*Article*

# A Hybrid FPGA-Based System for EEG- and EMG-Based Online Movement Prediction

**Hendrik Wöhrle [1,*], Marc Tabie [1], Su Kyoung Kim [1], Frank Kirchner [1,2] and Elsa Andrea Kirchner [1,2]**

[1]   DFKI GmbH, Robotics Innovation Center (RIC), Robert-Hooke-Str. 1, D-28359 Bremen, Germany;
     marc.tabie@dfki.de (M.T.); Su-Kyoung.Kim@dfki.de (S.K.K.); frank.kirchner@dfki.de (F.K.);
     elsa.kirchner@dfki.de (E.A.K.)
[2]   Robotics Group, Department of Mathematics and Computer Science, University of Bremen,
     Robert-Hooke-Str. 1, D-28359 Bremen, Germany
*   Correspondence: hendrik.woehrle@dfki.de; Tel.: +49-421-178-456-559

**Abstract:** A current trend in the development of assistive devices for rehabilitation, for example exoskeletons or active orthoses, is to utilize physiological data to enhance their functionality and usability, for example by predicting the patient's upcoming movements using electroencephalography (EEG) or electromyography (EMG). However, these modalities have different temporal properties and classification accuracies, which results in specific advantages and disadvantages. To use physiological data analysis in rehabilitation devices, the processing should be performed in real-time, guarantee close to natural movement onset support, provide high mobility, and should be performed by miniaturized systems that can be embedded into the rehabilitation device. We present a novel Field Programmable Gate Array (FPGA) -based system for real-time movement prediction using physiological data. Its parallel processing capabilities allows the combination of movement predictions based on EEG and EMG and additionally a P300 detection, which is likely evoked by instructions of the therapist. The system is evaluated in an offline and an online study with twelve healthy subjects in total. We show that it provides a high computational performance and significantly lower power consumption in comparison to a standard PC. Furthermore, despite the usage of fixed-point computations, the proposed system achieves a classification accuracy similar to systems with double precision floating-point precision.

## 1. Introduction

In the last couple of years, the analysis of physiological data has been successfully applied to achieve different research or application goals, e.g., to control different kinds of devices, to allow locked-in patients to communicate with their environment, or to adapt technical devices to the cognitive state of a human  [1–11].

Especially in the field of neurorehabilitation and support of daily activities the interest to use physiological data has increased. A restriction or loss of the mobility due to a neurological disease or injury, such as stroke or spinal cord injuries, usually reduces the patients quality of life considerably. In such a case, physiological data can be applied to adapt a rehabilitation or support device, for instance an active orthosis or exoskeleton, in order to accelerate or improve the rehabilitation process of patients or to provide a suitable amount of support in daily life.

In this context, the capabilities of the rehabilitation device are a relevant issue. A rehabilitation device should support therapy approaches that involve functional training that is tailored to the physical conditions and therapy state of the patient (assist-as needed) [12,13].

To achieve this, the adaptation of robotic rehabilitation systems (e.g., active orthoses or exoskeletons) [14–22] with real-time analysis of physiological data related to neuronal or muscular activity to predict movements seems to be a promising approach [23–30]. Data that can be acquired non-invasively from the human body, for instance surface Electroencephalography (EEG) and surface Electromyography (EMG), have a range of advantages for practical purposes, e.g., they provide a good time resolution, the data acquisition equipment is relatively cheap and portable. Several studies could show that the approach to adapt assistive or therapeutic devices with respect to physiological data motivates the patients and encourages the active participation in the exercises, which can result in an increased neuroplasticity in, e.g., stroke rehabilitation and, thus, therapy success [31–36].

One important aspect to achieve practical usability of a therapeutic device is to provide a high mobility [37,38]. This can be achieved by the development of small computing systems that control the device, but also process the physiological data and are embedded into the rehabilitation device or other equipment like wheelchairs [39] and exoskeletons [29]. However, systems like Brain-Computer Interface (BCI) are nowadays often used in artificial and stationary experimental setups. Reasons for this are (1) the immobility of the utilized hardware for data acquisition, (2) the experimental conditions in which BCI systems are still employed, but also (3) the size of the computing devices required for the analysis of the human EEG or EMG. Recently, substantial progress has been achieved regarding (1) and (2) (see Section 2.4). Although the development of highly performant and power-efficient miniaturized computing devices is an important aspect for rehabilitation applications, aspect (3) has rarely been in the focus of previous research.

For many rehabilitation applications, four different types of requirements are important for the development of embedded systems. First, the system should be as small as possible and have a low power consumption. Secondly, a computing device that is embedded into a larger rehabilitation device must provide the ability to communicate with various other systems and provide additional features such as the possibility to record data. These "high-level" functionalities are very diverse in structure and their implementation is usually based on a wide range of different software libraries. Fortunately, apart from certain types of communication (e.g., motor control to actuate a device), these functionalities are mostly not time-critical. Hence, these features are well-suited for a realization in software. and can be performed by a generic Central Processing Unit (CPU). Thirdly, for the actual processing of the physiological data, a set of different signal processing and Machine Learning (ML) algorithms has to be applied on the incoming stream of physiological data in real-time. Depending on the amount of data and the complexity of the algorithms, this part can be computationally expensive. If we aim to use movement prediction for the control of an assistive or therapeutic device physiological signals must be chosen that provide this information. However, their analysis should be performed with minimal latency. Fourthly, if different physiological signals are combined to predict movements in a hybrid approach, multiple different signal processing operations have to be applied asynchronously and in parallel to different streams of data. This can be a problem if the processing should be performed in miniaturized processing devices.

In this paper, we propose a hybrid Field Programmable Gate Array (FPGA)-based approach for the implementation of a hybrid BCI, since this can fulfill the requirements discussed above. Our system is based on the Xilinx Zynq® [40], that combines a generic CPU with programmable logic (which is equivalent to an FPGA). This combination makes it possible to implement high-performance processing devices, which are power-efficient, have small dimensions, and can replace a bulky stationary PC. The CPU of the system offers the possibility to use standard software for user interaction as well as high-level and configuration tasks, which are tasks that are usually not performance-critical. Computationally expensive and time-critical operations can be transferred to hardware-accelerators that are implemented using the FPGA, which assists the generic CPU. Multiple hardware accelerators can work independently and asynchronously on different tasks, e.g., to process multiple physiological signals at the same time. Since FPGAs offer the capability to exploit massive parallelism to accelerate computations, they represent a well-suited technology for the implementation of performance-critical

operations. FPGAs achieve this by offering the possibility to use computing architectures that can represent the underlying algorithms. For instance, in signal processing and machine learning systems such as BCIs, data is often processed by multiple consecutive operations. For these applications, the system can be represented by dataflow architectures, where data is streamed directly from one processing step to the next one. This allows exploiting pipelining parallelism and avoiding overhead due to, e.g., transfers between hardware accelerator and external memory. For the target application, a further advantage of FPGAs is the possibility to obtain a tight integration of the actuated rehabilitation device and the data processing system, since FPGAs offer the possibility combine motor control and commutation methods with data processing operations on the same die. Furthermore, an advantage of an FPGAs over other types of hardware-accelerators (e.g., hard-wired Application-Specific Integrated Circuits (ASICs)) is the possibility to benefit from hardware acceleration, but to still have the possibility to adapt the system to different application properties, for instance to the number of different channels used by a specific system, or to exchange or add specific components, which can be helpful if better methods become available or the application requirements change.

In many applications, it is helpful to rely on multiple physiological signals instead on a single one [41]. Since different physiological signals exhibit different properties and contain different information, it can be advantageous to switch between them or use a combination of them. Especially in robotic devices for neurorehabilitation, the combination of EEG and EMG provides advantages over a single modality. For instance, EEG and EMG can be combined to either predict as many movements as possible or to enhance the reliability of movement prediction. If the system provides the capability to combine EEG or EMG-based movement predictions in different ways, it can be used to adapt a possible rehabilitation system to the diagnosis of a patient or different needs of patients in different stages of therapy [25,42]. Depending on the type of combination, the system can be used in therapy sessions for successfully treated patients to enforce their involvement, or alternatively in therapy sessions for patients who still have great difficulties to initiate or execute movements or control the rehabilitation device by means of physiological data. EEG and EMG also exhibit different temporal properties in relation to a movement onset, i.e., it is possible to detect movement intentions early using EEG, while changes in the EMG can only be observed close to the actual movement onset. Moreover, the feasibility to predict movements using EMG might depend on the severity of a paresis, is not applicable for paralyzed patients, and is difficult for patients with spasms or tremor. Furthermore, the combination can help to enhance the reliability of control commands of the device [43,44]. A further advantage of such a combination is the possibility to *adapt* the data-dependent signal processing and machine learning methods used for the EEG-based movement detection by using the EMG-based predictions [45,46].

*Contributions and Structure of the Paper*

This paper makes two major contributions. The first contribution is the development of an FPGA-accelerated mobile system developed for the prediction of upcoming movements. The system is based on a self-developed hardware platform, the ZynqBrain (see Section 3.1), and provides the possibility to combine standard software for high-level tasks and user interaction with application-specific hardware accelerators that increase the computing performance of the overall system. The signal processing related parts of the proposed system are real-time compliant and provide a high amount of parallelism. The proposed system is validated using both an offline evaluation that is making use of former recorded data and an online evaluation during which physiological data is recorded and processed continuously. We show that the system outperforms a standard PC regarding processing speed, while providing an equivalent classification accuracy. Moreover, the system has a low power consumption and a small form factor. This enables the development of future embedded or wearable rehabilitation systems that require a tight connection between signal processing and control of the rehabilitation device.

In the second contribution, we show that the parallel processing capabilities of our proposed mobile system can be used to process multiple different streams of data at the same time in order to increase the reliability of the movement predictions. This is achieved by combining movement-related patterns with each other or with EEG patterns evoked by task-relevant signals, such as an instruction of the therapist to the patient. This is of special interest for patients in early stages of therapy that require instruction and show poor movement capabilities. We expect that the so called P300 is evoked by instructions given by the therapist to the patient. We showed already that this pattern is evoked by seldom, task relevant instructions and can be recorded in complex interaction scenarios [47]. In this work we use a more controlled scenario, where we combine an oddball paradigm with delayed responses on task-relevant events to evoke EEG patterns related to the recognition of task relevant stimuli and movement preparation. Here, we investigate in a controlled scenario whether single-trial P300 detection can be used to improve movement classification performance for controlled therapy sessions in which patients perform movements only after instruction of a therapist. For this purpose we systematically combine the analysis of EMG-data and the two different EEG-patterns to predict movements before they occur and to evaluate the effect of different combinations.

The remainder of this paper is structured as follows. In Section 2, we give an overview about related work. Subsequently, the architecture of the proposed system and the applied data processing methodology are discussed in Sections 3 and 4, respectively. The proposed system is evaluated on physiological data obtained in an example application scenario, the procedures for the experimental evaluation are described in Section 5. The obtained results are presented and discussed in Section 6. The conclusion and a plan for future work is given in Section 7.

## 2. Related Work

### 2.1. EEG-Based Movement Prediction

Electroencephalography (EEG) data can be analyzed to detect neural activity related to movement intentions or upcoming movements, such as Event-Related Desynchronization/Synchronization (ERD/ERS) or Movement Related Cortical Potentials (MRCPs) [48–50]. ERD/ERS reflects changes in the *oscillatory* activity of neural networks by an attenuation or increase in the power of specific frequency bands [51,52]. MRCPs are small changes in the *amplitude* of EEG related to movement planning and execution [53–55]. EEG-based movement prediction is especially helpful in assistive devices for telemanipulation [56,57] and rehabilitation [12,25,58]. Predicting movements using an MRCP-based approach has the advantage that upcoming movements can be detected very early [55]. However, due to the low Signal-to-Noise Ratio (SNR) of raw EEG data, a range of conceptually very different and computationally expensive algorithms has to be applied to extract useful information [59–61]. Hence, the utilized computing hardware must guarantee a sufficiently high performance and low latency to preserve earliness of prediction.

### 2.2. EMG-Based Movement Prediction

The Electromyography (EMG) measures electrical potentials generated by active muscles. Usually surface EMG is relevant for clinical applications and physiological computing [62,63]. The electrodes measure the summed activity of several muscle fibers. EMG analysis has been successfully applied to detect or predict movements [64] for a range of different applications, such as active orthosis and prosthesis or exoskeleton control [65–67]. However, the usage of EMG-based movement prediction is not always suitable, e.g., for paralyzed patients caused by spinal cord injuries.

### 2.3. Hybrid BCI Systems

The focus of most research on EEG-based BCIs is to provide basic communication or control to people with severe motor impairments [1,68,69]. The development of novel augmented BCIs, i.e., BCIs that can be used by individuals in everyday applications may enhance the way how people

interact with technology in their everyday environment [70]. Especially using BCIs for the control of neurorehabilitation devices or orthoses and prostheses [71–75] receives increasing attention. In this context, enhancing the applicability or reliability of BCI systems by combining multiple BCIs to create hybrid BCIs represents an important aspect [41,76,77].

A signal frequently used in such applications is the P300-Event-Related Potential (ERP) [47,78,79]. For instance, P300-based BCIs have been successfully combined with motor imaginary-based BCIs to enhance the control possibilities [72,80] or reliability [81] of the combined system. Hence, hybrid BCI systems can avoid wrong predictions of the upcoming movements to decrease frustration of the patient and increase therapy success. Especially, movement prediction combined with P300 detection can ensure a robust prediction of the upcoming movements with an increased prediction accuracy, since the P300 is detected with a higher accuracy compared to movement prediction. A further possibility is to combine an EEG-based BCI-system with EMG signals, which can similarly increase the reliability of the combined system [25,42,43,82–87]. However, to combine several modalities (e.g., EEG, EMG), it is important that the utilized computing system is capable to process several different data streams in parallel and real-time to be able to combine the individual predictions. Another possibility that we do not consider in this study is to increase the reliability of movement predictions by combining the predictions with other modalities, e.g., eye gaze or limb position [84,88].

### 2.4. Mobile and Embedded BCI Systems

Recent progress in the development of portable EEG systems and dry electrodes [89–94] improves aspects (1) and (2) as mentioned in Section 1. For example, to process the data acquired by the acquisition hardware, most systems use wireless connections such as Bluetooth and IEEE 802.15.4 Zigbee or RF transmission [95–101] to transfer the data to a separate processing device, stationary PC or laptop. However, this might generate additional latency and increase power consumption. Further, lossy compression to reduce the data rate can cause artifacts [102,103]. These are critical aspects especially for real-time movement prediction.

Hence, it would be beneficial to use devices which allow a tight integration of data acquisition hardware, computing device and rehabilitation equipment. In order to integrate computing devices into other systems such as orthosis or exoskeletons, they require small physical dimensions, low weight and low power consumption (in order to be able to use small accumulators) [104]. However, since the employed signal processing operations for the detection of specific EEG patterns are often computationally expensive, current mobile [105,106], and FPGA-based systems [107–111] BCI systems often rely on a small number of electrodes, which is advantageous for practical reasons and sufficient for *simple* approaches based on, e.g., the detection of Steady-State Visual Evoked Potential (SSVEP) [105,112] or seizures [113]. However, for real-time movement prediction or hybrid approaches, the ability of the system to process high-dimensional data is a relevant issue in the *single trial analysis* of EEG and EMG. This is in particular the case when the detection of specific EEG/EMG patterns is difficult and the data is affected by noise or artifacts, as it likely is the case in complex rehabilitation applications [11]. In this case, the usage of a high number of channels is beneficial for accurate classification, since it allows to apply spatial filter-based artifact attenuation [37]. Accordingly, specialized signal-processing systems that use complex algorithms and apply these in an online-fashion and real-time are needed. A recent trend ist the development of such systems for, e.g., fall prediction [110].

Resulting from the discussion above it is clear that real-time analytics of physiological data for rehabilitation applications places high methodological and technological demands on devices that should be used to achieve this: the dimensionality of the data is high, the SNR is low, and complex algorithms have to be used to extract useful information from the data. Furthermore, if the BCI system is hybrid, different streams of data have to be processed independently and in parallel. Finally, the applications might underly strict real-time constrains. In order to solve this problem, high performance embedded computing systems are required that can employ a range of different

signal processing and ML techniques to process the physiological data in real-time and have a low power consumption.

## 2.5. Field Programmable Gate Arrays

FPGAs are a technology that provide the possibility to build parallel architectures that process large amounts of data efficiently under hard real-time constraints and to perform multiple tasks at the same time. Hence, they are well-suited for the intended application of real-time movement prediction.

FPGAs consist of flexible programmable logic elements that contain Look-Up Tables (LUTs) and Flip Flops (FFs) to realize complex combinational logic functions and register-transfer logic. With the increasing technological capabilities, FPGAs became a popular alternative to CPUs and Graphics Processing Units (GPUs) for different embedded and high performance computing tasks [114–119]. To support these kinds of tasks, FPGAs contain different kinds of supporting hardware primitives that can be utilized for these operations, such as fast accessible memories (in terms of Block Random Access Memorys (BRAMs) [120]) to store intermediate data, and Digital Signal Processing (DSP) slices (e.g., the DSP48 slice in Xilinx FPGAs [121]) to support Multiply ACcumulate (MAC) operations.

Since many DSP and ML algorithms require primarily multiplications and additions but no complex control logic, FPGAs are a suitable candidate solution for their implementation. A development that became increasingly popular in recent years is the combination of programmable logic with CPU cores that allow to build System-on-Chips (SoCs) such as the Xilinx Zynq® [40] or Altera Stratix®, Arria® and Cyclone SoCs® [122]. These devices allow to combine standard software executed on the CPU with dedicated hardware accelerators implemented in programmable logic.

However, the development complexity of FPGA-based systems is significantly higher than the usage of a standard software library such as pySPACE [123] or OpenViBE [124]. A current trend to simplify the development process is High-Level-Synthesis (HLS), i.e., the usage of high-level languages such as C or OpenCL instead of the conventional hardware description languages [125,126]. Although this reduces the development complexity considerably, it has been found that designs obtained by using HLS often have a lower performance and higher resource utilization than customized hardware accelerators [127]. Hence, a possible solution is to provide a framework that contains a set of flexible IP cores that can be adapted to specific application requirements and can be easily integrated into larger SoC designs.

A further problem with FPGAs is that floating-point arithmetic is expensive in terms of resource utilization. Hence, usually fixed-point arithmetic is used as a replacement. However, fixed-point arithmetic provides a smaller dynamic range for the representation of values which can result in numerical problems or a decrease of the accuracy of the obtained results. Hence, it is important to evaluate whether the fixed-point arithmetic leads to a decrease of the classification performance.

## 2.6. Dataflow Architectures and Hardware Acceleration

The parallel processing capabilities of an FPGA can only be exploited if they are used to implement a suitable processing architecture. A paradigm that can implement architectures that achieve this and fit especially well to the structure of signal processing systems is the dataflow computing paradigm [128–132]. In this paradigm, independently and asynchronously operating actors exchange data via dedicated channels. If a sufficient amount of input data is available on the input channels, the execution of an actor is enabled. Subsequently, the actor consumes data from the input channels, processes it, and elicits the products of the computation on its output channels. The asynchronous execution of multiple actors makes it possible to design signal processing systems that utilize a large number of parallel operating Processing Elements (PEs). Hence, this computing paradigm is an approach well-suited to map signal processing and machine learning applications to FPGAs [133–135].

Dataflow models differ in expressivity and analyzability. A high analyzability is especially important for for safety-critical applications as it is the case in medical and rehabilitation applications. Hence, our system is based on the self-timed Synchronous Dataflow (SDF) model [129]. This model

requires no complex scheduling algorithms to coordinate the work of the actors and can guarantee deadlock-freeness and bounded memory usage. However, SDF-based systems are not Turing equivalent, i.e., it is not possible to use an SDF-based system to run ordinary software.

The actors can be of different granularity, ranging from fine-grained actors that implement single arithmetic operations up to coarse-grain actors that implement a complete algorithm and include control-flow instances or finite state machines as long as they obey dataflow behavior. A further possibility is to combine standard-hardware architectures, like von Neumann-based CPUs, with dedicated hardware accelerators that follow the dataflow enhance the computing performance of specific, time-critical operations [136]. Accordingly, dataflow architectures are a promising approach to build high-performance and real-time compliant embedded systems.

## 3. Hard- and Software Architecture of the Hybid System

To achieve the constraints related to processing performance, power consumption and physical size, but also provide a high usability, our system uses a combined CPU/FPGA solution. Choosing such an embedded two-in-one solution permits the usage of standard operating system like Linux and a wide range of software on the CPU (see Section 3.3), but provides also the possibility to accelerate specific computations using hardware accelerators implemented in the FPGA.

The basic building block of the system is the self-developed printed circuit board ZynqBrain (Section 3.1). The FPGA-based processing architecture consists of a set of application-specific hardware accelerators realized via the dataflow computing paradigm (Section 3.2). To provide a customary usability of the system, we use a software layer based on pySPACE (Section 3.3).

A further advantage of such a combined CPU/FPGA-based approach is the possibility to establish a direct interface between the components used for biosignal processing and a rehabilitation device. This makes it possible to incorporate the outputs of the movement prediction system directly in the control algorithms of the rehabilitation device. However, in the current study the main focus is to evaluate the accuracy, performance, and power consumption of the developed system and evaluate the feasibility of such an approach for future applications in robotic rehabilitation devices.

### 3.1. The ZynqBrain Electronics Platform

A schematic diagram of the ZynqBrain is shown in Figure 1a. The ZynqBrain is manufactured in Pico-ITX format ($70 \times 100$ mm). The central component is a Xilinx Zynq$^{®}$ ZC7030FBG484-3 [40]. A ZC7030 consists of two parts: a Processing System (PS) (which is an ARM$^{®}$ Cortex-A9 dual core CPU running at 1 GHz) and a Programmable Logic (PL) section, which is equivalent to an FPGA. 512 MB DDR3 SD-RAM are connected to the PS.

The amount of programmable resources of the Zynq ZC7030 is shown in Table 1. In all conducted experiments the dual-core ARM Cortex-A9 processor of the ZC7030 was operating at 1 GHz clock frequency, the programmable logic (PL) at 100 MHz.

**Table 1.** Programmable logic resources of the Xilinx ZC7030.

| Programmable Logic Cells | Look-Up Tables (LUTs) | Flip-Flops (FFs) | BRAMs (36 kb Each) | DSP Slices (DSP48) |
|---|---|---|---|---|
| 125,000 | 78,600 | 157,200 | 265 | 400 |

(a)



(b)



(c)
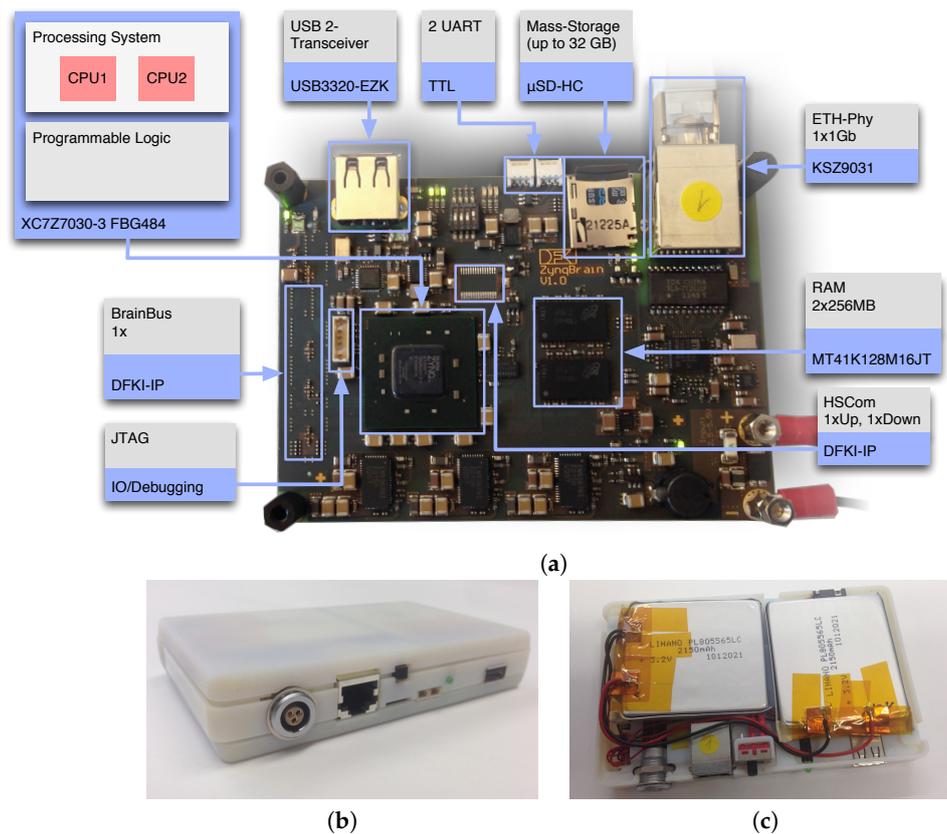
**Figure 1.** The ZynqBrain processing platform: (**a**) printed circuit board; (**b**) assembled system with 3D-printed cover; (**c**) battery pack.

The ZynqBrain provides the following standardized interfaces to connect to other systems: (1) a single USB 2 interface, (2) a 1 Gb Ethernet port, (3) two UART interfaces and (4) a JTAG port. An important additional feature is the availability of specialized interfaces developed for distributed computing in robotic systems, called BrainBus and HSCom [137]. These interfaces provide the possibility to establish connections to control motors [138] and represent an important key feature for the integration of the ZynqBrain into robotic rehabilitation equipment. The USB interface of the ZynqBrain was used to connect the ZynqBrain directly to the EEG-acquisition hardware (see below), a µSD-HC card interface was used as a persistent storage medium to, e.g., store operating system, software, and recorded EEG data.

For the system used in this paper, we created a 3D-printed cover. Besides the ZynqBrain, the cover contains a power management unit and two LiFePo cells with a total capacity of 4300 mAh.

## 3.2. Dataflow Hardware Accelerator Architecture

The processing architecture of the proposed system is shown in Figure 2 (for details about the specific algorithms, see Section 3.3). The PS of the ZC7030 is used for the software parts of the system, while the required Dataflow Hardware Accelerators (DFHWAs) are implemented in the PL.

For each investigated physiological signal, the PL of the Zynq is used to implement a so-called DFHWA. A DFHWA is a task-specific hardware accelerator based on the SDF model (see Section 2.6) and consists of a set of coarse-grain actors. Each actor implements a specific algorithm. Since each physiological signal requires a specific set of operations, the structure of a DFHWA is specific for the corresponding signal. For processing, the data is streamed through the sequence of actors and transformed on its way. Due to the dataflow paradigm, the system is has an inherent parallel architecture that provides the capability to exploit task- and pipeline parallelism to obtain a high

computational performance. Task parallelism is especially important for hybrid BCI systems, which have to apply different operations to different streams of data at the same time. Pipeline parallelism represents the separation of a computation in several stages that can be executed in parallel and fits well to stream based computations as required here. Due to the task parallelism, the different DFHWA operate in parallel without interference.

The DFHWAs are implemented using the reSPACE [132,135] framework, which consists of a library of predefined, parametrizable actors. Hence, all actors are configurable components that can be adapted to specific requirements and combined to build more complex systems. These include basic, widely used algorithms such as Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters, direct current offset removal, standardization, etc. For instance, for the system proposed in this paper, we use a specific set of actors to build four DFHWAs with different purposes (for details about the applied algorithms see Section 4). Another possibility provided by reSPACE is a customizable generator for matrix-multiply based algorithms, which generates a node according to the specification given by a domain specific language. The operations are mapped directly to the programmable logic to allow resource and power efficient arithmetic operations. This makes it possible to generate nodes that implement algorithms frequently used in BCIs, such as spatial filters [139,140] or classifiers.

The DFHWAs are connected to the CPU via an AXI-Lite bus for the transfer of configuration parameters (for details about the used algorithms, see Section 4) and data. Different parameters of the processing can be configured using a set of software-accessible registers. In order to process EEG or EMG data, it is transferred to the input First-In, First-Out (FIFO) buffers and the results are collected from the output FIFO buffers or result registers. The data transfer is performed using Direct Memory Access (DMA). This setup has the advantage that the host CPU is *not* involved in the data transfer and the computations performed in the DFHWAs and is therefore not occupied by these. Subsequently, the actual computations are performed asynchronously by the DFHWAs. As soon as the results are available, the CPU is notified via an Interrupt ReQuest (IRQ).

To access the DFHWAs from a software application, device drivers are needed that interface with the DFHWAs and run as kernel modules. They can be automatically generated by reSPACE according to the meta-data that provides all necessary information to generate the interface of the DFHWAs.
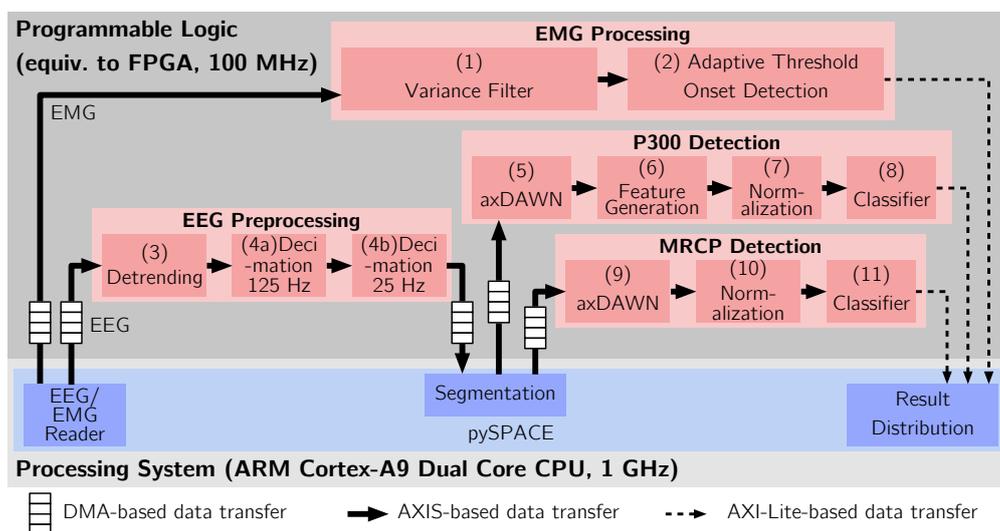


**Figure 2.** Hybrid parallel processing for electroencephalography (EEG) and electromyography (EMG) detection. The numbers shown in each processing step refer to the detailed description in Section 4.

For the hardware implementation in the corresponding DFHWAs, fixed-point operations were used, while all software implementations are based on double precision floating-point operations. For the proposed system, the integer and fractional parts of the fixed-point operations and representation were manually optimized stepwise to obtain an optimal accuracy.

### 3.3. Software Architecture

The software architecture is based on a customized Linux kernel (version 3.12) with a Debian user space system. This allows us to use the signal processing software framework pySPACE [123] in combination with the DFHWAs. The framework pySPACE is used as the high-level processing and infrastructure software to provide service functionalities, i.e., configuration of the data acquisition and recording, management of previously stored data, and to evaluate and store the obtained results while the actual data processing is performed by the DFHWAs.

The second application of pySPACE in this paper is the usage of pySPACE without hardware acceleration in the reference systems (see Section 5.5). Similar to the structure of the DFHWAs, pySPACE contains a set of signal processing and machine learning algorithms that can be combined to build complex signal processing systems. Note that although pySPACE is written in the Python scripting language, it uses Numpy [141] and Scipy [142] for most operations. Numpy and Scipy contain highly optimized C routines for operations such as matrix algebra and digital filtering. Furthermore, time consuming operations not provided by Numpy or Scipy have been implemented as optimized C extensions and parallelized using OpenMP. Hence, overhead related to the fact that Python is used as a top-level framework language are minimal.

### 3.4. Investigated Physiological Signals

To predict movements, we process the following physiological signals:

- MRCP: Both the ERD/ERS and the MRCP have been used to detect movement intentions (Section 2). Since the MRCPs-based movement-prediction is more reliable than the ERD/ERS-based procedure close to the actual movement onset [143], we choose MRCP for movement prediction in our system.
- EMG: Similarly, the EMG is used as a signal that indicates an upcoming movement.
- P300: The P300 is not directly related to an upcoming movement. However, it can be used to select one of several different modes of a BCI system. In this paper, we follow this approach. It is assumed that the subject is most of the time in an idle state and switches to an active state following the instruction of a human or virtual trainer or therapist. In such an application, the successful detection of the P300 in response to a command can be used as an indicator that the subject will perform a movement in the near future. Subsequent to the command, we expect a movement in a time window with a length of 5 s.

### 3.5. Combination of Signals for Hybrid Movement Prediction

An important point for rehabilitation applications is to obtain a high reliability, i.e., predict upcoming movements as early as possible, but avoid wrong predictions of a movement during rest. In order to achieve this, the individual predictions based on different data modalities can be combined to generate a combined prediction. Clearly, using a combination of several different modalities is computationally more demanding than the usage of a single modality, since it requires to apply different methods to several different data-streams in parallel. The proposed architecture has the advantage that the DFHWAs can operate independently and do not share a common resource, as it is the case when a generic CPU is used. The proposed system provides the possibility to apply all operations described above in parallel to different streams of data, and to collect and merge the predictions of the individual DFHWAs to generate a combined prediction.

Individual predictions can be merged in different ways. The proposed system can either (1) output the prediction of a single DFHWA as system output, or (2) combine the predictions from single DFHWAs and generate a combined prediction. Another possibility would be (3) to merge features of different modalities and use a meta-classifier to generate a common prediction. However, such a solution would be more inflexible, i.e., choosing a different combination would require to retrain the meta-classifier. Hence, approaches (1) and (2) are used in the proposed system.

In particular, we investigate the following single modalities and combinations:

No combination with P300 (NP3):

- MRCP: The prediction of the movement onset is based only on the prediction of the MRCP.
- EMG: The prediction of the movement onset is based only on EMG analysis.
- MRCP or EMG (MoE): The prediction of the movement onset is based on the combination of the MRCP and EMG predictions. The combination is obtained using a *logical or* combination of the single predictions. Formally, let $p_{EMG}(t), p_{MRCP}(t) \in \{0,1\}$ be the predictions of the EMG and EEG of an upcoming movement at time $t$, respectively, where 1 represents the upcoming movement. The combined prediction is then given by $p_{MoE}(t) = p_{EMG}(t) \vee p_{MRCP}(t)$.
- MRCP and EMG (MaE): Similar to MoE, but the combination is obtained using a logical *and*, i.e., $p_{MaE}(t) = p_{EMG}(t) \wedge p_{MRCP}(t)$.

Combination with P300 (CP3):

- P300 and MRCP (PaM): As discussed above, the P300 is used as a switch to select between an *idle* and an *active* state. Specifically, let $\hat{p}_{P300}(\hat{t}) \in \{0,1\}$ denote the detection of the P300 at time $\hat{t}$. If $\hat{p}_{P300}(\hat{t}) = 1$, we set $p_{P300}(t) \leftarrow 1$ for $t \in [\hat{t} + 1\,s, \ldots, \hat{t} + 5\,s]$ and compute $p_{PaM}(t) = p_{P300}(t) \wedge p_{MRCP}(t)$.
- P300 and EMG (PaE): Similar to PaM, but the EMG is used for the movement detection, i.e., we compute $p_{PaE}(t) = p_{P300}(t) \wedge p_{EMG}(t)$.
- P300 and MRCP or EMG (PaMoE): This combination is based on the combinations MoE and PaM, i.e., we compute $p_{PaMoE}(t) = p_{P300}(t) \wedge (p_{MRCP}(t) \vee p_{EMG}(t))$.
- P300 and MRCP and EMG (PaMaE): Similar to PaMaE with $p_{PaMaE}(t) = p_{P300}(t) \wedge p_{MRCP}(t) \wedge p_{EMG}(t)$.

## 4. Applied Signal Processing and Machine Learning Procedures

In the following, we discuss the applied operations. To compare the performance of the FPGA-based system to CPU-based systems, all operations were implemented two times. The first implementation is based on software components included in our signal processing software framework pySPACE [123]. The software implementation makes use of double precision floating-point arithmetic and represents a reference for comparison. The second implementation is based on the DFHWA architecture discussed in Section 3.2. In this case, all arithmetic operations are implemented using fixed-point arithmetic. The applied operations are shown in Figure 2 and are, apart from the arithmetic type, equivalent for the software and hardware implementation.

### 4.1. EMG Processing

The processing of the EMG was based on the procedure previously presented in [64]. The EMG processing was performed on data segments of 40 ms length, each segment was represented by a $200 \times c$ dimensional matrix if the sampling frequency of the EMG recording equipment is 5 kHz, where $c \in \{1, 4, 8\}$ refers to the number of EMG channels (see Section 5.2). The processing consisted of two steps:

(1) Variance Filter: To obtain a signal with smoothed baseline noise and enlarged signal amplitudes during movement phases, a running variance method was used for preprocessing of the EMG signals. The calculation was based on [144], but was computed as described in [145] to calculate the running variance $v$ at time $t$ as:

$$v(t) = \frac{1}{W_{VF} - 1} \sum_{i=-2w}^{0} x^2(t+i) - \left( \frac{1}{W_{VF} - 1} \sum_{i=-2w}^{0} x(t+i) \right)^2, \tag{1}$$

where $W_{VF} = 2w + 1$ is the window length. Variance filtering does not change the dimensionality of the data.

(2) Adaptive Threshold Comparison and Classification: The actual onset detection was based on the comparison of the variance-filtered signal with an adaptive threshold. The threshold was computed as

$$T(t) = \mu(t)_{W_{AT}} + p\sigma(t)_{W_{AT}}, \tag{2}$$

with $\mu$ the mean value, $\sigma$ the standard deviation, $W_{AT}$ the length of the window for the mean and standard deviation and $p$ the sensitivity factor of the threshold [146]. To compute the standard deviation, the methods described in [145] were used again. The adaptive threshold compensates slow drifts in the EMG signals or changing noise levels in the signal caused by, e.g., resistance changes at the electrode side. The classification was performed using a threshold comparison. That is, a segment was classified as belonging to the movement class if the single sample of the variance filtered signal of more than a previously chosen number of channels $n_{COT}$ were higher than $T$. Due to the adaptive threshold based movement detection, the flow for EMG-based movement prediction required no explicit training phase. A single classification was computed for each data segment.

### 4.2. EEG Preprocessing

The processing of the EEG data for the detection of the MRCPs and P300 was performed in three parts. Each part was implemented by a separate DFHWA. The first part is preprocessing. It was identical for both the P300 and the MRCP. Similar to the EMG processing, the EEG preprocessing is performed on segments of 40 ms length. Each segment was represented by a $200 \times c$ dimensional matrix, with $c \in \{32, 64, 96, 124\}$. The preprocessing consisted of two operations.

(3) Detrending: First, detrending was used to remove slowly varying signal components using an IIR filter [147], which could otherwise produce a bias in the data.

(4) (a + b) Decimation: Subsequently, the sampling rate was decimated in two steps [148,149], the first step reduced the sampling frequency from 5 kHz to 125 Hz. In the second step, a further reduction of the sampling frequency to 25 Hz was performed. The anti-alias FIR of the second step was parameterized so that all frequencies greater than 4 Hz were attenuated as proposed in [150,151]. In the decimation, each data segment is reduced to a $c$ dimensional vector. The output of the second decimation step was sent back to the main memory for further segmentation in software.

### 4.3. Data Segmentation

Before the data was processed by further processing methods, the preprocessed segments were combined to segments segments specific for the P300 or MRCP processing. The segments were transferred to the corresponding DFHWA and processed independently from each other.

### 4.4. P300 Processing

For the P300, segments of 1000 ms in length were created by assembling 25 preprocessed segments, whenever a target or standard marker was shown to the subject. Hence, each segment was represented by a $25 \times c$ dimensional matrix. For training and evaluation, we used a dataset-wise cross validation as discussed in Section 5.3. The P300 processing consisted of four steps.

(5) Spatial Filtering: The axDAWN spatial filter [140] was applied to decrease the number of channels to four, creating a $25 \times 4$ dimensional matrix. This operation can be realized as a matrix multiplication (using DSP48 [121] slices if realized in hardware) and was implemented using a specialized accelerator for matrix operations [152].

(6) Feature Generation: To generate features for classification the time samples were transformed into local straight line features, i.e., polynomial features of order one [47,57]. To this end, every segment

was divided into subsegments of length of 400 ms that are shifted by 120 ms. A polynomial was fitted to each subsegment. The polynomials allow to describe the P300 by a series of slope values. The extracted slopes were combined in a single 24-dimensional feature vector.

(7) Feature Standardization: A further operation was component-wise feature standardization, i.e., we computed $\mathbf{x}_s \leftarrow \frac{\mathbf{x} - \mu_T}{\sigma_T}$, where $\mu_T, \sigma_T$ are the element-wise mean and standard deviation of all vectors in the training data set, respectively. This operation removes bias from the data due to a constant offset. Feature standardization does not change the dimensionality of the feature vector.

(8) Classification: A Passive-Aggressive Algorithm, type-I (PA-1) [153] was used for classification. The PA-1 classifier is a linear binary online classifier, that is based on a maximum-margin hyperplane and can be applied to nonseparable problems. Preliminary investigations showed no significant differences in classification performance to a linear soft-margin Support Vector Machine (SVM). Similarly to a soft-margin SVM, the PA-1 provides a hyperparameter $C$ to control regularization. $C$ was optimized using a grid-search in the range $\{10^0, 10^{-1}, \ldots, 10^{-6}\}$. A stratified three-fold nested cross validation was used for model selection on the training data [154]. Subsequent to the classification, a threshold correction [155] was used to compensate the bias in the data related to class imbalance.

*4.5. MRCP Processing*

Since we are interested in the exact timing of the EEG-based movement prediction, we use MRCP-based procedures for the proposed system. MRCPs can be represented in the time-domain. For the MRCP detection, overlapping segments (see Figure 3 in Section 5.4), with a duration of 200 ms were used. Similarly to the EMG, a prediction was performed every 40 ms, adjacent segments overlapped by 160 ms. Each segment consisted of $c \times 5$ samples, where $c$ is the number of channels. For training of the spatial filter, classifier and feature standardization, the segments of the training phase were labeled as related to a *movement* or to a *rest* phase based on the temporal relationship to the movement onsets as shown in Figure 3. Similar to the P300 processing, we used a dataset-wise cross validation as discussed in Section 5.3.

(9) Spatial Filtering and Feature Generation: Since the MRCP is reflected by a temporal pattern in the data, time-domain features can be used for the detection of the MRCP. Accordingly, the axDAWN spatial filter [140] can be applied for the MRCP detection to reduce the number of channels to a single channel, i.e., creating a 20 dimensional feature vector.

(10) Feature Standardization: The feature vectors were standardized using the approach discussed in Section 4.4 (7), leaving the dimension of the feature vector unchanged.

(11) Classification: The 20 dimensional feature vector was classified using a Passive-Aggressive Algorithm, type-I [153]. Similar to the P300 detection, the regularization hyper-parameter $C$ was optimized in the range $\{10^0, 10^{-1}, \ldots, 10^{-6}\}$ using a 3-fold nested cross validation. Again, we observed no significant differences in classification performance to a linear soft-margin SVM in preliminary investigations and applied a threshold correction to compensate class imbalance as proposed in [155].

## 5. Experimental Evaluation

The evaluation was split into two different parts:

1. A preliminary offline evaluation to test and evaluate the system regarding different parameters and combination of modalities and select the best parameters for the online evaluation.
2. A subsequent online evaluation to verify that the system works in a real application.

The experimental setup was identical for both the online as well as the offline part.

### 5.1. Experimental Setup

In order to evaluate the proposed system, we developed an experimental setup that combines an oddball paradigm with movement prediction.

Twelve healthy male subjects with normal or corrected to normal vision participated in this study, eight subjects participated in the offline part (age $29.9 \pm 3.3$ years) and four subjects in the online part (age $26.8 \pm 1.3$ years). All participants gave written consent for participation and the study was approved by the ethics committee of the University of Bremen. In order to obtain a sample size sufficient for quantitative analysis, the current experiment was conducted by healthy subjects.
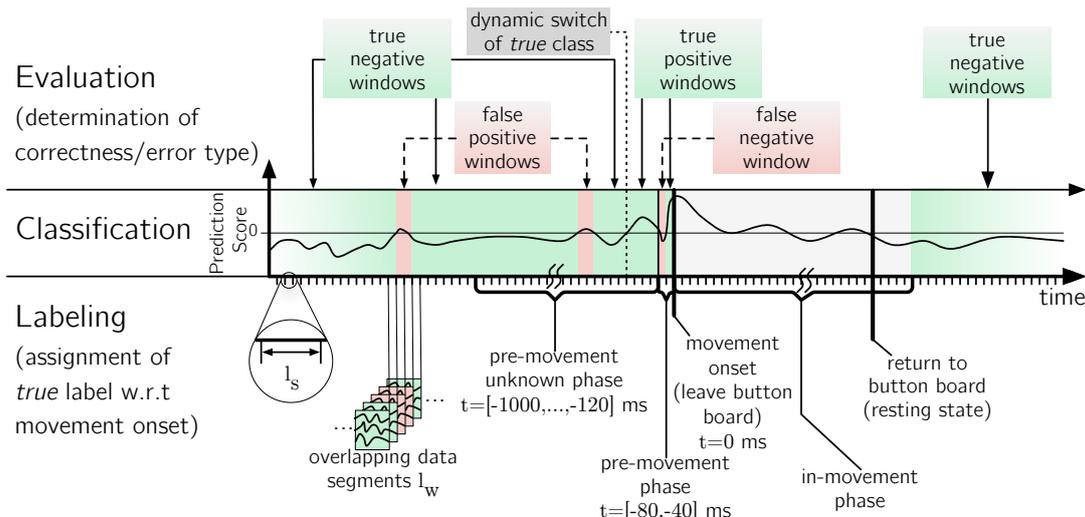


**Figure 3.** Segment-based evaluation. The true label was assigned to each segment based on the temporal relationship to a movement onset. Since we observed highly subject-specific differences of the time when a movement can be predicted before the actual movement onset, we used the following dynamic method to determine whether the true class of a segment is the movement or rest. In all cases the segments ending at $-80$ or $-40$ ms relative to the movement onset were expected to predict a movement (i.e., belong to the movement class, denoted as pre-movement phase). All segments ending in the interval $[-1\,\mathrm{s},\ldots,-120\,\mathrm{ms}]$ were further investigated whether they belong to a consecutive sequence of predictions of the movement class with at most one exceptional case (pre-movement unknown phase). All segments ending between the movement onset and the end of the movement plus an additional tolerance period of 200 ms were not included in the performance evaluation (post-movement phase). All remaining segments were assigned to the resting class. For training of the classifier, the segments ending in the interval $[-120,\ldots,120]$ ms were used as examples for the movement class, all remaining segments (apart from examples ending in the in-movement phase) were used as examples for the resting class.

Figure 4 shows the experimental setup and procedures. The subjects were seated in a chair in front of a table. A monitor, a button board and a buzzer were placed on the table (Figure 4a). The subjects were asked to perform 40 self-paced movements of their right arm according to an oddball paradigm. In the resting state, the hand of the subject was placed on the board. To perform a correct movement, the hand of the subject had to leave the button board, press the buzzer, and return to the button board. The distance between buzzer and button board was approximately 30 cm. Events from button board and buzzer were labeled in the acquired data to determine the movement onset and the end of the movement.
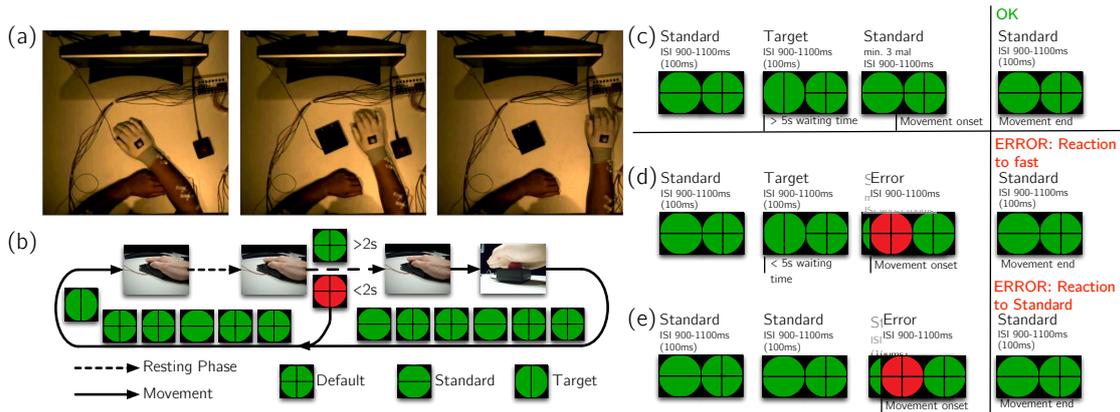
**Figure 4.** Illustration of the conducted experiments. (**a**) Three pictures of the setup, showing from left to right: the resting phase, the movement phase and the buzzer press; (**b**) Visualization of the paradigm; (**c**) Correctly performed movement; (**d**,**e**) Notification of incorrectly performed movements.

The oddball paradigm was implemented using the software Presentation (Neurobehavioral Systems, Inc., Albany, CA, USA). A green circle with a black fixation cross was shown to the subjects. Two different kinds of stimuli were shown to the subjects: Standards, which were indicated by the disappearance of the vertical bar of the fixation cross, and Targets, which were indicated by the disappearance of the horizontal bar for 100 ms. The Inter-Stimulus Interval (ISI) was $1\,\mathrm{s} \pm 100\,\mathrm{ms}$ (Figure 4b). After a Target stimulus, the subjects had to wait for at least 2 s before they were allowed to perform a valid movement (Figure 4c). The waiting time was unknown to the subjects. Movements performed too early (Figure 4d) or not related to a Target stimulus (Figure 4e) were reported to the subjects by a red flash of the circle as error notification and not taken into account for data analysis. A run was finished after 40 correct movements. The study consisted of one recording session per subject, each recording session was divided into three runs as shown in Figure 5 (resulting in 120 movements per subject and session; 80 movements belong to the training and 40 movements to the test phase). 24 datasets from 8 subjects (3 runs $\times$ 8 subjects) were used in the offline evaluation, and 4 datasets from 4 subjects (1 runs $\times$ 4 subjects) for the online evaluation.
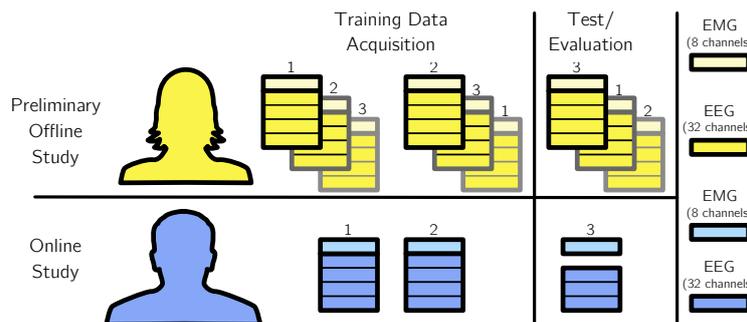


**Figure 5.** Illustration of the online evaluation/offline cross validation scheme (see also Section 5.3). In the preliminary offline study, three runs of data were recorded in total. The runs of the data were used as splits for the cross validation as follows. All combinations of runs in a session were used, resulting in three different training/evaluation combinations. Two recorded runs were used for training of the data dependent methods and the remaining run was used for the offline evaluation. In each run the data was recorded using the Magma-Box-Setup, i.e., it consisted of $4 \times 32 = 128$ channels of EEG data and 8 channels of EMG data. Due to the restrictions of the recording hardware discussed in the text, a modified recording scheme was used for the offline evaluation. Online evaluation: In this case, the two initial training sets were recorded using the Magma-Box-Setup. In the test/evaluation phase, a single run was recorded using the USB-Box-Setup. The physiological data was processed in real-time and the results recorded for subsequent analysis.

### 5.2. Data Acquisition

EEG and EMG data were recorded at 5000 Hz with a 128 electrode (extended 10–20 system) actiCAP system using four 32 channel BrainAmp DC amplifiers (BrainProducts GmbH, Munich, Germany) for the EEG and a BrainAmp ExG MR bipolar EMG-System for the 8 EMG channels. EMG was acquired using bipolar Ag/AgCl gel electrodes at four muscles of the right arm: M. brachioradialis (Channel 1), M. bizeps brachii (Channel 2), M. triceps brachii (Channel 3), and M. deltoideus (Channel 4). For comparison reasons, a similar electrode arrangement was attached to the left arm for Channels 5–8. Due to restrictions of the recording hardware, two different setups were used for data acquisition:

- Magma-Box Setup: In this setup, a Magma box with two Brain Products amplifier PCI cards were used, the first PCI card was used to acquire 128 channels of EEG, the second card to acquire 8 channels of EMG data. Electrodes I1, OI1h, OI2h, and I2 of the EEG were used for recording the electrooculogram, which is not considered in the following analysis. The Magma-Box setup was used for the offline evaluation and recording of the training data for the online evaluation.

- USB-Box Setup: In this setup, a Brain Products USB 2 Adapter was used to record the data. Since it is restricted to four BrainAmp DC amplifiers, three amplifiers were used for EEG data acquisition and one amplifier was used for EMG data acquisition.

To connect the actiCAP system in the USB-Box Setup to the ZynqBrain, the device drivers provided by the vendor have been ported to Linux. The maximal number of EEG and EMG channels used in the evaluation depends on the setup (see Figure 5 for details). During acquisition, the data was filtered initially by the recording hardware with a passband between 0.1 and 1000 Hz.

### 5.3. Evaluation Procedures

The evaluation scheme is shown in Figure 5. For the offline analysis, a 3-fold cross-validation was used in all evaluations. Two of the three recorded runs were used for training of the machine-learning and data-dependent signal processing methods. The remaining run was used for offline evaluation. This was performed for all possible combinations of training and evaluation runs. By using the predefined splits of the data as given by the runs, the temporal order inside each split is retained in order to apply the segment-based evaluation as shown in Section 5.4. In the online evaluation, only the first two runs were used for training, while the third run was used for online evaluation. No testing data was used in the training phase and vice versa.

For the offline evaluation, 124 EEG channels were recorded; in the offline analysis, we used a fixed number of 96 channels due to recording hardware restrictions discussed above.

The processing was performed in order to detect and combine three different potentials in EEG or EMG data, as shown in Figure 2. To achieve this, it consists of four DFHWAs with specific tasks, as described below. In all cases, the data acquisition was performed using pySPACE, which is capable to read data from files (for offline analysis) or acquire and process data directly from EEG systems (for online analysis). To evaluate the DFHWAs, the data was then transferred to the DFHWAs implemented in the PL via DMA. Furthermore, all acquired data was recorded in files for subsequent analysis.

Three key metrics were of interest for the study provided in this manuscript: classification performance, computing performance, and power consumption.

### 5.4. Classification Performance

We analyze the classification performance for the following purposes. First, we want to determine the number and combination of required channels that give the best results. The processing time of all systems and the required PL resources of the DFHWAs depends on the number of channels. Hence, a small number of channels can result in a low processing time, but can also affect the classification performance. Second, we want to compare the different modalities and combinations regarding their

classification performance. Third, as discussed above, the CPU implementations were using double precision floating-point arithmetic, while the FPGA-based system was using fixed-point arithmetic. Since fixed-point arithmetic can result in numerical problems, a further concern was that this might decrease the classification accuracy.

We used a segment-based method for performance evaluation for MRCP and EMG (see Figure 3), since this provides the possibility to evaluate the system under conditions close to a real application case, where a movement prediction system should perform an ongoing analysis of the physiological data. For the P300 detection, we use segments of the length of 1 s subsequent to each Standard and Target marker.

Regarding the classification performance, we are interested in three performance measures: the overall classification performance in terms of Balanced Accuracy (BA), as well as the types of errors as given by the False Negative Rate (FNR) and False Positive Rate (FPR). The BA is given by the arithmetic mean of the True Positive Rate (TPR)/recall and True Negative Rate (TNR)/specificity, i.e.,

$$BA = \frac{1}{2}TPR + \frac{1}{2}TNR,$$

with $TPR = \frac{TP}{TP+FN}$ and $TNR = \frac{TN}{TN+FP}$, where *TP, TN, FP, FN* denote the true positives, true negatives, false positives and false negatives, respectively. The BA was selected for the performance assessment, as it allows a simple interpretation when the class frequencies are highly imbalanced [156] as it is the case here, where the segment-based evaluation results in a class-ratio of $\approx$1:110 for movement vs. rest. Correspondingly, the FPR and FNR are given by $FNR = \frac{FN}{TP+FN}$ and $FPR = \frac{FP}{TN+FP}$.

### 5.5. Computing Performance: Reference Systems for Comparison

The computing performance of the proposed system (denoted as FPGA-based system) was compared to the following CPU-based reference systems:

- A mobile processor-based system (the dual core ARM CPU of the ZC7030 PS, running at 1 GHz, 512 MB DDR SDRAM at 533 MHz), denoted as Mobile CPU-based Reference System (MRS) in the following.
- A standard desktop PC with an 8-core Intel(R) Core(TM) i7 CPU that was running at 3.07 GHz and a Linux Ubuntu operating system, version 14.4. denoted as Standard-PC Reference System (SRS) in the following.

In preliminary performance investigations, we identified the application of the anti-alias FIR filter applied as part of the decimation operation (see Section 4) as the computationally most expensive step. Hence, we used OpenMP [157] to apply a channel-wise parallelization of the filtering procedure.

All unnecessary processes were stopped to avoid a distortion of the results. As stated above, we used pySPACE [123] for processing and evaluation of the software parts of all systems.

### 5.6. Measurement of Power Consumption

The power consumption is an important aspect for processing systems, especially if the systems should be used inside a rehabilitation device independent from a stationary power supply. The total input power consumption of all systems was measured using a Tektronix TCP312/TCPA300 current clamp/amplifier.

## 6. Results and Discussion

The results are divided into two parts. The first part (Sections 6.1–6.6) contains the results of the offline evaluation which was conducted to determine the optimal system configuration and parameters for the subsequent online evaluation session. Furthermore, the offline evaluation permits us to evaluate the classification accuracy of the single signals as well as their combinations. The second

part (Section 6.7) reports the results of the validation of the developed system in the subsequently performed online sessions.

*6.1. EMG-Based Movement Prediction: Classification Performance, Computing Time, Resource Utilization*

*Classification Performance:* A grid search was used for optimization of $W_{VF}$, $W_{AT}$, and $p$. Based on previous experience, the chosen parameter ranges for optimization were $W_{VF} \in \{10, 20, 50, 100, 200\}$ ms, $W_{AT} \in \{100, 200, 500, 1000, 2000, 5000\}$ ms, $p \in \{3, \ldots, 12\}$. Regarding the channel setup, we estimate the performance of each single channel as a baseline for comparison, and the following combinations of channels: $n_{COT} \in \{1, 2, 4\}$ for channels $1, \ldots, 8$ and $n_{COT} \in \{1, 2\}$ for channels $1, \ldots, 4$, where $n_{COT}$ denotes the number of channels that have to be over the computed threshold simultaneously (see Section 4.1).

To this end, the classification performances of eight single-channels and five different channel combinations (see Figure 6) were analyzed by a two-way repeated measures ANOVA with the channel setup (CS, 13 levels as described above) and hardware architecture (HA, 2 levels: CPU-based system with double precision floating point vs. FPGA-based system with fixed-point arithmetic) as within-subjects factors. Greenhouse-Geisser correction was applied when necessary and the corrected $p$-value is reported. We performed a post-hoc analysis to compare the channel setups and hardware architecture. Bonferroni-Holm correction was applied for multiple comparisons.

We obtained a significant main effect for CS [$F(12, 276) = 211.5, p < 0.001$], but no effect of HA [$F(1, 23) = 2.2, p = n.s.$]. The best channel setups are '$1 \in [1, \ldots, 4]$' and '$1 \in [1, \ldots, 8]$' [these setups vs. all other setups: $p < 0.01$]. There was no significant difference between CPU and FPGA.

Furthermore, we observe a significant interaction between CS and HA [$F(12, 276) = 6.2, p < 0.001$]. We found no significant differences between both types of hardware architectures for all channel setups [$p = n.s.$], except for channel 1, and for all combinations except '$2 \in [1, \ldots, 4]$'.

As expected, the channel placed on the left arm show worse performance compared to the other channel setups [$p < 0.001$]. Combinations of multiple channels improve the performance, the best performance is obtained by channel setups '$1 \in [1, \ldots, 8]$', '$2 \in [1, \ldots, 8]$' and '$4 \in [1, \ldots, 4]$'. Based on these results, all subsequent investigations related to the EMG based movement prediction are based on the setup '$1 \in [1, \ldots, 4]$'.
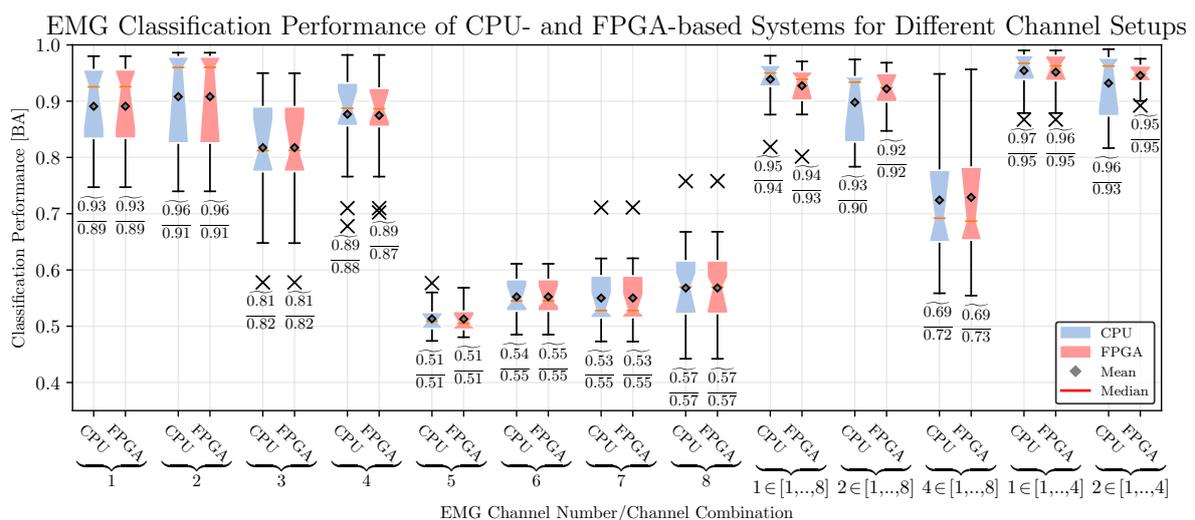


**Figure 6.** Classification performance for the prediction of movements based on the EMG. The median ($\widetilde{BA}$) and mean ($\overline{BA}$) classification performance is reported for each case.

*Computing Time:* Computing times were measured using all 40 ms segments from datasets of the offline study, in total this data consists of $N = 381{,}991$ segments. All measured times contain the time spent in the core algorithmic kernel and the time required to make the results accessible to the

Python-based top-level framework pySPACE, e.g., for the FPGA-based system this includes the time required to perform the data transfer between main memory and hardware accelerator.

The required time to process a segment of the proposed system (FPGA) and the reference systems (MRS and SRS, see Section 5.5) is shown in Figure 7 (left part). We can observe that the required time to process a segment depends on the number of channels and the utilized computing device. For all channel setups, the MRS requires the highest amount of time (2.431–15.97 ms, depending on the number of channels) and has the highest variability of time ($\sigma_t \approx 2.7$ ms). The SRS provides a significant speedup over the MRS system (5.6–11.8×, depending on the number of channels) and has a smaller variability of the processing time ($\sigma_t \approx 0.6$ ms). The highest performance and lowest variability of processing time can be obtained by the FPGA-based system (speedup over SRS: 9.0–21.6×, speedup over MRS: 50.6–253.5×, $\sigma_t \approx 0.003$ ms). The required time to process a segment represents a delay that might be, depending on the actual application and the related time constraints, unacceptable.
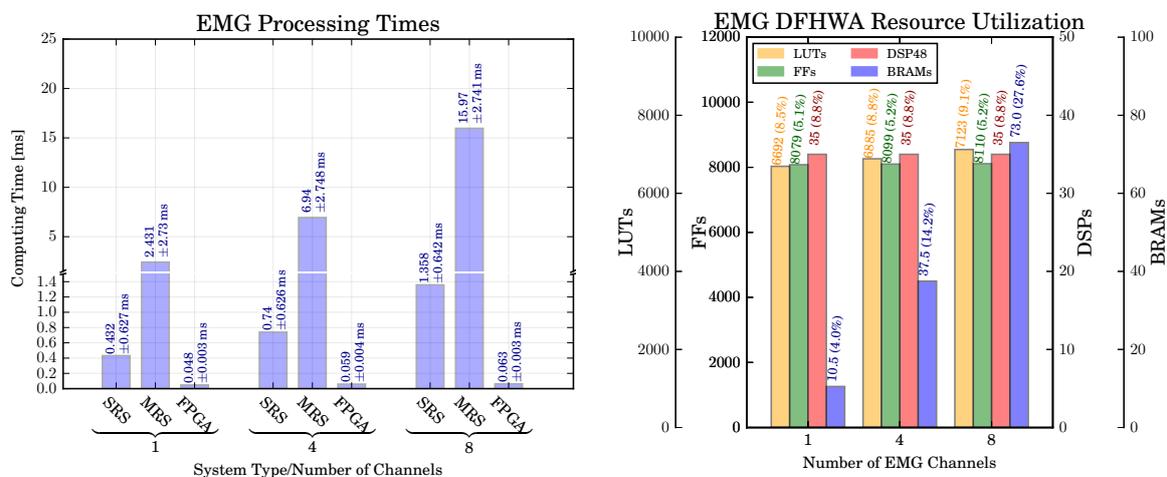


**Figure 7.** Left: Computation time for different computing setups for the detection of the movements based on the EMG. Right: Resource utilization of the Dataflow Hardware Accelerator (DFHWA) for different numbers of EMG channels. The percentage values refer to the amount of resources provided by the Zynq ZC7030 (see Section 3.1) (independently utilized 18 kb Block Random Access Memorys (BRAMs) are reported as 0.5 BRAMs).

*Resource Utilization:* The resource utilization is shown in Figure 7 (right part). It can be observed that there is only a small increase in the amount of utilized LUTs, FFs, and DSP48 for an increasing number of channels. In contrast to this, the increase of utilized BRAMs is more pronounced (10.5 BRAMs are required for a single channel, 37.5 BRAMs for four channels, and 73 BRAMs for eight channels). The reason for this increase of utilized BRAMs is the requirement to increase the buffer sizes of the variance filter and adaptive threshold computations to compute the sliding mean and variance (see Section 4.1), whereas the actual computational units are used in a multiplexed fashion for different data samples and channels.

*6.2. MRCP-Based Movement Prediction: Classification Performance, Computing Time, Resource Utilization*

*Classification Performance:* The classification performance for different numbers of channels according to the extended 10–20 system and both hardware architectures was analyzed using a repeated measures ANOVA with two within-subjects factors: numbers of channels ($NC$, 4 levels: $n_c$ channels with $n_c \in \{32, 64, 96, 124\}$) and type of hardware architecture ($HA$, 2 levels: $\{CPU, FPGA\}$). Greenhouse-Geisser correction was applied when necessary. Bonferroni-Holm correction was applied for multiple comparisons.

The obtained classification performance is shown in Figure 8. A significant main effect can be observed for $NC$ [$F_{(2.07, 47.6)} = 12.7$, $p < 0.01$], but not for $HA$ [$F_{(1, 23)} = 1.18$, $p = n.s.$]. The effect

of number of channels is not straightforward for each architecture type. In our application, the highest BA is achieved with 96 channels (except for 124 channels) [$n_c = 96$ vs. $n_c = 32$: $p < 0.001$, $n_c = 96$ vs. $n_c = 64$: $p < 0.001$, $n_c = 96$ vs. $n_c = 124$: $p = n.s.$].
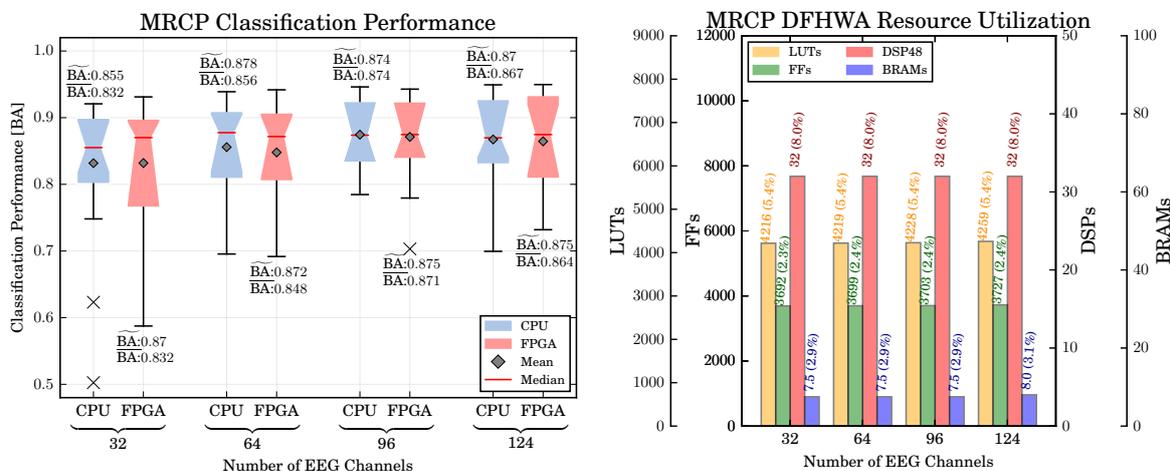


**Figure 8.** Left: Classification performance for the detection of the MRCP for CPU and FPGA for 32–124 channels. The median ($\widetilde{BA}$) and mean ($\overline{BA}$) classification performance is reported for each case. Right: Resource utilization of the DFHWA for different numbers of EEG channels. The percentage values refer to the amount of resources provided by the Zynq ZC7030.
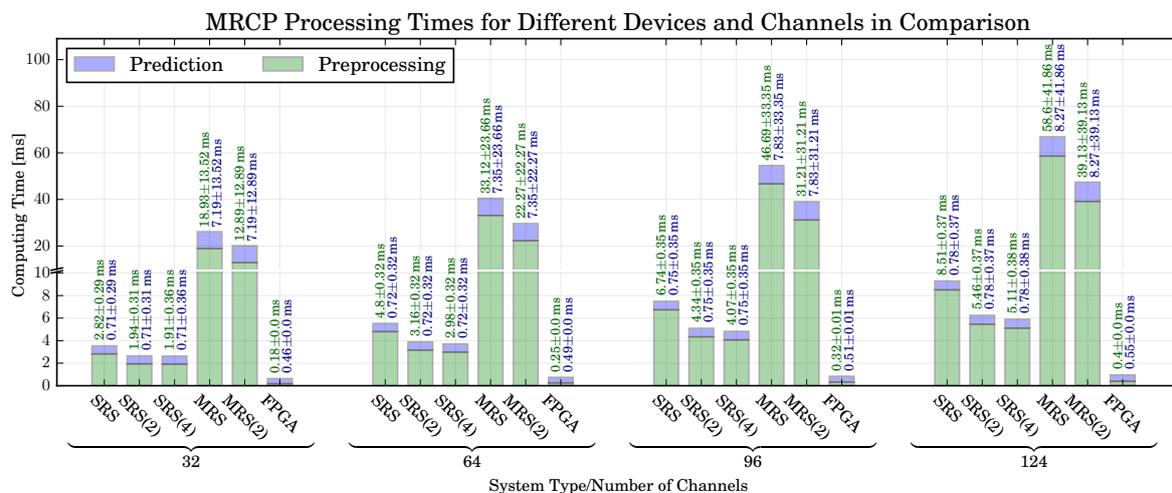


**Figure 9.** Computing times of different computing setups for 32–124 channels for the detection of the Movement Related Cortical Potential (MRCP). For the Standard-PC Reference System (SRS) system, the number of threads are put in parenthesis.

*Computing Times:* The computing times for the different reference systems discussed in Section 5.5 are shown in Figure 9. Reported is the amount of time required to perform a single prediction, i.e., to process 200 ms of EEG data. Two values are reported for each setup: the time required for the preprocessing (which corresponds to the Preprocessing DFHWA shown in Section 4.2) as well as the time for the actual prediction (which corresponds to the MRCP Processing DFHWA). It can be observed that, independent from the number of channels, the processing times of the SRS are sufficiently fast to predict movements online, i.e., they require less than 40 ms of time to process a segment of 40 ms of EEG data. We can only observe a small speedup by parallelizing the preprocessing (the smallest speedup of 47% is obtained for 32 channels, the highest speedup of 66% can be obtained for 124 channels). The reason for this observation is a high proportion of serial code related to object construction and

data exchange between nodes in pySPACE. A separate investigation (not shown) revealed that the speedup increases if the size of the data segments is increased. However, this would also result in an increase of the latency related to each prediction. Hence, a larger segment size is not feasible for the application of online movement prediction.

The MRS requires a significantly higher amount of time for each prediction. For a small number of channels ($n_c \in \{32, 64\}$), it is possible to use the MRS for online prediction (i.e., less than 40 ms are required to process a segment). However, for these channel setups we obtain a classification performance worse than for $n_c \in \{96, 124\}$. Hence, an optimal system configuration would consist of a configuration with a higher number of channels, i.e., $n_c \in \{96, 124\}$. For such a configuration the MRS does not provide a sufficient amount of computing performance for an online application.

If the DFHWAs are used, the computation times can be considerably reduced and the time constraints for real-time prediction are met. The achievable speedup depends on the number of channels. For 32 channels, achievable speedup is the lowest (FPGA vs. SRS: $\approx$ 4–5.5, FPGA vs. MRS $\approx$ 31.4–40.8), the highest speedups can be achieved for 124 channels (FPGA vs. SRS: $\approx$ 9.8–6.2, FPGA vs. MRS: $\approx$ 49.9–70.4).

*Resource Utilization:* A further concern was again the amount of required resources in relation to the number of channels. The resource utilization for different PL resources and number of EEG channels is shown in Figure 8 (right). It can be observed that the increase of resources for an increasing number of EEG channels is generally low, the highest increase of resources can be observed for the required LUTs in the preprocessing part of the MRCP (32 vs. 124 channels: 249.7%). A smaller increase can be observed for the FFs (32 vs. 124 channels: 10.3%), while no increase can be observed for the BRAMs and DSP48 slices. The reason for this observation is attributable to the *delay* scaling technique applied to scale the system to different numbers of channels. As discussed in Section 3.2, the application of delay scaling increases the number of registers in a DFHWA (which are implemented as LUTs and FFs), but not the number of multipliers and adders (which are mapped to DSP48 slices).

### 6.3. P300 Detection: Classification Performance, Computing Time, Resource Utilization

*Classification Performance:* The classification performance for the CPU- and FPGA-based systems of the P300 is shown in Figure 10 (left part). In addition, the The mean and median classification performance over all numbers of channels and both hardware architectures are 0.91 $\pm$ 0.04 and 0.92, respectively. Hence, in comparison to the MRCP, the P300 can be detected more reliable. For the P300 detection, we also investigate the effect of the number of channels and hardware architecture on the classification performance. To this end, we perform a two-way repeated measures ANOVA with the numbers of channels $n_c \in \{32, 64, 96, 124\}$ and the hardware architecture $HA = \{CPU, FPGA\}$ as within-subjects factors. Similar to the MRCP evaluation, we obtain no difference of the classification performance between both hardware architectures [$F(1, 23) = 1.4, p = n.s.$]. However, in contrast to the MRCP evaluation we obtain also no difference in the classification performance between the numbers of channels [$F(3, 69) = 1.5, p = n.s.$].
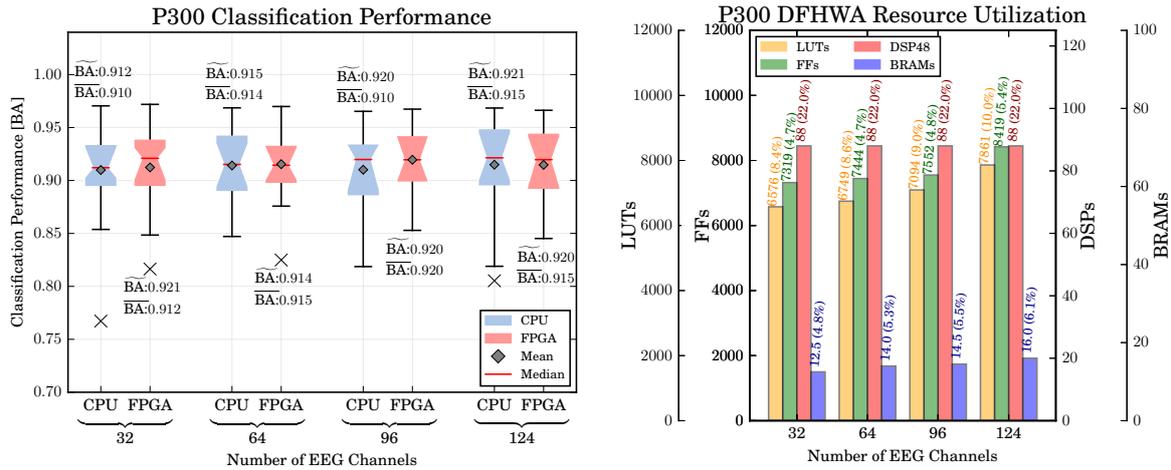
**Figure 10.** Left: Classification performance of the P300 detection for CPU and FPGA for 32–124 channels. Right: Resource utilization of the DFHWA for different numbers of EEG channels. The percentage values refer to the amount of resources provided by the Zynq ZC7030.

*Computing Time:* The P300 processing times are shown in Figure 11. The computing time refers to the time required to classify a single segment of EEG data. As discussed in Section 4.4, a corresponding segment has the length of 1000 ms. It can be observed that, similar to MRCP processing times, the time depends on the number of channels and the utilized system. The highest amount of time is required by the MRS. However, in this case the MRS is capable to perform online processing if less than 124 channels are used. Similar to the MRCP and EMG processing, the FPGA-based solution provides a significant speedup over the MRS and SRS.
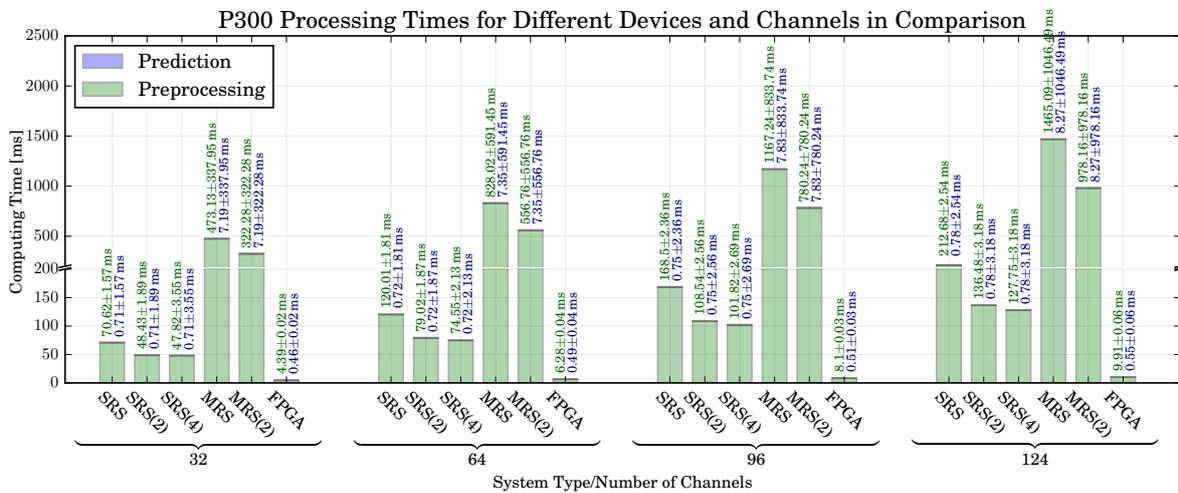


**Figure 11.** Computing times of different computing setups for 32–124 channels for the detection of the P300.

*Resource Utilization:* The resource utilization for different PL resources is shown in Figure 8 (right). Similar to the MRCP DFHWA resource utilization, it can be observed that the increase of resources for an increasing number of channels is generally low. However, due to higher number of processing steps, the resource utilization is higher than the MRCP DFHWA resource utilization.

### 6.4. Hybrid Movement Prediction: Classification Performance, Prediction Time

The last step in the offline analysis was to determine if the combination of different modalities as described in Section 3.5 can improve the classification performance. The classification performance shown in Figures 6 and 8 reveals that the EMG-based movement prediction provides a higher

classification performance than the MRCP-based movement prediction. However, it is well known that the EEG allows a prediction of movements earlier than the EMG. Hence, an important point for the usage in a rehabilitation device is to study the effect combining multiple predictions on the overall classification performance, reliability and prediction time. The proposed system provides the possibility to process several data streams in parallel. Hence, it permits to apply the hybrid combinations described in Section 3.5.

*Classification Performance:* The classification performance of the hybrid combinations for the CPU- and FPGA-based systems is shown in Figure 12. For comparison, we performed a three-way repeated measures ANOVA with the movement prediction method (method, 4 levels: EMG, MRCP, PaM, PaE), combination with P3 (P3, two levels: NP3, CP3), and hardware architecture (HA, 2 levels: CPU, FPGA) as within-subjects factors. Greenhouse-Geisser correction was applied when necessary and the corrected $p$-value is reported. For post-hoc comparisons, we performed paired t-tests. Bonferroni-Holm was applied for multiple comparisons. We obtained significant main effects for method [$F(3, 69) = 43.6, p < 0.001$] and P3 [$F(1, 23) = 222.4, p < 0.001$], but no effect of HA [$F(1, 23) = 1.5, p = n.s.$].

The best method is EMG, followed by MoE, MRCP, and MaE [$p < 0.001$ for all pairs of methods]. A combination with the P3 improves the classification performance [CP3 vs. NP3: $p < 0.001$]. There was no difference in classification performance between CPU and FPGA [$p = n.s.$].
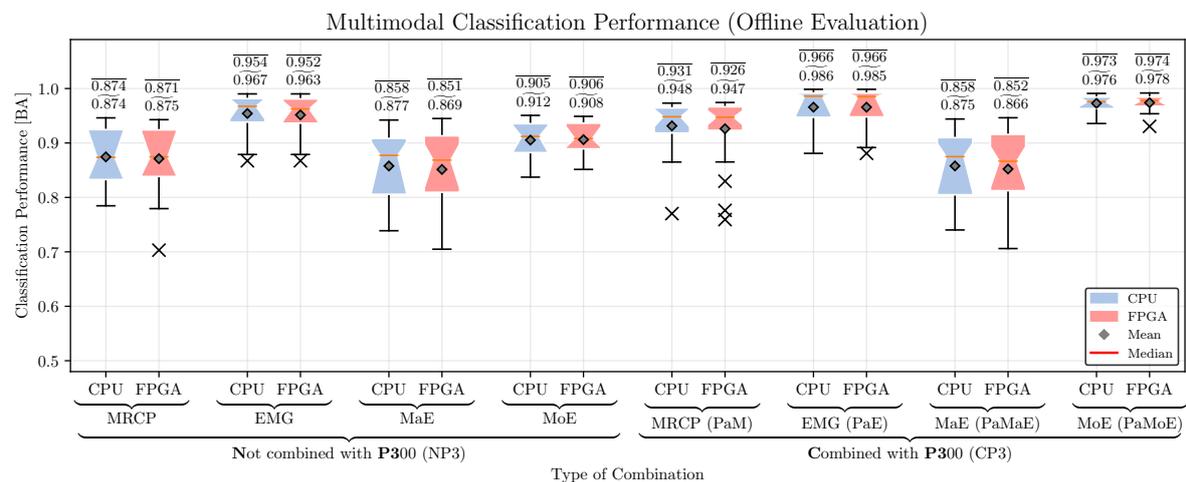


**Figure 12.** Classification Performance for Hybrid Movement Prediction.

Furthermore, we observe a significant interaction between movement prediction method and P3 combination [$F(3, 69) = 80.6, p < 0.001$]. The best performance is obtained by EMG, followed by MoE, irrespective if it is combined with P3 or not [$p < 0.05$]. However, for the MRCP we observe an increased performance compared to MaE only when it is combined with the P3 [MRCP vs MaE: $p < 0.05$ for P3, $p = n.s.$ for NP3]. P3 improves the classification performance for all methods except for MaE [$p = n.s.$ for MaE, $p < 0.05$ for other methods]. The hardware architecture interacts with neither P3 nor methods.

Finally, there was no interaction between HA, P3 and methods [$F(3, 69) = 1.3, p = n.s.$]. The post-hoc analysis reveals that the combination with the P3 improves the classification performance for all methods [$p < 0.001$], except for MaE [$p = n.s.$]. This pattern is observed for both architectures. Furthermore, for both CP3 and NP3 there was no difference in classification performance between both architectures for all methods [$p = n.s.$]. A combination with the P300 improves the classification performance for all methods except for MaE [$p = n.s.$ for MaE, $p < 0.01$ for other methods]. This pattern was observed again for both architectures.

Another important point in the direct comparison of different combinations is a consideration of the errors rates and precision. The mean FNRs, FPRs and recall for the combinations and both hardware architectures is given in Table 2. For comparison, we performed three-way ANOVAs with movement

detection method, combination with P300, and hardware architecture as within-subjects factors, which were performed separately for the FNR, FPR and precision as dependent variables, respectively.

For the FNR, we observe a significant main effect for the movement prediction method [$F(3, 69) = 72.8$, $p < 0.001$], but neither for combination with P300 [$F(1, 23) = 2.7, p = n.s.$] nor hardware architecture [$F(1, 23) = 0.8, p = n.s.$]. The lowest FNR is obtained by MoE, followed by EMG, and MRCP; the highest FNR is obtained for MaE [$p < 0.001$ for all pairs of methods].

For the FPR, we observe significant main effects for movement prediction method [$F(3, 69) = 131.2$, $p < 0.001$] and combination with P300 [$F(1, 23) = 240.5, p < 0.001$], but not for $HA$ [$F(1, 23) = 0.2, p = n.s.$]. Regarding the FPR, the best method is EMG, followed by MoE, MRCP, and MaE [$p < 0.001$ for all pairs of methods]. A combination with the P3 improves the FPR [CP3 vs. NP3: $p < 0.001$]. There was no difference in FPR between CPU and FPGA [$p = n.s.$].

**Table 2.** Mean False Negative Rate (FNR), False Positive Rate (FPR = 1-Recall) and Precision for the Hybrid Movement Prediction.

| | Not Combined with P300 (NP3) | | | | | | | | Combined with P300 (CP3) | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | MRCP | | EMG | | MaE | | MoE | | MRCP | | EMG | | MaE | | MoE | |
| | CPU | FPGA | CPU | FPGA | CPU | FPGA | CPU | FPGA | CPU | FPGA | CPU | FPGA | CPU | FPGA | CPU | FPGA |
| Mean FNR | 0.104 | 0.116 | 0.063 | 0.063 | 0.278 | 0.290 | 0.021 | 0.019 | 0.116 | 0.127 | 0.063 | 0.062 | 0.282 | 0.293 | 0.021 | 0.020 |
| Mean FPR | 0.147 | 0.142 | 0.029 | 0.034 | 0.006 | 0.007 | 0.169 | 0.168 | 0.022 | 0.021 | 0.005 | 0.006 | 0.002 | 0.002 | 0.033 | 0.033 |
| Mean Precision | 0.108 | 0.109 | 0.332 | 0.303 | 0.577 | 0.541 | 0.123 | 0.121 | 0.422 | 0.421 | 0.711 | 0.695 | 0.826 | 0.810 | 0.403 | 0.408 |

For the FPR, we observe again a significant interaction between movement prediction method and P3 combination [$F(3, 69) = 120.8, p < 0.001$]. A further analysis reveals that the combination with the P3 reduces the FPR for all methods [$p < 0.001$]. This pattern is observed for both architectures. Furthermore, for both NP3 there was a difference in FPR between both architectures for all methods [$p = n.s.$]. A combination with the P300 improves the FPR for all methods [$p < 0.01$]. This pattern was observed again for both architectures.

The precision depends substantially on the applied signal combination method. For the precision, we observe significant main effects for movement prediction method [$F(3, 69) = 205.5, p < 0.001$], combination with P300 [$F(1, 23) = 797.4, p < 0.001$], and for $HA$ [$F(1, 23) = 4, p < 0.05$]. Regarding the precision, the best method is MaE, followed by EMG, MRCP, and MoE [$p < 0.001$ for all pairs of methods except MRCP vs. MoE: $p = n.s.$]. A combination with the P3 improves the precision [CP3 vs. NP3: $p < 0.001$]. For the precision, we observed a difference between CPU and FPGA [$p < 0.01$].

Furthermore, we observe again a significant interaction between movement prediction method and P3 combination [$F(3, 69) = 12.0, p < 0.001$] and additionally a significant interaction between method and $HA$ [$F(3, 69) = 6.5, p < 0.001$]. A further analysis reveals that the combination with the P3 improves the precision for all methods [$p < 0.001$]. This pattern is observed for both architectures. Moreover, for NP3 there was a difference in precision between both architectures for EMG and MaE [$p < 0.01$], but not for MRCP and MoE [$p = n.s.$]. For CP3, there was no difference in precision for both architectures for all methods [$p = n.s.$]. A combination with the P300 improves the precision for all methods [$p < 0.01$]. This pattern was observed again for both architectures.

*Summary:* We observe that the MRCP-based movement prediction is less accurate than the EMG-based prediction. This can also be observed for the individual trials shown in Figure 13. The reason for the lower accuracy might be the detection of a movement planning or intention that does not result in a movement execution [10,158,159]. The classification performance is improved by the combination of the MRCP with the P300 and/or EMG. Depending on the type of combination, different properties regarding FPR, FNR and precision can be observed. The combination of the MRCP with the EMG by a logical and decreases the FPR, but increases the FNR. The opposite can be observed for the FNR and precision. In an additional combination with the P3000, we observe a decrease of the FPR, but an increase of the FNR for the MRCP and MoE. While the decrease of the FNR

is only small for the MRCP, we can observe a significantly higher increase for MaE. In this case, up to 25% of the movements are missed. A decrease of the FPR can be observed for the MoE combination. The precision is increased in general by a combination with the P300.
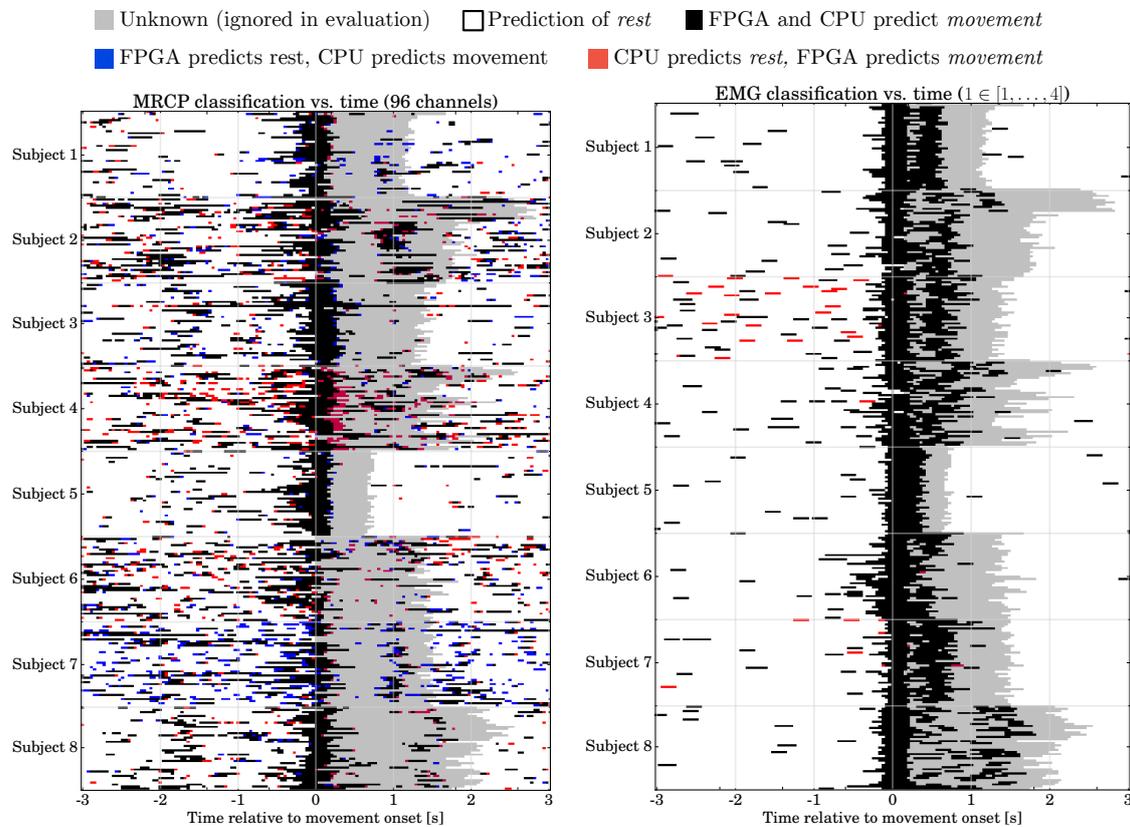


**Figure 13.** Predictions relative to the movement onset of the CPU and FPGA-based systems for the MRCP- and EMG-based movement prediction. The predictions of movements for the interval [−3 s, 3 s] is depicted relative to each movement onset for MRCP-based predictions (left) and EMG-based predictions (right). Predictions of rest are left white, the unknown class subsequent to each movement is shown with a grey background. The prediction of a movement is encoded as follows: the prediction of a movement by the CPU-based systems as well as the FPGA-based system is shown in black. The prediction of a movement by the CPU-based system but not the FPGA -based system is shown in blue, the opposite case is shown in red.

## 6.5. Prediction Time

An important point in the comparison of different modalities is not only the classification accuracy, but also the capability to predict an upcoming movement as early as possible. We define according to [25] the prediction time as the earliest point in time in the pre-movement unknown phase that predicts a movement onset. The mean ($\mu_t$) and 75/50/25% quartile ($Q_t^{75}, Q_t^{50}, Q_t^{25}$) prediction times for the EEG, EMG and all investigated combinations are shown in Table 3. As discussed in Section 6.2, the MRS is not able to predict movements based on the MRCP due to insufficient computing power. The provided times include the latency caused by the time required to compute the prediction according to Figures 6, 9 and 11. For a visualization of the prediction times relative to the movement onset see also Figures 2 and 14. It can be observed that the modality and type of combination affects the prediction time.
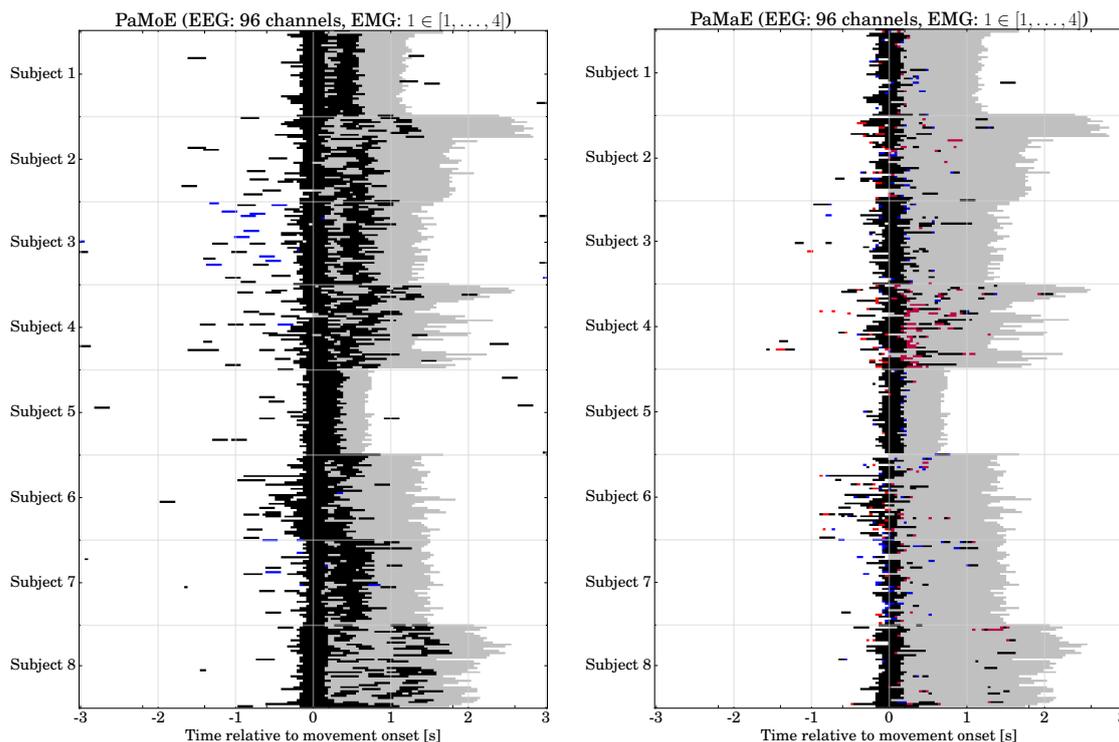
**Figure 14.** Predictions relative to the movement onset of the CPU and FPGA-based systems for the PaMaE and PaMoE combinations. For a more detailed figure description, see Figure 13.

**Table 3.** Prediction times in ms *before* movement onset (N/A: not available).

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Movement Prediction Times** | | | | | | | | | | | | | | | | |
| | | MRCP | | | | EMG | | | | MaE | | | | MoE | | |
| System | $\mu_t$ | $Q_t^{75}$ | $Q_t^{50}$ | $Q_t^{25}$ | $\mu_t$ | $Q_t^{75}$ | $Q_t^{50}$ | $Q_t^{25}$ | $\mu_t$ | $Q_t^{75}$ | $Q_t^{50}$ | $Q_t^{25}$ | $\mu_t$ | $Q_t^{75}$ | $Q_t^{50}$ | $Q_t^{25}$ |
| SRS | 515 | 234 | 514 | 834 | 140 | 39 | 79 | 199 | 124 | 34 | 74 | 154 | 585 | 274 | 594 | 914 |
| SRS (2) | 516 | 235 | 515 | 835 | 140 | 39 | 79 | 199 | 125 | 35 | 75 | 155 | 586 | 275 | 595 | 915 |
| SRS (4) | 516 | 235 | 515 | 835 | 140 | 39 | 79 | 199 | 125 | 35 | 75 | 155 | 586 | 275 | 595 | 915 |
| MRS | N/A | | N/A | | 134 | 33 | 73 | 193 | N/A | | N/A | | N/A | | N/A | |
| MRS (2) | N/A | | N/A | | 134 | 33 | 73 | 193 | N/A | | N/A | | N/A | | N/A | |
| FPGA | 542 | 239 | 559 | 879 | 136 | 40 | 80 | 200 | 129 | 39 | 79 | 159 | 596 | 269 | 599 | 919 |
| | | PaM | | | | PaE | | | | PaMaE | | | | PaMoE | | |
| System | $\mu_t$ | $Q_t^{75}$ | $Q_t^{50}$ | $Q_t^{25}$ | $\mu_t$ | $Q_t^{75}$ | $Q_t^{50}$ | $Q_t^{25}$ | $\mu_t$ | $Q_t^{75}$ | $Q_t^{50}$ | $Q_t^{25}$ | $\mu_t$ | $Q_t^{75}$ | $Q_t^{50}$ | $Q_t^{25}$ |
| SRS | 314 | 194 | 234 | 354 | 125 | 34 | 74 | 194 | 190 | 34 | 74 | 154 | 340 | 194 | 234 | 394 |
| SRS (2) | 315 | 195 | 235 | 355 | 126 | 35 | 75 | 195 | 191 | 35 | 75 | 155 | 340 | 195 | 235 | 395 |
| SRS (4) | 315 | 195 | 235 | 355 | 126 | 35 | 75 | 195 | 191 | 35 | 75 | 155 | 341 | 195 | 235 | 395 |
| MRS | N/A | | N/A | | N/A | | N/A | | N/A | | N/A | | N/A | | N/A | |
| MRS (2) | N/A | | N/A | | N/A | | N/A | | N/A | | N/A | | N/A | | N/A | |
| FPGA | 319 | 199 | 239 | 359 | 129 | 39 | 79 | 199 | 199 | 39 | 79 | 159 | 355 | 199 | 239 | 399 |

For both the SRS and the FPGA-based system, the MRCP can predict upcoming movements earlier than the EMG (difference of means: SRS: 375 ms/FPGA: 406 ms, difference of medians: 442 ms/479 ms). The MoE provides a small increase of the prediction time in comparison to the MRCP (difference of means: 70/54 ms, difference of medians: 80/40 ms). However, if an and-based combination (MaE, PaMaE) is used, the prediction times are reduced.

*6.6. Power Consumption*

The power consumption for all systems and the signal combinations is shown in Table 4. For the FPGA-based system we evaluate the minimal configuration to process the desired physiological signals, i.e., the PL section of the Zynq contains only the DFHWAs required for the specific setup.

It can be observed that the FPGA-based system requires considerably less power than the SRS when idling, but also during computations. However, the power consumption is higher than the power consumption of the MRS. This effect is caused by the programmable logic: the power consumption increases proportional to the amount of utilized resources. Hence, when the system is equipped with multiple DFHWAs, the idle as well as computing power consumption increases.

For all systems, the power consumption increases when multiple physiological signals are processed. The highest power consumption of the SRS and MRS as well as the FPGA-based system is required if the MRCP and P300 detection as well as EMG processing are performed in parallel (i.e., the PaMaE and PaMoE combinations). Under these conditions the chosen LiFePO batteries (see Section 3.1) provide an operating time of approx. 3 h, which would be sufficient for most rehabilitation applications.

**Table 4.** Power consumption, reference versus FPGA-based system with different configurations.

| SRS | Idle | 1 Core | 2 Cores | 4 Cores |
|---|---|---|---|---|
| EMG | 119.8 W | 130.2 W | 130.2 W | 130.2 W |
| MRCP | 119.8 W | 126.0 W | 126.4 W | 127.6 W |
| MaE, MoE | 119.8 W | 133.0 W | 133.3 W | 135.2 W |
| PaM | 119.8 W | 129.6 W | 130.2 W | 132.1 W |
| PaE | 119.8 W | 132.6 W | 132.6 W | 132.6 W |
| PaMaE, PaMoE | 119.8 W | 136.8 W | 138.1 W | 138.3 W |

| MRS | Idle | 1 Core | 2 Cores |
|---|---|---|---|
| EMG | 2.98 W | 3.28 W | 3.23 W |
| MRCP | 2.98 W | 3.23 W | 3.23 W |
| MaE, MoE | 2.98 W | 3.30 W | 3.61 W |
| PaM | 2.98 W | 3.29 W | 3.60 W |
| PaE | 2.98 W | 3.31 W | 3.64 W |
| PaMaE, PaMoE | 2.98 W | 3.32 W | 3.66 W |

| FPGA | Idle | Computing |
|---|---|---|
| EMG | 3.45 W | 4.12 W |
| MRCP | 3.45 W | 4.12 W |
| MaE, MoE | 3.49 W | 4.41 W |
| PPaM | 3.58 W | 4.48 W |
| PaE | 3.53 W | 4.49 W |
| PaMaE, PaMoE | 3.59 W | 4.51 W |

*6.7. Final Hybrid System: Verification in Application*

Based on the preliminary investigations (Section 6.2), 96 channels have been used for the MRCP and P300 configuration of the final hybrid system. Correspondingly, for EMG the highest classification performance and only a modest resource utilization is obtained for the channel combination '$1 \in [1, \dots, 4]$', so that this channel setup is chosen. During the online evaluation we recorded the outputs of all single signals and their combinations. In addition, the raw data was recorded for a subsequent comparative offline analysis with a CPU-based system, which has been performed to obtain a reference for these datasets.

The classification performance obtained by using the FPGA-based system and the subsequent reference analysis with a CPU-based system is shown in Figure 15. We obtain a decreased classification

performance in comparison to the offline evaluation (Figure 12) for the MRCP, but a similar performance for the EMG. The combination with other potentials can attenuate this reduction, i.e., we obtain a performance that is comparable to the offline performance. Similar to the offline analysis, we observe no difference in classification performance of the FPGA-based system in comparison to the CPU-based reference performance. Hence, the obtained results demonstrate that the proposed system can be used for online prediction under experimental conditions.
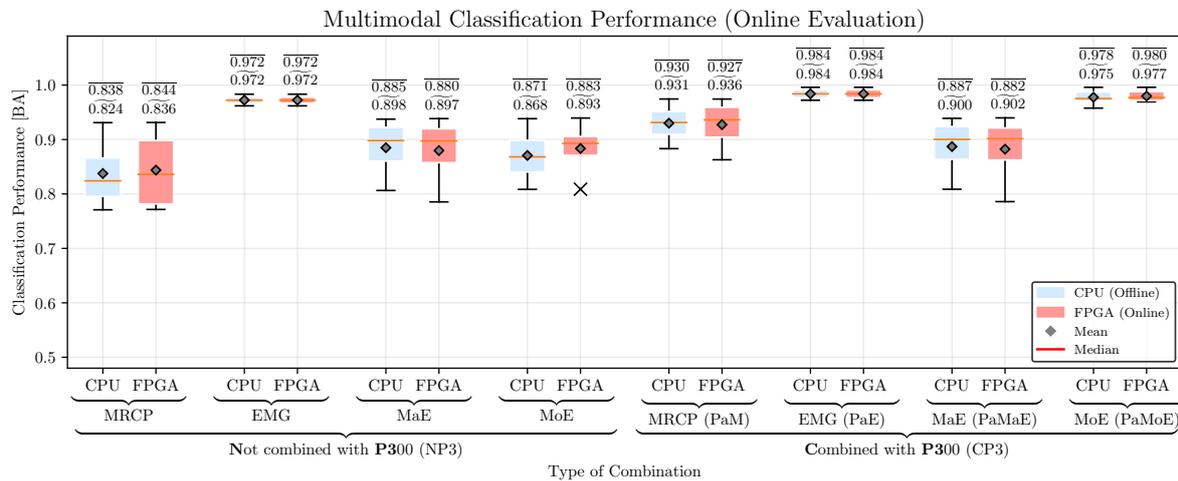


**Figure 15.** Classification Performance for Hybrid Movement Prediction for the Online Evaluation.

The overall resource utilization of the complete system is shown in Table 5. In addition to the PL resources required to implement the DFHWAs, resources needed to implement all auxiliary components (e.g., AXI bus system, DMA cores) are included in Table 5. The percentage values refer to the ZC7030 resources shown in Table 1. It can be observed that the resource utilization of the PL is less than 35% for all components. Hence, it would be possible to integrate further DFHWAs or other components into the system. Regarding the LUTs and FFs utilization of the complete system it can be observed that a significant proportion is required for the implementation of the auxiliary components, which is not the case for the BRAMs and DSP48 slices.

**Table 5.** Resource utilization for the complete system.

| Type of Resource | LUTs | FFs | 36 kb Each | DSP48 |
|---|---|---|---|---|
| DFHWAs | 15 755 (20.0%) | 16 786 (10.7%) | 53.5 (20.1%) | 136 (34%) |
| Auxiliary Components | 6 084 (7.7%) | 7 529 (4.8%) | 7.5 (2.8%) | 0 (0%) |
| Total | 21 839 (27.8%) | 24 315 (15.4%) | 61 (23.0%) | 136 (34%) |

*6.8. Summary*

The obtained results show that the developed system can be used to predict upcoming movements before the actual physical movement onset. We confirm previous results from literature that an EEG-based movement prediction can predict movements earlier, but with a higher error rate than EMG [25]. For the investigated signals and signal combinations, we observed no significant difference in classification performance between the CPU- and FPGA-based systems in most cases. The parallel architecture of the system provides the possibility to process several streams of data in parallel and without the intervention of a CPU. This has two advantages. First, it provides the possibility to enhance the reliability of the system by combining the outputs from the MRCP and/or EMG-based DFHWAs. Second, it provides the possibility to choose the most suitable combination of predictions for the specific application. For instance, depending on the control algorithm of the rehabilitation device and the intended application, the EEG might be preferable since it can predict movements earlier in time, although it is less accurate than the EMG. Furthermore, in paralyzed or

highly paretic patients caused by, e.g., spinal cord injury, the application of an EMG-based movement prediction might be infeasible.

The DFHWAs for all investigated signals provide significant speedups in comparison to the chosen mobile reference solution, but also in comparison to a standard PC. Furthermore, the system has a significantly decreased power consumption in comparison to the standard PC. Moreover, the resource utilization provides the possibility to integrate further algorithms in the same device, e.g., further DFHWAs for additional physiological signals and source localization methods to allow not only the prediction of upcoming movements, but also the affected extremity and/or movement direction [160,161]. Based on the provided computing performance, a future possibility would also be the possibility to perform movement prediction with an even higher time resolution than in the actual system, e.g., shift the segments for MRCP and EMG detection by 1 ms and apply a meta-classifier on the individual predictions.

Another advantage of our FPGA-based system is the possibility to establish a direct interface to other electronic devices. This makes it possible to control or adapt an assistive or rehabilitation device directly in real-time without the need of a software-based interface, which might introduce additional delays.

### 6.9. Comparison to Previous Work

Although combining ERD/ERS or MRCPs with EMG is a promising approach for the development of hybrid BCIs [42], there are still only few BCI systems that rely on this combination. Hence, the first part of the following comparison is limited to systems that combine ERD/ERS or MRCPs with EMG or other modalities. In the BCI systems listed in the first part of this comparison, the data processing is performed in standard personal computers or laptops, in contrast to out FPGA-based approach. However, as discussed in Section 1, neurorehabilitation applications would benefit from mobile or embedded BCI systems. Although FPGAs are well suited to build such systems (see Sections 2.4 and 2.5), only few FPGA-based BCI systems exist. The second part of the comparison compares our system with FPGA-based BCIs.

Similar to our work, the combination of EEG and EMG for the detection of an arm has been investigated in [82]. Two different methods for the combination of EEG and EMG were investigated. It was shown that certain combinations obtain a better and more stable performance for the detection of movements in comparison to a single signal. Due to the EEG/EMG combination, the system can be used to compensate exhaustion or fatigue of the user. In contrast to our work, the subjects could move the left or right arm. However, the combination with further modalities, such as the P300, has not been investigated.

Multiple additional modalities next to EEG and EMG (e.g., eye gaze, Electrooculography (EOG), hand position) and their combinations to predict targets of human reaching motions were investigated in [84]. Similar to our work, it was shown that EEG predicts movements earlier that EMG, but with a lower accuracy than EMG. Furthermore, it was shown that EOG or eye tracking can also be used to improve the performance of EEG-based movement predictions. However, in contrast to our work, the study is limited to the combination of two modalities, a combination with the P300 has not been investigated.

In a previous work [25], we investigated the effect of combining EEG and EMG on the classification performance and reliability for the prediction of movements in an offline evaluation. However, in that work we neither considered the combination with further signals (i.e., the additional combination with P300), nor did we use a mobile FPGA-based system. Furthermore, we did not validate the approach in an online test.

In most other works, the combination of EEG and EMG is used for neurorehabilitation applications, e.g., to control orthoses or prostheses. For instance, in [83] it was shown that the realtime control of an upper limb wearable robot can be improved by such a combination. However, EEG was only used to compensate a lack of EMG, i.e., to estimate the torques of elbow and forearm

based on EEG signals, if the EMG is has not enough power. The work shows that the torque estimation accuracy decreases if an EEG/EMG combination is used. The study does not investigate the effect of the combination on classification performance, if movements should be predicted.

The control of active orthoses for gait rehabilitation based on EEG and EMG has been presented in [43]. The work proposes an approach for the control of active orthoses that incorporates motion intention recognition based on the detection of ERD/ERS to predict the gait phase. In addition, EMG was used to estimate the active torque generated by the patient's muscles. However, the control approach did also not consider the fusion of EEG and EMG to enhance the classification accuracy to predict movements.

A hybrid control approach for an upper limb prosthesis based on EEG and EMG was investigated in [86]. EEG was used to control the exoskeleton asynchronously, joint angles were estimated using EMG. The study shows that it is possible to estimate the joint angles with root-mean-square errors of less than $6°$. However, similar to the studies discussed above, it was not investigated whether a combination of EEG and EMG can be used to predict movements with an improved accuracy.

A combined EEG/EMG classification for hand and wrist movements in transhumeral amputees was investigated in an offline study in [87]. It was shown that the EEG/EMG combination outperforms the classification by either EEG or EMG. However, in contrast to the study in this paper, the aim was to differentiate between different movements instead of predicting upcoming movements. The authors point out that a miniature and portable system is required for future practical applications.

In the literature discussed above, the data processing was based on standard PCs. The aim of FPGA-based BCI systems is to replace a standard PC. In [107,109], Shyu et. al. present an FPGA-based system for the detection of SSVEPs. The system can be used for the control of a multimedia system [107] or a hospital bed [109]. Especially the tight integration of EEG signal processing and motor control to actuate the hospital bed represents an interesting opportunity for other rehabilitation devices. However, since these systems target the detection of SSVEPs, they have been designed to process only two EEG channels, and are hence not usable for the detection of, e.g., MRCPs.

The work in [108] presents a low-cost FPGA-based P300 speller. Similar to us, they achieve a classification performance that is comparable to a standard CPU. However, most parts of the system are implemented using softcore CPUs (only a band-pass Butterworth filter is implemented in hardware), and only seven EEG channels can be used. However, the aim of the system is to detect the P300, the detection of ERD/ERS or MRCP is not considered.

A recent work uses an FPGA-based prototype for fall prediction in everyday life [110], which is based on a previous software-implementation [85]. Similar to our work they combine EEG processing (for the detection of movement related potentials) with EMG and show in an offline evaluation that it is possible to obtain a small numerical error, despite the usage of fixed-point arithmetic in the FPGA. However, for the fall prediction, a fixed number of only 7 EEG and 8 EMG channels is sufficient; the classification performance, resource utilization and computing time for different numbers of channels has not been investigated. The proposed system requires 56 ms for the prediction of a movement, which is within the reported application time limit of 300 ms for fall prediction, while the MRCP prediction of our system requires approx. 0.95 ms for the prediction of a movement using 124 EEG channels (see Figure 9). The study does not investigate the effect of different combination methods on the classification performance.

In previous works [162,163], we presented preliminary systems for the detection of the P300 and MRCPs, respectively, using FPGA-based systems. However, these systems were inflexible and did not exploit the full parallel processing capabilities to process multiple physiological signals at the same time, so that these systems were only capable to detect either the P300 or MRCP. These systems did not support the fusion of multiple signals, which can improve the classification performance significantly, as shown in this work. Furthermore, these preliminary systems have not been validated in an online evaluation.

Overall, this comparison shows that our FPGA-based system has, similar to other recent hybrid BCI systems, the capability to process EEG and EMG data in combination. However, to the best of our knowledge, the combination of MRCP, EMG and additionally the P300 has not been addressed before. Furthermore, the capabilities of our FPGA-based system can be useful in applications that require miniaturized BCI-systems with high requirements on performance and flexibility that cannot be fulfilled by other previously proposed FPGA-based BCI-systems. Hence, it can help to improve the capabilities of future neurorehabilitation devices.

*6.10. Limitations*

The evaluation of the proposed FPGA-based system shows that it is feasible to use it for movement prediction using EEG and EMG data. However, several limitations of the study should be noted.

First, the study has been conducted under laboratory conditions. Typical rehabilitation applications will be performed under more uncontrolled conditions. Thus, the EEG and EMG will be affected by artifacts and different kinds of noise. We have chosen the present study design to provide the possibility to determine the time of the movement onset exactly and to be able to compare the obtained results to other studies [25,64]. However, the methodology for the detection of the MRCP and P300 has already been successfully applied in other applications, e.g., the usage of MRCP-based movement prediction to enhance the user-experience of an exoskeleton by adapting the control algorithms [58] and the P300 detection for the monitoring of the user's state [57]. Hence, since we also observe no significant performance difference between the CPU and FPGA-based implementations, we assume that the developed system will work under such conditions without major modifications.

Secondly, the current study was conducted using healthy subjects. Although it has been shown that patients that suffer from, e.g., stroke, are feasible to perform movement-imagination despite chronic or severe motor impairments [68,69,164], different kinds of brain damages can affect the EEG [165–170] and EMG [171,172], and hence, possibly the performance of the proposed system. However, MRCP-based BCIs can be successfully used by patients suffering from stroke [173–175]. For other neurological diseases, such as amyotrophic lateral sclerosis, it has been shown that certain MRCP characteristics do not differ from healthy patients [176]. Hence, we assume that the proposed system can be used by patients with neurological motor impairments. Since the system supports different types of signal combinations for movement prediction, it is possible to choose an operation mode depending on the specific conditions of a patient. Nevertheless, the specific difference in classification performance between healthy and impaired subjects depending on impairment and operation mode has to be investigated in the future.

Thirdly, there are mobile systems available with more advanced CPU architectures and higher clock frequencies than the MRS that was used in the experiments for comparison. However, although these might provide a higher computational performance than the MRS, they are less extensible than the proposed FPGA-based system. Due to the low resource utilization of the DFHWAs, we are able to integrate further features like artifact removal methods, eye-tracking [177], physiological signals to monitor fatigue or exhaustion, and also control algorithms and motor commutation components to drive a rehabilitation device directly.

**7. Conclusions and Future Directions**

In this paper, we presented a novel mobile system based on reconfigurable hardware and dataflow computing for the online processing of multiple streams physiological data. The system provides the capability to predict movements based on an online analysis of multiple streams of data in parallel. This can be used to generate hybrid predictions by combining EEG and EMG-based movement predictions or movement predictions with task-related patterns, i.e., the P300. We showed that the hybrid movement prediction can result in a significantly improved classification performance over single predictions. The developed system has been evaluated in a study that consists of 12 subjects in total. It is able to outperform standard CPU-systems regarding prediction time and provides a lower

power consumption than a standard PC. Furthermore, we showed that the fixed-point arithmetic that was used in the FPGA-implementation does not result in a decrease of the classification performance.

The next steps are the integration of the proposed system into an exoskeleton for robotic rehabilitation scenarios [12] and the combination of the control algorithms of the exoskeleton with the movement prediction in a single SoC design. Such a tight integration will allow a real-time adaptation of the control algorithms of the exoskeleton to provide the possibility of an assist-as-needed rehabilitation [29]. Furthermore, in a future study we will evaluate the applicability of the system for patients suffering from movement impairments due to stroke and extend its capabilities with additional features to remove artifacts and monitor the exhaustion of patients.

**Author Contributions:** H.W. and F.K. designed the DFHWA architecture and ZynqBrain; H.W. implemented the DFHWA architecture; H.W. and S.K. analyzed the results; H.W. and M.T. designed and performed the experiments; H.W., M.T., S.K., E.A.K. and F.K. wrote the paper.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

**ASIC**  Application-Specific Integrated Circuit
**BA**  Balanced Accuracy
**BCI**  Brain-Computer Interface
**BRAM**  Block Random Access Memory
**CPU**  Central Processing Unit
**DFHWA**  Dataflow Hardware Accelerator
**DMA**  Direct Memory Access
**DSP**  Digital Signal Processing
**EMG**  Electromyography
**EOG**  Electrooculography
**ERD/ERS**  Event-Related Desynchronization/Synchronization
**EEG**  Electroencephalography
**ERP**  Event-Related Potential
**FIFO**  First-In, First-Out
**FIR**  Finite Impulse Response
**FF**  Flip Flop
**FNR**  False Negative Rate
**FPGA**  Field Programmable Gate Array
**FPR**  False Positive Rate
**GPU**  Graphics Processing Unit
**IIR**  Infinite Impulse Response
**ISI**  Inter-Stimulus Interval
**IRQ**  Interrupt ReQuest
**LUT**  Look-Up Table
**MAC**  Multiply ACcumulate
**ML**  Machine Learning
**MRCP**  Movement Related Cortical Potential
**MRS**  Mobile CPU-based Reference System
**PE**  Processing Element
**PL**  Programmable Logic
**PS**  Processing System
**SDF**  Synchronous Dataflow
**SoC**  System-on-Chip
**SSVEP**  Steady-State Visual Evoked Potential
**SNR**  Signal-to-Noise Ratio
**SVM**  Support Vector Machine
**SRS**  Standard-PC Reference System
**TPR**  True Positive Rate
**TNR**  True Negative Rate

## References

1. Birbaumer, N.; Ghanayim, N.; Hinterberger, T.; Iversen, I.; Kotchoubey, B.; Kübler, A.; Perelmouter, J.; Taub, E.; Flor, H. A spelling device for the paralysed. *Nature* **1999**, *398*, 297–298.

2. Guger, C.; Harkam, W.; Hertnaes, C.; Pfurtscheller, G. Prosthetic control by an EEG-based brain-computer interface (BCI). In Proceedings of the 5th European Conference for the Advancement of Assistive Technology (AATE), Düsseldorf, Germany, 1–4 November 1999; pp. 590–595.

3. Pfurtscheller, G.; Guger, C.; Müller, G.; Krausz, G.; Neuper, C. Brain oscillations control hand orthosis in a tetraplegic. *Neurosci. Lett.* **2000**, *292*, 211–214.

4. Wolpaw, J.; Birbaumer, N.; McFarland, D.; Pfurtscheller, G.; Vaughan, T. Brain-computer interfaces for communication and control. *J. Clin. Neurophysiol.* **2002**, *113*, 767–791.

5. Blankertz, B.; Dornhege, G.; Krauledat, M.; Müller, K.R.; Kunzmann, V.; Losch, F.; Curio, G. The Berlin brain-computer interface: EEG-based communication without subject training. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2006**, *14*, 147–152.

6. Birbaumer, N. Breaking the silence: Brain-computer interfaces (BCI) for communication and motor control. *J. Psychophysiol.* **2006**, *43*, 517–532.

7. Leeb, R.; Keinrath, C.; Friedman, D.; Guger, C.; Scherer, R.; Neuper, C.; Garau, M.; Antley, A.; Steed, A.; Slater, M.; et al. Walking by thinking: The brainwaves are crucial, not the muscles! *Presence* **2006**, *15*, 500–514.

8. Enzinger, C.; Ropele, S.; Fazekas, F.; Loitfelder, M.; Gorani, F.; Seifert, T.; Reiter, G.; Neuper, C.; Pfurtscheller, G.; Müller-Putz, G. Brain motor system function in a patient with complete spinal cord injury following extensive brain–computer interface training. *Exp. Brain. Res.* **2008**, *190*, 215–223.

9. Pfurtscheller, G.; Müller-Putz, G.R.; Scherer, R.; Neuper, C. Rehabilitation with Brain-Computer Interface Systems. *Computer* **2008**, *41*, 58–65.

10. Kirchner, E.; Wöhrle, H.; Bergatt, C.; Kim, S.; Metzen, J.; Kirchner, F. Towards Operator Monitoring via Brain Reading—An EEG-based Approach for Space Applications. In Proceedings of the 10th International Symposium on Artificial Intelligence, Robotics and Automation in Space (iSAIRAS-10), Sapporo, Japan, 29 August–1 September 2010; ESA: Sapporo, Japan, 2010.

11. Kirchner, E.; Kim, S.; Straube, S.; Seeland, A.; Wöhrle, H.; Krell, M.; Tabie, M.; Fahle, M. On the Applicability of Brain Reading for Predictive Human-Machine Interfaces in Robotics. *PLoS ONE* **2013**, *8*, e81732.

12. Kirchner, E.; Albiez, J.; Seeland, A.; Jordan, M.; Kirchner, F. Towards Assistive Robotics for Home Rehabilitation. In Proceedings of the 6th International Conference on Biomedical Electronics and Devices (BIODEVICES-13), Barcelona, Spain, 11–14 February 2013; pp. 168–177.

13. Iosa, M.; Morone, G.; Cherubini, A.; Paolucci, S. The Three Laws of Neurorobotics: A Review on What Neurorehabilitation Robots Should Do for Patients and Clinicians. *J. Med. Biol. Eng.* **2016**, *36*, 1–11.

14. Lum, P.S.; Burgar, C.G.; Shor, P.C.; Majmundar, M.; Van der Loos, M. Robot-assisted movement training compared with conventional therapy techniques for the rehabilitation of upper-limb motor function after stroke. *Arch. Phys. Med. Rehabil.* **2002**, *83*, 952–959.

15. Krebs, H.I.; Volpe, B.T.; Aisen, M.L.; Hening, W.; Adamovich, S.; Poizner, H.; Subrahmanyan, K.; Hogan, N. Robotic Applications in Neuromotor Rehabilitation. *Robotica* **2003**, *21*, 3–11.

16. Kwakkel, G.; Kollen, B.J.; Krebs, H.I. Effects of robot-assisted therapy on upper limb recovery after stroke: A systematic review. *Neurorehabil. Neural Repair* **2008**, *22*, 111–121.

17. Lo, A.C.; Guarino, P.D.; Richards, L.G.; Haselkorn, J.K.; Wittenberg, G.F.; Federman, D.G.; Ringer, R.J.; Wagner, T.H.; Krebs, H.I.; Volpe, B.T.; et al. Robot-Assisted Therapy for Long-Term Upper-Limb Impairment after Stroke. *N. Engl. J. Med.* **2010**, *362*, 1772–1783.

18. Díaz, I.; Gil, J.J.; Sánchez, E. Lower-limb robotic rehabilitation: Literature review and challenges. *J. Robot.* **2011**, *2011*, 759764.

19. Maciejasz, P.; Eschweiler, J.; Gerlach-Hahn, K.; Jansen-Troy, A.; Leonhardt, S. A survey on robotic devices for upper limb rehabilitation. *J. NeuroEng. Rehabil.* **2014**, *11*, 1–29.

20. Novak, D.; Riener, R. A survey of sensor fusion methods in wearable robotics. *Robot. Auton. Syst.* **2015**, *73*, 155–170.

21. Lambercy, O.; Maggioni, S.; Lünenburger, L.; Gassert, R.; Bolliger, M. Robotic and wearable sensor technologies for measurements/clinical assessments. In *Neurorehabilitation Technology*; Springer International Publishing, Cham, Switzerland, 2016; pp. 183–207.

22. Proietti, T.; Crocher, V.; Roby-Brami, A.; Jarrassé, N. Upper-Limb Robotic Exoskeletons for Neurorehabilitation: A Review on Control Strategies. *IEEE Rev. Biomed. Eng.* **2016**, *9*, 4–14.

23. Hogan, N.; Krebs, H. Physically interactive robotic technology for neuromotor rehabilitation. *Prog. Brain. Res.* **2011**, *192*, 59–68.

24. Benitez, L.; Tabie, M.; Will, N.; Schmidt, S.; Jordan, M.; Kirchner, E. Exoskeleton Technology in Rehabilitation: Towards an EMG-Based Orthosis System for Upper Limb Neuromotor Rehabilitation. *J. Robot.* **2013**, *2013*, 610589.

25. Kirchner, E.; Tabie, M.; Seeland, A. Multimodal Movement Prediction—Towards an Individual Assistance of Patients. *PLoS ONE* **2014**, *9*, e85060.

26. López-Larraz, E.; Montesano, L.; Gil-Agudo, Á.; Minguez, J. Continuous decoding of movement intention of upper limb self-initiated analytic movements from pre-movement EEG correlates. *J. NeuroEng. Rehabil.* **2014**, *11*, 1–15.

27. Riener, R.; Novak, D. Movement Onset Detection and Target Estimation for Robot-Aided Arm Training. *Automatisierungstechnik* **2015**, *63*, 286–298.

28. Lopez-Larraz, E.; Trincado-Alonso, F.; Rajasekaran, V.; Perez-Nombela, S.; del Ama, A.J.; Aranda, J.; Minguez, J.; Gil-Agudo, A.; Montesano, L. Control of an Ambulatory Exoskeleton with a Brain-Machine Interface for Spinal Cord Injury Gait Rehabilitation. *Front. Neurosci.* **2016**, *10*, 359.

29. Kirchner, E.; Will, N.; Simnofske, M.; Vaca Benitez, L.V.; Bongardt, B.; Krell, M.; Kumar, S.; Mallwitz, M.; Seeland, A.; Tabie, M.; et al. Recupera-Reha: Exoskeleton Technology with Integrated Biosignal Analysis for Sensorimotor Rehabilitation. In *Smart ASSIST*; Elsevier: Hamburg, Germany, 2016; pp. 504–517.

30. Soekadar, S.R.; Witkowski, M.; Gómez, C.; Opisso, E.; Medina, J.; Cortese, M.; Cempini, M.; Carrozza, M.C.; Cohen, L.G.; Birbaumer, N.; et al. Hybrid EEG/EOG-based brain/neural hand exoskeleton restores fully independent daily living activities after quadriplegia. *Sci. Robot.* **2016**, *5*, doi: 10.1126/scirobotics.aag3296.

31. Schaechter, J.D. Motor rehabilitation and brain plasticity after hemiparetic stroke. *Prog. Neurobiol.* **2004**, *73*, 61–72.

32. Dobkin, B. Brain-computer interface technology as a tool to augment plasticity and outcomes for neurological rehabilitation. *J. Physiol.* **2007**, *579*, 637–642.

33. Ang, K.; Guan, C.; Chua, K.; Ang, B.; Kuah, C.; Wang, C.; Phua, K.; Chin, Z.; Zhang, H. A clinical study of motor imagery-based brain-computer interface for upper limb robotic rehabilitation. In Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Minneapolis, MN, USA, 3–6 September 2009; pp. 5981–5984.

34. Daly, J.; Wolpaw, J. Brain-computer interfaces in neurological rehabilitation. *Lancet Neurol.* **2008**, *7*, 1032–1043.

35. Ang, K.; Chua, K.; Phua, K.; Wang, C.; Chin, Z.; Kuah, C.; Low, W.; Guan, C. A randomized controlled trial of EEG-based motor imagery brain-computer interface robotic rehabilitation for stroke. *Clin. EEG Neurosci.* **2014**, *46*, 310–320.

36. Van Dokkum, L.; Ward, T.; Laffont, I. Brain computer interfaces for neurorehabilitation—Its current status as a rehabilitation strategy post-stroke. *Ann. Phys. Rehabil. Med.* **2015**, *58*, 3–8.

37. Makeig, S.; Gramann, K.; Jung, T.P.; Sejnowski, T.J.; Poizner, H. Linking brain, mind and behavior. *Int. J. Psychophysiol.* **2009**, *73*, 95–100.

38. Kranczioch, C.; Zich, C.; Schierholz, I.; Sterr, A. Mobile EEG and its potential to promote the theory and application of imagery-based motor rehabilitation. *Int. J. Psychophysiol.* **2014**, *91*, 10–15.

39. Edlinger, G.; Guger, C. Laboratory PC and mobile pocket PC brain-computer interface architectures. In Proceedings of the 27th International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Shanghai, China, 17–18 January 2006; Volume 5, pp. 5347–5350.

40. Xilinx Inc. *UG585 Zynq-7000 All Programmable SoC Technical Reference Manual*, 1.10 ed.; Xilinx Inc.: San Jose, CA, USA, 2015.

41. Pfurtscheller, G.; Allison, B.; Bauernfeind, G.; Brunner, C.; Solis Escalante, T.; Scherer, R.; Zander, T.; Mueller-Putz, G.; Neuper, C.; Birbaumer, N. The hybrid BCI. *Front. Neurosci.* **2010**, *4*, 3.

42. Lalitharatne, T.; Teramoto, K.; Hayashi, Y.; Kiguchi, K. Towards hybrid EEG-EMG-based control approaches to be used in bio-robotics applications: Current status, challenges and future directions. *Paladyn* **2013**, *4*, 147–154.

43. Villa-Parra, A.; Broche, L.; Delisle-Rodríguez, D.; Sagaró, R.; Bastos, T.; Frizera-Neto, A. Design of active orthoses for a robotic gait rehabilitation system. *Front. Mech. Eng.* **2015**, *10*, 242–254.

44. Grimm, F.; Walter, A.; Spüler, M.; Naros, G.; Rosenstiel, W.; Gharabaghi, A. Hybrid Neuroprosthesis for the Upper Limb: Combining Brain-Controlled Neuromuscular Stimulation with a Multi-Joint Arm Exoskeleton. *Front. Neurosci.* **2016**, *10*, 367.

45. Tabie, M.; Wöhrle, H.; Kirchner, E.A. Runtime Calibration of Online EEG-based Movement Prediction using EMG Signals. In Proceedings of the International Conference on Bio-inspired Systems and Signal Processing (BIOSIGNALS), Angers, France, 3–6 March 2014; pp. 284–288.

46. Seeland, A.; Tabie, M.; Kim, S.; Kirchner, F.; Kirchner, E.A. Adaptive multimodal biosignal control for exoskeleton supported stroke rehabilitation. In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC-2017), Banff, AB, Canada, 1–4 October 2017.

47. Kirchner, E.A.; Kim, S.K.; Tabie, M.; Wöhrle, H.; Maurus, M.; Kirchner, F. An Intelligent Man-Machine Interface-Multi-Robot Control Adapted for Task Engagement Based on Single-Trial Detectability of P300. *Front. Hum. Neurosci.* **2016**, *10*, 291.

48. Bai, O.; Rathi, V.; Lin, P.; Huang, D.; Battapady, H.; Fei, D.; Schneider, L.; Houdayer, E.; Chen, X.; Hallett, M. Prediction of human voluntary movement before it occurs. *J. Clin. Neurophysiol.* **2011**, *122*, 364–372.

49. Ibáñez, J.; Serrano, J.; del Castillo, M.; Barrios, L.; Gallego, J.; Rocon, E. An EEG-Based Design for the Online Detection of Movement Intention. In *Advances in Computational Intelligence*; Cabestany J., Rojas I., Joya, G., Eds.; Springer: Berlin/Heidelberg, Germany, 2011; pp. 370–377.

50. Lew, E.; Chavarriaga, R.; Silvoni, S.; Millán, J. Detection of self-paced reaching movement intention from EEG signals. *Front. Neuroeng.* **2012**, *5*, doi:10.3389/fneng.2012.00013.

51. Pfurtscheller, G.; Neuper, C. Motor imagery activates primary sensorimotor area in humans. *Neurosci. Lett.* **1997**, *239*, 65–68.

52. Pfurtscheller, G.; Da Silva, F. Event-related EEG/MEG synchronization and desynchronization: Basic principles. *J. Clin. Neurophysiol.* **1999**, *110*, 1842–1857.

53. Stančák, A.; Feige, B.; Lücking, C.; Kristeva-Feige, R. Oscillatory cortical activity and movement-related potentials in proximal and distal movements. *J. Clin. Neurophysiol.* **2000**, *111*, 636–650.

54. Paradiso, G.; Cunic, D.; Saint-Cyr, J.; Hoque, T.; Lozano, A.; Lang, A.; Chen, R. Involvement of human thalamus in the preparation of self-paced movement. *Brain* **2004**, *127*, 2717–2731.

55. Shibasaki, H.; Hallett, M. What is the Bereitschaftspotential? *J. Clin. Neurophysiol.* **2006**, *117*, 2341–2356.

56. Kirchner, E.; Metzen, J.H.; Duchrow, T.; Kim, S.; Kirchner, F. Assisting Telemanipulation Operators via Real-Time Brain Reading. In Proceedings of the Machine Learning in Real-time Applications Workshop, Paderborn, Germany, 15–18 September 2009.

57. Woehrle, H.; Kirchner, E. Online Detection of P300 related Target Recognition Processes During a Demanding Teleoperation Task. In Proceedings of the International Conference on Physiological Computing Systems, (PhyCS 2014), Lissabon, Portugal, 7–9 January 2014; ScitePress: Setúbal, Portugal, 2014; pp. 13–19.

58. Seeland, A.; Woehrle, H.; Straube, S.; Kirchner, E. Online Movement Prediction in a Robotic Application Scenario. In Proceedings of the 6th International IEEE EMBS Conference on Neural Engineering (NER), San Diego, CA, USA, 6–8 November 2013; pp. 41–44.

59. Müller, K.R.; Krauledat, M.; Dornhege, G.; Curio, G.; Blankertz, B. Machine learning techniques for brain-computer interfaces. *Biomed. Tech.* **2004**, *49*, 11–22.

60. Lotte, F.; Congedo, M.; Lécuyer, A.; Lamarche, F.; Arnaldi, B. A review of classification algorithms for EEG-based brain-computer interfaces. *J. Neural Eng.* **2007**, *4*, R1.

61. Krusienski, D.; Grosse-Wentrup, M.; Galain, F.; Coyle, D.; Miller, K.; Forney, E.; Anderson, C. Critical issues in state-of-the-art brain computer interface signal processing. *J. Neural Eng.* **2011**, *8*, 025002.

62. Kleissen, R.; Buurke, J.; Harlaar, J.; Zilvold, G. Electromyography in the biomechanical analysis of human movement and its clinical application. *Gait Posture* **1998**, *8*, 143–158.

63. Reaz, M.B.I.; Hussain, M.S.; Mohd-Yasin, F. Techniques of EMG signal analysis: Detection, processing, classification and applications. *Biol. Proced. Online* **2006**, *8*, 11–35.

64. Tabie, M.; Kirchner, E. EMG Onset Detection-Comparison of different methods for a movement prediction task based on EMG. In Proceedings of the 6th International Conference on Bio-inspired Systems and Signal Processing (BIOSIGNALS-13), Barcelona, Spain, 11–14 February 2013; pp. 242–247.

65. Fleischer, C.; Reinicke, C.; Hommel, G. Predicting the intended motion with EMG signals for an exoskeleton orthosis controller. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Edmonton, AB, Canada, 2–6 August 2005; pp. 2029–2034.

66. Suberbiola, A.; Zulueta, E.; Lopez-Guede, J.M.; Etxeberria-Agiriano, I.; Van Caesbroeck, B. Arm Orthosis/Prosthesis Control Based on Surface EMG Signal Extraction. In *Hybrid Artificial Intelligent Systems*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 510–519.

67. Loconsole, C.; Dettori, S.; Frisoli, A.; Avizzano, C.A.; Bergamasco, M. An EMG-based approach for on-line predicted torque control in robotic-assisted rehabilitation. In Proceedings of the 2014 IEEE Haptics Symposium (HAPTICS), Houston, TX, USA, 23–26 February 2014; pp. 181–186.

68. Johnson, S.; Sprehn, G.; Saykin, A. Intact motor imagery in chronic upper limb hemiplegics: Evidence for activity-independent action representations. *J. Cogn. Neurosci.* **2002**, *14*, 841–852.

69. Ang, K.; Guan, C.; Chua, K.; Ang, B.; Kuah, C.; Wang, C.; Phua, K.; Chin, Z.; Zhang, H. A Large Clinical Study on the Ability of Stroke Patients to Use an EEG-Based Motor Imagery Brain-Computer Interface. *Clin. EEG Neurosci.* **2011**, *42*, 253–258.

70. Liao, L.; Lin, C.; McDowell, K.; Wickenden, A.E.; Gramann, K.; Jung, T.; Ko, L.; Chang, J. Biosensor technologies for augmented brain-computer interfaces in the next decades. *Proc. IEEE* **2012**, *100*, 1553–1566.

71. McFarland, D.J.; Wolpaw, J.R. Brain-Computer Interface Operation of Robotic and Prosthetic Devices. *Computer* **2008**, *41*, 52–56.

72. Rebsamen, B.; Burdet, E.; Zeng, Q.; Zhang, H.; Ang, M.; Teo, C.L.; Guan, C.; Laugier, C. Hybrid P300 and Mu-beta Brain Computer Interface to Operate a Brain Controlled Wheelchair. In Proceedings of the 2nd International Convention on Rehabilitation Engineering & Assistive Technology, Bangkok, Thailand, 13–15 May 2008; pp. 51–55.

73. Gancet, J.; Ilzkovitz, M.; Motard, E.; Nevatia, Y.; Letier, P.; de Weerdt, D.; Cheron, G.; Hoellinger, T.; Seetharaman, K.; Petieau, M.; et al. MINDWALKER: Going one step further with assistive lower limbs exoskeleton for SCI condition subjects. In Proceedings of the 4th IEEE RAS EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), Rome, Italy, 24–27 June 2012; pp. 1794–1800.

74. Do, A.; Wang, P.; King, C.; Chun, S.; Nenadic, Z. Brain-computer interface controlled robotic gait orthosis. *J. NeuroEng. Rehabil.* **2013**, *10*, 111.

75. Looned, R.; Webb, J.; Xiao, Z.; Menon, C. Assisting drinking with an affordable BCI-controlled wearable robot and electrical stimulation: A preliminary investigation. *J. NeuroEng. Rehabil.* **2014**, *11*, 51.

76. Allison, B.; Leeb, R.; Brunner, C.; Müller-Putz, G.; Bauernfeind, G.; Kelly, J.; Neuper, C. Toward smarter BCIs: Extending BCIs through hybridization and intelligent control. *J. Neural. Eng.* **2012**, *9*, 013001.

77. Amiri, S.; Fazel-Rezai, R.; Asadpour, V. A review of hybrid brain-computer interface systems. *Adv. Hum. Comput. Interact.* **2013**, *2013*, 1.

78. Lotte, F.; Guan, C. An efficient P300-based brain-computer interface with minimal calibration time. In Proceedings of the Assistive Machine Learning for People with Disabilities symposium (NIPS'09 Symposium), Vancouver, BC, Canada, 12 December 2009.

79. Wöhrle, H.; Kirchner, E. Online Classifier Adaptation for the Detection of P300 Target Recognition Processes in a Complex Teleoperation Scenario. In *Physiological Computing Systems*; da Silva, H., Holzinger, A., Fairclough, S., Majoe, D., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; pp. 105–118.

80. Su, Y.; Qi, Y.; Luo, J.; Wu, B.; Yang, F.; Li, Y.; Zhuang, Y.; Zheng, X.; Chen, W. A hybrid brain-computer interface control strategy in a virtual environment. *J. Zhejiang Univ.* **2011**, *12*, 351–361.

81. Riechmann, H.; Hachmeister, N.; Ritter, H.; Finke, A. Asynchronous, parallel on-line classification of P300 and ERD for an efficient hybrid BCI. In Proceedings of the 5th International IEEE/EMBS Conference on Neural Engineering (NER), Cancun, Mexico, 27 April–1 May 2011; pp. 412–415.

82. Leeb, R.; Sagha, H.; Chavarriaga, R.; del R Millán, J. A hybrid brain–computer interface based on the fusion of electroencephalographic and electromyographic activities. *J. Neural Eng.* **2011**, *8*, 025011.

83. Kiguchi, K.; Hayashi, Y. Motion Estimation Based on EMG and EEG Signals to Control Wearable Robots. In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, Manchester, UK, 13–16 October 2013; pp. 4213–4218.

84. Novak, D.; Omlin, X.; Leins-Hess, R.; Riener, R. Predicting targets of human reaching motions using different sensing technologies. *IEEE Trans. Biomed. Eng.* **2013**, *60*, 2645–2654.

85. De Venuto, D.; Annese, V.F.; de Tommaso, M.; Vecchio, E.; Sangiovanni Vincentelli, A.L. Combining EEG and EMG Signals in a Wireless System for Preventing Fall in Neurodegenerative Diseases. In *Ambient Assisted Living: Italian Forum 2014*; Springer International Publishing, Cham, Switzerland, 2015; pp. 317–327.

86. Kawase, T.; Sakurada, T.; Koike, Y.; Kansaku, K. A hybrid BMI-based exoskeleton for paresis: EMG control for assisting arm movements. *J. Neural Eng.* **2017**, *14*, 016015.

87. Li, X.; Samuel, O.; Zhang, X.; Wang, H.; Fang, P.; Li, G. A motion-classification strategy based on sEMG-EEG signal combination for upper-limb amputees. *J. NeuroEng. Rehabil.* **2017**, *14*, 2.

88. Corbett, E.A.; Sachs, N.A.; Körding, K.P.; Perreault, E.J. Multimodal decoding and congruent sensory information enhance reaching performance in subjects with cervical spinal cord injury. *Front. Neurosci.* **2014**, *8*, 123.

89. Gargiulo, G.; Bifulco, P.; Calvo, R.; Cesarelli, M.; Jin, C.; Schaik, A. A mobile EEG system with dry electrodes. In Proceedings of the IEEE Biomedical Circuits and Systems Conference, Baltimore, MD, USA, 20–22 November 2008; pp. 273–276.

90. Liao, L.D.; Wang, I.J.; Chen, S.F.; Chang, J.Y.; Lin, C.T. Design, fabrication and experimental validation of a novel dry-contact sensor for measuring electroencephalography signals without skin preparation. *Sensors* **2011**, *11*, 5819–5834.

91. Lin, C.T.; Liao, L.D.; Liu, Y.H.; Wang, I.J.; Lin, B.S.; Chang, J.Y. Novel dry polymer foam electrodes for long-term EEG measurement. *IEEE Trans. Biomed. Eng.* **2011**, *58*, 1200–1207.

92. Vos, M.D.; Kroesen, M.; Emkes, R.; Debener, S. P300 speller BCI with a mobile EEG system: Comparison to a traditional amplifier. *J. Neural Eng.* **2014**, *11*, 036008.

93. Ehinger, B.; Fischer, P.; Gert, A.; Kaufhold, L.; Weber, F.; Pipa, G.; König, P. Kinesthetic and vestibular information modulate alpha activity during spatial navigation: A mobile EEG study. *Front. Hum. Neurosci.* **2014**, *8*, 71.

94. Lo, C.C.; Chien, T.Y.; Chen, Y.C.; Tsai, S.H.; Fang, W.C.; Lin, B.S. A Wearable Channel Selection-Based Brain-Computer Interface for Motor Imagery Detection. *Sensors* **2016**, *16*, 213.

95. Matthews, R.; Turner, P.; McDonald, N.J.; Ermolaev, K.; Mc Manus, T.; Shelby, R.A.; Steindorf, M. Real time workload classification from an ambulatory wireless EEG system using hybrid EEG electrodes. In Proceedings of the 2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Vancouver, BC, Canada, 20–25 August 2008; pp. 5871–5875.

96. Lin, C.T.; Chen, Y.C.; Huang, T.Y.; Chiu, T.T.; Ko, L.W.; Liang, S.F.; Hsieh, H.Y.; Hsu, S.H.; Duann, J.R. Development of Wireless Brain Computer Interface With Embedded Multitask Scheduling and its Application on Real-Time Driver's Drowsiness Detection and Warning. *IEEE Trans. Biomed. Eng.* **2008**, *55*, 1582–1591.

97. Lin, C.T.; Ko, L.W.; Chiou, J.C.; Duann, J.R.; Huang, R.S.; Liang, S.F.; Chiu, T.W.; Jung, T.P. Noninvasive Neural Prostheses Using Mobile and Wireless EEG. *Proc. IEEE* **2008**, *96*, 1167–1183.

98. Riera, A.; Dunne, S.; Cester, I.; Ruffini, G. STARFAST: A wireless wearable EEG/ECG biometric system based on the ENOBIO sensor. In Proceedings of the 6th International Workshop on Wearable Micro- and Nanosystems for Personalised Health, Oslo, Norway, 24–26 June 2008.

99. Lin, C.T.; Ko, L.W.; Chang, M.H.; Duann, J.R.; Chen, J.Y.; Su, T.P.; Jung, T.P. Review of wireless and wearable electroencephalogram systems and brain-computer interfaces—A mini-review. *Gerontology* **2009**, *56*, 112–119.

100. Lin, C.T.; Ko, L.W.; Chang, C.J.; Wang, Y.T.; Chung, C.H.; Yang, F.S.; Duann, J.R.; Jung, T.P.; Chiou, J.C. Wearable and wireless brain-computer interface and its applications. In Proceedings of the International Conference on Foundations of Augmented Cognition, San Diego, CA, USA, 9–24 July 2009; pp. 741–748.

101. Brown, L.; van de Molengraft, J.; Yazicioglu, R.F.; Torfs, T.; Penders, J.; Van Hoof, C. A low-power, wireless, 8-channel EEG monitoring headset. In Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology (EMBC), Buenos Aires, Argentina, 31 August–4 September 2010; pp. 4197–4200.

102. Tolbert, J.R.; Kabali, P.; Brar, S.; Mukhopadhyay, S. An accuracy aware low power wireless EEG unit with information content based adaptive data compression. In Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Minneapolis, MN, USA, 3–6 September 2009; pp. 5417–5420.

103. Higgins, G.; Faul, S.; McEvoy, R.; McGinley, B.; Glavin, M.; Marnane, W.; Jones, E. EEG compression using JPEG2000: How much loss is too much? In Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology (EMBC), Buenos Aires, Argentina, 31 August–4 September 2010; pp. 614–617.

104. Casson, A.J.; Yates, D.C.; Smith, S.J.M.; Duncan, J.S.; Rodriguez-Villegas, E. Wearable Electroencephalography. What is it, why is it needed, and what does it entail? *IEEE Eng. Med. Biol. Mag.* **2010**, *29*, 44–56.

105. Wang, Y.; Wang, Y.; Cheng, C.; Jung, T. Developing stimulus presentation on mobile devices for a truly portable SSVEP-based BCI. In Proceedings of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Osaka, Japan, 3–7 July 2013; pp. 5271–5274.

106. Webb, J.; Xiao, Z.G.; Aschenbrenner, K.; Herrnstadt, G.; Menon, C. Towards a portable assistive arm exoskeleton for stroke patient rehabilitation controlled through a brain computer interface. In Proceedings of the 4th IEEE RAS EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), Rome, Italy, 24–27 June 2012; pp. 1299–1304.

107. Shyu, K.; Lee, P.; Lee, M.; Lin, M.; Lai, R.; Chiu, Y. Development of a low-cost FPGA-based SSVEP BCI multimedia control system. *IEEE Trans. Biomed. Circuits Syst.* **2010**, *4*, 125–132.

108. Khurana, K.; Gupta, P.; Panicker, R.; Kumar, A. Development of an FPGA-based real-time P300 speller. In Proceedings of the 22nd International Conference on Field Programmable Logic and Applications (FPL), Oslo, Norway, 29–31 August 2012; pp. 551–554.

109. Shyu, K.; Chiu, Y.; Lee, P.; Lee, M.; Sie, J.; Wu, C.; Wu, Y.; Tung, P. Total Design of an FPGA-Based Brain-Computer Interface Control Hospital Bed Nursing System. *IEEE Trans. Ind. Electron.* **2013**, *60*, 2731–2739.

110. Annese, V.F.; Crepaldi, M.; Demarchi, D.; de Venuto, D. A digital processor architecture for combined EEG/EMG falling risk prediction. In Proceedings of the Design, Automation Test in Europe Conference & Exhibition (DATE), Dresden, Germany, 14–18 March 2016; pp. 714–719.

111. Belwafi, K.; Ghaffari, F.; Djemal, R.; Romain, O. A Hardware/Software Prototype of EEG-based BCI System for Home Device Control. *J. Signal. Process. Syst.* **2016**, 1–17.

112. Chi, Y.; Wang, Y.; Wang, Y.; Maier, C.; Jung, T.; Cauwenberghs, G. Dry and Noncontact EEG Sensors for Mobile Brain-Computer Interfaces. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2012**, *20*, 228–235.

113. Page, A.; Sagedy, C.; Smith, E.; Attaran, N.; Oates, T.; Mohsenin, T. A Flexible Multichannel EEG Feature Extractor and Classifier for Seizure Detection. *IEEE Trans. Circuits Syst. II Express Briefs* **2015**, *62*, 109–113.

114. Bailey, D. *Design for Embedded Image Processing on FPGAs*, 1st ed.; John Wiley & Sons, Hoboken, NJ, USA, 2011.

115. McAllister, J., FPGA-based DSP. In *Handbook of Signal Processing Systems*; Springer: New York, NY, USA, 2013; pp. 363–392.

116. Vanderbauwhede, W.; Benkrid, K. *High-Performance Computing Using FPGAs*; Springer: New York, NY, USA, 2013.

117. García, G.; Jara, C.; Pomares, J.; Alabdo, A.; Poggi, L.; Torres, F. A survey on FPGA-Based sensor systems: Towards intelligent and reconfigurable low-power sensors for computer vision, control and signal processing. *Sensors* **2014**, *14*, 6247–6278.

118. Trimberger, S. Three Ages of FPGAs: A Retrospective on the First Thirty Years of FPGA Technology. *Proc. IEEE* **2015**, *103*, 318–331.

119. Tessier, R.; Pocek, K.; DeHon, A. Reconfigurable Computing Architectures. *Proc. IEEE* **2015**, *103*, 332–354.

120. Xilinx Inc. *UG473 7 Series FPGAs Memory Resources User Guide*, 1.11 ed.; Xilinx Inc.: San Jose, CA, USA, 2014.

121. Xilinx Inc. *UG479 7 Series DSP48E1 Slice User Guide*, 1.8th ed.; Xilinx Inc.: San Jose, CA, USA, 2014.

122. Altera Corp. *Altera's User-Customizable ARM-Based SoC*; Altera Corp.: San Jose, CA, USA, 2014.

123. Krell, M.; Straube, S.; Seeland, A.; Wöhrle, H.; Teiwes, J.; Metzen, J.H.; Kirchner, E.; Kirchner, F. pySPACE—A signal processing and classification environment in Python. *Front. Neuroinf.* **2013**, *7*, doi:10.3389/fninf.2013.00040.

124. Renard, Y.; Lotte, F.; Gibert, G.; Congedo, M.; Maby, E.; Delannoy, V.; Bertrand, O.; Lécuyer, A. Openvibe: An open-source software platform to design, test, and use brain–computer interfaces in real and virtual environments. *Presence* **2010**, *19*, 35–53.

125. Martin, G.; Smith, G. High-level synthesis: Past, present, and future. *IEEE Des. Test* **2009**, *26*, 18–25.

126. Windh, S.; Ma, X.; Halstead, R.J.; Budhkar, P.; Luna, Z.; Hussaini, O.; Najjar, W.A. High-Level Language Tools for Reconfigurable Computing. *Proc. IEEE* **2015**, *103*, 390–408.

127. Skalicky, S.; Wood, C.; Łukowiak, M.; Ryan, M. High level synthesis: Where are we? A case study on matrix multiplication. In Proceedings of the International Conference on Reconfigurable Computing and FPGAs (ReConFig), Cancun, Mexico, 9–11 December 2013; pp. 1–7.

128. Dennis, J. First version of a data flow procedure language. In *Programming Symposium*; Lecture Notes in Computer Science; Robinet, B., Ed.; Springer: Berlin/Heidelberg, Germany, 1974; Volume 19, pp. 362–376.

129. Lee, E.; Messerschmitt, D. Synchronous data flow. *Proc. IEEE* **1987**, *75*, 1235–1245.

130. Lee, E.A.; Parks, T.M. Dataflow process networks. *Proc. IEEE* **1995**, *83*, 773–801.

131. Bhattacharyya, S.; Deprettere, E.; Leupers, R.; Takala, J. *Handbook of Signal Processing Systems*; Springer: New York, NY, USA, 2013.

132. Wöhrle, H.; Kirchner, F. Reconfigurable Hardware-Based Acceleration for Machine Learning and Signal Processing. In *Formal Modeling and Verification of Cyber-Physical Systems*; Lecture Notes in Computer Science; Drechsler, R., Kühne, U., Eds.; , Springer: Wiesbaden, Germany, 2015.

133. Williamson, M.C.; Lee, E.A. Synthesis of parallel hardware implementations from synchronous dataflow graph specifications. In Proceedings of the Conference Record of the 30th Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 3–6 November 1996; pp. 1340–1343.

134. Roquier, G.; Bezati, E.; Mattavelli, M. Hardware and software synthesis of heterogeneous systems from dataflow programs. *J. Electr. Comp. Eng.* **2012**, *2012*, 484962.

135. Wöhrle, H.; Teiwes, J.; Krell, M.; Seeland, A.; Kirchner, E.; Kirchner, F. Reconfigurable Dataflow Hardware Accelerators for Machine Learning and Robotics. In Proceedings of European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, Nancy, France, 15–19 September 2014.

136. Yazdanpanah, F.; Alvarez-Martinez, C.; Jimenez-Gonzalez, D.; Etsion, Y. Hybrid dataflow/von-Neumann architectures. *IEEE Trans. Parallel Distrib. Syst.* **2014**, *25*, 1489–1509.

137. Zenzes, M.; Kampmann, P.; Stark, T.; Schilling, M. NDLCom: Simple Protocol for Heterogeneous Embedded Communication Networks. In Proceedings of the Embedded World Exhibition & Conference, Nuremberg, Germany, 23–25 February 2016.

138. Bartsch, S.; Manz, M.; Kampmann, P.; Dettmann, A.; Hanff, H.; Langosz, M.; von Szadkowski, K.; Hilljegerdes, J.; Simnofske, M.; Kloss, P.; et al. Development and Control of the Multi-Legged Robot Mantis. In Proceedings of the 47th International Symposium on Robotics (ISR-2016), Munich, Germany, 21–22 June 2016.

139. Rivet, B.; Souloumiac, A.; Attina, V.; Gibert, G. xDAWN Algorithm to Enhance Evoked Potentials: Application to Brain Computer Interface. *IEEE Trans. Biomed. Eng.* **2009**, *56*, 2035–2043.

140. Wöhrle, H.; Krell, M.; Straube, S.; Kim, S.; Kirchner, E.; Kirchner, F. An Adaptive Spatial Filter for User-Independent Single Trial Detection of Event-Related Potentials. *IEEE Trans. Biomed. Eng.* **2015**, *62*, 1696–1705.

141. van der Walt, S.; Colbert, S.; Varoquaux, G. The NumPy Array: A Structure for Efficient Numerical Computation. *IEEE Comput. Sci. Eng.* **2011**, *13*, 22–30.

142. Jones, E.; Oliphant, T.; Peterson, P. SciPy: Open Source Scientific Tools For Python. 2001. Available online: http://www.scipy.org (accessed on 21 January 2017).

143. Seeland, A.; Manca, L.; Kirchner, F.; Kirchner, E. Spatio-temporal Comparison Between ERD/ERS and MRCP-based Movement Prediction. In Proceedings of the 8th International Conference on Bio-inspired Systems and Signal Processing (BIOSIGNALS-15), Lisbon, Portugal, 12–15 January 2015.

144. Nikolic, M.; Krarup, C. EMGTools, an Adaptive and Versatile Tool for Detailed EMG Analysis. *IEEE Trans. Biomed. Eng.* **2011**, *58*, 2707–2718.

145. Krell, M.; Tabie, M.; Wöhrle, H.; Kirchner, E. Memory and Processing Efficient Formula for Moving Variance Calculation in EEG and EMG Signal Processing. In Proceedings of the International Congress on Neurotechnology, Electronics and Informatics (NEUROTECHNIX 2013), Vilamoura, Portugal, 16–17 November 2015; pp. 41–45.

146. Semmaoui, H.; Drolet, J.; Lakhssassi, A.; Sawan, M. Setting Adaptive Spike Detection Threshold for Smoothed TEO Based on Robust Statistics Theory. *IEEE Trans. Biomed. Eng.* **2012**, *59*, 474–482.

147. Dick, C.; Harris, F. FPGA signal processing using sigma-delta modulation. *IEEE Signal Process. Mag.* **2000**, *17*, 20–35.

148. Crochiere, R.; Rabiner, L. Optimum FIR digital filter implementations for decimation, interpolation, and narrow-band filtering. *IEEE Trans. Acoust. Speech Signal Process.* **1975**, *23*, 444–456.

149. Bellanger, M. Computation rate and storage estimation in multirate digital filtering with half-band filters. *IEEE Trans. Acoust. Speech Signal. Process.* **1977**, *25*, 344–346.

150. Jansen, B.; Allam, A.; Kota, P.; Lachance, K.; Osho, A.; Sundaresan, K. An exploratory study of factors affecting single trial P300 detection. *IEEE Trans. Biomed. Eng.* **2004**, *51*, 975–978.

151. Ghaderi, F.; Kim, S.; Kirchner, E. Effects of eye artifact removal methods on single trial P300 detection, a comparative study. *J. Neurosci. Methods* **2014**, *221*, 41–47.

152. Wöhrle, H.; Kirchner, F. CAEMO—A high performance matrix coprocessor for SoCs. *Microprocess. Microsyst.* **2017**, submitted.

153. Crammer, K.; Dekel, O.; Keshet, J.; Shalev-Shwartz, S.; Singer, Y. Online Passive-Aggressive Algorithms. *J. Mach. Learn. Res.* **2006**, *7*, 551–585.

154. Cawley, G.C.; Talbot, N. On over-fitting in model selection and subsequent selection bias in performance evaluation. *J. Mach. Learn. Res.* **2010**, *11*, 2079–2107.

155. Metzen, J.; Kirchner, E. Rapid Adaptation of Brain Reading Interfaces based on Threshold Adjustment. In Proceedings of the 35th Annual Conference of the German Classification Society, (GfKl-2011), Farankfurt, Germany, 30 August–2 September 2011; p. 138.

156. Straube, S.; Krell, M. How to evaluate an agent's behaviour to infrequent events? Reliable performance estimation insensitive to class distribution. *Front. Comput. Neurosci.* **2014**, *8*, doi:10.3389/fncom.2014.00043.

157. OpenMP Architecture Review Board. OpenMP Application Program Interface, Version 3.0. May 2008. Available online: http://www.openmp.org/wp-content/uploads/spec30.pdf (accessed on 14 December 2016).

158. Kornhuber, H.; Deecke, L. Hirnpotentialaänderungen bei Willkürbewegungen und passiven Bewegungen des Menschen: Bereitschaftspotential und reafferente Potentiale. *Pflüg. Arch. Eur. J. Physiol.* **1965**, *284*, 1–17. (In Germany)

159. Deecke, L.; Scheid, P.; Kornhuber, H. Distribution of readiness potential, pre-motion positivity, and motor potential of the human cerebral cortex preceding voluntary finger movements. *Exp. Brain Res.* **1969**, *7*, 158–168.

160. Wang, Y.; Makeig, S. Predicting intended movement direction using EEG from human posterior parietal cortex. In *International Conference on Foundations of Augmented Cognition*, San Diego, CA, USA, 19–24 July 2009; pp. 437–446.

161. Lew, E.Y.L.; Chavarriaga, R.; Silvoni, S.; Millán, J.d.R. Single trial prediction of self-paced reaching directions from EEG signals. *Front. Neurosci.* **2014**, *8*, 222.

162. Wöhrle, H.; Teiwes, J.; Kirchner, E.A.; Kirchner, F. A Framework for High Performance Embedded Signal Processing and Classification of Psychophysiological Data. In Proceedings of the International Conference on Biomedical Engineering and Technology (ICBET), Copenhagen, Denmark, 19–20 May 2013; pp. 60–66.

163. Wöhrle, H.; Teiwes, J.; Tabie, M.; Seeland, A.; Kirchner, E.A.; Kirchner, F. Prediction of Movements by Online Analysis of Electroencephalogram with Dataflow Accelerators. In Proceedings of the International Congress on Neurotechnology, Electronics and Informatics (NEUROTECHNIXX), Rome, Italy, 25–26 November 2014; pp. 31–37.

164. Johnson, S.H. Imagining the impossible: Intact motor representations in hemiplegics. *Neuroreport* **2000**, *11*, 729–732.

165. Platz, T.; Kim, I.H.; Pintschovius, H.; Winter, T.; Kieselbach, A.; Villringer, K.; Kurth, R.; Mauritz, K.H. Multimodal EEG analysis in man suggests impairment-specific changes in movement-related electric brain activity after stroke. *Brain* **2000**, *123*, 2475–2490.

166. Kaiser, V.; Daly, I.; Pichiorri, F.; Mattia, D.; Müller-Putz, G.; Neuper, C. Relationship between electrical brain responses to motor imagery and motor impairment in stroke. *Stroke* **2012**, *43*, 2735–2740.

167. Gong, W.; Zhang, T.; Shan, L. Cortical lateralization in stroke patients measured by event-related potentials during motor imagery. *Mol. Med. Rep.* **2013**, *8*, 1701–1707.

168. Castro, A.; Diaz, F.; Sumich, A. Long-term neuroplasticity in spinal cord injury patients: A study on movement-related brain potentials. *Int. J. Psychophysiol* **2013**, *87*, 205–214.

169. Xu, R.; Jiang, N.; Vuckovic, A.; Hasan, M.; Mrachacz-Kersting, N.; Allan, D.; Fraser, M.; Nasseroleslami, B.; Conway, B.; Dremstrup, K.; Farina, D. Movement-related cortical potentials in paraplegic patients: Abnormal patterns and considerations for BCI-rehabilitation. *Front. Neuroeng.* **2014**, *7*, 35.

170. Park, W.; Kwon, G.H.; Kim, Y.H.; Lee, J.H.; Kim, L. EEG response varies with lesion location in patients with chronic stroke. *J. NeuroEng. Rehabil.* **2016**, *13*, 21.

171. Dewald, J.; Sheshadri, V.; Dawson, M.; Beer, R. Upper-limb discoordination in hemiparetic stroke: Implications for neurorehabilitation. *Top. Stroke Rehabil.* **2001**, *8*, 1–12.

172. Cesqui, B.; Tropea, P.; Micera, S.; Krebs, H. EMG-based pattern recognition approach in post stroke robot-aided rehabilitation: A feasibility study. *J. Neuroeng. Rehabil.* **2013**, *10*, 75.

173. Jochumsen, M.; Niazi, I.; Mrachacz-Kersting, N.; Jiang, N.; Farina, D.; Dremstrup, K. Comparison of spatial filters and features for the detection and classification of movement-related cortical potentials in healthy individuals and stroke patients. *J. Neural Eng.* **2015**, *12*, 056003.

174. Jochumsen, M.; Niazi, I.; Taylor, D.; Farina, D.; Dremstrup, K. Detecting and classifying movement-related cortical potentials associated with hand movements in healthy subjects and stroke patients from single-electrode, single-trial EEG. *J. Neural Eng.* **2015**, *12*, 056013.

175. Kamavuako, E.; Jochumsen, M.; Niazi, I.; Dremstrup, K. Comparison of features for movement prediction from single-trial movement-related cortical potentials in healthy subjects and stroke patients. *Comput. Intell. Neurosci.* **2015**, *2015*, 71.

176. Gu, Y.; Farina, D.; Murguialday, A.R.; Dremstrup, K.; Birbaumer, N. Comparison of movement related cortical potential in healthy people and amyotrophic lateral sclerosis patients. *Front. Neurosci.* **2013**, *7*, 65.

177. Ando, T.; Moshnyaga, V.; Hashimoto, K. A low-power FPGA implementation of eye tracking. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Kyoto, Japan, 25–30 March 2012; pp. 1573–1576.