1991

INVESTIGATION, DEVELOPMENT AND APPLICATION OF KNOWLEDGE BASED DIGITAL SIGNAL PROCESSING METHODS FOR ENHANCING HUMAN EEGsJ

Hellyar, Mark Tremaine

http://hdl.handle.net/10026.1/1820

INVESTIGATION, DEVELOPMENT AND APPLICATION
OF KNOWLEDGE BASED
DIGITAL SIGNAL PROCESSING METHODS
FOR ENHANCING HUMAN EEGs.

Mark Tremaine Hellyar BSc(Hons), AMIEE.

July 1991

LIBRARY STORE

DEDICATION.

I dedicate this work to the memory of my father, Thomas George Cory Tremaine Hellyar. Without the inheritance of my fathers qualities of determination and simple stubbornness, together with the support and encouragement received from a small number of close friends, this work would not have been completed.

DECLARATION.

I hereby declare that whilst registered as a candidate for the degree of Doctor of Philosophy with the Council for National Academic Awards (CNAA) I have not been a registered candidate or enrolled student for another award of the CNAA or other academic or professional institution.

INVESTIGATION, DEVELOPMENT AND
APPLICATION
OF KNOWLEDGE BASED
DIGITAL SIGNAL PROCESSING METHODS
FOR ENHANCING HUMAN EEGs

Mark Tremaine Hellyar BSc(Hons), AMIEE.

ABSTRACT

This thesis details the development of new and reliable techniques for enhancing the human Electroencephalogram (EEG). This development has involved the incorporation of adaptive signal processing (ASP) techniques, within an artificial intelligence (AI) paradigm, more closely matching the implicit signal analysis capabilities of the EEG expert.

The need for EEG enhancement, by removal of ocular artefact (OA) , is widely recognised. However, conventional ASP techniques for OA removal fail to differentiate between OAs and some abnormal cerebral waveforms, such as frontal slow waves. OA removal often results in the corruption of these diagnostically important cerebral waveforms. However, the experienced EEG expert is often able to differentiate between OA and abnormal slow waveforms, and between different types of OA. This EEG expert knowledge is integrated with selectable adaptive filters in an intelligent OA removal system (IOARS). The EEG is enhanced by only removing OA when OA is identified, and by applying the OA removal algorithm pre-set for the specific OA type.

Extensive EEG data acquisition has provided a database of abnormal EEG recordings from over 50 patients, exhibiting a variety of cerebral abnormalities. Structured knowledge elicitation has provided over 60 production rules for OA identification in the presence of abnormal frontal slow waveforms, and for distinguishing between OA types.

The IOARS was implemented on personal computer (PC) based hardware in PROLOG and C software languages. 2-second, 18-channel, EEG signal segments are subjected to digital signal processing, to extract salient features from time, frequency, and contextual domains. OA is identified using a forward/backward hybrid inference engine, with uncertainty management, using the elicited expert rules and extracted signal features.

Evaluation of the system has been carried out using both normal and abnormal patient EEGs, and this shows a high agreement (82.7%) in OA identification between the IOARS and an EEG expert. This novel development provides a significant improvement in OA removal, and EEG signal enhancement, and will allow more reliable automated EEG analysis.

The investigation detailed in this thesis has led to 4 papers, including one in a special proceedings of the IEE, and been subject to several review articles.

ACKNOWLEDGEMENTS.

Contents.   Page

LIST OF ABBREVIATIONS.

ACF	Auto-correlation function.
ADC	Analogue to digital converter.
AI	Artificial intelligence.
ASP	Adaptive signal processing.
BSD	Bilaterally synchronous delta.
CCF	Cross-correlation function.
DAS	Data acquisition system.
DSP	Digital signal processing.
EEG	Electroencephalogram.
EOG	Electro-oculogram.
ES	Expert system.
FFT	Fast Fourier transform.
FIR	Finite impulse response.
HEM	Horizontal eye movement.
IA	Instrumentation amplifier.
IOARS	Intelligent ocular artefact removal system.
MPP	Modus ponendo ponens.
MTT	Modus tolendo tolens
MUX	Multiplexer.
NN	Neural network.
OA	Ocular artefact.
OLS	Ordinary least squares.
PCB	Printed circuit board.
PSD	Power spectral density.
PTM	Programmable timer module.
RAA	Ruductio ad absurdum.
RAM	Random access memory.
REM	Rolling eye movement.
RLS	Recursive least squares.
ROM	Read only memory.
VEM	Vertical eye movement.

LIST OF APPENDICES

PURPOSE OF THE INVESTIGATION.

--

. To investigate, develop, and implement new and reliable methods of removing ocular artefact (OA) from the electroencephalogram (EEG). Adaptive signal processing will be guided by a database of rules and heuristics commonly used by the EEG experts, for the identification of OA, and the differentiation between OA and abnormal slow cerebral waveforms. The rules will operate on characteristic features extracted from the EEG and electro-oculogram (EOG), using digital signal processing. The integration of Expert system (ES) techniques with conventional adaptive signal processing will more closely match the implicit signal analysis capabilities of the EEG expert.

.

. To provide the clinician with an intelligent, adaptive signal processing, tool capable of removing OA from the EEG whilst maintaining signals of diagnostic importance. This will lead to more reliable diagnosis of neurophysiological disorders and evaluation of treatments, and enable more accurate automated EEG analysis.

CHAPTER 1

INTRODUCTION

1.1 THE ORIGIN AND MEASUREMENT OF THE EEG

The Electroencephalogram (EEG) is a graphical representation of the electrical activity of the brain. The EEG is a widely used and relatively inexpensive tool for the diagnosis of neurological disorders and the evaluation of subsequent treatments. These include brain abnormalities such as those related to tumours, epilepsy, brain injury, Parkinson's disease, Huntin 's Chorea and schizophrenia [Ktonas, 1988].

The tiny electrical charges, produced by neuronal activity within the brain *[Creutzfeldt, 1974]*, are transmitted by cortical and skull tissue to the scalp surface, where they appear as voltages in the microvolt range. Electrodes, placed over the surface of the scalp, are used to record the voltage changes, using an international electrode placement method (The 10/20 system *[Jasper, 1958]*).

Each electrode measures the gross electrical activity from a vast number of underlying cortical neurons, having been modified by the transmission path. The observed effect is one of a fluctuating voltage containing various intermittent rhythms in the range of zero to thirty Hertz, and sudden sharp discharges with voltages in the millivolt range. Figure 1.1 illustrates the placement of electrodes, using the 10/20 system.
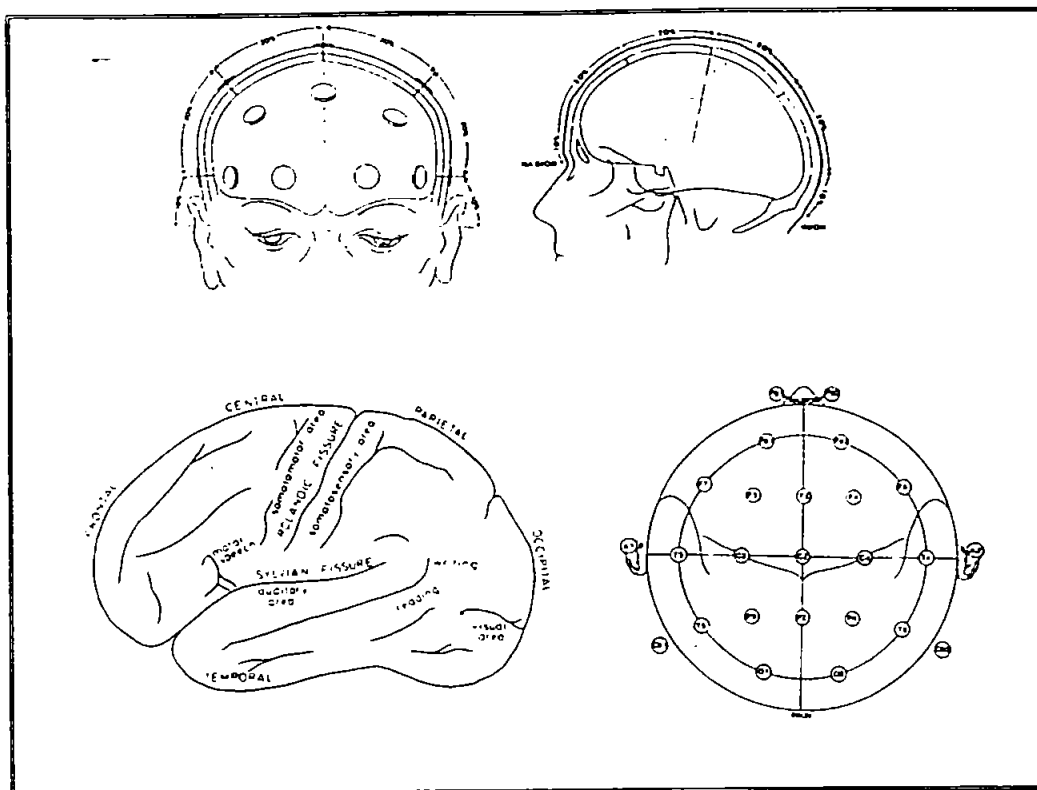
Figure 1.1 The 10/20 system of electrode localisation *[Jasper, 1958]*.

Silver silver chloride electrodes are commonly used for scalp recording electrodes because of their low impedance, low noise, and slow drift. Further improvement is made to variations in impedance caused by changes in skin electrode contact, i.e. from movement, by the use of a saline jelly acting as an interface between skin and electrode.

EEG measurement is either, between two electrodes (bipolar derivation), or between one electrode and a common reference point (referential derivation). The reference point must be as distant as possible, from both the EEG source and also from other electrical sources such as heart and muscles, to avoid the introduction of artefact. Linked ear lobes are a popular choice for this reference. Bipolar and referential recording EEG measurement is illustrated in figures 1.2 and

1.3 respectively

(a) Serial bipolar linkage. e1..5 are electrode potentials, v1..4 are channel input voltages.

(b) Potential distribution along a serial bipolar linkage using e and v as above.
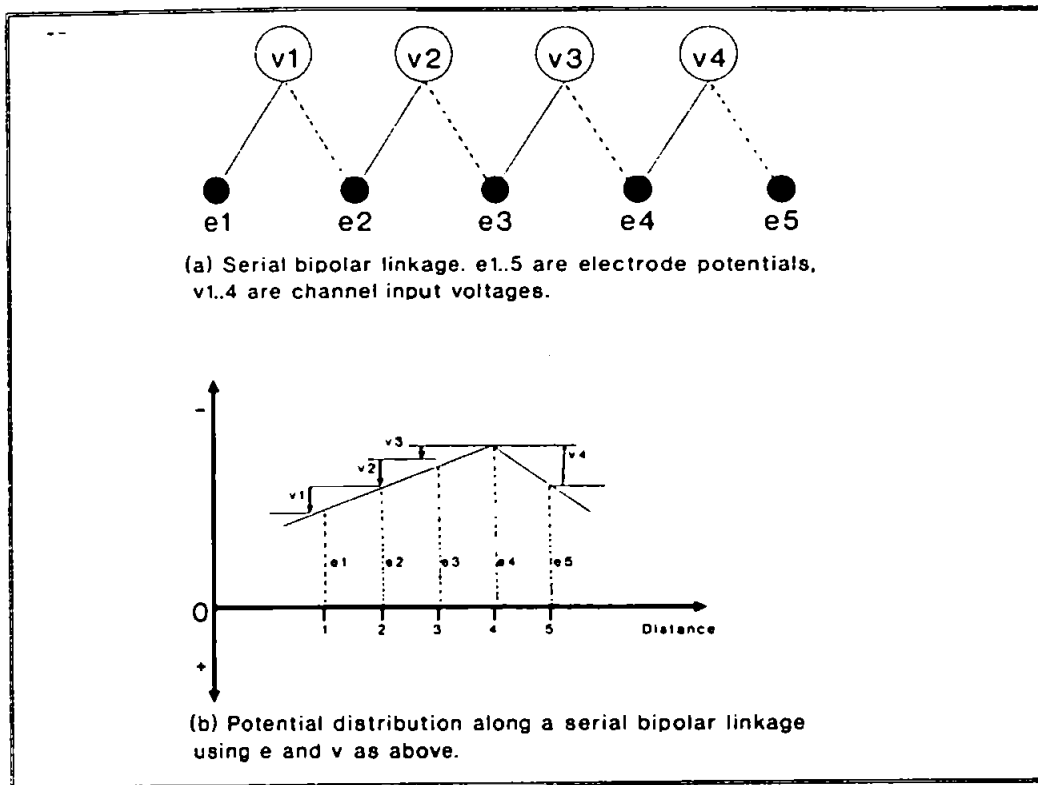
Figure 1.2 Bipolar derivation recording.

(a) Common reference derivation. e1..3 electrode potentials, v1..3 channel input voltages, er reference electrode.

(b) Potential distribution using common reference derivation using e and v as above.
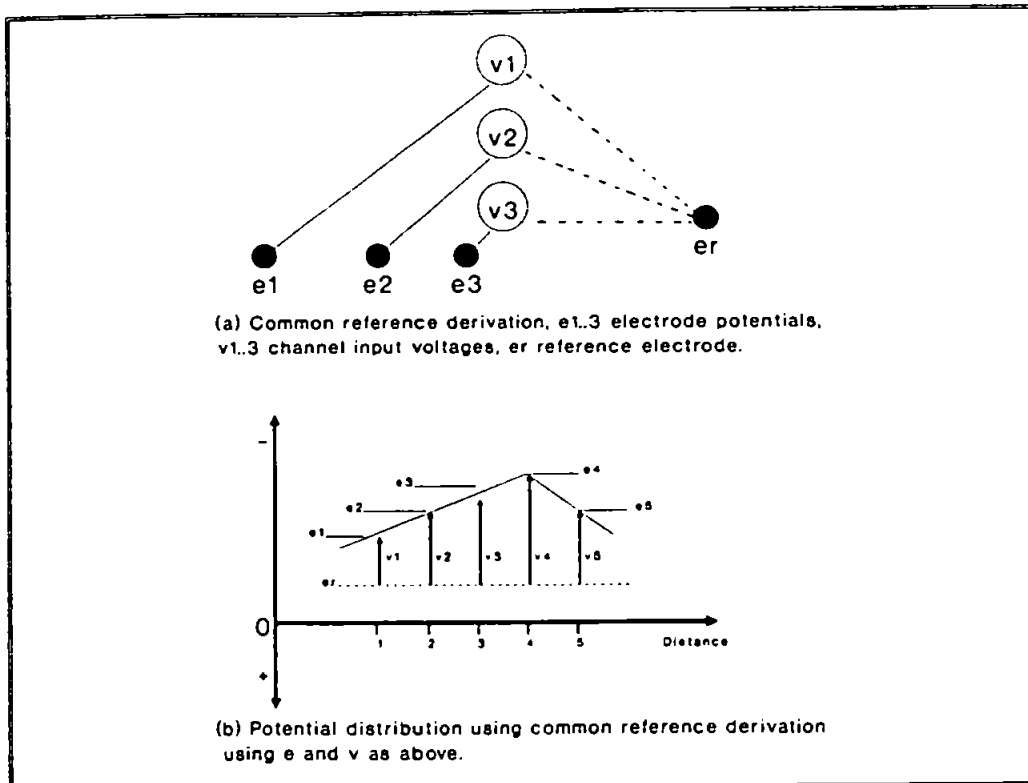
Figure 1.3 Common reference derivation recording.

A collection of suitably wired electrodes is referred to as a montage and each montage will contain as many as 21 channels, or electrode pairs. Each montage measures the EEG over a range of cortical areas (see figure 1.1). During recording of the EEG, several such montages will be used, enabling the clinician to better localise waveform sources such as the likely site of a tumour.

1.1.1 EEG SIGNAL PATTERNS.

The EEG has been previously described as containing a number of intermittent rhythms and isolated waveforms. For quantitative analysis EEG rhythms are divided into 4 main frequency bands: Delta (0.5 - 4Hz), Theta (4 - 8Hz), Alpha (8 - 13Hz) and Beta (13 - 30Hz). Figure 1.4 illustrates examples of EEG rhythms with frequencies in these bands.
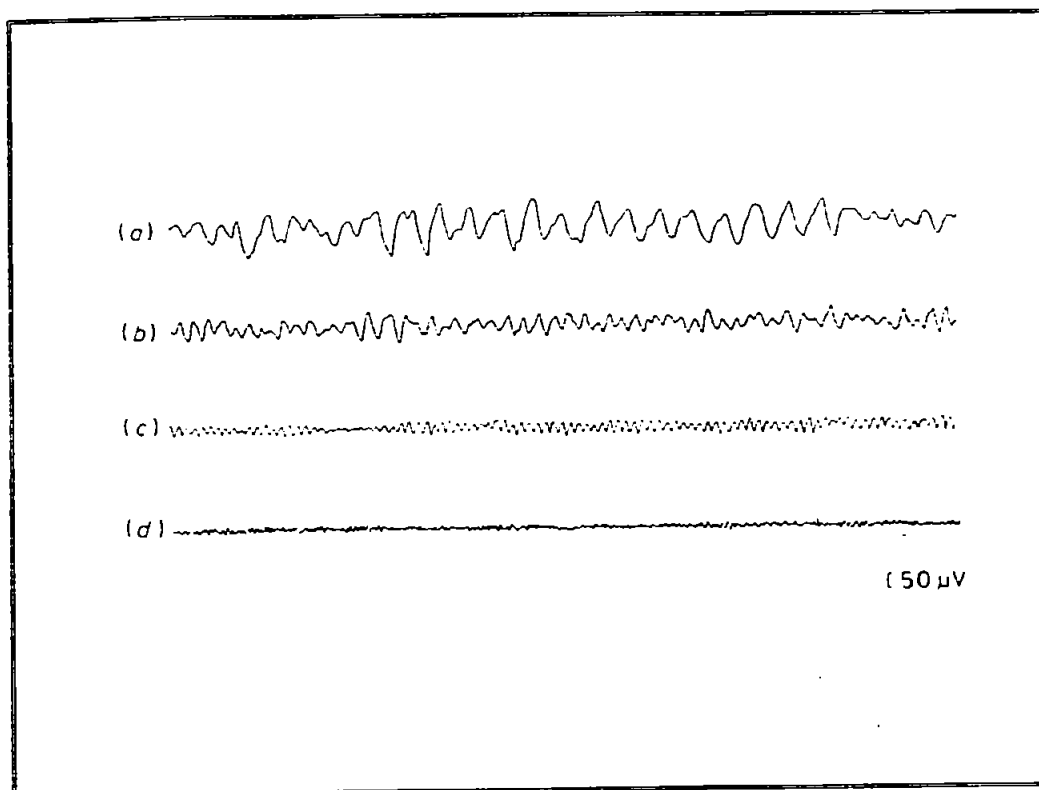
Figure 1.4 Example of rhythms from the standard EEG frequency bands. (a) delta, (b) theta, (c) alpha, and, (d) beta (from Cooper, et al.,1980).

The respective amplitudes of EEG signals in these bands is observed to be inversely proportional to their frequency *[Remond, 1977]*. The more common isolated waveforms are described by Hess, 1966. Figure 1.5 subdivides isolated waveforms into normal and pathological patterns.

The definitions of normal EEG frequencies are a complex function, related to age of patient, state of arousal, and region of the scalp or brain from which the EEG was obtained. In general, the EEG in infancy and early childhood is slower, higher in amplitude and shows less regional variation than of older children and adults. Maturation of the EEG occurs at an age of approximately 14, when the predominant 4-7 Hz (theta) measured in early childhood from all scalp regions, changes to a strong 8-12 Hz rhythmical waveform. This waveform is strongest in the posterior regions of the scalp - the alpha rhythm. The theta rhythm will remain but will gradually reduce to less than 25% of the total waveforms present by the age of 20, and become abnormal beyond the age of 30.

Cerebral neurons alter their activity when subjected to illness or injury. This is most often reflected in the scalp EEG by:

. Slowing and decrease in amplitude of the EEG.

. Increase in the EEG frequency.

. Presence of sudden EEG discharges (paroxysmal activity) differing from the background EEG either in frequency content or amplitude or pattern.

[Ktonas, 1988]

NORMAL PATTERNS:
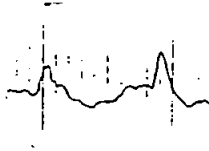
Lambda waves: < 50uV.
Positive   monophasic   transients   occurring   in   the
occipital region.

Sleep spindles: < 50uV.
Episodic   rhythm   at   about   14   Hz,   occurring   in   the
frontocentral location during certain stages of sleep.
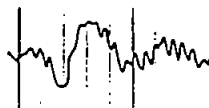
K-Complex: @200uV.
A   complex   pattern   consisting   of   one   or   several   slow
waves,   with   superimposed   faster   activity.   Maximum   at   the
vertex.

Vertex wave: < 300uV.
Negative sharp potential, maximum at the vertex.

PATHOLOGICAL PATTERNS:

Spikes:
A   clearly   distinguished   waveform   lasting   between   20-70
mS,   with   variable   amplitude   and   initial   negative
potential.
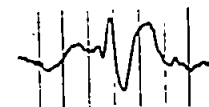
Sharp wave:
Similar   to   spike   but   of   longer   duration,   70-200   mS.
Spikes   and   sharp   waves   can   also   be   polyphasic,   i.e.
several negative and positive alternate deflections.
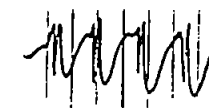
Spikes and waves: < 1000uV
Surface   negative   slow   waves,   usually   with   a   frequency   of
2.5-3.5 Hz having a spike associated with each wave.
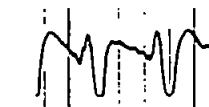
Slow spikes and waves:
Similar to spike and wave but with a frequency < 3Hz.
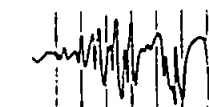
Multiple spikes and waves:
Bursts   of   spikes   followed   by   slow   waves,   occurring   as   a
single complex or repetitively.

Figure 1.5 examples of isolated EEG patterns.

Very slow activity in the delta EEG frequency band is considered
abnormal in any ~~woking~~ adult EEG. A slowing of the EEG cannot be attributable to
specific abnormalities and the same EEG slow activity may be produced by
encephalitis, head trauma, cerebrovascular disease or tumours *[Gibbs,
1951]*. Abnormal slowing of the EEG may be either focal or diffuse. Focal
slowing is indicative of an underlying cerebral disturbance which is focal
in nature, i.e. infarcts, tumours, contusions, abscess, or haemorrhage.
This may be spatially localised by observing the channel with either
maximum negativity in a common reference recording, or phase reversal in a
bipolar recording. Diffuse slowing is indicative of a more widespread
disturbance, i.e. inflammation, trauma, or metabolic problems. In general,
the more acute the underlying disturbance is, the slower and higher in
amplitude the abnormal waveforms will be.

1.2 ARTEFACTS AND THEIR ASSOCIATED PROBLEMS.

The measurement of the EEG is susceptible to a large number of signal
contaminations, or artefacts from a number of sources internal and
external to the body:-

. Internal sources include the electrical activity produced by head
and body movement, muscle and cardiac activity as well as extra-cranial
generators, such as tongue and eye movement.

. External sources of artefact commonly include mains interference and bad electrode contact.

Artefacts are defined as those signals present in the measured EEG signal that are not of cerebral origin *[Cooper, 1980]*. The large amplitude of these artefacts often obscures the true cerebral signals and the similarity of some artefacts to cerebral signals of interest can make interpretation of an EEG record considerably more difficult for even the EEG expert.

Figure 1.6 illustrates this problem by presenting two examples of EEG artefacts. Figure 1.6(a) is a sample of a conventional EEG trace showing contamination by cardiac activity which takes the appearance of abnormal spike patterns typical of epileptic activity. Figure 1.6(b) is a sample of a topographic EEG obtained from a brain mapper (Neuroscience Ltd.). Individual diagrams show the potential at electrode sites, in the delta frequency band and sampled at 0.5 second intervals. The upper trace clearly shows a focus of delta activity in the right parietal region of the scalp caused by poor electrode contact. Care would need to be taken by the clinician not to interpret this as indicative of tumour.

Improved electrical screening and correct electrode attachment techniques can help to eliminate the external sources of artefact *[Hector, 1980]*. However, the internal sources of artefact present a constant problem to the clinician, the greatest of these being caused by eye movement or ocular artefacts (OAs).
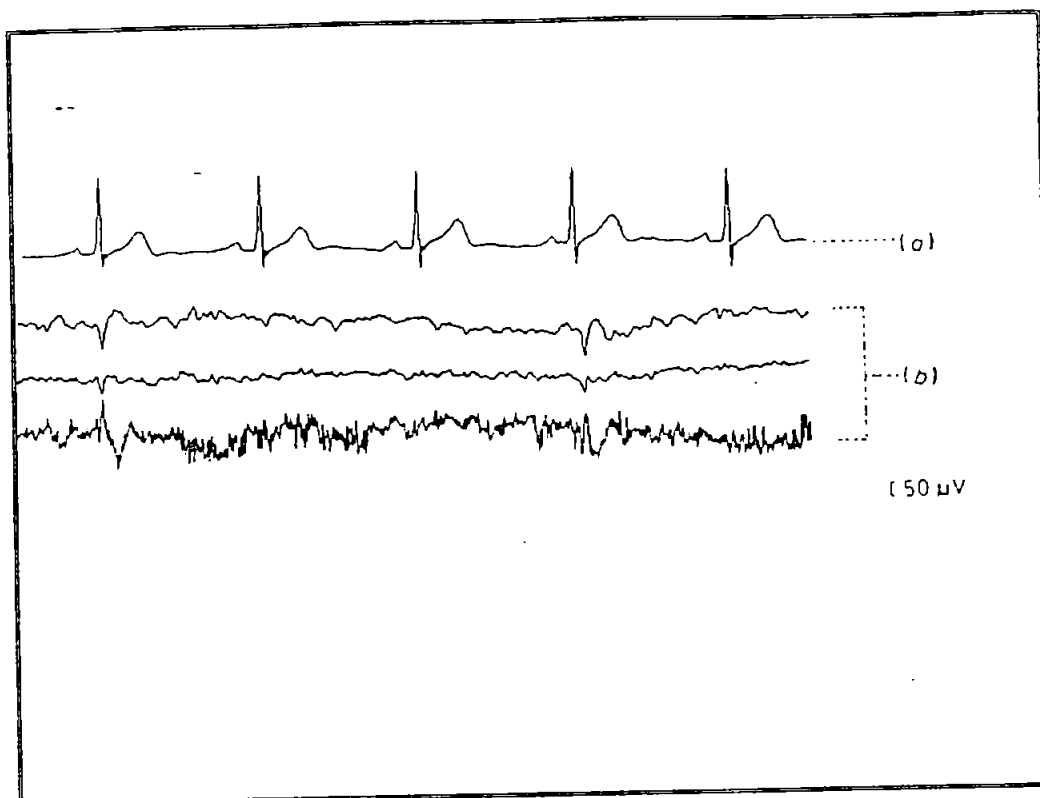
Figure 1.6 (a) ECG artefact. (a) ECG, (b) contaminated EEG.
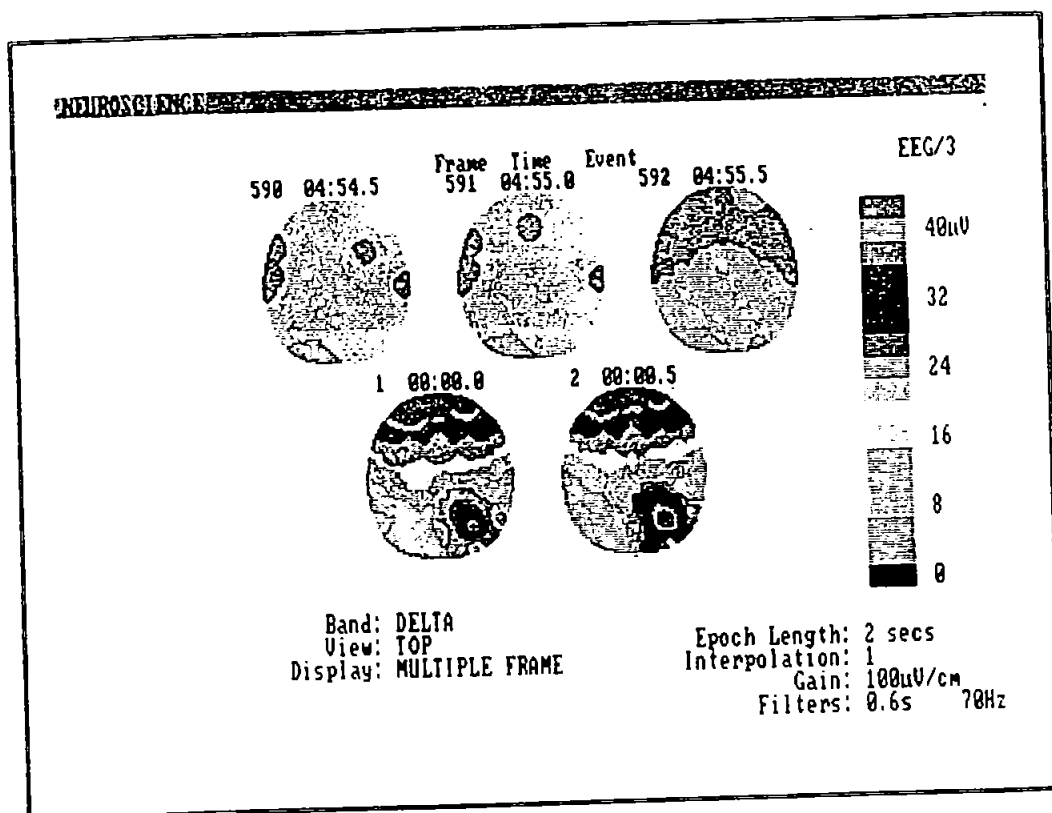
Figure 1.6 (b) Electrode artefact.
page 9

1.2.1 OCULAR ARTEFACT (OA).

Associated with the eye is a corneo-retinal dipole potential of about 100mV *[Young and Sheena, 1975]*. The relative movements of the eye and/or the eyelid *[Jervis et al., 1988]* cause a change in electrostatic field of the corneo-retinal dipole. This change in the potential field propagates over the entire scalp, being strongest in the frontal regions of the scalp. As a consequence, a signal measured at any point on the scalp will be a function of both genuine cerebral signals and OA.

Various techniques have been studied for recording eye movement, including optical methods which use light and a lens system, the impedance oculogram *[Sullivan and Weltman, 1963]* which measures changes in electrical impedance across the eyes with movement, and the electrooculogram (EOG) *[Shackel, 1967]* which directly measures the changing electrostatic field. These are reviewed extensively in Young and Sheena, 1975. All methods produce an electrical signal which is a function of either eye position, or eye and/or eyelid movement but employ different means to obtain this. The most commonly used indication of eye movement is that of the EOG which uses electrodes placed around the eyes to record the potential field in a similar manner to that of EEG measurement. The EOG method has several clear advantages over its alternatives:-

. It can be used with open or closed eyes.

. The voltage measured is directly proportional to the degree of eye movement and proportional to the OA measured at any point on the scalp.

.　　The method is simple and requires a minimum of calibration.

.　__　Standard EEG recording techniques and equipment can be used.

The number of electrodes used, and the positions for these electrodes for best measurement of the EOG is a matter for debate and several EOG montages have been exposed to rigorous analysis *[Ifeachor, et al., 1988; Jervis, et al., 1988; Brunia, et al., 1989]*. Figure 1.7 illustrates the common electrode positions used for EOG recording.
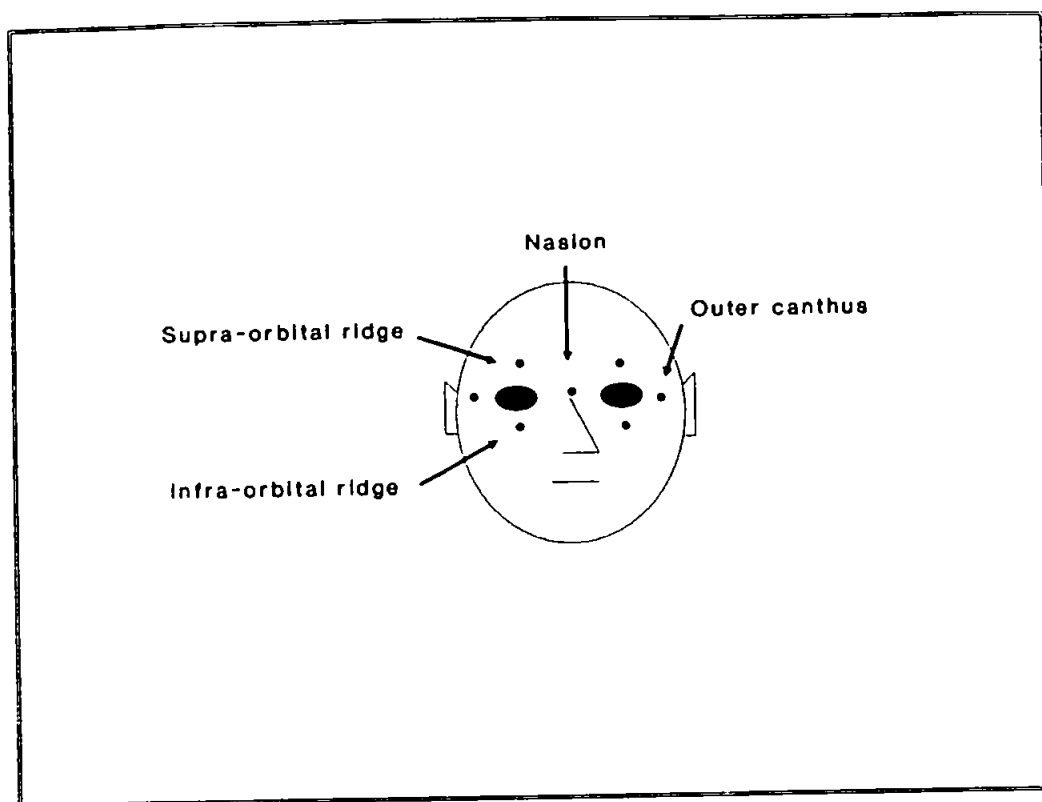
Figure 1.7 Measurement of eye movement: electrode positions.

Eye movements can be classified into 8 distinct types *[Shackell, 1967]*:---

- Saccadic.
- Smooth: pursuit and compensatory.
- Nystagmoid.
- Torsional.
- Vergent.
- Miniature.
- Blinking.
- Intra-ocular (lens, pupil).

The first three types are the common, larger movements, and the next three are equally common, smaller movements. Of the specified types in the above list, the first two comprise conjugate movements, that is both eyeballs move together in the same direction. The third and fourth are usually conjugate, while the fifth is not, the movements being in opposite directions. Miniature movements refer to the very small movements that occur continuously. These include drifts and flicks of the two eyeballs (sometimes referred to as physiological nystagmus), and oscillatory tremor *[Shackell, 1967]*.

Figure 1.8 illustrates a number of typical OAs by comparing the measured EOG and EEG respectively. EOG signals were recorded bipolarly between electrodes placed on the supra orbital ridge and the outer canthus for both left and right eyes. EEG signals were recorded referentially,

100μv

1 second.

(a)
(b)
(c)
(d)
(e)
(9)
(h)
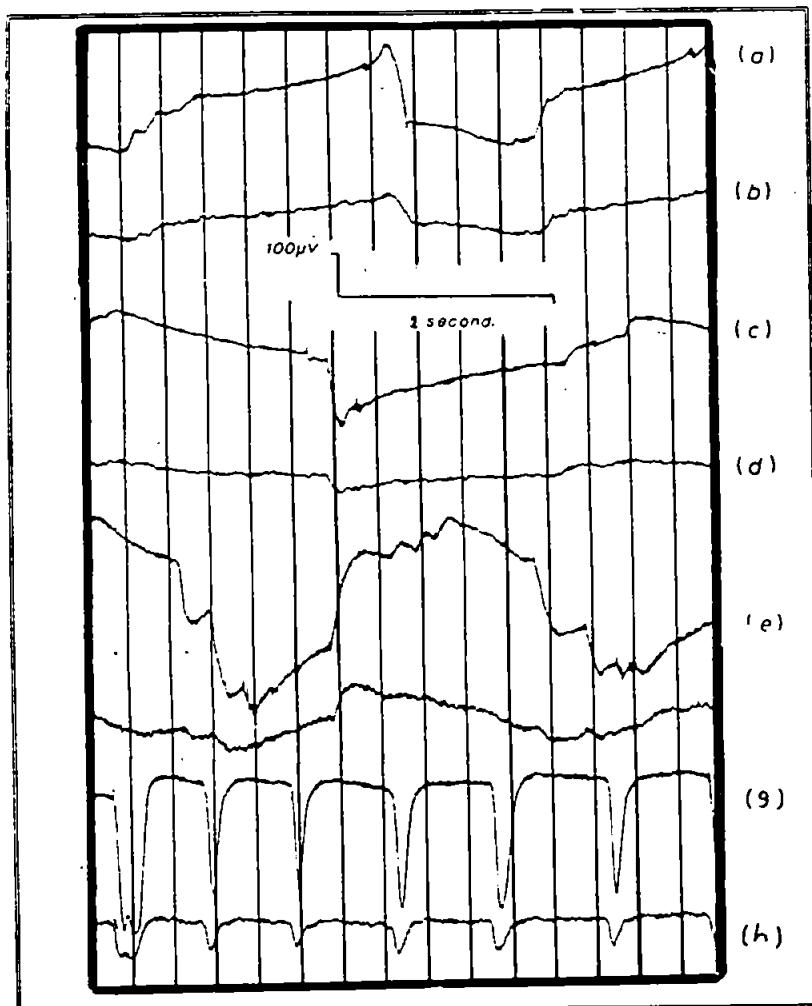
(a) EOG with VEM, (b) EEG with VEM OA.
(c) EOG with HEM, (d) EEG with HEM OA.
(e) EOG with REM, (f) EEG with REM OA.
(g) EOG with Blink, (h) EEG with Blink artefact.

Figure 1.8 Typical eye movements and their effect on the vertex EEG.

page 13

with respect to linked ear lobes, at the scalp vertex. The OAs illustrated are caused by vertical eye movement (VEM), horizontal eye movement (HEM), rolling eye movement (REM), and blink respectively. The extent of eye movement in each case, barring blink, was that of 30 degrees from the rest position. Figure 1.8 is seen to include examples of the larger saccadic movements for VEM and HEM, smooth movement for REM and finally blinking.

1.2.2 THE NEED FOR OA REMOVAL.

The large magnitude of OAs (upto several mV) can often obscure the smaller cerebral signals. Further difficulties arise because of the similarities between OAs and several of the EEG patterns previously described. For example, in the diagnosis of brain damage in babies, genuine abnormal slow waves cannot easily be distinguished from ocular artefact which is produced by moving or rolling of the eyes. Similarly, frontal slow waves caused by OA are often difficult to distinguish from abnormal frontal slow waves caused by a frontal tumour. Figure 1.9 shows a portion of an EEG record containing an abnormal waveform of large amplitude and a frequency of approximately 2Hz, termed bilaterally synchronous delta (BSD) and caused, in this case, by a gross cortical atrophy or loss of brain surface material. The waveform can be seen to be present simultaneously in all channels and will therefore be detected at both EEG and EOG electrodes making determination of origin even harder *[Hellyar, et al., (in preparation)]*.
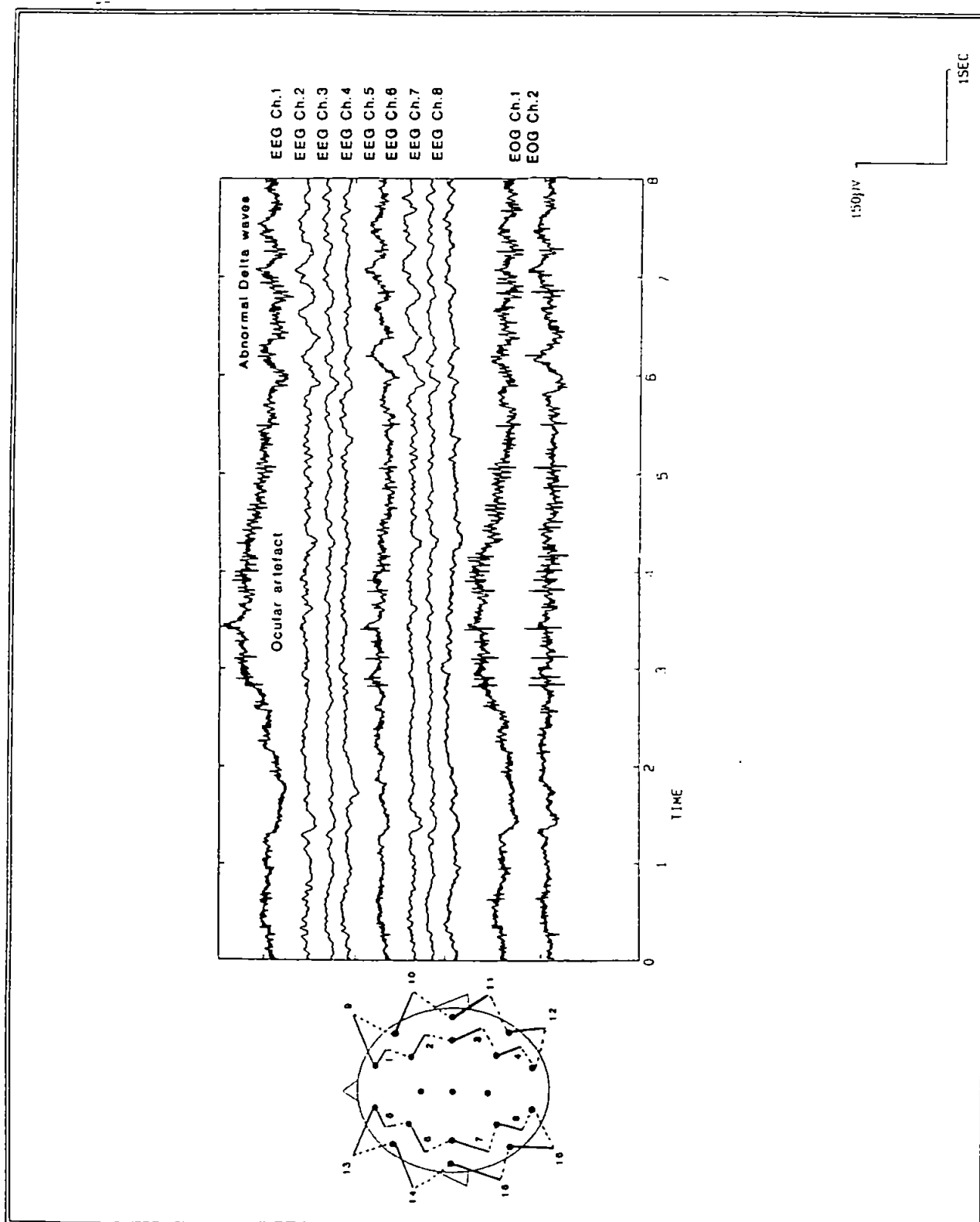
Figure 1.9 Genuine abnormal EEG slow waves.

An attempt to restrain eye movement, by request during an EEG recording, is often made by the recording staff. This approach is however unsuitable for uncooperative patients, patients with illnesses displaying uncontrollable eye movements (e.g. Huntingtons chorea), and small children. In these cases the extent of OA contamination of the EEG can, at best seriously reduce the usefulness of any EEG and at worst render the EEG useless *[Jervis, et al., 1985]*. This is also true for EEG event related potentials *[Low et al., 1966]* in which EEG signals are measured between two auditory or visual stimuli. In addition to the normal involuntary eye movements and blinks it has been found that the stimuli used to elicit the evoked response can often cause an eye movement or blink.

The value of automated EEG analysis using computer systems is becoming more widely recognised because of the potential increase in efficiency offered to the busy clinician and the improvement to EEG signal presentation and patient data archiving. Much of the clinicians expensive time is devoted to routine EEG analysis. A reliable automated analysis system would free this time for more demanding and specialised work. However, automation of the EEG signal analysis procedure, though desirable for reasons of efficiency and diagnostic reliability, is severely hampered by the lack of effective ways to deal with unwanted EEG signal contaminations, or artefacts, and this is particularly true of OA. A number of systems have been investigated for computerised EEG analysis (CEAN) *[Barlow, 1977; Cooper, 1980; Kronas, 1988]* which quantitatively

page 16

analyse a number of EEG signals to obtain either a spectral description of the background rhythm, or to identify certain paroxysmal events, such as spikes, the purpose of which is to indicate to the clinician possible areas of abnormal EEG activity. Ktonas describes a number of schemes for automated analysis of epileptic EEG, typified by paroxysmal waveforms. In all cases, false detections of true epileptic activity due to blink type artefact seriously degraded results. See also *[Gorman and Gloor, 1976]*.

It is therefore concluded that proper ocular artefact recognition and removal is necessary and will make the interpretation of EEG easier and allow more reliable automated EEG analysis.

1.3 OA REMOVAL PROCEDURES.

The need for OA removal has been clearly identified in the previous section as well as indicating some of the problems that are met when attempting to do so. These include the significant spectral overlap of EEG and OA and the similarity of many individual waveform patterns. OA removal has been investigated by a number of authors *[Jervis, 1988]* and several schemes for the removal of OA from the EEG have been established which can be divided into three main categories:-

   .   Eye fixation *[Borda and Hablitz, 1973]*.

   .   Artefact rejection *[Gevins, et al., 1977]*.

   .   Electro-oculogram (EOG) subtraction *[Quilter et al., 1977]*.

Eye fixation is probably the simplest and least reliable scheme for OA removal. Patients are often encouraged to fixate on a point or image or to refrain from blinking at critical times during a recording [Papakostopoulos, et al., 1973]. As previously discussed this is impractical and as many as 50% of subjects are unable to successfully fixate [Borda and Hablitz, 1973].

Artefact rejection effectively ignores sections of EEG signals in which artefact is detected above some pre-defined threshold. The remaining EEG is then considered free of artefact and open to further analysis [Gotman et al., 1973; John et al., 1977; Verleger et al., 1982]. The limiting disadvantages of this scheme are: Firstly, the difficulty arising in selection of a suitable detection threshold, a simple amplitude threshold is clearly insufficient when discriminating between similar waveform patterns and inter patient variations. Secondly, this scheme provides a remaining EEG signal which is unrepresentative of the original EEG signal, having large amount of signal removed along with the corrupted sections. This is particularly relevant in uncooperative patients and those with uncontrollable eye movements.

EOG subtraction offers the most promising way of removing OA. The scheme is based on the principle that the OA is additive to the background EEG and is linearly related to the EOGs. Therefore a 'corrected' EEG (with OA removed) may be obtained at any electrode site by subtracting, from the measured EEG, a simple proportion of the EOGs. This procedure has been attempted in the time domain [Quilter et al., 1977], and the frequency

domain *[Gasser et al., 1986]* and EOG proportions can be calculated off-line, prior to recording *[McCallum and Walter, 1968; Girton and Kamiya, 1973]*, or on-line during recording *[Quilter et al., 1977; Jervis, et al., 1985; Ifeachor et al., 1986]*. The method described here is that of on-line, time domain EOG subtraction as this allows inter and intra-subject differences to be accommodated, and allows real-time EEG signal processing.

1.3.1 TIME DOMAIN ELECTRO-OCULOGRAM (EOG) SUBTRACTION.

The measured EEG at any point on the scalp, at time t, can be modelled as a linear combination of EEG and ocular artefact, and is represented by the signal y(t).

$$y(t) = oa(t) + eeg(t) \qquad 1.1$$

The ocular artefact potential is estimated using a proportion of EOG signal, called the OA coefficient (k).

$$y(t) = k.eog(t) + eeg(t) \qquad 1.2$$

Rewriting equation 1.2 in discrete form gives.

$$y(i) = k.eog(i) + eeg(i) \qquad 1.3$$
$$i = 1,2, ..., m$$

where i represents the sample number

and   m is the number of data samples

Where multiple EOG signals are used to estimate the OA contamination, equation 1.3 is more conveniently represented using vectors for the EOGs and OA coefficients.

$$y(i) = k_1 eog_1(i) + k_2 eog_2(i) + \dots k_n eog_n(i) + eeg(i)$$
$$i = 1, 2, \dots, m$$

$$y(i) = K \ EOG^T(i) + eeg(i) \qquad 1.4$$

where $EOG^T(i) = [eog_1(i), eog_2(i), \dots eog_n(i)]$

and $K = [k_1, k_2, \dots, k_n]^T$

and   n is the number of EOG signals

and ^{T} indicates transposition

n.b. eeg(i) represents an error term in this equation.

If K can be estimated, then an estimate of the true EEG, $\hat{eeg}(i)$, can be obtained from

$$\hat{eeg}(i) = y(i) - \hat{K} \ EOG^T(i) \quad i = 1, 2, \dots, m \qquad 1.5$$

where $\hat{K}$ are estimates of K

and $\hat{eeg}(i)$ is the estimate of eeg(i)

and   m is the number of samples used in the estimation.

K are estimated either off-line or on-line with respect to signal recording and several techniques have been utilised for this estimation *[Jervis, et al., 1988]*. The numerous estimation techniques differ mainly in the number of EOG signals that are used and the way these are measured.

Off-line methods allow more sophisticated removal techniques to be employed. For example, the Ordinary Least Squares (OLS) method (see appendix A) optimally estimates K by minimising the sum of squares of the error term, represented by $\hat{eeg}(i)$ (i=1...m) in equation 1.5 *[Fortgens and de Bruin, 1983]*. This method makes a number of inaccurate statistical assumptions, such as eeg(i) being uncorrelated noise with zero expectation *[Ifeachor, 1984]*, and involves time consuming inverse matrix calculation. The requirements of the clinical laboratory, costs, and trends towards real-time signal processing, dictate that artefact removal should be done on-line, that is, as data are being acquired. This has prompted the use of real-time removal of OA from the EEG and the use of iterative methods for estimation of the Ks

1.3.2 ADAPTIVE OA REMOVAL.

Figure 1.10 illustrates an adaptive on-line ocular artefact remover *[Ifeachor, et al., 1986]*. EEG and EOG signals are digitised and K are estimated using a numerically stable form of the Recursive Least Squares (RLS) algorithm (Appendix A). The RLS algorithm updates K at each sample point so that changes in ocular movements can be reflected in K. The RLS
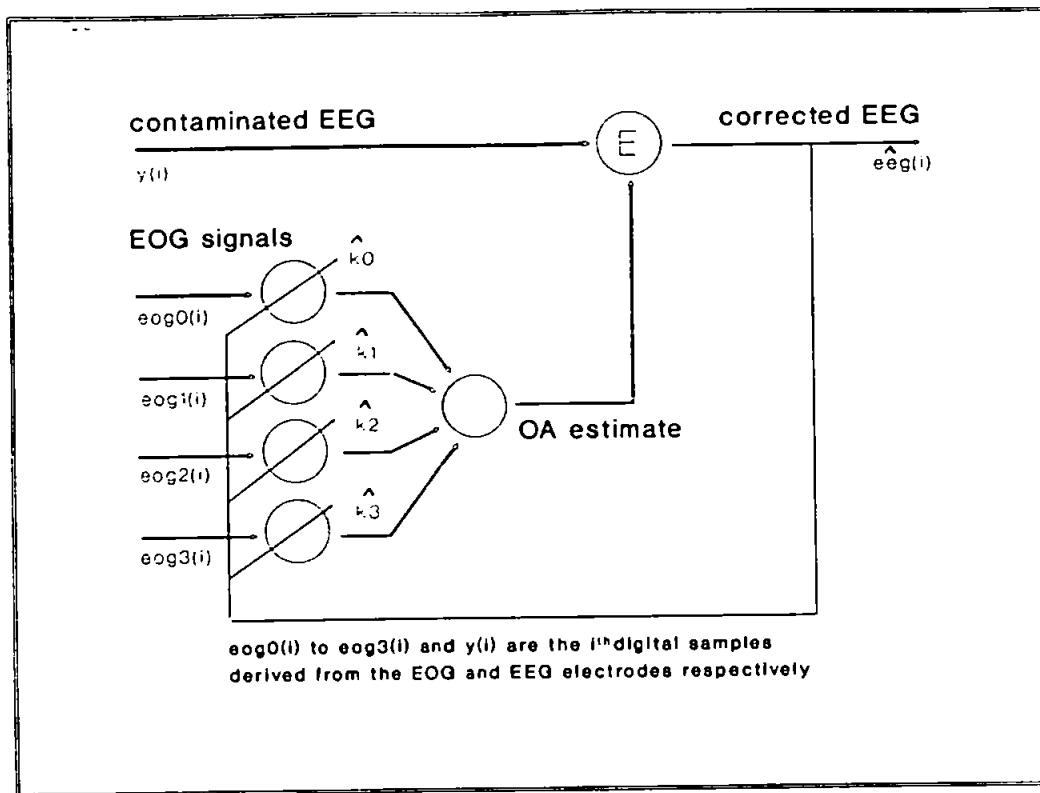
Figure 1.10 Adaptive OA remover.

algorithm is a recursive implementation of the OLS algorithm and overcomes the time consuming calculation of the inverse matrix by updating the K at each sample point according to the error in the previous estimate. The algorithm converges on an optimum value for K/Young, 1974; Haykin, 1986/. A suitable RLS algorithm is obtained by exponentially weighting past data to gradually remove its effect on the estimate.

The OA estimate is subtracted from the measured EEG to obtain the estimate of the true EEG. In figure 1.10, y(i) and eog0(i) to eog3(i) are the ith digital samples of the measured EEG and EOG signals respectively. Four EOG signals are used in this example. $\hat{k}(0)$ to $\hat{k}(3)$ are the ocular

artefact parameters and e$\hat{e}$g(ι) is the estimate of the true EEG. This then represents adaptive filtering of OA's from the EEG.

1.4 PROBLEMS ENCOUNTERED WITH OA REMOVAL.

The on-line adaptive OA removal technique of section 1.3.2 has been subjected to considerable experimentation *[Hellyar, et al., 1991(in press); Ifeachor, et al., 1988]*. This has revealed a number of areas where the performance of the removal algorithm is prone to significant errors. This section is intended to highlight these observed errors with a view to improving the OA removal algorithm.

1.4.1 INFRINGEMENT OF CORRELATION REQUIREMENTS.

The performance of the adaptive filter, (figure 1.10), relies on the EOG being highly correlated with the OA and weakly correlated with the EEG. This cannot be guaranteed in practice because of the wide range in EEG waveform patterns, and the lack of electrical isolation between EEG and EOG electrodes. The EEG may contain abnormal waveform patterns which resemble the OAs and/or the EOGs may contain signals of cerebral origin *[Gotman, et al., 1975; Muras and Binnie, 1970]*. Under these conditions, the correlation requirements of the OA removal algorithm are not satisfied and the adaptive filter cannot perform correctly. For example, it was found in previous work *[Jervis et al., 1988]* that when abnormal waveform patterns, such as slow waves or epileptic spike and wave

complexes, were detected at both the EEG and EOG electrodes, the corresponding waves in the EEG, after OA removal, were reduced in amplitude. The waveforms of figure 1.11 clearly illustrates this problem, showing the appearance of abnormal slow waveform patterns in both the EOG and EEG channels. Figure 1.11 illustrates the results from the use of the conventional OA removal algorithm on one signal from figure 1.9. The resultant estimate of the true EEG shows a significant attenuation and corruption of the clinically important slow waveform pattern. This is clearly an unsatisfactory result and represents a significant problem when it is considered that the abnormal EEG often appears as an increase in the slow waveform pattern activity (see section 1.1.1).
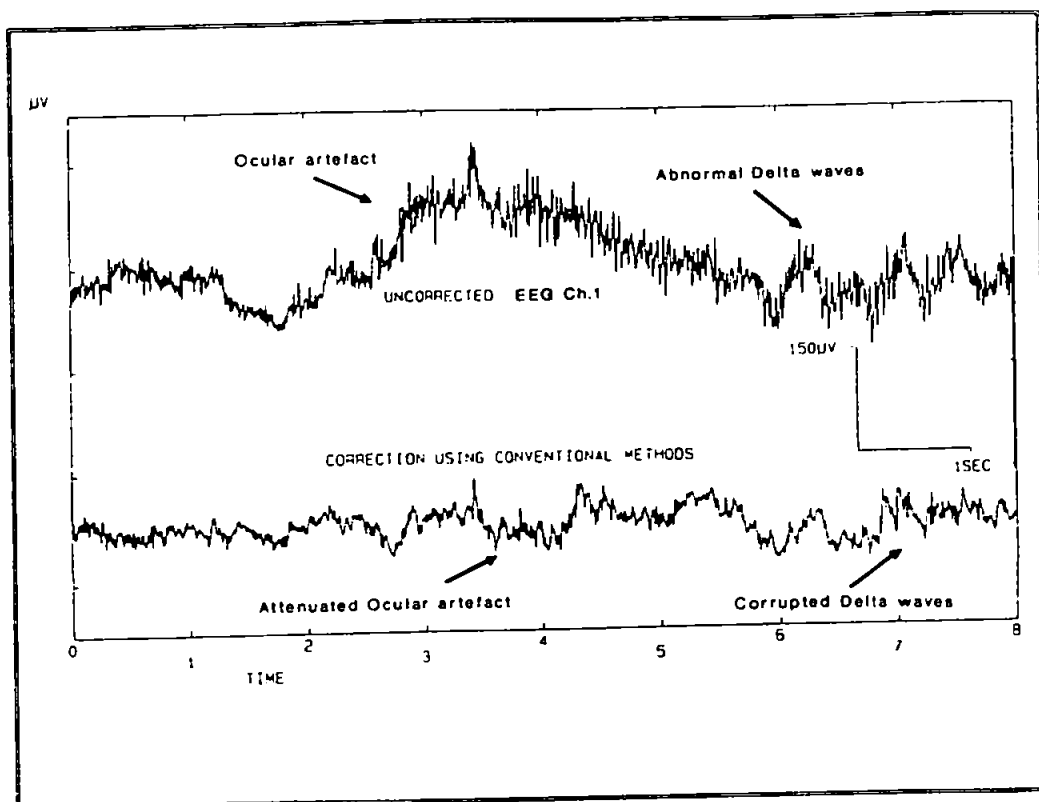
Figure 1.11 Conventional adaptive signal processing applied to the signals illustrated in figure 1.9.

1.4.2 NONSTATIONARITY OF THE EEG.

--

A typical EEG recording will show a variety of changing waveform patterns, an example of which would be the common waxing and waning of the alpha rhythm. The EEG is therefore regarded as being statistically nonstationary in that the statistics, such as mean and variance, of individual signals vary over time. This nonstationarity is also aggravated by the random occurrences of OAs and possible abnormal waveform patterns. Figure 1.12 illustrates a sample of EEG containing a VEM OA which changes significantly the underlying EEG signal statistics. This is clearly identified by the concurrent display of the respective OA parameter, and the estimate of the true EEG after OA removal. The OA parameter displays considerable disturbance at the onset of the OA and attached change in signal statistics, resulting in an estimate of the EEG which contains visible remnant OA.

The nonstationarity of the EEG creates an environment in which the operation of the conventional adaptive OA removal algorithm is prone to error. The object of the recursive algorithm, described above (see also appendix A), is to obtain the ocular artefact parameters K. The estimates of K are updated for each new sample of measured EEG and EOG and will, for
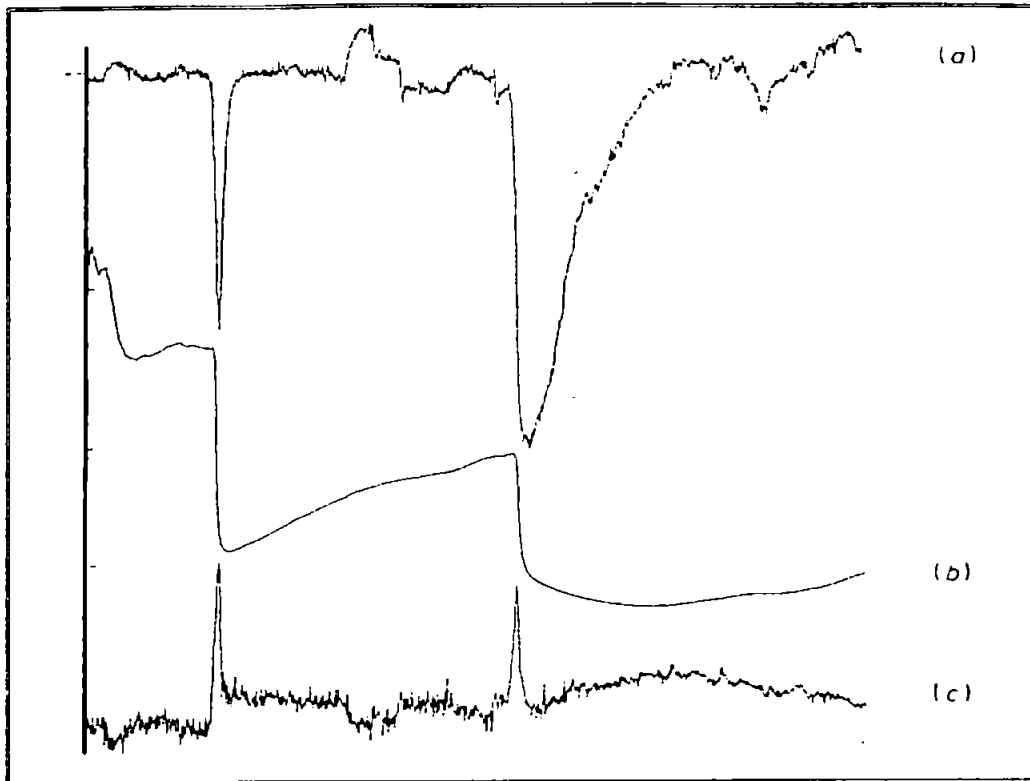
Figure 1.12 Random occurrences of large OA causes large variations in the

OA parameters (K), and significant error in the estimated true EEG.

(a) contaminated EEG, (b) K, (c) estimated true EEG.

a stationary signal, converge to an optimum set of parameters. However, for a nonstationary signal, the optimum set of parameters will be changing. Therefore, not only must the algorithm converge on an optimum set of parameters, but also track these parameters as they change. This can lead to significant lowering in performance unless the algorithm converges quickly [Ferrara and Widrow, 1981].

1.4.3 MULTIPLE ARTEFACTS.

The various occular movements defined in section 1.2.1 produce OAs in the EEG which have characteristic differences. These differences are seen to be attributable to the potential distribution and the waveform shape *[Overton and Shagass, 1969; Corby and Kopell, 1972]*. Figures 1.13 illustrate the potential distribution of three major OA waveforms: VEM, HEM, and blink.

The differences in potential distribution of the various types of OA will mean that each electrode position will be contaminated with different proportions of OA from the respective eyes. Therefore, each type of OA will require a different set of optimal OA parameters (K) for each electrode position. The simple resetting of the OA parameters, as discussed in section 1.4.2, will therefore, not provide adequate OA removal because of the need to converge on different OA parameters for each new OA type.

1.4.4 SECONDARY ARTEFACTS.

Signals occurring in the EOG which are not of eye movement origin, for example electrode noise or electrical activity from the orbital and temporal muscles, may be introduced into the estimated EEG as secondary artefacts. This is particularly relevant when there is little or no OA *[Quilter, et al., 1977]*. Figure 1.14 illustrates the introduction of secondary artefact into the estimated true EEG and is caused by an EOG

VEM potential spread

HEM potential spread

BLINK potential spread
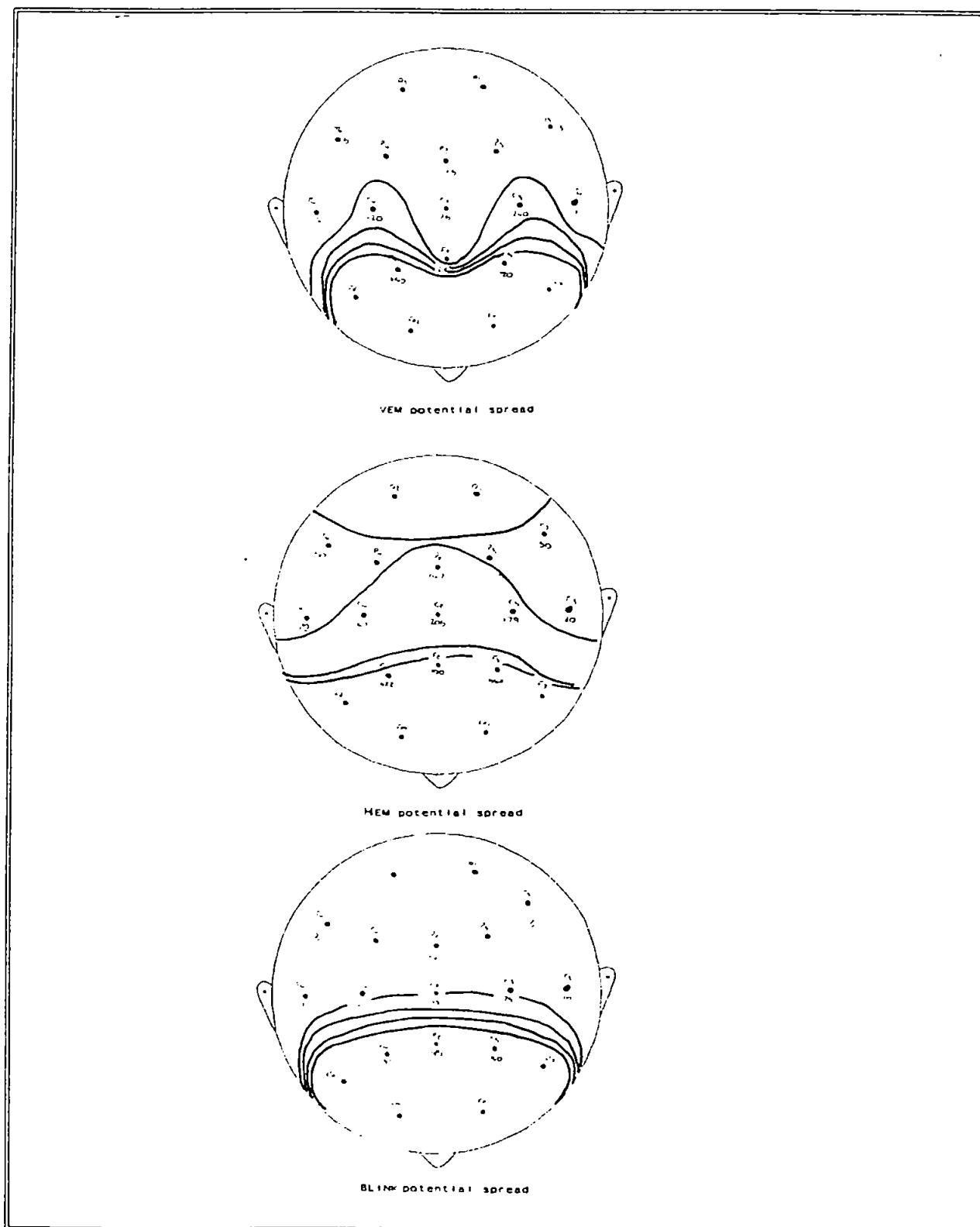
Figure 1.13 Potential distribution: Average of 30.
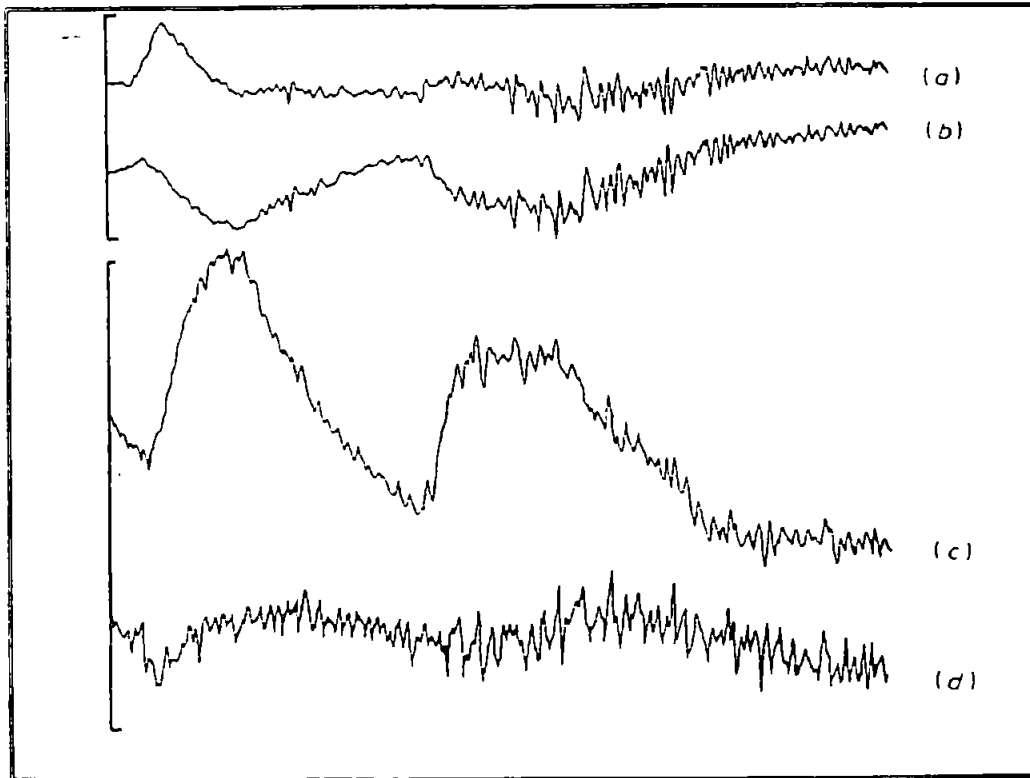
Figure 1.14 Secondary artefact contamination of the EEG.

(a) Fp2, (b) F8, (c) F8-T4, (d) estimated true EEG.

Note: EOG signals are on half vertical scaling.

signal which contains significant muscle potential. The estimate of the true EEG has attenuated OA but is corrupted by muscle activity not evident in the original signal.

1.4.5 OA REMOVAL ALGORITHM MEMORY.

The OA removal algorithm has a memory of previous input data (asymptotic sample length) that effects the present estimation, and represented by the formula:

$$\text{Memory(No of samples)} = \frac{1}{(1 - \text{ASL})}$$

This memory is weighted so as to gradually remove the effects of old data on the present estimate. An ASL of 0.998 represents a memory of 500 samples, or approximately 2 seconds, an ASL of 0.98 represents a memory of 50 samples, or 0.2 seconds. An ASL of 0.998 has been found empirically to provide an estimated EEG that does not fluctuate wildly with changes in input data [Ifeachor, 1984]. However, sudden nonstationarities, such as OA and in particular blink type OA, require the OA removal algorithm to respond quickly to the change in input signal, hence requiring shorter ASL when OA is detected. A longer algorithm memory will result in undercorrection of the OA, and a shorter algorithm memory will result in wild fluctuations in estimate, and therefore overcorrection of true EEG signal after the OA has ceased. Figure 1.15 illustrates this problem on a section of EEG containing two blink type OAs. Remnant OA is visable with ASL= 0.998 (b) and background EEG is attenuated with ASL= 0.98.
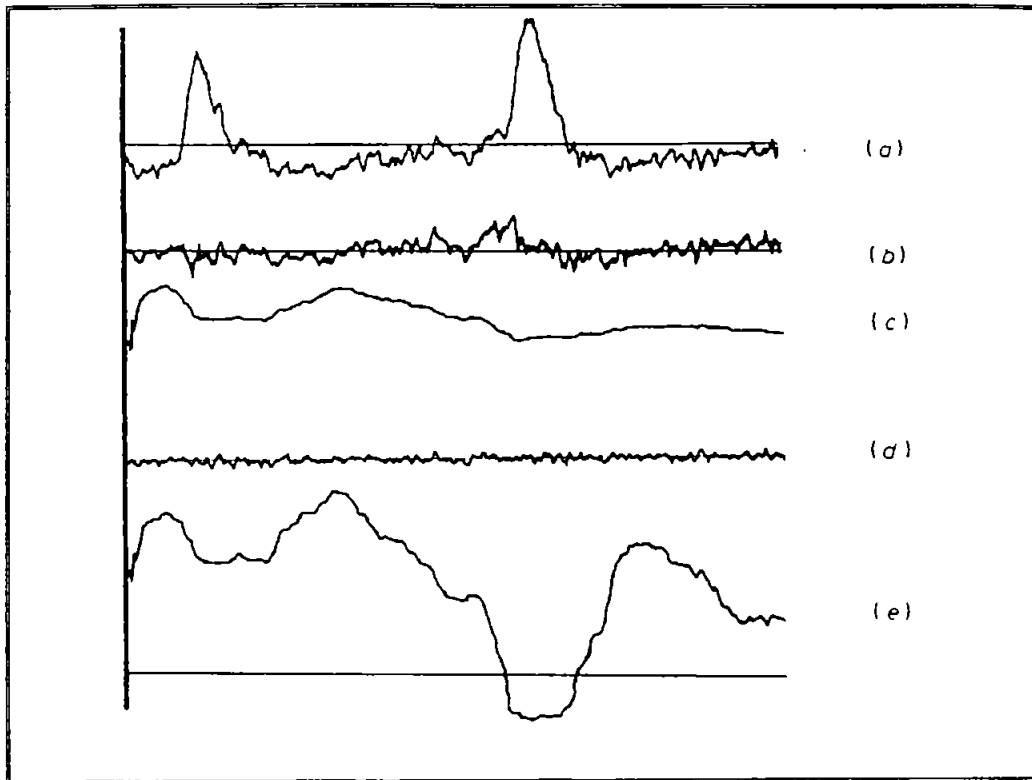
Figure 1.15 The effective memory of the OA removal algorithm

(a) corrupted EEG, (b) estimated EEG: ASL=0.998, (c) K: ASL= 0.998,

(d) estimated EEG: ASL= 0.98, (e) K: ASL=0.98.

1.5 PLAN OF THE INVESTIGATION.

This chapter has introduced the general problems associated with artefact contamination of the EEG. Particular attention is paid to OA which is described as a significant EEG contaminant. Current OA removal strategies are described, and section 1.4 details a number of significant problems associated with their implementation. These problems are not easily solved using conventional digital signal processing and this investigation aims to make improvements to the current OA removal by developing new and reliable techniques.

This investigation is divided into 5 work packages each of which is documented in this thesis, when possible, as a separate chapter. Figure 1.16 represents the temporal relationship between these work packages. The 5 work packages are :-

. Theoretical development of new OA removal techniques.

. Data acquisition.

. Investigation of OA removal procedures.

. Intelligent OA removal system development.

. Evaluation.

An initial literature search was to provide adequate theoretical material to enable new and reliable techniques for OA removal to be devised. Chapter 2 presents this theory in an applied form, addressing each of the problems associated with the conventional OA removal algorithm. The development of an intelligent OA removal system is

Time (Months)

Literature search

Current research
Background information

Data acquisition

.Hardware design
.Construction
.Testing
.Controlled experiments
.Database compilation

REGISTRATION (CNAA9R)

6

OA removal

Repeat current technique
Compare EOG models
Compare EEG montages
Quantify correction
Devise best removal strateg

12

Characterisation

OA characterisation
Abnormal wave
characterisation
.Knowledge elicitation
.Knowledge base compilation

18

TRANSFER (CNAA9T)
PAPER

Development

Software design
Implementation
Testing
Knowledge base
transcription

24

PAPER

Evaluation

.Limited feature set
.Full feature set

30

PAPER   *THESIS*

36

Figure 1.16 Investigation work packages, and their

temporal relationship.

considered which incorporates, into the OA removal algorithm, some of the expert-knowledge of the EEG clinician. The incorporation of this knowledge into the OA removal algorithm will improve the performance of the algorithm by identifying OA, and differentiating OA from abnormal slow cerebral waveforms.

Having carefully planned the investigation it was necessary to acquire sufficient EEG data from the collaborating hospital. This data was to:-

- Provide information on a range of EEG artefacts.
- Provide a database of both normal and abnormal EEG signals.
- Allow a comparison of OA removal techniques.
- Provide information with which to develop intelligent removal strategies.
- Enable evaluation of intelligent OA removal strategies and comparison with conventional OA removal.

In order to acquire sufficient data, in controlled experiments, a data acquisition system was designed and constructed. Chapter 3 details this development and forms the second work package.

Initial analysis of the acquired data from normal patients enabled conventional adaptive OA removal to be repeated. Further, several aspects of OA removal were investigated. For example, both referential and bipolarly recorded EOGs were used to remove OA from EEGs recorded in referential and bipolar montages. The results of this were compared to investigate the hypothesis that for best results, bipolar and referential

EEG montages require different OA removal strategies. This work package also enabled the best EOG signals to be identified for OA removal from a much larger number of EEG signals than previously examined. Chapter 4 details the procedures used in this work package and the results of the analysis.

Having identified appropriate EOG signals for OA removal from various combinations of EEG signals over the scalp, it was necessary to characterise the main OA waveform groups by analysis of, for example, potential distribution over the scalp. This would enable detection of possible OA signal contamination. Further, the acquired data from a number of abnormal patients exhibiting various abnormal EEG waveforms were analysed in a similar fashion and the comparison of characteristics allowed a number of rules to be developed to allow differentiation between, for example, abnormal slow waves and OA. The results of this analysis, together with elicited information from EEG experts, allowed a knowledge base of rules and heuristics to be compiled that captured the OA identification and differentiation skills that are commonly utilised by an experienced EEG expert.

Having developed intelligent strategies to allow the identification of OA waveforms and therefore the directed use of a selected OA removal algorithm, a software simulation system was developed. The system was implemented using personal computer (PC) based hardware and consisted of a graphical user interface, an expert system, and a number of selectable adaptive filter algorithms. The development of the intelligent OA removal system is detailed in chapter 5. This chapter represents the kernel of

this investigation and draws together the information contained in all previous chapters into a working system.

The performance of the intelligent OA removal system is evaluated in chapter 6. This takes the form of four distinct evaluation stages, representing distinct stages in system development. The first stage evaluates consistency of the transcribed expert knowledge. The second stage evaluates the operation of the system, using key features extracted from the EEG, using signal processing techniques. The third stage evaluates the performance of the system when used to remove OA after having been identified. The final stage of evaluation details the results of a preliminary clinical evaluation of the system. and provides important end user comments.

The final chapter of this thesis critically examines the results obtained from this investigation and discusses a number of aspects related to the intelligent removal of OA from the EEG. This discussion enables possible further work to be introduced and the thesis is concluded by summarising the principle advancements made during this investigation.

REFERENCES FOR CHAPTER 1.

Barlow, J.S.

*"Computerised clinical electroencephalography in perspective". IEEE
Tran. on Biomed. Eng., Vol.BME-26, No 7, pp. 377-391, July 1977.*

Barlow, J.S.

*"Eye movement artifact nulling in EEG's by multichannel on-line EOG
subtraction"". Electroenceph. and clin. neurophysiol., 1981, 52, pp.
418-423.*

Barry, W and Jones, G.M.

*"Influence of eyelid movement upon Electro-oculographic recording of
vertical eye movements". Aerospace Medicine, 1965, Vol. 36, pp. 855-861*

Bickford, R.G.

*"Computer analysis of background activity". In: EEG Informatics, pp.
215-232, 1977. Ed. Remond, A. Amsterdam; Elsevier.*

*Binnie, C.D., Batchelor, B.G., Bowring, P.A., Darby, C.E., Herbert,
L., LLoyd, D.S.L., Smith, D.M., Smith, G.F. and Smith, M.*

*"Computer-assisted interpretation of clinical EEGs". Electroenceph.
and clin. neurophysiol., 44, pp. 575-585, 1978.*

Borda, R.P. and Hablitz, J.J.

*"Use of a simple visual display to reduce eye movement artefacts in
CNV recordings". EEG Journal, 34, pp. 433-436, 1973.*

*Brunia, C.H.M., Möcks, J., van den Berg-Lenssen, M.M.C., Coelho, M.,
Coles, M.G.H., Elbert, T., Gasser, T., Gratton, G., Ifeachor, E.C.,
Jervis, B.W., Lutzenberger, W., Sroka, L., van Blokland-Vogelesang, A.W.,
van Driel, G., Woestenburg, J.C., Berg, P., McCallum, C., Tuan, P.T.,
Pocock, P.V., Roth, W.T.*

*"Correcting ocular artifacts in the EEG: a comparison of several methods",
Journal of Psychophysiology, Vol. 3, pp. 1-50, 1989.*

Carrie, J.R.G.

*"A technique for analyzing transient EEG abnormalities".
Electroenceph. and clin. neurophysiol., 1972a, 32, pp. 199-201.*

Carrie, J.R.G.

"A hybrid computer technique for detecting sharp EEG transients".
Electroenceph. and clin. neurophysiol., 1972b, 33, pp. 336-338.

Carrie, J.R.G.

"A hybrid computer system for detecting and quantifying spike and wave
EEG patterns". Electroenceph. and clin. neurophysiol., 1972c, 33,
pp. 339-341.

Cooper, R., Osselton, J.W. and Shaw, J.C.

"EEG Technology". 3rd ed., Butterworths, 1980.

Corby, J.C., and Kopel, B.S.

"Differential contributions of blink and vertical eye movements as
artefacts in EEG recording", Psychophysiology, Vol. 9, pp. 640-644, 1972.

Creutzfeldt, O.D. (Ed.)

"The neural generation of the EEG", in: E.C.N. handbook, Vol. 2, Pt. C,
Elsevier, Amsterdam, 1974.

Ferrara, E.R. and Widrow, B.

"Time sequenced adaptive filter". IEEE Trans. Acoust. Speech &
signal processing, 29, pp. 679-683, 1981.

Fortgens, C., and DeBruin, M.P.

"Removal of eye movement and ECG artifacts from the non-cephalic reference
EEG", Electroenceph. Clin. Neurophysiol., Vol. 56, pp. 90-96, 1983.

Gasser, T., Sroka, L., and Möcks, J.

"The correction of EOG artifacts by frequency dependant and frequency
independant methods", Psychophysiology, Vol. 23, pp. 704-712, 1986.

Gevins, A.S., Yeager, C.L., Zeitlin, G.M., Ancoli, S. and Dedon, M.F

"On-line computer rejection of EEG artifact". Electroenceph. and
clin. neurophysiol., 42, pp. 267-274, 1977.

Gibbs, F.A., and Gibbs, E.L.

"Atlas of electroencephalography", Addison-Wesley, Reading, Mass., 1951.

Girton, D.G. and Kamiya, J.

"A simple on-line technique for removing eye movement artifacts from the EEG". Electroenceph. and clin. neurophysiol., 1973, 34, pp. 212-216.

Gotman, J., Skuce, D.R., Thompson, C.J., Gloor, P., Ives, J.R. and Ray, W.F.

"Clinical applications of spectral analysis and extraction of features from electroencephalograms with slow waves in adult patients". Electroenceph. and clin. neurophysiol., 1973, 35, pp. 225-235.

Gotman, J., Gloor, P., and Ray, W.F.

"A quantitative comparison of traditional reading of the EEG and interpretation of computer extracted features in patients with supratentorial brain lesions". Electroenceph. clin. neurophysiol., 34, pp. 623-639, 1975.

Gotman, J. and Gloor, P.

"Automatic recognition and quantification of interictal epileptic activity in the human scalp". Electroenceph. and clin. neurophysiol., 1976, 41, pp. 513-529.

Gotman, J., Gloor, P. and Schaul, N.

"Comparison of traditional reading of the EEG and automatic recognition of interictal epileptic activity". Electroenceph. and clin. neurophysiol., 1978, 44, pp. 48-60.

Haykin, S.

"Adaptive filter theory". Prentice-Hall, information and system sciences series, 1986.

Hector, M.L.

"EEG recording, "2nd edition, Butterworths, 1980.

Hellyar, M.T., Ifeachor, E. C., Mapps, D.J., Allen, E.M. and Huson N.

"An expert system approach to EEG signal processing", In Preparation,
1991.

Hess, R.

"EEG handbook", Sandoz Ltd., Zurich, 1966.

Ifeachor, E.C.

"The removal of eye movement artefacts from the EEG signals - A
survey". Research Report No. 3, Plymouth Polytechnic, May 1983.

Ifeachor, E.C.

"Investigation of OA in the human EEG and their removal by a micro-
processor-based instrument, PhD thesis, Plymouth Polytechnic, England,
1984.

Ifeachor, E.C., Jervis, B.W., Allen, E.M. and Hudson, N.R.

"A new microcomputer-based online ocular artefact removal (OAR)
system". IEE proceedings, Vol. 133, Pt. A, No. 5, July 1986.

Ifeachor, E.C., Jervis, B.W., Allen, E.M, Morris, E.L., Wright, D.E.
and Hudson, N.R

"Investigation and comparison of some models for removing ocular
artefacts from EEG signals: Part 1 Review of models and data
analysis". Med. & Biol. Eng. & Comput., 26, pp 584-590, Nov. 1988.

Ifeachor, E.C., Jervis, B.W., Allen, E.M, Morris, E.L., Wright; D.E.
and Hudson, N.R

"Investigation and comparison of some models for removing ocular
artefacts from EEG signals: Part 2 quantitative and pictorial
comparison of models". Med. & Biol. Eng. & Comput., 26,
pp 584-590, Nov. 1988.

Jasper, H.H.

"Report of the commitee of methods of clinical examination in
electroencephalography". EEG jl., 10, pp 370, 1958.

Jervis, B.W., Nichols, M.J., Allen, E.M., Hudson, N.R. and Johnson,
T.E.

"The assesment of two methods for removing eye movement artefact from
the EEG". Electroenceph. and clin. neurophysiol., 1985, 61, pp.
444-452.

Jervis, B.W., Ifeachor, E.C., and Allen, E.M.

"The removal of ocular artefacts from the electroencephalogram:
a review". Med. & Biol. Eng. & Comput., 26, pp. 2-12, Jan. 1988.

John, E.R., Karmel, B.Z., Corning, W.C., Easton, P., Brown, D., Ahn,
H., John, M., Harmony, T., Prichep, L., Toro, A., Gerson, I.,
Bartlett, F., Thatcher, R., Kaye, H., Valdes, P. and Schartz, E.

"Neurometrics". Science, June 24th, 1977, Vol. 196, pp. 1393-1410.

Ktonas, P.Y.

"Automated analysis of abnormal electroencephalograms". CRC critical
reviews in biomedical engineering, Vol. 9, Issue 1, pp.39-97, 1983.

McCallum and Walter, W.G.

"The effects of attention and distraction on the contingent negative
variation in normal and neurotic subjects". Electroenceph. and clin.
neurophysiol., 25, pp. 319-328, 1968.

Muras, S. and Binnie.

"The recognition of frontal slow activity in the presence of eye
movements". Proc. Electrophysiological Technologists' Assoc., 17,
pp. 131-143, 1970.

Overton, and Shagass, C.

"Distribution of eye movement and eye blink potentials over the scalp".
Electroenceph. Clin. Neurophysiol. Vol. 27, pp. 546, 1969.

Papakostopoulos, D., Winter, A. and Newton, P.

"New techniques for the control of eye potential artefacts in
multichannel CNV recordings". Electroenceph. and clin. neurophysiol., 34,
pp. 651-653, 1973.

Quilter, P.M., MacGillivray and Wadbrook, D.G

"The removal of eye movement artefact from EEG signals using
correlation techniques". Random signal analysis, IEE conference
publication, No. 159, pp. 93-100, 1977.

Rémond, A. Ed.

"EEG Informatics". Amsterdam, Elsevier, 1977.

Shackel, B.

"Eye movement recording by Electo-oculography". A manual of
Psychophysiological methods, ed., P.H. Venables and I. Martin, North
Holland Pub. Co., 1967.

Sharman, K.C., Chambers, C. and Durrani, T.S.

"Rule driven adaptive signal processing". IEE coloquium on "The
application of A.I techniques to sensor systems", London, April, 1987.

Sullivan, G., and Weltman, G.

"The impedence oculogram" Journal of applied physiology, Vol. 18,
pp. 215-216, 1963.

Verleger, R., Gasser,T. and Möcks, J.

"Correction of EOG artefacts in event-related potentials of the EEG:
Aspects of reliability and validity". Psychophysiology, Vol. 19,
pp. 474-480, 1982.

Young, L.R. and Sheena, D.

"A survey of eye movement recording methods". Behavior Research
Methods and Instrumentation, Vol. 7(5), pp. 397-429, 1975.

Young, P.

"Recursive approaches to time series analysis". Bull. IMA, 10,
pp. 209-224, 1974.

CHAPTER 2

INTELLIGENT ENHANCEMENT OF THE EEG

2.1 INTRODUCTION.

Chapter 1 has described how artefactual waveforms can corrupt the EEG, and shown the need for their removal to maintain the integrity of abnormal slow waveforms that provide crucial diagnostic information. The main limitation with the current OA removal techniques, using conventional signal processing, is the inability to differentiate between artefactual and abnormal waveforms *[Ifeachor, et al., 1989]*. This can mean that the removal process can attenuate abnormal waveforms and can, at worst, remove them entirely. This problem and the ones outlined in section 1.4 are not easily solved using conventional signal processing techniques because of the similarity in characteristics of the artefactual waveforms and abnormal cerebral waveforms. This chapter introduces a number of techniques to overcome these deficiencies using an artificial intelligence (AI) approach to applied signal processing.

2.2 IMPROVING CONVENTIONAL OA REMOVAL.

An EEG expert is often able to recognise and differentiate between OAs and can distinguish OAs from abnormal slow cerebral waveforms. The expert utilises a wealth of experience and knowledge when analysing the EEG. It

has been found (see section 4.5) that the key elements used in EEG analysis are the use of time and frequency domain information. This is underpinned by a knowledge of the patients clinical history. Figure 2.1 illustates a conceptual view of EEG analysis.
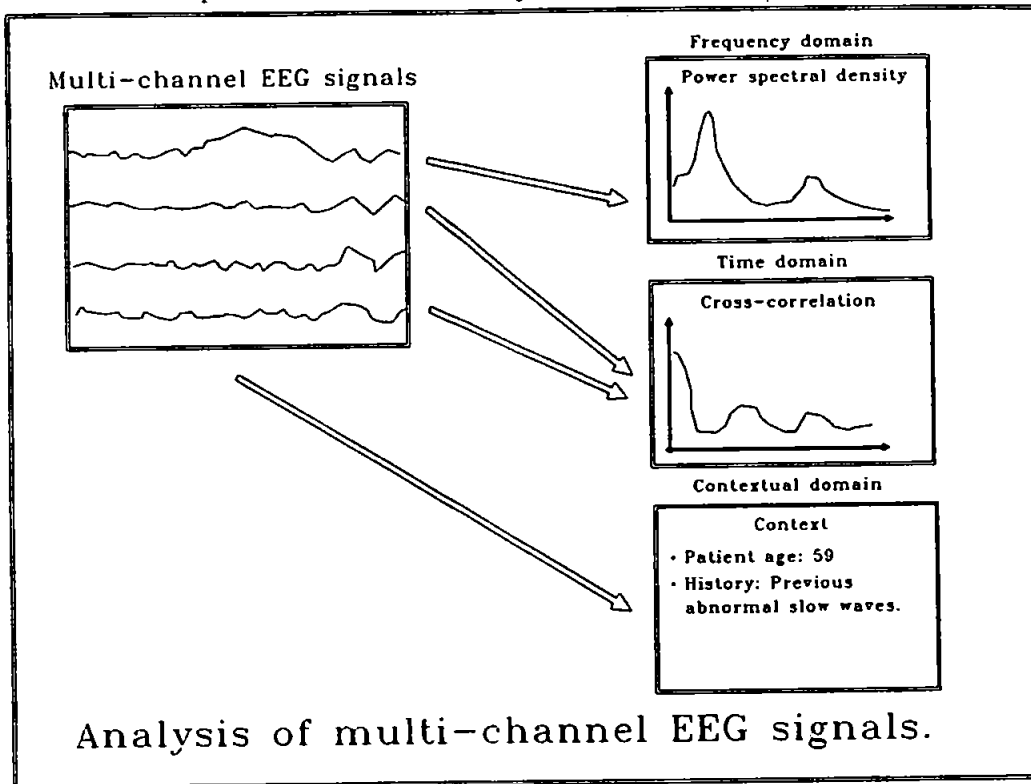
Figure 2.1 A conceptual view of EEG analysis.

The deficiencies of the present signal processing techniques can be overcome by incorporating EEG expert knowledge into the OA removal algorithm. The expert knowledge will allow the problems related to the conventional OA removal algorithm, detailed in the last chapter, to be addressed:

(a) Infringement of correlation requirements.

In order to overcome this problem, it will be necessary to distinguish between OAs and abnormal waveform patterns before applying the OA removal

algorithm. Only then, when identified, can the OA be removed without fear of further signal corruption. It will be necessary to characterise the OA and abnormal waveforms. Characterisation of the waveforms will involve elicitation of the knowledge used by the EEG expert in differentiating between OA and abnormal cerebral waveforms. Characteristic features will then be extracted from the EEG using digital signal processing techniques to identify areas requiring OA removal.

(b) Nonstationarity of the EEG.

A possible solution to this problem involves resetting the OA parameters of the algorithm to a suitable set of values when significant change in signal statistics is detected. Sharman, (1987), utilises a similar technique for the estimation of an all pole model for a set of incoming data. Detection of a change in the signal statistics are based on quantities such as:-

. Current and previous estimates of the all pole model parameters.

. Acceptable range bounds for the estimation error.

. Statistical measures of input signal (mean, variance etc.)

. History list of the estimation performance.

. History list of input samples.

. A priori probabilities and characteristics of interferences.

Upon detection of a change in signal statistics the current parameters are discarded and the algorithm is restarted. The finite memory of the OA removal algorithm ensures that the effect of old data are purged and only new data used in the subsequent adaptive process.

(c) Multiple artefacts.

The simple resetting of the OA parameters, as discussed in above, will not provide adequate OA removal because of the need to converge on different OA parameters for each new OA type. However, as there exist only a finite number of ocular artefacts it should be possible to store in memory an optimal set of OA parameters for each type of OA, and for each electrode position. Upon detection of possible OA and identification of the type of OA, by examination of the extracted characteristic features, it should be possible to reset the OA removal algorithm to the new stored optimal OA parameters. This regime would have the effect of applying a separate and most suitable filter to each OA type and would mean that for individual OAs, the optimal set of parameters would be found much more quickly without the need for re-convergence

(d) Secondary artefacts.

The effects of secondary artefact contamination can be minimised by :-

.   EOG signal pre-processing.
.   Directed OA removal.

EOG signal pre-processing would involve minimising the EOG artefacts. An FIR low pass digital filter can be used to attenuate signal contaminants such as muscle activity and electrode pop. Directed OA removal would involve applying the OA removal algorithm only where OA is detected and identified.

(e) OA removal algorithm memory.

It is preferable to apply the OA removal algorithm only in the regions of the EEG that are actually contaminated with OA. The effects of the old data can be minimised during the transition period of changing OA parameters by reducing the effective memory of the algorithm.

2.3 DECISION SUPPORT.

The role of an intelligent OA removal system is seen to be twofold. Firstly, such a system can be used at recording time to provide a OA free record of the EEG recording. This will be in addition to a conventional 'OA corrupted' record of the same EEG. OA removal will be under the control of the recording personnel and could incorporate a threshold for OA removal. The second role for an intelligent OA removal system, and probably the most important one is under the control of the EEG expert during post recording analysis and diagnosis. In this role the intelligent OA removal system will be able to scan through large amount of EEG data identifying OA without necessarily performing OA removal. The expert system will be able to justify any decisions made by retracing the inferences it made in order to reach the decision. This is a fundamental attribute of any expert system, but in an intelligent OA removal system this will prove to be a valuable decision support aid for the EEG expert. Once the expert is satisfied with the reasoning behind an OA identification, OA removal could be carried out at the touch of a button. This will provide the EEG expert with an intelligent real-time EEG signal processor.

The combination of an AI approach with conventional signal processing will produce an elegant solution to the problem of total OA removal. The OA recognition and differentiation knowledge, of an EEG expert, is embodied into an expert OA removal system. This enables a directed and selective, or intelligent signal processing, approach to OA removal, illustrated in figure 2.2.

conventional OA removal

Intelligent OA removal
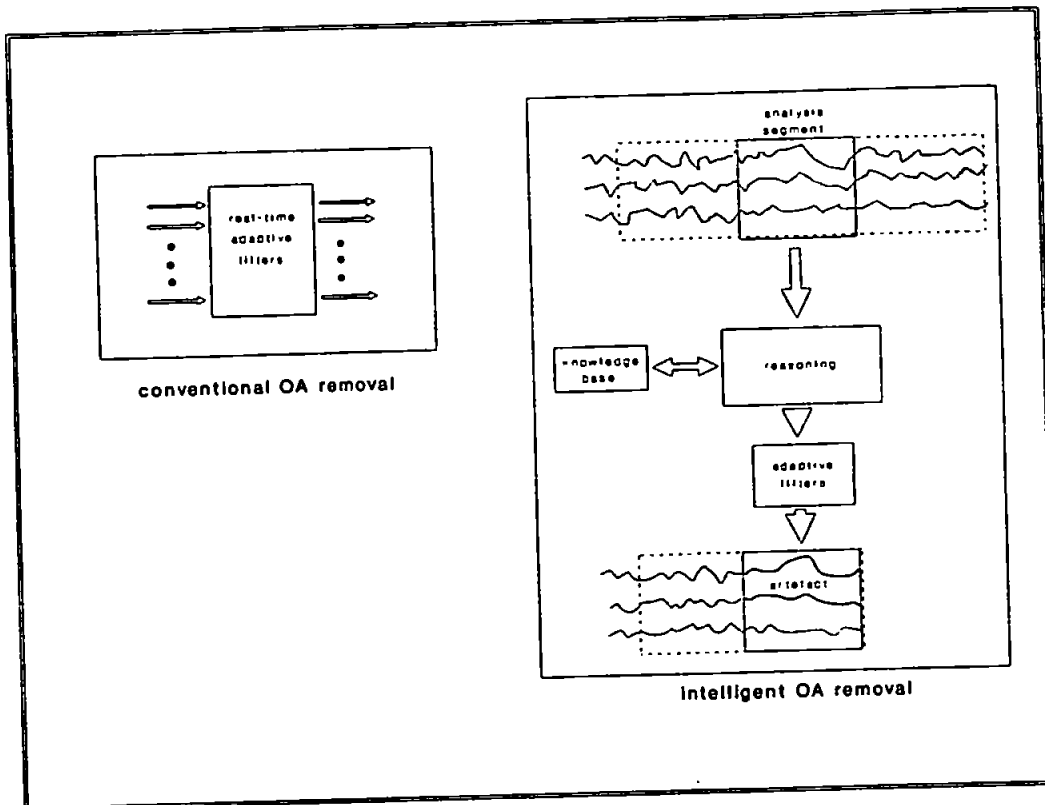
Figure 2.2 Comparison between conventional OA removal
and an intelligent approach.

OA removal is directed because OAs are only removed when they are present and OA removal is selective because different artefacts are distinguished from one another enabling the most suitable adaptive filter to be applied. The remainder of this chapter details the developmental aspects involved in the use of an AI approach to EEG signal processing.

2.4 ARTIFICIAL INTELLIGENCE.

Artificial intelligence (AI) is a branch of computer science which makes an attempt to model human cognitive processes using sophisticated computer techniques *[Winston, 1984]*. An expert system (ES) is an application of AI which involves the incorporation of a human experts knowledge, of a specific problem area, organised in a highly structured manner *[Forsyth, 1984]*. The main components of an ES are, a knowledge base, and a reasoning or inference mechanism. The knowledge contained in the knowledge base is likely to be a mixture of rigorous scientific formulae and heuristics, or rules of thumb, that an expert would regularly use to solve the particular problem. The inference mechanism is an algorithm for the manipulation of the knowledge, combined with certain acquired facts, to enable some new knowledge or fact to be inferred. This algorithm will generally obey some formal rule of logic and the resultant new knowledge or fact will represent a decision and will therefore dictate the next operation to be taken, either by some control mechanism or a human operator. The knowledge base will be highly structured and quite separate from any 'code' that will be used to manipulate it. This separation of 'code' and 'data' enables the knowledge base to be easily modified and allows the same inference mechanism to be used on a different knowledge base.

Expert systems have been shown to be successful in solving complex tasks in restricted problem areas, and are reported extensively *[Rauch-Hinden, 1985]*. An expert system, containing the OA identification capabiliies of the EEG expert will allow intelligent

enhancement of the EEG, by directing the OA removal algorithm. This will ensure minimal algorithm application, and hence the retention of maximum EEG information. An expert system approach is well suited to the intelligent enhancement of the EEG, for a number of reasons:

. There exist EEG experts who are able to communicate their knowledge and experience.

. The problem area is restricted to a finite number of artefacts.

. Extensive literature is available on artefacts.

. EEG analysis is by nature heuristic.

. There is an abundant supply of clinical data.

Expert systems are implemented using one of the following three formats:

. A commercial ES shell.

. A 'from scratch' development typically using an AI software language.

. A mixture of the two above.

A commercial ES shell generally provides a standard inference mechanism and an empty knowledge base. It is therefore, the developers responsibility to acquire and develop a separate and appropriately coded knowledge base on which the inference mechanism operates. An ES shell is generally considered to operate best when applied to a problem domain with a close similarity to the problem domain on which the ES shell was developed. The developer is also required to provide an appropriate means

of interfacing the ES with the application environment. This investigation has involved the development of an EEG ES 'from scratch' using an AI software language. This has enabled the inference mechanism, the user interface and the knowledge base to be developed specifically for the problem of OA detection and identification.

2.5 DEVELOPMENT CONSIDERATIONS FOR AN INTELLIGENT OA REMOVAL SYSTEM.

This section discusses some of the developmental aspects of an expert system for OA identification and differentiation and in particular focuses on the theoretical aspects involved. These aspects include the type of knowledge that the system must contain, how to represent it and how to use it in order to affect improved OA removal. Chapter 5 builds on these issues and describes the current working implementation.

2.5.1 KNOWLEDGE REPRESENTATION AND MANIPULATION.

Central to the development of an intelligent OA removal system is the way in which expert knowledge is represented, and the inference mechanism used to manipulate it. The representation of the knowledge will depend largely on the type and format of the knowledge, and how the inference mechanism will use this knowledge. OA detection and identification knowledge consists of heuristic rules and a large number of precise electrical measurements. The inference mechanism used for the

identification and differentiation of OAs will depend on the expert knowledge available and the way in which this is represented in the computer system. Because of the fundamental importance and strong interdependence of the knowledge representation and manipulation both must be carefully considered during the development of the intelligent OA removal system.

Common knowledge representations include :-

. Procedural: Where the expert knowledge, and the knowledge of how and in exactly what order to perform the task, are stored together. This is common in most high level computer languages.

. Semantic networks: Where the expert knowledge is represented by a network of interconnected nodes. The nodes represent objects and a connection between objects represents some relationship that one object has to the other *[Winston, 1984]*. Closely related to this representation is that of a 'frame' *[Minsky, 1981; Graham and Jones, 1988]* where a frame is capable of holding knowledge in 'slots' within the frame. Each slot represents some relationship in a similar way as a link, and the contents of the slot is the object of that relationship. In this manner the representation can be viewed as a relational database.

. Productions: Where the expert knowledge is represented by discrete statements, or rules, of knowledge *[Newell and Simon 1972]*. Each statement is described as a production rule and is used to encapsulate either factual or heuristic knowledge in the form:-

RULE A:
        IF   Very low frequency waveforms are present   2.1

       · · THEN artefacts are very likely to be present

The production rule representation of knowledge has been selected for this investigation because it is easy to understand, and it can represent knowledge of both heuristic and specialised nature. Because each rule forms a distinct piece of knowledge, it can easily be amended or replaced. This is also the most commonly used structure in existing systems *[Jansen, et al., 1985; Baas, et al., 1984; Jagannathan, et al., 1982; Bourne, et al., 1981; Sortliffe, 1976]*.

Production rules, representing the expert knowledge, are quite seperate from the inference mechanism that is used to manipulate that knowledge. However, each production rule forms a logical sentence that obtains its format from a subset of formal logic *[Kowalski, 1983]*. The use of such a logical statement, together with acquired facts enables a new fact to be inferred using a manipulation based on formal logic. In this way logic is used as the inference mechanism and is attractive for a number of reasons:-

.   Logic has a long and established tradition with a strong mathematical formalism.

.   Any inferences made using the rules of logic can be proved, assuming that the original rules and data are valid.

The remainder of this chapter details the use of a formal logic for the development of an ES for the detection and identification of OA. Section 2.5.1.1 introduces the use of logical sentences for the representation of EEG expert knowledge. Section 2.5.1.2 describes the use of logical rules to make inferences. Section 2.5.2 describes the implementation of the logical inference mechanism using the logic programming language PROLOG *[Clocksin and Melish, 1987]*. Finally, section 2.5.3 extends the standard PROLOG inference mechanism to allow more efficient inference.

2.5.1.1 FIRST ORDER PREDICATE LOGIC.

First order predicate logic *[Dowsing, et al., 1986]* allows the knowledge of the EEG expert to be represented unambiguiously. For example, consider the case when the knowledge of the EEG expert in identifying an OA waveform is simplified to the following implication :-

"To be an OA waveform implies that the waveform has a frequency in the delta band, is of maximum amplitude in the frontal electrode positions, and is not coincident with an obvious EEG waveform on another part of the scalp"

This implication is represented with first order predicate logic as :-

$$\forall (x) \, [OA(x) \Leftarrow ([\text{delta-frequency}(x)] \\ \wedge \qquad [\text{max-amp-front}(x)] \\ \wedge \neg \exists (y)[\text{equal-eeg}(x,y)])]$$

2.2

where:
x is an unidentified waveform.
OA(x) is a predicate that is true when x is an OA
delta-frequency(x) is a predicate that is true when x has a frequency
in the delta frequency band
max-amp-front(x) is a predicate that is true when x is of maximum
amplitude in the frontal regions of the head
equal-eeg(x,y) is a predicate that is true when there is an EEG
waveform coincident with x
∀(x) this is true for all x.
¬∃(y) there is no y that will make this true.
∧ represents 'and'
∨ represents 'or'
¬ represents 'not'
⇐ represents 'implies'

Equation 2.2 can be written in the form of the production rule of
equation 2.1:

IF   2.3
    frequency is delta
and
    amplitude is maximal in the front
and
    no equal EEG exists

THEN
    waveform is OA

2.5.1.2 INFERENCE.

If the rule represented by equation 2.2 is provable from a given set
of axioms, or facts, we write:-

    Facts ├─ Rule

The facts correspond to features extracted from the EEG signals, and the rules elicited from an EEG expert. A formal proof of the rule from a set of facts follows a finite sequence. Each step in the proof generates a new fact from existing ones, using a logical rule of inference. For example, in order to prove the rule represented in equation 2.2 it is first necessary to establish the truth of the individual components of the antecedent - IF clause. The truth of these components will be established from other rules in the knowledge base.

A production rule represents a logical rule of inference termed modus ponendo ponens (MPP). For example, given the facts:

1   waveform is OA $\Leftarrow$ frequency is delta

and amplitude is maximal in the front

and no equal EEG exists

2   frequency is delta

3   amplitude is maximal in the front

4   no equal EEG exists

then the rule of MPP allows the following new fact to be inferred:

---

5   waveform is OA

$1\ 2\ 3\ 4 \vdash 5$

In addition to MPP, the rule of Reductio Ad Absurdum (RAA) is used to remove facts from a knowledge base when a contradiction is found. RAA is

of principal importance when considering resolution using proof by refutation. Resolution is used to automate the process of inference and is used by the PROLOG language. Resolution can be used to prove the truth of a theorem, i.e. 'waveform is OA', by proving that the negation of this theorem cannot be true *[Amble, 1987]*. The theorem may consist of a large number of rules and facts. For example, Given the following facts:-

6   $\neg P \lor Q$
7   P

and the theorem to be proved is Q, then by negating this theorem,

8   $\neg Q$

Resolution of axioms 6 and 7, using RAA above, simplifies to:-

9   Q

which is in contradiction to 8, hence proving $\neg Q$ to be false and Q to be true. This is called proof by refutation.

Resolution requires that the facts are in a clause form that contain disjunctions of predicates and negated predicates only - a HORN clause. This is the from of facts 6 and 7 above. Equation 2.2 is converted to HORN clause form using the following procedure:

$$\forall \ (x) \ [OA(x) \Leftarrow ([delta\text{-}frequency(x)] \\ \land \qquad [max\text{-}amp\text{-}front(x)] \\ \land \neg \exists \ (y)[equal\text{-}eeg(x,y)])] \qquad\qquad 2.2$$

STEP 1: implications are eliminated using the substitution :

$$A \Rightarrow B \equiv \neg A \lor B.$$

$$\forall \ (x) \ [OA(x) \lor \neg([delta\text{-}frequency(x)] \\ \land \qquad [max\text{-}amp\text{-}front(x)] \\ \land \ \neg \ \exists \ (y)[equal\text{-}eeg(x,y)])] \qquad\qquad 2.4$$

STEP 2: negations are moved inside the atomic formulae.

$\neg \exists (x)[A(x)] \equiv \forall (x)[-A(x)]$

$\forall (x)$ [OA(x) $\vee$ ($\neg$ [delta-frequency(x)]
$\qquad \vee \quad \neg \quad$ [max-amp-front(x)]
$\qquad \vee \neg \forall (y)[\neg$ equal-eeg(x,y)])]   2.5

STEP 3: Universal quantifiers are moved to the left.

$\forall (x) \forall (y)$[OA(x) $\quad \vee \quad \neg \quad$ delta-frequency(x)
$\qquad\qquad\qquad \vee \neg \quad$ max-amp-front(x)
$\qquad\qquad\qquad \vee \quad$ equal-eeg(x,y)]   2.6

STEP 4: All variables are assumed universally quantified and can therefore be removed.

OA(x)$\vee\neg$delta-frequency(x)$\vee\neg$max-amp-front(x)$\vee$ equal-eeg(x,y)   2.7

Equation 2.7 is now in the correct form for proof by refutation. Further details can be obtained in Graham, (1988), and Kowalski (1983). For example, the analysis of a segment of EEG provides a number of measurements related to a suspect waveform x, and a genuine EEG waveform y. This allows the following facts to be established:-

10   OA(x)$\vee\neg$delta-frequency(x)$\vee\neg$max-amp-front(x)$\vee$ equal-eeg(x,y)
11   delta-frequency(x)
12   max-amp-front(x)
13   $\neg$equal-eeg(x,y)

To prove the theorem OA(x) is true, it is necessary to negate the theorem, i.e.

14   $\neg$OA(x)

Resolving 10 and 11 produces the resolvent:-

15   OA(x)$\vee\neg$max-amp-front(x)$\vee$ equal-eeg(x,y)

Resolving 15 and 12 produces the resolvent:-

16   OA(x) $\vee$ equal-eeg(x,y)

Resolving 16 and 13 produces the resolvent:-

17   OA(x)

The contradiction between facts 17 and 14 proves that the negated theorem is false and therefore proves that the waveform is an OA, given the initial facts.

2.5.2 PROGRAMMING IN LOGIC (PROLOG).

Representation of EEG expert knowledge and manipulation using the methods detailed above is automated when programming using PROLOG. PROLOG is a common name for a family of programming languages which implement Predicate logic as a programming language. PROLOG is a declarative programming language meaning that no explicit control structure exists, and in place of this is an implicit search of the program clauses for a contradiction to provide proof of a theorem *[Amble, T., 1987; Rowe, N., 1988]*. The program clauses consist of a list of rules and facts which are both represented in HORN clause form:-

$$a1 \vee \neg b1 \vee \neg b2 \ldots \vee \neg bn$$

which is equivalent in PROLOG syntax to :-

al :- b1,b2,...bn.

n = 1   represents a PROLOG rule
n = 0   represents a PROLOG fact

A theorem is proved by presenting the PROLOG program with a goal clause, which could consist of a conjunction of positive literals with variables or more simply a single literal, e.g.

remove-OA?

PROLOG will effectively negate this goal and add this to its list of clauses. The program will proceed by attempting to find a contradictory clause to the negated goal clause from the list of fact and rule clauses and to then resolve to produce a new negated goal clause. This process will continue until either no contradictory clause can be found - fail, or the goal clause is empty - true. Resolution is said to be linear because for any goal clause the contradictory clause will be found in the list of clauses. Resolution is said to be selected because the particular literal to resolve must be selected from the goal clause. In most PROLOG implementations this selected literal is the first in the goal clause.

The resolution of clauses can be viewed as a search path which is directly effected by the selection of literals to resolve. The path may not always directly provide an empty goal and when no contradictions can be found, alternate clauses are attempted by retracing, or backtracking, to the last resolution clause, replacing unified variables as necessary. This search of the list of clauses can be viewed as a search space of the rules and facts.

2.5.2.1 SEARCH.

To illustrate the concept of a search, consider figure 2.3. The goal clause here is that of 'remove-OA?'. Figure 2.3(a) is a list of the PROLOG programs rule and fact clauses, which is simplified in the second part of 2.3(a) for ease of presentation in the resolution diagram of figure 2.3(b). The resolution procedure of section 2.5.1.2 is followed and the

```
1  remove-OA? :- OA(x),no-abnormal.
2  OA(x) :- only-EOG.
3  OA(x) :-delta-freq(x),max-amp-front(x),not(equal-eog(x,y)
4  delta-freq(x) :- freq>0.5Hz,freq<4Hz.
5  freq>0.5Hz
6  freq<4Hz.
7  max-amp-front(x).
8  no-abnormal
```

This is simplified to the following

```
1  a:- b,c.
2  b:- d.
3  b:- e,f,g.
4  e:- h,i.
5  h.
6  i.
7  f.
8  c.
```

Figure 2.3(a) PROLOG program clauses.

Figure 2.3(b) Proof of a.

Figure 2.3 An example of theorum proving using PROLOG search.

diagram first shows the negated goal clause {ā}. The first clause found that causes a contradiction of the goal is that of clause 1 and the resolvent of this, i.e. {b̄,c̄} becomes the new goal clause. From this goal clause the negated literal b becomes the focus of attention and it is this literal for which a contradiction will next be sought from the list of clauses. The execution of the PROLOG program is therefore described as:-

.   Top down, because only goals are executed.

.   Left to right of the conditions.

.   First to last of the conditions.

.   Depth first search with backtracking.

PROLOG keeps a current sequence of conditions which it tries to solve with the same substitutions for all the stated variables. Such a condition sequence is called the goal.

It is clear from the above example that the order of the clauses is important to the efficiency of program execution, a reversal of clauses 2 and 3 would in this example improve the efficiency of execution. This indicates that the control structure of PROLOG is in fact a declarative and procedural hybrid and that careful consideration needs to be taken when compiling the list of fact and rule clauses for the intelligent OA removal system.

2.5.3 ADDITIONAL SEARCH STRATEGIES.

Section 2.5.2.1 has described PROLOGs implicit search strategy, based on a top down, depth first search with backtracking. This search strategy is effective for this application because of the large amount of low level feature data generated from the analysis of the EEG/EOG data segment. Classification of each analysis segment relies on a search through a decision tree, or search space, generated from the production rules elicited from the expert, similar to that of figure 2.3.

Repetition of the same search path for each new analysis segment can be wasteful of time and is often unnecessary. The nature of EEG signal analysis is such that waveforms are viewed in the context of the surrounding waveforms, and the history of the patient. For example, a suspect waveform occurring between two clearly abnormal waveforms is more

likely to be interpreted as an abnormal waveform also, and therefore remove the need for signal enhancement. Similarly, a suspect waveform occurring in the EEG of a patient with previously observed abnormal waveforms is also more likely to be interpreted as an abnormal waveform also. The depth first backward chaining search mechanism of PROLOG does not accommodate this type of heuristic search and therefore additions to the search mechanism are necessary.

2.5.3.1 FORWARD AND BACKWARD CHAINING HYBRID SEARCH.

The experts use of contextual features to aid with the identification of OA waveforms is simulated by the addition of several contextual facts into the knowledge base. These facts include the immediate past and future analysis segment classifications, and a frequency profile for all the past analysis segment classifications. Each successful segment classification then enables the contextual feature facts to be updated and asserted into the knowledge base. The decision tree is restricted, or pruned *[Winston, 1984]* by utilising special rules in the knowledge base that examine the contextual features and direct further search according to their success. For example:-

    Rule IV:
        IF   abnormal waveforms have been identified in the previous
              segment
        THEN there is an increased reason to believe that the present
              segment contains abnormal waveforms.

Upon success of this rule, subsequent search is continued at a point in the search space where this hypothesis can be corroborated by obtaining

further evidence to support it - a jump to a so called node marked 'abnormal only'. The belief in the previous segment classification is carried through to the belief in the classification of the present segment. If this search path provides insufficient evidence to support the heuristic, then search continues in the conventional manner. The addition of contextual information allows a mixed search strategy to be employed. The inclusion of contextual facts allow inference to a conclusion to be made by forward chaining. Additional facts are then sought to confirm the conclusion by backward chaining. This more closely matches the experts method of analysis as well as improving the efficiency of the inference mechanism for the vast majority of cases.

2.6 SUMMARY OF CHAPTER 2.

This chapter has advanced a number of important concepts for improved OA removal to overcome the deficiencies of the conventional OA removal algorithm detailed in chapter 1. Expert system techniques have been introduced and their role in a prospective intelligent OA removal system discussed. A structure for the representation of a human EEG experts knowledge has been detailed, together with a formal method of knowledge manipulation that utilises this structure. The PROLOG programming language has been introduced as a medium for the implementation of formal knowledge manipulation and an example of OA identification given that demonstrates the principles of search in a decision space. Lastly, extensions to this search technique are given which will more closely match that of a human EEG expert.

REFERENCES FOR CHAPTER 2.

Amble, T.

"Logic programming and knowledge engineering", Addison-Wesley, Reading, Mass., 1987.

Baas, L., and Bourne, J.R.

"A rule based microcomputer system for electroencephalogram evaluation", IEEE Transactions on Biomed. Eng., Vol BME-31, No. 10, pp. 660-664, 1984.

Bourne, J.R., Hamel, B., Giese, D., Woyce, G.M., Lawrence, P.L., Ward,J.W. and Teschan,P.E .

"The EEG analysis system of the national cooperative dialysis study". IEEE Tran. on Biomed. Eng., Vol.BME-27, No 11, pp. 656-664,November 1980a.

Bourne, J.R., Jagannathan, V., Giese, D. and Ward, J.W.

"A software system for the syntactic analysis of the EEG". Comput. Programs in biomed., Vol. 11, pp. 190-200, 1980b.

Bourne, J.R., Jagannathan, V., Hamel, B., Jansen, B.H., Ward,J.W., Hughes, J.R. and Erwin, C.W.

"Evaluation of a syntactic pattern recognition approach to quantitative electroencephalogra phic analysis". Electroenceph. and clin. neurophysiol., 1981, 52, pp. 57-64.

Bourne, J.R., Matousek, M., Friberg, S., and Arvidsson, A.

"SEER-1: The semantic EEG evaluation regimen", IEEE Tran. Biomed. Eng., Vol. BME-30, No. 4, April 1983.

Clocksin, W.F. and Mellish, C.S.

"Programming in prolog". 3rd edition, Springer-Verlag, 1987.

Dowsing, R.D., Rayward-Smith, V.J., Walter, C.D.

"A first course in formal logic and its applications in computer science", Blackwell Scientific Publications, Oxford, UK, 1986.

Forsyth, R. ed.

"Expert systems: principles and case studies". Chapman and Hall
computing, 1984.

Graham, I. and Jones, P. L.

"Expert systems: knowledge uncertainty and decision". Chapman and
Hall, 1988.

Hellyar, M.T., Ifeachor, E. C., Mapps, D.J., Allen, E.M. and Huson N.

"An expert system approach to EEG signal processing", In Press, 1991.

Ifeachor, E.C., Hellyar, M.T., Mapps, D.J., Allen, E.M.

"Intelligent enhancement of EEG signals". IEE colloqium on
"The application of artificial intelligence techniques to signal
processing", London, March, 1989.

Ifeachor, E.C., Hellyar, M.T., Mapps, D.J., Allen, E.M.

"Knowledge based enhancement of EEG signals".IEE Proceedings (Special
Edition), Vol. 137, Pt. F, No. 5, October, 1990.

Jagannathan, V., Bourne, J.R., Giese, D.A., Hamel, B. and Ward, J.W.

"Syntactic EEG analysis: Artifact and drowsiness detection". Proc.
Frontiers of Eng. in Health care, pp. 244-245, October 1979.

Jagannathan, V., Bourne, J.R., Jansen, B.H. and Ward, J.W.

"Artificial intelligence methods in quantitative electroencephalogram
analysis". Comput. Programs in Biomed., Vol. 15, pp. 249-258, 1982.

Jansen, B.H., Bourne, J.R. and Ward, J.W.

"Feature extraction methods for syntactic EEG analysis". Proc. 2nd
Annual IEEE/EMBS Conf., Washington, September 1980.

Jansen, B.H., Bourne, J.R., Jagannathan, V. and Ward, J.W.

"Evaluation of slow-waves in the electroencephalogram using syntactic
shape analysis". Proceedings - Southeastcon., pp. 406-410, 1981a.

Jansen, B.H. Bourne, J.R. Ward, J.W.

"Autoreggressive estimation of short segment spectra for computerised EEG analysis". IEEE Tran. on Biomed. Eng., Vol.BME-28, No 9, pp. 630-638, September 1981b.

Jansen, B.H., Bourne, J.R. and Ward, J.W.

"Identification and labelling of EEG graphic elements using autoregressive spectral estimates". Comput. Biol. Med., Vol. 12, No. 2, pp. 97-106, 1982.

Jansen, B.H.

"Automatic interpretation of electroencephalograms by means of an expert system". Optical Engineering, Vol. 24, No. 6, November/December 1985.

Kernighan, B. W. and Ritchie, D. M.

"The C programming language (second edition)". Prentice Hall software software series, 1988.

Kowalski, R.

"Logic for problem solving", Elsevier science publishing Co., Inc., 1983.

Lemmon, E.J.

"Beginning logic", Van Nostrand Reinhold (UK) Co. Ltd., 1983.

Minsky, M.L.

"A framework for representing knowledge", in: "Mind design", Hangeland (Ed.), M.I.T. Press, 1981.

Newell, A. and Simon, H.A.

"GPS: A program that simulates human thought". in: "Computers and thought", Feigenbaum, E.A. and Feldman, J.A. (Ed.), McGraw Hill, 1963.

Newell, A. and Simon, H.A.

"Human problem solving". Prentice-Hall, 1972.

Rauch-Hinden, W.B.

"Artificial intelligence in business, science and industry".
Prentice-Hall. 1985.

Rowe, N.C.

"Artificial intelligence through PROLOG". Prentice-Hall international,
Englewood Cliffs, N.J., 1988.

Simpson, P.K.

"Artificial neural systems: Foundations, Paradigms, Applications, and
Implementations", Pergamon Press Inc., 1990.

Shortliffe, E.H.

"Computer based medical consultations: MYCIN". American, Elsevier. 1976.

Winston, P. H. and Berthold, K. P. Horn.

"LISP", Addison-Wesley, Reading, MASS., 1981.

Winston, P. H.

"Artificial intelligence (second edition)". Addison-Wesley, Reading,
Mass., 1984.

CHAPTER 3

EXPERIMENTAL APPARATUS

3.1 INTRODUCTION.

The design of an intelligent system for the removal of OAs from the EEG requires that EEG and EOG data is acquired from a large sample of patients. This will provide data from a wide variety of cerebral and eye movement activity to enable the compilation of a comprehensive EEG/OA database. Intelligent strategies to overcome the deficiencies detailed in chapter 2 will be formulated using the information contained in the database, information elicited from consultation with EEG experts and appropriate data analysis techniques. The development of a suitable EEG data acquisition system (DAS) was therefore crucial to this investigation. In addition to the acquisition of regular EEG recording data, the DAS would also provide the necessary hardware interface required between an eventual intelligent OA removal system and the conventional EEG recording equipment. The following section describes the specifications for the DAS and the details of its design.

3.2 DESIGN SPECIFICATIONS.

The following specifications were required of the EEG DAS.

. The system must be compatible with standard EEG machines and be able

to acquire data from the auxiliary output of a standard EEG machine.

The system must be capable of collecting data continuously from 16 EEG/EOG channels for a maximum of 30 minutes.

. The system must cause minimal distortion to data signals in the frequency range of 0 to 30 Hz.

. Data must be archived for future reference in a convenient and easily accessible form.

. The system must be compact and unobtrusive for the clinical environment and present the minimal disturbance to patients.

. The system must be easily operated by an unskilled person.

. The cost of the system should be kept to a minimum.

. The system must conform to hospital health and safety conditions.

The above requirements are elegantly met by using a standard Personal Computer (PC) with additional specialised data acquisition hardware. The PC is responsible for master control via a suitable user interface and the storage of EEG/EOG data on Winchester disk. The additional hardware is responsible for continuous data acquisition, consisting of multi-channel input signal conditioning, analogue to digital conversion, short term data buffering and communication with the controlling PC.

Careful consideration was made to minimise signal distortion in the additional hardware. Particular attention was paid to the analogue input signal conditioning and the analogue to digital conversion. These being the two areas where signal distortion could easily occur.

Signal distortion in the analogue input signal conditioning is related to the quality of design, and component layout. High quality components and low distortion designs were used in this area. Signal distortion in the analogue to digital conversion process is directly related to the choice of sampling frequency in the digital system. Section 1.1 illustrated that the conventional EEG frequency bands ranged from 0.5Hz to 30Hz. The sampling frequency was chosen as a compromise of minimimal aliasing error, and simplified filter design and data storage.

3.3 AN EEG DATA ACQUISITION SYSTEM (DAS).

This section details the design of the EEG DAS used in this investigation. The system was installed in the department of clinical neurophysiology at Derriford hospital in Plymouth approximately 2 years ago and has been available for data acquisition for this period.

The DAS is positioned alongside conventional EEG recording equipment in the clinical environment and regular data acquisition is carried out by hospital technical staff. Plate 3.1 illustrates the DAS and plate 3.2 illustrates the utilisation of the DAS in the clinical environment.
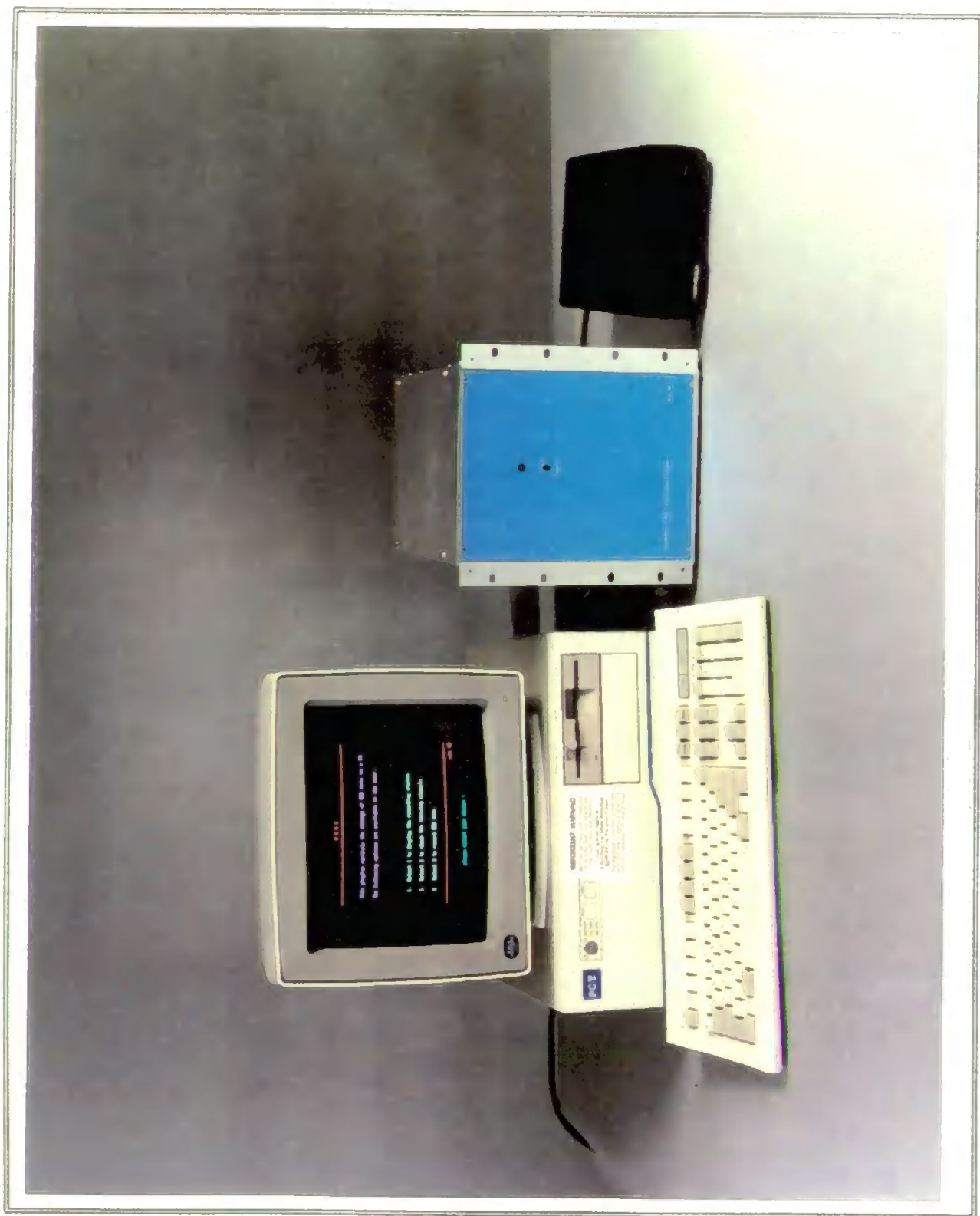
Plate 3.1 The Data acquisition system.

Plate 3.2 The DAS in use in the clinical environment.

3.3.1 SYSTEM OVERVIEW.

Figure 3.1 illustrates a conceptual diagram of the DAS. The system consists of three units. The first unit incorporates the analogue signal conditioning, analogue to digital conversion and digital control/data processing. The second unit is a standard personal computer (PC) with interfaces to the other units. The third unit is a standard Winchester disk backup tape streamer used for data archiving. Connection to the EEG machine is made via the first unit.
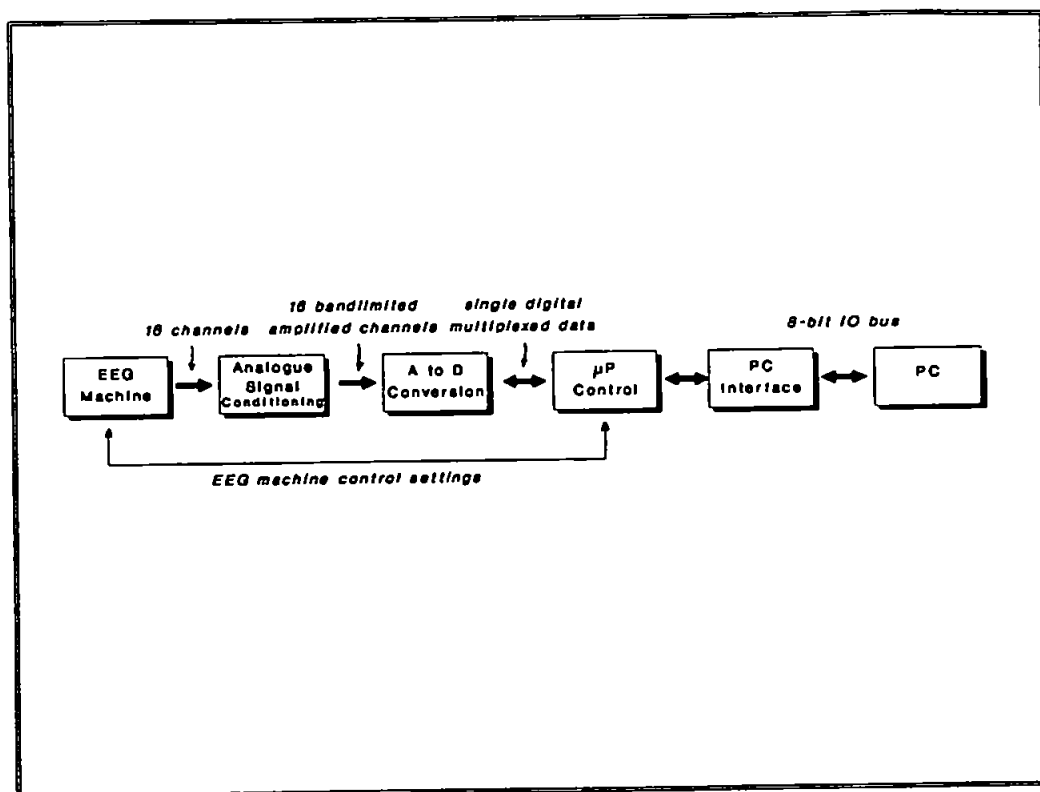
Figure 3.1 Conceptual illustration of the DAS.

The PC provides the main control features for the system. This includes the user interface which allows data acquisition to be started, stopped or paused at any time. The graphics facilities are utilised for real-time EEG/EOG signal display, both during recording, and as a previewing facility. The PC also provides temporary data storage on Winchester disk. Data archiving is provided by a standard backup tape streamer and conventional paper trace record. All three units are mounted on a trolley so that the system is easily transportable between EEG examination rooms.

At the start of each patient recording the equipment is turned on. The PC keeps a permanent record of data already on the Winchester disk so that data is not over-written after power is removed. On power up, devices are reset and the system will perform a short self examination of connections and disk space, reporting on any malfunctions. After entering a few patient details using the PC keyboard, the control of the data acquisition is all carried out using the PC keyboard space bar. During recording an index number is displayed on the PC screen. This enables the recording staff to mark the paper trace enabling easier cross referencing. Modern EEG machines, such as the Siemens Mingograph EEG21 provide an additional computer interface. This removes the need for manual cross referencing and allows the DAS to elicit, directly from the EEG machine, the montage configuration, and values of amplifier gain and filter settings.

3.3.2 ANALOGUE SIGNAL CONDITIONING.

The analogue signal conditioning, for each channel, consists of an instrumentation amplifier (IA), a band-limiting filter, and a sample and hold amplifier. The necessity to have 16 of these channels in the one unit meant that careful consideration had to be given to chip count and physical layout.

(a) Instrumentation ampliers.

Instrumentation amplifiers were chosen to minimise common mode signals and were designed around the common 3 operational amplifier configuration *[Riskin, 1984; Meiksin and Thackray, 1984]*. Monolithic components would have decreased the chip count but proved to be cost prohibitive. OP07 bipolar operational amplifier were used in the construction because of their high linearity, low input offset and high common mode rejection ratio (CMRR) *[Rutkowski, 1984]*. The auxiliary output of the EEG machine delivers a maximum of +/- 1.4 volts and therefore the gain of the IAs was set to 6 in order to utilise the full dynamic range of the analogue to digital converter circuits (see later). Appendix B contains the instrumentation amplifier circuit diagram.

(b) Bandlimiting filters.

Signals were bandlimited using a second order Butterworth low pass filter. The Butterworth filter was chosen because it has maximally flat passband, a reasonably good phase performance and only uses 1 operational

amplifier in it's construction *[Lynn, 1982]*. The disadvantages of this design were the relatively poor roll off rate of the filter (-40dB/decade) and a gain loss at the cut off frequency. In order to overcome these, the cut off frequency was chosen to be 40 Hz as opposed to the highest frequency of interest which was 30Hz and the sampling rate was chosen to be 256 Hz. This gives an aliasing error of approximately 4.8 % at 40 Hz *[Zuch, 1983(a)]*. Appendix B contains the bandlimiting filter circuit diagram.

(c) Sample and hold.

Simultaneous sampling was chosen to avoid the introduction of delays between corresponding time points. Monolithic sample and hold amplifiers have similar cost to discrete component ones and reduce chip count. Therefore, an LF398 sample and hold amplifier was used in each channel *[National; Chen, 1980; Zuch, (b)]*. Sample hold control signals were derived from the microprocessor control circuit (see later). The system is not operated at high frequencies, therefore, the charge holding characteristics are of more importance than the speed of the device. Charge was held using a polypropylene capacitor. This exhibits a high insulating resistance and a very low dielectric absorption. A 0.01 uF capacitor allows an acquisition time of 25 µS to 0.01% of target voltage and a final sag in the order of 2mV in 10V in a period of 3.9mS. This gives more then adequate performance with a final error in hold voltage of less than 0.03%, considerably less then half the quantisation level of a 12bit analogue to digital converter.

3.3.3 ANALOGUE TO DIGITAL CONVERSION

The resolution selected for the DAS was of 12 bits. This complies with common procedures in medical applications. A single SAR analogue to digital converter (ADC) integrated circuit (AD574) is used in conjunction with a multiplexer integrated circuit (AD7506). This minimises chip count and because of the use of multiple sample and hold amplifiers, does not introduce channel phase errors. Analogue signals are amplified to +/- 10v to utilise the full dynamic range of the ADC. The control of the signal sampling, mutiplexing, and conversion were carried out by the microprocessor control circuit. A software program carried out the sequencing of these events. Appendix B contains a circuit diagram of the multiplexing and conversion circuit.

3.2.4 MICROPROCESSOR CONTROL CIRCUIT

The Motorola 10 MHz MC68000 was chosen as the microprocessor (μP) used in this system. This was chosen because of the speed of the processor, the 16 Mbyte addressable memory space, the cost and the availability of software development tools. The MC68000 is also an industry standard and considerable experience exists in its use (Motorola, 1987; Wilcox, 1987). Figure 3.2 illustrates a block diagram of the microprocessor control circuit. The microprocessor control circuit is comprised of the microprocessor, the timing clock, the reset and watchdog timer circuits, the system random access memory (RAM), the system read only memory (EPROM), the programmable timer module (PTM), and the dual port ram (DPR).

MC68000 Clock Reset and Watchdod Timer PTM Address Decoder and DTACK PC Interface

Address bus

Data bus

EPROM RAM A to D Conversion DRB
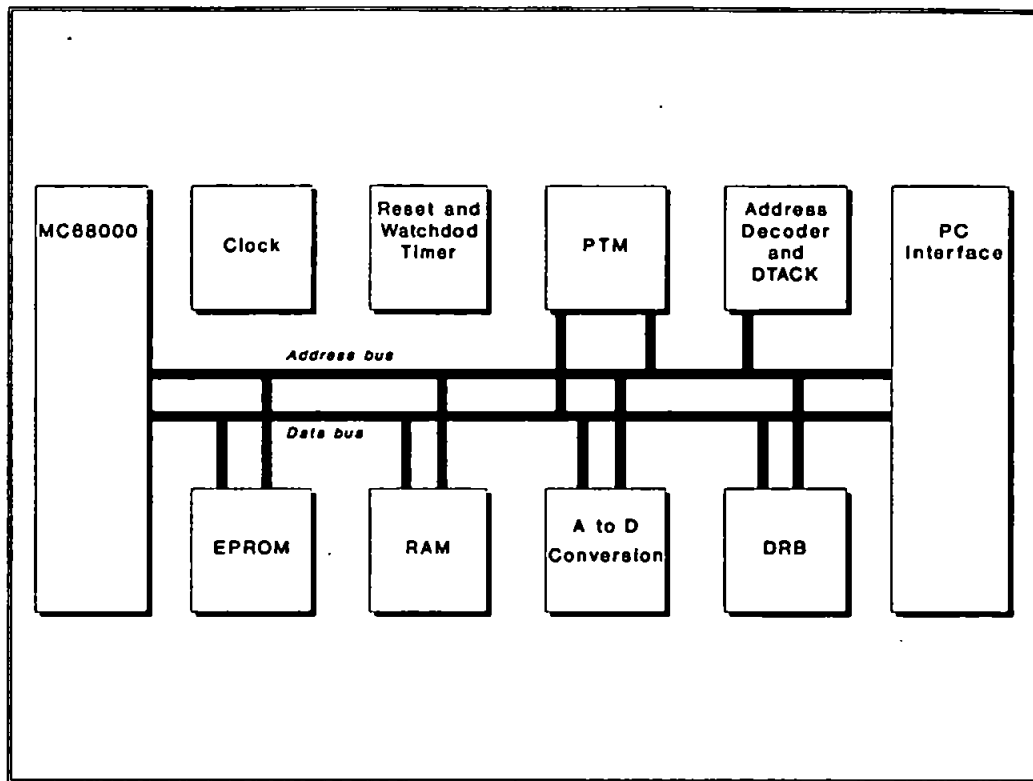
Figure 3.2 Microprocessor control circuit block diagram.

Each device in the DAS is memory mapped into the 16 Mbyte address space and device's address is only partially decoded, giving each device 1 Mbyte of address space.

The reset circuit provides a hard reset to the 68000 on power up and under the control of a front panel reset button. The watchdog timer monitors data hand-shaking within the 68000 system and initiates a reset should failure occur, preventing possible system hang ups. The decoder and DTACK generator circuits provide the necessary chip enable control lines and data acknowledge hand-shaking signals required by the 68000. EPROM and RAM provide the conventional memory areas for program code and data

respectively. The PTM is responsible for providing sample hold signal timing and is programmable during operation. The PTM sample hold signal generates a prioritised interrupt to the 68000 for every sample from the sample hold signal. The detected interrupt is serviced by the 68000 by jumping to a special software routine which is responsible for control of the analogue to digital converter circuit and data storage in the DPR. The DPR is a temporary buffer necessary to ensure continuous data acquisition and is addressable by both the DAS and the PC controlled by two latches. Data storage to Winchester disk is via two 64k RAM buffers which are each filled in approximately 8 seconds when the DAS operates at a sampling frequency of 256Hz. When one buffer is full, further data storage is transferred to the second buffer and the DAS instructs the PC to read and store the contents of the first buffer onto Winchester disk. This is carried out in approximately 2 seconds enabling the DAS to operate upto a maximum sampling frequency of 1024Hz. This process is repeated when the next buffer is full. At any time the DAS will be connected to read from one buffer and write to the other and the PC will have the reverse connections. The control of the latches which enable this communication is carried out by the 68000. Appendix B contains the microprocessor control circuit diagram.

3.3.5 PC INTERFACE.

The PC interface provides the communication link between the PC I/O address space and the DAS processor board. The interface is positioned in one expansion slot inside the PC. Both control latches and the DPR on the

DAS processor board are addressable by the PC through the IO addresses 300h and 301h. To read or write to a latch or DRP buffer the address is first written to 300h which enables decoding to produce the relevant chip enable ($\overline{CE}$) control lines. This is followed by a single byte read or write to or from that address. Appendix B contains the PC interface circuit diagram.

3.3.6 DATA STORAGE

The contents of each DPR buffer, containing 8 seconds of 16 channel EEG/EOG data, is saved to Winchester disk as a separate binary disk file. Files are sequentially numbered and each file index reflects the patient number and data block number. Files are automatically attributed date and time values by the Disc operating system (DOS). In addition to the signal data, a header file is stored for each patient that contains the patients age, sex, initials and any relevant information regarding the patient. For example a known history of epilepsy could be of importance.

When an examination is completed, patient data are archived using a standard backup tape streamer. The current system uses an Everex 40 Mbyte cassette tape steamer. This enables a 40 Mbyte Winchester disk to be backed up in approximately 10 minutes. A CT600H compact cassette backup tape will therefore hold the contents of between 3 and 5 patients data, depending on the length of the examination.

3.3.7 SOFTWARE.

The software used with the DAS is divided under two sections, (i) the control software, and (ii) the support data software. The control software consists of the system program, DAS3, and the PC control software, PCDSP14. The support data software consists of a number of data editing and manipulation programs used for data analysis, VIEW programs, (see appendix C).

3.3.7.1 DAS SYSTEM SOFTWARE.

The system software is written in 68000 assembly code, and is responsible for coordinating the data acquisition and storage routines under a supervising main program *[Bramer, 1986]*. Figure 3.3 illustrates a flow diagram for this program.

start

system
test

read PC
status latch

no start?

yes

enable
Interrupts

read PC
status latch

no stop?

yes

disable
Interrupts

end

Interrupt

chan=1
read
buffer No

convert
chan

store
data

Chan=1

no =16?

yes

no buffer
full

yes

write to PC
status latch

end of Interrupt
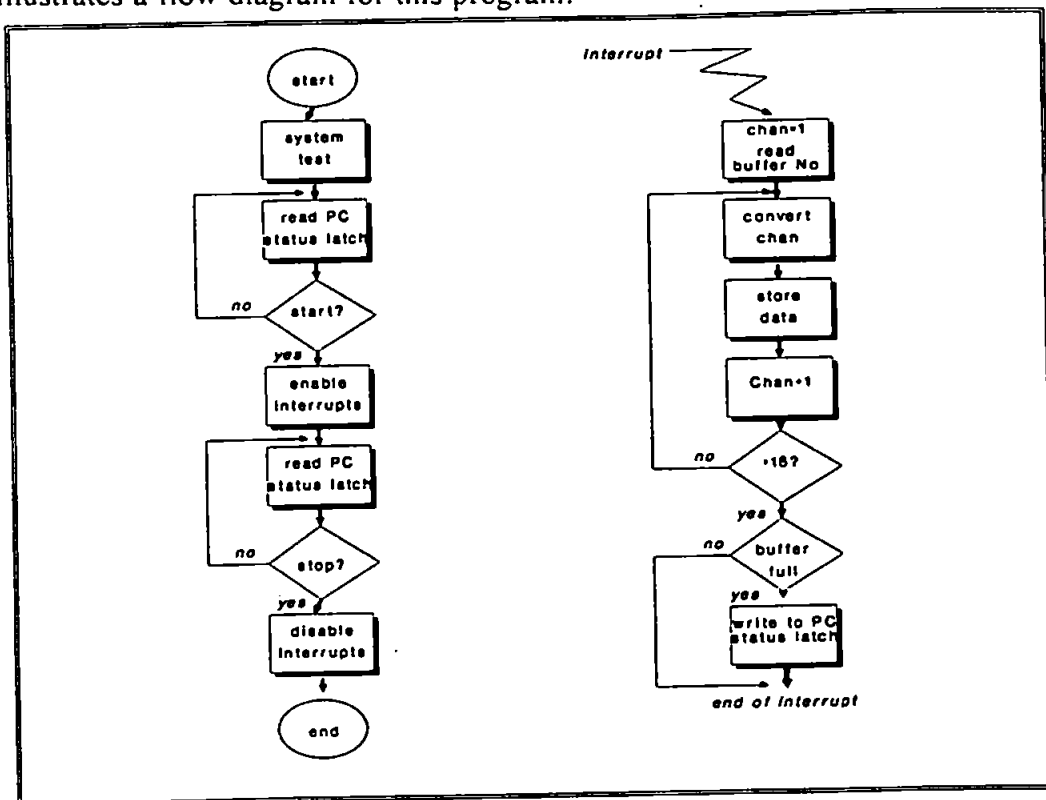
Figure 3.3 DAS system software flow diagram.
page 82

When powered up, the system software will run a short test on its status. Data acquisition is commenced, by enabling interrupts, when the system software detects the command word, in a latch written to by the PC. The DAS system is interrupt driven, the interrupt signal is derived from the PTM on the microprocessor board, which operates at a frequency of 256 Hz. This signal provides the sample and hold circuits with the hold command and also interrupts the microprocessor with a level 4 interrupt. During the interrupt service routine the uP selects each channel by writing the appropriate address to the MUX. Conversion is initiated by writing to the ADC. When the end of conversion signal is detected, the converted signal can be read and the next conversion initiated using one read of the ADC. Each 12-bit conversion is read as 16 bits this enables the remaining 4 bits to be used to indicate status conditions, such as breaks in data acquisition caused when a pause is requested. 16-bit words are stored in the DRB as 2 consecutive 8-bit words. This allows easy access for PCs with only 8-bit I/O ports. Once 64 Kbytes of samples have been acquired, every 8 seconds, the system software writes a status word to a latch which is read by the PC. Subsequent samples are written to the other 64 Kbyte RAM buffer in the DRB. The supervising program monitors the status of the PC to ensure that the RAM buffer has been read before it is written to and also to detect when data acquisition is to cease.

3.3.7.2 PC CONTROL SOFTWARE.

The PC control software is written in a Compiled BASIC with the data communication routines written in 8086 assembly code [IBM 1984; Scanlow, 1985]. Figure 3.4 illustrates a flow diagram of this program.
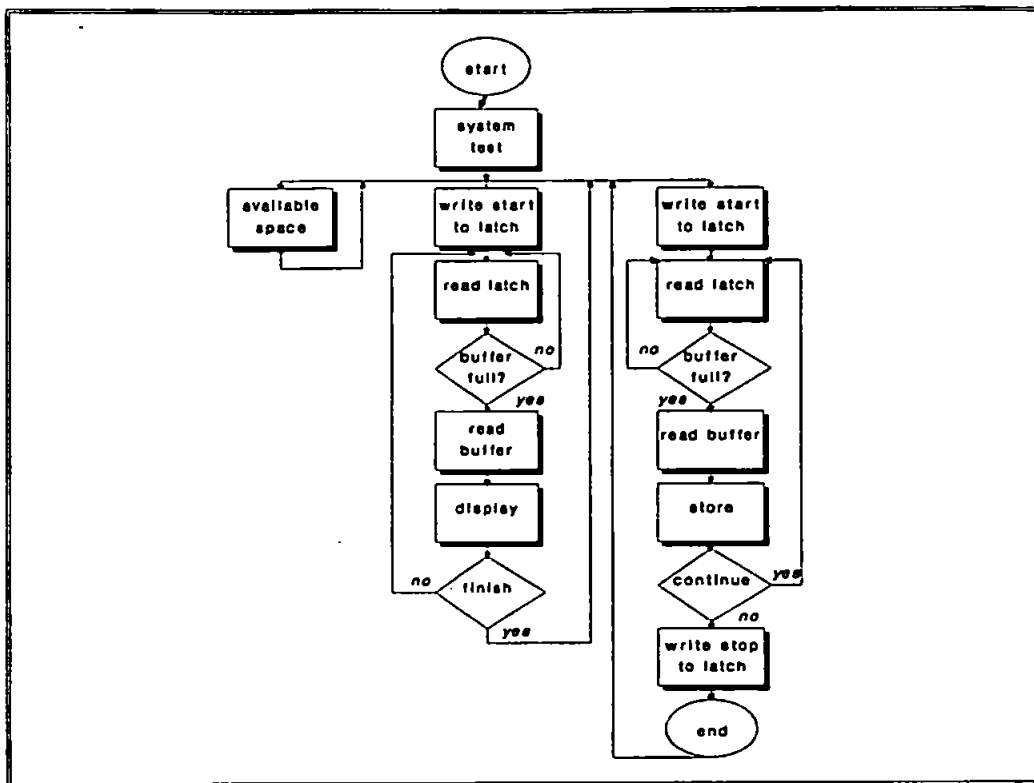
Figure 3.4 PC control sofware flow diagram.

The program consists of the user interface, data storage, data archiving and supervising routines. Compiled BASIC was chosen because of the ease with which a user friendly graphics interface could be written. Compiled BASIC is also considerably faster than its interpreted version and code can be written in an easily understood and structured manner. Assembly language routines were however necessary where speed was critical.

When the system is powered up the PC software tests the status of the system. Problems with the DAS will be reported on the screen and will normally request for the system to be reset using the button on the front

of the DAS. Provided this is successful the user will be presented with a menu .screen (see plate 3.2), from which 3 choices are available, (i) a status check on the available disk space, (ii) a preview of the data to be read, and (iii) to record data.

It was vital to include features such as start, stop and pause into a user interface that imposed as little as possible on the operator. Therefore, control of the program is achieved by selecting an option from a menu using a single key press However, the user interface also enables the operator to enter patient data such as initials, age, sex and any other pertinent details via the normal alphanumeric keys. This is used in order to make patient data identification easier as well as providing useful information. This additional data are stored as a separate ASCII file with the sampled data.

When record data are selected the software writes to a latch read by the DAS. This initiates recording and the PC software then monitors the status of a latch written to by the DAS to detect when a RAM buffer is full. Once detected the PC software reads the data into memory using an assembly coded routine, opens a new file on the Winchester and writes the data to disk as one 64 Kbyte block of binary data. This was found to be the fastest method of storage. Once data are written the PC informs the DAS by writing to the latch read by the DAS.

3.3.7.3 SUPPORT SOFTWARE.

The support data software programs were also written in compiled BASIC. Data editing is carried out off-line. Therefore, speed was less critical than good graphics and ease of code development. Programs in the VIEW suite (see appendix C) allow the user to edit 2,4 or 8 second segments from single or multiple recorded channels. The programs allow rapid scanning through recorded data by displaying multiple channels on the graphics display. The display is also used to identify data that are being viewed, so that recorded data and paper records are more easily synchronised. Breaks in data recording are indicated on the display as vertical bars through the data. Edited section of data are saved onto disk for separate channels as ASCII files. This allows easy transportation to the various analysis programs used later.

3.3.8 HARDWARE CONSTRUCTION

Careful consideration needed to be given on the construction of the DAS. Not only did the system have to conform to clinical safety conditions, but also the system needed to be robust, introduce minimal noise and be compact for the working environment. This section details the major aspects concerning the construction of the DAS.

The ADC circuit board was constructed on a double sided single Euro size PCB and mounted in a separate metal area within the DAS enclosure to allow electrical screening of the critical components. The microprocessor

control circuit was constructed on a double sided, double, extended EURO PCB. The use of a separate PCB enabled the high frequency digital components to be physically and electrically isolated from the critical analogue to digital conversion circuits.

Careful consideration had to be given to circuit layout and grounding procedures, to minimise signal distortion. The presence of both analogue and digital grounds on the circuit boards further added to this need. Each channel has its own signal ground, there is also a power supply ground and a digital ground for each board. These grounds are all connected to a star point at the power supply *[Brokaw, 1984; Rich, 1983]*. The use of a separate PCB for analogue to digital conversion components enabled better control over the critical circuit layout and grounding requirements of these circuits *[Bradshaw and Osgood, 1984]*. Analogue and digital components are separated on the PCB. Separate, multiple, analogue and digital grounds are provided as well as power supply grounds for decoupling components. These are all taken off the board and connected together at a star point at the power supply. Analogue and digital grounds are also connected close to the ADC to provide a reference for conversion. An analogue ground plane surrounds all the components on the ADC PCB.

3.4 SUMMARY OF CHAPTER 3.

Chapter 3 has detailed the specifications and design of a high quality and inexpensive data acquisition system suitable for interface to standard EEG measurement equipment. Use is made of a standard PC interface to allow

for data archiving using Winchester disk and explanatory user interface. Data acquisition has allowed an extensive database of EEG/EOG signals to be compiled and the developed acquisition system forms the hardware basis of a proposed intelligent OA removal system.

REFERENCES FOR CHAPTER 3.

Bradshaw, P. and Osgood, S.

"Do's and dont's of applying A/D converters". Application notes.
Intersil Inc.

Bramer, B.

"MC68000 Assembly language programming". Edward Arnold Ltd, 1986.

Brokaw, P.

"An IC amplifier users guide to decoupling grounding and making
things go right for a change". Analog Devices application note,
Vol. 1, pp. 21-31.

Brook, D. and Wynne, R. J.

"Signal processing. Principles and applications". Edward Arnold,
1988.

Chen, C.

"Introduction to the sampling theorem". National semiconductor
application note AN236, pp. 12.46-12.57, 1980

IBM personal computer XT technical reference manual. 1984.

Lynn, P.A.

"The analysis and processing of signals". 2nd ed., Macmillan Ltd,
1982.

Motorolla MC68000 Microprocessor users manual

Murray, W.H., and Pappas, C.H.

"80386/80286 Assembly language programming", Osbourne McGraw-Hill,
Berkley, Cal., 1986.

National semiconductor application notes.

"Monolithic sample and hold circuits". pp. 8.1-8.9, 1984.

Rich, A.

"How to exclude interference-type noise. What to do and why to do it - a rational approach". Analog Devices application notes, Vol. 1, pp. 20-85, 1983.

Riskin, J.R.

"A user's guide to IC Instrumentation amplifiers". Analog Devices Application notes, Vol. 1, pp. 21-30, 1984.

Rutkowski, G.B.

"Integrated circuit operational amplifiers". 2nd ed., Prentice-Hall Inc., 1984.

Scanlow, L.J.

"IBM PC and XT Assembly language, a guide for programmers". Brady Communications Company Inc., 1985.

Wilcox, A.D.

"MC68000 microprocessor system designing and troubleshooting". Prentice-Hall, 1987.

Zuch, E.

"Designing with sample and hold won't be a problem if you use the right circuit". Data acquisition and conversion handbook, Datel Intersil, pp. 155-160, 1983(a).

Zuch, E.

"Principles of data acquisition and conversion". Data acquisition and conversion handbook, Datel Intersil inc, 1983(b).

CHAPTER 4

DATA ACQUISITION AND ANALYSIS

4.1 INTRODUCTION

The equipment described in the previous section has been used in order to create a EEG database. The objective of this is to acquire data on (i) all types of ocular artefacts and (ii) cerebral signals whose interpretation is made difficult by the presence of OAs. A large and comprehensive database, has enabled the production of new intelligent OA removal strategies to be developed and tested. Section 4.2 describes the techniques used to acquire patient data, together with illustrative examples of the data obtained. Sections 4.3 - 4.5 detail the development of the new intelligent OA removal stategies. Section 4.3 extends previous research to establish suitable EOG subtraction models for OA removal from different EEG signal derivations. Section 4.4 investigates the application of the OA removal algorithm in a selective and directed manner. Selective, depending on the type of OA present, and directed to avoid applying the algorithm to uncontaminated EEG. Section 4.5 investigates the characterisation of OA and abnormal slow waveforms, to establish techniques to enable their differentiation.

4.2 EXPERIMENTAL TECHNIQUES AND DATA.

Patient data has been acquired from a selection of both normal and abnormal patient populations. This has enabled a wide variety of OA waveforms to be obtained. Normal patient data was acquired from volunteers, whilst abnormal patient data was acquired from patients normally admitted to the EEG department.

4.2.1 DATABASE ORGANISATION.

Patient data are stored on compact cassette backup tapes and as standard paper trace records. Each patients data consists of a header file and a number of data files. The header file contains the date and the age, sex and initials of the patient. Pertinent details, such as "prior signs of frontal slow waves", etc. can also be stored in this file. All patient data are governed by rules of confidentiality. To enable sections of data to be more easily referenced to paper records, the data files are divided into sequentially numbered 8-second blocks and are automatically time and date stamped. In addition to the digital and paper data records, technicians reports on patient data are stored. Each technicians report consists of general observations on the patients behaviour during the recording and also technical details of the records content. General observations include the patients state of awareness and whether eyes are open or closed etc., and technical details include a catalogue of the types of waveforms seen to be present. For example, a typical technical report will consist of:

Patient was conscious and alert throughout.

Regular alpha in the occipital region, upto $50 \mu V$...

The technicians report and paper record allowed relevant sections of EEG to be selected for further analysis. This was of particular importance when attempting to elicit knowledge from the EEG expert regarding specific instances of waveforms.

4.2.2 PRE-CLINICAL VOLUNTEER DATA ACQUISITION.

Pre-clinical testing of the equipment was required for a number of reasons. Firstly the DAS could be subjected to rigorous examination to enable the reliability of the system to be tested in the working environment. This was likely to be easier using volunteers than by using patients who were unlikely to be entirely cooperative. Secondly, data acquisition using volunteers would provide eye movement data from a normal or non-patient group. This data would contain no abnormal waveforms and therefore enable easier OA identification. Characteristics obtained from the volunteer OAs would be used as parameters for OA identification in the patient group data. Pre-clinical trials were conducted at Derriford Hospital in Plymouth, using 3 volunteers. Data acquired from volunteers has enabled a comparison to be made between EOG subtraction models and different OA removal strategies. In addition, these data have provided information on other EEG artefacts, such as muscle artefact caused by head, neck, jaw and face movement.

The pre-clinical trails consisted of volunteers making a series of repetitive eye/eyelid and facial movements over controlled excursions. Figure 4.1 details the pre-clinical trial protocol. Volunteers were positioned approximately 2m in front of a plain wall. The wall was marked with 8 points constructing a circle of radius 1.15m, the centre of which was in line with the volunteers eyes. The technician used a pointer with a bright spot on one end to point to the appropriate points on the circle. The volunteer was asked to maintain head position and to follow this point. Vertical eye movements (VEM), horizontal eye movements (HEM) and eye rolling movements (REM) were therefore measured over 60 degree movements, or 30 degrees from the rest position. Each set of movements lasted approximately 2 minutes and the whole test approximately 30 minutes.

Figure 4.2 illustrates the electrode positions used in the trials. 9 EEG signals were measured and 7 additional EOG signals. 4 of the EOG signals were also EEG signals so that in total 13 EEG signals were used. All signals were measured referred to linked mastoids (A1-A2). From these referential signals all frontal EEG signals and EOG signals could be derived. Figures 4.3 - 4.5 illustrate typical EEG recorded from a number of referential EEG signals. Figure 4.3 illustrates EEG contamination by vertical eye movement (VEM) OA. Figure 4.4 illustrates EEG contamination by horizontal eye movement (HEM) OA. Figure 4.4 illustrates EEG contamination by blink OA.

VOLUNTEER PRE-CLINICAL TRIAL PROTOCOL

Volunteers should be seated and positioned over mark on floor, facing the wall. Volunteers should be asked to maintain head position and to follow the pointer with their eyes only.

tests:

1. 20 seconds of data is acquired while the patient is at rest.

2. VEM: pointer to move to 6 o'clock and then to 12 o'clock from central position. 10 repetitions. Each movement to last approx 1 second.

3. 5 second pause.

4. HEM: pointer to move to 9 o'clock and then 3 o'clock from central position. 10 repetitions. Each movement to last approx 1 second.

5. 5 second pause.

6. DEM: pointer to move to 2 o'clock and then 8 o'clock from central position. 10 repetitions. Each movement to last approx 1 second.

7. 5 second pause.

8. REM: pointer to travel around circle clockwise. 10 repetitions. Each rotation to last approx 4 seconds.

9. 5 second pause.

10. Blinking: Blink 10 times at approx 1 second intervals.

11. 5 second pause.

12. Nystagmus: text is positioned 0.25m in front of volunteer at eye level. Volunteer is asked to read text.

13. 5 second pause.

14. Head movement: Volunteer should move head to look at four cornares of room. Repeat twice.

15. 5 second pause.

16. Jaw movement: Volunteer should chew teeth for approx 10 seconds.

17. 5 second pause.

18. Facial movement: Volunteer should raise eyebrows and move face for approx 10 seconds.

Figure 4.1 Pre-clinical volunteer trial protocol.
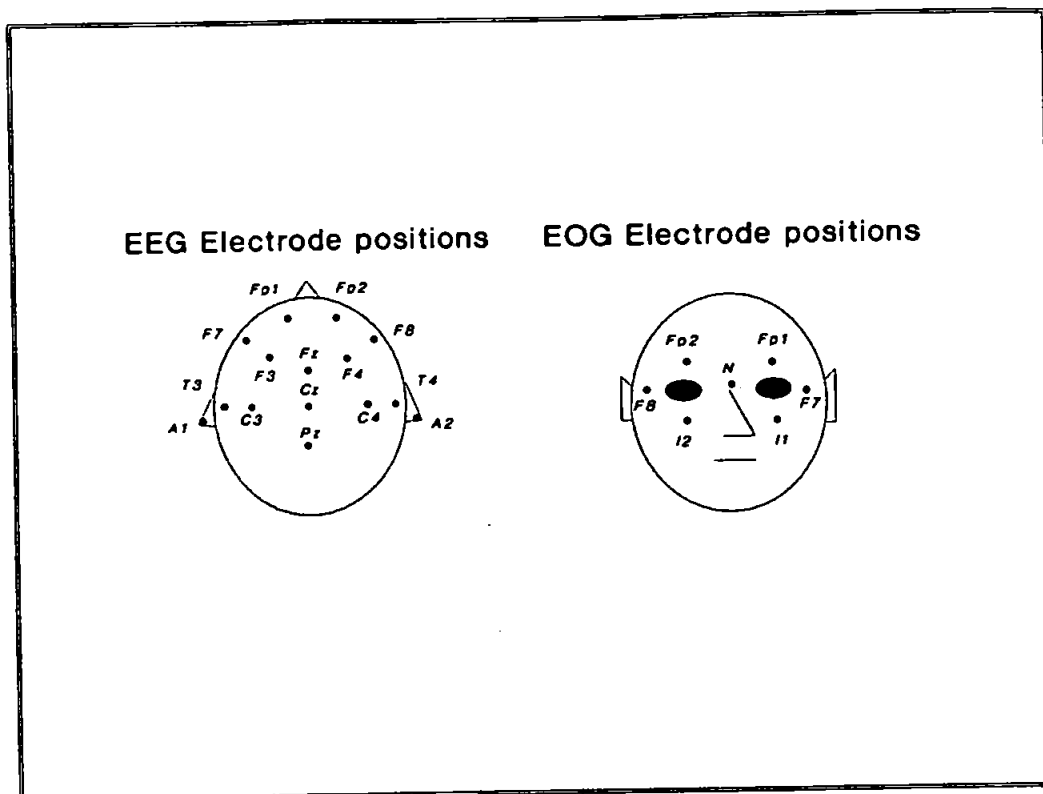
Figure 4.2 EEG and EOG electrode positions used in the

pre-clinical volunteer trials.
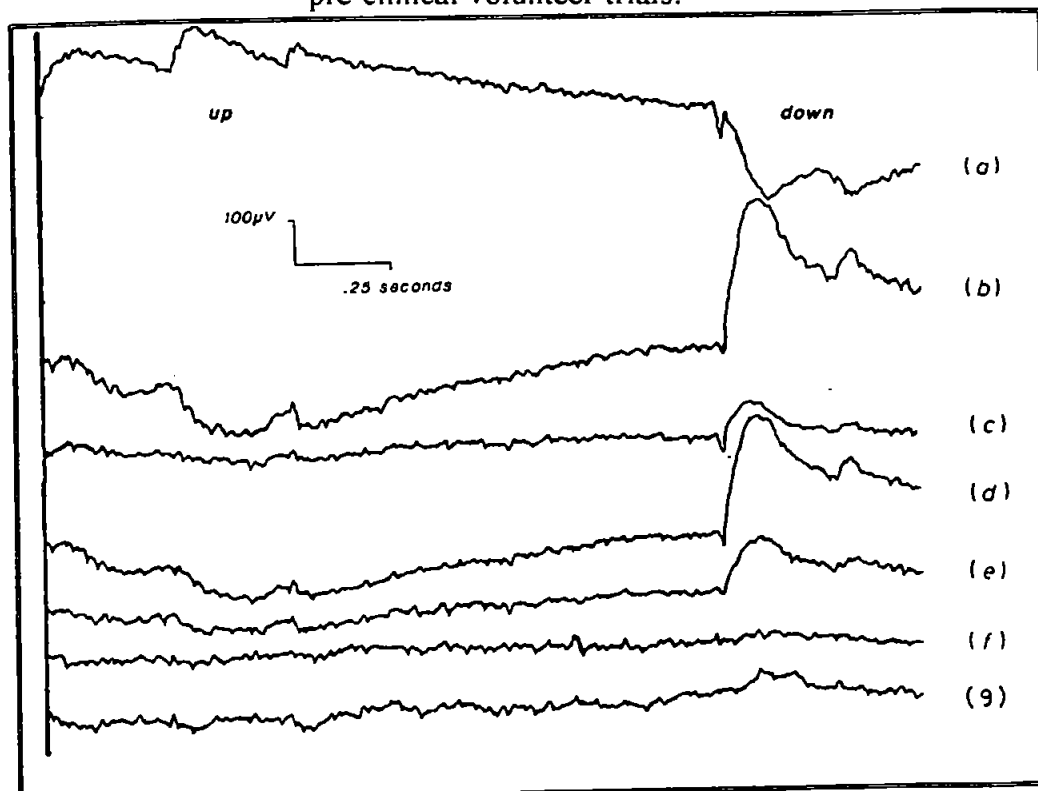
Figure 4.3 EEG and EOG signals illustrating typical Vertical eye

movement (VEM): (a)I2, (b)Fp2, (c)F8, (d)N, (e)F4, (f)T4, (g)Pz.

page 96
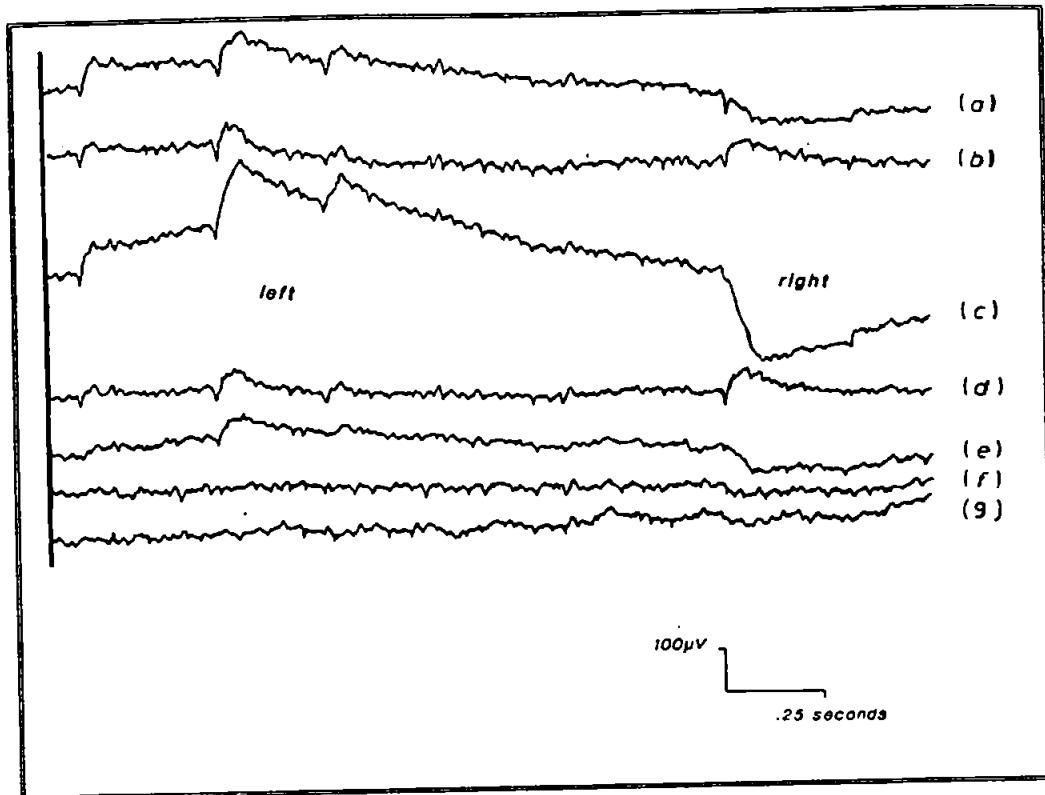
Figure 4.4 EEG and EOG signals illustrating typical Horizontal eye

movement (HEM): (a)I2, (b)Fp2, (c)F8, (d)N, (e)F4, (f)T4, (g)Pz.
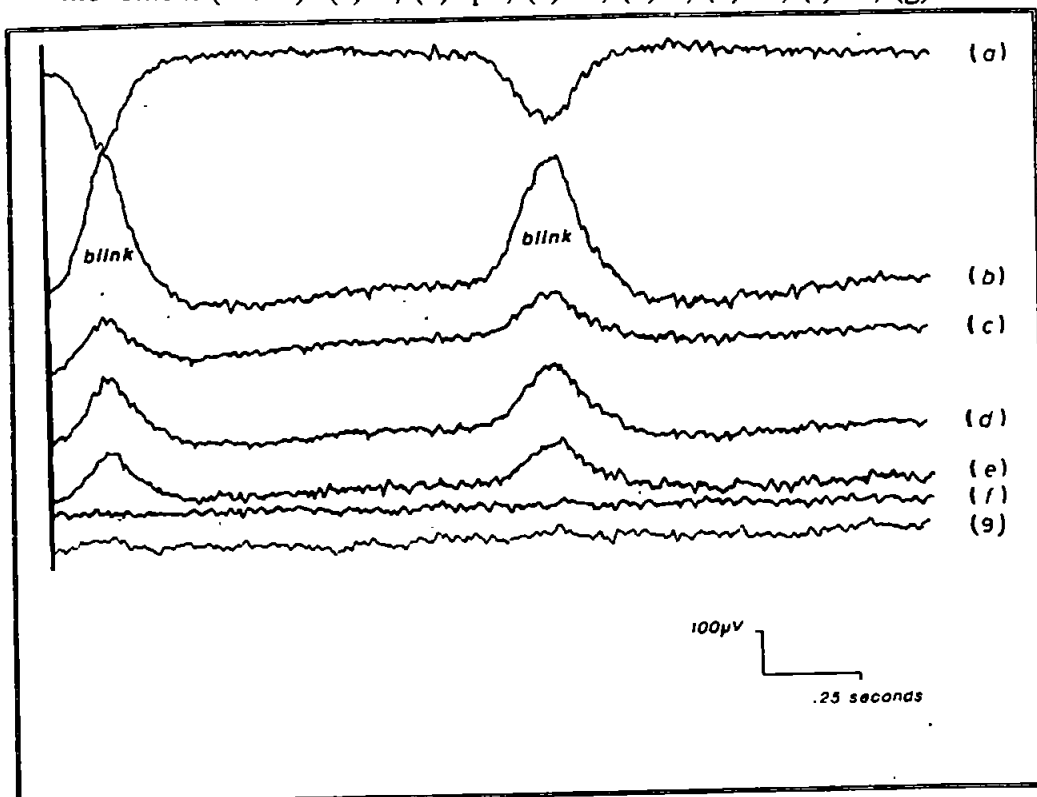
Figure 4.5 EEG and EOG signals illustrating

typical Blink: (a)I2, (b)Fp2, (c)F8, (d)N, (e)F4, (f)T4, (g)Pz.

4.2.3 CLINICAL PATIENT DATA ACQUISITION.

Clinical patient data acquisition was carried out on patients admitted to the department of Neurophysiology at Derriford Hospital. It has provided EEG/EOG data from 50 patients in all age groups containing a variety of abnormalities. Standard EEG examinations procedure was followed during the recording and no special electrode positions were used. Each recording typically consists of the measurement of cerebral activity using upto 8 electrode montages, which are shown in figure 4.6. Each recording typically lasted 20-30 minutes, during which time the patient was asked to open or close eyes and be subjected to a number of evokative procedures such as stroboscopic light. The specific recording protocol will vary according to the observations made by the recording technician. However, EEG montages 1, 2, 3, 4, and 8 are used as standard in the acquired patient data.

Figures 4.7 - 4.10 illustrate a sample of the data acquired from abnormal patient data acquisition. The samples illustrate common OA and abnormal frontal slow wave observed in the data. Each sample is accompanied with a summary of the technicians report for the particular patient. Appendix D contains the technicians reports for all acquired patient data for further reference. Figures 4.7 - 4.10 represent multi-channel EEG (montage 1) showing high amplitude frontal delta activity. Figures 4.7 and 4.8 illustrate common OA waveforms , and figures 4.9 and 4.10 illustrate common abnormal frontal slow waveforms. The VEM illustrated in figure 4.8 appears simmilar to the abnormal frontal delta waveforms illustrated in figure 4.10. However, the potential distribution is clearly different.
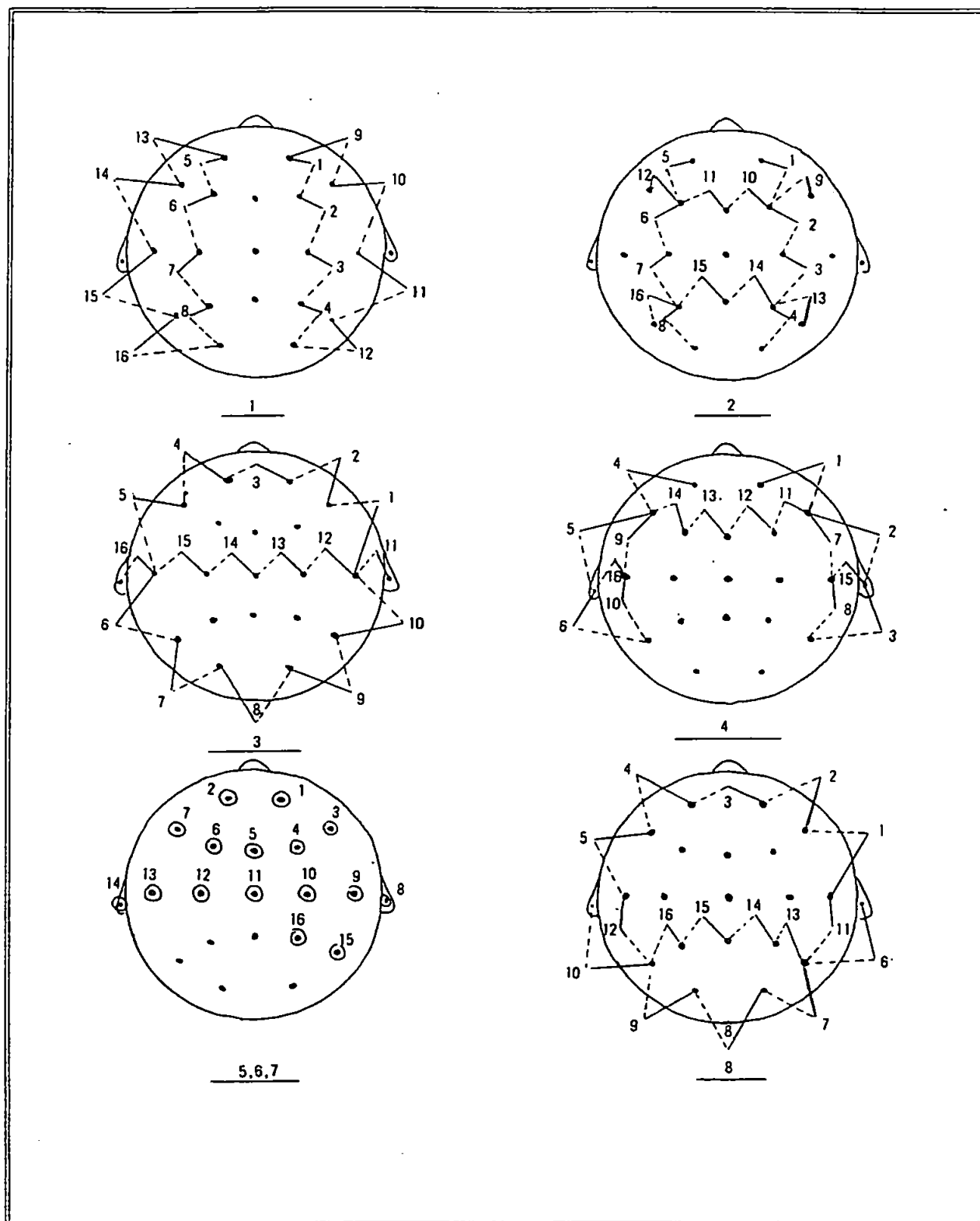
Figure 4.6 EEG electrode recording montages

used in clinical trials.

page 99
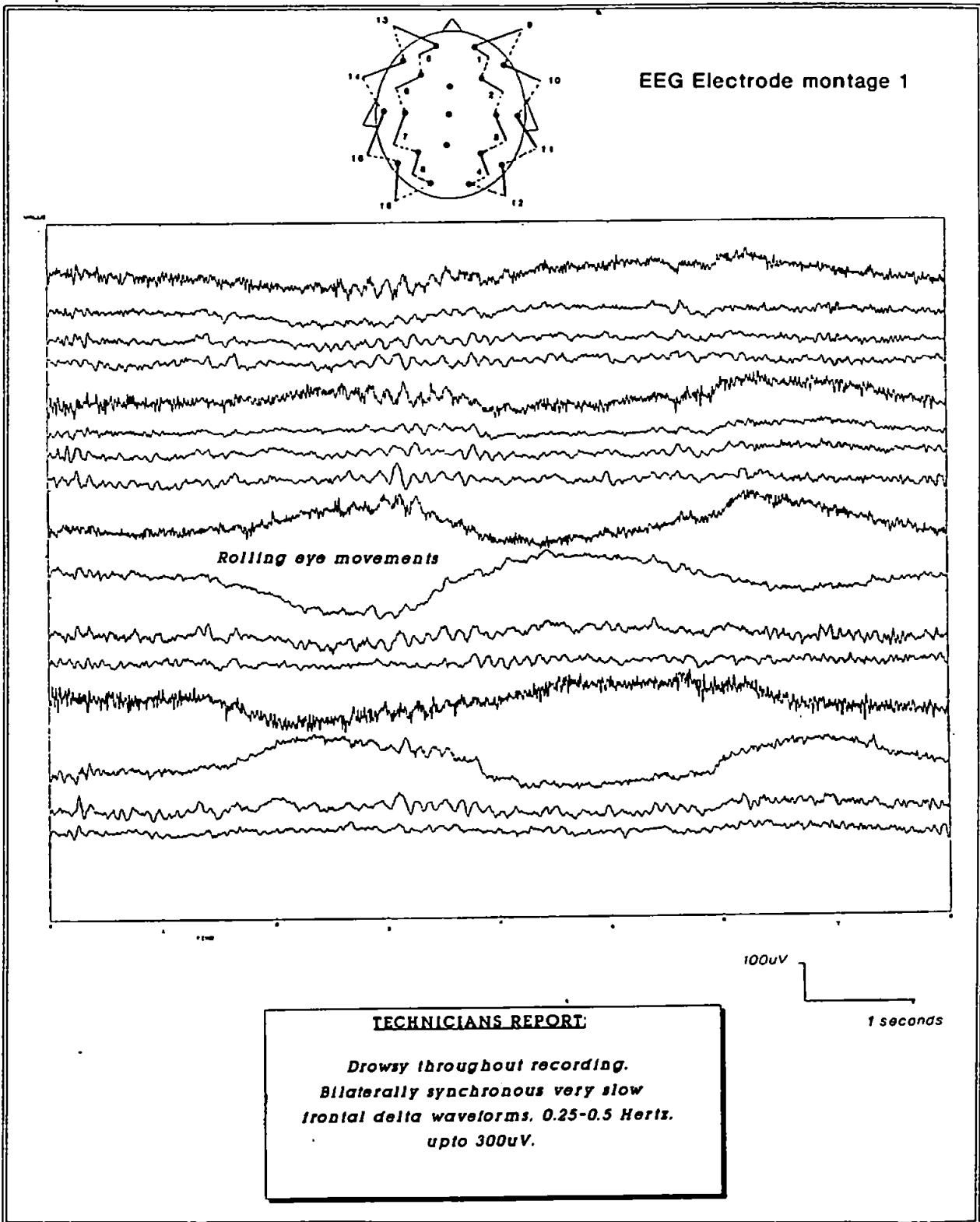
Figure 4.7 Sample data and technicians report 1

EEG Electrode montage 1

VALUE

fast vertical eye movements

TIME

1   2   3   4   5   6   7   8

100uV

1 seconds

TECHNICIANS REPORT:

*Theta elements with sharp components
in both temporal regions. Slow activity from
hyper-ventilation appearing bilaterally.
Bilaterally synchronous delta waveforms.
2-4 Hertz. upto 100 uV.*
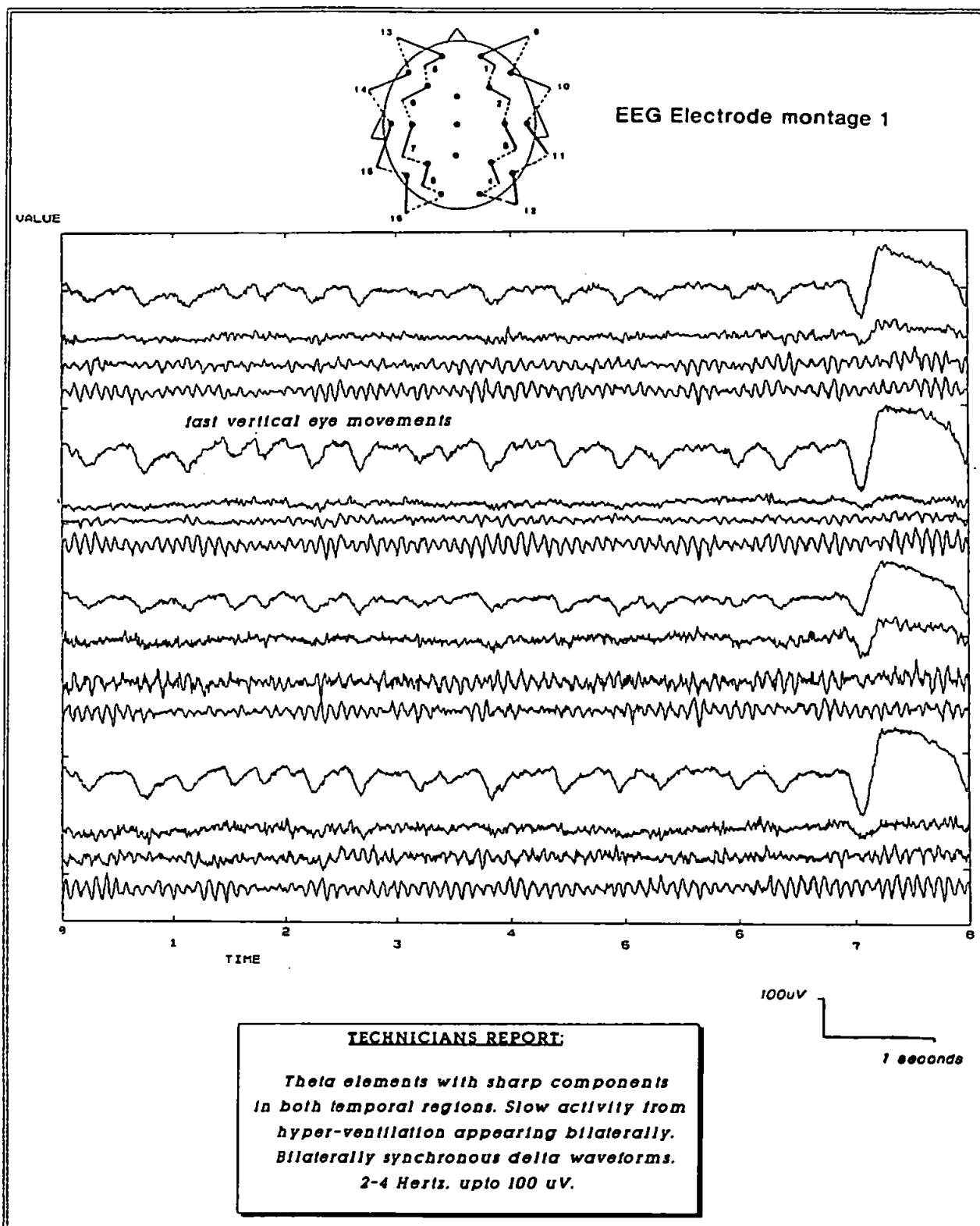
Figure 4.8 Sample data and technicians report 2

EEG Electrode montage 1

Abnormal slow waves

TIME

0   1   2   3   4   5   6   7   8

TECHNICIANS REPORT:

Slow waves 2-3.5 Hertz, 50-150uV, bilateral more on the right than the left, apearing in 1-5 second bursts. Delta, 1.5-3.5 Hertz, upto 150uV, both mid temporal areas. Generalised slow waves

100uV

1 seconds
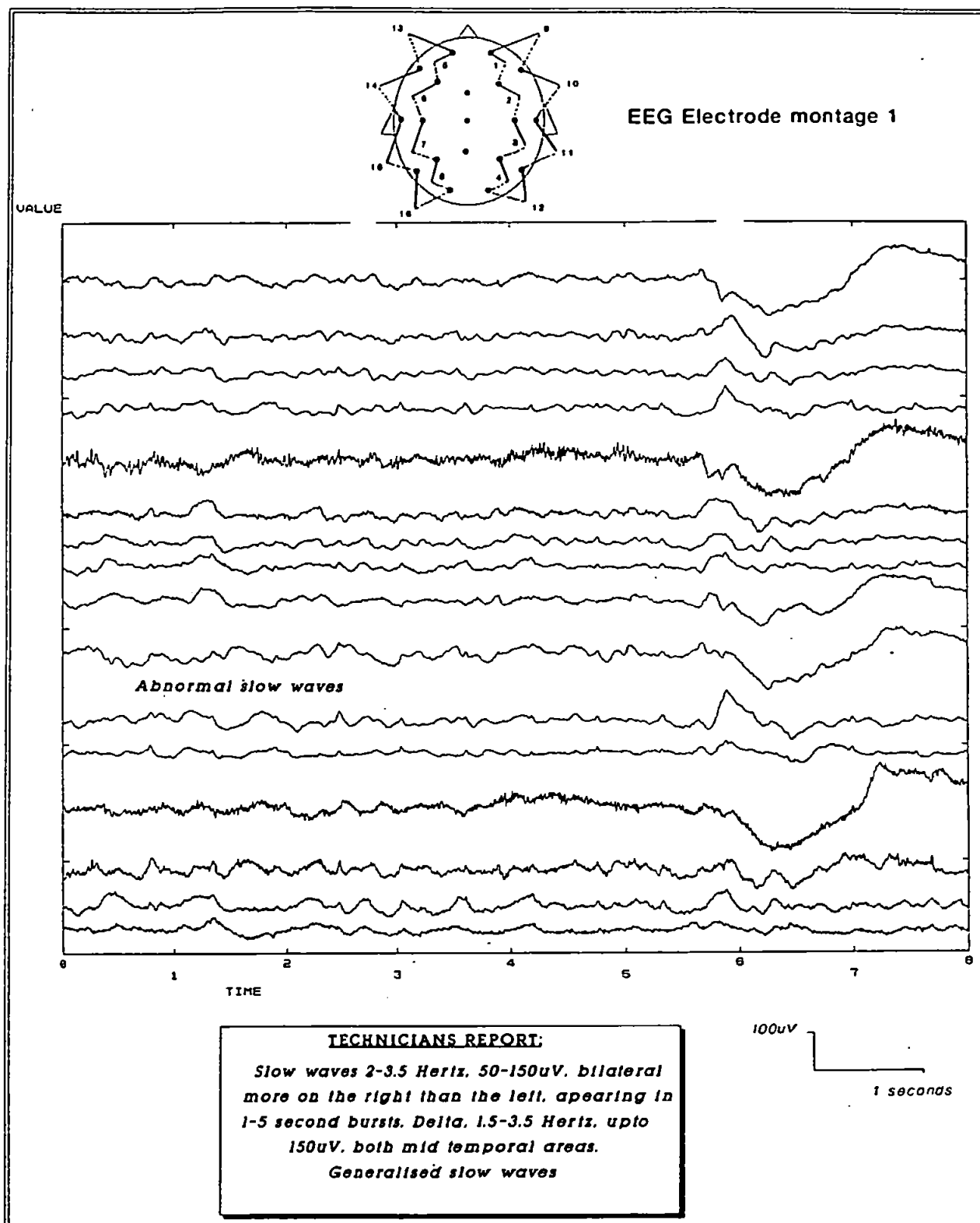
Figure 4.9 Sample data and technicians report 3
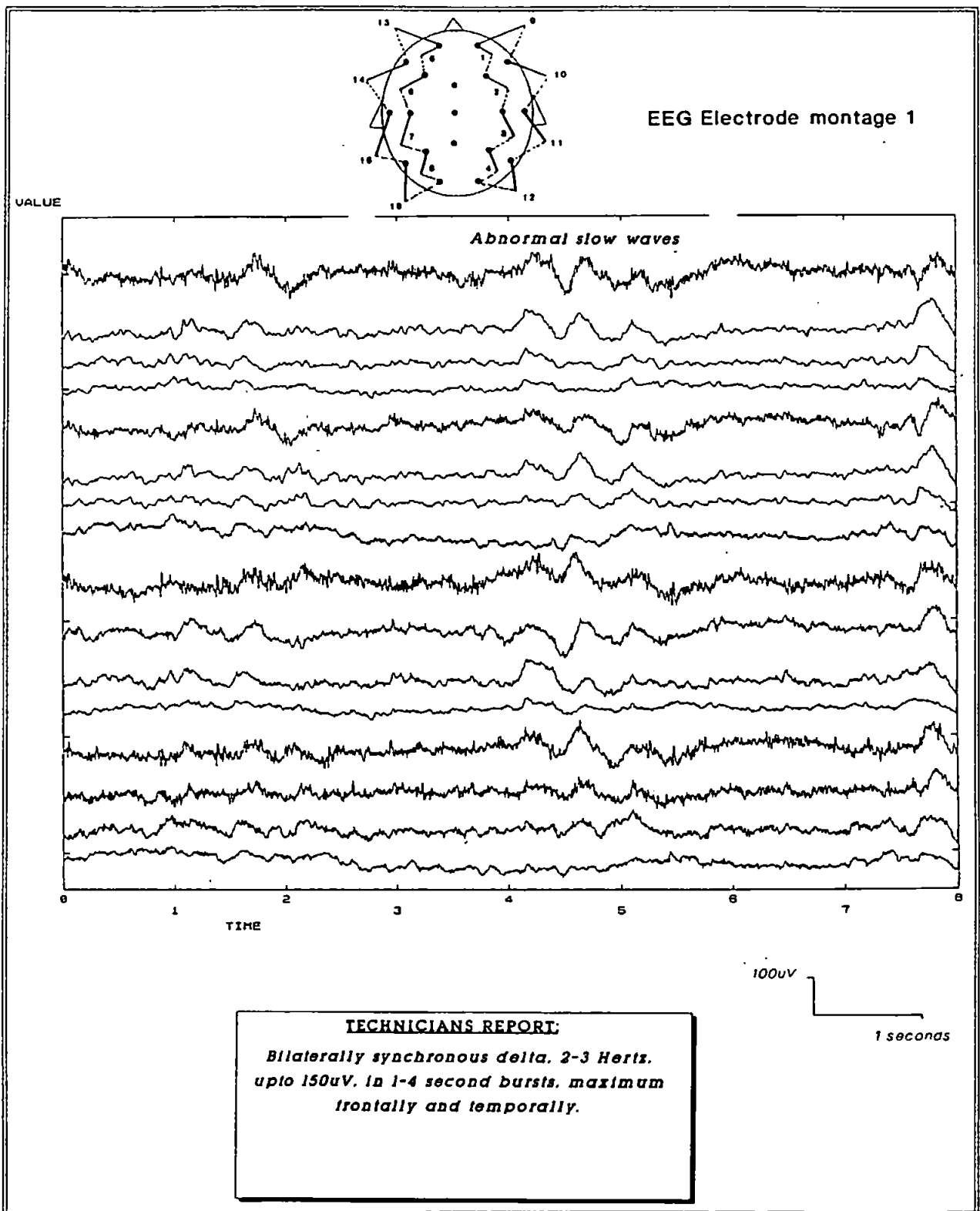
Figure 4.10 Sample data and technicians report 4

4.3 OFF-LINE INVESTIGATION INTO OCULAR ARTEFACT REMOVAL

The data acquired from volunteers has been used to investigate several OA removal hypotheses. These were: (i) bipolar and referential EEG montages require different EOG montages for best OA removal, and (ii) different OAs require different EOG models for best OA removal. This investigation was necessary to establish suitable EOG montages for all common types of OA and for all common EEG signal derivations. Suitable EOG montages were determined by comparing the corrected EEG obtained from application of the OA removal algorithm to sections of EEG contaminated with the most common OA types. Qualitative and quantitative comparisons were carried out using time and frequency domains. Previous research was extended to include many more EEG signals from different scalp positions.

Data analysis was carried out using the Interactive Laboratory System (ILS) /STI/ software package on a PC. This is a package that contains many mathematical, statistical, and digital signal processing routines and facilities as well as graphical display to enable hard-copies to be made.

4.3.1 INVESTIGATION PROTOCOL.

From the acquired data 504 2-second segments of EEG were edited, and their mean values removed. The edited segments were those that appeared to contain contamination from only one type of artefact, these being: vertical eye movement (VEM), horizontal eye movement (HEM), and blink. In

addition to this, segments were also chosen that appeared to contain no artefact contamination, to enable the effects of OA removal on uncorrupted data to be observed. Each 2-second segment of data provided a number of EEG and EOG signals and off-line data analysis was to firstly calculate OA parameters for each EEG/EOG combination, and secondly use these OA parameters to correct each EEG signal. OA parameters were calculated using linear multiple regression, and the performance of each correction compared in view of the above investigation objectives.

4.3.2 EEG SIGNALS AND EOG SUBTRACTION MONTAGES.

Analysis of the data has extended previous research *[Ifeachor, 1984; Jervis et al., 1988]* by using several previously successful EOG montages on a greater number of ·_Λ_ derivations. For the analysis, 9 referential and 9 bipolar EEGs were used. These were Fz, Cz, Pz, F4, C4, T4, F3, C3, T3, all referenced to linked mastoids, and F4-C4, Fp2-F4, Fp2-F8, C4-T4, F4-T4, T4-F8, C3-T3, F3-T3, T3-F7 (See figure 4.2). 4 different EOG montages were selected for the analysis. Two of these were derived referentially and two were bipolar in derivation. Figure 4.11 illustrates the EOG montages used.

The EOG montages used were chosen to enable a comparison to be made between smaller simpler montages that would enable easier, faster implementation and larger more complex montages that have been investigated previously *[Jervis, et al., 1988]*. The inclusion of both referential and bipolar EOG montages would enable a comparison to be made
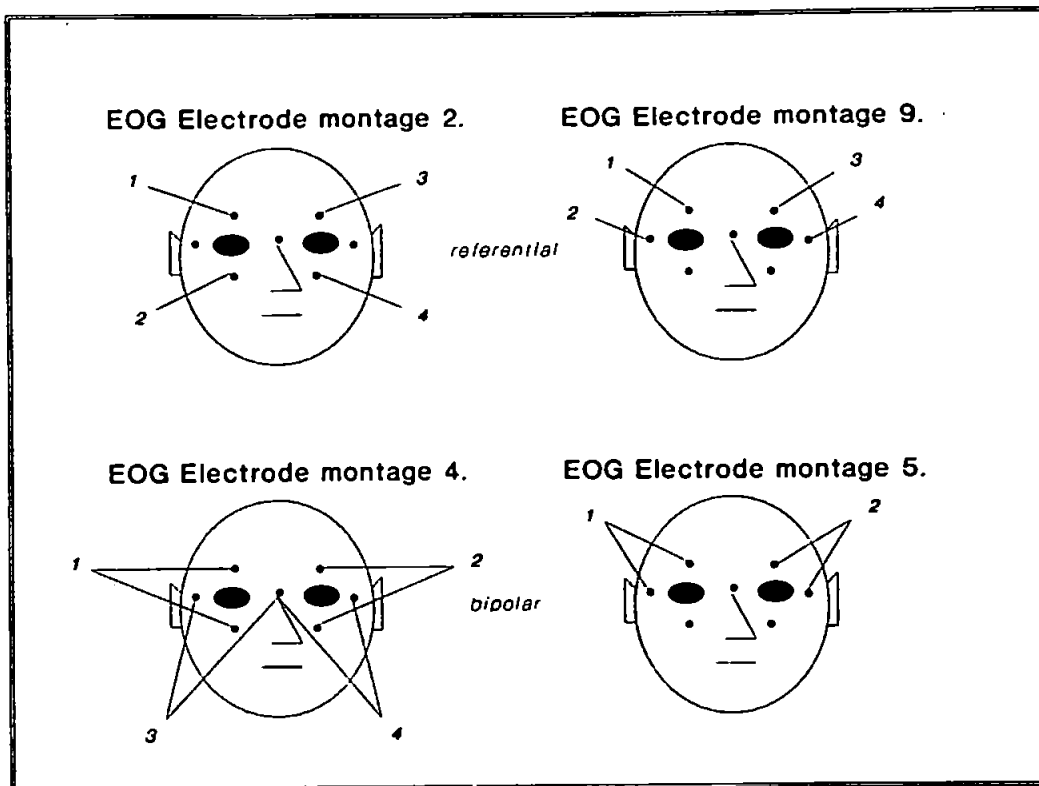
Figure 4.11 EOG montages used in pre-clinical trials.

of their respective abilities to remove OA from both bipolar and referential EEG montages. This is particularly true of models 5 and 9 which use identical electrode positions. The comparison was to be made using a large number of EEG signals measured over a large scalp area and not simply the vertex or frontal electrode positions used in previous studies. The montages used were not an exhaustive representation of all montages previously used. However, the method of recording would allow other montages to be investigated.

4.3.3 OCULAR ARTEFACT PARAMETERS.

Ocular artefact parameters were calculated off-line for each EOG/EEG combination by modelling the EEG/EOG samples as a linear multiple

regression given by:-

$$EEG = K1EOG1 + K2EOG2 + \ldots KmEOGm + \mathcal{E} \quad 4.1$$

Where   EEG is the measured EEG sample or response variable.

EOG are the EOG samples or regressor variables.

Km are the ocular artefact parameters.

$\mathcal{E}$ is the error in the response variable having zero mean.

The sample of n observations that are taken, consists of n sets of (m+1) values: one value is for the EEG, and one value for each of the m EOG measurements. For example for estimation of the OA parameters in a 2 second segment of EEG n will be equal to 512.

The best estimates of the OA parameters and the error variance is calculated using the least squares method *[Bajpai, et al., 1983]*. This necessitates the minimisation of:-

$$S = \sum_{j=1}^{n} \mathcal{E}_j^2 = \sum_{j=1}^{n} \left(EEG_j - \sum_{i=1}^{M} K_M EOG_{Mj} \right) \quad 4.2$$

and will produce the estimated regression equation:-

$$EEG = \hat{K}_1 EOG_1 + \ldots \hat{K}_m EOG_m \quad 4.3$$

Minimisation of S, for a fixed set of data depends upon the values of the parameters. Therefore we write $S = S(K_1, K_2, \ldots K_m)$. We require

$$\frac{\delta S}{\delta K_1} = \frac{\delta S}{K_2} = \ldots = \frac{\delta S}{K_m} = 0.$$ This set of conditions provides $(m+1)$ equations.

For example, for two EOG signals the OA parameters are estimated by minimising:-

$$S = \sum_{j=1}^{n} (EEG_j - K_1 EOG_{1j} - K_2 EOG_{2j})^2.$$

Differentiation of S and equating to zero leads to a set of normal equations *[Ifeachor, 1986; Bajpai, et al., 1983]* (see appendix A for further details). These are represented in matrix form as:-

$$\begin{bmatrix} \sum_{j=1}^{n} EEG_j EOG_{1j} \\[2ex] \sum_{j=1}^{n} EEG_j EOG_{2j} \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^{n} EOG^2_{1j} & \sum_{j=1}^{n} EOG_{1j} EOG_{2j} \\[2ex] \sum_{j=1}^{n} EOG_{1j} EOG_{2j} & \sum_{j=1}^{n} EOG^2_{2j} \end{bmatrix} \begin{bmatrix} \hat{K}_1 \\[2ex] \hat{K}_2 \end{bmatrix}$$

4.4

This gives the least squares estimate of the ocular artefact parameters Km and can be solved using a suitable matrix inversion technique. To calculate the large number of parameters involved in this stage of the investigation, a specialised digital signal processing software package, the interactive laboratory system (ILS) *[STI, 1988]* was utilised.

The true EEG was estimated using the estimated OA parameters from equation 4.4 and equation 1.5 from section 1.3.1, which is repeated here for clarity.

$$\hat{eeg}(i) = y(i) - \hat{K}\, EOG^T(i) \quad i = 1,2,\dots,m \qquad\qquad 1.5$$

where $\hat{K} = [\hat{k}_1, \hat{k}_2,\dots, \hat{k}_n]^T$ are the calculated OA parameters.

$\hat{eeg}(i)$ is the estimate of the true eeg(i)

$EOG^T(i)$ are the sampled EOGs

m is the sample number.

T indicates matrix transposition

4.3.3.1 AVERAGED OCULAR ARTEFACT PARAMETERS.

OA parameters were calculated for each of the 504 2-second EEG segments and an average was calculated for each parameter, at each electrode site, and for each of the three main OA types. This provided an averaged representation of OA potential distribution over the scalp (See appendix E).

4.3.4 COMPARATIVE RESULTS.

To enable the EOG subtraction montages to be compared, both qualitative and quantitative analysis was carried out on the results obtained above. A qualitative comparison of OA removal was carried out

using time and frequency domain representations, and a quantitative comparison was made by quantifying the accuracy of the regressive model.

4.3.4.1 QUALITATIVE COMPARISON

Qualitative analysis was achieved by plotting the uncorrected EEG and the corrected EEGs, from the four respective EOG OA correction montages, on a single graph, synchronised in time. This graphical representation allowed the corrections to be visually assessed by two EEG experts (1 consultant, 1 chief technician) in addition to the author.

Visual analysis lacks an easy quantitative description and can be subjective in nature. However, this type of analysis has been found, in previous studies, to be extremely sensitive to remnant artefact caused by unsatisfactory OA removal *[Ifeachor et al., 1988]*, and to the distortion of clinically significant EEG waveforms. In addition to this, visual analysis will be the final test for clinical acceptance and is no more subjective than conventional EEG analysis.

(a) TIME DOMAIN.

Figures 4.12-17 illustrate the uncorrected and corrected EEG with respective EOGs and OA parameters for various combinations of referential and bipolar EEG recordings represented in the time domain. These figures also show the results of visual assessment made by the two EEG experts. The EEG signals are chosen as those which are most effected by the respective OA.
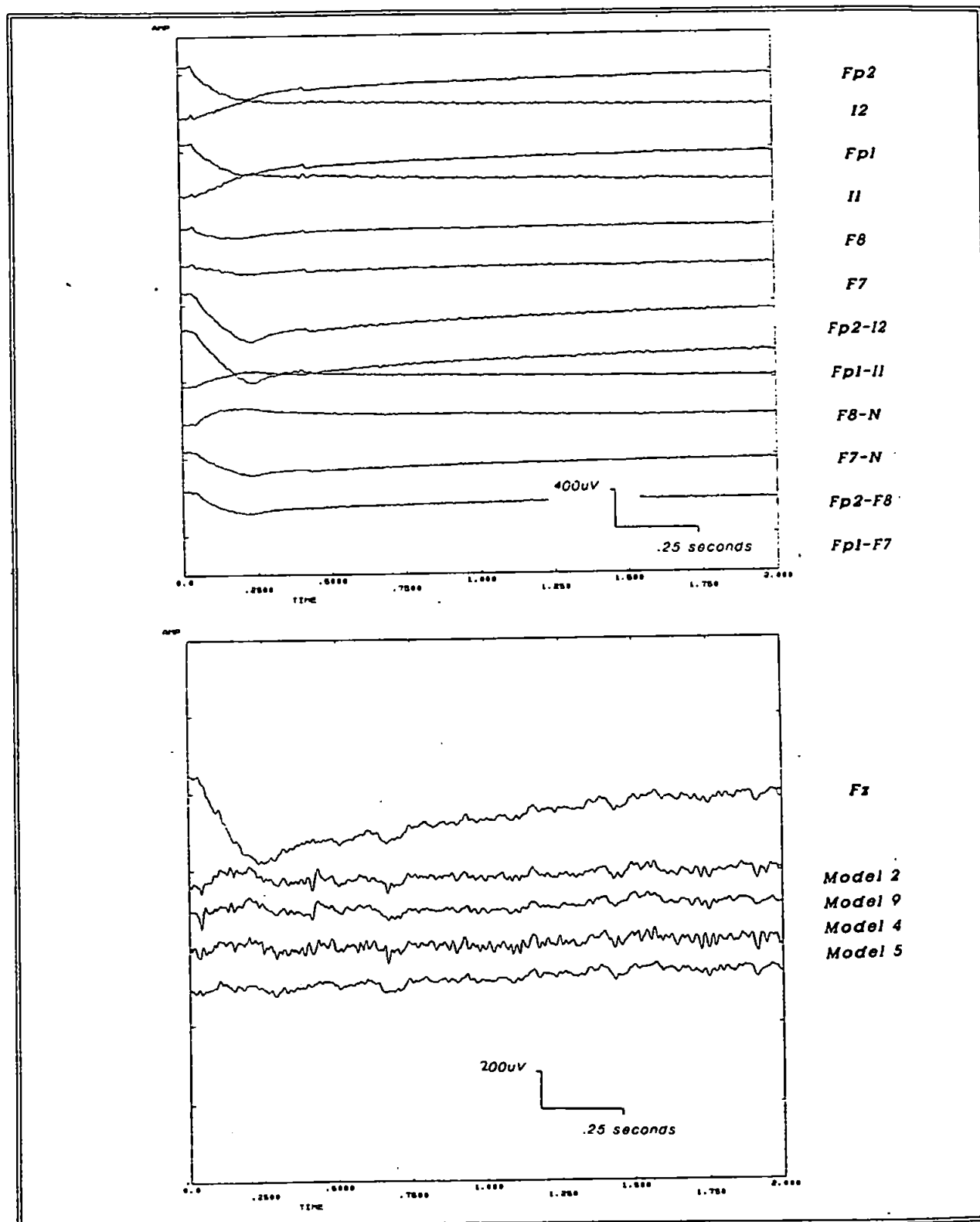
Figure 4.12 Comparison of four EOG montages for OA removal

from EEG recorded at Fz and containing vertical eye movement.
page 111
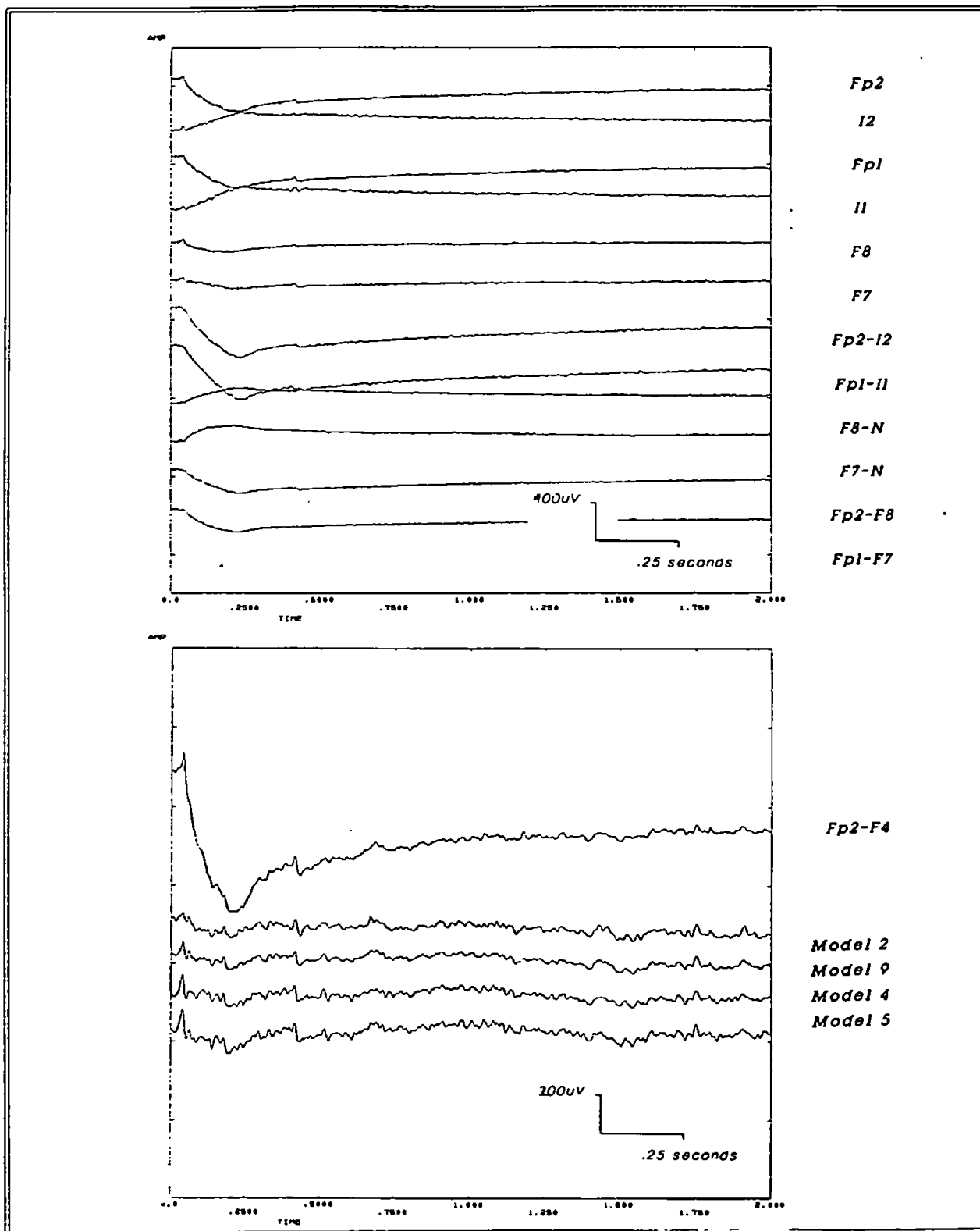
Figure 4.13 Comparison of four EOG montages for OA removal

from EEG recorded at Fp2-F4 and containing vertical eye movement.
page 112
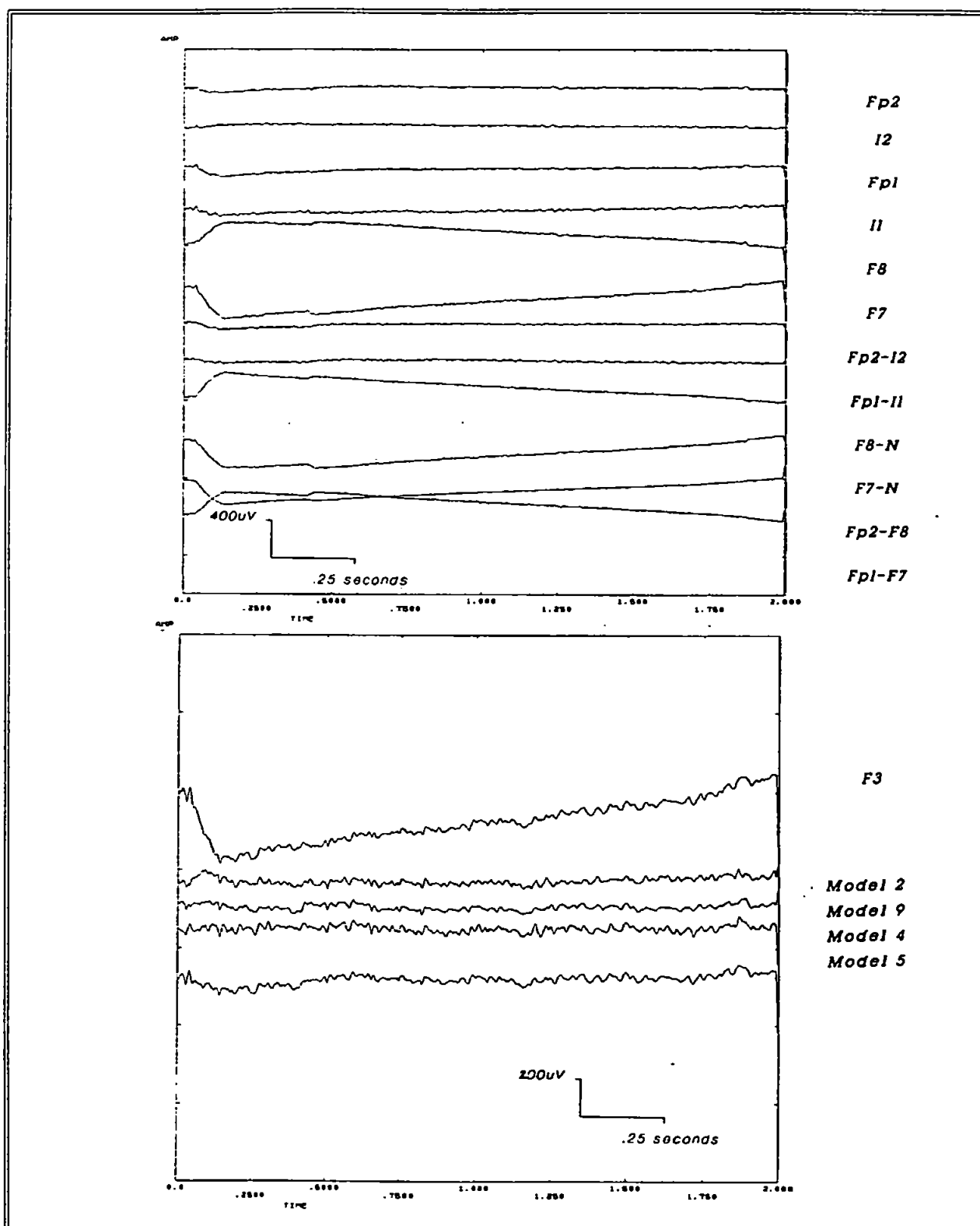
Figure 4.14 Comparison of four EOG montages for OA removal

from EEG recorded at F8 and containing horizontal eye movement.
page 113
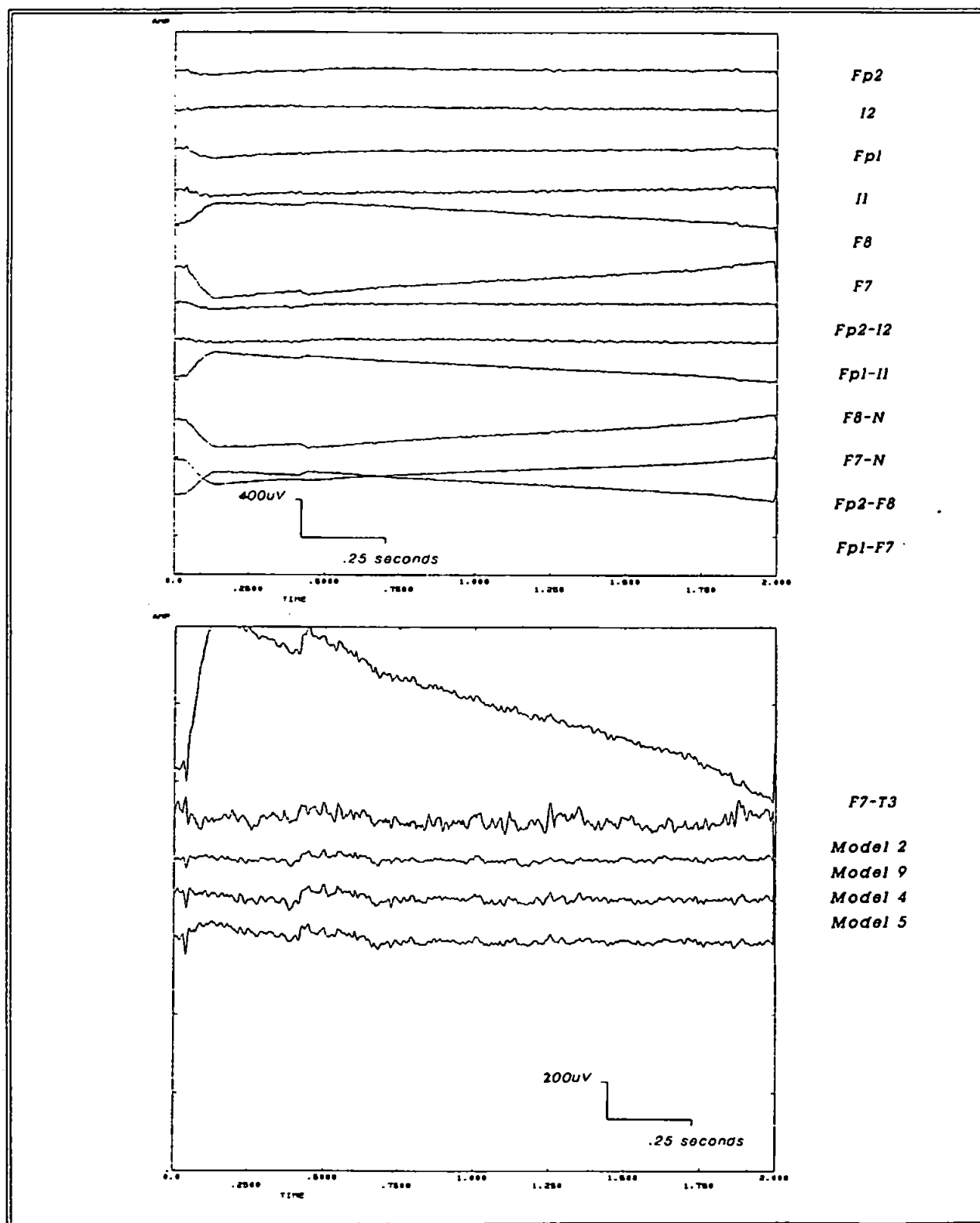
Figure 4.15 Comparison of four EOG montages for OA removal

from EEG recorded at F7-T3 and containing horizontal eye movement.
page 114
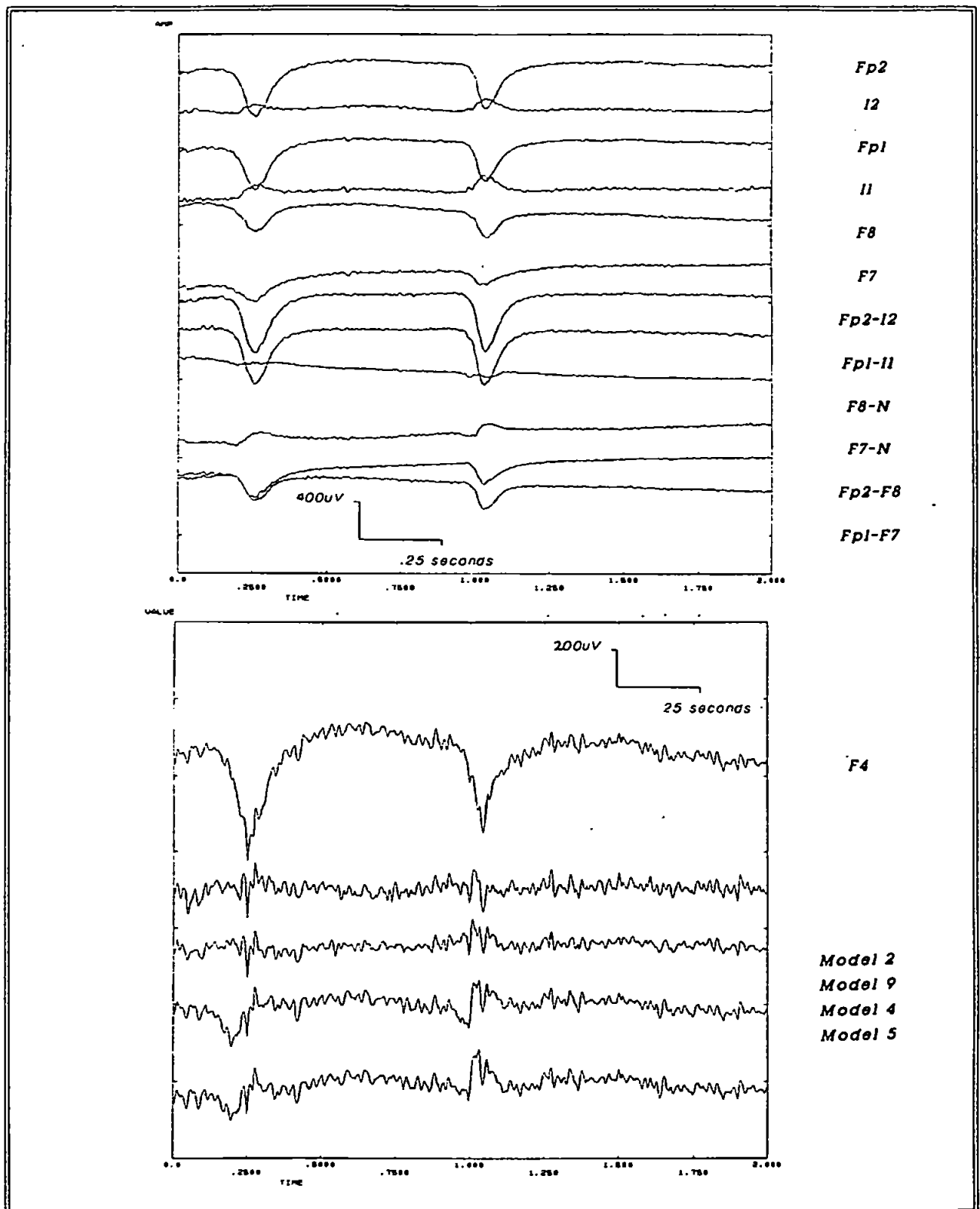
Figure 4.16 Comparison of four EOG montages for OA removal

from EEG recorded at F4 and containing blink.

Figure 4.17 Comparison of four EOG montages for OA removal

from EEG recorded at Fp2-F4 and containing blink.
page 116

Analysis of these results allows the following observations to be made:-

. For all instances of artefact using most electrode derivations, all 4 EOG models exhibited visually similar OA removal in that a significant attenuation in OA amplitude was observed. This is in agreement with Ifeachor, et al., 1986 but extends this to new EEG signals and EOG montages. However, it can be seen from the results that the different EOG montages generate corrected EEGs that differ in detail. This appears to be particularly true when comparing bipolar and referential EOG models. The bipolar EOG models appear to have maintained the detail of the uncorrected EEG to a greater extent than the referential models. This can be seen most clearly when comparing models 9 and 5 in figures 16 and 17 which employ identical electrode placements and only differ in derivation.

An explanation to this observation can be made by studying the measurement of the EOG signals. Consider the EOG signal measured between Fp2 (above the right eye) and F8 (to the right of the right eye), i.e. Fp2-F8. :-
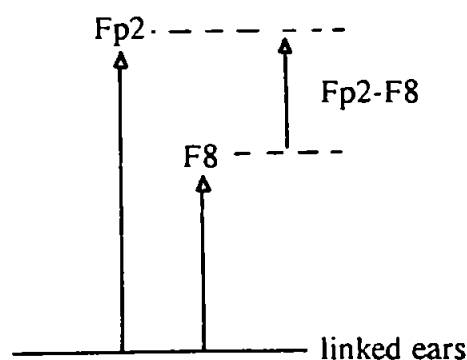
Figure 4.18 Derivation of a bipolar EOG signal.

It can be seen that a bipolarly derived signal will have the effect of removing any common signal between them. The common signal in the EOG is likely to include frontal EEG signals, and therefore the bipolarly derived EOG signals will reflect more accurately the true EOG. The removal of the EEG contamination in the EOG will therefore prevent this common signal from further corrupting the original EEG signal, hence reducing so called 'secondary artefacts' in the corrected EEG. This indicates that a bipolar EOG montage is more suitable for OA removal.

. EOG model 2 was derived from electrodes placed only above and below the eyes - the position where maximum amplitude is observed from VEM OA. It was therefore, not expected to correct as well for all types of OA as the other models which included additional electrodes. In particular, for HEM, model 2 was expected to give poorer results than model 4 which included the same electrodes as model 2, but additional electrodes placed on either side of the eyes - the position of maximum amplitude observed in HEM OA. However, the above results indicate that both models performed comparably when correcting for HEM, i.e. both models significantly attenuated the large OA potential due to eye movement. Comparing the corrections of models 2 and 4 only differ in detail of correction.

It must be borne in mind that there is no effective isolation between scalp electrodes, and therefore, although attenuated, it appears that the electrodes of model 2 contain sufficient information to enable significant attenuation of the OA waveform to be affected. Additional electrodes provide apparently no further information and might therefore be redundant. Observation of the respective OA parameters for models 2 and 4 support this hypothesis as it can be seen that for instances of HEM the

vertical EOG electrodes placed above and below the eyes in model 4 are of less significance than the same electrodes of model 2, and are of much less significance than the horizontal EOG electrodes of model 4.

. Three exceptions to the above observations were observed. In EEG derivations recorded at Fp2-F4, T4-F8, and T3-F7, the corrected EEG, using a bipolar EOG montage, was visually inferior to that of the referential EOG montage. It is noticed also that the EEG signals, in each exception, were of bipolar derivation, recorded close to the eyes, and showed differences in correction most noticably when the EEG signal contained large artefactual signals and/or a possibility of multiple artefact. This observation would indicate that the bipolar EOG models contain less information than the referential models and that as a consequence, remnant OA is left in the corrected EEG signal. It is hypothesised that this phenomenon is linked to that of observations made above, and that the subtraction of EOG signals of equal polarity has the effect of attenuating true EOG signal in addition to removing signal commonality.

The visual comparison of corrections from bipolar and referential EOG models showed that OA attenuation was, in the majority of EEG derivations, similar and significant, which is in agreement with Ifeachor, 1984. However, this investigation has extended this finding to new EOG montages and new EEG signals and found that better detail in correction is observed when using bipolar EOG montages. It is therefore concluded that a simple bipolar EOG montage, such as montage 5 above, provides adaquate OA removal from the EEG channels investigated.

(b) FREQUENCY DOMAIN.

Spectral analysis was carried out to enable a comparison to be made between the spectral components in the uncorrected and corrected EEG, for each EEG/EOG combination of (a) above. Figures 4.19 - 24 illustrate the Power spectral density for the uncorrected and corrected EEGs shown in figures 4.12-17. For clarity each figure also displays the respective OA parameters and visual assessment of the previous figures.
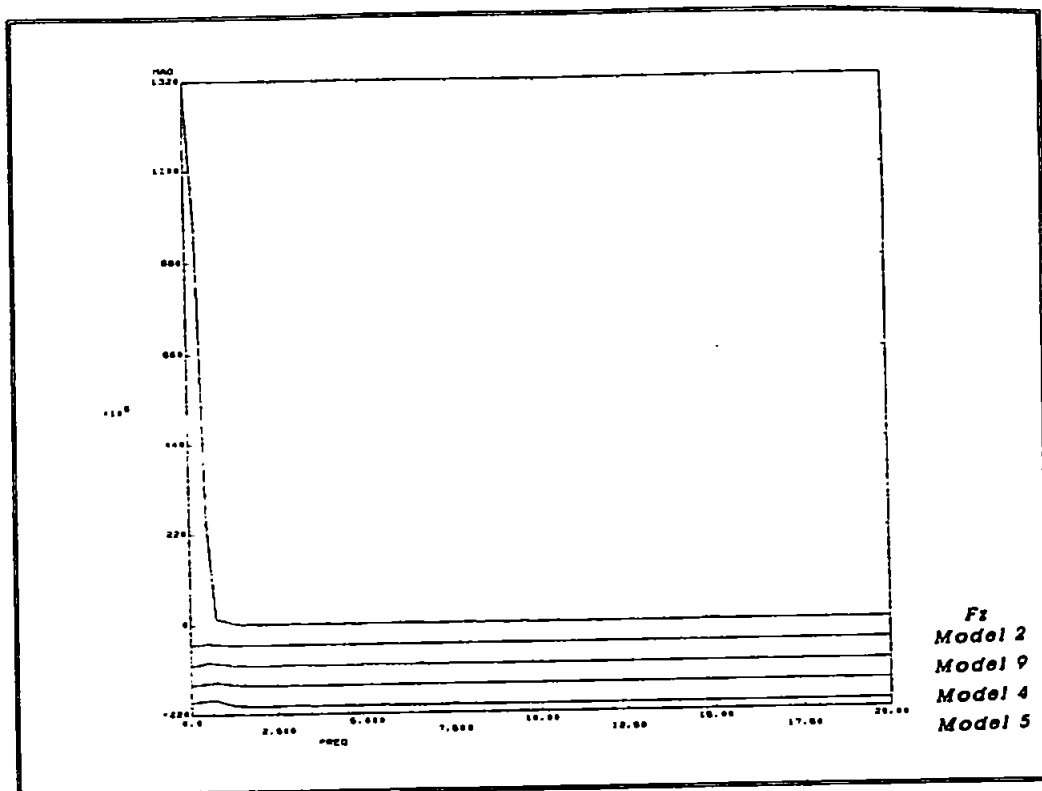
Figure 4.19 Comparison of four EOG montages for OA removal from EEG recorded at Fz and containing vertical eye movement.
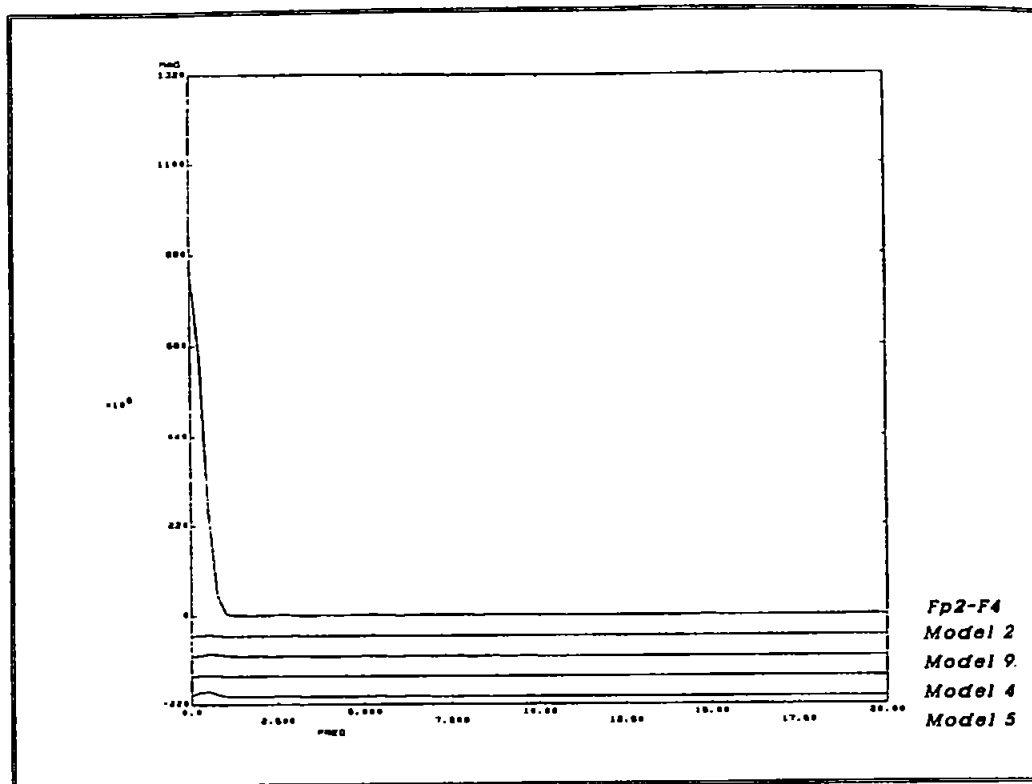
Figure 4.20 Comparison of four EOG montages for OA removal

from EEG recorded at Fp2-F4 and containing vertical eye movement.
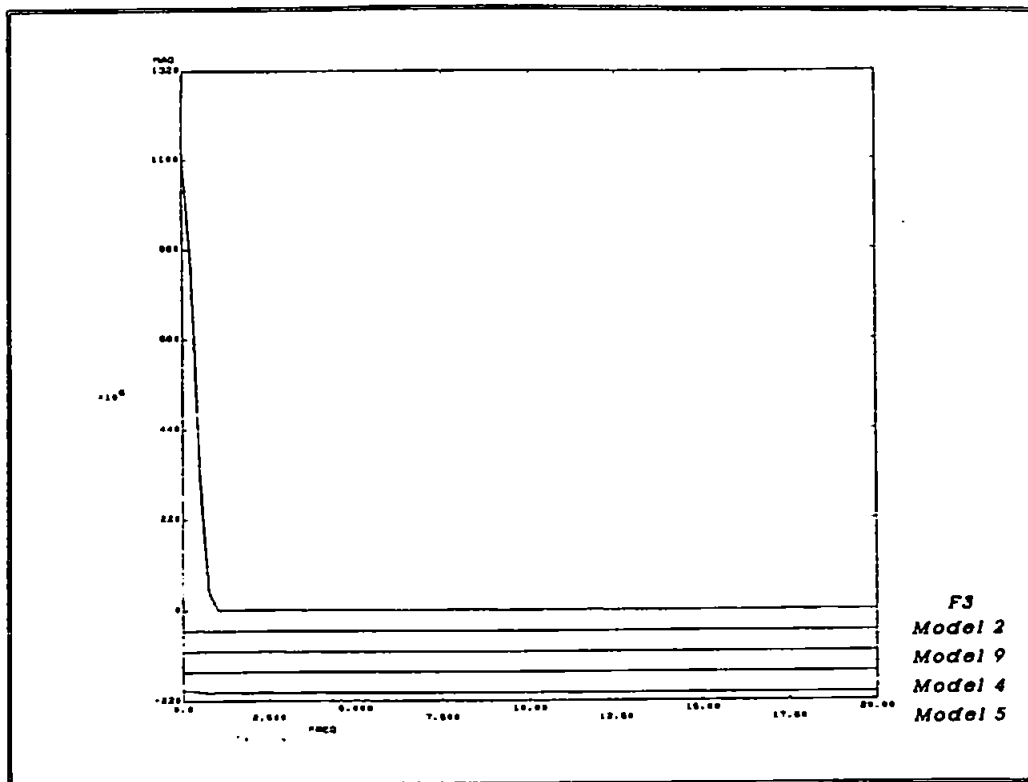
Figure 4.21 Comparison of four EOG montages for OA removal

from EEG recorded at F8 and containing horizontal eye movement.

page 121
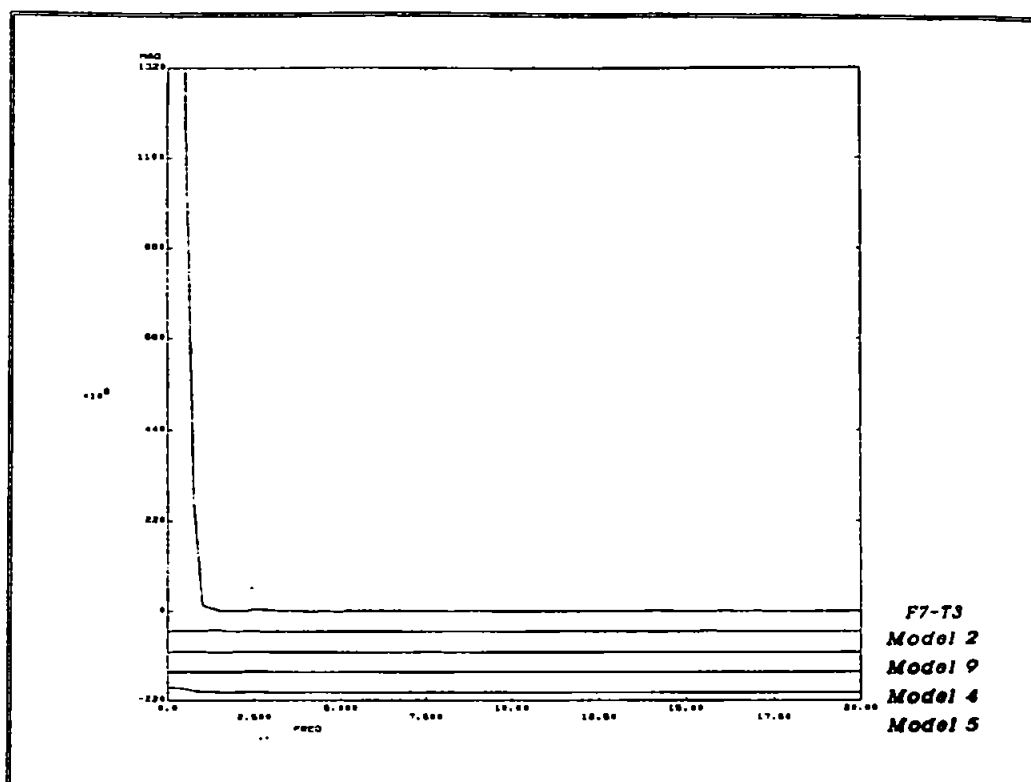
Figure 4.22 Comparison of four EOG montages for OA removal

from EEG recorded at F7-T3 and containing horizontal eye movement.
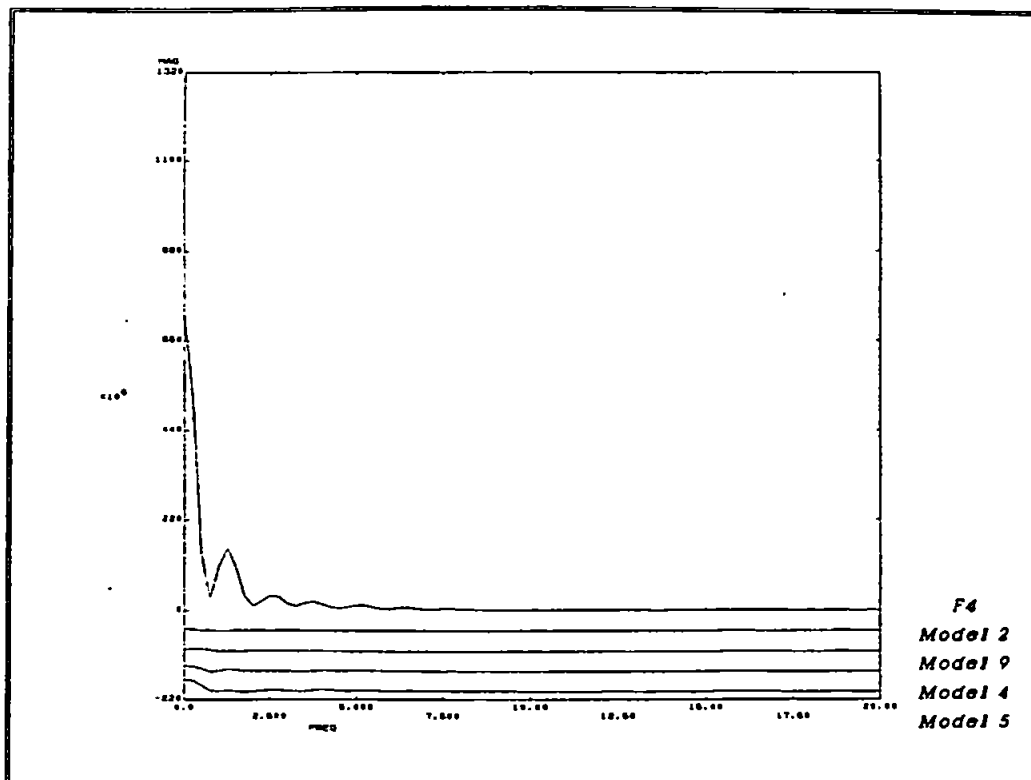
Figure 4.23 Comparison of four EOG montages for OA removal

from EEG recorded at F4 and containing blink.
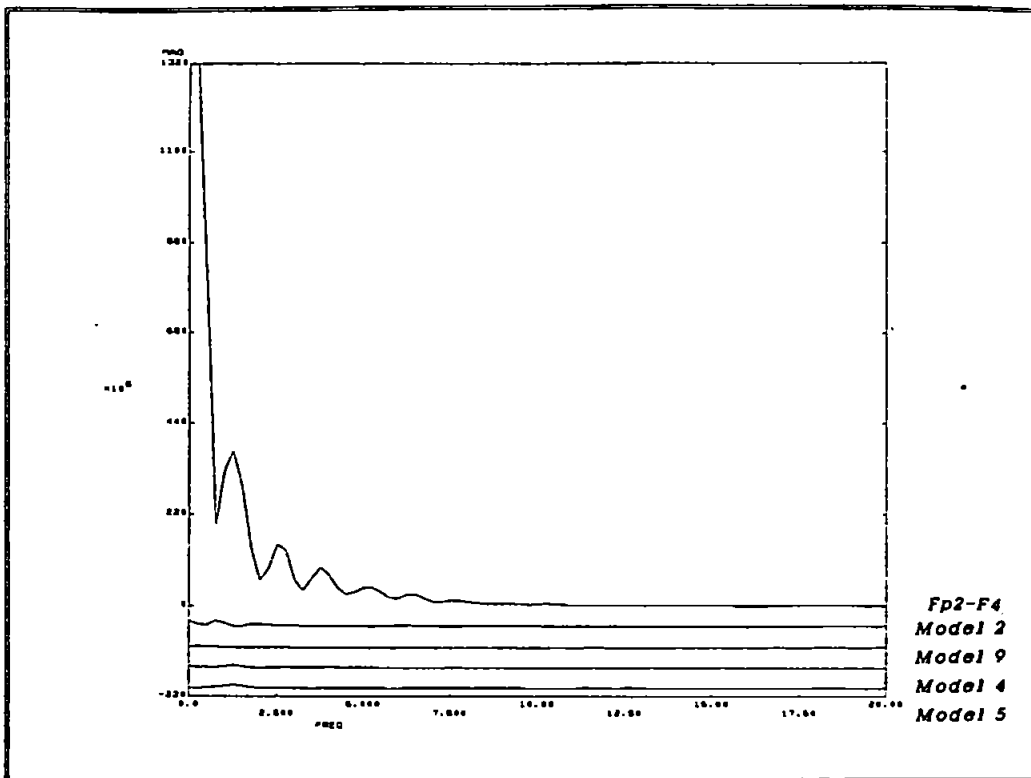
Figure 4.24 Comparison of four EOG montages for OA removal

from EEG recorded at Fp2-F4 and containing blink.

Analysis of these results allows the following observations to be made:-

. In the frequency range of 0-20Hertz, All EOG montages significantly attenuated the large low frequency component attributable to the OA.

. Montage 9 showed greater remnant OA spectral power.

. None of the models introduced additional spectral peaks in the 0-20Hertz frequency band. Model 2 introduced spectral peaks into the estimated EEG above 25 Hz. This is most clearly seen in the case of blink artefact in EEG channel Fp2-F4 on figure 4.20 (the frequency scale does

not show this).

The frequency domain montage comparison has further corroborated the previous findings. All EOG montages provide significant OA attenuation from the EEG. Bipolar montages provide least attenuation and this appears linked to the detail of the corrected EEG

4.3.4.2 QUANTITATIVE COMPARISON.

Quantitative comparison of different EOG subtraction models has proved difficult. Verleger et al. (1982) pointed out the lack of a quantitative assessment of the validity of OA removal techniques and several authors have addressed this problem since then. The major stumbling block for any attempt at quantitative comparison of EOG subtraction models, and in EEG analysis in general, is the absence of a measurable 'true' EEG signal with which to compare any corrected EEG signal. Jervis et al. (1988) discusses several quantitative techniques including measurement of covariances of EOG and EEG before and after OA removal *[Verleger, et al., 1982]*, measurement of deviations, between corrected EEG and an estimate of the true EEG taken from an area of EEG with no obvious OA contamination *[Gratton et al., 1983]*, and measurement of residual after OA removal *[Berg, 1986]*. However, it has been found that in some cases, these quantitative methods can give misleading results *[Ifeachor, et al., 1988]*. Such cases are when EOGs contain other artefacts not related to ocular movements, such as muscle artefact and frontal EEG activity.

The quantitative comparison carried out here is an adaptation of Gratton et al. (1983) and quantifies the amount of correction, for each EOG model, by calculating a value which represents the error between the estimated EEG from the regressive model, and the measured EEG. This was called the percentage error of fit (PEOF) and reflects the error in the fit of the regressive model, which uses the EOGs as the regressor variables.

The percentage error of fit (PEOF) is calculated as:

$$PEOF = \frac{\{ (y(i) - \hat{eeg}(i))^2 \}}{\{ y(i)^2 \}} \times 100 \qquad 4.5$$

where $y(i)$ is the measured EEG

and $\hat{eeg}(i)$ is the estimate of the

e.g. for a 2 parameter EOG model

$$\hat{eeg}(i) = \hat{k}_1 \, eog_1(i) + \hat{k}_2 \, eog_2(i)$$

A small PEOF will therefore indicate that the EOG model estimate provides a close approximation to the measured EEG signal. When comparing the PEOF from different EOG subtraction models, the model calculated as having the lowest PEOF will therefore have provided the highest amount of OA attenuation and would therefore be considered as the best EOG subtraction model. The PEOF of the multiple linear regression model was calculated for each EEG/EOG combination and table 4.1 gives the PEOF for the models and EEG signals illustrated in figures 4.12 - 17 (4.19 - 24) (See Appendix F for the complete set of results).

Model	VEM				HEM				BLINK			
	2	9	4	5	2	9	4	5	2	9	4	5
Fz	6.69	6.48	6.56	9.26								
Fp2-F4	7.39	7.18	6.87	7.23								
F3					2.08	1.54	2.03	2.37				
T3-F7					1.89	0.28	0.52	0.55				
F4									7.3	6.48	16.42	16.16
T4-F8									11.92	1.82	4.3	4.61

Table 4.1 Values of PEOF for the EEG signals and EOG models illustrated in figure 4.12-17.

Analysis of these values of PEOF and careful comparison to the results of visual qualitative EOG model comparison has enabled the following observations to be made:-

. The PEOF is proportional to the difference between the estimated EEG using the EOGs as regressor variables, and the measured EEG. This difference is viewed as the "true" EEG plus any non OA related artefact, as seen in equation 1.5 of section 1.3.1. The true EEG will give a finite error, or PEOF, and will be variable between EEG segments and channels, depending on the type of EEG activity present. For example, a segment of EEG containing high amplitude delta waveforms will produce a higher PEOF than one containing little or no significant EEG waveforms. Values of PEOF from different EOG subtraction models can therefore only be compared using the same segment of EEG signal. A further difficulty arises when EOG signal derivations contain frontal EEG activity. In this case a model giving a minimum PEOF might have introduced 'secondary artefact' into the corrected EEG signal and therefore might be visually assessed as having 'overcorrected' the EEG signal. This effect is illustrated in model 9 on the HEM OA illustrated in figure 4.15.

Referential EOG models give a smaller PEOF than the bipolar models but it can be seen that this is at the expense of reduced EEG signal amplitude. This is again linked to the observations made above and it can be seen that the respective EOG signal derivations for referential EOG models contain frontal EEG signals resulting in a corruption in the corrected EEG. It was concluded therefore that the PEOF is only of any real use when the difference in corrections of any two models is large. By contrast the qualitative visual inspection described above enabled subtle differences in correction to be evaluated.

4.3.5 SUMMARY OF RESULTS FROM OFF-LINE ANALYSIS.

In conclusion to this section of the research, the results to date indicate that it is not necessary to use a particular EOG model to remove a particular OA as all the models examined performed similarly when compared visually and with spectral analysis. This is true with most EEG derivations examined. The results also indicate that referential and bipolar EEG derivations do not require different EOG models for OA removal. In most cases the bipolar models gave similar artefact removal but gave visually better results when comparing detail. However, OA parameters were dependant on electrode derivation and OA type. This observation can be exploited by using the averaged OA parameter values, for each electrode derivation and type of OA, as a pre-loading value for any on-line adaptive OA removal. This would minimise the time taken for a particular OA removal algorithm to settle to an optimum value.

Qualitative differences in OA removal of EOG subtraction models has been largely attributed to the presence of artefacts within the EOG signals. There is therefore justification in enhancing the EOG signals, to remove artefact. This is likely to improve the quality of OA removal in both bipolar and referential EOG models. Analysis of EOG spectral content shows that little significant activity exists above 5 Hertz and none above 8 Hertz *[Whitton et al., 1978]*. A digital finite impulse response filter (FIR) *[Terrell, 1980]* will be used to process the EOGs prior to using the EOGs for OA removal.

The results to date indicate that the simple bipolar model 5 provides sufficient information on eye movement to enable visually adequate OA removal. This model will be easy to implement meaning faster correction. Model 5 also uses standard EEG electrode positions so that intelligent OA removal could be carried out without requiring special electrode placements. This is significant when only a limited number of channels is available on standard EEG equipment.

It will be necessary to develop a means of quantitatively measuring the correction from the EOG models. It is proposed to subject segments of EEG containing no artefacts to spectral analysis. From this powers in the main EEG frequency bands can be extracted and these used as a reference. Sections in which OAs have been removed may then be compared to this reference to quantify the correction. In the on-line case this reference will be updated to track any intra and inter patient changes in the EEG activity.

4.4 ON-LINE INVESTIGATION INTO OCULAR ARTEFACT REMOVAL.

In order to remedy the deficiencies of the present on-line correction method (see chapter 2) it was necessary to investigate further the Recursive Least Squares (RLS) algorithm (see section 1.3.2 and appendix A). In particular it was important to investigate: (i) The selective use of the OA removal algorithm by pre-loading the RLS algorithm with ocular artefact parameters most suitable for the type of artefact present. These were obtained from off-line investigation. (ii) Directing the OA removal

algorithm only to sections of contaminated EEG, i.e. the algorithm is only applied when an artefact is present. This investigation will provide a new means of applying the RLS algorithm in an intelligent manner. Section 4.4.1 describes the techniques and data used in this investigation. Section 4.4.2 - 4.4.3 details the RLS algorithm and compares the use of the recusive algorithm to the results obtained during off-line investigation. Section 4.4.4 - 4.4.5 investigate the use of the OA removal algorithm in a selective and directed manner and this is quantified in section 4.4.6.

4.4.1 INVESTIGATION PROTOCOL.

From the acquired patient data, 8-second segments of EEG and EOG signals were edited, and their means removed. On-line OA removal was simulated by presenting the edited signals, stored as a computer disk file, to a software program that employed the RLS algorithm to effect OA removal. The corrected EEG signal and calculated OA parameters were also stored as computer disk files to allow analysis and graphical presentation using the ILS software package. The RLS algorithm using the U-D factorisation (Appendix A) was implemented in software using the C programming language *[Kernighan and Ritchie, 1988]*. Figure 4.25 illustrates a conceptual view of the on-line OA removal process, and figure 4.26 presents the flow diagram for the OA removal software program (Appendix G lists this program). The software program enabled the selection of the same EEG and EOG signals as those used in the off-line investigation above.
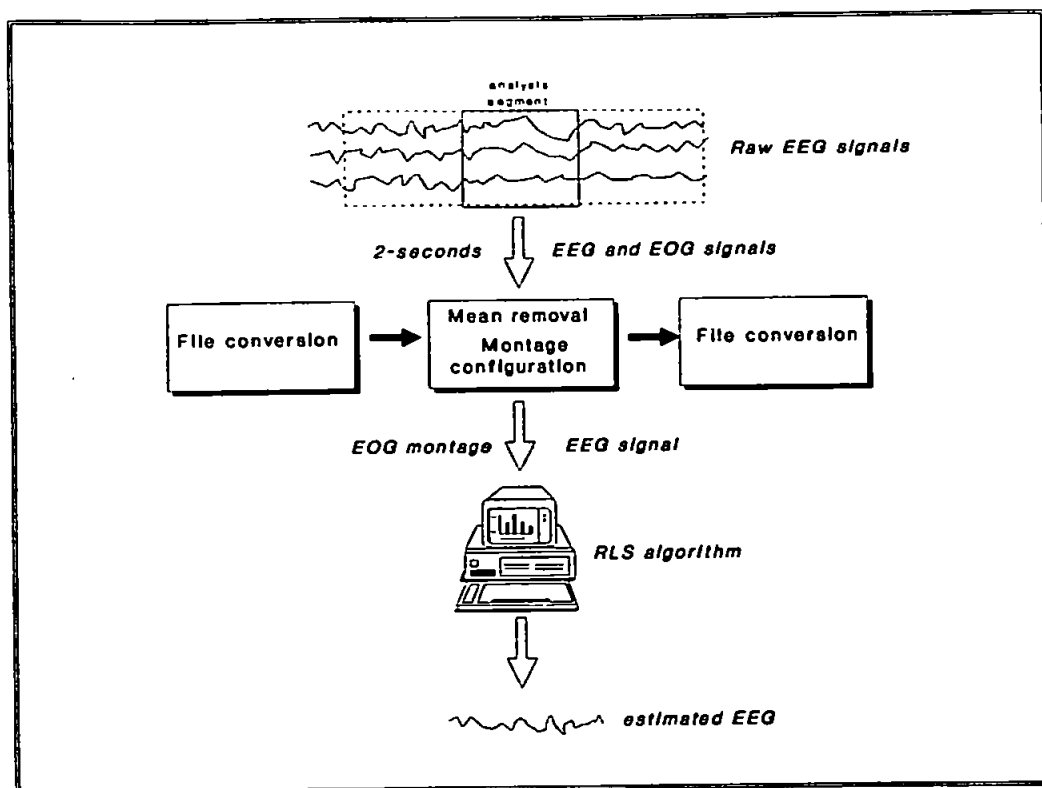
Figure 4.25 Conceptual view of on-line OA removal simulation.
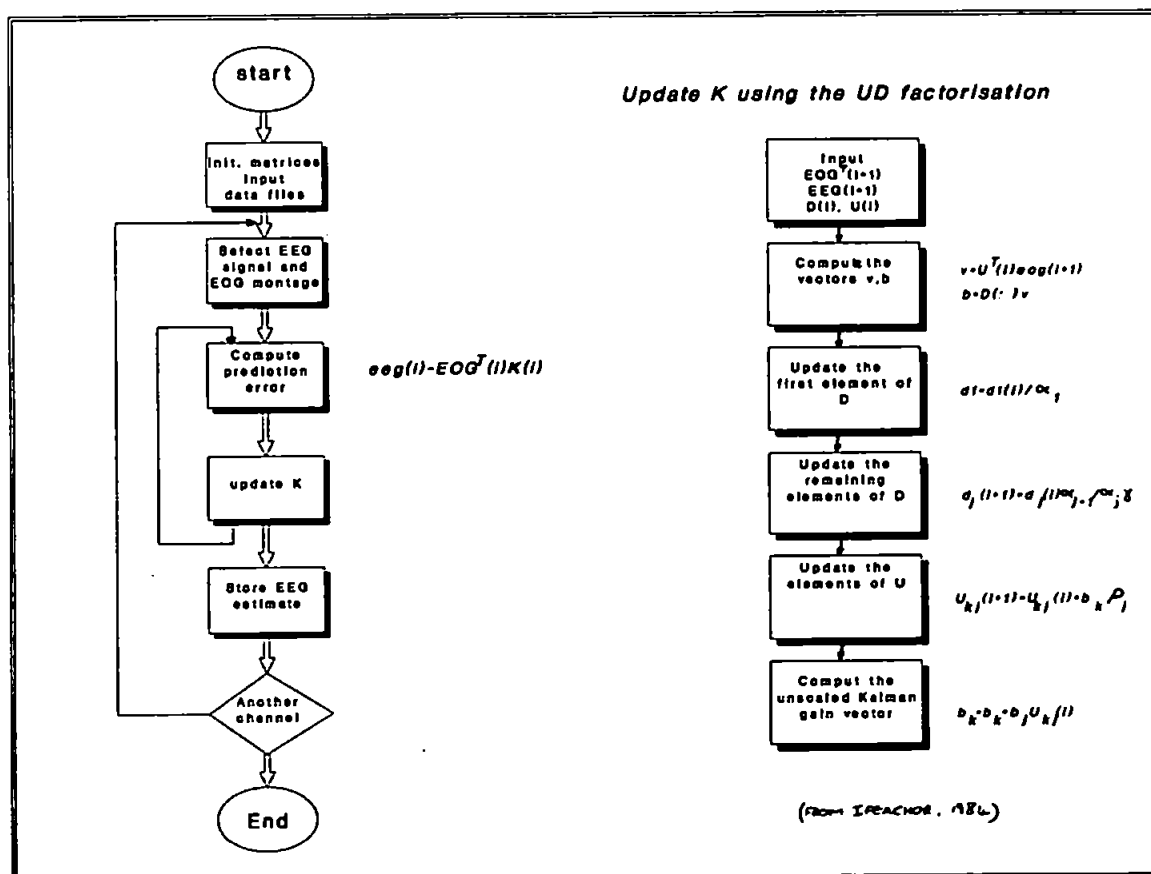
Figure 4.26 Flow diagram of the RLS algorithm.
page 131

4.4.2 THE MODIFIED RLS OA REMOVAL ALGORITHM.

The RLS algorithm is a recursive implementation of the least squares algorithm described above, which overcomes the time consuming calculation of the inverse matrix by updating the Ks (OA parameters) at each sample point according to the error between the previous sample of measured EEG and the previous estimated EEG *[Ifeachor, et al., 1986]*. The Ks will therefore converge on the optimum for a signal with stationary statistics.

From section 1.3.1 the true EEG is obtained from:-

$$\hat{eeg}(i) = y(i) - \hat{K}(i)EOG^T(i) \qquad\qquad 1.5$$

where $\hat{K}(i) = [\hat{k}_1(i), \hat{k}_2(i),..., \hat{k}_n(i)]^T$ are the
calculated OA parameters.

$\hat{eeg}(i)$ is the estimate of the true eeg(i)

$EOG^T(i)$ are the sampled EOGs

m is the sample number.

T indicates matrix transposition

Past EEG samples are exponentially weighting to gradually attenuate their effect on the current estimated EEG. Estimation of the Ks is obtained by minimising Werr, the weighted sum of squares of the estimated true EEG, or error term.

$$Werr = \sum_{i=1}^{m} \gamma^{n-i} \hat{eeg}^2(i) \qquad 0 < \gamma < 1 \qquad\qquad 4.6$$

Minimisation of Werr w.r.t the values of K leads to the recursive least squares algorithm (See appendix A).

$$\hat{K}(i+1) = \hat{K}(i) + G[\, y(i+1) - EOG^{T}(i+1)\, \hat{K}(i)]$$ 4.7(a)

$$P(i+1) = \frac{1}{\gamma}[\, P(i) - \frac{1}{\alpha} P(i)\, EOG^{T}(i+1)\, EOG^{T}(i+1)\, P(i)]$$ 4.7(b)

where

$$\alpha = \gamma + EOG^{T}(i+1)\, P(i)\, EOG(i+1)$$

$$EOG^{T}(i+1) = [eog_1(i+1)\ eog_2(i+1)\ ...\ eog_n(i+1)]$$

$$G = P(i+1)\, EOG(i+1) = P(i)\, EOG(i+1)\, / \alpha$$

N.B. P is referred to as the error covariance matrix and G as the gain vector.

The argument $(i+1)$ is used to emphasise the fact that the K are obtained from the last values. γ is referred to as the 'forgetting factor' and prevents the matrix $P(i+1)$ from tending to zero(and $K(i+1)$ to a constant) as i increases, thus allowing the tracking of slowly varying parameters within the signal.

To avoid the inherent numerical instability of the covariance update equation (4.7b), P is factorised using the U-D factorisation method *[Bierman, 1976]*. Appendix A provides a description of this method, and figure 4.26 illustrates a flow diagram representation of the U-D factorisation algorithm.

4.4.3 COMPARISON OF ON-LINE AND OFF-LINE OA REMOVAL.

The comparison of the on-line OA removal with the previous off-line OA removal was considered necessary in order to established the integrity of the on-line removal algorithm. Figure 4.27 illustrates a comparison of OA parameters obtained using the off-line regression model, and the modified RLS algorithm. The variation of the RLS OA parameters with time and signal content is clearly visible. It should also be observed that the RLS OA parameters finally settle to the same as those of the regression model obtained in the off-line investigation.

The settling of the RLS algorithm, and therefore the sensitivity of the algorithm to changes in the signal is related to the GAMMA, or 'forgetting factor', of equation 4.6 above. In figure 4.27 gamma is set equal to 1, so that all previous data estimates have an effect on the current estimate.

4.4.4 OCULAR ARTEFACT PARAMETER PRE-LOADING.

Section 1.4.2 introduced the problems associated with errors in OA removal caused by sudden changes in signal statistics. Such changes are most noticeble with random OA occurrences. Pre-loading of the OA parameters in the removal algorithm, when such a change is detected, was put forward as a possible solution to this problem. Off-line investigation of OA removal has provided a set of averaged OA parameters for a number of
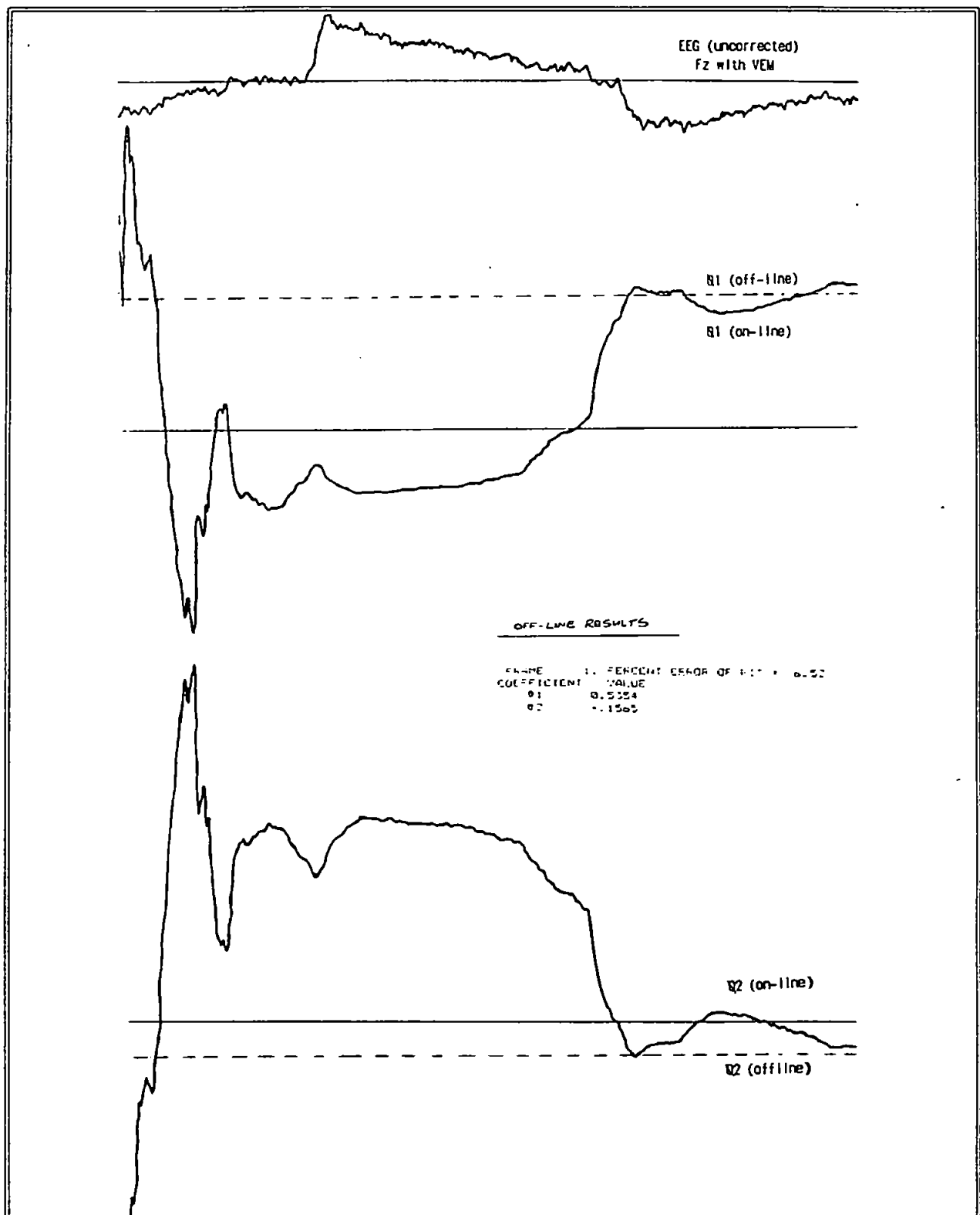
Figure 4.27 Comparison of off-line and on-line OA

parameters whilst removing VEM OA from EEG recorded at Fz.
page 135

electrode sites and OA types (See section 4.3.3.1). These OA parameters reflect the average strength of OA potential for a particular electrode position, and OA type. These parameters therefore represent the pre-loading values for the removal algorithm.

To pre-load the algorithm the following procedure was carried out prior to subjecting the algorithm to the data:-

.   Vector K is loaded with the averaged parameters for OA type and electrode position.

.   Matrix P is loaded to stabilise the parameter updating.

The error covariance matrix P is used by the algorithm to control the adjustment of the vector K at each sample point. Having pre-loaded K it is necessary to pre-load P to prevent wild fluctuations in estimates of K. A suitable value for P is calculated from the data as:-

$$P(m) = [EOG_m{}^T EOG_m]^{-1} \qquad\qquad 4.8$$

where m is the number of samples in the section of data.

P is also determined empirically by observing the values of P after the algorithm has settled, and was observed to between 4.0E-7 and 10.0E-7. This value compromised between preventing fluctuations whilst allowing the tracking of slowly varying parameters.

To illustrate the concept of pre-loading, consider figure 4.28. This illustrates a section of simulated EEG (10 hertz sine wave) contaminated by a large OA (square wave). The estimated true EEG (figure 4.28b) shows a significant error in OA removal as a result of the step change in signal mean at time 4 seconds. This error corresponds to the change in the OA parameter k (figure 4.28c) as the algorithm attempts to re-converge k to a new value. Figure 4.28d shows the squared error (log scale) between true EEG and the estimated EEG. Pre-loading K and P at time 4 seconds removes the error due to re-convergence almost entirely.

Three sections of EEG data have been isolated that contain significant OA contamination from VEM, HEM, and blink artefact respectively. The following section. illustrates the use of OA parameter loading on real EEG data and as an example uses EEG signal measured at electrode F4 and contaminated with VEM type OA. Figure 4.29a illustrates the measured EOG (Fp2-F8) and the contaminated EEG respectively. The EOG is observed to contain two major features: (a) a sharp potential, due to blink, at time 1.3 seconds, and (b) a longer duration and higher amplitude potential, due to VEM, at time 4.3 seconds. Both of these artefacts are measured at F4 where they appear attenuated.

Figures 4.29b and c expand the time axis of 4.29a and illustrate the comparison between OA removal using the conventional removal algorithm, and that obtained using the VEM OA pre-loaded algorithm at time 4.3 seconds. The estimated EEG for conventional OA removal shows the same disturbance as that obtained in the simulated data, i.e. the step change in signal mean at time 4.3 seconds results in an error in the estimated EEG. This error corresponds to the change in OA parameter k (figure 4.29c) as the algorithm attempts to re-converge k to a new value. The error
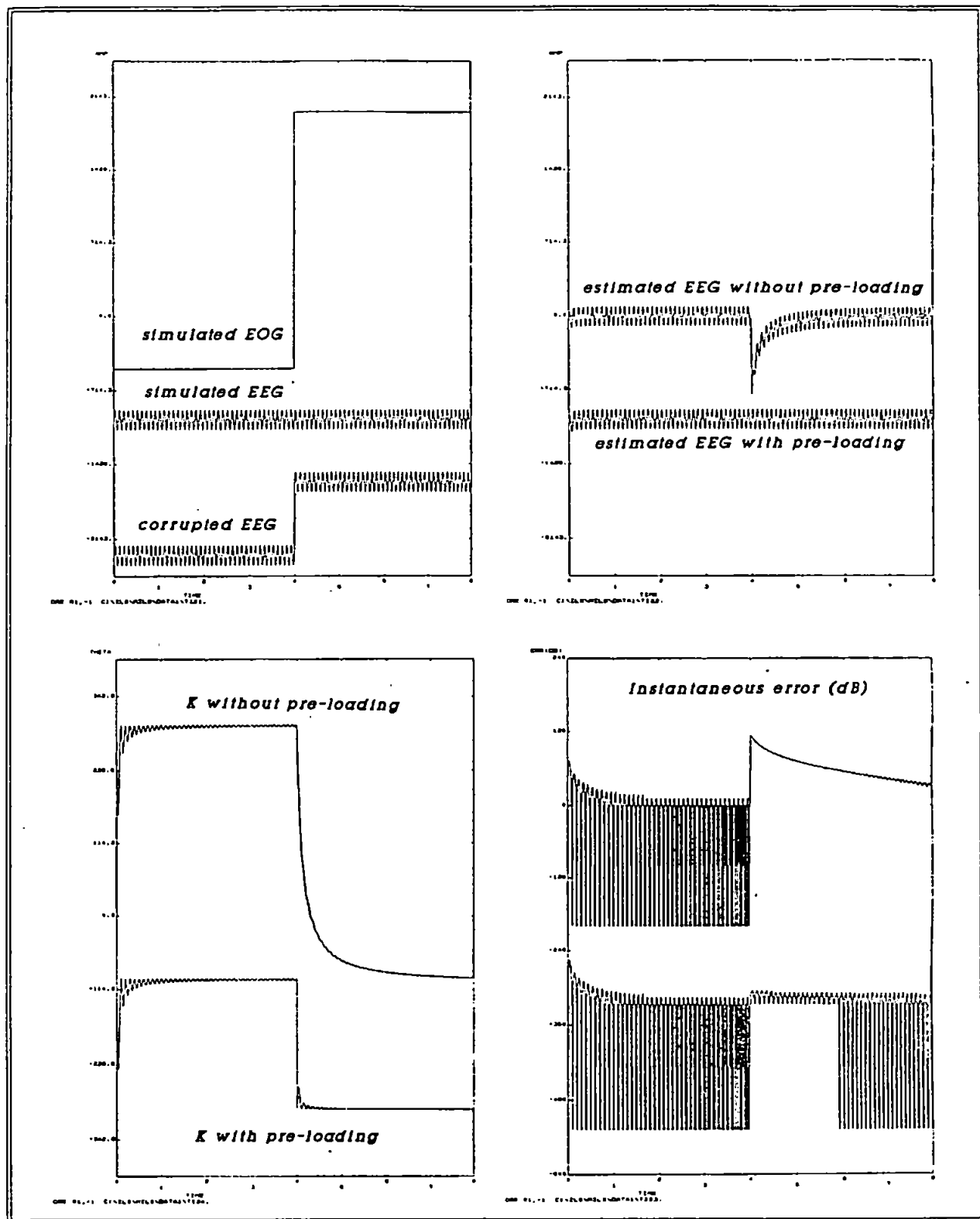
Figure 4.28 Comparison of conventional and pre-loaded adaptive OA removal algorithm using simulated EEG. (a) EOG, genuine EEG and measured corrupted EEG, (b) Corrected EEG with and without OA parameter pre-loading, (c) OA parameters with and without pre-loading, (d) Squared error in OA removal.

page 138
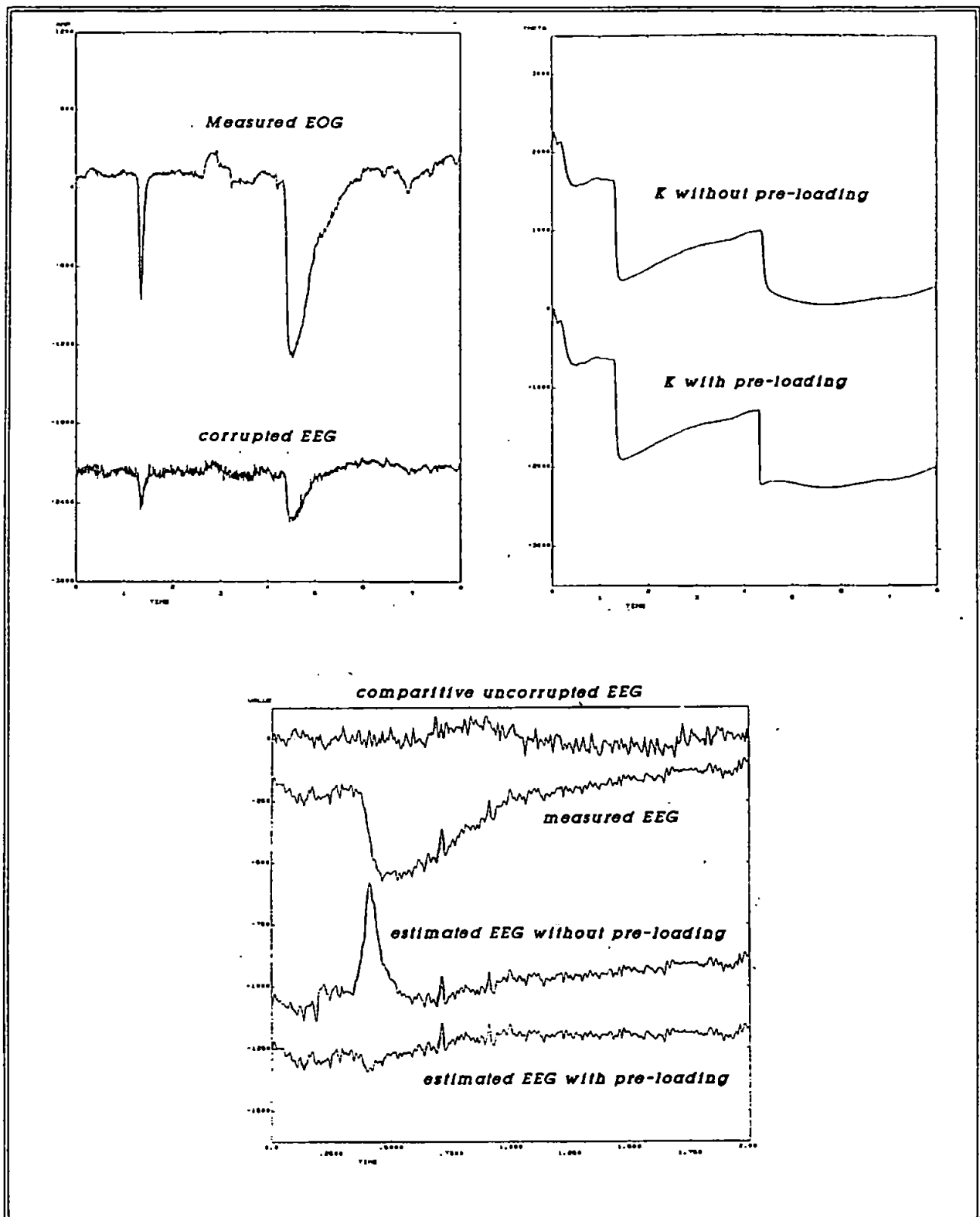
Figure 4.29 Comparison of conventional and pre-loaded adaptive OA removal algorithm using EEG measured at F4 and EOG measured at Fp2-F8. (a) EOG, and measured corrupted EEG, (b) Corrected EEG with and without OA parameter pre-loading, (c) OA parameters with and without pre-loading.

appears as a sharp negative potential in the estimated EEG representing a misleading estimate of the true EEG. This could lead to the interpretation of the waveform as abnormal (see section 1.1.1). Pre-loading K and P at time 4.3 seconds with the average VEM OA parameter reduces this error significantly and is visually assessed as better representing the true underlying EEG. For a quantitative analysis of OA removal and a discussion of the problems involved with this see section 4.4.6.

OA parameter pre-loading is seen to reduce the errors due to re-convergence of the adaptive OA removal algorithm. As such it is transforming the fixed set of updating equations within the algorithm to a selective set that are dependent on the signal contents. This selectivity forces the algorithm to operate in a non-continuous environment where identification of different OAs will force the algorithm into pre-defined states. The following section investigates the use of the 'selective' and adaptive OA removal algorithm in a non-continuous environment.

4.4.5 DIRECTED OA REMOVAL.

To operate selectively, the modified OA removal algorithm discussed above must operate in a non-continuous environment, i.e. the algorithm will only be applied to sections of EEG which have identified OA contamination. To investigate the operation of the algorithm under these conditions consider the following 8-second segment of EEG recorded at Fz and contaminated with blink artefact. Figure 4.30 illustrates the measured EEG and EOG signals. Eight major signal contaminations are observed in the measured EEG which correspond to blinks measured in the EOG.
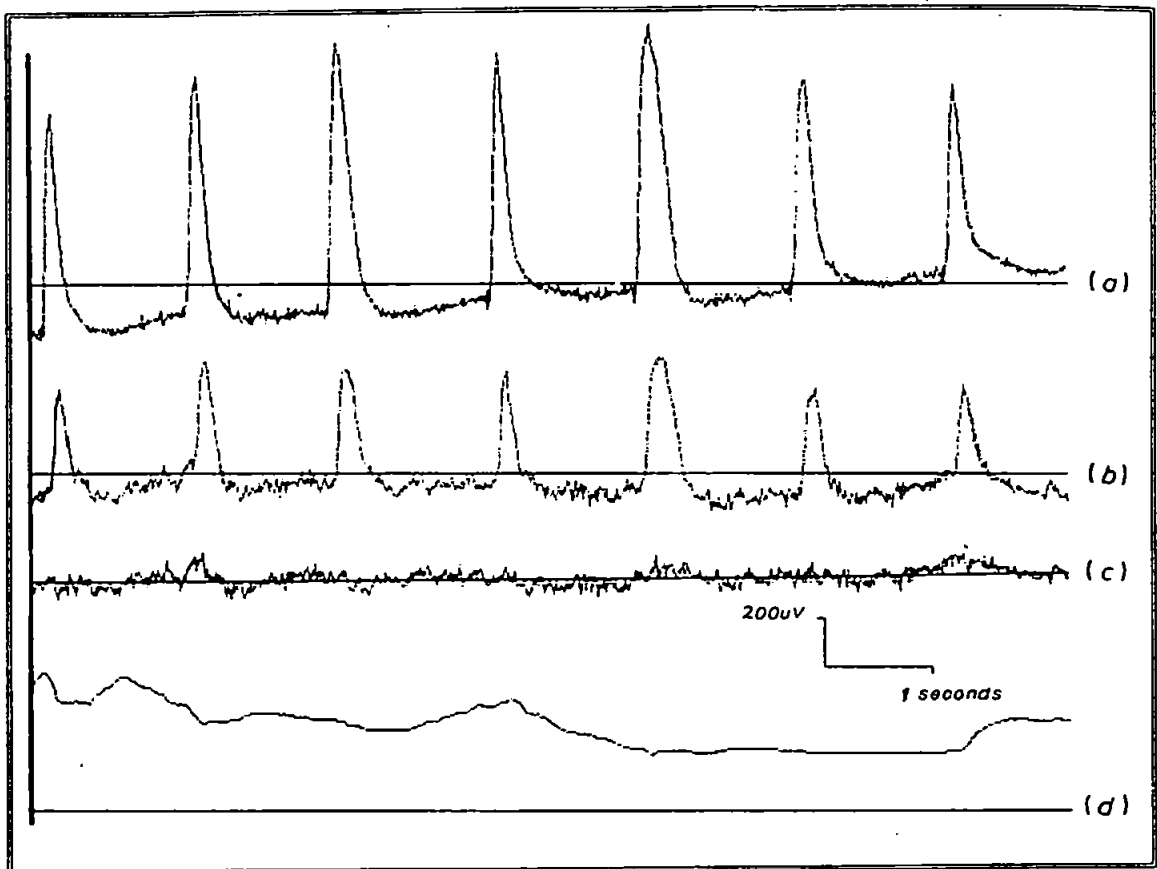
Figure 4.30 EEG measured at FZ corrupted with blink type

artefact. (a) EOG, (b) measured EEG, (c) estimated EEG, (d) K.

Blink OAs were identified with the aid of an EEG expert and the OA boundaries were manually identified from the data, see e.g. figure 4.31.
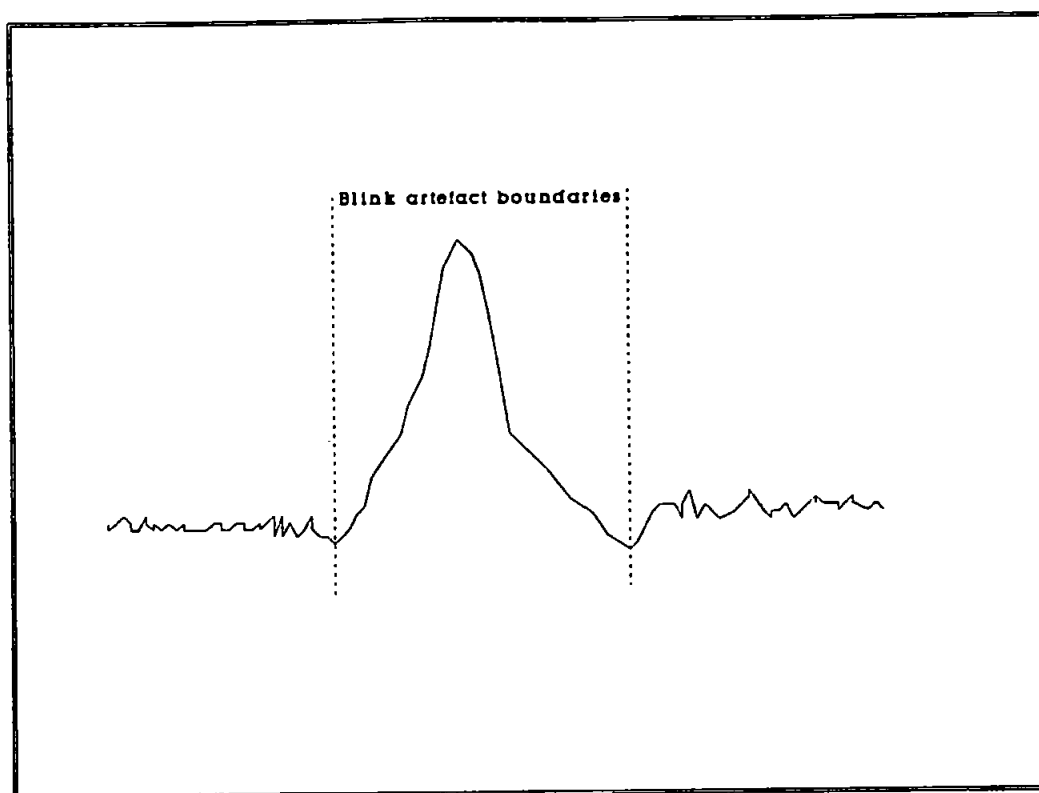
Figure 4.31 Blink artefact boundary identification.

The boundary of each blink was identified as the first minimum, below EEG signal mean value, and on either side of the blink maximum. The OA removal algorithm was 'directed' by applying the algorithm to the data (EOG model 5), only between the artefact boundaries having been pre-loaded with the averaged blink OA parameters.

Three methods of directed algorithm application were considered and figure 4.32 illustrates the results obtained using these methods on the selected data. The three directed methods were:-

The OA removal algorithm is applied continuously to the data but the estimated EEG is only output during artefact. During the inter-artefact period the measured EEG is simply passed through unaffected by the algorithm. This method involves the same number of calculations for each sample point as the conventional algorithm and will, because of re-convergence in the inter-artefact period, result in non optimised OA parameters.

The OA removal algorithm is normally inactive. When an artefact boundary is detected, the removal algorithm is started with the appropriate OA parameters. This method reduces the number of calculations overall because the algorithm is not updated for each sample point, but by starting with the same pre-loading values for each artefact the algorithm will not track intra-subject variations in individual artefact characteristics.

.The OA removal algorithm is normally inactive. When an artefact boundary is detected, the removal algorithm is restarted with the pre-loading values retained from the end of the previous instance of that artefact. For the first instance of artefact, this method is identical to that above, but subsequently the OA parameters are allowed to adapt further and to track intra-subject variations. Detection of the end of artefact will initiate the storage of the pre-loading values ready for the next artefact instance.

(a) estimated EEG using method 1

(b) k: Continuous calculation.

(c) estimated EEG using method 2

(d) k: pre-loaded at the start of each artefact

(e) estimated EEG using method 3

(f) K: pre-loaded at the start, and held at the end of each artefact

Gamma = 0.99

(g) estimated EEG using method 3

Gamma = 0.996

(h) k: pre-loaded at the start, and held at the end of each artefact
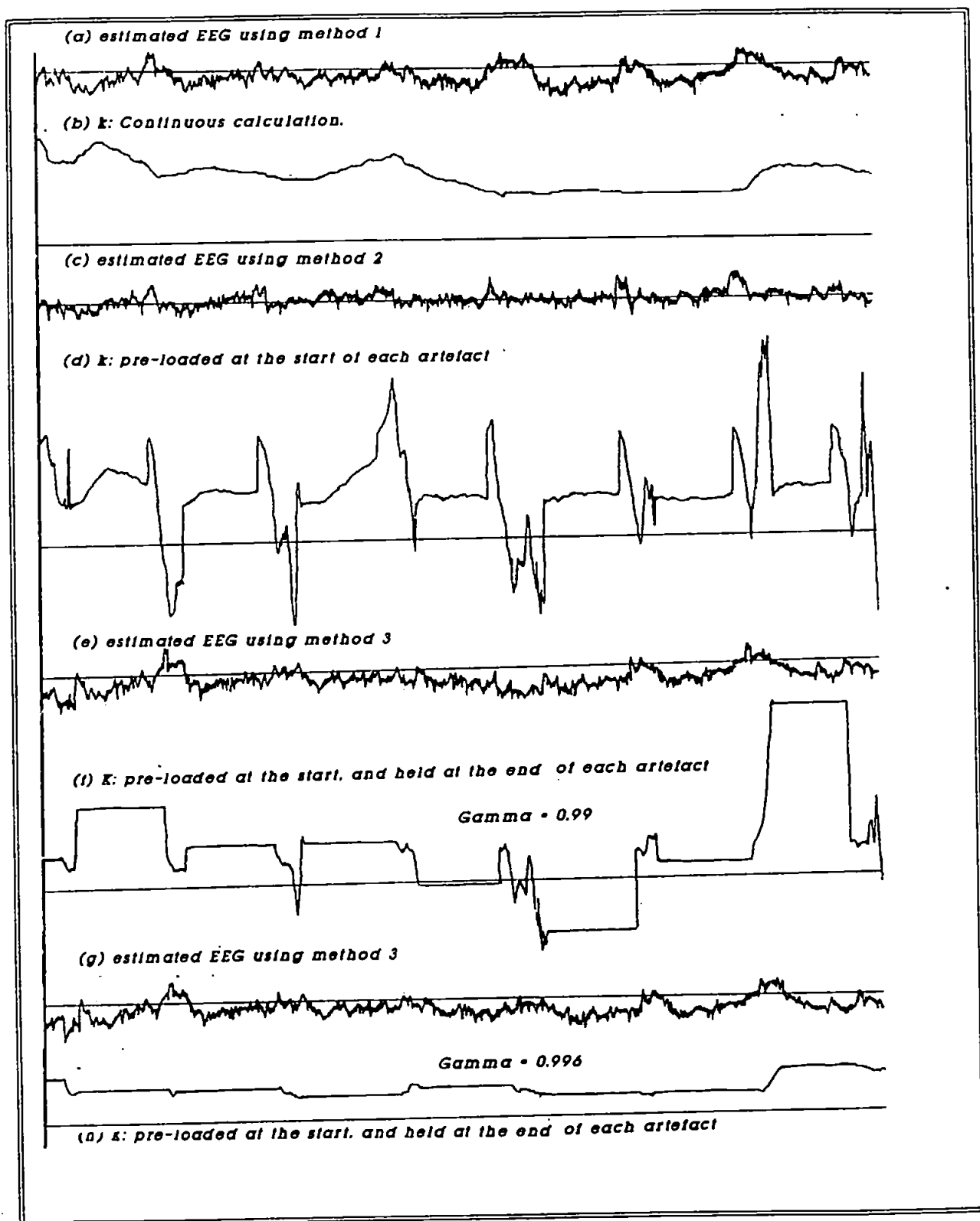
Figure 4.32 Comparison of directed OA algorithm application methods. EEG is the same as in figure 4.30. (a) directed method 1, (b) K for a, (c) directed method 2, (d) K for b, (e) and (g) directed method 3, (f) and (h) K for e.

page 144

Visual qualitative analysis of the results illustrated in figure 4.32 indicated that the selective and directed application of the OA removal algorithm, particularly in the third method of direction, resulted in a more accurate estimation of the true underlying EEG. However, in order to quantify the level of OA removal an approach put forward in section 4.3.5 will be adopted.

4.4.6 QUANTIFYING OA REMOVAL.

Quantification of OA removal is made particularly difficult because of the lack of a measurable 'true' EEG. Several attempts have been made to make such a quantitative comparison of OA removal *[Jervis, et al., 1988; Ifeachor, 1984]* These methods commonly utilise a measurement of correlation between measured EEG and that obtained after OA removal. The principle of this technique being that as the removal algorithm attempts to minimise this correlation, the lower the correlation the better the OA removal. However, as was highlighted in section 4.3.4.2, any correlation is dependent on signal content and as a result cannot be used in a comparison involving different EEG signals.

In an attempt to overcome this problem and to provide a means for continuous quantification of OA removal in an automated system, the following strategy has been developed based on that used by Gratton. Gratton et al. (1983) devised a quantitative means of representing OA removal in evoked EEG potentials by calculating the sum of squares of the deviations of the estimated EEG, obtained after OA removal, from the true

EEG, where the true EEG was taken as that obtained with no significant OA contamination. This method is however inappropriate for general EEG recording because of the inevitable phase and instantaneous signal differences between two sections of EEG. As a result the sum of squares of deviations in the time series is high and misleading.

To extend this method to general EEG recording the phase information is removed by performing a similar calculation on the power spectrum of the estimated EEG after OA removal and the true EEG, where the true EEG is taken as the last segment of EEG with insignificant OA contamination. This produces a measure of spectral correction (MOSC):-

$$MOSC = \frac{\sum_{i=0}^{M}(MEEG_i - REEG_i)^2 - \sum_{i=0}^{M}(CEEG_i - REEG_i)^2}{\sum_{i=0}^{M}(MEEG_i - REEG_i)^2} \times 100\%$$

4.9

where   MEEG is the power spectrum of the measured, OA corrupted, EEG.

and   REEG is the power spectrum of the measured uncorrupted EEG.

and   CEEG is the power spectrum of the estimated EEG obtained after OA removal.

and   i is the frequency sample number.

and   M is the total number of frequency samples

The MOSC is used on a frequency range of 0 - 20 Hertz representing the EEG signal bandwidth; this being indicated by $MOSC_{20}$.

To quantify OA removal, consider the waveforms illustrated in figure 4.33 which uses the same EEG signals illustrated earlier in figure 4.29. Figure 4.33a illustrates the respective power spectrums (0- 20 Hertz) for the signals shown in figure 4.29 (c) and is calculated using the Blackman-Tukey or moving average (MA) method *[Childers, 1978].*

The power spectrums shown are of the reference uncorrupted EEG, the corrupted EEG, and the estimated EEGs respectively. Figure 4.33b and c illustrate the modulus of the instantaneous spectral error between the two estimated EEGs and the reference true EEG.

The MOSC estimates the quantity of OA removal by calculating the summated spectral error, between true and estimated EEGs, as a percentage of the change in spectral content of the reference EEG caused by OA contamination. For a signal which, after OA removal, closely matches the reference spectrum, the spectral error will be minimal and therefore the MOSC will be maximum; 100% MOSC therefore represents zero error between reference and signal spectrums. In figure 4.33 the calculated values of MOSC for the two estimated EEGs is 96% and 98% respectively complementing the visual assessment made in section 4.4.5.
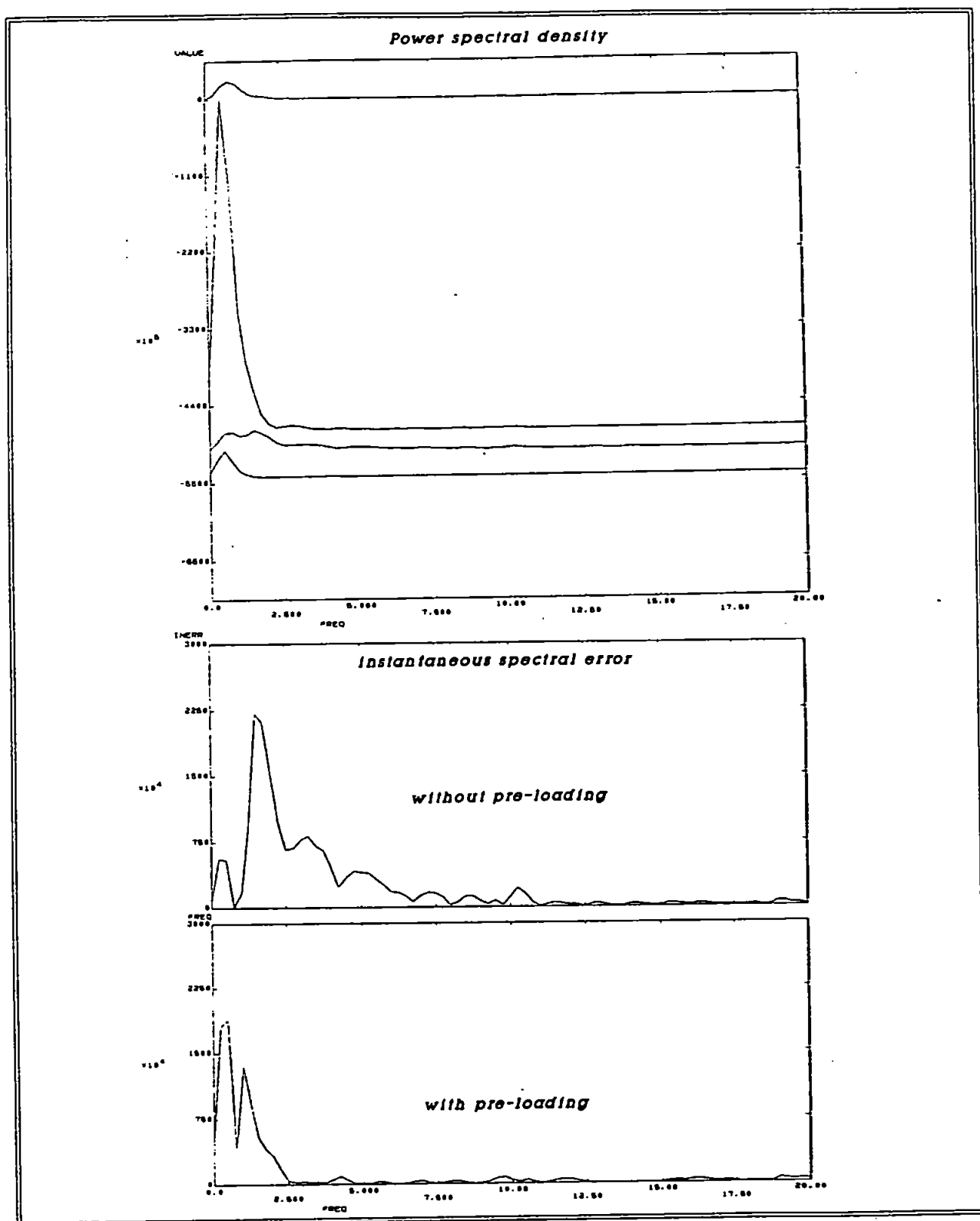
Figure 4.33 Quantification of OA removal. (a) Power spectral density of reference EEG, corrupted EEG, and estimated EEGs, (b) instantaneous spectral error in estimated EEG without pre-loading, (c) instantaneous spectral error in estimated EEG with pre-loading.

page 148

4.4.7 SUMMARY OF RESULTS FROM ON-LINE INVESTIGATION.

Analysis of the results obtained from the investigation of the on-line OA removal algorithm has allowed the following observations to be made:-

Presetting of the ocular artefact parameters allows the optimal point for OA removal, for each artefact/electrode combination, to be converged upon quicker than by conventional means. This results in a decrease in estimation error caused by the algorithm re-convergence

The selective RLS algorithm has been successfully employed in removing VEM and blink OAs from the frontal EEG signals investigated. A comparison of this method with conventional removal techniques demonstrates a qualitative improvement in estimated signal contents.

Directed application of the selective OA removal algorithm minimises unnecessary OA removal allowing uncontaminated EEG to pass through the filter unaffected. The effect of this is to reduce both 'overcorrection' of the EEG, and the introduction of secondary artefacts.

Intra subject variations in OA characteristics during EEG recording are better tracked by allowing the OA parameters to adapt during artefact instances only, and by freezing the OA parameters during the inter artefact period.

Quantification of OA removal using the MOSC allows continual evaluation of filter performance to an uncontaminated reference. However,

use of the MOSC has highlighted a number of shortcomings: (a) The MOSC is insensitive to subtle qualitative differences in OA removal, such as waveform shape. This is clearly observed when comparing the MOSC of the two estimated EEGs in figure 4.39. The MOSC will be biased towards the differences in low frequencies containing greater power. The 'spike' introduced by the conventional removal algorithm contains frequencies outside the MOSC bandwidth and of less power than the low frequency baseline variations and is therefore not represented adequately in the MOSC. (b) The MOSC is susceptible to errors caused by secondary artefacts, such as muscle activity, which will produce a small finite error. (c) The MOSC is susceptible to variations in EEG signal content. The underlying EEG signal can change often and suddenly (c.f. alpha rhythm waxing and waning). Whilst a very recent uncontaminated reference is likely to provide an accurate spectral estimation of that under examination, it is likely that, due to OA contamination, this reference will be of significant temporal disparity to that under examination, and therefore a significant spectral error might be unavoidable. Preliminary studies have shown that a 6% variation in background EEG over a period of 8 seconds is possible.

The directed OA removal algorithm is susceptible to baseline offset in both measured EEG and EOG. This can cause discontinuity in the estimated EEG during the period of algorithm application and result in a qualitative and quantitative dissatisfaction. Baseline offset adjustment is therefore necessary prior to algorithm application.

In conclusion to this stage of the investigation, the results obtained from the use of the new selective and directed OA removal algorithm show significant improvement over conventional application. Selective OA removal is dependent upon correct identification of OA type and directed OA removal is dependant upon correct artefact localisation. This has been achieved up to this point by 'manual' off-line methods with the aid of an EEG expert, and is obviously inadequate for any proposed automated system. The following section investigates methods for automated identification of OA using the knowledge of an experienced EEG clinician. The elicited knowledge, combined with the control structure developed in chapter 2, will enable the 'selective' and 'directed' OA removal algorithm to be applied under the control of the EEG expert knowledge.

4.5 CHARACTERISATION OF ARTEFACTS AND ABNORMAL SLOW WAVEFORMS.

Chapter 2 has identified a method of knowledge representation and described clearly the theoretical aspects of knowledge manipulation, using this representation. The remaining problem is one of eliciting from the expert, a series of IF-THEN, or production, rules that encapsulate the knowledge required to identify OAs and to differentiate them from abnormal waveforms. These rules, once coded, will allow sections of EEG containing significant OA to be identified so that the OA removal algorithm can be applied in a selective fashion.

4.5.1 KNOWLEDGE ELICITATION.

The database of EEG and EOG activity, detailed in section 4.2 has allowed extensive knowledge elicitation to be carried out using techniques detailed in the literature *[Welbank, 1983; Burton, 1987]*. Knowledge elicitation involving interviews with the consultant neurophysiologist and 4 experienced technicians in the department of neurophysiology, at Derriford Hospital in Plymouth have been carried out. Interviews were recorded onto cassette tape and later transcribed. Appendix I gives examples of two such interviews. The objectives of the interviews were: (i) to acquire as much information as possible on the subject of EEG interpretation, with particular emphasis on how the expert approaches the problem, and (ii) to elicit from the expert specific knowledge regarding identification of OA waveforms, and differentiation between OA waveforms and between OA and abnormal slow waveforms. The knowledge elicitation sessions were divided into two parts. Firstly, informal interviews were held to elicit general information, and secondly, structured interview techniques were used to elicit specific rules for the intelligent OA removal system.

4.5.1.1 INFORMAL INTERVIEWS.

Appendix I is the transcription of an informal interview held with a consultant clinical neurophysiologist. The purpose of this interview was to elicit general information on the subject of EEG analysis, leaving the expert to talk reasonably freely on the subject - to elicit course grain

information. Whilst this was informal in the respect that no specific information was being requested, data was presented to the expert on which to base the discussions. This data being drawn from the database of EEG/EOG data recorded using the DAS.

During transcription, an attempt was made to capture any hesitation and uncertainty in an answer by representing a 1 second hesitation by a full stop and uncertainty by the phraseology of the expert. Uncertainty in a rule is an important concept and will be covered later.

Each informal interview consisted of presenting the expert with an EEG paper trace recording. Each recording was of 16 channel EEG/EOG and were recorded using a variety of recording montages. The expert would scan the recording by turning pages forward and backward. This was to build an overall impression of the recording. The two main questions which were asked were (a) to identify any OA activity and (b) to identify any abnormal activity. Any section found to be of interest and used in the discussion was photocopied so that the respective digital data could be identified for analysis. Appendix I deals solely with patient P.R.

Visual EEG analysis is essentially heuristic in nature which is a desirable characteristic of a problem suited to an artificial intelligence (AI) approach. It has been found that the key elements used by the expert in analysing the EEG is the use of spatio-temporal information. Particular attention is paid to the symmetry or otherwise of any suspect wave. Ocular artefacts will tend to appear concentrated in the anterior

regions of the scalp rather than the posterior and also tend to be symmetrical. Figure 4.34 illustrates this on a section of EEG paper trace record.
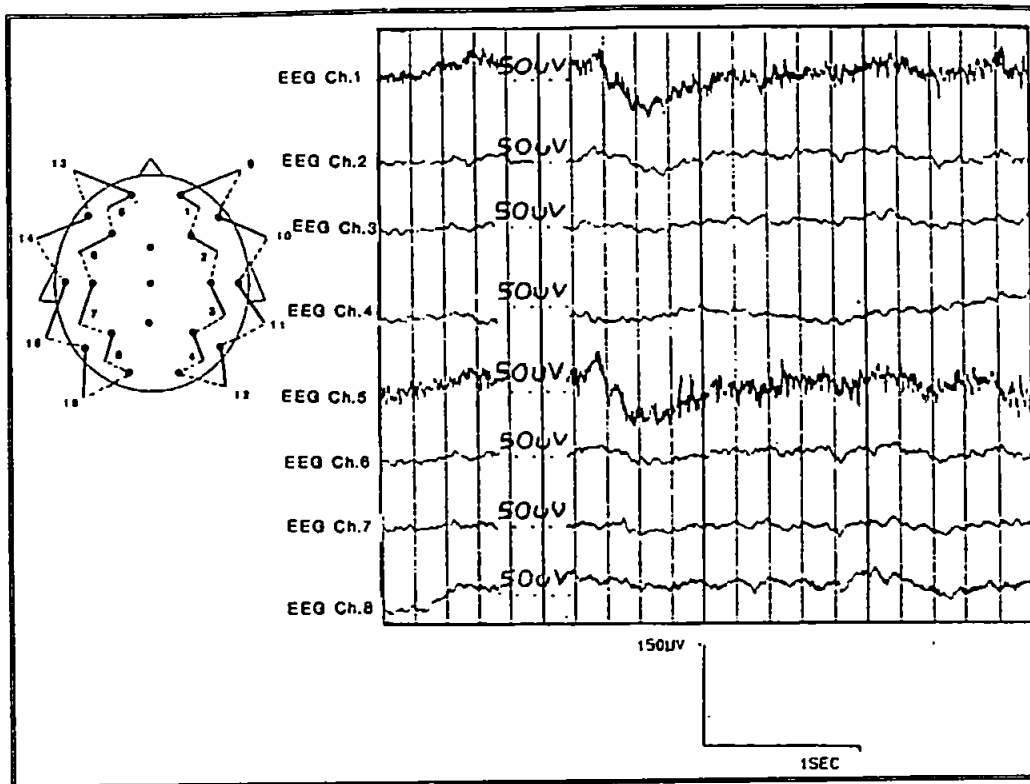
Figure 4.34 EEG signals showing the symmetry and anterior concentration of Ocular artefact.

The phase relationship between signals close to the eyes is also of importance. VEM and blink will produce signals which appear synchronous, whereas HEM will produce convergent signals one side and divergent on the other. Figure 4.35 illustrates this point. The time course of the artefacts tends also to be different. Blinks produce short duration positive potentials above the eye, and small negative potentials below the eye. Eye movement also can have associated with it faster muscle activity produced by the orbital muscles.
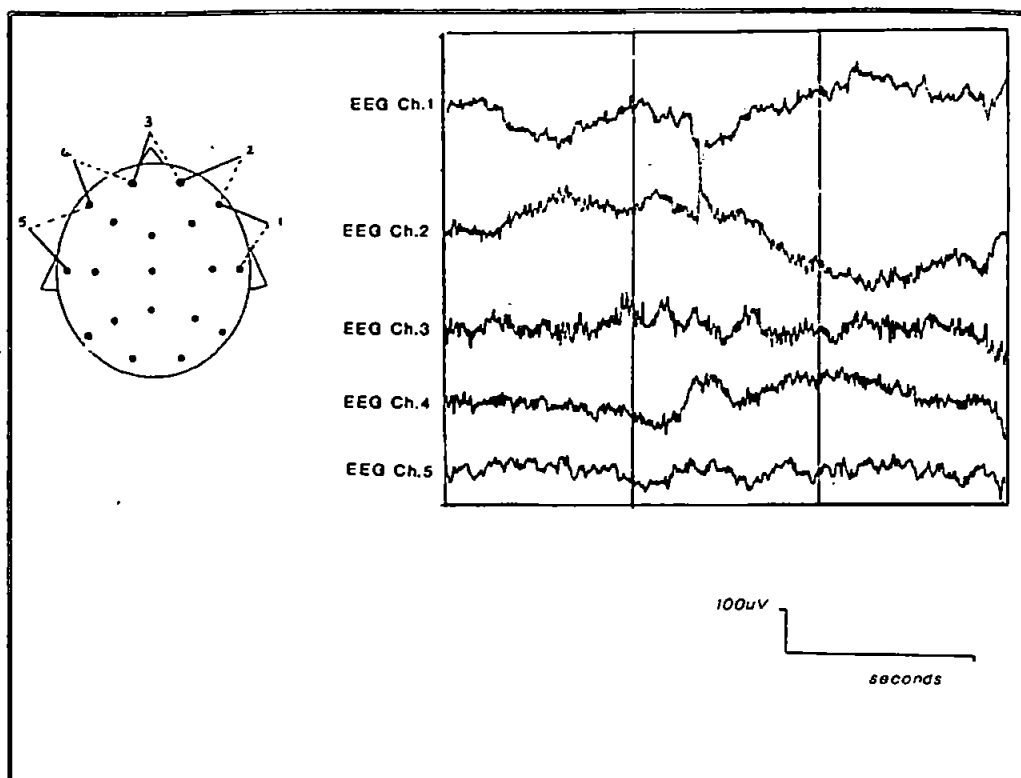
Figure 4.35 EEG showing phase reversal associated
with lateral eye movements.

An abnormal wave will have limited amplitude and time course. However, OAs can have similar appearance, in this case attention will be focused on the potential distribution of the wave. If the wave is confined or predominant to one hemisphere or localised in one part of the scalp this could indicate a tumour. Asymmetry is indicative of abnormality. The synchrony will also be compared at different parts of the scalp. Synchronous slow waves appearing on both sides of the scalp are likely to be abnormal. Figure 4.36 illustrates an EEG trace exhibiting bilaterally synchronous delta (BSD). This shows that the waves are symmetrical, synchronous and present in the frontal electrodes. The current OA removal

page 155
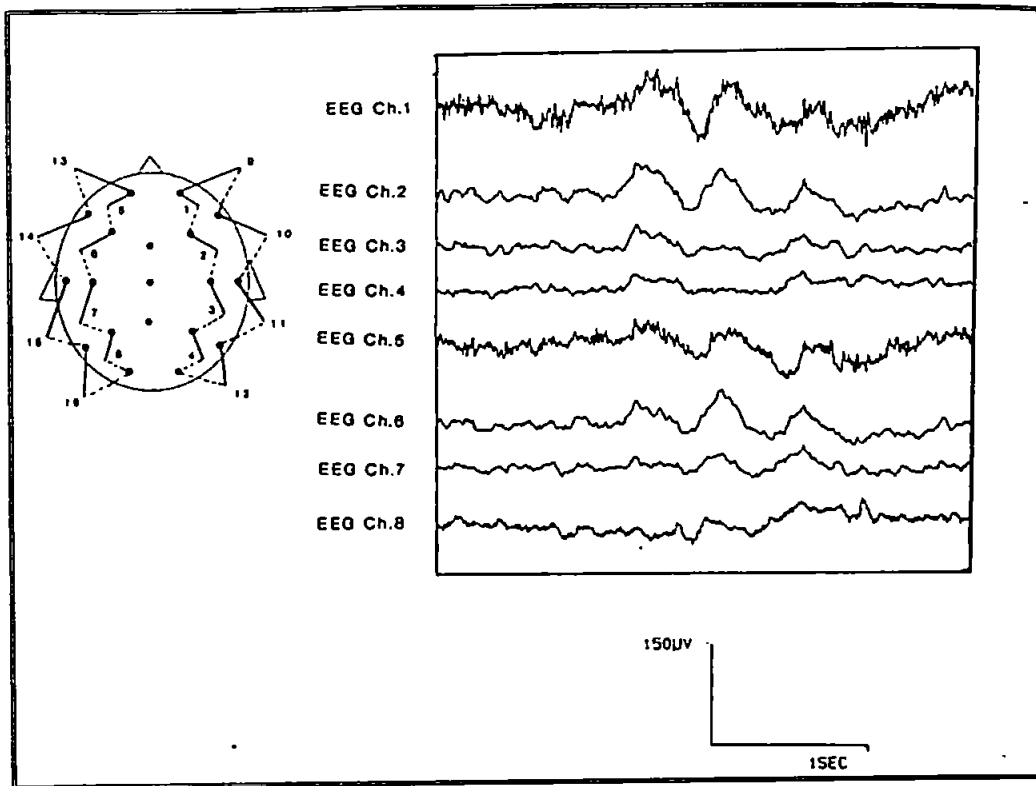
Figure 4.36 Bilaterally synchronous delta waveforms are symmetrical, synchronous and present in the frontal electrodes.

process will significantly attenuate these waves because of their presence in the frontal electrodes. However, the waveforms are of similar amplitude centrally, temporally and frontally, indicating a deeper source than the eyes which will exhibit waves that are more concentrated frontally.

Slow waves are described as more sinusoidal in shape and lack the fast edges of the OAs. They are also often observed with spike activity and therefore the presence of one might reinforce a suspicion of the other. Slow waves will often occur in regular bursts and tend to appear similar over the scalp. OAs tend to be random and appear different depending on

the point they are measured. An exception to this might be rolling eye movements which appear when a patient becomes drowsy and are quite regular in occurrence.

4.5.1.2 STRUCTURED KNOWLEDGE ELICITATION.

Appendix I is the transcription of a structured interview with a consultant clinical neurophysiologist. This is combined with a copy of the data used in this interview. The purpose of this interview was to elicit specific rules for the identification of OA waveforms, and for differentiation between OA and abnormal slow waveforms - to elicit fine grain information.

From the acquired patient data, 16 8-second portions of data were selected. Each portion contained 16-channels of EEG/EOG and were recorded using various standard electrode positions. Each portion was then subdivided into 4, 2-second segments which were to form the 'analysis segments'. Figure 4.37 illustrates this subdivision and is a portion of data used in the transcribed interview of appendix I. Structured interviews, with the expert, were undertaken in order to classify each of the resultant 64 analysis segments into one of the following four categories :-

1. Only artefacts present.

2. Only cerebral signals present

3. Both artefact and cerebral signals present.

4. Neither artefact nor cerebral signal present.
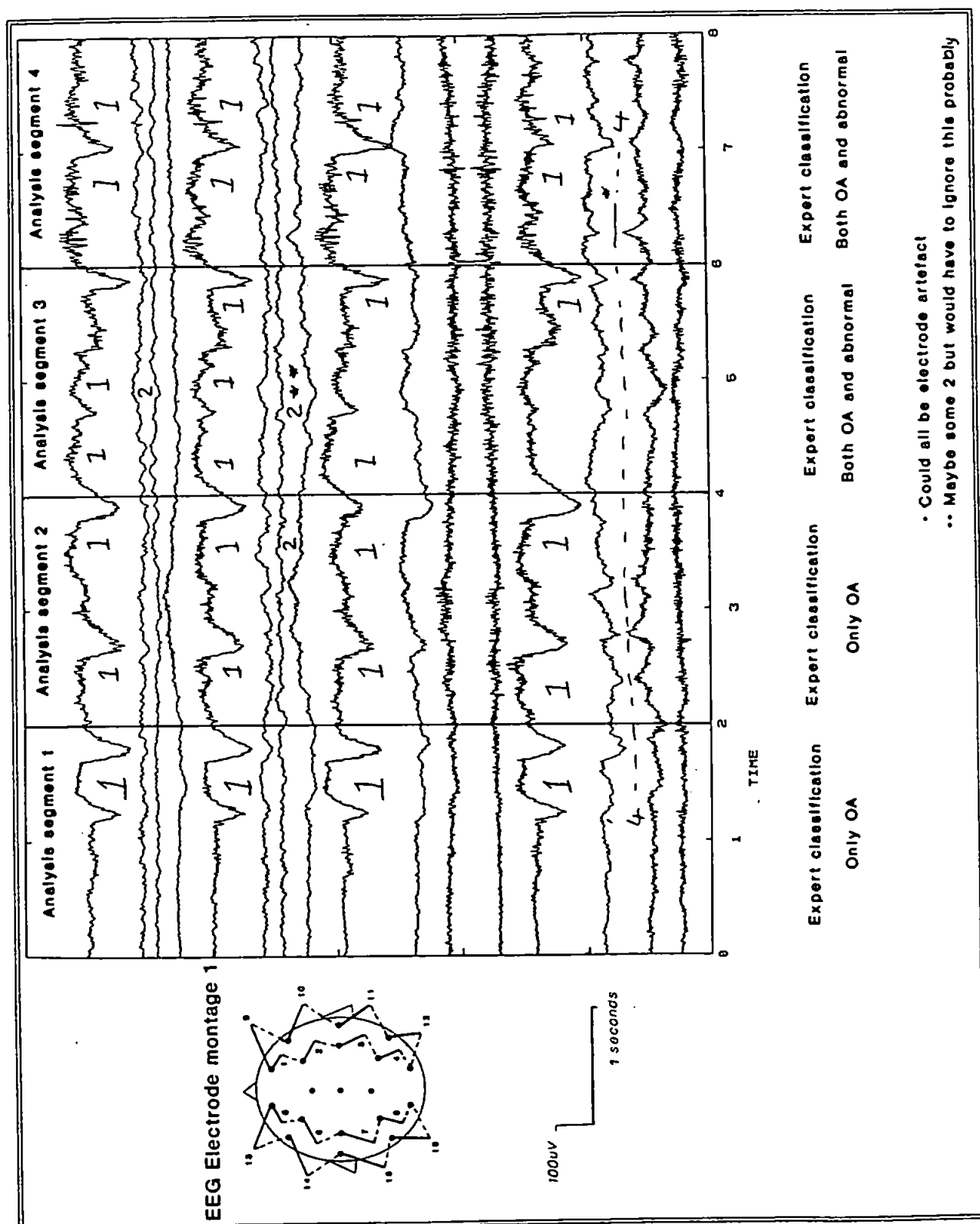
Figure 4.37. Example of data used in structured

knowledge elicitation, showing segmentation.
page 158

All data were presented to 2 EEG experts and the only additional information available to the experts were the individual signal derivations. The purpose of limiting the information available to the expert was to enable rules based on individual signal qualities and inter channel relationships to be separated from those based on contextual information, such as patient history, age, and remaining EEG recording.

During the interview the expert was encouraged to describe any problems found in categorising the segments and to note any points of interest on the appropriate analysis segment. This enabled the most important features that the expert uses to be identified. All such structured interviews were tape recorded and later transcribed. It can be seen in figure 4.37 that waveforms of significance to the expert have been noted by either underlining or by writing the appropriate identification number next to it. The overall classification for each analysis segment is noted at the bottom of each segment and is dependant on the contents of the segment. In the figure, all four segments contain delta slow waves which have been identified as OA. However, the expert has identified segments 2, 3 and 4 as containing additional suspected abnormal slow waves. The abnormal slow waves can be seen in channels 14 and 15 and are identified because of their measurement position on the scalp. Two rules which could therefore be inferred from this portion of data are:

RULE I:

IF slow waves are present on channels 1,5,9 and 13

THEN there is an increased reason to believe that OA is present.

RULE II:

IF slow waves are present on channels 14 and 15

THEN there is an increased reason to believe that non OA is present.

Channels 1,5,9 and 13 are the channels closest to the eyes and therefore it is unlikely that OAs are present if there is no activity in any of these channels.

Identification of OA may not always be possible based on one piece of information or one rule. For example, the expert will look at every waveform in context and will use information obtained from earlier in the recording and other montages as well as patient history information in order to make a decision. It was clear from the knowledge elicitation used in this stage of the investigation, that any rules devised would not have 100% certainty in identification success and careful consideration would therefore have to be given to how a successful rule would contribute to the belief in a final decision. A suspect waveform may not be identified unambiguously using any one piece of information. However, taken collectively, successful rules will increase the belief in a decision.

4.5.2 UNCERTAINTY MANAGEMENT.

Uncertainty is not easily incorporated into the logical representation that has been chosen. Firstly, the first order predicate logic used for

inference does not accommodate values of truth to be other than true or false. This is not realistic as an expert will rarely be able to make such a decision and the features, used by the expert, and obtained from the data will often be noisy so that certain identification is difficult. Secondly, most implementations of PROLOG do not make any extension to the logical inference mechanism to handle uncertainty, although such implementations exist *[Martin, T. 1985]*, they are not yet widely available.

This section details the extensions to the inference mechanism that will need to be made in order to accommodate the uncertainty associated with the knowledge required to identify OA waveforms. Uncertainty management is incorporated into the knowledge representation at two levels. These are uncertainty in the extracted features (antecedent level) and uncertainty in the rules themselves (consequent level).

4.5.2.1 FEATURE UNCERTAINTY.

Uncertainty at the antecedent level may be caused by a combination of measurement error, natural biological variations in signals, and externally introduced noise. For example, rule I above requires that each of channels 1, 5, 9 and 13 contain slow waveforms, or waveforms with frequencies in the delta EEG frequency band. Traditional binary logic will require that a suspect waveform either has a frequency in this band, or it doesn't. However, this binary classification is not always easy and small natural variations in the frequency of a waveform, combined with the

unavoidable inaccuracy involved in frequency estimation using a fast Fourier transform (FFT), will vary the measured frequency of the waveform. Traditional binary logic offers no flexibility to accommodate these variations and as a result, a waveform can be wrongly classified as not being a slow waveform even when its frequency lies barely outside the delta frequency band. However, an EEG expert is more likely to classify the waveform as slow but with a little less certainty under these conditions and therefore the traditional binary delta frequency band set of figure 4.38(a) is more naturally replaced by the fuzzy delta frequency band set of figure 4.38(b):-
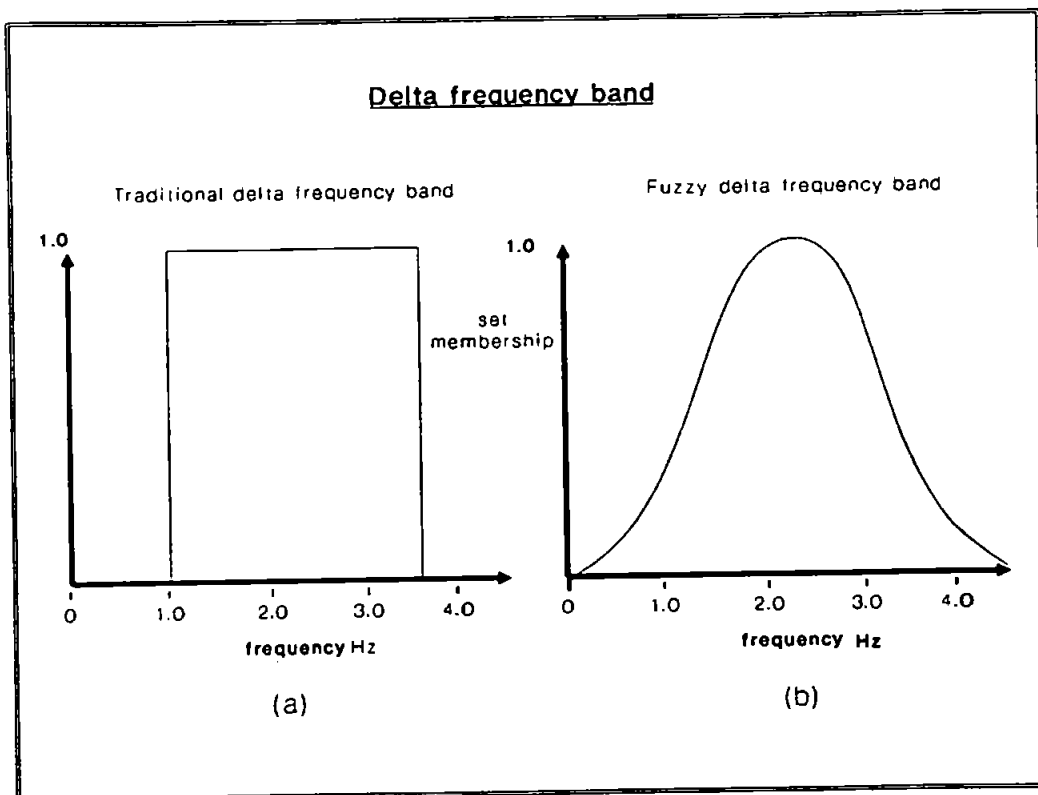
Figure 4.38 Comparison of traditional binary set and intuitive Fuzzy set for the representation of delta EEG frequencies.

Figure 4.38(b) shows that a waveform is attributed to the delta EEG frequency band if its frequency lies between 0Hz and 4.5Hz. The Y axis quantifies this as the set membership where the delta frequency band is the set. This representation is viewed as the certainty that a waveform has a frequency in the delta EEG band, and is described as the delta band 'Fuzzy set'. This representation is an example of fuzzy set theory *[Zadeh, 1965]* and extends first order logic by allowing truth values to have any value between 0 and 1. The Fuzzy set illlustrated in figure 4.38(b) can be simplified by considering that the output of the FFT algorithm is a discrete sampled frequency spectrum. This simplification of the 'discrete' Fuzzy set is illustrated in figure 4.39.

Delta frequency band

Discrete Fuzzy delta frequency band
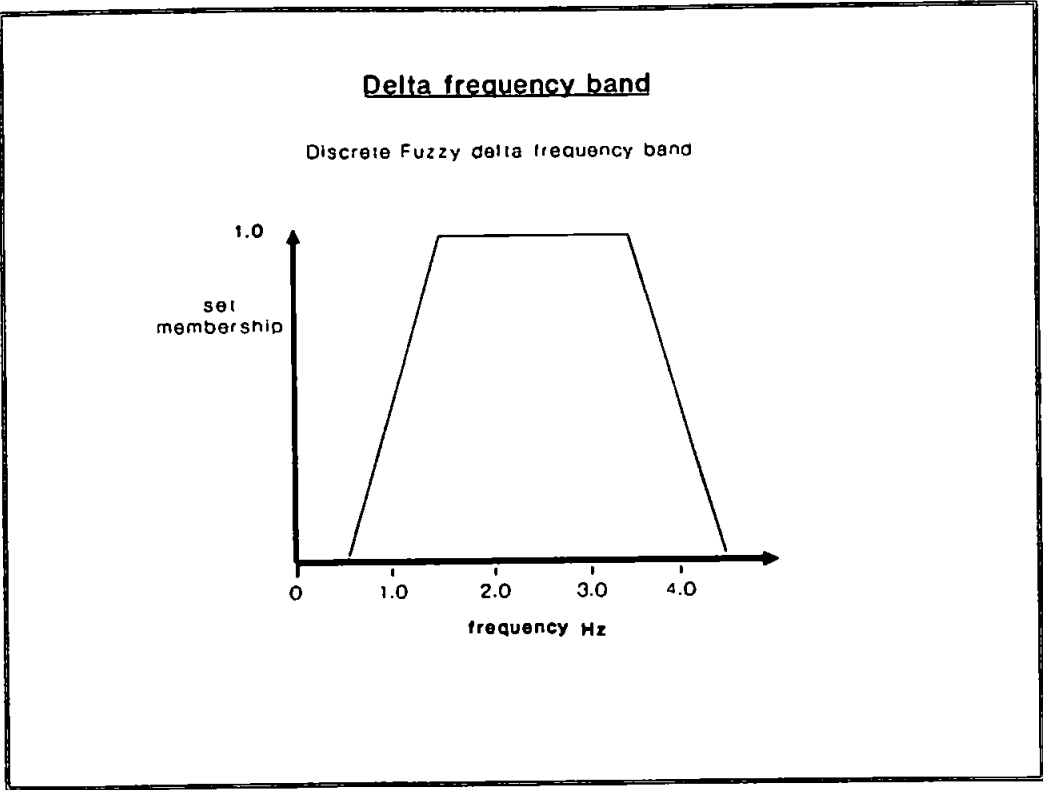
Figure 4.39 discrete fuzzy set for the delta EEG frequency band.

The delta frequency Fuzzy set is extended to the remaining EEG frequency bands in figure 4.40 illustrating each as a Fuzzy set.
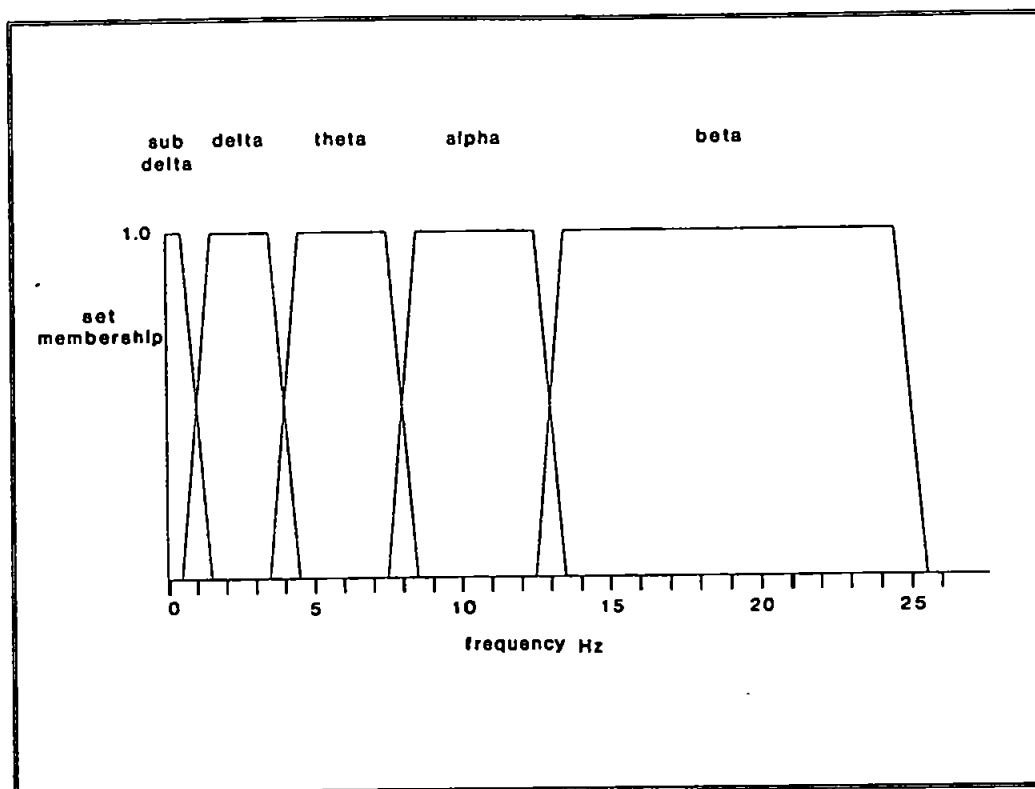
Figure 4.40 EEG frequency band fuzzy sets.

The frequency of a waveform will be attributed to a particular frequency set if its frequency lies within the range of the set. Its membership of the set will depend on its frequency. The more central to the set the frequency is, the greater the certainty of its membership. For example, rule I can be rewritten as:

RULE I:

IF delta waves are present on channel 1

AND delta waves are present on channel 5

AND delta waves are present on channel 9

AND delta waves are present on channel 13

THEN there is an increased reason to believe that OA is present.

A frequency transformation of the signals of figure 4.37 shows that the slow waves in channels 1,5,9 and 13 have a frequency of approximately 2.5Hz. Referring to figure 4.40 it can be seen that this frequency falls in the Delta frequency set with a set membership of 1.0. Each clause in the IF statement of rule I will therefore be true with a certainty of 1.0.

Waveforms with a frequency near the boundaries of the conventional EEG frequency bands (e.g. Delta band 0-4Hz) will be interpreted as belonging to two EEG frequency bands. The membership of the sets will vary according to the frequency and a small variation in the frequency will only change the value of this membership.

The combined certainty for multiple IF clauses in a rule is calculated as the minimum of the individual certainties.

$$AC(Rule) = \min \{ cert(clause\ 1),...,cert(clause\ n) \}$$

where   AC(Rule) is the combined antecedent certainty.

cert(Clause) is the fuzzy set membership.

n is the number of antecedent clauses to the rule.

This relationship is covered in depth in Kandel, 1982.

$$AC(Rule\ B) = min\ \{\ 1.0, 1.0, 1.0, 1.0\ \}$$
$$AC(Rule\ B) = 1.0$$

4.5.2.2 RULE UNCERTAINTY.

Consequent certainty represents the confidence level in a rule. i.e how often a rule is likely to have both antecedent and consequent satisfied simultaneously. This is a number ranging from 0 to 1 and has the effect of attenuating the certainty of the rules combined antecedent clauses.

$$OC(Rule) = RC(Rule)\ x\ AC(Rule)$$

where OC(Rule) is the overall certainty in the truth of the rule

RC(Rule) is the rule confidence.

For example if the confidence in Rule B was 0.8, OC(Rule B) is calculated as:

$$OC(\text{Rule B}) = 0.8 \times 1.0 = 0.8$$

Two methods were used to elicit the rule confidence values:-

. Heuristic values elicited from the experts.

. Statistical values derived from the acquired data.

Heuristc values were initially used in order to develop a prototype intelligent OA removal system. The EEG expert was requested, during structured knowledge elicitation, to attribute, to each rule, a measure of confidence. This measure was mutually agreed to be on a scale of 1 to 5, with 5 representing almost complete confidence in the rule and 1 representing very little confidence in the rule. The measure of confidence was then scaled linearly to the range of 0 to 1 respectively to provide the individual values of RC(Rule). The 1 to 5 scale chosen for the experts measure of confidence was chosen to be a compromise between being vague and being over precise. A scale of 'very little', 'average', 'almost complete' was considered too vague and would present the problem of translating the scale into a numeric value. A scale of 1 to 100 was considered too precise for the expert to be able to satisfactorily discriminate between similar numbers.

Structured knowledge elicitation enabled statistical values for the rule confidence to be compiled. From the 64 analysis segments used to generate the rules a ratio was calculated for each rule :-

$$\text{FORS} = \frac{\text{number of segments satisfying both rule antecedent and consequent}}{\text{total number of segments satisfying rule antecedent}}$$

This ratio was identified as the frequency of rule satisfaction (FORS) with a standard error calculated as:

$$SE = \sqrt{F(1-F)/N}$$

assuming an approximate Binomial distribution, where N is the total number of segments and F is the proportion of N which satisfies the conditions to the rule.

Table 4.2 compares heuristic rule confidence values elicited from the EEG expert to the FORS (frequency of rule satisfaction) for a sample of the rules elicited from the expert.

Rule	Heuristic	F.O.R.T	STD Error
3	.8-1.0	0.93	+/-0.046
4	.8-1.0	0.667	+/-0.046
5	.6-.8	0.857	+/-0.046
6	.6-.8	-	+/-0.046
7	.6-.8	0.667	+/-0.046
8	.4-.6	-	+/-0.046
10	.6-.8	0.5	+/-0.046
12	.4-.6	0.5	+/-0.046
13	.2-.4	0.33	+/-0.046
14	.8-1.0	0.8	+/-0.046

$$\text{F.O.R.S} = \frac{\text{number of segments satisfying both rule antecedent and consequent}}{\text{total number of segments satisfying rule antecedent}}$$

Table 4.2 Comparison of heuristic and statistical rules

of confidence for a selection of rules.

4.5.2.3 BELIEF PROPAGATION.

The belief in the classification of suspect waveform at any point in the inference path will be a function of the rules used to identify it and the certainty of evidence found to satisfy the rules. A final decision of whether or not to remove an OA, will be based on the final waveform classification and the belief in this classification. For example, the expert user could decide that OA removal should only be carried out when an OA waveform is identified with a measure of belief greater than 80%.

The final measure of belief is calculated by updating the accumulated measure of belief from previous successful rules. Two principles are important for this to be carried out:-

- Approximate reasoning *[Zadeh, 1976]*
- Certainty factors *[Shortliffe, 1976]*

Approximate reasoning uses and extends fuzzy logic in an attempt to reason with uncertainty. Antecedent clauses to rules which return fuzzy truth values can be propogated through to fuzzy consequent clauses - a fuzzy relation. For example rule I above may be simplified to:-

RULE I:

IF delta waves are definitely present on frontal channels

THEN the presence of OA is more likely.

The truth of both rule antecedent and consequent are represented by fuzzy sets. The fuzzy set representing the antecedent was illustrated in figure 4.40. The fuzzy set representing the conseqent is represented in figure 4.41.

OA likelyhood fuzzy set
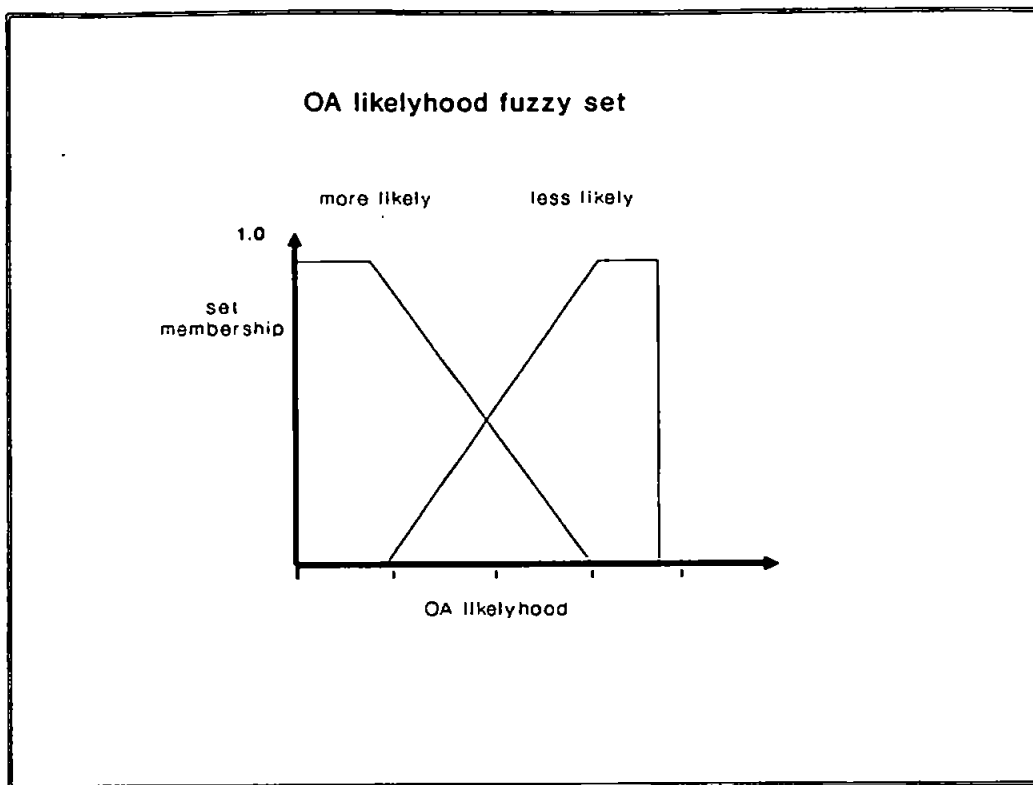
Figure 4.41 Rule consequent fuzzy set.

The rule can be represented as the product of these sets and is represented as a fuzzy set which is the Sup-Min composition [GRAHAM AND JONES, 938] of the two. This is shown geometrically in figure 4.42.

The output fuzzy set is determined by taking a section of figure 4.42(b) at the point of measured frequency. This is illustrated at frequencies of 2.5Hz and 3.5Hz and is represented by sections x and xx respectively. The output fuzzy set for these frequencies are illustrated in figure 4.43
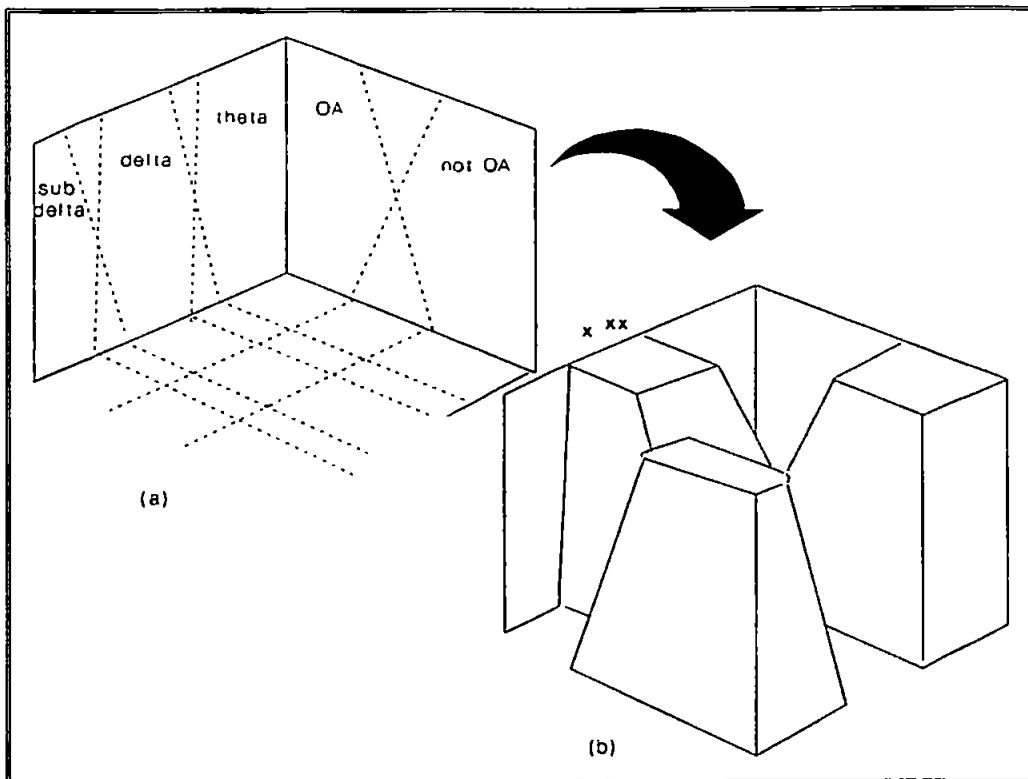
Figure 4.42 Sup-min composition of the antecedent and consequent.
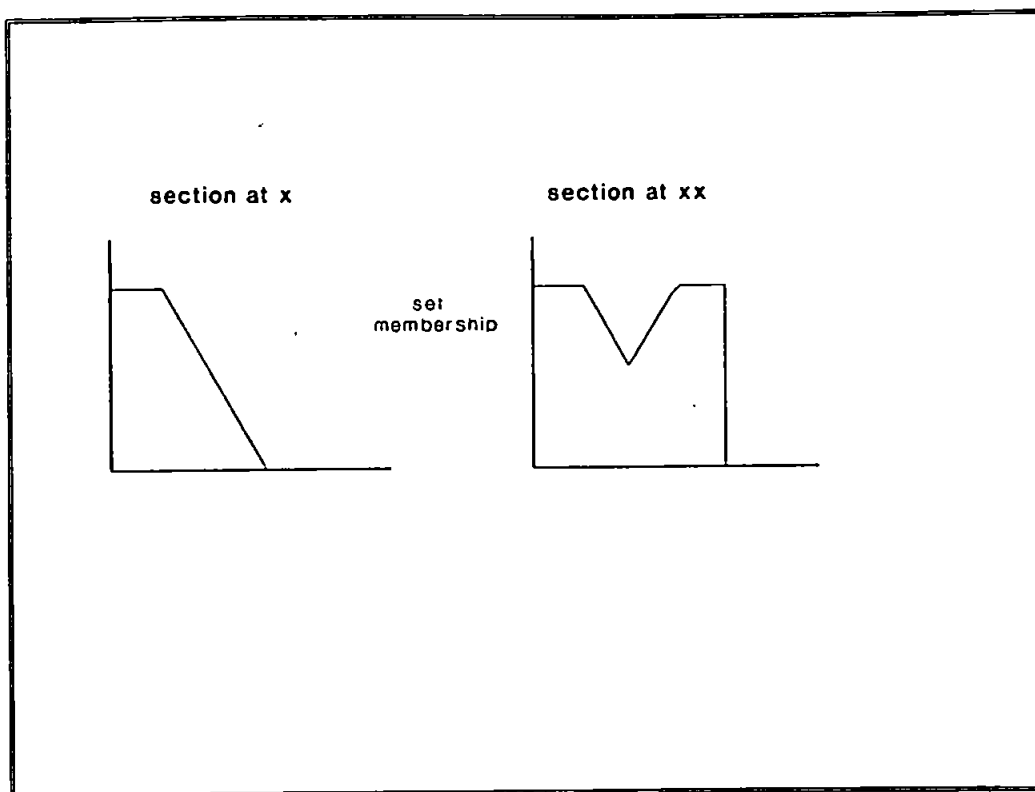
Figure 4.43 Output fuzzy sets showing likelyhood of OA for

waveform frequencies of 2.5Hz and 3.5Hz respectively.

The output fuzzy set will be passed to further rules which will use this set as an antecedent. For example, to invoke OA removal the following rule would use likelihood of OA fuzzy set.

RULE III

IF the presence of OA is very likely

THEN apply the appropriate OA correction algorithm.

The final measure of belief is calculated from the least squares distance metric of the fuzzy output set and the 'OA very likely' fuzzy set:-

$$MB = 1 - \sum_{i=1}^{N} (\mu_A - \mu_B) / N$$

where: μ_A is the previous rule output fuzzy set.

μ_B is the final rule antecedent fuzzy set.

and   N is the number of elements in the set.

Certainty factors *[Shortliffe 1976]* assumes that each successful rule provides further evidence to support the belief in a classification and each unsuccessful rule provides evidence to support the disbelief in a classification. The certainty in the classification of a waveform, C, at any time given evidence E is therefore calculated as the measure of belief (MB) in C given E minus the measure of disbelief (MD) in C given E.

$$CF(C:E) = MB(C:E) - MD(C:E)$$

where MB and MD both range from 0 to 1.

A simplification of the certainty factor principle is used for the classification of suspect waveforms, and assumes that only successful rules are used in the inference path, therefore removing the need for MD. For example, given the current measure of belief in a classification, MB(C:E), derived from previous rules and previous evidence E, than upon new evidence, OC(Rule), from a successful new rule, the new measure of belief, MB'(C:E,OC(Rule)), is calculated from :-

$$MB'(C:E,OC(Rule)) = MB(C:E) + \{ OC(Rule) \times [1 - MB(C:E)] \}$$

where   MB'(C:E,OC(Rule))   is   the   new   measure   of   belief   in classification C, given evidence E and OC(Rule).

In this manner successive new evidence will increase the measure of belief in a classification asymptotically towards 1, or complete belief.

4.6 SUMMARY OF CHAPTER 4.

Chapter 4 has provided several key results to demonstrate the success of the 'selective' and 'directed' approach to OA removal. It is clear that a significant improvement in OA removal is achievable using OA dependent adaptive filters only in sections of EEG containing OA contamination, leaving uncontaminated EEG uneffected. Further, inter artefact retention of OA parameters allows for tracking of intra subject artefact variations.

Extensive data acquisition and knowledge elicitation with experts in EEG analysis have provided a set of rules for the identification of OAs and for the differentiation of OAs and abnormal cerebral slow waves. Important features, such as frequency, potential distribution, correlation and contextual patient details, have been identified which form the critical feature primitives on which the expert will operate in order to identify OA. Key enhancements to the inference engine have been devised to allow reasoning which more closely matches that of the expert, including the use of uncertainty management and Fuzzy logic.

REFERENCES FOR CHAPTER 4.

Bajpai, A.C., Mustoe, L.R., and Walker, D.

"Advanced engineering mathematics", John Wiley and Sons, Ltd. 1983

Berg, P.

"The residual after correcting event-related potentials for blink
artefacts", Psychophysiology, Vol. 23, pp. 354-364, 1986.

Bierman, G.J.

"Measurement updataing using the U-D factorization", Automatica, Vol. 12,
pp. 375-382, 1976.

Burton, M. and Shadbolt, N.

"A formal evaluation of knowledge elicitation techniques for expert
systems: Domain 1". Proc. SERC workshop in knowledge acquisition for
engineering applications, pp. 20-28, 1987.

Burton, M. and Shadbolt, N.

"Knowledge engineering". Technical report 87-2-1 to appear in
Williams, N. and Holt, P. "Expert systems for users". McGraw-Hill.

Childers, D. G. (editor)

"Modern spectrum analysis". IEEE Press, 1978.

Gratton, G., Coles, M.G.H. and Donchin, E.

"A new method for off-line removal of ocular artifacts". Ibid., Vol. 55,
pp. 468-484, 1983.

Ifeachor, E.C.

"Investigation of OA in the human EEG and their removal by a micro-
processor-based instrument, PhD thesis, Plymouth Polytechnic, England,
1984.

Ifeachor, E.C., Jervis, B.W., Allen, E.M. and Hudson, N.R.

"A new microcomputer-based online ocular artefact removal (OAR)
system". IEE proceedings, Vol. 133, Pt. A, No. 5, July 1986.

Ifeachor, E.C., Jervis, B.W., Allen, E.M, Morris, E.L., Wright, D.E.
and Hudson, N.R

"Investigation and comparison of some models for removing ocular
artefacts from EEG signals: Part 1 Review of models and data
analysis". Med. & Biol. Eng. & Comput., 26, pp 584-590, Nov. 1988.

Ifeachor, E.C., Jervis, B.W., Allen, E.M, Morris, E.L., Wright, D.E.
and Hudson, N.R

"Investigation and comparison of some models for removing ocular
artefacts from EEG signals: Part 2 quantitative and pictorial
comparison of models". Med. & Biol. Eng. & Comput., 26,
pp 584-590, Nov. 1988.

Jervis, B.W., Ifeachor, E.C., and Allen, E.M.

"The removal of ocular artefacts from the electroencephalogram:
a review". Med. & Biol. Eng. & Comput., 26, pp. 2-12, Jan. 1988.

Kandel, A.

"Fuzzy techniques in pattern recognition". Wiley, 1982.

Kernighan, B. W. and Ritchie, D. M.

"The C programming language (second edition)". Prentice Hall software
software series, 1988.

Martin, T.P., Baldwin, J.F., and Pilsworth, B.W.

"FPROLOG - A fuzzy PROLOG interpreter", ITRC report 50, University of
Bristol, 1985.

Shortliffe, E.H.

"Computer based medical consultations: MYCIN". American, Elsevier. 1976.

STI (Signal Technology Inc.)

"ILS "(Interactive Laboratory system), 5951 Encina Road, Goleta, CA 93117,
U.S.A.

Terrell, T.J.

"Introduction to digital filters", Macmillan Publishers Ltd., 1985.

page 176

Verleger, R., Gasser, T. and Möcks, J.

"Correction of EOG artefacts in event-related potentials of the EEG:
Aspects of reliability and validity". Psychophysiology, Vol. 19,
pp. 474-480, 1982.

Welbank, M.

"A review of knowledge acquisition techniques for expert systems".
Martlesham Consultancy Services, British Telecommunications. 1989.

Whitton, L., Leu, F. and Moldofsky, H.

"A spectral method for removing eye movement artifacts from the EEG",
Electroencephalography and clinical neurophysiology, Vol. 44, pp. 735-
741, 1978.

Zedeh, L.A.

"Fuzzy sets". Information and control, Vol. 8, 1965.

Zimmermann, H.-J.

"Fuzzy set theory and its applications". Kluwer-Nijhoff. 1986.

CHAPTER 5

INTELLIGENT OA REMOVAL SYSTEM IMPLEMENTATION

Chapters 2 and 4 have provided the necessary foundations for the implementation of an intelligent OA identification and removal system. This includes :-

. Compilation of a knowledge base of rules to identify OA and to differentiate between OA and abnormal slow waveforms.

. Identification of a knowledge representation that will accommodate the rules.

. Description of an inference mechanism that will identify OA using the elicited rules and key spatio-temporal features.

. Identification of a software language that will allow implementation of the knowledge representation and inference mechanism.

. Development of the inference mechanism to accommodate uncertainty.

. Improvements to the OA removal algorithm to incorporate multiple OA dependent filters.

This chapter details the implementation of the Intelligent OA Removal System (IOARS) using standard personal computer (PC) technology and commercial software languages. The decision to use PC technology is founded on the progressive and inevitable movement towards computer based patient data recording and the universal use of PC tools. PC technology provides powerful computing facilities and also enables the development of a sophisticated graphical user interface.

5.1 OVERVIEW.

Figure 5.1 illustrates a conceptual block diagram of the Intelligent OA Removal System (IOARS). The IOARS can be divided into five functional blocks which are identified in Figure 5.1 using broken lines. The five functional blocks are:-

. Signal pre-processing.

. Feature extraction.

. Reasoning.

. Adaptive filters

. User interface.
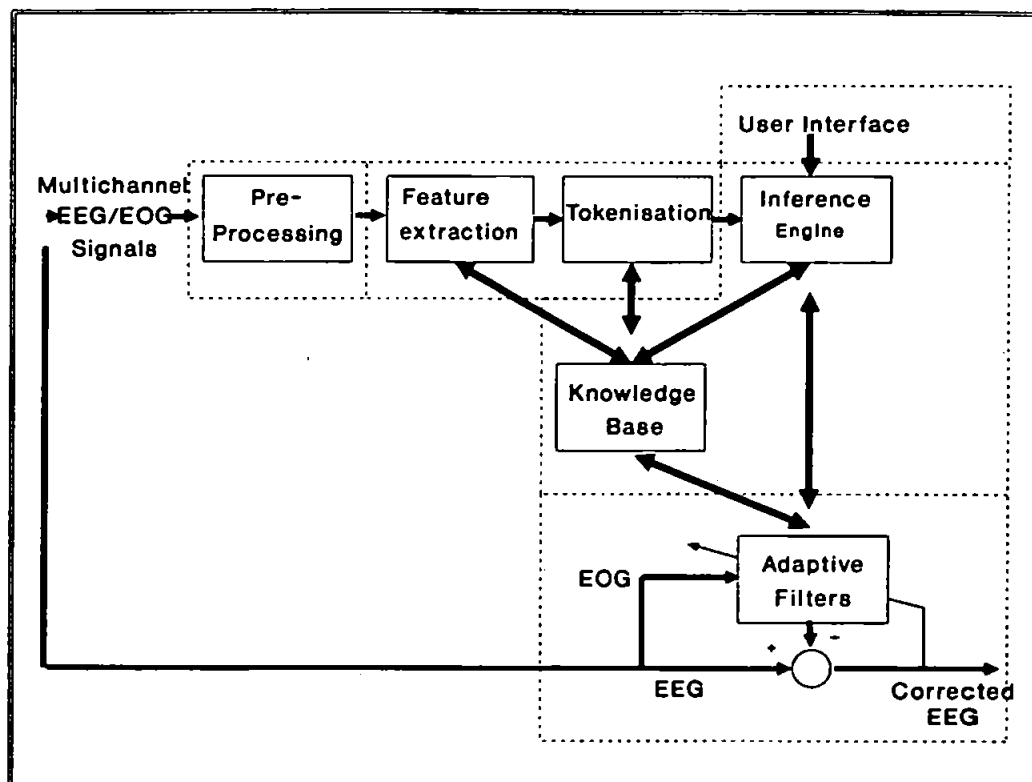
Figure 5.1 Intelligent Ocular Artefact Removal System (IOARS).

page 179

The overall function of figure 5.1 is to provide intelligent removal of OA from the EEG. This is achieved by allowing reasoning to control the OA removal algorithm and therefore adaptive filter. Reasoning employs the use of an inference engine that operates on a number of rules, contained in a knowledge base, and a number of key features extracted from the EEG/EOG. Multi-channel EEG/EOG signals are preprocessed to enhance feature extraction and features are tokenised into a symbolic form to allow manipulation by the inference engine. The knowledge base performs a central role in the operation of the IOARS and contains the elicited rules, details of channel relationships, tokenisation symbols, and lastly filter pre-loading parameters. The knowledge base is therefore a dynamic structure allowing knowledge flow in both directions from relevant functional blocks.

Each functional block has been implemented in software using one, or a combination, of the following languages:-

. PROLOG - *[Clocksin and Melish, 1987]*.

. C - *[Kernighan and Ritchie, 1988]*.

. 80286 Assembler - *[Murray and Pappas, 1986]*.

The dialects of PROLOG and C used were Turbo PROLOG (Borland International USA) and Turbo C (Borland International USA) respectively. Appendix K contains the software listings for all software routines and forms the system software.

PROLOG was used primarily for knowledge representation and inference. Secondly however, its power of symbolic manipulation was utilised in feature extraction, for tokenisation of numerical signal features. C was used to enhance and complement PROLOG in areas of numerical manipulation. These are areas where PROLOG is at its weakest and where the fast and efficient qualities of C are of most benefit. In addition to C, Assembler was used for maximum speed in areas of extreme numerical intensity. This included graphical image manipulation in the user interface. PROLOG was used to maintain ultimate control in the system, with the additional languages providing utility functions. As such the system was linked into one complete program to operate on the PC. The use of software packages from the same vendor eased the problems associated with interfacing multiple languages into one system.

The operation of the IOARS in its present form is essentially off-line. This is due to the software simulation of digital signal processing (DSP) functions such as multi-channel Fast Fourier Transform (FFT). Initial development has utilised the ILS (Signal Technology Inc. USA) software package for execution of all DSP algorithms. This however is extremely time consuming for multi-channel signals and therefore calculations are completed prior to analysis. A final implementation of the IOARS is likely to comprise of multi, DSP dedicated, processor units to enable real time operation.

5.2 SIGNAL PRE-PROCESSING.

Signals are acquired by the system in 8-second, 16-channel blocks, such as that illustrated in figure 4.37. This corresponds to standard EEG signal montages and provides 2048 samples of each signal/channel. Prior to extraction of key features from these signals two pre-processing operations are carried out in order to maximise the relevant information; these are:-

. Mean removal.

. Frequency band limiting.

Preprocessed signals provide a single left and right EOG signal and 16 EEG signals that reflect the activity in the left, right, front and back regions of the scalp. Finally, each 8-second block is divided into 4, 2-second analysis segments in preparation for feature extraction and signal analysis.

5.2.1 MEAN REMOVAL.

DC offset in a signal often occurs as a result of improper recording electrode calibration and variations in skin electrode resistance. The result of any DC offset is minimal in the time domain but is amplified in the frequency domain, due to the greater energy content, and produces a large DC frequency component. Due to the low frequencies involved in EEG and OA analysis, this large DC component can often mask underlying frequency content. DC offset is therefore minimised at the pre-processing stage by subtracting the mean from each signal.

5.2.2 BAND LIMITING FILTERS.

Analysis of the spectral characteristics of the EEG and OA (See section 1.1 - 1.2) has shown that most of the energy lies in the regions 0 - 30 Hertz and 0 - 5 Hertz, respectively. For this reason EOG signals are band-limited to 0.5 - 5 Hertz and EEG signals are band-limited to 0.5 - 30 Hertz. Band limiting EOG signals has the advantages of further attenuating very low frequency potentials caused by skin/electrode resistance variations and also attenuating higher frequency cerebral potentials which appear as artefacts in the EOG. Band limiting EEG signals again further attenuates very low frequency potentials, and also attenuates external sources of artefact such as muscle and electrical activity.

The band limiting of the EEG/EOG signals involved the design of three simple filter types:-

. low pass - 1 : cut off frequency = 5 Hertz.

. low pass - 2 :cut off frequency = 30 Hertz.

. high pass : cut off frequency = 0.5 Hertz.

The three filter types were used in combination, as illustrated in figure 5.2.
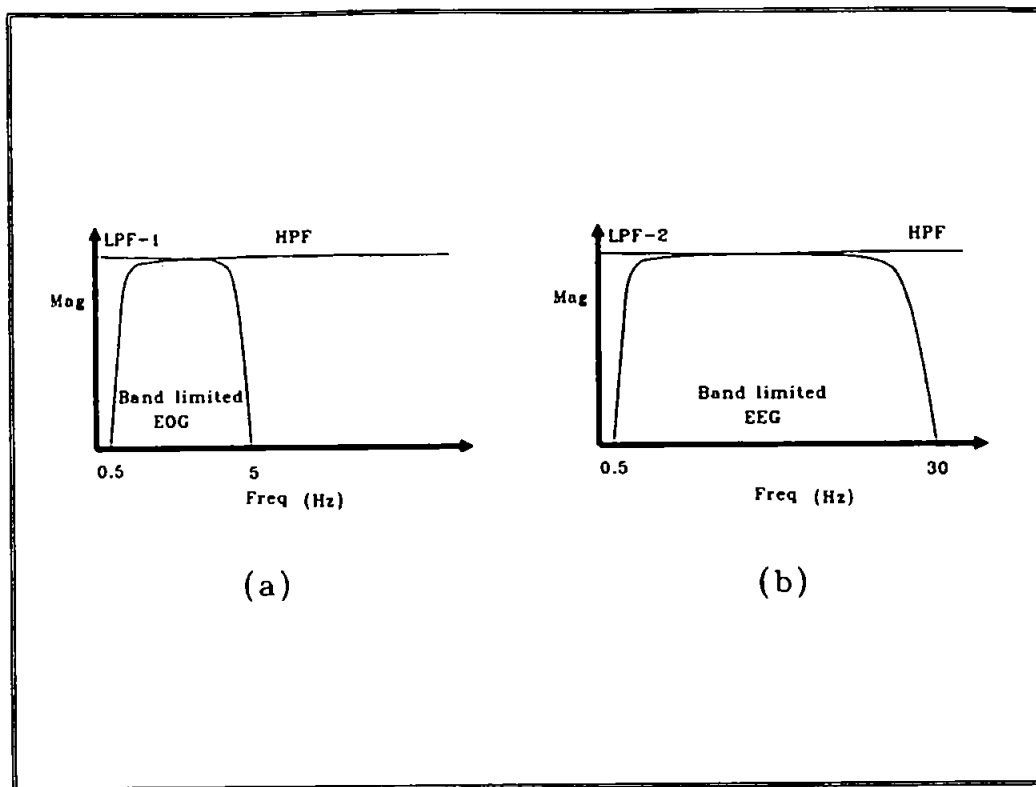
Figure 5.2 Band limiting EEG and EOG signals using simple filter
types. (a) band-limited EOG signal, (b) band-limited EEG signal.

Finite impulse response (FIR) digital filters *[Johnson, 1989]* were
chosen for all filters to allow a linear phase response over the critical
pass band. The following specifications were set for the three filter
types:

Low pass filter 1

.   cut off frequency: 5 Hertz

.   transition width: 1 Hertz

.   passband ripple: < 0.5dB

.   stopband attenuation: >20dB.

Low pass filter 2

.   cut off frequency: 25 Hertz

.   transition width: 5 Hertz

.   passband ripple: < 0.5dB

.   stopband attenuation: >20dB.

High pass filter

.   cut off frequency: 0.5 Hertz

.   transition width: 0.4 Hertz

.   passband ripple: < 0.5dB

.   stopband attenuation: >20dB.

Implementation of the filters was made more difficult by the low transition width to sampling frequency ratio. Approximate empirical relations have been obtained *[Rabiner and Gold, 1975]* that give satisfactory relations between the filter parameters and allow estimation of the number of required filter coefficients for the above specifications. These relations are:-

$$N = 1 + \frac{D_\infty(\delta_1, \delta_2) - f(\delta_1, \delta_2)}{\Delta F} \quad F \qquad\qquad 5.1$$

where:
  N is the required number of coefficients.
  ΔF is the transition width, normalised to the sampling frequency.
  δ_1 is the passband ripple.
  δ_2 is the stopband ripple.
  $D_\infty(\delta_1, \delta_2) = [0.005309(\log_{10}\delta_1)^2 + 0.07114 \log_{10}\delta_1$
  $- 0.4761] \log_{10}\delta_2 - [0.00266(\log_{10}\delta_1)^2$
  $+ 0.5941 \log_{10}\delta_1 + 0.4278]$
  $f(\delta_1, \delta_2) = 0.51244 \log_{10}(\delta_1/\delta_2) + 11.01$

dB Passband ripple is given by:

$$Ap = 20 \log(1 + \delta_1) \qquad\qquad 5.2\,(a)$$

dB stopband attenuation is given by:

$$As = 20 \log(\delta_2)$$

The ratio of transition width to sampling frequency is critical in determining the number of coefficients required for filter approximation and for the above specifications N is calculated, using equation 5.1 as:

Filter	N
LPF-1	203
LPF-2	61
HPF	525

The large number of coefficients required make filter coefficient approximation impractical using the optimal method (For which the above equations are derived). Filter coefficient approximation was therefore accomplished using the window method *[Rabiner and Gold, 1975; Oppenheim and Schafer, 1975]*.

For an FIR filter, the coefficients are approximated, using the window method, from the truncated impulse response h(n) of the desired filter. If $H_D(e^{jw})$ is the desired filter magnitude response, the impulse response $h_D(n)$ is calculated using the inverse Fourier transform.

$$h_D(n) = 1/2\pi \int_{-\pi}^{\pi} H_D(e^{jw}) \, e^{jwn} \, dw \qquad 5.3$$

The filter coefficients h(n) are calculated by weighting the desired impulse response $h_D(n)$ with an appropriate window function W(n).

$$h(n) = h_D(n) \, W(n) \qquad 5.4$$

where W(n) for a Hanning window is:

For odd N:

$$W(n) = 0.5 + 0.5\cos[2\pi n/(N-1)] \quad -(N-1)/2 < n < (N-1)/2 \qquad 5.5$$

For even N:

$$W(n) = 0.5 + 0.5\cos\left[\frac{2\pi(2n+1)}{2(N+1)}\right] \quad -(N/2) < n \ (N/2-1) \qquad 5.6$$

Figure 5.3 - 5 illustrate the magnitude and phase responses for the filters LPF-1, LPF-2, and HPF respectively.
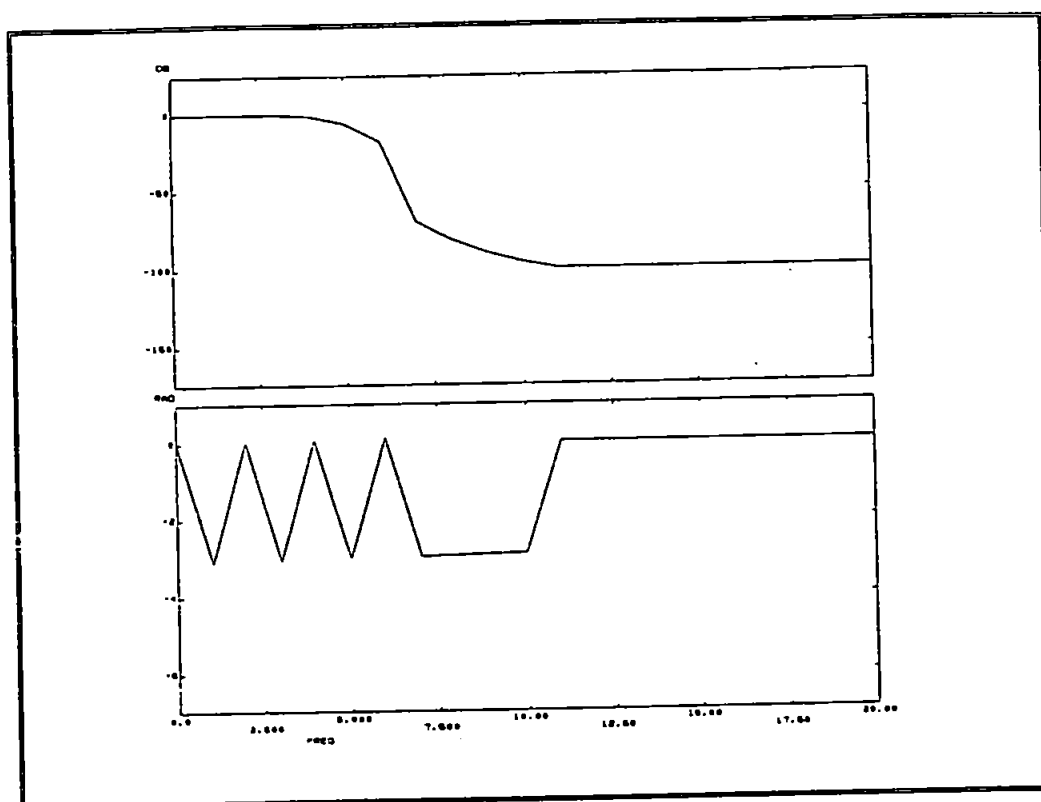
Figure 5.3 Magnitude and phase response of LPF1.
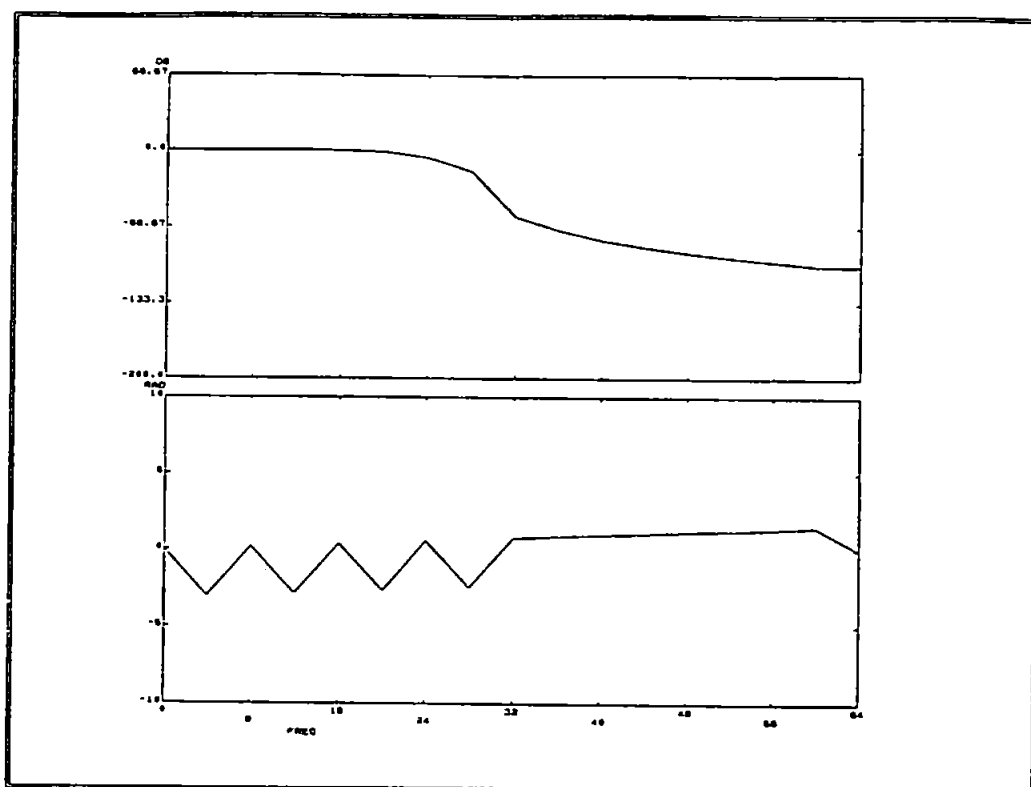
Figure 5.4 Magnitude and phase response of LPF2.
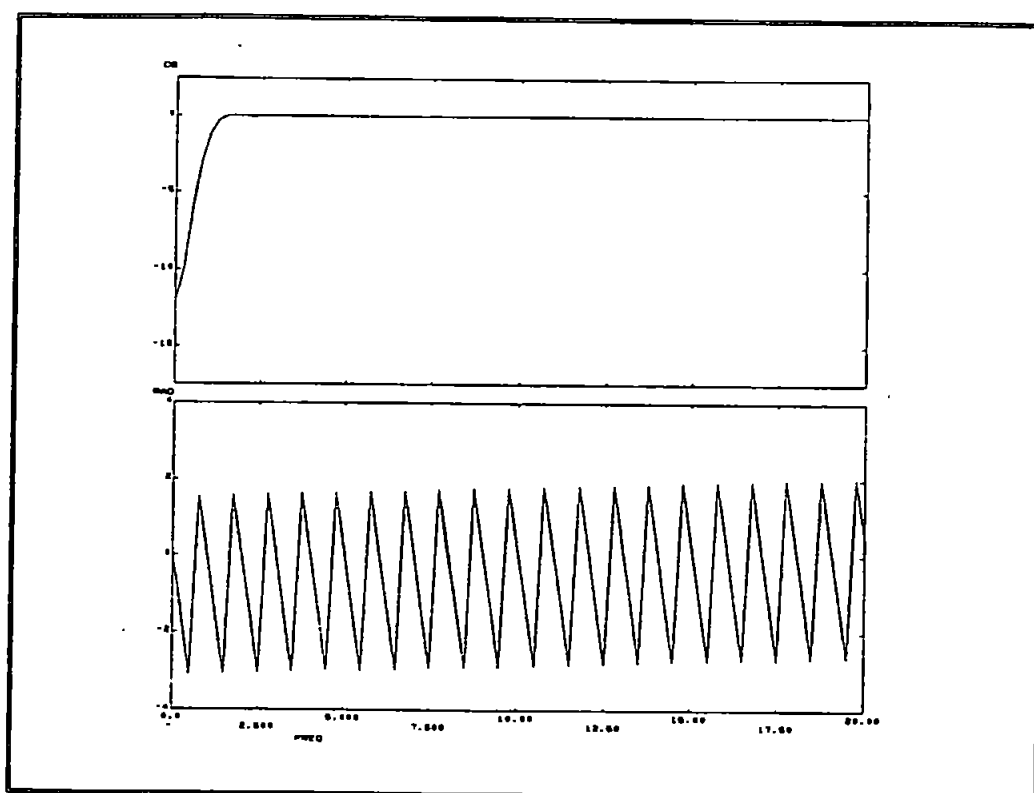
Figure 5.5 Magnitude and phase response of HPF.

5.3 SIGNAL FEATURE EXTRACTION.

Signal pre-processing provides the feature extraction functional block with 18 channels of 2-second signal segments. Feature extraction reduces this large quantity of numerical signal data into a concise symbolic list of features which closely represents those implicitly extracted by the EEG expert. Section 4.5 has shown the importance, when analysing the EEG, of the following signal characteristics:

. Frequency and amplitude

. Shape

. Phase relationships

. Previous recognised activity

These characteristics are obtained by extracting features from three domains, these being:

. Frequency domain.

. Time domain.

. Contextual domain.

Waveform frequency and magnitude is calculated from the frequency domain transformation of each signal. Waveform shape and phase relationships are calculated from the time domain signals using inter channel correlations. Previous recognised activity forms a context to signal analysis and contextual domain features include the comparison of spatio signal features and temporal signal features.

5.4.1 FREQUENCY DOMAIN.

Frequency domain features are extracted from the power spectral density (PSD) of each 2-second signal segment using Turbo C routines (Appendix K). Figure 5.6 illustrates the flow diagram for spectral feature extraction. PSD is calculated for each channel using the Blackman-Tukey or moving average (MA) method *[Childers, 1978]* (see also appendix H). From each PSD, spectral peaks are identified by comparison to a magnitude threshold which is related to the average power in the standard EEG frequency bands. This magnitude threshold is illustrated in figure 5.7.
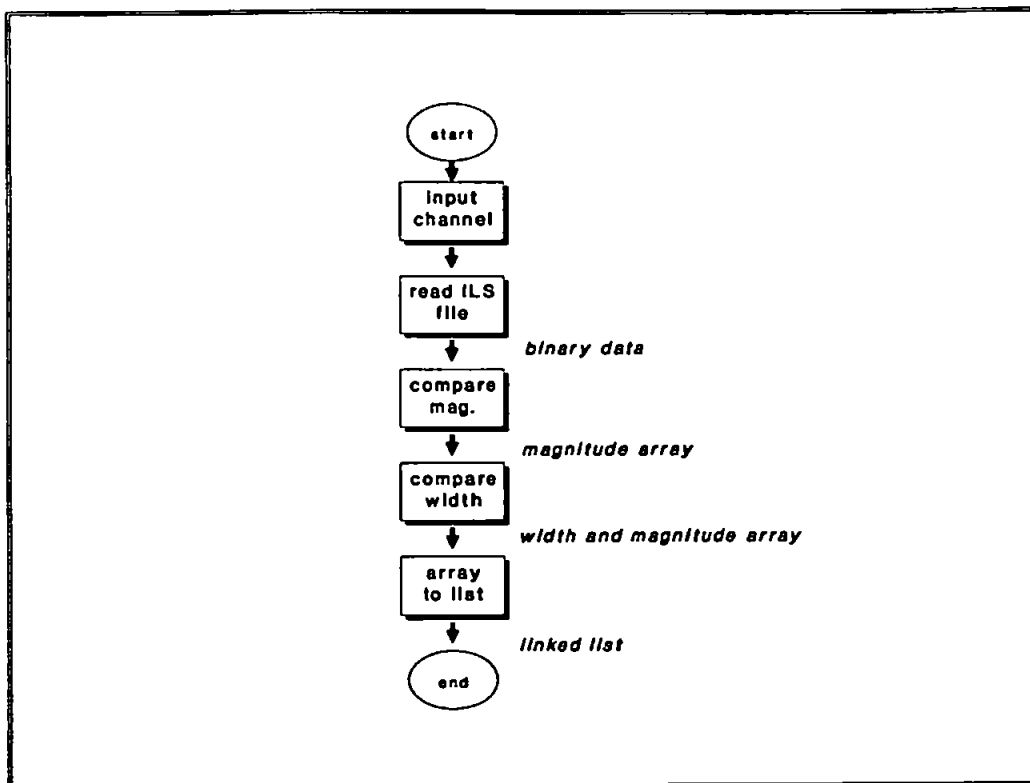
Figure 5.6 spectral feature extraction routine flow diagram.

The first three significant, and distinct, peaks in each channel are then used to compile a feature matrix that contains the magnitude, centre frequency and threshold width for each peak. Figure 5.8 illustrates the
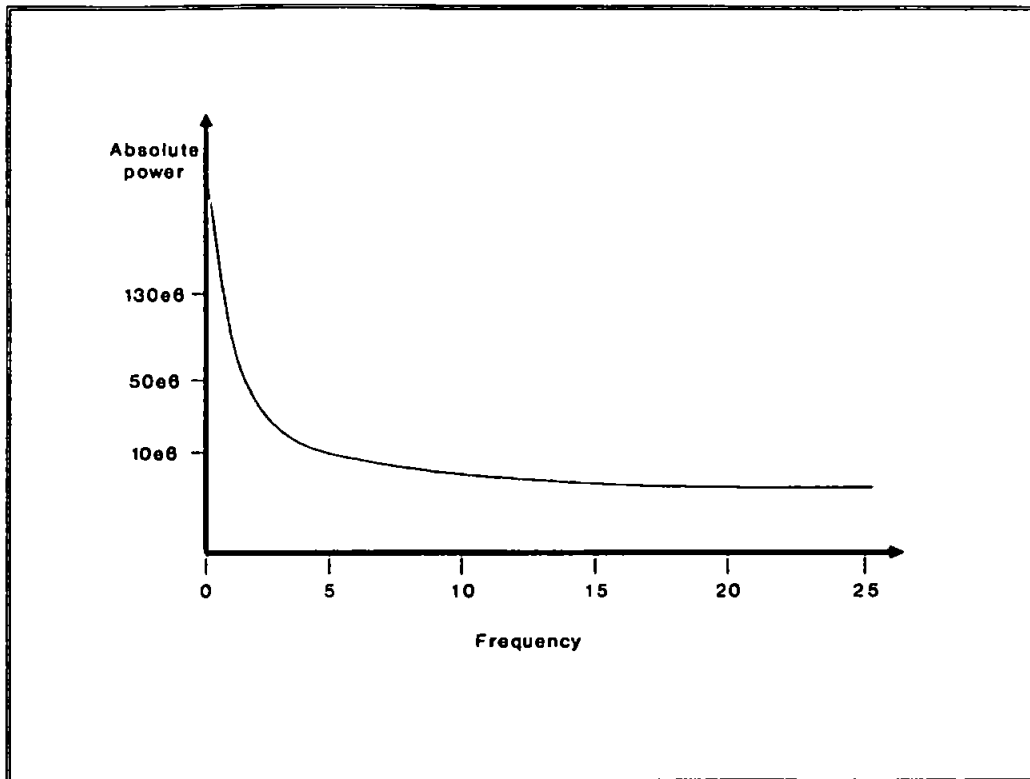
Figure 5.7 Spectral peak magnitude threshold.

process of spectral feature generation from a simplified signal segment.

The feature matrix is read by Turbo PROLOG routines which are used to convert the numerical features into symbolic tokens. The symbolic tokens are used to create individual spectral feature facts in a dynamic knowledge base that changes for each analysis segment. For example, in figure 5.8 the waveforms in channels 1 and 5 of figure 5.8(a) are transformed to the clear spectral peaks of frequency 2 Hertz in figure 5.8(b). The feature matrix of figure 5.8(c) shows that for channel 1, only one spectral peak greater than the threshold exists. This spectral peak has a centre frequency of 2.0 Hertz, a magnitude of 1.5e9, and a threshold width of 1.2 Hertz. Figure 5.8(d) shows the respective tokenised feature

(c) spectral peak feature matrix for channel 1.

["feature","fp2-f4","peak1","delta","med-mag","med-freq"][1.0]

(d) asserted feature fact for channel 1.

(a) 8 channel analysis segment time series   (b) 8 channel Power spectral density of (a).
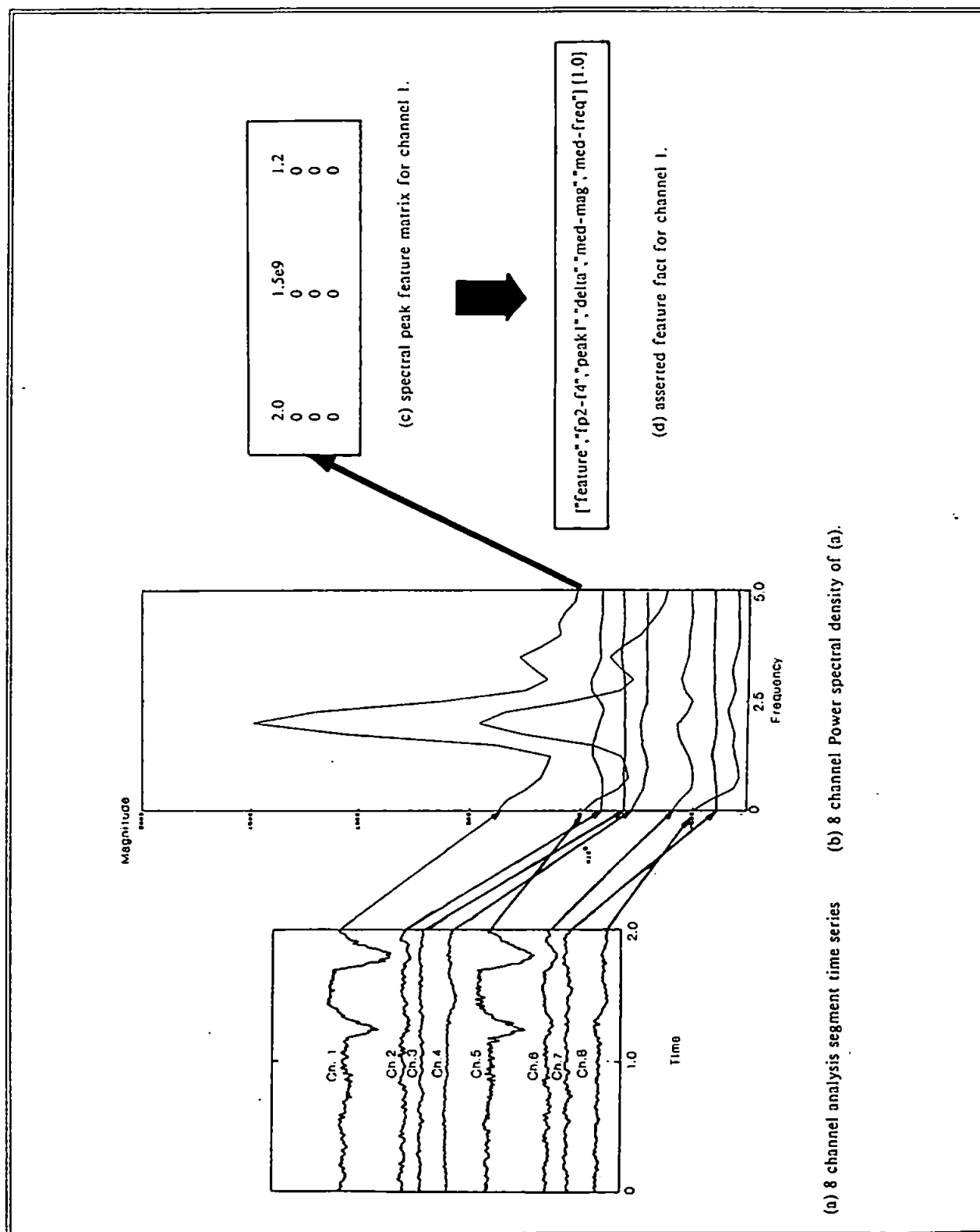
Figure 5.8 Spectral feature extraction and symbolic tokenisation.

page 192

fact which will be stored in the dynamic knowledge base. The feature fact states that a spectral peak exists in channel Fp2-F4 (channel 1), that it is the first spectral peak, and that it is attributable to the delta frequency band. The certainty of the fact is calculated as the membership of the delta frequency band, from the centre frequency of the spectral peak and the delta frequency band Fuzzy set (see section 4.5.2.1). This certainty is represented by the figure in the final parenthesis, i.e. 1.0. Incorporation of fuzzy set theory allows a spectral peak which has a frequency close to one of the conventional EEG frequency band boundaries to be represented as two feature facts in the dynamic knowledge base. For example, if the spectral peak of figure 5.8(c) had a centre frequency of 4.0 Hertz the respective feature facts would be:

["f-feature","fp2-f4","peak1","delta","med-mag","med-freq"] [0.5]

["f-feature","fp2-f4","peak1","theta","med-mag","med-freq"] [0.5]

where "theta" is the next adjacent EEG frequency band.

The last two elements in the first set of parenthesis for each spectral feature fact refer to the relative magnitude and frequency of the spectral peak. These values are used for spectral peak power distribution comparison over the scalp.

5.3.2 TIME DOMAIN.

Time domain features are extracted from the correlation calculated between various 2-second signal segments using Turbo C routines (appendix K). The cross-correlation between frontal EEG signal segments and posterior, temporal, and occipital signal segments is calculated together with the auto-correlation of frontal EEG and EOG signal segments. Figure 5.9 illustrates the flow diagram for the correlation feature extraction routine. Both cross and auto-correlation routines employ a similar feature extraction routine.

5.3.2.1 CROSS-CORRELATION.

The cross-correlation function (CCF) is a measure of the similarities or shared properties between two EEG/EOG signal segments. The CCF is used
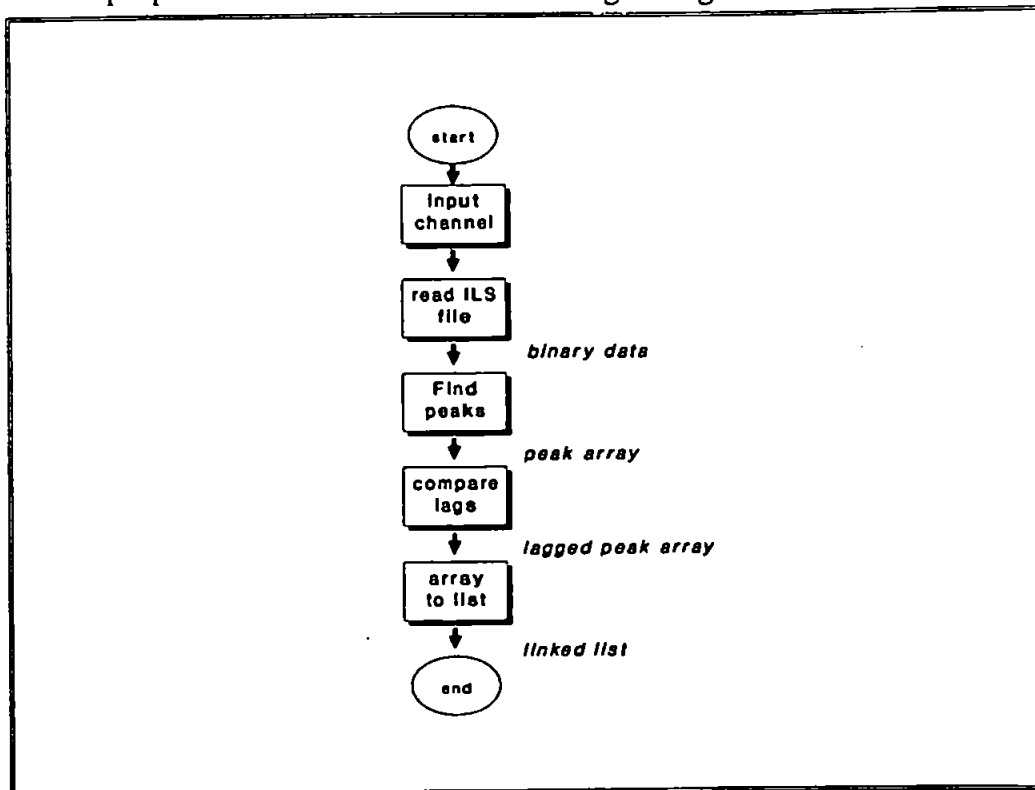
Figure 5.9 correlation feature extraction routine flow diagram.
page 194

to provide information on scalp potential distribution together with symmetry and phase differences between signals from different scalp regions. The cross-correlation function (CCF), for two signal segments x(k) and y(k), each with N samples, is calculated as:

$$CCF(n) = \frac{C_{xy}(n)}{[C_{xx}(0)C_{yy}(0)]^{1/2}} \qquad 5.7$$

where $C_{xy}(n)$ is the cross-covariance and defined as:

$$C_{xy}(n) = 1/N \sum_{k=0}^{N} x(k)y(k+n) \qquad n = 0,1,2, \dots \qquad 5.8$$

$$\text{and } C_{xx}(0) = 1/N \sum_{k=0}^{N} [x(k)]^2 , \; C_{yy}(0) = 1/N \sum_{k=0}^{N} [y(k)]^2$$

The cross-covariance is used to compile a feature matrix that contains the magnitude of the first peak, the time lag to the first peak, the number of peaks, and the time between peaks (see figure 5.8). The feature matrix is read by Turbo PROLOG routines (appendix K) which are used to convert the numerical features into symbolic tokens. The symbolic tokens are used to create individual covariance feature facts to append to the spectral feature facts in the dynamic knowledge base. Figure 5.10 illustrates the process of cross-covariance feature generation from a simplified signal segment. Figure 5.10 shows that the covariance of channels 1 and 5 in figure 5.10(a) is transformed to a clear peak at t=0 in the cross-covariance of figure 5.10(b). The feature matrix of figure 5.10(c) shows that only one peak exists in the cross-covariance of channels 1 and 5, peak has a magnitude of 500e5, and with a time lag of

(a) 8 channel analysis segment time series

(b) single cross-covariance of (a) channels 1:5

(c) cross-covariance feature matrix for channel 1:5

(d) asserted feature fact for channel 1:5

'1.feature':'Fp2-F4:Fp1-F3':'high-corr':'zero-lag':'zero-priodicity':'zero-freq'
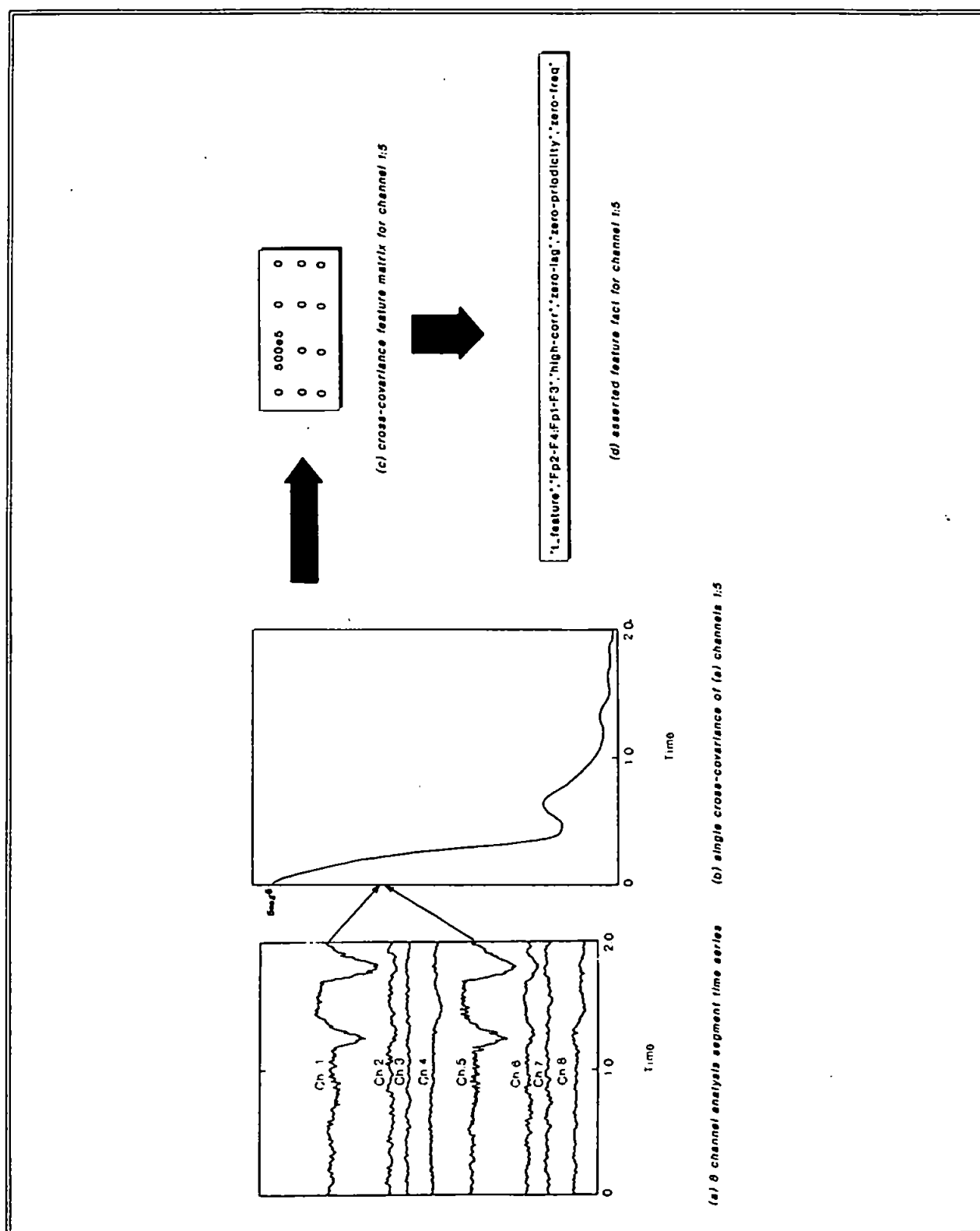
Figure 5.10 Cross-Covariance feature extraction

and symbolic tokenisation.

zero. No other peaks exits and therefore the last two cells are empty. Figure 5.10(d) shows the respective tokenised feature fact which will be stored in the dynamic knowledge base. The feature fact states that a time domain feature exists for channels Fp2-F4:Fp1-F3, and that there is a high similarity between these channels. This indicates a high degree of symmetry between the left and right frontal EEG channels. Unlike the spectral feature facts no uncertainty is incorporated into the covariance feature facts.

5.3.2.2 AUTO-CORRELATION.

The auto-correlation function (ACF) is a special form of the CCF involving only one signal. The ACF is used to provide an estimation of waveform shape by displaying the periodicity of the signal. A periodic signal will result in a periodic auto-correlation function of the same frequency as the original waveform but with attenuated high frequency noise and enhanced periodic signal. Any periodicity in a signal segment is indicative of an abnormal waveform because of the more likely random occurrences of OA.

The auto-correlation function (ACF), for a signal segment x(k), with N samples, is calculated as:

$$ACF(n) = \frac{C_{xx}(n)}{C_{xx}(0)} \qquad 5.9$$

where $C_{xx}(n)$ is the auto-covariance and defined as:

$$C_{xx}(n) \;=\; 1/N \sum_{k=0}^{N} x(k)x(k+n) \qquad n = 0,1,2, \dots \qquad\qquad 5.8$$

The auto-covariance is used to generate covariance feature facts in exactly the same manner as described above for the cross-covariance. For example figure 5.11 illustrates the process of auto-covariance feature generation from a simplified signal segment. The feature fact of figure 5.11(d) states that a time domain feature exists for channel Fp2-F4, and that the signal is highly periodic. This indicates the possible presence of abnormal slow waveforms

(a) 8 channel analysis segment time series

(b) 4-channel auto-covariance of (a) channels 1,5,9,13

(c) auto-covariance feature matrix for channel 1

(d) asserted feature fact for channel 1
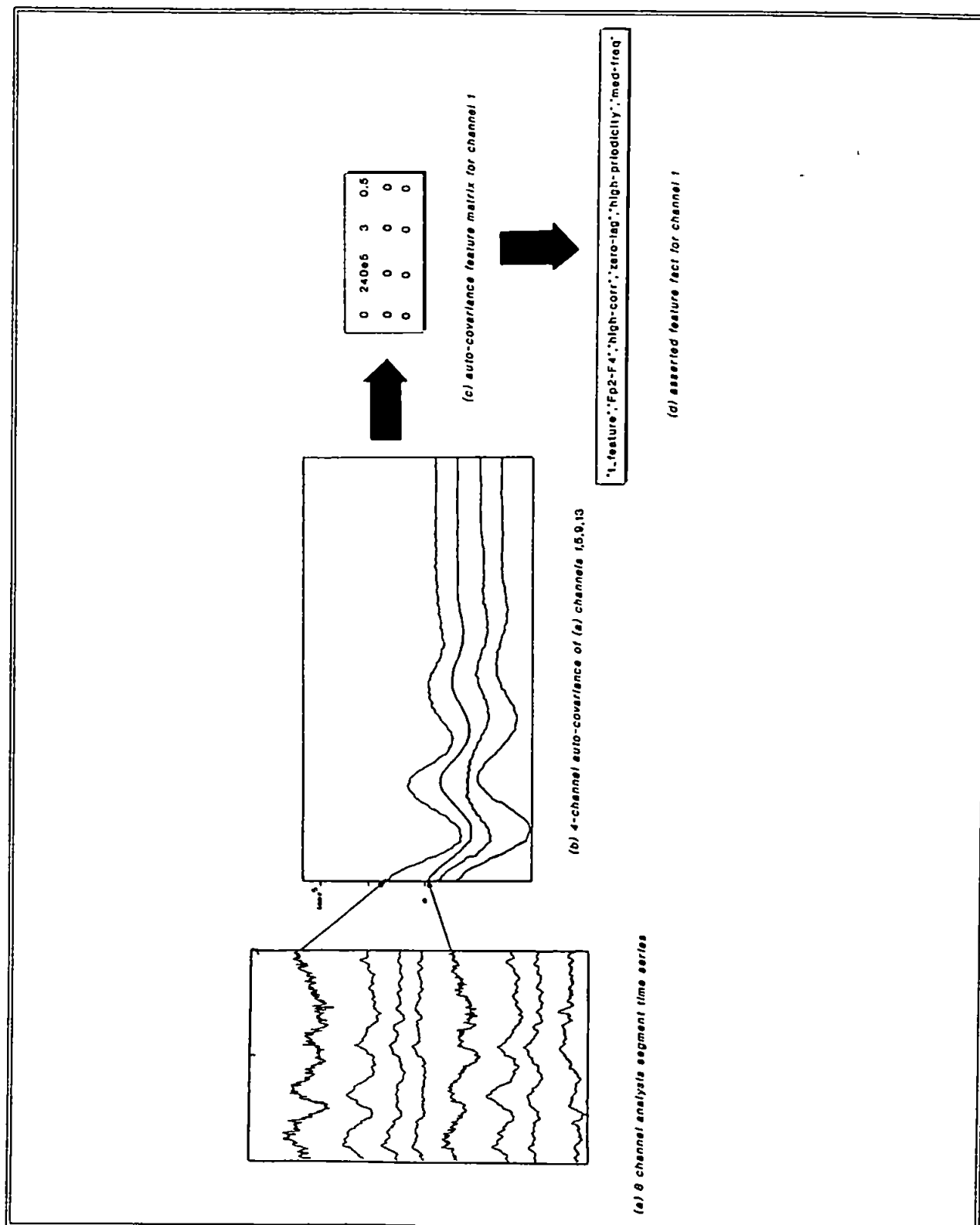
Figure 5.11 Auto-Covariance feature extraction

and symbolic tokenisation.

5.3.3 CONTEXTUAL DOMAIN.

A considerable amount of skill used by the EEG expert in the analysis of the EEG lies in the ability to view the EEG in context of previous and future activity, and knowledge of patient details. This allows the expert to 'ignore' insignificant details and to concentrate on particular areas. This contextual domain can be viewed as existing on three levels:

. Current context

. Local context

. Historical context.

Figure 5.11. illustrates the relationship of these contexts to a multi-channel section of EEG/EOG.
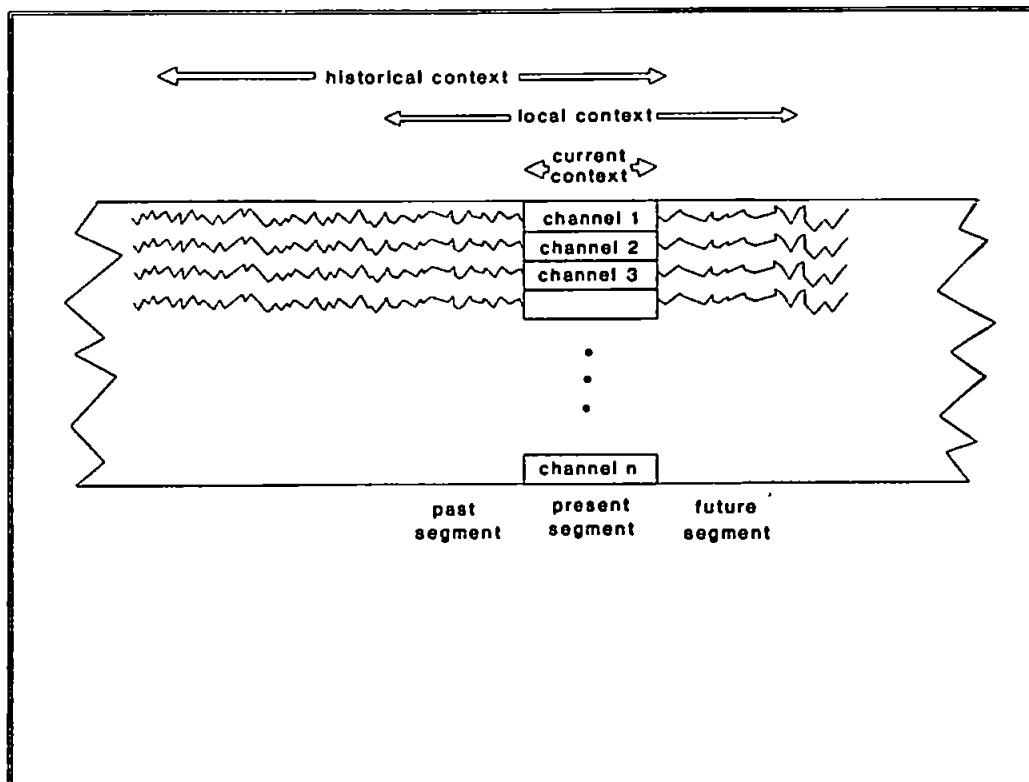
Figure 5.11 Conceptual representation of contextual domain features.
page 200

Current contextual features include spectral features and inter-channel correlations, obtained above from frequency and time domains respectively. Local contextual features include the relationship between the features of immediate past and future signal segments, to the current signal segment. For example, if a previous signal segment has been identified as containing abnormal slow waveforms, then the current signal segment is more likely to contain abnormal slow waveforms than it would have been should the previous segment not have contained any abnormal slow waveforms. Historical contextual features again will influence the analysis of the current signal segment, but the accumulated historical signal features will now be used. For example, an EEG that has contained abnormal slow waveforms some time previously, will make the possibility of the current signal segment containing abnormal slow waveforms that much greater. However, this possibility will decrease as time passes with no abnormal slow waveforms being detected.

Contextual domain features are generated during feature extraction and are influenced by acceptability of a final segment classification. Current contextual features are generated from frequency and time domain feature extraction. Local and Historical contextual domain features will be generated as a result of the user's response to segment classification.

If the previous segment has been classified to the user's satisfaction a local context is asserted into the dynamic knowledge base which consists of previous segment classification and final classification measure of belief (4.5.2.3), over-writting any previous local context. For example if the previous segment was correctly classified as 'abnormal slow waves

only' with a measure of belief of 0.8, then the following feature fact would be asserted into the dynamic knowledge base:

["c-feature","local","abnormal only"] [0.8]

In addition to the local contextual feature fact, the historical context is amended upon detection of an abnormality. Two contextual feature facts are used for this purpose:

["c-feature","total classifications"] [oldX]

["c-feature","historical","abnormality"] [oldY,oldZ]

where oldX is the total number of correctly classified signal segments.

and oldY is the total number of correctly classified abnormal segments.

and oldZ is the historical measure of belief in abnormality.

the historical feature facts will be updated after the above correctly classified segment as:

$$newX = oldX + 1$$

$$newY = oldY + 1$$

$$newZ = oldZ + [(1\text{-}oldZ)\ newY/newX\ MB]$$

where MB is the measure of belief of the correctly classified segment. The new historical contextual feature facts will be:

["c-feature","total classifications"] [newX]

["c-feature","historical","abnormality"] [newY,newZ]

Should the previous segment classification have been incorrect, then the historical feature facts are updated as:

newX = oldX

newY = oldY

newZ = oldZ - [oldZ (oldY-oldX)/oldY MB]

The measure of belief in the historical context is therefore either increased or decreased asymptotically in proportion to the measure of belief of classification and the proportion of abnormal segments to normal segments.

5.4 REASONING.

Feature extraction has provided a list of symbolic tokenised features which attempts to quantify those key aspects of the EEG that the expert uses in identifying OA, and in differentiating OA from abnormal slow waveforms. This section details the implementation of the reasoning system which operates on the stored rules and extracted features in order to select appropriate OA removal. The reasoning system is divided into two parts:

. The knowledge base.

. The inference engine.

5.4.1 KNOWLEDGE BASE.

The knowledge base is a self contained, separate, ASCII file divided into two sections:

. Dynamic knowledge.

. Static knowledge.

The dynamic, or 'short term', knowledge base contains the extracted segment features from the current analysis segment, and the contextual features obtained from other segments. The static, or 'long term', knowledge base contains the elicited rules from the expert, and static signal information including, channel numbers and their location, relationships between channels, and tokenisation values for feature extraction. Separation of the knowledge from the inference engine allows easy modification of the knowledge, and alternative knowledge bases to be used with the same inference engine.

5.4.1.1 DYNAMIC KNOWLEDGE.

The dynamic knowledge base is not initially part of the stored knowledge. Feature extraction generates the tokenised features and these are temporarily stored together with the static knowledge. The dynamic knowledge is updated for each segment being analysed and it is only the contextual features that maintain any information on previously analysed segments. Figure 5.12 illustrates the typical contents of the dynamic knowledge base.

[f-feature,fp2-f8,peak1,delta,med-mag,med-freq],[1.0]

[f-feature,fp2-f8,peak2,delta,low-mag,high-freq][1.0]

[f-feature,fp1-f7,peak1,delta,med-mag,med-freq][1.0]

[f-feature,fp1-f7,peak2,delta,low-mag,high-freq][1.0]

[f-feature,fp2-f4,peak1,delta,med-mag,med-freq][1.0]

[f-feature,fp2-f4,peak2,delta,low-mag,high-freq][1.0]

[f-feature,fp1-f3,peak1,delta,med-mag,med-freq][1.0]

[f-feature,fp1-f3,peak2,delta,low-mag,high-freq][1.0]

[t-feature,fp2-f8,high-corr,zero-lag,no-periodicity,zero-freq]

[t-feature,fp1-f7,high-corr,zero-lag,no-periodicity,zero-freq]

[t-feature,fp2-f8:fp1-f7,high-corr,zero-lag,low-periodicity,low-freq]

[t-feature,fp2-f8:f8-t4,med-corr,zero-lag,low-periodicity,low-freq]

[t-feature,fp1-f7:f7-t3,med-corr,zero-lag,low-periodicity,low-freq]

[c-feature,totals][4]

[c-feature,local,blink][0.8]

Figure 5.12 Typical dynamic knowledge base contents.

5.4.1.2 STATIC KNOWLEDGE.

Extensive knowledge elicitation and data analysis yielded a set of nearly 60 rules and 80 conditions for both segment identification and OA classification. These rules constitute the search space in which the inference engine will operate. Figure 5.13 illustrates a simplified search space for the implemented system, and each node in the search space represents one or more alternative rules.

segment

cerebral only   artefactual only

perspiration   movement   eye movement   other

vertical horizontal blink   complex
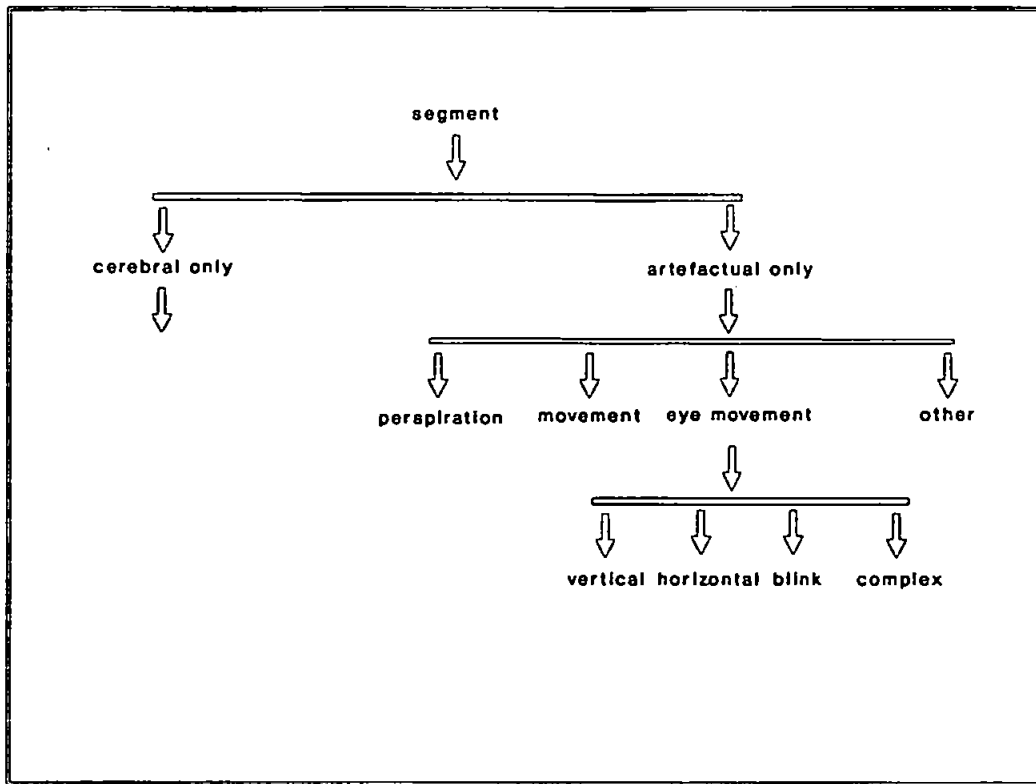
Figure 5.13 search space for the classification of EEG

segments: representation as a decision tree.

Appendix M lists all the elicited rules, in the knowledge base format, and figure 5.14 illustrates a sample of these rules showing the respective rule confidence (see section 4.5.2.2) in parenthesis after the rule. The rule confidence was calculated as the average of the heuristic and statistical values elicited in section 4.5.2.2.

Section 4.5 identified the key elements, used by the expert, in analysing the EEG, and in differentiating between OA and abnormal waveforms. In summary these were, the spatio-temporal, and spatio-spectral signal qualities. Particular attention was paid to the symmetry between

Rule 1:
segment could contain abnormal slow waveforms (0.5)
if   abnormal slow waveforms have been observed previously.

Rule 3:
segment is more likely to contain abnormal slow waveforms (0.6)
if   the last segment contained abnormal slow waveforms.

Rule 12:
segment contains no slow waves (1.0)
if   no significant spectral activity exists in the delta band.

rule 15:
segment contains artefact only (0.93)
if   slow activity is maximum in the frontal channels
and   pre-frontal channels are symmetrical
and   EOG channels are symmetrical
and   slow waves only appear in frontal channels

rule 23:
segment contains artefact only (0.4)
if   slow activity is maximum in the frontal channels
and   slow waves only appear in anterior channels
and   no slow waves exist in the EEG that are not present in the EOG

rule 32:
segment contains both artefact and abnormal waves (0.8)
if   slow activity is not maximum in the frontal channels
and   pre-frontal channels are symmetrical
and   slow waves exist in the EEG that are not present in the EOG

rule 46:
artefact only contains OA only (0.8)
if   EOG channels are symmetrical
and   slow waves are attributable to more than one electrode
and slow activity is maximum in the EOG channels

rule 53:
OA only contains blink artefact (0.9)
if   slow waves are larger in pre-frontal channels than in temporal
and   the slow waves are of short duration
and the slow waves are in phase in temporal channels
and the slow waves are phase reversed around the eyes

Figure 5.14 Sample of rules used in the IOARS.

cerebral hemispheres of any suspect wave, as well as signal phase relationships.

The key features utilised by the rules in categorising each analysis segment are:

. The context of previous segment content

. The position of maximum delta band spectral magnitude

. The similarity of spectral content in left and right EOG signals

. The cerebral distribution of spectral peaks in the delta band ·

. The correlation between combinations of EOG and EEG signals

. The periodicity of waveforms in frontal channels

These features are obtained from the dynamic knowledge acquired from each segment, such as that of figure 5.12. The rules of figure 5.14 form a small subset of the total rules shown in appendix M, but are used here to illustrate the classification of the waveforms shown in figure 5.8. It is assumed that the dynamic knowledge acquired from the waveforms of figure 5.8(a) are that of figure 5.12.

Rules are investigated sequentially in the knowledge base. Therefore there is a natural ordering to rules which result in the same consequent. Rules which have larger numbers of antecedent clauses are generally harder to satisfy; as a result the confidence in the rule, once satisfied, is larger. Consecutive rules have increasingly relaxed conditions, and hence reducing rule confidence, until no other rules can be found and that branch of investigation fails.

Figure 5.12 shows no abnormal slow waveforms in either historical or local context and as a result rules 1 and 3 are by-passed by the inference engine (see later). Rule 12 is used to ignore segments with insignificant spectral activity in the OA frequency band. This is a valid process because of the known OA frequency band and the limitation of this implementation to only remove OA.

Having focused the problem to one of identifying the slow wave activity detected in the segment, rules 15 onwards are utilised. Ocular artefact potential appears strongest in the anterior regions of the scalp, around the eyes, and is also symmetrical. Rules 15 and 23 identify a segment as containing only artefactual waveforms if the detected slow wave activity is maximum in the frontal channels. Rule 15 is investigated first and will succeed if all its conditions are satisfied with some degree of certainty (see section 4.5.2.1). Rule 23 is only investigated should rule 15 fail. For example rule 15 investigates the symmetry of the frontal EEG channels, and the EOG channels and is therefore given a higher confidence than rule 23 which does not have these conditions. The waveforms of figure 5.8(a) are symmetrical and only appear in the frontal channels, and therefore will satisfy the conditions of rule 15. The current measure of belief (MB_c) is therefore calculated as:

$$MB_c = 0.93.[min(1.0, 1.0, 1.0, 1.0)] \qquad \text{(see section 4.5.2.3)}$$
$$= 0.93$$

Segments identified as containing only artefact are further classified as containing only OA by examination of the EOG channels. For OA to be

present, the slow waves must have maximum power in the EOG channels, be present in more than one channel, and have a symmetrical spectral content in the EOG channels. Rule 46 investigates these conditions and for the waveforms of figure 5.8 will succeed. MB_c now becomes:

$$MB_c = 0.93 + \{(1 - 0.93)\ 0.9\ [\min(1.0, 1.0, 0.5)]\}$$
$$= 0.962$$

For OA identification the phase relationship between signals close to the eyes is also of importance. Vertical eye movement (VEM) and blink produces potentials that are in phase on both sides of the scalp, while horizontal eye movement (HEM) will produce convergent potentials on one side of the scalp and divergent potentials on the other. The time course of OAs are also different, blinks produce large, short duration, negative potentials above the eye, and small positive potentials below the eye. VEM and HEM produce longer duration potentials. Rule 53 investigates these time domain features by examination of the t-features in the dynamic knowledge base. The waveforms of figure 5.8 are finally identified as being attributable to blink-type OA with a measure of belief (MB_c):

$$MB_c = 0.962 + \{(1 - 0.962)\ 0.9\ [\min(1.0, 1.0, 1.0, 1.0)]\}$$
$$= 0.9962$$

This is sufficient measure of belief to allow the OA removal algorithm to be applied with coefficients preset to values most suitable to blink-type OA.

Segments are categorised into one of the four types detailed in section 4.5.1.2. Segments identified as containing OA only, with a sufficient measure of belief, are subjected to OA removal, and these segments are clearly identified. Adaptive filter coefficients can be retained at the end of each segment so that further segments of the same type of OA can utilise the stored coefficients.

The primary aim of the intelligent OA removal system is to apply the OA removal algorithm only where appropriate. This means that segments containing no significant OA, and those that contain both suspected abnormal waves and OA are passed unaltered. Rule 32 identifies a segment as containing both abnormal waves and artefactual waves when the maximum delta activity occurs away from the EOG channels but the EOG channels still contain delta waves and are symmetrical. These rules are attempted when the inference engine has failed to successfully identify a segment as containing artefact only. The conditions of these rules are often the inverse of conditions for OA only and there is no need to re-evaluate these because the inverse of a condition found to be false with certainty 0.8, is simply true with a certainty of 0.8. Segments where suspected OA and abnormal waves exist are clearly identified but are not altered.

Rules 12 and 23 of figure 5.14 are illustrated in the knowledge base format in figure 5.15. This illustrates the five main component parts of the static knowledge base, which are:

textual rules:

    text-rule(12,1,"segment","no significant delta waves",[4])
    text-rule(23,0.4,"segment","artefact only",[5,13,16])

textual conditions:

    text-cond(3,"any significant spectral activity exists in the delta band)
    text-cond(4,"not any significant spectral activity exists in the delta band)
    text-cond(5,"slow activity is maximum in the frontal channels")
    text-cond(6,"not slow activity is maximum in the frontal channels")
    text-cond(13,"slow waves only appear in anterior channels")
    text-cond(14,"not slow waves only appear in anterior channels")
    text-cond(15,"slow waves exist in the EEG that are not present in the EOG")
    text-cond(16,"not slow waves exist in the EEG that are not present in the EOG")

numerical conditions:

    num-cond(3,["f-feature","","","delta","",""])
    num-cond(5,["max-front"])
    num-cond(14,["posterior-features"])
    num-cond(15,["non-eog"])

feature demons:

    cl(["non-eog"],[],[
                ["intpred","channel","X","eeg"],
                ["f-feature","X","","delta","","A"],
                ["intpred","notpeak","peak","A"],
                ["anyeog","A"],
                ["intpred","notpeak","peak","A"]])

    cl(["any-eog","A"],[],[
                ["intpred","channel","Y","EOG"],
                ["f-feature","Y","","delta","","A"],
                ["intpred","assertpeak","peak","A"],
                ["intpred","cutback","",""]])

    cl(["any-eog","A"],[],[
                ["intpred","true","",""]])

contextual facts:

    anterior("fp2-f8")

    channel(0,"fp2-f8","EOG")

    freqbounds("delta",6,18).

Figure 5.15 The rules of figure 5.14 transcribed into knowledge base format.

. Textual rules.

. Textual conditions.

. Numerical conditions.

. Feature demons.

. Contextual facts.

In order to provide the explanation facilities crucial to this system, rules and conditions are stored in a textual format. This is augmented by the necessary numerical rules which operate on the features obtained from the data. Some numerical conditions can be implemented by searching the dynamic knowledge base for one simple asserted feature fact, for example condition 3, which searches for a delta frequency spectral peak. However, some conditions require more sophisticated search, in which case the numerical condition is implemented using a rule contained in the knowledge base and referred to as a feature demon. For example, examination of figure 5.15 shows the equivalence of textual and numerical condition 15 which tests for the existence of delta spectral peaks present in the EEG but not in the EOG. Condition 15 is the inverse of condition 16. Condition 15 is implemented by the use of the feature demon 'non-eog' (see inference engine later), which is a rule which searches for combinations of spectral peaks. Condition 15 will be found to be false if delta frequency waveforms are found in the EEG that are uncorrelated with those in the EOG, causing condition 16 to be found true with the appropriate certainty.

5.4.2 INFERENCE ENGINE.

The inference engine is built in PROLOG following the theory introduced in chapter 2. The basic inference engine is based on a simple expert system shell - GENI *[Borland international Inc.]*. This basic shell is enhanced by accommodating uncertainty, complex knowledge base structure, and hybrid search techniques. Figures 5.16 represents the intelligent OA removal system as a decision tree. This equates to a program flow chart for a procedural language. Figures 5.17-19 illustrate the simplified decision tree structure for the inference engine (INFER). Appendix K contains the full program listings for the complete system.

With reference to figure 5.16, it can be seen that the IOARS is essentially procedural in the initial stages. This involves loading the knowledge base and data, and displaying the data for user selection (see section 5.6). However, upon selection of the data further data processing is dependent on the outcome of the data classification, from the inference engine - infer. A path is taken which will result in either OA removal, OA indication or preservation of original data. Figure 5.16 also indicates that this process is a repetitive cycle of data selection,inference, and action.

Figure 5.17 illustrates the construction of the inference engine. This shows that the inference process is continually looking for rules to satisfy and if no rules are found then the search is assumed completed and a classification is made. However, a rule will be satisfied by satisfying all attached conditions and this process is a repetitive cycle until no
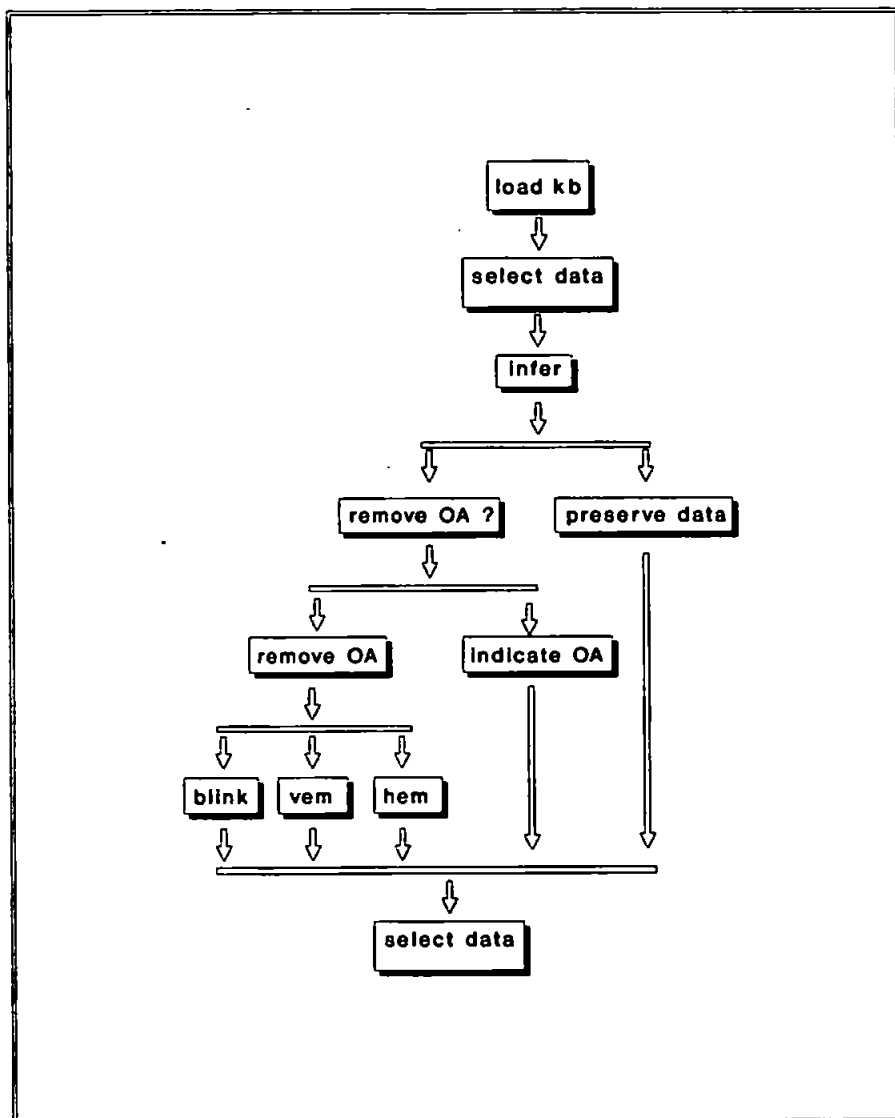
Figure 5.16 IOARS decision tree.
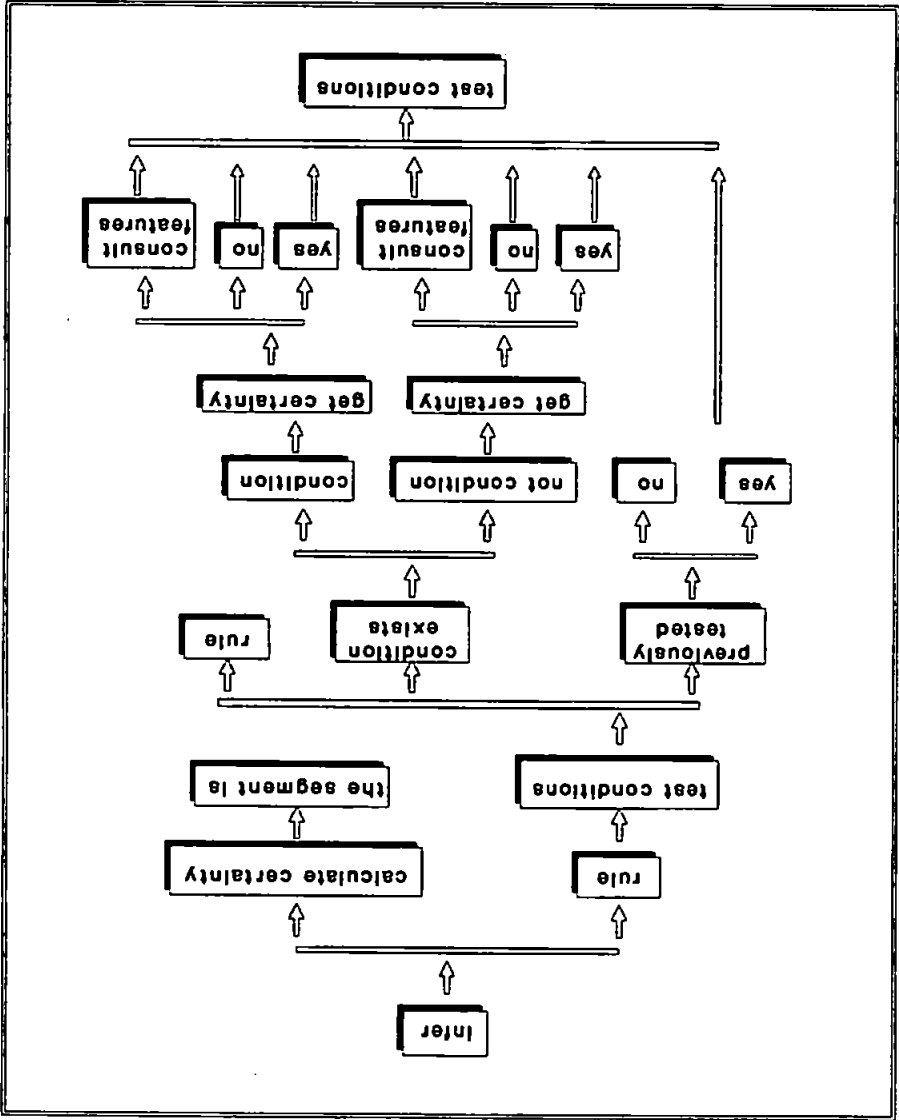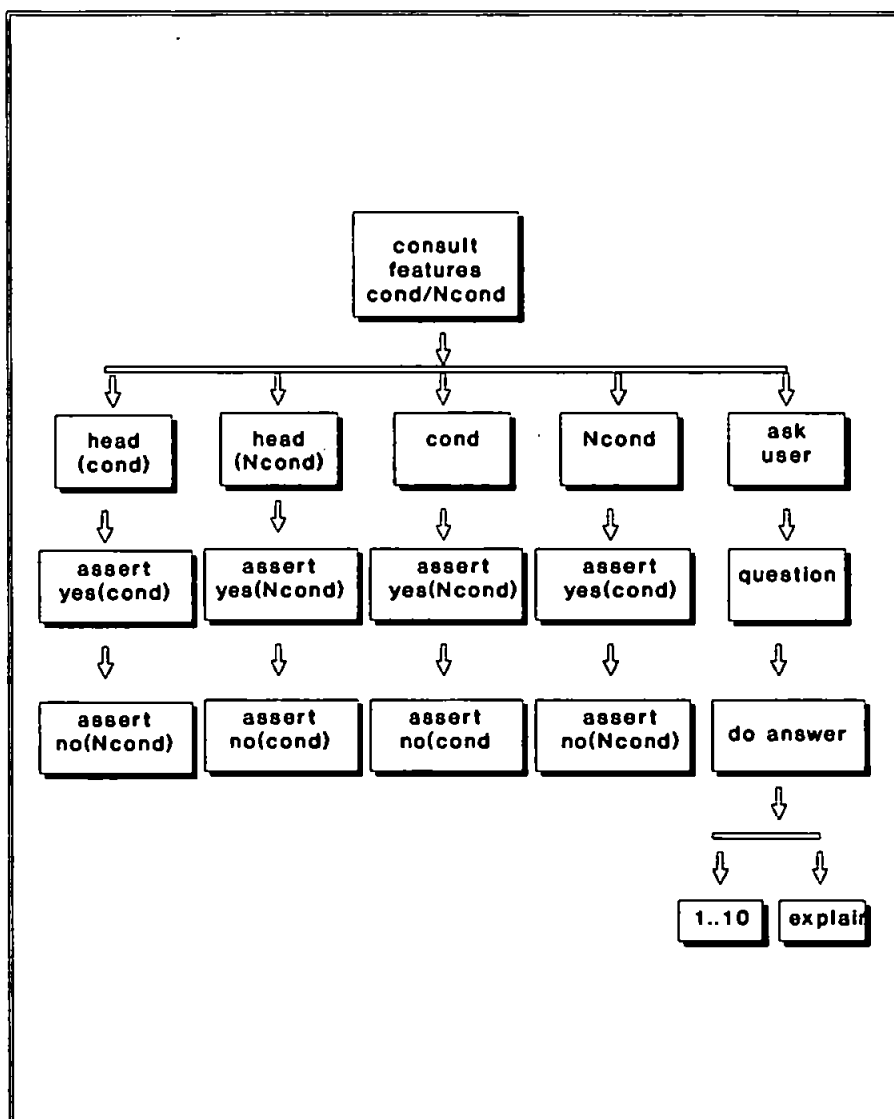
Figure 5.18 Feature consultation decision tree.
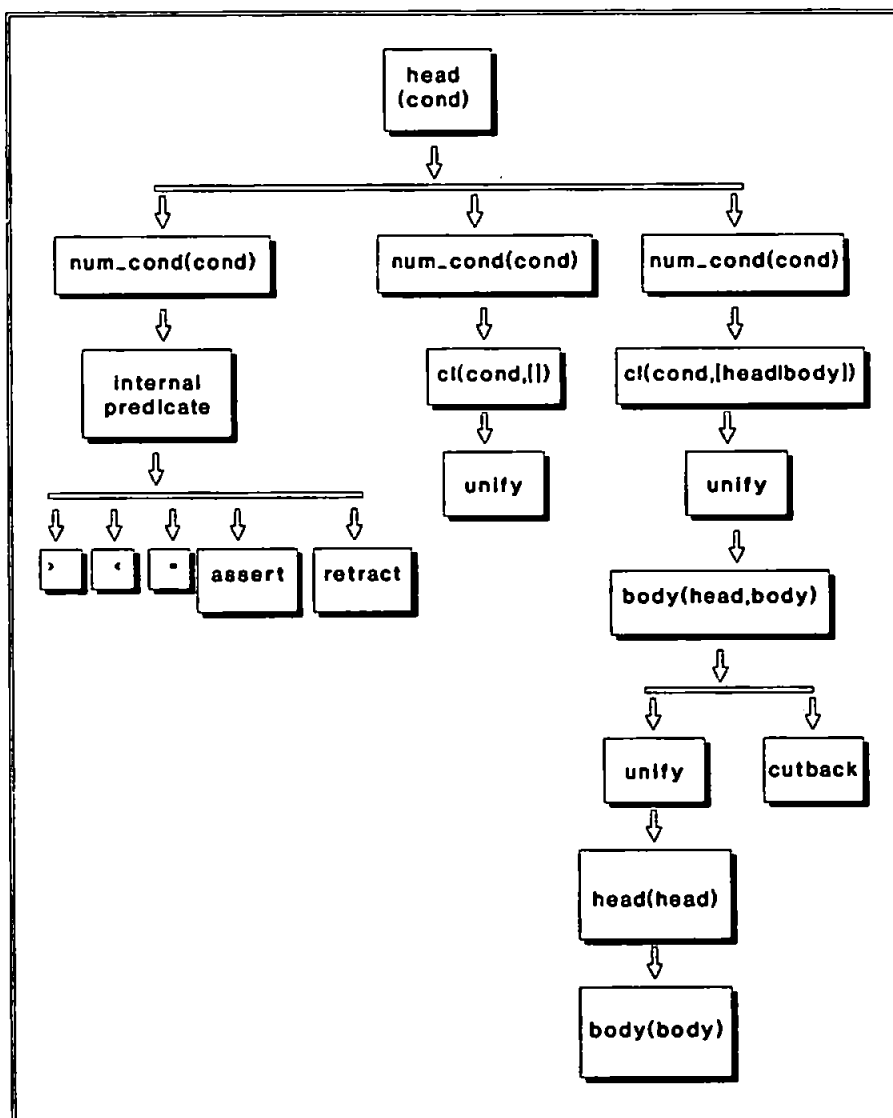
Figure 5.19 Dynamic knowledge base interrogation decision tree.

further conditions exist, in which case a new rule is searched for. A condition can be satisfied as being true or untrue in two ways. Firstly, if the condition has been previously proved true, or untrue, and secondly by searching for relevant facts in the dynamic knowledge base. If neither the condition or its inverse has been previously investigated, then the truth of the condition is determined by feature consultation - consult features.

Figure 5.18 illustrates the construction of the feature consultation clause for which five possibilities exist for satisfaction. Success of the condition or not condition will satisfy the first two, and result in assertion of the appropriate facts in the dynamic knowledge base. Failure of the condition will result in the opposite facts being stored in the dynamic knowledge base. Lastly, failure to find an appropriate condition will result in the user being asked to supply the relevant information. The dynamic knowledge base is interrogated by the clause - head.

Figure 5.19 illustrates the construction of the feature interrogation clause - head. Head provides several key extensions to the simple PROLOG inference engine by using 'Meta' programming techniques *[Borland international Inc. 1989, PROLOG reference; Heyshim, 1988].* Meta programming techniques enable the construction of control structures other than the simple backward chaining depth first of conventional PROLOG. These control structures can however, be written using PROLOG and act effectively as an interpreter within PROLOG. An interpreter was constructed for the IOARS to allow the following extensions to conventional PROLOG to be made:

. To allow rules to be contained in the knowledge base.

. To allow variables to be manipulated in the knowledge base

Reference to figure 5.15 shows the use of these two extensions in the feature demon 'non-eog'. Rules are contained in the knowledge base as a list of clauses where each clause can represent a feature fact or another feature demon. Variables are represented as capital letters. The interpreter is responsible for breaking each rule up and investigating each of the integral feature facts.

A condition can take one of three forms for the interpreter to deal with:

. Internal predicate

. Fact

. Rule

An internal predicate is used as a function of two variables, such as '>, <, =, etc., it is also used to assert and retract facts from the dynamic knowledge base. An internal predicate is always an integral condition of a rule. A fact is simply a feature fact extracted from the data, i.e. it is a rule with no conditions, it may however pass variables and must be unified after satisfaction. A rule may contain any number of feature facts, internal predicates and other rules. Each condition of the rule is interpreted by the clause 'body' which itself uses 'head' to satisfy individual conditions.

5.5 ADAPTIVE FILTERS.

Classification of a segment as containing OA only will allow the application of the OA removal algorithm. The OA removal algorithm is an adaptive filter and utilises the RLS algorithm, as described in section 4.4.2, for parameter updating. The algorithm has starting parameters which are dependent upon the classified OA type (see section 4.4.4). These 'pre-loading' parameters for the OA removal algorithm are stored as a number of matrices in the static knowledge base and represent the long term values obtained from statistical data (see section 4.3.3.1). During OA removal these values are transferred to the dynamic knowledge base where their value is free to be updated with the data. Each matrix represents a particular OA and will contain the pre-loading parameters for P and K for each relevant electrode position. A simplified matrix for blink type OA at a number of frontal electrode positions is illustrated in figure 5.20.

BLINK PRE-LOADING PARAMETER MATRIX

parameter channel	k_1	k_2	D_0	D_1	U_0
Fp2-F4	0.3	0.1	4.0E-7	4.0E-7	-1.0
Fp1-F3	0.1	0.3	4.0E-7	4.0E-7	-1.0
Fp2-F8	1.0	0.0	4.0E-7	4.0E-7	-1.0
Fp1-F7	0.0	1.0	4.0E-7	4.0E-7	-1.0

NB matrix assumes a 2 parameter filter model using the EOG channels detailed in section 4.3, and the UD factorisation algorithm.
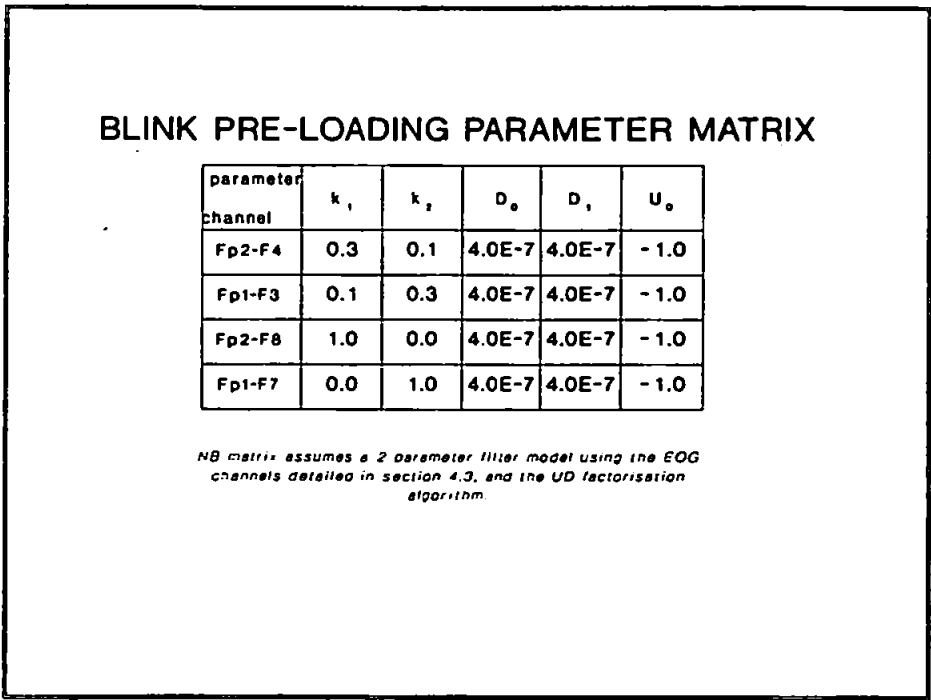
Figure 5.20 OA removal algorithm pre-loading matrix.
page 221

The matrix of figure 5.20 is stored as a number of facts in the knowledge base, where a fact exists for each combination of OA type and electrode position. For example, the matrix of figure 5.20 is stored in the knowledge base as the following facts:

oa-parameter(blink,fp2-f4,0.3,0.1,4.0e-7,-1.0,4.0e-7)

oa-parameter(blink,fp1-f3,0.1,0.3,4.0e-7,-1.0,4.0e-7)

oa-parameter(blink,fp2-f8,1.0,0.0,4.0e-7,-1.0,4.0e-7)

oa-parameter(blink,fp1-f7,0.0,1.0,4.0e-7,-1.0,4.0e-7)

Appendix G contains the software listing for the OA removal algorithm which is essentially the same as that shown in the flow chart of figure 4.32. The algorithm is implemented in 'C' and utilises the segment classification, and the oa-parameter facts contained in the knowledge base to effect OA removal.

5.6 USER INTERFACE.

The user interface, if not the most important, is surely the most prominent, and by definition visual, functional block of the system. A primary purpose of this investigation was to provide the clinician with an adaptive signal processing tool capable of removing OA from the EEG; the emphasis being on the word tool. Such a tool must be:-

. Reliable.

. Easily used.

. Convey, to the user, all necessary information in a form that is
easily understood.

These criteria have been addressed and partly met in this implementation by utilising the graphics display capabilities of the PC. Analysis of the above criteria allowed the following specifications to be made for an appropriate user interface:-

. To display real time EEG/EOG time domain signals.

. To allow, upon request, identified OA to be indicated.

. To provide adequate explanation of OA identification.

. To allow, upon identification and under user control, removal of OA.

Real time display of EEG and EOG signals is achieved elegantly using scrolling high resolution graphics display. This can be an extremely time consuming procedure using standard PC hardware and this area is one in which the use of fast Assembly language routines was unavoidable. The present system is capable of 8 channel display on a 640x480 graphics terminal, directly upgradable to 16 channel display on a 1024x768 display (See plate 5.1).

Indication of OA is achieved on a segment by segment basis, where a segment represents 2-seconds of multi-channel EEG/EOG. The user can select a segment for analysis and will be presented with a classification of this segment after a period of reasoning. Turbo PROLOG's windowing facilities allow this classification information to be displayed in a 'pop up' window.

One significant advantage in the use of expert system technology for this system lies in the inherent explanatory facilites offered. A

Plate 5.1

classification will be as a result of the successful navigation through a search space such as that depicted in figure 5.13. Explanation for a classification will therefore be readily available if a record is made of the successful rules and conditions encountered in the navigation of the search space. For the implemented system these rules and conditions are displayed to the user, upon request, in a Turbo Prolog 'pop up' window. Chronological ordering of the successful rules and conditions illustrate the reasoning process explicitly.

5.7 SUMMARY OF CHAPTER 5.

Chapter 5 has detailed the implementation of an expert system for OA identification and classification - the intelligent ocular artefact removal system (IOARS), and is based on the theoretical aspects of expert system technology introduced in chapter 2. Fundamental to the development of the expert system is the knowledge which it is to contain, and the implementation here details the manner in which the expert knowledge, elicited in chapter 4, is represented in the expert system. Chapter 4 identified key features used by the expert in the identification of OA and this chapter has firstly, described the process by which these features are extracted from the EEG/EOG by the IOARS, and secondly, the manner in which the knowledge base of rules utilises this feature information, in order to identify OA. Several important techniques are described to match closer the signal analysis abilities of the EEG expert. These include the use of feature fact uncertainty and the use of contextual information to influence a classification.

REFERENCES FOR CHAPTER 5.

Borland.

"Turbo C users guide/reference guide". Borland International, 1988.

Borland.

"Turbo Prolog users guide/reference guide". Borland International, 1989.

Childers, D. G. (editor)

"Modern spectrum analysis". IEEE Press, 1978.

Clocksin, W.F. and Mellish, C.S.

"Programming in prolog". 3rd edition, Springer-Verlag, 1987.

Hashim, S. H. and Seyer, P.

"Turbo Prolog. Advanced programming techniques". Tab books inc., 1988.

Johnson, J.R.

"Introduction to digital signal processing", Prentice-Hall International editions, N.J., U.S.A., 1989.

Kandel, A.

"Fuzzy techniques in pattern recognition". Wiley, 1982.

Kernighan, B. W. and Ritchie, D. M.

"The C programming language (second edition)". Prentice Hall software software series, 1988.

Murray, W.H., and Pappas, C.H.

"80386/80286 Assembly language programming", Osbourne McGraw-Hill, Berkley, Cal., 1986.

Oppenheim, A.V., and Schafer, P.W.

"Digital signal processing", Prentice-Hall, Englewood Cliffs, N.J., 1975.

Rabiner, L.R., and Gold, B.

"Theory and application of digital signal processing", Prentice-Hall,
Englewood, Cliffs, 1975.

Scanlow, L.J.

"IBM PC and XT Assembly language, a guide for programmers". Brady
Communications Company Inc., 1985.

Schildt, H.

"Advanced turbo Prolog. Version 1.1". Borland-Osbourne/McGraw-Hill
programming series, 1987.

Schildt, H.

"C power users guide". Osbourne McGraw-Hill, 1988.

Schildt, H.

"C: The complete reference". Osbourne McGraw-Hill, 1988.

Stan Kelly-Bootle

"Mastering Turbo C". Sybex inc., California, 1988.

STI (Signal Technology Inc.)

"ILS "(Interactive Laboratory system), 5951 Encina Road, Goleta, CA 93117,
U.S.A.

Weiskamp, K., Shammas, N. and Pronk, R.

"Turbo Libraries. A programmer's reference". Wiley, 1989.

CHAPTER 6

6.1 INTRODUCTION.

Chapters 1 to 5 have described the development of an expert system for intelligent removal of OA from the EEG. Chapter 5 has detailed the implementation of a system for OA identification which will enable intelligent OA removal. This implementation however, is the first stage in what is essentially an iterative development cycle, which will eventually lead to a system capable of working routinely in the clinical environment. Essential to this iterative cycle is the feedback obtained from system evaluation which will stimulate refinement and possible redesign during the development process *[Hayes-Roth, et al., 1983]*. This chapter details the procedures used in obtaining feedback from system evaluation, using real world data, together with the results obtained.

Gaschnig, et al., 1983, has proposed a number of criteria for system evaluation, which have been modified and extended by many other authors *[Liebowitz, 1986; O'Keefe et al., 1987; O'Leary et al., 1990; Berry and Hart, 1990]*. This investigation follows the original suggestions of Gaschnig et al., who describe evaluation as considering the following questions:

. Is the knowledge representation scheme adequate or does it need to be extended or modified?

. Is the system coming up with the right answers and for the right reasons?

. Is the embedded knowledge consistent with the experts?

. Is it easy for users to interact with the system?

. What facilities and capabilities do users need?

Of these points, the first three, are viewed as expert system verification, whilst the last two are viewed as decision tool validation. In order to address these questions, the evaluation process is carried out throughout the development of the system. Evaluation is viewed as continuous and provides constant feedback to the development cycle. The development and evaluation cycle is illustrated conceptually in figure 6.1.
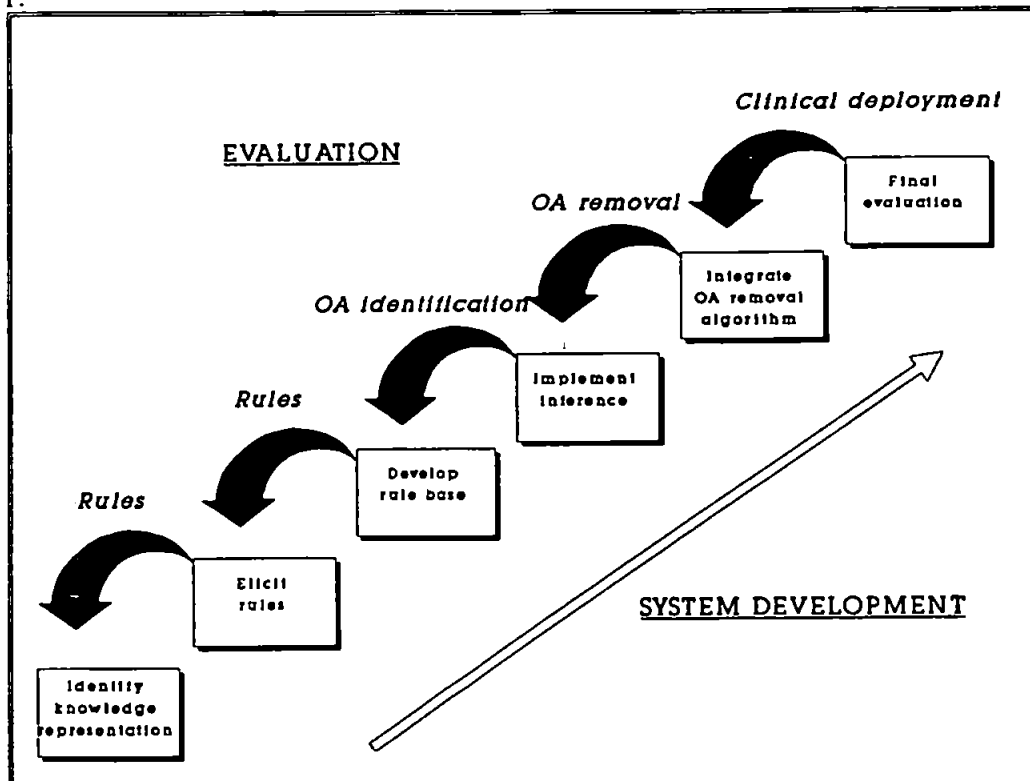
Figure 6.1 Phases of system development and evaluation.

The evaluation phase of the investigation has been planned as four distinct stages, reflecting clear developmental landmarks in the IOARS. Each stage addresses the questions above and are identified as:

STAGE 1: Rules.

STAGE 2: OA identification.

STAGE 3: OA removal.

STAGE 4: Clinical deployment.

Stage 1 evaluates the transcription of the knowledge elicited from the expert, into a set of production rules. Stage 2 evaluates the use of extracted signal features in satisfying the rules. Stage 3 evaluates the application of the OA removal algorithm, based on OA identification. Stage 4 evaluates the use of the tool in the clinical environment. Each stage relies on the verification of previous stages, and the final clinical deployment will be dependent on validation of the system as a genuine tool.

6.2 STAGE 1: RULES.

Section 5.4.1.2 has described a number of the rules elicited from the experts, and transcribed into the production rule format. The consistency of these rules, with the knowledge of the experts, was evaluated by presenting the rules to the expert - a face verification approach *[O'Keefe, et al., 1987]*[1]. The face verification approach is a

---

[1] O'Keefe, et al., use the term validation instead of verification in the description of qualitative evaluation methods. However, in order to maintain consistency with the definitions made in section 6.1, abd the Oxford English Dictionary definitions of verification and validation, the term verification is used here.

useful preliminary technique, which relies on the experts subjective analysis of the rules at face value. This is carried out using their knowledge and intuition of the problem domain, together with their understanding of the system specifications and acceptable performance range. By requesting the expert to provide a value of rule confidence for each rule (see section 4.5.2.2), the expert was forced to address the correctness, consistency, and completeness of the rule set. Rules were written in a form easily understandable by the expert (see figure 6.2) and perceived mistakes were rectified prior to further development.

Face verification of the initial rule set provided the following observations:

Experts Rule confidence (1-5)

Rule 1:
segment could contain abnormal slow waveforms   (5)
if   abnormal slow waveforms have been observed previously.

Rule 12:
segment contains no slow waves   (5)
if   the EEG is flat.

rule 15:
segment contains artefact only   (5)
if   slow activity is maximum in the frontal channels
and   pre-frontal channels are symmetrical
and   EOG channels are symmetrical
and   slow waves only appear in frontal channels

rule 23:
segment contains artefact only   (4)
if   slow activity is maximum in the frontal channels
and   slow waves only appear in anterior channels
and   no slow waves exist in the EEG that are not present in the EOG

Figure 6.2 Example of rules used in stage 1 evaluation.
Appendix N contains the complete list of rules.

. The EEG expert was comfortable with the representation of their knowledge in a production rule format. This does not mean that this is necessarily the best representation, but the heuristic nature of the knowledge lends itself to this representation, and more importantly, the expert was able to express their knowledge easily using this knowledge representation. This indicates that the expert would be comfortable making modifications of the knowledge base, if necessary.

. The expert was satisfied that the key factors in OA identification had been addressed in the rules. In particular, these were, the spatial distribution of potential, the symmetry of potential, and the synchrony of waveforms.

. The expert was less satisfied with the use of waveform shape as a key factor. The rules have related the sharpness of a waveform to classification of OA, i.e. a sharp waveform is likely to be attributable to OA. This was considered a contentious point when applied generally to the EEG, as abnormal spikes (see section 1.1.1) can often occur. However, it has been pointed out that this evaluation is concerned with abnormal slow waves only; In this frequency band it is blinks and fast eye movements only that contain any sharpness in the waveform.

. The expert found difficulty in attributing a value of rule confidence to each of the rules. This is primarily because, it is a task which is not normally carried out explicitly. These values are therefore more likely to be of use as a guide to the statistical values calculated from the data.

. The rules compiled here are elicited from a number of people with considerable experience in the field of EEG analysis. However, all experts are taken from only one centre of EEG analysis and therefore bias could

exist in the rule base. This observation deserves further investigation, and development of the knowledge base should accommodate the knowledge of experts from additional knowledge sources.

6.3 STAGE 2: OA IDENTIFICATION.

Having established a prototype rule set which was correct and consistent, further system development focused on automating the feature extraction procedure. Having obtained a functioning OA identification system implementing feature extraction by using signal processing techniques (see section 5.3), evaluation of system performance was carried out in a rigorous manner. Evaluation at this development stage was carried out at two levels:

. Evaluation using a limited feature set.

. Evaluation using a full feature set.

Evaluation of the system using a limited feature set enabled isolation of an important subsystem - a subsystem verification approach *[O'Keefe, et al., 1987]*[2]. Decomposition of the system into a limited feature extraction subset allows the performance of the subsystem to be observed under a given set of input data. The same input data can then be used with additional feature extraction subsets to observe the effect. In this way each feature extraction subset is verified as it is developed. O'Keefe, et

[2]See footnote 1 on page 230.

al., describes three significant advantages of a subsystem verification approach:

. Verification is incorporated into the development.

. Subsystems are easier to verify.

. Error detection is easier.

6.3.1 EVALUATION USING A LIMITED FEATURE SET.

In order to evaluate the performance of the system 140, 2-second, 16-channel EEG/EOG segments were selected from the extensive patient database (see section 4.2). For a fair cross section of data, segments were selected using a stratified sampling approach; that is, randomly selected within each identifiable result type. 40 segments were selected from normal volunteer data containing various types of OA, and 100 segments were selected from the patient data containing abnormal EEGs. The abnormal EEG segments contained a wide variety of both OA and abnormal slow waveforms. Use of data recorded using standard EEG methods, meant that data segments contained EEG signals recorded from a number of signal derivations. EOG signals were taken as those described in section 4.3.2 and were measured bipolarly using two electrodes for each eye. Figure 6.3 illustrates the EEG and EOG signals used for the data in this evaluation.

Appendix O contains the full evaluation data set, and shows the respective expert and IOARS classifications.
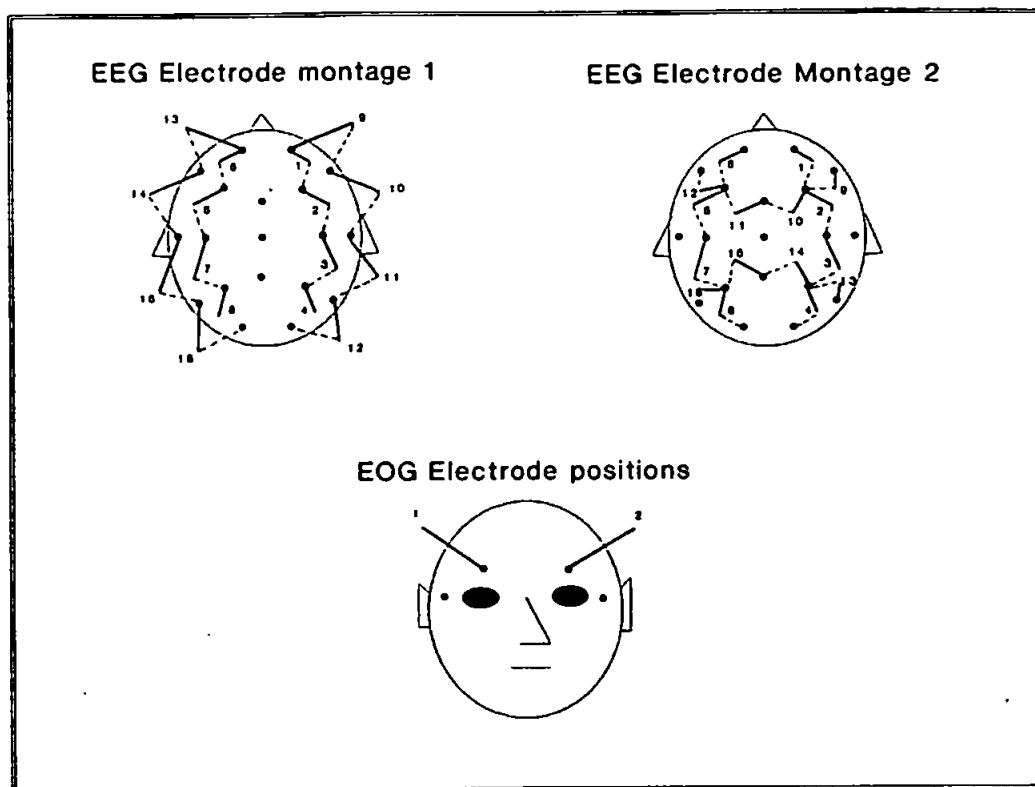
Figure 6.3 Stage 2 evaluation signal derivations.

O'Keefe, et al., describe a number of methods for qualitative system verification [3]. This investigation employs a predictive verification method, involving a comparison of the OA identification abilities between EEG expert, and IOARS.

The selected data segments were presented to an EEG expert, who was requested to classify the slow wave activity in each segment into one of the four categories detailed in section 4.5.1.2; these being:

[3]See footnote 1 on page 230.

. Abnormal only

. OA only

. Both OA and abnormal

. Neither

NB The experts' classification of a segment, as containing abnormal waveforms, was based only on the slow wave, or delta frequency band, contents of each segment, and not on any higher frequency abnormal waveforms.

Segments were also presented to the intelligent OA removal system which was to produce a similar classification. For the purposes of this subsystem evaluation, the Knowledge base could only call upon features extracted from frequency domain, these included:

. waveform frequency

. waveform magnitude

6.3.2 RESULTS OF EVALUATION USING A LIMITED FEATURE SET.

Tables 6.1 details the comparison between the segment classifications of the EEG expert and the IOARS. Appendix P contains the complete results. Table 6.1(a) shows the number of segments classified for the 40 normal, abnormality free, volunteer data segments, and the percentage distribution of classifications of the segments containing OA only. Table 6.1(b) shows

the number of segments classified for the patient data segments, and the percentage distribution of classification of the segments containing OA only.

Number of segments

expert system \	OA Only	None
OA Only	32	0
None	0	8

OA only

Parameter	True	False
Positive	100	0
Negative	0	100

Number of segments

expert system \	OA Only	Abnormal Only	Both	Neither
OA Only	24	1	0	0
Abnormal Only	1	15	4	0
Both	4	13	31	0
Neither	1	1	0	6

OA only

Parameter	True	False
Positive	80	1.4
Negative	20	98.6

(a) Normal Volunteers   (b) Abnormal Patients

Table 6.1 Comparison of segment classification between EEG
expert and IOARS, using a limited feature set.

The results show that for normal volunteer EEG data segments, 100% of segments containing only OA were correctly classified. These segments contained blink, vertical eye movement, horizontal eye movement and rolling eye movement. The abnormal patient EEG data segments show that 80% of segments containing only OA were correctly classified. For the abnormal data segments, the percentage of correct segment classifications was 75.2%. For both normal and abnormal data segments, the overall percentage

of correctly classified data segments was 82.2%. Appendix P shows the incorrectly classified segments as ▇.

Predictive verification of the OA classification, with a limited feature set has allowed the following observations to be made:

. Waveform frequency and magnitude are key features, used by the EEG expert, in the identification of OA.

. EEG exhibiting no abnormal slow waveforms, and containing OA were identified in every case. This is principally because of the large magnitude of the slow waveforms, and the spatial distribution of the waveform magnitude; significant waveform magnitude is often only present in frontal channels, and shows maximum magnitude in the EOG channels. These factors make the presence of abnormal slow waves unlikely. However, it must also be noted, that the data from which the normal data was selected, was from volunteers making deliberate large eye movements; natural eye movements occurring during patient EEG recording are likely to exhibit smaller, more ambiguous, waveform magnitude.

. Examination of table 6.1(b) reveals that 43% of segments containing abnormal slow waves only, from patient data, were wrongly classified, by the IOARS, as containing both OA and abnormal slow waves. These results are marked as   in appendix P. However, this disappointing result is to be expected at this stage. All abnormal slow waves used in this evaluation were of frontal focus; frontal slow waves are the principle problem when applying the OA removal algorithm (see section 1.4.1). Frontal abnormal slow waves are misinterpreted as a combination of abnormal waveforms and OA, when using frequency domain features only, because of a wide spatial

distribution of slow waveform magnitude. The extracted frequency domain features in this case have provided evidence of frontal slow activity, but this activity is not necessarily of maximum magnitude in the EOG channels. This indicates that waveform magnitude and frequency is not sufficient for differentiation between OA and frontal abnormal slow waveforms. It will be necessary, in these cases, to provide further features, such as correlation and phase, to the inference engine.

.   Examination of table 6.1(b) reveals that 87.5% of segments containing OA only, from patient data, were identified correctly. This confirms the suspicions cast above on the equivalent results obtained with normal volunteer data. Normal eye movements occur in a wide variety of types (see section 1.2.1) and also in varying degrees of magnitude. This raises several important issues, namely: (a) what extent of eye movement, if any, is the magnitude considered insignificant, with respect to background noise, to remove the need for OA removal; (b) is the classification of OA directly dependent upon the EEG machine gain adjustment during recording. The first of these issues is approached in the IOARS using a magnitude threshold, in the power spectrum, that is proportional to the average magnitude of EEG frequency band waveform. The magnitude threshold dictates the level of a significant spectral peak. However, a small amplitude OA, with several frequency components, for example when sharp element are visible in the time series, will appear 'smeared' in the power spectrum. Such an eye movement will fail to provide a spectral peak that will exceed the magnitude threshold, without reducing the threshold and introducing erroneous significant spectral peaks in the frequency domain feature list. The second issue is related to the first in that the calculated power spectrum is directly proportional to the

amplitude of the acquired time series data, and will consequently effect the significance of a spectral peaks. For this evaluation all data was recorded using the same value of EEG machine gain, and therefore no adjustment was necessary. However, any development should ensure that power spectral estimation is calculated from the true EEG signal and is irrespective of EEG machine amplifier gain settings.

. Errors occurring in the classification of segments tended to err on the side of safety, i.e. segments that contained slow waveforms which in any way might be considered abnormal, were classified as containing abnormal slow waveforms. This classification ensures minimal application of the OA removal algorithm, but also leads to insufficient OA removal. This observation, together with the previous observation regarding threshold adjustment, indicates that a level of user interaction might be necessary in application of OA removal. This is likely to take the form of a single adjustable level of OA removal, to be made by the user during decision tool use. This will be linked to the quantitative measure of OA removal described in section 4.4.6.

. Generally the final measure of belief for incorrectly classified segments was low. This final measure of belief will also affect the threshold, above which the OA removal algorithm will be applied.

. The measure of belief for segments classified as containing OA only, was higher than for other classifications. Segments classified as containing both OA and abnormal slow waveforms were attributed a much smaller measure of belief (indicated in appendix P as ⌐⎯⎯⌐). For the classification of a segment as OA only, a greater number of rules must be satisfied. Each satisfied rule adds proportionately to the final measure of belief in a classification, and therefore the final measure of belief

can be misleading. The principle of evidence adding to the belief in a hypothesis is sound, however it seems that careful consideration needs to given to the confidence in rules, to prevent inconsistencies. The expert, in providing a segment classification, was not asked to provide a value of belief, this might aid in this problem.

. The values of rule confidence, obtained from the relationship in section 4.5.2.2, were calculated from a small sample of patient EEG data, extracted from the EEG database, for knowledge elicitation. The database itself taken from a sample of patients most likely to display abnormal frontal slow waveforms. These values are appropriate for the evaluation carried out here, because the evaluation data is taken from the same source. However, further development will necessitate that these values are recalculated to accommodate the more general cases.

. The knowledge base compiled to date does not take into account the differences in OA potential caused by eyes being closed, and special medical cases, such as ocular occlusions (surgically removed eyes). It is assumed here that OA potentials are present for both eyes open and closed and simply attenuated for the eyes closed condition. This assumption needs to be further investigated. Special medical cases, such as ocular occlusions etc. will need additional knowledge base facts, to be instantiated upon receipt of further contextual features, such as patients medical history. Such rules will have to compensate for the inevitable resultant asymmetry of OA potential. These specialised cases have not been investigated here.

6.3.3 EVALUATION USING A FULL FEATURE SET.

The previous section has detailed the evaluation of the IOARS with a limited feature set. Development of the feature extraction facilities and extension of the knowledge base have provided an IOARS with feature extraction from the frequency domain, time domain, and the contextual domain (see section 5.3). A second evaluation is detailed here which investigates the performance of the IOARS with the full feature extraction facilities. This closely matches the features that are available to the expert, and will allow the effects of the further feature extraction subsystem to be analysed.

In order to compare this evaluation with that carried out previously, the same 140, 2-second, 16-channel, EEG/EOG segments were selected from the extensive patient database. The data was not shown to the expert a second time, and so the previous expert classification was considered as the true classification. The data was presented to the enhanced IOARS which was to classify the data into one of the same previous four classifications. The classification for each segment was noted, together with the extracted dynamic knowledge base for each segment. The operation of the IOARS was essentially the same as for the previous evaluation. However, incorporation of contextual domain features results in the classification of segment being dependant on previous segments within the same data. Data taken from consecutive time frames of the same patient also added to the contextual domain features.

6.3.4 RESULTS OF EVALUATION USING A FULL FEATURE SET.

Tables 6.2 shows the results of the evaluation of the enhanced IOARS. Appendix Q contains the complete results.

(a) Normal Volunteers

Number of segments

expert \ system	OA Only	None
OA Only	32	0
None	0	8

OA only

Percentage	True	False
Positive	100	0
Negative	0	100

(b) Abnormal Patients

Number of segments

expert \ system	OA Only	Abnormal Only	Both	Neither
OA Only	22	1	0	0
Abnormal Only	1	22	3	0
Both	4	5	33	0
Neither	3	2	0	6

OA only

Percentage	True	False
Positive	73.3	1.3
Negative	26.6	98.7

Table 6.2 Comparison of segment classification between EEG
expert and IOARS, using a full feature set.

The results show that for abnormal patient EEG data the percentage of correct segment classifications, using a full feature set was 80%. This is compared with 75.2% for a limited feature set. The overall percentage classification of segments, including both volunteer and abnormal EEG data, was 86% compared to 82.2% with a limited feature set. However, the

percentage of segments correctly classified as containing OA only fell from 80% to 76%. Appendix Q shows the incorrectly classified segments as ▨.

Predictive verification of the OA classification, with a full feature set has allowed the following observations to be made:

. Incorporation of time domain and contextual domain features into the dynamic knowledge base have improved the overall classification performance of the IOARS.

. Contextual domain features have contributed to classification of segments containing only abnormal slow waveforms. Comparison of appendices P and Q shows that data 14, 15, and 16 illustrate this to the greatest extent.

. Once a segment has been classified as containing abnormal slow waveforms only, the measure of belief in subsequent segment classification is increased. For example, data 16 shows that the measure of belief in the classification of abnormal slow waves only is increased from 0.61 to 0.94, as each segment is analysed. This reflects the increasing belief that an expert is likely to have in the classification, as further waveforms of similar characteristics are observed. The measure of belief is calculated for each segment based on previous classified abnormal segments (see section 5.3.3).

. Errors in classification can be divided into three main types: (a) Very slow artefacts, (b) Small posterior slow waves, (c) Segmentation errors.

Errors occur in segment classification, where the signal data is

contaminated with very slow artefacts, such as rolling eye movements (REM). For example data 4, segments 3 - 4 are incorrectly classified as containing no slow waves. The reason for this error lies in the way the data is presented to the EEG expert and the IOARS. Data is presented to the EEG expert in its raw, unprocessed, form. This allows the expert to analyse the data in a format which closely resembles their normal working medium. The IOARS is however, presented data which has undergone signal pre-processing (see section 5.2.2). This signal pre-processing includes mean removal, and signal band-limiting The effect of the high pass filters on the EEG/EOG is to remove the REM corruption, and therefore no significant spectral peaks are detected in the signals, that correspond to the artefact. This error does not therefore, present a serious problem to the system, because the very slow artefact will have been removed in filtering, and will not need further OA removal.

Six segments were incorrectly classified as containing abnormal slow waveforms. For example, data 5, segment 4 is classified as containing both abnormal slow waveforms and OA. The EEG expert classified this segment as containing only OA. Inspection of the power spectrum for this data, shows that spectral peaks exist in this segment, in posterior channels, and at the same frequency as those in previous segments. Previous segments were classified as containing possible abnormal slow waveforms. This observation indicates that either the threshold for waveform significance in posterior channels is too low, or that the expert has overlooked this waveform for whatever reason. Further investigation of this needs to be made in consultation with the EEG expert.

The segmentation of EEG/EOG signal data is carried out to enable signal processing to be carried out. However, segmentation can lead to misleading power spectral estimates, when slow waveform bursts occur on a segment boundary. A truncated waveform will result in attenuated fundamental component in the spectra, and additional harmonics. When such a condition occurs, power spectra often fail to provide significant spectral peaks in the delta band. The result of this is that a segment is incorrectly classified. Larger segments will create fewer segment boundaries, and hence reduce this problem. However, larger segments are more likely to contain changing statistical properties which can contravene mathematical assumptions made in spectral estimation. Bodenstein, et al., 1985, describe the use of an adaptive segmentation method that can segment the EEG into and segments of differing length, but 'weakly stationary' activity. This method could be investigated to overcome the problem of information loss due to segmentation. The significance of this problem is hard to estimate, however, it will be noted that the data of 15 and the subsequent continuing data of data 16 is correctly classified, once a segment is classified as containing abnormal slow waveforms.

6.4 STAGE 3: OA REMOVAL.

The IOARS has at this stage been evaluated at a number of levels. This evaluation has established the consistency and correctness of the rule set, and quantified the performance of the system in identifying segments of EEG, containing OA contamination. Stage 3 of the evaluation process is concerned with the evaluation of the OA removal process using the segmentation process described above.

The operation of the present IOARS is limited to segment identification only. OA removal is achieved by implementing a real-time adaptive filtering algorithm to those segments identified as containing OA only. The OA-dependant starting coefficients for the filter are derived from off-line analysis, using multiple regression, of segments containing the respective OA (see section 4.3.3.1). These coefficients pre-load the adaptive filter and are then adjusted in real-time, for the duration of the segment only, using the Recursive least squares algorithm. The final coefficients are then stored and act as the starting coefficients for the next segment containing that particular type of OA. This evaluation stage consisted of a qualitative and quantitative examination of OA removal using the metrics described in section 4.4.6.

6.4.1 QUALITATIVE AND QUANTITATIVE EVALUATION OF OA REMOVAL.

Figure 6.4 illustrates an example of the output obtained from the intelligent OA removal system, for a block of data containing OA and abnormal slow waves.
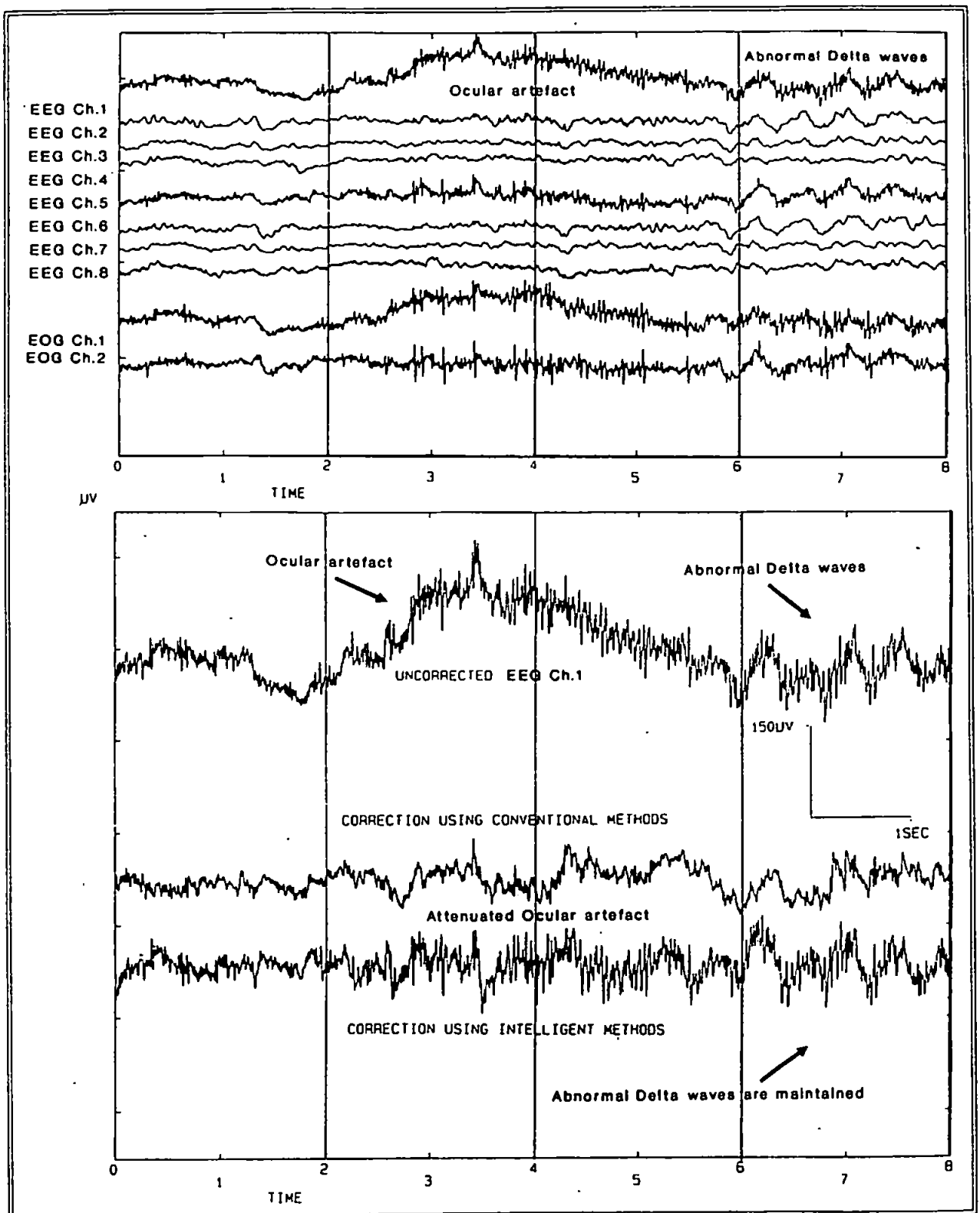
Figure 6.4 Example of the use of the use of the selected

OA removal algorithm on segmented EEG data.

Figure 6.4a shows four contiguous segments of the first eight recorded EEG signals of montage 1, and the two EOG signals. Figure 6.4b compares the EEG of channel 1, before and after OA removal, using the conventional OA removal algorithm, and the intelligent selective approach. Segments 1,2 and 3 are identified, by the IOARS, as containing only OA, whilst segment 4 is identified as containing abnormal slow waves. The classification is made because of the spatio spectral distribution of the slow waves, and the frontal channel correlations. The OA is further classified as being attributable to vertical eye movement (VEM), because of the spectral composition of channels Fp2-F8, and Fp1-F7; There is low frequency delta but none of the additional high frequency components associated with blink type waveforms. An adaptive filter is therefore applied to segments 1,2 and 3, pre-loaded with the VEM OA parameters (see section 4.3.3.1). Segment 4 is passed unaltered, which has the effect of avoiding the corruption of the abnormal slow waves introduced by the conventional OA removal algorithm (illustrated in the middle trace).

Figure 6.4 enables firstly, a qualitative comparison to made between the conventional OA removal approach and the intelligent selective approach, and secondly, a qualitative assessment of the appearance of the EEG after OA removal. Quantification of OA removal is achieved using the process described in section 4.4.6, to obtain the measure of spectral correction (MOSC). Figure 6.5 shows the respective power spectra of the raw EEG, and EEG after OA removal, from conventional, and intelligent methods.

PSD

POWER SPECTRAL DENSITY OF UNCORRECTED AND CORRECTED EEGS

2E9 ⌐

2.5HZ

UNCORRECTED EEG (FP1-F3)

CONVENTIONAL CORRECTION

INTELLIGENT CORRECTION

| 0.0 | | 5.000 | | 10.00 | | 15.00 | | 20.00 |

2.500   7.500   12.50   17.50

FREQ
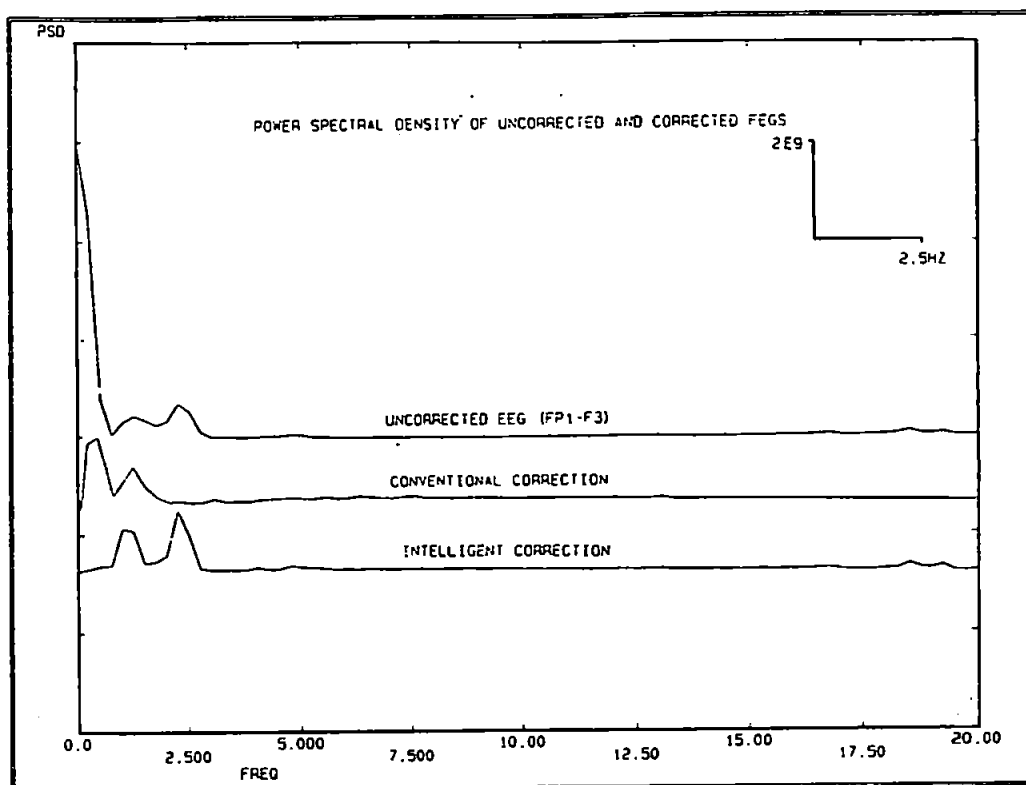
Figure 6.5 Power spectra of: (a) raw EEG, (b) EEG after conventional OA removal, (c) EEG after intelligent OA removal.

Analysis of the results obtained from qualitative and quantitative evaluation of OA removal has enabled the following observations to be made:

. Both methods of OA removal appear to have substantially attenuated the large potential attributable to OA.

. Abnormal slow waves in the EEG are left completely unaltered.

. high frequency 'fine grain' information has been maintained in the EEG after intelligent OA removal.

. The EEG after intelligent OA removal appears to have enhanced slow

waves prior to second 6. This could be linked to the previous observation, and may be caused by 'over correction' in the EEG using conventional OA removal. The enhanced slow waves are also corroborated by the power spectra, which shows a larger spectral peak at 15Hz.

. Baseline offset is an important consideration when applying the OA removal algorithm in the selected manner. This observation was first made in section 4.4.5, when the OA removal algorithm was applied in a directed manner to EEG contaminated with blink type artefact. Mean removal is not always appropriate for minimisation of baseline offset, due to the large monophasic OA potentials, As this can result in a mean which is offset from the natural signal baseline. Offset adjustment is therefore more appropriate through baseline calculation prior to OA removal algorithm application. The baseline can be calculated from a segment of data by constructing a histogram of signal amplitude. The most common signal amplitude is therefore used as the signal baseline. The calculation of this baseline becomes more accurate, as the segment length is increased. Figure 6.6 illustrates a block of evaluation EEG data that has undergone OA removal based on the classification of the IOARS. Figure 6.6b shows the discontinuity at segment boundaries, as a result of OA removal, when the EEG is improperly adjusted for baseline offset. Figure 6.6c shows the effects on OA removal with appropriate baseline adjustment.
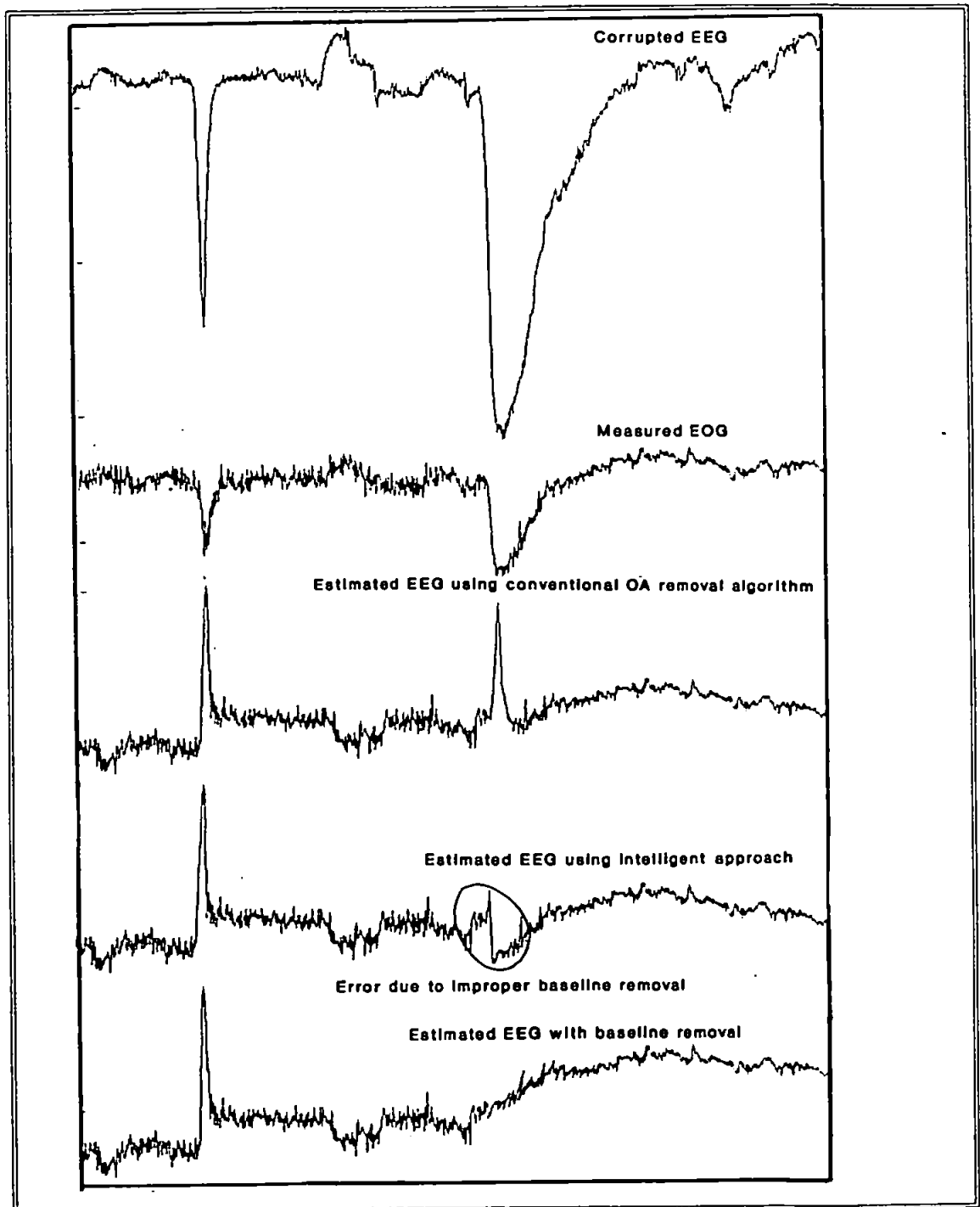
Figure 6.6 The effects of baseline offset on OA removal

using the intelligent, selective, approach.
page 252

6.5 STAGE 4: CLINICAL DEPLOYMENT.

Evaluation of system performance to this point has concentrated on the verification of the expert system side of the IOARS. This has included knowledge base correctness, and classification consistency. However, section 6.1 has described the evaluation process as one of decision tool validation, in addition to the expert system verification. This stage of the evaluation process is concerned with feedback from the potential end users of such a system. The end user feedback will be valuable for a number of reasons:

. To validate the concept of an OA removal decision tool.

. To provide further system verification using unknown data.

. To re-evaluate user requirements.

. To establish appropriate interactive interface requirements.

O'Leary, et al., 1990, propose a 'staged third party' verification paradigm, which involves three parties in the verification process; the system designer, the expert, and a third party who acts as coordinator and is responsible for verification. This paradigm has been adopted here for the clinical evaluation stage.

The clinical evaluation stage of any new development is likely to be long and rigorous. This investigation details the first preliminary clinical evaluation of the use of the IOARS, conducted in a clinical neurophysiology department of a busy general hospital.

6.5.1 PRELIMINARY CLINICAL EVALUATION.

Forty, 2-second, 16-channel EEG/EOG segments were selected from the extensive patient database (see section 4.2). The segment chosen were ones that had not previously been used for evaluation. For a fair cross section of data, segments were selected using a stratified sampling approach; that is, randomly selected within each identifiable result type. 16 segments were selected from normal volunteer data containing various types of OA, and 24 segments were selected from the patient data containing abnormal EEGs. The abnormal EEG segments contained a variety of both OA and abnormal slow waveforms. For this evaluation EEG data was standardised to the montage 1 configuration (see section 4.2.3). Abnormal patient data was selected from data recorded in montage 1. Volunteer data, which was recorded using referential techniques, was re-montaged into a montage 1 configuration; this was possible because the referential signals enabled derivation of the bipolar montage 1 signals, and all signals were reference to the same source (linked ear lobes). EOG signals were taken as those described in section 4.3.2 and were measured bipolarly using two electrodes for each eye.

The selected evaluation data was presented to the EEG expert (senior EEG technician) in order to make a further independent selection of the data with which to evaluate the system. The result of this selection was 32, 2-second, 8-channel, EEG/EOG data segments, that were unknown to the system designer. Having selected appropriate evaluation data, both the expert and the coordinator were issued with identical versions of the IOARS and evaluation data for them to evaluate the system. The full evaluation data is contained in appendix R.

The IOARS was designed to enable the user to select and display, in real time, any of the evaluation data blocks. Classification of a segment could be initiated by the user, who was informed of the classification and the reasons why. The user was requested to confirm or reject the classification that the system made, and to make observations on the reasoning behind the classification. The system was designed to log all user responses, and, for each segment, the extracted dynamic knowledge base and list of successful rule conditions. The use of a log enabled error tracking to be carried out, and the usability of the IOARS to be evaluated *[Berry and Hart. 1990]*. The expert log is attached as appendix S.

6.5.2 RESULTS OF PRELIMINARY CLINICAL EVALUATION.

Table 6.3 illustrates the quantitative results obtained from the data

OA only		
Pamamas	True	False
Positive	88	14
Negative	12	86

Number of segments				
expert \ system	OA Only	Abnormal Only	Both	Neither
OA Only	16	0	0	1
Abnormal Only	0	3	0	0
Both	0	0	0	0
Neither	2	0	0	3

(a) Abnormal Patients

Table 6.3 comparitive results in segment classification between

EEG expert and IOARS.

used in the preliminary clinical evaluation. Appendix T contains the complete set of evaluation results.

Analysis of the expert log, evaluation results, and user comments have enabled the following observations to be made:

. Discussion with the expert examined the applicability of the knowledge base developed so far. This is limited to one EEG recording montage and does not take into account the state of the eyes, i.e. open or closed. The knowledge base is specific to an EEG recording montage because of the need to know the derivation, and spatial positioning of signals, in order to infer their meaning. An alternative EEG recording montage is accommodated by supplying a new knowledge base with alternative signal characteristic facts (see section 5.4.1.2 which describes the structure of the static knowledge base). However, a final clinical IOARS would include the front end EEG recording equipment and so this problem would be most easily overcome, by recording all signals referentially. All traditional EEG montages could then be calculated from these signals for display purposes.

The state of the eyes does not present a serious problem to the IOARS. It is assumed that OA due to eye movements, even when closed, are characterised in a similar manner as that described in section 4.5, i.e. the spatio spectral characteristics remain unchanged, and it is only the amplitude and time domain characteristics that are effected. However, for the purposes of the evaluation carried out in this investigation, patient data was recorded early in an examination, using montage 1. The data is therefore from alert patients with eyes open. Transitory eye closures appear as normal eye movements and blinks.

. The expert expressed a satisfaction with the operation of the IOARS. The selection of data was straightforward and the explanation facilities were easily understood. The evaluation system was not equipped with full real time signal processing facilities and the graphical data representation requires further work. It was observed that user interaction via a standard keyboard was 'clumsy', and more sophisticated media, i.e. touch screens would be more appropriate.

. The display of data in this evaluation, only gave the expert the first eight channels of EEG on which to make a classification. This needs to be extended to a possible 24 or 32 to allow signals from the entire scalp to be used in the analysis.

. The clinical evaluation data showed an overall percentage of correctly classified segments of 88%, a percentage of segments correctly classified as OA of 88%, and a percentage of segments correctly classified as containing only abnormal slow waves of 100%, these results are not dissimilar to the results obtained from the second phase evaluation of stage 2, but the number of segments finally used in the evaluation is not high enough to give too much confidence in the results. The small number of segments was chosen because of the time needed on behalf of the expert, and the preliminary nature of this evaluation. A further more detailed evaluation is required, using the same evaluation paradigm. Appendix T shows errors in classification as﹎.

. The preliminary clinical evaluation showed a consistent error in the classification of segments containing only OA. These segments are further classified into the type of OA present, and this classification was consistently reversed, i.e. segments that the expert had classified as containing blink type OA, were wrongly classified as containing vertical

eye movement type OA, and segments that the expert had classified as containing vertical eye movement type OA, were wrongly classified as containing blink OA. Appendix T shows these errors in OA classification as▓.

. The backward chaining PROLOG inference engine causes a problem in OA classification. Section 2.2.2.1 described the importance of rule order to the resolution path, and that careful consideration needs to be taken when compiling the list of fact and rule clauses. For example, the static knowledge base contains a series of rules that are used to classify OA into OA type. These rules are attempted sequentially, i.e. the rule for blink type OA is only tried when the rule for VEM type OA has failed. The implication of this is that the conditions for the rule to classify VEM type artefact must be complete and correct. Failure to accommodate every variation of VEM type in the conditions that satisfy this rule will result in possible failure of the rule, and subsequent classification of the OA as an alternative OA type. This problem becomes apparent in the transcription from the conditions elicited by the expert, to the conditions that are used to obtain this information from the extracted features.

. The expert classification of OA type, is achieved using four key features: (a) the symmetry of the EOG, (b) the position of maximum OA potential, (c) the phase relationship between signals around the eyes, and (c) the shape of the OA. Investigation of the expert log, and in particular the dynamic knowledge base for the incorrectly classified segments, has revealed a number of sources for error in the completeness of the transcribed conditions that classify OA into OA type:

The symmetry of EEG signals is measured by comparing the power spectral peaks of each channel. In order to be symmetrical the same

spectral peaks must appear in both channels. This is clearly inadequate and has proved a source of a number of errors. For example, the IOARS reports asymmetry in the EOG channels for data 2, segment 1. This is clearly not the case. The reason for this error lies in the manner in which the spectral peaks are tokenised, to appear as features in the dynamic knowledge base. Spectral peaks are tokenised for magnitude, using three magnitude bands (see static knowledge base facts - power bounds, in appendix M). This tokenisation does not use fuzzy logic, and therefore a spectral peak is tokenised to a power band in a binary logic. This causes errors when two spectral peaks are close in power, but on different sides of the power band threshold. For data 1 segment 2, three spectral peaks are detected in the EOG channels, and only one is considered asymmetrical. The result is that the EOGs are reported as asymmetrical. Fuzzy logic could be used to provide flexibility in the power band tokenisation, thus avoiding the problem of threshold adjustment. However, a more appropriate method would be to use the cross-correlation of the two EOG channels to obtain the measure of symmetry. The cross-correlation with zero time shift will give the instantaneous symmetry, and the normalised cross-correlation coefficient will be regarded as the measure of certainty, that the inference engine will use.

The position of maximum OA potential, is measured by comparing the magnitude of spectral peaks above the eye, to those to the side of the eye. Vertical eye movement (VEM) and blink should be greatest above the eye, and horizontal eye movement (HEM) should be greatest to the side of the eye. However, the tokenisation of power bands, and the more likely occurrence of the greatest spectral magnitude in the EOG,

renders these conditions open to error. For example, data 1 segment 2 is clearly identified as containing OA attributable to VEM, but, the IOARS attributes the OA to blink. The alternative to this condition is to firstly test that the potential is maximum in frontal channels, and then test the phase of the EOG potential alone, using the cross-correlation of the two EOG channels. VEM and blink will provide EOGs which are in phase, whilst HEM will provide EOGs that are out of phase by approximately 180 degrees.

Shape of the OA is estimated using the frequency content of the EOG channels. It is observed that the spectral content of a VEM is a single wide spectral peak in the lower half of the delta EEG frequency band. The spectral content of a blink is that of a fundamental component in the lower half of the delta EEG frequency band, and a number, (2 or 3), of harmonics in the upper half of the delta EEG frequency band. OA is therefore classified as being attributable to blink, if there are power spectral peaks in the upper half of the delta EEG frequency band. However, this transcription of the condition is open to error, if the VEM contains very fast rising edges and is short in duration, i.e. the closer to blink that a VEM appears, the more prone the condition is to error. It is noted however, that the EEG expert also experiences difficulties in these situations. However, other conditions also result in spectral peaks in the upper half of the delta EEG frequency band, such as, noise, and abnormal slow waves. Under these conditions the spectral peak estimation of waveform shape is inadequate and alternative algorithms are required. Alternative algorithms could include, syntactic analysis *[Fu, 1982; Lister, and Bishop, 1988]*, fractals *[Katz, 1988]*, neural networks *[Gevins and Morgan, 1988]*.

6.6 SUMMARY OF CHAPTER 6.

Chapter 6 has detailed the evaluation of the implemented IOARS. The evaluation carried out to date has concentrated on the verification and validation of the system. This has included, the consistency, correctness, and completeness of the classification rules, together with end user requirements, and system I/O capabilities. Four distinct stages of evaluation are described which reflect the ongoing development of the system. The development, and hence the evaluation, is not complete, but three separate evaluations have indicated that the IOARS is capable of identifying segments containing OA, and to differentiate segments containing OA from those containing abnormal frontal slow waveforms. This is shown to be the case in approximately 80% of segments under evaluation. Evaluation has provided insight into errors in the transcription of rules elicited from the EEG expert, into rules which test the signal features extracted using signal processing. These errors are likely to continue, whilst complex human signal analysis is estimated using simple signal processing functions.

REFERENCES FOR CHAPTER 6.

Berry, D.C., and Hart, A.E.

"Evaluating expert systems", Expert systems, Vol. 7, No. 7, pp. 199-208, 1990.

Bodenstein, G. and Praetorius, H.M.

"Feature extraction from the electroencephalogram by adaptive segmentation". Proc. of the IEEE, Vol.65, No 5, pp. 642-652, May 1977.

Bodenstein, G., Schneider, W. and Malsburg, C.V.D.

"Computerised EEG pattern classification by adaptive segmentation and probability-density-function classification. Decription of method". Comput. Biol. Med, Vol. 15, No. 5, pp. 297-313, 1985.

Buchanan, B.G., Barstow, D., Bechtal, R., Bennett, J., Clancy, W., Kulikowski, C., Mitchell, T., Waterman, D.A.

"Constructing expert systems", in "Building expert systems", Hayes-Roth, Waterman, and Lenat, eds., Addison-Wesley, Reading Mass., 1983.

Fu, K.S.

"Syntactic pattern recognition and applications". Prentice-Hall, 1982.

Gaschnig, J., Klahr, P., Pople, H., Shortliffe, E., and Terry, A.

"Evaluating expert systems: Issues and case studies", in "building expert systems", Hayes-Roth, Waterman, and Lenat, (Eds.), Addison-Wesley, Reading, Mass., 1983.

Gevins, A.S., and Morgan, N.H.

"Application of neural-network (NN) signal processing in brain research", IEEE Tran. Acoustics, Speech and Signal Proc., Vol. 36, No. 7, 1988.

Hayes-Roth, F:, Waterman, D.A., Lenat, D.B. (Eds.)

"Building expert systems", Addison-Wesley, Reading, Mass., 1983.

Katz, M.J.

"Fractals and the analysis of waveforms", Comput. Biol. Med., Vol. 18, No. 3, pp. 145-156, 1988.

Liebowitz, J.

"Usefull approach for evaluating expert systems" Expert systems, Vol. 3, No. 2, pp. 86-96, 1986.

Lister, P.F., Bishop, M.L.

"General waveform shape analyser", IEE Proc., Vol. 135, Pt. F, No. 5, 1988.

O'Keefe, R.M., Balci, O., and Smith, E.P.

"Validating expert systems", IEEE expert, pp. 81-90, Winter, 1987.

O'Leary, T.J., Goul, M., Moffitt, K.E., Radwan, A.E.

"Validating expert systems", IEEE expert, pp. 51-58, June 1990.

CHAPTER 7

DISCUSSION, FUTURE WORK, AND CONCLUSIONS

7.1 INTRODUCTION.

This investigation has described the development of new and more reliable methods of removing ocular artefacts (OAs) from the human electroencephalogram (EEG). This development has involved the incorporation of digital signal processing (DSP) techniques, within an artificial intelligence (AI) paradigm, to match more closely, the implicit signal processing capabilities of the EEG expert *[Hellyar, et al., 1991]*. The integration of EEG expert knowledge into the OA removal algorithm, allows real-time adaptive filters to be applied to the EEG, in a selective and directed manner. The EEG is enhanced by only removing OA, when OA is identified. This avoids the corruption to the EEG observed when the EEG contains abnormal frontal slow waves *[Ifeachor, et al., 1988]*. The intelligent OA removal system (IOARS) presents a significant improvement in OA removal, and EEG signal processing, and will allow more reliable automated EEG analysis. This final chapter attempts to unify the fundamental aspects of this investigation, in the form of a critical discussion, a view of future work, and finally, the conclusions that can be made from this investigation.

7.2 DISCUSSION.

This section critically discusses the key issues which have emerged during the investigation, and draws upon, observations from developmental stages, results, and conclusions made from rigorous development evaluation. These issues are discussed under two main section headings:

. PERFORMANCE.

. VALIDITY.

Many key issues exist under these main section headings and figure 7.1 illustrates the relationship between the issues discussed in this chapter.
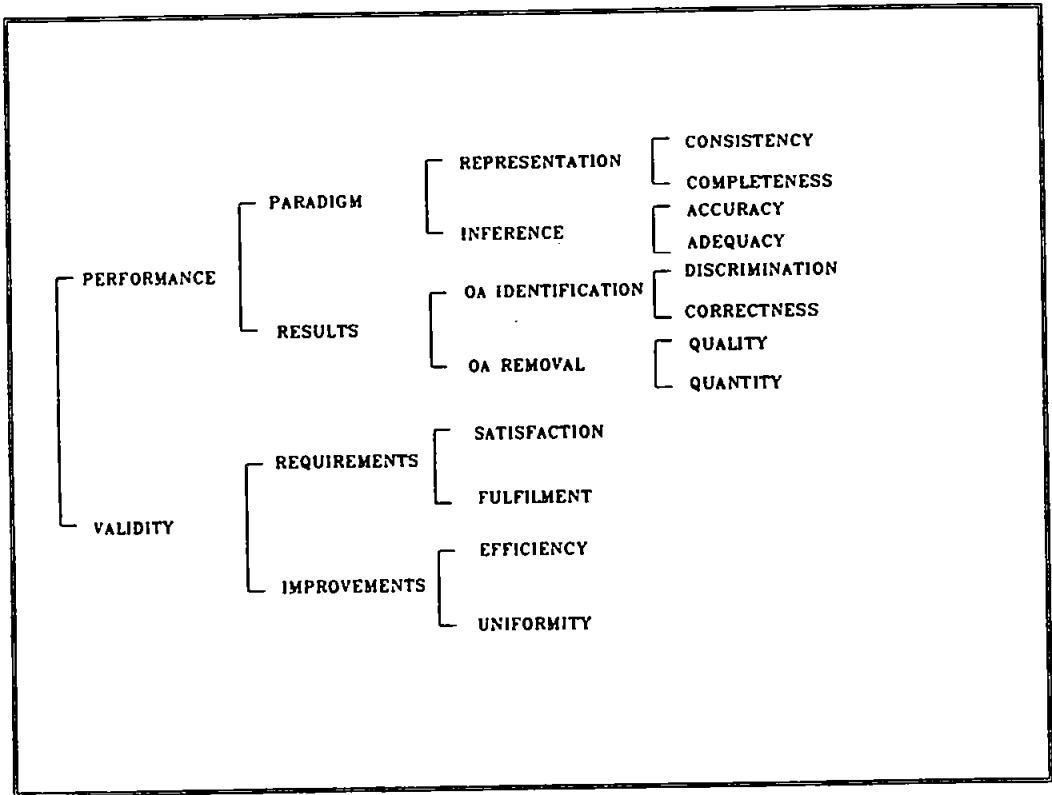
Figure 7.1 Relationship between discussion issues.

7.2.1 PERFORMANCE.

The performance of the IOARS has been evaluated in chapters 4 and 6 of this thesis. Chapter 4 has discussed, in detail, the modifications made to the adaptive on-line OA removal algorithm. The modifications made to the OA removal algorithm, were as a result of the problems encountered with current OA removal techniques, and described in chapter 1. The modifications made, followed the proposed techniques described in chapter 2, as a means of overcoming the deficiencies of the existing OA removal techniques. Chapter 6 has discussed the evaluation of the software implementation of the IOARS. The implementation of the IOARS was expounded in chapter 5, based on the proposed intelligent techniques described in chapter 2. The following sections discuss individual aspects of IOARS performance taken from those shown in figure 7.1.

(a) CONSISTENCY.

The issue of consistency is related to the representation of the elicited expert knowledge as a number of production rules. Questions that arise from this issue are: whether (i) production rules a good representation of the experts knowledge, (ii) the elicited rules accurately represent the experts knowledge, and, (iii) the transcribed conditions of rules, used by the system, match the conditions used by the expert. Production rules are not the only representation that could have been used in this investigation; other representations exist *[Graham and Jones, 1988; Winston, 1984]*. However, the production rule representation of knowledge is simple and easily modified, and has been used successfully

in a large number of applications *[Jansen, 1985; Bourne, et al., 1983; Jagannathan, 1981]*. The expert (consultant clinical neurophysiologist) found the representation immediately comprehendible, and was able to suggest new rules and modifications using this format. This enabled the accuracy of the rule representation to be evaluated at an early stage (chapter 6). The disadvantage of this representation lies in the large number of rules necessary to account for small variations in waveform characteristics. It was found that the expert had difficulty in expressing the reasons for some classifications. This places doubt on the validity of rule conditions. Knowledge elicitation, using additional experts from different locations, will generate more rules, but will also verify the conditions. This problem is linked to the question of condition transcription. It was found that transcription of the conditions of an experts rule, to that which the system could use to obtain information from the extracted features, was very subjective in nature. The transcription was based on the system designer's (the author) understanding of EEG characteristics. Whilst this was obtained from the expert, via knowledge elicitation, discrepancies have occurred (chapter 6). The transcription of conditions requires careful evaluation to avoid representational errors.

(b) COMPLETENESS.

The completeness of the rule set is related to the scope and efficiency of the knowledge elicitation. The structured knowledge elicitation used in chapter 4, relied on data from the EEG database (chapter 3). This data was acquired from patients exhibiting frontal slow waves, because of the need to improve OA removal in the presence of

abnormal frontal slow waveforms. The scope of the knowledge elicitation was therefore, limited to the characterisation of OA and frontal slow waves. The scope of the rule set will need to be extended to other types of artefact and other abnormal waveform types. The efficiency of the knowledge elicitation is indicated by the effectiveness of the IOARS at OA identification. The results shown in chapter 6 indicate an acceptable level of OA detection. However, further data will provide more variations in waveform characteristics, and therefore require rule set extension.

(c) ACCURACY.

The accuracy of the IOARS depends upon the correct classification of OA, for the right reasons. This was evaluated by the experts during both internal, and clinical studies. The inference path was traced for each classification to establish the reasoning process. This reasoning process was also logged during the preliminary clinical evaluation, an example of which is included in appendix S. An intrinsic part of the reasoning process, and one which will effect the removal of OA, when a classification is made, is the measure of belief in the classification, MB (chapter 4). The measure of belief is calculated from uncertainties accrued in the reasoning process. Uncertainty of rules was obtained as a combination of elicited values and statistical values, both of which are highly subjective; elicited values because of the use of only one expert, and statistical values because of the limited scope of patient data in the EEG database. Additional experts should be used to obtain these values, and a protocol for conflict resolution established. Further data acquisition should be carried out to obtain more accurate statistical

values. Uncertainty of conditions was obtained from digitally signal processing the waveforms. For example, the fuzzy frequency set membership of a peak in the power spectrum (chapter 5) *[Jagannathan, et al., 1982]*. However, the use of fuzzy sets was only utilised in the frequency dimension, and this need to be extended to the magnitude dimension of the power spectrum to account for variations in magnitude due to noise and inaccuracies in the FFT estimation techniques.

(d) ADEQUACY.

The adequacy of the inference mechanism used in the IOARS hinges on the use of the backward/forward chaining hybrid, with uncertainty handling. This discussion is not concerned with whether or not this paradigm is a good model of human reasoning, as this is beyond the scope of this thesis. This discussion is solely concerned with the use of this paradigm as a model for OA classification. This paradigm is logical and simple to comprehend, but significant limitations exist. Firstly, there is no accommodation for alternative classifications in the reasoning; the rules must be constructed to account for every waveform variation. Only when one rule fails, is another path/alternative attempted. This lead to several errors in OA type classification in the preliminary clinical evaluation (chapter 6). It would be preferable to be able to obtain several alternative classifications, and then to select the most likely, or the one with the greatest measure of belief, using some pre-defined protocol for conflict resolution. Secondly, the use of Certainty Factors *[Shortliffe, 1976]*, to accommodate uncertainty in the reasoning process, is open to some debate *[Tonn and Goeltz, 1990; Graham and*

Jones, 1988]. Tonn and Goeltz, (1990), found that the use of certainty factors was acceptable, but showed discrepancies with experts when ·combining compound certainty values. The use of certainty factors has been justified (chapter 4), and whilst no universally accepted alternative exists, this method represents a straightforward and easily comprehending solution.

(e) DISCRIMINATION.

The discrimination of the OARS refers to the ability of the system to differentiate between OA and abnormal frontal slow waves. This criterion was a primary motivation for this research, as it was found in chapter 1. The successful application of the OA removal algorithm was highly dependent upon this discrimination *[Ifeachor, 1984]*. Chapter 6 has documented a number of evaluations that incorporated the discrimination between OA and abnormal frontal slow waves taken from the acquired patient EEG database. The overall percentage of correctly discriminated data segments was 82.7%. This indicates that the discrimination of OA and abnormal frontal slow waves is possible, given the sample data, and that consequently, a reduction in EEG corruption due to incorrect OA removal algorithm application is possible.

(f) CORRECTNESS.

The IOARS has been shown to discriminate between OA and abnormal frontal slow waves in 82.7% of cases. However, the correctness of the classification, i.e. OA type, as a secondary issue to OA identification, will effect the OA removal algorithm application. The preliminary clinical

evaluation data (chapter 6), showed that there was a consistent error in this classification due to unsatisfactory condition transcription (see point (a) above). This error was in contradiction to the results obtained in a previous evaluation study (chapter 6) and further examination of this is required.

(g) QUALITY.

The quality of OA removal, having applied the removal algorithm in a selective manner was assessed visually in chapter 6. The visual assessment made was the same as described in chapter 4, and that prescribed by Ifeachor, (1984). Visual analysis lacks an easy quantitative description and can be subjective in nature. However, this type of analysis has been found, in previous studies, to be extremely sensitive to remnant artefact caused by unsatisfactory OA removal *[Ifeachor et al., 1988]*, and to the distortion of clinically significant EEG waveforms. In addition to this, visual analysis will be the final test for clinical acceptance and is no more subjective than conventional EEG analysis. The quality of OA removal, using the selective and directed OA removal algorithm was considered to be superior to that of the conventional algorithm.

(h) QUANTITY.

Quantification of OA removal was achieved using the measure of spectral correction (MOSC), detailed in chapter 4, and based upon the method of Gratton et al. (1983). The MOSC provides a continual evaluation of removal algorithm performance, by comparing an EEG signal before and

after OA removal, to an uncontaminated reference EEG with similar characteristics. However, use of the MOSC has highlighted a number of shortcomings: (a) the MOSC is insensitive to subtle qualitative differences in OA removal, such as waveform shape (chapter 4), (b) the MOSC will be biased towards the differences in low frequencies containing greater power, (c) the MOSC is susceptible to errors caused by secondary artefacts, such as muscle activity, which will produce a small finite error, and, (d) the MOSC is susceptible to variations in EEG signal content, which is exaggerated by the nonstationarity of the EEG signal. Whilst no measure of true EEG signal is available this quantitative measure provides a good indication of removal algorithm performance. A comparison of the MOSC obtained from contaminated EEG signals subjected to intelligent OA removal and conventional OA removal, was in agreement with the results obtained from qualitative analysis.

7.2.2 VALIDITY.

This section discusses the validity of the IOARS as a useful decision support tool for the clinical environment. The validity of the system address two issues, as illustrated in figure 7.1, i.e., the requirements made of the system, and the improvements made possible by the system. Many of the issues attached to IOARS validity will not be fully answered until comprehensive clinical evaluation can be made. However, close collaboration with the clinical department, experts, and preliminary clinical evaluation have allowed the validity to be partially assessed. The following sections discuss individual aspects of the validity issues.

(i) SATISFACTION.

Chapter 2 described the role of the IOARS as twofold; firstly, as a tool to be used at recording time, and secondly during post recording data analysis. This thesis has mainly addressed the later of these roles, as a post recording data analysis tool. In this role the IOARS is able to scan through a large amount of EEG data, allowing the user to select data segments for analysis, identifying OA without necessarily performing OA removal. The system is able to justify any identified OA, by tracing the reasoning process and features extracted from digital signal processing. This is a fundamental decision support aid for the EEG expert. Once the user is satisfied with the reasoning behind a segment with identified OA, the removal algorithm may be applied. This provides the EEG expert with an intelligent EEG signal processor.

(j) FULFILMENT.

The IOARS fails to fulfil a number of the initial requirements; (a) operation in real time, (b) fully graphical user interface. The operation, as described in chapter 5, is essentially off-line. This involved performing the time consuming DSP operations on multi-channel signals, prior to system operation. However, in addition to this, the time taken for segment analysis and OA identification would take as long as 10 seconds, for a 2-second, 18-channel data segment. This is obviously prohibitive to real-time operation, even if the DSP operations are performed on dedicated multi-channel DSP hardware. The solution to this problem is likely to be twofold. Firstly, overall speed improvement, is

achievable with faster computing resources; a 25MHz 80486 PC is likely to show at least a halving of the classification time. Secondly, the program code can be optimised. This will involve intensive code analysis and re-compilation, using C or assembler, in time consuming areas. The graphical user interface has been described in chapter 5. However, a real-time system would require OA to be identified during data acquisition; the OA would be indicated on the screen as the data scrolls across. This is likely to require dedicated graphic processing components and a great deal of further work before a commercially viable end product could be realised.

(k) EFFICIENCY.

Intelligent OA removal makes the concept of automating EEG analysis a more realisable goal. Previous research *[Kytonas, 1988; Bourne, et al., 1983; Michael and Houchin, 1979; Barlow, 1977; MacGillivray, 1977; Praetorius, et al., 1977]* have attempted to automate the evaluation of EEG signals. However, all systems documented to date, have failed to successfully address the issue of artefact contamination; at best detected artefactual signals are ignored. Chapter 1 described the presence of OA as continuous, and in a patient with uncontrollable eye movements, simple artefact rejection will result in meaningless data. Kytonas, (1988), describes a number of schemes for automated analysis of epileptic EEG, typified by sudden EEG waveforms. In all cases, false detections of true epileptic activity due to blink type artefact seriously degraded performance. Successful OA processing, and in particular intelligent EEG enhancement through selective OA removal, will allow the performance of

these systems to be drastically improved. Attention to the less common artefactual type will eventually allow the full potential of the automated EEG analysis systems to be realised.

(I) UNIFORMITY.

Chapter 2 described EEG analysis as heuristic, as such differences in interpretation can exist between EEG centres. Explicit documentation of the EEG analysis protocol, and waveform characterisation, have allowed the development of a small knowledge base. Extension of this knowledge base, to incorporate further expert knowledge from different centres, together with the characterisation of different waveform patterns, will standardise the analysis of the EEG.

7.3 FUTURE WORK.

This section discusses the issues concerned with the projection of this novel research, in to the routine clinical environment. The research documented in this thesis is still in an early stage of development, and extensive further evaluation will be necessary before this technology is readily accepted in the medical environment. This is inevitably a slow process, that will involve piecewise introduction of computer aided decision making, and the gradual familiarisation of the required technology, on the part of the medical personnel. This may seem a daunting task on initial inspection. However, the potential benefits associated with the introduction of integrated computer patient data

storage, diagnosis, and treatment evaluation, is recognised by many as incentive enough to pursue this challenge. This thesis has dealt with only one small aspect of clinical data analysis and management, and many other research areas and projects exist. However, the results indicate that EEG signal processing can be improved and this may lead to intelligent EEG analysis support tools which will enable the busy clinician to use their time more efficiently. Three areas of importance for future research in this project are: hardware implementation, neural networks, and automated learning. These issues will be dealt with briefly.

7.3.1 HARDWARE IMPLEMENTATION.

Much of the time consuming processing of the signals, currently carried out in C and using the ILS software package, will be more appropriately accomplished using dedicated signal processing hardware, such as the TMS32030 which performs a 1024 complex point FFT in less than 3 milliseconds. Figure 7.2 illustrates the conceptual view of an appropriate hardware implementation. Much of the hardware utilised in the data acquisition system, described in chapter 3, will be utilised. However, multiple dedicated DSP processing chips, such as the Texas TMS320C30 *[Texas, 1989]*, or the Motorola DSP96000 *[Motorola, 1989]*, are included in each signal path. These devices are chosen because of their speed and dynamic range. These processors will calculate power spectral density, auto and cross-correlations, to provide the relevant features for signal identification. Supervision of the signal processing channels, will be performed be another dedicated processor. Communication will be bidirectional to allow interrogation of signal channels should further processing be needed to resolve conflicting information.

```
                    Bandlimited signals           Extracted features

  →  ┌─────────────┐  →  ┌─────────┐  →  ┌─────────┐  →  ┌─────────┐
     │Instrumentation│    │Antialiasing│    │Analogue to│    │ PSD. and │
     │  Amplifier   │    │  Filter  │    │ digital  │    │Correlations│
     └─────────────┘    └─────────┘    │Conversion│    └─────────┘
                                        └─────────┘
                                                          TMS32030
  Raw EEG/EOG                                                or
                                                          DSP96000
                                                                    Dynamic knowledge

                  upto 32 signal channels...              ←→  ┌─────────┐ ←→
                                                               │Control and│
                                                               │ Mapping  │
                                                               └─────────┘
                                                               TMS32030
                                                                  or
                                                               DSP96000
```
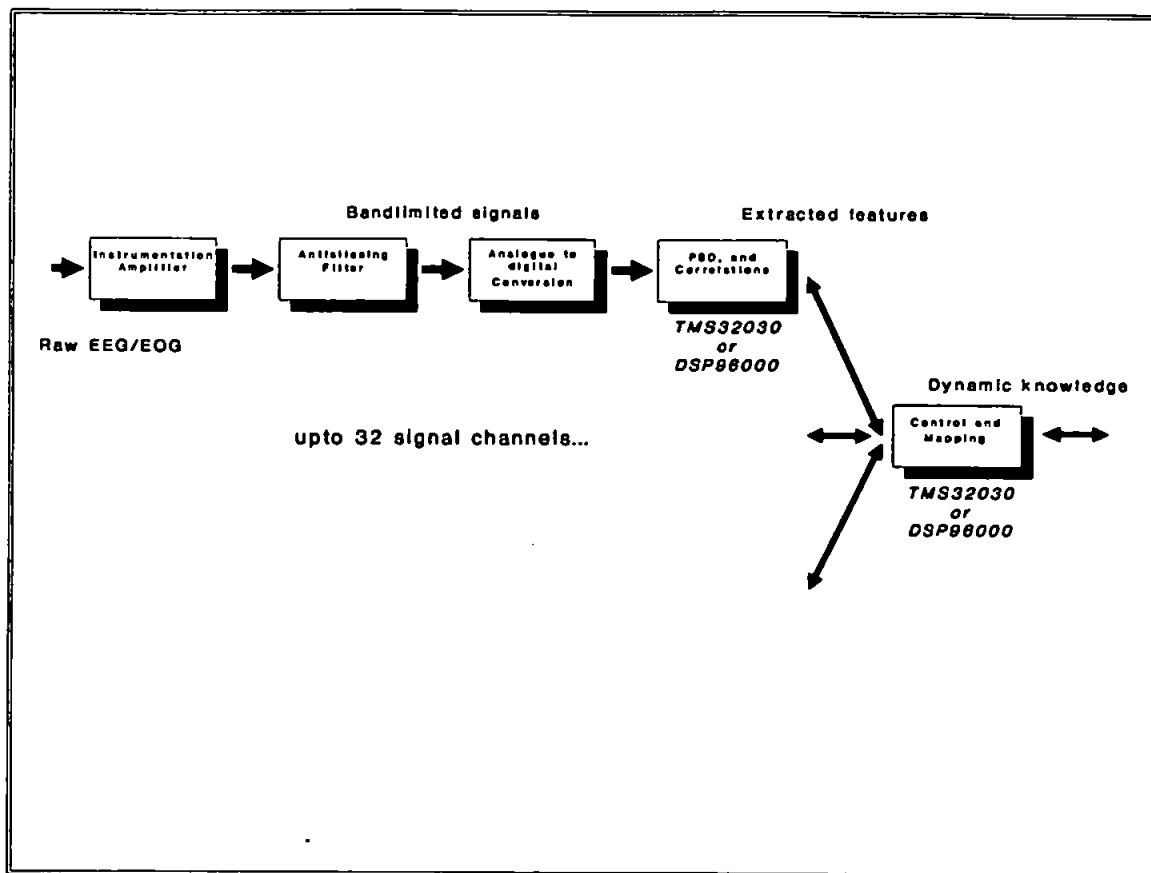
Figure 7.2 Conceptual hardware implementation of IOARS signal processing.

7.3.2 NEURAL NETWORKS.

In areas of waveform classification, where the expert finds difficulty in verbalising the analysis procedure, feature extraction may be too simplistic, or based on incorrect cognitive assumptions. Neural networks have been utilised in waveform classification, where traditional methods have proved inadequate *[Gevins and Morgan, 1988; Pao, 1989; Anderson and Rosenfeld (Eds.), 1989]*. Preliminary studies have investigated the use of neural networks as blink waveform identifiers in a hybrid neural network/ IOARS *[Patel, 1990]*. Neural networks were developed to act as

page 277

front end pattern recognition processors. The output of the neural network, for each channel, was converted to a symbolic feature for interface to the standard IOARS dynamic knowledge base. The final output node weight represented the fuzzy set membership. The hardware implementation of the IOARS will accommodate the use of neural network pattern recognition processors using the dedicated DSP chips or more specific hardware configurations.

7.3.3 LEARNING.

The present IOARS relies upon explicit coding of the experts knowledge as a series of productions, to form the knowledge base. When used as a post recording analysis tool, there would be a requirement for the system to learn, from the user, as new and as yet unidentified, waveforms emerge. The static knowledge base of this system would be dynamic and increase in size as exceptions to the rules are accommodated. This learning would ideally be transparent to the user, and require minimal programming skill on their part *[Michalski, et al., 1986; Forsyth and Rada, 1986]*. Learning would also be an integral part of the neural network pattern recognition processors described in section 7.3.2. Errors in blink waveform identification, would force the network into a new learning cycle to accommodate the wrongly identified waveform.

7.4 CONCLUSIONS.

This section concludes this thesis by stating concisely, the key achievements made during this investigation and the contribution that has been made to knowledge:

. Previous research has been extended by investigating various EOG subtraction models on new EEG signal derivations. This has resulted in establishing a simple EOG subtraction model for OA removal from both bipolar and referential EEG signal derivations.

. New techniques have been investigated that allow the real-time adaptive OA filter algorithm to be implemented in a non-continuous environment. This has enabled the algorithm to be succesfully applied, only to sections of EEG that contain OA corruption.

. Major OA types and abnormal frontal slow waveforms have been characterised by their spatio-temporal potential distribution. This has enabled a knowledge base of rules to be devised that: (i) allow the detection of OA, (ii) enable differentiation between OA and abnormal frontal slow waveforms, and (iii) reflect the analytical processes of the EEG expert.

. A novel intelligent EEG signal processing tool has been designed and implemented. The tool integrates EEG expert knowledge with OA dependent adaptive filters to enable 2-second segments of 16-channel EEG signals, containing OA contamination to be analysed. Segments identified as

containing OA are enhanced by removing OA in a selective and directed manner. This has significantly improved the quality and reliability of OA removal, and reduced further EEG corruption.

REFERENCES FOR CHAPTER 7.

Anderson, J.A, and Rosenfeld, E. (Eds.)

"Neurocomputing: foundations of research", MIT Press, Cam., Mass., 1989.

Baas, L., and Bourne, J.R.

*"A rule based microcomputer system for electroencephalogram evaluation",
IEEE Transations on Biomed. Eng., Vol BME-31, No. 10, pp. 660-664, 1984.*

Barlow, J.S.

*"Computerised clinical electroencephalography in perspective". IEEE
Tran. on Biomed. Eng., Vol.BME-26, No 7, pp. 377-391, July 1977.*

*Bourne, J.R., Hamel, B., Giese, D., Woyce, G.M., Lawrence, P.L.,
Ward,J.W. and Teschan,P.E .*

*"The EEG analysis system of the national cooperative dialysis study".
IEEE Tran. on Biomed. Eng., Vol.BME-27, No 11, pp. 656-664,November
1980a.*

*Bourne, J.R., Jagannathan, V., Hamel, B., Jansen, B.H., Ward,J.W.,
Hughes, J.R. and Erwin, C.W.*

*"Evaluation of a syntactic pattern recognition approach to
quantitative electroencephalographic analysis". Electroenceph. and
clin. neurophysiol., 1981, 52, pp. 57-64.*

Bourne, J.R., Matousek, M., Friberg, S., and Arvidsson, A.

*"SEER-1: The semantic EEG evaluation regimen", IEEE Tran. Biomed. Eng.,
Vol. BME-30, No. 4, April 1983.*

Cooper, R., Osselton, J.W. and Shaw, J.C.

"EEG Technology". 3rd ed., Butterworths, 1980.

Forsyth, R. and Rada, R.

*"Machine learning: Applications in expert systems and information
retrieval". Ellis Horwood series in Artificial Intelligence, 1986.*

Gaschnig, J., Klahr, P., Pople, H., Shortliffe, E., and Terry, A.

*"Evaluating expert systems: Issues and case studies", in "building expert
systems", Hayes-Roth, Waterman, and Lenat, (Eds.), Addison-Wesley,
Reading, Mass., 1983.*

page 281

Gevins, A.S., and Morgan, N.H.

"Application of neural-network (NN) signal processing in brain research",
IEEE Tran. Acoustics, Speech and Signal Proc., Vol. 36, No. 7, 1988.

Giese, D.A., Bourne, J.R. and Ward, J.W.

"Syntactic analysis of the electroencephalogram ". IEEE Tran. on
Syst. Man and Cybern., Vol. SMC-9, No. 8, August 1979.

Graham, I. and Jones, P. L.

"Expert systems: knowledge uncertainty and decision". Chapman and
Hall, 1988.

Hellyar, M.T., Ifeachor, E. C., Mapps, D.J., Allen, E.M. and Huson N.

"An expert system approach to EEG signal processing", In Press, 1991.

Ifeachor, E.C.

"Investigation of OA in the human EEG and their removal by a micro-
processor-based instrument, PhD thesis, Plymouth Polytechnic, England,
1984.

Ifeachor, E.C., Jervis, B.W., Allen, E.M. and Hudson, N.R.

"A new microcomputer-based online ocular artefact removal (OAR)
system". IEE proceedings, Vol. 133, Pt. A, No. 5, July 1986.

Ifeachor, E.C., Jervis, B.W., Allen, E.M, Morris, E.L., Wright, D.E.
and Hudson, N.R

"Investigation and comparison of some models for removing ocular
artefacts from EEG signals: Part 1 Review of models and data
analysis". Med. & Biol. Eng. & Comput., 26, pp 584-590, Nov. 1988.

Ifeachor, E.C., Jervis, B.W., Allen, E.M, Morris, E.L., Wright, D.E.
and Hudson, N.R

"Investigation and comparison of some models for removing ocular
artefacts from EEG signals: Part 2 quantitative and pictorial
comparison of models". Med. & Biol. Eng. & Comput., 26,
pp 584-590, Nov. 1988.

Ifeachor, E.C., Hellyar, M.T., Mapps, D.J., Allen, E.M.

"Intelligent enhancement of EEG signals". IEE colloqium on
"The application of artificial intelligence techniques to signal
processing", London, March, 1989.

Ifeachor, E.C., Hellyar, M.T., Mapps, D.J., Allen, E.M.

"Knowledge based enhancement of EEG signals".IEE Proceedings (Special
Edition), Vol. 137, Pt. F, No. 5, October, 1990.

Jagannathan, V., Bourne, J.R., Jansen, B.H. and Ward, J.W.

"Artificial intelligence methods in quantitative electroencephalogram
analysis". Comput. Programs in Biomed., Vol. 15, pp. 249-258, 1982.

Jansen, B.H., Bourne, J.R., Jagannathan, V. and Ward, J.W.

"Evaluation of slow-waves in the electroencephalogram using syntactic
shape analysis". Proceedings - Southeastcon., pp. 406-410, 1981a.

Jansen, B.H.

"Automatic interpretation of electroencephalograms by means of an
expert system". Optical Engineering, Vol. 24, No. 6, November/December
1985.

Jervis, B.W., Ifeachor, E.C., and Allen, E.M.

"The removal of ocular artefacts from the electroencephalogram:
a review". Med. & Biol. Eng. & Comput., 26, pp. 2-12, Jan. 1988.

Ktonas, P.Y.

"Automated analysis of abnormal electroencephalograms". CRC critical
reviews in biomedical engineering, Vol. 9, Issue 1, pp.39-97, 1983.

MacGillivray, B.

"The application of automated EEG analysis to the diagnosis of
epilepsy". In EEG Informatics, pp. 243-262, Ed. by Remond, A.
Amsterdam, Elsevier, 1977.

Michael, D. and Houchin, J.

"Automatic EEG analysis: A segmentation procedure based on the
autocorrelation function". Electroenceph. and clin. neurophysiol.,
1979, 46, pp. 232-235.

page 283

Michalski, R.S., Carbonall, J.G., and Mitchell, T.M. (Eds.)

"Machine learning: an artificial intelligence approach, Vol. 2", Morgan
Kaufmann Pub. Ltd., Los Altos, Cal., 1986.

Motorola.

"DSP96002 IEEE floating-point dual port processor user's manual",
Motorola Inc., 1989.

Pao, Y.

"Adaptive pattern recognition and neural networks", Addison-Wesley,
Reading, Mass., 1989.

Patel, S.

"A hybrid neural network/knowledge based approach to EEG analysis",
MSc Thesis, Polytechnic South West, Plymouth, UK, 1990.

Praetorius, H.M., Bodenstein, G. and Creutzfeldt, O.D.

"Adaptive segmentation of EEG records: A new approach to automatic EEG
analysis". Electroenceph. and clin. neurophysiol., 42, pp.
84-94, 1977.

Shortliffe, E.H.

"Computer based medical consultations: MYCIN". American, Elsevier. 1976.

Texas.

"

Tonn, B.E., and Goeltz, R.T

"Psychological validity of uncertainty combining rules in expert systems",
Expert systems, Vol. 7, No. 2, May 1990.

Winston, P. H.

"Artificial intelligence (second edition)". Addison-Wesley, Reading,
Mass., 1984.

LIST OF APPENDICES

1. THE ORDINARY LEAST SQUARES (OLS) METHOD.

Equation 1.5 estimates the true EEG (eeg(i)) from the measured EEG (y(i)), the EOG (EOG(i), and the OA coefficients (K) and is repeated here for convenience.

$$\hat{eeg}(i) = y(i) - K\, EOG(i) \quad i = 1,2,...m \qquad\qquad 1.5$$

where   K are estimates of K, the ocular artefact coefficients.
                 $\hat{eeg}(i)$ is the estimate of the true EEG
                 m is the number of samples used in the estimation.
                 $K = [k_1, k_2, ... k_n]$
                 $EOG^T(i) = [eog_1(i), eog_2(i), ... eog_n(i)]$
                 n is the number of eog signals used in the estimation.
                 T indicates transposition.

if eeg(i) represents an error term in the above equation, then a least squares estimate of K can be obtained by minimising the sum of squares of $\hat{eeg}(i)$ (E_{LS}), where:

$$E_{LS} = \sum_{i=1}^{m} \hat{eeg}^2(i)$$

$$= [\hat{eeg}(1), \hat{eeg}(2), ... \hat{eeg}(m)]^T [\hat{eeg}(1), \hat{eeg}(2), ... \hat{eeg}(m)] = \hat{E}_m^T \hat{E}_m \qquad A1$$

$$= (Y_m - \hat{K}_m EOG_m)^T (Y_m - \hat{K}_m EOG_m)$$

$$= Y_m^T Y_m - 2\hat{K}_m EOG_m^T Y_m + \hat{K}_m^T EOG_m^T \hat{K}_m \qquad A2$$

Differentiating A2 w.r.t $\hat{K}_m$ and equating to zero [Ifeachor, 1984]

$$\frac{\partial (E_{LS})}{\partial \hat{K}_m} = -2 EOG_m^T Y_m + 2 EOG_m^T EOG_m \hat{K}_m = 0 \qquad A3$$

Equation A3 can be simplified to:

$$\hat{K}_m = (EOG_m^T EOG_m)^{-1} EOG_m^T Y_m \qquad A4$$

Equation A4 gives the OLS estimate of the OA coefficients. From the estimates of the OA coefficients, an estimate of the true EEG is obtained from equation 1.5. Implicit in the OLS estimation is the assumption that the EOGs are not perfectly collinear. This would prevent the existence of the inverse matrix $(EOG_m^T EOG_m)$ and therefore the calculation of $\hat{K}_m$.

2. THE RECURSIVE LEAST SQUARES (RLS) METHOD.

The RLS algorithm allows an iterative estimation of $\hat{K}$ which will converge to an optimum set of values given a statistically stable input signal.

$$\hat{K}(i+1) = \hat{K}(i) + Gain(i+1) [y(i+1) - EOG(i+1)\hat{K}(i)] \qquad A5$$

$$= \hat{K}(i) + Gain(i+1) \, e\hat{e}g(i+1)$$

where $(i+1)$ indicates the new sample estimate
(i) indicates the previous sample.

An estimate of $\hat{K}$ is made at each sample point and the previous m samples contribute to the estimate. To reduce the effects of old data samples on the present estimate, previous error terms ($e\hat{e}g(i)$ $i = 1,2,...m$) are exponentially weighted:

$$WERR = \sum_{i=1}^{m} \gamma^{m-i} eeg^2(i) \qquad 0 < \gamma < 1 \qquad 4.6$$

This generates the following partitiioned matrices for Y_m and EOG_m respectively from A5:

$$Y_m = \begin{bmatrix} \gamma^{\frac{m-1}{2}} y(1) \\ \cdot \\ \cdot \\ \gamma^{\frac{1}{2}} y(m-1) \\ ---------- \\ y(m) \end{bmatrix} = \begin{bmatrix} \gamma^{\frac{1}{2}} Y_{m-1} \\ ---------- \\ y(m) \end{bmatrix} \qquad A6$$

$$EOG_m = \begin{bmatrix} \gamma^{\frac{m-1}{2}} EOG^T(1) \\ \cdot \\ \gamma^{\frac{1}{2}} EOG^T(m-1) \\ ---------- \\ EOG^T(m) \end{bmatrix} = \begin{bmatrix} \gamma^{\frac{1}{2}} EOG_{m-1} \\ ---------- \\ EOG^T(m) \end{bmatrix} \qquad A7$$

Equation A4 gave the OLS estimates of K as:

$$\hat{K}_m = (EOG_m{}^T EOG_m)^{-1} EOG_m{}^T Y_m \qquad A4$$

A matrix P_m is defined as:

$$P_m = (EOG_m{}^T EOG_m)^{-1} \qquad A8$$

which, using the definition of EOG_m in A7 above becomes:

$$P_m = \left([\gamma^{\frac{1}{2}} EOG_{m-1} \mid EOG^T(m)] \begin{bmatrix} \gamma^{\frac{1}{2}} EOG_{m-1} \\ ---------- \\ EOG^T(m) \end{bmatrix} \right)^{-1}$$

A 2

$$= [\gamma \, EOG^T_{m-1} EOG_{m-1} + EOG(m)EOG^T(m)]^{-1}$$

$$= [\gamma \, P^{-1}m-1 + EOG(m)EOG^T(m)]^{-1}$$

$$= \frac{1}{\gamma}[P_{m-1} \cdot G_m EOG^T(m)P_{m-1}] \qquad \qquad A9$$

where $G_m = P_m EOG(m)$

[Ifeachor, 1984]

From A4 $\hat{K}_m$ now becomes:

$$\hat{K}_m = P_m EOG_m{}^T Y_m \qquad\qquad A10$$

where

$$EOG_m{}^T Y_m = [\gamma \, EOG^T_{m-1} \; \vdots \; EOG(m)] \left[\begin{array}{c} \gamma^{\frac{1}{2}} \; Y_{m-1} \\ \hline y(m) \end{array} \right]$$

$$= \gamma EOG^T_{m-1} Y_{m-1} + EOG(m)y(m) \qquad\qquad A11$$

therefore:

$$\hat{K}_m = P_m[\gamma P^{-1}{}_{m-1} \hat{K}_{m-1} + EOG(m)y(m)] \qquad\qquad A12$$

$$= P_m[P^{-1}{}_m \cdot EOG(m)EOG^T(m)]\hat{K}_{m-1} + P_m EOG(m)y(m)$$

$$= \hat{K}_{m-1} + P_m EOG(m)[y(m) \cdot EOG^T_m \hat{K}_{m-1}] \qquad\qquad A13$$

Which is now in the form of equation A5, where Gain = $P_m EOG(m)$.

3. U-D FACTORISATION.

The error covarience matrix, P, is factored to avoid the numerical instability of the conventional RLS update formula A9. The main reason for this instability is that P is computed as a difference of positve semi-definite matrices [Bierman, 1976]. In the absence of any EOG, P will increase exponentially as it is being constanly divided by (less than 1).

P_{m+1} is factored as:

$$P_{m+1} = U_{m+1} D_{m+1} U^T_{m+1}$$

U_m is an upper triangular matrix with unit diagonal elements and D_m is a diagonal matrix. A9 is rewritten as:

$$P_{m+1} = \frac{1}{\gamma} U_m [D_m - \frac{1}{\alpha} v\, v^T] U^T_m \qquad\qquad A14$$

where $v = D_m U^T_m EOG_{m+1}$.

If the term in the square brackets is factored as:

$$\bar{U}_m \bar{D}_m \bar{U}^T_m = [D_m - \frac{1}{\alpha} v\, v^T]$$

where the bar is used to distinguish the U-D factors of $[D_m - 1\ v\ v^T]$ from those of P_{m+1}, then A14 becomes:

$$P_{m+1} = \frac{1}{\gamma} U_m \bar{U}_m \bar{D}_m \bar{U}^T_m U^T_m \qquad\qquad A15$$

noting that the product of two upper triangular matrices is itself an upper triangular matrix, and the symmetry in A15, then

$$U_{m+1} = U_m \bar{U}_m \; ; D_{m+1} = \frac{\bar{D}_m}{\gamma} \qquad\qquad A16$$

U_{m+1} and D_{m+1} are modified using the following algorithm: The algorithm is initiated by assigning starting values to both U and D. At each new sample the new values of EOG and EEG are used in the update algorithm to obtain a new P.

1. $v = U^T_m EOG$
2. $b_i = Di_m v \qquad\qquad i=2,3,....,n$
3. $\alpha_1 = \gamma + b_1 v_1$
4. $D1_{m+1} = D1_m / \alpha_1$
5. $b1 = D1_{m+1} v1$

for j = 2,3,....,n recursively evaluate 6 to 10.

6. $\alpha_j = \alpha_{j-1} + vj bj$
7. $P_j = -v_j / \alpha_{j-1}$

for k = 1,2,....,j-1 recursively evaluate 8 and 9.

8. $Ukj_{m+1} = Ukj_m + bk P_j$
9. $bk = bk + bj\, Ukj_m$

10. $Dj_{m+1} = Dj_m \alpha_{j-1} / (\alpha_j \gamma)$
11. $G_{m+1} = bi / \alpha_n$

A 4

APPENDIX B.

The following diagrams represent the combined circuit diagram for the
Data Acquisition System (DAS), detailed in chapter 3. The diagrams form the
implementation of the system conceptualised in figure 3.1. Each circuit
represents one block in figure 3.1, with the exception of the
instumentation amplifier circuit and ADC circuit which constitutes the
analogue signal conditioning block in figure 3.1.

Bl Instumentation amplifier circuit diagram.

0.22µF

+VS

18k   18k

Input

+

CA3140E

-VS

-

0.22µF

0.22µF

25k   1k   +VS

LF398

output

-VS

shold

0.01µF polypropylene

digital gnd

analogue gnd
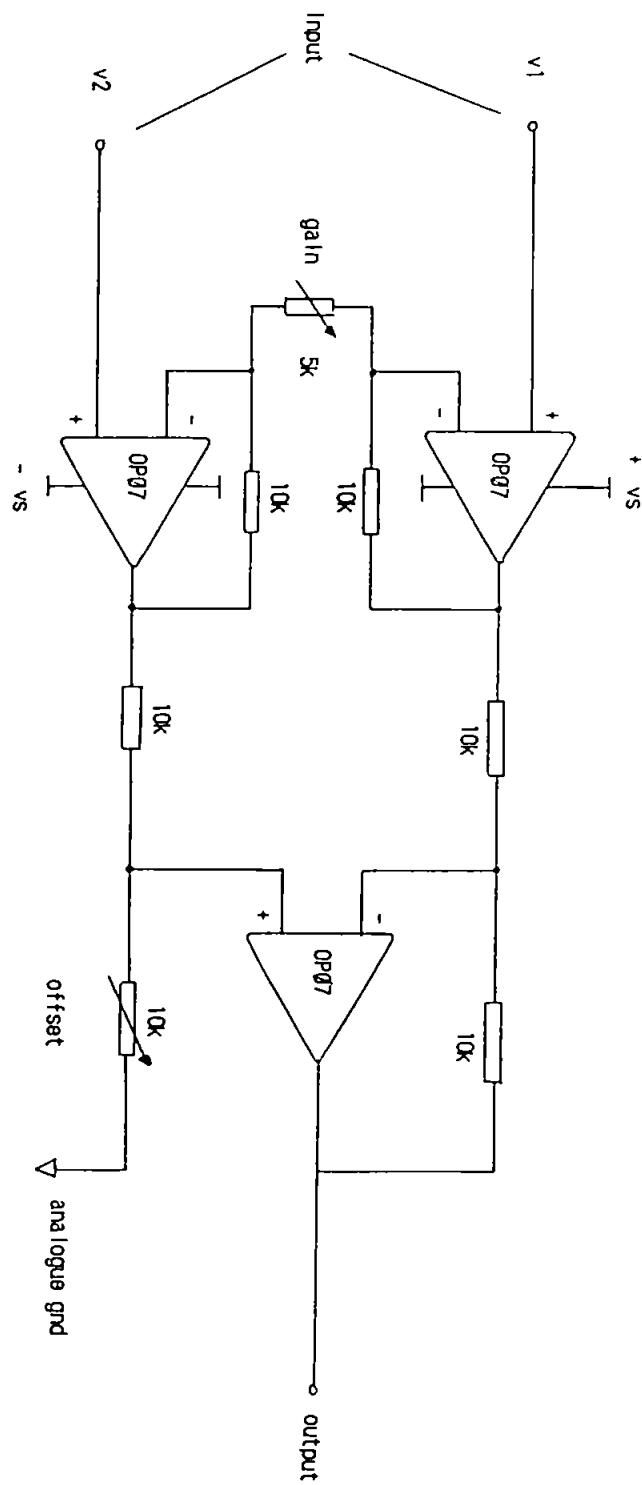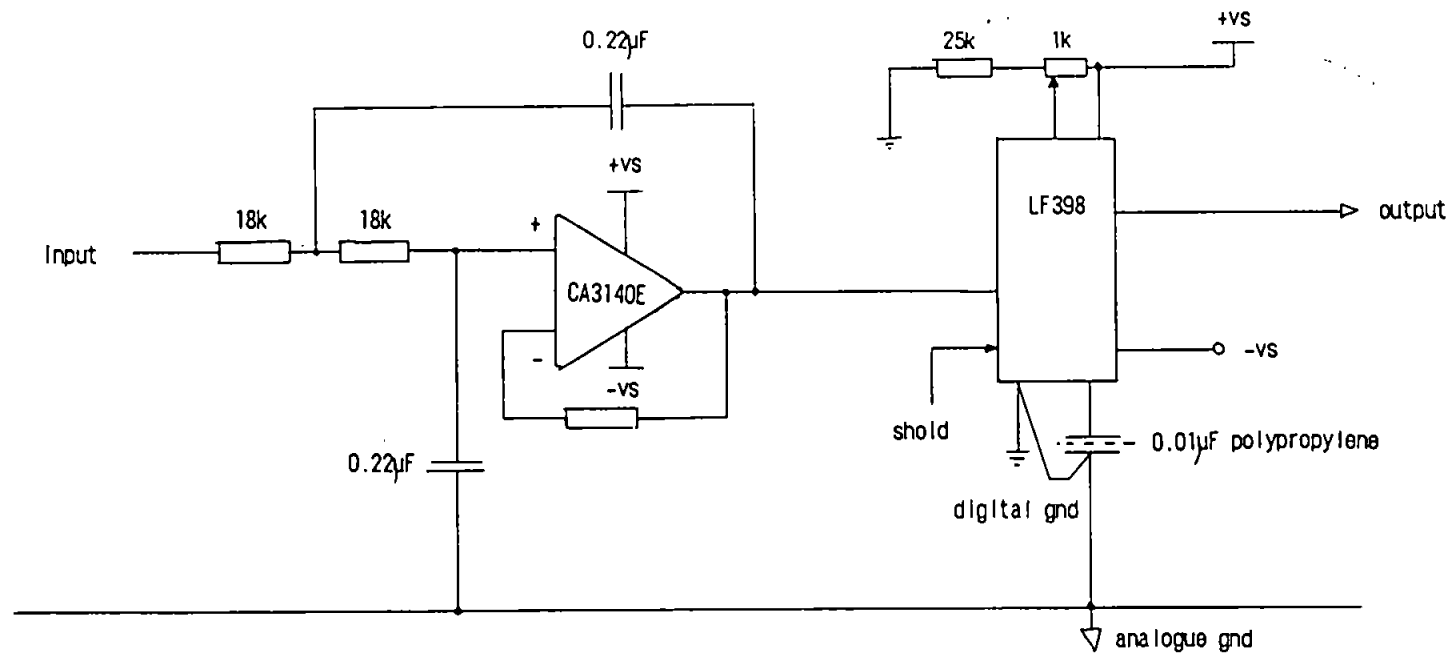
2nd order butterworth filter, gain = 1, fc = 40Hz

sample and hold

B2 Anti-aliasing filter and sample and hold amplifier circuit diagram

82
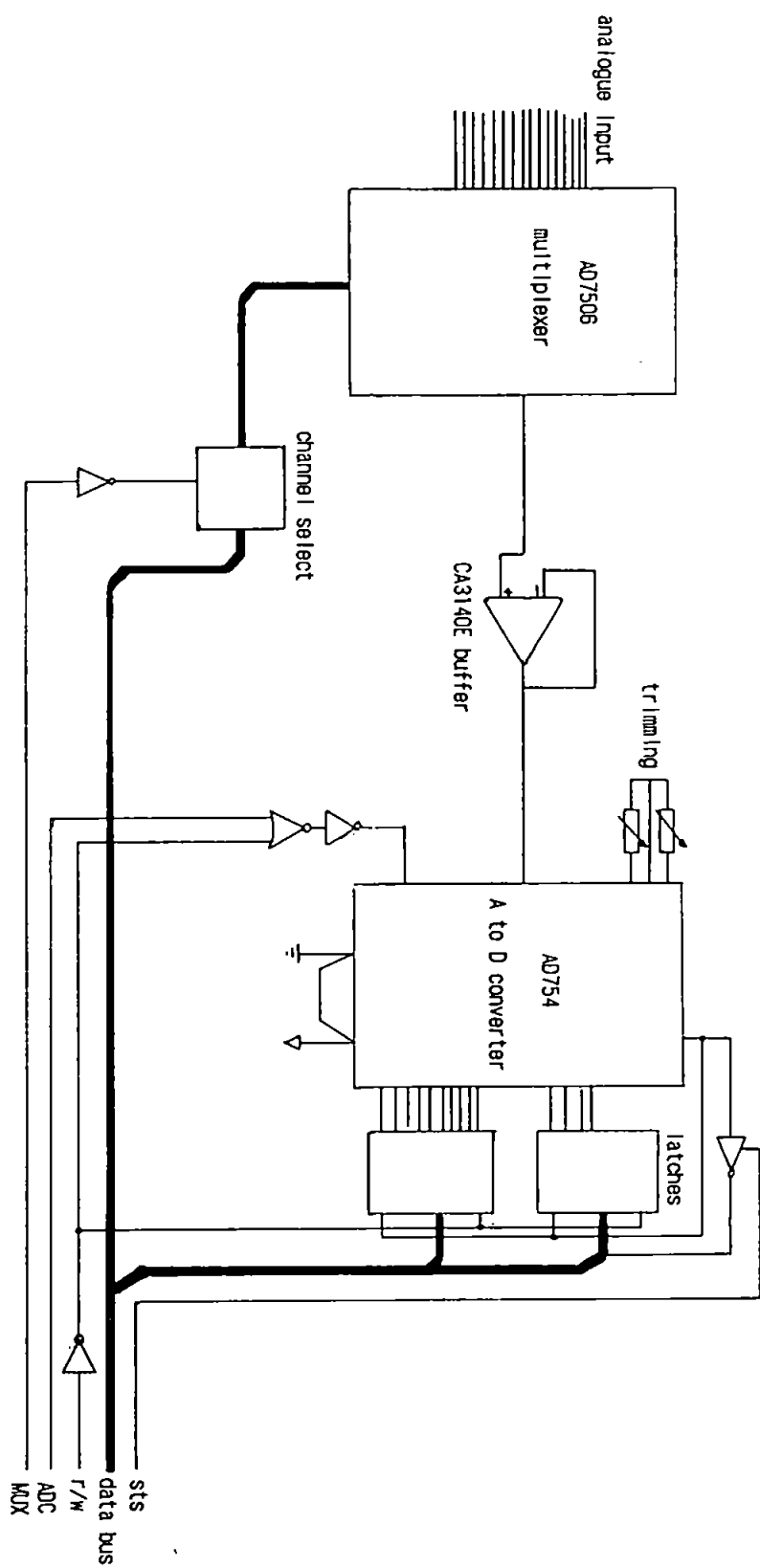
B3 Multiplexer and A to D converter circuit diagram.

307
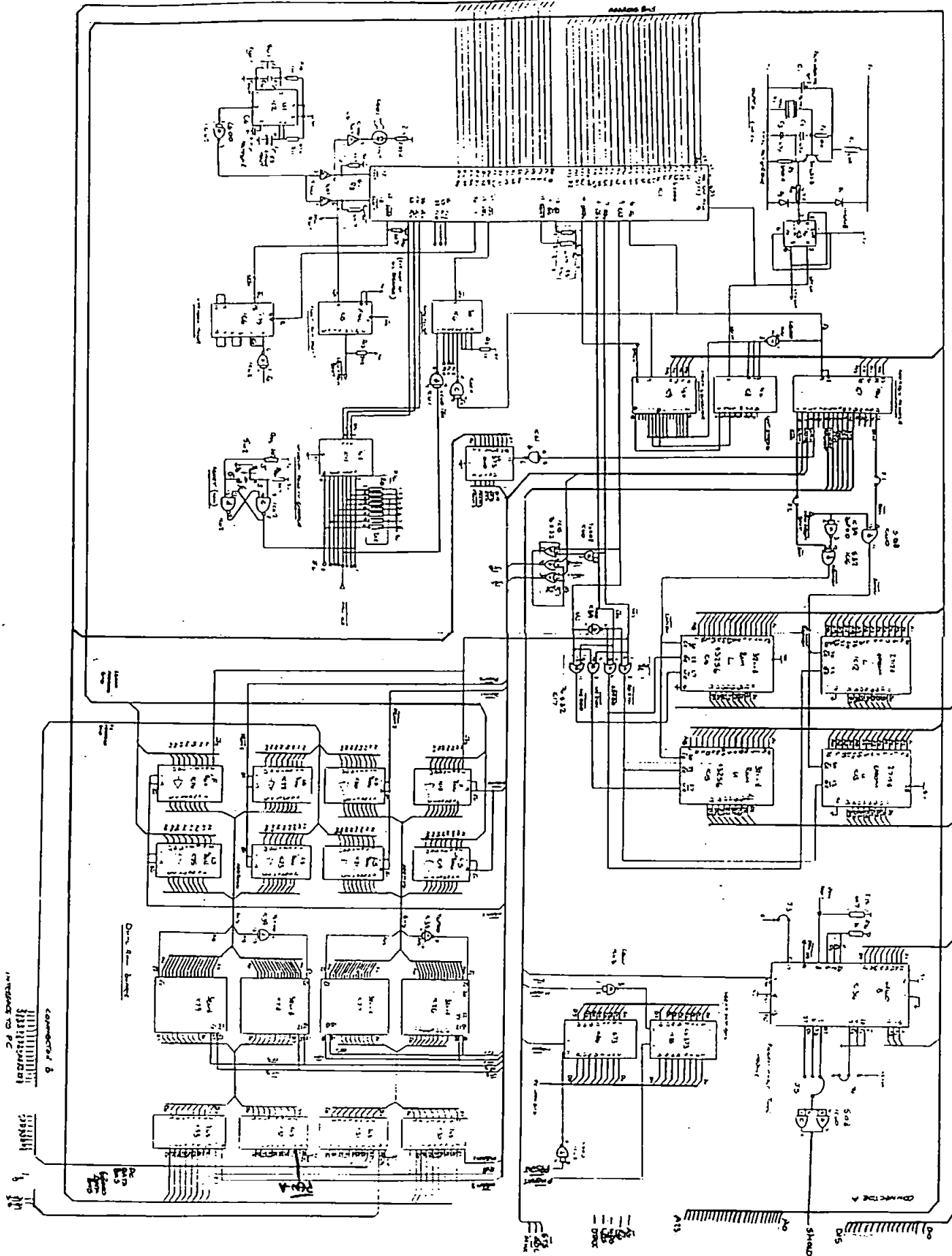
B4 Microprocessor control circuit diagram.

84

B5 Personal computer interface circuit diagram.

85

LS138

LS138

LS323

LS323

LS245

74LS00

74LS00

74LS521

GND

5V 100nf 47µ TANT

ACN

A9
A5
A8
A7
A6

G2B
G1
G2A
A
B
C

A2
A1
A0

G2A
G2B
G1
C
B
A

D7
D6
D5
D4
D3
D2
D1
D0
AEN

A9
A8
A7
A6
A5
A4
A3
A2
A1
A0

a

b

c

d

AL1
AL2

G
D
Q
OUT

PENA
PENB
PHSTAT
PPSTAT

A15
A14
A13
A12
A11
A10
A9
A8

A7
A6
A5
A4
A3
A2
A1
A0

D7

D0

A
B

D7
D6
D5
B4
D3
D2
D1
D0

3M FLAT IDC

A0
A2
A4
A6
A8
A10
A12
A14
PENA
PENB
PHSTAT
PPSTAT
D0
D2
D4
D6

A1
A3
A5
A7
A9
A11
A13
A15
5V
GND
D1
D3
D5
D7

5V GND 100nf 47µ TANT

IBM PC/LS3000 DIS INTERFACE CARD

APPENDIX C.

The following software listings represent the DAS software and data editing suite. These programs were used in conjunction with the DAS hardware to acquire EEG data for this thesis. The Software is comprised of:

.DAS system software listing.
.PC control sofware listing.
.Editing software listing.

The DAS system software listing is written in 68000 assembler and was developed using the METAI multi assembler system. The software provides the basic interrupt handling within the DAS microprocessor board. At each interrupt each of the 16 signal channels are sampled and selectively converted.

The PC Control software provides control of the system via the PC interface circuit and a buffered RAM area (DRB). Control is supplied to the user via a menu system displayed on the PC screen. Data is archived to winchester disk and later transferred to backup tape streamer. The software has been written in Basic using the MICROSOFT QUICK BASIC software package.

The editing sotware provides the neccesary tools for reviewing and selecting sections of EEG data for further analysis. Edited sections are transferred via ASCII file to allow easy access to the ILS signal processing software package. The software has been written in Basic using the MICROSOFT QUICK BASIC software package.

```
        .ctrl   27,15

        title   "68000 data acquisition system"

; This program is designed to test the 68000 D.A.S. as a complete system

        segment byte at F00000 'boot'
```

;--

```
; The first eight bytes of ROM are accessed at reset by appearing to be
; addresses 000000 to 000007 respectively, these hold the supervisor stack
; pointer and the program counter (pointing to ROM)

        dc.b    $00     ;SSP
        dc.b    $00
        dc.b    $10

        dc.b    $00
        dc.b    $00     ;PC
        dc.b    $F0
        dc.b    $00
        dc.b    $08
```

;--

```
;labels used throughout the code

adc     equ     $A00000
avect4  equ     $70
drb     equ     $2000
drba    equ     $100000
drbb    equ     $200000
drbcon  equ     $300000
mmstat  equ     $700000
mpstat  equ     $600000
mux     equ     $900000
pause   equ     $2010
ptm0    equ     $800000
ptm1    equ     $800002
ptm2    equ     $800004
ptm4    equ     $800008
ptm5    equ     $80000A
ramadr  equ     $2004
stat    equ     $800000
```

;--

```
;initialisation

init    move.b  #$57,mmstat     ;tell pc that data logger is reset
        move.l  #int,avect4     ;load autovector 4 with interrupt routine address
        move.b  #$0F,drbcon     ;select ORB_A for writing to.
        move.l  #drba,drb       ;ram register
        clr.l   ramadr          ;clear ram displacement register
        move.l  #$1000,pause    ;clear pause flag
        move.b  #$C7,ptm1       ;CR2
        move.b  #$81,ptm2       ;M = 129
        move.b  #$17,ptm5       ;L = 23
        move.b  #$01,ptm0       ;reset interrupts.
        move.b  #$00,ptm0       ;start PTM timer.
```

```
; (M+1)(L+1) 1.25uS = 3.9 mS which is the sampling period

;--------------------------------------------------------------------------

; the next routines wait for the PC to request recording to start until
; enabling the PTM to cause interrupts.

start   clr.l   D0              ;
        move.w  mpstat,D0       ;read the status of the PC
        and.w   #$00FF,D0       ;
        cmp.w   #$00AA,D0       ;has the PC requested that interrupts start?
        bne     start           ;if not, keep looking.

stint   move    #$2300,SR       ;supervisor mode and interrupt mask level 3

;--------------------------------------------------------------------------

; As soon as the PC has read a buffer it tells the 68000. the next routine
; then stops the PC from reading it again.

wait    clr.l   D0              ;
        move.w  mpstat,D0       ;read the status of the PC
        and.w   #$00FF,D0       ;
        cmp.w   #$FF,D0         ;has the PC requested that interrupts stop?
        beq.s   stop            ;if so, stop interrupts.
        bra.s   wait            ;wait for a stop or finished reading command.

stop    move.w  #$0000,pause    ;set the pause flag.
        move.b  #$50,mmstat     ;tell the PC not to read the buffer again
        move    #$2700,SR       ;supervisor mode and interrupt mask level 7
        bra.s   start           ;wait for a start command.




;--------------------------------------------------------------------------

; Interrupt service routine
; This routine is enterred whenever the processor recieves a level 4
; interrupt, meaning that the PTM has timed out.

int     movem.l D0-D7/A0-A6,-(A7)        ;save registers
        move.w  #$00,mux                 ;select channel 0
        bsr     status                   ;
        move.w  adc,D0                   ;dummy read

        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                 ;store lower 8 bits
        lsr.w   #8,D0                    ;shift right 8 bits
        and.w   #$1F,D0                  ;mask out insignificant bits
        move.b  D0,(A0)+                 ;store upper 4 bits
        addq.l  #2,ramadr                ;increment ram diplacement


        move.w  #1,mux
        bsr     status
        move.w  adc,D0          .
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
```

```
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement

        move.w  #2,mux
        bsr     status
        move.w  adc,D0
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement

        move.w  #3,mux
        bsr     status
        move.w  adc,D0
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement

        move.w  #4,mux
        bsr     status
        move.w  adc,D0
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement

        move.w  #5,mux
        bsr     status
        move.w  adc,D0
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement

        move.w  #6,mux
```

```
        bsr     status
        move.w  adc,D0
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement

        move.w  #7,mux
        bsr     status
        move.w  adc,D0
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement

        move.w  #8,mux
        bsr     status
        move.w  adc,D0
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement

        move.w  #9,mux
        bsr     status
        move.w  adc,D0
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement

        move.w  #10,mux
        bsr     status
        move.w  adc,D0
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
```

```
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement

        move.w  #11,mux
        bsr     status
        move.w  adc,D0
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement

        move.w  #12,mux
        bsr     status
        move.w  adc,D0
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement

        move.w  #13,mux
        bsr     status
        move.w  adc,D0
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement

        move.w  #14,mux
        bsr     status
        move.w  adc,D0
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement

        move.w  #15,mux
        bsr     status
        move.w  adc,D0
```

C5

```
        bsr     status
        move.w  adc,D0
        add.w   pause,D0
        move.l  drb,A0
        add.l   ramadr,A0
        move.b  D0,(A0)+                ;store lower 8 bits
        lsr.w   #8,D0                   ;shift right 8 bits
        and.w   #$1F,D0                 ;mask out insignificant bits
        move.b  D0,(A0)+                ;store upper 4 bits
        addq.l  #2,ramadr               ;increment ram diplacement


;       move.w  #1,mux

;       bsr     status
;       move.w  adc,D0


;       clr.l   D1                      ;
;       move.b  #2,D1                   ;set channel register
;
;read   move.w  D1,mux                  ;connect next channel
;       bsr     status                  ;
;       move.w  adc,D0                  ;read last channel
;
;       move.l  drb,A0                  ;read which buffer
;       add.l   ramadr,A0               ;add displacement
;
;       move.b  D0,(A0)+                ;store lower 8 bits
;       lsr.w   #8,D0                   ;shift right 8 bits
;       and.w   #$0F,D0                 ;mask out insignificant bits
;       move.b  D0,(A0)+                ;store upper 4 bits
;       addq.b  #1,D1                   ;increment channel register
;       addq.l  #2,ramadr               ;increment ram diplacement
;       cmp.b   #16,D1                  ;all channels converted ?
;       bmi.s   read                    ;if not read next channel
;       bsr     status                  ;
;       move.w  adc,D0                  ;read channel 16
;       move.l  drb,A0                  ;read which buffer
;       add.l   ramadr,A0               ;add displacement
;       move.b  D0,(A0)+                ;store lower 8 bits channel 16
;       lsr.w   #8,D0                   ;shift right 8 bits
;       and.w   #$0F,D0                 ;
;       move.b  D0,(A0)+                ;store upper 4 bits channel 16
;       addq.l  #2,ramadr               ;increment ram diplacement


        move.w  #$1000,pause            ;clear pause flag
        cmp.l   #$FFDF,ramadr           ;is ram full ?
                                        ;note: only 2047*32 bytes is stored
                                        ;in RAM. This is equal to 7.996
                                        ;seconds of data.
        bmi.s   endint                  ;if not, wait for next interrupt
;
;if the buffer is full, then the various control registers must be set
;so that on the next interrupt the other DRB is written to.
;The PC must also be told which DRB it should read from.
;
        clr.l   ramadr                  ;clear ram displacement
;
        move.l  drb,D4                  ;which buffer ?
        move.l  drb,D1                  ;
        eor.l   #$300000,D4             ;swap buffer
        move.l  D4,drb                  ;save ram register
```

C6

```
;
        move.b  #20,D2              ;find out which buffer is
        lsr.l   D2,D1               ;ready to be read.
;
        and.l   #3,D1               ;
        add.l   #$54,D1             ;
;
        cmp.b   #$56,D1             ;
        beq.s   brdy                ;set up other buffer to be
                                    ;written to.
ardy    move.b  #$0A,drbcon         ;
        bra.s   pcrdy               ;
;
brdy    move.b  #$05,drbcon         ;
;
pcrdy   move.b  D1,mmstat           ;tell PC which buffer to read.
;
;
;
endint  move.b  ptm1,D0             ;read PIM status
        move.b  ptm4,D0             ;read PIM counter
        movem.l (A7)+,D0-D7/A0-A6   ;restore registers

        rte

;....................................................................
;
; The following routine ensures that the ADC is not read before it has
; completed conversion of the channel.

delay   clr.l   D3
        move.b  #6,D3
delay1  dbra    D3,delay1
        rts

status  move.w  stat,D7
        move.w  D7,$2008
        and.w   #01,D7
        cmp.w   #01,D7
        beq.s   status
        rts

        end
```

C7

C8

C8

```
DECLARE SUB axis1 (ch%)
DECLARE SUB browse1 (bufseg%)
DECLARE SUB display1 (toread%, bufseg%, ch%)


        'PROGRAM PCDSP12.BAS


        'Program to handle the reading of data from the
        '68000 D.A.S and the storage of this data to the
        'P.C winchester hard disc drive.

        '(c) M.I.Hellyar August 1988 : All rights reserved.




COMMON SHARED stored%()
COMMON prstat, ppstat, block%
COMMON toread%, rdy%, hd%, b2%, c2
COMMON buf%()

DECLARE SUB check (toread%, lstrd%, red)
DECLARE SUB warning ()
DECLARE SUB collect (rdy%, block%, hd%, bufseg%)
DECLARE SUB scskel (title$)
DECLARE SUB contin ()
DECLARE SUB head (hd%)
DECLARE SUB status ()
DECLARE SUB stat ()
DECLARE SUB record (bufseg%, fullup)
DECLARE SUB menu ()
DECLARE SUB pause (key$)
DECLARE SUB full (key$)
DECLARE SUB datin (bufseg%)
DECLARE SUB display (toread%, bufseg%, chan%)
DECLARE SUB axis (chan%)
DECLARE SUB browse (bufseg%, chan%)
DECLARE SUB transfer (stored%())
DECLARE SUB win2disc (patient!, totalblks!)
DECLARE SUB newdisc ()


DIM stored%(1 TO 20)
REM $DYNAMIC
DIM buf%(0 TO 32760)
bufseg% = VARSEG(buf%(0))
bufptr% = VARPTR(buf%(0))



        DEF SEG

labels:

        prstat = 773
        rst = 0
        block% = 0
        toread% = 0
        hd% = 0
        rdy% = 0
        AL1 = 768
        AL2 = 769
        PENA = 771
        PENB = 772
        CONST false = 0, true = NOT false
```

```
REM Read 68000 status, has it been reset.
        COLOR 4, 0, 6
        CLS
        WHILE rst <> 84
                rst = INP(pmstat) AND 252
                LOCATE 10, 16
                COLOR 3, 0, 6
                PRINT "Please press the reset button on the data logger"
        WEND
        COLOR 7, 0, 6
        CLS


REM main program loop

        DO
                cont = false
                fullup = false
                CALL menu
                DO
                        LOCATE 23, 24: COLOR 3, 0, 6: PRINT "please select your choice :";
                         choice$ = INPUT$(1)
                LOOP UNTIL INSTR("1234a", choice$) <> 0
                cont = true
                IF choice$ = "1" THEN CALL status: GOTO again
                IF choice$ = "2" THEN CALL datin(bufseg%): GOTO again
                IF choice$ = "3" THEN CALL record(bufseg%, fullup): GOTO again
                IF choice$ = "4" THEN cont = false
                IF choice$ = "a" THEN CALL transfer(stored%()): GOTO again

again:  LOOP WHILE cont AND fullup = false

        IF fullup THEN CALL warning


        END

REM $STATIC
SUB axis (chan%)
DRAW "bm 0,0"
DRAW "m 0,200"
DRAW "bm 0,25"
DRAW "m 512,25"
DRAW "bm 0,75"
DRAW "m 512,75"
DRAW "bm 0,125"
DRAW "m 512,125"
DRAW "bm 0,175"
DRAW "m 512,175"
FOR i = 0 TO 512 STEP 64
        FOR j = 0 TO 3
                z = j * 50 + 23
                w = z + 5
                DRAW "bm=" + VARPTR$(i) + ",=" + VARPTR$(z)
                DRAW "m=" + VARPTR$(i) + ",=" + VARPTR$(w)
        NEXT j
NEXT i
label% = ((chan% - 1) * 4)
LOCATE 4, 70: PRINT "ch-"; label% + 1
LOCATE 10, 70: PRINT "ch-"; label% + 2
LOCATE 16, 70: PRINT "ch-"; label% + 3
LOCATE 22, 70: PRINT "ch-"; label% + 4
```

```basic
LOCATE 1, 65: PRINT "Press any key"
LOCATE 2, 65: PRINT "to continue."
LOCATE 24, 1: PRINT "0";
LOCATE 24, 33: PRINT "1";
LOCATE 24, 65: PRINT "2";
LOCATE 24, 70: PRINT "time(secs)";
END SUB


SUB axis1 (ch%)
DRAW "bm 0,0"
DRAW "m 0,200"
DRAW "bm 0,100"
DRAW "m 512,100"
FOR i = 0 TO 512 STEP 64
        z = 98: w = 102
        DRAW "bm=" + VARPTR$(i) + ",=" + VARPTR$(z)
        DRAW "m=" + VARPTR$(i) + ",=" + VARPTR$(w)
NEXT i
LOCATE 13, 70: PRINT "ch-"; ch%
LOCATE 1, 65: PRINT "Press any key"
LOCATE 2, 65: PRINT "to continue."
LOCATE 24, 1: PRINT "0";
LOCATE 24, 33: PRINT "1";
LOCATE 24, 65: PRINT "2";
LOCATE 24, 70: PRINT "time(secs)";

END SUB


SUB browse (bufseg%, chan%)
        ppstat = 774
        OUT ppstat, 170
        SCREEN 2
        LOCATE 12, 27: PRINT "Loading data buffer.."
        DO
                red = false
                toread% = INP(773) AND 3
                IF toread% = 1 OR toread% = 2 THEN CALL check(toread%, lstrd%, red)
                IF red = true THEN CALL display(toread%, bufseg%, chan%)
        LOOP WHILE INKEYS = ""
        OUT ppstat, 255
        SCREEN 0
END SUB


SUB browse1 (bufseg%)
        PRINT "Which channel do you wish to display ";
        INPUT ch%
        ppstat = 774
        OUT ppstat, 170
        SCREEN 2
        LOCATE 12, 27: PRINT "Loading data buffer.."
        DO
                red = false
                toread% = INP(773) AND 3
                IF toread% = 1 OR toread% = 2 THEN CALL check(toread%, lstrd%, red)
                IF red = true THEN CALL display1(toread%, bufseg%, ch%)
        LOOP WHILE INKEYS = ""
        OUT ppstat, 255
        SCREEN 0

END SUB


SUB check (toread%, lstrd%, red) STATIC
IF toread% = lstrd% THEN red = false ELSE red = true
IF red = true THEN lstrd% = toread%
```

C 11

```
END SUB

SUB collect (rdy%, block%, hd%, bufseg%) STATIC
block% = block% + 1
CALL drb(bufseg%, rdy%)
DEF SEG = bufseg%
path$ = "c:\markw\qb4\data\d"
b$ = STR$(block%)
c = LEN(b$) - 1
b2$ = STR$(hd%)
c2 = LEN(b2$) - 1
file$ = path$ + RIGHT$(b2$, c2) + RIGHT$(b$, c) + ".dat"
BSAVE file$, 0, 65504
DEF SEG
stored%(hd%) = block%
LOCATE 23, 9: COLOR 3, 0, 6
PRINT "Data from buffer "; rdy%; " has been saved as d"; RIGHT$(b2$, c2); RIGHT$(b$, c); ".dat, to winchester."
END SUB


SUB contin
LOCATE 23, 27: COLOR 3, 0, 6
PRINT "Press any key to continue": PRINT INPUT$(1)
END SUB


SUB datin (bufseg%)
        CALL scskel("CHECK INPUT SIGNALS")
        COLOR 5, 0, 6
        LOCATE 7, 23: PRINT "THIS OPTION DOES NOT RECORD DATA !"
        LOCATE 10, 10: PRINT "To ensure that data is being recieved by the data acquisition"
        LOCATE 12, 10: PRINT "system, you have the option of displaying any 4 channels of"
        LOCATE 14, 10: PRINT "the 16 data channels available."

        CALL contin
        DO
                CALL scskel("CHECK INPUT SIGNALS")
                COLOR 5, 0, 6
                LOCATE 6, 30: PRINT "Your options are :"
                LOCATE 9, 20: PRINT "1.  Select 1 to view channels 1 to 4"
                LOCATE 11, 20: PRINT "2.  Select 2 to view channels 5 to 8"
                LOCATE 13, 20: PRINT "3.  Select 3 to view channels 9 to 12"
                LOCATE 15, 20: PRINT "4.  Select 4 to view channels 13 to 16"
                LOCATE 17, 20: PRINT "5.  finish."
                DO
                        LOCATE 23, 24: COLOR 3, 0, 6: PRINT "please select your choice :";
                        choice$ = INPUT$(1)
                LOOP UNTIL INSTR("12345a", choice$) <> 0
                chan% = VAL(choice$)
                IF chan% > 0 AND chan% < 5 THEN CALL browse(bufseg%, chan%)
                IF chan% = 0 THEN CALL browse1(bufseg%)
                LOOP UNTIL chan% = 5
END SUB


SUB display (toread%, bufseg%, chan%) STATIC
        CLS
        CALL axis(chan%)
        CALL drb(bufseg%, toread%)
        DEF SEG = bufseg%
        x = (chan% - 1) * 8
        FOR j = 0 TO 511
                d = j * 32 + x
                FOR i = 0 TO 3
                        datpoint1% = PEEK(d + (i * 2)): datpoint2% = PEEK((d + (i * 2)) + 1)
                        CALL add(datpoint1%, datpoint2%)
```

```
                                    datpoint% = datpoint1%
                                    IF datpoint% > 4095 THEN datpoint% = datpoint% - 4096
                                    datpoint% = datpoint% / 82
                                    ' datpoint% = 50 - datpoint%
                                    datpoint% = datpoint% + (i * 50)
                                    PSET (j, datpoint%)
                            NEXT i
                    NEXT j
                    DEF SEG
        END SUB




SUB display1 (toread%, bufseg%, ch%)
        CLS
        CALL axis1(ch%)
        CALL drb(bufseg%, toread%)
        DEF SEG = bufseg%
        x = (ch% - 1) * 2
        DRAW "bm 0,100"
        FOR j = 0 TO 510
                d = j * 32 + x
                datpoint1% = PEEK(d): datpoint2% = PEEK(d + 1)
                CALL add(datpoint1%, datpoint2%)
                datpoint% = datpoint1%
                IF datpoint% > 4095 THEN datpoint% = datpoint% - 4096
                datpoint% = datpoint% / 21
                ' datpoint% = 200 - datpoint%
                'PSET (j, datpoint%)
                DRAW "m=" + VARPTR$(j) + ",=" + VARPTR$(datpoint%)
                NEXT j
        DEF SEG

END SUB

SUB full (key$) STATIC
OUT ppstat, 255
LOCATE 10, 19: PRINT "   DATA MEMORY IS FULL. PRESS ANY KEY.        "
PRINT INPUT$(1)
key$ = "n"
END SUB

SUB head (hd%) STATIC
CALL scskel(" PATIENT DATA")
LOCATE 8, 14: COLOR 5, 0, 6
PRINT "To enable easier identification of data, please enter"
LOCATE 10, 14
PRINT "the patients initials, age and sex. Special features"
LOCATE 12, 14
PRINT "of the patient, such as 'signs of frontal slowing' etc, "
LOCATE 14, 14
PRINT "could also be entered under additional information."
LOCATE 16, 14
PRINT "This information will be kept confidential. Thank you. "
CALL contin
CALL scskel(" SAVE PATIENT DATA")
hd% = hd% + 1
LOCATE 8, 14: COLOR 5, 0, 6
INPUT "Please enter the patients initials :"; init$
LOCATE 10, 14
INPUT "Please enter the patients age :"; age$
LOCATE 12, 14
INPUT "Please enter the patients sex :"; sex$
```

C.13

```
LOCATE 14, 14
INPUT "Additional information :"; ai$
path2$ = "c:\marku\qb4\data\hd"
b2$ = STR$(hd%)
c2 = LEN(b2$) - 1
file2$ = path2$ + RIGHT$(b2$, c2) + ".dat"
OPEN file2$ FOR OUTPUT AS #3
PRINT #3, init$
PRINT #3, &HFF
PRINT #3, age$
PRINT #3, &HFF
PRINT #3, sex$
PRINT #3, &HFF
PRINT #3, ai$
PRINT #3, &HFF
CLOSE #3
PRINT a$, b$, c$, d$
CALL contin
END SUB


SUB menu STATIC
CALL scskel("      M E N U")
PRINT CHR$(196)
LOCATE 7, 13
COLOR 5, 0, 6
PRINT "This program controls the storage of EEG data to a PC."
LOCATE 9, 13
PRINT "The following options are available to the user."
LOCATE 14, 15
COLOR 7, 0, 6
PRINT "1.  Select 1 to display the recording status."
LOCATE 16, 15
PRINT "2.  Select 2 to check the incoming signals."
LOCATE 18, 15
PRINT "3.  Select 3 to record EEG data."
END SUB


SUB newdisc
        CALL scskel("  TRANSFER DATA")
        LOCATE 10, 10: COLOR 5, 0, 6: PRINT "Insert a blank disc in drive A and press any key to continue."
        PRINT INPUT$(1)
        CALL scskel("  TRANSFER DATA")
        LOCATE 8, 1
END SUB


SUB pause (key$) STATIC
CALL scskel("DATA RECORDING PAUSE")
LOCATE 12, 22: COLOR 3, 0, 6
PRINT "Do you wish to continue recording ? "
LOCATE 14, 10
PRINT "Press 'SPACE BAR' to release the pause or any other key to stop.";
r$ = INPUT$(1)
IF r$ = " " THEN key$ = "y" ELSE key$ = "n"
IF key$ = "y" THEN CALL scskel("  RECORDING DATA"): LOCATE 10, 19: COLOR 5, 0, 6: PRINT " To pause data recording press SP
IF key$ = "y" THEN OUT 774, 170
END SUB


SUB record (bufseg%, fullup) STATIC
CONST false = 0, true = NOT false
key$ = "y"
ppstat = 774
block% = 0
CALL space(dx%, ax%, 3)
IF dx% < 0 THEN dx# = dx% + 65536 ELSE dx# = dx%
```

C14

```
IF ax% < 0 THEN ax# = ax% + 65536 ELSE ax# = ax%
room% = (dx# * 65536 + ax#) / 65536
CALL head(hd%)
CALL scskel(" RECORD DATA")
COLOR 5, 0, 6
LOCATE 10, 10: PRINT "Before recording data, please check that the cable connecting"
LOCATE 12, 10: PRINT "the data logger to the EEG machine is properly inserted"
LOCATE 14, 10: PRINT "at both ends. Thank you."
CALL contin
CALL scskel(" RECORD DATA")
COLOR 5, 0, 6
LOCATE 10, 10: PRINT "At the end of each test please pause recording so that only "
LOCATE 12, 10: PRINT "the relevent test data is recorded."
CALL contin
CALL scskel(" RECORDING DATA")
LOCATE 10, 19: COLOR 5, 0, 6: PRINT " To pause data recording press SPACE BAR"
TIME$ = "00:00:00"
OUT ppstat, 170

DO
        red = false
        LOCATE 18, 22: COLOR 7, 0, 6
        PRINT "Elapsed time is "; TIME$; " hr:min:sec"
        toread% = INP(773)
        toread% = toread% AND 3
        IF toread% = 1 OR toread% = 2 THEN CALL check(toread%, lstrd%, red)
        IF red = true THEN CALL collect(toread%, block%, hd%, bufseg%)
        IF INKEY$ = " " THEN OUT ppstat, 255: CALL pause(key$)
        IF block% = room% THEN CALL full(key$)
LOOP WHILE key$ = "y"
OUT ppstat, 255
IF block% = room% THEN fullup = true
END SUB

SUB scskel (title$)
CLS              .
LOCATE 3, 30
COLOR 6, 0, 6
PRINT title$
COLOR 6, 0, 6
LOCATE 4, 10
FOR i = 1 TO 60: PRINT CHR$(205); : NEXT i
LOCATE 20, 10
FOR i = 1 TO 60: PRINT CHR$(205); : NEXT i
LOCATE 21, 64: PRINT "mth 88"
END SUB

SUB status STATIC
CALL scskel(" RECORDING STATUS")
dx% = 0: ax% = 0
CALL space(dx%, ax%, 3)
IF dx% < 0 THEN dx# = dx% + 65536 ELSE dx# = dx%
IF ax% < 0 THEN ax# = ax% + 65536 ELSE ax# = ax%
REM dx$ = HEX$(dx%): ax$ = HEX$(ax%): PRINT dx$, ax$: END

LOCATE 10, 20: COLOR 5, 0, 6
IF dx# = 65535 THEN
        PRINT "DISC ERROR"
        ELSE PRINT "you have "; dx# * 65536 + ax#; " bytes left on disc"
END IF
emp = dx# * 65536 + ax#
timlft = emp / 8192
min% = timlft / 60
secs% = timlft - (min% * 60)
```

C/S

```
IF secsX < 0 THEN secsX = secsX + 60: minX = minX - 1
IF minX > 30 THEN minX = 30: secX = 0
LOCATE 14, 15:
PRINT "This is "; minX; " minutes and "; secsX; " seconds  of recording"
LOCATE 16, 15
PRINT "time available, out of a maximum of 30 minutes."
CALL contin
END SUB


SUB transfer (storedX())
        CALL scskel("  TRANSFER DATA ")
        LOCATE 9, 10
        storeddata = false
        FOR i = 1 TO 20
                totalblks = storedX(i)
                IF totalblks <> 0 THEN PRINT "patient "; i; ",";
                IF totalblks <> 0 THEN list$ = list$ + STR$(i)
                IF totalblks > 0 THEN storeddata = true
        NEXT i
        IF storeddata = false THEN
                LOCATE 12, 25: COLOR 5, 0, 6: PRINT "There is no data to transfer.          "
                CALL contin
                ELSE LOCATE 6, 20: COLOR 5, 0, 6: PRINT "Data exists for the following patients...."
                LOCATE 14, 10: PRINT "To transfer patient data to floppy disc, please enter the"
                LOCATE 16, 10: PRINT "patient numbers as a list, i.e  13  wil copy patients 1 "
                LOCATE 18, 10: PRINT "and 3's data but not patient 2's."
                LOCATE 23, 30: INPUT copy$
        END IF
        IF storeddata THEN CALL newdisc
        IF storeddata THEN
                FOR j = 1 TO 20
                        totalblks = storedX(j)
                        pat$ = STR$(j)
                        patlen = LEN(pat$) - 1
                        patient$ = RIGHT$(pat$, patlen)
                        IF INSTR(copy$, patient$) <> 0 AND INSTR(list$, patient$) THEN CALL win2disc(j, totalblks)
                NEXT j
        END IF
END SUB


SUB warning STATIC
CALL scskel("  DATA BUFFER FULL")
LOCATE 8, 16: COLOR 5, 0, 6
PRINT " You have filled the memory available to you."
LOCATE 10, 16
PRINT "To record more data you must restart the system."
LOCATE 14, 33: COLOR 7, 0, 6
PRINT "W A R N I N G   "
LOCATE 17, 13
PRINT "This will destroy the data you have already collected."
END SUB


SUB win2disc (patient, totalblks) STATIC
        pat$ = STR$(patient)
        patlen = LEN(pat$) - 1
        patient$ = RIGHT$(pat$, patlen)
        SHELL "copy c:\marku\qb4\data\hd" + patient$ + ".dat a:"
        SHELL "del c:\marku\qb4\data\hd" + patient$ + ".dat a:"
        FOR i = 1 TO totalblks
                index$ = STR$(i)
                blklen = LEN(index$) - 1
                tranfile$ = RIGHT$(index$, blklen)
                00
                        CALL space(dxX, axX, 1)
```

```
                              IF dx% < 0 THEN dx# = dx% + 65536 ELSE dx# = dx%
                              IF ax% < 0 THEN ax# = ax% + 65536 ELSE ax# = ax%
                              IF dx# = 65535 THEN PRINT "DISC ERROR"
                              emp = dx# * 65536 + ax#
                              IF emp > 65511 THEN
                                      SHELL "copy c:\marku\qb4\data\d" + patient$ + tranfile$ + ".dat a:"
                                      SHELL "del C:\marku\qb4\data\d" + patient$ + tranfile$ + ".dat"
                                      ELSE CALL newdisc
                              END IF
                      LOOP UNTIL emp > 65511
              NEXT i
              stored%(patient) = 0
      END SUB
```

C18

```
'PROGRAM VIEW2.BAS

'Program to handle the reading of stored data
'and displaying it one channel at a time
'to the screen


'(c) M.T.Hellyar August 1988 : All rights reserved.




COMMON pmstat, ppstat, block%
COMMON toread%, rdy%, hd%, b2S, c2
COMMON buf%(), cont
COMMON SHARED mont$()

DECLARE SUB axis (chan%)
DECLARE SUB plotdata (chan%, wind)
DECLARE SUB command (hd%, block%, chan%, wind, cont, hiblk%)
DECLARE SUB readdata (hd%, block%)
DECLARE SUB labelgraph (hd%, block%, chan%, wind)
DECLARE SUB edit (hd%, block%, chan%, wind!)
DECLARE SUB trans (hd%, block%, chan%, wind!)
DECLARE SUB break (x)

DIM mont$(1 TO 16)
REM $DYNAMIC
DIM buf%(0 TO 32760)
bufseg% = VARSEG(buf%(0))
bufptr% = VARPTR(buf%(0))




        DEF SEG = bufseg%

        CONST false = 0, true = NOT false

        ON ERROR GOTO errsub

REM main program loop

setup:  chan% = 1
        block% = 1
        wind = 1
        cont = true
        mont$(1) = "I2"
        mont$(2) = "FP2"
        mont$(3) = "F4"
        mont$(4) = "C4"
        mont$(5) = "T6"
        mont$(6) = "I1"
        mont$(7) = "FP1"
        mont$(8) = "F3"
        mont$(9) = "C3"
        mont$(10) = "T5"
        mont$(11) = "F8"
        mont$(12) = "N"
        mont$(13) = "F7"
```

C19

```
                mont$(14) = "FZ"
                mont$(15) = "CZ"
                mont$(16) = "PZ"

                SCREEN 2

                LOCATE 1, 1: INPUT "last block number "; hiblk%
                INPUT "Patient "; hd%
main:   DO
                        CLS
                        IF block% <> lastblock% OR hd% <> lasthd% THEN CALL readdata(hd%, block%)
                        lastblock% = block%: lasthd% = hd%
                        CALL axis(chan%)
                        CALL labelgraph(hd%, block%, chan%, wind)
                        CALL plotdata(chan%, wind)
                        CALL command(hd%, block%, chan%, wind, cont, hiblk%)
                LOOP WHILE cont

END 'end of main program


errsub:
        LOCATE 1, 1: PRINT "Insert correct disc and press any key to continue": PRINT INPUT$(1): RESUME main

REM $STATIC
SUB axis (chan%)
        DRAW "bm 0,0"
        DRAW "m 0,200"
        DRAW "bm 0,100"
        DRAW "m 512,100"
        FOR i = 0 TO 512 STEP 64
                z = 98: w = 102
                DRAW "bm=" + VARPTR$(i) + ",=" + VARPTR$(z)
                DRAW "m=" + VARPTR$(i) + ",=" + VARPTR$(w)
        NEXT i

        LOCATE 13, 70: PRINT mont$(chan%)
END SUB

SUB break (x)
DRAW "bm=" + VARPTR$(x) + ",0"
DRAW "m=" + VARPTR$(x) + ",200"
END SUB

SUB command (hd%, block%, chan%, wind, cont, hiblk%) STATIC
        DO
                choice$ = INPUT$(1)
        LOOP UNTIL INSTR("12567789zx", choice$) <> 0
        IF choice$ = "1" THEN LOCATE 1, 1: INPUT "channel"; chan: chan% = chan
        IF choice$ = "2" THEN CALL edit(hd%, block%, chan%, wind)
        IF choice$ = "5" THEN block% = block% + 1: IF block% > hiblk% THEN block% = hiblk%
        IF choice$ = "6" THEN LOCATE 1, 1: INPUT "patient "; head: hd% = head
        IF choice$ = "7" THEN block% = 1
        IF choice$ = "8" THEN LOCATE 1, 1: INPUT "block "; blk: block% = blk
        IF choice$ = "9" THEN cont = false
        IF choice$ = "z" THEN wind = wind - 1: IF wind < 1 THEN wind = 4: block% = block% - 1: IF block% < 1 THEN block% =
        IF choice$ = "x" THEN wind = wind + 1: IF wind > 4 THEN wind = 1: block% = block% + 1: IF block% > hiblk% THEN blo
END SUB

SUB curs (x%)
        DRAW "bm=" + VARPTR$(x%) + ",0"
        DRAW "m=" + VARPTR$(x%) + ",200"
END SUB
```

```
SUB edit (hd%, block%, chan%, wind)
LOCATE 1, 1: INPUT "save this block for analysis (y/n)"; anal$
IF anal$ = "y" OR anal$ = "Y" THEN CALL trans(hd%, block%, chan%, wind) ELSE LOCATE 1, 1: PRINT "
END SUB


SUB labelgraph (hd%, block%, chan%, wind)
        LOCATE 1, 70: PRINT "No "; hd%
        LOCATE 2, 70: PRINT "Block "; block%
        lo% = ((wind - 1) * 2) + ((block% - 1) * 8)
        hi% = (wind * 2) + ((block% - 1) * 8)
        lo$ = STR$(lo%)
        hi$ = STR$(hi%)
        c1 = LEN(lo$) - 1
        c2 = LEN(hi$) - 1
        LOCATE 24, 1: PRINT RIGHT$(lo$, c1);
        LOCATE 24, 65: PRINT RIGHT$(hi$, c2);
        LOCATE 24, 71: PRINT "time(secs)";

END SUB


SUB plotdata (chan%, wind) STATIC
        x = (chan% - 1) * 2
        DRAW "bm 0,100"
        FOR j = 0 TO 510
                d = (((wind - 1) * 512) * 32) + (j * 32) + x
                datpoint% = PEEK(d) + 256 * PEEK(d + 1)
                IF datpoint% > 4095 THEN CALL break(j)
                IF datpoint% > 4095 THEN datpoint% = datpoint% - 4096
                datpoint% = datpoint% / 21
                datpoint% = 200 - datpoint%
                DRAW "m=" + VARPTR$(j) + ",=" + VARPTR$(datpoint%)

        NEXT j

END SUB


SUB readdata (hd%, block%) STATIC
        path$ = "a:\d"
        b$ = STR$(block%)
        c = LEN(b$) - 1
        b2$ = STR$(hd%)
        c2 = LEN(b2$) - 1
        file$ = path$ + RIGHT$(b2$, c2) + RIGHT$(b$, c) + ".dat"
        BLOAD file$, 0
END SUB


SUB relabel
CLS
c = 1
DO
        PRINT "Enter mnemonic for channel "; c;
        INPUT a$
        IF a$ = "" THEN a$ = mont$(c)
        mont$(c) = a$
        c = c + 1
LOOP UNTIL c = 17

END SUB


SUB trans (hd%, block%, chan%, wind) STATIC
        path$ = "c:\mdata\ed"
        FOR x = 0 TO 30 STEP 2   'save 16 channels
                number% = number% + 1
                num$ = STR$(number%)
```

```
            length% = LEN(num$) - 1
            index$ = RIGHT$(num$, length%) + ".dat"
            filename$ = path$ + index$
            OPEN filename$ FOR OUTPUT AS #3
'insert to save 1 channel        x = (chan% - 1) * 2
            FOR j = 0 TO 510
                    d = (((wind - 1) * 512) * 32) + (j * 32) + x
                    datpoint% = PEEK(d) + 256 * PEEK(d + 1)
                    datpoint% = datpoint% - 2048
                    PRINT #3, datpoint%
            NEXT j
            CLOSE #3
            '       include the following lines to save a header file for each edit file.
            '       path$ = "c:\mdata\eh"
            '       filename$ = path$ + index$
            '       OPEN filename$ FOR OUTPUT AS #3
            '       PRINT #3, hd%
            '       PRINT #3, block%
            '       PRINT #3, wind
            '       PRINT #3, chan%
            '       CLOSE #3
    NEXT x
END SUB
```

C22

APPENDIX D

The following appendix contains an abrieviated technicians report for a selection of the data acquired using the DAS. This illustrates the type of data recorded during this investigation on which the rules developed for the IOARS were elicited. Inclusion of this appendix will enable subsequent researchers to understand the basis of any rules developed.

This file contains information regarding patients data held on tape for use with the EVEREX tape streamer system. Data is held on several cassete type data cartridges. Each patients data in held in a seperate dataset and are sequentially numberred. Each dataset contains a file HD*.dat which also identifies each dataset when it is reloaded onto winchester disk.

The following information describes the data presented in this file.

dataset number   patient identifier
tape number
data directory (ie c:\markw\qb4\data?)
header data file number (ie c:\markw\qb4\hd?.dat)

patients initials
patients age
patents sex
date of data recording

1.	V-1
tape	1
data dir	-
hd	1
initials	S.H
age	32
sex	F

    This data provided information regarding eye movements from a series of controlled experiments (see notes for details).

2.	ab-1
tape	1
data dir	-
hd	1
initials	N.H
age	70
sex	M
date	16-2-89

    Theta (4-6 Hz) post central right (posterior). 2.5-4 Hz max posterioral temporal and parietal, more on the left. Rolling eye movements.

```
3.              ab-2
tape            1
data dir        -
hd              1

initials        C.C
age             -
sex             F
date            -
```

Drowsy.

```
4.              ab-3
tape            1
data dir        -                                 MAX BLOCK  =   104
hd              1

initials        M.R
age             8 months
sex             M
date            23-2-89
```

Irregular slow activity 3-5 Hz 5uV left centro-temporal.

```
5.              ab-4
tape            1
data dir        -
hd              1                                 MAX BLOCK =  104

initials        E.H
age             53
sex             M
date            24-2-89
```

5-7 Hz post central, 10-40uV symetrical. Ill- defined theta/delta 3-5 Hz 4uV bilateral, maximal posterior. Some sharpish elements. Frontal theta.

```
6.              ab-5
tape            1/2
data dir        -                                 MAX BLOCK =  112
hd              1

initials        S.D
age             54
sex             F
date            2-3-89
```

Theta elements with sharp components, both temporal regions. Slow activity from hyper-ventilation bilaterally. bilaterally synchronous 2-4 Hz 100uV.

```
10.          ab-9
tape         2
data dir     -
hd           I
```

MAX BLOCK = 81

```
initials     T.O
age          I
sex          M
date         25-7-89
```

Post central rhythm 4-5 Hz, 30-100uV, symetrical and increases on eye closure. Intermittant low voltage Beta 15-25 Hz. Frequent slow waves 2-3 Hz, bilateral. Slow activity increases in sleep, bilaterally. Vertex sharp waves. Sigma 13-14 Hz. K complexes

```
montage  block
1        1
2        7
3        18
4        31
5
6
7
8        54
```

pause at 68

```
11.          ab-10
tape         2
data dir     2
hd           I
```

MAX BLOCK = 30

```
initials     A.V
age          21
sex          M
date         13-9-89
```

Short clinical attack 3-4 minutes.
4-8 Hz, 10-50uV fainter on the left. Low voltage beta up to 30Hz, diffuse, symetrical. Lamda 20-30Hz symetrical.
Bilaterally synchronous short duration bursts, up to 3 secs, 2-3.5 Hz, 150uV. Sharp and slow waves 2-3Hz 200uV, mainly anterior and on the left.
Infrequent 2-3.5Hz delta, generalised bilaterally.
2 mins of hyper ventilation evoked a generalised slowing of the background to 3-6Hz. Generalised increase in slow and sharp waves and miximal anteriorly. During the attack, theta 5-8Hz, 10-30uV generalised bilateral, followed by 6secs of flattening.

Block numbers were not recorded on the paper trace and the technician experienced frequent problems withe the EEG machines paper feeder. As a result data is likely to be fragmented and difficult to identify montages.
Montages   1,2,3,4,7,8   were   used.

```
7.            ab-6
tape          2
data dir      -
hd            1                          max block = 71

initials      N.T
age           85
sex           M
date          7-3-89
```

Alpha 9-10Hz, beta 15-25Hz bilateral, theta and delta elements 3-7 Hz temporal (right centro-temporal).

```
8.            ab-7
tape          2
data dir      -
hd            1                          MAX block = 49

initials      P.R
age           33
sex           M
date          16-3-89
```

Slow waves 2-3.5 Hz, 50-150uV bilaterally more right than left, 1-5 sec bursts. Delta 1.5-3.5 Hz, upto 150 uV, both mid-temporal areas, more right than left. Generalised slow waves.

```
9.            ab-8
tape          2
data dir      -
hd            1                          max block = 67

initials      S.M.T
age           58
sex           M
date          7-4-89
```

Bilaterally synchronous delta 2-3 Hz, upto 50uV in 1-4 sec bursts maximal anteriorally and temporally.

12.   ab-11
tape   2
data dir   4
hd   1

initials   C.G
age   27
sex   F
date   14-9-89

13.   ab-11
tape   2
data dir   4
hd   1   MAX BLOCK = 97

initials   C.G
age   27
sex   F
date   14-9-89

     Alpha 11-12 Hz and 50uV, symetrical. Intermittant low voltage beta
15-25 Hz bilateral. Theta 4-7 Hz frequently recorded bilaterally, mainly
frontal and temporal, more on the left.
Occasional delta, 3-3.5 Hz mainly left temporal, both anterior and
posterior.
When patient was drowsy there was an attenuation of the alpha which was
replaced by low voltage beta and theta.
Hyper-ventilation increased theta and delta. Rare sharp elements are also
recorded frontally.

montage block
1   1
2   43
3   53
4   64
5
6   76
7
8   86

H.V during blocks 19-34

```
14.          ab-12
tape         2
data dir     4
hd           2                              MAX BLOCK = 72
```

```
initials     D.H
age          83
sex          F
date         14-9-89
```

Drowsy during recording.

post central rythmn 7-8 Hz, 20-40 uV, symatrical. Intermittant low voltage beta 15-20 Hz

Widespread slow 3-5 Hz, bilateral, usually independantly, occasional higher voltage slow activity, occurs in bursts of between 1 and 2 seconds.

As patient becomes drowsy there is an attenuation of the background rhythmn and there are several bursts of bilaterally synchronous slow waves,maximal left and of 4 second duration.

montage	block
1	1
2	14
3	26
4	38
5	
6	66...may not be recorded
7	
8	62

```
15.          ab-13
tape         2
data dir     6
hd           1                              MAX BLOCK = 103
```

```
initials     R.B
age          20
sex          M
date         15-9-89
```

patient slept lightly.

low voltage beta 15-25 Hz dominant and symmetrical. rare alpha 10-11 20uV, bilateral. Occasional alpha 4-7 Hz, mainly frontal.

frequent low voltage sharp waves 40uV central parietal, left and occuring in repetitive runs.

Onset of drowsiness increases theta and sharp waves. During light sleep vertex sharp waves are recorded.

Hyper ventilation increases theta.

montage	block	
1	1	
2	42	ECG is measured at the wrist between blocks 47 and 53
3	53	
4	69	
5		
6	81	
7		
8	90 .	

103	photic	stimulation

```
16.          ab-14
tape         2/3
data dir     8
hd           1          -              MA r BLOCK = 102

initials     R.W
age          17
sex          M
date         21-9-89
```

ALpha 8-9 Hz, 20-50 uV ,symmetrical is measured but attenuated with eyes open. intermittant low voltage beta, 15-25 HZ, bilateral. Alpha 4-7 Hz diffuse bilaterally.
Frequent delta 2-3 Hz, predominant post temporal, both synchronously and independently, slightly more predominant on the right.
2 bursts of bilaterally synchronous spike and wave 2-4Hz, 300uV, 3 seconds (eyelid flickering occured and the patient momentarily opened his eyes).
Hyper-ventilation increased slow activity and sharp wave activity. Prolonged runs of 3Hz slow waves, posterior temporal but gradually spreading to dominate. several short bursts 1-2 seconds of spike and wave.

```
montage  block
1        1        hyper-ventilation during blocks 14-34
2        53        FP2 rejelled at block 56
3        65        FPI rejelled later at block 58
4        75
5
6
7
8        86

         102 photic stimulation
```

```
17.          ab-15
tape         3
data dir     8
hd           2              MM BLOCK - 45

initials     G.N
age          29
sex          M
date         21-9-89
```

Fairly regular post-central rhythmn 6-7 Hz, 30-80 uV, symmetrical and attenuated by eyes opening. Intermittant low voltage beta 15-25 Hz
Slow waves 3-5 Hz recorded bilaterally mainly temporal and in particual centro and posterio-temporal
NO hyper-ventilation

```
montage  block
1        1
2        6
3        15
4        27
5
6        36
7
8        37

         44 photic stimulation
```

APPENDIX E

This appendix contains a tabulation of the OA parameters obtained from a number of referential electrode positions. The average of these parameters is used to enable the OA removal algorithm to be pre-set to an appropriate value for a specific electrode postion on identification of an OA.

	Fz		Cz		Pz		F4		C4		T6		F3		C3		T5	
	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
	·625	·579	·5787	1·3	0·5128	·7184	·4984	·612	·748	-·1352	·672	-·0672	·486	-·136	·18	-·021	·087	-·028
	·5167	·291	·5597	1·22	·2713	·6215	·637	-·19	·205	-·026	·176	-·065	·39	·062	·134	·055	·136	·124
	·5555	·812	·354	·5239	·2746	·5152	·632	-·630	·503	-·490	·321	-·424	·536	·252	·376	-·189	·330	-·179
	·5027	-·241	·2028	-·077	·682	-·063												
	·3526	·0839	·2267	·2479	·1348	·2133												
	·632	-·076	·563	-·02	·1235	·022												
	·675	-·261	·503	-·224	·370	-·194												
Averages																		
	0·537	0·51	0·37	0·678	0·306	·335	0·51	0·34	0·301	0·217	0·16	0·177	0·463	0·157	0·24	0·88	0·184	0·11

	Fz		Cz		Pz		F4		C4		T6		F3		C3		T5	
	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
	1·185	-1·331	·967	-·134	·335	-·72	·065	·339	-·136	·93	-·176	·132	·271	-·136	·259	·031	-·072	·238
	3·06	-2·64	2·33	-2·27	1·23	-1·18	·236	·58	·172	-·008	·048	·0057	·436	-·163	·208	-·29	·293	-·233
	·44	·264	·21	·005	·106	-·031	·232	·333	·372	·119	·022	·173	·383	·101	·135	·0206	·221	-·49
	·163	·238	-·188	-·223	·204	·188												
	·3508	·1135	·2505	-·054	·308	-·044												
	·322	-·02	·127	-·09	·41	-·113												
	·279	·265	·127	·0397	·062	·035												
Averages																		
	0·82	0·63	0·59	0·58	0·35	0·33	0·177	0·27	0·126	0·11	0·08	0·06	0·37	0·13	0·13	0·06	0·19	0·06

E 1

HCM																	
Fz		Cz		Pz		F4		C4		T6		F3		C3		T5	
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
-.376	-.5052	-.435	-.6245	-.259	-.328	-.032	.298	-.108	.011	-.104	.047	.752	.327	.036	-.066	-.009	-.028
-.652	-.3953	-.233	-.624	-.262	-.217	.099	.057	-.162	-.204	-.215	-.160	.450	-.163	.006	.224	.207	-.132
.135	.269	.031	.132	.463	.113	.285	.296	.199	.096	.106	.133	.381	-.05	.186	-.071	.200	-.156
.1083	.1604	-.090	-.082	-.073	-.057												
.4463	.3504	.3388	.2549	.2293	.1290												
.0AS	-.063	.096	.260	-.017	-.150												
.282	.099	.A1	-.014	.257	.062												
AVERAGES																	
0.263	0.272	.24	0.255	.16	.147	0.45	.217	0.49	0.311	0.46	0.11	0.29	0.08	0.09	0.361	0.138	0.124

APPENDIX F

This appendix contains the full tabulated Percentage Error Of Fit (PEOF) results obtained during off-line analysis. The results are shown for four EOG electrode montages (2,4,5,9), nine referential electrode sites, and nine bipolar electrode sites. These results have enabled a suitable and simple EOG electrode montage to be determined for OA removal from both referential and bipolar EEG signals.

PATIENT DATA 1

Percentage error of fit		Model 2 $eeg_m = a_1 I_e + a_2 EP_2 + a_3 I_1 + a_4 FP_1 + eeg_T$				Model 9 $eeg_m = a_1 FP_2 + a_2 F8 + a_3 FP_1 + a_4 F7 + eeg_T$				Model 4 $eeg_m = a_1(FP_2 - I_e) + a_2(FP_1 - I_1) + a_3(F8-N) + a_4(F7-N) + eeg_T$				Model 5 $eeg_m = a_1(FP_2 - F8) + a_2(FP_1 - F7) + eeg_T$			
		VEM	HEM	BLINK	N-A	VEM	HEM	BLINK	N-A	VEM	HEM	BLINK	N-A	VEM	HEM	BLINK	N-A
UNIPOLAR Ref to linked ears	Fz	1.07	5.0	1.2	1.19	1.99	0.6	0.3	0.4	1.17	6.59	8.63	3.25	4.03	6.36	39.59	7.21
	Cz	2.41	4.04	1.44	2.39	4.67	1.45	0.64	1.37	2.12	6.15	8.58	3.27	6.65	4.66	28.5	4.69
	Pz	5.4	6.43	2.64	5.26	10.76	4.08	1.97	3.54	4.58	8.45	10.10	5.81	13.16	6.96	24.34	7.81
BIPOLAR	P4-P4			0.83													
	F4-C4	4.17	5.53	12.27	38.83	2.11	3.55	7.97	37.92	8.5	4.12	11.8	41.06	9.71	5.57	18.18	78.5
	FP2-F4	1.18	3.11	0.83	7.23	8.21	2.62	0.31	4.10	3.45	6.26	2.33	5.11	3.07	8.5	7.46	13.34
	FP2-F8	0.6	7.28	5.49	7.98	0.0	0.0	0.0	0.0	0.67	2.13	10.0	8.51	0.0	0.0	0.0	0.0

EEG

10 - 11 - 88

10 - 11 - 88   PATIENT DATA 2

PERCENTAGE ERROR OF FIT		MODEL 2 EEGm = a₁ I₁ + a₂ FP2 - a₃... + a₄ FP1 + EEGT				MODEL 9 EEGm = a₁ FP2 + a₂ F8 + a₃ FP₁ + a₄ F7 + EEGT				MODEL 4 EEGm = a₁(FP2 - I₂) + a₂(FP₁ - I₁) + a₃(F₄ - N) - a₄(F7 - N) + EEGT				MODEL 5 EEGm = a₁(FP2 - F8) + a₂(FP₁ - F7) + EEGT			
		VEM	HEM	BLINK	N-A	VEM	HEM	BLINK	N-A	VEM	HEM	BLINK	N-A	VEM	HEM	BLINK	N-A
UNIPOLAR Ref to linked ears	Ft	0.8	2.42	0.75	0.69	0.87	0.48	0.44	0.64	1.33	4.95	1.57	1.91	2.36	28.22	10.82	2.31
	Cz	2.47	4.12	1.78	1.5	2.91	1.4	1.25	1.61	2.50	5.36	1.91	2.89	3.66	20.54	8.23	2.87
	Pz	6.95	7.02	4.84	2.88	8.55	4.15	4.3	2.89	6.91	8.18	4.05	4.5	8.83	28.18	11.15	4.8
BIPOLAR	F4-C4	4.7	4.98	14.85	79.11	5.72	4.71	14.33	78.04	11.69	6.28	13.34	81.23	14.95	21.08	25.17	94.2
	FP2-F4	0.96	5.26	1.0	5.18	1.19	1.76	2.76	5.07	3.17	3.96	1.04	6.6	3.81	14.15	2.4	5.9
	FP2-F8	0.44	2.74	0.69	1.21	0.0	0.0	0.0	0.0	0.35	1.0	1.0	1.88	0.0	0.0	0.0	0.0

EEG

F2

16-11-88 PATIENT DATA 2 with mean removed

F3

PERCENTAGE ERROR OF FIT		MODEL 2 EEGM = a₁L₁ ...				MODEL 9 EEGM = a₁ Fp₂ ... Fe				MODEL 4 EEGM = a₁(Fp₂-Te) ...				MODEL 5 EEGM = a₁(Fp₂-Fe) ...			
		YEM	HEM	BLINK	N-A	YEM	HEM	BLINK	N-A	YEM	HEM	BLINK	N-A	YEM	HEM	BLINK	N-A
UNIPOLAR Ref to UNUED CHAS	Fe	1.63	13.61	6.4	45.45	2.38	13.85	5.28	46.80	1.75	2.07	11.86	95.93	2.75	22.02	12.61	98.40
	Ce	2.16	58.19	63.51	60.89	16.43	57.09	57.01	67.19	12.37	60.13	58.78	95.11	17.63	61.41	22.95	99.10
	Pe	16.62	69.95	22.32	7.29	21.22	70.76	73.28	73.30	17.54	69.95	66.82	96.70	22.29	71.04	89.42	99.38
EEG BIPOLAR	F4-C4	4.75	8.65	15.21	84.4	5.72	8.14	14.43	83.80	7.75	10.54	13.69	86.25	7.63	10.72	15.2	86.72
	Fe2-F4	2.91	3.16	1.27	43.24	1.72	3.23	1.04	42.29	2.48	4.52	1.32	48.41	1.77	5.34	1.27	43.39
	Fp2-F8	0.11	0.63	2.56	19.12	0.0	0.0	0.0	0.0	0.22	0.07	0.6	19.69	0.0	0.0	0.0	0.0

PATIENT DATA 4 MEAN REMOVED

PERCENTAGE ERROR OF FIT		MODEL 2 EEGm = a1 T3 + a2 FP2 + a3 ... + EEGT				MODEL 9 EEGm = a1 FP2 + a2 F8 + a3 FP1 + a4 F7 + EEGT				MODEL 4 EEGm = a1(FP2-IE) + a2(FP1-I1) + a3(F4-N) + a4(F3-N) + EEGT				MODEL 5 EEGm = a1(FP2-F8) + a2(FP1-F7) + EEGT			
		YEM	HEM	BLINK	N-A	YEM	HEM	BLINK	N-A	YEM	HEM	BLINK	N-A	YEM	HEM	BLINK	N-A
UNIPOLAR Ref to linked ears	FE	5.67	54.67	3.38	53.06	5.84	41.77	3.72	54.21	5.13	65.64	6.36	68.52	6.07	79.02	8.3	43.3
	CZ	32.06	83.89	49.81	93.77	39.27	70.09	54.05	93.11	38.21	79.63	57.07	18.41	44.26	85.3	73.23	99.9
	PZ	38.22	75.55	85.72	95.05	41.66	63.37	27.72	95.04	43.43	79.20	83.1	98.97	57.47	79.44	93.31	99.8
	F4	5.72	3.24	3.8	45.87	4.22	1.86	3.61	33.79	6.36	4.26	6.2	42.61	5.52	5.18	6.55	56.03
	C4	36.94	20.91	50.2	76.86	38.31	16.55	54.52	77.33	34.15	21.98	56.98	80.21	43.28	23.44	68.36	83.89
	T4	57.52	41.54	70.32	88.62	59.10	38.41	76.46	90.22	65.34	42.53	78.89	90.46	84.42	62.39	85.96	96.52
	F3	3.85	4.84	3.29	61.30	4.51	3.16	3.09	52.19	4.88	5.82	6.33	62.02	5.09	7.06	9.46	73.63
	C3	6.9	4.58	34.48	59.26	19.32	13.81	32.87	61.75	6.35	13.62	38.94	67.31	21.49	15.46	56.84	67.73
	T3	56.26	97.66	92.47	95.36	60.99	97.69	95.28	93.42	66.89	97.99	94.74	93.74	80.10	98.61	97.48	96.71
BIPOLAR EEG	F4-C4	10.71	7.48	6.04	66.38	8.22	6.07	6.02	52.89	19.77	6.67	6.68	57.10	18.85	6.96	7.0	69.75
	FP2-F4	1.2	3.66	0.5	39.63	1.25	6.85	0.55	39.88	7.98	9.23	0.83	80.12	5.64	8.95	0.9	47.26
	FP2-F8	1.72	0.75	0.41	19.71	0.0	0.0	0.0	0.0	0.86	0.11	0.26	2.6	0.0	0.0	0.0	0.0
	C4-T4	46.44	59.71	39.17	90.19	46.21	58.46	59.26	89.41	49.66	60.27	57.18	89.85	68.72	63.00	41.01	91.25
	F4-T6	6.96	9.26	6.01	60.83	6.23	8.7	5.86	48.65	16.79	9.79	6.56	54.71	4.8	10.87	7.04	62.49
	T6-F7	8.22	1.15	5.3	19.71	3.01	2.41	4.15	4.57	18.45	0.58	5.6	7.12	21.85	0.65	7.75	11.99
	C6-T5	56.32	47.23	68.02	96.52	56.33	46.43	68.98	95.98	62.88	44.99	69.22	95.5	63.71	47.87	71.9	98.0
	F7-T5	12.03	20.54	15.0	90.69	12.45	19.47	14.74	89.98	20.84	20.53	16.05	92.27	19.49	22.23	17.44	94.49
	T5-F7	10.01	2.03	16.02	43.26	4.92	1.26	14.21	27.17	16.93	1.51	15.63	27.0	20.77	1.83	17.64	36.68

PATIENT DATA S

Percentage error of fit		Model 2 $EEG = a_1 F_4 + a_2 FP_2 + a_3 P_4 + a_4 M_1 + CCGT$				Model 9				Model 4				Model 5			
		VEM	HEM	BLINK	N-A	VEM	HEM	BLINK	N-A	VEM	HEM	BLINK	N-A	VEM	HEM	BLINK	N-A
UNIPOLAR Ref to linked ears	F2	18.69	28.63	2.43	57.63	11.01	27.84	1.52	53.17	7.21	37.71	1.97	71.18	47.83	42.99	2.78	33.95
	C2	30.08	34.94	19.25	71.14	22.67	35.98	11.84	74.04	11.90	60.65	12.18	74.66	83.31	40.49	14.81	91.84
	P2	36.75	19.84	45.70	81.11	27.78	19.12	42.57	78.00	15.83	20.74	58.42	85.07	91.43	20.50	47.72	95.17
BIPOLAR	F4-C4	21.28	2.74	2.77	63.9	4.35	2.66	2.66	65.9	13.72	3.05	2.82	72.16	15.18	3.99	3.35	78.4
	FP2-F4	4.33	9.55	0.17	65.68	2.48	8.51	0.11	65.56	1.99	14.31	0.25	67.68	3.91	22.77	0.37	69.45
	P2-F8	0.96	0.69	3.69	44.29	0.0	0.0	0.0	0.0	0.25	0.03	0.34	26.3	0.0	0.0	0.0	0.0

FS

EEG

PATIENT DATA 6   FROM PATIENT 5

F6

Percentage error of fit		MODEL 2 $EEGm = a_1 I_1 + a_2 FP_2 - a_3 ... + EEGT$				MODEL 9 $EEGm = a_1 FP_2 - a_2 F8 + a_3 FP_1 - a_4 F7 + EEGT$				MODEL 4 $EEGm = a_1(FP_2-I_2) + a_2(FP_1-I_1) - a_3(F8-N) + a_2(F7-N) + EEGT$				MODEL 5 $EEGm = a_1(FP_2-F8) + a_2(FP_1-F7) + EEGT$			
		ROLL VEM	NYSTAG HEM	MUSCLE BLINK	N-A	ROLL VEM	NYSTAG HEM	MUSCLE BLINK	N-A	ROLL VEM	NYSTAG HEM	MUSCLE BLINK	N-A	ROLL VEM	NYSTAG HEM	MUSCLE BLINK	N-A
UNIPOLAR REF TO LINKED EARS	Fz	1.46	14.36	6.3	34.19	1.02	4.97	3.02	32.13	0.85	17.36	57.07	51.48	1.76	23.58	66.53	56.13
	Cz	8.98	11.32	8.89	63.24	1.84	13.68	5.25	46.38	1.62	9.69	57.79	67.00	1.98	15.68	67.12	71.5
	Pz	3.81	15.95	18.05	83.22	2.42	18.15	7.6	87.08	2.29	15.85	61.60	83.46	2.52	19.75	71.21	88.14
	F4	2.02	5.67	7.73	42.62	2.27	5.71	5.58	40.81	2.14	7.52	81.29	56.68	4.37	9.64	95.69	63.46
	C4	4.17	45.71	9.62	62.79	3.16	52.84	6.89	69.71	2.92	4.88	73.49	66.10	3.26	59.41	84.56	72.10
	T4	16.63	73.84	10.65	41.67	16.97	16.69	9.72	38.16	16.2	12.57	86.12	48.02	18.13	18.08	98.61	50.66
BIPOLAR	F4-C4	3.08	3.37	27.42	73.88	1.65	3.73	24.64	71.32	2.25	4.33	24.38	81.47	3.43	5.00	27.36	92.24
	Fz-F4	0.43	3.44	17.29	28.35	0.32	3.98	12.38	31.48	0.79	5.78	25.86	43.22	1.27	9.27	29.12	48.93
	Fz-F8	1.17	0.45	6.26	56.75	0.0	0.0	0.0	0.0	0.05	0.03	0.62	34.45	0.0	0.0	0.0	0.0
	C4-T4	2.66	6.19	21.49	67.29	1.66	5.58	20.23	66.87	1.56	5.58	21.52	65.11	1.68	5.99	24.12	73.72
	F4-T4	3.43	20.58	51.59	94.56	3.3	20.91	56.91	85.95	3.39	25.99	57.72	91.46	5.66	28.61	71.46	92.41
	T4-F8	1.79	0.46	2.47	48.93	2.1	0.13	1.82	43.63	0.38	0.23	2.32	34.45	0.46	0.38	5.10	62.74

EEG

PATIENT DATA 7 21-12-88

PERCENTAGE ERROR OF FIT		MODEL 2 EEGm = ... ·EEGT				MODEL 3 EEGm = ... ·EEGT				MODEL 4 EEGm = ... ·EEGT				MODEL 5 EEGm = ... ·EEGT			
		Vem	Hem	Bunk	N-A	Vem	Hem	Bunk	N-A	Vem	Hem	Bunk	N-A	Vem	Hem	Bunk	N-A
Ref UNI/O CMAS	Fz	6.69	13.88	4.17	1.86	6.48	14.67	5.27	1.83	6.56	14.62	12.7	1.76	9.25	16.32	12.71	10.6
	Cz	16.34	15.35	23.76	3.18	14.98	17.31	22.68	2.47	14.65	15.25	37.69	2.72	21.42	17.68	40.81	10.16
	Pz	21.52	16.51	51.35	3.65	18.15	16.16	52.41	2.89	19.14	15.26	66.6	3.79	28.02	16.48	71.52	8.42
	F4	6.79	34.05	7.3	1.37	7.12	33.25	6.48	0.95	7.25	41.53	16.42	1.3	9.83	48.81	16.16	7.83
	C4	19.05	37.07	28.37	5.77	18.5	43.8	30.02	4.26	18.26	37.34	43.56	5.16	27.47	46.53	45.99	19.48
	T4	16.29	44.07	34.25	6.31	15.62	53.46	25.81	4.01	16.33	50.21	43.06	5.66	26.55	62.66	43.02	18.0
	F3	6.79	2.08	8.04	2.73	6.25	1.54	7.96	2.5	6.99	2.03	14.74	0.39	11.53	2.37	4.73	1.93
	C3	17.85	6.32	33.31	1.82	17.22	7.10	37.67	1.33	16.46	6.57	45.02	1.4	25.55	7.43	48.68	4.14
	T3	19.27	2.62	21.16	0.85	15.87	2.06	17.59	2.55	16.99	2.17	24.2	0.53	29.38	2.52	24.87	1.16
BIPOLAR	F4-C4	53.31	16.35	11.76	0.55	47.03	18.52	10.79	0.25	47.94	17.67	16.26	0.3	59.61	21.73	16.02	0.81
	FP2-F4	7.39	16.0	0.46	0.26	7.18	18.1	0.49	0.19	5.87	23.85	1.6	0.4	7.23	29.47	1.43	0.75
	FP2-F8	0.44	1.64	7.0	0.53	0.0	0.0	0.0	0.0	0.21	0.06	0.27	0.07	0.0	0.0	0.0	0.0
	C4-T4	30.57	13.02	54.51	1.44	30.81	13.23	42.37	1.4	29.61	12.94	42.29	1.23	35.78	4.17	44.31	2.54
	F4-T4	10.38	54.66	11.28	1.16	9.29	58.24	11.26	1.12	10.39	58.55	14.86	1.03	10.05	64.94	4.39	4.46
	T4-F8	29.4	2.09	11.92	0.34	25.54	0.31	1.32	0.04	24.31	0.57	4.3	0.05	26.97	0.69	4.61	0.28
	C3-T3	75.04	21.55	64.33	2.33	69.69	21.71	61.84	2.45	74.11	9.93	58.92	2.62	76.12	22.43	62.03	4.0
	F3-T3	8.05	25.55	31.27	5.27	7.87	25.48	32.37	5.65	8.25	28.68	35.92	5.10	8.52	29.09	35.06	10.5
	T3-F7	62.48	1.89	7.37	0.97	69.23	0.28	1.89	2.11	64.12	0.52	3.92	0.16	72.67	0.55	4.41	0.66

F7

EEG

F8

APPENDIX G

This appendix contains the software implementation of the RLS OA removal algorithm with U-D factorisation. The software interfaces with the data edited using the sofware given in appendix C. Data is read in in 2-8 second blocks and the user is requested to select EEG signals and EOG montages. The same EEG signals and EOG electrode montages are used for this on-line algorithm to enable an objective comparison to be made with the off-line analysis results. The software is written in C using the BORLAND TURBO C software package.

```
/*FILTER: Program to filter the EEG data using the RLS EOG subtraction */
/*       algorithm and using the UD factorisation of the error        */
/*       covarience matrix.                                            */

/*-----------------------------------------------------------------------

   called as filter9 < filename1 >  < filename2 >

   where filename1 is a binary data file containing EEG/EOG samples
                                                   (i.e. from D.A.S)
   and   filename2 is an ASCII file that contains the corrected EEG

   -----------------------------------------------------------------------*/



#include < stdio.h >
#include < alloc.h >
#include < conio.h >
/* #include < graphics.h > */



#define inbuf_size 65502 /* buffer to hold his is for 8-seconds of 16 */
                         /* channel EEG/EOG data                      */
#define outbuf_size 2046 /* 2048 samples = 8seconds x 256 Hertz */

#define true 1
#define false 0



int input_data(char *input,char far *input_buffer);
int output_data(char *output,int far *output_buffer);
int filter(char far *input_buffer,int far *output_buffer,int j);
int udu_flt(double X[],double Y,int far *output_buffer,int model,int ll);


/*---------------------------------------------------------------------*/

/*   Initialisation of matrices - Global delcarations */


     /* UD factorisation variables */

     double U[10] = {4.0e-4,-1.0,4.0e-4,100,100,100,100,100,100,100};

     double THETA[4] = {0.3,0.3,0.0,0.0};

     double B[4] = {0,0,0,0};

     double V[4] = {0,0,0,0};

     double GAMMA = 0.998;

/*   double ALPHA = 1.00;*/


     /* EEG and EOG set ups */

     int NPAR[4] = {4,4,4,2};

     char *channel[] = {
          'Fz',
          'Cz',
```

```c
    "Pz",
    "F4",
    "C4",
    "T4",
    "F3",
    "C3",
    "T3",
    "F4-C4",
    "Fp2-F4",
    "Fp2-F8",
    "F4-T4",
    "C4-T4",
    "T4-F8",
    "F3-T3",
    "C3-T3",
    "T3-F7"
};

/* EOG model numbers */

int eog_model[4] = {2,9,4,5};


/* EOG chan1 byte index into filename1 */

int eog_chan1[4][4] = {
    0,2,10,12,
    2,20,12,24,
    2,12,20,24,
    2,12,0,0
};


/* EOG chan2 byte index into filename1 */

int eog_chan2[4][4] = {
    0,0,0,0,
    0,0,0,0,
    0,10,22,22,
    20,24,0,0
};


/* EOG chan1 multiplier */

int mult1[4][4] = {
    1,1,1,1,
    1,1,1,1,
    1,1,1,1,
    1,1,0,0
};

/* EOG chan2 multiplier */

int mult2[4][4] = {
    0,0,0,0,
    0,0,0,0,
    1,1,1,1,
    0,0,0,0
};

int artefact = false;
double base; /* last value of EEG before OA removal */
```

```c
/*----------------------------------------------------------------*/



/*   Start of main program  */


main(int argc,char *argv[])

{
    int mod_num,key_press;
    char far *input_buffer;
    int far *output_buffer;
    char *input = argv[1];


/*----------------------------------------------------------------*/
    /* check and allocate buffer storage space */

    if((input_buffer = (char *) farcalloc(inbuf_size,sizeof(char)) == NULL)
    {
        printf("\nFILTER9: memory allocation failure");
        exit(1);
    }

    if((output_buffer = (int *) farcalloc(outbuf_size,sizeof(int))) == NULL) {
        printf("\nFILTER: memory allocation failure");
        exit(1);
    }
/*----------------------------------------------------------------*/
    /* check calling argumant count */

    if(argc != 3) {
        printf("FILTER: argument mismatch\n");
        exit(1);
    }
/*----------------------------------------------------------------*/


    /* read data from filename1 */
    if((input_data(input,input_buffer)) == 0)
    {
        printf("FILTER9: data has been read\n\n");

        for(mod_num=3;mod_num<4;mod_num++) {
            char *output = argv[2];

            /* filter input file to output file */
            if((filter(input_buffer,output_buffer,mod_num)) == 0)
            {
                if((output_data(output,output_buffer)) == 1) exit (1);

                printf("\n\nFILTER9: job done!\n\nData from %s has been filtered into %s.\n",input, output);
                printf("\nFILTER9: PRESS ANY KEY TO CONTINUE....");
                key_press=getch();
            }
        }

    }else
        exit(1);


    farfree(input_buffer);
    farfree(output_buffer);
```

```c
        return 0;

}

/*  end of main program  */

/*---------------------------------------------------------------------*/

/*  input_data: to read data from das file to memory.  */

int input_data(char *input,char far *input_buffer)
{
        FILE *ifp;

        if((ifp = fopen(input,"rb")) ==NULL) {
                printf("FILTER: can't open %s\n",input);
                return 1;
        }else{
                if((fread((char far *)input_buffer,sizeof(char),inbuf_size,ifp)) != inbuf_size) {
                        printf("FILTER: file read error\n");
                        fclose(ifp);
                        return 1;
                } else
                        return 0;

        }

}
/*---------------------------------------------------------------------*/

/*  output_data: to write filtered data toan ILS file.  */
int output_data(char *output,int far *output_buffer)

{
        FILE *ofp;
        int i;

        if((ofp = fopen(output,"wt")) == NULL) {
                printf("FILTER: cant open %s\n",output);
                return 1;

        }else{
                for(i=0;i<(outbuf_size-1);i++)
                        fprintf(ofp," %d\n",output_buffer[i]);
        }
        return 0;

}

/*---------------------------------------------------------------------*/

/*  filter: controls the initialisation of matrices and invokation of the   */
/*      UD algorithm */

int filter(char far *input_buffer,int far *output_buffer,int model)
{

        int c,i,j,k,chan,index,eeg_chan1,eeg_chan2,c1,c2;
        double X1,X2,X[4],Y1,Y2,Y;


        double X1_old_mean[4] = {0.0,0.0,0.0,0.0};
```

```c
    double X1_new_mean[4] = {0.0,0.0,0.0,0.0};
    double X2_old_mean[4] = {0.0,0.0,0.0,0.0};
    double X2_new_mean[4] = {0.0,0.0,0.0,0.0};
double Y1_old_mean = 0;
    double Y1_new_mean = 0;
    double Y2_old_mean = 0;
    double Y2_new_mean = 0;


    /* command screen */

    textattr(LIGHTGRAY+(BLACK< <4));
    clrscr();
    textattr(WHITE+(BLUE< <4));
    gotoxy(28,1);
    cprintf("EEG ARTEFACT REMOVAL");
    gotoxy(19,4);
    textattr(WHITE+(BLUE< <4));
    cprintf("The following channels can be corrected");
    gotoxy(22,6);
    cprintf("1. Fz           10. F4-C4 ");
    gotoxy(22,7);
    cprintf("2. Cz           11. Fp2-F4");
    gotoxy(22,8);
    cprintf("3. Pz           12. Fp2-F8");
    gotoxy(22,9);
    cprintf("4. F4           13. F4-T4 ");
    gotoxy(22,10);
    cprintf("5. C4           14. C4-T4 ");
    gotoxy(22,11);
    cprintf("6. T4           15. T4-F8 ");
    gotoxy(22,12);
    cprintf("7. F3           16. F3-T3 ");
    gotoxy(22,13);
    cprintf("8. C3           17. C3-T3 ");
    gotoxy(22,14);
    cprintf("9. T3           18. T3-F7 ");
    gotoxy(31,20);
    cprintf("Using Model %d",eog_model[model]);
    gotoxy(2,24);
    cprintf("Select an EEG channel to filter.. \b\b");

    while((scanf("%d",&c)) <0 || c>18)
    {
        gotoxy(2,24);
    cprintf("FILTER: not an EEG channel please re-enter.. \b\b");
}
    textattr(LIGHTGRAY+(BLACK< <4));
    clrscr();

    if(c == 0)
      return 1;



    else
    {
        textattr(WHITE+(BLUE< <4));

        switch (c)
        {
            case 1:eeg_chan1 =26;eeg_chan2 = 1;c1 = 1;c2 =0;break;
            case 2:eeg_chan1 =28;eeg_chan2 = 1;c1 = 1;c2 =0;break;
            case 3:eeg_chan1 =30;eeg_chan2 = 1;c1 = 1;c2 =0;break;
            case 4:eeg_chan1 =4;eeg_chan2 = 1;c1 = 1;c2 =0;break;
```

65

```c
        case 5:eeg_chan1 =6;eeg_chan2 =1;c1 =1;c2 =0;break;
        case 6:eeg_chan1 =8;eeg_chan2 =1;c1 =1;c2 =0;break;
        case 7:eeg_chan1 =14;eeg_chan2 =1;c1 =1;c2 =0;break;
        case 8:eeg_chan1 =16;eeg_chan2 =1;c1 =1;c2 =0;break;
        case 9:eeg_chan1 =18;eeg_chan2 =1;c1 =1;c2 =0;break;
        case 10:eeg_chan1 =4;eeg_chan2 =6;c1 =1;c2 =1;break;
        case 11:eeg_chan1 =0;eeg_chan2 =0;c1 =1;c2 =0;break;
        case 12:eeg_chan1 =2;eeg_chan2 =20;c1 =1;c2 =1;break;
        case 13:eeg_chan1 =4;eeg_chan2 =8;c1 =1;c2 =1;break;
        case 14:eeg_chan1 =6;eeg_chan2 =8;c1 =1;c2 =1;break;
        case 15:eeg_chan1 =8;eeg_chan2 =20;c1 =1;c2 =1;break;
        case 16:eeg_chan1 =14;eeg_chan2 =18;c1 =1;c2 =1;break;
        case 17:eeg_chan1 =16;eeg_chan2 =18;c1 =1;c2 =1;break;
        case 18:eeg_chan1 =18;eeg_chan2 =24;c1 =1;c2 =1;break;
}

printf("\nFILTER: filtering channel %s using model %d, please wait...",channel[c-1],eog_model[model]);


for(i=0;i<outbuf_size;i++)
{
        index = (i * 32) + 7;
        Y1 = input_buffer[(index+eeg_chan1)] + 256 * input_buffer[(index+eeg_chan1+1)];
        Y1 = Y1-2048;

        /* remove mean from EEG */

        Y1_new_mean = (GAMMA*Y1_old_mean)+((1-GAMMA)* Y1);
        Y1 = Y1 - Y1_new_mean;
        Y1_old_mean = Y1_new_mean;

        Y1 = c1*Y1;
        Y2 = 0;

        if(c2)
        {
                Y2 = input_buffer[(index+eeg_chan2)] + 256 * input_buffer[(index+eeg_chan2+1)];
                Y2 = Y2-2048;

                Y2_new_mean = (GAMMA*Y2_old_mean)+((1-GAMMA)* Y2);
                Y2 = Y2 - Y2_new_mean;
                Y2_old_mean = Y2_new_mean;

Y2 = c2*Y2;
        }

        Y = Y1 - Y2;


        for(k=0;k<NPAR[model];k++)
        {
                X1 = input_buffer[(index+eog_chan1[model][k])] + 256*input_buffer[(index+eog_chan1[model][k]+1)];
                X1 = X1-2048;

                /* remove mean from EOG */

                X1_new_mean[k] = (GAMMA*X1_old_mean[k])+((1-GAMMA)* X1);
                X1 = X1 - X1_new_mean[k];
                X1_old_mean[k] = X1_new_mean[k];

                X1 = mult1[model][k]*X1;
        X2 = 0;

                if(mult2[model][k])
```

G6

```
                               {
                                    X2 = input_buffer[(index+eog_chan2[model][k])] + 256*input_buffer[(index+eog_chan2[model][k] + 1)];
                                    X2 = X2-2048;

                                    X2_new_mean[k] = (GAMMA*X2_old_mean[k])+((1-GAMMA)* X2);
                                    X2 = X2 - X2_new_mean[k];
                                    X2_old_mean[k] = X2_new_mean[k];

                          X2 = mult2[model][k]*X2;
                               }

                 X[k] = X1-X2;

                    }

                 if(i < 2048)
                 {
                      artefact = true;
                      if((udu_flt(X,Y,output_buffer,model,i)) != 0) return 1;
                 }
                 else
                 {
                 artefact = false;
                      output_buffer[i] = Y;
                 }

           }
     for(j=0;j<NPAR[model];j++) printf("\nFILTER: THETA%d = %f",j+1,THETA[j]);
     }
   return 0;
}
/*------------------------------------------------------------------*/


/* udu_flt: RLS filtering of the EEG using UD factorisation */


int udu_flt(double X[],double Y,int far *output_buffer,int mod,int I1)
{
     double SF,PERR,ALPHA,BETA,BETA1,DELTA,P;
     int J1,m,j,k,i;



     SF = 1 / GAMMA;
     PERR = Y;
     for(j=0;j<NPAR[mod];j++)
     {
          PERR = PERR - (X[j] * THETA[j]);

     }
     m = 0;
     V[0] = X[0];
     for(j=1;j<NPAR[mod];j++)
     {
          V[j] = X[j];
          J1 = j;
          for(k=0;k<J1;k++)
          {
               m = m + 1;
               V[j] = V[j] + (U[m]*X[k]);
          }
          m = m + 1;
          B[j] = U[m] * V[j];
     }
```

```
B[0] = U[0] * X[0];
ALPHA = GAMMA + (B[0] * V[0]);
DELTA = 1/ALPHA;
U[0] = U[0] * DELTA;

m =0;
for(j=1;j<NPAR[mod];j++)
{
    BETA1 = ALPHA;
    ALPHA = ALPHA + (B[j] * V[j]);
    P = -V[j] * DELTA;
    DELTA=1/ALPHA;
    J1 = j;
    for(k=0;k<J1;k++)
    {
        m = m+1;
        BETA = U[m];
        U[m] = BETA + (B[k] * P);
        B[k] =B[k] + (B[j] * BETA);
    }
    m = m + 1;
    U[m] = U[m] * BETA1 * DELTA * SF;
}

PERR = PERR/ALPHA;
for(j=0;j<NPAR[mod];j++)
{
    THETA[j] = THETA[j] + (B[j] * PERR);
}

output_buffer[I1] = Y;
for(i=0;i<NPAR[mod];i++)
{
    output_buffer[I1] = output_buffer[I1] - (X[i] * THETA[i]);
}
output_buffer[I1] = output_buffer[I1] +(artefact * base);

return 0;
}


/*-------------------------------------------------------------------*/
```

APPENDIX I

This appendix contains a example transcription of the non-structured interviews held with a consultant clinical neurophysiologist. This interview was carried out to obtain general and specific information regarding EEG analysis from data acquired using the DAS. This form of transcription is crucial for knowledge elicitation to help in understanding the problem domain and provide a hard copy reference for any rules developed.

Transcription of interview with Dr E.M. Allen

Date: 1st June 1989

Location: Dept of Neurophysiology
                Derriford Hospital,
                Plymouth.
                Devon.

EM so what is it you want. you want me to tell you about the abnormal bits?

MH Yes that's right well this is the first patient PR. Now what I would like you to do is to pick out some of the abnormal waves, and if we can mark it so that I can photocopy those sections later.

EA um err will I need to know. do I need to know where you have been recording..on the bit that you've got on your.

MH well the whole thing has been recorded.

EA ahh..good,right...

MH yes for this one yes I have slow waves..bilateral, more right than left..five second bursts.delta,yes those are all the notes, an abstract of the notes that I've taken...

EA Right well here's an example of the delta waves more marked on the right than the left. this is right...left..right.left.n and umm...if I take this channel here.channel two you will see that there are some slow waves here that are not on channel..six which is the equivalent channel on this side..um.suggesting .um you know and then one would go on to know wether or not that was like it all the time.

MH umhm

EA ah in there's another instance.where they don't look so prominent on the left as they do on the right.....and.here's another example and there are some slow waves there.which are not.um are different

MH yes they look quite sharp

EA yes they're slower..

MH yes

EA so this one occupies for instance more than half a second and on the other side..it occupies less than a second, if we take just that one slow wave.

MH yes..

EA see, that's a second and that more. so that you might say is a more or less equivalent.er wave, its in an equivalent channel.

MH and where about on the scalp is that

EA and that's in the temporal phase, temporal, and then going back to this parietal (unintelligible word) ahh it's a similar thing really, they are...sort of what we call theta waves here occupying about a fifth of a second.but here they occupy something more like a third of a second, see those wave here.um.one way of looking at that the slow waves are more abundant on the, on the right than the left..not only more in quantity and when they are there they are also slower...slower in frequency, so they are more abundant on the right than the left and um and slower in frequency and this occurs in both biparietal planes and the temporal plane..some of them to complicate matters are bilaterally synchronous here...

MH yes.

I 1

EA and the question arises therefore are they eye movements..

MH for instance that one you showed me, the slower delta.

EA yah

MH that,that has a similar appearance to some of the.the er eye movements that we've seen

EA yes

MH with the sharp leading edge and the trailing.

EA yes that's right

MH but you didn't seem to think that that was

EA that's because its coming from the wrong part of the head

MH right

EA its coming from the posterior aspects of the head..

MH ok

EA um.so that was the reason that I didn't suggest eye movements, its coming up posterior, all right

MH um hm

EA um..there are eye movements here.where I think its impossible to know which is which.that one and that one...

EA yes, yes, there's another instance there where they are bilaterally synchronous and I, I, I don't think I know there which is which.and I think what, in order to cope with that situation one looks for where one sees real eye movements and I think that that is a real eye movement there and there. That's a sort of equivalent to a blink..but um...

MH will it be possible just to mark, give this page a mark so that I can..refer back to it (unintelligible word)

EA yes

MH that's fine

EA um.there's a blink

MH right

EA blink, blink and an eye movement.... that's quite interesting that these eye movements are not absolutely symmetrical...

MH bigger on the.on the right, yes or.

EA bigger on the left, on the left

MH oh I see sorry, yes

EA now what this is I don't know.9,10,11.12..yes there just so asymmetrical..

MH could it be a lateral eye movement

EA yes I think it has to be a lateral eye movement.because they're not so bad when that they are a little asymmetrical a bit.but I think they are an oblique movement really...because they're just that little bit asymmetrical but not too, too - marked..now I'm just concentrating on the first 16 channels I'm not taking into account these other channels..ok

MH there's no other way you could tell, I suppose you wouldn't be interested but would there be any other way that you would tell that that was a lateral or.vertical eye movement. You mentioned in the past about um

EA well I think that you could consider that it might be in that.here we have um between 13 and 14, that's on the left ssside..its diverging but you would expect it to converge on this side, but it's not quite doing that either..I wonder if infact this chaps got what you might call a disconjugate gaze..

MH where one eye.

EA yes

MH moves and the other is stationary or?

EA yes that's right

MH does that happen alot.

EA it happens when people are very sick..

MH um hm

EA and I would imagine that by looking at this EEG that this chap is very sick.. um

MH yes that's interesting actually..because, yes one of the main attributes that we've drawn out is this..is this converging and diverging method of being able to identify eye movements, if the eyes aren't moving

EA yes if they are wandering around you would

MH it would cause problems.

EA I yes, um..it would be an un. I mean a fairly unusual situation and um.and I think that you call the.You would want to be in the knowing of what was happening at other parts of the..head. So it doesn't worry me unduly because..what I see happening here is um something that is alot more marked in the temporal..plane.

MH yes

EA that's 4 right.4 left in the parietal, 4 right, 4 left

MH again delta

EA delta is a higher voltage slower in the temporal than it is in the parietal. There isn't so much asymmetry this time...it's a very difficult record I might add, very difficult, but in that particular patch is symmetrical..

MH is it difficult because of the..cerebral signals

EA yes, yes no, there's no artefacts here to speak of here..I think the cerebral interpretation of what's happening cerebraly is very difficult..and what is so infuriating is that at the end of the day is that it probably doesn't make all that much difference to the patient. All one knows is that he's got a very sick brain..and that it's probably imflammatory.nature..and that added to which there's an ECG artefact all the way through...you can really notice that

MH Do you note that from the periodicity of it

EA Yes..um

MH just picking it out on, it seems to be the, is it the temporal?

EA It's the regularity...um

MH would you expect to see it on other parts of the scalp rather than round here

EA Usually you see it on the left temporal area more than elsewhere...but um..If someone is really hypertensive, got high blood pressure, which I don't know if this chap did or no..you can see it all over...and for some reason unhealthy brain seem to transmit..particularly dead brain, transmits an ECG artefact much better than a healthy brain.

MH so its actually the brain the transmits the artefact rather than a muscle or an artery

EA Yes, it's acting as a conductor..so there's an asymmetry and a temporal proponderance..there's.waves.that are..are quite synchronous and.pretty symetrical.......stop, and rest

MH I think that

EA And that.here,here's something that's eye movement, what's happening here. That's definitely rolling eye movement, all right

MH o.k yes, that's because of the amplitude and position on the scalp

EA The slowness of it all....now the question is, is this the rolling eye movement going the other way...in a slightly lateral way perhaps...I think it probably is because here's a situation where you can actually see..the divergence..there and the convergence, here, all right..so that's going away..so there was an upward eye movement..and here was a lateral eye movement, so we know that this chaps got rolling eye movements, slow

rolling eye movements..um probably going in all directions..another way to..(unintelligible word) and honestly in such a situation you could have got this disconjugate eye movements...

MH do those, er do those rolling eye movements, would they suggest anything else to you?

EA No, I mean it could be just going to sleep,it could be on a hot day and going to sleep..er...I think..that bit's not showing us anything that we've not already seen, so that, that's certainly the main gist of this..this..EEG..here's some slow waves sort of in the mid-line, now they are quite interesting because..we don't as it happens, we're not as it happens seing bilaterally synchronous eye movements

MH un hm

EA as such, sorry, bilaterally synchronous slow waves and sometimes when you've got bilaterally synchronous slow waves and you don't know wether they're eye movements or not, if you go to a position when you're, where you're recording across the head...then they tend to do, to do er ..this(starts to draw diagram)so you've got the, perhaps you've got the.eye movements going up like that in channels 1.5..1 and 5 and then you're going across the head, and then you have them diverging.accross the head.but when you've got some um.slow waves here,which across the head are in fact..doing that....and that is in the middle of the head.now that's normally some abnormal brain wave.because eye movements are not usually recorded there..so anything that occurs in what we call the mid-line

MH You're less likely to have any problems

EA Yes because they cancel each other out you see..thats..so if you see.slow waves which could be eye movements, actually recorded in the mid-line.I think we've mentioned this before, then they're less likely to be eye movements, and here's that wave again. This particular position is one that which goes round the head, so that's 2 right, I think, 1 in the middle and 2 left, and here you see eye movement, one and you see there's almost no eye movement there..and here.the slow wave is actually at it's maximum in the mid-line..all right..mid-line...

MH Do you think we could possibly mark this page as well...........

EA Right, so um..anything else

MH Well I think that you've just about shown me all the abnormalities

EA Lots of interesting, there's ECG don't forget to mention, you could wonder what that was.because..it could look like a funny blip..now here's something that's going on here, now this is, this is..seems to be..it's in position 2 (montage 2). So the first bit we talked about was position 1..what we're talking now about is position 2...now we've got some very funny stuff happening here...but here these um 1,2,3,4,5,6,7,8..we've got a slow wave here.occuring on the left.so here we have the situation where it's worse on the left than on the right..yes.because that slow wave is happening on the left and not on the right.I don't think its eye movement.this is left,right.because this is the position that I mentioned before and we're going across the head, so if anything is eye movement, that's eye movement, but I don't know wether it is or not, it could be and A) you have in the middle these particularly slow waves, yes, so they're happening in the mid-line..

MH Where you wouldn't expect any eye movement

EA Yes, that's right, and now if we come back to this bit, so in fact we've got eye movements happening here as well

MH so in this cas it's a combination of the 2

EA Yes, that's right. There's these great big slow wave in the mid-line, and.. and this is left..and I think that this little eye movement is..this

14

one

MH Why should that movement.on er.why should that movement not be part of the slow wave..

EA Um, because it's going in the opposite direction..as we've just mentioned

MH Yes

EA It's. that, and that, and we've got that in the middle(referring to diagram)....so that's where we've got a combination of it.and there's even a bit there.and a bit there..and there's another um........I don't know whether there's eye movement there or not..

(Break in interview caused by telephone)

APPENDIX K

This appendix provides the software listings for the complete IOARS. The software is comprised of three sections:

.PROLOG software listings.
.C software listings.
.Assembler software listings.

PROLOG is used for symbolic EEG feature conversion, knowledge manipulation, user interface, and rule representation. The PROLOG software listings consist of the following modules:

. main - provides the entry to other modules via a main loop.
. cnv - provides the conversion routines to map between numerical and symbolic EEG features.
. inf - provides the forward/backward inferencing mechanism with uncertainty management.
. sup - provides the support routines to handle data loading/saving.
. win - provides the window definition routines.
. ut - provides various utility routines.

The software was written using the BORLAND TURBO PROLOG software package.

C is used for numerical feature extraction. The C software listing consists of the following module:

. extoprol - this is a large program that includes three callable modules:
      . extract-psd - provides the PSD feature extraction routine.
      . extract-ac - provides the autocorrelation feature extraction routine.
      . extract-cc - provides the cross-correlation feature extraction routines.

The software was written using the BORLAND TURBO C software package.

80286 assembler is used for file transfer and fast graphical EEG display. The assembler software listings consist of the following modules:

. readPSD - reads an ILS PSD file.
. readCORR - reads an ILS correlation file.
. line - plots an EEG time series.
. load - loads a mode 12h graphics screen.
. readEEG - reads a binary EEG data file (from DAS)
. scrorig - scrolls the graphics screen.
. vidmode - sets the graphics video mode.

The various modules are linked together using a project file with the assembler files in a library called EEG.lib and the compiled extoprol file.

```
/* FILE: 'main2-mo.pro' */

/* make using EEG.prj and lib files Prolog,EEG,fp87,MathI and CI.*/
/*    EEG2.prj:
            main2-mo + /* includes dec6_mod.pro */
            cnv2-mod +
            inf6-mod + /* includes uni_mod.pro */
            sup3-mod +
            ut3-mod +
            win2-mod +
            extoprol.obj
*/
/* EEG incorporates assembler routines called from c external predicates */
/*    EEG.lib:
            ReadPSD.asm - read PSD ILS file.
            ReadCORR.asm - read Correlation ILS file.
            LINE.asm - plot EEG time series.
            LOAD.asm - load graphics mode 12 screen.
            READEEG.asm - read binary EEG data.
            READKBD.asm - scan keyboard(including extended keys).
            SCRORIG.asm -
                BLANKSCREEN - turns display off while image is loaded.
                SCREENORIGIN - sets start address of video display.
                SETWIDTH - sets width of display.
                SHOWSCREEN - turns display on.
                SPLITSCREEN - creates two graphics display areas.
            VIDMODE.asm - sets the graphics display mode.*/

/* called as EEG2.exe */


project 'EEG2'
include 'dec6-mod.pro'  /* global declarations module */


predicates          .

/*    run         declared in dec5_mod.pro as global*/
      main_choice(INTEGER)
      stop_main(INTEGER)
      control_choice(INTEGER)
      op_control(INTEGER)


goal

      run.


clauses

      run :-
/*          textmode(24,80),  special to old pcV vga */
            load_kb,
            new_conds,
            repeat,
            main_window,
            main_menu(Choice1),
            main_choice(Choice1),
            stop_main(Choice1),
            removewindow,
            !.
```

```
/*--------------------------------------------------------------------*/

/* "stop" is a backtrackable predicate to cause a failure if the choice is
   not 'ESC' or EXIT, hence repeat to the main menu */


        stop_main(0).
        stop_main(3) :-
            shiftwindow(1),
            cursor(12,17),
            write('F I N I S H, are you sure ? (y or n) '),
            readchar(C),
            write(C),
            C = 'y',
            trans_tru2log,
            goodbye_window,
            write('\n     EEG \n    Terminated"),
            wait_count(20000),
            wait_count(20000),
            wait_count(20000),
            removewindow,
            removewindow(1,0),
            exit.


        stop_control(0).
        stop_control(5).

/*--------------------------------------------------------------------*/
/* "main_choice" is a series of windows selected from the main window to give
   access to the highest level program functions. */

        main_choice(1):-
            get_data,
            clear_context,
            start_context,
            !.

        main_choice(2):-
            repeat,
            control_menu(Choice2),
            control_choice(Choice2),
            stop_control(Choice2),
            !.

        main_choice(3) :-
            !.

        main_choice(_) :-
            stop_main(3),
            !.

/*--------------------------------------------------------------------*/
/* "control_choice" is a series of windows selected from the main window to give
   access to the second level program functions. */

        control_choice(1) :-
            data(_),
            monitor,
            !.

        control_choice(2) :-
            data(_),
            select_segment,
```

```prolog
        !.

control_choice(3) :-
    data(_),
    seg(_),
    clear,
    new_data_conds,
    process_window,
    get_features,
    log_features,
    removewindow,
    decision_window,
    decision,
    removewindow,
    !.

control_choice(3) :-
    not(seg(_)),
    warning_window,
    write("You must select a segment first"),
    write("\nPress any key to continue..."),
    readchar(X),
    removewindow,
    !.

control_choice(4) :-
    inspect_kb,
    !.


control_choice(5) :-
    !.

control_choice(Choice):-
    not(data(_)),
    Choice < >0,
    warning_window,
    write("You must load data first"),
    write("\nPress any key to continue..."),
    readchar(X),
    removewindow,
    !.

control_choice(_) :-
    stop_control(5).
```

/*..*/

K4

```
/* File: dec6-mod.pro */

/* This module contains the global domains, databases and predicate
   declarations. */


global domains

    CONDITIONS = Cond_No*
    HISTORY = Rule_No*
    Rule_No, Cond_No, FNO = INTEGER
    CATEGORY = STRING
    data_file = string
    file = save_file
    slist = string*
    flist = real*
    CF = real
    CFS = CF*
    INT = reference INTEGER
    INTlist = INT*
    PARAMETER = reference SYMBOL
    PAIR = pair(PARAMETER,PARAMETER,INT)
      .IRlist = PAIR*
    PARMlist = PARAMETER*
    Llist = PARMlist*


global database

    rule(Rule_No,CF,CATEGORY,CATEGORY,CONDITIONS) /*text rule*/
    tex_cond(Cond_No,STRING) /*text condition*/
    num_cond(Cond_No,PARMlist) /*numerical condition*/
    yes(Cond_No,CF)      /*successful condition and certainty*/
    no(Cond_No,CF)       /*unsuccessful condition and certainty*/
    frf(CF) /*successful condition demon certainty*/
    minimum(CF)
    cl(PARMlist,CFS,Llist) /*clausal representation for interpreter*/
    eeg_peak(PARAMETER,PARAMETER,PARAMETER) /*fact asserted by demons*/
    sym_peak(PARAMETER,PARAMETER) /*fact asserted by demons*/
    larg(PARAMETER,PARAMETER) /*fact asserted by demons*/
     ta_file(data_file)
     .annel(INT,PARAMETER,PARAMETER) /*EEG/EOG channels present*/
    autocorr(INT) /* channels to obtain autocorrelation */
    crosscorr(INT,INT,INT) /*channels and file to obtain crosscorrelation */
    adjacent(PARAMETER,PARAMETER)
    lr_sym(PARAMETER,PARAMETER) /*symetrical attributes*/
    peak(real,PARAMETER) /*peak names*/
    freq_bounds(PARAMETER,real,real) /*fuzzy frequency bounds*/
    power_bounds(real,PARAMETER) /*fuzzy power bounds*/
    similar_bounds(real,PARAMETER) /*fuzzy correlation bounds */
    data(string) /* dataset number */
    seg(string) /* segment number */


global predicates

    run
    evalans(char,HISTORY,CATEGORY,CF) - (i,i,i,i)    /* clause in ut-mod.pro */
    clear /* clause in ut-mod.pro */
    wait_count(INTEGER) - (i)       /* clause in ut-mod.pro */
    reverse(CONDITIONS,CONDITIONS) - (i,o)    /* clause in ut-mod.pro */
    reverse1(CONDITIONS,CONDITIONS,CONDITIONS) - (i,i,o) /* clause in ut-mod.pro */
    no_sys_error(string,integer) - (i,i) /* clause in ut3-mod.pro */
    main_window   /* clause in wind-mod.pro */
```

KS

```
main_menu(INTEGER) - (o)      /* clause in wind-mod.pro */
control_menu(INTEGER) - (o) /* clause in win2-mod.pro */
process_window      /* clause in wind-mod.pro */
decision_window      /* clause in wind-mod.pro */
list_window /* clause in wind-mod.pro */
logopt_window(INTEGER) - (o) /* clause in wind-mod.pro */
listopt_window(INTEGER) - (o) /* clause in wind-mod.pro */
edit_window      /* clause in wind-mod.pro */
question_window(INTEGER) - (o)      /* clause in wind-mod.pro */
explain_window/* clause in wind-mod.pro */
goodbye_window      /* clause in wind-mod.pro */
kb_select_window(string) - (o) /* clause in wind-mod.pro */
data_select_window(string) - (o) /* clause in win2-mod.pro */
select_segment_window(string) - (o) /* clause in win2-mod.pro */
rules_select_window(string) - (o) /* clause in win2-mod.pro */
warning_window /* clause in win2-mod.pro */
load_kb        /*clause in sup2-mod.pro */
nondeterm repeat /*clause in tdoms.pro */
get_features      /*clause in conv-mod.pro */
decision    /*clause in inf4-mod.pro */
edit_kb      /*clause in sup2-mod.pro */
inspect_log      /*clause in sup2-mod.pro */
    pect_kb /*clause in sup3-mod.pro */
list    /*clause in sup2-mod.pro */
nondeterm save_kb /*clause in sup2-mod.pro */
get_kb /*clause in sup2-mod.pro */
get_data /*clause in sup3-mod.pro */
select_segment /*clause in sup3-mod.pro */
nondeterm show_rule(Rule_No,string) - (i,o) /*clause in inf4-mod.pro */
nondeterm report(HISTORY,string) - (i,o) /*clause in inf4-mod.pro */
nondeterm infer(HISTORY,CATEGORY,CF) - (i,i,i) /*clause in inf4-mod.pro */
erase /*clause in sup2-mod.pro */
start_context /*load dummy starting context profiles */
clear_context /*clear contextual features*/
context_assert(char,CATEGORY,CF) - (i,i,i) /* provide a local context for inference */
measure_of_belief(PARAMETER,CF,CF,CF,CF) - (i,i,i,i,o) /*certainty combination*/
new_conds /* clause in sup3-mod.pro */
new_data_conds /* clause in sup3-mod.pro */
log_features /* clause in sup3-mod.pro */
log_conds /* clause in sup3-mod.pro */
    q_results(char,CATEGORY,CF) - (i,i,i) /* clause in sup3-mod.pro */
    ns_tru2log /* clause in sup3-mod.pro */


/* the following predicates are external and written in c. All predicates */
/* are contained in the file extoprol.c because the prolog project facility */
/* seems unable to link multiple external object files. */

extractPSD(INTEGER,flist) - (i,o) language c /* clause in extoprol.c. INTEGER is the*/
                    /*ILS record file number, and flist is the returned linked*/
                    /*list of spectral peak features*/
extractCORR(INTEGER,flist) - (i,o) language c/* clause in extoprol.c. INTEGER is the*/
                    /*ILS record file number, and flist is the returned linked*/
                    /*list of correlation features*/
monitor - language c /* clause in mon.c. Displays graphics data */
```

```
/*FILE: cnv2-mod.pro*/

/* Program to convert input feature lists to asserted feature facts
   in clausal form. */

project "EEG2"
include "dec6-mod.pro"

domains
    element = p(PARAMETER);
          l(CFS)
    Xlist = element*
    Ylist = Xlist*


predicates
/*    get_features    declared in dec4-mod.pro */
      get_AC_features
      get_CC_features
      get_PSD_features
      tokenise_PSD(flist,PARAMETER)
      tokenise_AC(flist,PARAMETER)
        .enise_CC(flist,PARAMETER,PARAMETER)
      change_PSD(flist,Xlist)
      Pk_assert(Ylist,real,PARAMETER)
      freq_tok(real,PARAMETER,real)
      power_tok(real,PARAMETER)
      width_tok(real,PARAMETER,PARAMETER)
      lag_tok(real,PARAMETER,real)
      corr_tok(real,real,PARAMETER)
      fl_tok(real,PARAMETER)
      fm_tok(real,PARAMETER)
      append(PARMlist,PARMlist,PARMlist)
      calc_FRF(real,real,real,real)
      rel_speed(real,real,real,PARAMETER).
      garbage
      no_theta
      no_alpha
      no_beta
/*    start_context declared in dec5-mod.pro */
/*    'ear_context declared in dec5-mod.pro */
/*    .ntext_assert(char,CATEGORY,CF) declared in dec5-mod.pro */
      string_to_symbol(CATEGORY,PARAMETER)
      assert_loc_ctx(PARAMETER,CF) /* local context */
      assert_hist_ctx(PARAMETER,CF) /* historical context */
      get_old_hist(PARAMETER,CF,CF) /* old historical context */
      get_att_fac(PARAMETER,CF,CF,CF)
      any_abnormal(PARAMETER,PARAMETER,CF)
      isAnsYes(char)


clauses
    get_features:-
          get_AC_features,
          get_CC_features,
          get_PSD_features,
          !.

    get_features:-
          write("ERROR: failed to extract features."),
          write("\n\npress any key to continue..."),
          readchar(Ans).
```

K7

```prolog
get_AC_features:-
    autocorr(Chan_num),
    channel(Chan_num,Chan,_),
    writef("\nConsulting ILS for Autocorrelation \ndata from channel %s",Chan),
    extractCORR(Chan_num,Feat_List),
    nl,
/*  write(Feat_List),*/
    tokenise_AC(Feat_List,Chan),
/*  readchar(Ans),*/
    fail.


get_AC_features:-
    !.



get_CC_features:-
    crosscorr(Chan_num1,Chan_num2,File),
    channel(Chan_num1,Chan1,_),
    channel(Chan_num2,Chan2,_),
    writef("\nConsulting ILS for Crosscorrelation \ndata from channels %s and %s",Chan1,Chan2),
    extractCORR(File,Feat_List),
/*  write(Feat_List),*/
    tokenise_CC(Feat_List,Chan1,Chan2),
/*  readchar(Ans),*/
    fail.


get_CC_features:-
    !.


get_PSD_features:-
    channel(X,Chan,_),
    writef("\nConsulting ILS for PSD data from \nchannel %s",Chan),
    extractPSD(X,Feat_List),
    nl,
/*  write(Feat_List),*/
    write("\n\nTokenising PSD data....\n"),
    tokenise_PSD(Feat_List,Chan),
    fail.


get_PSD_features:-
    garbage,
    no_theta,
    no_alpha,
    no_beta,
/*  readchar(Ans),*/
    !.




tokenise_AC([_,First_lag,First_mag,Lag,Corr],Chan):-
    write("\nTokenising Autocorrelation\ndata...."),
    lag_tok(Lag,Speed,FRF),
    corr_tok(Lag,Corr,Cycles),
    fl_tok(First_lag,Phase),
    fm_tok(First_mag,Strength),
    append(["ac_feature",Chan],[Phase,Strength,Speed,Cycles],Tfeat),
    nl,
/*  write(Tfeat),*/
    assert(cl(Tfeat,[FRF],[])),
    !.


tokenise_AC(_,Chan):-
    writef("\nWARNING: failed to extract AC featuresfrom channel %s.",Chan).
```

```prolog
tokenise_CC([_,First_lag,First_mag,Lag,Corr],Chan1,Chan2):-
    write('\nTokenising Crosscorrelation\ndata....'),
    lag_tok(Lag,Speed,FRF),
    corr_tok(Lag,Corr,Strength),
    fl_tok(First_lag,Phase),
    fm_tok(First_mag,Similar),
    concat(Chan1,':',Label1),
    concat(Label1,Chan2,Label2),
    append(['cc_feature',Label2],[Phase,Similar,Speed,Strength],Tfeat),
    nl,
/*  write(Tfeat),*/
    assert(cl(Tfeat,[FRF],[])),
    !.


tokenise_CC(_,Chan1,Chan2):-
    writef('\nWARNING: failed to extract CC featuresfrom channel %s:%s.',Chan1,Chan2).




tokenise_PSD([],_):-
    !.


tokenise_PSD([Pk,A,B,C|L],Chan):-
    findall(Peak,change_PSD([A,B,C],Peak),Ylist),
    Pk_assert(Ylist,Pk,Chan),
    tokenise_PSD(L,Chan),
    !.


tokenise_PSD(_,Chan):-
    writef('\nWARNING: failed to extract PSD features\nfrom channel %s.',Chan).




Pk_assert([],_,_):-
    !.


Pk_assert([[p(B),p(A),p(S),l(FRFlist)]|Rest_Pks],Pk,Chan):-
    peak(Pk,Pk_sym),
    append(['f_feature',Chan,Pk_sym],[B,A,S],Feat),
    nl,
/*  write(Feat),*/
    nl,
/*  write(FRFlist),*/
    assert(cl(Feat,FRFlist,[])),
    Pk_assert(Rest_Pks,Pk,Chan),
    !.


/* change_PSD is a non deterministic predicate used in findall */
change_PSD([Freq,Power,Width],[p(Band),p(Amp),p(Speed),l([FRF])]):-
    freq_tok(Freq,Band,FRF),
    power_tok(Power,Amp),
    width_tok(Freq,Band,Speed).



/* freq_tok is a non deterministic predicate used to identify multiple bands */
/* for the same PSD peak... therefore there is no need for a cut. */
freq_tok(Freq,Band,FRF):-
    freq_bounds(Band,Lbound,Ubound),
    Freq > =Lbound,
```

```prolog
                    Freq < =Ubound,
                    calc_FRF(Lbound,Ubound,Freq,FRF).




power_tok(Power,Amp):-
        Power_bounds(Lbound,Amp),
        Power > =Lbound,
        !.




width_tok(Freq,Band,Speed):-
        freq_bounds(Band,Lbound,Ubound),
        rel_speed(Freq,Lbound,Ubound,Speed),
        !.



  ᵤ_tok(0,none_speed,1.0):-
        !.

lag_tok(Lag,Speed,FRF):-
        Freq =(256/Lag)*4,
        freq_tok(Freq,delta,FRF),
        freq_bounds(delta,Lbound,Ubound),
        rel_speed(Freq,Lbound,Ubound,Speed),
        !.

lag_tok(_,non_delta,1.0).




corr_tok(Lag,Corr,"high_cycles"):-
/*      Lag*(Corr-1)> =256,*/
        Lag*Corr > =256,
        !.

corr_tok(_,_,"low_cycles").


fl_tok(First_lag,"none_phase"):-
        First_lag < =10,
        !.



fl_tok(First_lag,"delay_phase").



fm_tok(First_mag,Strength):-
        similar_bounds(Lbound,Strength),
        First_mag > =Lbound,
        !.


append([],L,L):-
        !.

append([X|L1],L2,[X|L3]):-
```

```prolog
        append(L1,L2,L3),
        !.




calc_FRF(Lbound,Ubound,Freq,FRF):-
        Freq > = Lbound,
        Freq < = Lbound + 4,
        FRF = (1-((Lbound + 1)-Freq))/4,
        !.

calc_FRF(Lbound,Ubound,Freq,FRF):-
        Freq > = Ubound-4,
        Freq < = Ubound,
        FRF = (Ubound-Freq)/4,
        !.

calc_FRF(_,_,_,1.0).




rel_speed(Freq,Lbound,Ubound,"high_speed"):-
        Freq > = Ubound-4,  /*((Ubound-Lbound)/3),*/
        !.

rel_speed(Freq,Lbound,Ubound,"med_speed"):-
        Freq > = Lbound + 4, /*((Ubound-Lbound)/3),*/
        !.

rel_speed(_,_,_,"low_speed"):-
        !.


garbage:-          ·
        retract(cl([f_feature,_,_,_,_,_],[0],_)),
        fail.

garbage:-
        !.

no_theta:-
        retractall(cl([f_feature,_,_,theta,_,_],_,_)),
        fail.

no_theta:-
        !.

no_alpha:-
        retractall(cl([f_feature,_,_,alpha,_,_],_,_)),
        fail.

no_alpha:-
        !.

no_beta:-
        retractall(cl([f_feature,_,_,beta,_,_],_,_)),
        fail.

no_beta:-
        !.

/*--------------------------------------------------------------------*/
```

```prolog
/* context clauses. These clauses are used to assert contextual features. */
/* in clausal form, into the knowledge base. */

        clear_context:-
            retractall(cl([c_feature,_,_],_,_)).

        start_context:-
            assert(cl(["c_feature","class_totals"],[0],[])).

        context_assert(Ans,Classification,MB):-
            isAnsYes(Ans),
            string_to_symbol(Classification,Symbol),
            assert_loc_ctx(Symbol,MB),
            assert_hist_ctx(Symbol,MB),
            !.

        context_assert(_,_,_):-
            retractall(cl([c_feature,local,_],_,_)).

        isAnsYes(Ans):-
            Ans = 'y',
            !.

        isAnsYes(Ans):-
            Ans = 'Y',
            !.


        string_to_symbol(Classification,Symbol):-
            fronttoken(Classification,Symbol,_Class),
            !.


        assert_loc_ctx(Symbol,MB):-
            retractall(cl([c_feature,local,_],_,_)),
            assert(cl(["c_feature","local",Symbol],[MB],[])).


        assert_hist_ctx(Symbol_in,MB):-
            any_abnormal(Symbol_in,Symbol_out,Proportion),
            get_old_hist(Symbol_out,Old_MB,Old_num),
            cl([c_feature,class_totals],[Total],[]),
            New_num = Old_num + 1,
            New_tot = Total + 1,
            Fraction = Proportion*New_num,
            get_an_fac('+',Fraction,New_tot,AF),
            measure_of_belief('+',AF,MB,Old_MB,New_belief),
            retractall(cl([c_feature,historical,Symbol_out],[_,_],[])),
            assert(cl([c_feature,historical,Symbol_out],[New_belief,New_num],[])),
            retractall(cl([c_feature,class_totals],[_],[])),
            assert(cl([c_feature,class_totals],[New_tot],[])).
            !.

        assert_hist_ctx(_,MB):-
            get_old_hist(abnormal,Old_MB,Old_num),
            Old_num < >0,
            cl([c_feature,class_totals],[Total],[]),
            New_tot = Total + 1,
            get_an_fac('-',Old_num,New_tot,AF),
            measure_of_belief('-',AF,MB,Old_MB,New_belief),
            retractall(cl([c_feature,historical,abnormal],[_,_],[])),
            assert(cl([c_feature,historical,abnormal],[New_belief,Old_num],[])),
            retractall(cl([c_feature,class_totals],[_],[])),
            assert(cl([c_feature,class_totals],[New_tot],[])).
```

```prolog
        !.

assert_hist_ctx(_,_):-
        cl([c_feature,class_totals],[Total],[]),
        New_tot = Total + 1,
        retractall(cl([c_feature,class_totals],[_],[])),
        assert(cl([c_feature,class_totals],[New_tot],[])).


any_abnormal(mainly_OA,abnormal,0.2):-
        !.

any_abnormal(detectable_OA,abnormal,0.4):-
        !.

any_abnormal(mainly_abnormal,abnormal,0.6):-
        !.

any_abnormal(abnormal,abnormal,0.8).


get_old_hist(Symbol,MB,Num):-
        cl([c_feature,historical,Symbol],[MB,Num],[]),
        !.

get_old_hist(_,0,0).


get_att_fac("+",Num,Total,AF):-
        AF = Num/Total,
        !.

get_att_fac("-",Num,Total,AF):-
        AF = (Total-Num)/Total.



/*----------------------------------------------------------------*/
```

새기

```
/* File: inf6-mod.pro */

project "EEG2"
include "dec6-mod.pro"
include "uni-mod.pro" /* unification module for prolog interpreter */


/* This module contains the inference mechanism for the system. The inference
   system works by negotiating a tree type decision structure...



start state              wave
                          |
                          [
                |------------------------------------|
level 1         cerebral                    artefactual
                    |                             |
                |----|-----[        |----------|----------|----------[
                                     |          |          |
level 2    FUTURE EXPANSION        other  perspiration movement eye movement
                                                            |
                                                        |---|----|
l.     J(goal)                                       vem hem blink



The ultimate decision, of whether or not to correct, is at the goal level.
This decision is, however, influenced and dictated by the result of
negotiating levels 1, 2 and 3 of the decision tree. inf2-mod.pro also incorporates
uncertainty in it's reasoning, using similar CERTAINTY FACTORS as those used
in MYCIN. */



domains
     Mygoal = string
     NT = INTEGER

predicates

/* inference predicates */

/*      cision  calling predicate for reasoning - declared in dec5-mod.pro */
/*      infer(HISTORY,CATEGORY,CF)  predicate to explore decision tree - declared in dec5-mod.pro */
        combine_certainty(CF,CFS,CF,CF) /*combine rule,conditions and previous certaintys*/
        test_conds(Rule_No,HISTORY,CONDITIONS,CFS,CFS) /*find the truth of an individual condition*/
        get_certainty(Cond_No,Cond_No,Rule_No,HISTORY,STRING,NT,CF) /*get certainty from either kb, features or user*/
        enq_user(NT,HISTORY,Rule_No,Cond_No,Cond_No,STRING,CF) /*get user supplied certainty*/
        do_answer(NT,HISTORY,Rule_No,STRING,Cond_No,Cond_No,INTEGER,CF) /*process user certainty*/
/*      measure_of_belief(CF,CF,CF,CF) declared in dec5-mod.pro */
        lib_andcombine(CFS,CF) /*CF = minimum(CFS)*/
        con_andcombine(CFS,CF) /*CF = intersection(CFS)*/
        min_cf(CFS,CF) /*min*/
        min(CFS,CF) /*min*/
        cf_member(CF,CFS) /* to find out if CF is a member of the CFlist */
        assert_frf(PARMlist,CF) /*assert a fuzzy factor*/
        frf_combine(CF,CFS) /*fuzzy factor combination*/
        merge(CFS,CFS) /*produce a list of fuzzy factors from asserted facts*/



/* feature consultation predicates */

        consult_features(Cond_No,Cond_No,Rule_No,HISTORY,STRING,NT,CF) /*find truth of an individual condition*/
                                        /*by invoking rule interpretter*/
```

K15

```prolog
        head(PARMlist,PARMlist,PAIRlist,PAIRlist,INT) /*feature demon rule head*/
        body(Llist,PAIRlist,PAIRlist,INT,INT) /*feature demon rule head*/
        iPred(PARAMETER,PARAMETER,PARAMETER) /*interpreter access to internal predicates*/
        deunify(PARMlist,PAIRlist,PAIRlist) /*reverse acton of unification*/
        deunify_B(PARAMETER,PAIRlist,PAIRlist)


/* explanation predicates */

/*      show_rule(Rule_No,string)  display rule - declared in dec4-mod.pro */
        show_cond(Cond_No,string) /*display conditions*/
        sub_cat(CATEGORY,CATEGORY,CATEGORY)
        show_conditions(CONDITIONS,string)
        add(Cond_No,string,string)
/*      report(HISTORY,string)  display history of reasoning - declared in dec4-mod.pro */
/*      log_conds - save true conditions to temp log  - declared in dec6-mod.pro */
/*      log_results(char,CATEGORY,CF) - save results to temp log - declared in dec6-mod.pro*/
/*      trans_tru2log(char) - transfer temp log to prolog.log - declared in dec6-mod.pro */

clauses

/* ir"  'ence clauses */

        decision:-
            clearwindow,
            Mygoal = "segment1",
            nl,
            nl,
            infer([],Mygoal,0),
            !.

        decision:-
            write("\n\nThe knowledge base does not identify this...."),
            evalans('n',[],"",0).


        infer(HISTORY,Mygoal,Prev_CF):-
            not(rule(_,_,Mygoal,_,_)),
            !,
            data(Dname),
            seg(Sname),
            writef("\n%s %s contains %s.",Dname,Sname,Mygoal),
            writef("\nThe measure of belief = "),
            writef("%1.2f",Prev_CF),nl,nl,
            write("Is this correct? (enter y,n or ?)"),
            readchar(Ans),
            evalans(Ans,HISTORY,Mygoal,Prev_CF),
            process_window,
            writef("\n\nPlease wait a moment \nwhile I record this session..."),
            context_assert(Ans,Mygoal,Prev_CF),
            log_conds,
            log_results(Ans,Mygoal,Prev_CF),
            removewindow.

        infer(HISTORY,Mygoal,Prev_CF ):-
            rule(Rule_No,Rule_CF,Mygoal,NY,COND),
/*          write("\nMygoal is ",Mygoal," and NY is ",NY),*/
            writef("\n\nRule %d\n",Rule_No),
            test_conds(Rule_No,HISTORY, COND,[1],OUT_CFS),
            combine_certainty(Rule_CF,OUT_CFS,Prev_CF,New_CF),
            infer([Rule_No|HISTORY],NY,New_CF).


        test_conds(Rule_No,HISTORY,[999],IN_CFS,OUT_CFS):-
```

```prolog
        OUT_CFS =IN_CFS.
        !.


    test_conds(Rule_No,HISTORY,[Cond_No|REST],IN_CFS,OUT_CFS):-
/*        write("\ntest_conds 1").*/
        yes(Cond_No,CF),
        !,
        test_conds(Rule_No, HISTORY, REST,[CF|IN_CFS],OUT_CFS),
        !.

    test_conds(_,_,[Cond_No|_],IN_CFS,OUT_CFS):-
/*        write("\ntest_conds 2").*/
        no(Cond_No,_).
        OUT_CFS =IN_CFS.
        !.
        fail.

    test_conds(Rule_No,HISTORY,[Cond_No|REST],IN_CFS,OUT_CFS):-
/*        write("\ntest_conds 3").*/
        tex_cond(Cond_No,NCOND),
        fronttoken(NCOND,"not",_COND).
        frontchar(_COND,_,COND).
        tex_cond(Cond_No1,COND),
        get_certainty(Cond_No1,Cond_No,Rule_No,HISTORY,COND,1,CF).
        !,
        test_conds(Rule_No, HISTORY, REST,[CF|IN_CFS],OUT_CFS).
        !.

    test_conds(_,_,[Cond_No|_],IN_CFS,OUT_CFS):-
/*        write("\ntest_conds 4").*/
        tex_cond(Cond_No,NCOND).
        fronttoken(NCOND,"not",_COND).
        frontchar(_COND,_,COND).
        tex_cond(Cond_No1,COND).
        yes(Cond_No1,_).
        OUT_CFS =IN_CFS.
        !.
        fail.

    test_conds(Rule_No,HISTORY,[Cond_No|REST],IN_CFS,OUT_CFS):-
/*        write("\ntest_conds 5").*/
        tex_cond(Cond_No,COND).
/*        not(fronttoken(COND,"not",_COND)).*/
        concat("not ",COND,NCOND),
        tex_cond(Cond_No1,NCOND),
        get_certainty(Cond_No1,Cond_No,Rule_No,HISTORY,COND,0,CF).
        !,
        test_conds(Rule_No, HISTORY, REST,[CF|IN_CFS],OUT_CFS).
        !.

    test_conds(_,_,[Cond_No|_],IN_CFS,OUT_CFS):-
/*        write("\ntest_conds 6").*/
        tex_cond(Cond_No,COND).
/*        not(fronttoken(COND,"not",_COND)).*/
        concat("not ",COND,NCOND),
        tex_cond(Cond_No1,NCOND),
        yes(Cond_No1,_).
        OUT_CFS =IN_CFS.
        !.
        fail.


    test_conds(Rule_No,HISTORY,[Cond_No|REST],IN_CFS,OUT_CFS):-
```

```prolog
/*      write("\ntest_conds 7"),*/
        tex_cond(Cond_No,TEXT),
        enq_user(0,HISTORY,Rule_No,0,Cond_No,TEXT,CF),
        test_conds(Rule_No, HISTORY, REST,[CF|IN_CFS],OUT_CFS),
        !.



test_conds(_,_,[Cond_No|_],IN_CFS,OUT_CFS):-
        OUT_CFS=IN_CFS,
        !.



test_conds(_,_,[],IN_CFS,OUT_CFS):-
        OUT_CFS=IN_CFS.


get_certainty(Cond_No1,Cond_No,_,_,_,_,CF):-
        no(Cond_No1,CF),
        assert(yes(Cond_No,CF)),
        !.


get_certainty(Cond_No1,Cond_No,_,_,_,_,_):-
        yes(Cond_No1,CF),
        assert(no(Cond_No,CF)),
        !,
        fail.


get_certainty(Cond_No1,Cond_No,Rule_No,HISTORY,COND,NT,CF):-
        tex_cond(Cond_No,Condit),
        writef("\nCondition %d",Cond_No),
        consult_features(Cond_No1,Cond_No,Rule_No,HISTORY,COND,NT,CF),
/*      writef("\nCF = %f\n",CF),*/
        !.

/*------------------------------------------------------------*/

consult_features(Cond_No1,Cond_No,_,_,_,_,CF):-
        num_cond(Cond_No,Feature),
        head(Feature,Feature,[],_,0),
        frf_combine(CF,CFS),
/*      writef("\nmerged frfs = "),
        write(CFS),*/
/*      writef("\nmin frf = %f",CF),*/
/*      retractall(larg(_,_)),*/
        retractall(eeg_peak(_,_,_)),
        retractall(sym_peak(_,_)),
        assert(no(Cond_No1,CF)),
        assert(yes(Cond_No,CF)),
        !.

consult_features(Cond_No1,Cond_No,_,_,_,_,CF):-
        num_cond(Cond_No1,Feature),
        head(Feature,Feature,[],_,0),
        frf_combine(CF,CFS),
/*      writef("\nmerged frfs = "),
        write(CFS),
        writef("\nmin frf = %f",CF),*/
/*      retractall(larg(_,_)),*/
        retractall(eeg_peak(_,_,_)),
        retractall(sym_peak(_,_)),
        assert(no(Cond_No,CF)),
```

K 18

```prolog
        assert(yes(Cond_No1,CF)),
        !,
        fail.

consult_features(Cond_No1,Cond_No,_,_,_,_,CF):-
        num_cond(Cond_No,_),
        frf_combine(CF,CFS),
/*      writef("\nmerged frfs = "),
        write(CFS),
        writef("\nmin frf = %f",CF).*/
/*      retractall(larg(_,_)).*/
        retractall(eeg_peak(_,_,_)),
        retractall(sym_peak(_,_)),
        assert(no(Cond_No,CF)),
        assert(yes(Cond_No1,CF)),
        !,
        fail.

consult_features(Cond_No1,Cond_No,_,_,_,_,CF):-
        num_cond(Cond_No1,_),
        frf_combine(CF,CFS),
/*      writef("\nmerged frfs = "),
        write(CFS),
        writef("\nmin frf = %f",CF).*/
/*      retractall(larg(_,_)).*/
        retractall(eeg_peak(_,_,_)),
        retractall(sym_peak(_,_)),
        assert(no(Cond_No1,CF)),
        assert(yes(Cond_No,CF)),
        !.

consult_features(Cond_No1,Cond_No,Rule_No,HISTORY,COND,NT,CF):-
        enq_user(NT,HISTORY,Rule_No,Cond_No1,Cond_No,COND,CF),
        !.




 ·ad(Scl,Acl,IN,OUT,NUM):-
        Acl =["int_pred",O,P1,P2],
        iPred(O,P1,P2),
/*      unify(Scl,Acl,["int_pred",O,_,P2],IN,OUT,NUM,0).*/
        unify(Scl,Fcl,Acl,IN,OUT,NUM,0).


head(_,["int_pred",_,_,_],_,_,_):-
        !,
        fail.

head(Scl,Acl,IN,OUT,NUM):-
        Scl =["deunify"|List],
        deunify(List,IN,OUT),
        !.

head(Scl,Acl,IN,OUT,NUM):-
        cl(Fcl,FRFlist,[]),
        FRFlist =[FRF|Rest],
        unify(Scl,Fcl,Acl,IN,OUT,NUM,0),
        assert_frf(Acl,FRF).
```

```prolog
head(Scl,Acl,IN,OUT,NUM):-
    getbacktrack(BTOP),
    cl(Fcl,[],[H|T]),
    unify(Scl,Fcl,Acl,IN,TBLA,NUM,0),
    body([H|T],TBLA,OUT,NUM,BTOP).




body([],TBL,TBL,_,_).

body([Scl|_],IN,IN,NUM,BTOP):-
    Scl=["int_pred","cutback",_,_],
    cutbacktrack(BTOP).



body([Scl|Srest],IN,OUT,NUM,BTOP):-
    unify(Scl,Scl,Acl,IN,TBLA,NUM,1),
    head(Scl,Acl,TBLA,TBLB,NUM),
    body(Srest,TBLB,OUT,NUM,BTOP),
    !.




deunify([],IN,IN).

deunify([H|T],IN,OUT):-
    deunify_B(H,IN,List),
    deunify(T,List,OUT).




deunify_B(A,[],[]).

    .unify_B(A,[pair(A,Term,NUM)|T1],[pair(A,A,NUM)|T2]):-
        !,
        deunify_B(A,T1,T2).


deunify_B(A,[PAIR|T1],[PAIR|T2]):-
    deunify_B(A,T1,T2).
```

/*···─────────·····································*/
/* INTERNAL PREDICATE CALLS: Clausal conditions cannot access the standard */
/* Prolog predicates. This is ovecome by the use of the int_pred predicate */
/* which calls iPred to execute the internal predicate.   */

```prolog
iPred("channel",X,Y):-
    channel(_,X,Y).


iPred("lr_sym",X,Y):-
    lr_sym(X,Y),
    !.
```

```prolog
iPred("lr_sym",X,Y):-
    lr_sym(Y,X),
    !.


iPred(" < > ",X,Y):-
    !,
    X < > Y,
    !.


iPred(" = ",X,Y):-
    !,
    X = Y,
    !.


iPred("fail",_,_):-
    !,
    fail,
    !.

  red("true",_,_):-
    !,
    true,
    !.

iPred("not_peak",Mag,Freq):-
    !,
    not(ecg_peak("delta",Mag,Freq)),
    !.


iPred("assert_peak",Mag,Freq):-
    !,
    assert(ecg_peak("delta",Mag,Freq)),
    !.


iPred("not_sym_peak",M,_):-
    !,
    not(sym_peak("delta",M)),
    !.


iPred("ass_sym_peak",M,_):-
    !,
    assert(sym_peak("delta",M)),
    !.


iPred(" > ",X,Y):-
    !,
    X > Y,
    !.



iPred("read_lrg",C,V):-
    larg(chan,C),
    larg(amp,V),
    !.


ipred("ass_lrg",X,A):-
    retractall(larg(_,_)),
    assert(larg(chan,X)),
    assert(larg(amp,A)),
    !.
```

```prolog
ipred("ass_lrg_test",X,A):-
    assert(larg(chan,X)),
    assert(larg(amp,A)),
    !.

ipred("nt_adj",A,B):-
    not(adjacent(A,B)),
    !.
```

```
/*---------------------------------------------------------------*/
/* user enquiry predicate clauses. Should the feature database not contain*/
/* the information necessary to satisfy the rule conditions, then the user*/
/* is requested for the information. */
```

```prolog
enq_user(NT,HISTORY,Rule_No,Cond_No1,Cond_No,TEXT,CF):-
    write("Is it true that ",TEXT,": "),
    question_window(CHOICE),
    do_answer(NT,HISTORY,Rule_No,TEXT,Cond_No1,Cond_No,CHOICE,CF).


_answer(_,_,_,_,_,_,0,0):-
    erase,
/*  clear,*/
    removewindow,
    run.


do_answer(0,_,_,_,Cond_No1,Cond_No,CHOICE,CF):-
    CHOICE<7,
    CF=(6-CHOICE)/5,
    assert(yes(Cond_No,CF)),
    assert(no(Cond_No1,CF)),
    shiftwindow(3),
    write("\n + %g",CF),
    nl,
    !.


_answer(0,_,_,_,Cond_No1,Cond_No,CHOICE,_):-
    CHOICE>6,
    CHOICE<12,
    CF=(CHOICE-6)/5,
    assert(no(Cond_No,CF)),
    assert(yes(Cond_No1,CF)),
    shiftwindow(3),
    write("\n- %g",CF),
    nl,
    fail.

do_answer(1,_,_,_,Cond_No1,Cond_No,CHOICE,_):-
    CHOICE<7,
    CF=(6-CHOICE)/5,
    assert(no(Cond_No,CF)),
    assert(yes(Cond_No1,CF)),
    shiftwindow(3),
    write("\n + %g",CF),
    nl,
    fail.


do_answer(1,_,_,_,Cond_No1,Cond_No,CHOICE,CF):-
    CHOICE>6,
```

```prolog
        CHOICE < 12.
        CF = (CHOICE-6)/5.
        assert(yes(Cond_No,CF)),
        assert(no(Cond_No1,CF)),
        shiftwindow(3),
        writef("\n-%g",CF),
        nl,
        !.


do_answer(NT,HISTORY,Rule_No,TEXT,Cond_No1,Cond_No,12,CF):-
        !,
        explain_window,
        rule( Rule_No,_, Mygoal1, Mygoal2, _ ),
        sub_cat(Mygoal1,Mygoal2,Lstr),
        concat("It is necessary to show that: ",Lstr,Lstr1),
        concat(Lstr1,"\n\nBy using rule number ",Ls1),
        str_int(Str_num,Rule_No),
        concat(Ls1,Str_num,Ans),
        show_rule(Rule_No,Lls1),
        concat(Ans,Lls1,Ans1),
        report(HISTORY,Sng),
        concat(Ans1,Sng,Answ),
        display(Answ),
        removewindow,
        question_window(CHOICE),
        do_answer(NT,HISTORY,Rule_No,TEXT,Cond_No1,Cond_No,CHOICE,CF).


/*---------------------------------------------------------------------*/
/* dealing with uncertainty is a necessary part of this expert system. The  */
/* certainty of a final decision is derived from the combination of previous*/
/* certainties. Each atceedent of a rule will have a certainty or fuzzy    */
/* reliability representing uncertainty in its truth. when more than one    */
/* anteceeden
```

t exists the overall cerainty of the ateceedents is taken as */
/* the mean of the 'liberal and' and 'conservative and'. the liberal and is */
/* calculated as the maximum intersection of the certainties, ie when the   */
/* certainties are maximally dependent on one another. The conservative and */
/* is calculated as the minimum intersection of the certainties, ie when the*/
/* certainties are maximally independent of one another. The combined   */
/* anteceedent certainties are then attenuated by the rule certainty which */
/* reflects the degree of confidence that the expert has in the rule   */
/* indicating that the consequent is true given the ateceedents. A decision */
/* is represented by a conjunction of several such rules, where progressive */
/* rules form a subset of the preceeding one. The overall certainty is   */
/* therefore calculated as the minimum of the conjunctive rules. */


```prolog
        combine_certainty(Rule_CF,Certs,Prev_CF,New_CF):-
            write('\nCerts are '),
            write(Certs),
            lib_andcombine(Certs,Cond_CF1),
/*          con_andcombine(Certs,Cond_CF2).*/
/*          Cond_CF=(Cond_CF1+Cond_CF2)/2.0,*/
/*          writef('\nRule_CF=%g\nCond_CF1=%g\nPrev_CF=%g',Rule_CF,Cond_CF1,Prev_CF),*/
            measure_of_belief('+',Rule_CF,Cond_CF1,Prev_CF,New_CF).
/*          writef('\ncurrent measure of belief=%g',New_CF).*/


        measure_of_belief('+',Rule_CF,Cond_CF,Prev_CF,MB):-
            ATT_CF=Rule_CF*Cond_CF,
            MB=Prev_CF+((1-Prev_CF)*ATT_CF),
/*          lib_andcombine([ATT_CF,Prev_CF],New_CF).*/
            !.


        measure_of_belief('-',Rule_CF,Cond_CF,Prev_CF,MB):-
            ATT_CF=Rule_CF*Cond_CF,
            MB=Prev_CF-(Prev_CF*ATT_CF).


        lib_andcombine(CFS,CF):-
            assert(minimum(1)),
            min(CFS,CF),
/*          min_cf(CFS,CF).*/
            retractall(minimum(_)).

        min([Num|[]],Num):-
            minimum(Smallest),
            Num<=Smallest,
            !.

        min([Num|[]],Smallest):-
            minimum(Smallest),
            Num>Smallest,
            !.

        min([Num|Rest],Min):-
            minimum(Smallest),
            Num<=Smallest,
            retractall(minimum(_)),
            assert(minimum(Num)),
            min(Rest,Min),
            !.

        min([Num|Rest],Min):-
            min(Rest,Min),
            !.
```

K24

```prolog
min_cf(CFS,0.25):-
    cf_member(0.25,CFS).
    !.

min_cf(CFS,0.5):-                           .
    cf_member(0.5,CFS).
    !.

min_cf(CFS,0.75):-
    cf_member(0.75,CFS).
    !.

min_cf(CFS,1):-
    !.

cf_member(X,[X|_]):-
    !.

cf_member(X,[_|Y]):-
    cf_member(X,Y).


con_andcombine([X],X).

con_andcombine([X|L],0):-
    con_andcombine(L,X2).
    X+X2 < 1.0.

con_andcombine([X|L],Tot):-
    con_andcombine(L,X2).
    X3 = X + X2.
    not(X3 < 1.0).
    Tot = X + X2-1.0.


assert_frf(_,FRF):-
    assert(frf(FRF)).
    !.

  sert_frf(_,_).


frf_combine(CF,Out):-
    merge([1],Out).
    lib_andcombine(Out,CF).

merge(In,Out):-
    frf(Num).
    retract(frf(Num)).
    merge([Num|In],Out).
    !.

merge(Out,Out).
```

```
/*................................................................*/
```

```
/* explanation is necessary as to why a particular fact is being sought, or */
/* how a particual decision was arived at. the following clauses allow      */
/* interaction between the user and the inference mechanism.*/

/* explanation clauses */
```

```prolog
show_rule(Rule_No,Strg):-
    rule( Rule_No,CF, Mygoal1, Mygoal2, CONDINGELSER),
    str_int(Rule_No_str,Rule_No),
    str_real(CF_str,CF),
    concat("\n Rule ",Rule_No_str,Ans),
    concat(Ans,": ",Ans1),
    sub_cat(Mygoal1,Mygoal2,Lstr),
    concat(Ans1,Lstr,Ans2),
    concat(Ans2,"(",Ans3),
    concat(Ans3,CF_str,Ans4),
    concat(Ans4,")\n    if ",Ans5),
    reverse(CONDINGELSER,CONILS),
    show_conditions(CONILS,Con),
    concat(Ans5,Con,Strg).



show_conditions([],"").

show_conditions([COND],Ans):-
    show_cond(COND,Ans),
    !.

show_conditions([COND|REST],Ans):-
    show_cond(COND,Text),
    concat("\n    and ",Text,Nstr),
    show_conditions(REST,Next_ans),
    concat(Next_ans,Nstr,Ans).


show_cond(999,"ALWAYS TRUE"):-
    !.


show_cond(COND,Txt):-  /*change Txt to TEXT*/
    tex_cond(COND,TEXT),
    add(COND,TEXT,Txt),
    !.    /*remove*/



add(COND,TEXT,Txt) :-
    yes(COND,CF),
    str_real(CF_string,CF),
    concat(TEXT,"(+",Txt1),
    concat(Txt1,CF_string,Txt2),
    concat(Txt2,")",Txt),!.

add(COND,TEXT,Txt) :-
    no(COND,CF),
    str_real(CF_string,CF),
    concat(TEXT,"(-",Txt1),
    concat(Txt1,CF_string,Txt2),
    concat(Txt2,")",Txt),!.

add(COND,TEXT,TEXT) :-!.

sub_cat(Mygoal1,Mygoal2,Lstr):-
    concat(Mygoal1," contains ",Str),
    concat(Str,Mygoal2,Lstr).

report([],"").
```

```prolog
report([Rule_No|REST],Strg) :-
    rule( Rule_No._. Mygoal1. Mygoal2. _),
    sub_cat(Mygoal1,Mygoal2,Lstr),
    concat("\n\nIt has previously been shown that: ",Lstr,L1),
    concat(L1,"\nBy using rule number ",L2),
    str_int(Str_Rule_No,Rule_No),
    concat(L2,Str_Rule_No,L3),
    concat(L3,":\n ",L4),
    show_rule(Rule_No,Str),
    concat(L4,Str,L5),
    report(REST,Next_strg),
    concat(L5,Next_strg,Strg).
```

/*--*/ /* File: uni-mod.pro */

/* This module contains the additional unification algorithm predicates   */
/* necessary for the conditions to contain, as arguments, other predicate */
/* calls. The following binding tables are necessary to enable the   */
/* predicates to accept variables. */


```prolog
predicates
    ify(PARMlist,PARMlist,PARMlist,PAIRlist,PAIRlist,INT,INT)
/*  unify_A(PARMlist,PARMlist,PARMlist,PAIRlist,PAIRlist,INT,INT) */
    unify_B(PARMlist,PARMlist,PARMlist,PAIRlist,PAIRlist,INT,INT)
        unify_C(PARAMETER,PARAMETER,PARAMETER,INT,PAIRlist,PAIRlist,INT)
            unify_D(PARAMETER,PARAMETER,PARAMETER,INT,PAIRlist,PAIRlist)
                unify_E(PARAMETER,PARAMETER,PARAMETER,INT,PAIRlist,PAIRlist)
                unify_F(PARAMETER,PARAMETER,PARAMETER,INT,PAIRlist,PAIRlist)
            unify_G(PARAMETER,PARAMETER,INT,PAIRlist,PAIRlist)
            unify_H(PARAMETER,PARAMETER,PARAMETER,INT,PAIRlist,PAIRlist,INT)
            unify_J(PARAMETER,PARAMETER,PARAMETER,INT,PAIRlist,PAIRlist)

    variable(PARAMETER)
    capital(PARAMETER)
    member(PAIR,PAIRlist)


clauses

/*  ..nify([Head|Slist],[Head|Flist],[Head|Alist],IN,IN,NUM,0):-
        Head = int_pred.
        Slist = ["channel"._._].
        unify_A([Head|Slist],[Head|Flist],[Head|Alist],IN,OUT,NUM,0),
        !.

    unify([Head|Slist],[Head|Flist],[Head|Alist],IN,IN,NUM,0):-
        Head = int_pred.
        Slist = [" = "._._].
        unify_A([Head|Slist],[Head|Flist],[Head|Alist],IN,OUT,NUM,0),
        !.

    unify([Head|Slist],[Head|Flist],[Head|Alist],IN,IN,NUM,0):-
        Head = int_pred.
        Slist = ["read_Irg"._._].
        unify_A([Head|Slist],[Head|Flist],[Head|Alist],IN,OUT,NUM,0),
        !.

    unify([Head|Slist],[Head|Flist],[Head|Alist],IN,IN,NUM,0):-
        Head = int_pred.
        !.

*/
```

```prolog
unify([Head|Slist],[Head|Flist],[Head|Alist],IN,OUT,NUM,A):-
    unify_B(Slist,Flist,Alist,IN,OUT,NUM,A).



unify_B([],[],[],TBL,TBL,_,_):-
    !.

unify_B([S|Srest],[F|Frest],[A|Arest],IN,OUT,NUM,X):-
    unify_C(S,F,A,NUM,IN,OUTA,X),
    unify_B(Srest,Frest,Arest,OUTA,OUT,NUM,X).



unify_C('_',F,A,NUM,IN,IN,_):-
    !.

unify_C(S,Fbv,A,NUM,IN,OUT,_):-
    bound(Fbv),
    variable(Fbv),
    unify_D(S,Fbv,A,NUM,IN,OUT),
    !.

unify_C(_,F,Abv,NUM,IN,OUT,_):-
    bound(Abv),
    variable(Abv),
    unify_G(F,Abv,NUM,IN,OUT),
    !.

unify_C(S,F,Af,NUM,IN,OUT,A):-
    free(Af),
    unify_H(S,F,Af,NUM,IN,OUT,A),
    !.

unify_C(S,Ff,A,NUM,IN,OUT,_):-
    free(Ff),
    unify_J(S,Ff,A,NUM,IN,OUT),
    !.

 .ify_C(_,F,F,_,TBL,TBL,_).



unify_D(S,Fbv,Ab,NUM,IN,OUT):-
    bound(Ab),
    unify_E(S,Fbv,Ab,NUM,IN,OUT),
    !.

unify_D(S,Fbv,Af,NUM,IN,OUT):-
    free(Af),
    unify_F(S,Fbv,Af,NUM,IN,OUT).



unify_E(_,Fbv,Ab,NUM,TBL,TBL):-
    member(pair(Fbv,Ab,NUM),TBL).

unify_E(_,Fbv,Ab,NUM,IN,[pair(Fbv,Ab,NUM)|IN]):-
    not(member(pair(_,Ab,NUM),IN)).
```

```prolog
/* this is to catch variables */
    unify_E(S,Fbv,Ab,NUM,TBL,[pair(Fbv,Ab,NUM)|TBL]):-
        member(pair(S,Ab,NUM),TBL).
/*----------------------------*/




    unify_F(_,Fbv,_,NUM,[],[pair(Fbv,Fbv,NUM)]).

    unify_F(_,Fbv,Af,NUM,[pair(Fbv,Fbv,NUM)|R],[pair(Fbv,Fbv,NUM)|R]).

    unify_F(_,Fbv,Af,NUM,[pair(Fbv,Af,NUM)|R],[pair(Fbv,Af,NUM)|R]).

    unify_F(_,Fbv,Af,NUM,[PAIR|IN],[PAIR|OUT]):-
        unify_F(_,Fbv,Af,NUM,IN,OUT).




/*    ify_G(_,_,_,[],_):-
        !,
        fail.
*/

    unify_G(F,Abv,NUM,[],[pair(Abv,F,NUM)]).

    unify_G(F,Abv,NUM,[pair(Abv,Abv,NUM)|R],[pair(Abv,F,NUM)|R]).

    unify_G(F,Abv,NUM,[PAIR|IN],[PAIR|OUT]):-
        unify_G(F,Abv,NUM,IN,OUT).




    unify_H(S,F,F,_,[],[],_).

    unify_H(S,F,F,_,TBL,TBL,1).

    unify_H(S,F,F,NUM,[pair(S,S,NUM)|R],[pair(S,F,NUM)|R],_).

    unify_H(S,F,Af,NUM,[PAIR|IN],[PAIR|OUT],A):-
        unify_H(S,F,Af,NUM,IN,OUT,A).




    unify_J(S,Ef,Ef,_,[],[]).

    unify_J(S,Ef,A,NUM,[pair(S,S,NUM)|R],[pair(S,A,NUM)|R]).

    unify_J(S,S,A,NUM,[pair(S,A,NUM)|R],[pair(S,A,NUM)|R]).

    unify_J(S,Ef,A,NUM,[PAIR|IN],[PAIR|OUT]):-
        unify_J(S,Ef,A,NUM,IN,OUT).




    variable(String):-
        frontstr(1,String,CHAR,_),
        capital(CHAR).
```

```prolog
capital(Uchar):-
     upper_lower(Uchar,Lchar),
     Uchar < > Lchar.



member(X,[X|_]):-
     !.

member(X,[_|Y]):-
     member(X,Y).
```

/*- ....*/

```
/* File: sup3-mod.pro */

project "ceg2"
include "dec6-mod.pro"

/* This module contains the support predicates for the expert system. ie
   knowledge base updating and listing. */

/*project "EEG"
include "dec-mod.pro" */


predicates

/*   load_kb  - declared in dec4-mod.pro */
/*   save_kb  - declared in dec4-mod.pro */
/*   get_kb   - declared in dec4-mod.pro */
/*   get_data - declared in dec6-mod.pro */
/*   select_segment - declared in dec6_mod.pro */
     select_kb(string)
     select_data(string)
     select_rules(string)
      .opt(INTEGER,string)
/*   list  - declared in dec4-mod.pro */
     llist(HISTORY,string)
/*   edit_kb    - declared in dec4-mod.pro */
     save_y(char,string,string)
/*   erase - declared in dec4-mod.pro */
/*   inspect_log - declared in dec5-mod.pro */
     logopt(INTEGER,string,string)
     log_feats(PARAMETER)

clauses

     load_kb :-
          consult("mont1.kb").


     save_kb :-
          process_window,
          write("Saving scratch Knowledge base. please wait..."),
          save("scratch.kb"),
          clearwindow,
          wait_count(20000),
          wait_count(20000),
          removewindow.


     list :-
          findall(RNO,rule(RNO,_,_,_,_),List),
          llist(List,Str),
          !,
          listopt_window(CHOICE),
          listopt(CHOICE,Str),
          !.


     llist([],"") :-
          !.

     llist([RNO|List],Str):-
          llist(List,Oldstr),
          show_rule(RNO,RNO_Str),
          concat(RNO_Str,Oldstr,Str).
```

```
listopt(1,Str) :-
    writedevice(screen),
    display(Str),!.

listopt(2,Str) :-
    cursor(3,23),
    write('Waiting for printer...'),
    cursor(12,18),
    writedevice(printer),
    write('\27','\15'),
    write(Str),
    writedevice(screen),
    !.

listopt(3,Str) :-
    cursor(3,23),
    write('Writing KB to rules.asc...'),
    openwrite(save_file,'rules.asc'),
    writedevice(save_file),
    write(Str),
    closefile(save_file),
    writedevice(screen),
    !.


edit_kb :-
    select_kb(Kbname),
    file_str(KBname,Data),
    edit(Data,EditedData,'EEG Correction',
    KBname,'To exit the editor press ESC or F10.',1,'help.txt',
    1,0,1,0,_,_),
    clearwindow,
    write('Save Knowledge Base (enter y or n) '),
    readchar(Ans),
    save_y(Ans,EditedData,KBname),
    !.

edit_kb.

save_y('y',Data,KBname):-
    openwrite(save_file,KBname),
    writedevice(save_file),
    write(Data),
    closefile(save_file),!.

save_y(_,_,_).


get_data:-
    select_data(Dname),
    process_window,
    writef('Loading %s\n\nplease wait...',Dname),
    concat('copy c:\\ils\\mils\\',Dname,Dpath),
    concat(Dpath,'\\data.dat',FullDPath),
    system(FullDPath,0,Errlevel),
    no_sys_error(FullDpath,Errlevel),
    retractall(data(_)),
    assert(data(Dname)),
    removewindow,
    !.
```

K 32

```prolog
select_data(Dname):-
    data_select_window(Dpath),
    frontstr(18,Dpath,Path,Dname),
    removewindow.



get_kb:-
    select_kb(KBname),
    process_window,
    writef("Loading Knowledge base,\n\n %s\n\n\nplease wait...",KBname),
    erase,
    consult(KBname),
    removewindow,
    !.

select_kb(KBname):-
    kb_select_window(KBname),
    removewindow,
    !.


 iect_rules(Rname):-
    rules_select_window(Rname),
    removewindow,
    !.



select_segment:-
    data(Dname),
    select_segment_window(Spath),
    frontstr(17,Spath,Path,Sname),
    removewindow,
    process_window,
    writef("Loading data from %s\n\n\nplease wait...",Sname),
    concat("copy c:\\ils\\mils\\",Dname,Spath1),
    concat(Spath1,"\\",Spath2),
    concat(Spath2,Sname,Spath3),
    concat(Spath3,"\\*.*",FullSPath),
    system(FullSPath,0,Errlevel),
    no_sys_error(FullSpath,Errlevel),
    retractall(seg(_)),
    assert(seg(Sname)),
    removewindow,
    !.


erase:-
    retract(_),
    fail.

erase.

inspect_log:-
    file_str("prolog.log",Data),
    logopt_window(CHOICE),
    logopt(CHOICE,Data,"prolog.log"),
    removewindow,
    !.

inspect_log.


inspect_kb:-
```

```prolog
        select_rules(Rname),
        file_str(Rname,Data),
        logopt_window(Choice),
        logopt(Choice,Data,Rname),
        removewindow,
        !.

inspect_kb.


logopt(1,Data,Name):-
        edit(Data,EditedData," SPOOLED LOG ",
        Name,"To exit from viewing press ESC or F10.",1,"help.txt",
        1,0,0,1,_,_),
        !.

logopt(2,Data,Name) :-
        cursor(8,23),
        write("Waiting for printer..."),
        writedevice(printer),
        writef("\n\n%s",Name),
        write(Data),
        writedevice(screen),
        !.


new_conds:-
        system("del conds.tru",0,_),
        openwrite(save_file,"conds.tru"),
        closefile(save_file).


new_data_conds:-
        data(Dname),
        seg(Sname),
        openappend(save_file,"conds.tru"),
        writedevice(save_file),
        writef("\n%s %s\n",Dname,Sname),
        writef("List of extracted features\n = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =\n\n"),
        closefile(save_file),
        writedevice(screen),
        !.


log_features:-
        log_feats("c_feature"),
        log_feats("t_feature"),
        log_feats("f_feature").

log_feats(Type):-
        cl(Feature,[FRF|_],_),
        Feature=[Type|_],
        openappend(save_file,"conds.tru"),
        writedevice(save_file),
        nl,
        write(Feature),
        writef(" %f",FRF),
        closefile(save_file),
        fail.

log_feats(_).


log_conds:-
```

```
                    yes(Cnum,Ccert),
                    tex_cond(Cnum,Ctext),
                    openappend(save_file,"conds.tru").
                    writedevice(save_file).
                    nl,
                    write(Ctext),
                    writef(" %f",Ccert),
                    closefile(save_file).
                    fail.

log_conds.



log_results(Agree,Class,Cert):-
            openappend(save_file,"conds.tru").
            writedevice(save_file).
            writef("\n\nsegment was classified as containing\n%s with a certainty of %f.\n\nThe users response was %c\n",Class,Cert,Agree).
            closefile(save_file).
            writedevice(screen),
            !.


trans_tru2log:-
            file_str("conds.tru",Data).
            openappend(save_file,"prolog.log").
            writedevice(save_file),
            write(Data),
            closefile(save_file),
            writedevice(screen),
            !.
```

                                                ·

                                             ·

                                                                                                    ·

K36

```
/* FILE: ut3_mod.pro */

/* utilities module */

project "EEG2"
include "dec6-mod.pro"

predicates

/* these are all declared globally in file "dec-mod.pro" to enable access
   from the whole project */



clauses


        wait_count(0):-
                !.

        wait_count(Number) :-
                New_Number = Number-1,
                wait_count(New_Number).




        clear:-
                retractall(yes(_,_)),
                retractall(no(_,_)),
                retractall(cl([t_feature,_,_,_,_,_],_,_)),
                retractall(cl([f_feature,_,_,_,_,_],_,_)),
                retractall(eeg_peak(_,_,_)),
                retractall(sym_peak(_,_)),
                retractall(larg(_,_)).
/*              fail.*/ "
                !.
        clear.

        evalans('y',_,_,_):-
/*              write("\n\nIt's nice to be right !"),*/
                write("\n\nPress any key to continue."),
                readchar(Ans),
                write(Ans),
                !.

        evalans('?',HISTORY,Mygoal,Prev_CF):-
                explain_window,
                report(HISTORY,STNG),
                display(STNG),
                removewindow,
                infer(HISTORY,Mygoal,Prev_CF).
                !.




        evalans(_,_,_,_):-
                write("\n\nPlease update the Knowledge Base!"),
                write("\n\nPress any key to continue."),
                readchar(Ans),
                write(Ans),
                !.
```

K37

```
reverse(X,Y):-
       reverse1([],X,Y).


reverse1(Y,[],Y):-
       !.

reverse1(X1,[U|X2],Y):-
       reverse1([U|X1],X2,Y).


no_sys_error(_,ErrorLevel):-
       ErrorLevel=0,
       !.

no_sys_error(Command,ErrorLevel):-
       warning_window,
       write("SYSTEM ERROR %d ...\n%s",ErrorLevel,Command),
       write("\nPress any key to continue..."),
       readchar(X),
       removewindow.
```

K38

```
/* FILE: win2-mod.pro */

/* The windows definition module, called by file "main-mod.pro" */

project "EEG2"
include "dec6-mod.pro"
include "tdoms.pro"
include "tpreds.pro"
include "menu.pro"


predicates

/* these are all declared globally in file "dec-mod.pro" to enable access
   from the whole project */



clauses

        main_window :-
            makewindow(1,7,1,"[INTELLIGENT EEG CORRECTION (evaluation)]",0,0,24,80),
            cursor(21,4),
            write("evaluation release only               mth91").

        decision_window :-
            makewindow(3,3,14,"[Decision making]",3,0,21,80),
            clearwindow.

        question_window(CHOICE) :-
            menu(4,50,30,14,
            ["ÖááTrue ",
            "úá      ",
            "úá      ",
            "úá      ",
            "úá      ",
            "úááUnknown",
            "úá      ",
            "úá      ",
            "úá      ",
            "úá      ",
            "áááFalse ",
            "?why    "]," < Answer > ",1,CHOICE).

        explain_window :-
            makewindow(4,13,7," < Explanation > ",5,0,18,76).

        list_window :-
            makewindow(6,48,48," < Listing of Knowledge > ",2,0,20,70).

        listopt_window(CHOICE) :-
            menu(5,27,30,14,["Screen","Printer","File"]," < List to.. > ",1,CHOICE).

        logopt_window(CHOICE) :-
            menu(5,27,30,14,["Screen","Printer"]," < View using... > ",1,CHOICE).


        goodbye_window :-
            makewindow(8,64,7," < Goodbye > ",3,30,7,20).


        main_menu(Choice1) :-
            menu(5,27,30,14,
            ["Select Data     ",
            "Consultation    ",
```

```prolog
                  'eXit            '].
                  ' < Main Menu > ',
                  1,
                  Choice1).

control_menu(Choice2) :-
         menu(5,27,30,14,
         ['Preview data       ',
          'Select segment     ',
          'Classify segment   ',
          'List knowledge     ',
          'eXit            '],
          ' < Consultation Menu > ',
          2,
          Choice2).


process_window :-
         makewindow(2,30,30,' < Message Window > ',12,30,10,48).



   .it_window :-
         makewindow(5,7,33,' < Knowledge Base Editor > ',2,0,20,80).

kb_select_window(KBname):-
         makewindow(9,30,30,' < Select the Knowledge Base > ',4,10,10,60),
         dir('','*.kb',KBname).

rules_select_window(Rname):-
         makewindow(12,30,30,' < Select the Knowledge Base > ',4,10,10,60),
         dir('','*.asc',Rname).

data_select_window(Dname):-
         makewindow(7,30,30,' < Select the Data > ',4,10,10,60),
         dir('c:\\ils\\mils\\datas','*.*',Dname).


select_segment_window(Sname) :-
         makewindow(10,30,30,' < Select a segment to analyse > ',4,10,10,60),
         dir('c:\\ils\\mils\\segs','*.*',Sname).

warning_window:-
         makewindow(11,71,71,' < W A R N I N G > ',1,1,5,39).
```

```
/*EXTOPROL: Program to extract spectral features from EEG data and pass to PROLOG*/
/* called as: extract(chanel number, feature list) from PROLOG.*/
/* input TI file number for EEG/EOG data (represents montage 1 or 2)*/
/*                                              TIO = fp2-f4 etc.*/
/* output: a list of features (a1,b1,c1,d1,e1,a2.....e3)*/
/* a=channel,b=peak number,c=magnitude,d=frequency,e=width*/


/* compile this program using MHCPROL.TC options, ie large model, no underbars.*/



#define time 0
#define amplitude 1       /*feature matrix index for amplitude of spectral peak */
#define false 0
#define freq_discrim 2    /* spectral peak discriminator to prevent detection of too many peaks */
#define freq_thresh 6     /* spectral peak discriminator to prevent detection of too few peaks */
#define frequency 0       /* feature matrix index for frequency of spectral peak */
#define PSDBuf_size 81    /* 81 psd estimates/2-second segment/channel */
#define CorrBuf_size 512          /* 512 correlation samples */
#define filesize 6656
#define buffer_offset 390/* offset to start of data in ILS single record file */
#define max_PSD_peaks 4           /* consider only the first (max_peaks) peaks */
#define max_PSD_features 3        /* only (max_PSD_features) features held in array features */
#define max_c_peaks 8
#define max_c_attribs 2
#define max_c_features 4
#define max_peaks_row 10
#define max_peaks_col 10
#define max_features_row 10
#define max_features_col 10
#define true 1
#define width 2                   /* feature matrix index for width of spectral peak */
#define listfno 1
#define nilfno 2



typedef struct rlist{
        unsigned char functor;
        double value;
        struct rlist *next;

}dublist;




int compare_mag(float *input_buffer,double peaks[max_peaks_row][max_peaks_col]);
int compare_width(int excur_num,float *input_buffer,double peaks[max_peaks_row][max_peaks_col],double features[max_features_row][m
int multi_peak(int *peak_num,float *input_buffer,double peaks[max_peaks_row][max_peaks_col],double features[max_features_row][max_
int single_peak(int *peak_num,float *input_buffer,double peaks[max_peaks_row][max_peaks_col],double features[max_features_row][max
void array_to_list(int rows, int cols, dublist **list);
int find_peaks(float far *input_buffer,double peaks[max_peaks_row][max_peaks_col]);
int compare_lags(int c_peak_num,double peaks[max_peaks_row][max_peaks_col],double features[max_features_row][max_features_col]);
int extern ReadPSD(float far *input_buffer,int j);
int extern ReadCorr(float far *input_buffer,int j);
void *alloc_gstack(unsigned);
void *_malloc(unsigned);
void *_free(void *);



/*---------------------------------------------------------------*/
/* Global variable array initialisation */

        double peaks[max_peaks_row][max_peaks_col] = {
             (0)
```

```c
                                );

                double features[max_features_row][max_features_col] = {
                        {0}
                };


/* ......................................................... */


/* PSD spectral feature extraction procedure */


extractPSD_0(int j,dublist **outlist)


{
                int i,k,l,m,excur_num,peak_num;
                float far *input_buffer;
/*              float_list **outlist;*/



                input_buffer = (float far *) alloc_gstack(filesize);


/*              zwf("%lu bytes free after input alloc\n",coreleft());*/



                /* read 1 segment of 2 second psd estimates */


/*              zwf("%d\n",j);*/

                if((ReadPSD(input_buffer,j)) == 0)
                {

                        input_buffer+=buffer_offset;


/*                      test....print out data*/
/*       .               for(k=0;k<PSDBuf_size;k++)
                                zwf("%8.4e\n",input_buffer[k]);
*/

/*                      clear peak array.*/

                        for(l=0;l<max_peaks_row;l++)
                        {
                                for(m=0;m<max_peaks_col;m++)
                                {
                                        features[l][m]=0.0;
                                }
                        }



                        excur_num = compare_mag(input_buffer,peaks);



/*                      zwf("\n");*/
```

```
                for(l=0;l<2;l++)
                {
                        for(m=0;m<max_PSD_peaks;m++)
                        {
                                zwf("%d ",peaks[l][m]);
                        }
                        zwf("\n");
                }
*/

/*              clear feature array */

                for(l=0;l<max_features_row;l++)
                {
                        for(m=0;m<max_features_col;m++)
                        {
                                features[l][m]=0.0;
                        }
                }



                peak_num = compare_width(excur_num,input_buffer,peaks,features);


                zwf("\n");
                for(l=0;l<max_PSD_peaks;l++)
                {
                        for(m=0;m<max_PSD_features;m++)
                        {
                                zwf("%8.4e ",features[l][m]);
                        }
                        zwf("\n");
                }


                array_to_list(peak_num,max_PSD_features,outlist);

        }else
        {
                zwf("\nExtractPSD: file error T[%d\n",j);
        }

}




/*  end of extractPSD */




/*---------------------------------------------------------------------------*/


/* compare_mag: to compare the magnitude of the PSD with that of a preset   */
/*              threshold so that peaks can be identified. the input is the */
/*              array "input_buffer" and the output is the array "peaks"     */
/*              which stores the start and stop frequency of each excursion.*/


int compare_mag(float *input_buffer,double peaks[max_peaks_row][max_peaks_col])
{
```

```c
        int i,start,excur_num;
        float mag_thresh;

        start=false;
        excur_num=0;

        for(i=0;i<PSDBuf_size;i++)
        {
                if((i>=0)&(i<6))
                {
                        mag_thresh = 300e6-(i*28.333e6); /*300e6 to 130e6 in 0hz to 1.5 hz */
                }
                if((i>=6)&(i<12))
                {
                        mag_thresh = 130e6-((i-6)*13.333e6); /*130e6 to 50e6 in 1.5hz to 3hz */
                }
                if(i>=12)
                {
                        mag_thresh = 50e6-((i-12)*588.235e3); /*50e6 to 10e6 in 3hz to 20hz */
                }
/*              zwf("\nsample=%8.4e : thresh=%8.4e",input_buffer[i],mag_thresh);*/
                if((input_buffer[i] >= mag_thresh) & (!start))
                {
                        peaks[0][excur_num] = i-1;
                        start = true;
                }
                if((input_buffer[i] <= mag_thresh) & (start))
                {
                        peaks[1][excur_num] = (i);
                        start = false;
                        excur_num++;
                }
                if(excur_num > (max_PSD_peaks-1)) return max_PSD_peaks;

        }
        zwf("\nexcur_num=%d\n",excur_num);
        return excur_num;
}

/*----------------------------------------------------------------------*/

/* compare_width: to compare the widths of excursions of the psd amplitude */
/*                threshold, in case of multiple peaks. the input is the   */
/*                arrays "peaks" and "input_buffer" and the output is in the*/
/*                form of spectral features, ie peak frequency, amplitude  */
/*                and width. these are held in array "features" which stores*/
/*                this information for each peak and channel for a 2-second */
/*                segment of data.                                         */


int compare_width(int excur_num,float *input_buffer,double peaks[max_peaks_row][max_peaks_col],double features[max_featur

{
        int i,peak_num;

        peak_num = 0;
        for(i=0;i<excur_num;i++)
        {
                if(peak_num < max_PSD_peaks)
                {
/*                      zwf("\npeaks[1][%d] - peaks[0][%d]=%d\n",i,i,peaks[1][i]-peaks[0][i]);*/
                        if((peaks[1][i] - peaks[0][i]) >=freq_thresh)
```

```c
                  (
                          multi_peak(&peak_num,input_buffer,peaks,features,i);
                  ) else
                  (
                          single_peak(&peak_num,input_buffer,peaks,features,i);
                  )
            )

      )
/*      zwf("\npeak_num=%d\n",peak_num);*/
        return peak_num;
)


/*·································································*/

/* single_peak: is entered when an excursion over the amplitude threshold   */
/*              and narrower than 2Hz is detected in the array "peaks".      */
/*              this is likely to mean that there only one spectral peak.    */
/*              this routine finds the peak and stores the peak features in  */
/*              in the array "features".                                     */


int single_peak(int *peak_num,float *input_buffer,double peaks[max_peaks_row][max_peaks_col],double features[max_features_row][max

(
        int k;

        for(k = peaks[0][i];k < peaks[1][i];k++)
          (
/*                zwf("\nk-1=%f, k=%f, k+1=%f",input_buffer[k-1],input_buffer[k],input_buffer[k+1]);*/
                  if((input_buffer[k] > input_buffer[k-1]) & (input_buffer[k] > input_buffer[k+1]))
                    (
                          features[*peak_num][frequency] = k;
                          features[*peak_num][amplitude] = input_buffer[k];
                          features[*peak_num][width] = 0;
                          (*peak_num)++;
                          return;
                    )
                  if((input_buffer[k] > input_buffer[k-1]) & (input_buffer[k] >= input_buffer[k+1]))
                    (
                          if((input_buffer[k+1] >= input_buffer[k+2]))
                            (
                                  features[*peak_num][frequency] = k+1;
                                  features[*peak_num][amplitude] = input_buffer[k+1];
                                  features[*peak_num][width] = 0;
                                  (*peak_num)++;
                                  return;
                            )

                    )

          )


      )

)

/*·································································*/

/* multi_peak: is enterred when an excursion over the amplitude threshold    */
/*             and wider than 2Hz is detected in the array "peaks". this     */
/*             is likely to mean that there is more than one peak but close  */
/*             together. this routine finds the peaks and stores the peak    */
/*             features in the array "features".                             */
```

K 4-5

```c
int multi_peak(int *peak_num,float *input_buffer,double peaks[max_peaks_row][max_peaks_col],double features[max_features_ro

{
        int k;

        for(k = peaks[0][i];k < peaks[1][i];k++)
        {
                /* don't assert another peak if three have already been asserted */
                if(*peak_num<max_PSD_peaks)
                {
/*              zwf("\nk=%d\n",k);*/
                        /* does the present sample represent a local maximum? */
                        if((input_buffer[k] > input_buffer[k-1]) & (input_buffer[k] > input_buffer[k+1]))
                        {
                                /* first peak? */
                                if(*peak_num>0)
                                {
                                        /* no - peaks might be the same */
/*                                      zwf("\nfeatures[%d][0]=%8.4e\n%d-%8.4e=%8.4e\n",*peak_num-1,features[(*peak_num)-1]
                                        /* is the present local maximum closer than 0.5Hz to the last? */
                                        if(k-features[(*peak_num)-1][frequency] > freq_discrim)
                                        {
                                                /* yes - present maximum is a new peak */
                                                features[*peak_num][frequency] = k;
                                                features[*peak_num][amplitude] = input_buffer[k];
                                                features[*peak_num][width] = 0;
                                                (*peak_num)++;
                                                k+=1;
                                        }else
                                        {
                                                /* no - peak is half way between the two and overwrites the last peak */
                                                features[(*peak_num)-1][frequency] = k-1;
                                                features[(*peak_num)-1][amplitude] = input_buffer[k-1];
                                                features[*peak_num][width] = 0;
                                                k+=1;
                                        }
                                }else
                                {
                                        /* yes - assert the present local maximum */
                                        features[*peak_num][frequency] = k;
                                        features[*peak_num][amplitude] = input_buffer[k];
                                        features[*peak_num][width] = 0;
                                        (*peak_num)++;
                                        k+=1;
                                }
                        }

                }

        }

}

/*--------------------------------------------------------------------*/

void array_to_list(int rows, int cols, dublist **list)
{
        int j,k;

        for(j=0;j<rows;j++)
        {
                dublist *p=*list=alloc_gstack(sizeof(dublist));
```

```
                p->functor=listfno;
                p->value = j+1;
                list = &(*list)->next;
                for(k=0;k<cols;k++)
                {
                        dublist *p=*list=alloc_gstack(sizeof(dublist));
                        p->functor=listfno;
                        p->value = features[j][k];
/*                      zwf("%8.4e ",features[j][k]);*/
                        list = &(*list)->next;


                }


        }



        {
                dublist *p=*list=alloc_gstack(sizeof(char));
                p->functor = nilfno;

        }

)


/*.....................................................................*/

/* Correlation feature extraction procedure */


extractCORR_0(int j,dublist **outlist)

{
        int l,m,c_peak_num,c_feature_num;
        float far *input_buffer;


        input_buffer = (float far *) alloc_gstack(filesize);

/*      zwf("\ninput_buffer address = %X",input_buffer);*/

/*      zwf("%lu bytes free after input alloc\n",coreleft());*/


        /* read 1 segment of 2 second correlation */


/*      zwf("%d\n",j);*/

        if((ReadCorr(input_buffer,j)) == 0)
        {
                /* clear time lags -10 to 0 in correlation array - to enable peak ident */
                input_buffer+=(buffer_offset-10);
                for(l=0;l<10;l++)
                {
                        input_buffer[l]=0.0;
                }
                input_buffer+=10;

                /* clear time lags 512 to 522 in correlation array - to enable peak ident */
                input_buffer+=512;
                for(l=0;l<10;l++)
                {
                        input_buffer[l]=0.0;
```

```
            )
            input_buffer-=512;


            c_peak_num = find_peaks(input_buffer,peaks);

            if(c_peak_num>0)
            (
                    zwf("\n");
                    for(l=0;l<c_peak_num;l++)
                    (
                            for(m=0;m<max_c_attribs;m++)
                            (
                                    zwf("%8.4e  ",peaks[l][m]);
                            )
                            zwf("\n");
                    )

                    c_feature_num = compare_lags(c_peak_num,peaks,features);


                    zwf("\n");
                    for(l=0;l<max_c_features;l++)
                    (
                            zwf("%8.4e  ",features[0][l]);
                    )


                    if(c_feature_num)
                    (
                            array_to_list(1,max_c_features,outlist);
                    )


            )else
            (
                    zwf("\n\nNO SIGNIFICANT PEAKS");
            )


        )else
        (
                zwf("\nExtractCORR: file error NU%d\n",j);
        )


)


/*  end of extractCORR_0 */


/*-------------------------------------------------------------------*/
int find_peaks(float far *input_buffer,double peaks[max_peaks_row][max_peaks_col])
(
        int i,l,m,c_peak_num;

        /*      clear peak array */

        for(l=0;l<max_peaks_row;l++)
        (
                for(m=0;m<max_peaks_col;m++)
                (
                        peaks[l][m]=0.0;
```
K48

```c
            )
        )


        c_peak_num=0;



        for(i=0;i<CorrBuf_size;i++)
        {
/*              zwf("\ni=%d spl=%8.4e pk_num=%d",i,input_buffer[i],c_peak_num);*/
                /* don't assert another peak if eight have already been asserted */
                if(c_peak_num<max_c_peaks)
                {
                        if(input_buffer[i]>20e5)
                        {
/*                              zwf(">0");*/
                                /* does the present sample represent a local maximum? */
                                if((input_buffer[i] > input_buffer[i-1])
                                & (input_buffer[i] > input_buffer[i-2])
                                & (input_buffer[i] > input_buffer[i-3])
                                & (input_buffer[i] > input_buffer[i-4])
                                & (input_buffer[i] > input_buffer[i-5])
                                & (input_buffer[i] > input_buffer[i-6])
                                & (input_buffer[i] > input_buffer[i-7])
                                & (input_buffer[i] > input_buffer[i-8])
                                & (input_buffer[i] > input_buffer[i-9])
                                & (input_buffer[i] > input_buffer[i-10])
                                & (input_buffer[i] > input_buffer[i-11])
                                & (input_buffer[i] > input_buffer[i-12])
                                & (input_buffer[i] > input_buffer[i-13])
                                & (input_buffer[i] > input_buffer[i-14])
                                & (input_buffer[i] > input_buffer[i-15])
                                & (input_buffer[i] > input_buffer[i-16])
                                & (input_buffer[i] > input_buffer[i-17])
                                & (input_buffer[i] > input_buffer[i-18])
                                & (input_buffer[i] > input_buffer[i-19])
                                & (input_buffer[i] > input_buffer[i-20])
                                & (input_buffer[i] > input_buffer[i+1])
                                & (input_buffer[i] > input_buffer[i+2])
                                & (input_buffer[i] > input_buffer[i+3])
                                & (input_buffer[i] > input_buffer[i+4])
                                & (input_buffer[i] > input_buffer[i+5])
                                & (input_buffer[i] > input_buffer[i+6])
                                & (input_buffer[i] > input_buffer[i+7])
                                & (input_buffer[i] > input_buffer[i+8])
                                & (input_buffer[i] > input_buffer[i+9])
                                & (input_buffer[i] > input_buffer[i+10])
                                & (input_buffer[i] > input_buffer[i+11])
                                & (input_buffer[i] > input_buffer[i+12])
                                & (input_buffer[i] > input_buffer[i+13])
                                & (input_buffer[i] > input_buffer[i+14])
                                & (input_buffer[i] > input_buffer[i+15])
                                & (input_buffer[i] > input_buffer[i+16])
                                & (input_buffer[i] > input_buffer[i+17])
                                & (input_buffer[i] > input_buffer[i+18])
                                & (input_buffer[i] > input_buffer[i+19])
                                & (input_buffer[i] > input_buffer[i+20]))
                                {
                                        peaks[c_peak_num][time] = i;
                                        peaks[c_peak_num][amplitude] = input_buffer[i];
                                        c_peak_num++;
                                        zwf(" ****");
                                }
```

```
                    )/* else
                    if((input_buffer[i] > input_buffer[i-1])
                    & (input_buffer[i] >= input_buffer[i+1]))
                    (
                            if((input_buffer[i+1] >= input_buffer[i+2]))
                            (
                                    peaks[c_peak_num][time] = i+1;
                                    peaks[c_peak_num][amplitude] = input_buffer[i+1];
                                    c_peak_num++;
                            )


                    )*/

            )

     )
     return c_peak_num;
)


/*--------------------------------------------------------------------*/

int compare_lags(int c_peak_num,double peaks[max_peaks_row][max_peaks_col],double features[max_features_row][max_features_c


(
     int j;
     int mult;
     int lag;
     int corr_num;
     int i=1;



     if(c_peak_num<3)
     (
          features[0][0]=peaks[0][time];
          features[0][1]=peaks[0][amplitude];
          features[0][2]=peaks[(c_peak_num-1)][time]-peaks[0][time];
          features[0][3]=c_peak_num;
     )else
     (
          corr_num=2;
          do
          (
                j=i+1;
                mult=2;
                do
                (
                        zwf("\ni=%d:j=%d:mult=%d:corr_num=%d",i,j,mult,corr_num);


                        lag = (peaks[i][time]-peaks[0][time])*mult + peaks[0][time];
                        if((peaks[j][time] > (lag - 50)) & (peaks[j][time] < (lag +50)))
                        (
                                corr_num++;
                                j++;
                                mult++;
                        )else
                        (
                                if(peaks[j][time] > (lag +50))
                                (
                                        mult++;
                                )else
```

KSO.

```
                                   (
                                          j++;
                                   )
                          )

                   )while(j<c_peak_num);

                   if(corr_num>1)
                   (
                          features[0][0]=peaks[0][time];
                          features[0][1]=peaks[0][amplitude];
                          features[0][2]=(peaks[i][time]-peaks[0][time]);
                          features[0][3]=corr_num;
                          break;
                   )
                   i++;

            )while(i<(c_peak_num-1));

       )

       if(corr_num>0)
       (
              return 1;
       )else
              return 0;

)
/*..................................................................*/

/*MON.C : Program to control the display of graphics data */

#define XEND 640
#define YEND 480
#define VPAGESIZE ((XEND/8)*YEND)
#define buf_size 65502



void plot(unsigned char far *input_buffer);
void extern SplitScreen(int);
void extern ScreenOrigin(int,int);
void extern SetVideoMode(int);
int     extern InputData(unsigned char far *input_buffer);
void extern LoadScreen();
void extern BlankScreen();
void extern Showscreen();
int     extern PlotData();
int     extern ReadKey();



monitor_0()

/* main()*/
(
       int c;
       int x = 1;
       int y = 0;
       unsigned char far *input_buffer;

/*     if((input_buffer = farmalloc(buf_size)) == NULL)
       (
              printf("\nmemory allocation failure");
```

```c
                exit(1);
        }

*/

        input_buffer = (char far *) alloc_gstack(buf_size);


        if((InputData(input_buffer)) == 0)
        {
/*              printf("MON: data has been read\n\n");*/
                zwf("MON: data has been read\n\n");
        }


        SetVideoMode(0x12);

        splitScreen(460);
        ScreenOrigin(0,20);

        BlankScreen();

        LoadScreen();

        ShowScreen();

        do
        {
                c=ReadKey();
                switch(c)
                {
                        case 77:plot(input_buffer);break;
                }
        }while(c!=27);


        ScreenOrigin(0,0);

        SplitScreen(0x3ff);

        SetVideoMode(3);

}
/*...................................................................*/

void plot(unsigned char far *input_buffer)
{

        int i,x,y,y1,y2,index1,index2,yl,chan;

        int lst_scr_old[16] = {44,92,140,188,236,284,332,380,428,476,524,572,620,668,716,764};
        int this_scr_old[16] = {44,92,140,188,236,284,332,380,428,476,524,572,620,668,716,764};

        index1=7;
        index2=7;


        for(i=0;i<3;i++)        /*(int)(2048/(XEND*tc));i++)*/
        {
                for(x=1;x<=XEND;x++)
                {
                        ScreenOrigin(x,20);

                        for(chan=0;chan<8;chan++)
```

```
                    {
                            if(i)
                            {
                                    lst_scr_old[chan]=PlotData(input_buffer,chan,index1,x-1,lst_scr_old[chan],x,1);
                                    index1+=2;
                            }
                            this_scr_old[chan]=PlotData(input_buffer,chan,index2,x-1,this_scr_old[chan],x,14);
                            index2+=2;
                    }
                    index2+=112;   /*112 16 for last 8 chans and 3*32 to miss two samples*/
                    if(i)
                            index1+=112;


            }
        }

    }
/*....................................................................*/
/*....................................................................*/
```

```
;ReadPSD.asm:  read ILS PSD record file to memory

;this program is designed to read PSD data from an ILS record file. Data is held in
;floating point format. ReadPSD(input_buffer,i) is called from C where i is the index
;of the file (Ili), and input_buffer is a pointer to prealocated memory.
;
;input:
;              far pointer to the start of the allocated memory (input_buffer).
;              index of ti file (i).
;
;output:
;              AX=0 for corrrect termination else error code.




ReadPSD_TEXT    SEGMENT byte public 'CODE'

datgrp  GROUP   data,bss
                ASSUME  cs:ReadPSD_TEXT,ds:datgrp


CR              equ     00h             ;
LF              equ     0Ah             ;
;FILE_SIZE      equ     11776   ;
FILE_SIZE       equ     6656    ;

;stack frame


ReadPSDParms            struc
                dw      ?                       ;far return address
                dw      ?                       ;  ''
                dw      ?                       ;bp

BUF_OFF dw      ?                       ;input data buffer offset address
BUF_SEG dw      ?                       ;input data buffer segment address
Index   dw      ?

ReadPSDParms            ends



; ReadPSD is not _ReadPSD because of PROLOG requirements

                PUBLIC  ReadPSD

ReadPSD proc    far

                push    bp
                mov     bp,sp
                push    si
                push    di
                push    ds


                mov     ax,data
                mov     ds,ax



name:
                mov     ax,[bp+Index]                   ;file index to read
```

```
                cmp     al,10
                jb      single
double:
                mov     datgrp:Filename+2,31h   ;convert double decimal number to ascii
                mov     datgrp:Fname+4,31h
                sub     al,10                           ;
                add     al,30h                          ;
                mov     datgrp:Filename+3,al    ;store in data segment
                mov     datgrp:Filename+4,0             ;end marker
                mov     datgrp:Fname+5,al
                jmp     open


single:
                add     al,30h                          ;convert sinle decimal number to ascii
                mov     datgrp:Filename+2,al    ;store in data segment
                mov     datgrp:Filename+3,0            ;end marker
                mov     datgrp:Fname+4,al
                mov     datgrp:Fname+5,20h




Open:
;               mov     dx,offset Fname         ;print file name interfers with PROLOG
;               mov     ah,9
;               int     21h


                mov     ah,3dh                          ;dos interupt-open file
                mov     al,0
                mov     dx,offset Filename
                int     21h
                jc      NoFile
                mov     (Handle),ax


Input:  mov     ah,3fh                          ;dos interrupt-read file
                mov     bx,[Handle]
                mov     cx,FILE_SIZE                    ;number of bytes
                push    ds
                mov     ds,[bp+BUF_SEG]         ;segment address of pointer
                mov     dx,[bp+BUF_OFF]         ;offset address of pointer
                int     21h
                pop     ds
                jc      ReadErr


                mov     ah,3Eh                          ;dos interrupt-close file
                mov     bx,[Handle]
                int     21h

                mov     ax,0                            ;return correct termination code


                jmp     ReadPSDEnd

NoFile: ;mov    dx,offset ErrMsg1               ;file does not exist
                ;mov    ah,9
                ;int    21h
                mov     ax,1                            ;return error code
                jmp     ReadPSDEnd

ReadErr:        ;mov    dx,offset ErrMsg2               ;file read error.
```

```
                ;mov    ah,9
                ;int    21h
                mov     ax,2                                    ;return error code



ReadPSDEnd:

                pop     ds                                      ;restore regs.
                pop     di
                pop     si
                pop     bp
                ret

ReadPSD ENDP

ReadPSD_TEXT            ENDS


data            SEGMENT byte PUBLIC 'data'

Filename        db      'tix',0,0
Fname   db      cr,lf,'TIxx',cr,lf,'$'
handle  dw      0
ErrMsg1         db      cr,lf,'READ: file not found',cr,lf,'$'
ErrMsg2         db      cr,lf,'READ: file read error',cr,lf,'$'

data    ENDS

bss     segment word public 'bss'

bss     ENDS

        END
```

KSB

KSB

```
;ReadCORR.asm:  read ILS Correlation record file to memory

;this program is designed to read Correlation data from an ILS record file. Data is held in
;floating point format. ReadCorr(input_buffer,i) is called from C where i is the index
;of the file (NUi), and input_buffer is a pointer to prealocated memory.
;
;input:
;                 far pointer to the start of the allocated memory (input_buffer).
;                 index of NU file (i).
;
;output:
;                 AX=0 for correct termination else error code.




ReadCORR_TEXT    SEGMENT byte public 'CODE'

datgrp  GROUP    data,bss
                 ASSUME   cs:ReadCORR_TEXT,ds:datgrp


CR               equ     00h              ;
LF               equ     0Ah              ;
FILE_SIZE        equ     6656             ;

;stack frame


ReadCorrParms            struc
                 dw      ?                        ;far return address
                 dw      ?                        ;  ''
                 dw      ?                        ;bp

BUF_OFF dw       ?                        ; input data buffer offset address
BUF_SEG dw       ?                        ; input data buffer segment address
Index    dw      ?

ReadCorrParms            ends



; ReadCorr is not _ReadCorr because of PROLOG requirements

                 PUBLIC  ReadCorr

ReadCorr         proc    far

                 push    bp
                 mov     bp,sp
                 push    si
                 push    di
                 push    ds


                 mov     ax,data
                 mov     ds,ax


name:
                 mov     cl,30h                   ;init decimal count reg
                 mov     ax,{bp+Index}            ;file index to read
```

```
                    cmp     al,10
                    jb      single
double:

                    sub     al,10                   ;count decimal
                    add     cl,1                            ;
                    cmp     al,10                   ;
                    jge     double                  ;


                    mov     datgrp:Filename+2,cl    ;convert double decimal number to ascii
                    mov     datgrp:Fname+4,cl
                    add     al,30h                  ;
                    mov     datgrp:Filename+3,al    ;store in data segment
                    mov     datgrp:Filename+4,0         ;end marker
                    mov     datgrp:Fname+5,al
                    jmp     open


single:
                    add     al,30h                  ;convert sinle decimal number to ascii
                    mov     datgrp:Filename+2,al    ;store in data segment
                    mov     datgrp:Filename+3,0         ;end marker
                    mov     datgrp:Fname+4,al
                    mov     datgrp:Fname+5,20h




Open:
;                   mov     dx,offset Fname         ;print file name interfers with PROLOG
;                   mov     ah,9
;                   int     21h


                    mov     ah,3dh                  ;dos interupt-open file
                    mov     al,0
                    mov     dx,offset Filename
                    int     21h
                    jc      NoFile
                    mov     [Handle],ax


Input:  mov         ah,3fh                          ;dos interrupt-read file
                    mov     bx,[Handle]
                    mov     cx,FILE_SIZE                    ;number of bytes
                    push    ds
                    mov     ds,[bp+BUF_SEG]         ;segment address of pointer
                    mov     dx,[bp+BUF_OFF]         ;offset address of pointer
                    int     21h
                    pop     ds
                    jc      ReadErr


                    mov     ah,3Eh                          ;dos interrupt-close file
                    mov     bx,[Handle]
                    int     21h

                    mov     ax,0                                 ;return correct termination code


                    jmp     ReadCorrEnd

NoFile: ;mov        dx,offset ErrMsg1               ;file does not exist
```

```
                        ;mov    ah,9
                        ;int    21h
                        mov     ax,1                            ;return error code
                        jmp     ReadCorrEnd

ReadErr:                ;mov    dx,offset ErrMsg2       ;file read error.
                        ;mov    ah,9
                        ;int    21h
                        mov     ax,2                            ;return error code



ReadCorrEnd:

                        pop     ds                              ;restore regs.
                        pop     di
                        pop     si
                        pop     bp
                        ret

ReadCorr        ENDP

ReadCORR_TEXT           ENDS


data            SEGMENT byte PUBLIC 'data'

Filename        db      'NUx',0,0
Fname   db      cr,lf,'NUxx',cr,lf,'$'
handle  dw      0
ErrMsg1         db      cr,lf,'ReadCorr: file not found',cr,lf,'$'
ErrMsg2         db      cr,lf,'ReadCorr: file read error',cr,lf,'$'

data    ENDS

bss     segment word public 'bss'

bss     ENDS

        END
```

K62

```
;Fast assembler implementation of Bresenham's line drawing algorithm
;
;
Line_TEXT          segment byte    public 'code'
                   assume cs:Line_TEXT,ds:Data



EVGA_SCREEN_WIDTH_IN_BYTES      equ     80              ;640x480 pixels
EVGA_SCREEN_SEGMENT                     equ     0a000h  ;
GC_INDEX                                equ     3ceh            ;graphics cont'r index
                                                                ;reg'r port
SET_RESET_INDEX                 equ     0               ;GC register
ENABLE_SET_RESET_INDEX          equ     1               ;    ''
BIT_MASK_INDEX                          equ     8               ;    ''
GRAPHICS_MODE                           equ     5               ;    ''
COLOUR_COMPARE                          equ     2               ;    ''
READ_MAP                                equ     4
DISPLAYED_SCREEN_SIZE           equ     (640/8)*480
SC_INDEX                                equ     3c4h
MAP_MASK                                equ     2



;stack frame


PlotDataParms           struc
               dw      ?                       ;far return address
               dw      ?                       ;   ''
               dw      ?                       ;bp

BUF_OFF dw     ?                       ;input data buffer segment address
BUF_SEG dw     ?                       ;input data buffer offset address
CHAN           dw      ?                       ;data channel
INDEX   dw     ?               ;data index
X0             dw      ?                       ;parameters passed to EVGALine
Y0             dw      ?                       ;   ''
X1             dw      ?                       ;   ''
Colour  db     ?               ;   ''
               db      ?
PlotDataParms  ends


;Line drawing macros


;LINE1 is for drawing lines where deltaX is greater or equal to deltaY.

;input:
;               MOVE_LEFT: 1 if deltaX <0, 0 else
;               AL: pixel mask for initial pixel
;               BX: |deltaX| (X distance between start and end points)
;               DX: address of GC data register, with index register set to
;                       index of BIT_MASK register.
;               SI: deltaY (Y distance between start and end points)
;               DS:DI: display memory address f byte containing initial pixel

;output:
;               none.


LINE1   macro   MOVE_LEFT
               local   LineLoop,MoveXCoord,NextPixel,Line1End
```

```
        local       MoveToNextByte,ResetBitMaskAccumulator
        local       PixelOnLine,PixelNotOnLine,NextPixelEnd

        mov         ah,al           ;al is used to store accumulated bit mask
        mov         cx,bx           ;number of pixels
        jcxz        Line1End                ;no more pixels? Allways at least 1
        shl         si,1                    ;error term calculation
        mov         bp,si           ;
        sub         bp,bx           ;
        shl         bx,1                    ;
        sub         si,bx           ;
        add         bx,si           ;


        and         ds:[di],al      ;is initial pixel on a grid line?
        jz          LineLoop                ;no
        mov         al,0            ;yes


LineLoop:
        and         bp,bp           ;is error term -ve?
        js          MoveXCoord      ;yes - stay at same y coord


        out         dx,al           ;set up pixel bit mask
        mov         ds:[di],al      ;display byte
        add         di,EVGA_SCREEN_WIDTH_IN_BYTES    ;move down 1 line
        add         bp,si           ;adjust error


if MOVE_LEFT
        rol         ah,1                    ;adjust pixel mask according to
else                                        ;entry conditions
        ror         ah,1            ;
endif                       ;

        jnc         ResetBitMaskAccumulator ;didn't wrap to next byte
        jmp         short MoveToNextByte    ;did


MoveXCoord:
        add         bp,bx           ;adjust error term

if MOVE_LEFT
        rol         ah,1                    ;adjst pixel bit mask, adjusting display
else                                        ;memory address when wraps and writting
        ror         ah,1                    ;byte
endif

        jnc         NextPixel               ;same byte so don't modify display memory



        out         dx,al           ;display this byte
        mov         ds:[di],al      ;


MoveToNextByte:             ;modify dosplay memory address
if MOVE_LEFT
        dec         di
else
        inc         di
endif
```

```
ResetBitMaskAccumulator:                 ;clear pixel bit mask
               sub    al,al

NextPixel:                               ;
               push   ax                        ;save old bit mask accumulator
               mov    al,ds:[di]    ;is new pixel on grid line?
               and    ah,al         ;
               jz     PixelNotOnLine ;

PixelOnLine:
               pop    ax                        ;yes - don't update bit mask accumulator
               jmp    short NextPixelEnd

PixelNotOnLine:
               pop    ax                        ;no - update bit mask accumulator
               or     al,ah

NextPixelEnd:                            ;get next pixel
               loop   LineLoop

Line1End:


               out    dx,al         ;display last pixel
               mov    ds:[di],al

               endm




;
;
;LINE2 is for drawing lines where deltaX is less than deltaY.

;input:
;              MOVE LEFT: 1 if deltaX <0, 0 else
;              AL: pixel mask for initial pixel
;              BX: |deltaX| (X distance between start and end points)
;              DX: address of GC data register, with index register set to
;                   index of BIT MASK register.
;              SI: deltaY (Y distance between start and end points)
;              DS:DI: display memory address f byte containing initial pixel

;output:
;              none.


LINE2   macro  MOVE_LEFT
               local   LineLoop,MoveYCoord,ETermAction,Line2End
               local   PixelNotLBlue1,PixelNotLBlue2
               local   MoveYCoordEnd

               mov    cx,si         ;no of pixels
               shl    bx,1                      ;error term calculation
               mov    bp,bx         ;
               sub    bp,si         ;
               shl    si,1                      ;
               sub    bx,si         ;
               add    si,bx         ;


               mov    ah,ds:[di]            ;is initial pixel on grid line?
               and    ah,al                 ;
               jz     short PixelNotLBlue1;
```

K65

```
                mov     al,0              ;

PixelNotLBlue1:                   ;no - display byte
                out     dx,al             ;
                mov     ds:[di],ah        ;


                jcxz    Line2End                  ;no more pixels?


LineLoop:
                and     bp,bp             ;is error term -ve?
                jns     ETermAction       ;no - advance X coordinate

                add     bp,si             ;increase error term and increase y
                jmp     short MoveYCoord  ;

ETermAction:

if MOVE_LEFT
                rol     al,1                      ;adjust bit mask acccording to entry
                                                  ;conditions
                sbb     di,0
else
                ror     al,1
                adc     di,0
endif

                add     bp,bx             ;adjust error term


MoveYCoord:
                add     di,EVGA_SCREEN_WIDTH_IN_BYTES    ;move y coord down


                mov     ah,ds:[di]        ;is new pixel on grid line?
                and     ah,al             ;
                jnz     short MoveYCoordEnd ;
; ;              mov     al,0              ;

PixelNotLBlue2:                   ;no - display byte
                out     dx,al             ;
                mov     ds:[di],ah        ;



MoveYCoordEnd:
                loop    LineLoop

Line2End:



                endm


;
;
;line drawing routine

;               public  _PlotData
                public  PlotData

; _PlotData     proc    far
```

K66

```
PlotData                proc    far

                push    bp
        .   mov     bp,sp
                push    si
                push    di
                push    ds
                push    es


                mov     ax,[bp+BUF_SEG]
                mov     ds,ax
                mov     bx,[bp+BUF_OFF]
                mov     di,[bp+INDEX]
                mov     ah,[bx][di+1]           ;y
                mov     al,[bx][di]

        shr     ax,1
        shr     ax,1
        shr     ax,1
        shr     ax,1
        shr     ax,1
        shr     ax,1

                add     ax,14                   ;y=y+14

                mov     bx,[bp+CHAN]            ;y=y+(48xchan)
                shl     bx,1
                shl     bx,1
                shl     bx,1
        shl     bx,1
                mov     di,bx
                shl     bx,1
                add     bx,di

                add     ax,bx

        '       mov     es,ax




                mov     ax,EVGA_SCREEN_SEGMENT  ;address display memory
                mov     ds,ax


                mov     dx,GC_INDEX


                mov     al,GRAPHICS_MODE        ;set up write mode 0 and read mode 1
                out     dx,al           ;
                inc     dx              ;
;               in      al,dx           ;
                mov     al,00001000b    ;
;               jmp     $+2             ;
                out     dx,al           ;
                dec     dx              ;



                mov     al,SET_RESET_INDEX      ;set set/reset register to allways
                out     dx,al                   ;write pixels in colour
                inc     dx                              ;
                mov     al,[bp+Colour]          ;
```

```
        out     dx,al                   ;
        dec     dx                              ;
        mov     al,ENABLE_SET_RESET_INDEX       ;
        out     dx,al                   ;
        inc     dx                              ;
        mov     al,0fh                  ;
        out     dx,al                   ;


        dec     dx                              ;compare pixel colour with 15(light blue)
        mov     al,COLOUR_COMPARE    ;in read mode 1
        out     dx,al                   ;
        inc     dx                      ;
        mov     al,9                    ;15
        out     dx,al                   ;



        mov     si,es           ;(bp+Y1)                ;get deltaX
        mov     ax,[bp+Y0]              ;
        sub     si,ax                   ;
        jns     CalcStAddress           ;


        mov     ax,es           ;(bp+Y1)                ;if deltaX is -ve swap coords so that
        mov     dx,[bp+X0]              ;deltaY is allays +ve
        xchg    dx,[bp+X1]              ;
        mov     [bp+X0],dx             ;
        neg     si                              ;



CalcStAddress:
        shl     ax,1                            ;
        shl     ax,1                            ;
        shl     ax,1                            ;
        shl     ax,1                            ;
        mov     di,ax                   ;
        shl     ax,1                            ;
        shl     ax,1                            ;
        add     di,ax                   ;Y0 x 80
        mov     dx,[bp+X0]              ;
        mov     cl,dl                   ;pixel masking
        and     cl,7                            ;
        shr     dx,1                            ;
        shr     dx,1                            ;
        shr     dx,1                            ;byte address of column
        add     di,dx                   ;offset of line start in display segment


        mov     dx,GC_INDEX             ;set up GC index register to point to
        mov     al,BIT_MASK_INDEX       ;bit mask register and leave DX pointing
        out     dx,al                   ;to GC data register
        inc     dx                              ;

        mov     al,80h                  ;set up initial pixel mask
        shr     al,cl                   ;

        mov     bx,[bp+X1]             ;deltaX
        sub     bx,[bp+X0]             ;

        js      NDeltaX1                        ;handle correct line orientation
        cmp     bx,si                   ;
```

1468

```
                jb      Oct1                    ;

                LINE1 0                         ;
                jmp     PlotDataDone            ;

NDeltaX1:

                jmp     NDeltaX                 ;

Oct1:

                LINE2 0                         ;
                jmp     PlotDataDone            ;

NDeltaX:

                neg     bx                              ;
                cmp     bx,si                   ;
                jb      Oct2                            ;

                LINE1 1                         ;
                jmp     short PlotDataDone      ;


Oct2:

                LINE2 1                         ;


PlotDataDone:

                mov     al,0h                   ;0ffh in original listing
                out     dx,al                   ;
                dec     dx                              ;
                mov     al,ENABLE_SET_RESET_INDEX
                out     dx,al                   ;
                inc     dx                              ;
                sub     al,al                   ;
                out     dx,al                   ;




                mov     ax,es



                pop     es
                pop     ds
                pop     di
                pop     si
                pop     bp
                ret
; _PlotData              endp
PlotData                 endp


Line_TEXT               ends


;stackseg       segment para stack 'STACK'
;               db      512 dup (?)
;stackseg       ends

Data            segment word 'DATA'

data            ends
```

K69

end

K70

```
;program to load a mode 12h screen
;
;
Load_TEXT        segment byte   public  'code'
                 assume cs:Load_TEXT,ds:Data




EVGA_SCREEN_WIDTH_IN_BYTES      equ      80              ;640x480 pixels
EVGA_SCREEN_SEGMENT                      equ    0a000h   ;
GC_INDEX                                         equ    3ceh            ;graphics cont'r index
                                                                        ;reg'r port
SET_RESET_INDEX         equ      0                       ;GC register
ENABLE_SET_RESET_INDEX  equ      1                       ;    ''
BIT_MASK_INDEX                           equ    8                ;   ''
GRAPHICS_MODE                            equ    5                ;   ''
COLOUR_COMPARE                           equ    2                ;   ''
READ_MAP                                         equ    4
DISPLAYED_SCREEN_SIZE   equ      (640/8)*480
SC_INDEX                                         equ    3c4h
MAP_MASK                                         equ    2




;                public  _LoadScreen
                 public  LoadScreen
;                public  LoadScreen_0

;_LoadScreen     proc far
LoadScreen       proc far
;LoadScreen_0    proc far

                 push    bp
                 mov     bp,sp
                 push    si
                 push    di
                 push    ds
                 push    es


                 mov     ax,Data
                 mov     ds,ax

                 mov     dx,GC_INDEX
                 mov     al,BIT_MASK_INDEX
                 out     dx,al
                 inc     dx
                 mov     al,0ffh         .
                 out     dx,al



                 mov     ah,3dh
                 mov     dx,offset Filename
                 sub     al,al
                 int     21h

                 mov     [Handle],ax
                 jnc     RestoreTheScreen
                 mov     ah,9
                 mov     dx,offset ErrMsg1
                 int     21h
```

```
                jmp     short Done


        mov     dx,SC_INDEX
                mov     al,MAP_MASK
                out     dx,al
                inc     dx

                in      al,dx
                mov     [map],al

RestoreTheScreen:
                mov     [Plane],0               ;
RestoreLoop:
                mov     dx,SC_INDEX
                mov     al,MAP_MASK
                out     dx,al
                inc     dx

                mov     cl,[Plane]

                mov     al,1
                shl     al,cl

                out     dx,al
                mov     ah,3fh
                mov     bx,[Handle]
                mov     cx,DISPLAYED_SCREEN_SIZE
                sub     dx,dx
                push    ds
                mov     si,EVGA_SCREEN_SEGMENT
                mov     ds,si
                int     21h
                pop     ds
                jc      ReadError
                cmp     ax,DISPLAYED_SCREEN_SIZE
                jz      RestoreLoopBottom

ReadError:
                mov     ah,9
                mov     dx,offset ErrMsg3
                int     21h
                jmp     short DoClose

RestoreLoopBottom:
                mov     al,[Plane]
;               shl     ax,1
                inc     ax
                mov     [Plane],al
                cmp     al,3                    ;
                jbe     RestoreLoop

DoClose:

                mov     ah,3eh
                mov     bx,[Handle]
                int     21h

                mov     dx,SC_INDEX
                mov     al,MAP_MASK
                out     dx,al
                inc     dx

                mov     al,0fh
```

K72

```
                        out     dx,al


Done:
                        pop     es
                        pop     ds
                        pop     di
                        pop     si
                        pop     bp
                        ret
;_LoadScreen     endp
LoadScreen       endp
;LoadScreen_0    endp


Load_TEXT               ends


;stackseg        segment para stack 'STACK'
;                db      512 dup (?)
;stackseg        ends

Data             segment word 'DATA'
Filename                db      'Screen1.scr',0
ErrMsg1          db      '*** couldn''t open Screen1.scr **',0dh,0ah,'$'
ErrMsg3          db      '*** error reading Screen1.scr **',0dh,0ah,'$'
Handle           dw      ?
Plane            db      ?
map                     db      ?

data             ends


                        end
```

```
;ReadEEG.asm: to read in an EEG data file.
;
;input:
;                 far pointer to the start of the allocated memory.
;
;output:
;                 AX=0 for corrrect termination else error code.
```

K75

```
ReadEEG_TEXT    SEGMENT byte public 'CODE'

datgrp  GROUP   data,bss
                ASSUME  cs:ReadEEG_TEXT,ds:datgrp


CR              equ     00h             ;
LF              equ     0Ah             ;
FILE_SIZE       equ     65502   ;


;stack frame


InputDataParms          struc
                dw      ?                       ;far return address
                dw      ?                       ;  ''
                dw      ?                       ;bp

BUF_OFF dw      ?                       ;input data buffer offset address
BUF_SEG dw      ?                       ;input data buffer segment address

InputDataParms          ends



;               PUBLIC  _InputData
                PUBLIC  InputData

;_InputData proc         far
InputData proc  far

                push    bp
                mov     bp,sp
                push    si
                push    di
                push    ds


                mov     ax,data
                mov     ds,ax


Open:   mov     ah,3dh           ;dos interupt-open file
                mov     al,0
                mov     dx,offset Filename
                int     21h
                jc      NoFile
                mov     [Handle],ax

Input:  mov     ah,3fh           ;dos interrupt-read file
```

```
                        mov     bx,(Handle)
                        mov     cx,FILE_SIZE    ;number of bytes
                        push    ds
                        mov     ds,(bp+BUF_SEG) ;segment address of pointer
                        mov     dx,(bp+BUF_OFF) ;offset address of pointer
                        int     21h
                        pop     ds
                        jc      ReadErr


                        mov     ah,3Eh          ;dos interrupt-close file
                        mov     bx,(Handle)
                        int     21h

                        mov     ax,0                    ;return correct termination code


                        jmp     InputDataEnd

NoFile: mov     dx,offset ErrMsg1       ;file does not exist
                        mov     ah,9
                        int     21h
                        mov     ax,1                            ;return error code
                        jmp     InputDataEnd

ReadErr:        mov     dx,offset ErrMsg2       ;file read error.
                        mov     ah,9
                        mov     ax,2                            ;return error code
                        int     21h


InputDataEnd:

                        pop     ds                              ;restore regs.
                        pop     di
                        pop     si
                        pop     bp
                        ret

;_InputData     ENDP
InputData       ENDP

ReadEEG_TEXT            ENDS


data            SEGMENT byte PUBLIC 'data'

Filename        db      'd116.dat',0
Handle  dw      0
ErrMsg1         db      CR,LF,'InputData: file not found',CR,LF,'$'
ErrMsg2         db      CR,LF,'InputData: file read error',CR,LF,'$'

data            ENDS

bss             segment word public 'bss'

bss             ENDS


                        END
```

```
;ReadKBD.asm: to read any character from the keyboard.
;
;
;input    none.
;
;output al contains the keypressed.


ReadKBD_TEXT    SEGMENT byte public 'CODE'

        datgrp GROUP    data,bss
        ASSUME  cs:ReadKBD_TEXT,ds:datgrp


CR          equ     00h             ;
LF          equ     0Ah             ;

;           PUBLIC  _ReadKey
            PUBLIC  ReadKey


;_ReadKey   proc    far
ReadKey proc    far

        push    bp                      ;save registers
        mov     bp,sp
        push    si
        push    di
        push    ds

            .

        mov     ax,data         ;point to new data segment
        mov     ds,ax



        mov     ah,8                    ;character input without echo
        int     21h

        cmp     al,0                    ;extended key?
        jnz     ReadKeyEnd      ;no - output character in al


;           mov     ah,9
;           mov     dx,offset Msg1
;           int     21h

        mov     ah,8                    ;yes - repeat call to get extended code
        int     21h

ReadKeyEnd:
        sub     ah,ah

        pop     ds                                  ;restore regs.
        pop     di
        pop     si
        pop     bp
        ret
```

```
;_ReadKey        ENDP
ReadKey ENDP

ReadKBD_TEXT              ENDS




data             SEGMENT byte PUBLIC 'data'

Msg1             db      CR,LF,'extended','$'

data                     ENDS

bss              segment word public 'bss'

bss                      ENDS



                         END




                         END
```

K78

```
; FILE: scrorig.asm. This procedure controls horizontal and vertical
; pixel panning on the VGA monitor.  The procedure is called from C
; as:-
;               ScreenOrigin(int x,int y);  /* x and y are the coordinates
;                                                   of the screen origin */
;


AC_INDEX                equ     3C0h    ;attribute controller index register
INPUT_STATUS_1 equ      3DAh    ;colour mode address of the
                                        ;input status 1 register.



scrorig_TEXT    SEGMENT byte public 'CODE'

                ASSUME  cs:scrorig_TEXT



;               PUBLIC  _ScreenOrigin
                PUBLIC  ScreenOrigin

;_ScreenOrigin  PROC    far
ScreenOrigin    PROC    far

                push    bp
                mov     bp,sp


;
;               mov     ax,40h
;               mov     es,ax
;
;               mov     dx,es:[63h]
;               mov     al,13h
;               cli
;               out     dx,al
;               jmp     $+2
;
;               inc     dx
;               mov     al,50h
;               out     dx,al
;               sti
;


; setup for pixel x-coordinate

;               mov     dx,3C4h
;               mov     al,1
;               cli
;               out     dx,al
;               jmp     $+2

;               inc     dx
;               in      al,dx
;               sti
;               and     al,1
;               mov     cl,9
;               sub     cl,al

                mov     cl,8            ;?????
```

```
              mov     ax,[bp+6]
              div     cl

;             cmp     cl,8
;             je      LO1

;             dec     ah
;             jns     LO1
;             mov     ah,8

LO1:          mov     cl,ah
              mov     bl,al
              xor     bh,bh


; setup pixel y-coordinate

              mov     ax,40h
              mov     es,ax

              mov     dx,es:[63h]
              mov     al,9
              cli
              out     dx,al

              push    bx
              push    dx

              inc     dx
              in      al,dx
              sti

              and     ax,1fh
              inc     ax
              mov     bx,ax

              xor     dx,dx
              mov     ax,[bp+8]
              div     bx

              mov     ch,dl
              mov     bx,ax

              pop     dx
              push    dx
              mov     al,13h
              cli
              out     dx,al
              jmp     $+2

              inc     dx
              in      al,dx
              sti

              xor     ah,ah

              mul     bx
              shl     ax,1

              pop     dx
              pop     bx
```

K 80

```
                        add     bx,ax

                        add     dl,6

; update CTRC start address registers

L02:            in      al,dx
                test    al,8
                jz      L02

L03:    in      al,dx
                test    al,8
                jnz     L03

                cli
                sub     dl,6

                mov     ah,bh
                mov     al,0ch
                out     dx,ax

                mov     ah,bl
                inc     al
                out     dx,ax
                sti

; update CRTC preset row scan and
;attribute controller horizontal pel pan registers

                add     dl,6
L04:            in      al,dx   ;insert for EGA
                test    al,8            ;insert for EGA
                jnz     L04             ;insert for EGA

L05:    in      al,dx   ;
                test    al,8            ;
                jnz     L05             ;

                cli
                sub     dl,6
                mov     ah,ch
                mov     al,8
                out     dx,ax

                mov     dl,0C0h
                mov     al,13h OR 20h

                out     dx,al

                mov     al,cl
                out     dx,al
                sti

                mov     sp,bp
                pop     bp
                ret

;_ScreenOrigin  ENDP
ScreenOrigin    ENDP
```

```
;               PUBLIC   _SetWidth
                PUBLIC   SetWidth


;_SetWidth      PROC     far
SetWidth        PROC     far

                push     bp
                mov      bp,sp


                mov      ax,40h
                mov      es,ax

                mov      dx,es:[63h]
                mov      al,13h
                cli
                out      dx,al
                jmp      $+2

                inc      dx
                mov      al,40
                out      dx,al
                sti

                mov      sp,bp
                pop      bp
                ret

;_SetWidth      ENDP
SetWidth        ENDP




;               PUBLIC   _SplitScreen
                PUBLIC   SplitScreen


;_SplitScreen   proc     far
SplitScreen     proc     far

                push     bp
                mov      bp,sp

                mov      ax,40h
                mov      es,ax
                mov      dx,es:[63h]

;wait for vertical retrace

                add      dl,6
HI1:            in       al,dx
                test     al,8
                jnz      HI1

HI2:            in       al,dx
                test     al,8
                jz       HI2
                sub      dl,6

;isolate bits 0-7, bit 8 and bit 9 of the line compare value

                mov      ax,[bp+6]
                mov      bh,ah
```

K82

```
                mov     bl,bh
                and     bx,0201h

                mov     cl,4
                shl     bx,cl

                shl     bh,1

;update the CRTC registers

                mov     ah,al
                mov     al,18h
                out     dx,ax

                mov     al,7
                cli
                out     dx,al
                inc     dx
                in      al,dx
                sti
                dec     dx

                mov     ah,al
                and     ah,11101111b
                or      ah,bl
                mov     al,7
                out     dx,ax

                mov     al,9
                cli
                out     dx,al
                inc     dx
                in      al,dx
                sti
                dec     dx

                mov     ah,al
                and     ah,10111111b
                or      ah,bh
                mov     al,9
                out     dx,ax

;set bit 5 of the attribute controller mode control register

                mov     ax,1007h
                mov     bl,10h
                int     10h
                or      bh,20h

                mov     ax,1000h
                mov     bl,10h
                int     10h

                mov     sp,bp
                pop     bp

                ret

;_SplitScreen   ENDP
SplitScreen     ENDP


;               PUBLIC  _BlankScreen
                PUBLIC  BlankScreen
```

K83

```
;_BlankScreen    proc    far
BlankScreen      proc    far


                 push    bp
                 mov     bp,sp

                 mov     dx,INPUT_STATUS_1
                 in      al,dx    ;reset port 3c0h to index

                 mov     dx,AC_INDEX
                 sub     al,al    ;set bit 5 to zero
                 out     dx,al    ;blank the screen

                 mov     sp,bp
                 pop     bp

                 ret

;_BlankScreen    ENDP
BlankScreen      ENDP



;                PUBLIC  _ShowScreen
                 PUBLIC  ShowScreen

;_ShowScreen     proc    far
ShowScreen       proc    far


                 push    bp
                 mov     bp,sp

                 mov     dx,INPUT_STATUS_1
                 in      al,dx    ;reset port 3c0h to index

                 mov     dx,AC_INDEX
                 mov     al,20H   ;set bit 5 to one
                 out     dx,al    ;blank the screen

                 mov     sp,bp
                 pop     bp

                 ret

;_ShowScreen     ENDP
ShowScreen       ENDP



scrorig_TEXT     ENDS



                 END
```

K84

```
;VidMode.asm to set the graphics video mode
;
;
;called as SetVideoMode(0x??); - (c) where ?? is the hex mode number
;
;input int mode.
;
;output none.



VidMode_TEXT    SEGMENT byte public 'CODE'

                ASSUME cs:VidMode_TEXT




;stack frame



VideoModeParms          struc                      .
                        dw      ?               ;far return address
                        dw      ?               ; ""
                        dw      ?               ;bp

MODE                    dw      ?               ;Video mode
VideoModeParms          ends




;                       PUBLIC  _SetVideoMode
                        PUBLIC  SetVideoMode

;_SetVideoMode   proc    far
SetVideoMode    proc    far

                        push    bp              ;save registers
                        mov     bp,sp

                        mov     ax,[bp+MODE]    ;
                        int     10h                         ;set video mode


                        pop     bp
                        ret

;_SetVideoMode   ENDP
SetVideoMode    ENDP


VidMode_TEXT    ENDS

                        END
```

APPENDIX M

This appendix contains a sample of the initial rule set obtained from the data represented in appendix D using the knowledge elicitation techniques given in appendix I and acquired from the DAS (appendix B). The rule set is written in english to allow verification by the expert and elicitaion of expert rule confidence values. These rules are transcribed in to a logic representation and PROLOG encoded to operate in the IOARS.

Rule 1: segment contains no significant activity
   if the EEG is flat

Rule 2: segment contains no slow waves
   if no delta band waves exist

Rule 3: segment contains artefact only
   if the largest waves appear in a frontal channel
   and fp2-f4 and fp1-f3 are symmetrical
   and fp2-f8 and fp1-f7 are symmetrical
   and delta activity only appears in frontal channels

Rule 4: segment contains artefact only
   if the largest waves appear in a frontal channel
   and fp2-f4 and fp1-f3 are symmetrical
   and delta activity only appears in frontal channels

Rule 5: segment contains artefact only
   if the largest waves appear in a frontal channel
   and fp2-f4 and fp1-f3 are symmetrical
   and fp2-f8 and fp1-f7 are symmetrical
   and delta activity only appears in the anterior half of the scalp
   and no delta waves are present in the EEG that are not present in the EOG
   and fp2-f4 or fp1-f3 contains isolated non repetitive waveforms
   and fp2-f4 or fp1-f3 contains waveforms with sharp elements

Rule 6: segment contains artefact only
   if the largest waves appear in a frontal channel
   and fp2-f4 and fp1-f3 are symmetrical
   and fp2-f8 and fp1-f7 are symmetrical
   and delta activity only appears in the anterior half of the scalp
   and no delta waves are present in the EEG that are not present in the EOG

Rule 7: segment contains artefact only
   if the largest waves appears in a frontal channel
   and fp2-f4 and fp1-f3 are symmetrical
   and delta activity only appears in the anterior half of the scalp
   and no delta waves are present in the EEG that are not present in the EOG
   and fp2-f4 or fp1-f3 contains isolated non repetitive waveforms
   and fp2-f4 or fp1-f3 contains waveforms with sharp elements

Rule 8: segment contains artefact only
   if the largest waves appear in a frontal channel
   and fp2-f4 and fp1-f3 are symmetrical
   and delta activity only appears in the anterior half of the scalp
   and no delta waves are present in the EEG that are not present in the EOG

Rule 9: segment contains artefact only
   if the largest waves appear in a frontal channel
   and fp2-f4 and fp1-f3 are symmetrical
   and fp2-f8 and fp1-f7 are symmetrical
   and delta activity does not appear in occipital channels
   and no delta waves are present in the EEG that are not present in the EOG
   and fp2-f4 or fp1-f3 contains isolated non repetitive waveforms
   and fp2-f4 or fp1-f3 contains waveforms with sharp elements

Rule 10: segment contains artefact only
    if the largest waves appear in a frontal channel
    and fp2-f4 and fp1-f3 are symmetrical
    and delta activity does not appear in occipital channels
    and no delta waves are present in the EEG that are not present in the EOG
    and fp2-f4 or fp1-f3 contains isolated non repetitive waveforms
    and fp2-f4 or fp1-f3 contains waveforms with sharp elements

Rule 11: segment contains artefact only
    if the largest wave appear in a frontal channel
    and delta activity does not appear in occipital channels
    and no delta waves are present in the EEG that are not present in the EOG
    and fp2-f4 or fp1-f3 contains isolated non repetitive waveforms
    and fp2-f4 or fp1-f3 contains waveforms with sharp elements

Rule 12: segment contains artefact only
    if the largest waves appear in a frontal channel
    and the largest wave is below 1/2 Hz in frequency
    and no delta waves are present in the EEG that are not present in the EOG

Rule 13: segment contains both artefact and pathological slow waves
    if the largest waves appear in a frontal channel
    and fp2-f4 and fp1-f3 are symmetrical
    and delta activity does appear in occipital channels
    and no delta peaks are present in the EEG that are not present in the EOG
    and fp2-f4 or fp1-f3 contains waveforms with sharp elements

Rule 14: segment contains both artefact and pathological slow waves
    if the largest waves appear in a frontal channel
    and fp2-f4 and fp1-f3 are symmetrical
    and fp2-f4 or fp1-f3 contains waveforms with sharp elements
    and fp2-f4 or fp1-f3 do not contain isolated non repetitive waveforms
    and no delta waves are present in the EEG that are not present in the EOG

Rule 15: segment contains both artefact and pathological slow waves
    if the largest waves appear in a frontal channel
    and fp2-f4 and fp1-f3 are symmetrical
    and delta waves are present in the EEG that are not present in the EOG

Rule 16: segment contains both artefact and pathological slow waves
    if fp2-f4 and fp1-f3 are symmetrical
    and fp2-f4 or fp1-f3 contains waveforms with sharp elements
    and fp2-f4 or fp1-f3 do not contain isolated non repetitive waveforms
    and no delta waves are present in the EEG that are not present in the EOG

Rule 17: segment contains both artefact and pathological slow waves
    if fp2-f4 and fp1-f3 are symmetrical
    and delta activity does appear in occipital channels
    and fp2-f4 or fp1-f3 contains waveforms with sharp elements

Rule 18: segment contains pathological slow waves only
    if any of the above are not satisfied

Rule 19: artefact only contains eye movements
    if the wave is attributable to more than one electrode
    and the wave is more than 1/2 hz in frequency

Rule 20: artefact only contains non EM artefact
    if the above is not true

Rule 21: eye movements contains vem
    if the largest waves appear in channels fp2-f4 or fp1-f3
    and the waveform in channels fp2-f4 or fp1-f3 is slow
    and the waveform in channels fp2-f8 and f8-t4 are synchronous
    and the waveform in channels fp1-f7 and f7-t3 are synchronous

Rule 22: eye movements contains blink
    if the largest waves appear in channels fp2-f4 or fp1-f3
    and the waveform in channels fp2-f4 or fp1-f3 is fast
    and the waveform in channels fp2-f8 and f8-t4 are synchronous
    and the waveform in channels fp1-f7 and f7-t3 are synchronous

Rule 23: eye movements contains hem
    if the largest waves appear in channels fp2-f8, fp1-f7, f8-t4 or f7-t3
    and the waveform in channels fp2-f4 or fp1-f3 is slow
    and the waveform in channels fp2-f8 and f8-t4 are not synchronous
    and the waveform in channels fp1-f7 and f7-t3 are not synchronous

Rule 24: eye movements contains complex eye movements
    if the above are not satisfied

M3

APPENDIX N

This appendix contains the PROLOG encoded transcribed rules given in appendix M. The knowledge base consistes of the following sections:

- textual rules - provide rule explanation facilities and contain the list of conditions for rule satisfaction.
- textual conditions - provides the condition explanation facilities.
- numerical conditions - provide the feature extraction macro routines to interrogate the dynamic feature knowledge.
- feature deamons - provide the low level feature interrogation routines.
- contextual facts - provide the environmental information for EEG analysis.

rule(1,0.5,"segment1","segment2",[63])
rule(2,0,"segment1","segment3",[999])
rule(3,0.6,"segment2","segment5",[61])
rule(4,0.5,"segment2","segment5",[57])
rule(5,0.4,"segment2","segment5",[59])
rule(6,0.1,"segment2","segment5",[69])
rule(7,0,"segment2","segment3",[999])
rule(8,0.4,"segment3","segment5",[61])
rule(9,0.3,"segment3","segment5",[57])
rule(10,0.5,"segment3","segment4",[59])
rule(11,0.4,"segment3","segment4",[57])
rule(12,1,"segment3","no significant activity",[2])
rule(13,0.8,"segment3","only very slow artefact",[39])
rule(14,1,"segment3","no significant delta waves",[4])
rule(15,0.93,"segment3","artefact only",[5,7,9,11])
rule(16,0.67,"segment3","artefact only",[5,7,11,73])
rule(17,0.67,"segment3","artefact only",[5,9,11])
rule(18,0.86,"segment3","artefact only",[5,7,9,13,16,17,19])
rule(19,0.5,"segment3","artefact only",[5,7,9,13,16])
rule(20,0.67,"segment3","artefact only",[5,7,13,16,17,19])
rule(21,0.5,"segment3","artefact only",[5,7,13,16])
rule(22,0.5,"segment3","artefact only",[5,7,9,16,17,19,21])
rule(23,0.4,"segment3","artefact only",[5,16,17,19,21])
rule(24,0.4,"segment3","artefact only",[5,16,13])
rule(25,0,"segment3","segment4",[999])
rule(26,0.9,"segment4","both artefact and abnormal slow waves",[5,7,16,19,22])
rule(27,0.8,"segment4","both artefact and abnormal slow waves",[5,7,16,19,18])
rule(28,0.7,"segment4","both artefact and abnormal slow waves",[17,22,53])
rule(29,0.9,"segment4","both artefact and abnormal slow waves",[5,7,15])
rule(30,0.7,"segment4","both artefact and abnormal slow waves",[7,16,17,19])
rule(31,0.8,"segment4","both artefact and abnormal slow waves",[7,19,22])
rule(32,0.8,"segment4","both artefact and abnormal slow waves",[6,7,15,23])
rule(33,0.5,"segment4","both artefact and abnormal slow waves",[16,53])
rule(34,0.5,"segment4","both artefact and abnormal slow waves",[5,16,18])
rule(35,0,"segment4","segment5",[999])
rule(36,0.7,"both artefact and abnormal slow waves","mainly_OA with little abnormal slow waves",[5,9,17])
rule(37,0.6,"both artefact and abnormal slow waves","mainly_OA with little abnormal slow waves",[5,7,17])
rule(38,0.5,"both artefact and abnormal slow waves","mainly_OA with little abnormal slow waves",[5,18,66])
rule(39,0.7,"both artefact and abnormal slow waves","detectable_OA in abnormal slow waves",[18,56])
rule(40,0.9,"both artefact and abnormal slow waves","mainly_abnormal slow waves with insignificant OA",[18])
rule(41,0.8,"segment5","abnormal slow waves only",[3,25,42])
rule(42,0.6,"segment5","abnormal slow waves only",[3,42,75])
rule(43,0.5,"segment5","abnormal slow waves only",[3,18,71])
rule(44,0.4,"segment5","abnormal slow waves only",[3,17,71,75])
rule(45,0.7,"artefact only","eye movements",[67])
rule(46,0.8,"artefact only","eye movements",[9,25,53])
rule(47,0.6,"artefact only","eye movements",[7,16,25,41])
rule(48,0.4,"artefact only","eye movements",[9,16,25,41])
rule(49,0.3,"artefact only","eye movements",[53,25])
rule(50,0.9,"only very slow artefact","rem artefact only",[51,25])
rule(51,0.8,"only very slow artefact","electrode artefact",[52])
rule(52,0.9,"eye movements","vem artefact",[29,31,33,35])
rule(53,0.9,"eye movements","blink artefact",[29,32,33,35])
rule(54,0.9,"eye movements","hem artefact",[30,31,34,36])
rule(55,0.7,"eye movements","blink artefact",[32,33,35])
rule(56,0.6,"eye movements","weak em artefact",[53])
rule(57,0.6,"eye movements","complex eye movements",[5,13])
tex_cond(1,"any significant spectral peaks exist")
tex_cond(2,"not any significant spectral peaks exist")
tex_cond(3,"any significant spectral peaks exist in the delta band")
tex_cond(4,"not any significant spectral peaks exist in the delta band")
tex_cond(5,"the largest spectral peak appears in a frontal channel")
tex_cond(6,"not the largest spectral peak appears in a frontal channel")
tex_cond(7,"fp2-f4 and fp1-f3 are symmetrical for all delta peaks")

tex_cond(8,"not fp2-f4 and fp1-f3 are symmetrical for all delta peaks")
tex_cond(9,"fp2-f8 and fp1-f7 are symmetrical for all delta peaks")
tex_cond(10,"not fp2-f8 and fp1-f7 are symmetrical for all delta peaks")
tex_cond(11,"delta activity only appears in frontal channels")
tex_cond(12,"not delta activity only appears in frontal channels")
tex_cond(13,"delta activity only appears in the anterior half of the scalp")
tex_cond(14,"not delta activity only appears in the anterior half of the scalp")
tex_cond(15,"delta peaks are present that are not present in the EOG")
tex_cond(16,"not delta peaks are present that are not present in the EOG")
tex_cond(17,"fp2-f8 or fp1-f7 contains isolated non repetitive waveforms")
tex_cond(18,"not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms")
tex_cond(19,"fp2-f4 or fp1-f3 contains waveforms with sharp elements")
tex_cond(20,"not fp2-f4 or fp1-f3 contains waveforms with sharp elements")
tex_cond(21,"significant delta activity does not appear in occipital channels")
tex_cond(22,"not significant delta activity does not appear in occipital channels")
tex_cond(23,"the largest spectral peak exists in the sub-delta band")
tex_cond(24,"not the largest spectral peak exists in the sub-delta band")
tex_cond(25,"the largest spectral peak is attributable to more than one electrode")
tex_cond(26,"not the largest spectral peak is attributable to more than one electrode")
tex_cond(27,"spectral activity does not exists below 1Hz")
tex_cond(28,"not spectral activity does not exists below 1Hz")
tex   nd(29,"the largest delta peak appears in channels fp2-f4 or fp1-f3")
tex_   id(30,"not the largest delta peak appears in channels fp2-f4 or fp1-f3")
tex_cond(31,"the waveform in channels fp2-f4 or fp1-f3 is slow -low delta")
tex_cond(32,"not the waveform in channels fp2-f4 or fp1-f3 is slow -low delta")
tex_cond(33,"the waveform in channels fp2-f8 and f8-t4 are in phase")
tex_cond(34,"not the waveform in channels fp2-f8 and f8-t4 are in phase")
tex_cond(35,"the waveform in channels fp1-f7 and f7-t3 are in phase")
tex_cond(36,"not the waveform in channels fp1-f7 and f7-t3 are in phase")
tex_cond(37,"there is a significant spectral peak recorded atFP2-FP1")
tex_cond(38,"not there is a significant spectral peak recorded atFP2-FP1")
tex_cond(39,"there is only very slow activity present, less than 0.5Hz")
tex_cond(40,"not there is only very slow activity present, less than 0.5Hz")
tex_cond(41,"there is a significant spectral peak in an EOG channel")
tex_cond(42,"not there is a significant spectral peak in an EOG channel")
tex_cond(43,"the EOG is symetrical")
tex_cond(44,"not the EOG is symetrical")
tex_cond(45,"the spectral peak is localised to the temporal regions")
tex_cond(46,"not the spectral peak is localised to the temporal regions")
tex_cond(47,"correction of artefacts is required")
tex   d(48,"not correction of artefacts is required")
tex_   nd(49,"there is L-R symetrical low freq temporal activity")
tex_cond(50,"not there is L-R symetrical low freq temporal activity")
tex_cond(51,"the largest spectral peak appears anteriorly")
tex_cond(52,"not the largest spectral peak appears anteriorly")
tex_cond(53,"the largest spectral peak appears in the EOG")
tex_cond(54,"not the largest spectral peak appears in the EOG")
tex_cond(55,"there is a phase delay in frontal channels")
tex_cond(56,"not there is a phase delay in frontal channels")
tex_cond(57,"the local context is mainly abnormal slow waves with insignificant OA")
tex_cond(58,"not the local context is mainly abnormal slow waves with insignificant OA")
tex_cond(59,"the local context is detectable_OA in abnormal slow waves")
tex_cond(60,"not the local context is detectable_OA in abnormal slow waves")
tex_cond(61,"the local context is abnormal slow waves only")
tex_cond(62,"not the local context is abnormal slow waves only")
tex_cond(63,"the historical context is abnormal")
tex_cond(64,"not the historical context is abnormal")
tex_cond(65,"there is a zero lag correlation between frontal and posterior electrodes")
tex_cond(66,"not there is a zero lag correlation between frontal and posterior electrodes")
tex_cond(67,"delta spectral peaks exist only in the EOG")
tex_cond(68,"not delta spectral peaks exist only in the EOG")
tex_cond(69,"a local context exists")
tex_cond(70,"not a local context exists")
tex_cond(71,"there is a correlation between frontal and posterior electrodes")

tex_cond(72,"not there is a correlation between frontal and posterior electrodes")
tex_cond(73,"there are l/r symetrical delta peaks")
tex_cond(74,"not there are l/r symetrical delta peaks")
tex_cond(75,"there is a repetitive signal")
tex_cond(76,"not there is a repetitive signal")
num_cond(1,["f_feature","_","_","_","_","_","_","_"])
num_cond(3,["f_feature","_","_","_","delta","_","_","_"])
num_cond(5,["max_front"])
num_cond(8,["peak_asym","fp2-f4","fp1-f3"])
num_cond(10,["peak_asym","fp2-f8","fp1-f7"])
num_cond(12,["n_o_front"])
num_cond(14,["post_feat"])
num_cond(15,["non_eog"])
num_cond(18,["repetitive_eog"])
num_cond(19,["front_fast_delta"])
num_cond(22,["occ_feat"])
num_cond(23,["vslow_larg"])
num_cond(25,[">1_elec"])
num_cond(28,["f_feature","_","_","_","sub_delta","_","_","_"])
num_cond(29,["max_vert"])
num_cond(31,["front_slow_delta"])
num_cond(37,["f_feature","fp2-fp1","_","_","_","_","_","_"])
num_cond(40,["n_o_Vslow"])
num_cond(41,["sig_eog"])
num_cond(43,["lr_sym_eog"])
num_cond(46,["not_loc_temp"])
num_cond(49,["sym_lf","temp"])
num_cond(51,["max_anter"])
num_cond(53,["max_eog"])
num_cond(55,["front_phase_lag"])
num_cond(57,["c_feature","local","mainly_abnormal"])
num_cond(59,["c_feature","local","detectable_OA"])
num_cond(61,["c_feature","local","abnormal"])
num_cond(63,["c_feature","historical","abnormal"])
num_cond(65,["front-post_0_corr"])
num_cond(68,["more_than_EOG_delta"])
num_cond(69,["c_feature","local","_"])
num_cond(71,["front-post_corr"])
num_cond(73,["repetitive_ac"])
cl(["peak_asym","A","B"],[],[["f_feature","A","_","_","delta","M","_","_"],["asym","B","M"],["int_pred","not_sym_peak","M","_","_"]])
cl(["peak_asym","A","B"],[],[["f_feature","B","_","_","delta","M","_","_"],["asym","A","M"],["int_pred","not_sym_peak","M","_","_"]])
cl(["sym","Chan","M"],[],[["f_feature","Chan","_","_","delta","M","_","_"],["int_pred","ass_sym_peak","M","_","_"],["int_pred","cutback","_","_","_"]])
cl(["asym","Chan","M"],[],[["int_pred","true","_","_","_"]])
cl(["n_o_front"],[],[["f_feature","C","_","_","_","_","_","_"],["frontal","C"],["int_pred","not_peak","_","_","C"]])
cl(["frontal","C"],[],[["front","C"],["int_pred","assert_peak","peak","C"],["int_pred","cutback","_","_","_"]])
cl(["frontal","C"],[],[["int_pred","true","_","_","_"]])
cl(["post_feat"],[],[["f_feature","C","_","_","delta","_","_","_"],["posterior","C"]])
cl(["occ_feat"],[],[["f_feature","C","_","_","delta","_","_","_"],["occipital","C"]])
cl(["g_t_2"],[],[["f_feature","A","_","_","B","_","_","_"],["f_feature","C","_","_","D","_","_","_"],["int_pred","< >","A","C"],["int_pred"," = ","B","D"]])
cl(["g_t_2_lf"],[],[["f_feature","A","_","_","sub_delta","_","_","_"],["f_feature","C","_","_","sub_delta","_","_","_"],["int_pred","< >","A","C"]])
cl(["non_eog"],[],[["int_pred","channel","X","eeg"],["f_feature","X","_","_","delta","_","_","A"],["int_pred","< >","X","fp2-f4"],["int_pred","< >","X","fp1-f3
cl(["any_eog","A"],[],[["int_pred","channel","Y","eog"],["f_feature","Y","_","_","delta","_","_","A"],["int_pred","assert_peak","peak","A"],["int_pred","cutback"
cl(["any_eog","A"],[],[["int_pred","true","_","_","_"]])
cl(["n_o_Vslow"],[],[["f_feature","_","_","_","A","_","_","_"],["int_pred","< >","A","sub_delta"]])
cl(["sig_eog"],[],[["int_pred","channel","X","eog"],["f_feature","X","_","_","_","_","_","_"]])
cl(["lr_sym_eog"],[],[["f_feature","fp2-f8","_","_","A","B","C"],["f_feature","fp1-f7","_","_","A","B","C"]])
cl(["vslow_larg"],[],[["int_pred","read_lrg","C","V"],["f_feature","C","_","_","B","V","_"],["int_pred"," = ","B","sub_delta"]])
cl(["max_front"],[],[["int_pred","ass_lrg","o2","a"],["largest","_","_"],["int_pred","read_lrg","C","V"],["front","C"]])
cl(["max_anter"],[],[["int_pred","ass_lrg","o2","a"],["largest","_","_"],["int_pred","read_lrg","C","V"],["anterior","C"]])
cl(["max_vert"],[],[["int_pred","read_lrg","C","V"],["fp_vert","C"]])
cl(["max_eog"],[],[["int_pred","read_lrg","C","V"],["int_pred","channel","C","eog"]])
cl([">1_elec"],[],[["int_pred","ass_lrg","o2","a"],["largest","_","_"],["int_pred","read_lrg","C","V"],["f_feature","C","_","_","B","V","_"],["another_elec","B","_
cl(["another_elec","B","R"],[],[["f_feature","D","_","_","B","_","_","_"],["int_pred","< >","C","D"],["int_pred","nt_adj","C","D"],["int_pred","nt_adj","D","C"]
cl(["largest","Chan"],[],[["int_pred","channel","X","Chan"],["f_feature","X","_","_","_","_","A","_"],["int_pred","read_lrg","C","V"],["int_pred"," > ","A","V"],[

```
cl(["largest","_"],[],[["int_pred","true","_","_","_"]])
cl(["sym_lr","Region"],[],[["f_feature","X","_","_","sub_delta","_","_","_"],["Region","X"],["int_pred","lr_sym","X","Y"],["f_feature","Y","_","_","sub_delta
cl(["front_slow_delta"],[],[["front","X"],["f_feature","X","_","_","delta","_","_","low_speed"]])
cl(["front_fast_delta"],[],[["front","X"],["f_feature","X","_","_","delta","_","_","high_speed"]])
cl(["more_than_EOG_delta"],[],[["f_feature","X","_","_","delta","_","_","_"],["int_pred","channel","X","eeg"]])
cl(["repetitive_cog"],[],[["ac_feature","fp2-f8","_","_","_","_","high_cycles"]])
cl(["repetitive_cog"],[],[["ac_feature","fp1-f7","_","_","_","_","high_cycles"]])
cl(["repetitive_ac"],[],[["ac_feature","_","_","_","_","_","high_cycles"]])
cl(["front_phase_lag"],[],[["cc_feature","fp2-f4:f4-c4","lag_phase","high_similarity","_","_","_"]])
cl(["front_phase_lag"],[],[["cc_feature","fp1-f3:f3-c3","lag_phase","high_similarity","_","_","_"]])
cl(["front-post_0_corr"],[],[["cc_feature","fp2-f4:p4-o2","none_phase","_","_","_","_"]])
cl(["front-post_0_corr"],[],[["cc_feature","fp1-f3:p3-o1","none_phase","_","_","_","_"]])
cl(["front-post_corr"],[],[["cc_feature","fp2-f4:p4-o2","_","_","_","_","_"]])
cl(["front-post_corr"],[],[["cc_feature","fp1-f3:p3-o1","_","_","_","_","_"]])
cl(["anterior","fp2-f4"],[1],[])
cl(["anterior","f4-c4"],[1],[])
cl(["anterior","fp1-f3"],[1],[])
cl(["anterior","f3-c3"],[1],[])
cl(["anterior","fp2-f8"],[1],[])
cl(["anterior","f8-t4"],[1],[])
cl(["anterior","fp1-f7"],[1],[])
cl(["anterior","f7-t3"],[1],[])
cl(["anterior","fp2-fp1"],[1],[])
cl(["posterior","c4-p4"],[1],[])
cl(["posterior","p4-o2"],[1],[])
cl(["posterior","c3-p3"],[1],[])
cl(["posterior","p3-o1"],[1],[])
cl(["posterior","t4-t6"],[1],[])
cl(["posterior","t6-o2"],[1],[])
cl(["posterior","t3-t5"],[1],[])
cl(["posterior","t5-o1"],[1],[])
cl(["front","fp2-f4"],[1],[])
cl(["front","fp1-f3"],[1],[])
cl(["front","fp2-f8"],[1],[])
cl(["front","fp1-f7"],[1],[])
cl(["front","fp2-fp1"],[1],[])
cl(["fp_vert","fp2-f4"],[1],[])
cl(["fp_vert","fp1-f3"],[1],[])
cl(["temp","f8-t4"],[1],[])
cl(["temp","t4-t6"],[1],[])
cl(["temp","f7-t3"],[1],[])
cl(["temp","t3-t5"],[1],[])
cl(["occipital","p4-o2"],[1],[])
cl(["occipital","p3-o1"],[1],[])
cl(["occipital","t6-o2"],[1],[])
cl(["occipital","t5-o1"],[1],[])
cl(["c_feature","class_totals"],[0],[])
channel(16,"fp2-f8","cog")
channel(12,"fp1-f7","cog")
channel(0,"fp2-f4","eeg")
channel(4,"fp1-f3","eeg")
channel(1,"f4-c4","eeg")
channel(2,"c4-p4","eeg")
channel(3,"p4-o2","eeg")
channel(5,"f3-c3","eeg")
channel(6,"c3-p3","eeg")
channel(7,"p3-o1","eeg")
channel(9,"f8-t4","eeg")
channel(10,"t4-t6","eeg")
channel(11,"t6-o2","eeg")
channel(13,"f7-t3","eeg")
channel(14,"t3-t5","eeg")
channel(15,"t5-o1","eeg")
channel(8,"fp2-f8","cog")
```

```
channel(12,"fp1-f7","eog")
autocorr(0)
autocorr(4)
autocorr(8)
autocorr(12)
crosscorr(0,1,21)
crosscorr(0,2,22)
crosscorr(0,3,23)
crosscorr(4,5,24)
crosscorr(4,6,25)
crosscorr(4,7,26)
adjacent("fp2-f4","f4-c4")
adjacent("fp2-f4","fp2-f8")
adjacent("fp1-f3","f3-c3")
adjacent("fp1-f3","fp3-f7")
adjacent("f4-c4","fp2-f4")
adjacent("f4-c4","c4-p4")
adjacent("f3-c3","fp1-f3")
adjacent("f3-c3","c3-p3")
adjacent("f8-t4","fp2-f8")
adjacent("f8-t4","t4-t6")
ad     t("f7-t3","fp1-f7")
adj  ...t("f7-t3","t3-t5")
lr_sym("fp2-f8","fp1-f7")
lr_sym("f8-t4","f7-t3")
lr_sym("fp2-f4","fp1-f3")
peak(1,"pk1")
peak(2,"pk2")
peak(3,"pk3")
peak(4,"pk4")
freq_bounds("sub_delta",-4,6)
freq_bounds("delta",2,18)
freq_bounds("theta",14,34)
freq_bounds("alpha",30,54)
freq_bounds("beta",50,102)
power_bounds(1.2E+09,"vhigh_power")
power_bounds(300000000,"med_power")
power_bounds(10000000,"low_power")
similar_bounds(10000000,"high_similarity")
similar_bounds(10000,"low_similarity")
```

NS

APPENDIX O

This appendix contains the full evaluation data set. The data illustrated is that of 16-channel 8-second bipolar EEG data. This data is divided into 2-second analysis segments that are classified by the EEG expert and the IOARS into one of the following four categories:

. OA present and no abnormal slow waves.
. Abnormal slow waves present and no OA.
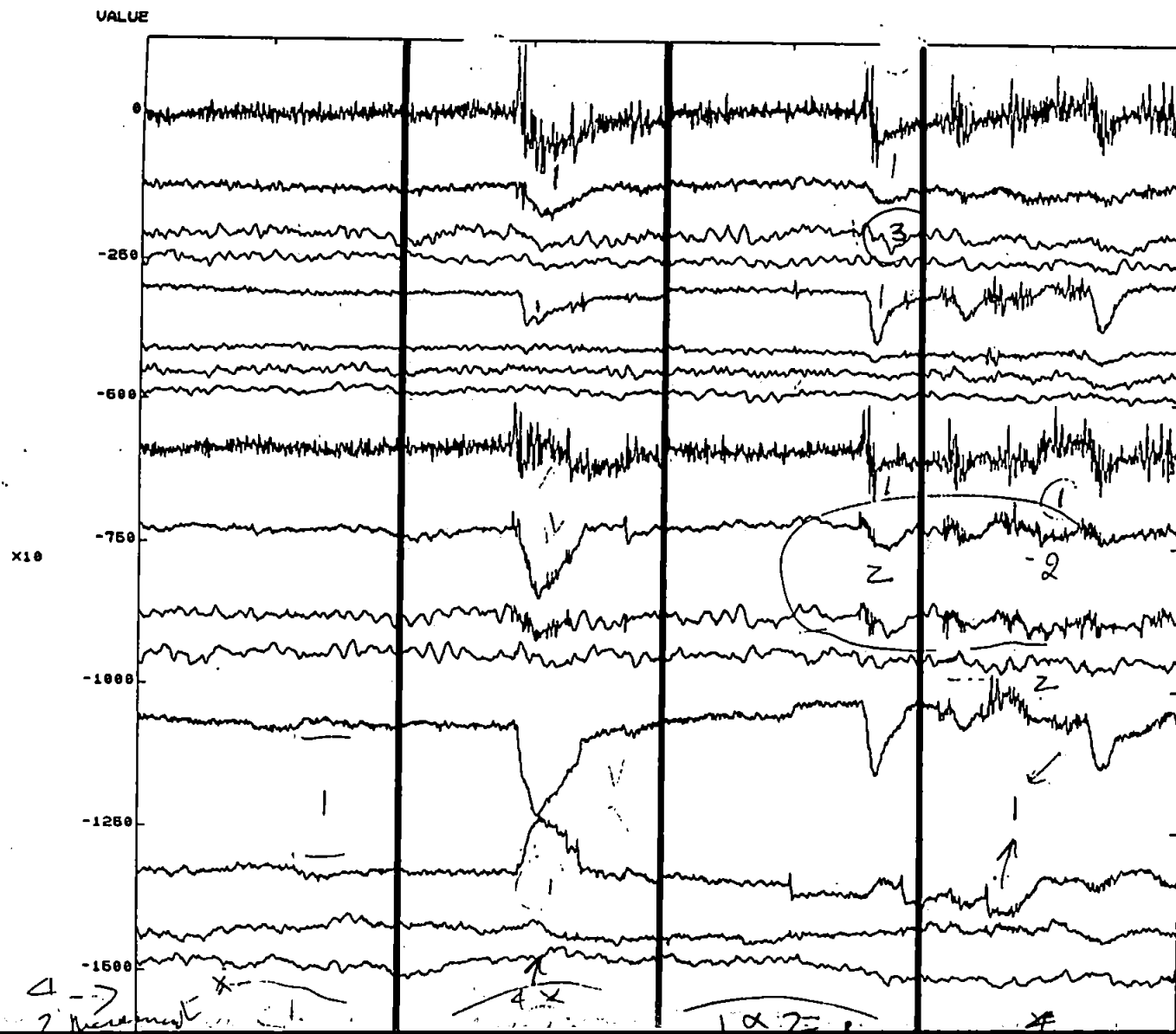. Both OA and abnormal slow waves present.
. normal.

The data set is included in this appendix to allow analysis of the data on which the IOARS has been evaluated as this will effect the significance of the results obtained.
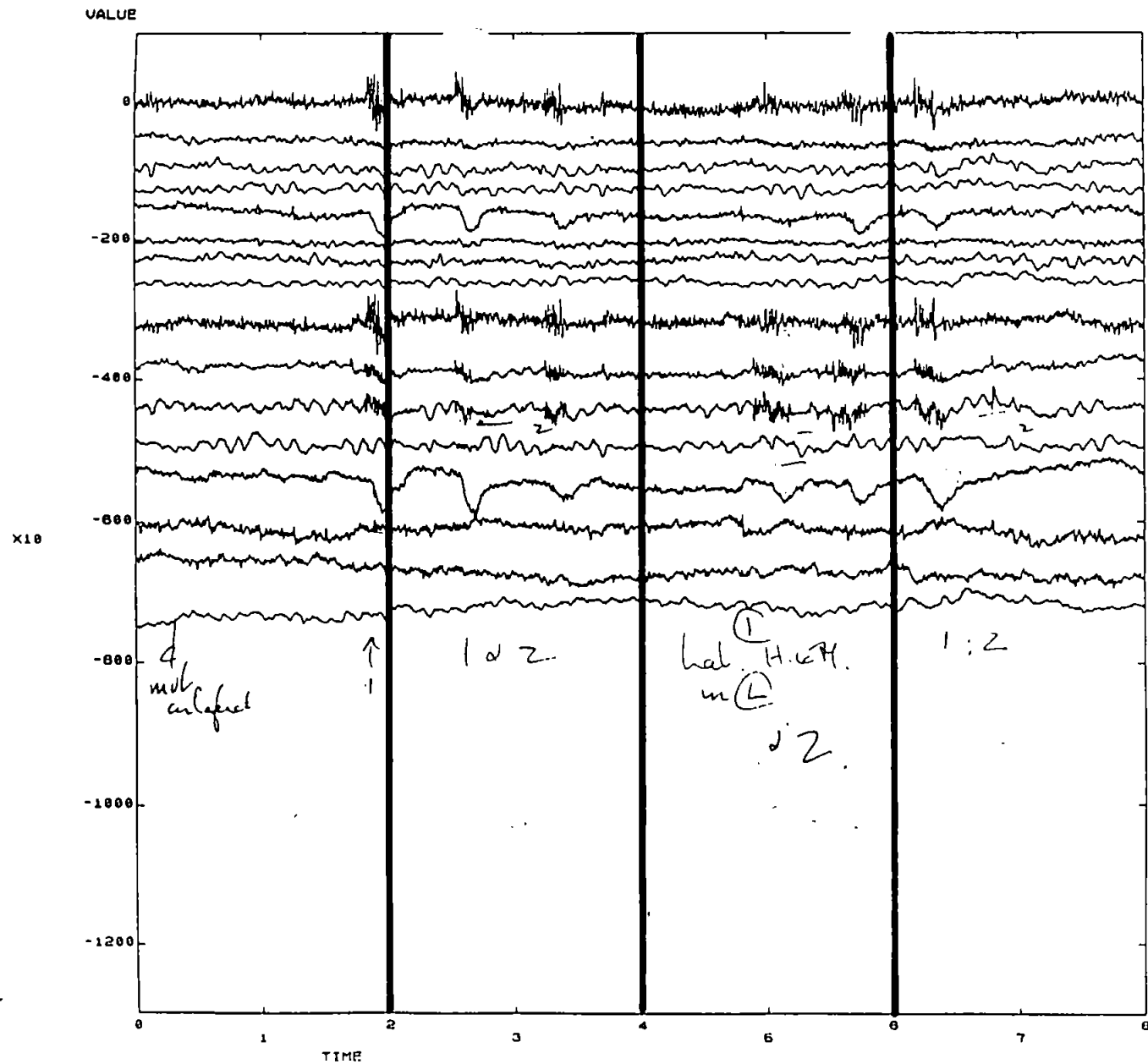
VALUE

4 CLASSES

1. EM ONLY
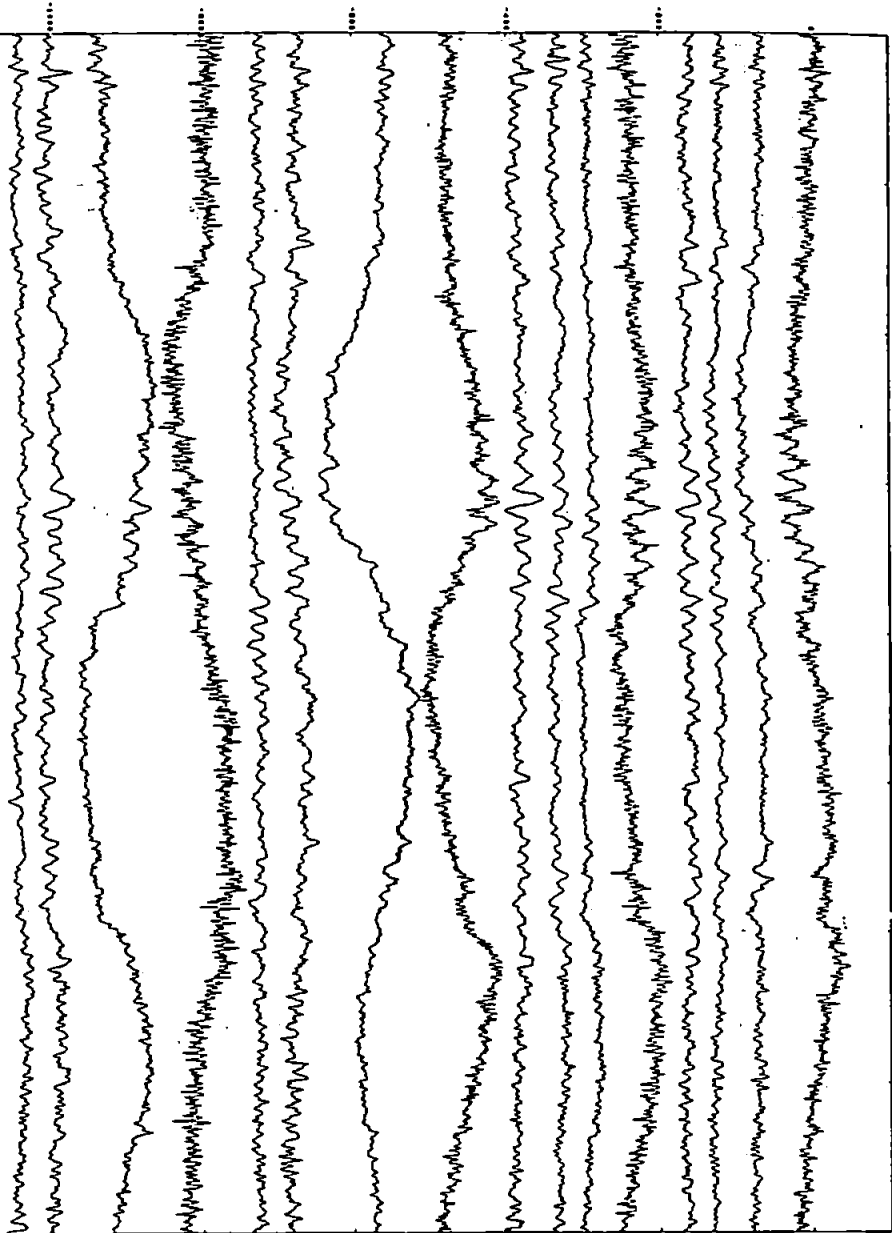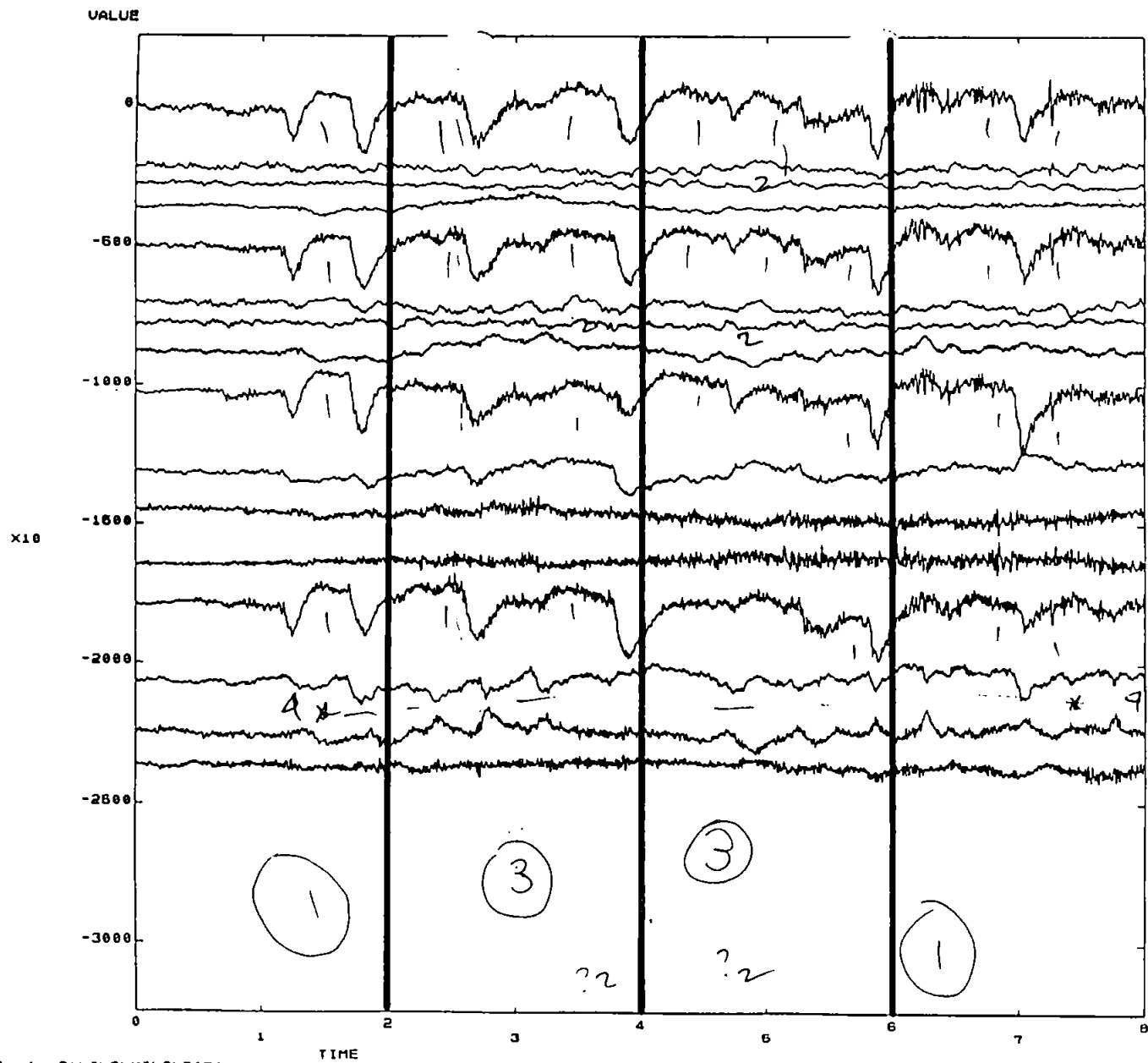2. CEREBRAL ONLY
3. BOTH
4. NEITHER

0

-200

X10

-400

-600

MONTAGE 1

-800

4
not
analysed

1 & 2

hal ① H.cM.
in ⓛ
& 2.

1:2

-1000

-1200

0   1   2   3   4   5   6   7   8
              TIME

VALUE

0

-500

MONTAGE 2

-1000

-1500

X10

-2000

4

electrode artefact

(P$_8$)

-2500

-3000

90

VALUE

X10

07

MONTAGE 1

← 4 → | ← 4 →

MONTAGE 1

→ most commonly is electrode artefact

? same 2 ~   1 & 4   ? same 2

↓ delta range.

but another sample cannot make a decision — need a few more w/ no artefact (more info)

0   1   2   3   4   5   6   7   8

TIME

0
-500
-1000
-1500
-2000
-2500
-3000

MONTAGE I

VALUE

x10

-1200
-1000
-800
-600
-400
-200
0

0 1 2 3 4 5 6 7

475

MONAGE 1

x10

010

VALUE

VALUE

012

MONTAGE 2

0

-200

X10

-400

-600

-800

-1000

-1200

1

1 2 4

4 ? electrode
- artefact

1

1

VALUE

0

-500

X10

-1000

-1500

-2000

-2500

-3000

MONTAGE 1

TIME

0   1   2   3   4   5   6   7   8

VALUE

X10

0

-500

-1000

-1500

-2000

-2500

-3000

MONTAGE 2

Muscle.

2   2   2   2

X10

MONTAGE 1

0
-250
-500
-750
-1000
-1250
-1500

0   1   2   3   4   5   6   7   8

2   2   2   2

VALUE

0

-500

-1000

-1500

-2000

-2500

-3000

X10

EEG

VOLTAGE 2

4 electrode artifact
(Eq)

(1) 1 & 2 4

(5) 1 & 2 4

(3) 1 & 2 4

(3) 1 & 2 4-

VALUE

X10

0
-200
-400
-600
-800
-1000
-1200

020

MONTAGE 2

2

2

2.
4? movement
artefact
- or electrode

2

4

4

VALUE

0
-200
-400
-600
-800
-1000
-1200

X10

montage 13

VALUE

X10

MONTAGE 1

3

1 anteriorly
) same 2

same 3   1 & 2   1 & 2.
→ posteriorly

⇐ Probably all 3 anteriorly →
but same 2 posteriorly

0   -200   -400   -600   -800   -1000   -1200

0   1   2   3   4   5   6   7   8

TIME

MONTAGE 1

2.   1   22.   2 1   1 0 2

MONTAGE 1

028

030

MONTAGE 2

2 α 4   2 ∠ 4   2 ∠ 7   2 ∠ 4   4 Electrode artefact

MONTAGE 1

electrode artefact

TIME
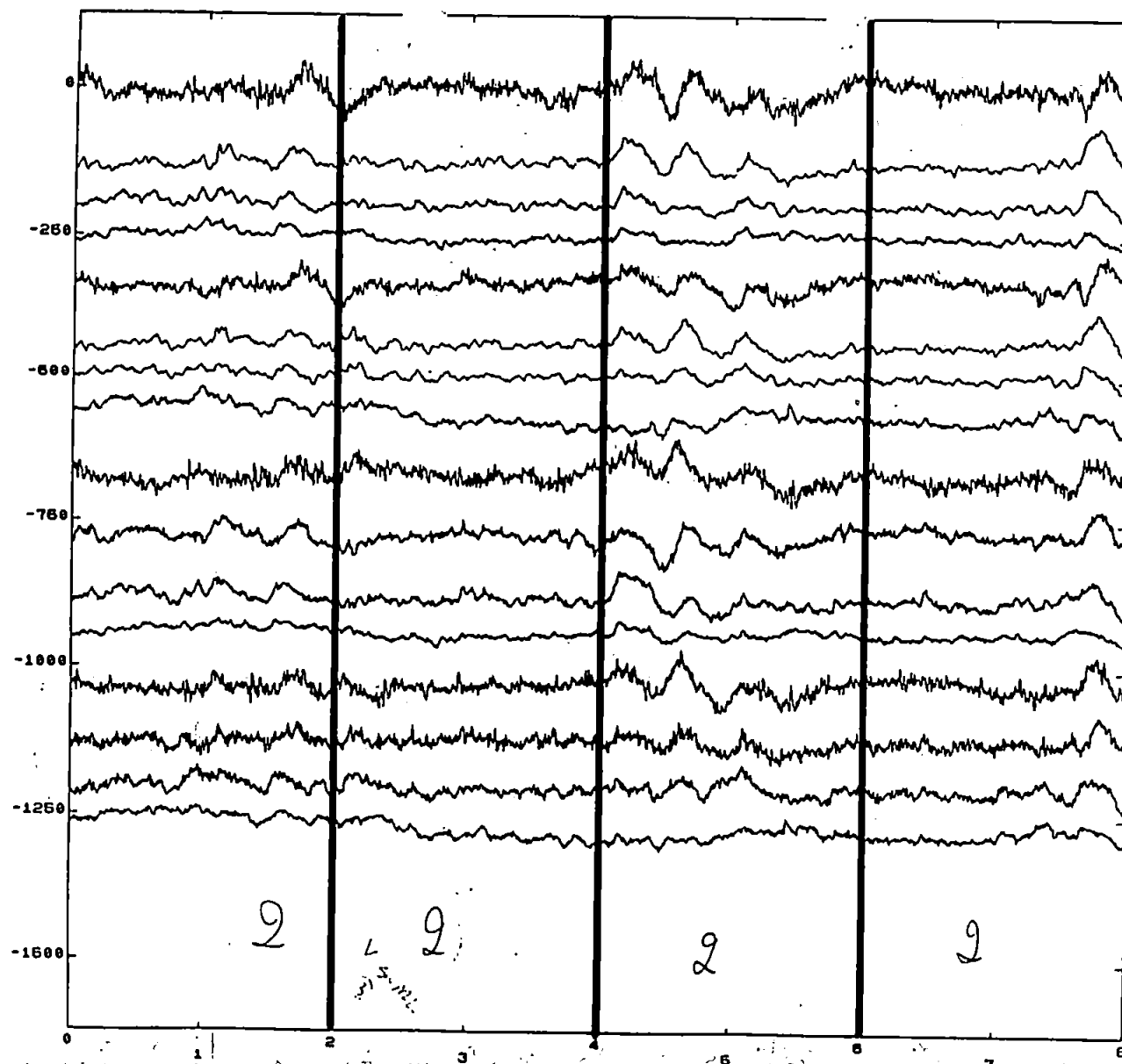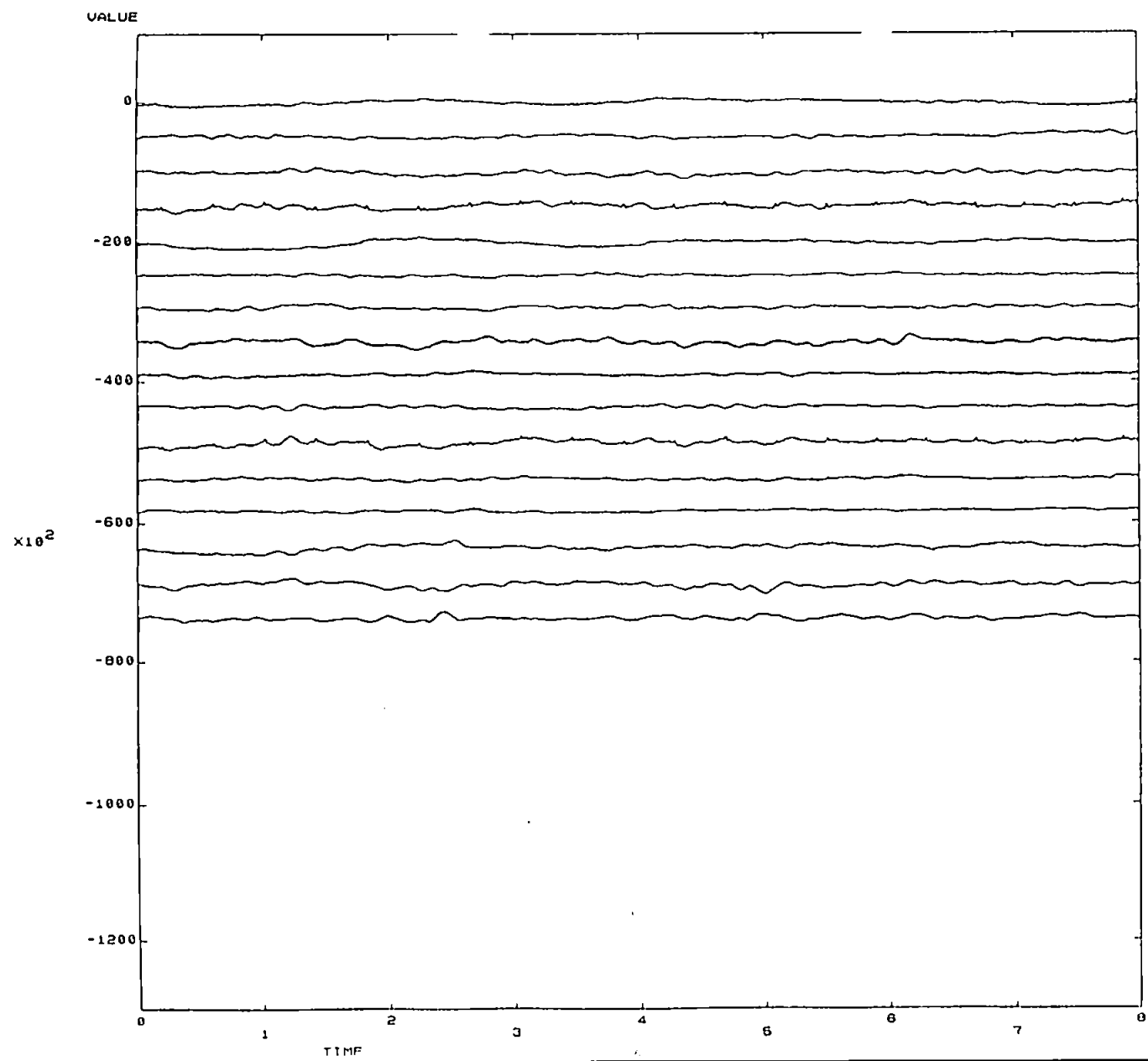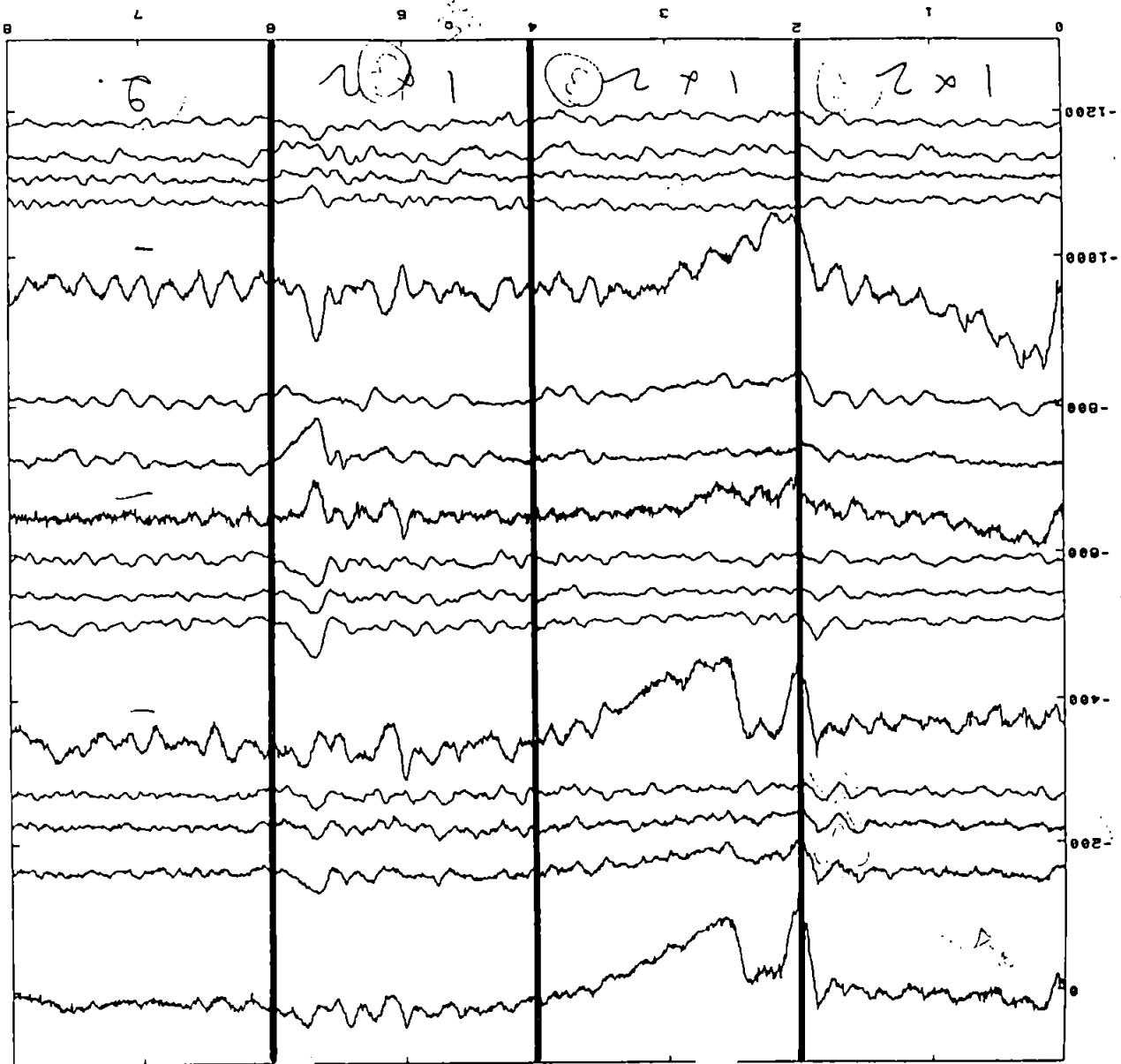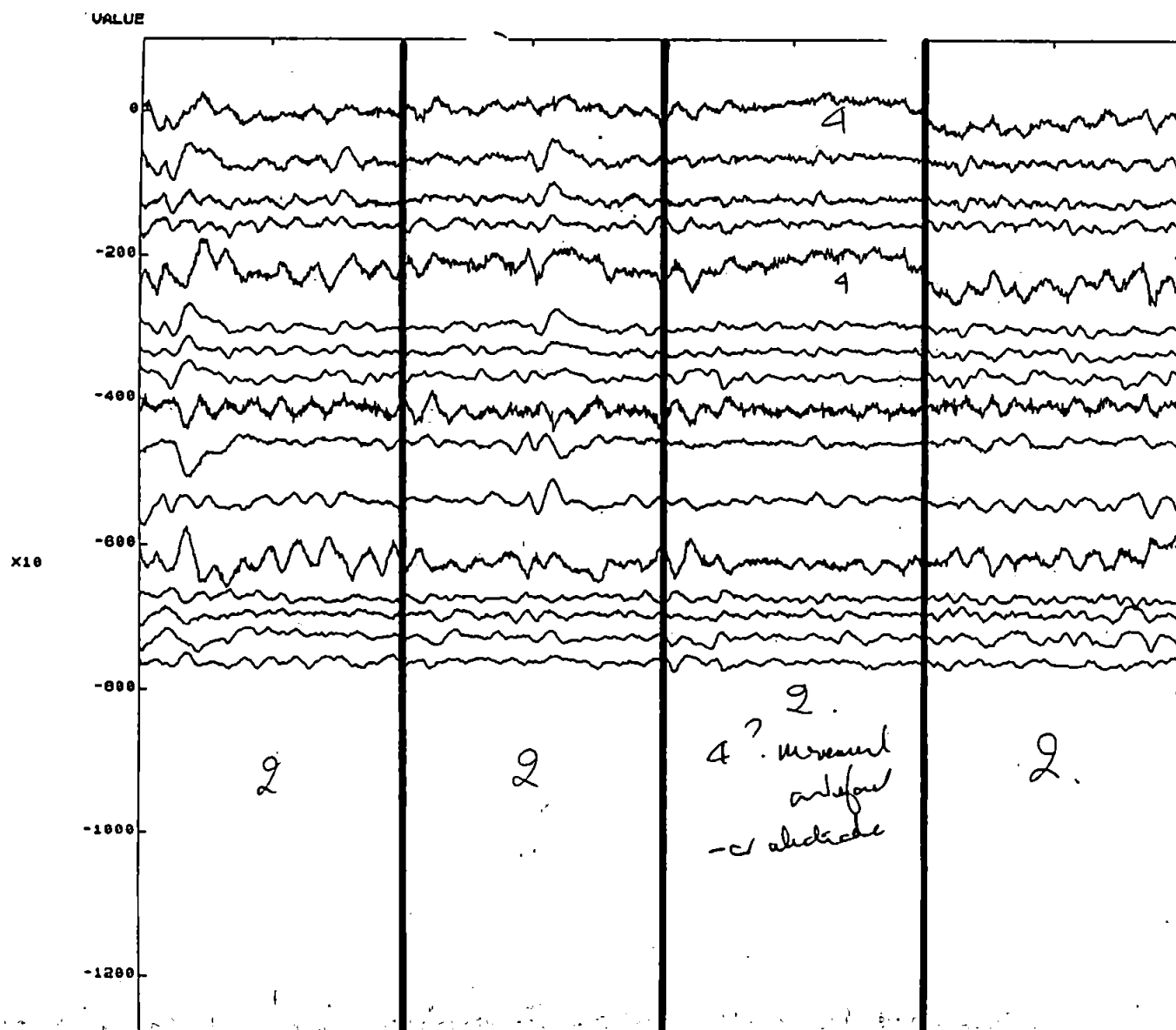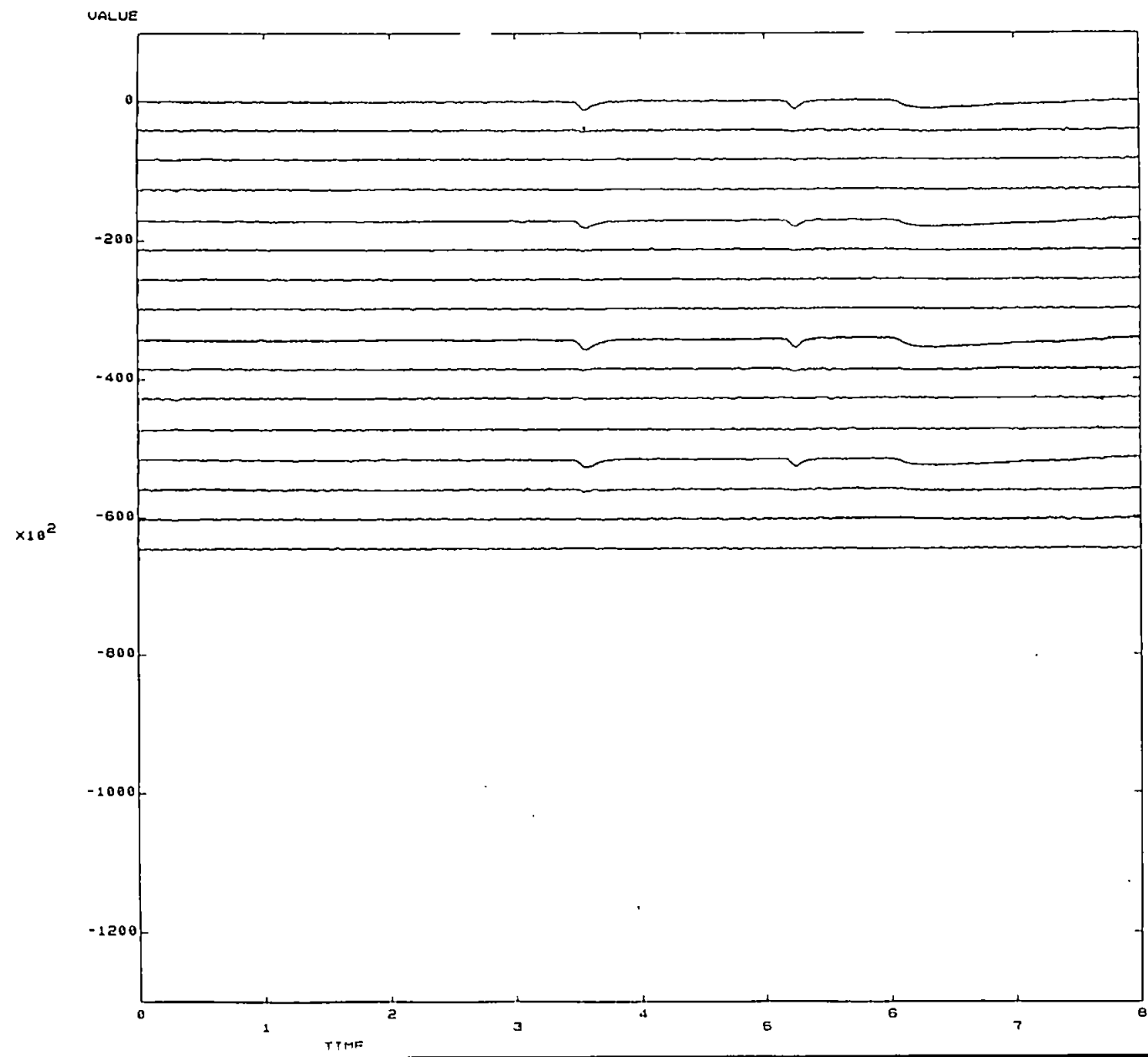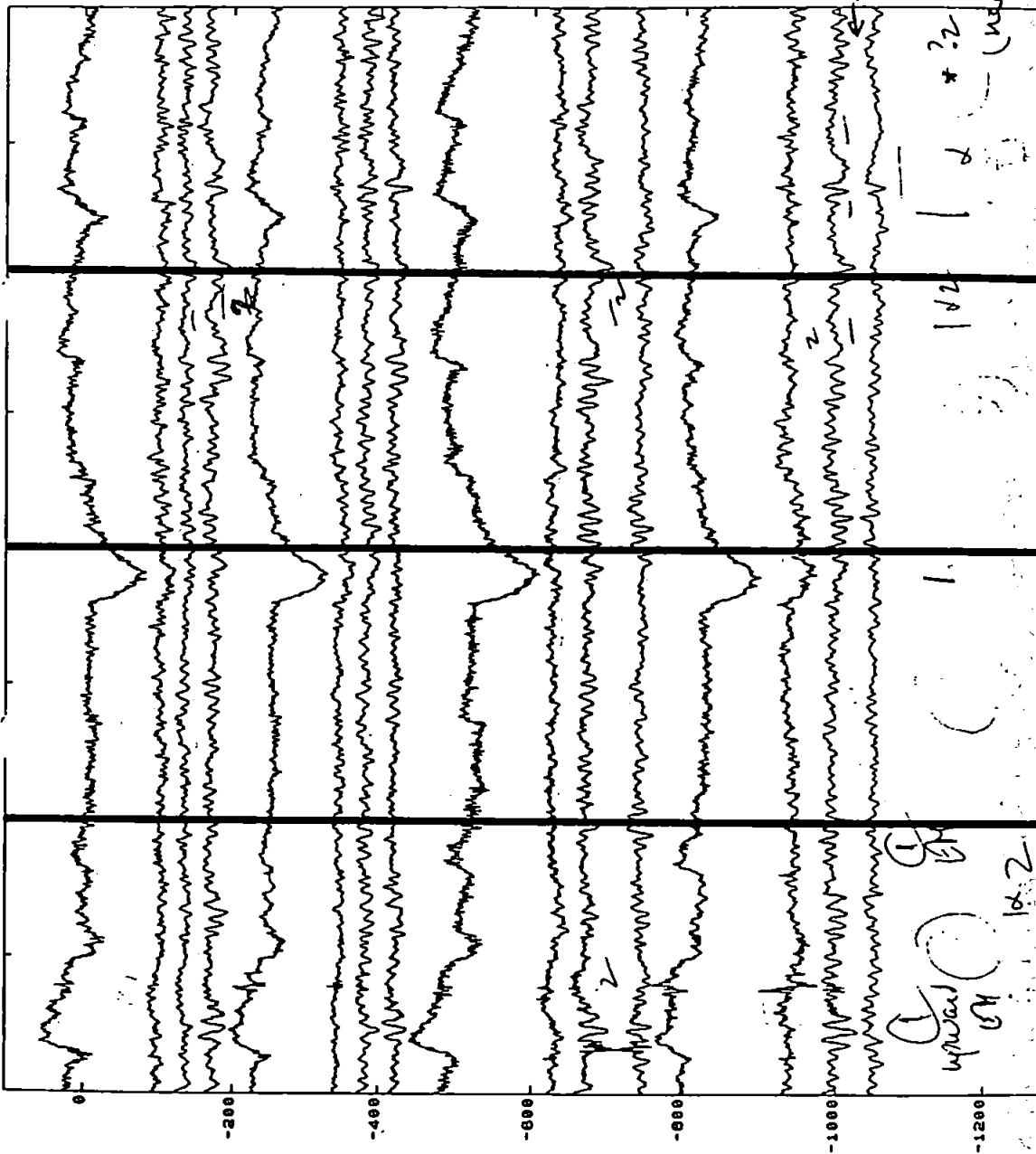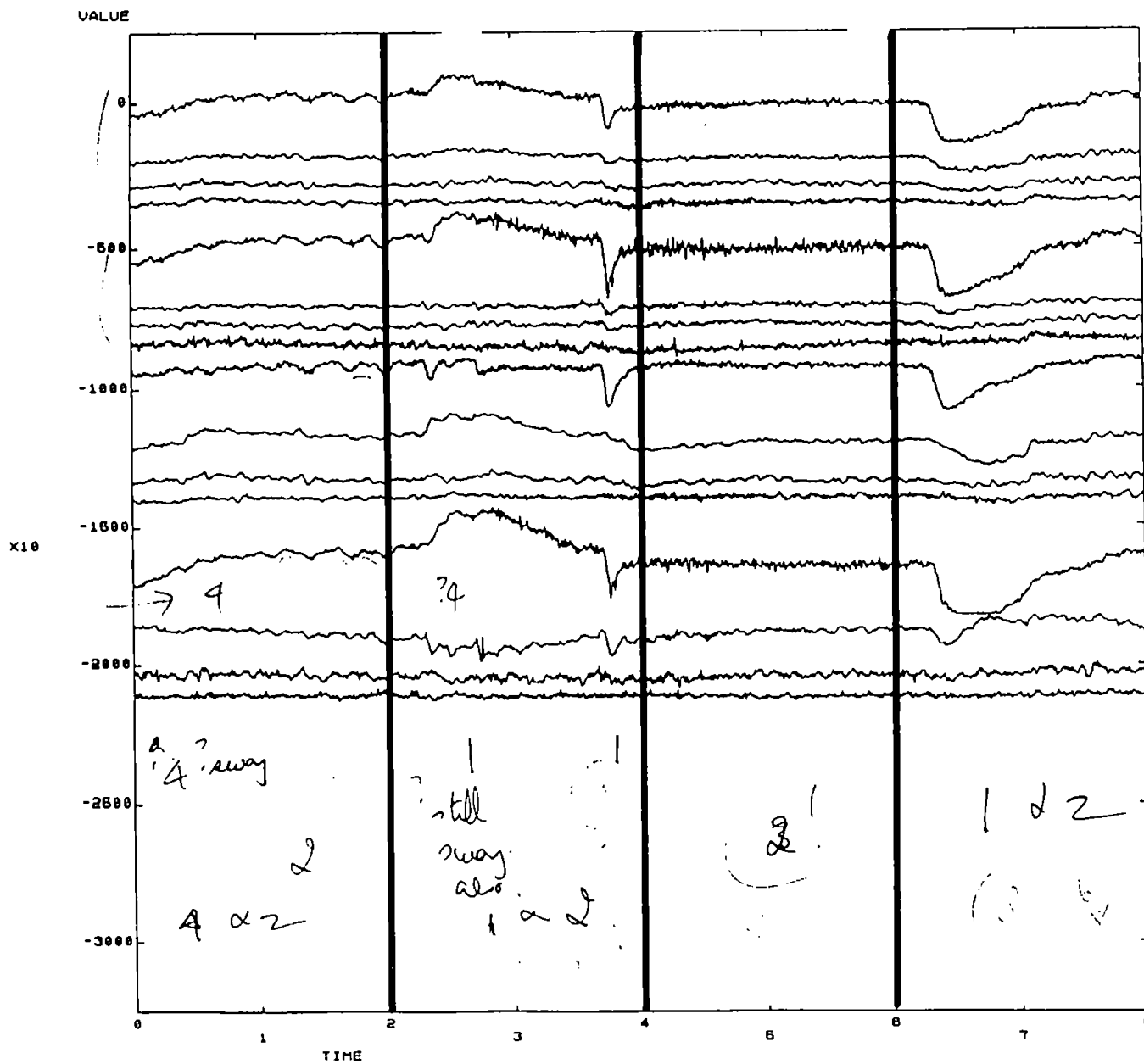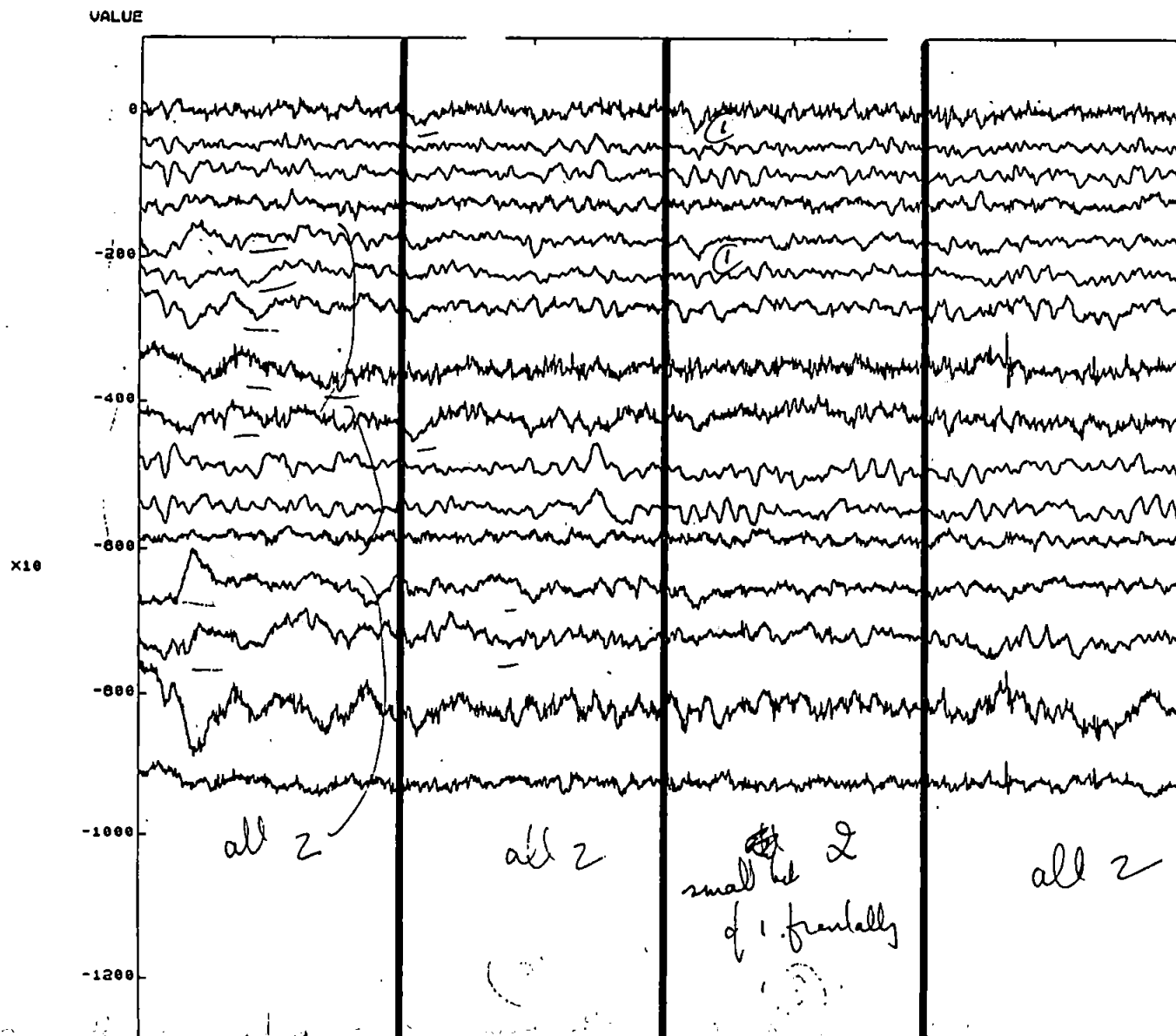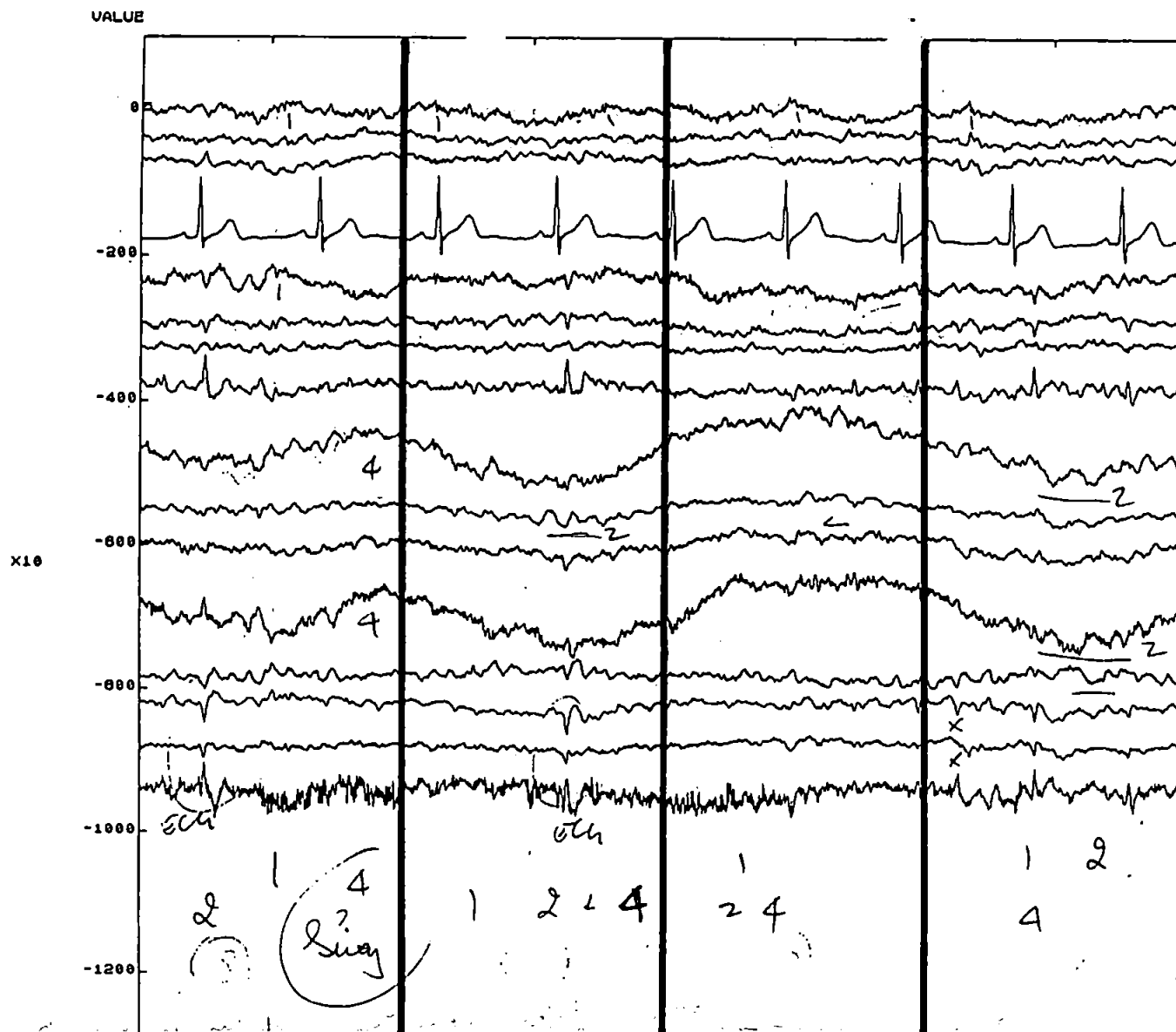
APPENDIX P

This appendix contains the full results of the comparison made, with a limited feature set, between the IOARS and expert in the classification of the the data segments given in appendix O.

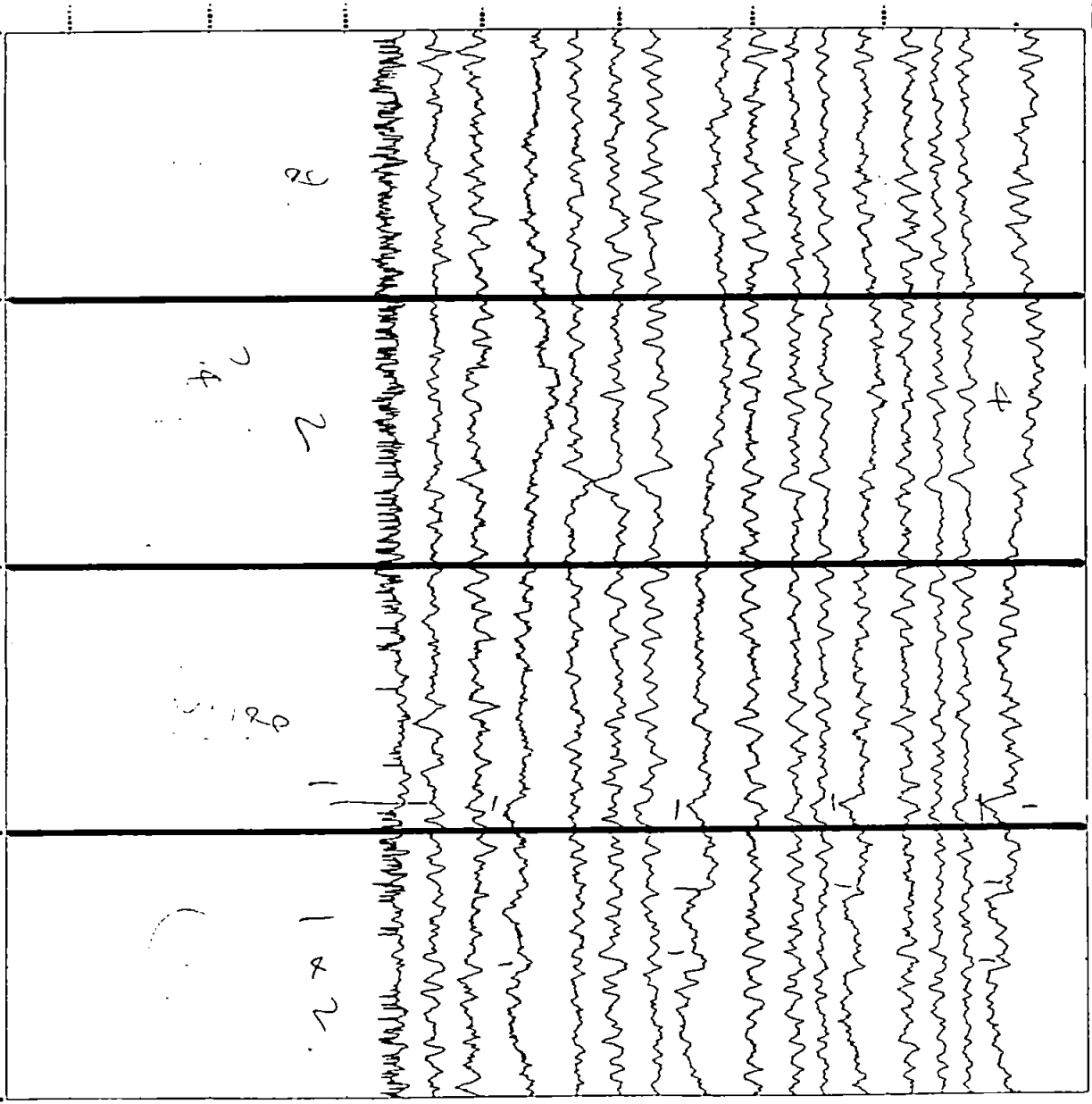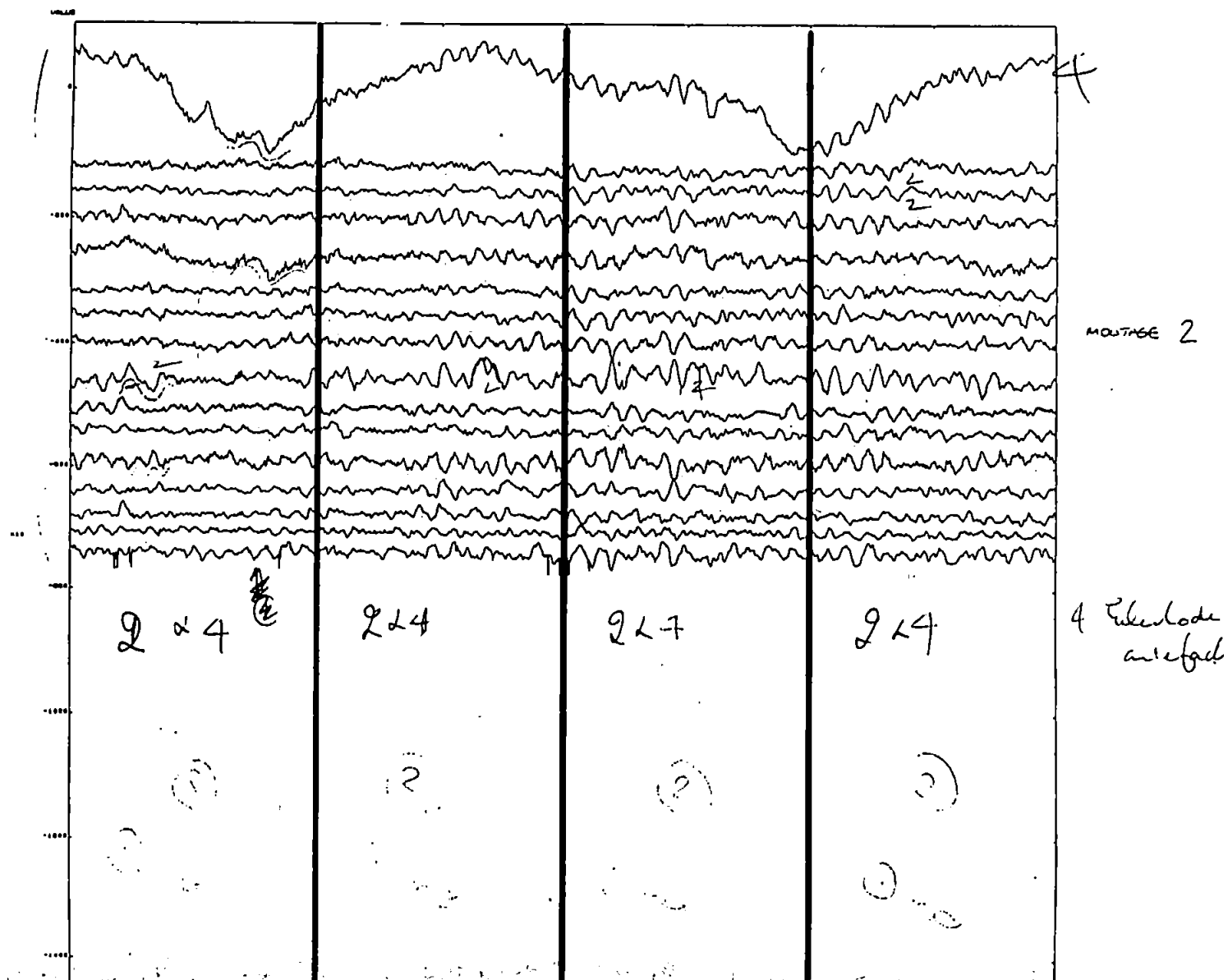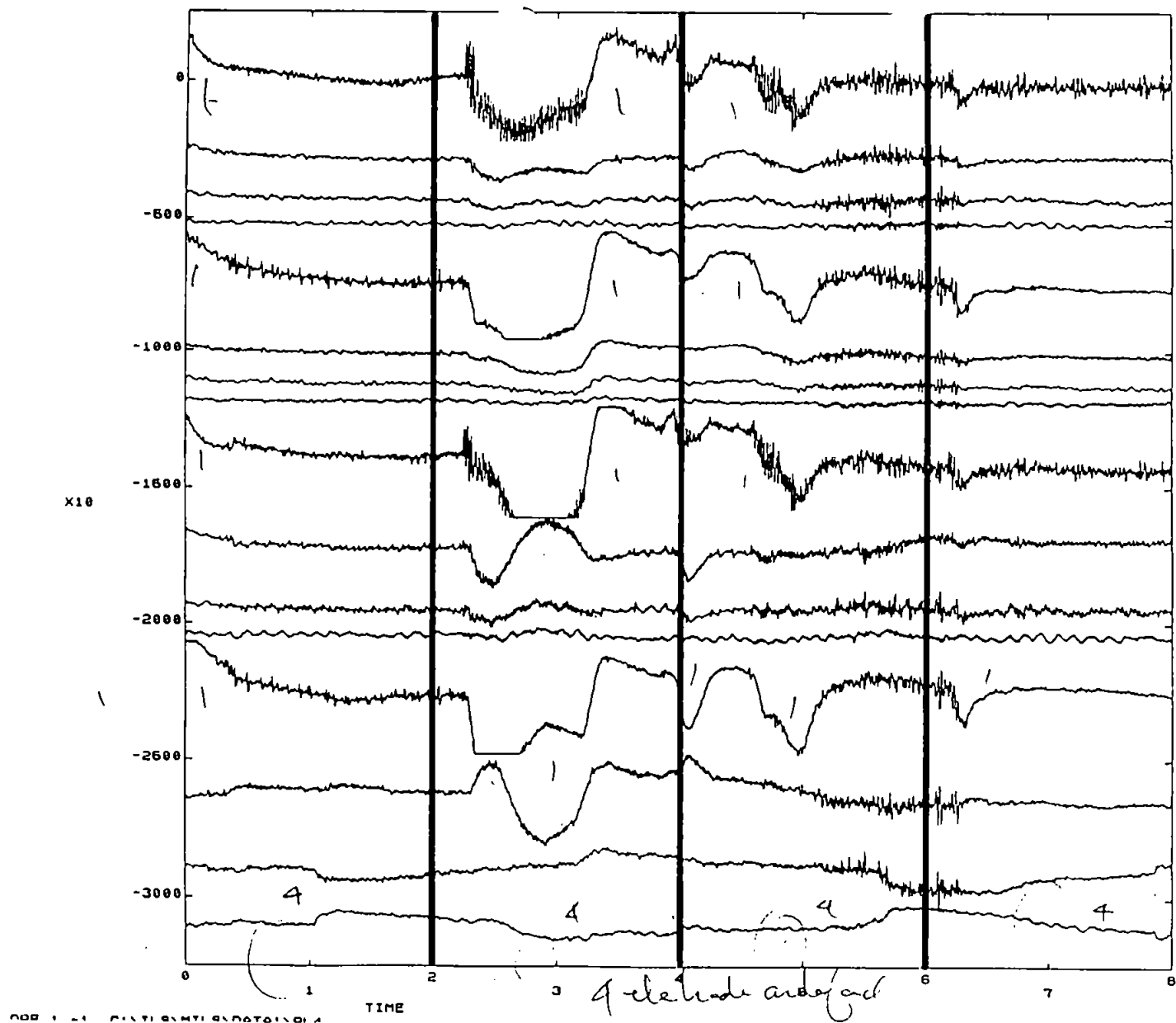| data	seg	expert	ICARS	MB
1	1	1	4	
1	2	3	3	
1	3	3	3	
1	4	3	3	
2	1	4	4	
2	2	1	1	
2	3	3	2	
2	4	3	3	
3	1	4	4	
3	2	4	4	
3	3	4	4	
3	4	4	4	
4	1	4	4	
4	2	3	2	
4	3	1	1	
4	4	1	1	
5	1	1	1	0.96
5	2	3	3	0.22
5	3	3	3	0.22
5	4	1	3	0.22
6	1	1	1	0.94
6	2	1	3	
6	3	1	3	
6	4	1	1	0.82
7	1	1	3	0.2
7	2	3	3	0.2
7	3	1	1	
7	4	3	3	0.13
8	1	3	3	0.43
8	2	1	1	0.82
8	3	3	3	0.7
8	4	3	3	0.5
9	1	1	1	0.47
9	2	1	2	0.8
9	3	1	1	0.94
9	4	1	1	0.96
10	1	1	1	0.93
10	2	1	1	1.0
10	3	1	1	0.87
10	4	1	1	9.98
11	1	1	1	0.94
11	2	1	1	0.98
11	3	1	1	0.92
11	4	1	1	0.96
12	1	1	1	1.0
12	2	1	1	0.94
12	3	1	1	0.96
12	4	1	1	0.93
13	1	2	2	0.75
13	2	2	2	1.0
13	3	3	3	0.22
13	4	3	3	0.53
14	1	2	3	0.22
14	2	2	2	0.8
14	3	2	3	0.43
14	4	2	3	0.2
15	1	2	1	
15	2	2	3	0.5
15	3	2	2	0.8
15	4	2	3	0.5
16	1	2	3	0.2
16	2	2	3	0.5
16	3	2	3	0.22
16	4	2	3	0.9
17	1			*quality*
17	2			*poor data*
17	3			
17	4			
18	1	2	3	0.22
18	2	2	2	0.8
18	3	2	2	0.8
18	4	2	2	0.8
19	1	3	3	0.22
19	2	3	3	0.4
19	3	3	2	0.8
19	4	2	2	0.8
20	1	2	2	0.7
20	2	2	2	0.7
20	3	2	2	0.7
20	4	2	2	0.7
21	1			*quality poor data*
21	2			
21	3			
21	4			
22	1	3	3	0.17
22	2	1	1	0.6
22	3	3	3	0.5
22	4	3	3	0.53
23	1	2	3	0.38
23	2	3	3	0.17
23	3	4		*data missing*
23	4	3	3	0.17
24	1	2	2	0.8
24	2	2	2	0.8
24	3	3	3	0.35

data	seq	expert	CARS	MB	data	seq	expert	CARS	MB
24	4	2	2	0.8	29	3	3	3	0.4
25	1	3	3	0.17	29	4	3	3	0.67
25	2	3	3	0.35	30	1			
25	3	3	3	0.45	30	2			
25	4	3	3	0.7	30	3			
26	1	—	—		30	4			
26	2	DATA	EEG CORRUPTED						
26	3	—	—						
26	4	—	—						
27	1	2	4						
27	2	3	3	0.67					
27	3	3	3	0.45					
27	4	3	3	0.35					
28	1	3	2	0.8					
28	2	3	3	0.7					
28	3	3	2	0.8					
28	4	3	3	0.22					
29	1	2	3	0.5					
29	2	2	3	0.5					

APPENDIX Q

This appendix contains the full results of the comparison made, with a full feature set, between the IOARS and expert in the classification of the the data segments given in appendix O.

data	seg	expert	iOAR	MB	data	seg	expert	iOAR	MB	data	seg	expert	iOAR	MB	data	seg	expert	iOAR	MB	data	seg	expert	iOAR	MB
1	1	1	4		5	4	1	3	0.77	10	3	1	1	0.87	15	2	2	3	0.81	20	1	2	2	0.2
1	2	3	3	0.88	6	1	1	1	0.85	10	4	1	1	0.18	15	3	2	4	—	20	2	2	2	0.7
1	3	3	3	0.95	6	2	1	3	0.24	11	1	1	1	0.99	15	4	2	2	0.61	20	3	2	2	0.7
1	4	3	3	0.95	6	3	1	3	0.61	11	2	1	1	0.48	16	1	2	2	0.57	20	4	2	2	0.7
2	1	4	2	0.8	6	4	1	1	0.69	11	3	1	1	0.92	16	2	2	2	0.6	21	1			
2	2	1	2	0.8	7	1	1	3	0.2	11	4	1	1	0.96	16	3	2	2	0.81	21	2			
2	3	3	3	0.97	7	2	3	3	0.2	12	1	1	1	1.0	16	4	2	2	0.84	21	3			
2	4	3	3	0.87	7	3	1	1		12	2	1	1	0.99	17	1		—		21	4			
3	1	4	4	1.0	7	4	3	3	0.13	12	3	1	1	0.96	17	2		—		22	1	3	3	0.17
3	2	4	4	0.98	8	1	3	3	0.43	12	4	1	1	0.83	17	3		—		22	2	1	1	0.96
3	3	4	4	0.98	8	2	1	1	0.82	13	1	2	2	0.75	17	4		—		22	3	3	3	0.5
3	4	4	4	0.98	8	3	3	3	0.7	13	2	2	2	1.0	18	1	2	3	0.69	22	4	3	3	0.53
4	1	4	4	0.98	8	4	3	3	0.5	13	3	3	3	0.22	18	2	2	2	0.7	23	1	2	3	0.38
4	2	3	4	0.98	9	1	1	1	0.97	13	4	3	3	0.53	18	3	2	2	0.85	23	2	3	3	0.17
4	3	1	4	0.98	9	2	1	1	0.8	14	1	2	2	0.53	18	4	2	2	0.87	23	3	4		—
4	4	1	4	0.98	9	3	1	1	0.24	14	2	2	2	0.78	19	1	3	3	0.22	23	4	3	3	0.17
5	1	1	1	0.37	9	4	1	1	0.95	14	3	2	2	0.85	19	2	3	3	0.4	24	1	2	2	0.8
5	2	3	3	0.32	10	1	1	1	0.83	14	4	2	2	0.87	19	3	3	2	0.8	24	2	2	2	0.8
5	3	3	3	0.32	10	2	1	1	1.0	15	1	2	1	0.95	19	4	2	2	0.8	24	3	3	3	0.35

data	seg	expert	OARS	MB	data	seg	expert	OARS	MB	data	seg	expert	OARS	MB	data	seg	expert	OARS	MB	data	seg	expert	OARS	MB	data	seg	expert	OARS	MB
24	4	2	2	0.8	29	3	3	3	0.4																				
25	1	3	3	0.7	29	4	3	3	0.67																				
25	2	3	3	0.35	30	1																							
25	3	3	3	0.45	30	2																							
25	4	3	3	0.7	30	3																							
26	1	—	—		30	4																							
26	2	—																											
26	3	—																											
26	4	—																											
27	1	2	4																										
27	2	3	3	0.67																									
27	3	3	3	0.45																									
27	4	3	3	0.35																									
28	1	3	2	0.8																									
28	2	3	3	0.7																									
28	3	3	2	0.8																									
28	4	3	3	0.22																									
29	1	2	3	0.5																									
29	2	2	3	0.5																									

EVALUATION II

APPENDIX R

This appendix contains the full pre-clinical evaluation data set. The data illustrated is that of 16-channel 8-second bipolar EEG data. This data is divided into 2-second analysis segments that are classified by the EEG expert and the IOARS into one of the following four categories:

. OA present and no abnormal slow waves.
. Abnormal slow waves present and no OA.
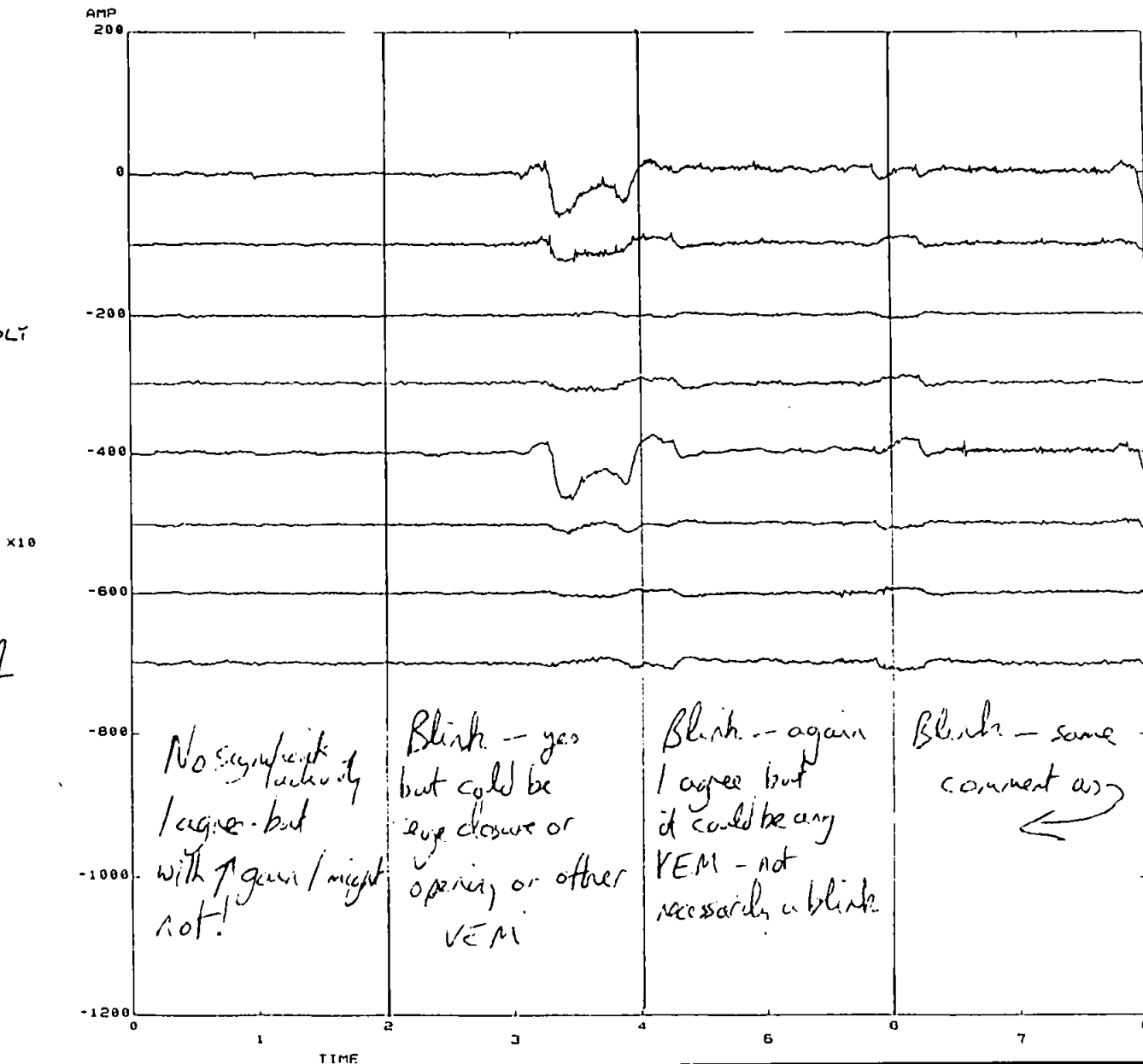. Both OA and abnormal slow waves present.
. normal.

The data set is included in this appendix to allow analysis of the data on which the IOARS has been evaluated as this will effect the significance of the results obtained.

AMP

200

0

O — 5HZ

0

-200

OA only

Abms Δ only

Both

together

-400

×10

-600

DATA 1

-800

No significant activity
I agree - but
with ↑ gain I might
not!

Blink — yes
but could be
eye closure or
opening or other
VEM

Blink — again
I agree but
it could be any
VEM - not
necessarily a blink

Blink — same
comment as →

-1000

-1200

0   1   2   3   4   6   6   7   8

TIME

R1

DATA 2

X10

AMP
200

0

-200

-400

-600

-800

-1000

Weak E/M	Weak E/M	No significant activity	Weak E/M
I don't agree - these are certainly blinks	— same comment ... as)	— agreed except for a bit of E/M at the beginning of segment.	no — blinks again.

R3

DATA 3 ×10

AMP

200

0

-200

-400

-600

-800

-1000

-1200

Blinks — certainly VEMS but difficult to say whether there are S/W as well — see channels 3 and 4

Blinks — I would probably say both VEM (not blinks) and S/W are possible here — difficult but you are right to be conservative.

Weak E/M — No not weak it is fairly strong — VEM & possibly S/W as well. Don't understand why it says weak

Weak E/M — same comment as)

0   1   2   3   4   6   6   7   8
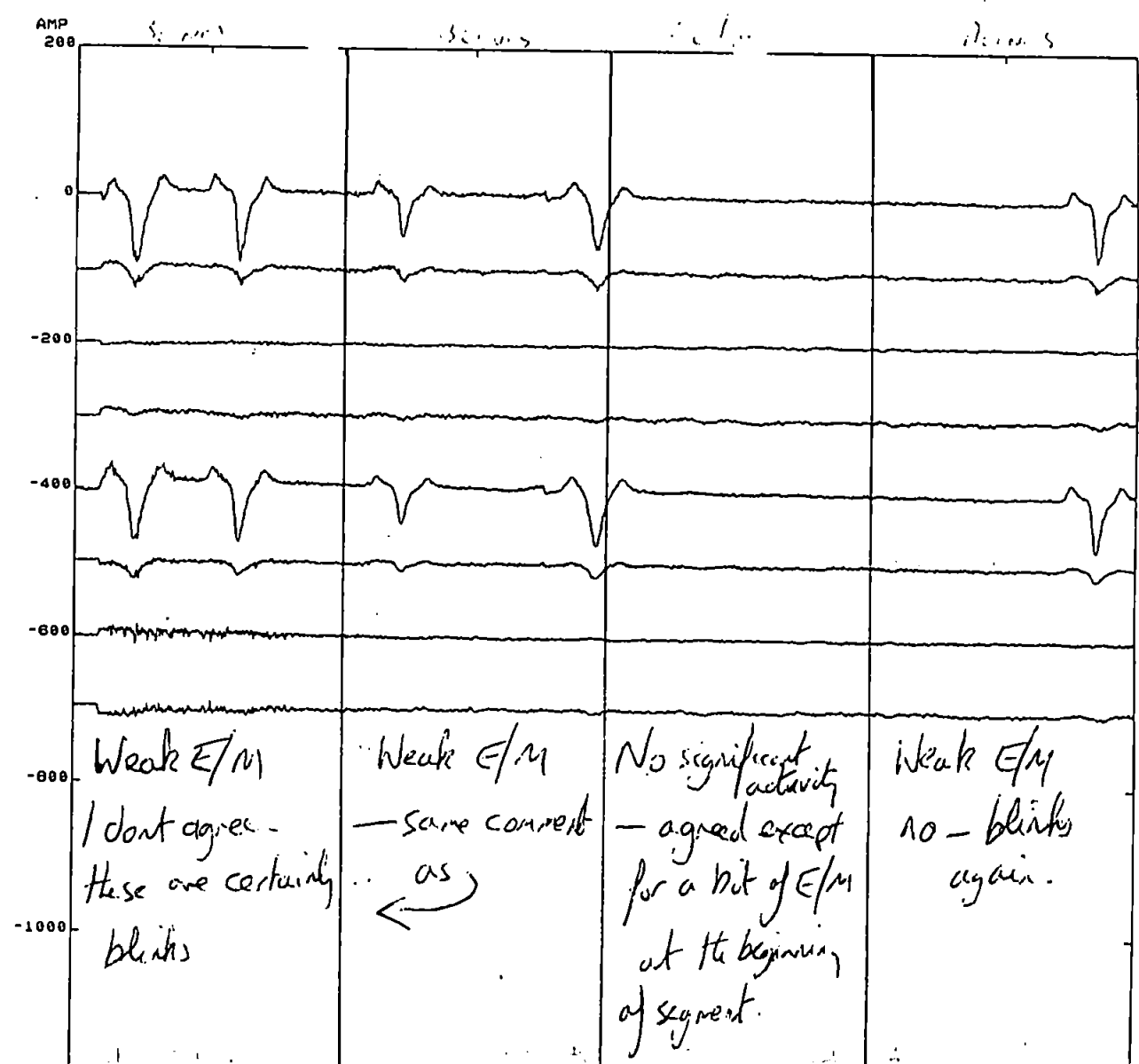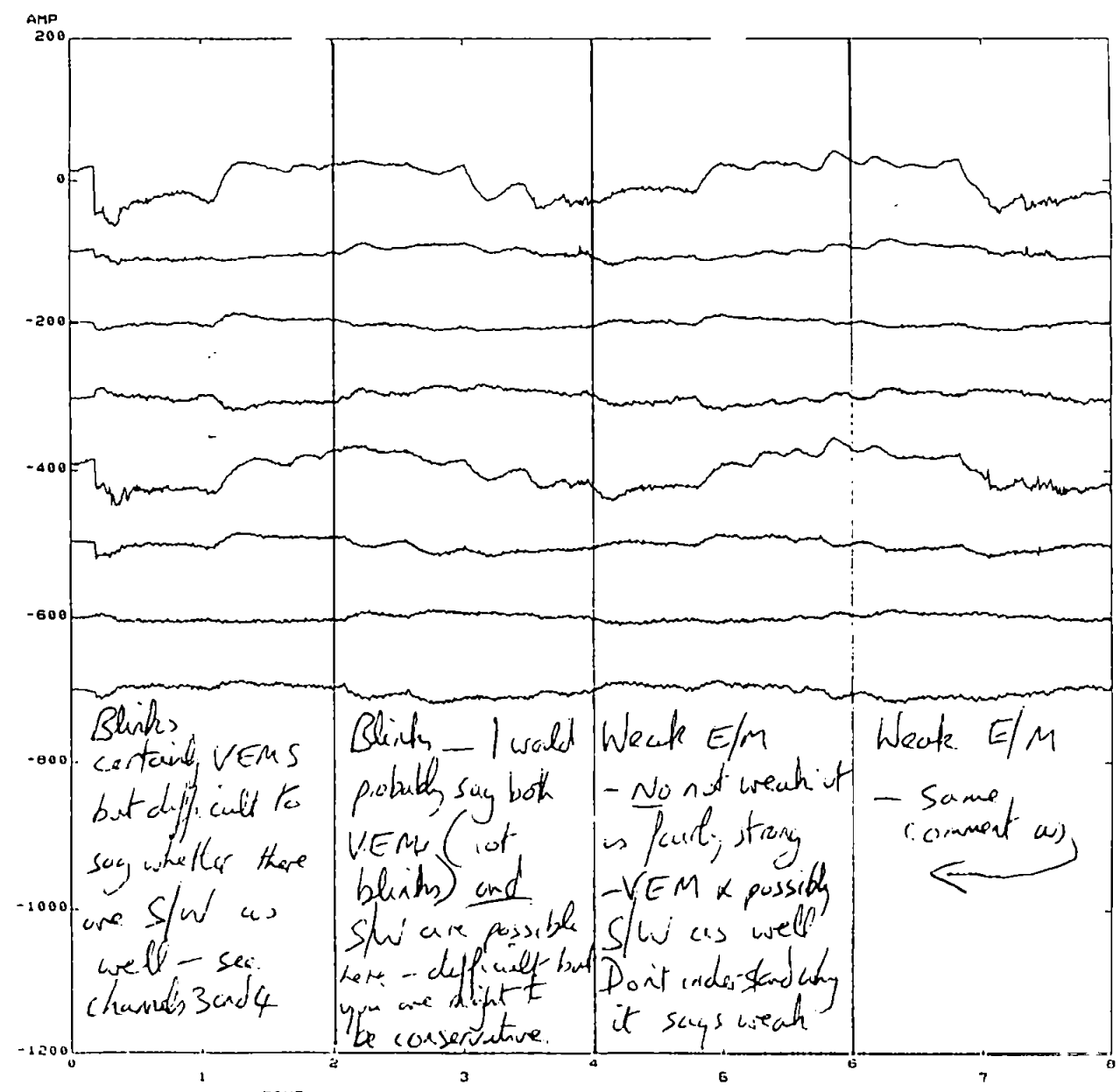
DATA 4

X10
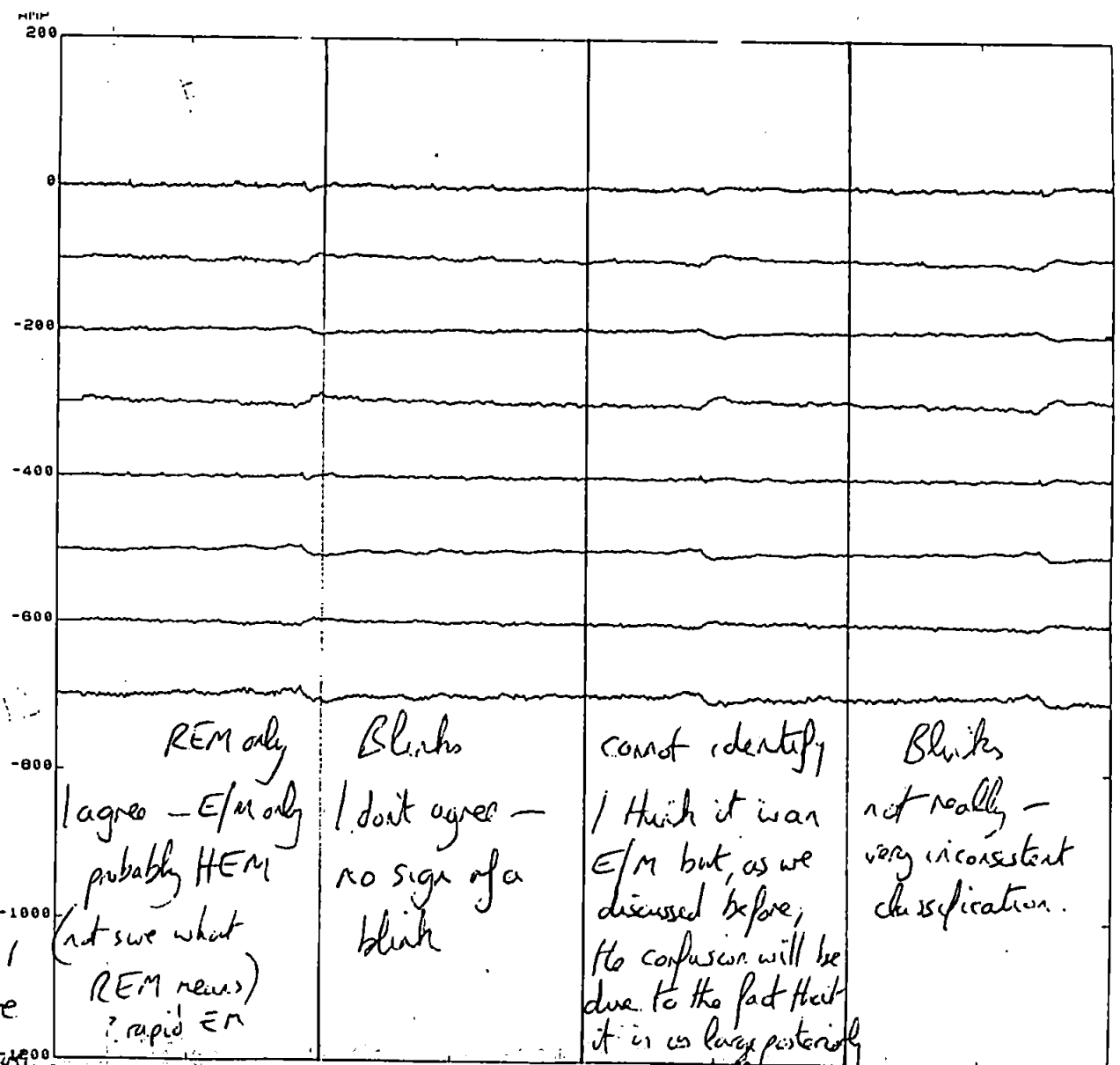
This has led to
what seem to be
inconsistencies in
the system. Seg1
Seg3 and Seg4 are
very similar but have...

REM only	Blinks	cannot identify	Blinks
I agree — E/M only probably HEM (not sure what REM means) ? rapid EM	I don't agree — no sign of a blink	I think it was E/M but, as we discussed before, the confusion will be due to the fact that it is as large posterior...	not really — very inconsistent classification.

DATA 5

detectable OA
in abnormal S/w
I agree — not
quite sure of the
OA bit though
? chan 1 underlined
bit

No significant artifact
— I don't agree —
probably some E/M
in channels 1 and 5
underlined — seems
inconsistent with
seg 1

Abnormal S/w
I agree

Abnormal S/w
I agree

27

AMP
200

DATA S

0

-200

X10

-400

-600

-800

-1000

-1200

0   1   2   3   4   5   6   7   8

Weak E/M
— I agree that
E/M are present
but not weak —
quite strong.

Blinks
— certainly
E/M but probably
not blinks, just
VEM

Blinks
Same comment
as →

No Significant
activity
— equivocal — Here is
probably some weak
E/M but significant
in a subjective
term.

DATA 6

X10

AMP
200

0

-200

-400

-600

-800

-1000

mainly OA with little abnormal S/W
I don't agree — just as likely to be abnormal S/W in EM

abnormal S/W only — I don't agree — probably both — can't be sure — inconsistent classification

cannot identify
I agree — difficult to be sure. but probably S/W and E/M

Abnormal S/W only
I don't agree — this segment is like the previous one to me inconsistent too

28

APPENDIX S

This appendix contains the logged expert responses for the pre-clinical evaluation of the IOARS. The log presents the following information for each analysis segment:

. date
. time
. segment identification number
. list of extracted features:
    . contextual domain features
    . time domain features
    . frequency domain features
. list of successful conditions
. system classification
. classification measure of belief
. experts verifying judgement.

12-3-1991 13:57
DATA1 SEG1

List of extracted features
===

["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
not any significant spectral peaks exist 1

segment was classified as containing
no significant activity with a certainty of 1.

The users response was n

12-3-1991 17:19
DATA3 SEG1

List of extracted features
==

["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","high_speed"] 1
["f_feature","fp2-f8","pk2","delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk3","delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk4","delta","med_power","high_speed"] 0.75
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","low_power","high_speed"] 0.75
["f_feature","fp2-f4","pk1","sub_delta","vhigh_power","high_speed"] 1
["f_feature","fp2-f4","pk2","delta","med_power","med_speed"] 1
["f_feature","fp2-f4","pk3","delta","low_power","med_speed"] 1
["f_feature","fp1-f3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f3","pk2","delta","low_power","med_speed"] 1
["f_feature","f4-c4","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","p3-o1","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","f8-t4","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f8-t4","pk2","delta","med_power","med_speed"] 1
["f_feature","f8-t4","pk3","delta","med_power","med_speed"] 1
["f_feature","f7-t3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f7-t3","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","f7-t3","pk2","delta","med_power","low_speed"] 0.75
["f_feature","f7-t3","pk3","delta","low_power","med_speed"] 1
["f_feature","f7-t3","pk4","delta","low_power","high_speed"] 0.75
["f_feature","t5-o1","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","high_speed"] 1
["f_feature","fp2-f8","pk2","delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk3","delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk4","delta","low_power","high_speed"] 1
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","low_power","high_speed"] 0.75
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1

S1

not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 0.25
not fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
not fp2-f8 and fp1-f7 are symmetrical for all delta peaks 1
not delta peaks are present that are not present in the EOG 0.75
not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 0.0529157667
delta activity only appears in the anterior half of the scalp 0.75
not delta spectral peaks exist only in the EOG 0.75
the largest spectral peak appears in the EOG 1
the largest spectral peak is attributable to more than one electrode 0.25
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
not the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 1
the waveform in channels fp2-f8 and f8-t4 are in phase 1
the waveform in channels fp1-f7 and f7-t3 are in phase 1

segment was classified as containing
blink artefact with a certainty of 0.75025.

Th   rs response was y

12-3-1991 17:26
DATA4 SEG1

List of extracted features
====================================

["c_feature","class_totals"] 1
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","high_speed"] 1
["f_feature","fp2-f8","pk2","delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk3","delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk4","delta","med_power","high_speed"] 0.75
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","low_power","high_speed"] 0.75
["f_feature","fp2-f4","pk1","sub_delta","vhigh_power","high_speed"] 1
["f_feature","fp2-f4","pk2","delta","med_power","med_speed"] 1
["f_feature","fp2-f4","pk3","delta","low_power","med_speed"] 1
["f_feature","fp1-f3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f3","pk2","delta","low_power","med_speed"] 1
["f_feature","f4-c4","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","p3-o1","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","f8-t4","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f8-t4","pk2","delta","med_power","med_speed"] 1
["f_feature","f8-t4","pk3","delta","med_power","med_speed"] 1
["f_feature","f7-t3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f7-t3","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","f7-t3","pk2","delta","med_power","low_speed"] 0.75
["f_feature","f7-t3","pk3","delta","low_power","med_speed"] 1
["f_feature","f7-t3","pk4","delta","low_power","high_speed"] 0.75
["f_feature","t5-o1","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","high_speed"] 1
["f_feature","fp2-f8","pk2","delta","vhigh_power","med_speed"] 1

52

["f_feature","fp2-f3","pk3","delta","med_power","med_speed"] 1
["f_feature","fp2-f3","pk4","delta","low_power","high_speed"] 1
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","low_power","high_speed"] 0.75
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 0.25
not fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
not fp2-f3 and fp1-f7 are symmetrical for all delta peaks 1
not delta peaks are present that are not present in the EOG 0.75
not fp2-f3 or fp1-f7 contains isolated non repetitive waveforms 0.0529157667
delta activity only appears in the anterior half of the scalp 0.75
not delta spectral peaks exist only in the EOG 0.75
the largest spectral peak appears in the EOG 1
the largest spectral peak is attributable to more than one electrode 0.25
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
no   waveform in channels fp2-f4 or fp1-f3 is slow -low delta 1
the waveform in channels fp2-f3 and f3-t4 are in phase 0
the waveform in channels fp1-f7 and f7-t3 are in phase 0

segment was classified as containing
blink artefact with a certainty of 0.1675.

The users response was n

12-3-1991 17:30
DATA7 SEG4

List of extracted features
=================================

["c_feature","class_totals"] 1
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["f_feature","fp2-f3","pk1","delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f3","pk2","delta","vhigh_power","high_speed"] 1
["f_feature","fp1-f7","pk1","delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","med_power","high_speed"] 1
["f_feature","fp2-f4","pk1","delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f4","pk2","delta","med_power","high_speed"] 1
["f_feature","fp1-f3","pk1","delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f3","pk2","delta","med_power","high_speed"] 1
["f_feature","fp2-f3","pk1","delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f3","pk2","delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f3","pk3","delta","med_power","high_speed"] 1
["f_feature","fp1-f7","pk1","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","med_power","high_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1

not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 1
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
fp2-f3 and fp1-f7 are symmetrical for all delta peaks 1
delta activity only appears in frontal channels 1
not delta spectral peaks exist only in the EOG 1
the largest spectral peak is attributable to more than one electrode 1
the largest spectral peak appears in the EOG 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
not the waveform in channels fp2-f4 or fp1-f3 is slow-low delta 1
the waveform in channels fp2-f3 and f3-t4 are in phase 1
the waveform in channels fp1-f7 and f7-t3 are in phase 1

segment was classified as containing
blink artefact with a certainty of 0.9958.

The users response was y

12-3-1991 17:33
D/   . SEG3

List of extracted features
▬ □ □ □ □ ▬ ▬ □ □ ▬ □ □ □ □ □ □ □ ▬ ▬ ▬ ═ ═ ═ ▬ ▬ ▬ ═

["c_feature","class_totals"] 2
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["f_feature","fp2-f3","pk1","sub_delta","low_power","high_speed"] 0.5
["f_feature","fp2-f3","pk1","delta","low_power","low_speed"] 0.5
["f_feature","fp2-f3","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk1","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp2-f4","pk1","delta","med_power","med_speed"] 1
["f_feature","fp1-f3","pk1","delta","med_power","med_speed"] 1
["f_feature","fp1-f3","pk2","delta","low_power","high_speed"] 0.5
["f_feature","f4-c4","pk1","delta","vhigh_power","med_speed"] 1
["f_feature","c4-p4","pk1","delta","med_power","med_speed"] 1
["f_feature","f3-c3","pk1","delta","vhigh_power","med_speed"] 1
["f_feature","c3-p3","pk1","delta","med_power","med_speed"] 1
["f_feature","p3-o1","pk1","delta","low_power","med_speed"] 1
["f_feature","f3-t4","pk1","delta","med_power","med_speed"] 1
["f_feature","t4-t6","pk1","sub_delta","low_power","high_speed"] 0.75
["f_feature","t4-t6","pk1","delta","low_power","low_speed"] 0.25
["f_feature","t4-t6","pk2","delta","med_power","med_speed"] 1
["f_feature","t6-o2","pk1","delta","low_power","med_speed"] 1
["f_feature","f7-t3","pk1","delta","med_power","med_speed"] 1
["f_feature","t3-t5","pk1","delta","med_power","med_speed"] 1
["f_feature","t5-o1","pk1","delta","low_power","med_speed"] 1
["f_feature","fp2-f3","pk1","sub_delta","low_power","high_speed"] 0.5
["f_feature","fp2-f3","pk1","delta","low_power","low_speed"] 0.5
["f_feature","fp2-f3","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk1","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 0.5
not there is only very slow activity present, less than 0.5Hz 0.5

any significant spectral peaks exist in the delta band 0.5
not the largest spectral peak appears in a frontal channel 0.25
not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 0.0529157667
not fp2-f4 and fp1-f3 are symmetrical for all delta peaks 0.5
not delta peaks are present that are not present in the EOG 0.25
not the largest spectral peak appears in the EOG 1
the largest spectral peak is attributable to more than one electrode 0.25
there is a significant spectral peak in an EOG channel 0.5
there is a correlation between frontal and posterior electrodes 1

segment was classified as containing
abnormal slow waves only with a certainty of 0.0264578834.

The users response was y

12-3-1991 17:35
DATA5 SEG1

List of extracted features
■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ = = ■ ■ ■ ■ □ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ =

["c_feature","local","abnormal"] 0.0264578834
["c_feature","historical","abnormal"] 0.0070554356
["c_feature","class_totals"] 3
["f_feature","fp2-f8","pk1","delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk1","delta","med_power","med_speed"] 1
["f_feature","fp2-f4","pk1","delta","low_power","med_speed"] 1
["f_feature","fp1-f3","pk1","delta","low_power","med_speed"] 1
["f_feature","f4-c4","pk1","delta","low_power","med_speed"] 1
["f_feature","c4-p4","pk1","delta","low_power","med_speed"] 1
["f_feature","f3-c3","pk1","delta","med_power","med_speed"] 1
["f_feature","c3-p3","pk1","delta","low_power","med_speed"] 1
["f_feature","p3-o1","pk1","delta","low_power","med_speed"] 1
["f_feature","f8-t4","pk1","delta","low_power","med_speed"] 1
["f_feature","t4-t6","pk1","delta","low_power","med_speed"] 1
["f_feature","f7-t3","pk1","delta","low_power","med_speed"] 1
["f_feature","t3-t5","pk1","delta","low_power","med_speed"] 1
["f_feature","t3-t5","pk2","delta","low_power","med_speed"] 1
["f_feature","t5-o1","pk1","delta","low_power","med_speed"] 1
["f_feature","fp2-f8","pk1","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk1","delta","med_power","med_speed"] 1
the historical context is abnormal 0.0070554356
the local context is abnormal slow waves only 0.0264578834
any significant spectral peaks exist in the delta band 1
the largest spectral peak is attributable to more than one electrode 1
there is a significant spectral peak in an EOG channel 1
not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 0.0529157667
there is a correlation between frontal and posterior electrodes 1

segment was classified as containing
abnormal slow waves only with a certainty of 0.0452924636.

The users response was y

12-3-1991 17:36
DATA5 SEG4

List of extracted features
■ ■ □ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ = = □ □ ■ ■ ■ ■ = = ■ ■ ■ ■ ■ ■ ■ =

["c_feature","local","abnormal"] 0.0452924636

["c_feature","historical","abnormal"] 0.0250445978
["c_feature","class_totals"] 4
["f_feature","fp2-f8","pk1","delta","low_power","med_speed"] 1
the historical context is abnormal 0.0250445978
the local context is abnormal slow waves only 0.0452924636
any significant spectral peaks exist in the delta band 1
not the largest spectral peak is attributable to more than one electrode 1
there is a significant spectral peak in an EOG channel 1
not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 0.0529157667
there is a correlation between frontal and posterior electrodes 1

segment was classified as containing
abnormal slow waves only with a certainty of 0.0647740454.

The users response was y

12-3-1991 17:37
DATA5 SEG2

List of extracted features
■ ■ ■ ■ ■ = = ■

["c_feature","local","abnormal"] 0.0647740454
["c_feature","historical","abnormal"] 0.0553574644
["c_feature","class_totals"] 5
the historical context is abnormal 0.0553574644
the local context is abnormal slow waves only 0.0647740454
not any significant spectral peaks exist in the delta band 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
a local context exists 0.0647740454
not any significant spectral peaks exist 1

segment was classified as containing
no significant activity with a certainty of 1.

The users response was y

12-3-1991 17:39
DATA5 SEG1

List of extracted features
■ ◦ ■ ■ ◦ ■ ◦ ◦ ◦ ◦ ◦ ◦ ◦ ◦ = = = = = = = = = = = = = =

["c_feature","class_totals"] 6
["c_feature","class_totals"] 0
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","high_speed"] 0.75
["f_feature","fp2-f8","pk1","delta","vhigh_power","low_speed"] 0.25
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","high_speed"] 0.75
["f_feature","fp1-f7","pk1","delta","vhigh_power","low_speed"] 0.25
["f_feature","fp1-f7","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","med_power","high_speed"] 1
["f_feature","fp2-f4","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","high_speed"] 1
["f_feature","fp2-f8","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","low_power","high_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1

not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 0.75
not there is only very slow activity present, less than 0.5Hz 0.25
any significant spectral peaks exist in the delta band 0.25
the largest spectral peak appears in a frontal channel 0.25
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
not fp2-f8 and fp1-f7 are symmetrical for all delta peaks 0.25
delta activity only appears in frontal channels 0.25
there are l/r symetrical delta peaks 0.0529157667
delta spectral peaks exist only in the EOG 0.25
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 0.25
the waveform in channels fp2-f8 and f8-t4 are in phase 0
the largest spectral peak appears in the EOG 1

segment was classified as containing
weak em artefact with a certainty of 0.681699676.

The users response was y

13-3-1991 11:53
D.... 1 SEG1

List of extracted features
■ ■ ■ ■ ■ ■ ■ ■ ■ ■ = = = = ■ ■ ■ ■ ■ ■ ■ ■ = ■ = = ■ ■ ■

["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
not any significant spectral peaks exist 1

segment was classified as containing
no significant activity with a certainty of 1.

The users response was y

13 - '991 11:55
D,   . SEG2

List of extracted features
■ ■ ■ ■ ■ ■ ■ ■ ■ = = = = = = = = = ■ ■ ■ ■ ■ ■ = = ■ ■ ■

["c_feature","local","no"] 1
["c_feature","class_totals"] 1
["f_feature","fp1-f7","pk1","delta","low_power","med_speed"] 1
["f_feature","fp2-f4","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp1-f3","pk1","sub_delta","med_power","high_speed"] 1
["f_feature","f8-t4","pk1","sub_delta","med_power","high_speed"] 1
["f_feature","f8-t4","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk1","sub_delta","med_power","high_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 1
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1

not fp2-f8 and fp1-f7 are symmetrical for all delta peaks 1
not delta activity only appears in frontal channels 1
delta activity only appears in the anterior half of the scalp 1
not delta peaks are present that are not present in the EOG 1
fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 1
not fp2-f4 or fp1-f3 contains waveforms with sharp elements 1
not delta spectral peaks exist only in the EOG 1
the largest spectral peak is attributable to more than one electrode 1
there is a significant spectral peak in an EOG channel 1
(not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
not the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 1
the waveform in channels fp2-f8 and f8-t4 are in phase 0
the waveform in channels fp1-f7 and f7-t3 are in phase 0

segment was classified as containing
blink artefact with a certainty of 0.8.

The users response was y

13-3-1991 11:57
DATA1 SEG3

Li: extracted features
===

['c_feature','local','blink'] 0.8
['c_feature','class_totals'] 2
['f_feature','fp2-f8','pk1','sub_delta','med_power','med_speed'] 1
['f_feature','fp2-f8','pk2','delta','low_power','high_speed'] 1
['f_feature','fp1-f7','pk1','delta','low_power','med_speed'] 1
['f_feature','fp1-f7','pk2','delta','low_power','high_speed'] 1
['f_feature','fp2-f4','pk1','sub_delta','med_power','med_speed'] 1
['f_feature','f7-t3','pk1','sub_delta','med_power','med_speed'] 1
['f_feature','fp2-f8','pk1','sub_delta','vhigh_power','med_speed'] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
an   .ificant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 1
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
fp2-f8 and fp1-f7 are symmetrical for all delta peaks 1
not delta activity only appears in frontal channels 1
delta activity only appears in the anterior half of the scalp 1
not delta peaks are present that are not present in the EOG 1
not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 1
delta spectral peaks exist only in the EOG 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
not the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 1
the waveform in channels fp2-f8 and f8-t4 are in phase 0
the waveform in channels fp1-f7 and f7-t3 are in phase 0

segment was classified as containing
blink artefact with a certainty of 0.85.

The users response was y

13-3-1991 11:59
DATA1 SEG4

List of extracted features

```
================================
```

["c_feature","local","blink"] 0.85
["c_feature","class_totals"] 3
["f_feature","fp1-f7","pk1","delta","low_power","med_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 1
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
not fp2-f8 and fp1-f7 are symmetrical for all delta peaks 1
delta activity only appears in frontal channels 1
there are l/r symetrical delta peaks 1
delta spectral peaks exist only in the EOG 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
not the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 1
the waveform in channels fp2-f8 and f8-t4 are in phase 0
th.   eform in channels fp1-f7 and f7-t3 are in phase 0

segment was classified as containing
blink artefact with a certainty of 0.901.

The users response was y

13-3-1991 12:1
DATA3 SEG1

List of extracted features
```
================================
```

["c_feature","class_totals"] 4
["c_feature","class_totals"] 0
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","high_speed"] 1
["f_feature","fp2-f8","pk2","delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk3","delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk4","delta","med_power","high_speed"] 0.75
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","low_power","high_speed"] 0.75
["f_feature","fp2-f4","pk1","sub_delta","vhigh_power","high_speed"] 1
["f_feature","fp2-f4","pk2","delta","med_power","med_speed"] 1
["f_feature","fp2-f4","pk3","delta","low_power","med_speed"] 1
["f_feature","fp1-f3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f3","pk2","delta","low_power","med_speed"] 1
["f_feature","f4-c4","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","p3-o1","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","f8-t4","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f8-t4","pk2","delta","med_power","med_speed"] 1
["f_feature","f8-t4","pk3","delta","med_power","med_speed"] 1
["f_feature","f7-t3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f7-t3","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","f7-t3","pk2","delta","med_power","low_speed"] 0.75
["f_feature","f7-t3","pk3","delta","low_power","med_speed"] 1
["f_feature","f7-t3","pk4","delta","low_power","high_speed"] 0.75
["f_feature","t5-o1","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","high_speed"] 1
["f_feature","fp2-f8","pk2","delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk3","delta","med_power","med_speed"] 1

59
```

["f_feature","fp2-f8","pk4","delta","low_power","high_speed"] 1
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","low_power","high_speed"] 0.75
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 0.25
not fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
not fp2-f8 and fp1-f7 are symmetrical for all delta peaks 1
not delta peaks are present that are not present in the EOG 0.75
not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 1
delta activity only appears in the anterior half of the scalp 0.75
not delta spectral peaks exist only in the EOG 0.75
the largest spectral peak appears in the EOG 1
the largest spectral peak is attributable to more than one electrode 0.25
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
not the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 1
the      cform in channels fp2-f8 and f8-t4 are in phase 0
the waveform in channels fp1-f7 and f7-t3 are in phase 0


segment was classified as containing
blink artefact with a certainty of 0.1675.


The users response was y


13-3-1991  12:6
DATA3 SEG2


List of extracted features
==============================


["c_feature","local","blink"] 0.1675
["c_feature","class_totals"] 5
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","med_power","med_speed"] 1
["f_feature","fp2-f4","pk1","sub_delta","med_power","high_speed"] 1
["f_feature","fp2-f4","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f3","pk1","sub_delta","med_power","high_speed"] 1
["f_feature","fp1-f3","pk2","delta","low_power","med_speed"] 1
["f_feature","f4-c4","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","c4-p4","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","p4-o2","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","f3-c3","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","p3-o1","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","f8-t4","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f8-t4","pk2","delta","low_power","med_speed"] 1
["f_feature","f8-t4","pk3","delta","low_power","med_speed"] 1
["f_feature","t4-t6","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","t6-o2","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","f7-t3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f7-t3","pk2","delta","med_power","med_speed"] 1
["f_feature","t5-o1","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","med_power","med_speed"] 1

not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 1
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
fp2-f3 and fp1-f7 are symmetrical for all delta peaks 1
not delta activity only appears in frontal channels 1
delta activity only appears in the anterior half of the scalp 1
not delta peaks are present that are not present in the EOG 1
not fp2-f3 or fp1-f7 contains isolated non repetitive waveforms 1
not delta spectral peaks exist only in the EOG 1
the largest spectral peak is attributable to more than one electrode 1
the largest spectral peak appears in the EOG 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
not the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 1
the waveform in channels fp2-f3 and f3-t4 are in phase 0
the waveform in channels fp1-f7 and f7-t3 are in phase 0


sei      , was classified as containing
blink artefact with a certainty of 0.9.


The users response was n

13-3-1991  12:9
DATA3 SEG3

List of extracted features
= = = = = = = = = = = = = = = = = = = = = = = = = = =


['c_feature','class_totals'] 5
['f_feature','fp2-f3','pk1','sub_delta','vhigh_power','med_speed'] 1
['f_feature','fp2-f3','pk2','sub_delta','med_power','high_speed'] 0.25
['f_feature','fp2-f3','pk2','delta','med_power','low_speed'] 0.75
['f_feature','fp2-f3','pk3','delta','med_power','med_speed'] 1
['f_feature','fp1-f7','pk1','sub_delta','vhigh_power','med_speed'] 1
['f feature','fp1-f7','pk2','sub_delta','low_power','high_speed'] 0.25
['1    ure','fp1-f7','pk2','delta','low_power','low_speed'] 0.75
['f_feature','fp1-f7','pk3','delta','low_power','med_speed'] 1
['f_feature','fp1-f7','pk4','delta','low_power','high_speed'] 0.75
['f_feature','fp2-f4','pk1','sub_delta','med_power','high_speed'] 1
['f_feature','fp1-f3','pk1','sub_delta','vhigh_power','high_speed'] 1
['f_feature','p4-o2','pk1','sub_delta','med_power','med_speed'] 1
['f_feature','f3-c3','pk1','sub_delta','med_power','med_speed'] 1
['f_feature','p3-o1','pk1','sub_delta','med_power','med_speed'] 1
['f_feature','f3-t4','pk1','sub_delta','vhigh_power','med_speed'] 1
['f_feature','f3-t4','pk2','delta','low_power','med_speed'] 1
['f_feature','f3-t4','pk3','delta','low_power','high_speed'] 1
['f_feature','t6-o2','pk1','sub_delta','med_power','med_speed'] 1
['f_feature','f7-t3','pk1','sub_delta','vhigh_power','med_speed'] 1
['f_feature','f7-t3','pk2','delta','low_power','med_speed'] 1
['f_feature','f7-t3','pk3','delta','low_power','high_speed'] 0.75
['f_feature','t5-o1','pk1','sub_delta','med_power','med_speed'] 1
['f_feature','fp2-f3','pk1','sub_delta','vhigh_power','med_speed'] 1
['f_feature','fp1-f7','pk1','sub_delta','vhigh_power','med_speed'] 1
['f_feature','fp1-f7','pk2','delta','low_power','high_speed'] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1

any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 0.25
any significant spectral peaks exist in the delta band 0.75
the largest spectral peak appears in a frontal channel 0.25
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
not fp2-f8 and fp1-f7 are symmetrical for all delta peaks 0.75
not delta activity only appears in frontal channels 0.25
delta activity only appears in the anterior half of the scalp 0.75
not delta peaks are present that are not present in the EOG 0.75
not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 1
not delta spectral peaks exist only in the EOG 0.75
the largest spectral peak is attributable to more than one electrode 0.25
there is a significant spectral peak in an EOG channel 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 0.75
the waveform in channels fp2-f8 and f8-t4 are in phase 0
the largest spectral peak appears in the EOG 1


segment was classified as containing
weak em artefact with a certainty of 0.7025.


The ***ers response was n


13-3-1991  12:11
DATA3 SEG4


List of extracted features
■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ = = =


["c_feature","class_totals"] 5
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f8","pk2","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f8","pk3","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","low_power","med_speed"] 1
["f_feature","fp2-f4","pk1","sub_delta","vhigh_power","high_speed"] 1
["f_feature","fp1-f3","pk1","sub_delta","med_power","high_speed"] 1
["f_feature","f3-c3","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","p3-o1","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","f8-t4","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f7-t3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f7-t3","pk2","delta","low_power","med_speed"] 1
["f_feature","t5-o1","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 0.25
any significant spectral peaks exist in the delta band 0.75
the largest spectral peak appears in a frontal channel 0.25
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
not fp2-f8 and fp1-f7 are symmetrical for all delta peaks 0.75
not delta activity only appears in frontal channels 0.25
delta activity only appears in the anterior half of the scalp 0.75
not delta peaks are present that are not present in the EOG 1
not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 1
not delta spectral peaks exist only in the EOG 0.75
the largest spectral peak is attributable to more than one electrode 0.25

any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 0.25
any significant spectral peaks exist in the delta band 0.75
the largest spectral peak appears in a frontal channel 0.25
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
not fp2-f8 and fp1-f7 are symmetrical for all delta peaks 0.75
not delta activity only appears in frontal channels 0.25
delta activity only appears in the anterior half of the scalp 0.75
not delta peaks are present that are not present in the EOG 0.75
not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 1
not delta spectral peaks exist only in the EOG 0.75
the largest spectral peak is attributable to more than one electrode 0.25
there is a significant spectral peak in an EOG channel 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 0.75
the waveform in channels fp2-f8 and f8-t4 are in phase 0
the largest spectral peak appears in the EOG 1


segment was classified as containing
weak em artefact with a certainty of 0.7025.


The ···ers response was n


13-3-1991  12:11
DATA3 SEG4


List of extracted features
■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■


["c_feature","class_totals"] 5
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f8","pk2","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f8","pk3","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","low_power","med_speed"] 1
["f_feature","fp2-f4","pk1","sub_delta","vhigh_power","high_speed"] 1
["f_feature","fp1-f3","pk1","sub_delta","med_power","high_speed"] 1
["f_ ·ture","f3-c3","pk1","sub_delta","med_power","med_speed"] 1
["f_ .ure","p3-o1","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","f8-t4","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f7-t3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f7-t3","pk2","delta","low_power","med_speed"] 1
["f_feature","t5-o1","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 0.25
any significant spectral peaks exist in the delta band 0.75
the largest spectral peak appears in a frontal channel 0.25
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
not fp2-f8 and fp1-f7 are symmetrical for all delta peaks 0.75
not delta activity only appears in frontal channels 0.25
delta activity only appears in the anterior half of the scalp 0.75
not delta peaks are present that are not present in the EOG 1
not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 1
not delta spectral peaks exist only in the EOG 0.75
the largest spectral peak is attributable to more than one electrode 0.25

List of extracted features

==============================

["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["f_feature","fp2-f4","pk1","delta","low_power","med_speed"] 1
["f_feature","fp1-f3","pk1","delta","low_power","med_speed"] 1
["f_feature","f4-c4","pk1","delta","low_power","med_speed"] 1
["f_feature","p4-o2","pk1","delta","low_power","med_speed"] 1
["f_feature","c3-p3","pk1","delta","low_power","med_speed"] 1
["f_feature","f8-t4","pk1","delta","low_power","med_speed"] 1
["f_feature","f7-t3","pk1","delta","low_power","med_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 1
fp2   and fp1-f3 are symmetrical for all delta peaks 1
fp2-f8 and fp1-f7 are symmetrical for all delta peaks 1
not delta activity only appears in frontal channels 1
not delta activity only appears in the anterior half of the scalp 1
delta peaks are present that are not present in the EOG 1
fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 1
not significant delta activity does not appear in occipital channels 1
not the largest spectral peak appears in the EOG 1

segment was classified as containing
mainly_OA with little abnormal slow waves with a certainty of 0.97.

The users response was n

.

13-3-1991  13:37
DATA6 SEG2

List of extracted features

==============================

["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["f_feature","c4-p4","pk1","delta","low_power","med_speed"] 1
["f_feature","t4-t6","pk1","delta","low_power","med_speed"] 1
["f_feature","t4-t6","pk2","delta","low_power","med_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
not the largest spectral peak appears in a frontal channel 1
fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 1
significant delta activity does not appear in occipital channels 1
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
delta peaks are present that are not present in the EOG 1
not fp2-f4 or fp1-f3 contains waveforms with sharp elements 1
not the largest spectral peak exists in the sub-delta band 1
the largest spectral peak is attributable to more than one electrode 1
not there is a significant spectral peak in an EOG channel 1

segment was classified as containing
abnormal slow waves only with a certainty of 0.8.

The users response was n

13-3-1991  16:55
DATA6 SEG3

List of extracted features
■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ = = ■ ■ ■ ■ = = ■ ■ = = = =

["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["f_feature","f4-c4","pk1","delta","low_power","med_speed"] 1
13-3-1991  16:58
DATA6 SEG4

List of extracted features
■ ■ = = ■ ■ ■ ■ ■ ■ = = = = = = = = = ■ ■ ■ ■ ■ ■ ■ ■ ■ = =

["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["f_feature","t4-t6","pk1","delta","low_power","med_speed"] 1
["f_feature","f7-t3","pk1","delta","low_power","high_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
not the largest spectral peak appears in a frontal channel 1
fp2-f3 or fp1-f7 contains isolated non repetitive waveforms 1
significant delta activity does not appear in occipital channels 1
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
delta peaks are present that are not present in the EOG 1
not fp2-f4 or fp1-f3 contains waveforms with sharp elements 1
not the largest spectral peak exists in the sub-delta band 1
the    est spectral peak is attributable to more than one electrode 1
not there is a significant spectral peak in an EOG  channel 1

segment was classified as containing
abnormal slow waves only with a certainty of 0.8.

The users response was n

13-3-1991  17:4
DATA4 SEG1

List of extracted features
= = = = = = ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ = = = = ■ ■ ■ = = ■ ■ = =

["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["f_feature","f3-t4","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","f7-t3","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp2-f3","pk1","sub_delta","low_power","med_speed"] 1

515

not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
there is only very slow activity present, less than 0.5Hz 1
the largest spectral peak appears anteriorly 1
the largest spectral peak is attributable to more than one electrode 1


segment was classified as containing
rem artefact only with a certainty of 0.98.


The users response was y

13-3-1991  17:6
DATA4 SEG2

List of extracted features
■ ■ ■ ■ ■ ◻ ◻ ■ ■ ■ ■ ■ ■ ● ● ◻ ◻ ● ◻ ◻ ◻ ◻ ◻ ◻ ■ ■ ◻ ◻ ◻ ◻ ◻ ◻ ◻


["c_feature","local","rem"] 0.98
["c_...ure","class_totals"] 1
["f_feature","fp2-f8","pk1","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk1","delta","low_power","med_speed"] 1
["f_feature","f8-t4","pk1","delta","low_power","med_speed"] 1
["f_feature","f7-t3","pk1","delta","low_power","med_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 1
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
fp2-f8 and fp1-f7 are symmetrical for all delta peaks 1
not delta activity only appears in frontal channels 1
delta activity only appears in the anterior half of the scalp 1
not delta peaks are present that are not present in the EOG 1
fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 1
no.    -f4 or fp1-f3 contains waveforms with sharp elements 1
not delta spectral peaks exist only in the EOG 1
the largest spectral peak is attributable to more than one electrode 1
the largest spectral peak appears in the EOG 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
not the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 1
the waveform in channels fp2-f8 and f8-t4 are in phase 0
the waveform in channels fp1-f7 and f7-t3 are in phase 0


segment was classified as containing
blink artefact with a certainty of 0.9.


The users response was n

13-3-1991  17:7
DATA4 SEG3

List of extracted features
■ ■ ■ ◻ ◻ = = ◻ ■ ■ ■ ■ ■ ◻ ◻ ◻ ◻ ◻ ◻ ◻ ◻ ◻ ◻ ■ ■ ■ ◻ ◻ ◻ = ◻


["c_feature","class_totals"] 1
["f_feature","fp2-f8","pk1","sub_delta","med_power","high_speed"] 0.75

["f_feature","fp2-f8","pk1","delta","med_power","low_speed"] 0.25
["f_feature","fp2-f8","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk1","sub_delta","med_power","high_speed"] 0.75
["f_feature","fp1-f7","pk1","delta","med_power","low_speed"] 0.25
["f_feature","fp1-f7","pk2","delta","med_power","med_speed"] 1
["f_feature","f8-t4","pk1","sub_delta","vhigh_power","high_speed"] 0.75
["f_feature","f8-t4","pk1","delta","vhigh_power","low_speed"] 0.25
["f_feature","f7-t3","pk1","sub_delta","med_power","high_speed"] 0.75
["f_feature","f7-t3","pk1","delta","med_power","low_speed"] 0.25
["f_feature","fp2-f8","pk1","sub_delta","med_power","high_speed"] 0.75
["f_feature","fp2-f8","pk1","delta","med_power","low_speed"] 0.25
["f_feature","fp1-f7","pk1","sub_delta","med_power","high_speed"] 0.75
["f_feature","fp1-f7","pk1","delta","med_power","low_speed"] 0.25
13-3-1991 17:9
DATA4 SEG4


List of extracted features
■ ■ ■ □ ■ ■ ■ ■ ■ ■ = = = = □ ■ ■ ■ ■ □ □ □ = ■ = = = ■ ■ ■ □


["c_feature","class_totals"] 1
["f feature","fp2-f8","pk1","delta","low_power","med_speed"] 1
["f   ure","f8-t4","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","f8-t4","pk2","delta","low_power","med_speed"] 1
["f_feature","f8-t4","pk3","delta","low_power","med_speed"] 1
["f_feature","f7-t3","pk1","sub_delta","med_power","high_speed"] 1
["f_feature","f7-t3","pk2","delta","low_power","med_speed"] 1
["f_feature","f7-t3","pk3","delta","low_power","med_speed"] 1
["f_feature","fp2-f8","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk2","delta","low_power","med_speed"] 1
["f_feature","fp2-f8","pk3","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk1","sub_delta","med_power","high_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 1
fp:    and fp1-f3 are symmetrical for all delta peaks 1
fp2-f8 and fp1-f7 are symmetrical for all delta peaks 1
not delta activity only appears in frontal channels 1
delta activity only appears in the anterior half of the scalp 1
not delta peaks are present that are not present in the EOG 1
fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 1
not fp2-f4 or fp1-f3 contains waveforms with sharp elements 1
not delta spectral peaks exist only in the EOG 1
the largest spectral peak is attributable to more than one electrode 1
the largest spectral peak appears in the EOG 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
not the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 1
the waveform in channels fp2-f8 and f8-t4 are in phase 0
the waveform in channels fp1-f7 and f7-t3 are in phase 0


segment was classified as containing
blink artefact with a certainty of 0.9.


The users response was n


13-3-1991  17:12
DATA7 SEG1

any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 1
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
fp2-f8 and fp1-f7 are symmetrical for all delta peaks 1
delta activity only appears in frontal channels 1
not delta spectral peaks exist only in the EOG 1
the largest spectral peak is attributable to more than one electrode 1
the largest spectral peak appears in the EOG 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
not the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 1
the waveform in channels fp2-f8 and f8-t4 are in phase 0
the waveform in channels fp1-f7 and f7-t3 are in phase 0


segment was classified as containing
blink artefact with a certainty of 0.936.


The users response was y


13-3-1991  17:19
DATA2 SEG1


List of extracted features
◻      ◻◻◻◻◻◻◻◻◻◻◻◻◻◻◻◻◻◻◻◻◻◻◻◻◻◻◻


["c_feature","class_totals"] 3
["c_feature","class_totals"] 0
["f_feature","fp2-f8","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f8","pk2","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f8","pk3","delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk4","delta","low_power","high_speed"] 0.5
["f_feature","fp1-f7","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk4","delta","med_power","high_speed"] 0.75
["f_feature","fp2-f4","pk1","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f4","pk1","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f4","pk2","delta","med_power","med_speed"] 1
["f_feature","fp2-f4","pk3","delta","med_power","high_speed"] 0.75
["f_feature","fp1-f3","pk1","sub_delta","med_power","high_speed"] 0.25
["t    ure","fp1-f3","pk1","delta","med_power","low_speed"] 0.75
["f_feature","fp1-f3","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f3","pk3","delta","low_power","high_speed"] 0.75
["f_feature","f8-t4","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f8-t4","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","f8-t4","pk2","delta","med_power","low_speed"] 0.75
["f_feature","f8-t4","pk3","delta","med_power","med_speed"] 1
["f_feature","f8-t4","pk4","delta","low_power","high_speed"] 0.75
["f_feature","f7-t3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f7-t3","pk2","sub_delta","low_power","high_speed"] 0.25
["f_feature","f7-t3","pk2","delta","low_power","low_speed"] 0.75
["f_feature","f7-t3","pk3","delta","low_power","med_speed"] 1
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f8","pk2","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f8","pk3","delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk4","delta","low_power","high_speed"] 0.5
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk4","delta","low_power","high_speed"] 0.75
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1

not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 0.25
any significant spectral peaks exist in the delta band 0.75
the largest spectral peak appears in a frontal channel 0.25
not fp2-f4 and fp1-f3 are symmetrical for all delta peaks 0.75
fp2-f8 and fp1-f7 are symmetrical for all delta peaks 0.5
not delta activity only appears in frontal channels 0.25
not delta peaks are present that are not present in the EOG 0.5
not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 0.6082251082
delta activity only appears in the anterior half of the scalp 0.5
not delta spectral peaks exist only in the EOG 0.5
the largest spectral peak is attributable to more than one electrode 0.25
the largest spectral peak appears in the EOG 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 0.75
the waveform in channels fp2-f8 and f8-t4 are in phase 0

segment was classified as containing
weak em artefact with a certainty of 0.712.


Th      rs response was n


13-3-1991  17:21
DATA2 SEG2

List of extracted features
■ = ■ = ■ ■ ■ ■ = ■ ■ ■ ■ ■ ■ ■ ■ = ■ = = = = ■ = ■ ■ ■ ■


["c_feature","class_totals"] 3
["c_feature","class_totals"] 0
["f_feature","fp2-f8","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f8","pk2","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f8","pk3","delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk4","delta","low_power","high_speed"] 0.5
["f_feature","fp1-f7","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","med_power","med_speed"] 1
["f    ure","fp1-f7","pk4","delta","med_power","high_speed"] 0.75
["f_feature","fp2-f4","pk1","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f4","pk1","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f4","pk2","delta","med_power","med_speed"] 1
["f_feature","fp2-f4","pk3","delta","med_power","high_speed"] 0.75
["f_feature","fp1-f3","pk1","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp1-f3","pk1","delta","med_power","low_speed"] 0.75
["f_feature","fp1-f3","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f3","pk3","delta","low_power","high_speed"] 0.75
["f_feature","f8-t4","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f8-t4","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","f8-t4","pk2","delta","med_power","low_speed"] 0.75
["f_feature","f8-t4","pk3","delta","med_power","med_speed"] 1
["f_feature","f8-t4","pk4","delta","low_power","high_speed"] 0.75
["f_feature","f7-t3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f7-t3","pk2","sub_delta","low_power","high_speed"] 0.25
["f_feature","f7-t3","pk2","delta","low_power","low_speed"] 0.75
["f_feature","f7-t3","pk3","delta","low_power","med_speed"] 1
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f8","pk2","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f8","pk3","delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk4","delta","low_power","high_speed"] 0.5

["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk4","delta","low_power","high_speed"] 0.75
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 0.25
any significant spectral peaks exist in the delta band 0.75
the largest spectral peak appears in a frontal channel 0.25
not fp2-f4 and fp1-f3 are symmetrical for all delta peaks 0.75
fp2-f3 and fp1-f7 are symmetrical for all delta peaks 0.5
not delta activity only appears in frontal channels 0.25
not delta peaks are present that are not present in the EOG 0.5
not fp2-f3 or fp1-f7 contains isolated non repetitive waveforms 0.6082251082
delta activity only appears in the anterior half of the scalp 0.5
not delta spectral peaks exist only in the EOG 0.5
the largest spectral peak is attributable to more than one electrode 0.25
the largest spectral peak appears in the EOG 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
the    eform in channels fp2-f4 or fp1-f3 is slow -low delta 0.75
the waveform in channels fp2-f3 and f3-t4 are in phase 0

segment was classified as containing
weak em artefact with a certainty of 0.712.

The users response was
13-3-1991  17:22
DATA2 SEG2

List of extracted features
= = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =


["c_feature","class_totals"] 3
["c_feature","class_totals"] 0
["f_feature","fp2-f3","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp2-f3","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f3","pk2","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f3","pk3","delta","med_power","med_speed"] 1
["f_feature","fp2-f3","pk4","delta","low_power","high_speed"] 0.5
["f_feature","fp1-f7","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk4","delta","med_power","high_speed"] 0.75
["f_feature","fp2-f4","pk1","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f4","pk1","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f4","pk2","delta","med_power","med_speed"] 1
["f_feature","fp2-f4","pk3","delta","med_power","high_speed"] 0.75
["f_feature","fp1-f3","pk1","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp1-f3","pk1","delta","med_power","low_speed"] 0.75
["f_feature","fp1-f3","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f3","pk3","delta","low_power","high_speed"] 0.75
["f_feature","f3-t4","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f3-t4","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","f3-t4","pk2","delta","med_power","low_speed"] 0.75
["f_feature","f3-t4","pk3","delta","med_power","med_speed"] 1
["f_feature","f3-t4","pk4","delta","low_power","high_speed"] 0.75
["f_feature","f7-t3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f7-t3","pk2","sub_delta","low_power","high_speed"] 0.25
["f_feature","f7-t3","pk2","delta","low_power","low_speed"] 0.75
["f_feature","f7-t3","pk3","delta","low_power","med_speed"] 1

List of extracted features
===================================

["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["f_feature","fp2-f8","pk1","delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk1","delta","med_power","med_speed"] 1
["f_feature","fp2-f4","pk1","delta","low_power","med_speed"] 1
["f_feature","fp1-f3","pk1","delta","low_power","med_speed"] 1
["f_feature","f4-c4","pk1","delta","low_power","med_speed"] 1
["f_feature","c4-p4","pk1","delta","low_power","med_speed"] 1
["f_feature","f3-c3","pk1","delta","med_power","med_speed"] 1
["f_feature","c3-p3","pk1","delta","low_power","med_speed"] 1
["f_feature","p3-o1","pk1","delta","low_power","med_speed"] 1
["f_feature","f8-t4","pk1","delta","low_power","med_speed"] 1
["f_feature","t4-t6","pk1","delta","low_power","med_speed"] 1
["f_feature","f7-t3","pk1","delta","low_power","med_speed"] 1
["f_feature","t3-t5","pk1","delta","low_power","med_speed"] 1
["f_feature","t3-t5","pk2","delta","low_power","med_speed"] 1
["f_feature","t5-o1","pk1","delta","low_power","med_speed"] 1
["f_feature","fp2-f8","pk1","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk1","delta","med_power","med_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 1
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
not fp2-f8 and fp1-f7 are symmetrical for all delta peaks 1
not delta activity only appears in frontal channels 1
not delta activity only appears in the anterior half of the scalp 1
not delta peaks are present that are not present in the EOG 1
no     f8 or fp1-f7 contains isolated non repetitive waveforms 1
not fp2-f4 or fp1-f3 contains waveforms with sharp elements 1
the largest spectral peak appears in the EOG 1
there is a zero lag correlation between frontal and posterior electrodes 0.5158730159
not there is a phase delay in frontal channels 1

segment was classified as containing
detectable_OA in abnormal slow waves with a certainty of 0.85.

The users response was y

List of extracted features
===================================

["c_feature","local","detectable_OA"] 0.85
["c_feature","class_totals"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1

["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f8","pk2","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f8","pk3","delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk4","delta","low_power","high_speed"] 0.5
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk4","delta","low_power","high_speed"] 0.75
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 0.25
any significant spectral peaks exist in the delta band 0.75
the largest spectral peak appears in a frontal channel 0.25
not fp2-f4 and fp1-f3 are symmetrical for all delta peaks 0.75
fp2-f8 and fp1-f7 are symmetrical for all delta peaks 0.5
not delta activity only appears in frontal channels 0.25
not delta peaks are present that are not present in the EOG 0.5
not fn2-f8 or fp1-f7 contains isolated non repetitive waveforms 0.6082251032
de:    tivity only appears in the anterior half of the scalp 0.5
not delta spectral peaks exist only in the EOG 0.5
the largest spectral peak is attributable to more than one electrode 0.25
the largest spectral peak appears in the EOG 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 0.75
the waveform in channels fp2-f8 and f8-t4 are in phase 0

segment was classified as containing
weak em artefact with a certainty of 0.712.

The users response was n

13-3-1991  17:24
DATA2 SEG3

List of extracted features
==============================

["c_feature","class_totals"] 3
["c_feature","class_totals"] 0
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
not any significant spectral peaks exist 1

segment was classified as containing
no significant activity with a certainty of 1.

The users response was y

13-3-1991  17:25
DATA2 SEG4

List of extracted features
================================

["c_feature","local","no"] 1
["c_feature","class_totals"] 4

["f_feature","fp2-f8","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f8","pk2","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f8","pk3","delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk4","delta","low_power","high_speed"] 0.5
["f_feature","fp1-f7","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk4","delta","med_power","high_speed"] 0.75
["f_feature","fp2-f4","pk1","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f4","pk1","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f4","pk2","delta","med_power","med_speed"] 1
["f_feature","fp2-f4","pk3","delta","med_power","high_speed"] 0.75
["f_feature","fp1-f3","pk1","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp1-f3","pk1","delta","med_power","low_speed"] 0.75
["f_feature","fp1-f3","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f3","pk3","delta","low_power","high_speed"] 0.75
["f_feature","f8-t4","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f8-t4","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","f8-t4","pk2","delta","med_power","low_speed"] 0.75
["f_feature","f8-t4","pk3","delta","med_power","med_speed"] 1
["f_feature","f8-t4","pk4","delta","low_power","high_speed"] 0.75
["f_     ure","f7-t3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f7-t3","pk2","sub_delta","low_power","high_speed"] 0.25
["f_feature","f7-t3","pk2","delta","low_power","low_speed"] 0.75
["f_feature","f7-t3","pk3","delta","low_power","med_speed"] 1
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f8","pk2","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f8","pk3","delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk4","delta","low_power","high_speed"] 0.5
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk4","delta","low_power","high_speed"] 0.75
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 0.25
any  ificant spectral peaks exist in the delta band 0.75
the largest spectral peak appears in a frontal channel 0.25
not fp2-f4 and fp1-f3 are symmetrical for all delta peaks 0.75
fp2-f8 and fp1-f7 are symmetrical for all delta peaks 0.5
not delta activity only appears in frontal channels 0.25
not delta peaks are present that are not present in the EOG 0.5
not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 0.6082251082
delta activity only appears in the anterior half of the scalp 0.5
not delta spectral peaks exist only in the EOG 0.5
the largest spectral peak is attributable to more than one electrode 0.25
the largest spectral peak appears in the EOG 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 0.75
the waveform in channels fp2-f8 and f8-t4 are in phase 0

segment was classified as containing
weak em artefact with a certainty of 0.712.

The users response was
13-3-1991 17:26
DATA2 SEG4

List of extracted features

━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━

```
["c_feature","class_totals"] 4
["f_feature","fp2-f3","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp2-f3","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f3","pk2","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f3","pk3","delta","med_power","med_speed"] 1
["f_feature","fp2-f3","pk4","delta","low_power","high_speed"] 0.5
["f_feature","fp1-f7","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk4","delta","med_power","high_speed"] 0.75
["f_feature","fp2-f4","pk1","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f4","pk1","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f4","pk2","delta","med_power","med_speed"] 1
["f_feature","fp2-f4","pk3","delta","med_power","high_speed"] 0.75
["f_feature","fp1-f3","pk1","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp1-f3","pk1","delta","med_power","low_speed"] 0.75
["f_feature","fp1-f3","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f3","pk3","delta","low_power","high_speed"] 0.75
["f_feature","f3-t4","pk1","sub_delta","vhigh_power","med_speed"] 1
["f    ure","f3-t4","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","f3-t4","pk2","delta","med_power","low_speed"] 0.75
["f_feature","f3-t4","pk3","delta","med_power","med_speed"] 1
["f_feature","f3-t4","pk4","delta","low_power","high_speed"] 0.75
["f_feature","f7-t3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","f7-t3","pk2","sub_delta","low_power","high_speed"] 0.25
["f_feature","f7-t3","pk2","delta","low_power","low_speed"] 0.75
["f_feature","f7-t3","pk3","delta","low_power","med_speed"] 1
["f_feature","fp2-f3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f3","pk2","sub_delta","med_power","high_speed"] 0.25
["f_feature","fp2-f3","pk2","delta","med_power","low_speed"] 0.75
["f_feature","fp2-f3","pk3","delta","med_power","med_speed"] 1
["f_feature","fp2-f3","pk4","delta","low_power","high_speed"] 0.5
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk3","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk4","delta","low_power","high_speed"] 0.75
```

not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
no   local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 0.25
any significant spectral peaks exist in the delta band 0.75
the largest spectral peak appears in a frontal channel 0.25
not fp2-f4 and fp1-f3 are symmetrical for all delta peaks 0.75
fp2-f3 and fp1-f7 are symmetrical for all delta peaks 0.5
not delta activity only appears in frontal channels 0.25
not delta peaks are present that are not present in the EOG 0.5
not fp2-f3 or fp1-f7 contains isolated non repetitive waveforms 0.6082251032
delta activity only appears in the anterior half of the scalp 0.5
not delta spectral peaks exist only in the EOG 0.5
the largest spectral peak is attributable to more than one electrode 0.25
the largest spectral peak appears in the EOG 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 0.75
the waveform in channels fp2-f3 and f3-t4 are in phase 0

segment was classified as containing
weak em artefact with a certainty of 0.712.

The users response was n

14-3-1991 13:28
DATA5 SEG1

List of extracted features
===============================

["c_feature","class_totals"] 0
["c_feature","class_totals"] 0
["f_feature","fp2-f8","pk1","delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk1","delta","med_power","med_speed"] 1
["f_feature","fp2-f4","pk1","delta","low_power","med_speed"] 1
["f_feature","fp1-f3","pk1","delta","low_power","med_speed"] 1
["f_feature","f4-c4","pk1","delta","low_power","med_speed"] 1
["f_feature","c4-p4","pk1","delta","low_power","med_speed"] 1
["f_feature","f3-c3","pk1","delta","med_power","med_speed"] 1
["f_feature","c3-p3","pk1","delta","low_power","med_speed"] 1
["f_feature","p3-o1","pk1","delta","low_power","med_speed"] 1
["f_feature","f8-t4","pk1","delta","low_power","med_speed"] 1
["f_feature","t4-t6","pk1","delta","low_power","med_speed"] 1
["f   ure","f7-t3","pk1","delta","low_power","med_speed"] 1
["f_feature","t3-t5","pk1","delta","low_power","med_speed"] 1
["f_feature","t3-t5","pk2","delta","low_power","med_speed"] 1
["f_feature","t5-o1","pk1","delta","low_power","med_speed"] 1
["f_feature","fp2-f8","pk1","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk1","delta","med_power","med_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 1
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
not fp2-f8 and fp1-f7 are symmetrical for all delta peaks 1
not delta activity only appears in frontal channels 1
not delta activity only appears in the anterior half of the scalp 1
not delta peaks are present that are not present in the EOG 1
no   f8 or fp1-f7 contains isolated non repetitive waveforms 1
not fp2-f4 or fp1-f3 contains waveforms with sharp elements 1
the largest spectral peak appears in the EOG 1
there is a zero lag correlation between frontal and posterior electrodes 0.5158730159
not there is a phase delay in frontal channels 1

segment was classified as containing
detectable_OA in abnormal slow waves with a certainty of 0.85.

The users response was y

14-3-1991 13:31
DATA5 SEG2

List of extracted features
===============================

["c_feature","local","detectable_OA"] 0.85
["c_feature","class_totals"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1

the local context is detectable_OA in abnormal slow waves 0.35
not the largest spectral peak appears in a frontal channel 1
not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 1
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
not delta peaks are present that are not present in the EOG 1
not fp2-f4 or fp1-f3 contains waveforms with sharp elements 1
not the largest spectral peak appears in the EOG 1
not any significant spectral peaks exist in the delta band 1
not any significant spectral peaks exist 1

segment was classified as containing
no significant activity with a certainty of 1.

The users response was n

14-3-1991 13:32
DATA5 SEG3

List of extracted features
================================

["c   ure","class_totals"] 1
["f_feature","fp2-f8","pk1","sub_delta","low_power","high_speed"] 0.5
["f_feature","fp2-f8","pk1","delta","low_power","low_speed"] 0.5
["f_feature","fp2-f8","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk1","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp2-f4","pk1","delta","med_power","med_speed"] 1
["f_feature","fp1-f3","pk1","delta","med_power","med_speed"] 1
["f_feature","fp1-f3","pk2","delta","low_power","high_speed"] 0.5
["f_feature","f4-c4","pk1","delta","vhigh_power","med_speed"] 1
["f_feature","c4-p4","pk1","delta","med_power","med_speed"] 1
["f_feature","f3-c3","pk1","delta","vhigh_power","med_speed"] 1
["f_feature","c3-p3","pk1","delta","med_power","med_speed"] 1
["f_feature","p3-o1","pk1","delta","low_power","med_speed"] 1
["f_feature","f3-t4","pk1","delta","med_power","med_speed"] 1
["f_feature","t4-t6","pk1","sub_delta","low_power","high_speed"] 0.75
["f_feature","t4-t6","pk1","delta","low_power","low_speed"] 0.25
["f_feature","t4-t6","pk2","delta","med_power","med_speed"] 1
["f_feature","t6-o2","pk1","delta","low_power","med_speed"] 1
["f   ure","f7-t3","pk1","delta","med_power","med_speed"] 1
["f_feature","t3-t5","pk1","delta","med_power","med_speed"] 1
["f_feature","t5-o1","pk1","delta","low_power","med_speed"] 1
["f_feature","fp2-f8","pk1","sub_delta","low_power","high_speed"] 0.5
["f_feature","fp2-f8","pk1","delta","low_power","low_speed"] 0.5
["f_feature","fp2-f8","pk2","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk1","delta","med_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 0.5
not there is only very slow activity present, less than 0.5Hz 0.5
any significant spectral peaks exist in the delta band 0.5
not the largest spectral peak appears in a frontal channel 0.25
not fp2-f8 or fp1-f7 contains isolated non repetitive waveforms 1
not fp2-f4 and fp1-f3 are symmetrical for all delta peaks 0.5
not delta peaks are present that are not present in the EOG 0.25
not the largest spectral peak appears in the EOG 1
the largest spectral peak is attributable to more than one electrode 0.25
there is a significant spectral peak in an EOG channel 0.5
there is a correlation between frontal and posterior electrodes 0.5158730159

526

segment was classified as containing
abnormal slow waves only with a certainty of 0.25.

The users response was y

14-3-1991 13:34
DATA5 SEG4

List of extracted features
= =
.

['c_feature','local','abnormal'] 0.25
['c_feature','historical','abnormal'] 0.1
['c_feature','class_totals'] 2
['f_feature','fp2-f8','pk1','delta','low_power','med_speed'] 1
the historical context is abnormal 0.1
the local context is abnormal slow waves only 0.25
any significant spectral peaks exist in the delta band 1
not the largest spectral peak is attributable to more than one electrode 1
there is a significant spectral peak in an EOG channel 1
not   f3 or fp1-f7 contains isolated non repetitive waveforms 1
there is a correlation between frontal and posterior electrodes 0.5158730159

segment was classified as containing
abnormal slow waves only with a certainty of 0.4007837302.

The users response was y

14-3-1991 13:35
DATA8 SEG1

List of extracted features
= =
.

['c_feature','class_totals'] 3
['c_feature','class_totals'] 0
['f_feature','fp2-f8','pk1','sub_delta','vhigh_power','high_speed'] 0.75
['f_feature','fp2-f8','pk1','delta','vhigh_power','low_speed'] 0.25
['f   ure','fp1-f7','pk1','sub_delta','vhigh_power','high_speed'] 0.75
['f_feature','fp1-f7','pk1','delta','vhigh_power','low_speed'] 0.25
['f_feature','fp1-f7','pk2','delta','med_power','med_speed'] 1
['f_feature','fp1-f7','pk3','delta','med_power','high_speed'] 1
['f_feature','fp2-f4','pk1','sub_delta','vhigh_power','med_speed'] 1
['f_feature','fp1-f3','pk1','sub_delta','vhigh_power','med_speed'] 1
['f_feature','fp2-f8','pk1','sub_delta','vhigh_power','high_speed'] 1
['f_feature','fp2-f8','pk2','delta','med_power','med_speed'] 1
['f_feature','fp1-f7','pk1','sub_delta','vhigh_power','med_speed'] 1
['f_feature','fp1-f7','pk2','delta','med_power','med_speed'] 1
['f_feature','fp1-f7','pk3','delta','low_power','high_speed'] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 0.75
not there is only very slow activity present, less than 0.5Hz 0.25
any significant spectral peaks exist in the delta band 0.25
the largest spectral peak appears in a frontal channel 0.25
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
not fp2-f8 and fp1-f7 are symmetrical for all delta peaks 0.25
delta activity only appears in frontal channels 0.25
there are l/r symetrical delta peaks 1

delta spectral peaks exist only in the EOG 0.25
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 0.25
the waveform in channels fp2-f8 and f8-t4 are in phase 0
the largest spectral peak appears in the EOG 1

segment was classified as containing
weak em artefact with a certainty of 0.725275.

The users response was y

14-3-1991 13:38
DATA8 SEG2

List of extracted features
==

["c_feature","local","weak"] 0.725275
["c_feature","class_totals"] 4
["f_feature","fp2-f8","pk1","sub_delta","med_power","med_speed"] 1
["f_feature","fp2-f8","pk2","delta","low_power","med_speed"] 1
["f_ ure","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
["f_feature","fp2-f4","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f4","pk2","delta","low_power","med_speed"] 1
["f_feature","fp2-f4","pk3","delta","low_power","high_speed"] 0.75
["f_feature","fp1-f3","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f3","pk2","delta","low_power","med_speed"] 1
["f_feature","fp2-f8","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp2-f8","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk1","sub_delta","vhigh_power","med_speed"] 1
["f_feature","fp1-f7","pk2","delta","low_power","med_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 0.75
fp: nd fp1-f3 are symmetrical for all delta peaks 0.75
fp2-f8 and fp1-f7 are symmetrical for all delta peaks 1
delta activity only appears in frontal channels 0.75
not delta spectral peaks exist only in the EOG 1
the largest spectral peak is attributable to more than one electrode 0.75
the largest spectral peak appears in the EOG 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
not the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 1
the waveform in channels fp2-f8 and f8-t4 are in phase 0
the waveform in channels fp1-f7 and f7-t3 are in phase 0

segment was classified as containing
blink artefact with a certainty of 0.879.

The users response was n

14-3-1991 13:39
DATA8 SEG3

List of extracted features
==

["c_feature","class_totals"] 4
["f_feature","fp2-f8","pk1","delta","low_power","med_speed"] 1
["f_feature","fp2-f8","pk2","delta","low_power","med_speed"] 1
["f_feature","fp1-f7","pk1","delta","low_power","med_speed"] 1
["f_feature","fp2-f8","pk1","delta","low_power","med_speed"] 1
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
any significant spectral peaks exist 1
not there is only very slow activity present, less than 0.5Hz 1
any significant spectral peaks exist in the delta band 1
the largest spectral peak appears in a frontal channel 1
fp2-f4 and fp1-f3 are symmetrical for all delta peaks 1
fp2-f8 and fp1-f7 are symmetrical for all delta peaks 1
delta activity only appears in frontal channels 1
delta spectral peaks exist only in the EOG 1
not the largest delta peak appears in channels fp2-f4 or fp1-f3 1
not the waveform in channels fp2-f4 or fp1-f3 is slow -low delta 1
the waveform in channels fp2-f8 and f8-t4 are in phase 0
the waveform in channels fp1-f7 and f7-t3 are in phase 0

se₁   . was classified as containing
blink artefact with a certainty of 0.979.

The users response was n

14-3-1991 13:40
DATA8 SEG4

List of extracted features
===================================

["c_feature","class_totals"] 4
not the historical context is abnormal 1
not the local context is abnormal slow waves only 1
not the local context is mainly abnormal slow waves with insignificant OA 1
not the local context is detectable_OA in abnormal slow waves 1
not any significant spectral peaks exist 1

se₁   . was classified as containing
no significant activity with a certainty of 1.

The users response was y

APPENDIX T

This appendix contains the full results of the comparison made, during pre-clinical evaluation, between the IOARS and expert in the classification of the the data segments given in appendix R.

PRELIMINARY CLINICAL EVALUATION

data	seq	expert	QARS	MB	data	seq	expert	QARS	MB	data	seq	expert	QARS	MB
1	1	4	4	1	5	4	2	2	0.06	10	3	selected		
1	2	1^v	1^B	2.8	6	1	selected	DATA NOT		10	4	selected		
1	3	1^v	1^B	0.85	6	2								
1	4	1^v	1^B	0.9	6	3								
2	1	1^B	1^w	0.7	6	4								
2	2	1^B	1^w	0.7	7	1	1^w	1^B	0.98					
2	3	4	4	1	7	2	expert failed to							
2	4	1^B	1^w	0.7	7	3	evaluate							
3	1	1^v	1^B	0.75	7	4	1^B	1^B	0.89					
3	2	1^v	1^B	0.9	8	1	1^v	1^4	0.68					
3	3	1^v	1^w	0.7	8	2	1^w	1^B	0.88					
3	4	1^v	1^w	0.7	8	3	1^w	1^B	0.98					
4	1	1^w	1^2	0.98	8	4	1^v	4	1					
4	2	4	1^B	0.9	9	1	to evaluate	experts failed						
4	3	1^4	4	1	9	2								
4	4	1^4	1^B	0.9	9	3								
5	1	2	2	0.04	9	4								
5	2	4	4	1	10	1	NOT	DATA						
5	3	2	2	0.02	10	2								