WIRELESS PHYSIOLOGICAL SIGNAL ACQUISITION SYSTEM

Thomas Meggitt

Bachelor of Engineering
Electronic Engineering

Department of Electronic Engineering
Macquarie University

November 5, 2016

Academic Supervisor: Dr Oya Sevimli
Industry Supervisor: Teck Loi

ACKNOWLEDGMENTS

I would like to express my greatest gratitude to the team of Engineers and Electrophysiologists within the National Acoustics Laboratory for their supervision throughout this research project, Teck Loi, Fabrice Bardy, Barry Clinch and Humphrey Qin have devoted much of their time to help me in the progression of this thesis paper. I would also like to thank Dr Oya Sevimli for her guidance not only in this thesis but also during my time at Macquarie University.

STATEMENT OF CANDIDATE

I, Thomas Meggitt, declare that this report, submitted as part of the requirement for the award of Bachelor of Engineering in the Department of Electronic Engineering, Macquarie University, is entirely my own work unless otherwise referenced or acknowledged. This document has not been submitted for qualification or assessment in any academic institution.

Student's Name: Thomas Meggitt

Student's Signature: Thomas Meggitt (electronic)

Date: 06/11/2016

ABSTRACT

Early detection of hearing difficulties in infants is pivotal to their growth and development in later stages of life. Cortical Auditory Evoked Potentials (CAEP) testing is used to determine the hearing thresholds of non-responsive patients such as infants. In this research thesis a method to wirelessly transmit various segments of the CAEP test system is devised and a prototype is developed that will utilise off line processing. The thesis will analyse the effect of the wireless system on the accuracy of the test due to electromagnetic interference caused by the transmission of the signals. The thesis will draw upon comparisons to current leading systems to aid in the analysis of results. The thesis results show the CAEP test can be successfully conducted using a wireless system utilizing off line processing to present the results. The results also show there is a presence of electromagnetic interference in the recordings however due to the current processing the interference will only be present in the processed response if two conditions are met, firstly if the transmission frequency is less than 30Hz and secondly if the transmission is time locked to the stimulus presentation. Further research is recommended in the areas of user ability and reliability in the system.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

The discipline of engineering has allowed for the improvement in the quality of life to people world wide, giving many disadvantaged the ability to live normally within the community and operate as a successful member of society. In Australia between 0.09% and 0.12% of children are born with moderate or greater hearing loss in both ears each year [13], which if undetected and untreated early will impact their speech and language development, communication and learning [1]. As such it is vital that these cases are detected at the young infant stages of a childs development however using conventional methods of hearing testing the results can be unclear simply through observing their behaviour to stimulus sounds due to their young age.

The Cortical Auditory Evoked Potentials (CAEP) test allows audiologists to overcome this hurdle, the test presents a repeated auditory stimulus signal to the patients ear resulting in automatic responses within the patients auditory cortex, a set number of electrodes attached to the scalp of the patient record this response using the method of an electroencephalogram (EEG) [18]. The signals are enhanced and analysed to determine the hearing threshold of the patient. Once the hearing threshold is calculated the appropriate course of action can be taken whether it is to advise the implementation of a hearing aid or to adjust a currently used hearing aid.

As technological advancements are made the ability to make systems of this nature more user friendly is increased through decreasing physical size and incorporating wireless functionality. The purpose of this thesis is to determine whether the advancements of technology have progressed enough to achieve a wireless CAEP test system.

1.2 Motivations

In todays fast moving world young kids are beginning their education at younger and younger ages, from understanding that the CAEP test is able to help young infants get the best start to life as early as possible it is obvious as to why this test needs to be further developed to allow it to become more accessible and safer to more people worldwide. The three key advantages to a more user friendly and wireless system are firstly the patient is isolated from any high power equipment in compliance with the IEC 60601 technical standards, secondly to reduce power line interference and thirdly to aid the testing of difficult to test patients. By undertaking this research project it is exciting to think it may allow people to live more fulfilling and productive lives.

1.3 Synopsis

The project is to be completed with the industry partner National Acoustics Laboratories (NAL), a research division of Australian Hearing. NAL is the world leader in hearing assessment, hearing loss prevention and hearing rehabilitation, their current hearing assessment technology HEARLab is a wired system capable of performing a variety of hearing tests including the CAEP test. With help from the team at NAL this project will aim to prove the concept of creating a wireless CAEP testing system capable of delivering equal results to that demonstrated by the HEARLab system.
The proof of concept will be undertaken by two ENGG411 students completing their research theses, Thomas Meggitt the author of this thesis and Zhihao Cui with his thesis "Physiological Signal Acquisition System for Wireless Communication". The theses although separate will aim to achieve a combined end goal by investigating and undertaking individual components of the project. Each thesis will be individual however the thesis project will be made of 30% combined work with the remaining 70% to be individual, the project has been split the way it has to ensure the research area is thoroughly examined. Both theses will attempt to determine whether the CAEP test can be successfully conducted with the design and implementation of a wireless system to transmit a stimulus signal from a PC to the patient and to transmit the recorded signals from the patient to a PC. The success of the system is based upon whether it is capable of functioning to the requirements of the test and whether the use of this system will affect the test results. The combined 30% will be made of the PC software and the test systems firmware to transmit and receive the signal, once this is achieved the thesis goals will differ such that Zhihao Cui will investigate the digital signal processing and display of the recording results in real time. The goal for this section of his thesis is to prove the real time functionality of previous wired systems can be maintained with the adaptation to a wireless system. In contrast this thesis will investigate the digital signal processing and display of result using off line methods that occur after the recording has been performed. The goal of this section of the thesis is to prove the concept in a simple, fast and effective method to allow for further investigation into Electromagnetic Interference. This investigation will attempt to observe the effect of wireless transmission in the form of electromagnetic radiation caused by the system on the results and thus investigate the concept of wireless transmission for this testing use in more depth.
Other contributors to the development of the prototype include development Engineer Barry Clinch and Software Engineer Humphrey Qin in the components of hardware and

software respectively. All parts of the development of the prototype relating to this thesis including those undertaken by others have been detailed within this thesis to demonstrate a thorough understanding of each component.
Figure 1.1 aids to outline the distinction between both theses and how they work concurrently to achieve a common end goal.

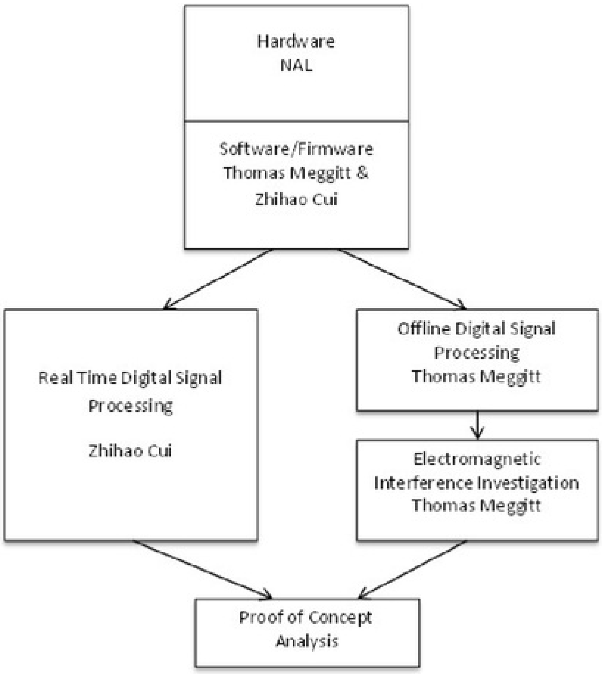


Figure 1.1: Various CAEP responses differing by age [3]

1.4 Thesis Goals

This section will outline the project goals that have been developed to determine the success of the research thesis. The goals have been split into two main categories, that off the prototype development and the electromagnetic interference investigation.

Prototype Development Goals

1. To successfully use a microcontroller to transmit physiological signals without disturbances from hardware
2. To successfully use a microcontroller to receive and present a stimulus signal to a patient

3. To identify a Wi-Fi protocol capable of transmitting signals for a reliable CAEP test

4. To develop digital signal processing to successfully display the result of the test offline

Electromagnetic Interference Goals

1. To identify if or if not electromagnetic interference is present within the recordings of a CAEP test using a wireless system.

2. To determine that the source of the interference is the test system and determine any characteristics of the interference

3. To determine if the interference is present in the final result of a CAEP test and to determine which/what processing is/could be performed to remove the interference from the final result.

1.5 Thesis Overview

This section will give a brief overview of the layout of the thesis and the content within each chapter of the thesis.

Chapter 2: Literature review

This section of the thesis will outline and analyse other relevant research material, through the researched performed this section will outline the technical requirements of the CAEP test in terms of the various electrical signals that are evoked, recorded, enhanced and displayed in order to adapt the wired CAEP test into a wireless version. It is essential to understand why the test is performed how it is so the system can successfully perform the requirements of the test.

Chapter 3: Prototype development

In this Chapter all aspects of the design and development of the prototype system will be outline in terms of all components within the system and who were the main contributors to each component.

Chapter 4: Experimental Procedure

Chapter 4 provides a detailed explanation of the test performed to determine the success or failure of the prototype system developed. This chapter also details the testing performed to detect the presence of electromagnetic interference.

Chapter 5: Results and Discussions

The processed responses of each experiment will be given in this section, the chapter will give a description of what can be seen within each of the results. After the results of each section the discussion of them will be conducted this will tie together the results of each section to provide a clear overview of what is achieved.

Chapter 6: Conclusions and Summaries

This section draws conclusion from the results and discussion present previously. It also allows a summary of the project to critically analyse the thesis and include the results found by Zhihao Cui.

Chapter 7: Future Work

This section will outline future work that may be conducted in this area to further the research performed.

Chapter 8: Abbreviations

All abbreviations used within the thesis will be stated here for reference.

1.6 Project Plan

1.6.1 Scope

The scope of the project is derived from the goals stated in Section 1.4. As a team the project stages will include the development of a wireless system prototype to operate in real time and off line, once complete both versions of the system will be tested. For my contributions to the project there are three main stages firstly software and firmware design and implementation in conjunction with Zhihao for the development of the prototype, secondly off line digital signal processing and thirdly electromagnetic interference investigation.

1.6.2 Timing Budget

To ensure the quality of work throughout the entire project a timing schedule was created so each task was allowed adequate and fair time. The Gant chart shown in Figure 1.2 was followed closely for most tasks however testing of the system and Electromagnetic Interference (EI) proved to be more time consuming than expected due to debugging the system. The project was still finished on time.

1.6.3 Financial Budget

The estimated cost of the project is equal to $83.99, fortunately the software was obtained for free from Texas Instruments or with a student license dramatically reducing the cost to just components required. Additionally the equipment obtained is used by both Zhihao and myself so the cost to each thesis is $41.99, this cost however was covered by the industry partner NAL thus the student budget of $400 was unused.

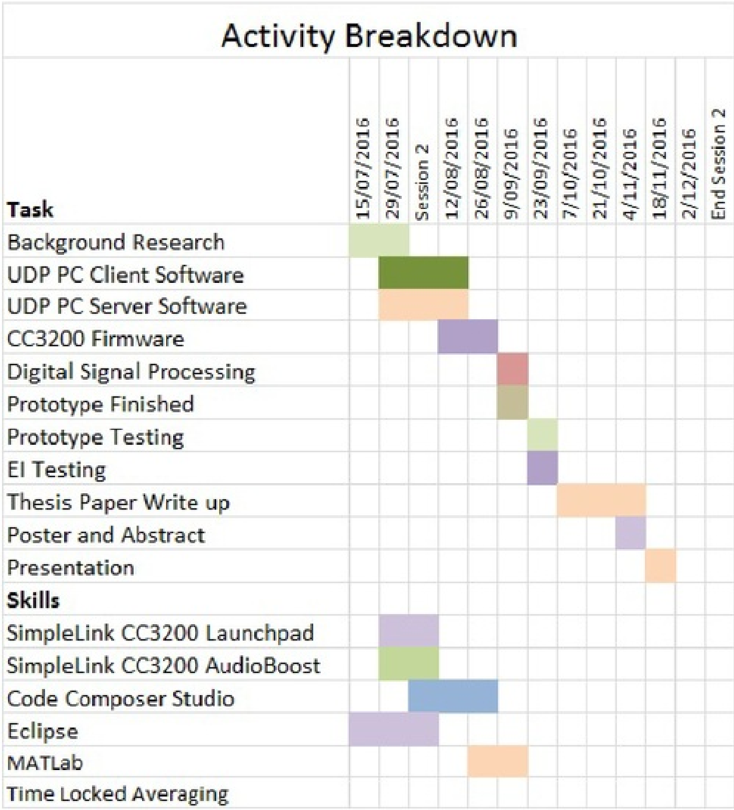


Figure 1.2: Thesis Gant Chart

Resource	Description	Supplier	Costs ($)
Code Composer Studio	Integrated Development Environment	Texas Instruments	0
Eclipse	Integrated Development Environment	Eclipse [Online]	0
MATLab	Software Environment	MathWorks	0
Router	LAN Server	DickSmith	20.00
CC3200 Module Launchpad	Microcontroller	Texas Instruments	34.99
CC3200 Audio BoosterPack	Amplifier and CODEC	Texas Instruments	29.00
EEG Test Equipment	EEG test analogue signals	NAL	0
Pre amplification Board	Amplifying IC	NAL	0
Total	-	-	83.99

Table 1.1: Financial Budget.

Chapter 2

Background and Related Work

2.1 Cortical Auditory Evoked Potentials Testing

2.1.1 Background Theory

The Cortical Auditory Evoked Potentials test or CAEP test is used to observe the brains involuntary response to an auditorily evoked stimulus allowing audiologists to reliably estimate the hearing threshold of patients unable to provide behavioural feedback. An auditory stimuli is presented to the patients ear creating a neural response within the cochlear which travels through the auditory brainstem to the auditory cortex [20].
Due to the weakness of the target signal within the brains ongoing activities, enhancements are required to the observed potential to determine if a response has been triggered, these enhancements include filtering, amplification and averaging, and will be covered later on. Once enhanced the morphology of the response will give a P1-N1-P2 complex, shown in Figure 2.1 [3], that is highly variable depending on the age of the patient. An adult response will give a clear P1 at 50-70ms after the stimulus is evoked, N1 at 100-120ms and P2 at 170-190ms whereas an infant response is characterized by a large P1 followed with a broad N1 between 200-250ms.

The CAEP test is preferred over other similar tests such as the Auditory Steady State Response (ASSR) and Auditory Brainstem Response (ABR) firstly as the stimuli can be generated to be associated with speech recognition and detection and secondly as it allows for an assessment on the entire auditory network up to the cortex [18].

2.1.2 Stimuli

There are various types of stimuli tones used to evoke the potentials within the auditory cortex, [8] uses stimuli aimed at imitating common speech sounds: /m/, /g/ and /t/ with a duration of 30, 21 and 30 ms respectively centred around three frequencies of 250Hz, 1250HZ and 3250Hz and sampled at a rate of 44.1kHz. Due to an averaging processes required to be detailed in Section 2.1.4 the stimulus is to be repeated multiple times to

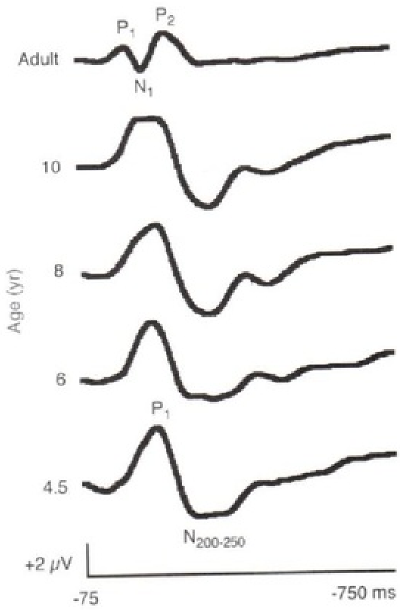


Figure 2.1: Various CAEP responses differing by age [3]

a patient, in some cases up to 900 repetitions in one assessment sitting. In these cases the stimulus sound is changed each time it is evoked to present a different tone from the previous, this is to keep the cortex engaged in the assessment. Each stimulus tone usually consists of a simple Sine waveform starting with a gradual increase in amplitude or a 'ramp'.

Figure 2.2 [20] shows the latency and amplitude of early, middle and late latency responses caused by three different auditorily evoked potentials tests, CAEP is considered a late latency response as the response to be measured is triggered in or near the auditory cortex requiring a lengthy period of 50ms-500ms to be completely observed after the onset of the stimulus [20]. During testing the inter stimulus time is required to be higher than the latency period to allow for the response to be observed uninterrupted, for CAEP testing [8] use a inter stimulus time of between one and two seconds and can be repeated up 120 times per test sequence depending on the quality of the measurements taken [2]. The test program will regulate the sound level of the stimulus being presented to the patient. The sound level in decibels will start at a selected point and gradually decrease by set amounts when an evoked potential has been identified by the program indicating the patient is capable of hearing the stimulus at its current level [2].

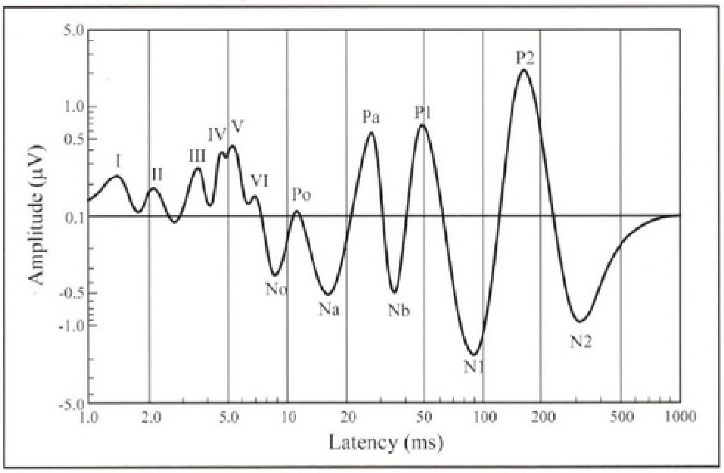


Figure 2.2: Auditory EPs plotted using logarithmic axiss [20].

2.1.3 Response Recording

The response to the stimulus is processed and hence measured in or near the auditory cortex through the method of an Electroencephalogram or EEG. The EEG uses a minimum of three electrodes, one on the forehead as the ground, one on the top of the scalp as the active and one on the mastoid (behind the ears) as the reference. This is considered single channel measuring as only the response to one auditory cortex it observed [2]. The EEG works by observing the voltage potentials at the active and reference electrodes and determining the potential difference between these two points. Prior to CAEP testing impedance testing is required to asses the impedance between the electrode and the subcutaneous tissue. The test is conducted by applying a small current across the electrodes, a result of less than 10k is recommended [20].

2.1.4 Response Processing

To prepare the acquired signals for analysis several phases of amplification and filtering are required. This is done both by hardware immediately after the analogue signal is captured by the electrodes, and by software after the signals have been digitalised [4].
As shown in Figure 2.2 [20] auditory evoked potential responses are in the magnitude of a few nanovolts to microvolts dependant on the latency of the response, with CAEP responses typically between 1-5uV [20]. Firstly the signals must be amplified to a factor of between 1000 and 1,000,000 in order to be evaluated by digital processing. Secondly the signals are filtered to attenuate obscuring electrical activity within the signals, [20] states most activity within late latency responses occur within 5Hz and thus filtering should exclude signals on either side of this, parameters of 0.15Hz and 15Hz are suggested. Thirdly

an analogue to digital conversion is performed to allow analysis by the computer, consideration is given to the rate of the conversion to ensure the frequencies present are reproduced in the recorded response and also the amplitude resolution expressed in bits. Paper [7] proposes a method to observe the hearing threshold of adults through CAEP testing. An amplification factor of 1210 is used once the signal is captured by the electrodes. This amplification is created by three amplifiers, two of 1:10 and one of 1:9 prior to being high-pass filtered at 0.16Hz with an analogue first order filter. A digital signal is acquired from the analogue at a 16kHz sample rate and 16 bit Pulse-Code Modulation (PCM) to then be down sampled to 1kHz which is low pass filtered online at 30Hz with an order of 128 and a zero time delay. The signal enhancement performed in this thesis correlate closely with that suggested in [20] with only a slight difference in the low pass filter size.
Due to the stimulus being evoked on a one to two second time cycle the continuous response signal will contain 500ms of desired response and upto 1500ms of undesired noise. Therefore to allow for further processing and to be efficient segments of the signal are cut into epochs containing the section of 200ms prior to stimulus onset and 600ms post stimulus onset [7].
Each epoch is subjected to baseline correction and artefact rejection. Baseline correction adjusts the reference point of each response epoch based on its value at 100ms prior to stimulus onset. Artefact rejection omits response epochs that record values of over 150uV at any point during the epoch suggesting an excessive noise event occurred during this time, possibly caused by eye movement [7].

2.1.5 Noise Reduction

Each epoch gathered contains both the desired CAEP response in the magnitude of 1-5 uV and random undesired noise from background activity within the brain to the magnitude of 10-30uV [3]. In order to successfully observe the CAEP responses an averaging process is used to decrease random signals whilst un-altering constant signals. Due to the differing range in background noise in patients a fixed number of epochs to be averaged cannot be given to guarantee an acceptable quality or signal to noise ratio, SNR [10].
[3] States that the rate at which noise diminishes through averaging is not linear but equal to 1/SQRT(N) where N is equal to the number of epochs averaged. [10] suggests this presumes three conditions: the CAEP response is deterministic, the background noise is stationary and the CAEP and background noise are independent of one another. [10] found that the success of this method is only the mean trend and that continued averaging can result in deterioration of the response signal without obvious changes in noise. This is especially apparent with background noise containing low frequency components. Figure 2.3 [3] demonstrates the relationship between noise and the number of epochs averaged [3].

A study on CAEP testing on 22 infants is presented in [6] and demonstrates the effectiveness of this method in reducing noise. The paper shows the mean RMS amplitude of the noise in the waveform, initially 27.6 $\pm$ 2.6 uV, drops by the square root of the

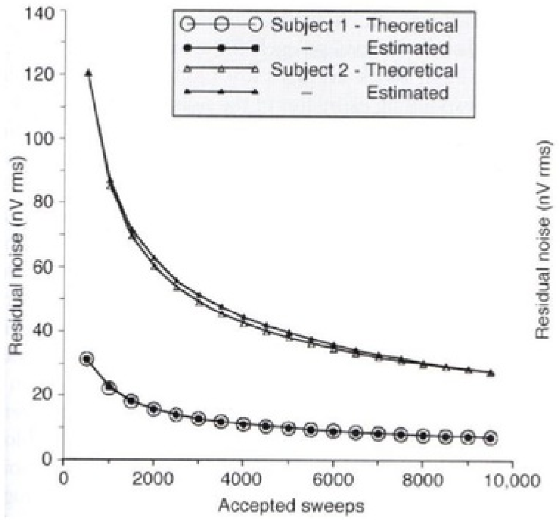


Figure 2.3: Relationship of noise against number of epochs averaged [3].

number of epochs recorded. With 200 epochs the amplitude of the noise is equal to 1.95 $\pm$ 0.18uV. Similarly the paper [7] depicts the mean RMS noise amplitude from 1,227 recordings to be 12.1uV as shown in Figure 2.4 [7]. Using the averaging method above for 120 epochs the noise amplitude was reduced to 1.93uV allowing CAEP responses of 2uV or larger to be observed.
To apply the averaging technique to the response a portion of each epoch is sampled into 9 bins. The start point of the sampled bins and width of the bins are subject to the age of the patient and thus the shape of the response waveform, [7] uses 9 bins from 51 to 347ms with a width of 33ms. The averaging technique is then applied to each respective bin of all collected epochs. This method has the added advantage of allowing for the objective response detection through the use of Hotellings T2 test [12].

2.1.6 Current Leading Assesment Device

An example of a current CAEP testing system is the HEARLab developed by National Acoustics Laboratories (NAL), the system is commercially available and is distributed to many countries around the world. The HEARLab has been developed such to work as a general assessment utility potentially allowing for a range of assessments to be perform with the device depending on the software modules. The initial software module allowed for the assessment of two varieties of testing to be performed such as Aided Cortical Assessment (ACA), such as CAEP testing, and Cortical Threshold Estimation (CTE). https://hearlab.nal.gov.au/ The HEARLab system is comprised of two units, the electrode processor and stimulus controller. The electrode processor has a gain signal gain factor of

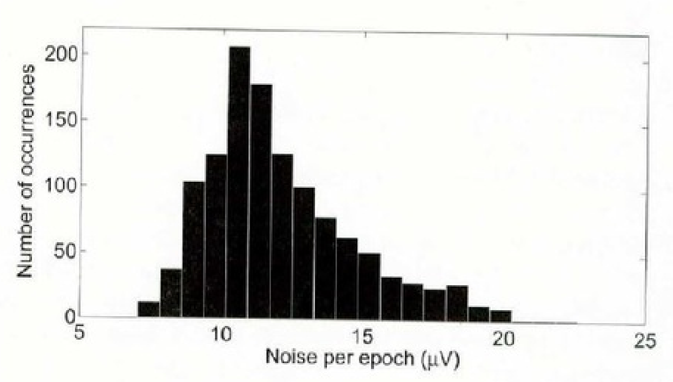


Figure 2.4: Histogram of EEG noise RMS amplitudes of 1,227 available recordings [7].

10 and allows for the physical connection of up to five electrodes each having a separate gain factor of 121. The stimulus controller as suggested controls the stimulus displayed to the patient. The system appears to be very simple allowing for an intuitive use for audiologist, the ability to modify the existing system with software module updates is a great feature, however it has a large physical size and requires a wired connection to a PC. https://hearlab.nal.gov.au/ The most recent and advanced software module to be used in conjunction with the HEARLab system is CATE, or Cortical Automatic Threshold Estimation, as the name suggests this module allows the system to automatically estimate the cortical threshold of the patient in testing without any input from an audiologist besides applying the electrodes correctly and pressing the start button. The CATE software can be used for multiple response test including CAEP and ABR. The software uses Hotellings T2 statistical analysis to determine the presence of a CAEP response during a real time recording allowing the system to commencing testing for the next threshold by lowering the stimulus level [2].

2.2 Transmission

2.2.1 Transmission Protocols

A transport protocol is required to provide the ability to interact from a source to a destination without a wired connection. Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) are two such transport protocols that are widely used for multimedia transfer. [22] compares the performance characteristics of three main transport protocols TCP, UDP and TCP Friendly Rate Control (TFRC) through multimedia transfer to identify their limitations. Separate simulations are conducted for each protocol in a static wireless network consisting of two source nodes, five bottleneck nodes and two destination nodes. Both TCP protocols have a packet sized fixed at 1000 bytes with the

UDP packet size fixed at 210 bytes. The paper finds UDP is capable of sending data at a constant rate regardless of the congestion within the network, however UDP suffers from a high number of packets dropped. TCP reacts to a high congestion by lowering the data transfer rate in comparison to UDP but manages to drop far fewer packets [22]. From the results it can be concluded that UDP is useful when a constant data rate is more important over a low packet drop rate, this will be the case for CAEP testing.

2.2.2 Client/Server Relationship

The client server relationship is used to describe the transfer and display of information from a source to a display application used by the user. The client starts a connection with the server over a local area network (LAN) or wide area network (WAN) and requests a service or resource from the server, once the server fore-fills the clients request by transferring the desired information the connection is terminated [23].

In the Java programming language the client server relationship is established through the use of sockets, one end point of a two way communication channel between two system on a network. The client will attempt to rendezvous with the server at the servers specified IP address and port number, upon a successful connection the server will create a new port to continue the communication so it can continue listening for further connection requests [28].

The information sent using the UDP protocol is sent as a package of data called a datagram, a self-contained, independent message which follows the characteristics of the UDP protocol in that its delivery is not always guaranteed. To implement this in Java firstly a socket is created in the client to create a place to send the datagrams from, secondly datagrams a created from the desired information stream and thirdly the datagrams are sent to the socket and thus sent to the selected server. The following code sends packets of information of the file one-liners.txt from the client port 4445 to the server IP address "127.0.0.1" and port number 9999 [29].

```
socket = new DatagramSocket(4445);

    try {
        in = new BufferedReader(new FileReader("one-liners.txt"));
packet = new DatagramPacket(buf, buf.length, 127.0.0.1, 9999);
socket.send(packet);
}
    catch (FileNotFoundException e){
        System.err.println("Couldn't open quote file.  Serving time instead.");
    }
}
```

The server side is very similar to the client, it will open a socket and write received data into a datagram packet which can then be used to write to a media stream or array for further use [29].

```
socket = new DatagramSocket(9999);
    try {
        packet = new DatagramPacket(buf, buf.length);
socket.receive(packet);
String received = new String(packet.getData(), 0, packet.getLength());
System.out.println("Quote of the Moment: " + received);
}
    catch (FileNotFoundException e){
        System.err.println("Couldn't open quote file.");
    }
}
```

2.2.3 Audio Playback

Reading a file from a source location onto a program requires several steps and interfaces within Java. The audioInputStream class allows the program to obtain an external audio file and specify the audio format and length of that file. The file can then be easily accessed by other Java classes once contained within the stream.
The interface of dataLine can be used to implement media functionality such as audio within a program by drawing information from the audioInputStream. The dataLine interface includes methods such as start, stop, drain and flush to control the data within the line. There are two types of data lines, the targetDataLine to allow a program to read data from a dataline and a sourceDataLine which allow a program to write to a dataLine [30]. The sourceDataLine can be used to contain a file of data which can be packaged and sent as a datagram, the code below demonstrates an audio file being read onto an inputStream which is read onto the sourceDataLine [5].

```
File audioFile = new File(audioFilePath);
AudioInputStream audioStream = AudioSystem.getAudioInputStream(audioFile);
AudioFormat format = audioStream.getFormat();
DataLine.Info info = new DataLine.Info(SourceDataLine.class, format);
SourceDataLine audioLine = (SourceDataLine) AudioSystem.getLine(info);
```

The ensure the continuity of playback or display of data as it is received through a socket and sent to its required method a buffer can be used, the buffer allows data to be stored to a temporary memory before it is read back out again only when enough data has been accumulated to ensure continuous playback [26]. The buffer can be applied to the client and server sides to ensure data is sent, received and used at an optimal rate [27].

2.3 Electromagnetic Interference

When performing an auditory evoked potentials test many potentials can be recorded by the electrodes that have not been generated by the auditory nervous system, these potentials are called artefacts [20]. Artefacts are classified as physiological when they are generated within the subject or physical when they are generated in the surrounding environment and also whether they are stimuli related or not.

Performing the CAEP test on Cochlear Implant (CI) patients has been proven to be problematic due to the electrical artefacts on the scalp created during auditory stimulation interfering with the CAEP identification [11]. Figure 2.5 [11] demonstrates the CAEP test on two children, one normal hearing and one fitted with a CI. An expected response is demonstrated for the normal hearing child with a P1 at 100ms post stimulus onset and long N2 at 230ms post onset. However as is seen for the child fitted with a CI a large magnitude pedestal obscures the response from marginally after the onset to marginally after the offset. [24] observed that the CAEP tests of 12% of CI fitted children were obscured in the first 100ms after onset, suggesting the obscurity is caused by the creation of the stimulus within the CI. It was determined this obscurity occurred for CIs from all manufactures.
Two methods of minimising this artefact obscurity within the CAEP response are analysed in [11]. The methods include Independent Component Analysis (ICA) as a pre-processing filter and using an optimised differential reference. The study tests 5 children with a mean age of 10.5 who have all been fitted with a CI. The ICA is applied to the contaminated EEG file to identify multiple components that can be attributed to the implanted devices, these identified unwanted signals are removed from the file and the EEG can be reprocessed according to the CAEP test. This method was successful in minimising the effect of the artefact with at least two components found in all five cases that could be attributed to the stimulus artefact. The paper suggested for this to be a viable method a large number of recording electrodes would be required, which is unrealistic, therefore rendering this an inviable method. The second minimisation method makes use of a reference electrode to allow the active electrode to be differentially recorded relative to this reference. This method does manage to minimise the artefact impact however similar to the first method the impact is minimal.

The effect of Wi-Fi signals on brain operation is demonstrated in [19] paying close attention to P300 event related potentials component. An EEG was twice performed on 30 participants within which one of the two tests was performed in the presence an electromagnetic field created by a Wi-Fi access point operating with a frequency of 2.4GHz. The EEG used 30 electrodes placed on the scalp to observe an occurring potential related to a stimulus event, each signal was subject to enhancement similar to the CAEP testing reviewed earlier. The access point was located 1.5m from the participant resulting in a field strength of 0.49 V/m at the participants head. The study found the amplitudes of the recorded responses were affected by the presence or absence of the Wi-Fi signals however it found that the response was different for male and female participants. The mean amplitudes for males was found to be suppressed from 7 uV to 2 uV when the EMF was presented, where as for females the mean response was enhanced from 4 uV to 8 uV when the EMF was presented. The paper provides no explanation for these findings.

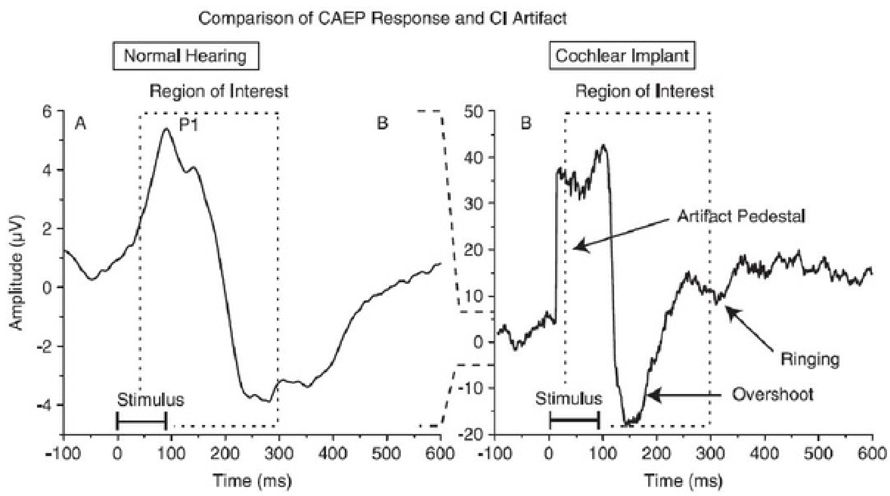


Figure 2.5: (A) Normal CAEP response from normal hearing patient (B) CAEP response from Cochlear Implant patient [11].

Chapter 3

Prototype Development

In this Chapter the development of the prototype system will be detailed, it will consist of the development of each major component of the system and outline how each individual section will form a whole.
To understand the importance of the prototype we must revisit the goals of this thesis project:

- To successfully use a microcontroller to transmit physiological signals without disturbances from hardware

- To successfully use a microcontroller to receive and present a stimulus signal to a patient

- To develop digital signal processing to successfully display the result of the test offline

To achieve these goals one must obtain data that can successfully prove the operation of such a system in the desired way, hence the development of a prototype is required.
As mentioned previously in section 1 the development of the prototype will be conducted within a team, the majority of the work will be divided between myself and fellow ENGG411 student Zhihao Cui, to meet the requirement of 70% individual work it was decided to complete several aspects of the prototype development together to make up the combined 30% as these common components were required in both thesis projects.
In addition to Zhihao the development team will consist of several engineers from NAL including Development Engineer Barry Clinch and Software Engineer Humphrey Qin.
This chapter will detail the development of all components of the prototype essential to this thesis project, which will include the work performed by others with each section outlining the major contributors.

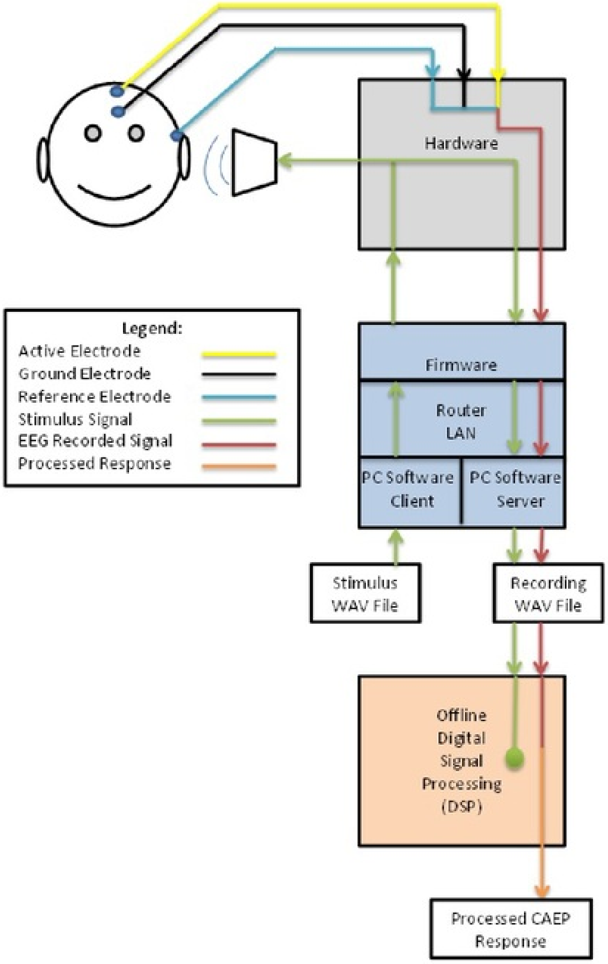


Figure 3.1: Block Diagram of Prototype.

The prototype is to be developed in three distinct stages consisting of the software and firmware for the PC and CC3200 respectively, the hardware and the digital signal processing (DSP). A black box diagram can be seen in Figure 3.1 of the prototype setup, the figure shows each major component of the system including their minimum inputs and outputs to make the system operational. Research in Section 2.1.3 [2] suggested the minimum number of electrodes for an EEG to record from are three such that one will be used to ground the system and two will be used to make a differential recording, therefore the EEG recording requires a minimum of one channel. We also found due to time locked averaging a second channel is required to contain the stimulus [3], thus the system will be based around the transmission of two signal channels, the EEG channel and the Stimulus channel shown in red and green respectively in Figure 3.1.

The hardware will consist of 4 inputs that being three electrodes to be recorded from the patients scalp and one stimulus channel from the firmware of the CC3200. The hardware will then consist of three outputs two of which are made up of the recorded channels, these will be made up of one recorded EEG signal plus the stimulus channel routed through the hardware to act as the trigger for the time locked averaging. The third output for the hardware is the auditory stimulus out to trigger the response within the patients ear, this will be through a speaker.
The software and firmware for the system will be split again into three sections consisting of the software client, software server and firmware client/server. The software client will allow the transmission of two audio signals from the PC to the firmware of the CC3200, the input of the client will be a WAV file containing the stimulus file to be transmitted. The software server will work oppositely to the client and receive two audio signals from the CC3200 firmware with an output of two audio signals in the form of a single WAV file. The CC3200 firmware will be the connection between the PC software and the hardware with three inputs and three outputs. After receiving the digital audio stimulus signal from the software client the firmware will output a single analogue stimulus signal to the hardware to be used as required. The firmware will then receive two analogue signals for recording from the hardware and transmit both of them to the software server. The communication between the firmware and software will take place on a Local Area Network, LAN, established by a router.
The Digital signal processing will use only two inputs to provide one output, the two inputs shall consist of the two signals saved as the single WAV file by the software server. The two signals being the EEG channel and the time locked stimulus channel shall be used to perform the DSP. Once completed the DSP will output a single processed epoch of the recorded channel with a length of 700ms long

3.1 Software/Firmware

The development of the software and firmware will consist of three stages including the PC client, PC server and the firmware within the hardware. This section is to be completed by both Zhihao and myself to account for the 30% combined work within the project. Both PC programmes will be written in Java language on the Integrated Development Environment Eclipse [9], Java was chosen as it was the language Zhiahao and myself have the most familiarity with. The firmware will be written in C in the IDE Code Composer Studio (CCS) [16]. This is the language used by most of the example applications created by Texas Instruments.

3.1.1 PC Client

The primary objective of the PC software is to act as a client and server to allow for transmission and receiving of signals. After analysis of the transport protocols it was found that for this application the UDP would provide the most adequate system due to its characteristic of transporting data at a constant data rate with its limitation of occasional data loss being considered acceptable. An example client and server code using UDP for Java was sourced from [14], the code was capable of transmitting from a microphone of a PC running the client application to the speakers of a PC running the server application.

The client application needed to be changed in several ways to allow the desired functionality, firstly the client needed to transmit a saved WAV file, secondly the playback of this file needed to be seamless and thirdly the user needed control to stop and start the transmission as required.

Opening WAV file

An inputstream was used to open the WAV file in the program and allow the data to be written to useable arrays such as buffers, along with this a source data line was opened to also contain the WAV data.

```
String strFilename = "Lorde.wav";
soundFile = new File(strFilename);
InputStream = AudioSystem.getAudioInputStream(soundFile);
adFormat = InputStream.getFormat();
DataLine.Info dataLineInfo = new DataLine.Info(SourceDataLine.class, adFormat);
sourceLine = (SourceDataLine) AudioSystem.getLine(dataLineInfo);
```

The sourceDataLine was not essential in this application as datagram packets to be sent would be taken from the inputStream through a buffer however it was found that the packets were being created and sent without speed regulation such that a sound file of 4 minutes could be sent in less than 7 seconds. As the stimulus file was to be played to the patient and recorded back through the system for processing a real time feed was essential, to solve this issue information from the buffer was written to the sourceDataLine to allow the Java class to slow the creation of packets to the correct tempo as it attempted to play back the audio through the PC speakers however muting is used to stop the actual playback. Initially thread.sleep() was used to slow the program but this created multiple pauses and was considered less than satisfactory for an EEG recording.

UDP Transmission Socket

Once the data was written to the inputstream a while loop was used to continuously write small chunks of data to a buffer array of size 1024*16 bits, each chunk was converted to a packet containing the IP address and port number of the reciprocal server and send to the

UDP client socket to be sent. The while loop was controlled by a simple GUI containing two buttons to start the transmission and stop the transmission.

```
while (!stopaudioCapture) {
int cnt = InputStream.read(tempBuffer, 0, tempBuffer.length);

if (cnt > 0) {
sourceLine.write(tempBuffer, 0, cnt);
DatagramPacket sendPacket = new DatagramPacket(tempBuffer,
 tempBuffer.length, IPAddress, 5050);
clientSocket.send(sendPacket);
byteOutputStream.write(tempBuffer, 0, cnt);
}}
```

Once run in Eclipse the client program provides the user a simple GUI to start and stop the stimulus transmission.

3.1.2 PC Server

The server is used by the PC to receive and play the audio data transmitted by the CC3200. The server consists of two main components to make up the multiple functions of the program, firstly a function is used to receive UDP packets via a socket from the CC3200 client and allow those packets to be read elsewhere, secondly a class is used to queue the packets received by the socket and thread them to a buffer to be played as required. NAL Software Engineer Humphrey Qin was a large contributor in the development of the server program, Humphrey aided the development with the implementation of the queuing buffer which is to be detailed below.

UDP Receive Socket

To receive the data the server program uses a while loop to receive datagram packets and operates continuously with no conditions to pause or stop. Each datagram packet is of size 40960 bytes, normally the size of the packet being received is critical to the continuous playback of the audio data as to small a packet and the player will run dry resulting in multiple stops and too large a packet and the player will be flooded resulting in jumbled playback, however in this instance each packet received is sent to a queue function outside of the UDP socket to be stored and played back when instructed by other functions.

Queue and Play

A Class is used to store the queued datagram packets and playback the audio concurrently with the operation of the UDP socket function. The Class opens a sourcedataline which as mentioned earlier is used to allow the playback of audio data through the PC speakers.

A buffer class is then used to write the data from the queue to the sourcedataline, the buffer regulates the rate at which data is transferred from the queue to the sourcedataline by only allowing the transfer when there is space on the sourcedataline and when there is information within the queue. This buffer regulation allows for a constant flow with zero data lost.

Operation

Once the server is complete the program can be run inside eclipse, if there is a signal being transmitted to the PCs IP address and to the port designated by the server program the program will begin to receive the data and play it through the PCs audio system. To save the received data as a WAV file a separate recording program called Audacity was used, the program allows the user to record and observe any audio being played through a PCs audio system. The use of this method to save the received data was utilised to ensure reliability of each recording taken, it was often found that recordings failed due to hardware reasons and as such it was decided to simplify the software to remove as much potential for error as possible.

3.1.3 Firmware

The primary objective of the microcontroller firmware is to act as a client and server to allow for signals to be received and transmitted from and to the PC. The microcontroller must also convert the digital signals to analogue to allow playback of the received signals and convert analogue to digital to allow the transmission of the recorded CAEP signals. Texas Instruments have supplied many sample applications one of which allows for the Bi-directional audio streaming and playback when used in conjunction with the Module LaunchPad and Audio BoosterPack package. The application transmits audio sourced from the built in microphone or the Line IN input from one CC3200 package using the UDP, a second CC3200 package is used to receive and playback the data via the Line-Out [15]. The application is especially useful in this application as it transmits two audio channels in both directions and manages the operation of the CODEC to convert D to A and visa versa.

Wi-Fi Application Modifications

Several source and header files of the Wi-Fi Audio App were changed to allow the connection to the PC client. The Common header was changed to incorporate the new wifi settings and the port in Network header was changed so the app could be found by the client on the PC. The port and IP address to search for the PC server was set to a standard value, this meant each time the system was to be used the IP address of the PC being used would need to be changed to match that set.
Small modifications were made to the functionality of the application, firstly the multicast technique allowing the firmware to transmit to multiple destinations was disabled to

allow only the unicast technique, this was performed simply in the network and microphone source files. Secondly it was found the stimulus was not presented with enough power thus the CODEC functionality was modified to increase the output power for this channel, this modification was performed in the audio codec source file.
Once the Wi-Fi Audio Application was modified a software tool was used to flash the program onto the CC3200 device. Flashing allows the user to write a program to a microcontroller such that the program cannot be removed or modified when power is disconnected from the device. The software tool used was the CCSUniFlash [17], a program designed specifically for use with Texas Instrument devices such as the CC3200.

3.2 Pre amplification hardware and interface

The pre amplification hardware and interface was comprised of five main components, amplifying electrodes, amplification board, CC3200 interface PCB, the CC3200 microcontroller and mounting structure. The amplifying electrodes and amplification board were adopted from the existing NAL testing system HEARLab, these components were interfaced to the CC3200 through the use of a specifically designed PCB with the entire hardware system supported by a mounting structure. The mounting structure was constructed using bolts to hold the frame together with a Perspex cover for protection. A diagram for the hardware system is given in Figure 3.2.

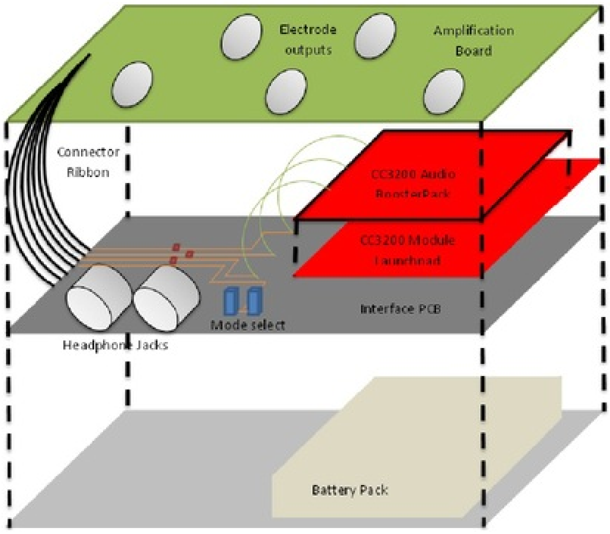


Figure 3.2: Diagram of Hardware.

3.2.1 Micro-controller Selection

The microcontroller to be used for the wireless receiving and transmission of test signals is the SimpleLink CC3200 Module Launchpad and Audio BoosterPack developed by Texas Instruments. The CC3200 was chosen for a number of reasons, firstly it is one of the only microcontrollers to use the Wi-Fi protocol for wireless transmission at the chip level. Secondly there are many extra packages to be added to the base unit to allow for extra capabilities, in this case the Audio BoosterPack is added to allow for a class D power amplifier, audio codec for D to A and A to D conversion of two channels and two headphone jacks to serve as inputs and outputs. Thirdly there is a large range of example material supplied for free by Texas Instruments to aid in the development of projects.
Using the CC3200 microcontroller for the wireless functionality of the system a wireless local area network is required to achieve communication between the microcontroller and PC. For this system a wireless router was purchased to provide the network, the router will provide a wireless server from which the microcontroller and PC can interact with each other as required. The microcontrollers used are shown in Figure 3.3 to demonstrate how they connect physically.

Figure 3.3: CC3200 Module Launchpad and Audio BoosterPack.

3.2.2 Electrodes and Pre-Amplification

The pre amplification hardware and interface unit operates in two modes, impedance testing and acquisition, which is selected using the high and low toggle on the interface

PCB, low for impedance testing and high for acquisition. The impedance testing mode allows the user to monitor the level of impedance caused by the electrode to skin and skin to electrode contact between an operating pair of electrodes. A known voltage is delivered to the Ground electrode which is observed using the active and reference electrodes, this observed voltage is used to calculate the impedance within each channel and thus the skin contact impedance.

Amplifing Electrodes

The electrodes used within the system are considered amplifying as they give amplification to the signal in the head of the electrode, this design is unique to the NAL HEARLab system as most electrodes merely receive the signal with no amplification. The active electrode consists of two 1:10 amplifiers giving a total amplification factor of 121. During acquisition mode the reference channel is sent back to the active electrode to create a differential amplification between the two channels. During impedance testing mode the active electrode uses an attenuator to reduce the signal so to not saturate the analogue to digital codec with the testing voltage after it has been amplified. A circuit diagram for the Active electrode can be seen in Appendix C.2

Pre-Amplification Board

The pre-amplification board receives and relays signals from both the electrodes and the interface PCB. Its primary acts are to amplify the active signal further by a factor of 10 and direct the signals based on the mode the system is operating in. An impedance testing line connects two main switches, a switch allows or blocks the stimulus signal to be fed back in the ground electrode and work as the impedance mode. A second switch allows or blocks the reference channel to feed back into the active electrode and operate in acquisition mode. The circuit Diagram for the pre-amplification board can be seen in Appendix C.3

3.2.3 Interface PCB

The interface PCB was designed and constructed by Barry Clinch and allows communication between the amplification board, the CC3200 and the headphone output lines. The PCB connects headphone jack J3, headphone jack J4 and pin 10 on the CC3200 AudioBoost to the recording signal lines, the stimulus signal line and the mode selection line respectively. The stimulus signal line uses an amplifier with a gain of 1 to act as a buffer to ensure the same voltage is held under high load. The PCB interface also connects the amplification board to the CC3200s power supply, a regulator created by a series of parallel capacitors is used to reduce the noise of the power supply. The speakers outputs of the AudioBoost, right and left, are connected to two accessible headphone jacks for display to the patient. The complete system is shown in Figure 3.4 and the circuit diagram for the Interface PCB can be seen in Appendix C.4.

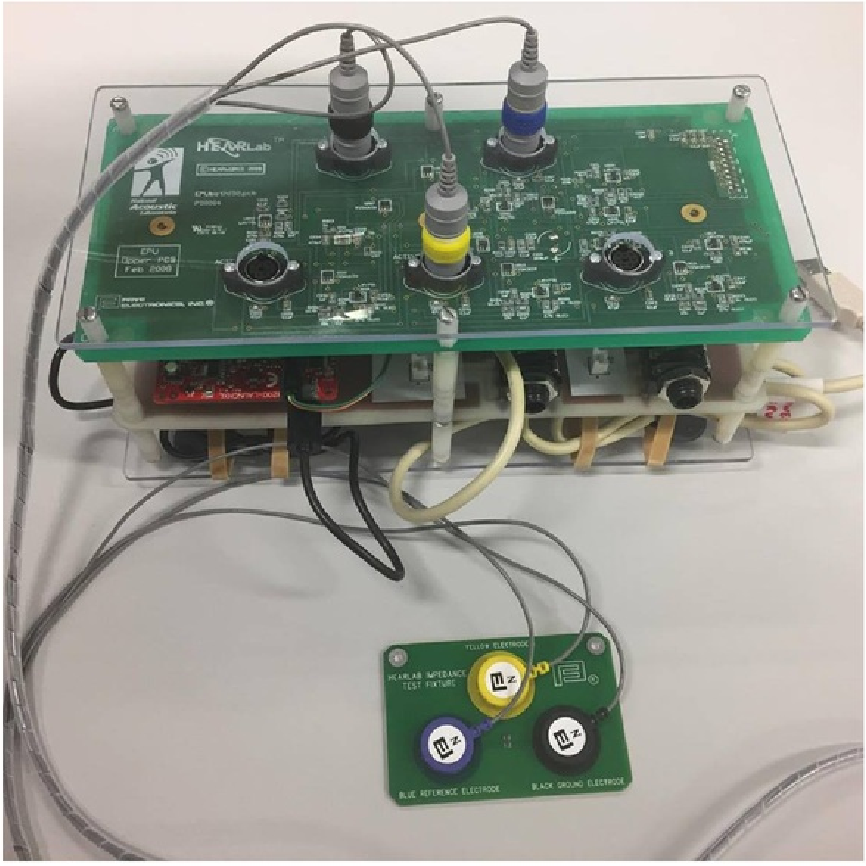


Figure 3.4: Completed hardware system.

3.2.4 Hardware Testing

Once the hardware was complete simple tests were performed on the system to confirm its operation, these tests include transmitting a simple stimuli to the hardware and recording the output in acquisition mode. For this test the electrodes were connected to a testing pad which supplied 20 ohms of resistance to the signals to imitate connection to a patient. Through the tests a series of issues were found which were required to be rectified to allow for an accurate functionality of the system, these issues were identified and solved by a team including NAL Engineers Barry Clinch and Teck Loi and also Zhihao Cui and myself.

Crosstalk

The initial recording of the system with the resistor pad showed a large sine wave pulse in the EEG recording channel each time the stimulus pulse is triggered, the sine wave pulse has the same frequency and is time locked with the stimulus pulse so it is believed this is cross talk between the two recorded lines. Figure 3.6 displays a processed epoch in which the cross talk can be dramatically seen from 0s to 0.05s, this shows that since it is a regular time locked occurrence it cannot be removed by the digital signal processing. Thus several hardware solutions were implemented on the system in an attempt to remove the issue.

Firstly the two recording lines running from the amplification board through the interface PCB to the CC3200 were replaced by insulated wires, it was proposed that a voltage was being induced through the parallel running wires on the interface PCB when a large signal such as the stimuli pulse was transmitted. This was shown to have minimal impact on the cross talk issue.

Secondly it was proposed the single battery pack which drove the entire system through the CC3200 was driving the stimulus through the system in to the amplification board and forcing cross talk between the two lines. A second battery pack was added to power the amplification board and the electrodes separately from the CC3200 and the interface PCB. Again this was shown to have little to no impact on the cross talk effect.

Finally a series of test were performed with which the output from the CC3200 to display the stimuli to a patient was modified in several ways. Initially the system used a non-amplified speaker connected to an amplified output of the CC3200, it was discovered that by detaching the speaker from the cables connected to the output the cross talk was significantly reduced as shown in Figure 3.7.
Furthermore by removing the cable connected to the output the cross talk was eliminated completely. Consulting the CC3200 schematic it was discovered a class D amplifier was used to amplify the signal, it is believed the switching stage of the Class-D amplifier was causing capacitive coupling and allowing the stimulus to enter the recorded signal. To ensure complete functionality of the system an amplified speaker was connected to the non-amplified headphone output of the CC3200 which did not use a Class-D amplifier so a stimuli could be presented to the patient without the cross talk. This system produced zero cross talk when used with the resistor pad as an artificial patient.

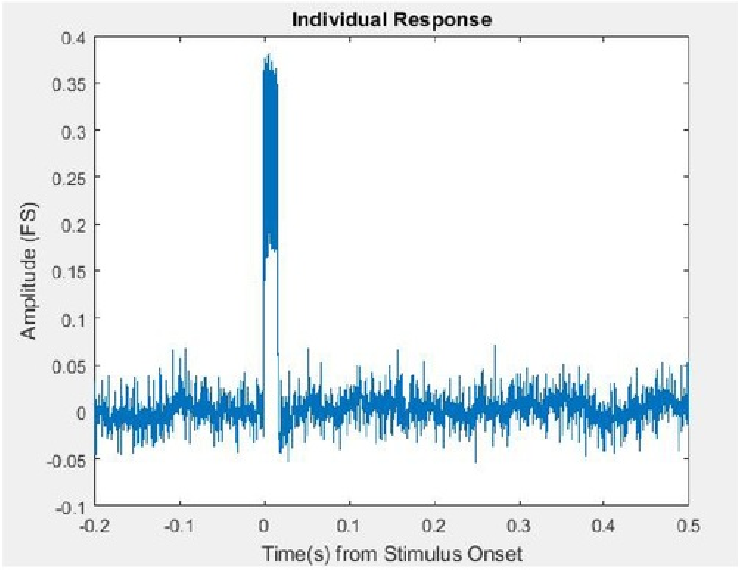


Figure 3.5: Individual Epoch with Crosstalk.

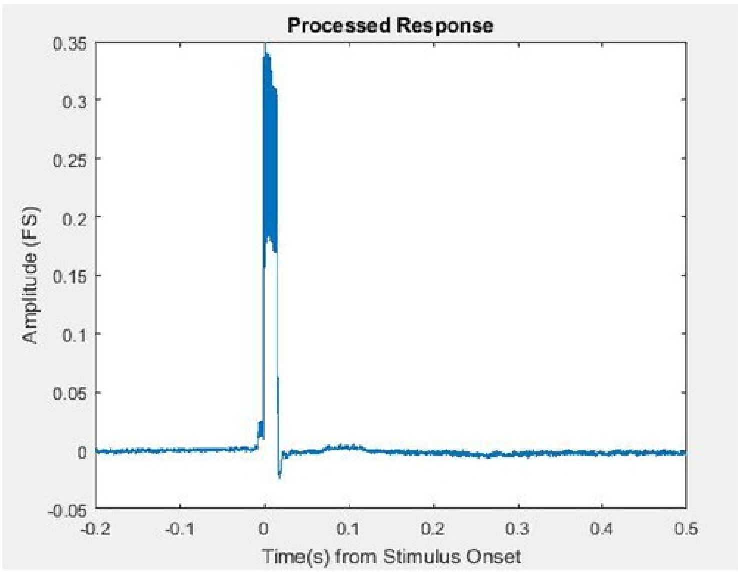


Figure 3.6: Complete Processed Response with Crosstalk.

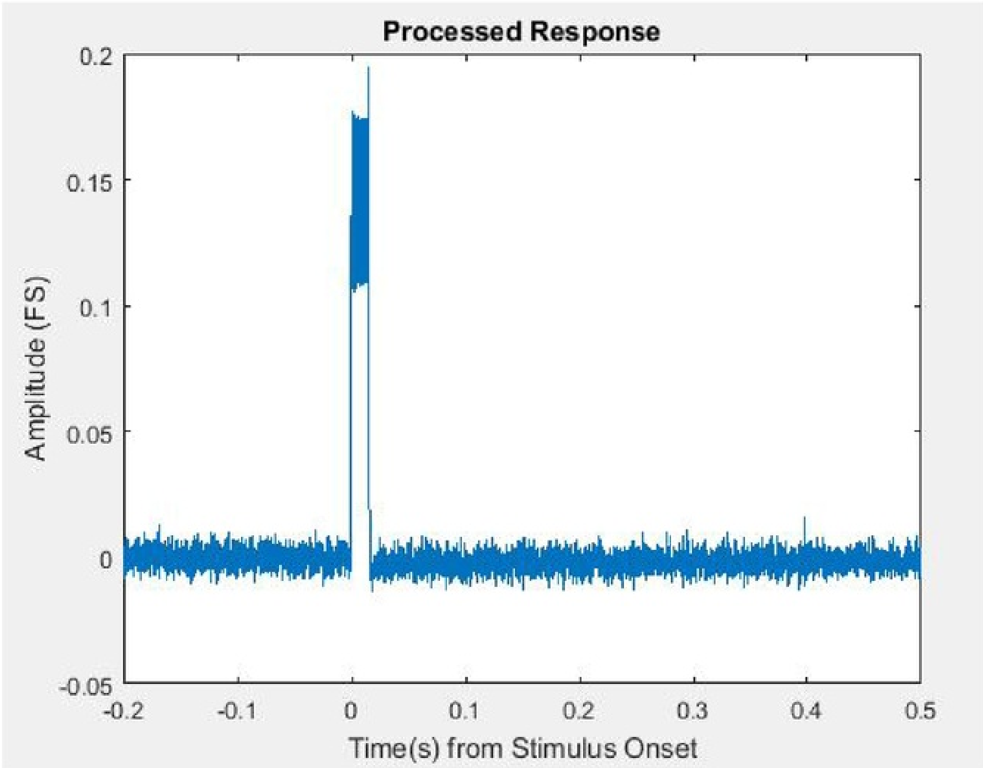


Figure 3.7: Individual Epoch with Crosstalk with speaker grounded.

50 Hz Hum

With the cross talk issues solved it was now possible to complete a recording on a person and potentially see an expected result. A test was performed on myself using the test setup to be outlined in section 4.2.1. The results of the recording show a large sinewave with a frequency of 50Hz through the EEG recorded channel as shown in Figure 3.8, the sine wave has an amplitude of 6 times that of anticipated minimal noise.
Modifying the test setup shows this hum is removed by either using a resistor pad instead of a human patient or by removing the amplified speaker connected to mains power, demonstrating this hum was caused by the mains power seeping back through the system into the recording channel from the amplified speaker, this was only visible when a human patient was used with the skin to electrode contact provided far higher resistance than the resistor pad allowing the feedback for the electrodes to carry the signal. To address the issue the system was changed such that the stimulus would be delivered via a battery powered amplified speaker, this remove the 50Hz AC signal from the mains by using a constant DC from the battery.

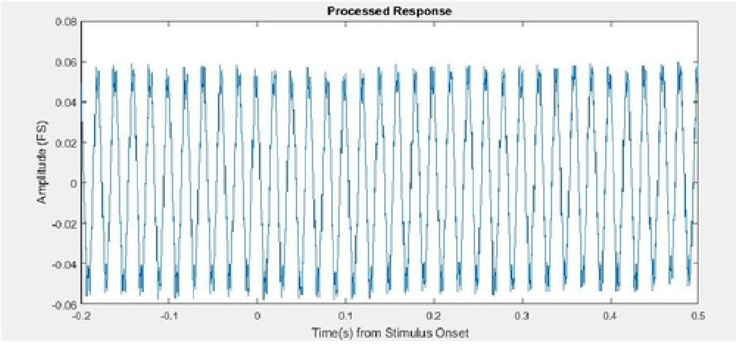


Figure 3.8: 50Hz Hum Displayed in the Processed Response.

Low Frequency Blocking

Re-testing the system on a human patient after the removal of the 50Hz hum and using the MATLab DSP analysis on the recorded results it was found that no CAEP response could be observed whereas other characteristics of the signal such as noise and artifacts could be observed clearly. It was deduced the low frequency range waves of the CAEP response, 1Hz to 10Hz, were being filtered by two DC blocking capacitors at the input line of the CC3200. Shorting these out allowed the response to be recorded unfiltered however allowed a DC voltage to bias the recorded results, this bias was reduced by inserting one 10k resistor to each record line.

3.3 Off Line Digital Signal Processing

As described by the research in Section 2.1.4 the CAEP responses occurring around 50ms after stimulus onset will be buried in noise with a frequency of around the 5Hz mark. The aim of the digital signal processing is to highlight this response amongst all other ongoing responses of the brain such that it can ultimately be identified with certainty by an audiologist. The strategy to achieve this identifiable signal will include two techniques to reduce the noise it is buried in, firstly filtering to remove unwanted frequencies and secondly averaging to remove random unwanted noise by a factor of 1/SQRT(N) where N is the number samples averaged. Each technique will require multiple stages to prepare the signals and allow the techniques to be performed successfully, the following section outlines the stages used to achieve the end result, to aid in the description of stages sample raw data obtained from NAL for the purpose of development will be used.

As stated in Section 1.3 the DSP to be undertaken in this thesis is to be offline and will be performed using MATLAB R2016b. The code shall be created such that a WAV file containing both the observed EEG recording and the time locked stimulus recording in their entirety will be processed and displayed as per the digital signal processing (DSP) requirements.

Using the characteristics of the desired response a series of requirements for the DSP can be produced to provide the best chance of observing it, the following are the DSP requirements as outlined by the National Acoustics Laboratories (NAL). These requirements have been adopted from the current NAL testing system HEARLab running the module CATE (Cortical Automatic Threshold Estimation) as detailed in Section 2.1.6.

Action	Value
Time mark response signals	Stimulus onset identifiable on response channel
High pass filter	0.16Hz
Down sample	1000 samples/sec
Low pass filter	30Hz
Epoch extraction	-200ms to + 500ms
Baseline removal	use -200ms to 0ms
Artefact rejection filter	-70uV
Epoch averaging	Time locked to epochs

Table 3.1: DSP Requirements.

3.3.1 Filtering

Downsampling

Both channels are to be recorded and saved as a WAV file with a sampling rate of 44.1kHz meaning a 4 minute recording would contain 10.58 million samples for each of the two channels. Thus down sampling was performed first to reduce the size of the data to be processed by further actions. Research suggests that down sampling to 1kHz is the minimum acceptable size before aliasing can occur and distort the response, however through trial and error it was found that this sample rate was still too low thus was set to 10kHz. In MATLAB the down sampling is performed using a re sampling function.

Filtering

The filtering was completed prior to epoch cutting as the MATLAB filtering is best suited for longer strings of data in comparison to multiple short strings. A Butterworth bandpass filter was created to filter between 0.16Hz and 30Hz so the desired response would be left undisturbed whilst the majority would be removed. The Butterworth filter was chosen due to its characteristics of being maximally flat with minimal ripples in the passband [25], the filter was given an order of 1 which gave the best result contrary to the researches order of 128.
The script below shows the transfer functions a and b would be returned for the 1st order

bandpass digital Butterworth filter with the cut off frequencies of high pass and low pass. Once the filter transfer functions are created the data can be zero-phase digital filtered using filtfilt. Figure 3.9 demonstrates the frequency domains of the recorded signal both before and after the filter was applied, it can be seen the filter is highly effect in removing the unwanted signals outside of the desired frequency range.

```
[b,a] = butter(1, [highPass, lowPass], 'bandpass');
EEG_Signal = filtfilt(b, a, raw_EEG_Signal);
```

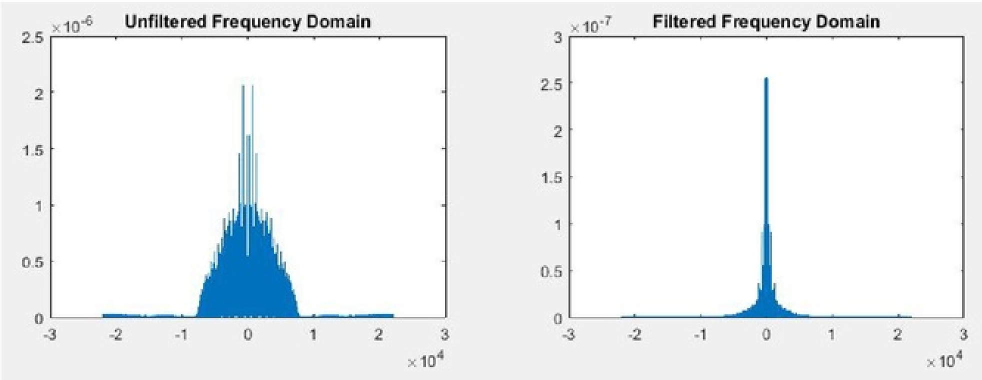


Figure 3.9: Comparison between Unfiltered and Filtered Signal.

3.3.2 Time Locked Averaging

Onset Detection

Onset detection is used to detect the start of each epoch within the data, it is performed by analysing the stimulus channel and observing the locations of each stimulus pulse within that data. The simplest form of onset detection is to analyse each sample of the stimulus channel and identify when the samples achieve over a certain value indicating a stimulus pulse, this simple method is reliant on the recording of the stimulus channel to be quiet with no large artefacts that could imitate a stimulus pulse. Figure 3.10 shows the onset of a stimulus pulse with the red circle indicating the first sample that is above the detection level thus indicating the presence of a stimulus. To achieve this method in MATLAB each sample of the stimulus channel was observed using a while statement as shown in the code below, an if statement was then used to detect any sample with a value greater than specified indicating the presence of a stimulus. When a stimulus is found on the stimulus channel an epoch is created from the EEG channel consisting of 200ms before the stimulus onset to 700ms after the stimulus onset.

```
while signal_sift + 1 < test_size - epoch_end
      i = signal_sift;
      artifact_flag = 0;
         if  Stimulus(i) > epochDetectionLevel
         epoch_sift = 1;
         epoch_cut=EEG_Signal(i-(epoch_start):i+(epoch_end));
         else
         signal_sift = signal_sift + 1;
         end
end
```

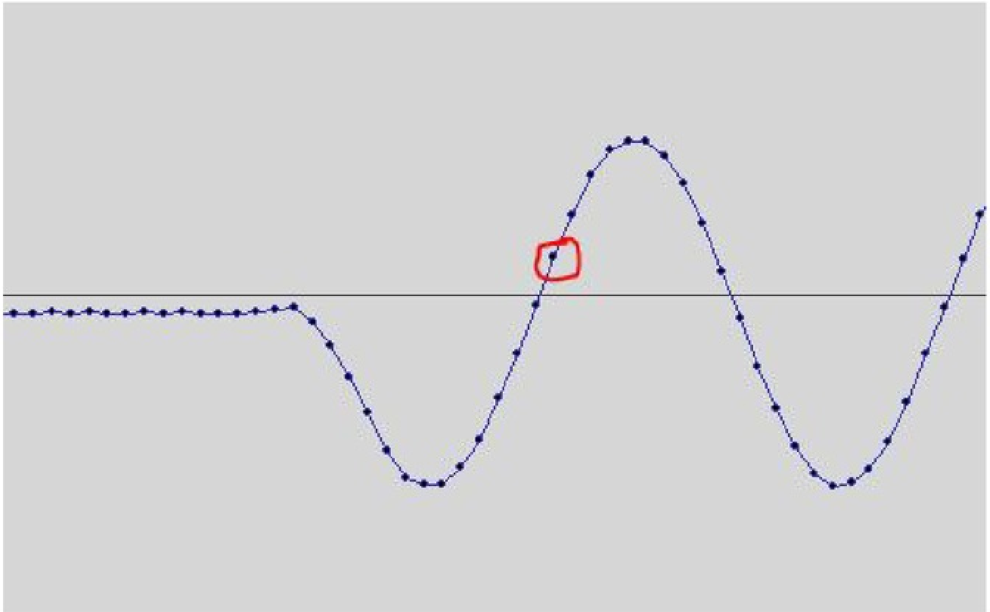


Figure 3.10: Stimulus Onset Detection Method.

Baseline Correction

Once created each epoch is subject to baseline correction to remove the majority of DC voltage within an epoch. The DC voltage can vary with each epoch so it is essential to apply unique baseline correction to each epoch, thus this was done immediately after the epoch was cut. Baseline correction averages the first 200ms worth of samples from each epoch to return the average offset from zero which is then removed from each sample of the entire epoch thus setting a new average of zero. Figure 3.11 displays a raw epoch before and after it has been subjected to baseline removal.

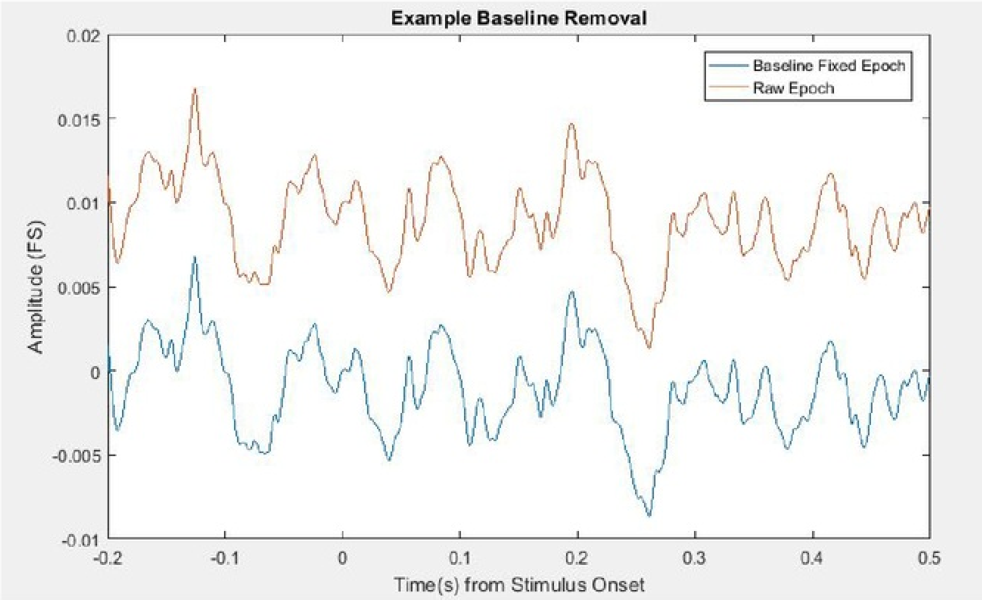


Figure 3.11: Comparison of no Baseline Removal against with Baseline Removal.

Artefact Rejection

Artefacts can be caused by many internal and external disturbances to the system or patient, as an example an artefacts can be caused by the patient blinking as shown in Figure 3.12 in which a large single wave pulse can be seen. Due to their magnitude these artefacts can saturate the processing and cause false data readings thus it is required to eliminate epochs that contain artefacts over a desired value as outlined by the requirements. This is performed similar to the onset detection on the stimulus channel, a while loop is used to sift through each sample of the EEG epoch with an if statement detecting any samples over the desired value. If an artefacts is found within an epoch, a flag is triggered and the sifting is stopped. A second if statement is used to check the artefacts flag and determine whether to save the epoch to the epoch array or discard the epoch.

```
while epoch_sift < epoch_length
 ii = epoch_sift;
   if baselineCorrect_epoch(ii) > artifactDetectionLevel
     epoch_sift = epoch_length + 1;
     artifact_flag = 1;
     artifact_count = artifact_count + 1;
    else
     epoch_sift = epoch_sift + 1;
```

```
    end
  end
```

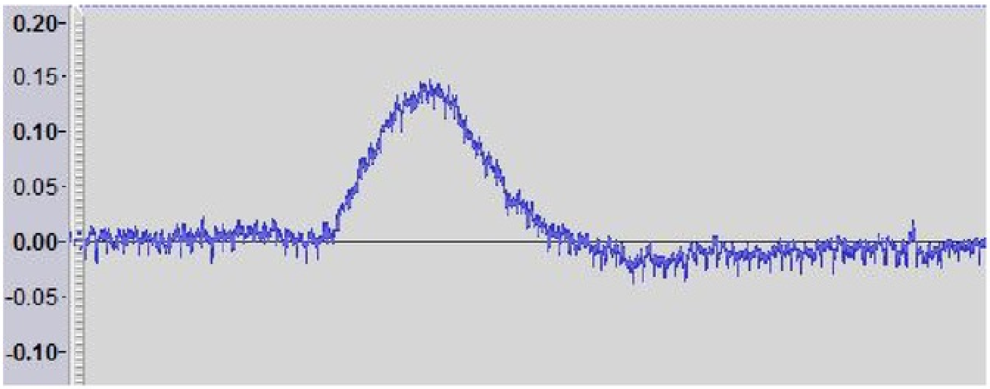


Figure 3.12: Artefact caused by Eye Blink.

Time Locked Averaging

Time locked averaging is used to reduce the background noise of the EEG signal and to highlight the CAEP response. The technique works on the principal that random noise within the signal is reduced towards zero by a factor of 1/SQRT(N) and thus requires the desired response to be located at the same point within each epoch.
Figure 3.13 demonstrates the effectiveness of the technique by comparing three randomly selected epochs shown as blue, yellow and red against the average of 932 epochs shown in purple, it is interesting to note the P1-N1-P2 morphology of the CAEP response can be made out within averaged response.
The process was achieved through MATLAB by averaging each sample with the same relative samples in other accepted epochs stored within the array, since each epoch was cut with respect to the same point on the stimulus pulse the resultant average would be time locked. Simple MATLAB arithmetic was used to output a single epoch containing the average.

```
epochs_sum = sum(epoch_array,2);
epochs_average = epochs_sum./ epoch_count;
epoch_size = size(epochs_average,1);
```

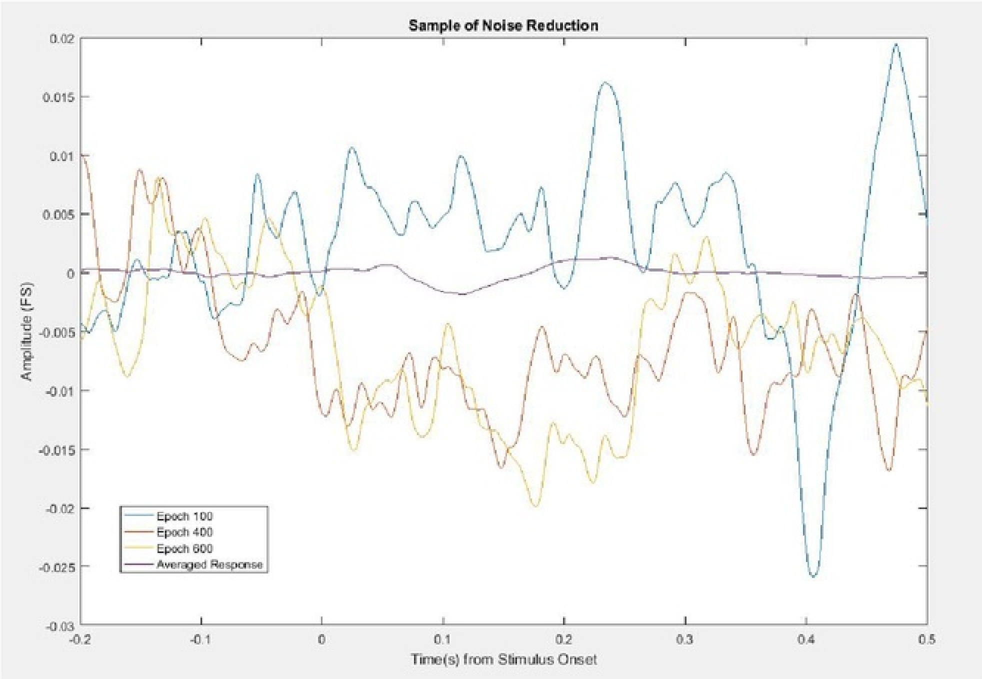


Figure 3.13: Comparison of individual epochs against averaged epoch.

3.3.3 Stimulus Creation

The stimulus created was a 0.5s pure tone with a frequency of 1000Hz to be repeated every 1.5s with silence in-between. The single 0.5s tone was created using an online tone generator [21] with a sine waveform, the generator produced a single WAV file with the tone demonstrated in Figure 3.14. The research conducted in Section 2.1.2 suggests a stimulus pulse that starts with a gradual increase in waveform amplitude akin to a ramp provides an auditory stimulus capable of evoking a stronger response, however due to the simplicity of the onset detection a pulse with a constant initial amplitude was proven more reliable. MATLab code was created to include the tone and 1.45s of zeros, this stimulus is then added onto its self 220 times to create the entire stimulus using a for loop. The code outputs a file in WAV format with the desired stimulus sequence.

```
stim = [stim;zeros(1.45*FS,1)];
Nb_Stim = 220;
Stim_Seq = [];
for i =1:Nb_Stim
    Stim_Seq = [Stim_Seq;stim];
end
```

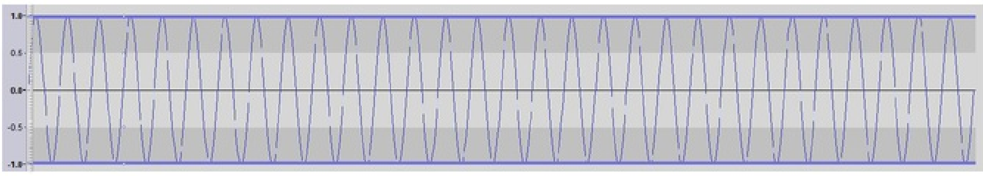


Figure 3.14: One Single 0.3s Stimulus.

Chapter 4

Experimental Procedure

In this chapter testing will be carried out on the prototype system, the testing will be in two parts, firstly to analyse the functionality and accuracy of the system and secondly to observe electromagnetic interference caused by the electromagnetic radiation of the CC3200. The testing will first involve a simulation of the MATLab coding which will include a comparison to a script developed by NAL for commercial use. Once the accuracy of the MATLab coding is confirmed several recordings will be conducted on both a human patient and artificial patient to determine the effectiveness of the system and conduct the investigation into Electromagnetic Interference.

4.1 Off Line Digital Signal Processing Testing

To observe a CAEP response audiologists will use a 700ms long epoch line graph that contains the processed data, if the stimulus is successfully heard by the patient the graph should display the P1-N1-P2 complex as shown in the research in section 2.1.1. The goal of the MATLab digital signal processing is to process the raw data and display this 700ms graph accurately to highlight the desired response. To test the developed MATLab code a comparison will be performed using a script developed by NALs electrophysiologist Fabrice Bardy for use in the commercially available system HEARLab as the standard. The same piece of EEG data in the form of a WAV file captured by NAL for CAEP testing purposes will be presented to each script for analysis to ensure continuity. Each script shall return a 700ms long processed graph to display its interpretation of the recorded response; the outputs will be compared to demonstrate the accuracy of the code written for prototype.

4.2 Complete System Testing

4.2.1 Experimental Setups

Once the MATLab processing has been proven, testing of the entire system can be undertaken this will involve delivering a stimulus stream to a patient and recording their response from the electrodes and delivering this response to the PC to be processed. Thus to test the prototype system as a whole it is essential to have a measurable CAEP response that can be observed and recorded, this can be created with a human patient or by generating a response with an artificial patient. The testing setups for each case are outlined below.

Artificial Patient

An artificial response will be created using a function generator to generate a sine wave signal to imitate a brains response to an evoked stimuli, the signal generated can be varied in multiple ways including frequency, amplitude, number of cycles and start time.
The stimulus was used to create a pulse in a trigger generator to trigger the function generator to output the desired signal. The Stimulus will be drawn out of the amplified speaker and inserted into an Audio isolation transformer to step-up the voltage of the signal and allow for a definite trigger within the trigger box. Similarly an audio isolation transformer was used from the output of the function generator to the electrodes so to step-down the voltage of the signal and not saturate the CODEC once the signal has passed through amplification in the prototype systems hardware.
A setup diagram is shown in Figure 4.1 with the flow of the signals shown, the actual setup is shown in Figure 4.2. It can be noticed from Figure 4.1 a signal splitter is used after the audio isolation transformer, this device ensures the active and reference electrodes do not measure the same signal such that when the differential amplifier is used the function generated remains. The setting for the function generator are shown in Table 4.1.

Characteristic	Value
Amplitude mVp-p	0.65
Frequency Hz	10
Pahse Shift	90°
Number of Cycles	1
Triggered on	Falling Edge

Table 4.1: Function Generator Settings for artificial CAEP.

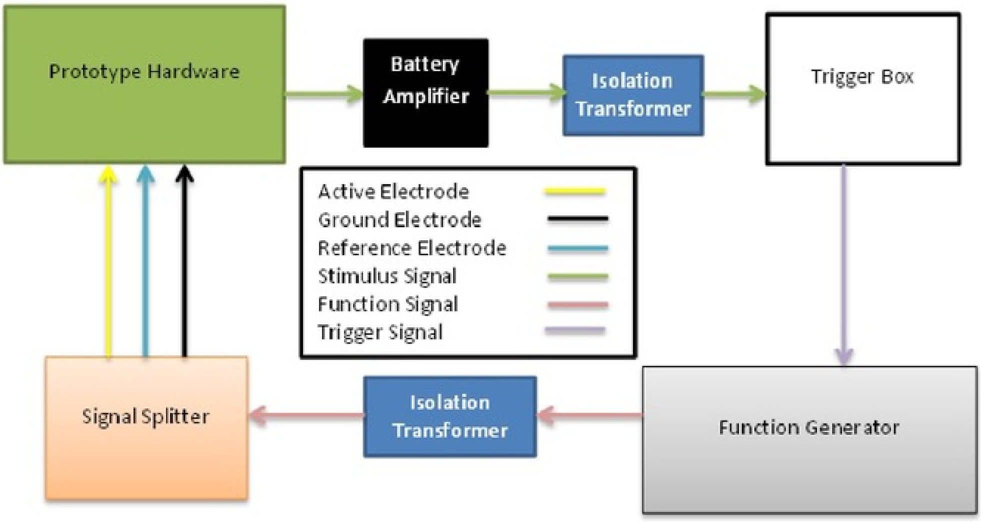


Figure 4.1: Block Diagram of Test System.

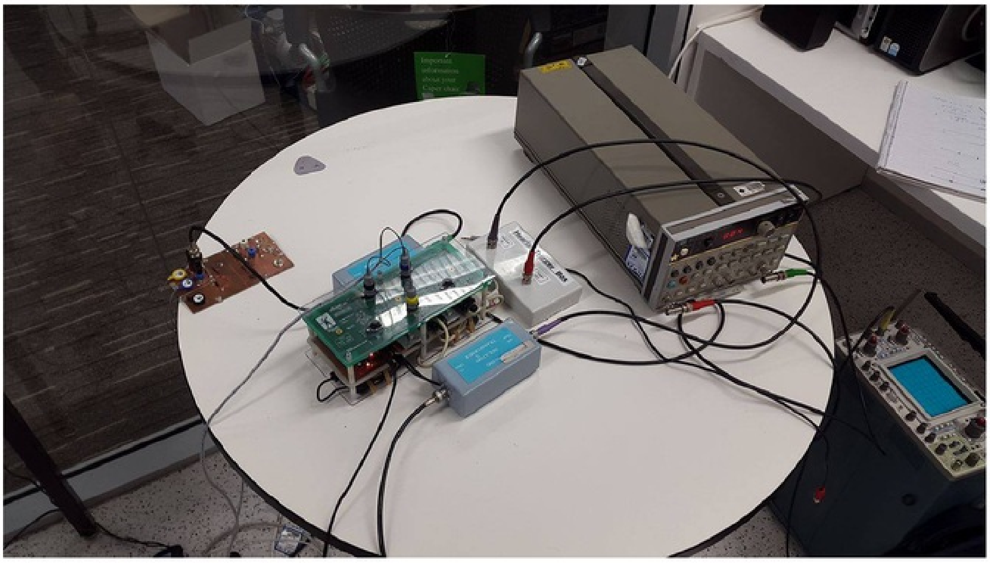


Figure 4.2: Test system set to distance of 0cm.

Human Patient

Testing of a human patient will involve three electrodes placed onto the head, the site of each electrode requires cleaning and gentle scrubbing to remove any dry or dead skin before an adhesive pad is attached to ensure minimal skin-electrode resistance. The three electrodes will be placed such that the active is on the mastoid, the reference is on the scalp and the ground is on the forehead. The Stimulus will be delivered to the patient by the amplified speaker to be located no more than 40cm from the patients head. Figure 4.3 shows the setup with myself as the human patient.

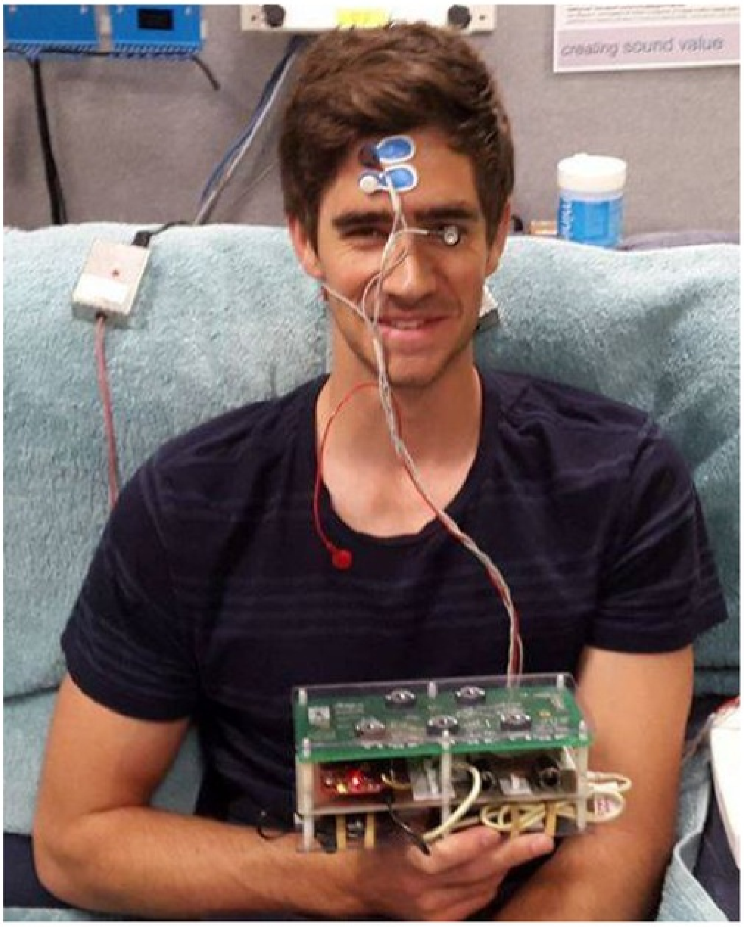


Figure 4.3: Prototype system with human patient setup.

4.2.2 Experimental Procedure

In order to design the test it is required to revisit the goal of the project, which is to determine whether the CAEP test can be successfully conducted with the implementation of a wireless system. This goal requires that the implementation of a wireless system will not affect the outcome of a CAEP test such that the output of the wireless system should be equal to that of a wired system. Thus the tests will be based upon proving this principle by comparing what is already known to what is actually achieved.

Artificial Patient Test

The first test will use the artificial patient setup allowing us to compare the signal known to be generated by the function generator to that displayed by the prototype system. The test will comprise of three recordings each consisting of 100 epochs, the function generator will generate a signal similar to that of the CAEP response with an amplitude of 0.65mV p-p, frequency of 10Hz and one sine wave cycle.
The test will produce three 700ms long epoch graphs displaying the processed result of each recording, the morphology of these recording will be compared and if successful will draw out a waveform similar to that produced by the function generator. The test will be deemed successful if a comparable sine wave of equal frequency, number of cycles and constant amplitude can be observed at 100ms post stimulus onset.

Human Patient Test

The second prototype system test will vary significantly from the first however still work with the principle of comparing to what is already known. The second test is devised to analyse the operation with a human patient, the purpose of the test is to observe and identify any issues with the system in a real life situation that may not occur with a simulated patient. The test involves performing three recordings on myself using the prototype system, the same testing conditions will then be replicated and three recordings will be completed using a commercially available wired system similar to HEARLab. In this instance the wired system will act as the standard to compare and evaluate the results of the prototype system against. Each of the six recordings will consist of a stimulus stream of 220 epochs.

4.3 Interference Testing

Interference testing will be carried out to conduct the second major analysis of this thesis, the interference testing will make use of the unique characteristics of electromagnetic fields and Wi-Fi transmission. The research in Section 2.3 showed that electromagnetic radiation will operate with a few discernible characteristics such as field strength will decrease with an increase in distance from the source and the power of the field strength will increase and decrease with each packet of data at regular time intervals, using these characteristics one test has been developed to potentially highlight the presence of electromagnetic interference within a CAEP recording.

4.3.1 Electromagnetic Interference Test 1

The test will attempt to observe electromagnetic interference using the characteristic of decreasing field strength with an increasing distance from the source. To achieve this the distance between the CC3200, the source of electromagnetic radiation, and the recording electrodes will be varied to analyse the effect of the electromagnetic field created by the CC3200 on the response of the patient and recorded by the electrodes. Two recordings will be completed with the CC3200 located at 0cm and 60cm from the observing electrodes. The test will be repeated twice, initially with the artificial patient and thereafter with a human patient.

Artificial Patient

To observe the results snippets of the EEG channel will be observed for interference artefacts, MATLab script will be written to analyse the snippets to aid in determining the presence of interference. Firstly the artificial patient will be used as this is expected to provide a larger interference with a more noticeable artefact due to the low noise of the EEG recording. With this recording a detection method will be used to detect the onset of an artefact over a certain value indicating the potential presence of interference, once detected epochs will be cut from the recording at the same moment of onset for each artefact detected allow us to perform time locked averaging and determine similarities in the artefacts observed. Information such as the power induced and frequency of the artefacts will be gathered from this initial test.

Human Patient

Once the artificial human has demonstrated the presences of artefacts the EEG recording channel from a human patient will be used in an attempt to observe the same result. As the noise levels when testing a human patient are far greater than an artificial patient we will be unable to successfully detect artefacts within the recorded signal, to overcome this we will rely on the results of the previous test, specifically the frequency of each artefact, to determine the time marking for time locked averaging. To aid in the understanding of this concept Figure 4.4 has been included which displays the raw EEG recording channels of the artificial patient on top and human patient underneath, the artefact peak is clearly discernible within the top recording and not so in the recording beneath it. The principal of this strategy is to highlight repetitions within the recorded signal whilst reducing random noise through averaging. This test will demonstrate the presence of an artefact within the EEG recording of a human patient.
The processing for both parts of the first test will included down sampling, baseline correction and time locked averaging, filtering and artefact detection will not be used as they may remove the sought signals.

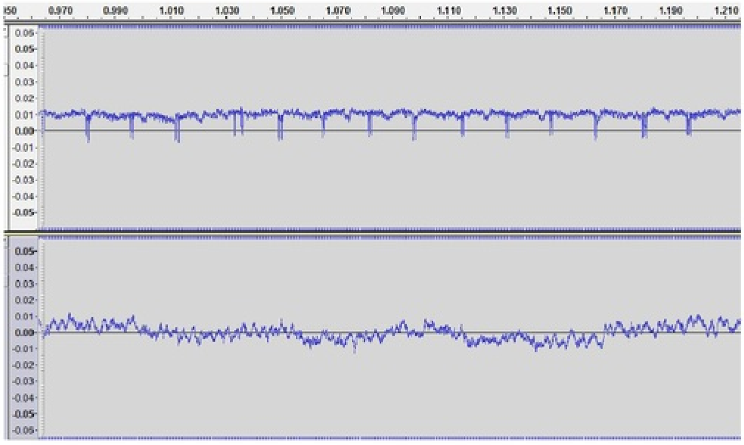


Figure 4.4: Comparison of Raw Recorded Signal from Artificial Patient (top) and Human Patient (bottom).

4.3.2 Electromagnetic Interference Test 2

A second test is to be completed by comparing the effect of a CAEP recording with and without the presence of electromagnetic interference. A commercially available wired testing unit will be used to produce the stimuli and record the response of a human patient. The CC3200 is to be used as a source of electromagnetic interference, the prototype system is to be set up to receive and transmit so to recreate the electromagnetic radiation as seen in the previous experiment.
Four recordings are to be completed, two with the interference and two without any source of interference. The results will be processed as per normal CAEP digital signal processing requirements including filtering and artefact detection, the aim of this experiment is to identify the presence of an interference artefact within a completely processed response.

Chapter 5

Results and Discussions

In this chapter the of the tests will be presented and analysed. the disccussion has been encorporated into the results as some results are best viewed with an understanding of the previous results

5.1 Off Line Digital Signal Processing Testing Results

The test to compare the MATLab script developed for the prototype with that developed by NAL for the HEARLab system was successfully undertaken, the raw data of two recordings provided by NAL was used to output a single processed epoch for each recording for both scripts. The two outputs from each script were then averaged to provide Figure 5.1 which shows the epoch outputs, for comparison purposes the outputs of both scripts can be seen in the figure. The graphs show the first 200ms prior to stimulus onset followed by 500ms post stimulus onset, the recordings were made up of 932 epochs. The point of interest within the comparison is within the response time from 0 to 0.3s, two large diversion points can be seen in the comparison at the first trough and second peak in the response with the prototype script 0.5 uV below the NAL script for both.

5.2 Off Line Digital Signal Processing Testing Disscussion

The research suggests the expected response after the stimulus onset will consist of a small peak at 0.05s followed by a large trough at 0.1s finishing with a large peak at 0.25s, with the epoch consisting of random noise both before and after the response. Using this morphology of the expected response it can be seen that for both scripts a response is clearly visible in Figure 5.1. More importantly it can be seen that the morphology of the script developed for the wireless prototype closely follows the morphology of the NAL developed script during the time period of the expect response.

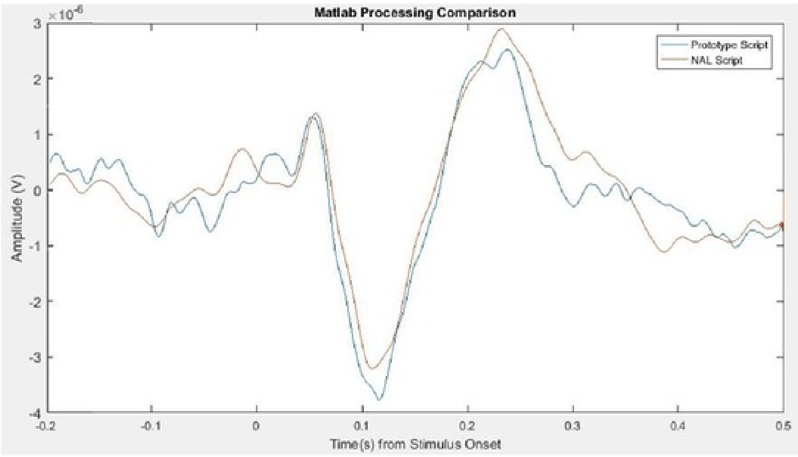


Figure 5.1: Comparison of MATLab results using the same data .

It should be noted that large deviations do occur in the wave form before and after the response and it is believed this is as a result of the differing in filter mechanisms used. This test is to be viewed as a success as it demonstrates the processing script developed for the prototype is capable of highlight a response with CAEP characteristics within a raw recording, this is essential as it allows us to rule out the processing as a potential issue within the prototype system when developing further.

5.3 Complete System Testing Results

5.3.1 Artificial Patient Test

Figure 5.2 shows the processed responses of two recordings using the artificial patient setup, a strong correlation between the two recordings at the time of the expect response can be seen with both recordings showing a trough and peak of equal amplitude, start time and frequency with the function generator with each recording consisting of 90-100 accepted epochs. There is less than 0.1 uV deviation between the two recordings for the time of the response.

5.3.2 Human Patient Test

Figure 5.3 shows the processed response of three recordings performed on a human patient using a 0.3s stimulus with each recording made up of 90-100 epochs. Each recording was performed under the same conditions on the same patient with two minutes break between recordings and no modifications made to the experimental setup.All three recordings show a large trough at 0.1s with a deviation between trough peaks of 0.6uV. No other obvious

features of a response are portrayed in this graph.
Figure 5.4 demonstrates the same test performed with a wired system on the same patient for comparison purposes. The recordings show a definite peak at 0.05s and a trough at 0.1s with no other obvious similarities throughout the responses.

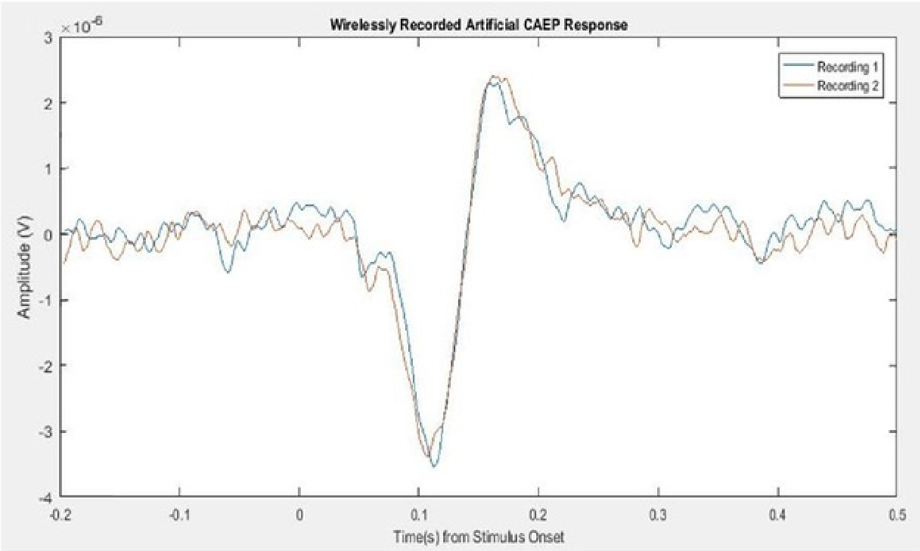


Figure 5.2: Wireless Recordings performed with Artificial Patient on Prototype System.

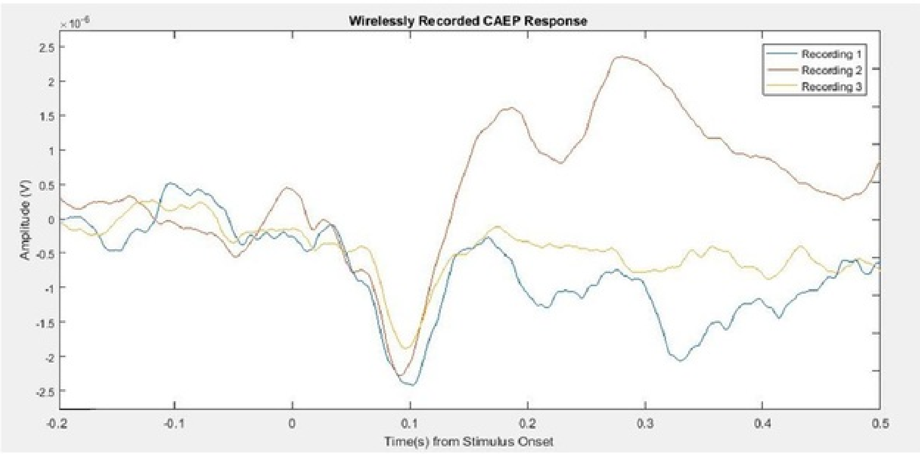


Figure 5.3: Wireless Recordings performed with Human Patient on Prototype System.

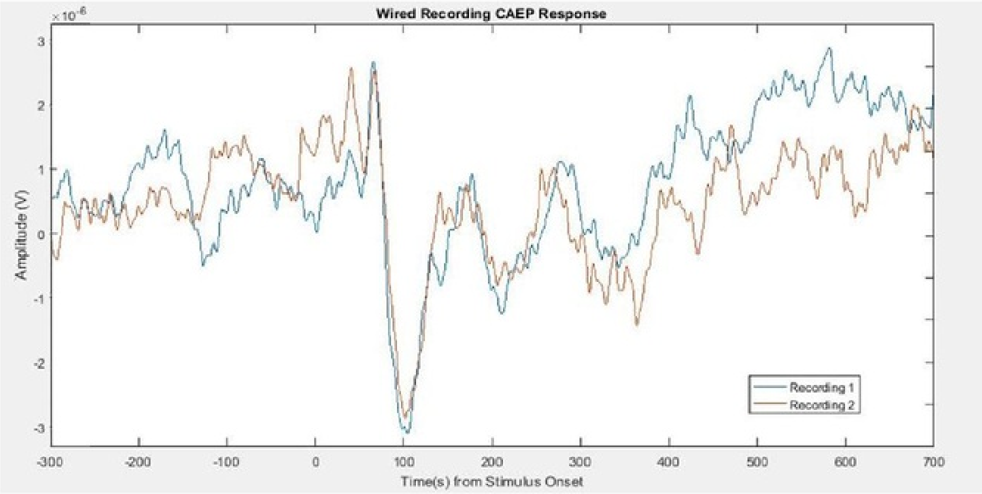


Figure 5.4: Wired Recordings performed with Human Patient on Commercial System.

5.4 Complete System Testing Disscussion

5.4.1 Artificial Patient

For this test the generator was set to create a function at 0.05s after the stimulus onset, the function would consist of a single sine wave pulse with an amplitude of 0.65mV p-p, frequency of 10Hz and phase shifted by 900, the result of which would be a single large trough followed by a large peak. The result demonstrates the function produced is clearly visible within the recording as shown in Figure 5.2. This result show us that a response with the frequency of 10Hz, equal to that of a CAEP response, can be successfully observed by the prototype system, this is important as it proves the wireless system will not impede the transmission of the response due to its low frequency. Unfortunately this test cannot conclusively prove the success of the system as the artificial patient setup provides far less noise than that of a human patient.

5.4.2 Human Patient

The recordings of Figure 5.3 show a significantly larger variation in all areas of the epoch in comparison to the previous two figures, despite this each recording can be seen to complete a trough at 0.1 seconds after the stimulus onset which is characteristic of the CAEP response. This single trough is enough to prove the presence of a response as it shares the same characteristics of frequency and start point in each recording. The characteristic of the amplitude of the troughs is not as important as it is well documented that responses will decrease in amplitude as exposure to the same stimulus is continued. It is believed the brain tires from repeated exposure to the same stimulus and the evoked

response to the stimulus is reduced.
The morphology of the recordings do not show the two peaks expected of a CAEP response at 0.05s and 0.25s, small raises in the wave can be seen at these time locations however these are no more outstanding than the surrounding random noise and thus cannot be definitely attributed to a response. As the tests on the MATLab processing and complete system with an artificial human were highly successful in displaying the response in full it can be believed that the issue arises when observing the response of a human.
Figure 4 shows two recordings each of 150 epochs conducted on the same patient as done in Figure 5.3, the system used to conduct these recordings was a commercially available wired system similar to HEARLab used by audiologist at NAL. As the results of patients can vary depending on their age the results of the wired system given in Figure 5.4 can be used as a standard for the patient, the morphology of this response shows a strong trough at 0.1 seconds after stimulus onset similar to that found in Figure 5.3 by the wireless prototype. Figure 5.4 additional displays a large initial peak of the response at 0.05s post stimulus onset that is not seen in the prototypes processed epoch. Thus using Figure 5.4s morphology as a standard it can be seen that the prototype is able to highlight the presence of a CAEP response however for this patient the initial peak of the response cannot be observed.

5.5 Interference Testing

5.5.1 Electromagnetic Interference Test 1 Results

Artificial Patient

The results of Interference Test 1 on the artificial patient are shown below in Figures 5–8. All Figures shown in the time domain in this section are shown using Full Scale for amplitude, the full scale of the system is equal to 0.5V. Figures 5.5 and 5.7 show the processed response of 90 epochs with the source of wireless transmission located at 0cm and 60cm respectively from the source of the electromagnetic radiation created by the wireless transmission. In both Figures two distinct pulses can be seen with a separation between peaks of 0.2ms. By comparing both figures it can be seen the amplitude of the peak in Figure 5.5 is slightly large than that in Figure 5.7 with a difference of 2.5 mFS (1.25mv). Further analysis of the results shows the mean separation of each artefact is 704 samples, this results in a time delay of 16ms between each artefact.
Figures 6 and 8 show the frequency domain of the raw data used to display the outputs of Figures 5.5 and 5.7 respectively. A large peak can be seen at 61 Hz for both spectrum's, by comparing these peaks at 61Hz between Figures 5.6 and 5.8 it can be seen that the system at 0cm, Figure 5.6, has slightly high power at -62.5 dB compared to -63dB.

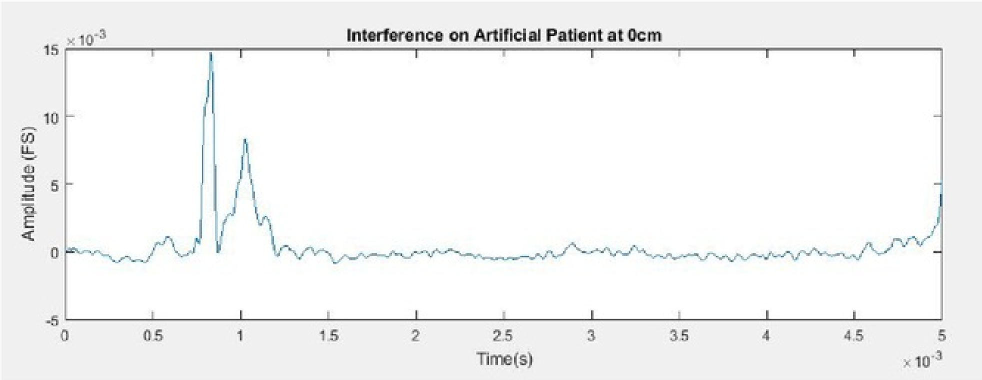


Figure 5.5: Processed artefacts on Artifical Patient EEG redcording channel at 0cm distance.

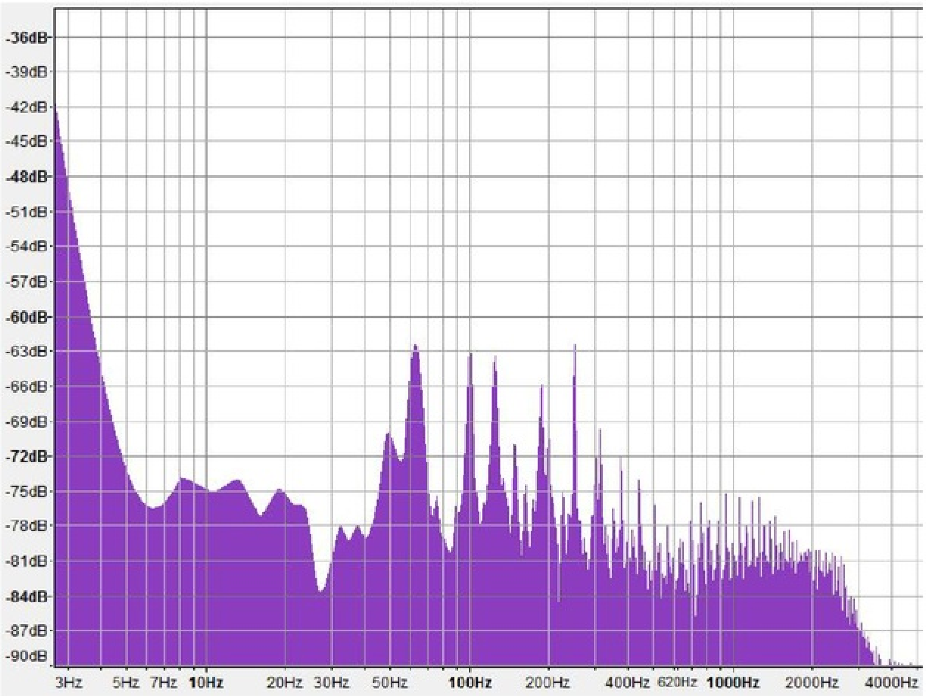


Figure 5.6: Spectral analysis of processed artefacts on Artifical Patient EEG redcording channel at 0cm distance.

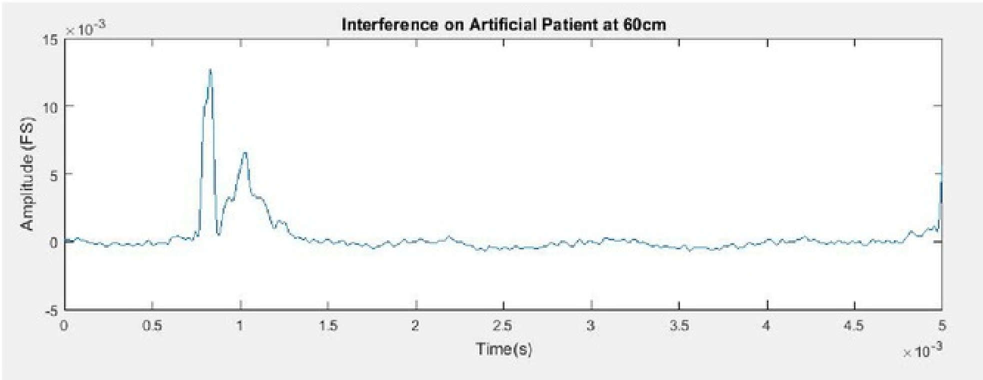


Figure 5.7: Processed artefacts on Artifical Patient EEG redcording channel at 60cm distance.

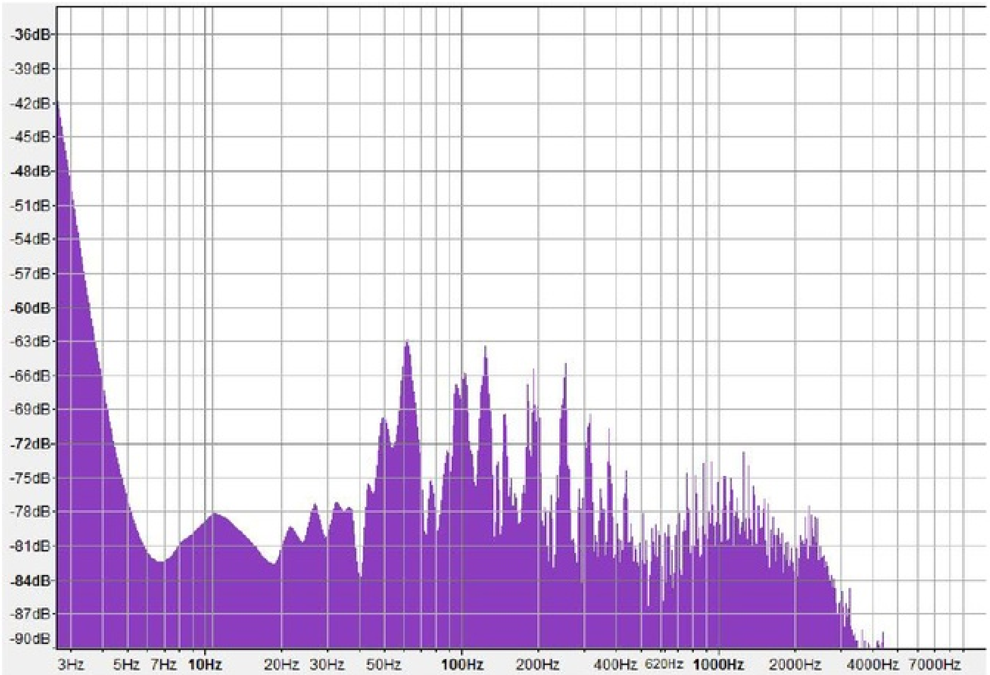


Figure 5.8: Spectral analysis of processed artefacts on Artifical Patient EEG redcording channel at 60cm distance.

Human Patient

Figures 5.9 and 5.10 display the results of Interference Test 1 on the human patient, similar to the artificial patient test the location of the electrodes from the source of the electromagnetic interference is the variable with Figure 5.9 displaying the results for a distance 0cm and Figure 5.10 60cm. Both results consist 80 epochs each that were subject to similar processing methods used on the artificial patient with the exception of the epoch cutting method. Figure 5.9 displays two distinct peaks with a separation of 0.8ms, whilst Figure 5.10 displays the morphology similar to a mound however the mound is quite definite. Figure 5.9 reaches an amplitude of 4 mFS (0.002mV) at its peak and Figure 5.10 is 2 mFS (0.001mV) at its peak.

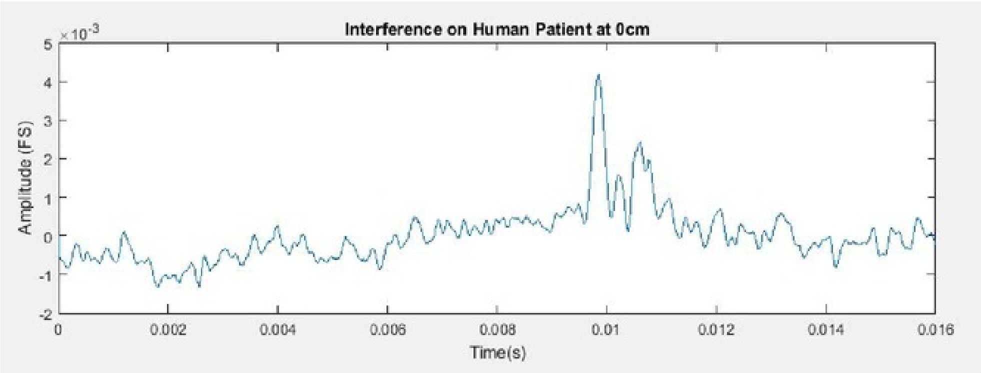


Figure 5.9: Processed artefacts on Human Patient EEG redcording channel at 0cm distance.

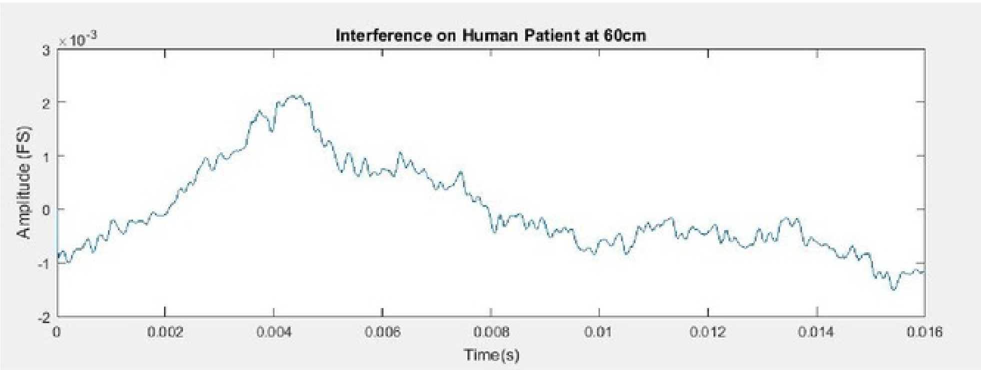


Figure 5.10: Processed artefacts on Human Patient EEG redcording channel at 60cm distance.

5.5.2 Electromagnetic Interference Test 1 Disscussion

Artificial Patient

This test attempted to observe the interference created by the wireless transmission on the artificial patient test setup shown in Section 4.2.1. As outlined in Section 4.2.2 the test used time locked averaging to observe any repeated interference within the EEG recorded signal. Similar to the process used to observe the CAEP response an epoch of 5ms was cut from the recorded signal each time an artefact was detected, these epochs were then used to output a single processed epoch. The test attempted to confirm the regularity of the artefacts within the recorded signal and confirm their relation to the use of wireless transmission.

Due to the time locked processing of the outputs shown in Figures 5.5 and 5.7 it can be deduced that the peaks present in both Figures are a repeating occurrence which repeat at very regular intervals of 16ms. Converting the period of 16ms between pulses to a frequency we achieve a value of roughly 60Hz, indicating these pulses are creating the large peaks in the frequency spectrums for both distance instances.
By comparing the combined results of both distance instance against each other, that of 0cm and 60cm from the source of electromagnetic interference, it was observed that when the electrodes were closer to the source of radiation on average a pulse with higher amplitude and more power was induced in the signal. This indicates the pulses induced and observed within the results have a correlation to the distance of the source of the electromagnetic radiation, and are thus more than likely created by the wireless transmission of the CC3200.

Human Patient

The results from the artificial patient test demonstrate that electromagnetic radiation is induced in the recording signal when using the artificial patient setup. The results from this test were used to define the parameters for the human patient interference test. This used the same principal of time locked averaging in an attempt to observe a repeated artefact caused by the electromagnetic interference, however the cutting of each epoch was not dictated by the presence of an artefact but was cut at regular time intervals of 16ms. The aim of this was to potentially observe any artefact without being able to quantitatively identify the artefact in the raw signal.
Similar to the results gathered from the artificial patient, Figures 5.9 and 5.10 show that an artefact is present within the EEG recorded signal when a recording is performed on a human. Since the figures display a time locked processed output it is established that the artefact is repeated at regular intervals and further more since the time locking period is derived from the artefacts found using the artificial human it can be concluded that they are caused by the same source. Comparing the Figures 5.9 and 5.10 we see that the morphology of two peaks is markedly more distinct in Figure 5.9 than in 5.10, it is possible that an error occurred in the transmission causing one of the signal pulses to be

delayed by a split second resulting in a disrupted average. Nonetheless the amplitudes of the peaks in both results have a sizeable difference indicating that the distance of the electromagnetic radiation source from the patient affects the amplitude of the signal induced in the EEG recording.
It is believed the artefact observed in the test is recorded by the electrodes and not induced into the system later on by the hardware. This assumption is established firstly from the characteristic of a decreasing amplitude in the artefact with an increase in the distance of the electrodes from the test setup and secondly from the difference in amplitude between the results of the artificial patient and human patient. By comparing the results taken at 0cm from the source of electromagnetic radiation for both the artificial patient and human patient we see a sizeable difference in the resultant artefact induced. It is believed the artificial patient induces the signal through inductors used in the transformer to step down the signal from the function generator to the electrodes, whilst it is believed the skin of the human patient induces the artefact signal.

5.5.3 Electromagnetic Interference Test 2 Results

Figure 5.11 shows the processed recordings with and without electromagnetic interference, the results show small peak at 0.05s and a large trough at 0.1s for all recordings. There is little deviation between the recordings for all parts of the result, it can be seen the trough at 0.1s has the largest deviation between all recordings.

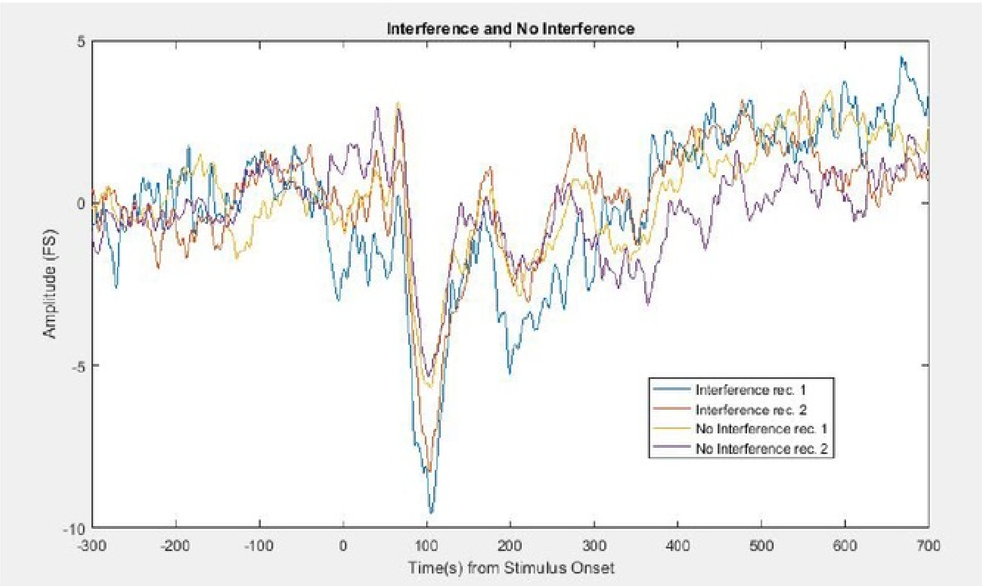


Figure 5.11: Comparison of with and without Interference in processed response.

5.5.4 Electromagnetic Interference Test 2 Disscussion

Interference Test 2 was used to compare the difference of a processed response with and without the presence of electromagnetic radiation. During the test four recordings were completed on a commercially available wired testing system similar to HEARLab to observe the CAEP response. For two of the recordings electromagnetic interference was presented at a range of 10cm–15cm to the patients head supporting the electrodes. Each recording was processed by the digital signal processing designed specifically for the use with the wired system.

The aim of this test was to observe whether the electromagnetic interference observed in the raw signal would be viewed in the processed output when CAEP specific processing is used. As can be seen in Figure 5.11 the CAEP response can be seen in each recording with the troughs located at 0.1s after stimulus onset. When observing the response closely it can be seen the magnitude of each response differs, interestingly the interference affected responses have a larger magnitude response than the non-interference affect responses, this is a commonly known issue when recording CAEP responses that the magnitude of a response will decrease the longer a patient is tested in one sitting and thus has no relation to the presence of interference. Further analysis of the results shows there is little to suggest the electromagnetic interference had any impact on the processed outputs of the recordings as the morphology of all result follows a similar paths with minor deviations.

The disappearance of the electromagnetic interference in the CAEP processed responses demonstrates the processing is effective in removing the interference signal from the recording. To understand how it is effective we must revisit the result of Interference Test 1 specifically the spectral analysis shown in Figure 5.6, in the plot we see and proved the peak at roughly 60 Hz is caused by the interference artefact given its period of 16ms. Given the band pass filtering for the CAEP digital signal processing occurs at 0.16Hz to 30Hz it is expected this artefact is removed when the filter is applied as seen in Figure 5.11.

Understanding the mechanism that stops the interference from being present in the response will allow us to determine the limitations of this mechanism and determine at what point the interference would be present. There are two separate conditions that can allow its presence. Firstly if the artefact remained with a frequency of roughly 60Hz however was time locked to the stimulus and thus response, in this instance a low order filter may let some of the power caused by the interference through the passed band and allow it to be present in the same location of each epoch subsequently allowing it to be present in the processed response. Secondly if the rate of transmission of the electromagnetic radiation was reduced to 33ms the artefact would have a frequency of 30 Hz and thus would not be filtered out of the signal, in this case however it would more than likely be considered random when compared to the stimulus and be removed through averaging. However If both conditions were met the artefact would likely be present in the processed response at there will be no affective mechanism to remove it.

Chapter 6

Conclusions

This chapter will outline the conclusions and summaries of this thesis and of the combined project of Zhihao and myself.

6.1 Conclusions

The primary objective of this research thesis was to develop a simple functional prototype of a wireless CAEP test system, the prototype was to prove the concept of wirelessly transmitting the recordings of the test including a stimulus channel and EEG recorded channel such that they could be processed to display the results of the test on a PC. This thesis also aimed to observe the impact of electromagnetic interference caused by the wireless system on the results of the CAEP test in terms of how and why it could be affected and how it could be mitigated.

6.1.1 Prototype

Analysis of the offline prototype developed for this thesis shows the system was successful in producing the results required for a CAEP test, the successful result of this analysis was gained when comparing the results of this prototype to that of a wired system designed to achieve the same results. Several key conclusions can be drawn from this analysis, these include:

1. With minor alterations a microcontroller with wireless functionality such as the CC3200 Module LaunchPad combined with the Audio BoosterPack can transmit physiological signals without disturbances from the hardware. The alterations include removal of capacitors on the input line that can cause filtering of low frequency signals.

2. A microcontroller with wireless functionality such as the CC3200 Module LaunchPad combined with the Audio BoosterPack can be used to receive and present a stimulus signal to a patient when used with a battery powered amplifying speaker. The microcontroller can also be used to present the stimulus to the hardware

3. The Wi-Fi protocol UDP can be used to successfully to transmit the signal within the CAEP prototype system. Even though the UDP protocol has the issue of dropped packets, it was observed for this specific use the amount of droped packets per recording is negligible due to the use of time locked averaging.

4. MATLab can be successfully used to digitally process the two recorded and saved signals of a CAEP test offline to display the required response.

6.1.2 Electromagnetic Interference

This thesis also analysed the presence of electromagnetic interference caused by the micro controller from the wireless transmission of the test recordings. The following conclusions are draw from the results of the experiment undertaken:

1. Electromagnetic interference is present within the raw recordings of a CAEP test both when using an artificial patient and using a human patient. The electromagnetic interference when using the CC3200 Module LaunchPad combined with the Audio BoosterPack using the Wi-Fi Audio Application had a frequency of 61Hz. It is believed this is due to the firmware that transmits the signal through the UDP socket. Changing the frequency of transmission will also change the frequency of interference.

2. The interference is markedly larger in the artificial patient than in the human patient. This is due to the setup of the artificial human containing a two inductors to allow for the reduction of power from the function generator, it is concluded the interference is induced with these. The interference from the human patient is induced with the skin and thus can be recorded by the electrodes connected to the skin.

3. The magnitude of the electromagnetic interference is reduced as the distance of the source of the electromagnetic radiation is increased from the patient attached to the electrodes. This was found with both the artificial patient and human patient at distances of 0cm and 60cm. This demonstrates the CC3200 transmission is the source of the electromagnetic interference.

4. The interference measured with a frequency of 61Hz has no effect on the result of the CAEP test. It was found by comparing the results with an without a source of interference the digital signal processing was successful in removing any presence of the interference from the final result allow the CAEP test to be conducted normally.

5. By understanding the processing to remove the interference from the final result of a CAEP test with the interference at a frequency of 60Hz it can be determined the interference would impact the final result if it satisfied two conditions, firstly if it had a frequency similar to that of the CAEP response or within the band pass of the filter, 0.16Hz to 30Hz and secondly if it was time locked to the stimulus pulse allowing it to be highlight with time locked averaging.

6.2 Summary

The goal of the project undertaken by both Zhihao and myself was to prove the concept of a wireless EEG system capable of delivering accurate and informative results when used to conduct the Cortical Auditory Evoked Potentials (CAEP) assessment. The results gathered and the methods shown by both students demonstrate this goal has been achieved to varying degrees of success as outlined in both theses, more importantly the theses show the concept is plausible if more resources such as time and money are available to further develop and refine the system.

The primary outcomes of the project found a wireless EEG system could be created to function without wires from the central unit at the patient end to the host PC at the audiologist end whilst delivering either real time and off line recording analysis. Despite real time recording analysis not function as desired the concept was proven. The project also found the Electromagnetic Interference generated from the wireless unit would have negligible impact on the displayed response.

Secondary outcomes of the project stem from problems encountered throughout the development of the project. Firstly working in a team environment on an industry project gave a great insight into the approaches used by professional Engineers to solve problems as a team, we were able to sit in on some engineering team meetings participate in product discussions. Secondly with so much independence within the project we were forced to learn from our mistakes to ensure our project stayed on the correct path and our goals were met. Lastly with a large time frame for the project, time management became an essential skill to ensure our time wasn't taken advantage of and enough time was left to complete each task as required.

Chapter 7

Future Work

Now that the concept of wireless transmission for a physiological signal acquisition system has been proven to work, future research would be in the area of refining the system to make it more user friendly, accurate and ready for commercialisation.
Further development of the system to increase the user friendliness would include the development of a multifunction GUI to allow the user more control over the system and understanding as to the progress of the test, this was developed and documented within Zhihaos thesis however the GUI needs an intuitive design. The Wi-Fi connectivity should be modified to allow an easy connect system that can connect to any PC without issue or thought of the PCs current IP address or port number, it was found most of the difficulties in setting up the system was to create the connection between the hardware and PC.
The test stimulus created for the system was of very simple design to demonstrate the functions of the system, a stimulus used in a commercial system would contain multiple different stimuli pulses aimed at keeping the patients cortex responding to its highest by keeping it alert, these stimuli use ID tags within the stimulus stream to denote what pulse is coming next. Further research could be performed to allow the wireless system to incorporate this functionality for an increase in accuracy of recordings. Further research could be performed on expanding the capabilities of the system to including testing outside of CAEP including Auditory Brain-stem Response testing.
To be acceptable for commercialisation, along with all the conditions required for user friendliness the systems hardware would need to be developed into a neater and more compact version to make it more robust and appealing to a buying market.

Chapter 8

Abbreviations

ABR	Auditory Brainstem Response
ASSR	Auditory Steady State Response
CAEP	Cortical Auditory Evoked Potentials
CCS	Code Composer Studio
CI	Cochlear Implant
EEG	Electroencephalogram
EI	Electromagnetic Interference
EMF	Electromagnetic Field
GUI	Graphical User Interface
ICA	Independent Component Analysis
IEC	International Electrotechnical Commission
LAN	Local Area Network
NAL	National Acoustics Laboratory
PC	Personal Computer
PCB	Printed Circuit Board
PCM	Pulse-Code Modulation
TCP	Transmission Control Protocol
TFRC	TCP Friendly Rate Control
UDP	User Datagram Protocol
WAN	Wider Area Network

Appendix A

Software Code

A.1 Overview

This appendix contains the Code used to create the software client and server for the prototype system

A.2 Java Client

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.io.*;
import java.net.*;
import javax.sound.sampled.*;
import java.net.InetAddress;

public class Client extends JFrame {

boolean stopaudioCapture = false;
ByteArrayOutputStream byteOutputStream;
AudioFormat adFormat;
AudioInputStream InputStream;
SourceDataLine sourceLine;
Graphics g;
static File soundFile;

public static void main(String args[]) {
new Client();
}
public Client() {
```

```
final JButton capture3 = new JButton("16");
final JButton stop = new JButton("Stop");

capture3.setEnabled(true);
stop.setEnabled(false);
capture3.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent e) {
capture3.setEnabled(false);
stop.setEnabled(true);
captureAudio3();
}
});
getContentPane().add(capture3);
stop.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent e) {
capture3.setEnabled(true);
stop.setEnabled(false);
stopaudioCapture = true;
sourceLine.close();
}
});
getContentPane().add(stop);

getContentPane().setLayout(new FlowLayout());
setTitle("Capture/Playback Demo");
setDefaultCloseOperation(EXIT_ON_CLOSE);
setSize(250, 100);
getContentPane().setBackground(Color.white);
setVisible(true);

g = (Graphics) this.getGraphics();
}


private void captureAudio3() {

String strFilename =  "DemoSequence220.wav";

try {
soundFile = new File(strFilename);
} catch (Exception e) {
e.printStackTrace();
```

```
System.exit(1);
}

try {
InputStream = AudioSystem.getAudioInputStream(soundFile);
} catch (Exception e) {
e.printStackTrace();
System.exit(1);
}

try {

adFormat = InputStream.getFormat();

DataLine.Info dataLineInfo = new DataLine.Info(SourceDataLine.class,
 adFormat);
sourceLine = (SourceDataLine) AudioSystem.getLine(dataLineInfo);
sourceLine.open(adFormat);
    BooleanControl muteControl=(BooleanControl)sourceLine.getControl
    (BooleanControl.Type.MUTE);
    muteControl.setValue(true);
sourceLine.start();
Thread captureThread = new Thread(new CaptureThread());
captureThread.start();

} catch (Exception e) {
StackTraceElement stackEle[] = e.getStackTrace();
for (StackTraceElement val : stackEle) {
System.out.println(val);
}
System.exit(0);
}
}

class CaptureThread extends Thread {
byte tempBuffer[] = new byte[1024 * 16];
public void run() {
byteOutputStream = new ByteArrayOutputStream();
stopaudioCapture = false;

try {
DatagramSocket clientSocket = new DatagramSocket(null);
//InetAddress IPAddress = InetAddress.getByName("10.126.215.114");
```

```
InetAddress IPAddress = InetAddress.getByName("192.168.1.11");

while (!stopaudioCapture) {
int cnt = InputStream.read(tempBuffer, 0, tempBuffer.length);
if (cnt > 0) {
sourceLine.write(tempBuffer, 0, cnt);
DatagramPacket sendPacket = new DatagramPacket(tempBuffer,
 tempBuffer.length, IPAddress, 5050);
clientSocket.send(sendPacket);
byteOutputStream.write(tempBuffer, 0, cnt);
}
}
byteOutputStream.close();
sourceLine.drain();
sourceLine.close();
} catch (Exception e) {
System.out.println("CaptureThread::run()" + e);
System.exit(0);
}
}
}
}
```

A.3 Java Server

```
import java.io.*;
import java.net.*;
import java.util.Arrays;
import java.util.LinkedList;
import java.util.Queue;

import javax.sound.sampled.AudioFormat;
import javax.sound.sampled.AudioInputStream;
import javax.sound.sampled.AudioSystem;
import javax.sound.sampled.DataLine;
import javax.sound.sampled.LineUnavailableException;
import javax.sound.sampled.SourceDataLine;
import javax.sound.sampled.UnsupportedAudioFileException;
import javax.swing.JFrame;

class Server extends JFrame{
```

```
private static final int BUFFER_SIZE = 4096;
SourceDataLine audioLine;
AudioInputStream audioStream;
boolean stopaudioCapture = false;


public class Player
{
AudioFormat myformat;
boolean initialized = false;
Queue<byte[]> playqueue = new LinkedList();
int bufferSize = BUFFER_SIZE * 100;


public Player(AudioFormat fmt)
{
myformat = fmt;
}

public void Open()
{

try {
DataLine.Info info = new DataLine.Info(SourceDataLine.class,
 myformat);
audioLine  = (SourceDataLine) AudioSystem.getLine(info);
audioLine.open(myformat, bufferSize);
Thread BufferThread = new Thread(new BufferThread());
BufferThread.start();
audioLine.start();

} catch (LineUnavailableException e) {
e.printStackTrace();
}
}

public void enqueue(byte[] audiodata, int size)
{
playqueue.add(Arrays.copyOf(audiodata, size));
}
```

```
  class BufferThread extends Thread {

  long totalbytes = 0;

      public void run() {
          try {

              while (true) {

               int playeravailable = audioLine.available();

               if (playeravailable >0 && playqueue.size() > 0){

               byte[] head = (byte[]) playqueue.peek();
               int headsize = head.length;

               if (playeravailable > headsize){

               totalbytes += headsize;

               System.out.println("Send " + totalbytes + " to play");

               audioLine.write((byte[]) playqueue.poll(), 0, headsize);

               }

               }

              }

          } catch (Exception e) {
              System.out.println(e);
              System.exit(0);
          }
      }
  }
}

Player myplayer;

private void UDPServer()
{
```

```
      try
      {
       DatagramSocket serverSocket = new DatagramSocket(5001);
       String audioFilePath = "C:/Users/Meggitt/Documents/ENGG411/test.wav";
       File audioFile = new File(audioFilePath);
       AudioInputStream audioStream = AudioSystem.getAudioInputStream(audioFile);
       AudioFormat format = audioStream.getFormat();
       audioStream.close();
       myplayer = new Player(format);
       myplayer.Open();
       byte[] receiveData = new byte[4096 * 10];
       System.out.println ("Waiting for datagram packet");
       int totalreceived = 0;

       while(true)
         {

           DatagramPacket receivePacket =
           new DatagramPacket(receiveData, receiveData.length);

           try{
           serverSocket.receive(receivePacket);
         }catch (SocketTimeoutException s) {
             System.out.println("Socket timed out!");
             serverSocket.close();
             break;
       }
           int size = receivePacket.getLength();
           totalreceived+=size;
           myplayer.enqueue(receivePacket.getData(), size);
         }
        serverSocket.close();
             audioLine.drain();
             audioLine.close();

      }catch (UnsupportedAudioFileException ex) {
         System.out.println("UDP Port 5001 is occupied.");
         System.exit(1);
      }catch(IOException ex){
     System.out.println("UDP Port 5001 is occupied.");
     System.exit(1);
      }
}
```

```
public static void main(String args[]) throws Exception
    {
new Server().UDPServer();
    }

}
```

Appendix B

MATLab Script

B.1 Overview

This section contains the MATLab script used to process the recordings.

B.2 CAEP Processing

```
clear all
clc

[highSample,fsHigh] = audioread('0cmTomInterference.wav');

%Parameters

downSampleTo = 10000; %set sample rate
lowPassFilterKHz = 30;
highPassFilterKHz = 0.16;
epochDetectionLevel = 0.01;
artifactDetectionLevel = 0.04;

%Downsample

downSampleFrom = fsHigh;
lowSample = resample(highSample, downSampleTo, downSampleFrom);
num_high_samples = size(highSample);
num_low_samples = size(lowSample);


fs = downSampleTo;
test_size = (num_low_samples(1,1));
```

```
raw_EEG_Signal = lowSample(:,1); %EEG
Stimulus = lowSample(:,2); %stimulus


% Construct filter and filter the averaged epochs

order = 1; %First order filter
highPass = highPassFilterKHz / (fs/2); %high pass filter from 0.16Hz
lowPass = lowPassFilterKHz / (fs/2); %low pass filter from 30Hz
[b,a] = butter(order, [highPass, lowPass], 'bandpass');
%construct the Butterworth filter
EEG_Signal = filtfilt(b, a, raw_EEG_Signal); % filter the signal



%Onset Detection and Epoch Cutting

signal_sift = round(fs*0.2);
epoch_count = 0; %to count how many epochs have been found
epoch_array = []; %to store found epochs
artifact_array = [];
artifact_count = 0;

%for testing system

epoch_start = round(fs*0.25);

epoch_end = round(fs*0.45);


%For human system
 %epoch_start = round(fs*0.2);
 %epoch_end = round(fs*0.5);
jump = round(fs*1.5);
epoch_length = round(fs*0.7);

while signal_sift + 1 < test_size - (fs*0.6) %loops whilst still inside sample
      i = signal_sift;
      artifact_flag = 0;
      %choose left or right for which channel
      if  Stimulus(i) > epochDetectionLevel %Stimulus found
          epoch_sift = 1;
          jitter = 0;
```

```
        epoch_cut = EEG_Signal(i+jitter-(epoch_start):i+jitter+(epoch_end));
        epoch_cut = 1 .* epoch_cut;
        % Baseline Correction

        epoch_chunk = epoch_cut(1:round(fs*0.178),1);
        epoch_chunk_sum = sum(epoch_chunk);
        epoch_chunk_size = size(epoch_chunk,1);
        epoch_chunk_average = epoch_chunk_sum ./ epoch_chunk_size;
        epoch_cut(1) = epoch_chunk_average;
        baselineCorrect_epoch = epoch_cut - epoch_chunk_average;

        %Artifact detection

        while epoch_sift < epoch_length
            ii = epoch_sift;

            if baselineCorrect_epoch(ii) > artifactDetectionLevel
             %artifact found
                epoch_sift = epoch_length + 1;
                artifact_flag = 1;
                artifact_count = artifact_count + 1;
            else
                epoch_sift = epoch_sift + 1;
            end
        end

        % Contaminated epoch rejection, or good epoch save

        if artifact_flag == 1
           artifact_array = [artifact_array baselineCorrect_epoch];
           signal_sift = signal_sift + (jump);

        else

           epoch_array = [epoch_array baselineCorrect_epoch];
           signal_sift = signal_sift + (jump);
           epoch_count = epoch_count + 1;
        end
    else %epoch hasn't been found
        signal_sift = signal_sift + 1;
    end
end
```

```
%Epoch Averaging

epochs_sum = sum(epoch_array,2);
epochs_average6 = epochs_sum./ epoch_count;
epoch_size = size(epochs_average6,1);

epochs_average5 = epochs_average6 .* 0.0011;




%Output Processed WAV File

filename = 'Average of epochs.wav'; % creates a name
audiowrite(filename,epochs_average,fs)


%plot results

displayFSPoints = epoch_size;
display_time_scale = linspace(-0.2,0.5,displayFSPoints)';
FigHandle = figure('Position', [300, 100, 1000, 700]);
figure, plot(display_time_scale, epochs_average5)

title('Processed Response');
xlabel('Time(s) from Stimulus Onset');
ylabel('Amplitude (V)');
hold off

figure, plot(display_time_scale,unbase);
title('Example Baseline Removal');
xlabel('Time(s) from Stimulus Onset');
ylabel('Amplitude (FS)');
legend('Baseline Fixed Epoch','Raw Epoch');
hold off
```

B.3 Interference Processing

```
clear all
clc
```

```
[highSample,fsHigh] = audioread('InterferenceCheck1.wav');

%Parameters
downSampleTo = 10000; %set sample rate
lowPassFilterKHz = 30;
highPassFilterKHz = 0.16;
epochDetectionLevel = 0.004;
artifactDetectionLevel = 3;
num_high_samples = size(highSample);
fs = fsHigh;
test_size = (num_high_samples(1,1));
EEG_Signal = highSample(:,1); %EEG
Stimulus = highSample(:,2); %stimulus




%Onset Detection and Epoch Cutting
signal_sift = round(fs*0.001);
epoch_count = 0; %to count how many epochs have been found
epoch_array = []; %to store found epochs
artifact_array = [];
artifact_count = 0;

%for testing system
epoch_start = round(fs*0.003);
epoch_end = round(fs*0.016);


%For human system
 %epoch_start = round(fs*0.001);
 %epoch_end = round(fs*0.5);
jump = 680;
epoch_length = round(fs*0.005);
Epoch_start_array = [0];
averageStartArray = [];
while signal_sift + 1 < test_size  - (fs*0.017)
%loops whilst still inside sample
      i = signal_sift;
      artifact_flag = 0;
      %choose left or right for which channel
      if  EEG_Signal(i) < epochDetectionLevel %Stimulus found
```

```
    epoch_sift = 1;
    jitter = 0;
    epoch_cut = EEG_Signal(i+jitter-(epoch_start):i+jitter+(epoch_end));
    epoch_cut = -1 .* epoch_cut;
    % Baseline Correction
    epoch_chunk = epoch_cut(1:round(fs*0.0009),1);
    epoch_chunk_sum = sum(epoch_chunk);
    epoch_chunk_size = size(epoch_chunk,1);
    epoch_chunk_average = epoch_chunk_sum ./ epoch_chunk_size;
    epoch_cut(1) = epoch_chunk_average;
    baselineCorrect_epoch = epoch_cut - epoch_chunk_average;
      Epoch_start_array = [ Epoch_start_array i];
      average_start = Epoch_start_array(end) - Epoch_start_array(end-1);
      averageStartArray = [averageStartArray average_start];

    %Artifact detection
    while epoch_sift < epoch_length
        ii = epoch_sift;

        if baselineCorrect_epoch(ii) > artifactDetectionLevel
        %artifact found
            epoch_sift = epoch_length + 1;
            artifact_flag = 1;
            artifact_count = artifact_count + 1;
        else
            epoch_sift = epoch_sift + 1;
        end
    end

    % Contaminated epoch rejection, or good epoch save
    if artifact_flag == 1
       artifact_array = [artifact_array baselineCorrect_epoch];
       signal_sift = signal_sift + (jump);

    else

       epoch_array = [epoch_array baselineCorrect_epoch];
       %adds new epoch to array of found epochs
       signal_sift = signal_sift + (jump);
       epoch_count = epoch_count + 1;
        %adds to the epoch count to show another has been found
    end
else %epoch hasn't been found
```

```
            signal_sift = signal_sift + 1;
             %increase countpoint to search next sample point
        end
end


%Epoch Averaging
epochs_sum = sum(epoch_array,2);
%adds each column of epoch array to create a sum
epochs_average6 = epochs_sum./ epoch_count; %averages the sum of epochs
epoch_size = size(epochs_average6,1);


%Output Processed WAV File
filename = 'goodInterferenceTest60cmCheck.wav'; % creates a name
%audiowrite(filename,epochs_average6,fs)


%plot results
displayFSPoints = epoch_size;
display_time_scale = linspace(0,0.005,displayFSPoints)';
FigHandle = figure('Position', [300, 100, 1000, 700]);
subplot(2,2,[1 2])
plot(display_time_scale, epochs_average6)

title('Interference on Artificial Patient at 60cm');
xlabel('Time(s)');
ylabel('Amplitude (FS)');
hold off

subplot(2,2,[3 4])
semilogx(display_time_scale, epochs_average6)
title('Logarithmic Processed Response');
xlabel('Time(s) from Stimulus Onset');
ylabel('Amplitude (FS)');
```

Appendix C

Hardware Circuit Diagrams

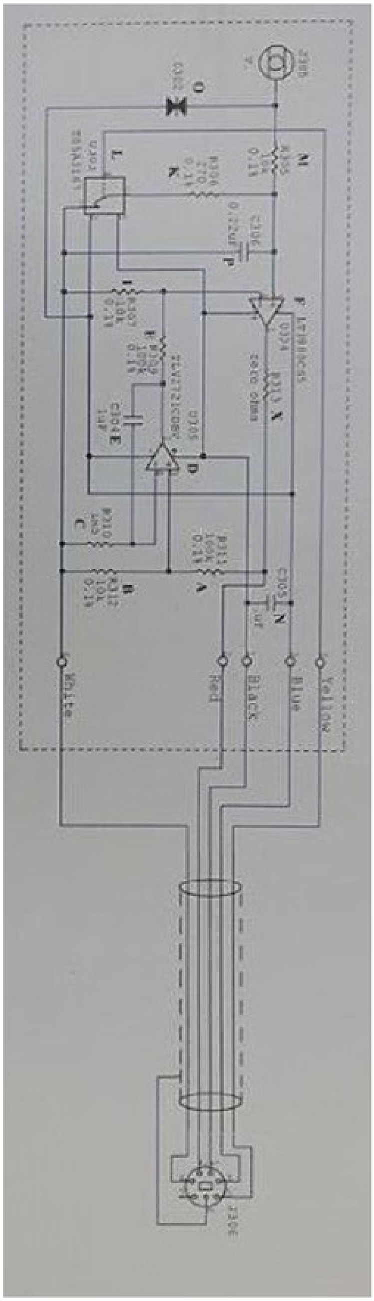
C.1 Overview

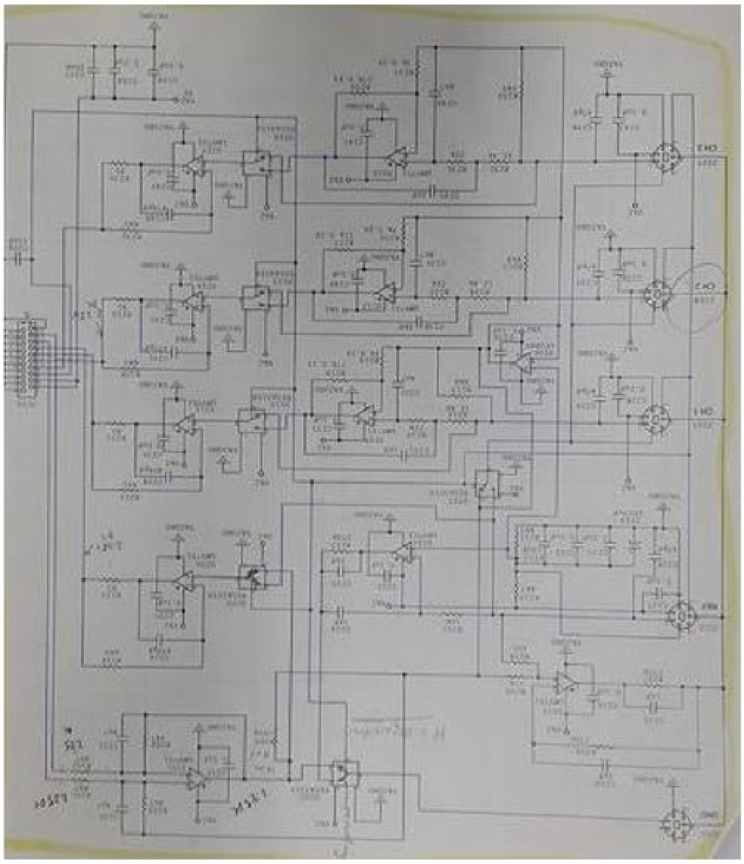
This section contains the Circuit diagrams of the integrated circuits used to develop the hardware of the prototype system.

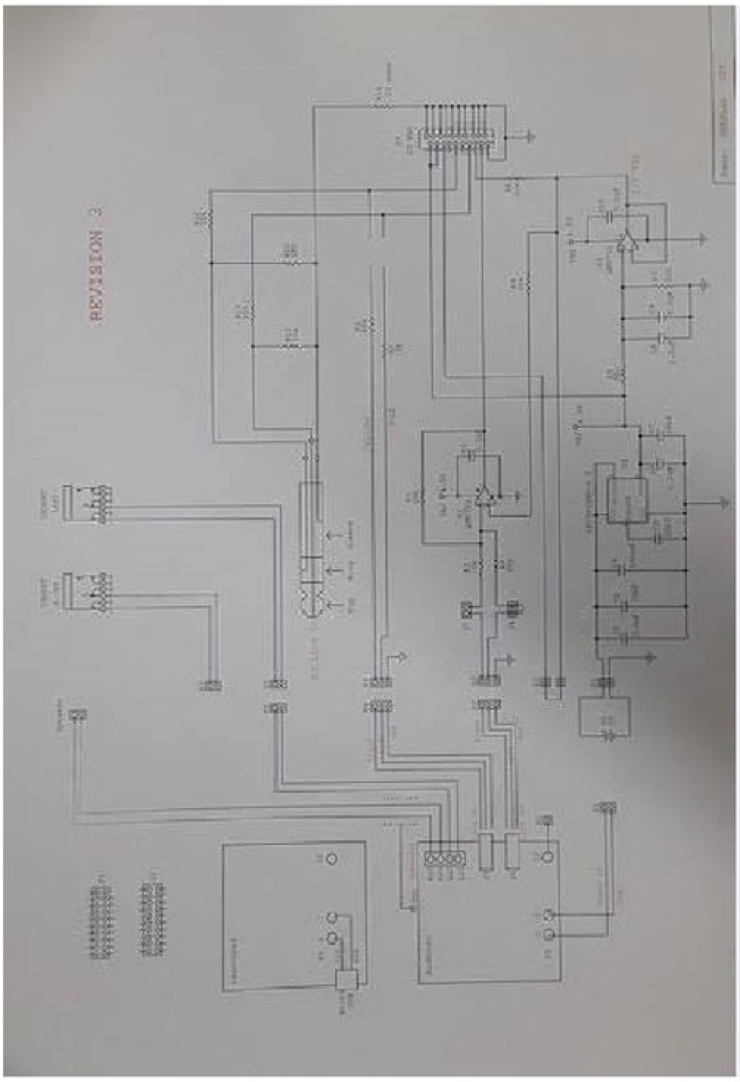
C.2 Active Electrode

C.3 Pre Amplification

C.4 Interface PCB


White
Red
Black
Blue
Yellow




REVISION 3

Appendix D

Attendance Form

D.1 Consultation Meetings and Attendance Form

Consultation Meetings Attendance Form

Week	Date	Comments (if applicable)	Student's Signature	Supervisor's Signature
-2	18/7	Project understanding & Commencement (Division of duties)	[illegible]	[illegible]
1	2/8	UDP connection issues	[illegible]	[illegible]
2	11/08	UDP connection / buffer issues	[illegible]	[illegible]
3	17/08	WIFI Audio streaming successful	[illegible]	[illegible]
4	24/08	Hardware & interface PCB explained by Barry. Teck & Barry	[illegible]	[illegible]
5	1/09	Electromagnetic interference & progress report Teck discussed	[illegible]	[illegible]
6	9/9	Meeting with Oya about progress	[illegible]	[illegible]
7	13/09	project progression & Electromagnetic Interference	[illegible]	[illegible]
Holiday Week 1	20/09	Hardware, methods for IF testing, overcoming noise issues.	[illegible]	[illegible]
Holiday Week 2	27/09	Matlab processing	[illegible]	[illegible]
8	5/10	Cross talk in hardware	[illegible]	[illegible]
9	11/10	Progress Report Disscusion	[illegible]	[illegible]
10	17/10	Matlab process	[illegible]	[illegible]
11	26/10	Matlab results	[illegible]	[illegible]
12	1/11	Report	[illegible]	[illegible]

Bibliography

[1] A. S.-L.-H. Assosiation. (2016) Effects f hearing loss on development. American Speech-Language-Hearing Assosiation. [Online]. Available: http://www.asha.org/public/hearing/Effects-of-Hearing-Loss-on-Development/

[2] F. Bardy, B. V. Dun, H. Dillon, M. Seeto, H. Qin, T. Loi, and R. Cowan, "The cortical automatic threshold estimation in adults," *The Hearing Journal*, 2016.

[3] R. F. Burkard, M. Don, and J. J. Eggermont, *Auditory Evoked Potentials Basic Principles and Clinical Applications*. Lippincott Williams & Wilkins, 2007.

[4] L. Carter, H. Dilon, J. Seymore, M. Seeto, and B. V. Dun, "Cortical auditory-evoked potentials (caeps) in adults in response to filtered speech stimuli," *American Academy of Audiology*, vol. 24, no. 9, 2013.

[5] CodeJava. (2015, July) How to play back audio in java with examples. CodeJava. [Online]. Available: http://www.codejava.net/coding/how-to-play-back-audio-in-java-with-examples

[6] B. V. Dun, L. Carter, and H. Dillon, "Sensitivity of cortical auditory evoked potential detection for hearing-impaired infants in response to short speech sounds," *Audiology Research*, vol. 2, 2012.

[7] B. V. Dun, H. Dillon, and A. Kania, "Cortical auditory evoked potentials (caeps) in (un)aided normal-hearing and hearing-impaired adults," *Seminars in Hearing*, 2015.

[8] B. V. Dun, H. Dillon, and M. Seeto, "Estimating hearing thresholds in hearing-impaired adults through objective detection of cortical aduitory evoked potentials," *American Academy of Audiology*, 2015.

[9] Eclipse. (2016) Eclipse downloads. [Online]. Available: https://www.eclipse.org/downloads/download.php?file=/oomph/epp/neon/R1/eclipse-inst-win64.exe

[10] C. Elberling and M. Don, "Qualitypotentials of averaged auditory brainstem responses," *Scandinavian Audiology*, vol. 13, 1984.

[11] P. M. Gilley, A. Sharma, M. Dorman, C. C. Finley, A. S. Panch, and K. Martin, "Minimization of cochlear implant stimulus artifact in cortical auditory evoked potentials," *Clinical Neurophysiology*, vol. 117, 2006.

[12] M. Golding, H. Dillon, J. Seymore, and L. Carter, "The detection of adult cortical auditory evoked potentials (caeps) using an automated statistic and visual detection," *International Journal of Audiology*, vol. 48, 2009.

[13] A. Hearing. (2013, December) Causes of hearing loss in australia. Australian Hearing. [Online]. Available: https://www.hearing.com.au/causes-hearing-loss-australia/

[14] K. S. Inc., *Java 6 Programming Black Book*, H. Impressions, Ed. Dreamtech Press, 2009.

[15] T. Instruments. (2015, July) Cc32xx wifi audio application. Wiki. [Online]. Available: http://processors.wiki.ti.com/index.php/CC32xx_Wifi_Audio_Application

[16] ——. (2016) Code composer studio (ccs) integrated development environment (ide). [Online]. Available: http://www.ti.com/tool/ccstudio

[17] ——. (2016) Uniflash standalone flash tool for ti microcontrollers (mcu). [Online]. Available: http://www.ti.com/tool/uniflash

[18] A. King, L. Carter, B. V. Dun, V. Zhang, W. Pearce, and T. Ching, "Australian hearing aided cortical evoked potentials protocols," 2014. [Online]. Available: https://hearlab.nal.gov.au/images/Australian%20Hearing%20CAEP%20protocols%20-%20Public.pdf

[19] C. C. Papageoriou, C. D. Hountala, A. E. Maganioti, M. A. Kyprianou, A. D. Rabavilas, G. N. Papadimitriou, and C. N. Capsalis, "Effects of wi-fi signals on the p300 component of event-related potentials during an auditory hayling task," *Integrative Neuroscience*, vol. 10, no. 2, pp. 189 – 202, 2011.

[20] T. W. Picton, *Human Auditory Evoked Potentils*. Plural Publishing, 2011.

[21] D. I. S. Pidgeon. (2014) Online audio frequency signal generator. [Online]. Available: http://www.wavtones.com/functiongenerator.php

[22] R. Rajaboina, P. C. Reddy, and R. A. Kumar, "Performance comparison of tcp, udp and tfrc in static wireless environment," *Electronics and Communication Systems*, 2015.

[23] M. Rouse and J. Sullivan. (2008, October) client/server (client/server model, client/server architecture). SearchNetworking. [Online]. Available: http://searchnetworking.techtarget.com/definition/client-server

[24] A. Sharma, M. F. Dorman, and A. J. Spahr, "A sensitive period for the development of the central auditory system in children with cochlear implants: Implications for age of implantation," *Ear and Hearing*, vol. 23, no. 6, 2002.

[25] E. Tutorials. (2015) Butterworth filter design. [Online]. Available: http://www.electronics-tutorials.ws/filter/filter_8.html

[26] T. J. Tutorials. (2015) Buffered streams. Oracle. [Online]. Available: https://docs.oracle.com/javase/tutorial/essential/io/buffers.html

[27] ——. (2015) Playing back audio. [Online]. Available: https://docs.oracle.com/javase/tutorial/sound/playing.html

[28] ——. (2015) What is a socket. Oracle. [Online]. Available: http://docs.oracle.com/javase/tutorial/networking/sockets/definition.html

[29] ——. (2015) Writing a datagram client and server. Oracle. [Online]. Available: https://docs.oracle.com/javase/tutorial/networking/datagrams/clientServer.html

[30] ——. (2016) Interface dataline. Oracle.