BCI Based Home Automation using User Controlled Blinks

5 authors, including:

Kelvin Ortíz Chicaiza
Escuela Politécnica Nacional
4 PUBLICATIONS   **12** CITATIONS

SEE PROFILE

José Luis Murillo López
Escuela Politécnica Nacional
6 PUBLICATIONS   **20** CITATIONS

SEE PROFILE

Sang Guun Yoo
Escuela Politécnica Nacional
78 PUBLICATIONS   **705** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project Device Control using Brainwaves View project

Project Smart Parking View project

BCI Based Home Automation using User Controlled Blinks

[1, 2]Sebastián Poveda Zavala, [1, 2]Kelvin Ortíz Chicaiza, [1, 2]José Luis Murillo López,
[1, 2]Juan Sulca and [1, 2]Sang Guun Yoo
[1]Departamento de Informática y Ciencias de la Computación, Escuela Politécnica Nacional,
Quito, Ecuador
[2]Smart Lab, Escuela Politécnica Nacional, Quito, Ecuador

Abstract: Nowadays, Internet of Things (IoT) is becoming a major research area, since, its applications can improve people's life. Even when its purpose is to facilitate daily tasks for everyone, it does not fully consider people with disabilities. Even voice or gesture command technologies can help most of people with disabilities, Motor Neuron Diseases (MND) patients who have lost most of their movement cannot be benefitted of such technologies. For this reason, this research aims to design and develop a scalable BCI based home automation system for people with movement disabilities which is easy to use and comfortable for the user. The proposed solution gathers signals generated by voluntary eye blinks from the FP1 electrode position and classify them on short and composed long blinks. Then, a combination of these blinks allows the user to navigate in a Graphic User Interface (GUI) of an application created to control different devices of a smart home via. MQTT protocol. Test results obtained in this work shows that the proposed system could be used in real life solutions.

Key words: Brain computer interface, BCI, home automation, Internet of Things (IoT), smart home, MQTT, disabled people

INTRODUCTION

The proliferation of the Internet of Things (IoT) has allowed the interaction of different kind of devices with their users and with the environment where they are installed, delivering benefits to people with new services such as monitoring and automation solutions. However, even though IoT was conceived as an inclusive technology, most of real solutions do not contemplate disabled people as users.

According to World Health Organization (WHO), about 15% of the world's population live with some form of disability (WHO and WB., 2011) which includes mental and mood disorders, genetic diseases, neurological disorders, physical impediments, among others. Most of these conditions hinder the interaction between the person and its environment. For example, Motor Neuron Diseases (MND), remove the possibilities of their patients to interact with the environment by the progressive destruction of motor neurons. This group includes progressive diseases that deprive affected patients of essential voluntary movements such as talking, walking or even breathing in the final stages (NIH., 2012). Some examples of MNDs are Amyotrophic Lateral Sclerosis (ALS), Progressive Bulbar Palsy (PBP), Primary Lateral Sclerosis (PLS), among others. In any case because of

these difficulties, all these people are losing the benefits of new IoT technologies, since, most of the sensors that work as IoT devices input are based on user's movements, gestures or voice commands. Even when voice commands could be very useful for most disabled people, advanced stages of MNDs hinder the patient talking. And actual BCI based IoT solutions use a limited and non-scalable user-interface or directly leave out it, making the user use and memorize complex and large operation commands. So, thinking on the user comfort, this researche proposes a scalable system with a friendly user-interface.

Background: In recent times, a variety of technologies has been developed to help people with disabilities and one of the most active one has been the delivery of a system in which people with disabilities could regain their independence and improve their lifestyle by controlling different devices inside their home. In this aspect, several alternatives for home automation using BCI devices have been proposed so far.

Bhemjibhaih *et al.* (2008) have proposed a BCI based home controlling system that locates the position and orientation of the user with the help of a Kinect motion sensor and then identifies the device that the user is targeting by obtaining the attention values of the user. This value is captured periodically and if such value goes

Corresponding Author: Sang Guun Yoo, Departamento de Informática y Ciencias de la Computación,
Escuela Politécnica Nacional, Quito, Ecuador

beyond a threshold, then it is taken as a user input. Finally, the viewing angle of the user is calculated to help with the identification of the device that wants to be controlled. The control can be as simple as on/off or multilevel (Bhemjibhaih *et al*., 2008). Even though the system is very interesting, it presents several limitations such as that the system is usable only in indoor scenarios, the Kinect sensor must be installed in every room and the user has to be in the same room of the device that he/she wants to control and finally, the user must move to a position in a way that he/she is looking at the device to be controlled.

On the other hand, Zavala *et al*. (2018) have developed a system that allows the user to control devices using blinks. The system uses the MUSE headband to detect raw brain waves and classifies them into long blinks and short blinks. These blinks are translated into an encoding system which is similar to the morse code (Zavala *et al*., 2018). However, this system is not scalable since the number of blinks utilized for a command increments with the number of devices and this could produce fatigue to the user.

Additionally, Shi *et al*. (2017) have proposed a virtual remote-control system using a P300 BCI that uses the attention level of the user. The remote control corresponds to a 4×5 matrix of symbols including 0-9, switch, up, down, left, right, OK, mute, back, '*' and '#'. The classification algorithm was done utilizing a support vector machine ensemble (Shi *et al*., 2017). Despite the system presents an overwhelming setup with 14 electrodes, 2 references, 2 HEOG (Horizontal Electrooculography) and 2 VEOG (Vertical Electrooculography), the system can only be used when the user is focused. This could prove to be a difficult task in order to maintain adequate control and finally, the long response times could produce visual fatigue to the user.

Finally, Alrajhi *et al*. (2017) have developed a system that allows people who suffer from quadriplegia to be able to open and close doors using a combination of BCI and smart home technologies. The system uses Emotiv EPOC+ and two different suites from Emotiv (the Cognitive Suite and Facial Expressive Suite) to control the developed BSH system that will handle the control of the door (Alrajhi *et al*., 2017). Even though the proposed system is very interesting, the system is limited to providing control over only doors. Additionally, it could detect unwanted facial expressions which could lead in undesirable door controls. Besides, it needs assistance of another person to start and stop the system, once analyzed different solutions, the intention of this study is to develop a real-world system that solves the different types of limitations that are presented in previous works.

MATERIALS AND METHODS

Proposed solution: Movement related disabilities have a negative effect on the people who suffer from them, since, it reduces the ability to do tasks on their own. In many cases, very simple tasks such as turning on a television, a light bulb and other devices of daily use, become difficult to do without the help of another person. IoT-based solutions have been implemented over time exploring ways to provide a better lifestyle for people with disabilities (Lopes *et al*., 2014). With this in mind, we have thought of a solution that allows people with disabilities to perform simple tasks by obtaining signals from the subjects which will be filtered and sent to a server that will be responsible for the control of the devices. Figure 1 shows the general architecture of the proposed solution.

The proposed system provides people with disabilities an easy way to control electronic devices by using EEG signals generated when the user blinks. The blinks of the user is detected by the EEG and such information is transmitted to the computer that executes a BCI application with filters and classifies the raw blink signals. Once understood the command that the user wants to execute the BCI applications sends the command to the IoT devices through the communication server. Using the proposed system, the user can have the control of IoT devices (such as Smart TV, Lightbulbs and Fans) without being close or having physical interaction.

Implementation: The design proposed in the previous section has been implemented using the diagram shown in Fig. 2. The EEG device used in this implementation was the OpenBCI V3 which handles the signal acquisition process. OpenBCI was used for its advantages over other
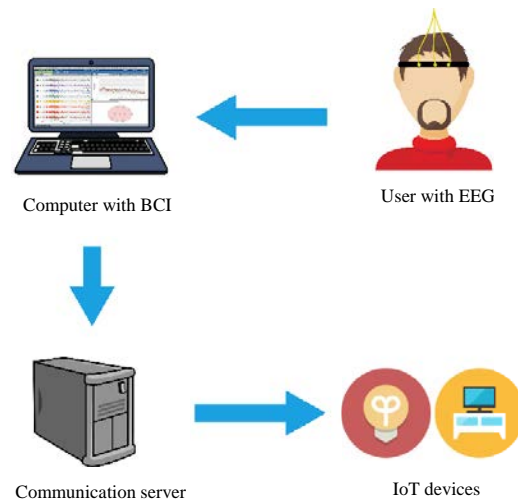

Fig. 1: General architecture of the proposed BCI system

EEG systems such as modularity, relatively low cost, open architecture and compatibility with different research platforms such as OpenViBE, EEGLAB, BCILAB, MATLAB, etc. and it has APIs for Python, C#, Java, processing, etc. (OpenBCI, 2019). On the other hand, the acquired data is sent to a Single Board Computer (SBC) called ODROID XU-4 which is responsible of the feature extraction and feature translation processes. SBC also has the function of publishing the result of the feature extraction/translation process to an MQTT server which allocates the states of the IoT devices controlled by the user, i.e., a lightbulb and a television in the present implementation. The lightbulb is controlled by a solution using an ESP32 microcontroller and the television is controlled by using a solution based on Raspberry Pi 3 B+. In the next subsections, the details of the implementations are presented.

Signal acquisition: As mentioned before, the signal acquisition process was executed using the 32 bit OpenBCI V3 system. The EEG signals are captured by the OpenBCI and transmitted wirelessly using its Bluetooth interface to the main program of the system. As mentioned before OpenBCI was used for its benefits such as modularity, relatively low cost, open architecture and compatibility with different research platforms such as OpenViBE, EEGLAB, BCILAB, MATLAB, etc. and it has APIs for Python, C#, Java, Processing, etc. (OpenBCI, 2019).

Main controller: The SBC used as the main controller was ODROID XU4. Thi component stores and executes the application responsible of signal processing and user interface. The application is executed on Ubuntu 18.04 Mate and the user interface of the application is shown in a 7-inch HDMI display-C with 1046×600 resolution. ODROID XU4 was chosen because it comes with improved processing capabilities (ARM® Cortex® - A15™ up to 2.0 GHz and ARM® Cortex® -A7™ up to 1.4 GHz) over traditional SBCs such as Arduino and Raspberry Pi while maintaining an efficient power consumption (Hardkernel Co. Ltd., 2019).

The developed user interface has a high degree of usability, since, the navigability in the environment is very intuitive (Fig. 3). The interface allows to change the states of the IoT devices established on the MQTT server. For the development of the user interface, the open-source framework called electron was used, since, it allows to develop lightweight cross-platform application using JavaScript, HTML and CSS.

In addition, for the delivery of messages between the application responsible for obtaining EEG signals and the
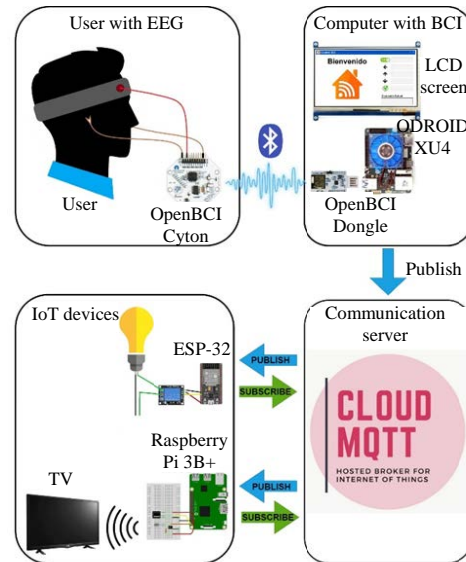

Fig. 2: Complete diagram of the proposed BCI system

Fig. 3: Interface of the developed application

delegated application of controlling the user interface, a UDP socket communication was implemented which was used for its lightness.

MQTT server: Message Queuing Telemetry Transport (MQTT) is an extremely lightweight and simple messaging protocol (RPF., 2019). Mosquitto implements the MQTT which provides lightweight methods for sending messages using a publication/subscription message queue model. The types of publication/subscription messages have been grouped by themes for the control of a television and a light bulb. The need for a light messaging service is crucial, for this reason we have found the need to hire a message agent hosted for the internet of things. The MQTT protocol was chosen to be implemented in the system, since, it optimizes the resources of the device and the network bandwidth used by the application.

Controlled IoT devices

Lightbulb control: To control the lightbulb, an ESP32 microcontroller was used (Fig. 4). The board is designed to support Wi-Fi and Bluetooth. It has 36 GPIOs, 14 of which are analog to digital converters that can be connected to sensors (Allafi and Iqbal, 2017). The role of the microcontroller within the system is to connect to the MQTT server to read the state of the light bulb (i.e., on or off state data) and use such data to change the state of the lightbulb physically. This component of the system uses the WiFi network to connect to the MQTT server (which can be in a local network or over the internet).

TV control: To control the television, an Infrared Radiation (IR) controller was built. The controller captures the TV's original IR remote control signals for executing the main functions, i.e., turn on/off the television and change channels. Those signals are stored with a unique name inside the proposed controller to clone the functionality of the TV's remote control and then they are sent by the IR transmitter of the controller when an action is required to be executed. The controller also maintains a communication with the MQTT broker to update/receive the status of the corresponding topic.

For the implementation of the prototype, Raspberry Pi 3 B+ was used since it is the last model of the Raspberry Pi 3 family with the best specifications, in other words, raspberry Pi 3 B+ offers the features of a small computer at a low price. It has a 64 bit quad-core processor (Broadcom BCM2837B0, Cortex-A53 SoC @ 1.4 GHz), 1 GB of LPDDR2 SDRAM, dual-band 2.4 GHz and 5 GHz wireless LAN (RPF., 2019). Regarding the software, it works with the Raspbian Operating System (Kernel version 4.14) and uses the LIRC package to decode and send the IR signals (Fig. 5).

Feature extraction and processing: EEG signals are obtained using the Cyton biosensing board (OpenBCI V3) with one active reusable flat snap electrode (TDE-202) located on the surface of the scalp (FP1) according to the extended 10-20 system (Klem *et al.*, 1999) using a sampling frequency set to 250 Hz. Standard Ten20 conductive electrode paste was applied between the electrodes and the scalp in order to bring the impedance down. In order to stabilize the signal, we applied a 0.5 Hz high pass filter, a 50 Hz notch filter and finally, band-passed by 5-25 Hz. These filters allowed to remove the DC offset that is present in the DC-coupled EEG amplifiers of ADS1299 used in the Cyton biosensing

board. The offsets are also known as slow cortical potentials (Birbaumer, 1999) and they can be positive or negative. Figure 6 shows the difference between the raw data and the filtered data.

Feature translation: The acquired signals were classified in short blinks and composed long blinks. A short blink is detected when the user blinks normally. When this is done, the signal drops below -75 uV and then rises above 85 uV in a time frame of 1-1.5 sec (Fig. 7). On the other hand, a composed long blink is detected when the user closes their eyes for 2-5 sec, opens them and then blinks quickly again, when this happens, the signal drops below 75 uV, it then returns to a normal state (between -25 and 25 uV), rises above 100 uV and then
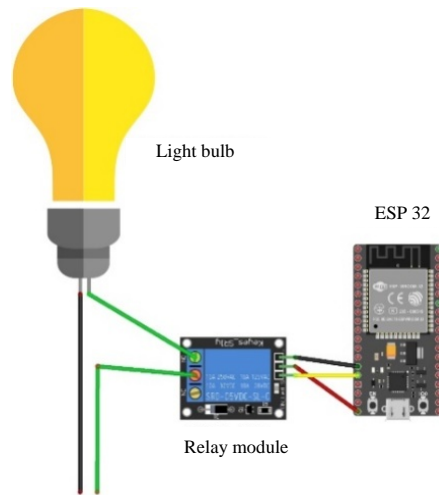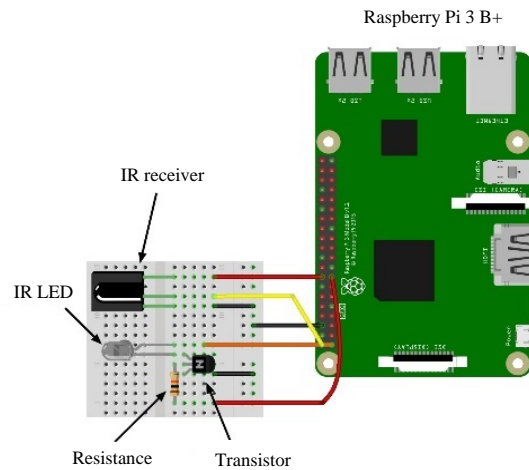

Fig. 4: Lightbulb control circuit

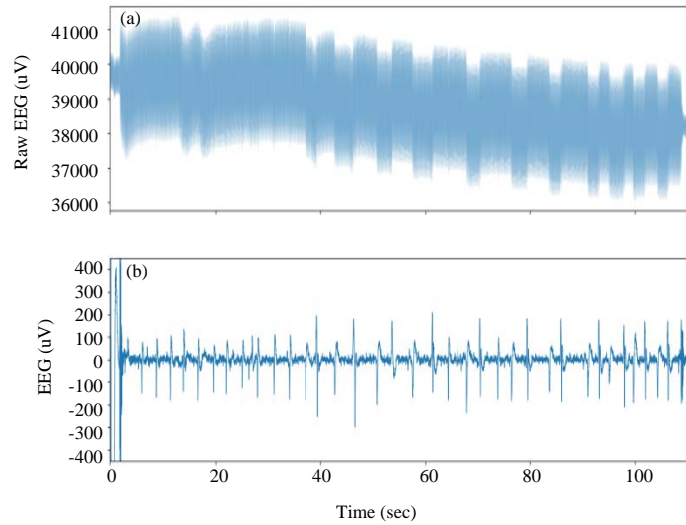Fig. 5: IR based TV controlled circuit

Fig. 6(a-b): Extracted raw data vs. filtered data, (a) Raw data and (b) Filtered data
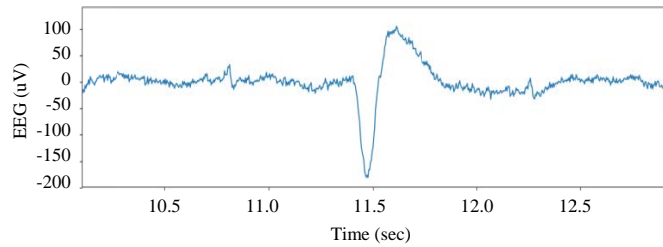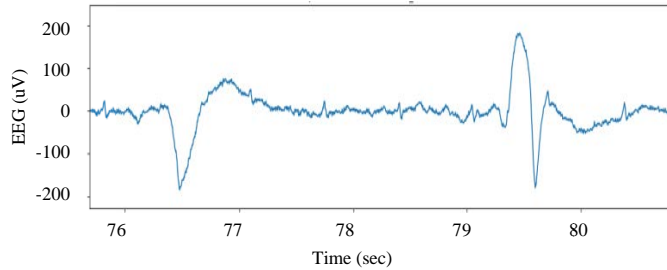

Fig. 7: Short blink

Fig. 8: Composed long blink

drops below 75 uV again all these changes are done in a time frame of 2-5 sec (as shown in Fig. 8).

Signal classification: In this system, the classification was done utilizing Python 3.7. The prototype was developed in this language, since, it is totally compatible with OpenBCI Software library and it supports other useful libraries such as numpy/scipy/pandas.

Once established the signals to be used as a composed long blink and a short blink, the encoding system for correlating the types of blinks with commands to control the IoT devices was developed. For ease of interpretation, we have decided to represent the short blinks with dots (.) and composed long blinks with dashes (-). It is important to mention that, since this encoding system must not interfere with the normal blinking patterns of users, the system was configured to use four consecutive composed long blinks for system activation. We have chosen this combination because this pattern is not presented commonly in a normal blink sequence. Based on this antecedent, the present work has used the following encoding system (Table 1).

Command	Description	Blink combination
Activate/ deactivate	Activates or deactivates the system	----
Go back	Returns to the previous screen	---.
Go up	Selects the upper option in the menu	--..
Go down	Selects the next option in the menu	--.-
Accept	Executes the selected option	-...

RESULTS AND DISCUSSION

Two experiments were conducted in this work. Three 20-23 year-old healthy peoples participated in the experiments. Before experiments, each participant used the system for a total of 4 h in order to get used to its usage.

Experiment 1 system testing: The aim of the first experiment was to test the real processing time of the system. Since, the functionality of the proposed solution is highly dependent on the internet connection (due to MQTT server), its execution time will be affected by this variable and for this reason it was suppressed.

In this situation, the communication between the user application (EEG and ODROID) and IoT devices, i.e., lightbulb control and tv control having the intermediation of MQTT server was measured to understand the overhead of the system in terms of delay. In this experiment, we have completed 120 tests for each device and the obtained results are summarized in Table 2.

The results indicate that the processing time of the system is not even the 5% of the total execution time. In other words, most of the execution time depends on the speed of the internet connection. Inside the processing time the ODROID's job of verifying the command and publishing a message to the MQTT server are included. As you can see, the control of the lightbulb is very low thanks to the benefits of using an ESP32 microcontroller. On the other hand, Raspberry Pi 3 B+ takes much more time, since, it has to read the message published on the topic and since, it make use of the LIRC to send the corresponding Consumer Infra-Red (CIR) command.

Experiment 2 application testing: In experiment 2, the participants were asked to complete two tasks. The first task consisted on the following steps: activate the system, turn the lamp on and deactivate the system. The finite state machine in Fig. 9 illustrates the first task.

On the other hand, the second task consisted on the following steps: activate the system, turn the television on, change the television's transmission channel, return to previous channel and deactivate the system. To complete this task, the participants had to execute the finite state machine illustrated in Fig. 10. In the present experiment, when the participant fails completing a command, it has been considered as a
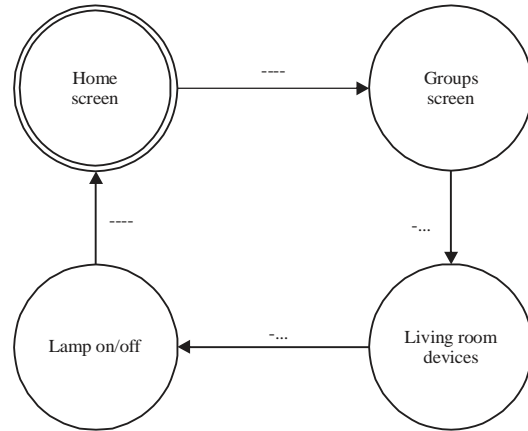

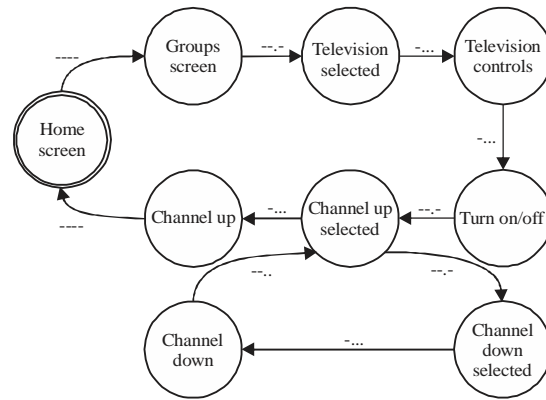Fig. 9: Finite state machine for task 1

Fig. 10: Finite state machine for task 2

Table 2: System testing results

Device	Average processing time (msec)
Lightbulb control test	
ODROID XU-4	8.827
ESP32	1.200
Total processing time (average)	10.027
Total execution time (average)	267.977
TV control test	
Device	Average processing time (msec)
ODROID	10.798
Raspberry Pi 3 B+	82.875
Total processing time (average)	93.673
Total execution time (average)	3319.597

failed attempt. Table 3 shows the average attempts the user needed to complete the first and second tasks as well as the elapsed time executing such tasks. We believe that 1.33 attempts at 44.52 sec for the first task and 1.38 attempts at 83.49 sec for the second task are acceptable for the first prototype. Regarding, the 9 attempts executed by the third participant on the first task, we believe that this could happened due to the fatigue that the user could be felt in the repetitive execution of the experiment.

Table 3: Number of attempts for tasks 1 and 2

Participant	No./task 1	Elapsed time (sec)	Task 2	Elapsed time (sec)
1	1	42.05	1	71.98
	0	31.88	0	67.45
	0	30.15	4	139.56
	0	32.35	1	101.14
	0	32.22	0	68.9
	0	32.43	4	118.8
	0	31.44	2	99.74
2	1	49.24	0	65.84
	2	40.31	2	89.12
	0	38.56	1	68.14
	0	37.55	1	72.98
	0	35.88	0	67.49
	3	56.32	0	66.36
	3	68.29	3	91.92
3	0	36.97	2	83.51
	1	42.85	1	83.93
	0	32.69	1	65.69
	0	31.09	1	78.74
	5	59.23	0	76.28
	9	109	0	71.65
	3	64.51	5	104.07
Average	1.333	44.525	1.381	83.49

CONCLUSION

In the present researcher, a BCI based home automation system using user's eye blinks was developed. The system has shown that it requires a low amount of training and possess a fast response time. Experimental results showed that users were capable of using the system with an acceptable grade of error and a low amount of time to complete a task. Considering that the system was created for people with disabilities, the system could give back a certain level of independence to those people that suffer from movement related disabilities.

LIMITATIONS

Even when the proposed system was successfully implemented, the testing phase revealed some minor limitations and areas for future improvement. The most important limitation is related to the system scalability, it was designed to scale up to a large number of devices without any problem but it has to be mentioned that usability will decrease since navigate between menus and operation commands will become harder and exhausting.

Another limitation was system's network dependency. As shown in table IV, the Rasberry Pi 3 B+ network capabilities proved to be inefficient and unstable but it could be easily solved by using a more reliable SBC. Furthermore, the system needs the implementation of an MQTT broker infrastructure which involves an additional infrastructure to manage. The system could use a third-party MQTT provider as it was done in this researcher but this could increment the network delay.

REFERENCES

Allafi, I. and T. Iqbal, 2017. Design and implementation of a low cost web server using ESP32 for real-time photovoltaic system monitoring. Proceedings of the 2017 IEEE Electrical Power and Energy Conference (EPEC'17), October 22-25, 2017, IEEE, Saskatoon, Canada, pp: 1-5.

Alrajhi, W., D. Alaloola and A. Albarqawi, 2017. Smart home: Toward daily use of BCI-based systems. Proceedings of the 2017 International Conference on Informatics, Health & Technology (ICIHT'17), February 21-23, 2017, IEEE, Riyadh, Saudi Arabia, pp: 1-5.

Bhemjibhaih, D.P., G.D. Sanjay, V. Sreejith and B. Prakash, 2008. Brain-computer interface based home automation system for paralysed people. Proceedings of the 2018 IEEE International Conference on Recent Advances in Intelligent Computational Systems (RAICS'18), December 6-8, 2018, IEEE, Thiruvananthapuram, India, pp: 230-233.

Birbaumer, N., 1999. Slow cortical potentials: Plasticity, operant control and behavioral effects. Neuroscientist, 5: 74-78.

Hardkernel, 2019. ODROID-XU4 special price. Hardkernel Company, South Korea. https://www.hardkernel.com/shop/ODROID-xu4-special-price/

Klem, G.H., H.O. Luders, H.H. Jasper and C. Elger, 1999. The ten- twenty electrode system of the International Federation: The International Federation of clinical neurophysiology. Electroencephalogr Clin Neurophysiol, 52: 3-6.

Lopes, N.V., F. Pinto, P. Furtado and J. Silva, 2014. IoT architecture proposal for disabled people. Proceedings of the 2014 IEEE 10th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob'14), October 8-10, 2014, IEEE, Larnaca, Cyprus, pp: 152-158.

NIH., 2012. Motor neuron diseases. National Institutes of Health, Bethesda, Maryland, USA. https://catalog.ninds.nih.gov/ninds/product/Motor-Neuron-Diseases/12-5371

OpenBCI, 2019. Open source hardware certified. OpenBCI, New York, USA. https://openbci.com/

RPF., 2019. Raspberry Pi 3 Model B+. Raspberry Pi Foundation, UK. https://www.raspberrypi.org/ products/raspberry-pi-3-model-b-plus/

Shi, K., N. Gao, Q. Li and O. Bai, 2017. A P300 brain-computer interface design for virtual remote control system. Proceedings of the 2017 3rd IEEE International Conference on Control Science and Systems Engineering (ICCSSE'17), August 17-19, 2017, IEEE, Beijing, China, pp: 326-329.

WHO. and WB., 2011. World Report on Disability. World Health Organization Press, Geneva, Switzerland, ISBN:9789241564182, Pages: 325.

Zavala, S.P., J.L.L. Bayas, A. Ulloa, J. Sulca, J.L.M. Lopezm and S.G. Yoo, 2018. Brain computer interface application for people with movement disabilities. Proceedings of the International Conference on Human Centered Computing (HCC'18), December 5-7, 2018, Springer, Cham, Switzerland, pp: 35-47.