



Development of a Low Cost, Open-source, Electroencephalograph-Based Brain-Computer Interface

by

Ronan Byrne

This Report is submitted in partial fulfilment of the requirements of the Honours Degree in Electrical and Electronic Engineering (DT021A) of the Dublin Institute of Technology

May 28th, 2018

Supervisor: Dr Ted Burke

DECLARATION

I, the undersigned, declare that this report is entirely my own written work, except where otherwise accredited, and that it has not been submitted for a degree or other award to any other university or institution.

Signed: _____
Ronan Byrne

Date: _____
28/05/2018

Abstract

Electroencephalography (EEG), the measurement of the electrical activity of the human brain, has been performed since the 1920s and has been well researched in a clinical setting. However, it is only in recent years that consumer-grade hardware has become available to the general public. Although low in cost relative to clinical equipment, the cost of these consumer devices (as well as most self-built devices) remains substantial and presents a significant barrier to non-clinicians or non-biomedical engineers who are interested in learning, experimenting or innovating in this area. A brain-computer interface (BCI) is a system that allows a user to communicate or control devices using brain activity. These systems have also been extensively researched in a clinical setting. However, with the advent of consumer-grade EEG, it is now possible for enthusiasts who do not have access to clinical-grade equipment to experiment with BCI technology. It is hoped that increased access to recording systems may yield innovations in signal processing and interface design.

The aim of this project was to design and build a low-cost, open-source BCI, including hardware (an EEG recording system) and software (signal processing and human-machine communication). The battery-powered single-channel EEG measurement circuit comprises a protection circuit, instrumentation amplifier, gain stage, anti-alias filter, driven right leg and optoisolation circuit. Low-cost EEG electrode cups were designed and built to further reduce the cost of the system which came to a total of €30 for the circuit and €0.70 per electrode. The BCI is a virtual keyboard comprising a text box and five checker boxes, each of which flashes at a unique frequency, providing a stimulus for steady-state visually evoked potential (SSVEP). The user selects characters one at a time from a set which includes the letters of the English alphabet, ten digits and assorted special characters by looking at the checker box corresponding to the character they wish to select. Lastly, 1016 seconds of no-stimulus EEG data and 761 seconds of stimulus EEG data at each of the five SSVEP stimuli frequencies was collected. This data was used to create two statistical models of FFT magnitude at five specific target frequencies - the first models the FFT magnitude when no SSVEP stimulus is present; the second models FFT magnitude when the subject is looking at a checker stimulus which flashes at the target frequency. The use of these models in predicting which (if any) of a number of stimuli a subject is observing is also discussed.

Extensive testing was conducted on the hardware and software to validate and characterise their behaviour before the system was tested on a user. All user testing was conducted on one subject. Measurement electrodes were placed on the occipital lobe (back of the head) to measure alpha waves modulated by the user opening and closing their eyes to verify that the circuit measures valid EEG. The user was then presented with a single large flashing checker box to act as an SSVEP stimulus which the circuit was also able to measure. The 1016 seconds of no-stimulus data and 761 seconds of stimulus data were used to fit probability density functions (PDFs) to the histograms of the five SSVEP frequencies used in the BCI. The mean FFT magnitude observed at the target frequency when the flashing checker stimulus was present was greater than the no-stimulus data and there was some separation in the no-stimulus and stimulus PDFs meaning that a statistical method for SSVEP could be developed. This was not implemented or tested, but is discussed here.

Acknowledgements

I would like to thank my project supervisor Dr Ted Burke for his continuous help, guidance and support throughout the project. I would also like to thank the technical staff of DIT Kevin Street for their valuable assistance throughout the project. Lastly, I'd like to thank my friends, family, classmates and DIT staff for their support throughout my time in DIT.

Table of Contents

1	Introduction	1
1.1	Background.....	1
1.2	Aims and objectives.....	1
1.3	Project overview	2
1.4	Ethics	3
1.4.1	Complying with standards.....	3
1.4.2	Consent.....	4
1.4.3	Open-source license	4
1.4.4	Effects of long-term use	4
1.4.5	Consequences of errors in user communication.....	4
1.4.6	Misuse of system	5
2	Literature Review	5
2.1	Existing work.....	5
2.1.1	Current human-computer interfaces	5
2.2	Theory behind the proposed device.....	7
2.2.1	Electroencephalography	7
2.2.2	Biopotential electrodes	7
2.2.3	Steady-state visually evoked potentials.....	9
2.2.4	EEG frequency bands.....	10
2.2.5	Noise and interference during measurement	10
2.2.6	EEG measurement circuit.....	11
2.2.7	Feature extraction and probability distributions.....	13
3	Design and implementation.....	15
3.1	Circuit design.....	16
3.1.1	Powering the circuit	18
3.1.2	Protection circuit	18
3.1.3	Instrumentation amplifier	20
3.1.4	Gain stage and alias filter	21
3.1.5	Driven right leg circuit	22
3.1.6	CMRR of circuit.....	23
3.1.7	Frequency response of the circuit.....	25
3.1.8	Communication between microcontroller and PC	26
3.1.9	Electrodes	28

3.2	Software.....	29
3.2.1	Microcontroller.....	29
3.2.2	PC	30
4	Testing and results.....	34
4.1	Testing EEG measuring circuit with ECG	34
4.2	Measuring alpha waves	35
4.3	Alpha wave BCI	36
4.4	Fitting probability density functions.....	37
5	Conclusion.....	42
5.1	Summary.....	42
5.2	Project limitations.....	43
5.2.1	Limitation of data and PDFs collected	43
5.2.2	Possible issue with classifier suggested	43
5.2.3	Number of SSVEP stimuli and harmonics	43
5.2.4	Holding the electrodes in place	43
5.3	Future work.....	43
6	Bibliography.....	45
7	Appendix	47
7.1	Appendix A - Bill of materials	47
7.2	Appendix B - Full circuit.....	47
7.3	Appendix C - Simulated frequency responses.....	49
7.4	Appendix D - Nucleo code flow diagram.....	50
7.5	Appendix E – BCI code flow diagrams	50
7.6	Appendix F – BCI code	52
7.6.1	eegScope.py.....	52
7.6.2	eegInterface.py	55
7.7	Appendix G – Histograms and probability density functions	60

Table of Figures

Figure 1	User wearing the EEG measuring circuit.....	2
Figure 2	(a) Window showing real-time EEG plot and real-time FFT of EEG plot. (b) BCI user interface, top level.....	3
Figure 3	Apache License 2.0 summary of “Low-Cost-EEG-Based-BCI” Github repository	4
Figure 4	An example of the stimulation frequency arrangements generated assuming 30 frequencies ranging from 5 Hz to 7.9 Hz with a span of 0.1 Hz	6
Figure 5	10-20 Electrode placement system	7

Figure 6 (a) Bipolar electrode placement (O2-O1). (b) Unipolar electrode placement (Oz-left ear).	8
Figure 7 (a) Wet EEG cup electrodes - Images from Kandel Medical[21]. (b) Dry EEG electrodes - Image from Guger Technologies [22]	8
Figure 8 Different areas of the brain – Image from the Mayfield Clinic[1].....	9
Figure 9 (Blue) Frequency spectrum of EEG without stimulus. (Red) Frequency spectrum of EEG with 15Hz stimulus showing 2nd and 3rd harmonics	9
Figure 10: EEG frequency bands and associated mental states – Image from M.G.N.Garcia et al [2]	10
Figure 11 Example protection circuit	12
Figure 12 Biomeasurement circuit utilising a driven right leg circuit.....	13
Figure 13 Two representative histograms of the envelope and probability distributions.....	14
Figure 14 Block diagram for SSVEP-based BCI for character selection on a virtual keyboard	16
Figure 15 Block diagram of EEG measurement circuit connected to the user.....	17
Figure 16 EEG measurement circuit and optoisolation circuit on a breadboard.....	17
Figure 17 Connections to Nucleo	18
Figure 18 Circuit diagram of the protection circuit.....	19
Figure 19 (a) Input (yellow) and output (blue) of protection circuit	19
Figure 20 Circuit diagram of the instrumentation amplifier.....	20
Figure 21 Circuit diagram for gain stage and alias filter.....	22
Figure 22 Circuit diagram for the driven right leg.	23
Figure 23 (a) Bitscope Micros ground connected to Vref, generating a differential signal into protection circuit.	24
Figure 24 (a) Bitscope Micros ground connected to Vref, generating a common mode signal into protection circuit.	24
Figure 25 Frequency response of circuit output (simulated and measured).....	26
Figure 26 Frequency response of DRL (simulated and measured)	26
Figure 27 Circuit diagram for optoisolating UART lines.	27
Figure 28 (a) Traces of input (analog-yellow, logic-pink) and output (analog-green, logic-blue) of the PC817 using input resistor of 220Ω and output resistor of (a) 220Ω . (b) 120Ω	27
Figure 29 (a) Traces from input (analog-yellow, logic-pink) and output (analog-green, logic-blue) of PC817 with input from Nucleo’s UART Tx pin operating at (a) 57600 baud. (b) 119200 baud.	28
Figure 30 (a) Left: ECG electrode pad. Right: Homemade EEG cup electrodes.	29
Figure 31 Real-time plot from an input signal of 20 Hz	30
Figure 32 BCI user Interface (top level).....	31
Figure 33 BCI user interface selection tree	31
Figure 34 Square produced from measuring 10 Hz checker box flashing using phototransistor....	32
Figure 35 Einthoven’s triangle configurations for measuring ECG.....	34
Figure 36 ECG measurement	35
Figure 37 ECG measurement using low-cost electrodes and DRL.	35
Figure 38 Time domain and spectrogram of EEG measured between O1 and O2.....	36
Figure 39 Demonstrating a rudimentary alpha wave BCI.....	37
Figure 40 Single large flashing checker box used for triggering SSVEPs.....	38
Figure 41 (a) FFT of EEG with no-stimulus. (b) FFT of EEG with 5 Hz stimulus.	39
Figure 42 Normalised FFT magnitude histogram of EEG at 12 Hz (no-stimulus) with fitted PDFs.....	40
Figure 43 Normalised FFT magnitude histogram of EEG at 12 Hz (12 Hz stimulus) with fitted PDFs	40
Figure 44 Normalised probability density functions of the FFT magnitudes of EEG at 5 Hz	41
Figure 45 LTspice EEG measurement circuit diagram	48
Figure 46 LTspice diagram of optoisolation circuit.....	49
Figure 47 Simulated frequency response of circuit output from a differential input of $50 \mu V$	49

Figure 48 Simulated frequency response of driven right leg from a common mode input of 200 mV	49
Figure 49 Microcontroller code flow diagram	50
Figure 50 eegScope.py code flow diagram	51
Figure 51 eegInterface.py code flow diagram	52
Figure 52 Histograms of normalised FFT magnitudes with no stimuli, overlaid with multiple fitted probability density functions	61
Figure 53 Histograms of normalised FFT magnitudes with stimuli, overlaid with multiple fitted probability density functions	62
Figure 54 Normalised probability density functions of the FFT magnitudes with no stimuli (black) and with stimuli (blue)	63

Table of Tables

Table 1 Table of theoretical input voltage specifications for the circuit	18
Table 2 CMRR of the circuit without the DRL and with the DRL at 10 Hz, 50 Hz and 100 Hz.....	24
Table 3 Table of normalised means and standard deviations for no-stimulus and stimulus histograms	41
Table 4 Bill of materials (circuit)	47
Table 5 Bill of Materials (Electrode)	47

Table of Acronyms

Acronym	Definition
ADC	Analog-to-digital converter
BCI	Brain-computer interface
CMRR	Common mode rejection ratio
DRL	Driven right leg
EEG	Electroencephalography
FFT	Fast Fourier Transform
In-amp	Instrumentation amplifier
MLE	Maximum likelihood estimate
PDF	Probability density function
SSVEP	Steady-state visually evoked potentials

1 Introduction

1.1 Background

The *electroencephalograph* (EEG) was first recorded in 1924 [3] and has been researched thoroughly since then but has only been applied to non-medical and commercial uses, such as *brain-computer interfaces* (BCIs), in recent decades [4]. The cost of these commercially available EEG recording systems ranges from \$100-\$1000 for a consumer-grade EEG headset up to \$25,000 or more if buying a for a high-end clinical system [5]. Even the low-end cost of \$100 may pose an obstacle to some members of the maker community who are interested in recording EEG. The OpenEEG project, a distributor of open-source EEG hardware states that their “ModularEEG” will cost \$200-\$400 to build [6]. OpenBCI is a distributor of hardware and open-source software and their cheapest EEG hardware costs \$199 [7]. This high cost hinders the possible adoption of this hardware within the maker community, reducing the likelihood of innovation in the area from this community.

Open-source hardware and software provides developers and designers with a degree of flexibility and self-determination rarely afforded by proprietary offerings. Users of open-source systems have the option to modify or refine those systems themselves. Users of a proprietary system, on the other hand, are typically subject to the commercial priorities of the company who own and develop it. When fixing or adding features to a product is not seen as likely to benefit the company in terms of profit, they are unlikely to allocate resources to that work.[8, 9].

BCIs are a less mature technology than general purpose EEG recording systems and were first researched to be used as assistive devices for disabled people. In recent years however, BCIs have begun to be used by able-bodied people in computer games and other leisure activities[10]. By further increasing the accessibility of EEG measurement systems and BCI software to the maker community through low-cost, open-source designs, more applications and further innovation in these areas may be seen in the future.

1.2 Aims and objectives

The aim of the project was to develop and test a low-cost EEG-based BCI system which can stimulate, measure and then react to particular EEG signals - specifically, *steady-state visually evoked potentials* (SSVEP). As the amplitude of an EEG is in the range of microvolts [11], the circuit also has to isolate the EEG signal from the many types of interference which could distort or obscure the EEG signal. The cost target for the entire system was set at €50, half the price of the lowest cost EEG headset found online [5]. A secondary objective was to make the EEG measurement circuit flexible in terms of the type of EEG signal it could measure, i.e. that it not be able to measure SSVEPs in the range of 5-20 Hz but should also be able to measure other biopotentials such as *electrocardiography* (ECG), *electrooculography* (EOG) and *electromyography* (EMG). This makes the circuit an all-purpose bioamplifier with some alterations.

The software developed in this project is able to stimulate SSVEPs in the user which can be measured by the circuit. Multiple SSVEP stimuli will be presented to the user to allow them to make choices and navigate the BCI. The software must therefore, to

detect the presence of an SSVEP at one of a number of pre-defined stimulus frequencies in real time and react to them. This was be done by developing a statistical method for estimating the probability the user is looking at one of the stimuli. Lastly, the circuit and software was be disseminated online to provide a reference design for the purpose of other to learn, modify and create new hardware/software.

To summarise, the objectives of the project were to:

- Develop a low-cost circuit capable of EEG measurement while limiting interference.
- Develop a brain-computer interface which can stimulate and react to SSVEPs.
- Develop a statistical method for SSVEP detection.
- Disseminate the hardware designs and source code under an open-source license.

1.3 Project overview

A circuit was designed to measure the EEG (specifically SSVEPs) from a user while limiting the amount of interference during measurement. Low-cost EEG electrode cups were also designed and built to reduce the cost of the whole system. The circuit has a measured passband of ~5-75 Hz and a measured *common mode rejection ratio* (CMRR) of 89 dB at 50 Hz. The amplified EEG signal was read into a microcontroller and transmitted to a PC through optoisolated serial lines.

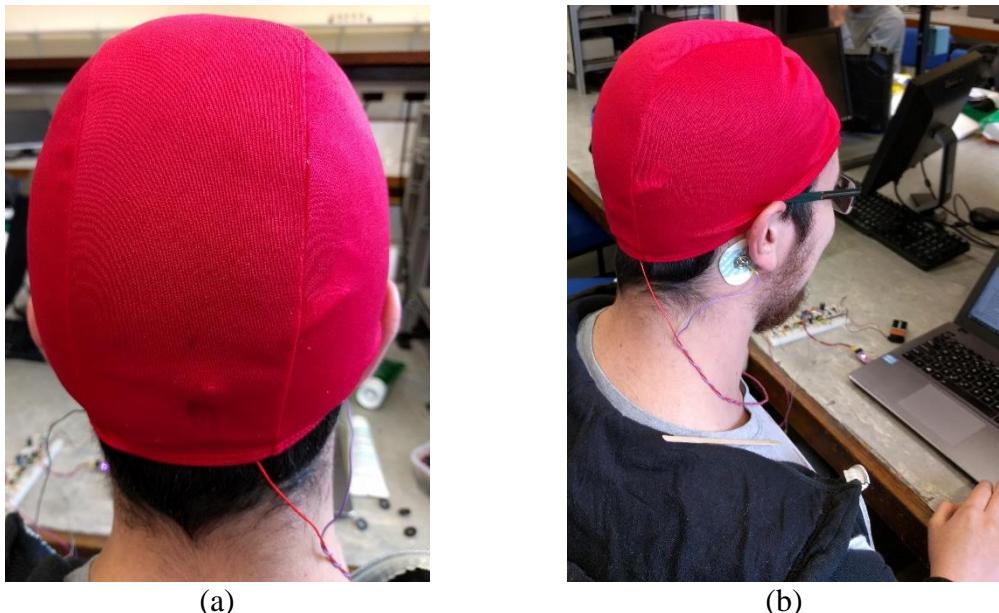
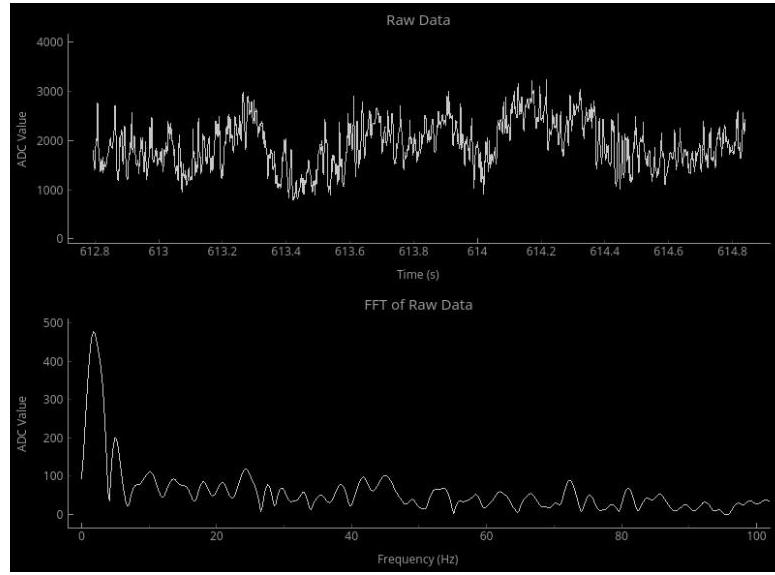
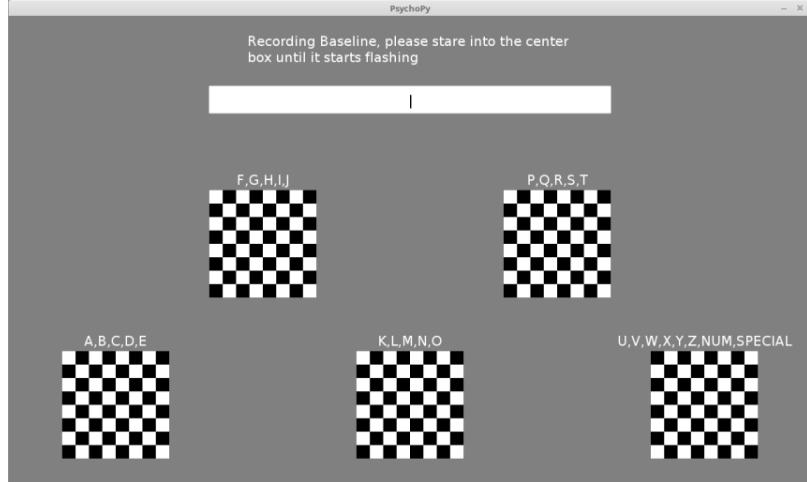


Figure 1 User wearing the EEG measuring circuit. The positive electrode using low-cost EEG cup-Oz, the negative electrode using ECG pad-right mastoid and the driven right leg electrode using ECG pad-left mastoid. (a) Back of head view. (b) Right side view.

The PC ran a Python script which displays two main windows, the first showing the real-time EEG and real-time FFT of each second of data, and the second showing a virtual keyboard consisting of five SSVEP stimuli (flashing checker boxes) and a text box. Character options were shown above each of the checker boxes. The user selects a character (or character group) by looking at the associated checker box. Each selected character is added to the text box, allowing a message to be spelled out.



(a)



(b)

Figure 2 (a) Window showing real-time EEG plot and real-time FFT of EEG plot. (b) BCI user interface, top level.

A large dataset of FFT magnitudes of the EEG signal from one user when they were and were not looking at an SSVEP stimulus was collected to develop a statistical method for SSVEP detection. This method was not fully implemented or tested but the results from the large data suggest it may be possible to detect SSVEPs using this method. Finally, the hardware designs and source code were published to Github¹ under the Apache licence 2.0. Links to videos demonstrating the project² and a rudimentary alpha wave BCI³ can be found in the footnotes.

1.4 Ethics

1.4.1 Complying with standards

The ethical issues for any engineering project should always be considered but in the case of a project that involves a physical interface between an electrical system and a user's body, safety is of paramount importance. Fortunately, in this case, the nature of

¹ Link to GitHub repository: <https://github.com/RonanB96/Low-Cost-EEG-Based-BCI>

² Link to video demonstrating circuit and software: https://youtu.be/Ilv_VNvS42w

³ Link to video demonstrating alpha wave BCI in operation: https://youtu.be/Ehdn_71upWc

the interface is such that certain simple controls (described below) can be put in place to eliminate potential risks to the user. The device should comply with ASTM F9836 standard consumer safety specification for toy safety [12] and the directive 2014/53/EU for making available on the market of radio equipment [13]. Similar devices such as the “The Force Trainer II: Hologram Experience” from NeuroSky have had to comply with these regulations [14]. The type of system proposed at the outset of this project (and subsequently implemented) is not a medical device and therefore does not need to comply with IEC 60601 for medical electrical equipment. However, it was deemed to prudent to review what that standard requires in terms of basic safety and performance for similar medical devices.[15].

1.4.2 Consent

The system was not tested on subjects other than the author, but had the system been tested on a group of willing participants, each participant would have needed to sign a detailed consent form describing the test which they were to undertake and that their subsequent test results would be anonymised and may have been published online.

1.4.3 Open-source license

The hardware designs and source code of the BCI system were published to Github under the Apache licence 2.0. This allows people to distribute and/or modify the designs freely but requires them to state any modifications and licence it under the Apache licence 2.0 [16].

RonanB96/Low-Cost-EEG-Based-BCI is licensed under the Apache License 2.0	Permissions	Limitations	Conditions
 A permissive license whose main conditions require preservation of copyright and license notices. Contributors provide an express grant of patent rights. Licensed works, modifications, and larger works may be distributed under different terms and without source code.	<ul style="list-style-type: none"> ✓ Commercial use ✓ Modification ✓ Distribution ✓ Patent use ✓ Private use 	<ul style="list-style-type: none"> ✗ Trademark use ✗ Liability ✗ Warranty 	<ul style="list-style-type: none"> ⓘ License and copyright notice ⓘ State changes

Figure 3 Apache License 2.0 summary of “Low-Cost-EEG-Based-BCI” Github repository

1.4.4 Effects of long-term use

Due to the short duration of this project, it was not possible to assess the potential effects of long-term use of the system. However, this is an issue which others should consider if making use of the published designs in the future or using the system on an ongoing basis. While there is no reason to expect that short-term use of the system will pose any threat to the user, long-term use of any user interface can have unexpected negative consequences (e.g. eye strain from long-term use of BCI). The Apache licence does protect the author from liability and does not provide a warranty.

1.4.5 Consequences of errors in user communication

In case others make use of the design to facilitate communication in a situation where miscommunication could result in significant negative consequences, it is important that the published designs be accompanied by clear guidelines on the reliability of the system and a realistic indication of the expected error rate. As the system is not yet fully functional, an expected error rate cannot yet be estimated. To give a sense of the potentially grave consequences of communication errors in a BCI, a study was

conducted by *U.Chaudhary et al* [17], tested the ability to communicate of completely locked-in syndrome(CLIS) patients using a BCI system. Once communication was deemed reliable, the four patients in the study were asked many yes or no type questions, one being if they wished to sustain their lives through artificial ventilation, all four communicated yes. Had an error-prone BCI system been used for this task, and had the people operating the system not been aware of the likelihood of errors, the doctors could have fulfilled the “perceived” request of the patient to not use the artificial ventilation and led to the patient’s death if the BCI system interpreted the signals incorrectly.

1.4.6 Misuse of system

If this device facilitated communication for someone who cannot currently communicate (e.g. due to paralysis), there exists the potential for that person to subsequently engage in problematic activities (e.g. writing threatening or abusive messages on the virtual keyboard) which would otherwise not have been possible. Realistically, however, since the system will not facilitate communication beyond what able-bodied people already enjoy, it is the author’s position that what the user communicates through the system is his/her own ethical responsibility, rather than that of the system designer.

2 Literature Review

2.1 Existing work

2.1.1 Current human-computer interfaces

Human-computer interfaces (HCIs) are systems (hardware and software) which take inputs from a human user and convert them into actions through software. HCI is a term which covers a large variety of technologies but this section will focus on existing or proposed HCIs which are controlled by the user’s brain activity.

2.1.1.1 Brain-computer interface

Brain-computer interfaces (BCIs) are systems which allow communication between a user and a computer through their brain activity. There are different type of BCI, which use different methods to observe their user’s brain activity:

- EEG-based, such as this project.
- Magnetoencephalography (MEG) based.
- Electrocorticography (ECoG) based.
- Functional magnetic resonance imaging (fMRI) based.
- Near-infrared spectroscopy (NIRS) based.

EEG is non-invasive and measures the electrical activity produced using the brain by electrode placed on the surface of the scalp. EEG will be discussed in greater detail in a later chapter. ECoG is invasive and places the electrodes on the surface of the brain to measure the electrical activity produced. ECoG can give higher temporal and spatial resolution compared to EEG but as it is invasive so is outside the scope of this project. MEG, fMRI and NIRS are all non-invasive but were also outside the scope of the project due to the cost equipment but will be discussed briefly. MEG measures the

magnetic disturbances caused by neural activity. fMRI applies electromagnetic fields around the head and, in BCI systems, typically measures the blood oxygen level dependent (BOLD) during neural activity. Lastly, NIRS uses infrared light when can penetrate the skull by a few centimetres and measures the light reflected. The light reflected which varies depending on the oxyhemoglobin and deoxyhemoglobin concentrations. All of the above descriptions can be found in more detail in [18].

Returning to EEG-based BCIs, various EEG signals have been used to control a computer such as visually evoked potentials (VEP), slow cortical potentials (SCPs), P300 evoked potentials and sensorimotor rhythms (SMR). BCIs are more commonly based on VEPs and SMRs and will be discussed below.

VEPs are electrical responses in the occipital lobe (back of the head), produced as a direct response to a visual stimulus. A fixed-frequency flashing stimulus causes an ongoing oscillatory VEP, which is referred to as a steady-state visually evoked potential (SSVEP). This project focuses on an SSVEP-based BCI and SSVEPs will be discussed in more detail in a later section. An example of an SSVEP-based BCI is the QWERTY style LED keyboard created by *H. Han-Jeong et al* [19]. Each key on the keyboard has an LED behind it, flashing at a different frequency and the key the user is looking at can be discerned from looking at the FFT of the user occipital EEG.

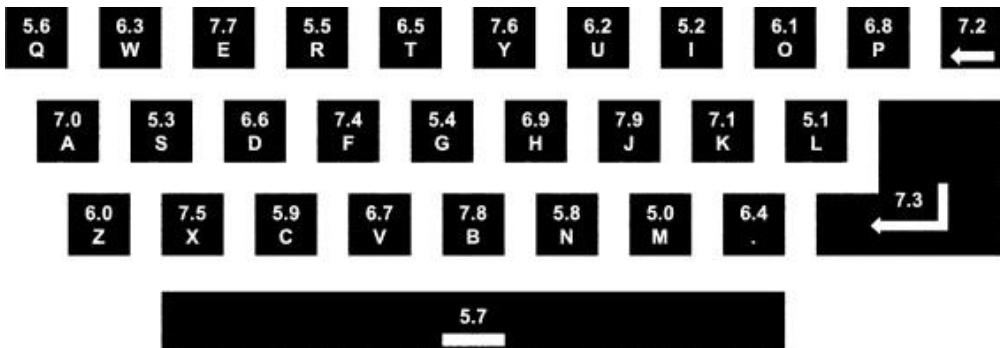


Figure 4 An example of the stimulation frequency arrangements generated assuming 30 frequencies ranging from 5 Hz to 7.9 Hz with a span of 0.1 Hz - Image from *H. Han-Jeong et al*[19]

Sensorimotor rhythms (SMRs) or mu (7-13 Hz) and beta (13- 30 Hz) rhythms are oscillations located in the sensorimotor cortex within the motor strip (see Figure 8 for the location of the motor strip) which correspond to motor tasks or motor planning. Increased SMR indicates a person is more relaxed and not moving; decreased SMR means the person is moving or planning to move [18]. SMR-based BCIs have been used to control object 1-D, 2-D and 3-D virtual environments such as a virtual drone [10]. Two sets of electrodes are usually used in SMR-based BCIs which are placed on the left and right side of the motor strip respectively to measure the activity associated with the motor tasks on either side of the body. This allows the user to perform four separate actions to get a response from the BCI:

- Don't imagine a motor task (no SMR activity reduction)
- Imagine doing a motor task on the left side of the body (SMR reduction on right motor strip)
- Imagine doing a motor task on the right side of the body (SMR activity reduction on left motor strip)

- Imagine doing motor tasks on both sides of the body (SMR activity reduction on both of motor strips)

This allows for a maximum of four options which is usually reduced to three if no SMR activity reduction is interpreted as the “do nothing” state. To achieve good performance, the user must train themselves to perform these motor imagery tasks[10] which may take weeks to months of training sessions [18]. SSVEP-based BCIs, by contrast, require no training, but are limited by the number of detectable stimuli that fit in the user’s field of view, making them more suitable for untrained users. It was for this reason that SSVEP-based BCI was chosen as the focus of this project.

2.2 Theory behind the proposed device

2.2.1 Electroencephalography

An electroencephalogram (EEG) recording system is a non-invasive measuring system which records the electrical activity produced by the brain using electrodes attached to the scalp. It can be used for a variety of purposes such as assessing a person’s alertness, monitoring their reaction to external stimuli, detecting and/or diagnosing sleep and other neurological disorders and many more. However, in recent decades [20] the idea of combining the signals produced from the EEG recording system with external devices led to the building of BCI systems which use the EEG signals to control a computer.

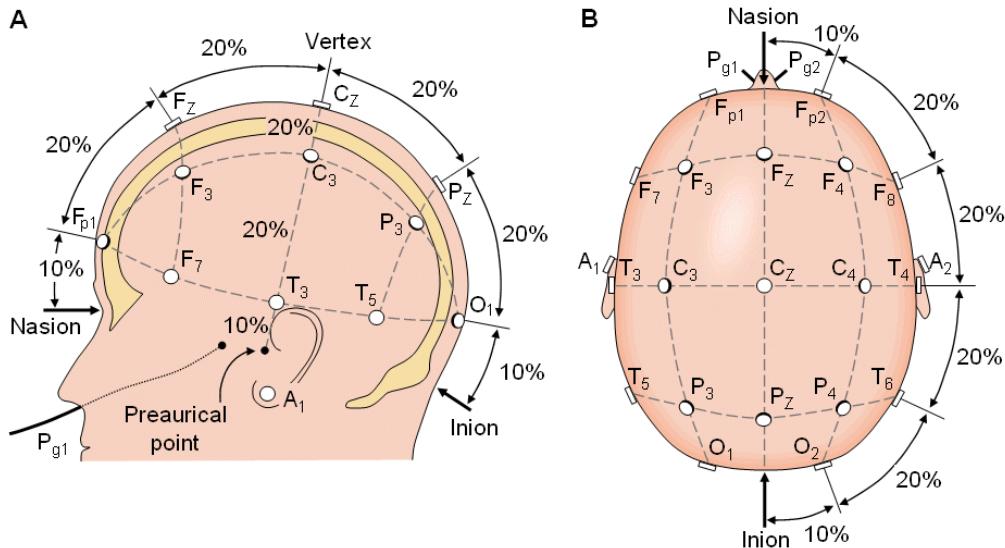


Figure 5 10-20 Electrode placement system seen from (A) left and (B) above the head – Image from J. Malmivuo and R. Plonsey [11]

The signals measured by an EEG recording system depend on the placement of the biopotential electrodes on the scalp as well as the current brain activity of the user. This is because each section of the brain is responsible for different functions. The most widely used standard for electrode placement is the 10-20 system (Figure 5) [11].

2.2.2 Biopotential electrodes

Biopotential electrodes are used to measure the electrical potentials produced by living tissue, which originate in activatable cells such as neurons or muscle cells. In EEG, each

channel uses a pair of biopotential electrodes to measure the time-varying difference in electrical potential between two points on the scalp (bipolar configuration) or between one point on the scalp and a reference voltage such as the ear (unipolar configuration) [11].

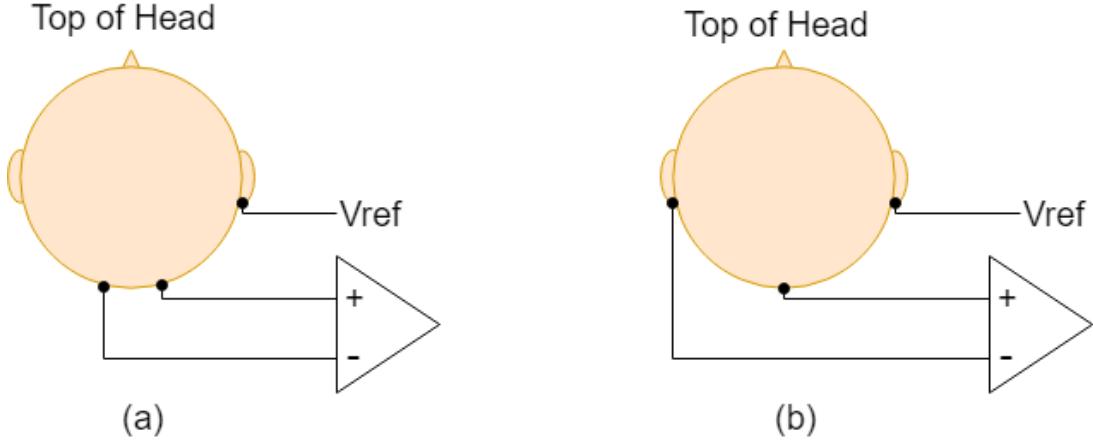


Figure 6 (a) Bipolar electrode placement (O2-O1). (b) Unipolar electrode placement (Oz-left ear)

Biopotential electrodes can be divided into two main categories: *wet* or *dry*. Wet EEG electrodes are usually cup-shaped and filled with a conductive gel to make a connection to the scalp through the hair. Dry electrodes do not use any gel but have teeth which protrude through the hair to make the connection.



Figure 7 (a) Wet EEG cup electrodes - Images from Kandel Medical[21]. (b) Dry EEG electrodes - Image from Guger Technologies [22]

Wet electrodes are often less comfortable if worn for long periods of time due to the gel drying which needs to be scrubbed as it degrades signal quality. This can leave the skin more sensitive, leading to discomfort for the user. Dry electrodes can be more comfortable and require less maintenance but are to be more susceptible to motion artefacts and increased electrode-skin impedance [23]. Different materials have different properties. Common electrode types include silver/silver chloride, gold, silver, stainless steel and tin. Reusable EEG electrodes can cost \$10 or more per lead [24]. This is why low-cost, homemade electrodes were used in this project.

Another useful classification of electrodes is *active* or *passive*. Active electrodes use a buffer amplifier within or very close to the electrode which mitigates the effect of high and/or variable electrode-skin impedance as well as reducing the effect of motion artefacts and cable movement [25]. Passive electrodes do not have this buffer amplifier and are simply connected to the bio-amplifier via a long wire.

2.2.3 Steady-state visually evoked potentials

As mentioned above, different parts of the brain have different functions, meaning that electrodes have to be located appropriately to detect specific activity. A diagram of the different parts of the brain can be seen in Figure 8. The occipital lobe, shown in pink, is the region of interest for this project as the proposed BCI harnesses signals originating in this area.

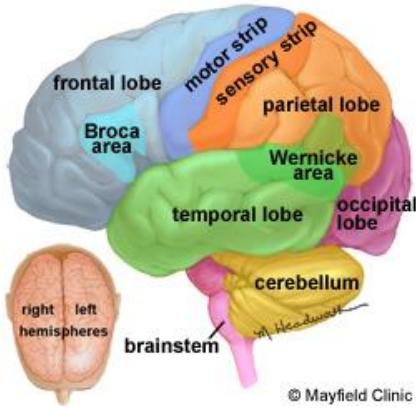


Figure 8 Different areas of the brain – Image from the Mayfield Clinic[1]

The occipital lobe is responsible for processing visual information [26] and an electrical response can be triggered in this area by flashing a light at a constant frequency in the person's field of view [27]. The fundamental frequency of the flashing light along with its harmonics will be seen in the frequency spectrum of the EEG in this region. This electrical response to the flashing light is referred to as a steady-state visually evoked potential (SSVEP) [27].

Two examples of flashing stimuli which have been proven to produce SSVEPs are flashing LEDs and a periodically reversing checker box on a computer screen [18]. When a reversing checker box stimulus is used, the observed frequency of the SSVEP is the frequency of *reversal* of the stimulus [27].

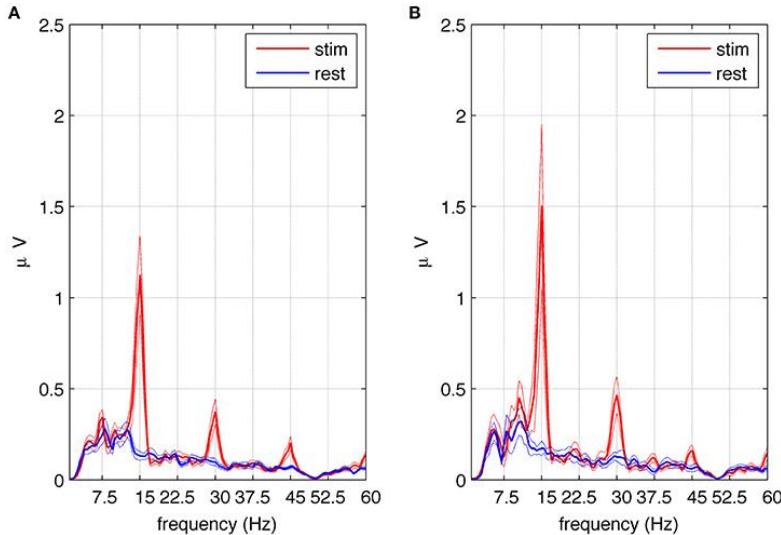


Figure 9 (Blue) Frequency spectrum of EEG without stimulus. (Red) Frequency spectrum of EEG with 15Hz stimulus showing 2nd and 3rd harmonics – Image From L. Maciej et al. [28]
To avoid distortion, the frequency of a periodic flashing stimulus displayed on a computer screen must divide evenly into the refresh rate of the display. This limits the number of stimuli and therefore the number of options on an interactive computer-based BCI. As harmonics of the fundamental stimulus frequency will be present in the frequency spectrum of the EEG, this will interfere with any stimuli which share a fundamental frequency with the harmonics of another. For example, a 60 Hz screen can display the following frequencies: 60, 30, 20, 15, 12, 10, 6, 5, 3, 2, 1 Hz. Having five flashing stimuli which will not interfere with each other is not possible in this case. But this can be overcome but does add more complexity to the BCI

compared to an LED stimulus, which is not subject to the same limitation and can in principle be delivered at any frequency in the SSVEP range.

SSVEPs provide a good basis for an BCI because they require no training, have been shown to achieve high communication rates (>60 bits/min) and allow a larger number of choices compared to other EEG signals using just one channel [18]. *Kelly et al.*[28] found that if a subject is presented with two stimuli but does not look directly at either, the user can increase the amplitude of response to one or other stimulus without moving their eyes simply by concentrating his/her attention on it. This means that SSVEP-based BCIs can potentially be used by those who cannot move their eyes or head.

2.2.4 EEG frequency bands

SSVEPs are event-related potentials (ERPs), which are electrical responses in the brain that are produced in response to external stimuli. Non-ERP related activity can be separated into different frequency bands, each reflecting different elements of a subject's the user's current mental state. frequency bands into which spontaneous EEG is conventionally divided are summarised in Figure 10 [2].

Alpha waves are of interest in this report because their magnitude

increase significantly when the eyes are closed. This can be used to quickly validate if an EEG measurement circuit is functional as well as if the electrodes are making a good connection [29]. Alpha waves can be measured throughout the brain but are observed particularly strongly in the occipital lobe when the eyes are closed [18]. Alpha waves in the frontal lobe have been used to gauge a user's attention as these frequencies tend to be suppressed when the user is actively concentrating on a task [18] but because of its binary nature, alpha activity is not suitable for use in a complex, interactive BCIs.

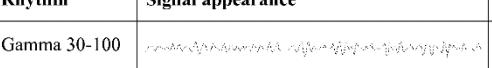
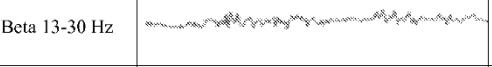
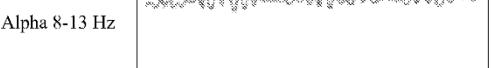
Rhythm	Signal appearance	Main behavioral trait
Gamma 30-100 Hz		Represents binding of different populations of neurons for the purpose of carrying out a certain
Beta 13-30 Hz		Usual waking rhythm associated with active thinking and active
Alpha 8-13 Hz		It is usually found over the occipital regions. Indicates relaxed awareness without attention or
Theta 4-8 Hz		Theta waves appear as consciousness slips towards drowsiness. Theta increases have
Delta 1-4 Hz		Primarily associated with deep (slow) wave sleep.

Figure 10: EEG frequency bands and associated mental states – Image from M.G.N.Garcia et al [2]

2.2.5 Noise and interference during measurement

The accurate measurement of a user's EEG may prove difficult due to a number of factors such as making a good connection to the scalp with the electrode, movement of the electrode, interference from mains power lines, interference from muscle activity or other unrelated brain activity. A few of these issues will be discussed in this section.

Electrode-skin impedance is the variable impedance between the electrode and the skin. If it is too high, it can cause distortion and attenuation of the already small EEG signals. It can vary due to the amount of dead skin and sweat between the electrode and the scalp, movement of the electrode, the type of electrode used and the type of conductive gel used (if any). As an EEG measurement requires at least two measurement

electrodes, the difference in impedance between them may cause the common mode rejection ratio (CMRR) of the instrumentation amplifier (in-amp) to be decreased [30] resulting in the undesired amplification of signals common to both electrodes. Abrading the skin to remove dead skin and oils and following proper electrode mounting procedure can reduce the impedance.

Motion artefacts are electrical interference caused by the movement of the electrode and/or wires. The magnitude of the electrical interference can be many times that of the EEG which can be less than $100 \mu\text{V}$ [29] while the electrical signals produced from muscle movement can be tens of millivolts which can completely obscure the signals of interest [31].

50 Hz (or 60 Hz) interference is ordinarily present in EEG recordings, due to *electromagnetic interference* (EMI) from nearby main electrical lines. The magnitude of the interference can be larger by orders of magnitude than the EEG signal. Many biopotential amplifiers reduce this interference through the use of a *driven right leg* (DRL) circuit or filters (analog and/or digital). The DRL circuit actively cancels the common mode signal measured in the circuit, this will be discussed in more detail in a later section. *Bulent et al.* [32] suggest a CMRR greater than 80 dB is a necessity for biosignal acquisition.

Electrode half-cell potentials appear as a slowly varying series DC voltage on the electrode due to the material and temperature of the electrode and the conductive gel (if used). This voltage can be hundreds of millivolts [33], orders of magnitude larger than the EEG signal. If amplified, these half-cell potentials could saturate the output of the amplifiers.

2.2.6 EEG measurement circuit

Most bioamplifier circuits can be split into four parts: protection circuit, in-amp, gain stage/s and mains interference rejection. An overview of the conventional designs and how they function will be discussed in this section.

The protection circuit is used to limit the current to and from the user in the event of a fault. The simplest protection circuit is just a resistor in series with the electrode entering the in-amp, this will limit the current but will not protect the amplifier from voltages outside its stated range. Voltage clamping can be used to protect the amplifier from overvoltages by the use of diodes. Two diodes can be connected to ground from each signal input, one in reverse bias and one in forward bias. The diodes will clamp the signal if their threshold voltage is exceeded, but will act as open circuits otherwise.

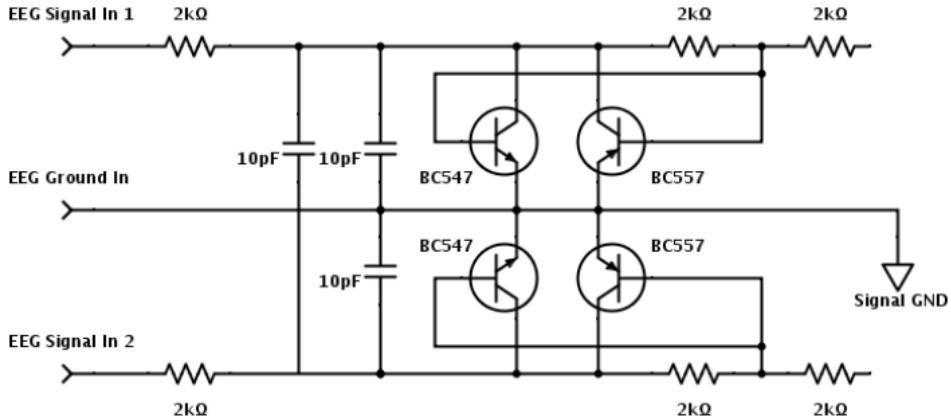


Figure 11 Example protection circuit – Image from B. Fong et al [34]

The circuit shown in Figure 11 uses transistors instead of diodes but they serve the same purpose. It is also common [29, 34, 35] to have a high cut-off, low pass filter on the input to the protection circuit that filters *radio frequency interference* (RFI) which the circuit may pick up. In the above circuit, there is a low pass, common mode filter on each signal line and a low pass, differential filter shared by both lines. If the cut-off frequency of the differential filter is chosen to be at least a decade lower [36] than the common mode filters, it will also compensate for mismatches between the two common mode filters which would decrease the CMRR.

The output of the protection circuit is connected into the in-amp which will amplify the difference between the two signal lines. As mentioned above, having a CMRR of 80 dB or more is regarded as desirable for a good signal to noise ratio when measuring biopotentials. The components of the protection circuit should therefore be of the same values and have low tolerances on each signal line. The in-amp itself should also have a high CMRR value. The gain of the amplifier should be chosen to amplify the signal to a reasonable level so that the main gain stage can amplify it effectively. In-amp gains are often between 5-20 [29, 34, 37] so that DC offsets caused by half-cell potentials do not to saturate the amplifier.

The gain stage/s provide additional amplification, thousands of times in the case of an EEG signal, to bring the signal amplitude to a level suitable for input to an ADC. Multiple op-amp-based gain stages are often used to reduce the effect of input offset voltages, noise introduced by the amplifier and reduction in amplifier bandwidth. If a single amplifier with a large gain is used, all three of the above can limit the capability of the circuit to measure the EEG effectively. There are amplifiers available which are designed to have low input offsets and low noise verse gain relationships. Using multiple stages may increase the cost and footprint of the circuit. The gain stage usually has high pass filtering to eliminate the half-cell potentials and input offset voltages as well as an anti-alias filter on the output.

The mains interference rejection is usually done in one of two ways, through the use of filters, be it a notch or low pass filters, or through the use of driven right leg circuit. The filters can filter out mains interference very well if 50 Hz/60 Hz is not within the frequency band of interest but if it is and a notch filter is used, EEG signals which contain 50 Hz/60 Hz will be significantly distorted. Also, if a notch filter is used, the

stop band will need to be adjusted for whichever country the device is being used in. The driven right leg should not interfere with the EEG signal itself and should only cancel out common mode voltages which will primarily consist of mains interference but can include interference from other sources. A diagram of a simplified driven right leg circuit connected to a person is shown in Figure 12.

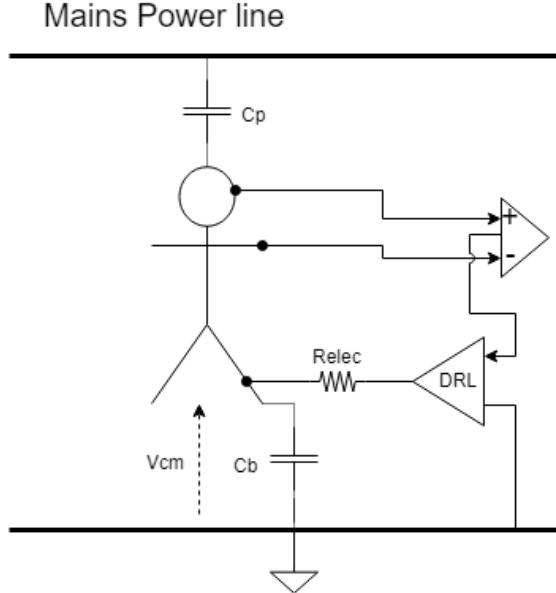


Figure 12 Biomeasurement circuit utilising a driven right leg circuit

The placement of the electrodes in the diagram is not important. Mains interference is coupled to the body through the stray capacitance C_p and C_b is the capacitance between the body and ground. The time-varying voltage between the body and ground is the common mode voltage (V_{cm}). Reducing the impedance between the body and ground will reduce the size of V_{cm} . This is done through the DRL. The simplest DRL circuit is an inverting amplifier which measures the common mode voltage as seen at the input of in-amp and inverts and amplifies it, and drives the DRL electrode to the resulting voltage. This effectively creates a parallel path through the resistance “ $Relec$ ” to ground when the DRL is referenced to ground. The larger the gain of the DRL, the lower the effective impedance of $Relec$ resulting in the reduction of V_{cm} . However, a large gain can cause the closed loop system to become unstable [38]. As previously mentioned, the DRL can cancel out other common mode voltages on the body. Any common mode voltages measured by the in-amp will be input into the DRL and should be cancelled. This is an advantage of the DRL over just using filters.

2.2.7 Feature extraction and probability distributions

Methods for feature extraction in biological signals vary in complexity from thresholding [32], where the current value is compared to a threshold to establish whether or not activity is present, to more complex methods such as statistically-based methods which estimate the probability that a signal belongs to one of several classes [39]. The naïve Bayes classifier method will be described below but the fundamentals of fitting probability density function to data must be discussed first.

The *probability density function* (PDF) of a continuous random variable is a function which describes the probability of seeing a certain value given a model of a process. PDFs are modelled using many types of function and the most appropriate one should be fit to the data to give an accurate representation of the probability. This can be done by recording a large dataset and creating a histogram. The histogram gives a discrete probability while the PDF will give an estimated continuous probability for a given value. The more observations which are used in the histogram, the better the representation of the probability and therefore the PDF will be more accurate. *Saji et al* [40] found that the amplitude of the EEG recorded from preterm infants could be modelled to a log-normal distribution, a type of probability distribution function, and the location, scale and shape parameters could be used to estimate their brain development. The fitting of different PDFs to the datasets of *Saji et al.* can be seen in Figure 13.

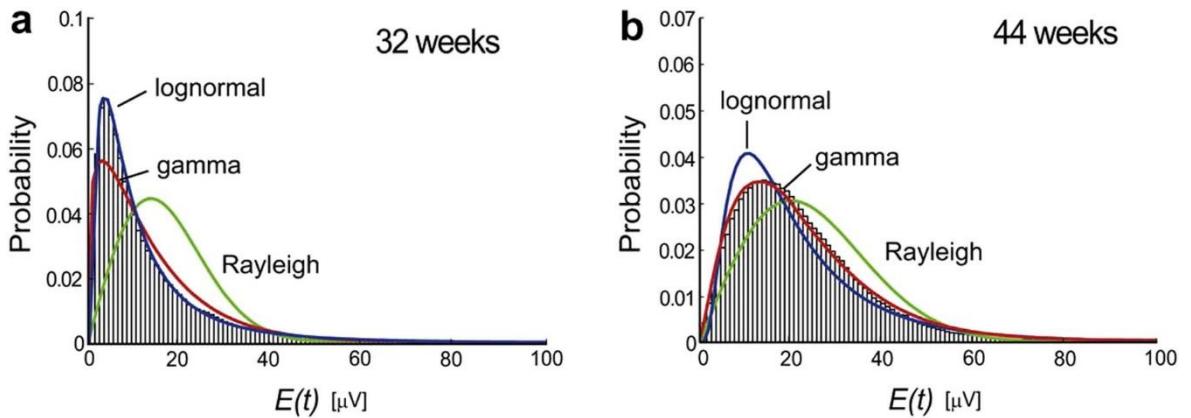


Figure 13 Two representative histograms of the envelope and probability distributions (blue line profile: lognormal distribution; red: gamma; green: Rayleigh) for the preterm EEGs. *a*. An example of a probability distribution well fitted by a lognormal distribution. The data was obtained from an infant of the postconceptional age of 32 weeks *b*. An example of a probability distribution well fitted by a gamma distribution. The data was obtained from an infant of the postconceptional age of 44 weeks – Image from R.Saji et al [40]

A common method for finding the parameters which best describe the PDF is called the *maximum likelihood estimate* (MLE). It is used to maximise the likelihood function by finding the PDF parameters which best match the given data [41]. The location defines the centre, the scale defines the spread and the shape defines the skewness of the PDF. For example, the PDFs in Figure 13 are skewed to the left. Once a set of MLE parameters are found for various PDFs, they can be compared using the negative log-likelihood function. A smaller negative log likelihood is equivalent to better MLEs [42].

In classification problems where each observation is to be assigned to one of a number of discrete classes, a PDF may be given or estimated for each class. In such cases, a naive Bayes classifier provides a useful method of predicting the likelihood of an observation belonging to each class (and hence identifying the most likely class). The equation for the naive Bayes classifier is shown below.

$$p(C_k|x) = \frac{p(x|C_k) * p(C_k)}{p(x)} = \frac{p(x|C_k) * p(C_k)}{\sum_{k=1}^{N-1} p(x|C_k) * p(C_k)} \quad (1)$$

C_k *Class k*

x	<i>new observation</i>
$p(C_k/x)$	<i>probability of C_k given x (posterior)</i>
$p(x/C_k)$	<i>probability of x given C_k (likelihood)</i>
$p(C_k)$	<i>probability of C_k (prior)</i>
$p(x)$	<i>probability of x (sum of the probability of x given each class multiplied by the prior probability of that class)</i>
N	<i>number of classes</i>
n	<i>class n, $0 \leq n < N$</i>

An example using *Saji et al.* PDFs will be used to explain a naïve Bayes classifier but this is not the statistical method suggested by the author later in the report. Taking the two PDFs in Figure 13 as the two classes, suppose that a new observation of 30 µV was observed in the EEG, and that the age of the infant was known to be either 32 weeks or 44 weeks postconceptional age (PCA). Since there are only two possibilities, the sum of the probabilities for these two classes (32 weeks and 44 weeks) must be one. Suppose 10 infants were previously tested, of whom 6 were 32 weeks PCA. This would make:

$$p(C_{32\text{weeks}}) = \frac{6}{10} = 0.6 \quad p(C_{44\text{weeks}}) = 1 - 0.6 = 0.4 \quad (2)$$

This is the prior probability created from previous data. Now using the 30 µV reading, the likelihood of this reading in each class can be measured from the PDFs (using the log-normal PDF for the 32 weeks data and the gamma PDF for the 44 weeks data).

$$p(30 \mu V | C_{32\text{weeks}}) \cong 0.005 \quad p(30 \mu V | C_{44\text{weeks}}) \cong 0.015 \quad (3)$$

The probability that the observed value is from a 32 week PCA or a 44 week PCA infant is shown below:

$$\begin{aligned} p(C_{32\text{weeks}} | 30\mu V) &= \\ \frac{p(30 \mu V | C_{32\text{weeks}}) * p(C_{32\text{weeks}})}{p(30 \mu V | C_{32\text{weeks}}) * p(C_{32\text{weeks}}) + p(30 \mu V | C_{44\text{weeks}}) * p(C_{44\text{weeks}})} & \quad (4) \\ &= \frac{0.005 * 0.6}{0.005 * 0.6 + 0.015 * 0.4} = 0.3333 \end{aligned}$$

$$p(C_{44\text{weeks}} | 30\mu V) = 1 - 0.333 = 0.667 \quad (5)$$

It is more likely that the observation is from a 44 week PCA infant than a 32 week. This demonstrates a naïve Bayes classifier for two classes with one observation but the classifier also works for multiple classes and observations. If multiple observations are used in a single probability estimation, then each observation is “naively” assumed to be completely independent of the others. The theory behind naïve Bayes classifiers is described in more detail by *Murphy* [43].

3 Design and implementation

This chapter details the design of the EEG measurement circuit, the serial optoisolation circuit, the design of EEG electrode cups and the development of the BCI software. The EEG measurement circuit records the EEG from the user and amplifies it so that the microcontroller can obtain an accurate digital representation of the EEG. The circuit also isolates the EEG signal from the common mode interference through the use of a driven right leg circuit. The serial optoisolation circuit allows serial communication

between the microcontroller and PC while keeping the two electrically isolated from each other.

The BCI software has displays windows. The first window displays the time-domain signal and FFT of the EEG in real time. The second window displays multiple stimuli to the user through the use of flashing checker boxes. A simplified keyboard is split between the checker boxes and the user selects a character by looking at the corresponding checker box. The user may need to make multiple selections before they can select the character of interest. A block diagram of the system is shown in Figure 14.

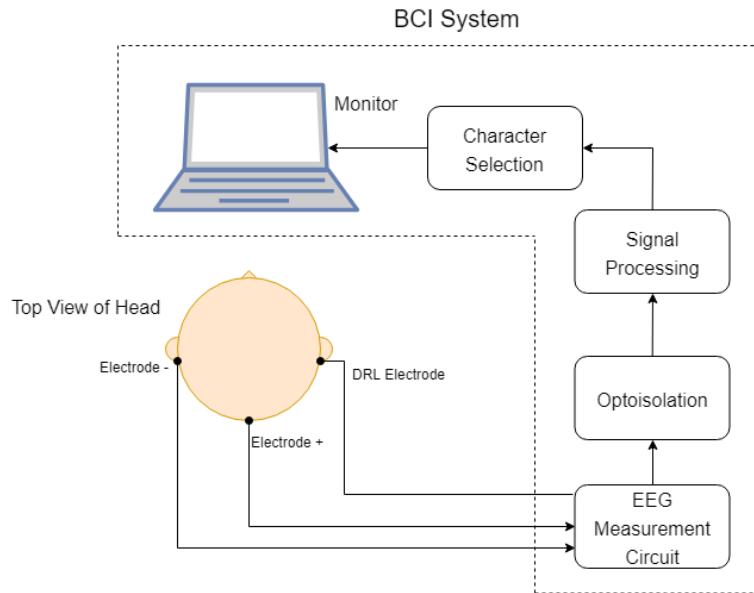


Figure 14 Block diagram for SSVEP-based BCI for character selection on a virtual keyboard

A local git repository was used for version control on both the circuit diagrams and software. Both the circuits and the software were validated and characterised as described later in this chapter.

3.1 Circuit design

As the EEG signal is in the region of 1-100 μV , it requires a high level of voltage amplification before it is sampled by the microcontroller. But as the electrodes and circuit carry, as well as introduce, interference to the signal which could be larger in amplitude than the EEG, this interference has to be either filtered out or otherwise mitigate so that the EEG can be amplified to an appropriate level without saturating of the amplifier. This was the primary design objective of the EEG measurement circuit, but circuit complexity and cost were also factors in the design and selection of components. The bill of materials for the circuit and electrodes can be is shown in Appendix A - Bill of materials

A block diagram showing the main components of the EEG measuring circuit can be seen in Figure 15 with the electrodes connected to Oz, the left mastoid and right mastoid according to the 10-20 electrode placement system.

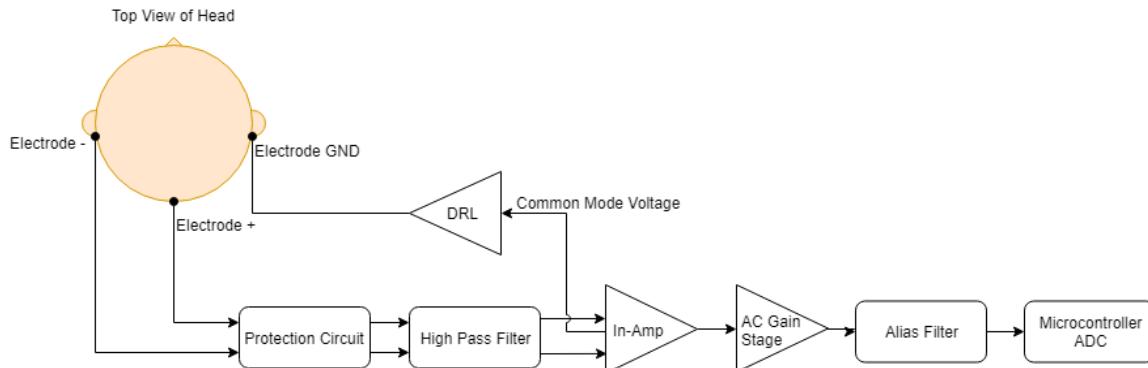


Figure 15 Block diagram of EEG measurement circuit connected to the user

The EEG signal is being measured between Oz and the left mastoid (unipolar); the right mastoid is connected to the output of the DRL. The Nucleo-F303K8 microcontroller was chosen for its small size, high clock speed (up to 72 MHz), floating point unit and 12-bit ADC. The microcontroller can be compared to an Arduino Nano which has the same form factor but is worse compared to the Nucleo in all of the specifications stated above. The author also wished to gain experience using ARM-based microcontrollers.

A picture of the completed circuit on a breadboard can be seen in Figure 16 without any external connections (e.g. electrode wires, battery or serial output).

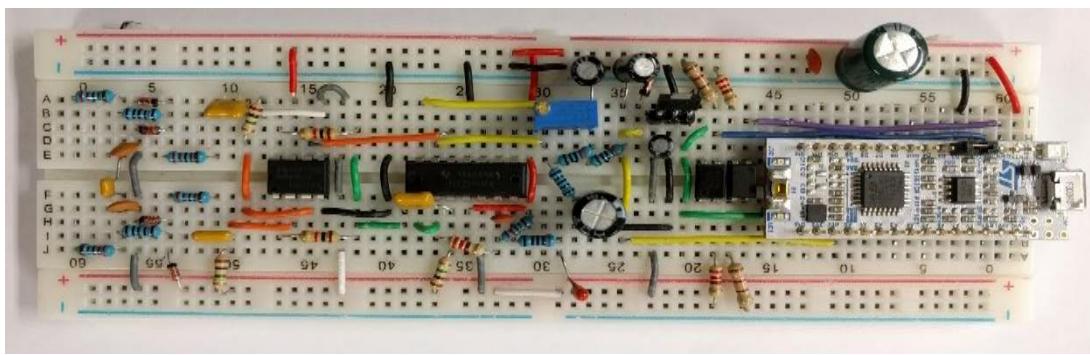


Figure 16 EEG measurement circuit and optoisolation circuit on a breadboard (without electrodes, serial-USB adapter or battery)

The total cost of the circuit was €30.08 and the cost per electrode was €0.70. Many of the components were purchased in sets. In those cases, the unit cost was estimated by dividing the total cost by the number of units in the set. The circuit cost is for a single channel EEG measurement circuit but if more channels were to be added, the microcontroller, optoisolation circuit and DRL could be reused so the cost after the first channel is €4.99 per channel with a maximum of nine channels (the microcontroller has nine analog inputs). The speed of the microcontroller code and sample rate may limit this to a smaller number of channels.

The theoretical input voltage specifications for the circuit are shown below in Table 1. The maximum peak-peak amplitude at the ADC pin is taken as 3 V or peak amplitude of 1.5 V as the negative peak will clip with larger voltages. The resolution is from the 3.3 V, 12-bit ADC.

Table 1 Table of theoretical input voltage specifications for the circuit

Values	At Minimum Gain	At Maximum Gain
Gain	51	25500
Input Max Peak-Peak	58.8 mV	117.6 uV
Input Resolution	15.8 uV	31.6 nV
Max Input DC	± 500 mV	± 500 mV

3.1.1 Powering the circuit

The circuit has three power rails, labelled as 5 V, Vref and GND in the following circuit diagrams. 5 V is from the 5 V pin of the Nucleo which is regulated, Vref is the reference voltage for the EEG signal being measured and is the voltage the user's body will be held at. Vref is 1.5 V from a MIC37100-1.5WS voltage regulator. GND is the GND pin on the Nucleo. 1.5 V was chosen as the signal reference as it is close to the mid-range input voltage of the 3.3 V ADC. The ADC is the A0 pin on the microcontroller. The whole circuit is powered by a 9 V battery connected to the microcontrollers. Current consumption was not a major concern when designing the circuit. Nucleo connections can be seen in Figure 17.

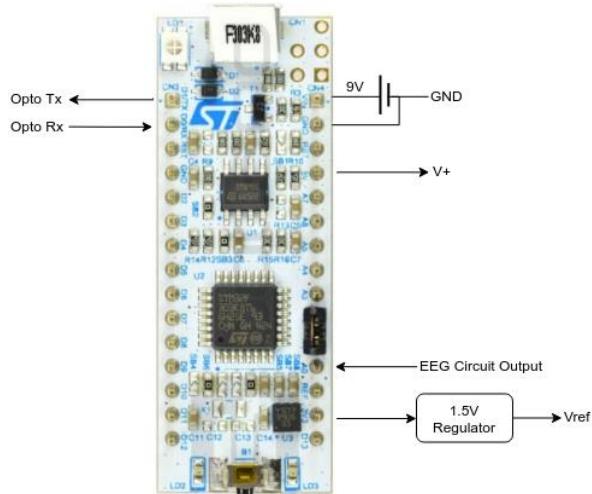


Figure 17 Connections to Nucleo – Image adapted from Botland [44]

The above diagram doesn't include the capacitors on the power rails, this can be seen in the full circuit diagram in the Appendix B - Full circuit.

3.1.2 Protection circuit

The EEG electrodes are connected to the protection circuit. This circuit protects the user and as well as the circuitry it feeds into from relatively high voltages and currents. The circuit shown in Figure 18 is a common protection circuit used for EEG measurement circuits and the values used here were taken from Bo Luan et al. [35].

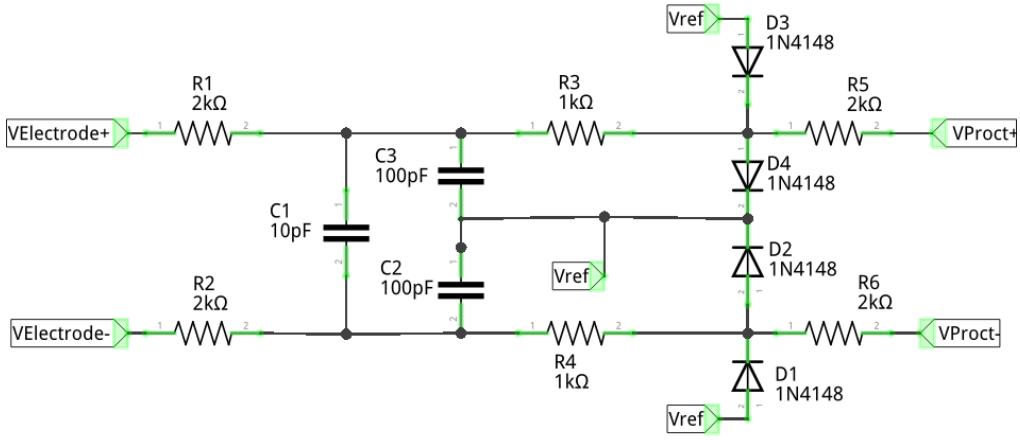


Figure 18 Circuit diagram of the protection circuit . VElectrode is the input from the positive and negative electrode. VProct is the positive and negative output of the protection circuit

The two input signals from the electrodes first pass through a differential and common mode low pass filter. The low pass common mode filters formed by R1 and C3, and R2 and C2 are filter out EMI on each signal line which may include radio frequency interference. The differential filter formed by R1, R2 and C1 creates a second order low pass filter on both signal lines. The transfer functions of both filters are shown below.

$$\frac{V_{filtDiff}(s)}{V_{inDiff}(s)} = \frac{1}{(R_1 + R_2)C_1 s + 1} = \frac{1}{40 * 10^{-12}s + 1} \quad (6)$$

$$\frac{V_{filtcm}(s)}{V_{in}(s)} = \frac{1}{R_1 C_2 s + 1} = \frac{1}{200 * 10^{-12}s + 1} \quad (7)$$

After these filters, the signal passes through two resistors which further limit the current on each signal line. The diodes (1N4148) following are used to clamp the input voltage to the next stage which is an in-amp. The diodes start conducting around 500 mV will act as open circuits below this threshold. Voltages on either line within ± 500 mV of reference to Vref, therefore pass through this stage unaffected. The voltages being measured in the EEG will be in the range of microvolts with some DC offsets of tens of millivolts caused by the half-cell potentials mentioned previously. Figure 19 shows the input and output of the protection circuit to a 140 mV and 2 V peak-to-peak input.

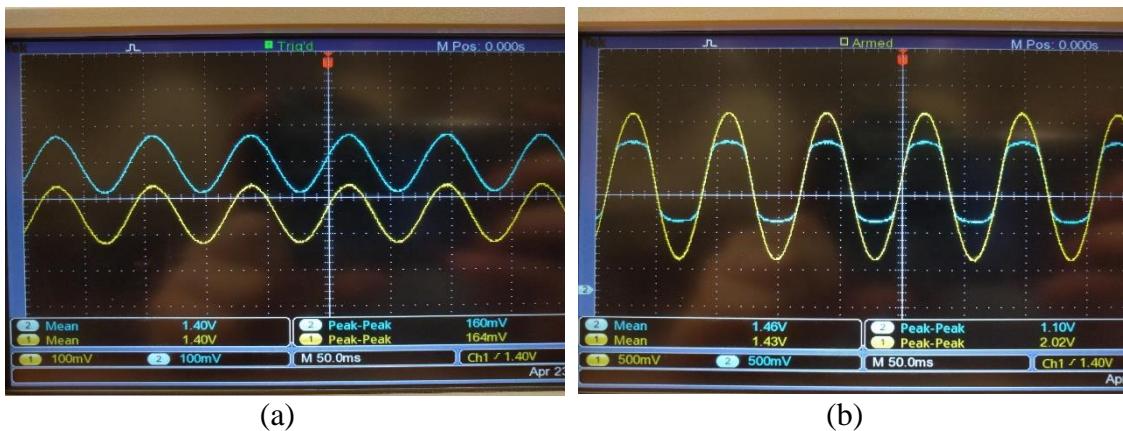


Figure 19 (a) Input (yellow) and output (blue) of protection circuit from input within the normal operating range. (b) Input (yellow) and output (blue) of protection circuit from input outside the normal operating range.

In Figure 19 (a), the output is almost unaffected by the protection circuit but in Figure 19 (b) with a much larger input voltage, the diodes clamp the voltage leaving the protection circuit.

3.1.3 Instrumentation amplifier

The output of the protection circuit is connected to the in-amp which amplifies the difference between the two electrode voltages. Due to the DC offset that can occur due to half-cell potentials, the inputs to the in-amp are high-pass filtered prevent amplification of DC voltages in the in-amp. The cut off frequency for these filters was 0.159 Hz. The value of $1\text{ M}\Omega$ for R_3 and R_4 was chosen so as to not reduce the large input impedance of the in-amp significantly.

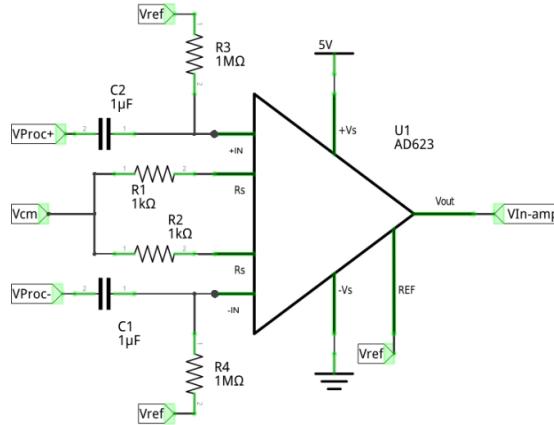


Figure 20 Circuit diagram of the instrumentation amplifier. V_{Proc} is the positive and negative output from the protection circuit. V_{cm} is the common mode voltage measured by the in-amp. $V_{in\text{-amp}}$ is the output of the in-amp. V_{ref} is the 1.5 V reference voltage

The in-amp used was an Analog Devices AD623, which was chosen due to its high CMRR (mean CMRR of 80 dB), its ability to be powered from a single supply and its rail-rail output. After the DC components of the signal were filtered out, the AC components were amplified by a gain of 51 using the two $1\text{ k}\Omega$ resistors. The gain (A_{inAmp}) of AD623 is calculated as shown below:

$$A_{inAmp} = \frac{100k}{R_g} + 1 = \frac{100k}{1k + 1k} + 1 = 51 \quad (8)$$

This is a higher gain than what is usually seen [35, 37] in bio-amplifier circuits as most designs do not filter the DC offset into the amp and have to restrict their gain to a small value to prevent of saturating the output. Lastly, the common mode voltage (V_{cm}) as seen from the inputs of the in-amp is measured from the middle of the two gain resistors and is given by:

$$V_{cm} = \frac{V_{inAmp_{in+}} + V_{inAmp_{in-}}}{2} \quad (9)$$

V_{inAmp+} Input voltage at the positive input of in-amp
 V_{inAmp-} Input voltage at the negative input of in-amp

V_{cm} is used in the driven right leg (DRL) circuit to increase the overall CMRR of the circuit, as described in Section 3.1.5. The transfer function of the in-amp circuit from the input of the high pass filters to output is shown below.

$$\frac{V_{inAmp_{out}}(s)}{V_{inAmp_{in+}}(s) - V_{inAmp_{in-}}(s)} = A_{inAmp} * \frac{s}{s + \frac{1}{C_1 R_4}} = 51 * \frac{s}{s + 1} \quad (10)$$

A high CMRR is required to be able to separate the EEG signal from the interference, especially that from 50 Hz mains. The mean CMRR of 80 dB stated by the AD623 datasheet is for the in-amp alone, without components connected to the inputs. So the in-amp would be able to suppress the 50 Hz interference to a high degree with no complementary components added. But the individual component tolerances on both signal lines will decrease the CMRR if components are not matched. Metal-oxide film resistors were used in the protection circuit due to their lower tolerance of 1-2% compared to the 5% of carbon film resistor. Variable electrode-skin impedance would also reduce of the CMRR. This is why the DRL right leg circuit is used to increase the CMRR of the circuit.

3.1.4 Gain stage and alias filter

The output of the in-amp is connected to another gain stage based on the TLC2274 quad op-amp which has a variable AC gain of ~1-501 chosen by varying R5 shown in Figure 21. The TLC2274 was chosen due to its rail-to-rail output, low noise and low input offset voltage. The DC gain is ~1 so DC will be pass through unamplified but the AC components will be amplified. The cut-off frequency of the gain stage is ~0.338 Hz. When the circuit is switched on, capacitor C1 needs to charge up to Vref which it does through R5 and R4. To speed this up, R5 should be reduced to a low resistance, then increased to the required gain once the circuit has settled. The transfer function for the gain stage is shown below:

$$\frac{V_G(s)}{V_{inAmp_{out}}(s)} = \frac{R_5 C_1 s}{R_4 C_1 s + 1} + 1 = \frac{R_5 * 470 * 10^{-6} s}{0.47 * 10^{-6} s + 1} + 1 \quad (11)$$

V_G Output voltage from gain stage

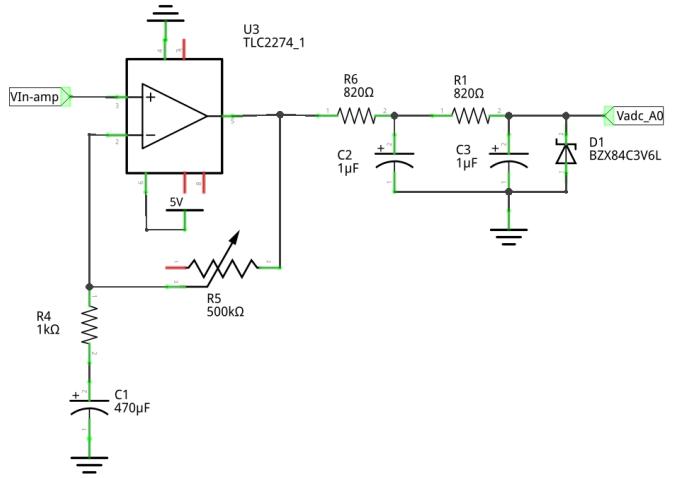


Figure 21 Circuit diagram for gain stage and alias filter. Vin-amp is the output of the in-amp. Vadc_A0 is the output of the alias filter connected to analog pin A0 on the Nucleo

The output of this gain stage is connected to a second order, passive low pass filter to prevent out any high frequencies entering the ADC. This reduces the likelihood of aliasing in the digital signal. The cut-off frequency is 194 Hz which shouldn't attenuate frequencies in the band of interest in this application (5-20 Hz), but dramatically attenuates frequencies above the Nyquist frequency of 500 Hz. The transfer function is shown below.

$$\frac{V_{lp(s)}}{V_{G(s)}} = \left(\frac{1}{R_6 C_2 s + 1} \right) \left(\frac{1}{R_1 C_3 s + 1} \right) = \frac{1}{(820 * 10^{-6} s + 1)} \frac{1}{(820 * 10^{-6} s + 1)} \quad (12)$$

V_{lp} Output voltage from low pass filters

Lastly, before entering the ADC pin of the microcontroller, the output of the alias filter is connected to a 3.6 V zener diode in reverse bias connected to ground. The ADC pin is 3.3 V tolerant and has a maximum rating of 4 V. When the voltage approaches the breakdown voltage of the zener diode, it starts conducting and limits the voltage to approximately 3.6 V. A 3.3 V zener was tested but it started conducting around 2.5 V and the top of the waveform entering the ADC was being clipped so a 3.6 V zener was used to accommodate a larger voltage range. At 5 V, the 3.6 V zener diode clamped the output voltage at 3.27 V. The non-linear distortion of the signal due to clipping would result in an incorrect FFT of the input signal.

3.1.5 Driven right leg circuit

The DRL uses the common mode voltage measured by the in-amp and inverting, amplifying and driving it back into the DRL electrode to try to counteract the interference measured on the body or in the circuit. The DRL circuit can be seen in Figure 22.

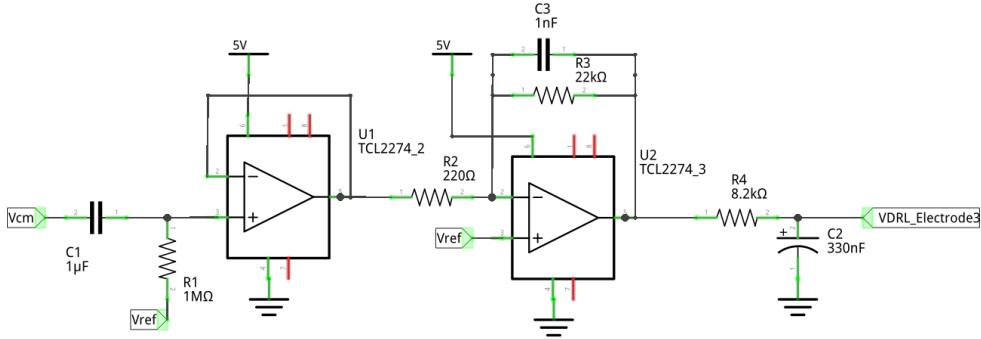


Figure 22 Circuit diagram for the driven right leg. V_{cm} is the common mode voltage measured by the in-amp. $VDRL_Electrode3$ is the output of the DRL connected to the third electrode. $Vref$ is the 1.5 V reference voltage

The common mode voltage first passes through a high pass filter to filter out the DC component of the V_{cm} . There was a 600 mV offset caused a transistor at each signal input of the in-amp which needed to be filtered out before the DRL so as not to saturate the DRL output. The same values for the in-amp high pass filter were used before entering a buffer amp (TCL2274).

Following the buffer amp is a low pass, inverting op-amp (TCL2274). Common gains for DRLs vary from -10 to -100, so a variable resistor for R_{10} would be useful here but due to limited space on the breadboard, a 220Ω was used giving a gain of 100. Due to the two cascaded high-pass filters before the DRL, the phase of the input to the DRL was leading the common mode voltage on the body which would cause the DRL to become unstable. As the measured V_{cm} was out of phase with the actual V_{cm} , the DRL would not cancel the common mode voltage on the body and possibly make it worse. The phase was corrected by adding the capacitors C_3 and C_2 and resistor R_4 and the instability was not seen in either simulation or testing afterwards.

R_4 is also used to limit the amount of current that can flow into the user from the DRL. The transfer function for the DRL is shown below.

$$\frac{V_{DRL}(s)}{V_{buff(s)}} = \left(-\frac{R_3}{R_2} * \frac{1}{1 + R_3 C_3 s} \right) \left(\frac{1}{R_4 C_2 s + 1} \right) = \\ \left(-100 * \frac{1}{1 + 22 * 10^{-6} s} \right) * \left(\frac{1}{2.703 * 10^{-3} s + 1} \right) \quad (13)$$

It would be expected that the increase in CMRR from the DRL would be equal to its gain which in this case is 40 dB.

$$CMRR_{DRL} = 20 * \log_{10}(A_{DRL}) = 20 * \log_{10}(100) = 40dB \quad (14)$$

Finally, the output of the DRL is connected to the third electrode to act as the ground electrode.

3.1.6 CMRR of circuit

The common mode rejection ratio was measured for the circuit with and without the DRL circuit to see its effect. The common mode rejection ratio is given by:

$$CMRR = 20 * \log_{10} \left(\frac{A_{diff}}{A_{cm}} \right) \quad (15)$$

A_{diff} Differential Gain
 A_{cm} Common Mode Gain

The CMRR was tested using the Bitscope Micro (USB oscilloscope/signal generator). The signal generator needed to be floating and not connected to earth so that the DRL could drive it up and down. The Bitscope was connected to generate a differential signal as shown in Figure 23.

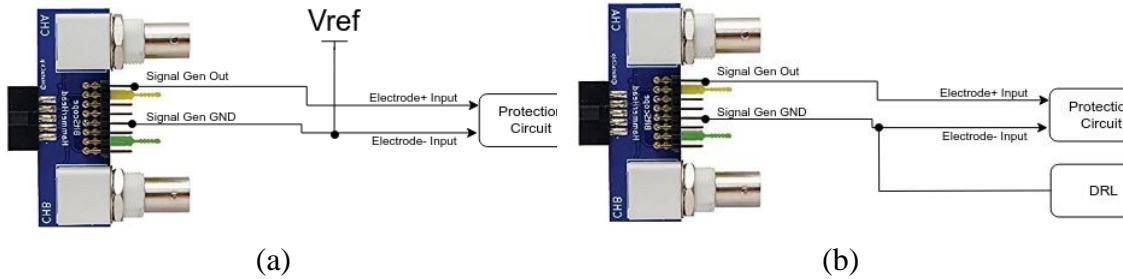


Figure 23 (a) Bitscope Micros ground connected to V_{ref} , generating a differential signal into protection circuit. (b) Bitscope Micros ground connected to DRL, generating a differential signal – Image adapted from Amazon [45]

A signal of 300 mV at 10 Hz, 50 Hz and 100 Hz were input into the circuit and the output from the circuit (after alias filter) was measured. The gain of the in-amp was reduced to 6 and the gain stage to 1 so not to saturate the op-amps. The Bitscope was then connected to generate a common mode signal as shown in Figure 24.

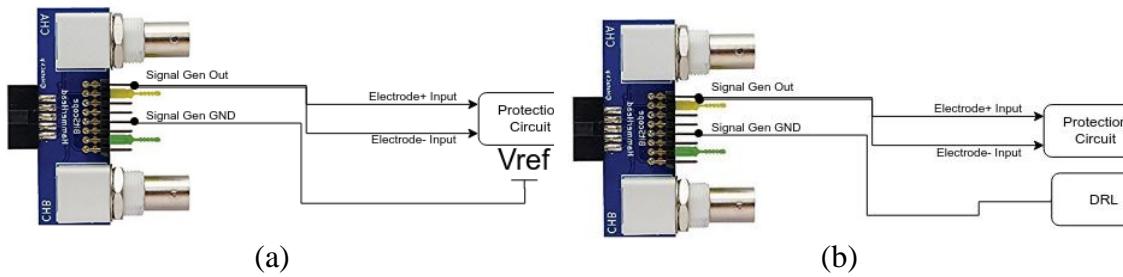


Figure 24 (a) Bitscope Micros ground connected to V_{ref} , generating a common mode signal into protection circuit. (b) Bitscope Micros ground connected to DRL, generating a common signal into protection circuit – Image adapted from Amazon [45]

The in-amp gain was kept at 6 but the gain stage gain was increased until a common mode signal was clearly visible and measurable on the oscilloscope. The resistance of the variable resistor in the gain stage was measured to calculate the gain. The measured output was then divided by this gain so that the differential signal and common mode signal had the same overall gain. The calculated CMRR at each frequency without the DRL and with the DRL is shown in Table 2.

Table 2 CMRR of the circuit without the DRL and with the DRL at 10 Hz, 50 Hz and 100 Hz

Frequency (Hz)	Vref CMRR(dB)	DRL CMRR(dB)
10	53.00	78.84
50	54.92	89.99
100	55.58	86.75

From the three tests conducted, the DRL increase the CMRR by roughly 30 dB with the highest CMRR of 35 dB achieved at 50 Hz which is close to the expected value of 40 dB.

3.1.7 Frequency response of the circuit

The frequency response of the output of the circuit (after alias filter) was designed to have a passband between 5 Hz to 20 Hz as all of the SSVEP stimuli will within this range. The DRL was designed to have a passband from 1-100 Hz but to be working perfectly at 50 Hz-60 Hz as this is the main interference which needs to be counteracted. The phase response of the DRL needs to be a constant -180° throughout the passband so as not to become unstable due to phase lead or lags in the closed loop.

The frequency response was tested using the Bitscope Micro's signal generator and was connected as shown in Figure 23 (b) for the circuit output response and connected as shown in Figure 24 (b) for the DRL response. The lowest frequency the Bitscope could generate was 5 Hz. The gain of the in-amp was set to 6 and the gain of the gain stage to 1 for both tests. The frequency response was simulated in LTspice with an input voltage of 136 mV for output response and an input of 220 mV for the DRL response. All of the LTspice components were ideal components apart from the in-amp (AD632) and the zener diode (BZX84C3V6). LTspice calculates the magnitude value in dBs in reference to 1 V so the measured values were calculated the same way.

$$A_V = 20 * \log_{10} \left(\frac{V_{out}}{1V} \right) \quad (16)$$

Figure 25 shows the frequency response of the output of the circuit. Every point on the measured gain and phase graphs match the simulated response other than the first point at 5 Hz, there may be some additional high pass filtering which is not replicated in the simulation. The gain at 5 Hz is 1.28 dB which is not below the -3 dB point from the maximum gain (3.17 dB) so it is still within the passband. The gain from 10-20 Hz is flat and has a value of roughly 3 dB and sharply decreases afterwards giving a gain of -16.5 dB at 500 Hz. This will suppress any frequencies above the Nyquist frequency of 500 Hz greatly. The measured passband of the circuit can be taken as roughly 3-75 Hz as these are the -3 dB points on the measured graph. The phase response is not of interest for measuring SSVEP so there were no design criteria for it.

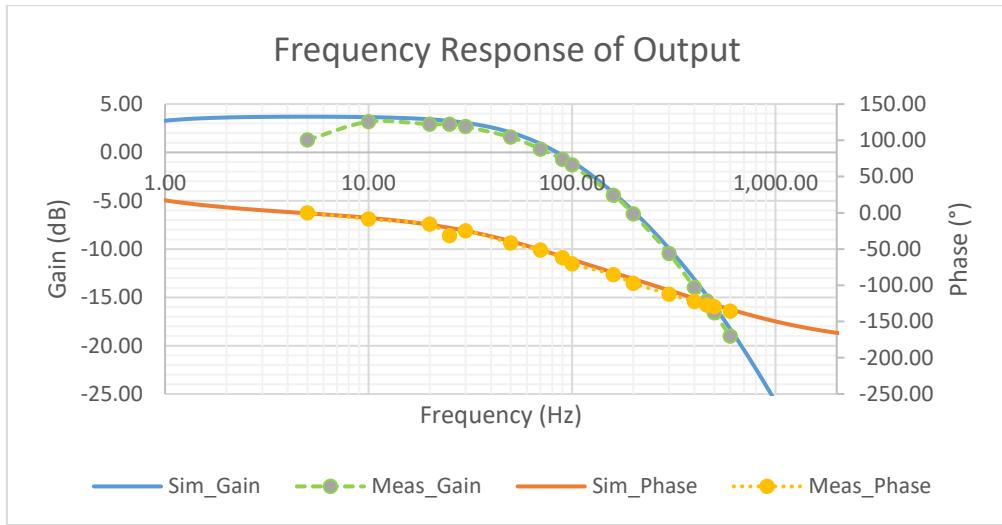


Figure 25 Frequency response of circuit output (simulated and measured)

Figure 26 shows the measured and simulated frequency response of the DRL circuit. Again, there seems to be some high pass filtering which was not observed in the simulation. Because the additional high pass filtering is seen in both the DRL and output response means it is likely to originate within the protection circuit or in-amp. The upper end of the measured gain response differs from the simulated one and rolls off a few kHz before the simulated one. This may arise to additional low-pass filtering not modelled in the simulation within the closed loop. The phase response between 5 Hz and 400 Hz stays very close to -180° so this range can be taken to give a measured bandwidth of 395 Hz for the DRL. But for the purpose of the DRL, 50 Hz is the main frequency to be compensated for and, as seen in the graphs, the gain and phase match the simulated response very well. Overall the circuit works as expected. A simulated frequency response for the output and DRL over a wider range of frequencies with full gain is shown in Appendix C - Simulated frequency responses.

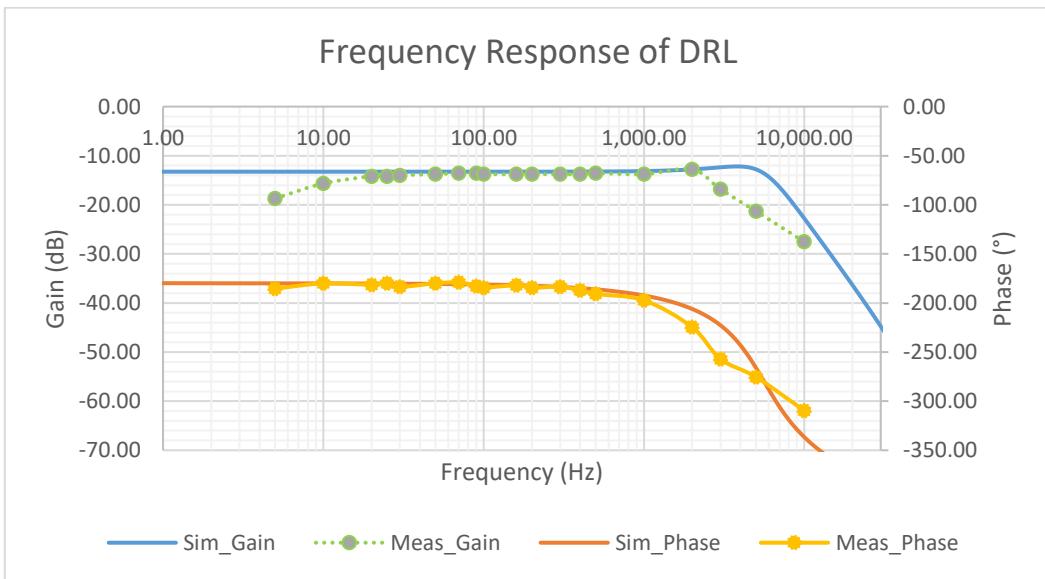


Figure 26 Frequency response of DRL (simulated and measured)

3.1.8 Communication between microcontroller and PC

The circuit needed to communicate with a PC to interact with the BCI software but it couldn't be connected directly to the PC as this could provide a path from the user to mains voltage through the PC under fault conditions. So the circuit communicated to the PC through optoisolated serial lines. The TX and RX pins of the microcontrollers UART were connected to optocouplers which were then connected to a CH340G serial-to-USB breakout board which was connected to the PC. The circuit diagram for this can be seen in Figure 27.

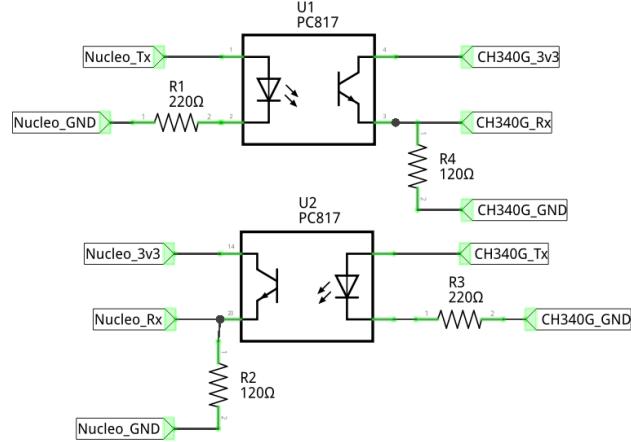


Figure 27 Circuit diagram for optoisolating UART lines. CH340G (serial to USB converter) is connected to a PC

The optocoupler used was a PC817 which is low cost (€0.02 each) and provides 5 kVrms isolation between the inputs and outputs. The values of 220Ω and 120Ω for R1/R3 and R4/R2 were chosen to have fast rise and fall times for switching the phototransistor on and off. These values were chosen from graphs provided in the datasheet and from testing the rise and fall times. Figure 28 shows the input and output from the PC817 using two 220Ω resistors for R1 and R4 in the diagram above and then using 220Ω and 120Ω while the Nucleo was repeatedly transmitting at 57600 baud.

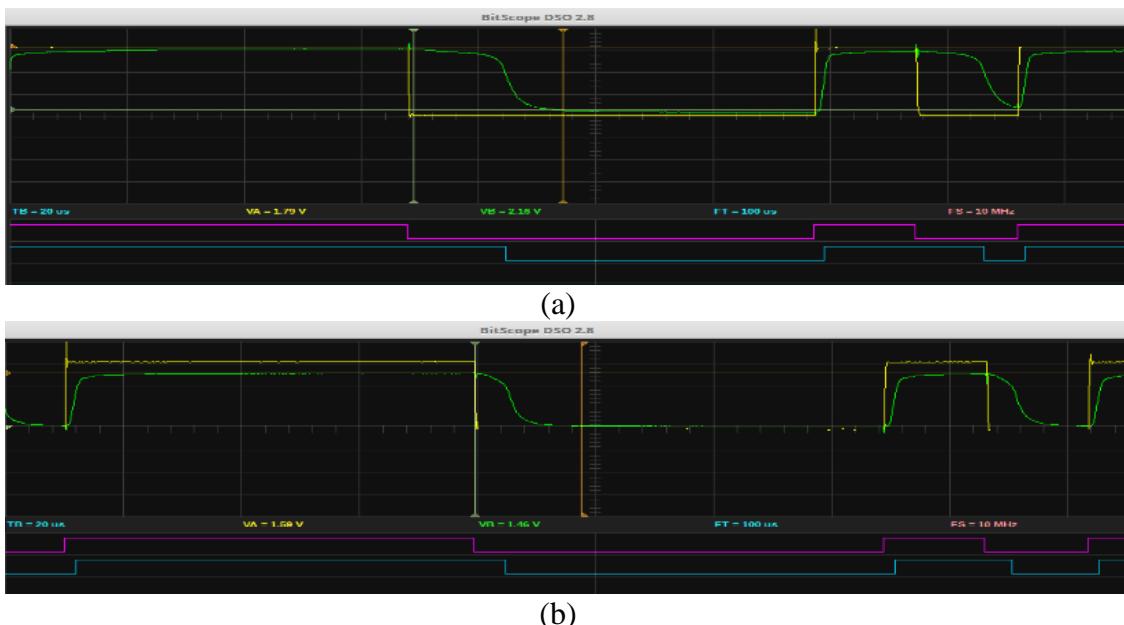


Figure 28 (a) Traces of input (analog-yellow, logic-pink) and output (analog-green, logic-blue) of the PC817 using input resistor of 220Ω and output resistor of (a) 220Ω . (b) 120Ω . Input for both figures is from the Nucleo's UART Tx pin operating at 57600 baud

Figure 28 shows the rise time is very fast for the transistor output, but the fall time is much slower. The fall time using the $220\ \Omega$ resistor for the output of the transistor is very long (nearly a bit interval long on the logic trace) and the PC wasn't able to decode the input correctly. But using the $120\ \Omega$, the fall time is much shorter and the logic traces of the input and output of the PC817 are very similar. For reference, the orange and green vertical cursors are placed in the same position for both of the analog traces.

The maximum baud rate as mentioned before was found to be 57600, traces of this baud rate and a baud rate of 115200 are shown in Figure 29.

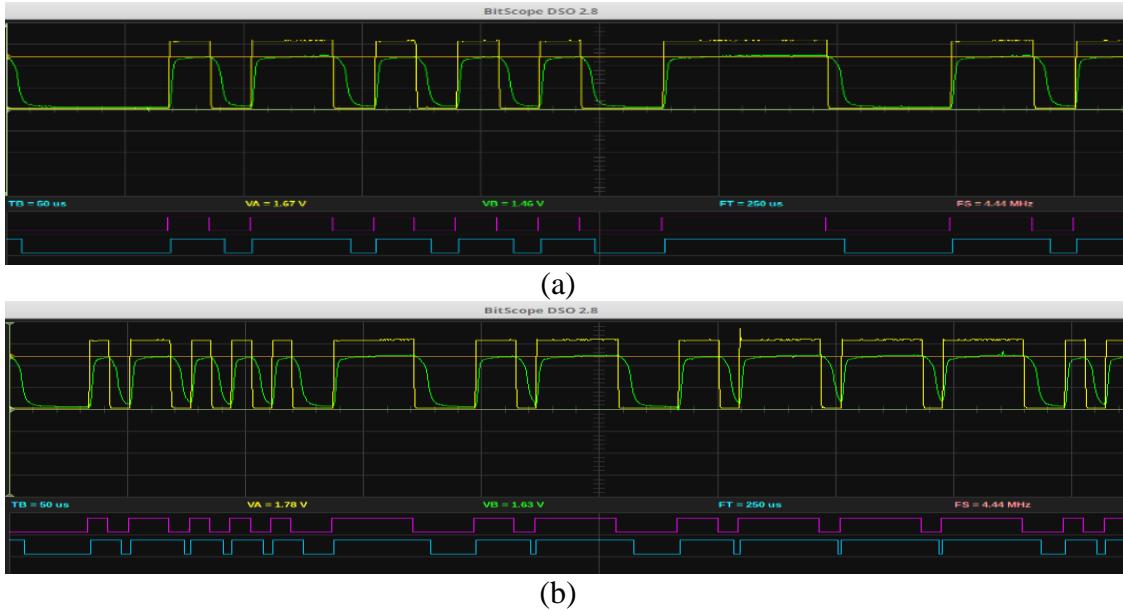


Figure 29 (a) Traces from input (analog-yellow, logic-pink) and output (analog-green, logic-blue) of PC817 with input from Nucleo's UART Tx pin operating at (a) 57600 baud. (b) 119200 baud.

The fall time again was the problem and looking at the logic trace of the output for the 119200 baud rate, the zeros are very short and the PC wasn't able to decode the messages correctly. At 57600, there was no problem decoding. 1000 bytes of data from the Nucleo's ADC were transmitted to the PC with no errors.

3.1.9 Electrodes

EEG electrode cups can be very expensive in terms of a low-cost project (\$10 or more per lead) so it was decided to make low-cost, reusable EEG electrode cups. This was done by cutting out the centre of a standard disposable ECG electrode pad, which can be bought in bulk (packs of 50) at very low-cost, and gluing a rubber shower hose washer onto it.



Figure 30 (a) Left: ECG electrode pad. Right: Homemade EEG cup electrodes. (b) Top: Homemade EEG cup electrode connected to button snap. Bottom: Button snap

The electrode is 10 mm in diameter so shower hose washers with an inner diameter of 10 mm were super glued on to form a reservoir for the conductive gel. The washers are 2 mm in depth which gives an inner volume of 157 mm³ which is similar to large dome EEG cups with a volume of 137 mm³ [46]. The electrodes can be connected to by a standard 13 mm metal button snap making the cost per electrode less than €1. These electrodes can be classified as passive wet electrodes.

All electrodes used were non-shielded, multi-strand wire and the measurement electrode wires were tightly twisted together to reduce EMI along the wire.

3.2 Software

This section will discuss the design and non-user testing of both the microcontroller software and BCI software.

3.2.1 Microcontroller

The Nucleo was programmed using the System Workbench for STM32s (SW4STM32) which is an Eclipse-based integrated development environment (IDE). There were two main options for which drivers to use for the Nucleo, using the hardware abstraction layer (HAL) drivers and low-level (LL) drivers. LL drivers are for those who wish to program “close to the metal”, working close to the registers. This can be very microcontroller specific so is not portable. HAL provides high-level API type functions which are very portable between STMs boards. As this project is open-source, making the code more portable is desirable, HAL was chosen.

The STM32CubeMX software was used to create a skeleton of the code with the peripherals setup created. STM32CubeMX is a GUI which allows the user to choose which peripherals are enabled such as ADC, DMA controllers, etc and generates the initialization code. It was used in this project to create the initialization code for a single ADC pin, one USART Tx and Rx channel, one timer, interrupts, DMA controller and setting the clock frequency and other frequencies for the peripherals.

On start-up, the Nucleo handshakes with the PC to make sure there are no byte misalignments. The ADC is sampling a 1 kHz and transmitting data in 256-sample chunks to the PC. The USART is set up to run at 57,600 baud. The code uses DMA channels for the ADC and USART to be quick and non-blocking. A flow diagram of the

Nucleo code can be found in Appendix D - Nucleo code flow diagram and the code in the CD at the back of the appendix.

3.2.2 PC

The PC software is written in Python 3 and is split into two main scripts, the real-time plotting window and the user interface which are both spawned by a “master” script. Both scripts needed to be on the main thread to function predictively so a “master” script was created to start both as separate processes but to link them with a pipe so they can share data.

3.2.2.1 Plotting real-time data

The first script, called the eegScope.py, uses the PyQtGraph module to create the real-time EEG and FFT plots. On start-up, it creates a window with the two plots and sets up a timer to trigger updating of the plot and a thread to read from the serial port. The serial thread first waits for the microcontroller to handshake before starting. Data from the serial buffer is then plotted in real time as it is received.

When there is 1 second of new data, the data is piped to the interface script and an FFT is plotted using a Hamming window. The data displayed on the plot along with the sample rate of the Nucleo were validated by inputting sinusoid of known frequency from a signal generator into the ADC pin of the Nucleo and seeing if the peaks of the FFT matched the input frequency. A 20 Hz sinewave input can be seen in Figure 31. The flow diagram of the eegScope.py can be found in the Appendix E – BCI code flow diagrams and the code in Appendix F – BCI code and in the CD at the back of the appendix.

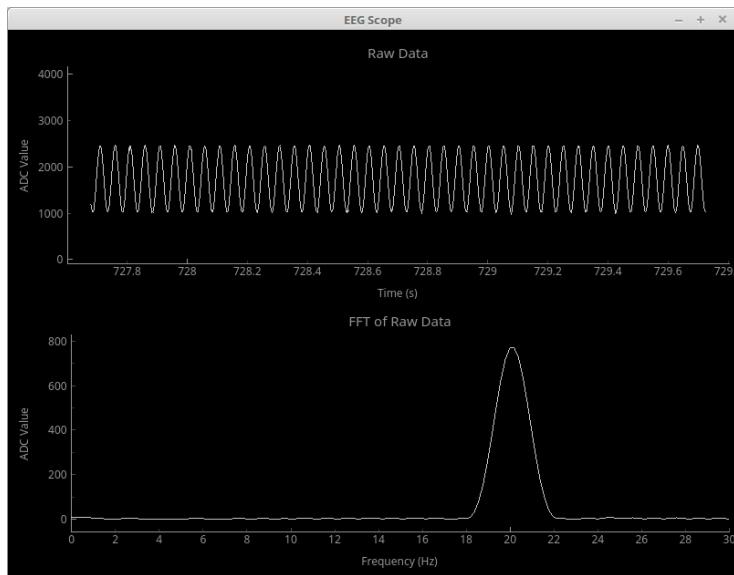


Figure 31 Real-time plot from an input signal of 20 Hz

3.2.2.2 User interface

The interface script is eegInterface.py and uses the PsychoPy module to create the window and stimuli. There are five flashing checker boxes flashing at 2.5, 5, 6, 7.5 and 10 Hz respectively. These frequencies were chosen because each flashing frequency

needs to divide evenly into the screen refresh rate. The VEP from a flashing stimulus will be double the flashing frequency as mentioned before. This means the EEG signal should contain the frequencies 5, 10, 12, 15 and 20 Hz when the user is looking at one of the stimuli. Another factor in choosing the frequencies was the sample rate and sample window size. The sample rate was 1000 Hz and the window size, to make the interface feel responsive, was chosen to be 1 second. This means each bin of the FFT represents 1 Hz, so to increase the reliability of the feature extraction algorithm, the SSVEP frequencies were chosen to match the bin frequencies. The interface window can be seen in Figure 32.

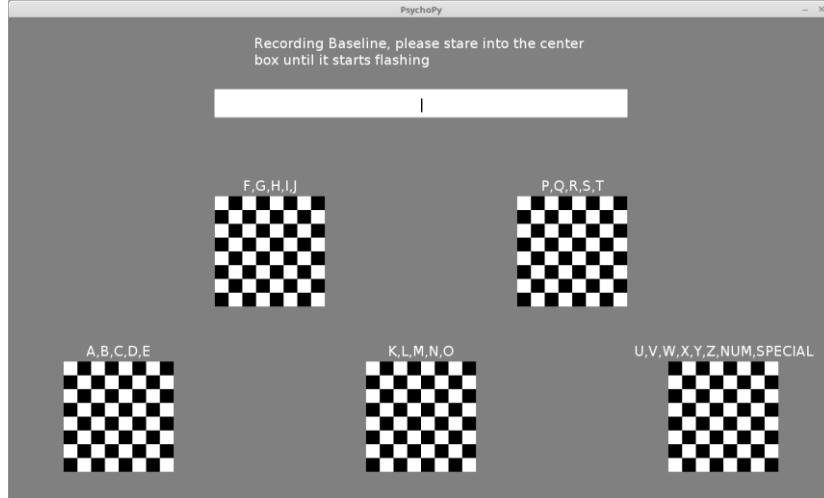


Figure 32 BCI user Interface (top level)

The checker boxes make up a simplified keyboard containing the letter of the English alphabet, ten digits and assorted special characters. The white box is where the user's input will be typed and above that is the instructions for the user. Every option which can be selected is shown in Figure 33.

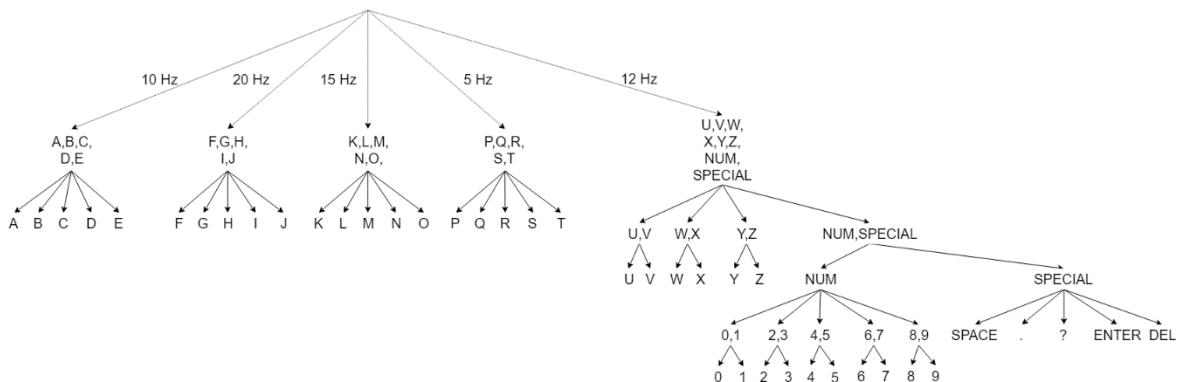


Figure 33 BCI user interface selection tree

The above selection tree is not the most efficient layout of all of the choices but will serve to show the functionality of the BCI. Changing the selection tree to have all vowels and common special inputs such as "DEL" at the top level, reducing the number of selections for these common selections would be more efficient. If there are not enough choices to fill each box i.e. the "U" and "V" branch, the unused checker boxes will allow the user to return to the top level.

On start-up, the average frame rate is recorded and the number of frames at this frame rate to flash each stimulus is calculated. This means the SSVEP frequency may be slightly off if the frame rate is not the expected value. The PsychoPy module provides a function which blocks until the screen has refreshed so this is used to keep track of the number of frames passed and which checker boxes need to be updated. Every object on the screen needs to be redrawn every screen refresh so the main loop code must be quick and efficient so that it does not take longer than a frame period, for a 60 Hz screen this is 16.67 ms. The main code loop was timed and the maximum value observed was 8 ms so this shouldn't prove a problem.

The actual flashing frequency of each checker box was measured to see if it was constant and accurate. This was done by creating a voltage divider using a phototransistor (SFH 310) and a resistor and holding the phototransistor to the checker boxes on screen. The rise and fall time of the SFH 310 was less than 12 μ s so would be fast enough for this application. The square produced was monitored over time to see if there was any variance in the pulse width. After this, the pulse width was measured to compare against the flashing frequency. An example of measuring the 20 Hz SSVEP stimulus (10 Hz checker box) can be seen in Figure 34.



Figure 34 Square produced from measuring 10 Hz checker box flashing using phototransistor (a) Four periods (b) Two periods.

The pulse width during the tests was never the intended value but was always within 0.5 Hz from the value, this can partly be attributed to the measured frame rate never being exactly 60 Hz. For example in the above figure, the measured frequency was 19.60 Hz instead of 20 Hz.

3.2.2.3 Recording baseline and feature extraction

Also on start-up, the interface instruction the user to stare into the centre of the screen while a baseline is recorded for 30 seconds. An FFT is calculated on every last second and the magnitude of each of the stimulus frequencies is stored. After the 30 seconds have elapsed, all of the stored values are used to create a probability density function (PDF) using the most appropriate PDF which was found from using a large data set of no-stimulus data. The scipy module provides a function to find the MLE parameters of a given PDF using a set of data. A gamma distribution was found to fit the no-stimulus histograms best, this is discussed in the results chapter.

```
# Calculate shape, scale and location of gamma distribution
parameters_g = st.gamma.fit(freq_sig_base_val[i])
```

```
# Calculate gamma pdf
fitted_pdf = st.gamma.pdf(x, parameters_g[0], parameters_g[1],
parameters_g[2])
```

A large dataset of stimulus values was also collected and the normalised means and standard deviations were calculated. Using the relative means and standard deviations of the FFT magnitude at the SSVEP frequencies in the presence and not in the presence of stimulus, the expected PDF of the FFT magnitudes in the presence of a stimulus can be created using the 30 seconds of no-stimulus data.

In total there are five no-stimulus PDFs and five stimulus PDFs, one for each stimulus frequency. The naïve Bayes classifier can then be used to calculate the probability that the user is looking at a stimulus. This is more complicated compared to the example given in section 2.2.7 as there are now six classes rather than two. For simplicity, the calculations could be split up into whether the user is looking at stimulus X or is not looking at stimulus X. The prior probability ($p(C_k)$) is unknown in this case but as the user must be looking or not looking at the stimulus, a probability of 0.5 can be used initially.

$$p(LS|x) = \frac{p(x|LS) * p(LS)}{p(x|LS) * p(LS) + p(x|NLS) * p(NLS)} \quad (17)$$

LS Looking at stimulus

NLS Not looking at stimulus

This can be repeated for all five stimulus frequencies and then the highest probability above a threshold probability can be taken to decide if the user is currently looking at a stimulus. If none of the probabilities for all of the stimuli are greater than the threshold probability, it is likely that the user is not looking at a stimulus. This threshold would need to be determined but would have to be greater than 0.5, i.e. the presence of the stimulus is more probable than no-stimulus.

A more complex calculation could be done including all of the classes but this would take more time to develop. The first method was also never implemented or proved due to the time limitation of the project.

A simpler method was implemented and briefly tested while the large dataset was being collected. When it was implemented, there was enough no-stimulus data to choose an appropriate PDF but there was not enough stimulus data to choose a stimulus PDF. This method also recorded a baseline for 30 seconds and used this data to calculate a complementary cumulative density function (CCDF), also called a survival function, which can be given as either:

$$\begin{aligned} F(x) &= \Pr_x[X > x] \\ F(x) &= 1 - \int_{-\infty}^x f(\mu)d\mu \end{aligned} \quad (18)$$

Pr[X>x] Probability that a value is greater than x of a PDF

f(μ) Normalised likelihood at μ of a PDF

A threshold probability was taken as 0.1 meaning that if a new FFT magnitude at a stimulus frequency was less than 10% likely to occur if no-stimulus was present, it may be caused by a stimulus. If two consecutive FFT magnitudes at the same stimulus frequency were past this threshold, the checker box corresponding to this stimulus was

selected. This was briefly tested and when a stimulus was detected, which was roughly 30% of the time when the stimulus was present, the correct stimulus was selected roughly 50% of the time out of five stimuli. This method was used in the rudimentary alpha wave BCI discussed in the following chapter but it was decided that this was not the correct method for SSVEP detection and no more work was done on it.

4 Testing and results

This chapter details the testing of the circuit and software on a user, circuit and software validation and characterisation was discussed in the previous chapter. All of the circuit and software tests were tested on one user. The electrode-skin impedance was not measured during any of the tests.

4.1 Testing EEG measuring circuit with ECG

To validate that the circuit could measure biopotentials, it was first tested as an ECG measurement circuit as these potentials are tens to hundreds of times bigger in magnitude than an EEG [11] and standard ECG electrode pads could be used. Using the ECG pads will make it easier to make a good connection as hair won't be in between the electrode and the skin, as would be the case for the EEG measurement. The three electrodes were connected in the lead II Einthoven's triangle configuration shown in Figure 35 as it produces the largest signal out of the three [47].

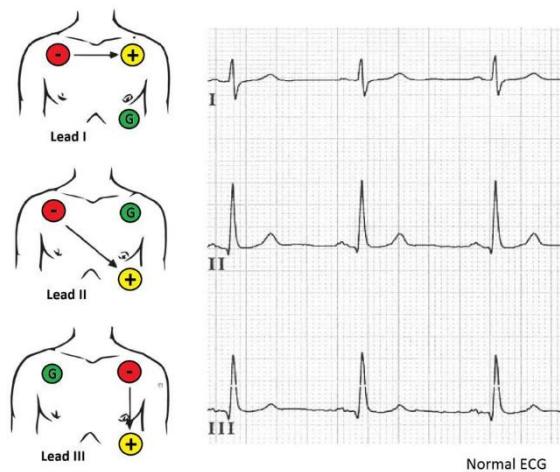


Figure 35 Einthoven's triangle configurations for measuring ECG – Image taken from HRW [48]

The ECG was recorded with and without the DRL to see its effect in real testing conditions. Images from the eegScope during this test can be seen in Figure 36.

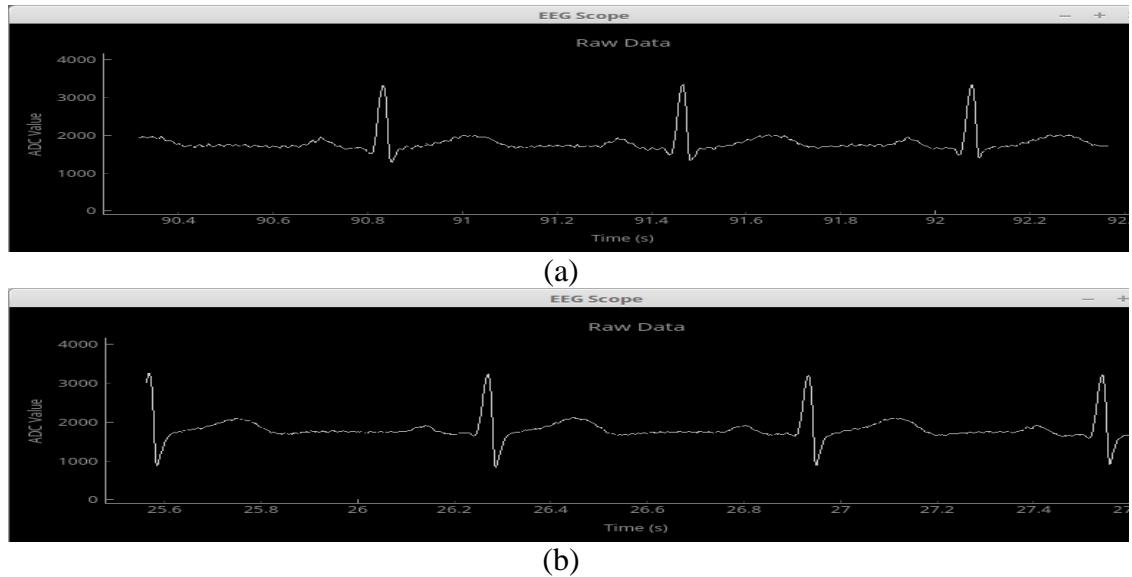


Figure 36 ECG measurement (a) without using DRL. (b) with using DRL.

Both the traces with and without the DRL show a good ECG signal but the DRL contains less interference. After the circuit was validated by measuring an ECG and seeing that the DRL made an improvement to the signal, the low-cost electrodes were tested with the DRL. The electrodes were filled with “Signa gel, Electrode Gel” and the held in place with tape. There was no skin preparation before the test. An image from the eegScope is shown in Figure 37.

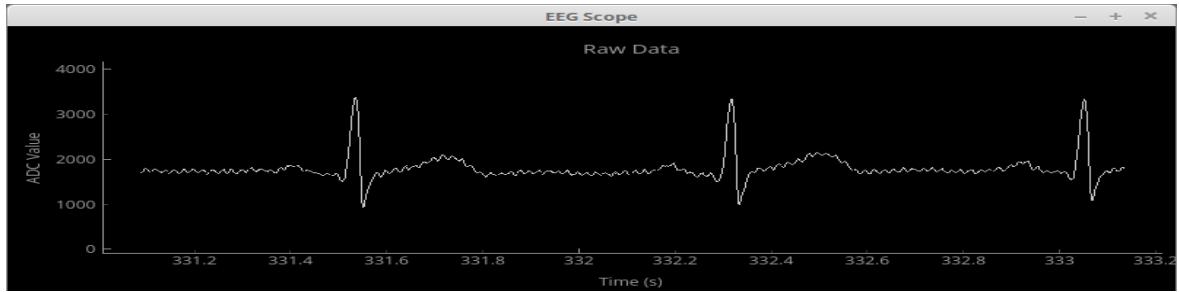


Figure 37 ECG measurement using low-cost electrodes and DRL.

There is 50 Hz interference on top of the ECG which is most likely caused by a greater electrode-skin impedance compared to the normal ECG electrode pads but the ECG is still clearly visible meaning the low-cost electrodes are viable for measuring biopotentials.

4.2 Measuring alpha waves

The next test was to see if the circuit was capable of measuring and amplifying an EEG signal while limiting 50 Hz interference. The easiest to trigger a recognisable EEG response is alpha waves as mentioned previously. The user modulated their alpha wave activity by opening and closing their eyes. A large alpha wave response can be measured in the occipital lobe [27] so the positive and negative electrodes were attached to O1 and O2. A second reason was that O1 and O2 are close to each other so that the wires could be twisted right up to the electrodes, reducing the chance of EMI on the wires. The test procedure went as follows:

1. The DRL was connected to the right mastoid using an ECG pad
2. Nuprep abrasive gel was applied to O1 and O2 to clear oils and dead skin which would increase the electrode-skin impedance
3. Two low-cost electrodes were filled with Signa gel
4. The two electrodes were placed on O1 and O2 and held in place with a swimming cap
5. The test was started and the user kept their eyes open for 60 seconds
6. The user then closed their eyes for 30 seconds

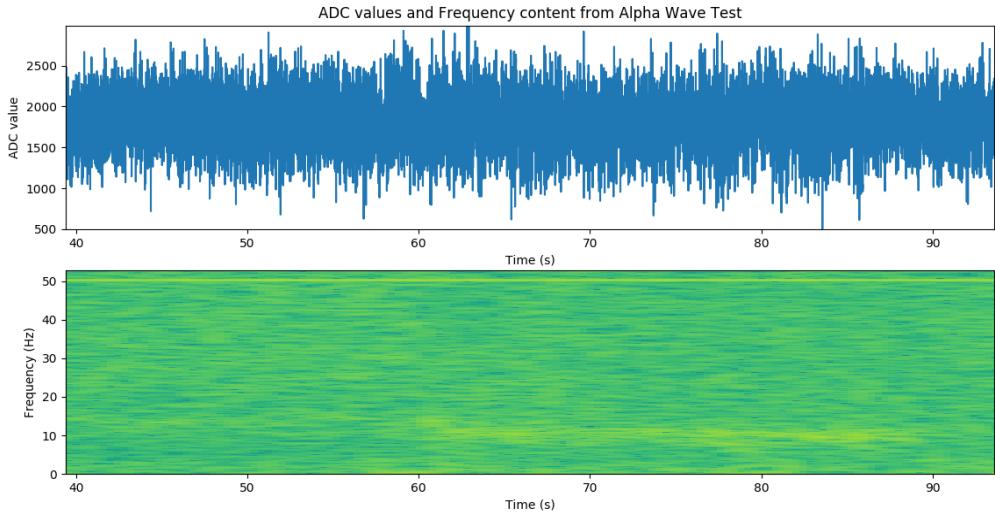


Figure 38 Time domain and spectrogram of EEG measured between O1 and O2. The user kept their eyes open from 0-60 second then closed them from 60-90 second and opened them until the end of the recording. From 60-90 second in the spectrogram, a yellow line can be seen around 10 Hz which is caused by the increase in alpha wave activity.

The spectrogram in Figure 38 shows the frequency content of the EEG over time. At roughly 60 seconds, there is an increase in activity around 10 Hz for 30 seconds (yellow line from 60-90 seconds) which correspond to the user closing their eyes. A constant 50 Hz can be seen throughout the test. A video recording of the real-time plot was taken and the real-time FFT also showed an increase in activity around 10 Hz at the 60 seconds mark showing that the frequency content of the EEG can be processed in real time.

4.3 Alpha wave BCI

As the alpha waves were easy to trigger and resulted in a large magnitude relative to the surrounding frequencies, a rudimentary alpha wave BCI was tested using the CCDF method described in section 3.2.2.3. The user interface was changed to have two white boxes, one with the word “Yes” and one with the word “No” in it. The user would control the size of the boxes depending on their alpha wave activity controlled by opening and closing their eyes. Low alpha wave activity would result in the “No” box being bigger than the “Yes” box and vice versa. This BCI could be used to communicate “Yes” or “No” answers and the questioner could see the answer by looking at the size of the two boxes. The electrode connections and setup were the same as the alpha wave test in the previous section. The user stared into the centre of the screen for the first 30 seconds to record the baseline and then opened or closed their eyes to change the size of the boxes. There was no questioner in this experiment, it was

just a proof of concept that the hardware and software were capable of communicating a user's brain activity to a PC. A link to a video⁴ demonstrating this BCI is in the footnote.

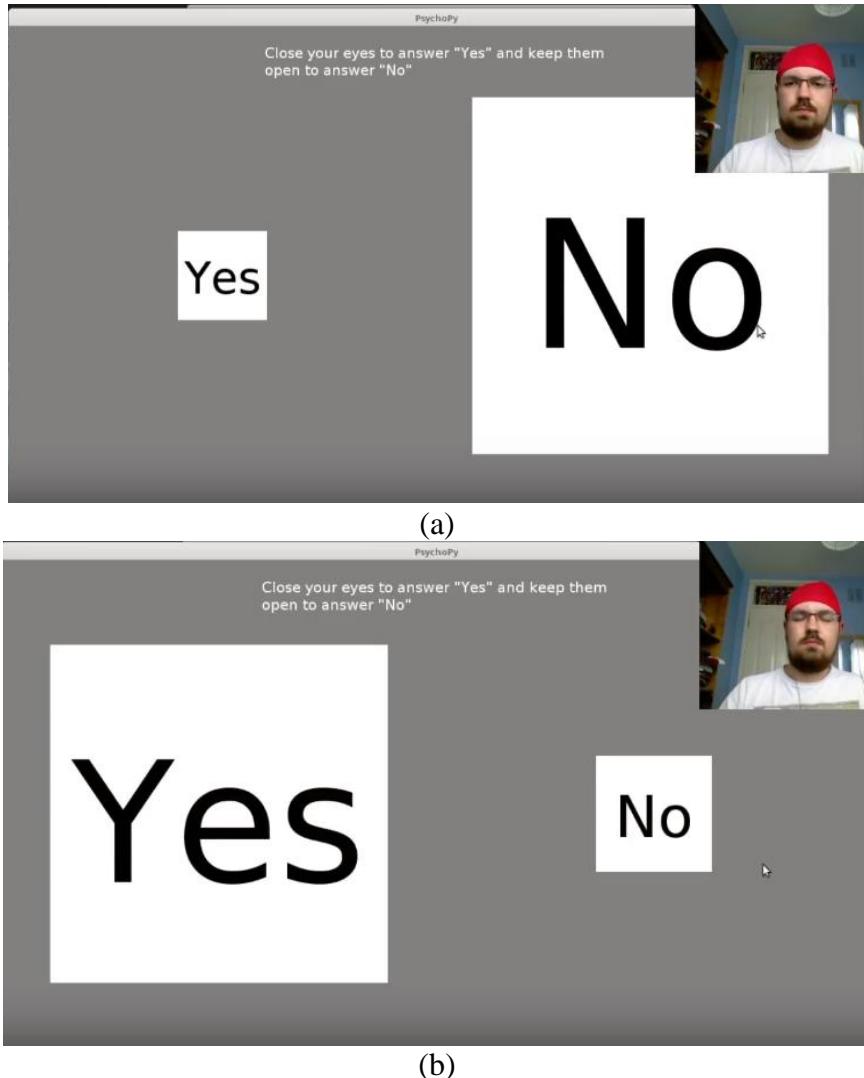


Figure 39 Demonstrating a rudimentary alpha wave BCI. (a) Eyes are open so low alpha wave activity and "No" box is large. (b) Eyes are closed and alpha wave activity is increased so the "Yes" box is large.

4.4 Fitting probability density functions

A large dataset would be required to create an accurate model for the method suggested in section 3.2.2.3. The data was collected by presenting the user with a single large checker box flashing at a single frequency shown in Figure 40.

⁴ https://youtu.be/Ehdn_71upWc

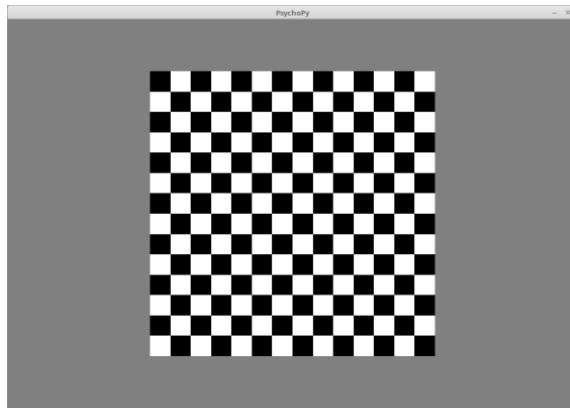


Figure 40 Single large flashing checker box used for triggering SSVEPs

The test procedure went as follows:

1. The DRL was connected to the right mastoid and the negative electrode was connected to the left mastoid, both using ECG pads.
2. Nuprep abrasive gel was applied to Oz to clear oils and dead skin which would increase the electrode-skin impedance
3. One low-cost electrodes was filled with Signa gel
4. The low-cost electrode was placed on Oz and held in place with a swimming cap
5. The test was started and the user stared at a blank screen for 30 seconds to record a no-stimulus baseline
6. The user then stared at a single large checker box stimulus flashing at a single frequency for 120 seconds to collect the stimulus data

This was repeated for all of the five frequencies used in the main application. Oz was chosen as it has been recorded to produce the largest response to SSVEPs [27] and initial tests measuring across O1 and O2 yielded no SSVEP activity. All of the data used were validated by plotting two FFTs, one of the whole no-stimulus duration and one of the whole stimulus duration to see if there was a peak at the stimulus frequency compared to the no-stimulus FFT. A second validation test was to plot a spectrogram of the whole test including both the no-stimulus and stimulus duration to see if the stimulus frequency was present during the time the user was looking at the stimulus.

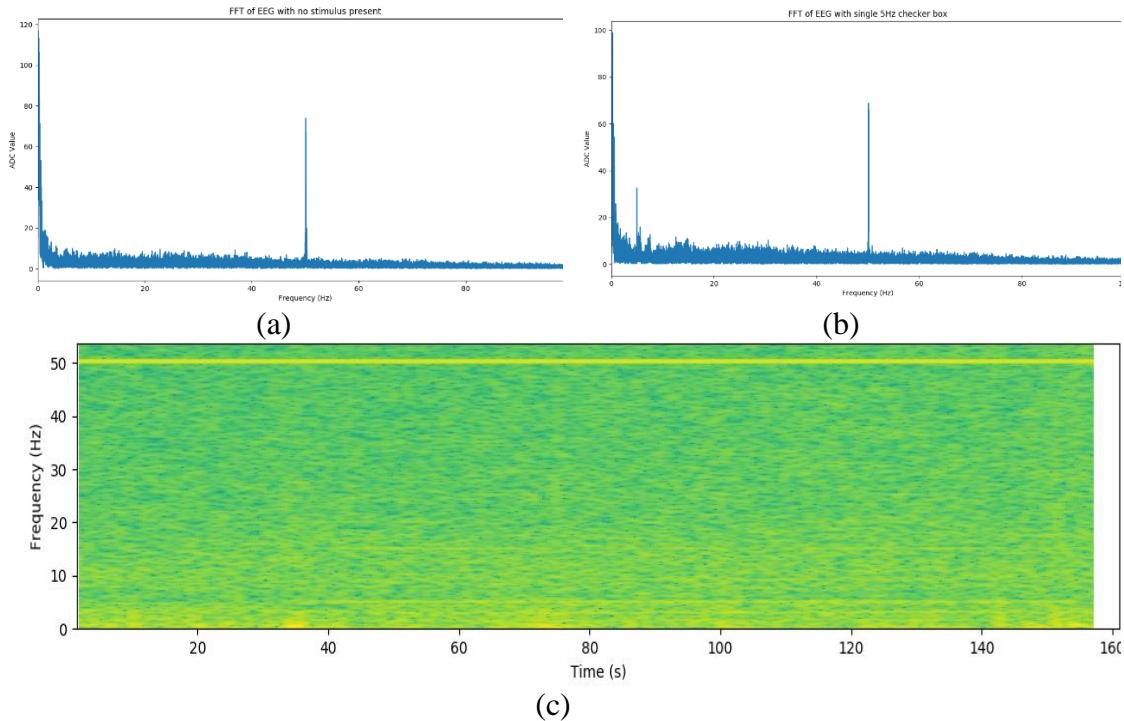


Figure 41 (a) FFT of EEG with no-stimulus. (b) FFT of EEG with 5 Hz stimulus. A spike at 5Hz can be seen in this FFT. (c) Spectrogram of EEG with 5Hz stimulus onset at roughly 35 s. A faint yellow line can be seen from this point in time onwards at 5Hz.

This test was fatiguing over a number of tests, so the full dataset were collected over multiple sessions. Because the data was collected over multiple sessions, the gain settings, electrode-skin impedance and interference could differ causing the magnitude distribution to differ between tests. Because of this, the results were normalised so that they could be combined and compared. This was done by calculating the mean FFT magnitude of each of the SSVEP frequencies when no-stimulus was present for each session. All of the FFT magnitudes of that session were normalised to the no-stimulus mean of that session. In total, 1016 s of data were used to create the no-stimulus histograms and 761 s of data for the stimulus histograms.

Five different PDFs were tested against the histograms and they were assessed visually by how well they matched the histogram and by calculating which of the PDFs yielded the smallest value from the negative log likelihood function (NNLF) in scipy.

```
# Find the pdf which yields smallest negative loglikelihood function
# Code from stack overflow - Martin
# https://stackoverflow.com/questions/21623717/fit-data-to-all-possible-distributions-and-return-the-best-fit
distributions = [st.norm, st.rayleigh, st.gamma, st.chi, st.lognorm]
nnlfss = []
for distribution in distributions:
    pars = distribution.fit(buff)
    nnlf = distribution.nnlf(pars, buff)
    nnlfss.append(nnlf)
results = [(distribution.name, nnlf) for distribution, nnlf in
zip(distributions, nnlfss)]
best_fit = sorted(zip(distributions, nnlfss), key=lambda d: d[1])[0]
print("Best fit reached using {}, NNL value:
{}".format(best_fit[0].name, best_fit[1]))
```

The 12 Hz no-stimulus histogram with five fitted PDFs is shown in Figure 42.

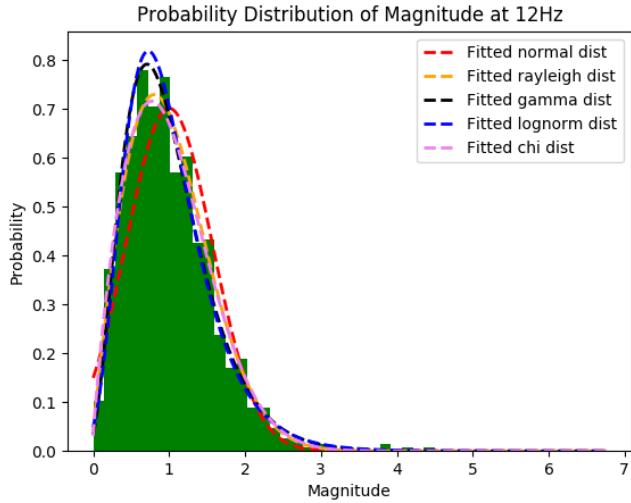


Figure 42 Normalised FFT magnitude histogram of EEG at 12 Hz (no-stimulus) with fitted PDFs

The gamma distribution (black) and the log-normal distribution (blue) match the histogram very well but the gamma PDF yielded the smallest NNLF value. The gamma PDF yielded the smallest NNLF for all frequencies but 5 Hz when no-stimulus was present. A Log-normal PDF matched the 5 Hz histogram slightly better than the gamma PDF but the gamma PDF was chosen to model the no-stimulus PDFs as it best fit all of the frequencies overall. The 12 Hz stimulus histogram with five fitted PDFs is shown in Figure 43.

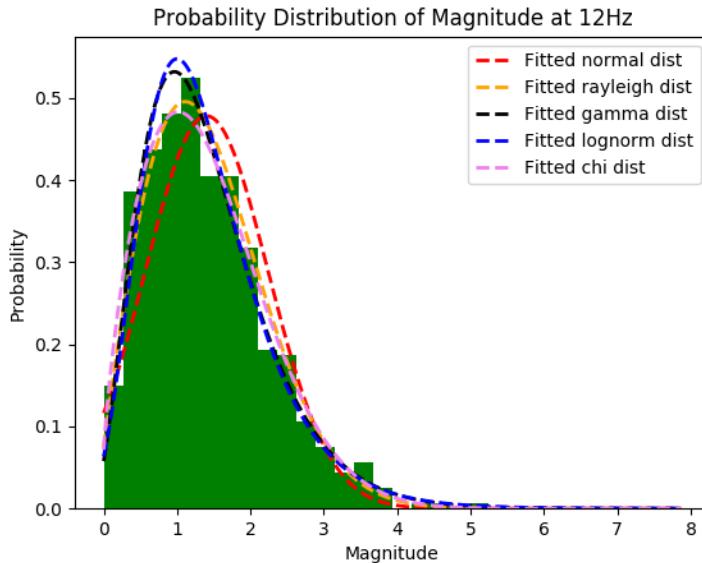


Figure 43 Normalised FFT magnitude histogram of EEG at 12 Hz (12 Hz stimulus) with fitted PDFs

The PDF which yielded the smallest NNLF for all of the stimulus histograms was the chi PDF (pink) so this was chosen to model the FFT magnitude in the presence of stimulus. All of the histograms with fitted PDFs can be seen in Appendix G – Histograms and probability density functions and the code for generating the histograms and PDFs is in CD at the back of the appendix.

The mean and standard deviation for each of the histograms were calculated and are shown in Table 3.

Table 3 Table of normalised means and standard deviations for no-stimulus and stimulus histograms

Frequency (Hz)	No-stimulus		Stimulus	
	Mean	Standard Deviation	Mean	Standard Deviation
5	1	0.63	1.71	0.81
10	1	0.57	1.23	0.65
12	1	0.57	1.40	0.83
15	1	0.53	1.39	0.75
20	1	0.57	1.55	0.88

The mean for all of the stimulus data is greater than the no-stimulus data which would be expected but the larger standard deviation for most would mean the PDFs for the stimulus data will be more spread out and may not have a large separation from the no-stimulus PDFs. Not having a large separation will make differentiating a stimulus value from a no-stimulus value more difficult using the naïve Bayes classifier.

The fitted PDFs for each of the frequencies with no-stimulus and with stimulus were plotted to compare the separation between them. 5 Hz showed the best separation and can be seen in Figure 44.

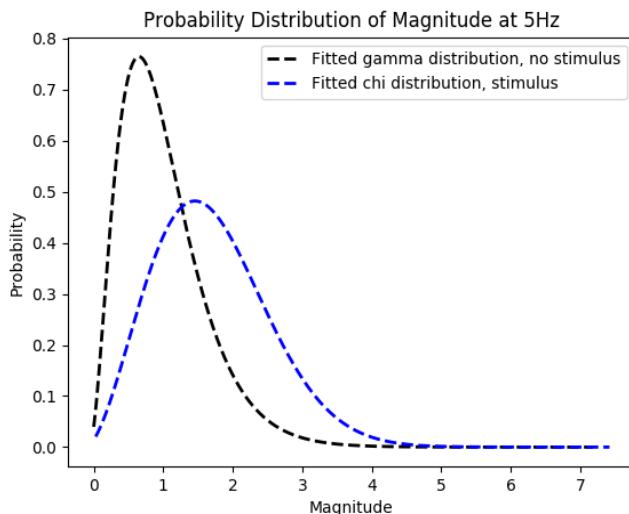


Figure 44 Normalised probability density functions of the FFT magnitudes of EEG at 5 Hz with no-stimulus (black) and with 5 Hz stimulus (blue)

The 5Hz PDFs shows a good separation meaning that it may be possible to compare the probabilities of the two PDFs to assess the presence of SSVEP stimulus. Using the calculated means and standard deviations and along with a baseline recording of no-stimulus data, it may be possible to create the expected stimulus PDF for each frequency and therefore be able to statistically determine the presence of an SSVEP. This is under the assumption that the means and standard deviations calculated do not vary vastly between users and tests and they are an accurate way to describe the FFT magnitudes caused by SSVEP stimulus.

Due to the time limitation of the project, the suggested method for detecting SSVEP stimulus as well as the full BCI software was not tested on users.

5 Conclusion

5.1 Summary

A circuit was designed, built and tested which could measure EEG from a user while limiting interference (measured CMRR of 89 dB at 50 Hz). Low-cost EEG cup electrodes were built and were proven to be viable for measuring EEG. The circuit was capable of measuring alpha waves modulated by the user opening and closing their eyes and was capable of measuring SSVEPs stimulated by the software created. The total cost of the circuit was €30.08 and the low-cost electrode were €0.70 each so the objective of keeping the total cost of the hardware below €50 was achieved.

The software (virtual keyboard) was capable of stimulating SSVEPs in a user at five unique frequencies but a reliable method for SSVEP detection in real time was not developed or tested. A rudimentary alpha wave BCI was demonstrated showing the hardware and software were capable of transforming EEG from the user to actions on a PC. The hardware designs and source code were published to Github under the Apache licence 2.0.

A large dataset of the FFT magnitudes of the EEG at each of the SSVEP frequencies were measured at Oz in the presence and not in the presence of SSVEP stimulus. PDFs were fitted to each of the no-stimulus and stimulus histograms for each frequency and the mean of the stimulus PDFs was greater than the no-stimulus PDFs which would be expected. A method for approximating a stimulus PDFs given a small dataset of no-stimulus using the relative means and standard deviations from the large dataset was suggested but not tested. Lastly, a method for classifying the presence of SSVEPs using a naïve Bayes classifier was suggested but not tested.

In summary, two of the four objectives were fully achieved, these were to:

- Develop a low-cost circuit capable of EEG measurement while limiting interference.
- Disseminate the hardware designs and source code under an open-source license.

A circuit for measuring EEG while limiting interference was built and tested and the hardware designs and source code was published to GitHub under the Apache licence 2.0.

The other two objectives which were achieved to a degree were to:

- Develop a brain-computer interface which can stimulate and react to SSVEPs.
- Develop a statistical method for SSVEP detection.

A brain-computer interface was developed which could stimulate SSVEPs but could not reliably react to them. But a rudimentary alpha wave BCI was developed and was functional proving the capability of the hardware and software. A statistical method for SSVEP detection was suggested and data were collected which shows the possibility that the method may work but was not implemented or tested.

5.2 Project limitations

5.2.1 Limitation of data and PDFs collected

The large dataset of FFT magnitudes collected and resulting PDFs, means and standard deviations may have limited use for the following reasons. First, the EEG was measured from Oz which may mean that the PDFs and values calculated only correctly describe this electrode configuration in the presence and not in the presence of SSVEP stimuli. Second, the stimulus data was collected with only one SSVEP stimulus in the user's field of view, the PDFs and values calculated may not correctly describe the FFT magnitudes with other SSVEP stimuli in the user's field of view, as is the case in the BCI developed. The single checker box stimulus was a lot larger than what is used in the BCI interface, this may also affect its reliability. Third, the dataset was collected from one user, the PDFs and calculated values for this user may not represent all users.

The large dataset was also collected over multiple testing sessions and the FFT magnitudes were normalised to the mean magnitude of each SSVEP frequency with no-stimulus of that session so that the data could be combined. This assumed that the only difference between sessions was the amplitude and that the normalised mean and standard deviation of each session would be the same. This may not be true as other interferences and brain processes may have influenced the measurements making the PDFs calculated incorrect.

5.2.2 Possible issue with classifier suggested

The naïve Bayes classifier method suggested does not account for the harmonics of SSVEPs. If the user is looking at a 10 Hz stimulus, the first harmonic of 20 Hz will increase in amplitude and therefore increase the probability it is selected over the correct 10 Hz stimulus.

5.2.3 Number of SSVEP stimuli and harmonics

A limitation stated previously is the number of SSVEP stimuli that can be used on a 60 Hz monitor. Five stimulus frequencies (5, 10, 12, 15, 20 Hz) were chosen in this application and two of them (5 Hz and 10 Hz) have harmonics which match the fundamental frequency of other stimuli further increasing the error rate of the system.

5.2.4 Holding the electrodes in place

The method for holding the electrodes to the head leads a lot to be desired. The swimming cap holds the electrodes in place somewhat tightly but they can slip easily. Also attaching the electrodes to places other than the back of the head can be difficult.

5.3 Future work

A better method for holding the electrodes to the head would make the system much easier to setup and use. The swimming cap could have holes cut-out and have a way for the electrodes to snap onto it, similar to a standard EEG hat.

Making a PCB for the circuit would increase the robustness of the circuit. Placing the circuit in a grounded enclosure would help limit the interference on the circuit. Adding more channels would allow more applications of the circuit such as SMR-based BCIs.

The power consumption of the circuit could be improved to increase battery life, the optoisolation circuit is assumed to be (current consumption was not measured) the most power hungry due to the low resistances used. A battery circuit could be built to supply a regulated voltage to the circuit which would drop out when the voltage reached a certain level. Unexpected behaviour was seen when the battery voltage was low. The 470 μ F capacitor takes some time to charge up when the circuit is powered, using a diode connected to a digital output which is high on start-up would speed up the charge time.

More data could be collected for other users to assess if the PDFs, means and standard deviations from the one user could accurately describe other users. The naïve Bayes classifier described could be fully implemented and tested initially using a single checker box and if that continuously chooses the correct stimulus, test it with the virtual keyboard. The layout for the selection on the keyboard could be improved to have the most common characters at the top level and to include a back button if the wrong checker box is selected.

6 Bibliography

- [1] M. Clinic. (2016, January 16). *Brain Anatomy, Anatomy of the Human Brain*. Available: <http://www.mayfieldclinic.com/PE-AnatBrain.htm>
- [2] M. G. N. Garcia, T. K. TSONEVA, D. BALDO, D. Zhu, P. H. A. GABRIELSSON, and K. P. N.V., "Device and method for cognitive enhancement of a user," 2014. Available: <https://www.google.com/patents/EP2681729A1>.
- [3] L. Haas, "Hans Berger (1873–1941), Richard Caton (1842–1926), and electroencephalography," (in eng), *Journal of Neurology, Neurosurgery, and Psychiatry*, vol. 74, no. 1, p. 9, Jan 2003.
- [4] S. N. Abdulkader, A. Atia, and M.-S. M. Mostafa, "Brain computer interfacing: Applications and challenges," *Egyptian Informatics Journal*, vol. 16, no. 2, pp. 213-230, 2015/07/01/ 2015.
- [5] imotions. (2017). *EEG Headset Prices – An Overview of 15+ EEG Devices*. Available: <https://imotions.com/blog/eeg-headset-prices/>
- [6] T. O. Project. *The ModularEEG*. Available: <http://openeeg.sourceforge.net/doc/modeeg/modeeg.html>
- [7] OPenBCI. *Ganglion Board (4-channels)*. Available: <https://shop.openbci.com/products/pre-order-ganglion-board>
- [8] E. V. Hippel, Open Source Software: Innovation By and For Users – No Manufacturer Required!: Massachusetts Institute of Technology, 2003. [Online]. Available: <http://ebusiness.mit.edu/research/papers/133%20von%20hippel%20OSS%20innovation.pdf>.
- [9] D. Ehls, *Joining decisions in open collaborative innovation communities: A discrete choice study*. Springer Science & Business Media, 2013.
- [10] H. Yuan and B. He, "Brain-Computer Interfaces Using Sensorimotor Rhythms: Current State and Future Perspectives," (in eng), *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 5, pp. 1425-35, May 2014.
- [11] J. Malmivuo and R. Plonsey, *Bioelectromagnetism - Principles and Applications of Bioelectric and Biomagnetic Fields*. 1995.
- [12] *Standard Consumer Safety Specification for Toy Safety*, 2018.
- [13] *DIRECTIVE 2014/53/EU OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 16 April 2014 on the harmonisation of the laws of the Member States relating to the making available on the market of radio equipment and repealing Directive 1999/5/EC*, 2018.
- [14] NeuroSky. *The Force Trainer II: Hologram Experience*. Available: <https://images-na.ssl-images-amazon.com/images/I/A1AOlb0qV1L.pdf>
- [15] *IEC 60601-1:2005+A1:2012(E)*, 2012.
- [16] *Apache License, Version 2.0*, T. A. S. Foundation, 2004.
- [17] U. X. Chaudhary, B. Silvoni, S. Cohen, LG. Birbaumer, N., "Brain–Computer Interface–Based Communication in the Completely Locked-In State," *PLoS Biol*, vol. 15, no. 1, 2018.
- [18] L. F. Nicolas-Alonso and J. Gomez-Gil, "Brain Computer Interfaces, a Review," (in eng), *Sensors (Basel)*, vol. 12, no. 2, pp. 1211-79, 2012.
- [19] H.-J. Hwang, J.-H. Lim, Y.-J. Jung, H. Choi, S. W. Lee, and C.-H. Im, "Development of an SSVEP-based BCI spelling system adopting a QWERTY-style LED keyboard," *Journal of Neuroscience Methods*, vol. 208, no. 1, pp. 59-65, 2012/06/30/ 2012.
- [20] J. J. Vidal, "Toward Direct Brain-Computer Communication," *Annual Review of Biophysics and Bioengineering*, vol. 2, no. 1, pp. 157-180, 1973.
- [21] K. Medical. (2018). *Rhythmlink Deep Cup Dome Electrodes*. Available: <https://kandel.com.br/eletrodos/eeg/deep-cup/>
- [22] G. Technologies. (2018). *g.SAHARAelectrode (16mm) Active Dry Electrodes*. Available: <http://www.gtec.at/Products/Electrodes-and-Sensors/g.Electrodes-Specs-Features>
- [23] J. Saab, B. Battes, and M. Grosse-Wentrup, "Simultaneous EEG Recordings with Dry and Wet Electrodes in Motor-Imagery," presented at the 12th Conference of Junior Neuroscientists of Tübingen, Heiligkreuztal, Germany., 2011/10, 2011. Available: <http://mlin.kyb.tuebingen.mpg.de/BCI2011JS.pdf>
- [24] Bio-Medical. (2018). *Electrodes for EMG/EEG/ECG/EKG*. Available: <https://bio-medical.com/supplies/electrodes.html>
- [25] g. t. M. Engineering. (2018). *Comparision: active versus passive electrodes*. Available: <http://www.gtec.at/News-Events/Newsletter/Newsletter-September-2010-Volume-29/articles/Comparision-active-versus-passive-electrodes2>
- [26] P. A. Abhang, B. W. Gawali, and S. C. Mehrotra, "Chapter 1 - Introduction to Emotion, Electroencephalography, and Speech Processing," in *Introduction to EEG- and Speech-Based*

Emotion Recognition, P. A. Abhang, B. W. Gawali, and S. C. Mehrotra, Eds.: Academic Press, 2016, pp. 1-17.

- [27] A. M. Norcia, L. G. Appelbaum, J. M. Ales, B. R. Cottreau, and B. Rossion, "The steady-state visual evoked potential in vision research: A review," (in eng), *Journal of Vision*, vol. 15, no. 6, 2015.
- [28] S. P. Kelly, E. C. Lalor, C. Finucane, G. McDarby, and R. B. Reilly, "Visual spatial attention control in an independent brain-computer interface," *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 9, pp. 1588-1596, 2005.
- [29] L. Zhang, X. j. Guo, X. p. Wu, and B. y. Zhou, "Low-cost circuit design of EEG signal acquisition for the brain-computer interface system," in *2013 6th International Conference on Biomedical Engineering and Informatics*, 2013, pp. 245-250.
- [30] M. Leonardo Casal and Guillermo La, "Skin-electrode impedance measurement during ECG acquisition: method's validation," *Journal of Physics: Conference Series*, vol. 705, no. 1, p. 012006, 2016.
- [31] M. Jansen *et al.*, "Motion-related artefacts in EEG predict neuronally plausible patterns of activation in fMRI data," *Neuroimage*, vol. 59, no. 1-3, pp. 261-270, 02/23/received

06/28/revised

06/30/accepted 2012.

- [32] A. B. Usakli and S. Gurkan, "Design of a Novel Efficient Human–Computer Interface: An Electrooculogram Based Virtual Keyboard," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 8, pp. 2099-2108, 2010.
- [33] A. B. Usakli, "Improvement of EEG Signal Acquisition: An Electrical Aspect for State of the Art of Front End," *Computational Intelligence and Neuroscience*, vol. 2010, 2010.
- [34] B. Fong, M. Harvey, and K. Lagoutchev "Portable EEG Recording Using a Lock-in Amplifier," Electrical And Computer Engineering, University of Illinois, 2014.
- [35] B. Luan, M. Sun, and W. Jia, "Portable Amplifier Design for a Novel EEG Monitor in Point-of-Care Applications," *Proceedings of the IEEE ... annual Northeast Bioengineering Conference. IEEE Northeast Bioengineering Conference*, vol. 2012, pp. 388-389, 2012.
- [36] T. Instruments. (2018). *Three guidelines for designing anti-aliasing filters - Precision Hub - Archives - TI E2E Community*. Available: <https://e2e.ti.com/blogs/archives/b/precisionhub/archive/2015/11/06/three-guidelines-for-designing-anti-aliasing-filters>
- [37] M. Buttlar, J. Hansmann, and A. R. *The ModularEEG*. Available: <http://openeeg.sourceforge.net/doc/moddeeg/modEEGamp-v1.0.png>
- [38] B. B. Winter and J. G. Webster, "Driven-right-leg circuit design," *IEEE Transactions on Biomedical Engineering*, vol. BME-30, no. 1, pp. 62-66, 1983.
- [39] M. R. Ahsan, M. Ibrahimi, and O. Khalifa, *EMG Signal Classification for Human Computer Interaction A Review*. 2009, pp. 480-501.
- [40] R. Saji, K. Hirasawa, M. Ito, S. Kusuda, Y. Konishi, and G. Taga, "Probability distributions of the electroencephalogram envelope of preterm infants," *Clinical Neurophysiology*, vol. 126, no. 6, pp. 1132-1140, 2015/06/01/ 2015.
- [41] Andrew. (2013). *Fitting a Model by Maximum Likelihood*. Available: <https://www.r-bloggers.com/fitting-a-model-by-maximum-likelihood/>
- [42] Quantivity. (2011). *Why Minimize Negative Log Likelihood?* Available: <https://quantivity.wordpress.com/2011/05/23/why-minimize-negative-log-likelihood/>
- [43] K. P. Murphy, "Naive bayes classifiers," *University of British Columbia*, vol. 18, 2006.
- [44] Botland. (2018). *STM32 NUCLEO-F303K8 - STM32F303K8 ARM Cortex M4*. Available: <https://botland.com.pl/stm32-nucleo/4891-stm32-nucleo-f303k8-stm32f303k8-arm-cortex-m4.html>
- [45] Amazon. *BITSCOPE MP01A BNC ADAPTER, MICRO OSCILLOSCOPE*. Available: <https://www.amazon.com/BITSCOPE-MP01A-ADAPTER-MICRO-OSCILLOSCOPE/dp/B01E9QZYBA>
- [46] M. M. Equipment. (2018). *Technomed Reusable EEG Cup Electrode (12/Pkg)*. Available: <https://mfimedical.com/products/technomed-reusable-eeg-cup-electrode>
- [47] T. S. Physiologist. (2015). *The ECG Leads, Polarity and Einthoven's Triangle*. Available: <https://thephysiologist.org/study-materials/the-ecg-leads-polarity-and-einthovens-triangle/>
- [48] H. R. Warner. (2018). *Measure ECG and arrhythmia detection*. Available: <http://www.hrwproject.com/ecg.html>

7 Appendix

7.1 Appendix A - Bill of materials

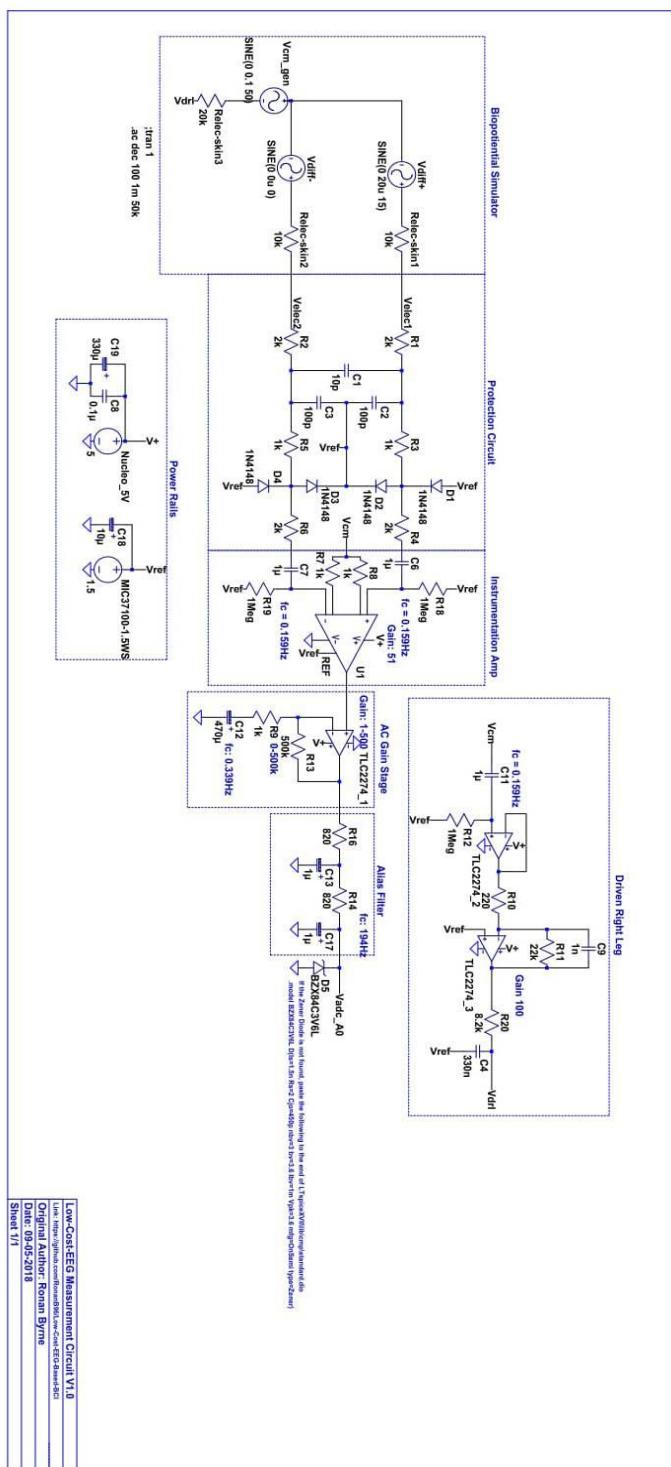
Table 4 Bill of materials (circuit)

Component	Source	Units	Unit Cost (€)	Total Cost (€)
Total Circuit Cost				30.08
NUCLEO-F303K8	Farnell	1	10.98	10.98
TLC2274ACN Quad Operation Amplifier	Farnell	1	1.94	1.94
MIC37100-1.5WS 1.5V Regulator	Farnell	1	0.48	0.48
AD623AN Instrumentation Amplifier	Farnell	1	5.45	5.45
USB-TTL Converter	Ebay	1	1.31	1.31
PC817N Opto-Coupler	Ebay	2	0.023	0.047
1N4148 Diode	Ebay	4	0.021	0.082
BZX84C3V6L Zener Diode	Ebay	1	0.012	0.012
9V Battery	Hardware Shop	1	4.8	4.8
9V Battery Snap	Ebay	1	0.057	0.057
Breadboard	Ebay	1	1.815	1.815
10p Ceramic Capacitor	Ebay	1	0.004	0.004
100p Ceramic Capacitor	Ebay	2	0.004	0.007
1n Ceramic Capacitor	Ebay	1	0.004	0.004
100n Ceramic Capacitor	Ebay	1	0.004	0.004
330n Ceramic Capacitor	Ebay	1	0.004	0.004
1u Ceramic Capacitor	Ebay	3	0.004	0.011
1u Polarised Capacitor	Ebay	2	0.037	0.074
10u Polarised Capacitor	Ebay	1	0.037	0.037
330u Polarised Capacitor	Ebay	1	0.037	0.037
470u Polarised Capacitor	Ebay	1	0.037	0.037
120 Resistor	Ebay	2	0.006	0.012
220 Resistor	Ebay	3	0.006	0.018
820 Resistor	Ebay	2	0.006	0.012
1K Resistor	Ebay	5	0.006	0.029
2K Resistor	Ebay	4	0.006	0.024
8.2K Resistor	Ebay	1	0.006	0.006
22K Resistor	Ebay	1	0.006	0.006
1M Resistor	Ebay	3	0.006	0.018
500k Trimmer	Farnell	1	2.77	2.77

Table 5 Bill of Materials (Electrode)

Component	Source	Units	Unit Cost (€)	Total Cost (€)
Cost Per Electrode				0.70
ECG Electrode Pads	Medisave	1	0.084	0.08
Rubber Shower Hose Washer (10mm)	Amazon	1	0.45	0.45
Button Snaps (13mm)	Hickeys	1	0.1625	0.16

7.2 Appendix B - Full circuit



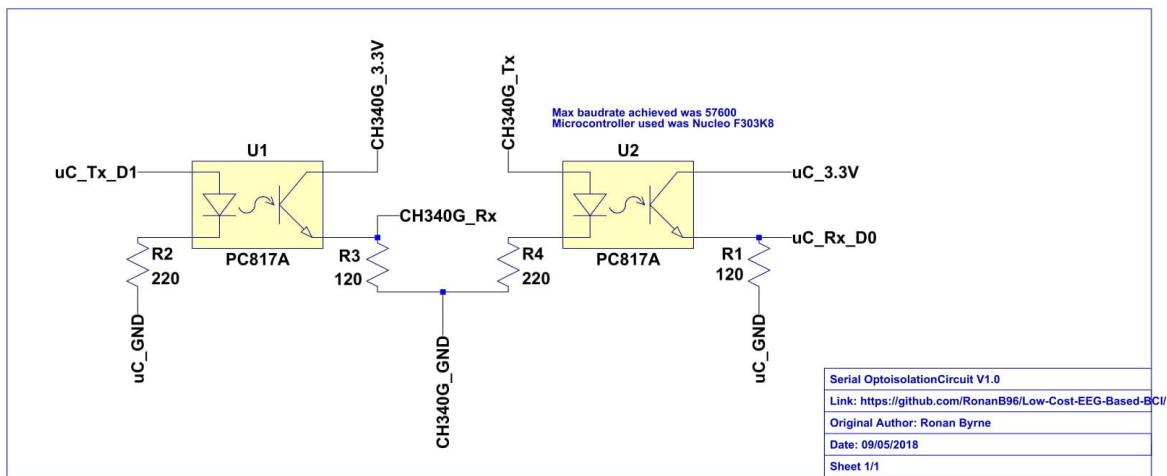


Figure 46 LTspice diagram of optoisolation circuit

7.3 Appendix C - Simulated frequency responses

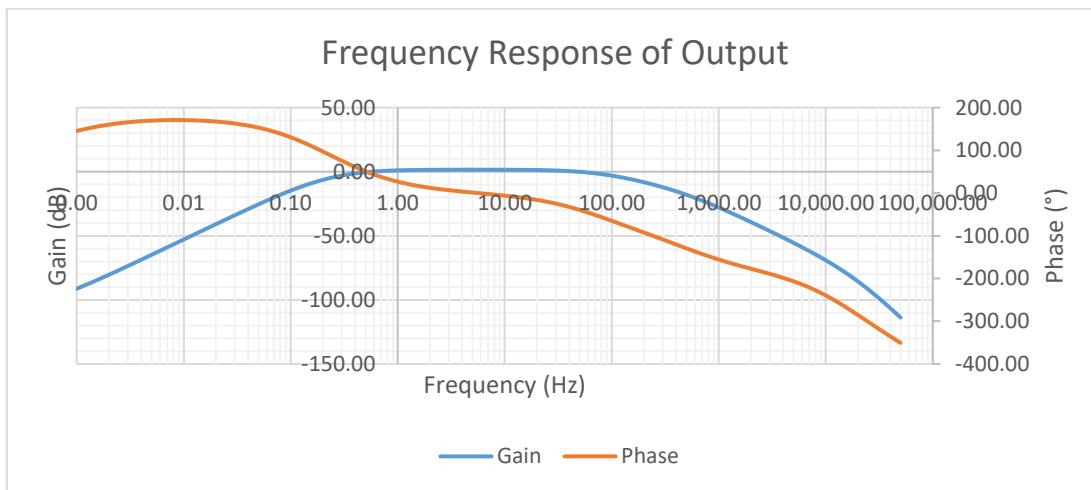


Figure 47 Simulated frequency response of circuit output from a differential input of 50 μ V

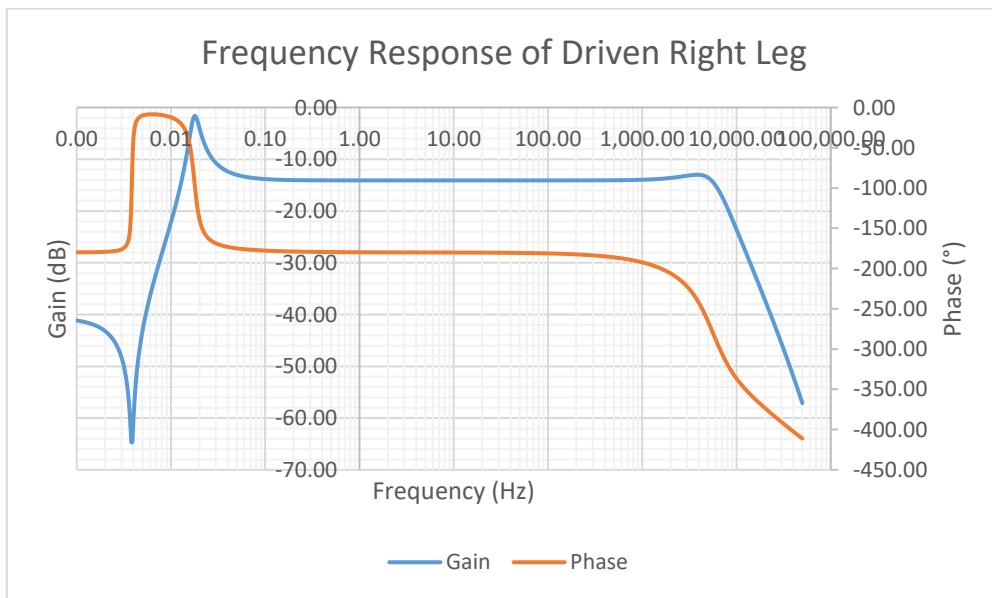


Figure 48 Simulated frequency response of driven right leg from a common mode input of 200 mV

7.4 Appendix D - Nucleo code flow diagram

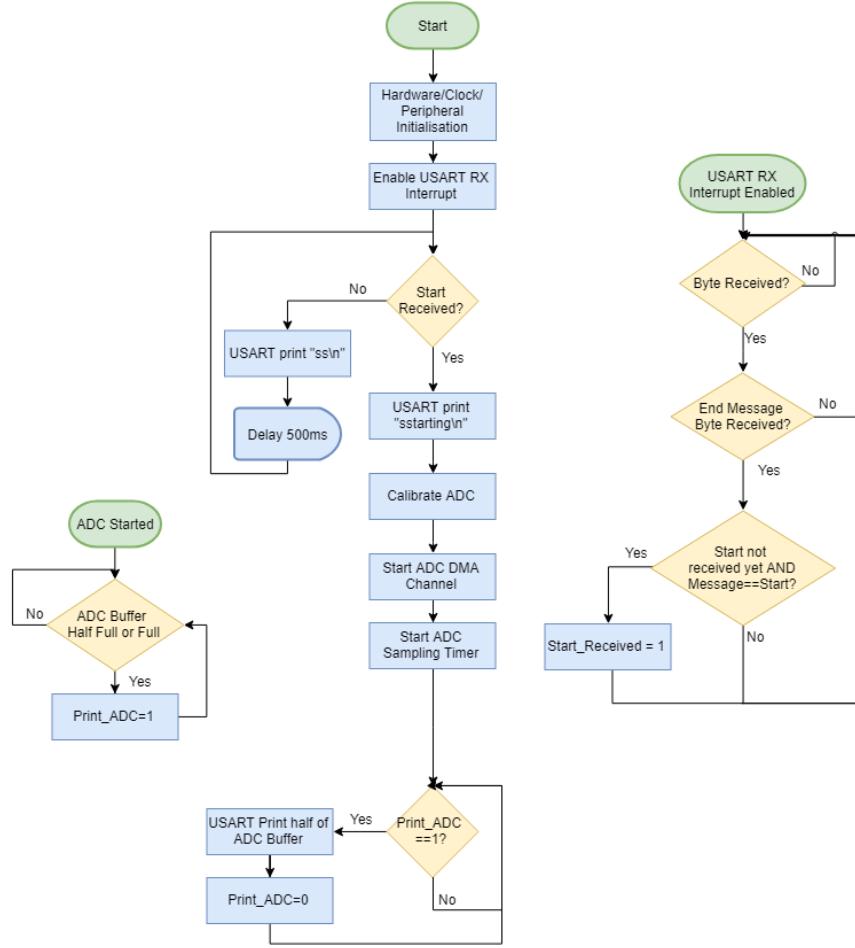


Figure 49 Microcontroller code flow diagram

7.5 Appendix E – BCI code flow diagrams

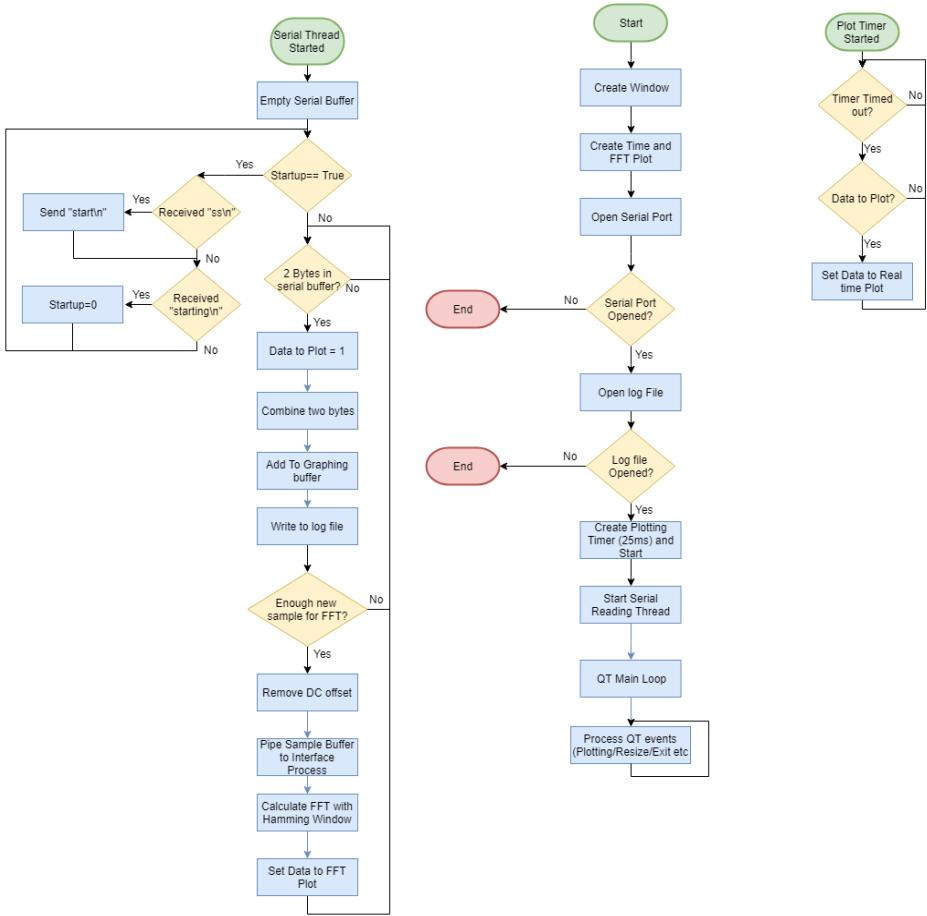


Figure 50 `eegScope.py` code flow diagram

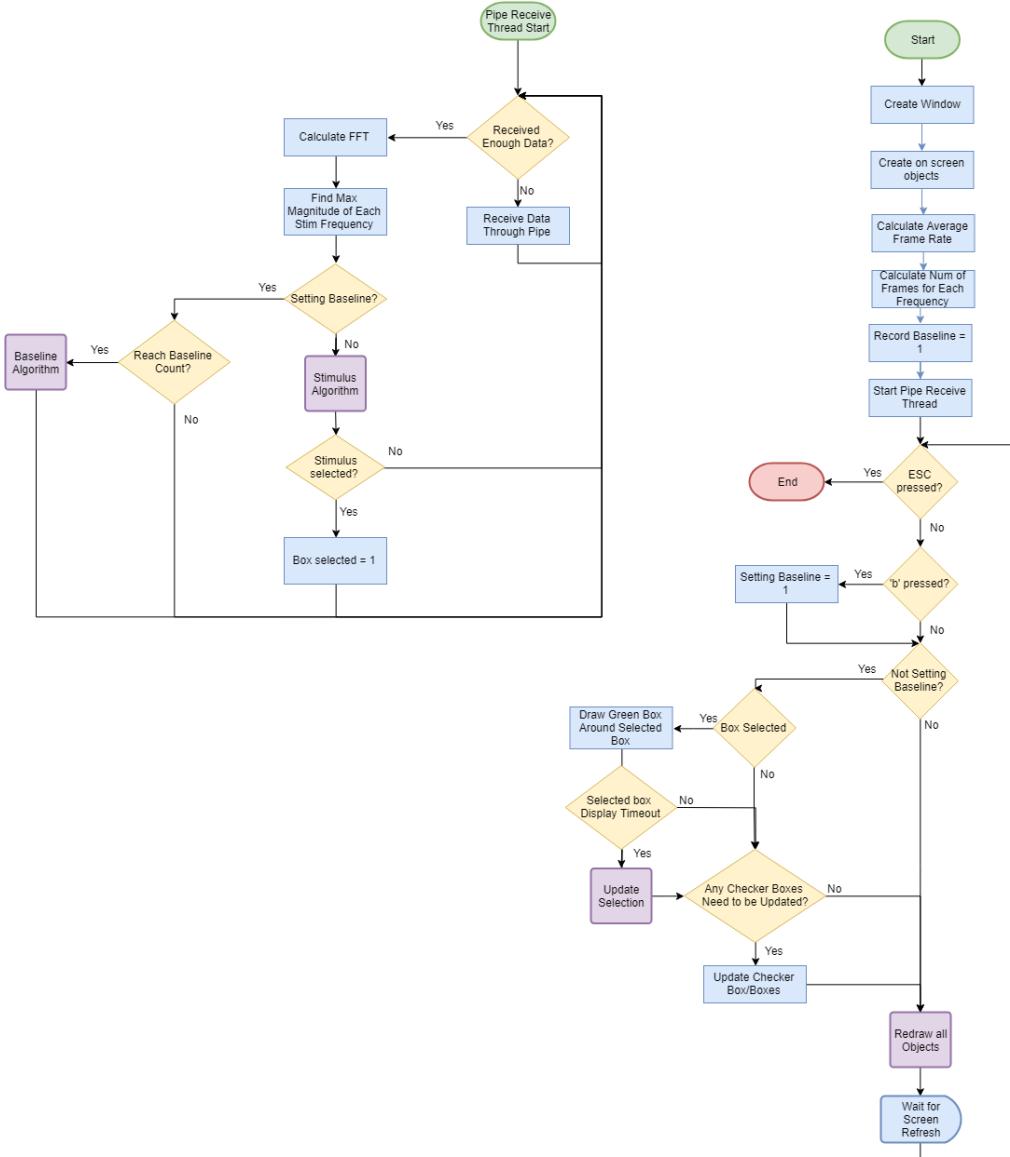


Figure 51 eegInterface.py code flow diagram

7.6 Appendix F – BCI code

7.6.1 eegScope.py

```
# eegScope.py
#
# Receives analog data over serial port and plots the data
# in real time along with a real time FFT
# Author: Ronan Byrne
# Last Updated: 09/05/2018
#
from pyqtgraph.Qt import QtCore, QtWidgets
import pyqtgraph as pg
import numpy as np
import serial
import threading
import time
import sys

class Scope(object):
    def __init__(self, port, pipe):
        self.pipe = pipe
        self.app = QtWidgets.QApplication(sys.argv)
        #self.app.aboutToQuit.connect(self.exit()) # Close start away when uncommented
```

```

# Create Window
self.win = pg.GraphicsWindow()
self.win.setWindowTitle('EEG Scope')

# Add Real time plot
self.main_plot = self.win.addPlot(title="Raw Data")

# Add FFT plot
self.win.nextRow()
self.fft_plot = self.win.addPlot(title="FFT of Raw Data")

self.y_min = 0
self.y_max = 4095
self.Fs = 1000
self.sample_interval = 1/self.Fs

# Main plot setup
self.main_plot_t_start = 0
self.main_plot_t_size = 2047
self.main_plot_t_end = self.main_plot_t_start + self.main_plot_t_size
self.main_plot.setYRange(self.y_min, self.y_max)
self.main_plot.setLabel('left', 'ADC Value', '')
self.main_plot.setLabel('bottom', 'Time', 's')

# Actually create the plot
self.graph = self.main_plot.plot()
self.graph_time = np.arange(self.main_plot_t_start, self.main_plot_t_end/self.Fs,
self.sample_interval)
self.graph_pos = self.graph_time[-1] # Place cursor at the far right of the screen

# FFT variables
self.fft_sample_size = 1000
self.fft_sample_num = 0
self.fft_padding = 5
# Frequency axis
self.fft_freq = np.fft.rfftfreq(self.fft_sample_size * self.fft_padding, 1 / self.Fs)

# FFT graph setup
#self.fft_plot.setYRange(0, 250)
# Frequencies above 100Hz aren't of interest
self.fft_plot.setXRange(0, 100)
self.fft_plot.setLabel('left', 'ADC Value', '')
self.fft_plot.setLabel('bottom', 'Frequency', 'Hz')

self.fft_graph = self.fft_plot.plot()
self.fft_graph_fft_mag = np.zeros(int((self.fft_sample_size*self.fft_padding)/2)+1)

# Calculation Variables
self.first_run = True
self.saved_values = np.zeros(self.main_plot_t_size, dtype='uint16')

self.port = port

# Graphing and serial variables
# Roughly 2 seconds of data
self.graph_N = 2047
self.graph_data_read = False
self.graph_head = 0
self.graph_tail = self.graph_head
self.graph_buff = int(self.y_max / 2) * np.ones(self.graph_N, dtype='uint16')
self.serial_thread = threading.Thread(target=self.serial_read, daemon=True)
self.ser = None
self.plot_timer = None

# Logging variables
self.filename = ''
self.logfile = None

# Start automatically if pipe was given
if self.pipe is not None:
    self.start()

def start(self):
    # Serial variables
    self.ser = serial.Serial(self.port, baudrate=57600, parity=serial.PARITY_NONE,
stopbits=serial.STOPBITS_ONE, bytesize=serial.EIGHTBITS)

    if not self.ser.isOpen():
        print('Failed to open port')
        return

    # Create log file
    self.filename = '/home/ronan/Documents/EEG_csv/' + time.ctime() + '.csv'
    self.filename = self.filename.replace(':', '-')
    # colon not allowed in windows codename
    print('Opening data log file ' + self.filename + ' ...')
    self.logfile = open(self.filename, 'w')

```

```

if not self.logfile:
    print('Failed to open logfile')
    return

# Create a timer to update graph every timeout
self.plot_timer = QtCore.QTimer()
self.plot_timer.timeout.connect(self.update_plot)
# 25ms timer
self.plot_timer.start(25)

# Start thread to read from serial port
self.serial_thread.start()

# Main Loop
self.app.exec_()

```

```

def serial_read(self):
    self.ser.reset_input_buffer()
    start_up = True
    print("waiting for uC")
    while True:
        # Wait for the uC to handshake to ensure byte alignment is correct
        while start_up:
            line = self.ser.readline()
            # decode will not work if the uC is sending analog data
            try:
                line = line.decode('utf-8')
            except UnicodeDecodeError:
                print("Unable to decode, try restart uC")
            print(line)
            # uC is starting, move on
            if line == "sstarting\n":
                start_up = False
            # uC will continuously send ss until start is sent
            elif line == "ss\n":
                self.ser.write('start\n'.encode())
        # uC sends data in 2 bytes
        while self.ser.inWaiting() >= 2:
            # data has been read, so if plot timer is called, it can update
            self.graph_data_read = True
            lowbyte = self.ser.read()
            highbyte = self.ser.read()

            # Shift buffer to the left, this is faster than np.roll
            self.graph_buff[:-1] = self.graph_buff[1:]
            self.graph_buff[-1] = (ord(highbyte) << 8) + ord(lowbyte)

            self.graph_time[:-1] = self.graph_time[1:]
            # Shift cursor
            self.graph_pos += self.sample_interval
            self.graph_time[-1] = self.graph_pos

            # Write value to log file
            self.logfile.write(str(self.graph_buff[-1]) + '\n')

            # Shift head forward
            self.graph_head = (self.graph_head + 1) % self.graph_N

            self.fft_sample_num += 1
            # If we have enough samples for FFT
            if self.fft_sample_num == self.fft_sample_size:
                self.fft_sample_num = 0
                self.calc()

def update_plot(self):
    # Only update plot if data has been read
    if self.graph_data_read:
        self.graph_data_read = False
        # Set new data to plot
        self.graph.setData(self.graph_time, self.graph_buff)

def calc(self):
    # store current buff so it doesn't change as we're doing calculations
    temp_buff = self.graph_buff[self.main_plot_size-self.fft_sample_size:]
    # Remove DC offset
    temp_buff = temp_buff - np.mean(temp_buff)

    # If there is a pipe, send temp before
    if self.pipe is not None:
        self.pipe.send(temp_buff[:])

    # FFT calculations
    ham = np.hamming(self.fft_sample_size)
    y_ham = temp_buff * ham
    self.fft_graph_mag = 4/self.fft_sample_size * \
        np.abs(np.fft.rfft(y_ham, self.fft_sample_size * self.fft_padding))

    # Set FFT data

```

```

        self.fft_graph.setData(self.fft_freq, self.fft_graph_fft_mag)

    def exit(self):
        print("exit called")
        sys.exit()

if __name__ == '__main__':
    scope = Scope('/dev/ttyUSB0')
    #scope = Scope('COM9', None)
    scope.start()

7.6.2 eegInterface.py

# eegInterface.py
#
# Creates a keyboard made of flashing checkerboxes which can
# be selected by the user looking and concentrating on an individual box
# A baseline is recorded for first 30s. The EEG data is compared against
# the baseline data to determine if user is looking at a certain box
# Author: Ronan Byrne
# Last Updated: 09/05/2018
#

from psychopy import visual, event
import numpy as np
import scipy.stats as st
import threading

# Interface arguments
window_size = [1200, 700]
checker_cycles = 4
checker_size = 160
checker_tex = np.array([[1, -1], [-1, 1]]) # One black and one white box
checker_frequency = np.array([10, 20, 15, 5, 12]) # Flashing Frequencies

special_text = 'SPECIAL'
num_text = 'NUM'
text_boxes = [
    ['A', 'B', 'C', 'D', 'E'],
    ['F', 'G', 'H', 'I', 'J'],
    ['K', 'L', 'M', 'N', 'O'],
    ['P', 'Q', 'R', 'S', 'T'],
    ['U', 'V', 'W', 'X', 'Y', 'Z', num_text, special_text]
]
num_boxes = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']

enter_text = 'ENTER'
del_text = 'DEL'
space_text = 'SPACE'
special_boxes = [space_text, '.', '?', enter_text, del_text]

class BCI(object):
    def __init__(self, win_size, freq_array, checker_size, checker_cycles, checker_tex, pipe):
        self.pipe = pipe
        self.win_size = win_size
        self.win = visual.Window(self.win_size,
                                units='pix',
                                monitor='testMonitor',
                                )
        self.mouse = event.Mouse(win=self.win)

        self.tex = checker_tex
        self.checker_size = checker_size
        self.checker_cycles = checker_cycles

        self.freq_array = freq_array

        self.stim = []
        self.selection_boxes = []
        self.xy = []
        self.top_level_boxes = text_boxes
        # Create checker boxes and text above boxes
        # Some magic numbers here for the positions, not fully customisable
        for i in range(len(self.top_level_boxes)):
            self.xy.append([i * 220 - self.win_size[0] / 2 + self.checker_size,
                           -self.win_size[1] / 2 + 0.75 * self.checker_size +
                           (not ((i + 1) % 2)) * (1.5 * self.checker_size)])
            self.stim.append(visual.GratingStim(self.win,
                                                tex=self.tex,
                                                size=self.checker_size,
                                                units='pix',
                                                sf=self.checker_cycles / self.checker_size,
                                                pos=self.xy[i]))
        self.selection_boxes.append(visual.TextStim(self.win,
                                                   text='.'.join(self.top_level_boxes[i]),

```

```

        pos=[self.stim[i].pos[0], self.stim[i].pos[1]
             + 0.60 * self.checker_size)))
    self.num_of_stim = len(self.stim)

    self.instructions_text1 = 'Recording Baseline, please stare into the center box until it
starts flashing'
    self.instructions_text2 = 'Stare into the center of the box which corresponds to the
character ' \
                           'you want to choose'
    self.instructions_box_update = False
    # TextStim is slower than TextBox but TextBox was unreliable
    self.instructions_box = visual.TextStim(self.win,
                                             text= self.instructions_text1,
                                             pos=[0, self.win_size[1] / 2 - 50],
                                             alignHoriz='center',
                                             alignVert='center')
    # White rectangle for textbox
    self.entered_background = visual.Rect(self.win,
                                           units='pix',
                                           width=self.win_size[0]/2,
                                           height=40,
                                           fillColor=[1, 1, 1],
                                           pos=[0, self.instructions_box.pos[1] - 75])
    # Textbox text
    self.entered_textbox = visual.TextStim(self.win,
                                           text='|',
                                           color=[-1, -1, -1],
                                           pos=[self.entered_background.pos[0],
                                                 self.entered_background.pos[1]],
                                           alignHoriz='center',
                                           alignVert='center')
    # Green box to outline which box was selected
    self.selected_box = visual.Rect(self.win,
                                    units='pix',
                                    width=self.checker_size,
                                    height=self.checker_size,
                                    lineWidth=40,
                                    lineColor=[-1, 1, -1],
                                    fillColor=None,
                                    pos=[0, 0] # will be changed when used
                                    )

    # Number of frames the green box is shown for
    self.selected_box_on_frames = 30
    self.selected_box_frames = 0
    self.selected_index = -1

    # Average Frame rate
    self.frame_rate = self.win.getActualFrameRate()
    self.frame_interval = 1 / self.frame_rate
    # The time interval for each box
    self.interval = 1 / self.freq_array
    # Number of frames each checker box is shown
    self.stim_frames = np.round(self.interval / self.frame_interval)
    print("Frame rate is {0}. Actual Flashing Frequency will be {1}.".format(self.frame_rate, str(
        1 / (self.stim_frames * self.frame_interval))))
    self.box_selected = False
    # Flag for if we are in the bottom level of the selection tree
    self.bottom_level = False

    self.baseline_count = 0
    self.pipe_thread = threading.Thread(target=self.pipeReceive, daemon=True)

    if self.pipe is not None:
        self.setting_baseline = True
        self.pipe_thread.start()
        self.start()
    else:
        self.setting_baseline = False

def start(self):
    self.mouse.clickReset()
    self.win.flip()
    frame_count = 0
    while not event.getKeys('escape'):

        # If 'b' pressed, recorded baseline again
        if event.getKeys('b') and self.pipe is not None:
            self.setting_baseline = True
            self.baseline_count = 0
            self.instructions_box.text = self.instructions_text1
            self.instructions_box_update = True
            self.entered_textbox.text = '|'

        # If not recording baseline, check if any boxes were selected or need to be updated
        if not self.setting_baseline:
            # Check if the left button was clicked and a box is not already selected

```

```

        if self.mouse.getPressed().count(1) and (self.selected_index == -1):
            self.mouse.clickReset()
            pos = self.mouse.getPos()
            # Check if the mouse was clicked inside one of the boxes
            for i in range(self.num_of_stim):
                if self.stim[i].contains(pos):
                    self.selected_index = i
                    self.selected_box.pos = self.stim[self.selected_index].pos
                    self.selected_box_frames = self.selected_box_on_frames
                    self.selected_box.draw()
                    break
            # A Box was selected, redraw
        elif self.selected_index is not -1:
            self.selected_box.draw()
            self.selected_box_frames -= 1
            # Last redraw of selection box, update boxes with new selection
            if self.selected_box_frames <= 0:
                self.update_selection()

        # Check if any of the checker boards need to be updated
        for i, x in enumerate(self.stim_frames):
            if (frame_count % x) == 0:
                # Swap checkerboard pattern
                self.stim[i].tex = -self.stim[i].tex

        # Update all things on screen
        self.draw_screen()
        # win.flip() blocks until the screen fresh so is used to count number of frames passed
        self.win.flip()
        frame_count += 1

    # Group choices together
    def group_choices(self, boxes):
        # Selection boxes can hold 1-4 values each
        boxes_len = len(boxes)
        if boxes_len / 2 <= self.num_of_stim:
            j_max = 2
        elif boxes_len / 3 <= self.num_of_stim:
            j_max = 3
        elif boxes_len / 4 <= self.num_of_stim:
            j_max = 4
        else:
            print("unsupported length, resetting to top level")
            self.reset_to_top_level()
            return
        offset = 0
        # Update boxes with new selections
        for i in range(0, self.num_of_stim):
            self.selection_boxes[i].text = ''
            for j in range(j_max):
                try:
                    self.selection_boxes[i].text += (boxes[i + j + offset] + ',')
                except:
                    # reach end of selections, remove ',' from last box
                    self.selection_boxes[i].text = self.selection_boxes[i].text[:-1]
                    break
            offset += 1
            # remove ',' from end of each box selection
            self.selection_boxes[i].text = self.selection_boxes[i].text[:-1]

    # Reset the selection to the top level
    def reset_to_top_level(self):
        self.bottom_level = False
        for i, x in enumerate(self.stim):
            self.selection_boxes[i].text = ','.join(self.top_level_boxes[i])
            self.selection_boxes[i].draw()

    # draw everything on screen
    def draw_screen(self):
        for i, x in enumerate(self.stim):
            self.stim[i].draw()
            self.selection_boxes[i].draw()
        if self.instructions_box_update:
            self.instructions_box.text = self.instructions_text2
            self.instructions_box_update = False
        self.instructions_box.draw()
        self.entered_background.draw()
        self.entered_textbox.draw()

    # Update the screen with new selections
    def update_selection(self):
        # Not to the lowest level selection
        if not self.bottom_level:
            # cant display all element with each box
            if len(self.selection_boxes[self.selected_index].text.split(',')) > self.num_of_stim:
                self.group_choices(self.selection_boxes[self.selected_index].text.split(','))
            # Non empty selection which can be split into max boxes or less
            elif len(self.selection_boxes[self.selected_index].text) > 1:

```

```

        temp_text = self.selection_boxes[self.selected_index].text.split(',')
        self.bottom_level = True
        # Place one choice in each box
        for i in range(self.num_of_stim):
            try:
                self.selection_boxes[i].text = temp_text[i]
            except:
                # No more text to display, rest of boxes will have no text
                self.selection_boxes[i].text = ''
            self.selection_boxes[i].draw()
        # Empty box chosen, reset to top level
        elif len(self.selection_boxes[self.selected_index].text) == 0:
            print("Empty box chosen, resetting to top level")
            self.reset_to_top_level()
        else:
            print("Unknown state, resetting")
            self.reset_to_top_level()
    # Bottom level selection
    else:
        temp_boxes = []
        # Non single character selection
        if len(self.selection_boxes[self.selected_index].text) > 1:
            # Special selection choices
            # NUM box was chosen
            if self.selection_boxes[self.selected_index].text == num_text:
                temp_boxes = num_boxes
                if len(temp_boxes) > self.num_of_stim:
                    self.group_choices(temp_boxes)
                    self.bottom_level = False
            # SPECIAL box was chosen
            elif self.selection_boxes[self.selected_index].text == special_text:
                temp_boxes = special_boxes
                if len(temp_boxes) > self.num_of_stim:
                    self.group_choices(temp_boxes)
                    self.bottom_level = False
            # ENTER box was chosen
            elif self.selection_boxes[self.selected_index].text == enter_text:
                self.bottom_level = False
                self.entered_textbox.text = '|'
                # TODO something on enter
            # DEL box was chosen
            elif self.selection_boxes[self.selected_index].text == del_text:
                self.bottom_level = False
                self.entered_textbox.text = self.entered_textbox.text[:-1]
            # SPACE was chosen
            elif self.selection_boxes[self.selected_index].text == space_text:
                self.bottom_level = False
                self.entered_textbox.text += ' '
        else:
            print('Unknown case, resetting to top level')
            self.bottom_level = False

        # Display bottom level selection
        if self.bottom_level:
            for i in range(self.num_of_stim):
                try:
                    self.selection_boxes[i].text = temp_boxes[i]
                except:
                    self.selection_boxes[i].text = ''
                self.selection_boxes[i].draw()
            else:
                # something was selected and there was no sub-level
                # reset selection to top level
                if len(temp_boxes) == 0:
                    self.reset_to_top_level()
        # Empty box chosen, reset to top level
        elif len(self.selection_boxes[self.selected_index].text) == 0:
            print("Empty box chosen, resetting to top level")
            self.reset_to_top_level()
    # Single character selection
    else:
        # Remove cursor
        if self.entered_textbox.text is '|':
            self.entered_textbox.text = self.selection_boxes[self.selected_index].text
        # Append selection
        else:
            self.entered_textbox.text += self.selection_boxes[self.selected_index].text
        self.reset_to_top_level()
        self.selected_index = -1

def pipeReceive(self):
    # TODO receive initial parameters through pipe
    fft_padding = 5          # pad fft with 5 times the length
    window_len = 1000         # fft window size
    recv_window_len = 1000    # size of data sent through pipe
    fs = 1000
    cdf_per = 10.0           # 10% probability from cumulative density function
    max_baseline_time = 30   # time to get baseline

```

```

max_baseline_count = int(max_baseline_time*fs/window_len)

ham = np.hamming(window_len)
# Frequency points from fft
freq_axis = np.fft.rfftfreq(window_len * fft_padding, 1 / fs)
signal_buff = np.zeros(window_len)

freq_array_len = len(self.freq_array)

# Signal magnitude for each fft
freq_sig_snr = np.zeros([freq_array_len, 2])
# SNR of last two fft
freq_sig_mean_snr = np.zeros([freq_array_len, 1])
# Baseline magnitude of each frequencies
freq_sig_base_val = np.zeros([freq_array_len, max_baseline_count])
# Signal threshold which is 10% or less probablitiy
freq_sig_val_thresh = np.zeros([freq_array_len, 1])
# Freq tolerance to check magnitude
freq_tol = 0
while True:
    # Read in data from pipe, pipe.recv blocks until data is received
    for i in range(int(window_len/recv_window_len)):
        signal_buff[i*recv_window_len:i*recv_window_len+recv_window_len] = self.pipe.recv()

    y_ham = signal_buff * ham
    # Calculate fft assuming signal is real, returns first half of spectrum
    rfft = np.fft.rfft(y_ham, window_len * fft_padding)
    # Calculate magnitude
    rfft_mag = 4 / window_len * np.absolute(rfft)

    # loop through each frequency calculating maximum magnitude within the freq tol
    for index, f in enumerate(self.freq_array):
        freq_start = f - freq_tol
        freq_end = f + freq_tol
        freq_max = 0
        for i in range(0, len(freq_axis)):
            if freq_axis[i] >= freq_start:
                if rfft_mag[i] > freq_max:
                    freq_max = rfft_mag[i]
            if freq_axis[i] >= freq_end:
                if self.setting_baseline:
                    # Add max value to baseline array for later
                    freq_sig_base_val[index][self.baseline_count] = freq_max
                else:
                    # Only save value is greater than threshold
                    if freq_sig_val_thresh[index] < freq_max:
                        freq_sig_snr[index][1] = freq_max / freq_sig_val_thresh[index]
                    else:
                        freq_sig_snr[index][1] = 0
                break

        if self.setting_baseline:
            self.baseline_count += 1
            # Enough baseline values
            if self.baseline_count == max_baseline_count:
                # Calculate gamma cumulative distribution function for each frequency
                for i in range(freq_array_len):
                    std = np.std(freq_sig_base_val[i])
                    mean = np.mean(freq_sig_base_val[i])
                    # create a line from the min magnitude to 1.5 * max magnitude
                    x = np.linspace(min(freq_sig_base_val[i]), max(freq_sig_base_val[i]) * 1.5,
1000)
                    # Calculate shape, scale and location of lognormal distribution
                    parameters_1 = st.lognorm.fit(freq_sig_base_val[i])
                    # Calculate lognormal cdf
                    fitted_cdf = st.lognorm.cdf(x, parameters_1[0], parameters_1[1],
parameters_1[2])
                    # Find the point on the cdf where the magnitude is less tha the cdf percent
                    threshold
                    for j in range(len(x)):
                        if (1 - fitted_cdf[j]) < (cdf_per / 100.0):
                            freq_sig_val_thresh[i] = x[j]
                            break
                    # if the cdf percent threshold is outside of range, just use 2 standard
                    deviations
                    if freq_sig_val_thresh[i] == 0:
                        freq_sig_val_thresh[i] = mean+2*std
                    print("freq {} mean {}, std {}, thresh {}".format(self.freq_array[i], mean,
std,
freq_sig_val_thresh[i]))
                    freq_sig_snr[i][0] = freq_sig_base_val[i][-1]/freq_sig_val_thresh[i]
                self.baseline_count = 0
                self.setting_baseline = False
                # Have to set a flag here instead of updating because setting the text in
                # another thread causes problems
                self.instructions_box_update = True
            else:
                print("freq sig val{}".format(freq_sig_snr.tolist()))

```

```

# Loop through frequencies to calculate mean snr
for i in range(freq_array_len):
    # If the last two fft snr are not zero, it may be from the stimulus
    if freq_sig_snr[i][0] > 0 and freq_sig_snr[i][1] > 0:
        freq_sig_mean_snr[i] = np.mean(freq_sig_snr[i])
    else:
        freq_sig_mean_snr[i] = 0
        # shift snr value back
        freq_sig_snr[i][0] = freq_sig_snr[i][1]
# Find index of max snr
max_sig_val_index = np.argmax(freq_sig_mean_snr)
# if all snr's are zero, it will return the first in the array
if freq_sig_mean_snr[max_sig_val_index] > 0:
    print("max freq snr {}".format(self.freq_array[max_sig_val_index]))
    # draw a green box around selected box
    self.selected_index = max_sig_val_index
    self.selected_box.pos = self.stim[self.selected_index].pos
    self.selected_box_frames = self.selected_box_on_frames
    self.selected_box.draw()

if __name__ == '__main__':
    bci = BCI(win_size=window_size,
              freq_array=checker_frequency, checker_cycles=checker_cycles, checker_size=checker_size,
              checker_tex=checker_tex, pipe=None)
    bci.start()

```

7.7 Appendix G – Histograms and probability density functions

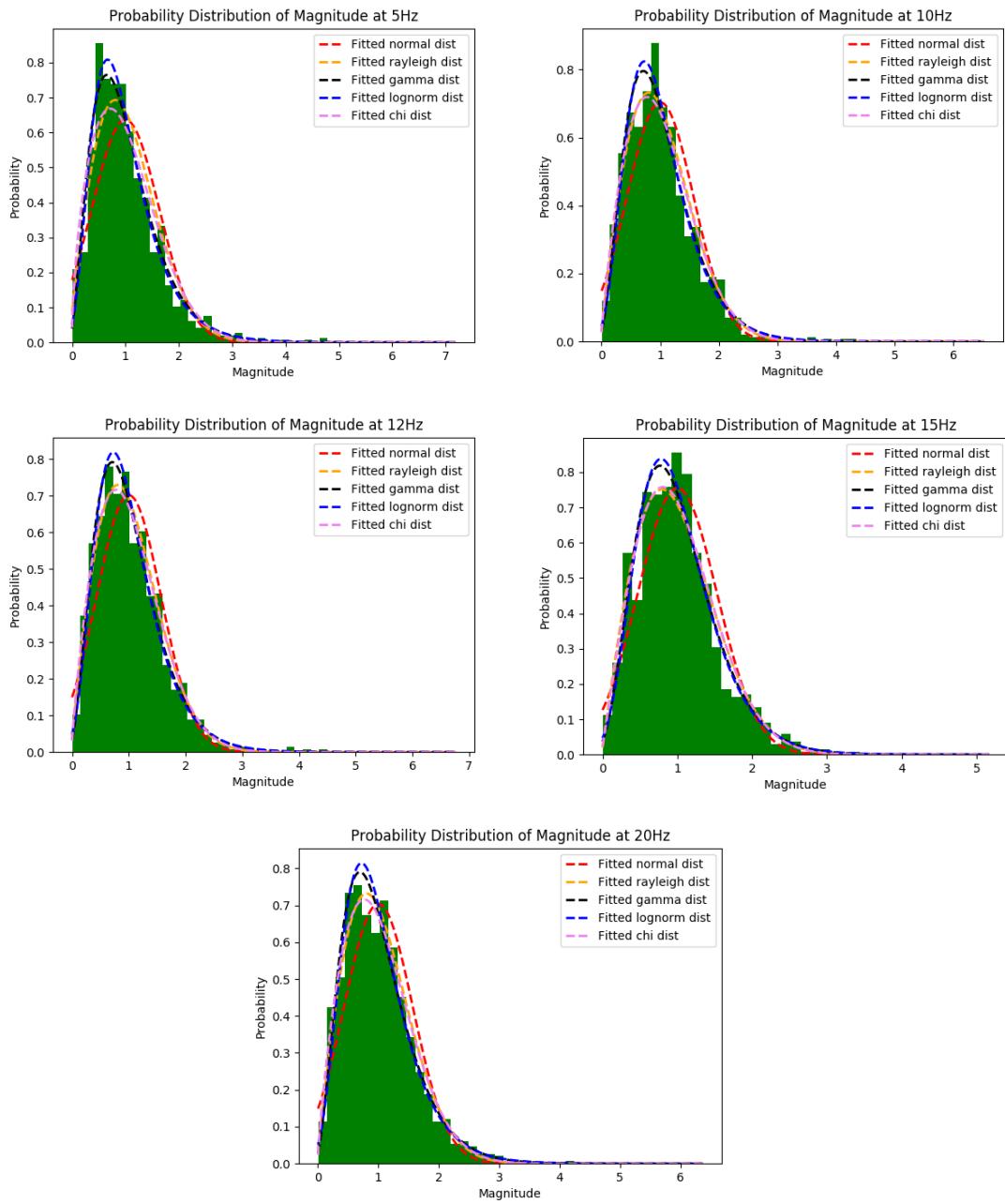


Figure 52 Histograms of normalised FFT magnitudes with no stimuli, overlaid with multiple fitted probability density functions. A sample size 1016

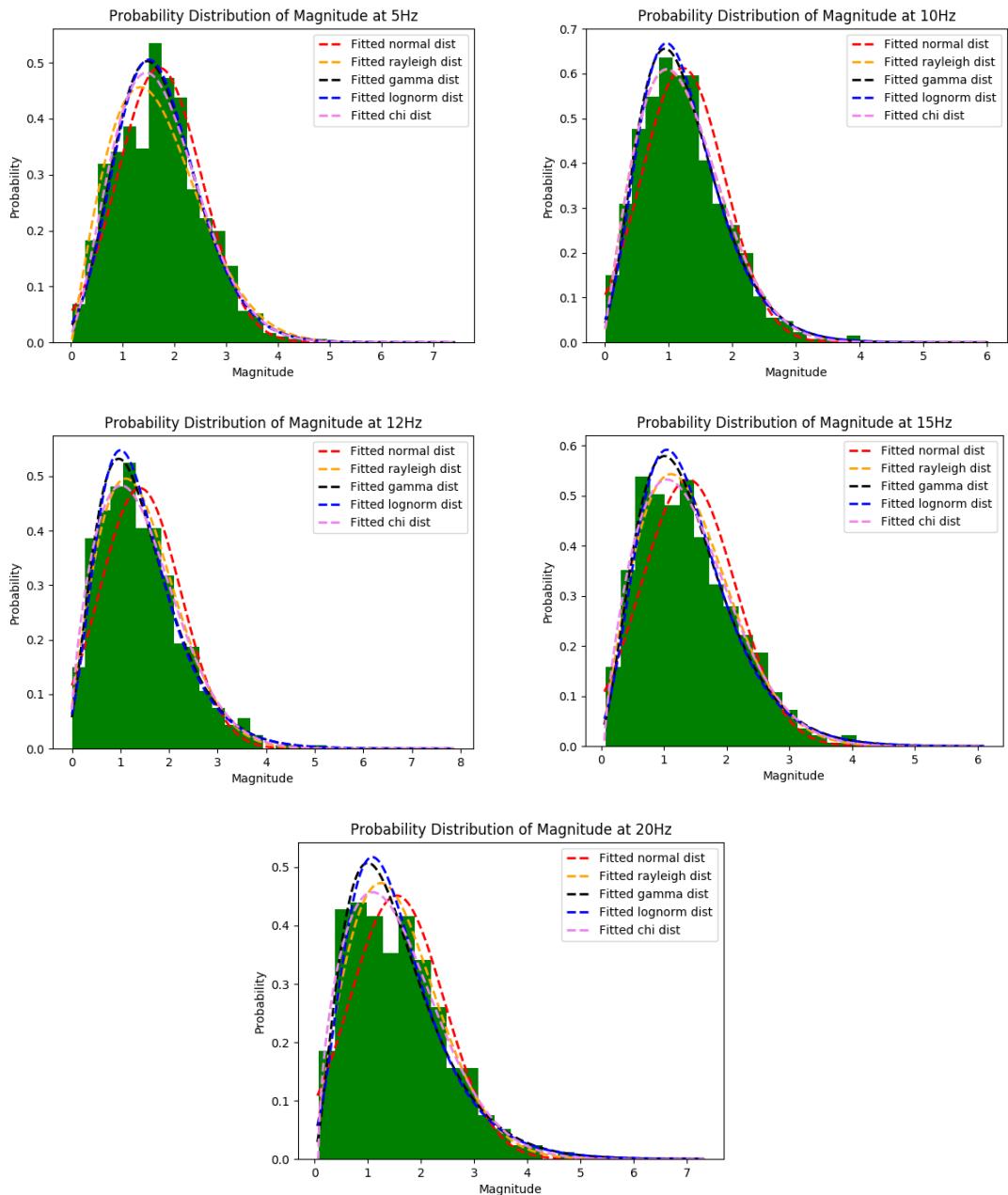


Figure 53 Histograms of normalised FFT magnitudes with stimuli, overlaid with multiple fitted probability density functions. A sample size of 761

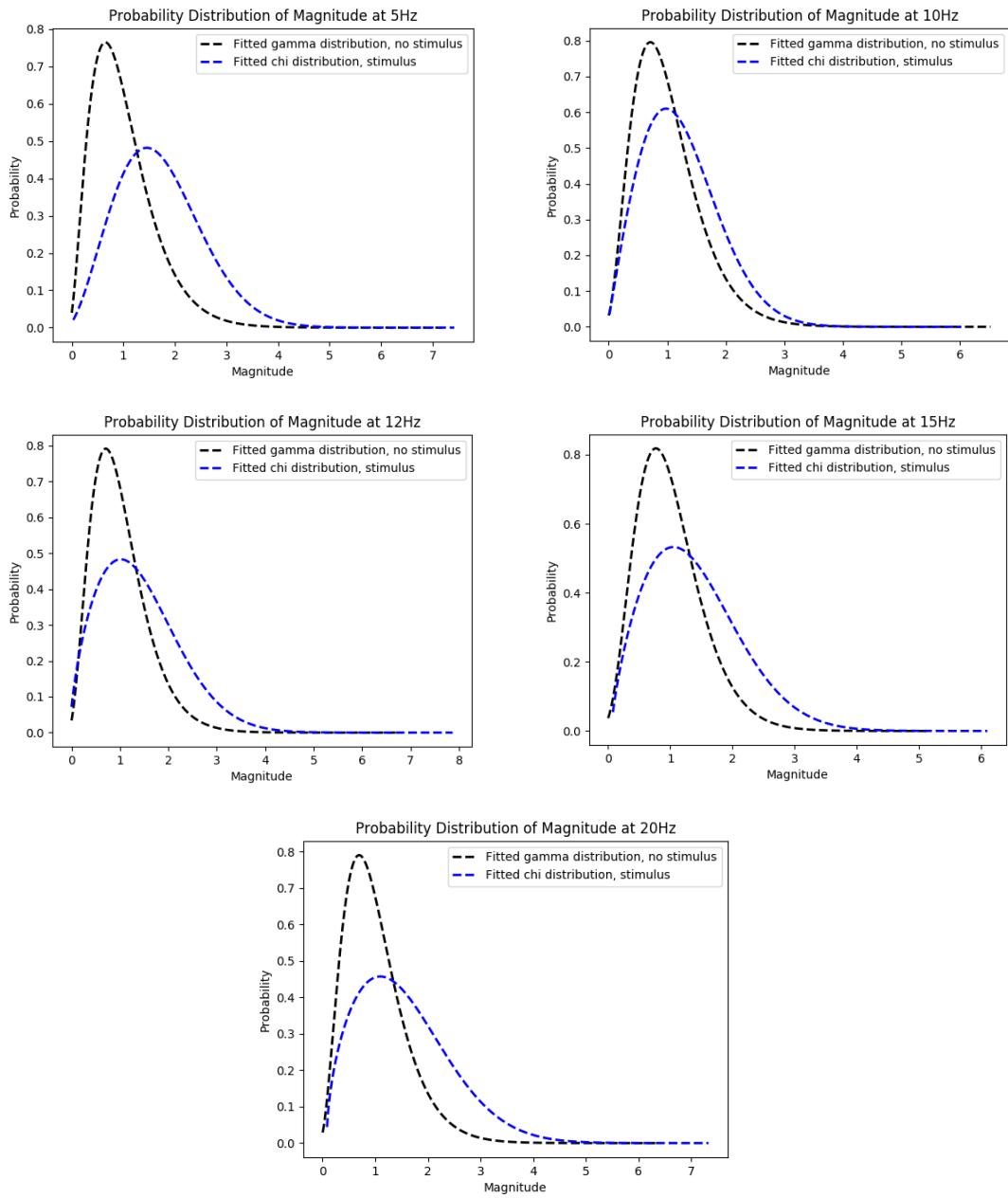


Figure 54 Normalised probability density functions of the FFT magnitudes with no stimuli (black) and with stimuli (blue)



Development of a Low Cost, Open Source, Electroencephalograph-Based Brain-Computer Interface Interim Report

by

Ronan Byrne

This Report is submitted in partial fulfilment of the requirements of the Honours Degree
in Electrical and Electronic Engineering (DT021A) of the Dublin Institute of
Technology

February 2nd, 2018

Supervisor: Dr Ted Burke

DECLARATION

I, the undersigned, declare that this report is entirely my own written work, except where otherwise accredited, and that it has not been submitted for a degree or other award to any other university or institution.

Signed: _____
Ronan Byrne

Date: _____
02/02/2018

Abstract

A brain computer interface (BCI) is a system (hardware and software) which allows the user to communicate with external devices using their brain activity, be it actively or passively controlled. BCIs have been thoroughly researched in the clinical setting but there are also some commercial BCIs on the market. But the cost of these commercial systems or even DIY systems can be quite high which restricts the number of non-clinicians or non-biomedical engineers to learn about, build and innovate in area of BCIs. In this report, the initial work plan for designing, building and testing of a low-cost, open source, electroencephalograph (EEG) based BCI system will be described. The EEG recording system will be comprised of a simple and low-cost circuit but will still have the sensitivity and functionality required to be effective in a BCI system. The final BCI system will be tested by a group of human subjects to assess its reliability, usability and comfortability.

Table of Contents

Abstract.....	3
1 Introduction	5
2 Objectives	5
3 Ethics	5
4 Research.....	7
4.1 The electroencephalograph	7
4.1.1 Biopotential electrodes	7
4.1.2 Areas of interest for BCIs in the brain.....	8
4.1.3 Types of signal that can be measured in an EEG	9
4.1.4 Noise and interference during measurement	10
4.1.5 The bio-amplifier.....	10
4.2 Brain-Computer interface	11
4.2.1 Uses of brain-computer interfaces	12
5 Work plan	12
5.1 Circuit design.....	12
5.2 Software design.....	14
5.2.1 BCI system using PC	15
5.2.2 BCI system without using PC.....	15
5.3 Testing	15
5.3.1 Initial circuit testing.....	15
5.3.2 Full system testing	15
5.4 Disseminating project outcomes	16
5.5 Tasks and durations	16
5.6 Resources and equipment	16
6 Bibliography	18
7 Appendix	20
7.1 Appendix A: Work breakdown structure	20
7.2 Appendix B: Gantt chart	21

1 Introduction

The primary objective of this project is to build a simple, but functional Electroencephalograph (EEG) based Brain-Computer Interface (BCI). This report sets out the aims of the project, summarises the research carried out to date, discussed the ethical issues posed by a project of this type and provides an initial work plan.

2 Objectives

The aim of the project as given by the brief was

“...to develop a simple EEG recording system to be used as a brain-computer interface and to explore signal processing techniques for implementing an EEG-based BCI system.”

The EEG was first recorded in 1924 [4] and has been researched thoroughly since then but has only been applied to non-medical and commercial uses, such as BCIs, in recent decades [5]. The cost of these commercially available EEG recording systems ranges from \$100-\$1000 for a “low-cost” EEG headset up to \$25,000 or more if buying a high-end system for clinical use [6]. Even the low-end cost of \$100 may pose an obstacle for some members of the maker community who are interested in recording EEG. This leads to the new aim of the project, to design and build a low-cost, open source BCI, including hardware (an EEG recording system) and software (signal processing and human-machine communication).

The specific objectives of the project are:

- Design and build a low-cost EEG recording system
- Design and implement a BCI
- Disseminate the hardware designs and source code under an open source license.

The first two objectives currently remain somewhat broad. The exact specifications of the EEG recording system and BCI software will be determined by the background research which is currently ongoing. Once a section of the brain and certain brain activity is chosen for the project, the objectives can be more refined such as the frequency range and noise level of the EEG or the specifications of the BCI. The knowledge outcomes of the project (hardware designs and source code) will be publicly disseminated under an open source license, with the aim of making EEG-based BCI more accessible to the maker community. It is hoped that the availability of a simple, low-cost means of EEG recording, together with a reference design for a functional BCI system, will stimulate wider interest in the use of EEG for BCI and lead to greater innovation in the area.

3 Ethics

The ethical issues for any engineering project should always be considered but in the case of a project that involves a physical interface between an electrical system and a user’s body, safety is of paramount importance. Fortunately, in this case, the nature of the interface is such that certain simple controls (described below) can be put in place to eliminate potential risks to the user. The device will comply with the ASTM F983 standard consumer safety specification for toy safety [7] and the directive 2014/53/EU

for making available on the market of radio equipment [8]. Similar devices such as the “The Force Trainer II: Hologram Experience” from NeuroSky have had to comply with these regulations [9].

The device in its finished state will be tested on a group of human subjects. A consent form must be signed by each subject before any testing is conducted. As well as consenting to partake in the test, each subject must also consent to their anonymised data being published online.

Because the design of the resulting system will be publicly available under an open source license, the potential for others to misuse or modify the design in a way that could cause harm or injury must be considered. Clear documentation with disclaimers will accompany the published designs. These disclaimers, in addition to the terms of the chosen license, must protect the author from any liability in the event that others using the design suffer or cause injury or loss. The choice of an appropriate license will therefore form an important element of the project work, which must be undertaken prior to dissemination.

At the time of writing, the author does not condone the use of the device being modified to control any equipment or machinery which could be considered dangerous if the equipment or machinery is to lose control.

In case others make use of the design to facilitate communication in a situation where miscommunication could result in significant negative consequences, it is important that the published designs be accompanied by clear guidelines on the reliability of the system and a realistic indication of the expected error rate. A study which was conducted by *U. Chaudhary et al* [10], tested the ability to communicate of completely locked-in syndrome(CLIS) patients using a BCI system. Once communication was deemed reliable, the four patients in the study were asked many yes or no type questions, one being if they wished to sustain their lives through artificial ventilation, all four communicated yes. If the BCI system was unreliable for this task, the doctors may have fulfilled the “perceived” request of the patient to not use the artificial ventilation and lead to the patient’s death if the BCI system interpreted the signals wrong.

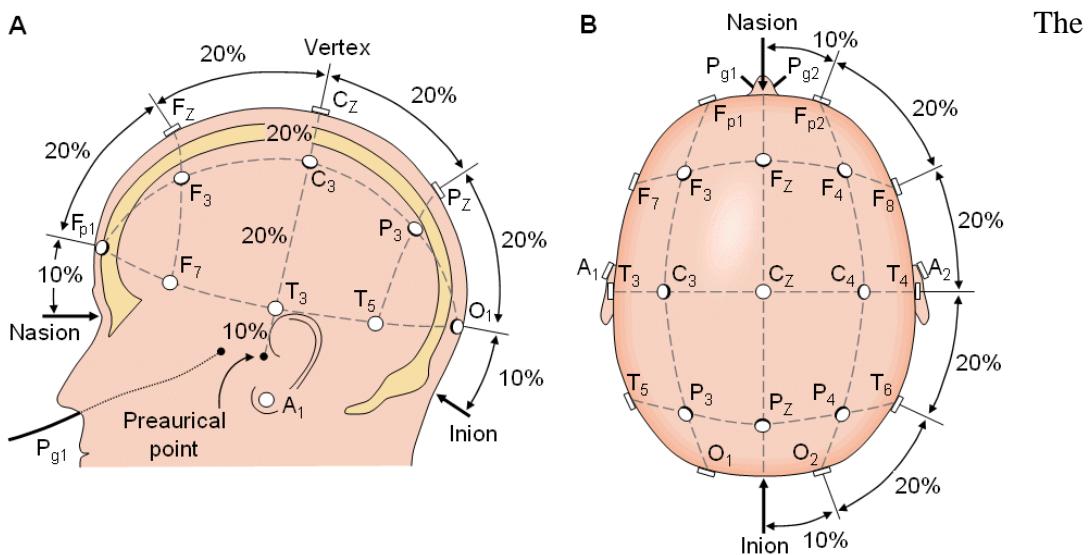
Due to the short duration of this project, it will not be possible to assess the potential effects of long-term use of the system. However, this is an issue which others should consider if making use of the resulting designs in the future or using the system on an ongoing basis. While there is no reason to expect that short-term use of the system will pose any threat to the user, long-term use of any user interface can have unexpected negative consequences (e.g. repetitive strain injury experienced by users of computer keyboards).

If this project achieves the laudable aim of facilitating communication by someone who cannot currently communicate (e.g. due to paralysis), there exists the potential for that person to subsequently engage in problematic activities (e.g. sending threatening or abusive messages online) which would otherwise not have been possible. Realistically however, since the system will not facilitate communication beyond what able-bodied people already enjoy, it is the author’s position that what the user communicates through the system is his/her own ethical responsibility, rather than that of the system designer.

4 Research

4.1 The electroencephalograph

An EEG recording system is a non-invasive measurement of electrical activity produced by the brain using electrodes attached to the scalp. It can be used for a variety of purposes such as assessing a person's alertness, monitoring their reaction to external stimuli, detecting and/or diagnosing sleeping and other neurological disorders and many more. However, in recent decades [11] the idea of combining the signals produced from the EEG recording system with external devices led to the building of BCI systems which uses the EEG signals to control a computer. BCIs will be described in more detail in a later chapter.



signals measured by an EEG recording system depend on the placement of the biopotential electrodes on the scalp as well as the current brain activity of the user. This is because each section of the brain is responsible for different functions. The most widely used and internationally recognised standard for electrode placement is the 10-20 system (Figure 1) which places the electrodes 10% and 20% relative distances from each other around the scalp. The electrodes can be placed in different configurations called montages using the 10-20 system. The combination of the 10-20 system and different montages cover most applications, but more electrodes can be added for higher resolution.

4.1.1 Biopotential electrodes

Biopotential electrodes are used to measure the electrical potentials produced by organic tissue, this could be from a muscle contraction or changes in neural activity in the brain. In an EEG, each channel uses a pair of biopotential electrodes to measure the time varying difference in electrical potential between two points on the scalp or between one point on the scalp and a reference voltage.

There are many types of biopotential electrodes, but they can be divided into two main categories: wet or dry. A wet electrode is attached to the scalp and a conductive gel

covers the electrode and hair to make a better connection to the skin, dry electrodes do not use any gel.

Wet electrodes are often less comfortable if worn for long periods of time due to the gel drying which needs to be scrubbed as it degrades signal quality, which can leave the skin more sensitive, leading to discomfort for the user. Dry electrodes can be more comfortable and require less maintenance but have been found to be more susceptible to motion artefacts and increased electrode-skin impedance [12]. Different materials are used for electrodes because of their different properties. Electrodes such as silver/silver chloride, gold, silver, stainless steel and tin are commonly used. Buying pre-made electrodes can be expensive in terms of a low-cost project. Reusable electrodes can cost \$10 or more per lead [13]. This is why homemade electrodes will be tested in this project.

Another useful classification of electrodes is active or passive. Active electrodes use a buffer amplifier within or very close to the electrode which mitigates the effect of high and/or variable electrode-skin impedance as well as reducing the effect of motion artefacts and cable movement [14]. Passive electrodes do not have this buffer amplifier and just connect straight to the bio-amplifier via a long wire.

4.1.2 Areas of interest for BCIs in the brain

As mentioned before, different parts of the brain have different functions, meaning that electrodes have to be placed in the correct locations to capture certain brain activities. Some areas will be more of more interest than others in this project, primarily because they exhibit behaviour which is readily detectable in the EEG and subject to some degree of conscious control. A diagram of the different parts of the brain can be seen in Figure 2.

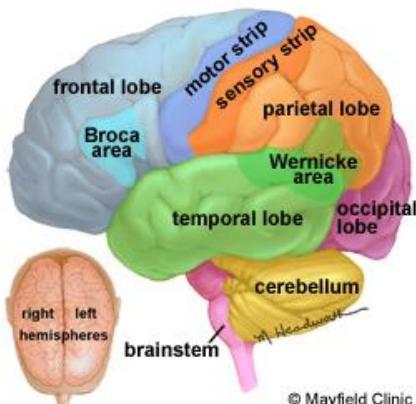


Figure 2: Different areas of the brain –
Image from the Mayfield Clinic [1]

The motor strip, seen in dark blue, is one area of interest as this controls the planning (motor imagery) and voluntary movement (real motion) of skeletal muscles and the activity in this area can be readily detected in the EEG. This can be used to control artificial limbs for people who have lost limbs or with damaged nervous systems. It could also be used to augment the actions of able-bodied people as well [15].

The occipital lobe, seen in pink, is responsible for processing visual information [16] and an electrical response can be triggered in this area by flashing a light at a constant frequency in the person's field of view [17]. The frequency of the flashing light along with its harmonics will be seen in the frequency spectrum of the electrical signal in this region. This electrical response to the flashing light is referred to as a steady-state visual evoked potential (SSVEP) [17].

The sensory strip, seen in orange, is another area of interest as it processes sensory information such as touch, the sensation of hot or cold or feeling pain [18]. This could be used for monitoring the pain levels of someone unable to communicate, such as a baby or a paralysed person [19].

The frontal lobe, seen in light blue, is responsible for a large array of functions but one thing that can be measured in terms of the EEG is mental engagement. This has been used in for market research to gauge people's reactions to products and advertisements [5].

4.1.3 Types of signal that can be measured in an EEG

SSVEPs, which were mentioned above, are an example of event-related potentials (ERPs), which are electrical responses in the brain that are produced as a direct response to external stimuli. However, ERPs are not, in general, steady state signals. Transient visual evoked potentials (TVEPs), for example, occur when the frequency of the stimulus is less than 6Hz. SSVEPs occur in response to stimuli with higher frequencies than this [20]. TVEPs are not usually used for BCIs but SSVEP-based BCIs are common and can be used to locate a person's gaze on a screen if there are flashing stimuli with different frequencies located at different parts of the screen [20].

Slow Cortical Potential (SCPs) are slow voltage shifts with a very low frequency, usually below 1Hz, and is associated with an increase or decrease of neuronal activity [20]. People can be trained to generate varying SCP changes, this has been used to help people with amyotrophic lateral sclerosis (ALS) with total motor paralysis to communicate, but took many sessions to train [20, 21].

The P300 wave is a positive voltage deflection which occurs in response to an unusual or unexpected stimulus, be it auditory, visual or somatosensory [20, 22]. Due to the P300 wave being caused by "random" stimuli and being a transient time domain occurrence rather than a constant frequency domain occurrence, it has been proven unreliable in BCI systems [20].

Sensorimotor rhythms (SMR) or mu (7-13 Hz) and beta (13-30 Hz) rhythms are oscillations located in the sensorimotor cortex within the motor strip which correspond to motor tasks or motor planning [20]. Increased SMR indicates a person is more relaxed and not moving; decreased SMR means the person is moving or planning to move [20, 23]. This type of BCI system has been well researched and has been successfully used to facilitate communicating by people with motor impairment or paralysis. This type of system has also been used for non-medical applications, such as playing computer games [22].

Rhythm	Signal appearance	Main behavioral trait
Gamma 30-100 Hz		Represents binding of different populations of neurons for the purpose of carrying out a certain
Beta 13-30 Hz		Usual waking rhythm associated with active thinking and active
Alpha 8-13 Hz		It is usually found over the occipital regions. Indicates relaxed awareness without attention or
Theta 4-8 Hz		Theta waves appear as consciousness slips towards drowsiness. Theta increases have
Delta 1-4 Hz		Primarily associated with deep (slow) wave sleep.

Figure 3: EEG frequency bands and associated mental states – Image from M.G.N.Garcia et al [3]

All of the above signals except the P300 wave and TVEP are frequency domain phenomena, but aspects of the current mental state of the brain can be assessed by looking at certain frequency bands, each band being associated with a given mental state. These bands, together with the associated mental state can be seen in Figure 3 [3].

4.1.4 Noise and interference during measurement

A few of the following topics have been briefly mentioned above but will be explained in more detail in this section.

Electrode-skin impedance is the impedance between the electrode and the skin, if it is too high, can cause distortion and attenuation of the already small EEG signals. It can also decrease the common mode rejection ratio (CMRR) of the instrumentation amplifier if there is a mismatch between the electrode-skin impedances of the positive and negative electrodes [24].

Motion artefacts are electrical interference caused by the movement of the electrode or electrical signals produced due to the user's movement such as eye blinks or teeth clenching. The magnitude of the electrical interference can be many times that of the EEG, sometimes completely obscuring the signals of interest.

50 Hz (or 60 Hz) interference is ordinarily present in EEG recordings, due to the subject being exposed to electromagnetic interference from nearby main electrical lines. The magnitude of the noise can be large relative to the EEG signal

Electrode half-cell potentials act as a series DC voltage on the electrode due to the material and temperature of the electrode. This voltage can be 100s of millivolts [25], multiple times larger than the EEG signal and if amplified, could saturate the instrumentation amplifiers output.

4.1.5 The bio-amplifier

4.1.5.1 Pre-Amplifier

In this context, the term pre-amplifier refers to active electrodes, which have already been discussed in a previous section but will be briefly revisited here. A buffer amplifier is connected within or very close to the electrode to anchor the signal so that it is vulnerable to interference, whether it be from motions artefacts or environmental noise. It also helps mitigate the effect of the electrode-skin impedance. This does mean more wires have to be connected to each electrode, increasing its size and cost.

4.1.5.2 Protection circuit

The protection circuit is where the EEG signal first enters the main circuit, it is used to not only protect the user but also the instrumentation amplifier. It is used to limit the voltage and current on the electrode lines. It can be done through a diode or transistor bridge with current limiting resistors.

4.1.5.3 Instrumentation amplifier

The instrumentation amplifier (in-amp) is a circuit comprising of multiple operational amplifiers (op-amps), which takes in two signals and amplifies the difference, depending on the gain, using a differential and non-inverting op-amp configuration. This can be done using three op-amps or using a specially designed instrumentation amplifier IC. The in-amp has a differential voltage input and a single-ended voltage output. It requires high gain to sense the tiny voltage variations measured

during EEG measurement and must have high CMRR in order to reject common mode interference present at the differential voltage input.

4.1.5.4 Common mode rejection

Although in-amps are designed to have high CMRR, the circuits overall CMRR can be further improved by adding a driven right leg (DRL) circuit. The DRL circuit senses the CM voltage at the electrodes, amplifies and inverts it, and feeds it back into the body, resulting in a very significant reduction in CM interference [26].

4.1.5.5 High-pass filter

A high-pass filter can be incorporated to attenuate unwanted low frequencies such DC offsets from half-cell potentials, instrumentation offset errors or non-DC signals like lower frequency EEG bands which are not of interest.

4.1.5.6 Low-pass filter

As with the high-pass filter, it can attenuate unwanted signals such as higher frequency components such as the higher EEG bands and interference such as 50 Hz – 60 Hz if not measuring to this range. If a low-pass filter is not used, these higher frequencies may lead to aliasing in the digital data if the sample rate is too low.

4.1.5.7 Notch filter

If frequencies around or above 50 Hz – 60 Hz are to be measured but not these frequencies themselves, a notch filter can be used to attenuate this select frequencies.

4.1.5.8 Isolation

BCI systems often use computers to process the EEG signal and react to it. This could pose a risk if the computer was connected to the mains supply. If there is a fault in the computer, there could be a path from mains to the user which could be fatal. So to eliminate this risk, the circuit should be electrically isolated from mains voltage. This is often done through optocouplers or wireless communication.

4.2 Brain-Computer interface

A brain-computer interface, as mentioned above, uses signals from the brain to control an external device. This technology is a lot younger than the EEG and was first designed to be used in assistive devices for disabled people but in recent years has been seeing uses in games and other leisurely activities for able-bodied people [23].

Many BCIs do not use EEG. There is also electrocorticography (ECOG), functional magnetic resonance imaging (fMRI), functional near-infrared spectroscopy (fNIRS) and magnetoencephalography (MEG) [5]. ECOG is an invasive measurement of the electrical activity of the brain by placing electrodes directly on its surface. fMRI which measure the blood flow around the brain using magnetic resonance. fNIRS does the same as fMRI but uses light rather than magnetic resonances to measure the blood flow around the brain. And lastly, MEG measures the magnetic fields generated by the changing currents passing around the brain [20].

4.2.1 Uses of brain-computer interfaces

First of all, a system which only displays the user's EEG does not fall under the umbrella of BCI as it does not react to the signals or change the user's environment [27]. However, if the BCI system allows the user to communicate by means of actively varying the signals measured by an EEG recording system, this would count as a BCI as it is reacting to the EEG signals. BCI systems have also been used to diagnose if a patient had locked-in syndrome (LIS) [20], a person who is aware of their surroundings but physically paralysed, or was in a vegetative state, a person who is unaware of their surroundings [28]. After the diagnosis of LIS, the patient could be trained to communicate using the BCI [20].

BCI have often been used to control the user's environment or an object such as a prosthetic hand [23]. This has been used for patients who are partially paralysed or have lost limbs to make their lives a bit easier. Another external device which has been successfully integrated into a BCI system is a wheelchair.

The EEG recording systems and BCIs have left the clinical setting and have started to be used by the general public as human-machine interfaces and games controllers [5, 23]. BCI systems have been used as a replacement for the traditional mouse and keyboard or other controllers in video games. Non-video games such as "MindFlex Duel" which employs the user's concentration levels to move a ball through an obstacle course [29].

BCIs have also seen use in security. For example, it was used for authentication of a user logging in. The mental state of the user had to match the saved mental state of the user who created the password to log in, making it very difficult for someone other than the creator of the password to sign in [30].

5 Work plan

In this chapter, an initial work plan for the project will be discussed with estimations of task durations and the types of tasks to be completed.

As previously mentioned, the exact specifications of the EEG recording system and BCI software are to be determined and a design of the BCI software has not been fully decided on. For example, if the BCI will incorporate a PC or solely be on the microcontroller. If the BCI software is fully implemented on the microcontroller, a basic PC application will still be made to visualise, record and analyse the EEG signal. So a work plan describing how the microcontroller and PC software will be discussed.

5.1 Circuit design

Below in Figure 4 is the initial design of the Bio-Amplifier circuit to be tested. The circuit was designed in OrCAD Capture.

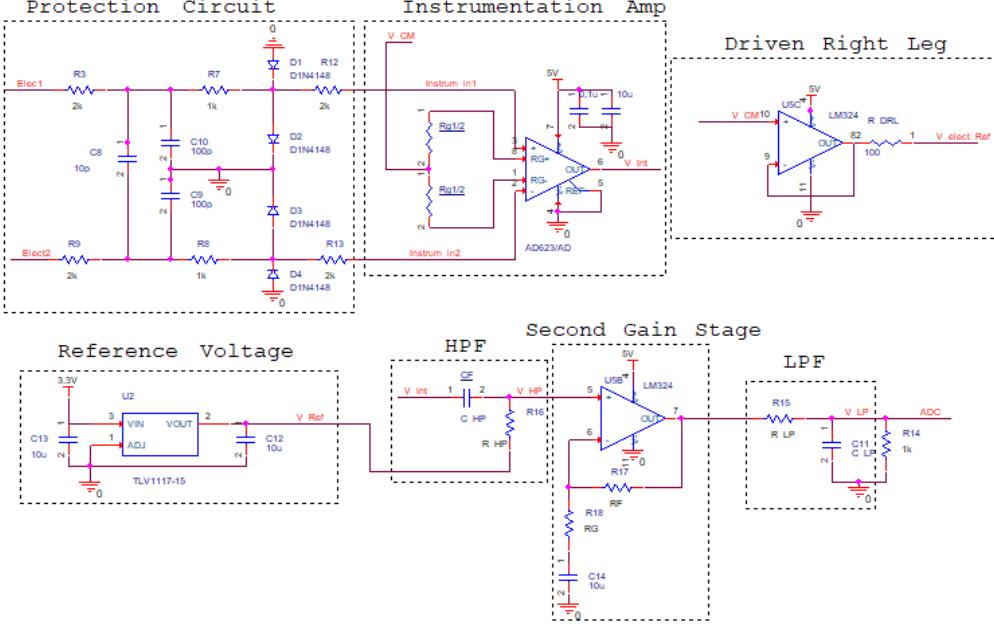


Figure 4: Initial bio-amplifier circuit design which is mainly comprised of a protection circuit, instrumentation amplifier, driven right leg circuit, high and low-pass filters, a variable gain stage and a reference voltage circuit.

The signals from the electrodes, V_elect+ and V_elect-, are fed into the protection circuit which clamps the voltage if they get too large and limit the current flow. The diodes (1N4148) start conducting around 0.5 V from the datasheet and OrCAD simulation, this means the voltage limits are ± 0.5 V. Within this voltage range, the signals are unaffected. This protection circuit was taken from *B. Luan et al [31]*.

After the protection circuit, the signals are fed into the instrumentation amp (AD623) and the difference between the two is amplified by the gain given by the equation below.

$$A_i = 1 + \frac{100k}{R_g} \quad (1)$$

The two capacitors connected to V+ are taken from the datasheet for a single-rail circuit. To increase the common mode rejection ratio, a driven right leg circuit is connected between the two gain resistors, this is also taken from the AD623 datasheet. The output of the DRL circuit is connected to a third electrode attached to the user at a location separate from the scalp.

A reference voltage of 1.5 V is created using the TLV1117-15. This is used to create a DC offset so that the microcontroller ADC with a range of 0-3.3 V can read in the full signal. The output from the instrumentation amp is high-pass filtered to attenuate any unwanted DC offsets but this is where the reference voltage is inserted. The high-passed signal is passed through an AC coupled non-inverting amp to amplify the small AC signal without changing the DC offset, the gain for this amp will be adjustable through a potentiometer. The output of the amp is finally low-pass filtered to attenuate the higher unwanted frequencies. The equations for the cut-off frequency of the filters and the op-amp gain are shown below.

$$f_{cut} = \frac{1}{2 * \pi * R * C} \quad (2)$$

$$A_{non} = 1 + \frac{R_f}{R_G} \quad (3)$$

$$A_T = A_i * A_{non} = \left(1 + \frac{100k}{R_g}\right) \left(1 + \frac{R_f}{R_G}\right) \quad (4)$$

The equation for the high-pass and low-pass filters are same. The cut-off frequencies and gains are still to be determined. The low-pass filter may be influenced by the internal capacitor in the ADC and this will be investigated. A pulldown resistor is added to the ADC pin to stop it from floating.

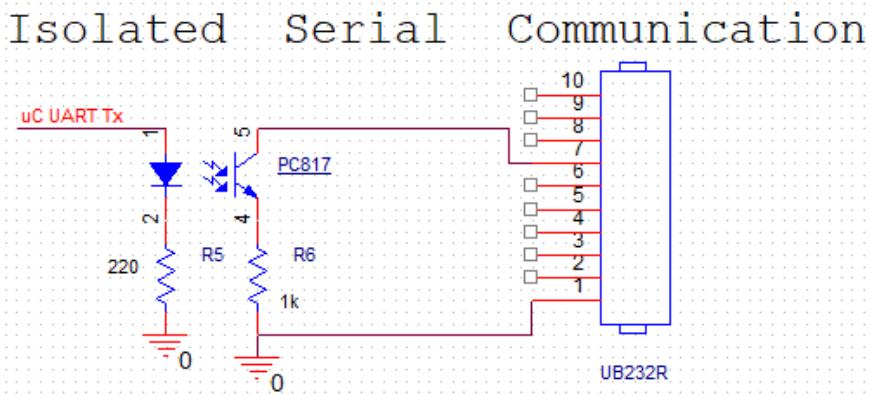


Figure 5: Isolated serial communication circuit between microcontroller and PC

It is necessary to isolate the user from mains voltage, which can be done through optoisolation using the PC817. If the microcontroller was directly connected to a computer which was powered by mains, there is a danger that the user might inadvertently be exposed to mains voltage in the event of a fault. Under normal working conditions, the user would be safe but the circuit is electrically isolated to mitigate this risk. The microcontroller will use serial communication through the PC817 and serial-to-USB IC (UB232R) to send the EEG data to the PC.

Depending on the outcome of investigation into their construction of electrodes, the electrodes used may be either active or passive. In either case, it is envisaged that they will be made from cheap, easy to find materials such as hair clips to keep the cost low. Electrodes which are easy and quick to set up will be investigated as well due to the fact that most EEG headsets which allow the changing of electrode placement (most commercial headsets do not allow for this) do not have these properties.

The circuit which is directly connected to the user will be battery powered.

5.2 Software design

The initial design for the software will be detailed in this chapter. Two work plans will be described, one using the microcontroller on its own as the BCI system and second one using both the microcontroller and the PC as the BCI system. After further analysis of both options, one of the described BCI systems will be built. In both case, the software is planned to work asynchronously.

5.2.1 BCI system using PC

If this option is selected, the code for the microcontroller will be simple as all it will do is read in the ADC data and send it over serial to the PC, it is unlikely any digital signal processing (DSP) will be required here.

The PC will receive and process the serial data, looking for the features of interest. The BCI software window may include different display windows, such as the control window and the EEG plotting window. The data will be saved to a CSV file during recording and have a folder system. The software will use DSP methods such as the Fast Fourier Transform (FFT) to analyze the frequency content of the signal.

5.2.2 BCI system without using PC

If this option is selected, the microcontroller will record and process the ADC data, for example by performing an FFT and feature extraction or other DSP which could be significantly slower than performing the same operations on the PC.

Even though the PC is not being used in the BCI system, a PC application will be developed to visualise the data, comparing different feature extraction techniques and save the data. This may be used for calibration when setting up the device but will also be used for testing and troubleshooting.

5.3 Testing

5.3.1 Initial circuit testing

The first part of the circuit that will be tested is the protection circuit. This will be done by passing in a sine wave with a voltage above the saturation voltage of the diode. If the circuit works correctly, the output voltage of the circuit will be clamped at the saturation voltage.

To measure the CMRR of the instrumentation amp, a sine wave with noise will be connected to both of the electrode inputs and the output of the amp will be measured, if the output is close 0 V, the CMRR is high.

After the whole circuit is proven to work by passing in a small signal into the instrumentation amp and seeing it amplified at the output of the low-pass filter, the circuit will be tested as an ECG recording system. The ECG testing will be done to see if the circuit works as it will be easier to setup and validate than an EEG test. It will be easier to validate as the voltage range for an ECG is 0.1 mV – 5 mV whereas EEG is 0.5 uV – 100 uV [32, 33]. The driven right leg circuit will also be tested during the ECG to see how it affects the common mode rejection of the circuit.

If the ECG test is successful, an EEG test using SSVEPs will be conducted as this ERP is relatively easy to elicit and monitor. If both tests are successful, integration of the EEG recording circuit and BCI will begin.

5.3.2 Full system testing

As the full system design is not fully realised, the plan for user testing cannot yet be fully described. However, a group with no prior experience of BCI systems will use the

system to perform a task or set of tasks. Depending on the final design of the system, the subject may need time to be trained with the device before any results are taken.

When the subject is trained and feels comfortable using the system, testing will begin and the number of times the system acts in accordance with or against the user's intention will be recorded to get an estimate of the error rate and reliability. The task completion pass/fail criteria is planned to be objective so that the ease of use of the system can be estimated.

Once testing is finished, the subject can then rate how comfortable the system was to wear as well as how difficult it was to use.

5.4 Disseminating project outcomes

The circuit and software for the BCI system will be disseminated online through GitHub, under an appropriate license and with documentation detailing how to set up the system along with videos of it working.

5.5 Tasks and durations

The work breakdown structure (WBS) showing the main tasks along with sub-tasks is shown in Figure 6. Under the "Code" topic shows the tasks which could be included in either the microcontroller only BCI system or the microcontroller and PC BCI system.

Estimates of the duration of the main tasks were calculated and formatted in a high-level Gantt chart seen from Figure 7 to Figure 12. This will be used throughout the project to assess progress.

Weekly meetings have been set up with the project supervisor to assess progress week by week.

5.6 Resources and equipment

Below is a table of resources and equipment that may be used during the project, this will most likely differ from the final table as more suitable components may be found during the course of the project. The quantity and cost of the resources are also shown with the actual amount paid for each resource shown in most cases. If the amount paid could not be found, the cheapest price found online was used. This includes buying in bulk from online retailers.

Table 1: Table of resources and equipment

Resource/Equipment	Quantity	Cost(€)	Notes
Circuit		€29.56	
Assorted Resistors	16	€0.00	<i>Sizes to be assessed</i>
Assorted Capacitors	10	€0.00	<i>Sizes to be assessed</i>
Diode	4	€0.46	<i>IN4148</i>
Instrumentation Amplifiers	1	€1.12	<i>AD623</i>
Operational Amplifiers	1	€0.03	<i>LM358</i>
Wires	-	€0.00	
Electrodes	3	-	<i>Various Homemade Electrodes to be built and compared</i>
Voltage Regulators	1	€0.34	<i>TLV1117-15</i>
Battery	1	€1.45	9V
Microcontroller	1	€9.30	<i>NUCLEO-F303K8</i>
Optocoupler	1	€0.02	<i>PC817</i>
USB to Serial	1	€14.90	<i>UB232R*</i>
Solder	-	€0.00	
Breadboard	1	€1.94	
Software			
Microcontroller	-	-	<i>SW4STM32 IDE Using HAL Library</i>
PC	-	-	<i>PyCharm IDE Using Python 3</i>
Equipment			
Soldering Iron	-	-	
Hot Glue Gun	-	-	

* A cheaper IC will be investigated

6 Bibliography

- [1] M. Clinic. (2016, January 16). *Brain Anatomy, Anatomy of the Human Brain*. Available: <http://www.mayfieldclinic.com/PE-AnatBrain.htm>
- [2] J. Malmivuo and R. Plonsey, *Bioelectromagnetism - Principles and Applications of Bioelectric and Biomagnetic Fields*. 1995.
- [3] M. G. N. Garcia, T. K. TSONEVA, D. BALDO, D. Zhu, P. H. A. GABRIELSSON, and K. P. N.V., "Device and method for cognitive enhancement of a user," 2014. Available: <https://www.google.com/patents/EP2681729A1>.
- [4] L. Haas, "Hans Berger (1873–1941), Richard Caton (1842–1926), and electroencephalography," (in eng), *Journal of Neurology, Neurosurgery, and Psychiatry*, vol. 74, no. 1, p. 9, Jan 2003.
- [5] S. N. Abdulkader, A. Atia, and M.-S. M. Mostafa, "Brain computer interfacing: Applications and challenges," *Egyptian Informatics Journal*, vol. 16, no. 2, pp. 213-230, 2015/07/01/ 2015.
- [6] imotions. (2017). *EEG Headset Prices – An Overview of 15+ EEG Devices*. Available: <https://imotions.com/blog/eeg-headset-prices/>
- [7] Standard Consumer Safety Specification for Toy Safety, 2018.
- [8] DIRECTIVE 2014/53/EU OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 16 April 2014 on the harmonisation of the laws of the Member States relating to the making available on the market of radio equipment and repealing Directive 1999/5/EC, 2018.
- [9] NeuroSky, "The Force Trainer II: Hologram Experience," ed, p. 2.
- [10] U. X. Chaudhary, B. Silvoni, S. Cohen, LG. Birbaumer, N., "Brain–Computer Interface-Based Communication in the Completely Locked-In State," *PLoS Biol*, vol. 15, no. 1, 2018.
- [11] J. J. Vidal, "Toward Direct Brain-Computer Communication," *Annual Review of Biophysics and Bioengineering*, vol. 2, no. 1, pp. 157-180, 1973.
- [12] J. Saab, B. Battes, and M. Grosse-Wentrup, "Simultaneous EEG Recordings with Dry and Wet Electrodes in Motor-Imagery," presented at the 12th Conference of Junior Neuroscientists of Tübingen, Heiligkreuztal, Germany., 2011/10, 2011. Available: <http://mlin.kyb.tuebingen.mpg.de/BCI2011JS.pdf>
- [13] Bio-Medical. (2018). *Electrodes for EMG/EEG/ECG/EKG*. Available: <https://bio-medical.com/supplies/electrodes.html>
- [14] g. t. M. Engineering. (2018). *Comparision: active versus passive electrodes*. Available: <http://www.gtec.at/News-Events/Newsletter/Newsletter-September-2010-Volume-29/articles/Comparision-active-versus-passive-electrodes2>
- [15] Y. Wang and T. P. Jung, "A Collaborative Brain-Computer Interface for Improving Human Performance," (in eng), *PLoS One*, vol. 6, no. 5, 2011.
- [16] P. A. Abhang, B. W. Gawali, and S. C. Mehrotra, "Chapter 1 - Introduction to Emotion, Electroencephalography, and Speech Processing," in *Introduction to EEG- and Speech-Based Emotion Recognition*, P. A. Abhang, B. W. Gawali, and S. C. Mehrotra, Eds.: Academic Press, 2016, pp. 1-17.
- [17] A. M. Norcia, L. G. Appelbaum, J. M. Ales, B. R. Cottreau, and B. Rossion, "The steady-state visual evoked potential in vision research: A review," (in eng), *Journal of Vision*, vol. 15, no. 6, 2015.
- [18] P. J. Tuite and J. Konczak, "Cortical Sensory Dysfunction and the Parietal Lobe," *Encyclopedia of Movement Disorders*, pp. 254-257Accessed on: 2010/01/01/. doi: <https://doi.org/10.1016/B978-0-12-374105-9.00170-2> Available: <http://www.sciencedirect.com/science/article/pii/B9780123741059001702>
- [19] C. Hartley *et al.*, "Nociceptive brain activity as a measure of analgesic efficacy in infants," *Science Translational Medicine*, vol. 9, no. 388, 2017.
- [20] L. F. Nicolas-Alonso and J. Gomez-Gil, "Brain Computer Interfaces, a Review," (in eng), *Sensors (Basel)*, vol. 12, no. 2, pp. 1211-79, 2012.

- [21] A. Kubler *et al.*, "The thought translation device: a neurophysiological approach to communication in total motor paralysis," (in eng), *Experimental Brain Research*, vol. 124, no. 2, pp. 223-32, Jan 1999.
- [22] T. W. Picton, "The P300 wave of the human event-related potential," (in eng), *Journal of Clinical Neurophysiology*, vol. 9, no. 4, pp. 456-79, Oct 1992.
- [23] H. Yuan and B. He, "Brain-Computer Interfaces Using Sensorimotor Rhythms: Current State and Future Perspectives," (in eng), *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 5, pp. 1425-35, May 2014.
- [24] M. Leonardo Casal and Guillermo La, "Skin-electrode impedance measurement during ECG acquisition: method's validation," *Journal of Physics: Conference Series*, vol. 705, no. 1, p. 012006, 2016.
- [25] A. B. Usakli, "Improvement of EEG Signal Acquisition: An Electrical Aspect for State of the Art of Front End," *Computational Intelligence and Neuroscience*, vol. 2010, 2010.
- [26] A. Wong, K.-P. Pun, Y.-T. Zhang, and C.-S. Choy, *An ECG measurement IC using driven-right-leg circuit*. 2006, pp. 4 pp.-348.
- [27] J. J. Shih, D. J. Krusinski, and J. R. Wolpaw, "Brain-Computer Interfaces in Medicine," (in eng), *Mayo Clinic Proceedings*, vol. 87, no. 3, pp. 268-79, 2012.
- [28] S. B. Laureys, J. Goldman, J., "Yearbook of Intensive Care and Emergency Medicine 2001," in *Cerebral Function in Coma, Vegetative State, Minimally Conscious State, Locked-in Syndrom*: SpringerLink, 2018, pp. 386-396.
- [29] NeuroSky. (2018). *Mindflex Duel*. Available: <https://store.neurosky.com/products/mindflex-duel>
- [30] T. v. Kišasondi, Ivan., "Two factor authentication using EEG augmented passwords - IEEE Conference Publication," presented at the Proceedings of the ITI 2012 34th International Conference on Information Technology Interfaces, Cavtat, Croatia, of Conference: 25-28 June 2012 Date Added to IEEE Xplore: 20 September 2012 ISBN Information: Electronic ISBN: 978-953-7138-26-4 Print ISBN: 978-1-4673-1629-3 CD-ROM ISBN: 978-953-7138-25-7 ISSN Information: Print ISSN: 1334-2762 INSPEC Accession Number: 13000982 DOI: 10.2498/iti.2012.0441 Publisher: IEEE Conference Location: Cavtat, Croatia. Available: <http://ieeexplore.ieee.org/document/6308035/>
- [31] B. Luan, M. Sun, and W. Jia, "Portable Amplifier Design for a Novel EEG Monitor in Point-of-Care Applications," *Proceedings of the IEEE ... annual Northeast Bioengineering Conference. IEEE Northeast Bioengineering Conference*, vol. 2012, pp. 388-389, 2012.
- [32] L. Zhang, X. j. Guo, X. p. Wu, and B. y. Zhou, "Low-cost circuit design of EEG signal acquisition for the brain-computer interface system," in *2013 6th International Conference on Biomedical Engineering and Informatics*, 2013, pp. 245-250.
- [33] A. Yu. (2018). *Electrocardiogram Amplifier Design Using Basic Electronic Parts*. Available: <http://www.realworldengineering.org/index.php?page=project&project=591>

7 Appendix

7.1 Appendix A: Work breakdown structure

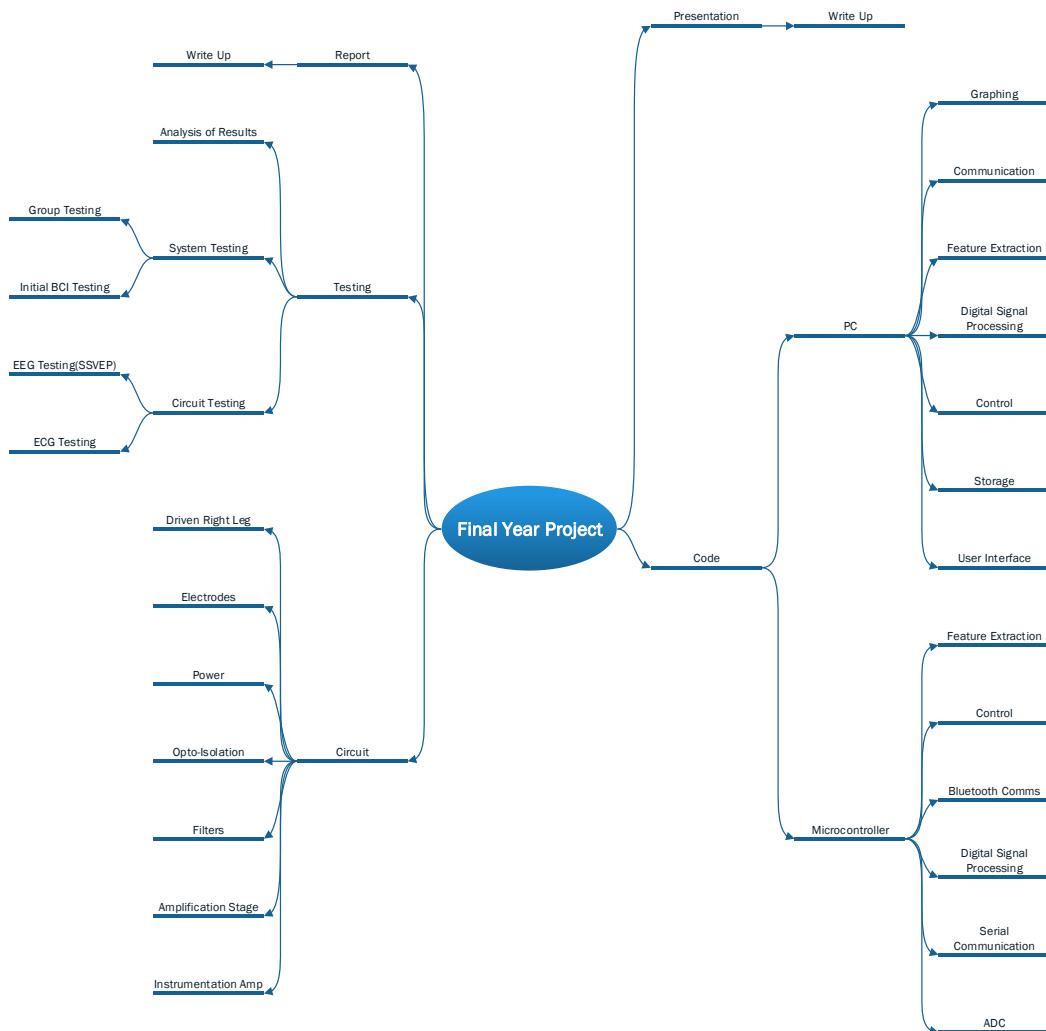


Figure 6: Work Breakdown Structure

7.2 Appendix B: Gantt chart

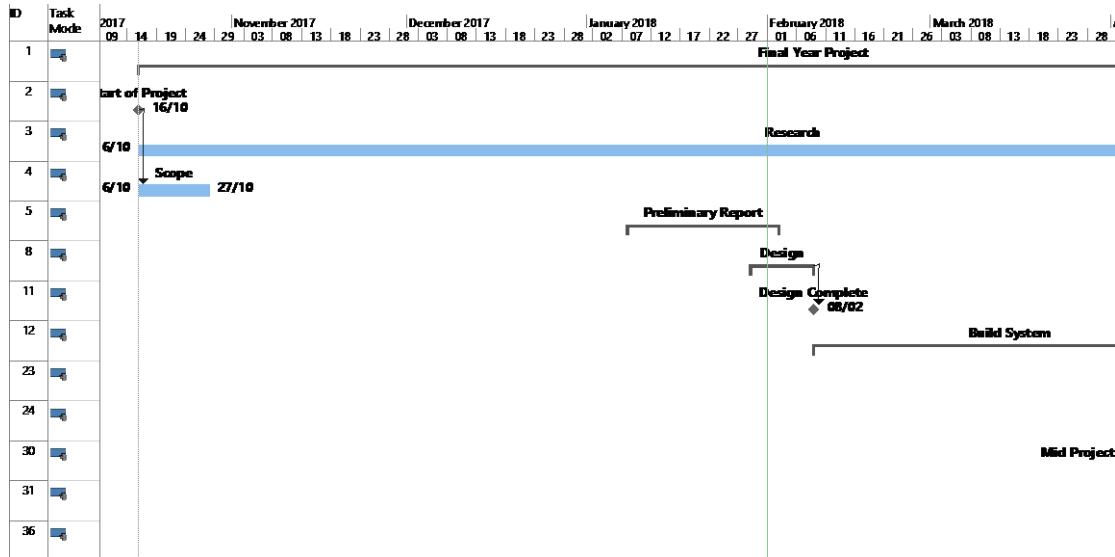


Figure 7: Gantt Chart Overview 1

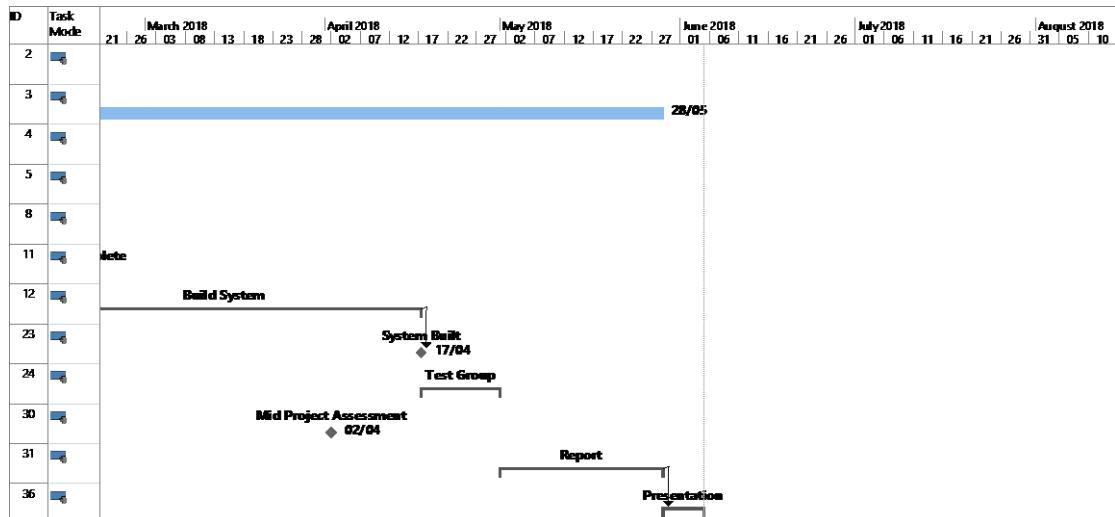


Figure 8: Gantt Chart Overview 2

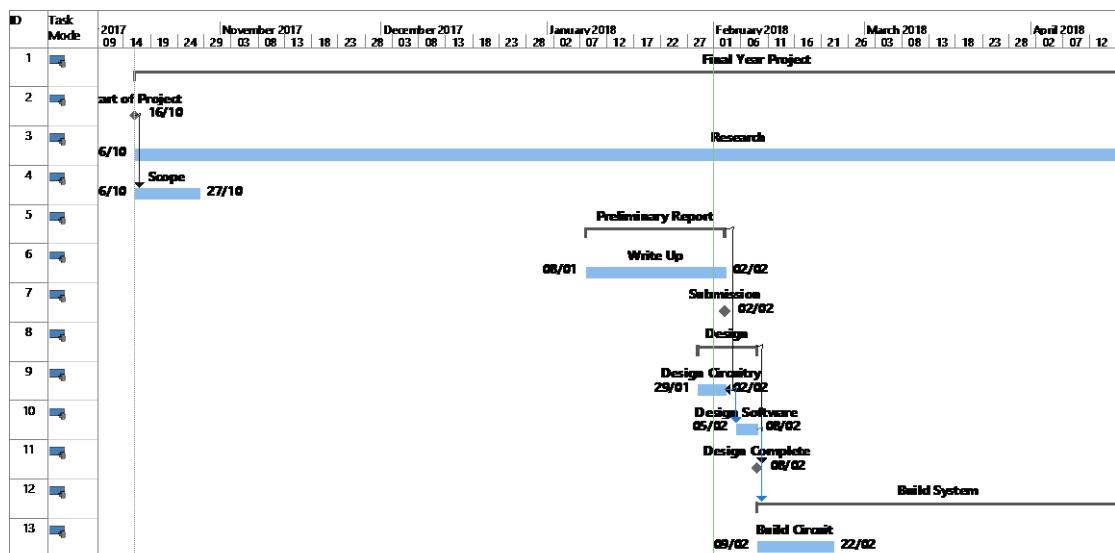


Figure 9: Gantt Chart Detailed 1

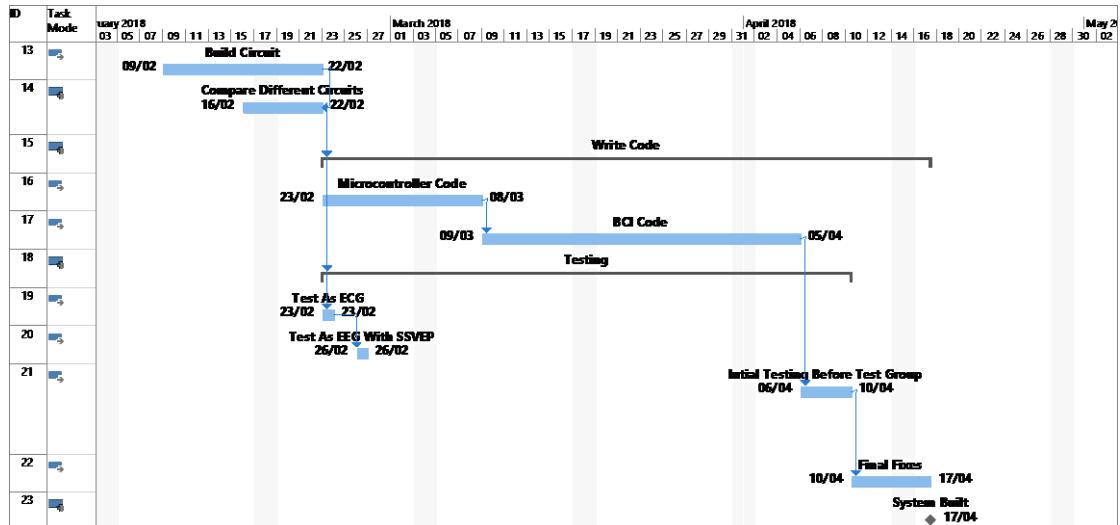


Figure 10: Gantt Chart Detailed 2

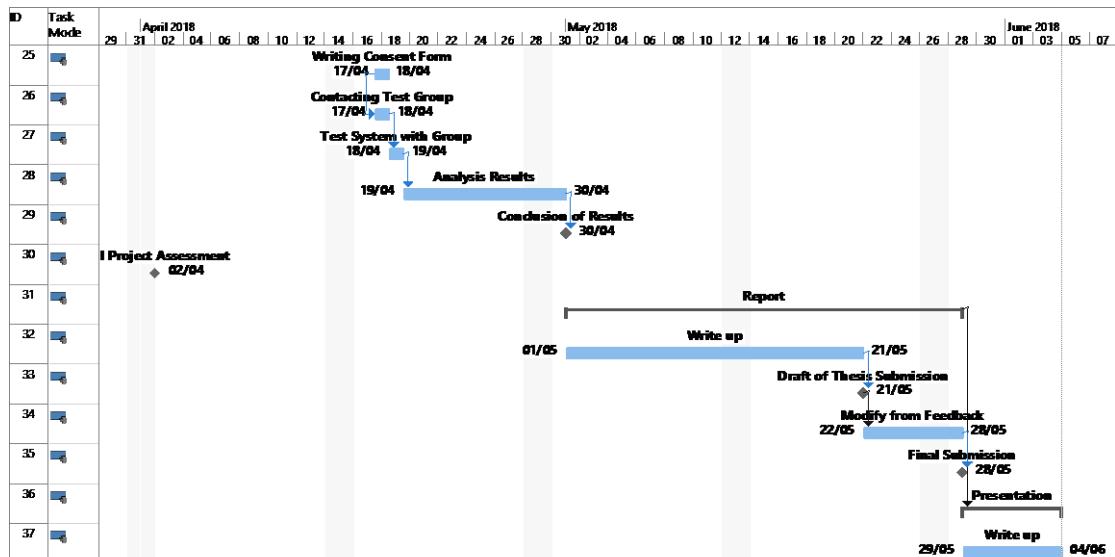


Figure 11: Gantt Chart Detailed 3

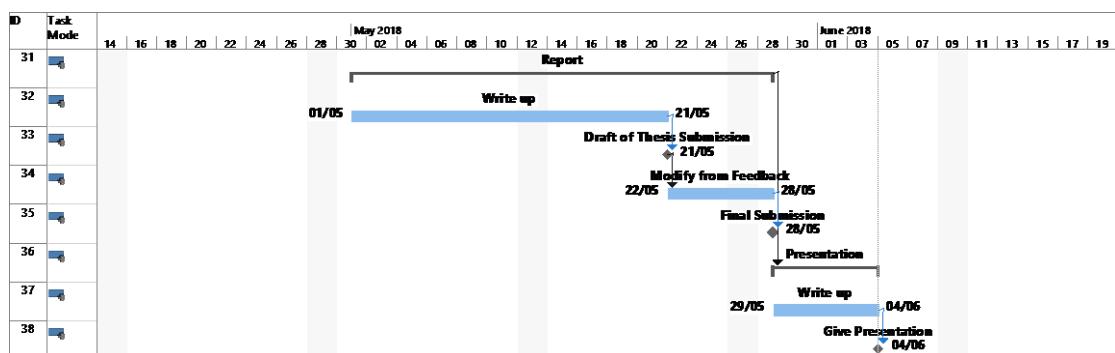


Figure 12: Gantt Chart Detailed 4

PROJECT SAFETY PLAN

All projects conducted in the School of Electrical and Electronic Engineering are required to put in place a Project Safety Plan. This involves identifying the hazards which may be present and stating what control measures will be put in place to control the risk associated with the hazard. This form is typical of that used in other third level institutions.

The following sets out the steps to be taken by the supervisor and the student(s) to ensure the correct completion of the Project Safety Plan.

1. The supervisor should arrange to meet with the student(s) as early as possible in order to review the Project Safety Plan and to complete the Project Safety Plan form. The supervisor should explain any hazards that might arise during the course of the project and ensure that the student(s) understand the nature of the hazard and the control measure to be employed.
2. The supervisor and student must jointly sign the completed Project Safety Plan form.
3. The original signed form (with a copy of the project proposal) must be submitted to the SEEET Administration Office. Both the supervisor and the student(s) should retain a copy.
4. The SEEET Health & Safety Advisory Committee will review the submitted Project Safety Plan and where deemed necessary the form may be returned for further clarification or the inclusion of additional control measures.
5. The Project Safety plan should be reviewed with the student on a regular basis over the project period at project meetings. In particular if over the period of the project the project work plan or focus changes, the Project Safety Plan should be reviewed and a revised Project Safety Plan should be submitted.

Project Safety Plan



Student(s) Name	Ronan Byrne		
Student(s) Number	C13323261		
Programme Code and Year	DT021A/4		
Start Date of Project	16 th October 2017		
Estimated Finish Date of Project	28 th May 2018		
Location(s) of Project (i.e. building and room number)	Kevin Street		
Supervisor Name	Dr Ted Burke		
Title of Project	Development of a Low Cost, Open Source, Electroencephalograph Based Brain-Computer Interface		
Description of Project	<p>An Electroencephalograph based brain-computer interface will be designed, built and tested on a human test group. The circuit will be battery powered and electrically isolated from the computer. The project will be built and tested within college premises. Testing of the device will be supervised. A soldering iron will be used to build the circuit. The project will involve lone working. No chemicals will be used and there will be no field work involved.</p>		
Hazard Identification			
Potential Hazards	Present (Yes/No)	Details	Controls Required
Work Equipment	No		
Work Environment	No		
Electrical Shock Hazard	Yes	Attaching low voltage circuitry and a laptop to the users scalp	Circuit will be electrical isolated from the connected laptop through an optocoupler and the circuit will be battery powered Proper safety procedures will be adhered to while using tools
Hand tools / Power Tools	Yes	Various power/hand tools may be required during the project	
Slips, trips and falls	No		
Manual Handing	No		
Soldering	Yes	Soldering will be required during the project and therefore there is a chance of burning and fume inhalation	Safety glasses and fume extractor will be used during soldering. Soldering will be done in a non-cluttered environment
Compressed Air	No		

Rotating Machinery	No		
Noise	No		
Computer Usage	Yes	A laptop will be used to program the software and the display will be used in the BCI	No control needed
Lone Working	Yes	Project will involve lone working	No control needed
Sudden Illness and/or Medical Emergencies	No		
Fire/Emergency Evacuation	No		
Biological Agents	No		
Chemical	No		
Gases	No		
Heat Sources/High Temperatures/Hot Surfaces	Yes	Soldering will be required during the project	Soldering will be done in a non-cluttered environment
Lasers	No		
Vibrations	No		
Working off Campus / Field Work	Yes	The student may continue work off campus	No control needed
Working at Height (incl. use of ladders)	No		
Cross Contamination between participants	Yes	Cross contamination between participants of hair or skin	Electrodes will be cleaned or replaced between uses
Category of Supervision Required			
Category	Level of Risk Present		
Category A	The risks associated with the work and/or the inexperience of the student(s) are such that the work must be supervised directly by a competent person (the supervisor or his/her authorized nominee), at least until the supervisor is satisfied that the student(s) can follow correctly the appropriate scheme of work. Note: this might apply only to a small section of the whole project.		
Category B	The risks associated with the work and/or the inexperience of the student(s) is such that the work may not be started without the supervisor's or his/her authorized nominee's advice and approval.		
Category C	The Risks are such that the work requires considerable care but it is considered that the student(s) is adequately trained and competent in the procedures involved.		
Category D	The risks are low and carry no special supervision requirements.		

This project plan has been completed to the best of my abilities based on the information available to me, my understanding of the task at hand and of the expected workplace conditions.

Student(s) Sign and Date	Supervisor Sign and Date
Ronan Byrne 02/02/18	Ted Burke 2/2/2018

THE SIGNED FORM SHOULD BE SUBMITTED TO THE SEE Administration OFFICE (THE PROJECT SUPERVISOR AND THE STUDENT(S) SHOULD RETAIN A COPY).