



ARCHIVIO ISTITUZIONALE DELLA RICERCA

Alma Mater Studiorum Università di Bologna Archivio istituzionale della ricerca

Creamino: a Cost-Effective, Open-Source EEG-based BCI System

This is the final peer-reviewed author's accepted manuscript (postprint) of the following publication:

Published Version:

Creamino: a Cost-Effective, Open-Source EEG-based BCI System / Chiesi, Matteo; Guermandi, Marco; Placati, Silvio; Franchi Scarselli, Eleonora; Guerrieri, Roberto. - In: IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING. - ISSN 0018-9294. - STAMPA. - 66:4(2019), pp. 8424880.900-8424880.909. [10.1109/TBME.2018.2863198]

This version is available at: <https://hdl.handle.net/11585/661710> since: 2019-04-17

Published:

DOI: <http://doi.org/10.1109/TBME.2018.2863198>

Terms of use:

Some rights reserved. The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

(Article begins on next page)

This item was downloaded from IRIS Università di Bologna (<https://cris.unibo.it/>).
When citing, please refer to the published version.

This is the final peer-reviewed accepted manuscript of:

*M. Chiesi, M. Guermandi, S. Placati, E. F. Scarselli and R. Guerrieri, "Creamino: **A Cost-Effective, Open-Source EEG-Based BCI System**" in IEEE Transactions on Biomedical Engineering, vol. 66, no. 4, pp. 900-909, April 2019*

The final published version is available online at:

<https://doi.org/10.1109/TBME.2018.2863198>

Rights / License:

The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (<https://cris.unibo.it/>)

When citing, please refer to the published version.

Creamino: a Cost-Effective, Open-Source EEG-based BCI System

Matteo Chiesi, Marco Guermandi, Silvio Placati,
Eleonora Franchi Scarselli, *Member, IEEE*, and Roberto Guerrieri

Abstract—This paper presents an open source framework called *Creamino*. It consists of an Arduino-based cost-effective quick-setup EEG platform built with off-the shelf components and a set of software modules that easily allow users to connect this system to Simulink or BCI-oriented tools (such as BCI2000 or OpenViBE) and set up a wide number of neuroscientific experiments. *Creamino* is capable of processing multiple EEG channels in real-time and operates under Windows, Linux and Mac OS X in real-time on a standard PC. Its objective is to provide a system that can be readily fabricated and used for neurophysiological experiments and, at the same time, can serve as the basis for development of novel BCI platforms by accessing and modifying its open source hardware and software libraries. Schematics, gerber files, bill of materials, source code, software modules, demonstration videos and instructions on how to use these modules are available free of charge for research and educational purposes online at <https://github.com/ArcesUnibo/creamino>. Application cases show how the system can be used for neuroscientific or BCI experiments. Thanks to its low production cost and its compatibility with open-source BCI tools, the system presented is particularly suitable for use in BCI research and educational applications.

Index Terms—Electroencephalography (EEG), Brain Computer Interface (BCI), Rapid prototyping, Open-Source, Hybrid Hardware/Software Platform, Real time Acquisition.

I. INTRODUCTION

ELECTROENCEPHALOGRAPHY (EEG) is an electro-physiological monitoring method used to record the electrical activity of the brain by means of a set of electrodes placed along the scalp of the subject. EEG has been used in the clinical context to diagnose diseases such as epilepsy, sleep disorder, coma, etc., and most recently its applications have included Brain-Computer Interface (BCI), neurofeedback and rehabilitation [1] [2].

The last few years have seen a blooming development of hardware and software solutions for BCI, from innovative electrodes [3], to novel stimuli presentation methods [4], to new algorithms and software [5] and new applications. All these works rely on signal acquisition and processing frameworks which either build upon custom made systems for allowing maximum flexibility in accessing and modifying internal blocks, or on commercial systems to speed up

development. Commercial systems have hardware (and often software) which is non openly accessible by the user and can not be modified according to specific needs. This may be a significant issue for research groups developing new BCI systems [6]. Their cost can be a major disadvantage too, often limiting the number of devices which can be acquired and therefore the ability to perform tests running for more than a few hours and on more than 10/20 subjects.

The main issue of EEG systems developed by research groups [2], [7]–[12] or technology enthusiasts (e.g. HackEEG [13]) is the lack of detailed hardware descriptions (down to schematics and bills of materials) and of a complete software suite for running neurophysiological experiments. They rely on home-made C++ or Matlab functions to visualize, store and process EEG signals, while more specific software for BCI are not available. There are no standard libraries that allow one to connect them to the previously described BCI software tools. OpenBCI is an open-source system which overcomes many of these limitations. From the hardware standpoint, the main limitation of OpenBCI is the use of traditional, passive electrode, which increases the setup time with respect to state of the art systems based on active electrodes of, at least, 10 times [14], [15]. This also hampers the possibility of effectively using dry electrodes to acquire high quality EEG signals [16], which are the basis for allowing these kind of devices to be used at home. From a software standpoint, drivers for integrating OpenBCI [17] with the above mentioned software platforms are only provided by third parties, and access to source code and documentation is fragmented, non integrated and partially incomplete. The lack of complete support for synchronization with stimulus presentation on PC also complicates its use for event-related potential experiments and several other neuroscientific research tests [18]. To address these issues an open-source framework called *Creamino* was developed. It consists of an Arduino-based cost-effective quick-set up EEG system and a set of software modules that easily allow users to set up neuroscientific or BCI experiments using open-source software tools. *Creamino* allows users to acquire, visualize and process EEG signals in real time and convert these signals into commands that operate a device. It has a fabrication cost of about \$80 for the first 8 channels (working system, including wet-contact active electrodes) and \$30 for each of the additional 8 channels. These numbers are particularly attractive for systems designed to be used outside clinical environments, such as in home care or research-oriented applications. The main objective is two-fold. First, to provide a complete HW/SW suite with state of the art performance, minimizing both

This work was supported in part by the CREAM project. The CREAM project received funding from the European Union under the Seventh Framework Programme, grant agreement 612022.

The authors are with the Advanced Research Center on Electronic Systems (ARCES), University of Bologna, Bologna 40123, Italy (e-mail: {matteo.chiesi, marco.guermandi, silvio.placati, eleonora.franchi, roberto.guerrieri}@unibo.it)

Manuscript received April 19, 2005; revised August 26, 2015.

fabrication and development time and cost. Secondly, it can still be modified at any level in order to suit several possible research applications. These can range from the testing of new electrodes, to new data processing tools/software which can rely on documented interfaces, to BCI applications/algorithms developed under existing free software suites like OpenVIBE [19] or BCI2000 [20], to neuroscientific research which relies on MATLAB based tools like Psychtoolbox or Cogent [21]. From a software standpoint, examples of these platforms are BCI2000, OpenVIBE and BioSig [22]. In order to provide a consistent platform, the contribution of this work is:

- Design (schematics, gerber files, bill of materials and firmware) of a cost-effective quick-setup EEG acquisition system, comparable in performances to its clinical counterparts;
- A Java-based custom-built acquisition software with a graphical user interface (GUI) that can be used to acquire, filter, process, classify, visualize and store up to 64 EEG channels in real time;
- A set of software modules running on a standard PC which allows one to connect the system to BCI2000, OpenViBE, MATLAB and SIMULINK;
- Detailed instructions on how to compile (if needed), modify and use the software;
- A set of experiments carried out using *Creamino* and showing the usefulness of the platform.

Creamino, with executables, a source code and instructions, is available free of charge for research and educational purposes at <https://github.com/ArcesUnibo/creamino>. The rest of the paper is organized as follows. Section II describes the hardware and the firmware implementation of *Creamino*, including electrical and functional characterization. Section III discusses the standalone acquisition software and the libraries used to import and process EEG signals in BCI software tools. Application cases are presented in Section IV. Section V contains the discussion and some conclusions are drawn in Section VI.

II. CREAMINO-BASED EEG-BCI SYSTEM

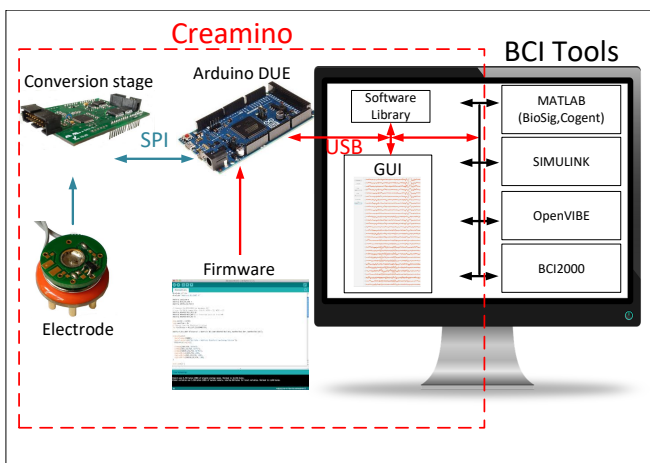


Fig. 1. Overview of a Creamino-based EEG acquisition system.

Fig. 1 shows an overview of a *Creamino*-based BCI system: it consists of a set of active electrodes for acquiring EEG signals connected to a board which mounts an AFE that performs signal amplification and analog-to-digital conversion. For human safety, isolation circuitry is used in supplying AFEs with power. The board is plugged in on top of an Arduino DUE system. Arduino recovers the EEG signals from the Serial Peripheral Interface (SPI) and sends them to a PC via a USB by means of a dedicated Java-based acquisition software or software libraries which allow one to filter and process these signals using Simulink or BCI software tools. The next sections describe the hardware and software modules in detail.

A. System Specifications

Specifications for EEG signal quality can be derived from the International Federation of Clinical Neurophysiology (IFCN)'s standards [23] which are commonly adopted in clinical practice, and from the IEC 60601 series of technical standards for the safety and essential performance of medical electrical equipment [24]. According to the first, acquisition must be available for at least 24 channels. For acquisition and for storage of EEG data the minimum digital sampling rate is 200 samples/s. Preamplifier input impedance should be higher than 100 M Ω and additional noise introduced in the recording should be less than 0.5 μ V root-mean-square at any frequency from 0.5 to 100 Hz including 50–60 Hz. The main specifications which can be derived from IEC60601 standards for type BF parts (body floating equipment which makes physical contact with the patient) are in terms of isolation (3000 Vac), maximum patient auxiliary current (10 μ A in normal conditions, 50 μ A in a single fault condition), CMRR (approximately 95 dB when considering a complete mismatch of contact impedance on the different electrodes [25]).

B. Hardware Architecture

The acquisition system is depicted in Fig. 2. The hardware presented is a fully functional EEG acquisition system which includes unity-gain active electrodes (for quick setup with zero-preparation wet or dry electrodes) and A/D conversion based on TI ADS1298 Analog Front End (AFE). This AFE mostly targets ECG signal acquisition but, thanks to 24-bit resolution and input programmable gain amplifiers, it can be used for EEG acquisition as well [25], with significant advantages in terms of cost and with a reduction in power supply rails and power consumption with respect to AFEs specific for EEG signal acquisition (e.g. T.I. ADS1299). It should be pointed out that modifications to the hardware presented (e.g. for use with passive electrodes, inclusion of circuits for common mode rejection improvements such as DRL and DgRL [25]) are possible without the need for modifications to the Creamino framework, as long as the interface with Arduino DUE is preserved.

Signal buffering is performed on the active electrode by a low-power, low-noise, rail-to-rail Operational Amplifier (O.A.) connected as a unity-gain buffer (OAE in Fig. 2). Protection resistors with $68\text{ K}\Omega$ are used to limit patient auxiliary current

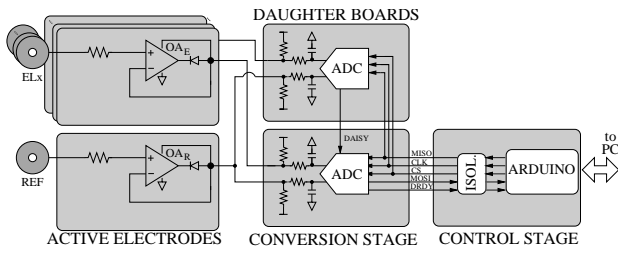


Fig. 2. Simplified representation of the acquisition system developed. MISO, CLK, CS, MOSI and RDY are SPI interface signals.

in cases of single fault condition below the applicable limit of $50 \mu\text{A}$. The O.A. is an OPA378 from Texas Instruments, which has a quiescent current of $125 \mu\text{A}$ and low voltage noise, with no significant increase at low frequency. The input leakage current is below 1 nA and low enough to guarantee very low values of patient auxiliary currents, while common mode input capacitance is below 5 pF , which translates into an input impedance in excess of $300 \text{ M}\Omega$ in the EEG band. To minimize the number of wires, the output signal of the active electrode is used for providing power supply to the O.A. as well [2]. A forward-biased diode connects the output pin of the O.A. to the power supply. The output is biased by the following conversion stage which entails a $2.2 \text{ k}\Omega$ resistor toward a 3.3 V power supply. The DC non-inverting input and output of the buffer can swing between 2.2 V , which is the minimum power supply voltage of the O.A. and 3 V which enables the O.A. to run on the supply current of $125 \mu\text{A}$. This voltage range is acceptable at the rail-to-rail input of the O.A. and is sufficient to account for DC offset from either wet or dry electrodes. As the O.A. output can not swing up to the positive rail, a forward-biased diode between the output and the positive supply guarantees that the voltage difference between the two nodes to be around 600 mV . The gain of the active electrode is fixed to one, so as to limit degradation of CMRR, as described in [25]. Active electrodes with higher gains can be used, provided that solutions to improve the CMRR are adopted, such as Driving Right Leg circuits [25], as described, e.g., in [2]. For generality's sake, the schematics provided and layouts for the active electrode enable one to adopt both solutions by choosing appropriate passive components. The board presents a gold ring which allows its direct use for wet contact, without the need for an external electrode. For use with dry-electrodes, an electrode specific connector can be soldered to the ring (for g. SAHARA electrodes a simple snap fastener).

The Analog/Digital conversion is based on a low-power, 8-channel, 24-bit AFE for biopotential measurements (TI ADS1298 [26]). 2-wire ribbon cables connect the active electrode to the conversion stage, with one wire carrying the ground connection and the other for both power supply and active electrode output as shown in Fig 2. The electrode signals are low-pass filtered by a passive RC-filter with a 16 kHz bandwidth and sent to the non-inverting inputs of the ADC. Inverting inputs are all connected to the output of the active reference electrode. ADCs can be daisy-chained to increase the number of EEG channels by piling up daughter boards on top of the main conversion board and on top of each other (see

Fig. 3). A schematic and layout for daughter boards extending the channel number by 8 each are also provided.

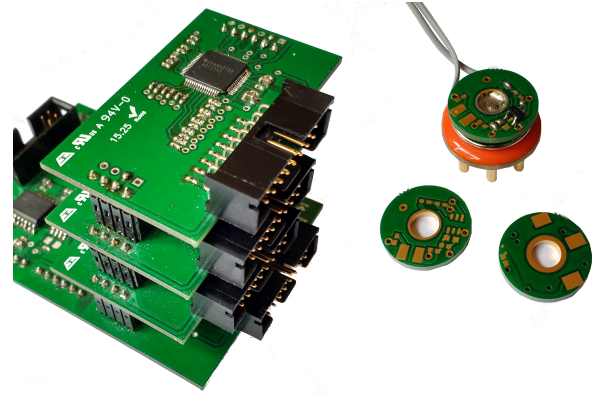


Fig. 3. On the left: detail of Creamino with the main conversion board and 3 daughter boards for 32-channel EEG acquisition. On the right: active electrode boards with gold ring for direct wet contact with the skin and attached to a commercial dry electrode.

The back-end unit exchanges data between the AFEs and the PC and provides power supply to the conversion stage and the active electrodes. For simplicity's sake the board that performs the conversion stage is plugged in on top of an Arduino board (<https://www.arduino.cc/>) which carries power supply and ground, plus the five digital signals for SPI data transfer through isolators. Isolation, according to IEC60601 specifications, is performed through Analog Devices ADUM 6000 for power supply and two ADUM 2401 for SPI signals which guarantee isolations in excess of 5000 V . For legacy purposes, the presented hardware can be used with both Arduino UNO and Arduino DUE board, although firmware and software is provided and discussed here only for the vesion Arduino DUE, since it presents two separate USB ports which, as will be clarified later in the paper, are required for transferring trigger signals for, e.g., synchronization with stimulus presentation in BCI tools. The schematic and layout of an adapter board are provided to allow Creamino to be connected to Arduino DUE boards. A jumper allows one to set SPI voltage level at 3.3 V (for Arduino DUE) or 5 V (for Arduino UNO).

Creamino is based on Arduino since it is an open-source easy-to-use prototyping platform which allows non-professional users to easily modify the firmware according to their needs using the Arduino Integrated Development Environment (IDE). Arduino can be connected to a PC via a USB, but additional shields to perform wi-fi or Bluetooth communication are also available. It appears to a PC as a virtual COM port that applications can access using COM-port application program interfaces (APIs) or libraries [27]. The COM-port software interface provides a way for PCs to exchange data with the EEG-system.

Arduino provides COM port drivers that can be used on Windows, Linux and Mac OS X operating systems, making the system platform independent. In addition Arduino DUE is equipped with two USB ports. This makes it possible to carry out neuroscientific experiments where EEG signals must be

inherently time-synchronized with visual and auditory stimuli without the need for any additional hardware devices. One USB port is managed by the software used to transfer the EEG signals to a PC while the other is managed by the software that creates the stimulus and is used to send a trigger together with the stimulus. This trigger is combined with EEG signals (acquired in the same time frame) and sent to the PC that performs the acquisition as explained in detail in Section IV-A.

C. Electrical and functional characterization

Noise has been measured by shorting all active electrode inputs to the ground electrode. Fig. 4 presents results at varying values for the Programmable Gain Amplifier (PGA) gain of the ADS1298 AFE. Integrated noise between 0.5 and 100 Hz is $2.3 \mu\text{V}_{RMS}$ (gain=1), $0.7 \mu\text{V}_{RMS}$ (gain=6), $0.55 \mu\text{V}_{RMS}$ (gain=12). It should be pointed out that the highest gain still allows for an input range of ± 250 mV, which is enough to accommodate for electrode offset and drifts.

The Common Mode Rejection Ratio was measured according

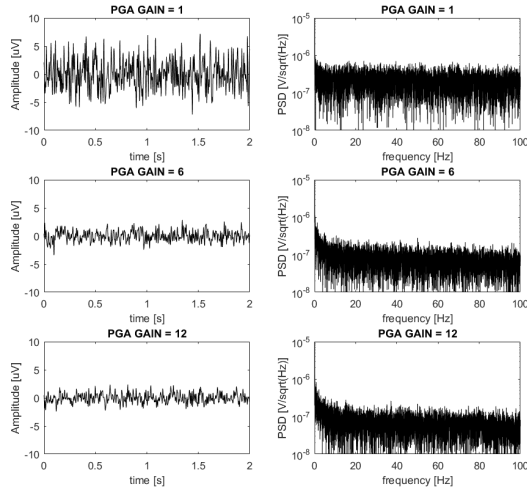


Fig. 4. Noise characterization with input shorted, for different gain values of the ADS1298 programmable gain amplifier. On the left, time domain signal. On the right, power spectral density.

to requirements by the IEC60601 standards family. It prescribes that a 10 V_{RMS} signal at mains frequency (50 Hz) with 200 pF source capacitance, connected between earth and all lead wires connected together, shall not produce an output signal greater than 1 mV peak-to-valley. This translates into a value of 95 dB. In series with each electrode there shall be a 51 K Ω resistor in parallel with a 47 nF capacitor to emulate contact impedance. Compliance is checked by shorting such impedance on every electrode, one-by-one, in order to simulate a complete imbalance in the contact impedance.

Creamino suite allows for calibrating the gain of each electrode by multiplying the measured signal by a user configurable parameter in the GUI, as will be presented in the next section. This value can easily be calculated by shorting every electrode input to an AC signal source in the EEG band. Ideally, the signal would be at a frequency close to 50 Hz and

Resolution	24 bits
Number of Channels	8 to 64
Sampling Rate	250 to 32K SPS
Input Range	± 250 mV
NOISE [0.5-100] Hz	$0.55 \mu\text{V}_{RMS}$
CMRR (w/o calibration)	98 dB
CMRR (w/ calibration)	108 dB

TABLE I
PERFORMANCES OF CREAMINO FOR PGA GAIN SET TO 12.

an amplitude in the order of few tens of mV. The calibration factor for each channel can easily be calculated by computing the ratio between the signal amplitude on each channel. Native CMRR is measured at 98 dB and already satisfies the relevant standard requirement. Additional 10 dB can be obtained after calibration.

Fig. 5 shows the functionality in the acquisition of EEG signals, using commercial available dry electrodes. We tested two different types of electrode, namely g.SAHARA from g.tec GmbH [28] which are used in high-end commercial systems (left column) and cheap (less than 1\$ per electrode) reusable TDE-200 electrodes from Florida Research Instruments Inc. (right column) [29]. One EEG channel is shown, acquired in the occipital region of the right hemisphere; alpha waves at approximately 12 Hz are visible when the subject has his eyes closed at time intervals of 20 to 40 seconds and 60 to 80 seconds. Signal quality is very similar for both type of electrode.

D. Creamino Firmware Implementation

Figure 6 shows the flow chart of the data acquisition process. It is executed by the embedded MCU (AT91SAM3X8E [30]) mounted on the Arduino board. The firmware implementing the process was designed using an open-source Arduino IDE.

IDE is a simplified environment which runs on Windows, Mac OS X and Linux and makes it easy to write code and upload it to the board via a USB. The firmware was written to manage communication with one or more ADS1298 AFEs [26], but can easily be modified to work with other AFEs or include other functions. The communication protocol realized is described below. Via a PC software module (one of those listed in the following Section) the user needs to select the setup parameters depending on the number of EEG channels and the AFE used. A handshaking protocol is implemented to connect the PC to Arduino using a virtual COM port over USB. The PC sends a *synchronize* message, and Arduino replies with a *synchronize-acknowledgement* message (should the communication not be successful, an error message will be displayed and the procedure must be repeated). At this point the USB communication is opened and Arduino awaits the parameters needed to set up the acquisition system. Then, via SPI it configures the analog front-end and waits for generation of an external signal from the AFE signifying that the first set of data are ready to be transferred.

In order to verify that no data are lost during transmission a *synchronize* word is sent and checked at the end of each

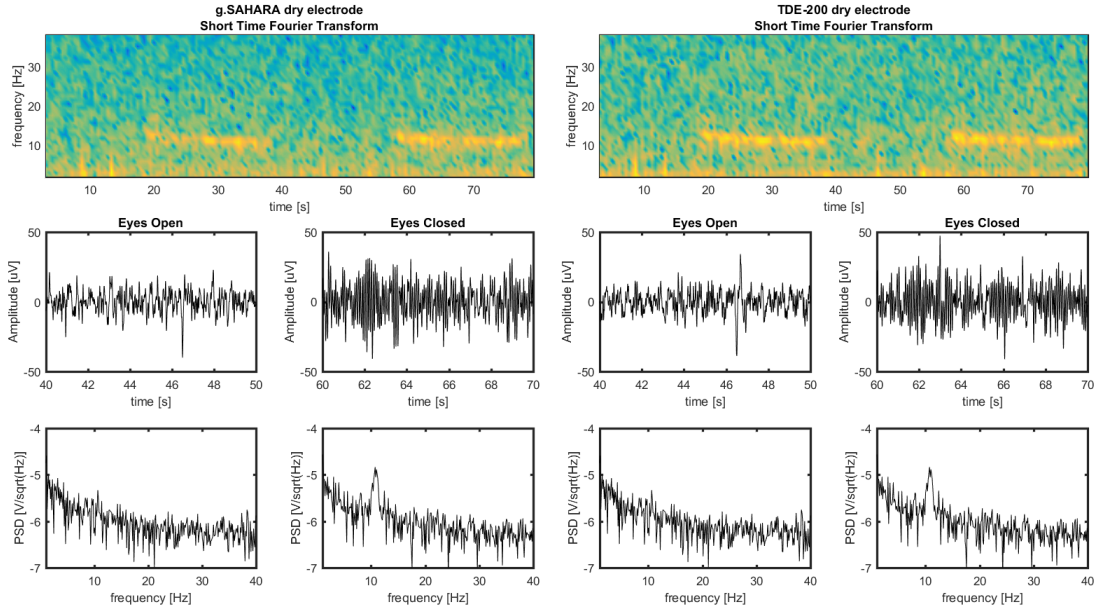


Fig. 5. The EEG signal is acquired in the right occipital position O2, with reference on Fp2. Alpha waves are visible when the subject has his eyes closed at time intervals of 20 to 40 seconds and 60 to 80 seconds (lighter tones correspond to higher power spectral density in the short time Fourier transform graph and peaks are visible in the PSD computed in the eyes-closed condition).

packet transfer. Once the session is completed the user stops acquisition and closes communication over the COM Port. The next Section describes the software and the libraries provided with a view to linking together BCI tools and an EEG system.

III. CREAMINO SOFTWARE MODULES

Three software libraries enabling one to connect *Creamino* to BCI2000, OpenViBE and Matlab & Simulink (The MathWorks, Inc., Natick, Massachusetts, U.S.) and a standalone Java-based acquisition software were developed in order to acquire and elaborate EEG signals through a PC. All these modules rely on four basic software functions that allow the user to perform the following operations:

- 1) Configure the dimension of the buffer which will store the EEG data;
- 2) Configure the EEG system and start acquisition from the AFE to Arduino via SPI;
- 3) Stop acquisition and de-initialize the system;
- 4) Wait until the specified amount of data has been acquired via USB to a PC, and then return a pointer to a buffer containing the data in its first argument.

All the software allows users to run-time change the functionality of the system (such as filter type, number of channels visualized, etc.). Specifically, before starting acquisition the software can be used to:

- select the sample rate used to acquire EEG signals;
- select the number of channels depending on the number of AFEs connected in daisy-chain mode.

During acquisition the main functionalities of the software are:

- Real time data visualization;

- Data storage in binary format (tools are also available for easy importing into a MATLAB environment for further postprocessing);
- Referencing (towards one channel or the average of two or more channels);
- Modifying calibration factors;
- Modifying channel montage and naming;
- Modifying filter options (low-pass with different cut-off frequencies);
- Notch filter for network interference removal.

A. BCI2000 driver

BCI2000 can be used to measure specific features of brain activity and translate them into device control signals. In order to connect a *Creamino*-based system to BCI2000 it was necessary to develop a new source module including the functions listed above, together with some helper functions to perform initialization and clean-up tasks.

The driver as developed uses the functions of the `<Windows.h>` library in order to exchange data over the COM port under Windows OS. The functions developed were included in the BCI2000 source project. The entire project was then compiled and linked using Microsoft Visual Studio 2010 (MVS - <https://www.visualstudio.com/>) following the instructions given (see also <http://www.bci2000.org/wiki/>).

B. OpenViBE driver

OpenViBE consists of a set of software modules devoted to acquisition, preprocessing, processing and visualization of cerebral data as well as to interacting with virtual reality displays.

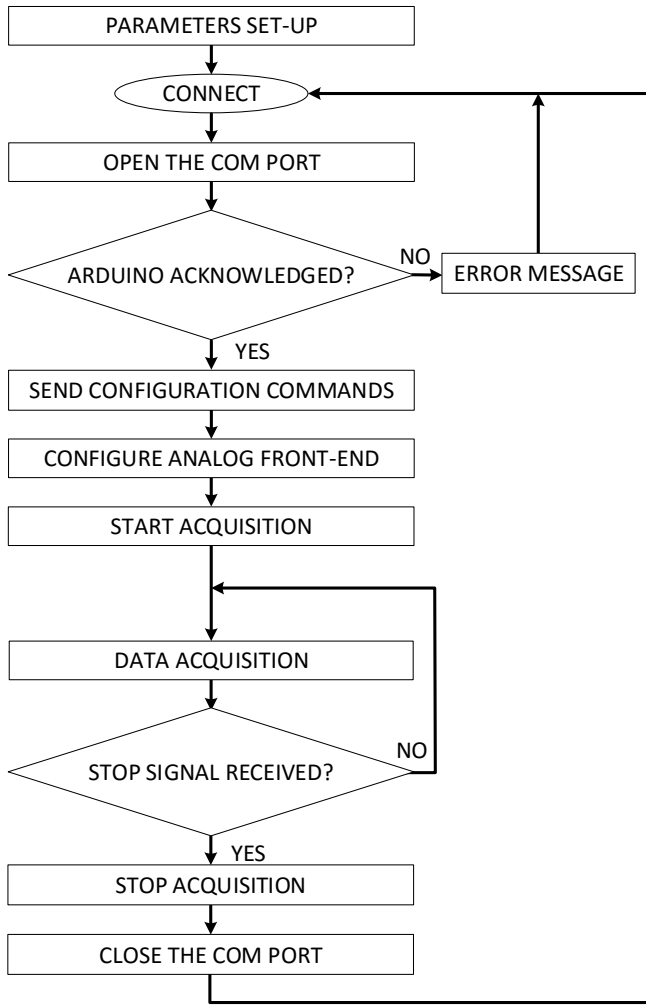


Fig. 6. Creamino Control Flow Diagram

In order to connect *Creamino* to OpenViBE a new acquisition driver was created. The driver consists in a single object capable of communicating with *Creamino* and correctly sampling data. As in the case of BCI2000, the driver performs the same four separate stages of execution. Starting from the same functions as used to connect *Creamino* to BCI2000, a C++ source file was developed and compiled using MVS 2010. The acquisition driver was developed using the Skeleton-generator developer tool provided by OpenViBE and the custom library was linked to the files generated and used to manage communication with the hardware. The entire project was compiled and linked using MVS 2010. Finally, a dedicated GUI was set up so that the user might configure the driver.

C. Simulink block

A user-defined block allowing one to import signals coming from *Creamino* directly into Simulink was developed. This block makes it possible to manage communication with the hardware and to view simulation results live with the same features as described for the other software modules inside a Simulink environment. In addition, real time analysis of data collected by *Creamino* can be performed by adding other

Simulink blocks. The user-defined block was written starting from the template of a level 2 Matlab S-function which allows one to use Matlab language to create a custom block. Since Matlab language is interpreted and not compiled for a specific target architecture, the block can be executed on platforms running Simulink and equipped with Microsoft Windows, Linux or MAC OS X without any need for re-compilation. The block developed has one output port capable of handling the EEG signal, including all the features to manage the COM Port inside the block itself. Simulation parameters can be set by means of a dedicated GUI. The block was built as a discrete block so that it produces an output at specific points in time, and can be configured to support different sample rates. All software is available for reading and editing in the GitHub repository. A video is also available, showing how to set up and use Simulink blocks.

D. Java-based Acquisition Software

Real time signal acquisition and processing can be performed by connecting *Creamino* to a PC and using the Java-based acquisition software developed. The software was written in Java, while the GUI was designed using JavaFx. This allows users to employ the software on multiple platforms (Microsoft Windows, Linux and Mac OS X operating systems) while keeping the same graphic outlay, level of visualization and operation. All software is available for reading and editing in the GitHub repository. A video is also available, showing how to set up and use the interface.

IV. APPLICATION CASES

Creamino allows one to realize a complete platform that can be employed for any kind of experiment ranging from neurofeedback to BCI applications. In this section three examples of possible applications are described and discussed. All these experiments were carried out using a laptop *Lenovo Y50* equipped with an Intel i7-4710HQ processor, 16 GB of RAM and a Nvidia GeForce GTX 960M graphics card. The setup parameters which allow *Creamino* to configure the AFE are set at run-time before starting each experiment. Contact with the patient's skin is obtained by means of wet-contact electrodes which are kept in place by a standard EEG cap.

A. Event Related Potentials experiment

This neurofeedback experiment was set up and carried out to prove the functionality of the Java-based acquisition software presented in Section III-D and the trigger strategy implemented. In particular, *Creamino* was used to acquire multi-trial Event Related Potentials (ERP) obtained from visual stimuli. In ERP experiments, in order to achieve clear signal characterization and allow data analysis consistent with the related potential event, EEG data must be inherently time-synchronized with visual and auditory stimuli (i.e. the visual stimulus provided must be correlated to the corresponding patient response detected by the hardware at the same point in time).

Synchronization is accomplished by sending a trigger

through USB2 port to *Creamino* whenever the stimulus occurs, directly upon a command from the software used to create the stimulus. This experiment used Cogent (<http://www.vislab.ucl.ac.uk/Cogent>), an open-source toolbox for Matlab. Directly from the Matlab environment the COM port was opened and used to send this trigger. *Creamino* wrapped the trigger together with the EEG signals and sent them to the PC via USB1 port. Java-based software was used to perform the acquisition of EEG signals and save these data together with trigger signals in a binary format. Then the data were post-processed using EEGLab [31], an interactive Matlab toolbox for processing event-related EEG data. The setup of the experiment is depicted in Fig. 7.

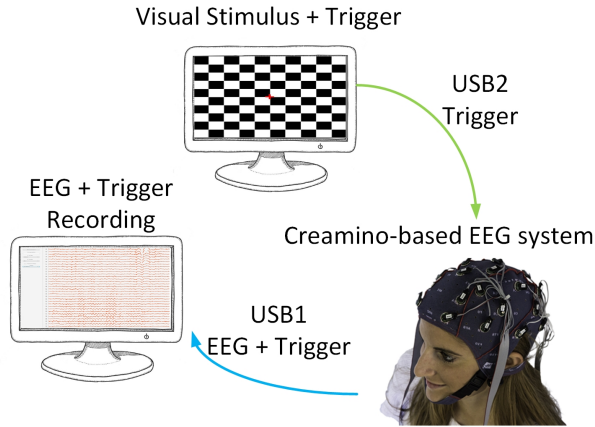
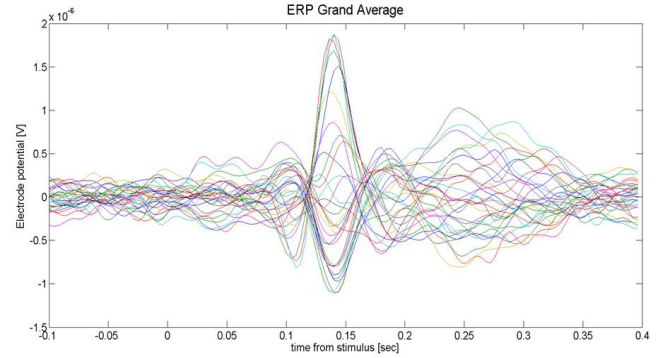


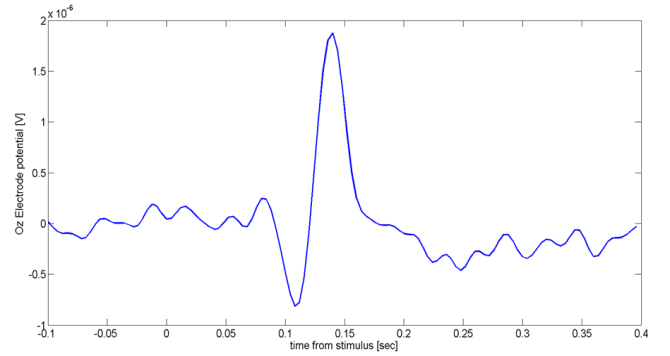
Fig. 7. Simplified representation of the triggering strategy.

Seven volunteers (male, age range 21 to 34) participated in ERP recordings, chosen from among the researchers and students at the ARCES research center, University of Bologna. The stimuli were presented on a 24-inch LCD panel located approximately 70 cm from the subject. A square 8x8 black and white contrast checker-board was presented, flickering at an approximate rate of 2 Hz. Three possible positions of the checker-board on the screen were considered, one centered with respect to the subject's visual field, one slightly displaced to the left, one to the right. The objective was to stimulate the left and right visual fields symmetrically or asymmetrically [32]. The flickering was repeated 22 times per sequence at an interval of random duration comprised between 500 and 650 ms. A random inter-stimulus interval (ISI) of 3 to 5 seconds was inserted between each sequence in order to allow the subject to blink. For maximum speed of set up time, the system allocated 32 wet-electrodes (4X ADS1298 AFEs) positioned in an extended 10-20 system and the EEG signal was sampled at 500 Hz. Data were imported into the Matlab environment and processed by the EEGLAB toolbox. EEG signals were digitally filtered between 1.5 and 48 Hz in order to remove some low frequency noise. Epochs relative to the three different checker-board pattern positions were aligned and averaged for each subject after removal by eye-inspection of any epochs affected by artefacts. Grand-averages among the different subjects were finally computed. These are presented in Fig. 8, where the P100, N150/155 and P300 responses are visible in different regions of the scalp. Comparison with

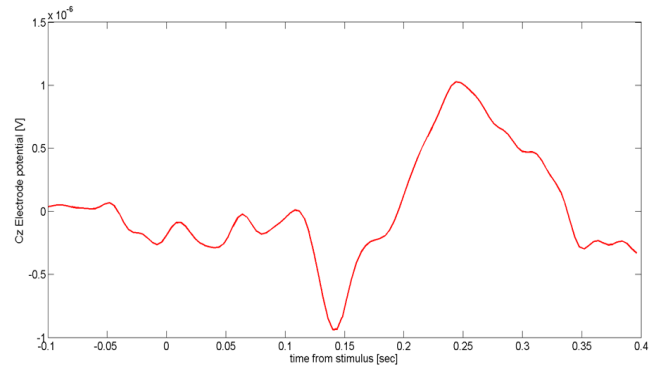
typical responses [32] allows us to confirm the ability of *Creamino* to detect ERP correctly.



(a) Grand average of ERP responses for right visual field stimulation in 32 electrodes



(b) Grand average of ERP responses on Oz. Positive P100 deflection is clearly visible at approximately 105 ms. Negative N150/155 is visible at 145 ms latency.



(c) Grand average of ERP responses on Cz. Positive P100 deflection is almost negligible. Negative N150/155 is visible at 145 ms latency (inverted with respect to Oz). P300 is visible due to processing of the information in the central region.

Fig. 8. Grand average of ERP responses. a) 32 channels, b) on Oz, c) on Cz. Graphs were obtained using EEGLab

B. EEG Source Localization

Analysis of EEG sources has been proved to be useful in allowing to develop new tools for BCI, e.g. for motor imagery classification [33] [34]. In this test case, we show how this real time data-intensive computing application can be performed

using *Creamino* and the Simulink library developed. In particular, the system was used to measure the electric potentials due to impressed current density distribution which is generated by neuronal post synaptic activity in the cortex of the brain. By exploiting electric potentials measured on the scalp, it is possible to generate a map of the electrical activity in the brain, solving the inverse problem [35]. In particular, the specific aim of this experiment was to monitor the power of the alpha signal in a subject.

To perform this test we used a library allowing one to connect the device to Simulink and adopted an sLORETA linear estimator for EEG source imaging [36] to solve the inversion step. To manage the expensive computational cost which arises from combining the inversion step and the real time constraints, we used CUDA-based parallel implementation of the sLORETA developed in [37] which runs on GPU.

This parallel implementation is used to develop a user-defined Simulink block that loads a dynamic library generated starting from the CUDA source code of the algorithm. In this case too, the system allocates 32 wet-electrodes positioned in an extended 10-20 system and the EEG signal is sampled at 500 Hz. Gray matter voxels are parcellated in 96 different brain regions according to the Harvard-Oxford cortical and sub-cortical structural atlas [38]. Domain parcellation allows one to monitor the instantaneous power of the current density in specific regions composing the discretized domain. The Simulink model used to perform this experiment is shown in Fig. 9.

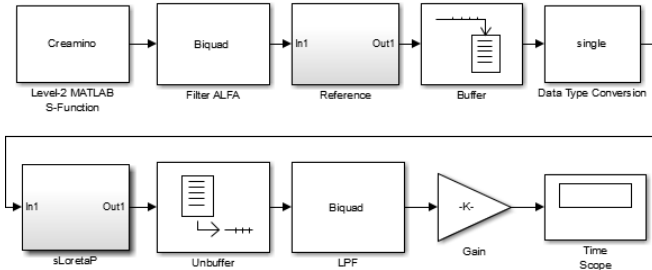


Fig. 9. Simulink model for real time alpha rhythm monitoring.

The signal was acquired using the developed Simulink model of *Creamino*, then band-pass filtered. Average referencing was performed using a standard Simulink block. A buffer was necessary to gather the input samples to be passed to the sLoreta block after conversion of the data into single precision format. A low pass filter was used to remove the high-frequency components from the output signal. Finally, the power of the region of interest (ROI) given by the parcellated blocks was normalized as a function of the number of voxels composing each region through a suitable gain. A time scope was used to visualize a 1-minute acquisition where the subject opened and closed his eyes for 30s.

The power of the alpha signal was monitored in both the occipital and the frontal region. In the real time plot obtained in the occipital ROI shown in Fig. 10 (label 48 of Harvard-Oxford cortical and sub-cortical structural atlas) an increase in power was measured when the subject closed his eyes, while

in the frontal ROI (label 1) no variations were observed. Alternatively, source reconstruction can be performed on the whole set of cortical voxels, so as to provide a 3-dimensional map of cortex activation while the subject is performing a given task. Fig. 11 shows the power densities of the localized sources in two different time frames, when the subject has eyes open and closed respectively. Activation in the occipital region is again clearly visible in the second picture.

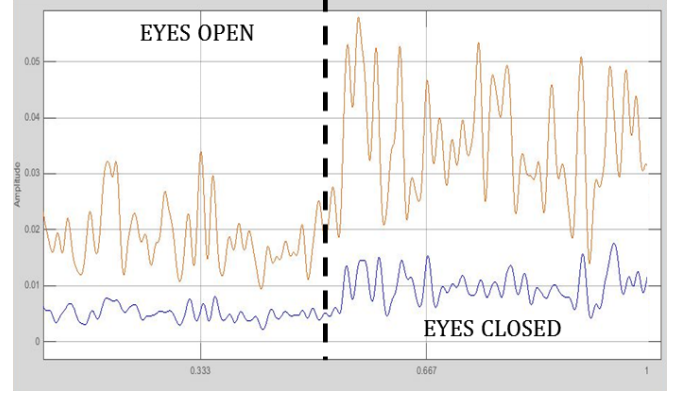


Fig. 10. Power of the alpha signal in both the occipital and the frontal region.

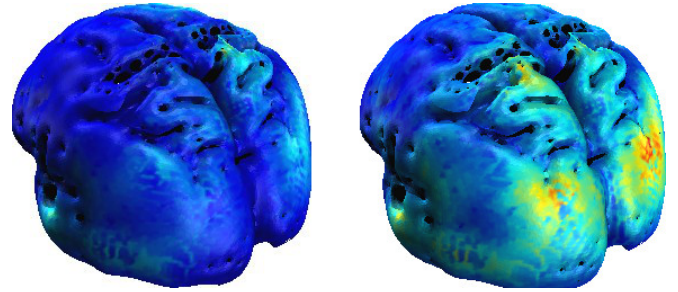


Fig. 11. 3-dimensional map of cortical activation while the subject is performing a simple eyes-open/eyes-closed task. Activation in the occipital region is clearly visible in the picture on the right (eyes-closed condition), as compared to the one on the left (eyes-open condition).

In this graph, the yellow top line represents the trend of the normalized power of the occipital ROI while the blue bottom line represents the normalized power of the frontal region. As expected, during the eyes-closed period the amount of power in the occipital ROI markedly increased as compared to the normalized power obtained in the frontal region.

C. BCI2000 and OpenViBE acquisitions

Finally, in order to provide an overview of all the libraries developed, two further acquisitions were performed connecting *Creamino* to BCI2000 and OpenViBE. Fig. 12 shows the time domain representation of the acquired EEG signal for a full EEG band from 0.5 to 100 Hz (Fig. 12a was obtained using BCI2000 while Fig. 12b was obtained using OpenViBE). *Creamino* was configured to acquire 16 channels (2x ADS1298 AFEs). The electrodes on the cap were arranged according to the 10/20 system setup, so the upper part of the graphs displays the EEG signals in the frontal region, while the lower part shows the signals in the occipital region. Both the graphs display 10s

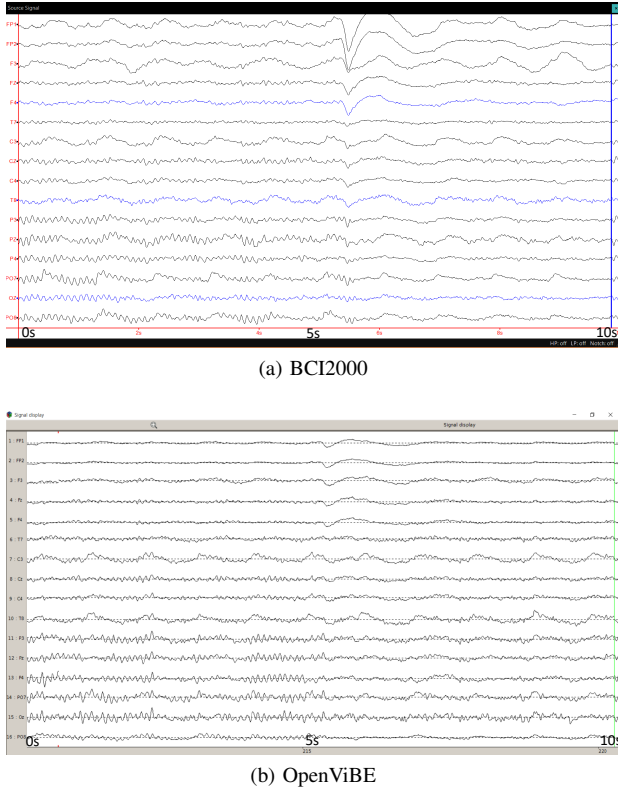


Fig. 12. Signal in the time-domain during eye transition from closed to open, band-pass filtered between 0.5 and 100 Hz (0s-5s eyes closed, 5s ->10s eyes open)

of acquisition.

In the first 5s both plots highlight an alpha rhythm at approximately 10-12 Hz frequency in the lower waveforms, taken in the occipital region when the subject kept his eyes closed. The eye movement artefact located the time instant at which the subject opened his eyes. (time equal to 5s - in the middle of both plots) After this event, the amplitude of the EEG signal decreased significantly and the alpha rhythmic oscillation reduced. These simple experiments were made to test acquisition using the libraries developed. Hence, all the modules provided with BCI2000 and OpenViBE can be configured and executed using *Creamino*.

V. DISCUSSION

BCI is changing the interaction between people and systems. Technological improvements have led to innovative solutions prompting an explosion of BCI research and technology development [39]. Current research is developing new applications that use non-invasive and rapid-setup EEG solutions which allow long-term data collection. Over the coming decades, neurotechnology will further improve: new sensing capabilities will be created and the ability for sensors to be integrated into user clothing and environmental devices will increase too [40]. *Creamino* is intended to help this evolution. Based on a low-cost, widespread, open-source hardware platform that can be reproduced for research and educational purposes, *Creamino* exploits monolithic chips that combine programmable amplification, filtering and high resolution analog to digital

conversion in order to develop powerful cost-effective EEG-based BCI systems. *Creamino* is based on Arduino, an open-source easy-to-use prototyping platform which allows non-professional users to easily modify the firmware according to their needs using the Arduino Integrated Development Environment (IDE). The Arduino platform is evolving and, as an example, the MKR1000 version has a smaller form factor than Arduino Due and supports Wi-Fi communications. This makes it possible to further reduce the size of the system and to develop miniaturized prototypes that can easily be worn all day long. As long as the prototype is still based on the same family of AFEs, only minor modifications to hardware and software will be required.

In addition, *Creamino* can be used to record Electrocardiographic (ECG) waveforms or Electromyographic (EMG) muscle signals.

The software developed in this work can be used to acquire EEG signals or to connect these platforms to some of the most widespread BCI tools. This simplifies the evolution of new BCI devices, allowing researchers to analyze neurophysiological signals in real time, or to develop applications capable of providing practical assistance for patient diagnosis, treatment, and rehabilitation [41], [42]. This paper shows how *Creamino* is capable of acquiring EEG signal with the same performances as are expected of commercial devices at a fraction of their cost and with an open, easily modifiable system. *Creamino* allows one to record up to 64 channels at 24-bit resolution and sampling rates up to 32 KSPS. Noise in the EEG band is $0.55 \mu V_{RMS}$.

The system was designed to have as few implementation dependencies as possible. Hence part of the software, specifically the Java-based acquisition software and the Simulink block, can run on all platforms without the need for recompilation. The application cases discussed in Section IV address the ability of *Creamino* to be used in a wide range of BCI and neurofeedback applications and its flexibility in supporting the most commonly used software tools.

VI. CONCLUSION

This work has presented an open-source BCI platform based on low-cost, easily available off-the shelf components enabling one to acquire, visualize, filter and process EEG signals and perform BCI experiments. This reduces the time needed to achieve a prototype and allows the non-professional user to set up a system. *Creamino* has a manufacturing cost of about \$80 for the first 8 channels and \$30 for each of the additional 8 channels. This marks an important advantage over commercial systems. It is released with a set of software modules that allow one to acquire and process EEG signals. In particular, it is capable of processing up to 64 channels in real time and operates under Windows, Linux and Mac OS X on a standard PC. Thanks to its production cost and its compatibility with open-source BCI tools, the system presented is an enabling platform for BCI research and educational applications.

APPENDIX A

AVAILABILITY OF *Creamino*

Creamino, with schematics, gerber files, bill of materials, executables and source code is published free of charge at <https://github.com/ArcesUnibo/creamino> under MIT licence. The file README contains all the instructions to download the repository. Documents on how to use and compile the software, suitable for designers as well as for end users, are provided. Demo videos are included to clarify how to use the custom Java interface and the user defined Simulink block.

REFERENCES

- [1] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain-computer interfaces for communication and control," *Clinical neurophysiology*, vol. 113, no. 6, pp. 767–791, 2002.
- [2] M. Guermandi, A. Bigucci, E. F. Scarselli, and R. Guerrieri, "EEG acquisition system based on active electrodes with common-mode interference suppression by Driving Right Leg circuit," in *2015 Int. Conf. Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2015, pp. 3169–3172.
- [3] C. Grozea, C. D. Voinescu, and S. Fazli, "Bristle-sensors—low-cost flexible passive dry eeg electrodes for neurofeedback and bci applications," *Journal of neural engineering*, vol. 8, no. 2, p. 025008, 2011.
- [4] G. Townsend, B. LaPallo, C. Boulay, D. Krusienski, G. Frye, C. Hauser, N. Schwartz, T. Vaughan, J. R. Wolpaw, and E. Sellers, "A novel p300-based brain-computer interface stimulus presentation paradigm: moving beyond rows and columns," *Clinical Neurophysiology*, vol. 121, no. 7, pp. 1109–1120, 2010.
- [5] A. Stopczynski, C. Stahlhut, J. E. Larsen, M. K. Petersen, and L. K. Hansen, "The smartphone brain scanner: a portable real-time neuroimaging system," *PloS one*, vol. 9, no. 2, p. e86733, 2014.
- [6] A. von Lüthmann, C. Herff, D. Heger, and T. Schultz, "Toward a wireless open source instrument: functional near-infrared spectroscopy in mobile neuroergonomics and bci applications," *Frontiers in human neuroscience*, vol. 9, p. 617, 2015.
- [7] A. Ghomashchi, Z. Zheng, N. Majaj, M. Trumpis, L. Kiorpes, and J. Viventi, "A low-cost, open-source, wireless electrophysiology system," in *2014 Int. Conf. Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2014, pp. 3138–3141.
- [8] A. Jain, I. Kim, and B. J. Gluckman, "Low cost electroencephalographic acquisition amplifier to serve as teaching and research tool," in *2011 Int. Conf. Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2011, pp. 1888–1891.
- [9] B. Ilie, "Portable equipment for monitoring human functional parameters," in *2010 Int. Conf. RoEduNet*. IEEE, 2010, pp. 299–302.
- [10] J. Duan, C. Chen, S. H. Pun, F. Wan, P. U. Mak, P. I. Mak, M. I. Vai, and Y. Hu, "A wearable wireless general purpose bio-signal acquisition prototype system for home healthcare," in *2012 Int. Conf. Biomed Eng Biotechnol ICBE*, 2012, pp. 1176–1179.
- [11] P. Arulmozhiarman, G. R. Reddy, and R. Tatavarti, "Low cost eeg signal acquisition system," in *2013 Conf. Biosignals and Biorobotics (BRC)*. IEEE, 2013, pp. 1–3.
- [12] M. Pflugradt, N. Hofmeyr, and R. Orglmeister, "Multi-channel biosignal processing on an omap 3530 system," in *2012 Education and Research Conference (EDERC)*. IEEE, 2012, pp. 124–128.
- [13] S. LLC. Hackeeg shield for arduino due. [Online]. Available: <https://www.starcat.io/products/hackeeg-shield/>
- [14] M. A. Lopez-Gordo, D. Sanchez-Morillo, and F. P. Valle, "Dry eeg electrodes," *Sensors*, vol. 14, no. 7, pp. 12 847–12 870, 2014.
- [15] C. Guger, G. Krausz, B. Z. Allison, and G. Edlinger, "Comparison of dry and gel based electrodes for p300 brain-computer interfaces," *Frontiers in neuroscience*, vol. 6, p. 60, 2012.
- [16] M. Guermandi, R. Cardu, E. Franchi Scarselli, and R. Guerrieri, "Active electrode IC for EEG and electrical impedance tomography with continuous monitoring of contact impedance," *IEEE Trans. Biomed. Circuits Syst.*, vol. 9, no. 1, pp. 21–33, 2015.
- [17] OpenBCI. Open-source brain computer interfaces. [Online]. Available: <http://www.openbci.com>
- [18] C. Guger, C. Groenegrass, C. Holzner, G. Edlinger, M. Slater, and M. V. Sánchez-Vives, "Brain computer interface for virtual reality control," *Cyberpsychology & Behavior*, 2009, vol. 12, num. 1, p. 84–84, 2009.
- [19] Y. Renard, F. Lotte, G. Gibert, M. Congedo, E. Maby, V. Delannoy, O. Bertrand, and A. Lécuyer, "Openvibe: an open-source software platform to design, test, and use brain-computer interfaces in real and virtual environments," *Presence*, vol. 19, no. 1, pp. 35–53, 2010.
- [20] G. Schalk, D. J. McFarland, T. Hinterberger, N. Birbaumer, and J. R. Wolpaw, "Bci2000: a general-purpose brain-computer interface (bci) system," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 1034–1043, 2004.
- [21] M. Kleiner, D. Brainard, D. Pelli, A. Ingling, R. Murray, C. Broussard *et al.*, "What's new in psychtoolbox-3," *Perception*, vol. 36, no. 14, p. 1, 2007.
- [22] A. Schlogl and C. Brunner, "Biosig: a free and open source software library for bci research," *Computer*, vol. 41, no. 10, pp. 44–50, 2008.
- [23] M. R. Nuwer, G. Comi, R. Emerson, A. Fuglsang-Frederiksen, J.-M. Guérit, H. Hinrichs, A. Ikeda, F. J. C. Luccas, and P. Rappelsburger, "Ifcn standards for digital recording of clinical eeg," *Electroencephalography and clinical Neurophysiology*, vol. 106, no. 3, pp. 259–261, 1998.
- [24] C. Sidebottom, H. Rudolph, M. Schmidt, and L. Eisner, "Iec 60601-1—the third edition," *Journal of Medical Device Regulation-May*, p. 9, 2006.
- [25] M. Guermandi, E. F. Scarselli, and R. Guerrieri, "A driving right leg circuit (dgrl) for improved common mode rejection in bio-potential acquisition systems," *IEEE Trans. Biomed. Circuits Syst.*, 2015.
- [26] *8-Channel, 24-Bit Analog to Digital Converter with Integrated ECG Front End*, Texas Instruments, 08 2015, rev. K.
- [27] J. Axelson, "Serial port complete second edition: Com ports, usb virtual com ports, and ports for embedded systems," 2007.
- [28] g.tec GmbH. g.sahara. [Online]. Available: <http://www.gtec.at/Products/Electrodes-and-Sensors/g.SAHARA-Specs-Features>
- [29] F. R. I. Inc. Disposable / reusable dry eeg electrode [tde-200]. [Online]. Available: <https://fri-fl-shop.com/product/tde-200/>
- [30] *Atmel Smart ARM-based MCU*, Atmel, 03 2015, rev. 23 Mar 2015.
- [31] A. Delorme and S. Makeig, "Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis," *Journal of neuroscience methods*, vol. 134, no. 1, pp. 9–21, 2004.
- [32] A. Capilla, P. Pazo-Alvarez, A. Darriba, P. Campo, and J. Gross, "Steady-state visual evoked potentials can be explained by temporal superposition of transient event-related responses," *PloS one*, vol. 6, no. 1, p. e14543, 2011.
- [33] L. Qin, L. Ding, and B. He, "Motor imagery classification by means of source analysis for brain-computer interface applications," *Journal of neural engineering*, vol. 1, no. 3, p. 135, 2004.
- [34] B. J. Edelman, B. Baxter, and B. He, "Eeg source imaging enhances the decoding of complex right-hand motor imagery tasks," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 1, pp. 4–14, 2016.
- [35] R. D. Pascual-Marqui, M. Esslen, K. Kochi, D. Lehmann *et al.*, "Functional imaging with low-resolution brain electromagnetic tomography (loreta): a review," *Methods and findings in experimental and clinical pharmacology*, vol. 24, no. Suppl C, pp. 91–95, 2002.
- [36] A. Delorme and S. Makeig, "Standardized low-resolution brain electromagnetic tomography (loreta): Technical details," *Methods and Findings in Experimental and Clinical Pharmacology*, vol. 24, no. 5, pp. 5–12, 2002.
- [37] F. Ries, "Heterogeneous multicore systems for signal processing," Ph.D. dissertation, Advanced Research Center on Electronic Systems for Information and Communication Technologies E. De Castro, 2011.
- [38] K. J. Gorgolewski, G. Varoquaux, G. Rivera, Y. Schwartz, S. S. Ghosh, C. Maumet, V. V. Sochat, T. E. Nichols, R. A. Poldrack, J.-B. Poline, T. Yarkoni, and D. S. Margulies, "Neurovault.org: A web-based repository for collecting and sharing unthresholded statistical maps of the human brain," *Frontiers in Neuroinformatics*, vol. 9, no. 8, 2015.
- [39] B. Hamadicharef, "Brain-computer interface (bci) literature-a bibliometric study," in *2010 Int. Conf. Inform. Sci. Signal Processing and their Applications (ISSPA)*. IEEE, 2010, pp. 626–629.
- [40] B. J. Lance, S. E. Kerick, A. J. Ries, K. S. Oie, and K. McDowell, "Brain-computer interface technologies in the coming decades," *Proceedings of the IEEE*, vol. 100, no. Special Centennial Issue, pp. 1585–1599, 2012.
- [41] E. Yin, Z. Zhou, J. Jiang, Y. Yu, and D. Hu, "A dynamically optimized ssvep brain-computer interface (bci) speller," *IEEE Trans. Biomed. Eng.*, vol. 62, no. 6, pp. 1447–1456, 2015.
- [42] C. Vidaurre, A. Schlogl, R. Cabeza, R. Scherer, and G. Pfurtscheller, "A fully on-line adaptive bci," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 6, pp. 1214–1219, 2006.