

A 3D Printed, Non-Invasive Brain Computer Interface Prosthetic Hand

Samuel Parker



Completed under the supervision of

Dr Kaushik Mahata, Ph.D.

18 June 2020

A thesis submitted in partial fulfilment of the requirements for the degree of Bachelor of Engineering in Electrical Engineering at The University of Newcastle, Australia.

# Abstract

Millions of people across the world are living with a form of upper limb disability. Impairment of the upper limbs can be caused by a breakdown in the neural signalling pathway, between the motor cortex and the skeletal muscles in the affected limb. In many cases, the function of the cortex remains sufficiently intact for the neural signalling patterns to be intercepted, decoded, and an appropriate motor action executed by an assistive device. This is the role of Brain Computer Interface (BCI) Neuroprosthesis. This project will explore the design and development of a 1 Degree of Freedom (DOF) prosthetic hand, controlled by the real-time, online classification of electroencephalography (EEG) signals obtained from the scalp of a healthy 22-year-old male. It is also hoped that the systems developed as part of this project can become part of an educational biosignal acquisition platform, targeting undergraduate students interested in the application of biosignals in bringing about positive change in the lives of people living with disabilities. To achieve these outcomes, hardware and software design, implementation, and documentation, fused deposition manufacturing (FDM), simulation, data-acquisition, machine learning, embedded systems design, and signal processing was conducted. This report will provide the biological background behind voluntary movement in the context of a BCI system, a summary of non-invasive BCI systems published in literature, a simulation of published feature extraction and classification techniques, an outline of the designed system, and testing methodologies. It was found that a Linear SVM outperforms the other classifiers considered in this report when comparing the systems on the same EEG dataset (74.59% classification accuracy). Due to its sophisticated dimensionality reduction characteristics, features extracted using a Discrete Cosine Transform (DCT) approach result in reduced classification time for equivalent classification accuracy when compared to the current state-of-the-art Short Time Fourier Transform (STFT) approach. The application of the DCT to EEG classification is a novel approach, and could represent a significant decrease in classification time for online BCIs. The proposed system includes a wireless EEG data acquisition system, a Classification and Control board, and a carbon fibre 3D printed InMoov robotic hand, designed by Gael Langevin.

# Acknowledgements

# List of Contributions

# Table of Contents

[Abstract i](#_Toc42799000)

[Acknowledgements ii](#_Toc42799001)

[List of Contributions iii](#_Toc42799002)

[Table of Contents iv](#_Toc42799003)

[Nomenclature vii](#_Toc42799004)

[1. Introduction 1](#_Toc42799005)

[2. Biological Background 3](#_Toc42799006)

[2.1. Neurons 3](#_Toc42799007)

[2.2. Cortical Areas for Voluntary Movement 4](#_Toc42799008)

[2.3. Electroencephalography (EEG) 5](#_Toc42799009)

[3. Literature Review 8](#_Toc42799010)

[3.1. Data Acquisition and Pre-processing 8](#_Toc42799011)

[3.1.1. Signal Recording Methods 8](#_Toc42799012)

[3.1.2. Digitalisation Methods 9](#_Toc42799013)

[3.2. Feature Extraction 10](#_Toc42799014)

[3.2.1. Adaptive Autoregressive Model (AAR) 10](#_Toc42799015)

[3.2.2. Discrete Wavelet Transform (DWT) 13](#_Toc42799016)

[3.2.3. Short Time Fourier Transform (STFT) 14](#_Toc42799017)

[3.2.4. Discrete Cosine Transform (DCT) 15](#_Toc42799018)

[3.3. Classification 16](#_Toc42799019)

[3.3.1. Linear Discriminant Analysis (LDA) 17](#_Toc42799020)

[3.3.2. Quadratic Discriminant Analysis (QDA) 17](#_Toc42799021)

[3.3.3. Support Vector Machines (SVM) 17](#_Toc42799022)

[3.3.4. *k*-Nearest Neighbours (kNN) 18](#_Toc42799023)

[3.4. Simulation 19](#_Toc42799024)

[3.4.1. Simulation Methodology 19](#_Toc42799025)

[3.4.2. Simulation Results 22](#_Toc42799026)

[4. Proposed System 25](#_Toc42799027)

[4.1. EEG Hardware 26](#_Toc42799028)

[4.2. Data Acquisition System Hardware 27](#_Toc42799029)

[4.2.1. Power Supply and Grounding 30](#_Toc42799030)

[4.2.2. ADS1299 30](#_Toc42799031)

[4.2.3. ATmega328PB 31](#_Toc42799032)

[4.2.4. Communications 31](#_Toc42799033)

[4.2.5. Interfaces 32](#_Toc42799034)

[4.3. Data Acquisition System Software 33](#_Toc42799035)

[4.3.1. SPI Timing Requirements for ADS1299 34](#_Toc42799036)

[4.3.2. Configuring and Operating ADS1299 34](#_Toc42799037)

[4.3.3. Processing and Transmission 37](#_Toc42799038)

[4.4. Classification and Control Hardware 37](#_Toc42799039)

[4.4.1. Power Supply 38](#_Toc42799040)

[4.4.2. STM32VGT 40](#_Toc42799041)

[4.4.3. Measurement and Control 41](#_Toc42799042)

[4.4.4. Communications 43](#_Toc42799043)

[4.4.5. Interfaces 43](#_Toc42799044)

[4.5. Classification and Control Software 44](#_Toc42799045)

[4.5.1. Serial Library 44](#_Toc42799046)

[4.5.2. Feature Extraction Module 45](#_Toc42799047)

[4.5.3. Classification Module 45](#_Toc42799048)

[4.5.4. Control Module 47](#_Toc42799049)

[4.6. 3D Printed Prosthetic Arm 47](#_Toc42799050)

[4.7. Demonstration Hardware and Software 50](#_Toc42799051)

[4.7.1. Demonstration App 50](#_Toc42799052)

[4.7.2. Demonstration Board Hardware 50](#_Toc42799053)

[4.7.3. Demonstration Board Software 52](#_Toc42799054)

[4.8. Modular Biosignal Front End 54](#_Toc42799055)

[5. Progression Tasks **Error! Bookmark not defined.**](#_Toc42799056)

[5.1. Testing and Evaluation Processes 57](#_Toc42799057)

[5.1.1. Functionality Testing of ADS1299 57](#_Toc42799058)

[5.1.2. EEG Data Acquisition using the ADS1299 58](#_Toc42799059)

[5.1.3. Establishment of the Bluetooth Link 58](#_Toc42799060)

[5.1.4. Generation of the PWM Control Signal 60](#_Toc42799061)

[5.1.5. Calculating the DCT 60](#_Toc42799062)

[5.1.6. Testing the SVM 61](#_Toc42799063)

[5.1.7. Classifying Real Time Data 61](#_Toc42799064)

[5.1.8. All Together Now! 61](#_Toc42799065)

[5.2. Generating Classifier Training Data 61](#_Toc42799066)

[5.2.1. Training Data Acquisition Setup 62](#_Toc42799067)

[5.2.2. Training Paradigm 62](#_Toc42799068)

[7. Conclusion 65](#_Toc42799069)

[8. Reference List 66](#_Toc42799070)

[Appendix A – Data Acquisition System Schematic A-1](#_Toc42799071)

[Appendix B – Data Acquisition System Gerber Files B-1](#_Toc42799072)

# Nomenclature

|  |  |
| --- | --- |
| AAR | Adaptive Autoregressive |
| ADC | Analog to Digital Converter |
| AFE | Analog Front End |
| Ag | Silver |
| AgCl | Silver chloride |
| AP | Action Potential |
| AR | Autoregressive |
| BCI | Brain Computer Interface |
| CONFIG1 | ADS1299 configuration register number 1 |
| CONFIG3 | ADS1299 configuration register number 3 |
| DC | Direct Current |
| DCT | Discrete Cosine Transform |
| DOF | Degree(s) of Freedom |
| DRDY | ADS1299 Data Ready Pin |
| DWT | Discrete Wavelet Transform |
| ECoG | Electrocorticography |
| EEG | Electroencephalography |
| EMI | Electromagnetic Interference |
| ERD | Event Related Desynchronisation |
| ERS | Event Related Synchronisation |
| FDM | Fused Deposition Modelling |
| FIFO | First In, First Out |
| FRI | Florida Research Instruments |
| FTDI | Future Technology Devices International |
| GPIO | General Purpose Input Output (pin) |
| IDE | Integrated Development Environment |
| IMDCT | Inverse Modified Discrete Cosine Transform |
| K | Potassium |
| kNN | *k*-Nearest Neighbours |
| LDA | Linear Discriminant Analysis |
| LSTM | Long Short-Term Memory (classifier) |
| M1 | Motor Cortex |
| MDCT | Modified DCT |
| MISC1 | ADS1299 reference electrode configuration register |
| MIT | Massachusetts Institute of Technology |
| MOSI | Master Out Slave In (SPI communication) |
| MS | Multiple Sclerosis |
| MSB | Most Significant Bit |
| Na | Sodium |
| PC | Personal Computer |
| PCB | Printed Circuit Board |
| PWM | Pulse Width Modulation |
| QDA | Quadratic Discriminant Analysis |
| RBF | Radial Basis Function |
| RC | Resistor-Capacitor |
| RDATA | ADS1299 command to retrieve data |
| RESET | Command to reset ADS1299 |
| SCLK | Serial Clock Pin |
| SDATAC | Command to stop ADS1299 continuous data conversion |
| SPI | Serial Peripheral Interface |
| SSVEP | Steady State Visual Evoked Potentials |
| START | ADS1299 command to start conversion |
| STATUS | ADS1299 Communication Header Packet |
| STFT | Short Time Fourier Transform |
| SVM | Support Vector Machine |
| TI | Texas Instruments |
| UART | Universal Asynchronous Receiver/Transmitter |
| USB | Universal Serial Bus |
| Vpp | Volts (peak to peak) |
|  | A Normal distribution function |
|  | Quantisation Coefficients for DCT Analysis |
|  | Forgetting Factor in AAR model |
|  | Error in the Least Squares criterion |
|  | Noise function in AAR model |
|  | Order of AAR model |
|  | Chip Select Pin (active low) |
|  | Number of bits per ADS1299 reading |
|  | Number of ADS1299 Channels |
|  | DCT Coefficients |
|  | Quantised DCT Coefficients |
|  | Period of ADS1299 Data Rate |
|  | Period of Serial Clock (SPI communication) |
|  | Time delay between sending bytes during a multi-byte command to ADS1299. |

# Introduction

Across the globe, millions of people are living with upper limb disability [1]. This disability may be caused by several factors, including amputation, spinal cord injury, paralysis, stroke, or degenerative disease [2-5]. This disability reduces the independence and quality of life of the patient, resulting in increased levels of care and medical attention. In many cases, the cortical activity of people living with these conditions remains unaffected. Brain Computer Interface (BCI) technology aims to utilise the patterns of neuronal activity to decode the intent of the patient, then implement the intended outcome with the assistance of a computer [6]. Currently, research is being conducted at several leading universities across the globe to develop advanced BCI neuroprosthetic devices [7, 8]. These devices will use the patient’s neural signalling to control a robotic prosthetic, restoring their ability and independence. These devices will also reduce the financial impact on healthcare services presented by the care and rehabilitation of people with an upper limb disability. Most of the advanced neuroprosthesis being researched rely on implanted subcranial cortical electrode arrays [9]. This approach is called electrocorticography (ECoG), and offers a higher signal to noise ratio, and better spatial resolution than other approaches. However, a survey conducted in 2015 indicated that the high risk associated with the implantation procedure made this approach unattractive to people living with upper limb disability [10].

This project will assess the aptitude of electroencephalography (EEG), a non-invasive neural recording approach, as a BCI input modality. It is anticipated that through the application of several signal processing techniques, including the Short Time Fourier Transform (STFT) and Discrete Cosine Transform (DCT), the most advantageous approach can be devised. Similarly, by comparing several classification techniques including *k*-nearest neighbours (kNN), Support Vector Machines (SVM), and Adaptive Autoregressive techniques (AAR), the most superior classifier (most accurate per unit time) can be determined. Some publications have conducted a similar comparison [11-14], however these publications compared each approach on different datasets, acquired under different conditions. Therefore, conducting a comparison of feature extraction and classification techniques on the same dataset represents a novel contribution.

Once the most advantageous approach has been determined, it is anticipated that a system to implement this approach in an online BCI neuroprosthetics setup can be designed. To reduce artefacts (introduced from lead-sway, switching transients, etc.), a wireless EEG data acquisition system will be designed. Additionally, the classification and control hardware will be designed to fit within the spatial constraints of a 3D printed prosthetic forearm.

Through the development of a non-invasive online BCI neuroprosthetic device, an understanding of the advantages and disadvantages of EEG as a BCI input modality can be ascertained. It is hoped that the insights gained throughout the classification methodology assessment and the implementation of the system may guide future research in this area.

# Biological Background

The human brain is an incredible complex entanglement of various specialisations of cells. The modern field of neuroscience can be traced back to the 19th century, when Jean Legallois concluded that the medulla oblongata had a dominant role in respiration [15]. Since then, while unprecedented progression and understanding has occurred, there are still many neurological processes, such as memory and consciousness, with unknown mechanisms. Recently, advanced studies in the physical structure of the brain, connectomics, aims to increase humanity’s understanding by examining the connectedness of functional structures of the brain. Additionally, if the neural connections are known to a sufficient degree, whole regions of the brain may be computationally simulated [16]. This would provide a testbed for novel neuroactive medicines, or predictions of the impact of a stroke in a particular brain region.

## Neurons

The general term for the cells that make up the brain are neurons. A neuron consists of dendrites (inputs), a soma, an axon, and axon terminals (outputs) [17]. The neurons that make up the brain communicate through electrochemical signalling, called action potentials (APs). The cell membrane of the neuron sits at a resting potential of approximately -70 mV [18]. The membrane potential is set by the concentration of different ions inside the intracellular fluid of the soma. The intracellular fluid contains a higher concentration of potassium ions ([K+]) and a lower concentration of sodium ions ([Na2+]) than the intercellular fluid. The concentration gradient of the sodium ions is greater than that of the potassium ions, due to an increased permeability of the membrane to potassium [19].

Received excitatory APs from neighbouring neurons raises the membrane potential of the neuron incrementally, until a threshold potential is reached, when a series of events occur. Initially, voltage gated sodium channels open, allowing an influx of Na2+ ions from the intercellular fluid to enter the cell, depolarising it further. Once the membrane potential is slightly more positive than the intercellular fluid, voltage gated potassium channels open, allowing for K+ efflux from the cell body, and the voltage gated sodium channels close. This allows the cell to depolarise again. When the cell returns to its resting potential, the voltage gated potassium channels close [20].

Figure 1 - A schematic diagram indicating the locations of cortical areas responsible for voluntary movement [21].

APs initiate in the soma of the neuron, then propagate along the axon to the axon terminals, where the neuron may synapse (connect) to a neighbouring neuron [22]. After passing through various cortical areas, the signal may synapse to a projection neuron, which will carry it down the spinal cord and into the peripheral nervous system.

## Cortical Areas for Voluntary Movement

The brain has been divided into defined cortical areas, each with a specific set of connections and functions. The cortical area responsible for voluntary motor control is the motor cortex (M1) [23], located in the precentral gyrus, and shown in Figure 1.

The M1 is present in the left and right hemispheres of the brain. Left side movements are lateralised to the right hemisphere M1, while right side movements are lateralised to the left hemisphere M1, thus the M1 possesses contralateral control [24].

The imagination or execution of a motor control command (e.g. right-hand grasp) will case an Event Related Desynchronisation (ERD) or an Event Related Synchronisation (ERS) [25]. During an ERD, neurons which were typically firing APs at a similar frequency will begin to fire APs at differing frequencies, resulting in a decrease in the spectral power of the old, synchronous frequency. Contrarily, during an ERS, neurons which were typically firing APs at different frequencies will begin to fire APs at a similar frequency, resulting in an increase in the spectral power at the new, synchronous frequency [26]. Many BCI systems classify motor intent by considering the change in spectral power characteristic to an ERD/ERS [27-29].

During the trial conducted by Burgess in [30], participants were presented with a series of photos of faces. Their EEG spectral density pre- and post-stimulus was compared to identify the incidence of a “seen” or “new” face. The key results relating to ERD/ERS is shown in Figure 2. Note that there is a significant reduction in spectral power at 10 Hz in the post-stimulus curve, characteristic of an ERD. Both curves were recorded at the Oz electrode site (on the centre line, above the occipital lobe) in the 10-20 international system.



Figure 2 - Comparison of pre- and post-stimulus action potential frequency at the Oz electrode, indicating an ERD [30].

## Electroencephalography (EEG)

EEG is a low-cost, non-invasive recording method to measure neural activity. It relies on the electrical currents generated by populations of thousands of neurons firing action potentials to conduct a small voltage signal to the cutaneous tissue of the scalp. The signals are detected using an array of electrodes.

Figure 3 - (a) A gold cup "wet" electrode [31], and (b) a “dry” electrode [32].



(a)

(b)

Since EEG is measuring the activity of a general population of neurons, the spatial resolution of EEG is low. It is difficult to distinguish activity originating directly under the EEG electrode from activity emanating from a cortical region under an adjacent electrode [33]. However, EEG systems offer a high temporal resolution, limited only by the sampling rate of the data acquisition system [34]. This makes EEG neural recording methodologies attractive in a time-frequency Brain Computer Interface environment.

The electrodes commonly used to record the scalp potentials are classified as “wet” or “dry”. Wet electrodes require extensive scalp preparation, and the application of a conductive gel [35]. This made wet electrodes cumbersome, and slowed the testing process. Recent advances in dry electrodes have improved their signal acquisition properties to perform similarly to wet electrodes [36-38]. Dry electrodes do not require extensive preparation of the scalp, and can perform effectively through hair, due to the spiked nature of the electrode. Figure 3 compares the visual characteristics of wet and dry electrodes.

The international standard EEG electrode locations follow the 10-20 system, presented in Figure 4. A standardised approach to electrode location ensures the same cortical areas are measured across trials and subjects. Additionally, this approach enables hardware to be manufactured to suit the standard, rather than follow proprietary location schemes. Wang et. al, identify the optimal locations for EEG electrodes to record motor imagery are C3, C4, and FCz on the 10-20 scheme, as these locations correspond to the right hand, left hand, and sensory motor area respectively.

Figure 4 - A diagram representing the 10-20 standard EEG electrode locations [39].

# Literature Review

To facilitate an online Brain Computer Interface control system, it is widely accepted that three main functional blocks are required. These blocks are presented in Figure 5.

Figure 5 - System block diagram indicating the basic functional blocks of a BCI system

Each of these blocks can consist of different approaches and procedures. A literature review, in conjunction with a Matlab simulation, was conducted to predict the relative performance of each potential system. This enabled a reduction in the number of potential solutions, and the eventual derivation of an optimal system.

## Data Acquisition and Pre-processing

To analyse the neurological changes initiated by motor imagery processes, a method of effectively capturing the electrical signalling of the brain is required. This method should minimise the effect of noise from external sources. Then, since the device is a Brain Computer Interface, a method of digitising, communicating, and recording the results is necessary. Following this, some pre-processing may be necessary.

### Signal Recording Methods

There are several well documented means of recording neural activity from the human brain. At a high level, these can be divided into invasive and non-invasive methods. Invasive methods, such as electrocorticography (ECoG), offer some advantages over non-invasive methods. These methods exhibit reduced signal attenuation due to the sensors being placed under the parietal bones of the skull [40], improving the signal to noise ratio. They are also not susceptible to artefacts introduced from oculomotor or skeletal motor events. However, placement of these subdural electrodes poses significant risk, and is usually only implanted for clinical reasons in patients for epilepsy monitoring [41]. This risk is so significant, that a survey conducted by Engdahl et al. concluded that only 39% of participants indicated an interest in prosthesis controlled by (implanted) cortical interfaces, compared to 83% for myoelectric control [10]. Additionally, the scope of this report restricts the final system to be low-cost, which precludes an invasive recording method.

Scalp-based electroencephalography (EEG) is a common low-cost, portable recording method [42]. Using this approach, electrodes are placed on the scalp of the subject, which detect the ionic currents generated by the brain during an action potential [36]. These signals are extremely small, and are prone to radiated EMI from proximal sources. For decades, the accepted standard for electrodes were Ag/AgCl wet electrodes. These devices were passive, and required extensive preparation of the skin and electrode to ensure a low electrode-skin impedance. Usually this preparation is conducted by a trained specialist, took several minutes, and left a gel residue on the subject’s head [37]. Recently, however, new technology in the form of an active, dry electrode has enabled rapid acquisition of EEG signals, requiring practically no setup. These dry electrodes have been evaluated by multiple sources to perform to an equivalent standard as wet electrodes [36-38]. Such devices are attractive in this project, as they will enable rapid testing and development, at a low cost.

### Digitalisation Methods

Due to the signal’s susceptance to EMI, and to reduce the impact of aliasing introduced by time-domain sampling, most systems proposed in literature have included some form of lowpass filtering on the input stage of the analog to digital converter (ADC) [43, 44]. Application notes for some EEG analog front end (AFE) integrated circuits indicate that a single order passive RC lowpass filter is sufficient to mitigate the coupling of high frequency noise sources, and reduce the amplitude of high frequency content of the signal, in turn reducing the effect of aliasing [45]. This is an attractive solution, since these filters consume no power, enabling the data acquisition front end to be powered from a DC battery. Powering the device in such a manner reduces the likelihood of introducing conduction EMI into the signal from a rectified source [46].

Published literature appreciates that designing an AFE from scratch is a non-trivial task [47], and many papers suggest using an off the shelf component [48]. Several studies have selected the Texas Instruments (TI) ADS1299 chip to serve as the AFE when designing a BCI [47-52]. This chip is an attractive solution, due to its 24-bit resolution, high sample rate range (from 250 to 16k samples per second), high common-mode rejection ratio (-110 dB), low input referred noise (1 µVpp over a 70Hz bandwidth), and small, self-contained package [53]. The performance of this chip exceeds the performance of the AFE designed by Hu et al. in [47]. The chip interfaces to a microcontroller through an SPI interface. Additionally, the ADS1299 can be configured to use a single reference electrode, in a referred montage scenario, or establish differential signals when recording in a differential montage [54]. An evaluation board is available for this chip, however, TI has published a recommended schematic and PCB layout, which may serve as a valuable starting point from which a custom front end can be designed [45]. Thus, due to extensive prior works, and detailed supporting information, the TI ADS1299 AFE may be a viable solution to the EEG digitalisation problem.

## Feature Extraction

To reduce the computational intensity required to classify the EEG event, a feature extraction algorithm is applied to the acquired signal. Proposed algorithms include a Short Time Fourier Transform approach (STFT), an Adaptive Autoregressive Model (AAR), a Discrete Wavelet Transform (DWT), and a Discrete Cosine Transform (DCT) approach. It is noted that the articles cited use different testing data and methodologies, thus a valid absolute performance comparison cannot be made without evaluating these approaches against the same dataset. Additionally, the quoted performance of the feature extraction algorithm is dependent on the performance of the classification technique employed by the study, again reducing the validity of an absolute performance comparison.

### Adaptive Autoregressive Model (AAR)

An Adaptive Autoregressive Model (AAR) is a variation on an Autoregressive Model (AR), which allows the parameters of the model to adjust over time. This feature is beneficial when applied to an EEG signal, as the signal is non-stationary [55]. An AAR model of order is described by:

(1)

, and

(2)

Where is the n-th sample of the EEG observation, is the zero-mean-Gaussian noise with variance , and are the time-varying AR coefficients. Thus, a sample is predicted by the past samples, with new information introduced to the model through the innovation function , which in the above example is a normal distribution [56].

In some models, it is not beneficial to apply an equal weighting to all historical values. In this case, a “forgetting factor” is introduced, where:

(3)

which usually takes the form .

As proven by Haykin in [57], a recursive algorithm is applied to find the vector such as to minimise the sum of the error squares:

(4)

where is the vector of desired responses.

Applying the following variable changes:

(5)

enables the criterion presented in Equation (4) to be rewritten as the standard least squares criterion:

(6)

Thus, assuming a causal signal, the least squares solution can be obtained as:

(7)

where:

(9)

(8)

It can be shown through the matrix inversion formula, that:

(10)

can be written as:

(11)

Defining and

(12)

it can be shown that:

(13)

Thus, it is now possible to derive a time-update equation for :

(14)

where

Now all the required equations to form the recursive least squares algorithm have been produced, and are summarised as follows:

(16)

(15)

(17)

(18)

It is shown by Haykin that this recursive algorithm can be solved with relative computational ease, however the computational intensity increases with [57].

Such a model considers only time-domain parameters, thus utilising the high temporal resolution offered by EEG data acquisition systems at the peril of ignoring the prominent frequency-domain characteristics induced by ERPs. Low order models () are not computationally intensive, however accuracy increases with model order. Prior works indicate that AAR approaches may result in low accuracy, with Rodríguez-Bermúdez & García-Laencina reporting a 62.2% accuracy [58].

### Discrete Wavelet Transform (DWT)

The Discrete Wavelet Transform (DWT) is a time-frequency approach, thus utilising both the temporal resolution of EEG signals, and the frequency characteristics of ERPs. This approach involves passing the signal through a multi-stage filter bank, consisting of highpass and lowpass filters. The highpass filtered signal produces the “detail coefficients” at that level of the filter bank, while the lowpass filtered signal produces the “estimate coefficients”. The lowpass filtered signal is then passed to the next level of the filter bank, as shown in Figure 6.

Since the frequency content of the signal has been bandlimited to one half of its original bandwidth, by Nyquist’s theorem, the signal can be subsampled by a factor of 2, reducing the number of samples at each step. In turn, this doubles the frequency resolution. Thus, DWT approaches offer high spectral resolution over the low frequency content. This characteristic is beneficial when extracting features from EEG data for motor imagery, since these signals occur in the low frequency spectrum, however a higher resolution will increase the computation time compared to analysing the STFT over a similar bandwidth.

Figure 6 - The filter bank representation of the DWT [59].

When paired with a Long Short-Term Memory (LSTM) classifier, Jie Zhou et al. achieved an accuracy of 91.43% on the Graz 2003 Brain Computer Interface Competition III dataset [60]. Thus, the DWT method may be suitable for online motor imagery feature extraction.

### Short Time Fourier Transform (STFT)

The Short Time Fourier Transform (STFT) approach is another time-frequency feature extraction method, again making use of the high temporal resolution offered by EEG signals and the identifiable frequency characteristics of ERPs. Under this approach, a sliding feature extraction window of defined length is passed over the EEG signal (in online analysis, this can be implemented using a FIFO buffer), and the discrete Fourier transform (DFT) of the signal in the window is taken. There are several windows to choose from, including:

1. Rectangular. This is the least computationally intensive window, however leads to increased sidelobes in the frequency domain which will need to be filtered.
2. Triangular. This window is slightly more computationally intensive, however the sidelobes will be somewhat compressed.
3. Hanning. This window is more computationally intensive, however offers the best compression of frequency domain sidelobes.

Huang et al. tested rectangular, triangular, Hamming, and Hanning windows when extracting features from Steady State Visual Evoked Potentials (SSVEP). Each of the windowed signals were passed to the same Linear Discriminant Analysis (LDA) classifier. It was found that if no highpass filter was applied, the triangular, Hamming, and Hanning windows performed equivalently (classification accuracy was within 0.5%), while the rectangular window did not perform as well. However, after a 4th order Butterworth highpass filter (cut-off frequency of 2 Hz) was applied to all the windowed signals, the rectangular window outperformed the rest of the windows. Thus, it is recommended by the authors of that paper that a rectangular window be used in conjunction with a 4th order highpass filter, for SSVEP analysis [61]. However, if a 4th order highpass filter is not applied, a triangular window is a computationally efficient, high-performance window function.

To circumvent the amplitude suppression introduced at the fringes of the window, the feature extraction window is “slid” over the length of the EEG signal. This means that each time domain sample of the signal (except those within one half of the window length from the extremities of the signal) spend equal amounts of time at the centre of the window.

Coyle et al. implemented a STFT approach to extract features from left and right hand movement EEG signals. The features were then classified by an LDA classifier. This study produced an average classification accuracy of 88.425% across 4 trials on 3 subjects [62]. Thus, a STFT approach to feature extraction may be a viable solution to online BCI.

### Discrete Cosine Transform (DCT)

The Discrete Cosine Transform (DCT) was developed by Chen and Pratt in 1984, and is similar to the DFT, however carries some advantages. Rather than decomposing the signal into a summation of sine and cosine functions (represented as complex exponentials), the DCT uses only cosines, and is thus completely real-valued. This reduces computational complexity [12]. The DCT coefficients, , for an -point signal, , can be computed by Equation (27) [63].

(19)

Additionally, through applying a zonal coding strategy, the coefficients obtained by applying Equation (27) can be compressed. A quantisation vector of values, , is devised such that:

Selective determination of the quantisation coefficients can reduce the amplitude of some frequency contributions to such a degree that they can be ignored, effectively compressing the signal [64]. The quantisation coefficients are often chosen such that the quantised signal described by possesses 90% of the power of the original signal described by . In many cases, this results in the high frequency DCT coefficients being eliminated, as the low frequency components of the signal possess more power [65]. Since EEG signals are usually low frequency, and any high frequency elements of the signal are generally noise or artefacts, this behaviour is desirable. Thus, the DCT process contains some inherent dimensionality reduction, which is favourable when performing online signal classification.

(20)

Applying the DCT to a time-frequency application introduces the modified discrete cosine transform (MDCT). This approach was developed by Princen and Bradley in 1986 to reduce the effect of time domain aliasing introduced by windowing the signal [66]. This approach consists of windowing the signal with 50% overlapped windows (i.e. the latter half of the first window includes the same data as the first half of the second window), then applying the DCT. When the inverse MDCT (IMDCT) is applied, each block introduces several artefacts. However, when the subsequent block of inverse transformed data is added, the errors introduced cancel out via time domain aliasing cancellation [67]. Since the recorded EEG signal will not need to be reconstructed in this application, the reconstruction errors introduced by time-domain aliasing artefacts will not be considered.

## Classification

There are several methods used to classify the features extracted from an EEG signal. Methods proposed by Wang et al. include Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), Support Vector Machine (SVM) prediction, and *k*-Nearest Neighbours (kNN) algorithms [14]. SVMs and LDA were observed to be widely used in classifiers [14, 58, 68, 69].

### Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis (LDA) has historically been one of the most common classification techniques for BCIs. LDA aims to find a linear combination of features that can best separate two or more classes [13]. Following training of the classifier on a training data set, a set of eigenvectors representing each class is obtained. When a test observation is to be classified, the linear combination of features developed during testing is used to generate an eigenvector representative of the observation. The classifier then selects the class that produces the minimum Euclidean distance between the class eigenvector and the observation eigenvector [69]. As LDA is a machine learning approach, LDA classifiers can suffer from overfitting, especially when the training dataset is small. However, it is proposed by Pang et al. that regularising the data by unsupervised clustering can reduce the likelihood of overfitting [70]. Similarly, since this approach is linear, LDA tends not to suffer from overfitting as much as nonlinear classifiers [11] but must operate under the assumption that the covariance of each class is the same [71]. Wang et al. achieved a classification accuracy of 82.86% on the 2003 Gratz BCI Competition Dataset III when classifying motor imagery EEG signals. The authors concluded that LDA, as a linear classifier is suitable for BCI applications due to its simplicity and stability [14].

### Quadratic Discriminant Analysis (QDA)

Quadratic Discriminant Analysis (QDA) operates in a similar manner to LDA, however removes the assumption that the covariance matrices of each class is the same [72]. This removes the benefits discussed in Section 3.3.1 (resistance to overfitting) and introduces additional complexity. Wang et al. achieved a classification accuracy of 78.57% on the 2003 Gratz BCI Competition Dataset III when classifying motor imagery EEG signals [14]. Thus, QDA is not a competitive classification solution in this application.

### Support Vector Machines (SVM)

Support Vector Machines (SVM) are another popular approach to EEG classification. These systems again employ machine learning, however in this case the goal is to define an *n*-1 dimensional plane in *n*-dimensional Euclidean space that divides the data classes, where the Euclidean distance between each class is maximised [73, 74]. This is achieved by first finding the support vectors. These are *n*-dimensional vectors defining the data points that are closest to the hyperplane (that is, hardest to classify). There is a theoretically infinite number of hyperplanes that can divide the classes [75], thus the SVM solves a trivial optimisation problem to determine the optimal plane. Once a new data point is supplied for classification, the dot product of the new data point’s vector and the plane is taken to determine what side of the plane it falls on. This is very computationally simple. The elegance of SVMs surrounds their implementation of the “kernel trick”, which enables non-linearly separable data to be separated by defining additional separation functions called kernels. These kernels can include polynomials, exponentials, and sigmoids [76]. Wang et al. achieved a classification accuracy of 82.86% on the 2003 Gratz BCI Competition Dataset III when classifying motor imagery EEG signals using a Linear SVM, and 84.29% using a Gaussian SVM [14]. Thus, the SVM is a viable solution to the EEG classification problem.

### *k*-Nearest Neighbours (kNN)

A *k*-Nearest Neighbours (kNN) algorithm establishes each training data point in an *n*-dimension parameter space, which can be defined by a vector of features (the features are obtained by the methods outlined in Section 3.2). When a new data point is to be classified, it is placed into the same parameter space. The data point is then classified by a voting system, where the class of the new data is the class of the majority of its “*k*-nearest neighbours”, or *k* closest training data points [77]. There are several ways to calculate the “distance” between neighbouring data points. The most basic is Euclidean distance. However, measures such as cosine similarity, defined below, are also used in some applications (generally when the data to be classified is not sparse, creating large angular differences).

Where, is the similarity of data points and , and is the angle between data points and [78].

(21)

This calculation is not computationally intensive in lower dimensions. However, the kNN algorithm suffers drastically in higher dimensions, as almost the entire parameter space is required to find the closest *k* data points, particularly if the data points are not close to each other [79]. This increases computation time significantly. Thus, in order to be implemented in an online BCI system, an extensive (accurate) feature selection procedure is required.

Wang et al. achieved a classification accuracy of 84.29% on the 2003 Gratz BCI Competition Dataset III when classifying motor imagery EEG signals using a kNN approach [14]. Thus, the kNN approach is a viable solution to the EEG classification problem, however would require additional feature selection to be viable in an online classification setting.

## Simulation

As noted previously, most publications collate classification results comparing feature extraction or classification techniques using different datasets. By applying the approaches established above to a single dataset, relative performance can be assessed, and the optimal system can be determined, from which a system can be designed and implemented.

### Simulation Methodology

Using the dataset published by Kaya, et al. [80], classifications made by Quadratic, Gaussian, Radial Basis Function (RBF), Polynomial and Linear SVMs, LDA, and kNN classifiers were compared when supplied with features extracted by STFT and DCT methods. The classification time and accuracy of each approach was compared, and the optimal system was defined as the system that produced the highest classification accuracy per unit time.

The dataset contains EEG recordings from a healthy 20-25 year-old female with no prior BCI experience. There are recordings from 19 10-20 system recording sites, referenced to two ground leads attached to the earbuds. The signals were 0.53 – 70 Hz bandpass filtered by the recording software, and recorded at 200 samples per second. The amplitude resolution of the recordings was 0.01μV at 24 bits. The data was recorded in a synchronous BCI paradigm (i.e. the start and end times of the event were dictated by software, not the user). When an event was started, a stimulus was provided on a screen for 1 second, during which the subject imagined the corresponding movement (right hand closed, left hand closed, or idle). Following the 1s stimulus, a random screen-off time of 1.5 – 2 seconds occurred, allowing the subject to reset between events. The stimulus was encoded into the dataset [81].

To evaluate each extraction and classification technique, first the EEG data was low-pass filtered using a 10th order Butterworth filter with a cut-off frequency of 30 Hz. The data was then epoched. This was achieved by identifying the rising edge of the “marker” signal (i.e. where the stimulus changed from “idle” to “right hand”, “left hand”, or “rest”), then saving the following 2 seconds of data in its own vector. To enable 2 class classification, “left hand” trials were removed. The C3 data was then multiplied by a Hanning window to reduce the frequency domain artefacts, and the data from other channels was disregarded.

In the STFT case, by applying a 215-point FFT to the 2 second epoch, the frequency spectrum of each event was calculated. By observing the time-frequency distribution across all trials, it was clear that all the activity was bounded below the 2458th frequency bin, so these low-frequency bins were selected as features. This distinction can be seen by observing Figure 7.



(a)

(b)

Figure 7 - Heatmaps of spectral power obtained by taking the STFT of the EEG dataset presented in [81]. (a) shows the full feature vector, while (b) limits the features to below 2458 Hz.

In the DCT case, the DCT coefficients of each 2 second epoch were determined, and these coefficients were used as the classification features. No further feature selection was required, as the DCT exhibits inherent dimensionality reduction.

The sets of features were passed to several classifiers. The classifiers were as follows:

1. Linear SVM,
2. Gaussian SVM,
3. RBF SVM,
4. Polynomial SVM,
5. LDA, and
6. kNN (15 neighbours).

Each classifier had a holdout percentage of 20%. The time taken to classify the remaining datasets was measured using Matlab’s tic and toc functions. Training time was not included, as in the proposed system, the model will be trained offline, then applied online. The simulations were run using Matlab 2016a on a system consisting of an Intel Core i7-5500U (2.4 GHz) with 16 GB of DDR3 RAM (1600 MHz).

### Simulation Results

Figure 8 shows the accuracy of each of the classification algorithms when classifying the STFT feature set. Under this approach, the Linear SVM had the greatest accuracy, 71.70%, and a run-to-run variance of approximately 11%. This was achieved in 0.0148 seconds. This gave the Linear SVM the highest overall score of 48.52. The accuracy, classification time, and score of each classifier is presented in Table 1.

Table 1 - A summary of classification results obtained using the STFT feature set.

|  |  |  |  |
| --- | --- | --- | --- |
| Classifier | Accuracy (%) | Classification time (seconds) | Score |
| Linear SVM | 71.70 | 0.0148 | 48.52 |
| Gaussian SVM | 50.94 | 0.0534 | 9.54 |
| RBF SVM | 50.94 | 0.0526 | 9.69 |
| Polynomial SVM | 66.04 | 0.0266 | 24.86 |
| LDA | 50.31 | 0.0311 | 16.16 |
| kNN | 64.78 | 0.0774 | 8.37 |

Figure 8 - A graph showing classification accuracy for several classifiers when supplied with the STFT feature set.

Figure 9 shows the accuracy of each of the classification algorithms when classifying the DCT feature set. Again, the Linear SVM had the greatest accuracy, 74.59%, and a run-to-run variance of 9%. This was achieved in 0.0038 seconds. This gave the Linear SVM the highest overall score of 196.30. The accuracy, classification time, and score of each classifier is presented in Table 2.

Table 2 - A summary of classification results obtained using the DCT feature set.

|  |  |  |  |
| --- | --- | --- | --- |
| Classifier | Accuracy (%) | Classification time (seconds) | Score |
| Linear SVM | 74.59 | 0.0038 | 196.30 |
| Gaussian SVM | 50.79 | 0.0095 | 53.46 |
| RBF SVM | 50.79 | 0.0075 | 67.72 |
| Polynomial SVM | 50.79 | 0.0038 | 133.66 |
| LDA | 67.02 | 0.0079 | 84.84 |
| kNN | 48.69 | 0.0210 | 23.185 |

Figure 9 - A graph showing classification accuracy for several classifiers when supplied with the DCT feature set.

These results clearly show the improvements in classification time brought about by the DCT’s reduced feature set. Classification accuracies remained similar between the two approaches. Therefore, due to the significant decrease in classification time, the optimal feature extraction technique analysed is the DCT.

While the DCT has been extensively studied in the field of image compression [65, 82, 83], applying the DCT to EEG classification is a novel approach, which may carry several benefits, including reduced classification-space dimensionality, leading to reduced classification time. This is especially attractive to online BCI systems, where classification time is necessarily as short as possible.

# Proposed System

Following the completion of the simulation, the most advantageous system had been identified. This enabled a proposed system to be developed. Figure 5 now can be considered as shown in Figure 10.

Figure 10 - A modified system block diagram, indicating the selected functional blocks of the proposed system.

The architecture of such a system could be designed as presented in Figure 11.



**Electrical Isolation**

Figure 11 - A schematic diagram showing the implementation blocks and interfaces for the proposed system.

## EEG Hardware

Following the literature review, it was clear that dry electrodes would be preferential to wet electrodes due to their single person operation and reduced setup time. After comparing suppliers, Florida Research Instruments (FRI) offered an extensive range at a reasonable cost. They offer package deals, bundling an EEG headband with several dry electrodes and leads. The headband does not offer the full 10-20 location set, however it includes several key sites, including C3 and C4. The headband is shown in Figure 12.

Figure 12 - A photo of the Florida Research Instruments EEG Headband [84].

The proposed system will only require 1 EEG channel (C3), however several electrodes will be purchased for potential future expansion by the University of Newcastle. The system will be designed to accommodate 8 channels, as this may be desired in future, educational applications. Additional to the signal electrode, the system requires a ground electrode in the form of an ear clip. FRI offers these devices. The final bill of materials for the EEG hardware is presented in Table 3.

Table 3 - Bill of Materials for EEG Hardware.

|  |  |  |  |
| --- | --- | --- | --- |
| Item | Quantity | Price (AUD) | Notes |
| Basic EEG Headband Package | 1 | $104 | Includes:   * 15 Dry Electrodes * 10 Wet Electrodes * An EEG Headband * 5 Leads |
| Silver/Silver-Chloride Ear Clip | 1 | $29.62 |  |

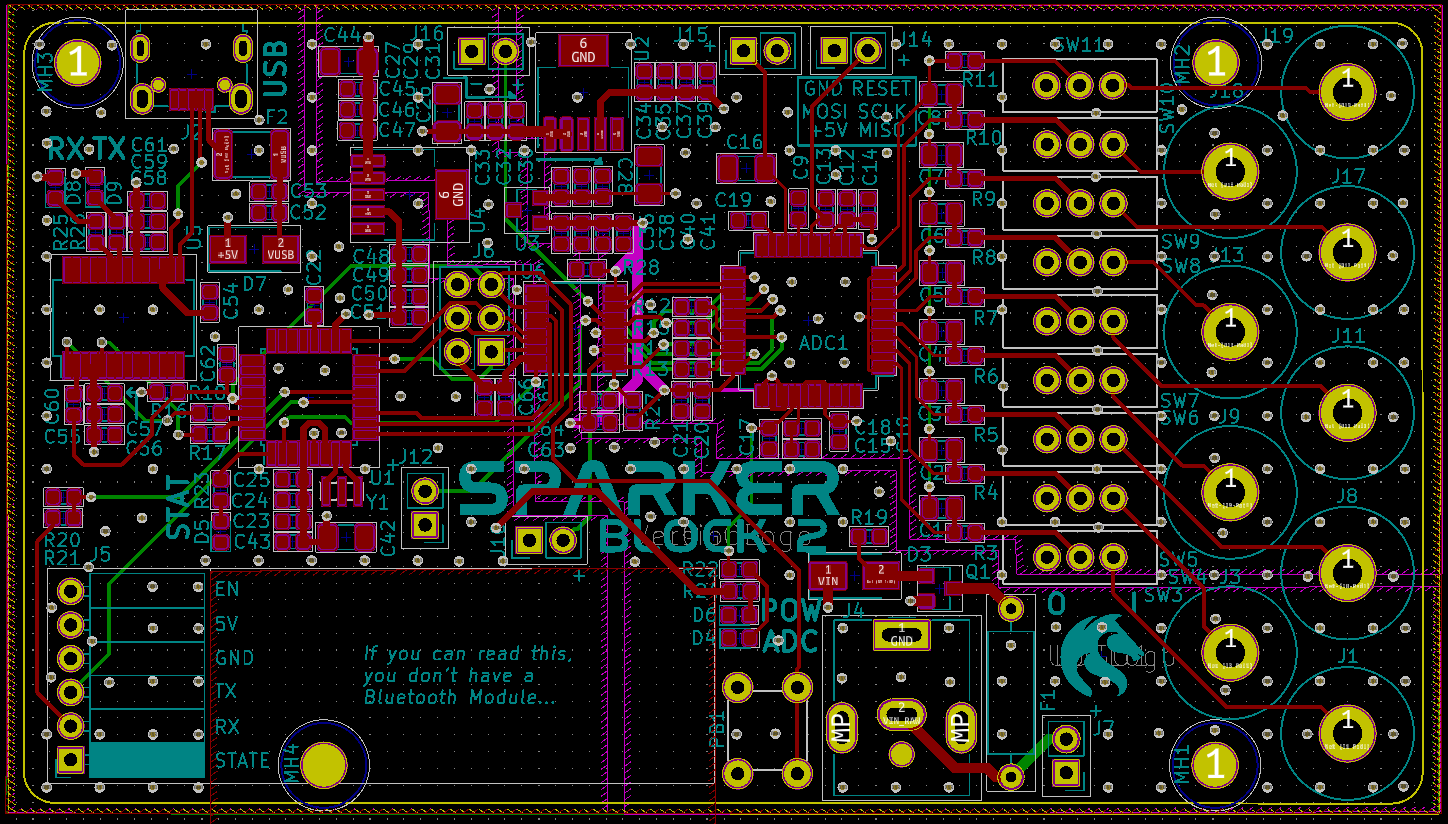
## Data Acquisition System Hardware

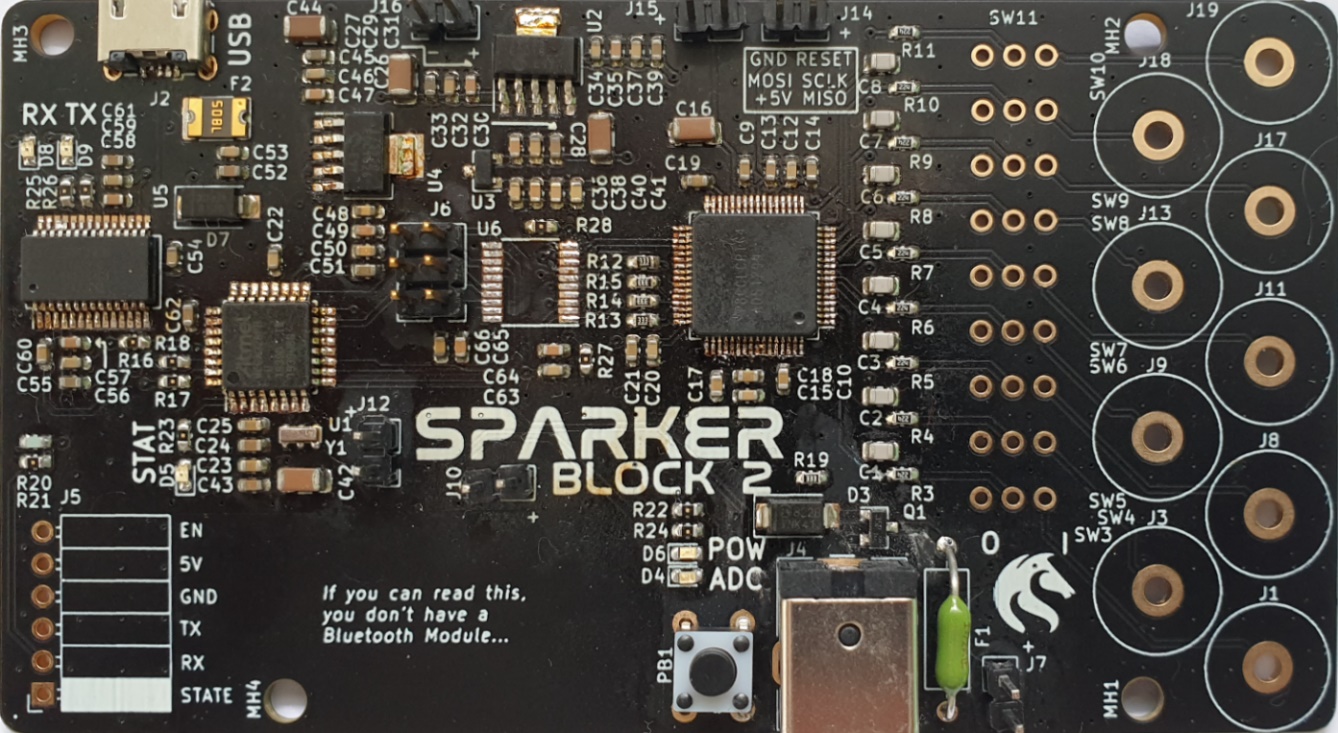
The signal detected by the electrodes will be processed by the data acquisition system. This system contains passive filtering components, analog to digital conversion, and wireless transfer, facilitated by an ATmega328P microcontroller. The design criteria for this system include a small footprint, since it will be mounted to the head of the subject, and power efficiency, to enable battery operation (reducing the impact of rectification or switched mode power supply noise on the analog signals).

The incoming EEG signals will be passively low-pass filtered, using an RC filter with a cutoff frequency of approximately 30 Hz. While this filter is of lower order than that used in the simulation, it will still offer adequate rejection of high-frequency noise. The filtered signal will then be digitised using a delta-sigma ADC. The Texas Instruments ADS1299 was found during the literature review to be a high-performance ADC with several EEG-specific features [47-52]. To facilitate future expansions, the Data Acquisition System will be designed to allow the operation of all 8 channels available on the ADS1299 model. This was completed despite this application only requiring 1 channel, which could have been adequately served by the lower-cost, 4-channel ADS1299-4. This chip was selected to enable future utilisation of this hardware for applications requiring the additional channels. The ADS1299 will sample the input signal (referenced to the reference electrode) at 250 samples per second in continuous conversion mode. This sampling frequency is sufficiently high to detect the low-frequency signals present in EEG, and is higher than the sampling frequency used in the data published in [81].

The data will be read from the ADC over an SPI bus by the ATmega328PB. The microprocessor will then decimate the signal to 125 samples per second. This reduces the required communication bandwidth and reduces the length of the time domain vector from which features are extracted. The ATmega328PB then performs a simple conversion to UART data, which will be sent by an HC-05 UART-to-Bluetooth module to the Classification and Control board for feature extraction and classification. The layout of the Data Acquisition System is presented in Figure 13, and a render of the board is presented in Figure 14. The schematics of the Data Acquisition System are presented in Appendix A – Data Acquisition System Schematic. The PCB Gerber files are presented in Appendix B – Data Acquisition System Gerber Files. A Bill of Materials (BoM) is presented in <APPENDIX>. This section will decompose the board into functional sub-circuits and discuss the associated design choices.

Power can be supplied to the board through a DC connector, capable of regulating 16 V. The linear regulator used has a dropout voltage of less than 1V at full load, so the device can alternatively be powered by connecting a 9V battery to the battery connection header. This will increase portability and reduce conducted EMI from a mains-powered rectifier and switch mode power supply. The completed layout has a board area of 95mm × 60mm, which is small enough to comfortably attach to posterior portion of the EEG head strap.

Figure 13 - The PCB layout of the Data Acquisition Board.

Figure 14 - A photo of the Data Acquisition Board.

### Power Supply and Grounding

This board contains both analog and digital signals. This introduces additional PCB layout considerations, especially surrounding isolating analog signals from EMI generated by digital switching transients [85]. ST Electronics has compiled a list of general recommendations for mixed signal design, which have been followed in the design of this board [86]. Two 5V voltage rails are created from the 6V – 16V input, one for the 5V digital logic and the other for the 5V analog supply required by the ADS1299. This ensures the transients introduced onto the 5V rail by the MCU are not present on the measured analog signal. Additionally, every power pin on the 4 ICs on the board is decoupled by a collection of decoupling capacitors. These capacitors are placed as close to the power pin as possible, to minimise the inductance of this matching network.

The requirement of an unbroken ground plane resulted in the development of a 4-layer PCB. This ensured that the current return path was as short as possible. Further, ground pours were placed on all unused space on the top and bottom layers, and these pours were stitched to the internal ground plane using a grid of vias spaced approximately every 4mm. These parallel vias reduce the impedance to ground, and ensure there is no “dead copper”, which can act as an antenna and couple with high frequency noise.

The board is fitted with reverse polarity protection. Simulating the circuit in Analog Electronics’ LTSpice indicated that under reverse polarity conditions, the protection devices dissipated picowatts (pW) of power, and did not present any negative voltages at the terminals of the LDOs.

### ADS1299

The Texas Instruments ADS1299 chip has strict support circuitry requirements. These are outlined in the device datasheet [54], and were confirmed through correspondence and design reviews with TI application engineers. Critically, there are several analog bypass capacitor pins (VCAPx), which are highly sensitive to capacitance and ESR. The ADS1299 is capable of outputting a current signal used to bias the scalp of the patient to the middle of the ADC measurement range. Communicating with the Psychology Department at the University of Newcastle, it was conveyed that patient biasing is rarely used, and accurate signals can be recorded without implementing this feature, however it may be implemented future iterations of this design. The ADS1299 digital input pins are rated to a maximum voltage of 3.6V. The first iteration of the Data Acquisition System did not account for this, which lead to the destruction of an ADC. This was circumvented using a bidirectional logic translator, as outlined in Section 4.2.4. Care was taken to isolate the analog signals from the portions of the PCB that would be susceptible to digital noise (from the MCU) or RF noise (from the Bluetooth module). Additionally, the pin is pulled HIGH by the logic translator during the programming phase of the MCU to limit the reception of unexpected instructions since the MCU is programmed over the SPI interface.

### ATmega328PB

The implementation details for the ATmega328PB were based on the Arduino Nano schematic [87]. The chip will be programmed using Atmel’s ICSP protocol, and loaded with the OptiBoot bootloader. This will allow the chip to be programmed via USB.

### Communications

The Data Acquisition System has 2 external communication channels. First, a UART for the Bluetooth module is used for sending EEG data to the Classification and Control Board under normal operation. To simplify processing, this channel is free of other data. To facilitate debugging and communicating additional messages to the operator, a USB connection is provided for connection to a PC.

When the Data Acquisition System is generating training data for the SVM classifier model, the channels are reversed. Now, the instructions are provided over Bluetooth, for display on the user’s smartphone. The USB interface is used by the operator to set parameters and to gather the data. This data can be read directly by Matlab, or saved to a .csv/.mat file for offline processing.

As previously mentioned, the ATmega328PB and ADS1299 use different logic levels (5V and 3.3V respectively). To facilitate communication between these two devices, a TXS0108EPWR 8-bit bidirectional level shifter translates the SPI interface and control signals between these logic levels.

### Interfaces

The external interface specifications for this board are presented in Table 4. Additionally, the internal interface specifications for this board are presented in Table 5.

Table 4 – External interface specifications for the Data Acquisition board.

| Interface | Connector | Notes |
| --- | --- | --- |
| Analog Signal Input | J1, J3, J8, J9, J11, J13, J17 – J19 | 1 Reference Signal (J19), 8 Data Signals. |
| Serial-over-USB | J2 | Mini-USB port to UART facilitated by FTDI chip. FTDI drivers required. SW2 must be set to enable USB communication. |
| Battery Power In | J7 | 6V – 16V battery connection. |
| DC Power In | J4 | 6V – 16V DC barrel jack. |
| Bluetooth | J5 | UART to Bluetooth module header, suitable for HC-05. |
| ICSP | J6 | Compatible with ATMEL In-System Debuggers [88]. |

Table 5 – Internal interface specifications for the Data Acquisition board.

| Interface | Devices | Notes |
| --- | --- | --- |
| SPI | ATmega328PB and ADS1299 | SCLK frequency 1MHz. Passes through TXS0108EPWR level shifter. Bidirectional. |
| UART1 | ATmega328PB and FTDI Chip | 115200 baud, bidirectional. |
| UART2 | ATmega328PB and HC05 | 38400 baud, bidirectional |
| START | ATmega328PB and ADS1299 | A single pin to control data conversion. If taken LOW, the MCU can initiate and cease data conversion using SPI commands. Otherwise, data is converted when this pin is HIGH. |
|  | ATmega328PB and ADS1299 | A single pin to place the ADS1299 in standby mode. |
|  | ATmega328PB and ADS1299 | A single pin taken LOW by the ADS1299 to indicate that data has been converted and is ready to be read. |

## Data Acquisition System Software

The software for the ATmega328P on the Data Acquisition board establishes a connection with the ADS1299, sets up the ADS1299 for the required configuration, establishes a connection with the Bluetooth module, and facilitates real-time data transfer between these two devices. The documentation for the Data Acquisition Software is presented in <APPENDIX>. An overview of the whole software system is presented in Figure 15.

Figure 15 - Software module interface diagram for the Data Acquisition board.

### SPI Timing Requirements for ADS1299

There are several timing requirements for the SPI interface of the ADS1299. When the device is first addressed by taking the LOW, there must be a 6 ns delay between LOW and the rising edge of SCLK. The SCLK frequency is determined by Equation (30), from the ADS1299 datasheet [54].

(30)

where is the period of the serial clock, , is the period of the master clock, is the ADC resolution, and is the number of channels being sent. In this application, , since the requested data rate is 250 samples per second. , and is defined by the internal oscillator of the ADS1299. , since each reading is a 24-bit value, and , since the project is using the ADS1299 chip (8 channels) rather than the ADS1299-6 (6 channels) or the ADS1299-4 (4 channels). Substituting these values into Equation (30) yields a , equivalent to an .

When sending multi-byte commands, a period of must elapse between each byte. is defined as 4 clock periods. The internal oscillator is being used in this application, which has a period of 2.048 MHz. Therefore, .

### Configuring and Operating ADS1299

The ADS1299 is an extremely versatile biosignal analog front end. To configure its many parameters, it has several configuration registers that must be set appropriately for each application. These registers are configured by writing commands over the SPI interface. The SPI interface is set to operate in SPI Mode 1 (CPOL = 0, CPHA = 1).

Figure 16 indicates the process of configuring the ADS1299. Once the flow chart in Figure 16 has been completed, the device is ready to start the data conversion. This is triggered by setting the START pin HIGH, or setting the START pin LOW and sending the START command over the SPI bus. Once the conversion has started, the DRDY pin is monitored. When the DRDY pin goes LOW, there is 1 sample of data in the output shift register. The RDATA command can be used to retrieve the data. During this read action, no data should be written to the MOSI line. The data will arrive in 9 24-bit packets, MSB first. The first packet is a STATUS packet, containing lead-off and GPIO information. The remaining 8 packets contain the channel data, starting with Channel 1 and ending with Channel 8. This data can be processed, and sent over the serial bus as required, then the DRDY line should be monitored again, waiting for more data.

The ADS1299 has 10 functions, which have been enumerated, allowing for a single function to be called by passing it a command. This provides a higher-level interaction layer than what would be used if the protocol was interacted with directly. Similarly, the device has 24 configuration registers, each with up to 8 configurable parameters. Additional to functions used to read from or write to a whole register, “get” and “set” functions have been written for each parameter to enable a more intuitive interface between the devices. Each setting for each parameter has been enumerated, making the configuration process much more human-readable. Examples of the bare-protocol, and the readable protocol is presented in Table 6.

Table 6 – A comparison of the bare protocol to the implemented human readable protocol.

|  |  |  |
| --- | --- | --- |
| Bare Protocol | Readable Protocol | Notes |
| SPI.transfer(0x08) | ADS1299.send\_command(START) | Start data conversion |
| SPI.transfer16(0x4500)  SPI.transfer(0x21) | ADS1299.set\_channel\_gain(CH1, PGA4) | Set Channel 1 gain to 4 |
| SPI.transfer16(0x5700)  SPI.transfer(0x08) | ADS1299.set\_conversion\_mode(SINGLE\_SHOT) | Set conversion mode to Single Shot |

Figure 16 - ADS1299 configuration flowchart.

The MCU holds information about each register in a Register Array. Each register is represented by a structure, containing information such as a unique, enumerated ID, the register’s default value, a read-only flag, etc. It is desirable to structure the C++ code in such a way as to ensure that if the number of registers changes, patterns such as loops or conditions do not need to be rewritten throughout the entire program. Additionally, adding or removing a register should not require this change to be mirrored in multiple places, as this risks sections of code becoming outdated, leading to bugs, or even crashing the program. This has been achieved through X-Macros, which utilise the C pre-processor to automatically generate the code to create the array of register structures, and enumerate the Register IDs at compile-time, rather than run-time. In this implementation, there is one “truth-source”, a #defined Register Table, from which the enumeration, structures, and arrays are created. It also creates the boundary conditions that valid Register IDs are compared against, or looped over. This is an incredibly useful code pattern, and has contributed to the readability, performance, and flexibility of this software.

These software elements have been implemented to make the software more useable and robust to future changes. While they are not directly needed to make this project “functional”, they present value as they will make future work on software platform more approachable.

### Processing and Transmission

The data from the ADS1299 will then be sent over the UART serial bus to the Bluetooth module, where it will be transmitted to the Classification and Control board. Bluetooth was selected as there are readily available, low cost commercial modules available, and a wireless protocol enables total electrical isolation of the head-based unit from the arm. This reduces the risk of electric shock to the user. The data will be processed into 9 32-bit integers. The first integer contains the sample ID, a sequential number, starting at 1. A sample ID of 0 will correspond to an error. The next 8 32-bit integers is the channel data, starting with Channel 1, and ending with Channel 8. At the completion of the transmission, the newline character will be sent to signify that a new sample is about to begin.

## Classification and Control Hardware

The feature extraction and classification method was implemented on an STM32F407VGT (“STM32”) Cortex-M4 based microcontroller from ST Electronics. This processor was selected due to familiarity with the system and prior experience with the discovery board. Additionally, it operates with a 32-bit architecture, enabling greater precision and speed during arithmetic operations [89]. Similarly, the 168MHz clock speed will be sufficient to apply the DCT and classify the signal online. There is significant documentation to accompany this chip, which may serve to reduce design complexity.

The STM32 chip will first be provided the classification vector determined by the linear SVM model in Matlab using the USB interface. This vector will be saved in memory, and subsequent feature vectors will be compared to this vector to determine the class of the observation. Once the classification vector has been received, the STM32 will read data from a HC05 Bluetooth-to-UART module. Once an epoch worth of data (1 to 2 seconds) has been received and compiled into a time-domain vector, features will be extracted to create the observation vector, which will be classified by the linear SVM. Once a class has been determined, the STM32 will generate the required PWM signals to move the fingers to the desired position. While the fingers are moving, the current drawn by each of the servos is measured to determine if any finger has become “stuck” or blocked by an object. If the current usage rises above a threshold, the fingers will stop moving, to prevent dangerous crushing actions.

Training of the linear SVM model will be conducted offline using Matlab. Once the EEG hardware has been acquired, it will be determined if the dataset provided by Kaya et. al in [81] is representative of the data acquired in this application. If this is true, the data in [81] will be used as training data. However, if the differences are too large, and a reliable classifier cannot be trained using this data, a training paradigm will be devised to train the SVM using data acquired using the system outlined in this report. This training paradigm will be discussed further in Section 5.2.

The control and classification board should be installed inside the prosthetic arm. This poses significant size constraints. Using Autodesk Tinkercad, the boundaries of the arm void were determined. These are presented in Figure 17.

Using the application notes provided by ST Electronics and the STM32 Discovery Board Schematics, the Classification and Control Board schematic and subsequent PCB was developed [90, 91]. The complete schematic is presented in <APPENDIX>. The PCB Gerber files are presented in <APPENDIX>. The Classification and Control Board BoM is presented in <APPENDIX>. This section will discuss the design rationale to support each sub-circuit of the board.

### Power Supply

Both the high-power servos and low-power microcontroller must be supplied from the same power source. This was selected to be a 7.5V, 2A laptop power supply, as this allowed the servos to operate at the top end of their voltage rating while

Figure 17 - A CAD model of the prosthetic forearm, with a shape representing the maximum PCB area.



Prosthetic Forearm

Maximum PCB Area

52.00mm

62.85mm

consuming their rated current. Stepping down the voltage to levels appropriate for the microcontroller occurred in 2 stages. First, the 7.5V is stepped down to 5V using an LDO regulator. This created a 5V rail from which to power the HC05 Bluetooth module. The 5V rail was then stepped down to 3.3V to power the microcontroller, again using an LDO. Each of these LDOs were adequately decoupled, beyond the minimum specifications provided in their datasheets <REFERENCE><REFERENCE>. Additionally, bulk supply capacitors were included to smooth any voltage transients induced by driving the servos.

A schematic of the reverse polarity protection system included on this board is presented in Figure 18. Simulating this circuit in Analog Electronics’ LTSpice indicated that under reverse polarity conditions, the devices dissipated picowatts (pW) of power, and did not present any negative voltages at the terminals of the initial LDO.

Figure 18 – A schematic of the Classification and Control Board reverse polarity protection circuitry.

Due to the space restrictions imposed by placing the PCB in the arm, the power connection has been set back from the edge of the board, and a right-angle power connector is necessary.

### STM32VGT

By examining the STM32 Discovery Board schematic [90], appropriate modifications could be made to tailor this board to the application required by this project. The STM32F407VGT chip is provided by ST Electronics with a bootloader, which upon power-up will check the state of the BOOT0 and BOOT1 pins. If BOOT0 is LOW, the chip will poll the SWD interface and load any data present into its flash program memory. If no data is present, it will commence the program stored in flash. This is the boot configuration required for this project, as no external SRAM is necessary or included on the board. The bootloader configuration has been left configurable, however, as this feature may be desired in the future. The bootloader configuration implementation information is presented in Figure 19. The STM32 clock is provided with an 8MHz crystal resonator. By configuring the device PLL such that and , then setting the PLL as the clock source, the system synthesises its maximum frequency, 168MHz.

Figure 19 – A schematic of the Classification and Control board bootloader configuration circuit.

### Measurement and Control

It is desired to measure the current consumed by each servo. This is achieved through a 50mΩ current sense resistor placed in series with the power supply pin of each servo. By measuring the voltage drop across the current sense

resistor, then applying Ohm’s Law, the current consumed can be calculated. Since the power supply of all the servos is at the same potential, it can only be measured once, then used for all current calculations. Therefore, to measure the current consumed by all 6 servos, 7 ADC channels are required. The servos operate at from 0V to 7.5V. This range is too large to be measured by the STM32 (the maximum ADC supply voltage is 3.6V) [91]. Therefore, the supply voltage, and the voltage after the current sense resistor must be divided using a resistor divider to within the measurable range. Dividing the voltage by 6 will achieve this. Therefore, each ADC channel measures its voltage after a 500kΩ series resistor, and has a 100kΩ shunt resistor to ground. These large values were selected to minimise the current that travels into this node, as this leaves more supply current available for the servos. A diagram of the measurement topology is presented in Figure 20 and Figure 21. Each ADC channel takes its measurement from the feedback points FB\_REF, FB1 through FB6.

Figure 20 – The Classification and Control Board servo current consumption measurement voltage dividers.

Figure 21 – The Classification and Control Board servo connections, showing current sense resistors.

### Communications

The Classification and Control Board has 2 communication channels. First, a UART for the Bluetooth module is used for receiving EEG data from the Data Acquisition System. To simplify processing, this channel is free of other data. To facilitate debugging and loading the classification vector, a USB connection is provided. This physical connection enables the classification vector to be uploaded to the Board by Matlab using the same script created to generate the SVM model. It can also provide debug messages, for example, when the sample ID of a sample received from the Data Acquisition System does not match the expected sample ID.

### Interfaces

The various on-board components will interface to the external components through the connections specified in Table 7. Similarly, the on-board components will communicate with each other using the interfaces presented in Table 8.

Table 7 – External interface specifications for the Classification and Control board.

|  |  |  |
| --- | --- | --- |
| Interface | Connector | Notes |
| Bluetooth | 6-Pin Horizontal Female Header | UART to Bluetooth module header, suitable for HC-05. This will communicate with the Data Acquisition Board or Android App. Mounted on back side of board |
| SWD | 6-Pin Vertical Male Header | A single wire debugging programming header, for use with an ST-LINK V2 programmer. |
| DC Power In | 2.1mm ID, 5.5mm OD Barrel Jack | 6V – 16V DC barrel jack. |
| Servo Headers | 6 x 3-Pin Horizontal Male Headers | Headers to power and control the servos. One connector per finger, one for the wrist. |
| Micro-USB | Horizontal Micro-USB Female Connector | Mounted at board edge to facilitate communication with PC. |

Table 8 – Internal interface specifications for the Classification and Control board.

|  |  |  |
| --- | --- | --- |
| Interface | Devices | Notes |
| UART1 | STM32 and FTDI Chip | 115200 baud, bidirectional |
| UART2 | STM32 and HC05 | 38400 baud, bidirectional |

## Classification and Control Software

The software for the classification and control board is comparatively more complex than the data acquisition board. While the software on the ATmega328PB had many hardware interfaces, the software on the STM32 has many other software modules to interface with, each completing a component function to achieve the online classification of EEG signals. An overview of the software structure is presented in Figure 22.

Figure 22 - Software module interface diagram for the Classification and Control board.

This section will follow the dataflow through the software, and discuss the role of each module.

### Serial Library

Utilising the Hardware Abstraction Layer (HAL) functions provided by ST Electronics [92], a 38400 baud connection to the Bluetooth module is established. During the setup phase, the serial library is provided with the *n*-dimensioned separator vector (the output of the SVM model that divides the two classes). This vector is passed to the Classification Module. The serial module is provided with a buffer sufficiently sized to contain a single epoch of data. Once this buffer fills, an interrupt is triggered. In the interrupt service routine, the data is copied out of the serial buffer, and the interrupt is re-enabled. While the buffer is filling, the epoch that triggered the interrupt can be processed uninterrupted.

### Feature Extraction Module

Once a full epoch of data has been collected, a pointer to the epoch is passed to the Feature Extraction Module. This module first applies a triangular window to the data, and computes the DCT of the signal, transforming the signal from the time domain into feature-space. This process is the most computationally intensive link in the software chain. A modified version of the DCT, the Fast DCT-2 (FDCT2) has been implemented by Kaushik Mahata. This minimises the invocation of the cos() function, reducing classification time. The module can successfully convert, window, and transform a 128-sample epoch while maintaining its real-time guarantee. This ensures the system will not bottleneck, and can perform classification in real time. The quantisation vector ( in Equation (20)) has been hardcoded into the classification software. This vector was determined by examining the EEG signals recorded by Kaya et al. in [81], then using Matlab to calculate the quantisation coefficients such that the quantised DCT of the EEG signals contains 90% of the power of the unquantised DCT. Once the observation vector has been computed, a reference to this vector is passed to the Classification Module.

### Classification Module

The classification module is provided with two inputs. As mentioned in Section 4.5.1, this module is provided with the decision vector, which is determined offline by the Matlab Linear SVM training script. The module is also passed each observation vector. Vapnik and Cortes describe methods for determining the class of an observed vector in [73]. Their paper indicates that the linear decision function value, , can be found using the following relation:

where is the dot product between the support vector and observation vector in feature space, is the scaling coefficient of the *i*-th support vector, and is an offset [73]. Both and are parameters of the support vector model generated by Matlab. Therefore, by determining the sign of the decision function, an epoch of EEG data can be classified as “hand open” or “hand closed”. This is represented by Figure 23. The green observations are to the left of the hyperplane, and the result of the dot product of the decision vector and the observation vector plus the offset is negative. If the observation was on the hyperplane, the result would be 0.

(31)

Figure 23 – A diagram showing the value of the decision scores for each class in a linear SVM [93].

Hence, once provided the decision vector and the observation vector, and knowing the SVM parameters and *b*, the Classification Module can perform a simple dot product to determine the motor intent of the user. This is extremely computationally efficient, since this application is a linear SVM. Other kernel functions are not as efficient, and require more complicated calculations to determine the result of the decision function [93].

### Control Module

The implemented Control Module detects object collisions through comparing the current drawn by each servo to a predetermined threshold. This is achieved through concurrently reading 7 ADC channels (one reference voltage and one channel per servo), and calculating the current draw using Ohm’s Law. The ADC data is loaded into device memory using a DMA controller, initialised using the HAL functions. If it is determined that a finger can move in the desired location (i.e. the current draw remains below the threshold), the Control Module will use the determined classification class label calculated by the Classification Module to generate an appropriate PWM signal using the HAL functions. Since the SVM is performing binary classification, the two defined PWM duty cycles for each position have been determined empirically, and defined in the software. Ramping between these two positions has not been implemented, however is trivial to include.  
  
While this control methodology is relatively simple, the interfaces to other functions have been established and tested. This has laid the foundation for a more complicated control algorithm (possibly including pressure transducers on the fingertips) to be implemented with minimal alteration to the other software modules.

## 3D Printed Prosthetic Arm

The end effector of the system will be a 3D printed prosthetic arm. This device will serve as visual feedback for the subject. The arm was designed by Gael Langevin, and has gained popularity as a section of the InMoov 3D printed humanoid robot. These designs are freely available, and were selected for their detailed assembly instructions, low cost, aesthetics, and global community of makers, who were of great assistance during the assembly process. The arm consists of approximately 26 .stl files, which were printed by Phillip Dombkins at the University of Newcastle.

The arm was printed on a Markforged 3D printer using Onyx carbon fibre filament. This printer was capable of printing to an extremely high resolution, contributing to the ease of assembly. Additionally, the printed structures are lightweight and exhibit excellent structural rigidity, both characteristics favourable to prosthesis design [94].

The main components of the arm are the servo bed (which houses the 5 servos for the fingers and is located in the forearm), the wrist, and the hand. The servo bed was designed to accommodate 5 standard size servos. The Tower Pro MG946R servo is recommended, and has been used in this project. Due to the space constraints within this wrist cavity, a smaller sized servo, the Tower Pro MG996 was used. These servos

have a rated stall torque of 13 kg/cm, and draw 1.2A at locked rotor. Due to the high current rating of these servos, supplying adequate current to operate 5 servos simultaneously is a rigid design criterion which has be considered in the classification and control board design. The hand operates by rotating the servos inside the forearm. Ligaments of nylon braid connect each side of the servo horn to the

fingertip of the corresponding finger. When closing the hand, the servo horn turns through 180o, increasing the tension on the tensor ligament running along the superior side of the hand, while simultaneously decreasing tension on the extensor ligament on the inferior side of the hand. This causes the finger to contract. The fingers are hinged by slotting a pin of filament into the holes located at the joints. The pins are then friction mounted to the external casing of the hinge, while the internal component is free to rotate.

The arm was straightforward to assemble. Minimal tools were required, as most of the components were secured using Phillips head screws or Loctite 401. Loctite 401 was used on recommendation from Loctite for fusing nylon components. Other cyanoacrylates may produce a less successful bond between the components. Photos of the arm during assembly and post-assembly are presented in Figure 24.

Figure 24 - (a) The inferior surface of the constructed arm. (b) the superior surface of the constructed arm. (c) the arm being assembled, with the cover removed.



(a)



(b)



(c)

## Demonstration Hardware and Software

To facilitate demonstration of the arm during the interim seminar, a demonstration board possessing minimal functionality has been developed and fabricated. This board can activate the 6 actuation points in the prosthetic arm, and is controlled via Bluetooth using an Android app.

### Demonstration App

The demonstration app was developed rapidly in MIT App Inventor 2, a web-based Android app programming tool. The app prompts users to connect to the arm by presenting them with a list of paired Bluetooth devices. Once connected, the 6 sliders (1 for each servo) become active. When the position of a slider is changed by the user, the app sends “<Servo\_Name>: <position>\n” where <Servo\_Name> is the servo being controlled, e.g. “wrist”, and <position> is the slider position in percent, where 0 corresponds to fully closed and 100 corresponds to fully open. Additionally, there are 3 buttons along the bottom of the screen to set the hand to preselected poses: fist (all fingers at 0), point (all fingers at 0 except the index finger, which is at 100), and open (all fingers at 100). The commands are terminated using a newline character. This app represents basic functionality, and is the minimum viable system to demonstrate the function of the prosthetic hand.

A screenshot of the app is presented in Figure 25.

### Demonstration Board Hardware

A smaller cavity was found in the arm, and it was proposed that the demonstration board fit in this cavity. The dimensions of the cavity are approximately 70 × 30mm. The limited available space restricts the demonstration board to including the minimal hardware required to operate the arm wirelessly. This limits the components to:

1. an Arduino Nano, soldered directly to the PCB,
2. an HC-05 Bluetooth Module, soldered directly to the PCB,
3. a DC power jack, and
4. the 6 servo headers for the servos.

Building the demonstration board around the Arduino Nano accelerated the development of this board. The Arduino Nano is self-contained, so it was not

Figure 25 - A screenshot of the demonstration app.

necessary to devote time to designing the power circuits, oscillators, indicator LEDs, etc. Similarly, the UART to USB interface was also included. This limited the hardware design to simply arranging the peripheral hardware. The layout and render of the demonstration board is presented in Figure 26. One drawback to this design is that the Arduino cannot be reprogrammed once it has been soldered to the device, as the serial programming ports are shared with the Bluetooth module, which cannot be disconnected. A second iteration of this design would have included a switch to disconnect the Bluetooth Module. Headers could not be used due to the limited vertical space.



(a)

(b)

Figure 26 - (a) the PCB layout of the demonstration board, and (b) a 3D render of the demonstration board.

### Demonstration Board Software

The software required to operate the demonstration board is minimal. The basic flowchart is presented in Figure 27.

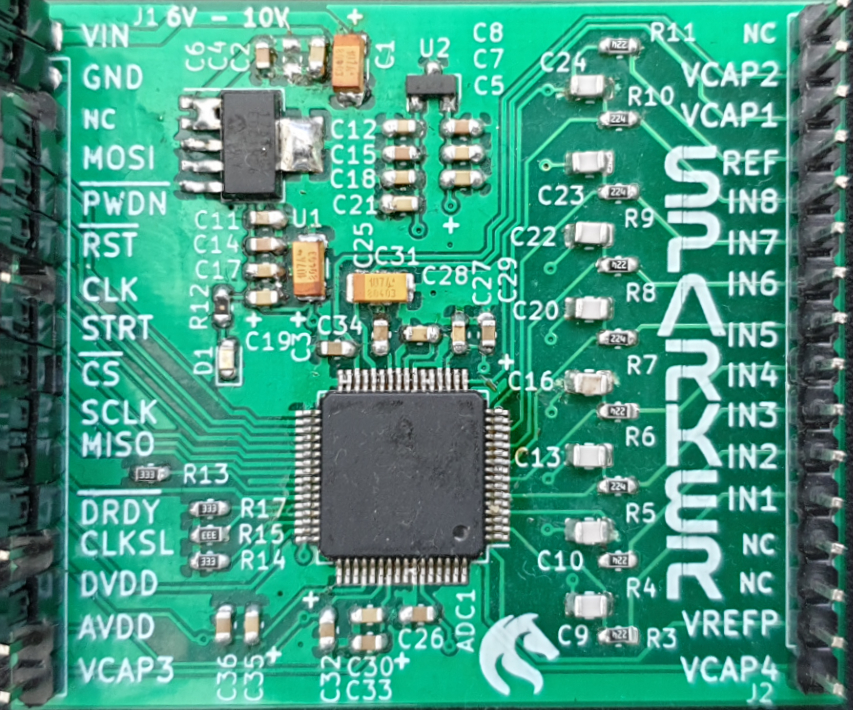
The program will continue to read from the buffer until there is a newline character. Once this has occurred, the string will be parsed to distinguish between the desired actuator and the position. The actuator string will be compared against the items of the list of known actuators. If a match is found, the duty cycle of the signal being passed to that actuator will be set to the required value. If no match is found, the command is ignored. This is the minimal software required to demonstrate basic functionality.



Figure 27 - A flowchart of the demonstration software.

## Modular Biosignal Front End

One of the aims of this project was to produce a hardware platform on which students can learn about biosignal processing, including EEG analysis, BCI devices, and electromyography (EMG). The ADS1299 is a chip capable of serving this purpose [44, 53, 54, 95]. To facilitate the inclusion of the ADS1299 in systems other than the data acquisition system designed for this project, and to facilitate prototyping and debugging of new systems, an ADS1299 breakout board has been designed and tested. A photo of the board is presented in Figure 28.

Figure 28 – A photo of the Modular Biosignal Front End

This board has two 2.54mm pitch headers to access important voltages, such as the VCAP voltages, the positive reference voltages, analog and digital supply voltages, and clock signals if desired. It also includes a 3.3V SPI interface for connection to any desired compatible MCU, opening the ADS1299 to a broad variety of applications. The board has input pins for all 8 channels, and includes a single reference pin. This configuration is ideal for referential montage EEG recording, however is applicable to a wide gamut of biosignal recording applications [96].

This board, in conjunction with the software library developed to configure and control the ADS1299 (see <APPENDIX> and <https://daq.netlify.app/>), has made the setup, configuration, and data acquisition process trivial, and it can be implemented in a matter of minutes. Therefore, the board can form part of a modular, educational biosignal analysis kit. Students can, for example, include their desired MCU, add additional low noise filter or amplifier stages in front of the analog input, and rapidly develop prototype biomedical devices.

Singh et al. conducted a study analysing the impact of the inclusion of 3 practical biomedical engineering courses in the mechanical engineering program at Widener University, USA [97]. This study concluded that the inclusion of practical, cross-disciplinary projects was considered favourably by participating students. All students felt that the project encouraged entrepreneurship. Overall, students agreed that doing their senior design project added to their understanding of the interdisciplinary and collaborative nature of the medical device innovation, encouraged their critical thinking skills, and added to their life-long learning skills [97].

Additionally, The University of Newcastle recently announced the commencement of their Bachelor of Medical Engineering (Honours) program [98]. This program enables students to select from one of four majors:

1. Medical Devices,
2. Medical Biomechanics,
3. Medical Signal Analysis, and
4. Medical Computing.

Students in the Medical Devices, or Medical Signal Analysis streams may be particularly interested in developing a biosignal analysis device. The modular system developed as part of this project greatly simplifies the process of designing such a device, enabling greater accessibility, and reducing the prior knowledge required. Dr. Shamus Smith (former Program Coordinator for Medical Engineering) of the University of Newcastle indicated in personal communication [99] that “often a barrier to applied work is the need to build the underlying hardware (which may be outside the skillet of the students or timeframe of the project.” Professor Rick Middleton, Head of the School of Electrical and Computer Engineering, indicated that should regulatory and safety concerns be addressed, there is potential value in exposing students to a modular system like the one developed for this project [100]. A mock MENG3800 Medical Engineering Research project description document detailing the set-up and operation of the modular biosignal front end can be found in <APPENDIX>.

# System Evaluation

Various testing phases were implemented to test the functionality of the system. Initially, tests focused on validating basic hardware functionality. Test tests were gradually built on until full system capability tests were conducted. Additionally, due to differences between the real-world signals and data published in [81], it was necessary to define a training paradigm to generate data used to train the SVM model.

## Testing and Evaluation Processes

### Functionality Testing of ADS1299

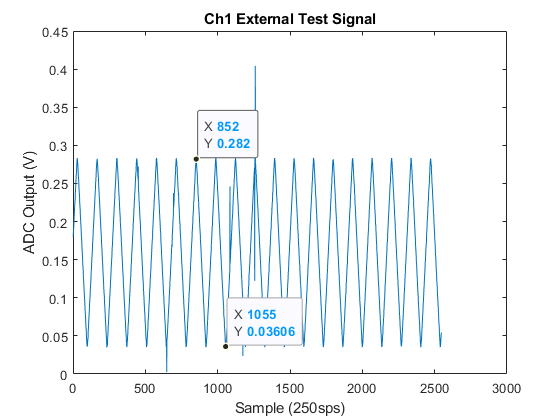
To test the functionality of the ADS1299, a 1.85Hz, 246mVpp triangle wave was applied with a DC offset of 159mV. This signal was generated using a function generator, and remains between 0.3V of the voltage supply rails, therefore within the linear range of the ADC [54]. The converted data was read continuously from the ADS1299 chip over the SPI interface. The data was then sent to a PC in real time using a serial link. The waveform was then plotted in Matlab. The result of this test is presented in Figure 29.

Figure 29 – A plot of the ADC output, converted to Volts, when a 1.85Hz, 246 mVpp triangle wave with a DC offset of 159mV was applied.

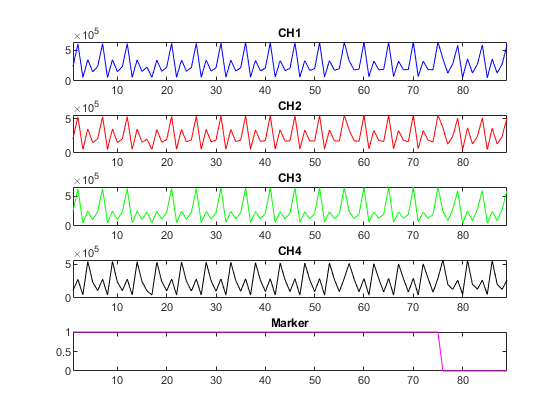
Successful completion of this test indicated that the ADS1299 circuitry was designed correctly, and the chip could be programmed and read from over the SPI interface.

### Establishment of the Bluetooth Link

The next test established a Bluetooth connection between the data acquisition board and the classification and control board. First, the two Bluetooth modules were configured appropriately. The Data Acquisition module was configured as a slave device, and its address was recorded. Configuring the devices as a slave enabled this system to connect to different devices (the arm, a laptop, or a smartphone). The Classification and Control Board was configured as a master, and was bound to the address of the slave, ensuring the two devices will automatically pair if both powered on and in range. The Data Acquisition System sent a stream of data in the approximate format of the recorded EEG data, and this data was read with 100% accuracy by the Classification and Control board. This test ensured the serial communication between the ATmega328P and its Bluetooth module, and the STM32F407VGT and its Bluetooth module, was set up correctly.

### EEG Data Acquisition using the ADS1299

Once the working condition of the ADS1299 had been established, real-world EEG data was recorded. This test established the working condition of the EEG electrodes, and identified that there were no impedance or scalp contact issues. To ensure the safety of the user, the power supply of the data acquisition system was a 9V battery, and the system communicated with the laptop over a Bluetooth link. This ensured total electrical isolation from mains power. A photo of the setup is presented in <FIGURE>, and a plot of the recorded waveform is presented in Figure 30. Completion of this test indicates that a wireless EEG data acquisition system has been successfully designed and implemented. This concluded evaluation of the data acquisition system.

Figure 30 – The time domain EEG signal recorded with the Data Acquisition System.

Computing the DCT of the recorded signal using Matlab produced the heatmap shown in Figure 31.

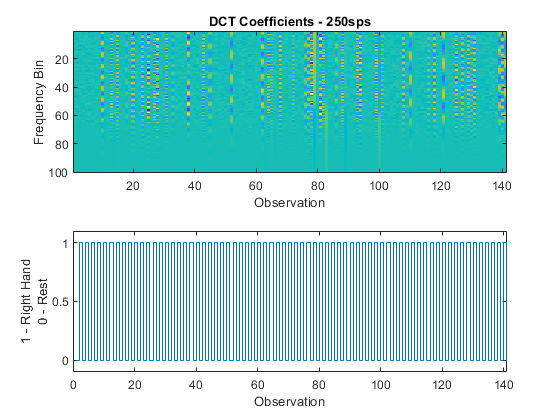


Figure 31 – (Top) A heatmap of the DCT coefficients of each observation. (Bottom) The cue given to the subject.

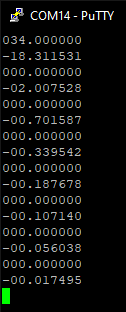
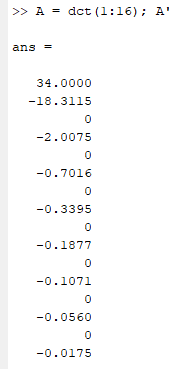
### Generation of the PWM Control Signal

It is critical that the classification and control board can generate the 6 required PWM signals simultaneously. This test was passed as 6 unique PWM signals were observed on the oscilloscope. These signals were then be used to control the prosthetic hand.

### Calculating the DCT

The classification and control board needs to calculate the DCT of a signal in real time. This includes multiplying the signal by a Hanning window function. The classification and control board was sent an epoch of EEG data from [80] over the Bluetooth link. Using the feature extraction module, the DCT of the signal was be calculated. The output vector was be compared to the output vector calculated using Matlab. As indicated in Figure 32, the output vectors matched identically. Therefore, this test was passed successfully.

Figure 32 – A comparison between the calculated DCT coefficients by (a) Matlab, and (b) the Classification and Control Board.



(a)

(b)

### Testing the SVM

Using a model trained on a PC, the SVM on the classification and control board was evaluated for classification accuracy. A time series of 601 observations from [80] was sent to the classification and control board using the serial link. DCT features were extracted from these observations, which were classified using the onboard SVM classifier, and the estimated data label was sent back over the serial link. The classification accuracy was then calculated as a percentage of correct classifications. The classifier accuracy was found to be 77.05%, compared to the Matlab model accuracy of 71.31%. Therefore, since the classification accuracy of the onboard classifier is within run-to-run variance of the PC-based model, this test was passed successfully. At the completion of this test, a successful offline BCI analysis system had been developed. This concluded evaluation of the classification and control board.

### Classifying Real Time Data

Once the Control and Classification board had been successfully evaluated, it was paired with the tested and working Data Acquisition board to classify real-time data. Once an epoch of data had been recorded by the data acquisition system and sent to the classification and control board, the data label was determined using the SVM, and the label was sent back to the data acquisition system, then to the PC where it was recorded. 68.75%

### All Together Now!

Once all the constituent parts have been tested and evaluated, and all the interfaces have been shown to operate successfully in real time, the whole system can be integrated. This will result in the acquisition and real-time classification of EEG brain waves, resulting in a 3D printed prosthetic hand executing the motions imagined by the subject.

## Generating Classifier Training Data

Due to differences in data acquisition systems and recording conditions, and biological differences between subjects, the data published in [81] was not be a suitable dataset to use for training the SVM classifier. Therefore, training data was generated using the system developed in this project.

### Training Data Acquisition Setup

Since training is a memory-intensive process [101], the training of the model is conducted using Matlab, on a PC. The serial link between the data acquisition system and the PC is well tested. When the Data Acquisition board is connected to the PC using a Mini-USB cable, Matlab can connect to the COM port and read the EEG data in real time.

### Training Paradigm

SVMs are supervised classifiers [102]. Therefore, when using test data to generate a classifier, the computer must know which class the data belongs to. This is achieved by issuing a prompt to the user, which is be stored with the corresponding EEG data (similar to the “Marker” channel in the data published in [81]). The training paradigm follows the timeline presented in Table 9 below.

Table 9 - Proposed classifier training data acquisition paradigm.

|  |  |
| --- | --- |
| Time (s) | Event |
| 0 - 1 | Prepare |
| 1 - 2 | Cue Given |
| 2 - 4 | Rest |

This training paradigm is similar to that presented in [81]. The first second consists of a “rest” period, where the subject will fixate on a cross at the centre of the screen. The preparation phase serves to reduce oculomotor artefacts by instructing the user to avoid eye movements, which may otherwise couple into the EEG signal [103]. The cue (“Open” or “Close”) will remain on the screen for 2 seconds. Following this, the user will be instructed to rest for a further 2 seconds.

The EEG data recorded from the COM port will be saved as a vector (vector length will be 4 seconds × 250 samples per second = 1000 samples long). This vector will form a component of a data structure, along with an integer encoding the state information, 1 representing hand closed and 0 representing hand open.

The recorded and processed data is then used to train a linear SVM model. The exact number of test datapoints required is not well defined. This value varies with the quality of EEG data. Poorer quality data contains more artefacts and EMI, reducing the class separation and thus reducing classification effectiveness. For this project, a typical recording session contained approximately 160 observations, with a holdout percentage of 20%.

### Training Results

It was found that EEG data recorded using the Data Acquisition System could be classified using the Matlab Linear SVM developed in Section 3.4 with an accuracy of 68.75%. While this is lower than the accuracy achieved using the same classifier and the data from [81], this result is higher than the accuracy achieved by Rodríguez-Bermúdez & García-Laencina in [58].

The reduced accuracy compared to the simulated classifier can be attributed to a variety of sources.

1. Reduced training dataset size. The classifier tested previously was trained using 611 observations, while the generated dataset consisted of only 161 observations. Due to the lack of BCI experience of the subject, they found it difficult to totally focus on modulating their neural activity for an extended period (longer than 30 minutes). This could be improved with practice.
2. Increased prevalence of EMI. The 50Hz component of the recorded EEG signals is much greater than the 60Hz component of the data from [81]. This may be due to the proximity of the recording equipment to 50Hz equipment (a computer, lighting, etc.). While visiting the University of Melbourne’s NeuroEngineering Lab, Prof. David Grayden (Clifford Chair in Neural Engineering) indicated that their laboratory mitigates these noise sources using an EMI protected recording room, with Faraday cage shielding, and DC lighting systems.
3. Electrode contact quality. While dry electrodes require less preparation than their wet counterparts (as highlighted in Section 2.3), the user manual for the FRI dry electrodes used in this project still recommend preparing the scalp with isopropyl alcohol. Mahmood et. al detail further procedures preparing the scalp contact areas with abrasive skin preparation gel, and removing residual dead skin cells with adhesive tape before placing dry electrodes [104]. Prof. Grayden indicated in personal communication that classifying data recorded using dry electrodes compared to wet electrodes can result in an accuracy decrease on the order of what has been observed in this project [105].

# Conclusion

At the completion of this semester, an extensive understanding of the current state of BCI research has been attained. Through this understanding, knowledge of the advantages and disadvantages of each approach has been gained. This knowledge, supplemented through a simulated comparative analysis, led to the development of the optimal system architecture. This system will use dry electrodes to measure neural activity during imagined grasping, then extract classification features using the DCT. The application of the DCT to EEG analysis is a novel approach. The feature vector will then be classified using a linear SVM, the output of which will be used to control a 3D printed prosthetic hand. The implementation of the system requires the design and construction of a wireless EEG data acquisition system, and a classification and control system, which is to be installed in the prosthetic hand. A clear path forward has been devised, to increase the probability of project success at the completion of next semester.

# Reference List

[1] L. Walters. (n.d., 3/11/2019). *Statistics on Hand and Arm Loss*. Available: http://www.aboutonehandtyping.com/statistics.html

[2] Physiopedia contributors. (2019, 3/11/2019). *Upper Limb Function in Spinal Cord Injury*. Available: https://www.physio-pedia.com/index.php?title=Upper\_Limb\_Function\_in\_Spinal\_Cord\_Injury&oldid=215844

[3] L. Poltawski *et al.*, "Assessing the impact of upper limb disability following stroke: a qualitative enquiry using internet-based personal accounts of stroke survivors," *Disability and Rehabilitation,* vol. 38, no. 10, pp. 945-951, 7 May 2016 2015.

[4] Disability Benefits Help. (2019, 3/11/2019). *How Disability is an Upper Extremity Amputation*. Available: https://www.disability-benefits-help.org/how-disabling-is-my-condition/upper-extremity-amputation

[5] The Christopher and Dana Reeve Foundation. (2013, 15/09/2019). *Prevalence of paralysis in the United States*. Available: https://www.christopherreeve.org/living-with-paralysis/stats-about-paralysis

[6] L. Garcia, V. Lespinet-Najib, M. Menoret, B. Claverie, J. M.André, and R. Ron-Angevin, "Brain–Computer Interface: Analysis of Different Virtual Keyboards for Improving Usability," *Neuroergonomics,* pp. 269-270, 30/11/2018 2018.

[7] The University of Pittsburgh. (2019, 03/11/2019). *Neuroprosthetics | Rehabilitation and Neural Engineering Laboratory | The University of Pittsburgh*. Available: http://www.rnel.pitt.edu/research/neuroprosthetics

[8] Amputee Coalition. (2019, 03/11/2019). *Promising Neuroprosthetic Research Projects to Keep an Eye On - Amputee Coalition*. Available: https://www.amputee-coalition.org/neuroprosthetic-research-projects-2019/

[9] A. P. Tsu, M. J. Burish, J. GodLove, and K. Ganguly, "Cortical Neuroprosthetics from a Clinical Perspective," *Neurobiology of Disease,* vol. 83, pp. 154-160, 25/09/2017 2017.

[10] S. M. Engdahl, B. P. Christie, B. Kelly, A. Davis, C. A. Chestek, and D. H. Gates, "Surveying the interest of individuals with upper limb loss in novel prosthetic control techniques," *Journal of NeuroEngineering and Rehabilitation,* vol. 12, no. 53, 13/08/2015 2015.

[11] M. Misaki, Y. Kim, P. A. Bandettini, and N. Kriegeskorte, "Comparison of multivariate classifiers and response normalizations for pattern-information fMRI," *NeuroImage,* vol. 53, no. 1, pp. 103-118, 15/10/2010 2010.

[12] A. B. Roy, D. Dey, D. Banerjee, and B. Mohanty, "Comparison of FFT, DCT, DWT, WHT Compression," *International Journal of Computer Applications* vol. 975, no. 8887, 2018.

[13] B. Szuflitowska and P. Orlowski, "Comparison of the EEG Signal Classifiers LDA, NBC and GNBC Based on Time-Frequency Features," *Pomiary Automatyka Robotyka,* vol. 21, pp. 39-45, June 2017 2017.

[14] B. Wang, C. M. Wong, F. Wan, P. U. Mak, P. I. Mak, and M. I. Vai, "Comparison of different classification methods for EEG-based brain computer interfaces: A case study," presented at the International Conference on Information and Automation, Zhuhai, China, 22/06/2009, 2009. Available: https://ieeexplore-ieee-org.ezproxy.newcastle.edu.au/stamp/stamp.jsp?tp=&arnumber=5205138

[15] W. B. Fye, "Julien Jean César Legallois," *Clinical Cardiology,* vol. 18, no. 10, 1995.

[16] The Virtual Brain. (n.d., 15/09/1009). *The Virtual Brain: Scalable Brain Simulation*. Available: https://www.thevirtualbrain.org/tvb/zwei/neuroscience-simulation

[17] A. Woodruff. (2019, 15/09/2019). *What is a neuron? = Queensland Brain Institute - University of Queensland*. Available: https://qbi.uq.edu.au/brain/brain-anatomy/what-neuron

[18] lumenlearning. (n.d., 15/09/2019). *Resting Membrane Potential | Biology for Majors II*. Available: https://courses.lumenlearning.com/wm-biology2/chapter/resting-membrane-potential/

[19] KhanAcademy. (n.d., 15/09/2019). *Membrane potential (resting membrane potential) (article) | Khan Academcy*. Available: https://www.khanacademy.org/science/biology/human-biology/neuron-nervous-system/a/the-membrane-potential

[20] Rice University. (n.d., 15/09/2019). *12.4 The Action Potential - Anatomy and Physiology*. Available: https://opentextbc.ca/anatomyandphysiology/chapter/12-4-the-action-potential/

[21] OperativeNeurosurgery. (2016, 15/09/2019). *motor\_cortex [Operative Neurosurgery]*. Available: https://operativeneurosurgery.com/doku.php?id=motor\_cortex

[22] Queensland Brain Institute. (2017, 15/09/2019). *Action potentials and synapses - Queensland Brain Institute - University of Queensland*. Available: https://qbi.uq.edu.au/brain-basics/brain/brain-physiology/action-potentials-and-synapses

[23] S. Schwerin. (2013, 15/09/2019). *The Anatomy of Movement - Brain Connection*. Available: https://brainconnection.brainhq.com/2013/03/05/the-anatomy-of-movement/

[24] J. Knierim. (n.d., 15/09/2019). *Motor Cortex (Section 3, Chapter 3) Neuroscience Online: An Electronic Textbook for the Neurosciences | Department of Neurobiology and Anatomy - The University of Texas Medical School at Houston*. Available: https://nba.uth.tmc.edu/neuroscience/m/s3/chapter03.html

[25] K. Nakayashiki, M. Saeki, Y. Takata, Y. Hayashi, and T. Kondo, "Modulation of event-related desynchronization during kinematic and kinetic hand movements," *Journal of NeuroEngineering and Rehabilitation,* vol. 11, no. 90, 30/05/2014 2014.

[26] G. Pfurtscheller. (2016, 15/09/2019). *EEG Event-Related Desynchronization (ERD) and Event-Related Synchronization (ERS) | Neuropsy Key*. Available: https://neupsykey.com/eeg-event-related-desynchronization-erd-and-event-related-synchronization-ers/

[27] E. Thomas, J. Fruitet, and M. Clerc, "Investigating brief motor imagery for an ERD/ERS based BCI," in *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, San Diego, USA, 2012, vol. 32.

[28] Z. Tang, S. Sun, S. Zhang, Y. Chen, C. Li, and S. Chen, "A Brain-Machine Interface Based on ERD/ERS for an Upper-Limb Exoskeleton Control," *Sensors,* vol. 16, no. 12, 02/12/2016 2016.

[29] A. Kubacki and D. Rybarczyk, "Development of a Speech System Using BCI Based on ERD/ERS for Patients Suffering from Amyotrophic Lateral Sclerosis," presented at the Advances in Manufacturing, 20, 2017. Available: https://link.springer.com/chapter/10.1007/978-3-319-68619-6\_27

[30] A. P. Burgess, "Towards a Unified Understanding of Event-Related Changes in the EEG: The Firefly Model of Synchronization through Cross-Frequency Phase Modulation," *PLoS ONE,* vol. 7, no. 9, 25/09/2012 2012.

[31] Natus Neuro. (n.d., 04/11/2019). *Grass Reusable Stamped-Cup EEG Electrodes | Natus*. Available: https://neuro.natus.com/products-services/grass-reusable-stamped-cup-eeg-electrodes

[32] Florida Research Instruments. (n.d., 04/11/2019). *New Longer 5mm Spike disposable/reuseable dry EEG Electrode TDE 210 - Florida Research Instruments Inc*. Available: https://www.fri-fl-shop.com/product/new-longer-5mm-spike-disposable-reusable-dry-eeg-electrode-tde-210/

[33] S. Hrishkesan. (2018, 16/09/2019). *Advantages, Disadvantages and Applications of EEG - Electronics and Communication Study Materials*. Available: https://www.electronicsandcommunications.com/2018/08/advantages-disadvantages-applications-of-eeg.html

[34] B. Burle, L. Spieser, C. Roger, L. Casini, T. Hasbroucq, and F. Vidal, "Spatial and temporal resolutions of EEG: Is it really black and white? A scalp current density view," *International Journal of Psychophysiology,* vol. 97, no. 3, pp. 210-220, September 2015 2015.

[35] R. Degabriele and J. Lagopoulos, "Techniques for effective EEG subject preparation," *Acta Neuropsychiatrica,* vol. 20, no. 4, pp. 218-219, August 2008 2008.

[36] M. A. Lopez-Gordo, D. Sanchez-Morillo, and F. P. Valle, "Dry EEG Electrodes," *Sensors,* vol. 14, no. 7, pp. 12647-12870, 18/07/2014 2014.

[37] J. W. Y. Kama *et al.*, "Systematic comparison between a wireless EEG system with dry electrodes and a wired EEG system with wet electrodes," *NeuroImage,* vol. 184, pp. 119-129, 01/01/2019 2019.

[38] K. E. Matthewson, T. J. Harrison, and S. A. Kizuk, "High and dry? Comparing active dry EEG electrodes to active and passive wet electrodes.," *Psychophysiology,* vol. 54, no. 1, pp. 74-82, Jan 2017 2017.

[39] Y. Wang, X. Gao, B. Hong, and S. Gao, "Practical Designs of Brain–Computer Interfaces Based on the Modulation of EEG Rhythms," in *Brain-Computer Interfaces*, B. Graimann, G. Pfurtscheller, and B. Allison, Eds. (The Frontiers Collection, Berlin, Heidelberg: Springer, 2010, pp. 137-154.

[40] P. Shenoy, K. J. Miller, J. G. Ojemann, and R. P. N. Rao, "Generalized Features for Electrocorticographic BCIs," *IEEE Transactions on Biomedical Engineering,* vol. 55, no. 1, pp. 273-280, Jan. 2008 2008.

[41] New York University. (n.d., 18/08/2019). *Electrocorticography (ECoG) | NYU Cognitive Neurophysiology Laboratory*. Available: https://med.nyu.edu/thesenlab/research-0/intracranial-eeg/

[42] B. Farnsworth. (2019, 25/09/2019). *EEG vs. MRI vs. fMRI – What are the Differences?* Available: https://imotions.com/blog/eeg-vs-mri-vs-fmri-differences/

[43] W. C. Chen, Y. C. Chen, C. C. Chou, and W. C. Fang, "An EEG analog front-end design with wireless communication module for a portable EEG monitoring system," presented at the 2015 IEEE 5th International Conference on Consumer Electronics, Berlin, Germany, 06/09/2015, 2015. Available: https://ieeexplore.ieee.org/abstract/document/7391321

[44] D. Acharya, A. Rani, and S. Agarwal, "EEG data acquisition circuit system Based on ADS1299EEG FE," presented at the 2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions), Noida, India, 2015. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7359346&isnumber=7359191

[45] Texas Instruments. (2012, 16/08/2019). *EEG Front-End Performance Demonstration Kit (B ed.)* [Datasheet]. Available: http://www.ti.com/lit/ug/slau443b/slau443b.pdf

[46] Maxim Integrated. (1996, 16/08/2019). *Circuit Tradeoffs Minimize Noise in Battery-Input Power Supplies*. Available: https://www.maximintegrated.com/en/app-notes/index.mvp/id/653

[47] X. Yang, J. Hu, Z. Chen, and H. Yang. (2017, 18/08/2019). *An 8-Channel General-Purpose Analog Front-End for Biopotential Signal Measurement*. Available: https://pdfs.semanticscholar.org/99e1/47917ea94e804a7c021bcacb8f58ed8dd3ed.pdf

[48] T. Uktveris and V. Jusas, "Development of a Modular Board for EEG Signal Acquisition," *Sensors,* vol. 18, no. 7, 03/07/2018 2018.

[49] S. Feng, M. Tang, F. Quivira, T. Dyson, F. Cuckov, and G. Schirner, "EEGu2: an embedded device for brain/body signal acquisition and processing," in *27th International Symposium on Rapid System Prototyping: Shortening the Path from Specification to Prototype*, Pittsburgh, Pennsylvania, 2016, pp. 19 - 25.

[50] B. Senevirathna et al., "Low cost mobile EEG for characterization of cortical auditory responses," presented at the IEEE International Symposium on Circuits and Systems (ISCAS), Montreal, Canada, 22/05/2016, 2016. Available: https://ieeexplore-ieee-org.ezproxy.newcastle.edu.au/document/7527437/authors#authors

[51] T. T. Vo, N. P. Nguyen, and T. V. Van, "WEEGEE: Wireless 8-Channel EEG Recording Device," presented at the 6th International Conference on the Development of Biomedical Engineering in Vietnam (BME6), 24/09/2017, 2017. Available: https://link-springer-com.ezproxy.newcastle.edu.au/chapter/10.1007%2F978-981-10-4361-1\_106

[52] M. Wild, R. Pegan, and M. Lara. (2015, 18/08/2019). *Wearable Bluetooth Brain-Computer Interface for Detection and Analysis of Ear-EEG Signals*. Available: https://pdfs.semanticscholar.org/bae8/2d84b4384bd86cf942139ac9a45326f3083d.pdf?\_ga=2.179954020.1547926640.1568593385-769052707.1568593385

[53] Texas Instruments. (n.d., 21/08/2019). *ADS1299 Low-Noise, 8-Channel, 24-Bit Analog-to-Digital Converter for Biopotential Measurements | TI.com*. Available: http://www.ti.com/product/ADS1299

[54] Texas Instruments, "ADS1299-x Low-Noise, 4-, 6-, 8-Channel, 24-Bit, Analog-to-Digital Converter for EEG and Biopotential Measurements," Jan. 2017 2017, Available: http://www.ti.com/lit/ds/symlink/ads1299.pdf.

[55] W. Klonowski, "Everything you wanted to ask about EEG but were afraid to get the right answer," *Nonlinear Biomedical Physics,* vol. 3, no. 1, 26/5/2009 2009.

[56] M. Pal, S. Bhattacharyya, A. Konar, D. N. Tibarewala, and R. Janarthanan, "Decoding of wrist and finger movement from electroencephalography signal," presented at the IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), Bangalore, India, 06/01/2014, 2014. Available: https://ieeexplore-ieee-org.ezproxy.newcastle.edu.au/stamp/stamp.jsp?tp=&arnumber=6740323

[57] S. S. Haykin, *Adaptive Filter Theory*, 4 ed. (Prentice-Hall information and system sciences series). Prentice Hall, 2002, p. 920.

[58] G. Rodríguez-Bermúdez and P. J. García-Laencina, "Automatic and Adaptive Classification of Electroencephalographic Signals for Brain Computer Interfaces," *Journal of Medical Systems,* vol. 36, pp. 51-63, 02/11/2012 2012.

[59] A. A. Suraj, M. Francis, T. S. Kavya, and T. M. Nirmal, "Discrete wavelet transform based image fusion and de-noising in FPGA,"

*Journal of Electrical Systems and Information Technology,* vol. 1, no. 1, pp. 72-81, May 2014 2014.

[60] J. Zhou, M. Meng, Y. Gao, Y. Ma, and Q. Zhang, "Classification of motor imagery eeg using wavelet envelope analysis and LSTM networks," presented at the Chinese Control And Decision Conference (CCDC), Shenyang, China, 09/06/2018, 2018. Available: https://ieeexplore-ieee-org.ezproxy.newcastle.edu.au/stamp/stamp.jsp?tp=&arnumber=8408108

[61] G. Huang, J. Meng, D. Zhang, and X. Zhu, "Window Function for EEG Power Density Estimation and Its Application in SSVEP Based BCIs," in *International Conference on Intelligent Robotics and Applications*, Aachen, Germany, 2011, vol. 7102, pp. 135-144: Springer, Berlin, Germany.

[62] D. Coyle, G. Prasad, and T. M. McGinnity, *A Time-Frequency Approach to Feature Extraction for a Brain-Computer Interface with a Comparative Analysis of Performance Measures,* vol. 19, pp. 3141-3151, 04/11/2004 2004.

[63] "DCT/IDCT Customer Tutorial," ed: Xilinx, 2000.

[64] W.-H. Chen and W. K. Pratt, "Scene Adaptive Coder," in *IEEE Transactions on Communications*, Dallas, 1984, vol. 32, pp. 225-323: IEEE.

[65] E. Roberts, "Lossy Data Compression: JPEG," ed: Stanford, n.d.

[66] J. Princen and A. Bradley, "Analysis/Synthesis filter bank design based on time domain aliasing cancellation," in *IEEE Transactions on Acoustics, Speech, and Signal Processing*, London, 1986, vol. 34, pp. 1153-1161: IEEE.

[67] B. Lincoln, "Modified Discrete Cosine Transform (MDCT)," ed: Stanford University, 1998.

[68] S. K. Bashar and M. I. H. Bhuiyan, "Classification of motor imagery movements using multivariate empirical mode decomposition and short time Fourier transform based hybrid method," *Engineering Science and Technology, an International Journal,* vol. 19, no. 3, pp. 1457-1464, Sept. 2016 2016.

[69] A. Bhardwaj, A. Gupta, P. Jain, A. Rani, and J. Yadav, "Classification of human emotions from EEG signals using SVM and LDA Classifiers," presented at the 2nd International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, Feb. 2015, 2015. Available: https://ieeexplore.ieee.org/document/7095376

[70] Y. Pang, S. Wang, and Y. Yuan, "Learning Regularized LDA by Clustering," in *IEEE Transactions on Neural Networks and Learning Systems*, 2014, vol. 25, no. 12, pp. 2191-2201: IEEE.

[71] F. Lauer. (2014, 21/08/2019). *Linear and Quadratic Discriminant Analysis (LDA / QDA)*. Available: http://mlweb.loria.fr/book/en/lda.html

[72] Penn State University. (n.d., 21/08/2019). *9.2.8 - Quadratic Discriminant Analysis (QDA)*. Available: https://onlinecourses.science.psu.edu/stat508/book/export/html/696

[73] V. Vapnik and C. Cortes, "Support-Vector Networks," *Machine Learning,* vol. 20, pp. 273-297, 1995.

[74] R. Pupale. (2018, 21/08/2019). *Support Vector Machines(SVM) -- An Overview - Towards Data Science*. Available: https://towardsdatascience.com/https-medium-com-pupalerushikesh-svm-f4b42800e989

[75] D. Sontage, "Support Vector Machines (SVMs) Lecture 2," ed: Massachusetts Institute of Technology, n.d.

[76] R. Berwick, "An Idiot’s guide to Support vector machines (SVMs)," ed: Massachusetts Institute of Technology, n.d.

[77] K. Teknomo. (n.d., 21/08/2019). *K Nearest Neighbors Tutorial: How K-Nearest Neighbour (KNN) Algorithm works?* Available: https://people.revoledu.com/kardi/tutorial/KNN/HowTo\_KNN.html

[78] (n.d., 21/08/2019). *kNN*. Available: http://cs.carleton.edu/cs\_comps/0910/netflixprize/final\_results/knn/index.html

[79] (2018, 21/09/2019). *Lecture 2: k-nearest neighbors / Curese of Dimensionality*. Available: http://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote02\_kNN.html

[80] B. M. Kaya M, Ozbay E, Yanar H, Mishchenko Y, "A large electroencephalographic motor imagery dataset for electroencephalographic brain computer interfaces," ed: figshare, 2018.

[81] B. M. Kaya M, Ozbay E, Yanar H, Mishchenko Y, "A large electroencephalographic motor imagery dataset for electroencephalographic brain computer interfaces," *Scientific Data,* vol. 5, no. 180211, 2018.

[82] K. Cabeen and P. Gent. (n.d., 03/11/2019). *Image Compression and the Discrete Cosine Transform*. Available: https://www.math.cuhk.edu.hk/~lmlui/dct.pdf

[83] A. B. Watson, "Image Compression Using the Discrete Cosine Transform," *Mathematica,* vol. 4, no. 1, pp. 81-88, 1994.

[84] Florida Research Instruments. (n.d., 04/11/2019). *EEG Headband Packages - Florida Research Instruments Inc*. Available: https://www.fri-fl-shop.com/product/eeg-headband-packages/

[85] Microchip, "Analog and Interface Guidew," vol. 1, n.d.

[86] ST Electronics, "AN5043: Layout recommendations for the design of boards with ST25R3911B/ST25R391x devices," January 2018 2018.

[87] E. Vita, "Arduino Nano - Rev 3.2," 30 December 2014 2014.

[88] Atmel, "AVR910: In-System Programming Application Note," 11/2016 2016.

[89] S. Thornton. (2016, 04/11/2019). *Trade-offs in Choosing 8-bit vs. 16- and 32-bit Architectures*. Available: https://www.microcontrollertips.com/trade-offs-choosing-8-bit-vs-16-32-bit-architectures/

[90] ST Electronics, "UM1472: Discovery kit with STM32F407VG MCU User manual," May 2017 2017.

[91] ST Electronics, "STM32F407xx Datasheet," September 2016 2016.

[92] ST Electronics, "UM1725: Description of STM32F4 HAL and LL drivers," User Manual February 2017 2017.

[93] S. Sayad, "Support Vector Machine," n.d. n.d.

[94] A. Mota. (2017, 04/11/2019). *Materials of Prosthetic Limbs*. Available: https://broncoscholar.library.cpp.edu/bitstream/handle/10211.3/193171/MotaAnissa\_LibraryResearchPaper2017.pdf?sequence=1

[95] M. B. Pires, J. J. A. Mendes, and S. L. Stevan, "Development of an 8 channel sEMG wireless device based on ADS1299 with Virtual Instrumentation," August 2018 2018.

[96] M. Z. Jamal, "Signal Acquisition Using Surface EMG and Circuit Design Considerations for Robotic Prosthesis," in *Computational Intelligence in Electromyography Analysis - A Perspective on Current Applications and Future Challenges*, G. R. Naik, Ed. London: IntechOpen, 2012.

[97] A. Singh, D. Ferry, and S. Mills, "Improving Biomedical Engineering Education Through Continuity in Adaptive, Experiential, and Interdisciplinary Learning Environments," *Journal of Biomechanical Engineering,* vol. 140, no. 8, August 2018 2018.

[98] The University of Newcastle. (2017, 7 June 2020). *Medical Engineering Degree Launch / Faculty of Engineering and Built Environment*. Available: https://www.newcastle.edu.au/events/faculty-of-engineering-and-built-environment/medical-engineering-degree-launch

[99] S. Smith, "An educational modular biosignal front end," S. Smith, Ed., ed, 2020.

[100] R. Middleton, "An educatrional modular biosignal acquisition system," S. Parker, Ed., Email ed, 2020.

[101] J. Nalepa and M. Kawulok, "Selecting training sets for support vector machines: a review," *Artificial Intelligence Review,* vol. 52, no. 2, pp. 857-900, August 2019 2018.

[102] S. Winters-Hill and S. Merat, "SVM clustering," *BMC Bioinformatics,* vol. 8, no. 7, 1/11/2007 2007.

[103] H. Nakatani, N. Orlandi, and C. van Leeuwen, "Precisely timed occulomotor and parietal EEG activity in perceptual switching," *Cognitive Neurodynamics,* vol. 5, no. 4, pp. 399 - 409, 5/11/2011 2011.

[104] M. Mahmood *et al.*, "Fully portable and wireless universal brain–machine interfaces enabled by flexible scalp electronics and deep learning algorithm," *Nat Mach Intell,* vol. 1, no. 9, pp. 412-422, 20 March 2019 2019.

[105] D. Grayden, "EMI shielding for non-invasive BCI systems," D. Grayden, Ed., ed, 2020.

# Appendix A – Data Acquisition System Schematic





# Appendix B – Data Acquisition System Gerber Files



Top Layer – Signal with Ground Pour



Second Layer – Ground Plane



Third Layer – Power Pours

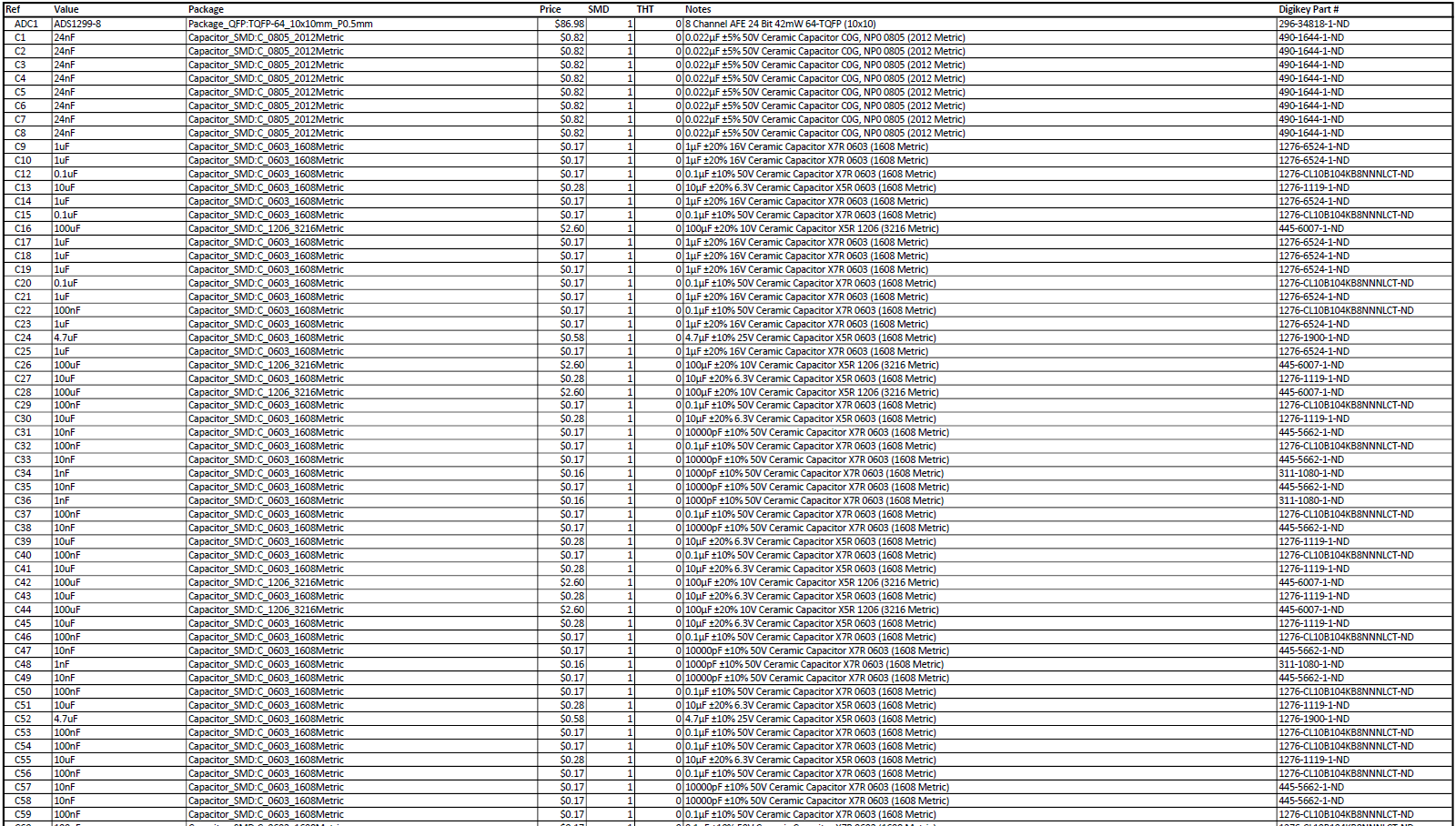


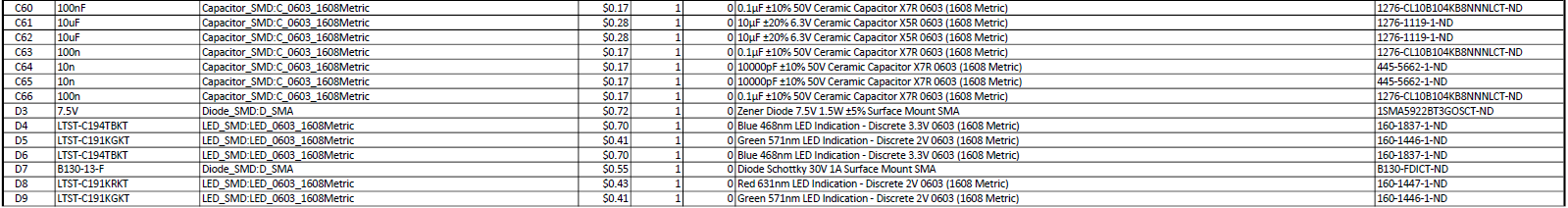
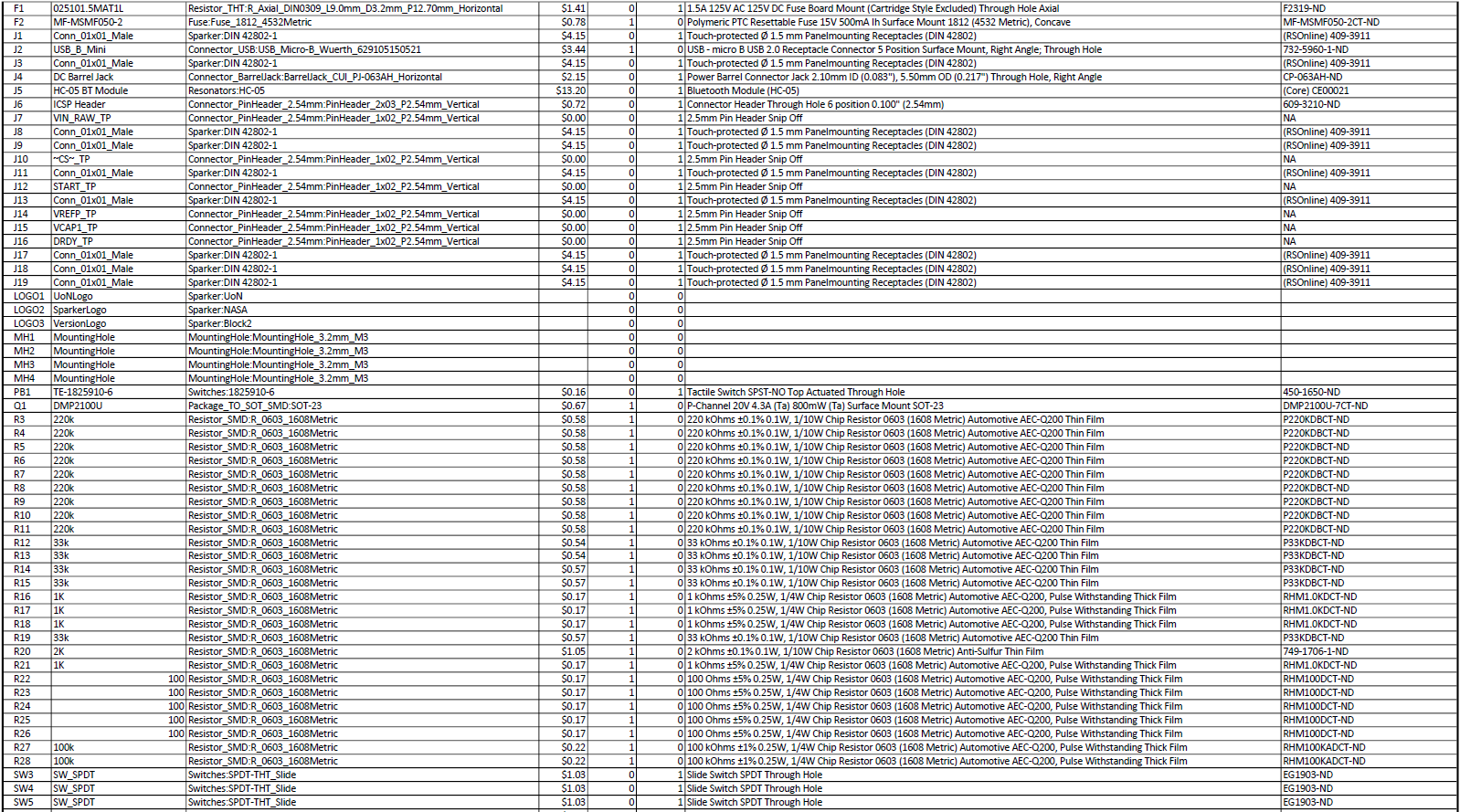
Bottom Layer – Signal with Ground Pour

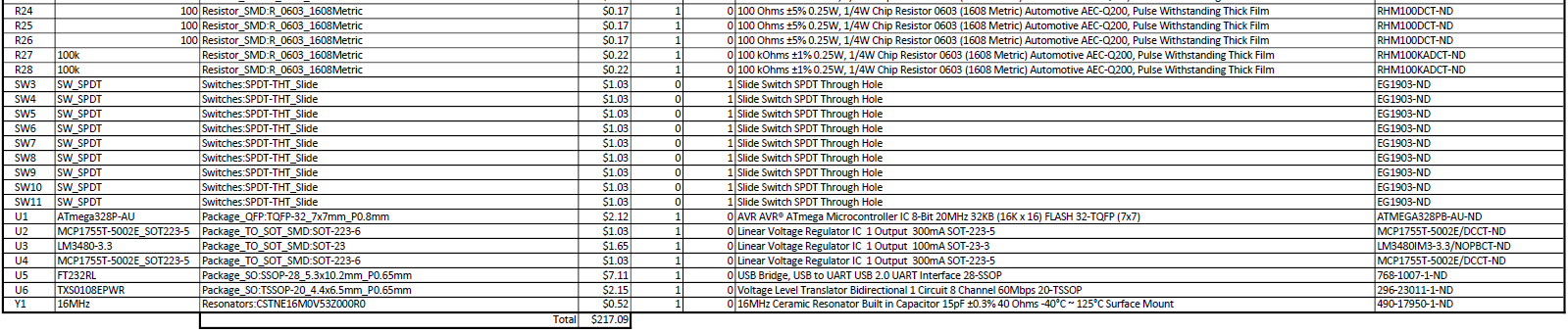


Fabrication Layer

# Appendix C – Data Acquisition Board Bill of Materials







# Appendix D – Mock MENG3800 Project Description (Based on 2018 ENGG3850 Project List [106])

**6 Accessible Computer Input Device (3 groups max)**

The goal of this project is to develop an accessible computer input device for a person living with tetraplegia following cervical spinal fracture. The designed solution will enable the user to control a computer cursor, and make keyboard inputs, without the use of their extremities beyond muscle flexion of the biceps and triceps. It is required that the system operate in real time. Features to be considered in the design of the accessible computer input device include, but should not be limited to:

* Recording and transmission of the desired biosignal,
* Recording and classification of training dataset(s),
* Design and implementation of the data acquisition system, and
* Design and implementation of the GUI (keyboard, cursor).

Further extensions could include:

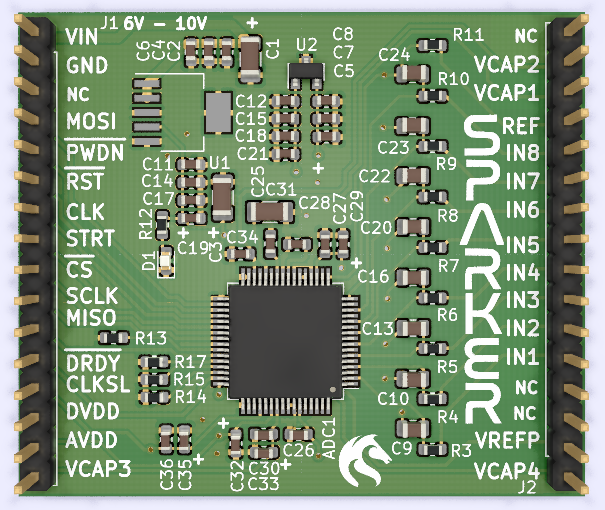
* HID standard encoding, enabling use with Windows,
* Cortical control (using EEG rather than EMG), or
* A means of recalibrating the device while in use.

The project can be built on the existing Modular Biosignal Front End, and its associated software library. Configuration instructions to enable basic recording can be found attached.

NOT FOR RELEASE

1. **Basic Hardware Setup**

NOT FOR RELEASE

Using the Modular Biosignal Front End:

1. Connect the GND pin to the Ground of the supply voltage. Earth reference the supply.
2. Jump the , , and CLK pins to DVDD.
3. Connect the pin to a **3.3V** GPIO Pin on a microcontroller.
4. Connect the SPI bus (MOSI, MISO, SCLK) to their respective **3.3V** SPI pins on the microcontroller.
5. Connect the VIN pin to a DC voltage source (between 6V and 10V).
6. Power on the device. D1 should illuminate.
7. After a moment, the voltage on VCAP1 should exceed 1.1V.
8. Once VCAP 1.1V, should toggle at 250Hz. The hardware is now ready.
9. **Basic Software Setup**

NOT FOR RELEASE

Using the ADS1299 library (full usage notes at [daq.netlify.app](http://daq.netlify.app)):

1. Set up the SPI interface (1MHz SCLK, Clock Polarity = 0, Clock Phase = 1). Data is sent MSB first. Set the CS pin as output.
2. Reset the device (ADS1299::RESET()).
3. Stop continuous data conversion (ADS1299::SDATAC()).
4. Configure the device as desired, e.g.

ADS1299::setGain(PGA24);

ADS1299::InternalReferenceOn(true);

ADS1299::setDataRate(250SPS);

ADS1299::AllChannelSRB1Short(true);

ADS1299::setElectrodeConnectionType(CH1, NORMAL);

1. Re-enable continuous data read (ADS1299::RDATAC()).
2. The ADS1299 will now take the pin LOW when data is ready to be read.

Once the hardware and software has been configured following these instructions, the device is completely set up to facilitate continuous data reading, no additional configuration is required.

NOT FOR RELEASE