Smart Hub Controlled Using EEG Technology

A Project Report
Presented to
The Faculty of the Computer Engineering Department
San Jose State University
In Partial Fulfillment
Of the Requirements for the Degree
Bachelor of Science in Computer Engineering

By
Mohdi Habibi
Amir Jabbari
Taylor Keairns
Sara Sepasian
12/2017

APPROVED FOR THE COLLEGE OF ENGINEERING

Preet Kang, Project Advisor

Professor Rod Fatoohi, Instructor

Dr. Xiao Su, Computer Engineering Department Chair

ABSTRACT

Smart Hub Controlled Using EEG Technology

By Mohdi Habibi, Amir Jabbari, Taylor Keairns, Sara Sepasian

Smart Hub Controlled Using EEG Technology is an application of Electroencephalography (EEG) technology; smart EEG interfaces use the brain's electrical impulses to communicate with an array of smart devices. This technological feat is the next step to making appliances universally accessible. EEG technology allows a deeper understanding into the human mind by amplifying and recording human neural brain signals. Smart devices, in turn, allow humans to seamlessly interface with appliances, applications, and other devices. Currently, leading technology companies—such as Netflix acting under the name MindFlix—are investigating EEG technology as a means of enhancing user experience; however, this technology lays on the cutting edge of modern science, and has yet to be fully actualized by industry. One of the goals is to simply prove the feasibility in utilizing these technologies, and to further advance the research and implementations done thus far.

Currently, human to machine interaction is limited to either tactile or speech interfaces—e.g., keyboards, touchscreens, and Apple's Siri. In the past decade, systems using speech to text have been adopted by most tech companies, and they have been integrated into a wide range of products. The vast shortcomings of speech to text are apparent; these shortcomings include miscommunication or misinterpretation, time delay, and general inconvenience. As smart

appliances become more widespread and feature sets soar, ensuring that these technologies are accessible for all users increases in importance. The next logical step in this technological evolutionary process is to enable humans to control machines with their minds, and have machines be seamless extensions of the human nervous system.

This project employs cutting edge EEG technology that will enable users to interface with smart devices just by thinking or performing a set of discrete triggers. This project's implementation of these technologies will include a mobile application, wherein the user will be able to perform more complex actions from a relatively small set of EEG commands. The fruitions of this project will be multiple smart appliances made universally accessible using EEG technology, and a user interface where the data can be seen in real time along with analytics.

\

Acknowledgments

Much thanks to our advisor Preet Kang for all his help.

Table of Contents

List of Figures

List of Tables

Chapter 1. Introduction

1.1 Project Goals and Objectives

The primary objective of this project is to use EEG technology to establish an interface between a user's brain and smart devices. EEG technology is primarily used in healthcare applications to record a patient's brainwave activity. The starting point for our project begins a thorough investigation into neural signals, their patterns, and the methods of interpreting this data. Then procuring, designing, developing, and testing the necessary hardware to capture and filter the data. Once data is captured by the hardware, the signals will then be processed through various machine learning algorithms to further discretize patterns in users' neural activity. Finally, these discretized signals will be translated into actionable commands which the user can view through a mobile application. All of the hardware is tested for reliability and accuracy. Each piece of code is unit tested, and integration tested through various scripts. The result will be an accessible and robust headset that will allow users to train machine learning models and issue commands from the headset without any prerequisite knowledge.

1.2 **Problem and Motivation**

This project's motivation is not so much driven by an immediate problem, but rather by an opportunity. By creating seamless interactions between the brain and a device, not only are we redefining accessibility standards but also ushering a new age of social equality for disabled people. Social justice and accessibility often come hand in hand, which is why we are positioned at an important junction to forever alter the landscape of technology. Our project will allow all disabled people the same harmonious interaction with technology which others take for granted. This project will also increase workplace efficiency, and future iterations of this project will provide a platform for other developers to generate software to further advance these goals. By creating a means for the disabled or physically diminished to integrate themselves with devices, we are creating a truly innovative product that will redefine accessibility for generations to come.

1.3 Project Application and Impact

A project with as wide a scope as this one will have definitive long-term impacts on society. By allowing users to communicate with smart devices through neural signals, this project is ensuring universal accessibility. Modern devices thus far are catered to the physically apt or able, which make it difficult for individuals with disabilities or other shortcomings to utilize these helpful devices. The lack of accessibility is primarily due to the market driven nature of technology. By implementing a device that can interface with any individual—regardless of physical ability, creed, sex, or race, numerous existing product lines can be utilized by the masses. By allowing users that speak different languages to communicate with smart devices, manufacturers are exposed to a greater market, and they have a lower overhead of product localization.

Academically, this project will expose students and faculty to the possibilities of EEG technology and allow them to further propagate new and existing technologies. Throughout our meticulous research on the methods and capabilities of current EEG technologies we have found that there is an immense opportunity for further research in this field. This project, being a testament to the capabilities of EEG technology will hopefully work to further the advancement of this research.

Regarding the EEG industry, this project will expand the use of EEG technology and enable EEG device manufacturers to explore the possibilities of EEG technologies in fields other than medicine. As EEG devices proliferate, the EEG industry is sure to witness an influx of revenue and attention from many of the major players in the tech industry. By creating a platform through which many tech companies can integrate smart devices, we have provided a crucial stimulus in the rise of EEG industry.

On the societal scale, this project will begin the discussion of bringing universal accessibility to smart devices. By creating a product that changes the way humans communicate with technology, we are part of the third wave of human computer interactions, prefaced only by touch and voice. Such a device could change how society perceives and interacts with technology; by providing an unparalleled level of accessibility to the technology which we use every day. However, implications and other far-reaching effects of this project cannot be completely projected, but the potential that lays in this technology is unquestionable.

1.4 Project Results and Deliverables

There will be several items as part of our deliverables for this project. A custom designed PCB that provides the necessary functionality, a 3D printed helmet that can house the PCB board and the electrodes, and a host of software that provides a wide range of features.

The hardware provides the ability to extract the neuroelectrophysiology information from a user's head through analog to digital conversion from the EEG electrodes, along with a band pass filter to ensure that the signals that we are receiving are within those emitted by neurons. This conversion board communicates with another standard module of higher processing power where the machine learning will take place, and wherein the machine learning model will be stored to issue the commands.

The helmet plays an essential role here as well; different parts of the brain output different electrical impulses, and appropriate electrode placement is essential for precise data collection.

As part of our software deliverables, there are be several libraries that are written to support the communication between the EEG conversion module, the processing module and the user application. The overall code structure and execution is controlled by a sequencer and a corresponding sequence tester. The sequence hosts a state machine that keeps track of the user's current state and runs the appropriate scripts to facilitate user commands. Likewise, a corresponding test sequencer can be ran to ensure that all parts of the state machine are able to work together by mimicking state changes. The scripts executed by the sequencer include, a script that interfaces the processing module on which the sequencer runs from the EEG conversion module, a script that trains the model on user command, and a third script that receives data in real time from the EEG conversion module and runs it through the saved machine learning model for classification. Another deliverable is a mobile application that allows user to interface with the headset, train models and issue commands. Finally, the result will be a lightweight noninvasive headset and companion mobile application that will enable user to issue discrete commands.

1.5 Project Report Structure

The rest of this paper consists of a summary of the background of EEG technology, starting from medical use all the way to experimental consumer applications. This will be followed by an overview of the project requirements entailing the specific goals and objectives that this project will complete. The project requirements are followed by the project design, which contains the specifics of the design process in both software and hardware. Once the design philosophy is described, the specifics of the implementation are detailed in the following section. Following this section, is a comprehensive list of the tools and standards used. Lastly, the testing and experimental procedures of the final product will be detailed.

Chapter 2 Background and Related Work

2.1 Background and Used Technologies

Electroencephalography (EEG) can be used to analyze the neurophysiological phenomena of an individual. Current applications of EEG technologies are typically found in medical industries, but this technology can also be used as a mechanism to control external devices. EEG technology can be utilized to generate a so-called Brain Controlled Interface, or BCI. Additionally, this could be used for controlling or integrating an individual with a virtual environment. The brain is an individual's focal point for informational processing and controlling the other mechanisms of the body. Visual and kinetic processing is computed within the brain, and they are revealed as electrical impulses upon measurement. Many of the other physical phenomena enacted by the brain, e.g. muscle contraction, can be more easily measured than EEG. Electromyography (EMG) and Electrocardiography (ECG) are other commonplace technologies that measure other electrical impulses within the human body, and give a wealth of information regarding an individual's health. Researches primarily within academia have sought numerous methods that may offer a look inside an individual's mind. Recently, functional Magnetic Resonance Imaging (fMRI) technologies have allowed researchers at UC Berkeley to project imagery from a user's mind.

Human bodies provide a set of sensory inputs to the brain; these sensory inputs include vision, sound, balance, touch, smell, taste, hunger or thirst, spatial, and more. Each of these sensory inputs are processed by the mind, and all contribute to an individual's interaction with their environment. EEG data consists of small amplitudes within the microvolt range (μV)—which speaks to each signal's sensitivity and susceptibility to noise. Large, low frequency electromagnetic fields, radio waves, and many others greatly contribute toward added noise whilst measuring EEG data. Due to the amount of noise present, it is tantamount that the components that are utilized within this project contain the features required to reliably and accurately extract EEG data.

Various technologies and programming methodologies are used in the development of this project. Programming languages used, include C, Python, NodeJS, Swift. C was used in the development of the drivers to communicate with the EEG processing modules which are directly connected to the EEG boards. To assure the fastest data rate possible SPI drivers were written to communicate with these modules. Corresponding, unit test were also written in C. Python is used to train the machine learning model using Python's robust Scikit-Learn machine learning library which provides implementations for many of the most commonly used machine learning algorithms and corresponding benchmarks for use in testing. NodeJS was used for the overall sequencing engine because of its robust ability to interface with web applications and databases such as Firebase, the one used in this project. The NodeJS sequencing engine is in charge of the state machine that the user progresses through the mobile application. NodeJS makes heavy use of asynchronous programming methodologies through callback functions which allow us, for example, to run multiple scripts in other threads in parallel without having our state machine have to stop and wait for the completion of that script. This concept is used heavily in the sequencing engine where multi-threading calls are made to the communication script and the machine learning training and execution scripts.

Even though this project is a result of all the courses taken at SJSU, and the overall inquisitive nature and drive of that SJSU fosters, there are some courses that resulted directly in portions of this project; those are listed in Table 1.

Table 1. List of courses and their applications to this project

Course	Application
CMPE 188 - Machine Learning	Training machine learning model and classifying EEG data in real time.
CMPE 172 - Enterprise Applications	NodeJS scripting and online database interfacing.

CMPE 146 - Embedded Programming	Interfacing with EEG electrode communication modules, include ADS 1299.
CMPE 124 - Digital Design	Creating PCB design.
CMPE 110 - Circuit Analysis	In PCB layout and overall PCB design to create basic communication between the chips.

2.2 Literature Search

In 1875, a scientist in the name of Richard Caton reported in the British Medical Journal that animals with exposed cerebral hemispheres present electrical phenomena, but it was until 1924 when Hans Berger recorded the first electroencephalogram (EEG) signals from humans. An oscillatory activity in the brain was identified by Berger when analyzing EEG traces. He was able to identify the Berger's wave (8– 12 Hz), also known as alpha wave. In 1970, the field of brain computer interface (BCI) was initiated, and it mainly targeted neuroprosthetics applications such as restoring damaged movement, hearing and sight. In the mid-1990s and after experimenting on animals, appeared the first neuroprosthetic devices implanted in humans. Low-cost BCI-based interfaces for the gaming industry and relaxation applications were introduced into the market in 2006 by Sony. In 2007, NeuroSky released the first dry sensor technology as a consumer based EEG. Also, a device for video games that use EEG was developed by OCZ Technology in 2008. In 2009, Mattel and NeuroSky released Mindflex which is a steering ball through an obstacle course game. Around the same period, Emotiv Systems released EPOC, a 14 channels EEG device that can detect thirteen conscious states and four mental states. In 2009, intendiX was released into the market. Using intendiX, a user can trigger an alarm, type on a keyboard matrix, and copy text into an e-mail. In 2012, Neurowear produced Necomimi, a cat-like ears that are controlled by NeuroSky, a brain-wave reader. In the same year, 2012, g.tec presented Screen Overlay Control Interface (SOCI), a new intendiX module. SOCI is capable of allowing users to play several games using their mind with an accuracy of 99% for detecting different brain signals. Nowadays EEG signal acquisition devices are available at low cost (~$200), which has contributed to a wide spread of such devices which have been used for various mental tasks. Depending on the frequency, waves are classified into different types. Each wave type consists of carries information that provides a conveys the current state of the brain.

2.3 State-of-the-art Summary

Potential breakthroughs in EEG-related markets are present and more are soon to come. Advancements in material technologies allow EEG sensors to gain in precision, and greater understanding of physics allows for lower-noise electrical circuits. These two components are the basis for measuring the neural activity of an individual's brain. Neuroelectrophysiology and our increased general understanding of physiology demystifies the role of the brain in the body.

Another important aspect of EEG devices is that they are cheaper and more readily-available than previous decades, and although these devices are cheaper, they still maintain medical precision. This gives more tools at the disposal of hobbyists, researchers, or others interested in neuroelectrophysiology. EEG sensors comprised of various materials expand the number of possible applications—especially dry touch sensors that do not require special liquid solutions. Low-noise Analog-to-Digital converters that interface with EEG sensors are specially designed by Texas Instruments. TI's ADS1299 and the rest of the ADS129x series offer wide ranges of functionality, and they help a designer streamline the microcontroller design process for embedded applications. On the software level, companies like EMOTIV have designed advanced programs that can read in and interpret EEG data with machine learning. These software and hardware advancements lay the foundation for future commercial breakthroughs with this technology.

Many scientists and engineers believe human integration with technology will be the next great advancement of the human race. Speculation is also made by researchers in Artificial Intelligence where computer systems and algorithms are designed to mimic the neural pathways of the human brain. Others also suggest the possibility of individuals 'uploading' their conscious minds to machines, following the advent of conscious artificial intelligence. Greater understanding of EEG, consciousness, neuroelectrophysiology and its related fields can only assist in making these endeavors a reality.

Chapter 3: Project Requirements

3.1 Domain and Business Requirements

Implementing a business model around a cutting-edge technology presents a number of challenges. One of the primary business requirements would be to bring on skilled businessperson who are capable of expressing both the technical intricacies of our project and usability of the product to the everyday consumer. Additionally, the companies and employees chosen to collaborate with, must be selected deliberately to ensure a quality high-end product. Because our product relies on and is in the domain of new cutting-edge technologies, it's success is dependent on quality components, informed product development, and security in protecting the neurological information of users.

3.2 System (or Component) Functional Requirements

Functional requirements of this project include the design of the Printed Circuit Board (PCB), the firmware code that is written to the microcontroller to be able to communicate with the Analog to Digital Converter (ADC), flash memory, as well as the Bluetooth (BT or BLE) interface. The microcontroller is the center of all data processing.

The firmware written on the microcontroller should be able to sample data from the ADC as data becomes available. This data should then be processed by a machine learning model that is continuously running and being fed the new data. The machine learning algorithms will filter out any discrepancies in the users thought pattern in order to distinguish command signals from ordinary brain activities, and then it interprets the data into discrete commands that can be sent to the smart devices.

3.3 Non-functional Requirements

Non-functional requirements include many human factors as well as reliability, robustness, and security of hardware and software. The headset shall sit securely on the user's head such that the electrodes are touching the scalp. The headset shall also allow for user interaction in the form of on off button. The headset shall be adaptable to a variety of head shapes and be comfortable. The PCB shall fit in the headset and shall process the EEG data reliably. The connection from the headset to the smart device should be reliable such that commands are not lost, and such that necessary indication is sent to the user if this connection is not complete. The on-board memory shall have enough capacity to hold enough data for at least two commands that shall be sent from the headset to the smart device. The transfer of data from the headset to the smart device shall take less than 5 seconds from user neural signals to smart device reaction.

3.4 Context and Interface Requirements

Since the technology that we are working on is in its relative infancy, context environments are only valid in the case of the products that we used in order to achieve our final product. The product has been designed in such a way that the hardware is robust, meaning that any change in hardware due to supply chain issues can be remedied easily with the refactoring of some base firmware code thereby leaving the abstract classes the same.

3.5 Technology and Resource Requirements

Since we have chosen to make most components of this project from scratch our context requirements are low. The parts that we have used including the ADS1299, flash memory, and microcontroller can be interchanged and are not necessarily specific. Although, though the implementation details highlight the importance of correctly communicating analog data is serialized at a high rate, therefore requiring a reasonably fast microprocessor, as well as an ADC with enough precision to be able to distinguish commands. The IDE used to develop the software is a standard eclipse version that can be interchanged with any basic C compiler. Additionally, Python provides the necessary portability to alternative computing devices. The PCB was created on Eagle Design Tool, and once the Gerber or schematic file of the PCB is made, almost all PCB manufacturers will be able to print the PCB. PCB assembly is the next requirement, as many of the components utilized are small and sensitive, and as a result require precise placement by machines.

Chapter 4 System Design

4.1 Architecture Design

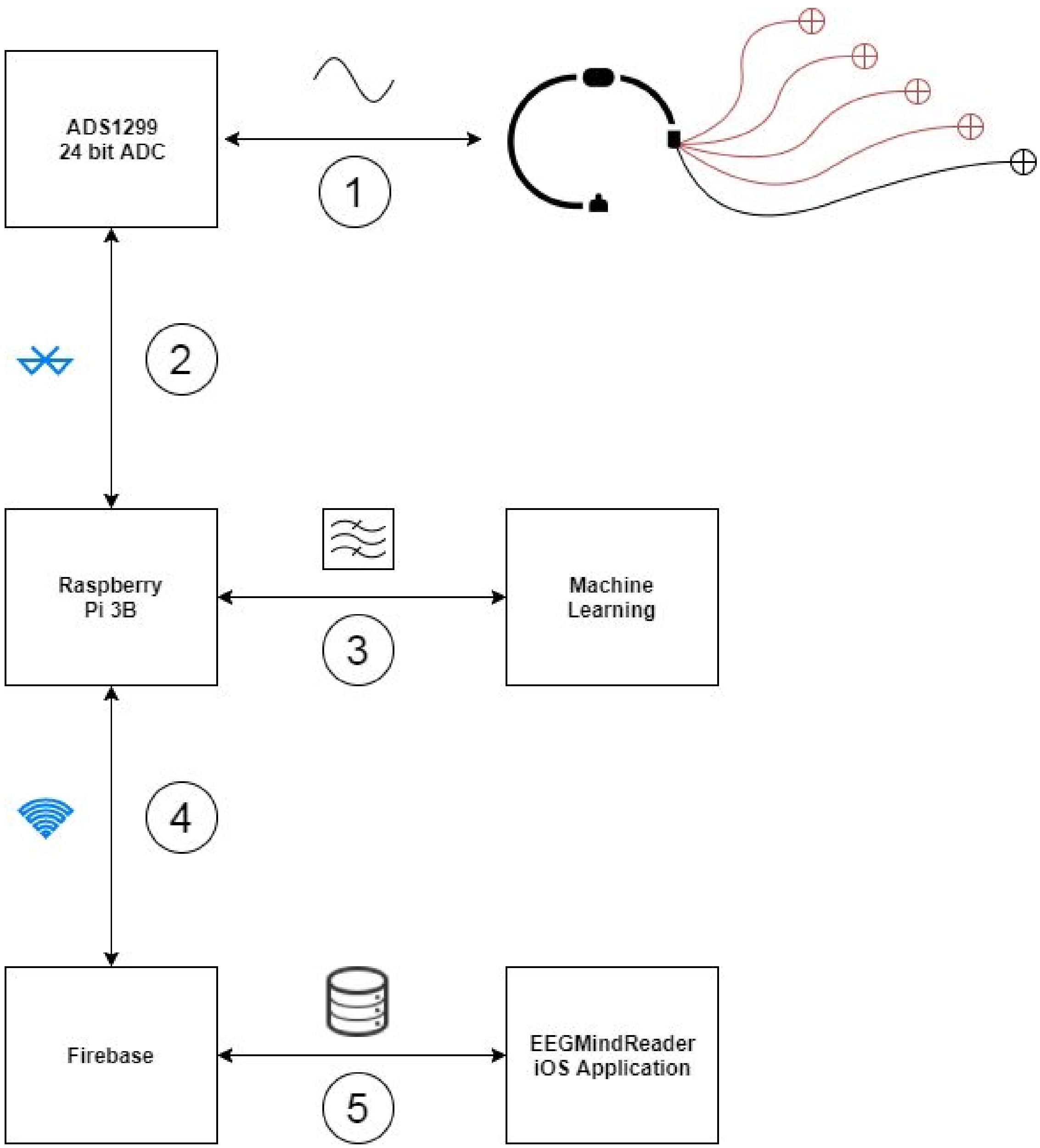


Figure 1. System Architecture and Data Flow Diagram

Figure 1 depicts the overall system architecture and the data path from user to application. The steps noted within Figure 1 occur sequentially, and are described in the following list:

1. Raw EEG data is read from the electrodes over a wire in the form of analog signals, and is passed to the ADC for conversion.
2. EEG data is packed into hardware buffers and is eventually passed over a bluetooth connection to the Raspberry Pi 3B.
3. The Pi is executing the machine learning algorithms locally. It then takes the EEG data that was previously fetched over Bluetooth, and proceeds to feed in CSV files containing the EEG data to the machine learning process.
4. After data filtering and processing within the machine learning process, the data fed in has been classified. Once data is discretized, commands can now be issued, and they are stored into Firebase for collection by the iOS application.
5. EEGMindReader iOS App receives an event notice, denoting that new commands are queued into the Firebase command database. These new commands are now actionable, and are finally fetched and viewed by the app.

A general architectural solution for this project is split into two parts—hardware and software. The hardware consists of a printed circuit board (PCB), electrodes and a 3D printed headset as a device enclosure, as seen on figure 1 below.

The dry electrodes will measure changes in the user's' brain activity on the order of a milliVolt of potential change. This analog signal will then be processed through the ADS1299 on the PCB. The ADS1299 features an 8-channel, low-noise, 24-bit analog to digital converter with built in programmable gain amplifier. The ADS1299 will be interface through a low pass filter with 500 Hz cutoff, since most EEG signals are between 0 to 400 Hz. Communication to the ADS1299 will occur through a LPC1758 microcontroller, featuring an ARM Cortex M3 processor.

The software required is also divided into two sections, C drivers for interfacing to the microcontroller and machine learning code for signal processing. The C drivers written can be seen under the communication protocol of the Interconnect Table above. These include drivers for SPI, UART and I2C communication. A compatible driver must also be designed and developed to interface with the ADS1299. Almost all information as how to interface between the microcontroller and the ADS1299 as well as other peripherals can be found on the respective datasheets of the devices. A bluetooth driver over UART will also need to be developed as will drivers to interface with the corresponding smart devices.

The machine learning portion will involve signal pre-processing which include the process of Epoching and Averaging the signal data. Epoching and Averaging is simply the discretizing of signals and mapping signal changes to corresponding sections of the neural membrane. Further signal processing includes the removal of physiological artifacts such as blinks and heartbeats. Initially, basic machine learning algorithms such as K Nearest Neighbors can be applied to further discretize signals into sets. Further investigation might include the combination of other machine learning algorithms.

4.2 Interface and Component Design

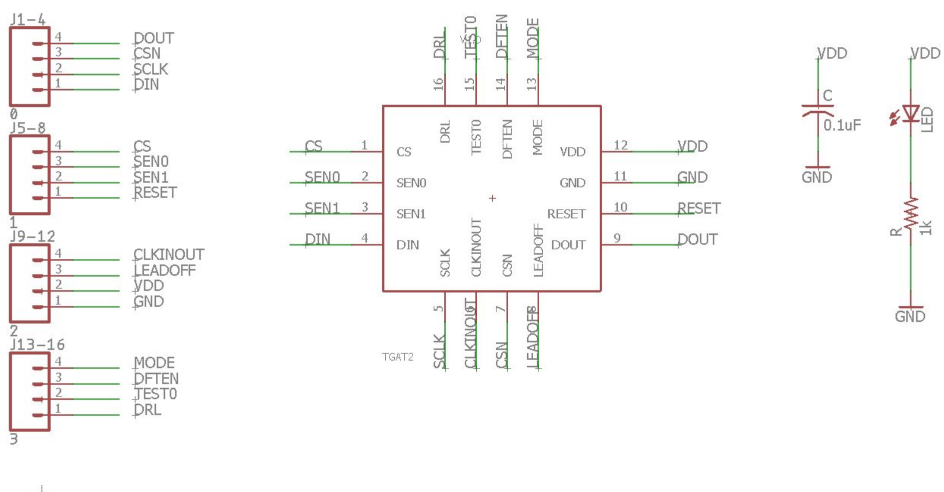


Figure 2. PCB Breakout Schematic

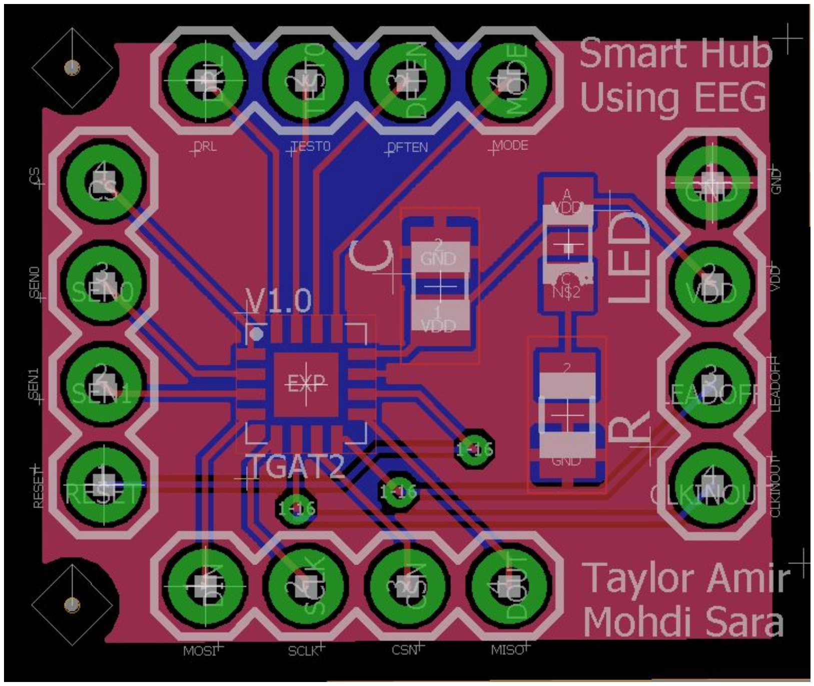


Figure 3. PCB Layout and Design

Figures 2 and 3 depict the PCB design for a secondary electrode interface. This module is much smaller in size than the ADS1299; this secondary board contains a TGAT2 which is three by three millimeters in area and contains sixteen external pins, whereas the ADS1299 requires sixty-four pins, ten by ten millimeters surface area, and external discrete components for functionality. This specialized PCB is necessary for a compact and mobile design, which gives an increased number of functionalities for our product.

The headset is 3D printed and contains gold metal electrodes touching the subject's head as well as a custom PCB and a power supply. The custom PCB consists of an ADC connected directly to the electrodes. This ADC sends digital serialized data to an ARM CPU which caches the serialized data in Flash Memory until there is an appropriate connection to send the data. This connection is in the form of Bluetooth to the smart devices. A visualization of the 3D printed headset can be found in Figures 4 and 5 found below.

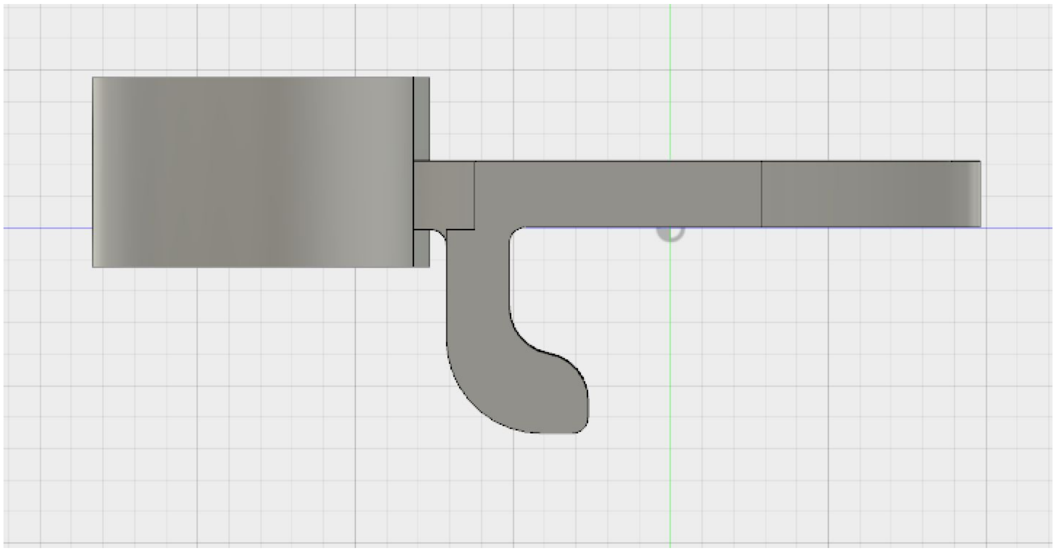


Figure 4. EEG Headset side view

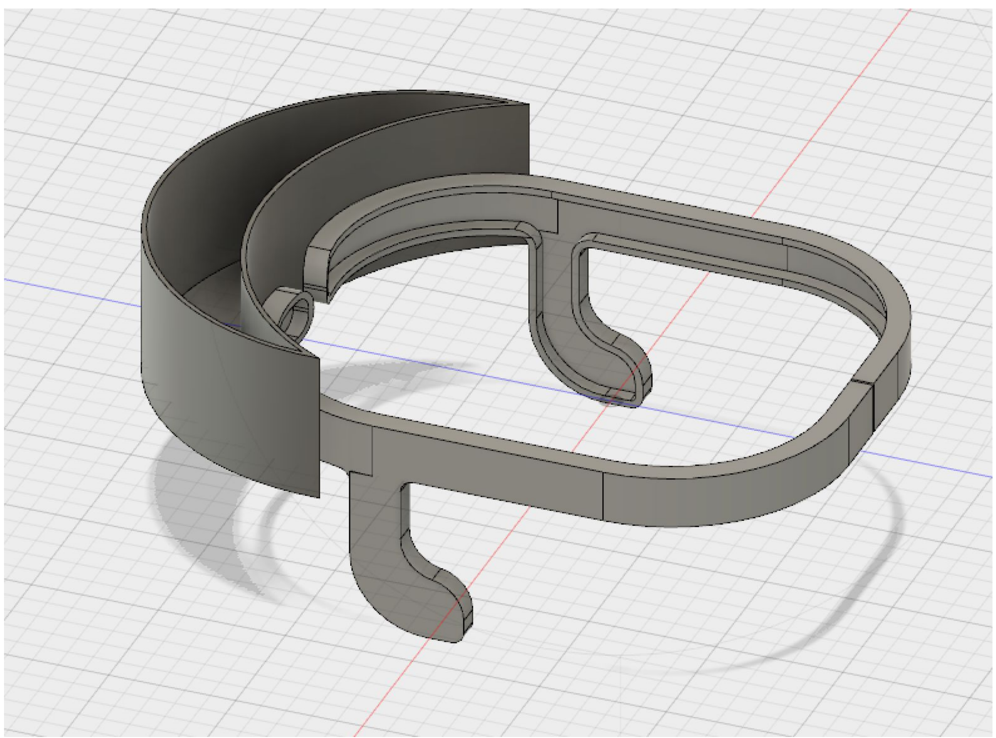


Figure 5. EEG Headset 3D view

Figure 6. Login Page

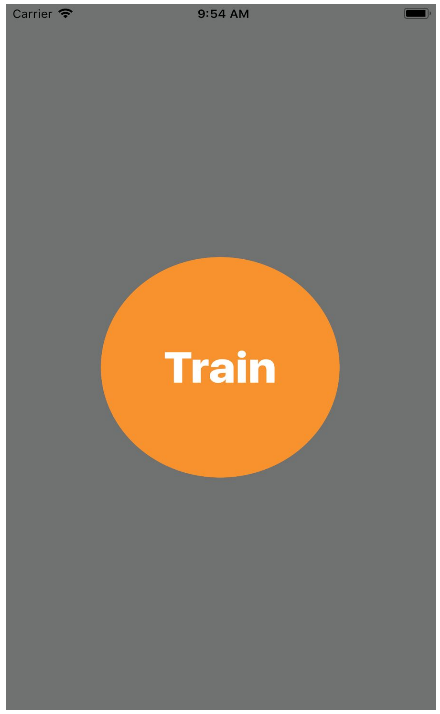


Figure 7. Home Page

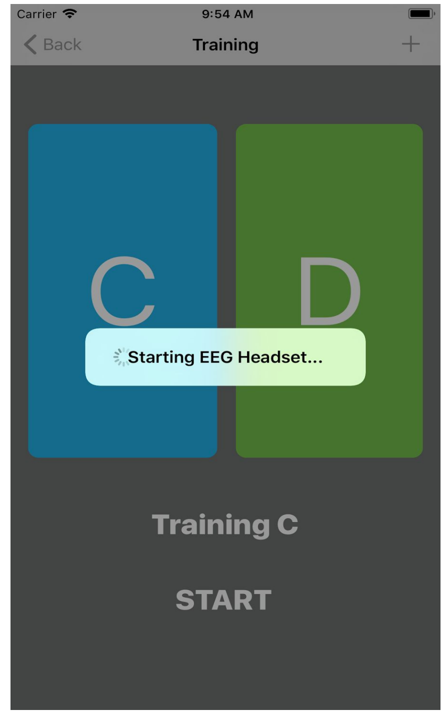


Figure 8. Loading Page

Figure 9. Training Page

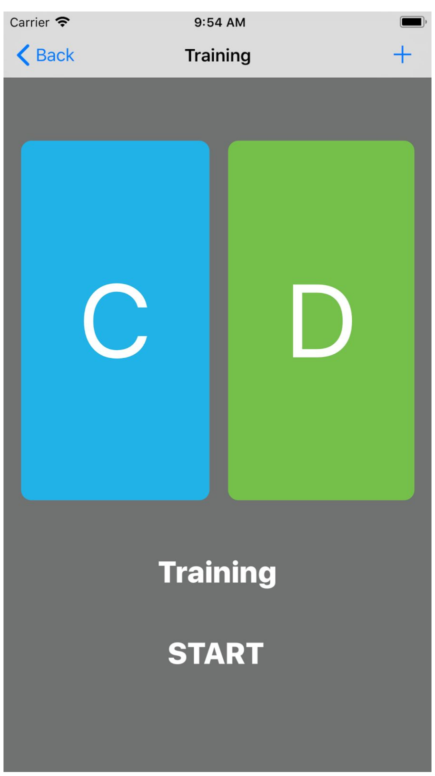


Figure 10. Theme

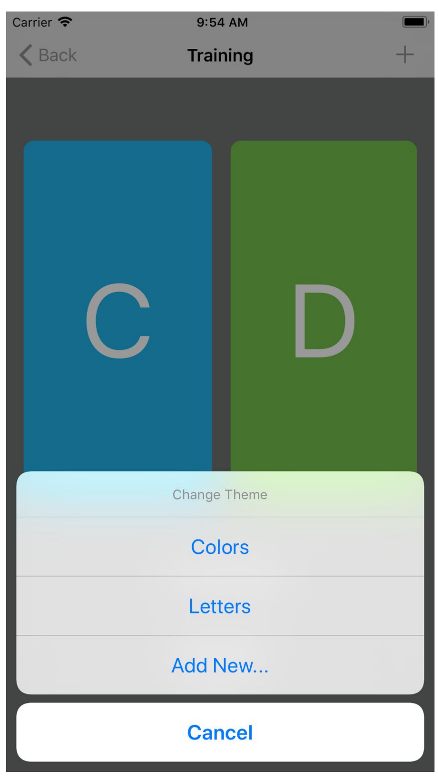
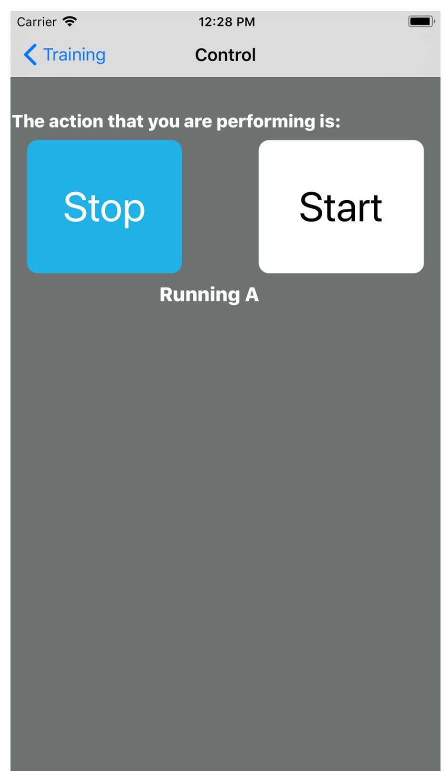


Figure 11. Running Classifier



4.3 Structure and Logic Design

The structure and design of the final form factor shall take multiple iterations to finalize. However, the headset will contain the electrodes placed in such positions for optimal signal strength and accuracy. The PCB fits in the back of the headset. The firmware of the PCB has several components. First communicating with the ADC entails programming registers initially, as well as creating a state machine which will take an echo (time segment) of EEG data and enter this into the machine learning algorithm. A combination of KNN and random forest algorithms will allows us to remove noise and further distinguish the signal. This data is then sent through UART communication protocol to a bluetooth transmitter which will then communicate to the smart devices.

4.3 Design Constraints, Problems, Trade-offs, and Solutions

4.4.1 Design Constraints and Challenges

Several design constraints exist on such a specialized process. Economically, the budget of college students is very small, and with no form of external funding several of the design decision made had cost as a deciding factor. The most expensive component in this project is the PCB. The most expensive component on the PCB is the ADC. The cost of the microcontroller, electrodes, accelerometer, gyroscope, and other passive components are rather negligible in terms of the overall cost.

In terms of resources, the faculty at school as well as online publications are very helpful in the design decision making process. As with any other endeavor that hasn't been done before there is very limited information on the application side, that is, issuing commands to the smart devices from the headset. Most publications on the topic of EEG have been solely medical related so there is little guidance as to our application.

In regards to society and environment, while this project is structured such that accessibility is an inherent factor, the environment has also been taken into consideration. By using minimal parts all from local manufacturers we strive to make the environmental impact of this project minimal. The headset will be printed with biodegradable environmentally safe material.

In terms of the constraints placed on hardware design, the extremely low resolution and high sensitivity to noise of the EEG signal makes it difficult to attain a reliable signal. The signal, on the range of 0-400 Hz is very sensitive to any sort of external noise. On the PCB the signal is so sensitive that the raw signal data should not be traced in between layers and that the shortest path should be used between the raw signal data and the processing device. In designing the PCB another constraint taken into account is the size of the board. The board should be as small as possible so as not to have a big heavy device on the user's head. When decreasing the size of the PCB not only does the cost of manufacturing go up exponentially so does the noise of signals in proximity. In terms of software, one design constraint is the choice of signal pre-processing and machine learning algorithms.

In terms of safety and reliability, all user interaction with the product will be through button on the headset. The only direct contact the user will have is to the electrodes themselves which is necessary. The electrodes are one-sided such that current is never able to go through the electrodes to the user's brain. The design will be thoroughly tested for reliability throughout all stages of the design process.

4.4.2 Design Solutions and Trade-offs

The design solutions that follow are results of immense deliberation and considerations. Economically, in order to work with such a limited budget, the first decision made was to use the LPC1758 microcontroller. This microcontroller was chosen over others because of its low cost and reliability. The tradeoff made here is the higher computation power provided with newer microcontrollers, however, our application is not in immediate need of much faster processing power. Choosing a cheaper microcontroller also allowed us to purchase a more expensive ADC.

In terms of resources, most of the reference material in terms of the EEG signals had to be adapted from medical publications. Medical publications use the EEG signals in a much different way as the large majority of publications are attempting to map some sort of illness or deficiency to neural signals. The tradeoff made here between medical publications and information on EEG device manufacturers is that the information provided by EEG manufacturers is more general and suited to a variety of applications. We found that even though medical publications were more specific material derived from them was much more relevant in terms of our application.

The decision to purchase the ADS1299 was the biggest design decision made. There were several alternate ADC's that converted analog to digital signals such as the ADS1256 and the LTC2216 that provide low noise and high resolutions. However, the ADS1299 was chosen because not only was it an 8-channel device with low noise but it also has an array of programmable gain amplifier which are necessary for EEG signals. When designing the PCB the problem of space and signal reflection was solved by increasing the number of layers on the PCB, while insuring that the raw data input had the shortest path to the ADC. Additionally, low pass filters, termination resistors, and harmonious management of both analog and digital components within one device was achieved.

Chapter 5 System Implementation

5.1 Implementation Overview

We have used EEG sensors to communicate with Raspberry Pi 3B+ Microcontroller. We are able to process the analog data by analyzing the incoming signals. Currently, we have experimented with numerous electrodes since getting accurate data is the most important part of this project, being that the data defines the efficacy of this project. The EEG system records the potential of various location on the scalp. Upon four separate locations on the scalp resides an electrode. These EEG sensors are typically made from metal and contact the skin through conductive gel. Alternative sensors used in the system include MEMS-based, dry electrodes that create a good contact with the skin without requiring gel. This makes for a much more convenient product setup and allows for a product that is capable of long-term recording by not requiring a user to reapply conductive gel. An additional feature of our front-end circuitry is that it can be used equally well with dry sensors or gel-based sensors. We solved the need for mobility in EEG recording equipment by creating a daisy-chain of digital outputs consisting of all the electrode boards. The wiring requirements for power, control, and readout are reduced making a wireless solution now it will be much more feasible. Also, mobility is possible because only a small amount of power is required for the whole signal processing system within the headset. Small batteries can be employed to power the system for longer period. As we receive data we are able to train the data and tweak our algorithm autonomously to create a personalized model base on the user's brain waveform. Python code was the main language for our data processing due to the available Python modules and code portability.

We constructed an iOS application which communicates to the headset, and based on the trained data, displays the commands which is user is capable of changing at any time. Our main dependencies are hardware related. For instance, to deal with the different noise levels of the dry electrodes a zero-phase bandwidth FIR bandpass filter with the range of 1Hz to 20Hz was applied to data.

To train the system we adapted calibration process which users were given 5 minutes to think and receive immediate feedback starting from the first second. After calibrating the data, this feature was turned off and the overall system was tested with fixed classifiers.

5.2 Implementation of Developed Solutions

For testing and integration phase, we have a dynamic stopping method that collects the data within the hardware buffer up to a predefined limit, we then average buffer data, and pass the data through the specified filters. Once the data is filtered, it is then passed on to templates which were created from existing machine learning models—these models have been classified by large amounts of correlating data. Data correlation was used in our work for better dynamic stopping. Using different methods would have increased the process of dynamic testing and generating templates from the previous models would have certainly not be one of our option.

To evaluate the performance of the EEG data paths and to compare the different methods we used the following metrics: accuracy is the percentage of correctly selected commands, while the bitrate is based on the information transfer rate.

5.3 Implementation Problems, Challenges, and Lesson Learned

Creating an accurate software filter and hardware methods were certainly our biggest challenges. We have learned many electrical techniques to clear out the noise of our data, such as low pass filters, decoupling capacitors, and more. As far as software filtering, we have had to adjust our filters on real time based on the noise. Noise filtering in real time based on the signal levels and floors could be challenging. Logistical difficulties in the placement of the electrodes was another challenge. We were able to stabilize the project phase by doing unit tests and it has been the biggest success of this project. Our group was able to nicely integrate different parts because of unit test.

Chapter 6: Tools and Standards

6.1. Tools Used

There were several tools used to complete this project. Since this project has both software and hardware components, there are several software tools were used. On the software side, the main Integrated Development Environment used for the development and testing of the code is Eclipse Mars. We chose Eclipse Mars because it is open source and seen by many as the industry standard for writing C/C++ code.

We designed our custom PCB on Eagle v5.23 because Eagle gives you the ability to generate the standard Gerber files while also being free and providing a vast library of parts. We designed our custom headset using Autodesk, a free version of the industry standard Autocad, because it has all the standard CAD features, and it also is compatible with the 3D printers provided at our university. Finally, the most fundamental tool to our project and perhaps our success was the source versioning tool, Git, by using Git we were all able to collaborate, testing each other's code and make sure that only the most optimized versions of code were placed on the final product.

6.2. Standards

While many of the tools used in this project were industry standard, very little standard documentation was used in the construction of this project. This was due to the fact that this project is rather new even in its field. Sparse documentation and other reference materials available online were mostly in medical journals where the focus of the technology was described as clinical trials.

Chapter 7 Testing and Experiment

7.1 Testing and Experiment Scope

Since the system design of this project has many intricate and complex facets, a great amount of care must be taken when testing. Starting with the code, most of the key functions that performed any kind of comparing, weighting, filtering or transmitting of data were unit tested. The goal of this testing was to get rid of any errors in programming. A series of integration testing was performed with the device and the headset, and again with the headset complete and the mobile application. Stress testing was also done to test the reliability and functionality of the headset over a long period of time. The mobile application was also unit tested as were the individual components inside the application.

7.2 Testing and Experiment Approach

Most of the code in this project was unit tested, since unit testing requires intimate knowledge of the code, unit testing in our case was white box testing. The unit testing was completed using the standard c-test library that is used for testing C/C++ code. The next level higher is the state machine that controls the main micro controller which was unit tested for all possible values. The machine learning model was black box tested since the algorithms implementation was part of a third party library. Since the machine learning model and its implementation were written in Python the standard pytest library was written for all of the testing.

7.3 Testing and Experiment Results and Analysis

The accuracy of the machine learning library while the headset is on a user is 79.59 percent using the Root Mean Square Error indicator. Even though the unit test coverage is not 100% of the code base, the unit tests that are in place do all pass. The grey box testing on the state machine has validated our code and we have not came upon a scenario where it might fail.

Chapter 8 Conclusion and Future Work

The development of this product throughout the past several months has truly been the culmination of the work and the knowledge that we have attained at SJSU. In successfully, developing an EEG headset that is able to execute discrete commands to a smart device we have furthered the field of Electroencephalography in hope that the field will continue to grow once we have moved onto our future endeavors.

In order to go further with our project there must be advancements made in the field of EEG sensors such that reliable accuracy of neural signals can be attained using a noninvasive form factor. The biggest challenge in completing our goal was the limitation of EEG sensors and the inconsistent data that accompanied such sensors. While we were able to temporarily remedy this issue by providing a companion mobile application that would allow the user to select between two commands at a time, this proves to extremely diminished the user experience.

While there might be a lot of work involved in polishing our product into a marketable item, our proof of concept shows that this sort of brain to computer interface is not too far off into the horizon. In the difficulties faced while developing this product we learned more than we could have in a traditional classroom setting, not only about computer hardware and software but arguably more importantly about teamwork, communication, and the value of integrity and hard work.

References

1. J.D. Bayliss, D.H. Ballard (2000), A virtual reality testbed for brain-computer interface research. IEEE Trans Rehabil Eng. 8(2):188-90.

2. Pfurtscheller G, Neuper C, Guger C, Harkam W, Ramoser H, Schlögl A, Obermaier B, Pregenzer M. (2000), Current trends in Graz Brain-Computer Interface (BCI) research. IEEE Trans Rehabil Eng. 8(2):216-9.

3. A. Schlögl, B. Kemp, T. Penzel, D. Kunz, S.-L. Himanen,A. Värri, G. Dorffner, G. Pfurtscheller (1999). Quality Control of polysomnographic Sleep Data by Histogram and Entropy Analysis. Clin. Neurophysiol. 110(12): 2165 - 2170.

4. S.P. Levine, J.E. Huggins, S.L. BeMent, R. K. Kushwaha, L.A. Schuh, M.M. Rohde, E.A. Passaro, D.A. Ross K.V. Elisevich, B.J. Smith (2000). A Direct Brain Interface Based on Event-Related Potentials. IEEE Trans Rehabil Eng. 8(2), 180-185. A.

5. Schlögl (2000), The electroencephalogram and the adaptive autoregressive model: theory and applications, Shaker Verlag, Aachen, Germany.

6. G. Pfurtscheller and C. Neuper (2001). Motor imagery and direct brain-computer communication. Proc. IEEE 89(7), 1123-1134.

7. McAlone, N. (2017, January 30). These engineers rigged a device to let you control Netflix with your mind. Retrieved December 04, 2017, from

http://www.businessinsider.com/mindflix-lets-you-control-netflix-with-your-mind-2017-1

8. Yasmin Anwar, Media Relations | September 22, 2011March 24, 2016, & Anwar, Y. (2016, March 24). Scientists use brain imaging to reveal the movies in our mind. Retrieved December 04, 2017, from http://news.berkeley.edu/2011/09/22/brain-movies/