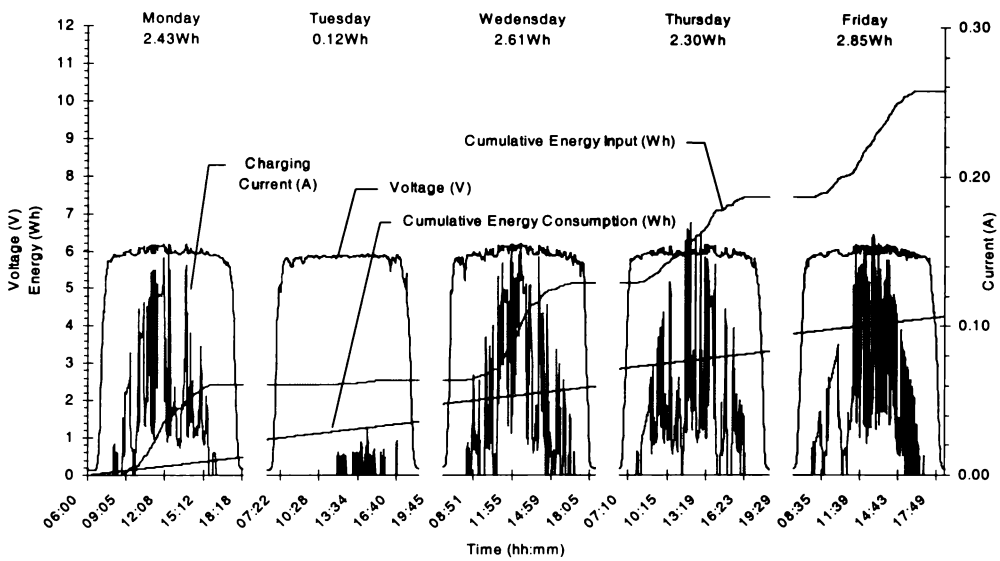


Figure 3.6 – Typical FRPM Energy Storage/Consumption Profile

Full power monitoring and battery charging control is implemented in the operating system firmware. This monitoring allows estimation of the remaining battery life as well as ensuring optimum charging of the battery pack. The battery monitor can also force the FRPM into a "safe-mode" to wait for a satisfactory battery charging period. In this safe-mode no experimental activity is allowed and radio activity is heavily reduced in order to maintain memory contents.

3.5 FRPM System Software

The FRPM System Software forms the second major component of the unit after the hardware itself. The System Software is based on an operating system/application program architecture: application programs are run from a command line interface, on demand, to perform some task. The operating system provides the application with access to resources such as a filing system, robust communications, power management, event handling, etc. while the application performs the functions specific to the task at hand. Applications that were developed as a part of this project include the Evoked Response acquisition and signal processing application, a general purpose waveform capture application, a general data logging application, and an application for file compression.

A more detailed discussion of the FRPM software design can be found in Appendix 3, while detailed descriptions of the applications developed for the FRPM can be found in the FRPM User Manual[87].

As an alternative to the command line interface, the System Software provides a scheduler based on the system real-time clock that enables applications to be run at particular times on either a fixed time or a periodic basis. Using this feature the researcher can automate and synchronize data collection from multiple FRPM units simultaneously.

Figure 3.7 gives an architectural overview of the system software.

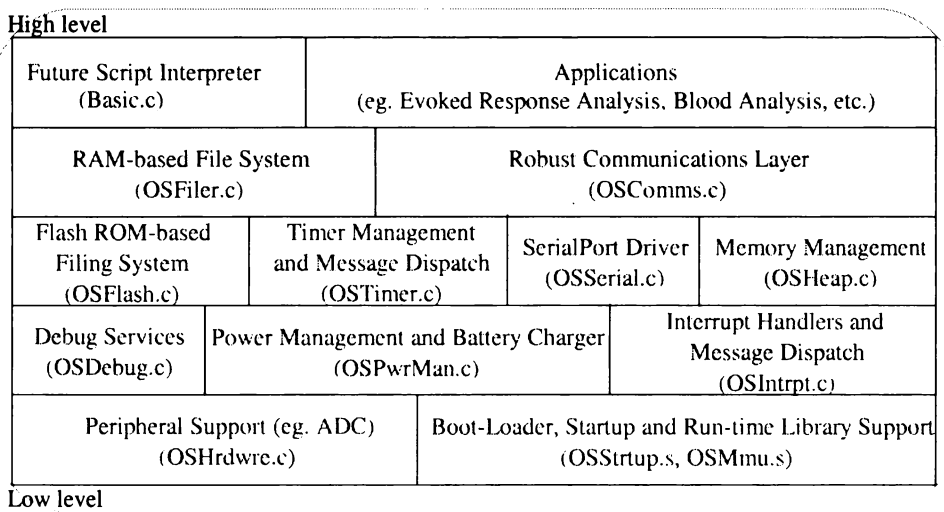


Figure 3.7 – FRPM System Software Block Diagram

The System Software is a full 32-bit system and takes advantage of the hardware memory protection and exception traps provided by the ARM710a processor to terminate any application that attempts an illegal memory access or attempts to execute an illegal instruction. Any resources that were allocated to an application that was terminated are always recovered, ensuring reliability by eliminating resource and memory leaks.

The System Software occupies approximately 100kB of memory and is copied from relatively slow Flash memory to the faster DRAM memory from where it is executed during the boot process.

3.5.1 System Software Specifications

The FRPM System Software specifications are presented in Table 3.3.

Size	Currently uses 106kB of the 1MB available for System Software
System Protection	Illegal memory access Illegal instruction execution Illegal operating system call "Application not responding" protection
File System	10MB RAM-based file system 1792kB Flash-based file system
Communications	RS232 communications support Radio modem communications support IrDA modem support
Timer Support	Two programmable-rate 16-bit timers Applications can register as timer overflow event handlers
Real-time Clock Support	Real-time clock with alarm Applications can register as real-time clock alarm event handlers
Debug Support	Separate debugging communications channel Status LED support
Interrupt Support	Sixteen potential interrupt sources Applications can register as interrupt event handlers
Display Support	Supports a single 320x200 pixel LCD display with 16 gray levels Graphics primitives include pixels, lines, rectangles, fills, and text
Application Support	Maximum useable application space is 4MB Command line parameter support Automatic resource and memory cleanup on application exit
Other Features	Command line interface for system management Real-time clock based sequencer for automated data collection 1MB of RAM reserved for future addition of a script interpreter
Power Management	Supports both Suspend and Idle modes Applications can register as power management event handlers

Table 3.3 – FRPM System Software Specifications

3.6 Physiological Interface

The FRPM records physiological signals via a small plug-in interface board containing the necessary signal conditioning circuitry for several different signal sources. In its current form the interface board provides support for the acquisition of EEG/Evoked Response/ECG signals, physical activity, respiration rate, temperature measurement and other anxiety indicators. Figure 3.8 shows a block diagram of the physiological interface board.

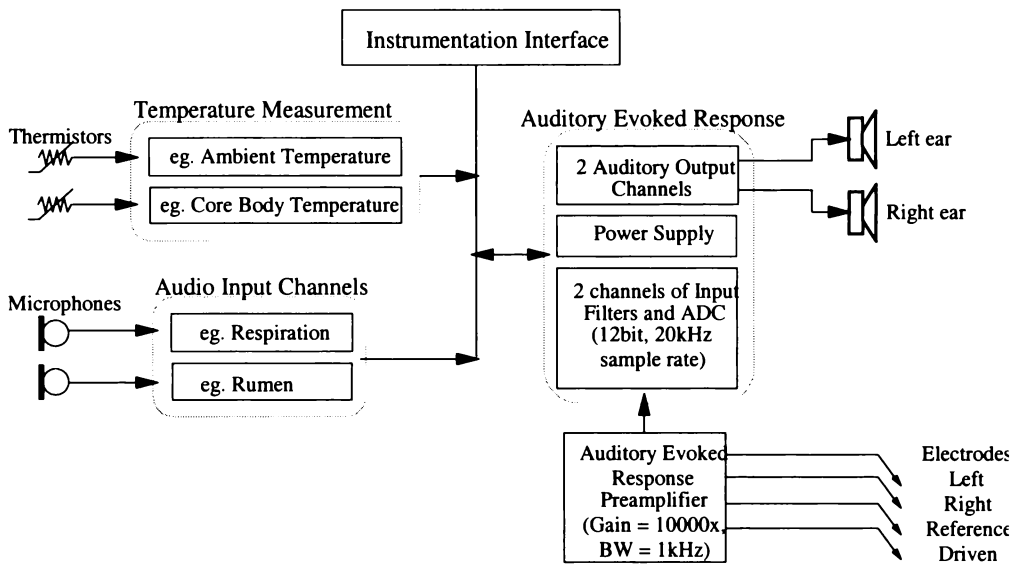


Figure 3.8 – FRPM Physiological Interface Block Diagram

The interface board contains a separate 12-bit ADC to allow two channels of either EEG/Evoked Response or ECG to be recorded at up to 10kHz sample rate per channel. Two auditory click output channels with variable attenuation (0 to 100% over 256 steps) for Evoked Response acquisition are also present. The sound pressure level produced by these outputs depends on the choice of loudspeaker: both piezoelectric and electromechanical speakers can be used.

The two available temperature measurement channels can use either precision thermistors or digital temperature sensors. The calibration of the thermistors generally gives a reading accurate to within 0.1°C while the digital sensors are typically accurate to 0.03°C.

Two variable gain (-22dB to +26dB over 256 steps) inputs provide support for recording respiration (using a thermistor or chest expansion), physical activity (with an accelerometer) and digestive activity and teeth grinding (using a microphone).

Input and output protection from over-voltage or electrostatic events such as contact with an electric fence is provided by a combination of semiconductor transient suppressors and current limiting resistors. The interface board is constructed on a separate plug-in sub-assembly to allow maximum flexibility for different uses of the FRPM and to aid in any interface upgrades in the future since on its own it is a relatively inexpensive and easily produced part of the overall system.

3.6.1 Physiological Interface Specifications

The physiological interface board specifications are presented in Table 3.4.

Size	60mm by 44mm by 11mm
Weight	16.3g for assembled printed circuit board
Construction	2 layer printed circuit board using surface mount components for minimum size and weight connectors are the only through-hole components.
Power Consumption	Maximum power consumption is 50mW
Protection	Input lines are protected with a combination of transient suppressors and current limiting resistors Output lines are protected with transient suppressors
Evoked Response Interface	Dual multiplexed recording channels with a fixed gain of 2x and bandwidth from DC to 1.6kHz. Dual stimulus generator provides clicks of programmable amplitude and polarity. Click amplitude is programmable from 0 to 5V over 256 steps. Dual-rail power supply provides +/-5V at up to 100mA.
Audio Inputs	2 Audio input channels (GND, +3.3V, Audio, Shield) with programmable gain from -22dB to +26dB over 256 steps. Bandwidth is 0.4Hz to 2.3kHz. These input channels can be configured for use with accelerometers, microphones, or thermistors.
Temperature Measurement Inputs	Dual temperature measurement channels (GND, Bias, Sense, Shield) for use with either thermistors or digital temperature sensors. A dual 2.6V reference supply is provided for use with thermistors for maximum accuracy. Typical accuracy is 0.1°C with thermistors but is dependent on sensor assembly. Typical accuracy is 0.03°C with DS1820[i] digital temperature sensor part.
Miscellaneous	Status LED connection Wakeup key connection ±5V supply connection +3.3V supply connection

Table 3.4 – FRPM Physiological Interface Board Specifications

[i] DS1820 1-wire™ Digital Thermometer, Dallas Semiconductor Corp., Dallas, Texas 75244-3292, USA.

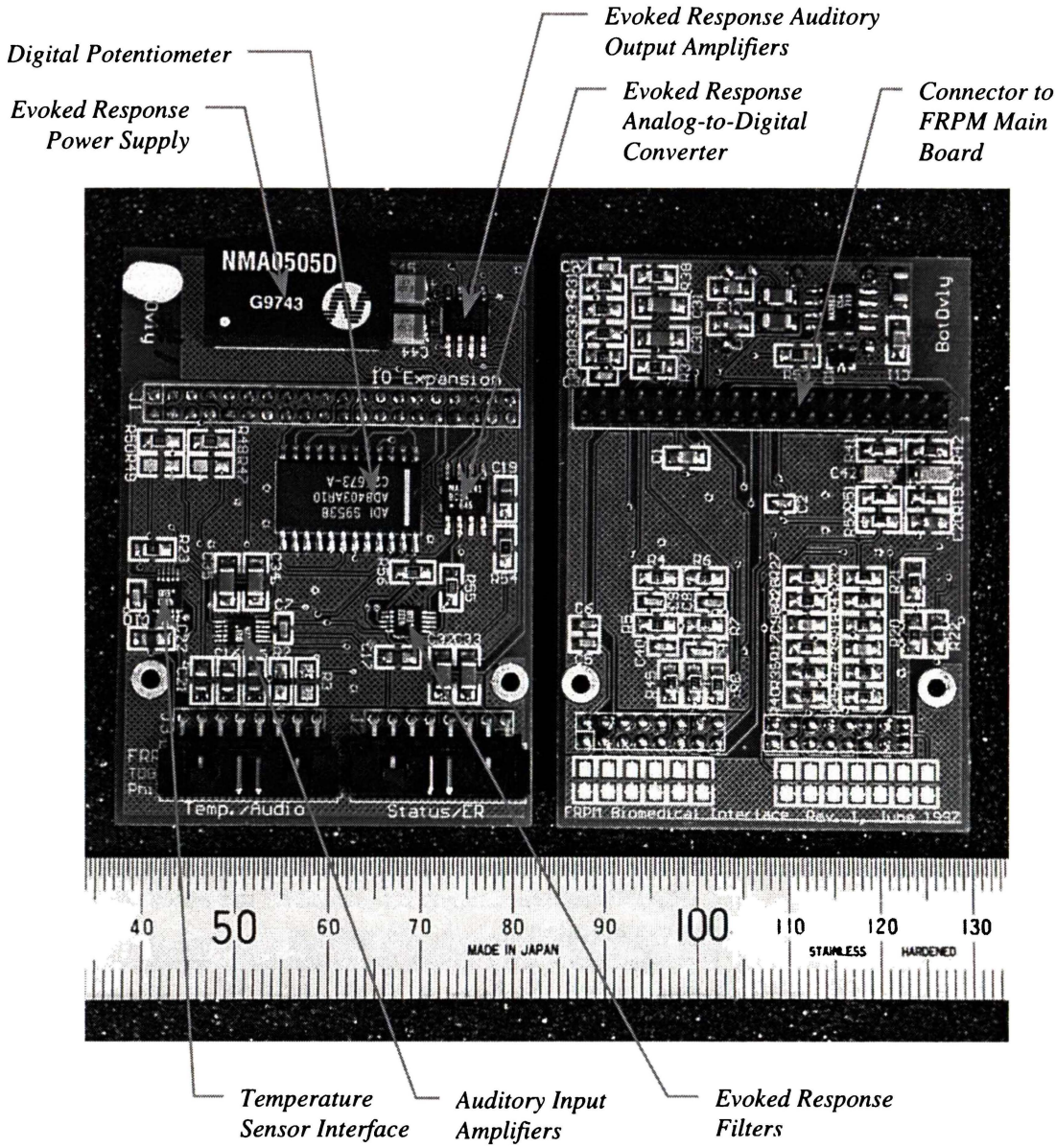


Figure 3.9 – The Physiological Interface Board

3.7 Attachment of FRPM to Animal

Long-term attachment of the FRPM unit to the subject animal posed one of the more interesting challenges. A goal for use of the FRPM was for it to remain attached to the animal for at least four weeks.

The case of the FRPM unit includes two stainless steel rods running the length of the unit on each side. These rods were intended to be used to fasten the FRPM to a harness on the animal. The original harness design was a two-piece design consisting of a wide elasticized girth strap with adjustable clips at each end together with a non-elasticized Y-shaped strap that was attached to the girth strap. This second strap passed from under the animal, up and over its shoulders, with adjustable clips connecting it to the FRPM. It included a shaped cup that was intended to locate over the sternum of the animal. The FRPM itself had an I-shaped piece of strapping with four clips - the two at the back connected to the girth strap while the two at the front connected to the shoulder strap. Figure 3.10 shows the harness attached to a 40kg sheep.

Figure 3.10 – FRPM Harness Assembly on Animal Subject

As Figure 3.10 shows, the harness as designed was not applicable to animals across a wide range of weights. With this design, it would have been necessary to construct different harnesses for each weight range. The figure also shows two major problems with the overall harness: the harness (and hence FRPM) tends to ride forward on the animal interfering with neck movement, and the harness tends to roll off the animal's back over a period of time and hang down the side of the animal. This loss of a square sitting position could be uncomfortable to the animal.

Attachment of the FRPM to the animal in the field using the harness was also found to be difficult as it required two steps. The first required four clips to be fastened before the FRPM was secured to the animal. This was followed by the adjustments of four straps to ensure the harness fitted the

animal being used. This operation frequently required three people: one on each side to adjust the straps plus one restraining the animal.

Any alternative attachment method needed to be suitable for animals of varied size and to hold the FRPM centered on the back of the animal, preventing slippage. It was also desirable to have a method of attachment workable by a single person. The attachment method should also ensure that the weight of the FRPM be spread evenly across a large area.

To achieve these aims a large Velcro®-type loop patch was glued to the back of the animal and the FRPM fastened to this patch using a corresponding hook patch of Velcro on the bottom of the FRPM. The Velcro used was 100mm wide and at 260mm was slightly longer than the FRPM. To attach the Velcro to the animal, an area slightly larger than the Velcro was clipped short on the animal and a line drawn down the animal's backbone with permanent marker while the animal was standing. The clipped area was then inspected for any foreign debris before the 'loop' part was glued to the animal with a thin layer of cyanoacrylate glue. The drawn line was used to ensure the Velcro was centered on the animal. The Velcro was placed so its front edge was just behind the shoulders of the animal and the corners of the Velcro were rounded off to minimize the chance of skin irritation by abrasion. Any edges of the Velcro not completely glued down are touched-up with extra glue to minimize any strain on the skin-Velcro interface which could cause irritation.

As the curing of cyanoacrylate glue is driven by the presence of water, and is an exothermic reaction, it is best to ensure the animal is dry before using this glue over a large area or discomfort to the animal could result due to heat generation. Cyanoacrylate glue appears to cause no skin irritation when used on animals.

A high-density foam packer in the general shape of the sheep's back was glued to the FRPM itself with the 'hook' part of the Velcro glued to the packer. The packer ensures that the weight of the FRPM is spread evenly over the attachment surface.

To attach the FRPM to the animal, the FRPM is aligned with the line previously drawn on the animal and moderate pressure is applied to cause the Velcro to make contact. A single elasticized girth strap with a single adjustable clip is usually fastened around the animal to ensure the FRPM does not completely loose contact with the animal. This attachment procedure is easily accomplished by a single person while restraining the animal. An animal with the FRPM attached in this way is shown in Figure 3.11 (the sheep shown is the same animal as in Figure 3.10 but three months later).

Figure 3.11 – Sheep with FRPM Attached by Velcro

This attachment method has been used to attach FRPM units to sheep for periods of up to four weeks with indication that the period could have been much longer. Since the Velcro on the sheep is glued to the wool rather than to the skin, at the end of any trial this Velcro is left on the animal awaiting sufficient wool growth to enable the Velcro to the removed with clippers.

In practice this attachment method appears to cause no discomfort or irritation to the animal. After an initial settling period of less than 10 minutes the animal with the FRPM appears to be completely comfortable with the FRPM. Surrounding animals appear to quickly get used to the sight of FRPM units attached to their fellow sheep. After several periods of using the FRPM on an animal it quickly gets used to the attachment process and the extra weight of the FRPM and little or no settling period is observed.

Because the FRPM is not attached to a rigid surface, rapid movement of the animal can cause detachment and reattachment of the Velcro causing the characteristic 'ripping' noise of Velcro. The sheep seem to get used to this noise also and it causes little disturbance to the flock. Typical sheep movement such as eating, walking and even moderate running are usually not sufficient to cause any extra noise. The girth strap ensures the FRPM cannot become completely detached from the animal during extreme movement.

3.8 Performance of the FRPM

To date the FRPM has been successfully used in the field on up to 4 subject animals simultaneously for periods of up to three weeks. Signals recorded from the subjects have included Auditory Evoked Response, ECG and body temperature.

Figure 3.12 shows an example AER waveform recorded by the FRPM from a confined sheep. Clicks were produced by a speaker in the animal's right ear at a 6.7Hz mean rate at an intensity of 95dB at one inch. The recorded EEG was sampled at 5kHz with stimulus responses being averaged 2048 times to produce the final response. Brainstem activity can be seen before 10ms, with primary cortex activity around 15ms and secondary cortex activity around 40ms[82]. The noise limits are 99% confidence limits based on noise measurement during the recording period (see Chapter 5).

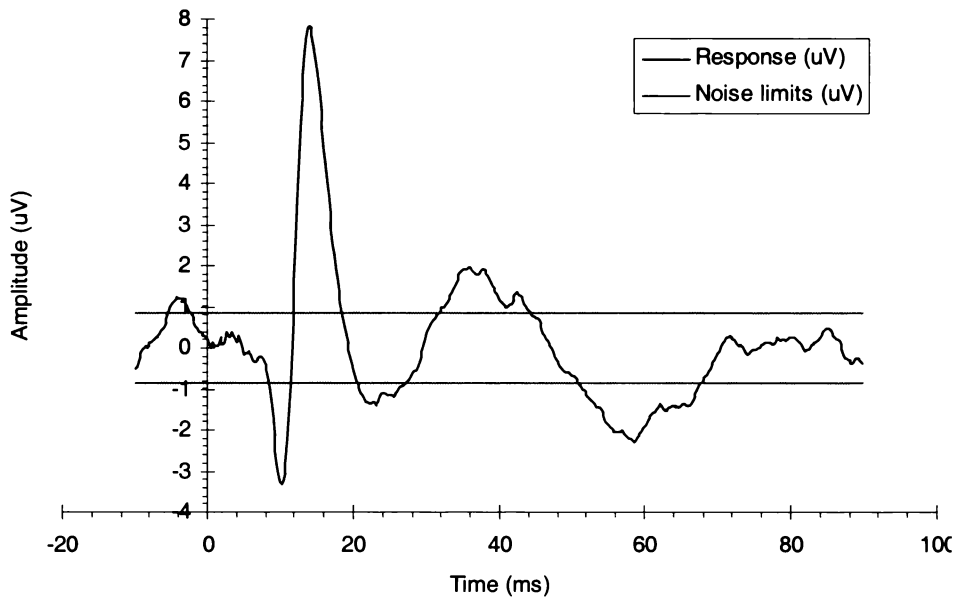


Figure 3.12 – Example Sheep Auditory Evoked Response

During Evoked Response acquisition at a 5kHz sample rate, the FRPM is observed to be 90.9% utilized. This figure is high due to a 32kHz hardware-based timer being used to provide a timebase for all acquisition timing. This timebase is in turn used by a software state machine which generates the sampling and stimulus clocks by software emulation, resulting in high processor usage.

Figure 3.13 shows a portion of an example sheep ECG waveform recorded by the FRPM. The sampling rate was 500Hz and the amplifier used for the recording was the same as that used for the Evoked Response waveform in Figure 3.12. This amplifier was not designed for ECG and has

a low frequency cut-off at 2Hz - too high for clinical ECG analysis as it distorts the waveform, but quite suitable for monitoring heart rate.

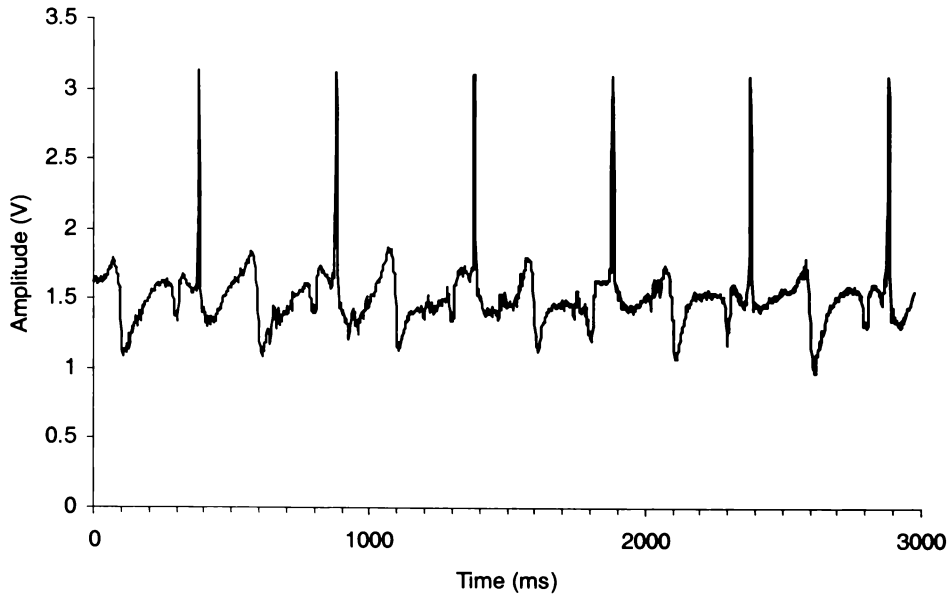


Figure 3.13 – Example Sheep Electrocardiogram

During ECG acquisition at a 500Hz sample rate, the FRPM is observed to be 19% utilized, leaving considerable time for other processing. However, this acquisition uses a similar software-based timing scheme to that of the Evoked Response acquisition. By using the CL-PS7110 hardware-based Synchronous Serial Port (SSP) to transfer samples from the ADC to the CL-PS7110 the processor utilization is reduced to 10%, but the SSP can only be used with the ADC on the FRPM Main Board and is not applicable to the Interface Board.

Clearly, for high-speed sampling additional hardware resources would be required, probably in the form of an FPGA connected to the Memory Expansion Interface to generate the acquisition timing and to move samples into the processor or directly into main memory. However, for the current applications of the FRPM the current hardware and software provides ample processing power.

3.9 Future Improvements

There are several areas in which the FRPM could be improved. These fall into three main categories: size and weight, data storage, and computing performance.

3.9.1 Size and Weight Reduction

Currently the FRPM dimensions are 260mm by 120mm by 30mm and its assembled weight is 980g. Most of the size comes from the linear internal layout of the Main Board, the Radio Module, and the EEG Preamplifier. A long but low profile unit was chosen instead of these components being stacked to provide the most compact unit. Most of the weight (approximately 650g) is the aluminium case alone, while the Radio Module (125g) and battery (108g) also contribute in large proportion.

By changing the case construction from aluminium to molded reinforced plastic construction the robustness of the unit could be retained at a much lower total weight. This construction has the initial disadvantage of a lack of electromagnetic shielding from external interference but this could be eased by the addition of carbon particles to the plastic stock before molding, or by spraying a conductive shielding film on the molding after manufacture. It would be reasonable to expect a plastic molded case to weigh less than 200g (glass-reinforced Nylon 66), resulting in a weight reduction of approximately 46%.

A molded case would also allow a better internal arrangement of components and would greatly ease manufacturing and sealing the unit against dust and water. The result would be a smaller, more compact unit.

Another area for weight loss is the Radio Module. Based on in-field use there appear to be two distinct applications for the FRPM. The long-term monitoring application has a definite requirement for the radio link but for short-term applications that may only run for several hours the radio link is often not required. Therefore it may be advantageous for some weight-critical applications to remove the Radio Module, saving another 13%. Regardless of this, the current momentum of the communications industry should see smaller, lighter, and lower power consumption radio modules become available in the near future.

A final reduction in weight could be obtained by changing the battery from NiMH technology to Polymer-based Lithium battery. Currently the battery pack used in the FRPM achieves an energy density of 62Wh/kg. By switching to a Lithium battery this energy density could be improved to 125Wh/kg[i] allowing the use of a battery pack 50% lighter than that currently used.

The other major path toward smaller size and less weight is a higher level of circuit integration, particularly in the FRPM Main Board. Based on current technology, it should soon be possible to

[i] Ultralife Batteries, Inc., 1350 Route 88 South, Newark, NY 14513, USA

at least halve the size of this board, mainly by integrating the processor, DRAM memory and Flash memory into a single integrated circuit.

Taken together these three areas of improvement could result in an FRPM of dimensions as little as 120mm by 60mm by 15mm and weighing 480g with the Radio Module or 355g without, making the FRPM suitable for a much larger range of applications and animal species.

3.9.2 Data Storage

Although the FRPM currently has 10MB of DRAM memory and 1792kB of Flash memory for the storage of experimental data and applications, this memory could be quickly filled during long-term monitoring applications. Of the available memory on the FRPM, only the Flash memory can be guaranteed to survive a loss of power in the FRPM due to a lack of sufficient sunlight for charging the battery or an internal failure. In the future it may prove desirable to add additional storage to the FRPM that would provide large capacity and protection against power failure. Two potential technologies exist for this additional storage: Flash PC Card storage or a miniature Hard Drive.

The Flash PC Card media are currently available in capacities of up to 112MB[i] in the standard PC Card Type-II format, or up to 64MB[i] in the CompactFlash[ii] format commonly used in digital cameras. These cards are also removable, allowing them to be taken from the FRPM and plugged directly into a researcher's computer, dramatically speeding the 'download' of data from the FRPM to the computer when compared to data transmission over the serial link ($20MBs^{-1}$ versus $11kBs^{-1}$).

Miniature Hard Drives are also becoming available as a result of the industry drive towards 'palmtop' portable computing. IBM[iii] has recently released a 340MB hard drive that weighs less than 17g and measures only 43mm by 36mm by 5mm. The use of such a storage media would greatly increase the storage capacity of the FRPM, and be protected against power failure. It is expected that these drives will also become available in the removable CompactFlash Type-II format.

3.9.3 Computing Performance

Any re-design of the FRPM to incorporate improvements would also include an investigation into upgrading the CL-PS7110 processor used in the FRPM for better computing performance. The ARM7 core used in the CL-PS7110 is now at least three generations behind the leading edge of portable computing and several processors are now available that could result in dramatic improvements to the FRPM. Although other processor families are emerging with features

[i] SMART Modular Technologies, Inc., 4305 Cushing Parkway, Fremont, CA 94538, USA
[ii] CompactFlash Association, PO Box 51537, Palo Alto, CA94303, USA
[iii] IBM Storage Systems Division, International Business Machines Corporation, 5600 Cottle Road, San Jose, CA 95193, USA

suitable for the FRPM (notably the Hitachi SH series[i]), the software developed for the FRPM and the experience gained by using an ARM processor encourage the use of another ARM processor for any upgrade.

The StrongARM[ii] processor core was originally developed by ARM and Digital Semiconductor but is now owned by Intel Corporation. A variant on the ARM architecture and code-compatible with the CL-PS7110, this core has one of the highest performance/power consumption figures currently available. For example, the CL-PS7110 achieves 15MIPs at 18MHz and 66mW power consumption, giving a performance of 227MIPs/W. The SA-1100 achieves 250MIPs at 220MHz and 550mW power consumption, giving a performance of 454MIPs/W - twice that of the CL-PS7110. With a peak speed of 16 times that of the CL-PS7110 for only half the power consumption per instruction, the SA-1100 would dramatically improve the computing performance of the FRPM. The SA-1100 also includes a wider variety of integrated peripherals including multiple serial ports and PC Card support, and is produced in the Mini-Ball Grid Array (mBGA), occupying only 28% of the area of a CL-PS7110 processor, allowing for a smaller FRPM.

3.10 Discussion

Initial development of the FRPM was carried out to acquire the signals, and perform the computation processes, needed to perform Auditory Evoked Response acquisition from ambulatory animal subjects. The ability to analyze and summarize recorded data on the subject provides significant benefits for experiments involving several ambulatory or unconstrained subjects and long periods of time. It is a natural extension of the capabilities of the FRPM to add measurement of other critical physiological parameters, some of which require data compression, such as the feature extraction from ECG or sleep state analysis from EEG.

Although this monitor was originally developed for animal applications it has wide-ranging use in all medical monitoring fields and with future enhancements could lead towards a miniaturized, fully-integrated biomedical instrumentation system. In terms of human application, the FRPM could enable activities such as recording sleep patterns in subjects while at home or research into such areas as the ambulatory monitoring of the onset of epileptic seizures.

[i] Hitachi Semiconductor (America) Inc., 2000 Sierra Point Pkwy, Brisbane, CA 94005-1897, USA
[ii] Intel Corporation, 2200 Mission College Blvd., Santa Clara, CA 95052-8119, USA

Chapter 4 The Design of the EEG Preamplifier

In recording the electroencephalogram (EEG) or the Evoked Response, the design of the preamplifier is critical. The preamplifier must be capable of amplifying a signal with an amplitude of about 50μV[88] by a factor of around 1000, while coping with DC offsets of up to 500mV due to polarized electrodes[89] and high levels (up to 100mV[90]) of interference.

Much of the early work for recording EEG used DC coupled galvanometers with optical magnification. The deflections of the galvanometer light beam were typically recorded on photographic film. In the 1930s the invention of the thermionic valve enabled high-gain AC coupled valve amplifiers to be used with the waveforms being recorded on photographic film via an oscilloscope. In 1938 the development of the differential amplifier[91,92] made it possible to eliminate from the recording much of the interference from external sources. The 1940s saw the development of pen writers with sufficient speed to provide a real-time record of electrical activity in the brain. With the invention of the transistor in 1948 and the development of chopper amplifiers in the 1950s a return to DC recording was made. Since then most instrument development work has focused on miniaturization through the use of integrated circuits and the design of multi-channel recorders.

A high-gain, low-noise preamplifier was designed and built to record the Electroencephalogram signal from which the Evoked Response waveform is reconstructed. The preamplifier was based on a design by Metting van Rijn[93] which was then optimized for our operating parameters. A SPICE circuit simulation package was used to aid in the design of the preamplifier, enabling the performance of the preamplifier to be predicted before the final design was constructed.

The Metting van Rijn design is based around a two op-amp instrumentation amplifier topology. This design was chosen because it offered good performance from a small number of components and had good potential for scaling to a multiple-channel design. Although the conventional three op-amp instrumentation amplifier topology uses less parts than the chosen design, it suffers from

being inherently limited to a single channel configuration due to the lack of an explicit reference channel that can be shared among multiple channels. It also makes difficult modification to effectively block DC potentials without severely affecting common-mode interference rejection due to the topology's requirement for a high gain, perfectly symmetrical, differential input amplifier stage for optimal CMRR performance. The two op-amp topology has a dedicated reference input that can be used by as many channels as necessary, and due to the single-sided high gain amplification in each channel, it becomes possible to introduce DC blocking along with a large signal gain in the first stage of the amplifier while maintaining high CMRR. However, this flexibility does come at the cost of added components and complexity.

4.1 Preamplifier Requirements

The required preamplifier characteristics were:

- High differential gain obtained within a single stage to maximize noise performance and minimize the preamplifier's size and power consumption
- A high common-mode rejection ratio to overcome the need for any form of mains-frequency notch filter and to help reduce susceptibility to artifacts
- A high level of DC signal rejection (necessary due to the high gain) obtained without the use of capacitors in the signal path which would otherwise degrade the common-mode rejection ratio and cause poor recovery from overload conditions
- High input impedance and low input bias current to reduce the effect of variable electrode impedance on common-mode rejection and electrode DC polarization
- A wide bandwidth at high gain necessary to accurately capture the early components of the Evoked Response waveform

Based on the signal characteristics of the Auditory Evoked Response (see Chapter 5), these requirements were formalized and are listed in Table 4.1.

Parameter	**Required Performance**
Differential Gain	1000
Bandwidth	Greater than 1Hz to 1kHz
Noise Spectral Density	Less than $30nV/\sqrt{Hz}$ Less than 1μV in bandwidth
Input Impedance	>10MΩ
Input Bias Current	Less than 200pA
AC Common Mode Rejection Ratio	Greater than 100dB
Input Offset Rejection	Greater than ±500mV

Table 4.1 – Required Evoked Response Preamplifier Specifications

In addition to these parameters the preamplifier had also to be unconditionally stable, i.e. the preamplifier must not oscillate for any combination of input voltages and frequencies. The preamplifier was also required to have a modest power consumption.

4.2 Preamplifier Design

Figure 4.1 outlines the general topology of the preamplifier design by Metting van Rijn. This topology has three main parts: the unity-gain reference amplifier, the high-gain main amplifier, and the feedback circuit. The feedback circuit is unusual since it uses integrators to provide a low overall gain for low-frequency signals while still allowing high gain in the pass-band. This allows the circuit to exhibit excellent DC rejection while avoiding high-pass filters in the input signal path which can severely effect CMRR performance.

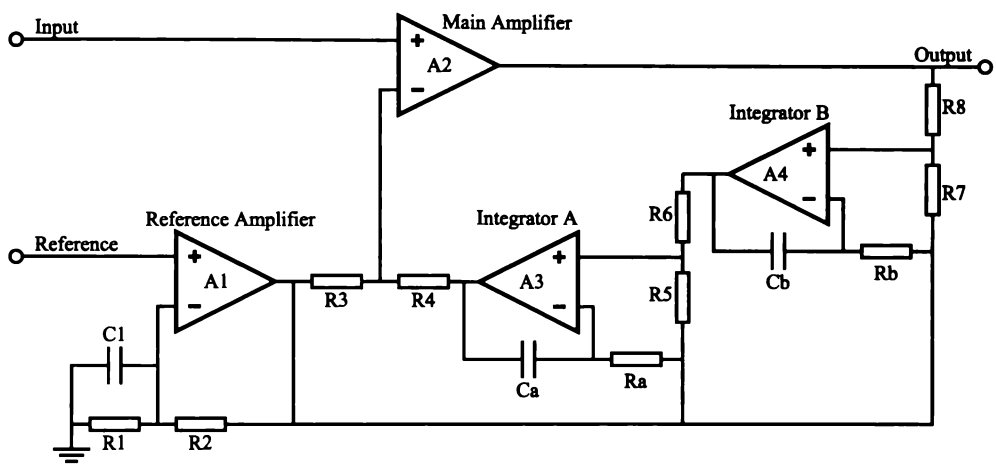


Figure 4.1 – Metting van Rijn Preamplifier Topology

The small-signal operation of this preamplifier can be described as follows: Assume that the reference input is at 0V and some small AC signal, V_{in}, is present at the recording input. If the preamplifier is operating within limits then this input signal must also be present at the inverting input of A2. The signal voltage at the output of A3 must then be:

$$V_{OUT}(A3) = \frac{R_3 + R_4}{R_3} V_{in} \tag{4.1}$$

The integrator based around A3 consisting of R_A and C_A has unity gain for frequencies above:

$$f_{HP} = \frac{1}{2\pi R_A C_A} \tag{4.2}$$

The signal voltage at the output of A4 must be:

$$V_{OUT}(A4) = V_{OUT}(A3)\frac{R_5 + R_6}{R_5} = \frac{R_3 + R_4}{R_3}\frac{R_5 + R_6}{R_5}V_{in} \tag{4.3}$$

Similarly, the output of A2 must be:

$$V_{OUT}(A2) = V_{OUT}(A4)\frac{R_7 + R_8}{R_7} = \frac{R_3 + R_4}{R_3}\frac{R_5 + R_6}{R_5}\frac{R_7 + R_8}{R_7}V_{in} \tag{4.4}$$

The differential gain A_{diff} of the preamplifier is thus:

$$A_{diff} = \frac{R_3 + R_4}{R_3}\frac{R_5 + R_6}{R_5}\frac{R_7 + R_8}{R_7} \tag{4.5}$$

The low-pass frequency limit of this design is determined by the gain bandwidth product (GBP) of A2 and the differential gain of the circuit. The low-pass corner frequency is:

$$f_{LP} = \frac{GBP(A2)}{A_{diff}} \tag{4.6}$$

For DC operation assume that the reference input is at 0V and some small DC offset is present at the recording input. If the preamplifier is operating within limits then this input offset must also be present at the inverting input of A2. The voltage at the output of A3 must then be the same as for the small signal case above. Because the integrator based around A3 has a very large gain at DC, the offset voltage at the output of A4 will be very small. Because A4 is also an integrator the output offset of A2 will be virtually zero, i.e. The DC offset at the input is rejected.

It can be shown that if A3 and A4 have open-loop gains of A_{OL}, then for an input offset voltage V_{dc} the output offset voltage will be:

$$A_{dc} = \frac{R_3 + R_4}{R_3}\frac{R_5 + R_6}{R_5}\frac{R_7 + R_8}{R_7}\frac{1}{A_{OL}^{\ 2}} = \frac{A_{diff}}{A_{OL}^{\ 2}} \tag{4.7}$$

The DC gain will be less than unity provided that:

$$A_{OL} > \sqrt{A_{diff}} \tag{4.8}$$

The integrator based around A3 performs the function of blocking any DC offset from the signal source while A4 blocks DC offset that could otherwise be present due to the input offset voltages of A2 and A3 combined with the high loop gain A_{diff}. Because of the DC blocking action of the two integrators, it is not necessary to place any AC coupling capacitors in the high-impedance signal paths to the non-inverting inputs of A1 and A2. This is advantageous since the poor tolerance of most capacitors would severely degrade the Common-Mode Rejection Ratio.

The Common-Mode Rejection Ratio (CMRR) is a measure of how well a differential amplifier can reject a pass-band signal that is common to both the recording and reference inputs. This can

also be considered as a measure of how susceptible the amplifier is to converting a common-mode signal into a differential signal. The CMRR is defined as:

$$CMRR = 20\log\left(\frac{A_{diff}V_{cm}}{V_{out}}\right) \tag{4.9}$$

where A_{diff} is the differential gain of the amplifier, V_{cm} is the amplitude of the common-mode signal used for the test, and V_{out} is the amplitude of the amplified common-mode signal at the output of the preamplifier.

Assume a signal V_{cm} is present at both the recording and reference inputs. For perfect CMRR rejection performance V_{out} should be zero. If A1 and A2 can be considered as perfect op-amps then for a V_{out} of zero the non-inverting and inverting inputs of A2 must be equal. It can be shown[93] that perfect CMRR can only be achieved if:

$$\frac{R_1}{R_2} = A_{diff} - 1 \tag{4.10}$$

In practice component tolerances prevent perfect CMRR from being achieved; however, a CMRR of around 120dB is achievable.

For optimum CMRR over the entire bandwidth of the preamplifier, R_1 is compensated with capacitor C_1 to counteract the reduced gain of A2 at frequencies around f_{LP}. A good approximation for the optimal value of C_1 is:

$$C_1 \approx \frac{A_{diff}}{2\pi \cdot GBP \cdot R_1} \tag{4.11}$$

The best CMRR performance can only be achieved with equal source impedance at both the reference and recording inputs. For this reason both A1 and A2 should be of the same type. These op-amps should be selected on the basis of low input bias current (giving high input impedance), low noise voltage density, and high CMRR.

To ensure stability of this preamplifier the GBP of the two integrator op-amps, A3 and A4, should be greater than that of A1 and A2. These two op-amps should also have a low input bias current for good performance as integrators. The input voltage range of these op-amps should also be higher than the output voltage range of A1 and A2 to a void the possibility of 'latch-up' in the preamplifier.

4.2.1 The Driven Electrode

Figure 4.2 shows a basic model of common mode interference. Note that the differential signal itself is modeled as two voltage sources, each with half the amplitude of the signal. The mid-point of these two sources is connected to another voltage source that represents the common-mode interference encountered during recording. This source is referenced to the amplifier common. The two signals at the inputs to the amplifier are then:

$$V_+ = \frac{V_{diff}}{2} + V_{cm} \qquad \text{and} \qquad V_- = \frac{-V_{diff}}{2} + V_{cm} \tag{4.12}$$

By inverting the V_- signal and then adding the result in series with the common connection it is possible to force the V_- input to zero, making the V_+ input equal to V_{diff}, hence significantly reducing the common-mode interference before the signal is applied to the amplifier, and greatly improving the CMRR performance of the system overall.

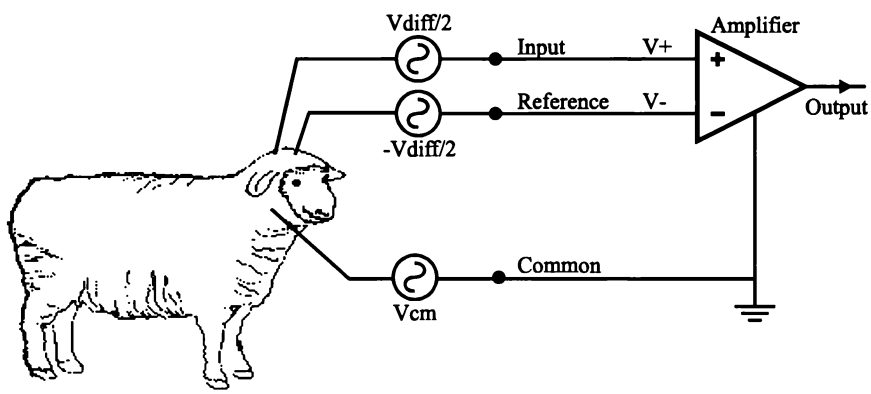


Figure 4.2 – Basic Common-Mode Interference Model

The purpose of a driven electrode[94,95] is to attempt to reduce any common-mode interference before the signals are amplified. The circuitry associated with the driven electrode inverts the common-mode signal present in the signal leads and places the resulting signal in series with the common connection to the subject, actively reducing the common-mode interference. In addition, the driven electrode configuration isolates the subject from the amplifier common, limiting any dangerous currents that could develop due to a fault either in the amplifier itself or a related system.

In practice much of the common-mode interference comes from mains-frequency currents induced in the subject by nearby wiring. Another major source is low frequency interference caused by electrode polarization or muscle activation potentials associated with movement. Based on this pattern of interference, the driven electrode configuration should be optimized for mains-frequency interference rejection and should provide a DC current path for the elimination of any common-mode DC offset.

A typical driven electrode configuration is shown in Figure 4.3. The driven electrode circuitry is used as a connection between the signal source and the amplifier common. The common-mode interference is reduced by actively driving the subject to the potential of the amplifier common. The series resistor R_d isolates the subject from any fault current. Using the capacitor C to set the gain ensures that the common-mode suppression of the driven electrode increases at lower frequencies, compensating for the decreasing CMRR of the preamplifier at lower frequencies. In practice the additional CMRR provided by the driven electrode is usually limited to about 40dB at 50Hz to ensure overall stability of the amplifier system.

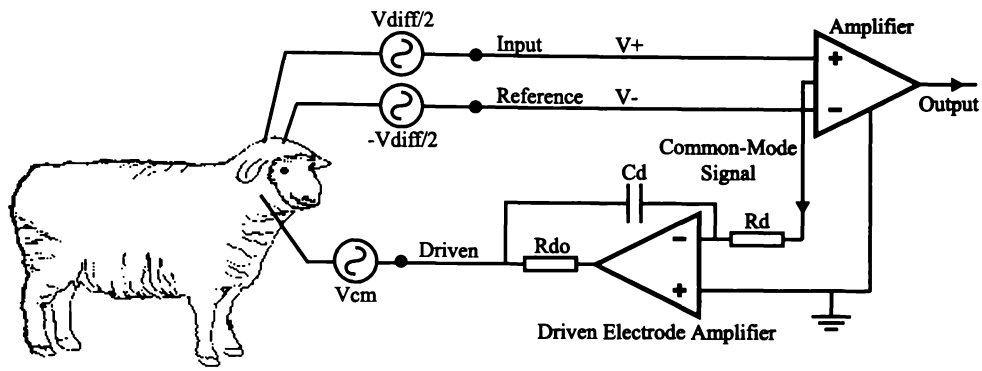


Figure 4.3 – Typical Driven Electrode Circuitry

Because the use of a driven electrode closes another feedback loop, care must be taken to ensure that the overall system remains stable. In this case the feedback loop includes unpredictable and time-varying stray capacitance and resistance due to the subject/electrode interface. The verification of circuit stability allowing for the variable nature of this loop requires simulation and testing over a wide range of electrode interface parameters.

4.2.2 Input Lead Guarding

Any imbalance in the input impedance of the preamplifier can have a major negative effect on CMRR performance. This becomes more significant at higher frequencies where stray capacitance begins to dominate. The major sources of this imbalance are the electrodes and the stray capacitance associated with cabling.

Figure 4.4 shows a typical model[94] of a set of electrodes with associated cabling, stray capacitance, and the input impedance of the preamplifier. Note the stray capacitance (C_{pow}, C_{body}) that results in a small amount of mains frequency current flowing through the subject and the isolation capacitance (C_{iso}) that results in a portion of this current flowing through the amplifier common. Also shown are the electrode impedance (Z_{e+}, Z_{e-}), the cable shield capacitance (C_{si}), and the amplifier input impedance (Z_{in+}, Z_{in-}).

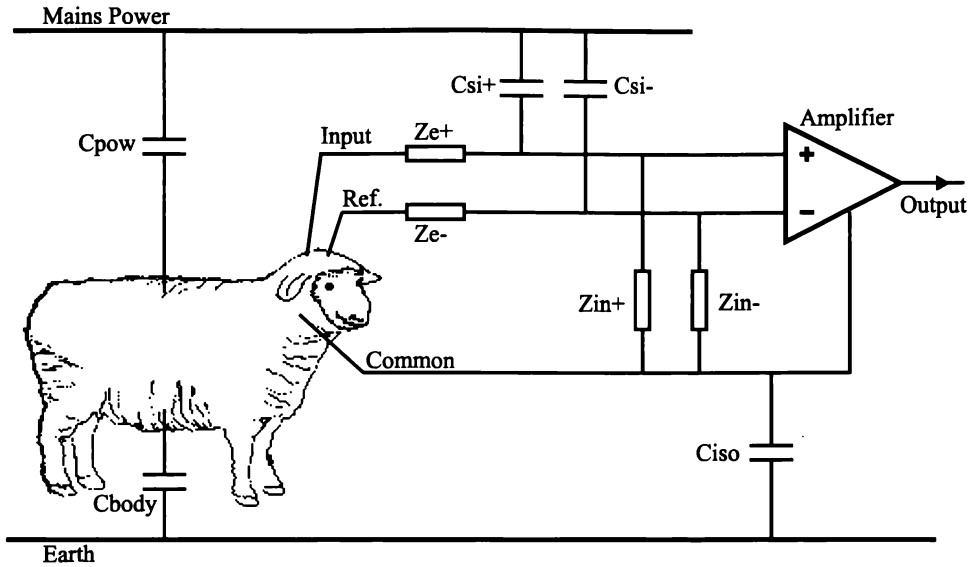


Figure 4.4 – Typical Recording Configuration Model

The electrode, cable, and input impedance of the amplifier form a potential divider that produces a (small) change in the amplitude and phase of the signal. If the amplitude and phase change is different between the recording and reference inputs, due to an impedance mismatch, the difference takes the form of a differential signal and is amplified by the preamplifier. In this way a common-mode signal can be transformed into a differential signal. A relatively small impedance imbalance can thus have a major effect on the CMRR.

The easiest way to combat this problem is to design an amplifier with a very high input impedance. If V_s is the signal at the subject and V_{amp} is the signal at the amplifier input, the potential divider[96] can be written as:

$$\frac{V_{amp}}{V_s} = \frac{R_{in}}{R_{in} + Z_e} \tag{4.13}$$

where R_{in} is the (perfect, assuming input capacitance is very small) preamplifier input impedance and Z_e is the electrode impedance. As R_{in} gets very large, impedance imbalances in the electrodes have less effect, reducing interference transformation problems.

However if the effect of the capacitance between the shield and the core of the electrode cables is included, increasing the input impedance no longer helps. If C_s is the shield capacitance the preamplifier input impedance becomes frequency dependent:

$$Z_{in} = \frac{R_{in}}{1 + j\omega C_s R_{in}} \tag{4.14}$$

Any mismatch or variation in cable capacitance now affects the CMRR performance of the preamplifier.

If the cable shield can be driven to the same potential as the common-mode signal in the cable core, the cable capacitance as it appears to common-mode signals can be effectively reduced to zero. If the effective capacitance becomes very small the potential divider effect can be significantly reduced. This technique is usually known as guarding[97]. Note that the cable capacitance still remains for differential mode signals, however for cables of moderate length this has minimal effect.

A typical cable guarding circuit is shown in Figure 4.5. The guarding circuitry consists of a buffer with current limiting. The common-mode signal gain of the reference signal amplifier is attenuated to 0.99 to guarantee stability. The resistor R_G protects the guard circuit output from fault conditions.

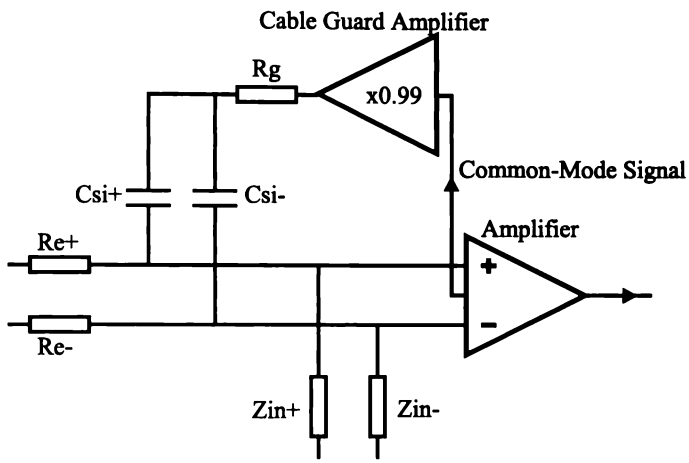


Figure 4.5 – Typical Cable Guarding Circuit

With the addition of the cable guard, system stability becomes much more difficult to ascertain. However the use of packages such as SPICE enable a large combination of parameters to be tested quickly and accurately.

4.3 Simulation

Figure 4.6 shows the preamplifier circuit used for the SPICE simulations. Note the signal source model consisting of common-mode interference, DC offset, and signal. Basic electrode and cable models are also presented. The main signal amplifiers, A1 and A2, were based around LT1012[i] op-amps. The main features of the LT1012 is presented in Table 4.2. The excellent input bias current and CMRR specifications are achieved at the expense of a moderate input noise voltage spectral density.

Parameter	Value
Input bias current	50pA (typ.)
Common mode rejection ratio (CMRR)	130dB (typ.)
Gain-bandwidth product (GBP)	1MHz (typ.)
Input noise voltage spectral density	15nV/√Hz at 100Hz (typ.)
Input noise current spectral density	8fA/√Hz at 100Hz (typ.)
Input voltage range (on ±5V supply)	±4V (typ.)

Table 4.2 – LT1012 Operational Amplifier Key Specifications

The integrators, A3 and A4, were based around TL071[ii] op-amps. These op-amps were selected for low input bias current and high Gain-Bandwidth Product (GBP). The relevant features of the TL071 for this application are presented in Table 4.3.

Parameter	Value
Input bias current	65pA (typ.)
Gain-bandwidth product (GBW)	3MHz (min.)
Input noise voltage spectral density	25nV/√Hz at 100Hz (typ.)

Table 4.3 – TL071 Operational Amplifier Key Specifications

The version of SPICE used for the simulations was the Sim98® package, part of the Protel98® suite[iii]. SPICE models of the LT1012 and TL071 were obtained from their respective manufacturers.

The aim of the simulations was to prove that the design was suitable for recording Evoked Response waveforms and to optimize the performance of the preamplifier for the task. Stability of the preamplifier combined with the driven electrode and the input lead guarding was also examined.

---

[i] LT1012 Precision Operational Amplifier, Linear Technology Corp., Milpitas, CA 95035, USA
[ii] TL071 JFET Operational Amplifier, Texas Instruments Inc., Dallas, Texas 75265, USA
[iii] Protel98 Electronic Design System, Protel International Inc., Frenchs Forest 2086, NSW, Australia

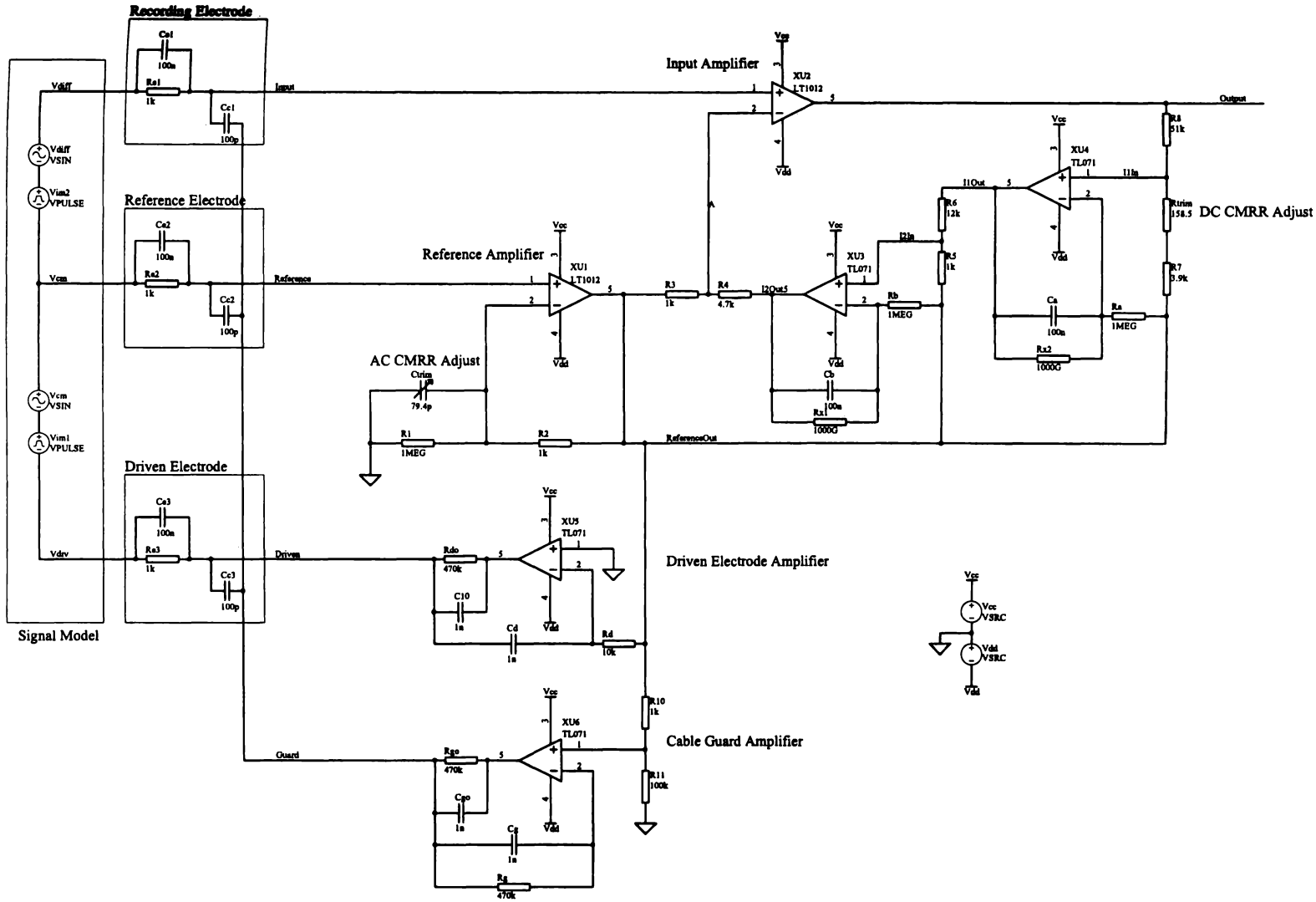


Figure 4.6 – The Simulated Preamplifier Circuit

Title	EP Pre-Amplifier Differential Amplifiers
Drawn by:	Phil Harris
Size: A4	File: L:\WELFARE\PapersDesign\Misc\EPPreAmplifier\Schematics\
Number:	Revision:
Date: 14-Dec-1999 Time: 20:45:09	Sheet 1 of 1

Engineering Development Group
Hori Research NZ Ltd
Rauhara Research Centre
Rauhara Road

4.3.1 Frequency Response

The frequency response of the preamplifier was simulated by connecting the V_{cm} node to the preamplifier common. The signal source V_{diff} was set to an amplitude of 1V and a small-signal analysis performed. Figure 4.1 shows the response of the preamplifier. The bandwidth is from 2.5Hz to 600Hz and the pass-band gain is 60dB or 1000. The two integrators produce a 2nd order high-pass roll-off at 40dB/decade. The low-pass corner is limited by the main amplifier A2. The LT1012 has a typical GBP of 1MHz as stated in the manufacturers datasheet, however the SPICE model of the LT1012 has a GBP of only 600kHz. For the required preamplifier gain of 1000 this predicts a bandwidth of 600Hz, in agreement with the simulation.

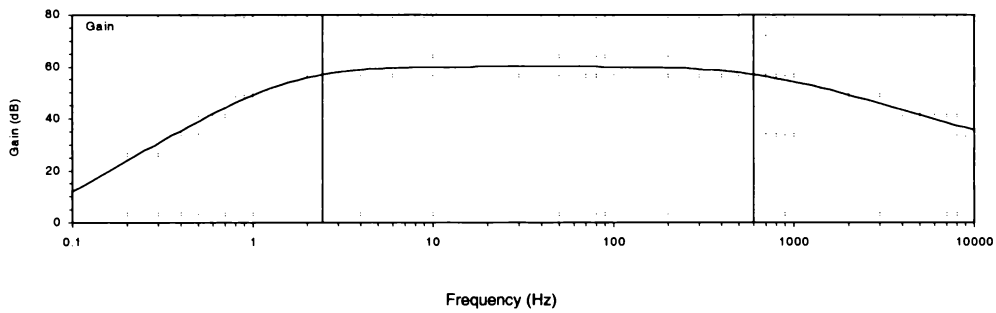


Figure 4.7 – Simulated Frequency Response of the Preamplifier

It is worth noting that the stated gain-bandwidth performance of operational amplifiers is often less than the actual performance. Although the 600Hz limit was lower than required, there was a good chance that the preamplifier would perform close to target when built.

4.3.2 Noise Performance

The preamplifier noise performance is simulated by SPICE at the same time as the small-signal analysis. All noise sources in the preamplifier are included and the resulting noise at the output is divided by the preamplifier frequency response to give an input-equivalent noise spectral density. Figure 4.8 shows the results of the simulation.

The input-equivalent noise spectral density is 36nV/√Hz in the pass-band with a $1/f$ or 'flicker' noise corner of 0.35Hz. Integrating the noise spectral density over the preamplifier's bandwidth gives a total noise figure of 1.05μV. Both the noise spectral density and the total noise compare well with the required specifications.

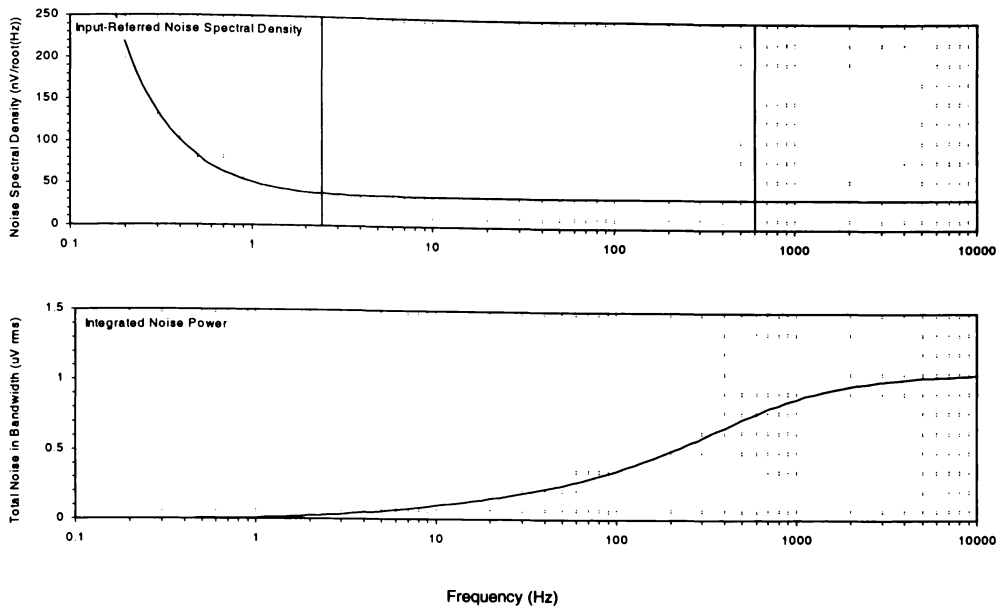


Figure 4.8 – Simulated Preamplifier Noise Performance

Because the noise corner frequency is below the high-pass corner of the preamplifier frequency response, and the total preamplifier bandwidth is much larger than the flicker noise corner, the flicker noise has little effect on the total noise generated by the preamplifier. On this basis relatively noisy operational amplifiers such as the LT1012 are acceptable.

Since the noise spectral density is dominated by the performance of the amplifiers A1 and A2, it could be expected that actual performance may be better than the simulation. This is because the SPICE models use the amplifier manufacturer's guaranteed noise specifications rather than the typical values encountered in amplifiers that have passed the manufacturer's quality control testing.

4.3.3 DC Input Offset Suppression

The recovery of the preamplifier from the imposition of a large DC offset was simulated by connecting the V_{cm} node to the preamplifier common. The signal source V_{diff} was set to an amplitude of 1mV and the transient source V_{im2} was programmed to produce a 7s duration, 500mV amplitude offset. The resulting signals at the input, integrator outputs, and preamplifier output were recorded. Figure 4.9 shows the results of the simulation. The signal is invisible in the input and A3 output traces but is visible in the A4 output trace and at the preamplifier output. Note that the input signal of 1mV has also been amplified to 1V, a factor of 1000.

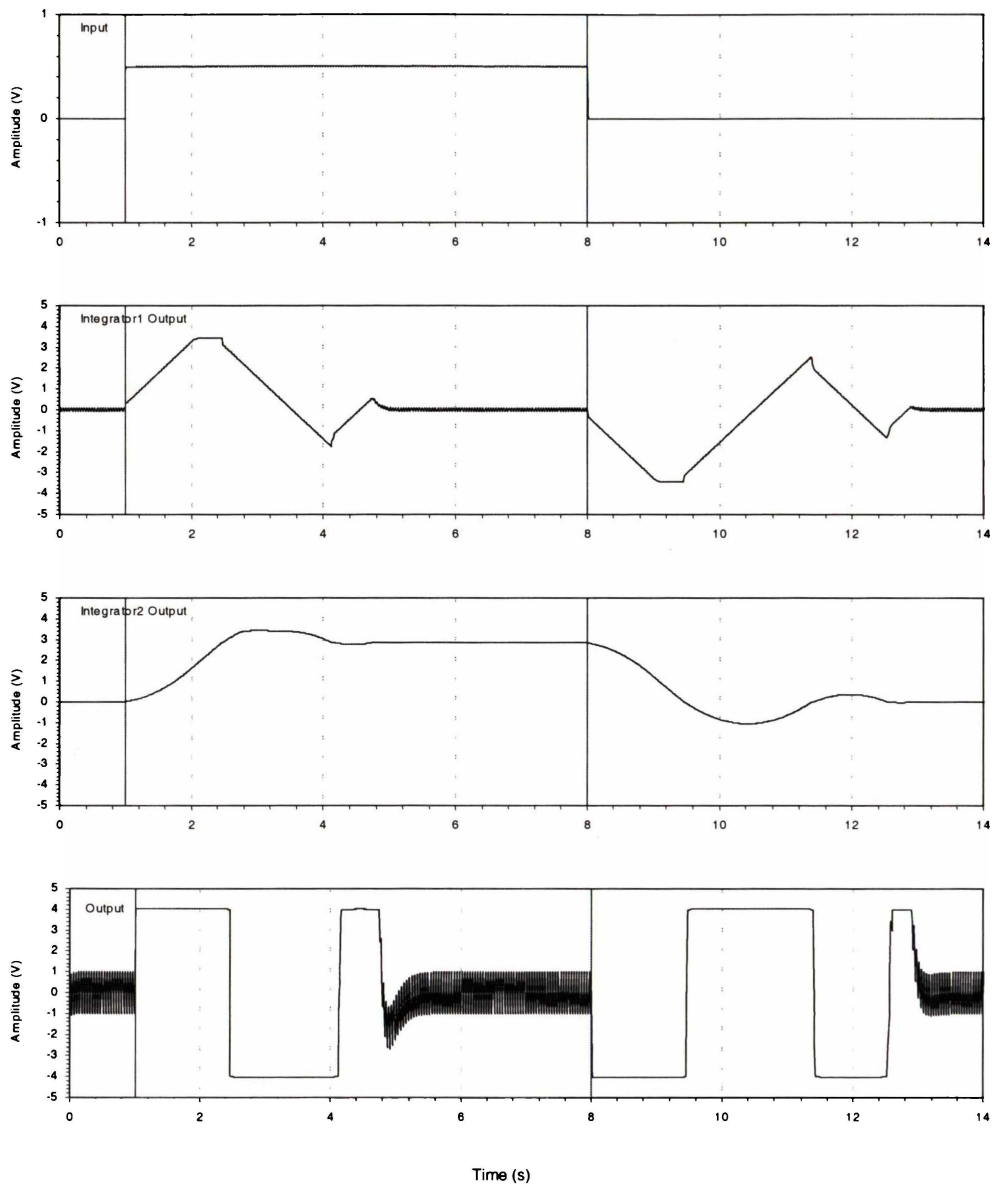


Figure 4.9 – Simulated Preamplifier DC Offset Suppression Performance

As soon as the offset is generated the output of the two integrators begins to rise, forcing the inverting input of A2 to the same amplitude as the input offset. After a period of approximately 5s

the output of the preamplifier is returned to a stable baseline. At this point the output of A3 is 2.85V, and the inverting input of A2 is at 500mV. Both positive and negative transitions of the DC offset show a return to a stable baseline within 6s.

The predicted DC rejection performance of the preamplifier can be calculated from this simulation. The input offset is 500mV and the simulation gives a stable output offset of 35mV at 7s. The preamplifier has an overall gain of 1000. The DC rejection ratio is then:

$$DCRR = 20\log\left(\frac{A_{diff}V_{in}}{V_{out}}\right) = 20\log\left(\frac{1000*0.5}{0.035}\right) = 83\text{dB} \qquad (4.15)$$

Simulations showed that the preamplifier could effectively suppress offsets of up to ±600mV. This compares well with the desired range of ±500mV.

4.3.4 Common-mode Interference Rejection

The common-mode rejection performance of the preamplifier was tested both with and without the driven electrode configuration. In both cases the cable guard was disabled by connecting the Guard node to the preamplifier common. To simulate the preamplifier without the driven electrode, the V_{drv} node was connected to the preamplifier common. The common mode interference was simulated by setting the source V_{cm} to an amplitude of 1V and performing a small-signal analysis. The preamplifier common-mode gain was measured at the preamplifier output.

Figure 4.10 shows the results of the simulation. Without the driven electrode the preamplifier manages a peak CMRR of 123dB at 50Hz, while adding the driven electrode increases this figure to 173dB. This is a 50dB improvement due to the gain of the driven electrode amplifier:

$$A_{drv} = -\frac{1}{\omega C_d R_d} = -\frac{1}{2\pi(50\text{Hz})(1.0\text{nF})(10\text{k}\Omega)} = 50\text{dB} \qquad (4.16)$$

At low frequencies, the driven electrode also performs well at removing any DC common-mode offset from the subject due to its integrator configuration.

Simulations with varying values for the AC CMRR adjust capacitor Ctrim and the DC CMRR adjust resistor Rtrim (Figure 4.10), gave values of 158Ω for Rtrim and 79pF for Ctrim to achieve maximum CMRR.

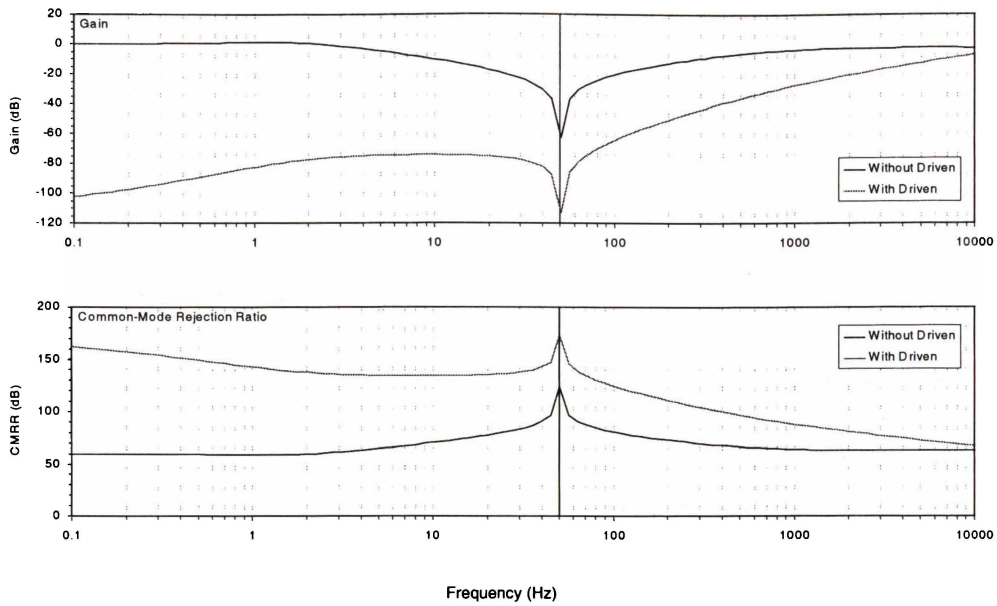


Figure 4.10 – Simulated Preamplifier Common-Mode Interference Rejection Performance

Note that without the driven electrode the preamplifier common-mode gain at low frequencies is 0dB. When both the signal and reference inputs of the preamplifier are at the same potential the differential amplifier configuration becomes equivalent to a unity-gain buffer. This is because the integrators in the feedback path result in a very low loop gain at low frequencies.

The total CMRR value predicted by the simulation is very high. Splitting this figure between the main amplifier and the driven electrode helps to make this figure justifiable. However, the simulation does not include the effect of mismatched component values due, perhaps, to component tolerances. Simulation showed that component tolerances of 1% could lead to a degradation of around 20dB. Taking this into account, the best CMRR performance that can probably be hoped for is around 100dB for the preamplifier alone, or 150dB for the preamplifier with the driven electrode.

4.3.5 Stability

To assess the overall preamplifier stability with both the driven electrode and the cable guard in operation, and typical electrode and cable capacitance values, the source V_{cm} was set to 1V and a small-signal analysis was performed. The signal source V_{diff} plays no part in the overall stability since it is outside the feedback loops formed by the reference amplifier, the driven electrode amplifier, the cable guard amplifier, and the electrode and cable models. The basic preamplifier without the driven electrode and cable guard was found to be unconditionally stable. The feedback loop gain was measured at the reference amplifier output. The results of the simulation are shown in Figure 4.11.

The electrode impedance Re was varied from 10Ω to 100kΩ, while the cable capacitance Cc was varied from 1pF to 10nF. In practice, typical values[90] for the electrode impedance are 1kΩ to 10kΩ, while most screened cables have a capacitance of around 100pF/m. When varying the electrode impedance the cable capacitance was set to 100pF. When varying the cable capacitance the electrode impedance was set to 1kΩ.

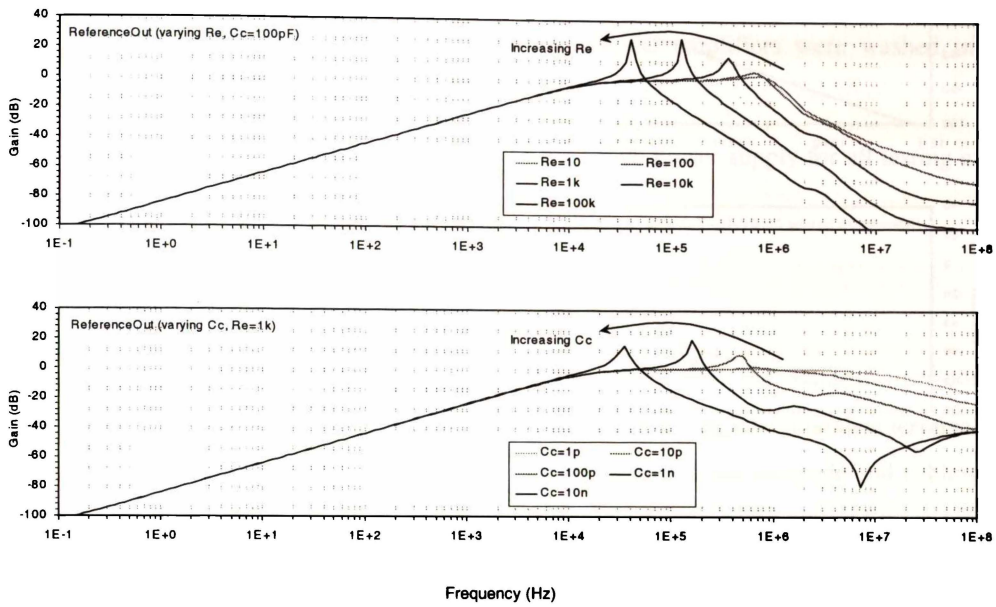


Figure 4.11 – Simulated Preamplifier Stability for Changing Electrode Resistance

Both plots show that instability rises as either Re or Cc increases. For large values of Re and Cc the gain can be greater than 20dB, but for all values the gain can exceed unity, indicating a system likely to oscillate. The notch in the frequency response at lower frequencies is due to the CMRR being tuned to a maximum at 50Hz.

Note that the resonance peaks in both plots occur at the same frequencies indicating that the major cause of the resonance is a pole in the frequency response created by a combination of the electrode impedance and the cable capacitance. Further investigation showed that the resonant feedback path also contained the driven electrode amplifier. This gives three options for eliminating the instability: reduce the cable capacitance by using short cables, reduce the electrode impedance, or reduce the loop gain by reducing the gain of the driven electrode amplifier.

The first two options are limited by physical considerations, reducing their usefulness. The last option is viable but would result in a reduction in the CMRR. However, further investigation showed that by including the effects of capacitance in the electrode (placing a capacitance Ce in parallel with Re), the resonance was reduced and the system become more stable. Cooper[98] states a typical value for Ce of 50nF for a 2mm length of 0.15mm diameter stainless steel wire. To

simulate the 0.7mm diameter, 24mm long needle electrodes that would be used with this preamplifier, an estimated lower limit of 100nF was used for the electrode capacitance Ce. The results of this simulation are shown in Figure 4.12.

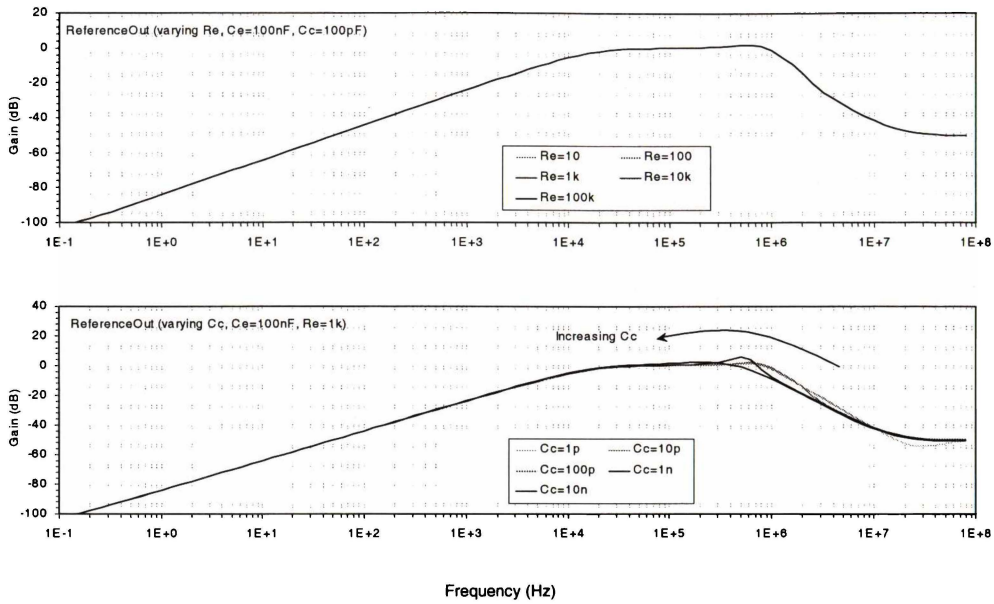


Figure 4.12 – Simulated Preamplifier Stability with Changing Electrode Capacitance

The plots show that including the effects of electrode capacitance in the simulation greatly reduces the chance of instability. The electrode capacitance dominates the electrode resistance at frequencies of interest, causing the plots of varying Re to completely overlap. Varying the cable capacitance still shows some resonance peaking of about 6dB at about 500kHz. However this is for a cable capacitance of 10nF, corresponding to a 10m length of cable. In practice cable lengths are less than 1m and for these lengths the preamplifier is stable. Simulation also showed that varying Ce between 1nF and 10μF did not adversely effect preamplifier stability.

Based on these simulations the preamplifier design was considered to be stable under reasonable recording conditions.

4.4 Actual Performance

Based on the results of the simulations, six identical preamplifiers were constructed. Each preamplifier consisted of two signal channels, with shared reference, driven electrode, and cable guard sections.

The preamplifiers were constructed using 1% tolerance metal film resistors and 5% tolerance ceramic capacitors. Each operational amplifier type was from the same manufacturing batch. Standard surface mount construction was employed and the amplifiers were washed in flux remover after assembly to ensure cleanliness.

The preamplifier was powered by an isolated dual ±5V bench power supply for all testing except the noise performance where two 9V batteries were used.

4.4.1 Frequency Response

The gain of the preamplifiers signal channels was tested by applying a transformer-coupled 700μV, 100Hz test signal between the input and reference terminals. The reference terminal was also connected to the preamplifier common. The outputs of the driven electrode and cable guard amplifiers were also connected to the common. The transformer coupling ensured there were no ground loops between the signal generator and the oscilloscope.

The average gain of the 12 preamplifiers signal channels was 940. The minimum and maximum values were 906 and 961 respectively. This compares well with the gain of 1000 estimated by the simulation.

The actual gain differed from the predicted gain because the amplifiers were adjusted for maximum CMRR performance rather than exact gain. To obtain maximum CMRR the resistor Rtrim is adjusted, directly affecting the feedback loop gain and also accounting for the relatively large range of gain values. This average gain value suggests that the actual value of R1/R2 is 939 rather than the 1000 to be expected with perfect components (as used in the simulation). This suggests an average error of around 6.3% in the value of either R1 or R2, much higher than expected from the use of 1% tolerance components.

The frequency response of the preamplifier was tested in the same manner as the pass-band gain, but the signal frequency was varied to find the –3dB points in the preamplifier frequency response. The average upper frequency limit of the amplifiers was found to be 1.78kHz, considerably in excess of the 600Hz predicted by the simulation and the manufacturers datasheet. The average lower frequency limit was found to be 1.68Hz, lower than the 2.5Hz predicted by simulation.

The bandwidth was considerably larger than expected and with this design limiting the bandwidth can cause problems with tuning the preamplifier for optimal CMRR, especially with two signal

channels. A better option is to rely on later buffering and level matching stages to limit the bandwidth.

The FRPM Evoked Response input channels consist of the preamplifier, a level translator which amplifies the signal by a factor of two and shifts the 0V DC output offset of the preamplifier to have a DC offset of 1.5V, and a buffer that amplifies the signal by a factor of two before the signal is digitized by a analog-to-digital converter (ADC). Both the level translator and buffer include filtering to prevent any signal frequency content above 1kHz, providing anti-aliasing filtering for the ADC. The preamplifier design uses these two extra stages to meet the 1kHz bandwidth specification for the preamplifier.

4.4.2 Noise Performance

Noise measurements were made using a digital oscilloscope to record the output noise which was then analyzed using Fast-Fourier Transform (FFT) techniques to extract the noise spectral density (NSD) of the preamplifier.

To test the noise performance of the preamplifier the input and reference terminals were connected to the preamplifier common via high quality 10kΩ metal film resistors. The outputs of the driven electrode and cable guard amplifiers were also connected to the common. The entire preamplifier assembly was then placed in a die-cast aluminum enclosure, with the enclosure also connected to the preamplifier common. The preamplifier was powered by two 9V batteries during the noise testing, with the batteries also inside the enclosure. The output of the preamplifier was connected to a BNC-type coaxial connector, the only connection from the outside to the inside of the enclosure, to enable noise measurements with the preamplifier completely shielded and isolated from earth.

The recorded noise signal consisted of four 4096 sample recordings. Four recordings were made to enable averaging of the results of the Fourier analysis. The sample rate was 2.5kHz, giving a total recording time of 1.638s, a frequency resolution of 0.61Hz and a Nyquist frequency of 1250Hz. The samples had 8-bit resolution. The DC offset was removed from the samples prior to Fourier analysis. Each recorded segment of noise was transformed using the FFT, the results of the four transforms were averaged and the NSD calculated. The noise contribution from the 10kΩ input resistors was then subtracted to give a final NSD for the preamplifier alone. Figure 4.13 shows the results of this analysis.

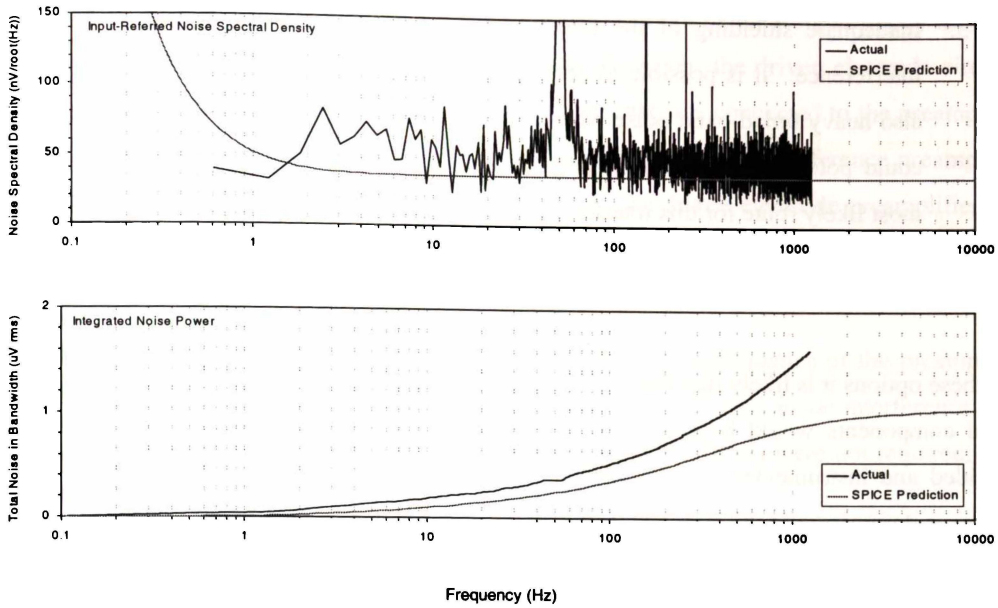


Figure 4.13 – Actual Preamplifier Noise Performance

The first plot shows the calculated NSD of the preamplifier. Despite the care taken with shielding and grounded during noise testing, the fundamental mains frequency of 50Hz plus its 3rd, 5th, and 7th harmonics are clearly visible. For calculating the average NSD and the integrated noise power the contributions from these interference sources were removed. The average NSD is 45nV/√Hz, equivalent to the noise generated by a 1.5kΩ resistor at 300K. This compares well with the 36nV/√Hz predicted by the simulation.

The flicker noise of the preamplifier is not visible because of the sharp roll-off at low frequencies caused by the integrators. The Fourier calculations derived the input-equivalent NSD by dividing the output NSD by the preamplifier gain at 100Hz. This contrasts with the SPICE prediction where the output NSD is divided by the preamplifier gain as a function of frequency. Because of the low gain at low frequencies, SPICE causes a large boost to NSD at low frequencies resulting in the disparity.

The total noise voltage in the preamplifier bandwidth is greater than the SPICE simulation since the actual NSD is slightly higher. In practice the level matching and filtering circuitry limits the bandwidth to 1kHz resulting in a total input equivalent noise voltage of about 1.7μV, higher than the required specification of 1μV, but still acceptable.

It is uncertain where the discrepancy between the predicted and actual noise figures arises from. The three main possibilities are:

1. The manufacturer supplied models underestimate the noise of the devices they model. This would seem unlikely due to the conservatism already mentioned in the models
2. 'Extra' noise generated by non-ideal real-world components. In particular the resistors actually used in construction, but also possibly noise from imperfect capacitors

3. Inadequate shielding of the circuitry under test from environmental electromagnetic interference. It is possible to see heavy interference at mains frequencies, but there is also heavy broad-band noise whose origin is uncertain: it may be from the amplifier but could potentially be radiated or conducted interference from an external source. The most likely route for this interference to be conducted into the circuit under test would be through the test and measurement equipment itself.

Of these options it is likely that the final is the major contributor, although it is expected that non-ideal components would have some effect. Further testing of the preamplifier in a properly shielded and instrumented electromagnetic compatibility testing room would determine which source is dominating in this case. Unfortunately, access to such a facility to further investigate the noise performance of the preamplifier was unavailable.

4.4.3 DC Input Offset Suppression

The DC Input Offset Suppression was tested by connecting the reference input to the preamplifier common and connecting the signal input to a variable voltage source. The outputs of the driven electrode and cable guard amplifiers were also connected to the common. The offset voltage was varied from –600mV to 600mV and the DC offset at the preamplifier output was measured.

The average DC rejection ratio over the full range was 101dB. The limits for successful canceling of the input offset were 683mV and –626mV. All of these results are well within the specifications and are consistent with the simulation.

4.4.4 Common-mode Interference Rejection

The CMRR of the preamplifiers signal channels was tested by applying a transformer-coupled 1V, 50Hz test signal between the signal input terminal and the preamplifier common. The reference terminal was connected to the signal input terminal. The outputs of the driven electrode and cable guard amplifiers were connected to the common.

The CMRR was measured with a digital oscilloscope set to averaging mode with the time-base synchronized to the signal generator. This arrangement ensured that the 50Hz signal could be extracted and accurately measured at the output regardless of any noise that could be present.

The average CMRR of the 12 preamplifier signal channels was 100dB. The minimum and maximum values were 96.1dB and 103.4dB respectively. This is significantly less than the predicted value of 123dB and is probably due to component tolerance mismatches. Also, because both channels of each preamplifier share a common reference preamplifier, the CMRR was adjusted for a best average value for the two channels rather than maximizing the performance of a single channel.

The CMRR of the preamplifier with the driven electrode could not be measured directly as it was beyond the capabilities of the available instrumentation. However, the driven electrode circuitry was tested by moving the lead of the coupling transformer that was connected to the preamplifier common to the driven electrode terminal and measuring the output of the reference preamplifier using the same synchronized time-base method as above. The driven electrode preamplifier was found to give 50dB of common-mode suppression at 50Hz, in agreement with simulation and direct calculation.

The CMRR tests suggest upper and lower limits for the CMRR performance of the preamplifier of 150dB and 100dB respectively. In practice no problems with common-mode interference were experienced with any of the amplifiers and mains-frequency notch filters were not required, even with the nearby presence of fluorescent lamps or large induction motors.

4.4.5 Stability

The stability of the completed preamplifier was tested by using the signal generator and coupling transformer setup used for the CMRR test to look for resonance peaks at high frequencies.

The signal input and reference terminals of the preamplifier were connected to one terminal of the secondary winding of the transformer via a 1m length of coaxial cable (Cc=100pF) and a 10kΩ resistor in parallel with a 100nF capacitor. The driven electrode terminal was connected to the other secondary winding terminal through another 1m length of coaxial cable and resistor/capacitor combination. The two resistors simulated the electrode resistance (Re=10kΩ) while the capacitors simulated the electrode capacitance (Ce=100nF). The cable guard terminal was connected to the shields of both coaxial cables. This arrangement was intended to simulate a typical recording environment.

The output of the signal generator was then swept over a frequency range of 1Hz to 10MHz with an amplitude of 1V, while the amplitude of the signal at the output of the reference preamplifier was observed on an oscilloscope.

No instability was detectable on the oscilloscope at any frequency, suggesting the preamplifier was stable. During actual recording sessions no oscillation or other instability was observed in any of the six preamplifier units built.

The SPICE simulation showed a slight tendency to oscillation at frequencies above 100kHz. However this was not observed in practice. The lack of instability is probably due to variation in component tolerances dampening any oscillation sufficiently to prevent instability.

4.5 Discussion

Table 4.4 gives a summary of the performance of the preamplifier based on the initial requirement, simulation, and actual performance. Apart from its gain and noise performance, the constructed preamplifier performs better than the requirements and comparable to the SPICE simulation.

Parameter	Requirement	Simulation	Actual
Differential Gain (at 100Hz)	1000	1000	940
Bandwidth (-3dB)			
High-pass corner	Around 1Hz	2.5Hz	1.68Hz
Low-pass corner	Greater than 1kHz	600Hz	1.78kHz
Noise			
NSD (at 100Hz)	Less than 30nV/√Hz	36nV/√Hz	45nV/√Hz
Total (in bandwidth)	Less than 1μV total	1.05μV total	1.7μV total
Input Impedance	>10MΩ	Not simulated	>10MΩ[#]
Input Bias Current	Less than 200pA	30pA	Not measured[+]
Common-Mode Rejection Ratio (at 50Hz)	Greater than 100dB	123dB without driven 173dB with driven	100dB without driven 150dB with driven[*]
Input Offset Rejection			
Range	Greater than ±500mV	±600mV	±626mV
Rejection Ratio	Not specified	83dB DCRR	101dB DCRR
Power Consumption	Not specified	404mW	175mW
Size			48mm by 34mm by 9mm
Weight			19g for assembled printed circuit board
Construction			2 layer printed circuit board using surface mount components for minimum size and weight

[#]Actual figure beyond measurement limits of available instrumentation
[+]Could not be measured with available instrumentation
[*]Could not be measured directly. Inferred from preamplifier CMRR plus driven electrode CMRR

Table 4.4 – Summary of Preamplifier Performance

The difference in gain can be compensated for in software although it would probably be better to add an additional adjustable gain stage after the preamplifier to avoid unit-to-unit variation. The discrepancy in noise performance is probably of the most concern but is still considered acceptable.

The input impedance could only be measured as being much greater than 10MΩ due to limitations of available instrumentation. The limiting factors for this figure include the intrinsic resistance of the PCB material and the insulation resistance of the connectors and cables, hence it is unlikely to be greater than about 100MΩ. This range is much greater than expected electrode

impedances, virtually eliminating potential divider effects that could be caused by a lower amplifier input impedance.

The 404mW power consumption predicted by simulation of is well in excess of the actual power consumption of 175mW. No reason for this discrepancy could be found. Although the actual power consumption might be considered high, the preamplifier is only powered during relatively short recording times. This reduces the average power consumption and is the reason no maximum power consumption was specified.

In practice the six constructed preamplifier units performed well in all respects. No changes to design or component values were needed (apart from the expected adjustment of Ctrim and Rtrim for maximum CMRR performance) and all amplifiers worked correctly when first powered.

No instability was observed during recording. The driven electrode configuration and high CMRR of the preamplifier eliminated any sign of mains frequency interference and made notch filtering unnecessary. The DC offset suppression also helped in eliminating any output baseline wander.

Due to electrical screening requirements the preamplifier was constructed on a separate PCB in a shielded compartment within the FRPM case, minimizing any possible interference from the other FRPM circuitry inside the case. Figure 4.14 shows both sides of a constructed preamplifier unit.

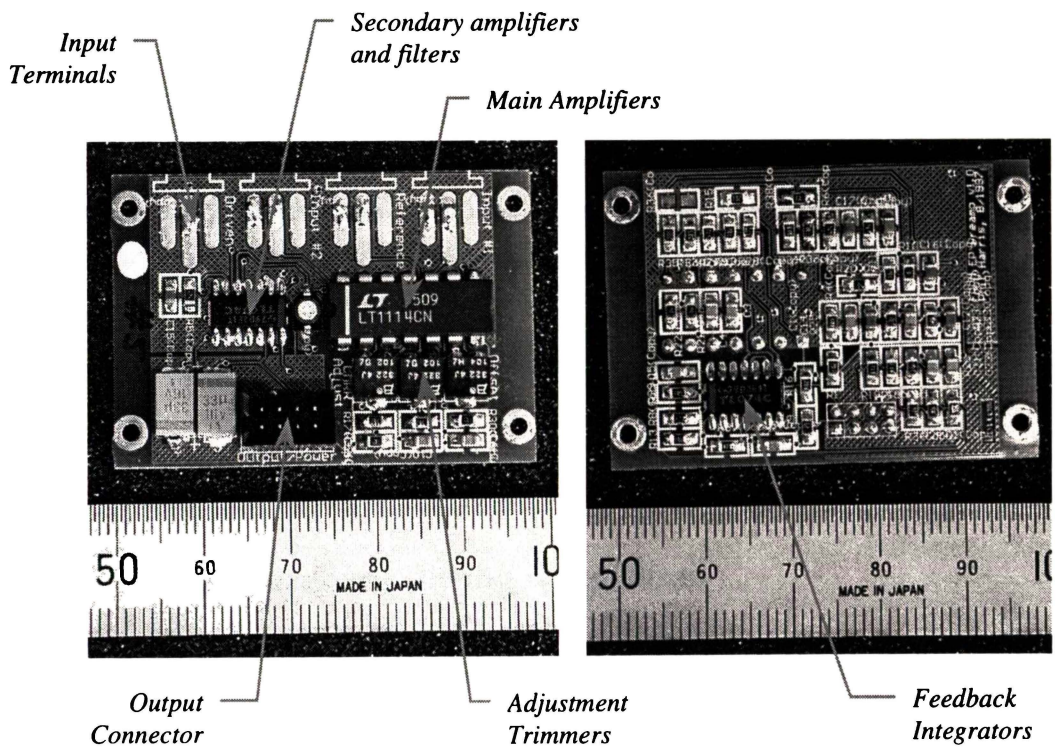


Figure 4.14 – The Completed EEG/ECG Preamplifier Board

The SPICE simulations were invaluable for designing this preamplifier. The simulations allowed rapid characterization of performance with varying component values and gave confidence that the preamplifier would work as designed. Given the large number of varying feedback loops in

typical recording configurations, simulation helped to identify potential problem areas and find solutions considerably faster than would be possible with a physical prototype.

Chapter 5 Data Processing and Analysis

The Evoked Response waveforms were processed in three steps: compression during acquisition, feature extraction after acquisition, and statistical analysis on the extracted features. The final aim of these three steps is to give a statistical probability either that a significant feature exists in an individual waveform or that a significant difference exists between two different waveforms. Given this statistical probability an experimental hypothesis can be either proved or rejected.

Many algorithms for feature extraction and classification of Evoked Response waveforms exist in the literature[99,100,101,102], but typically these use either frequency domain methods or only consider the positions of identifiable peaks. For the stress trial it was considered desirable to statistically compare the constituent features of the waveforms directly in the time domain and without the limitation of identifying individual peaks.

This chapter presents the technique developed for the statistical analysis of the Evoked Response waveforms.

5.1 Data Processing during Acquisition

The first step in the data processing sequence occurs during signal acquisition. In the case of the FRPM Evoked Response recording system this step takes place entirely within the FRPM on the animal. The aim of this step is to reconstruct an Evoked Response waveform from the recorded electroencephalogram signal with minimal corruption of the waveform by noise or motion artifacts and to give an estimation of the residual noise in the waveform to enable the robustness of the response to be determined.

5.1.1 Recording

For the Auditory Evoked Response acquisition performed on the FRPM, the electroencephalogram was typically recorded at a sample rate of 5kHz in 100ms bursts, including a 10ms baseline period prior to the stimulus, at 12-bit resolution with around 1024 clicks being presented to the ear of the animal with a repetition interval of approximately 150±50ms. The click interval also included a random component of between 0 and 100ms to ensure that click presentation did not accidentally synchronize with any external source of interference. An entire recording thus took approximately 3 minutes.

Before being sampled, the signal had been amplified by a factor of 1880 and had been filtered with a typical pass-band range of 1.7Hz to 1.8kHz. A DC offset of approximately 1.5V was added before sampling to allow the bipolar signal to be sampled by a unipolar analog-to-digital converter.

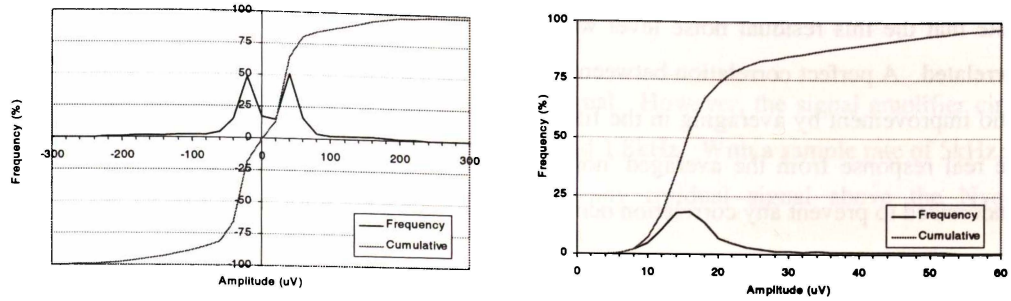
After each segment of electroencephalogram was recorded the raw sample data consisted typically of 500 signed 32-bit integer values (containing only 12 significant bits). This data then had its mean (DC) value removed before being scanned for artifacts.

5.1.2 Artifact Rejection

Artifact rejection involved scanning the raw data for single data values that were outside reasonable limits and for characteristics of the data that indicated the data had been corrupted by interference. Possible causes of data corruption included excessive signal baseline wander (e.g. amplifier saturation, faulty electrodes or electrode leads), external interference producing impulsive noise (e.g. car ignition systems), excessive high frequency noise (e.g. two-way radios), and excessive low frequency noise (e.g. mains frequency interference).

The artifact rejection algorithm considered two parameters of the recorded data to form a decision: the minimum and maximum values in the data (to identify amplifier saturation or excessive drift), and the root-mean-square (RMS) value of the data (to identify excessive noise). If either of these parameters exceeded a preset value the data was rejected as being corrupt and another segment was recorded to replace it.

The limiting values were determined by a statistical analysis of waveforms collected from several animals. Figure 5.1 shows typical distributions of the minimum, maximum, and RMS values of the data segments that made up these waveforms. These typical distributions were calculated by averaging the distributions from 12 separate recordings, which accounts for the apparent discrepancy between the cumulative and frequency plots. The limits distribution shows that more than 75% of segments recorded have maximum and minimum values of less than 60μV while the noise distribution shows that greater than 75% of segments have noise levels of less than 22μV RMS.



(a) Maximum and Minimum Values

(b) RMS Noise Values

Figure 5.1 – Typical Evoked Response Signal Characteristic Distributions

The rejection thresholds were based on these distributions by considering the effect of either a single impulse noise event or a single noisy data segment in an otherwise noise-free recording.

For impulse noise it was assumed that such an event is rare so it would be unlikely that two such events would occur in the same sample of two different data segments. The source of such impulse interference could be a motion artifact caused by excessive animal movement. Such an event can lead to amplifier saturation producing an extreme sample value; however, such events are usually short in duration and it is likely that the amplifier would have re-settled before the next segment was recorded.

A single impulse noise event of magnitude A in a single segment of an otherwise impulse free waveform comprised of N segments will produce an impulse in the output of A/N due to the averaging process (see Chapter 5). If the impulse was 60μV in amplitude in a recording comprised of 1024 segments (clicks) the resulting impulse will be 0.059μV in magnitude. For the detection of features on the order of 1μV this impulse would have little effect on the waveform. In practice a rejection limit of 130μV was chosen, which would reject approximately 10% of the segments recorded. Given these conditions, the largest impulse remaining in a recording would be limited to 0.127μV.

For the analysis of the effect of a noisy segment it was assumed that all of the recorded segments are equally noisy. This situation could occur given such circumstances as incorrect electrode positioning leading to an increase in background noise from sources such as muscle activity.

In the case of a noisy data segment, an appropriate model is that many noise sources contribute to each data value in the final waveform, and that for simplicity the noise sources are uncorrelated. If there are N segments each with a noise amplitude of A and whose noise sources are independent but have the same distribution, then the residual noise in the recording after averaging will be equal to $A/\sqrt{N}$. If the noise level was 22μV RMS in each of 1024 segments then the residual noise level would be 0.69μV. Based on a feature size of 1μV this level would again have little effect on the resulting waveform. In practice a noise level threshold of 42μV was chosen to reject only 10% of segments, giving a residual noise level of 1.3μV.

Note that the this residual noise level would be much worse if any of the noise sources were correlated. A perfect correlation between these noise sources would produce zero signal-to-noise ratio improvement by averaging in the final waveform and it would be impossible to distinguish the real response from the averaged 'noise'. For this reason the interval between stimuli was randomized to prevent any correlation occurring in the noise in different segments.

Due to this rejection algorithm more than the specified number of clicks could be presented to the animal - each time a segment of data resulting from a click was rejected an additional click would be presented to the animal to obtain replacement data. The algorithm bounded the number of additional clicks presented to the animal to 25% of the number requested. If more than 20% of the total number of clicks presented to the animal were rejected, the amount of data collected was less than was requested at the start of recording.

The maximum, minimum, and RMS values for each segment were stored in a separate file during collection for offline reconstruction of background noise profiles during the experiments. This record could then be compared off-line with the waveform to investigate the cause of distorted or noisy waveforms. Listing 5.1 shows a segment of one of the run statistics files.

```
Evoked Response Run Statistics File.
Title:,< No Title >
Filename:,erdata_07.runs.csv
Time recorded:,13:29:35 09-06-1998

Number of Runs:,763
Runs Attempted:,1280
Number of Samples:,500

Signal Path Gain:,0.2045,uV/adu
Sample Hard Limit:,132.95,uV
Signal RMS Limit:,40.91,uV

Run #,Offset (uV),RMS Value (uV),Minimum (uV),Maximum (uV)
0,372.66,22.09,-63.41,43.36
1,386.98,39.27,-83.45,159.94
2,370.00,38.04,-118.83,111.88
3,389.02,22.70,-46.63,43.77
4,377.57,83.04,-337.68,355.48
5,389.23,17.79,-35.59,45.61
...
1277,377.16,104.11,-338.09,417.04
1278,366.11,21.89,-53.79,40.91
1279,395.77,18.00,-43.36,43.16
```

Listing 5.1 – Evoked Response Recording Runs Statistics File

This segment is from a particularly bad recording and shows several instances of maximum, minimum, and RMS values exceeding the rejection threshold. From the 1024 segments requested the rejection algorithm asked for 256 extra segments to replace rejected segments but eventually quit after only keeping 763 acceptable segments. For this recording the distribution was obviously very different from the typical distribution in Figure 5.1.

5.1.3 Filtering

No digital filtering was performed on the recorded signal. However, the signal amplifier circuit contained a 4th order low-pass filter with a cutoff around 1.8kHz. With a sample rate of 5kHz, the preamplifier and subsequent filtering ensured that any residual signal above the Nyquist frequency of 2.5kHz would be reduced by at least 35dB.

Due to the high common-mode rejection ratio of the preamplifier and the electrical isolation of the instrumentation from any ground current loops, no mains power frequency notch filter was required.

5.1.4 Synchronous Averaging

As each segment of electroencephalogram was recorded and accepted it was added to the final waveform to form an average result thus providing an improvement in signal to noise ratio (SNR). This is the process of synchronous averaging whereby consecutive segments are aligned with respect to the instant of presentation of the stimuli and averaged.

The averaging process is required to provide sufficient SNR improvement to extract the small Evoked Response signal from the much larger background electroencephalogram signal. Based on typical recordings the original SNR is typically around –20dB which the averaging process improves to around +10dB.

It will be shown that the expected SNR improvement is equal to $\sqrt{N}$ where N is the number of stimuli presented. It is assumed that:

1. The signal and noise sum linearly to produce the recorded waveform
2. The signal waveform is the same for each repetition of the stimulus
3. The noise waveforms are sufficiently irregular from stimulus to stimulus that they can be considered as statistically independent samples of a random process

When uncorrelated noise sources are added together, the total noise power is equal to the sum of the individual noise powers[103]. Adding N noise voltage sources, Vn_i, together on a power basis gives:

$$Vn_{total}^2 = \sum_{i=1}^{N} Vn_i^2 \tag{5.1}$$

If all of the noise sources are treated as having the same distribution and zero mean, this equation simplifies to:

$$Vn_{total}^2 = NVn^2 \Rightarrow Vn_{total} = \sqrt{N}Vn \tag{5.2}$$

i.e. the summation of the N noise sources is equivalent to constructing a single noise source, Vn_{total}, $\sqrt{N}$ times larger in magnitude, with the same distribution, as the individual Vn_i noise sources.

Since the signal part, Vs, of the recording is always correlated (by definition), the summation of N such signals produces a larger signal N times larger in magnitude than the original.

The original signal to noise ratio was:

$$SNR_1 = \frac{Vs}{Vn} \tag{5.3}$$

The final signal to noise ratio after summation is:

$$SNR_2 = \frac{NVs}{Vn_{total}} = \frac{NVs}{\sqrt{N}Vn} = \sqrt{N}\frac{Vs}{Vn} \tag{5.4}$$

The SNR improvement from averaging N waveforms is $\sqrt{N}$. Note that this derivation is equivalent to considering an averaging process since the final division operation in averaging will have no effect on the SNR ratio.

5.1.5 Residual Noise Estimation

If each segment is considered to be one row of a matrix where each column is the position of a single sample and there are N rows then averaging the samples in each column produces the row vector that represents the final waveform. Taking this one step further, if N is large and the noise is assumed to be normally distributed then a good estimate of the residual noise at each sample position can be obtained by also recording the standard deviation of the samples in each column. This method is attributed to Elberling *et al.*[104].

This standard deviation allows a significance level to be placed on the amplitude difference between any features present in the waveform, giving the ability to separate real features from 'features' due to the effects of noise. Figure 5.2 illustrates an Evoked Response waveform with significance limits generated by this method. These limits indicate 99% significance levels from a zero mean. The time at which the stimulus occurred is t=0. This type of representation clearly shows that the response recorded was not due to chance alone and provides the basis for extracting only statistically significant features for further analysis.

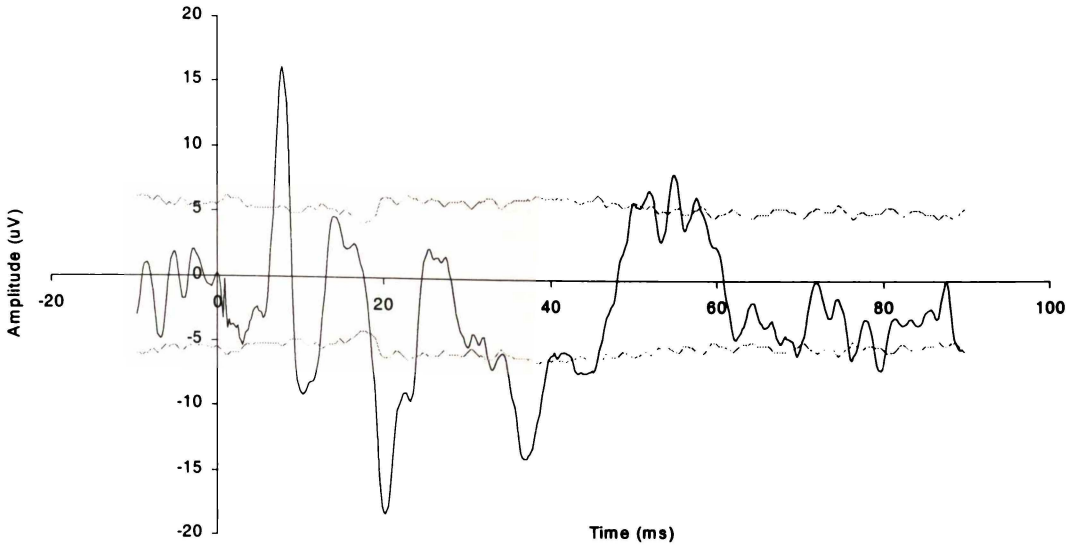


Figure 5.2 – Evoked Response Waveform with Significance Limits

As an (extreme) example of placing a significance level on the amplitude difference between two features consider the peaks present at approximately 8ms and 20ms. The amplitude difference between these two features is approximately 34μV (16μV + 18μV). Given a 99% (3σ) significance level of 6μV, these two features are different at approximately 17σ level of significance. There is very little probability that these features were due to the effects of noise alone.

This concludes the description of the Evoked Response acquisition and processing performed on the FRPM. For each Evoked Response waveform recorded the FRPM generates two data files, a waveform file and a run statistics file, which form the basis of all further analyses. Listing 5.2 shows a segment of one of the waveform files.

```
Evoked Response Waveform File.
Title:,< No Title >
Filename:,erdata_04.csv
Time recorded:,11:05:10 23-06-1998

Stimulus Channel:,Left
Stimulus Count:,1024
Stimulus Attempted:,1056
Stimulus Interval:,100,ms
Stimulus Intensity:,100
Stimulus Duration:,100,us
Stimulus Delay:,10,ms
Sample Channel:,Left
Sample Duration:,100,ms
Sample Rate:,5000,Hz
Sample Count:,500

Time (ms),Average (uV),+3 Sigma (uV),-3 Sigma (uV),Minimum
(uV),Maximum (uV)
-10.0,0.02,1.89,-1.89,-125.38,92.65
-9.8,0.04,1.89,-1.89,-127.01,94.70
-9.6,0.00,1.88,-1.88,-123.33,93.47
...
89.4,0.34,1.82,-1.82,-91.43,85.09
89.6,0.32,1.83,-1.83,-96.13,78.95
89.8,0.38,1.85,-1.85,-100.83,82.02
```

Listing 5.2 – Evoked Response Recording Runs Statistics File

The waveform file comprises of header information, a time-base, the recorded waveform, the 99% significance levels, and the minimum and maximum recorded values for each sample.

The run statistics file was discussed in Section 5.1.2.

5.2 Post-Acquisition Data Processing

Once the waveform has been recorded and transferred from the FRPM to a host computer, further processing is required to extract the constituent features of each waveform. These extracted features can then be used as a basis for statistical evaluation of similarities or differences between waveforms.

Two different approaches to feature extraction were used. The first approach was to extract only the latency (time from stimulus presentation) and the amplitude of significant peaks from the recorded waveforms. These times and amplitudes could then be used to investigate changes in the latencies and amplitudes and to match corresponding peaks in different waveforms. The second approach involved decomposing a set of waveforms into a new orthogonal set of component waveforms, a technique known as Principle Component Analysis (PCA). Using this technique the overall shape of the waveform can be investigated, allowing development of a measure of similarity between waveforms.

5.3 Automated Feature Extraction

Traditionally the analysis of Evoked Response waveforms focuses on the identification of peaks (both positive and negative) in the recorded waveform. Although this is a relatively trivial task for a human operator, the definition of an automatic algorithm for extraction of significant peaks which produces results similar to the intuitive choices of a human operator can pose considerable challenge. This challenge arises largely due the fact that the significance of a peak is not only due to its height, but also to the relative significance of nearby peaks.

This section describes an original peak extraction algorithm derived for use with the stress-related Evoked Response recordings.

A broad outline of a peak extraction algorithm will be now defined. The steps the algorithm performs are:

1. Identification of peaks in the waveform
2. Extraction of all features in the waveform
3. Removal of insignificant features from identified features

4. Output of feature parameters for statistical analysis

5.3.1 Identification of Peaks

If only significant peaks are to be extracted from the waveform some level of significance must first be established to allow the significant peaks to be separated from insignificant peaks. The raw sample files produced by the FRPM contain an estimate of the residual noise at each sample point in the waveform. These estimates assume a normal distribution of noise at each sample and are presented as 99% confidence levels. The average of all the noise estimates is used to establish the level of significance used for the peak detection process.

Consider a recorded Evoked Response waveform where the amplitudes x_i have been sampled at times t_i for i=1 to N samples. Let s_i be the estimated residual noise level for each sample. Each s_i is assumed to have zero mean and similar distribution. A global level of significance σ is defined for the entire waveform by averaging the residual noise values at each sampling time instant:

$$\sigma = \frac{1}{N}\sum_{i=1}^{N} s_i \tag{5.5}$$

By definition a peak has a slope, or derivative, with respect to time of zero. Calculating the derivative of the Evoked Response waveform and searching for any zero-crossing points forms the basis of the peak detection algorithm.

The derivative of the recorded waveform is calculated as:

$$d_i = \begin{cases} \dfrac{x_i - x_{i-1}}{t_i - t_{i-1}}, & i > 1 \\ 0, & i = 1 \end{cases} \tag{5.6}$$

The resulting derivative is then smoothed by applying a non-linear filter:

$$y_i = \begin{cases} F(d_1, d_2, d_3), & i = 1 \\ F(d_1, d_2, d_3, d_4), & i = 2 \\ F(d_{i-2}, d_{i-1}, d_i, d_{i+1}, d_{i+2}), & 2 < i < N-1 \\ F(d_{N-3}, d_{N-2}, d_{N-1}, d_N), & i = N-1 \\ F(d_{N-2}, d_{N-1}, d_N), & i = N \end{cases} \tag{5.7}$$

$$F(a_1, a_2, \ldots, a_C) = \frac{1}{C-2}\left(\sum_{j=1}^{C} a_j - Min(a_j) - Max(a_j) \right)$$

Where C is the number of arguments passed to F (either 3, 4, or 5). This filter produces the mean of a group of samples and removes the extreme data from the group[105].

There are four possible cases for the detection of a zero-crossing point in discrete data sequence as shown in Figure 5.3.

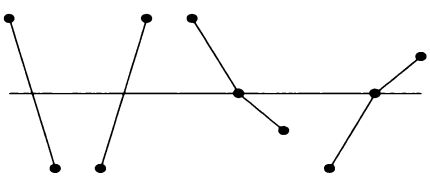


Figure 5.3 – The Four Possible Cases of a Zero-Crossing

The first two cases can be described as a sign change between two consecutive samples and the final two cases as a sign change between the neighboring samples of a central sample which must be zero. Therefore, the position of the peaks can be found by searching for the zero-crossing points z_i using the criteria:

$$z_i = \begin{cases} False, & i = 1, N \\ [\text{sgn}(y_i) \neq \text{sgn}(y_{i-1})] \vee \ldots & \\ \qquad [(y_i = 0) \wedge (\text{sgn}(y_{i-1}) \neq \text{sgn}(y_{i+1}))], & 1 < i < N \end{cases} \tag{5.8}$$

z_i will be true if a zero-crossing point has been found at sample i.

Using z_i, the amplitudes and positions of possible peaks in the recorded waveform can be extracted from the original waveform x_i and the corresponding time scale t_i. This sorting procedure generates two new variables p_k, the amplitude of peak k, and l_k, the latency of peak k

from time t=0 where k=1...M. M is the number of potential peaks found (i.e. The number of true z_i values).

The variables p_k, l_k, and M are used in the search for significant peaks in the large number of non-significant peaks found by the zero-crossing algorithm.

Figure 5.4 shows the performance of the peak detection algorithm on a relatively noisy waveform. Detected peaks are indicated by the dots on the waveform. The algorithm appears to have detected all possible peak positions in the waveform. However, many of these peaks have no significance and must be removed by further processing.

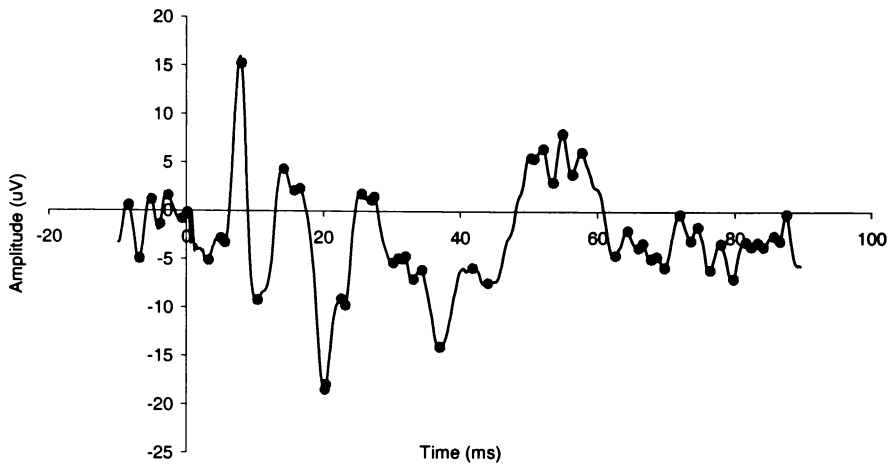


Figure 5.4 – Performance of the Peak Detection Algorithm

5.3.2 Extraction of Features

Once a list of possible peak positions are available the process of eliminating insignificant peaks can begin. When a waveform is scanned by the human eye the eye tends to identify the position of significant peaks in two ways: peaks that look significant on their own (a local feature), and groups of smaller peaks that if merged would create a single, larger, significant peak (a regional feature). These two identification processes can be termed respectively as a local search and a regional search.

From the previous steps there is now established a significance threshold σ and the variables p_k and l_k which describe the amplitude and latency of each of the M peaks found using the zero-crossing algorithm. These variables can be used to eliminate insignificant peaks.

Any significant peak has two edges so the algorithm begins by scanning p_k for significant edges. An edge can be either leading or trailing and is considered significant if the amplitude of the edge is greater than the significance threshold σ. Two new Boolean variables, $leading_k$ and $trailing_k$, are defined as:

$$leading_k = \begin{cases} False, & k = 1 \\ |p_k - p_{k-1}| > \sigma, & 1 < k \le M \end{cases}$$
$$trailing_k = \begin{cases} |p_{k+1} - p_k| > \sigma, & 1 \le k < M \\ False, & k = M \end{cases} \quad (5.9)$$

Hence $leading_k$ will be true if a significant leading edge was found at peak k while $trailing_k$ will be true if a significant falling edge was found at peak k.

Now a local search can be performed to locate the groups of peaks that will be considered in the regional search. Define another two Boolean variables $full_k$ and $half_k$. The variable $full_k$ will be true if the peak k is significant in the local sense. The variable $half_k$ will be true if the peak k marks either the start or end of a group of peaks to be considered for the regional search. The variables are defined as:

$$full_k = leading_k \wedge trailing_k$$
$$half_k = (leading_k \vee trailing_k) \wedge \neg full_k = leading_k \forall trailing_k \quad (5.10)$$

The peaks picked out by these two variables can be seen in Figure 5.5. It can be seen that the $full_k$ variable has performed the local search function while the $half_k$ variable has correctly found the edges of the groups for the regional search function.

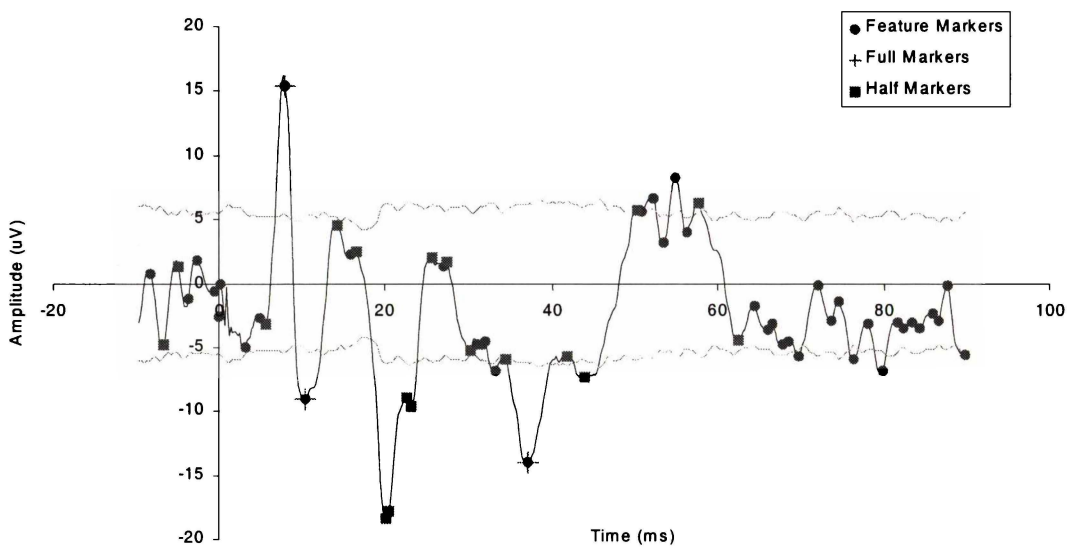


Figure 5.5 – Features Used for Finding Significant Peaks

There is now enough information to classify every peak. The peaks are classified into four categories *Feature*, *FeatureStart*, *InsideFeature*, and *FeatureEnd* according to the relation:

$$c_1 = InsideFeature$$

$$c_{i>1} = \begin{cases} Feature, & full_i \\ FeatureStart, & half_i \wedge [(c_{i-1} = Feature) \vee (c_{i-1} = FeatureEnd)] \\ FeatureEnd, & half_i \wedge [(c_{i-1} = FeatureStart) \vee (c_{i-1} = InsideFeature)] \\ InsideFeature, & Otherwise \end{cases} \quad (5.11)$$

where c_i is the category for each peak.

Based on these categories each peak is assigned to a feature, a feature being either an individual peak classified as a *Feature* , or all those peaks between a *FeatureStart* and the next *FeatureEnd*. The next step in the algorithm is to combine the peaks in each feature to complete the search process.

5.3.3 Removal of Insignificant Features

For each feature identified, the mean and standard deviation of both the amplitude and latency of the peaks in the feature is calculated. For local features the mean amplitude and latency are defined as the original amplitude and latency of the peak. The standard deviation of the amplitude is defined as the significance threshold σ and the standard deviation of the latency as zero.

Regional features combine the statistics of a number of features. The mean amplitude and latency are defined respectively as the mean amplitude and latency over every peak in the feature while the standard deviation of the amplitude is defined as the significance threshold σ divided by the square-root of the number of peaks in the feature. The standard deviation of the latency is the sample standard deviation of the latencies of every peak in the feature.

This mean amplitude and latency of each feature marks the final position of the feature and completes the search process.

As an example of the algorithm consider Table 5.1. It contains the peaks that were found by the zero-crossing algorithm that will contribute to the features at 8ms, 10.4ms, and around 50ms in Figure 5.1. The significance threshold σ is 5.56μV. This table in turn gets transformed into Table 5.2 which gives the final positions of the features found in the waveform.

Time (ms)	Amplitude (μV)	Lead	Trail	Full	Half	Classification	Feature No.
...							
5.6	-3.31	False	True	False	True	*GroupEnd*	2
8.0	15.23	True	True	True	False	*Local*	3
10.4	-9.24	True	True	True	False	*Local*	4
14.2	4.33	True	False	False	True	*GroupStart*	5
...							
44.0	-7.5	False	True	False	True	*GroupEnd*	11
50.4	5.57	True	False	False	True	*GroupStart*	12
50.8	5.45	False	False	False	False	*InsideGroup*	12
52.2	6.5	False	False	False	False	*InsideGroup*	12
53.6	2.99	False	False	False	False	*InsideGroup*	12
55.0	8.09	False	False	False	False	*InsideGroup*	12
56.4	3.83	False	False	False	False	*InsideGroup*	12
57.8	6.15	False	True	False	True	*GroupEnd*	12
62.6	-4.61	True	False	False	True	*GroupStart*	13
...							

Table 5.1 – Example of Local and Regional Peak Search

Feature No.	Time (ms)	Amplitude (μV)	Time StdDev (ms)	Amplitude StdDev (μV)
...				
3	8	15.23	0	5.56
4	10.4	-9.24	0	5.56
...				
12	53.74	5.51	2.60	2.10
...				

Table 5.2 – Results of Peak Search

Figure 5.6 shows the performance of this algorithm on the waveform of Figure 5.5. It shows that all significant features have been identified although there are several detected features that a human operator may not consider significant.

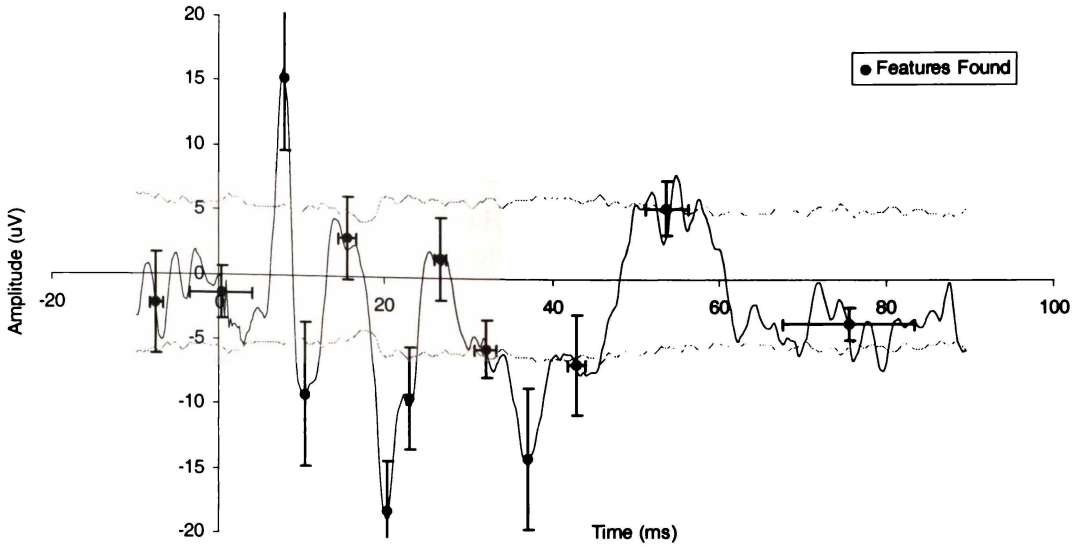


Figure 5.6 – Peaks Found by the Local and Regional Search

Of the 13 features detected in Figure 5.6, six can be considered inconsequential and should be eliminated. This elimination can be performed by a relatively simple pattern search based on two rules:

1. Any two consecutive positive peaks should have a negative peak between them and any two consecutive negative peaks should have a positive peak between them
2. The difference in amplitude between any peak and its two closest neighbors should be greater than the level of significance σ

If two new variables a_i and t_i are respectively defined as the amplitude and latency of feature i, and N as the number of features found in the previous step these two rules can be formally defined.

The first rule performs the task of removing inflections from the side of a feature and can be written as:

$$
\begin{aligned}
&\text{For i} = 2 \text{ To N} - 1 \\
&\qquad \text{BackwardSlope} = \frac{a_i - a_{i-1}}{t_i - t_{i-1}} \\
&\qquad \text{ForwardSlope} = \frac{a_i - a_{i+1}}{t_i - t_{i+1}} \\
&\qquad \text{If Sgn(BackwardSlope)} = \text{Sgn(ForwardSlope) Then} \\
&\qquad\qquad \text{MarkFeatureDiscarded(i)} \\
&\qquad \text{End If} \\
&\text{Next i}
\end{aligned}
\tag{5.12}
$$

Note that the first (i=1) and last (i=N) features are left unmodified.

The second rule performs the task of discarding a peak if it is not significantly different from its neighbors. This can occur as a result of the taking the average amplitude of peaks within a feature. This rule can be written as:

```
If |a_1 - a_N| < σ Then
        MarkFeatureDiscarded(1)
        MarkFeatureDiscarded(N)
End If

For i = 2 to N - 1

        For j = i - 1 To 1 step - 1
                If ¬IsFeatureDiscarded(j) Then
                        If |a_i - a_j| < σ Then
                                MarkFeatureDiscarded(i)
                        End If
                        Exit For
                End If
        Next j

        For j = i + 1 To N
                If ¬IsFeatureDiscarded(j) Then
                        If |a_i - a_j| < σ Then
                                MarkFeatureDiscarded(i)
                        End If
                        Exit For
                End If
        Next j

Next i
```

(5.13)

Applied consecutively, these two filters remove the inconsequential features.

5.3.4 Output of Feature Parameters

Given the waveform in Figure 5.2 and performing the zero-crossing peak detection, the local and regional feature searches, and the two inconsequential feature removal algorithms the final result obtained is shown in Figure 5.7. A tabular summary of the features found is presented in Table 5.3.

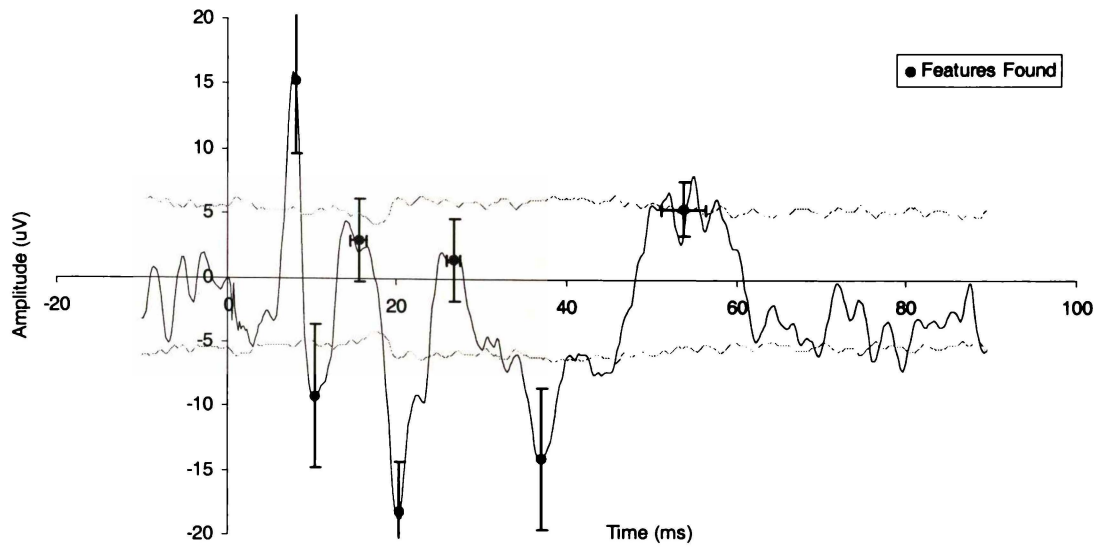


Figure 5.7 – Final Extracted Evoked Response Features

Feature No.	Time (ms)	Amplitude (μV)	Time σ (ms)	Amplitude σ (μV)
1	8.00	15.41	0	5.56
2	10.40	-9.06	0	5.56
3	15.53	3.09	1.00	3.21
4	20.30	-18.10	0.10	3.93
5	26.67	1.63	0.77	3.21
6	37.00	-13.95	0	5.56
7	53.74	5.69	2.60	2.10

Table 5.3 – Final Extracted Evoked Response Features

Figure 5.7 and Table 5.3 show that the presented algorithms perform well in automating the extraction of significant features from an Evoked Response waveform in a manner similar to that in which a human operator might perform the task.

5.4 Principle Component Analysis

In addition to the identification of features of significance in the Evoked Response waveforms, it was considered desirable to establish a method for comparing waveforms directly, based on their shape. It was hoped that establishing such a method could provide a means to classify responses both within and between animals, as well as providing the ability to look for changes in the overall shape of the waveform without the need to focus on particular individual features.

There are many methods for decomposing, or transforming, waveforms to allow comparison on the basis of shape. Examples in the literature include Principal Components Analysis[106], the Fourier transform[107], and the wavelet transform[108]. Because it was hoped to identify the features of the original waveform that were causing any changes, it was decided to use the time-domain technique of Principal Components Analysis.

Given a data set with p numeric variables, p principle components can be computed. The principle components are linear combinations of the original variables and are equal to the eigenvectors of the correlation matrix of the original data set. The principle components are sorted by descending order of their corresponding eigenvalues, which are equal to the variances of the components. By solving a simple linear equation the coefficients for reconstructing the original data set from a linear combination of the principal components can be calculated.

Principal components have a number of useful properties[109], some of which are:

1. The eigenvectors are orthogonal, so the principal components represent jointly perpendicular directions through the space of original variables
2. The first principal component has the largest variance of any normalised linear combination of the observed variables. The last principal component has the smallest variance of any normalised combination of the observed variables
3. The first j principal components give a least-squares solution to the model:

   $$\mathbf{Y} = \mathbf{XB} + \mathbf{E}$$

   Where $\mathbf{Y}$ is an $n \times p$ matrix of the centered observed variables; $\mathbf{X}$ is the $n \times j$ matrix of coefficients on the first j principal components; $\mathbf{B}$ is the $j \times p$ matrix of principal components; $\mathbf{E}$ is an $n \times p$ matrix of residuals. Principal components analysis effectively minimizes the trace ($\mathbf{E'E}$), the sum of all squared elements in $\mathbf{E}$.

   This can also be stated as: the first j principal components are the best linear predictors of the original variables among all possible sets of j variables, although any nonsingular linear transformation of the first j principal components would provide equally good prediction
4. In geometric terms, the j-dimensional linear subspace spanned by the first j principal components gives the best possible fit to the data points as measured by the sum of squared perpendicular distances from each data point to the subspace

Because of these properties, all of the Principal Components pass through the center of gravity of the original variables: i.e. they pass through the point whose coordinates are the means of the original variables. Hence the means of the data are typically removed before Principal Components Analysis to force the intersection, or origin, of the Principal Components to zero.

In the case of the Evoked Response waveforms, Principal Components Analysis allows the transformation of a set of p waveforms into a new, orthogonal, set of p waveforms, the principal components. Each original waveform can be reconstructed perfectly by a linear combination of these components. Alternatively, by keeping only the first j components (the **B** matrix from above) the original waveforms can be reconstructed with the least possible error (in the least-squares sense) of any j components. The use of PCA for analysis of Evoked Response waveforms was first described by Glaser and Ruchkin[110] and Picton *et al.*[111] give a good review.

In terms of describing each original waveform for shape comparisons, the components can be discarded and the component coefficients (the **X** matrix from above) used as the sole basis for comparison. In effect these coefficients form a 'fingerprint' for each of the original waveforms.

The principal components analysis allows the recorded data to be reduced by a significant amount. For example, given 12 waveforms, each comprising 500 samples, a description of these waveforms can be obtained using the component coefficients from the PCA. This description comprises only 144 values, a 41:1 reduction in data. If it is found that the first four components allow reconstruction of any original waveform with only 1% error, for example, this description can be reduced to only 48 values. This would give a 125:1 reduction in data with only minimal loss of descriptive information. Such a reduction eases later statistical analysis considerably by removing redundant information and noise present in the original waveforms.

To analyze the Evoked Response waveforms, the recorded waveforms for each animal were grouped together into a single $n \times p$ matrix **Y**, where n is the number of waveforms (typically 48, 12 waveforms from each of 4 experiments) and p is the number of samples in each waveform (typically 500 samples). The mean waveform of these n waveforms was then subtracted from each row in the matrix to emphasize the differences between the waveforms. The $p \times p$ covariance matrix of this difference matrix is formed and an eigenvector decomposition performed to extract the n principal components. The results of this decomposition are then sorted by decreasing value of the n related eigenvalues to obtain the vector **B**.

After the components have been sorted, the $n \times n$ matrix **X** of coefficients required to reconstruct the original waveforms from the components can be found by solving the equation $\mathbf{X} = \mathbf{YB}^{-1}$. The original waveforms can be reconstructed using the equation $\mathbf{Y} = \mathbf{XB} + \mathbf{E}$ where **E** is an $n \times$p matrix of residuals generated by truncating the number of components used to reconstruct the waveforms. If the full complement of n components is used to reconstruct each waveform then **E** will be zero, otherwise it will be non-zero.

For the Evoked Response waveform analysis only the first few components were used for reconstruction. The corresponding coefficients of this limited number of components was used

for the later statistical analysis of the waveforms. By limiting the number of coefficients used in the statistical analysis, the analysis would only be applied to those components that accounted for the significant features in the original waveforms.

Figure 5.8 combines two plots: the upper plot shows the cumulative amount of variance accounted for by using a varying number of components for reconstruction, while the lower plot shows how much additional variance each additional component adds to the total. This figure is based on the variance associated with 48 components calculated from a PCA on each of 14 animals (48 waveforms per animal). The single darker plot is the average of these 14 distributions. When the 100% mark is reached, the original waveforms have been reconstructed perfectly. By using this plot a reasonable minimum number of component coefficients to use in the statistical analysis can be found.

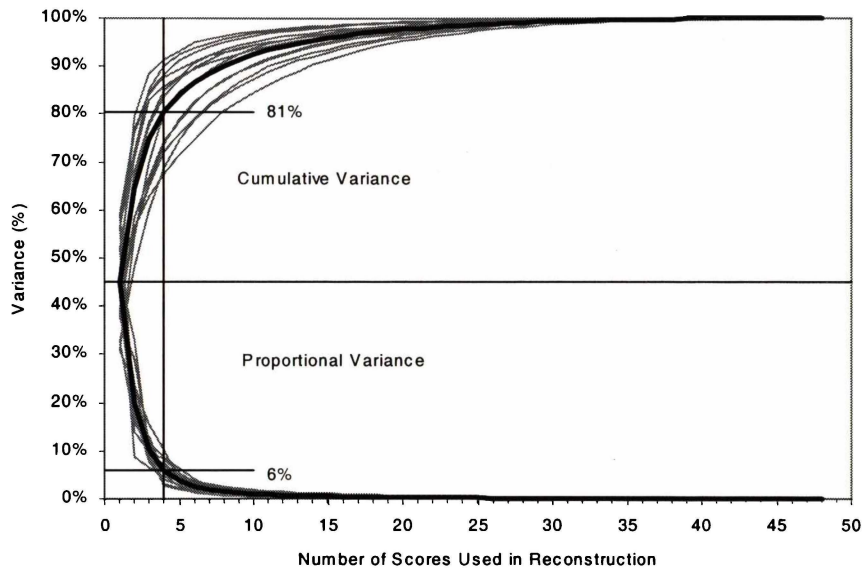


Figure 5.8 – Reconstruction Variance vs. Number of Components Used

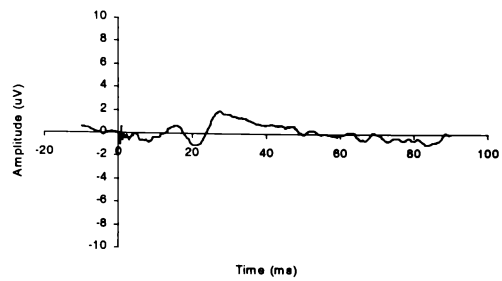
Table 5.4 lists the number of components required to meet various benchmarks and shows the quickly diminishing returns of using more than just a few components.

Variance accounted for	Number of components used
44.9%	1
68%	3
80%	4
90%	8
95%	14
99%	27
100%	48

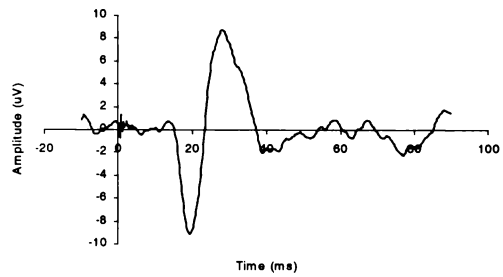
Table 5.4 – Reconstruction Variance vs. Number of Components Used

Based on this table it was decided to keep only the first four of the original 48 components from the PCA for use in the statistical analysis of the waveforms.

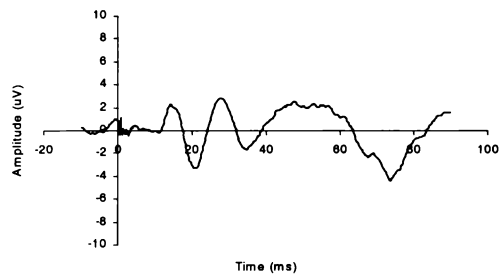
As an example of using PCA to analyze a series of Evoked Response waveforms consider the three Evoked Response shown in Figure 5.9. These three waveforms were taken from a total set of 12 waveforms taken from the same animal but under control, stressed, and recovery conditions.



Waveform 1



Waveform 2



Waveform 3

Figure 5.9 – Three Example Evoked Response Waveforms

The mean waveform of these three waveforms is shown in Figure 5.10 with the difference waveforms after subtracting the mean waveform from each the three recorded waveforms shown in Figure 5.11. Subtracting the mean waveform emphasizes the differences between the example waveforms.

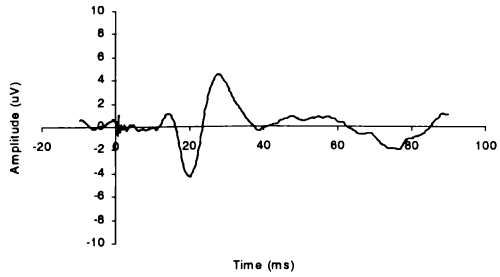
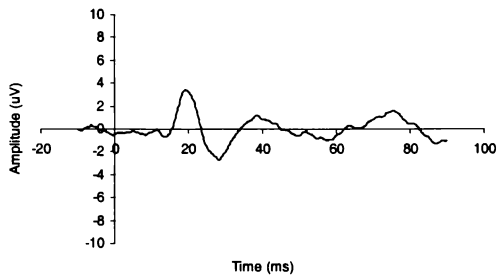
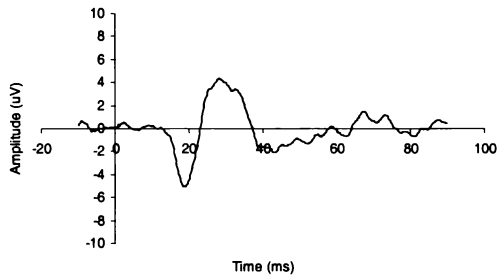


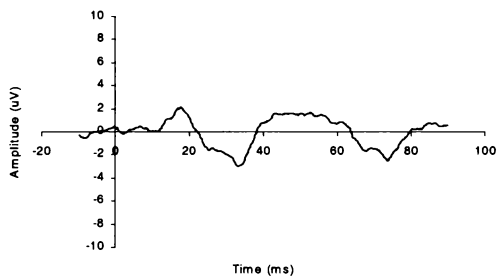
Figure 5.10 – The Mean of the Three Example Waveforms



Difference Waveform 1



Difference Waveform 2



Difference Waveform 3

Figure 5.11 – The Three Difference Waveforms

The first four principal components (out of a total of 12) found by the PCA are shown in Figure 5.12. For this particular animal, Figure 5.13 shows the proportional and cumulative variance accounted for by using a specified number of components for reconstruction. Significant features

of the difference waveforms can be seen in these components while the plot of variance shows a steady decline in useful information contained in the later components. Using only the first four components accounts for 85% of the total variance contained in the original set of 12 waveforms.

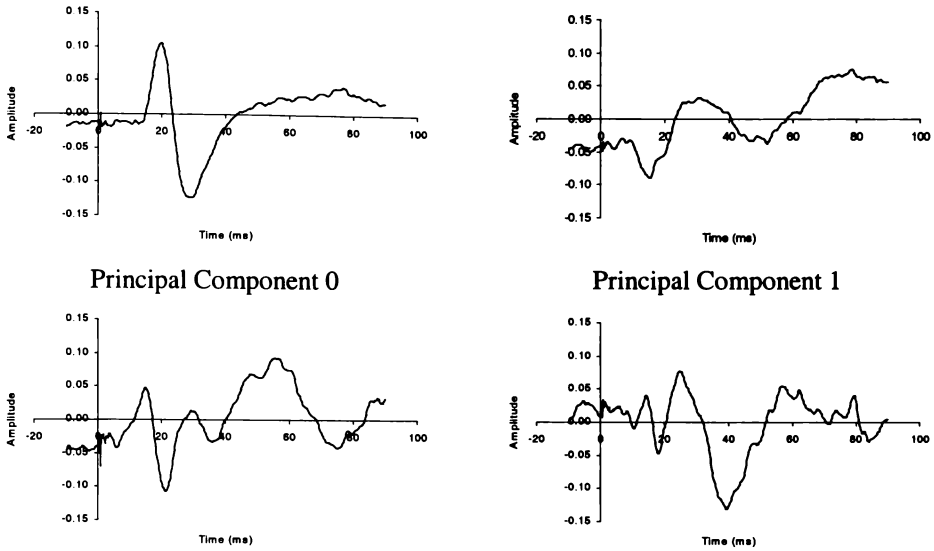


Figure 5.12 – The First Four Components Produced by the PCA

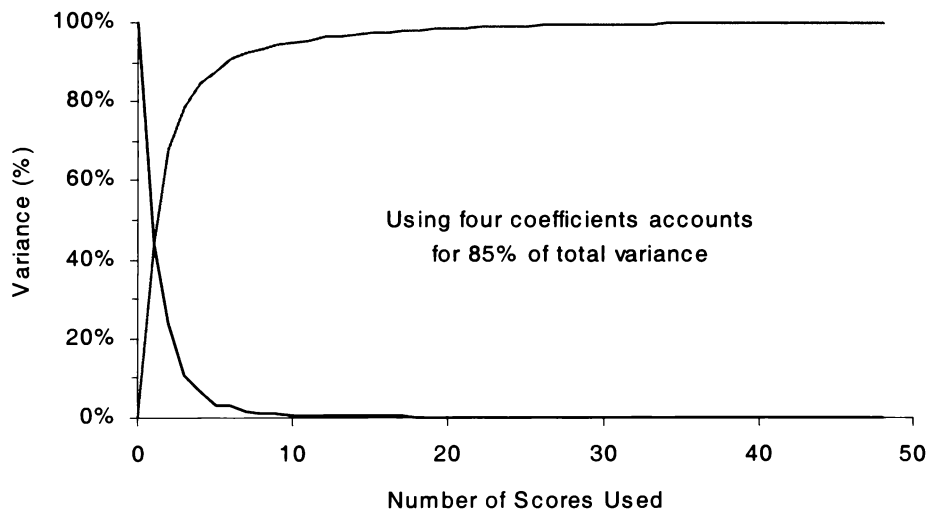
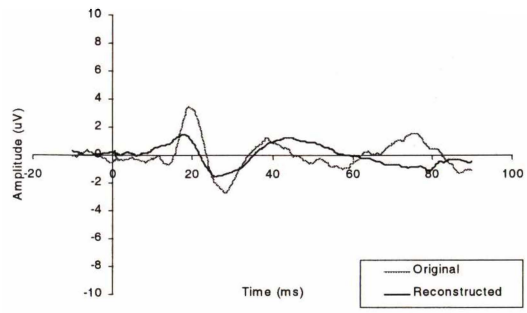


Figure 5.13 – The Proportional Variance of each Component

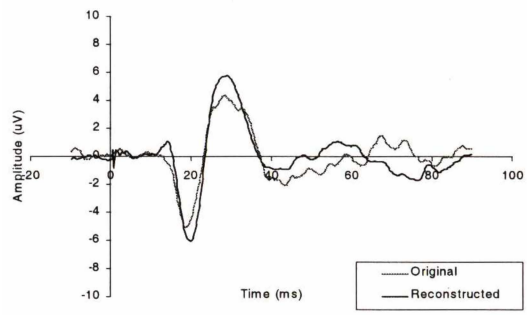
The four PCA coefficients required to provide the best-fit reconstruction of the difference waveforms using the Principal Components generated by the PCA are presented in Table 5.5. Keeping the first four of these coefficients and using them to reconstruct the difference waveforms produces the waveforms in Figure 5.14.

	c0	c1	c2	C3
Difference Waveform 1	5.61	-12.14	3.88	-9.09
Difference Waveform 2	-39.96	6.20	15.08	12.98
Difference Waveform 3	3.16	-25.90	27.80	-5.42

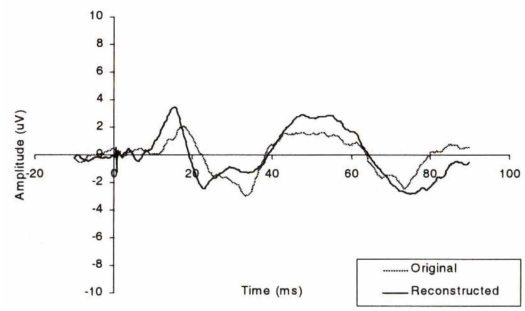
Table 5.5 – The Four PCA Coefficients used to Reconstruct the Difference Waveforms



Difference Waveform 1 - Reconstructed



Difference Waveform 2 - Reconstructed



Difference Waveform 3 - Reconstructed

Figure 5.14 – The Difference Waveforms Reconstructed by using a subset of the Components and Coefficients Generated by the PCA

These reconstructed difference waveforms compare well with the original difference waveforms. Although there are noticeable differences the broad features of each waveform have been preserved.

Adding the mean waveform of Figure 5.10 to each of these three reconstructed difference waveforms allows the original recorded waveforms to be reconstructed, as shown in Figure 5.15.

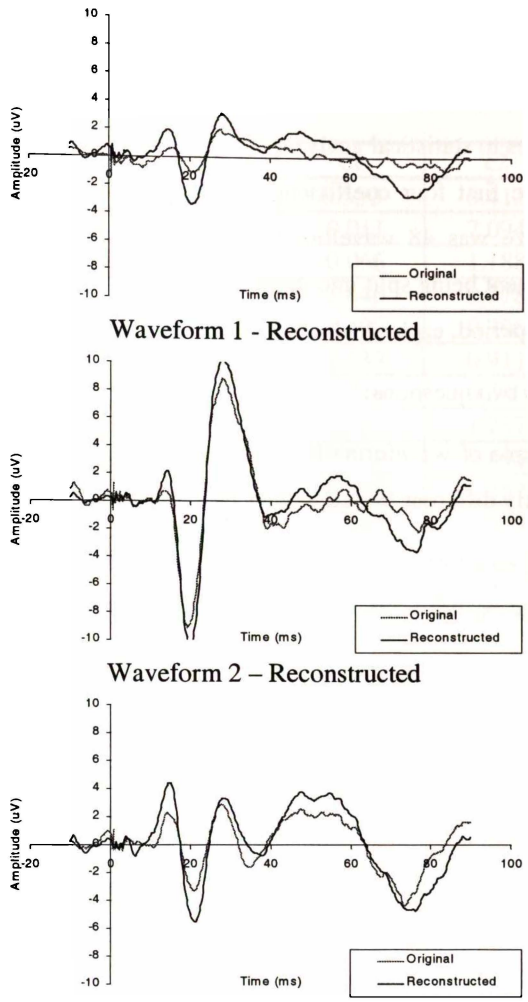


Figure 5.15 – The Reconstructed Recorded Waveforms

Again, the reconstructed waveforms compare well with the original waveforms with any difference being attributable to the 15% of variance unaccounted for due to truncating the number of components used in the reconstruction process.

By performing the PCA on the recorded Evoked Response waveforms and obtaining the principal components required to reconstruct the original waveforms, each waveform can be replaced with a small number of coefficients, in this case four, that provide a description of the features of each waveform. In the case of the example Evoked Response waveforms, each originally described by 500 individual samples, this is a data reduction of 125:1.

The four coefficients obtained as a description of each waveform can now be used for statistical analysis and to create a classifier for waveforms based on their features.

5.5 Statistical Analysis of Waveforms

Once the recorded Evoked Response waveforms have been transformed into a more manageable description, either by Feature Detection or PCA, groups of waveforms can be subjected to statistical analysis to assess the level of similarity of different waveforms or to classify individual waveforms into a series of categories.

Two different approaches to statistical analysis of the recorded Evoked Response waveforms were taken, both utilizing the first four coefficients obtained by a PCA on a group of waveforms. Typically the group size was 48 waveforms consisting of 12 waveforms from each of four experiments, an experiment being split into three consecutive periods: a Control period, a Stressor period, and a Recovery period, each of which consisted of four waveforms.

This arrangement raises two questions:

1. Is the sample of waveforms from each of the three periods within each experiment significantly different from the other two periods?
2. Given the waveforms in each of the three periods, is it possible to form a classifier that can successfully match a waveform to its period?

The first question can be answered by performing an Analysis of Variance on the component coefficients from the PCA. The second question can be answered by using Discriminant Analysis to form a classifier for the waveforms, again based on the coefficients obtained by the PCA.

Note that there is a difficulty in applying either of these techniques to time-series data: both techniques assume that the samples are all statistically independent samples of a random process. In this case the data exhibits serial dependency and hence from a strict statistical point of view neither ANOVA or Discriminant Analysis is suitable. However, in view that no suitable statistical methods could be found as a replacement, both of these analyses are used with the observation that the statistical results from either of these methods can only be considered as exploratory.

5.5.1 Analysis of Variance

One-way Analysis Of Variance (ANOVA) is a statistical method for comparing several population means. By taking a random sample for each population and applying an ANOVA analysis it is possible to test the null hypothesis that the population means are equal[112].

Given a group of 12 waveforms from an experiment, each described by four coefficients, an ANOVA is performed to determine if the three periods that make up the experiment are distinct from each other. This process requires four separate ANOVA to be performed: one for each of the four coefficients. Since the first component from the PCA accounts for the most variance in the recorded waveforms, the ANOVA performed on the first coefficient is the most important.

As an example consider the set of coefficients in Table 5.6 obtained from a group of waveforms from an experiment. The first four waveforms comprise the Control period, the middle four waveforms the Stressor period, and the final four waveforms the Recovery period. Note that each column has been normalised to have zero mean and a unit standard deviation. This has no effect on the ANOVA since the ANOVA is performed only within a single column.

Waveform	Category	c0	c1	c2	c3
1	Control	-1.725	0.550	3.413	0.168
2	Control	-1.416	-0.011	2.094	0.172
3	Control	-0.866	-0.066	1.188	0.772
4	Control	2.137	-2.646	0.778	-3.566
5	Stressor	2.559	-2.192	0.796	0.918
6	Stressor	1.268	-1.233	0.911	1.557
7	Stressor	0.694	-0.714	0.340	2.310
8	Stressor	2.700	-1.712	1.257	1.642
9	Recovery	-0.322	-0.409	0.645	-0.699
10	Recovery	-0.624	-0.418	1.352	-1.460
11	Recovery	-1.594	-0.149	2.379	-1.599
12	Recovery	-0.229	-0.919	-0.124	0.364

Table 5.6 – PCA Coefficients from an Evoked Response Experiment

Figure 5.16 shows the result of plotting c0 on a category basis and suggests there is some difference between the categories. Note the outlier in the Control category: it is probable that this reflects a departure from the specified experimental methodology by accidentally exposing the animal to the stressor before the Control period was over.

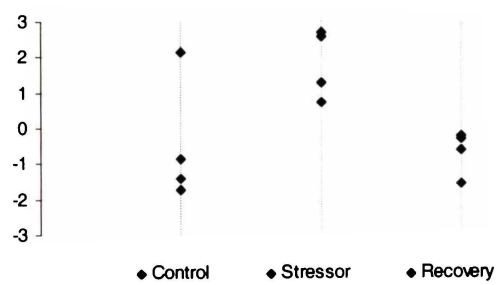


Figure 5.16 – Plot of PCA Coefficient c0 from an Evoked Response Experiment

The ANOVA will assess how significant these differences are. Table 5.7 lists the results from an ANOVA performed on c0 from Table 5.6 and Figure 5.16 on the basis of links between the three periods. The ANOVA shows that the Stressor period is different from both the Control and Recovery periods with at least 95% probability. There is only 20% probability that the Control and Recovery periods are different.

Categories	Difference	Student's t	Pr. > t'	Significant?
Control ~ Stressor	-2.27	-2.63	0.028	Yes
Control ~ Recovery	0.22	0.26	0.801	No
Stressor ~ Recovery	2.50	2.89	0.018	Yes

Table 5.7 – Results of an ANOVA on PCA Coefficient c0

The XLStat[113] add-in for Microsoft® Excel™ was used to perform the ANOVA analysis.

5.5.2 Discriminant Analysis

Given a group of categorical observations, each described by one or more quantitative variables, Discriminant Analysis allows us to find any of the following[114]:

- A mathematical rule, or discriminant function, for predicting which category an observation belongs, based only on knowledge of the quantitative variables of that observation
- A set of linear combinations of the quantitative variables that best reveals the differences among the categories

Of the many variants of discriminant analysis, Classificatory Discriminant Analysis is used to classify observations into several groups on the basis of one or more quantitative variables. This was the technique used to analyze the Evoked Response waveforms.

Typically two groups of observations are used: the first group, the training set, is used to form the discriminant function while the second group, the validation set, is used to test the performance of the discriminant function. The performance of the discriminant function is evaluated by estimating the probabilities of misclassification when the discriminant function is applied to the validation set.

However, if the total number of observations available for analysis is small, it is often difficult to split the small available number of observations into two groups without severely compromising the accuracy of the model. In this case it is better to use all of the observations as the training set and use cross-validation techniques to compare how the resulting discriminant function classifies each observation with respect to its original classification.

To form the discriminant function, the discriminant analysis transforms the quantitative variables into a new set of canonical variables, typically by a linear transformation. Given n variables and m categories, it is only possible to form a minimum of n or $(m\text{-}1)$ discriminant functions. This can be rationalized by the observation that only $(m\text{-}1)$ planes (functions) are required to separate any m categories in space. However, if $(m\text{-}1)$ exceeds the number of variables we are limited in our discrimination by only being able to generate a maximum of n discriminant functions from our n observations before all the variability in the variables is explained (this also occurs when

calculating the PCA). The worst case is therefore only one variable per category, possibly with additional empty categories left over.

Geometrically, if each observation consists of n variables, this has the effect of transforming the n-dimensional space of observations so that the most variability in the observations is projected onto the first canonical variable. The variability accounted for is removed from the observations and the space is again rotated to project the maximum amount of any remaining variability onto the second canonical variable. This continues until n canonical variables have been formed.

After transforming the variables from each observation by the equations the resulting canonical variables can be plotted against each other to give a graphical representation. Figure 5.17 shows the data from Table 5.6 which has been subjected to a discriminant analysis. Each axis is a canonical variable, four canonical variables and three categories producing two discriminant functions (the minimum of 4 or 3-1 is 2). It clearly shows that most of the variability, 97%, in the observations can be accounted for in the projection onto the first canonical variable, while the second canonical variable accounts for the remaining 3% variability. Note that between to two canonical variables all of the variability (100%) in the variables is accounted for.

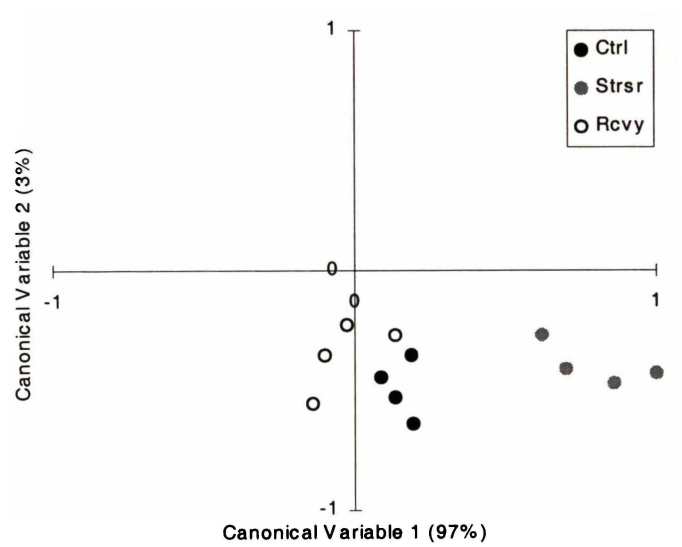


Figure 5.17 – Example Discriminant Analysis Plots on PCA Coefficients

The analysis also produces a table that indicates the probability that any category is the same as any other category. Table 5.8 shows these probabilities for the example categories. Note that there is only 7.2% probability that the Control and Stressor categories are the same and only 3.8% probability that the Stressor and Recovery categories are the same.

	Probability of no difference		
	Control	**Stressor**	**Recovery**
Control	1.0000	0.0719	0.5632
Stressor	0.0719	1.0000	0.0377
Recovery	0.5632	0.0377	1.0000

Table 5.8 – Example Discriminant Analysis Category Significance

These results differ from those in Table 5.7 because the ANOVA analysis only assesses the significance of the first coefficients while the discriminant analysis assesses all four coefficients. Geometrically, the ANOVA assesses the distance between the distributions of the different categories in a one-dimensional space consisting solely of the first coefficient. This differs from the discriminant analysis which assesses the distance between the distributions on a four-dimensional basis using all four of the coefficients.

The canonical variables can also be used to reclassify the original observations, allowing the model developed by the analysis to be tested by cross-validation. This effectively uses the same set of observations for both training and validation. Table 5.9 presents the results from this test. The probability that each observation belongs to any particular category is given, and based on this, the observations are classified, possibly into a different category than that to which they were originally assigned.

Observation	**a priori category**	**a posteriori category**	**Probability for Control**	**Probability for Stressor**	**Probability for Recovery**
Waveform 1	Control	Control	0.9757	0	0.0243
Waveform 2	Control	Control	0.8697	0	0.1303
Waveform 3	Control	Control	0.7161	0	0.2839
Waveform 4	Control	Control	0.6364	0	0.3636
Waveform 5	Stressor	Stressor	0	1	0
Waveform 6	Stressor	Stressor	0.0001	0.9999	0
Waveform 7	Stressor	Stressor	0.0007	0.9993	0
Waveform 8	Stressor	Stressor	0	1	0
Waveform 9	Recovery	Recovery	0.0544	0	0.9456
Waveform 10	Recovery	Recovery	0.0714	0	0.9286
Waveform 11	Recovery	Recovery	0.2489	0	0.7511
Waveform 12	Recovery	Recovery	0.3662	0	0.6338

Table 5.9 – Example Discriminant Analysis Reclassification Test

Note that although waveforms from the Control and Stressor categories are easily identified, the canonical variables have more trouble identifying whether the Recovery observations belong to the Control or Recovery category. This is identifiable in the plots that contain canonical variable 1 in Figure 5.17 which show the Control and Recovery categories tend to overlap.

The XLStat[113] add-in for Microsoft® Excel™ and MATLAB[115] were used to perform the Discriminant Analysis.

5.6 Extraction of Heart Rate from the Electrocardiogram

In addition to acquiring Evoked Response, electrocardiogram (ECG) recordings were made during experiments for the purpose of extracting heart-rate profiles for comparison with changes in the Evoked Response waveforms. The measurement of heart-rate was based around a Fast-Fourier Transform[116] (FFT) to extract the frequency components of the recorded ECG signal followed by a peak detection process to determine the frequency of the peak associated with the heart-rate itself.

The recorded ECG signal consisted of 30s of data sampled at 100Hz, giving 3000 samples. When transformed with the FFT this recording could be expected to give a frequency resolution of 33mHz over a range of 0 to 49.5Hz. Figure 5.18 shows a typical segment of an ECG recording

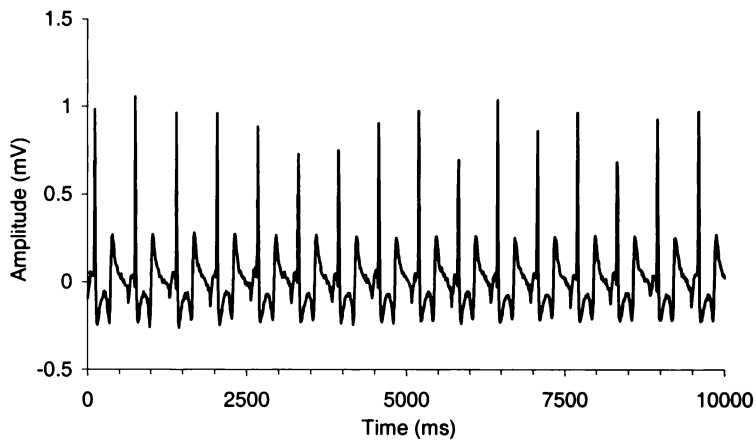


Figure 5.18 – Example Segment of an ECG Recording

The recorded samples first had the mean DC value extracted from each sample before being multiplied by a Blackman[117] window to reduce spectral leakage during the FFT. The FFT was performed over the full set of 3000 samples and the magnitude of each frequency component used to produce a spectra for finding the heartrate. Figure 5.19 shows a section of a typical result of the FFT. From this plot the heart-rate of the subject would be taken as 1.6Hz or 96 beats per minute (bpm). The largest peak at 3.2Hz is the second harmonic while the fourth harmonic is also visible at 6.4Hz. These even harmonics are due to the triangular shape of the strong T-wave component in the recorded ECG signal.

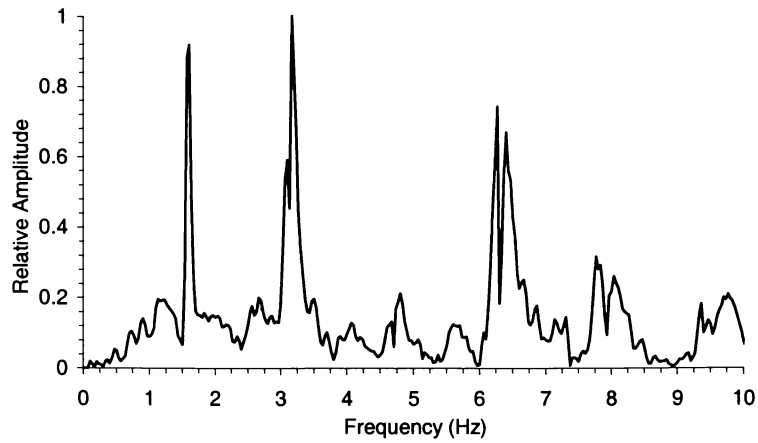


Figure 5.19 – Example Result of Fourier Transform on ECG Segment

For automated heart-rate extraction, the heartrate was defined to be the first major peak within the range of 0.5Hz to 5Hz. A major peak was defined as the having an amplitude of greater than 50% of the highest peak in the frequency range.

This process was operator-supervised as changes in heartrate during recording often produced two equal, or nearly equal, amplitude peaks in the spectra. In these cases the operator chose the frequency considered to be the best average of the two peaks based on the relative heights. This gave the estimate of the mean heartrate during the 30s recording period.

The heartrate data was used to assess the relative effect of each of the four stressors (see Chapter 7): the Control, the Distraction, the Isolation and the Fear Stressors. Figure 5.20 shows a profile of the heartrate of an animal during a one-hour experiment. Note the sudden increase heartrate as the subject is isolated at the 20min mark, followed by a steady decline until the another smaller increase marking reunion with the rest of the flock. Contrast this with the relatively steady heartrate of the animal during an experiment where no stressor was presented.

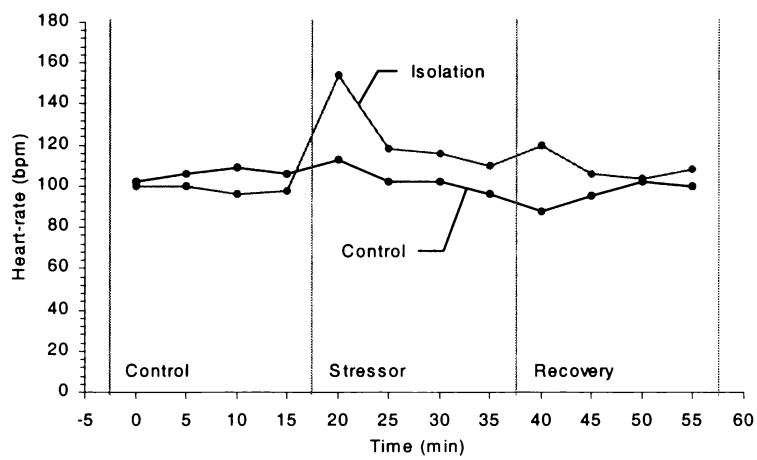


Figure 5.20 – Example Heartrate Profile Extracted by Fourier Transform Method

5.7 Summary

Two methods have been presented for the analysis of Evoked Response waveforms. The first, detection of statistically significant features in a waveform, is a rule-based technique which searches any individual waveform for any peaks that may be present, then proceeds to discard and merge individual peaks until only features that can be considered significant based on the residual noise are left. The second method, Principal Components Analysis, breaks a collection of waveforms down into a set of orthogonal component waveforms that can then be linearly combined by weighting coefficients to reconstruct the original waveforms, the component waveforms and weighting coefficients together forming a complete description of the original waveform set.

Of these two techniques, it was decided to use Principal Components Analysis (PCA) for the analysis of Auditory Evoked Response waveforms collected in this thesis. This decision was based on the consideration that PCA allows reconstruction of a highly recognizable original waveform from just a few, in this case four, coefficients and component waveforms. In this way it is a better descriptor of the overall shape of a waveform, with fewer variables, than the list of peaks generated by the peak detection method. In practice, since all waveforms are decomposed into the same set of component waveforms in PCA, it is possible to discard the component waveforms and describe and categorize the original waveforms solely based on the weighting coefficients.

To compare the waveform descriptions obtained via PCA, two methods were considered. The first used ANOVA analysis to compare the weighting coefficients, while the second used Discriminant Analysis. It was shown that Discriminant Analysis was a better choice for statistical analysis of changes between waveforms given a small set of weighting coefficients since it attempted to form a model using all of the given coefficients whereas ANOVA could only use the first coefficient. For ANOVA, any useful information in latter coefficients was ignored.

Based on these observations, the overall method for statistical analysis of the Evoked Response waveforms was Principal Components Analysis on a group of original waveforms, followed by Discriminant Analysis on the first four PCA coefficients of the waveforms to establish a probability for a statistically significant change in the original waveforms between experimental periods.

For the analysis of the heartrate recordings, Fourier analysis was used to extract the fundamental heartrate frequency from the recording, while ANOVA was used to compare changes in heartrate between experimental periods.

Chapter 6 Experimental Methodology

This chapter discusses the methodology involved with recording both the Auditory Evoked Response (AER) and the electrocardiogram (ECG) from free-ranging animals. This methodology includes electrode application and placement and, in the case of AER, the auditory stimulator. Preliminary work performed to establish recording parameters is also discussed. Finally, the general experimental protocol and the special requirements of the stress trial are described.

6.1 Evoked Response Stimulation and Recording

The recording of an Evoked Response waveform requires two main interfaces with the subject. The stimulator must ensure that a stimulus of adequate intensity to evoke a response is presented to the subject, while the electrodes must be of sufficient quality to ensure the small signal evoked by the stimulus can be recorded without undue noise or artifact.

6.1.1 The Evoked Response Electrodes

Needle electrodes were used instead of less-invasive Ag/AgCl disc electrodes in this study for several reasons. Firstly, it was felt that the needle electrodes would be more reliable due to a lower sensitivity to contaminants both on the skin and on the wool, and were more robust toward the physical abuse they might encounter in field use. Secondly, the use of needle electrodes allowed penetration of the thick skin and tissue covering the head of the sheep thus ensuring that the signal could be recorded from immediately above the skull itself. It should also be noted that the literature suggests that there is little, if any, signal degradation due to the use of needle electrodes over disc electrodes[118].

The Evoked Response needle electrodes are based around 0.7mm diameter, 24mm long stainless steel hypodermic syringe needles[i] attached to low-microphonic coaxial cable for connection to the low-noise pre-amplifier inputs. The low-microphonic cable minimizes the effect of cable movement and vibration causing additional noise at the amplifier inputs.

6.1.1.1 Evoked Response Electrode Construction

Figure 6.1 shows the construction of the electrodes. Construction proceeded as follows:

1. About 60mm of outer insulation is stripped from the co-axial cable and the exposed shielding layer is removed. The remaining exposed shield layer is protected with a 5mm length of "heat shrink" tubing
2. The inner core is stripped back to 5mm from any remaining shielding layer and is twisted for robustness before being carefully threaded through the shaft of the hypodermic needle
3. The end of the needle is then brazed with a gas torch and solder to electrically connect the core to the needle shaft. The end of the needle is then polished to restore its sharpness
4. The cable entry end is filled with epoxy to provide strain-relief for the exposed inner core
5. A layer of "heat shrink" tubing is added to cover the cable entry, providing further strain relief.

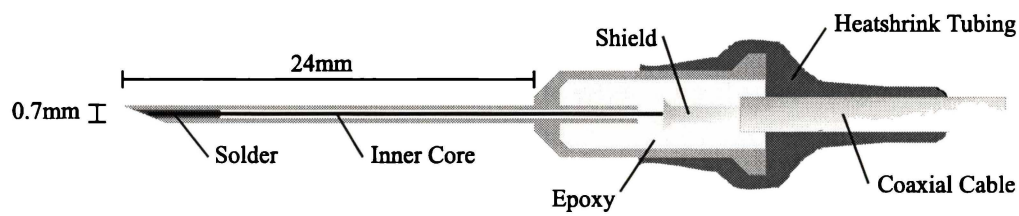


Figure 6.1 – Construction of Evoked Response Needle Electrodes

The process of brazing the core to the needle tends to place the mildly corrosive flux used up the shaft of the needle beyond the reach of the solder where it cannot be neutralized. Over time this flux will cause the core conductor to corrode and eventually break, destroying the electrical connection. However this time period is likely to be in excess of twelve months by which time the needles will have long lost their sharpness and been discarded or replaced.

For Evoked Response recordings using the FRPM, the needles are used in an assembly of three electrodes: a Reference, a Recording, and a Driven electrode. This assembly is connected to the

[i] Monoject Hypodermic Needle, Part No. 250222, Sherwood Medical, St. Louis, MO 63103, USA.

FRPM via a single DB9-type connector. This assembly also includes the wiring for the electrocardiogram electrode.

The shield conductors of each of the Reference, Recording, and ECG cables is connected to the Guard output from the pre-amplifier to minimize input common-mode capacitance in the cables which can significantly degrade the overall common-mode rejection of the system. The Driven electrode cable is not guarded since it is an output from the amplifier.

The electrode connector also brings out monitoring signals for both of the pre-amplifier channels so electrode placement and noise levels can be checked via an oscilloscope during animal preparation.

6.1.1.2 Evoked Response Electrode Application

The Evoked Response waveform was recorded at the Recording electrode with respect to the Reference electrode. The Driven electrode provided a DC current path to the FRPM unit and provided active cancellation of common-mode interference. The Recording and Reference electrodes were placed near each ear at the C_3 and C_4 positions[119] respectively. The Driven electrode was placed in the neck at the O_Z position. The major factor driving the selection of these positions was the need to avoid the significant muscles associated with the ears and the jaw of the sheep[120], factors that are not so much of a concern with human subjects. If these muscles were not avoided, the grazing and ear movements associated with ambulatory, non-restrained, animals caused sufficient electrical noise due to muscle activity and electrode movement to significantly degrade the Evoked Response recordings. This decision override the consideration that in human studies, although the primary response to a unilateral auditory stimulus occurs in the ipsilateral cortex, a substantial response is also seen in the contralateral cortex[121]. Figure 6.2 shows the positions of the Evoked Response electrodes on the animals head.

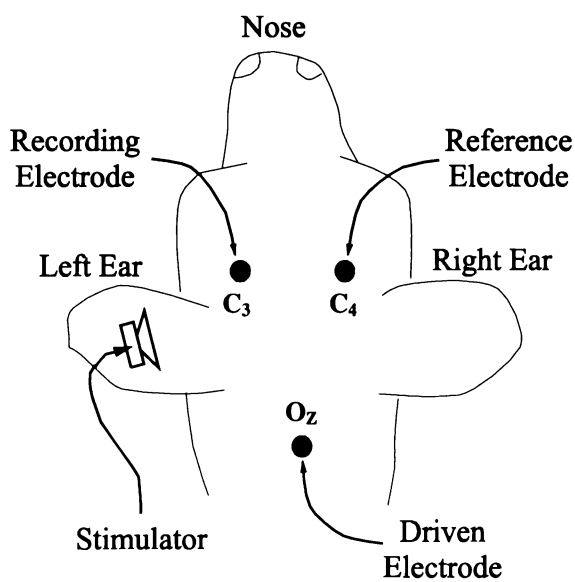


Figure 6.2 – Electrode and Stimulator Positions

Note that this electrode placement results in a bipolar recording configuration where the Recording electrode corresponds to the bipolar configuration positive electrode and the Reference electrode to the bipolar configuration negative electrode.

All Evoked Response waveforms presented in this thesis use the convention that a waveform peak in the direction of the positive y-axis corresponds to a positive potential at the Recording electrode with respect to the Reference electrode.

To insert the electrodes, the animal is first placed in a restraining cradle to prevent movement during instrumentation. The holding cradle is shown in Figure 6.3. Electrode site preparation with clipping the wool on the head of the animal to a short length (<1mm). The electrode sites are then disinfected first with a 5% concentration of Savlon[i] followed by a solution of 0.5% Hibitane[ii] in 70% ethanol. Both disinfectants are massaged into the scalp to ensure thorough coverage. The disinfected area is then dried using paper towels.

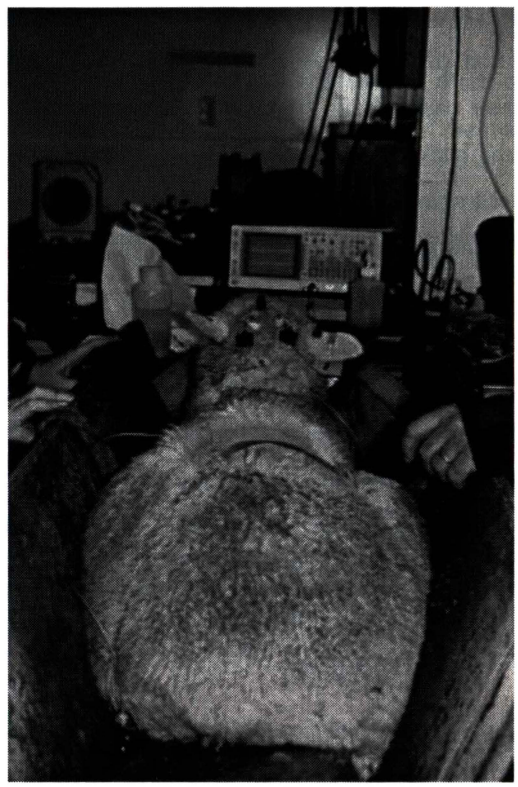


Figure 6.3 – The Sheep Holding Cradle

In initial experiments, freeze spray[iii] was used to provide local anesthesia before needle insertion. However it appeared, subjectively, that the extreme cooling action caused more distress to the animal than inserting the electrodes without anesthetic. As a result the needles are inserted without any local anesthetic.

---

[i] Savlon Antiseptic Liquid, Palmolive Skincare Laboratories, 245 George Street, Sydney, NSW, Australia.
[ii] Hibitane, ICI Pharmaceuticals, ICI Australia Operations Pty. Ltd., 1 Nicholson Street, Melbourne, Victoria 3000, Australia.
[iii] Freeze Spray, CRC Chemicals Australia Pty. Ltd., Castle Hill, Australia 2154.

Before insertion each needle electrode is polished with a fine grade of carborundum paper to ensure the needle is free of any burrs or residue glue from previous use. Needle sharpness is also inspected. The needles are then disinfected with the same Hibitane solution as above before use.

To insert the needles, the animal's head is held firmly and the full length of the needle is inserted under the skin of the animal. The needles are inserted almost parallel with the skin to avoid contact with the skull of the animal.

For Evoked Response recordings being made in the restraining cradle the above preparation is sufficient. The electrode cables are typically fastened with tape to the side of the cradle in such a way to ensure the animal can move its head freely without causing any movement or strain on the electrodes.

Additional protection is needed for ambulatory recording as exposed electrode cables could catch on objects in the field causing injury to the animal. This additional protection includes the fastening of three cable clips to the head of the animal with cyanoacrylate glue[i]. These clips provided strain relief and helped to reduce cable movement which could result in discomfort for the animal. The electrode cables were looped around these clips several times to provide strain relief if the electrode cables are pulled. The electrodes themselves are also attached to the animal with a small amount of cyanoacrylate glue, to prevent electrode movement or an electrode. Cable length is minimised as much as possible and excess loops of cable are taped together. However because of the large amount of movement in a sheep's neck and head sufficient free cable must be kept available to prevent such movement placing strain on the cables. Figure 6.4 shows electrodes attached to a sheep.

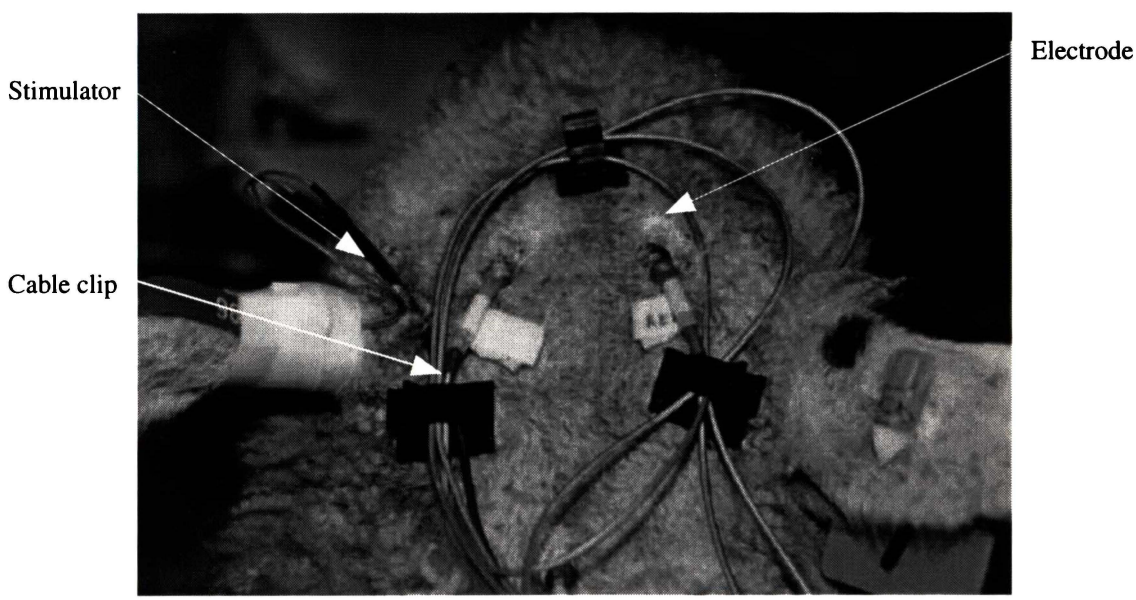


Figure 6.4 – Evoked Response Electrodes Attached to Sheep

[i] Loctite® 454 Instant Adhesive, Loctite Australia Pty. Ltd., Caringbah, NSW 2229, Australia.

After the electrodes are securely fastened the electrode assembly is connected to the FRPM. The electrode assembly is then tested by observing the signals and associated noise levels from the output of the pre-amplifier on an oscilloscope (100mV/div vertical, 200ms/div horizontal) to determine if the electrodes are functioning correctly. Excessive noise (greater than 50μV rms) or baseline drift (greater than 200μV peak-to-peak) in the observed signal typically indicates that either the electrode placement is incorrect or an electrode is damaged. Electrode impedance tests are not usually performed as the signal noise level gives a good indicator of any impedance problems. By lightly tapping the exposed electrode end and by moving the electrode cables the effects of animal motion on the recorded signal can be mimicked and checked.

In practice the above methodology has proved successful. In the over 80 cumulative hours of ambulatory trials only one needle detached from an animal with that failure being attributed to not using enough glue to attach the needle to the animal.

6.1.2 The Auditory Stimulator

The stimulator used for generating the auditory clicks in the ear of the animal was a small and lightweight sub-miniature speaker[i]. The stimulus for the Evoked Response acquisition consisted of short rectangular voltage pulses sent to the stimulator speaker which in turn generated clicks. The sound pressure wave produced by a speaker in response to a brief (less than 200μs) electrical pulse has the approximate appearance of a rapidly decaying sine wave. Frequency analysis of this wave shape shows a broad-band spread of power from approximately 500Hz to 4kHz[122], although this is dependent on speaker design. A high output intensity of at least 80dB Sound Pressure Level (SPL)[123] over the frequency range of the click stimulus was necessary to ensure that adequate stimulation was given to the animal to evoke a clear response. A short stimulus duration of 100μs was used to minimize to possibility of stimulus artifacts appearing in the recordings.

6.1.2.1 Auditory Stimulator Characteristics

To measure the stimulator output characteristics, the stimulator was driven by a signal generator (output impedance 50Ω) producing a square wave at the required amplitude and frequency. The maximum amplitude of interest was 5V, the maximum possible output from the FRPM. The frequency range of interest was from 500Hz to 4kHz to ensure good click reproduction by the speaker.

The sound intensity from the stimulator was measured by a sound level meter[ii] at 25mm from the stimulator. This sound level produced a reading based on the response characteristics of the human ear[124].

Figure 6.5 shows a plot of the stimulator output sound intensity versus frequency for a square-wave waveform of 2V peak-to-peak. Since the impedance of the stimulator changes with frequency due to a reactive component, the terminal voltage across the stimulator will change with frequency due to the finite output impedance of the signal generator. The normalised plot was generated by compensating the measured intensity for the changing drive voltage due to this changing impedance. The plot shows that the peak output intensity of 108dB SPL occurs at approximately 3kHz.

[i] Part No. 224-479, Farnell Electronic Components Ltd., Auckland, New Zealand.
[ii] YF-20 Sound Level Meter, Yu Fung Electronics, Taiwan.

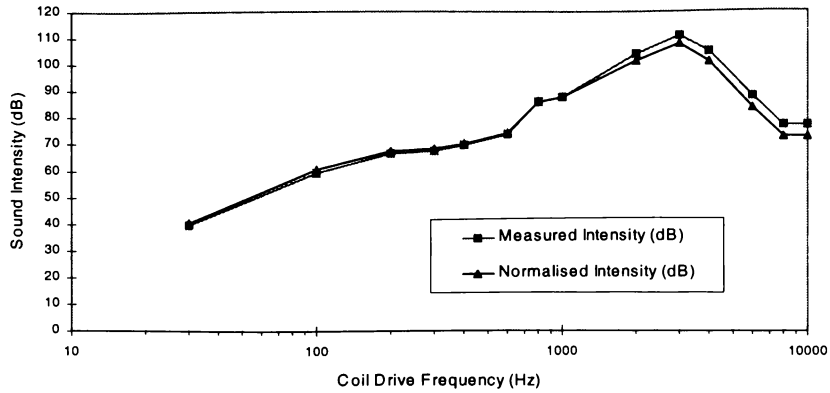


Figure 6.5 – Stimulator Sound Intensity versus Frequency

Figure 6.6 shows the stimulator output intensity for different levels of drive voltage. The same setup that was used to generate Figure 6.5 was also used to generate this plot except the driving frequency was kept constant at 1kHz and the waveform amplitude varied.

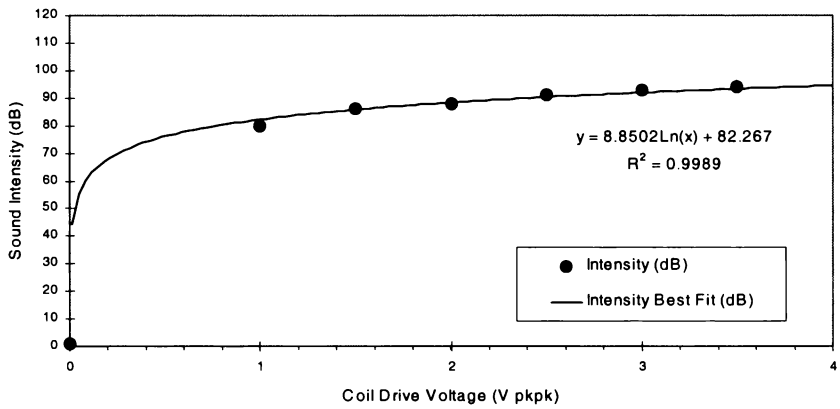


Figure 6.6 – Stimulator Sound Intensity versus Drive Voltage

6.1.2.2 Auditory Stimulator Application

For attachment to the animal, the stimulator was placed gently into the entrance of the ear canal. Tape was positioned around the ear, pinching the stimulator into place. The ear passage was visually checked for buildup of wax or other blocking before the stimulator was inserted. The right ear was left unobstructed.

The wiring leading to the stimulator, from the FRPM, was taped to the electrode wiring assembly. This avoids loops of wiring that could catch on objects in the animal's environment.

6.1.3 Electrocardiogram Recording

Recording Evoked Response waveforms from the subject animal uses only one of the two high-gain input channels available on the FRPM. The second input channel was typically used for recording an Electrocardiogram (ECG) signal from the animal to measure the animals heartrate.

6.1.3.1 Electrocardiogram Electrode Configuration

The ECG waveforms were acquired by recording the voltage differential between an electrode placed just behind the sternum of the animal and the Evoked Response Reference electrode on the animals head. Since the ECG signal is robust it was possible to obtain a good signal despite the noise generated by neck muscle activity and the large distance between the two electrodes. The sternum electrode consisted of a disposable self-adhesive ECG electrode[i]. Figure 6.7 shows the ECG electrode attached to a sheep.

Figure 6.7 – Electrocardiogram Electrode Attached to Sheep

[i] Red Dot 2259 Monitoring Electrodes, 3M Health Care, St. Paul, Minnesota 55144-1000, USA.

The Driven electrode provides common-mode signal cancellation when recording the ECG signal. This is also the case when recording an Evoked Response waveform. The common-mode signal the Driven electrode attempts to cancel is measured at the Reference electrode. For ECG recordings, this signal is not a good estimate of the background noise because the Recording and Reference electrodes are so far apart. Hence the common-mode rejection provided by the Driven Electrode has negligible effect and is effectively reduced to providing a DC current path to the amplifier. In comparison, the electrode placement for recording the Evoked Response where the Recording and Reference electrodes are equidistant from, and relatively close to, the Driven electrode provides good common-mode cancellation. However, the amplitude of the ECG signal is large enough, and the CMRR of the amplifier is sufficient, to enable most features of the ECG waveform to be identified in a recording made using this electrode placement. It is also sufficient for the measurement of heartrate.

If ECG recordings were to be made individually (without Evoked Response) additional adhesive electrodes for the Reference and Driven electrodes would be placed on the animal's body closer to the heart to give a better recording.

6.1.3.2 Electrocardiogram Electrode Application

Site preparation for the sternum electrode began with clipping the wool around the electrode site to a short length (1-2mm). The site was then checked for cleanliness before placement of the electrode. To ensure long-term adhesion of the electrode to the site cyanoacrylate glue was used to bond the electrode to the animal. Additional conductive gel[i] was also applied to the active part of the electrode to ensure low electrical impedance.

Connection to the FRPM is made via the Evoked Response electrode assembly that includes a cable with a dome-type connector[ii] which pushed onto the ECG electrode to make the connection.

6.1.3.3 Electrocardiogram Recording Parameters

The ECG signal was recorded at a rate of 100Hz with 12-bit resolution for 30s periods. This sample rate is too low for clinical diagnosis of the ECG waveform but is sufficient for heartrate analysis while minimizing storage requirements. The signal from the ECG electrodes was amplified with a gain of approximately 470 in a frequency range of 1.7Hz to 1.8kHz. In this case the bandwidth was far beyond the 50Hz Nyquist frequency limit but for simple heartrate acquisition this did not pose any problems due to the large amplitude of the ECG signal. This decision was made to avoid the need to modify and then recalibrate the pre-amplifier modules.

---

[i] Redux Gel, Part No. 651-1024, Medical Products Group, Hewlett Packard, Andover, Massachusetts 01810, USA.

[ii] Tronomed 6" Snap/0.125 SOC 10 LD Leadwire, Part No. S106110, 3M Health Care, St. Paul, Minnesota 55144-1000, USA

6.2 General Experimental Protocol

6.2.1 Animals

The use of animals of this study was approved by both the Ruakura Animal Ethics Committee (Protocol 95HortMirinz 1A) and the University of Waikato Animal Ethics Committee (Protocol No. 270). The animals consisted of 14 sheep, Romney-Cross ewes, of approximately 9 months age. The animals had a mean body weight of 36.1kg with a minimum body weight of 30.6kg and a maximum body weight of 40.4kg. All 14 animals had been habituated to wearing the FRPM for periods of up to three weeks over several months prior to the experimental work. The animals were individually identified by numbered plastic ear-tags. This same number was also sprayed on their backs for easy identification during the behavioral observations.

For more than four weeks prior to, and between, experimentation, animals were kept as a flock in an open paddock. The animals' diet consisted of grass, occasionally supplemented with hay. Water was available *ad libitum* via an open water trough. During experimental work no food was made available to the animals, however water was available at all times. Experiment duration was not more than 1.5hr.

During experimental work only one animal out of the 14 was instrumented at any time, with the exception of the behavior experiment (see Section 6.4.3).

6.2.2 Instrumentation

The experimental protocol consisted of three steps: attaching the instrumentation to the animal, the experiment itself, and removing the instrumentation from the animal.

Several weeks prior to the start of the experiment all 14 sheep had been clipped and had the Velcro pad glued to their backs to hold the instrumentation, as described in Section 3.7. During these weeks preceding the experiment, the animals were periodically instrumented to ensure each animal was familiar with both the instrumentation and the process of attaching the instrumentation to the animal.

At the start of every experiment the animals were first rounded up and placed in a small holding pen. The appropriate target animal was selected and placed in the holding cradle. The animal's head was then disinfected and the three Evoked Response electrodes were then inserted into the animal, the ECG electrode was attached to the animal's sternum, and the stimulator speaker was inserted into the animal's ear. The various electrode and stimulator cabling were fastened to the animal with the use of plastic clips which had earlier been glued to the animal's head.

After the electrodes and stimulator were attached to the animal the FRPM unit was attached to the animal's back the Velcro pad. The electrode and stimulator cables were connected to the FRPM and the recorded signal levels were checked on an oscilloscope for high noise levels which might

indicate a faulty electrode or connection. Electrode susceptibility to movement artifacts was also checked by lightly tapping each electrode and its cable while watching the oscilloscope for extreme artifacts. Faulty electrodes or connections usually resulted in measurements on the animal being missed on that day while repairs were undertaken.

6.3 Preliminary Work

Preliminary work was performed on the group of 14 animals to select a suitable stimulus amplitude and a suitable stimulus rate for the stress trial. Evoked Response waveforms were acquired with different stimulus amplitudes and different inter-stimulus intervals. For the Evoked Response amplitude versus stimulus intensity experiments the stimulus amplitude was varied over a range of 20mV to 5V (approximately 37dB to 98dB SPL) in steps of 15dB, and at each step a waveform was acquired. A waveform was also acquired with no applied stimulus, making a total of six waveforms. For the Evoked Response amplitude versus inter-stimulus interval experiment the interval was varied in five steps between 125ms and 2s (steps at 125ms, 250ms, 500ms, 1s, and 2s), and at each setting a waveform was acquired. A nominal 512 stimuli were presented to the animal to obtain each waveform. Typically one Evoked Response waveform was acquired every 5min.

6.3.1 Auditory Evoked Response Amplitude versus Stimulus Intensity

The stimulus amplitude was selected by recording evoked response traces from several animals at several discrete amplitudes. The amplitudes of several prominent peaks in the response were normalised then plotted against frequency. The stimulus rate for this experiment was 150ms.

Figure 6.8 shows a plot of peak amplitude versus stimulus intensity for identifiable peaks in three different animals (animal numbers 8, 11, and 14). If a peak was not identifiable it was omitted. The thick line is the average of the normalised amplitudes of the features identified. The average shows a steady increase in amplitude with stimulus intensities of up to 75dB SPL. After 75dB saturation begins to occur, limiting the response amplitude for stimuli of higher intensities.

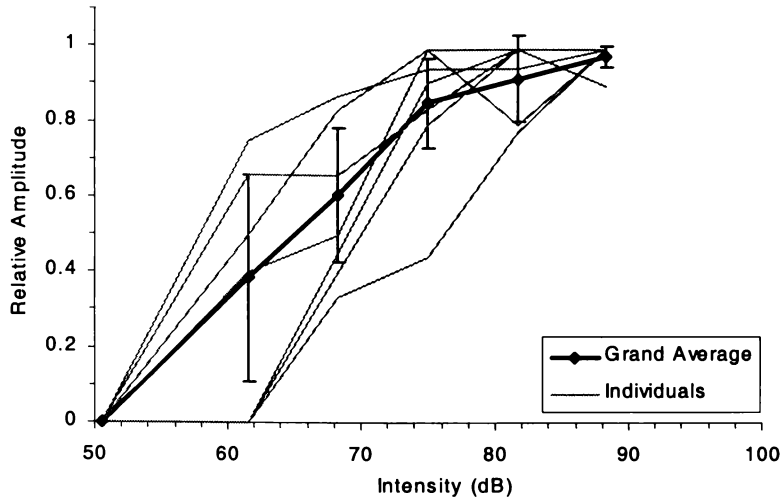


Figure 6.8 – Response Amplitude vs. Stimulus Intensity (Error bars are ±1 standard deviation)

Although the characteristic shows an optimal point at about 75dB, the stimulus intensity chosen for the stress trial was 90dB, the maximum intensity able to be produced by the instrumentation. This high stimulus intensity was chosen to account for inter-animal variability, as is visible in Figure 6.8, and to ensure that a response of maximum amplitude was produced.

6.3.2 Evoked Response Amplitude versus Stimulus Rate

The stimulus rate was selected by recording evoked response traces from several animals at several discrete inter-stimulus intervals. The amplitudes of several prominent peaks in the response were normalised then plotted against the stimulus presentation rate. The stimulus amplitude for this experiment was 90dB SPL.

Figure 6.9 shows a plot of peak amplitude versus stimulus rate for identifiable peaks in three different animals (animal numbers 8, 11, and 14). If a peak was not identifiable it was omitted. The thick line is the average of the normalised amplitudes of the features identified. The average shows a relatively stable response amplitude for stimulus rates of greater than 2Hz.

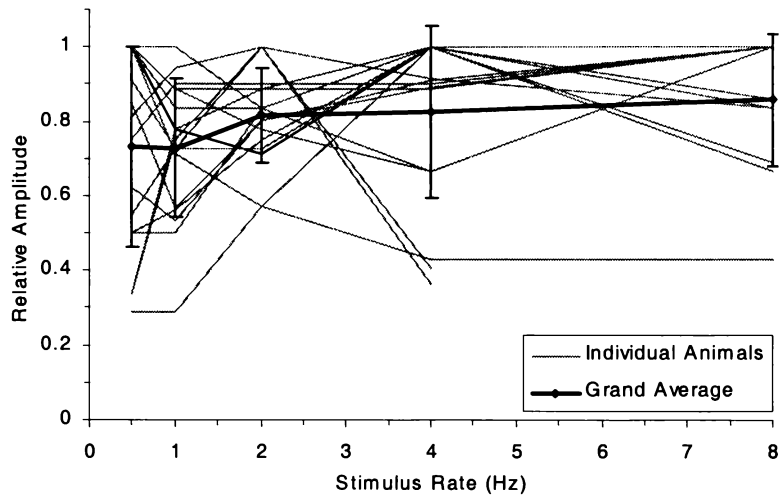


Figure 6.9 – Response Amplitude vs. Stimulus Rate (Error bars are ±1 standard deviation)

Based on this plot a stimulus rate of 6.7Hz (inter-stimulus interval of 150ms) was chosen for the stress trial. Using this stimulus rate a single evoked response recording could be expected to take about 3min including data processing and storage.

6.4 Stress Trial Design

6.4.1 Experimental Protocol

The general protocol for the stress trial was the same as that presented in Section 6.2.

Once the instrumentation had been attached to the animal and the signal levels checked the girth strap was placed around the animal. After a final check on cable placement and flexibility with regard to animal movement, the recording sequence was started on the FRPM and the animal was released into the pen where the other animals were waiting.

After a 5min delay to allow the target animal to settle back into the group the experiment comprising of three 20min periods, as described in Section 6.4.3, was begun. During these periods an Evoked Response waveform and an ECG waveform were acquired every 5min. A total of twelve Evoked Response waveforms and twelve ECG recordings were acquired during each experiment.

Once the experiment was complete all of the animals were again rounded up and placed in the small holding pen. The instrumentation was removed from the target sheep and its head was again disinfected. The animals were then either released to graze in the field or another animal was prepared for an experiment.

The radio module in the FRPM was not used in these experiments as it was deemed unnecessary to have a live data feed from the animal. Instead, the recorded and processed Evoked Response waveforms were saved in the FRPM file system after each acquisition, and the data downloaded from the FRPM to a computer after removal from the animal.

6.4.2 Recording Parameters

The stress experiment used the baseline values established during preliminary work (Section 6.3) to determine the stimulus intensity and inter-stimulus interval. For the stress experiment, the stimulus pulses had an amplitude of 2V and a duration of 100μs, the amplitude corresponding to an acoustic intensity of approximately 90dB SPL. The clicks were of the condensation[125] type i.e. initial movement of the speaker diaphragm was towards the tympanic membrane. The inter-stimulus interval was 150±50ms, the uncertainty being added by the acquisition program to ensure stimulus presentation did not accidentally synchronize with any source of external interference and also to avoid the possibility of time aliasing caused by possible responses after 90ms latency "wrapping around" and interfering with subsequent stimulus/response pairs. A nominal 1024 stimuli were presented to the animal to obtain each waveform although in some cases extra stimuli were presented if the acquisition program discarded parts of the recording due to noise as described in Section 5.1. The maximum number of stimuli that could be presented to obtain any one waveform was limited to 1280 clicks.

For each stimulus the signal from the Evoked Response electrodes was sampled at a rate of 5kHz with 12-bit resolution for 100ms. Of this 100ms, 10ms was recorded before the stimulus was presented to the animal and 90ms after the stimulus was presented. The pre-stimulus period was used to establish a base-line for later processing. The signal from the electrodes was amplified with a gain of approximately 1900 in a frequency range of 1.7Hz to 1.8kHz.

Recordings were made in 5min slots, each slot comprising one Evoked Response recording and one ECG recording. Four slots made up each experimental period giving a total of twelve slots for each experiment. The recording sequence for each of the twelve 5min recording slots is shown in Figure 6.10. Note that the Evoked Response and ECG recordings were not concurrent.

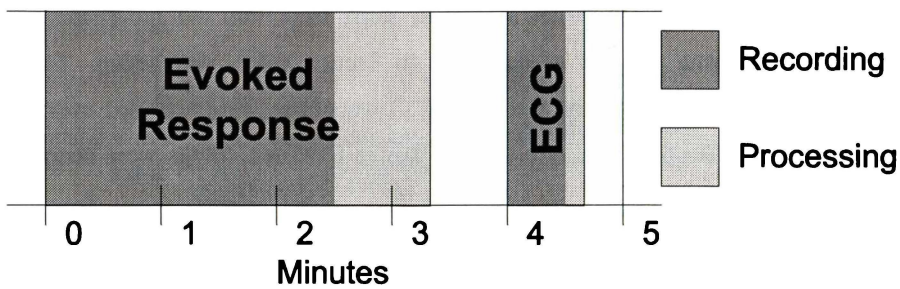


Figure 6.10 – Time Sequence of a Recording Slot

The time taken to acquire each Evoked Response waveform was approximately 200s. Of this total, actual acquisition took approximately 154s, the rest comprising signal processing time. The time taken to acquire the ECG recording was approximately 40s, comprising 30s of recording and 10s of signal processing.

6.4.3 Stressors

Each experiment consisted of three 20min periods following FRPM instrumentation. The first, or control, period involved holding all of the animals together in a pen as shown in Figure 6.11. The final, or recovery, period also involved holding the animals together in the same pen. During the middle, or stress, period one of four different putative stressors was applied to the animals. The stressors consisted of a control stress, a 'distraction' stress, an isolation stress, and a 'fear' stress.

The randomized order in which individual sheep were exposed to the control, distraction, and isolation stressors is shown in Table 6.1. The first experiment with animal 3 and the isolation stress, the last experiment with animal 2 and the distraction experiment. Approximately four experiments were conducted each day, dependent on weather and equipment operation.

Animal	1	2	3	4	5	6	7
Control	32	8	30	10	31	33	13
Distraction	38	42	9	23	11	34	39
Isolation	7	21	1	26	20	12	3

Animal	8	9	10	11	12	13	14
Control	22	15	4	5	18	28	19
Distraction	14	40	16	25	27	41	29
Isolation	2	35	24	17	36	6	37

Table 6.1 – Control/Distraction/Isolation Stress Exposure Order

6.4.3.1 Control Stress

The control stress was exactly the same as the normal control and recovery periods and was designed to produce stable baseline responses against which response from the other stressors could be compared.

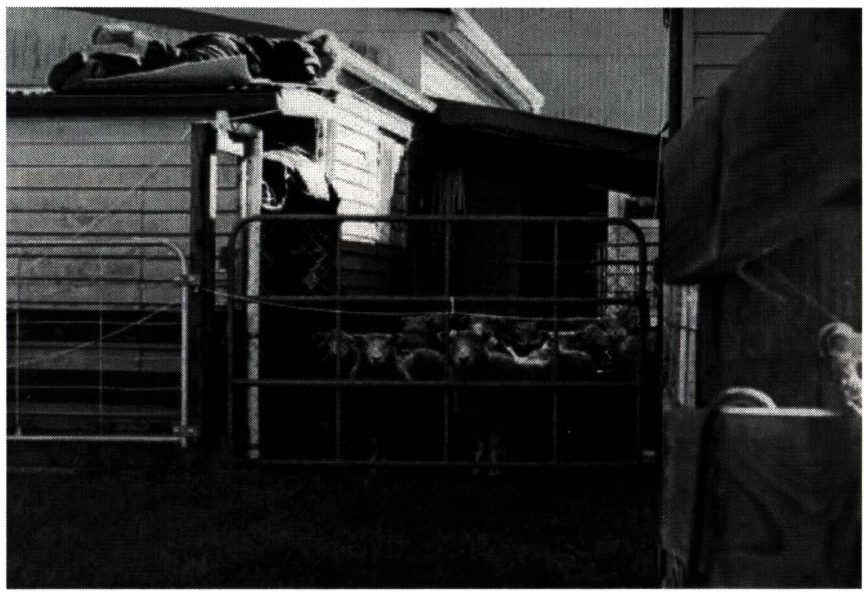


Figure 6.11 – The Control Period

6.4.3.2 Distraction Stress

The distraction stress consisted of a moving device driven by water. Figure 6.12 shows this device in operation during a distraction experiment. The device consisted of two different lengths (41cm and 55cm) of 78mm diameter plastic tubing that could pivot about their center. The lower end of these tubes were sealed. Water was poured into these tubes from a water hose. As the water level in each tube rose the tube would become unbalanced and tip, dumping the load of water onto the ground. The arc of movement was approximately 180°. The tube then rotated back to its resting position and slowly refilled. Since the tubes were different lengths the cycle

time for filling and tipping was different for each tube. The longer tube tipped at a rate of approximately once every 45s while the longer tub tipped approximately every 90s. The noise (water pouring onto concrete and the tube hitting its mechanical stops) and motion from the tubes tipping was used as the distraction stressor. The device was placed approximately 2m from the sheep, who were kept together as a group.

Figure 6.12 – The Distraction Stress Device

6.4.3.3 Isolation Stress

The isolation stress consisted of isolating the instrumented animal from the rest of the flock. During the control and recovery periods the instrumented animal was kept in a separate pen immediately next to the other animals. The arrangement of the pens allowed the instrumented animal virtually unobstructed view of the other animals. During the stress period an outside gate was opened remotely, allowing the other sheep to leave their pen and graze in the nearby field. During this time the instrumented animal was kept in its pen. This was designed to induce stress in the instrumented sheep by isolating it by some distance from the rest of the flock. Figure 6.13 shows an animal undergoing the isolation stress while the other animals graze outside the pen. Note that the instrumented animal is in a separate pen to that used for the control stress.

Figure 6.13 – The Isolation Stress

6.4.3.4 Fear Stress

The experiments involving 'fear' were run as a separate experiment one week after the combined control, distraction, and isolation stress experiment. The use of fear as a stressor was only considered after completion of this first experiment. The fear stress involved exposing the animals to a dog from a distance of approximately 3m. Dogs are a recognized predator to sheep[126], therefore I hypothesized that the presence of a dog would induce fear and anxiety states in the sheep. Table 6.2 shows the order in which sheep were exposed to the dog. Note the two separate groups of 7 sheep.

Group 1 Animals	**1**	**4**	**5**	**8**	**9**	**12**	**13**
Fear	2	5	6	9	10	13	14

Group 2 Animals	**2**	**3**	**6**	**7**	**10**	**11**	**14**
Fear	1	4	3	8	7	12	11

Table 6.2 – Fear Stress Exposure Order and Grouping

The dog used was a large female Rottweiler which was chained to a nearby fence, preventing it from approaching the sheep. All animals in the group were exposed in every experiment during the 'fear' or stress phase. To reduce familiarization of adaptation to the dog over the course of experimental repetitions, the animals were split into two groups of seven. At any time only one group, including the instrumented animal, was exposed to the dog. The two groups were also grazed separately outside of experiment time. Figure 6.14 shows a group of animals being exposed to the fear stress.

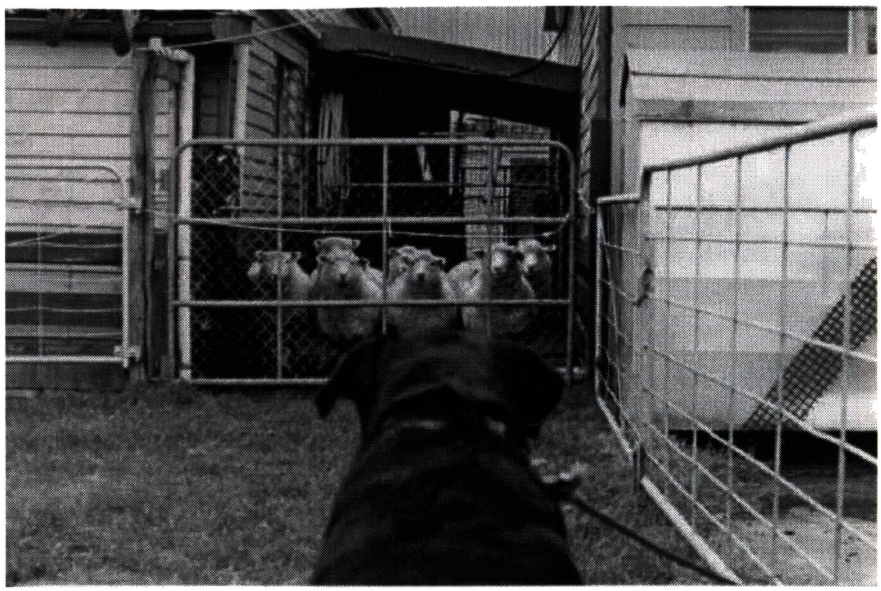


Figure 6.14 – The Fear Stress

6.4.4 Behavioral Observations

Behavioral observations were made during the course of experiments. These observations were to assess the effects of the stressors on the animals, and to look at the effect of other factors such as instrumentation on the animals' behavior. The types of behaviors that were recorded are described in Table 6.3.

Target Animal - Focal Scan		**Other Animals - Episodic**
Cumulative Behaviors	**Behavioral Events**	**Behaviors Recorded at 5min Intervals**
Lying	Fighting	Fighting
Standing	Body shake	Body shake
Walking	Fence chewing	Fence chewing
Grazing	Defecation	Defecation
	Drinking	Drinking
	Grooming	Grooming
	Head scratch with leg	Head scratch with leg
	Head shake	Head shake
	Leg stamp	Leg stamp
	Pawing other animals	Pawing other animals
	Rubbing against fence	Rubbing against fence
	Urinating	Urinating
	Vocalization	Vocalisation
	Pawing the ground	Pawing the ground
	Jumping	Jumping
		Lying
		Standing
		Walking
		Grazing

Table 6.3 – Behaviors Recorded During Experimental Periods

The target animal behavior was recorded on both a cumulative, and on an event basis, by continuous (focal) sampling. Continuous sampling involves recording the exact sequence of behavior patterns over the entire observation period[127]. The cumulative behaviors were recorded as the total number of seconds the target animal spent lying, standing, or walking. The event behaviors were recorded as the number of times the target animal performed the behavior.

The representative behavior of the other uninstrumented animals was recorded by randomly selecting four animals. The number of animals in this sample that were performing a behavior from the pre-defined list of categories, was recorded every 5min. For the lying, standing, and walking behaviors this number was later scaled to produce an averaged cumulative figure for each 5min period. For example, if three animals were observed lying and one standing at the sampling instant this would be converted to 225s (0.75 * 300s) of lying and 75s (0.25 * 300s) of standing. In this way the basic behaviors of lying, standing, and walking could be directly compared between the representative sample and the target animal.

The event behaviors listed in Table 6.3 were not directly compared between the target and sample animals but were only used to assess the behavior of the target animal between the control, stress, and recovery periods.

6.4.5 Signal Processing and Statistical Analysis

The recorded Evoked Response waveforms were processed with the feature extraction algorithm as described in Section 5.3 and by Principle Component Analysis followed by Discriminant Analysis as described in Section 5.4.

The recorded ECG waveform was analyzed using the Fast Fourier Transform method described in Section 5.6 to extract the mean heartrate from the 30s recording periods.

Chapter 7 The Auditory Evoked Response as an Indicator of Stress

7.1 Baseline Measurements

In the Control experiments the animals were instrumented but not subjected to experimental stressors. Data obtained were used to investigate both inter- and intra-animal variation in both the recorded Evoked Response waveforms and the measured heartrate.

7.1.1 Auditory Evoked Response

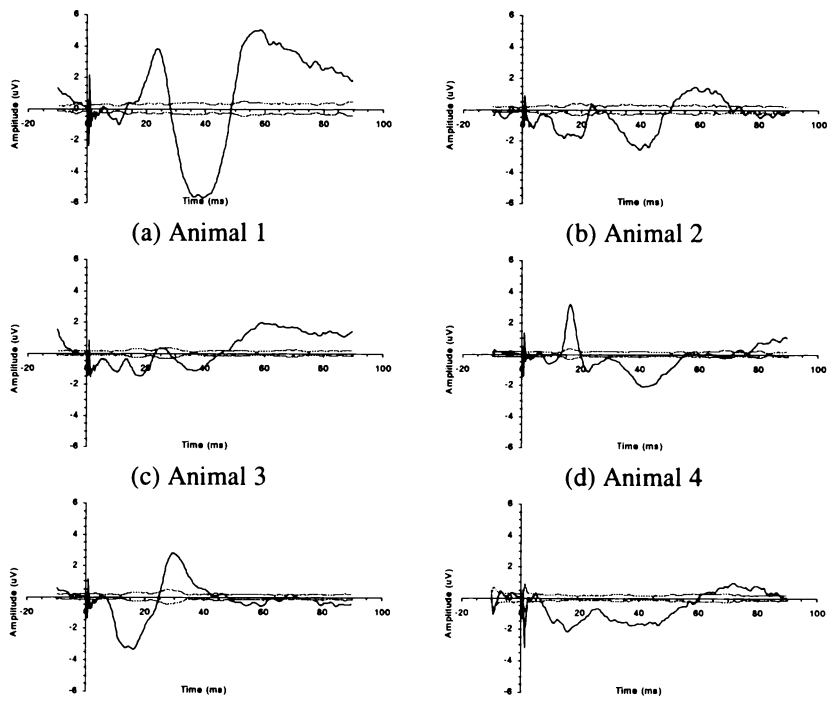
To investigate variation in the Evoked Response waveform within an individual animal, the 12 original waveforms obtained from each animal during the Control experiment were averaged. Each waveform from the animal was then correlated against this mean waveform. The mean and range of the 12 resulting correlation coefficients for each animal is presented in Table 7.1.

Of the 14 animals recorded, Table 7.1 shows that 12 animals had mean correlation coefficients of greater than 0.70 indicating that the recorded Evoked Response waveforms were repeatable within individual animals. The exceptions are animals 10 and 11 which show poor mean correlations of 0.62 and 0.56 respectively. These poor figures could possibly be due to faulty or damaged electrodes as the Control recordings from both of these animals were made consecutively on the same day.

Animal Number	Correlation Coefficients		
	Minimum	Mean	Maximum
1	0.89	0.94	0.98
2	0.46	0.74	0.90
3	0.70	0.86	0.95
4	0.62	0.83	0.95
5	0.80	0.88	0.93
6	0.41	0.76	0.92
7	0.10	0.80	0.96
8	0.43	0.76	0.92
9	0.38	0.73	0.87
10	0.02	0.62	0.91
11	0.15	0.56	0.78
12	0.29	0.82	0.95
13	0.69	0.84	0.95
14	0.66	0.87	0.95

Table 7.1 – Mean and Range of Evoked Response Waveform Correlation Coefficients Within Each Animal for the Control Experiment

Figure 7.1 shows the mean of the 12 original waveforms of the Control experiment from each animal, together with the upper and lower Standard Error in the Mean (SEM) limits.

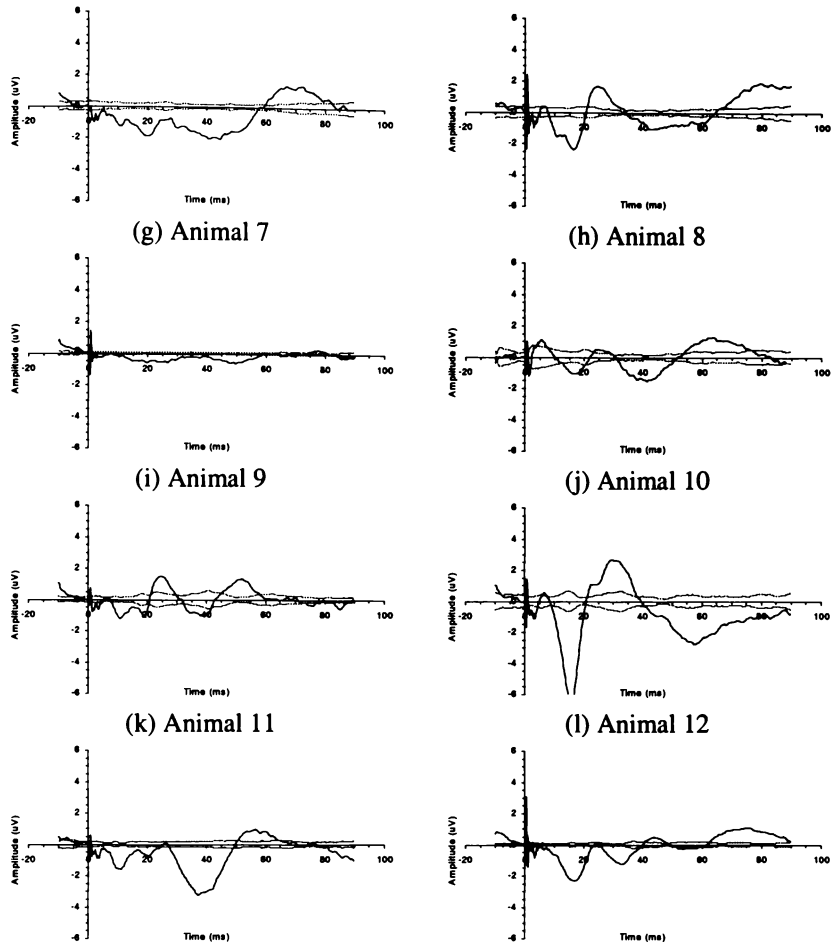


(a) Animal 1 (b) Animal 2

(c) Animal 3 (d) Animal 4

(e) Animal 5 (f) Animal 6

Figure 7.1 (i) – Mean Evoked Response Waveforms of the Animals for the Control Experiment (Mean ± SEM)



(g) Animal 7 (h) Animal 8

(i) Animal 9 (j) Animal 10

(k) Animal 11 (l) Animal 12

(m) Animal 13 (n) Animal 14

Figure 7.1 (ii) – Mean Evoked Response Waveforms of the Animals for the Control Experiment (Mean ± SEM)

From the waveforms in Figure 7.1 it can be seen that the Evoked Response waveforms recorded from different animals were significantly different and were not readily comparable between animals. Table 7.2 presents the correlation coefficients between animals based on the mean waveforms in Figure 7.1.

Animal	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	•													
2	0.84	•												
3	0.75	0.80	•											
4	0.50	0.25	0.21	•										
5	-0.29	0.11	0.12	-0.44	•									
6	0.64	0.64	0.77	0.34	0.12	•								
7	0.65	0.65	0.76	0.44	0.04	0.91	•							
8	0.33	0.32	0.48	0.15	0.44	0.65	0.55	•						
9	0.21	0.29	0.43	0.24	0.16	0.66	0.72	0.40	•					
10	0.76	0.84	0.66	0.26	0.11	0.75	0.74	0.39	0.40	•				
11	0.44	0.55	0.41	-0.18	0.27	0.13	0.03	0.23	-0.15	0.34	•			
12	-0.39	-0.12	-0.20	-0.47	0.81	0.00	-0.12	0.52	0.11	-0.03	0.17	•		
13	0.90	0.84	0.66	0.47	-0.18	0.59	0.60	0.20	0.28	0.74	0.57	-0.33	•	
14	0.29	0.40	0.68	-0.20	0.32	0.71	0.63	0.59	0.48	0.36	0.26	0.20	0.25	•

Table 7.2 – Mean Evoked Response Waveform Correlation Coefficients Between Animals for the Control Experiment

Table 7.2 confirms the observation that the waveforms recorded from different individuals were not readily comparable as the correlation coefficients are typically low. Notable exceptions were animals 6 and 7 (r=0.91) and animals 1 and 13 (r=0.90).

To further investigate the similarity between animals a Cluster Analysis was performed. Cluster Analysis[128] provides an automated method to group a number of observations into clusters so that all cases within a cluster are more similar to each other than those observations in other clusters. This clustering process results in a hierarchy of relationships between clusters, ranging from a single cluster containing all the observations to as many clusters as there are observations, each cluster containing a single observation. There are many ways to cluster observations but no general agreement on the best method, hence clustering is most often used only as a data exploration tool. Typically the optimum number of clusters to best describe the data set is chosen arbitrarily.

To perform the Cluster Analysis, a Principal Components Analysis (PCA) of the waveforms in Figure 7.1 was performed first which produced a set of 14 components and hence 14 coefficients per animal. The first four coefficients (which accounted for 92% of the inter-animal variance) were then used as the input to the complete-linkage Cluster Analysis. Figure 7.2 shows the results of the Cluster Analysis.

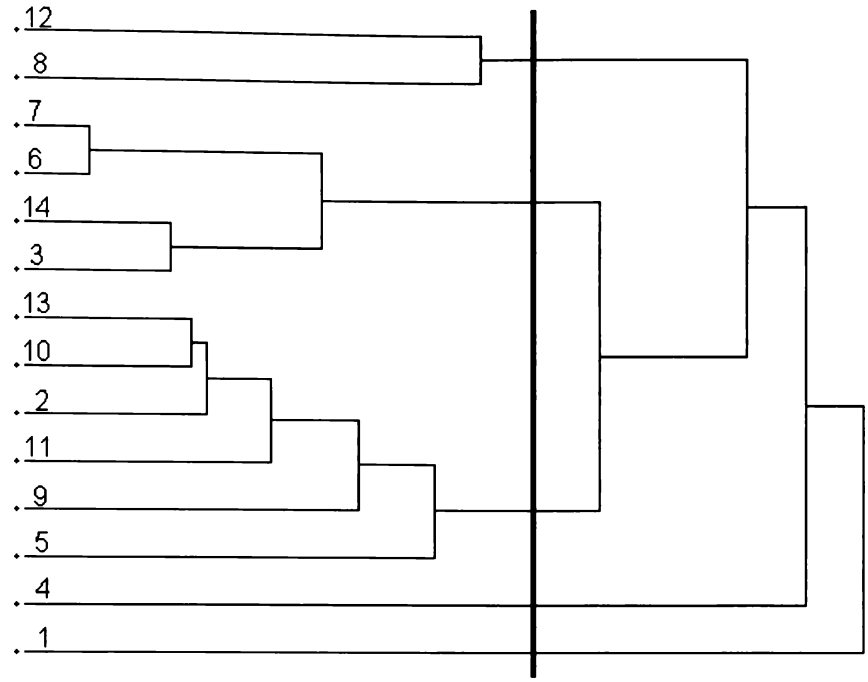


Figure 7.2 – Cluster Analysis of Similarity between Animals

The Cluster Analysis suggests that there are five reasonably distinct groups of animals, the animals within each group exhibiting similar waveforms with each other but not necessarily with the animals from the other groups.

Arbitrarily taking five clusters as the splitting plane (marked by a bold vertical line in Figure 7.2), the animals are clustered as presented in Table 7.3. The clusters in Table 7.3 are listed in order of significance, as are the animals within each cluster.

Cluster	Animals in Cluster
1	Animal 1
2	Animal 4
3	Animals 8, 12
4	Animals 5, 9, 11, 2, 10, 13
5	Animals 3, 14, 6, 7

Table 7.3 – Cluster Analysis of Similarity between Animals (5 clusters)

Figure 7.3 shows the five clusters obtained. The average waveforms from the animals in the cluster have been normalised for amplitude to emphasize similarity in shape. Because of the normalisation, the y-axis units have no meaning and have been omitted. The mean waveform of all the animals is also presented.

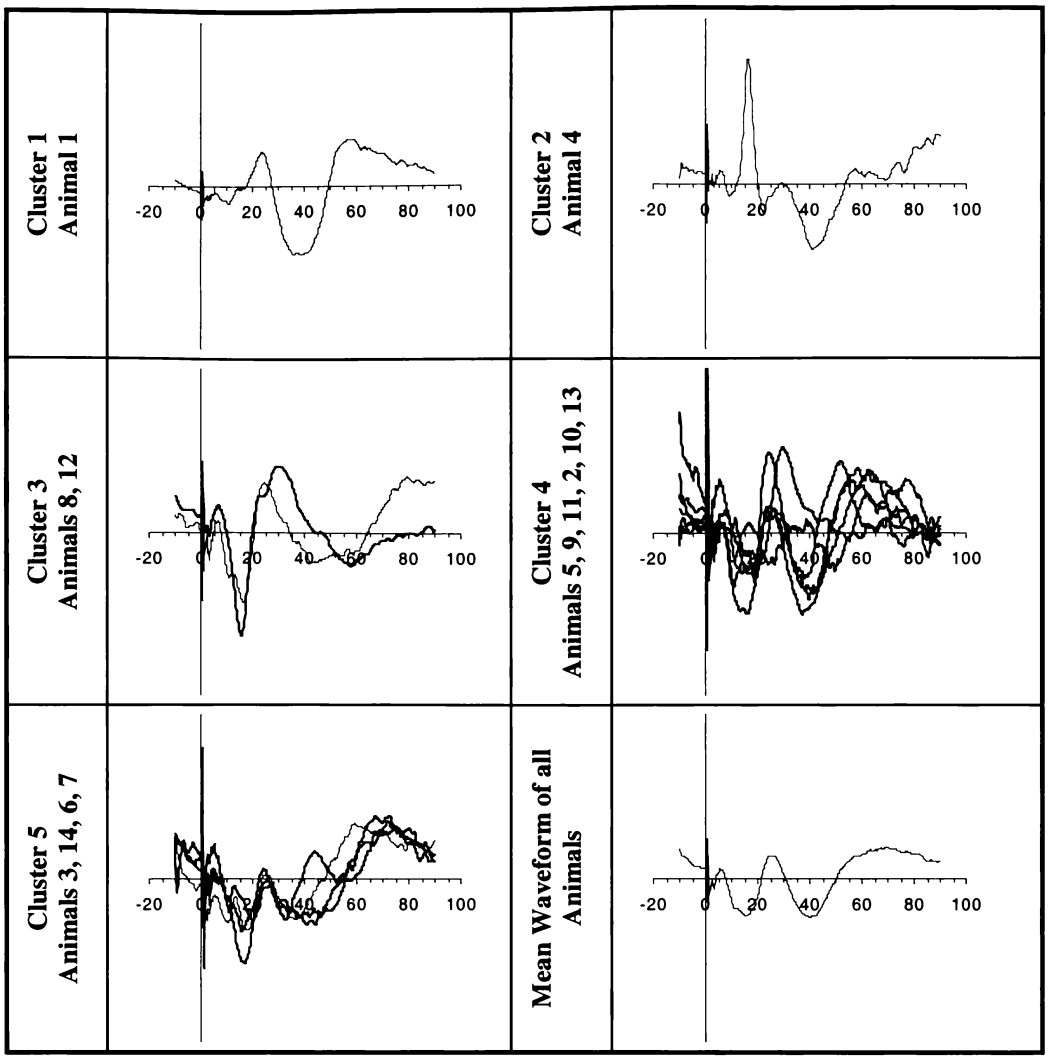


Figure 7.3 – Evoked Response Waveforms of Animals Sorted by Cluster Analysis (x-axis is time in ms)

The results of the Cluster Analysis differ from the observations based on the correlation coefficients. This is attributable to the fact that the correlation coefficients are a measure of how well individual samples in the waveform compare whereas, by using the first four significant coefficients as the input to the Cluster Analysis, the Cluster Analysis performed its clustering based on the shape of the waveforms. By looking at the shape of the waveform the effect of samples that are contaminated by residual noise is minimised. In the case of the correlation, this noise can considerably reduce the resulting correlation coefficient. Another advantage of the PCA/Cluster Analysis approach is that the clustering is performed on the differences between waveforms rather than on the whole waveform as is the case when using correlation. By analyzing the differences between waveforms, relatively small features that differ between waveforms have a much greater effect on the result.

Figure 7.3 shows that there are significant differences in the response waveforms between groups, but that the waveforms are reasonably similar within any particular group. It is unclear what the origin of these groupings could be, however possible sources include genetic differences between

animals (resulting in physiological differences between animals), or non-repeatable electrode placement between animals. It is unlikely that the differences are due to recording artifacts caused by, for example, insufficient averaging or faulty electrodes, since these would also lead to a greatly increased residual noise level in the final waveforms which was not observed in the recorded data.

Although the animals used in the experiment were the same breed and the same approximate age, size, and weight, it is possible that there are genetic differences between the animals as the exact pedigree was not available from farm records. However, it is unlikely that this could account for such significant differences. Much more likely is difficulty in repeating the exact electrode placements between animals. This could be expected to have a much greater effect as the thickness of tissue and muscle on the skull of a sheep can vary greatly between animals, as can the thickness and local shape of the skull itself[120]. This can lead to difficulty in the location of the bone 'markers' on the skull of the animals that serve as reference points for electrode placement between animals, and can also make electrode insertion difficult due to ridges in the surface of the skull.

In this case the Cluster Analysis has produced no clear pattern for the variation between animals, hence further research over a much larger sample of samples is required to better identify the source of these differences between the animals.

Based on the strong waveform similarity seen within an individual animal, compared to that between animals, it was decided to analyze the Evoked Response waveforms on an individual animals basis rather than as a group. The Discriminant Analysis used for analyzing the effect of stressors on the Evoked Response thus treats each animal as a separate case. It uses a different set of basis vectors obtained from the Principal Components Analysis for each animal.

7.1.2 Heartrate

To investigate variation in heartrate within individual animals without exposure to a stressor, the 12 values of heartrate obtained during the Control experiment from each individual animal was analyzed. The average heartrate of the 14 animals as a group was also calculated. The results are presented in Table 7.4 as beats per minute (bpm).

Animal Number	Average Heartrate (bpm)	Heartrate Variation (bpm)
1	111.12	8.87
2	94.89	6.06
3	100.71	7.89
4	101.49	5.20
5	95.96	8.24
6	101.83	6.02
7	92.07	8.89
8	104.46	3.96
9	89.91	7.34
10	80.65	6.24
11	82.04	9.63
12	84.94	3.44
13	77.87	3.75
14	89.34	7.17
Average	93.00	4.07

Table 7.4 – Heartrate Variation Within Each Animal and as a Group for the Control Experiment (Mean ± SD)

These results show that the average resting heartrate of the animals can vary considerably between individuals over the 60min period of time used to obtain the 12 measurements from each animal. However, the variability within each animal and within the group is small. Although the average heartrate for the Control experiment differs between animals, it is more useful to look at the heartrate profile before, during, and after exposure to a stressor. The shape of these profiles is independent of the Control period baseline heartrate making the heartrate differences between animals in Table 7.4 unimportant.

Since both the intra- and inter-animal variability is typically small during the Control experiment, it suggests that the heartrate data can be used for comparisons both within an individual animal and between animals in the group.

7.2 Effect of Stressors

Each of the 14 animals was exposed to four stressors while both Auditory Evoked Response (AER) waveforms and heartrate were recorded from the target animal in the group (see Section 6.4.2). The stressors consisted of a Control stressor (no applied stress), a Distraction stressor (a water-driven mechanism), an Isolation stressor (the target animal removed from the flock), and a Fear stressor (exposure to a dog at short distance).

Each experiment consisted of three 20min periods: a Control period to establish baseline measurements, an Experiment period when the target animal was exposed to the stressor, and a Recovery period where the target animal was monitored as it recovered from exposure to the stressor.

7.2.1 Auditory Evoked Response

Auditory Evoked Response waveforms were recorded from the target animal every 5min, producing a total of 12 waveforms for each experiment.

7.2.1.1 Auditory Evoked Response Example Waveforms

As examples of the Evoked Response waveforms recorded from different animals during exposure to different stressors, the original waveforms recorded from three different animals are presented. As already described (Section 5.4), animals were examined on an individual basis and are presented as such. Animals 1, 4 and 8 are significant since they all show good examples of changes in the recorded Evoked Response over the course of each experiment.

Each sequence consists of 12 waveforms, separated in time by 5min. Because of the 20min experimental periods the first four waveforms represent the Control period, the middle four represent the Experimental period where the animal is exposed to the stressor, and the last four waveforms represent the Recovery period. Figure 7.4 shows the arrangement of each sequence.

Figure 7.5 through Figure 7.7 show the sequences from animals 1, 4, and 8 during each of the four different experiments. The recorded waveform is accompanied by 99% confidence levels based on the noise level during recording (see Section 5.1.5).

Control period	Waveform C1 Time = 0min	Waveform C2 Time = 5min	Waveform C3 Time = 10min	Waveform C4 Time = 15min
Experiment period	Waveform E1 Time = 20min	Waveform E2 Time = 25min	Waveform E3 Time = 30min	Waveform E4 Time = 35min
Recovery period	Waveform R1 Time = 40min	Waveform R2 Time = 45min	Waveform R3 Time = 50min	Waveform R4 Time = 55min

Figure 7.4 – Arrangement of Evoked Response Waveform Sequences

Animal 1 showed a stable response during the Control experiment. During the Distraction Experiment this animal exhibited a slightly different response marked by higher amplitude response peaks at around 25ms, 40ms, and 55ms. In the Recovery period the waveform tends back towards its the original Control period shape, although the amplitude has increased significantly. The Isolation experiment also showed changes in the Experiment period with respect to the Control period shape, in particular double peaks appearing around 35ms and 60ms. However, the Recovery period, when the rest of the group was reunited with the target animal, shows a complete change in the response waveform. The waveform developed several high amplitude peaks before 20ms and, although noisy, appears repeatable in the last three waveforms. Finally the Fear experiment also showed a change in response between the Control and Experiment periods marked by an increase in the amplitude of the peaks at 20ms, 35ms, and 40ms. There also appeared to be a recovery trend from waveforms E1 through E4, and waveform R1 through R4. The approach and retreat of the dog at waveforms E1 and R1 may account for the large responses present in these waveforms.

Animal 4 also showed a stable response during the Control experiment and no apparent changes in the Distraction experiment. The Isolation experiment may have produced a change in response during the Experiment period although no return to waveform shape of the Control period was visible during the Recovery period. The Fear experiment for this animal provided the clearest example of a stable Control period, a sustained change in the response during exposure to the dog in the Experiment period, and a return to the original Control period waveform shape in the Recovery period. In particular, exposure to the dog was marked by the sustained appearance of a large peak at 35ms during the Experiment period. This peak decayed to its Control period amplitude during the Recovery period.

Animal 8 showed a stable response during the Control experiment despite the unexplained slope that appeared during the Recovery period. However there may have been a recovery-type trend at the start of the Control period due perhaps to the handling of the animal immediately prior to the

start of recording. A similar trend can be observed in all of the response sequences for this animal. The Distraction experiment showed no apparent change during the Experiment period. The Isolation experiment showed large responses at both the start and finish of isolation. There may have been a recovery trend during the Recovery period, however more waveforms would have been needed to confirm this. Support for this comes from the final waveform shape in the Recovery period matching the first waveform in the Control period. The Fear experiment also showed large responses at the approach and retreat of the dog. Waveforms 7 and 8 appear to return to baseline levels (waveforms 2, 10, 11, 12) despite the dog still being present, possibly indicating adaptation.

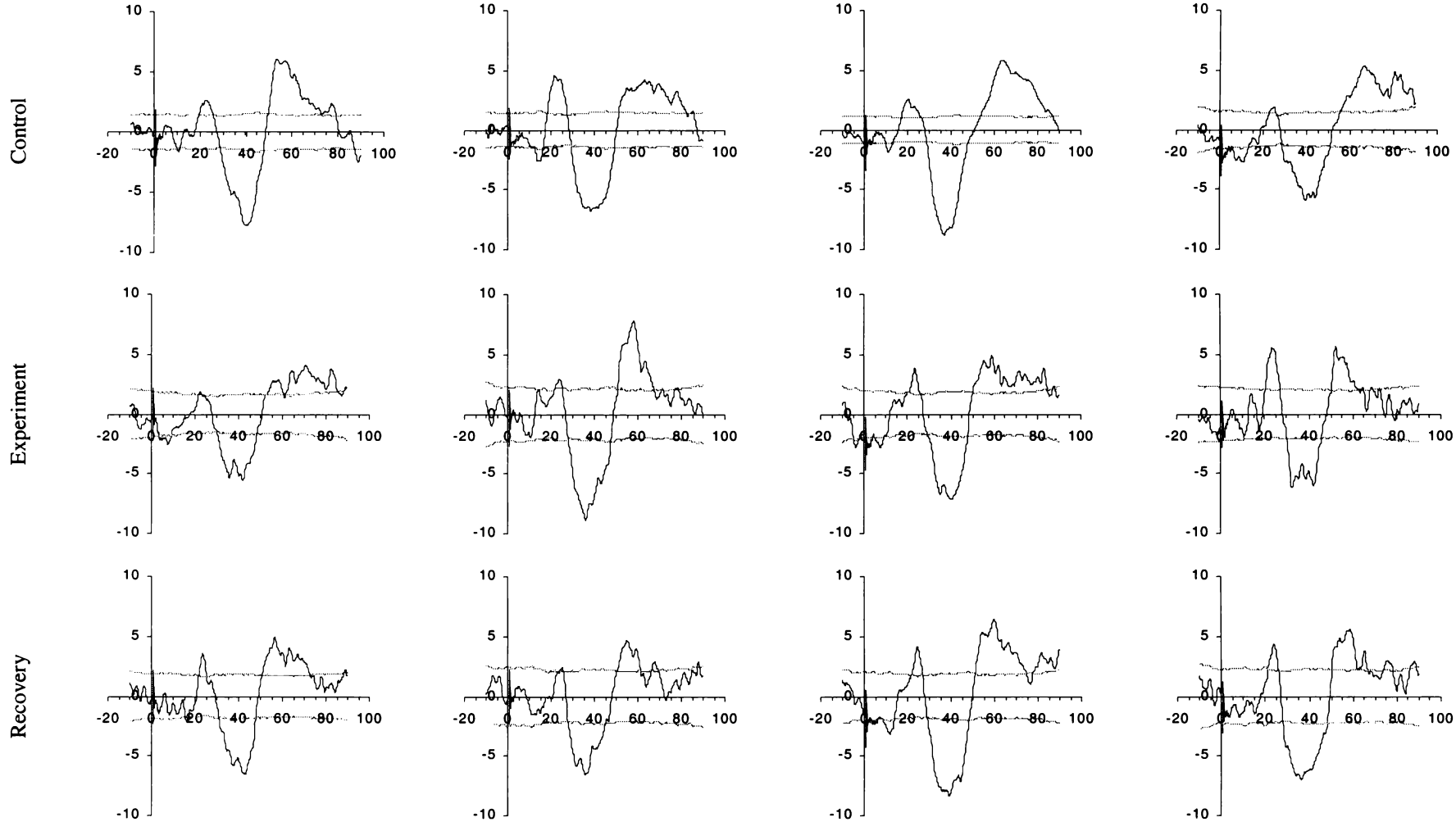


Figure 7.5 (a) – Animal 1 Control Experiment Evoked Response Waveforms

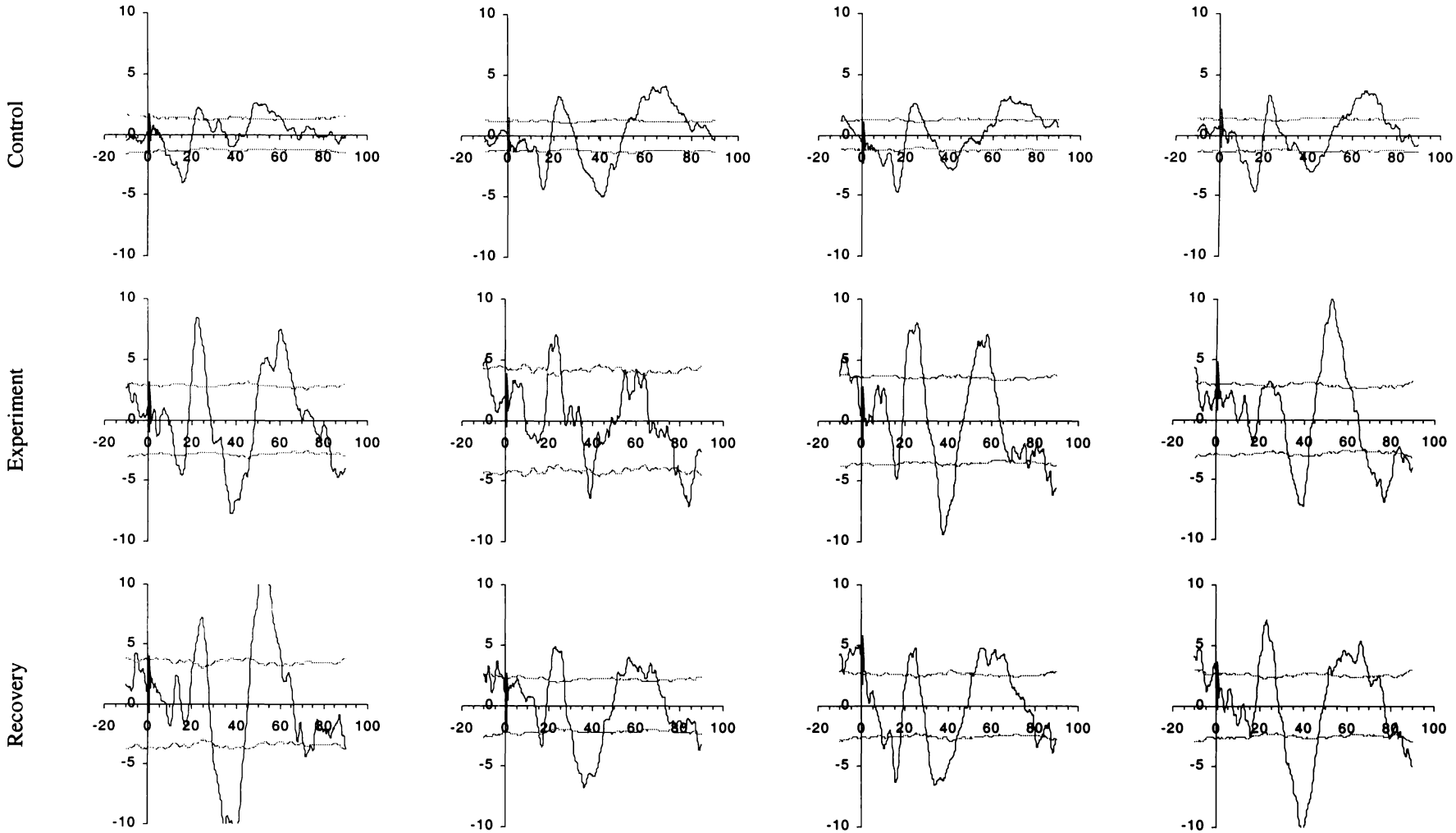


Figure 7.5 (b) – Animal 1 Distraction Experiment Evoked Response Waveforms

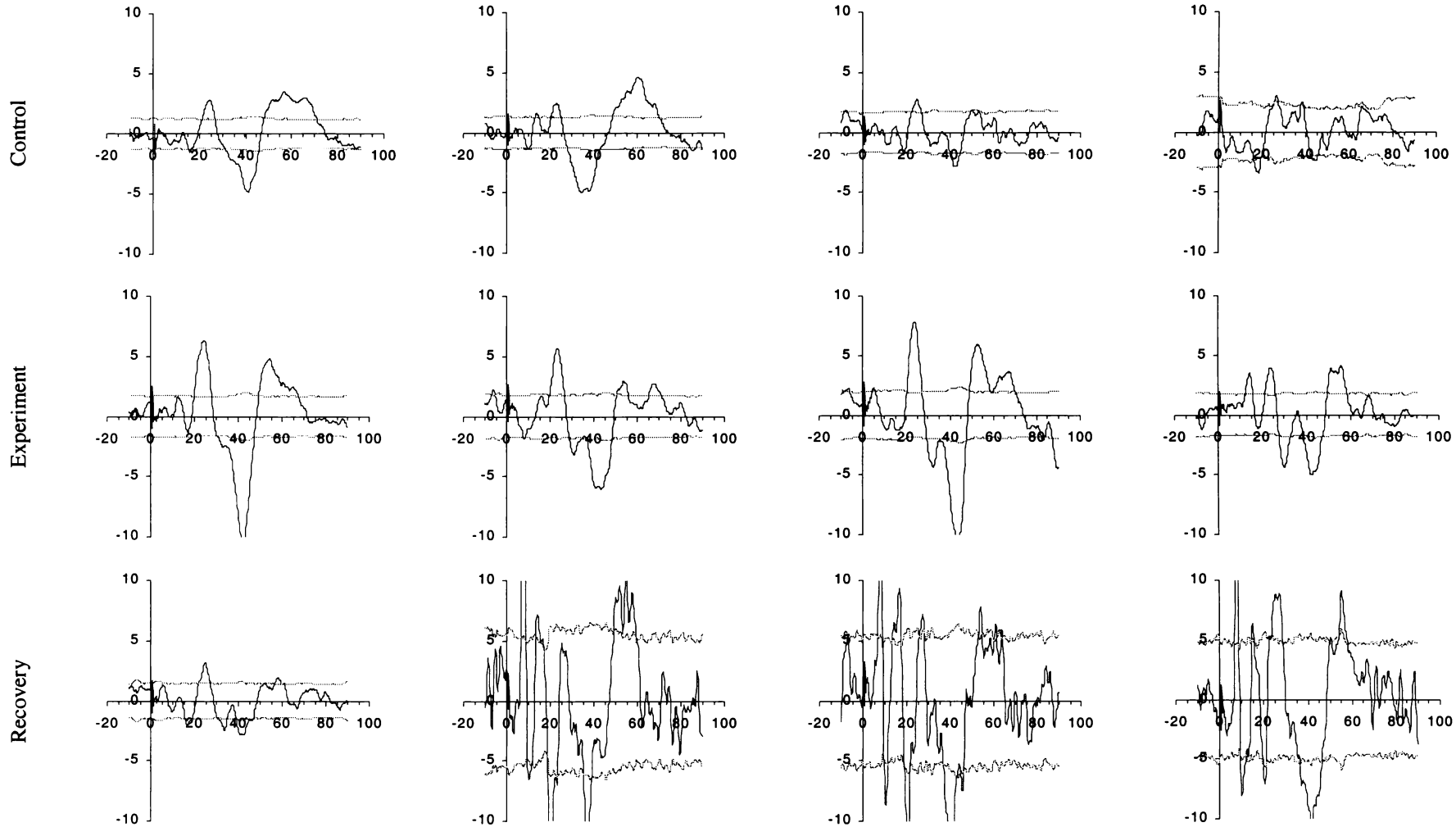


Figure 7.5 (c) – Animal 1 Isolation Experiment Evoked Response Waveforms

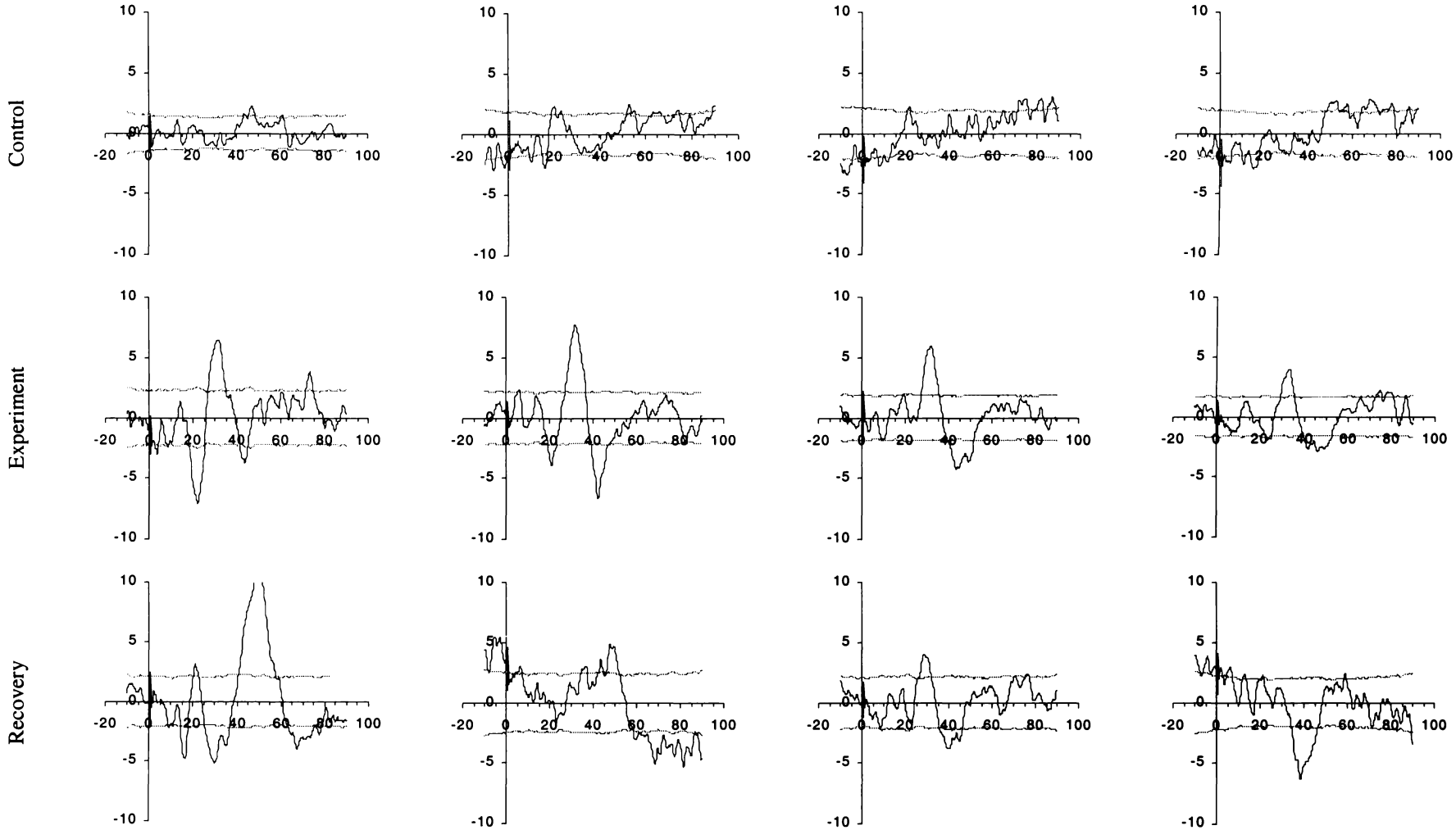


Figure 7.5 (d) – Animal 1 Fear Experiment Evoked Response Waveforms

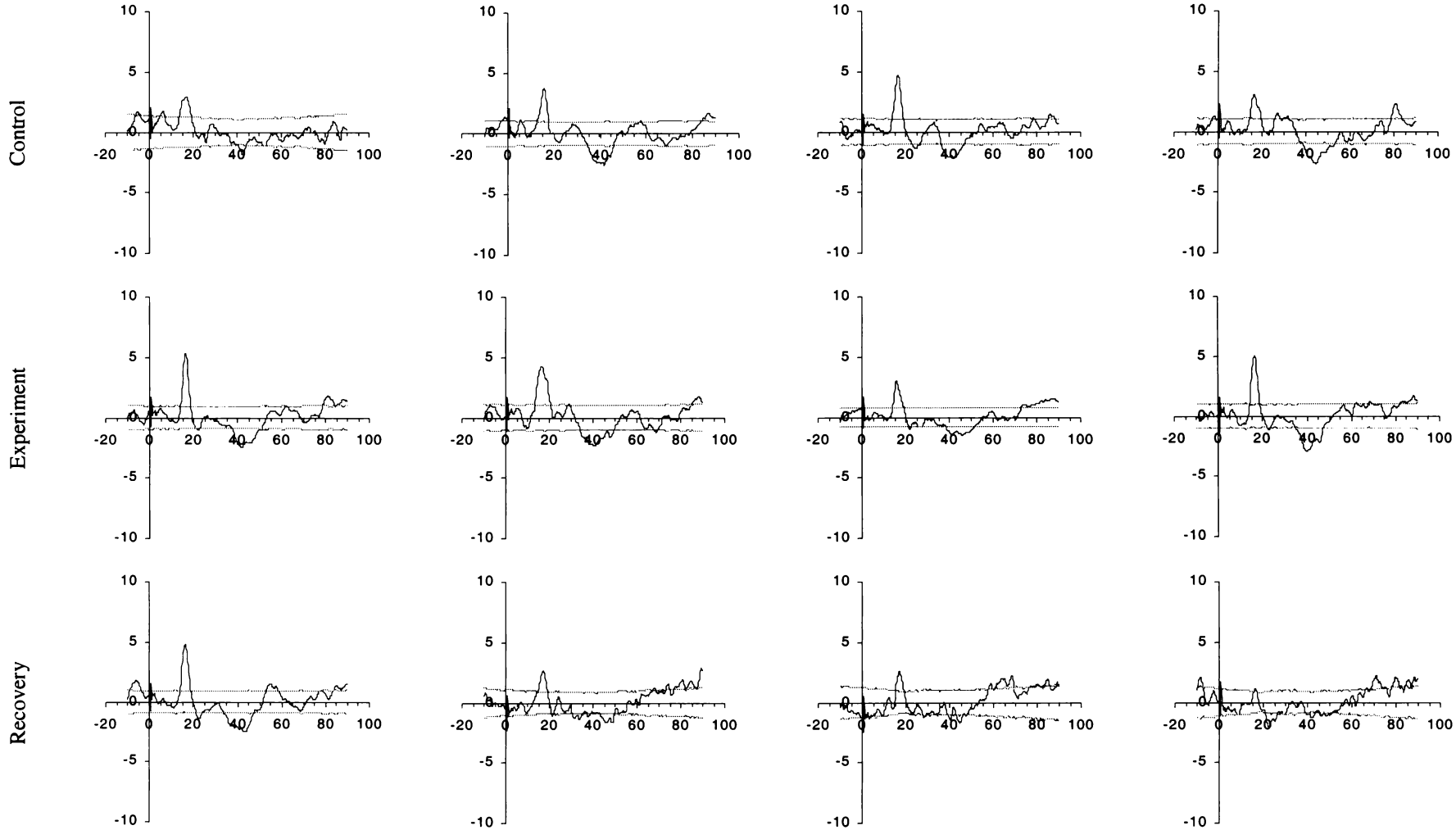


Figure 7.6 (a) – Animal 4 Control Experiment Evoked Response Waveforms

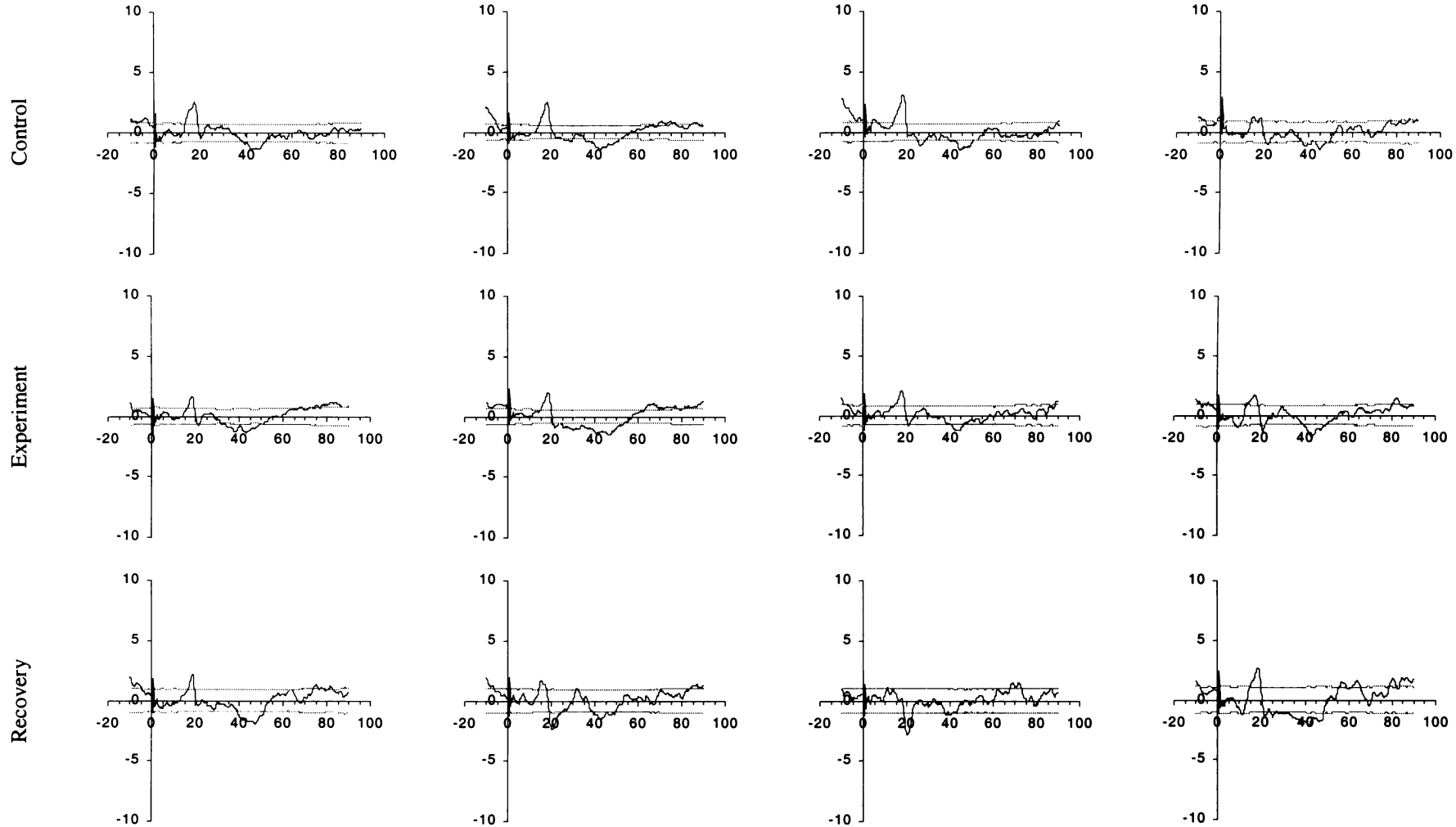


Figure 7.6 (b) – Animal 4 Distraction Experiment Evoked Response Waveforms

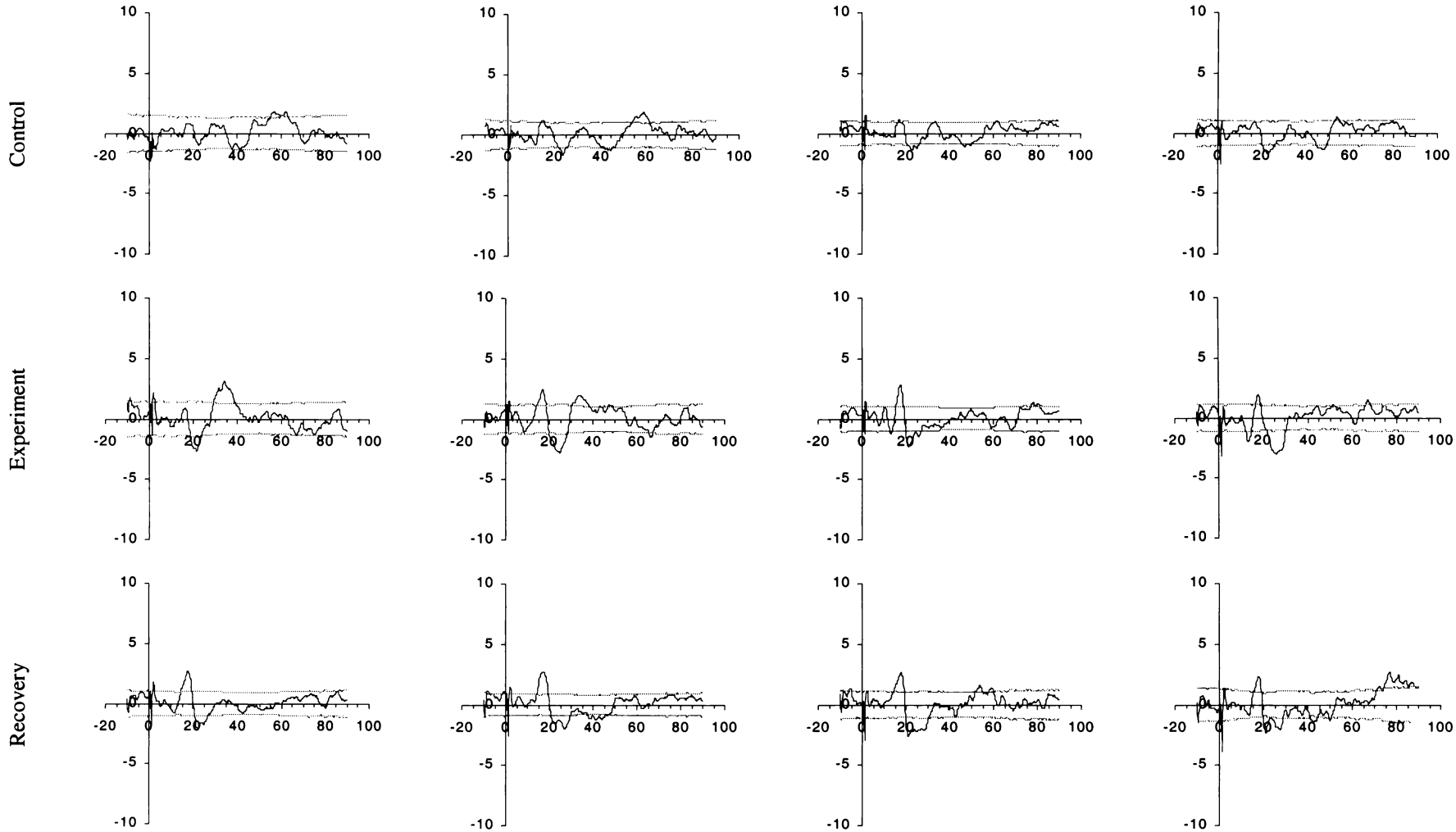


Figure 7.6 (c) – Animal 4 Isolation Experiment Evoked Response Waveforms

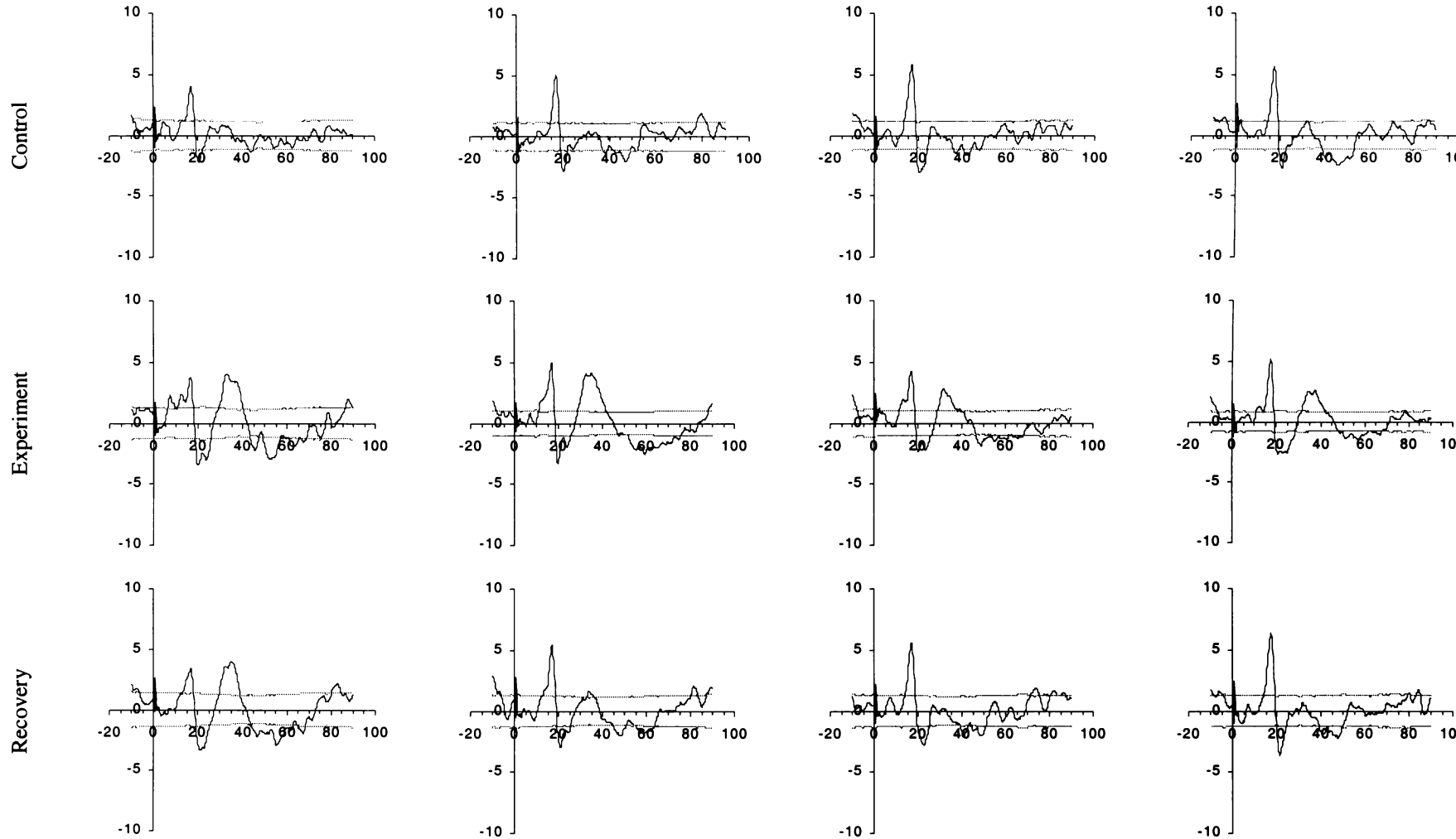


Figure 7.6 (d) – Animal 4 Fear Experiment Evoked Response Waveforms

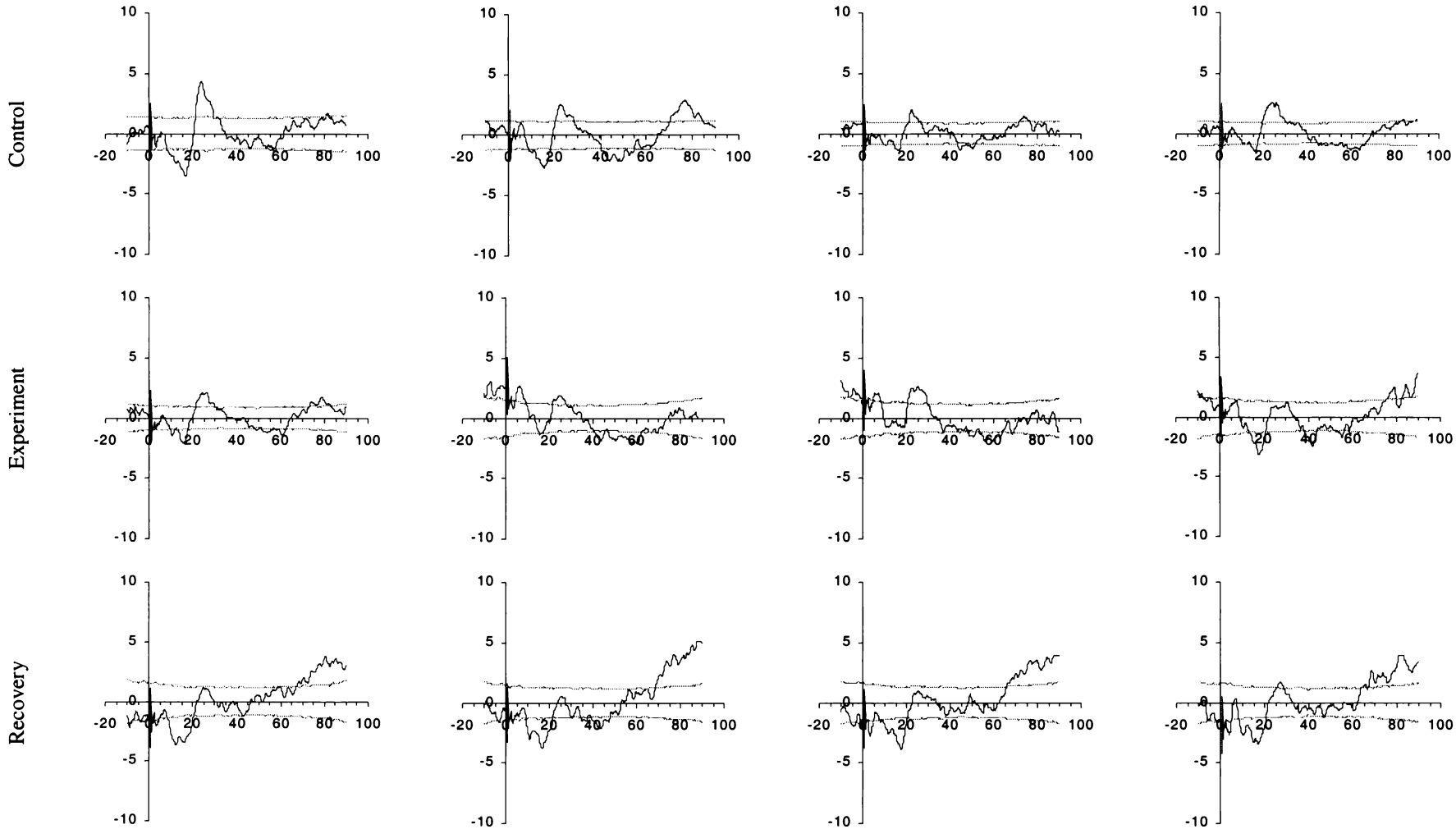


Figure 7.7 (a) – Animal 8 Control Experiment Evoked Response Waveforms

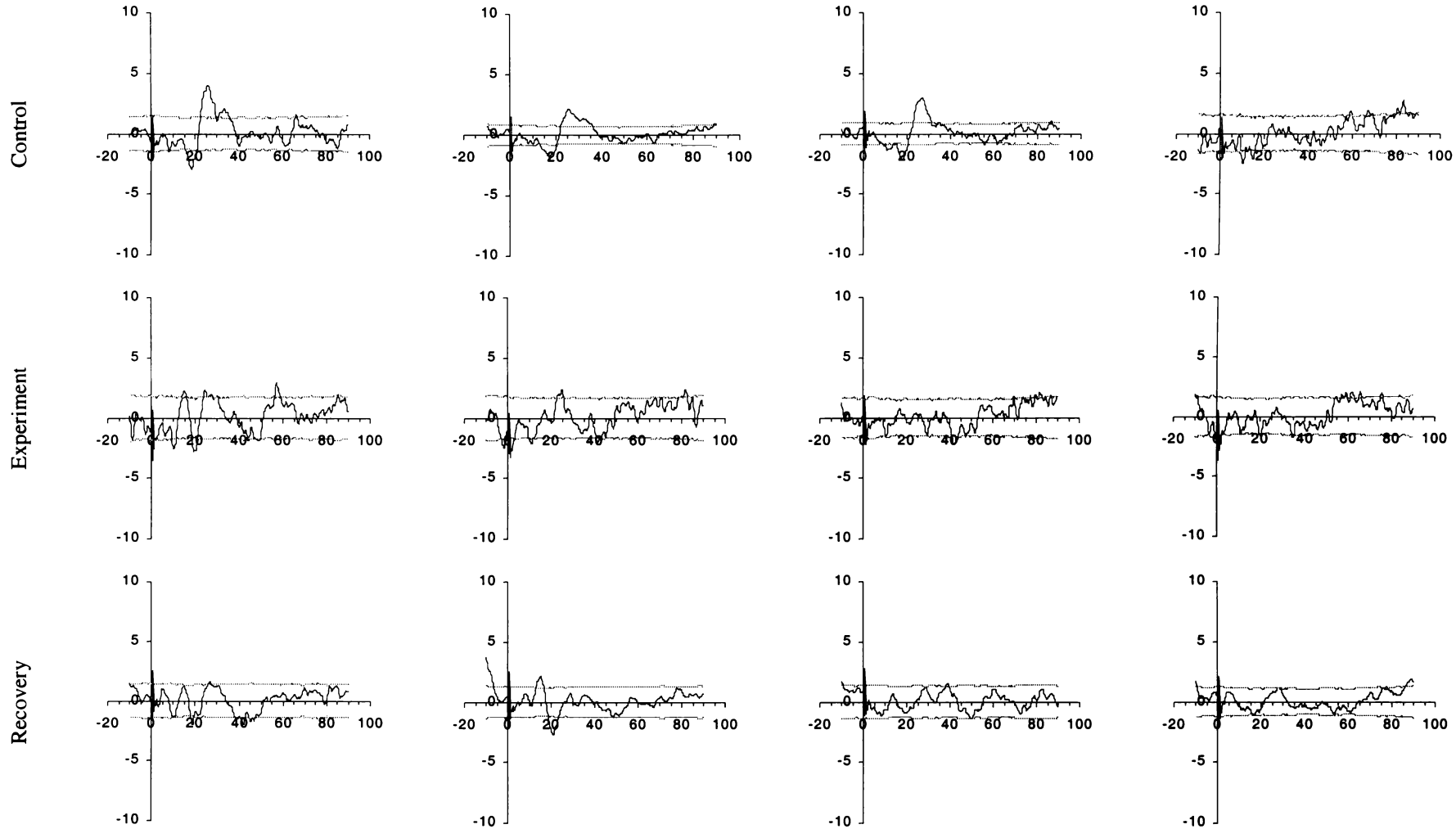


Figure 7.7 (b) – Animal 8 Distraction Experiment Evoked Response Waveforms

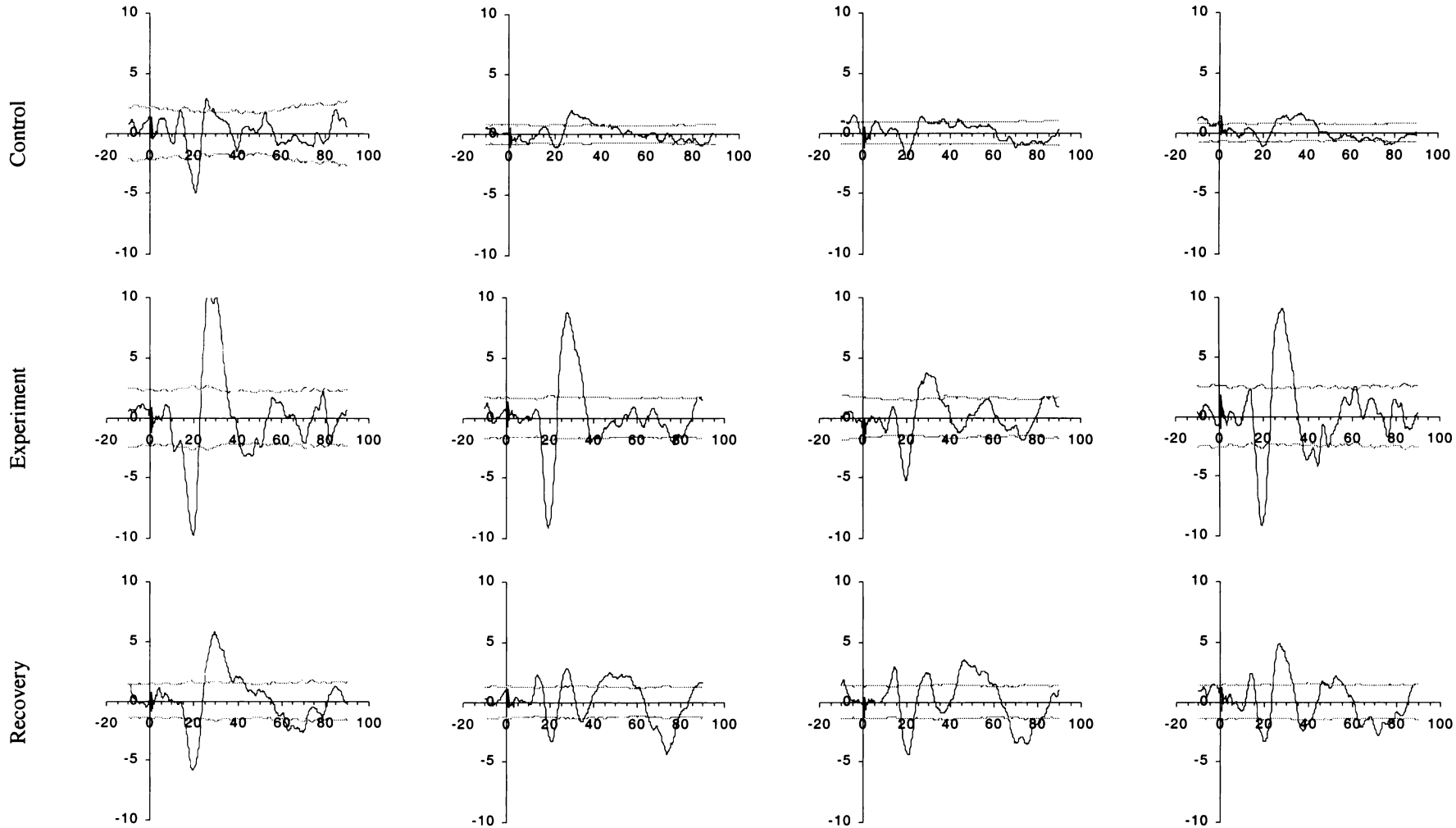


Figure 7.7 (c) – Animal 8 Isolation Experiment Evoked Response Waveforms

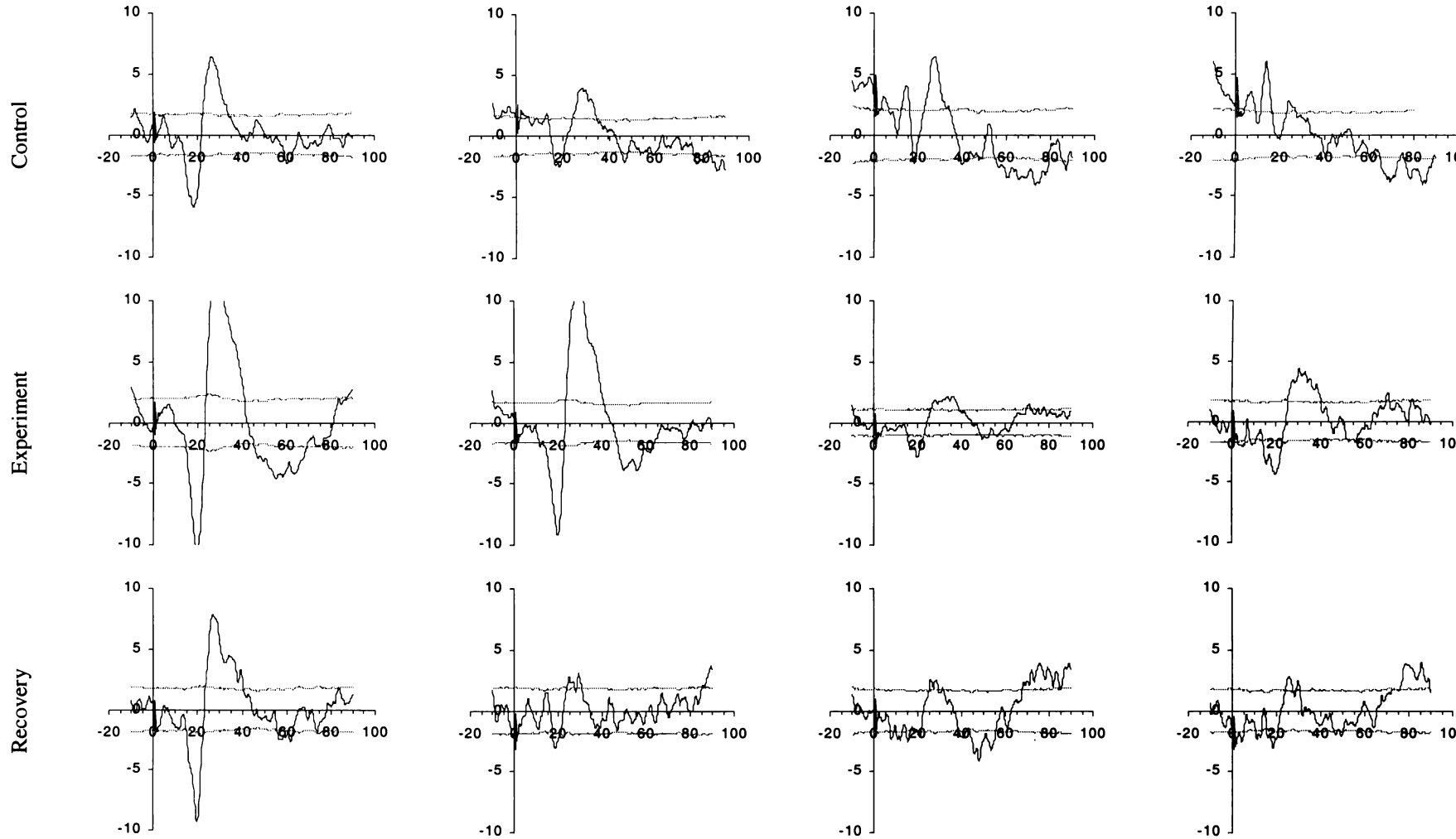


Figure 7.7 (d) – Animal 8 Fear Experiment Evoked Response Waveforms

7.2.1.2 Auditory Evoked Response Analysis

The recorded Auditory Evoked Response waveforms were analyzed using PCA followed by Discriminant Analysis (see Sections 5.4 and 5.5.2 respectively). The recorded waveforms were transformed by the PCA to obtain the components and coefficients required to reconstruct the recorded waveforms. The four coefficients that accounted for most of the variance in the recorded waveforms were kept, while the remaining coefficients and components were discarded. This set of four PCA coefficients were then treated as shape descriptors of the original set of recorded waveforms, and were analyzed using the Discriminant Analysis to search for differences in the Evoked Response waveforms between different periods of the stress experiment.

Figure 7.8 shows the results of the Discriminant Analysis which has arranged each of the 12 recorded waveforms from a single experiment on a two-dimensional plot based on the similarity of waveform shape between the waveforms. On each plot the x-axis corresponds to the first canonical variable derived from the Discriminant Analysis while the y-axis corresponds to the second canonical variable. The symbols used on the plots correspond to each of the three experimental periods so that changes in waveform shape between experimental periods should show up as distinct clusters of the same symbol. Note that because of the inter-animal variability in the recorded waveforms the plots for each animal use a different set of Principal Components as the input to the Discriminant Analysis. Because of this plots from the same animal can be compared directly but plots from different animals cannot.

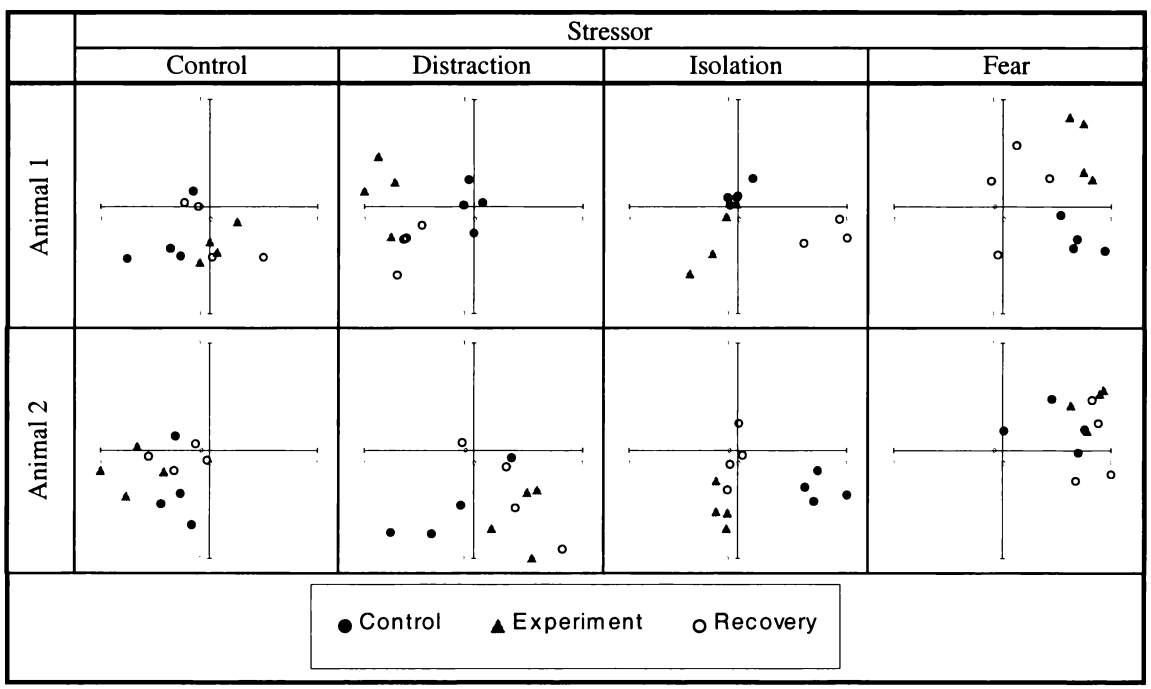


Figure 7.8 (i) – Discriminant Analysis of Evoked Response Waveform by Experimental Period versus Stressor and Animal

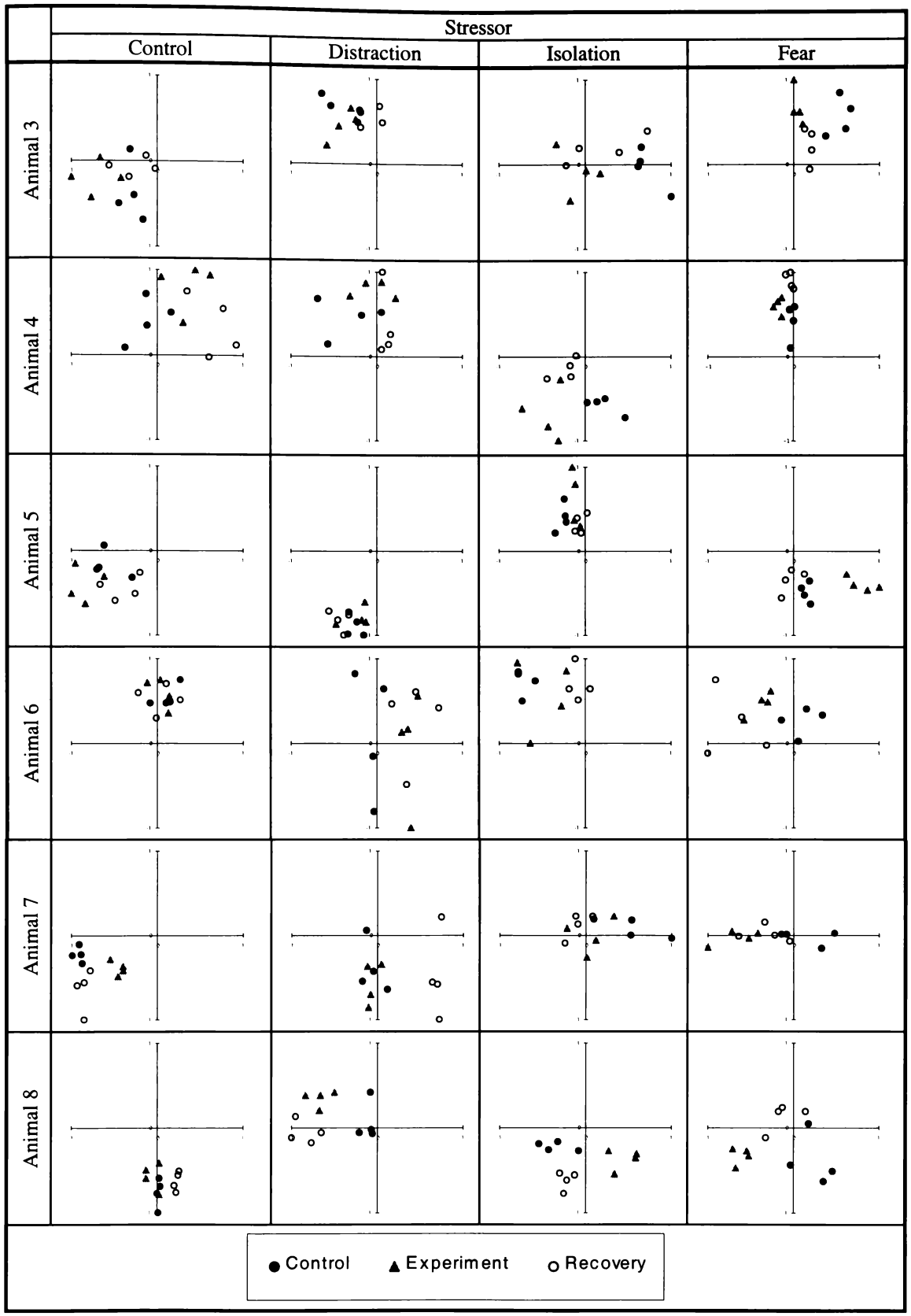


Figure 7.8 (ii) – Discriminant Analysis of Evoked Response Waveform by Experimental Period versus Stressor and Animal

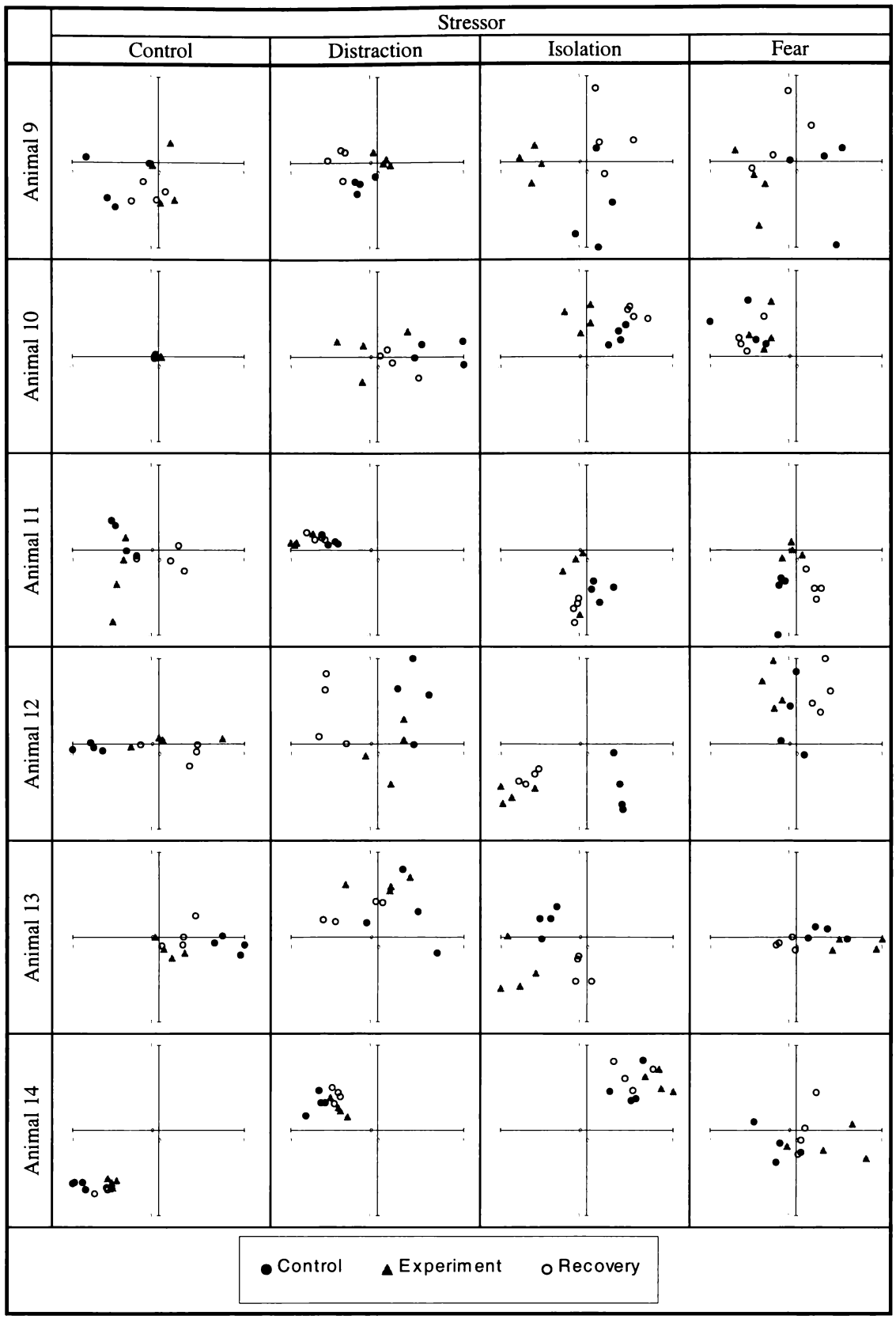


Figure 7.8 (iii) – Discriminant Analysis of Evoked Response Waveform by Experimental Period versus Stressor and Animal

Based on the clustering of each group (Control, Experiment and Recovery) and the distance from other groups, the Discriminant Analysis generates statistics that indicate the probability that any two groups are the same. Table 7.5 summarizes these probabilities for each experiment. Note that the probabilities presented are colour-coded to indicate cases where the differences between categories have been found to be significant: those with a probability below 0.05 are shaded red, those below 0.10 are shaded orange, and those below 0.20 are shaded yellow.

Animal	Experiment	Probability of No Difference between Experimental Periods		
		Control – Experiment	Experiment - Recovery	Control - Recovery
1	Control	0.6072	0.9889	0.6290
1	Distraction	0.0216	0.2569	0.0492
1	Isolation	0.5507	0.2850	0.3646
1	Dog	0.3299	0.1652	0.1744
2	Control	0.4742	0.4309	0.8089
2	Distraction	0.5647	0.9604	0.6559
2	Isolation	0.0198	0.3404	0.0283
2	Dog	0.6346	0.7454	0.6102
3	Control	0.6251	0.9934	0.5316
3	Distraction	0.7949	0.5378	0.5301
3	Isolation	0.3208	0.7101	0.5263
3	Dog	0.0463	0.3311	0.0959
4	Control	0.4013	0.5243	0.2154
4	Distraction	0.6303	0.6748	0.5659
4	Isolation	0.1723	0.2882	0.2353
4	Dog	0.1113	0.1315	0.2241
5	Control	0.5332	0.4166	0.5950
5	Distraction	0.9264	0.6460	0.7239
5	Isolation	0.2532	0.6428	0.1434
5	Dog	0.0719	0.0377	0.5632
6	Control	0.9909	0.9932	0.9896
6	Distraction	0.3090	0.9886	0.2603
6	Isolation	0.7342	0.4278	0.1838
6	Dog	0.4364	0.6339	0.1908
7	Control	0.0261	0.0371	0.3067
7	Distraction	0.9895	0.0191	0.0196
7	Isolation	0.5620	0.9070	0.3992
7	Dog	0.2573	0.7394	0.6003
8	Control	0.5958	0.0720	0.1362
8	Distraction	0.0961	0.3068	0.0564
8	Isolation	0.0708	0.0833	0.1401
8	Dog	0.0896	0.1829	0.2332
9	Control	0.2353	0.8647	0.5422
9	Distraction	0.1860	0.0671	0.1721
9	Isolation	0.0956	0.0660	0.4281
9	Dog	0.2165	0.5770	0.4302
10	Control	0.3997	0.3793	0.6604
10	Distraction	0.2723	0.7818	0.5338
10	Isolation	0.1266	0.0641	0.2875
10	Dog	0.8390	0.8776	0.9753
11	Control	0.6769	0.2022	0.2271
11	Distraction	0.1644	0.4257	0.5739
11	Isolation	0.2286	0.5543	0.2414
11	Dog	0.2506	0.1314	0.0665
12	Control	0.2694	0.5946	0.2188
12	Distraction	0.4397	0.1106	0.0544
12	Isolation	0.0124	0.5581	0.0202
12	Dog	0.3564	0.0589	0.1919
13	Control	0.1038	0.7031	0.1649
13	Distraction	0.6750	0.6598	0.5886
13	Isolation	0.1388	0.0488	0.1270
13	Dog	0.3374	0.1524	0.3070
14	Control	0.0680	0.3432	0.1052
14	Distraction	0.2315	0.6597	0.2819
14	Isolation	0.4784	0.5050	0.9249
14	Dog	0.4253	0.7462	0.7530

Table 7.5 – Probability of No Change in Evoked Response Waveforms between Experimental Periods – Target Animal
(Values are shaded by significance: those below 0.05 are shaded red, those below 0.10 are shaded orange, and those below 0.20 are shaded yellow)

Table 7.5 shows that of the 168 cases (14 animals times 4 experiments times 3 experimental periods) only 29 cases had a probability of no change in the Evoked Response waveform of less than 0.10. Table 7.6 presents a summary of these cases. It is notable that animals 1 and 4 which were presented in the sequences of Figure 7.5 and Figure 7.6 do not show much significance. However, animal 8 does show significant changes in line with the observations of Section 7.2.1.1.

	Control - Experiment	Experiment – Recovery	Control – Recovery	Total
Control	2	2	0	4
Distraction	2	2	4	8
Isolation	4	4	2	10
Fear	3	2	2	7
Total	11	10	8	29

Table 7.6 – Summary of Significant Cases – Target Animal

Table 7.6 shows that most cases occurred in the Isolation experiment and between the Control-Experiment experimental periods.

Figure 7.9 also presents a summary of the data in Table 7.5 but includes all 168 cases. For each experiment it shows where the probabilities of Table 7.5 were grouped. The y-axis is the number of cases (as a percentage) where the probability of similarity is less than the value on the x-axis.

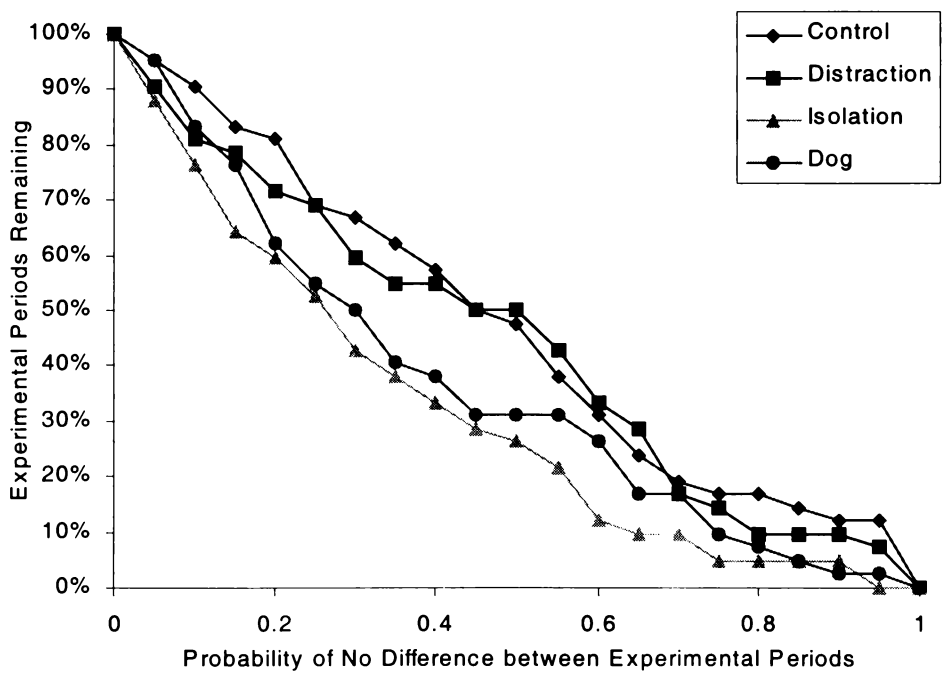


Figure 7.9 – Histogram of Significance of No Difference in Evoked Response Waveforms between Experimental Periods for each Stressor

Figure 7.9 clearly shows that the probability distributions of the four stressors follow different paths. The Control and Distraction stressors tend to zero slower than the Isolation and Fear stressors. This indicates that, on average, the probability values of the Isolation and Fear stressors are lower than the other two stressors. Table 7.7 presents the summary statistics and an ANOVA analysis of the probability distributions.

Stressor	Mean Probability	Standard Deviation of Probability
Control	0.4703	0.26654
Distraction	0.4457	0.31850
Isolation	0.3431	0.26693
Fear	0.3306	0.24480

Stressor	Control	Distraction	Isolation	Fear
Control	1.000			
Distraction	0.997	1.000		
Isolation	0.073	0.230	1.000	
Fear	0.421	0.769	0.963	1.000

Table 7.7 – Summary and Analysis of Probability Distributions for each Stressor

Table 7.7 shows that with better than 0.95 probability the Control and Distraction experiments and the Isolation and Fear experiments have the same probability distributions. There is at best only a 0.769 probability that these four experiments could have the same probability distribution. This confirms the observation as made for Figure 7.9 that the recorded Evoked Response waveforms tend to have greater differences between experimental periods in the Isolation and Fear experiments than in the Control and Distraction experiments.

Although the data processing and statistical analysis do not result in a clear trend for changes in the Evoked Response due to presence of a stressor, there are indications that an effect is present in the analysis of the probability tables produced by the Discriminant Analysis.

Possible reasons for the failure of the data processing and subsequent analysis include the following:

- The data analysis does not treat the waveforms as a time sequence, merely as groups of waveforms
- Differences in inter- and intra-animal variation in response to stressors lead to waveform data containing significant noise

Each of these factors are considered in turn.

The Discriminant Analysis used to analyze changes in the recorded Evoked Response waveforms treats the 12 waveforms it processes at any one time as three separate categories (Control,

Experiment, Recovery), each comprising of four waveforms. It takes no account of the time series relationship between both the categories and the waveforms contained within them. This time series itself contains important information which is ignored. For example, the sequences presented in Figure 7.5 to Figure 7.7 show evidence of recovery, or adaptation, trends that are significant merely by their presence. It is not possible for the Discriminant Analysis to detect these trends, yet their presence has the effect of blurring one category into another, reducing the probability that a statistically significant difference may exist between the categories. The sequence for animal 4 in Figure 7.6 is a good example of this effect - visibly significant changes occurred due to exposure to the dog yet the statistical analysis by Discriminant Analysis in Table 7.5 yields no significant changes (the actual probabilities in this case are Control-Experiment = 0.1113, Experiment-Recovery = 0.1315, Control-Recovery = 0.2241). The probability distribution analysis in Figure 7.9 helps to identify this problem.

The response of an animal to a stressor depends on both prior experience of the particular stressor involved, and the level of short- and long-term stress the animal has been exposed to previously[129]. The response to a stressor can thus vary on a moment to moment basis, both within any particular animal or between any two animals. This problem appears as the adaptation curves of the response sequences and also as a side-effect of the experimental protocols used in this study. For example, during the Fear experiment the group of animals was kept together throughout the experiment. Due to the effect of adaptation it can be expected that the target animal when the group was first exposed to the dog should have a greater stress response than the target animal when the group was last exposed to the dog. This is the reason why the group of 14 animals was split into two groups of 7 animals - to minimize the effect of adaptation in the experiment. The Distraction experiment suffers from the same problem, however the protocol of the Isolation experiment does not since only the target animal was exposed to the stressor in any particular experiment. The Control experiment is a different case since the animals were well used to being held in the pen before any experiments began. In this case adaptation may take the form of a drift in the animals cognitive state due to a lack of stimulus and activity ('boredom'). Since the stress response of any individual animal is continuously changing with time, it becomes very difficult to duplicate any measurement with the aim of improving the accuracy of a model. The effect of adaptation and the inability to repeat a measurement leads to an increase in the noise level of the available data. In the case of the Discriminant Analysis this in turn makes the classification process difficult and leads to low probabilities of class ownership.

There are three areas in which improvements could be made to mitigate these problems. These areas are: the experimental protocol, data acquisition, and data processing.

The most obvious improvement to the experimental protocol would be to increase the number of animals in the experiment. An increased number of animals could be used to two ways. The first way would be to apply the experiment to all the animals in the hope of identifying a pattern, or patterns, in the waveforms recorded. Logistically this method would be time-consuming and may not yield any improvement in understanding. Perhaps a better way to utilize the animals would be

to have many small groups (say three animals per group), instrument only one animal in the group, and only ever expose the group once to any stressor. This method would ensure that the instrumented animal never had the opportunity to adapt to the stressor before recording took place. In this way the extra animals in the experiment act as 'packing' to maintain a social group for the target animal while avoiding adaptation.

As an alternative to the stressors used in the experiment, anxiogenic and anxiolytic drugs could be used to produce a repeatable anxiety state in the target animal. Caffeine has been extensively used as an anxiety agent in animal studies, including sheep[130]. A dosage rate of 5mg/kg by body weight delivered intramuscularly as a sterile solution is reported as producing mild anxiogenesis but is well under the known toxicity range of caffeine (LD50 > 250mg/kg). As an alternative, Zolpidem[i] is an anti-anxiety agent, and is mildly anxiolytic if administered at a dose of 0.5mg/kg by body weight in sterile solution, intramuscularly, but not within toxicity ranges (LD50>100mg/kg). Either of these agents would allow the same animal to be tested multiple times in a repeatable manner and would easily allow multiple animals to be tested simultaneously.

Data acquisition could be improved to obtain more waveforms in each experiment, providing better time resolution of changes in the Evoked Response waveform and allowing such features as recovery curves to be better identified. The easiest way to improve the acquisition would be to acquire waveforms using a smaller number of stimuli and then average these waveforms to improve the Signal-to-Noise Ratio (SNR) while obtaining better time resolution. Currently each of the 12 acquired waveform are formed from 512 stimuli. If 48 waveforms were acquired, each comprising of 64 stimuli, the time resolution would be improved by a factor of four, and by averaging each group of four waveforms, the current SNR would still be achieved. If the acquired waveforms were numbered 0, 1, 2, 3, ... then the first output waveform would be comprised of waveforms 0, 1, 2, and 3. The next output waveform would be comprised of waveforms 1, 2, 3, 4 and so on. Using this method, if 48 waveforms were initially acquired, 44 outputs waveforms would be obtained. Taken at the extreme, this overlap-add method could be applied with individual stimuli: apply stimuli continuously and record a waveform for each stimuli, then form the output from the average of the 512 (for instance) waveforms surrounding each individual waveform. Using the current stimulus rate of 8Hz, approximately 28800 waveforms would be acquired in the hour long experiment. If each group of 512 waveforms was averaged, as is done currently, 28288 output waveforms would be obtained, each with the same SNR as currently obtained, but with a time resolution of 125ms instead of the current 5min. This would enable changes in the Evoked Response to be observed in fine detail, however memory requirements and data processing complexity would also increase dramatically.

The current processing methods would still be applicable for a moderate increase in the number of waveforms or the proposed changes to the experimental protocol but the limitations already identified would still be present. However, with the large number of waveforms generated by the overlap-add method with individual stimuli, the current methods would be unwieldy and slow and

[i] G.D. Searle & Co., 5200 Old Orchard Road, Skokie, IL 60077, USA

alternative methods would have to be used. The Principal Components Analysis (PCA) should still be suitable for the large number of waveforms if the components search was terminated after only the first few major components were found (the NIPAL[131] algorithm is suitable for this), however the Discriminant Analysis would need to be replaced by an algorithm that took advantage of the extra information contained in the time sequence of the data. With the high time resolution obtained by the overlap-add method the algorithms used for Evoked Response peak tracking during anesthesia[132] would be applicable if suitable peaks to track could first be manually identified.

In summary, there is considerable room for improvement in experimental protocol, data acquisition, and data processing. However, each improvement has a varying ease of implementation. The more advanced data processing techniques depend on an expanded data acquisition and storage capability for the FRPM, possibly by the use of miniature hard-drive technology (see Section 3.9.2), at the probable expense of greater size, weight, and shortened battery life for the FRPM. If this is undesirable, the most promising direction of improvement using the current FRPM capability would be the use of anxiogenic and anxiolytic drugs to produce repeatable anxiety states in the target animal. This approach would allow repeated measures on an individual animal leading to better characterization of that individual's response at discrete levels of stress. This approach would also allow a minimum number of animals to be used experimentally, an ethical consideration in animal experimental design.

7.2.2 Heart Rate

Electrocardiogram recordings were acquired from the target animal every 5min for 30s, producing a total of 12 recordings for each experiment. The recorded waveforms were then analyzed by Fourier analysis to extract the mean heartrate from each recording (see Section 5.6). Figure 7.10 shows the heartrate profiles from each of the individual animals.

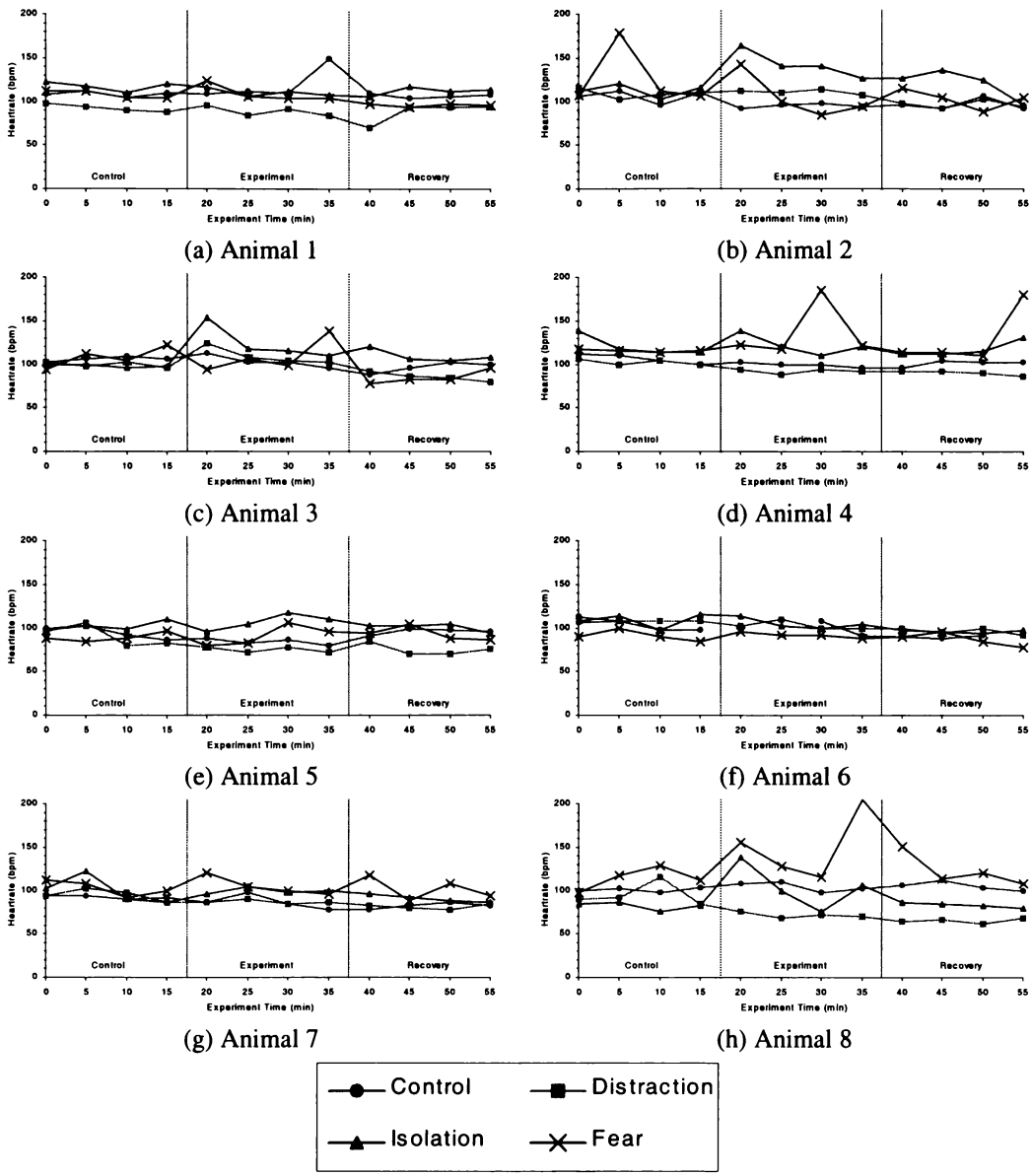


Figure 7.10 (i) – Individual Animal Heartrate Profiles versus Experiment Time for each Stressor

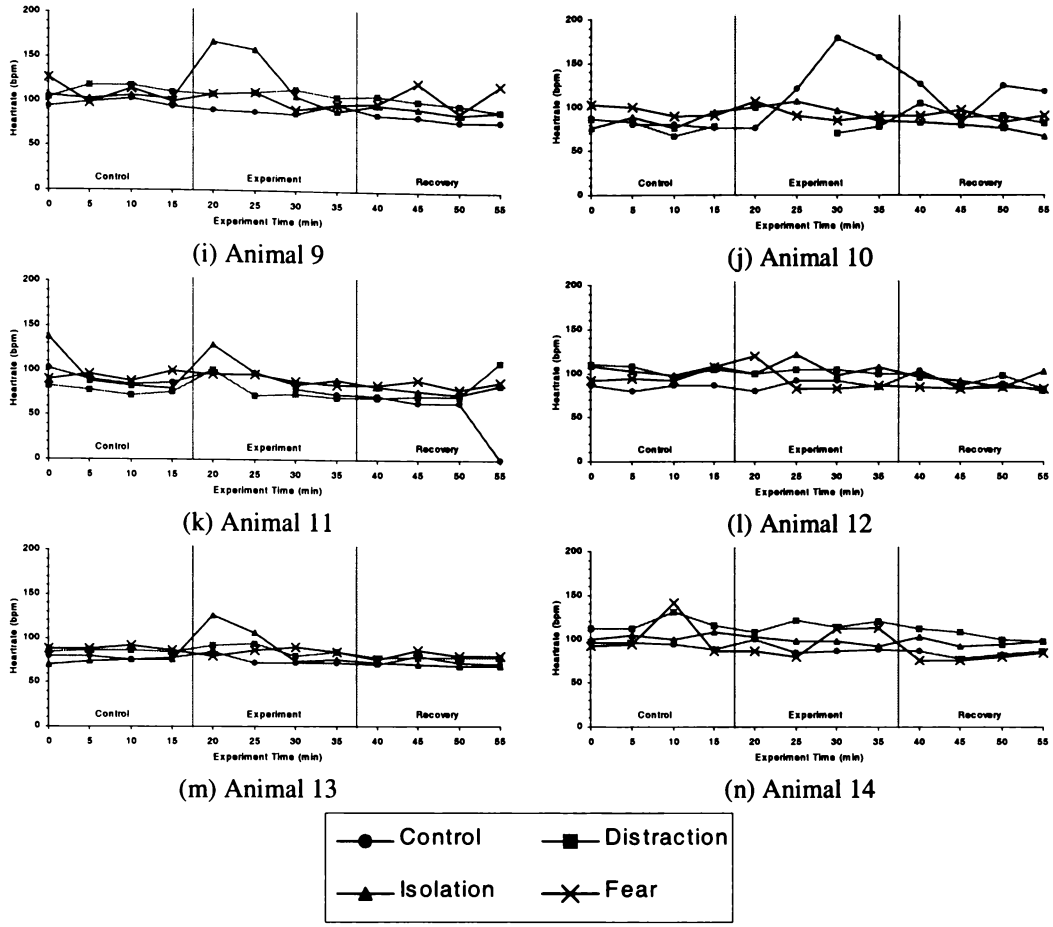


Figure 7.10 (ii) – Individual Animal Heartrate Profiles versus Experiment Time for each Stressor

The heartrate profiles from the animals are quite similar within experiments. Heartrate response to stress generally has the tachycardia form, with no instances of bradycardia observed. Several of the animals (4 and 11, for example) show an increased heartrate at the beginning and end of experiments, probably due to handling. Missing data points are generally due to electrode malfunction or electrical interference. Animals 1 and 10 show large heartrate increases during the Control experiment due to environmental interference, outside the control of the experiment.

The Distraction experiment appears to have negligible effect on the sheep although animals 3 and 11 may have exhibited some response.

Isolation caused a significant heartrate elevation in several animals, in particular animals 2, 3, 8, 9, 11 and 13. This increase was most likely to have been mainly due to the increased physical activity brought about by the pacing and general agitation of the isolation. The Isolation stress is significant in that the increase in heartrate is typically large and sustained whereas the other stressors typically only produced transient heartrate increases.

Both animals 4 and 8 show marked cardiac responses when the dog is present during the Fear experiment with animal 8 showing peaks in heartrate both when the sheep were first exposed to the dog and when the dog was removed. This is also observed in animals 2 and 7 although to a

lesser degree. Animal 8 was significantly affected by both the Isolation and Fear stressors, suggesting this animal coped poorly with the presented stressors. At the other extreme animals 1, 6, and 10 showed little, if any, heartrate response to any of the stressors.

Figure 7.11 shows the average heartrate profiles of the 14 animals for each of the four experiments.

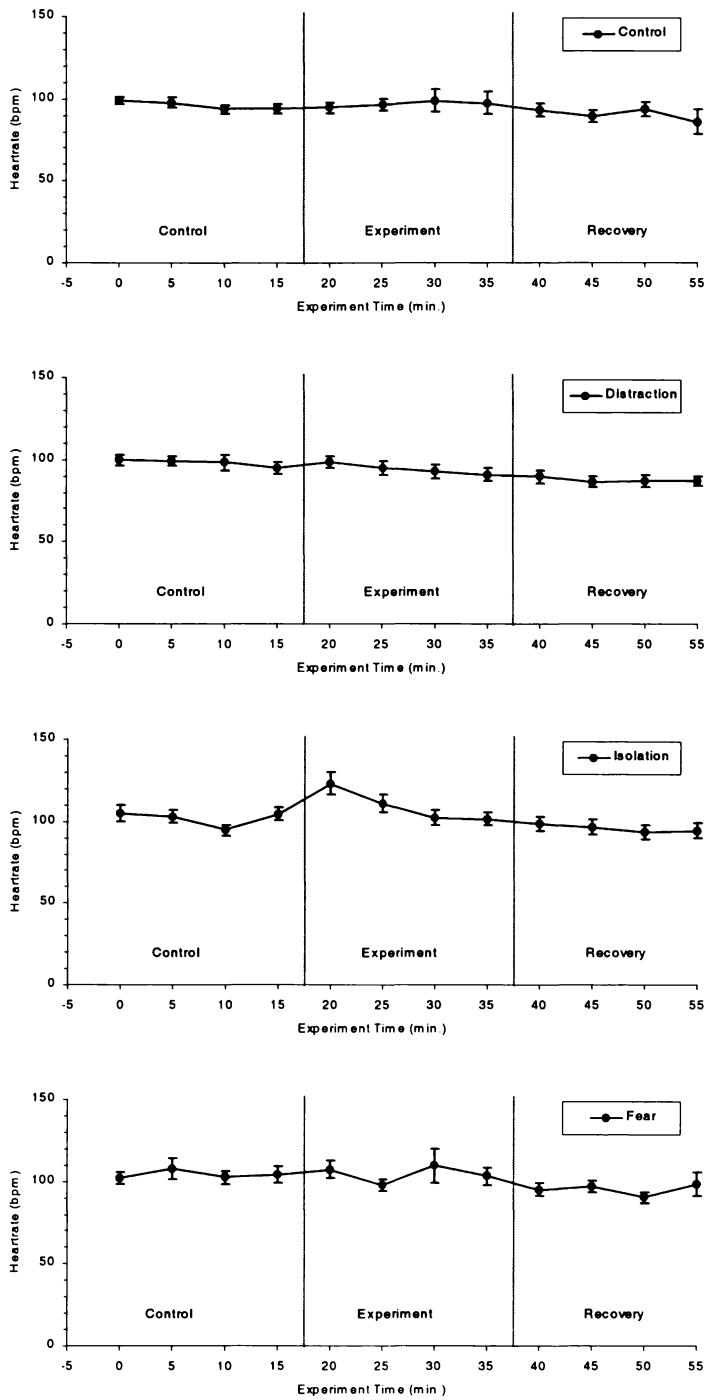
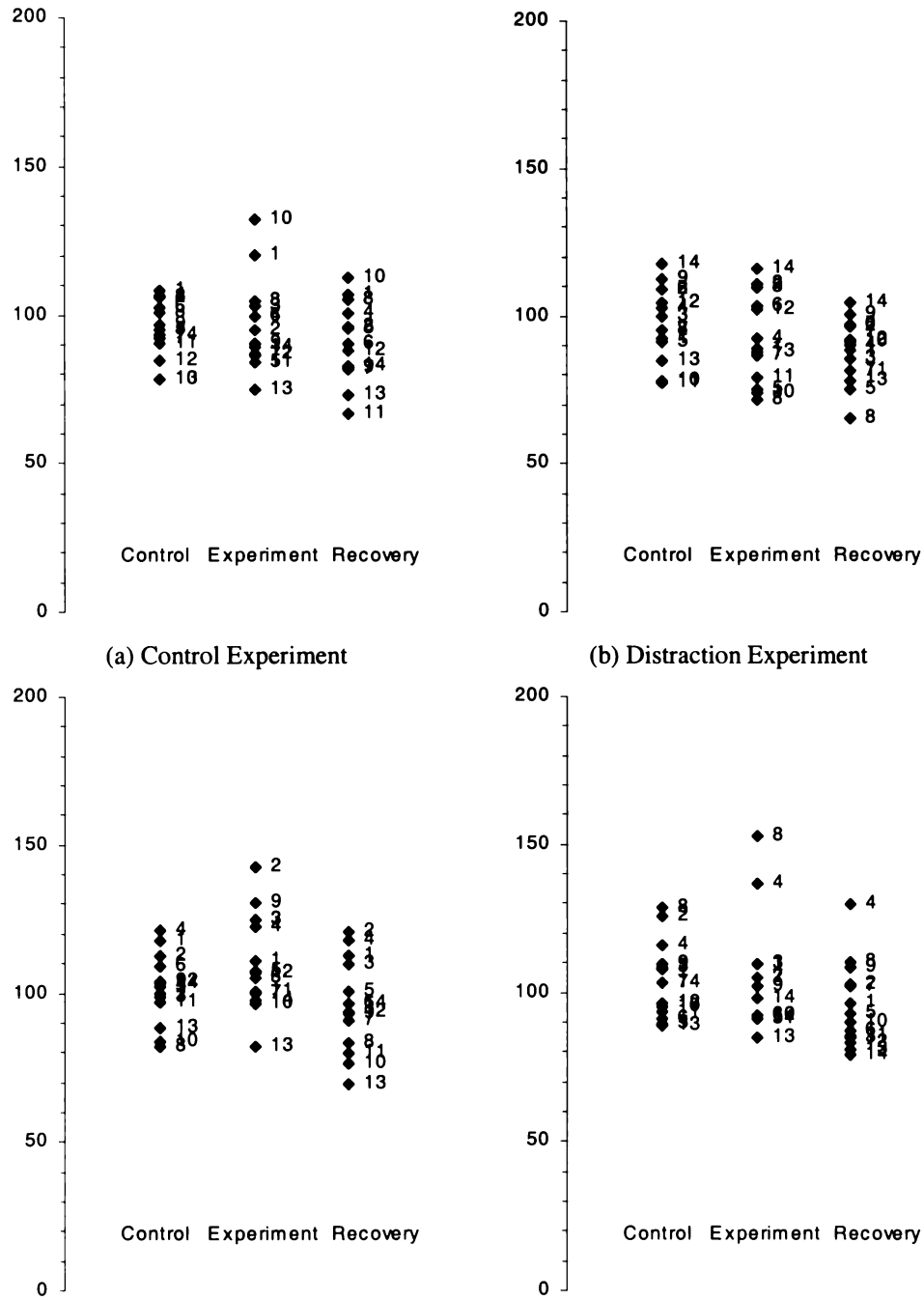


Figure 7.11 – Average Heartrate Profile versus Experiment Time for each Stressor (Mean± SEM)

Figure 7.11 shows that the average heartrate of the animals remained relatively constant for the Control and Distraction experiments but that Isolation caused a marked change in heartrate, possibly due to pacing behavior, during the experiment period. Exposure to the dog caused little change in heartrate possibly due to the animals standing still while the dog was present.

Figure 7.12 shows the mean heartrate distribution by animal for each experiment. Each point is the mean heartrate of the four measurements from each period.



(a) Control Experiment

(b) Distraction Experiment

(c) Isolation Experiment

(d) Fear Experiment

Figure 7.12 – Mean Heartrate Distributions versus Experimental Period by Animal

Table 7.8 shows the significance of the change in the mean heartrate between experimental periods for each experiment based on ANOVA analysis.

Experiment	Experimental Periods	Probability of No Change between Periods
Control	Control – Experiment	NS
	Experiment - Recovery	NS
	Control - Recovery	NS
Distraction	Control - Experiment	NS
	Experiment - Recovery	0.048
	Control - Recovery	0.001
Isolation	Control - Experiment	0.087
	Experiment - Recovery	0.001
	Control - Recovery	NS
Fear	Control - Experiment	NS
	Experiment - Recovery	0.046
	Control – Recovery	0.087

Table 7.8 – Significance of No Change in Heartrate between Experimental Periods
(Only significance values below 0.10 are presented, NS signifies an insignificant result)

Table 7.8 shows that there is no significant change in the mean heartrate of all of the animals during the Control experiment. The Recovery period of both the Distraction and Fear experiments shows a significantly different heartrate from the other two periods, possibly due to a slowing of heartrate related to the removal of the stressor. The Isolation experiment shows a significant difference while the target animal is isolated, however the Control and Recovery periods are not significantly different. The Dog experiment shows little change in the heartrate profile, however the heartrate variability during the Experiment period is greater than during either the Control or Recovery periods. This is possibly due to the target animal remaining motionless during the period when the dog was present (see Section 6.4.3.4) but with brief startle reactions in response to activity by the dog.

The results presented in Table 7.8 are somewhat similar to those in presented in Table 7.6 in relation to significant differences between AER waveforms recorded in the four experiments. Both tables suggest that significant differences exist in the Experiment period of the Distraction, Isolation, and Fear experiments, but few differences were found in the Control experiment. Both tables also appear to be susceptible to the 'blurring' between categories discussed in Section 7.2.1.2 in respect to the analysis of the AER waveforms. In the case of the heartrate data this 'blurring' is caused by the ANOVA analysis not taking into account the sequential nature of the data, in a similar way to that of the Discriminant Analysis used in the analysis of the AER waveforms. This suggests that heartrate profiles as presented in Figure 7.11 may be more useful in the detection of patterns in the cardiac response to stress than the use of ANOVA analyzed results.

7.2.3 Behavior

The behavior of both the target animal and a sample of the rest of the group was made continuously during each experiment (see Section 6.4.4). The behavior data was analyzed by calculating the average behavior of the 14 animals in the three experimental periods of each experiment. The average behavior of the 14 animals is presented in Figure 7.13.

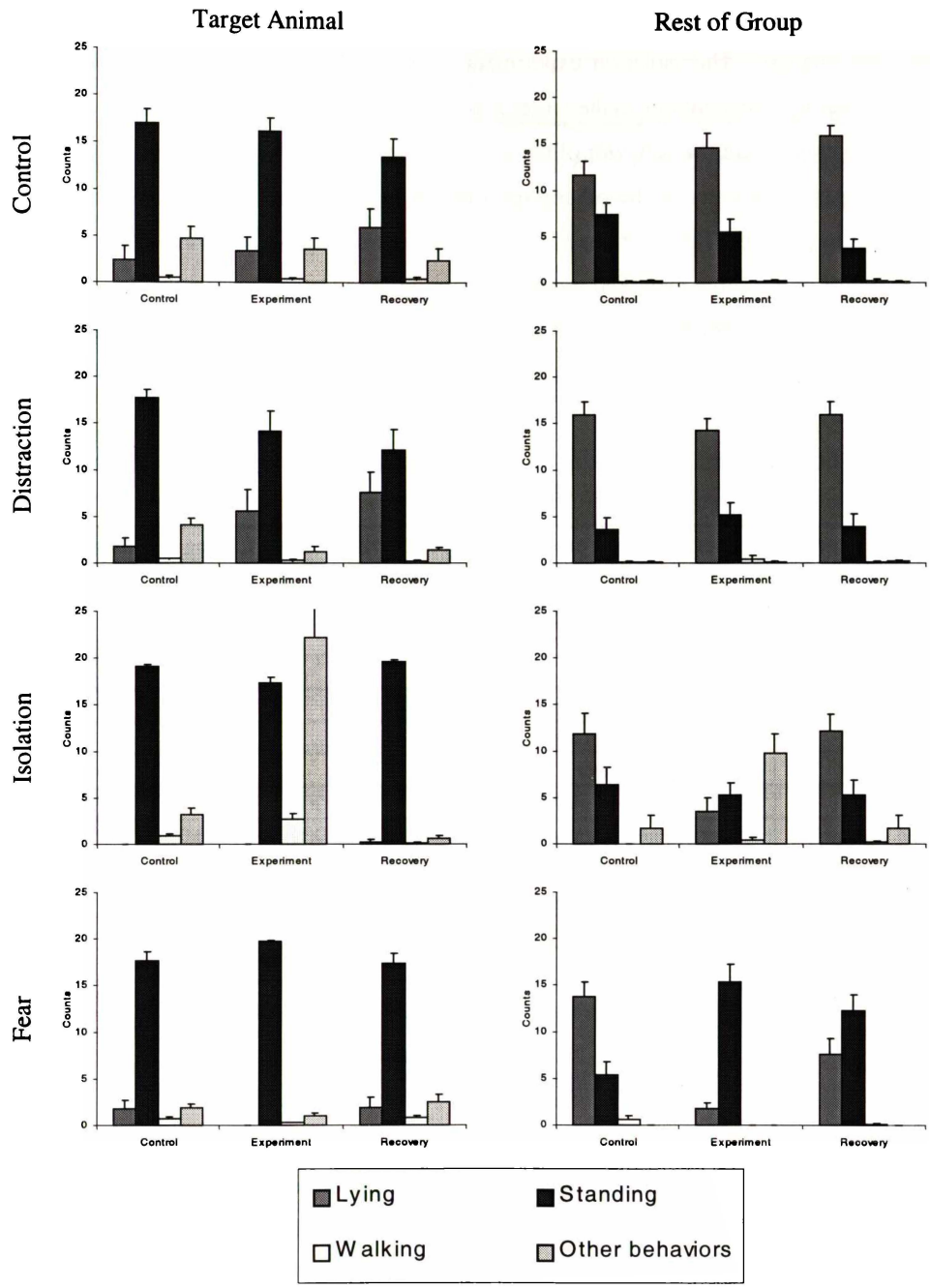


Figure 7.13 – Averaged Animal Behavior versus Stressor and Experimental Period (Mean± SEM)

There is an obvious discrepancy between the lying and standing behavior of the instrumented animal and the rest of the group that is present in all of the experiments. This lead to an investigation of the effect of carrying instrumentation on the behavior of an animal and is discussed in Section 7.3.

Despite this discrepancy, the behavior of the instrumented animal shows good relation to the experiment and the experimental period. The Control period shows little change in behavior of both the target animal and the control group throughout. The Distraction experiment shows a steady trend towards lying, possibly due to fatigue, in the target animal while the control group shows little change. The Isolation experiment shows a marked increase in pacing and other behaviors (mainly vocalization) in the target animal during the experimental period. Note that for the Isolation experiment the behavior of the control group during the Experiment period cannot be readily compared with any of the other experimental periods since during this time the animals were let out to pasture. The Fear experiment shows little change in behavior of the target animal but the control group show a marked transition from lying to standing while the dog is present and during the Recovery period. Table 7.9 and Table 7.10 summarize the significance of behavior changes in the target animal and the control group respectively for each of the experimental periods.

Experiment	Experimental Periods	Probability of No Change in Behavior			
		Lying	Standing	Walking	Other
Control	Control - Experiment				
	Experiment - Recovery				
	Control - Recovery				
Distraction	Control - Experiment				0.003
	Experiment - Recovery				
	Control - Recovery				0.006
Isolation	Control - Experiment		0.004	0.002	0.008
	Experiment - Recovery		0.001	0.001	0.003
	Control - Recovery				
Fear	Control - Experiment				
	Experiment - Recovery			0.047	
	Control - Recovery				

Table 7.9 – Significance of Behavior Changes between Experimental Periods – Target Animal
(Only significance values below 0.10 are presented)

Table 7.9 shows significant changes in the behaviors of the target animal during the Experiment period of the Isolation experiment as compared to the Control and Recovery periods. The Control period of the Distraction experiment also shows a significant difference in other behaviors due mainly to defecation and body shake by the animal. The presence of the dog in the Fear experiment also evokes a change in behavior between the Experiment and Recovery periods.

Experiment	Experimental Periods	Probability of No Change in Behavior			
		Lying	Standing	Walking	Other
Control	Control - Experiment				
	Experiment - Recovery				
	Control - Recovery				
Distraction	Control - Experiment				
	Experiment - Recovery				
	Control - Recovery				
Isolation	Control - Experiment	0.008		0.089	0.005
	Experiment - Recovery	0.006			0.005
	Control - Recovery				
Fear	Control - Experiment	0.001	0.001		
	Experiment - Recovery	0.013			
	Control - Recovery	0.022	0.015		

Table 7.10 – Significance of Behavior Changes between Experimental Periods – Rest of Group
(Only significance values below 0.10 are presented)

The differences in behavior in Table 7.10 during the Isolation experiment are due entirely to the group being released to pasture during the Experiment period - note that the behavior between the Control and recovery periods is not significantly different. The presence of the dog during the Fear experiment caused a significant change in behavior as any lying animals stood up to face the dog.

For the target animal, Table 7.9 shows similarities with both Table 7.6 (relating to significance of changes in the AER with stress) and Table 7.8 (relating to significance of changes in heartrate with stress). All three tables indicate that there is little change from baseline conditions throughout the Control experiment and that significant changes occur in the Distraction experiment only between the Control and Recovery periods. All three tables also suggest that the Experiment period of the Isolation experiment induces the most change in the target animal. However in the Fear experiment, while the heartrate results show a significant change in heartrate while the dog is present during the Experiment period, both the AER and behavioral results show little significant change.

The similarities between these three summary tables, obtained by largely independent measurements, suggests the measurement techniques are observing a common causative source and gives confidence to the individual results obtained by each technique.

7.2.4 Comparison of the Auditory Evoked Response, Heartrate, and Behavior as Indicators of Stress

It is difficult to directly compare the AER results with those of heartrate and behavior as the AER results are intra-animal only, whereas the heartrate and behavior results are averaged over the entire group of animals. However, comparison based on general trends is still possible.

The results from each of the AER, heartrate, and behavior measurements are summarized in Table 7.11. Changes in the AER are represented by the average significance of all animals that produced a significant change for the particular experimental period, the number in brackets being the number of animals that had a significant change (this is based on Table 7.5 and Table 7.6). Significance of changes in heartrate is taken directly from Table 7.8. Changes in the walking behavior of the target animal (from Table 7.9) is taken as representative of behavior.

Experiment	Experimental Periods	Significance of Change		
		AER	Heartrate	Behavior (Walking)
Control	Control - Experiment	0.047 (2)		
	Experiment - Recovery	0.055 (2)		
	Control - Recovery			
Distraction	Control - Experiment	0.059 (2)		
	Experiment - Recovery	0.043 (2)	0.048	
	Control - Recovery	0.045 (4)	0.001	
Isolation	Control - Experiment	0.050 (4)	0.087	0.002
	Experiment - Recovery	0.066 (4)	0.001	0.001
	Control - Recovery	0.024 (2)		
Fear	Control - Experiment	0.069 (3)		
	Experiment - Recovery	0.048 (2)	0.046	0.047
	Control - Recovery	0.081 (2)	0.087	

Table 7.11 – Significance of Changes between Experimental Periods (Only significance values below 0.10 are presented)

Table 7.11 shows a trend towards an increase in significant changes in the Isolation and Fear experiments using all three stress measurement techniques. Of the four experiments, Isolation shows the most significant changes using all three techniques. Although the AER technique shows significant changes in the Control experiment, this is for only a small number of animals and is not collaborated by either heartrate or behavioral measurement. These observations suggest the experiments can be ranked in order of 'stressfulness' from Control (lowest stress), to Distraction, to Fear, and finally to Isolation (highest stress).

The general agreement between all three largely independent measurement techniques suggests that each of the AER, heartrate, and behavior measurement techniques can be used to observe the same variable, stress.

7.3 Effect of Instrumentation on Behavior

The results of the behavior analysis from the Evoked Response work (Section 7.2.3) showed a clear preference for the target (instrumented) animal to remain standing while the rest of the animals in the group were lying. To investigate this effect a study was undertaken to assess what facet of the instrumentation caused this behavior difference.

Individual animals in a group of 14 were equipped with different combinations of the instrumentation used to conduct the Evoked Response experiment. These animals were then placed back in the group and the behavior of both the target and group animals recorded over a 60min period with no exposure to a stressor other than the wearing of the instrumentation. The protocol for this experiment was effectively the same as that of the Control experiment in the Evoked Response study except that the amount of instrumentation the target animal carried was variable and no Evoked Response recordings or heartrate measurements were made.

Four different combinations of instrumentation were carried by the target animals, each designed to isolate the particular part of the instrumentation causing the behavioral difference observed in the initial studies. The four combinations applied to the target animal were:

1. FRPM and girth strap only. Neither electrodes or clicker present
2. FRPM and electrodes (FRPM fastened to animal by Velcro only). Neither girth strap or clicker present
3. FRPM, girth strap, and electrodes. No clicker present
4. Full Evoked Response instrumentation on animal: FRPM, girth strap, electrodes, and clicker in animal's ear and clicking

The results from these instrumentation combinations are presented in Figure 7.13 while Table 7.12 presents the results from an ANOVA analysis on this data.

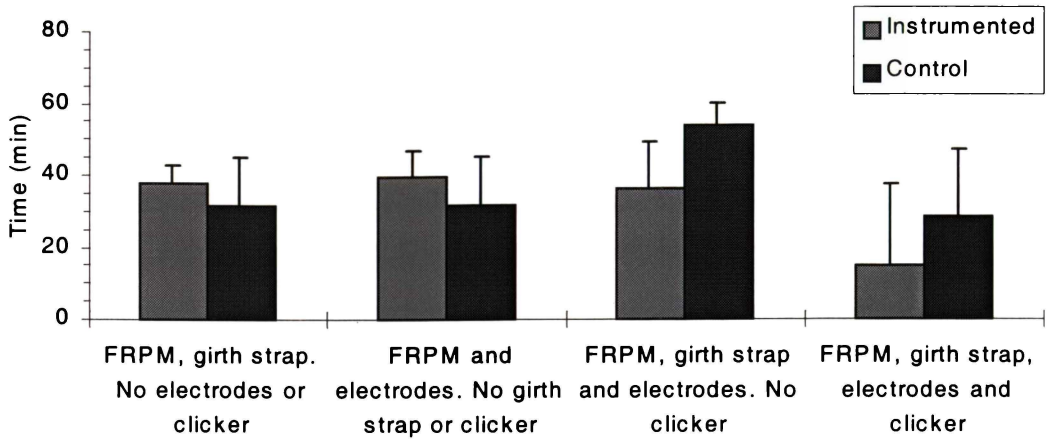


Figure 7.14 – Averaged Animal Lying Behavior versus Instrumentation Present on Animal (Mean ± SD)

Experiment	Probability of No Change in Behavior
FRPM and girth strap No electrodes or clicker	0.876
FRPM and electrodes No girth strap or clicker	0.728
FRPM, girth strap and electrodes No clicker	0.607
FRPM, girth strap, electrodes and clicker	0.650

Table 7.12 – Probability of No Behavior Change between Target Animal and Control Group versus Instrumentation Present on Target Animal

No significant differences were observed with any combination of instrumentation. However a trend towards the more instrumentation placed on the target animal the more likely the behavior of the target to differ from the rest of the group was observed.

Although the behavior analysis in Section 7.2.3 suggested that the FRPM influenced the behavior of the target animal in the group, this could not be confirmed by the present investigation. This repeat work illustrates the complexity and difficulty of using behavior as an assessor of stress. Behavior is a complex outcome contributed to by a variety of physiological changes, animal history, and animal individuality. As a consequence of this its expression in individuals and groups can be highly variable. This can make the use of behavior difficult as either an intra- or inter-animal measure of stress.

7.4 Discussion

Overall the results of the three measurement techniques, AER, heartrate, and behavior assessment show good agreement. This gives confidence to the results of each individual technique and suggests that all three techniques are measuring effects from the same causative source.

The advantages and disadvantages of each technique will be discussed in turn.

Based on the data acquired in this study, the use of the AER as an indicator of stress appears to be valid. Waveform sequences acquired from individual animals subjected to stressors showed considerable modification during exposure to the stressor and, in several cases, suggested evidence of recovery-type trends upon stressor application and removal. This was not the case for all animals in the experimental group, neither was it the case for all heartrate and behavioral data. This variation is indicative of the variation in each animal's response to a stressor.

The AER is a complex measurement, making its analysis more difficult than heartrate or behavior analysis. Baseline AER recordings differ considerably between individuals, making inter-animal comparison difficult, however the AER appears to be repeatable within an individual. The inter-animal variability has the result that, for each group of animals assessed using this technique, every animal must be individually characterized as the availability of normative data regarding changes in the AER, with respect to stress, is unlikely. This is not necessarily true for heartrate or behavior measurements.

The techniques for recording AER are also complex. The recording process is very sensitive to electrode or lead movement or damage due to the small signal levels involved. The use of needle electrodes in this study is also invasive compared to ECG self-adhesive electrodes, however alternative, less-invasive, electrodes may be suitable. To record the AER high-gain amplifiers, high-speed sampling, and synchronous stimuli generation are required of the recording instrument, adding instrumentational complexity to the study. In comparison, recording the ECG requires modest amplifier gains and sample rates and allows the use of simple yet robust self-adhesive electrodes. This leads to eased instrumentation demands and, if only heartrate is required, the instrumentation becomes even simpler. The recording of behavior requires little more instrumentation than a stop-watch, pen, and paper however, this recording technique requires significantly more labor input than either AER or heartrate measurement leading to a stronger potential for observer subjective bias.

Despite the added complexity required for AER measurement and analysis, and the obvious difficulty in interpretation due to inter-animal variation, I believe that the results contain useful information that cannot be obtained by either heartrate or behavior analysis. For instance, the sustained changes in waveform shape and the distinct recovery-type trends upon stressor application and removal are not seen in either the heartrate or the behavioral data. The AER also offers a measure of central nervous system processing during stress, a dimension of sophistication not available through other measures. However, there are still areas in which the AER acquisition

techniques described in this study could be improved. Such improvements have already been discussed (Section 7.2.1.2) and include better time resolution between waveforms and enhanced data analysis. Less invasive electrodes and lighter instrumentation may also reduce any instrumentational effects influencing the acquired data.

In comparison to the AER, the measurement of heartrate and its use as an indicator of stress requires much simpler instrumentation and analysis, and this adds to its attractiveness as an often-used indicator of stress. Despite the simplicity in acquisition, the interpretation of the results is difficult as it is often contaminated by physical activity adding to cardiac demands. Although the heartrate response to stress can vary significantly between animals, the simple measure of rate can easily be added across individuals to obtain an average response for the group. In this way normative, or typical, responses to stress can be obtained allowing transfer of knowledge of typical responses between different groups of animals. This may not be true for the AER.

Like the AER, the measurement of heartrate or ECG is well suited to ambulatory monitoring of free-ranging animals and the necessary recording equipment can be miniature and lightweight. For ambulatory heartrate measurement the limiting factors for long duration use are typically electrode failure or other damage, (although this can be minimised by implantation of the recorder) and restricted data storage memory in the recorder. However, interpretation of the heartrate data often needs an additional measure, such as behavior, of what the animal was doing at each instant, to match heartrate changes or abnormalities against. Often such an uninterrupted recording of behavior over the long distances and time periods involved in ambulatory studies is not practical. This is a major drawback of ambulatory monitoring in general, however it is particularly present in heartrate studies due to physical activity artifact.

Although the measurement of heartrate is simple, much useful information is discarded to obtain it. The use of this extra information, for example inter-beat interval variation and detailed QRS-complex analysis, would add complexity to both the acquisition and analysis of the data but also add extra information and possibly insight into the stress response. Despite the widespread use of these more complex measures in human studies, their use in animal welfare studies appears rare, possibly because of the added complexity. These types of analysis would bring the analysis of cardiac activity up to a similar level of complexity as a systemic measure as AER is as a central nervous system measure.

Analysis of behavior is often considered the most fundamental of measures for animal welfare studies. It has obtained this position due to its ease of implementation and analysis, and low cost. However, it too suffers from difficulties in interpretation, inter-animal variation, and repeatability (Section 7.3, for example). It can also be a rather imprecise method, relying on categorical data and human interpretation and recording of the raw signal in real-time and is not well suited for free-range animal studies due to the problem of logistics in long-term observation. Despite its limitations, the recording of behavior provides useful information, both in itself, and as a sequence of event markers that can provide insight to the results obtained from more complex measures.

In summary, each of the three techniques used to provide measures of stress in animals that were used in this study have their own advantages and disadvantages, but together they can provide extra information and depth to animal welfare studies. The key to the successful use of these techniques and others not used in this study, resides in the integration of quite different data from several complementary sources into an overall picture of how an individual or group of animals perceives and responds to stress in its environment.

The integration of the AER, heartrate, and behavior data obtained in this study into an overall model was beyond the scope of this thesis, yet this type of integration must be an area of vital interest for future work in animal welfare studies as more flexible and more capable instrumentation becomes more available in the coming years, allowing the researcher to obtain multiple different measurements on many animals simultaneously as they freely behave in the field.

Chapter 8 Summary and Suggestions for Future Research

8.1 The Free-Range Physiological Monitor

To enable the remote monitoring of freely-behaving animals in the field I developed the Free-Range Physiological Monitor, or FRPM. The FRPM is a state-of-the-art ambulatory physiological monitor, designed to allow researchers to record multiple aspects of the stress response simultaneously, and on a number of free-ranging animals. The FRPM incorporates a powerful microprocessor, the ARM710a with performance equal to that of a 33MHz Intel® 486 microprocessor, and 16MB of memory, to enable complex analysis of recorded data to be performed on the animal itself. Low power consumption and power management received special attention, resulting in a design that can operate for extended periods in the field by solar recharging of its internal batteries. The FRPM incorporates a $115.2kbs^{-1}$ spread spectrum radio modem with a range of up to 2km to allow the transmission of measurements in real-time to a base computer and to communicate with the researcher. This radio modem allows a network of up to 255 FRPM units to be connected with the base station.

An operating system/application architecture was used for the FRPM System Software allowing for flexible configuration of the FRPM for any particular experiment. The System Software includes a file system to allow storage of both applications and data and provides 10MB of DRAM memory storage, with an additional 1792kB of Flash memory storage. The System Software enhances the reliability of the FRPM by rigidly separating data collection and processing from critical system activity such as power management, and by its ability to terminate

any application that threatens system stability through illegal memory accesses or illegal instructions.

In its current configuration the FRPM provides for the assessment of central nervous system activity through the recording of both the electroencephalogram and the Auditory Evoked Response (AER), allows the monitoring of cardiac (heart) activity through the electrocardiogram (ECG), and provides support for measurement of additional stress parameters such as temperature, respiration rate, and physical activity.

The completed FRPM unit measures 260mm by 120mm by 30mm and weighs 980g. The enclosure was designed to handle both the physical abuse and environmental extremes encountered in field work, and has performed well in practice. There is good potential for both size and weight reduction of the FRPM, and several technologies have been identified that could result in an enhanced FRPM of less than half the size and weight of the current design.

The long-range and extended duration capability, and the types of information that can be measured by the FRPM, combine to make the FRPM a unique tool for welfare assessment on free-ranging animals. The potential of the technology for application in animal welfare research is very broad.

8.2 The Auditory Evoked Response as an Indicator of Stress

As an initial application, following development of the FRPM, the use of the AER as an indicator of stress in freely behaving sheep was investigated. As the AER originates from the brain itself, it was hypothesised that any change in the operation of the brain, such as that associated with stress, might cause the AER to be modified. To test this hypothesis 12 Romney-cross sheep were individually instrumented with the FRPM, and AER responses recorded as each animal was exposed to potential stressors. The AER results were complemented by heartrate, also recorded by the FRPM, and behaviour observations which helped to provide additional insight into the animals' response to the stressors. In total, each animal was exposed to four separate stressors. These comprised of Control (no stressor), Distraction (a moving, noisy water sculpture), Isolation (separation from the flock), and Fear (exposure to a dog) stressors.

During both the Control and Distraction stressors the AER was observed to be repeatable within individual animals. However, when a severe stressor such as isolation was applied to a single animal, or the flock was exposed to a predator (dog), the AER was observed to change significantly in some individual animals. Changes with stress appeared to be unique to each individual, making inter-animal comparison difficult. However, broad patterns including both sustained changes during the presence of the stressor and adaptation trends at the onset and end of stressor periods were observed.

The results suggest that different subjects can show quite individual responses to the same stressor, making the creation of a general model for changes in the AER as a quantifier of stress difficult. However, the AER may be a useful measure within an individual and when combined with more conventional measures such as heartrate and behaviour, which in themselves can be difficult to interpret, the AER should provide additional insight into the stress responses of freely-behaving animals.

Due to the significant difficulties encountered with intra- and inter-animal differences in the AER, significant further work is required to justify the use of the AER in standard practice as a reliable measure of stress over such conventional measures as heartrate or biochemical markers. In particular, future research should address the need to investigate the relationship in many more animals than the 14 studied here. Such a study should initially investigate the variation between animals under control conditions, taking into account additional effects such as genetics, followed by an investigation into the relationship between stress and the AER in a large population of animals. This much larger study may help to shed light on the variation in response observed under control conditions, between the animals in this study. It would also provide a much larger database to investigate changes in the Evoked Response between animals exposed to the same stressors. It would also be useful to correlate the AER results directly against other more conventional stress measures such as the cortisol concentration in saliva or urine.

A necessary requirement of any much larger study of a significant number of animals would be the formation of a standardised stressor among many animals. Perhaps such a standard could usefully employ anxiogenic or anxiolytic drugs to produce a metered level of stress change in the subject. Such a standard would prove useful not just for the study of the relationship between AER and stress, but also for other conventional measures currently used in animal stress studies. It may even prove possible to compare the efficacy of different stressors based on the dose of some specific anxiogenic agent that is required to produce the same response in the same animal under control conditions. Such a measure would help greatly in the comparison and ranking of stressors that animals are regularly exposed to under current farming practices.

Appendix 1 Publications and Presentations

A1.1 New Zealand Branch of the Australasian College of Physical Scientists and Engineers in Medicine Annual Conference

Harris, P.J., Cook, C.J., Schaare, P.N., Henderson, J.D., "The Free-Range Physiological Monitor and Auditory Evoked Response Measurement", Presented at the New Zealand Branch of the Australasian College of Physical Scientists and Engineers in Medicine Annual Conference, Hamilton, New Zealand, November 1997.

Abstract - Monitoring stress-induced changes in central nervous system function by relatively non-invasive means in freely moving subjects poses considerable challenge. One possible approach is to measure changes in the neural evoked responses of the subject.

This paper will discuss the remote recording and analysis of auditory evoked response from ambulatory subjects using the Free-Range Physiological Monitor (FRPM) and how these recordings may be used as an indicator of stress.

The FRPM is a novel instrumentation pack designed to record and analyze physiological processes in ambulatory subjects for extended periods of time with no experimenter intervention. Several monitors can be used simultaneously, with the summarized results being transmitted back to the experimenter.

Currently the FRPM is being used to investigate stress levels in sheep during transportation and handling for animal welfare studies.

A1.2 Biology of Animal Stress Conference (1)

Harris, P.J., Cook, C.J., Drake, K., Payne, S., Schaare, P.N., Henderson, J.D., "Evoked Response as an Indicator of Anxiety in Freely-Behaving Sheep", In: Proceedings of the Biology of Animal Stress Conference, University of California, Davis, California, August 1998.

Abstract - We have been investigating the use of the Auditory Evoked Response as an indicator of anxiety in freely behaving sheep. The Auditory Evoked Response is an electrical manifestation of the central nervous system's reception of and response to an external sound stimulus, in this case an auditory click. These responses can be recorded from small electrodes positioned on the head of an animal. They are extremely low in amplitude, and because of their admixture with normal background brain electrical activity and various electromyographic and movement artifacts, need to be separated out by signal processing techniques. The resulting signal consists of a series of electrical peaks that occur at consistent time intervals after stimulus, each peak representing the sequential processing of the stimulus by different groups of neurons. Because the signal originates from the brain itself it was hypothesized that any change in the operation of the brain, such as that brought on by anxiety, could cause the response to be modified.

To test this hypothesis we simultaneously recorded both Auditory Evoked Response and electrocardiogram signals from freely-behaving sheep using the Free-Range Physiological Monitor (FRPM) developed by our group. Under control conditions the Auditory Evoked Response was observed to be repeatable. However when a stressor such as isolation was applied to a single animal, or the flock was exposed to a predator (dog), the response could change significantly. The results show that different subjects can exhibit quite different responses to the same stressor, making the creation of a general model for changes in the Evoked Response as a quantifier of anxiety unlikely. However it may be a useful individual measure. The results also suggest that although heartrate, a conventional stress measure, may detect the onset of a stressor, it does not follow a sustained level of anxiety in the subject, whereas the changes that occur in the Evoked Response waveform remain as long as the stressor is present.

A1.3 Biology of Animal Stress Conference (2)

Harris, P.J., Schaare, P.N., Cook, C.J., Henderson, J.D., "An Ambulatory Physiological Monitor For Animal Welfare Research", In: Proceedings of the Biology of Animal Stress Conference, University of California, Davis, California, August 1998.

Abstract - The Free-Range Physiological Monitor, or FRPM, is a recently developed tool for the recording and analysis of physiological signals from freely-behaving animals. This ambulatory physiological monitor has been specially developed for use in animal welfare studies and allows multiple subjects to be monitored over extended periods of time and considerable distance.

The FRPM has been designed to operate attached to an animal for extended periods of time with no researcher maintenance. Power is supplied from a solar array with battery backup sufficient to power the unit for up to seven days in the event of heavy cloud. A radio network links as many as 250 FRPMs to a base station attached to the researcher's computer allowing the researcher to connect to any FRPM and execute application software or exchange data with the FRPM. The radio has a range of around 2km and is particularly resistant to interference. The use of a high-speed microprocessor with 16MB of memory allows complex signal processing algorithms to be executed on-board the FRPM with only the results being saved and transmitted to the researcher. Both the signal conditioning circuitry and the various software applications are modular, allowing the FRPM to be quickly optimized for different applications.

The FRPM is of robust and waterproof construction and has been designed to withstand both the weather extremes and physical abuse often encountered during long-term field trials. High density surface mount components were used throughout the design to minimize the size and weight of the unit.

Currently the FRPM is being used in animal welfare trials to determine if the Auditory Evoked Response can be used as an indicator of anxiety in sheep. It has also been used to record electrocardiogram signals from sheep undergoing washing by high-pressure water jet to monitor animal welfare aspects of the procedure.

A1.4 International Conference of the IEEE Engineering in Medicine and Biology Society

Harris, P.J., Schaare, P.N., Cook, C.J., Henderson, J.D., "An Ambulatory Physiological Monitor For Extended Use In Animal Welfare Studies", In: Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Hong Kong, October 1998.

Abstract - Monitoring physiological activity in freely moving subjects poses a considerable challenge. The FRPM is a novel instrumentation pack designed to record and analyze physiological processes in ambulatory subjects for extended periods of time with no researcher intervention. Several monitors can be used simultaneously, with significant data analysis and compression being performed on the unit before the summarized results are transmitted by radio link back to a base computer. Currently the FRPM is being used to investigate stress levels in sheep during transportation and handling for animal welfare studies.

Appendix 2 FRPM Hardware Design

The requirements of the FRPM demanded the use of both a powerful microprocessor and a large amount of memory in the unit. Although a Digital Signal Processor (DSP) may have provided excellent support for computational algorithms, such devices were quickly excluded due to a lack of power management support, on-board peripherals and general input/output capability. To obtain low power consumption, high performance, good power management, and a system with a minimal external component count, the CL-PS7110[i] microprocessor was chosen.

This processor was specially designed for use in low-power, high-performance applications. It is based around an ARM710a[ii] core, a 32-bit Reduced Instruction Set Computer (RISC) CPU with integrated cache and memory management. This core gives an average of 15 Vax™-MIPS of performance at 18MHz, typically matching the performance of a 33MHz Intel® i486, while consuming only 60mW of battery power. During standby the processor current draw is only 10μW.

The FRPM is built around the CL-PS7110 with all other circuitry being either controlled by or providing support for this part. The surrounding circuitry can be divided into four main areas: System Support, Memory Interface, Peripherals, and Power Management. Figure A2.1 through Figure A2.6 show the circuit diagrams of the Main Board of the FRPM. Figure A2.1 shows an overview of the functional blocks, while Figure A2.2 to Figure A2.6 show in detail the System Support, Memory, and Power Management areas. Figure A2.4 shows the Peripheral Interface block in detail.

[i] CL-PS7110 Microprocessor, Cirrus Logic Inc., 3100 West Warren Avenue, Fremont, CA 94538, USA
[ii] ARM710a Core, Advanced RISC Machines Ltd., 90 Fulbourn Road, Cherry Hinton, Cambridge CB1 4JN, UK

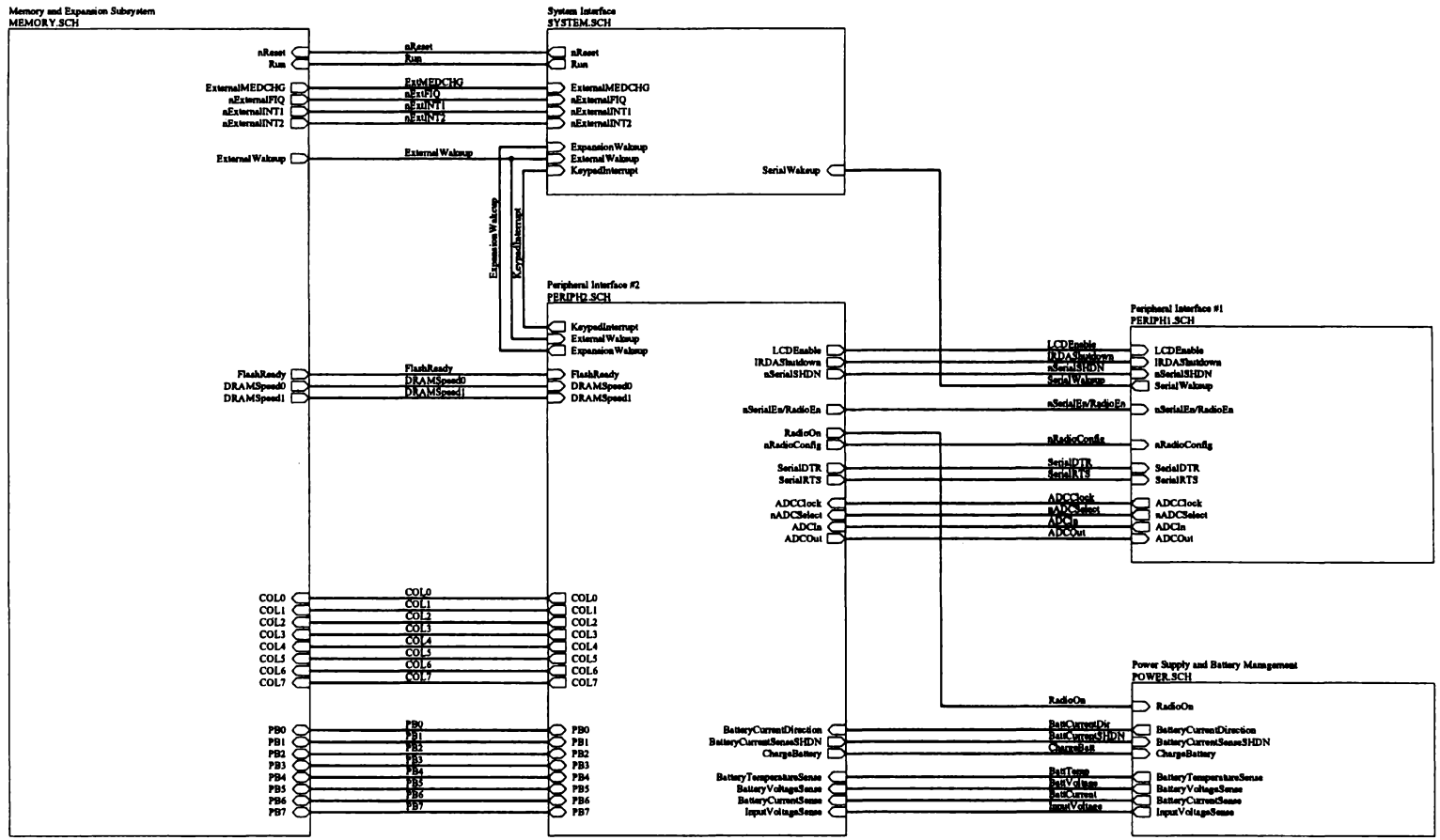


Figure A2.1 – FRPM Functional Blocks

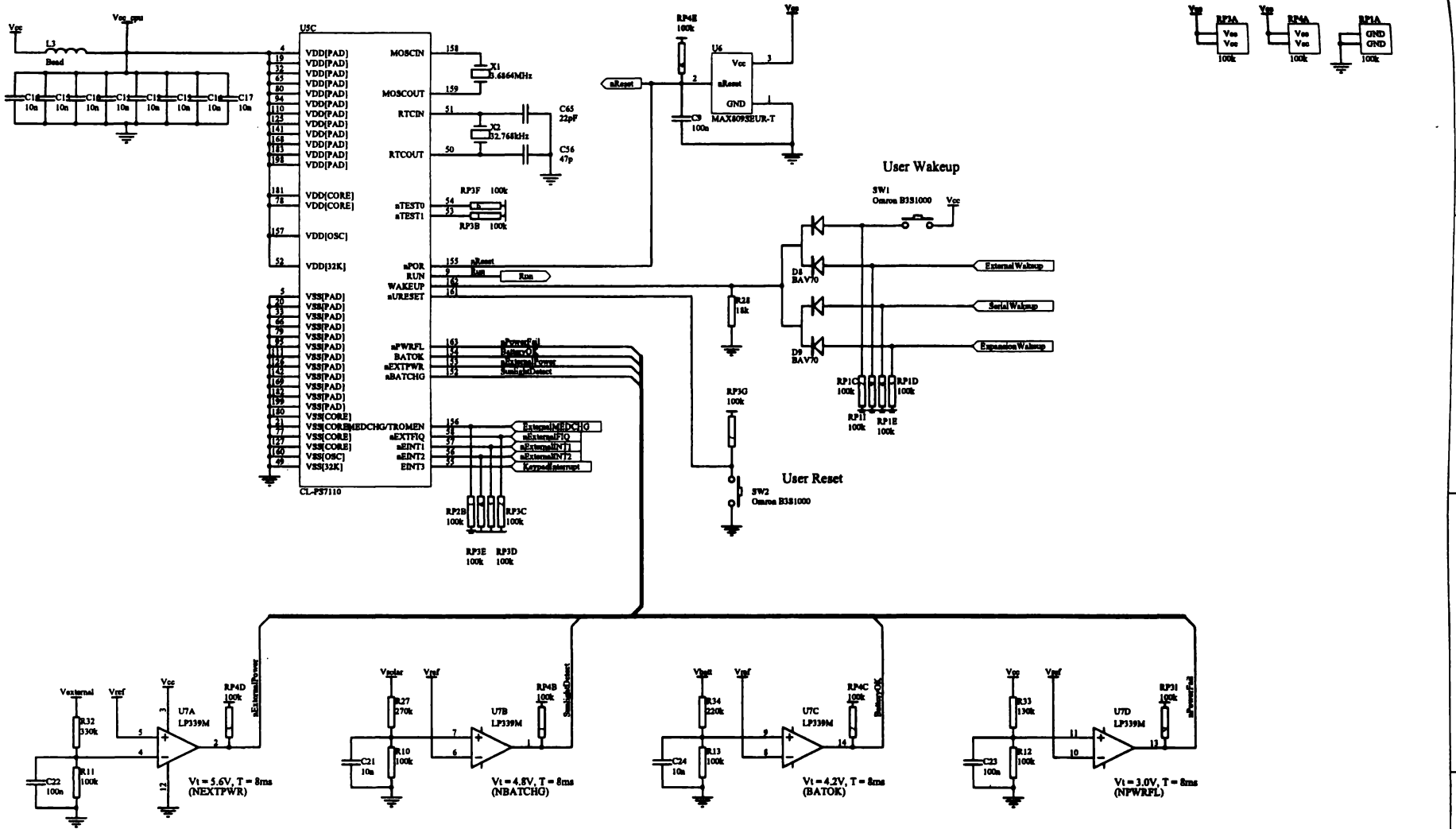


Figure A2.2 – FRPM System Support Circuitry

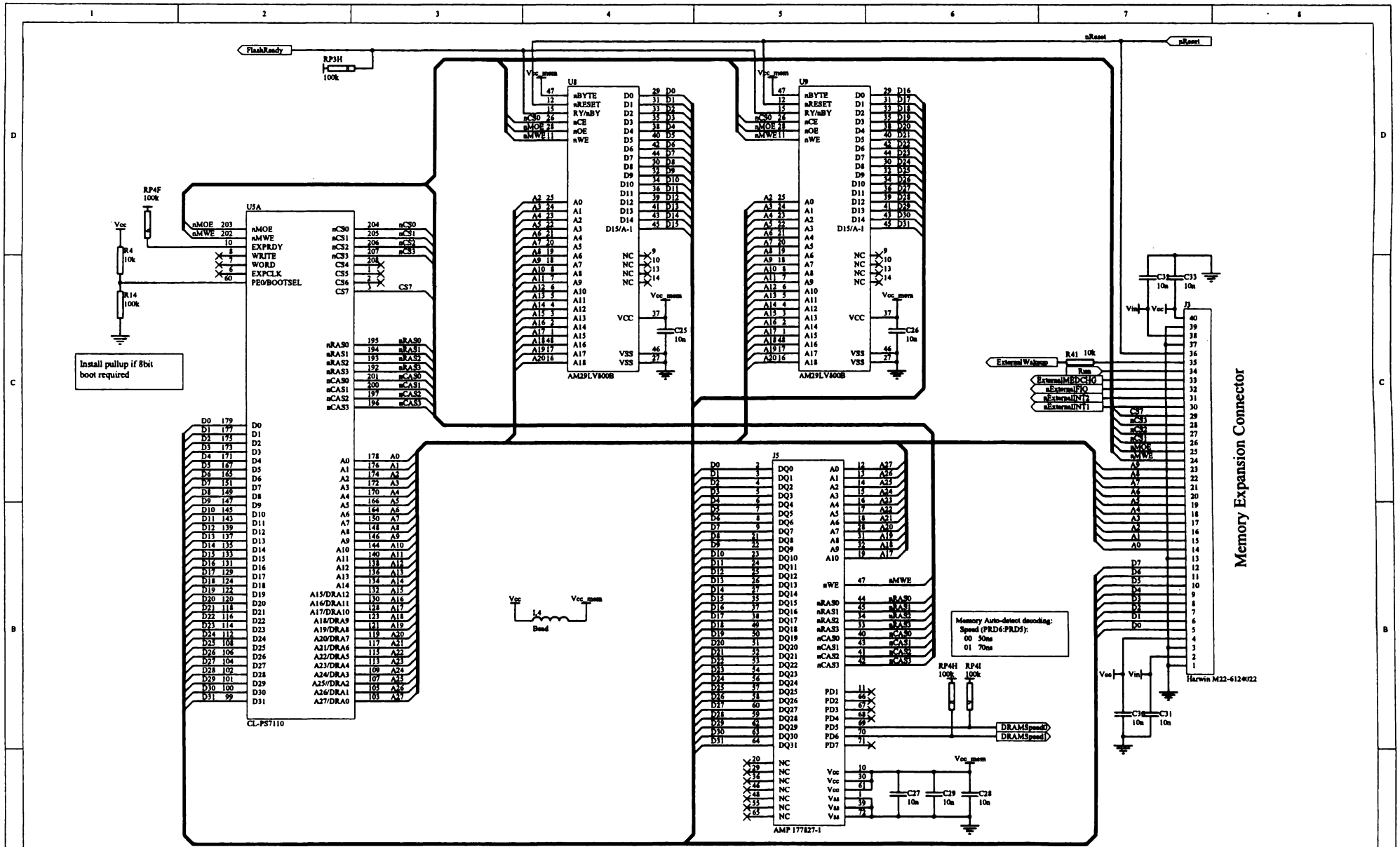


Figure A2.3 – FRPM Memory Circuitry

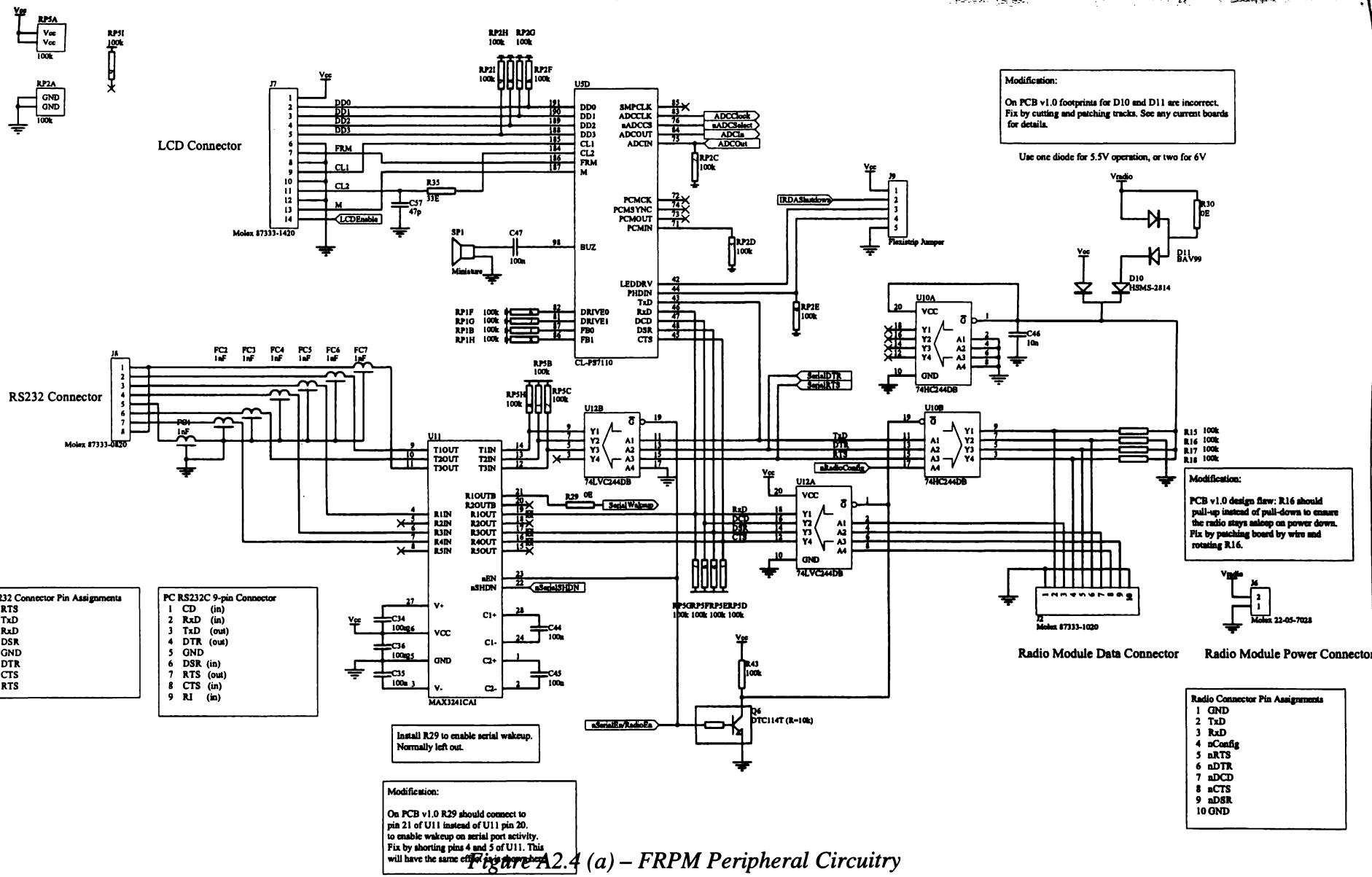


Figure A2.4 (a) – FRPM Peripheral Circuitry

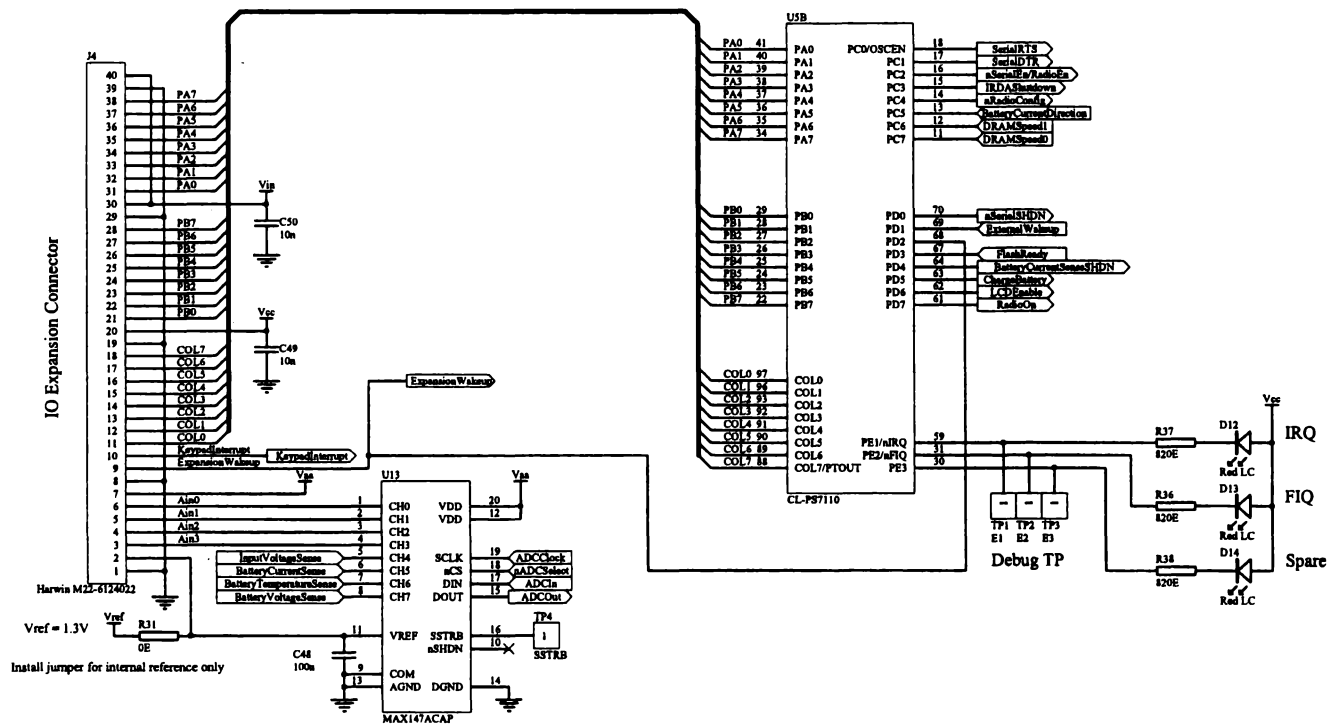


Figure A2.5 (b) – FRPM Peripheral Circuitry

Title	FRPM II Peripheral Interface, Part 2		Engineering Development Group Hort Research NZ Ltd Ruakura Research Centre Ruakura Road
Drawn by:	Phil Harris		
Size: A4	File: L:\WELFARE\FrpmDesign\Elec\FrpmDesign\Schematics\PE2.DSN		
Number:	Revision:	Date: 14-Dec-1999 Time: 20:47:18	Sheet 6 of 6

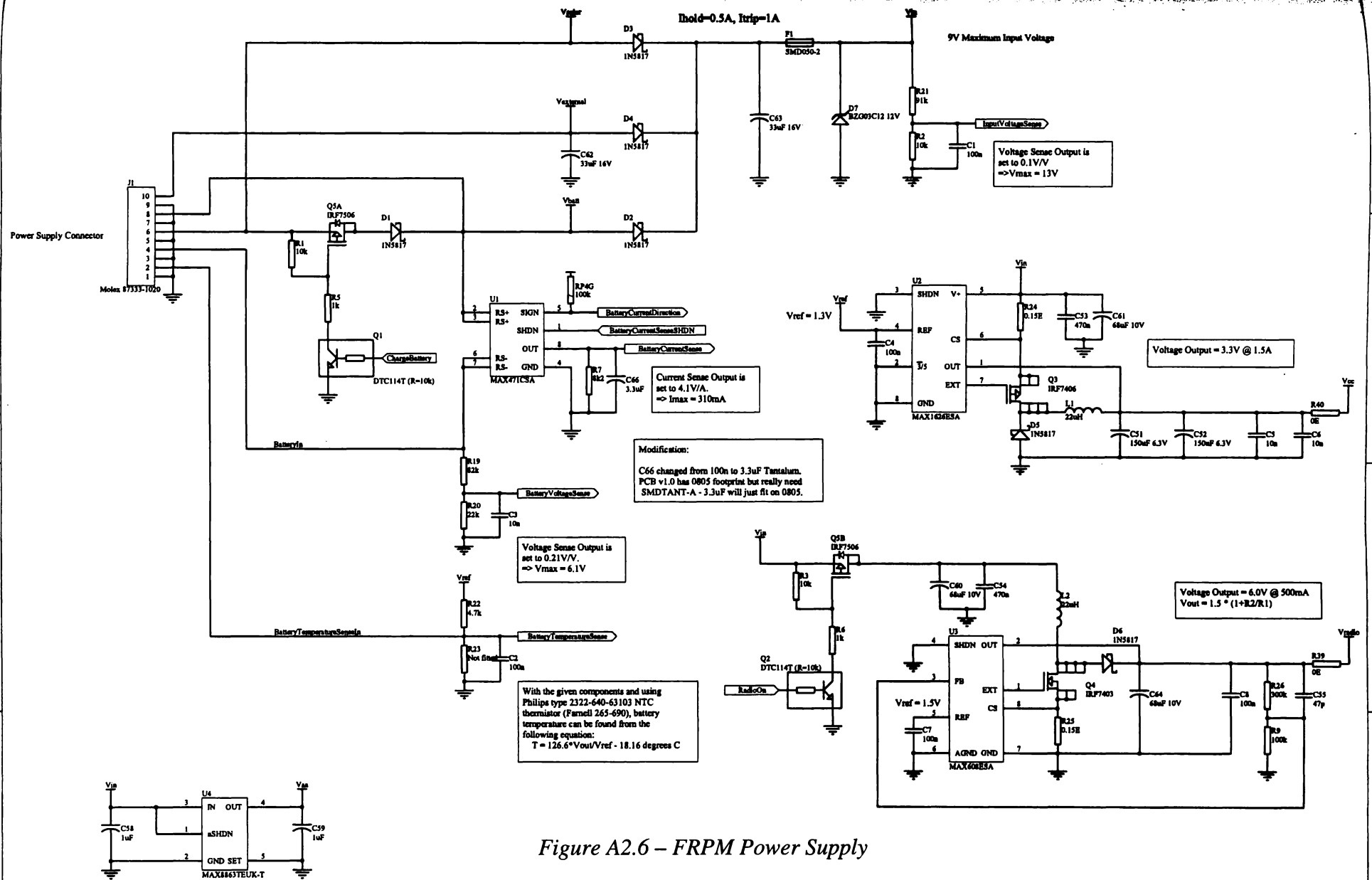


Figure A2.6 – FRPM Power Supply

A2.1 System Support

The System Support circuitry ensures that the processor is correctly initialized, is kept within its operating limits, and is able to provide some form of basic status information for system testing and debugging.

The System Support circuitry for the FRPM is straightforward due to the high level of integration within the CL-PS7110 and comprises four sections: Reset Management, Oscillators, Operating Mode Control and Debugging Support.

A2.1.1 Reset Sources

The CL-PS7110 has three possible reset sources which are the Power-On Reset (nReset), Power Fail Reset (nPowerFail), and User Reset (nUserReset).

The Power-On Reset signal is provided by the MAX809[i] (U6) reset device which generates a reset signal for approximately 240ms after the supply voltage Vcc reaches 2.93V. The filter comprising RP4E and C9 ensures that the nReset line is kept low even when Vcc is below the MAX809's minimum operating voltage during the startup time of the main power supply (see Section A2.3.1). This signal places the processor in Standby state and is the only complete reset available on the CL-PS7110.

The Power Fail Reset signal is provided by the LP339[ii] comparator (U7D). This comparator monitors the voltage on the Vcc supply rail to ensure that it stays above 3.0V, the recommended minimum supply voltage for the CL-PS7110. While Vcc is below 3.0V the CPU is held in its Standby state with the real-time clock still running and DRAM memory contents maintained via the DRAM Self-Refresh function.

The User Reset signal allows the user to force the processor into a known state by pressing the switch SW2. This reset has the same effect as Power Fail and is de-bounced inside the processor by the 64Hz real-time clock ticker.

A2.1.2 Oscillators

The CL-PS7110 requires two oscillators for operation. The main oscillator is based around a 3.6864MHz crystal (X1) which is multiplied by five inside the processor to give the 18.432MHz CPU clock. The secondary oscillator is based around a 32.768kHz crystal (X2) which is divided down to give a 1Hz source for the real-time clock. This oscillator also provides a 64Hz clock

[i] MAX809 Microprocessor Reset Circuit, Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086, USA.
[ii] LP339 Quad Comparator, National Semiconductor Corp., 2900 Semiconductor Drive, Santa Clara, CA 95052-8090, USA.

used for the startup and power management state machines internal to the CL-PS7110. C56 and C65 help with stabilization and startup of the secondary oscillator.

When in the Standby state the main oscillator is stopped while the secondary oscillator continues to run to keep the correct time in the real-time clock. Both oscillators run in the Run and Idle states.

A2.1.3 Operating Mode Control

The CL-PS7110 can be switched from its Suspend state to its Run state by a real-time clock alarm interrupt, an external interrupt, or a rising edge on its WAKEUP pin. Waking the device by either of the first two methods is performed internally within the CL-PS7110.

The FRPM Main Board allows four possible sources to wake the CL-PS7110 via its WAKEUP pin. These sources are the User Wakeup switch (SW1), a source attached to the Input/Output Expansion socket, a source attached to the Memory Expansion socket, or the MAX3241[i] RS232 serial port transceiver (U11). These sources are combined with the diode arrays D8 and D9 which perform a logical OR operation on the sources. A logical high level from any of these sources will cause the CL-PS7110 to switch to its Run state. This input is de-bounced inside the CL-PS7110 by the 64Hz clock. Any activity on the RS232 RxD input of the RS232 interface source will generate a wakeup signal, even if the receiver part of the RS232 transceiver device is shutdown via its nEN pin. The RS232 activity wakeup source can be disabled by removing the jumper R29.

A2.1.4 Debugging Support

Basic debugging support is provided on the FRPM Main Board by the three LED devices D12, D13, and D14 attached to the input/output port E and their associated test-points TP1, TP2, and TP3. These LEDs can be used to provide a visual indication of the current System Software state while the test-points provide the ability to measure software execution times or to trigger external test equipment.

If the DBGEN flag in the CL-PS7110 internal register SYSCON is set then TP1 and TP2 respectively represent the state of the Interrupt Request (IRQ) and Fast Interrupt Request (FIQ) inputs to the ARM710a core.

[i] MAX3241, True RS232 Transceiver , Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086, USA.

A2.2 Memory Interface

The memory available on the FRPM Main Board consists of 2MB of Flash Memory and 16MB of DRAM memory. This memory occupies only a small fraction of the 4GB address space available on the CL-PS7110. Table A2.1 shows the physical memory map for the FRPM Main Board.

Start Address	End Address	Size	Function
0x00000000	0x001FFFFF	2MB	Flash Memory Area
0x10000000	0x100003FF	1kB	Memory Expansion Area #1 Decoded by nCS1
0x20000000	0x200003FF	1kB	Memory Expansion Area #2 Decoded by nCS2
0x30000000	0x300003FF	1kB	Memory Expansion Area #3 Decoded by nCS3
0x70000000	0x700003FF	1kB	Bootstrap Loader Area Decoded by CS7
0x80000000	0x8000087F	2.125kB	CL-PS7110 Internal Registers Area
0xC0000000	0xC0FFFFFF	16MB	DRAM Memory Area

Table A2.1 – FRPM Main Board Physical Memory Map

The DRAM memory was chosen for the FRPM Main Board because it provided large amounts of storage in a very small format at a modest power consumption. Static-type RAM of the memory size required was excluded due to its high cost and low storage density. An equivalent-capacity static RAM array also had a higher power consumption in the Standby state than a self refreshing DRAM module. For example, the CY62148 512kB static RAM[i] has a standby power consumption of 2.75mW per device whereas the standby power consumption of the Simple Technology STI324004D1-60SVG[ii] 16MB module used in the FRPM is 1.65mW per device. If the CY62148 devices were used to obtain 16MB of memory, 32 devices would be needed giving a total standby power consumption of 88mW compared to 13mW for the STI324004D1-60SVG module. By choosing a widely used laptop computer memory module, ready availability at a low cost was also assured.

The Flash memory provides permanent storage for System Software, vital experiment data, and applications. At FRPM startup the System Software is copied from the Flash memory to DRAM where it is executed, since executing the software in DRAM is almost twice as fast as from Flash memory.

[i] Cypress Semiconductor, 3901 North First Street, San Jose, CA 95134, USA.
[ii] STI324004D1-60SVG, SO-DIMM DRAM Module, Simple Technology Inc., 3001 Daimler Street, Santa Ana, CA 92705, USA.

A2.2.1 The DRAM Memory Interface

The DRAM memory interface is based around a 32-bit wide, 16MB module organized as 11 address columns by 11 address rows. This 'square' addressing structure ensures that the DRAM memory space appears as a single linear address space and is not broken into smaller segments by the CL-PS7110 addressing scheme.

Typically 32-bit DRAM modules are split internally into two 16-bit wide sections. To access these sections as a combined 32-bit wide memory, the module nRAS0 and nRAS1 inputs are both connected to the CL-PS7110 nRAS0 output to give a single 32-bit wide, 16MB memory bank from physical address `0xC0000000` (hexadecimal) through to `0xC0FFFFFF`.

The CL-PS7110 supports both individual word access and sequential (page mode) accesses to memory. Sequential accesses are always faulted after four words to allow any LCD display DMA transfer to occur. This is essential to keep a steady, flicker-free, image on the display. Figure A2.7 shows both individual and sequential memory reads while Figure A2.8 shows both individual and sequential memory write accesses.

For 32-bit individual word memory transfers to the DRAM module the row address is clocked into the module on the falling edge of nRAS followed by the column address on the falling edge of all of the nCASx lines. For a memory read (nMWE high), the data output from the module is read into the CL-PS7110 on the rising edge of the nRAS line. For a memory write (nMWE low) the data is clocked into the module on the rising edge on the nRAS line. Reading or writing an individual word in memory takes 3 cycles (160ns) on the CL-PS7110.

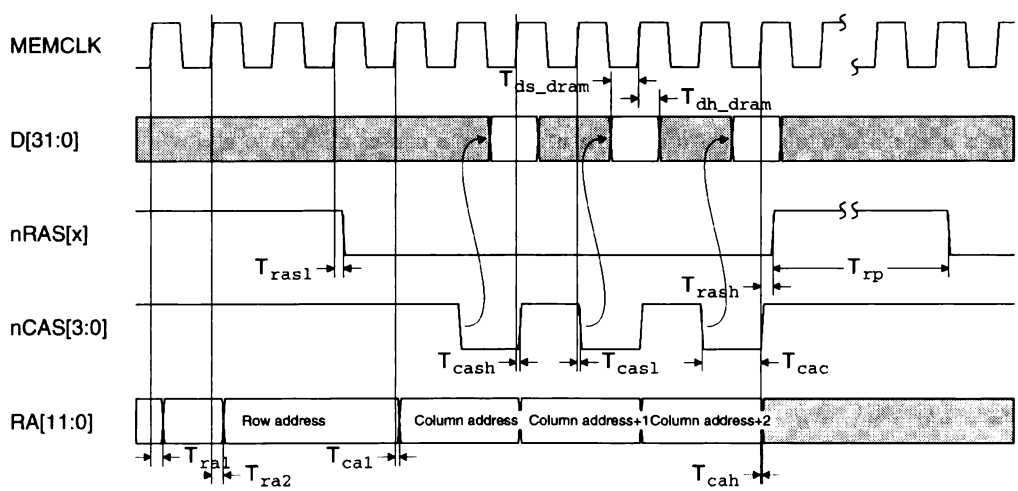


Figure A2.7 – DRAM Read Cycle Timing

In some instances a sequential access to memory can occur. Examples are a processor cache line fetch or a processor core write buffer flush. If the sequential access occurs within a DRAM row the CL-PS7110 can perform a sequential access to the DRAM memory where the first word is accessed in the conventional manner, but for subsequent words only the column address is

changed. For a sequential read, the first word takes two cycles with subsequent words taking one cycle each. Sequential accesses are always faulted (stopped) after four words have been transferred so that four words can be read at an average of 1.25 cycles per word. For a sequential write, all words take two cycles each hence there is no improvement over random words.

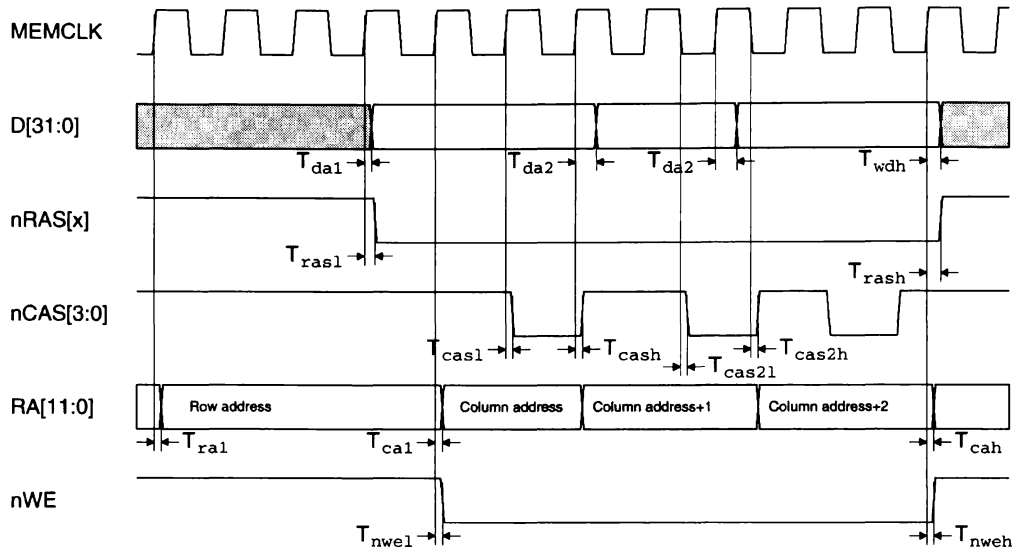


Figure A2.8 – DRAM Write Cycle Timing

The CL-PS7110 also supports byte transfers to memory. For a byte read the word containing the required byte is read into the processor and is then rotated so that the required byte occupies the eight least significant bits. All other bits are set to zero. For a byte write the byte is replicated in each of the four bytes within a data word. The resulting word is then transferred to memory with only the appropriate nCASx being taken low. For example, a byte at position 0 within the word causes only nCAS0 to go low while a byte at position 3 causes nCAS3 to go low. In this way the other bytes within the addressed word in memory are protected from being overwritten.

Either Fast Page Mode (FPM) or Extended Duration Output (EDO) types of DRAM may be used on the FRPM Main Board. However, if an EDO device is used then the EDO capability will not be utilized as the CL-PS7110 does not support it. The module access time should be 60ns or better.

DRAM memory requires refresh cycles to preserve memory contents[133]. This is because an individual memory cell within the memory is implemented as a transistor/capacitor pair with a bit being stored within the cell as a voltage on the capacitor. Over time the voltage on this capacitor decays and must be restored to prevent loss of memory contents. The CL-PS7110 generates a refresh cycle by performing a CAS-before-RAS sequence. This sequence is similar to a read sequence except in this case a falling edge on the nCASx lines is followed by a falling edge on the nRASx lines before the nRASx and nCASx lines are restored to their resting high state. Compare this with the normal RAS-before-CAS sequence of a data transfer. A single refresh cycle refreshes the contents of an entire column (8kB) in the module and takes 4 cycles (217ns) on the

CL-PS7110. Because they occur relatively infrequently they have little effect on overall performance. For example the 128ms refresh time of the DRAM modules used in the FRPM require a refresh cycle rate of 16kHz to maintain contents, hence the module is occupied by a refresh cycle only 0.35% of the time. The refresh cycle rate is programmed via the DRFPR register in the CL-PS7110.

To enable memory contents to be kept intact during the CL-PS7110 Standby state the DRAM module must also support Self-Refresh operation. In this mode the module generates internal refresh cycles at a reduced rate to keep its contents intact. The module is placed in Self-Refresh mode by performing a standard CAS-before-RAS refresh cycle but with the nRAS low condition extended indefinitely until full-speed operation is again required.

The current consumption of the DRAM module in Self-Refresh mode dominates the FRPM Standby power consumption. The first memory modules obtained for use in the FRPM were Micron MT8LDT432HG-60XL[i] 16MB, 60ns parts and gave an FRPM Standby current consumption of 9mA. By switching to Simple Technology STI324004D1-60SVG[ii] 16MB, 60ns parts the Standby current consumption dropped to 4mA.

A2.2.2 Flash Memory Interface

The Flash memory interface is based around two Am29LV800B[iii] 1MB, 16-bit wide devices. The connection of the Flash memory devices to the CL-PS7110 is relatively straightforward: one device (U8) provides the lower 16 bits, while the other (U9) provides the upper 16 bits of data. Since the nBYTE pin of each device is tied high, both devices are organized as 512k by 16 bits giving a total Flash memory size of 2MB. The nCE pin of each device is attached to the nCS0 line from the CL-PS7110 causing the Flash memory to occupy a memory area from physical address `0x00000000` (hexadecimal) through to `0x001FFFFF`. The internal programming state machine of the Am29LV800B devices is reset at power-on via the nRESET pin and the MAX809 reset device (U6).

Since the ARM710a processor starts at execution address `0x00000000` after a reset, the Flash memory contains the system startup software. Note also that the ARM exception vectors (interrupts, memory management aborts, and execution error) occupy addresses `0x00000004` to `0x0000001F`. Although the Flash memory has a slower access time than the DRAM memory, the frequent calling of the Interrupt Request (IRQ) exception in particular will cause the 16 bytes around the IRQ vector addresses (Data Abort, IRQ and Fast Interrupt Request (FIQ) vectors) to be present within the cache most of the time, preventing a Flash memory access and allowing full performance. Similarly, execution of application code which uses the Software Interrupt (SWI)

---

[i] MT8LDT432HG-60XL SO-DIMM DRAM Module, Micron Technology Inc., 8000 South Federal Way, Boise, ID 83707-0006 USA.
[ii] STI324004D1-60SVG, SO-DIMM DRAM Module, Simple Technology Inc., 3001 Daimler Street, Santa Ana, CA 92705, USA.
[iii] Am29LV800B, Advanced Micro Devices Inc., One AMD Place, Sunnyvale, CA 94088-3453, USA.

exception to communicate with the operating system will cause the SWI vector to be available from the cache.

The Am29LV800B devices have 120ns access times and hence two wait states are required to provide correct timing to interface the devices to the CL-PS7110. Figure A2.9 shows the timing diagrams for Flash memory, and also Expansion Interface, read and write accesses. Note that write access to a Flash device is distinct from programming a Flash device in that a write access is merely a transfer of data, whereas a programming cycle is a sequence of operations (multiple read and write accesses) performed on the Flash device to program data into to the Flash memory.

For read access the address is placed on A2:A20 and a falling edge on nCS0 and nMOE during the 1st MEMCLK cycle causes the Flash device to make the data word available for reading into the CL-PS7110 during the 4th cycle (2 wait states, transfer shown has 1 wait state). Hence a single read access to Flash memory takes 4 cycles or 217ns. The Am29LV800B devices do not support the faster sequential access mode.

For write access the address is placed on A2:A20 and a falling edge on nCS0 selects the device during the 1st MEMCLK cycle. The data word to be written is then placed on the data bus for 2 cycles (2 wait states, transfer shown has 1 wait state) and a falling edge on the nMWE line at the end of this period causes the data to be written to the device. Another cycle later and the transfer is complete. Hence a single write access to Flash memory takes 4 cycles or 217ns.

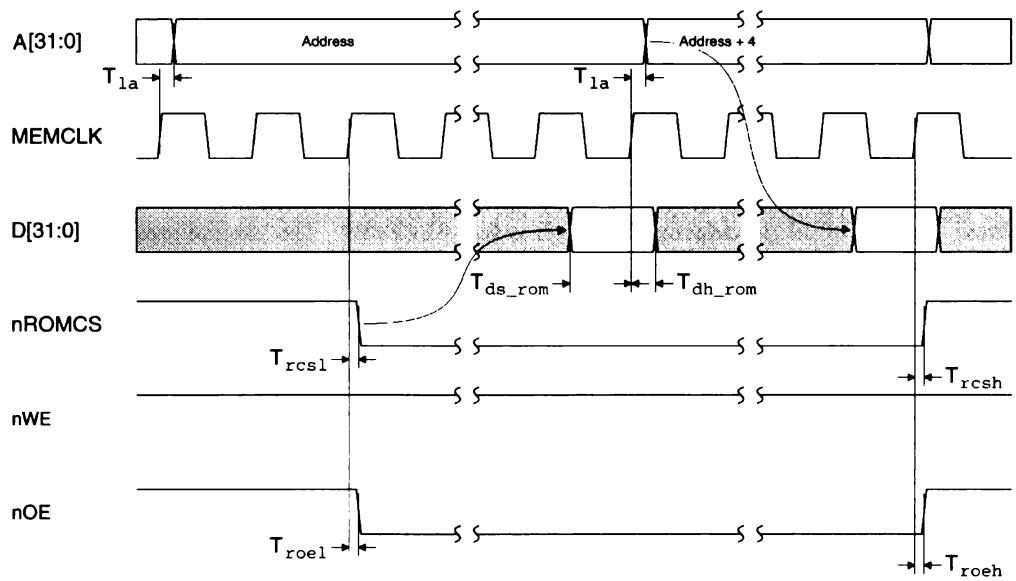


Figure A2.9 – Flash Memory and Expansion Interface Read/Write Cycle Timing

For minimum (1) wait state random access the read and write access times are 80ns. Each wait state adds 54ns to this time hence both t_{EXRD} and t_{EXWR} for Flash memory accesses are set to 134ns to satisfy the Am28L800B 120ns minimum requirement.

The Am29L800B devices do not support byte access hence all access must be by 32-bit words. The Flash memory interface in the FRPM supports the reading of individual bytes in Flash

Memory but does not support the programming of individual bytes. Byte-wise read access is possible because when the ARM CPU reads a byte location it actually fetches a complete word, rotates the required byte within the word to fill the least-significant bits and masks off, or ignores, the undesired three bytes. For write access however, the ARM CPU duplicates the byte to be written to fill a complete word which is then written to memory where the memory interface is expected to ignore the undesired bytes. The Flash Memory interface has no means to ignore these bytes hence a byte-wise write will cause a full word write to occur. Since the Flash memory is only used for bulk storage this poses few limitations.

For early generations of Flash memory devices, in-circuit erasing and programming was a relatively awkward task requiring high programming voltages (>12V) and precise timing. However, the Am29LV800B devices are single-supply devices and incorporate an internal state machine that manages both erase and programming operations. By issuing command sequences of bus write cycles to the Flash memory, erase and program functions can be executed easily. An incorrect command sequence will fail to unlock the Flash memory for the desired erase or program function and the command will be ignored.

It is possible to abort a command sequence by performing any read access from Flash memory. However, once a command sequence has been finished it cannot be revoked.

The progress of a programming or erase operation is monitored in the FRPM by the CL-PS7110 FlashReady input (pin 67) which watches the RY/nBY open-drain outputs of the two Flash memory devices (pin 15). When this output is low, one or both devices are busy with some operation and should not be accessed (accessing will return invalid data). When the output returns high the operation is complete and the Flash Memory can be accessed as usual.

To program a 32-bit word in the FRPM Flash memory the command sequence of bus cycles presented in Table A2.2 must be performed.

Bus Cycle	Operation	Address	Data
1	Write	0x00015554	0xAAAAAAAA
2	Write	0x0000AAA8	0x55555555
3	Write	0x00015554	0x55555555
4	Write	Address to program	Data to program

Table A2.2 – Flash Memory Write Sequence

The first three cycles unlock the Flash memory devices for programming while the last cycle actually programs the memory location. Once this sequence is sent to the Flash devices no other action needs to be taken by the FRPM to complete programming. Any command sequences written to the Flash memory during programming will be ignored. Note that this sequence must, however, be repeated for every word to be programmed and that data programmed as '0' cannot be programmed to a '1' without an intervening erase command sequence. It typically takes 11µs to program an individual word.

To erase the entire FRPM Flash Memory the command sequence of bus cycles presented in Table A2.3 must be performed.

Bus Cycle	Operation	Address	Data
1	Write	0x00015554	0xAAAAAAAA
2	Write	0x0000AAA8	0x55555555
3	Write	0x00015554	0x80808080
4	Write	0x00015554	0xAAAAAAAA
5	Write	0x0000AAA8	0x55555555
6	Write	0x00015554	0x10101010

Table A2.3 – Flash Memory Chip Erase Sequence

In this case it takes six cycles to unlock the Flash memory devices for the erase to occur. Note that erasing the entire Flash memory also erases the ARM CPU exception vectors and this could result in a System Software failure unless care is taken. It typically takes around 1 second to erase the entire Flash memory.

The FRPM Flash Memory is divided by the internal structure of the Am29LV800B devices into 19 sectors, each of which can be erased individually if desired. For example, erasing sectors 0-3 allows the System Software to be updated to a newer version. Table A2.4 shows the memory map of these 19 sectors and their function in the FRPM.

Sector	Start Address	Length	Function
0	0x00000000	32kb	System Software Storage
1	0x00008000	16kb	System Software Storage
2	0x0000C000	16kb	System Software Storage
3	0x00010000	64kb	System Software Storage
4	0x00020000	128kb	Interpreter Application Storage
5-18	0x00040000 + 0x00020000 * (Sector-5)	128kb	Flash File System (Total File Space = 1792kb)

Table A2.4 – FRPM Flash Memory Sector Memory Map

To erase any individual sector in the FRPM Flash Memory the command sequence of bus cycles presented in Table A2.5 must be performed.

Bus Cycle	Operation	Address	Data
1	Write	0x00015554	0xAAAAAAAA
2	Write	0x0000AAA8	0x55555555
3	Write	0x00015554	0x80808080
4	Write	0x00015554	0xAAAAAAAA
5	Write	0x0000AAA8	0x55555555
6	Write	Sector Address	0x30303030

Table A2.5 – Flash Memory Sector Erase Sequence

In this case it takes six cycles to unlock the Flash memory devices for the sector erase to occur. It is possible to erase multiple sectors simultaneously by performing the 6th bus cycle multiple times, each time with the address of the next sector to erase. Provided each additional sector is added within 80μs of the previous one, the additional sector will also be erased. Note that erasing sector 0 erases the ARM CPU exception vectors and this could result in a System Software failure unless care is taken. It typically takes around 1 second to erase a sector or group of sectors.

The Am29LV800B devices also support sector erase suspend and resume commands which allow the system to interrupt a sector erase to read or program sectors not currently being erased. However, the current FRPM System Software does not take advantage of this feature.

A2.2.3 Memory Expansion Interface

The Memory Expansion Interface is a byte-wide interface that allows for the direct addressing of up to 1024 bytes of either data memory or memory-mapped registers, in up to 4 devices. This interface is provided mainly to cater for the addition of memory-mapped peripherals to the FRPM. Such additions may include such devices as a high-speed analog-to-digital converter or a Field-Programmable Gate Array (FPGA) device to perform some dedicated task. The pin allocations for the 40-pin Memory Expansion Interface connector (J3) are shown in Table A2.6.

The timing of this interface is similar to that of the Flash Memory Interface although accesses are always byte-wide. The correct number of wait states for additional devices attached to the Memory Expansion Interface should be set using the CL-PS7110 MEMCFG1 and MEMCFG2 registers.

Pin #	Name	Direction	Description
1,3,13, 37,39	GND	Power	Ground
2,38	Vin	Power	Unregulated power supply input
4,40	Vcc	Power	Main logic supply
5-12	D0-D7	In/Out	8-bit expansion data bus
14-23	A0-A9	Output	Expansion byte address
24	nMWE	Output	Expansion write enable - active low
25	nMOE	Output	Expansion output enable - active low
26	nCS1	Output	Expansion device select #1 - active low
27	nCS2	Output	Expansion device select #2 - active low
28	nCS3	Output	Expansion device select #3 - active low
29	CS7	Output	Expansion device select #7 - active high
30	nINT1	Input	Interrupt request (IRQ) input #1 - active low
31	nINT2	Input	Interrupt request (IRQ) input #2 - active low
32	nFIQ	Input	Fast interrupt request (FIQ) input - active low
33	MedChg	Input	Media changed interrupt request input - active high
34	Run	Output	High when system is in Operating or Idle state - low while system is in Standby state
35	Wakeup	Input	Rising edge forces system into Operating state
36	nReset	Output	Power-on reset output

Table A2.6 – Memory Expansion Connector Pin Assignments

A2.2.4 Bootstrap Loader

Currently, the main application of the Memory Expansion Interface is to allow the system to be started from a completely dead state where no software is present in either Flash or DRAM memory and must be loaded from an external source. The device that accomplishes this task is called the Bootstrap Loader.

The CL-PS7110 can be forced to begin execution from an external 8-bit memory device attached to CS7 following a power-on reset and subsequent wakeup by holding the MEDCHG input high during the transition from low to high of its nPOR pin. This mode can be used to allow the Bootstrap Loader to be temporarily attached to the Main Board to initialize the system.

The Bootstrap Loader consists of a PCB containing a small 8-bit wide EPROM attached to the Memory Expansion Interface and selected via CS7. The EPROM contains a small program that initializes all of the CL-PS7110 internal registers, enables the serial port, transmits a startup message, and waits to receive a binary image to program into the Flash Memory. Because only address lines A0-A9 are present on the Memory Expansion Connector (J3) the entire Bootstrap Loader software must fit in 1024 bytes of memory or 256 ARM CPU instructions.

After the Bootstrap Loader board is removed and the unit experiences another power-on reset the CL-PS7110 will begin execution from Flash memory. In the case of the FRPM, the Bootstrap Loader is used to program either the System Software image or a more flexible program loader into the Flash memory.

A2.3 Power Management

The Power Management section of the FRPM Main Board is focused on monitoring the supply status and maintaining battery condition. It also converts unregulated power from one of several sources to the regulated supply voltage required by most of the Main Board circuitry.

The FRPM can accept power from four sources - the battery, the solar panel, an external power supply, or an external battery charger. The input power is then converted into four different internal supplies, these being a smoothed but unregulated supply (Vin), the main 3.3V logic supply (Vcc), a separate quiet 3.15V analog supply (Vaa), and the 6.0V radio power supply (Vradio).

The four input power sources are combined by diodes D2, D3, and D4. These diodes select the source with the highest voltage as the supply to use. Schottky diodes were used here for their low forward voltage drop (V_f). The resulting voltage is smoothed by C63 with self-resetting fuse F1 and Zener diode D7 providing over-current and over-voltage protection respectively. The input

voltage is measured by channel 4 of the MAX147[i] ADC (U13) via the divider R21 and R2. The formula for converting ADC counts to input voltage is:

$$V_{in} = 1.3\text{V}\frac{N}{4096}\frac{91\text{k}\Omega + 10\text{k}\Omega}{10\text{k}\Omega} \qquad \text{(A1.1)}$$

Where N is the number of ADC counts and V_{in} is the actual input voltage.

A2.3.1 Power Monitoring

The power supply status is monitored by watching for four different conditions: the use of an external power source, sunlight detection as indicated by the solar panel terminal voltage, battery terminal voltage failing, and the main logic supply falling below an acceptable limit.

Any attached external power supply is detected by comparator U7A. If the Vexternal input on connector J1 exceeds 3.9V it is considered that an external power supply is powering the FRPM and the nEXTPWR on the CL-PS7110 is forced low, causing the DCDET flag in the CL-PS7110 internal register SYSFLG to be set.

The sunlight detection is used by the battery charging software to manage charging cycles. If comparator U7B detects a voltage greater than 4.8V on the solar panel the NBFLG bit in the SYSFLG register is set which causes the battery charger task of the FRPM operating system to start a charging cycle. Normally the NBFLG bit is used to indicate to the CL-PS7110 that the device it controls has had its battery replaced.

While the battery terminal voltage is above 4.2V comparator U7C will generate a BATOK signal. If this signal transitions to a low state a Fast Interrupt Request (FIQ) will be issued which will cause the FRPM operating system to immediately force the system into Standby mode. If this event occurs the operating system will keep the system in Standby for at least 24 hours in an attempt to allow the battery to be recharged by the solar panel. Note that the CL-PS7110 will not allow itself to leave Standby mode until either the BATOK signal returns to a high level or the nEXTPWR signal is low.

The monitoring of the main logic supply is discussed in Section A2.3.3.

A2.3.2 Battery Management

Good battery management is essential for maximizing battery life while the FRPM is in field use. A complete battery management system should monitor the battery terminal voltage, current, and temperature. From these three measurements the internal state of the battery can be accurately determined[134].

[i] MAX147 8-channel Serial 12-bit ADC, Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086, USA.

The FRPM was designed to be compatible with two types of rechargeable battery technology: Nickel-Cadmium (NiCd) and Nickel-Metal Hydride (NiMH). Of these NiCd battery technology is the most common but has the disadvantage of requiring full charge/discharge cycles to maximize battery lifetime available capacity (the 'memory effect'). The newer NiMH battery technology has the advantage of almost twice the stored energy density of NiCd as well as allowing partial charge/discharge cycles without loss of available capacity. The battery selected for use in the FRPM was a 4.8V 1300mAh NiMH unit comprising of 4 'AA' type cells[i] and weighing 104g total. Typical charging profiles for NiCd and NiMH cells are shown in Figure A2.10.

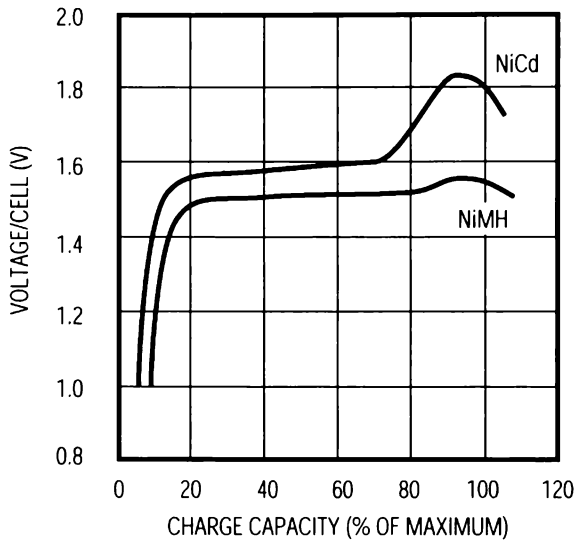


Figure A2.10 – Charging Profiles of NiCd and NiMH Cells

The battery can be recharged from two possible sources: the solar panel output can be controlled by the FRPM System Software via the ChargeBattery output and high-side switch Q5A to produce an optimal battery charging profile, or alternatively an external battery charger can be used. A charge cycle is started by the FRPM System Software whenever a sunrise condition is detected from the solar panel (see Section A2.3.1). The FRPM has no control over any external charger.

When charging NiCd cells the dip in terminal voltage after 100% capacity is used for charge termination. However, in NiMH cells this dip is very small and may not even be present and so NiMH charge cycles are usually terminated by detecting the rapid rise in cell temperature after 100% capacity is reached. The FRPM System Software terminates charging when either of these conditions is reached, allowing both NiCd and NiMH batteries to be used. The battery is also protected by terminating the charge if the battery temperature exceeds minimum or maximum limits or if the terminal voltage exceeds a maximum limit.

[i] GP130AAH NiMH Cells, GPI International Ltd., Kwai Chung, NT, Hong Kong.

Battery voltage is measured by channel 7 of the MAX147 (U13) ADC via the divider R19 and R20. The formula for converting ADC counts to battery terminal voltage is:

$$V_{batt} = 1.3\text{V}\frac{N}{4096}\frac{82\text{k}\Omega + 22\text{k}\Omega}{22\text{k}\Omega} \tag{A1.2}$$

Where N is the number of ADC counts and V_{batt} is the actual battery voltage.

Battery current is measured the MAX471[i] (U1) current sense amplifier. This part has a gain of 500μA/A which is converted to 4.1V/A by R7 and measured by channel 5 of the MAX147 ADC. The direction of current flow is indicated by level on the SIGN pin, a low level indicating the battery is discharging. The formula for converting ADC counts to battery current is:

$$I_{batt} = 1.3\text{V}\frac{N}{4096}\frac{1}{(8200\text{k}\Omega)(500\mu\text{A/A})} \tag{A1.3}$$

Where N is the number of ADC counts and I_{batt} is the actual battery current.

Battery temperature is measured via a precision thermistor[ii] attached to a cell in the battery. This thermistor forms a voltage divider with R22 with the output voltage being measured by channel 6 of the MAX147 ADC. The formula for converting ADC counts to battery temperature is:

$$\begin{aligned} V_t &= 1.3\text{V} * N / 4096 \\ R_t &= \frac{V_t * 4700\text{k}\Omega}{1.3\text{V} - V_t} \\ T &= \frac{\beta}{\ln(R_t.K)} - 273.0 \\ \text{Where: } K &= \frac{1}{R_0}\exp(\frac{\beta}{T_0}) \end{aligned} \tag{A1.4}$$

Where N is the number of ADC counts and T is the actual temperature in °C. The variables β, R_0, and T_0 are the three characteristic parameters of the thermistor.

A2.3.3 Main Power Supply

The Main Power Supply is responsible for providing a smooth and stable 3.3V voltage supply for the FRPM from the varying input voltage supplied by any of the four possible sources (Section A2.3). This supply, labeled Vcc in the circuit diagrams, is used by almost all of the circuitry in the FRPM. The Main Power Supply is based around a MAX1626[iii] (U2) Step-down DC to DC converter controller which is designed to control a step-down (or Buck) DC-to-DC converter at high efficiency across a wide load current range. This part also provides the 1.3V (nominal)

[i] MAX471 High-side Current Sense Amplifier, Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086, USA.
[ii] Precision Thermistor, Part number 10K3A1, Betatherm Ireland Ltd., Ballybrit Business Park, Galway, Ireland.
[iii] MAX1626 Step-down DC-DC Controller, Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086, USA.

reference voltage (V_{REF}) used by the MAX147 ADC and the power management comparators (U7). The component values chosen were based on design calculations included in the MAX1626 Datasheet.

The Main Power Supply in the FRPM is considerably over-rated for the FRPM's load requirements hence the converter is always operates in discontinuous inductor current mode. A description of the operation of the Buck converter can be found either in the MAX1626 datasheet or in[135].

The Buck Converter comprises switching MOSFET Q3, commutation diode D5, and storage inductor L1. The capacitor bank C51, C52, C5 and C6 provide smoothing, bulk charge storage and switching noise suppression for the converter output. In practice it was found that C51 performed adequately on its own and C52 was not required. The capacitor bank comprising C53 and C61 reduces the peak current draw from the input source which can potentially damage batteries and which increases EMI emissions. Low MOSFET on-resistance (R_{ON}), inductor series resistance (R_S), diode forward voltage drop (V_F), and capacitor equivalent series resistance (ESR) are all important to maximize converter efficiency[135] and reduce converter switching interference[136] (both radiated emissions and supply line noise).

The MAX1626 controller manages the Buck Converter by determining the optimum times to turn the switch on and off. The controller senses the switch current via resistor R24 through the CS input (pin 6) while the output voltage is monitored by the OUT input (pin 1). By tying the SHDN input (pin 3) low the MAX1626 is forced permanently on.

If the output voltage falls 0.5% below the required voltage (3.3V) MOSFET Q3 is switched on. When the inductor current (sensed via R24) reaches a set maximum ($i_{L(MAX)}$= 100mV/R24), or the output voltage is in regulation, Q3 is switched off. To reduce switching noise, minimum on- and off-times (both 2μs) are enforced. This method of operation is termed "Pulse Frequency Modulation" where the converter only switches as needed to maintain regulation, resulting in a switching frequency that varies with the load current. This can significantly reduce the switching losses encountered in fixed-frequency converters, although this is not really an issue in this application since the load currents are relatively low. At the 5mA (drawn from Vcc) typical Standby current of the FRPM the Main Power Supply draws 3.6mA from a 5V supply, giving an efficiency of 92% with a quiescent current consumption of 340μA. With a Vcc load of 250mA efficiency is 95% giving a quiescent current consumption of 8.7mA. The increased losses are almost entirely a result of increased switching losses in the MOSFET Q3 and diode D5 due to the higher average switching frequency. The FRPM Main Power Supply was tested with a load of 5A at a 5V input voltage and showed an efficiency of around 93%.

As the input voltage approaches the output voltage (for example, as the FRPM batteries discharge) the MOSFET conducts for longer and longer periods to maintain the output voltage until the MAX1626 can no longer regulate the output voltage. At this point the MOSFET is left switched on permanently (100% duty cycle), effectively connecting the load directly to the input

voltage source. If the input voltage begins to rise again, the MAX1626 will restore switching operation.

Because of the high switching frequency and high current pulses present in DC-to-DC converter designs care must be exercised when laying out the circuit on a printed circuit board (PCB). All tracks that carry switching currents must be kept as short and as wide as possible to minimize stray inductance that can reduce stability and efficiency[136]. The commutation current loop (D5, L1, C51 and C52 in Figure A2.6) must be kept as tight as possible to reduce radiated emissions. The use of a copper ground plane underneath this loop also helps to minimize radiated emissions. The design as implemented in the FRPM occupies $4cm^2$ of PCB area with most of this space taken up by the capacitors C51 and C52 and inductor L1.

A2.3.4 Radio Power Supply

The efficiency and output regulation requirements placed on the Radio Power Supply are more relaxed than for the Main Power Supply (Section A2.3.3) since the radio is only operational for short periods of time and the radio contains its own internal power supply regulators to stabilize and smooth the incoming supply. This supply, labeled Vradio in the circuit diagrams, supplies the 6V 250mA required by the radio module and is switched on and off via the RadioOn signal from the CL-PS7110.

The Radio Power Supply is based around a MAX608[i] (U3) Step-up DC to DC converter controller which is designed to control a step-up (or Boost) DC-to-DC converter at high efficiency across a wide load current range. The component values chosen were based on design calculations included in the MAX608 Datasheet. This power supply is considerably overrated for the radio module's load requirements hence the converter is always operates in discontinuous inductor current mode. A description of the operation of the Buck converter can be found either in the MAX1626 datasheet or in[135].

In the FRPM, the Radio Power Supply Boost converter comprises switching MOSFET Q4, commutation diode D6, and storage inductor L2. The capacitor bank C64 and C8 provide smoothing, bulk charge storage and switching noise suppression for the converter output. The capacitor bank comprising C60 and C54 reduces the peak current draw from the input source which can potentially damage batteries and increases EMI emissions[136]. Low MOSFET on-resistance (R_{ON}), inductor series resistance (R_S), diode forward voltage drop (V_F), and capacitor equivalent series resistance (ESR) are all important to maximize converter efficiency and reduce converter switching interference (both radiated emissions and supply line noise).

Unlike the Buck converter which has the switch in the main current path, the Boost converter cannot isolate its output from its input since a current path always exists through the inductor and commutation diode. For this reason it was necessary to provide the high-side switch comprising

[i] MAX608 Step-up DC-DC Controller, Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086, USA.

Q5B and Q2 to ensure that when the radio is turned off it is disconnected from the power source. By tying the SHDN input (pin 4) low the MAX608 will start as soon as the high-side switch is turned on.

The MAX608 controller manages the Boost Converter by determining the optimum times to turn the MOSFET switch on and off. The controller senses the switch current via resistor R25 through the CS input (pin 8) while the output voltage is monitored by the FB input (pin 3) through the voltage divider R26/R9. C55 provides feed-forward compensation to the controller to improve voltage regulation. The MAX608 is powered by the converter output through the OUT input (pin 2). It can be seen that when power is initially applied to the converter (Vout=0) the commutation diode will conduct, bringing the output voltage up to a level close to the input voltage. This voltage is sufficient to supply the MAX608 which then starts switching to bring the output voltage up to its desired level.

If the output voltage falls below the required voltage (6.0V) MOSFET Q4 is switched on. When the inductor current (sensed via R25) reaches a set maximum ($i_{L(MAX)}$= 100mV/R25) or the output voltage is in regulation, Q4 is switched off. To reduce switching noise and component stress, a maximum on-time (16μs) and minimum off-time (2.3μs) is enforced. This method of operation is termed "Pulse Frequency Modulation" where the converter only switches as needed to maintain regulation, resulting in a variable switching frequency that varies with the load. This significantly reduces the switching losses encountered in fixed-frequency converters, although this is not really an issue in this application since the radio module only operates for short periods of time.

As the input voltage approaches the output voltage the MOSFET conducts for shorter and shorter periods until the MAX608 can no longer regulate the output voltage. At this point the MOSFET is left switched off permanently (0% duty cycle), effectively connecting the load directly to the input voltage source. If the input voltage begins to fall again, the MAX608 will restore switching operation.

The Radio Power Supply was constructed on the FRPM PCB with similar considerations to the Main Power Supply (see Section A2.3.3). The design as implemented in the FRPM occupies 4.5cm^2 of PCB area with most of this space taken up by capacitors C60 and C64 and inductor L2.

A2.3.5 Analog Power Supply

A separate linear power supply was provided to supply the MAX147 ADC and any input/output expansion circuitry with a voltage source free of the switching and digital noise associated with the main logic supply. The MAX8863[i] (U4) Linear Regulator provides this quiet 3.15V Vaa supply at up to 100mA.

[i] MAX8863 Linear Regulator, Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086, USA.

A2.4 Peripherals

The FRPM Main Board contains on-board support for several peripherals. These are the RS232 serial communications interface, the Radio Module, an Infra-Red communications interface, an Analog-to-Digital Converter, and a Liquid Crystal Display (LCD) interface. The peripherals section also includes the Input/Output Expansion Interface which allows extra devices and functions to be added to the FRPM.

A2.4.1 RS232 and Radio Channels Multiplexer

The RS232 and Radio Module interfaces form a major part of the FRPM peripheral circuitry. Because of the requirement for two communications channels in the FRPM both the RS232 and Radio interfaces had to be multiplexed onto the single channel that the CL-PS7110 provides.

The Radio Module is supplied by the 6.0V Vradio supply which is regulated inside the module to power the 5V-level internal circuitry. This gives the Radio Module a maximum output voltage level of 5V. This voltage is not compatible with the $V_{IH\ (max)}$ (maximum input voltage) level for the CL-PS7110 of 3.6V (which is typical of CMOS circuitry operating on a 3.3V supply). Hence the Radio Module outputs must be translated to reasonable 3.3V logic levels.

The Radio Module has stated maximum V_{IL} (input low voltage) and minimum V_{IH} (input high voltage) levels of 0.9V and 3.5V respectively. The high level is not compatible with the matching minimum V_{OH} (output high voltage) levels for the CL-PS7110 of 3.0V (also typical of CMOS circuitry). Hence the Radio Module inputs must be translated to higher levels from the 3.3V logic levels. Figure A2.11 and Figure A2.12 show the relative signal levels between the Radio Module and the CL-PS7110.

The problem of level translation from the CL-PS7110 to the Radio Module was solved by using an HC-series logic part, the 74HC244[i] Octal Buffer (U10). When operated from a 5V supply, this device has a maximum V_{IL} specification of 1.0V and a minimum V_{IH} of 2.7V, both of which are compatible with the CL-PS7110 outputs. The maximum V_{OL} (output low voltage) and minimum V_{OH} specifications are 0.1V and 4.9V respectively, both of which are compatible with the Radio Module inputs. By powering this device from the V_{RADIO} supply via two diodes D11 and D10 to drop the supply to around 5V, a suitable level translation circuit was found.

[i] 74HC244, Octal Buffer/Line Driver, Philips Semiconductor, 811 East Arques Ave., Sunnyvale, CA 94088-3409, USA.

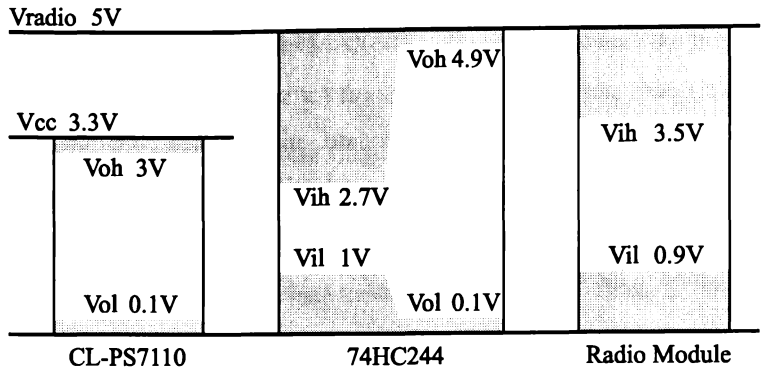


Figure A2.11 – CL-PS7110 to Radio Module Level Translation

The problem of level translation from the Radio Module to the CL-PS7110 was solved by using the same functional device as above but from the LVC-series, the 74LVC244[i] Octal Buffer (U12) which has 5V-level compatible inputs when operating from a 3.3V supply. This part has a maximum V_{IL} specification of 0.8V and a minimum V_{IH} of 2.0V, both of which are compatible with the Radio Module outputs, while the V_{OL} and V_{OH} specifications are 0.2V and 3.1V respectively when the device is operating from a 3.3V supply, both which are compatible with the CL-PS7110 inputs.

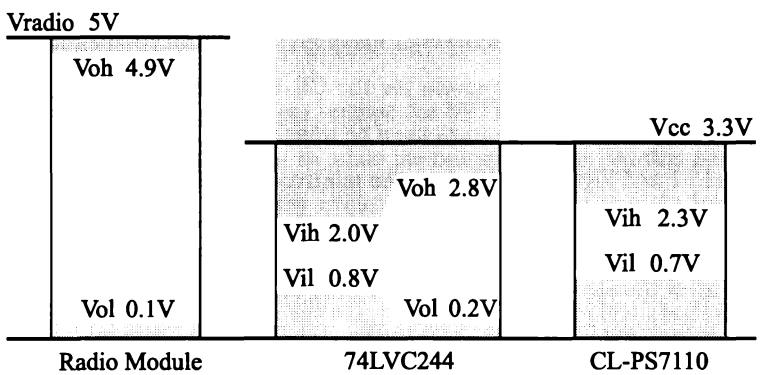


Figure A2.12 – Radio Module to CL-PS7110 Level Translation

The 74x244 devices have tri-state outputs which effectively become disconnected from their loads when the devices nG input (pins 1 and 19) are low. This allows a multiplexer function to select the appropriate communications channel to be combined with level translation. The nSerialEn/RadioEn signal from the CL-PS7110 determines which channel is selected. If this signal is low, the outputs of U12B and the RS232 transceiver U11 (which also has tri-state outputs enabled by the nEN input (pin 23) being low) are enabled while the outputs of U12A and U10B are disabled through the inverter consisting of transistor Q6. This effectively connects the RS232 transceiver to the CL-PS7110 and disconnects the Radio Module. If the

[i] 74LVC244 Octal Buffer/Line Driver, Philips Semiconductor, 811 East Arques Ave., Sunnyvale, CA 94088-3409, USA.

nSerialEn/RadioEn signal is high the RS232 transceiver is disconnected while the Radio Module is connected.

The pull-up resistors RP5B-H and R15-R18 ensure that their attached signal lines are held in their default (safe) high levels when the corresponding output driver is disabled as these signals cannot be allowed to float.

Although this design appears to be a relatively convoluted approach, it occupies only $1.5cm^2$ of PCB area and has performed well in practice.

A2.4.2 RS232 Communications Interface

The RS232 communications interface is based around the MAX3241 RS232 Transceiver (U11) which can provide true RS232C standard communications at rates of up to $120kbs^{-1}$. Of the five available signal receivers in this part only three are used: one each for the RxD (Receive Data), DSR (Data Set Ready), and CTS (Clear to Send) inputs. The three available signal transmitters are used for the TxD (Transmit Data), DTR (Data Terminal Ready), and RTS (Request to Send) outputs. These six signals are made available at the RS232 Connector (J8) for direct connection to a computer or other communications device. The RI (Ring Detect) input signal present in the RS232C standard is not implemented while the DCD (Data Carrier Detect) signal is connected internally to the RTS signal. The RS232 Connector pin assignments are designed to eliminate the need for a Null Modem cable between the FRPM and the external device. The pin assignments for the RS232 Connector (J8) and a computer 9-pin RS232C connector are presented in Table A2.7.

Pin #	FRPM (J8) RS232C Signal	Computer RS232C Signal
1	RTS (Output)	DCD (Input)
2	TxD (Output)	RxD (Input)
3	RxD (Input)	TxD (Output)
4	DSR (Input)	DTR (Output)
5	GND	GND
6	DTR (Output)	DSR (Input)
7	CTS (Input)	RTS (Output)
8	RTS (Output)	CTS (Input)
9	Not present	RI (Input)

Table A2.7 – FRPM RS232 Connector and Computer 9-pin RS232C Connector Pin Assignments

The RxD and TxD signals connect directly to their respective CL-PS7110 UART pins (pins 46 and 43 respectively). The DTR and RTS outputs are obtained from the SerialDTR and SerialRTS general purpose outputs from the CL-PS7110 (see Section A2.4.7). The DCD, DSR and CTS inputs have dedicated pins on the CL-PS7110 and any activity on two of these inputs (CTS and DSR) can be used to cause an interrupt request (IRQ) to be generated allowing communications channel flow control to be implemented.

If the jumper R29 is installed, any activity on the RxD input will cause the processor to exit Standby mode since the R1OUTB and R2OUTB outputs on the RS232 transceiver are permanently enabled (not affected by the nEN input).

Capacitors C34-C36, C44 and C45 form part of the charge-pump converter used internally by the MAX3241 to generate the RS232C voltage levels from the 3.3V supply voltage. A low level on the nSHDN input (pin 22) disables this charge-pump, forcing the MAX3241 to enter a low power standby state.

The noise suppression devices FC1-FC7 protect the MAX3241 from external ESD events, reduce the possibility of emissions from the FRPM module, and protect the FRPM from externally generated interference. The ground plane beneath the connector is isolated from the main ground plane, forcing any ground current from outside the FRPM to be filtered by FC1.

A2.4.3 Radio Interface

The radio module used in the FRPM is the WIT2400M[i], a 2.4GHz spread-spectrum design which can support a network of up to 255 modules with a network data rate of 250kbs^{-1}. Up to 16 networks can be in operation simultaneously. The radio modules support full error correction via an internal automatic repeat request (ARQ) protocol and data buffering with transmission retry if a module temporarily drops from the network. The modules typically consume 180mW of power while linked to the network and up to 850mW during transmission and have proved in practice to operate reliably at ranges of up to 2km even around obstacles such as low hills and buildings. When waiting between experiments the FRPM typically uses the radio in 5 second bursts every few minutes to "check-in" to the base station for new messages to reduce the unit's average power consumption.

The frequency hopping spread-spectrum design of these radio modules ensures maximum resistance to noise and multi-path fading, and robustness in the presence of interfering signals, while operation in the 2.4GHz International Scientific and Medical (ISM) band allows license-free operation and worldwide compliance. The radio data network is based on a "star" topology with the base station, which is connected to the user's computer, acting as the synchronizing master with all modules in the FRPMs acting as slave units.

The Radio Module interface requires the TxD, RxD, RTS, DTR, CTS and DSR signals with the addition of the DCD input and nConfig output (Radio Configure) signals. The DCD signal indicates when the Radio Module has successfully acquired the base station signal while a falling edge on the nConfig output forces the Radio Module to enter its configuration mode. The radio module is connected to the FRPM via the Radio Module Data Connector (J2) and the Radio Module Power connector (J6). The pin assignments for these connectors and the radio module are presented in Table A2.8.

[i] WIT2400M Spread Spectrum Radio, Digital Wireless Corp., 1 Meca Way, Norcross, GA 30093, USA.

The radio module is switched on by the RadioOn signal from the CL-PS7110 going high and selected by the nSerialEn/RadioEn signal going high. Initially the System Software should hold the TxD, nConfig, RTS and DTR radio module inputs high to hold the radio module in its suspend state which holds the radio module outputs RxD, DCD, CTS and DSR high. Bringing the DTR signal low wakes the radio which responds by pulling its DSR output low to indicate the radio has woken. When the radio is ready it will pull its CTS output low to indicate it can accept data at its TxD input for transmission. If the radio is within range of the base station and the transmission path is not obstructed then the radio will synchronize itself with the network frequency hopping pattern and pull its DCD output low to indicate it has successfully linked to the base station. If the radio loses contact with the base station the DCD output will go high until contact is re-established. If the transmit buffer inside the radio module begins to fill (due perhaps to loss of contact with the base station) the CTS output will go high to advise the processor to hold back data to avoid buffer overflow. The processor can request the radio stop sending data to the processor from its RxD output and instead buffer it internally by placing the RTS signal high. The nConfig signal is not currently used in the FRPM since the radio modules are pre-programmed before connecting them to the FRPM.

	Pin #	FRPM Radio Module Signal	Radio Module Signal
J2	1	GND	GND
	2	TxD (Output)	TxD (Input)
	3	RxD (Input)	RxD (Output)
	4	nConfig (Output)	nCFG (Input)
	5	RTS (Output)	RTS (Input)
	6	DTR (Output)	DTR (Input)
	7	DCD (Input)	DCD (Output)
	8	CTS (Input)	CTS (Output)
	9	DSR (Input)	DSR (Output)
	10	GND	TXC (Input)
	11	Not present	RXC (Output)
	12	Not present	Reserved
	13	Not present	Reserved
J6	1	GND	GND
	2	V+	Vcc

Table A2.8 – FRPM Radio Module Connectors and Radio Module Pin Assignments

To correctly shut down the radio module without loss of data, the DTR signal should be placed high and 100ms (four hop durations) later the radio should be deselected by sending the nSerialEn/RadioEn signal going low and the power supply V_{RADIO} switched off.

The radio modules provide a transparent communications link from the FRPM to the base station - all data sent to the radio will be sent to the base station and no extra information is required. In effect the radio link looks the same to the FRPM as the RS232 interface. However to communicate between the user's computer and the base station information must be added to the

data stream to distinguish which FRPM radio modules should receive which data or which FRPM radio modules sent which data. To provide this information a special packet mode is used between the computer and the base station. To distinguish data streams to or from different remotes the packet-based format shown in Figure A2.13 is used.

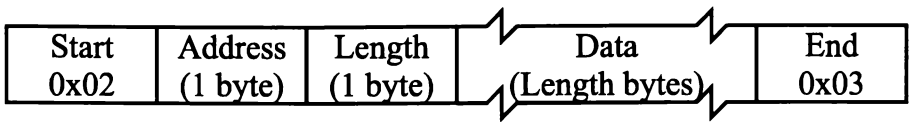


Figure A2.13 – Radio Base Station Packet Format

The STX character (0x02) marks the beginning of a packet and the ETX character (0x03) marks the packet end. In the case of the base station receiving data from an FRPM, the Address field is the source address of the FRPM that sent the data. In the case of the base station sending data to an FRPM, this field is the destination address of the FRPM that the data is meant for. The Length field contains the number of data bytes in the packet, with a range of 0 to 255 bytes. Only the Data section is included in the Length count. The Data section holds the data being either sent or received.

A2.4.4 Infra-red Communications Interface

Provision is made on the FRPM Main Board for adding an Infra-red Data Association (IrDA) compliant data link module such as the HSDL-1001[i]. The CL-PS7110 supports IrDA link data rates of up to 115.2kbs^{-1}. The IrDA module should be attached to J9. Currently the IrDA protocols are not supported by the System Software.

A2.4.5 ADC Interface

The MAX147 (U13) analog-to-digital converter allows the FRPM to monitor eight input voltages with 12-bit accuracy. Four of the input channels are reserved for internal measurements while the other four are brought out to the Input/Output Expansion connector (J4) for general use. Table A2.9 presents the channel assignments. The Physical Channel number is the actual input channel while the MAX147 Channel number is the number that needs to be sent to the MAX147 to get the corresponding physical channel. e.g. To measure the voltage on input CH3 the channel number that needs to be sent to the MAX147 is 5.

The reference voltage for the MAX147 is obtained from the V_{REF} output of the MAX1626 main voltage supply controller. By removing jumper R31 it is possible to use an external reference which should not exceed 3.15 volts. The reference voltage sets the range of the input voltage.

[i] HSDL-1001 Infrared IrDA Compliant Transceiver, Hewlett-Packard Corp., 3000 Hanover Street, Palo Alto, CA 94304, USA.

For the current system the ADC input voltage range is 0 to 1.3V. Where possible, input voltages should be ratiometric with respect to V_{REF} to maximize the measurement accuracy.

Physical Channel #	MAX147 Channel #	Parameter
0, 1, 2, 3	0, 4, 1, 5	Available to user
4	2	FRPM Input voltage
5	6	FRPM Battery current
6	3	FRPM Internal temperature
7	7	FRPM Battery voltage

Table A2.9 – FRPM Analog Channel Allocations

The MAX147 is attached to the CL-PS7110 synchronous serial interface and is controlled via the SYNCIO internal register. The correct configuration for reading the MAX147 from the CL-PS7110 is Frame Length = 25 bits, SMPCLK disabled and TXFRMEN bit set. The ADC configuration word should be set to `0x598F` (start conversion, unipolar mode, single ended, external clock) with the appropriate MAX147 channel number occupying the 3 least significant bits.

A2.4.6 LCD Interface

Provision is made on the FRPM Main Board for adding an LCD display panel with resolutions of up to 1024 by 256 pixels and 16 colours. The LCD interface comprises the circuitry around socket J7. R35 and C57 help minimize jitter in the pixel clock signal (CL2).

Although no display is currently incorporated in the FRPM, a Hitachi LMG6912[i] 320 by 240 pixel, 16 level greyscale LCD panel has been used during development work as an additional debugging communications channel.

[i] LMG6901 LCD Panel, Hitachi Semiconductor, 2000 Sierra Point Parkway, Brisbane, CA 94005, USA

A2.4.7 Reserved General Purpose Input and Output Signals

Of the 36 general purpose input/output pins provided on the CL-PS7110, 20 are reserved for controlling the FRPM Main Board while the remaining 16 are available to the user. Table A2.10 lists the reserved ports. The free ports available for user applications are made available via the Input/Output Expansion interface described in the following section.

Port	Name	Direction	Function
PC0	SerialRTS	Output	RS232 RTS output signal
PC1	SerialDTR	Output	RS232 DTR output signal
PC2	nSerialEn/RadioEn	Output	A low level selects the MAX3241 RS232 transceiver as the communications channel, a high level selects the radio module
PC3	IRDAShutdown	Output	IrDA module shutdown signal
PC4	nRadioConfig	Output	A low level places the radio module in configuration mode
PC5	BattCurrentDir	Input	A low level indicates battery is discharging, high indicates battery is charging
PC6,7	DRAMSpeed1,2	Input	Used for detecting DRAM memory speed (obsolete)
PD0	nSerialShutdown	Output	High level places MAX3241 RS232 transmitter section in low power state
PD1	ExternalWakeup	Input	Allows monitoring of memory expansion wakeup signal
PD2	ExpansionWakeup	Input	Allows monitoring of IO expansion wakeup signal
PD3	FlashReady	Input	Allows monitoring of flash memory program/erase functions
PD4	BattCurrentShutdown	Output	High level places MAX471 in low power state
PD5	ChargeBattery	Output	High level allows solar panel to charge battery
PD6	LCDEnable	Output	LCD panel enable signal
PD7	RadioOn	Output	High level turns on the radio power supply
PE0	BootSelect	Input	High level forces Cl-PS7110 to boot from external EPROM
PE1-3	DebugLED0-2	Output	Debug LED drive signals

Table A2.10 – Reserved Input and Output Port Assignments

A2.4.8 Input/Output Expansion Interface

The I/O Expansion Connector (J4) provides access to the unused general purpose input/output signals and ADC input channels. The CL-PS7110 keyboard scanning signals are also made available. The pin allocations for the 40-pin I/O Expansion Interface connector (J3) are presented in Table A2.11.

Pin #	Name	Direction	Description
1, 8	AGND	Power	Analog circuitry quiet ground
2	VREF	Power	ADC reference voltage
3-6	Ain3-Ain0	Analog Input	ADC input channels (1.3V max.)
7	Vaa	Power	Analog circuitry quiet supply
9	Wakeup	Input	Rising edge forces system into Operating state
10	KeypadInterrupt	Input	High level will cause interrupt request (IRQ)
11-18	COL0-7	Output	Keyboard column scanning drives
19, 29, 39	GND	Power	Ground
20	Vcc	Power	Main logic supply
21-28	PB0-PB7	Input/Output	General purpose I/O
30, 40	Vin	Power	Unregulated power supply input
31-38	PA0-PA7	Input/Output	General purpose I/O

Table A2.11 – I/O Expansion Connector Pin Assignments

The direction of the PA0-7 and PB0-7 signals can be configured through the PADDR and PBDDR registers. These ports can be read or written through the PADR and PBDR registers.

A2.5 FRPM Main Board Implementation

The FRPM Main Board was implemented as a 6-layer PCB with components mounted on both sides. Surface mount components were used throughout to minimize board size. One layer was reserved for use as a ground plane to minimize ringing on signal lines by reducing power supply inductance[136]. Ideally a power supply (Vcc) plane should also have been present for optimum power supply decoupling and EMI performance, however this would have required an 8-layer board which was considered too expensive. Despite the lack of this power plane, no EMI or power supply decoupling problems were detected. The ground plane was split around the ADC to reduce ground plane noise caused by the digital circuitry, and around the RS232 connector to help provide electrostatic discharge (ESD) and external interference protection. A wide ground trace was placed around the perimeter of the board on all layers to help with ESD protection. This ground trace also connects to the four mounting holes to provide a return path for any ESD current to the FRPM case.

Signal traces were kept as short as possible to minimize transmission line effects particularly in the memory interfaces. Power traces were kept as wide as possible to minimize inductance and maximize capacitive coupling to the ground plane which together help reduce supply noise. Particular care was taken with laying out the switching power supplies since these can be a major source of system noise (see Sections A2.3.3 and A2.3.4).

The completed PCB measures 110mm by 70mm. A picture of the completed board is shown in Figure A2.14. The largest visible component on the board is the CL-PS7110 processor, the next biggest components being the two Am29LV800B Flash Memory devices. The 16MB DRAM module plugs into the SO-DIMM socket on the top side of the board. The board area is nearly equally shared between the processor and memory systems, and the power management and power supply systems.

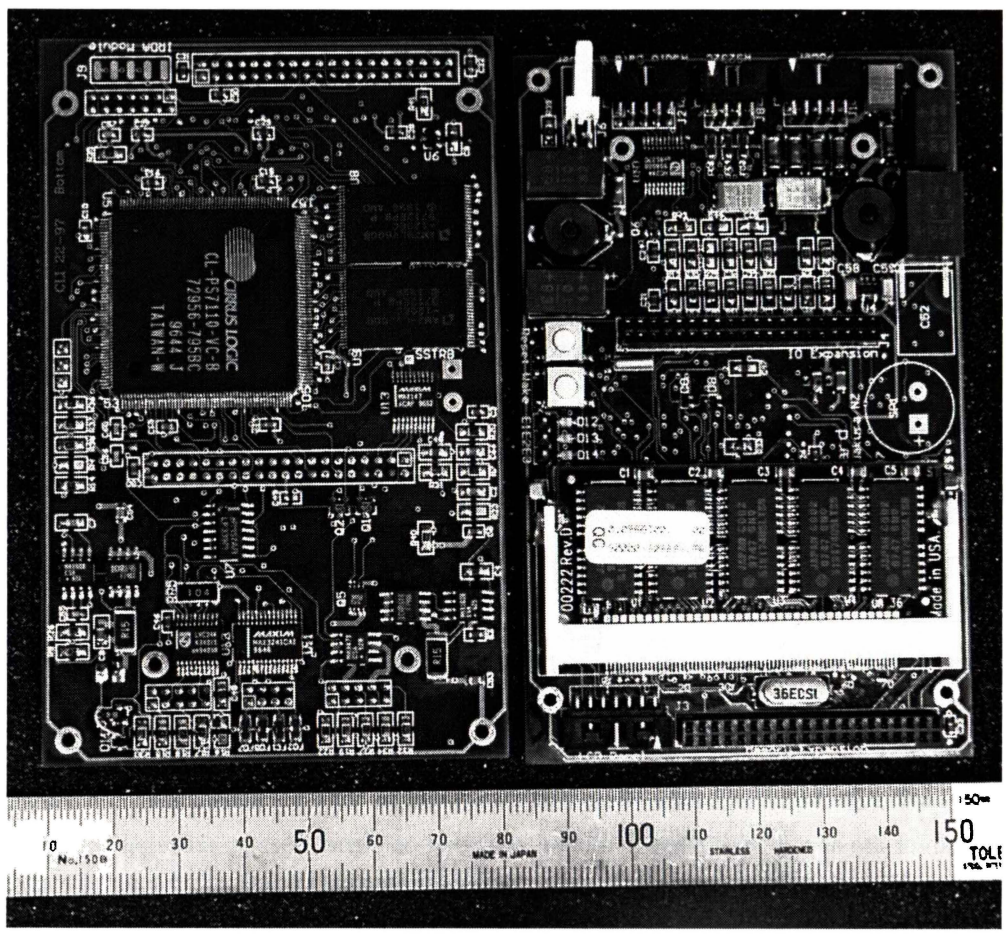


Figure A2.14 – The Complete FRPM Main Board

The two areas with greatest opportunity for reducing the size of the FRPM Main Board are simplification or higher integration of the power management section and the combining of the processor core with a large on-board DRAM memory array in a single part. Both of these opportunities were starting to appear within 12 months of designing the system. Together these techniques could result in a board half the size of the current design.

A2.6 Physiological Interface

The FRPM records physiological signals via a small Physiological Interface board which contains all the necessary signal conditioning circuitry for several different physiological signal sources. In its current form the interface board provides support for the acquisition of EEG, Evoked Response, and ECG signals, and provides support for the measurement of physical activity, respiration rate, and temperature.

The current Physiological Interface measures 44mm by 60mm and is designed to be plugged directly into the FRPM Input/Output Expansion Connector (J4) on the FRPM Main Board. The Physiological Interface is constructed as a separate plug-in sub-assembly to allow maximum flexibility for different applications of the FRPM and to assist in future interface upgrades since on its own it is a relatively low cost part of the overall system and is easy to manufacture.

Figure A2.15 through Figure A2.19 show the Physiological Interface circuitry.

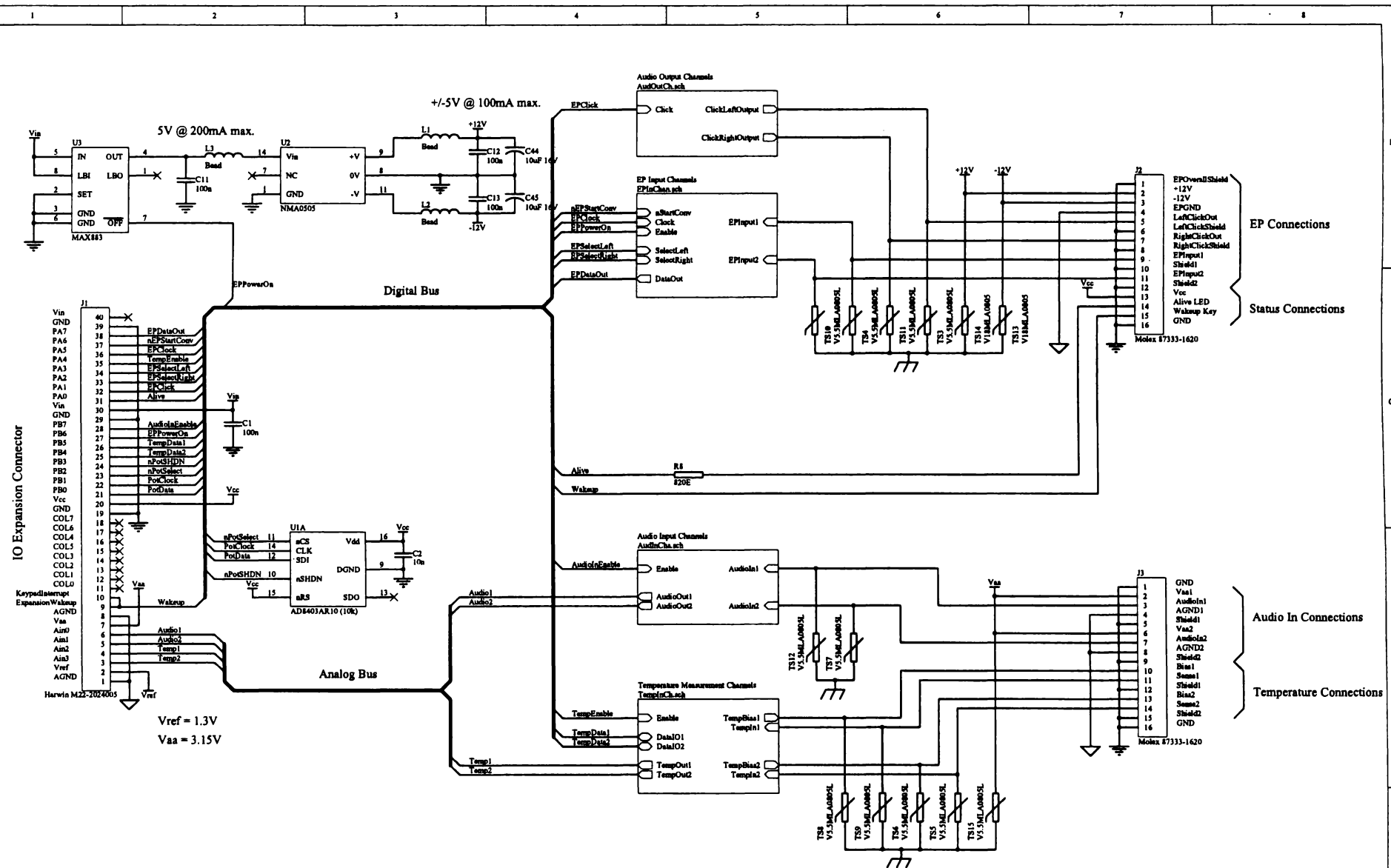


Figure A2.15 – Physiological Interface Functional Blocks

Title: FRPM Input Interface		Engineering Development Group Hort Research NZ Ltd Ruakura Research Centre Ruakura Road Hamilton, New Zealand
Drawn by: Phil Harris		
Size: A4	File: L:\WELFARE\FrpmDesign\Elec\FrpmInterface\Schematics\InputMagnetics.sch	
Number:	Revision:	Date: 14-Dec-1999 Time: 20:47:57 Sheet 1 of 6

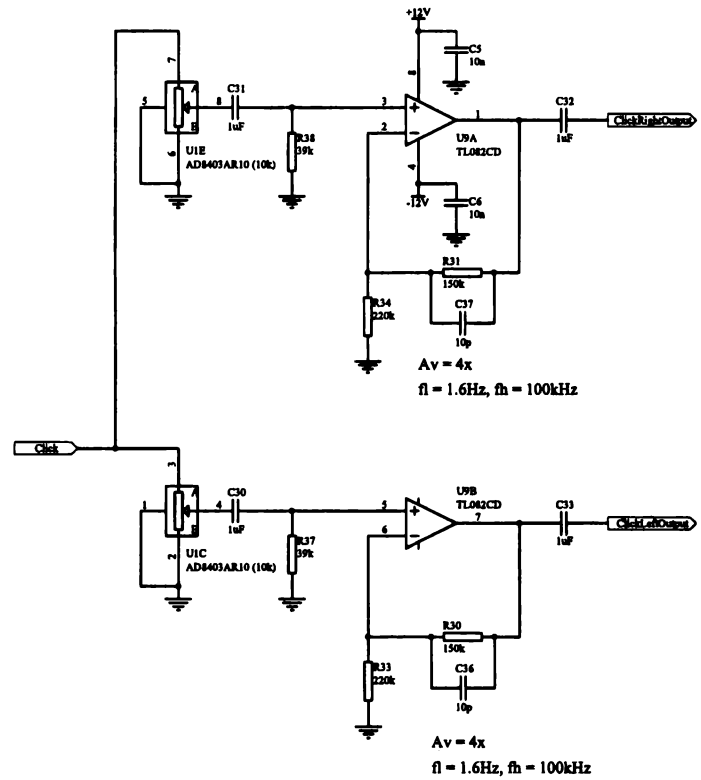
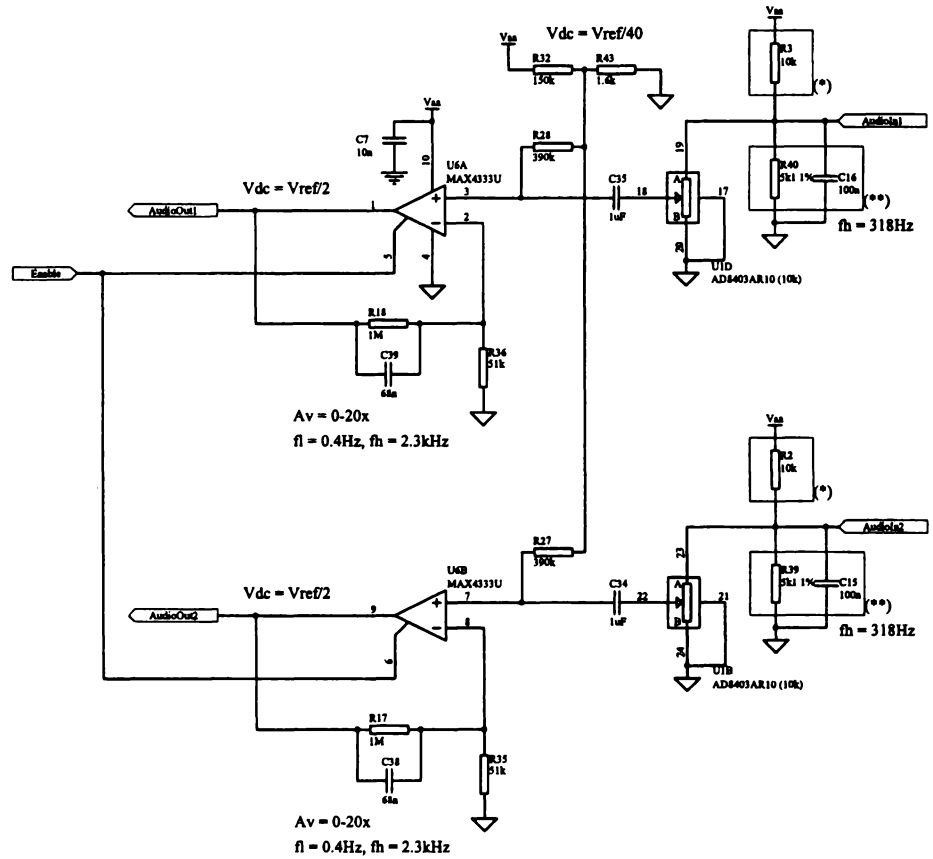


Figure A2.16 – Physiological Interface Audio Output Circuitry

Title: FRPM Audio Output Interface		Engineering Development Group Hort Research NZ Ltd Ruakura Research Centre Ruakura Road Hamilton, New Zealand
Drawn by: Phil Harris		
Size: A4	File: L:\WELPA\REV\pmDesign\ElecPipes\Interface\Schematics\A4\Obt#Block	
Number:	Revision:	Date: 14-Dec-1999 Time: 20:48:14 Sheet 3 of 6



This circuitry can use either electret microphones (respiration monitoring, rumen activity monitoring) or thermistors (respiration monitoring) as the input source.

Install components marked (*) when using microphone.

Install components marked (**) when using thermistors.

Figure A2.17 – Physiological Interface Audio Input Circuitry

Title	FRPM Audio Input Interface
Drawn by:	Phil Harris
Size: A4	File: L:\WELFARE\FrpmDesign\Elec\FrpmInterface\Schematics\A4\FRPMAudio.Sch
Number:	Revision:
Date: 14-Dec-1999 Time: 20:48:33	Sheet 4 of 6

Engineering Development Group
Hort Research NZ Ltd
Ruakura Research Centre
Ruakura Road
Hamilton, New Zealand

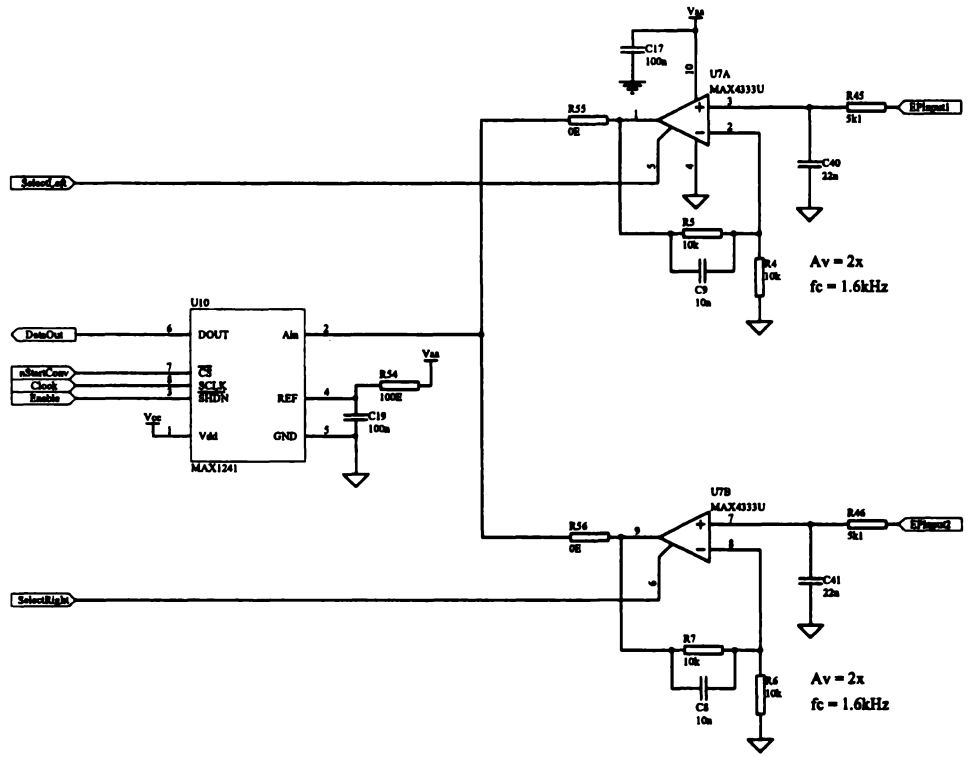


Figure A2.18 – Physiological Interface Evoked Response Input Circuitry

Input must be centered at Vss/4 with a maximum swing of Vss/4.

Conversion starts on falling edge of nStartConv.

Title: FRPM Evoked Response Input Channels	Engineering Development Group, Hort Research NZ Ltd, Ruakura Research Centre, Ruakura Road, Hamilton, New Zealand
Drawn by: Phil Herrin	
Size: A4 File: L:\WELFARE\PiguetDesign\Elec\PiguetInterface\Schematics\EPInputChannels.sch	
Number: Revision: Date: 14-Dec-1999 Time: 20:48:31	Sheet 5 of 6

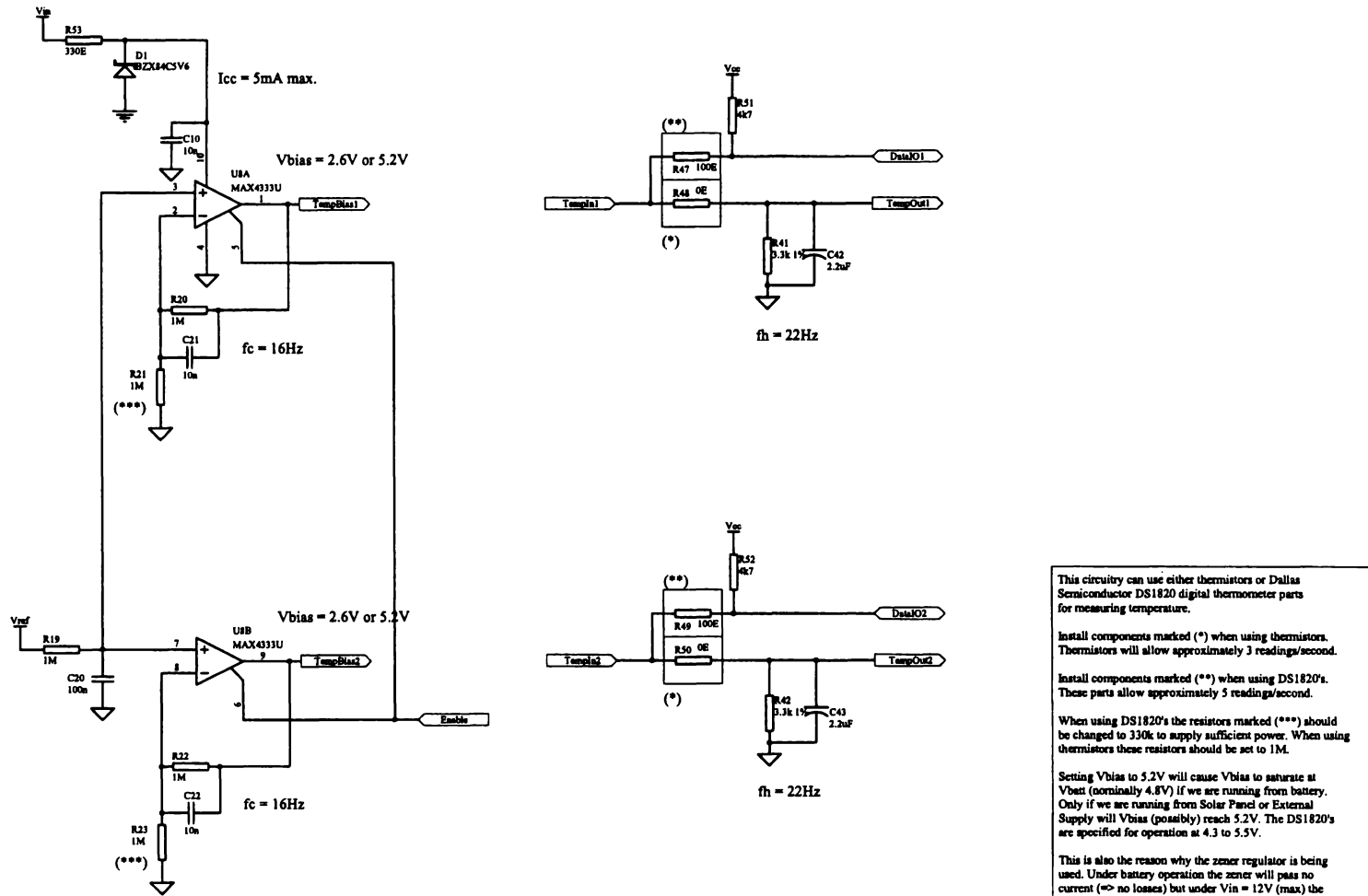


Figure A2.19 – Physiological Interface Temperature Measurement Circuitry

This circuitry can use either thermistors or Dallas Semiconductor DS1820 digital thermometer parts for measuring temperature.

Install components marked (*) when using thermistors. Thermistors will allow approximately 3 readings/second.

Install components marked (**) when using DS1820's. These parts allow approximately 5 readings/second.

When using DS1820's the resistors marked (***) should be changed to 330k to supply sufficient power. When using thermistors these resistors should be set to 1M.

Setting Vbias to 5.2V will cause Vbias to saturate at Vbatt (nominally 4.8V) if we are running from battery. Only if we are running from Solar Panel or External Supply will Vbias (possibly) reach 5.2V. The DS1820's are specified for operation at 4.3 to 5.5V.

This is also the reason why the zener regulator is being used. Under battery operation the zener will pass no current (=> no losses) but under Vin = 12V (max) the supply to the MAX4333 must be limited to less than 6.5V. Under worst case conditions the zener will be passing 20mA (Pzener = 110mW, Presistor = 200mW).

Title: FRPM Temperature Measurement Channels		Engineering Development Group Hort Research NZ Ltd Ruakura Research Centre Ruakura Road Hamilton New Zealand
Drawn by: Phil Harris		
Size: A4	File: L:\WELFARE\FrpmDesign\Elec\FrpmInterface\Schematics\TempBlock	
Number:	Revision:	Date: 14-Dec-1999 Time: 20:49:12 Sheet 6 of 6

The Physiological Interface plugs directly into the FRPM Main Board Input/Output Expansion Connector via the connector J1. The Expansion Connector signal assignments for the Physiological Interface are shown in Table A2.12. The signal functions are clarified in their related sections.

Table A2.13 and Table A2.14 presents the pin assignments of the two connectors on the Physiological Interface that connect to external sources or sensors. These signal functions are also clarified in their related sections. Connector J2 is mainly related to EEG/Evoked Response signal acquisition while J3 handles the various audio input and temperature measurement channels.

Pin #	Name	Direction	Description
1, 8	AGND	Power	Analog circuitry quiet ground
2	VREF	Power	ADC reference voltage
3	Ain3/Temperature2	Analog In	Temperature Sensor Input #2
4	Ain2/Temperature1	Analog In	Temperature Sensor Input #1
5	Ain1/AudioInput2	Analog In	Audio Input Channel #2
6	Ain0/AudioInput1	Analog In	Audio Input Channel #1
7	Vaa	Power	Analog circuitry quiet supply
9, 10	Wakeup	Input	External Wakeup Key Input
11-18	COL0-7	Output	Not used
19, 29, 39	GND	Power	Ground
20	Vcc	Power	Main logic supply
21	PA0/PotData	Output	Digital Potentiometer Data
22	PA1/PotClock	Output	Digital Potentiometer Clock
23	PA2/nPotSelect	Output	Digital Potentiometer Select (Active low)
24	PA3/nPotShutdown	Output	Digital Potentiometer Shutdown (Active low)
25	PA4/TempData1	Input	Digital Temperature Sensor Data Input #1
26	PA5/TempData2	Input	Digital Temperature Sensor Data Input #1
27	PA6/EPPowerOn	Output	Evoked Response Power On
28	PA7/AudioInEnable	Output	Audio Input Channels Enable
30	Vin	Power	Unregulated power supply input
31	PA0/StatusLED	Output	Status LED drive output
32	PA1/ERClick	Output	Evoked Response Click Stimulus
33	PA2/ERSelectLeft	Output	EEG Left Input Channel Select
34	PA3/ERSelectRight	Output	EEG Right Input Channel Select
35	PA4/TempEnable	Output	Temperature Sensor Bias Enable
36	PA5/ERADCClock	Output	EEG ADC Clock
37	PA6/nERADCConvert	Output	EEG ADC Start Conversion
38	PA7/ERADCData	Input	EEG ADC Result Data Input
40	Vin	Power	Not used

Table A2.12 – I/O Expansion Connector Pin Assignments for the Physiological Interface

Pin #	Name	Direction	Description
1	GND	Power	Ground
2	+5V	Power	+5V power supply
3	-5V	Power	-5V power supply
4	GND	Power	Ground
5	LeftClickOut	Analog Output	Evoked Response Left Stimulus
6	LeftClickShield	Power	Shield for above
7	RightClickOut	Analog Output	Evoked Response Right Stimulus
8	RightClickShield	Power	Shield for above
9	EPInput1	Analog Input	Evoked Response Input Channel #1
10	Shield1	Power	Shield for above
11	EPInput2	Analog Input	Evoked Response Input Channel #2
12	Shield2	Power	Shield for above
13	Vcc	Power	+3.3V power supply
14	StatusLED	Output	Status LED drive output
15	WakeupKey	Input	Wakeup key input
16	GND	Power	Ground

Table A2.13 – Physiological Interface J2 Connector Pin Assignments

Pin #	Name	Direction	Description
1	GND	Power	Ground
2	Vaa	Power	+3.15 quiet analog power supply
3	AudioIn1	Analog Input	Audio input channel #1
4	AGND	Power	Quiet analog ground
5	AudioIn1Shield	Power	Shield for audio input channel #1
6	Vaa	Power	+3.15 quiet analog power supply
7	AudioIn2	Analog Input	Audio input channel #2
8	AGND	Power	Quiet analog ground
9	AudioIn2Shield	Power	Shield for audio input channel #2
10	TempBias1	Power	Bias drive for temperature sensor #1
11	TempSense1	Analog Input	Input for temperature sensor #1
12	TempShield1	Power	Shield for temperature sensor #1
13	TempBias2	Power	Bias drive for temperature sensor #2
14	TempSense2	Analog Input	Input for temperature sensor #2
15	TempShield2	Power	Shield for temperature sensor #2
16	GND	Power	Ground

Table A2.14 – Physiological Interface J3 Connector Pin Assignments

The Physiological Interface contains an AD8403[i] (U1) Four-channel Digital Potentiometer that allows amplifier gains and signal levels to be adjusted in different parts of the Physiological Interface circuitry. The channel allocations for this device are shown in Table A2.15. Each potentiometer in the AD8403 has an end-to-end resistance of 10kΩ over 256 steps, giving a resolution of 39Ω/step.

The desired position for a given channel is loaded into the AD8403 by clocking a data word into the device using its SDI, CLK and nCS pins which are connected to the PotData, PotClock and nPotSelect signals (see Table A2.12) respectively. Because these signals are connected to the CL-

[i] AD8403 4-channel Digital Potentiometer, Analog Devices Inc., One Technology Way, Norwood, MA 02062-9106, USA.

PS7110 general purpose I/O pins, each bit must be individually clocked into the AD8403 under software control. The data format and timing diagram for this transfer is shown in Figure A2.20. Of the ten data bits making up a full transfer, the first two bits indicate the potentiometer channel with the next 8 bits representing the desired channel position. While the maximum allowable clock frequency is 10MHz the (unoptimized) FRPM System Software only achieves around 150kHz, with a full transfer taking approximately 70μs.

Potentiometer Channel #	Function
0	Audio Input #2 Gain
1	Evoked Response Left Stimulus Intensity
2	Audio Input #1 Gain
3	Evoked Response Right Stimulus Intensity

Table A2.15 – Physiological Interface Digital Potentiometer Channel Allocations

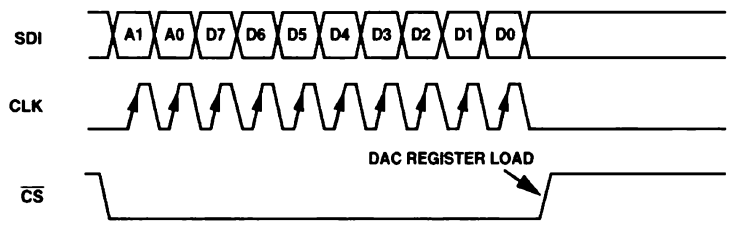


Figure A2.20 – Physiological Interface Digital Potentiometer Data Transfer Timing

If the nSHDN input of the AD8403 digital potentiometer is pulled low, each of the four channels is set their minimum resistance positions (0Ω) and the AD8403 enters a low power standby state. When the nSHDN input returns high each of the four channels is returned to its previous position.

The Physiological Interface also provides a wakeup input and an output for driving a status LED, both of which are accessible from the outside of the FRPM housing, allowing the user to wake the FRPM from its Standby state and allowing the FRPM to present rudimentary status information to the user in the form of a flashing LED.

All of the Physiological Interface external input and output signals are protected from over-voltage or electrostatic discharge events by a combination of semiconductor transient suppressors and current limiting resistors.

A2.6.1 EEG Acquisition Inputs

The Physiological Interface provides support for recording two channels of EEG signals at up to 10kHz sample rate per channel. Due to electrical screening requirements the pre-amplifier for these signals (see Chapter 4) is contained on another PCB in a separate shielded compartment within the FRPM housing. ECG signals can also be recorded by the EEG acquisition circuitry, however the QRS complex of the signal is likely to be clipped due to the high gain of the pre-amplifier.

The EEG pre-amplifier is powered from the FRPM by the power supply consisting of the MAX883[i] 5V Linear Regulator (U3) and the NMA0505[ii] +/-5V DC-to-DC converter (U2) which converts the nominal FRPM input voltage to the +/-5V required by the pre-amplifier. The MAX883 limits the input voltage to the NMA0505 to 5V regardless of the FRPM input voltage and also functions as a switch via its nOFF input (ERPowerOn signal) which when low effectively disconnects the NMA0505, turning off the pre-amplifier. Although the NMA0505 is a relatively inefficient converter with an efficiency of only 80%, the pre-amplifier is used only infrequently, making the efficiency gains of a more complicated and costly supply less attractive. The filters comprising inductors L1-L3 and capacitors C11-C13 reduce switching noise interference from the NMA0505.

The EEG acquisition circuitry consists of the MAX4333[iii] dual op-amp (U7) which amplifies the incoming signal by a factor of 2. This op-amp also performs an anti-aliasing function with a second-order Butterworth response with a corner frequency of 1.6kHz, and functions as a multiplexer to allow either of the two input channels to be sampled by the MAX1241[iv] 12-bit analog-to-digital converter (U10). The desired input channel is selected by setting either the EPSelectLeft or EPSelectRight signal to a high level which enables the output driver stage of U7A or U7B respectively. Note that only one of either U7A or U7B should be enabled at any one time. The MAX4333 amplifier is particularly suited to this design since it offers a rail-to-rail input and output range, low operating current, and allows an over-voltage condition on its inputs which is possible due to the EEG pre-amplifier operating from +/-5V supplies.

A sample is acquired by presenting a falling edge to the nCS input (nERADCConvert signal) of the MAX1241 ADC, waiting approximately 7.5µs for the conversion to complete, then clocking the resulting sample out of the MAX1241 via the SCLK and DOUT pins (ERADCClock and ERADCData signals), most significant bit first. Because these signals are connected to the CL-PS7110 general purpose I/O pins each bit must be individually clocked out of the MAX1241

---

[i] MAX883 Linear Regulator, Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086, USA.

[ii] NMA0505 Isolated DC-DC Convertor, Newport Components, Tanners Drive, Milton Keynes MK14 5NA, UK.

[iii] MAX4333 Dual Operational Amplifier, Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086, USA.

[iv] MAX1241 Serial 12-bit ADC, Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086, USA.

under software control. Figure A2.21 shows the related timing diagram. Note that the first bit is a dummy bit, the next 12 bits contain the actual sample data, and any further bits are always '0'.

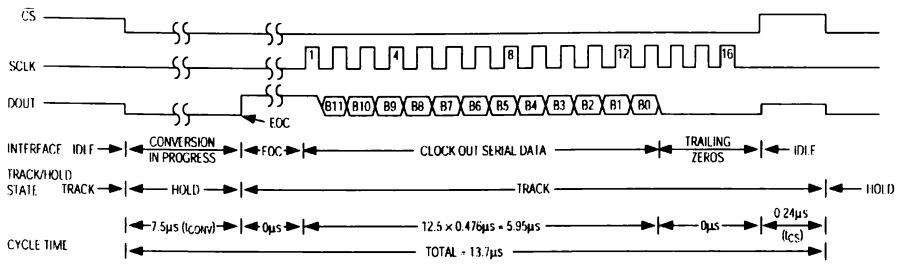


Figure A2.21 – Physiological Interface EEG ADC Interface Timing

The FRPM System Software achieves a bit rate of around 280kHz, with a full conversion cycle taking approximately 70µs allowing a theoretical sampling rate of 14kHz. Because of the overheads of data management the maximum rate is slightly lower. This approach creates 1µs of sampling jitter.

The effect of this jitter on Signal-to-Noise Ratio (SNR) and hence Effective Number of Bits (ENOB) can be calculated from the following two equations[137]:

$$SNR = 20\log_{10}\left(\frac{1}{2\pi f t_a}\right) \tag{A1.5}$$

$$ENOB = \frac{SNR - 1.76\text{dB}}{6.02\text{dB}} \tag{A1.6}$$

Where t_a is the RMS jitter time (also called aperture jitter) and f is the full-scale sinusoid input frequency. Given t_a =1µs and a desired resolution of 10 bits, we require an SNR of 58.44dB which implies a maximum input frequency of 127Hz for 10-bit accuracy.

A2.6.2 Evoked Response Stimulus Outputs

Two auditory click stimulus output channels with variable attenuation (0 to 100% over 256 steps) for Evoked Response acquisition are provided by the Physiological Interface. The sound pressure level produced by these outputs depends on the choice of loudspeaker: both piezoelectric and electromechanical speakers can be used.

The click source takes the form of a digital output, ERClick, from the CL-PS7110 which drives the inputs of two attenuators based around independent channels of the AD8403 digital potentiometer (U1). By AC coupling the output of the attenuators to the output driver amplifiers either positive (condensation) clicks or negative (rarefaction) clicks can be produced by the speakers. For instance by holding the ERClick high and then pulsing it low for a short period of time to produce the click, a negative click will be generated at the driver output.

The output drivers are based around the TL082[i] dual op-amp which is powered from the same +/- 5V power supply as that for the EEG pre-amplifier. The 3.3V maximum amplitude of the ERClick signal is amplified by a factor of 1.7 to produce the pulse that is used to drive the speaker.

Overall, the stimulus generators have a bandwidth from 4.1Hz to 100kHz while the amplitude of the output click can be found from:

$$A = (3.3\text{V})(1.7)\frac{P}{256} \tag{A1.7}$$

where A is the output amplitude of the output click and P is the wiper position value loaded into channel 1 or 3 of the digital potentiometer. Note that for high values of P the output click amplitude will be limited by the power supply voltage (5V). The effects of the digital potentiometer wiper resistance can be ignored here since they are small (50Ω) when compared to the input impedance of the output driver circuitry (about 39kΩ). Note also that even if P is zero an audible click may still be heard due to capacitive coupling through the digital potentiometer. The best way to eliminate this click is to design any evoked response acquisition software so that when the desired amplitude is zero the ERClick signal is disabled.

A2.6.3 Temperature Measurement Inputs

The two available temperature measurement channels can use either precision thermistors or digital temperature sensors such as the DS1820[ii]. The calibration of the thermistors typically gives a reading accurate to within 0.1°C while the digital sensors are typically accurate to <0.05°C. At present only thermistors have been used with the FRPM.

If thermistors are to be used then jumpers R48 and R50 should be present while jumpers R47 and R49 should be removed. The thermistors are provided with a bias voltage of twice the V_{REF} voltage to ensure that the temperature measurement process remains ratiometric with respect to any variations in the MAX147 ADC reference voltage. This bias voltage is amplified by the MAX4333 Dual Op-amp (U8) which also allows its output drivers to be disabled via the TempEnable signal to reduce power drain when temperature measurements are not required. The 10ms time constant of the bias generators is intended to reduce bias voltage variation and noise and should be accounted for when temperature measurements are made. At least an 80ms delay should occur between the bias generators being enabled and the measurement sample being taken to enable the bias voltage to stabilize.

The actual temperature is measured via the resistor divider consisting of the temperature sensing thermistor connected between TempBias and TempIn and the respective load resistor, either R41 or R42. The capacitors C42 and C43 limit the measurement bandwidth to a nominal 36Hz.

---

[i] TL082 Dual Operational Amplifier, Texas Instruments Inc., Dallas, TX75380-9066, USA.
[ii] DS1820 Digital Thermometer, Dallas Semiconductor Corp., 4401 South Beltwood Parkway, Dallas, TX 75244, USA.

The resulting voltage across the load resistor is measured by either channel 2 or channel 3 of the MAX147 ADC on the FRPM Main Board. The formula for converting the resulting ADC counts to the thermistor temperature is:

$$R_t = \frac{(2)(3300\text{k}\Omega)(4096)}{N} - 3300\text{k}\Omega$$
$$T = \frac{\beta}{\ln(R_t.K)} - 273.0 \qquad \text{(A1.8)}$$
$$\text{Where: } K = \frac{1}{R_0}\exp(\frac{\beta}{T_0})$$

Where N is the number of ADC counts and T is the actual temperature in °C. The variables β, R_0, and T_0 are the three characteristic parameters of the thermistor.

If digital temperature sensors are to be used then jumpers R47 and R49 should be present while jumpers R48 and R50 should be removed. Resistors R21 and R23 should also be changed to 330kΩ so that the bias voltage becomes the lesser of V_{IN} or 5.2V to provide the 5±0.5V requirements of the DS1820 sensors. In case V_{IN} is particularly high, for instance when the FRPM is running from an external supply, R53 and D1 are provided to ensure that the MAX4333 maximum V_{CC} rating of 6.5V is not exceeded. Resistors R51 and R52 are pull-up resistors for the digital sensor data bus.

A2.6.4 Audio Recording Inputs

The two variable gain auditory input channels provide support for the recording of respiration (via microphone, thermistor or chest expansion), physical activity (via accelerometer) and digestive activity or teeth grinding (via microphone) by the FRPM. In a more general sense, the auditory input channels provide two variable gain, AC coupled inputs for the recording of signals where the time varying characteristics of the signal are more important than the absolute voltage level.

The configuration of the auditory input channels allows for the interfacing of a wide variety of sensor types. Pull-up resistors R2 and R3 allow devices that require a bias voltage, such as electret microphones, to be interfaced while the pull-down resistors R39 and R40 allow devices that require a load, such as thermistors, to be interfaced. Capacitors C15 and C16 are intended for input signal filtering in conjunction with either the pull-up or pull-down resistors.

The AD8403 digital potentiometer provides variable attenuation for the two channels with attenuation being related to wiper position. The output of the attenuator is then AC coupled before being amplified with a fixed gain of 26dB by the MAX4333 (U6) non-inverting amplifier. The DC bias network comprising resistors R32 and R43 restores sufficient offset voltage to cause the output of the amplifier to be centered at half the MAX147 ADC input voltage range. Resistors R27 and R28 provide sufficient isolation between the two input channels to prevent a signal in one channel modulating the signal in the other channel via the DC bias network. The overall signal path gain is:

$$A(dB) = 26dB + 20\log\left(\frac{P}{256}\right) \tag{A1.9}$$

Where A is the gain in dB and P is the wiper position value loaded into the appropriate channel of the digital potentiometer.

The audio input channel frequency response is from at most 0.4Hz to 2.3kHz, the lower corner depending on the wiper position of the attenuator and the input component values.

The output of the two audio input channels is recorded via channels 0 and 1 of the MAX147 ADC on the FRPM Main Board. Power saving is accomplished by disabling the output drivers of the MAX4333 amplifiers via the AudioInEnable signal and placing the AD8403 digital potentiometer in standby via the nPotSHDN signal. This signal has the effect of setting the wiper position to ground. Power management should be separately implemented for any sensor attached to an auditory input channel.

Appendix 3 FRPM System Software Design

The flexibility required of the FRPM dictated that the System Software should be modular in design and that software present on the FRPM could be tailored at any time to a particular application. Reliability also had to be paramount due to the FRPM being required to operate remotely and independently for long periods of time. Based on these requirements an operating system/application program architecture was chosen for the FRPM System Software. Application programs are run from a command line interface, on demand, to perform some task. The System Software provides the application with access to resources such as a filing system, robust communications, power management, event handling, and debugging and system protection while the application performs the functions specific to the task at hand.

Applications that have been developed for the FRPM to date include an Evoked Response acquisition and signal processing application (ERAQU), a general purpose waveform capture application (SAMPLER), an ECG data compression application (COMPRESSECG), a general data logging application (MEASURE), and an application for file compression (CONTAINER). Information on these applications can be found in the FRPM Users Manual[138].

As an alternative to the command line interface, the System Software provides a scheduler based on the system real-time clock that enables applications to be run at particular times on either a fixed time or a periodic basis. Using this feature the researcher can automate and synchronize data collection from multiple FRPM units simultaneously.

Figure A3.1 gives an architectural overview of the System Software.

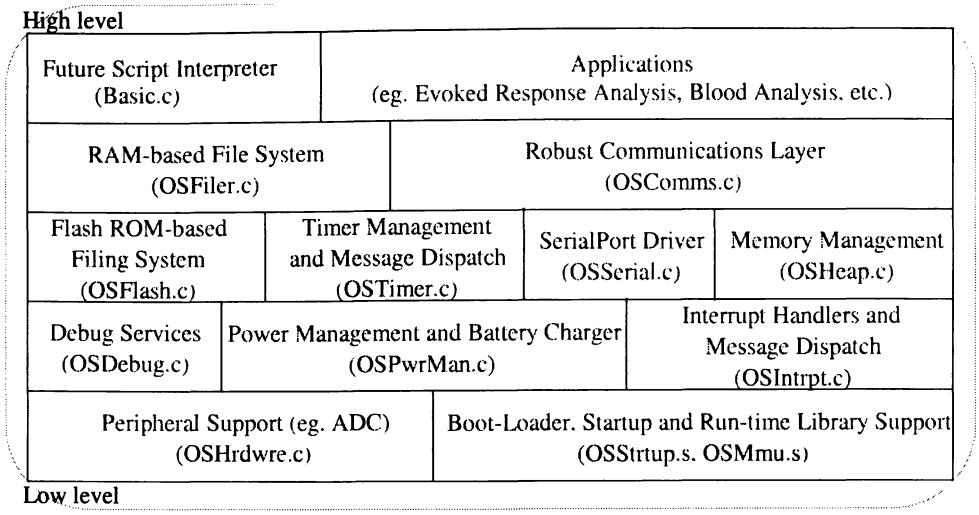


Figure A3.1 – Overview of the FRPM System Software

The System Software is a full 32-bit system and takes advantage of the hardware memory protection and exception traps provided by the ARM710a processor to terminate any application that attempts an illegal memory access or attempts to execute an illegal instruction. Any resources that were allocated to an application that was terminated are always recovered, ensuring reliability by eliminating resource and memory leaks.

The System Software occupies approximately 100kB of memory.

A3.1 Development Environment

The FRPM software was developed using the ARM Software Development Toolkit[i]. This toolkit comprises a command line-based compiler, assembler, and linker together with a Microsoft Windows®-based project manager and debugger. The project manager was not used in favor of UNIX-style makefiles. Build variant control for generating different executable images (for example the release executable, various debug executables and extra code required for the debugging environment) was implemented using conditional directives in the makefile.

The debugger included in the toolkit is cycle-accurate and supports both simulation of the target system as well as allowing single-step execution on actual target hardware via either a JTAG test port or via a software-based monitor running on the target hardware. Using the debugger with the target hardware was rejected due to the lack of a JTAG port on the CL-PS7110 and known problems with porting the ARM-supplied monitor to new hardware platforms.

The debugger as supplied simulates only the processor core (CPU, cache memory, and Memory Management Unit) itself with little, or no, simulation of the real external memory array or any

[i] ARM Software Development Toolkit, Advanced RISC Machines Ltd., Fulbourn Road, Cherry Hinton, Cambridge CB1 4JN, UK.

peripherals. However, it does provide for expansion via the Microsoft Windows Dynamic Link Library (DLL) mechanism to allow users to add such models. Software development for the FRPM included writing an additional DLL module to allow the debugger to simulate the FRPM hardware, including both peripherals within the CL-PS7110 and the external memory map. The module included simulations of both DRAM and Flash memory together with memory protection (the memory protection model included in the basic simulator was flawed). It also simulated the CL-PS7110 interrupt, timer, real-time clock, and power management functions. Serial communications were simulated by redirecting serial port activity to and from files on the host computer.

Using this module, all of the basic FRPM System Software including the boot-loader, startup code, file system, event mechanisms and application support were tested before being used on the actual hardware. Not all peripherals were simulated, notable exceptions being the analog-to-digital converter, LCD display, and radio link, however sufficient system resources were available to develop those areas on the actual hardware without the aid of external support.

A3.2 System Memory Map

The ARM710a microprocessor core allows for the direct addressing of 4GB of memory space of which only 18MB is utilized in the FRPM. Table A3.1 shows the FRPM Memory Map as seen by the System Software. Any areas not included in this Memory Map are unused in the FRPM.

The FRPM System Software is responsible for the management of the 16MB of main memory and 2MB of Flash memory. The main memory holds all executing code with separate areas for the System Software, an optional interpreter (not currently implemented), and an executing application. A 10MB section is reserved for the Main Filing System (see Section A3.11.1) that provides for the storage of both data and application executable images. The Flash memory holds a copy of the System Software executable image to allow the FRPM to start from a system reset, and a copy of the interpreter executable image. 1792kB of the Flash memory is reserved for the Flash Filing System (see Section A3.11.2) that provides for the non-volatile storage of commonly-used applications or critical data that must be preserved through any possible power failure.

The System Software has 960kB of available space of which approximately 100kB is currently taken by the actual System Software code leaving around 860kB for the System Software heap. The interpreter has a 1MB area reserved for its eventual implementation. It is intended that the interpreter code occupy the low end of this area, followed immediately by the interpreter heap. The interpreter stack should grow downwards from the top of this area to give the heap the maximum available memory.

The application space has 4MB of main memory reserved for the execution of a single application. This allows the application to utilize a large heap for such memory-intensive algorithms as EEG state analysis or the compression of large data records such as an ECG recording. Currently none of the provided applications have code sizes larger than 50kB. The entry point for the application is the application base address 0xC0C00000. The application heap begins at the end of the application image while the application stack grows downwards from 0xC1000000.

Also present in the FRPM Memory Map are the areas reserved for access to the CL-PS7110 Internal Registers and any expansion devices attached to the FRPM Memory Expansion Connector. The Internal Registers occupy 4kB in the memory map while the four possible expansion devices each occupy 1kB. The Memory Management Unit (MMU) Translation Table occupies 16kB of Main Memory while the stacks for the various ARM710a modes occupy a total of 9.5kB. The display memory and character map for the LCD display together occupy 38.5kB.

Start Address	Length	Access	Description
0xC1000000			Top of Main Memory
0xC0C00000	4MB	System RW/User RW	Application Space
0xC0200000	10MB	Locked	Main File System
0xC0100000	1MB	System RW/User RW	Interpreter Space
0xC0010000	960kB	System RW/User RO	System Software Space
0xC000C000	16kB	System RW/User RO	MMU Translation Table
0xC000BF00	256B	System RW/User RO	ABORT Mode Stack
0xC000BE00	256B	System RW/User RO	UNDEF Mode Stack
0xC000BC00	512B	System RW/User RO	FIQ Mode Stack
0xC000BA00	512B	System RW/User RO	IRQ Mode Stack
0xC0009A00	8kB	System RW/User RO	SVC Mode Stack
0xC0009600	1kB	System RW/User RO	LCD Display Character Map
0xC0000000	37.5kB	System RW/User RO	LCD Display Memory
0xC0000000			Start of Main Memory
0x80000000	4kB	Any RW	CL-PS7110 Internal Registers
0x70000000	1kB	Currently No Access	Expansion Device #4 (CS7)
0x30000000	1kB	Currently No Access	Expansion Device #3 (nCS3)
0x20000000	1kB	Currently No Access	Expansion Device #2 (nCS2)
0x10000000	1kB	Currently No Access	Expansion Device #1 (nCS1)
0x00200000			Top of Flash Memory
0x00040000	1792kB	Locked	Flash Filing System
0x00020000	128kB	Locked	Interpreter Image
0x00000000	128kB	Locked	System Software Image
0x00000000	20B	Locked	ARM Exception Vectors
0x00000000			Start of Flash Memory

Table A3.1 – FRPM System Software Memory Map

The FRPM System Software uses the ARM710a Memory Management Unit (MMU) to provide protection for the System Software, Flash memory, and file system areas from erroneous access by any executing application or possibly by the System Software itself. Table A3.1 also indicates the access rights for each area: System access implies any code executing in a privileged mode

(all ARM710a modes except USER32) while User implies any code executing in the USER32 mode. RW is permanent read and write access for that area, RO is read-only access for that area, while Locked implies that read access is permanently available but the area must be unlocked via the MMU by trusted sections of System Software code for write access.

A3.3 Overview of the ARM Processor

The CL-PS7110 microprocessor used in the FRPM is based around the ARM710a, a general purpose 32-bit microprocessor core combined with an 8kB cache, write buffer and Memory Management Unit (MMU). This core is a Reduced Instruction Set Computer (RISC) design and incorporates all the features typical of a typical RISC architecture including a large uniform register file, a load-store architecture (data processing operations only operate on register contents), simple addressing modes, conditional execution of all instructions, and uniform and fixed length instructions. In addition the ARM architecture provides features more typical of DSP processors such as control over the shifter on every arithmetic operation, and auto-increment and auto-decrement addressing modes.

These features enable the ARM architecture to produce processors that occupy minimal silicon area while providing high performance for low power consumption.

The on-chip mixed data and instruction cache together with the write buffer substantially raise the average execution speed and reduce the average amount of memory bandwidth required by the processor. The MMU supports a conventional two-level page-table structure.

A3.3.1 The ARM710a Microprocessor Core

The ARM710a core has thirty-one, 32-bit registers. At any one time 16 of these are visible while the others are used to speed up exception processing. Any of the 16 visible registers can be used as the source or destination for any of the ARM register-orientated instructions, i.e. any register can act as the accumulator. Figure A3.2 shows the ARM processor registers. The shaded registers are those that are banked, i.e. are visible only within their particular mode.

In the FRPM, the System Software runs in the SVC32 mode, switching to any of the four exception modes (IRQ32, FIQ32, ABORT32 and UNDEF32) as the need arises. The Interpreter and applications always execute in the USER32 mode, switching to the SVC32 mode via the Software Interrupt (SWI) mechanism to call any privileged System Software services.

R15 is the Program Counter (PC) and can be used in most instructions as a pointer to the instruction which is two instructions after the instruction currently being executed (the delay being due to the instruction pipeline). Register `R14` is the Link Register (LR) and is used to hold the address of the next instruction after a Branch-with-Link (BL) instruction which is the instruction used to make a subroutine call. Moving the contents of LR to the PC performs a

return from a subroutine to the caller. If a subroutine wishes to call another subroutine it must save the contents of LR before the call and restore it after the subroutine returns otherwise the contents of LR will be lost. `R13`, the Stack Pointer (SP), functions as a stack pointer for the current stack frame while `R11`, the Frame Pointer (FP), points to the base of the current stack frame. All other registers are usually available for general use however the guidelines of the ARM Procedure Call Standard (APCS-3) should be followed.

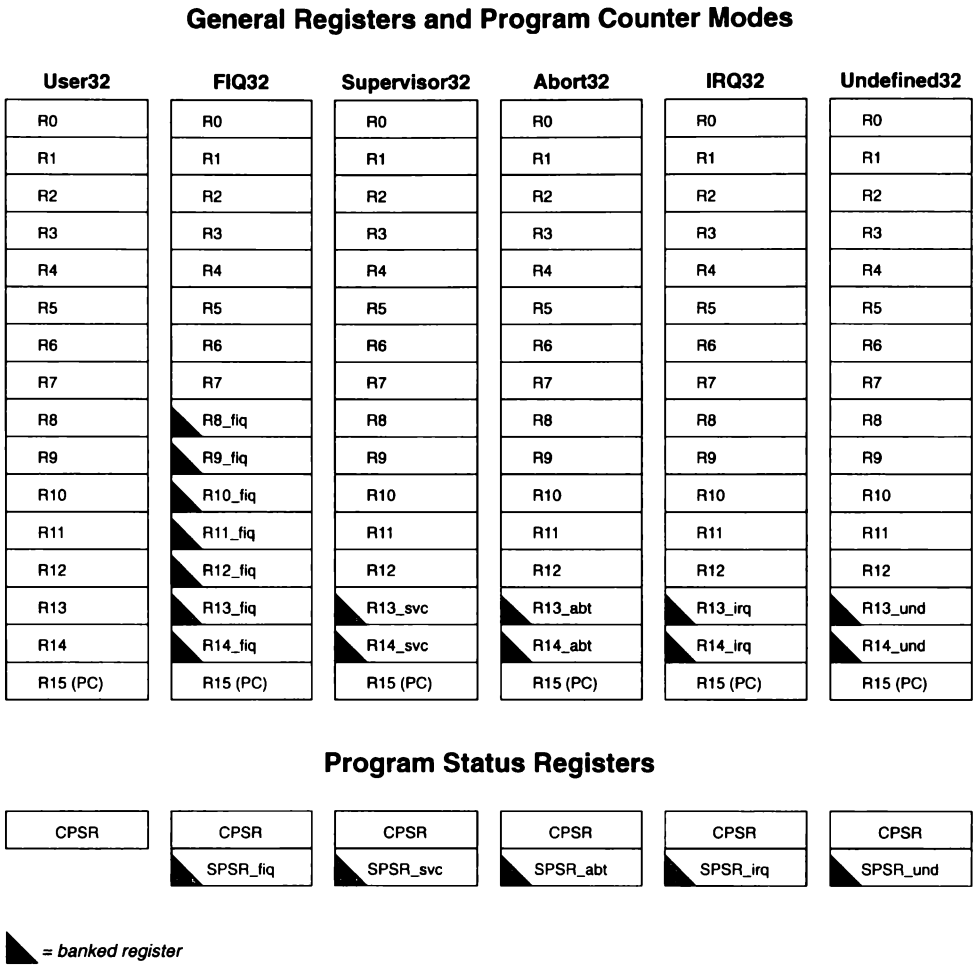


Figure A3.2 – ARM Register Organization

The CPU state for the current mode is held in the Current Program Status Register (CPSR). All five exception modes also have a Saved Program Status Register (SPSR) which holds the value of the CPSR immediately before the exception occurred.

The ARM710a supports eleven classes of instruction covering 39 distinct instructions. However because the load/store operations allow several different addressing modes and many instructions allow shifting of operands, the number of instruction variants is 224. Table A3.2 shows the various instruction classes.

Instruction Class	Number of Instructions/Variants
Data Processing	16/64
Multiply	4/4
Branch	2/2
Load/Store Single	4/64
Load/Store Multiple	2/32
Swap (semaphore)	2/2
Co-processor	5/56
Software Interrupt	$1/2^{24}$ possible
Undefined	$1/2^{24}$ possible

Table A3.2 – ARM710a Instruction Classes

Two mechanisms are provided for extending the instruction set. The Software Interrupt (SWI) allows the coding of up to 2^{24} additional instructions that when executed will cause a switch from any mode to SVC32 mode where the instruction can be emulated in software. This mechanism is most often used to implement calls from the under-privileged USER32 mode to privileged, and protected, operating system code operating in SVC32 mode. The other mechanism is the Undefined Instruction (UNDEF) that allows another 2^{24} possible instructions to be emulated by software. This mechanism is often used to provide software emulation of hardware such as a floating-point unit or DSP engine during code development. The Software Interrupt mechanism is used in the FRPM System Software to allow operating system calls from applications running in USER32 mode while maintaining protection around the System Software. The Undefined Instruction mechanism is not used and the execution of any undefined instructions will cause the task that issued the instruction to be terminated.

The ARM architecture also allows for up to 16 co-processors to be attached directly to the processor core that can intercept and execute instructions present in the instruction stream without any processor intervention. In the ARM710a variant present in the CL-PS7110 the only co-processor present is #15 which provides a means to communicate with the MMU.

A3.3.2 Exceptions

Exceptions are generated by both internal and external sources to cause the processor to stop executing its current instruction stream and instead process the event that issued the exception. The processor state before the exception occurred must be preserved during exception processing and restored to allow the processor to continue from where it was interrupted. More than one exception may occur at any one time.

The ARM architecture supports 7 types of exception and has a privileged processor mode for each. Table A3.3 lists the various types of exceptions, which mode they cause the processor to switch to, the exception vector address the processor will jump to, and the priority level. Note that although the Undefined Instruction and Software Interrupt exception priorities are identical, no race condition exists since both result from mutually exclusive decoding of the current instruction.

Exception Type	Mode	Vector Address	Priority
Reset	SVC32	0x00000000	1 (Highest)
Undefined Instruction	UNDEF32	0x00000004	6 (Lowest)
Software Interrupt	SVC32	0x00000008	6 (Lowest)
Prefetch Abort	ABORT32	0x0000000C	5
Data Abort	ABORT32	0x00000010	2
Interrupt Request	IRQ32	0x00000018	4
Fast Interrupt	FIQ32	0x0000001C	3

Table A3.3 – ARM710a Exception Processing Modes

When an exception is taken the current PC is saved to the matching exception mode LR and the current CPSR is saved to the matching exception mode SPSR. The processor mode is switched to the required exception mode. Interrupt processing is disabled and Fast Interrupt processing is disabled if the exception was a Reset or Fast Interrupt. The PC is then loaded with the exception vector address and exception handling begins. Exception handling is finished by restoring the original CPSR from the SPSR and the original PC from the LR.

In the FRPM the most heavily used exceptions are the Software Interrupt and the Interrupt Request. The Software Interrupt is used by any executing application to call any privileged SVC32 mode System Software services it requires from its own under-privileged USER32 mode. This is the only possible way to access SVC32 code from the USER32 mode without causing an MMU Permission Fault (see Section A3.4).

The Interrupt Request is used by nearly all of the interrupt-generating hardware within the CL-PS7110, including the serial port and hardware timers. The Interrupt Request exception handler IRQHandler (module OSIntrpt) must prioritize and dispatch any interrupt sources to their appropriate handler. The dispatching of interrupts is discussed in Section A3.8.1. The Fast Interrupt is used by all other interrupt-generating hardware and uses the FIQHandler (module OSIntrpt) handler routine. This exception is designed to support data transfer or channel-type processes, and has sufficient private registers to remove the need for register saving in such applications. Unfortunately this capability is under-utilized in the CL-PS7110.

The Undefined Instruction exception is trapped by the FRPM System Software and will result in the assembler-coded _UndefHandler (module OSStrtup) being called which saves the state of the processor at the instant the exception happened. This routine in turn calls the C-coded UndefHandler (OSSystem) which looks at the saved processor state to determine where the faulting access originated from. If the offending instruction originated from an application that application is terminated, all System Software resources currently in use by the application are freed (files are closed, communications buffers are flushed, etc.) and control is returned to the System Software. If the System Software itself was the source of the access the current behavior is to perform a Software Restart to reinitialize the System Software.

In the current implementation, the one-to-one virtual to physical memory mapping of the MMU causes the exception vectors to be obtained from the relatively slow Flash memory. However, the Interrupt Request and Software Interrupt exceptions are used often enough in practice that their respective vectors are almost always present in the processor cache, eliminating the need for an access to Flash memory. Because a cache line consists of 4 consecutive words the use of these two vectors will also bring all the other vectors into the cache.

A3.4 The Memory Management Unit and System Protection

The ARM processor Memory Management Unit (MMU) performs two primary functions: it translates virtual addresses into physical addresses via the Translation Table, and controls memory access permissions via Domains. The hardware required to perform these functions consists of a Translation Look-aside Buffer (TLB), access control logic, and Translation Table access logic.

The Translation Table is an area in main memory that contains an array of pointers that map the virtual addresses used by the processor to the blocks of physical addresses available in system hardware. The entries in the table also contain access permission control data. The ARM710a supports the use of both Sections (1MB blocks of memory) and Pages (Large Pages are 64kB blocks, while Small Pages are 4kB blocks) to describe physical memory blocks. The Translation Table should not contain duplicate entries as this can result in cache coherency and access permission errors. The Translation Table must reside on a 16kB boundary.

The MMU also manages the ARM710a cache and write buffer. The 8kB cache attempts to store commonly used code and data for fast access without the need for a main memory fetch and performs predictive fetches in the hope of having code or data in the cache before the CPU requires it. The write buffer buffers and attempts to re-order main memory write accesses so write accesses can occur in the background and in the optimum sequence while the CPU continues at full speed.

In the case of the FRPM the Translation Table occupies 16kB of memory in DRAM starting at `0xC000C000`, this address being known as the Translation Table Base. The mapping chosen is one-to-one to ease MMU management, hence the entire 4GB virtual memory space in the FRPM can be described by 4096 4-byte Section descriptors. When the MMU is disabled the virtual address is output directly as the physical address and no translation occurs. However, if the MMU is disabled the cache and write buffer are also unavailable and processor performance suffers greatly.

The Translation Look-aside Buffer caches up to 64 Translation Table entries so that usually the required entry to map a virtual to physical address can be found without accessing main memory.

The ARM710a MMU supports the use of up to 16 separate Domains. A Domain is a collection of areas of memory that can be defined to possess combined access rights so that the access permissions for large areas of memory can be changed quickly without modifying several Translation Table entries.

The process of translating a virtual address into a physical address involves reading a descriptor from the Translation Table and using it to modify the virtual address. Because Page descriptors (second-level descriptors) are not used in the FRPM, only the process of using Section descriptors will be discussed. A Section descriptor can be described as in Listing A3.1.

```
typedef struct {
      unsigned Identifier : 2;    // Should be 0x02 (lsb)
      unsigned IsBuffereable : 1; // TRUE if write buffer is
                                  // enabled for this section
      unsigned IsCacheable : 1;   // TRUE if cache buffer is
                                  // enabled for this section
      unsigned _Padding1 : 1;     // Should be one
      unsigned Domain : 4;        // This sections domain
      unsigned _Padding2 : 1;     // Should be zero
      unsigned Access : 2;        // Access permissions
      unsigned _Padding3 : 8;     // Should be zero
      unsigned Base : 12;         // Base address of this
                                  // section in memory (msb)
      } SectionDescriptorType;    // Describes a section
```

Listing A3.1 – Section Descriptor Format

If the Identifier field is set to `0x02` it identifies this entry as being a Section Descriptor. An Identifier of `0x01` indicates the entry refers to another (2nd level) translation table for control Page-level access. All other values are illegal. The IsCacheable and IsBufferable flags indicate whether the contents of this Section are allowed to be stored in the cache and write buffer respectively. Sections that contain input/output ports or memory mapped peripherals should be marked as neither cacheable nor bufferable since doing so could lead to out-of-order read or write accesses resulting is data corruption or loss. The Base field holds the 12 most-significant-bits (MSBs) of the Section's address in physical memory. The Domain and Access fields allow access permission checking and are discussed later.

The translation of a virtual address from the processor to a physical address requires the steps outline in Figure A3.3. The 12 MSBs of the virtual address are appended to the 18 MSBs of the Translation Table Base Address to obtain the address of the required Descriptor in memory. The two LSBs of the resulting address are always set to zero since the Descriptor always resides on a word boundary. The Descriptor is then fetched from memory (or from the TLB if it is present there) and its 12 MSBs are joined with the virtual addresses' 20 LSBs to form the full 32-bit physical address.

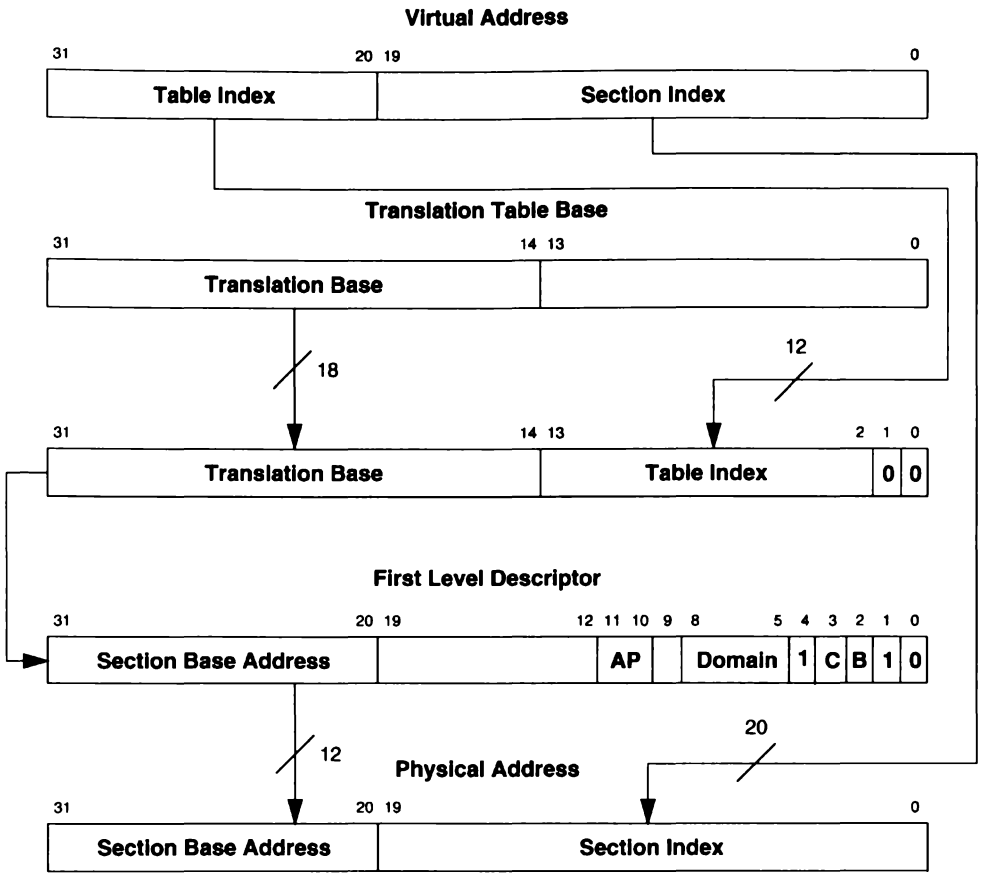


Figure A3.3 – Virtual to Physical Address Section Translation

The access permission bits in the Section descriptor control access to the corresponding section. Table A3.4 describes the meaning of the access permission bits in the context of the FRPM System Software. If an access is made to an area of memory without the required permission, a Permission Fault is generated which results in either a Prefetch or Data Abort exception depending on whether the access was for code or data.

Access	Privileged Mode Permissions	User Mode Permissions
00	No Access	No Access
01	Read/Write	No Access
10	Read/Write	Read Only
11	Read/Write	Read/Write

Table A3.4 – Access Permissions

Access to each of the 16 Domains is controlled by a 2-bit field in the Domain Access Control Register. Each field allows access to an entire domain to be enabled and disabled very quickly. Two kinds of domains are supported: Client domains are guarded by the access permissions of the individual sections that make up the domain, Manager domains control the behavior of the domain and are not guarded by the access permissions i.e. no access checking is performed on

manager domains. The decoding of the fields within the Domain Access Control Register are shown in Table A3.5. If a memory access does not have the required permission to access a particular domain, a Domain Fault will be generated when the access occurs.

Value	Domain Type	Description
00	No Access	Any access will generate a domain fault
01	Client	Accesses are checked against the access permission field in the section descriptor. Illegal accesses will result in a permission fault.
10	Reserved	Illegal value
11	Manager	Accesses are not checked against the access permission field in the section descriptor hence a permission fault cannot be generated

Table A3.5 – Domain Access Types

In the FRPM System Software six domains are used to protect areas of memory as shown in Table A3.6. Areas labeled as Locked RO mean that normally the area is read-only access in any mode but trusted portions of System Software code may unlock the area via the Domain Access Control Register and perform any desired read/write operation, before re-locking the area once the operation is finished. In this way the Flash memory and Main File System areas are protected against stray writes by misbehaving code executing in any mode. The System Software area is only protected against stray writes originating from User mode code while the Application and CL-PS7110 Internal Register areas have no protection. Trusted code (which must be running in a privileged execution mode) unlocks and locks desired areas by modifying the contents of the Domain Access Control Register.

Domain	Function	Domain Size	Access
0	Expansion Areas	1792MB	Any RO
1	Register Access Area	1MB	Any RW
2	Flash Memory Area	2MB	Locked RO
3	System Software Area	2MB	Supervisor RW/User RO
4	Main File System Area	10MB	Locked RO
5	Application Area	4MB	Any RW

Table A3.6 – Domain Allocation in the FRPM System Software

In the ARM710a MMU there are nine potential conditions that would cause a memory access can fail. In the CL-PS7110 implementation, and in particular the FRPM, only four of these conditions are possible. These are the Alignment Fault, Section Translation Fault, Section Domain Fault, and Section Permission Fault conditions. Figure A3.4 shows the sequence of fault checking in the MMU hardware.

If a fault was caused by a data load/store operation a Data Abort exception is generated. Faults arising from instruction fetches are simply flagged as the instruction enters the instruction

pipeline. Only when (and if) the instruction is executed will it cause a Prefetch Abort exception i.e. if the instruction is not used (e.g. it is branched around) the fault, and resulting exception, will not occur. The MMU hardware protects the system memory by stopping any access that would cause a fault to occur before any memory access can take place and can be interrogated to determine the exact cause of the fault.

An Alignment Fault indicates that the access is not aligned on a word boundary and would result in unpredictable data being read from memory. A Section Translation Fault suggests that the MMU Translation Table has been corrupted. A Domain Fault indicates the access is to an domain designated in the Domain Access Control Register as No Access. A Permission Fault indicates that the access does not have sufficient permission to perform the access.

In larger systems the Domain and Permission Faults would provide hooks to implement such schemes as virtual memory. However, in the relatively simple FRPM environment all faults can only be the cause of an illegal memory access by some offending piece of code. As a result, in the FRPM protection scheme, all of these faults result in the assembler-coded AbortHandler (module OSStrtup) being called which saves the state of the processor at the instant the abort happened. This routine in turn calls the C-coded IllegalAccessHandler (OSSystem) which looks at the saved processor state to determine where the faulting access originated from.

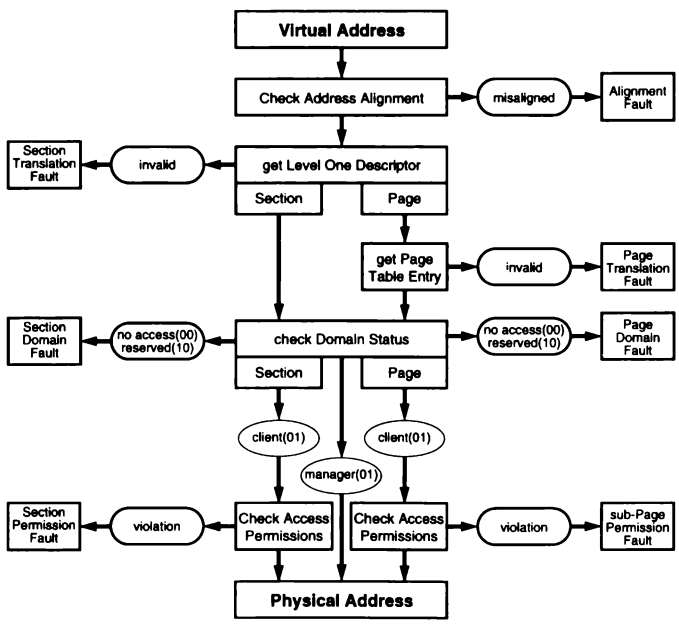


Figure A3.4 – MMU Fault Checking Sequence

If the access originated from a misbehaving application the application is terminated, all System Software resources currently in use by the application are freed (files are closed, communications buffers are flushed, etc.) and control is returned to the System Software. If the System Software itself was the source of the access the current behavior is to perform a Software Restart to reinitialize the System Software. In principle this behavior could be changed to only restart the

offending System Software module to increase system reliability, however the current approach is the surest and the safest.

In APCS-3 register `R10` (SL) is defined to contain the limit of the current stack frame to allow software stack checking upon entry to any procedure. If SP moves below SL a stack overflow error has occurred and corrective measures should be taken (the current task terminated or the stack enlarged) to protect system integrity. The FRPM System Software currently does not support stack checking because the ARM embedded C library does not support it (code compiled with stack checking enabled cannot be used with code with stack checking disabled). However in the FRPM, stack overflow should quickly manifest itself in the task responsible as either an illegal memory access or an attempted illegal instruction, in which case an exception would result and corrective measures could be taken. As an additional safety measure the various FRPM stacks are considerably larger than required, for example the SVC32 mode stack is 8kB whereas 2kB would be sufficient.

A3.5 System Initialization

The OSStrtup module manages system initialization after a hardware reset or a software restart. This module is written in ARM assembly language and performs all operations required to create an environment in the FRPM hardware for executing the System Software C code.

On the ARM710a a reset causes the MMU control register to place the CPU in little-endian mode (least significant byte within a word occupies the lowest byte address) with a 26-bit data and program address space with the write buffer off, cache off, address alignment fault off (memory accesses need not occur on word boundaries), and MMU off. The CPU is placed in the privileged supervisor mode with interrupts disabled.

OSStrtup first switches the MMU to enable a full 32-bit code and data address space. It then initializes the CL-PS7110 UART to allow diagnostic information to be sent from the serial port during initialization. All input/outputs are set to their appropriate directions and values and the CL-PS7110 System Control (SYSCON), Memory Configuration (MEMCFG1 and MEMCFG2), and DRAM Refresh Control registers are also set to their default values.

The next step is to ensure the System Software will execute from the base of DRAM memory, `0xC0000000`. If the system was started directly from Flash memory execution the program counter `R15` (PC) will currently be below `0x00020000` whereas if the system was started via the Boot Loader program the image will already be located at the base of DRAM memory and `R15` will be above `0xC0000000`. Hence the value of `R15` is used to decide if the image should be copied from Flash memory to DRAM or whether is step can be skipped because the image is already in DRAM memory.

The MMU is then initialized by creating the Translation Table at address 0xC000C000 to perform a one-to-one map of virtual memory addresses to physical memory addresses. Of the sixteen available domains six are used within the FRPM. These provide protection to the Flash memory area, Main File System memory area, System Software memory area, and Application memory area with a further two domains used to combine the scattered open- and no-access memory areas into logical blocks. Finally the MMU is enabled with both the cache and write buffer enabled and the instruction pipeline is flushed by a branch instruction to ensure that all further instructions are fetched from their translated address via the MMU Translation Table.

Once the MMU is enabled the System Software's static and zero-initialized data areas can be initialized and any System Software variables that depend on the startup state are also initialized. The five system-related stacks, SVC, IRQ, FIQ, UNDEF, and ABORT are initialized to their appropriate areas and an ARM Procedure Calling Standard (APCS-3) stack frame is set up for the SVC mode. Finally, the startup code branches to the initial C procedure FRPMOS_Start (OSMain) which initializes and starts the System Software. Control is never returned to the OSStrtup code.

A3.6 Application Support

In addition to the System Software thread, the FRPM System Software provides support to execute a single application thread. This application support is intended to allow separate code development for different functions without the need to directly link any new functions to the System Software itself. For instance, separate applications have been written to record and analyze Evoked Response waveforms, record ECG signals, and log temperature profiles.

Support could have been provided for multiple application threads however there are few applications of the FRPM that would required such multi-tasking capability. For most applications the combination of the Command Sequencer and a single application thread is sufficient.

An application takes the form of an executable image of the application stored as a file in the filing system. To execute the application the System Software copies the image from the protected memory area of the filing system to the application space. It then saves its own state before initializing an environment suitable for the application to execute in. It then branches to the application's entry point. During application execution the System Software provides its services to the application via the SWI mechanism and enforces system protection, terminating the application if it transgresses. When the application terminates, either on its own or by force, the System Software ensures that any resources used by the application are released and restored to their original state. The System Software then restores its own state and continues from where it left off.

Because applications are stored in the filing system several applications can be present on the FRPM at any time, either in volatile DRAM memory or protected in Flash memory. Treating applications as files also allows applications to be transferred to the FRPM and manipulated as any other file without any special consideration.

A3.6.1 Execution and Termination

In the FRPM applications are held as an executable image in ARM Image Format (AIF) in the filing system, allowing multiple applications to be resident on the FRPM at any time. Although the AIF format includes support for relocation of the image upon execution, the images are linked with an entry point of `0xC0C00000`, the base of the 4MB area of memory reserved as application space.

To execute the application the System Software calls SystemExecute (module OSSystem) with the filename of the application and the command line to pass to it upon entry. SystemExecute first opens the application file and copies the image to the base of the application space. It then resets the application-related system protection mechanisms, including the Application Watchdog Timer, and calls _SystemExecute (OSSupprt) to perform the execution context switch which switches the current execution thread from the System Software to the application.

_SystemExecute first saves the current state of the System Software to the SVC32 stack, switches the processor to the USER32 mode to provide the application thread with its own execution environment, creates an initial stack frame for the application, and then branches to the application entry point. Because the application will manipulate the SVC32 stack via calls to the various System Software services, the SVC32 stack pointer is also saved separately before the mode change to ensure it can be correctly restored.

The initial entry point into the application is through AppStartup (module _AppStart) which initializes the application's static and zero-initialized data areas and constructs the necessary C runtime environment for the application before branching to AppEntry (module AppStart). AppEntry checks the application was compiled for the correct Software Interrupt table version and initializes the application heap before converting the command line used to run the application into the argc and argv variables commonly used by C applications to parse the command line for parameters. Once the _AppStart and AppStart modules have initialized and created a suitable environment for the application, the first application-specific procedure, AppMain (module depends on the application), can be entered. The application should exit by calling the SystemExit service provided by the System Software.

The System Software will forcibly terminate an executing application for causing an illegal memory access, making an illegal System Software call, executing an illegal instruction, or for not responding to the system (watchdog expired). Any of these conditions cause the application exit code be called immediately to terminate the application.

To terminate an application SystemExit (OSSystem) is called to restore any system resources the application used to their original state and to retrieve the exit code returned from the application. It then calls _SystemExit (OSSupprt) to perform the context switch to restore the saved System Software execution thread. At this point the exception stacks are also reset to their start-up condition in case the application terminated abnormally. Because of the context switch, when _SystemExit returns it returns directly to the SystemExecute procedure which resets the application-related protection mechanisms and in turn returns control to it caller, completing the process of executing an application from the System Software.

A3.6.2 System Software Interface

Applications can call any of the public System Software functions via the Software Interrupt mechanism. Currently 135 functions are provided for application use. The Software Interrupt mechanism provides the only method for an application to call the protected System Software code without generating an illegal memory access.

To call a System Software function the application executes a Software Interrupt instruction, `swi n`, where `n` is the identifying number of the function and registers `R0-R3` hold any parameters the function requires. This instruction causes the processor to receive a Software Interrupt exception, switch to SVC32 mode, and jump to the Software Interrupt handling code _SWIHandler (module OSStartup). This handler saves registers `R4-R12` to the stack (`R0-R3` should have been saved by the caller in accordance with APCS-3) and extracts `n` from the instruction that caused the Software Interrupt. It then sets a flag indicating Software Interrupt handling is in progress, resets the application watchdog timer, and calls the main Software Interrupt handler, SWIDispatcher (module OSSwi) with the value of `n`.

After checking that `n` represents a valid function, SWIDispatcher uses `n` to perform a table lookup of the function entry address and branches to the resulting address thus calling the function. When the function returns it does so directly to _SWIHandler. SWIHandler then restores the stack, switches the processor back to USER32 mode, and returns control to the application along with the optional function return value. If `n` was found to represent an invalid function, SWIDispatcher calls the SystemExit function and the application is forcibly terminated for calling an illegal System Software function.

The APCS-3 standard used for building the System Software allows functions with up to 4 arguments and a single return value to be called with no extra overhead. If more parameters need to be sent to a function, the compiler places the parameters in a block of memory and passes a pointer to the block to the function instead.

A3.6.3 Application Watchdog Timer

The System Software may forcibly terminate an application if it is determined to be unresponsive by the Application Watchdog Timer (module OSTimer). The watchdog is currently set to have a 10 second period as determined by the 64Hz SlowTimer (see Section A3.8.2). If this watchdog is not reset by the application before it counts down to zero the current application is flagged as being unresponsive via the OSSystem module.

An application that has been flagged as unresponsive is detected by the IRQHandler (module OSIntrpt) when it next finishes interrupt processing. It then forces the application to be terminated via the SystemExit function (module OSSystem) which performs all necessary cleanup tasks to restore the system to the state it was in before the application started execution. Whenever the Watchdog reaches zero it is reloaded, hence the system will attempt to shutdown an unresponsive application every 10 seconds until it succeeds.

If the application thread is currently halted by a System Software routine (e.g. DelayOnTimer) the application will not be flagged as unresponsive. The watchdog is only enabled during application execution and is disabled again once the application terminates.

The application can reset the Watchdog Timer either indirectly by calling any System Software routine via the Software Interrupt mechanism or by calling ResetWatchdogTimer explicitly. Note that the Application Watchdog Timer can be defeated by an application that either manipulates the SlowTimer hardware directly or globally disables interrupts.

A3.7 Debugging Support

The FRPM System Software includes support for the debugging of both applications and the System Software itself. This support takes the form of functions to allow the output and storage of debug information during program execution, and tools to assist management of both release and debug versions of the same code base. Low level debugging facilities include control of the three debugging LEDs on the FRPM Main Board, direct use of the serial port without using the OSSerial module, and "Panic Block" management.

The debug LEDs and direct serial port access can be used to display status information during program startup when no communications facilities exist or when time-critical code cannot afford the time to use a communications resource. The LEDs can also be used to collect timing and performance information during program execution.

The Panic Block is used when an unforeseen circumstance forces the System Software to restart and any form of communications can no longer be treated as reliable. It takes the form of a block of memory which can be used to store a message to describe the condition that caused the restart. The Panic Block is not overwritten during the startup sequence allowing the System Software to

recover any message left there. The Panic Block is only intended to store messages relating to software failure since most hardware failures will cause the DRAM memory to lose its contents. Conditions such as power failure can be determined by looking at the System Flags (SysFlags) register.

Conditional compilation directives in the application and System Software frameworks enable any module to be compiled as either a release version with no debugging information generated during execution or as a debug version which will output both control flow, and the internal state of the module. The debug versions suffer a considerable performance penalty from communicating the debug information hence release versions of the code are used whenever possible.

A3.8 Interrupt and Timer Support

The FRPM System Software provides support for event-driven programming with four different types of events: interrupt events, timer events, real-time clock alarm events, and power management events. Power Management events are discussed in Section A3.9. An event can be used to either call a specified function or increment a counter (flag) when the event occurs.

All event management functions are re-entrant (any event can be manipulated from within any event function). Interrupt and timer events can also be blocked (disabled) and unblocked (enabled) for critical code sections. Blocking operations are cumulative - every time an event is blocked its Enabled counter decrements while unblocking operations cause this counter to increment. If Enabled is negative or zero the event is considered blocked. The Enabled counter cannot be incremented above 1 hence an event is guaranteed to always be blocked after a single Block call. The converse is not always true - a single Unblock call may not always totally unblock the event.

Any function called by an interrupt or timer event executes in either the ARM710a IRQ32 or FIQ32 modes with all interrupts disabled. Because of this event handling functions should be kept as short as possible since any event function directly affects the system interrupt handling latency.

Any events created by an application are marked as such and destroyed upon application exit by the System Software if the application has not destroyed them itself.

A3.8.1 The Interrupt Event Dispatcher

Any of the sixteen CL-PS7110 interrupt sources can be selected as an event source and used to either call an event handler function or increment a counter variable (flag) whenever that interrupt occurs. Depending on the source of the interrupt the handler function will execute in either the

IRQ32 or FIQ32 mode with all interrupts disabled. All interrupt handling processes reside in the OSIntrpt module.

There are two duplicate interrupt event handlers IRQHandler for Interrupt Request exceptions and FIQHandler for Fast Interrupt exceptions. Duplicate handlers must be provided since Fast Interrupts exceptions have a higher priority than Interrupt Request exceptions.

The interrupt event dispatcher consists of a list of virtual interrupt handlers, each handler being described by an InterruptHandle (Listing A3.2). Every time an interrupt occurs this list is searched and all matching virtual interrupt handlers are processed. Blocked (inactive) interrupts are ignored. Only a fixed number of interrupt handlers are available to ensure interrupt latencies are not excessively compromised.

```
typedef struct IHT {
      int Enabled;                  // Incremented upon enable
                                    // decremented upon disabled
      unsigned Reason;              // The interrupt source which
                                    // will trigger this callback
      volatile unsigned *Signal;    // Signal to increment on
                                    // interrupt
      void (*Callback)(struct IHT *Handle, unsigned Reason);
                                    // The callback routine
      } InterruptBlockType;

typedef InterruptBlockType *InterruptHandle;
```

Listing A3.2 – Interrupt Handle Data Structure

At present all interrupts are processed on a non-reentrant basis through the Interrupt Event Dispatcher and all have the same priority level. When the interrupt dispatcher is operating all interrupt sources are disabled. Because the Timer Event Dispatcher is derived from the Interrupt Event Dispatcher the processing of any Timer Events can cause interrupt processing to be unavailable for relatively long periods of time.

The solution to this problem is to prioritize interrupt sources and re-enable interrupt processing inside the Interrupt Event Dispatcher when low priority interrupts are being processed. For instance, the serial communications interrupt should be given a higher priority than the slow timer interrupt i.e. the serial communications interrupt should be able to interrupt an already executing slow timer interrupt in order to service the serial port but a slow timer interrupt should not be able to interrupt a serial communications interrupt. This solution has not yet been implemented in the FRPM since its current performance level is satisfactory, however any future development should consider implementing prioritization.

In most processors interrupt sources are prioritized in hardware either on a programmable or a fixed basis. It is curious that this feature is not present in the CL-PS7110 device. The ARM processor also provides two interrupt priority levels, the low priority interrupt request (which causes a switch to IRQ32 mode) and the high priority fast interrupt request (which causes a switch to FIQ32 mode), of which the fast interrupt request is intended to support channel-type services

with minimal overhead such as Direct Memory Access (DMA). It is then doubly curious that the CL-PS7110 does not use the Fast Interrupt Request for the high performance requirement of serial communications but does use it for the seemingly low performance requirement of the Battery Low interrupt source.

A3.8.2 The Timer Event Dispatcher

The System Software provides support to allow the three hardware timers available in the CL-PS7110 to be shared between tasks. The three available timers are the FastTimer, the SlowTimer, and the ApplTimer. The FastTimer and SlowTimer have fixed tick rates of 2kHz and 64Hz respectively, while the ApplTimer has a variable rate between 0.03Hz and 256kHz that can be selected by the application. Note however that the System Software may not be able to reliably process tick rates much higher than 10kHz due to event handling latencies.

The FRPM System Software abstracts these timers in the OSTimer module by providing functions to create, change, and delete virtual timers. These virtual timers can be used to either call an event handler function or increment a counter variable (flag) at a specified rate. For instance if a virtual timer using the SlowTimer is created with a period of 32 it will generate an event every 500ms.

Each physical timer has an independent event dispatcher with a list of virtual timers associated with it, each timer being described by a TimerHandle (Listing A3.3). Every time the physical timer ticks an interrupt event causes this list to be searched and all active virtual timers to have their period decremented. Blocked (inactive) timers are ignored. When the period reaches zero it is reloaded and any associated event (either a flag or event function) is processed. Only a fixed number of timers are available to ensure interrupt latencies are not excessively compromised. If the list is empty the associated interrupt event is blocked to avoid the time that would be wasted by searching an empty list.

```
typedef struct TBT {
      unsigned Timer;        // The type of this timer block
      int Enabled;           // Positive if timer is enabled
      unsigned Period;       // Period of timer in ticks
      unsigned TickCounter; // Decremented on each tick
                             // When zero reloaded with Period
      volatile unsigned *Signal;   // Signal to increment on
                                   // interrupt
      void (*Callback)(struct TBT *Handle, TimerType timer);
                                   // The callback routine
      } TimerBlockType;

typedef TimerBlockType *TimerHandle;
```

Listing A3.3 – Timer Handle Data Structure

A3.8.3 The Real Time Clock and Alarm Event Dispatcher

The CL-PS7110 incorporates a Real-Time Clock (RTC) based around a 32-bit counter (RTCCount) that increments once per second and a 32-bit match register (RTCMatch) that can be set to generate an alarm event when its time matches the time in the RTC count register. The RTCDIV field of the System Flags register (SysFlags) contains an extended count for the RTC count register giving the number of 64Hz ticks that have passed since the RTC count register was last incremented.

The zero-time reference for the FRPM is the same as that of the UNIX operating system (Midnight, January 1, 1970) which gives the time when the FRPM clock overflows back to zero as sometime in 2106. However, the UNIX-derived C runtime library used in the FRPM software will cause an earlier failure on January 18, 2038.

The FRPM System Software abstracts the RTC in the OSTimer module by providing functions to set and retrieve the current time, and to create, change, and delete alarms. Alarm events can be used to either call an event function or increment a counter variable (flag) at a specified time.

When an alarm is created, if the desired alarm time is in the past relative to the system clock the alarm creation will fail. When an alarm is changed, if the new desired alarm time is in the past relative to the system time the event function is called immediately. If the system time is changed all alarm times are changed so that their relative time remains unchanged.

The alarm event dispatcher is implemented in a similar way to the timer event dispatchers and uses the same data structures. However, an alarm event only occurs once, when the alarm time matches the current system time. At this point the alarm time should be either changed or the alarm deleted. Alarm events can be blocked and unblocked using the respective timer blocking and unblocking functions.

A3.9 Power Management

To enable the FRPM to operate on solar power alone for long periods of time, power management received critical attention during design of the FRPM. The CL-PS7110 processor provides hardware support for several power management modes while the FRPM hardware provides support for measurement and identification of the overall power consumption state of the system.

The power management services are based largely around the OSPwrMan module with other modules assisting via the standby and wakeup events to minimize power consumption as much as possible.

A3.9.1 Standby, Idle and Safe Modes

The CL-PS7110 processor supports both Standby and Idle power saving modes allowing a considerable increase in battery life if these modes are used wherever possible in both System Software and application code. In Standby mode the CPU is stopped and all peripherals except the real-time clock are disabled. In Idle mode only the CPU is stopped with all peripherals left active. Safe mode is a software extension of the Standby mode that forces the FRPM into hibernation if the remaining battery charge gets critically low.

Idle mode is entered by calling the Idle procedure which writes to the EnterIdle register, halting the CPU. Once in idle, the processor hardware will only re-enable the CPU once an interrupt is received, allowing execution to continue. Idle mode saves power by stopping the CPU execution and by preventing accesses to external memory.

Standby mode is entered by calling the Standby procedure which forces the FRPM hardware into a low power state before forcing the CL-PS7110 to enter standby. Once in Standby mode the only mechanisms to allow the processor to wake, allowing execution to continue, are the processor's Wakeup input (currently connected to the User Wakeup push-button, see Section A2.1.3), any external interrupt sources (currently not used), and the real-time clock alarm interrupt.

Before entering Standby mode the Standby procedure first searches the list of standby event handlers, servicing any tasks or modules that need to perform additional work to prepare themselves for Standby mode, and switches off any hardware the OSPwrMan module itself is responsible for, before placing the processor in standby. Once the processor is woken the list of wakeup event handlers is searched and registered tasks and modules are notified of the event. It should be noted that if the processor was brought out of standby by an interrupt, that interrupt will be serviced before the wakeup event is posted.

Safe mode is intended as a last-resort measure to save the system from a power failure if the remaining battery charge reaches a critically low level. It forces all of the FRPM hardware into the lowest possible power consumption state and attempts to recharge the battery over a 24 hour period before allowing any attempt at a wakeup. Although this may result in the disruption of an experiment and loss of communication with the unit for a period of time, it is considerably better than losing the entire system completely, particularly if the unit is in the field and retrieval is difficult.

The critical battery point is currently detected via the processor's BatteryOK input being forced low once the battery terminal voltage reaches 4.2V. The interrupt resulting from the transition in the BatteryOK input forces the current battery state to BatteryFlat. If the battery is still in this state the next time Standby is called it will first process the list of event handlers to show some cooperation before calling SafeMode to force all peripherals to their lowest power consumption state, disable all interrupt sources except the real-time clock interrupt, and set the real-time clock alarm to attempt to wake the FRPM in 24 hours' time before placing the processor in standby.

Once the FRPM is in standby, any attempt to wake it will be blocked by the CL-PS7110 hardware unless the BatteryOK signal becomes high again (indicating the battery is charged) or the FRPM is plugged into an external power source. Upon wakeup the system's original state is restored, any wakeup event handlers are called, and execution continues.

Safe mode can potentially be thwarted by any task that places a large power demand on the battery, including radio module operation. If the battery voltage only just exceeds the critical threshold when the FRPM is woken, a large power demand may pull the weakened terminal voltage down forcing the FRPM into Safe mode once again. If the charging rate of the battery is very low then the battery will eventually discharge to the point where the processor can no longer be restarted from suspend and the system will fail. However, the 24 hour period before any wakeup attempt is made should prevent this problem since even in very cloudy weather sufficient charging would occur to at least partially recharge the battery. In practice, Safe mode has yet to be entered by a unit in field use.

A3.9.2 Standby and Wakeup Events

The Power Management software provides support for modules or applications to receive events indicating that the FRPM is leaving or entering its Standby state. The Standby event is intended to allow modules to finalize any activity, including flushing communications channels and switching off any extra hardware, before the FRPM enters Standby mode. The Wakeup event is intended to provide the opposite service to allow modules to re-initialize themselves and switch on any extra hardware before the main execution thread continues.

Both of these events allow only handler functions to be called unlike interrupt and timer events which allow both handler functions and flags. The event dispatchers are implemented as two separate lists of callbacks, one for Standby events and the other for Wakeup events. When the Standby function is called the Standby list is searched and all valid event handlers are called before the Standby function places the FRPM in Standby mode. Upon wakeup, the Standby function calls all valid Wakeup handlers in the Wakeup list before returning control to the caller.

The power management events and their handlers execute in SVC32 mode and are completely re-entrant (any event can be manipulated from within any event function). Any events created by an application are marked as such and destroyed upon application exit by the System Software if the application has not destroyed them itself.

A3.9.3 Power Manager Tasks

Two tasks are used to manage the power state of the FRPM, these being the UpdatePowerManager and UpdateBatteryCharger tasks. Both of these tasks are queued to run on a periodic basis by separate timer events. UpdatePowerManager is queued every 10 seconds while UpdateBatteryCharger is queued every 10 minutes. Every time EnterIdle is called it checks to see if either to these tasks are queued. If the task is queued then it is executed.

The UpdatePowerManager task updates the PowerManager structure to the current state by measuring the current battery temperature, battery voltage, battery current, and input voltage. From these four measures the current power source and battery charging source can be determined. This information affects how the power management software behaves in respect to battery charging and power saving modes. For example, if the FRPM is currently supplied by an external power source the auto-off function for the command line interface will be disabled.

The UpdateBatteryCharger task manages the battery charging operation. The current configuration of the FRPM uses NiMH batteries which are sufficiently rated to continuously accept the full charge current from the solar panel without damage. Therefore battery charging is always enabled at the full solar panel output current. By arranging the battery charger as a task it enables more complex charging strategies to be implemented in the future.

A3.10 Communications Support

The FRPM currently supports only two communications channels, these being RS232 Serial Communications and the FRPM Radio Data Network. There is also support on the FRPM Main Board for an IrDA Infrared Data Link but this is not currently implemented in software. Future enhancements to the FRPM communications system could include the packetization of data to allow for multiple virtual communications channels within one data channel or the use of Ethernet-based radio modems to improve system flexibility. Because of these various options the FRPM communications support was designed to provide maximum flexibility and the ability to hide any future enhancements from any developed applications.

A3.10.1 Communications Abstraction Layer

To hide the details of any communications channel currently being used the FRPM System Software implements a Communications Abstraction Layer in the OSComms module that isolates communications as seen from the application from whatever real communications implementation is being used.

The Abstraction Layer allows channels to be created by any executing application and given an identifier that would associate that channel with the endpoints at both the FRPM and base station ends of the channel. This would allow multiple virtual channels to occupy the one physical channel. From that point on, any read or write access to that channel would occur independently of any other channel. However, this feature is not yet fully implemented and the created channels cannot be regarded as independent. Currently all channel accesses are reflected to the Serial Communications module OSSerial.

The Abstraction Layer also supports redirection of any channel output to a file where any data written to the channel will instead be written to a file in the FRPM Main File System. Note that read access to a redirected channel will always fail.

A3.10.2 Serial Communications

The OSSerial module provides the serial communications implementation for the FRPM System Software. This module supports all of the possible serial communications channels: RS232 communications, the FRPM radio data network, and the IrDA data link (currently this echoes to the RS232 port). It provides support for access on an individual byte, zero-terminated string, or block basis with optional hardware handshaking. Data rates from $110bs^{-1}$ to $115.2kbs^{-1}$ are supported and the serial word format is always 8 data bits, one stop bit, no parity bit.

The OSSerial module is effectively split into two halves, these being the receive and transmit functions. Each function is based around a circular buffer implemented as a byte array. The serial interface hardware in the CL-PS7110 consists of both receive and transmit FIFO buffers (UartData) and the serial interface control register (UartControl). Several related flags are also found in the system flags register (SysFlags).

Whenever the 16-byte receive FIFO is more than half full, or has data and no new data has been received for a three-character period, the SerialReceiveHandler is called to move data from the receive FIFO into the receive buffer. Once the receive buffer becomes full any received data will be discarded until space in the buffer becomes available again. It is the responsibility of the OSComms abstraction layer to recover from any such error. Similarly, any word received with a framing error will be discarded. If the serial port receive FIFO buffer overflows, its entire contents are discarded.

When the 16-byte transmit FIFO is less than half full the SerialTransmitHandler is called to fill the transmit FIFO from the transmit buffer. If the transmit FIFO can no longer be filled due to the transmit buffer being empty the SerialTransmitHandler is disabled until new data is available. The transmit buffer will only accept data from if space is available otherwise it will return an error to the function caller.

The OSSerial module also implements hardware RTS/CTS handshaking to avoid data loss for both receive and transmit functions if a channel that supports handshaking is selected. If receive queue is getting full the SerialTransmitHandler requests the source pause transmission by setting RTS high. If the source does not acknowledge the request and data continues to arrive then the receive FIFO may overflow and data loss will result. As soon as a reasonable amount of space is available in the receive queue RTS is set low again to signal the source to resume transmission.

As long as the CTS line is kept low the SerialTransmitHandler will continue transmitting data. If the CTS line becomes high then SerialTransmitHandler will be disabled until CTS becomes low again. A falling edge on CTS will cause the SerialStatusChangedHandler to be called which will re-enable the SerialTransmitHandler. Note that when SerialTransmitHandler is disabled by CTS

going high there may be up to 16 characters present in the transmit FIFO which will still be sent. Therefore when the destination gets near full it should signal CTS high with at least 16 bytes of free space left in its receive queue or else data loss will result.

A3.10.3 Radio Management

Because of the transparent nature of the WIT2400 radio modules, the transmission and reception of data via the radio link looks no different than the RS232 interface. The WIT2400 modules manage all error correction and buffers data in case of link fading or drop-out. All that is required of the System Software is to ensure the radio module powers-up and powers-down in the correct sequence. Communication with the radio module is performed with hardware handshaking enabled at $115.2kbs^{-1}$. The OSRadio module contains any radio channel-specific functions, particularly those for powering up and shutting down the radio module.

A3.11 The File System

The FRPM System Software includes a file system, the Filer, to enable the storage of both applications and data on the FRPM. The Filer consists of two areas of memory: a 10MB area of DRAM (the Main File System) which provides non-permanent (volatile) storage space and is expected to mainly hold transient data sets or applications, and a 1792kB area of Flash memory (the Flash File System) which provides permanent (non-volatile) storage to hold commonly-used applications or data sets that must be preserved, even in the event of a total power failure.

The Filer allows the use of filenames of up to 63 characters. No file sub-directory support is provided since it is envisaged that the Filer contents will tend to be several large files rather than many small, difficult to manage files. The Filer does not support shared access to a file - even if a task opens a file for read-only access no other task will be allowed any access. Because the FRPM System Software only supports a single application executing at any one time this is not an inconvenience.

The OSFiler module is the main access point for any file operations. If the Filer determines that the file actually resides in the Flash File System then the operation request is re-directed to the OSFlash module.

A3.11.1 The Main File System

The Main File System is based around a doubly-linked list of file headers (Listing A3.4) into which files can be both inserted (created) or removed (deleted). Files are always created at the 'front' of the list with the doubly-linked list structure allowing the deletion of any file without breaking the list. Each header in turn points to the actual file data, whether it be stored in Flash

memory or DRAM memory, with the data stored as an uncompressed continuous array of bytes. The file header contains information about the file data itself as well as information on the current access mode and the current offset within the file.

```
typedef struct FIBT {
        struct FIBT *Previous;      // Previous FileInfoBlock
        struct FIBT *Next;          // Next FileInfoBlock
        byte *File;                 // Memory allocated to this file
        char Name[FILE_NAME_LENGTH];            // Name of the file
        char Creator[FILE_CREATOR_LENGTH];      // Creator of file
        char Date[FILE_DATE_LENGTH];            // Last changed
        unsigned Checksum;          // Checksum over the whole file
        unsigned Open : 1;          // True if this file is open
        unsigned OpenedByApp : 1;   // TRUE if opened by application
        unsigned Type : 2;          // The file type
        unsigned Access : 2;        // The current file access mode
        unsigned InFlash : 1;       // Is this file in flash memory?
        unsigned FlashSectors : 19;// If the file is in flash,
                                    // what sectors it uses
        unsigned _BitPad : 6;       // Padding
        unsigned Allocated;         // Allocated size of this block
        unsigned Length;            // Size of this block so far
        unsigned Position;          // Position we are currently
                                    // accessing
        } FileInfoBlockType;
```

Listing A3.4 – File Header Data Structure

The underlying framework of the file system is a heap data structure, itself a doubly-linked list of memory block descriptors which record where blocks of memory are reserved and how large the blocks are. Whenever the size of a file is increased the heap is searched for a free continuous block of memory that can contain the new size. The file is then moved to this new space and the file header and heap block descriptor list updated.

The Main File System cluster size of 16kB which helps to avoid heavy fragmentation of memory caused by frequent re-allocation of memory blocks and helps to ensure that any released blocks will still be large enough to be useful. By searching the heap from the current position to the end rather than starting each search for a free block from the start of the heap, the earlier allocated blocks are given the maximum possible time to be freed and collapse into larger, more useable, blocks. This method also generally guarantees the shortest possible search time to find a free block.

File integrity is determined by the use of a checksum over the file data. Any files opened by an application are marked as such and closed upon application exit by the System Software if the application has not closed them itself. Provided power is not lost to the Main File System, there is usually sufficient error checking provided by cross-checking the list linkage and checksums to enable the Filer contents to be reconstructed after a system restart.

The reconstruction algorithm first attempts to reconstruct the heap from both ends, if this fails the File System is considered irretrievably lost and is completely reset. If the heap is found to be intact in either direction it is reconstructed and the Filer linked-list is stepped through from the start of the list to the end and checked for breaks. If a break is found in the list another search

begins from the end of the list towards its start. If another break is found all files between the two breaks are considered lost but the list is patched to rejoin the two halves and recover the files they contain. If either search makes it all the way to the other end the Filer list is either intact or can be completely repaired.

Once the Filer linked-list is restored any files that are resident in Flash memory are removed and the remaining files are each checked against the list of allocated blocks in the heap. Any allocated blocks that do not match a stored file are freed, allowing memory reserved by lost files to be recovered. All file contents are also checked against their respective checksum and any corrupted file is deleted. After re-inserting any files resident in Flash memory the Filer has been completely reconstructed and any files within it can be considered intact.

The Main File System is protected from the potentially corrupting effects of an accidental write access by describing the Filer memory area as a separate domain in the MMU Translation Table and keeping this Filer Domain locked (read-only access) at all times. Whenever a function within the Filer wishes to have write access to a file the File Domain is unlocked by the System Software to allow both read and write access. When the write access is finished the domain is again locked. Using this technique, if any executing code outside of the privileged Filer functions attempts write access to the Main File System, an Illegal Access exception will be generated, the write will be blocked and the IllegalAccessHandler called.

A3.11.2 The Flash File System

The Flash File System provides permanent (non-volatile) storage space for up to 14 files at any one-time. Each file occupies at least a single 128kB sector and could potentially occupy all 14 sectors if no other files were present. By organizing the files on a sector basis a single sector erase operation can be used to delete that file from flash memory (multiple sectors can be erased by a single sector erase operation). Although this 128kB cluster size is convenient for manipulating the Flash memory it can potentially lead to large under-utilized areas in the Flash filing system that a more complex system could overcome.

Files are always created in the Main File System which allows unrestricted read/modify/write access to any files in contains. Once a file has been created and then closed it can be moved to the Flash File System at the request of the application. A file in Flash memory can be neither modified nor written to, it can only be read. To modify a file in Flash memory it must first be moved back into the Main File System, modified, then optionally moved back to the Flash File System.

Because of the need to access any files resident in Flash memory through the Main File System the Flash File System is quite closely linked to the data structures involved with managing the Main File System. When a file is moved from the Main File System into Flash memory the original header is maintained in the Main File System linked list with the File field adjusted to point to the file data's new location within Flash memory and some of the file's flags adjusted to

indicate its new read-only status. The header stored with the data in Flash memory is only used if the File System has to be reconstructed after a power fail condition. The stored header is then copied out of Flash memory and inserted into the Main File System's list.

Since the Flash File System contents are physically organized on a sector-basis the natural data structure to represent its contents is a 14-element array where each element is the size of a sector of Flash memory. By overlaying this data structure against the Flash memory a convenient method of accessing the Flash memory as a file system is obtained.

Each file in the Flash File System starts on a sector boundary. The first few bytes are taken up with a file header (FileInfoBlockType) with the file data following immediately after the header. A valid file in Flash memory has the Next and Previous fields of its header set to `0x12345678` allowing a simple search of the first two words of every sector to in Flash memory to locate any valid files. If a Flash-resident file occupies more than a single sector then it is unlikely that the first two words of a sector boundary within the file will contain this pattern.

Files resident in the Flash File System are 'deleted' from Flash memory by setting the Next and Previous fields of its header to `0x00000000` to mark the sector as unused (Flash memory technology allows ones to be programmed to zeros but not vice-versa). The contents of the file are still present until the creation of a new Flash-resident file requires the space and issues a sector erase operation. The process of moving a file from Flash memory to the Main File System actually consists of a memory copy operation followed by this deletion operation. By only marking a file as deleted and performing the erase operation later system response is improved during deletion at the expense of a longer programming time.

The Flash File System is protected from the potentially corrupting effects of an accidental write access by describing the Flash memory area as a separate domain in the MMU Translation Table and keeping this Flash Domain locked (read-only access) at all times. Whenever a privileged function wishes to have write access to the Flash memory area the Flash Domain is unlocked to allow both read and write access. When the write access is finished the domain is again locked. Using this technique, if any executing code outside of the privileged Flash functions attempts write access to the Flash memory area, an Illegal Access exception will be generated, the write will be blocked and the IllegalAccessHandler called.

The OSFlash module also manages the Flash-resident images of the System Software and the not-yet-implemented Interpreter, both of which occupy fixed sectors within Flash memory. There are routines within OSFlash that allow either of these images to be replaced under control of the System Software.

A3.12 Command Line Interface

The FRPM System Software contains a command-line interface (module OSCli) that allows for system management by the user. The command-line interface provides support for such operations as file management, obtaining status information and setting parameters such as the system time. It also provides a mechanism to run applications with any desired parameters.

The command-line interface is the default interface to the FRPM System Software. Its basic operation is to wake the FRPM every minute via an alarm event, open the currently selected communications channel, check for any commands for 5 seconds, close the channel and put the FRPM back into standby mode again. If any commands are received the command line is started and any valid commands are executed. The command line will automatically exit and place the FRPM in standby again if no valid commands are received for a two minute period.

By keeping the FRPM in standby and only waking it for short periods the battery life can be greatly extended, especially when operating on solar power with the radio as the communications channel.

Full details on the command syntax and capabilities of the command-line interface can be found in the FRPM Users Manual[138].

A3.13 Command Sequencer

The Command Sequencer allows the user to load the FRPM System Software with a sequence of instructions to be executed at set times. This facility is generally used for starting experiments, or parts of an experiment, at particular times. It also allows multiple FRPM units to be synchronized within an experiment to ensure that all units are taking the same measurements at the same time. The Command Sequencer is implemented in OSSqunce module.

The Command Sequencer can execute any available command, or group of commands, that the Command Line supports and can execute that command either once at a set time (a Single-Shot command) or several times at fixed intervals (a Periodic command).

The Sequencer scans a text-format sequence file from the File System to extract the commands that need to be executed and the information on when to execute them. The file is first scanned for Periodic commands which are loaded into an array of InstructionType structures (Listing A3.5) which keeps track of how many times the command has been executed and when it should be next executed. Once all Periodic commands have been found the array is scanned to find the command that will occur next and an alarm event is set for that time. When the alarm occurs the array is searched and any commands with a matching time are executed via the Command Line Interface. Once the command is finished the list is searched for the command that will occur next and the alarm event is changed accordingly.

```
typedef struct {
      byte IsPeriodic;     // TRUE if command is periodic
      unsigned StartTime; // Start time of instruction execution
      unsigned Interval;  // The time between command executions
      unsigned Count;     // Number of times to execute command
      unsigned Line;      // Line instruction is in sequence file
      char *CommandLine;  // The command line to execute
      } InstructionType;
```

Listing A3.5 – Instruction Information Data Structure

Any Single-Shot commands present in the sequence file are assumed to be present in their sequence of execution hence the file is scanned again and each command found is used in that sequence. Every command found causes an alarm event to be set and the FRPM placed in Standby mode until that alarm occurs whereupon the command will be executed via the Command Line Interface and the file scanned further until another command is found. In this way all Single-Shot commands are processed. If the start time of a command is in the past relative to the FRPM system time the command is executed immediately. Once all Single-Shot commands have been executed the Sequencer waits until all Periodic commands have finished before returning control to the Command Line Interface.

As an example, the sequence file in Listing A3.6 will begin a periodic command (record ECG for 3 minutes at 100 samples/sec) at 9am in the morning of the same day (`reltoday`) the sequence file was started. This period command will be executed every 5 minutes, 72 times, creating 72 files of the form `ecg_nn.raw`. At 4pm on the same day, a single-shot command (compress all files matching `*.raw` are save result to Flash memory) will be executed. After execution of this last command the sequence file is finished and control returns to the Command Line Interface.

```
# Example sequence file
# Execution flags
! reltoday logfile Trial.log
# Periodic commands
# [Start hh:mm:ss dd-mm-yy],[Interval hh:mm:ss],[Count
n]\t[Command]
9:00:00 0-0-0,00:05:00,72  sampler -raw -file ecg.raw -rate 100 -
count 18000 -adc 11
# Single-shot commands
# [Start hh:mm:ss dd-mm-yy]\t[Command]
16:00:00 0-0-0             container ecgdata -add *.raw -flash
# End of sequence file
```

Listing A3.6 – Example Sequence File

Full details on the Command Sequencer and its capabilities can be found in the FRPM Users Manual[138].

References

[1] Hemsworth, P.H., Barnett, J.L., Beveridge, L., Matthews, L.R., "The welfare of extensively managed dairy cattle: A review", Applied Animal Behaviour Science, Vol. 42, pp. 161-182, 1995.

[2] Cook, C.J., Devine, C.E., Gilbert, K.V., and Maasland S.A., "Animal welfare research in the meat industry", In: Proceedings of NZVA Branch Meeting on Animal Welfare (ed. Petterson, G). Rotorua, 1992.

[3] Sapolsky, R.M., "Stress, the aging brain, and the mechanisms of neuron death", Bradford Books, MIT Press, 1992.

[4] Estimate based upon animal industry export earnings in 1996 NZ Yearbook and on conservative estimates of between 1-10% downgrading due to poor quality.

[5] Vail, G., " 'Cruel' farming in the name of freedom", New Scientist, 9 July 1994, pp. 4.

[6] Bonner, J., "Roads to the abattoir", New Scientist, 11 March 1995, pp. 30-31.

[7] Ladewig, J., Matthews, L.R., "The importance of physiological measurements in farm animals stress research", Proceedings of the New Zealand Society of Animal Production 1992, Vol. 52, pp. 77-79, 1992.

[8] Sapolsky, R (1982) Stress, the aging brain and the mechanisms of neuron death. MIT Press, Cambridge, Massachusetts.

[9] Sapolsky, R (1982) The endocrine stress response and social status in the wild baboon. Hormones and Behavior 15, 279-286.

[10] Rollin, BE (1997) Animal ethics, social change and the meat industry. In: Proceedings of the 43rd International Congress on Meat Science and Technology. Auckland, NZ.

[11] Diamond, DM (1990) Exposure to a novel environment. Psychobiology 18, 273-281.

[12] Bennet, MC, Diamond, DM, Fleshner, M and Rose, G (1991) Serum corticosterone level predicts the magnitude of hippocampal primed burst potentiation and depression in urethane anaesthetised rats. Psychobiology 19(4), 301-307.

[13] Cooper, R., Osselton, J.W., Shaw, J.C., EEG Technology, 3rd edition, Butterworths, London, 1980.

[14] Brodal, A., "Neurological anatomy in relation to clinical medicine", 3rd edition, Oxford University Press, New York, 1981.

[15] Kolb, B., Whishaw, I.Q., "Fundamentals of Human Neuropsychology", 3rd Ed., W.H. Freeman and Co., New York, 1990.

[16] Sherrington, C., "The integrative action of the nervous system", 2nd Ed., Yale University Press, New Haven, 1967

[17] Jones, E.G., Fiedman, D.P., "Projection pattern of functional components of thalamic ventrobasal complex on monkey somatosensory cortex", Journal of Neurophysiology, Vol. 48, pp. 521-544, 1982

[18] Scholl, D. A., "Organisation of the Cerebral Cortex", Methuen, London, 1956.

[19] Kandel, E.R., Schwartz, J.H., Jessell, T.M., "Principles of neural science", 3rd edition, Appleton and Lange, Norwalk, Connecticut, 1991.

[20] Kandel, E.R., Schwartz, J.H., Jessell, T.M., "Principles of neural science", 3rd edition, pp. 19, Appleton and Lange, Norwalk, Connecticut, 1991.

[21] Anderson P., Eccles, H. C., Løyning, Y., "Recurrent Inhibition in the Hippocampus with Identification of the Inhibitory Cell and its Synapses", Nature, Vol. 198, No. 540, 1963.

[22] Li, C. L., Jasper, H. H., "Microelectrode studies of the electrical activity of the cerebral cortex in the cat.", Journal of Physiology, Vol. 121, No. 117, 1953.

[23] Pons, T., Garraghty, P., Friedman, D., Mishkin, M., "Physiological evidence for serial processing in somatosensory cortex", Science, Vol. 237, pp. 417-420, 1987.

[24] Abraham, K., Ajmone-Marsan, C., "Patterns of cortical discharges and their relation to routine scalp electroencephalogram", Electroencephalography and Clinical Neurophysiology, Vol. 10, No. 447, 1958.

[25] Jacobson, A., Kaler, A., Lehmann, D., Zweizig, J., "Somnambulism: All-night electroencephalographic studies", Science, Vol. 148, pp. 975-977, 1965.

[26] Cooper, R., Osselton, J.W., Shaw, J.C., EEG Technology, 3rd edition, pp. 187-188, Butterworths, London, 1980.

[27] Picton, T.W., Lins, O.G., Sherg, M., "The recording and analysis of event-related potentials", In: Boler, F., Grafman, J., (Eds.), "Handbook of Neuropsychology, Vol. 10", pp. 7-10, Elsevier, Amsterdam, 1995.

[28] Cooper, R., Osselton, J.W., Shaw, J.C., EEG Technology, 3rd edition, pp. 198-200, Butterworths, London, 1980.

[29] Chiappa, K.H., "Evoked potentials in clinical medicine", pp. 158-190, Raven Press, New York, 1983.

[30] Cooper, R., Osselton, J.W., Shaw, J.C., EEG Technology, 3rd edition, pp. 218-221, Butterworths, London, 1980.

[31] Chiappa, K.H., "Evoked potentials in clinical medicine", pp. 270-312, Raven Press, New York, 1983.

[32] Van Orden, K.F., Ahlers, S.T., *et al.*, "Moderate cold exposure shortens evoked potential latencies in humans", Aviation, Space and Environmental Medicine, Vol. 61, pp. 636-639, 1990

[33] Creel, D.J., Dustman, R.E., "Visual evoked responses in the rat, guinea pig, cat, monkey, and man", Experimental Neurology, Vol. 40, No. 2, 1973.

[34] Vaughan, H.G., "The analysis of scalp recorded brain potentials", In: Thompson, R.F., Patterson, M.M., (Eds.), "Bioelectric recording techniques, Part B, Electroncephalography and human brain potentials", pp. 157-207, Academic Press, New York, 1974.

[35] Glaser, E.M., Ruchkin, D.S., "Principles of neurobiological signal analysis", New York: Academic Press, 1976.

[36] Cooper, R., Osselton, J.W., Shaw, J.C., EEG Technology, 3rd edition, pp. 189-194, Butterworths, London, 1980.

[37] Picton, T.W., Lins, O.G., Sherg, M., "The recording and analysis of event-related potentials", In: Boler, F., Grafman, J., (Eds.), "Handbook of Neuropsychology, Vol. 10", pp. 13-15, Elsevier, Amsterdam, 1995.

[38] Cooper, R., Osselton, J.W., Shaw, J.C., EEG Technology, 3rd edition, pp. 198-200, Butterworths, London, 1980.

[39] Chiappa, K.H., "Evoked potentials in clinical medicine", pp. 1-5, Raven Press, New York, 1983.

[40] Cooper, R., Osselton, J.W., Shaw, J.C., "EEG technology", 3rd edition, pp. 214, Butterworths, London, 1980.

[41] Chiappa, K.H., "Evoked potentials in clinical medicine", pp. 180-182, Raven Press, New York, 1983.

[42] Cooper, R., Osselton, J.W., Shaw, J.C., EEG Technology, 3rd edition, pp. 214, Butterworths, London, 1980.

[43] Chiappa, K.H., "Evoked potentials in clinical medicine", pp. 106, Raven Press, New York, 1983.

[44] Strain, G.M., Green, K.D., Twedt, A.C., Tedford, B.L., "Brain stem auditory evoked potentials from bone stimulation in dogs", American Journal of Veterinary Research, Vol. 54, No. 11, 1993

[45] Pierson, L.L., Gerhardt, Griffiths, S.K., K.J., Abrams, R.M., "Auditory Brainstem Response in Sheep. Part I: Prenatal Development", Developmental Psychobiology, Vol. 28, pp. 293-305, 1995.

[46] Griffiths, S.K., Pierson, L.L., Gerhardt, K.J., Abrams, R.M., Peters, A.J.M., "Auditory Brainstem Response in Sheep. Part II: Postnatal Development", Developmental Psychobiology, Vol. 29, No. 1, 1996.

[47] Kulli, J., C., "Does anesthesia cause loss of consciousness?", Trends in Neuroscience, Vol. 14, pp. 6-10, 1991.

[48] Mustafa, K.Y., Aneja, I.S., Khogali, M., Nasreldin, A., Arar, I., "Effect of Hyperthermia on Brainstem Auditory Evoked Potentials in the Conscious Sheep", Electroencephalography and Clinical Neurophysiology, Vol. 71, pp. 133-141, 1988.

[49] Näätänen, R., Alho, K., "Event-related potentials in human selective attention research", In: Boler, F., Grafman, J., (Eds.), "Handbook of Neuropsychology, Vol. 10", pp. 76-79, Elsevier, Amsterdam, 1995.

[50] Cooper, R., Osselton, J.W., Shaw, J.C., "EEG technology", 3rd edition, pp. 221-222, Butterworths, London, 1980.

[51] Sutton, S., Braren, M., Zubin, J., John, E.R., "Evoked Potential Correlates of Stimulus Uncertainty", Science, Vol. 150, pp. 1187, 1965.

[52] Cooper R., McCallum, W.C., Newton, P., Papakostopoulos, D., Pocock, P.V., Warren, W.J., "Corticol potentials associated with the detection of visual events", Science, Vol. 196, pp. 74, 1977.

[53] Walter, W.G., *et al.*, "Contingent negative variation: an electric sign of sensorimotor association and expectancy in the human brain", Nature, Vol. 203, pp. 380-384, 1964

[54] Proulx, G.B., Picton, T.W., "The effects of anxiety and expectancy on the CNV", Annals of the New York Academy of Sciences, Vol. 425, pp. 617-622, 1984.

[55] Low, M.D., Swift, S.J., "The contingent negative variation and the "resting" DC potential of the human brain: Effects of situational anxiety", Neuropsychologia, Vol. 9, pp. 203-208, 1971.

[56] Knott, J.R., Irwin, D.A., "Anxiety, stress and the contingent negative variation", Arch Gen Psychiatry, Vol. 29, pp.538-541, 1973.

[57] Chiappa, K.H., Gladstone, K.J., Young, R.R., "Brainstem auditory evoked responses: Studies of waveform variations in 50 normal human subjects", Arch. Neurol., Vol. 36, pp. 81-87, 1979.

[58] Sapolsky, R.M., "Stress, the aging brain, and mechanisms of neuron death", MIT Press, Cambridge Massachusetts, 1992.

[59] Sapolsky, R (1992) Neuroendocrinology of the stress-response. In: Becker J, Breedlovers, Cres D (eds). Behavioural Endocrinology. MIT Press, Cambridge, Massachusetts.

[60] Inoue, T, Koyama, T and Yamashita, I (1993) Effect of Conditioned Fear Stress on Serotorin Metabolism in the rat brain. Pharmacology Biochemistry and Behaviour 44, 371-374.

[61] Rose, R (1985) Psychoendocrinology. In Wilson, J, Foster, D (eds) Textbook of Endocrinology (7th Edit) Saunders, Philadelphia. USA.

[62] Herman, J, Adams, D and Prewitt, C, (1995) Regulatory changes in neuroendocrine stress-integrative circulty produced by a variable stress paradigm. Neuroendocrinology 61,180-190.

[63] Fraser, A.F., Broom, D.M., "Farm animal behaviour and welfare", 3rd Edition, Baillière Tindall, London, 1990.

[64] Cook, CJ and Devine, CE (1995) Glutamate receptors and nitric oxide contribute to habituation to repeated stress in sheep. Proceedings of the Physiological Society of NZ. 14, 53.

[65] Toates, F., "Stress: conceptual and biological aspects", John Wiley and Sons, New York, 1995.

[66] Kandel, E.R., Schwartz, J.H., Jessell, T.M., "Principles of neural science", 3rd edition, Appleton and Lange, Norwalk, Connecticut, 1991.

[67] Cohen, S., Kessler, R.C., Underwood-Gordon, L., "Strategioes for measuring stress in studies of psychiatric patients", In: Cohen, S., Kessler, R.C., Underwood-Gordon, L., (Eds.), "Measuring stress: A guide for health and social scientists", Oxford University Press, Oxford, 1995.

[68] Haynes, S.N., "Principles of Behavior Assessment", Gardner Press, New York, 1978.

[69] Prokasy, W.F., Raskin, D.C. (Eds.), "Electrodermal activity in psychological research", Academic Press, New York, 1973.

[70] Ingram, J.R., Matthews, L.R., McDonald, R.M., "Remote blood sampling device - A stress free blood sampling technique for free ranging animals", Proceedings of the New Zealand Society of Animal Production 1994, 54:39-42, 1994.

[71] Krantz, D.S., Falconer, J.J., "Measurement of cardiovascular responses", In: Cohen, S., Kessler, R.C., Underwood-Gordon, L., (Eds.), "Measuring stress: A guide for health and social scientists", Oxford University Press, Oxford, 1995.

[72] Lacey, J.I., "Somatic response patterning and stress: some revisions of activation theory", In: M.H. Appley, R. Trumball, (Eds.), "Physiological stress", Appleton-Century-Crofts, New York, 1967.

[73] Öst, L., Sterner, U., Lindahl, I., "Physiological responses in blood phobias", Behavior Research and Therapy, Vol. 22, pp. 99-108, 1984.

[74] Hernandez-Peon, R., Scherrer, H., Jouvet, M., "Modification of electric activity in cochlear nucleus during attention in unanesthetized cats", Science, Vol. 123, pp. 331-332, 1956.

[75] Drake, M.W., Pakalnis, A. Phillips, B., Padaman, H., Hietter, S.A., "Auditory evoked potentials in anxiety disorder", Clinical Electroencephalography, Vol. 22, No. 2, 1991.

[76] Bond, A.J., James, D.C., Ladar, M.H., "Physiological and psychological measures in anxious patients", Psychological Medicine, Vol. 4, pp. 364-373, 1971.

[77] Isreal, J.B., Wickens, C.C., Chesney, G.L., Donchin, E., "The event-related brain potential as an index of display-monitoring workload", Human Factors, Vol. 22, No. 2, pp. 211-224, 1980.

[78] Wilson, G.F., Fullenkamp, P., Davis, I., "Evoked response, cardiac, blink, and respiration measures of pilot workload in air-to-ground missions", Aviation, Space and Environmental Medicine, Vol. 65, No. 2, 1994.

[79] Kline, J.P., Schwartz, G.E., Fitzpatrick, D.F., Hendricks, S.E., "Defensiveness, anxiety, and the amplitude/intensity function of the auditory-evoked potentials", International Journal of Psychophysiology, Vol. 15, pp. 7-14, 1993.

[80] Grillon, C., Ameli, R., "P300 assessment of anxiety effects on processing novel stimuli", International Journal of Psychophysiology, Vol. 17, pp. 205-217, 1994.

[81] Chattopadhyay, P., Cooke, E., Toone, B., Lader, M., "Habituation of physiological responses in anxiety", Biological Psychiatry, Vol. 15, No. 5, 1980.

[82] M.W. Hill, R.P. Heavens, and B.A. Baldwin, "Auditory Evoked Potentials recorded from conscious sheep", Brain Research Bulletin, vol. 15, no. 5, pp. 453-458, 1985.

[83] Cook, C.J., Williams, C., Gluckman, P.D., "Brainstem Auditory Evoked Potentials in the Fetal Sheep, *in utero*", Journal of Developmental Physiology, vol. 9, pp. 429-439, 1987.

[84] K.Y. Mustafa, I.S. Aneja, M.Khogali, A. Nasreldin, and I. Arar, "Effect of hyperthermia on brain auditory evoked potentials in the conscious sheep", Electroencephalography and Clinical Neurophysiology, vol. 71, pp. 133-141, 1988.

[85] Chiappa, K.H., "Evoked potentials in clinical medicine", pp. 189-190, Raven Press, New York, 1983.

[86] Walter, W.G., Cooper, R., *et al.*, "Contingent negative variation and evoked responses recorded by radio-telemetry in free-ranging subjects", Electroencephalography and Clinical Neurophysiology, Vol. 23, pp. 197-206, 1967.

[87] The Horticulture and Food Research Institute of New Zealand Ltd., "FRPM Users Manual", Hamilton, New Zealand, 1998.

[88] Chiappa, K.H., "Evoked Potentials in Clinical Medicine", pp. 3, Raven Press, New York, 1983.

[89] Geddes, L.A., "Electrodes and the measurements of bio-electric events", pp. 44-94, John Wiley and Sons, New York, 1972.

[90] Wood, D.E., Ewins, D.J., Balachandran, W., "Comparative analysis of power-line interference between two- or three-electrode bio-potential amplifiers", Medical and Biological Engineering and Computing, Vol. 33, No. 1, 1995.

[91] Toennies, J.F., "Differential Amplifiers", Review of Scientific Instruments, Vol. 9, No. 25, 1938.

[92] Matthews, B. H. C., "A Special Purpose Amplifier", Journal of Physiology, Vol. 81, No. 28P, 1934.

[93] Metting van Rijn, A.C., Peper, A., Grimbergen, C.A., "Amplifiers for bio-electric events: a design with a minimal number of parts", Medical and Biological Engineering and Computing, Vol. 32, No. 3, 1994.

[94] Metting van Rijn, A.C., Peper, A., Grimbergen, C.A., "High-quality recordings of bio-electric events", Medical and Biological Engineering and Computing, Vol. 28, No. 5, 1990

[95] Winter, B.B., Webster, J.G., "Driven-Right-Leg Circuit Design", IEEE Transactions on Biomedical Engineering, Vol. 30, No. 1, 1983.

[96] Huhta, J.C., Webster, J.G., "60Hz interference in electro-cardiography", IEEE Transactions on Biomedical Engineering, Vol. 20, pp. 91-101, 1973.

[97] Morrison, R., "Grounding and shielding techniques in instrumentation", 2nd Ed., pp. 132-133, John Wiley and Sons, New York, 1977.

[98] Cooper, R., Osselton, J.W., Shaw, J.C., "EEG Technology", 3rd Ed., pp. 25, Butterworths, London, 1980.

[99] Moser, J.M., Aunon, J.I., "Classification and Detection of Single Evoked Brain Potentials Using Time-Frequency Amplitude Features", IEEE Transactions on Biomedical Engineering, Vol. 33, No. 12, 1986.

[100] Zouridakis, G., Jansen, B.H., Boutros, N.N., "A Fuzzy Clustering Approach to EP Estimation", IEEE Transactions on Biomedical Engineering, Vol. 44, No. 8, 1997.

[101] Clarson, V.H., Liang, J.J., "Mathematical Classification of Evoked Potential Waveforms", IEEE Transactions on Systems, Man, and Cybernetics, Vol. 19, No. 1, 1989.

[102] Thakor, N.N., Xin-Rong, G., Yi-Chun, S., Hanley, D.F., "Multiresolution Wavelet Analysis of Evoked Potentials", IEEE Transactions on Biomedical Engineering, Vol. 40, No. 11, 1993.

[103] Ott, H.W., "Noise Reduction Techniques in Electronic Systems", 2nd Ed., John Wiley and Sons, New York, 1988.

[104] Elberling, C., Don, M., "Quality estimation of averaged auditory evoked responses", Scand. Audiol., Vol. 13, pp. 187-197, 1984.

[105] Data Description Inc., "DataDesk Statistics Guide, Version 6.0", Ithaca, NY 14852, 1997, pp. 33/1-33/20.

[106] Data Description Inc., "DataDesk Statistics Guide, Version 6.0", Ithaca, NY 14852, 1997, pp. 27/1-27/10.

[107] Baher, H., "Analog and Digital Signal Processing", John Wiley and Sons, New York, 1990, pp. 61-121.

[108] Wojtaszczyk, P., "A Mathematical Introduction to Wavelets", Cambridge University Press, Cambridge, 1997.

[109] SAS Institute Inc., "SAS/STAT User's Guide, Version 6, Fourth Edition, Volume 2", Cary, NC, 1989, pp. 1242.

[110] Glaser, E.M., Ruchkin, D.S., "Principles of neurobiological signal analysis", Academic Press, New York, 1976.

[111] Picton, T.W., Lins, O.G., Scherg, M., "The recording and analysis of event-related potentials", In: Boller, F., Grafman, J., (Eds), "Handbook of neuropsychology", Vol. 10, pp. 33-46, Elsevier, Amsterdam, 1995.

[112] Moore, D.S., and McCabe, G.P., "Introduction to the Practice of Statistics, Second Edition", W.H. Freeman and Company, New York, 1993.

[113] Fahmy, T., "XLStat: An Excel Add-in for Statistics and Data Analysis", fahmy@xlstat.com, http://www.xlstat.com/

[114] SAS Institute Inc., "SAS/STAT User's Guide, Version 6, Fourth Edition, Volume 1", Cary, NC, 1989, pp. 45.

[115] MATLAB, The MathWorks Inc., 24 Prime Park Way, Natick, MA 01760-1500, USA.

[116] Baher, H., "Analog and Digital Signal Processing", John Wiley and Sons, New York, 1990, pp. 339-378.

[117] Baher, H., "Analog and Digital Signal Processing", John Wiley and Sons, New York, 1990, pp. 368.

[118] Chiappa, K.H., "Evoked Potentials in Clinical Medicine", Raven Press, New York, 1983, pp. 5.

[119] American Electroencephalographic Society, "Guidelines for standard electrode placement nomenclature", Journal of Clinical Neurophysiology, Vol. 8, pp. 200-202, 1991.

[120] May, N.D.S., "The Anatomy of the Sheep: A Dissection Manual", 3rd Ed., University of Queensland Press, 1970.

[121] Chiappa, K.H., "Evoked Potentials in Clinical Medicine", Raven Press, New York, 1983, pp. 113-114.

[122] Chiappa, K.H., "Evoked Potentials in Clinical Medicine", Raven Press, New York, 1983, pp. 128.

[123] Hill, M.W., Heavens, R.P., Baldwin, B.A., "Auditory evoked potentials recorded from conscious sheep", Brain Research Bulletin, Vol. 15, No. 5, pp. 453-458, 1985.

[124] Acoustical Society of America, "ANSI S3.4 – 1980 (R 1992) American national standard procedure for the computation of loudness of noise", New York, NY 10005-3993, USA.

[125] Chiappa, K.H., "Evoked Potentials in Clinical Medicine", Raven Press, New York, 1983, pp. 106.

[126] Cook, C.J., "Oxytocin and prolactin suppress cortisol responses to acute stress in both lactating and non-lactating sheep", Journal of Diary Research, Vol. 64, pp. 327-339, 1997.

[127] Altmann, J., "Observational study of behavior: sampling methods", Behavior, Vol. 49, pp. 227-267, 1974.

[128] Data Description Inc., "DataDesk Statistics Guide, Version 6.0", Ithaca, NY 14852, 1997, pp. 26/1-26/5.

[129] Cook, C.J., Maasland, S.H., Devine, C.E., "Social behaviour in sheep relates to behaviour and neurotransmitter responses to nociceptive stimuli", Physiology and Behaviour, Vol. 60, pp. 741-751, 1996

[130] Danielson, T.J., Golsteyn, L.R., "Systemic clearance and demethylation of caffeine in sheep and cattle", Drug Metabolism and Disposition, Vol. 24, No. 10, pp. 1058-1061, 1996

[131] Wold, H., "Estimation of principal components and related models by iterative least squares", In: Krishnaiah, P.R., (Ed), "Multivariate Analysis", Academic Press, New York, 1966

[132] Kong, X., Thakor, N.V., "Adaptive Estimation of Latency Changes in Evoked Potentials", IEEE Transactions on Biomedical Engineering, Vol. 43, No. 2, pp. 189-197, 1996

[133] Crecraft, D.I., Gorham, D.A., Sparkes, J.J., "Electronics", Chapman and Hall, London, 1993.

[134] Berndt, D., "Maintenance-Free Batteries; Lead-Acid, Nickel-Cadmium, Nickel-Hydride: A Handbook of Battery Technology", John Wiley and Sons, New York, 1993.

[135] Rashid, M. H., "Power Electronics, Circuits, Devices, and Applications", 2nd Ed., Prentice Hall, 1993.

[136] Ott, H.W., "Noise reduction techniques in electronic systems", 2nd Ed., John Wiley and Sons, New York, 1988.

[137] Analog Devices Inc., "1993 System Applications Guide", Analog Devices Inc., Norwood MA, 1993, pp. 13-2, 13-33.

[138] Horticulture and Food Research Institute of New Zealand Ltd., "FRPM Users Manual", Hamilton, New Zealand, 1998.