

CeDeROM Brain Computer Interface

Tomasz Boleslaw CEDRO, Antoni GRZANKA

Abstract Brain Computer Interface is a biomedical equipment used to perform user interaction with computer equipment based on a brain activity measurement. Modular Research System presented in this article is a prototype platform under development aimed for supporting the research groups with low-cost versatile and flexible hardware platform for BCI research but also easy construct and verification of commercial products that can find its use in solving various scientific and real life problems.

1 Introduction

There are two general types of modern Brain Computer Interfaces – invasive and noninvasive – depending on the technique applied for signal acquisition from the tissue. Noninvasive devices acquire biological signals from outside of the human body with use of various tomography methods (such as fMRI) or simple bioelectrical activity readout (such as ECG or EEG). Invasive methods requires electrodes (or more general sensors) to be implanted directly into the organ/tissue, therefore they are inaccessible in amateur applications, especially in an early stage of academic research. Both approaches however require great amount of interdisciplinary knowledge, well coordinated team of highly skilled enthusiasts, legal support, proper funding and laboratory equipment.

Prof. Antoni GRZANKA (BSc MSc PhD Hab)

Research Group on Biocybernetic Aparatus, Institute of Electronic Systems, Faculty of Electronics and Information Technologies, Warsaw University of Technology, Poland

e-mail: A.Grzanka@ise.pw.edu.pl

Tomasz Boleslaw CEDRO (BSc MSc)

Cybernetic Research Student Group, Institute of Electronic Systems, Faculty of Electronics and Information Technologies, Warsaw University of Technology, Poland

e-mail: tomek@cedro.info

Because high-end equipment is usually not available for amateur / academic research, while free and open solution usually does not provide minimal functional level, this document presents results of a research aimed at creating inexpensive but versatile modular hardware BCI platform. This work is an ongoing effort to prove that such system can become a reality with use of commercial off the shelf components and the Open-Source Software, some basic laboratory equipment, determination and knowledge.

2 Existing Solutions

All existing solutions presented in this section are somehow baseline for our research, this is why we mention them here, because we want our system to be inexpensive and even more versatile than solutions already available on the market. Tendency to lower the final price enforces smart solutions that make system work without expensive components. Also the computation power offload from hardware into various license-restricted software elements is noticeable in modern solutions which may indicate that signal processing and smart approach is far more important in this kind of solutions than the signal acquisition hardware itself [4] [5].

2.1 *OpenEEG*

*OpenEEG*¹ is the most popular low-cost platform for amateur BCI/Neurofeedback research with fully open documentation and great support from the community that gathered around this project for the past 9 years. There are however some major disadvantages of the platform – hardware design is outdated as it use 8-bit microcontroller with 8-bit ADC and RS232 for data transmission (modern computers does not have this port even installed). Analog amplifier is not equipped with 50/60Hz notch filter so the power interferences are sometimes blocking the measurement. Luckily the software utilities created to work with this hardware can be also used to work with other devices that conform to the communication standards (open protocols named P2 and P3), what makes it possible to reuse them in different research projects.

This is the reference design for large amount of amateur projects, especially academic ones. Although open architecture the price of the hardware estimates around \$500 and still require some crafting skills (Figure 1), as the pre-assembled modules come without casing, electrodes and power supply. This is why already existing low-cost commercial solutions may seem to be a good alternatives for home use BCI research for similar price...

¹ <http://openeeg.sf.net/>

2.2 *NeuroSky MindWave*

One of the simplest commercial BCI devices is the *NeuroSky MindWave*². Using wireless connection to the Microsoft Windows based Personal Computer with attached USB dongle, MindWave use only two electrodes – one is placed at the central point of a forehead, second use ear clip – to measure biopotential in one channel differential manner. Device measures „user concentration ratio” to drive various computer programs, therefore its space navigation abilities are limited, but it seems to be very good candidate for improving learning process and techniques, considering lowest price of 100EUR among this kind of devices.

2.3 *Emotiv EPOC*

*Emotiv EPOC*³ seems to be most advanced low-cost BCI device for home use. Just as MindWave, it uses wireless connection to the Microsoft Windows based Personal Computer using attached USB dongle. Software is closed-source and restricted by various licenses, so user can purchase more sophisticated components for higher price. Multiple measurement channels are available to the system mostly placed at the front-top and back-bottom part of the head. Hardware use saline sensors for signal acquisition, so they are low cost but require hydration before each use and replacement felt sensors (made of hard foam) as they tend to drop off. Software can detect quality of electrode connections with simple coloured head diagram and system needs to have all channels operating before start. There is also an accelerometer built into the headset and the attached application can read its position in a realtime, so having read the muscle activity and even separate channel for the eye blink it is possible to drive mouse cursor on the computer screen. In order to perform „control by thoughts” system software requires complicated training sessions prior using the control abilities in a videogame environment – user needs to retry displayed cube movement forward, backward, left, right, etc to correlate brain activity with the object movements – but it did not seem to work for more than one type of movement at time or it really required hard few days (weeks?) training to learn how to operate that device.

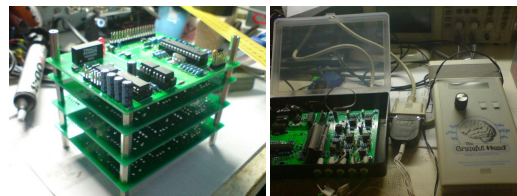


Fig. 1 OpenEEG device during assembly and connected to an EEG equipment calibration unit.

² <http://neurosky.com/Products/MindWave.aspx>

³ <http://www.emotiv.com/apps/epoc/299/>

2.4 *gTec.at Devices*

gTec.at⁴ offers high-end equipment for biopotential measurement that became a standard BCI platform among many different research groups around the world, but the price of tens of thousands euros makes it inaccessible for academic and amateur research. Company also conducts its own research in area of BCI, various biopotential measurement techniques and methods, including innovative „dry electrodes” still unavailable on the market from other manufacturers. We did not have access to this kind of hardware, but according to its parameters stated on a website, it is our goal to create something at least as good as this platform.

2.5 *Software*

There are several applications that can work with OpenEEG system, therefore we can reuse them in our research, including OpenViBE⁵ an Open-Source and multi-platform application for BCI interaction in Virtual-Reality environment, BCI2000⁶ [1] well known BCI framework for Windows with some licensing restrictions for commercial use but free for academic research, Open-Source BrainBay⁷, and many more.

There are also other existing applications that can work with OpenEEG, but they work only on Windows platform, they are not Open-Source so it is not possible to fix bugs in relatively old releases, some of them are restricted by licensing constraints. These programs however are the best (often the only) choice home or academic user can get to experiment with BCI systems without large software development investment. This is why Open-Source is so important – such applications can be ported to other operating systems, they can be extended to match new devices, bugs can be fixed by other developers, all this free of charge.

3 Our Solution – CeDeROM BCI

Professional BCI research project with billion dollar funding can be only seen in serious scientific publications or popular science TV programs, but they are inaccessible to an early stage academic research groups. The technology for reading biological signals of brain activity in non-invasive manner is already available. Computational capabilities of modern computer systems are good enough to perform advanced signal processing [9] and classification [2] that can be used for decision

⁴ <http://www.gtec.at>

⁵ <http://openvibe.inria.fr>

⁶ <http://www.bci2000.org>

⁷ <http://www.shifz.org/brainbay>

making to drive actions „by thoughts”. It is probably the good classification algorithm that will eventually solve the problem, or smart method of reading and stimulating brain activity in a closed loop feedback used for external object control. The algorithm however cannot exist without a hardware platform that makes it possible to operate and interact with real world data. Because commercial solutions are too expensive and free also cost some money but does not provide enough efficiency, we have decided to take the challenge and create such system from scratch, both in software and hardware domain. This solution should be relatively inexpensive because it use commercial off-the-shelf components (COTS) and free/open-source software for licensing freedom, cost reduction, and better insight into the „black boxes” being used in such system.

The main goal of solution presented in this article is to provide hardware platform for BCI research, that will be inexpensive but provide superior possibilities as compared to existing solutions – it must be better than gadget-like and open-source solutions, but also allow to perform tasks available only to expensive and closed-source solutions, even if this means creating solutions from scratch. Modular design is a very important factor because single component or whole configuration will be available for verification in short period of time with no need to redesign whole system which also means low research expenses. The software part is aimed to reuse existing open-source applications that are well known to the enthusiast and/or professional community, so the hardware will be just as easily applicable as all existing solutions before. Open-Source allows modification freedom, better testing and worldwide feedback from the community, no licensing restrictions, better insight into details and many more advantages at cost of time necessary to create such solutions, often from scratch.

3.1 Hardware

Figure 2 presents block diagram of the *CeDeROM Brain Computer Interface* platform. It consists of many modules that can be interconnected together to match desired configuration – their detailed description can be found in later sections. Modular design allows easy reconfiguration and practical verification of theoretical ideas or different system components. The system is divided into following functional areas:

- *Processing and Control* (CPU_BRD) that allows signal processing, decision making, results presentation and analysis, etc. At the moment those functionalities are provided by Personal Computer, Mobile Device, and Programmable Logic (FPGA). Each of the provided hardware platform has its advantages that offer unique possibilities of final solution application. Personal Computer is a very popular platform for design, development and simple experimentation. Mobile experiments with constant monitoring is important in holter-like applications, it can be also equipped with GSM/GPS module to allow realtime interaction no matter the location. FPGA solution is best for real-time high computa-

tional complexity applications that cannot be implemented on a PC, also the final verification of the integrated circuit can be performed here before manufacturing process, or any other standalone application that would have required a computer otherwise.

- *Analog-To-Digital Conversion* (ADC_BRD) provides bitstream output of the converted analog input signals connected to the system. Different types and configurations of converters can be applied, tested, and parametrized this way. High-resolution and multi-channel conversion requires powerful enough control modules in terms of I/O throughput as it turned out during our research.
- *Analog Amplification* (AMP_BRD) provides analog front-end for the analog-to-digital modules in form of a biological signal amplifier. It is possible to verify different operational amplifiers, their parameters and configurations during the research of a new design, but also commercial amplifiers can be connected and used for comparison or reference.
- *Power Supply* (PWR_BRD) provides supply for both analog and digital circuits. In most cases also the galvanic separation can be implemented in this module, because of biomedical equipment safety requirements. It is important to remember that data lines also need the galvanic barrier when separation is applied on the power supply unit.
- *Adapters* (ADP_BRD) provide mechanical and electrical connection between modules and the control systems, so the modules can be connected to different hardware platforms such as ARM or FPGA.

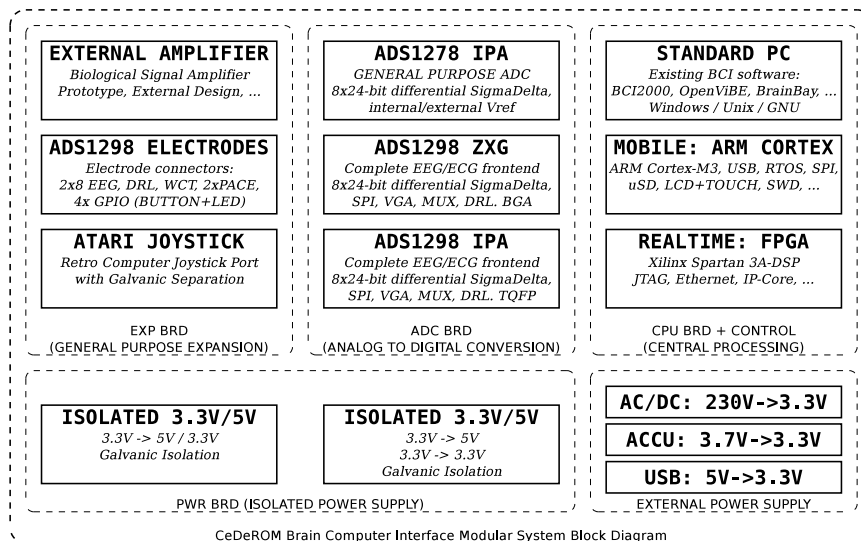


Fig. 2 CeDeROM BCI Block Diagram resembling pool of available hardware modules / features to match specific configuration of a target application. New features can be easily added to the system in form of a dedicated hardware boards or software modules.

- *Expansion Boards* (EXP_BRD) provide various expansions for the platform that can be used for user interaction (i.e. audio–visual signallization, push buttons), signal acquisition (i.e. electrodes), information exchange and interconnect with other systems (i.e. videogame joystick, machine control, etc).

3.1.1 Hardware Implementation

Figure 3 presents photos of assembled modules boards stacked on top of each other and then connected to control units. In this example configuration top board contains push–buttons, LED and electrode connectors. Electrodes lead signal to the analog board below or the analog–to–digital conversion unit with analog frontend provided internally. Below the ADC unit there is a power supply with galvanic separation mounted directly on the adapter board. Bottom board is the control unit (FPGA or ARM based) that provides both power supply and input/output control signals. Additional expansion board allows to control external hardware with electrical signals. Detailed description of the modules is presented below.

3.1.2 CPU_BRD: Xilinx Spartan–3A DSP FPGA

FPGA⁸ technology gives ability to create dedicated digital solutions that can contain whole computer system and dedicated real–time digital processing units in one integrated circuit with rapid implementation and reconfiguration time. Using Open–Source Intellectual Property Cores the cost is minimal. As a demo application we have implemented standalone PONG⁹ videogame with VGA display that can also drive external computer hardware using joystick port, still it is possible to implement more advanced methods [8]. More sophisticated applications are possible such as presented in [5].



Fig. 3 CeDeROM BCI Assembled Circuit Boards.

⁸ <http://en.wikipedia.org/wiki/Fpga>

⁹ <http://en.wikipedia.org/wiki/Pong>

The FPGA CPU module is based on Xilinx Spartan 3A–DSP Development Board¹⁰. This is the high–capacity device for DSP (Digital Signal Processing) applications that can contain both digital components such as CPU (i.e. ARM, AVR, PowerPC, etc) with operating system (i.e. FreeRTOS, FreeBSD, Linux, etc) and various signal processing circuits using dedicated adder/multiplier logical blocks present in this chip. Development Board is equipped with 10/100/1000Gbit Ethernet controller, JTAG Port, RS–232 Serial Port, low cost VGA output, 128MB DDR2 RAM, 16Mx8 BPI Flash, 64MBit SPI Flash, built–in voltage regulators, and many bipolar/unipolar input/output lines to general use. A special Adapter Board was designed to match the *CeDeROM BCI* modules mechanical and electrical standards.

3.1.3 CPU_BRD: Stm32Primer2 (ARM Cortex–M3)

The ARM CPU module is based on Stm32Primer2¹¹, a popular Development Kit based on ARM Cortex–M3 CPU with built–in 400mAh accumulator, color graphical LCD display with backlight, push buttons, micro SD card slot, USB 2.0 Device connector, RLink JTAG/SWD Debug Interface, audio codec with speaker and microphone. All those features make it perfect candidate for mobile or even wearable BCI applications such as presented in [4]. A special Adapter Board was designed to match the *CeDeROM BCI* modules mechanical and electrical standards.

We have also chosen Stm32Primer2 because of its good support by the community gathered around its creator, Raisonance company, with dozens of example applications with source code and schematics available. The software components should be easy to integrate and combine, just as the hardware modules we have designed, resulting in rapid solution development. This might be also a good platform for first steps into an embedded systems world of microelectronics.

3.1.4 ADC_BRD: ADS1298

Analog–to–Digital Conversion Boards (ADS1298 ADC_BRD) are based on ADS1298¹² an innovative integrated circuit from BurrBrown / Texas Instruments which provides a full featured frontend for biological signal measurement with variable–gain input amplifiers (VGA), advanced multiplexer (MUX), DRL negative feedback reference potential generator, electrode lead–off detection and 8 independent differential 24–bit Sigma–Delta analog–to–digital converters with SPI (Serial Peripheral Interface) for interfacing with control circuit (CPU_BRD in our case). There is even a special ADS1298R (R suffix) version of the device that allows respiratory measurement based on body impedance variation. Device contains internal oscillator and voltage references, so it is a perfect candidate for single chip mobile applications where no

¹⁰ <http://www.xilinx.com/products/boards-and-kits/DO-SD1800A-DSP-SK-UNI-G.htm>

¹¹ <http://www.stm32circle.com>

¹² <http://www.ti.com/product/ads1298>

advanced analog circuitry is necessary. It is possible to use external analog circuit with additional filters and amplifiers but this will decrease some features performance such as electrode lead-off detection. This one IC contains all features necessary for our application, so it is a perfect candidate and even without sophisticated analog frontend it should suffice for many experimental and production applications. It is also possible to attach additional AMP_BRD analog module or simply use ADS1298 Electrodes EXP_BRD to connect standard EEG electrodes.

3.1.5 ADC_BRD: ADS1278

Although ADS1298-based module is very flexible biological signal acquisition device it might be necessary to measure some other kind of signals or use totally different analog front-end that ADS198 could not fit. This is why separate module, a ADC_BRD: ADS1278, was created using ADS1278¹³ integrated circuit from BurrBrown / Texas Instruments.

ADS1278 contains 8 independent 24-bit Sigma-Delta analog-to-digital converters. It does not use SPI for communications (but can act as SPI-like transmitter-only) so it requires more control and signal lines, which all of them are also galvanically separated with optocouplers. Power supply needs to be supplied from Isolated PWR_BRD because module has its own ultra-low-noise precision 3.3V and 1.8V voltage regulators. There are three built-in reference voltage sources for ADC – two precision reference voltage source providing 2.048V / 4.096V and the double 16-bit Ultra-Low-Glitch DAC (Digital-To-Analog Converter) based on DAC8552 device that can provide any given voltage from available range if properly programmed. Converter use inputs of analog connector, so it can be connected to another AMP_BRD with dedicated analog front-end of preferred design.

3.2 EXP_BRD: ADS1298 Electrodes

This module is dedicated to ADS1298 ADC_BRD module providing sockets for analog EEG inputs and also input/output section for user interaction based on ADS1298's four GPIO lines. All of four GPIO lines are buffered and can act as inputs and/or outputs – by default all lines are pulled high (as they cannot float). When line is set as input pressing button will force low state, when line is set as output low state will activate the buffered LED, so all lines are „active-low” both for input and output, otherwise pulled high by a resistor. It is possible to mount buzzer instead of LED to generate an audio tone. Buttons can be used for testing user interaction such as reflex or stimuli response. LEDs and buzzers can be used to generate stimuli or a time marker (i.e. audio-video binary pulse sequence) on video recording for better synchronization and easier result analysis.

¹³ <http://www.ti.com/product/ads1278>

3.2.1 EXP BRD: Atari Joystick

It is important to research and demonstrate „control by thoughts” to be in interaction with external multimedia and entertainment devices such as gaming platforms as they create market that is still expanding into new areas of everyday life. This is why a joystick module was developed. The joystick port is a standard DB-9M port, but the plug should be longer and narrow to fit into casing. Old computers used 5V logic, so the joystick port had one reference voltage output, one ground output, and five inputs for each direction (up, down, left, right, fire). There were also two analog inputs for X and Y axis, but standard joysticks used simple on/off scheme to send signals. Joystick therefore is a simple module that separates galvanically our BCI system from the target computer with optocouplers producing on/off digital signals and making use of biomedical equipment safe to use with different types of target hardware, multimedia devices, even motor control as presented in [6].

3.3 Software Implementation

Open-Source¹⁴, GNU¹⁵ and BSD¹⁶ software was used for solution design, development, implementation and application, unless absolutely necessary and noted accordingly (i.e. inexpensive Eagle CAD¹⁷, or free version of Xilinx ISE¹⁸). Programs available for other free and open BCI solutions should work with design provided in this document with no additional configuration or environment change. We believe that this approach, although more time and work consuming, brings more benefit not only to this project itself but also for many other designers that can reuse tools created in this project and give additional features in return. This is the power of Open Source and Free Software¹⁹ that we want to use and extend ourselves.

3.3.1 Brain Computer Interface Open Protocol

Brain Computer Interface Open Protocol²⁰ is our approach to create free, fully featured, open-source and compact protocol for data exchange with Brain Computer Interface class devices. Different groups around the world use often closed source applications, devices and protocols, therefore information exchange in a standard manner is difficult. This is why BCIOP was invented and implemented using a

¹⁴ http://en.wikipedia.org/wiki/Open_source

¹⁵ <http://www.gnu.org>

¹⁶ <http://www.freebsd.org>

¹⁷ <http://www.cadsoftusa.com>

¹⁸ <http://www.xilinx.com/products/design-tools/ise-design-suite/ise-webpack.htm>

¹⁹ <http://www.fsf.org>

²⁰ <http://bciop.sf.net>

C programming language so it can work on almost any embedded hardware, it can be included into a bigger software project source tree, exported as a static library, connected to an external application as dynamic library, or expanded with Python scripting and user interface. To minimise data overhead TLV is used to transfer data in binary manner, that is Tag–Length–Value builds one data packet. Tag determines function, Length determines data length with octet (octet = 8bits) as base unit. Value is the payload. BCIOP is still under development, we plan to extend its functionalities according to our future needs and create high level applications in Python programming language for data analysis and conversion.

3.3.2 Device Drivers in Matlab

Designing advanced data acquisition systems requires whole set of methods to verify a prototype and the data that it produces. This can be achieved by use of Matlab²¹, or free Matlab–like software (ie. Octave²², SciLab²³, or similar). With its wide spread of signal spectrum analysis, filter design and data modelling techniques, this is perfect environment for research related activities. Data can be transferred from device hardware into matlab environment by use of commercial Data Acquisition Toolbox and standard data bus, or design specific dedicated data bus and self written device driver using a method that we have discovered.

Device Driver is usually a set of functions performing device configuration and data transfer. In Matlab environment this can be implemented as set of MEX files or standard dynamic library. MEX files are Matlab specific, can be written in C or Fortran language, and they are easier in later use. Dynamic libraries can be written in any programming language, used by any other application, and they are just a bit harder in later use with Matlab. Dynamic libraries are the key method in creating highly customisable device drivers (take a look at „*MATLAB Interface to Generic DLLs*” Matlab Manual section for detailed information). Functions included in a dynamic library can be loaded into Matlab runtime memory and become accesible directly from the interpreter commadline. In most cases data cast is automatic and Matlab types are preffered. Dynamic Libraries can also be written in languages other than C, but the library interface must conform to the C dynamic library standard. Therefore creating proper dynamic library with set of functions that can perform control over a specific device it is possible to have free and independent device driver.

²¹ <http://www.mathworks.com/>

²² <http://www.gnu.org/software/octave/>

²³ <http://www.scilab.org/>

3.3.3 JTAG / IEEE1149.1

JTAG²⁴ stands for *Joint Test Action Group* and is a common name for *IEEE1149.1* [7] protocol used for various testing mechanisms in silicon manufacturing, printed circuit board (PCB) connection verification, early stage embedded system development, and many more. JTAG requires target system to have special connector with standard-defined signals (TDI, TDO, TCK, TMS, TRST) and special silicon block residing on integrated circuit that allows JTAG operations. Connection is possible using dedicated interface that interconnects target system with a personal computer running software that can interact with the silicon block using JTAG as command transport system. Internal register access can be used to verify proper functionality (i.e. after fresh silicon fabrication), connections on the board and communication with other devices such as flash memory, driving the input-output pins, or even debugging program execution when dedicated In-Circuit-Emulation is embedded into CPU...

JTAG is used in our research to upload firmware into microcontroller and FPGA devices using open-source applications such as UrJTAG²⁵ and OpenOCD²⁶. We use and support open and low-cost solutions where possible.

3.3.4 Serial Wire Debug

Serial Wire Debug²⁷ is a new low-level embedded system access introduced by ARM Corporation²⁸ in their new CPU design *ARMv7* named *Cortex*. It is compliant to ARM Debug Interface²⁹ version 5 that specifies all requirements and capabilities of this transport. We will call it transport, because its purpose is to transport commands between debug software on the host computer and debug port on the target system, just as JTAG does, but in a different fashion. JTAG use state machine design, while SWD use packed-based half duplex serial link with lower pin count than JTAG.

Having already insight and tools to work with this new Serial Wire Debug transport gives an opportunity to work with upcoming devices in near future and gain skills on already existing ones. Creating development tools before developers can have it is also great benefit, considering there was no open application to work with SWD before. The LibSWD³⁰ that we have created is a **first in the world open implementation** of the Serial Wire Debug Open Framework already integrated with

²⁴ http://en.wikipedia.org/wiki/Joint_Test_Action_Group

²⁵ <http://urjtag.sf.net>

²⁶ <http://openocd.sf.net>

²⁷ <http://www.arm.com/products/system-ip/debug-trace/coresight-soc-components/serial-wire-debug.php>

²⁸ <http://www.arm.com>

²⁹ <http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ih0031a/index.html>

³⁰ <http://libswd.sf.net>

UrJTAG and OpenOCD low-level embedded systems access software utilities. The LibSWD is still under heavy development.

3.3.5 Universal Serial Bus vs. Ethernet and TCP/IP

USB³¹ stands for Universal Serial Bus, a serial interface to exchange data between computer and peripherals. It uses Plug'n'Play Host-Device architecture, where Host is responsible for all data manipulation on the bus, device identification and driver load/unload, while Device can receive or send data when asked by the Host. Device can be dynamically connected and disconnected without need to turn off or restart computer. Devices are identified by PID and VID number pair (Product and Vendor ID) along with Identifier String. There are many different Transfer Modes to fit needs of a specific project/application. USB Device can have more than only one interface using common physical connection what makes it good candidate for multipurpose devices.

Unlike dedicated serial-to-USB conversion circuits³² that can provide single Virtual COM Port for devices with no USB connectivity, modern microcontrollers are equipped with *USB Device* peripheral silicon block and it is possible to create almost any type of device using proper protocol and descriptors implementation. We have implemented serial console management port and data transfer port on ARM LPC2148 microcontroller using open source libraries, still being able to extend its capabilities as necessary (i.e. so device become HID, etc).

Ethernet³³ is the common standard for Local Area Networks building the Internet, using TCP/IP³⁴ protocol for information exchange. Ethernet interfaces are now built into Personal Computers, just as the USB ports, so they are both standard equipment. Ethernet requires similar complexity of stack implementation into firmware and even less programming from the use application developer perspective. Different operating systems use different USB host and stack implementation, it is relatively hard to implement in firmware, and there are also other disadvantages of the bus we are considering Ethernet to be somehow better standard for information exchange for our devices. Although LibUSB³⁵ is a good standard for multiplatform and open implementation of USB communications to the user application, Ethernet and TCP/IP works as fast as USB, has better support from operating system and software development perspective, and allows to connect devices directly into the Internet network.

³¹ <http://www.usb.org>

³² <http://www.ftdichip.com/Products/ICs/FT232H.htm>

³³ <http://en.wikipedia.org/wiki/Ethernet>

³⁴ http://en.wikipedia.org/wiki/TCP/IP_model

³⁵ <http://www.libusb.org>

4 Summary

The architecture of a complete system for Brain Computer Interface has been presented. The paper describes a concept and vision of the BCI as the work package for the larger team of engineers. The hardware, analog and digital, components has been specified, the software tools also proposed. Some subsystems and protocols are made available on open source and/or free software rules.

References

1. Schalk, G. McFarland, D.J. Hinterberger, T. Birbaumer, N.; Wolpaw, J.R: BCI2000: a general-purpose brain-computer interface (BCI) system, *Biomedical Engineering, IEEE Transactions on*, June 2004, Vol. 51 (6), 1034 - 1043.
2. F Lotte, M Congedo, A Lcuyer, F Lamarche and B Arnaldi et al: A review of classification algorithms for EEG-based braincomputer interfaces 2007 *J. Neural Eng.*
3. Stamps, K. and Hamam, Y., Towards Inexpensive BCI Control for Wheelchair Navigation in the Enabled Environment A Hardware Survey. *Brain Informatics*, (2010), 336-345.
4. Luca Piccini, Sergio Parini, Luca Maggi and Giuseppe Andreoni: A Wearable Home BCI system: preliminary results with SSVEP protocol, *Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference Shanghai, China, September 1-4, 2005.*
5. Kuo-Kai Shyu, Po-Lei Lee, Ming-Huan Lee, Ming-Hong Lin, Ren-Jie Lai, Yun-Jen Chiu, Dept. of Electr. Eng., Nat. Central Univ., Jungli, Taiwan, Development of a Low-Cost FPGA-Based SSVEP BCI Multimedia Control System, *Biomedical Circuits and Systems, IEEE Transactions on*, Issue Date: April 2010, Volume: 4 Issue:2, On page(s): 125 - 132, ISSN: 1932-4545.
6. Brice Rebsamen, Etienne Burdet, Cuntai Guan, Haihong Zhang, Chee Leong Teo, Qiang Zeng, Christian Laugier, Marcelo H. Ang Jr., "Controlling a Wheelchair Indoors Using Thought," *IEEE Intelligent Systems*, vol. 22, no. 2, pp. 18-24, Mar./Apr. 2007, doi:10.1109/MIS.2007.26.
7. Osseiran, A.; EIG, Eng. Sch. of Geneva: Test standards (with focus on IEEE1149.1). *Circuits and Systems, 1995.*, *Proceedings.*, *Proceedings of the 38th Midwest Symposium on Issue Date: 13-16 Aug 1995 On page(s): 708 - 711 vol.2 Meeting Date: 13 Aug 1995 - 16 Aug 1995 Location: Rio de Janeiro , Brazil Print ISBN: 0-7803-2972-4.*
8. Wang Qiang, Olga Sourina, Nguyen Minh Khoa; School of EEE, Nanyang Technological University Singapore: A Fractal Dimension Based Algorithm for Neurofeedback Games. *The Visual Computer Volume 27, Number 4*, 299-309, DOI: 10.1007/s00371-011-0551-5.
9. Cichocki, A.; Washizawa, Y.; Rutkowski, T.; Bakardjian, H.; Anh-Huy Phan; Seungjin Choi; Hyekyoung Lee; Qibin Zhao; Liqing Zhang; Yuanqing Li; RIKEN Brain Sci. Inst., Wako: Noninvasive BCIs: Multiway Signal-Processing Array Decompositions. This paper appears in: *Computer Issue Date: Oct. 2008, Volume: 41, Issue:10, On page(s): 34 - 42, ISSN: 0018-9162.*